IoT Based AQI Monitoring System

A Mini Project Report Submitted

in Partial Fulfillment of the Requirements

for the Degree of

BACHELORE OF TECHNOLOGY

in

Electronics & Communication Engineering

by

MOHD TABISH HUSAIN

Under the Supervision of

Mr. Amit Singh & Dr. Ghanshyam Singh
(Assistant Professors)
(Department of Elec. & Comm. Engg)

Dr. Digvijay Singh Chauhan
(Associate Professor & H.O.D.)
(Department of Elec. & Comm. Engg)

Feroze Gandhi Institute of Engineering and Technology,

Raebareli

Dec, 2022

DECLARATION

I hereby declare that the work presented in this report entitled "IoT Based AQI Monitoring System", was carried out by me. I have not submitted the matter embodied in this report for the award of any other University or Institute. I have given due credit to the original authors/sources for all the words, ideas, diagrams, graphics, computer programs, experiments, results, that are not my original contribution. I have used quotation marks to identify verbatim sentences and given credit to the original authors/sources.

I affirm that no portion of my work is plagiarized, and the experiments and the results reported in the report are not manipulated. In the event of a complaint of plagiarism and the manipulation of the experiments and results, I shall be fully responsible and answerable.

Mohd Tabish Husain

Place: Raebareli

Date: __ __ / __ __ / __ __ __ __

CERTIFICATE

This is to certify that the project work presented in this report entitled **"IoT Based AQI Monitoring System"** is the bonafide work carried out by **Mohd Tabish Husain,** student of B.Tech in the department of Electronics and Communication Engineering, Feroze Gandhi Institute of Engineering and Technology (FGIET), Ratapur, Raebareli, affiliated to Dr. A. P. J Abdul Kalam Technical University (AKTU), Lucknow, Uttar Pradesh, during the academic year 2022 – 2023 in partial fulfillment of the requirements for the degree of Bachelor of Technology in Electrical Engineering and this project has not submitted previously for the award of any other degree, diploma and fellowship.

Signature of Guide
Mr. Amit Singh
(Assistant Professor)
(Department of Elec. & Comm. Engg.)

Signature of Guide
Dr. Ghanshyam Singh
(Assistant Professor)
(Department of Elec. & Comm. Engg.)

Signature of H.O.D
Dr. Digvijay Singh
(Associate Professor & H.O.D)
(Department of Elec. & Comm. Engg.)

ACKNOWLEDGEMENT

It is my great fortune that I have got an opportunity to carry out this project work under the supervision of Mr. Amit Singh and Dr. Ghanshyam Singh in the Department of Electronics and Communication Engineering, Feroze Gandhi Institute of Engineering and Technology (FGIET), Ratapur, Raebareli, affiliated to Dr. A. P. J Abdul Kalam Technical University (AKTU), Lucknow, Uttar Pradesh. I express my sincere thanks and deepest sense of gratitude to my guides for their constant support, unparalleled guidance and limitless encouragement. I wish to convey my gratitude to Dr. Digvijay Singh Chauhan, HOD, Department of Electronics and Communication Engineering, FGIET and the authority of FGIET for providing all kinds of the infrastructural facility for the research work. I would also like to convey my gratitude to all the faculty members and staffs of the Department of Electronics and Communication Engineering, FGIET for their wholehearted cooperation to make this work turn into reality.

ABSTRACT

Air pollution is one of the biggest threats to the present-day environment. Everyone is being affected by air pollution day by day including humans, animals, crops, cities, forests and aquatic ecosystems. Besides that, it should be controlled at a certain level to prevent the increasing rate of global warming. This project aims to design an IOT-based air pollution monitoring system using the internet from anywhere using a computer or mobile to monitor the air quality of the surroundings and environment. There are various methods and instruments available for the measurement and monitoring quality of air. The IoT-based air pollution monitoring system would not only help us to monitor the air quality but also be able to send alert signals whenever the air quality deteriorates and goes down beyond a certain level.

In this system, NodeMCU plays the main controlling role. It has been programmed in a manner, such that, it senses the sensory signals from the sensors and shows the quality level via led indicators. Besides the harmful gases (such as CO_2, CO, smoke, etc) temperature and humidity can be monitored through the temperature and humidity sensor by this system. Sensor responses are fed to the NodeMCU which displays the monitored data in the ThingSpeak cloud which can be utilized for analyzing the air quality of that area.The following simple flow diagram (as shown in Fig. 1) indicates the working mechanism of the IoT-based Air Pollution Monitoring System.

TABLE OF CONTENTS

List of Acronyms

DHT	Digital Humidity and Temperature
IoT	Internet of Things
PPM	Parts Per Molecule
PM	Particulate Matter
CO	Carbon Monoxide
$\mu g/m^3$	Micro Gram per Metric Cube
CO2	Carbon Dioxide
LED	Light Emitting Diode
LPG	Liquid Petroleum Gas
IDE	Integrated Development Environment

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 AIM OF PROJECT

Air is getting polluted because of the release of toxic gases by industries, vehicle emissions and increased concentration of harmful gases and particulate matter in the atmosphere.

The level of pollution is increasing rapidly due to factors like industries, urbanization, increase in population, vehicle use which can affect human health. Particulate matter is one of the most important parameters having a significant contribution to the increase in air pollution. This creates a need for measurement and analysis of real-time air quality monitoring so that appropriate decisions can be taken in a timely period.

This project presents real-time standalone air quality monitoring. Internet of Things (IoT) is nowadays finding profound use in each and every sector, plays a key role in our air quality monitoring system too. The setup will show the air quality in PPM on the webpage so that we can monitor it very easily.

In this IoT project, we can monitor the pollution level from anywhere using your computer or mobile.

This IoT Based Air Quality Index Monitoring System will upload the AQI data over a Thingspeak server using the Wi-fi module. We will use an PM2.5 GP2Y1010AU0F Dust Smoke Particle Sensor that can detect the level of various air pollutant.

This PM2.5 GP2Y1010AU0F Dust Smoke Particle Sensor is an infrared emitting diode (IRED) and a phototransistor are diagonally arranged into this device. It detects the reflected light of dust in the air. Especially, it is effective to detect very fine particle like the cigarette smoke. In addition, it can distinguish smoke from house dust by the pulse pattern of the output voltage.

The module is mainly used for dust removal equipment alarm equipment, air purification equipment, dust robots, fire alarm, etc. industry equipment can detect smoke particles, pollen spores, and other particles.

We will interface PM2.5 GP2Y1010AU0F Dust Smoke Particle Sensor with ESP8266 Microcontroller Module (WiFi).

Major metro cities like Delhi, Mumbai, Bangalore etc. have different agencies working to keep check on air quality but tier 3 cities like Raebareli, do not get that kind of attention. That is why we need a low-cost AQI Monitoring system that works well in different air conditions and has the capability to detect most polluting agents.

The main objective of this project is to provide an AQI Monitoring system which will be easy to configure and handle

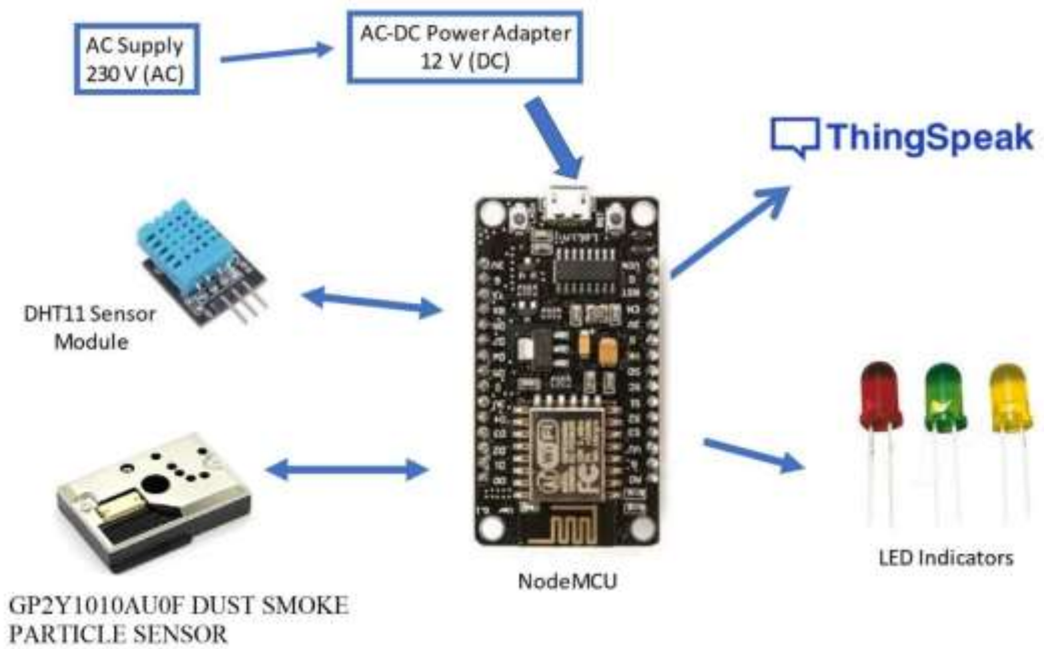


Fig. 1 IoT Based AQI Monitoring System

1.3 LITERATURE REVIEW

The explanation of the Air Quality Index (AQI) and its standard ranges are taken from [1]. From 0-100 ppm the atmosphere is safe for living. If the ppm level increases above 100 then it moves out of the safety zone. If the ppm value rises above 200 then it becomes extremely dangerous for human life. The DHT11 sensor module is used to measure the temperature and the humidity of the surroundings [2]. The PM2.5 GP2Y1010AU0F dust smoke particle sensor is used to measure the air quality of the surroundings [3]. It can be calibrated with respect to fresh air, alcohol, carbon dioxide, hydrogen and methane. In this project, it has been calibrated with respect to fresh air [9], [10].

In [4] the controlling action of NodeMCU has been described. This research has shown the uses of C as the programming language for scripting the software code. It has an inbuilt Wi-Fi module which allows the project to implement IoT easily. Arduino IDE is used to implement the coding part of the project [5], [8]. ThingSpeak cloud is used for the cloud service. It has a free version which requires a delay of 15 seconds to upload an entry in the cloud [6], [7]. As this project uses two sensors, of which DHT11 has internal heater elements and withdraw more power(P=V*I), so though both sensors are turned ON, their output voltage levels vary and show unpredictable values due to insufficient power drive. So, we used a separate voltage regulator IC for the DHT11 sensor as NodeMCU alone is not sufficient to drive two sensors [9].

CHAPTER 2

THEORY AND DESCRIPTION OF THE COMPONENTS

2.1 WHAT IS AQI AND AQI IN INDIA?

An Air Quality Index (AQI) is a number used by government agencies to measure the air pollution levels and communicate it to the population. As the AQI increases, it means that a large percentage of the population will experience severe adverse health effects. The measurement of the AQI requires an air monitor and an air pollutant concentration over a specified averaging period. The results are grouped into ranges, and each range is assigned a descriptor, a color code and a standardized public health advisory.

Historically, all Replica watch continents have been using the AQI established by the United States Environmental Protection Agency (EPA), which is a piecewise linear function of the pollutant concentration. If multiple pollutants are measured at a monitoring site, then the biggest AQI value in one hour average is reported for that location. However, air pollution in each country is very much specific to the country pollution type.

The pollution sources in India – and in most asian countries – are numerous and incompletely understood. In Delhi, for example, it comes mostly from light and heavy duty vehicle traffic emissions, road dust, solid fuel combustion for heating and cooking, biomass, waste burning, thermal power plants, diesel generators, construction and small-scale local industries. That's why each country has to have their own AQI values.

The National Air Quality Index Standard (NAQI) in India was launched by The Minister for Environment, Forests & Climate Change, Shri Prakash Javadekar, on 17 September 2014. The initiative constitutes part of the Government's mission to introduce the "culture of cleanliness", as the air pollution has been a huge concern in the country, especially in urban areas. The National Air Monitoring Program (NAMP), that covers 240 cities in the country, has been operated by the Central Pollution Control Board (CPCB) and developed by the Indian Institute of Technology, Kanpur (IIT), providing data in public domain, on real time basis.

To understand how the AQI works, we just have to know the six range categories (Good, Satisfactory, Moderately Polluted, Poor, Very Poor, and Severe) and check the associated health impacts. Solutions should be taken based on the AQI Category and the pollutant associated to it, as follows:

AQI Category (Range)	PM_{10} 24-hr	$PM_{2.5}$ 24-hr	NO_2 24-hr	O_3 8-hr	CO 8-hr (mg/m^3)	SO_2 24-hr	NH_3 24-hr	Pb 24-hr
Good (0-50)	0-50	0-30	0-40	0-50	0-1.0	0-40	0-200	0-0.5
Satisfactory (51-100)	51-100	31-60	41-80	51-100	1.1-2.0	41-80	201-400	0.6 –1.0
Moderate (101-200)	101-250	61-90	81-180	101-168	2.1- 10	81-380	401-800	1.1-2.0
Poor (201-300)	251-350	91-120	181-280	169-208	10.1-17	381-800	801-1200	2.1-3.0
Very poor (301-400)	351-430	121-250	281-400	209-748*	17.1-34	801-1600	1201-1800	3.1-3.5
Severe (401-500)	430 +	250+	400+	748+*	34+	1600+	1800+	3.5+

Think of the AQI as a yardstick that runs from 0 to 500. The higher the AQI value, the greater the level of air pollution and the greater the health concern. For example, an AQI value of 50 represents good air quality with little potential to affect public health, while an AQI value over 300 represents hazardous air quality.

EPA calculates the AQI for 8 major air pollutants regulated by the Clean Air Act: PM10, PM2.5, NO2(Nitrogen Dioxide), O3(Ozone), CO(Carbon Monoxide), SO2(Sulfur Dioxide), NH3(Ammonia) and Pb(Lead). For each of these pollutants, EPA has established national air quality standards to protect public health. Ground-level ozone and airborne particles are the two pollutants that pose the greatest threat to human health in our country.

2.2 WHAT IS IOT?

The Internet of Things (IoT) describes the network of physical objects – “things” – that are embedded with sensors, software, and other technologies for the purpose of connecting and exchanging data with other devices and systems over the internet. These devices range from ordinary household objects to sophisticated industrial tools.

The field has evolved due to the convergence of multiple technologies, including ubiquitous computing, commodity sensors, increasingly powerful embedded systems, and machine learning.

Traditional fields of embedded systems, wireless sensor networks, control systems, automation (including home and building automation), independently and collectively enable the Internet of things. In the consumer market, IoT technology is most synonymous with products pertaining to the concept of the "smart home", including devices and appliances (such as lighting fixtures, thermostats, home security systems, cameras, and other home appliances) that support one or more common ecosystems, and can be controlled via devices associated with that ecosystem, such as smartphones and smart speakers. IoT is also used in healthcare systems.

There are a number of concerns about the risks in the growth of IoT technologies and products, especially in the areas of privacy and security, and consequently, industry and governmental moves to address these concerns have begun, including the development of international and local standards, guidelines, and regulatory frameworks.

IoT devices are a part of the larger concept of home automation, which can include lighting, heating and air conditioning, media and security systems and camera systems. Long-term benefits could include energy savings by automatically ensuring lights and electronics are turned off or by making the residents in the home aware of usage.

A smart toilet seat that measures blood pressure, weight, pulse and oxygen levels. A smart home or automated home could be based on a platform or hubs that control smart devices and appliances. For instance, using Apple's HomeKit, manufacturers can have their home products and accessories controlled by an application in iOS devices such as the iPhone and the Apple Watch. This could be a dedicated app or iOS native applications such as Siri. This can be demonstrated in the case of Lenovo's Smart Home Essentials, which is a line of smart home devices that are controlled through Apple's Home app or Siri without the need for a Wi-Fi bridge. There are also dedicated smart home hubs that are offered as standalone platforms to connect different smart home products and these include the Amazon Echo, Google Home, Apple's HomePod, and Samsung's SmartThings Hub. In addition to the commercial systems, there are many non-proprietary, open-source ecosystems; including Home Assistant, OpenHAB and Domoticz.

Significant numbers of energy-consuming devices (e.g. lamps, household appliances, motors, pumps, etc.) already integrate Internet connectivity, which can allow them to communicate with utilities not only to balance power generation but also helps optimize the energy consumption as a whole. These devices allow for remote control by users, or central management via a cloud-based interface, and enable functions like scheduling (e.g., remotely powering on or off heating systems, controlling ovens, changing lighting conditions, etc.).The smart grid is a utility-side IoT application; systems gather and act on energy and power related information to improve the efficiency of the production and distribution of electricity. Using advanced metering infrastructure (AMI) Internet-connected devices, electric utilities not only collect data from end-users but also manage distribution automation devices like transformers.

Another example of integrating the IoT is Living Lab which integrates and combines research and innovation processes, establishing a public-private-people-partnership. There are currently 320 Living Labs that use the IoT to collaborate and share knowledge between stakeholders to co-create innovative and technological products. For companies to implement and develop IoT services for smart cities, they need to have incentives. The governments play key roles in smart city projects as changes in policies will help cities to implement the IoT which provides effectiveness, efficiency, and accuracy of the resources that are being used. For instance, the government provides tax incentives and cheap rent, improves public transport, and offers an environment where start-up companies, creative industries, and multinationals may co-create, share a common infrastructure and labor markets, and take advantage of locally embedded technologies, production process, and transaction costs. The relationship between the technology developers and governments who manage the city's assets is key to providing open access to resources to users in an efficient way.

In this project, we have tried to implement the concept of IoT to monitor the temperature, humidity and air quality of the surroundings

2.3 COMPONENTS USED

❖ HARDWARE COMPONENTS

1. ESP8266 Microcontroller module
2. DHT11 (Humidity & Temperature Sensor)
3. PM2.5 GP2Y1010AU0F dust smoke particle sensor
4. AMS1117(3.3V voltage regulator IC)
5. RGB LED
6. +9V battery
7. LM7805 (5V voltage regulator IC)
8. 220 μF capacitor (3)
9. 150K & 100R resistance
10. Printed circuit board (PCB)

❖ SOFTWARE COMPONENTS

1. ThinkSpeak Cloud
2. Arduino IDE

2.4 BRIEF DESCRIPTION OF THE COMPONENTS

❖ ESP8266 MICROCONTROLLER MODULE

ESP8266 Microcontroller module is an open-source ESP8266 development kit, armed with the CH340G USBTTL Serial chip. It has firmware that runs on ESP8266 Wi-Fi SoC from Espressif Systems. Whilst cheaper, CH340 is super reliable even in industrial applications. It is tested to be stable on all supported platforms as well. It can be simply coded in Arduino IDE. It has a very low current consumption between 15 µA to 400 mA.

The pinout Diagram of ESP8266 Microcontroller module is shown in Fig. 2.1.

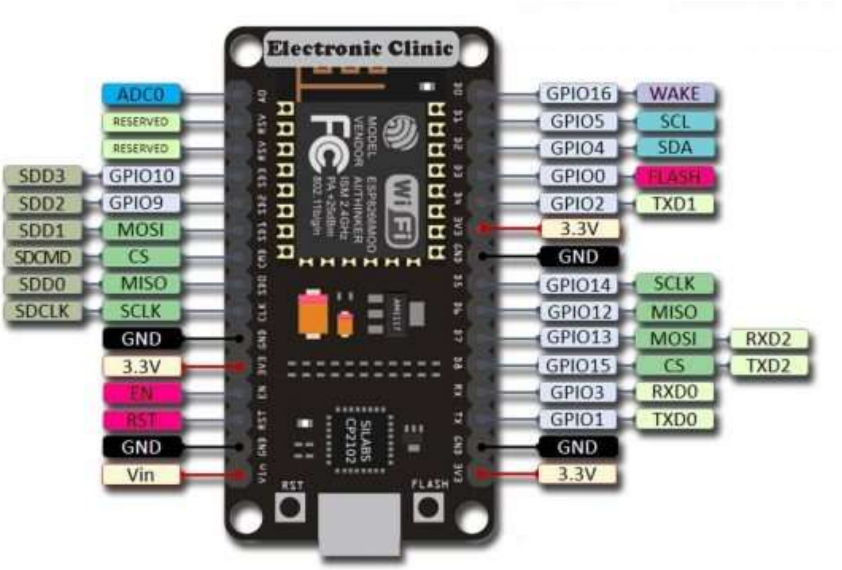


Fig. 2.1 Pinout Diagram of ESP8266 Microcontroller module

❖ DHT11 SENSOR MODULE

The DHT11 is a temperature and humidity sensor that gives digital output in terms of voltage. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air.

As shown in Fig. 2.2, we need to supply a voltage of 5V (DC) to the Vcc pin and ground it to the GND pin. The sensor output can be easily read from the Data pin in terms of voltage (in digital mode).

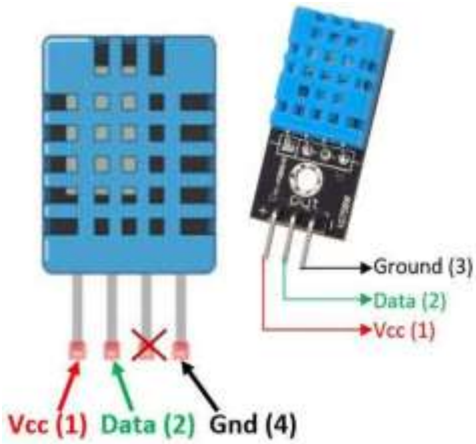


Fig 2.2 (Pinout Diagram of DHT11sensor)

Humidity Measurement: The humidity sensing capacitor has two electrodes with a moisture-holding substrate as a dielectric between them as shown in Fig 2.3. Change in the capacitance value occurs with the change in humidity levels. The IC measure, process these changed resistance values and then converts them into digital form.

Temperature Measurement: For measuring the temperature, the DHT11 sensor uses a negative temperature coefficient thermistor, which causes a decrease in its resistance value with an increase in temperature. To get a wide range of resistance values, the sensor is made up of semiconductor ceramics or polymers.

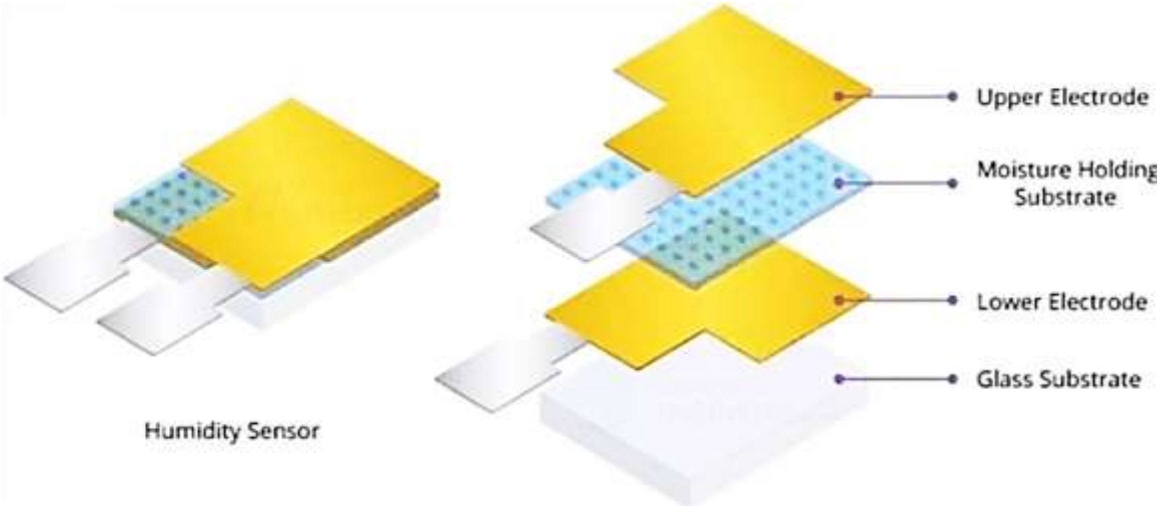


Fig 2.3 The structure of the humidity sensor

- **PM2.5 GP2Y1010AU0F DUST SMOKE PARTICLE SENSOR**

GP2Y1014AU0F is a tiny six-pin analog output optical air quality/optical dust sensor that is designed to sense dust particles in the air. It is very much Smaller in size. It detects the reflected light of dust in air. Especially, it is effective to detect very fine particle like the cigarette smoke. In addition it can distinguish smoke from house dust by pulse pattern of output voltage and is commonly used in air purifier systems. It works on the principle of laser scattering. Inside the sensor module, an infrared emitting diode and a photosensor are diagonally arranged near the air inlet hole as shown in Fig 2.4.

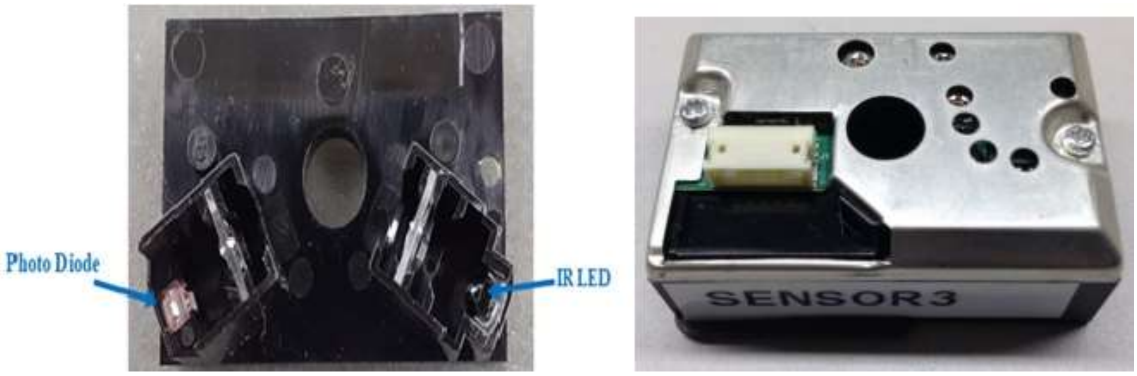


Fig 2.4 GP2Y1014AU0F with Internal view of vents

When air containing dust particles enters into the sensor chamber, the dust particles scatter the IR LED light towards the photo-detector. The intensity of the scattered light depends on the dust particles. The more dust particles in the air, the greater the intensity of light. Output voltage at the VOUT pin of the sensor changes according to the intensity of scattered light.

The sensor has a very low current consumption (20mA max, 11mA typical), and can be powered with up to 7VDC. The output of the sensor is an analog voltage proportional to the measured dust density, with a sensitivity of 0.5V/0.1mg/m3.

❖ AMS1117 (3.3V VOLTAGE REGULATOR IC)

AMS1117 3.3V, 1A, SOT-223 Voltage Regulator IC is a series of low dropout three -terminal regulators with a dropout of 3.3V at 1A load current. AMS 1117 features a very low standby current 2mA compared to 5mA of a competitor. Other than a fixed version, Vout = 1.2V, 1.5V, 1.8V, 2.5V, 3.3V, 5V, and 12V, AMS1117 has an adjustable version, which can provide an output voltage from 1.25 to 12V with only two external resistors.

Fig 2.5 AMS1117(3.3 V Voltage Regulator IC

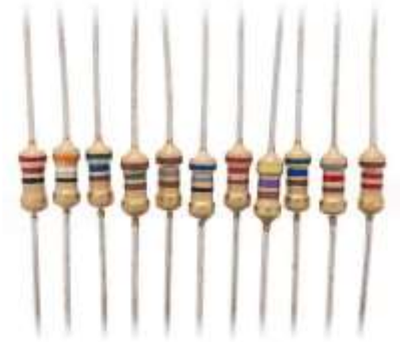
AMS 1117 offers thermal shut down and current limit functions, to assure the stability of chip and power system. And it uses a trimming technique to guarantee output voltage accuracy within 2%. Other output voltage accuracy can be customized on demand, such as 1%

❖ RGB LED

A light-emitting diode (LED) is a semiconductor light source that emits light when current flows through it. Electrons in the semiconductor recombine with electron holes, releasing energy in the form of photons. The colour of the light (corresponding to the energy of the photons) is determined by the energy required for electrons to cross the band gap of the semiconductor. White light is obtained by using multiple semiconductors or a layer of lightemitting phosphor on the semiconductor device. LEDs have many advantages over incandescent light sources, including lower power consumption, longer lifetime, improved physical robustness, smaller size, and faster switching. In exchange for these generally favourable attributes, disadvantages of LEDs include electrical limitations to low voltage and generally to DC (not AC) power, inability to provide steady illumination from a pulsing DC or an AC electrical supply source, and lesser maximum operating temperature and storage temperature. In contrast to LEDs, incandescent lamps can be made to intrinsically run at virtually any supply voltage, can utilize either AC or DC current interchangeably, and will provide steady illumination

Fig 2.6 RGB LED

when powered by AC or pulsing DC even at a frequency as low as 50 Hz. LEDs usually need electronic support components to function, while an incandescent bulb can and usually does operate directly from an unregulated DC or AC power source.



❖ 9V BATTERY

The 9V battery is an extremely common battery that was first used in transistor radios. It features a rectangular prism shape that utilizes a pair of snap connectors which are located at the top of the battery. A wide array of both large and small battery manufacturers produce versions of the 9V battery. Possible chemistries of primary (non-rechargeable) 9V batteries include Alkaline, Carbon-Zinc (Heavy Duty), Lithium. Possible chemistries of secondary (rechargeable) 9V batteries include nickel-cadmium (NiCd), nickel-metal hydride (NiMH), and lithium ion. The performance and application of the battery can vary greatly between different chemistries, meaning that some chemistries are better suited for some applications over others.

Fig 2.7 9V Battery

❖ LM7805 (5V VOLTAGE REGULATOR IC)

Voltage sources in a circuit may have fluctuations resulting in not providing fixed voltage outputs. A voltage regulator IC maintains the output voltage at a constant value. 7805 Voltage Regulator, a member of the 78xx series of fixed linear voltage regulators used to maintain such fluctuations, is a popular voltage regulator integrated circuit (IC).

Fig 2.8 LM7805 5V Voltage Regulator IC

Like most other regulators in the market, it is a three-pin IC; input pin for accepting incoming DC voltage, ground pin for establishing ground for the regulator, and output pin that supplies the positive 5 volts.

The xx in 78xx indicates the output voltage it provides. 7805 IC provides +5 volts regulated power supply with provisions to add a heat sink.

❖ RESISTORS

A resistor is a passive two-terminal electrical component that implements electrical resistance as a circuit element. In electronic circuits, resistors are used to reduce current flow, adjust signal levels, to divide voltages, bias active elements, and terminate transmission lines, among other uses. High-power resistors that can dissipate many watts of electrical power as heat may be used as part of motor controls, in power distribution systems, or as test loads for generators. Fixed resistors have resistances that only change slightly with temperature, time or operating voltage.

Fig 2.9 Resistors

❖ PCB

Printed circuit boards (PCBs) are the foundational building block of most modern electronic devices. Whether simple single layered boards used in your garage door opener, to the six layer board in your smart watch, to a 60 layer, very high density and high-speed circuit boards used in super computers and servers, printed circuit boards are the foundation on which all of the other electronic components are assembled onto.

Semiconductors, connectors, resistors, diodes, capacitors and radio devices are mounted to, and "talk" to one another through the PCB.

PCB's have mechanical and electrical attributes that make them ideal for these applications. Most PCB's manufactured in the World are rigid, roughly 90% of the PCB's manufactured today are rigid boards. Some PCB's are flexible, allowing the circuits to be bent and folded into shape, or sometimes they are used where the flexible circuit will survive hundreds of thousands of flex cycles, without any break in the circuits. These flexible PCB's comprise roughly 10% of the market. A small subset of these types of circuits are called rigid flex circuits, where one part of the board is rigid – ideal for mounting and connecting components, and one or more parts are flexible, providing the advantages of flexible circuits listed above.

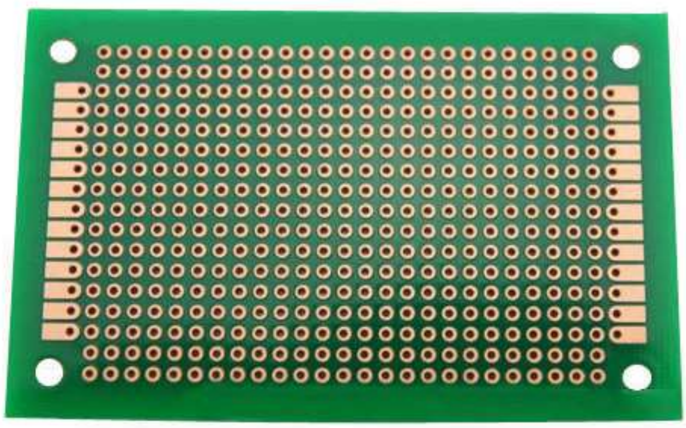


Fig 2.10 A Blank PCB

❖ THINGSPEAK CLOUD

ThingSpeak is open-source software written in Ruby which allows users to communicate with internet-enabled devices. It facilitates data access, retrieval and logging of data by providing an API to both the devices and social network websites. ThingSpeak was originally launched by ioBridge in 2010 as a service in support of IoT applications. ThingSpeak has integrated support from the numerical computing software MATLAB from MathWorks, allowing ThingSpeak users to analyse and visualize uploaded data using MATLAB without requiring the purchase of a MATLAB license from MathWorks.

Fig 2.11 ThingsSpeak Cloud

❖ ARDUINO IDE

The Arduino IDE is open-source software, which is used to write and upload code to the Arduino boards. The IDE application is suitable for different operating systems such as Windows, Mac OS X, and Linux. It supports the programming languages C and C++. Here, IDE stands for Integrated Development Environment. The program or code written in the Arduino IDE is often called sketching. We need to connect the Genuino and Arduino board with the IDE to upload the sketch written in the Arduino IDE software. The sketch is saved with the extension '.ino.'

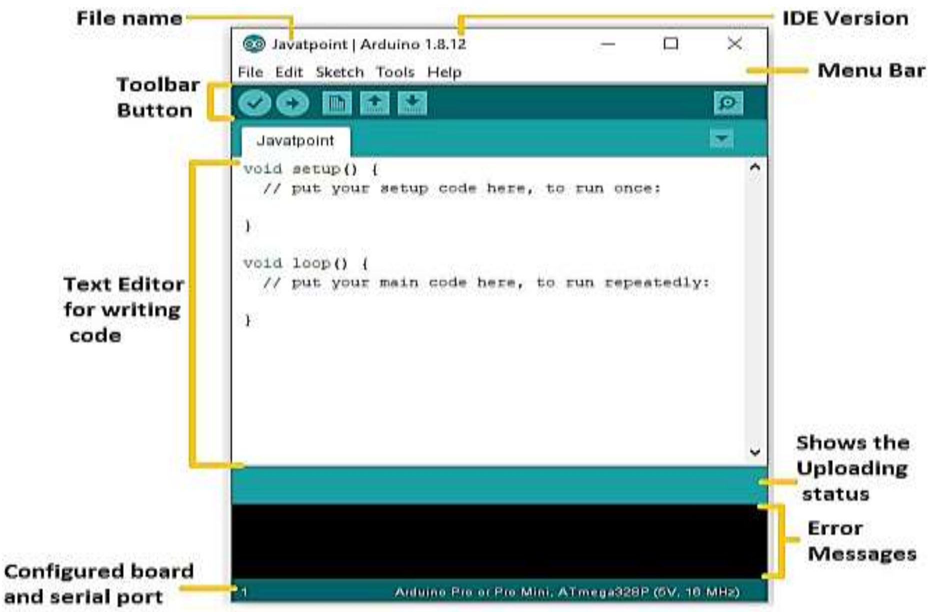


Fig 2.12 Arduino IDE

2.5 WORKING PROCEDURES

ESP8266 MCU plays the main controlling role in this project. It has been programmed in a manner, such that, it senses the sensory signals from the sensors and shows the quality level via led indicators. The DHT11 sensor module is used to measure the temperature and the humidity of the surroundings.With the help of the PM2.5 GP2Y1010AU0F dust smoke particle sensor module, air quality is measured in $\mu g/m^3$. These data are fed to the ThinkSpeak cloud over the internet. We have also provided LED indicators to indicate the safety levels.

STEP 1. Firstly, the calibration of the GP2Y1010AU0F dust smoke particle sensor module is done. The sensor is set to preheat for 2 minutes. Then the software code is uploaded to the NodeMCU followed by the hardware circuit to calibrate the sensor has been performed.

STEP 2. Then, the DHT11 sensor is set to preheat for 2 minutes.

STEP 3. The result of calibration found in STEP 1 is used to configure the final working code.

STEP 4. The final working code is then uploaded to the NodeMCU.

STEP 5. Finally, the complete hardware circuit is implemented.

The software codes and the hardware circuits are described in the following chapters.

CHAPTER 3

HARDWARE MODEL

3.1 HARDWARE MODEL TO PREHEAT DHT11 SENSOR MODEL

As discussed earlier, we need to preheat the DHT11 sensor so that it can work accurately. The following steps were performed to preheat the DHT11 sensor module:

STEP 1: The Vcc pin of the DHT11 sensor module was connected with the VU pin of NodeMCU.

STEP 2: The Gnd pin of the DHT11 sensor module was connected with the Gnd pin of NodeMCU.

STEP 3: The NodeMCU is powered with a 9V battery 2 minutes.

STEP 4: The setup was then disconnected

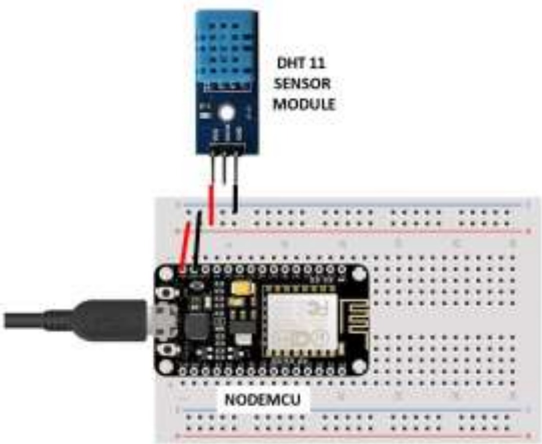


Fig 3.1 Circuit diagram to preheat the DHT11 Sensor module

3.2 HARDWARE MODEL TO PREHEAT AND CALIBRATE GP2Y1010AU0F DUST SMOKE PARTICLE SENSOR MODULE

The following steps were performed to preheat the GP2Y1010AU0F dust smoke particle sensor module

STEP 1 : The Vcc pin of the GP2Y1010AU0F dust smoke particle sensor module was connected with the VU pin of NodeMCU.

STEP 2 : The Gnd pin of the GP2Y1010AU0F dust smoke particle sensor module was connected with the Gnd pin of NodeMCU.

STEP 3 : The NodeMCU is powered with a 9V battery for a day.

STEP 4 : The setup was then disconnected.

Fig. 3.2 shown below describes the foresaid connections

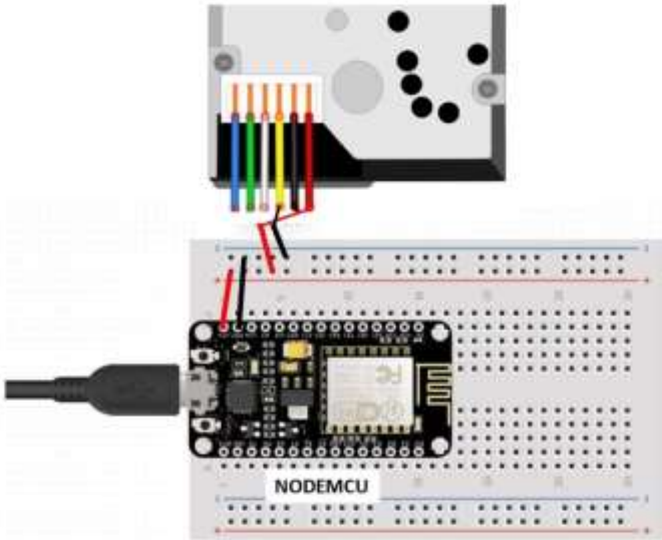


Fig 3.2 Circuit Diagram to Preheat the GP2Y1010AU0F dust smoke particle sensor

3.3 FINAL HARDWARE MODEL

The following steps were performed to execute the project

STEP 1 : 9V battery is connected to LM7805 voltage regulator IC which convert 9 v into 5V. This 5V supply goes to DHT11 Temperature & Humidity sensor. Also, this 5V supply goes to ASM 1117 3.3 V regulator IC which converts 5V into 3.3 V.

STEP 2 : ASM 1117 3.3 V IC is connected to ESP 8266 Microcontroller and this ASM 1117 3.3 V is connected to PM 2.5 Optical dust particle sensor(GP2Y1010AU0F).

STEP 3 : In PM2.5 dust particle sensor V-LED pin is connected to positive terminal 220uF capacitor and 150 ohm resistor. LED GND pin is connected to ground of the circuit and negative terminal of 220uf capacitor is also connected to ground. LED pin is connected to digital pin of general purpose input output pins 4(D2) which work as enabler and disabler of the IR LED.

STEP 4 : The analog DATA pin of the GP2Y1010AU0F dust smoke particle sensor module was connected with the ADC Pin of the NodeMCU. Pin 4(S GND) is use as ground pin which connected to ground of the circuit. Pin 6 (VCC) is connected to 3.3V of circuit.

STEP 5 : The DATA pin of the DHT11 sensor module was connected with the D0 pin of the NodeMCU.

STEP 6 : The anode of the RGB LED indicator was connected to the D4 pin of the NodeMCU respectively.

STEP 7 : The software code to execute the project was then uploaded to the NodeMCU.

STEP 8 : The setup was then powered with 9V Battery.

It can be now turned ON/OFF as per the requirements. Fig 3.4 represents the circuit diagram of the setup

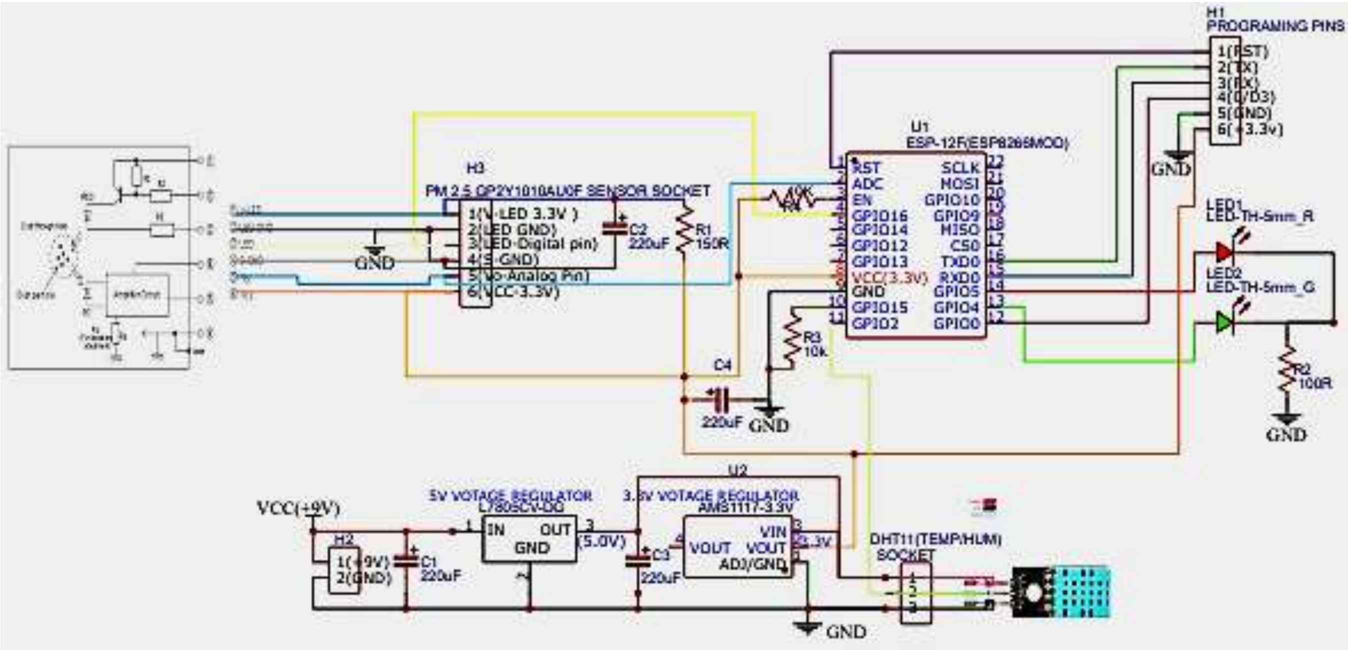
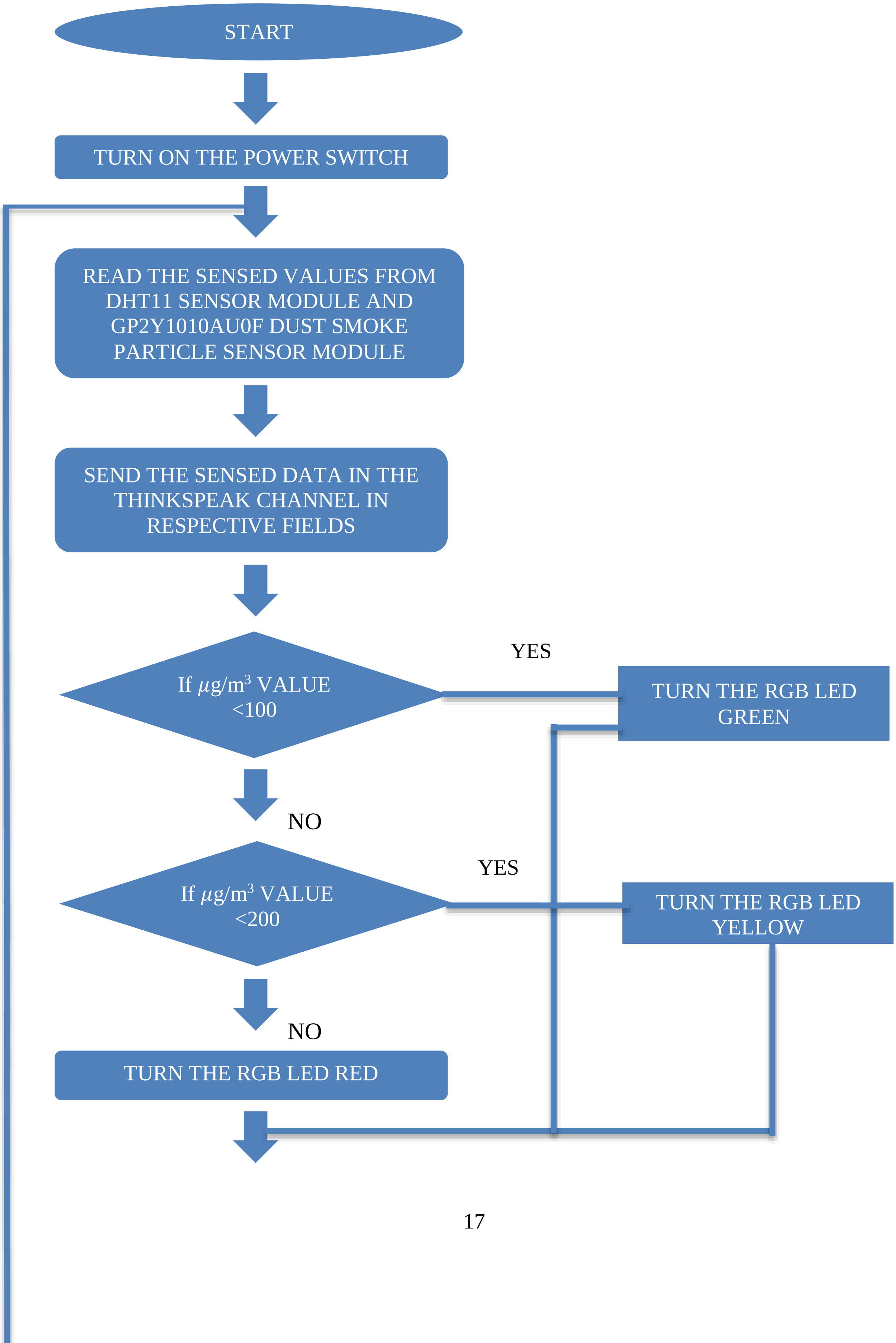


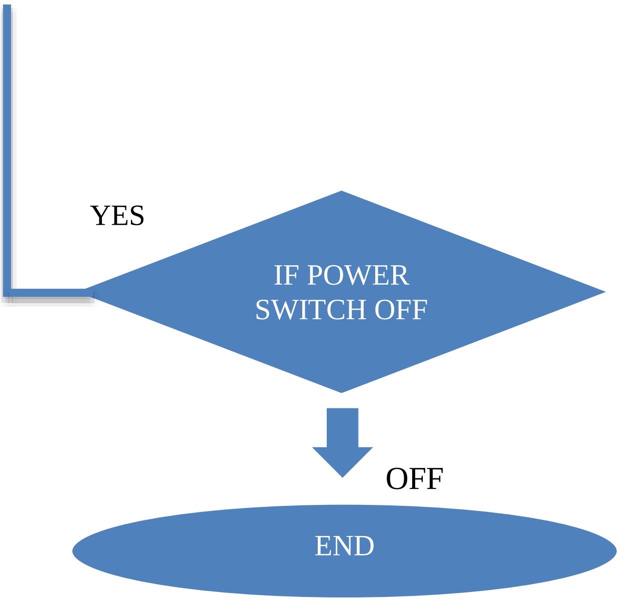
Fig 3.3 Circuit Diagram of the setup

CHAPTER 4

SOFTWARE IMPLEMENTATION

4.1 WORKING ALGORITHM





4.2 EXECUTION OF MAIN PROGRAM

```
#include <ESP8266WiFi.h>
#include <GP2YDustSensor.h>
#include <Adafruit_Sensor.h>
#include <DHT.h>
#define DHTPIN 13 //Pin D7
#define R 14 //Pin D5
#define G 15 //Pin D8
#define B 12 //Pin D6
const uint8_t SHARP_LED_PIN = 2; // Sharp Dust/particle sensor Led Pi---Pin
D4
const uint8_t SHARP_VO_PIN = A0; // Sharp Dust/particle analog out pin used
for reading
#define DHTTYPE DHT11 // DHT 11
GP2YDustSensor dustSensor(GP2YDustSensorType::GP2Y1010AU0F, SHARP_LED_PIN,
SHARP_VO_PIN);
DHT dht(DHTPIN, DHTTYPE);
uint32_t delayMS;
int PM2_5VALUE = 0;
float h,t,f,hic,hif;
String apiKey = "WFGLSL6I3D9S12H9";
const char* ssid = "XXXXXXXXXX"; //your ssid(hostpot name)
const char* pass = "XXXXXXXXXXX"; //your ssid pasword
const char* server = "api.thingspeak.com";
WiFiClient client;
void setup() {
pinMode(R, OUTPUT);
pinMode(G, OUTPUT);
pinMode(B, OUTPUT);
digitalWrite(R, 1);
digitalWrite(G, 1);
digitalWrite(B, 1);
Serial.begin(9600);
```

```
delay(10);
Serial.println();
Serial.println("Connecting to ");
Serial.println(ssid);
WiFi.begin(ssid, pass);
while (WiFi.status() != WL_CONNECTED)
{
delay(500);
Serial.println(".");
}
Serial.println("");
Serial.println("WiFi connected");
Serial.println(WiFi.localIP());
//dustSensor.setBaseline(0.4); // set no dust voltage according to your own
experiments
//dustSensor.setCalibrationFactor(1.1); // calibrate against precision
instrument
dustSensor.begin();
// Initialize device.
dht.begin();
}
void loop()
{
AQI();
HT();
API();
if(PM2_5VALUE > 0 && PM2_5VALUE < 30)
{
analogWrite(R, 880);
analogWrite(G, 900);
analogWrite(B, 1010);
}
else if(PM2_5VALUE > 30 && PM2_5VALUE < 60)
{
analogWrite(R, 1024);
analogWrite(G, 800);
analogWrite(B, 1024);
}
else if(PM2_5VALUE > 60 && PM2_5VALUE < 90)
{
analogWrite(R, 0);
analogWrite(G, 900);
analogWrite(B, 1010);
}
else if(PM2_5VALUE > 90 && PM2_5VALUE < 120)
{
analogWrite(R, 0);
```

```
analogWrite(G, 1010);
analogWrite(B, 1024);
}
else if(PM2_5VALUE > 120 && PM2_5VALUE < 250)
{
analogWrite(R, 0);
analogWrite(G, 1020);
analogWrite(B, 1024);
}
else if(PM2_5VALUE > 250)
{
analogWrite(R, 0);
analogWrite(G, 1024);
analogWrite(B, 1024);
}
}
void AQI()
{
PM2_5VALUE = dustSensor.getRunningAverage();
Serial.print("Dust density: ");
Serial.print(dustSensor.getDustDensity());
Serial.print(" µg/m3; Running average: ");
Serial.print(dustSensor.getRunningAverage());
Serial.println(" µg/m3");
delay(1000);
}
void HT()
{
// Wait a few seconds between measurements.
delay(2000);
// Reading temperature or humidity takes about 250 milliseconds!
// Sensor readings may also be up to 2 seconds 'old' (its a very slow
sensor)
h = dht.readHumidity();
// Read temperature as Celsius (the default)
t = dht.readTemperature();
// Read temperature as Fahrenheit (isFahrenheit = true)
f = dht.readTemperature(true);
// Check if any reads failed and exit early (to try again).
if (isnan(h) || isnan(t) || isnan(f))
{
Serial.println(F("Failed to read from DHT sensor!"));
return;
}
// Compute heat index in Fahrenheit (the default)
hif = dht.computeHeatIndex(f, h);
// Compute heat index in Celsius (isFahreheit = false)
```

```
hic = dht.computeHeatIndex(t, h, false);
Serial.print(F("Humidity: "));
Serial.print(h);
Serial.print(F("% Temperature: "));
Serial.print(t);
Serial.print(F("°C "));
Serial.print(f);
Serial.print(F("°F Heat index: "));
Serial.print(hic);
Serial.print(F("°C "));
Serial.print(hif);
Serial.println(F("°F"));
}
void API()
{
if (client.connect(server,80)) // "184.106.153.149" or api.thingspeak.com
{
String postStr = apiKey;
postStr +="&field1=";
postStr += String(PM2_5VALUE);
postStr +="&field2=";
postStr += String(hic);
postStr +="&field3=";
postStr += String(h);
postStr += "\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");
client.print(postStr);
}
client.stop();
}
```

CHAPTER 5

RESULTS

The working of the designed prototype has been investigated for the 3 sets of experiments as described in the following sections

EXPERIMENT 1:

Aim: To demonstrate the working of the system in smoky atmosphere.

Experimental Condition: The experiment was performed in the presence of smoke coming from an incense stick placed near the setup.

Observations in ThinkSpeak Cloud

Fig 5.1 Observations for Experiment 1

Setup:

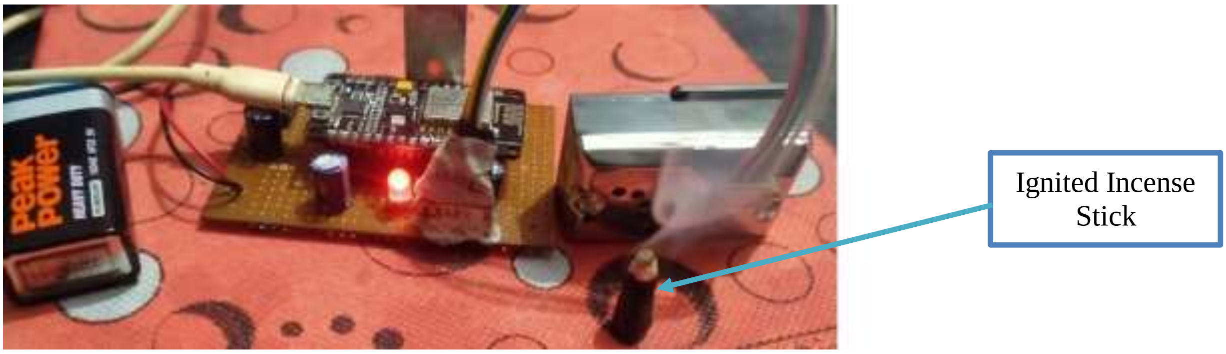


Fig 5.2 Setup for Experiment 1

Conclusion:

We can observe from the results that the presence of smoke near the setup can be easily detected by the system. We have taken the reference from the Accuweather.com for verifying the values. It matched with a +0.74 error with the temperature data, -2 error with the humidity data and +7 error with the $\mu g/m^3$ data. Hence, it can be concluded that we can detect the presence of smoke with the help of this monitoring system.

EXPERIMENT 2:

Aim: To demonstrate the working of the system in cold and cloudy atmosphere.

Experimental Condition: The experiment was performed on a cold cloudyy day in a local outdoor area.

Observations in ThinkSpeak Cloud:

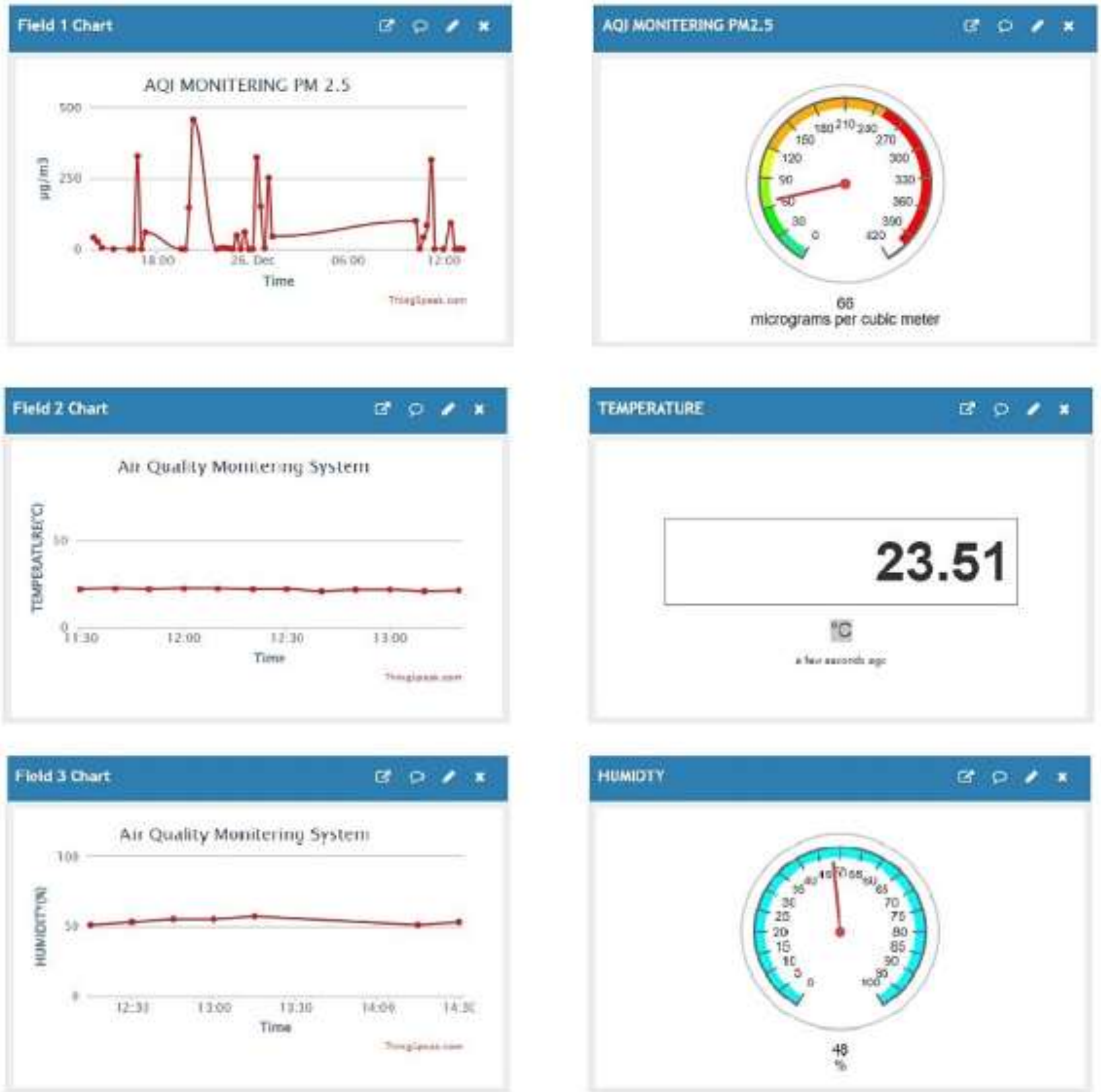


Fig 5.3 Observations for Experiment 2

Setup:

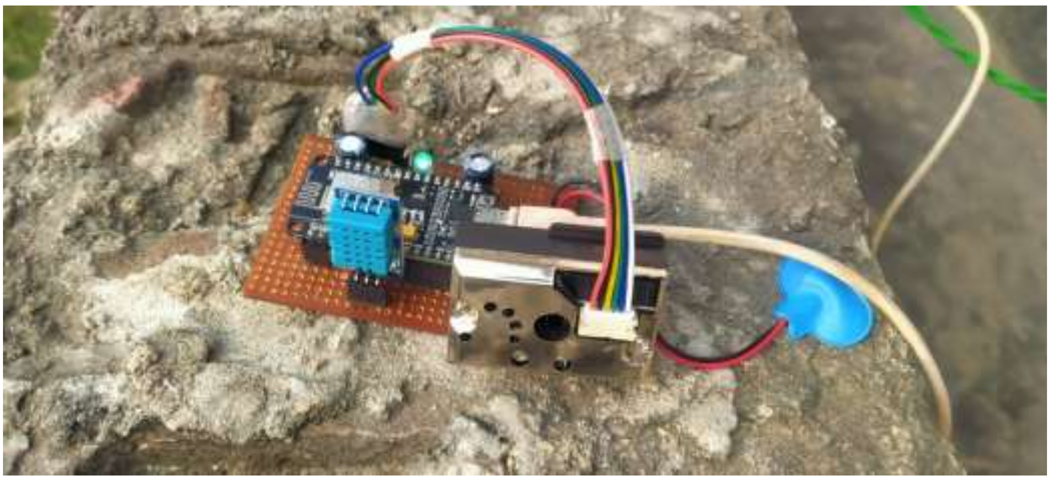


Fig 5.4 Setup for Experiment 2

Conclusion:

We have taken the reference from Accuweather.com for verifying the values. It matched with a +1 error with the temperature data, +5 error with the humidity data and -11 error with the $\mu g/m^3$ data. Hence, we can conclude that the setup has measured the temperature and humidity around the setup area successfully.

EXPERIMENT 3:

Aim: To demonstrate the working of the system in a normal indoor atmosphere

Experimental Condition: The experiment was performed at room temperature.

Observations in ThinkSpeak Cloud:

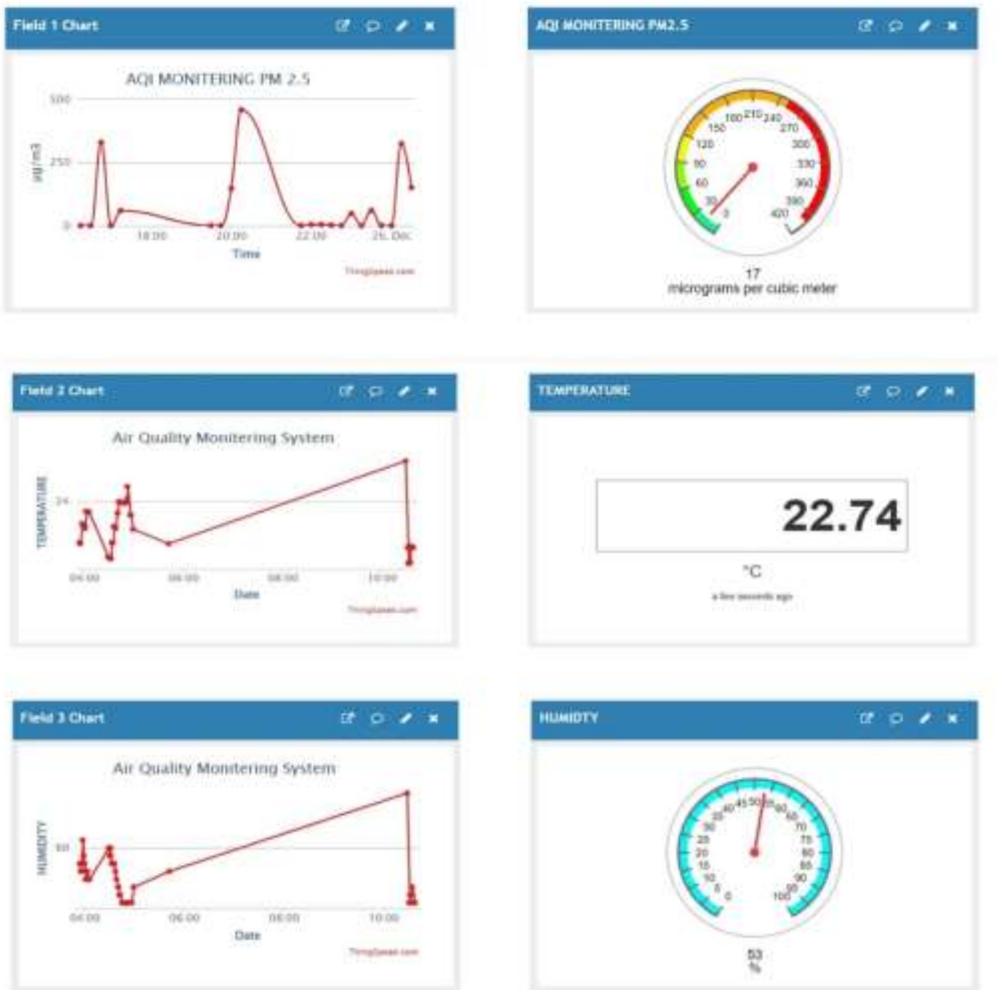


Fig 5.5 Observations for Experimetn 3

Setup:

Fig 5.6 Setup for Experiment 3

Conclusion:

We have taken the reference from Accuweather.com for verifying the values. It matched with a +0.6 error with the temperature data, +2 error with the humidity data and -9 error with the μg/m^3 data. Hence, we can conclude that the setup has measured the temperature and humidity around the setup area successfully.

Table 5.1 Experimental Results

Expt No.	Temperature (in celsius)			Humidity (in %)			Air Quality (in μg/m^3)		
	Project Reading	Accuweather.com Reading	Error	Project Reading	Accuweather.com Reading	Error	Project Reading	Accuweather.com Reading	Error
1	22.74	22	+0.74	53	55	-2	252	245	+7
2	23.51	23	+0.51	45	47	-2	66	71	-5
3	20.6	20	+0.6	49	47	+2	17	26	-9

CHAPTER 6

CONCLUSION

In this project IoT based on measurement and display of Air Quality Index (AQI), Humidity and Temperature of the atmosphere have been performed. From the information obtained from the project, it is possible to calculate Air Quality in μg/m^3. The disadvantage of the GP2Y1010AU0F dust smoke particle sensor module is that it doesn't have any internal or external fan due to which it totally depends upon wind to pass through inlet chamber. If there is no wind or air doesn't itself passes through inlet chamber, the sensor is unable to take accurate reading.

After performing several experiments, it can be easily concluded that the setup is able to measure the air quality in μg/m^3, the temperature in Celsius and humidity in percentage with considerable accuracy. The results obtained from the experiments are verified through Google data. Moreover, the RGB LED indicator help us to detect the air quality level around the setup. However, the project experiences a drawback that is it cannot measure the μg/m^3 values of the pollutant components separately. This could have been improved by adding gas sensors for different pollutants. But eventually, it would increase the cost of the setup and not be a necessary provision to monitor the air quality. Since it's an IOT-based project, it will require a stable internet connection for uploading the data to the ThinkSpeak cloud. Therefore, it is possible to conclude that the designed prototype can be utilized for air quality, humidity and temperature of the surrounding atmosphere successfully.

REFERENCES

[1] https://gaslab.com/blogs/articles/carbon-monoxide-levels

[2] https://www.instructables.com/Measuring-Humidity-Using-Sensor-DHT11

[3] https://pdf1.alldatasheet.com/datasheet-pdf/view/1307647/WINSEN/GP2Y1010AU0F.html

[4] https://components101.com/development-boards/nodemcu-esp8266-pinout-featuresand-datasheet

[5] https://www.arduino.cc

[6] https://thingspeak.com

[7] Pasha, S. (2016). ThingSpeak based sensing and monitoring system for IoT with Matlab Analysis. International Journal of New Technology and Research, 2(6)

[8] IoT based Air Quality monitoring system by Agniva Banerjee B.Tech EE RCC Institute of Information Technology, Kolkata, Bonny Sikdar B.Tech EE RCC Institute of Information Technology, Kolkata, Sourav Roy B.Tech EE RCC Institute of Information Technology, Kolkata, Dr. Debasish Mondal Professor & HOD Department of Electrical Engineering RCC Institute of Information Technology, Kolkata.

[9] Kumar, N. S., Vuayalakshmi, B., Prarthana, R. J., & Shankar, A. (2016, November). IOT based smart garbage alert system using Arduino UNO. In 2016 IEEE Region 10 Conference (TENCON) (pp. 1028-1034). IEEE.

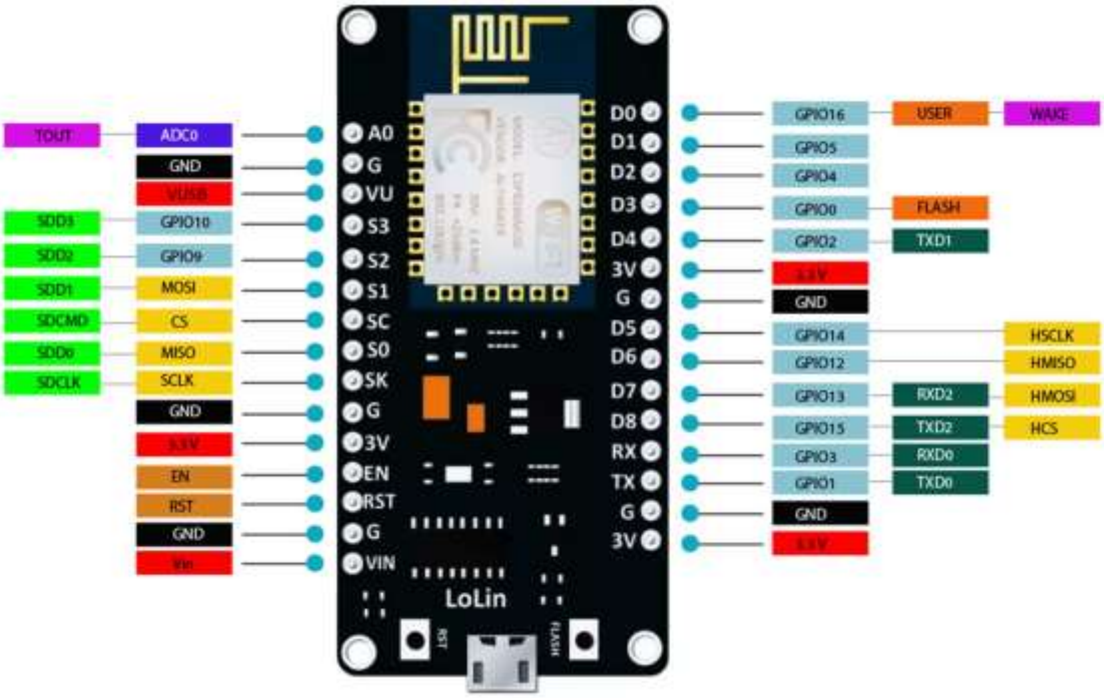
[10] IoT based Air Quality monitoring system using MQ135 & MQ7 with Machine Learning analysis by Kinnera Bharath Kumar Sai M.Tech CSE VIT University, Vellore Subhaditya Mukherjee B.Tech CSE VIT University, Vellore Dr. Parveen Sultana H Associate Professor Department of CSE, VIT University.

[11] https://www.codrey.com/electronic-circuits/how-to-use-GP2Y1010AU0F-dust-smoke-particle-sensor

APPENDIX

A.1 PIN DESCRIPTION OF ESP8266 MCU

Pinout diagram of the NodeMCU:



Description:

Pin Category	Name	Description
Power	Micro-USB, 3.3V, GND, Vin	Micro-USB: NodeMCU can be powered through the USB port 3.3V: Regulated 3.3V can be supplied to this pin to power the board GND: Ground pins Vin: External Power Supply
Control Pins	EN, RST	The pin and the button reset the microcontroller
Analog Pin	A0	Used to measure analog voltage in the range of 0-3.3V

GPIO Pins	GPIO1 to GPIO16	NodeMCU has 16 general purpose input-output pins 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.
12C Pins		NodeMCU has I2C functionality support but due to the internal functionality of these pins, you have to find which pin is I2C.

A.2 DESCRIPTION OF SOFTWARE LIBRARIES USED

ESP8226WiFi Library

The ESP8266WiFi library provides a wide collection of C++ methods (functions) and properties to configure and operate an ESP8266 module.

Commands used are as follows:

- **WiFi.begin(" WiFi Name", "WiFiPassword");** → Command to connect with WiFi network.

- **WiFi.status();** → To check the status of the connection.

  If it returns – WL_CONNECTED → WiFi is connected

  If it returns – WL_IDLE_STATUS → WiFi is connected but no internet found

  If it returns – WL_CONNECT_FAILED → WiFi is not connected

DHT11 sensor Library

The DHT sensor library provides a wide collection of C++ methods (functions) and properties to configure and operate the DHT11 sensor module.

The commands used are as follows:

- **DHT dht(D5, DHT11);** → Set the pin for reading data.
- **dht.begin();**→ Command to connect with DHT11 sensor module.
- **dht.readTemperature();** → Returns the value of the temperature in Celsius.
- **dht.readHumidity();** → Returns the value of humidity in percentage.

ThinkSpeak Library

The ThinkSpeak library provides a wide collection of C++ methods (functions) and properties to configure and operate the ThinkSpeak cloud.

The commands used are as follows:

- **ThingSpeak.writeField(myChannelNumber, 1, t, myWriteAPIKey);** → To upload data in the ThinkSpeak Field

A.3 COST ESTIMATION OF THE PROJECT

For making the project we have used the following components (as mentioned in Table 2). As per the pricing on the online websites for electronic components, we have formulated a cost estimation.

Table 2: Cost Estimation of the Project

Components	**Price (in Rs)**
ESP8266 Microcontroller module	210
DHT11 (Humidity & Temperature Sensor)	95
PM2.5 GP2Y1010AU0F dust smoke particle sensor	450
AMS1117(3.3V voltage regulator IC)	20
RGB LED	11
+9V battery	30
LM7805 (5V voltage regulator IC)	15
220 μF electrolytic capacitor (50V & 25V,3 pieces each)	25
Printed circuit board (PCB)	25
40 Pin 2.54 mm Pitch Male Berg Strip	22
40 Pin 2.54 mm Pitch Female Berg Strip Connector	22
150 & 10K ohm Resistor (5 pieces each)	11
Solder wire tube	150
Total	**1086**