Iot Based Air Quality Data Acquisition System

P.G Bhavyashree, Pooja.K,Rohan, Shushal.P
Department of Electrical and Electronics Engineering
Dayananda Sagar College of Engineering
Bengaluru,

---***---

Abstract - - At present, the global environment is facing serious air pollution problems, which pose a serious threat to biology, ecosystems and climate, especially human health in densely populated urban areas. Air quality monitoring systems require special functions, such as accurate measurement and analysis of parameters, and timely taking very simple decisions to monitor and control air quality. The ultimate goal of these systems is to provide accurate and timely information. Due to many factors such as population growth, increased vehicle use, industrialization and urbanization, pollution has increased over time. These factors have an adverse impact on human well-being and directly affect public health. In order to monitor air pollution, we will create an air pollution monitoring system based on the Internet of Things. We will monitor the air quality through a web server on the Internet for air pollutants like CO_2, smoke, alcohol, benzene NO_X and SO_X. It displays PPM air elements on a Web page so that we can control them very easily. In this project, you can use a computer or mobile device to monitor pollution anytime and anywhere.

Key-Terms – Air Pollution, PPM, Pollutants,Quality,Health

1.INTRODUCTION

With the rise of civilization and the increase of industrial and automobile pollutant emissions, atmospheric conditions are deteriorating every year [1]. Although air is an important resource, many people are indifferent to the seriousness of air pollution or have only recently realized it. Different types of pollutants such as CO, SO_2, CO_2, and NO_2 are the most dangerous and serious pollutants, which can lead to climate change and life-threatening diseases [2]. According to the World Health Organization (WHO), 90% of the population currently breathes polluted air. Air pollution causes 7 million deaths each year, of which 1.7 million are in India. In addition, recent global air pollution issues (such as ozone depletion) and air pollutants have also had a negative impact on the ecosystem [3]. Therefore, air quality control and management is a serious problem.

Severe air pollution can cause various health problems, even for healthy people who may experience respiratory irritation or shortness of breath during exercise or outdoor activities. The actual risks or side effects depend on your current health, the type and concentration of pollutants, and the duration of exposure to pollutants. Severe air pollution can cause direct health problems, including Heart and respiratory infections along with increased pressure on the heart and lungs and they can have an irreversible impact on health. For example, like:

- Accelerated lung aging
- Loss of lung function and decreased lung function
- The development of diseases such as asthma, bronchitis, emphysema, and possibly cancer.
- Reduced life expectancy.

There are many ways to measure air pollution, mainly the traditional air quality control and air monitoring methods. The Central Pollution Control Board (CPCB) recommends several air monitoring methods based on certain standards, with the help of these methods to monitor pollutants in real time. Methods are constantly expanding, and the demand for new methods is also increasing. In recent years, the emergence of technologies such as the Internet of Things (IoT) and cloud computing has opened up new possibilities for real-time monitoring in various fields. Because this technology uses sensor networks to automatically transmit, process, analyze, and visualize data, the combination of these new technologies also has great advantages for improving air quality monitoring [6]

2. COMPONENTS OVERVIEW

1) *Raspberry pie 3B+:* This product's key feature include a high-performance 64-bit quad-core processor, dual-display support at resolutions up to 4K via a pair of micro-HDMI ports, up to 1GB of RAM, dual-band 2.4/5.0 GHz wireless LAN, Bluetooth 4.0, Gigabit Ethernet, USB 3.0.

2) *Arduino Uno:* It is an open source microcontroller board base on the Microchip ATmega328P. It has an operating voltage of 5 volts and requires an input voltage of 7 to 20 volts. It contains 14 digital I/O pins of which 6 can provide PWM output and has one UART, I2C and SPPI. Arduino Uno has 6 analog input pins. DC current per I/O Pin is 20 mA and DC current for 3.3V Pin is 50 mA.

3) *MQ-2 Sensor:* A sensor which detects the concentration of LPG, Butane, Methane and mainly smoke. The concentration of the gas depicts the anolog output level of the sensor.The range of the

sensor varies from 200ppm to 10000ppm. Capability of gas detection varies with the density of gas near the sensors.

4) *MQ-3 Sensor:* This is a sensor which is used to detect the Alcohol, Benzene, CO, CH4, Hexane and LPG. The different gases can be identified by the change in the resistance value. There is a high sensitivity towards alcohol and very small sensitivity to benzene.The range of the alcohol sensor is 0.4mg/L to 4mg/L.

5) *MQ-4 Sensor:* This is the natural gas sensor which detects CNG along with propane and butane. It has a high sensitivity for CNG and methane, also a good sensitivity for propane and butane. It is responsiveness and is extremely reliable. It has a long life and has a low cost. This sensor is widely used in the gas leak detection.

6) *MQ-9 Sensor:* Sensor is used for measuring carbon monoxide, methane and LPG i.e., flammable gases. We usually use this in gas leakage detectors in industries, factories etc.

7) *MQ-135 Sensor:* This sensor is used to detect the air quality by measuring components like CO, Ammonia, Benzene, Alcohol and Smoke. This is also known as a NOx detector which determines the concentration and has a high sensitivity to Ammonia, Sulphide and Benzene steam.The range is between 10-300ppm for NH3, 10-1000ppm for benzene, 10-300 for Alcohol.

8) *DHT-11 Sensor:* This sensor is used to find the temperature and humidity of the environment around it. It is a 4 pin sensor module consisting of VCC=3.5/5v,data pin,ground pin etc.

9) USB Connector: Used for connecting Ardunio Uno to Raspberry pi module.

3. HARDWARE CONFIGURATIONS

The circuit built for air quality monitoring consists of sensors, an analog to digital converter, here it is Arduino Uno and a raspberry pi which is as an IoT device also providing the necessary DAS.
The connection is very simple. Each of the sensors are connected to the input pins of the Arduino uno using jumper cables. From the Arduino uno, with the help of USB cable, it is connected to Raspberry pi board. To configure the Pi board, we first install the Raspbian OS on to the SD card which is etched onto the pi board. The OS is accessed by connecting mouse, keyboard and a monitor to the pi board [21].
Here, we make sure that it is connected to a WIFI network to which the programmer's laptop is connected. Then using a software called as VNC viewer, we access the Raspbian OS on to our laptop.

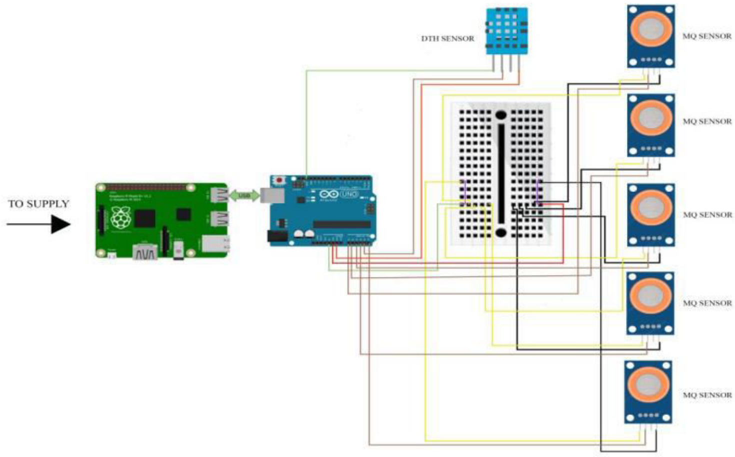


Fig -1. Circuit Set-up

4. SOFTWARE CONFIGURATIONS

The code for the sensors connected to the Arduino Uno is as follows:

```
#include <dht.h>
dht DHT;
#define DHT11_PIN 11
/////////////////////////////
int sensorValue1;
int sensorValue2;
int sensorValue3;
int sensorValue4;
int sensorValue5;

void setup()
{
  Serial.begin(9600);

}

void loop()
{

  int chk = DHT.read11(DHT11_PIN);
  int i = DHT.temperature;
  Serial.print("Temp = ");
  Serial.println(i);
  /////////////////////////////////////////////////////////////

  int j = DHT.humidity;
  Serial.print("Humidity = ");
  Serial.println(j);
  /////////////////////////////////////////////////////////////
  sensorValue1 = analogRead(A0);       // read analog input pin 0
  Serial.print("MQ3=");
  Serial.println(sensorValue1);              // prints the value read
  /////////////////////////////////////////////
  sensorValue2 = analogRead(A1);       // read analog input pin 0
  Serial.print("MQ135=");
  Serial.println(sensorValue2);              // prints the value read
  /////////////////////////////////////////////
  sensorValue3 = analogRead(A2);       // read analog input pin 0
  Serial.print("MQ9=");
  Serial.println(sensorValue3);              // prints the value read
  /////////////////////////////////////////////
  sensorValue4 = analogRead(A3);       // read analog input pin 0
  Serial.print("MQ2=");
  Serial.println(sensorValue4);              // prints the value read
  /////////////////////////////////////////////
  sensorValue5 = analogRead(A4);       // read analog input pin 0
  Serial.print("MQ4=");
  Serial.println(sensorValue5);              // prints the value read
  /////////////////////////////////////////////
  Serial.println("-----------------------------------------------");
  delay(1000);
}
```

1)Initially we start by including the DHT library in the header, this library helps us to read the data from the DHT-11 sensor.

2)We then define the pin to which the DHT sensor is connected.

3)Next, we include the input of the other sensors used, which are MQ-3 , MQ-135 , MQ-9 , MQ-2 , MQ-4.

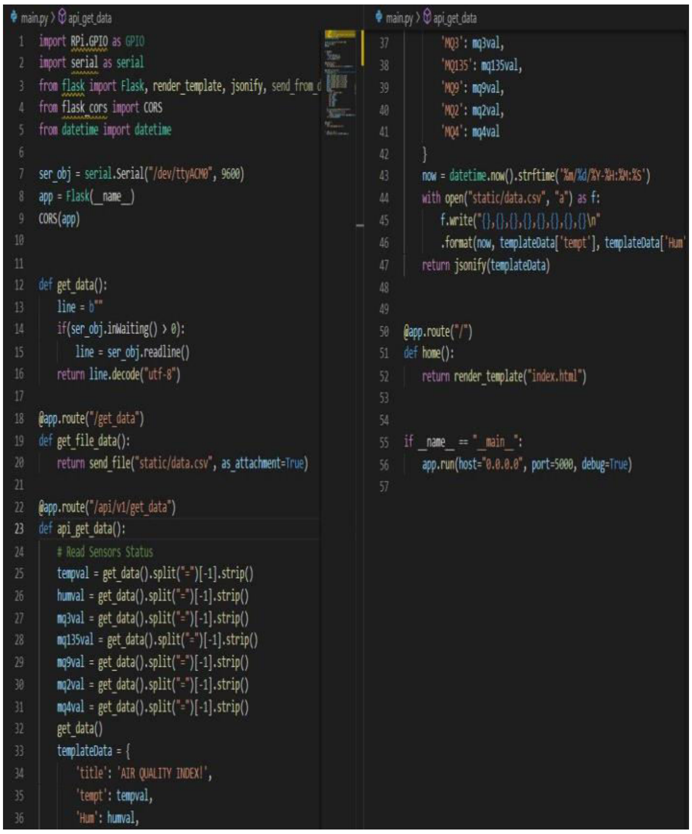
4)In the setup (), we initialize the Serial Monitor at a baud rate of 9600 for debugging purposes.

5)To get Temperature and Humidity, we just need to use the temperature() and humidity() method on the DHT object. By giving the object name as dht.temperature and dht humidity.

6)In loop function () the sensor value is read by analogRead() function and displayed on the serial monitor. This is used to read the data from the various MQ sensors.

7)At the end, there's a delay of 2 seconds. This delay is needed to give enough time for the sensor to take readings.

The back-end code for website implementation is as follows:



```
main.py > api_get_data
import RPi.GPIO as GPIO
import serial as serial
from flask import Flask, render_template, jsonify, send_from_d
from flask_cors import CORS
from datetime import datetime

ser_obj = serial.Serial("/dev/ttyACM0", 9600)
app = Flask(__name__)
CORS(app)


def get_data():
    line = b""
    if(ser_obj.inWaiting() > 0):
        line = ser_obj.readline()
    return line.decode("utf-8")

@app.route("/get_data")
def get_file_data():
    return send_file("static/data.csv", as_attachment=True)

@app.route("/api/v1/get_data")
def api_get_data():
    # Read Sensors Status
    tempval = get_data().split("=")[-1].strip()
    humval = get_data().split("=")[-1].strip()
    mq3val = get_data().split("=")[-1].strip()
    mq135val = get_data().split("=")[-1].strip()
    mq9val = get_data().split("=")[-1].strip()
    mq2val = get_data().split("=")[-1].strip()
    mq4val = get_data().split("=")[-1].strip()
    get_data()
    templateData = {
        'title': 'AIR QUALITY INDEX!',
        'tempt': tempval,
        'Hum': humval,
        'MQ3': mq3val,
        'MQ135': mq135val,
        'MQ9': mq9val,
        'MQ2': mq2val,
        'MQ4': mq4val
    }
    now = datetime.now().strftime('%m/%d/%Y-%H:%M:%S')
    with open("static/data.csv", "a") as f:
        f.write("{},{},{},{},{},{},{},{}\n"
        .format(now, templateData['tempt'], templateData['Hum'
    return jsonify(templateData)


@app.route("/")
def home():
    return render_template("index.html")


if __name__ == "__main__":
    app.run(host="0.0.0.0", port=5000, debug=True)
```

Flask is the programming language that is being used to configure the data obtained from the serial transmission and the raspberry pi. Necessary library files are included and the serial port is declared to a variable. Using GET method, the transmitted data is obtained, which is decoded and stored in excel file.

Also a function is called to ensure that the same transmitted values are being stored in the Web template designed so as to obtain the real time operating data for every 3s.

The layout for the excel sheet is set which mentions the date and time of when the data is stored in the excel file. The web-page is rendered through a function so that when the server runs, the web-page is displayed on the port defined.

5.FLOW CHART

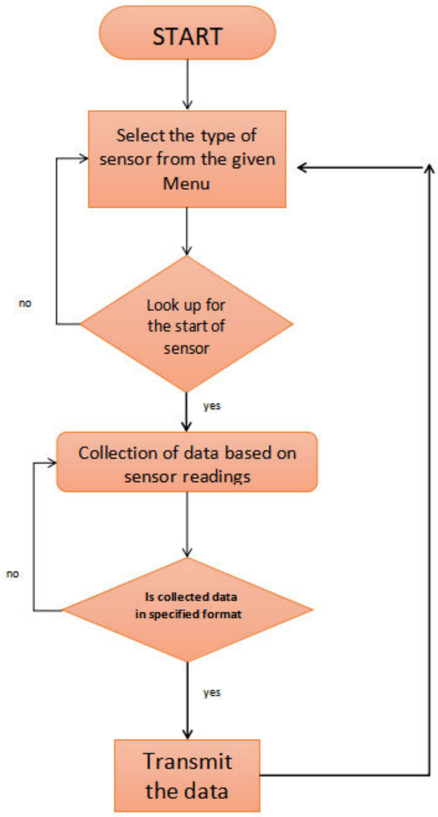


Fig -2. Transmission-Block

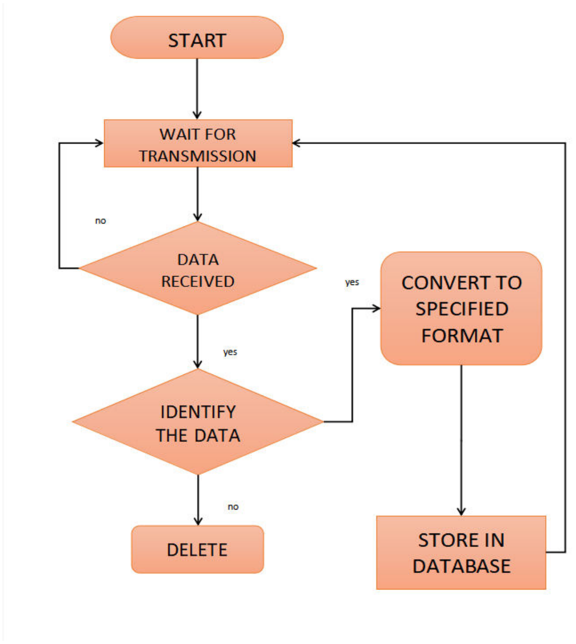


Fig. 3. Reception-Block

There are two flow charts with respect to transmission and reception :

For Transmission:

1) User can start the circuit setup,once the setup is completed ,one must look up for the start of the required sensor .
2) After the sensor starts collecting data the ,it is sent back to the raspberry pie module where the data collection is analyzed further.
3) This data many contain a garbage value which has to be deleted manually or using automation.
4) The data so obtained after deletion of garbage value is further processed to obtain specified format.
5) This formatted data is transmitted from the pie module to web page.

For reception:

1) In order to upload on the website one has to wait for the transmission to complete.
2) After receiving all the data from the module , the data is further evaluated for garbage values obtained due to transmission.
3) From the above step the data further converted to user specified format mainly with respect to time in a csv file.
4) This csv file cant be obtained from the website in the form of excel sheet where data is collected every 3 min is noted down in the database.

6. WORKING

The working of the circuit can be explained by the concept of transmission and reception. The basics of data acquisition system is rooted in this project very deeply.All the sensors are powered through the Arduino uno which is in return powered by the raspberry pi which is given a DC supply of 5V [17]. The sensors are connected through their analog pins to the respective analog input pins of the Arduino board. The supply is given from the 5V pin on the Arduino board and ground pin of it is connected to the gnd pin of the sensors[22].

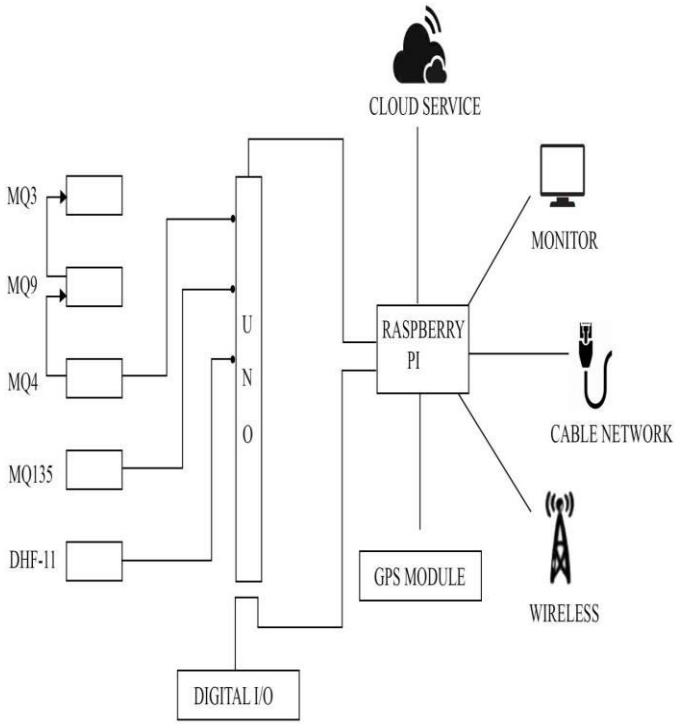


Fig -4. Block Diagram

After we have connected the sensors, the Arduino is loaded with the program which is configured to obtain the data. This data is transmitted to the raspberry pi where the data will be sent to the web-page.

Raspberry pi is configured through a back-end programming with Flask language to make sure the real time data is being sent to the web-page. We use raspberry pi because of the availability of a WiFi module [19].

To set up our output screen, we develop a web page using HTML, CSS and JavaScript. We intricate the back-end programming to obtain a user interface. The data will be programmed to be updated for every 2 minutes interval. The webpage will have a container which shows us the real-time operating data indicating the concentration of the particular substitute of air. Also, we would be able to download this data obtained over the active period of the device [15].

7. RESULTS

The real time data obtained from the prototype is collected from the raspberry-pi module and sent to a cloud server, which in turn displays the output on the webpage designed.

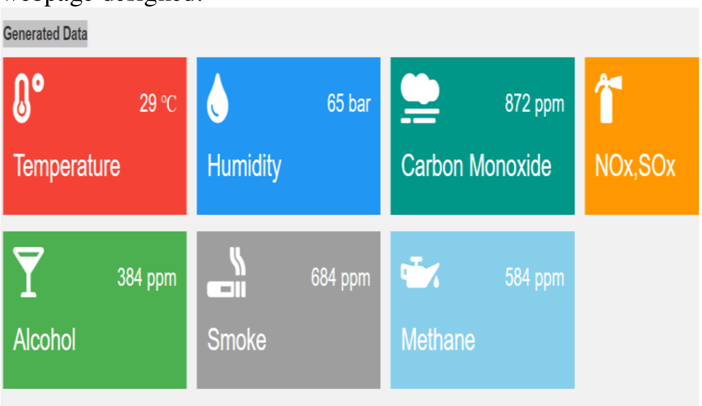


Fig -5. Output Data

Further, the cloud platform used to save and store the data for the web-page is GitHub the data stored in the cloud will be sent over to the web-page for the user to view.

The following snapshots show how the final web-page is presented.

The features of the web-page are highlighted as:

1) A display of the generated data showing the levels of different pollutants, humidity and temperature in standard units
2) Standard AQI levels of the pollutants along-with it's meaning
3) A table showing the concentration of pollutants in different environments.
4) A graphical representation of the pollutant levels with the implementation of a pi-graph.

Fig -6. Web UI

The data-set is an excel sheet with a collection of real-time data collected in different times under different conditions to show the change in the levels of sensors under the changing environment.

1	05/23/2021-14:39:01	Temperature	Humidity	Alcohol	NOx SOx	Carbon Monoxide	Smoke	Methane
2	05/23/2021-14:39:02	29	66	383	843	887	691	589
3	05/23/2021-14:39:05	29	66	383	844	886	689	589
4	05/23/2021-14:39:08	29	66	383	844	886	688	588
5	05/23/2021-14:39:11	29	66	396	843	886	689	589
6	05/23/2021-14:39:16	29	66	390	843	886	689	589
7	05/23/2021-14:39:19	29	66	390	843	886	689	589
8	05/23/2021-14:39:22	29	66	389	843	886	689	589
9	05/23/2021-14:39:25	29	66	388	843	886	689	589
10	05/23/2021-14:39:28	29	66	389	843	886	689	589
11	05/23/2021-14:39:31	29	66	393	843	887	689	589
12	05/23/2021-14:39:34	29	66	392	843	887	692	589
13	05/23/2021-14:39:37	29	66	392	844	886	689	589
14	05/23/2021-14:39:40	29	66	393	841	886	689	590
15	05/23/2021-14:39:43	29	66	392	843	886	689	589
16	05/23/2021-14:39:46	29	66	388	844	886	689	590
17	05/23/2021-14:39:49	29	66	387	843	886	689	589
18	05/23/2021-14:39:52	29	66	386	843	886	690	589

Fig -7. Data Storage

8. CONCLUSION

The Internet of Things based Data Acquisition System will be equipped with different possible ways of data transmission which uses Raspberry Pi as processor. The data transmission will take place with WI-FI and Bluetooth for medium range and GSM and Ethernet will be used for long range transferring capabilities.The proposed system is a generalized system which can be used in various fields such as medical, industrial, robotics etc. with all connectivity options and different range options. The system is small in size and portable. All the options of data transfer will be supported with an Android application to observe and analyse the data and control accordingly.

It is recommended to test other gas levels which pertaining to air quality and pollution such as ozone, carbon dioxide and other gases. It is also suggested to focus future researches on external casing designs to improve the portability and placements of the device.

9. FUTURESCOPE

The future scope that can be seen would be to improve the set-up in several areas as follows,

1)To improve the data interpretation and analytics. This stream proves correlation between so many factors from basic Data Acquisition to AI and Machine Learning .

2)AI or data mining algorithms can also be implemented in further developments to obtain the data by studying data sets and training their modules with the implementation of Machine Learning.

3)Another important area of improvement is 'the IoT edge devices' replacement with IP67 outdoor resistant devices for all three layers i.e. Sensors, IoT nodes and gateways.

4)Applying this project in different medium cities in cooperation with the local authorities to see the traffic impact simulations and to integrate with other smart city programs.

REFERENCES

[1] low-power real-time air quality monitoring system using LPWAN based on LoRa: Published in: Solid-State and Integrated Circuit Technology (ICSICT), 2016 13th IEEE International Conference Authors: Sujuan Liu, Chuyu Xia, Zhenzhen Zhao (College of Electronic Information and Control Engineering, Beijing University of Technology, 100124, China) .

[2] A WIFI-enabled indoor air quality monitoring and control system: Published in: Control & Automation (ICCA), 2017 13th IEEE International Conference Authors: Xiaoke Yang, Lingyu Yang, Jing Zhang (School of Automation Science and Electrical Engineering, Beihang University, Beijing, 100191, China) .

[3] An IoT Based Low-Cost Air Pollution Monitoring System Gagan Parmar, Sagar Lakhani, Manju K. Chattopadhyay School of Electronics Devi Ahilya University Indore, India gaganspl12@gmail.com.

[4] Edge Computing Based IoT Architecture for Low-Cost Air Pollution Monitoring Systems: A Comprehensive System Analysis, Design Considerations & Development by Zeba Idrees, Zhuo Zou ID and Lirong Zheng Received: 5 July 2018; Accepted: 23 August 2018; Published: 10 September 2018.

[5] International Journal of Grid and High-Performance Computing Volume 11 • Issue 4 • October-December 2019Copyright © 2019, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited. IoT-Based Real Time Air Pollution Monitoring System.

[6] Development of an IoT-Based Indoor Air Quality Monitoring Platform JunHo Jo, ByungWan Jo, JungHoon Kim, SungJun Kim and WoonYong Han. Department of Civil and Environmental Engineering, Hanyang University, 04763 Seoul, Republic of Korea, Smart IS, 22101 Incheon, Republic of Korea.

[7] IoT enabled proactive indoor air quality monitoring system for sustainable health management: - Published in: Computing and Communications Technologies (ICCCT), 2017 2nd International Conference Authors: M.F.M Firdhous, B.H Sudantha, P.M Karunaratne (Dept. of Information Technology, University of Moratuwa, Sri Lanka).

[8] A wireless system for indoor air quality monitoring: Published in: Industrial Electronics Society, IECON 2016 - 42nd Annual Conference of the IEEE Authors: R du Plessis, A Kumar, GP Hancke (Department of Electrical, Electronic and Computer Engineering, University of Pretoria, South Africa).

[9] An efficient cloud-based management of IoT devices for air quality monitoring: Published in: Research and Technologies for Society and Industry Leveraging a better tomorrow (RTSI), 2016 IEEE 2nd International Forum. Authors: Giovanni B Fioccola, Raffaele Sommese, Imma Tufano, Roberto Canonico, Giorgio Ventre.

[10] An embedded system model for air quality monitoring: Published in: Computing for Sustainable Global Development (INDIACom), 2016 3rd International Conference Authors: Sneha Jangid, Sandeep Sharma (School of ICT, Gautam Buddha University, Greater Noida, India) .

[11] Air quality sensing and monitoring 11 A real-time ambient air quality monitoring wireless sensor network for schools in smart cities: Published in: Smart Cities Conference (ISC2), 2015 IEEE First International Authors: H. Ali, J. K. Soe, Steven. R. Wel (School of Electrical Engineering & Computer Science, The University of Newcastle, Callaghan, NSW 2308, Australia).

[12] A smart sensor system for air quality monitoring and massive data collection: Published in: Information and Communication Technology Convergence (ICTC), 2015 International Conference Authors: Yonggao Yang, Lin Li (Department of Computer Science, Prairie View A&M University, Prairie View, TX 77446, U.S.A).

[13] Indoor air quality monitoring using wireless sensor network. In Sensing Technology (ICST), 2012 Sixth International Conference on (pp. 422-427). IEEE Authors: Bhattacharya, S., Sridevi, S., & Pitchiah, R. (2012, December).

[14] P-sense: A participatory sensing system for air pollution monitoring and control. Published in: In Pervasive Computing and Communications Workshops (PERCOM Workshops), 2011 IEEE International Conference on (pp. 344-347). IEEE Author: Mendez, D., Perez, A. J., Labrador, M., & Marron, J. J.

[15] Urban air pollution monitoring system with forecasting models: IEEE journal, 2016.Proposed by- E. Rezk, A. Kadri, K. B. Shaban.

[16] Wireless Sensor Network Based Pollution Monitoring System in Metropolitan Cities Proposed by: Shwetal Raipure. Deepak Mehetre.

[17] Distributed System as Internet of Things for a new low-cost, air pollution Wireless Monitoring on Real Time By: Walter Fuertes, Diego Carrera, César Villacís, Theofilos Toulkeridis, Fernando Galárraga, Edgar Torres, and Hernán Aules.

[18] Design and Analysis of IoT based Air Quality Monitoring System Ajitesh Kumar, Mona Kumari and Harsh Gupta. **Published in:** 2020 International Conference on Power Electronics & IoT Applications in Renewable Energy and its Control (PARC), 28-29 February 2020.

[19] An IoT Based Air Pollution Monitoring System for Smart Cities, ICSETS 2019, Harsh Gupta, Dhananjay Bhardwaj, Himanshu Agarwal, Vinay Anand Tikkiwal, Arun Kumar.

[20] IoT Solution for Smart Cities' Pollution Monitoring and the Security Challenges Cristian Toma , Andrei Alexandru, Marius Popa and Alin Zamfiroiu Department of Economic Informatics and Cybernetics, Faculty of Cybernetics, Statistics and Economic Informatics, Bucharest University of Economic Studies, 010552 Bucharest, Romania. Correspondence: cristian.toma@ie.ase.ro (C.T.); marius.popa@ie.ase.ro (M.P.) Received: 30 May 2019; Accepted: 30 July 2019; Published: 2 August 2019.

[21] Development of a Low-Cost Air Quality Data Acquisition IoT-based System using Arduino Leonardo Louis Anton A. Cruza , Maria Teresa T. Griño a , Thea Marie V. Tungola , Joel T. Bautistaa a Philippine Science High School - Central Luzon Campus, Pampanga, 2009, Philippines Received: 17 November 2018; Accepted: 15 February 2019; Published: 08 May 2019.

[22] Low cost air pollution monitoring systems: A review of protocols and enabling technologies Zeba Idreesa,b, , Lirong Zheng a School of Information Science and Engineering, Fudan University, China b University of Engineering and Technology Lahore, Pakistan.

[23] An IoT Based Low-Cost Air Pollution Monitoring System Gagan Parmar, Sagar Lakhani, Manju K. Chattopadhyay Proceeding International conference on Recent Innovations is Signal Processing and Embedded Systems (RISE -2017) 27-29 October,2017.

[24] Edge Computing Based IoT Architecture for Low-Cost Air Pollution Monitoring Systems: A Comprehensive System Analysis, Design Considerations & Development Zeba Idrees, Zhuo Zou ID and Lirong Zheng * School of Information Science and Engineering, Fudan University, Shanghai 200433, China; izeba17@fudan.edu.cn * Correspondence: zhuo@fudan.edu.cn (Z.Z.); lrzheng@fudan.edu.cn (L.Z.) Received: 5 July 2018; Accepted: 23 August 2018; Published: 10 September 2018.

[25] Air pollution monitoring and prediction using IoT, Proceedings of the 2nd International Conference on Inventive Communication and Computational Technologies (ICICCT 2018) IEEE Xplore Compliant - Part Number: CFP18BAC-ART; ISBN:978-1-5386-1974-2, Temesegan Walelign Ayele Department of Information Technology PIET, Parul University, Rutvik Mehta Asst. Prof, Department of Information Technology PIET, Parul University Vadodara, India.