Air Quality Monitoring

INTRODUCTION

Air is getting polluted because of 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, increasing in population, vehicle use which can affect human health. Particulate matter is one of the most important parameter having the significant contribution to the increase in air pollution [2]. This creates a need for measurement and analysis of realtime air quality monitoring so that appropriate decisions can be taken in a timely period. This paper presents a real-time standalone air quality monitoring. Internet of Things 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 in webpage so that we can monitor it very easily. In this IoT project, you can monitor the pollution level from anywhere using your computer or mobile [1]. The setup will show the air quality in PPM in webpage so that we can monitor it very easily. In this IoT project, you can monitor the pollution level from anywhere using your computer or mobile. Air condition is much polluted. In recent years, car emissions, chemicals from factories, smoke and dust are everywhere. That is the reason why now air condition is much polluted. The effect of air pollution is very bad for our health, especially for place where the air in our body is taken for breathing. In our lungs may cause some diseases, such as asthma, cough, lung disorders [1].The air pollution cannot be detected by human feelings. The air pollution may contain a lot of dangerous substances, such as LPG gas, carbon monoxide, and methane [2]. Substances in the polluted air are very dangerous. For example, if the carbon monoxide is above 100ppm, it makes human feel dizzy, nauseous, and within minutes they could die.

This research makes human find out which content of the air is polluted. With module node mcu esp8266, we can monitor the air pollution remotely, because there is a Wi-Fi in nodemcu esp8266. This makes the air condition can be monitored every time.

LITERATURE REVIEW

1. IOT Based Air Pollution Monitoring System Using Node MCU

The level of pollution has increased with times by lot of factors like the increase in population, increased vehicle use, industrialization and urbanization which results in harmful effects on human wellbeing by directly affecting health of population exposed to it. In order to monitor In this project we are going to make an IOT Based Air Pollution Monitoring System in which we will monitor the Air Quality over a web server using internet and will trigger a alarm when the air quality goes down beyond a certain level, means when there are sufficient amount of harmful gases are present in the air like CO2, smoke, alcohol, benzene and NH3. It will show the air quality in

PPM on the LCD and as well as on webpage so that we can monitor it very easily. In this IOT project, you can monitor the pollution level from anywhere using your computer or mobile.

2. IOT Based Air Pollution Monitoring System

Air pollution is the biggest problem of every nation, whether it is developed or developing. Health problems have been growing at faster rate especially in urban areas of developing countries where industrialization and growing number of vehicles leads to release of lot of gaseous pollutants. Harmful effects of pollution include mild allergic reactions such as irritation of the throat, eyes and nose as well as some serious problems like bronchitis, heart diseases, pneumonia, lung and aggravated asthma. According to survey, due to air pollution 50,000 to 100,000 premature deaths per year occur in the U.S. alone. Whereas in EU number reaches to 300,000 and over 3,000,000 worldwide. IOT Based Air Pollution Monitoring System monitors the Air quality over a web server using Internet and will trigger an alarm when the air quality goes down beyond a certain threshold level, means when there are sufficient amount of harmful gases present in the air like CO2, smoke, alcohol, benzene, NH3, LPG and NOx. It will show the air quality in PPM on the LCD and as well as on webpage so that it can monitor it very easily.

3) IOT Based Air Quality Monitoring System

The main objective of this project is to monitor the air eminence in industrial and urban areas. The proposed outline includes a set of gas sensors (CO, and NO2) that are positioned on masses and structure of a IOT (Internet of things) and a dominant server to support both short-range realtime incident management and a continuing deliberate planning. In this Arduino platform is used to communicate the data simply and quickly. WSN (Wireless sensor network) acts as the trans receiver. This provide a real-time low rate monitoring system over the use of low rate, low information rate, and little control wireless communication technology. The projected monitoring system can be transferred to or shared by different applications. Through IOT we can able to visualize the values from the globe.

The problem in this paper is they haven't calibrated the sensor and not even converted the sensor output value into PPM. As per the guidelines by UN Data, 0-50 is SAFE value and 51-100 is moderate. Delhi is the most polluted city in the world recorded 350PPM. While using two sensors, as both sensors have internal heat element, it draws more power(P= VxI), so though the both sensors are turned ON, its output voltage levels varies and shows unpredicted values due to insufficient drive. So we used a 9V battery and a 7805 family REGULATOR for the CO sensor MQ7. For MQ135 we have given the power from Arduino only.

(4) Arduino Based Weather Monitoring System

This Paper makes use of 3 sensors to measure the weather/environment factors such as temperature, humidity, light intensity, dew point and heat index. The values read from the sensors are processed by the Arduino micro-controller and stored in a text file which can be processed upon to derive analysis. The readings are also displayed on an on board LCD for quick viewing. All these readings can be analyzed

to get the weather characteristics of a particular area and record the weather pattern. These recorded parameters are essential and vary from places to places.

(5) IoT Based Air Pollution Monitoring System

The level of pollution has increased with times by lot of factors like the increase in population, increased vehicle use, industrialization and urbanization which results in harmful effects on human wellbeing by directly affecting health of population exposed to it. In order to monitor In this project we are going to make an IOT Based Air Pollution Monitoring System in which we will monitor the Air Quality over a web server using internet and will trigger a alarm when the air quality goes down beyond a certain level, means when there are sufficient amount of harmful gases are present in the air like CO2, smoke, alcohol, benzene and NH3. It will show the air quality in PPM on the LCD and as well as on webpage so that we can monitor it very easily. we have used MQ135 sensor which is the best choice for monitoring Air Quality as it can detects most harmful gases and can measure their amount accurately. In this IOT project, you can monitor the pollution level from anywhere using your computer or mobile.

This paper assumed completely wrong assumption where they have showed the output 997PPM as the fresh air, where Delhi which is the most polluted city recording 350PPM. Its clear understanding that they haven't calibrated the sensor and didn't even convert the raw sensor data into PPM using derivations we did. They have used LocalHost which is limited where they are able to see the output only on the laptop within the experimental setup connected. But we have used premium iot platforms which are highly secured and open source IoT platform.

PROPOSED METHODOLOGY

We used Thingspeak IoT platform and we clearly defined the derivations that mentions the correct ppm on the screen with correct calibration. We have implemented it with less cost i.e., when we are pushing the data to the cloud, no need to see the output on LCD which adds more cost to the project [1]. When we are targeting IoT as a platform, our intension should be to present the idea on internet using the platforms like thinger.io or thingspeak or Cayenne website which are beautifully designed to present the output and even able to download the dataset. When doing an experiment air quality monitoring, no need to use LPG or methane detecting sensors as it is used for Home/office safety. We have used WiFi to push the data onto the cloud rather using GSM or GPRS module [2]. The problem in another paper that cited at [3] hasn't calibrated the sensor and not even converted the sensor output value into PPM. As per the guidelines by UN Data, 0-50 PPM is SAFE value, 51-100 is moderate as shown in figure 1. Delhi is the most polluted city in the world recorded around 250PPM. As we are using two sensors, both of them have internal heat element, it draws more power(P= V*I), so though the both sensors are turned ON, its output voltage levels varies and shows unpredictable values due to insufficient power drive. So we used a 9V battery and a 7805 family LM7805 Regulator for the CO sensor MQ7.

We have used Arduino Uno Development kit that comes with ATMega328P microcontroller. In order to provide WiFi Support for it, we have used cost effective ESP-01 WiFi module which helps us to connect to the ThingSpeak Platform. The connections between them is mentioned in the connections diagram.

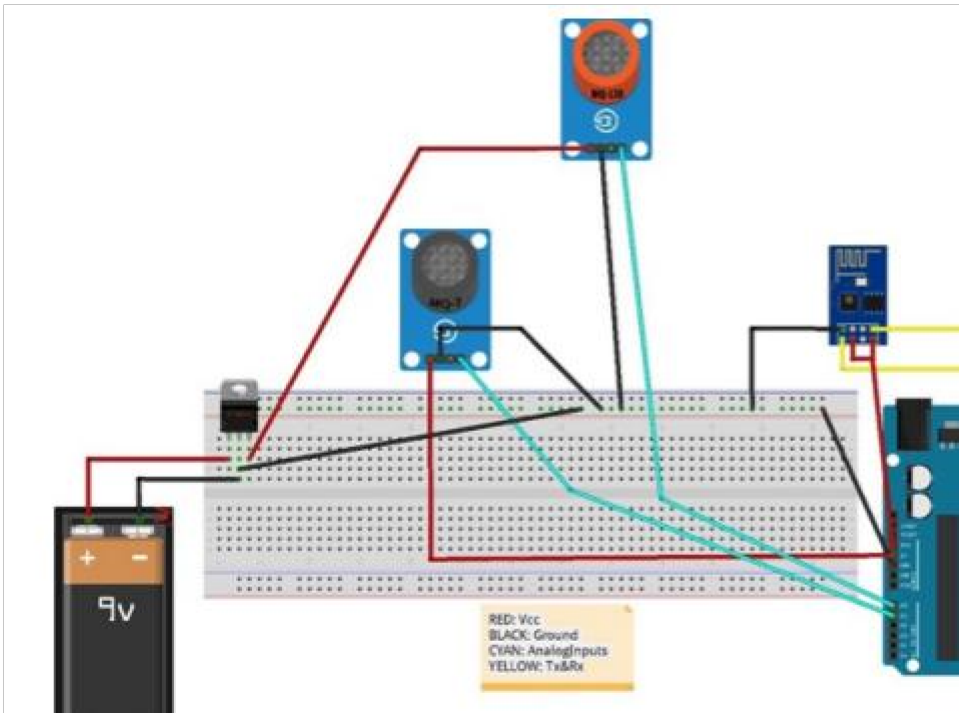

DIAGRAM MADE THROUGH FRITZIG

The most important step is to calibrate the sensor in fresh air and then draw an equation that converts the sensor output voltage value into our convenient units PPM (parts per million). Here are the mathematical calculations derived [6]. Fig 5: Internal circuit diagram of MQ135 sensor RS and RL combined From Ohm's Law, at constant temperature, we can derive I as follows:

$$I = V/ R \qquad (1)$$

From fig , equation 1 is equivalent to

$$I = Vc / Rs + Rl \qquad (2)$$

From , we can obtain the output voltage at the load resistor using the value obtained for I and Ohm's Law at constant temperature. V = I*R

$$VRl = [Vc/ Rs + Rl] * RL \qquad (3)$$

$$VRl = [Vc * Rl [Rs + Rl] \qquad (4)$$

$$(VRl * Rs) + (VRl * Rl) = Vc * RL \qquad (5)$$

$$VRl * Rs = (Vc * Rl) - (VRl * Rl) \qquad (6)$$

$$Rs = (Vc * Rl) - (VRl * Rl) VRl \qquad (7)$$

$$Rs = (Vc * Rl) VRl - Rl \qquad (8)$$

Equation 9 help us to find the internal sensor resistance for fresh air

$$Rs = (Vc * Rl)/ VRl - Rl \qquad (9)$$

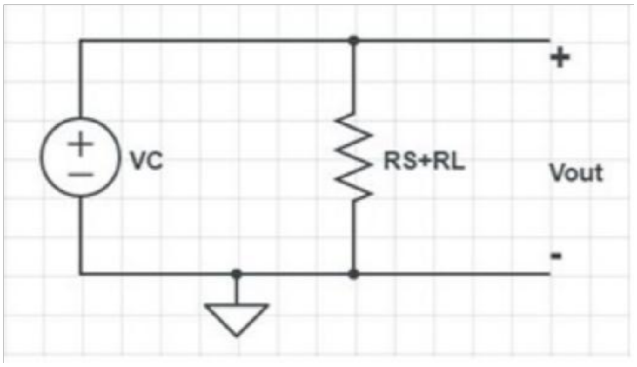

Fig. 3. Internal circuit diagram of MQ135

Equation 10 is depicted from the datasheet mentioned in Fig 6. To calculate R0, we will need to find the value of the Rs in fresh air. This will be done by taking the analog average readings from the sensor and converting it to voltage. Then we will use the Rs formula to find R0. First of all, we will treat the lines as if they were linear. This way we can use one formula that linearly relates the ratio and the concentration. By doing so, we can find the concentration of a gas at any ratio value even outside of the graph's boundaries. The formula we will be using is the equation for a line, but for a log-log scale. The formula for a line is [9]: From above Figure 3, we try to derive the following calculations.

$$y = mx + b \quad (11)$$

For a log-log scale, the formula looks like this:

$$\log10y = m * \log10x + b$$

Now that we have m, we can calculate the y intercept. To do so, we need to choose one point from the graph (once again from the CO2 line). In our case, we chose (5000,0.9) $\log(y) = m * \log(x) + b$ (17)

$b = \log(0.9) - (-0.318) * \log(5000)$ (18) $b = 1.13$ (19)

Now that we have m and b, we can find the gas concentration for any ratio with the following formula:

$$\log(x) = \log(y) - b\,m \quad (20)$$

However, in order to get the real value of the gas concentration according to the loglog plot we need to find the inverse log of x: $x = 10 / m$
Using equations 9 and 21, we will be able to convert the sensor output values into PPM (Parts per Million). Now we developed the Code and flashed into the Arduino Uno giving proper connections as mentioned

RESULTS

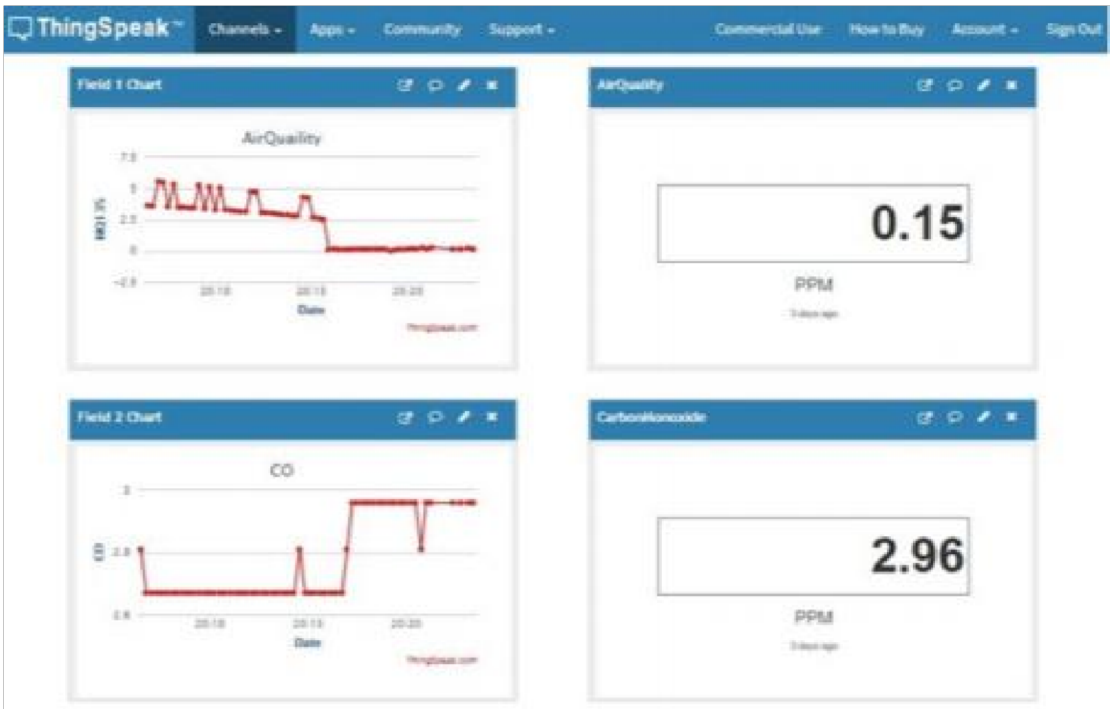

Fig. 5. Output on Thingspeak

After connecting the wifi successfully to the ESP-01, it gets established with Thingspeak account with the help of the API key of our account provided. Thingspeak needs 15 seconds of refresh interval to push to the data. Fig 7 shows the field charts of MQ135 and MQ7 sensor values which made to convert to PPM [7] [8]. Fig 8 showing the graphical analysis of the values collected with time on X axis and AirQuality PPM on Y axis.

Coding

```
 import serial
import time


def read_sds011_data(ser):
    data = []
    while len(data) < 10:
        data.append(ser.read(1))
    return data

def parse_sds011_data(data):
    pm25 = (data[3] + data[2]) / 10.0
```

```
    pm10 = (data[5] + data[4]) / 10.0
    return pm25, pm10


def main():
    ser = serial.Serial('/dev/ttyUSB0')  # Change this to the actual serial port of your
SDS011
    ser.timeout = 2  # Set a timeout for reading data


    try:
        while True:
            ser.write(b'\xAA\xB4\x04\x00\x00\x00\x00\x00\x00\xFF\xFF\x05')
            time.sleep(2)
            data = read_sds011_data(ser)
            pm25, pm10 = parse_sds011_data(data)
            print(f'PM2.5: {pm25} µg/m³, PM10: {pm10} µg/m³')
            time.sleep(10)  # Read data every 10 seconds
    except KeyboardInterrupt:
        ser.close()

if __name__ == '__main__':
    main()
```