ABSTRACT

The term internet of things (IOT) is considered as an important and popular technology aspect nowadays. Lifestyle controllability based on (IOT) became considerably simpler and easier especially in the communicating approaches among the smart devices. The methodology of (IOT) allows accessing and controlling the devices anywhere and anytime. The tendency of this aspect is to communicate among various devices with respect to ESP8266 Node MCU module. The devices even can be controlled with respect to the employed ESP stations. The adeptness of data transfer among the proposed remote locations depends mainly on the behavior of the system while the security and the applicability of the system are considered more efficient. The system proposed in this project is to control the humidity and temperature in the real crop field or
In a greenhouse using sensors and IoT devices by foggers and flooding in the crop field due to heavy rainfall which has effects on quality of crops. This work shares the information of air quality, Temperature and Humidity in crop fields at remote locations with each other based on ESP module through webserver. In addition, two control actions are taken automatically in accordance with temperature, humidity and air purity output results. To minimize the manual activities to watch over the crops all time and to automate the crop protection using temperature and humidity sensors in this IoT project. Furthermore, these control actions even can be applied manually by webserver administrative person. Finally, the ESP based system is classified as inexpensive project comparing to GSM module.

Index Terms— Smart Crop Protection System, Control Based ESP8266 Node MCU, Control system design, Automatic Control

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

ABBREVATION	EXPANSION
IOT	INTERNET OF THINGS.
MCU	MICRO CONTROLLER UNIT.
ESP	EXTENDED SERVICES PROCESSOR.

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

The climate controlling is most important aspects for the better health and yield of crop. So, weather monitoring is an important key aspect of crop protection in natural calamaties. A weather station is a technology that collects data related to the weather & environment using different electronics sensors. There are two types of weather station, one who is having their sensors and the second type of weather station is where we pull data from the weather station servers. In this project, we are designed by our weather station. We all know that a weather station is not a single device, but it is a combination of many small tools to form a larger system. It contains various sensors and gadgets that work together but in specific ways to transmit proper and accurate data of the weather parameters. It is quite tricky to uses of WEB server-based weather station to non-technical peoples, so we are providing web server-based user interface as well as Android application. We are well known today most mobile units running on Android OS, and many peoples are well known to use the android phone. So, our application is beneficial for such purpose. This device is all about IOT based Live Weather data Monitoring Using Node MCU ESP8266. We will interface DHT11 Humidity & Temperature Sensor, BMP180 Barometric Pressure Sensor and FC37 Rain Sensor with Node MCU ESP8266-12E Wi-Fi Module.

A Weather station is an innovation that gathers information identified with the climate and climate utilizing extraordinary gadgets sensors of numerous little apparatuses to shape a bigger framework. It contains different sensors and contraptions that cooperate however in explicit manners to send legitimate and precise information of the climate boundaries g Using Node MCU ESP8266. We will interface DHT11 Humidity and Temperature Sensor, and FC37 Rain Sensor with Node MCU ESP8266-12E Wi-Fi Module. The Rain Sensor Module's Sensing Pad comprises of two nickel- covered arrangement copper tracks. Likewise, it has two Header sticks; these are inside associated with the two copper tracks of the Sensing Pad. These pins are utilized to associate the Sensing Pad to the sensor module circuit through two jumper wire. Continuously, one pin of the downpour sensor circuit gives a +5v power supply to the one track of the detecting cushion, and another pin is gotten the return power supply from another track of the detecting cushion. Regularly under dry conditions, the detecting cushion gives high opposition and low conductive. In this way, the 5v force supply can't be passed starting.

1.2 PROBLEM STATEMENT

IOT systems are interconnected and communicate over networks. So, the system offers little control despite any security measures, and it can lead to various kinds of network attacks.

An IOT application is used here to monitor the crop which helps to monitor the environmental condition of any area of crop field, and with the help of the internet everyone can view the condition. It is the foremost concern while connecting devices, applications and cloud platforms. Connected devices that provide useful front and information are extremely valuable. But poor connectivity becomes a challenge where IOT sensors are required to monitor process data and supply information.

CHAPTER 2

METHODOLOGY

2.1 EXISTING SYSTEM

Weather monitoring of the environmental condition of any local area or a surrounding area of a crop field has been monitored using ESP8266 via IOT cloud. There are several IOT cloud platform available on internet such as android IOT, blink version 1.0, blink version 2.0. These are the available platform to connect through the cloud. The rain sensors and the pressure sensors libraries has been updated. They connect these sensors through these cloud platforms and see the output through cloud communication.

2.2 PROPOSED SYSTEM

The working of the system is starts with collection of the data from the sensors and the data are processed and transfer through node MCU module to the cloud platform thing speak there we can view the output of the sensors in the form of a output console with more accuracy and data transfer speed is greater compare to the other IOT cloud platform and also connectivity of the node MCU is more stronger than the other MCU. When this system is powered on, the Node board connects to the algorithm development system called MATLAB through the Thing speak cloud. Then, values are obtained from the sensors. Also, these values are sent to the thing speak app using the internet. Then, we can see the values as visualization on the screen.

CHAPTER 3

WORKING OF SYSTEM

3.1 SYSTEM ARCHITECTURE

Temperature sensor (DHT11) Pressure sensor (BMP180) ,Rain sensor (rain drop module) these sensor were the inputs connected to the Wi-Fi module ESP8266 which act as the controller and the Wi-Fi connecting module which is based on the node MCU the ESP8266 is connected to the thing speak cloud IOT platform through mobile hotspot we can able to view outputs of the sensors.

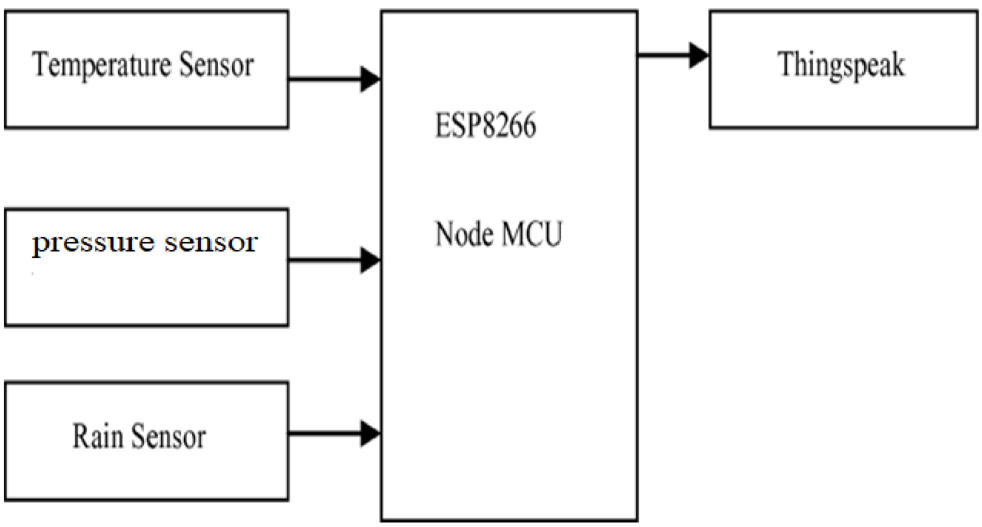

Fig 3.1 SYSTEM ARCHITECTURE

3.2 TEMPERATURE SENSOR

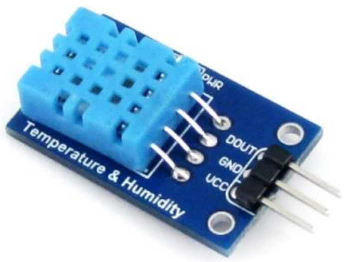

Fig 3.2 TEMPERATURE SENSOR

DHT11 element is strictly calibrated in the laboratory that is extremely accurate on humidity calibration. The calibration coefficients are stored as programmes in the OTP memory, which are used by the sensor’s internal signal detecting process. The single-wire serial interface makes system integration quick and easy. Its small size, low power consumption and up to 20 meters signal transmission making it the best choice for various applications, including those most demanding ones. The component is 4-pin single row pin package. It is convenient to connect, and special packages can be provided according to users’ request.

3.3 TECHNICAL SPECIFICATION

Item	Measurement Range	Humidity Accuracy	Temperature Accuracy	Resolution	Package
DHT11	20-90%RH 0-50 ℃	±5%RH	±2℃	1	4 Pin Single Row

TABLE 1 Technical Specification

Parameters	Conditions	Minimum	Typical	Maximum
Humidity				
Resolution		1%RH	1%RH	1%RH
			8 Bit	
Repeatability			±1%RH	
Accuracy	25℃		±4%RH	
	0-50℃			±5%RH
Interchangeability	Fully Interchangeable			
Measurement Range	0℃	30%RH		90%RH
	25℃	20%RH		90%RH
	50℃	20%RH		80%RH
Response Time (Seconds)	1/e(63%)25℃，1m/s Air	6 S	10 S	15 S
Hysteresis			±1%RH	
Long-Term Stability	Typical		±1%RH/year	
Temperature				
Resolution		1℃	1℃	1℃
		8 Bit	8 Bit	8 Bit
Repeatability			±1℃	
Accuracy		±1℃		±2℃
Measurement Range		0℃		50℃
Response Time (Seconds)	1/e(63%)	6 S		30 S

TABLE 2 Parameters

3.4 TYPICAL APPLICATION

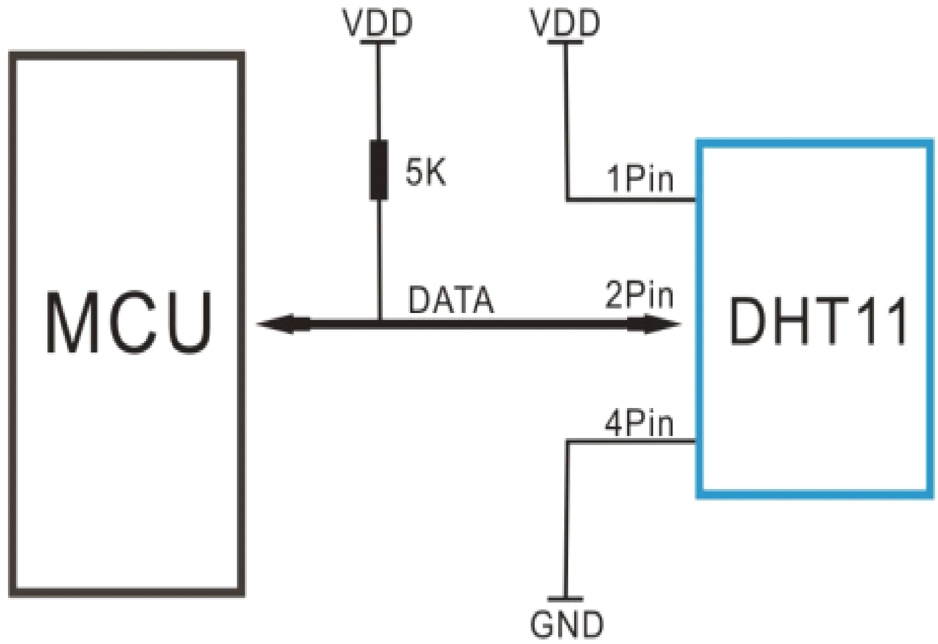

Fig 3.3 Typical Application

Note: 3Pin – Null; MCU = Micro-computer Unite or single chip Computer

When the connecting cable is shorter than 20 metres, a 5K pull-up resistor is recommended; when the connecting cable is longer than 20 metres, choose a appropriate pull-up resistor as needed.

POWER AND PIN: DHT11’s power supply is 3-5.5V DC. When power is supplied to the sensor, do not send any instruction to the sensor in within one second in order to pass the unstable status. One capacitor valued 100nF can be added between VDD and GND for power filtering.

3.5 PRRESSURE SENSOR

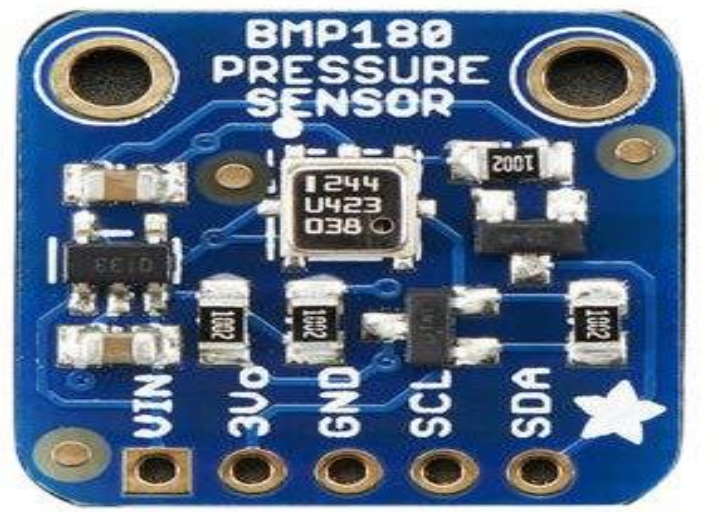

Fig 3.4 Pressure sensor

The BMP180 is the new digital barometric pressure sensor of Bosch Sensor Tec, with a very high performance, which enables applications in advanced mobile devices, such as smart phones, tablet PCs and sports devices.

3.6 BMP180 SENSOR FEATURES

- Can measure temperature and altitude.
- Pressure range: 300 to 1100hPa
- High relative accuracy of ±0.12hPa
- Can work on low voltages

- 3.4Mhz I2C interface
- Low power consumption (3uA)
- Pressure conversion time: 5msec
- Portable size

Pin Name	Description
VCC	Connected to +5V
GND	Connected to ground.
SDA	Serial Data pin (I2C interface)
SCL	Serial Clock pin (I2C interface)
3.3	VIf +5V is not present. Can power module by connecting +3.3V to this pin.

TABLE 3 Pin configuration

3.7 BMP180 SENSOR SPECIFICATIONS

- Operating voltage of BMP180: 1.3V – 3.6V
- Input voltage of BMP180MODULE: 3.3V to 5.5V
- Peak current: 1000uA
- Consumes 0.1uA standby
- Maximum voltage at SDA, SCL : VCC + 0.3V
- Operating temperature: -40ºC to +80ºC

3.8 TECHNICAL DATA

Technical Data	BMP180
Pressure range	300 ... 1100 hPa
RMS noise expressed in pressure	0.06 hPa, typ. (ultra low power mode) 0.02 hPa, typ (ultra high resolution mode)
RMS noise expressed in altitude	0.06 hPa, typ. (ultra low power mode)
	0.02 hPa, typ. (ultra high resolution mode)
Relative accuaracy pressure V_{DD} = 3.3 V	950 ... 1050 hPa @ 25 °C ±0.12 hPa ±1.0 m
	700 ... 900 hPa 25 ... 40 °C ±0.12 hPa ±1.0 m
Absolute accuracy p=300 ... 1100hPa (T=0 ... +65°C,	Pressure: -4.0 ... +2.0hPa Temperature: -1 hPa (+/- 1 hPa), typ.
Average current consumption (1Hz data refresh rate)	3 μA, typ. (ultra-low power mode)
	32 μA, typ. (advanced mode)
Peak current	650 μA, typ.
Stand-by current	0.1 μA, typ.
Supply voltage V_{DDIO} Supply voltage V_{DD}	1.62 ... 3.6 V 1.8 ... 3.6 V
Operation temp. range full accuracy	-40 ... +85 °C 0 ... +65 °C
Pressure conv. time	-5 msec, typ. (std. mode)
I²C date transfer rate	3.4 MHz, max.
Package type / pin no.	LGA / 7
Package dimensions	3.6 x 3.8 x 0.93 mm³

TABLE 4 Technical data

3.9 RAIN SENSOR

Fig 3.5 Rain sensor

Rain Sensors also called raindrop sensors are very handy sensors that are used in a variety of use cases. Alone a rain sensor can only detect if it is raining and how strong it rains but in combination with other electrical devices you can build useful applications.

For me the most useful application is to detect an open window when it starts to rain because I am often not sure if I closed a window after I left the house.

The rain sensor consists of 2 components, which we consider in more detail below:

- Rain drop module to detect if it is raining or not
- Control board to process the data from the rain drop module

3.10 Functionality of the Rain Drop Module

The rain drop module is a printed circuit board in a rectangular shape. The size can differ between models, but the construction is the same for each model.

The ground plate of the circuit board consists of fibre reinforced plastic that is not conductive. On top of this ground plate there are two pins mounted. From each of these pins starts one conductor track, build like an “E” with an offset against the other conductor track. Therefore, the tracks are not connected but the conductor tracks are close together.

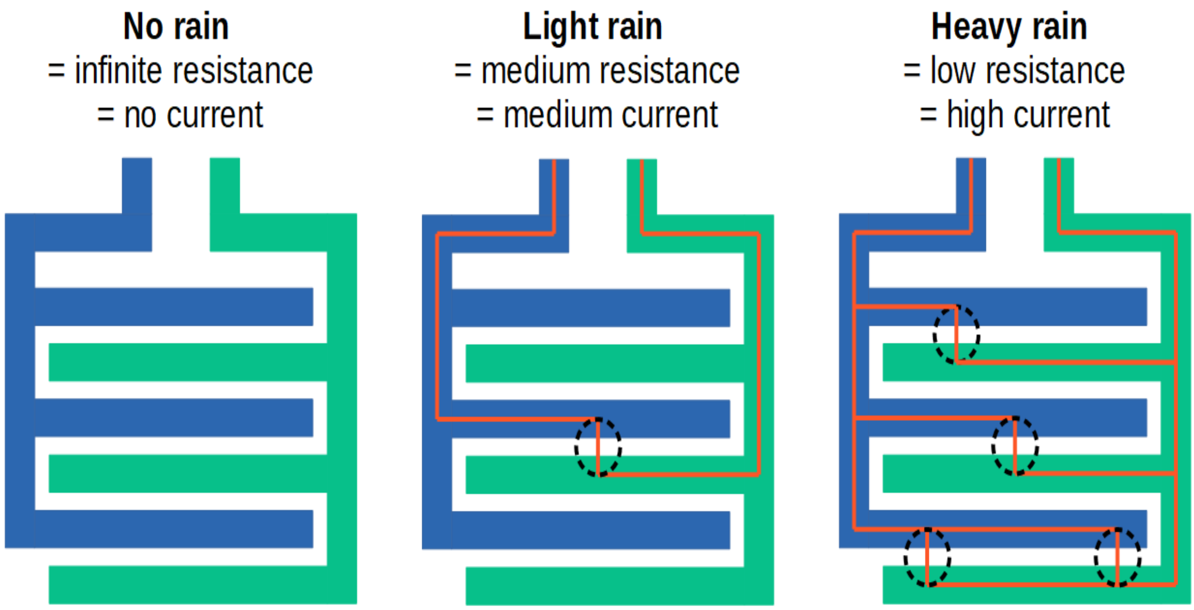

Fig 3.6 Functionality of the Rain Drop Module

3.11 Functionality of the Control Board of a Rain Sensor

The control board consists of two input pins and four output pins. The input pins are connected to the rain board and the output pins to your favourite microcontroller, for example an Arduino Uno or an ESP32 Node MCU.

On the control board you find multiple resistors that also functions are the voltage divider to provide an analogue signal for the rain intensity. Therefore as input we get a resistance from the rain board and the control board converts this resistance into a voltage drop between the analogue pin and ground. The microcontroller uses the internal analogue to digital converter (ADC) to convert the voltage from the analogue pin to a digital value between 0 and 1023 that can be printed to the serial output in your Arduino IDE.

The biggest part on the control board is the potentiometer to adjust the sensitivity of the rain detector. The potentiometer is only a variable resistor whose resistance is changed with the setting wheel at the top. We need this potentiometer to compare the resistance of the potentiometer with the resistance of the rain board. If the resistance of the rain board is lower than the threshold, defined by the potentiometer, the digital output of the control board changes from 1 HIGH to 0 LOW.

3.12 PIN CONFIGURATION OF RAIN SENSOR

S. No:	Name	Function
1	VCC	Connects supply voltage- 5V
2	GND	Connected to ground
3	D0	Digital pin to get digital output
4	A0	Analogue pin to get analogue output

TABLE 5 Pin Configuration

Most rain sensor implementations employ an infrared light that is beamed at a 45-degree angle onto the windshield from inside the car. If the glass is wet, less light makes it back to the sensor.

3.13 ESP8266 (NODE MCU)

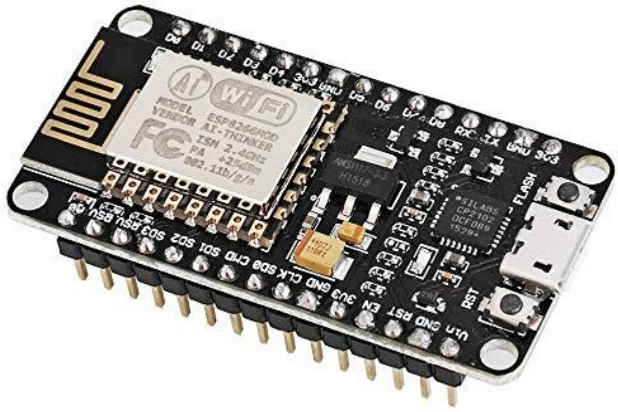

Fig 3.7 ESP8266

ESP8266 is a Wi-Fi SOC (system on a chip) produced by Express if Systems . It is a highly integrated chip designed to provide full internet connectivity in a small package.

ESP8266 can be used as an external Wi-Fi module, using the standard AT Command set Firmware by connecting it to any microcontroller using the serial UART, or directly serve as a Wi-fi enabled micro controller, by programming a new firmware using the provided SDK.

The GPIO pins allow Analog and Digital IO, plus PWM, SPI, I2C, etc.

This board has been around for almost a year now, and has been used mostly in IOT contexts, where we want to add connectivity for example to an Arduino project. A wide adoption has been facilitated by the very modest price, ranging from 2.50 to 10 USD depending on the features offered by the manufacturers.

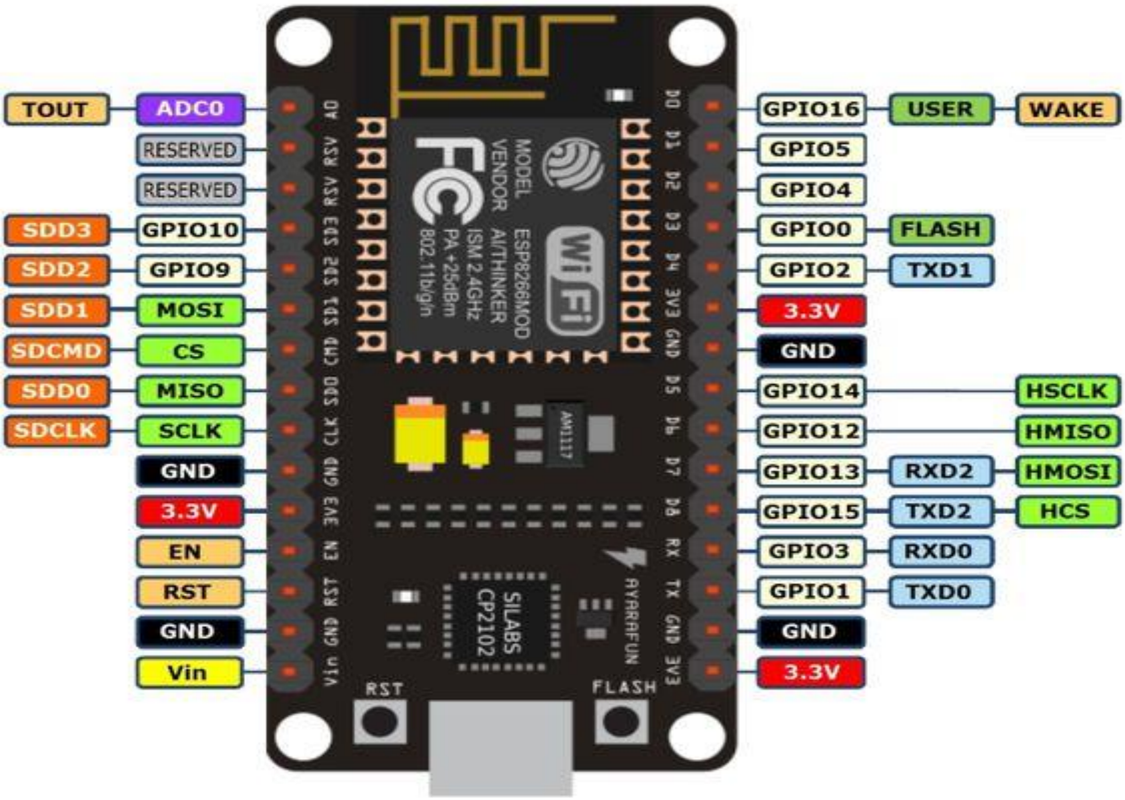

Fig 3.8 ESP8266 (NODE MCU)

3.14 TECHNICAL FEATURES

- 802.11 b / g / n
- Wi-Fi Direct (P2P), soft-AP
- Built-in TCP / IP protocol stack
- Built-in TR switch, Balun, LNA, power amplifier and matching network
- Built-in PLL, voltage regulator and power management components
- 802.11b mode + 19.5dBm output power
- Built-in temperature sensor
- Support antenna diversity
- off leakage current is less than 10uA
- Built-in low-power 32-bit CPU: can double as an application processor
- SDIO 2.0, SPI, UART
- STBC, 1×1 MIMO, 2×1 MIMO
- A-MPDU, A-MSDU aggregation and the 0.4 Within wake

- 2ms, connect and transfer data packets
- standby power consumption of less than 1.0mW (DTIM3)

3.15 ADVANTAGE OF DIRECT PROGRAMMING OF ESP8266

- In order to program the ESP directly you need to install a tool chain and firmware upload utility.
- The ESP8266 Wi-Fi explains the process in detail.
- You should use the ESP Open SDK to build your tool chain.
- Express if also provides an SDK containing the AT Firmware and the proprietary libraries.
- Finally, a ready to use Virtual Box virtual machine is available here
- As an alternative you might want to program the ESP using the Arduino libraries. A port of the Arduino IDE 1.6.x is available here .

3.16 CLOUD COMPUTING

Cloud computing is the delivery of different services through the Internet. These resources include tools and applications like data storage, servers, databases, networking, and software.

Rather than keeping files on a proprietary hard drive or local storage device, cloud-based storage makes it possible to save them to a remote database. As long as an electronic device has access to the web, it has access to the data and the software programs to run it.

Cloud computing is a popular option for people and businesses for several reasons including cost savings, increased productivity, speed and efficiency, performance, and security.

3.17 KEY TAKE AWAYS OF CLOUD COMPUTING

- Cloud computing is the delivery of different services through the Internet, including data storage, servers, databases, networking, and software.
- Cloud-based storage makes it possible to save files to a remote database and retrieve them on demand.

- Services can be both public and private—public services are provided online for a fee while private services are hosted on a network to specific clients.

3.18 Types of Cloud Services

Regardless of the kind of service, cloud computing services provide users with a series of functions including:

- Email
- Storage, backup, and data retrieval
- Creating and testing apps
- Analysation of data
- Audio and video streaming
- Delivering software on demand

Cloud computing is still a new service but is being used by a number of different organizations from big corporations to small businesses, non-profits to government agencies, and even individual consumers.

3.19 TYPES OF CLOUD COMPUTING

1. **Software-as-a-service (SAAS)** involves the licensure of a software application to customers. Licenses are typically provided through a pay-as-you-go model or on-demand. This type of system can be found in Microsoft Office's 365.
2. **Infrastructure-as-a-service (IAAS)** involves a method for delivering everything from operating systems to servers and storage through IP-based connectivity as part of an on**-demand service. Clients can avoid the need to purchase software or servers, and instead** procure these resources in an outsourced on-demand service. Popular examples of the IAAS system include IBM Cloud and Microsoft Azure.
3. **Platform-as-a-service (PAAS)** is considered the most complex of the three layers of cloud-based computing. PAAS shares some similarities with SAAS, the primary difference being that instead of delivering software

online, it is a platform for creating software that is delivered via the Internet. This model includes platforms like Salesforce.com

3.20 THINGSPEAK IOT CLOUD

Thing Speak is an IOT analytics platform service that allows you to aggregate, visualize, and analyse live data streams in the cloud. You can send data to Thing Speak from your devices, create instant visualization of live data, and send alerts.

- Collect
- Send sensor data privately to the cloud.

- Analyse
- Analyse and visualize your data with MATLAB
- Act
- Trigger a reaction.

3.21 THINGSPEAK FEATURES

Collect data in private channels.

- **Share data with public channels.**
- **REST ful and MQTT APIs.**
- **MATLAB® analytics and visualizations.**
- **Event scheduling.**
- **Alerts**
- **App integrations.**

WORKS WITH

- **MATLAB® & Simulink®.**
- **Arduino®.**
- **Particle devices.**
- **ESP8266 and ESP32 Modules.**
- **Raspberry Pi™.**
- **LoRaWAN®.**
- **Things Network.**
- **Senet.**
- **Libelium.**
- **Beckhoff.**

THINGSPEAK CLOUD

Thing Speak is **an IOT analytics platform service that allows you to aggregate, visualize, and analyse live data streams in the cloud**. You can send data to Thing Speak from your devices, create instant visualization of live data, and send alerts.

3.22 THINGSPEAK USED FOR

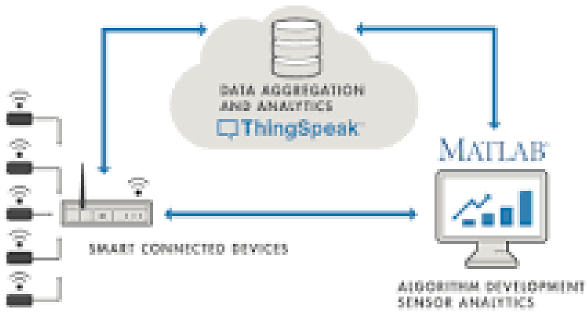

Fig 3.9 Thing speak Used For

ThingSpeak provides instant visualizations of data posted by your devices to Thing Speak. With the ability to execute MATLAB® code in Thing Speak you can perform online analysis and processing of the data as it comes in. Thing Speak is often used for proto typing and proof of concept IOT systems that require analytics.

3.23 THINGSPEAK IS A DATABASE

At the heart of Thing Speak is a time-series database. Thing Speak provides users with free time-series data storage in channels. Each channel can include up to eight data field

CIRCUIT DIAGRAM AND CONNECTIONS

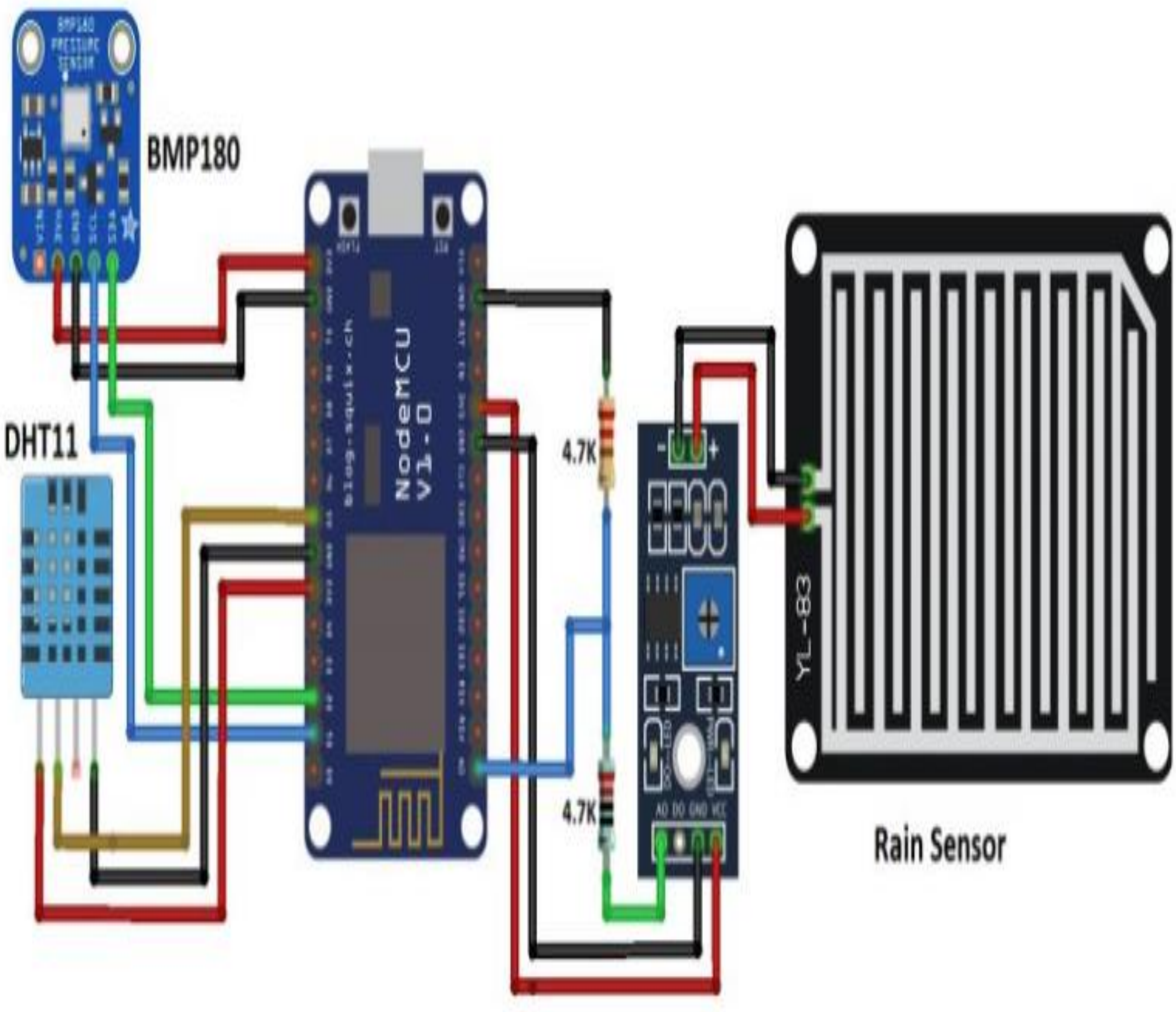

Fig 3.10 Circuit Diagram

CHAPTER 4
EXPERIMENTAL ANALYSIS

4.1 SYSTEM CONFIGURATION

4.1.1 Software requirements

- **Firstly library files has to be included for the sensors**

```
/*Weather monitoring system with Thing speak.
  Created by the SriTu Hobby team.
  http://srituhobby.com
*/

#include <SFE_BMP180.h>
#include <Wire.h>
#include <ESP8266WiFi.h>
#include "DHT.h"
```

- **Secondly, objects and variables are created for these libraries.**

```
DHT dht(D3, DHT11);
SFE_BMP180 bmp;
double T, P;
char status;
WiFiClient client;
```

- **Thirdly, let’s include the apiKey and WI-FI connection details.**

```
String apiKey = "0AKC9Y6NSYAB0DHE";
const char *ssid =  "Nooby";
const char *pass =  "nirmalnaveen";
const char* server = "api.thingspeak.com";
```

- **In the setup function,**

```
void setup() {
  Serial.begin(115200);
  delay(10);
  bmp.begin();
  Wire.begin();
  dht.begin();
  WiFi.begin(ssid, pass);



  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected");
}
```

- **The loop function includes all the main functions. These are described below.**

```
void loop() {
  //BMP180 sensor
  status =  bmp.startTemperature();
  if (status != 0) {
    delay(status);
    status = bmp.getTemperature(T);
```

```
  status = bmp.startPressure(3);// 0 to 3
  if (status != 0) {
    delay(status);
    status = bmp.getPressure(P, T);
    if (status != 0) {

    }
  }
}

//DHT11 sensor
float h = dht.readHumidity();
float t = dht.readTemperature();

if (isnan(h) || isnan(t)) {
  Serial.println("Failed to read from DHT sensor!");
  return;
}

//Rain sensor
int r = analogRead(A0);
r = map(r, 0, 1024, 0, 100);


if (client.connect(server, 80)) {
  String postStr = apiKey;
  postStr += "&field1=";
  postStr += String(t);
  postStr += "&field2=";
  postStr += String(h);
  postStr += "&field3=";
```

```
    postStr += String(P, 2);
    postStr += "&field4=";
    postStr += String(r);
    postStr += "\r\n\r\n\r\n\r\n";

    client.print("POST /update HTTP/1.1\n");
    client.print("Host: api.thingspeak.com\n");
    client.print("Connection: close\n");
    client.print("X-THINGSPEAKAPIKEY: " + apiKey + "\n");
    client.print("Content-Type: application/x-www-form-urlencoded\n");
    client.print("Content-Length: ");
    client.print(postStr.length());
    client.print("\n\n\n\n");
    client.print(postStr);

    Serial.print("Temperature: ");
    Serial.println(t);
    Serial.print("Humidity: ");
    Serial.println(h);
    Serial.print("absolute pressure: ");
    Serial.print(P, 2);
    Serial.println("mb");
    Serial.print("Rain");
    Serial.println(r);

  }
  client.stop();
  delay(1000);
}
```

CHAPTER 5

RESULT

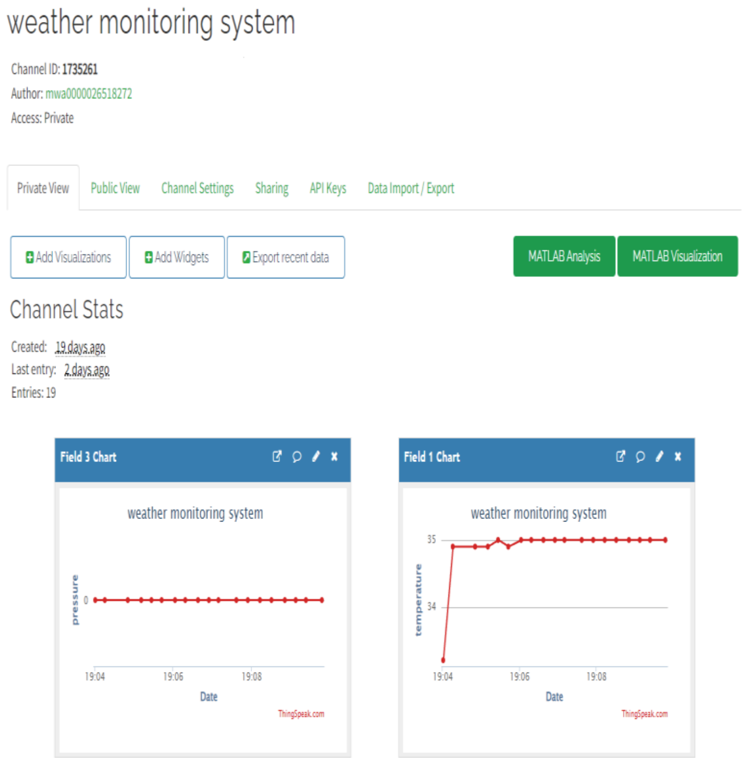

Fig 5.1 Output

CHAPTER 6

CONCLUSION AND FUTURE WORK

Processing the data from the input of the rain sensor, temperature sensor and pressure sensor here we can build a mini weather station for an better understanding of the weather in the surrounding by getting the input from sensors to the ESP8266 we can connect the MCU to cloud connecting to the thing speak IOTS platform.in future we can upgrade it by using PIR sensor for more accuracy and durability. Switching to the ESP8266 to the esp32 for better connectivity. We can also change the IOT platform for our convenience here we can move to Arduino IOT cloud a new platform introduced for IOT projects which has been created specific for the Arduino controllers and node MCU.

REFERENCE

1.H. Jain and R. Jain, "Big data in weather forecasting: Applications and Challenges", *International Conference on Big Data Analytics And Computational Intelligence (ICBDAC)*, pp. 138-142, 2017.
Show in Context View Article Full Text: PDF (168) Google Scholar

2.M. Patil, S.R. Pachpande, JP. Chaudari and K.P. Rane, "Study of Literature on Weather Monitoring System", *International Journal of Computer Application*, vol. 153, no. 3, November 2016.
Show in Context Google Scholar

3.R. K. Kodali and S. Mandal, "IoT based weather station", *International Conference on Control Instrumentation Communication and Computational Technologies (ICCICCT)*, pp. 680-683, 2016.
Show in Context View Article Full Text: PDF (352) Google Scholar

4.R. K. Kodali and K. S. Mahesh, "A low-cost implementation of MQTT using ESP8266", *International Conference on Contemporary Computing and Informatics (IC3I)*, pp. 404-408, 2016.
Show in Context View Article Full Text: PDF (542) Google Scholar

5.Y. Gunardi, A. Adriansyah and T. Anindhito, "Small smart community: An application of internet of things", *Journal of Engineering and Applied Sciences*, vol. 10, pp. 6341-6347, 2015.
Google Scholar

6.G. Solano and J. Tarrillo, "Monitoring weather parameters from difficult access places", *IEEE XXVI International Conference on Electronics Electrical Engineering and Computing (INTERCON)*, pp. 1-4, 2019.
Show in Context View Article Full Text: PDF (859) Google Scholar

7.J. P. Guruprasadh et al., "Intelligent soil quality monitoring system for judicious irrigation", *International Conference on Advances in Computing Communications and Informatics (ICACCI)*, pp. 443-448, 2017.
Show in Context View Article Full Text: PDF (740) Google Scholar

8.P. Nagarajan et al., "Biological treatment of domestic wastewater by selected aquatic plants", *International Conference on Technological Advancements in Power and Energy (TAP Energy)*, pp. 1-4, 2017.
Show in Context View Article Full Text: PDF (454) Google Scholar

9.V. S. Babu, U. A. Kumar, R. Priyadharshini, K. Premkumar and S. Nithin, "An intelligent controller for smart home", *International Conference on Advances in Computing Communications and Informatics (ICACCI)*, pp. 2654-2657, 2016.
Show in Context View Article Full Text: PDF (476) Google Scholar

10.M. V. Ramesh et al., "Water quality monitoring and waste management using IoT", *IEEE Global Humanitarian Technology Conference (GHTC)*, pp. 1-7, 2017.
Show in Context View Article Full Text: PDF (516) Google Scholar

11.A. Malhotra, S. Som and S. K. Khatri, "IoT Based Predictive Model for Cloud Seeding", *2019 AmityInternational Conference on Artificial Intelligence (AICAI)*, pp. 669-773, 2019.
Show in Context View Article Full Text: PDF (198) Google Scholar

12.R. Praveen Kumar and S. Smys, "A novel report on architecture protocols and applications in Internet of Things (IoT)", *2nd International Conference on Inventive Systems and Control (ICISC)*, pp. 1156-1161, 2018.