Design and Analysis of IoT Based Air Quality Monitoring System

Supervised By:

Md. Mostafa Amir Faisal

Assistant Professor,

Dept. Of ETE, IIUC

By,

Sazzad Ur Rahman

ID: T161033

MD. Rabiul Hassan

ID: T173017

Akib Ahmed Chowdhury

ID: T173022

Department of Electronics and Telecommunication Engineering

International Islamic university Chittagong

Kumira, Sitakunda Chittagong- 4314

January, 2022

Design and Analysis of IoT Based Air Quality Monitoring System.

By,

Sazzad Ur Rahman

MD. Rabiul Hassan

Akib Ahmed Chowdhury

A project submitted as partial fulfilment of the requirement for the degree of

BACHELOR OF SCIENCE IN ELECTRONICS AND TELECOMMUNICATION ENGINEERING

Department of Electronic & Telecommunication Engineering

INTERNATIONAL ISLAMIC UNIVERSITY CHITTAGONG

January 2022

CERTIFICATE OF APPROVAL

Sazzad Ur Rahman, bearing Matric ID. T161033, Md. Rabiul Hassan, bearing Matric ID. T173017, MD. Akib Ahmed Chowdhury – T173022, presented a project titled " Design and Analysis of IoT Based Air Quality Monitoring System " to the Department of Electronics & Telecommunication Engineering (ETE), International Islamic University Chittagong, in the Autumn 2020 session (IIUC). It was approved as satisfactory and accepted for the review conducted on January, 2022, in partial fulfillment of the criteria for the degree of Bachelor of Science in Engineering.

Signature.

____

Supervisor

Engr. Md. Mostafa Amir Faisal.

Assistant Professor,

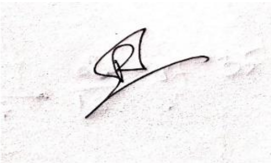
Department of Electronic and Telecommunication Engineering

International Islamic University Chittagong.

DECLARATION

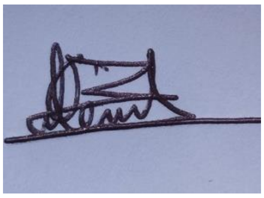
This work has been completed by us, and no part of the work contained in this project has been submitted elsewhere for the granting of a degree or diploma.

25/11/21



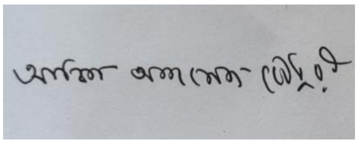
Sazzad Ur Rahman

25/11/21



Md. Rabiul Hassan

25/11/21



Akib Ahmed Chowdhury

ACKNOWLEDGEMENT

In the name of the Most Generous and Merciful Almighty. First and foremost, we give thanks to the Almighty for giving us the ability to think. We appreciate our family's affection, spiritual support, and financial assistance. We found this to be incredibly valuable. We’d like to honor our honorable supervisor, Engr. Md. Mostafa Amir Faisal, Asst. Professor, Dept. of ETE, IIUC, for providing us with assistance with project planning and inspections. His invaluable assistance in the form of constructive remarks and suggestions has contributed to the project's success throughout the experiment and project work. Finally, we are grateful to our instructors, friends, students, seniors, and lab assistants for their unwavering support and aid in every manner. We were blessed to have such a kind individual on our side.

Authors.

ABSTRACT

Air pollution is one of the most serious environmental issues, with numerous health consequences for humans, ecosystems, and the climate. Vehicles are, without a doubt, the primary source of air pollution in all urban areas, with industry serving as a secondary source of pollution. The widespread use of automobiles has resulted in a substantial rise in the number of pollutants in the environment. This is the primary source of environmental degradation, which has a negative impact on human health. Other respiratory problems, such as asthma attacks and skin rashes, have also developed as a result. People's apathy has been exacerbated by the fact that the contaminants that contaminate the air are invisible. As a result, today's most important criteria are general acceptance. The Internet of Things (IoT) has progressed to the point that it can now monitor and even manage the number of harmful compounds in air particles. The goal is to develop a system that can be applied to detect toxic particles in the air. The detection results can be check over a webserver as well as a smartphone application. The purpose of this research is to reliability analysis monitoring and computation of indoor air quality using a microbudget system. Our project work is good for ordinary users since it allows them to quickly determine the level of indoor air quality and to track it in instantaneously.

TABLE OF CONTENTS

List of Figures

List of Tables

List of Abbreviation

LPG: Liquefied Petroleum Gas

SDG: Sustainable Development Goals

UN: United Nations

AQI: Air Quality Index

GUI: Graphical User Interface

GPIO: General-Purpose input/output

PCB: Printed circuit board

HTML: Hypertext Markup Language

PHP: Personal Home Page

API: Application Programming Interface

LDAP: Lightweight Directory Access Protocol

IMAP: Internet Message Access Protocol

SNMP: Simple Network Management Protocol

NNTP: Network News Transfer Protocol

HTTP: Hypertext Transfer Protocol

PPM: Parts Per Million

CHAPTER 1
INTRODUCTION

Air pollution is a worldwide problem. The amount of carbon monoxide, led, nitrogen oxides, ground-level ozone, particle pollution, sulfur oxides, and other pollutants in the air influences whether it is healthy or not. Pollution emissions cannot be prevented in the actual world. However, if the pace of decline exceeds the WHO's (World Health Organization) regulation limits, we should interfere, and diminishing the density of pollutants in an indoor environment is simple. According to the WHO, pollution of the air is responsible for roughly 5 million premature deaths worldwide, including 91 percent of population lost lives in areas where natural quality exceeds WHO standards. It is essential to monitor the air, and it has to be IoT-based in order to preserve record of the air quality of a specific location from anywhere. An Internet of Things (IoT) links and transmits data among devices over the internet. This will make a significant difference. It will make our data transportable, empowering us all to take immediate action. It will exhibit air quality on an LCD screen via tracking the number of harmful objects, and the addition of some sensors will keep us updated through indicating humidity, temperature, pressure, and dust particles, among several other things. The data collected will be analyzed utilizing standard values through the device. The information will be kept in a cloud storage interface, from which a user-friendly smartphone app will generate real-time air quality data.

Air quality refers to how safe it is to breathe for people, plants, and animals. Clear, clean air with low concentrations of particulate and chemical contaminants characterizes an excellent air quality environment. Both human health and the ecosystem suffer when there is poor air quality, which is often cloudy and includes high quantities of pollutants. Pollutant concentrations were measured using the Air Quality Index (AQI) at a particular area. [1]

Three important gas sensors, which are mainly responsible for the bulk of indoor air pollution, are utilized in the system to discover the appropriate outcome of the comprehensive air quality. CO and Industrial Ammonia Gas have indeed been highlighted as the main sources to air quality degradation, but they are both utilized in the system. Essentially everyone should have an Android device and internet access nowadays, a server and an android application were added to the project to keep track of the statistics.

1.1 Motivation

Air pollution occurs when a chemical, physical, or biological agent affects the natural qualities of the atmosphere. It is difficult to live a long life amid chronic foul air. Air pollution, both outdoors and indoors, is a major cause of respiratory and other illnesses, as well as mortality. Every day, we require 13,000 litters of pure air. Every year, pollution-related diseases kill more people than automobile accidents. Sustainable consumption and production patterns; urgent action against climate change and its impacts; conservation of oceans, seas and marine resources; sustainable management of forests, desertification, halt and reverse land degradation and biodiversity loss, according to the Sustainable Development Goals (SDG). These are the main factors for choosing the subject.

1.2 Objective

Design and implement an Android application for something like an IoT-based air quality and meteorological monitoring system. The following measures will be achieved more by implemented project:

1. Combine innovative detection tools to achieve a low-cost, comprehensive monitoring and controlling system with advanced capabilities.

2. Monitoring and reflect indoor based places temperature and air pressure.

3. A dedicated web server and a simple application that can monitor air quality through the internet.

1.3 Project Thesis Outline

- Chapter 1, begins with an introduction, motivation for the project, and project objective.
- Chapter 2, The previous work in the issue of air pollution being reviewed.
- Chapter 3, The technique of the proposed project, but also the different components and their interpretations are discussed.
- Chapter 4, The findings and explanation are presented
- Chapter 5, summarizes the article by talking about the issues that were recognized including the project's ongoing growth.

CHAPTER 2
LITERATURE REVIEW

Air is the most fragile ingredient of the ecosystem, and it has been degraded by the pollutants thrown in and out on a continuous basis. This proposed methodology is a wireless sensor network that works exclusively to monitor pollution in an indoor place in order to measure the amount quality in the air. It's a minimal monitoring system with modest but functional sensors.

Some preceding works, such as Air Quality Monitoring System Based on ISO/IEC/IEEE 21451 Standards [2], became introduced in 2016. It effectively evaluated the rate of toxin gas, which are responsible for air pollution. To comprehend the toxic gasses and their influence, Air quality monitoring using Raspberry Pi based on IoT [3] was proposed. Design of Sensor Network for Urban Micro-Climate Monitoring [4] was proposed which can monitor air quality of urban environment. IoT based urban climate monitoring using Raspberry Pi [5] which is able to monitor the city environment using raspberry pi system.

Looking at recent data, we may assume that poor air quality has reached unprecedented levels. If it is not eradicated immediately, the entire region will be vulnerable to severe events in the future. There are various other types of pollution, such as water contamination, excessive noise, industrial effluents, and soil contamination, but future studies may show that air pollution is the most serious issue that needs to be addressed in order to breath safe air.

According to the World Health Organization: WHO, air pollution presents a serious threat to human health and the environment, from pollutants hanging over cities to indoor smoking. Each year, the combined effects of environmental (outside) and domestic air pollution result in approximately 7 million premature deaths, the majority of which are caused by increased mortality from stroke, heart disease, chronic obstructive pulmonary disease, lung cancer, and acute respiratory infections. Over 80% of people who live in urban areas where air pollution is measured are exposed to levels of pollution that exceed the WHO's recommended limit of 10g/m3, with the largest exposures occurring in low- and middle-income countries.

Around the world, 9 out of 10 people breathe harmful pollutants. According to WHO, air pollution alone caused approximately 4.2 million deaths in 2016, while domestic air pollution from cooking with polluting fuels and technology caused approximately 3.8 million casualties during the same time period. So, the aim was to create a device that would alert individuals to the quantity of harmful air they were inhaling. This method incorporates past research studies that demonstrate how critical it is to work on such a topic. The basic objective was to create a gadget that is portable and easy to setup. The number of Android device and internet users has skyrocketed.

It would be a big help for the humanity if they always be alert about the environment quality they consume in every second. For detecting air pollution in indoor a detection device can be placed on the centre of an indoor environment. The primary target is to determine the number of harmful gases emitted in a closed environment. These sensors will capture real data in real time various gases (air and water) existing in the environment like- Carbon Monoxide, Carbon Dioxide, Ammonia, LPG, smoke, Nitrous Oxide etc. Our proposed monitoring system will allow us to monitor the air quality through a smartphone application conveniently and also from a dedicated webserver.

CHAPTER 3
METHODOLOGY

The system's proposed model is as follows. The system's block diagram depicts how the system will operate for a certain area. The sensors will be set up to collect air pollutants data and a baseline value will be established. The device will gather data and display the output based on the configured values.

3.1 Proposed Model block diagram

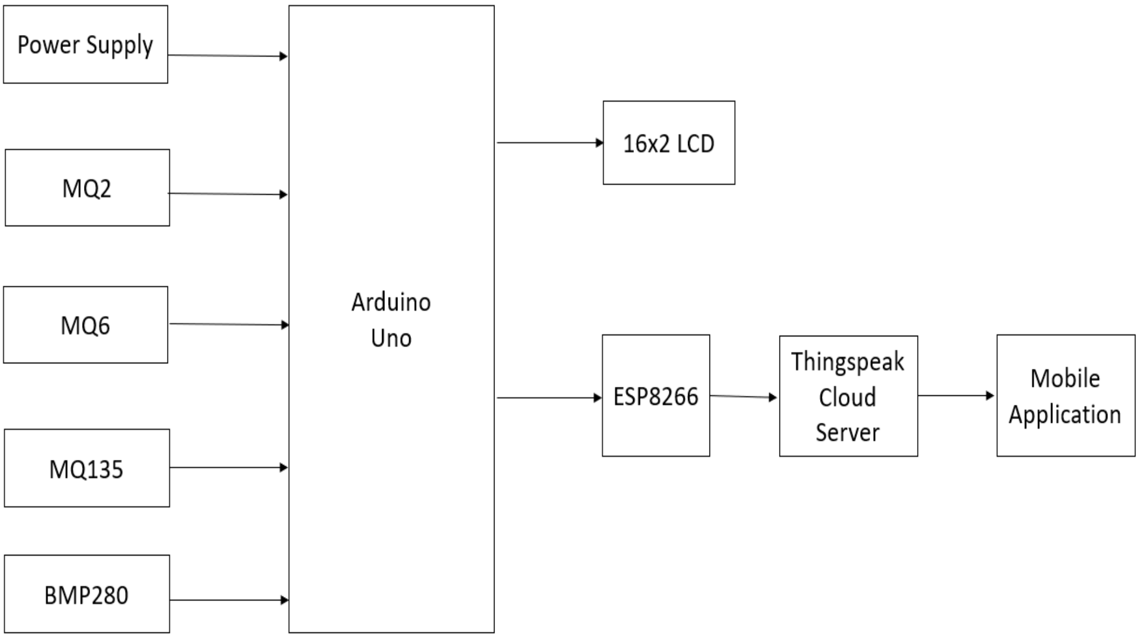


Fig. 3.1: Block diagram for proposed model of the system

Block Diagram 3.1 depicts the suggested system model. Three gas sensors and a temperature, pressure, altitude sensors are attached to the microcontroller (Arduino Uno) as input and a 16x2 LCD display is connected to show the project's output: air quality, carbon monoxide, LPG gas, temperature, air pressure and altitude. A Wi-Fi module named ESP8266 is also connected allowing us to transfer data on the Thing Speak web server. Then we collected the API key from the webserver which helped us send those sensors data to a smartphone application that we have made. We also have a huge amount of data saved in the webserver which can be used for future studies.

3.2 Flow Chart of Proposed System

The flow chart of the suggested model system is shown in Fig.3.2.

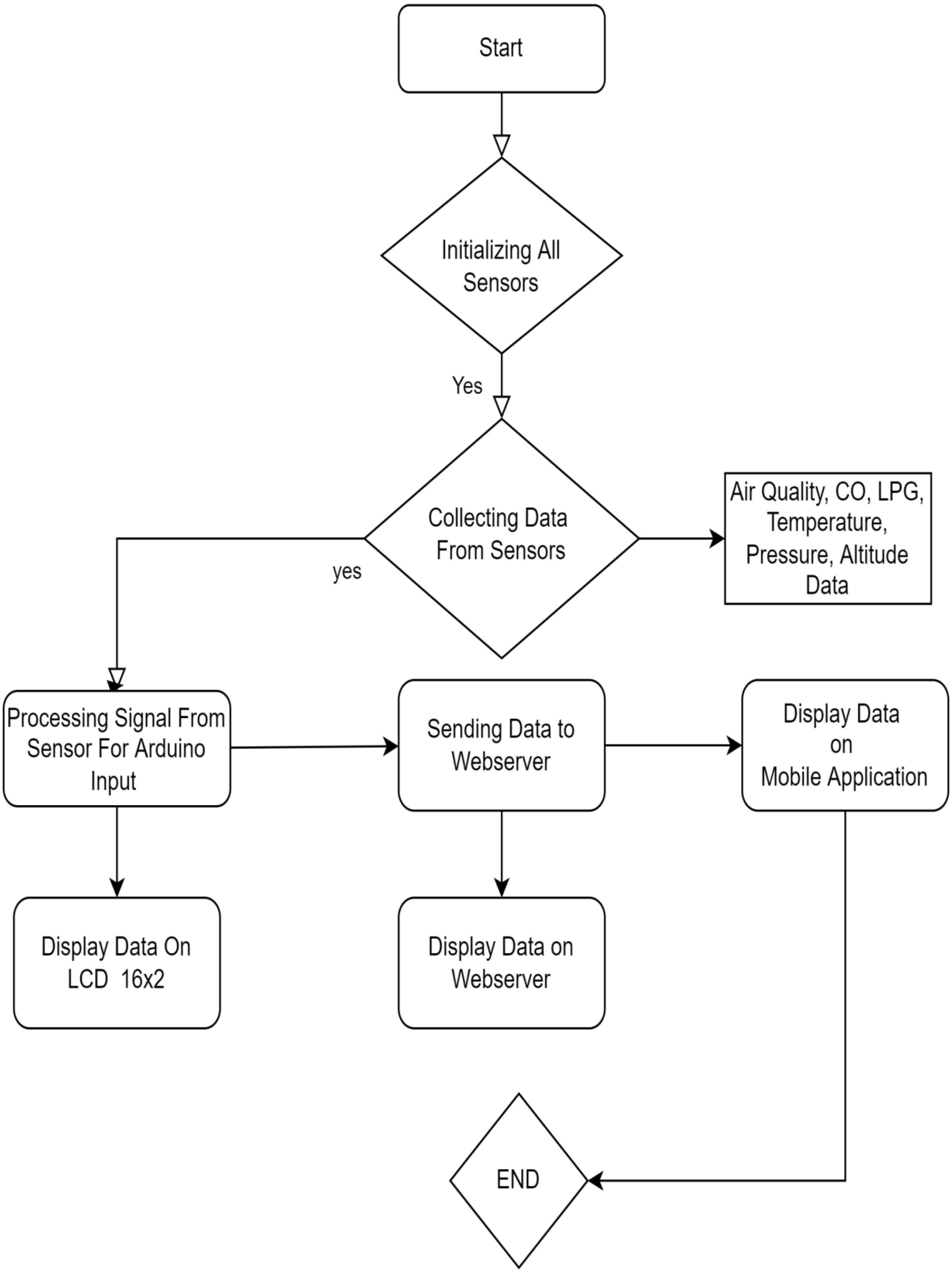


Fig. 3.2. Flow Chart for proposed model of the system

3.3 Circuit Diagram of Proposed System

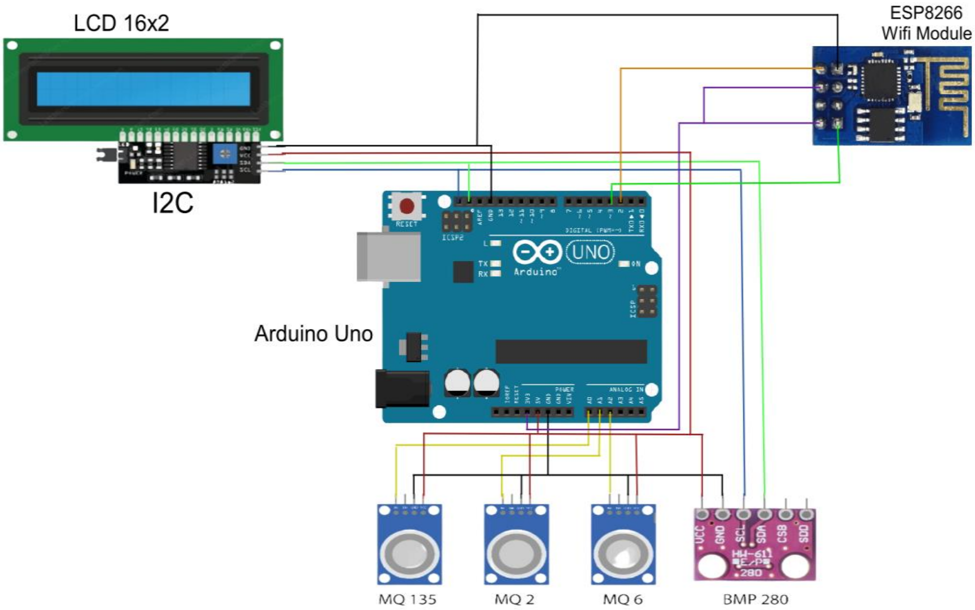


Fig. 3.3. Circuit diagram for proposed model of the system

3.4 Proposed Model in Real Life

The Proposed Model in Real Life is depicted in Fig.3.4.

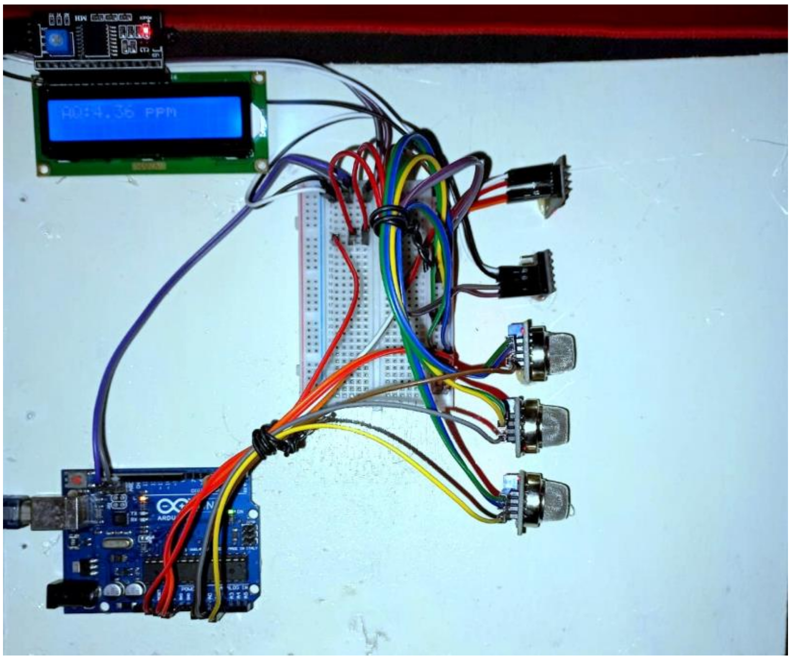


Fig. 3.4. Proposed Model in Real Life

3.5 Components of the Device

It is a gadget that gathers data from the environment in which our investigation will take place. The gadget is fitted with a variety of sensors that gather data by sensing its surroundings. The sensors capture analog data from the environment, which is then transformed to digital using a Raspberry Pi and sent to a server for storage. This gadget consists of the following components:

- Arduino Uno
- ESP8266 Wifi Module
- MQ2 CO Sensor
- MQ6 LGP Sensor
- MQ135 Air Quality Sensor
- BMP280 Absolute Barometric Pressure Sensor
- 16x2 LCD Display
- I2C Serial Adapter Module

3.5.1 Arduino Uno

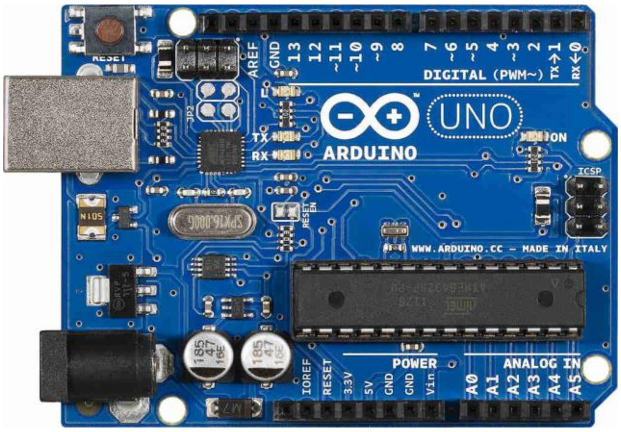


Fig. 3.5. Arduino Uno [8].

The ATmega328P microprocessor is used in the Arduino Uno microcontroller board. Figure 3.5 depicts Ar-duino Uno. It has 14 digital I/O pins, 6 analog I/O pins, a 16 MHz ceramic resonator (CSTCE16M0V53-R0), a USB connection, a power connector, an ICSP header, and a reset button. It includes everything you'll need to get started with the microcontroller; just connect it to a computer through USB or power it with an AC-to-DC converter or bat-tery. The following are the Arduino Uno Pin Functions:

Microcontroller	: ATmega328
Operating Voltage	: 5 V
Input Voltage (recommended)	: 7-12 V
Input Voltage (limit)	: 6-20 V

Digital I/O Pins	: 14 (PWM Output - 6)
Analog Input Pins	: 6
DC Current per I/O Pin	: 40 mA
DC Current for 3.3V Pin	: 50 mA
Flash Memory	: 32 KB (for Bootloader - 0.5 KB)
SRAM	: 2 KB
EEPROM	: 1 KB
Clock Speed	: 16 MHz

3.5.2 ESP 8266 Wi-Fi Module

ESP8266 has a 32-bit Tensilica processor and a lot of standard digital interfaces. The antenna switches, RF bal-un, power amplifier, low-noise receiver amplifier, filters and power management modules are also there. Because of its large operating temperature range, it can function reliably in industrial settings. The chip provides dependability, compactness, and robustness because to highly integrated on-chip functionalities and a low number of external discrete components. The ESP 8266 Wi-Fi Module is shown in Figure 3.6.

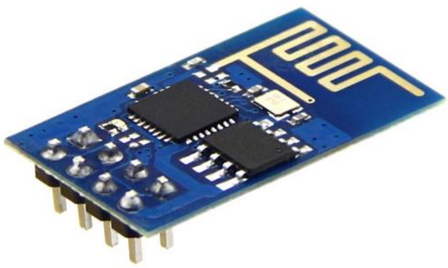


Fig. 3.6. ESP 8266 Wi-Fi Module [9].

In the ESP8266E microcontroller, there is a 32-bit RISC processor called the Tensilica L106 that uses very little power and can run at speeds up to 160 MHz. Because of the Real-Time Operating System (RTOS) and Wi-Fi stack, about 80% of the computer's processing power can be used by users to write programs and develop new programs.

3.5.3 MQ-2 Gas sensor

The MQ-2 sensor is a versatile gas sensor that can detect a variety of gases, including alcohol, carbon monoxide, hydrogen, isobutene, liquefied petroleum gas, methane, propane, and smoke, among others. This module has a "grove" style connector as well as a 3pin male header interface. A voltage divider network can detect gas concentrations. Gas Sensor MQ2 uses 5V DC and around 800mW. It detects LPG, Smoke, Alcohol, Propane, Hydrogen, Methane, and Carbon Monoxide from 200 to 10000ppm. The MQ2 gas sensor can detect LPG, Propane, Hydrogen, Methane, and other flammable steam. It is cheap and versatile. It detects flammable gas and smoke. The smoke sensor runs on 5 volts.

Fig. 3.7. MQ-2 CO Gas sensor [11]

3.5.4 MQ-6 LPG Gas sensor

The MQ6 (LPG Gas Sensor) is a straightforward LPG sensor. This sensor is appropriate for detecting LPG, iso-butane, propane, and LNG and can be utilized in gas leakage detection devices in consumer and industrial applications. Alcohol, cooking odors, and cigarette smoke should all be avoided.

Fig. 3.8 MQ-6 LPG Gas sensor [10]

The MQ-6 can detect gases from 200 to 10,000 ppm (ppm). This sensor's sensitivity is outstanding, as is its reaction time. The sensor's output is analogue resistance. The driving circuit is simple: 5V to the heating coil, a load resistance, and an ADC output.

3.5.5 MQ-135 Sensor

Fig. 3.9. MQ135 Air quality sensor module [12].

The MQ135 is an Air Quality sensor (Figure 3.9). Gas-sensitive elements and ammonia may be detected using this device. Smoking, alcohol, and NOx are all detected by air quality sensors. For office or industrial application, it's simple driving and monitoring circuitry is appropriate.

3.5.6 BMP280 Absolute Barometric Pressure Sensor

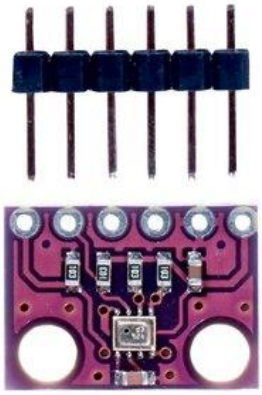


Fig. 3.10. BMP280 Absolute Barometric Pressure Module [13].

The BMP280 is a mobile-friendly absolute barometric pressure sensor. Its tiny size and low power consumption enable it to be used in battery-powered devices including phones, GPS units, and watches. The BMP280 sensor module requires 1.71V minimum voltage (VDD), while earlier versions required 1.8V. (VDD). BMP280 uses 2.7uA of current. BMP280 has new filter modes. For applications like floor level detection, the BMP280's relative precision of 0.12 hPa (equal to 1 m difference in height) and offset temperature coefficient (TCO) of 1.5 Pa/K (equivalent to 12.6 cm/K) make it ideal.

3.6 Software Requirements

1. Arduino IDE
2. THINGSPEAK website
3. Android Studio

3.7 Problem Formulation and Decision

For the formulation we are taking Carbon monoxide concentrations.

Consider the following equation:

R0 = Sensor resistance in clean air.

Rs = Sensor resistance in the shown gases.

Factor of pure air R0 = 10 (provided in the datasheet of the sensor).

pcurve = 3, values for a given gas retrieved from the sensor datasheet's curve.

The sensor is calibrated by dividing the sensor resistance by the R0 clean air factor. To obtain the gas concentration, you must first compute the gas percentage.

Gas percentage = $(log(RsRoRatio) / pcurve2 - pcurve1 + pcurve0)$ 10 --------------- (3.1)

To find the gas concentration, gas percentage need to be divided by the Ro,

concentration = $gas\ percentage\ Ro$ --- (3.2)

3.8 Dataset

The following information was gathered from the gas sensors that were connected to the Arduino. We exported them to an Excel Sheet in a certain format. These data will be analysed later on. The more data we have, the better our chances of getting more accurate results. We can also forecast future air quality data. The standard dataset of the pollutants, air quality and our device dataset are shown ahead.

3.8.1 Standard Dataset

Table 3.1: US Embassy Dataset for Air Quality. [6]

Air Quality Value	AQI	Details
0-50	Good	Air quality is rated good, and air pollution is considered to pose little or no harm.
51-100	Moderate	Air quality is satisfactory, although some pollutants may provide a mild health risk to a very small number of persons who are especially sensitive to air pollution.
101-150	Unhealthy for some sensitive groups	Members of vulnerable populations may suffer health consequences. The general population is unlikely to be harmed.
151-200	Unhealthy	Everyone could begin to see health implications; individuals of vulnerable groups may notice more severe health problems.
201-300	Very Unhealthy	Health alerts for emergency situations. The population as a whole is more likely to be affected.
301-500	Hazardous	Everyone is at risk of experiencing more significant health impacts.

3.8.2 Carbon Monoxide Limit of Exposure in Indoor

In households without gas stoves, average values range from 0.5 to 5 parts per million (ppm). Readings at well-adjusted gas stoves are frequently 5 to 15 ppm, but readings at badly adjusted stoves might be 30 ppm or more.

Occupational Safety and Health Administration (OSHA) permitted the limit of exposure of CO is around 50ppm (parts per million) as a time-weighted hour of 8. [7]

3.8.3 LPG Limit of Exposure in Indoor

A study in National Institute of Occupational Safety and Health suggest that safe exposure limit of LPG is 1000ppm for an 8-hour working day. LPG is regarded immediately hazardous to life and health due exclusively to safety concerns about the chance of explosion. LPG may generate a combustible mixture with air at concentrations ranging from 2% to 10%. [16]

3.8.4 Second Dataset of Our devices

CO (ppm)	LPG	Temperature	Pressure	Altitude	AQ
12.65	5.81	26.06	1013.77	20.90	40.21
12.11	5.53	26.13	1013.79	20.78	35.99
11.84	8.4	23.8	1013.84	20.92	29.09
17.56	8.9	24.21	1013.85	19.75	35.99
17.09	8.83	24.38	1013.82	20.62	31.26
14.9	9.34	24.58	1013.76	20.72	33.55
15.32	7.93	24.67	1013.75	20.79	27.46
13.79	8.97	24.91	1013.74	20.92	19.25
14.9	8.62	25.12	1013.59	19.91	15.06

Table 3.2: Our devices Dataset

3.9 Summary

In this section, we talked about our system device, which we manufactured ourselves. This part included a circuit diagram and a block diagram. We also briefly described and introduced our formula for detecting gas percentage and concentration.

CHAPTER 4

RESULT AND DISCUSSION

4.1 Introduction

This chapter is all about our devices experiment and its results. We put our device in different situations to get data as much as possible.

4.2 Dataset Description

We used the US Embassy of Bangladesh dataset to measure and compare the performance of our device Air quality data. There are 2500 records in this collection, each with its own AQI score. These data sets were gathered from their website. We got the exposure limit for carbon monoxide gas from Environmental Protection Agency (EPA). They release the latest and safest threshold limit for any toxin gas which are dangerous for health.

4.2.1 Displaying data in LCD Display

Figures 4.1(a), 4.1(b), 4.1(c), 4.1(d), 4.1(e) and 4.1(f) illustrate the output of the LCD Display

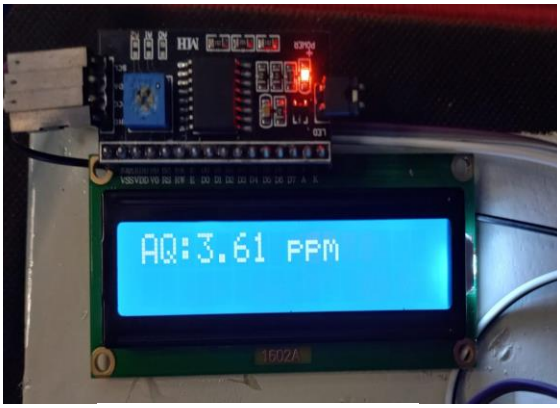


Fig 4.1(a): Air Quality

Fig 4.1(b): CO

Fig 4.1(c): LPG

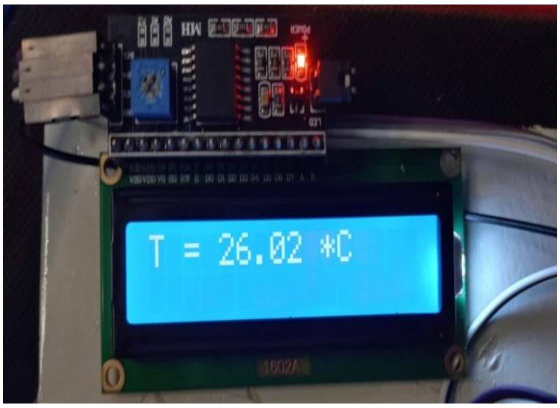


Fig 4.1(d): Temperature

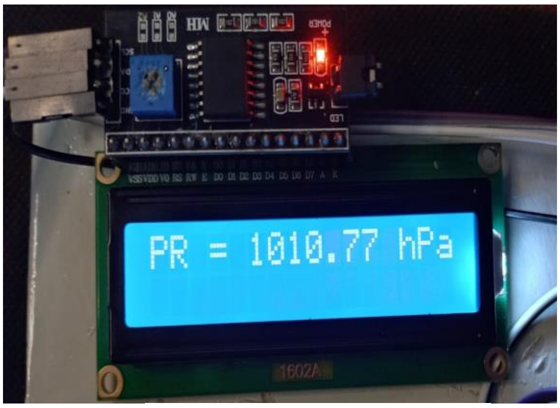


Fig 4.1(e): Air Pressure

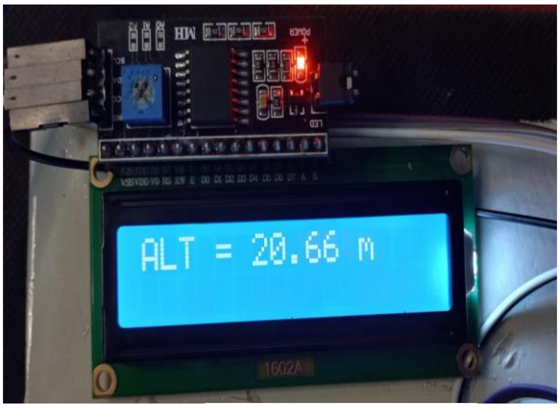


Fig 4.1(f): Altitude

4.2.2 Sending data to the cloud platform

We've chosen 6 data points to submit to the cloud platform. Every 20 seconds, the server receives new input from the Arduino. We used six variables to store the data and sent it to five distinct cloud channel fields. The Mobile Application perspective of the system is shown in Fig. 4.2(a), 4.2(b) and 4.2(c). Figures 4.3(a), 4.3(b), 4.3(c), 4.3(d), 4.3(e), and 4.3(f) depict the output of the cloud platform for air quality, CO, temperature and humidity, and LPG, respectively.

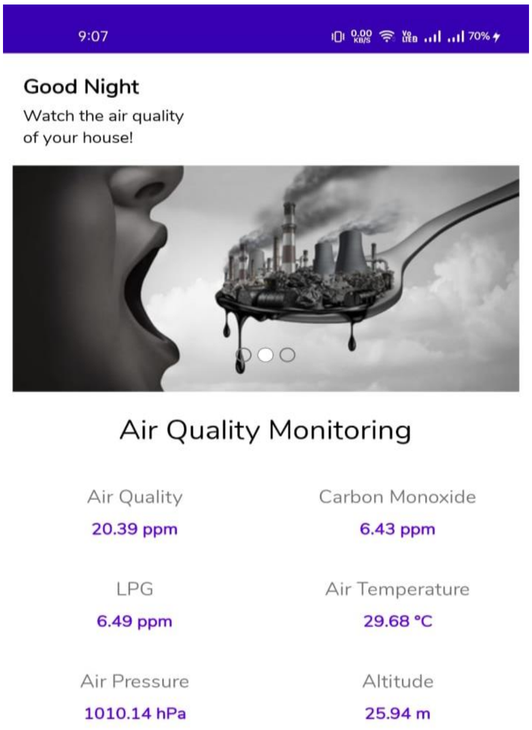


Fig. 4.2(a) Mobile Application view

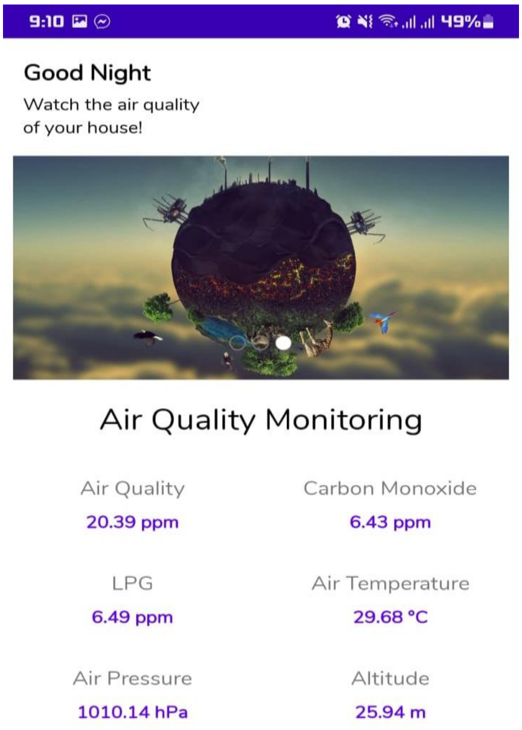


Fig. 4.2(b) Mobile Application view

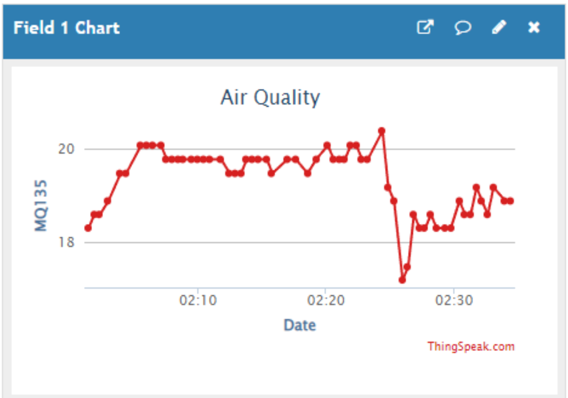


Fig. 4.3(a) Cloud platform (Air Quality)

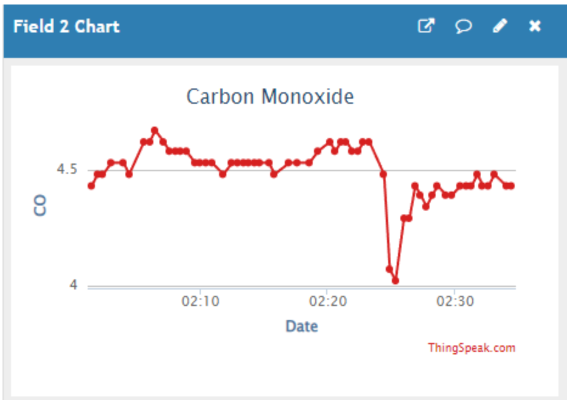


Fig. 4.3(b) Cloud platform (CO)

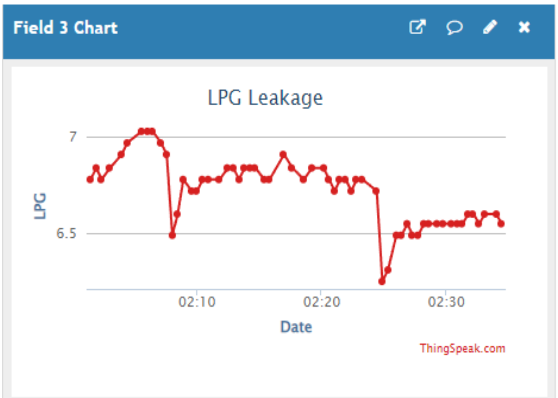


Fig. 4.3(c) Cloud platform (LPG)

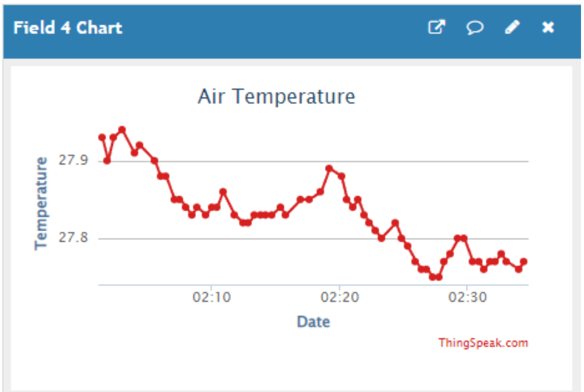


Fig. 4.3(d) Cloud platform (Temperature)

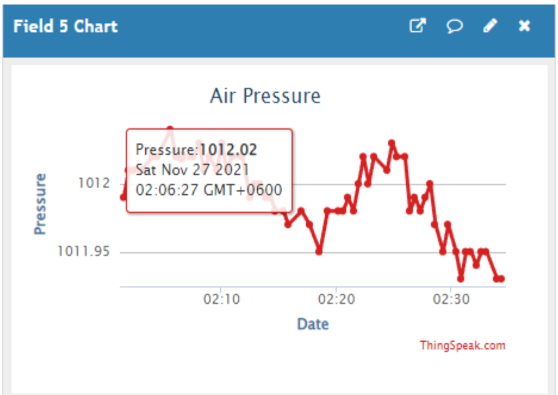


Fig. 4.3(e) Cloud platform (Air Pressure)

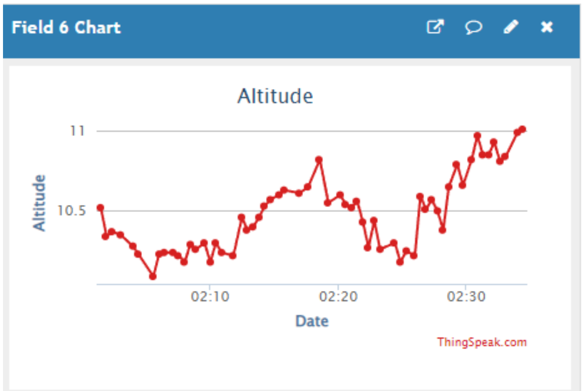


Fig. 4.3(f) Cloud platform (Altitude)

4.3 Result

The standard safety rate of carbon monoxide is 0-9 ppm. For indoor environment OSHA's permitted level of carbon monoxide is 50 ppm and **ACGIH TLV** known as The American Conference of Govt Industrial Hygienists has declared CO's threshold limit is 35 ppm as an 8-hour TWA. Our device recorded an average value of 12ppm which gives an assurance that our device giving the accurate value of CO. For the standard value of air quality we took the dataset from US Embassy of BD which recommends a numerical 0-50 for a safer level. Our device getting an average value of 25 which is said to be good in indoor environment according to the dataset and also assure that out device giving an accurate value. OSHA and PEL suggested a safety level for LPG leakage in the air is 1000 ppm with an explosion chance of concentration between 2%-10%. Our device detected LPG concentration in the air is 5-10 ppm on average which is an assurance of accurate data and also maintaining safety level.

4.3 Cost Analysis

The project's cost analysis is shown in Table 3.4.

Table: 3.3: Cost Analysis of the Project

Equipment	**Quantity**	**Price (BDT)**
Arduino UNO	1	450
Esp8266	1	250
MQ-2	1	135
MQ-6	1	135
MQ-135	1	270
BMP280	1	210
16x2 LCD Display	1	190
I2C	1	140
PCB Board	1	50
Jumper Wire(M-M) (M-F)	1+1	80
Total		**1910**

CHAPTER 5
CONCLUSION AND FUTURE WORKS

5.1 Conclusion

The overall air quality of Bangladesh has been classified as UNHEALTHY, according to a study published by IQAIR in 2020, with an average score of 162 out of 200. A 2020 survey also identified Bangladesh as the world's most polluted nation, ranking it first out of 106 countries worldwide for air pollution. [14] We must maintain a close watch on the air quality in order to avoid pollution to the greatest extent feasible. Numerous people are worried about the pollution in the air they breathe since it has the potential to have both short- and long-term health implications. Titas gas transmission authority data shows that there were at least 3,819 gas leak-related incidents in 2013-14, an increase from 5,123 in 2014-15, according to a study by the authority.[15] The system we are proposing is Air Quality Monitoring system employs various sensors and a microprocessor to assess air quality in indoor locations, as well as a WIFI Module to send real-time data and information to a cloud server, which can be accessed by smart phones and web-enabled devices. We will also create a user-friendly Android application that will allow you to check the measured data quickly and conveniently. We tried to create a device which costs significantly low than the previous works. We also tried to provide more accurate than the previous works provided. The paper elaborates how our system works, what algorithm it follows and analyzed the result we have got from the device. Thus, this research enables people to monitor the air they breathe, particularly those with underlying health issues.

5.2 Future work

In this project we did not include any dust sensor which is really essential for developing countries like ours where we can not walk in the road without wearing face mask due to excessive dust in the air. Currently our device can bring out 6 data from the sensors. But our sensors have potential to detect a lot more gasses which are dangerous for health. But due to lack of time we could not research and calculate on those toxin gasses. In future we are looking forward to add more gas input. Our system can store a lot of data in the webserver including exact date. We are looking forward to implement machine learning algorithms to provide a better service out of our device.

REFERENCES

1. "Center for Science Education", https://scied.ucar.edu/learning-zone/air-quality/what-is-air-quality
2. K. S. E. Phala, A. Kumar, and Gerhard P. Hancke "Air Quality Monitoring System Based on ISO/IEC/IEEE 21451 Standards" in IEEE (2016)
3. Somansh Kumar, Ashish Jasuja "Air Quality Monitoring System Based on IoT using Raspberry Pi" in International Conference on Computing, Communication and Automation (ICCCA2017)
4. Mukesh Jha, Prashanth Reddy Marpu, Chi-Kin Chau, Peter Armstrong "Design of Sensor Network for Urban Micro-Climate Monitoring" in 2015 IEEE First International Smart Cities Conference (ISC2)
5. Shete, Rohini, and Sushma Agrawal "IoT based urban climate monitoring using Raspberry Pi" in IEEE International Conference In Communication and Signal Processing (ICCSP), pp. 2008- 2012, 2016
6. "US Embassy Dataset", https://bd.usembassy.gov/embassy/air-quality-data/
7. "Environmental Protection Agency", https://www.epa.gov/indoor-air-quality-iaq/carbon-monoxides-impact-indoor-air-quality
8. "Arduino Uno r3", https://store.arduino.cc/usa/arduino-uno-rev3
9. "ESP 8266", https://components101.com/wireless/esp8266-pinout-configuration-features-datasheet
10. "MQ-6 LPG Gas Sensor", https://components101.com/sensors/mq-6-gas-sensor-pinout-equivalent-datasheet
11. "MQ-2 CO Gas Sensor", https://components101.com/sensors/mq2-gas-sensor
12. "MQ 135", https://components101.com/sensors/mq135-gas-sensor-for-air-quality
13. "BMP280 Temperature Sensor", https://components101.com/sensors/gy-bmp280-module
14. "IQAIR", https://www.iqair.com/bangladesh.
15. "Dhaka Tribune", https://archive.dhakatribune.com/feature/tech/2017/06/09/sniffer-made-dhaka-reduce-household-gas-accidents
16. "Occupational Safety and Health Administration", www.osha.gov/chemicaldata/484

APPENDIX

Microcontroller Code of the proposed model:

```
#include <Wire.h>
#include <LCD.h>
#include <SPI.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE);
#include <SoftwareSerial.h>
SoftwareSerial ser(2, 3); // RX, TX

#include <Adafruit_BMP280.h>
#define BMP_SCK  (13)
#define BMP_MISO (12)
#define BMP_MOSI (11)
#define BMP_CS   (10)
Adafruit_BMP280 bmp;

int gas_sensor = A0;
float m = -0.3376;
float b = 0.7165;
float R0 = 15.52;

int CO_sensor = A1;
float m1 = -0.6527;
float b1 = 1.30;
float R01 = 6.42;

int LPG_sensor = A2;
float m2 = -0.6527;
float b2 = 1.30;
float R02 = 6.42;
```

```
String apiKey = "BNUH2MS2Y4HL9KGH";
void setup() {

 Serial.begin(115200);
 lcd.begin(16, 2);
 lcd.backlight();
 lcd.setCursor(0, 0);
 lcd.print(" Air Quality");
 lcd.setCursor(0, 1);
 lcd.print("Monitoring System");
 delay(2000);
 lcd.clear();

ser.begin(115200);
ser.println("AT+RST");
char inv ='"';
String cmd = "AT+CWJAP";
String Host_Name = "chLoRoFoRm"; // Edit Host_Name
String Password = "onemanarmy"; // Edit
cmd+= "=";
cmd+= inv;
cmd+= Host_Name;
cmd+= inv;
cmd+= ",";
cmd+= inv;
cmd+= Password;
cmd+= inv;
ser.println(cmd);
 pinMode(gas_sensor, INPUT);
 pinMode(CO_sensor,INPUT);
```

```
 pinMode(LPG_sensor,INPUT);
 pinMode(9,OUTPUT);
  Serial.println(F("BMP280 test"));
  if (!bmp.begin(0x76)) {
    Serial.println(F("Could not find a valid BMP280 sensor, check wiring or "
                      "try a different address!"));
    while (1) delay(10);
  }
 bmp.setSampling(Adafruit_BMP280::MODE_NORMAL,
                  Adafruit_BMP280::SAMPLING_X2,
                  Adafruit_BMP280::SAMPLING_X16,
                  Adafruit_BMP280::FILTER_X16,
                  Adafruit_BMP280::STANDBY_MS_500);
}
void loop() {
float sensor_volt;
 float RS_gas;
 float ratio;
 float sensorValue = analogRead(gas_sensor);
 sensor_volt = sensorValue*(5.0/1023.0);
 RS_gas = ((5.0*10.0)/sensor_volt)-10.0;
 ratio = RS_gas/R0;
 double ppm_log = (log10(ratio)-b)/m;
 double ppm = pow(10, ppm_log);
 Serial.print("Air Quality = ");
 Serial.println(ppm);
 lcd.clear();
 lcd.setCursor(0,0);
 lcd.print("AQ:");
 lcd.print(ppm);
 lcd.print(" ppm ");
```

```
delay(2000);
lcd.clear();

float sensor_volt1;
float RS_gas1;
float ratio1;
float sensorValue1 = analogRead(CO_sensor);
sensor_volt1 = sensorValue1*(5.0/1023.0);
RS_gas1 = ((5.0*10.0)/sensor_volt1)-10.0;
ratio1 = RS_gas1/R01;
double ppm_log1 = (log10(ratio1)-b1)/m1;
double ppm1 = pow(10, ppm_log1);
Serial.print("CO PPM = ");
Serial.println(ppm1);
lcd.clear();
lcd.setCursor(0,0);
lcd.print("CO:");
lcd.print(ppm1);
lcd.print(" ppm ");
delay(2000);
lcd.clear();

 float sensor_volt2;
 float RS_gas2;
 float ratio2;
 float sensorValue2 = analogRead(LPG_sensor);
 sensor_volt2 = sensorValue2*(5.0/1023.0);
 RS_gas2 = ((5.0*10.0)/sensor_volt2)-10.0;
 ratio2 = RS_gas2/R02;
 double ppm_log2 = (log10(ratio2)-b2)/m2;
 double ppm2 = pow(10, ppm_log2);
```

```
 Serial.print("LPG PPM = ");
 Serial.println(ppm2);

     lcd.clear();
     lcd.setCursor(0,0);
     lcd.print("LPG:");
     lcd.print(ppm2);
     lcd.print(" ppm ");
     delay(2000);
     lcd.clear();

Serial.print(F("Temperature = "));
Serial.print(bmp.readTemperature());
Serial.println(" *C");
 lcd.clear();
 lcd.setCursor(0,0);
 lcd.print("T = ");
 lcd.print(bmp.readTemperature());
 lcd.print(" *C");
 delay(2000);

 Serial.print(F("Pressure = "));
 Serial.print(bmp.readPressure()/100);
 Serial.println(" Pa");
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("PR = ");
    lcd.print(bmp.readPressure()/100);
    lcd.print(" hPa");
    delay(2000);
```

```
    Serial.print(F("Approx altitude = "));
    Serial.print(bmp.readAltitude(1013.25));
    Serial.println(" m");
      lcd.clear();
      lcd.setCursor(0,0);
      lcd.print("ALT = ");
      lcd.print(bmp.readAltitude(1013.25));
      lcd.print(" m");
      delay(2000);

 String state1=String(ppm);
 String state2=String(ppm1);
 String state3=String(ppm2);
 String state4=String(bmp.readTemperature());
 String state5=String(bmp.readPressure()/100);
 String state6=String(bmp.readAltitude(1013.25));
 String cmd = "AT+CIPSTART=\"TCP\",\"";
 cmd += "184.106.153.149";
 cmd += "\",80\r\n\r\n"; // port 80
 ser.println(cmd);
 Serial.println(cmd);
 if(ser.find("Error")){
 Serial.println("AT+CIPSTART error");
 return;
}
String getStr = "GET /update?api_key=";
 getStr += apiKey;
 getStr +="&field1=";
 getStr += String(state1);
 getStr +="&field2=";
 getStr += String(state2);
```

```
 getStr +="&field3=";
 getStr += String(state3);
 getStr +="&field4=";
 getStr += String(state4);
 getStr +="&field5=";
 getStr += String(state5);
 getStr +="&field6=";
 getStr += String(state6);

 getStr += "\r\n\r\n";
 cmd = "AT+CIPSEND=";
 cmd += String(getStr.length());
 ser.println(cmd);
 Serial.println(cmd);
if(ser.find(">")){
 ser.print(getStr);
 Serial.print(getStr);
 }
 else{
 ser.println("AT+CIPCLOSE");
 // alert user
 Serial.println("AT+CIPCLOSE");
 }
}
```