AIR MONITORING SYSTEM USING IOT

MINI PROJECT-I REPORT

Submitted By

RAJESH K S	**(621321205038)**
SIVA HARISH R	**(621321205054)**
KAVIRAJ D	**(621321205303)**

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY

KONGUNADU COLLEGE OF ENGINEERING AND TECHNOLOGY

(AUTONOMOUS)

ANNA UNIVERSITY: CHENNAI 600 025

NOVEMBER 2023

KONGUNADU COLLEGE OF ENGINEERING AND TECHNOLOGY
(AUTONOMOUS)
NAMAKKAL- TRICHY MAIN ROAD, THOTTIAM, TRICHY.

COLLEGE VISION & MISSION STATEMENT

VISION

"To become an Internationally Renowned Institution in Technical Education, Research, and Development by Transforming the Students into Competent Professionals with Leadership Skills and Ethical Values."

MISSION

- ❖ Providing the Best Resources and Infrastructure.
- ❖ Creating Learner-Centric Environment and continuous Learning.
- ❖ Promoting Effective Links with Intellectuals and Industries.
- ❖ Enriching Employability and Entrepreneurial Skills.
- ❖ Adapting to Changes for Sustainable Development.

DEPARTMENT OF INFORMATION TECHNOLOGY

VISION:

To produce competent IT professionals, researchers and entrepreneurs with moral values in the field of Information Technology.

MISSION:

- ❖ Enrich the students' programming and computing skills through the best teaching-learning processes, laboratory practices, and through project-based learning.
- ❖ Inculcate real-world challenges, and emerging technologies and endeavor the students to become entrepreneurs or make them employable.
- ❖ Inculcating moral and ethical values to serve society and focus on students' overall development.

PROGRAM EDUCATIONAL OBJECTIVES (PEOs)

- **PEO I:** Graduates shall become IT professionals with specialization in Software Engineering, Networking, Data Mining, and Cloud computing.
- **PEO II:** Graduates shall build IT solutions through analyzing, designing, and developing software and firmware solutions for real-world problems and social issues.
- **PEO III:** Graduates shall have professional ethics, team spirit, life-long learning, good oral and written communication skills, and adopt the corporate culture, core values, and leadership skills.

PROGRAM SPECIFIC OUTCOMES (PSOs)

- **PSO1: Professional skills:** Students shall understand, analyze and develop IT applications in the field of Data Mining/Analytics, Cloud Computing, Networking, etc., to meet the requirements of industry and society.
- **PSO2: Competency:** Students shall qualify at the State, National, and Intern the on-level competitive examination for employment, higher studies, and research.

PROGRAM OUTCOMES (POs)

Engineering Graduates will be able to:

1. **Engineering knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis** : Identity, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using the first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions** : Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for public health and safety, and cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems**: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage** : Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society**: Apply reason informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and sustainability**: Understand the impact of professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of theengineering practice.
9. **Individual and team work**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **Communication**: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
11. **Project management and finance** : Demonstrate knowledge and understanding of the engineering and management principles and apply these to one’s work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. **Life-long learning**: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change

KONGUNADU COLLEGE OF ENGINEERING AND TECHNOLOGY
(AUTONOMOUS)
NAMAKKAL- TRICHY MAIN ROAD, THOTTIAM, TRICHY.

BONAFIDE CERTIFICATE

Certified that this project report titled "**AIR MONITORING SYSTEM USING IOT**" is the bonafide work of " **RAJESH K S (621321205038), SIVA HARISH R (621321205054)** and **KAVIRAJ D (621321205303)** " who carried out the project under my supervision.

SIGNATURE	**SIGNATURE**
HEAD OF THE DEPARTMENT	**SUPERVISOR**
Mr. N. PREMKUMAR, M.E., (Ph.D.),	Mr. J. SATHISHKUMAR ,M.Tech., (Ph.D)
Associate Professor,	Assistent Professor,
Department of Information Technology,	Department of Information Technology,
Kongunadu College of Engineering and	Kongunadu College of Engineering and
Technology, Thottiam, Trichy- 621215	Technology, Thottiam, Trichy- 621215

Submitted for the Mini project -I Viva-Voce Examination held on.............

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

We wish to express our sincere gratitude to our beloved chairman **Dr.PSK.R.PERIASWAMY,** for providing the facilitation to make us extensively internationalized to meet global competence with almost perfection.

We would like to express our sincere thanks to our Principal **Dr.R.ASOKAN, M.S.,M.Tech., Ph.D.,** for the facilities and the encouragement given to the progression and completion of this project.

We proudly render our immense gratitude to **Mr.N.PREMKUMAR, M.E.,(Ph.D).,** Associate Professor and Head of the Department, Department of Information Technology for his effective leadership, encouragement and supportive guidance to this project.

We proudly render our thanks to our Mini project-I Co-ordinator **Mr. J.SATHISHKUMAR ,M.Tech., (Ph.D) ,** Professor, Department of Information Technology for his valuable ideas, encouragement and supportive guidance throughout this project.

We wish to extend our heartfelt regard and sincere thanks to our Mini project -I guide **Mr.J.SATHISHKUMAR ,M.Tech., (Ph.D) ,** Professor, Department of Information Technology for his Motivation, Continuous encouragement and expert guidance throughout this project.

We wish to extend our sincere thanks to all faculty members of Information Technology Department for their valuable suggestions, kind cooperation and encouragement on successful completion of this project.

ABSTRACT

Air pollution is one of the biggest threat to present day environment everyone is affected. Air pollution affects day by day including humans, animals, crops, cities, forest an aquatic ecosystems. Besides that it should be controlled certain level to prevent the increasing rate of global warming. This project aims to design an IoT-based air pollution monitoring system. In the system not only Monitoring the air pollution andindicate the alert signal depends on the level of signal. The system provides the remedies to the current situation of the air pollution. The monitoring results can be displayed in a mobile app with integrated real- time data transmitted by connecting to Wi-Fi network and GSM through ESP8266 The Internet of Things (IoT) based in the air quality monitoring system plays an important part. Accordingly, the users can receive updates from the mobile apps at anytime and anywhere.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ABBREVIATIONS	TITLE
DHT	**D**igital **H**umidity and **T**emperature
IoT	**I**nternet **o**f **T**hings
PPM	**P**arts **P**er **M**olecule
PM	**P**articulate **M**atter
CO	**C**arbon Monoxide
CO2	**C**arbon Dioxide
LED	**L**ight **E**mitting **D**iode
LPG	**L**iquid **P**etroleum **G**as
IDE	**I**ntegrated **D**evelopment **E**nvironment

CHAPTER 1
INTRODUCTION

An air monitoring system using IOT (Internet of Things) is a sophisticated and interconnected solution designed to continuously monitor, measure, and analyze various air quality parameters in real-time. This system leverages IOT technology to collect, transmit, and process data from sensors placed in the environment. These sensors can detect a wide range of air quality indicators, such as pollutants, temperature, humidity, and more. Here's an introduction to such a system:

The quality of the air we breathe is a critical factor in determining public health and the overall well-being of our planet. Poor air quality can lead to a variety of health issues, including respiratory diseases, and can negatively impact the environment. To address these challenges, we have developed an innovative solution that harnesses the power of the Internet of Things (IOT) to create a comprehensive air monitoring system

Our IOT-based air monitoring system is state-of-the-art and provides accurate, dependable, and real-time data on air quality. Its purpose is to gather information on different air contaminants, weather patterns, and other environmental factors on an ongoing basis. The system provides a comprehensive view of air quality across regions by utilizing an integrated network of IOT sensors that are strategically located in urban, industrial, and rural locations.

1.1 Aim of the 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 paper 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 web page so that we can monitor it very easily.

1.2 Literature Survey

The explanation of the Air Quality Index (AQI) and its standard ranges are described in [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 . The MQ-135 gas sensor is used to measure the air quality of the surroundings . It can be calibrated with respect to fresh air, alcohol, carbon dioxide, P a g e hydrogen and methane. In this project, it has been calibrated with respect to fresh air

In 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 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 . As this project uses two sensors, both of them have 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 power supply for the sensors as NodeMCU alone is not sufficient to drive two sensors

CHAPTER 2

SYSTEM ANALYSIS

2.1 EXISTING SYSTEM

The existing system of air monitoring system is help to identify the air Quality only. In the system there is no air level to be fixed in the system. These systems utilize IoT devices to collect, transmit, and analyze air quality data in real-time. These systems encompass a range of components, commencing with a network of sensors designed to measure crucial air quality indicators. These sensors are connected to micro-controllers or microprocessors, such as Arduino or Raspberry Pi, which process sensor data and manage communication with cloud-based or local servers. Communication modules, including Wi-Fi, cellular, Lo Ra, and Sig fox, transmit the collected data to a cloud platform, where data is stored and analyzed.

DISADVANTAGES

- Limited Accuracy
- Limited Range
- Limited Compatibility
- Data Interpretation
- Limited Ability

2.2 PROPOSED SYSTEM

The Designing a proposed air monitoring system using IoT involves integrating various components to create an effective and efficient solution. The System Architecture contains sensor, Communication module, micro-controller,cloud platform, data processing and analysis and user interface. These IoT devices are equipped with communication modules, enabling them to send real-time data to a central server or cloud platform. User accessibility and engagement are paramount in the proposed system. An intuitive and user-friendly interface, often in the form of a web-based or mobile application, grants users the ability to monitor air quality in their surroundings. Moreover, the system promotes data transparency, with options for open data access through Application Programming Interfaces (APIs).

ADVANTAGES

- Alerts and Notifications
- User Engagement
- Remote Management
- Flexibility

CHAPTER 3

THEORY & DESCRIPTION OF THE COMPONENTS

3.1 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.

Another example of integrating the IoT is Living Lab which integrates and combines research and innovation processes, establishing a public-private-people-partner ship. There are currently 320 Living Labs that use the IoT to collaborate and share knowledge between stakeholders to co-create innovative products.

3.2 Components Description

❖ NodeMCU V3

NodeMCU V3 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 Esp8266 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 pin-out Diagram of NodeMC3 is shown in Fig. 3.1.

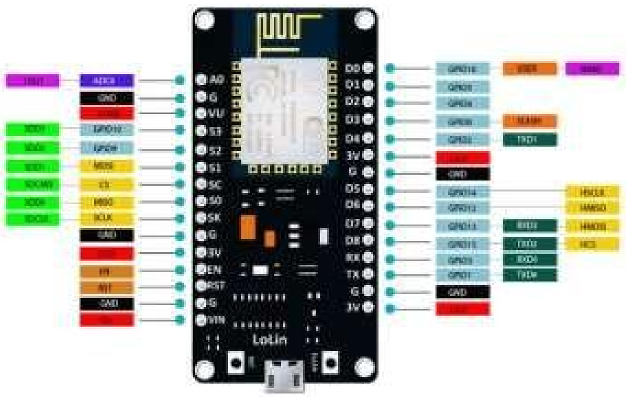


Fig. 3.1 Pin-diagram for NodeMCU

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 temperature to measure the surrounding air

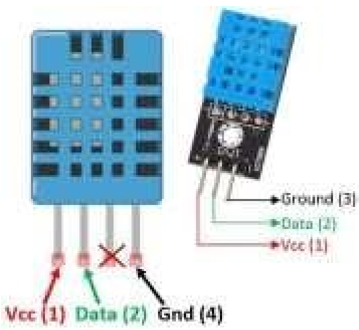


Fig 3.2 (Pin-out Diagram of DHT11sensor)

Humidity Measurement: The humidity sensing capacitor has two electrodes with a moisture-holding substrate as a dielectric between them as shown in Fig 3.3. 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.

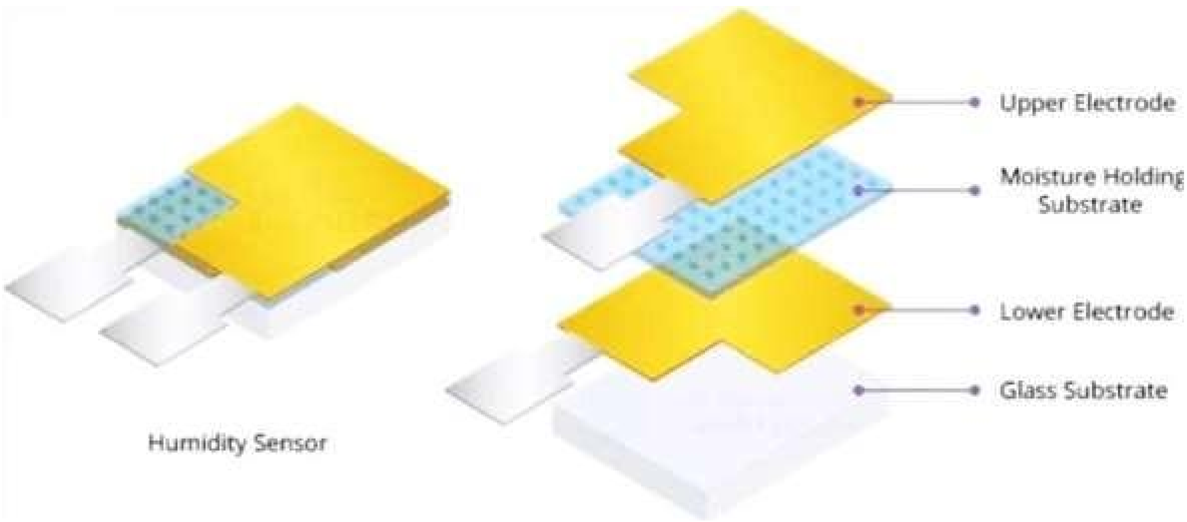


Fig 3.3(The structure of the humidity sensor)

MQ-135 Gas Sensor Module:

The material of MQ135 is SnO2, it is a special material: when exposed to clean air, it is hardly being conducted, however, when put in an environment with combustible gas, it has a pretty performance of conductivity. Just make a simple electronic circuit, and convert the change of conductivity to a corresponding output signal. MQ135 gas sensor is sensitive to Ammonia, Sulphide, Benzene steam, smoke and other harmful gases. Used for family, surrounding environment noxious gas detection device, apply to ammonia, aromatics, sulphur, benzene vapor, and other harmful gases/smoke, gas detection, tested concentration range: 10 to 1000ppm. In a normal environment, the environment which doesn't have detected gas set the sensor's output voltage as the reference voltage, the analog output voltage will be about 1V, when the sensor detects gas, harmful gas concentration increases by 20ppm per voltage increase by 0.1V.

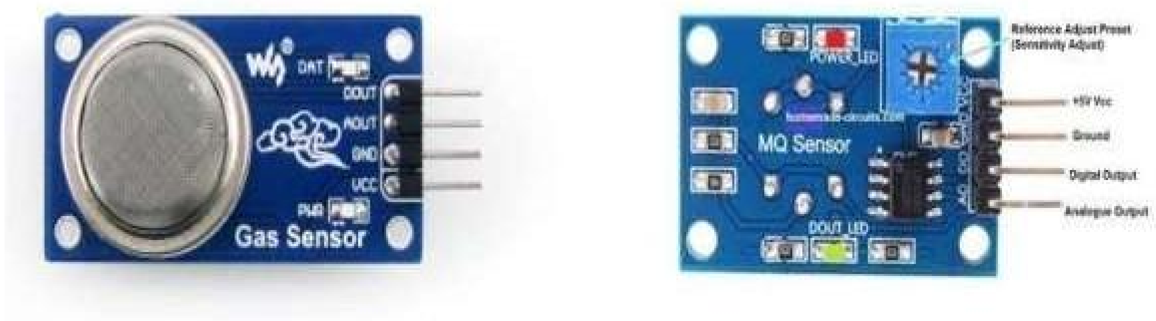


Fig 3.4 (MQ-135 Gas Sensor Module)

Arduino IDE:

The Arduino IDE is open-source software, which is used to write and upload code to the Arduino boards. The IDE application i s 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 arduino and Arduino board with the IDE to upload the sketch written in the Arduino IDE software. The sketch is saved with the extension '.ino.'

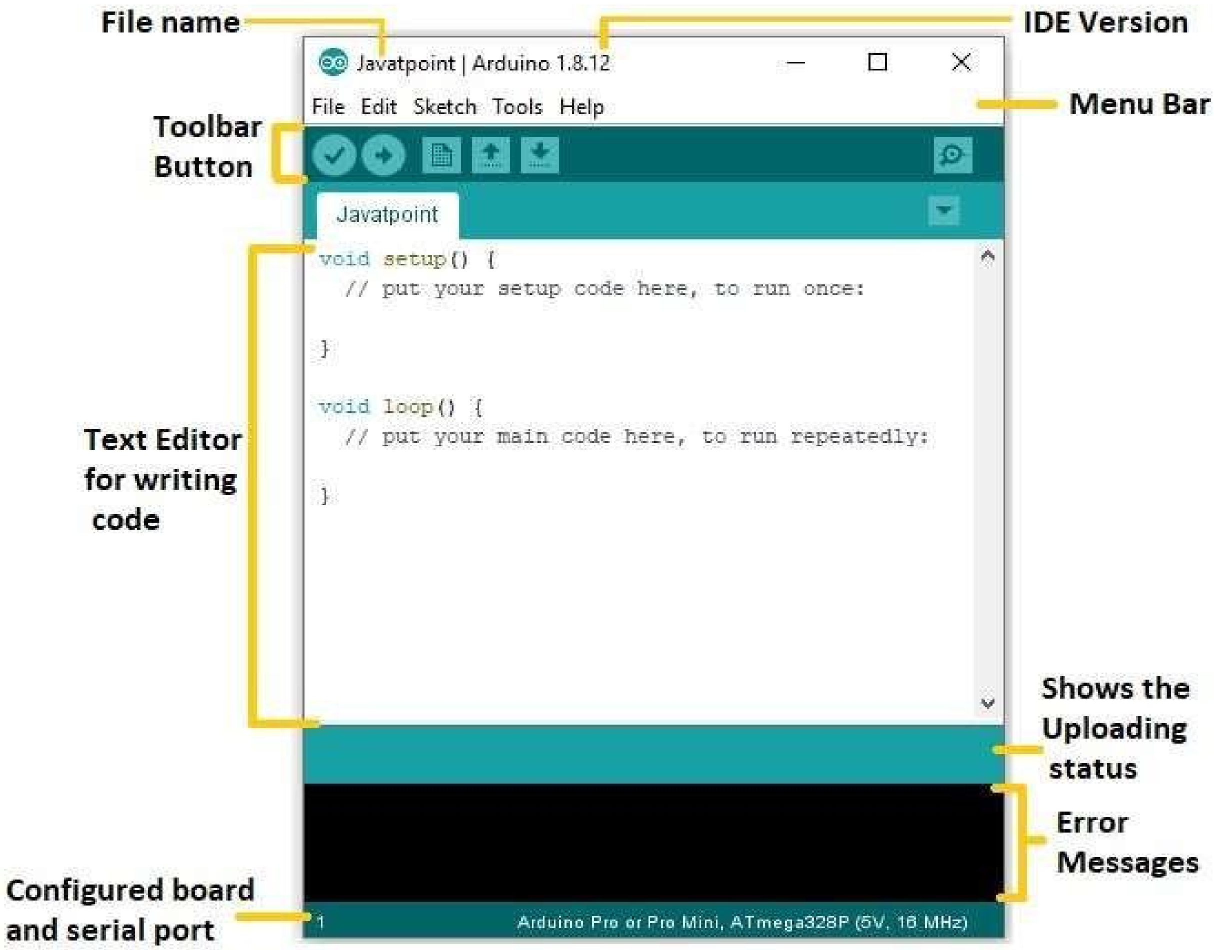


Fig 3.5Arduino IDE

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 3.6 Working process

3.3 Working Procedures

NodeMCU 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 MQ-135 gas sensor module, air quality is measured in ppm. 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 MQ-135 gas sensor module is done.The sensor is set to preheat for 24 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 10 minutes

STEP 3: The result of calibration found in STEP 1 is used to configure thefinal 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 4

HARDWARE MODEL

4.1 Hardware Model to Preheat DHT11 Sensor Module

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 Node MCU

STEP 2 : The Gnd pin of the DHT11 sensor module wasconnected with the Gnd pin of Node MCU.

STEP 3 : The Node MCU is powered with a 12V DC via AC-DC adapter for 20 minutes.

STEP 4 : The setup was then disconnected.

Fig. 4.1 shown below describes the fore said connections.

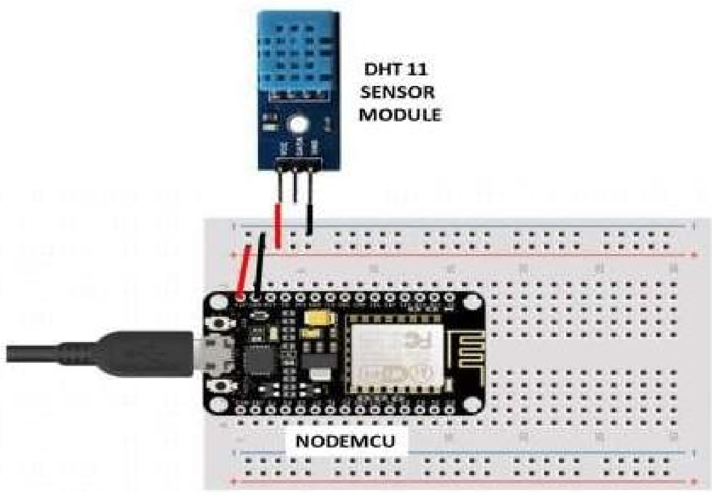


Fig. 4.1 dht connection

4.2 Hardware Model to Preheat and Calibrate MQ-135 GasSensor Module

The following steps were performed to preheat the MQ-135 gas sensor module

STEP 1: The Vcc pin of the MQ-135 gas sensor module was connected with the VU pin of NodeMCU.

STEP 2: The Gnd pin of the MQ-135 gas sensor module was connected with the Gnd pin of NodeMCU.

STEP 3: The NodeMCU is powered with a 12V DC via AC-DCadapter for a day.

STEP 4: The setup was then disconnected.

Fig. 4.2 shown below describes the foresaid connections.

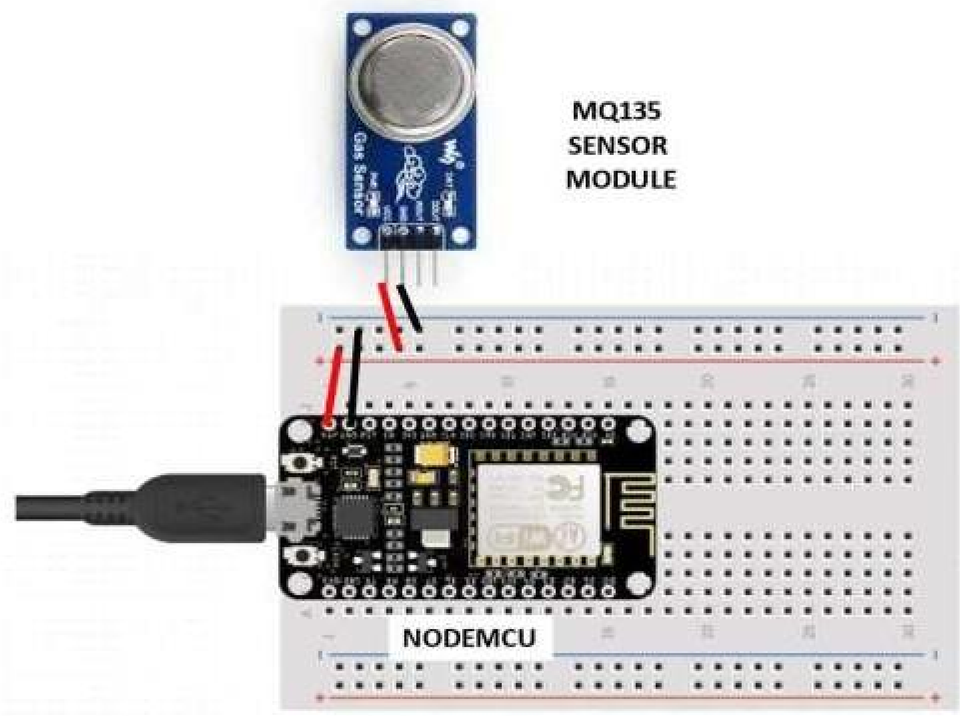


Fig. 4.2(Circuit Diagram to Preheat the MQ-135 Gas sensor module)

The following steps were performed to calibrate the MQ-135 gas sensor module

STEP 1 : The Vcc pin of the MQ-135 gas sensor module was connectedwith the VU pin of NodeMCU

STEP 2 : The Gnd pin of the MQ-135 gas sensor module was connectedwith the Gnd pin of NodeMCU.

STEP 3 : The analog DATA pin of the MQ-135 gas sensor module was connected with the A0 Pin of the NodeMCU.

STEP 4 : The software code to calibrate the sensor is then uploaded to the NodeMCU and the value of R0in fresh air is collected from the serial monitor of the Arduino IDE.

STEP 5 : The setup was then disconnected.

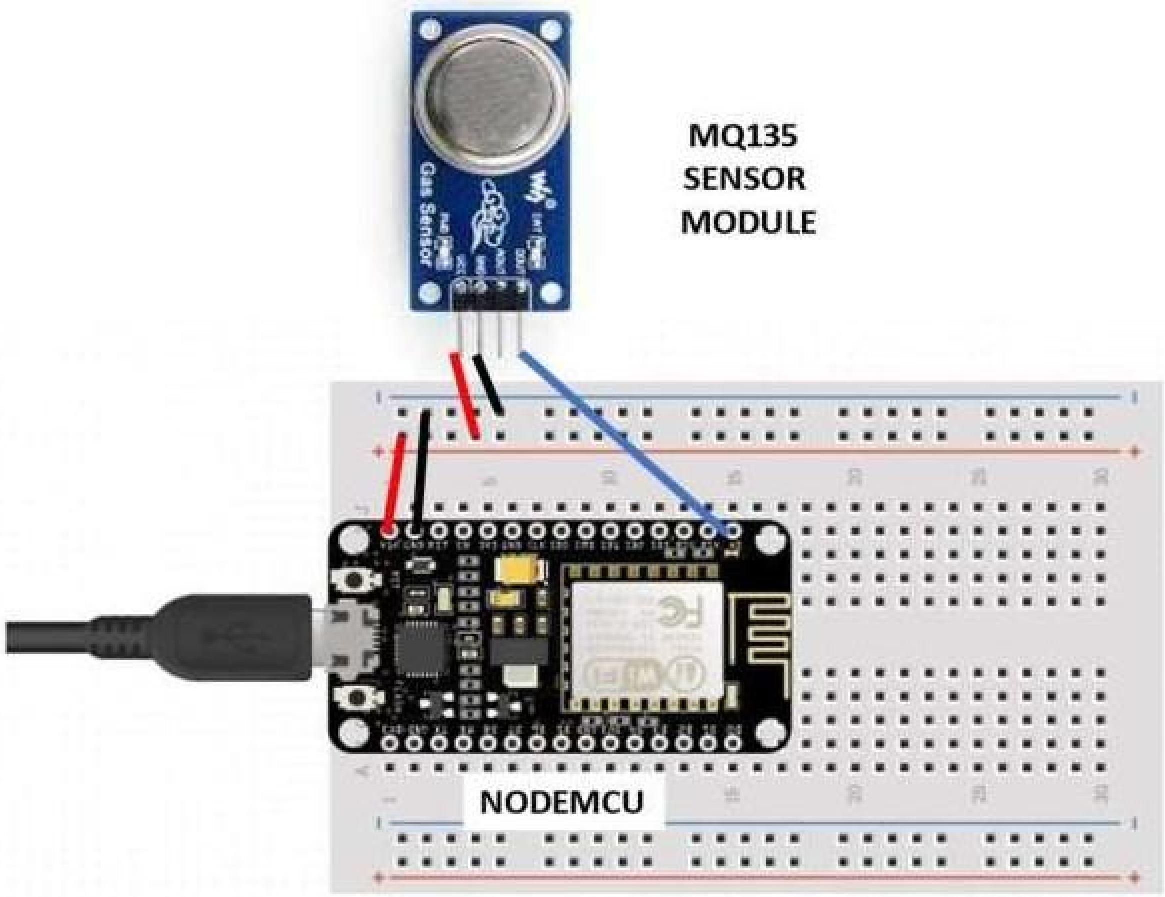
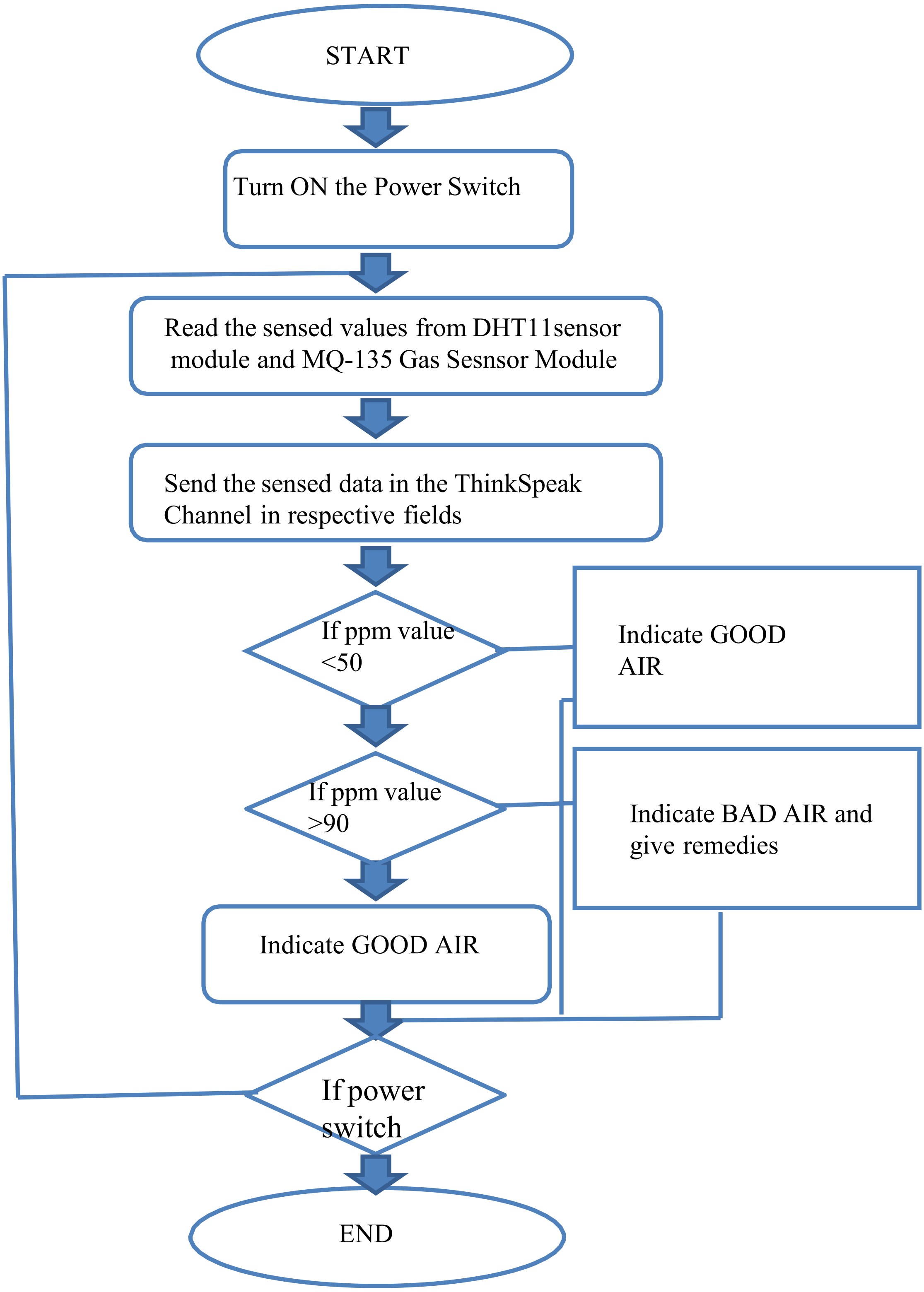


Fig. 4.3(Circuit Diagram to Calibrate the MQ-135 Gas sensor module)

CHAPTER 5

SOFTWARE IMPLEMENTATION

5.1 Working Algorithm



CHAPTER 6

SYSTEM SPECIFICATION

6.1 Software Requirements:

- Arduino IDE
- Thinkspeak Server

6.2 Hardware Requirements:

- ESP8266
- MQ-135
- DHT11
- Jumber wires
- Bread board

CHAPTER 7

RESULTS

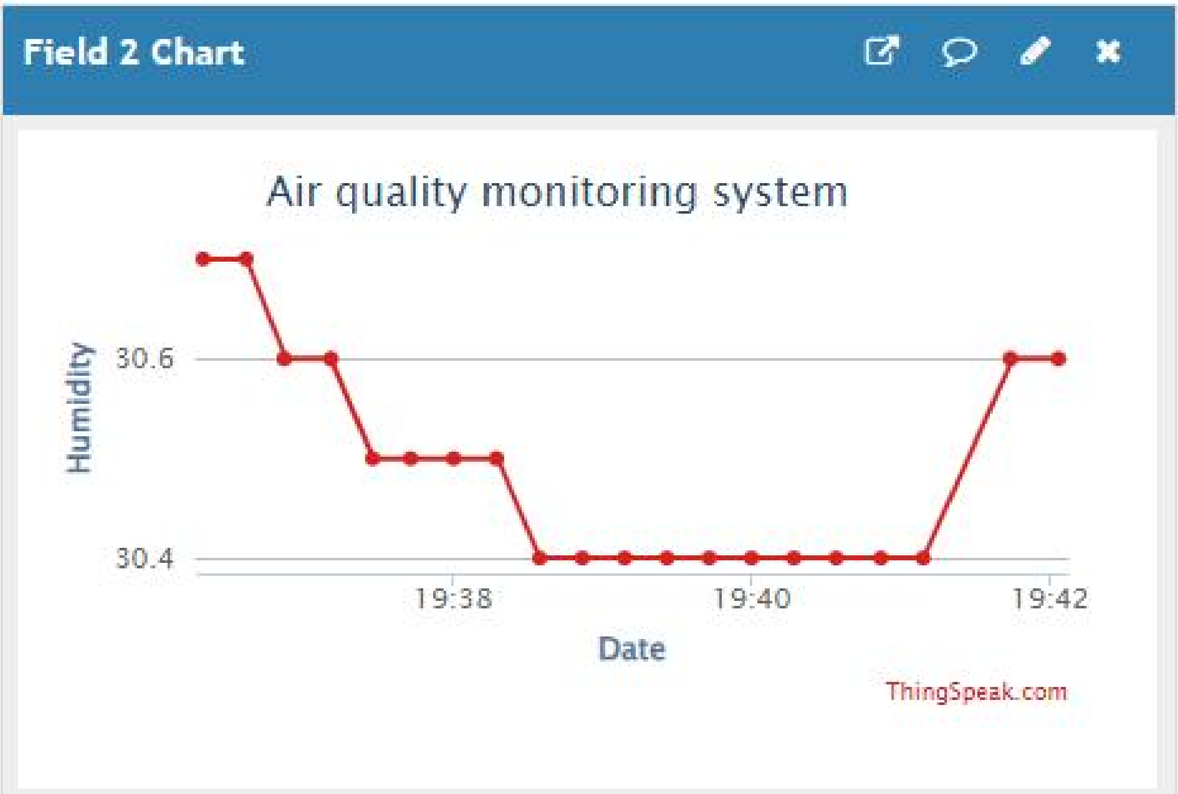
The working of the designed prototype has been investigated for the 5 sets of experiments as described in the following section.

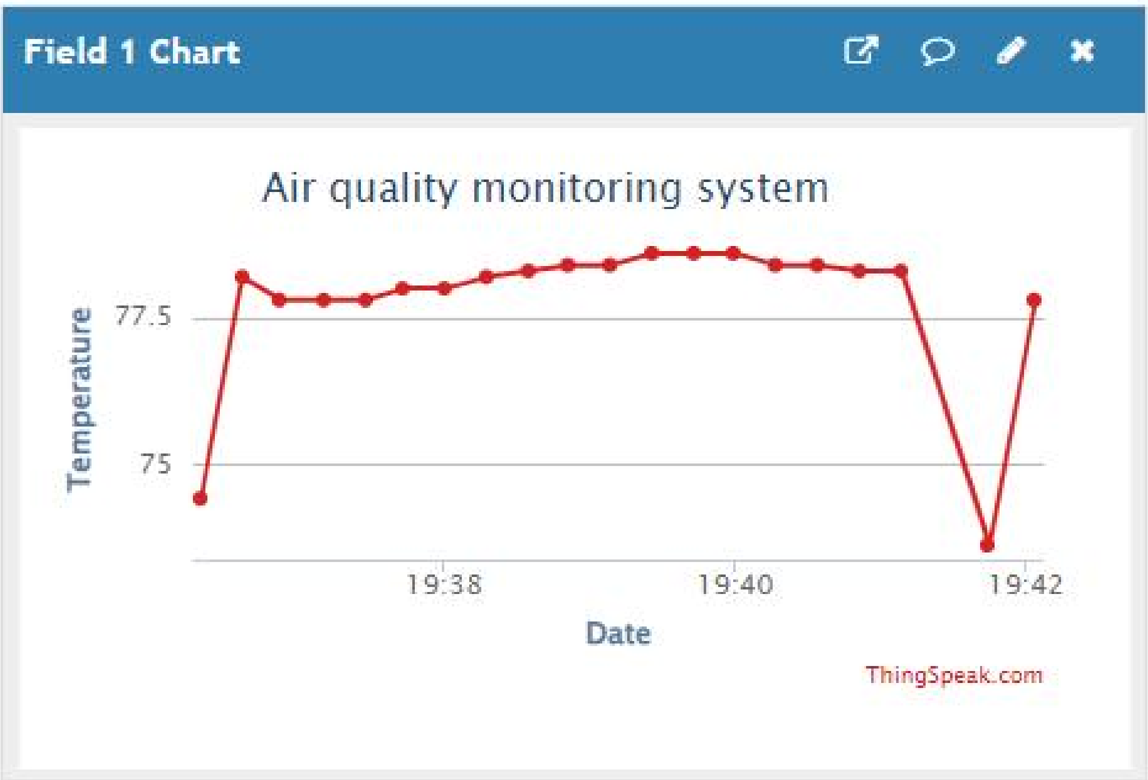
7.1 EXPERIMENT 1:

Aim: To demonstrate the working of the system in a warm and humid outdoor atmosphere.

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

Observations in ThingSpeak Cloud:





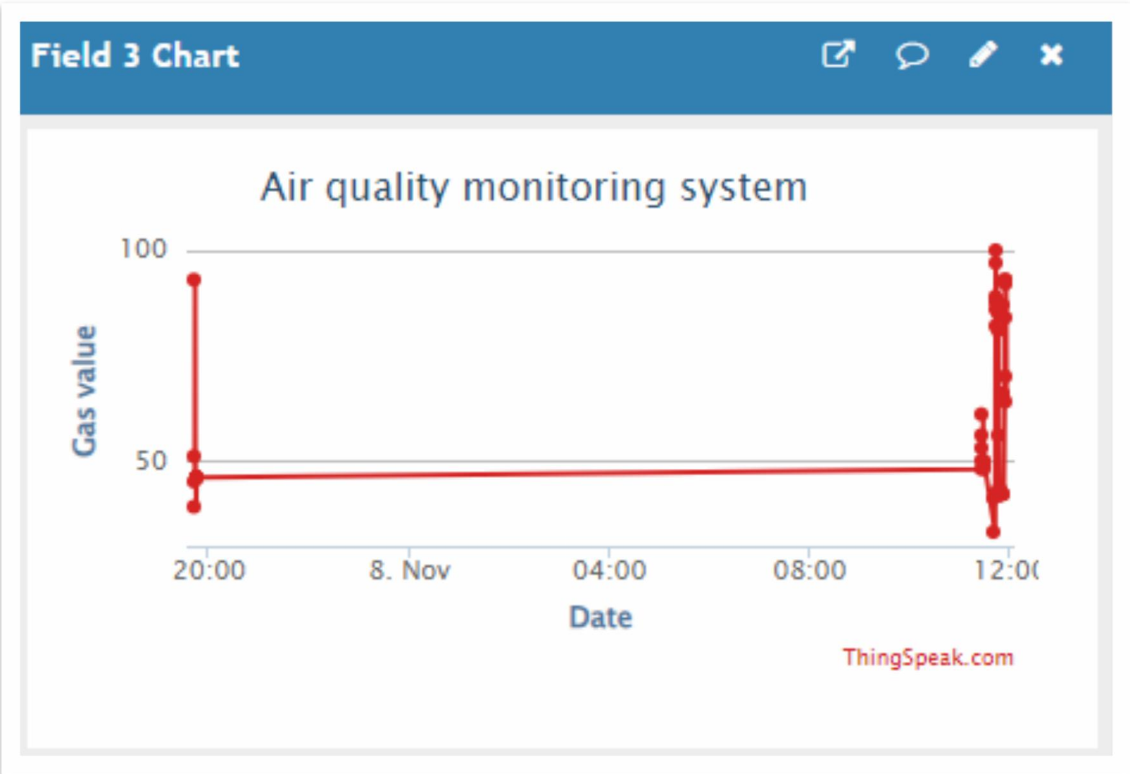


Fig: 7.1 Observations for Experiment 1

Setup:

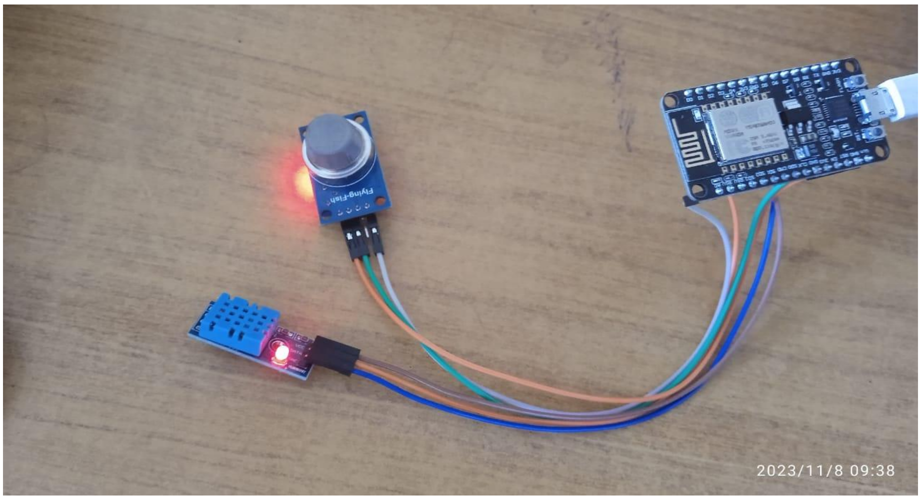


Fig: 7.2 Setup for Experiment 1

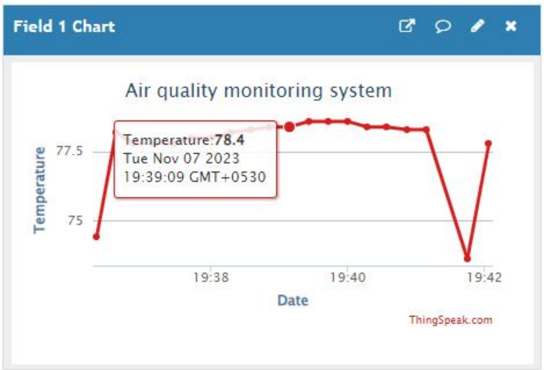
Conclusion: We have taken the reference from the Samsung mobile weather app for verifying the values. It matched with a +1.20 error with the temperature data, +5 error with the humidity data and +0.11 error with the PPM data. Hence, we can conclude that the setup has measured the temperature and humidity around the setup area successfully.

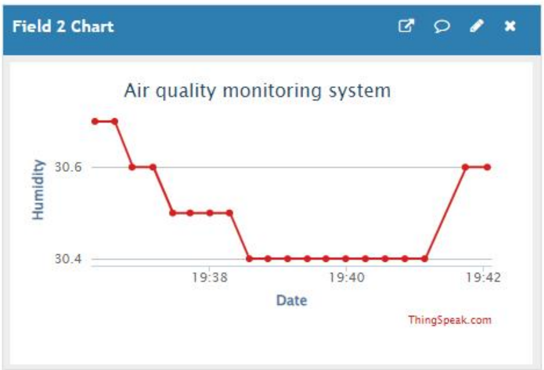
7.2 EXPERIMENT 2:

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

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

Observations in ThinkSpeak Cloud:





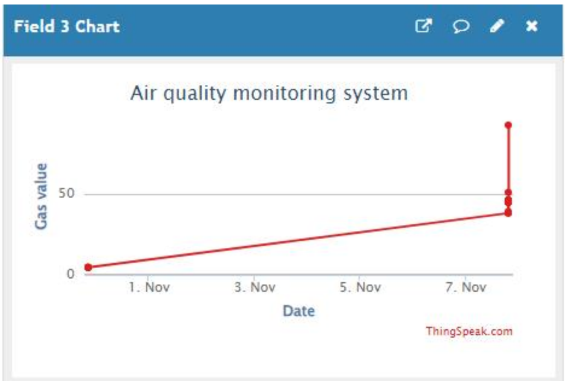


Fig: 7.3 Observations for Experiment 2

Setup:

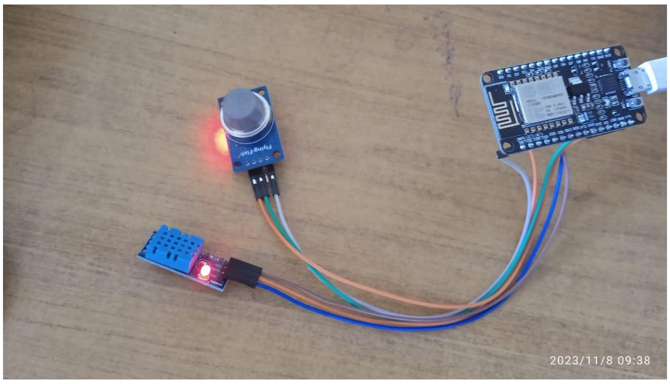


Fig: 7.4 Setup for Experiment 2

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 Samsung mobile weather app for verifying the values. It matched with a +1.80 error with the temperature data, +4 error with the humidity data and -0.7 error with the PPM data. Hence, it can be concluded that we can detect the presence of smoke with the help of this monitoring system.

CHAPTER 8

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 PPM. The disadvantage of the MQ135 sensor is that specifically it can’t tell the Carbon Monoxide or Carbon Dioxide level in the atmosphere, but the advantage of MQ135 is that it is able to detect smoke, CO, CO2, NH4, etc harmful gases

After performing several experiments, it can be easily concluded that the setup is able to measure the air quality in ppm, the temperature in Celsius and humidity in percentage with considerable accuracy. The results obtained from the experiments are verified through Google data. Moreover, the led indicators help us to detect the air quality level around the setup. However, the project experiences a drawback that is it cannot measure the ppm 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

CHAPTER 9

APPENDICES

9.1 Pin description of NodeMCU

Pin-out diagram of the NodeMCU:

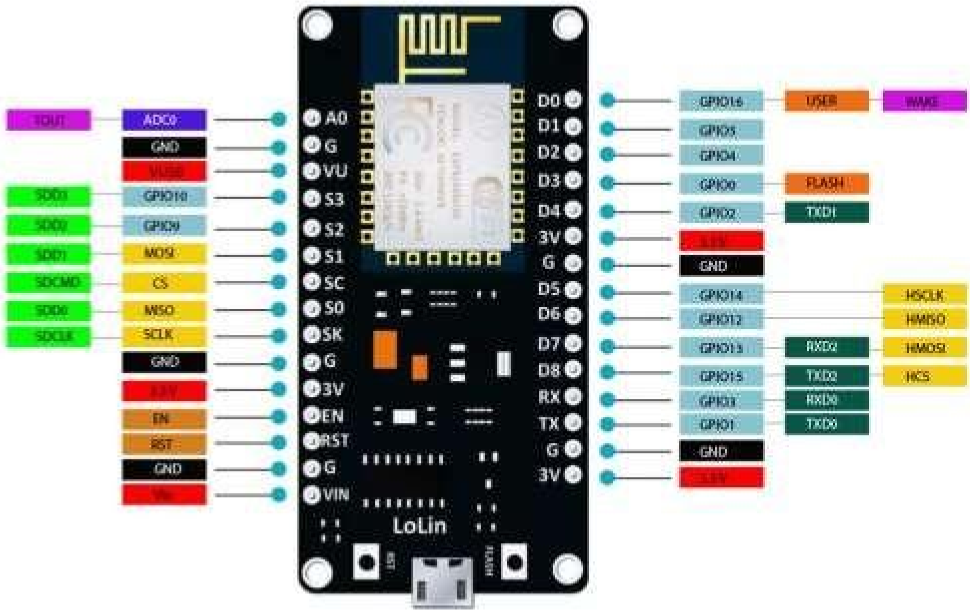


Fig. 9.1(pin-out diagram)

Table 1: Description of Pin-Diagram

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.
I2C Pins		NodeMCU has I2C functionality support but due to the internal functionality of these pins, you have to find which pin is I2C.

Table 2. Description of Pin-diagram

9.2 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 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, my Write API Key);

→ To upload data in the ThinkSpeak Field.

9.3 Sample Code:

Code for MQ-135:

```
#include <ESP8266WiFi.h>

 String apiKey = "SL8AS9C2DOY2TTXS";
const char *ssid = "Redmi 9";
const char *pass = "9626314249";
const char *server = "api.thingspeak.com";
WiFiClient client;

void setup() {
 Serial.begin(115200);
 delay(1000);
 Serial.println("Connecting to ");
 Serial.println(ssid);
 WiFi.begin(ssid, pass);

while (WiFi.status() != WL_CONNECTED) {
   delay(5000);
   Serial.print(".");
 }

 Serial.println("");
 Serial.println("WiFi connected");
}

void loop() {
float h = analogRead(A0);

if (isnan(h)) {
  Serial.println("Failed to read from MQ-135 sensor!");
```

```
  return;
}

if (client.connect(server, 80)) {
  String postStr = apiKey;
  postStr += "&field3=";
  postStr += String(int(h));
  postStr += "\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-Length: ");
  client.print(postStr.length());
  client.print("\n\n");
  client.print(postStr);

  Serial.print("Gas Level: ");
  Serial.println(int(h));
  Serial.println("Data Sent to ThingSpeak");
}

delay(500);
client.stop();
Serial.println("Waiting...");
delay(15000);
}
```

Code for DHT11:

```
#include <DHT.h>
#include <ESP8266WiFi.h>

String api_key = "SL8AS9C2DOY2TTXS";
const char *ssid =  "Redmi 9";
const char *pass =  "9626314249";
const char* server = "api.thingspeak.com";

#define DHTPIN 0
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);

WiFiClient client;

void setup() {
 Serial.begin(115200);
 delay(2000);
 pinMode(2, OUTPUT);
 digitalWrite(2, 0);
 Serial.println("Connecting to ");
 Serial.println(ssid);

 WiFi.begin(ssid, pass);

 while (WiFi.status() != WL_CONNECTED) {
   delay(500);
   Serial.print(".");
 }
 Serial.println("");
 Serial.println("WiFi connected");
 dht.begin();
}
void loop() {
 int chk = dht.read(DHTPIN);
 float h = dht.readHumidity();
 float t = dht.readTemperature();
```

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

  if (client.connect(server, 80)) {
    String data_to_send = api_key;
    data_to_send += "&field1=";
    data_to_send += String(h);
    data_to_send += "&field2=";
    data_to_send += String(t);
    data_to_send += "\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: " + api_key + "\n");
    client.print("Content-Type: application/x-www-form-urlencoded\n");
    client.print("Content-Length: ");
    client.print(data_to_send.length());
    client.print("\n\n");
    client.print(data_to_send);
    delay(1000);
    Serial.print("Temperature: ");
    Serial.print(t);
    Serial.print(" degrees Celsius, Humidity: ");
    Serial.print(h);
    Serial.println("%. Sent to ThingSpeak.");
  }
  client.stop();
  Serial.println("Waiting...");


  delay(15000);
}
```

9.3 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)
NodeMCU V3	415
DHT11 Sensor Module	140
MQ135 Gas Sensor Module	275
Connecting Wires	200
Breadboard	92
Total	1122

REFERENCES

[1] https://components101.com/development-boards/nodemcu-esp8266 pinout-features_x0002_and-datasheet

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

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

[4] 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.

[5] https://pdf1.alldatasheet.com/datasheetpdf/view/1307647/WINSEN/MQ135.html

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

[7] https://thingspeak.com

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

[9] https://www.codrey.com/electronic-circuits/how-to-use-mq-135-gas-sensor

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