IOT BASED SOLAR ENERGIZED WEATHER FORECAST MONITORING SYSTEM

A PROJECT REPORT

Submitted by

SANTHYA M (2017504037)

NIKKILESHH M (2017504062)

SRINIVASAN SHYAM VIJAYARAGAVAN (2017504070)

in fulfillment for the award of the degree
of

BACHELOR OF ENGINEERING
IN
ELECTRONICS AND COMMUNICATION ENGINEERING

MADRAS INSTITUTE OF TECHNOLOGY

ANNA UNIVERSITY : CHENNAI 600 044

APRIL 2021

ANNA UNIVERSITY, CHENNAI

BONAFIDE CERTIFICATE

Certified that this project titled **"IOT BASED SOLAR ENERGIZED WEATHER FORECAST MONITORING SYSTEM"** is the bonafide work of **Ms. SANTHYA M (2017504037), Mr. NIKKILESHH M (2017504062), and Mr. SRINIVASAN SHYAM VIJAYARAGAVAN (2017504070)** who carried out under my supervision.

SIGNATURE

Dr. M. GANESH MADHAN

Head of the Department,
Department of Electronics Engineering,
Madras Institute of Technology,
Anna University,
Chennai - 600 044.

SIGNATURE

Dr. P.T.V.BHUVANESWARI

Faculty and Supervisor,
Department of Electronics Engineering,
Madras Institute of Technology,
Anna University,
Chennai - 600 044.

ACKNOWLEDGMENT

We would like to record our sincere thanks to **Prof. T. THYAGARAJAN,** Dean, Madras Institute of Technology, for having given consent to carry out the project work at Madras Institute of Technology campus, Anna University.

We wish to express our sincere appreciation and gratitude to **Dr. M. GANESH MADHAN**, Head, Department of Electronics Engineering, Madras Institute of Technology campus, Anna University.

With esteemed respect, we express our sincere appreciation and gratitude to our supervisor **Mrs. P. T. V. BHUVANESWARI,** Department of Electronics Engineering, for her guidance, constant encouragement and support. Her extensive vision, practical inputs and creative thinking has been a source of inspiration throughout this project.

We sincerely thank all the project panel members **Dr. K. INDRAGANDHI, Dr. S. EZHILARASI** and **Ms. P. KASTHURI** for their valuable suggestions.

SANTHYA M (2017504037)

NIKKILESHH M (2017504062)

SRINIVASAN SHYAM VIJAYARAGAVAN (2017504070)

ABSTRACT

Weather monitoring system provides users with access to real-time weather data from different locations. The integration of IoT along with green energy enables monitoring of weather data from any part of the world. The system proposed in this paper is an advanced solution to get live reporting of weather conditions. It monitors weather parameters at a particular place and is powered using solar energy, a renewable and abundant energy source. The parameters are visualized graphically in Thingspeak cloud platform and can also be monitored in Adafruit dashboard. A client-server based model is developed with the help of MQTT application which enables continuous updation of weather parameters in IoT cloud. Live location is tracked using an inbuilt GPS module which updates latitude and longitude in a HTML web page. Weather forecasting is performed using machine learning algorithms to predict future weather conditions at a particular place and provide prior information on weather status.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ABBREVIATIONS	DESCRIPTION
IOT	Internet of Things
LCD	Liquid Crystal Display
GPS	Global Positioning System
MQTT	Message Queuing Telemetry Transport
M2M	Machine to Machine
MM2MM	Many Machines to Many Machines
LSTM	Long Short Term Memory
WSN	Wireless Sensor Networks
UV	Ultra Violet
API	Application Programming Interface
MCU	MicroControllerUnit
RH	Relative Humidity
HI	Heat Index
RTD	Resistive Temperature Detectors
BMP	Barometric Pressure Sensor
SPI	Serial Peripheral Interface
RMS	Root Mean Square
I2C	Inter-Integrated Circuits
CS/CSB	Chip Select
SDO	Serial Data Out
SDA	Serial Data
SCK/SCL	Serial Clock

AQI	Air Quality Index
TTL	Transistor-Transistor Logic
PPM	Parts Per Million
RDB	Rain detection Board
ADC	Analog to Digital Convertor
LED	Light Emitting Diode
PCB	Printed Circuit Board
WHO	World Health Organization
GNSS	Global Navigation Satellite Systems
SoC	System on Chip
SDK	Software Development Kit
RAM	Random Access Memory
CPU	Central Processing Unit
HTML	HyperText Markup Language
UART	Universal Asynchronous Receiver
USB	Universal Serial Bus
GPIO	General Purpose Input Output
PWM	Pulse Width Modulation
FIFO	First In First Out
TCP	Transmission Control Protocol
RSS	Residual Sum of Squares
RSE	Residual Standard Error
GSM	Global System for Mobile Communication

CHAPTER 1

INTRODUCTION

With the introduction of high-speed internet, more and more people all over the world are becoming linked. The Internet of Things (IOT) takes this a step further by connecting not only people but also electronic devices that can communicate with one another [1]. With the decreasing cost of Wifi-enabled devices, this trend can only gain traction. The key idea behind the IoT is to connect different electronic devices through a network, then retrieve data from these devices (sensors), which can be distributed in any way, and upload it to any cloud service where the data can be analyzed and processed [2]. These data can be used in the cloud service to warn people in different ways.

IoT allows devices to talk directly to each other, make joint decisions, and exchange data between devices without the need for the cloud or servers.The word "Things" can represent physical devices, machines and sensors which are used to collect data from the environment. Weather monitoring aims at analyzing different weather parameters at various locations. Climatic change and environmental monitoring have received much attention. Hence monitoring weather parameters becomes the need of the hour. Weather monitoring systems (WMS) are being developed particularly in the view of building smart cities by giving the weather update at any particular place [3].

Existing WMS can process weather data using microcontrollers namely Raspberry pi, Arduino Uno and displays the parameters either in an LCD or in a serial monitor [5]. They are powered using conventional sources of power supply such as AC adapters. These systems have no portability hence cannot be carried over long distances. The systems which developed in [1] can measure only a minimum number of parameters. Conventional weather stations do not have a tracking feature such as GPS. These existing systems use SMS/E-mail based data transfer which limits the amount of data to be transmitted.

Using solar energy to power the entire setup has enormous advantages which includes low maintenance costs and diverse applications. Hence, this research aims at developing a smart WMS which is powered using a renewable energy source such as solar energy. The monitored parameters are displayed in an LCD screen and graphically visualized in Thingspeak and Adafruit cloud. Live location is tracked by using a GPS module interfaced with Nodemcu. Besides, forecasting is also performed using machine learning algorithms.

Implementing conventional weather stations for higher performance has the issue of heavy expense. The hard drive based data logging facility requires a separate system setup for its operation. Power required for operating such weather stations is high[1]. The above problems are the primary concerns of a weather monitoring setup and the proposed system aims at providing a cost effective solution to provide efficient weather monitoring. The objective is to monitor weather parameters at a particular place by powering the system using a renewable energy source and to display the output in an LCD and graphically visualize the weather parameters in Thingspeak IoT cloud platform. This research aims to use MQTT broker such as Adafruit_IO[17], to facilitate GPS monitoring of the system and to carry out weather prediction using a machine learning model by using a real-time dataset uploaded in Adafruit_IO cloud.

The contents of the report are organized as follows. Chapter II elaborates about the related literatures to the proposed work. Chapter III describes the proposed methodology. Chapter IV details out the results and analyzes the output. Chapter V concludes the proposed work with possible future work.

CHAPTER II

LITERATURE SURVEY

In [1] the authors have developed a system that uses Wireless sensor networks (WSN) which are well suited for long term data acquisition for IoT. Environmental cloud computing is used for long term monitoring of environmental parameters. The research deals with the functional design and implementation of the WSN platform that can be used for long-term environmental monitoring as an IoT application. The application requirements are minimal as it features low cost, low power, long life time, low maintenance, high quality of service, fast deployment are considered in the specification of this particular WSN platform.

In [2] the authors have developed a weather station which predicts weather parameters using neural networks. Using long short term memory (LSTM) model, the neural network is trained for different combinations of weather parameters such as precipitation, temperature, dew point and humidity. After training the model with these parameters, prediction is carried out.

In [3] the authors have developed a wireless sensor network and weather monitoring system which keeps track of fire detection, earthquake detection, temperature, humidity and rainfall level. The weather readings are displayed in real time and can also track historical readings on an hourly or daily basis.

In [4] the authors have designed an integrated system for environmental monitoring and management that integrates information system with Internet of Things. It also performs cloud computing on the stored database to infer climatic patterns. Multi-sensors and other web services are used to collect data and other information for the perception layer. The public networks and private networks are used to access and transport bulk data and other information in the network layer. The application layer provides the functions of storing, sharing, organizing and processing of data and other important information, as well as the functions of environmental monitoring and management applications.

In [5] the authors have developed an environment monitoring system for smart cities which enables effective utilization of resources and better quality of services to the citizens. It provides services such as air quality management, weather monitoring and automation of homes and buildings in a smart city. The basic parameters concentrated are temperature, humidity and CO_2. An Android application was developed through which data is transferred from LabVIEW to a smartphone, for monitoring data remotely.

In [6] the authors designed a weather station based on IoT which measures weather parameters such as Temperature, Atmospheric pressure, Humidity percentage and visible light intensity. Whenever the values of the weather parameters exceed the chosen threshold limit, a SMS, an E-mail and a Tweet post is published alerting the owner of the application for necessary measures.

In [7] the authors have developed an advanced solution for weather monitoring that uses IoT to make real time data easily accessible over a very wider range of locations. The system deals with monitoring weather and climate changes namely temperature, humidity, wind speed, moisture, light intensity, UV radiation and carbon monoxide levels in the air using multiple environmental sensors. These sensors send the data to the web page and the sensed data are plotted as graphical statistics. The system consists of an App that sends notifications as an effective alert system to warn people about sudden and drastic weather changes. For predicting more complex weather forecasts, the system uses an API that analyses the data collected by the sensors.

This 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 uses. The system is efficient and can thus be powered by solar panels. With more updation to weather parameters and forecasting algorithms the system can be of great use to meteorological departments, weather stations, aviation and marine industries and even the agricultural industry.

In [8] the authors have developed a weather station that efficiently monitors weather parameters in Thingspeak IoT cloud platform and the estimated trend is displayed in a browser system. The obtained weather results are combined with the concept of Vertical agricultural system[8] which is used to grow crops in small areas which can be monitored remotely.

In [9] the authors have developed a system to create a small smart off-grid solar cell system prototype to provide a backup power source for a smart mushroom farm. This study used IoT, voltage and current sensors to compute and track the voltage of solar cell charging, the ampere of current charging from the solar panel into a battery, and the ampere of current loading from the battery to the irrigation system.

In [10] the authors have developed a system to create a solar based irrigation monitoring system based on sustainable energy and cost efficiency. The possibility of a programmed water system strategy with solar as the primary vitality hotspot for power is demonstrated in this paper. Furthermore, this paper includes the creation of a system using sensors and Node MCU that can monitor even the movement of creatures that may crush the harvests in horticultural fields, as well as capture a picture of those activities and send it to the proprietor.

In [12] the authors pinpointed the Internet of Things (IoT) ability to integrate a large number of diverse and heterogeneous end systems transparently and effortlessly, while still providing an open access to the selected subsets of data for the creation of a variety of digital services. In this paper, the work focuses on an urban IoT scheme which is distinguished by its application domain.

From the above literature survey it is found that weather monitoring stations do not use renewable sources of energy such as solar energy, wind energy, hydrothermal energy to power the whole system and the number of weather parameters measured is limited to the sensors used. Hence, the proposed system is an attempt to use green energy sources to power a weather monitoring station which measures all the basic weather parameters.

CHAPTER III

PROPOSED METHODOLOGY

The proposed system is a progressive solution for weather monitoring which involves data sensing, data processing and cloud enablement through the internet. The system architecture of the proposed IoT based Solar Energized Weather Forecast Monitoring System (I-SWFMS) is illustrated in Fig 1. It comprises the following modules. They are data sensing, data processing, cloud enablement, event prediction, alert notification and solar harvesting.

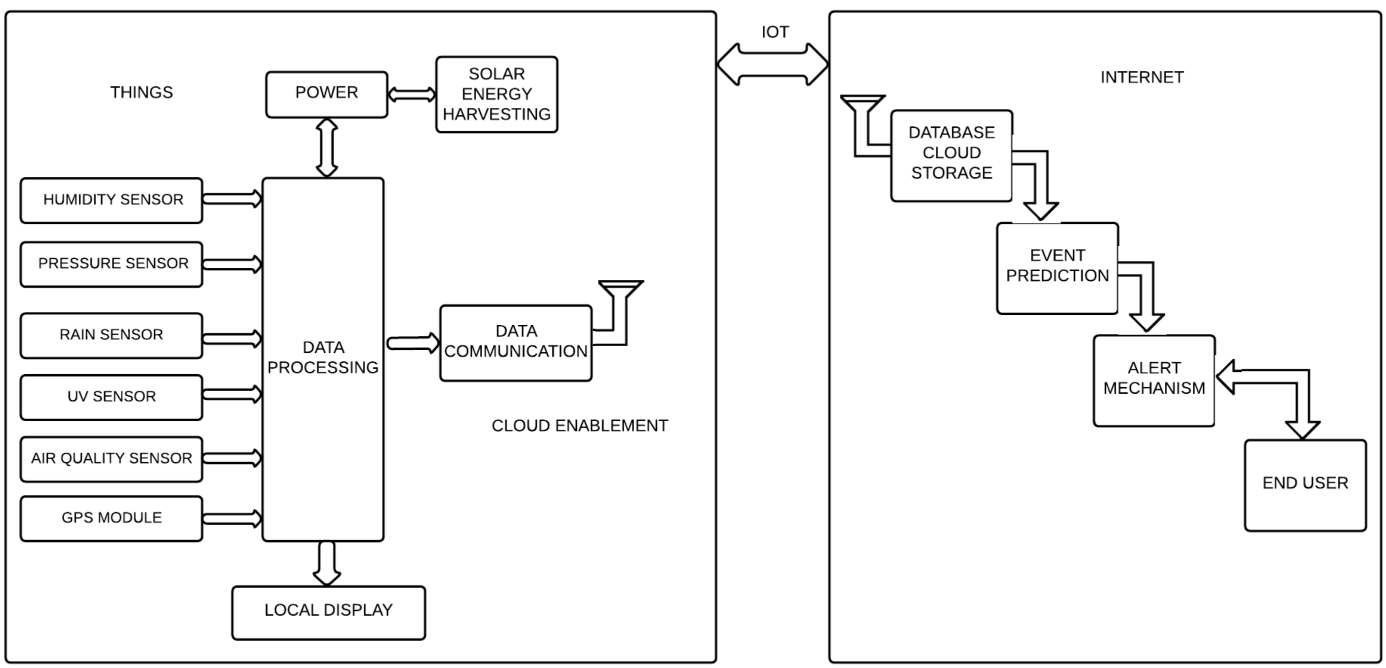


Fig 1: System architecture of proposed I-SWFMS system

The data sensing unit comprises all the weather monitoring sensors along with a GPS module. The data processing unit comprises NodeMCU microcontroller which features data communication through wireless internet for cloud enablement. The solar harvesting unit acts as a renewable source of power to data sensing and data processing units. The integration of IoT enables the system to upload weather data to a cloud database storage which performs event prediction and alerts the end user in case of any anomaly detected in weather conditions. The generalized architecture can be extended for a series of Things which can be extended from "1 to N" for a variety of data sensing units. The details of each module is explained in the subsequent sections of the paper.

3.1 DATA SENSING

A number of environmental sensors have been used to measure real time weather data on a given location. In the proposed system, each sensor measures a minimum of three weather parameters and thus the number of sensors used for measurement is minimized, which inturn improves the efficiency of the system. Since the system is solar powered, it increases the longevity of the application. The following are the sensors used in the proposed I-SWFMS.

3.1.1 TEMPERATURE AND HUMIDITY SENSOR

Temperature is a physical quantity that expresses hot and cold characteristics. It is the manifestation of thermal energy, present in all matter. This energy is the source of the occurrence of heat, a flow of energy, when a body is in contact with another that is colder or hotter [18]. Temperature is one of the most important parameters in a weather monitoring system. Humidity is the concentration of water vapor present in the air. Water vapor, the gaseous state of water, is generally invisible to the human eye. Humidity indicates the likelihood for precipitation, dew, or fog to be present. Relative humidity, expressed as a percentage, indicates a present state of absolute humidity relative to a maximum humidity[19].

Relative Humidity can be expressed as a mathematical formula as given below [19].

Let RH be the relative humidity of atmosphere at a given temperature, $\mathbf{P_{H20}}$ be the partial pressure of water vapor at a given temperature and $\mathbf{P^{*}_{H20}}$ be the equilibrium vapor pressure of water over a flat surface of pure water at a given temperature. Then Relative Humidity can be given as:

$$\text{Relative Humidity (RH)} = \mathbf{P_{H20} / P^{*}_{H20}} \text{ ----- } \mathbf{(3.1)}$$

Relative humidity gives the percentage value indicating the level humidity at a given temperature. Based on temperature and relative humidity yet another parameter known as dew point is computed. Dew point is defined as the temperature to which air must be cooled to become saturated with water vapor [20]. When cooled further, the airborne water vapor will condense to form liquid water (dew). When air cools to its dew point through contact with a surface that is colder than the air, water will condense on the surface.

When the temperature is below the freezing point of water, the dew point is called the frost point, as frost is formed via deposition rather than condensation to form dew. The measurement of the dew point is related to humidity. A higher dew point means there is more moisture in the air[20].

Dew point is calculated based on the expression which involves temperature and relative humidity.

Let Td be the dew point temperature (in degrees Celsius), T is observed temperature (in degrees Celsius), and RH is relative humidity (in percentage). Then, dew point temperature (Td) is given as:

$$\mathbf{Td = T - ((100 - RH)/5) \text{ ----- } (3.2)}$$

The Heat Index (HI) is an index that combines air temperature and relative humidity [21]. It describes the hotness of the atmosphere when relative humidity is factored in with the actual air temperature. The result is also known as the "felt air temperature", "apparent temperature", "real feel" or "feels like". For example, when the temperature is 32 °C (90 °F) with 70% relative humidity, the heat index is 41 °C (106 °F) [21].

The human body normally cools itself by perspiration, or sweating. Heat is removed from the body by evaporation of that sweat. However, high relative humidity reduces the evaporation rate. This results in a lower rate of heat removal from the body, hence the sensation of being overheated [21].

The National weather service (NWS) by National oceanic and Atmospheric administration (NOAA) offers a heat index chart for areas with high heat index and low humidity shown in Fig 2 [22].

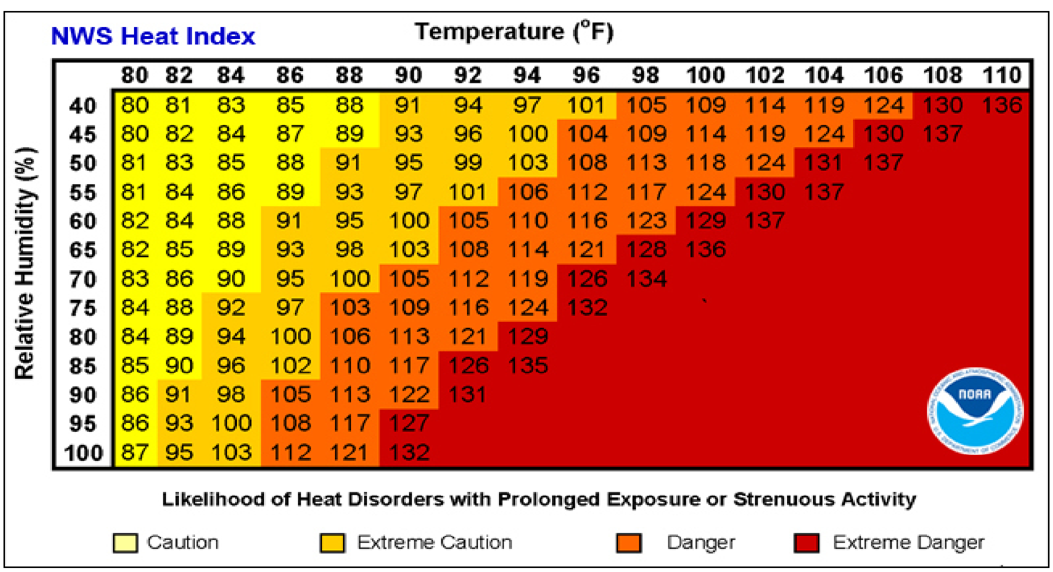


Fig 2: NWS Heat index chart

DHT22 is a low-cost digital sensor used for measuring temperature, relative humidity, heat index and dew point. This sensor can be easily interfaced with any microcontroller such as Arduino, Raspberry Pi etc. to measure humidity and temperature instantaneously. To measure the surrounding air temperature and humidity, this sensor uses a thermistor and a capacitive humidity sensor. Among various temperature sensors namely Thermistors, Resistive Temperature Detectors (RTD) and DHT11, the DHT22 sensor is chosen for the following advantages [23].

1. It has a wider temperature range from -40°C to 80°C
2. It has very low current consumption as low as 0.3mA
3. It has a higher humidity range value 0% to 100 % compared to DHT11 which has a range of 20% to 80%
4. It has high accuracy for temperature and humidity measurements i.e., ±1°C for temperature and ±1% relative humidity

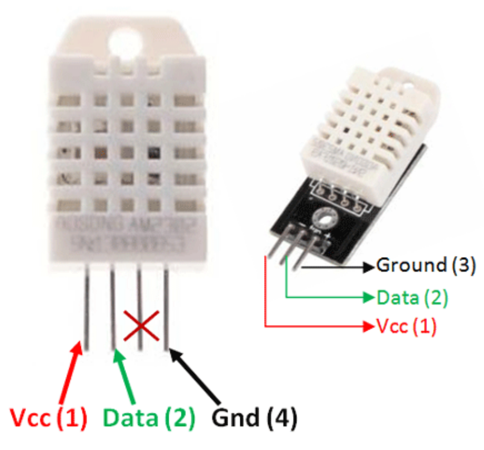


Fig 3: DHT22 Pin configuration

Fig 3 shows the pin configuration of temperature and humidity sensor DHT22. The sensor consists of a capacitive humidity sensing element and a thermistor for sensing temperature. The humidity sensing capacitor has two electrodes with a moisture holding substrate as a dielectric between them. Change in the capacitance value occurs with the change in humidity levels. The IC measures, processes this changed resistance values and changes them into digital form (Voltage).

The pin configuration of DHT22 temperature and humidity sensor is shown in Table 1.

Table 1: Pin configuration of DHT22

S.NO	PIN NAME	PIN DESCRIPTION
1.	Vcc	Power supply 3.5 to 5.5V
2.	Data	Serial data output pins
3.	Ground	Connected to the ground

DHT22 is used in various applications such as measuring humidity and temperature values in heating, ventilation and air conditioning systems [26]. Weather stations also use these sensors to predict weather conditions. The humidity sensor is used as a preventive measure in homes where people are affected by humidity. Offices, cars, museums, greenhouses and industries also use this sensor for measuring humidity values and for safety measures [26].

3.1.2 BAROMETRIC PRESSURE SENSOR

Atmospheric pressure, also known as barometric pressure, is the pressure within the atmosphere of Earth [25]. Atmospheric pressure is closely approximated by the hydrostatic pressure caused by the weight of air above the measurement point. As elevation increases, there is less overlying atmospheric mass, so that atmospheric pressure decreases with increasing elevation. Pressure measures force per unit area, with SI units of pascals (1 Pascal = 1 newton per square meter, 1 N/m^2)[25] .

Altitude refers to height above sea-level or distance above sea-level. It is usually expressed in meters. Altitude has an inverse relation with pressure i.e., as altitude increases pressure decreases and vice-versa. Persistence of this relation is due to the following two phenomenons[25].

The first phenomenon is gravity: The gravity of Earth pulls air as close to the surface as possible.

The second phenomenon is density: As altitude increases, the amount of gas molecules in the air decreases. The air becomes lesser dense than air nearer to sea level and is called "thin air." Thin air exerts less pressure than air at a lower altitude.

The atmospheric pressure at a height ‘h’ can be calculated using the following expression:

$$\mathbf{P_h = P_0\, e^{-mgh/kT} \text{ ----- } (3.3)}$$

where P_h is the pressure at height h, P_0 be the sea level pressure, g is the acceleration due to gravity, k is the Boltzmann's constant ($1.38064852 \times 10^{-23}$ m^2 kg s^{-2} K^{-1}), T is the absolute temperature and m is the mass of one air molecule [25].

The Barometric pressure sensor is a low-cost, accurate sensor that is used to measure pressure and altitude [28]. The BMP280 is an upgraded version of BMP180. It comes with a smaller footprint, lower power consumption, lower noise measurements, higher resolutions for pressure and temperature, lower RMS noise, newly added interface SPI, more measuring modes, higher measuring rate and newly added filter against environmental interference. Since the atmosphere pressure reading is affected by altitude and temperature, compensation features have been added in the BMP280 sensor library.

Hence, it is more reliable in providing precise temperature, atmospheric pressure values and altitude data. The features of BMP280 compared to other pressure sensors are listed below:

1. The BMP280 sensor is the next generation upgrade to the BMP085/ BMP180 /BMP183 which has pressure and altitude measurement
2. This sensor is greater for all sorts of weather sensing and can even be used in both I2C and SPI interfaces
3. This precision sensor is the best low-cost, precision sensing solution for measuring barometric pressure with ±1 hPa absolute accuracy. As pressure changes with altitude, it can also be use it as an altimeter with ±1 meter accuracy.

The BMP280 sensor operates from 3.3V [30]. Hence it requires 3.3V power and must be driven with 3.3V logic levels. If needed to be operated at 5V, it can be made possible by using voltage regulator and level shifters. It is typically recommended to operate it on 3.3V and maximum at 3.6V DC. The GY-BMP280 module simply supports both I^2C and SPI interfaces and comes with a default I^2C address of 0x76. The Chip Select (CS) and Serial Data Output (SDO) pins of the BMP 280 are necessary only when SPI-based (four-wire) communication is applied. I^2C is a two wire interface SDA SCK.

Pin 6 of the module (SDO) is left unconnected to set the I^2C address to 0x76 – the on-board resistor pulls the SDO pin low setting the address to 0x76.

To change the I^2C address to 0x77, connect pin 6 of the module (SDO) to Vcc which would typically be the 3.3V supply.

Pin 5 of the module (CSB) must be connected to Vcc to select the I^2C interface. This is already done by an on-board pull-up resistor, so pin 5 can be left disconnected when using the I^2C interface [29].

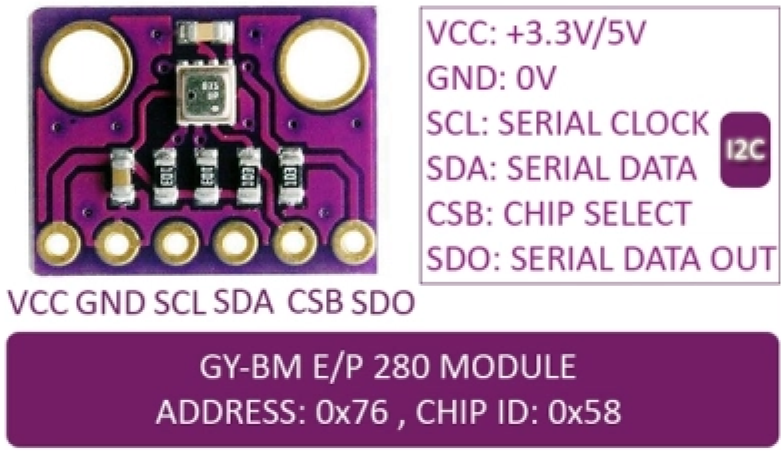


Fig 4: BMP280 Pin configuration

The Pin configuration of BMP280 pressure sensor is shown in Fig 4. The working of pressure sensor is governed by the following steps:

1. The working principle of BMP280 is based on weight of the air.
2. At higher temperatures, air is not as dense and heavy, so it applies lesser pressure on the sensor.
3. At lower temperatures, air is denser and weighs more, so it exerts more pressure on the sensor.
4. The sensor uses real-time temperature measurement to compensate the pressure readings for changes in air density.

The technical specifications of barometric pressure sensor are listed below:

1. Model: BMP280 -3.3
2. Chip: BMP280
3. Power supply: 3V/3.3V DC
4. Peak current: 1.12mA
5. Air pressure range : 300-1100hPa (equivalent to +9000m to -500m above sea level)

6. Temperature range: -40 °C to +85 °C
7. Digital interfaces: I²C (up to 3.4 MHz) and SPI (3 and 4 wire, up to 10 MHz)
8. Current consumption of sensor BMP280: 2.7µA @ 1 Hz sampling rate
9. Pin pitch: 2.54mm
10. Module size: 11.5mm*15mm [30]

The pin configuration of BMP280 pressure sensor is given in Table 2.

Table 2: Pin configuration of BMP280

S.NO.	PIN NAME	DESCRIPTION
1.	Vcc	Power source of 3.3VDC
2.	GND	Ground
3.	SLC	Serial Clock
4.	SDA	Serial Data
5.	CSB	CSB pin to GND to have SPI and to V_{cc} (3.3V) for I2C It's an input to the chip
6.	SDO	**S**erial **D**ata **O**ut / **M**aster **I**n **S**lave **O**ut pin, for data se from the BMP280 to NodeMCU

The pressure sensor is used for numerous environmental and research applications. Some of the most used sectors of applications are listed below:

1. Enhancement of GPS navigation (e.g. time-to-first-fix improvement, dead-reckoning, slope detection)
2. Indoor navigation (floor detection, elevator detection)
3. Outdoor navigation, leisure and sports applications
4. Weather forecast, Home weather stations
5. Health care application
6. Vertical velocity indicator (e.g. risk/sink speed)
7. Handsets such as mobile phones, tablet PCs, GPS devices
8. Botany management [29]

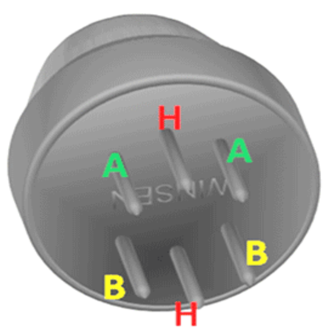
3.1.3 AIR QUALITY INDEX SENSOR

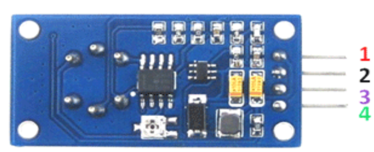
Air Quality Index (AQI) is used by government agencies to communicate information pertaining to air quality to the public h Public health risks increase as the AQI rises. Different countries have their own air quality indices, corresponding to different national air quality standards [34]. When air quality is good, the air is clear and contains only small amounts of solid particles and chemical pollutants whereas poor air quality presence of high levels of pollutants, it is often hazy and dangerous to health and the environment. Air quality is described according to the Air Quality Index (AQI), which is based on the concentration of pollutants present in the air at a particular location [35]. Air quality is an additional feature of the weather monitoring system apart from weather parameters. In this research, AQI is added considering public health risks, MQ-135 gas sensor is used to measure them.

The MQ-135 Gas sensors are suitable for detecting toxic gases namely NH3, NOx, Alcohol, Benzene, Smoke, and CO_2. It comes with a Digital Pin which makes this sensor to operate even without a microcontroller. The analog pin is TTL driven and works on 5V and so can be used with most common microcontrollers [36].

The key features of MQ135 is compared to other gas sensors are listed below:

1. Wide detecting scope
2. Fast response and High sensitivity
3. Stable and long life
4. Operating Voltage is +5V
5. Detect/Measure NH3, NOx, alcohol, Benzene, smoke, CO2, etc
6. Good sensitivity to harmful gases in a wide range.
7. It has a long life and low cost.
8. Possesses high sensitivity to ammonia, benzene, sulfide gases.
9. It features a simple drive circuit [36]





Pin No.	Pin Name
1	Vcc(+5V)
2	Ground
3	Digital Out
4	Analog out

Fig 5: MQ135 Pin configuration

The Vcc, Gnd and analog pins of the MQ135 gas sensor are shown in Fig 5. It consists of a tin dioxide (SnO2), a perspective layer inside aluminum oxide microtubes (measuring electrodes), and a heating element inside a tubular casing. The end face of the sensor is enclosed by a stainless steel net and the backside holds the connection terminals.The circuit diagram and the connection arrangement of an MQ 135 alcohol are shown in Fig 6 [36].

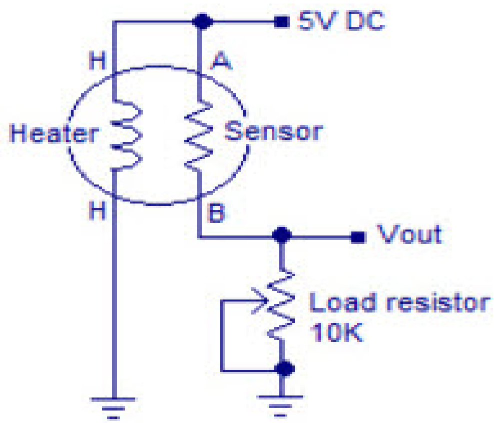


Fig 6: Connection arrangement of MQ135

The gas sensor layer of the sensor unit is made up of tin dioxide (SnO_2); it has lower conductivity compared to clean air and due to air pollution the conductivity increases. The air quality sensor detects ammonia, nitrogen oxide, smoke, CO_2, and other harmful gases. The air quality sensor has a small potentiometer that permits the adjustment of the load resistance of the sensor circuit. The 5V power supply is used for air quality sensor. The A,B and H pins of MQ135 are shown in Fig 7 [36].

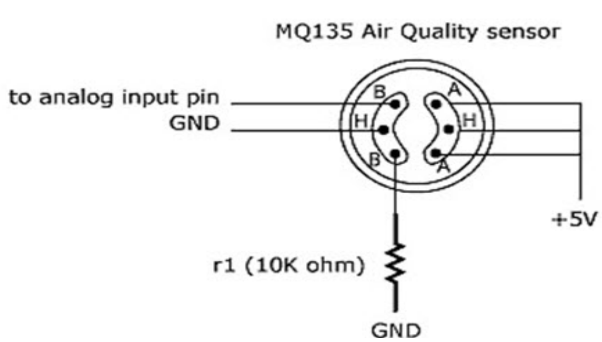


Fig 7: Internal details of MQ135

The air quality sensor has two outputs: analog output and TTL output. The TTL output is a low signal light which can be accessed through the IO ports on the microcontroller. The analog output is a concentration, i.e. increasing voltage is directly proportional to increasing concentration. This sensor has a long life and reliable stability as well [36]. The PPM measurements are done using the MQ135 sensor as illustrated below.

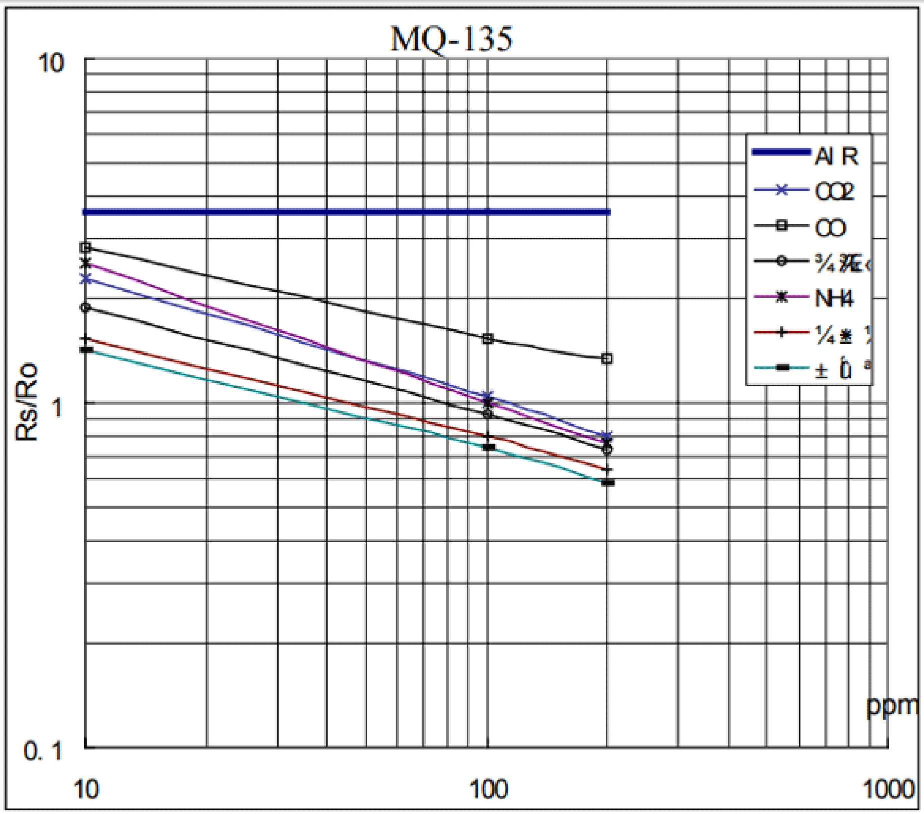


Fig 8: (Rs/Ro) v/s PPM graph of MQ135

It shows the typical sensitivity characteristics as illustrated in Fig 8. Let Ro be the sensor resistance at 100ppm of NH_3 in the clean air, Rs be the sensor resistance at various concentrations of gases.

The value of Ro indicates resistance in fresh air while value of Rs indicates resistance in gas concentration. The sensor should first be calibrated by finding the values of Ro in fresh air and then the value is used to find Rs using the below formula [36].

$$\textbf{Resistance of sensor}(R_s) = (V_{cc}/V_{RL}-1) \times R_L\text{-----} \ (3.4)$$

Once Rs and Ro values are calculated, the ratio is determined. Then using Fig 8, the equivalent value of PPM for that particular gas is computed. The pin configurations of gas sensor and module are given in Table 3.

Table 3: Pin configuration of MQ135

PIN NO.	PIN NAME	PIN DESCRIPTION
1	Vcc	Used to power the sensor, Generally the operating voltage is +5V.
2	Ground	Used to connect the module to system ground.
3	Digital Out	This sensor can also be used to get digital output from this pin, by setting a threshold value using the potentiometer.
4	Analog Out	This pin outputs 0-5V analog voltage based on the intensity of the gas.

Besides,the sensor consists of H-Pins, A-Pins, B-Pins which are internally connected where A pins and B pins are interchangeable. One pin will act as output while the other will be pulled to ground.

The MQ135 gas sensor is used for a variety of pollution monitoring applications. The major applications of gas sensor are listed below:

1. Used to detect leakage/excess of gases like Ammonia, nitrogen oxide, alcohols, aromatic compounds, sulfide and smoke.
2. Air quality monitor
3. Detection of harmful gases
4. Industrial air pollution detection [36]

3.1.4 RAINFALL SENSOR

Raindrop Sensor is used for measuring percentage of rainfall. It consists of two modules: a rain detection board (RDB) that detects rain and a control module, which compares the analog value, and converts it to a digital value[31] .

Rainfall detection is an important aspect of any weather monitoring system. Based on the frequency of rainfall in an area on a daily basis, rainfall prediction can be performed using machine learning or deep learning models. The FC-37 rain sensor module [37] is used in this research due to its advantages:

1. It has a separate control module and rain detection board (RDB) The RDB can be kept outside in the environment to detect rain.
2. The FC-37 rain sensor has both analog and digital pins
3. It has very high response to rain detection and thus the rain status can be updated in real-time on various cloud platforms

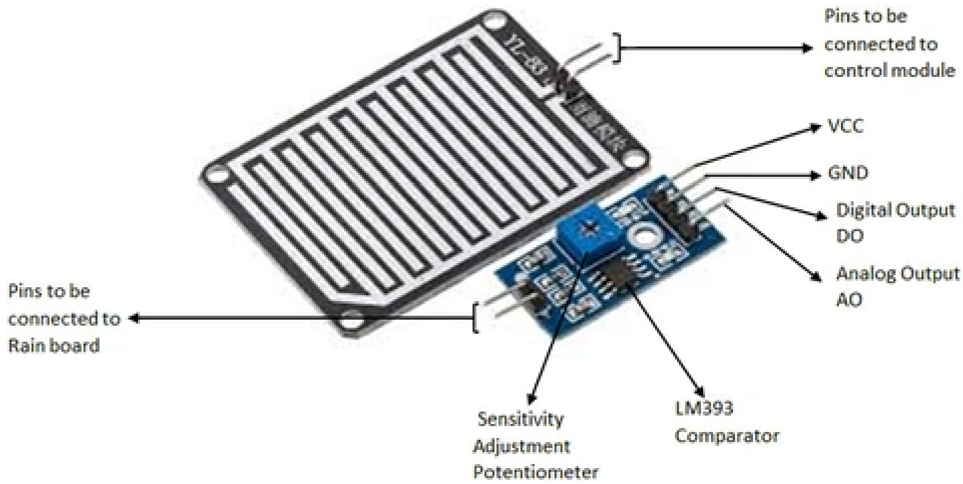


Fig 9: Rain sensor FC37 pin configuration

The pin configuration for the control module and RDB are shown in Fig 9. The control module of the raindrop sensor has 4 outputs. VCC is connected to a 5V supply. The GND pin of the module is connected to the ground. The D0 pin is connected to the digital pin of the microcontroller for digital output or the analog pin can be used. To use the analog output, the A0 pin can be connected to the ADC pin of a microcontroller. The sensor module consists of a potentiometer, LN393 comparator, LEDs, capacitors and resistors [31].

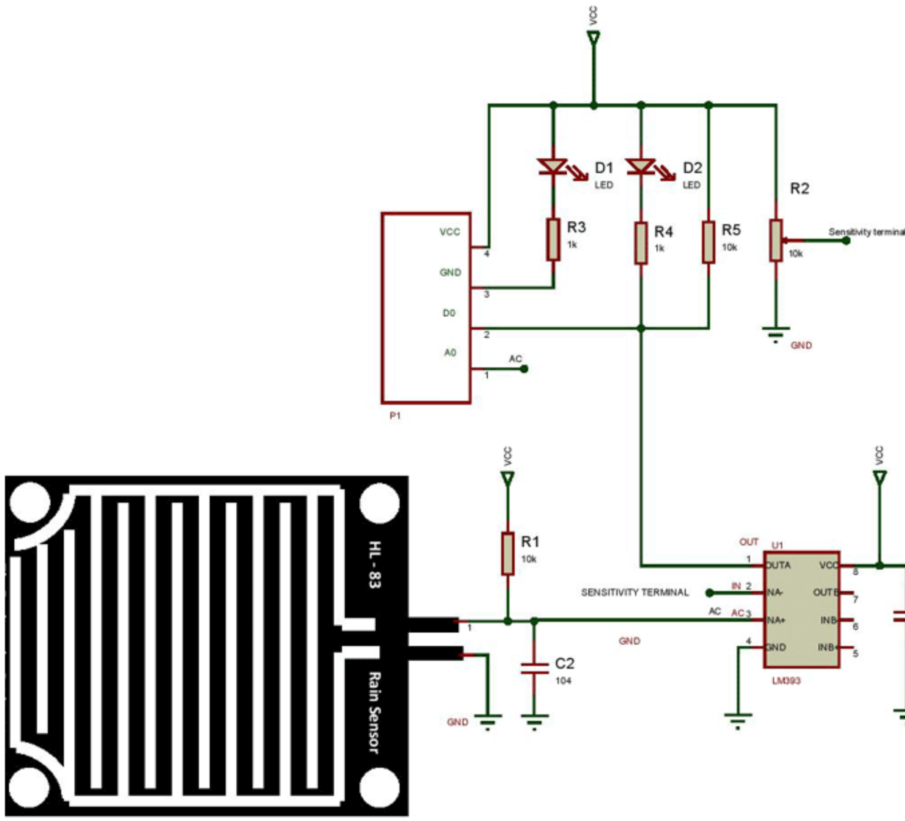


Fig 10: Internal details of FC37 rain sensor

Fig 10 shows the description of FC37 rain sensor. Here the R1 resistor and the RDB acts as a voltage divider. Capacitors C1 and C2 are used as a biasing element. The input for the Non-inverting terminal is taken from the connection point of the R1, and RDB. Another lead is taken from this connection and connected to the A0 terminal of the control module.

The input to the inverting terminal of the LM393 is taken from the potentiometer (R2). The R2 resistor acts as a voltage divider, and by varying R2, the input voltage can be varied to the inverting terminal, which in turn affects the sensitivity of the control module. The connections are shown in Fig 10. The resistors R3 and R4 act as current limiting resistors, while resistor R5 will act as a pull-up resistor to keep the bus in a high state when not in use[31] .

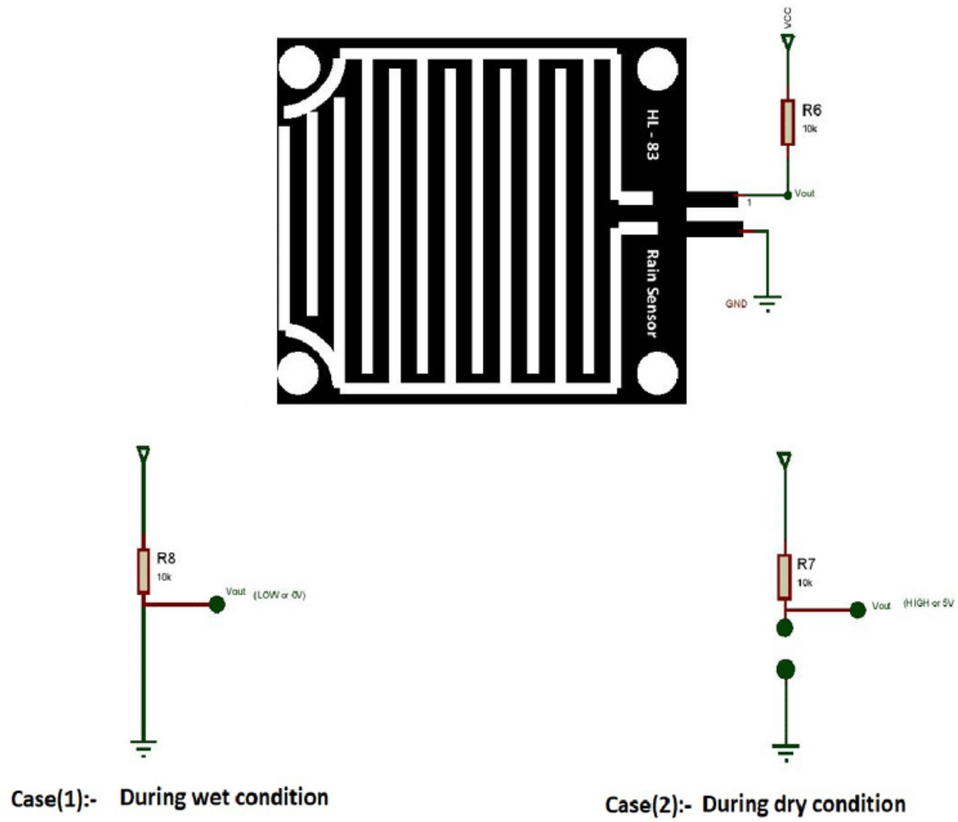


Fig 11:Rain sensor modes of operation

The modes of operation during wet and dry conditions are shown in Fig 11. The two cases of operational amplifier (OpAmp) operation are discussed below [31].

Case1: When the input of the inverting terminal is higher than the input of the non-inverting terminal.

Case2: If the input of the inverting terminal is lower than the input of non-inverting terminal.

The input to the inverting terminal is set to a certain value by varying the potentiometer and the sensitivity is set. When the surface of RDB is exposed to rainwater, the surface of the rain board module will be wet, and it offers minimum resistance to the supply voltage.

Due to this, the minimum voltage will be appearing at the non-inverting terminal of LM393 OpAmp. The comparator compares both inverting and non-inverting terminal voltages. If the condition falls under case(1), the output of the Op-Amp will be digital LOW. If the condition falls under case(2), the output of the Op-Amp will be digital HIGH [31].

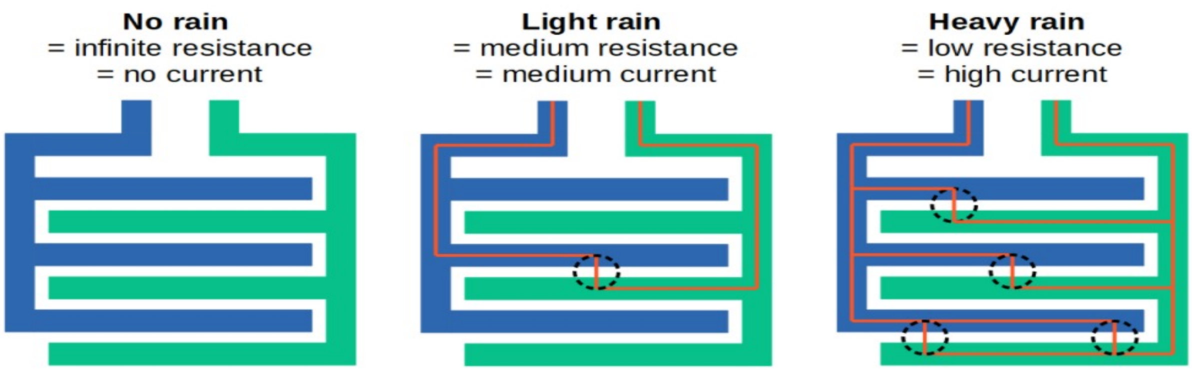


Fig 12: Rain sensor resistance variation

Basically, the resistance of the collector board varies accordingly to the amount of water on its surface. Fig 12 illustrates the change in resistance with respect to the amount of water present in the rain board. When the RDB is wet, the resistance increases, and the output voltage decreases whereas when the RDB is dry, the resistance is lower, and thus the output voltage is higher. This sensor module uses good quality double-sided material. It has anti-conductivity & oxidation with long time use. The area of this sensor includes 5cm x 4cm and can be built with a nickel plate on the side. The sensitivity can be adjusted by a potentiometer

The pin configuration of FC37 rain sensor is given in table 4.

Table 4: Pin configuration of FC37 rain sensor

S.NO	PIN NAME	PIN FUNCTION
1	VCC	Connects supply voltage- 5V
2	GND	Connected to ground
3	D0	Digital pin to get digital output
4	A0	Analog pin to get analog output

The technical specifications of FC37 rain sensor are listed below [31].

1. Working voltage 5V
2. Output format: Digital switching output (0 and 1), and analog voltage output AO
3. Potentiometer adjust the sensitivity
4. Uses a wide voltage LM393 comparator
5. Comparator output signal clean waveform is good, driving ability, over 15mA
6. Anti-oxidation, anti-conductivity, with long use time
7. With bolt holes for easy installation
8. Small board PCB size: 3.2cm x 1.4cm

The main application of rain sensor is detection of rain drops and also the frequency of occurrence of rain. Some of the other applications include Weather monitoring, Automatic windshield wipers and smart Agriculture [31].

3.1.5 UV INDEX SENSOR

The ultraviolet (UV) index, is an international standard measurement of the strength of sunburn-producing ultraviolet (UV) radiation at a particular place and time [38]. It is primarily used in daily forecasts aimed at the general public, and is increasingly available as an hourly forecast as well. The UV index is designed as an open-ended linear scale, directly proportional to the intensity of UV radiation that causes sunburn on human skin. The purpose of the UV index is to help people effectively protect themselves from UV radiation, which has health benefits in moderation but in excess causes sunburn, skin aging, DNA damage, skin cancer, immunosuppression, and eye damage as cataracts [38].

Thus UV Index is an important parameter regarding weather as it is directly involved with UV radiation from the sun. Thus to measure UV Index parameter, the GY/SI1145 sensor is used for the following reasons:

1. Proximity detection adjustable from under 1 cm to over 50 cm
2. Three independent LED drivers 15 current settings from 5.6 mA to 360 mA for each LED driver 25.6 µs LED driver pulse width 50 cm proximity range with single pulse (<3 Klaxon) 15 cm proximity range with single pulse (>3 Klaxon)
3. Operates at up to 128 Klaxon (direct sunlight)
4. High EMI immunity without shielded packaging

The Si1145/46/47 is an active optical reflectance proximity detector, UV index, and ambient light sensor whose operational state is controlled through registers accessible through the I^2C interface. The host can command the Si1145/46/47 to initiate on-demand proximity detection or ambient light sensing. The host can also place the Si1145/46/47 in an autonomous operational state where it performs measurements at set intervals and interrupts the host after each measurement is completed. Fig 13 shows the internal structure of the Si1145 UV sensor [33].

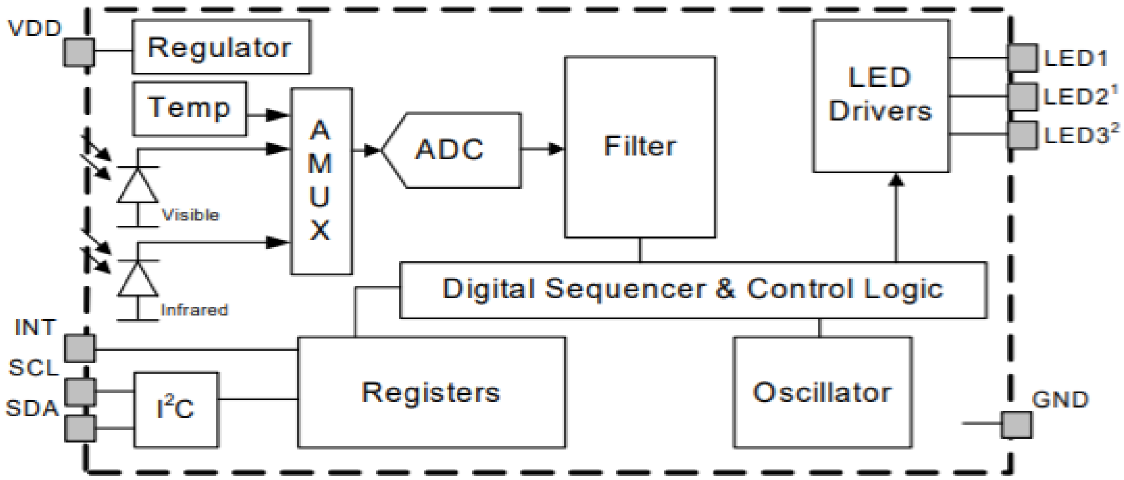


Fig 13: UV sensor structure

The Si1145/46/47 is a low-power, reflectance-based, infrared proximity, ultraviolet (UV) index, and ambient light sensor with I^{2}C digital interface and programmable event interrupt output. This touchless sensor IC includes an analog-to-digital converter, integrated high-sensitivity visible and infrared photodiodes, digital signal processor, and one, two, or three integrated infrared LED drivers with 15 selectable drive levels. The Si1145/46/47 offers excellent performance under a wide dynamic range and a variety of light sources including direct sunlight. The Si1145/46/ 47 can also work under dark glass covers. The photodiode response and associated digital conversion circuitry provide excellent immunity to artificial light flicker noise and natural light flutter noise. With two or more LEDs, the Si1146/47 is capable of supporting multiple-axis proximity motion detection. The Si1145/46/47 devices are provided in a 10-lead 2x2 mm package and are capable of operation from 1.71 to 3.6 V over the –40 to +85 °C temperature range [33].

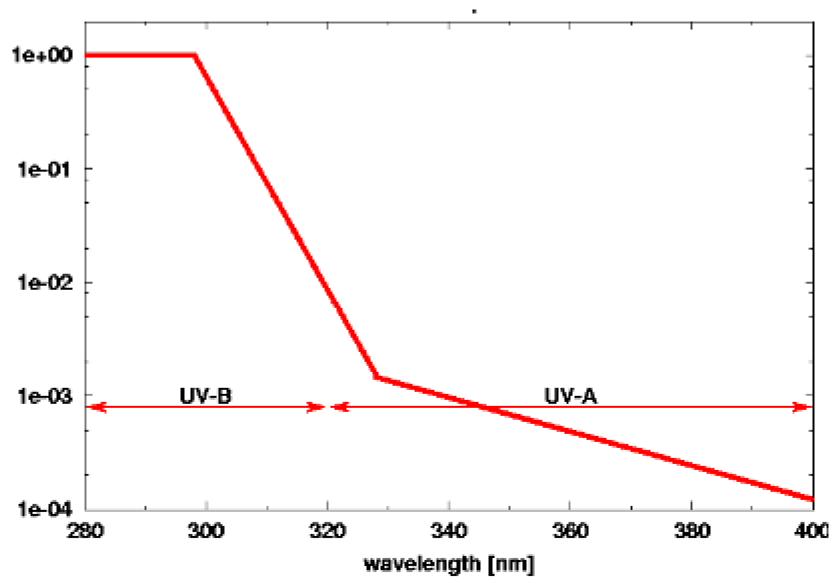


Fig 14: CIE Erythemal Action Spectrum

The UV Index is a number linearly related to the intensity of sunlight reaching the earth and is weighted according to the CIE Erythemal Action Spectrum [33] as presented in Fig 14. This weighting is a standardized measure of human skin's response to different wavelengths of sunlight from UV-B to UV-A. The UV Index has been standardized by the World Health Organization (WHO) and includes a simplified consumer UV exposure level as shown in Fig 15.

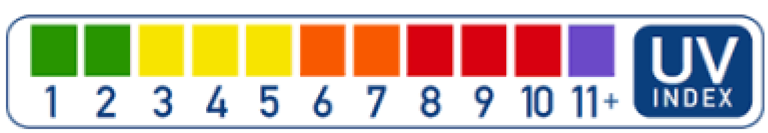


Fig 15: UV Index scale standardized by WHO

The Si1145/46/47 has photodiodes capable of measuring both visible and infrared light. However, the visible photodiode is also influenced by infrared light. The measurement of illuminance requires the same spectral response as the human eye. If an accurate luminous intensity measurement is desired, the extra IR response of the visible-light photodiode must be compensated. Therefore, to allow corrections to the infrared light influence, the Si1145/46/47 reports the infrared light measurement on a separate channel [33].

The pin configuration of the SI1145 UV sensor is given in table 5.

Table 5: Pin configuration of SI1145 UV sensor

S.NO	PIN NAME	PIN DESCRIPTION
1.	Vin	Connects supply voltage -3.3V
2.	GND	Connected to ground
3.	SCL	I2C Clock Pin to microcontroller
4.	SDA	I2C data pin to microcontroller

Apart from measuring UV index and ambient light intensity, the SI1145 UV sensor is used in Handsets, portable consumer electronics, security panels, tamper detection circuits and audio products.

3.1.6 GPS SENSOR MODULE:

The Global Positioning System (GPS), is a satellite-based radio navigation system owned by the United States government and operated by the United States Space Force. It is one of the Global Navigation Satellite Systems (GNSS) that provides geolocation and time information to a GPS receiver anywhere on or near the Earth where there is an unobstructed line of sight to four or more GPS satellites. Obstacles such as mountains and buildings block the relatively weak GPS signals.

The GPS does not require the user to transmit any data, and it operates independently of any telephonic or internet reception, though these technologies can enhance the usefulness of the GPS positioning information. The GPS provides critical positioning capabilities to military, civil, and commercial users around the world. The United States government created the system, maintains it, and makes it freely accessible to anyone with a GPS receiver [39].

GPS sensor is included in this research to obtain localized information. This information will be more useful in weather prediction For this purpose the NEO-6MV2 GPS module is utilized in the developed work.

Significant features of NEW-6MV2 GPS sensor:

1. Maximum navigation update rate: 5Hz
2. Default baud rate: 9600bps
3. EEPROM with battery backup
4. Sensitivity: -160dBm
5. Supply voltage: 3.6V
6. Maximum DC current at any output: 10mA
7. Operation limits: Gravity-4g, Altitude-50000m, Velocity-500m/s
8. Operating temperature range: -40ºC TO 85°C

The NEO-6MV2 is a GPS module used for navigation. The module checks its location on earth and provides output data which is longitude and latitude of its position. It is from a family of stand-alone GPS receivers featuring high performance u-blox 6 positioning engine. These flexible and cost effective receivers offer numerous connectivity options in a miniature (16 x 12.2 x 2.4 mm) package. The compact architecture, power and memory options makes it ideal for battery operated mobile devices. The innovative design of NEO-6MV2 provides excellent navigation performance under most challenging environments [37].

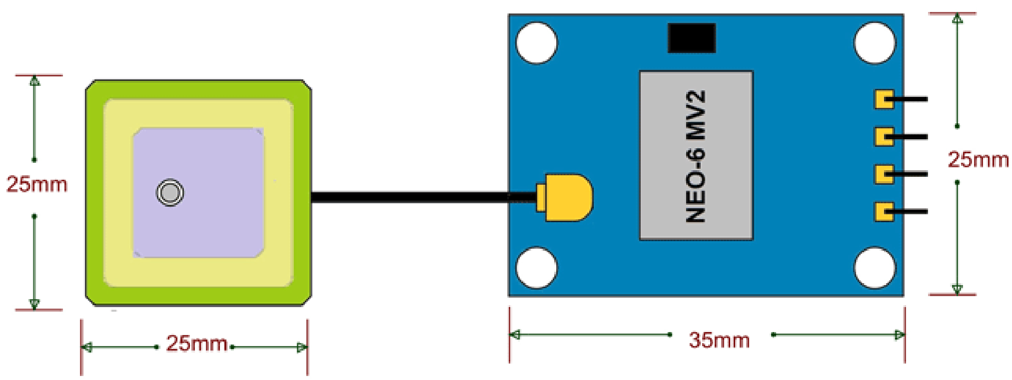


Fig 16: GPS module connections

Fig 16 shows the connection of the GPS module to the antenna. The GPS module works by figuring out how far it is from a number of satellites. The NEO-6MV2 module is pre-programmed to know where the GPS satellites are at any given time. The satellites transmit information about their position and the current time in the form of radio signals towards the Earth. These signals identify the satellites and tell the receiver where they are located [40].

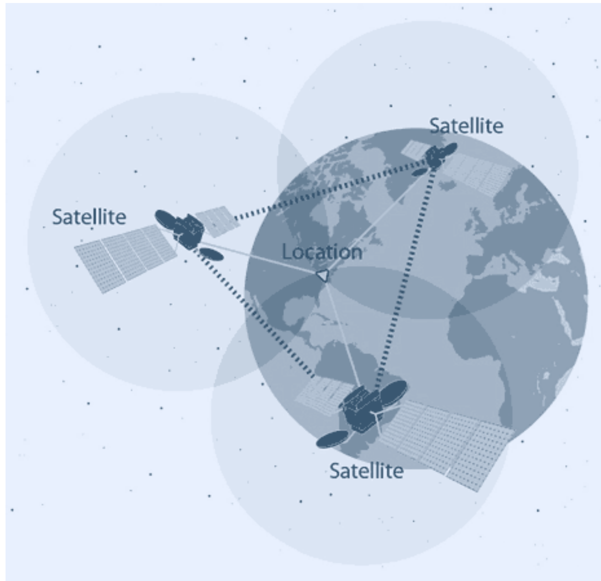


Fig 17: GPS module working

Fig 17 shows the working of a GPS module to detect exact latitude and longitude of a particular location. The module then calculates how far away each satellite is by figuring out how long it took for the signals to arrive. Once it has information on how far away at least three satellites are and where they are in space, it can pinpoint the location on Earth. This process is known as Trilateration [40]. The pin configurations for the GPS module are given in table 6.

Table 6: Pin configuration of GPS module

S.NO	PIN NAME	PIN DESCRIPTION
1	VCC	Positive power pin
2	RX	UART receive pin
3	TX	UART transmit pin
4	GND	Ground

The main application of the GPS module is to detect real-time latitude and longitude location of the weather monitoring setup. Other applications include:

1. GPS application
2. Smartphone and tablets
3. Navigation systems
4. Drones
5. Hobby projects

3.2 DATA PROCESSING

3.2.1 NODEMCU MICROCONTROLLER

The NodeMCU is a microcontroller unit with open-source software and hardware development environment built around an inexpensive System-on-a-Chip (SoC) module named ESP8266 [41]. The ESP8266 is designed and manufactured by Espressif Systems. The crucial elements of this module are: CPU, RAM, WiFi and modern operating system and SDK. These features make it an excellent choice for the Internet of Things (IoT) based applications. The reason for opting NodeMCU as a microcontroller for this research is as follows:

1. Compared to other microcontrollers such as Arduino UNO , Arduino mega or the Raspberry pi, the NodeMCU is best suited for IoT based applications since these applications require less processing power and low storage as data gets stored in the cloud.
2. Another main advantage of using NodeMCU is that it has an inbuilt wifi module i.e., ESP8266 which enables it to connect it to the internet without any external modules whereas in microcontrollers like Arduino UNO, an external ESP8266 wifi

module needs to be interfaced with the microcontroller.

3. Since the weather monitoring system is solar powered, it is essential that the microcontroller and sensors used must have high efficiency (use low power), thus using microcontrollers like Raspberry pi would result in decrease in efficiency of the system.

The algorithm for data sensing/ data collection, weather data display and uploading weather parameters in IoT cloud is explained below:

1. Include libraries for all sensors, GPS and LCD from the Arduino library manager
2. Initialize variables for all weather parameters and I2C addresses for sensors and GPS module
3. Create ssid and password for wireless internet connection to connect to the local area network
4. Define Adafruit IO and Thingspeak user credentials and create new feeds for each weather parameter
5. Start reading parameters from 1 to n, with a delay of 5 sec and display the measured parameter in the LCD using the command "lcd.print(weather_parameter)"in a loop format
6. Upload the weather parameters in Adafruit cloud by using the command "feed_name.publish(weather_parameter)" with a delay of one second between each parameter
7. Upload the parameters in Thingspeak cloud using the command "Thingspeak.setfield(channel_no , weather_parameter)" with a delay of two seconds between each parameter

8. Call the function gps() after each step to update time, date, latitude and longitude of the weather monitoring system in a HTML webpage in real-time
9. Repeat the steps 5 to 8 using a scheduling mechanism with a delay of 8 seconds for continuous monitoring of weather parameters in IoT cloud and to display the parameters in an LCD

The NodeMCU ESP8266 development board comes with the ESP-12E module containing ESP8266 chip having Tensilica Xtensa 32-bit LX106 RISC microprocessor. This microprocessor supports RTOS and operates at 80MHz to 160 MHz adjustable clock frequency. NodeMCU has 128 KB RAM and 4MB of Flash memory to store data and programs. Its high processing power with in-built Wi-Fi / Bluetooth and Deep Sleep Operating features make it ideal for IoT projects. NodeMCU can be powered using Micro USB jack and VIN pin (External Supply Pin). It supports UART, SPI, and I2C interface [41]. NodeMCU board configuration is as shown in Fig 18.

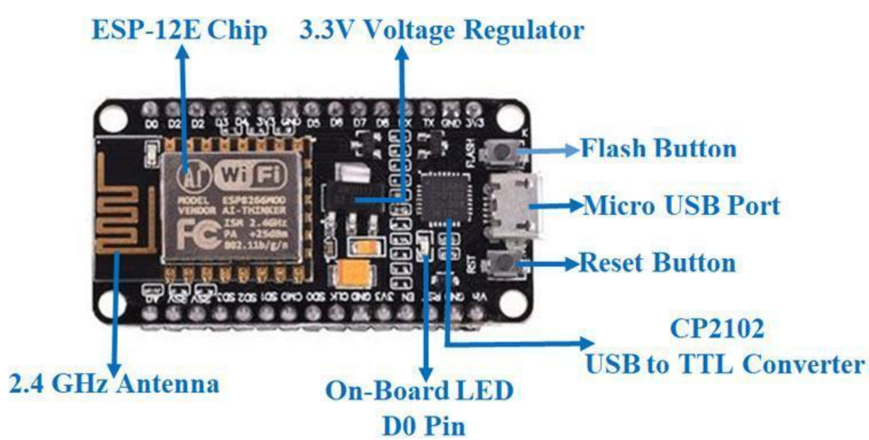


Fig 18: NodeMCU Board Configuration

USB to TTL converter is used to provide connectivity between USB and serial UART interface. The Arduino research project created an open-source hardware design and software SDK for their versatile IoT controller. Similar to NodeMCU, the Arduino hardware is a microcontroller board with a USB connector, LED lights, and standard data pins. It also defines standard interfaces to interact with sensors or other boards. But unlike NodeMCU, the Arduino board can have different types of CPU chips (typically an ARM or Intel x86 chip) with memory chips, and a variety of programming environments. There is an Arduino reference design for the ESP8266 chip as well. However, the flexibility of Arduino also means significant variations across different vendors. For example, most Arduino boards do not have WiFi capabilities, and some even have a serial data port instead of a USB port [42]. Pin configuration of NodeMCU is shown in Fig 19.

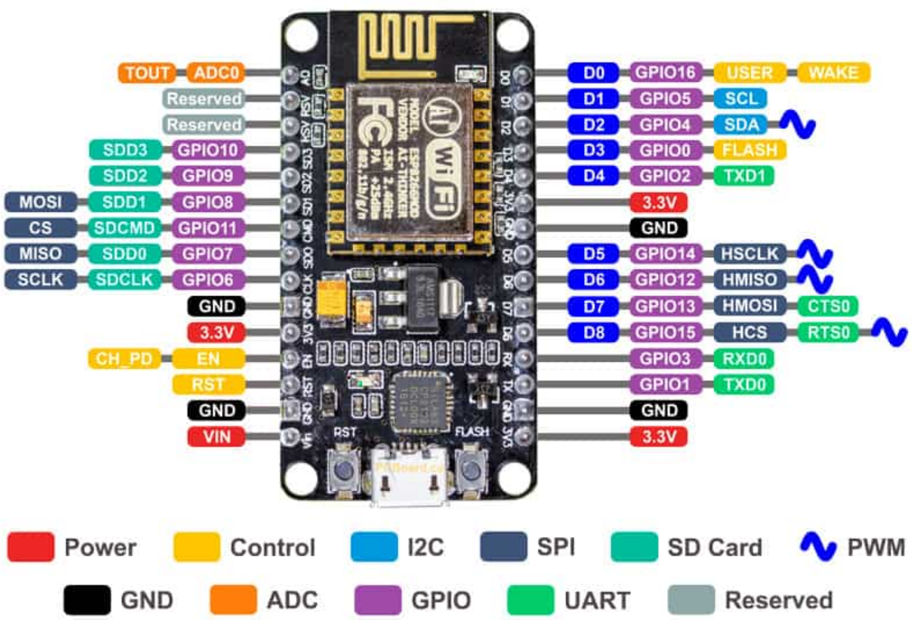


Fig 19: NodeMCU Pin Configuration

There are four power pins. VIN pin and three 3.3V pins.

1. VIN can be used to directly supply the NodeMCU/ESP8266 and its peripherals. Power delivered on VIN is regulated through the onboard regulator on the NodeMCU module – 5V regulated input to the VIN pin can also be supplied
2. 3.3V pins are the output of the onboard voltage regulator and can be used to supply power to external components.

GND: The ground pins of NodeMCU/ESP8266. I2C Pins are used to connect I^2C sensors and peripherals. Both I^2C Master and I^2C Slave are supported. I^2C interface functionality can be realized programmatically, and the clock frequency is 100 kHz at a maximum. It should be noted that I^2C clock frequency should be higher than the slowest clock frequency of the slave device.

GPIO Pins: NodeMCU/ESP8266 has 17 GPIO pins which can be assigned to functions such as I2C, I2S, UART, PWM, IR Remote Control, LED Light and Button programmatically. Each digital enabled GPIO can be configured to internal pull-up or pull-down, or set to high impedance. When configured as an input, it can also be set to edge-trigger or level-trigger to generate CPU interrupts.

ADC Channel: NodeMCU is embedded with a 10-bit precision SAR ADC. The two functions can be implemented using ADC. Testing power supply voltage of VDD3P3 pin and testing input voltage of TOUT pin. However, they cannot be implemented at the same time.

UART Pins: NodeMCU/ESP8266 has 2 UART interfaces (UART0 and UART1) which provide asynchronous communication (RS232 and RS485), and can communicate at up to 4.5 Mbps. UART0 (TXD0, RXD0, RST0 & CTS0 pins) can be used for communication.

However, UART1 (TXD1 pin) features only data transmit signals, so it is usually used for printing logs.

SPI Pins: NodeMCU/ESP8266 features two SPIs (SPI and HSPI) in slave and master modes. These SPIs also support the following general-purpose SPI features:

1. 4 timing modes of the SPI format transfer
2. Up to 80 MHz and the divided clocks of 80 MHz
3. Up to 64-Byte FIFO

SDIO Pins: NodeMCU/ESP8266 features Secure Digital Input/output Interface (SDIO) which is used to directly interface SD cards. 4-bit 25 MHz SDIO v1.1 and 4-bit 50 MHz SDIO v2.0 are supported.

PWM Pins: The board has 4 channels of Pulse Width Modulation (PWM). The PWM output can be implemented programmatically and used for driving digital motors and LEDs. PWM frequency range is adjustable from 1000 μs to 10000 μs (100 Hz and 1 kHz).

Control Pins are used to control the NodeMCU/ESP8266. These pins include Chip Enable pin (EN), Reset pin (RST) and WAKE pin.

1. EN: The ESP8266 chip is enabled when EN pin is pulled HIGH. When pulled LOW the chip works at minimum power.
2. RST: RST pin is used to reset the ESP8266 chip.
3. WAKE: Wake pin is used to wake the chip from deep-sleep.

Control Pins are used to control the NodeMCU/ESP8266. These pins include Chip Enable pin (EN), Reset pin (RST) and WAKE pins

1. EN: The ESP8266 chip is enabled when EN pin is pulled HIGH. When pulled LOW the chip works at minimum power.
2. RST: RST pin is used to reset the ESP8266 chip.
3. WAKE: Wake pin is used to wake the chip from deep-sleep [42]

Pin configuration of NodeMCU is tabulated in Table 7.

Table 7: Pin configuration of NodeMCU

PIN CATEGORY	PIN NAME	PIN DESCRIPTION
Power	Micro-USB, 3.3V, GND, Vin	Micro-USB: NodeMCU can be powered through t USB port 3.3V: Regulated 3.3V can be supplied to this pin t power the board
GPIO Pins	GPIO1 to GPIO16	NodeMCU has 16 general purpose input-output pi on its board
SPI Pins	SD1, CMD, SD0, CLK	NodeMCU has four pins available for SPI communication.
UART Pins	TXD0, RXD0, TXD2, RXD2	NodeMCU has two UART interfaces, UART0 (RXD0 & TXD0) and UART1 (RXD1 & TXD1). UART1 is used to upload the firmware/program.
I2C Pins		NodeMCU has I2C functionality support but pins must be found out in prior.

NodeMCU is a low power and cost effective microcontroller used for numerous applications due to its simple design. It is used in the proposed work due to low computational requirements and low power consumption. Some of the other applications include, prototyping of IoT devices, low power battery operated applications, network projects, projects requiring multiple I/O interfaces with Wi-Fi and bluetooth functionalities.

3.3 CLOUD ENABLEMENT

Cloud computing is the means of storing and accessing data over the Internet instead of using them locally. It allows sharing of resources across devices through centralized storage. The proposed I-SWFMS system uses Adafruit and Thingspeak IoT cloud databases to store time-series weather data in tabular databases. The databases can further be downloaded in JSON or CSV format which is used for weather prediction using machine learning algorithms. Features of cloud computing are as shown in Fig 20.

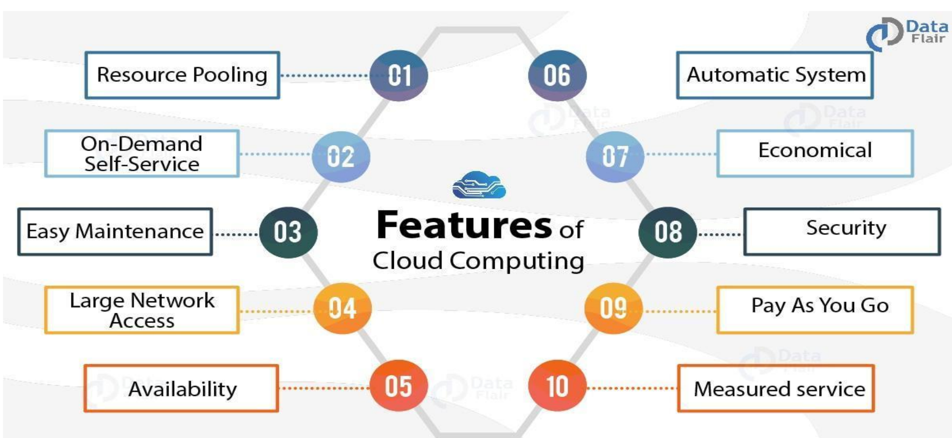


Fig 20: Features of Cloud Computing

Cloud computing can be classified into 3 types of services:

1. Infrastructure as a Service (IaaS) - Provides processing, storage, and physical resources for computing.
2. Platform as a Service (PaaS) - Created apps that can be deployed in the cloud.
3. Software as a Service (SaaS) - Third party business operations that can be deployed in the cloud.

3.3.1 ADAFRUIT IO

With the rise in digital transformations, IoT deployments in the cloud have become more popular. Deployment of IoT solutions on the cloud are performed to realize the following benefits for the considered applications.

- Cost - Reduces the cost of computing and storage by using various cloud services
- Scalability - The "pay-as-you-go" pricing model allows a flexible pay model, also allowing scalability of the application
- Data control - Data backup and recovery with high security
- Server uptime - Allows very minimum or no downtime, with high server availability

Adafruit IO is one such cloud provider focusing more on IoT deployments on the cloud. It supports different hardware namely Raspberry Pi, ESP2866, and Arduino.

Adafruit is preferred over other cloud services such as Amazon azure and Ubidots for the following reasons:

1. Powerful API - Provides libraries for various programming languages enabling built-in user interface support
2. Dashboard - Understanding data via charts and graphs enhances decision making process
3. Privacy - Data is secured in the cloud platform with better encryption algorithms
4. Documentation & Community - Many blogs with amazing community support allows continuous developments of the products [46].

Data on Adafruit IO is kept in time-series databases called **feeds**. Each feed contains time-stamped data points using which each weather parameter is updated as a data points [49].

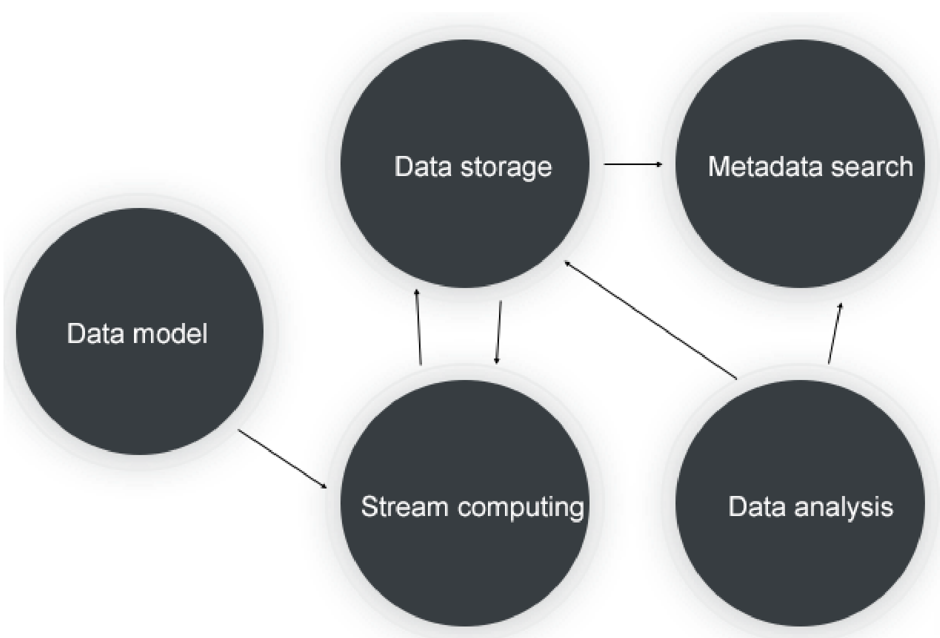


Fig 21: Time data series processing

The core procedure of time series data processing is illustrated in Fig 21:

1. Data model: For the standard definition of time series data, the collected time series data must conform to the definition of the model, including all the characteristic attributes of the time series data.
2. Stream computing: Pre-aggregation, down sampling, and post-aggregation for the time series data.
3. Data storage: The storage system provides high-throughput, massive volume, and low-cost storage, and supports separation of cold/hot data, as well as efficient range query.
4. Metadata retrieval: Provides the storage and retrieval of timeline metadata in the order of tens of millions to hundreds of millions, and supports different retrieval methods (multidimensional filtering and location query).
5. Data analysis: Provides time series analysis and computing capabilities for time series data [49].

The time-series data collection of temperature feed of Adafruit cloud is shown in Fig 22.The columns feed_id and value shows the temperature feed to which the time-series data with unique id is uploaded to the feed in real-time.

	A	B	C	D
1	id	value	feed_id	created_at
2	0EP6QSZY8FA1V400AMYC17B6E3	29.7	1533500	2021-03-13 10:37:29 UTC
3	0EP6QTC2S0S5R9CJNRJJTVJYZW	29.2	1533500	2021-03-13 10:38:09 UTC
4	0EP6QW7MMYVDBB42FHCYSGYZYD	28.7	1533500	2021-03-13 10:41:24 UTC
5	0EP6RERVC5HD5XEEZWJQ0S7PNR	28.1	1533500	2021-03-13 11:13:48 UTC
6	0EP6VE8TKBMTZFWDK4BGG3CW7A	30.2	1533500	2021-03-13 14:00:42 UTC
7	0EP6VF5565X18VSKWZV0RVS0CB	30	1533500	2021-03-13 14:02:15 UTC
8	0EP6VFH49PJ6YH51XG0379VD89	29.9	1533500	2021-03-13 14:02:54 UTC
9	0EP6VGFWSJGS0P33BJ31QCHG7S	29.7	1533500	2021-03-13 14:04:35 UTC
10	0EP6VGVQ2X4X2XNBYF8HFBAYVS	29.8	1533500	2021-03-13 14:05:13 UTC
11	0EP6VHFRG3BQEDY4VAM1CZNN8T	29.8	1533500	2021-03-13 14:06:19 UTC
12	0EP6VHVKZYDX4RBMN91TEH0ZXV	29.7	1533500	2021-03-13 14:06:58 UTC
13	0EP6VJ7K14PHN9PWTWSD588NMV	29.5	1533500	2021-03-13 14:07:37 UTC
14	0EP6VK447CPBYKGRYV6MDP1STQ	29.4	1533500	2021-03-13 14:09:11 UTC
15	0EP6VKG1YKPQ9M5YT0PXDWV791	29.3	1533500	2021-03-13 14:09:50 UTC

Fig 22: Temperature feed of Adafruit cloud

Adafruit IO acts as MQTT broker and thus follows the publish-subscribe protocol. Subscribers can subscribe to the broker i.e, Adafruit IO cloud and receive weather data in real time. Thus not only the weather parameters are observed in Adafruit dashboard but also in a number of devices such as smartphones, tablets etc.

3.3.2 THINGSPEAK CLOUD

ThingSpeak is an open-source IoT application and API to store and retrieve data from things using the HTTP [45] and MQTT [45] 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. ThingSpeak is an IoT Cloud platform where the data from the sensor can be uploaded to the cloud. It is also used to analyze and visualize data with MATLAB . The ThingSpeak service is operated by MathWorks. It includes a Web Service (REST API) [45] that allows collection and storage of sensed data in the cloud to develop IoT based applications. It works with Arduino, Raspberry Pi and MATLAB (premade libraries and APIs exist), but it should work with all kinds of Programming Languages, since it uses a REST API and HTTP [45].

Thingspeak framework is shown in Fig 23.

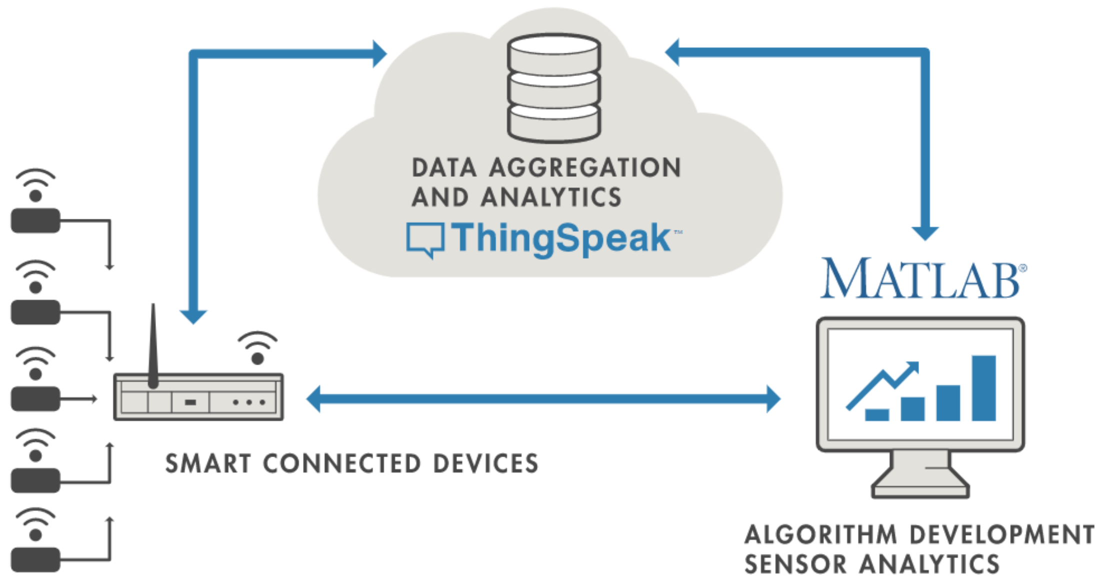


Fig 23: Thingspeak Cloud Framework

The core element of ThingSpeak is a ThingSpeak Channel. A channel stores the data that is uploaded to ThingSpeak. It comprises of the following elements:

1. 8 fields for storing data of any type - These can be used to store the data from a sensor or from an embedded device.
2. 3 location fields - Can be used to store the latitude, longitude and the elevation. These are very useful for tracking a moving device or obtaining geolocalized data.
3. 1 status field - A short message to describe the data stored in the channel [11].

Similar to Adafruit IO cloud, Thingspeak also uses time-series databases for storing weather data in real-time. The main advantage of thingspeak is the addition of MATLAB functionality which enables to visualize weather parameters in graphical manner.

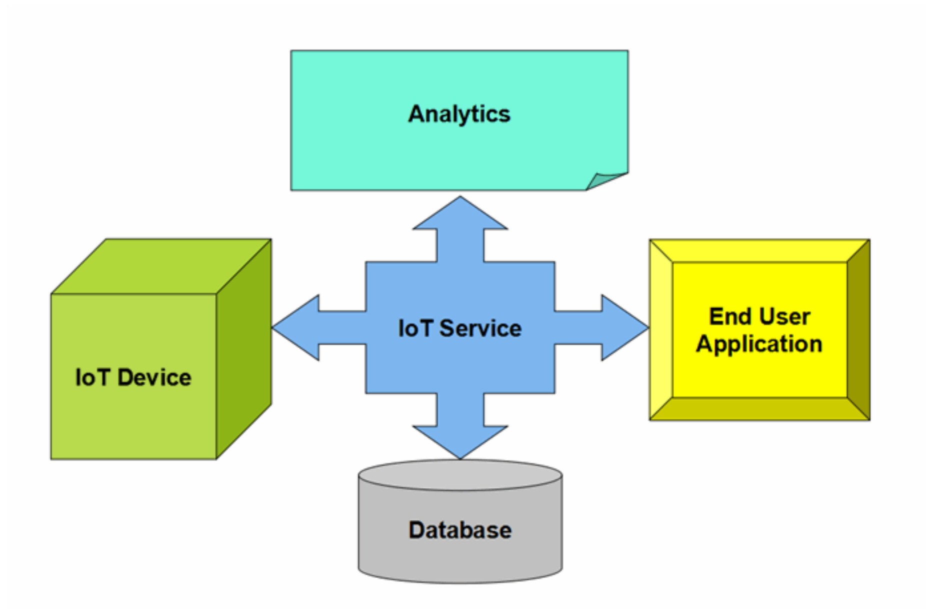


Fig 24: IoT Services offered by Thingspeak Cloud

Fig 24 shows the IoT services offered by Thingspeak cloud platform. The thingspeak IoT service enables data analytics for time-series data such as weather. The database gets updated from the IoT device which is the NodeMCU and after successful data analytics using MATLAB, the end user can visualize various weather parameters in a parameter vs. time graph.

3.3.3 MQTT APPLICATION

Message Queuing Telemetry Transport (MQTT) is a lightweight application-layer messaging protocol that works on the publish/subscribe (pub/sub)[17] model. HTTP is based on request-response model which is one of the reasons it is not preferred in IoT applications. In the pub/sub model, multiple clients (sensors) can be connected to a central server called a broker and subscribe to applications they are interested in. Clients can also publish messages to specific applications of their interest through the broker. The broker is a common interface for sensor devices to connect to and exchange data. Another important point to note about MQTT is that it utilizes TCP connection on the transport layer for connections between sensors and broker which makes the communication reliable. MQTT communication protocol is shown in Fig 25.

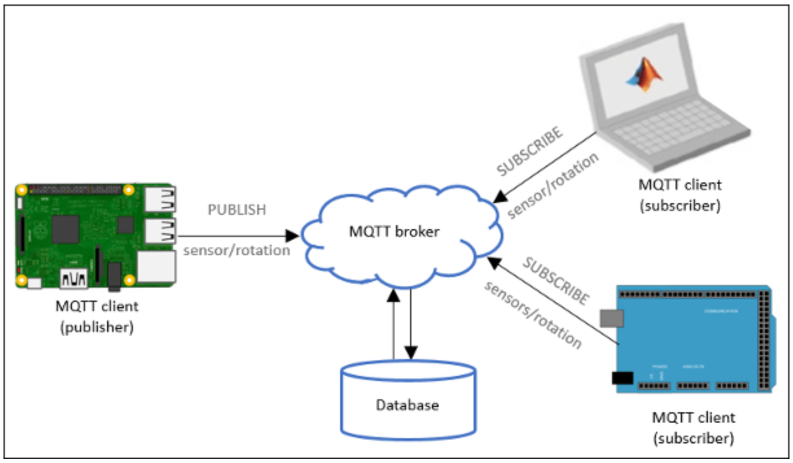


Fig 25: MQTT Communication Protocol

Messages in MQTT are always published on application of interest, which basically represents the destination address for that message. A client may subscribe as well as publish to multiple topics. Every client subscribed to a topic receives all the messages published to that topic. As a standard practice, topics should follow a hierarchy using a slash (/) as a separator. This allows for logical grouping/arrangement for a network of sensors. The high-level architecture of MQTT is shown in Fig 26 and explained below.

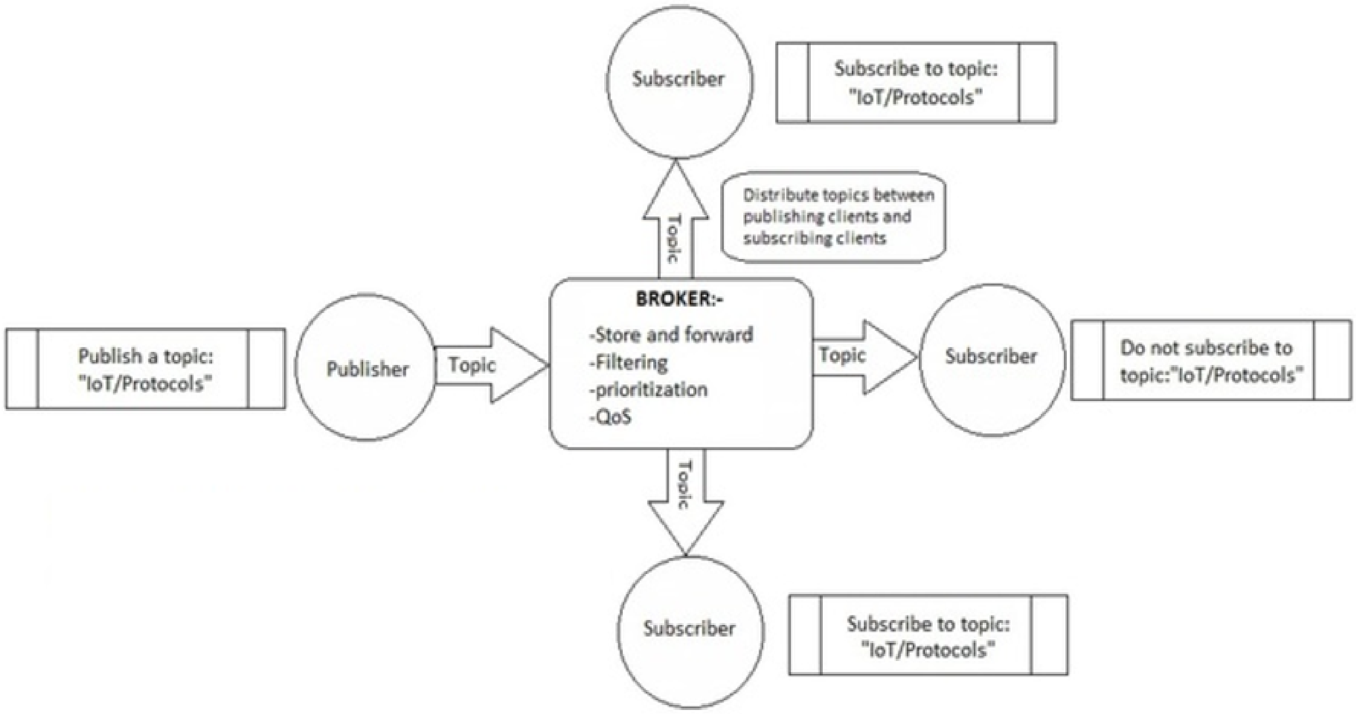


Fig 26: MQTT Architecture

Other than acting as a router for routing the messages, the broker has some additional responsibilities which are discussed briefly below:

1. Quality of Service: The Quality of Service (QoS) feature allows the MQTT protocol to provide additional messaging qualities of service depending on the requirement (severity level) of the data or application. The QoS feature ensures that the message in transit is delivered as required by the service. There are 3 QoS levels available namely: "Fire and forget" or 0, "Delivered at least once" or 1, "Delivered exactly once" or 2. Higher levels of QoS are more reliable, but involve higher latency and have higher bandwidth requirements. The application developers have to think

about what is the importance of the data being carried and choose this level accordingly.

2. Store and Forward: MQTT provides support for storing persistent messages on the broker. When publishing messages, clients may request that the broker persists the message. When a client subscribes to a topic, any persisted message will be sent to the client when this feature is used. Only the most recent persistent message is stored. However, unlike the traditional messaging queues, MQTT broker does not allow these persisted messages to back up inside the server.

3. Security: MQTT broker may require username and password authentication from clients to connect for security. To ensure privacy of messages in transit, the TCP connection may be encrypted with SSL/TLS.

MQTT is a many-to-many communication protocol for passing messages between multiple clients through a central broker. It decouples producer and consumer by letting clients publish and having the broker decide where to route and copy messages. While MQTT has some support for persistence, it does best as a communications bus for live data. MQTT clients make a long-lived outgoing TCP connection to a broker. This usually presents no problem for most of the IOT requirements. However, if need be, there are other protocols that rely on UDP for lower bandwidth and system resources.

3.4 WEATHER PREDICTION MODEL

Weather forecasting is performed by developing an appropriate machine learning model and training the data using a predefined dataset. After successful training of data, testing is performed using the dataset obtained from Adafruit IO cloud. For the basic four weather parameters viz. Temperature, pressure, humidity and altitude, a weather prediction is performed for the next 'n' days based on the number of time-series entries in the testing dataset.

Machine learning workflow refers to the series of stages or steps involved in the process of building a successful machine learning system [43].

The various stages involved in the machine learning workflow are listed below:

1. Data Collection
2. Data Preparation
3. Choosing Learning Algorithm
4. Training Model
5. Evaluating Model
6. Predictions

3.4.1 DATA COLLECTION

In this stage,

1. Data is collected from different sources.
2. The type of data collected
3. It depends upon the type of desired project.
4. Data may be collected from various sources such as files, databases etc.
5. The quality and quantity of gathered data directly affects the accuracy of the desired system.

3.4.2 DATA PREPARATION

In this stage,

1. Data preparation is done to clean the raw data.
2. Data collected from the real world is transformed to a clean dataset.

3. Raw data may contain missing values, inconsistent values, duplicate instances etc.
4. So, raw data cannot be directly used for building a model.

Different methods of cleaning the dataset are-

1. Ignoring the missing values
2. Removing instances having missing values from the dataset
3. Estimating the missing values of instances using mean, median or mode
4. Removing duplicate instances from the dataset
5. Normalizing the data in the dataset

This is the most time consuming stage in machine learning workflow.

3.4.3 CHOOSING LEARNING ALGORITHM

In this stage,

1. The best performing learning algorithm is researched.
2. The learning algorithm is chosen based on the number of features and speed of the training model.

3. It depends upon the type of problem that needs to be solved and the type of data available.

4. If the problem is to classify and the data is labeled.
5. Classification algorithms are used.
6. If the problem is to perform a regression task and the data is labeled, regression algorithms are used.
7. If the problem is to create clusters and the data is unlabeled, clustering algorithms are used.

The following chart in Fig 27 provides the overview of learning algorithms:

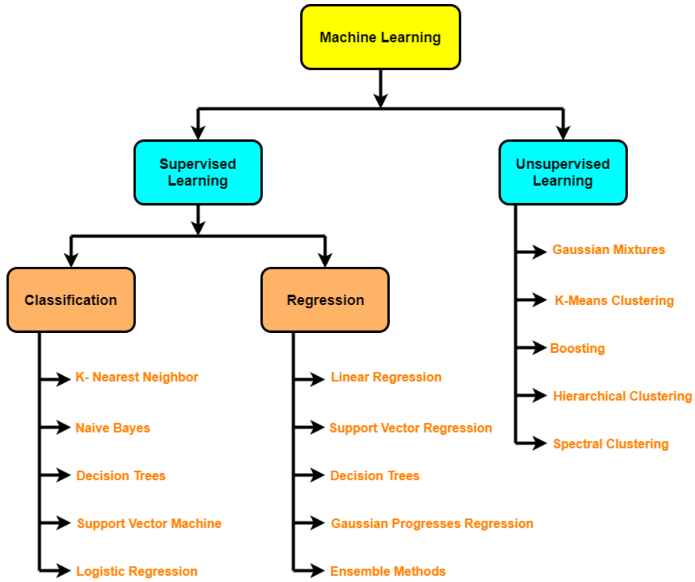


Fig 27: Machine learning algorithms chart

3.4.4 TRAINING MODEL

In this stage,

1. The model is trained to improve its ability.
2. The dataset is divided into training dataset and testing dataset.
3. The training and testing split is of the order 80/20 or 70/30.
4. Training dataset is used for training purposes.
5. Testing dataset is used for testing purposes and training dataset is fed to the learning algorithm.
6. The learning algorithm finds a mapping between the input and the output

3.4.5 EVALUATING MODEL

In this stage,

1. The model is evaluated to test if the model is any good.
2. The model is evaluated using the kept-aside testing dataset.
3. It allows testing the model against data that has never been used before for training.
4. Metrics such as accuracy, precision, recall etc are used to test the performance.
5. If the model does not perform well, the model is re-built using different hyper parameters.
6. The accuracy may be further improved by tuning the hyper parameters.

3.4.6 WEATHER PREDICTIONS

In this stage,

1. The built system is finally used to do something useful in the real world.
2. Here, the true value of machine learning is realized

3.4.7 LINEAR REGRESSION MODEL

Linear regression models are preferred to other machine learning models because of its simple prediction nature. It has the simplest form of regression formula with one dependent and one independent variable. Linear regression is a basic and commonly used type of predictive analysis. The regression estimates are used to explain the relationship between one dependent variable and one or more independent variables. The simplest form of the regression equation with one dependent and one independent variable is defined by the formula **y = c + b*x,** where y = estimated dependent variable score, c = constant, b = regression coefficient, and x = score on the independent variable

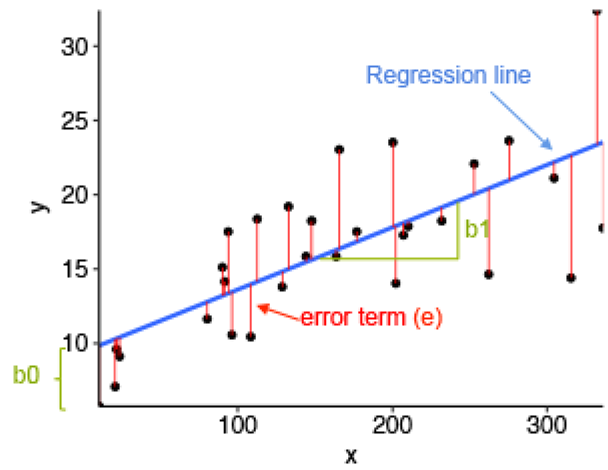


Fig 28: Linear Regression graph

Linear Regression graph is shown in Fig 28. The mathematical formula of the linear regression can be written as y = b0 + b1*x + e, where:

1. b0 and b1 are known as the regression beta coefficients or parameters:
2. b0 is the intercept of the regression line; that is the predicted value when x = 0.
3. b1 is the slope of the regression line.
4. e is the error term (also known as the residual errors), the part of y that can be explained by the regression model

The figure below illustrates the linear regression model, where:

1. the best-fit regression line is in blue
2. the intercept (b0) and the slope (b1) are shown in green
3. the error terms (e) are represented by vertical red lines

From the scatter plot above, it can be seen that not all the data points fall exactly on the fitted regression line. Some of the points are above the blue curve and some are below it; overall, the residual errors (e) have approximately mean zero [44].

The sum of the squares of the residual errors is called the Residual Sum of Squares or RSS.

The average variation of points around the fitted regression line is called the Residual Standard Error (RSE). This is one the metrics used to evaluate the overall quality of the fitted regression model. The lower the RSE, the better it is.

Since the mean error term is zero, the outcome variable y can be approximately estimated as follow:

$$y \sim b0 + b1*x \text{ ----- } (3.5)$$

Mathematically, the beta coefficients (b0 and b1) are determined so that the RSS is as minimal as possible. This method of determining the beta coefficients is technically called least squares regression or ordinary least squares (OLS) regression.

Once, the beta coefficients are calculated, a t-test is performed to check whether or not these coefficients are significantly different from zero. A non-zero beta coefficient means that there is a significant relationship between the predictors (x) and the outcome variable (y).

3.4.8 ALGORITHM FOR WEATHER FORECASTING:

1. Import numpy, pandas, matplotlib and sklearn libraries in google collaboratory
2. Upload the "weather_forecast" dataset which contains the basic four parameters viz. Temperature, pressure, altitude and humidity as a time-series data
3. Clean the dataset for NULL or out of bound values and split the dataset into training and testing in the ratio 70/30
4. Using linear regression machine learning model, prediction is performed for the training dataset. After obtaining high accuracy for trained data, testing is performed

for the weather_forecast dataset

5. For each of the four parameters prediction is performed and a graph of actual vs. predicted parameter values is plotted
6. Mean absolute error and Root mean squared error are calculated for the predicted values and the error is kept less than 1 for high accuracy of prediction

3.5 POWER BUDGET ANALYSIS

Solar energy is used to power the weather monitoring setup as it’s a low power application which works under 3.3V. As solar energy is used, the power consumption must be kept low in order to achieve longer hours of battery backup. In this work solar energy is used as an alternate source of energy to improve the efficiency of the system in terms of energy usage and power consumption.

3.5.1 SOLAR HARVESTING UNIT

The solar harvesting unit comprises the following blocks as shown in Fig 29.

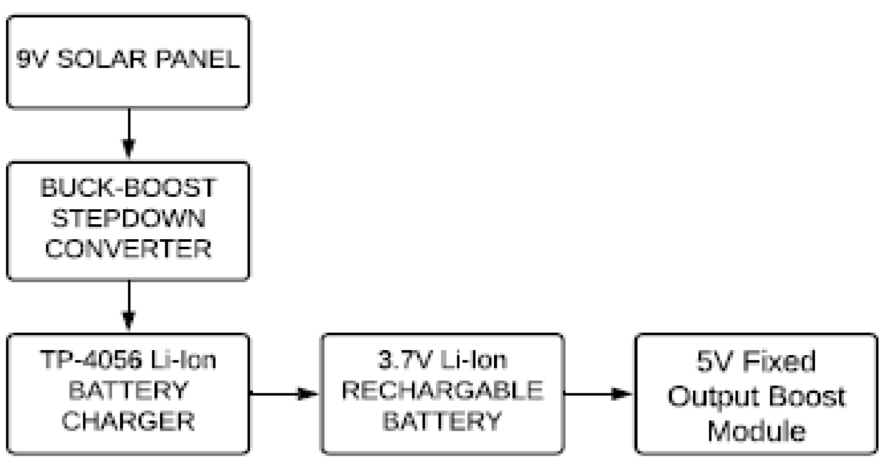


Fig 29:Solar Harvesting Unit

Steps involved in Solar harvesting:

1. The 9V solar panel is stepped down using a buck-boost step-down converter to 5V which is fed into TP4056 Li-ion battery charger module
2. The charging module is used to charge 3.7V Li-ion rechargeable batteries whose output is fed into 5V fixed output boost module
3. The output from the boost module is given to the microcontroller which powers the whole setup

In solar characteristics, time vs. voltage measured, time vs. current drawn and time vs. temperature of the solar panel are studied graphically.

The power budget calculation for each sensor is performed by calculating power consumed per hour and the energy dissipated by each sensor. The solar power characteristics, battery charging and discharging characteristics are also discussed here.

3.5.2 POWER CALCULATION FOR EACH SENSOR

Power is given by:

Power (P) =Voltage (V) x Current (I) ----- (3.6)

which is calculated in milliwatts for each sensor, GPS module and NodeMCU micro-controller.

The energy dissipated in one hour by each sensor and GPS module is calculated as,

Energy(E) =Power(P) x Time(T) ----- (3.7)

For NodeMCU microcontroller,

Let Vn be the operating voltage , In be the operating current and Is be the startup current required by the NodeMCU then the power consumed by NodeMCU is calculated as follows,

Power consumed, Pn = Vn x (In + Is) **----- (3.8)**

where the unit of Pn is milliwatts or “mW”

Similarly for one hour of usage, the Energy dissipated by NodeMCU (En) is calculated using the following formula,

En = Pn x T (in hrs) **----- (3.9)**

where the unit of En is millijoules or “mJ”

For weather sensors,

Let Vb, Vd, Vr, Vg, Vuv be the operating voltages and Id, Ir, Ig, Iuv be the operating currents of pressure, temperature, rain and UV sensors respectively. Then the power and energy dissipated by each sensor is calculated as:

Power consumed by pressure sensor : Pb= Vb x Ib (mW) **----- (3.10)**

Thus Energy dissipated : Eb= Pb x T (mJ) **----- (3.11)**

Power consumed by temperature sensor : $Pd = Vd \times Id$ (mW) **----- (3.12)**

Thus Energy dissipated : $Ed = Pd \times T$ (mJ) **----- (3.13)**

Power consumed by rain sensor : $Pr = Vr \times Ir$ (mW) **----- (3.14)**

Thus Energy dissipated : $Er = Pr \times T$ (mJ) **----- (3.15)**

Power consumed by gas sensor : $Pg = Vg \times Ig$ (mW) **----- (3.16)**

Thus Energy dissipated : $Eg = Pg \times T$ (mJ) **----- (3.17)**

Power consumed by UV sensor : $Puv = Vuv \times Iuv$ (mW) **----- (3.18)**

Thus Energy dissipated : $Euv = Puv \times T$ (mJ) **----- (3.19)**

For GPS module,

Let Vgps operating voltage and Igps be the operating current of GPS module, then the power and energy consumed is given by:

Power consumed, $Pgps = Vgps \times Igps$ **----- (3.20)**

Energy dissipated in one hour, $Egps = Pgps \times T$ (in hrs) **----- (3.21)**

Let Ptot be the total power drawn by the load in a time period of one hour. Then

Total power consumed in milliwatts is given as:

$$Ptot = Pn + Pb + Pd + Pr + Pg + Puv + Pgps \text{ (in mW)} \quad \textbf{----- (3.22)}$$

Let Vmax be the maximum voltage of the setup. Then the total current consumed by load (It) is given as:

$$It = Ptot / Vmax \text{ ----- } (23)$$

3.5.3 BATTERY CHARGING AND DISCHARGING CALCULATIONS

To calculate charging time of Li-ion batteries, let's assume that Tch be the charging time(in hrs), C be the capacity of two Li-Ion batteries and Ic be the charging current. Then,

Charging current Ic can be calculated as Ic= 10% of C

$$Tch = C / Ic \text{ ----- } (3.24)$$

To calculate discharging time of Li-ion batteries, let Tdh be the discharging time(in hrs) and C be the capacity of the batteries and Ij be the current drawn by the load. Then discharging time can be calculated as,

$$Tdh = C / Ij \text{ ----- } (3.25)$$

3.6 LINK BUDGET ANALYSIS

A Link Budget shows all of the gains and losses from a transmitter, through the medium (free space, cable, waveguide, fiber, etc.) to the receiver in a telecommunication system. It’s used to predict the performance of a transmitter and receiver communication link to show in advance if its performance is acceptable, or if one option is better than another. It accounts for the attenuation of the transmitted signal due to propagation, as well as the antenna gains, feedline, miscellaneous losses and added margin [53].

Since NodeMCU has an inbuilt transreceiver wifi module ESP8266, it acts as both a transmitter and receiver. The system requires a wireless network connection which enables it to upload weather data into cloud databases.

For this the radius of coverage from the internet source to NodeMCU microcontroller is calculated.

Internet Source= Wifi Router/ Mobile Hotspot

The radius of coverage (Rc) is calculated in feet and then converted to meters using the formula:

$$\mathbf{Rc_m = Rc_f / 3.2808 \text{ ----- } (3.26)}$$

Where $\mathbf{Rc_m}$ is radius of coverage in meters (m) and $\mathbf{Rc_f}$ is radius of coverage in feet (ft)

The area of coverage (Ac) can be found out using the formula:

$$\mathbf{Ac = 3.14 \times Rc^2 \ (m^2) \text{ ----- } (3.27)}$$

3.7 PCB DESIGNING

A printed circuit board (PCB) mechanically supports and electrically connects electrical or electronic components using conductive tracks, pads and other features etched from one or more sheet layers of copper laminated onto and/or between sheet layers of a non-conductive substrate. Components are generally soldered onto the PCB to both electrically connect and mechanically fasten them to it.

Printed circuit boards are used in all but the simplest electronic products. They are also used in some electrical products, such as passive switch boxes. PCBs can be single-sided (one copper layer), double-sided (two copper layers on both sides of one substrate layer), or multi-layer (outer and inner layers of copper, alternating with layers of substrate). Multi-layer PCBs allow for much higher component density, because circuit traces on the inner layers would otherwise take up surface space between components [54].

PCB design services are used to design the electronic circuits. Apart from electrically connecting the circuit connections, it also gives mechanical support to the electrical components and thus it has the following advantages:

1. Ease of Repair and Diagnostics of components
2. Efficient time usage as circuit connections take much longer to assemble
3. Immune to Movement
4. Tight connections and Short Circuits are avoided
5. Low Electronic Noise
6. Low Cost
7. High Reliability [56]

Among various PCB designing softwares like kiCad EDA, Autodesk EAGLE etc., EasyEDA PCB designer software is used for a number of reasons. Using EasyEDA has the following advantages compared to other traditional tools like OrCad, Altium, and DipTrace etc.

1. No need to install any special softwares, a web browser is enough
2. Will work on all operating systems which supports a modern browser
3. No need of large resources or high power computers to do big projects

4. No need to update the software, it is automatically applied to all clients when the server software is updated
5. Design sharing feature with online community [57]

The PCB design for the proposed work is based on the system design developed using the Lucid.app designer shown in Fig 30.

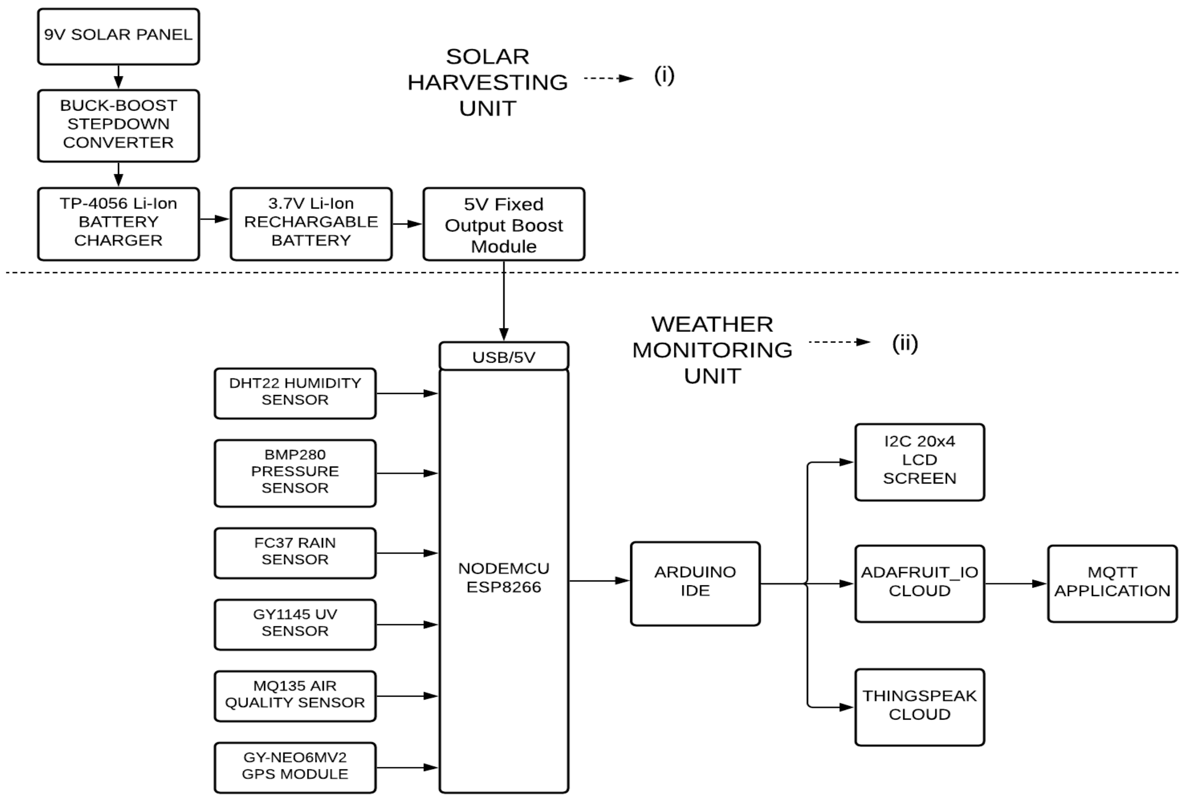


Fig 30: System Design

The system design shown in Fig 29 is the foundation for PCB designing using the EasyEDA designer which is split into solar harvesting unit and weather monitoring unit. The solar harvesting unit is used for continuous powering of the weather monitoring unit. The monitoring setup displays weather parameters in an 20x4 I2C LCD in loop format and uploads the weather data into Thingspeak and Adafruit IoT cloud systems to facilitate remote weather monitoring.

CHAPTER IV

RESULTS AND DISCUSSION

4.1 SENSOR CHARACTERISTICS

4.1.1 DHT22 TEMPERATURE AND HUMIDITY SENSOR

The characteristics of temperature and humidity sensor(DHT22) are studied based on readings taken at three different temperatures. The parameters are monitored for a span of 5 days and the average value is taken for graphical representation shown in Fig 31.

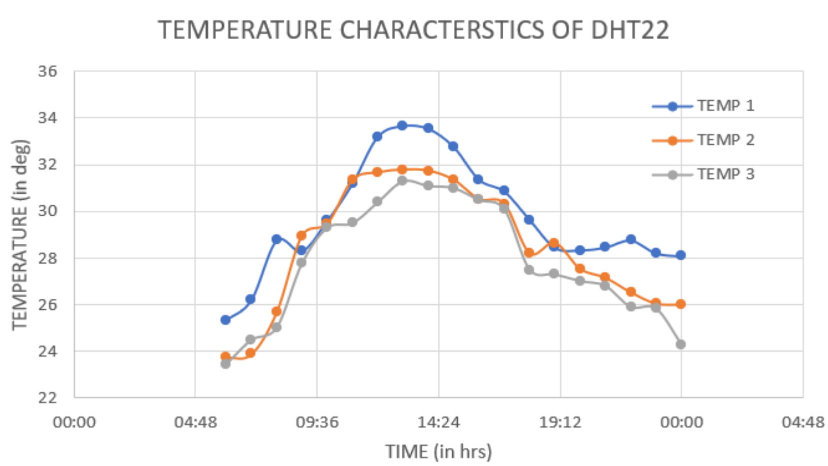


Fig 31: Temperature characteristics of DHT22

The temperature characteristics show a trend that at noon the temperature spikes around 12:00 noon -2:00 PM and gradually reduces overnight, this is mainly because of the intensity of sunlight which spikes at noon and thus increases the temperature of the atmosphere due to the increased radiation energy of photons from the sun. This also increases the UV radiation from the sun which results in an increase in UV Index(UVI) which can be observed in Fig 32.

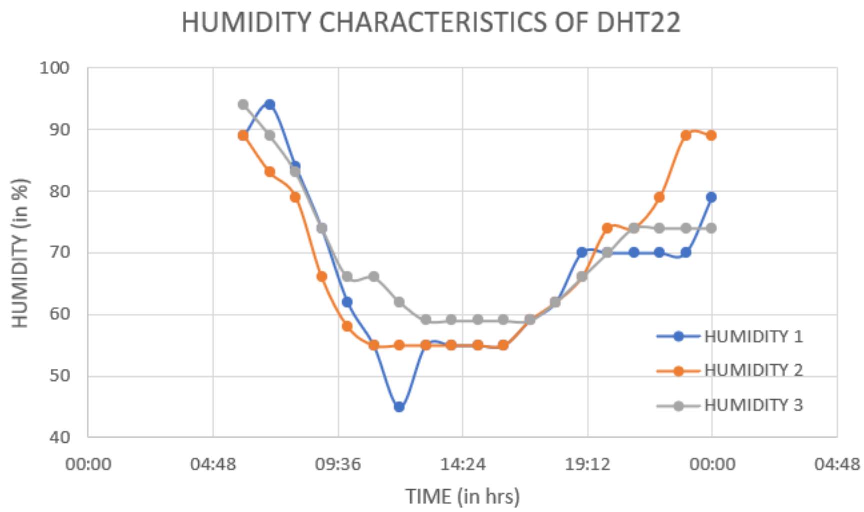


Fig 32: Humidity characteristics of DHT22

It is observed that at 12.00 noon-2.00 pm, temperature reaches its peak. As per ideal gas law, if temperature increases air can hold more water molecules, and its relative humidity decreases. When temperatures drop, relative humidity increases and when temperatures spike, relative humidity decreases. High relative humidity of the air occurs when the air temperature approaches the dew point value. Thus at 12:00 noon to 2 pm as the temperature spikes, a sharp drop at relative humidity percentage value is observed in Fig 32. From the graph it is observed that temperature and relative humidity are inversely proportional.

4.1.2 BMP280 PRESSURE SENSOR

The BMP280 is an absolute barometric pressure sensor, which is especially feasible for mobile applications. Its small dimensions and its low power consumption allow for the implementation in battery-powered devices such as mobile phones, GPS modules or watches. The BMP280 is based on Bosch's proven piezo-resistive pressure sensor technology featuring high accuracy and linearity as well as long-term stability and high EMC robustness. Numerous device operation options guarantee for highest flexibility. The device is optimized in terms of power consumption, resolution and filter performance.

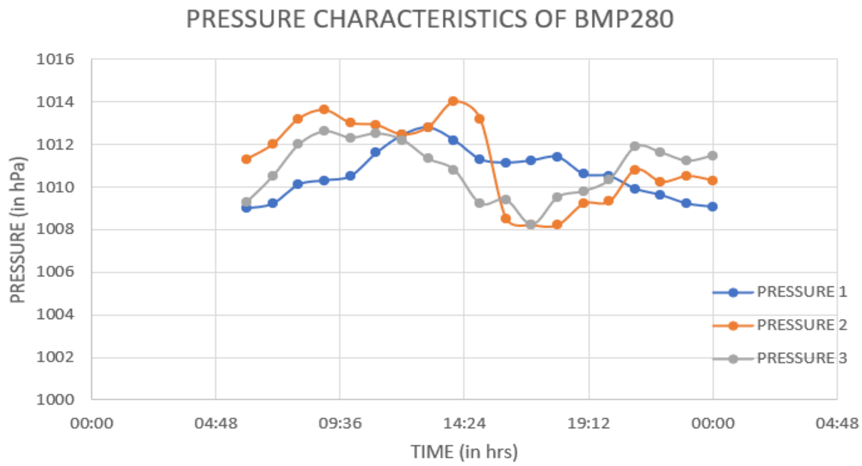


Fig 33:Pressure characteristics of BMP280

Fig 33 shows Pressure characteristics of BMP280. The Ideal gas law states that $PV = NkT$, where P is the absolute pressure of a gas, V is the volume it occupies, N is the number of atoms and molecules in the gas, and T is its absolute temperature [24]. As pressure and temperature are directly proportional, an increase in temperature at noon results in an increase in atmospheric pressure as well. This phenomenon is explained by ideal gas law i.e., if temperature increases, the particles move faster and therefore have greater speeds, so greater momentum and therefore greater force when they collide with the walls, so the pressure increases. Altitude characteristics of BMP280 are as shown below in Fig 34.

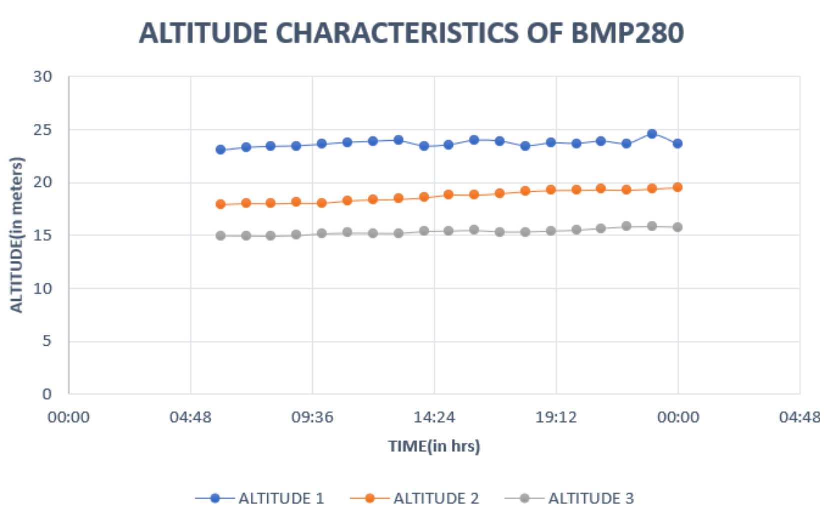


Fig 34: Altitude characteristics of BMP280

As the altitude is increased, the air gets increasingly rarer, which leads to a loss of pressure. Temperature is directly proportional to pressure, so the lower the pressure, the lower the temperature. In short, the higher the altitude, the lower the temperature. The heat index, also known as the apparent temperature, is what the temperature feels like to the human body when relative humidity is combined with the air temperature. This has important considerations for the human body's comfort. When the body gets too hot, it begins to perspire or sweat to cool itself off. Heat index characteristics are shown below in Fig 35.

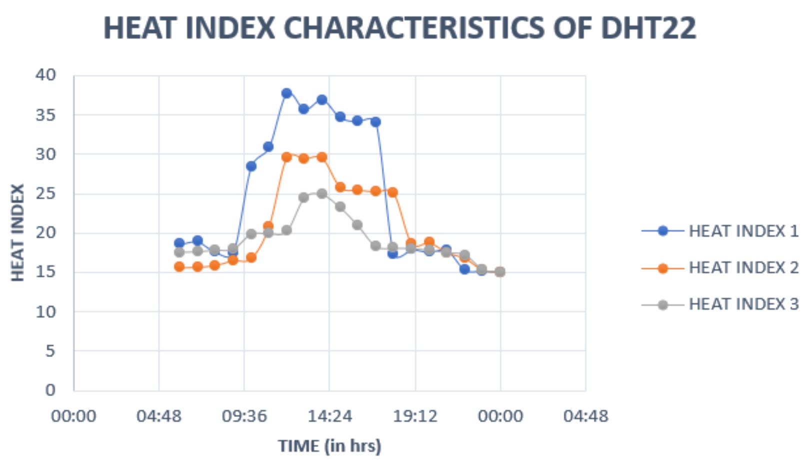


Fig 35: Heat Index characteristics of DHT22

It can be inferred that the heat index falls mostly in the range of 15 to 36. As the temperature value increases, heat index value tends to reach its peak. Hence, it can be observed that,12 noon to 2.00 pm has the maximum heat index value. The dew point is the temperature to which the air needs to cool down in order to become completely saturated, or reach 100 percent relative humidity. Once the air temperature cools below its dew point, water vapor in the atmosphere will condense.The relative humidity will go up at night when the air temperature approaches the dew point, and the relative humidity will go down as the air temperature warms farther away from the dew point during the day.

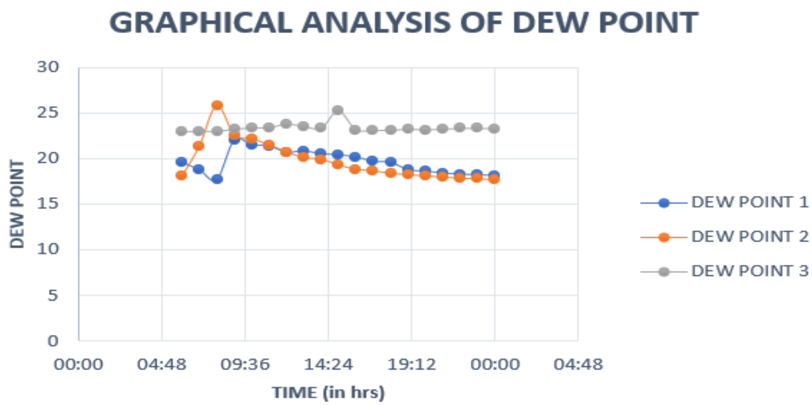


Fig 36: Graphical analysis of Dew point

From the Fig 36, it can be observed that dew point is almost constant for a particular day. The UVI is a measure of the level of UV radiation. The values of the index range from zero upward - the higher the UVI, the greater the potential for damage to the skin and eye, and the less time it takes for harm to occur.

4.1.3 GY1145 UV SENSOR

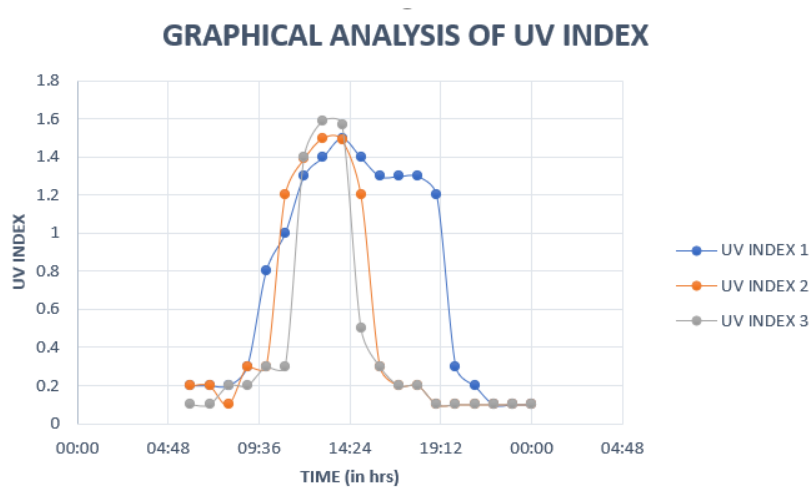


Fig 37: Graphical analysis of UV index

From Fig 37, it is observed that during day and night, UV index falls in the range of 0-1.2

UV index has a value of 1.5 around noon. The air quality index (AQI) is an index for reporting air quality on a daily basis. It is a measure of how air pollution affects one's health within a short time period.

4.1.4 MQ135 AIR QUALITY SENSOR

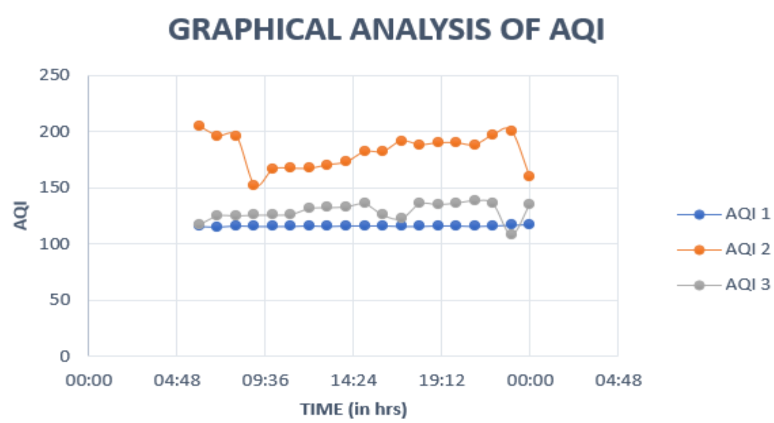


Fig 38: Graphical analysis of AQI

The graph showing Air quality index in three different zones is shown in Fig 38. It can be observed that air quality level is almost a constant value in a particular location. The purpose of the AQI is to help people know how the local air quality impacts their health. AQI values above 900 (in PPM) are considered harmful for humans as it may cause breathing problems.

4.2 EXPERIMENTAL ANALYSIS

Sensors are interfaced with a microcontroller like nodemcu and are powered using a renewable energy source. The output is displayed in an LCD screen and also graphically visualized in Thingspeak, Adafruit cloud and MQTT Application. The obtained data further used for making weather predictions.

4.2.1 DAYTIME

Experimental setup during day time in the presence of sunlight is shown in Fig 39.

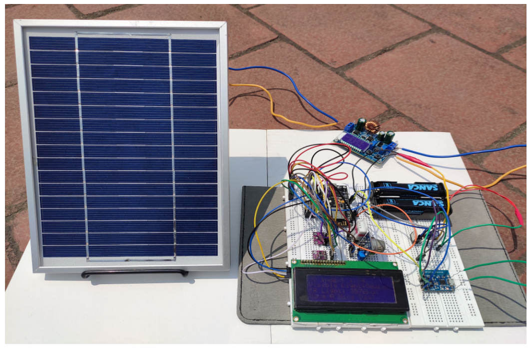


Fig 39:Experimental setup during daytime

During daytime, the complete setup is powered using solar energy which is the most abundant renewable energy. The input from the solar panel is stepped down to 5 V using buck-boost converter and is fed to the TP4056 module from which the battery is charged and the setup is powered simultaneously. The monitored parameters are displayed in an LCD screen in a loop-back format.

4.2.2 NIGHT-TIME

Experimental setup during night time in the absence of sunlight is shown in Fig 40.

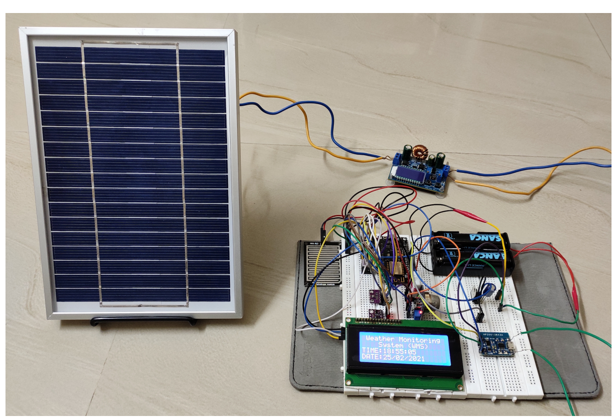


Fig 40:Experimental setup during nighttime

During night time, the solar panel and Buck-boost convertor are kept idle. The setup is powered using the charge stored in the battery. The output is displayed in an LCD screen and is graphically visualized in Thingspeak and Adafruit cloud.

4.3 CLOUD DATA STORAGE

4.3.1 ADAFRUIT IO CLOUD

The monitored parameters are stored in cloud platforms in real-time and further visualization is performed. The output obtained from the Adafruit dashboard is shown in Fig 41.

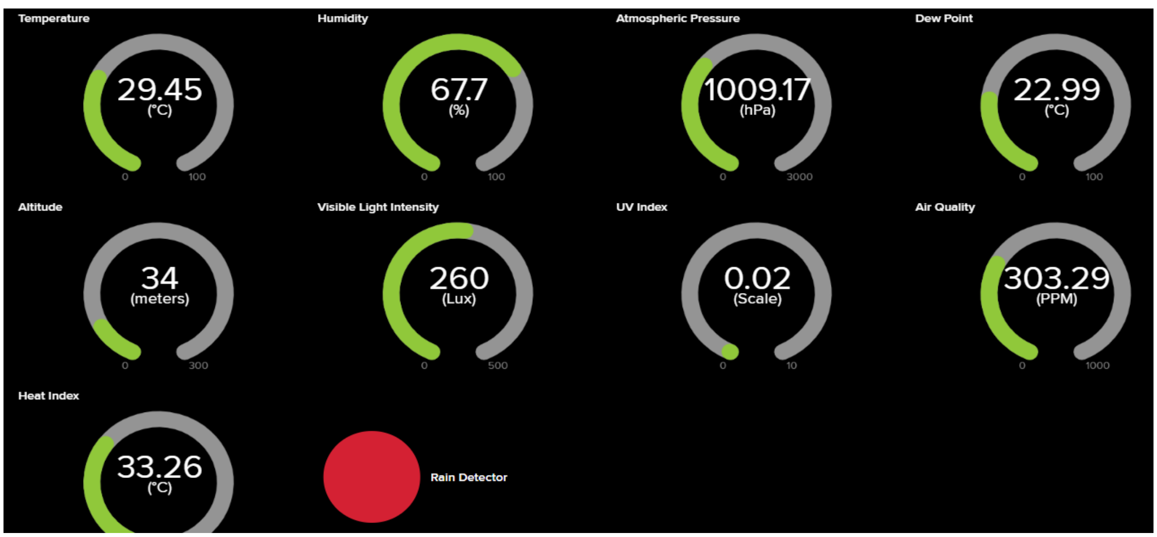


Fig 41:Adafruit cloud Output

Each parameter is displayed in a widget. The maximum and minimum range is set and units are marked for each parameter. By clicking on the feed name, each parameter can be graphically visualized. Rain indicator changes from red to blue while sprinkling droplets which indicates real time weather updates can be easily visualized using an Adafruit cloud. The monitored parameters can be downloaded from the Adafruit cloud and can be further used for analysis, such as for weather prediction. Graphical visualization of the monitored weather parameters can be seen in Thingspeak cloud which uses MATLAB for analysis.

4.3.2 THINGSPEAK CLOUD

The outputs of Thingspeak cloud for 8 weather parameters are visualized as shown below.

Fig 42:Temperature and Humidity plot in Thingspeak cloud

Temperature and Humidity plot for certain time intervals is shown in Fig 42. As air temperature increases, air can hold more water molecules, and its relative humidity decreases. When temperatures drop, relative humidity increases. This can be observed graphically from the above graphs. Pressure and altitude plots for certain time intervals are shown in Fig 43.

Fig 43:Pressure and altitude plot in Thingspeak cloud

The pressure of a given amount of gas is directly proportional to the temperature at a given volume. When the temperature of a system goes up, the pressure also goes up, and vice versa. Thus it is observed from the graphs that Temperature and pressure are linearly relatable. Pressure on Earth varies with the altitude of the surface; so air pressure on mountains is usually lower than air pressure at sea level. Thus it can be seen that air pressure decreases as altitude increases. Dew point and Air quality plot is shown in Fig 44.

Fig 44:Dew point and Air quality plot in Thingspeak cloud

Dew point is the atmospheric temperature (varying according to pressure and humidity) below which water droplets begin to condense and dew can form. Dew point is calculated and plotted. Peak dew point is noted as 24 deg C (approx.) Hence, below 24 deg C ,water droplets begin to condense and form dew. Air quality index at a particular location ranges from 100 to 125. It can be observed that air quality is worse at the peak hours. Understanding the Air Quality Index is important because it gives people vital information about the conditions of the air in their location and how the quality of the air in their city can impact their health.

Fig 45:Heat Index and Dew point plot of Thingspeak cloud

Heat index and UV index plot at different locations with various data entries is shown above in Fig 45. Heat index has a mid-value of 34 and it increases as temperature increases. Knowing the heat index helps us understand how hot it feels outside. The UV Index reminds people to protect themselves when engaging in their normal outdoor activities. UV radiation exposure poses varying degrees of risk for all people because it affects eyes and skin. People with sensitive skin should always take action to protect them. The UV index plot shows that it ranges from 0.1 and reaches a maximum of 1.5. Since the UV peak is at 1.5, no protection is required.

4.3.3 MQTT APPLICATION

MQTT is used for data exchange between constrained devices and server applications. It keeps bandwidth requirements to an absolute minimum, handles unreliable networks, requires little implementation effort .The output is also displayed in MQTT application as shown in Fig 46 for multiple subscribers to visualize.

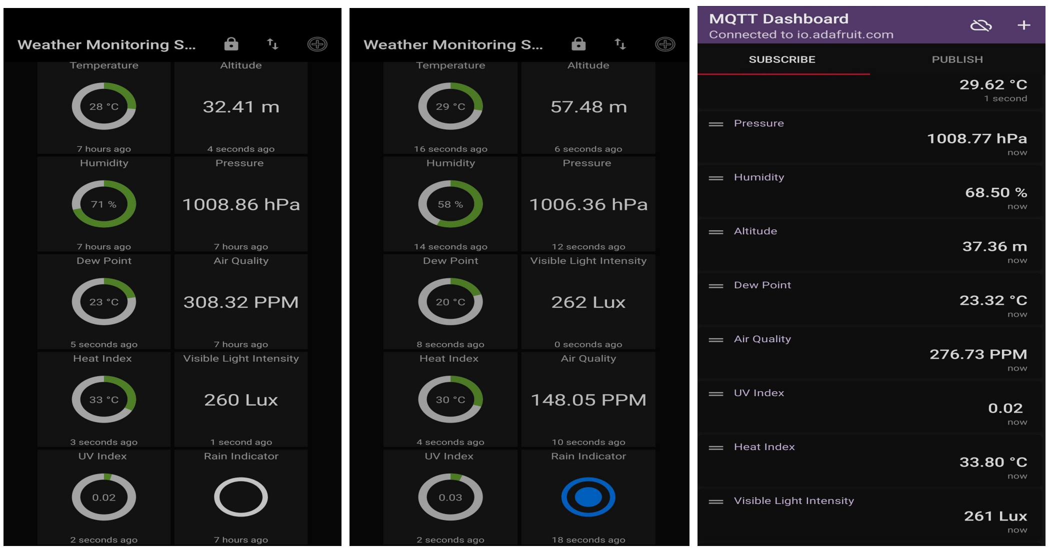


Fig 46:MQTT application outputs

From the first two figures, the variations in all the weather parameters can be observed. It is because of weather variations at different locations. The second figure has a blue LED blinking indicating rainy condition. GPS is used for live tracking of location and other details like date and time are also displayed. The location details where our setup is placed is shown below in Fig 47 which is taken by copying the IP address of the local web server that is displayed in the serial monitor.

4.3.4 GPS LOCATION

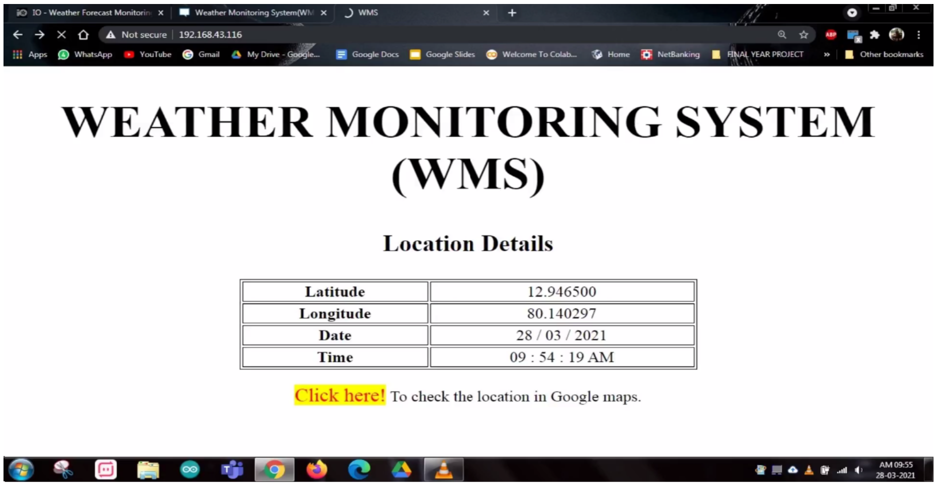


Fig 47:Location details shown by GPS module displayed in web server

HTML coding is used to visualize GPS location tracking details in a web page. The IP address of the local web server is obtained from the serial monitor. GPS Module will transmit data in multiple strings at 9600 Baud Rate. If UART is used with a terminal of 9600 baud rate, the data would be received by GPS. This webpage redirects us to Google maps where location can be seen in a map view which is shown in Fig 48.

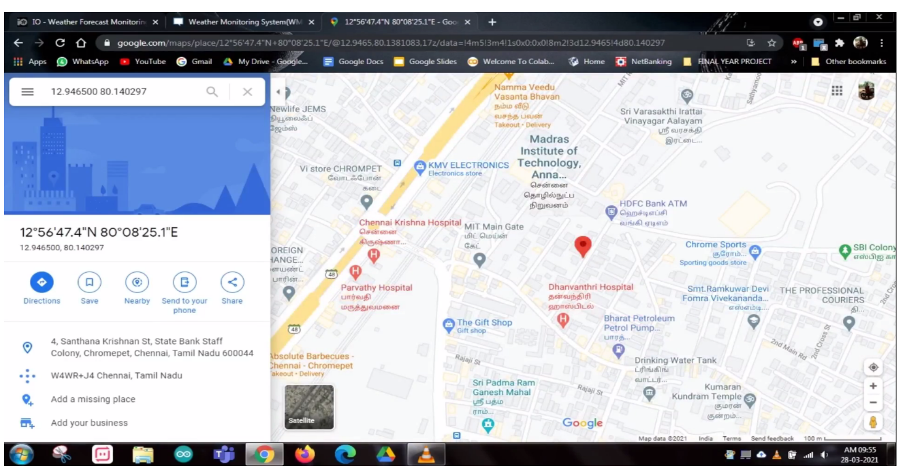


Fig 48:Map view showing the location where setup is placed

4.4 WEATHER FORECASTING ANALYSIS

Weather forecasting is the application of science and technology to predict the conditions of the atmosphere for a given location and time.There are various models to predict weather.Here regression technique is used considering its simplicity.It is performed by using a linear regression machine learning model. The forecasted weather parameters are plotted using Google colaboratory for graphical visualization.

4.4.1 TEMPERATURE

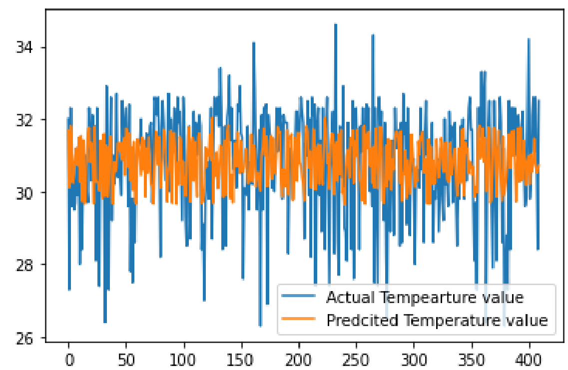


Fig 49:Graphical analysis of forecasted temperature and actual temperature

Temperature dataset which is obtained from Adafruit cloud is fed to the machine learning model. The model preferred is linear regression model and training is done.Half of the dataset is trained and testing is done on the remaining dataset. Prediction is performed on the deployed model and graph is plotted as shown above in Fig 49. Yellow lines indicate predicted temperature value and blue lines indicate actual temperature value. The error obtained is minimal which indicates accuracy is better.

4.4.2 ATMOSPHERIC PRESSURE

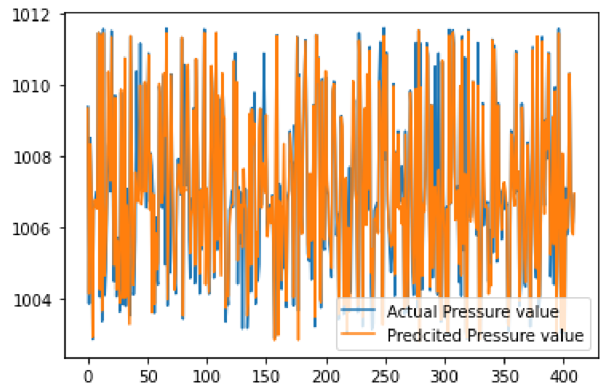


Fig 50:Graphical analysis of forecasted Pressure and actual Pressure

Pressure dataset which is obtained from Adafruit cloud is fed to the machine learning model. Half of the dataset is trained and testing is done on the remaining dataset. Prediction is performed on the deployed model and graph is plotted as shown above in Fig 50. Yellow lines indicate predicted pressure value and blue lines indicate actual pressure value. The error obtained is minimal which indicates accuracy is much better.

4.4.3 HUMIDITY PERCENTAGE

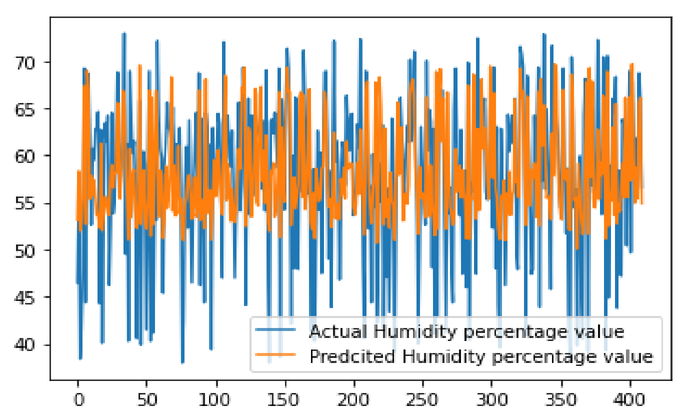


Fig 51:Graphical analysis of forecasted humidity and actual humidity

Humidity dataset which is obtained from Adafruit cloud is fed to the machine learning model. Half of the dataset is trained and testing is done on the remaining dataset. Prediction is performed on the deployed model and graph is plotted as shown above in Fig 51. Yellow lines indicate predicted Humidity value and blue lines indicate actual Humidity value. The error obtained is minimum which indicates accuracy is high.

4.4.4 ALTITUDE

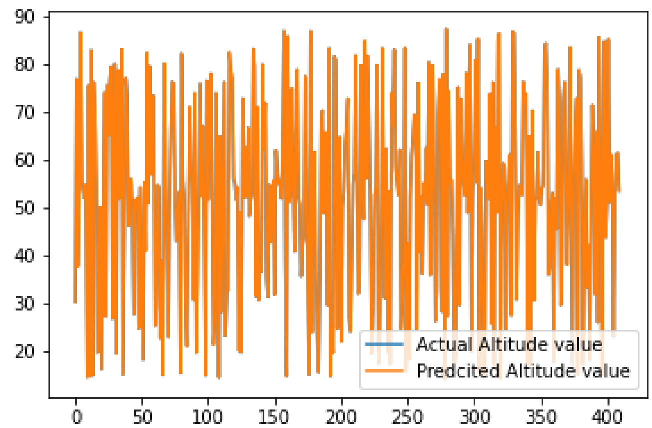


Fig 52:Graphical analysis of forecasted altitude and actual altitude

Altitude dataset which is obtained from Adafruit cloud is fed to the machine learning model. Half of the dataset is trained and testing is done on the remaining dataset. Prediction is performed on the deployed model and graph is plotted as shown above in Fig 52. Yellow lines indicate predicted altitude value and blue lines indicate actual altitude value. The error obtained is the minimum, by observing that yellow lines and blue lines coincide which means predicted value is almost the same as actual value. Hence accuracy is the best. Thus, it can be concluded that the machine learning model chosen seems to be best for simple weather prediction results.

4.5 POWER BUDGET CALCULATIONS AND ANALYSIS

The power budget calculations for the microcontroller, sensors and GPS module have been analyzed below:

Power (P) =Voltage (V) x Current (I) which is calculated in milliwatts for each sensor and NodeMCU micro-controller.

The energy dissipated in one hour by each sensor and GPS module is calculated as,

$$E = P \times T \text{ (from eqs. 6 \& 7)}$$

The energy dissipated (E) is calculated for T=8hrs

4.5.1 NODEMCU MICROCONTROLLER

Operating Voltage (Vn) = 3.3 V; Operating Current (In)= 80 mA;

Start-up current (Is) = 400mA

From eqs. (8) & (9),

Power consumed (Pn) = Vn x (In + Is)

$$= 3.3 \times (80 + 400) = 1584\text{mW}$$

Energy dissipated (En) = Pn x T (in hrs)

$$= 1584\text{mW} \times 8 \text{ hrs} = 12.672\text{J}$$

4.5.2 WEATHER MONITORING SENSORS

BMP280 Pressure Sensor:

Operating Voltage(Vb)= 3.3 V; Operating Current(Ib)= 1.12 mA

From eqs. (10) & (11);

Power consumed by pressure sensor : Pb= Vb x Ib (mW)

= (3.3 x 1.12) mW = 3.696mW

Thus Energy dissipated : Eb= Pb x T (mJ)

= 3.696mW x 8 hrs = 29.57mJ

DHT22 Humidity Sensor:

Operating Voltage(Vd)= 3.3 V; Operating Current(Id)= 1.5 mA

From eqs. (12) & (13);

Power consumed by temperature sensor : Pd= Vd x Id (mW)

= (3.3 x 1.5) mW

=4.95 mW

Thus Energy dissipated : Ed = Pd x T (mJ)

= 4.95mW x 8 hrs = 39.6 mJ

FC37 Rain Sensor:

Operating Voltage(Vr)= 3.3 V; Operating Current(Ir)= 15 mA

From eqs. (14) & (15);

Power consumed by rain sensor : $Pr = Vr \times Ir$ (mW)

$= (3.3 \times 15 \text{ mW}) = 49.5$ mW

Thus Energy dissipated : $Er = Pr \times T$ (mJ)

$=49.5\text{mW} \times 8\text{hrs} = 396$ mJ

MQ135 Air Quality Sensor:

Operating Voltage(Vg)= 3.3 V; Operating Current(Ig)= 150 mA

From eqs. (16) & (17);

Power consumed by gas sensor : $Pg = Vg \times Ig$ (mW)

$= (3.3 \times 150)$ mW = 495mW

Thus Energy dissipated : $Eg = Pg \times T$ (mJ)

$= 495\text{mW} \times 8 \text{ hrs} = 3.96$J

GY1145 UV Sensor:

Operating Voltage(Vuv)= 3.3 V; Operating Current(Iuv)= 5.5 mA

From eqs. (18) & (19);

Power consumed by UV sensor : $Puv= Vuv \times Iuv$ (mW)

$= (3.3 \times 5.5)$ mW= 18.15 mW

Thus Energy dissipated : Euv= Puv x T (mJ)

=18.15mW x 8hrs = 145.2 mJ

4.5.3 GPS MODULE

Operating Voltage(Vgps)= 3.3 V; Operating Current(Igps)= 10 mA

From eqs. (20) & (21);

Power consumed, Pgps = Vgps x Igps

= (3.3 x 10) mW = 33 mW

Energy dissipated in one hour, Egps = Pgps x T (in hrs)

=33mW x 8hrs = 264 mJ

From eq. (22);

Total power consumed in milliwatts is given as:

Ptot = Pn + Pb + Pd + Pr + Pg + Puv + Pgps (in mW)

Ptot = 1584mW+ 3.696mW + 4.95mW + 495mW + 49.5mW + 18.15mW + 33mW

Ptot = 2188.296 mW = 2188.3 (approx.)

From eq. (23);

Total current consumed by load:

Current (It) = Power drawn by the load (Ptot) /maximum voltage (Vmax)

It = 2188.3 mA / 3.3 V = **663.121 mA**

4.5.4 BATTERY CHARGING AND DISCHARGING CALCULATIONS

1. Charging time:

Charging current Ic can be calculated as Ic= 10% of C

From eq. (24);

Charging current of 5200 mAh Battery

= 5200 x 10/100 = 1000 mA

Charging time (Tch) = Battery capacity in Ah (C) / Charging current (Ic)

$\therefore$ Charging time = 5200 mAh/ 1000 mA= **5.2 hrs.**

2. Discharging time:

From eq. (25); Discharging time (Tdh) = Capacity of battery (C) / Current drawn by the load (Ij)

= 5200 mAh / 663.121 mA = 7.841 hrs.

= **8 hrs. @ 663 mA *(approx.)***

4.5.5 SOLAR PANEL CHARACTERISTICS

In the area of renewable energy, solar energy is at the forefront, because producing energy by using the power of the sun is the easiest and commercially viable way of renewable energy. Speaking of solar panels, the output power of a solar panel output needs to be monitored in order to get optimum power output from the panels. In a large solar power plant, it can also be used to monitor the power output from each panel which helps to identify the dust buildup. It also prevents any fault conditions during the time of operation. The voltage characteristics of solar panel are visualized graphically as shown below in Fig 53:

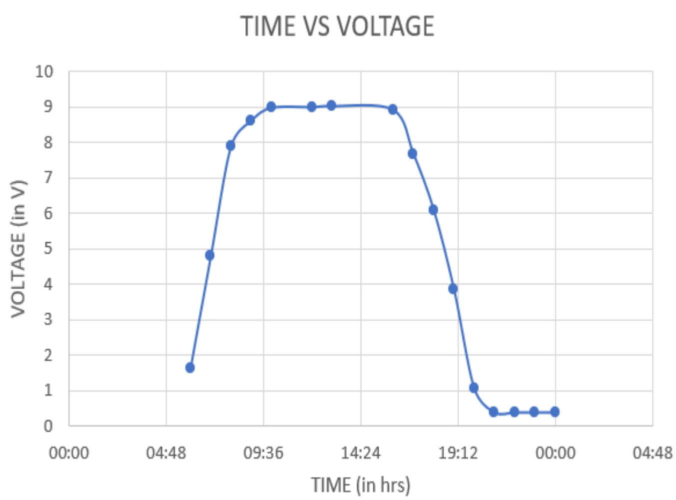


Fig 53:Solar panel voltage characteristics

As time increases, solar voltage reaches its peak and remains constant for a few hours and then starts decreasing gradually. It reaches almost minimum value during night time which shows that the solar panel has no operation during night time. The current characteristics of solar panel are visualized graphically as shown below in Fig 54.

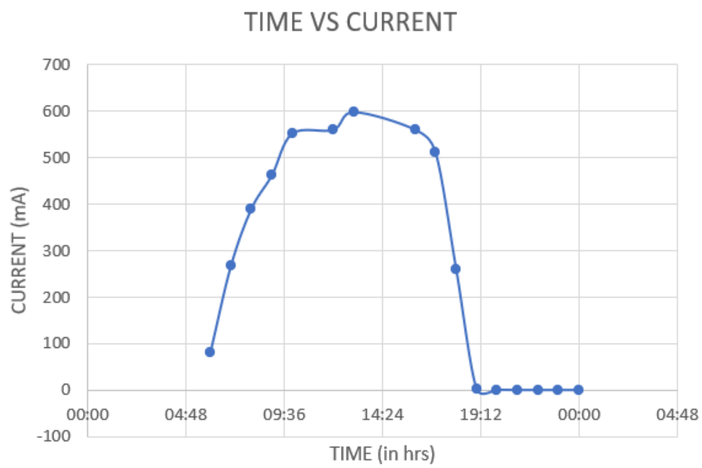


Fig 54:Solar panel current characteristics

As time increases, solar current reaches its peak and remains constant for a few hours and then starts decreasing gradually. It reaches almost minimum value during night time which shows that the solar panel has no operation during night time. The temperature characteristics of solar panel are visualized graphically as shown below in Fig 55.

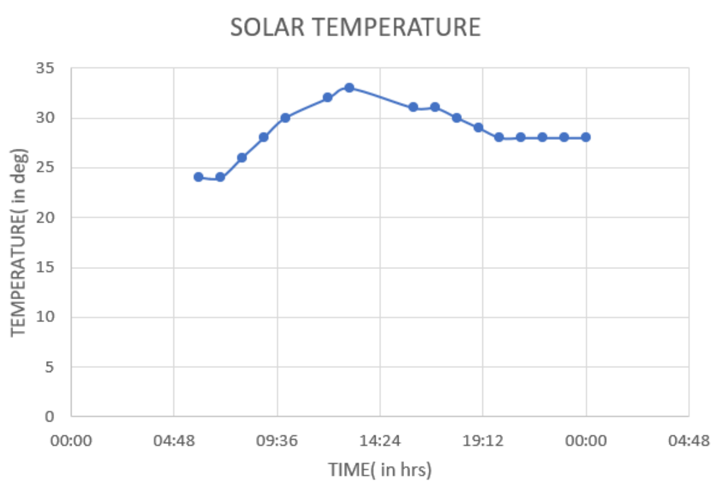


Fig 55:Solar panel temperature characteristics

Solar panel power output has a direct connection with the temperature of the solar panel as a solar panel's temperature starts to increase the output current from the solar panel increases exponentially while the voltage output starts to reduce linearly. As per the power formula, Wattage is equal to voltage times current (W = V x A), decreasing output voltage also decreases the solar panel output power even after the increase of current flow.

4.5.6 BATTERY CHARACTERISTICS

1. Battery Charging Profile:

The graph given in Fig 56 shows the number of hours taken to fully charge a Li-ion battery. This battery charges upto 4.2 V. The cut off voltage of LI-ion batteries is 2.3 V.

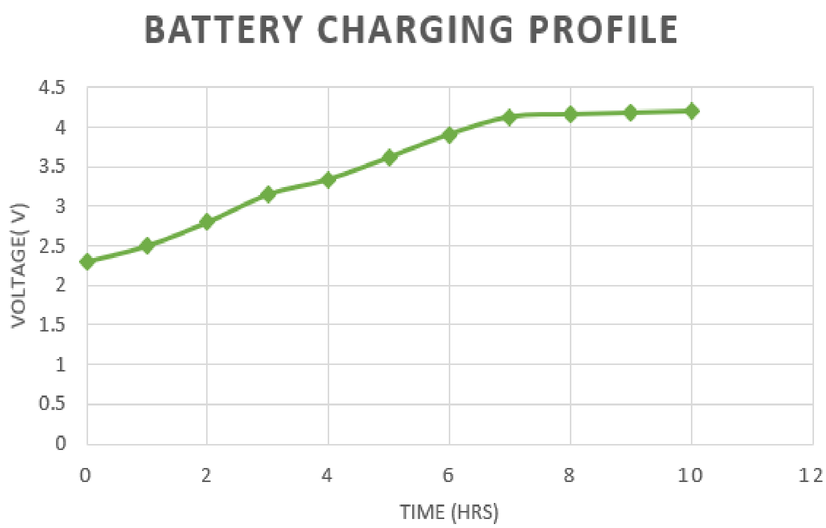


Fig 56:Battery charging profile

It can be inferred that, as time increases, voltage also increases linearly. After it reaches a peak of 4.2 V, it remains constant for some particular time after which self-discharge takes place.

2. Battery charging characteristics:

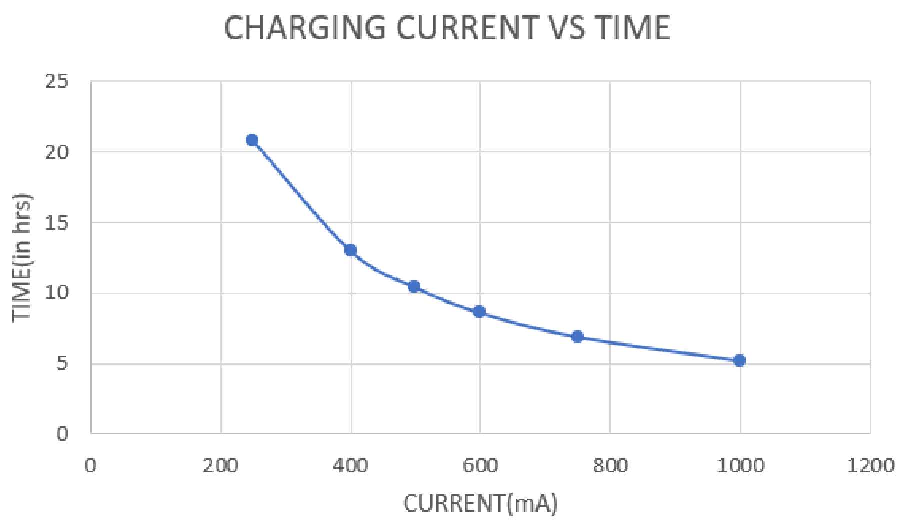


Fig 57:Battery charging characteristics

The above graph in Fig 57 shows the relation between current in mA and the time it takes to charge a Li-ion battery. To charge the battery with 1000 mA current, it requires only 5 hrs and to charge a battery with 250 mA current, it takes nearly 20 hrs. Thus, it can be concluded that as charging current increases, charging time decreases. Charging current is inversely proportional to charging time.

3. Battery discharging characteristics:

A battery exhibits capacitor-like characteristics when discharging at high frequency. This allows higher peak currents than is possible with a DC load. Nickel- and lithium-based batteries have a fast chemical reaction; lead acid is sluggish and requires a few seconds to recover between heavy loads.

The graph given in Fig 58 shows how the performance of Lithium Ion batteries deteriorates as the time increases.

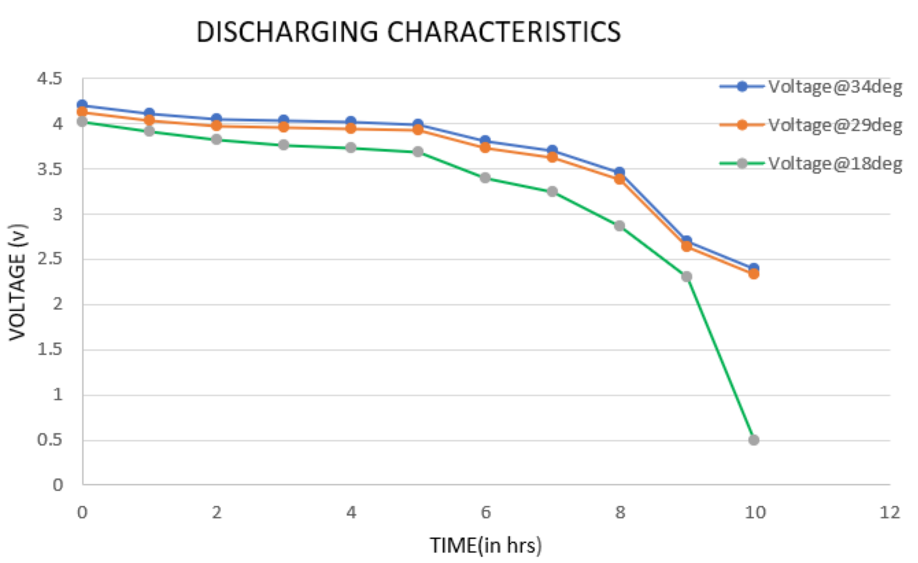


Fig 58:Battery discharging characteristics

Cell performance can change dramatically with temperature. At the lower extreme, in batteries with aqueous electrolytes, the electrolyte itself may freeze setting a lower limit on the operating temperature. At low temperatures Lithium batteries suffer from Lithium plating of the anode causing a permanent reduction in capacity. At the upper extreme the active chemicals may break down destroying the battery. In between these limits the cell performance generally improves with temperature.

4.6 LINK BUDGET CALCULATIONS

4.6.1 Area of Coverage:

Internet Source= Wifi Router/ Mobile Hotspot

The radius of coverage (Rc) is calculated in feet and then converted to meters using eq. (26):

Rc in meters (m) =Rc in feet(ft) /3.2808

Radius for which the signal strength was appreciable and the setup was functioning without network loss =130 ft.

Converted to meters, Radius=40 m

From eq. (27);

Area of circle = 3.14 x 40 x 40 m^2

= 5024 m^2

= **5 km^2** *(approx.)*

4.7 PCB SCHEMATICS

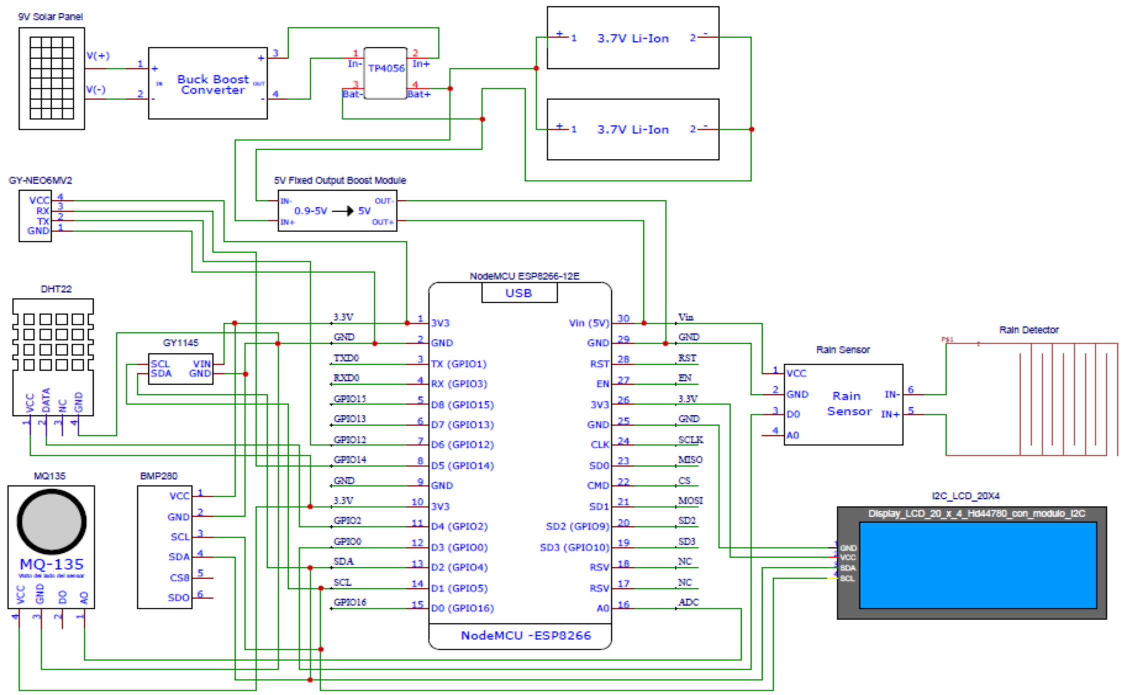


Fig 59:PCB schematics of Solar energized weather monitoring setup

Fig 59 shows the PCB schematics of Solar energized weather monitoring setup which is designed using EasyEDA PCB designing tool. Each of the components in the schematics are converted into footprints and then a Gerber file is generated. The Gerber files are used to describe the printed circuit board images: copper layers, solder mask, legend, drill data, etc. using which the final hardware PCB layout can be created.

CHAPTER V

CONCLUSION AND FUTURE WORK

Thus a prototype of GPS enabled weather monitoring system is developed which is powered using solar energy, a cost effective solution for long term energy requirements. The weather monitoring system is portable and can be further expanded to monitor developing cities and manufacturing zones for pollution monitoring. The system uploads live weather data to cloud services such as Adafruit_IO that can further be used for weather forecasting and to analyze weather patterns. To protect the public health from pollution, this model provides an efficient and low cost solution for unceasing monitoring of the environment.

The addition of GPS module to the weather monitoring setup increases the security and reliability of the system. Obtaining real-time latitude and longitude values along with date and time facilitates remote monitoring of the system from an HTML webpage. Moreover with the addition of Google maps functionality in the webpage, the setup can be tracked in satellite or map view to exact latitude and longitude coordinates at any given time. Thus it increases the portability of the self-powered weather monitoring system.

The weather forecast monitoring system helps farmers and gardeners plan for crop irrigation and protection (irrigation, scheduling, and freeze protection). The weather prediction model can eventually be used to analyze climate patterns, as well as for other meteorological purposes. Atmospheric parameters at high altitudes or remote and inaccessible areas can be recorded by suspending the device from a weather balloon. The system can be protected by using a tough exterior cover so that it would not be affected by harsh weather conditions. Wind Speed and wind velocity can be measured using an anemometer connected to a microcontroller. Forecasted weather parameters along with current weather parameters can be displayed in a webpage by making use of HTML coding.

This work can further be extended by adding more number of gas sensors such as MQ-3 and MQ-7 to feature air pollution monitoring. This model can be expanded to industrial zones for pollution monitoring. Users can receive alert messages in the form of SMS from time to time by the addition of a GSM module to the weather monitoring setup which ensures 24x7 network connectivity even in remote areas. Automatic Email delivery system can be added which sends alert emails to users if the value of weather parameters cross a threshold value. For example if the UV index reaches above 5, the system can alert by sending a SMS and an email to the registered users. The proposed work allows remote access to sensing data via the cloud, which was tested on a laptop or personal computer. Additionally, the whole system can be controlled via an Android application, which may be more user friendly and convenient for weather monitoring purposes.

REFERENCES

[1] Tamilarasi B, Saravanakumar P, “Smart Sensor Interface for Environmental Monitoring in IoT”, International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE) - Volume 5, Issue 2, February 2016.

[2] Ruano AE, Mestre G, Duarte H & Silva S, “A neural network based intelligent weather station”, 2015 IEEE 9th International Symposium on Intelligent Signal Processing (WISP), Siena, Italy, 2015, pp. 1-6.

[3] Ayyappadas R, (Communication Systems), A.K.Kavitha, Dr.S.MaryPraveena, Dr.R.M.S.Parvathi, ,“Design and Implementation of Weather Monitoring System using Wireless Communication”, International Journal of Advanced Information in Engineering Technology (IJAIET) ISSN: 2454-6933 Vol.4, No.5, May 2017

[4] Shifeng Fang; Li Da Xu; Yunqiang Zhu; JiaerhengAhati; Huan Pei; Jianwu Yan; Zhihui Liu., “An integrated system for regional environmental monitoring and management based on internet of things”, IEEE Transactions on Industrial Informatics,vol.10, no.2, May-Jun. 2014, pp.1596-1605.

.

[5] J. Shah and B. Mishra, “IoT enabled environmental monitoring system for smart cities,” in 2016 International Conference on Internet of Things and Applications (IOTA), 2016, pp. 383–388.

[6] Kodali R & Mandal S, “IoT based weather station”, 2016 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT), Kumaracoil, India, 2016, pp. 680-683.

[7] Yashaswi Rahut, Rimsha Afreen, Divya Kamini, “Smart weather monitoring and real time alert system using IoT”, IRJET, Vol 5,Issue:10, Oct 2018.

[8] Arun Chakravarthy R, Bhuvaneswari M, Arun M, “IOT based environmental weather monitoring and farm information tracking system”, Journal of Critical Reviews, ISSN-2394-5125 Vol 7, Issue 7, April 2020.

[9] O. Chieochan, A. Saokaew and E. Boonchieng, "Internet of things (IOT) for smart solar energy: A case study of the smart farm at Maejo University," 2017 International Conference on Control, Automation and Information Sciences (ICCAIS), Chiang Mai, Thailand, 2017, pp. 262-267.

[10] J. G. Ramani, A. LakshmiPriya, S. Madhusudan, P. J. R. Kishore, M. Madhisha and U. Preethi, "Solar Powered Automatic Irrigation Monitoring System," 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 2020, pp. 293-297

[11]https://www.codeproject.com/Articles/845538/An-Introduction-to-ThingSpeak#:~:text=ThingSpeak%20is%20a%20platform%20providing,targeted%20for%20building%20IoT%20applications.&text=A%20channel%20stores%20the%20data,or%20from%20an%20embedded%20device.

[12] A. Zanella, N. Bui, A. Castellani, L. Vangelista and M. Zorzi, "Internet of Things for Smart Cities," in IEEE Internet of Things Journal, vol. 1, no. 1, pp. 22-32, Feb. 2014, doi: 10.1109/JIOT.2014.2306328.

[13] C. L. Stergiou, K. E. Psannis and B. B. Gupta, "IoT-Based Big Data Secure Management in the Fog Over a 6G Wireless Network," in IEEE Internet of Things Journal, vol. 8, no. 7, pp. 5164-5171, 1 April1, 2021, doi: 10.1109/JIOT.2020.3033131.

[14]https://www.arrow.com/en/research-and-events/articles/buck-boost-converters-the-versatile-voltage-component

[15]https://learn.adafruit.com/welcome-to-adafruit-io?embeds=allow

[16]https://www.theengineeringprojects.com/2018/10/introduction-to-arduino-ide.html

[17]https://mqtt.org/getting-started/

[18]https://en.wikipedia.org/wiki/Temperature

[19]https://en.wikipedia.org/wiki/Humidity

[20]https://en.wikipedia.org/wiki/Dew_point

[21]https://en.wikipedia.org/wiki/Heat_index

[22]https://www.weather.gov/safety/heat-index

[23]https://www.elprocus.com/a-brief-on-dht11-sensor/

[24]https://courses.lumenlearning.com/physics/chapter/13-3-the-ideal-gas-law/#:~:text=The%20ideal%20gas%20law%20states,T%20is%20its%20absolute%20temperature.

[25]https://en.wikipedia.org/wiki/Atmospheric_pressure

[26]https://www.elprocus.com/a-brief-on-dht11-sensor/#:~:text=This%20sensor%20is%20used%20in,people%20are%20affected%20by%20humidity.

[27]https://www.nationalgeographic.org/encyclopedia/altitude/#:~:text=Altitude%2C%20like%20elevation%2C%20is%20the%20distance%20above%20sea%20level.,8%2C000%20feet)%20into%20the%20atmosphere.

[28]https://wiki.seeedstudio.com/Grove-Barometer_Sensor-BMP280/#:~:text=Grove%20%2D%20Barometer%20Sensor%20(BMP280)%20is%20a%20breakout%20board%20for,approximate%20altitude%20of%20a%20place.

[29]https://components101.com/sensors/gy-bmp280-module

[30]https://cdn-shop.adafruit.com/datasheets/BST-BMP280-DS001-11.pdf

[31]https://components101.com/sensors/rain-drop-sensor-module

[32]https://components101.com/sensors/mq135-gas-sensor-for-air-quality

[33]https://cdn-shop.adafruit.com/datasheets/Si1145-46-47.pdf

[34]https://en.wikipedia.org/wiki/Air_quality_index

[35]https://scied.ucar.edu/learning-zone/air-quality/what-is-air-quality

[36]https://www.elprocus.com/mq-135-alcohol-sensor-circuit-and-working/#:~:text=An%20alcohol%20sensor%20detects%20the,which%20is%20suitable%20for%20breathalyzers.

[37]https://components101.com/modules/neo-6mv2-gps-module

[38]https://en.wikipedia.org/wiki/Ultraviolet_index

[39]https://en.wikipedia.org/wiki/Global_Positioning_System

[40]https://lastminuteengineers.com/neo6m-gps-arduino-tutorial/

[41]https://components101.com/development-boards/nodemcu-esp8266-pinout-features-and-datasheet#:~:text=The%20NodeMCU%20ESP8266%20development%20board,to%20store%20data%20and%20programs.

[42]https://www.make-it.ca/nodemcu-arduino/nodemcu-details-specifications/

[43]https://www.gatevidyalay.com/machine-learning-workflow-process-steps/

[44]http://www.sthda.com/english/articles/40-regression-analysis/167-simple-linear-regression-in-r/

[45]https://www.halvorsen.blog/documents/technology/iot/thingspeak/thingspeak.php

[46]https://www.section.io/engineering-education/introduction-to-adafruit-io/

[47]https://lastminuteengineers.com/neo6m-gps-arduino-tutorial/

[48]https://www.allaboutcircuits.com/projects/how-to-perform-uv-monitoring-with-the-adafruit-si1145/

[49]https://alibaba-cloud.medium.com/table-store-time-series-data-storage-architecture-f686e85cf259

[50]https://components101.com/modules/xl6009-dc-dc-step-down-voltage-regulator-module

[51]https://www.goodenergy.co.uk/how-do-solar-panels-work/

[53]https://acqnotes.com/acqnote/careerfields/link-budget

[54]https://en.wikipedia.org/wiki/Printed_circuit_board

[56]https://www.edgefx.in/advantages-using-printed-circuit-board-pcb/

[57]https://electrosome.com/easyeda-cloud-pcb-design-software/