Mini-Project(2020-21)

IoT Based Air Quality Monitoring System

Mid-Term Report

Institute of Engineering and Technology

Submitted by-
Abhishank Pandey
(02)

Supervised by-
Mr. Amir Khan
Technical Trainer
Department of Computer Engineering & Applications

Content

Abstract

As the world's population is becoming increasingly urban, the cities are under pressure to remain livable. In recent years, the air quality of the cities has become one of the major cause of concern around the world.Thus,in this project it is necessary to constantly monitor the air quality index of a city to make it smart and livable. In this paper, we propose and develop an IoT based Air Quality Monitoring System for Smart Cities or even other small cities and towns. The real-time data of the air quality is accessed through the smart devices and analyzed to measure the impact on city dwellers. The smart devices are capable of measuring the Temperature, Humidity, Carbon Monoxide, LPG, Smoke and other hazardous particulate matters like PM2.5 and PM10 levels in the atmosphere. The gathered data is accessed globally through an Android Application or any cloud data server of IoT.

IoT technology has evolved due to the convergence of multiple technologies real-time analytics,machine learning,commodity sensors, and embedded systems.Traditional fields of embedded systems, wireless sensor networks,control systems,automation(including home and building automation)and others all contribute to enabling 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 that support one or more common ecosystems, and can be controlled via devices associated with that ecosystem, such as smartphones and smart speakers.

There are a number of serious concerns about dangers in the growth of IoT, 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 standards.

1.Introduction(1.1)

To develop this IoT project first of all we need to know the basics of sensors,some basic knowledge of C and Python programming along with very little knowledge of embedded c programming(not enough) that is it should be sufficient.To build any type of IoT project we need these basic components first of all such as arduino uno development board, sensors,a well high speed internet connections, a cloud server through which we will be sending ,uploading , transferring and reading the data with it, well established database connectivity,etc. In this IoT project of " Air Quality Monitoring System" we need components such as: a arduino uno development board, node mcu(esp8266 wifi module), sensors such as MQ6 and MQ135 gas sensors,DHT11 sensor for measuring the temperature and humidity,bread board,jumper wires,16x2 LCD display for iot project ,a dc 5V power supply if needed for node mcu .

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 a 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. LPG sensor is added in this system which is used mostly in houses. The system will show temperature and humidity. The system can be installed anywhere but mostly in industries and houses where gases are mostly to be found and gives an alert message when the system crosses threshold limit.

1.2 Analysis Of Project

IOT Based Air Pollution Monitoring System monitors the Air Quality over a webserver using internet and will trigger an alarm when the air quality goes down beyond a certain level, means when there are amount of harmful gases present in the air like CO2, smoke, alcohol, benzene, NH3, NOx and LPG.

The system will show the air quality in PPM on the LCD and as well as on webpage so that it can be monitored very easily. Temperature and Humidity is detected and monitored in the system.

LPG gas is detected using MQ6 sensor and MQ135 sensor is used for monitoring Air Quality as it detects most harmful gases and can measure their amount accurately. In this IOT project, it can monitor the pollution level from anywhere using your computer or mobile. This system can be installed anywhere and can also trigger some device when pollution goes beyond some level, like we can send alert SMS to the user.

1.3 Hardware And Software Requirements

Hardware Requirements:

1) MQ135 Gas sensor

2) Arduino Uno

3) Wi-Fi module ESP8266

4) 16x2 LCD display

5) Breadboard

6) MQ 6 LPG gas sensor

7) Buzzer

8) DHT-11 (both a temperaure and humidity sensor)

Software requirements:

1) Arduino 1.6.13 Software

2)Embedded C Language

3) ThingSpeak cloud data server for displaying output to the lcd display screen.

1.4 Components Description

1) Arduino UNO:-

Arduino Uno is a microcontroller board based on the ATmega328P It has 14 digital input/output pins 6 analog inputs, a 16 MHz quartz crystal, a USB Connection, power jack, an ICSP header and a reset button.

2) MQ135 sensor:-

The MQ135 sensor can sense NH3, NOx, alcohol, Benzene, smoke, CO2 and some other gases.It gives the output in form of voltage levels.

3) WIFI Module (ESP8266/Node MCU):-

The ESP8266 is a low-cost Wi-Fi chip with full TCP/IP stack and MCU (microcontroller unit) capability. It runs on 3.3V and gives our system access to Wi-Fi or internet.

4) Buzzer:-

A Buzzer or beeper is an audio signaling device.Whenever the air pollution goes above the threshold level the Buzzer starts beeping indicating Danger

5) LCD (Liquid Crystal Display):-

This is a basic (16x2) 16 character by 2 line display. Black text on Green background. It is used to indicate the Air and Humidity in PPM.

6) LPG Sensor:-

MQ-6 sensor is a simple-to-use liquefied petroleum gas (LPG) sensor, suitable for sensing LPG (composed of mostly propane and butane) concentrations in the air. The MQ-6 can detect gas concentrations anywhere from 200 to 10000ppm.

7) DHT-11 Temperature & Humidity Sensor:

DHT11 is a low-cost digital sensor for sensing temperature and humidity. This sensor can be easily interfaced with any micro-controller such as Arduino, Raspberry Pi etc… to measure humidity and temperature instantaneously.

2 Objective

Main objective or the motive of this project is to prevent the harmful effects of pollutants present in the air so that a healthy surrounding can be maintained by data analysis of stored data in IoT cloud.In today's world many pollution monitoring systems are designed by considering the different environmental parameters.The other objective of this project would be to analyze the data on any IoT cloud data server so that in future if needed we can improve the quality of air by minimizing the use of fossil fuels and other resources and can give more preference to renewable source of energy.

3 Implementation of Project

Proposed concept

a) Design Ideas: Air quality sensors are installed in the targeted area on top of buildings, industrial areas, traffic and residential areas. These sensors are connected to a microcontroller to control the sensors network. The data collected by the microcontroller is transmitted to the cloud for analysis. The analyzed data is shared to the public through a smartphone app.

b) How it works?: Wireless sensors placed at strategic locations sense the level of dust particles, carbon dioxide, carbon monoxide, nitrogen dioxide and sulfur dioxide in the air. This information is transmitted to a gateway which forwards it to a cloud database by means of cellular or WiFi communication. In the cloud, the data are analyzed to provide information on the air quality. The information on the air quality is shared through a smartphone app. This allows the relevant authority to take remedial action and the community to take precautionary measures

c) What makes it innovative?: This project proposes an idea to install monitoring applications on smartphones. It is innovative because it provides easy access to the public to monitor real time air quality in their area. It uses low cost and readily available devices such as a dust sensor, carbon monoxide gas sensor, carbon dioxide gas sensor, and nitrogen dioxide gas sensor. For controlling these sensors, microcontrollers are used and the microcontrollers also act as transmitter to transmit the data to the cloud database. The information on air quality can be accessed through a smartphone app in real time.

d) How would it be produced?: The IoT based air pollution monitoring system can be produced by using sensors and microcontrollers available in the market. The microcontrollers are programmed to take the sensors as input and transmit the data to the cloud. An algorithm is developed to analyse the data and send it to the smartphones app or computer or laptops. Smartphone app will be developed for user to access the air quality information in real-time.

e) Where would it be applied?: i. Sensors will be installed on top of buildings, industrial areas, traffic and residential areas.

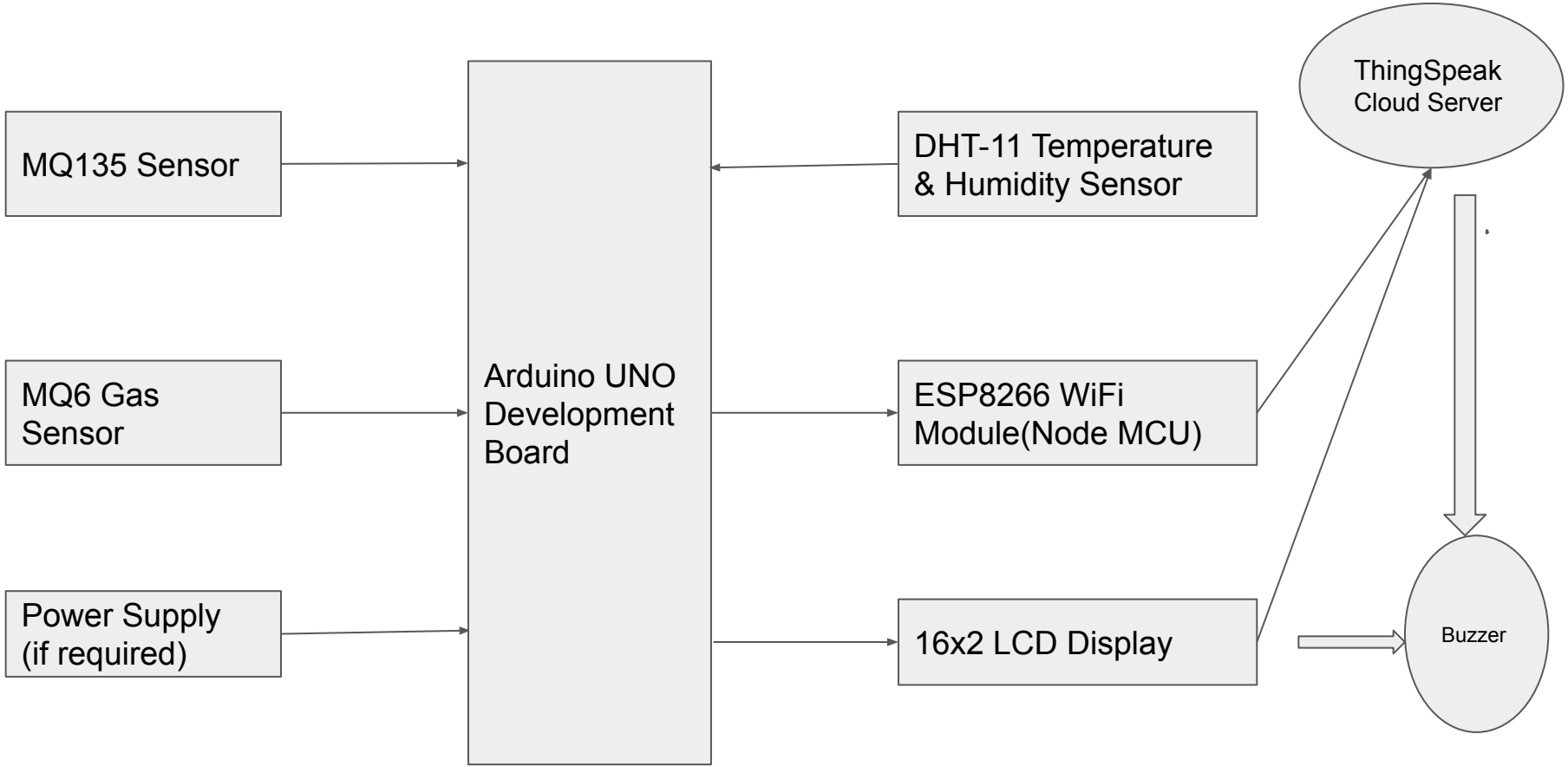
ii. Smartphone app for monitoring is installed in individual smartphone.

Potential impact around the globe?

There are many cities around the world facing air quality issues. The contaminated air results in death every year and decline in health conditions as people are exposed to unhealthy air quality. Awareness of the contaminated air enables the community to take precautionary steps. With this project the community can enjoy cleaner air and improved health conditions

.

Block Diagram Of Project



4. Area(Scope) of IoT in Computer Science

Internet of Things (IoT) has been identified as one of the emerging technologies in IT. Future of IoT is very bright as this is feeding and empowering Data Science and Artificial Intelligence in a big way. Data from IoT network enables us to have better tracking, monitoring, prediction, management and control of various systems in different industries.

Career Opportunities

- Data Analytics
- Network and Structure
- Cell and UI Development Artificial Intelligence
- Infonomics and Data Broking
- Shift from Intelligent Edge to Intelligent Mesh
- IoT Governance
- Sensor Innovation
- Social, Legal and Ethical IoT
- Innovation on the Chip

Explanation Content

First of all we will connect the **ESP8266 with the Arduino**. ESP8266 runs on 3.3V and if you will give it 5V from the Arduino then it won't work properly and it may get damage. Connect the VCC and the CH_PD to the 3.3V pin of Arduino. The RX pin of ESP8266 works on 3.3V and it will not communicate with the Arduino when we will connect it directly to the Arduino. So, we will have to make a voltage divider for it which will convert the 5V into 3.3V. This can be done by connecting three resistors in series like we did in the circuit. Connect the TX pin of the ESP8266 to the pin 10 of the Arduino and the RX pin of the esp8266 to the pin 9 of Arduino through the resistors.

ESP8266 Wi-Fi module gives your projects **access to Wi-Fi or internet**. It is a very cheap device and make your projects very powerful. It can communicate with any microcontroller and it is the most leading devices in the IoT platform.

Then we will connect the **MQ135,MQ6 sensor with the Arduino**. Connect the VCC and the ground pin of the sensor to the 5V and ground of the Arduino and the Analog pin of sensor to the A0 of the Arduino.

Connect a buzzer to the pin 8 of the Arduino which will start to beep when the condition becomes true.

In last, we will connect LCD with the Arduino.The connections of the LCD are as follows

- Connect pin 4 (RS) to the pin 12 of the Arduino.
- Connect pin 5 (Read/Write) to the ground of Arduino. This pin is not often used so we will connect it to the ground.
- Connect pin 6 (E) to the pin 11 of the Arduino. The RS and E pin are the control pins which are used to send data and characters.
- The following four pins are data pins which are used to communicate with the Arduino.

Connect pin 11 (D4) to pin 5 of Arduino.

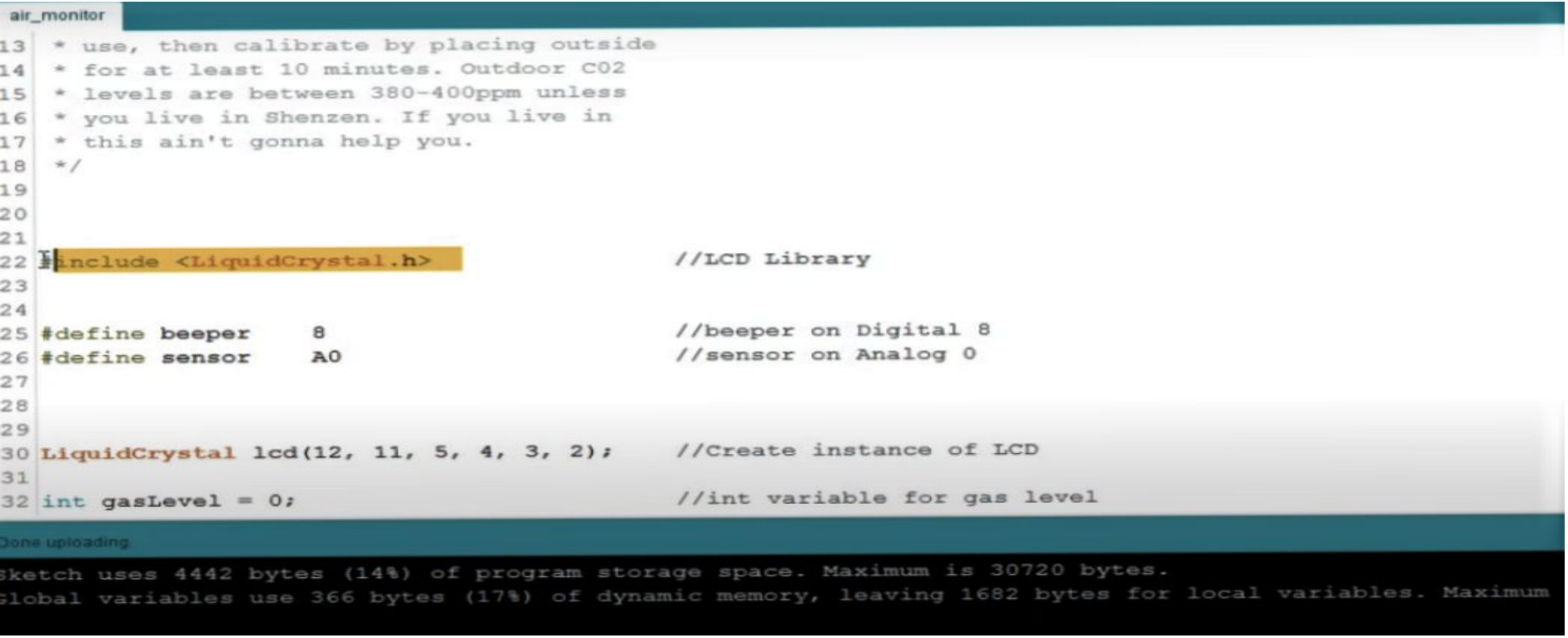
Connect pin 12 (D5) to pin 4 of Arduino.

Connect pin 13 (D6) to pin 3 of Arduino.

Connect pin 14 (D7) to pin 2 of Arduino.

- Connect pin 15 to the VCC through the 220 ohm resistor. The resistor will be used to set the back light brightness. Larger values will make the back light much more darker.
- Connect pin 16 to the Ground.

5. Screenshots



air_monitor

```
25 #define beeper     8                    //beeper on Digital 8
26 #define sensor     A0                   //sensor on Analog 0
27
28
29
30 LiquidCrystal lcd(12, 11, 5, 4, 3, 2);     //Create instance of LCD
31
32 int gasLevel = 0;                         //int variable for gas level
33 String quality ="";
34
35 void setup() {
36   Serial.begin(9600);                     //start serial comms
37   pinMode(beeper,OUTPUT);                 //set beeper for output
38   pinMode(sensor,INPUT);                  //set sensor for input
39
40
41   lcd.begin(16, 2);                       //initial LCD setup
42   lcd.setCursor (0,0);                    //splash screen and warmup
43   lcd.print("                ");
44   lcd.setCursor (0,1);
```

Done uploading.

```
Sketch uses 4442 bytes (14%) of program storage space. Maximum is 30720 bytes.
Global variables use 366 bytes (17%) of dynamic memory, leaving 1682 bytes for local variables. Maxi
```

air_monitor

```
40
41    lcd.begin(16, 2);                    //initial LCD setup
42    lcd.setCursor (0,0);                 //splash screen and warmup
43    lcd.print("                ");
44    lcd.setCursor (0,1);
45    lcd.print("                ");
46    lcd.setCursor (0,0);
47    lcd.print("   Air Sensor   ");
48    lcd.setCursor (0,1);
49    lcd.print("   Warming Up   ");
50    delay(2000);                         //set for at least 2 minutes
51
52    lcd.setCursor (0,0);                 //clear screen
53    lcd.print("                ");
54    lcd.setCursor (0,1);
55    lcd.print("                ");
56
57 }
58
59 void loop() {
```

Done uploading.

```
Sketch uses 4442 bytes (14%) of program storage space. Maximum is 30720 bytes.
Global variables use 366 bytes (17%) of dynamic memory, leaving 1682 bytes for local variables. Maximum
```

air_monitor

```
49   lcd.print("   Warming Up    ");
50   delay(2000);                              //set for at least 2 minutes
51
52   lcd.setCursor (0,0);                      //clear screen
53   lcd.print("                ");
54   lcd.setCursor (0,1);
55   lcd.print("                ");
56
57 }
58
59 void loop() {
60   gasLevel = analogRead(sensor);
61
62   if(gasLevel<175){
63     quality = "GOOD!           ";
64   }
65   else if (gasLevel >175 && gasLevel<225){
66     quality = "Did you fart?   ";
67   }
68   else if (gasLevel >225 && gasLevel<300){
```

Done uploading.

```
Sketch uses 4442 bytes (14%) of program storage space. Maximum is 30720 bytes.
Global variables use 366 bytes (17%) of dynamic memory, leaving 1682 bytes for local variables. Maximum is 2048 bytes.
```

air_monitor

```
58
59 void loop() {
60   gasLevel = analogRead(sensor);
61
62   if(gasLevel<175){
63     quality = "GOOD!             ";
64   }
65   else if (gasLevel >175 && gasLevel<225){
66     quality = "Did you fart?     ";
67   }
68   else if (gasLevel >225 && gasLevel<300){
69     quality = "Something dead?";
70   }
71   else if (gasLevel >300){
72     quality = "Pretty fk'n bad";
73   }
74 lcd.setCursor (0,0);
75 lcd.print("Air Quality is:");
76 lcd.setCursor(0,1);
77 lcd.print(quality);
```

Done uploading.

```
Sketch uses 4442 bytes (14%) of program storage space. Maximum is 30720 bytes.
Global variables use 366 bytes (17%) of dynamic memory, leaving 1682 bytes for local variables. Maximum is 2048 bytes.
```

```
Air Quality: 97.21  PPM

Temperature =  85.8
Humidity =  69.9
Pressure =  944.5
Pressure Inch = 27.89
Dew Point =  75.0
.................................................
PM1.0: 524  ug/m3
PM2.5: 1053  ug/m3
PM10 : 1373  ug/m3

Air Quality: 91.38  PPM

Temperature =  85.8
Humidity =  69.8
Pressure =  944.5
Pressure Inch = 27.89
```

Air Pollution/Quality Monitoring

thingspeak.com/channels/901352/private_show

ThingSpeak™ Channels Apps Support Commercial Use How to Buy Account Sign Out

Field 1 Chart

Field 2 Chart

Field 3 Chart

Field 4 Chart

Air Pollution/Quality Monitoring

thingspeak.com/channels/901352/private_show

ThingSpeak™ Channels Apps Support Commercial Use How to Buy Account Sign Out

Field 1 Chart

Field 2 Chart

Field 3 Chart

Field 4 Chart

6. References

[1]https://store.arduino.cc/arduino-uno-rev3

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

[3] www.iotwebplanet.com

[4] content/uploads/2008/04/5v-r

And many more but not knowing now.