International Journal of Engineering Applied Sciences and Technology, 2022
Vol. 7, Issue 8, ISSN No. 2455-2143, Pages 151-159
Published Online December 2022 in IJEAST (http://www.ijeast.com)

IMPLEMENTATION OF IOT FOR AIR QUALITY SURVEILLANCE

Debasish Mondal, Agniva Banerjee
Dept. of Electrical Engineering
RCC Institute of Information Technology
Kolkata, West Bengal, India

Abstract—**This paper aims to implement Internet of Thing (IoT) for surveillance of air quality of the surroundings and environment using a computer or mobile from anywhere. A low cost open source IoT platform, Node MCU has been utilized for the main controlling role. It has been configured in a manner, such that, it receives the measuring signals from various sensors and displays monitored data through Thing Speak cloud in any connected PC or mobile phone. Besides numeric display, the quality levels of the air are also indicated via different colors of LED indicators. The proposed system not only conveys the AQI information in ppm but also monitor the temperature and humidity successfully in different environmental conditions.**

Keywords—**Air Quality Index (AQI), Arduino IDE**, **Internet of Thing (IoT), NodeMCU, Thing Speak Cloud,**

I. INTRODUCTION

The problem of air pollution and maintaining the quality of air is one of the crucial challenges to the present-day environment. The impact of greenhouse gases makes the situation more worsening day by day. Everything is being affected by the pollution including humans, animals, crops, cities, forests and also aquatic co systems. To prevent the increasing level of global warming, air pollution should be controlle data certain level of ppm. There are various methods and attempts have been made by the researchers to measure and monitor the air quality in different operating conditions [1]-[2]. A WSN based air excellence monitoring system for industrial and urban areas has been introduced in [3], where individual gas sensors are used for measurement of various air quality metrics and all the values are sent to the mobile via IOT central server. A real-time air quality monitoring system based on the wireless communication technology has been developed in [4]. The proposed system is cable to monitor a wide area comprising of large number sensor network. Many researchers aimed to implement wireless network for the monitoring of environmental pollution. In [5]-[6] wireless sensor network has been enforced for sensing and monitoring CO_2 emission generated through vehicular and transportation system. Besides that an air pollution monitoring system based on Geo sensor Network has been reported in [7], where the status of air pollution on the remote place has been monitored through flexible data acquisition policy with alarm and safety guidelines.

However, IoT based air quality monitoring system not only enable us to monitor the air quality but also possible to send instant alert when quality of air deteriorates in a region or goes down beyond a certain predefined level. In this context an IoT based air pollution monitoring system has been reported in [8], where MQ2 and MQ7 type of pollution sensor has been utilized to detect and monitor air quality through the mobile and wireless devices.

Several efforts have been made by the researchers to implement IoT for this purpose for many years. In [9] IoT based omnipresent sensing and monitoring system has been implemented for domestic as well as non-domestic environments applying Xively platform. An IoT based low cost ubiquitous smart sensing system has been reported in [10] for measurement and transmission of regular domestic conditions. In [11] a Thing speak based air quality sensing and monitoring system using IoT has been presented. Here, MATLAB also utilized to display the status of the variables graphically. A new technique has been proposed in [12], where IoT with machine learning approach has been implemented for prevention and monitoring vehicle emitting air pollution.

In this work a real-time IoT based standalone air quality monitoring system has been designed. The Node MCU has been programmed to sense the measuring signals in an area which display the real-time data in the mobile phone or in computer through the Thing Speak cloud. This real time data has been utilized further for analyzing the air quality in PPM in that area. The quality level of the air has also been indicated via three different color led indicators.

The remaining part of the paper is organized as follows. The schematic layout and functions of the components of the proposed prototype are demonstrated in Section II. Section III describes the integration of different hardware components in the prototype and their preheat methods. The method of calibration of gas sensor is illustrated in Section IV. The algorithm of the implemented software has been presented in Section V. The proposed model has been validated in Section VI through multiple set of experimental results in different environment conditions.

International Journal of Engineering Applied Sciences and Technology, 2022
Vol. 7, Issue 8, ISSN No. 2455-2143, Pages 151-159
Published Online December 2022 in IJEAST (http://www.ijeast.com)

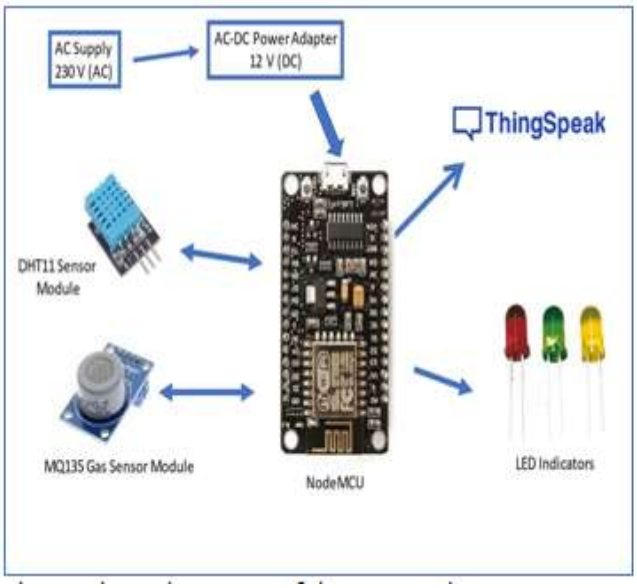

Fig. 1 Schematic Layout of the Proposed System

II.COMPONENTLAYOUT OF THE PROPOSED SYSTEM

The schematic layout of the proposed system is depicted in the Fig. 1. The power supply of the Node MCU is provided from a 12V DC adopter through main. Two primary air quality sensors DHT11 and MQ-135 are incorporated with the Node MCU for the measurement of Humidity& Temperature and the gases like ammonia, sulphide, benzene, smoke and other harmful gases. The description and function of the hardware and software components used in this work are described as follows;

A. NodeMCU (ver.3)

The NodeMCU is a popular open-source firmware, based on CPU ESP8266 development kit [13]. It can communicate with the IOT based sensors and objects and perform data transfer using Wi-Fi protocol. In addition to that this microcontroller has some important features like, GPIO, PWM and ADC etc. which can be utilized solely to solve many real time project works without additional hardware. The pin diagram of a NodeMCU ver. 3 has been shown in the Fig. 2.It has 13 nos (D0-D12) General Purpose Input/output (GPIO) interfacing pin. The pin D0 (GPIO16) does not support functions like open-drain, interrupt, PWM or I²C. It is only used for the data read and write. The advantage of Node MCU is that it can be simply coded in Arduino IDE. It has a very low current consumption between15µA to 400mA.

B. DHT11SensorModule

The DHT11 sensor module [14] is a combination of temperature and humidity sensor that produces digital output in terms of voltage signal. The sensor is a combination of a capacitive type humidity sensor and a NTC type temperature sensor, Thermistor. The humidity sensing capacitor has moisture-holding substrate as a dielectric. Any changes in humidity levels in air changes the capacitance value. The Thermistor is made of by semi conductor ceramics or polymers materials to get wide range of resistance values. The value of the resistance decreases with an increase in temperature. The pin diagram of a DHT11 IC is presented in the Fig. 3. A supply voltage of 5VDC is given to the VCC pin 1 and GND pin 3 used for ground. The Data pin 2 read sensor output in terms of voltage.

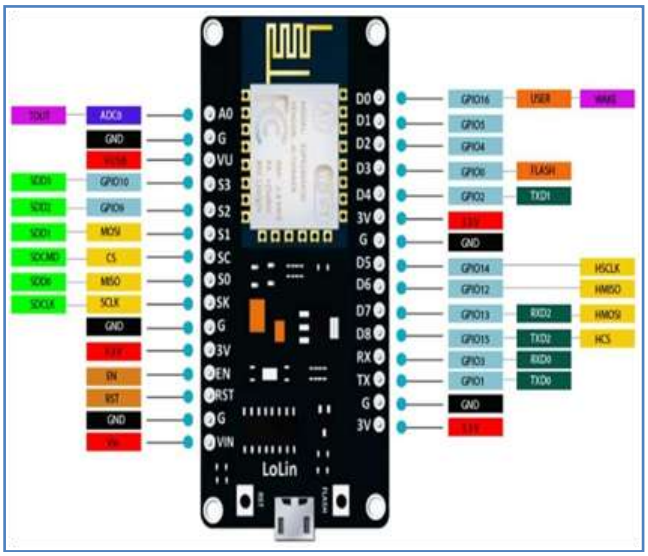

Fig. 2 The pin diagram of a NodeMCU (ver.3)

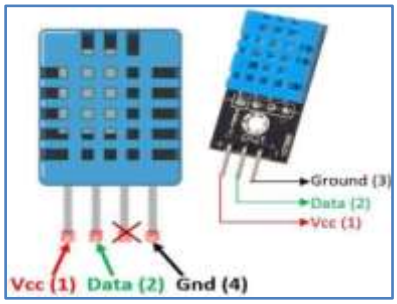

Fig. 3. DHT11SensorModule

C. MQ-135GasSensorModule [15]

The MQ-135 gas sensor works on the principle of conductivity measurement of gas. It is made of special oxide material, SnO2 which is nonconductive to clean air; however, it has a pretty performance of conductivity when exposed to combustiblegases and converts thechange of conductivity to a corresponding output voltage signal.

Ina fresh environment the sensor's output voltage is set as the reference voltage. The range of output voltage of the sensor is around 1.0 V. The sensitivity of the sensor when detects harmful gases is 0.1V per increase of gas concentration by20ppm. The measuring range of gas concentration of this type of sensor is about 10 to1000ppm. The construction and

International Journal of Engineering Applied Sciences and Technology, 2022
Vol. 7, Issue 8, ISSN No. 2455-2143, Pages 151-159
Published Online December 2022 in IJEAST (http://www.ijeast.com)

appearance of the MQ-135 gas sensor is presented in the Fig. 4 (a) & (b).

Fig. 4 (a) Front view of MQ-135GasSensor

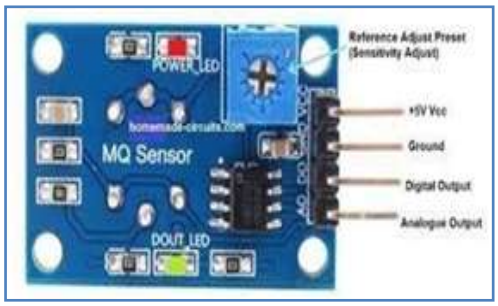

Fig. 4 (b) Back view of MQ-135GasSensor

D. Arduino IDE

The IDE stands for Integrated Development Environment. Arduino IDE is open-source software, which is used to write, compile and upload code to the Arduino boards. It supports two popular programming languages C and C++.The IDE application is compatible to different standard operating systems such as Windows, Mac OS X, and Linux etc [16].The programming code that uses by the Arduino IDE is called the Sketch. The user defined Arduino sketch is uploaded to, run in the Arduino board. The coding file of sketch is saved with the file extension name '.ino.' There are two basic Arduino sketch functions; setup() and loop().

E. Thing Speak Cloud

Thing Speak [17] was originally launched by "io Bridge" in 2010 as a service in support of IoT applications (Fig. 5).Thing Speak is open-source cloud platform service written in Ruby, which allows users to communicate with the IoT-enabled devices. It facilitates users to data access, aggregate, visualize, and analyze live data streams in the cloud. It is possible to transmit data to Thing Speak from any IoT enabled devices and can create instant visualization of the live data streams, and generate alerts. Specialty of Thing Speak is that, it has integrated support of MATLAB software from Math Works Inc.

III. INTEGRATION OF HARDWARE COMPONENTS

It is to be noted that each sensor modules need to preheat before put it to actual operation so that it can work accurately.

A. Preheat DHT11Sensor Module

The following steps were performed to preheat the DHT11 sensor module:

STEP 1: The VCC pin of the DHT11sensormoduleis connected with the VU pin of the Node MCU

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

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

STEP 4: The setup was then disconnected

Fig.5. ThingSpeak Cloud

B. Preheat MQ-135 Gas Sensor Module

The following steps were performed to preheat the MQ-135 gassens or module:

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

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

STEP 3: The NodeMCU is powered with a 12VDC via AC- D C adapter foraday.

STEP4: The setup was thend is connected.

C. Calibration of 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 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 R0 in fresh air is collected from the serial monitor of the Arduino IDE.

STEP5: The setup was thend is connected.

D. Complete Hardware Model

In order to make the system full operational the complete hardware model is embedded as shown in the Fig. 6. The

International Journal of Engineering Applied Sciences and Technology, 2022
Vol. 7, Issue 8, ISSN No. 2455-2143, Pages 151-159
Published Online December 2022 in IJEAST (http://www.ijeast.com)

successive steps for connection of all the components are given as follows;
STEP1:The Vccpin of the MQ-135 gas sensor module and DHT11sensor module was connected via Vero board with a5V adapter.
STEP 2: The Gnd pin of the MQ-135 gas sensor module, and the DHT11 sensor module and the cathode of the LED indicators were connected via Vero board with the Gnd pin of the Node MCU.
STEP 3: The analog DATA pin of the MQ-135 gas sensor module was then connected with the A0 Pin of the Node MCU.
STEP 4: The DATA pin of the DHT11 sensor module was connected with the D0 pin of the Node MCU.
STEP 5: The anode of the three LED indicators (green, yellow, and red) were connected to the D2, D3, and D4 pins of the Node MCU respectively.
STEP 6: The software code to execute the operation was then uploaded to the Node MCU.
STEP 7: The setup was then powered with 9V DC via AC-DC adapter.

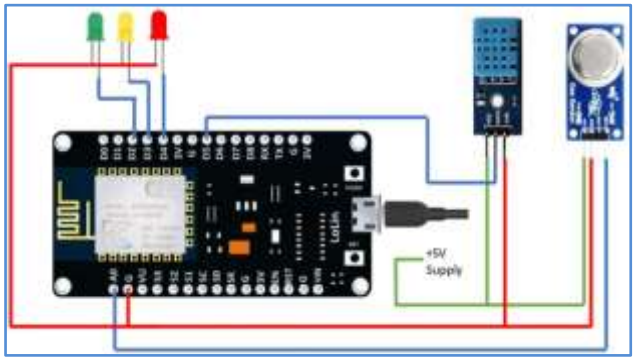

Fig. 6. Complete C circuit Diagram of the Setup

IV. THEORY OF CALIBRATION OF MQ-135 GAS SENSOR

In this work the most important part is to calibrate the MQ-135 sensor in the fresh air and then derive an equation that converts the sensor output voltage value into the convenient units PPM (parts per million). The mathematical abstraction associated with the calibration of the MQ-135sensor is derived as follows [18]-[19];
From Fig. 7 the voltage across the load resistance R_L is given by

$$V_{RL} = \frac{V_C * R_L}{R_S + R_L} \qquad (1)$$

where R_S is the sensor resistance and VC is the supply voltage to the sensor. The internal sensor resistance R_S can be solved from the equation (1) as

$$R_S = \frac{V_C * R_L}{V_{RL}} - R_L \qquad (2)$$

Fig. 8 demonstrates the concentration of a gas in part per million (ppm) with the resistance ratio of the sensor (R_S/R_0). Here, R_S denotes sensor resistance which changes with the gas concentration and R_0stands for the resistance of the sensor at any known gas concentration in the absence of other gases, or in fresh air. From the graph it can be seen that the resistance ratio in fresh air is a constant quantity and is given by:

$$\frac{R_S}{R0} = 4.4$$

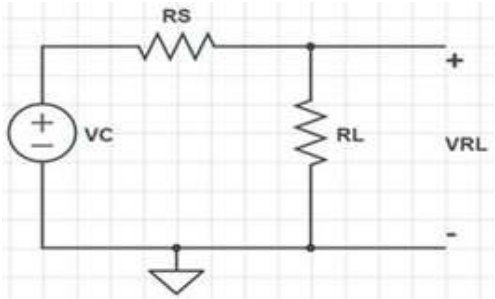

Fig. 7. Equivalent CcircuitDiagramMQ-135 gas Sensor

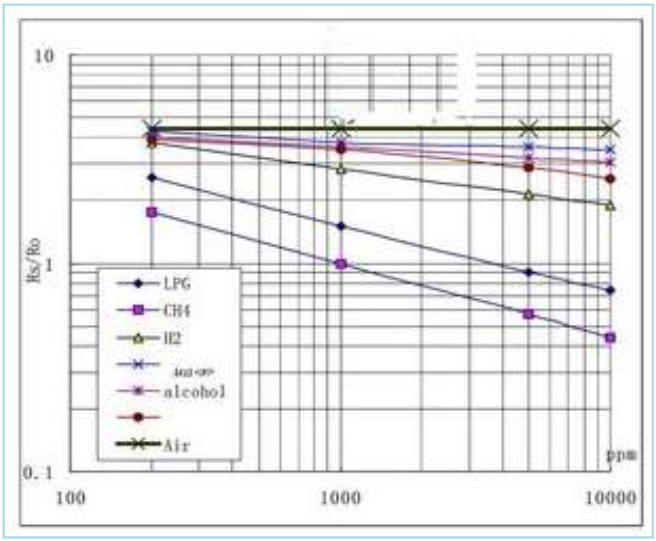

Fig. 8.Plot of gas concentration in ppm with the resistance ratio

It is need to calculate the value of R_0in order to get R_S in fresh air. This can be achieved in the Arduino IDE by converting analog average readings received from the sensor into a voltage. From the log-log graph (Fig. 8) it is possible to find the concentration of a gas at any ratio using the formula for a straight line in a log-log scale. The formula for a line in log-log scale is given by:

$$log_{10}^{y} = m * log_{10}^{x} + b \qquad (3)$$

wherey and x are the values. m is slope of the line and b is the Y intercept. Considering any two known points on the graph

International Journal of Engineering Applied Sciences and Technology, 2022
Vol. 7, Issue 8, ISSN No. 2455-2143, Pages 151-159
Published Online December 2022 in IJEAST (http://www.ijeast.com)

the value of the slope m and the intercept bfor any gas can be calculated.The equation (3) can be further rewritten as

$$log_{10}^{x} = \frac{log_{10}^{y} - b}{m} \qquad (4)$$

$$x = 10^{\frac{log_{10}^{y} - b}{m}} \qquad (5)$$

Once we get the values of m and b, any unknown gas concentration can be calculated using the formula (5). Using (2) and (5) it is possible to convert the sensor output values into PPM (Parts perMillion). The Programming codes written and executed in Arduino IDE are given in the Appendix A.1 & A.2.

V. ALGORITHM OF THE IMPLEMENTED SOFTWARE

The implemented algorithm of the proposed work has been presented in a flow diagram (Fig. 9). Software code has been configured based on this flow diagram. The measured data from the individual sensors are sending to the cloud through ThingSpeakplatform over the internet and based on the ppm value of the air quality different colorsof LEDs indicators glow. The air quality in terms of ppm, humidity and the temperature of the atmosphere can be displayed through any IoT enabled devices.

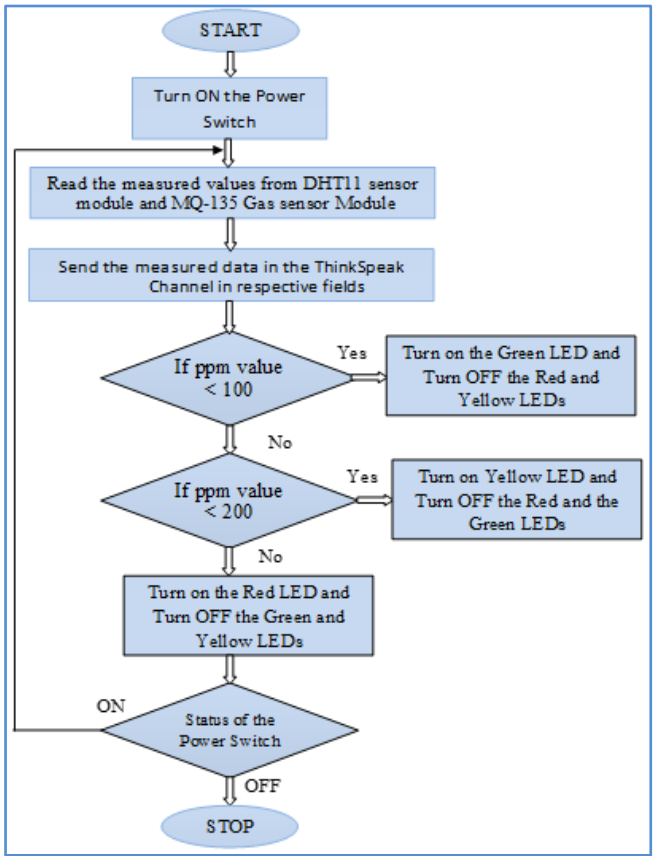

Fig. 9. Flow chart of implement Algorithm

VI. EXPERIMENTAL RESULTS

Theworkingofthedesignedprototypehasbeeninvestigatedforthe 5setsofexperimentsasdescribedinthefollowingsections. All experimental results and graphical views are displayed through Thing Speak Cloud.

Expt. 1: To demonstrate the working of the system in a warm and humid outdoor atmosphere, the experiment was performed on a warm sunny day in a local outdoor area. The results with its graphical observationsinThingSpeakCloud are presented in Fig. 10.

Expt. 2: To demonstrate the working of the system in the presence of alcoholic gases, the experiment was performed indoor in the presence of alcoholic gases. The spray of an alcoholic mixture was used to produce alcoholic vapors.The in ThingSpeakCloud is shown in the Fig. 11.

Expt. 3: To investigate the working of the system in smoky conditions, this experiment was performed in the presence of smoke near the sensor setup and the corresponding results are presented in the Fig. 12.

Expt. 4: To study the working of the system in a warm and humid outdoor atmosphere, this experiment was performed at night and results demonstrated in Fig. 13

Expt. 5: In order to investigate the performance of the system in an air-conditioned indoor atmosphere, the experiment was further performed at room temperature. All observations are displayed through ThingSpeakCloud as shown in the Fig. 14.

Fig. 10. Obs. of Expt. 1 for Temperature, Humidity and PPM

International Journal of Engineering Applied Sciences and Technology, 2022
Vol. 7, Issue 8, ISSN No. 2455-2143, Pages 151-159
Published Online December 2022 in IJEAST (http://www.ijeast.com)

Fig. 11. Obs. of Expt. 2 for Temperature, Humidity and PPM

Fig. 12. Results of Expt. 3 for Temp., Humidity and PPM

Fig. 13. Obs. of Expt. 4 for Temperature, Humidity and PPM

Fig. 14. Results of Expt. 5 for Temp., Humidity and PPM

International Journal of Engineering Applied Sciences and Technology, 2022
Vol. 7, Issue 8, ISSN No. 2455-2143, Pages 151-159
Published Online December 2022 in IJEAST (http://www.ijeast.com)

In order to validate the experimental results each set of reading are compared with the Samsung Weather App readings taken through a Smart Phone (Table -1). It has been observed that readings noted from the ThnigSpeak cloud platform are almost comparable with the results displayed in Smart Phone. In view of this study it is possible to conclude that the prototype arrangement designed in this work is highly suitable for close surveillance of the Air Quality at any time and any place.

Table -1 Experiment Results and its Comparison

Measuring Variables	Expt. No	Reading from Thing Speak Cloud	Samsung Weather App Reading	Calculated error
Temperature (in Celsius)	1	34.2	33	1.2
	2	33.3	32	1.3
	3	33.8	32	1.8
	4	34.2	33	1.2
	5	22.6	22	0.6
Humidity(in%)	1	70.0	65	5.0
	2	70.0	65	5.0
	3	74.0	70	4.0
	4	74.0	69	5.0
	5	59.0	57	2.0
AirQuality(inppm)	1	8.61	8.5	0.11
	2	42.25	42	0.25
	3	52.30	53	-0.70
	4	4.26	4.34	-0.08
	5	0.67	0.70	-0.03

VII.CONCLUSIONS

In this work, IoT basedmeasurement and display of Air Quality Index (AQI), Humidity and Temperature of the atmosphere has been performed.A low cost prototype has been design using a popular open source IoT platform, NodeMCU, which plays the primary role in control and communication. The measuring and monitoring data from different IoT based sensors are displayed through Thing Speak cloud in any connected PC or mobile phone. The quality levels of the air are also indicated via different colors of LED indicators. The working of the prototype has been investigated in different time in various environmental conditions. It has been seen that thepresent setup is able to measure the air quality in ppm, the temperature in Celsius and humidity in percentage with a considerable accuracy.The results obtained from the experiments are verified with Google App data through a Samsung smart phone. Therefore, it is possible to conclude that the designed prototype can be utilized at any place and at any time for monitoring air quality, humidity and temperature of the surrounding atmosphere successfully.

The drawback of this setup is thatit cannot measure the ppm values of the pollutant components separately. However, it can be achieved by incorporating individual gas sensors for different pollutants which in turn increases the cost of the setup and not be anessential provision to monitor the air quality. Moreover, a stable internet connection is required for continuous uploading and monitoring the data through the ThinkSpeak cloud.

VIII. APPENDIX A

A.1. Software Code for Calibration ofMQ135Sensor

```
voidsetup()
{
Serial.begin(9600);//Baudrate
pinMode(A0,INPUT);
}
voidloop( )
{
float sensor_volt; //Demark for sensor voltage variable
float RS_air; //Demark for sensor resistance variable
float R0; //Demarks variable for R0
floatsensorValue; //Demark analog readings values
for(int x = 0 ; x < 500 ; x++) //Starting of the for  loop
 {
sensorValue = sensorValue + analogRead(A0); //Addition
//500 times of analog readings of sensor
  }
sensorValue = sensorValue/500.0;         //Averaging of analog
//readingsof sensor
sensor_volt = sensorValue*(5.0/1023.0); //Conversionof
//average reading to voltage
RS_air = ((5.0*10.0)/sensor_volt)-10.0; //Compute RS  //for
fresh air
```

```
R0 = RS_air/4.4; // Workout R0
 Serial.print("R0 = "); //Displaying the value  "R0"
Serial.println(R0); //Vvalue of R0
delay(1000);//Hold-up for 1 second
}
```

A.2. Software Code to Calculate Gas Concentration

```
#include <ESP8266WiFi.h> #include <DHT.h>
#include <ThingSpeak.h>
DHT dht(D5, DHT11);
#define LED_GREEN D2
#define LED_YELLOW D3
#define LED_RED D4
#define MQ_135 A0
int ppm=0;
float m = -0.3376; // Define Slope
float b = 0.7165;                // Define Y-Intercept
float R0 = 3.12;   // Define Resistance of the sensor in fresh air
//frompreceding program WiFiClientclient;
longmyChannelNumber = 123456;                // Channel id
const char myWriteAPIKey[] = "API_Key";

void setup() {
// Place setup code again here, to run once:
Serial.begin(9600);
pinMode(LED_GREEN,OUTPUT);
pinMode(LED_YELLOW,OUTPUT);
pinMode(LED_RED,OUTPUT);
pinMode(MQ_135, INPUT);
WiFi.begin("WiFi_Name", "WiFi_Password");
while(WiFi.status() != WL_CONNECTED)
{
delay(200); Serial.print(".");
}
Serial.println();
Serial.println("NodeMCU is connected!");
Serial.println(WiFi.localIP());
dht.begin(); ThingSpeak.begin(client);
}

void loop() {
floatsensor_volt;//Demark for sensor voltage variable float RS
_gas;     //Demark for  sensor resistance variablefloat ratio;
          //Demarks variable for the ratio RS/R0
intsensorValue;   //Store the analog valuesfrom MQ-135
float h;
float  t;
floatppm_log;                //Obtain value of ppm in linear scale
as// per the ratio value
float ppm;          //Convert ppm value to log scale
h = dht.readHumidity();
delay(4000);
t = dht.readTemperature();
delay(4000);
sensorValue = analogRead(gas_sensor); //Take analog values
// from sensor
sensor_volt = sensorValue*(5.0/1023.0);      //Conversion      of
//analogvalues to voltage signal
RS_gas = ((5.0*1.0)/sensor_volt)-1.0;//Compute value of RS
ratio = RS_gas/R0;          // Evaluate ratio RS_gas/RS_air
ppm_log = (log10(ratio)-b)/m; //Evaluate ppm value in linear
//scale as per the ratio value
ppm = pow(10, ppm_log); //Convertion of  ppm to log scale
Serial.println("Temperature: " + (String) t);
Serial.println("Humidity: " + (String)h);
Serial.println("OurdesiredPPM="+(String)ppm);
ThingSpeak.writeField(myChannelNumber,1,t,myWriteAPIK
ey);
delay(20000);
ThingSpeak.writeField(myChannelNumber,2,h,myWriteAPIK
ey);
delay(20000);
ThingSpeak.writeField(myChannelNumber,3,ppm,myWriteA
PIKey);
delay(20000);
if(ppm<=100)
{
digitalWrite(LED_GREEN,HIGH);
digitalWrite(LED_YELLOW,LOW);
digitalWrite(LED_RED,LOW);
elseif(ppm<=200)
{
digitalWrite(LED_GREEN,HIGH);
digitalWrite(LED_YELLOW,LOW);
digitalWrite(LED_RED,LOW);
}
else
{
digitalWrite(LED_GREEN,HIGH);
digitalWrite(LED_YELLOW,LOW);
digitalWrite(LED_RED,LOW);
}
delay(20000);}
```

IX. REFERENCES

[1]. Kumar SaiK. B., Mukherjee S. and Sultana HP. (2019). Low Cost IoT Based Air Quality Monitoring Setup Using Arduino and MQ Series Sensors With Dataset Analysis. Int. conf. on recent trends in advanced computing, Procedia Computer Science, vol. 165, (pp. 322–327).

[2]. Kumar P. et. al. (2016). Indoor air quality and energy management through real-time sensing in commercial buildings. Energy and Buildings, vol. 111, (pp. 145-153).

[3]. Saikumar C.V., Reji M. and Kishoreraja P.C. (2017). IOT Based Air Quality Monitoring System. International Journal of Pure and Applied Mathematics, vol. 117, no. 9, (pp. 53-57).

[4]. Liu S., Xia C. and Zhao Z. (2016). A low-power real-time air quality monitoring system using LPWAN based on LoRa. 13th IEEE International Conference on Solid-State and Integrated Circuit Technology (ICSICT), (pp. 379-381). China.

[5]. Hu S-C., WangY-C., HuangC-Yu.and TsengY-C. (2009). A vehicular wireless sensor network for CO2 monitoring. IEEE Sensors Conference,New Zealand

[6]. North R. et. al. (2008). A mobile environmental sensing system to manage transportation and urbanair quality. IEEE Int. Symposium on Circuits and Systems (ISCAS 2008),(pp.1994 – 1997).

[7]. Jung Y. J.et. al. (2008). Air Pollution Monitoring System based on GeosensorNetwork. IEEE Int. Geoscience Remote Sensing Symposium (IGARSS 2008), vol. 3, (pp. 1370 – 1373).

[8]. Jayakumar A. A. et. al. (2021). IoT Based Air Pollution Monitoring System. International Research Journal of Engineering and Technology (IRJET). vol. 8, no. 3, (pp. 2458-2462).

[9]. Sinha N., PujithaK. E. and AlexJ. S. R. (2015). Xively based sensing and monitoring system for IoT. 2015 IEEE Int. Conf. on Computer Communication and Informatics (ICCCI 2015), (pp. 1-6), Coimbatore, India.

[10]. KellyS. D. T., SuryadevaraN. K. and MukhopadhyayS. C. (2013). Towards the implementation of IoT for environmental condition monitoring in homes. IEEE Sensors Journal, vol. 13, no. 10.

[11]. Pasha S. (2016). Thingspeak Based Sensing and Monitoring System for IoT with Matlab Analysis. International Journal of New Technology and Research (IJNTR), vol. 2, no. 6, (pp. 19-23).

[12]. Dhanalakshmi M., et al. (2021). A survey paper on vehicles emitting air quality and prevention of air pollution by using IoT along with machine learning approaches. Turk. Journal of Comput. Math. Educ. (TURCOMAT), vol. 12, no. 11, (pp. 5950–5962).

[13]. https://components101.com/development-boards/nodemcu -esp8266-pinout-features-and-datasheet.

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

[15]. https://pdf1.alldatasheet.com/datasheet-pdf/view/1307647/ WINSEN/MQ135.html

[16]. https://www.javatpoint.com/arduino-ide

[17]. https://thingspeak.com

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

[19]. Understanding a Gas Sensor: Tutorial, Available [online] https://jayconsystems.com/blog/ understanding -a-gassensor