VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"Jnana Sangama" Belagavi – 590 010

PROJECT REPORT ON

"DESIGN AND IMPLEMENTATION OF MICRO WEATHER STATION"

Submitted in partial fulfillment of the requirements for the award of degree

BACHELOR OF ENGINEERING
IN
ELECTRONICS & COMMUNICATION ENGINEERING

Submitted By

Name	USN
Abhishek M Shastry K	4AL17EC002
Abhishek Vasudev Mahendrakar	4AL17EC003
Akshay	4AL17EC008
K Muthu	4AL17EC038

Under the Guidance of
Mrs. Vijetha T S

Asst. Professor

Department of E&C Engineering

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

ALVA'S INSTITUTE OF ENGINEERING & TECHNOLOGY

MOODBIDRI – 574 225.

2020-2021

VISVESVARAYA TECHNOLOGICAL UNIVERSITY

“Jnana Sangama” Belagavi – 590 010

PROJECT REPORT ON

“DESIGN AND IMPLEMENTATION OF MICRO WEATHER STATION”

Submitted in partial fulfillment of the requirements for the award of degree

BACHELOR OF ENGINEERING
IN
ELECTRONICS & COMMUNICATION ENGINEERING

Submitted By

Name	USN
Abhishek M Shastry K	4AL17EC002
Abhishek Vasudev Mahendrakar	4AL17EC003
Akshay	4AL17EC008
K Muthu	4AL17EC038

Under the Guidance of
Mrs. Vijetha T S
Asst. Professor
Department of E&C Engineering

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

ALVA’S INSTITUTE OF ENGINEERING & TECHNOLOGY

MOODBIDRI – 574 225.

2020-2021

ALVA'S INSTITUTE OF ENGINEERING & TECHNOLOGY

MOODBIDRI – 574 225

(Affiliated to VTU, **BELAGAVI**)

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

CERTIFICATE

*Certified that the project work entitled "***DESIGN AND IMPLEMENTATION OF MICRO WEATHER STATION***" is a bona fide work carried out by*

Abhishek M Shastry K	**4AL17EC002**
Abhishek Vasudev Mahendrakar	**4AL17EC003**
Akshay	**4AL17EC008**
K Muthu	**4AL17EC038**

in partial fulfillment for the award of **BACHELOR OF ENGINEERING** in **ELECTRONICS & COMMUNICATION ENGINEERING** of the **VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI** during the year 2020–2021. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of Project work prescribed for the Bachelor of Engineering Degree.

Signature of the Guide	**Signature of the H.O.D**	**Signature of the Principal**
Mrs. Vijetha T S	**Dr. D V Manjunatha**	**Dr. Peter Fernandes**

EXTERNAL VIVA

Name of the Examiners — Signature with date

1……………………………............. ....

2……………………………............. ....

ABSTRACT

The advancement of Internet of Things (IoT) has made a major impact on technology. It collects an enormous amount of data which can be used in an application. One such system is Micro Weather Station, which gives us different environment variable values such as temperature, humidity, soil moisture, Ultraviolet (UV) radiation, air pressure, air quality, and rainfall. Initially research or advancements done on weather monitoring were limited, but over the last century it has evolved into a well-organized and professional global activity that reflects its crucial importance for a wide range of economic, environmental, civil protection and farming activities. Due to human activities these days there is a drastic change in the climate, hence an accurate and cost-efficient system is needed which is used to monitor the changes in the environment. The application of weather station is not just bounded for getting the live data and prediction of weather, but also includes the advancement in the agricultural sector and the military applications. Cloud storage technology and Geo-tagging have made it much simpler to get the data of any place at any time.

ACKNOWLEDGEMENT

The project of any research work depends so much on: the quality of education received the quality of teachers, research resources and enabling and encouraging environment. Studying in **Alva's Institute of Engineering and Technology**, Mijar provides all these above-mentioned facilities which have made possible the successful outcome of this research work.

Firstly, our gratitude goes to our guide, **Mrs. Vijetha T S.**, Assistant Professor, Department of Electronics and Communication, AIET, who is our source of encouragement and motivation throughout this project. Without her valuable guidance, this work would never have been a successful one.

We would like to express our heartily gratitude to our Project coordinator, **Mr. Santhosh S.**, Assistant Professor, Department of Electronics and Communication, AIET, for his consistent guidance, regular source of encouragement and assistance throughout this project.

We would like to express our sincere gratitude to our Head of the Department of Electronics & Communication Engineering, **Dr. D V Manjunatha** for his guidance and inspiration.

We would like to thank our Principal **Dr. Peter Fernandes** for providing all the facilities and a proper environment to work in the college campus.

We are indebted to **Management of Alva's Education Foundation, Moodbidri** for providing an environment which helped us in completing our project.

We are thankful to all the teaching and non-teaching staff members of Department of Electronics & Communication Engineering for their help and needed support rendered throughout the project.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ADC	Analog to Digital Converter
AES	Advanced Encryption Standard
AHP	Analytical Hierarchy Process
API	Application Programming Interface
APRS	Automatic Packet Reporting System
AQI	Air Quality Index
ASIC	Application Specific Integrated Circuit
AWS	Automatic Weather Station
BLE	Bluetooth Low Energy
COSMO	Consortium for Small-scale Modelling
CPU	Central Processing Unit
CSV	Comma Separated Values
DAC	Digital to Analog Converter
EMS	Environmental Monitoring System
FTP	File Transfer Protocol
GND	Ground
GPIO	General Purpose Input Output
GPR	Gaussian Process Regression
GPRS	General Packet Radio service
GPS	Global Positioning System
GSM	Global System for Mobile Communication
GUI	Graphical User Interface
HDMI	High-Definition Multimedia Interface

HTML	Hypertext Markup Language
IC	Integrated Circuit
IDE	Integrated Development Environment
IFrame	Inline Frame
IITH	Indian Institute of Technology, Hyderabad
IMD	India Meteorological Department
IOT	Internet of Things
JS	JavaScript
JSON	JavaScript Object Notation
LCD	Liquid Crystal Display
LDR	Light Dependent Resistor
LED	Light Emitting Diode
LGA	Land Grid Array
LoRaWAN	Long Range Wide Area Network
LPG	Liquefied Petroleum Gas
LR	Logistic Regression
MA	Moving Average
MLP	Multi-Layer Perceptron
MOS	Metal Oxide Semiconductor
MQTT	Message Queuing Telemetry Transport Protocol
MWFS	Mesoscale Weather Forecasting System
MWS	Micro Weather Station
NOAA	National Oceanic and Atmospheric Administration
NodeMCU	Node Micro Controller Unit
OS	Operating System
PHP	Hypertext Preprocessor

POE	Power Over Ethernet
PPM	Parts Per Million
RES	Renewable Energy Source
RISC	Reduced Instruction Set Computer
RN	React Native
RTU	Remote Terminal Unit
SCL	Serial Clock
SD	Secure Digital
SDA	Serial Data
SNR	Signal to Noise Ratio
SPI	Serial Peripheral Interface
SQL	Structured Query Language
SRTS	Soft Real-Time System
SVM	Support Vector Machine
TTL	Transistor to Transistor Logic
UDP	User Datagram Protocol
UI	User Interface
UniDalog	Universal Data Logger
UV	Ultra-Violet
VS	Visual Studio
WRF	Weather Research and Forecasting
WSN	Wireless Sensor Network
WSP	Weather Service Provider

INTRODUCTION

CHAPTER 1

INTRODUCTION

1.1 Prelude

Micro weather station is a system which measures the environmental variables and it is transferred to server where the computations are made and displays the result in web page or mobile applications. The sole purpose of micro weather station is to gather/collect major weather attributes like temperature, humidity, soil moisture, Ultra-Violet (UV) radiation, air pressure and air quality at remote location with most effective and cost-efficient manner with the help of micro weather station equipment. This uses multiple micro weather stations and aggregates the data provided by multiple data collection points in real time. Raspberry Pi is interfaced with various sensors using General Purpose Input Output (GPIO) pins. The Raspberry Pi has an inbuilt SD card slot that helps to fetch and store data. All the data sensed are time tagged and then pushed to the cloud, where the required calculations are made and displays the data in terms of table/graph with the help of web application in real-time. This enables the user an easy and reliable way to understand the data and make further decisions accordingly. The data collected over time can be used for various purposes such as weather prediction, study the change in weather pattern and for research purposes.

Modern weather tools didn't begin development until the 1400's. Before this, weather observation was extremely rudimentary, mostly based on the appearance of the sky and the feel of the air. Much of the development of these weather tools was not just necessitated by agriculture, but also due to an increase in sea travel. Because storms at sea can be deadly, and ships were propelled by the wind, the ability to predict weather conditions relevant to sailing was extremely important.

Weather observation as we know it truly began to be developed in the early to mid-1400’s. In 1441, a Korean crown prince named Munjong of Joseon developed the first standard rain gauge. Following developments in weather monitoring included the first water thermometer in 1593, the first practical humidity measurement system in 1664, and the first barometer in 1643. Over time, these weather tools were further improved and refined.

By the 18th century, recording of weather became more standard, with Gabriel Fahrenheit and Anders Celsius giving us some standard ways we measure temperature that are still with us today. By the early 1800s, Luke Howard began to formulate cloud types in a standard form that allowed better understand of subsequent patterns in rainfall or atmospheric moisture. By 1817, Alexander Humboldt came up with the idea of climatic maps to indicate the pattern of average global temperature. This began the modern measurement and monitoring of global temperatures that have affected our own understanding of change such as global climate change. The 1830s and 1840s also saw many new types of equipment being utilized for weather monitoring, including measuring barometric pressure, use of storm clocks to measure meteorological data, and use of early cameras for weather observation.

In the late 19th century, more complex and larger weather phenomena were better understood, such as hurricanes. Weather monitoring stations were utilized in the Atlantic to begin with, to record and provide warning about hurricane developments. By 1904, Australia developed a unified meteorological service to better standardize and monitor national weather. However, the next major game-changer for understanding and monitoring weather occurred with the development of radar in the 1940s. While radar technologies have changed, this has remained among the key ways in which we still can monitor and use data for forecasting purposes.

Today, there are digital weather stations and many Wireless Fidelity weather stations with wireless internet connectivity that can better consolidate and report information so it is easy to view and understand. Some weather stations can even connect to smartphone apps or online services so people can access their weather information from anywhere. Modern weather observers may use this information to manage their gardens, monitor their farms, or simply as a hobby. Digital weather stations and Wireless Fidelity weather stations tend to be more accurate and easier to use.

1.2 Importance of Weather Monitoring

Initially research or advancements done on weather monitoring were limited, but over the last century it has evolved into a well organised and professional global activity that reflects its crucial importance for a wide range of economic, environmental, civil protection and farming activities. Weather parameters such as air temperature, humidity, rainfall, air quality, air pressure and soil moisture have become important factors in determining the course of a wide range of events occurring around our environment. For example, agriculture has always been heavily dependent on the weather and weather forecasts, both for its control on the quality and quantity of a harvest and its effect on the farmer's ability to work on land or to graze his stock. Water resources generally depend critically not just upon rainfall, but also other weather phenomenon that together drive plant growth, photosynthesis and evaporation.

Weather monitoring is also important not just in defining present climate, but also for detecting changes in climate and providing the data to input into models which enable us to predict future changes in our environment. Because of the wide variety of uses for the information, there are a large number of environmental variables which are of interest to different groups of people. These include solar radiation, wind speed, wind direction, barometric pressure, air temperature, humidity, soil moisture, air quality and net radiation. The demand for these data, usually on an hourly or more frequent timescale, has increasingly been met by the development and widespread deployment of Automatic Weather Stations (AWS).

Weather is the most important element that affects farm production. It can influence crop growth, total yield, pest occurrence, water and fertilizer need, and all farm activities carried out during the growing season. In other words, farming under the open sky is greatly reliant upon the weather and is subject to its moody conditions, especially nowadays, when climate change leads to unpredictable weather which is beyond human control. Being aware of real-time weather conditions like air and dew temperature, precipitation, and humidity is the best way to protect crops and secure a high and healthy yield. Extreme weather such as drought, flood, hail, or frost can cause instant plant stress, thus leading to failed production and increased cost.

1.3 Block Diagram

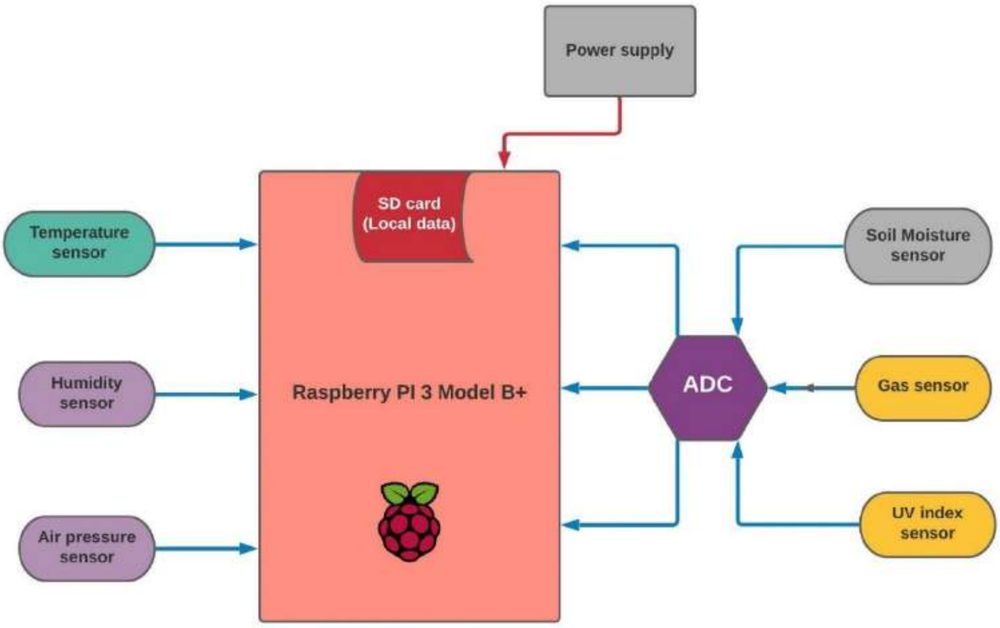


Figure 1.1: Basic Block Diagram of the System

The Micro Weather Monitoring System consists of a central component raspberry pi, a powerful credit-card sized single board computer which controls overall operation of the system as shown in Figure 1.1. The sensors such as temperature sensor, humidity sensor, air quality sensor, soil moisture sensor, air pressure sensor and UV radiation sensor are interfaced with raspberry pi with the help of GPIO pins for independent data-logging and sequential data acquisition of the accurate weather data captured by multiple sensors. Data-logging is done with the help of SD card, installed on a raspberry pi inbuilt SD card slot. The stored data is then pushed on to the database where the required calculations and analysis are done on the data. The analysis done the data is displayed through the user interface (webapp) in real-time. This helps the user an easy and reliable way to understand the pattern of the data collected and make further decisions accordingly.

1.4 Motivation

Today's weather is changing drastically leading to an effected life cycle on earth due to the random change in our environment. Most of the changes include increase in temperature, decrease in oxygen level, increase in level of toxic gases, excessive or scanty rainfall from standard level and decrease in overall air quality. All these changes have led to an unhealthy life and indirectly affecting all living beings on earth. Also, the agricultural practices around the world are largely dependent on local weather conditions. As controlling the weather condition is very difficult, monitoring of local weather conditions can help in taking corrective or remedial actions to minimize loss due to changing conditions.

The effects can be controlled by monitoring weather parameters like temperature, humidity, soil moisture, air quality and air pressure. Micro Weather Station have completely transformed the way weather is monitored today. The invention of MWS drastically simplified the process of documenting and analysing the weather data. Using MWS, weather conditions can be monitored remotely and also analysed for longer periods of time, helping generate accurate weather forecasts.

1.5 Issues of the project

The field applications of weather station are in vast areas, the accuracy of the data is taken as one of the major factors to decide the quality of the system. But if in an application accuracy is needed, then the cost of system increases due to high grade sensors used. Similarly, if application needs to be built at low cost then accuracy has to be compensated. Also, the system should be reliable to the user to access and maintain the data. If there is any problem in the system it must be easy to repair/replace the parts. And if the system encounters any network issues, then there must be some way to retrieve the lost data with ease.

1.6 Objectives of the project

In Micro Weather Station different sensors are interfaced with the micro-computer to get different weather parameters and stored in the server by pushing the data over a network, calculations are made for ease of readability and understand the data for the user by displaying the data graphically in the website which can be accessed from anywhere on the planet. The cost and accuracy are the two main factors that defines the cost quality of the system, and the problems during network dis-connectivity. Hence the objectives of the system are:

- To bring up the accuracy of the data collected.
- To create a feasible system that can handle data from multiple sensors in real-time.
- Usage of multiple micro weather stations and aggregates the data provided by multiple data collection points in real time.

1.7 Tools used

- **Hardware tools**

  - Raspberry Pi 3 Model B+
  - DHT22 (Humidity sensor)
  - DS18B20 (Temperature sensor)
  - MQ135 (Air quality sensor)
  - BMP280 (Air pressure sensor)
  - GUVA S12SD (UV radiation sensor)
  - Capacitive Soil moisture sensor

- **Software tools**

  - Python IDE
  - Expo
  - VS code
  - ThingSpeak cloud platform

1.8 Applications

- Agriculture - Using sensors for weather data collection makes water usage, farming, etc. more accurate using less resources thus reducing labour, time and money.
- Military - During missile, rocket launching the exact environment variables are required.
- Real-time weather forecasting.
- Weather prediction algorithm can be built by using the data collected over time.

LITERATURE SURVEY

CHAPTER 2

LITERATURE SURVEY

2.1 Introduction

The motive of this chapter is to provide a background information on the factors that has to be considered for the proposed system and also it emphasizes to check if the there is any similarities in the work carried out by other authors.

The micro weather station system consists of a control unit which controls the overall operations performed by the system. This control unit can be a micro-controller like Arduino boards, ESP8266, ESP32 and any other application specific boards built for the weather monitoring purpose or it can be a micro-computer like Raspberry pi boards. The usage of these boards depends mainly on the number of sensors used and also on the connectivity to the internet for displaying or plotting the captured data in the website. If a greater number of sensors are used then a control unit with better performance, gives faster response has to be used or if internet connectivity is required then control unit which supports this feature has to be used.

Interfacing of the sensors with the control unit is done with the help of GPIO pins. The two main factors accuracy and cost determines the quality of the system. To obtain accurate data, Sensors having better measuring performance or having lesser tolerance (having lesser deviation from the actual value) has to be selected. Cost of the system will increase if accuracy is given more importance and if the system has to build in a cost-efficient manner then accuracy has to be compensated.

2.2 Literature Survey

Vijayalakshimi and Lakshmi [01] have presented a model to develop simple, low-cost real time remote weather monitoring system with fast and accurate data transfer using the advantages of Raspberry pi and wireless technologies. The system fetches weather conditions continuously using various sensors interfaced with Raspberry pi to measure various weather parameters like temperature, humidity, pressure, gas concentration, light intensity etc. System displays it on Liquid Crystal Display (LCD) for local monitoring, transfers to web page created for remote monitoring and stores in data base for further analysis. The web page also has an option to display location of the system on Google with the use of Global Positioning System (GPS) values transferred by raspberry pi. Real time weather monitoring system function is developed using C++ & Python languages in Qt creator Integrated Development Environment (IDE) on raspberry pi, webpage is developed using Hypertext Markup Language (HTML) and Hypertext Preprocessor (PHP) coding on Web server, accessing of database using Structured Query Language (SQL) queries in side of PHP script. The project is useful for any users who wish to monitor the weather conditions of a location without being physically present.

Uma and Kumar [02] have proposed a Weather Monitoring System. The system continuously monitors the weather status and updated in public cloud. The data can be used in various safety measure applications. It consists of Temperature sensor and CO gas sensor. The model consists of two nodes kept at two different places. The data are collected by the Arduino Uno Microcontroller and it is uploaded to the ThingSpeak sever with the help of Zigbee Transmitter. A Raspberry Pi interfaced with ZigBee transmitter is used as a base station where the data is extracted from the server and displayed on the LCD Display module.

Ladi *et al* [03] have proposed a system that uses ESP8266 as microcontroller and interfaced to different sensors that gives the data, the obtained data is pushed to server (https://www.thingspace.com) where the data is stored. The proposed system uses DHT11 temperature and humidity sensor whose accuracy is less when compared to DHT22. Hence the overall accuracy of the system decreases. The whole system is powered directly from the switch there is no backup power system that ensures the system to work till there is direct power supply.

Patel and Dave [04] presented a design & development of Arduino based Internet of Things (IoT) model for measuring humidity, temperature and CO is measured using MQ135 sensor. Information is received by specially designed application interface running on pc connected through Wireless Fidelity wireless link. The data generated is in excel as well as in graphical form using LabVIEW software for analysis purpose. The system is also capable of generating short time alerts based on weather parameters. It gives an on-line and real-time effect. The idea behind the work is to monitor the weather facility & warn from environment effect.

Shewale and Gaikwad [05] have proposed an idea of real-time monitoring of different environmental parameters using IoT at low-cost. For this purpose, ARM based Raspberry Pi board is used. Raspbian operating system is selected to use with Linux kernel for Raspberry Pi. Python language is used for programming. The system consists of some digital as well as analogue sensors like DHT11 to measure the Temperature and Humidity, BMP180 is used to measure the Pressure and Altitude, Light Dependent Resistor (LDR) used to measure Light intensity and marked scale with ULN2803 are used for environmental parameter measuring. The data are then fed to the ThingSpeak server where the data converted into a graph and also stored in the form of CSV as well as text files.

Kusriyanto and Putra [06] have proposed the system in which test is done by comparing weather parameters obtained from weather station with PCE-THB 40 module and also the measurements and data provided on the website. The weather parameters measured are temperature, humidity and air pressure. With comparison of the measured data with that of results obtained from PCE-THB 40 module gave average error of 3.74% for temperature. 2.14% for air humidity and 0.32% for the air pressure which motivates to increase the accuracy of the system.

Sravani *et al* [07] proposed a method for recovering ecological circumstances like temperature, humidity, CO level etc. at a precise position and create the information accessible anywhere on the world using IoT. The model comprises of numerous modest remote sensors, which are equipped for gathering, putting away, handling natural data, and speaking with neighbouring hubs. Arduino is used as the microcontroller which collects the data from the sensors. A Wireless Fidelity module is used to send these sensors data to the cloud where it'll be stored in the form of a spreadsheet. Using this data different graphs can be plotted for better understanding.

Vestenicky *et al* [08] have proposed a system which uses different sensors interfaced with Arduino MEGA2560 to take the environment parameters and send it to server over a network where the computations are made which are used for different applications. The system is mainly solar powered and uses main line as a backup. Also, the system has a backup memory that ensures there is no loss of data during network issue. The sensors used are all generic which gives the approximate values; hence the accuracy of the system is low. Arduino MEGA2560 heats up when it is interfaced with many sensors as the system if the system is turned ON for long time hence, the system needs to hibernate after the readings are taken.

Girija *et al* [09] proposed an advanced solution for monitoring the weather conditions at a particular place and make the information visible anywhere in the world. The technology behind the system is IoT, which is an advanced and efficient solution for connecting the things to the internet and to connect the entire world of things in a network. The system deals with monitoring and controlling the environmental conditions like temperature, relative humidity and CO level with sensors and sends the information to the web page and then plot the sensor data as graphical statistics. Accurate sensor values are collected and uploaded to the google spread sheet cloud platform. Model can be further expanded to monitor the developing cities and industrial zones. To protect the public health from pollution, the model provides an efficient and low-cost solution for continuous monitoring of environment.

Muck and Homam [10] have designed a project of weather station with real time notifications for climatology monitoring, interface it to a cloud platform and analyse weather parameters. A weather station is assembled using SparkFun weather shield, weather meter and Arduino Uno R3 to collect weather parameters. Data collected from the sensors are stored in google cloud SQL using Raspberry Pi 3 Model B which acts as a gateway between them and analysis of weather data are done. A website and mobile application are developed using google data studio and android studio respectively to display the real-time weather conditions in graphical representation which are accessible by administrator and users. Users will receive notification regarding the weather conditions at a particular place on social media platform regularly and irregularly.

Deekshath *et al* [11] presented a paper which uses an Arduino UNO, Wireless Fidelity module that helps in processing and transferring the sensed data to the ThingSpeak cloud. The parameters received are stored in ThingSpeak cloud platform. The changes in the environment are updated in the form of a database through the cloud computing method. ThingSpeak also provides a feature to create a public based channel to analyse and estimate it through the public. An android application is created for the direct access of the accurate measured parameters. The measured parameters from the sensors are continuously updated that is viewed by the user using the Environmental Monitoring System (EMS) application. Thus, the data is directly accessed, accurate and is purely independent of third parties.

Patil *et al* [12] developed a weather monitoring system for informing farmers about weather changes and provides them with respective guidelines to plan their field. The weather parameters that are being monitored are temperature and humidity. The system uses wireless sensor network for sending information over long distances consuming low power. Low power proves to be an advantage and the system can be easily installed and managed at locations where hardwiring is impossible or there is no access to electricity. IoT technology is used to connect devices and sensors involved using Internet and the collected data and analysis results is directly available to the end user.

Bedge and Purohit [13] have proposed a project which is of low-cost and accurate weather monitoring system supported by IoT platform. System uses Message Queuing Telemetry Transport Protocol (MQTT) data protocol of IoT. The light weight feature of MQTT protocol makes it perfectly suitable for all IoT applications which involves messaging between low power sensors, microcontrollers, Mobile phones and various computing devices. A web application is made to subscribe the broker, hence all the data from sensor node is routed to web application through MQTT broker.

Rahut *et al* [14] have proposed a system which is an advanced solution for weather monitoring that uses IoT to make the real time data easily accessible over a very wide range. The system can record climate changes like temperature, humidity, wind speed, moisture, light intensity, Ultra -Violet radiation and carbon monoxide levels in the air using multiple sensors interfaced to a microcontroller. These sensors are placed at different locations to collect the data to forecast the behaviour of a particular area of interest. The data collected are displayed in the web page with the help of graphical statistics. The main aim the monitoring system is to monitor required parameters remotely using internet and the data gathered from the devices are stored in the cloud and is used to predict trend on the web browser. For predicting more complex weather forecast which cannot be done by using sensors alone, an Application Programming Interface (API) can be used for analysing the data collected by the sensors and predict an accurate outcome. The API can be used to access the data anywhere and at any time with relative ease and can also be used to store data for future use.

Prabhu *et al* [15] have proposed their work in network layer of weather station where the stations are located in remote areas, there is a need of network to transmit the data from the weather station to the cloud storage. System includes Long Range Wide Area Network (LoRaWAN) for transmission of data which uses less power to transmit or receive data over wide range of area and uses double Advanced Encryption Standard (AES) encryption to provide security over the network layer.

Semenov *et al* [16] proposed a system which is capable of sending SMS messages with weather data and a forecast for the nearest future. The proposed system uses Arduino Mega as a micro controller to interface the sensors, GPS to give the location of the plant and Global System for Mobile Communication (GSM) for the network and send SMS. Usage of Zambretti algorithm for the weather prediction gives an accuracy of 90 percent when compared against with the data from meteonovosti.ru website. Since the algorithm averages the data of an hour there is large error induced when Zambretti algorithm is used.

Hastak *et al* [17] have proposed a weather station to provide the information about temperature, humidity, atmospheric pressure, wind flow, etc. There are various sensors present such as humidity and temperature sensor, raindrop sensor, pressure sensor. The main part of the device is the ESP8266 based Wireless Fidelity module Node Micro Controller Unit (NodeMCU). NodeMCU is referred as the heart of the system and provides the platform for IoT. NodeMCU is a Wireless Fidelity module having ESP8266 firmware. All the other sensors are connected to the micro-controller. Sensors send the accurate values measured to the micro-controller and it uploads all the values to the cloud where the values are analysed and the aggregate values will be displayed with the help of webpage.

Dhawale and Bawane [18] has presented the method of processing huge amount of data which is collected by the sensors. The data is then analyzed and stored in a cost-effective way. Environmental conditions variations will affect the overall yield of the crop. Plants require proper very specific conditions for optimal growth and health. Monitoring the condition of crop field is very much necessary so sensors are used. DHT-11 sensor is used to get temperature and humidity values. Soil Moisture sensor is used to get the moisture content of the soil. The NodeMCU is used to get the sensor data. NodeMCU has an inbuilt Wireless Fidelity using this, the data are uploaded to the cloud where it can be used for further computation.

Tsao *et al* [19] have proposed a system which can monitor various environment parameters such as temperature, humidity, wind direction and speed. The system is divided into two, first is called the node station where the different sensors like temperature and humidity sensor, wind speed and wind direction sensor are interfaced with the Arduino MKR1000. Second, the host system where the Raspberry Pi is used as a microcontroller to visualise the data using website built on PHP script-language in the Linux environment. The Node system and the host system is communicated over Wireless Fidelity using MQQT server, hence the node system need to be placed where in it gets Wireless Fidelity connection. The systems accuracy depends on the quality of the sensors used and the reliability of the system depends on the Wireless Fidelity strength.

Kaiyi *et al* [20] have proposed a system which uses effective means for weather monitoring and to indicate sudden changes in the weather. The Doppler weather radar can quantitatively detect the spatial distribution of rainfall echo intensity, radial velocity and velocity spectrum. With its high spatial-temporal resolution, timely and accurate remote sensing detection capability, it has become an important tool for disastrous weather monitoring and early warning. The designed monitoring system consists of a wireless temperature and humidity sensor and a ZigBee wireless gateway. The temperature and humidity sensors are placed in the radome and the temperature, relative humidity parameters collected are processed, and sent to the router. The temperature and humidity information obtained from the sensors are displayed in real time.

Kavin *et al* [21] have proposed a system for monitoring weather conditions like temperature, humidity, CO Level using sensors to detect changes in environment accurately and send it to the users for making statistical analysis. IoT is the technology used for monitoring, collecting, controlling and connecting the system to worldwide, which is the more efficient and advanced solution for accessing the information anywhere in the world. Above system consists of ATmega328 which acts as a central processing unit for the whole system and all the sensors available are connected to it. The data collected are sent to the cloud platform were all the computations on the data are performed and display the accurate data to the users in the form of webpage.

Verma *et al* [22] have implemented a real time weather prediction system that can be used in number of applications like homes, industries, agriculture, stadiums etc. for predicting the weather information. The system utilizes a temperature and humidity sensor DHT11 and a light intensity sensor Light Dependent Resistor (LDR). The sensed data from the sensors are uploaded to a ThingSpeak cloud server using NodeMCU and ESP8266-01 module. The data is also displayed on a customized HTML webpage for monitoring the real time values. A logistic regression model is used for setting up the machine learning environment. The data records are also transferred to the Jupyter notebook that utilizes a python environment. The real time data is used to test the model and prediction is done for a particular value to test the accuracy of the system.

Mullapudi *et al* [23] proposed an advanced solution for weather monitoring which uses AWS IoT to make its real time data easily accessible over a very wide range. The system deals with monitoring weather and climate changes like temperature, humidity, pressure, altitude using sensors and Raspberry pi as a controller. These sensors data will be stored in the AWS Dynamo DB. The project consists of a MQTT protocol that sends messages/mails as an effective alert system to warn people about sudden and drastic weather changes. Due to the compact design and fewer moving parts the design requires less maintenance. The components in the project consumes very less power. Compared to other devices that are available in the market the smart weather monitoring system is cheaper and accurate. The project can be of great use to weather stations, aviation and marine industries.

Jadhav *et al* [24] have proposed a reliable solution in monitoring and controlling the environment monitoring system. DHT11 is used to measure the temperature and humidity. ADIS16220 digital vibration sensor that combines industry-leading sensing technology with signal processing, data captures, and a convenient Serial Peripheral Interface (SPI). LPG Gas Sensor - MQ-6 is used to measure the LPG (composed of mostly propane and butane) concentration in the air. Raspberry-Pi is used to get the data from the sensors and uploaded to the ThingSpeak server with the help of Wireless Fidelity connection. The data stored in the cloud then plotted and can be converted into different file format for better understanding.

Iswanto *et al* [25] in their work presented an IoT based system to measure, read and process the physical quantities of weather conditions. The weather conditions mentioned are temperature and humidity, intensity of light, rainfall and also wind speed and direction. The reading of these quantities is carried out with analog and digital sensors integrated with the ESP8266 microcontroller. Anemometer is used to measure the windspeed, rotary encoder is used to detect the wind direction. Temperature and humidity are measured with the help of DHT11 sensor. An ESP8266 microcontroller is used which acts as a field station connected with all the sensors. The collected data are sent to a server with the help of Modem. The data from the server is then collected by a base station which consists of Raspberry Pi and monitor to display the data.

Guo and Song [26] proposed a system where the system uses GSM to establish connection between the AWS with the Host computer. The system is reliable and stable and anti-interference therefore it can also be placed in areas where it is difficult to place with different terrains like desert, hills, forest, etc. The host computer sends the commands to take input from Automatic Weather station, the AWS reads the data from the sensors interfaced with ATmega64 microcontroller, the command signals are sent using GSM network. If the transmission fails or error in the values of the data, the host resends the command signal to take the readings. The accuracy and cost of the whole system depends on the sensors used; hence selection of sensors decides the quality of the system.

Khotimah *et al* [27] conducted a research to develop a Remote Terminal Unit (RTU) of mini weather monitoring station. In the previous research, a single master-multi slave microcontroller communication method was developed and the microcontroller was able to communicate using unicast communication, i.e., the master gave orders to one slave address via the master-slave network that has star topology. Then the slave which has the same address is requested to respond or take action in accordance with the master command. Design and implementation of RTU is done by using two microcontrollers. One is ATMega8535 microcontroller which acts as a slave and ATMega128L microcontroller which acts as the master. Temperature and wind speed sensors provided accurate results and data acquisition was with the help of microcontroller based RTU for mini weather stations model.

Kodali and Mandal [28] proposed a system which basically senses the temperature, pressure, humidity, light intensity and rain value. The micro-controller used for the system is ESP8266 based Wireless Fidelity module NodeMcu. Four sensors that are connected to the NodeMCU to provide accurate values are temperature and humidity sensor (DHT11), pressure sensor (BMP180), raindrop module, and LDR. Whenever the values from the sensor exceed a chosen threshold limit an SMS, an E-mail and a Tweet post is published alerting the owner of the system to take necessary actions. IoT is used to connect the system through a network and then retrieve the data from the system which can be distributed or upload to any cloud service where one can analyse and process the gathered information.

Saini *et al* [29] have presented their work about AWS where the live environment parameters like temperature, relative humidity, air pressure and wind speed from the sensors of the Arduino based weather station where the sensors are embedded in a compact manner in order to reduce the size of the weather station. The results of the Arduino based weather station is compared with the data from meteorological station, Chandigarh which gave an accuracy with correlation of 0.88 which needs to be improved. And also helps to warn people during disasters.

Sose and Sayyad [30] in their paper have compared the traditional approach of building a Weather station where all the components used are analog in nature which is not flexible and gives more error when compared with the modern approach of building a weather station where in digital components are used which are flexible and gives more accurate results. Due to advancement in technology the cost of the modern weather station is less when compared with traditional weather stations. Also, the wireless systems are much more reliable than wired systems. The problem of complexity of system is solved by moving from the traditional systems to modern systems and the operating range was increased with reduced power requirements from analog to digital system.

Varghese *et al* [31] have proposed their weather station which goals for an economic and purposeful system which can provide a solution for instant weather information and prediction of the weather data. The system comprises of Raspberry Pi 3 as a micro-computer which is interfaced by DHT22 for temperature and humidity and a BMP180 sensor for air pressure. The Raspberry Pi 3 is codded in python language works on kernel of Linux and coding part is done in the Thonny text editor. The data is sent to ThingSpeak server over Wireless Fidelity network for data logging. The data can be downloaded in csv format for the further studies. The weather prediction is done using linear regression model using Machine learning. The weather prediction can be done on a data collected over a period of 2 years. The proposed system was less accurate with an average error of +/-1.2 degree in temperature, mean error of 2% in humidity and average error of 2hPa in pressure. Also, the main disadvantage of the system is that it cannot be implemented in the remote places.

Nikhilesh *et al* [32] have proposed a system where a weather system is installed for a community which can give information to a particular community and warn the people of the community via alerts or emails. The system consists of ESP8266-12E as a node MCU which is interfaced with temperature sensor, LM35, LDR, raindrop sensor and cup anemometer. The data from LDR is used to turn ON/OFF the street lights. The data can be directly viewed by the user with the display provided else can subscribe to the network and can see the data using Blynk application. There is also a desktop version of Blynk for those who want to access the data or by downloading the data in csv format from the server. The issue with the system is that it is restricted to a community of people. And the data is not secure as anyone can subscribe to the system and can make changes in the system. Also, for the accuracy of the system the data collected is not compared with any of the relative system data.

Kapoor and Barbhuiya [33] have proposed their cloud-based weather station which is divided into two sub systems. First is the node station which consists of the node MCU and the sensors interfaced to it and a power supply unit to the whole node. Second is the host MCU this may be microcomputer to the PC or laptops. The node MCU used in the proposed system id Raspberry Pi Zero W board and the host MCU used is Raspberry Pi 3 microcomputer. The host MCU and node MCU are connected wirelessly over Wireless Fidelity network to transfer the data collected by the node from the sensors to the host for further computations. The systems workflow is divided into four states, firstly the data is collected from the sensors to the node MCU, and then the sensor data is pushed to the server with the help of Wireless Fidelity. The server stores the data which can be used for future use. For further computations Amazon Web Services (AWS) Analytics platform is used so that Machine Learning (ML) tools and models can be used for analysis. After the training the model in the cloud warnings and alerts are sent automatically in case of specified alarm conditions. The accuracy of the system can be increased by placing multiple sensors for a single weather parameter to be calculated. The system is portable, reliable, scalable and modularity.

Nandagiri and Mettu [34] explains in their paper about the implementing methods for implementing the weather monitoring system. Classification of sensors is done as primary input quantity, transduction principle, material and technology, property and application. Classification of different types of sensors is done based on property used for measuring different environment parameters such as temperature, pressure, flow, level sensor, proximity and displacement, biosensors, image, gas and chemical, acceleration and others. The most frequently used once are temperature sensor, infrared sensor, ultrasonic sensor, proximity sensor, pressure sensor, level sensor, smoke and gas sensor, touch sensor and humidity and temperature sensor. Demonstration of a general system is done which can give temperature and humidity of a targeted area using Arduino UNO R3 interfaced with DHT11. The accuracy of the system varies with the type of sensors used, usually sensors with high accuracy are costly. Hence the sensors need to be selected based on the application to balance cost and accuracy.

Kanaka and Vidyasagar [35] have designed a system that senses the environment parameters with the help of sensor and Arduino UNO collects the data and sends to the ESP8266 which has inbuilt Wireless Fidelity module, that is connected to internet. The data is then pushed to the cloud storage where the data is analysed and processed. The stored data can also be used for weather prediction. The proposed system uses Arduino UNO as a microcontroller that collects data from LM35 (temperature sensor), MQ-7 (carbon monoxide sensor), sound sensor, LDR. The data collected by the microcontroller is uploaded to the cloud with the help of ESP8266 Wireless Fidelity module. The data collected can be presented with the help of websites, mobile applications or desktop applications, else the data can be downloaded in CSV format. The data collected over time for around year or two can be used for weather prediction.

Chandana and Sekhar [36] have proposed an IoT system which can monitor and control the various sensing parameters such as temperature, relative humidity, light force and CO2 level with the help of sensors that are interfaces to LPC2148 microcontroller. Also, LPC2148 is connected with ESP8266 which server the purpose of Wireless Fidelity module and helps in connecting to the server for the collected data to pushed to the ThingSpeak server where in the data collected is plotted graphically which gives the user an interface to learn and understand variations in the data over time. On-board LEDs shows the used the level of light with the input given by LDR. The relay is used to control the speed of the fan to which the input is given by the temperature sensor, when the temperature in the surrounding increases above the threshold point given the fan automatically turns on, and turns off as the temperature goes below threshold. Since the power requirement is high it must be powered directly, hence it cannot be placed in a remote place. The accuracy of the system depends on the quality of the system.

Adityawarman and Matondang [37] have proposed a reliable system which can provide the user with various environment parameters such as temperature, relative humidity and barometric pressure. The system uses BME280 which has inbuilt temperature sensor, relative humidity sensor and barometric pressure sensor. The sensed data is taken to the ATmega328P microcontroller. To provide connectivity between host and node LoRa Transceiver module is used. The system is powered with the help of rechargeable Li-ion battery. The is sent over the network in Automatic Packet Reporting System (APRS) format, where the sensor data is given by 7 bytes for Wind speed and direction, 4 bytes for gust, 4 bytes for temperature, 4 bytes for rainfall, 3 bytes for humidity and 6 bytes for barometric pressure. The system is complex, the data received from the node station need to be decrypted and it is difficult for the user to understand. The system needs to be more optimised in power, also when comes to the application where the accuracy is needed the system gives large average error.

Govardhan *et al* [38] have proposed a system that can provide various environment data such as Temperature, relative humidity and barometric pressure. The system uses Raspberry Pi Mod B+ as a microcontroller which is interfaced with Sense HAT and a Wireless Fidelity dongle. Sense HAT has several sensors integrated on it used for various applications. The Sense HAT has the following features: gyroscope, accelerometer, magnetometer, barometer, temperature sensor, relative humidity sensor, 8x8 LED display and a small 5 button joystick. The Wireless Fidelity dongle is used to provide Raspberry Pi with internet connectivity. The Sense HAT collects the data from the sensors and gives it to the Raspberry Pi. The Raspberry Pi connected to internet sends the data to ThingSpeak server, where the various data is displayed graphically. The accuracy of the system is low because there are no dedicated sensors, but the cost of the system is low.

Baste and Dighe [39] have proposed a system which is a low-cost weather monitoring station using raspberry pi, which can monitor the weather data like wind speed, wind direction, air temperature, humidity, atmospheric pressure, rain, and solar radiation. Raspberry pi 3 Model b has 40 GPIO pins which is the best option for interfacing of the weather sensors. For communication purposes, on board Wireless Fidelity is used, which can be easily configured with wireless access point. The accurate weather parameters measured by the sensors is continuously stored in micro-SD memory card installed in raspberry pi and then the data is transferred to a database server developed by using MySQL through a

Wireless Fidelity network. The stored data on the server can be accessed through a web application where the user can get real time data.

Majumdar *et al* [40] have proposed a system which is a combination of both home automation and weather monitoring to manage household activities of the user easily with the help of home automation system and also a weather monitoring system which is used to monitor different weather parameters accurately. The system consists of a microcontroller, Arduino Uno and sensors such as temperature-humidity sensors, LDR, atmospheric pressure sensors, LPG gas or smoke sensors and sound sensor to monitor the surroundings. For home automation, DC motor is used to control of doors, windows, LEDs to indicate the operation of heater, AC, home lights and water heater.

Math and Dharwadkar [41] have proposed an IoT based real-time local weather station for Precision Agriculture, that provide farmers a way to automate their agricultural practices like irrigation, fertilization and harvesting at the right time. The proposed system would also help the farmers to carry out the agricultural tasks in real-time basis, which in turn helps farmers to use the agricultural resources in effective and efficient manner. The hardware part consists of microcontroller (ESP32) that has integrated Wireless Fidelity and dual mode bluetooth. The parameters monitored by the system includes humidity, temperature, rain, solar radiation and UV radiations. The data collected is sent to the ThingSpeak cloud server for visualization and Analysis and notifications are sent either in the form of Tweet or email to the users whenever the parameters cross thresholds levels.

Pujari *et al* [42] have proposed a system that monitors weather without human interaction. The system captures parameters such as temperature, wind speed, humidity, light intensity and atmospheric pressure. With these weather parameters, the other parameters such as dew-point, wind chill can be calculated. The system uses ARM-7 controller to control all operations. The data captured from sensors is transmitted to the central GSM node or coordinator node which will send the data to the personal computer through gateway. A server is connected to the database, which has threshold value of all the weather parameters. Power supply for the system is supplied through solar power panel to make the system eco-friendly.

Popa and Iapa [43] have proposed a weather station which consists of temperature, humidity, pressure and luminosity sensors embedded to a microcontroller-based board. The system uses the SEN-08311 USB weather board, which includes SHT15 temperature and humidity sensor, SCP1000 pressure sensor and TEMT6000 luminosity sensor. The software part of the system is written in python language. The weather station is connected to the Internet for displaying accurate weather parameters acquired by the sensors in real-time and the data can be accessed with the help of a PC or a mobile phone with the General Packet Radio service (GPRS).

Savic and Radonjic [44] have proposed a weather monitoring system which collects accurate information about current air temperature, air humidity, atmospheric pressure and relative altitude. The data received from the sensors are processed and transmitted from measuring point to the distant location via User Datagram Protocol (UDP) protocol. Data acquisition from the sensors and the UDP server are implemented with the help of Raspberry Pi platform. The system operates on a local Wireless Fidelity network where the acquired sensor values are transferred to an android application. On the receiving end an android application is used which requests and displays acquired weather data in real time.

Joseph [45] developed a weather monitoring system using various sensors connected to Raspberry Pi. The system monitors weather parameter temperature, humidity, PM 2.5 and PM 10 concentration and Air Quality Index (AQI). The data collected is visualized in graphical means using Raspberry Pi as server accessed over internet. The data is made to store in cloud servers over the internet and is made public in order to access the weather parameter anywhere in the world. The data collected is made available to download in csv and json format.

Ayyappadas *et al* [46] have presented a work that keeps track of temperature, humidity, rainfall, fire detection and earthquake. The system tracks historical information on an hourly basis daily and displays the readings in real time and also stores it in files. The files are then imported to excel automatically and data are cleaned for better representation. Using IoT the data is made available all over the world. The system is reliable and efficient by providing accurate data for agriculture and research purpose.

Ngebani *et al* [47] proposed a solution which provides real time information about environmental conditions to users using IoT. The system uploads the data from the sensor to the cloud platform ThingSpeak. Time stamp and location at which the data is obtained are pinned for each data. The data in graph indicates every environmental parameter fluctuating between a certain maximum and minimum values. The system provides accurate data for the users with relatively low cost.

Feng *et al* [48] have presented an automatic weather station comprising internal detection point classification, status definition, encode design and file structure rules. The station information is compiled into a file to upload the data centre according to the rules which helps to carry out AWS fault diagnosis and analysis. The result shows that the system is successful in status information design and acquisition. Accurate information is provided with continuous adjustment and perfect equipment state decision points and its judging standards.

Kushwaha *et al* [49] have proposed a weather station which operates on IoT technology. The weather bot reports the climate data around the college campus by displaying the sensor data in real time. The sensors are interfaced with e-bulletin and ESP8266 and the data are stored and analysed. The system can also be extended to screen urban zones for contamination observing. The system provides accurate data and less expensive as the sensors and module used are of low power consumption.

VivekBabu *et al* [50] developed a monitoring system which provides accurate information about environmental changes on a more local level. The system monitors temperature, humidity, soil moisture, rainfall and light intensity using various sensors interfaced with Raspberry Pi. The system with comparative analysis is deployed in an agricultural field for effective monitoring of the farm which results in cost reduction, asset saving and productive management in farming.

Omoze *et al* [51] have developed a low-cost server-based automatic weather station for remote locations. The system is built using sensors interfaced with Arduino Mega 2560 and the data are collected either hourly or daily basis. Using notepad, a webpage and a serial data capturing application were developed and an advanced serial data logger software is used for logging data and displaying real time weather conditions. The data obtained is compared with data obtained from campbell weather station for testing its accuracy and the result obtained is in good proportion.

Kadrolli *et al* [52] developed a system used for measuring temperature, humidity and wind over the surroundings and the data collected are reported over the internet. The sensors are interfaced with Eduram and using Wireless Fidelity module the data is continuously transmitted. The data is sent to PC via serial port and testing protocol is used to calculate the parameters measured. The system predicts accurate weather parameters and prediction error is reduced about 10 percent. The design makes the system to be low-cost wireless portable weather station.

Tenzin *et al* [53] have designed and developed a low-cost reliable micro climate weather station in Edamame farm. Cloud based station is efficient in storing temperature, relative humidity and wind speed. The meteorological data gathered from the sensors is compared with the commercial Davis Vantage Pro 2 installed in the farm and the result shows that the weather station is equivalently efficient and accurate. The system design helps the low-income farmers to integrate it in their farming practice.

Yonekura *et al* [54] have developed a short-term local weather forecasting method and data-mining methodologies for dense weather data. Surface weather data is essential for local short-term weather forecast. POTEKA sensors are located in every 2 to 3 km and using mobile network the data is made available in real time. A deep learning architecture is developed for forecasting exactly on the location of the dense weather station and to interpolate the prediction to cover whole range of locations around the interested area. The local prediction is useful in fields like farming, construction and others.

Bharathi and Seshashayee [55] have proposed a method which monitors and identifies the progressions occurring over the current geographical location and gives adequate approaches to the clients to get to the data from anyplace through a cloud storage. The system consists of rain meter connected with Arduino UNO which acts as an ADC for rain meter sensor. Then it is coupled with the Raspberry Pi interfaced with gas sensors such as MQ-6, MQ-7, MQ-135 and also a DHT-11 to get temperature and humidity values. The data is uploaded to the google cloud platform where the data can be accessed from anywhere in the world.

Halder *et al* [56] have proposed a method of finding the temperature and humidity value of remote location in real time. The temperature and humidity are read by EE06-FT1A1 integrated sensor module. The integrated sensor module is interfaced with the microcontroller development board to get the required data. The data is read by the ATmega2560 Board which acts as an ADC. Using an USB dongle, the data is uploaded to the server for mobile access. Also using an RJ-45 cable the data saved in an offline base station.

Haque *et al* [57] have proposed a weather estimating system mainly focused on agribusiness. The system is a Renewable Energy Source (RES) powered weather station which monitors the weather parameters. The system contains sensors for detecting temperature, humidity, raindrop, carbon monoxide, smoke, LPG in the environment, barometric pressure, altitude etc. The information from the sensors is gathered by the Arduino. Arduino sends the sensors information in LCD display. Additionally, the device sends an SMS which contains weather information to the user with the assistance of a GSM module.

Rao *et al* [58] have proposed an advanced solution n for monitoring the weather conditions at a particular place and make the information visible anywhere in the world. The system deals with monitoring and controlling the environmental conditions like temperature, relative humidity, light intensity and CO level with sensors and sends the information to the web page and then plot the sensor data as graphical statistics. The Arduino UNO interfaced with the DHT11, LDR sensor, sound sensor, CO gas sensor receives the data and uploads to the google cloud platform server with the help of ESP8266 where the sensor data is represented in terms of graphical statistics.

Kuwari *et al* [59] have proposed smart-home monitoring and automation system that is based on the IoT technology. The selected platform is the EmonCMS that uses a cloud server to collect data from sensor nodes using the IoT principle. LM35 temperature sensor, HIH4000 humidity sensor, LDR are used to find the weather parameters. The NodeMCU combined with the ESP2866 was used as the main processing unit that collects the data from the sensors, processes it and then uploads it to the EmonCMS cloud server. The NodeMCU can also read data and commands from the same server and control switching devices.

Shaout *et al* [60] have proposed a system to measure weather parameters like wind speed, wind directions and temperature. Freescale Dragonl2-Plus2 board from HCS12 microcontroller family with MC9S12DG256 is used as the microcontroller which acts as ADC. The data received by the microcontroller is sent to the serial port that can be read by a personal computer. The outputs of the system are displayed on the LCD. The data obtained from the intelligent weather station will be stored on a personal computer and according to the history more useful information can be predicted.

Lee and Kim [61] have proposed a high-resolution automatic weather system. An algorithm was proposed for determining real-time road surface conditions using ensemble learning. Learning data was organized using time-series mapping of observed data from the AWS. The road condition determination model uses machine learning. Road conditions are measured using image data collected from weather observation devices installed on the roadsides and surface sensors on the road surfaces. Using Support Vector Machine (SVM), Multi-Layer Perceptron (MLP) machine learning algorithm the real time road conditions are measured by applying the algorithms on the data collected by the automatic weather system AWS.

Li *et al* [62] have proposed a method to measure the structure and operation quality of automatic weather station, according to the theory of fuzzy Analytical Hierarchy Process (AHP), a quality assessment system for automatic weather station operation. Combining with the occurrence of operational failure in recent years, the fuzzy AHP is used to assess the automatic weather station and a new method to assess operating quality of the automatic weather station is proposed. With the advance of construction of automatic weather stations, the single manual ground meteorological observation gradually turns into manual and automatic operation or single automatic way. Automatic observation will eventually replace the manual observation. It becomes a serious problem to set up a scientific, objective, quantitative, and impartial assessment system about operating quality of automatic weather station and determine the factors that impact the operating quality of the automatic weather station.

Li *et al* [63] have proposed a concept of micro-automatic weather station was proposed to meet the needs of modern power grid analysis and control. Based on the STM32 controller and meteorological sensors, the hardware and server software of the micro-automatic weather station are designed and developed, which can monitor wind speed, wind direction, light intensity, temperature and humidity of surroundings atmosphere. All real-time meteorological data together with geographic information offered by GPS are sent to the base station via wireless GPRS network, the networked data are analysed in the server. The real experiment showed that weather station can run smoothly in the natural environment, providing accurate and real-time meteorological data for users. Consequently, micro-automatic weather station makes it possible for researchers to get a large amount of meteorological information at a low price in a short term.

Mittal *et al* [64] have developed an online smart weather station system for studying the correlation amongst multiple weather parameters data, collected over a period of 18 months. The system consists of sensors that generate weather data, local storage, wireless transfer to control centre, and web-based online representation and analysis of data. Six weather parameters such as air temperature, relative humidity, wind-speed, rain, rain-rate and solar energy are studied. Statistical measures such as mean, median, standard deviation, min, max and normalized standard deviation are computed from the weather data. The correlation among temperature, humidity and wind-speed parameters is presented. Few important observations are made, for possible applications in agriculture, construction and manufacturing activities. Further a web-based service can be aimed at automated analysis of weather data generated by the smart weather station.

Miu and Djokic [65] have proposed a method for evaluation and prediction of wind energy resources at unmonitored sites. For that purpose, a simple linear regression model is developed, based on spatio-temporal correlation of recordings at monitored sites, requiring only the coordinates of a target location of unmonitored site. The presented results demonstrate that the proposed evaluation model works well at sites where there are strong regional-scale wind regimes, simulating wind speed time series and annual average wind speeds with an accuracy of 90 percent, or higher. However, the performance of the model deteriorates significantly when applied to sites with highly individual wind regimes.

Sattyanarayana *et al* [66] have proposed a system to measure the weather parameters in an effective way at the place of interest. This paper projects an easy way to measure the dynamic parameters of weather without human interpretation. The proposed method employs mobile app and IoT technology, collected weather parameters in a remote area can be uploaded to cloud as well as particular mobile app. The uploaded data can be verified and used, at anytime and anywhere in the world. The proposed system uses Raspberry-pi embedded with weather sensors to collect weather conditions.

Munandar *et al* [67] proposed the design of real-time weather monitoring system based on a mobile application using AWS. The system connects to the AWS equipped with several sensors for collecting data and storing the data to the web server. Data from weather sensor is taken from the AWS-Davis Instrument using the WeatherLink software. The data is transmitted through the data logger using serial communication, uploaded via File Transfer Protocol (FTP) and stored on a webserver. The Android application reads the files and displays the information provided by the web server in real-time. The system has successfully show real-time monitoring of weather through the mobile application with a flexibility in the parameters and the need of User Interface (UI) design compared to the other solution.

Handani *et al* [68] proposed an application that can provide information about weather conditions at a particular location in real time. The application is still cannot provide direct weather information from the desired location accurately and real-time. The system is designed using DHT11 sensors, rain sensors, LDR sensors, BMP 280 sensors and IP Cameras. The real-time's concept used is Soft Real-Time System (SRTS). Based on the testing process that performed at different times, a temperature variation of 220 °C – 380 °C, humidity 10 percent to 70 percent RH is generated and the response time of the system is in range 151ms – 157ms. This data then compared with the data from several weathers forecast application in some media. Data is sent and stored to the data base and then displayed in web application also mobile application. Based on that information, the user can get information directly in real time conditions due to weather conditions at the destination location by click the camera button.

Nsabagwa and Byamukama [69] have proposed a method where the weather stations distributed across the country periodically collect weather data and store it in the data logger, which is usually located at or near the weather station. After a period between one to three hours, the data logger sends the data to an observation station manually or automatically using either GSM/GPRS, Wireless Fidelity or ethernet connections for internet access. At the observation station, the data is aggregated from all the weather stations in the country to be fed into forecast models. The AWSs transmit their data to repositories. The repository may contain legacy data which is imported from different weather stations. Once at the repository, modelling using tools like Consortium for Small-scale Modelling (COSMO) and Weather Research and Forecasting (WRF) model is performed on the data for prediction purposes.

Liu *et al* [70] have proposed a framework based on Wireless Sensor Network (WSN). There are two layers called device layer and application layer respectively in the new intelligent framework. Analog signal acquisition, meteorological factor calculation and formatted output are assembled at the device frontend (or layer). The application layer is formed by at least one embedded system, which can be used for real time data collection, data quality control, extreme value statistic, information dispatch and other complex functions as well. Comprehensive weather station named CAWSmart under the new framework. The CAWSmart has recognition intelligence, networking intelligence and diagnosis intelligence. With the metrological system, the CAWSmart can archive calibration intelligence. This novel structure has clearer layers, and better extensibility. Each observed factor will be more independent, flexible and anti-destroyed. The device layer focus on observing stability, meanwhile the application layer focus on functional extension. The observing system will be smarter with the increment of different embedded software.

Weerasinghe *et al* [71] developed a system consists of three separate modules for data collection, data storage and data communication. The modules communicate with each other serially and are controlled by three separate PIC18F452 microcontrollers. The data collection module is interfaced to a set of sensors to collect weather parameters. The data storage module saves the captured data in real-time to a micro-SD card and transmits data to a central station through a GSM network. The weather data can be viewed in real-time through a Graphical User Interface (GUI) located at central station.

Eleftheriou *et al* [72] proposed a system which predicts the wind speed on short-term weather conditions for an on-board vessel weather station. Several machine learning models were developed for different forecasting horizons and their efficiency for this study was presented across a number of metrics. A regression machine learning algorithm was chosen for sea trials on Lincoln vessels, due to the practical limitations of the application. The ultimate goal of a learning model is to minimize this true error and attain good generalization performance.

Ram *et al* [73] proposed an advanced solution for monitoring the weather conditions at a particular place and make the information visible anywhere in the world. The system deals with monitoring and controlling the environmental conditions like temperature, relative humidity, light intensity and CO2 level with sensors and sends the information to the web page and then plot the sensor data as graphical statistics. It is also a less expensive solution due to usage of low power wireless sensors and SoC contained Wireless Fidelity module.

Iliev *et al* [74] have presented realization, calibration and initial experimental data analysis of the prototype version of adaptive weather station. The device relays on an ultra-low power micro architecture and adaptive power distribution mechanism. It is dynamically reconfigurable for working in high performance, real time transfer mode with direct operator control and ultra-low power, fully autonomous, self-monitoring, long-term measurement mode. For convenience the collected data of the environmental parameters is initially analysed and visualized by specialized end-user software tools.

Kulkarni *et al* [75] developed an IoT enabled weather systems to collect data from vehicles on the road. Vehicles moving on the road will wirelessly communicate the weather data that is inclusive of air temperature, visibility or light and other data needed. This data helps to build more accurate forecast and monitoring at different time horizon. The sensors with IoT technology contribute to collect weather data. IoT technology is beneficial to transport system and farmers. Remote sensing technology used for examining the weather forecasting helps to gather and analyse weather data and use to build the database for weather forecasting.

Khotimah *et al* [76] have proposed the approach in designing database for weather and tide monitoring data center. The data center is developed from the existing report system universal data logger (UniDalog). The data is scattered in several operational databases. Contributions made in this paper is a method of tracking the history of changes to the attribute table column names in the data storage station. The approach is to adopt the universal table concept in multitenant database table schema in the table data storage channel sensor and slow changing dimension for storage usage history channel.

Kruger *et al* [77] presented some of the techniques often used to impose fault tolerance and describe a system developed for fault injection, which operates by inserting faults in certain memory regions to change the data and to cause crashes in a rapid prototyping platform for microcontrollers. Rapid prototyping platforms with microcontrollers from different manufacturers normally contains libraries that provide functions and classes already implemented. Due to the rapidity with which these platforms are placed on the market, the probability of programming errors is greater and the use of techniques of fault tolerance becomes elementary.

Kunjumon *et al* [78] presented different data mining techniques and methodology for weather prediction such as supervised and unsupervised machine learning algorithms, artificial neural network, SVM, frequent pattern growth algorithm, hadoop with map reduces, K-medoids algorithm, Naive Bayes algorithm and decision tree classification algorithm. Based on result, it is concluded that the support vector machine algorithm can gives better weather prediction with higher than 90 percent accuracy and provides better result.

Lage *et al* [79] developed a weather system for crop protection in rural areas. The system consists of mini weather stations that are commanded by a central station which processes the data collected and based on these, it generates alarms, either by falling frost or the appearance of Zonda winds. The system, in addition to the occurrence of Zonda winds can predict with several hours in advance of possible frost formation, giving the farmer a very useful tool when having to deal with such events.

Fang *et al* [80] have proposed a micro weather station, which can sense temperature, relative humidity, pressure and anemometer, and is portable in small size and possesses high precisions. The system consists of multi-sensor chip, anemometer, measurement system, display system and power management system. The multi-sensor chip is integrated with sensors such as temperature, relative humidity and pressure. A drag force wind sensor using the torque of cantilever to measure the velocity of wind is developed. The wind direction is measured by perpendicularly encapsulating the two-wind sensor.

Laskar *et al* [81] have proposed a basis system that can give the basic environment parameters such as temperature, moisture, altitude and pressure. The system used Arduino UNO as a micro controller, DHT11 is used as humidity and temperature sensor, BMP085 is used as pressure sensor, ADXL 335 is used for altitude measurements and RF transmitter and receiver is used for transmission and reception of data. The sensors sense the data and sends it to Arduino UNO to which the sensors are interfaced. The RF transmitter is interfaced with the Arduino UNO and the receiver is connected to the system that collects the transmitted data. Since the system uses DHT11, BMP085 and ADXL 335, the accuracy of the system is low. The system is portable and cost efficient. Also, the system does not use internet the system can be placed even in remote areas where it is difficult to get internet, but the limitations is that it cannot be used for long distance transmission.

Raju and Laxmi [82] have proposed a system that can predict weather using ML using matlab code. Various ML techniques were applied and tested for the accuracy of various weather parameters such as temperature. The system uses DHT11 sensor to sense surrounding temperature and PZEM004T is used to measure the voltage and current. The Arduino UNO is used to interface with DHT11 and PZEM004T, ESP8266 is used as an external Wireless Fidelity unit that connect to cloud wirelessly by interfacing it with Arduino. The data is packetized and sent to the cloud storage where different weather prediction algorithms are applied to the data over time, machine learning algorithms such as linear regression algorithm, SVM algorithm, ensemble algorithm, fine tree algorithm and gaussian process regression algorithm. The results were sent to ThingSpeak cloud which helps to plot the data over time. Out of various prediction algorithms with multiple variables and gaussian process regression algorithms were evident and more accurate for weather prediction after a training period of 24hrs.

Manikumar and Singla [83] have proposed a system which can give various environment parameters such as temperature, dampness, wind speed and heading, precipitation sum etc. The information can be displayed on the monitor as well as sent to the webpage and after the plot of the sensor data as a graphical representation. This data can be accessed from anywhere and at any time on around the globe. The system uses LPC2138 microcontroller as main processing unit for entire system and all the sensors and devices that are connected to it. SIM800L is used as GPS/GPRS module and supports the GPRS coding module. Initially the environment parameters are measured by the sensors for the duration of time set by the programmer, the threshold valued are checked for the alarm conditions. Then the data is sent to the server where the decisions are based on the data and the results are displayed graphically in the website. If any alarm conditions occur a message is sent to the users.

Zahoor [84] has designed a weather station that is specific to a particular zone. This system can be mainly used in the cities where there is need of the weather information of a specific area or area wise data is required. Now-a-days there is a drastic change in the climatic conditions of cities hence need of the need of zone-specific weather station. The system uses

Raspberry Pi as a microcontroller that is used to interface various sensors used to sense the weather parameters, DHT22 to measure temperature and humidity, BMP180 to measure pressure and altitude, MQ3 to measure alcohol, ethanol and smoke, MQ7 to measure carbon monoxide and MQ3208 analog to digital converter. Weather forecasting techniques require a huge amount of previous data, usually 30-40 years, to accurately predict the future weather conditions. This huge amount of data set is then used in neural networks for training and after training it can predict the future values within the margin of error.

Roa *et al* [85] have proposed a system that can collect various weather data such as temperature and humidity, wind speed, wind direction and gust, rain, air quality. They have used as ESP8266 microcontroller enabled with Wireless Fidelity at the node. Temperature and humidity data were gathered using an AM2315 I2C capacitive integrated humidity and temperature sensor, wind speed and direction were measured using a cupped wind anemometer and arrow-type wind vane, barometric pressure was measured using a BMP280 I2C piezo-resistive pressure sensor, power is provided to the weather board and all its sensors by four 6V solar panels which were sized to provide a maximum current of 1320 milliamps (mA) to provide enough power to the entire data collection system, Raspberry Pi Model B was used to provide a means of remote monitoring, control, and data management of the embedded data collection system. The results had an average error of 5 percent that is the system had an accuracy of 95 percent.

Chandrathilake *et al* [86] have proposed a system that can reduce the executional and computational time required to run the algorithm. The data collection mechanism done in static and fixed time durations, the problem with this approach was that weather forecast or prediction algorithm have to process large volumes of data from weather stations of various places which are not relevant. Hence the time complexity can be reduced by computing the data of area of interest, by this the execution time decreases and the algorithm is applied only on the data required. Therefore, it is very fast to predict the weather data of the area of the required region. Initially the thunderstorm indices are selected for a particular data then the complex event processing component ML algorithms are applied for weather prediction then the weather research and forecasting are done on the results. By this approach the time for the computation has been reduced by 75 percent than the normal computations.

Bumbary [87] has developed an inexpensive infrastructure of weather stations, to create more accurate forecasting networks for citizens and governmental officials of third world and developing countries. The research compares the precision of inexpensive weather stations to commercial grade weather nodes, and data gathered from the National Oceanic and Atmospheric Administration (NOAA). The research also assesses the forecasting accuracy of the inexpensive forecasting technology systems, against popular weather media such as the weather channel and the national weather service. The ultimate goal is to install the inexpensive weather infrastructure in developing countries, in order to help government officials accurately forecast and prepare for critical weather events.

Yamanouchi *et al* [88] have proposed a real time weather monitoring system which helps in reducing the damage caused by the disaster. The collected data is not only used for disaster but also agriculture, educational purposes, research and so on. The AWSs alarms when there is high temperature, dust and heavy rain. AWSs are connected to internet for sending data to the server. Two servers are used for duplication. One server installed in India Meteorological Department (IMD) and another server is in Indian Institute of Technology, Hyderabad (IITH). IITH server is synchronizing to IMD server because of IITH has 2 hour rolling blackouts every day.

Voiculescu *et al* [89] have proposed a solution for making the autonomous weather monitoring system more efficient requires optimizing consumption for the computing and storage facilities are they require more energy to function than the sensors. Effects of frequency in power consumption on a modern Reduced Instruction Set Computer (RISC) based system implementation, Texas Instruments OMAP3 family of devices. In order to study the impact of frequency scaling alone without the additional effects brought to by voltage scaling or methods of idling the silicon, a modern microkernel is used as the basis of a software operating system.

Ruan *et al* [90] have developed a system to access real-time weather data in regional mesoscale weather monitoring using a data browser system based on various WebGis components using a distributed system, such a data browser system can decompose the entire data processing task into several subsystems, but the database can integrate and manage each subsystem, create a product pre-processor, and carry out standardization for data dissemination. A B/S type of software is designed and developed, which allows to analyse the data, rapidly disseminate the geographic informational data, as well as interactively plot and statistically analyse the data. This system provides an application platform under internet and intranet environment.

Xiangjuan *et al* [91] have proposed a method of utilizing active resistance network to calibrate the temperature channel of data-acquisition unit. This method changes gate-source voltage by micro-controller so as to simulate the resistance. Compares resistance value measured by data-acquisition unit with the simulated resistances to calibrate the performance of the data-acquisition unit. By software commands, output current and sample duration time of the data-acquisition unit temperature channel are changed to resolve the problem of parameters mismatch, then this data-acquisition unit is calibrated effectively. The experiment result shows this approach provide a reliable means to calibrate AWS data-acquisition unit.

Chebbi *et al* [92] have designed system that can give alert to the user under the Tunisian conditions. The field micro-climate monitoring is important for crop water requirements determination. Traditional weather stations are expensive, on demand data collection and not flexible for new sensors addition or WSN integration. This paper aims at the design of a custom low-cost weather station hardware and software node (WS-node) for irrigation scheduling in developing countries context. These sensors data are transmitted to the decision support system in a WSN infrastructure, taking into account all particularities of such environments constraints. Cost factors, field conditions, farmer knowledge, devices availability are the most important ones. Minimizing the cost of deployment is of paramount importance. Since WSN is a nascent technology, many of the existing general purposes solutions in the market are expensive and/or they are not well tailored for use in developing countries.

Chen *et al* [93] have designed a system that used mobile ambient resources for the weather forecasting. The goal of this study is to design and implement an augmented reality service with context aware weather information prediction on a mobile device. The predicted weather information would be calculated based on the past weather information and the user's position located by the GPS senor in the mobile device. After the prediction calculation is finished, the user can easily retrieve the result via any ubiquitous networks, such as 3G or Wireless Fidelity networks. Moreover, to make the user more easily comprehend the result, the weather forecast information can be rendered in an augmented reality scene on mobile devices by meshing up the caught image from the build-in camera and the final predicted weather information. We called it the Mesoscale Weather Forecasting System (MWFS).

Karvelis *et al* [94] have designed a system that can do weather prediction on-board in real-time. The system is composed of a commercial weather station integrated with an industrial IoT-edge data processing module that computes the wind direction and speed forecasts without the need of an Internet connection. A regression machine learning algorithm was chosen so as to require the smallest amount of resources (memory, CPU) and be able to run in a microcontroller. The algorithm has been designed and coded following specific conditions and specifications. The system has been tested on real weather data gathered from static weather stations and on-board during a test trip. The efficiency of the system has been proven through various error metrics. In particular, linear regression models can be considered as an evolution of the Moving Average (MA) algorithm that allow a different weight for each parameter instead of the equal weight of each parameter in the case of the MA method. The efficiency of the system has been proven by testing the system in real sea conditions where weather parameters have been recorded. The algorithm performances have also been evaluated using other datasets, acquired from land weather stations located in different geographical and micro-climatic areas.

Baht *et al* [95] have designed a weather forecast and green building on micro grid energy management system. The specialized software was employed in order to study the effects of climate and other external factors on energy consumptions and the related costs. Two sites, highly different with respect to climate conditions, were addressed firstly (Bagdad from Iraq and Craiova from Romania) in order to evaluate the volume of CO2 emissions and respectively the effectiveness of power generation by using solar and wind energies. It is notable that the issue of technical competences with the systems serves as one of the main challenges that can undermine the effectiveness with which micro grid systems are applied into the framework of renewable energy resources. While some factors are influential in the short-term, others apply in the long-term. Therefore, an effective evaluation of the distinctive ways in which power and energy management can be influenced by green building is required. The control of the energy management system in micro grids can be made in a centralized or decentralized manner. The communication and control of data flows through the LC in case of the centralized control, while all the information is sent to the decentralized control network.

Vishwarupe *et al* [96] have proposed a zone-specific weather monitoring system, where the main focus is to make use of the telecom infrastructure which normally lies dormant, apart from its primary use. Telecom towers are enacted by the service providers and once installed; they have a lifespan of around 20-25 years with less maintenance. Each tower has for itself unique geographical coordinates to identify its location, a power supply, grid-map (on the server side) and a database on the administrator end to keep a check of its operations. All these aspects can be used to monitor zone-specific weather. The installation of sensor nodes consisting of sensors like temperature, humidity, ambient light, rainfall and gas, which detect the atmospheric elements atop the towers. By resorting to this strategy, the sensors are safe from human intervention thereby increasing their lifetime in the long run.

Pang *et al* [97] have fabricated to integrate pressure sensor into multi-sensor for micro weather station. Differing from traditional silicon piezo resistive or capacitive pressure sensor, we use platinum piezo resistive pressure sensor in the integration, which can greatly simplify the whole process and also has an excellent performance. We also use adhesive bonding with SU-8 to replace the traditional bonding methods. This bonding method outweighs the traditional bonding methods in many ways. The testing results of the pressure sensor indicate that this simple and convenient fabrication technology is advantageous for the integration. Bonding is the most crucial process of the whole fabrication. Through investigating different parameters, we found that pre-bake time and vacuum pumping time are the most key factors while the bonding pressure, temperature are also quite important. The fabrication results show that the adhesive bonding with SU-8 in vacuum is very successful. The testing results show that the performances of the pressure sensor are completely promising, especially in the aspects of precision and stability.

Ramani *et al* [98] have proposed a low cost, reliable and deployable solar power weather monitor system for all climate condition is build up. The online monitoring system using sensors gather weather condition and transmit to the system through bluetooth. The bluetooth system can be readily accessible to follow real-time weather changes and monitoring the device by using a DC supply or battery. The Arduino microcontroller board is helping to control the temperature, humidity values and also used to monitor the voltage and current values of the solar panel. The solar is continuously providing the voltage to the consumer with different climate conditions. The different climate changes values are stored by bluetooth device. The climate changes are stored in Bluetooth devices, that values are modified by simulation using matlab software. The solar voltage level is monitored and displayed in matlab scope. This scope result shows that the level of the solar voltage and another one scope is displayed in climate changes levels. Arduino is being used to automatically collects and transmit the measured data. The proposed project needs a solar power for that purpose installed 250 watts and set for the continuous analysis. And these criteria such as irradiation, temperature level, humidity level, light intensity level, current and voltage may vary for every period hence there would be a source for continuous analysis.

Kurniawan *et al* [99] developed a weather prediction system using fuzzy logic algorithm for supporting General Farming Automation. The weather calculation system works by taking a weather prediction data from the Weather Service Provider (WSP). Also, it retrieves soil moisture sensor value and rainfall sensor value. Finally, the system will calculate using fuzzy logic algorithm whether the plant should be watered or not. The weather calculation system will help the performance of the General Farming Automation Control System in order to work automatically. So, the plants still obtain water and nutrients intake are not excessive.

Byamukama *et al* [100] gives guidelines for ultra-low power gateways in environment monitoring WSN. The techniques that can be used to reduce the power consumption of gateways in WSN deployed in environment monitoring applications, such as AWS. The challenge is the deployment of these networks in locations that are far from a consistent power source, such as a national grid. Such stations must be autonomous and power consumption must be minimized. Models for Internet connectivity to individual sensors in a WSN is standardized by IETF. In favour of robustness and simplicity, we have rather chosen to focus on a low complexity WSN model in which all sensor nodes are broadcasting their data to the gateway via an always-listening sink node, directly in one hop or via a relaying node just repeating the broadcast. The way forward in this regard to explore the impact of radio duty-cycling is to implement the ContikiMAC RDC protocol for the ATmega256RFR2, which has already been proven to be attain very low power consumption with the hardware interventions alone.

2.3 Summary

From the above Literature survey, the major attributes used for the contributions of significant studies and survey of existing works, the following major points are considered for Micro Weather Station.

- The accuracy of the system mainly depends on the quality of the sensors which will be costly.
- The Application Specific Integrated Circuit (ASIC) boards with the inbuilt sensors gives a large error due to its compact design.
- There are various weather prediction models such as Logistic Regression (LR) algorithm, SVM algorithm, ensemble algorithm, fine tree algorithm and Gaussian Process Regression (GPR) algorithm. The best of all these is LR with multiple variables and GPR.
- In case of multiple nodes, to reduce the execution and computation time only data of area of interest is taken in consideration.

Since weather station can be used in various fields, the requirements of the system depend on the field of application. In order to overcome and accomplish the required results the following points are taken into consideration.

- Based on the field of application the type and quality of sensors are selected.
- For weather station near locality a direct power supply can be given with a battery backup.
- For weather station in remote areas solar power supply with a battery backup is ideal.
- For weather station near locality Wireless Fidelity would be the best option.
- For weather station in remote areas GSM/GPRS would be the best option.

PROPOSED SYSTEM

CHAPTER 3

PROPOSED SYSTEM

3.1 Introduction

The system mainly concentrates on providing the user with precise weather parameters which is affordable to common man. In this system Raspberry Pi 3 Model B+, a powerful micro-computer is interfaced with different sensors to get accurate weather parameters. The current means of weather monitoring are indeed rather limited and make use of very expensive weather stations, leading to lack of comprehensively monitoring due to cost constraints and inconveniences. But the proposed system is small in size and possesses high precision as similar to that of high-end weather station with significantly less cost. This is achieved by using multiple micro weather station with low-cost sensors which produces accurate data with lesser power consumption.

The design of using multiple nodes with interfacing low-cost sensors can significantly bring down the cost since rather than using a single large expensive weather station, multiple nodes can be used which are interconnected to each other to achieve the same performance. In this system, each node with multiple sensors interfaced provides the full fetched functionality of collecting weather parameters similar to that of a high-end weather station. The data collected independently from different nodes are aggregated in the webserver and finally displayed through the user interface (webapp) in real-time which helps the user in understanding the pattern of the data collected and make further decisions accordingly.

3.2 Advantages

- Provides Real-Time weather parameters.
- Study climate changes and weather patterns on the weather parameters.
- Reduction in cost using low-cost sensors.
- Accurate data is measured as the data collected from multiple nodes are aggregated.
- The Accurate data can be used for research purposes.

❖ Weather forecasting can be achieved by applying certain algorithms on the weather data.

3.3 Workflow of the system

The overall working of the system is controlled by Raspberry Pi, which takes care of all the data transfer activities. The power supply is given to Raspberry Pi which in turn gives power supply to all the sensors.

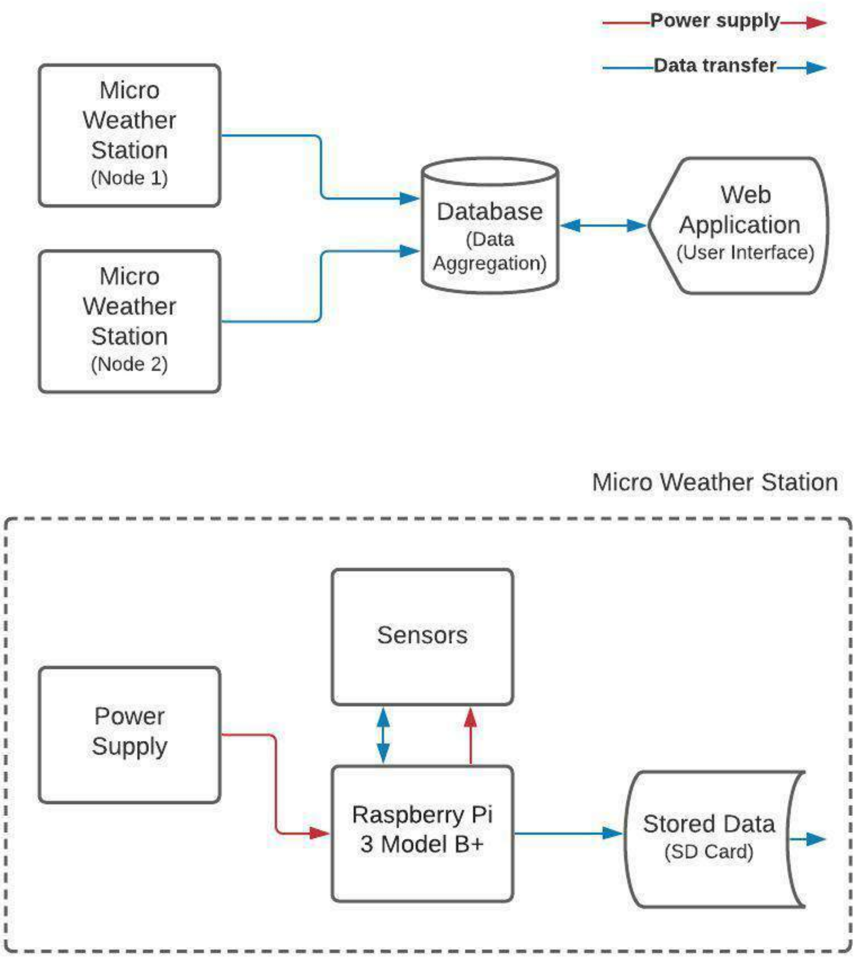


Figure 3.1 Workflow of the system

3.4 Methodology

Step 1: Initial Setup

The system consists of a central component raspberry pi, a powerful micro-computer which controls all the operation of the system as shown in Figure 3.1. The micro-computer can be powered by 2.5A micro-USB power supply for configuration when the implementation of the system is nearer to a supply source. The data is uploaded into the webserver using the internet connection through Wi-Fi.

Step 2: Sensor Interfacing

The system comprises of multiple sensors such as temperature sensor, humidity sensor, capacitive soil moisture sensor, air quality sensor, ultraviolet (UV) detection sensor and air pressure sensor are interfaced with the raspberry pi with the help of GPIO pins. In order to read the analog values from the capacitive soil moisture sensor, UV detection sensor and the air quality sensor an Analog to Digital Convertor (ADC) module is used. The sensors are configured and calibrated using python IDE tool. All the sensors are powered individually from the PCB board which provides the supply voltage parallelly.

Step 3: Data Logging

Collecting and storing the data over a period of time in order to analyse specific trends or to record the data based on events is very important in this system. The accurate weather data which is time tagged collected from multiple sensors are independently data logged.

The sensor data is primarily stored in the SD card by partitioning the unused memory from the Raspbian OS. The data can be stored in the form of Comma-Separated Values (CSV) file or JavaScript Object Notation (JSON) file depending upon the requirement using the python programming. The log files are generally found in /var/log folder and most of the files in there can only be read by the root user.

Step 4: Database

A web database can be accessed from a local network or the internet instead of one that has its data stored on a desktop or its attached storage. The data logged from each node is further pushed to the webserver with the help of internet connection through Wi-Fi and also the same is stored in the SD card as a backup. Over a period of time the data is updated to the server in real-time. The real-time data collected from multiple nodes are aggregated by performing some mathematical operations to ensure accuracy.

Step 5: Web App

The analysing and visualizing of the data collected from the sensors is important which makes it easier for the user to understand the data. To build a web app, expo framework is used which is a set of tools and services built around React Native and native platforms that helps to develop, build, deploy, and quickly iterate on iOS, Android, and web apps from the same JavaScript/TypeScript codebase. React Native is an open-source framework with one of the largest support communities. Unlike typical hybrid applications, this new framework targets mobile platforms. It's based on the JavaScript library, React Native allows you to create a single JavaScript codebase that will work on different mobile devices (iOS, Android & Windows).

The user interface is the part of the web application which a user interacts with. In simple terms, it's everything a user can see and touch, such as buttons, colours, fonts, navigation, etc. The user interface is used to display real-time data from the sensors and also the data is visualized by plotting the graph. The data stored in the webserver can be accessed/imported in the web app, for applications like weather predictions, research etc.

3.5 Summary

The chapter discusses the methodology of the proposed system and it briefs about the hardware and software control flow. It also provides some information about the data acquisition process from multiple nodes. The workflow of the system is explained step by step in this section.

HARDWARE AND SOFTWARE REQUIREMENTS

CHAPTER 4

HARDWARE AND SOFTWARE REQUIREMENTS

4.1 Introduction

This chapter gives brief idea about various hardware and software tools/components used in Design and Implementation of Micro Weather Station. The Raspberry Pi 3 Model B+ is user as the micro-computer. The explanation to various parts of Micro Weather Station is given below.

4.2 Block diagram

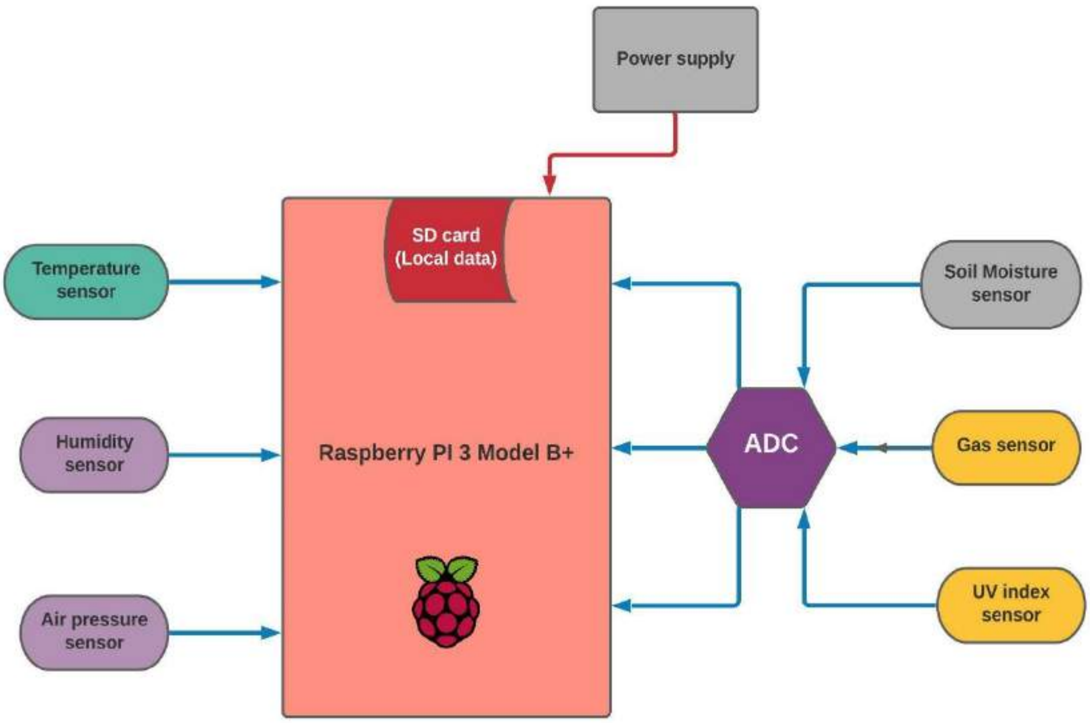


Figure 4.1 Block diagram of node of a Micro Weather Station

The Figure 4.1 shows the block diagram of single weather station that represents a node for a whole system. This node senses and accumulates all the weather parameter with the help of various sensors interfaced with the Raspberry Pi 3 Model B+.

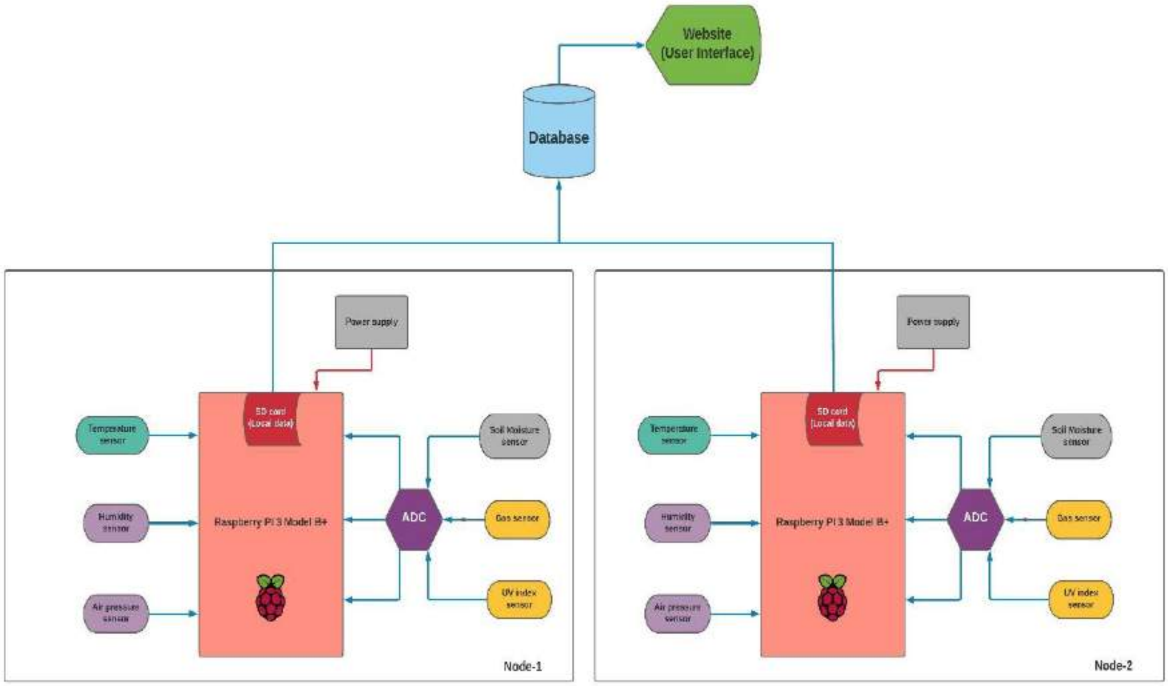


Figure 4.2 Micro Weather Station with two nodes.

For simplification in this chapter, we will be discussing about a hardware and software requirements for single node of a Micro Weather Station, same with the multiple is needed in case of multiple working nodes, as shown in the Figure 4.2.

The single node of a Micro Weather Station majorly has 5 components.

- ❖ Raspberry Pi 3 Model B+
- ❖ Sensors and ADC
- ❖ Power supply
- ❖ Database
- ❖ Website/User Interface (UI)

4.3 Raspberry Pi 3 Model B+

Raspberry Pi 3 Model B+ is a 64-bit quad-core processor with 1.4GHz clock frequency. It comes with dual band wireless LAN, Bluetooth 4.2/BLE (Bluetooth Low Energy), faster Ethernet and Power over Ethernet support.

Figure 4.3 Raspberry Pi 3 Model B+

To get started we need an SD card with NOOBS OS preinstalled in it and a high quality 5V/2.5A micro-USB power supply. It has inbuilt Broadcom BMC2837B0 and Cortex- A53 processor, 1GB of SDRAM. For wireless communication it has 2.4 GHz and 5GHz IEEE 802.11.b/g/n/ac wireless LAN and a Bluetooth 4.2/BLE. For wired communication it has Gigabit Ethernet over USB 2.0 with maximum throughput of 300Mbps. For interfacing it has extended 40 GPIO pins, full size High-Definition Multimedia Interface (HDMI), 4 USB 2.0 ports, CSI camera port, DSI display port for connecting touchscreen display and a Micro SD port for loading of OS and storing data as shown in Figure 4.3.

The sensors are interfaced to Raspberry Pi 3 by connecting it to GPIO pins. The data from some of the sensors is analog hence an Analog to Digital converter (ADC) is need since Raspberry Pi 3 accepts only digital data. The four USB 2.0 ports are used to connect the external peripherals such as mouse, keyboard and pen drive. The full-size HDMI port is used to connect screen or monitor. Overall, it acts as a micro-computer use for different application and projects.

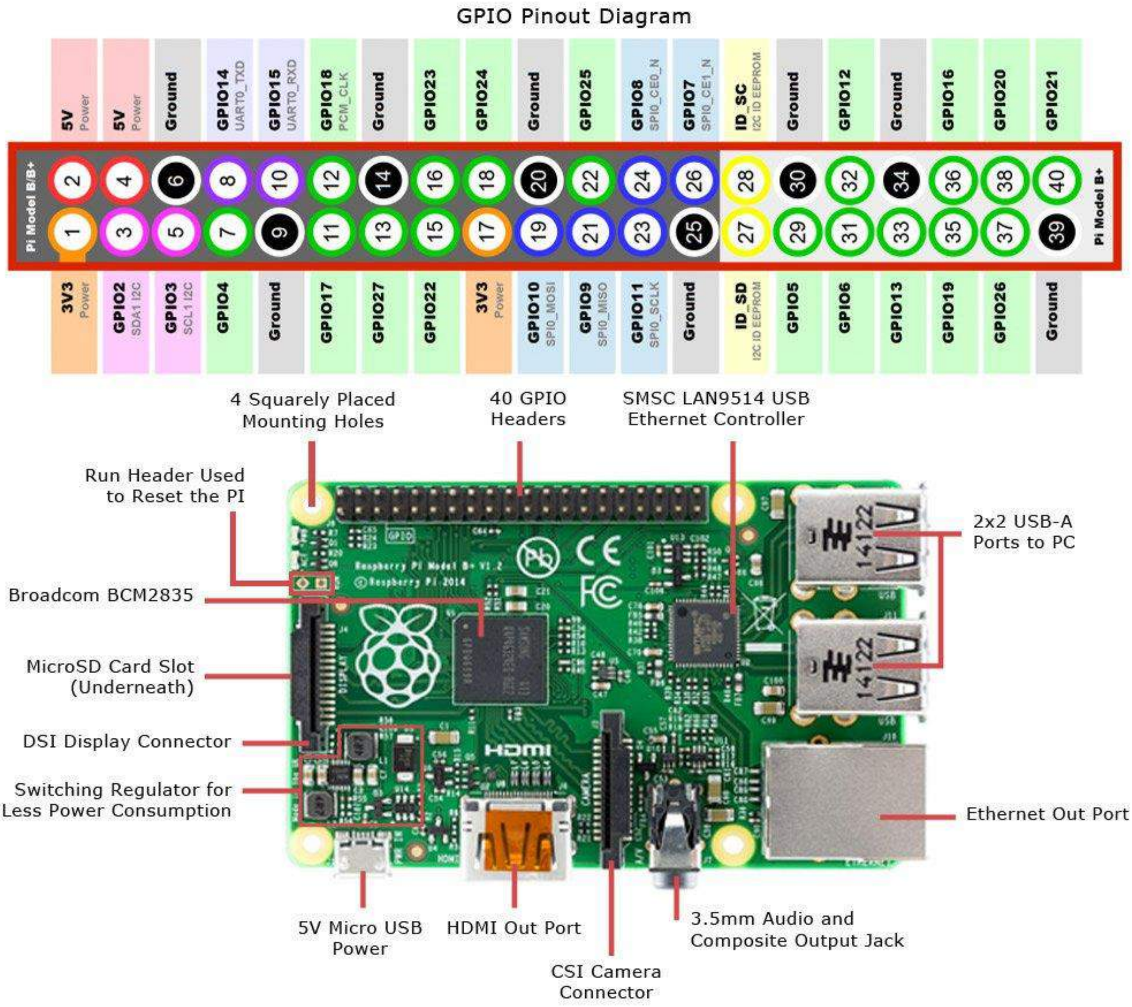


Figure 4.4 Pinout Diagram of Raspberry Pi 3 Model B+

The GPU uses OpenGL ES version 2.0, hardware-accelerated OpenVG API, and 1080p30 H.264 high-profile decode. It can provide up to 1Gpixel/s, 1.5Gtexel/s, or 24 GFLOP's of a general-purpose computer. This connector is used for delivering 5V power to the board. It consumes approx. 170 to 200mA more power than model B.

The power connector is also re-positioned in the new B+ model and placed next to the HDMI socket. B+ model comes with a facility of Power over Ethernet (PoE) a new feature added in this device which allows us to power the board using the Ethernet cables. The B+ version also comes with other improvements like the SD memory slot is replaced by a micro-SD memory card slot as shown in Figure 4.4. The status LEDs on the board now only contain red and green colors and are relocated to the opposite end of the board.

40 pin headers are used to develop an external connection with the electronic device. This is the same as the previous versions, making it compatible with all the devices where the old versions can be used. Out of 40 pins, 26 are used as the digital input output pins and 9 of the remaining 14 pins are termed as dedicated input output pins which indicate they don’t come with alternative functions.

Pin 3 and 5 comes with an onboard pull up resistor with 1.8kΩ and pin 27 and 28 are dedicated to ID EEPROM. In B+ model the GPIO header are slightly re-positioned to allow more space for the additional mounting hole. The device that are compatible with the B model may work with B+ version, however they may not sit identically to the previous versions.

4.4 Sensors and ADC

Sensor is an electronic device that is used to measures quantity or quality of a parameter that it is desired to measure. There are mainly two types of sensors based on the type of output the sensor gives, they are Analog sensors and Digital sensors.

In some cases, the output of the sensor is converted from one form to another i.e. analog to digital or vice versa using Analog to Digital converter (ADC) or Digital to Analog converter (DAC). In the design and implementation of Micro Weather Station the following sensors are used to measure various weather parameters.

- DSB18B20: Temperature sensor
- Capacitive soil moisture sensor v1.2
- MQ135: Gas sensor
- DHT22: Temperature and Humidity sensor
- BMP280: Pressure Sensor
- GUVA-S12SD: UV Light sensor

4.4.1 Temperature sensor: DS18B20

The DSB18B20 sensor is a high precision temperature sensor. It is a digital sensor which uses single bus to acknowledge and receive the temperature. The principle used by this series of temperature sensor is that the resistance of the conductor decides the temperature in surrounding. One must use One Wire protocol for acknowledge and receive the temperature data.

The sensor is encapsulated with waterproof high-quality stainless steel which helps to protect the conductor from external factors and also avoids rusting of conductor. DSB18B20 takes 3.0V~5.5V to power up. It has 3 connections, red wire (VCC), white wire (Data) and black wire (GND). It comes with the 9-12 bit select-able resolution. It has a high accuracy with +/- 0.5°C temperature ranging from -55°C to +125°C.

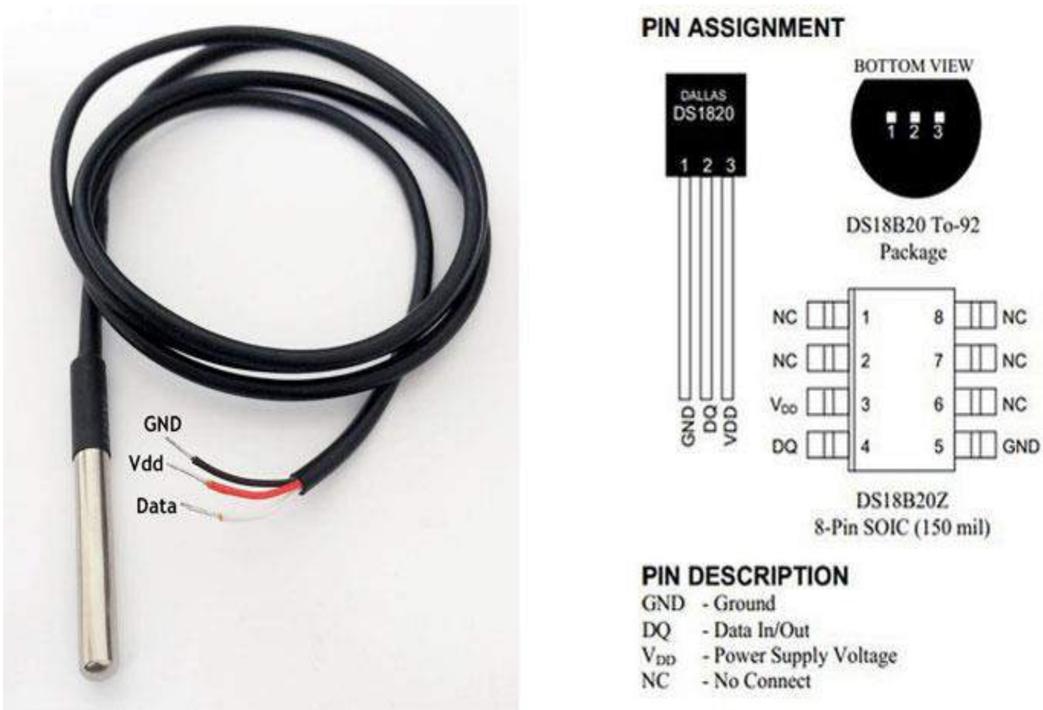


Figure 4.5 DS18B20 Temperature sensor

The OneWire bus system uses a single bus master to control one or more slave devices. The DS18B20 is always a slave. When there is only one slave on the bus, the system is referred to as a “single-drop” system; the system is “multi drop” if there are multiple slaves on the bus. All data and commands are transmitted least significant bit first over the OneWire bus as shown in Figure 4.5.

The OneWire bus requires an external pull-up resistor of approximately 5kΩ as shown in Figure 4.6, thus the idle state for the OneWire bus is high. If for any reason a transaction needs to be suspended, the bus must be left in the idle state. Infinite recovery time can occur between bits so long as the 1-Wire bus is in the inactive (high) state during the recovery period. If the bus is held low for more than 480µs, all components on the bus will be reset.

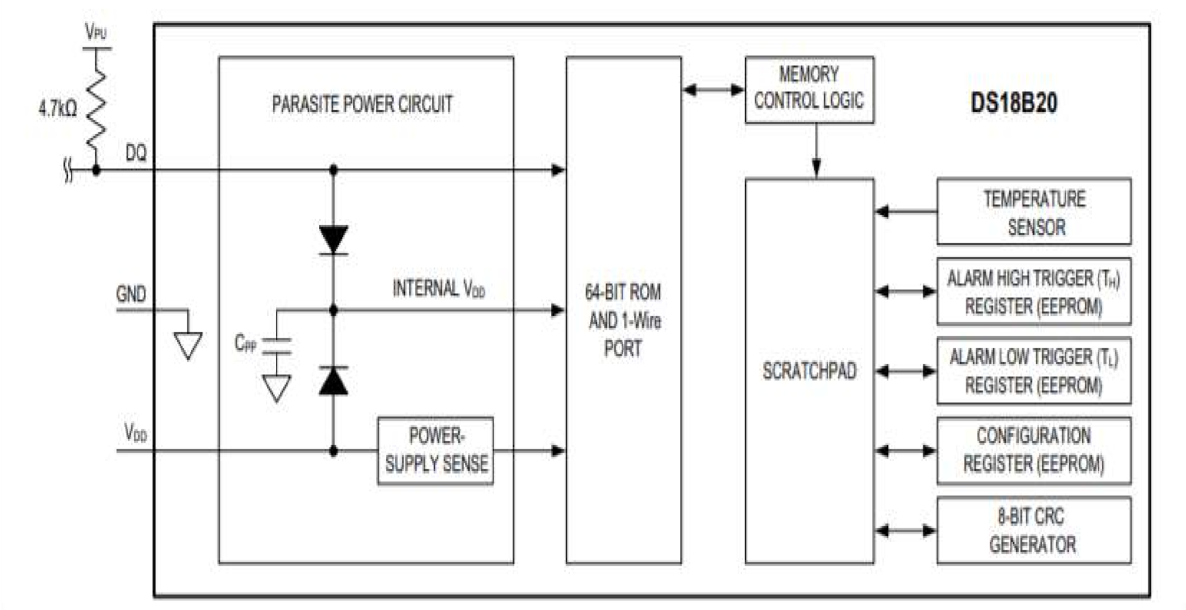


Figure 4.6 Block diagram of DSB18B20 Temperature sensor

Table 4.1 Temperature/Data Relationship

Temperature(°C)	Digital Output (BINARY)	Digital Output (HEX)
+125	0000 0111 1101 0000	07D0h
+85	0000 0101 0101 0000	0550h
+25.0625	0000 0001 1001 0001	0191h
+10.125	0000 0000 1010 0010	00A2h
+0.5	0000 0000 0000 1000	0008h
0	0000 0000 0000 0000	0000h
-0.5	1111 1111 1111 1000	FFF8h
-10.125	1111 1111 0101 1110	FF5Eh
-25.0625	1111 1110 0110 1111	FE6Fh
-55	1111 1100 1001 0000	FC90h

Table 4.2 Temperature Register Format

	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
LS BYTE	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}
	BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8
MS BYTE	S	S	S	S	S	2^6	2^5	2^4

S = SIGN

The DS18B20 output temperature data is calibrated in degrees Celsius; for Fahrenheit applications, a lookup table or conversion routine must be used. The temperature data is stored as a 16-bit sign-extended two's complement number in the temperature register as shown in Table 4.2. The sign bits(s) indicate if the temperature is positive or negative: for positive numbers S = 0 and for negative numbers S = 1. If the DS18B20 is configured for 12-bit resolution, all bits in the temperature register will contain valid data. For 11-bit resolution, bit 0 is undefined. For 10-bit resolution, bits 1 and 0 are undefined, and for 9-bit resolution bits 2, 1, and 0 are undefined. Table 4.1 gives examples of digital output data and the corresponding temperature reading for 12-bit resolution conversions.

The DS18B20 uses Maxim's exclusive 1-Wire bus protocol that implements bus communication using one control signal. The control line requires a weak pull-up resistor since all devices are linked to the bus via a 3-state or open-drain port (the DQ pin in the case of the DS18B20). In this bus system, the microprocessor (the master device) identifies and addresses devices on the bus using each device's unique 64-bit code. Because each device has a unique code, the number of devices that can be addressed on one bus is virtually unlimited.

4.4.2 Capacitive soil moisture sensor v1.2

This soil moisture sensor measures soil moisture levels by **c**apacitive sensing rather than resistive sensing like other sensors on the market. It is made of corrosion-resistant material which gives it excellent service life. Insert it into the soil around your plants and monitor the real-time soil moisture data. This module includes an on-board voltage regulator which gives it an operating voltage range of 3.3 ~ 5.5V. It is perfect for low-voltage micro-controller with both 3.3V and 5V power supply.

The greater is the soil moisture, the higher the capacitance of the sensor. The voltage on the data pin can be measured by an analog pin of the micro controller which represents the humidity of the soil. There is a fixed frequency oscillator that is built with a 555 Timer IC. The square wave generated is then fed to sensor like a capacitor. To a square wave signal that capacitor, however, has a certain resistance, or for argument's sake a resistance that forms a voltage divider with a pure ohm type resistor (the 10k one on pin 3).

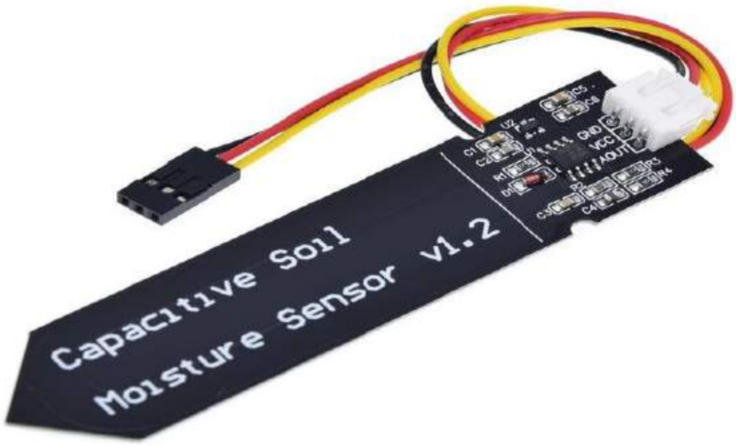


Figure 4.7 Capacitive soil moisture sensor

The analog capacitive soil moisture sensor as shown in Figure 4.7 measures soil moisture levels by capacitive sensing, i.e., capacitance is varied on the basis of water content present in the soil. The capacitance is converted into voltage level basically from 1.2V to 3.0V maximum. The advantage of Capacitive Soil Moisture Sensor is that they are made of a corrosion-resistant material giving it a long service life. This sensor measures the volumetric content of the water inside the soil and gives us the moisture level as the output.

A low dropout 3.3 V voltage regulator supplies a TL555I CMOS timer whose output signal feeds a low pass filter (10 kΩ resistor and the moisture sensing co-planar capacitor). The main function of this stage is to produce a stationary saw-tooth double-exponential waveform whose average value is the same average value of the TL555I output. However, the peak-to-peak voltage of the waveform depends on the effective dielectric constant of the soil. Then, a peak voltage detector provides the analog output signal that we acquire through the ADC of the micro-controller.

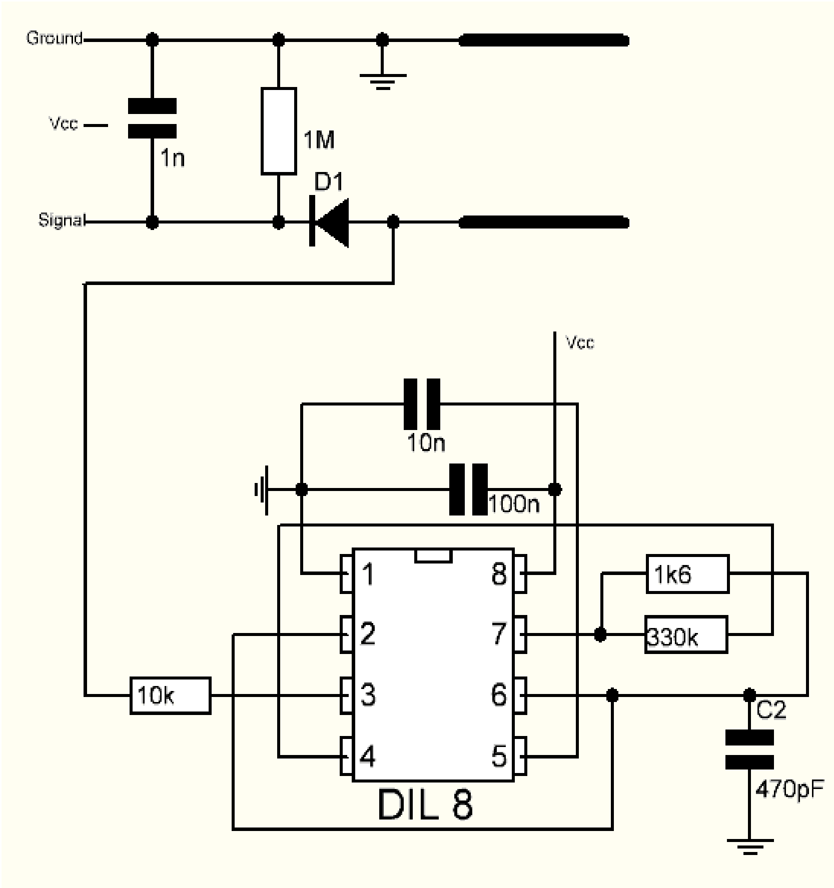


Figure 4.8 Schematic diagram of Capacitive soil moisture sensor

While talking about the accuracy, the capacitive soil moisture sensor is not so much accurate as expected. But you can do the calibration to get the closest accurate reading. Just upload the simple code to micro controller and check the sensor analog reading when the sensor is in dry air and when the sensor is in water as shown in Figure 4.8. From here you can find the maximum and minimum analog value that can be mapped to percentage value from 0 to 100% as per program.

4.4.3 Gas sensor: MQ135

When it comes to measuring or detecting a particular gas the MQ series gas sensors are the most inexpensive and commonly used ones. MQ135 is available as a module or as just the sensor alone. If in the application where one has to detect (not measuring PPM) the presence of a gas then you can buy it as a module since it comes with an op-amp comparator and a digital output pin. But in case if one has to measure the Parts Per Million (PPM) of a gas it is recommend buying the sensor alone without module.

Figure 4.9 Air quality sensor MQ135

The MQ-135 Gas sensor as shown in Figure 4.9 is used in air quality control equipment's and are suitable for detecting or measuring of NH3, NOx, Alcohol, Benzene, Smoke, CO2. The MQ-135 sensor module comes with a Digital Pin which makes this sensor to operate even without a micro-controller and that comes in handy when you are only trying to detect one particular gas. To measure the gases in PPM the analog pin needs to be used. The analog pin is Transistor to Transistor Logic (TTL) driven and works on 5V and so can be used with most common micro-controllers.

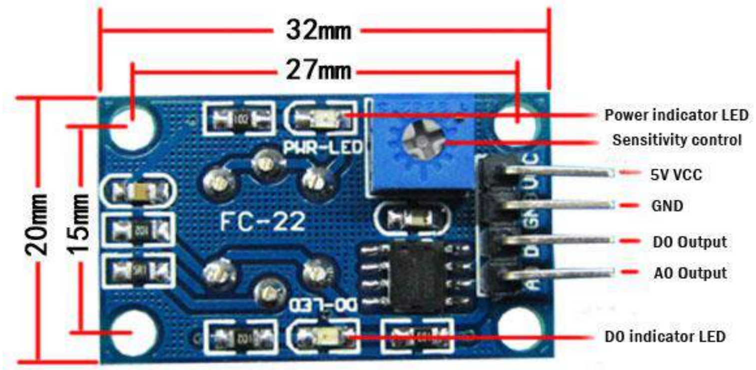


Figure 4.10 Dimensions and Pinout diagram of MQ135

MQ135 either uses the digital pin or the analog pin to do this. Simply power the module with 5V and you should notice the power LED on the module to glow and when no gas it detected the output LED will remain turned off meaning the digital output pin will be 0V as shown in Figure 4.10. Remember that these sensors have to be kept on for pre-heating time before one can actually work with it. Now, introduce the sensor to the gas you want to detect and you should see the output LED to go high along with the digital pin, if not use the potentiometer until the output gets high. Now every time the sensor gets introduced to this gas at this particular concentration the digital pin will go high (5V) else will remain low (0V).

The analog pin is also used to achieve the same thing. Read the analog values (0-5V) using a micro-controller, this value will be directly proportional to the concentration of the gas to which the sensor detects.

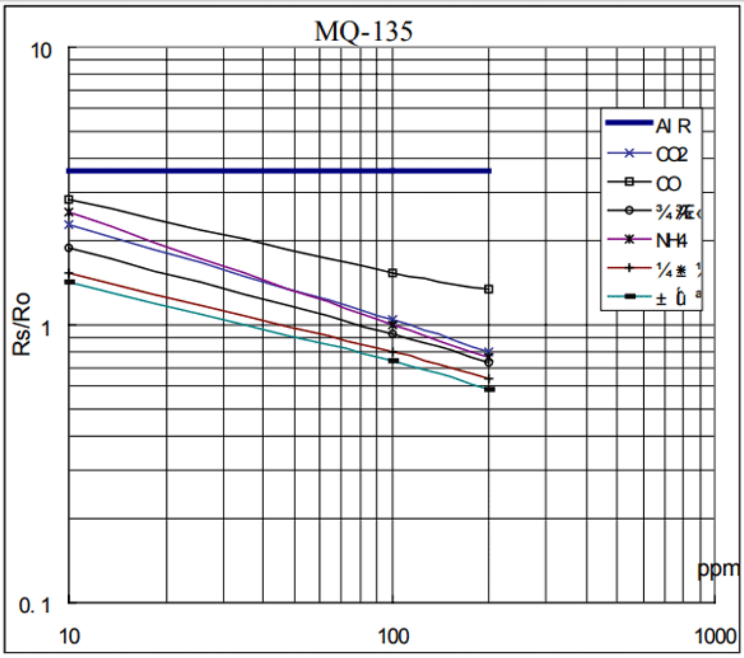


Figure 4.11 Typical sensitivity characteristics of the MQ-135 for several gases

MQ-135 gas sensor applies SnO2 which has a higher resistance in the clear air as a gas-sensing material. When there is an increase in polluting gases, the resistance of the gas sensor decreases along with that. To measure PPM using MQ-135 sensor we need to look into the (Rs/Ro) v/s PPM graph taken from the MQ135 data-sheet shown in the Figure 4.11.

The typical sensitivity characteristics of the MQ-135 sensor for several gases can be given as:

- ❖ Temp: 20℃, Humidity: 65%, O2 concentration 21%, RL=20kΩ,
- ❖ Ro: sensor resistance at 100ppm of NH3 in the clean air.
- ❖ Rs: sensor resistance at various concentrations of gases.

The value of Ro is the value of resistance in fresh air (or the air with we are comparing) and the value of Rs is the value of resistance in Gas concentration. First one should calibrate the sensor by finding the values of Ro in fresh air and then use that value to find Rs using the below formula:

Resistance of the sensor (Rs) = (Vc/VRL-1) × RL

Calculate Rs and Ro, then find the ratio and then using the graph shown above calculate the equivalent value of PPM for that particular gas.

4.4.4 Temperature and Humidity sensor: DHT22

DHT22 is an output calibrated digital signal. It utilizes exclusive digital signal collecting technique and humidity sensing technology, assuring its reliability and stability.

The DHT22 sensor is the successor of the DHT11 module, it can either be purchased as a sensor or as a module. Either way the performance of the sensor is same. The only difference between the sensor and module is that the module will have a filtering capacitor and pull-up resistor inbuilt, and for the sensor you have to use them externally if required. DHT22 is slightly costly than the DHT11, but it has a higher measuring range and slightly better accuracy.

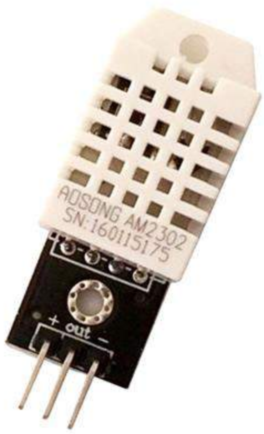


Figure 4.12 DHT22 Temperature and Humidity sensor

The only difference between the sensor and module is that the module will have a filtering capacitor and pull-up resistor inbuilt, and for the sensor you have to use them externally if required. DHT22 as shown in Figure 4.12 is slightly costly than the DHT11, but it has a higher measuring range and slightly better accuracy.

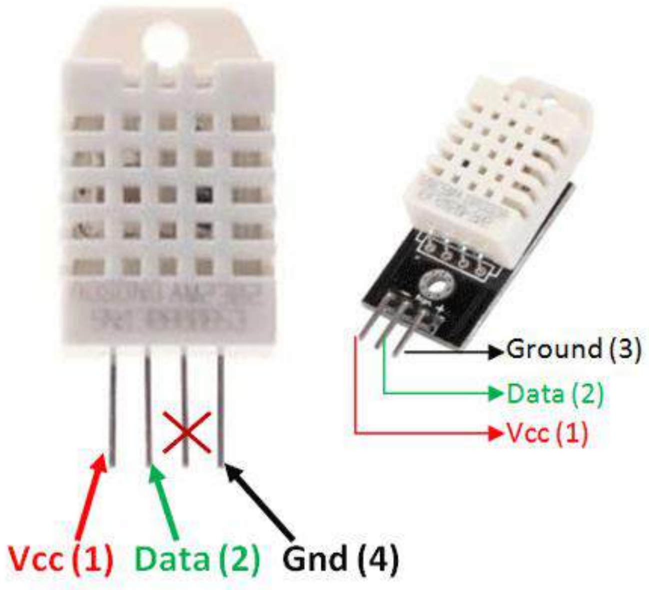


Figure 4.13 Pin-out diagram of DHT22 sensor(left) and module(right).

The sensor will come as a 4-pin package as shown in Figure 4.13 out of which only three pins will be used whereas the module will come with just three pins as shown in the DHT22 pin-out. Power's voltage should be 3.3-6V DC. When power is supplied to sensor, don't send any instruction to the sensor within one second to pass unstable status. One capacitor valued 100nF can be added between VDD and GND for wave filtering.

Single-bus data is used for communication between MCU and DHT22, it takes 5ms for single time communication. When MCU send start signal, DHT22 change from low-power-consumption-mode to running-mode. When MCU finishes sending the start signal, DHT22 will send response signal of 40-bit data that reflect the relative humidity and temperature information to MCU. Without start signal from MCU, DHT22 will not give response signal to MCU. One start signals for one time's response data that reflect the relative humidity and temperature information from DHT22. DHT22 will change to low-power-consumption-mode when data collecting is finished or if it doesn’t receive start signal from MCU again.

DHT22 has a polymer capacitor that is used to measures temperature in the range from -40°C~ +80°C with an accuracy of +/-1°C and relative humidity form 0~100% with an accuracy of +/-1%. It has sensing period of 2s.

4.4.5 Pressure sensor: BMP280

BMP280 Barometric Pressure and Altitude Sensor I2C/SPI Module is a cheapest and tiny Atmospheric Sensor Breakout to measure barometric pressure, and temperature readings all without taking up too much space. Basically, anything you need to know about atmospheric conditions you can find out from this tiny breakout. The BMP280 Breakout has been designed to be used in indoor/outdoor navigation, weather forecasting, home automation, and even personal health and wellness monitoring.

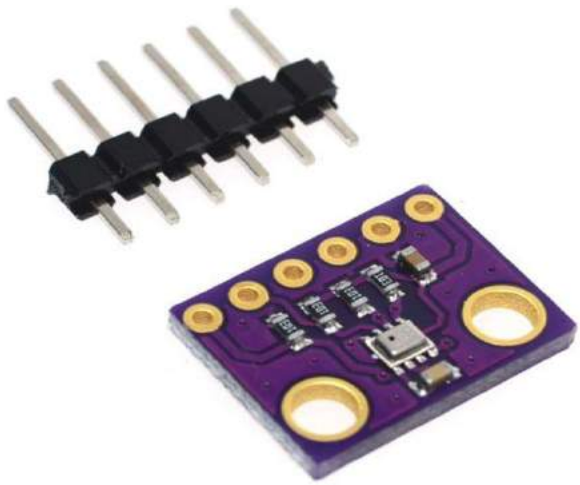


Figure 4.14 BMP280 Pressure and Altitude module

This module uses an environmental sensor manufactured by Bosch with temperature, a barometric pressure sensor that is the next generation upgrade to the popular BMP085/BMP180/BMP183 Sensor as shown in Figure 4.14. This sensor is great for all sorts of weather sensing and can even be used in both I2C and SPI; This precision sensor from Bosch is the best low-cost, precision sensing solution for measuring barometric pressure with ±1 hPa absolute accuracy, and temperature with ±1.0°C accuracy. Because pressure changes with altitude and the pressure measurements are so good; you can also use it as an altimeter with ±1-meter accuracy.

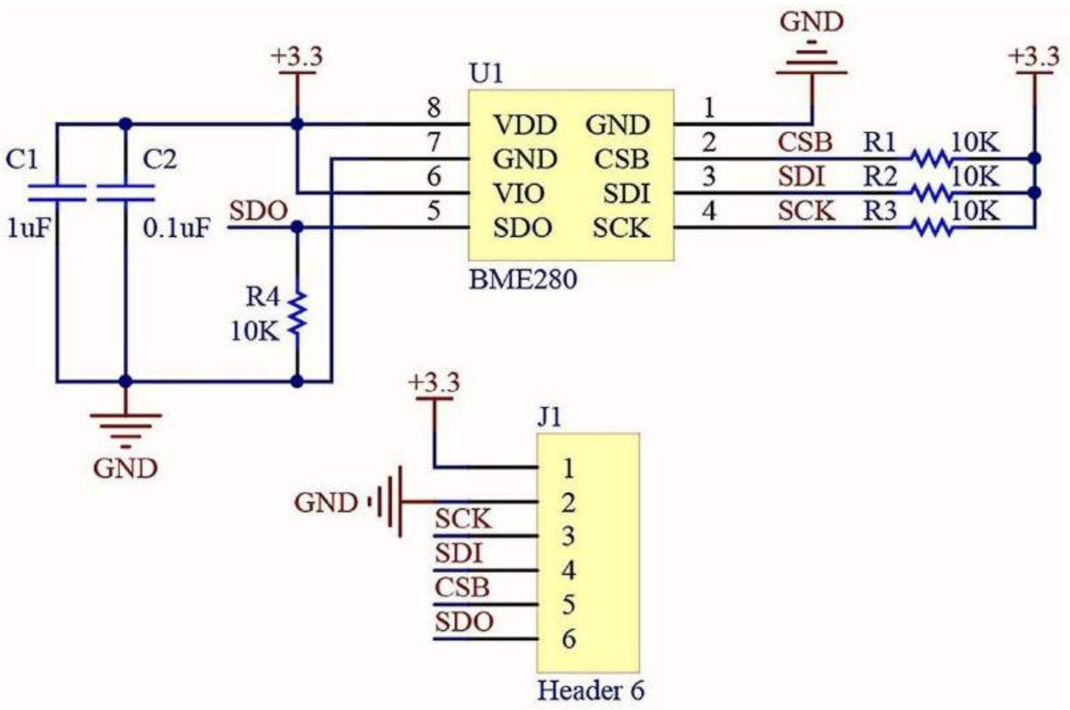


Figure 4.15 Circuit diagram of BMP280.

The BMP280 is the next-generation of sensors and is the upgrade to the BMP085/BMP180/BMP183 – with a low altitude noise of 0.25m and the same fast conversion time. It has the same specifications but can use either I2C or SPI as shown in Figure 4.15. For simple easy wiring, go with I2C. If you want to connect a bunch of sensors without worrying about I2C address collisions, go with SPI.

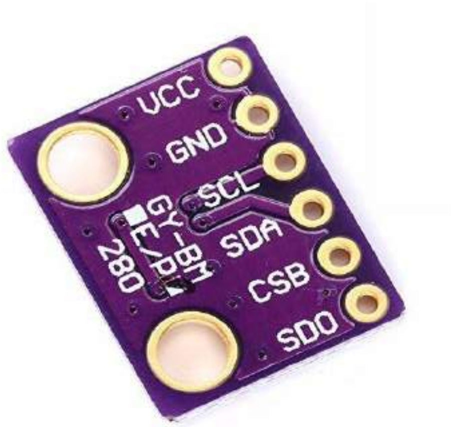


Figure 4.16 Pinout diagram of BMP280

The operating voltage of BMP280 is 1.71V to 3.6V, typical operating voltage is 3.3V, the operating temperature is -40°C to +85°C. BMP280 module is able to define pressure from 300hPa to 1100hPa as shown in Figure 4.16.

4.4.6 UV detection module: GUVA-S12SD

An UV Sensor is used for detecting the intensity of incident ultraviolet (UV) radiation lying in the ultraviolet range, with wave lengths shorter than light but longer than X-rays. UV sensors are widely used in many different applications, including automobiles, pharmaceuticals and robotics. Now a days, a vast range of pre-wired UV sensor modules are readily available for use by amateur and professional design engineers and hobbyists. These micro-controller compatible modules can be used for detecting the intensity of incident ultraviolet radiation, like the UV radiation in sunlight.

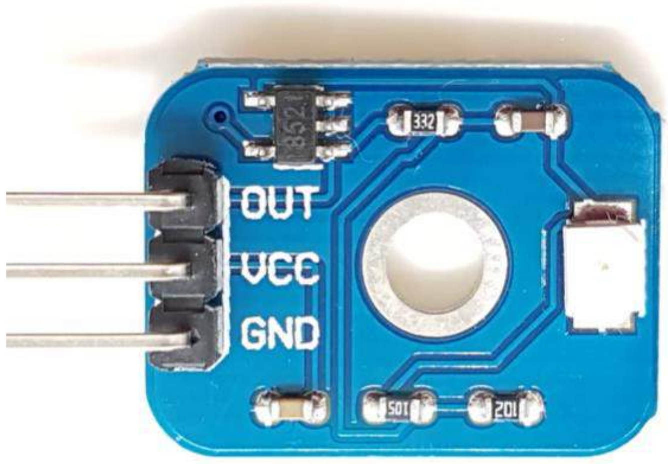


Figure 4.17 UV light sensor GUVA-S12SD

The UV Sensor is used for detecting the intensity of incident ultraviolet (UV) radiation like UV radiation in sunlight. This form of electromagnetic radiation has shorter wavelengths than visible radiation. This module as shown in Figure 4.17 is based on the sensor GUVA-S12SD and SGM8521 Op-amp, which has a wide spectral range of 200nm-370nm. The module outputs calibrated analog output voltage which varies with the UV intensity.

To use, power the sensor and op-amp by connecting V+ to 2.7-5.5VDC and GND to power ground. Then read the analog signal from the OUT pin.

Output Voltage (Vo) = 4.3 * Diode Current(µA).

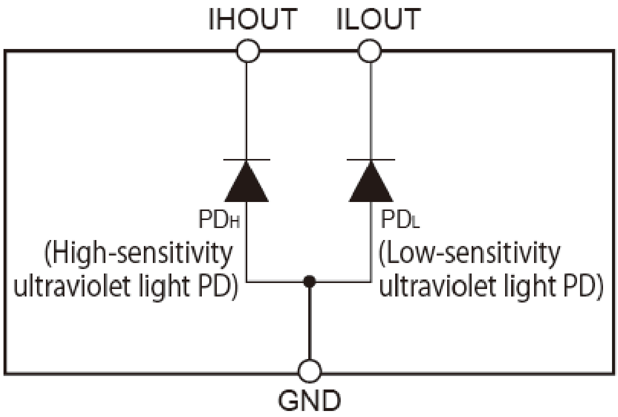


Figure 4.18 Block diagram of GUVA-S12SD

The UV Sensor module as shown in Figure 4.18 is based on the renowned sensor GUVA-S12SD from EOC. GUVA-S12SD is a Gallium Nitride material based Schottky-type photodiode, optimized for photovoltaic mode operation. Next main component is an Op-Amp IC SGM8521 from SG-MICRO. The SGM8521 is a rail-to-rail input and output voltage feedback amplifiers offering low cost. This Op-Amp have a wide input common-mode voltage range and output voltage swing, and take the minimum operating supply voltage down to 2.1V and the maximum recommended supply voltage is 5.5V. Besides, SGM8521 provides 150kHz bandwidth at a low current consumption of 4.7 µA. Figure 4.19 represents UV index value for different output voltage value.

UV Index	0	UV INDEX 1	UV INDEX 2	UV INDEX 3	UV INDEX 4	UV INDEX 5
Vout(mV)	<50	227	318	408	503	606

UV Index	UV INDEX 6	UV INDEX 7	UV INDEX 8	UV INDEX 9	UV INDEX 10	UV INDEX 11+
Vout(mV)	696	795	881	976	1079	1170+

Figure 4.19 Classification of UV index based on the output of sensor

4.4.7 Analog to Digital converter (ADC)

In electronics, an analog-to-digital converter (ADC, A/D, or A-to-D) is a system that converts an analog signal, such as a sound picked up by a microphone or light entering a digital camera, into a digital signal. An ADC may also provide an isolated measurement such as an electronic device that converts an input analog voltage or current to a digital number representing the magnitude of the voltage or current. Typically, the digital output is a two's complement binary number that is proportional to the input, but there are other possibilities.

There are several ADC architectures. Due to the complexity and the need for precisely matched components, all but the most specialized ADCs are implemented as integrated circuits (ICs). These typically take the form of metal–oxide–semiconductor (MOS) mixed-signal integrated circuit chips that integrate both analog and digital circuits. An ADC converts a continuous-time and continuous-amplitude analog signal to a discrete-time and discrete-amplitude digital signal. The conversion involves quantization of the input, so it necessarily introduces a small amount of error or noise.

Figure 4.20 4-channel stereo multiplexed analog-to-digital converter ADS1115

The performance of an ADC as shown in Figure 4.20 is primarily characterized by its bandwidth and signal-to-noise ratio (SNR). The bandwidth of an ADC is characterized primarily by its sampling rate. The SNR of an ADC is influenced by many factors, including the resolution, linearity and accuracy (how well the quantization levels match the true analog signal), aliasing and jitter.

4.5 Microsoft Visual Studio

Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services and mobile apps. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed code.

Visual Studio includes a code editor supporting IntelliSense (the code completion component) as well as code refactoring. The integrated debugger works both as a source-level debugger and a machine-level debugger. Other built-in tools include a code profiler, designer for building GUI applications, web designer, class designer, and database schema designer. It accepts plug-ins that expand the functionality at almost every level, including adding support for source control systems (like Subversion and Git) and adding new tool-sets like editors and visual designers for domain-specific languages or tool-sets for other aspects of the software development lifecycle (like the Azure DevOps client: Team Explorer).

Visual Studio supports 36 different programming languages and allows the code editor and debugger to support (to varying degrees) nearly any programming language, provided a language-specific service exists. Built-in languages include C, C++, C++/CLI, Visual Basic .NET, C#, F#, JavaScript, TypeScript, XML, XSLT, HTML, and CSS. Support for other languages such as Python, Ruby, Node.js, and M among others is available via plug-ins. Java (and J#) were supported in the past.

Visual Studio Code collects usage data and sends it to Microsoft, although this can be disabled. In addition, because of the open-source nature of the application, the telemetry code is accessible to the public, who can see exactly what is collected. According to Microsoft, the data is shared with Microsoft-controlled affiliates and subsidiaries, although law enforcement may request it as part of a legal process.

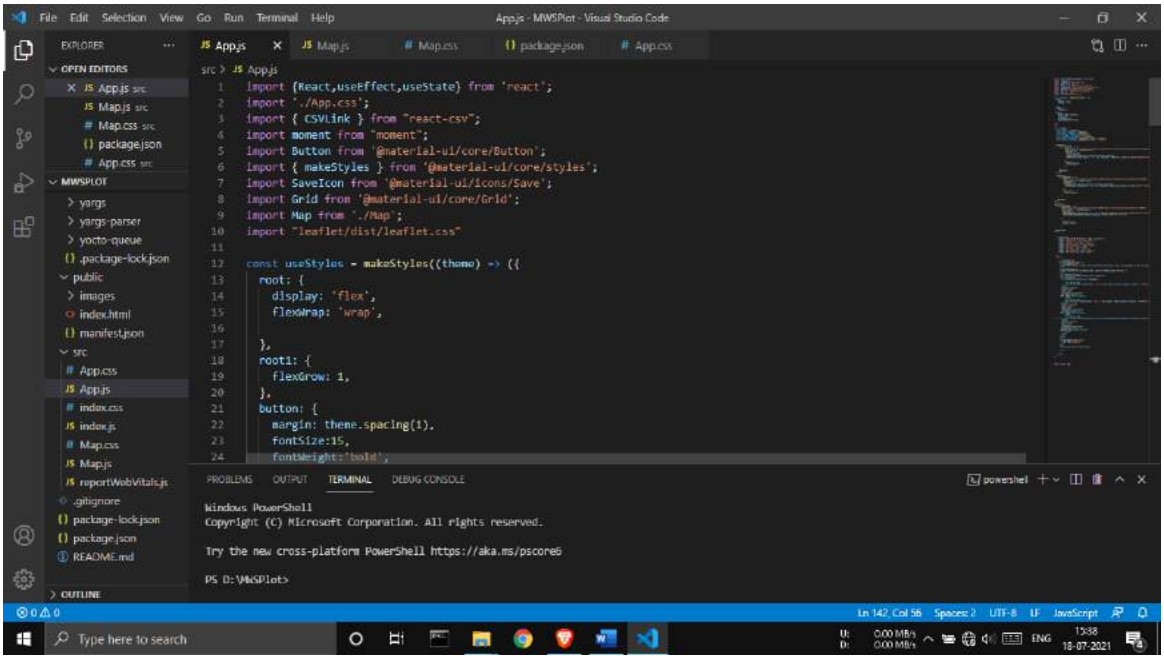


Figure 4.21 User interface of Visual Studio Code.

Support for programming languages is added by using a specific VS Package called a Language Service. A language service defines various interfaces which the VS Package implementation can implement to add support for various functionalities. Functionalities that can be added this way include syntax coloring, statement completion, brace matching, parameter information tool tips, member lists, and error markers for background compilation. If the interface is implemented, the functionality will be available for the language. Language services are implemented on a per-language basis. The implementations can reuse code from the parser or the compiler for the language. Language services can be implemented either in native code or managed code. For native code, either the native COM interfaces or the Babel Framework (part of Visual Studio SDK) can be used. For managed code, the MPF includes wrappers for writing managed language services.

Visual Studio (like any other IDE) as shown in Figure 4.21 includes a code editor that supports syntax highlighting and code completion using IntelliSense for variables, functions, methods, loops, and LINQ queries. Intelli-Sense is supported for the included languages, as well as for XML, Cascading Style Sheets, and JavaScript when developing web sites and web applications. Auto complete suggestions appear in a modeless list box over the code editor window, in proximity of the editing cursor. In Visual Studio 2008 onwards, it can be made temporarily semi-transparent to see the code obstructed by it. The code editor is used for all supported languages.

Visual Studio includes a debugger that works both as a source-level debugger and as a machine-level debugger. It works with both managed code as well as native code and can be used for debugging applications written in any language supported by Visual Studio. In addition, it can also attach to running processes, monitor, and debug those processes. If source code for the running process is available, it displays the code as it is being run. If source code is not available, it can show the disassembly. The Visual Studio debugger can also create memory dumps as well as load them later for debugging. Multi-threaded programs are also supported. The debugger can be configured to be launched when an application running outside the Visual Studio environment crashes.

4.6 Raspberry Pi OS

Raspberry Pi Operating System (OS) is highly optimized for the Raspberry Pi line of compact single-board computers with ARM CPUs. It runs on every Raspberry Pi except the Pico micro-controller. Raspberry Pi OS uses a modified LXDE as its desktop environment with the Openbox stacking window manager, along with a unique theme. The distribution is shipped with a copy of the algebra program Wolfram Mathematica and a version of Minecraft called Minecraft: Pi Edition, as well as a lightweight version of the Chromium web browser.

Raspberry Pi OS looks similar to many common desktops, such as macOS and Microsoft Windows. The menu bar is positioned at the top and contains an application menu and shortcuts to Terminal, Chromium, and File Manager. On the right is a Bluetooth menu, a Wi-Fi menu, volume control, and a digital clock. The Raspberry Pi documentation recommends at least a 4GB microSD card for Raspberry Pi OS Lite, an 8GB microSD card for Raspberry Pi OS, and a 16GB microSD card for Raspberry Pi OS Full. The image files themselves are 442MB, 1,175MB, and 2,868MB respectively.

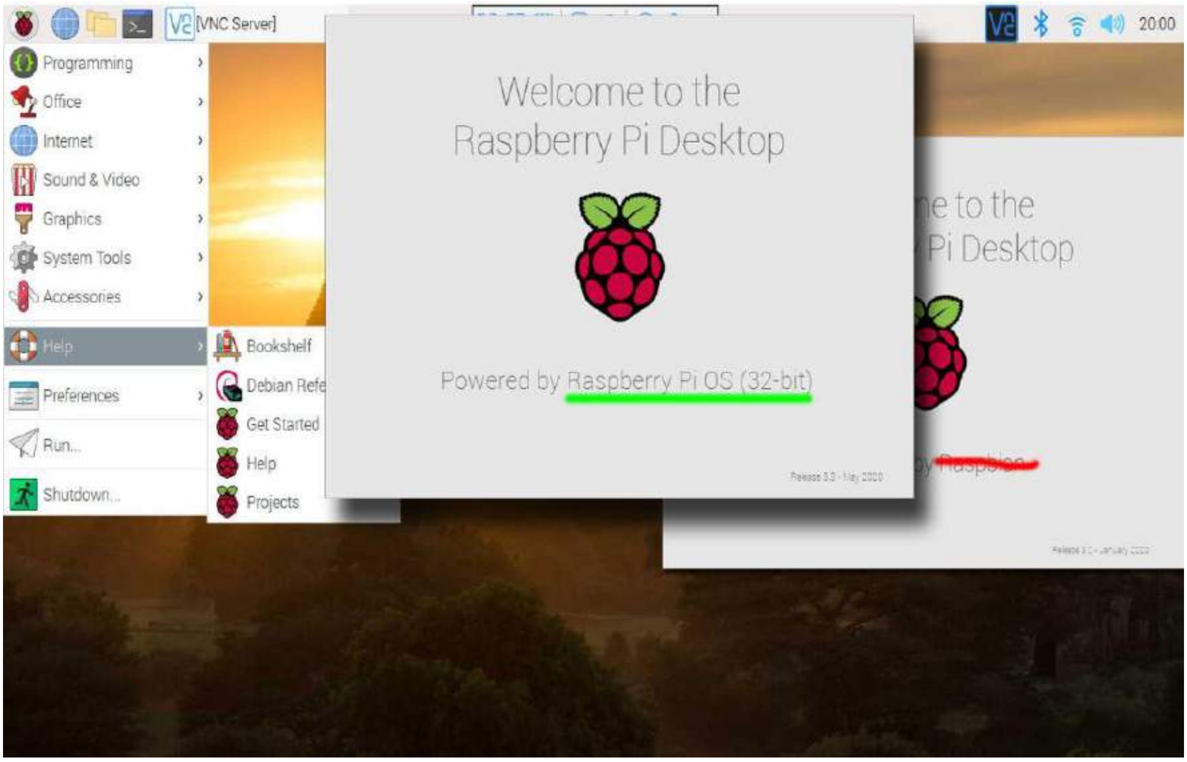


Figure 4.22 Interface of Raspberry Pi Operating system

In late January 2021, Raspberry Pi OS' raspberrypi-sys-mods package added a trusted GPG key and sources.list.d entry to APT without user consent as shown in Figure 4.22. This addition granted Microsoft the ability to install and run any software during the daily critical update process on all Pi that had done a manual apt upgrade to receive the change. The change was not pushed as a critical update and, as of yet, the excessive permission has not been abused by Microsoft and would seem unlikely to ever be abused. The author of the change acknowledged on GitHub that too many rights were granted to Microsoft and also acknowledged delaying the public release of the source code for the change.

In addition to the permissions, the change also causes Pi running an updated Raspberry Pi OS to contact packages.microsoft.com daily and thereby reveal their IP address as a Raspberry Pi OS user for potential use in tracking or marketing efforts. On 8 February 2021, the original author made another change that restricted Microsoft's ability to install software to packages beginning with the string "code" but Microsoft can still run code as root so this restriction is trivial to bypass. As of 8 February 2021, the issue is not resolved and the Raspberry Pi Foundation has locked or deleted many of the related threads on their public forum and their GitHub pages but has acknowledged there is a problem to be resolved and that they are working on it.

4.7 Expo

Expo is an open-source platform for making universal native apps that run on Android, iOS, and the web. It includes a universal runtime and libraries that let you build native apps by writing React and JavaScript. This repository is where the Expo client software is developed, and includes the client apps, modules, apps, and more. The Expo CLI repository contains the Expo development tools.

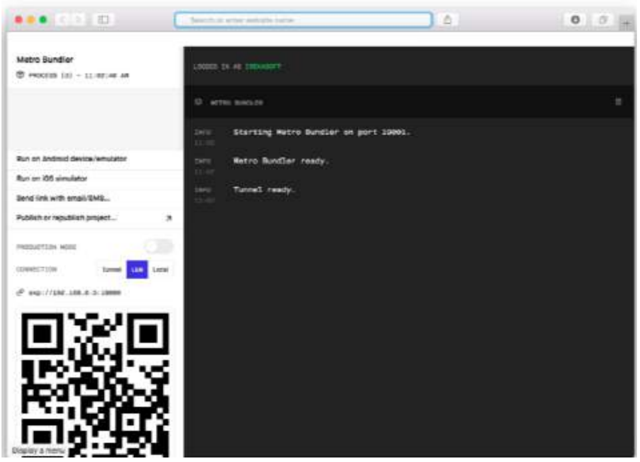


Figure 4.23 User interface of Expo

Expo lets web developers build truly native apps that work across both iOS and Android by writing them once in just JavaScript (JS). So, Expo as shown in Figure 4.23 is a set of tools built on top of React Native (RN). These tools depend on one key belief held at Expo: it's possible to build most apps without ever needing to write native code, provided that you have a comprehensive set of APIs exposed to JavaScript.

React Native does not give all the JS API’s you need out of the box, but only most primitive features. React Native developers are expected to use Android Studio/XCode to link additional native libraries. Expo aims to enhance RN and provide all the JS API one need for the most common needs. It is basically a set of well-defined quality native libraries already packaged in a single library- ExpoKit. Sometimes these libraries are actually already existing in RN world, and integrated into ExpoKit. It is also important to notice that the Expo team can't include every lib out there into ExpoKit because the hello world app size would grow, as it would ship a lot of APIs that wouldn't be used in most apps.

4.8 ThingSpeak

ThingSpeak is an open-source Internet of Things (IoT) application and API to store and retrieve data from things using the HTTP and MQTT protocol over the Internet or via a Local Area Network. ThingSpeak enables the creation of sensor logging applications, location tracking applications, and a social network of things with status updates.

Figure 4.24 User interface of ThingSpeak

ThingSpeak was originally launched by ioBridge in 2010 as a service in support of IoT applications. ThingSpeak as shown in Figure 4.24 has integrated support from the numerical computing software MATLAB from MathWorks, allowing ThingSpeak users to analyze and visualize uploaded data using MATLAB without requiring the purchase of a MATLAB license from MathWorks.

ThingSpeak has a close relationship with MathWorks, Inc. In fact, all of the ThingSpeak documentation is incorporated into the MathWorks MATLAB documentation site and even enabling registered MathWorks user accounts as valid login credentials on the ThingSpeak website. The terms of service and privacy policy of ThingSpeak.com are between the agreeing user and MathWorks, Inc.

HARDWARE IMPLEMENTATION

CHAPTER 5

HARDWARE IMPLEMENTATION

5.1 Introduction

This chapter gives the brief introduction to hardware implementation of micro weather station. Raspberry Pi is used to achieve this. The connection of hardware components required to design the micro weather station is discussed below.

5.2 Interfacing of sensors with Raspberry Pi

Interfacing all six sensors with Raspberry Pi and connection of hardware components required to design the micro weather station.

5.2.1 Interfacing DS18B20 Temperature sensor with Raspberry Pi

The DS18B20 is a waterproof sensor, that can provide temperatures over a one-wire interface. The waterproof feature of the sensor makes it perfect for using in wet environment. The sensor looks like a very long cord with a thick part on one end. If the sensor is considered to be a plain version without wiring and waterproofing, then it looks exactly like a transistor. The sensor is pretty accurate being within 0.05°C of the actual temperature. It can handle temperatures of up to 125°C (260°F), but it’s recommended to keep it below 100°C (210°F). The device also has an onboard analog to digital converter, so it can be easily hook it up to a digital GPIO pin on the Pi.

The equipment required for interfacing DS18B20 with Raspberry Pi is listed below.

- Raspberry Pi
- Micro SD card
- DS18B20 Temperature sensor
- Breadboard
- Breadboard wire

- ❖ 4.7K ohm Resistor

The circuit required to build is pretty straightforward, as only a resistor and the temperature sensor is used as shown in Figure 5.1. The breadboard and the breadboard wire are optional, but it is highly recommended using these as they may make working with circuitry a lot easier. The following steps are involved in interfacing DS18B20 Temperature sensor with Raspberry Pi.

- ❖ Connect the 3v3 pin from the Pi to the positive rail and a ground pin to the ground rail on the breadboard.
- ❖ Place the DS18B20 sensor onto the breadboard.
- ❖ Place a 4.7k resistor between the positive lead and the output lead of the sensor.
- ❖ Place a wire from the positive lead to the positive 3v3 rail.
- ❖ Place a wire from the output lead back to Pin 7 (GPIO4) of the Raspberry Pi.
- ❖ Place a wire from the ground lead to the ground rail.

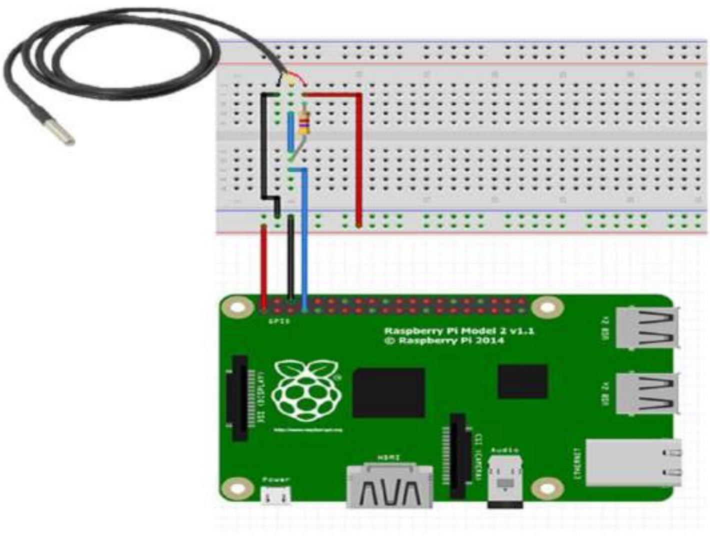


Figure 5.1 Interfacing DS18B20 Temperature sensor with Raspberry Pi

5.2.2 Interfacing DHT22 Humidity sensor with Raspberry Pi

The DHT22 is a versatile and low-cost humidity sensor that can also calculate the temperature of an area. The sensor has a relatively long transmission distance, allowing the sensor to transmit data through wires up to 20m away from the Raspberry Pi. In addition, the DHT22 is a digital sensor with an inbuilt analog to digital converter. The biggest downside to the DHT11 and DHT22 sensors is that they are quite slow sensors. They have a sampling rate of once every second for the DHT11 and once every 2 seconds for the DHT22.

The equipment required to connect the DHT22 Humidity Sensor to your Raspberry Pi are:

- Raspberry Pi
- Micro SD card
- Breadboard wire
- DHT22 Humidity sensor
- 10K ohm Resistor

The DHT22 being a digital sensor, it is incredibly straightforward to connect to the Raspberry Pi as shown in Figure 5.2. The single data pin is able to connect directly to the Raspberry Pi's GPIO pins.

The following steps are involved in interfacing DHT22 Humidity sensor with Raspberry Pi.

- Place a 10k resistor between Pin 1 and Pin 2 of the DHT22.
- Wire Pin 1 of the DHT22 to Physical Pin 1 (3v3) on the Pi.
- Wire Pin 2 of the DHT22 to Physical Pin 11 (GPIO17) on the Pi.
- Wire Pin 3 of the DHT22 to Physical Pin 6 (GND) on the Pi.

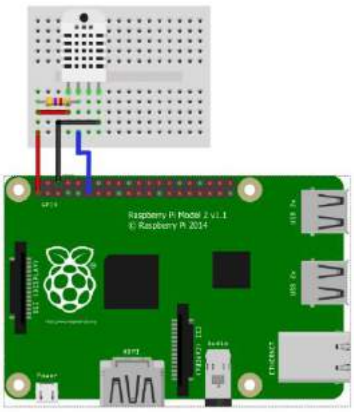


Figure 5.2 Interfacing DHT22 Humidity sensor with Raspberry Pi

5.2.3 Interfacing BMP280 Pressure sensor with Raspberry Pi

The BMP280 is a Piezo-resistive pressure sensor technology featuring high EMC robustness, high accuracy and linearity and long-term stability. The sensor module is housed in an extremely compact 8-pin metal lid Land Grid Array (LGA) package. Its small dimensions and its low power consumption of 2.7µA @ 1Hz allow the implementation in battery driven devices such as mobile phones, GPS modules or watches. The sensor operates at lower noise, supports new filter modes and a Serial Peripheral Interface (SPI) within a footprint 63% smaller than the BMP180.

The BMP280 sensor can be easily interfaced with Raspberry Pi as shown in Figure 5.3 with CircuitPython and the Adafruit CircuitPython BMP280 module. The module allows to write Python code that reads the temperature and pressure from the sensor. The sensor can be used with any CircuitPython microcontroller board or with a computer that has GPIO and Python interpreter.

The equipment required to connect the BMP280 Pressure Sensor to Raspberry Pi are:

- ❖ Raspberry Pi
- ❖ Micro SD card
- ❖ Breadboard
- ❖ BMP280 Pressure sensor

The following steps are involved in interfacing BMP280 Pressure sensor with Raspberry Pi.

- ❖ Sensor VIN Pin is connected to Pi 3V3.
- ❖ Sensor GND Pin is connected to Pi GND.
- ❖ Sensor SDI Pin is connected to Pi MOSI, Pin 19 (GPIO10).
- ❖ Sensor SDO Pin is connected to Pi MISO, Pin 21 (GPIO9).
- ❖ Sensor SCK Pin is connected to Pi SCLK, Pin 23 (GPIO11).
- ❖ Sensor CSB is connected to Pi Pin 29 (GPIO5).

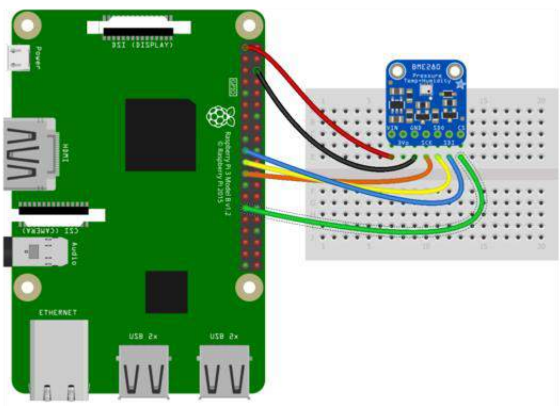


Figure 5.3 Interfacing BMP280 Pressure sensor with Raspberry Pi

5.2.4 Interfacing MQ135 Gas sensor with Raspberry Pi using ADS1115 (ADC)

The MQ-135 Gas sensors are used in air quality control equipments and are suitable for detecting or measuring of NH3, NOx, Alcohol, Benzene, Smoke, CO2. The MQ-135 sensor module comes with a Digital Pin which makes this sensor to operate even without a microcontroller and that comes in handy when you are only trying to detect one particular gas. If you need to measure the gases in PPM the analog pin need to be used. The analog pin is TTL driven and works on 5V and so can be used with most common microcontrollers.

The ADS1115 is an analog to digital converters that are easy to use with the Raspberry Pi using its I2C communication bus. The ADS1115 is a 12-bit ADC and is a higher precision 16-bit ADC with 4 channels. In addition, it has a programmable gain from 2/3x to 16x in order to amplify small signals and read them with higher precision. The ADS1115 uses I2C communication protocol to read analog values. Before wiring the ADC to the Pi, I2C has to be enabled on the Raspberry Pi using raspi-config as shown in Figure 5.4.

The equipment required to connect the MQ135 Gas sensor to Raspberry Pi are:

- ❖ Raspberry Pi
- ❖ Micro SD card
- ❖ Breadboard
- ❖ MQ135 Gas sensor
- ❖ ADS1115 Analog-to-Digital Converter

The MQ135 Gas sensor is connected to the Raspberry Pi as follows:

- ❖ Sensor VDD to Raspberry Pi 3.3V
- ❖ Sensor GND to Raspberry Pi GND
- ❖ Sensor A0 (Analog Out) to ADS1115 A0 (Analog Input 0)
- ❖ ADS1115 VDD to Raspberry Pi 3.3V
- ❖ ADS1115 GND to Raspberry Pi GND
- ❖ ADS1115 SCL to Raspberry Pi SCL, Pin 5 (GPIO3)
- ❖ ADS1115 SDA to Raspberry Pi SDA, Pin 3 (GPIO2)

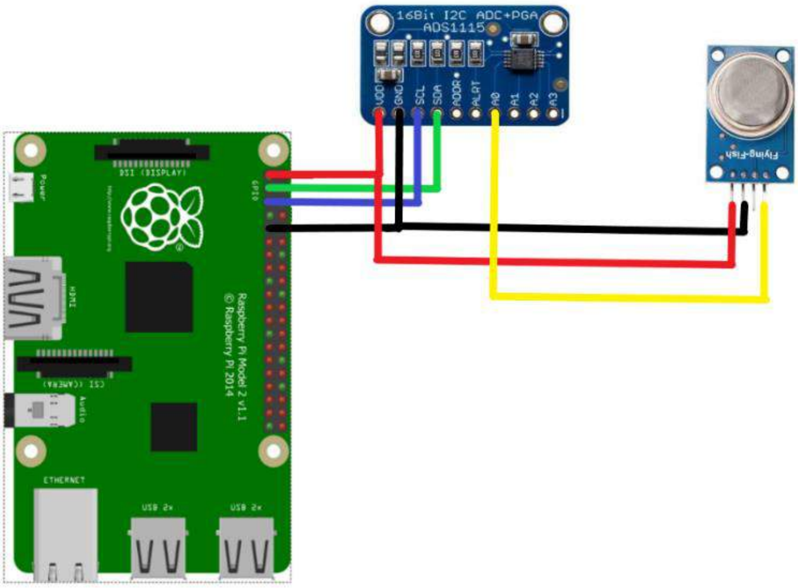


Figure 5.4 Interfacing MQ135 Gas sensor with Raspberry Pi using ADS1115

5.2.5 Interfacing Capacitive Soil Moisture sensor with Raspberry Pi using ADS1115

A capacitive soil moisture sensor works by measuring the changes in capacitance caused by the changes in the dielectric. Capacitive measuring basically measures the dielectric that is formed by the soil and the water is the most important factor that affects the dielectric. Capacitive measuring has some advantages, it not only avoids corrosion of the probe but also gives a better reading of the moisture content of the soil as opposed to using a resistive soil moisture sensor. The capacitance of the sensor is measured by means of a 555 based circuit that produces a voltage proportional to the capacitor inserted in the soil. We then measure this voltage by use of an Analog to Digital Converter which produces a number that we can then interpret as soil moisture.

The ADS1115 is an analog to digital converters that are easy to use with the Raspberry Pi using its I2C communication bus. The ADS1115 is a 12-bit ADC and is a higher precision 16-bit ADC with 4 channels. In addition, it has a programmable gain from 2/3x to 16x in order to amplify small signals and read them with higher precision. The ADS1115 uses I2C communication protocol to read analog values. Before wiring the ADC to the Pi, I2C has to be enabled on the Raspberry Pi using raspi-config.

The equipment required to connect the capacitive soil moisture sensor to Raspberry Pi as shown in Figure 5.5 are:

- Raspberry Pi
- Micro SD card
- Breadboard
- Capacitive Soil Moisture Sensor
- ADS1115 Analog-to-Digital Converter

The Capacitive soil moisture sensor is connected to the Raspberry Pi as follows:

- Sensor VDD to Raspberry Pi 3.3V
- Sensor GND to Raspberry Pi GND
- Sensor A0 (Analog Out) to ADS1115 A1 (Analog Input 1)
- ADS1115 VDD to Raspberry Pi 3.3V
- ADS1115 GND to Raspberry Pi GND
- ADS1115 SCL to Raspberry Pi SCL, Pin 5 (GPIO3)
- ADS1115 SDA to Raspberry Pi SDA, Pin 3 (GPIO2)

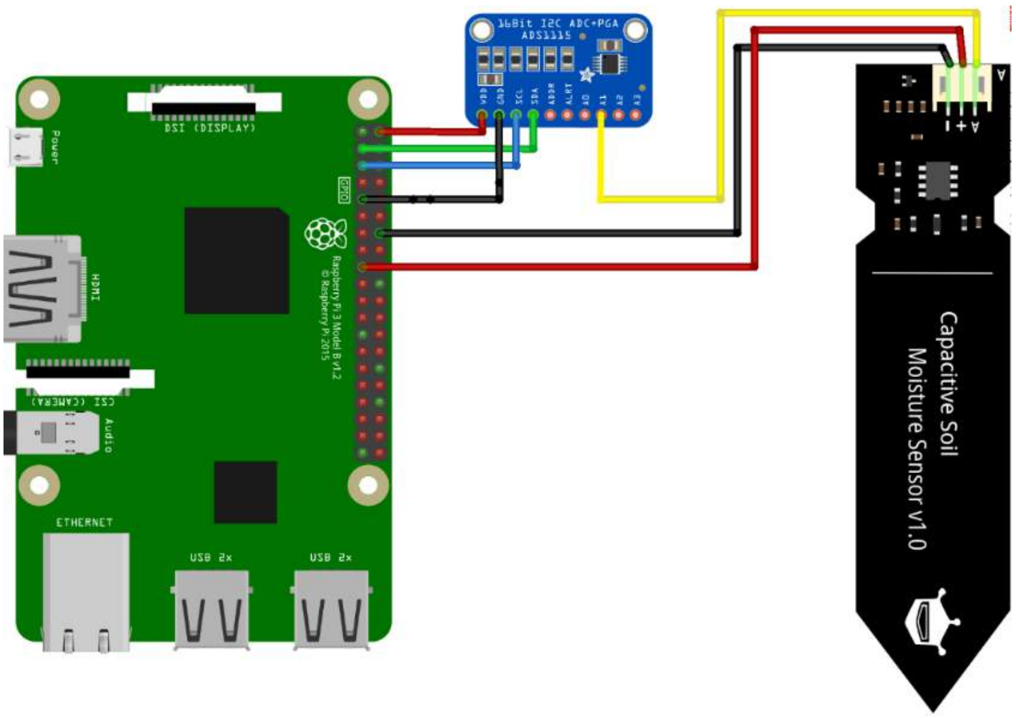


Figure 5.5 Interfacing Capacitive Soil Moisture sensor with Raspberry Pi using ADS1115

5.2.6 Interfacing GUVA-S12SD UV sensor with Raspberry Pi using ADS1115

The GUVA-S12SD UV Sensor chip is suitable for detecting UV radiation in sunlight. It can be used in any application where we want to monitor for the amount of UV light and is simple to connect to any microcontroller. The module, with a typical UV detection wavelength of 200 – 370nm, outputs a calibrated analog voltage which varies with the UV light intensity. So, basically all we need to do is connect this to an ADC input and read in the value. This value ties in with the UV index.

The ADS1115 is an analog to digital converters that are easy to use with the Raspberry Pi using its I2C communication bus. The ADS1115 is a 12-bit ADC and is a higher precision 16-bit ADC with 4 channels. In addition, it has a programmable gain from 2/3x to 16x in order to amplify small signals and read them with higher precision. The ADS1115 uses I2C communication protocol to read analog values. Before wiring the ADC to the Pi, I2C has to be enabled on the Raspberry Pi using raspi-config.

The equipment required to connect the GUVA-S12SD sensor to Raspberry Pi as shown in Figure 5.6 are:

- Raspberry Pi
- Micro SD card
- Breadboard
- GUVA-S12SD UV Sensor
- ADS1115 Analog-to-Digital Converter

The GUVA-S12SD UV sensor is connected to the Raspberry Pi as follows:

- Sensor VDD to Raspberry Pi 3.3V
- Sensor GND to Raspberry Pi GND
- Sensor SIG (Signal) to ADS1115 A2 (Analog Input 2)
- ADS1115 VDD to Raspberry Pi 3.3V
- ADS1115 GND to Raspberry Pi GND
- ADS1115 SCL to Raspberry Pi SCL, Pin 5 (GPIO3)
- ADS1115 SDA to Raspberry Pi SDA, Pin 3 (GPIO2)

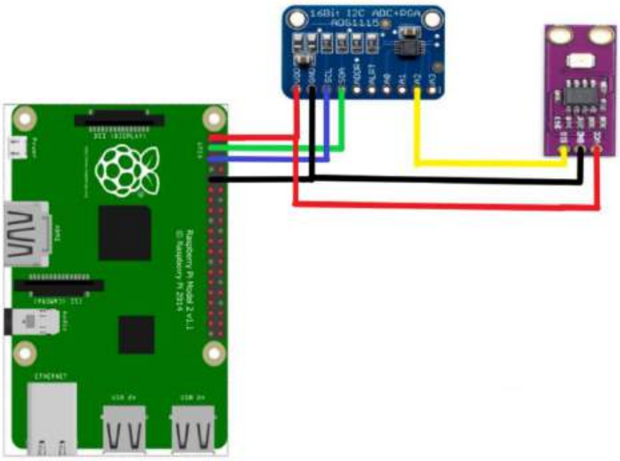


Figure 5.6 Interfacing GUVA-S12SD UV sensor with Raspberry Pi using ADS1115

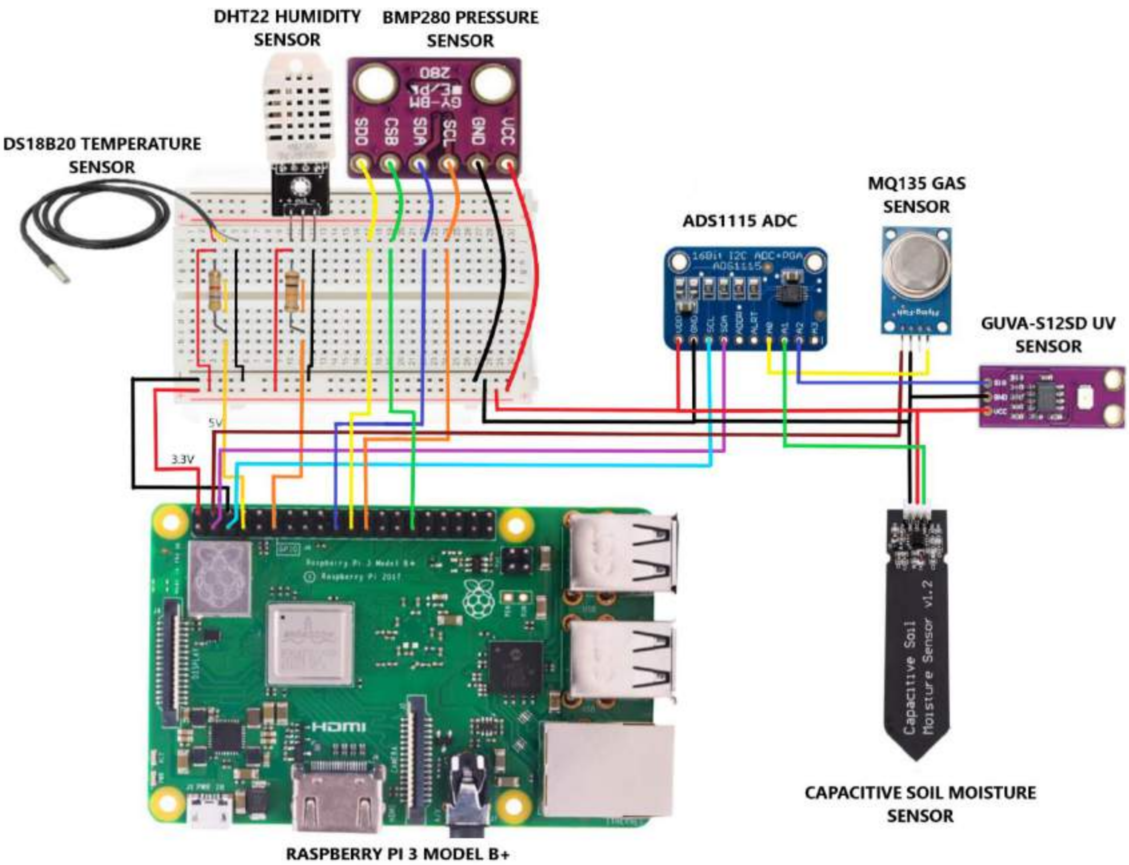


Figure 5.7 Interfacing all sensors with Raspberry Pi

Figure 5.7 represents interfacing all six sensors with Raspberry Pi and connection of hardware components required to design the micro weather station.

SOFTWARE IMPLEMENTATION

CHAPTER 6

SOFTWARE IMPLEMENTATION

6.1 Introduction

This section briefs about the programmatic design, source code editing or programming part for interfacing Raspberry pi with sensors and displaying the sensor data in a web application. It also includes how the system programming design should be built, ensuring that system is operational and it meets quality standard. This series of tasks represents how software procedures, algorithms, or graphical models are produced.

6.2 Algorithm and flowchart for interfacing sensors and Raspberry Pi

Algorithm and flowchart for interfacing all six sensors with Raspberry Pi and software implementation required to design the micro weather station.

6.2.1 Interfacing DS18B20 sensor with Raspberry Pi

- ❖ Since the DS18B20 sensor communicates through 1-Wire method, enable the one wire communication on Raspberry Pi. The basis of 1-Wire technology is a serial protocol using a single data line plus ground reference for communication.
- ❖ To enable 1-Wire in Raspberry Pi, in the Command prompt sudo nano /boot/config.txt command is used to open the config file. Inside the config file line dtoverlay=w1-gpio is added and file is saved.
- ❖ The DS18B20 sensor senses the temperature and gives a raw value which is then converted to a readable value dividing by 1000.
- ❖ The obtained temperature value is sent to the cloud storage by using write API command as shown in Figure 6.1.
- ❖ After a certain time interval, the sensor again senses the temperature and sends the data to a cloud storage and this process repeats continuously.

Flowchart

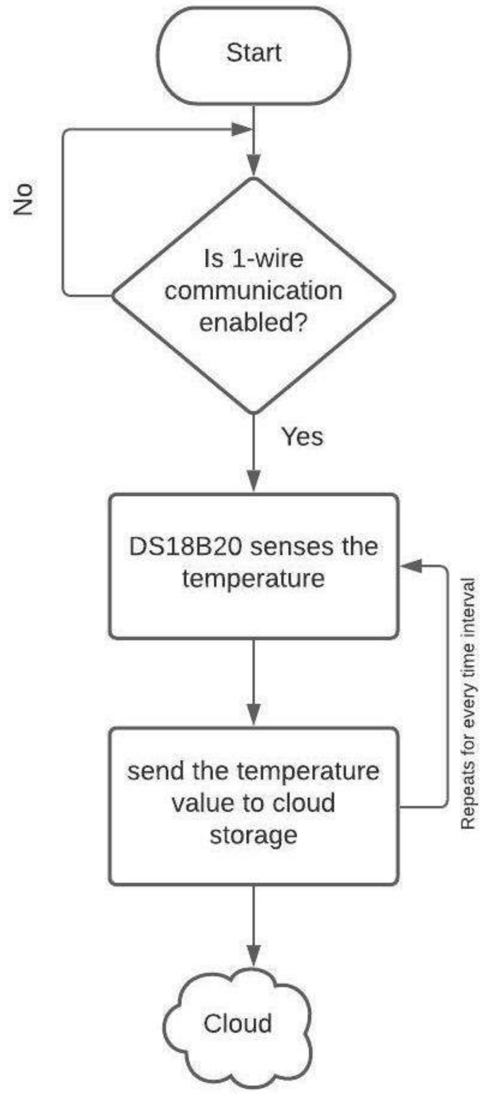


Figure 6.1 Flowchart for DS18B20 sensor Interfaced with Raspberry Pi

6.2.2 Interfacing DHT22 sensor with Raspberry Pi

- To interact or communicate with DHT22 sensor “Adafruit_DHT” library is used. The library easily retrieves the temperature and humidity from the sensor with a few lines of python code as shown in Figure 6.2.
- The “read_retry” function is used to retrieve the data continuously from the sensor.
- The obtained humidity value is sent to the cloud storage by using write API command.
- After a certain time interval, the sensor again senses the humidity and sends the data to a cloud storage and this process repeats continuously.

Flowchart

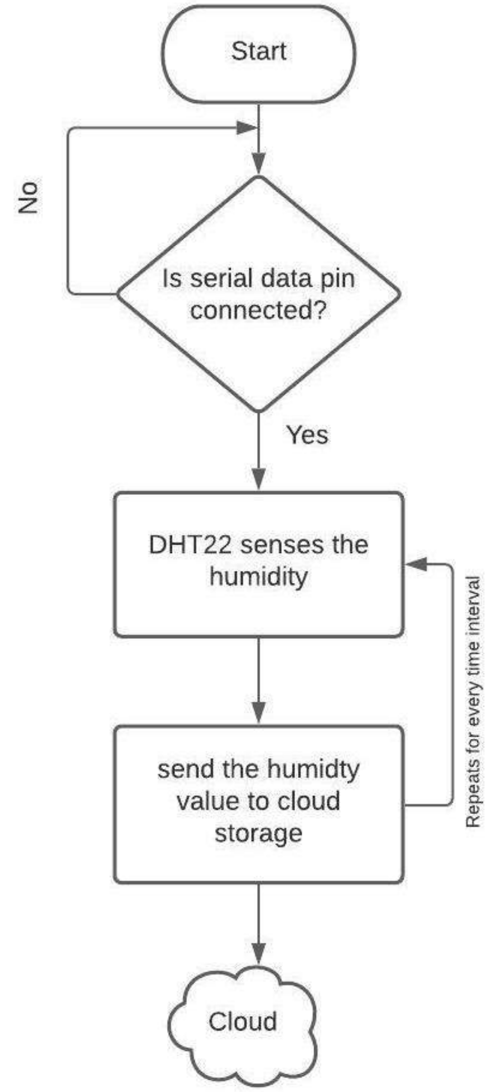


Figure 6.2 Flowchart for DHT22 sensor Interfaced with Raspberry Pi

6.2.3 Interfacing BMP280 sensor with Raspberry Pi

- In order to use BMP280 module, SPI (Serial Peripheral Interface) protocol should be enabled on the Raspberry Pi with the help of SDA (Serial Data) and SCL (Serial Clock) lines as it is not enabled by default as shown in Figure 6.3.
- SDA line is used for the master and slave to send and receive data and SCL line carries the clock signal.
- The obtained pressure value is sent to the cloud storage by using write API command.

- ❖ After a certain time interval, the sensor again senses the pressure and sends the data to a cloud storage and this process repeats continuously.

Flowchart

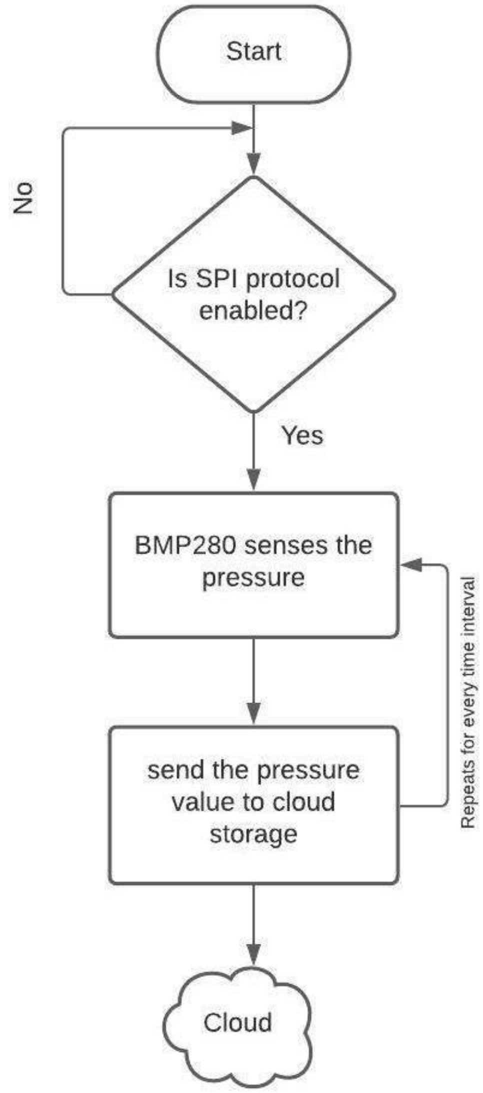


Figure 6.3 Flowchart for BMP280 sensor Interfaced with Raspberry Pi

6.2.4 Interfacing MQ135 sensor with Raspberry Pi using ADS1115 (ADC)

- ❖ Since raspberry pi does not have an ADC, external ADC is used to interface MQ135 with raspberry pi with the help of I2C protocol as shown in Figure 6.4.
- ❖ MQ135 provides analog value of air quality, so it is converted to a readable value which is Air Quality Index (AQI) with help of suitable equations.
- ❖ The obtained AQI value is sent to the cloud storage by using write API command.

- After a certain time interval, the sensor again senses the air quality and sends the data to a cloud storage and this process repeats continuously.

Flowchart

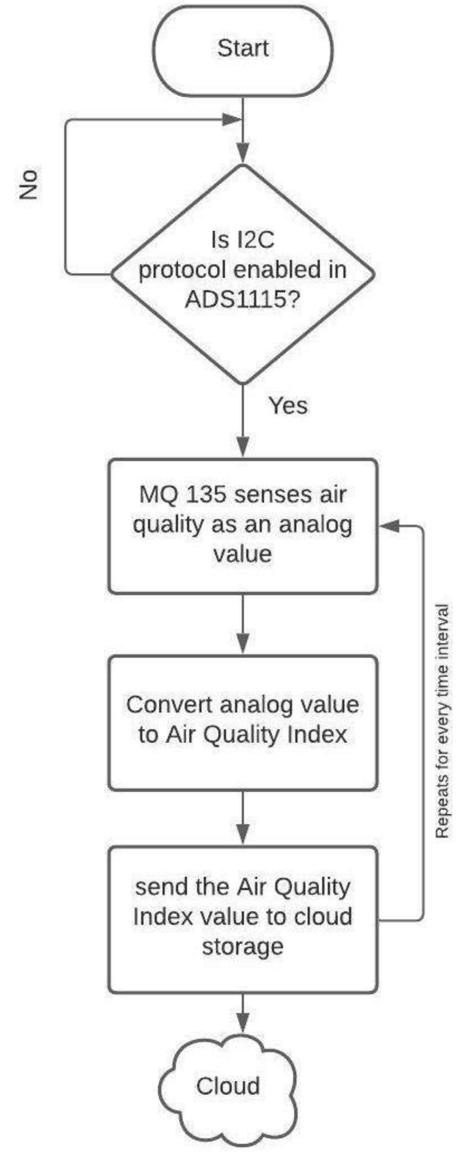


Figure 6.4 Flowchart for MQ135 sensor Interfaced with Raspberry pi using ADS1115

6.2.5 Interfacing Capacitive soil moisture sensor with Raspberry Pi using ADS1115

- Capacitive soil moisture sensor requires analog output pin, so it is interfaced with raspberry pi using ADS1115 which is an ADC as shown in Figure 6.5.

- Since the output analog value has a large range, it has to be mapped to a smaller range and convert value as a percentage.
- The obtained moisture value is sent to the cloud storage by using write API command.
- After a certain time interval, the sensor again senses the soil moisture and sends the data to a cloud storage and this process repeats continuously.

Flowchart

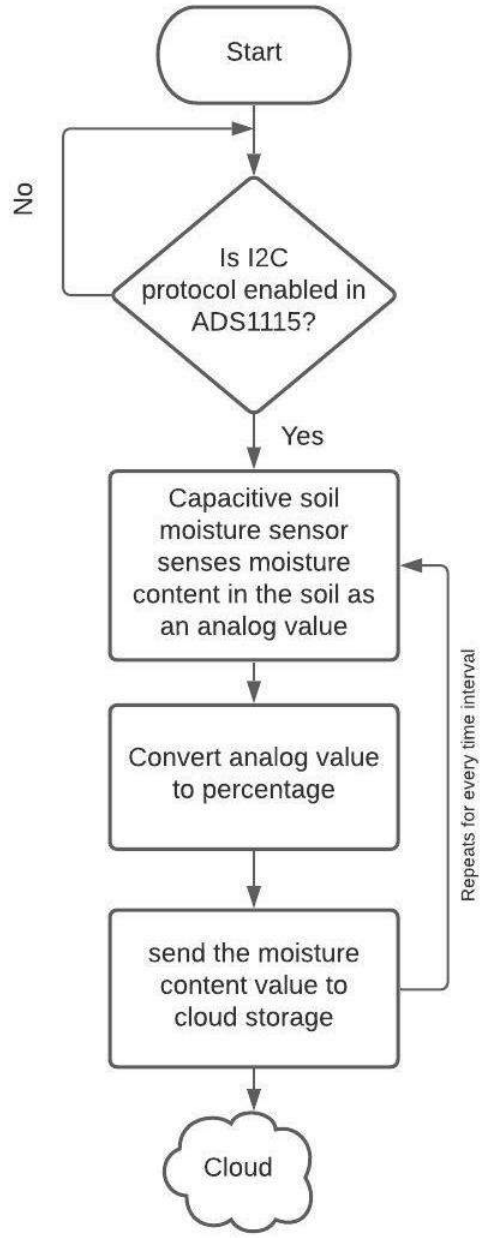


Figure 6.5 Flowchart for Capacitive soil moisture sensor Interfaced with Raspberry Pi using ADS1115

6.2.6 Interfacing GUVA-S12SD sensor with Raspberry Pi using ADS1115

- Since UV sensor gives analog output as voltage value, it is mapped to a readable range (0 to 11) and then this value is sent to the cloud storage by using write API command.
- After a certain time interval, the sensor again senses the ultraviolet radiation and sends the data to a cloud storage and this process repeats continuously as shown in Figure 6.6.

Flowchart

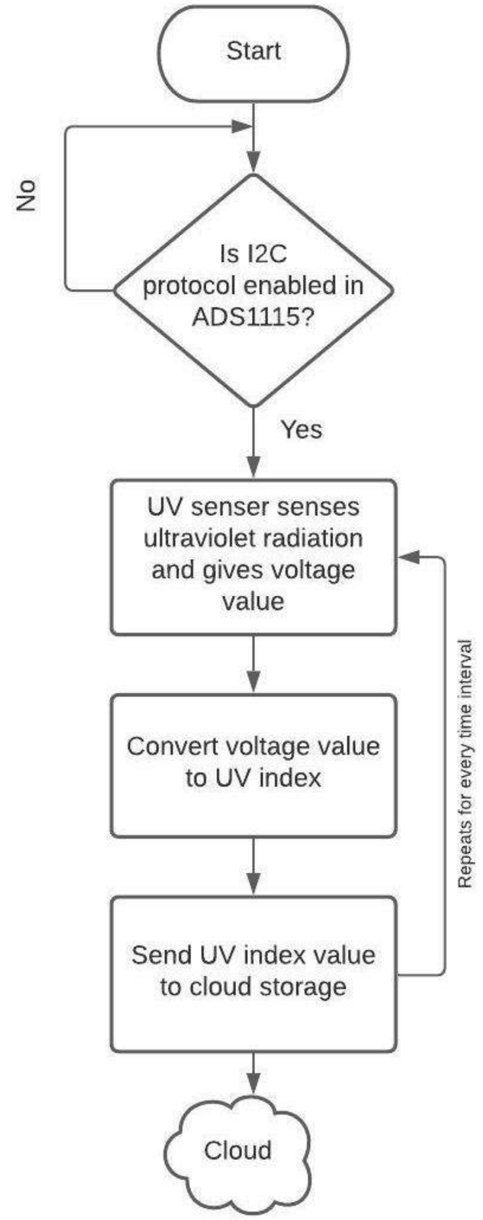


Figure 6.6 Flowchart for GUVA-S12SD sensor Interfaced with Raspberry Pi using ADS1115

6.3 Algorithm and flowchart to show data access in cloud storage

- ❖ The weather parameters captured from sensors are stored in a ThingSpeak cloud. The cloud includes data from node 1, node 2 and average of both the nodes as shown in Figure 6.7.
- ❖ Each node has different channel in the cloud to store data and this data can be accessed by using read API call with the help of unique channel ID. Here each channel includes six fields which contain different sensor values and also a location field which contains coordinate value of the node.
- ❖ To get averaged data of the two nodes, MATLAB analysis is used in the ThingSpeak platform which uses MATLAB code to average the data. Here read API call is made for the two nodes to get current data and then the values are averaged and sent to a new channel so that averaged data can be accessed easily.
- ❖ To display the sensor values in a user interface, web application is built with the help of Expo platform which is an open-source platform for making universal native apps for Android, iOS, and the web with JavaScript and React.
- ❖ Since ThingSpeak platform contains plots of real-time sensor values, the plot can be displayed in the web application by using iframe tag of the plot. An IFrame (Inline Frame) is an HTML document embedded inside another HTML document on a website. The IFrame HTML element is often used to insert content from another source.
- ❖ The location (latitude and longitude) of the nodes is displayed in the web application by using react-leaflet which is a light-weight, open-source mapping library that utilizes OpenStreetMap. And to get the complete address from latitude and longitude coordinates TomTom Reverse Geocoding API is used.
- ❖ In the user interface by default node 1 data and plots will be displayed and also there is an option to view data of the individual nodes. The data from individual nodes as well as averaged data can be downloaded in csv format. The excel sheet contains columns such as Time (IST) at which the data has been pushed to the cloud and sensor values such as Temperature (°c), Humidity (%), Pressure (hPa), Soil moisture (%), UV index and Air quality index (ppm).

Flowchart

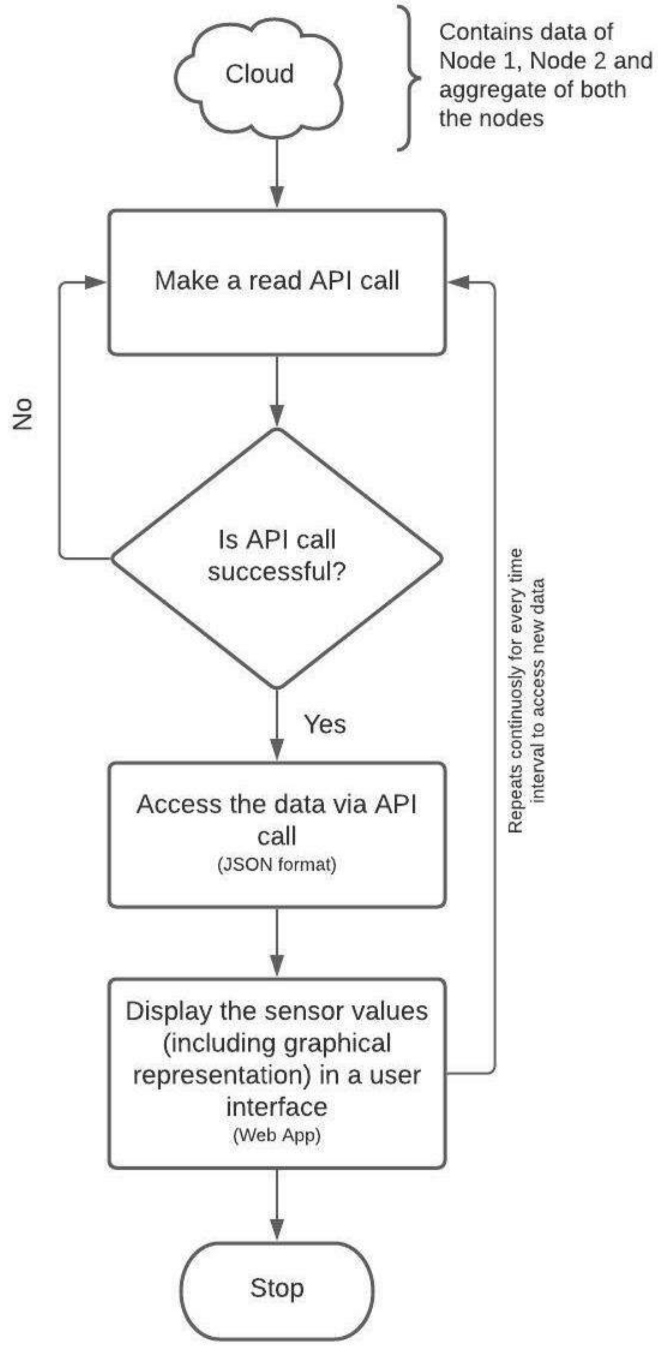


Figure 6.7 Flowchart to show data access in cloud storage

EXPERIMENTAL RESULTS

CHAPTER 7

EXPERIMENTAL RESULTS

7.1 Introduction

In this chapter the results are discussed for Micro Weather Station. Initially programs are dumped to the Raspberry Pi using the python IDE to execute the tasks. The detailed results are discussed in this chapter.

Step 1:

Dumping code onto the Raspberry Pi is done via Python IDE installed in Raspbian OS. The dumping of code is shown in Figure 7.1.

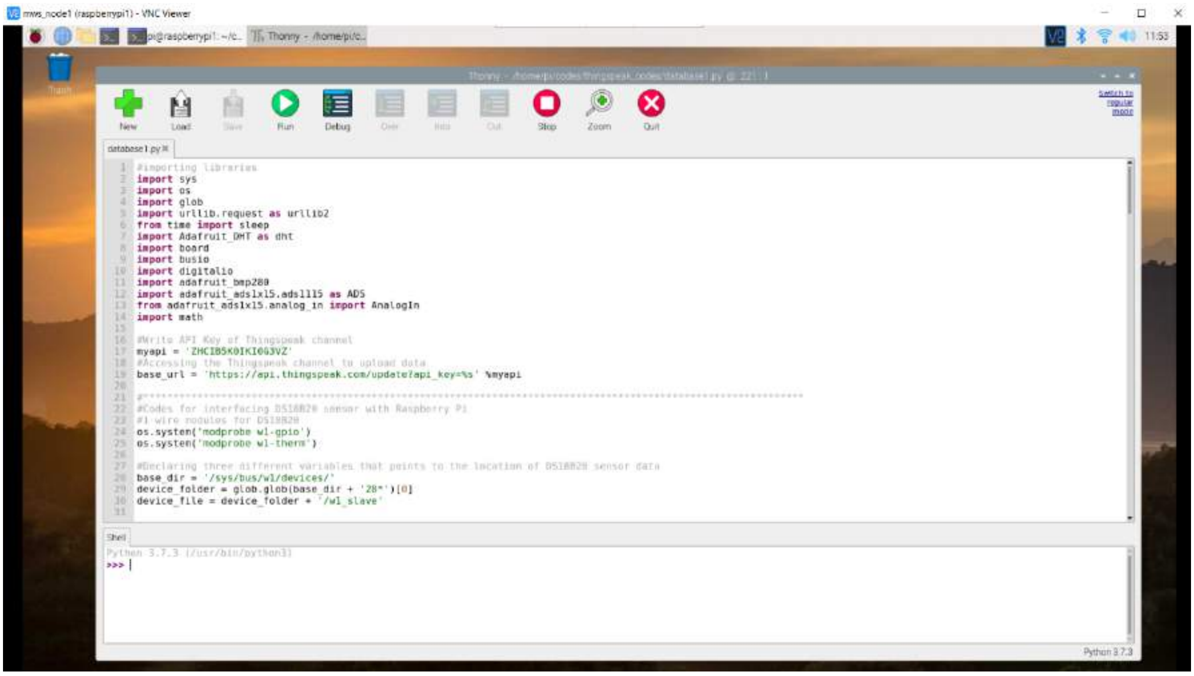


Figure 7.1 Dumping code to Raspberry Pi

Step 2:

The Raspberry Pi is powered by a 5v adaptor which in turn powers up all the sensors and then Raspberry Pi automatically connects to Wireless Fidelity to start capturing weather parameters with the help of sensors. This setup is shown in the Figure 7.2.

Figure 7.2 MWS setup

Step 3:

The captured data from the sensors are pushed to the cloud via a write API call to the ThingSpeak channel. Uploading the sensor data is shown in Figure 7.3.

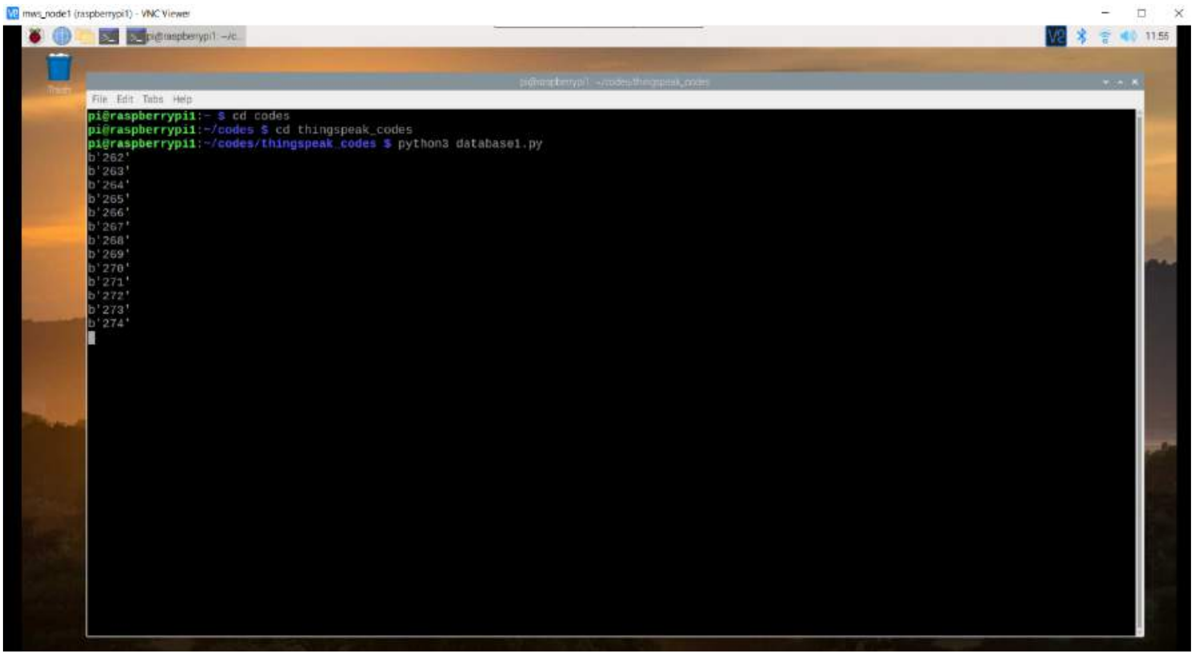


Figure 7.3 Uploading the sensed data in ThingSpeak

Step 4:

The real-time data is visualised and stored in the ThingSpeak cloud platform. Graphical representation of real-time sensor data in the ThingSpeak channel is shown in Figure 7.4.

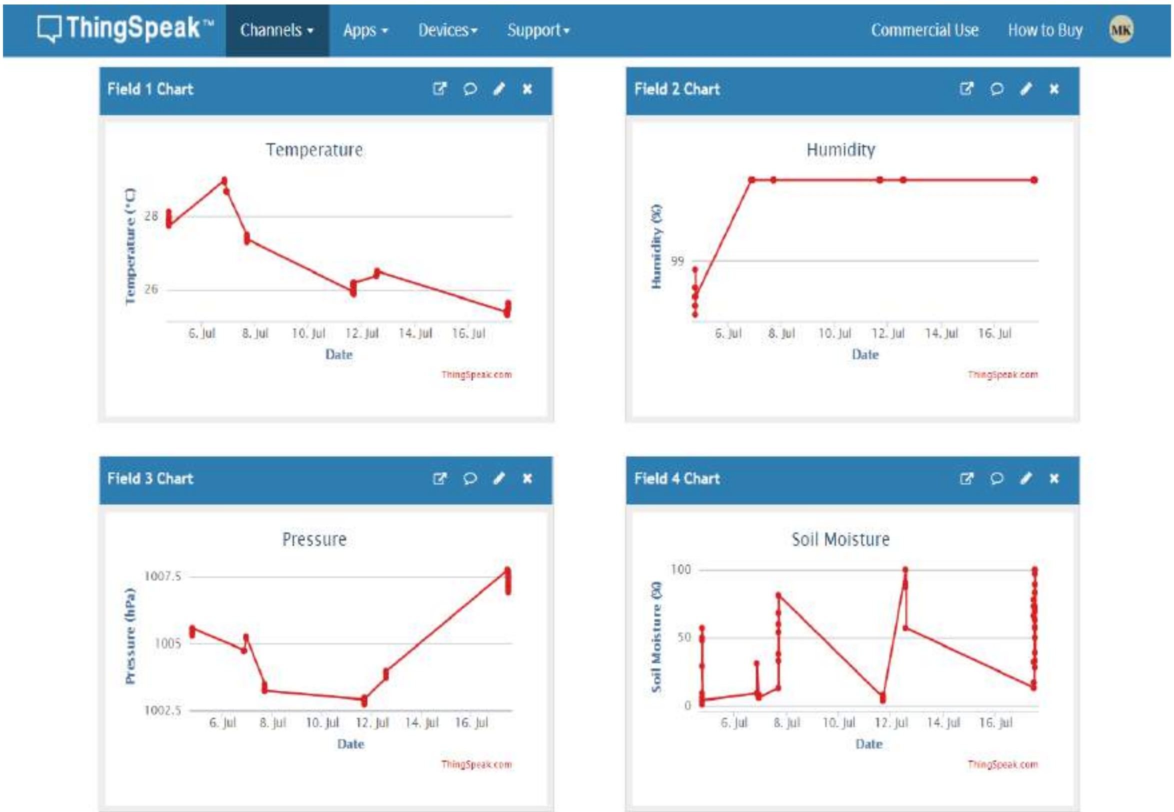
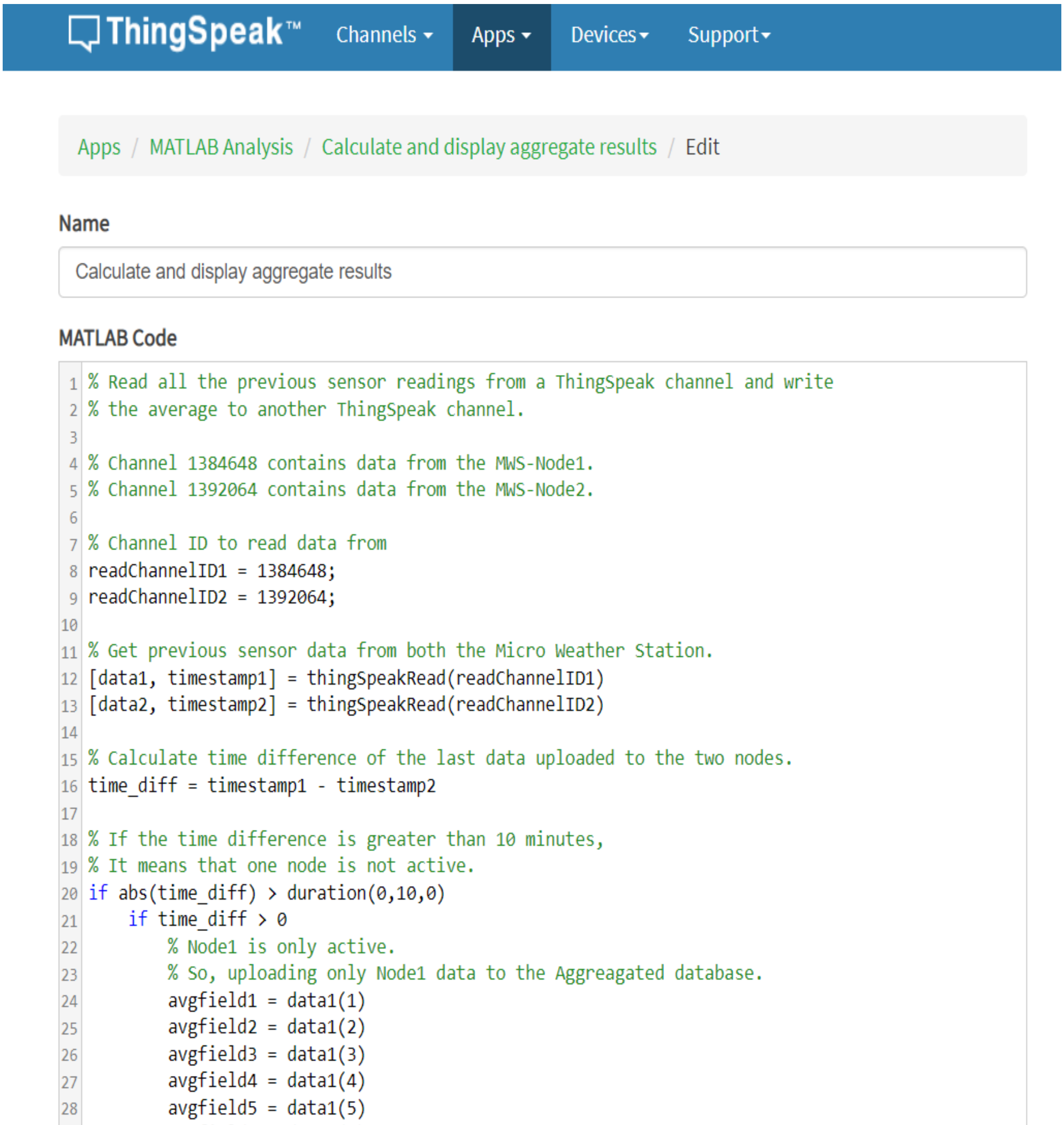


Figure 7.4 Graphical representation of real-time data

Step 5:

Aggregating of real-time data is done with the help of MATLAB Analysis present in the ThingSpeak platform. Aggregation of data is done when either of the nodes (Node1 or Node2) gets triggered i.e., when Node1 or Node2 captures the sensor data, the data aggregation is performed on the latest values captured by both the nodes. Finally, the real-time aggregated data is pushed into another ThingSpeak channel. MATLAB Analysis is shown in Figure 7.5.




```
% Read all the previous sensor readings from a ThingSpeak channel and write
% the average to another ThingSpeak channel.

% Channel 1384648 contains data from the MWS-Node1.
% Channel 1392064 contains data from the MWS-Node2.

% Channel ID to read data from
readChannelID1 = 1384648;
readChannelID2 = 1392064;

% Get previous sensor data from both the Micro Weather Station.
[data1, timestamp1] = thingSpeakRead(readChannelID1)
[data2, timestamp2] = thingSpeakRead(readChannelID2)

% Calculate time difference of the last data uploaded to the two nodes.
time_diff = timestamp1 - timestamp2

% If the time difference is greater than 10 minutes,
% It means that one node is not active.
if abs(time_diff) > duration(0,10,0)
    if time_diff > 0
        % Node1 is only active.
        % So, uploading only Node1 data to the Aggreagated database.
        avgfield1 = data1(1)
        avgfield2 = data1(2)
        avgfield3 = data1(3)
        avgfield4 = data1(4)
        avgfield5 = data1(5)
```

Figure 7.5 MATLAB analysis code for Aggregating the data

Step 6:

· To display the sensor values in a user interface, a web application is built with the help of Expo platform. Since the ThingSpeak platform contains plots of real-time sensor values, the plot can be displayed in the web application by using the iframe tag of the plot as shown in Figure 7.6.

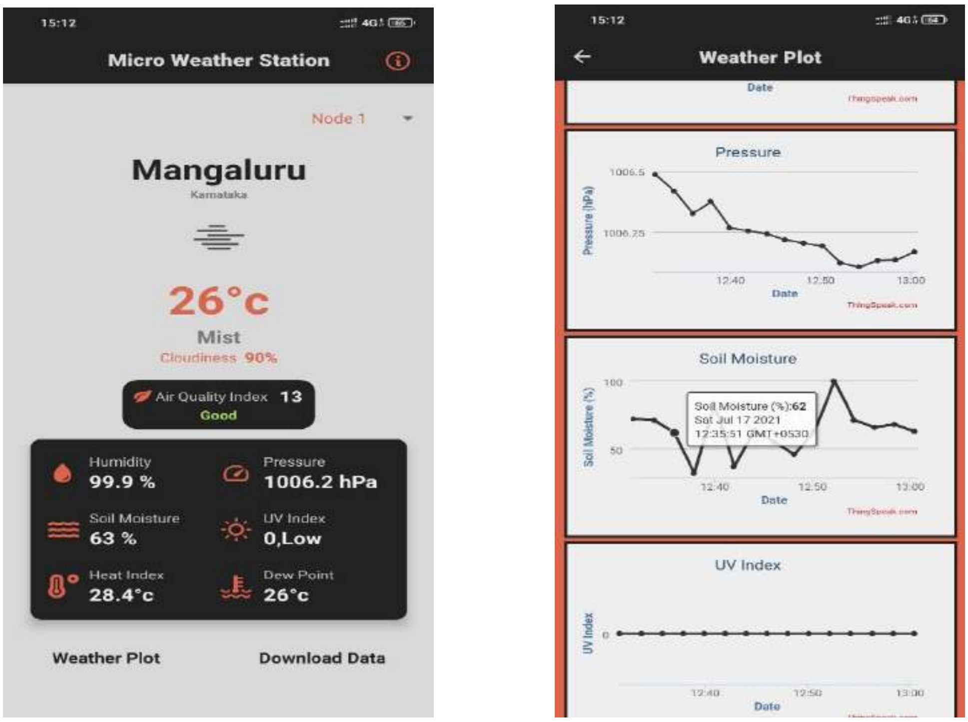


Figure 7.6 User interface showing the sensor data in Web app

Step 7:

The location (latitude and longitude) of the nodes is displayed in the web application by using react-leaflet as shown in Figure 7.7. And to get the complete address from latitude and longitude coordinates TomTom Reverse Geocoding API is used. The data from individual nodes as well as averaged data can be downloaded in csv format as shown in Figure 7.8.

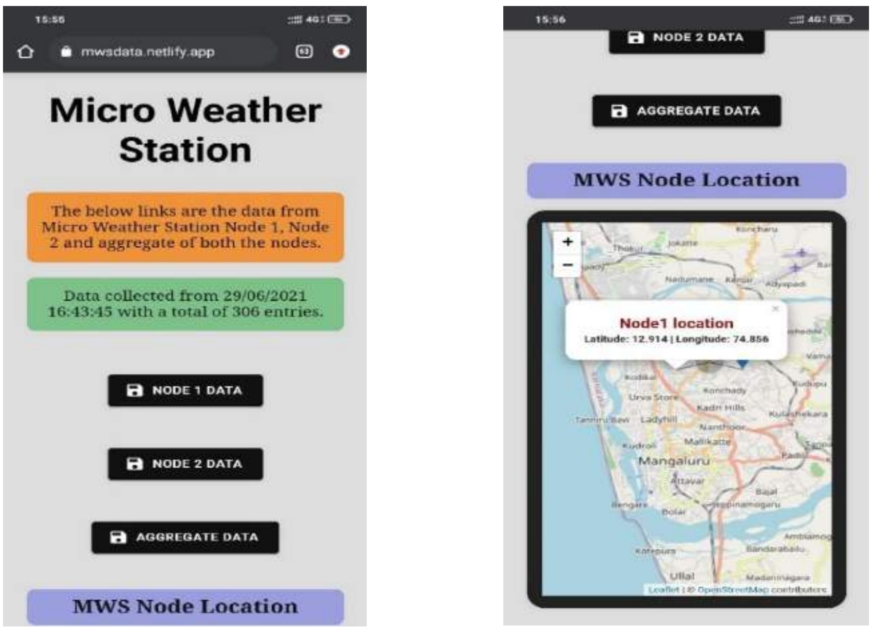


Figure 7.7 User interface showing option to download data and location of the node

	A	B	C	D	E	F	G	H	I
1	Time (DD/MM/YY HH:mm:ss)	Temperature (°c)	Humidity (%)	Pressure (hPa)	Soil Moisture (%)	UV Index	AQI (ppm)	Heat Index (°c)	Dew Point (°c)
2	29-06-2021 16:43	30.062	91.2	1004.655	75	3	2.212521	41.356	28.467
3	29-06-2021 16:45	29.937	91.1	1004.679	50	3	2.518836	40.836	28.324
4	29-06-2021 16:47	29.75	91.1	1004.682	100	2	2.599738	40.119	28.139
5	29-06-2021 16:49	30	91.3	1004.758	100	11	2.599738	41.147	28.424
6	29-06-2021 16:51	30.062	91.3	1004.602	100	3	2.682412	41.39	28.485
7	29-06-2021 16:54	29.937	90.6	1004.758	100	7	2.76688	40.674	28.23
8	29-06-2021 16:56	30.187	90.5	1004.611	100	11	2.682412	41.607	28.457
9	29-06-2021 16:58	30.187	89.6	1004.812	100	6	2.599738	41.299	28.285
10	29-06-2021 17:00	29.937	89.3	1004.831	21	4	2.76688	40.256	27.981
11	29-06-2021 17:02	29.937	86	1004.71	0	3	2.682412	39.228	27.337
12	29-06-2021 17:04	29.875	84.8	1004.647	100	3	2.682412	38.656	27.036
13	29-06-2021 17:06	29.875	86.4	1004.661	100	3	2.682412	39.133	27.355
14	29-06-2021 17:08	30.187	88.7	1004.625	99	3	2.599738	40.996	28.112
15	29-06-2021 17:10	30.375	90.1	1004.725	73	3	2.599738	42.202	28.567
16	29-06-2021 17:12	30.25	90.3	1004.757	29	2	2.682412	41.783	28.482
17	29-06-2021 17:14	30.062	90.8	1004.766	56	2	2.682412	41.222	28.391
18	29-06-2021 17:16	29.812	91.1	1004.706	100	2	2.682412	40.355	28.201
19	29-06-2021 17:18	29.875	90.5	1004.731	3	2	2.682412	40.405	28.149

MWSAgg

Figure 7.8 Download sensor data in CSV format

CONCLUSION AND FUTURE SCOPE

CHAPTER 8

CONCLUSION AND FUTURE SCOPE

8.1 Conclusion

Micro weather station is a system which measures the environmental variables and it is transferred to server where the computations are made and displays the result in web page or mobile applications. The sole purpose of micro weather station is to gather/collect major weather attributes like temperature, humidity, soil moisture, Ultra-Violet (UV) radiation, air pressure and air quality at remote location with most effective and cost-efficient manner with the help of micro weather station equipment.

The design of using multiple nodes with interfacing low-cost sensors can significantly bring down the cost, since using a single large expensive weather station, a multi node system can be used which are interconnected to each other to achieve the same performance. In this system, each node with multiple sensors interfaced provides the full fetched functionality of collecting weather parameters similar to that of a high-end weather station. The data collected independently from different nodes are aggregated in the webserver and finally displayed through the user interface (webapp) in real-time which helps the user in understanding the pattern of the data collected and make further decisions accordingly. The collected data can also be downloaded which helps the student and researchers to study over data collected over time.

8.2 Future scope

In the future, the proposed system can be made more useful by adding additional capabilities that can be involved. This project in future may be more desirable by adding following features:

- Weather prediction algorithm can be built by using the data collected over time.
- Weather forecasting can be done in the later stages of the work.
- Combination of Solar panel and Battery with the system can make the system portable.

- The sensors can be added or removed based on the application.
- It can also be used in Green houses and Laboratories where continuous observations are to be made.

REFERENCES

REFERENCES

[01] Kurella Vijayalakshimi, G.V. Maha Lakshmi, “Real Time Weather Monitoring from Remote Location Using Raspberry pi”, International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol. 5, No. 12, Dec 2016.

[02] Devarakonda Uma, K. Raja Kumar, “Design of Weather Monitoring System using Raspberry Pi and Arduino”, IJATIR, Vol. 08, No. 24, Pages: 4633 4638, December-2016.

[03] Kavya Ladi, A V S N Manoj, G V N Deepak, “IOT Based Weather Reporting System to Find Dynamic Climatic Parameters”, International Conference on Energy, Communication, Data Analytics and Soft Computing, IEEE, 2017.

[04] Damini H Patel, Monali Dave, “IOT Compatible Wireless Smart Portable Mini Weather Analyzer”, International Research Journal of Engineering and Technology (IRJET), Vol. 4, No. 5, May 2017.

[05] S. D. Shewale, S.N. Gaikwad, “An IoT Based Real-Time Weather Monitoring System Using Raspberry Pi”, International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol. 6, Issue 6, June 2017.

[06] Medilla Kusriyanto, Agusti Anggara Putra, “Weather Station Design Using IoT Platform Based on Arduino Mega”, IEEE, 2018.

[07] A. Sravani, Ch. Sudhakar, N. Thirupathi Rao, Debnath Bhattacharyya, Tai-Hoon Kim,".A Concrete Weather Reporting System Model using IoT", International Journal of Control and Automation, Vol. 11, No. 4 (2018), pp.11-20.

[08] Martin Vestenicky, Slavomir Matuska, Robert Hudec, Patrik Kamencay, “Sensor network proposal based on IoT for a prediction system of the power output from photovoltaic panels”, IEEE, 2018.

[09] Girija C, Harshalatha H, Andreanna Grace Shires, Pushpalatha H P, "Internet of Things (IOT) based Weather Monitoring System", International Journal of Engineering Research & Technology (IJERT), 2018.

[10] P Y Muck, M J Homam, "IoT Based Weather Station Using Raspberry Pi 3", International Journal of Engineering & Technology, 2018.

[11] Deekshath, P. Dharanya, K. R. Dimpil Kabadia, G. Deepak Dinakaran, S. Shanthini, "IoT Based Environmental Monitoring System using Arduino UNO and Thingspeak", International Journal of Science Technology & Engineering (IJSTE), Vol. 4, No. 9, March 2018.

[12] Nilesh Patil, Yash Sawant, Tanveet Kaur, Siddhi Kate, Chintan Wagh, "IOT Based Weather Monitoring for Agriculture", International Journal on Recent and Innovation Trends in Computing and Communication, Vol. 6, No. 4, April 2018.

[13] Gaurav A. Bedge, Vijay M. Purohit, "IOT BASED WEATHER MONITORING SYSTEM USING MQTT", International Conference on Smart Systems (ICSS-2018), School of Engineering, Ajeenkya DY Patil University, Pune, India, 27th - 28th September, 2018.

[14] Yashaswi Rahut, Rimsha Afreen, Divya Kamini, "Smart weather monitoring and real time alert system using IoT", International Research Journal of Engineering and Technology (IRJET), Volume: 05 Issue: 10, Oct 2018.

[15] Shimona Prabhu M, Annapurna D, Shamshirun Ibrahim, Kaja Venkata Sai Shravan, Preethu Raj Roy, Remote Weather Assistant using LoRa Network, an Enhancement of IoT Design", Advancement of Computer Technology and its Applications Volume 2 Issue 2, 2019.

[16] E S Semenov, G S Ivanchenko, A V Kharchenko and R V Kolobanov, "Mobile weather station based on ATmega2560 microprocessor", IOP Conference Series: Materials Science and Engineering, 2019.

[17] O.G.Hastak, Isha Ninave, Manisha Meshram, Minakshi Ramteke, Pallavi Maske, “IoT and Arduino Based Weather Station”, JETIR, Volume 6, Issue 3, March 2019.

[18] Kajal N. Dhawale, Narendra Bawane "IoT Based Smart Agriculture System", IOSRJEN, Vol. 09, Issue 5, May 2019, S (VIII), PP 04-09.

[19] Yung-Chung Tsao, Yin Te Tsai, Yaw-Wen Kuo, Chaokung Hwang, “An Implementation of IoT-Based Weather Monitoring System”, IEEE International Conferences on Ubiquitous Computing & Communications (IUCC) and Data Science and Computational Intelligence (DSCI) and Smart Computing, Networking and Services (Smart CNS), 2019.

[20] Liu Kaiyi, Kang Hengyuan, Meng Huansheng, Zhang Fan, “Design of a New Generation of Weather Radar Intelligent Temperature and Humidity Monitoring System Based on ZigBee”, IEEE, 2019.

[21] R. Kavin, K. Lakshmi, S. Sheeba rani, K. Rameshkumar, “Weather Monitoring System using Internet of Things”, IEEE, 2020.

[22] Gaurav Verma, Pranjul Mittal, Shaista Farheen, “Real Time Weather Prediction System Using IOT and Machine Learning”, IEEE, 2020.

[23] Rama Krishna Mullapudi, Mallika G, Nikitha K, Bhanu Sree M, “Weather Monitoring using AWS Cloud Computing”, Weather Monitoring using AWS Cloud Computing, Vol. 3, No. 2, 2020.

[24] Gaurav Jadhav, Kunal Jadhav, Kavita Nadlamani, “Environment Monitoring System using Raspberry-Pi”, International Research Journal of Engineering and Technology (IRJET), Vol. 3, No. 4, April-20.

[25] Iswanto, Prisma Megantoro, Brahmantya Aji Pramudita,” IoT-based weather station with python user interface for measurement technique of educational purpose”, AIP Conference Proceedings 2296, (2020).

[26] Xingang Guo, Yu Song, “Design of Automatic Weather Station Based on GSM Module”, International Conference on Computer, Mechatronics, Control and Electronic Engineering (CMCE), 2010.

[27] Purnomo Husnul Khotimah, Dikdik Krisnandi, Bambang Sugiarto, “Design and Implementation of Remote Terminal Unit on Mini Monitoring Weather Station Based on Microcontroller”, 6th International Conference on Telecommunication Systems, Services, and Applications, IEEE, 2011.

[28] Ravi Kishore Kodali, Snehashish Mandal, “IoT Based Weather Station”, International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT), IEEE, 2016.

[29] Hardeep Saini, Abhishek Thakur, Satinderpar Ahuja, Nitant Sabharwal, “Arduino based automatic wireless weather station with remote graphical application and alerts”, 3rd International Conference on Signal Processing and Integrated Networks, IEEE, 2016.

[30] Dipak V Sose, Ajij D Sayyad, “Weather Monitoring Station”, Int. Journal of Engineering Research and Application, Vol. 6, Issue 6, (Part -1) June 2016, pp.55-60.

[31] Levin Varghese, Gerard Deepak, Santhanavijayan A, “An IoT Analytics Approach for Weather Forecasting using Raspberry Pi 3 Model B+”, Fifteenth International Conference on Information Processing (ICINPRO), 2019.

[32] K. Sai Nikhilesh, Y. Harish Raaghavendra, P.J. Madhu Soothanan, Resmi R, “Low-cost IoT based weather monitoring system for smart community”, Fourth International Conference on Inventive Systems and Control (ICISC 2020).

[33] Palak Kapoor, Ferdous Ahmed Barbhuiya, “Cloud Based Weather Station using IoT Devices”, IEEE, 2019.

[34] Kiranmai Nandagiri, Jhansi Rani Mettu, “Implementation of Weather Monitoring System”, International Journal of Pure and Applied Mathematics, Volume 118 No. 16, 2018.

[35] Priyanka V Kanaka, Vidyasagar .M, “IOT Based Weather Monitoring System”, International Journal of Innovative Research in Computer and Communication Engineering, Vol.5, Special Issue 4, June 2017.

[36] L. Siri Chandana, A. J. Raja Sekhar, “Weather Monitoring Using Wireless Sensor Networks based on IOT”, IJSRST, Volume 4, Issue 5, 2018.

[37] Yudi Adityawarman, Josef Matondang, “Development of Micro Weather Station Based on Long Range Radio Using Automatic Packet Reporting System Protocol”, International Conference on Information Technology Systems and Innovation (ICITSI), 2018.

[38] G. Govardhan, S. Javeed Hussain and S. A. K. Jilani, “A Smart Gadget to Analyse the Weather Changes Using Sense Hat Sensor and Internet of Things (IoT)”, Indian Journal of Science and Technology, Vol 9(35), September 2016.

[39] Pravin Baste, D. Dighe, “Low-Cost Weather Monitoring Station Using Raspberry Pi”, International Research Journal of Engineering and Technology (IRJET), Volume: 04 Issue: 05, May -2017.

[40] Indrasom Majumdar, Bikram Banerjee, Mutyala Tara Preeth, Malaya Kumar Hota, “Design of Weather Monitoring System and Smart Home Automation”, IEEE, Cornell University Library, 2020.

[41] Rajinder Kumar M. Math, Nagaraj V. Dharwadkar, “IoT Based Low-cost Weather Station and Monitoring System for Precision Agriculture in India”, Second International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud), IEEE Xplore Part Number: CFP18OZV-ART, 2018.

[42] Vinayak Aappasaheb Pujari, M. M. Raste, A. A. Pujari, "Cost Effective Automatic Weather Station-A Review", International Conference on Recent Innovations in Engineering and Management, 2016.

[43] Mircea Popa, Catalin Iapa, "Embedded Weather Station with Remote Wireless Control", 19th Telecommunications forum TELFOR, 2011.

[44] Trifun Savic, Milutin Radonjic, "One Approach to Weather Station Design Based on Raspberry Pi Platform", 23rd Telecommunications forum TELFOR, 2015.

[45] Ferdin Joe John Joseph, "IoT Based Weather Monitoring System for Effective Analytics", International Journal of Engineering and Advanced Technology (IJEAT), Vol. 8, No. 4, April, 2019.

[46] Ayyappadas R, A K Kavitha, S Mary Praveena, R M S Parvathi, "Design and Implementation of Weather Monitoring System using Wireless Communication", International Journal of Advanced Information in Engineering Technology (IJAIET), Vol. 4, No. 5, May 2017.

[47] Ibo Ngebani, Ketshabile Nfanyana, Sajid Mubashir Sheikh, Zibani Ishmael, "Design of a Microcontroller Based Internet of Things Based Weather Station", Ibo Ngebani Journal of Engineering Research and Application, Vol. 8, No. 8, Aug 2018.

[48] Li Feng, Li Yan, Zhou Qing, "Acquisition and application of running state information on automatic weather station", International Conference on Electronic Measurement & Instruments (ICEMI), 2017.

[49] Alpana Kushwaha, Bharathi MalakReddy, "IoT based Low-cost Weather Monitoring System", International Journal of Scientific Research & Engineering Trends, Vol. 4, No. 2, April 2018.

[50] K VivekBabu, K Anudeep Reddy, C M Vidhyapathi, B. Karthikeyan, "Weather Forecasting Using Raspberry Pi with Internet of Things (IOT)", ARPN Journal of Engineering and Applied Sciences, Vol. 12, No. 17, Sept 2017.

[51] E.L. Omoze, T.J. Timiyo, P.E. Orukpe, “Development of an Automatic Weather Monitoring System”, IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE), Vol. 14, No. 6, Dec 2019.

[52] Vijayalaxmi Kadrolli, Karuna Melge, Yogita Kamane, Pranali Gaikwad, Rushikesh Sangle, “Portable Weather Station Using GUI”, International Conference on Computing Methodologies and Communication (ICCMC), 2019.

[53] Sonam Tenzin, Satetha Siyang, Theerapat Pobkrut, Teerakiat Kerdcharoen, “Low-Cost Weather Station for Climate-Smart Agriculture”, IEEE, 2017.

[54] Kazuo Yonekura, Hitoshi Hattori, Taiji Suzuki, “Short-term local weather forecast using dense weather station by deep neural network”, IEEE International Conference on Big Data (Big Data), 2018.

[55] R. Udaya Bharathi, M. Seshashayee,” Weather and Air Pollution real-time Monitoring System using Internet of Things”, International Journal of Innovative Technology and Exploring Engineering (IJITEE), Volume-8 Issue-9, July 2019.

[56] Sourabh Halder, G Sivakumar,” Embedded based Remote Monitoring Station for Live Streaming of Temperature and Humidity”, International Conference on Electrical, Electronics, Communication, Computer and Optimization Techniques (ICEECCOT), 2017.

[57] Md. Imranul Haque, Mokammal Hossain, Abu Hena MD.Shatil, Abu Noman Tusar, Md. Hasanur Rahman,” Renewable Powered Portable Weather Update Station”, International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST), 2019.

[58] Bulipe Srinivas Rao, K. Srinivasa Rao, N. Ome, ” Internet of Things (IOT) Based Weather Monitoring system”, International Journal of Advanced Research in Computer and Communication Engineering, Vol. 5, Issue 9, September 2016.

[59] Majid Al-Kuwari, Abdulrhman Ramadan, Yousef Ismael, Laith Al-Sughair, Adel Gastli," Smart-Home Automation using IoT-based Sensing and Monitoring Platform".

[60] Adnan Shaout, Yulong Li, Mohan Zhou, Selim Awad," Low-Cost Embedded Weather Station with Intelligent System", 2014.

[61] Moohun Lee, Mingyu Kim, "Development of Real-time Road Surface Condition Determination Algorithm Using an Automatic Weather System", IEEE, 2016.

[62] Jian-Ming Li, Lei Han, Shu-Yong Zhen, Lun-Tao Yao, "The Assessment of Automatic Weather Station Operating Quality Based on Fuzzy AHP", Ninth International Conference on Machine Learning and Cybernetics, Qingdao, 11-14 July 2010.

[63] Hao Li, Manesh Kumar Ochani, Hengxu Zhang, Lei Zhang, "Design of micro-automatic weather station for modern power grid based on STM32", 6th International Conference on Renewable Power Generation (RPG), 19–20 October 2017.

[64] Yash Mittal, Anshika Mittal, Diksha Bhateja, Komal Parmaar, V. K. Mittal, "Correlation Among Environmental Parameters using An Online Smart Weather Station System", IEEE, 2015.

[65] L. M. Miu, S. Z. Djokic, "Evaluation of Wind Energy Resources at Unmonitored Micro-Wind Application Sites from Public Weather Station Records", IEEE, 2017.

[66] K. N. V. Satyanarayana, S. R. N. Reddy, K. N. V. Suresh Varma & P. Kanaka Raju, "Mobile App & IoT Based Smart Weather Station", International Journal of Electronics, Communication & Instrumentation Engineering Research and Development (IJECIERD), Vol. 7, Issue 4, Aug 2017, 7-14.

[67] Aris Munandar, Hanif Fakhrurroja, Muhammad Ilham Rizqyawan, Rian Putra Pratama, Jony Winaryo Wibowo, Irfan Asfy Fakhry Anto, “Design of Real-time Weather Monitoring System Based on Mobile Application using Automatic Weather Station”, 2nd International Conference on Automation, Cognitive Science, Optics, Micro Electro-Mechanical System, and Information Technology 6(ICACOMIT), October 23, 2017.

[68] Megatari Wahyu Ningrum Handani, Giva Andriana Mutiara, Devie Ryana Suchendra, “Remote Access Weather Monitoring System Based on Soft Real-Time System (SRTS)”, The International Conference on Control, Electronics, Renewable Energy and Communications (ICCEREC), 2018.

[69] Mary Nsabagwa, Maximus Byamukama, “Network Densification Strategies for Automatic Weather Stations: Challenges and Opportunities for Uganda”, IST-Africa 2016 Conference Proceedings.

[70] Jun Liu, Binbin Zhang, Baolei Lv, “An intelligent framework of Automatic Weather Station”, ICAWS-2017 meeting.

[71] R. M. Weerasinghe, M.S.M Aroos, A.S. Pannila, M.K. Jayanand, D.U.J. Sonnadara, “Construction of an Automated Weather Station for Ground Level Weather Measurements”, Annual Transactions of IESL, 2011.

[72] Anastasia Eleftheriou, Petros Karvelis, Kostas Kouvaris, Chrysostomos Stylios1, Micro Climate Prediction Utilising Machine Learning Approaches”, Micro Climate Prediction Utilising Machine Learning Approaches, IEEE, 2018.

[73] Kondamudi Siva Sai Ram, A.N.P.S.Gupta, “IoT based Data Logger System for weather monitoring using Wireless sensor networks”, International Journal of Engineering Trends and Technology (IJETT), Vol. 32 No. 2, February 2016.

[74] Dian Milchev Iliev, Vassil Nikolov Gourev, Mityo Georgiev Mitev, “Realization of Adaptive Weather Station for Work in Antarctic Conditions, International Scientific Conference Electronics, September 12 - 14, 2016.

[75] Atul Kulkarni, Debajyoti Mukhopadhyay, “Internet of Things Based Weather Forecast Monitoring System”, Indonesian Journal of Electrical Engineering and Computer Science, Vol. 9, No. 3, March 2018.

[76] Purnomo Husnul Khotimah, Devi Munandar, “Data Center for Integrating Weather Monitoring Systems”, International Conference on Computer, Control, Informatics and Its Applications, 2013.

[77] Kleber Kruger, Fábio Iaione, “Development of a fault injection system to test a weather station based on rapid prototyping platform”, IEEE International Conference on High Performance Computing and Communications & IEEE International Conference on Embedded and Ubiquitous Computing, 2013.

[78] Christy Kunjumon, Sreelekshmi S Nair, Deepa Rajan S, L. Padma Suresh, Preetha S L, “Survey on Weather Forecasting Using Data Mining”, IEEE Conference on Emerging Devices and Smart Systems (ICEDSS), 2-3 March 2018.

[79] Andres Lage, Jaun Carlos Correa, “Weather station with cellular communication network”, IEEE, 2015.

[80] Zhen Fang, Zhan Zhao, Lidong. Dul, Jiangang Zhang, Cheng Pang, Daoqu Geng, “A New Portable Micro Weather Station”, 5th IEEE International Conference on Nano/Micro Engineered and Molecular Systems, 2010.

[81] M. Rahaman Laskar, R. Bhattacharjee, M. Saugiri and P. Bhattacharya, “Weather Forecasting using Arduino Based Cube-Sat”, Twelfth International Multi-Conference on Information Processing-2016 (IMCIP-2016).

[82] M. Pratapa Raju, A. Jaya Laxmi, “IOT based Online Load Forecasting using Machine Learning Algorithms”, Third International Conference on Computing and Network Communications (CoCoNet’19), Procedia Computer Science 171 (2020).

[83] Battu Manikumar, Sriram Singla, “Application of Weather Monitoring in Agriculture its Design and Implementation Using IoT”, IJRECE, Vol. 6, Issue 3 July - September 2018.

[84] Saniya Zahoor, "IoT Sensor Based Zone Specific Weather Forecasting and Environmental Monitoring System", International Journal of Latest Trends in Engineering and Technology, Vol. (10), Issue (2), pp. 161-166.

[85] A. Aponte-Roa, Luis Benitez Montalvan, Christopher, Albert A. Espinoza, Luis Feliciano Velazquez, Ricardo Serrano, "Evaluation of a Low-Cost, Solar-Powered Weather Station for Small-Scale Wind Farm Site Selection", IEEE Instrumentation and Measurement Society, 2018.

[86] Chandrathilake, H.T.S. Hewawitharana, R.S. Jayawardana, A.D.D. Viduranga, H.M.N. Dilum Bandara, Suresh Marru, Srinath Perera, "Reducing Computational Time of Closed-Loop Weather Monitoring: A Complex Event Processing and Machine Learning Based Approach", IEEE, 2016.

[87] R.M. Weerasinghe, M.S.M Aroos, A.S. Pannila, M.K. Jayananda and D.U.J. Sonnadara, "Construction of an Automated Weather Station for Ground Level Weather Measurements", Annual Transactions of IESL, 2011.

[88] Masato Yamanouchi, Hideya Ochiai, Y K Reddy, Hiroshi Esaki, Hideki Sunahara, "Case study of constructing weather monitoring system in difficult environment", 2014 IEEE International Conference on Ubiquitous Intelligence and Computing/International Conference on Autonomic and Trusted Computing.

[89] Valentin-Gabriel Voiculescu, Vlad-Alexandru Grosu, Lucian Milea, Paul Schiopu, Orest Oltu, Eduard Franti, "Impact of Frequency Scaling on Power Management of an Autonomous Weather Monitoring Embedded System", IEEE, 2011.

[90] Zheng Ruan, Feng Li, Mei Gao, Wenhua Zhang, Lianshu Jie, "Information Analysis and Dissemination using WebGIS-based Mesoscale Weather Monitoring and Warning", IEEE, 2009.

[91] Wu Xiangjuan, Lv Wenhua, Xing Hongyan, Zou Yingquan, "Research on temperature channel calibration of data-acquisition unit of automatic weather station", Tenth International Conference on Electronic Measurement & Instruments, 2011.

[92] W. Chebbi, M. Benjemaa, A. Kamoun, M. Jabloun, A. Sahli, "Development of a WSN Integrated Weather Station node for an Irrigation Alert Program under Tunisian Conditions", 8th International Multi-Conference on Systems, Signals & Devices, 2011.

[93] Cheng-Tsung Chen, Jenq-Shiou Le, Kuan-Wu Su, Zhe-Yi Zhu, Tung-Hung Chiang, "Design and Implementation of a Mobile Ambient Intelligence Based Mesoscale Weather Forecasting System", IEEE International Conference on Consumer Electronics (ICCE), 2012.

[94] Petros Karvelis, Daniele Mazzei, Matteo Biviano, Chrysostomos Stylios, "PortWeather: A Lightweight Onboard Solution for Real-Time Weather Prediction", Sensors 2020.

[95] Ihsan Mizher Baht, Petre Marian Nicolae, Marian tefan Nicolae, "Impact of Weather Forecasts and Green Building on Micro Grid Energy Management System", IEEE, 2019.

[96] Varad Vishwarupe, Mangesh Bedekar, Saniya Zahoor, "Zone Specific Weather Monitoring System Using Crowdsourcing and Telecom Infrastructure", International Conference on Information Processing (ICIP), 2015.

[97] Cheng Pang, Zhan Zhao, Jiangang Zhang Li Shi, Lidong Du, Zhen Fang and Yonghong Liu, "An Advantageous Fabrication Technology to Integrate Pressure Sensor into Multi-sensor for Micro Weather Station", IEEE, 2009.

[98] U.Ramani, R.Nithya, S.Sathieshkumar, T.Santhoshkumar, "Automatic Weather Monitoring Analysis for Renewable Energy System", International Conference on Electronics and Sustainable Communication Systems (ICESC 2020).

[99] Aris Pujud Kurniawan, Agung Nugroho Jati, Fairuz Azmi, "Weather Prediction Based on Fuzzy Logic Algorithm for Supporting General Farming Automation System", 5th International Conference on Instrumentation, Control, and Automation (ICA), August 9-11, 2017.

[100] Maximus Byamukama , Janet Nakato Nannono , Kabonire Ruhinda , Björn Pehrson, Mary Nsabagwa, Roselyn Akol, Robert Olsson, Geoffrey Bakkabulindi, Emmanuel Kondela, "Design Guidelines for Ultra-low Power Gateways in Environment Monitoring Wireless Sensor Networks", IEEE African 2017 Proceedings.

A Review on Design and Implementation of Micro Weather Station

Vijetha T S
Assitant Professor
Department of Electronics and Communication
Alva's Institute of Engineering and Technology
Mangalore, India

Abhishek M Shastry K
Final year BE student
Department of Electronics and Communication
Alva's Institute of Engineering and Technology
Mangalore, India

Abhishek Mahendrakar
Final year BE student
Department of Electronics and Communication
Alva's Institute of Engineering and Technology
Mangalore, India

Akshay
Final year BE student
Department of Electronics and Communication
Alva's Institute of Engineering and Technology
Mangalore, India

K Muthu
Final year BE student
Department of Electronics and Communication
Alva's Institute of Engineering and Technology
Mangalore, India

***Abstract*: The advancement of Internet of Things (IoT) has made a major impact on technology. It collects an enormous amount of data using which can be used in an application. One such system is Micro Weather Station, which gives us different environment variable values such as temperature, humidity, soil moisture, Ultraviolet (UV) radiation, air pressure, air quality, and rainfall. Initially research or advancements done on weather monitoring were limited, but over the last century it has evolved into a well-organized and professional global activity that reflects its crucial importance for a wide range of economic, environmental, civil protection and farming activities. Due to human activities these days there is a drastic change in the climate, hence an accurate and cost-efficient system is needed which is used to monitor the changes in the environment. The application of weather station is not just bounded for getting the live data and prediction of weather, but also includes the advancement in the agricultural sector and the military applications. Cloud storage technology and Geo-tagging have made it much simpler to get the data of any place at any time.**

Keywords: Raspberry Pi, Sensors, IoT, Accuracy, Real-time monitoring.

I. INTRODUCTION

Micro weather station is a system which measures the environmental variable values and it is transferred to server where the computations are made and displays the result in web page or mobile applications. The sole purpose of the micro weather station is to gather or collect major weather attributes like temperature, humidity, soil moisture, UV radiation, air pressure, and air quality at a remote location in a most effective and cost-efficient manner with the help of micro weather station equipment. This uses multiple micro weather stations and aggregates the data provided by multiple data collection points in real-time. Raspberry Pi is interfaced with various sensors using General-Purpose Input/Output (GPIO) pins. The Raspberry Pi has an inbuilt Secure Digital (SD) card slot that helps to fetch and store data. All the data fetched are time tagged from the Raspbian OS, in addition to this, the system is Geo-tagged which gives the location of the weather station. The data fetched by Raspberry Pi will then be pushed to the web server where the required calculations are made and display the data in terms of table/graph with the help of webpage in real-time. This enables the user an easy and reliable way to understand the data and make further decisions accordingly. The data collected over time can be used for various purposes such as weather prediction, study the change in the weather pattern and research purposes. The supply for the micro weather station system is provided with the help of a battery and a solar panel.

II. LITERATURE SURVEY

Sravani et al [01] proposed a method for recovering ecological circumstances like temperature, humidity, and Carbon Monoxide (CO) level at a precise position and create information accessible anywhere in the world using IoT. The model comprises numerous modest remote sensors, which are equipped for gathering, handling natural data, and speaking with neighboring hubs. Arduino is used as the microcontroller which collects the data from the sensors. A Wireless Fidelity module is used to send these data to the cloud where it'll be stored in the form of a spreadsheet. Using this data different graphs can be plotted for better understanding.

Kajal and Narendra [02] have presented the method of processing a huge amount of data which is collected by the sensors. Plants require very specific conditions for optimal growth and health. So, for monitoring the condition of the crop field DHT-11 sensor is used to get temperature and humidity values. A soil moisture sensor is used to get the moisture content of the soil. The Node Micro Controller Unit is used to get the sensor data. Node Micro Controller Unit (NodeMCU) has an inbuilt Wireless Fidelity, using this the data are uploaded to the cloud. Where it can be used for further computation.

Shewale et al [03] have proposed an idea of real-time monitoring of different environmental parameters using IoT and Raspberry Pi board. The system consists of some digital as well as analog sensors like DHT11 to measure the Temperature and Humidity, BMP180 is used to measure the Pressure and Altitude, Light Dependent Resistor (LDR) is used to measure the Light intensity, and a marked scale with ULN2803 is used to measure environmental parameters. The data are then fed to the ThingSpeak server where the data converted into a graph and also stored in the form of Comma Separated Values (CSV) as well as text files.

Nikhilesh et al [04] have proposed a system where a weather system is installed for a community that can give information to a particular community and warns the people of the community via alerts or emails. The system consists of ESP8266-12E as a NodeMCU which is interfaced with a temperature sensor, LM35, LDR, raindrop sensor, and cup anemometer. The data from LDR is used to turn ON/OFF the street lights. The data can be directly viewed by the user with the display provided else they can subscribe to the network and can see the data using the Blynk application. There is also a desktop version of Blynk for those who want to access the data or by downloading the data in CSV format from the server.

Gaurav et al [05] have implemented a real-time weather prediction system that can be used in several applications like homes, industries, agriculture, stadiums, etc. for predicting the weather information. The system utilizes a temperature and humidity sensor DHT11 and a light intensity sensor LDR. The data from the sensors are uploaded to a ThingSpeak cloud server with help of ESP8266 and the NodeMCU module. The data is also displayed on a webpage for real-time monitoring. The real-time data is used to test the logistic regression model and prediction is done for a particular value to test the accuracy of the system.

Kurella and Lakshmi [06] have presented a model to develop a simple, low-cost real-time remote weather monitoring system with fast and accurate data transfer using the advantages of Raspberry pi and wireless technologies. The System fetches weather conditions continuously using various sensors interfaced with Raspberry pi to measure various weather parameters like temperature, humidity, pressure, gas concentration, light intensity, etc. The system displays the data on a Liquid Crystal Display (LCD) screen for local monitoring and also transfers the data to a web page created for remote monitoring and stores it in a database for further analysis. The web page displays the location of the system on Google maps with the help of Global Positioning System (GPS) coordinate values provided by a raspberry pi.

Devarakonda and Kumar [07] have proposed a Weather Monitoring System. The system continuously monitors the weather conditions and updates in a public cloud. The data can be used in various safety measure applications. It consists of temperature sensor and CO gas sensor. The model consists of two nodes kept at two different places. The data are collected by the Arduino Uno Microcontroller and it is uploaded to the Thingspeak sever with the help of Zigbee Transmitter. A Raspberry Pi interfaced with ZigBee transmitter is used as a base station where the data is extracted from the server and displayed on the LCD Display module.

Bharathi and Seshashayee [08] have proposed a method that monitors and identifies the progressions occurring over the current geographical location and gives adequate approaches to the clients to get to the data from any place through cloud storage. The system consists of a rain meter connected with Arduino UNO which acts as an Analog to Digital Converter (ADC) for the rain meter sensor. Then it is coupled with the Raspberry Pi interfaced with gas sensors such as MQ-6, MQ-7, MQ-135, and also a DHT-11 to get Temperature and Humidity values. The data is uploaded to the Google cloud platform where it can be accessed from anywhere in the world.

Bulipe et al [09] have proposed a system that monitors weather at a particular place and makes the information available for the users. The system deals with monitoring the weather conditions like temperature, humidity, light intensity, and CO level with sensors and sends the information to the web page. The Arduino UNO interfaced with the DHT11, LDR sensor, and CO gas sensor receives the data and uploads it to the google cloud platform server with the help of ESP8266 where the sensor data is represented in terms of graphical statistics.

Muck and Homam [10] have designed a weather station with real-time notifications for climatology monitoring, interfaced to a cloud platform to analyze weather parameters. The weather station is built using SparkFun Weather Shield, Arduino Uno R3, and Weather meter. The collected data from the sensors are stored in Google Cloud using Raspberry Pi 3 Model B and analysis of weather data are done. To display the real-time weather conditions in graphical representation a website and mobile application are developed using Google Data Studio and Android Studio respectively which are accessible by users. Notifications will be sent to the users regarding the weather conditions regularly through social media platforms.

Girija et al [11] have proposed an advanced solution for monitoring the climate conditions at a particular place and the information obtained is made visible anywhere in the world. The technology behind the system is IoT, which is an advanced and efficient solution for connecting things to the internet. The data obtained by monitoring weather conditions like temperature, humidity, and CO level are sent to a web page for graphical representation. To protect public health from pollution, the model provides an efficient and low-cost solution for continuous monitoring of the environment.

Nilesh et al [12] have developed a weather monitoring system that provides information about weather changes to farmers and helps them with respective guidelines to plan their field. The weather parameters temperature and humidity are monitored. The system uses a wireless sensor network for sending information over long distances consuming low power. Low power proves to be an advantage and the system can be easily installed and managed at locations where there is no access to electricity. IoT technology is used to connect devices and the collected data and analysis results are directly available to the end-user.

Yashaswi et al [13] have proposed a system that is an advanced solution for weather monitoring that uses IoT to make data easily accessible in real-time. The system senses weather changes like temperature, humidity, moisture, wind speed, UV radiation, light intensity, carbon monoxide levels in the air using multiple sensors interfaced to a microcontroller. These sensors are placed at different locations to collect the data to forecast the behavior of a particular area of interest. The data collected are displayed on the web page with the help of graphical statistics.

VivekBabu et al [14] developed a monitoring system that provides accurate information about environmental changes on a more local level. The system monitors temperature, humidity, soil moisture, rainfall, and light intensity using various sensors interfaced with Raspberry Pi. The system with comparative analysis is deployed in an agricultural field for effective monitoring of the farm which results in cost reduction, asset saving, and effective management in farming.

Ravi and Snehashish [15] proposed a system that senses the temperature, pressure, humidity, light intensity, and rain value. The micro-controller used for the system is ESP8266 based Wireless Fidelity module NodeMCU. The sensors like temperature and humidity sensor, pressure sensor, raindrop module, and light-dependent resistor are connected to NodeMCU to provide accurate values. Whenever the values exceed a chosen threshold limit an SMS, an E-mail, and a Tweet post is sent alerting the user of the system. IoT is used to connect the system through a network and then retrieve the data from the system which can be distributed or upload to any cloud service where one can analyze and process the gathered information.

Iswanto et al [16] in their work presented an IOT based system to measure, read and process the physical quantities of weather conditions. The weather conditions mentioned are temperature and humidity, intensity of light, rainfall and also wind speed and direction. The reading of these quantities is carried out with analog and digital sensors integrated with the ESP8266 microcontroller. Anemometer is used to measure the windspeed, rotary encoder is used to detect the wind direction. Temperature and humidity are measured with the help of DHT11 sensor. An ESP8266 microcontroller is used which acts as a field station connected with all the sensors. The collected data are sent to a server with the help of Long-Term Evolution Modem. The data from the server is then collected by a base station which consists of Raspberry Pi and monitor to display the data.

Varghese et al [17] have proposed their weather station which goals for an economic and purposeful system which can provide a solution for instant weather information and prediction of the weather data. The system comprises of Raspberry Pi 3 as a micro-computer which is interfaced by DHT22 for temperature and humidity and a BMP180 sensor for air pressure. The Raspberry Pi 3 is codded in Python language works on kernel of Linux and coding part is done in the Thonny text editor. The data is sent to ThingSpeak server over Wi-Fi network for data logging. The data can be downloaded in .csv format for the further studies. The weather prediction is done using linear regression model using Machine learning. The weather prediction can be done on a data collected over a period of 2 years. The proposed system was less accurate with an average error of +/-1.2 degree in temperature, mean error of 2% in humidity and average error of 2hPa in pressure. Also, the main disadvantage of the system is that it cannot be implemented in the remote places.

Kanaka and Vidyasagara [18] have designed a system that senses the environment parameters with the help of sensor and Arduino UNO collects the data and sends to the ESP 8266 which has inbuilt Wi-Fi module, that is connected to Internet. The data is then pushed to the cloud storage where the data is analysed and processed. The stored data can also be used for weather prediction. The proposed system uses Arduino UNO as a microcontroller that collects data from LM35 (temperature sensor), MQ-7 (Carbon Monoxide sensor), Sound sensor, LDR (Light dependent resistor). The data collected by the microcontroller is uploaded to the cloud with the help of ESP8266 Wi-Fi module. The data collected can be presented with the help of Web sites, mobile applications or desktop applications, else the data can be downloaded in CSV format. The data collected over time for around year or two can be used for weather prediction.

Govardhan et al [19] have proposed a system that can provide various environment data such as Temperature, relative humidity and barometric pressure. The system uses Raspberry Pi Mod B+ as a microcontroller which is interfaced with Sense Hat and a Wi-Fi Dongle. Sense Hat has several sensors integrated on it used for various applications. The sense Hat has the following features: Gyroscope, Accelerometer, Magnetometer, Barometer, Temperature sensor, Relative Humidity sensor, 8x8 LED display and a small 5 button joystick. The Wi-Fi dongle is used to provide Raspberry Pi with internet connectivity. The Sense Hat collects the data from the sensors and gives it to the Raspberry Pi. The Raspberry Pi connected to internet sends the data to ThingSpeak server, where the various data is displayed graphically. The accuracy of the system is low because there are no dedicated sensors, but the cost of the system is low.

Mittal et al [20] have developed an ‘Online Smart Weather Station System’ for studying the correlation amongst multiple weather parameters data, collected over a period of 18 months.

The system consists of sensors that generate weather data, local storage, wireless transfer to control center, and web-based online representation and analysis of data. Six weather parameters such as air temperature, relative humidity, wind-speed, rain, rain-rate and solar energy are studied. Statistical measures such as mean, median, standard deviation, min, max and normalized standard deviation are computed from the weather data. The correlation among temperature, humidity and wind-speed parameters is presented. Few important observations are made, for possible applications in agriculture, construction and manufacturing activities. Further a web-based service can be aimed at automated analysis of weather data generated by the smart weather station.

CONCLUSION

The proposed micro weather station system consists of multiple sensors interfaced with a Raspberry Pi microcomputer, which provides accurate weather parameters in real-time. The accuracy of the system mainly depends on the quality of the sensors used. The captured data from the sensors will be displayed in a webpage real-time. The data collected over time in the SD card can be used for various purposes such as weather prediction and research purposes. For weather station located in remote areas, solar power supply with a battery backup is used.

REFERENCES

[1] A. Sravani, Ch. Sudhakar, N. Thirupathi Rao, Debnath Bhattacharyya, Tai-Hoon Kim, "A Concrete Weather Reporting System Model using IoT", International Journal of Control and Automation, Vol. 11, No. 4 (2018), pp.11-20.

[2] Kajal N. Dhawale, Dr. Narendra Bawane "IoT Based Smart Agriculture System", IOSRJEN, ISSN (e): 2250-3021, ISSN (p): 2278-8719, Vol. 09, Issue 5 (May. 2019), ||S (VIII) || PP 04-09.

[3] S.D. Shewale, S.N. Gaikwad, “An IoT Based Real-Time Weather Monitoring System Using Raspberry Pi”, International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol. 6, Issue 6, June 2017.

[4] K. Sai Nikhilesh, Y. Harish Raaghavendra, P.J. Madhu Soothanan, Resmi R, “Low-cost IoT based weather monitoring system for smart community”, Fourth International Conference on Inventive Systems and Control (ICISC 2020).

[5] Gaurav Verma, Pranjul Mittal, Shaista Farheen, “Real Time Weather Prediction System Using IOT and Machine Learning”, IEEE, 2020.

[6] Kurella Vijayalakshimi, Dr. G.V. Maha Lakshmi, “Real Time Weather Monitoring from Remote Location Using Raspberry pi”, International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol. 5, No. 12, Dec 2016.

[7] Devarakonda Uma, DR. K. Raja Kumar, “Design of Weather Monitoring System using Raspberry Pi and Arduino”, IJATIR, ISSN 2348–2370, Vol. 08, No. 24, Pages:4633 4638, December-2016.

[8] R. Udaya Bharathi, M.Seshashayee, ”Weather and Air Pollution real-time Monitoring System using Internet of Things”, International Journal of Innovative Technology and Exploring Engineering (IJITEE), ISSN: 2278-3075, Volume-8 Issue-9, July 2019.

[9] Bulipe Srinivas Rao, Prof. Dr. K. Srinivasa Rao, Mr. N. Ome,” Internet of Things (IOT) Based Weather Monitoring system”, International Journal of Advanced Research in Computer and Communication Engineering, Vol. 5, Issue 9, September 2016.

[10] P Y Muck, M J Homam, “Iot Based Weather Station Using Raspberry Pi 3’, International Journal of Engineering & Technology, 2018.

[11] Girija C, Harshalatha H, Andreanna Grace Shires, Pushpalatha H P, “Internet of Things (IOT) based Weather Monitoring System”, International Journal of Engineering Research & Technology (IJERT), 2018.

[12] Nilesh Patil, Yash Sawant, Tanveet Kaur, Siddhi Kate, Chintan Wagh, “IOT Based Weather Monitoring for Agriculture”, International Journal on Recent and Innovation Trends in Computing and Communication, Vol. 6, No. 4, April 2018.

[13] Yashaswi Rahut, Rimsha Afreen, Divya Kamini, “Smart weather monitoring and real time alert system using IoT”, International Research Journal of Engineering and Technology (IRJET), Volume: 05 Issue: 10, Oct 2018.

[14] K VivekBabu, K Anudeep Reddy, CM Vidhyapathi, B. Karthikeyan, “Weather Forecasting Using Raspberry Pi with Internet of Things (IOT)”, ARPN Journal of Engineering and Applied Sciences, Vol. 12, No. 17, Sept 2017.

[15] Ravi Kishore Kodali, Snehashish Mandal, “IoT Based Weather Station”, International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT), 978-1-5090-5240-0/16, IEEE, 2016.

[16] Iswanto,Prisma Megantoro,Brahmantya Aji Pramudita,” IoT-based weather station with python user interface for measurement technique of educational purpose”, AIP Conference Proceedings 2296, 020027 (2020).

[17] Levin Varghese, Gerard Deepak, Santhanavijayan A, “An IoT Analytics Approach for Weather Forecasting using Raspberry Pi 3 Model B+”, Fifteenth International Conference on Information Processing (ICINPRO), 2019.

[18] Priyanka V Kanaka, Vidyasagar .M, “IOT Based Weather Monitoring System”, International Journal of Innovative Research in Computer and Communication Engineering, Vol.5, Special Issue 4, June 2017.

[19] G. Govardhan, S. Javeed Hussain and S. A. K. Jilani, “A Smart Gadget to Analyse the Weather Changes Using Sense Hat Sensor and Internet of Things (IoT)”, Indian Journal of Science and Technology, Vol 9(35), September 2016.

[20] Yash Mittal, Anshika Mittal, Diksha Bhateja, Komal Parmaar, V. K. Mittal, “Correlation Among Environmental Parameters using An Online Smart Weather Station System”, IEEE, 978-1-4673-6540-6/15, 2015.

Project Paper- Design and Implementation of Micro Weather Station

Mrs. Vijetha T S
Asst Professor, AIET, India
tsvijetha@gmail.com

Mr. Abhishek M Shastry K
Student, AIET, India
abhishekshastry1999@gmail.com

Mr. Abhishek V Mahendrakar
Student, AIET, India
abhi2244mahendrakar@gmail.com

Mr. Akshay
Student, AIET, India
akshaymarkal99@gmail.com

Mr. K Muthu
Student, AIET, India
muthu26july@gmail.com

Abstract

The advancement of IoT has made a major impact on technology. It collects enormous amount of data using which can be used in an application. One such system is Micro-Weather Station, which gives us different environment variables such as temperature, humidity, soil moisture, luminous intensity, air pressure, wind speed, rain fall etc. Due to human activities these days there is a drastic change in the climate, hence an accurate and cost-efficient system is needed which is used to monitor the changes in the environment. The application of weather station is not just bounded for getting the live data and prediction of weather, but also includes the advancement in agriculture sector and in the military applications. The cloud storage technology and Geo-tagging has made it much simpler to get the data of any place any time.

Keywords: Micro Weather Station; IoT; Micro-controller; Climate; sensor interfacing; Data logging; Database; Cloud storage; Weather monitoring.

Introduction

Micro weather station is a system which measures the environmental variables which is transferred to server where the computations are made and displays the result in web page or mobile applications. The sole purpose of this micro weather station is to gather/collect major weather attributes like temperature, humidity, soil moisture, luminous intensity, air pressure, wind speed, rain fall etc. at remote location by using most effective and cost-efficient manner with the help of micro weather station equipment. This uses multiple micro weather stations and aggregates the data provided by multiple data collection points in real time. Raspberry Pi is used as micro-controller to which various sensors are interfaced using general purpose input output (GPIO) pins. The Raspberry Pi has an inbuilt SD card slot that helps to store the fetched data. All the data fetched are time tagged from the Raspbian OS, in addition to this the system is Geo-tagged which gives the location of weather station. The data fetched by Raspberry Pi will then be pushed to Web server where the required calculations are made and displays the data in terms of table/graph with the help of Webpage. This enables the user an easy and reliable way to understand the data and make further decisions accordingly. The data collected over time can be used for various purposes such as weather prediction, change in weather pattern, research purpose, etc.

Motivation

Today's weather is changing drastically leading to an effected life cycle on earth due to the random change in our environment. Most of the changes include increase in temperature, decrease in oxygen level, increase in level of toxic gases, excessive or scanty rainfall from standard level and decrease in overall air quality. All these changes have led to an unhealthy life and indirectly affecting all living beings on earth. Also, the agricultural practices around the world are largely dependent on local weather conditions. As controlling the weather condition is very difficult, monitoring of local weather conditions can help in taking corrective or remedial actions to minimize loss due to changing conditions.

The effects can be controlled by monitoring weather parameters like temperature, humidity, rainfall, wind speed and toxic gases. Automatic weather stations have completely transformed the way weather is monitored today. The invention of automatic weather stations drastically simplified the process of documenting and analysing the weather data. Using AWS, weather conditions can be monitored remotely and also analysed for longer periods of time, helping generate accurate weather forecasts.

Objectives

In Micro Weather Station different sensors are interfaced with the microcontroller to get different weather parameters and stored in the server by pushing the data over a network, calculations are made for ease of readability and understand the data for the user by displaying the data graphically in the website which can be accessed from anywhere on the planet. The cost and accuracy are the two main factors that defines the cost quality of the system, and the problems during network dis-connectivity. Hence the objectives of the system are:

- To bring up the accuracy of the data collected.
- To create a feasible system that can handle data from multiple sensors in real-time.

- ❖ Usage of multiple micro weather stations and aggregates the data provided by multiple data collection points in real time.

Proposed system

The system mainly concentrates on providing the user with precise weather parameters which is affordable to common man. In this system Raspberry Pi 3 Model B+, a powerful micro-computer is interfaced with different sensors to get accurate weather parameters. The current means of weather monitoring are indeed rather limited and make use of very expensive weather stations, leading to lack of comprehensively monitoring due to cost constraints and inconveniences. But the proposed system is small in size and possesses high precision as similar to that of high-end weather station with significantly less cost. This is achieved by using multiple micro weather station with low-cost sensors which produces high precision data with lesser power consumption.

Hardware and Software used

Hardware's tools used:

- ❖ Raspberry Pi board
- ❖ ADS1115: Analog-to-Digital converter
- ❖ DSB18B20: Temperature sensor
- ❖ Capacitive soil moisture sensor v1.2
- ❖ MQ135: Gas sensor
- ❖ DHT22: Temperature and Humidity sensor
- ❖ BMP280: Pressure Sensor
- ❖ GUVA-S12SD: UV Light sensor

Software tools used:

- ❖ Microsoft Visual Studio
- ❖ Raspberry Pi OS
- ❖ EXPO
- ❖ KiCAD

Languages used:

- ❖ Python
- ❖ JavaScript
- ❖ MATLAB
- ❖ HTML

Methodology

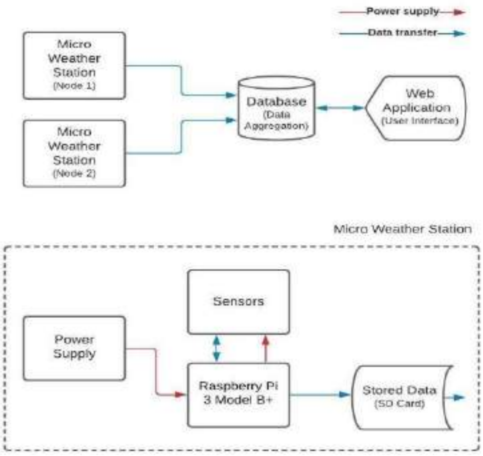


Figure 1. Workflow of the system

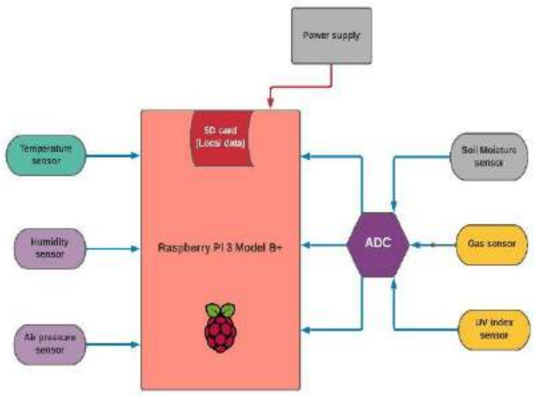


Figure 2. Block diagram of a single node

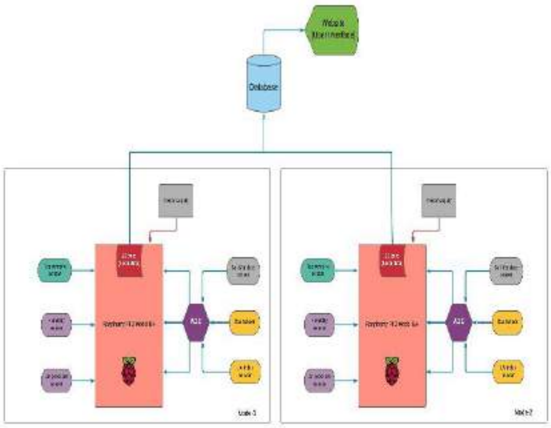


Figure 3. Block diagram of Micro Weather Station

Step 1: Initial Setup

The system consists of a central component raspberry pi, a powerful micro-computer which controls all the operation of the system. The micro-computer can be powered by 2.5A micro-USB power supply for configuration when the implementation of the system is nearer to a supply source. The data is uploaded into the webserver using the internet connection through Wi-Fi.

Step 2: Sensor Interfacing

The system comprises of multiple sensors such as temperature sensor, humidity sensor, capacitive soil moisture sensor, air quality sensor, ultraviolet (UV) detection sensor and air pressure sensor are interfaced with the raspberry pi with the help of GPIO pins. In order to read the analog values from the capacitive soil moisture sensor, UV detection sensor and the air quality sensor an Analog to Digital Convertor (ADC) module is used. The sensors are configured and calibrated using python IDE tool. All the sensors are powered individually from the PCB board which provides the supply voltage parallelly.

Step 3: Data Logging

Collecting and storing the data over a period of time in order to analyse specific trends or to record the data based on events is very important in this system. The accurate weather data which is time tagged collected from multiple sensors are independently data logged.

The sensor data is primarily stored in the SD card by partitioning the unused memory from the Raspbian OS. The data can be stored in the form of Comma-Separated Values (CSV) file or JavaScript Object Notation (JSON) file depending upon the requirement using the python programming. The log files are generally found in /var/log folder and most of the files in there can only be read by the root user.

Step 4: Database

A web database can be accessed from a local network or the internet instead of one that has its data stored on a desktop or its attached storage. The data logged from each node is further pushed to the webserver with the help of internet connection through Wi-Fi and also the same is stored in the SD card as a backup. Over a period of time the data is updated to the server in real-time. The real-time data collected from multiple nodes are aggregated by performing some mathematical operations to ensure accuracy.

Step 5: Web App

The analysing and visualizing of the data collected from the sensors is important which makes it easier for the user to understand the data. To build a web app, expo framework is used which is a set of tools and services built around React Native and native platforms that helps to develop, build, deploy, and quickly iterate on iOS, Android, and web apps from the same JavaScript/TypeScript codebase.

React Native is an open-source framework with one of the largest support communities. Unlike typical hybrid applications, this new framework targets mobile platforms. It's based on the JavaScript library, React Native allows you to create a single JavaScript codebase that will work on different mobile devices (iOS, Android & Windows).

The user interface is the part of the web application which a user interacts with. In simple terms, it's everything a user can see and touch, such as buttons, colours, fonts, navigation, etc. The user interface is used to display real-time data from the sensors and also the data is visualized by plotting the graph. The data stored in the webserver can be accessed/imported in the web app, for applications like weather predictions, research etc.

Issues of the project

The field applications of weather station are in vast areas, the accuracy of the data is taken as one of the major factors to decide the quality of the system. But if in an application accuracy is needed, then the cost of system increases due to high grade sensors used. Similarly, if application needs to be built at low cost then accuracy has to be compensated.

Also, the system should be reliable to the user to access and maintain the data. If there is any problem in the system it must be easy to repair/replace the parts. And if the system encounters any network issues, then there must be some way to retrieve the lost data with ease.

Results

Initially programs are dumped to the Raspberry Pi using the python IDE to execute the tasks.

Step 1: Dumping code onto the Raspberry Pi is done via Python IDE installed in Raspbian OS.

Step 2: The captured data from the sensors are pushed to the cloud via a write API call to the ThingSpeak channel.

Step 3: The real-time data is visualised and stored in the ThingSpeak cloud platform. Graphical representation of real-time data in the ThingSpeak channel.

Step 4: Aggregating of real-time data is done with the help of MATLAB Analysis present in the ThingSpeak platform. Aggregation of data is done when either of the nodes (Node1 or Node2) gets triggered. Finally, the real-time aggregated data is pushed into another ThingSpeak channel. MATLAB Analysis.

Step 5: · To display the sensor values in a user interface, a web application is built with the help of Expo platform. Since the ThingSpeak platform contains plots of real-time sensor values, the plot can be displayed in the web application by using the iframe tag of the plot.

Step 6: The location (latitude and longitude) of the nodes is displayed in the web application by using react-leaflet. And to get the complete address from latitude and longitude coordinates TomTom Reverse Geocoding API is used. The data from individual nodes as well as averaged data can be downloaded in csv format.

Conclusion

The design of using multiple nodes with interfacing low-cost sensors can significantly bring down the cost since rather than using a single large expensive weather station, multiple nodes can be used which are interconnected to each other to achieve the same performance. In this system, each node with multiple sensors interfaced provides the full fetched functionality of collecting weather parameters similar to that of a high-end weather station. The data collected independently from different nodes are aggregated in the webserver and finally displayed through the user interface (webapp) in real-time which helps the user in understanding the pattern of the data collected and make further decisions accordingly.

References

[01] S. D. Shewale, S. N. Gaikwad, "An IoT Based Real-Time Weather Monitoring System Using Raspberry Pi", An ISO 3297:2007

[02] Bulipe Srinivas Rao, Prof. Dr K. Srinivasa Rao, Mr. N. Ome, "Internet of Things (IOT) Based Weather Monitoring system", International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE), Vol. 5, No. 9, Sep- 2016.

[03] Devarakonda Uma, DR. K. Raja Kumar, "Design of Weather Monitoring System using Raspberry Pi and Arduino", IJATIR, ISSN 2348–2370, Vol. 08, No. 24, Pages:4633-4638, December-2016.

[04] Sourabh Halder, G Sivakumar, "Embedded based Remote Monitoring Station for Live Streaming of Temperature and Humidity", International Conference on Electrical, Electronics, Communication, Computer and Optimization Techniques (ICEECCOT), 2017.

[05] A. Sravani, Ch. Sudhakar, N. Tirupati Rao, Debnath Bhattacharyya, Tai-Hoon Kim, "A Concrete Weather Reporting System Model using IoT", International Journal of Control and Automation, Vol. 11, No. 4 (2018), pp.11-20

[06] Kajal N. Dhawale, Dr Narendra Bawane, "IoT Based Smart Agriculture

System", IOSRJEN, Vol. 09, No. 5 (May. 2019).

[07] R. Udaya Bharathi, M. Seshashayee, "Weather and Air Pollution real-time Monitoring System using Internet of Things", International Journal Innovative Technology and Exploring Engineering (IJITEE), Vol. 8, No. 9, July- 2019.

[08] Md. Imranul Haque, Abu Hena MD. Shatil, Abu Noman Tusar, "Renewable Powered Portable Weather Update Station", International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST), 2019.

[09] Iswanto, Prisma Megantoro, Brahmantya Aji Pramudita, "IoT-based weather station with python user interface for measurement technique of educational purpose", AIP Conference Proceedings 2296, 020027 (2020)

[10] Gaurav Jadhav1, Kunal Jadhav2, Kavita Nadlamani3, "Environment Monitoring System using Raspberry-Pi", International Research Journal of Engineering and Technology (IRJET), Vol. 3, No. 4, April-20

APPENDIX

CERTIFICATE

OF PUBLICATION

ISSN-NO- 1671-9727

This is to certify that

PAPER ID: CDU-0721-178

Vijetha T S

Assitant Professor, Department of Electronics and Communication, Alva's Institute of Engineering and TechnologyMangalore, India

Published a paper entitled

"A Review on Design and Implementation of Micro Weather Station"

in

Journal of Chengdu University of Technology

Editor-in-chief

OPEN ACCESS

.periodicos.

CERTIFICATE

OF PUBLICATION

ISSN-NO- 1671-9727

This is to certify that

PAPER ID: CDU-0721-178

Abhishek M Shastry K

Final year BE student, Department of Electronics and Communication, Alva's Institute of Engineering and Technology Mangalore, India

Published a paper entitled

"A Review on Design and Implementation of Micro Weather Station"

in

Journal of Chengdu University of Technology

Editor-in-chief

ज्ञान-विज्ञान विमुक्तये

OPEN ACCESS

CAPES .periodicos.

CERTIFICATE

OF PUBLICATION

ISSN-NO- 1671-9727

This is to certify that

PAPER ID: CDU-0721-178

Abhishek Mahendrakar

Final year BE student, Department of Electronics and Communication, Alva's Institute of Engineering and Technology Mangalore, India

Published a paper entitled

"A Review on Design and Implementation of Micro Weather Station"

in

Journal of Chengdu University of Technology

Editor-in-chief

ज्ञान-विज्ञान विमुक्तये

OPEN ACCESS

.periodicos.

CAPES

CERTIFICATE

— OF PUBLICATION —

ISSN-NO- 1671-9727

This is to certify that

PAPER ID: CDU-0721-178

Akshay

Final year BE student, Department of Electronics and Communication, Alva's Institute of Engineering and Technology Mangalore, India

Published a paper entitled

"A Review on Design and Implementation of Micro Weather Station"

in

Journal of Chengdu University of Technology

Editor-in-chief

OPEN ACCESS

.periodicos.

CERTIFICATE

OF PUBLICATION

ISSN-NO- 1671-9727

This is to certify that

PAPER ID: CDU-0721-178

K Muthu

Final year BE student, Department of Electronics and Communication, Alva's Institute of Engineering and Technology Mangalore, India

Published a paper entitled

"A Review on Design and Implementation of Micro Weather Station"

in

Journal of Chengdu University of Technology

Editor-in-chief

OPEN ACCESS