IOT BASED AIR AND SOUND POLLUTION MONITORING SYSTEM

Submitted in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY
IN

ELECTRONICS AND COMMUNICATION ENGINEERING

Under the guidance of

Dr. Neeru Sharma

Assistant Professor (Senior Grade)

By

Nipun Aggarwal (141085)

Shashi Shekhar Shukla (141090)

Raaj (141098)

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY

WAKNAGHAT, SOLAN – 173234

HIMACHAL PRADESH, INDIA

Candidate's Declaration

I hereby declare that the work presented in this report entitled**"IOT Based Air and Sound Pollution Monitoring System"** in partial fulfillment of the requirements for the award of the degree of bachelor of technology in electronics and communication engineering submitted at **Jaypee University of Information Technology, Vaknaghat India,** is an authentic record of my work carried out over a period from July 2017 to May 2018 under the supervision of Dr**. Neeru Sharma (Assistant Professor, E.C.E Department)** . The matter embodied in the report has not been for the award of any other degree or diploma

Nipun Aggarwal	Shashi Shekhar Shukla	Raaj
(141085)	(141090)	(141098)

This is to certify that above statement made by the candidate is true to the best of my knowledge.

Dr. Neeru Sharma

Assistant Professor (Senior Grade)

Electronics and Communication Engineering

Jaypee University of Information Technology, Waknaghat, India

Dated:

Acknowledgement

We would like to express our special gratitude to our project coordinator **Dr.Neeru Sharma** who helped us in to form a concept and idea of the project and actual building of steps used to complete the project. We would also like to thank our Head of Department for providing us this golden opportunity to work on a project like this.

This project would have been impossible if it was not for the kind support, extreme support and continuous counsel of my learned project guide. The deep sense of heartfelt gratitude that I owe to my learned guide and supervisor **"Dr Neeru Sharma"** cannot be expressed in words. Her unconquerable help, constant inspiration, key counsel, generous nature are among the very few favour that she give upon me time to time from the inception of this project to this day.

Nipun Aggarwal	Shashi Shekhar Shukla	Raaj
(141085)	(141090)	(141098)

(Department of Electronics and Communication Engineering)

CONTENTS

List of Figures

List of Tables

IOT Based Air and Sound Pollution Monitoring System

Abstract

Pollution of air and sound is increasing abruptly. To bring it under control its monitoring is majorly recommended. To overcome this issue, we are introducing a system through which the level of sound and the existence of the harmful gases in the surroundings can be detected. The growing pollution at such an alarming rate has started creating trouble for the living beings, may it be high decibels or toxic gases present in the environment leaves a harmful effect on human's health and thus needs a special attention.

The main objective of IOT Air & Sound Monitoring System is that the Air and sound pollution is getting larger these days. It is necessary to detect air quality and keep it under control for a better future and healthy living for all. Therefore we initiate an air and sound pollution system that allows us to assess and examine live air quality as well as sound pollution in an area through Internet of Things. Model uses air sensor to recognise presence of harmful gases present in the atmosphere and repeatedly convey the data. Also, system keeps measuring sound level and reports it. This allows authorities to monitor air pollution in different areas and act against it. Therefore officials can keep a watch on the noise pollution near schools, hospitals and areas where noise is not allowed, and if system detects air quality and noise issues it alerts authorities so they can take measures to control the issue.

Chapter 1: Introduction

1.1 Introduction: Internet of Things

Internet of Things (IOT) is a network of devices that connect directly with each other to capture and share data through a secure service layer that connects to a central command and control server in the cloud. The closure look suggest that the way people collect, record and analyze data—not just in health care but in every industry today. Theidea of devices connecting directly with each other is basically called Internet of Things.

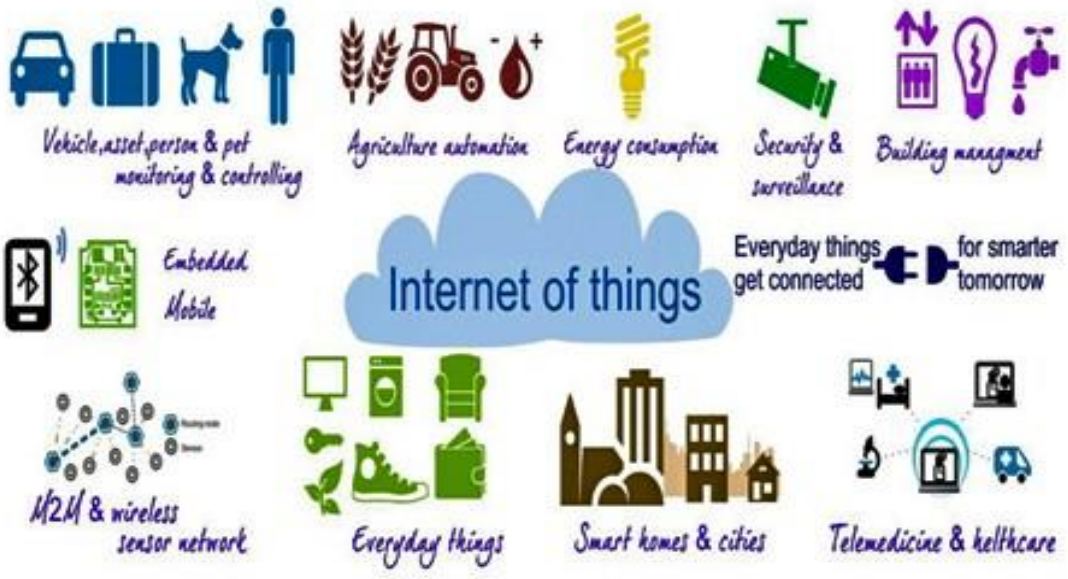

Figure 1.1: Internet of Things

The Internet of Things also called the Internet of Objects, refers to a wireless network between objects, usually the network will be wireless and self-configuring. Internet of Things (IOT) is one of the major component advances in present time that links the internet with everyday sensor and working devices. Smart objects play an important role in the Internet of Things vision, since embedded communication and information technology would have the potential to change the utility of these objects. Using sensors they are able to recognize their condition, and via built-in connecting power they would be able to interact with each other.

Internet of Things (IOT) came into existence in 2009. It encircles the idea of connecting all gadgets and devices to the internet. The concept of Internet of Things is actually trying to change our world by connecting everything with each other. It is basically making our health, and businesses, society by developing products which would lead to a comfortable life. By 2020 it is estimated that, about 50 billion devices would be linked to the Internet and the market would be worth around 14 trillion USD.

IOT in different application domains alike web property of everyday objects may be accustomed remotely confirm their state in order that we will perpetually collect the info and knowledge on physical objects and processes. This qualifies several options of the important world to be spot at a antecedently unattained level of detail and at terribly low value nearly too negligible. This could not afford an improved understanding of the underlying processes, however additionally for a lot of economical management and management. The potential to react to events during this world in associate automatic manner not solely disclose new opportunities for addressing composite or unfavorable things, however additionally allows a large type of business processes to be optimized.

.

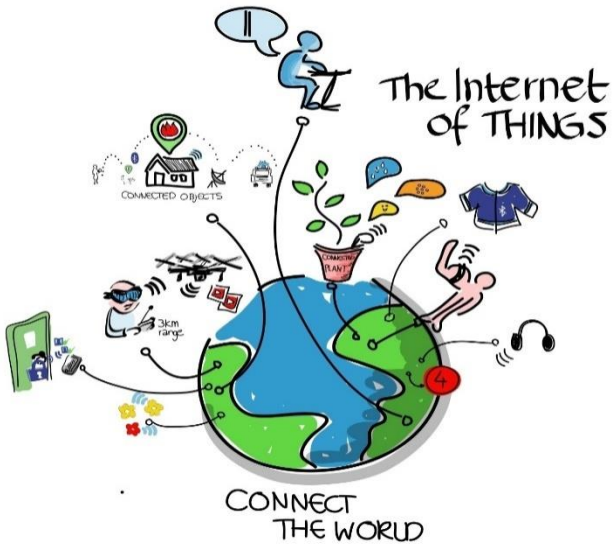

Figure1.1.1: IOT Connecting the World

1.2 Benefits of IOT

Ubiquitous networks -: Personal Wi-Fi on your smart phones and on many of the other devices. Everyone (and everything) wants as well as needs to be connected.

Connected computing -:We want all of the devices, smart phones, televisions(colored or black and white), dvd players, vehicles etc. to keep record of what we are doing, seeing, reading, and/or listening to as we sway through the day, from one place to another – the handoffs from device to device is happening already.

Analytics-as-a-Service-: The API and App economies are already wide and growing which enables to "do something interesting" as long as it can be connected to an API or can invoke an App that carries out a network-based service. The thing is a data generator as well as collector that learns from, makes forecast, and maybe even takes data-driven actions in response to the data that are collected too.

Marketing automation-: Smart phone customer engagement, geological-location, Apple's iBeacon etc. are all developing a network of knowledge and information regarding customers' locations, intentions, preferences, as well as buying patterns. Obviously, the degree of geological location-based knowledge needs to maintain the right balance between user privacy as well as the timely delivery of important and significant products and services to the particular user.

Supply Chain Analytics-: Delivering the just-in-time products at the time of need (inclusive of the use of RFID-based tracking). Significantly, everything is a customer (inclusive of machines, automobiles, manufacturing plants, ATM machines, etc.), as well as the IOT is monitoring, watching, as well as waiting for the product needs to arise.

1.3 Internet of Things in Pollution Monitoring

Due to recent technological advances, the construction material for small and low cost sensors became technically and economically feasible. Even though, Industrialization increase the degree of automation at the same time it increases the pollution by releasing the unwanted parameters in environment especially in industrial areas. So there should be a system to monitor and assess the industrial pollution. Particular attention is given to factors which may affect human health and the health of the natural system itself. Industrial monitoring is the gathering of data at different locations of industries and at orderly gap of time in order to provide the information which may be used to define present conditions. Due to the complexity of parameters large variations are found between different industries.

To build a robust system that can measure the industrial pollution and help to reduce it and to decrease human interference in monitoring the industrial pollution to reduce pollution and provide a healthy environment for the workers to work in. To make the industrial pollution monitoring a wireless system. To protect the environment from industrial pollution. To build a robust system that evaluates the industrial pollution continuously and indicates when there is an increase in the emissions and takes action to control it using wireless technology that is IOT. These improvements can be used to develop and implement its environmental policy and manage its environmental aspects.

Figure 1.2: Air Pollution (Smoke)

Chapter 2: Literature Review

2.1 SomayyaMadakam, R. Ramaswamy "Internet of Things"(2015) [3]

This paper deals with that future is Internet of Things, which will modify the real world things into intelligent practical things. IOT works on the principal to unite everything in our world under same infrastructure, giving us not only control of objects around us, but also keeping us informed of the condition of the objects. The objective of this paper is to deliver an abstract of Internet of Things, architectures, and essential technologies and their usages in our daily life. Hence, this document will give good comprehension for the new researchers, who want to do research in the field of Internet of Things.

2.2AnjaiahGuthi "Efficient Noise and Air Pollution Monitoring System" (2016) [4]

This paper deals with smart sensor networks that are an emerging field of research which combines many challenges of computer science, wireless communication and electronics. In this research paper a solution for monitoring the noise and air pollution levels in industrial environment or any other area of interest using wireless embedded computing system is proposed.

The running representation of the initiated hardware is evaluated using original implementation, consisting of Arduino UNO board, sensor devices and MATLAB with Arduino hardware support package. The hardware is tested for two or three parameters like noise, CO2 and radiation levels with respect to the normal behavior levels or given specifications which provide a control over the pollution monitoring to make the environment smart.

2.3 PalaghatYaswanthSai (2017) [5]

This document deals with the IOT Based Air Pollution Monitoring System in which we will gather the air value in ppm as well as sound value in decibel over a web using internet and with the help of Wi-Fi module and will trigger an alarm when the air quality goes down further a certain level, means when there are sufficient amount of harmful gases are present in the atmosphere like CO2, smoke, alcohol, benzene and NH3. It will show the air quality in PPM on the LCD 16x2 display and as well as on web so that we can gather information very easily. In this MQ135 gas sensor is used which is the best choice for monitoring air quality as it can detect most harmful gases and can measure their value efficiently. In this project, you can gather the level of air pollution and noise from anywhere using your computer or mobile. System can be set anywhere in the world and we can also activate some objects like for example when pollution goes beyond some set level we can switch on the exhaust fan.

2.4 Uppugunduru Anil Kumar, G Keerthi (2017) IOT Based Air and Noise Pollution Monitoring using Raspberry Pi [6]

A systematic environmental monitoring hardware is required to monitor the conditions in case of exceeding the level of parameters (e.g., noise, CO2). When the things like environment equipped with sensor devices, microcontroller and various software applications becomes a self-protecting and self-monitoring environment. People need different types of monitoring hardware which are depends on the type of data monitored by the sensor. Event Detection based and Spatial Process Estimation are the two categories to which applications are classified. Initially the sensor devices are deployed in environment to detect the parameters (e.g., noise, CO and radiation levels etc.) while the data acquisition, computation and controlling action (e.g., the variations in the noise and CO levels with respect to the specified levels). Sensor devices are placed at different locations to collect the data to predict the behaviour of a particular area of interest. The

main focus of the this document is to model and implement an efficient monitoring hardware through which the required values are measured remotely using web and the information gathered from the sensors are stored in the cloud and to project the estimated trend on the web browser.

2.5 Ashvini S .kale in Air Pollution Monitoring System Using Mobile GPRS Sensors [7]

This paper contains brief introduction to vehicular pollution. The proposed system consists of a transmitter and receiver part. The transmitter part is integrated with single-chip microcontroller, air pollution sensors array, a General Packet Radio Service Modem (GPRS Modem), and a Global Positioning System Module (GPS-Module) for transmitting the information.

2.6 L.Ezhilarasi, K.Sripriya, "A SYSTEM FOR MONITORING AIR AND SOUND POLLUTION USING ARDUINO CONTROLLER WITH IOT TECHNOLOGY" (2017) [8]

The document contains embedded device for measuring noise and air levels in atmosphere and to make the environment intelligent or interactive with things. The proposed model is flexible and distributive in nature to measure the environmental parameters. The architecture is developed for noise and air pollution monitoring. Smart sensor network are the coming field of research and investigation which may lead to many challenges of computer science, wireless communication and electronics.

2.7 Dr. Siva yellampalli "IOT Based Air and Noise Pollution Monitoring in Urban and Rural Areas, Important Zones like Schools and Hospitals in Real Time" (2017) [9]

Today's major environmental & public issue is air pollution. According to the report of World Health Organization (WHO), air pollution is significant risk factor for multiple health conditions including skin & eye infection, irritation of nose, throat & eyes. It also causes serious conditions like heart disease, lung cancer difficulty in breathing & many. Parking management is also main public issue in most of metropolitan cities and that is also the reason of many problems. The main objective of project is by using various sensors, GSM/GPRS module and Cloud/server to design an efficient and remote system to monitoring the level of various pollutants causing pollution and to minimize the effect of these parameters without affecting the natural environment and provide live updates to avoid conflicts.

CHAPTER 3: Proposed Model

3.1 Introduction

Pollution of air and sound is increasing abruptly. To bring it under control its monitoring is majorly recommended. To overcome this issue, we are introducing a system through which the level of sound and the existence of the harmful gases in the surroundings can be detected. The growing pollution at such an alarming rate has started creating trouble for the living beings, may it be high decibels or toxic gases present in the environment leaves a harmful effect on human's health and thus needs a special attention.

The main objective of IOT Air & Sound Monitoring System is that the Air and sound pollution is getting larger these days. It is necessary to detect air quality and keep it under control for a better future and healthy living for all. Therefore we initiate an air and sound pollution system that allows us to assess and examine live air quality as well as sound pollution in an area through Internet of Things. Model uses air sensor to recognise presence of harmful gases present in the atmosphere and repeatedly convey the data. Also, system keeps measuring sound level and reports it. This allows authorities to monitor air pollution in different areas and act against it. Therefore officials can keep a watch on the noise pollution near schools, hospitals and areas where noise is not allowed, and if system detects air quality and noise issues it alerts authorities so they can take measures to control the issue.

3.2 Block Diagram

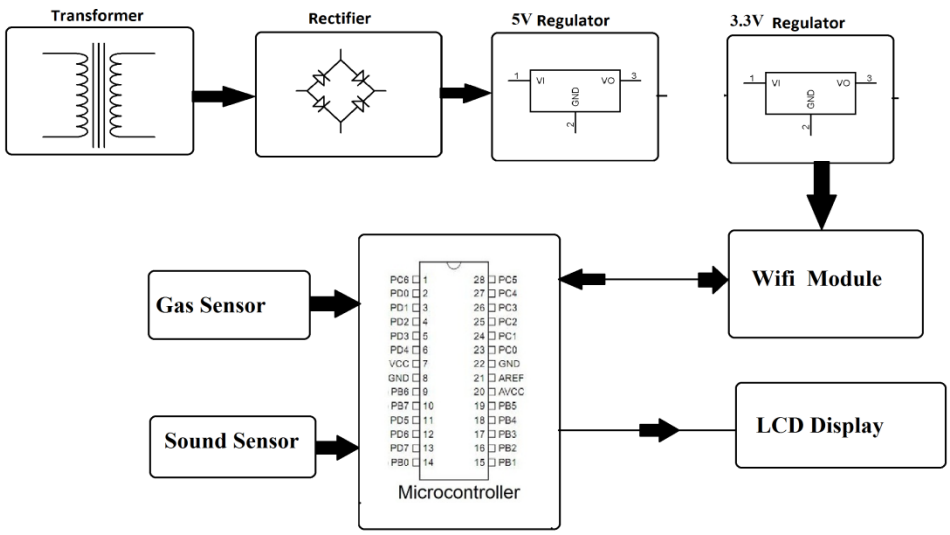

Figure 3.2: Block Diagram

3.3 Hardware Requirements

3.3.1 MQ135 Gas sensor:

MQ-135 Module sensor has lesser conductivity in clean air. When the ideal combustible gas is there, the sensors conductivity is more peaked along with the gas concentration raising. Convert change of conductivity to parallel output signal of gas immersion. Gas sensor has greater sensitivity to Ammonia (NH3), Sulphide (S2-) and Benzene (C6H6) also it is sensitive to smoke and other harmful materials. It is with low expense and suitable for different applications.

Figure 3.3.1: MQ135 Sensor

3.3.2 Sound Sensor:

Using mic as a sound sensor in the proposed system. This sensor operates on 3.3V to 5V dc. It provides a digital output on the display. It detects sound from the atmosphere and output digital trigger signal. It has a sound set point adjust, to adjust the threshold value. So if the reading goes beyond the threshold value authority will able to know the sound pollution is at a high level of that particular place.

Figure 3.3.2: Sound Sensor

3.3.3 Wi-Fi Module:

It is the leading IOT devices in the world in which it is very affordable and fruitful to use. The hardware connections required to connect to the wifi module are fairly easy and straight-forward but there are a couple of important things related to power:

- The ESP8266 needs 3.3V power–do not power it with 5 volts.
- The ESP8266 requires communicating via serial at 3.3V and does not have 5V tolerant inputs so you need level changeover to communicate with a 5V microcontroller like most Arduino use.

ESP8266 Wi-Fi module gives our projects access to Wi-Fi or internet. It is a very inexpensive device and makes our projects very powerful. It can communicate with any microcontroller and it is the most leading devices for the IOT projects.

Figure 3.3.3: Wi-Fi Module (ESP8266)

3.3.4 Atmega 328 Microcontroller:

The Atmega 32 bit microcontroller combines 32 kilobyte ISP flash memory with read-and write capabilities, 1 kilobyte EEPROM, 2 kilobyte SRAM, 23 general purpose input output lines, 32 general purpose working registers, 3 flexible timer/counters with compare modes, internal and external interrupts.

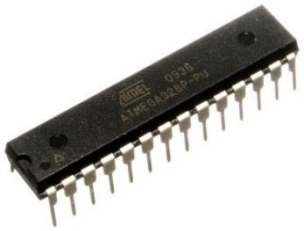

Figure 3.3.4: Microcontroller ATmega 328

3.3.5 Arduino Uno

The Arduino Uno is a microcontroller board based on the ATmega328P. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller. “Uno” means one in Italian and was chosen to mark the release of Arduino Software (IDE) 1.0.

The Uno board and version 1.0 of Arduino Software (IDE) were the reference version of Arduino, now evolved to newer releases. This board is the first in a series of USB Arduino boards

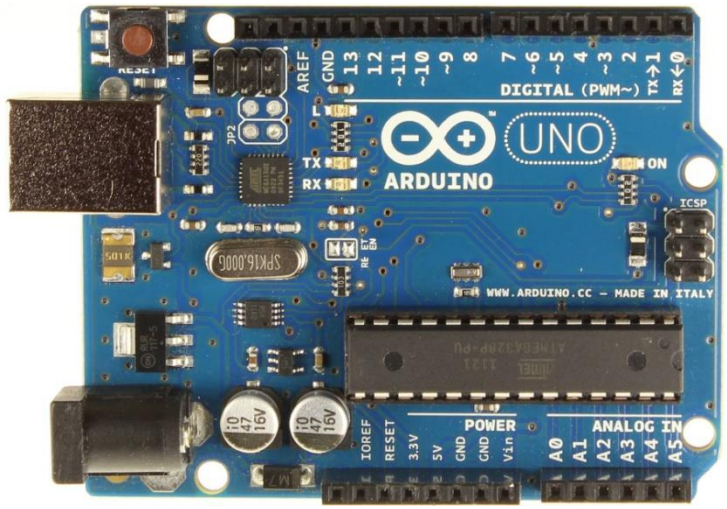

Figure 3.3.5: Arduino Uno

Table 1: Technical specifications

Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O pin	20mA

Dc Current for 3.3V pin	50mA
Flash Memory	32 KB (ATmega328P)
SRAM	2 KB (ATmega328P)
EEPROM	1KB (ATmega328P)
Clock Speed	16 MHz
Length	68.6 mm
Width	53.4 mm

3.3.6 Pin Diagram

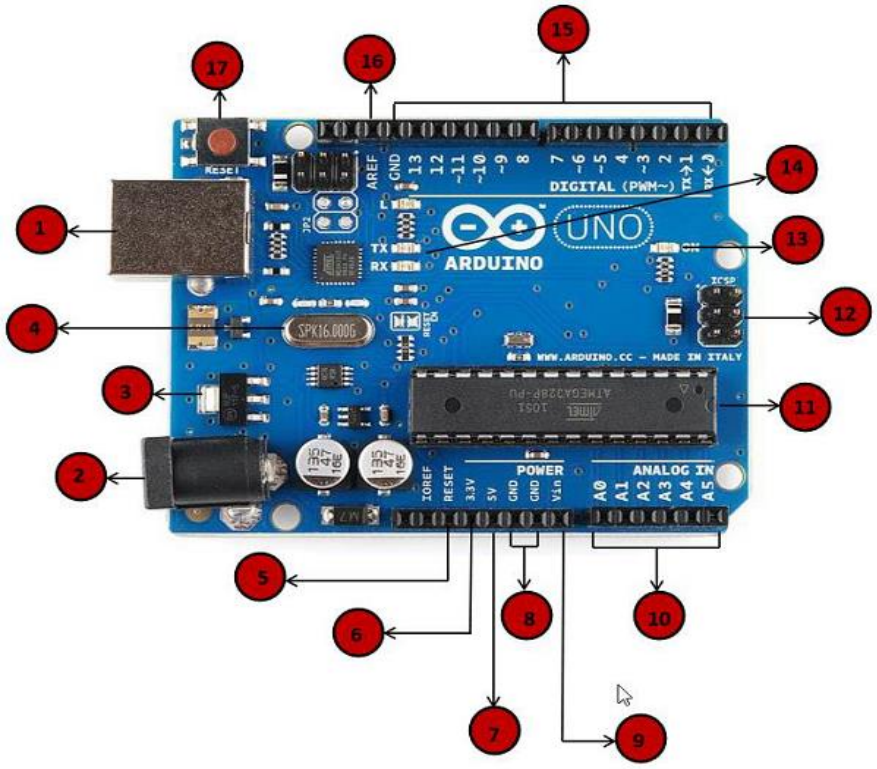

Figure 3.3.6: Pin Diagram of Arduino Uno

1

Power USB

Arduino board is high-powered by victimisation the USB cable from your laptop. All you would like to try to to is connect the USB cable to the USB affiliation (1).

2

Power (Barrel Jack)

Arduino boards are often directly high-powered from the AC mains power provide by connecting it to the Barrel Jack (2).

3

Voltage Regulator

The main perform of transformer is to regulate the voltage given to the UNO board and stabilize the DC volt utilized by the processor and alternative parts.

Crystal Oscillator

4

Crystal oscillator helps Arduino board in handling time problems. however will Arduino calculate time? the solution is, by victimisation the quartz oscillator. the amount written on high of the Arduino crystal is sixteen.000H9H. It tells North American nation that the frequency is 16,000,000 Hertz or sixteen Mega Hertz.

5,17

Arduino Reset

You can reset your Arduino board, i.e., starts your program from the start. you'll reset the UNO board in 2 ways in which. First, by exploitation the push button (17) on the board.

Pins (3.3, 5, GND, Vin)

6,7 8,9

- 3.3V (6) − Supply 3.3 output volt
- 5V (7) − Supply 5 output volt
- Most of the components used with Arduino board works fine with 3.3 volt and 5 volt.
- GND (8)(Ground) − There are several GND pins on the Arduino, any of which can be used to ground your circuit.
- Vin (9) − This pin also can be used to power the Arduino board from an external power source, like AC mains power supply.

Analog pins

10

The Arduino UNO board has five analog input pins A0 through A5. These pins will scan the signal from AN analog device just like the gas device or humidness device and convert it into a digital price that may be scan by the silicon chip.

Main microcontroller

11

Each Arduino board has its own microcontroller (11). you'll be able to assume it because the brain of your board. the most IC on the Arduino is slightly totally different from board to board.. you need to recognize what IC your board has before loading up a replacement program from the Arduino IDE.. For a lot of details concerning the IC construction and functions, you'll be able to confer with the information sheet.

ICSP pin

12

Mostly, ICSP (12) is associate degree AVR, a small programming header for the Arduino consisting of MOSI, MISO, SCK, RESET, VCC, and GND. it's typically mentioned as associate degree SPI (Serial Peripheral Interface), that may well be thought-about as associate degree "expansion" of the output. Actually, you're slaving the output device to the master of the SPI bus.

Power LED indicator

13

This diode activates once you plug your Arduino into an influence supply to indicating that your board is obtaining power properly. If this light-weight doesn't activate, then there's one thing wrong within the hardware association.

TX and RX LEDs

14

On your board, you'll notice 2 labels: TX-(Transmit) and RX-(receive). they seem in 2 places on the Arduino UNO board. 1st at the digital pins zero and one, to point the pins accountable for serial communication. Second, the Transmit and receive junction rectifier (13). The Transmit junction rectifier flashes with completely different speed whereas causing the serial knowledge.

Digital I/O

15

The Arduino UNO board has fourteen digital Input-output pins (15) (of that six give PWM (Pulse dimension Modulation) output. These pins will be organized to figure as input digital pins to browse logic values (0 or 1) or as digital output pins to drive completely different modules like LEDs, relays, etc.

16

AREF

AREF stands for Analog Reference. it's typically, wont to set Associate in Nursing external reference voltage between zero and five Volts because the higher limit for the analog input pins.

3.3.7 LCD 16X2 Display

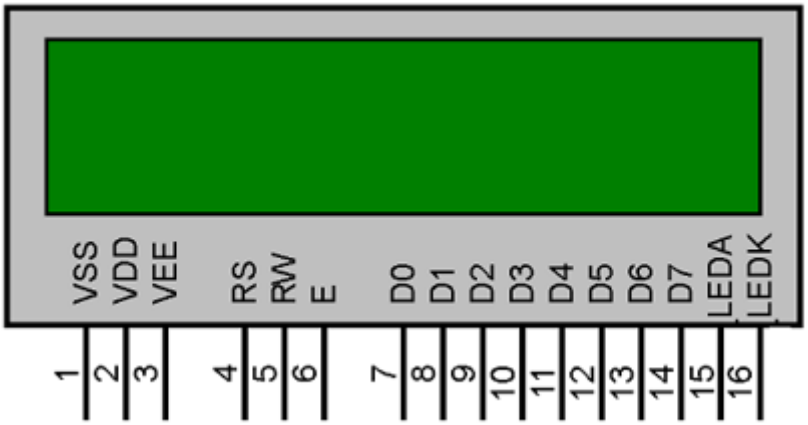

Figure 3.3.7: Pin Diagram of 16x2 LCD Display

1	Pin 1	Ground	Source Pin	This is a ground pin of LCD	Connected to the ground of the MCU/ Power source
2	Pin 2	VCC	Source Pin	This is the supply voltage pin of LCD	Connected to the supply pin of Power source
3	Pin 3	V0/VEE	Control Pin	Adjusts the contrast of the LCD.	Connected to a variable POT that can source 0-5V
4	Pin 4	Register Select	Control Pin	Toggles between Command/Data Register	Connected to a MCU pin and gets either 0 or 1. 0 -> Command Mode

5	Pin 5	Read/Write	Control Pin	Toggles the LCD between Read/Write Operation	Connected to a MCU pin and gets either 0 or 1. 0 -> Write Operation 1-> Read Operation
6	Pin 6	Enable	Control Pin	Must be held high to perform Read/Write Operation	Connected to MCU and always held high.
7	Pin 7-14	Data Bits (0-7)	Data/Command Pin	Pins used to send Command or data to the LCD.	In 4-Wire Mode Only 4 pins (0-3) is connected to MCU In 8-Wire Mode All 8 pins(0-7) are connected to MCU
8	Pin 15	LED Positive	LED Pin	Normal LED like operation to illuminate the LCD	Connected to +5V
9	Pin 16	LED Negative	LED Pin	Normal LED like operation to illuminate the LCD connected with GND.	Connected to ground

3.4 Software Requirements:

3.4.1 Arduino Integrated Development Environment (IDE)

The Arduino Integrated progress Environment - or Arduino Software (IDE) - It includes a text editor to write a program, a message field, a text area, a toolbar which has buttons for a common method and a series of menus. It connects with the Arduino and Genuino hardware to load the program and establish communication with them. These sketches are to be written in the text editor and are saved with the extension .ino.

The editor has characteristics for cutting/pasting and for searching/replacing text. The message area gives information while saving and also show errors. The console shows text output by the Arduino Software (IDE), which includes complete error messages and other information. The lowermost right-hand corner of the window shows the configured board and serial port. The toolbar buttons allow you to validate and upload programs, create, open, and save sketches, and open the serial monitor.

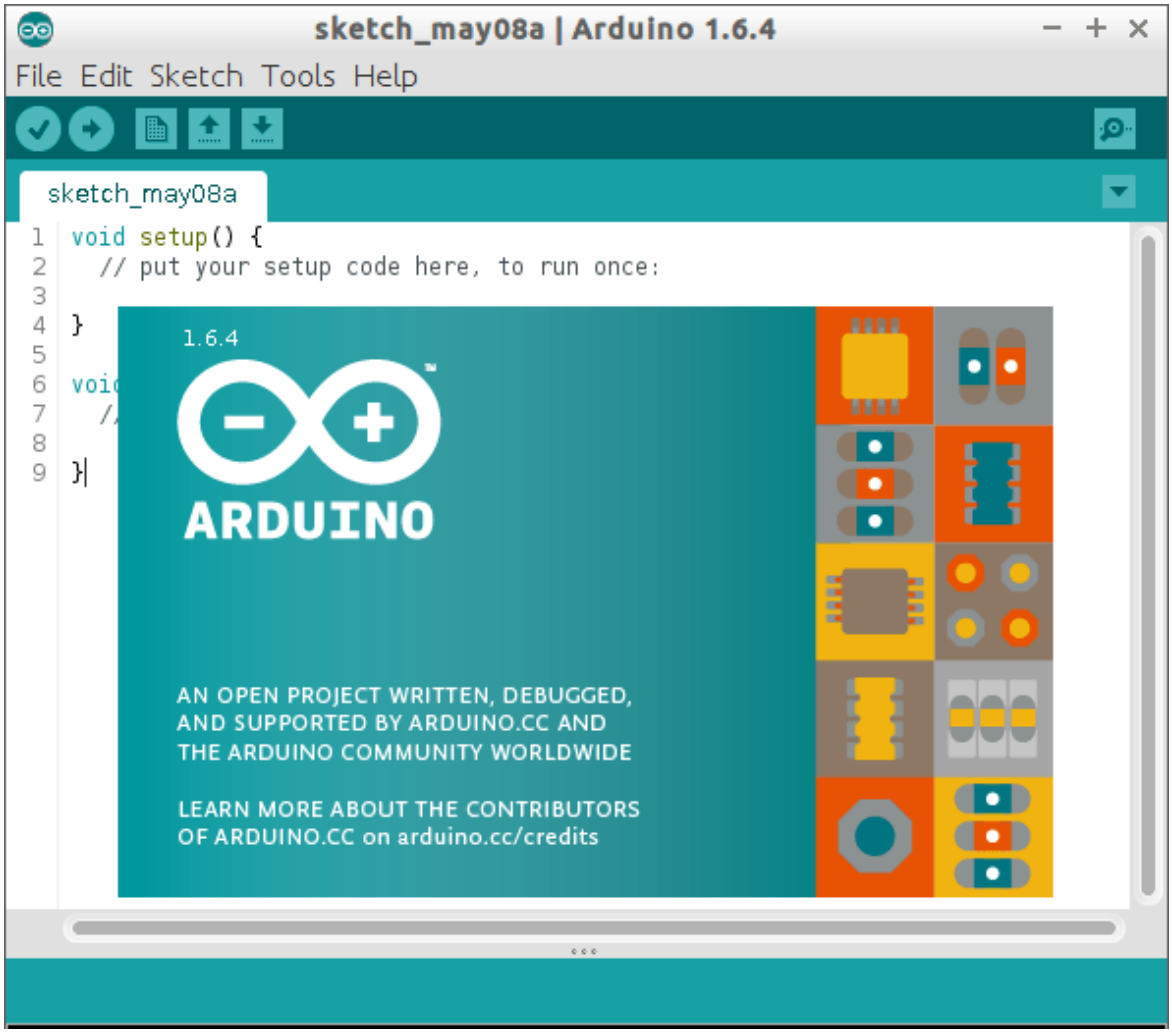

Figure 3.4.1: Arduino (IDE)

3.4.2 IOT Online Open Source Development Platform (IOT Gecko)

The IOT Development opportunity has risen steadily. The internet of things provides us various features such as control more than just digital objects and also puts forward a new dimension on the internet. Using IOTGecko we can develop our own IOT based system which can read sensor values, would operate motorized machines, monitor stuff and can do a lot more. IOTGecko cloud platform open doors to a new dimension with API support over Arduino, Microcontrollers, Raspberry Pi, and other controller boards. The IOTGecko GUI builder brings internet of things programming skills to life and customized application creator system. By using this open source internet of things development platform we can make desired IOT systems.

Features

- Efficient GUI Builder
- Process Sensor Values Over IOT
- Operate Motors Over IOT
- 2 Way Communication
- Develop Enterprise Level IOT Systems
- Online Development Support

Select your desired theme or make your own IOT Layout online. Monitor and operate your IOT system with desired GUI using IOTGecko. Select a wide variety of IOT themes ranging from Home automation to liquid sensing and pollution monitoring.

Chapter 4: Working

4.1 Circuit Diagram

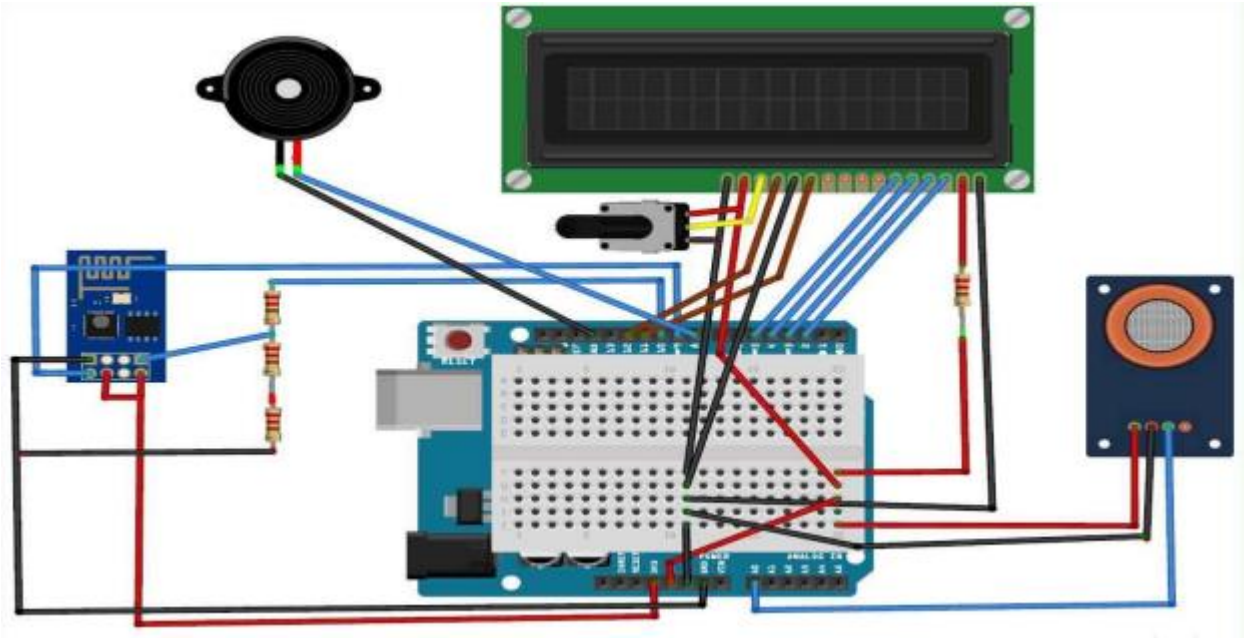

Figure 4.1: Circuit Diagram

The hardware consists of power offer used for providing offer to the elements and therefore the chip. The facility offer output vary between 5V and three.3V. The 5V is employed by sensors and processor whereas 3.3 V is needed for Wi-Fi module.

System consists of the sensors used for exploit needed knowledge from the atmosphere. Sensors used for measurement air pollutants square measure MQ7, MQ6, MQ135 and LM35 square measure used. For the measuring of sound levels a sound sensing element module- mic is employed.

The MQ135 device has the aptitude to sense Ammonia(NH3), NOx, alcohol, Benzene, smoke, carbonic acid gas and a few alternative harmful gases, therefore an ideal gas device for our Air Quality observation System. Once it's connected to Arduino it senses the gases by conduction property, and provides us level. The Pollution level in PPM (parts per million). The gas device provides the output in type of voltage levels and that we need to convert it into PPM. Thus we've used a library for the MQ135 device for changing the output in PPM that has been explained very well in "Code Explanation" section below.

The gas detector provides the worth of ninety once there was no gas gift there and also the safe level of air quality starts from 350 PPM and it mustn't exceed one thousand PPM. Once it exceeds the limit of one thousand PPM, then it starts cause Headaches, drowsiness and stagnant, stale, stuffy air and if exceeds on the far side 2000 PPM then it will cause multiplied rate and plenty of different diseases n below.

When the worth are going to be but one thousand PPM, then the show LCD digital display alphanumeric display} and webpage can display "Fresh Air". Whenever the worth can increase one thousand PPM, then the buzzer can begin beeping and also the show LCD digital display alphanumeric display and webpage can display "Poor Air, Open Windows". If it can it'll increase 2000 then the buzzer can keep beeping and also the show LCD digital display alphanumeric display and webpage will display "Danger! Move to recent Air".

Using sound detector we have a tendency to notice close sound. This board in conjunction with the mike, encompasses a little inherent electronic equipment (integrated circuit LM386), as a result of solely the mike wouldn't be ready to send signal to Arduino. The affiliation theme is extremely clean, composed of solely three pins: Vcc, GND and signal.

Within the middle of the plate, there's a potentiometer for sensitivity adjustment. The board works with 5V voltage, and therefore the signal pin ought to be connected ideally to AN analog port of Arduino, since the generated signal is variable, and so we are able to see the various levels of noise picked up by the mike.

Air pollution sensors live the standard of air whereas noise pollution sensors live the sound levels. Knowledge from these sensors square measure primarily analog signals. These analog signals square measure regenerate to its equivalent digital kind. The information will be displayed on the 16x2 liquid crystal display connected to the Arduino. To send knowledge to a distant location the information from system is distributed to the Wi-Fi module (ESP8266).

Wi-Fi module is connected to the microcontroller exploitation soap 232.The Wi-Fi module interacts with microcontroller exploitation 2 ports i.e. transmitter and receiver provided on that. The measured information is shipped from the module to any location among its vary from the info is fetched employing a laptop computer /mobile. For that we've got to offer module the Wi-Fi details to attach to net, and so offer the scientific discipline address of the web site.

4.1.1 Process of Wi-Fi Module ESP8266:

First we have to connect our wifi module to wifi router for network connectivity. After that we will access the local server, and finally send the data to web. This process is done by giving different command of wifi module. So the process is as follows:

(i) First test the wifi module by giving **AT** command it will give back in return **ok**.

(ii) After this we will give command **AT+CWMODE=mode_id**, we have used Mode id =3.

(iii) Now we have to reset our wifi module using command **AT+RST.** It is optional step.

(iv) Now we will connect our wifi module to Wi-Fi router using command **AT+CWJAP=”wifi_username”,”wifi_password”**.

(v) After that we have to generate IP address by using following command i.e. **AT+CIFSR** it will return IP address.

(vi) After that type the generated IP address in the address bar of web browser and press enter key. User can see your transmitted information in the web browser.

4.2 Flow Chart

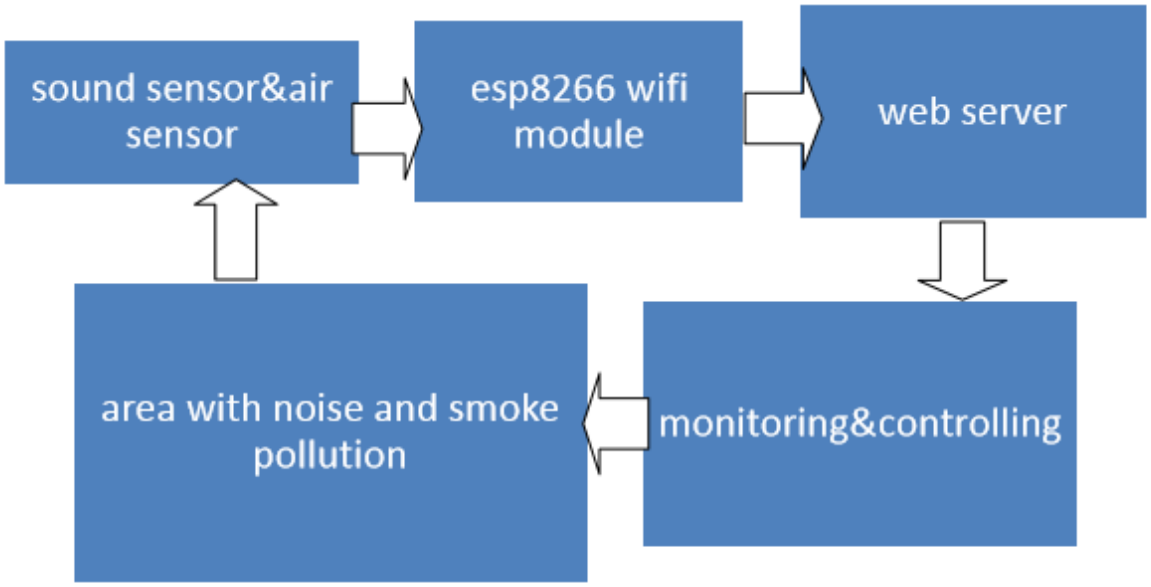

Figure 4.2: Flow Chart

Chapter 5: Conclusion

5.1 Result

After sensing the information from totally different device devices, which are placed specially square measure of interest. The perceived information will be mechanically sent to the online server, once a correct connection is established with sever device the online page provides the information regarding the intensity of sound and also the CO level variations in this specific region, wherever the embedded monitoring system is placed. To implement this want to deploy the device devices within the surroundings for collecting the information and analysis. By deploying device devices within the surroundings, we are able to bring the surroundings into world i.e. it will act with alternative objects through the network.

If air quality is less than 500 ppm then it is fresh air and if it is between 1000 ppm to 2000 ppm then it is poor air quality we should open the windows of the room and at last if it is greater than 2000 ppm then it is danger the area is very much polluted.

When we start sensing air and noise pollution the area where we placed our air and sound come under the range where air quality is in between 200ppm to 750 ppm it comes under fresh air quality region. Some of the observations and hardware implementation is given below.

5.2 Future Scope

The project is intended victimization structured modeling and is ready to supply the required results. It is with success enforced as a true Time system with bound modifications. Science is discovering or making major breakthrough in varied fields, and thus technology keeps dynamic from time to time. Going more, most of the units is fictional on one in conjunction with microcontroller so creating the system compact thereby creating the present system simpler.

To make the system applicable for real time functions parts with larger vary must be enforced. This model is any enlarged to observe the developing cities and industrial zones for pollution monitoring. To safeguard the general public health from pollution, this model provides associate economical and low price resolution for continuous observance of atmosphere.

Observations:

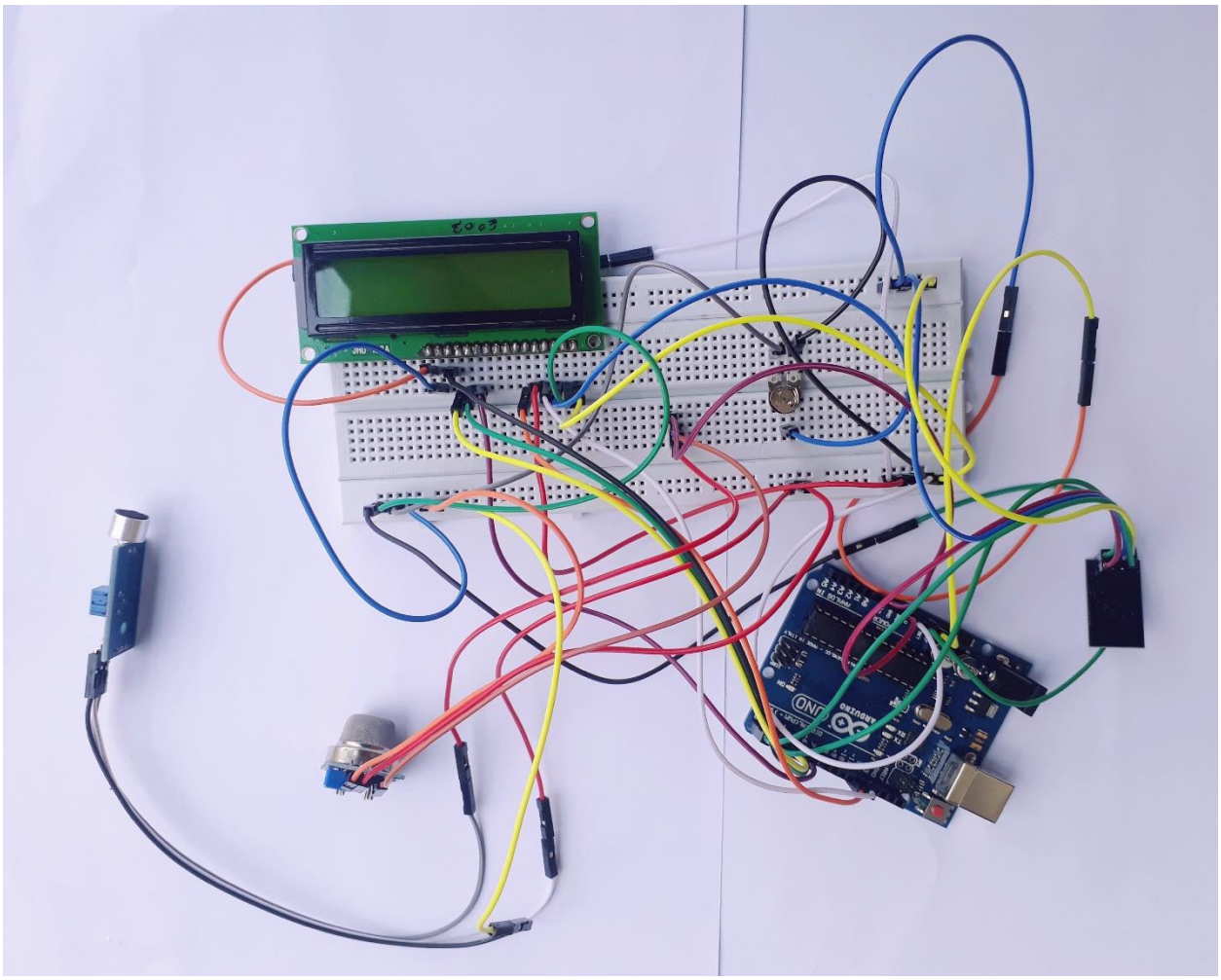

Fig 5.1.1- Hardware Implementation

Fig 5.1.2- Pollution Monitoring

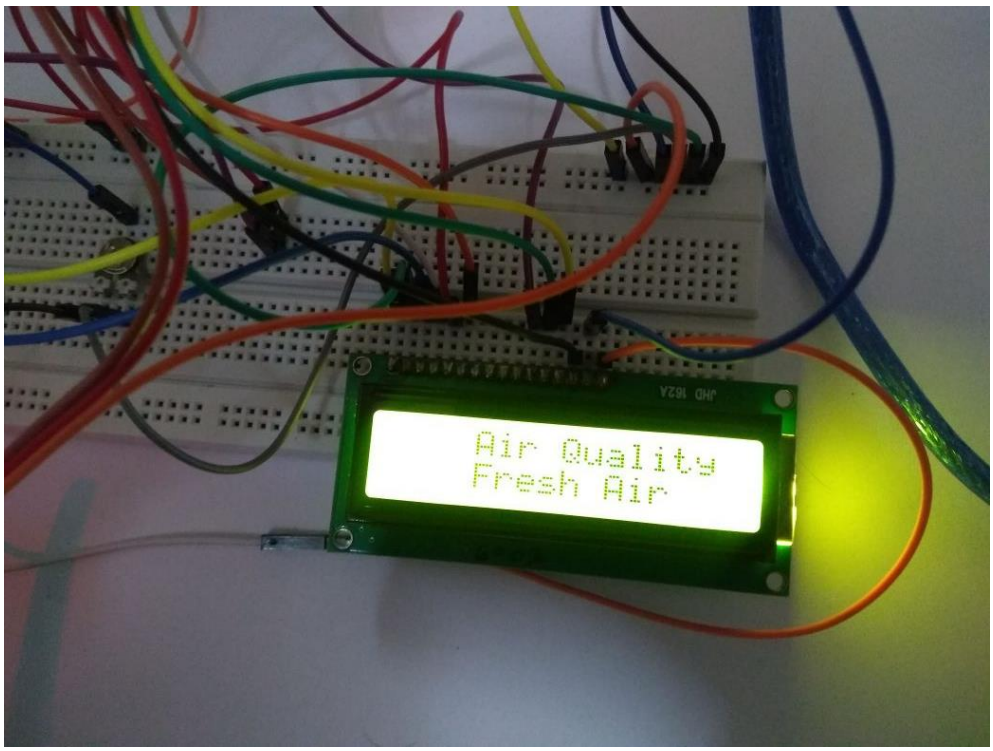

Fig 5.1.3- Air Quality

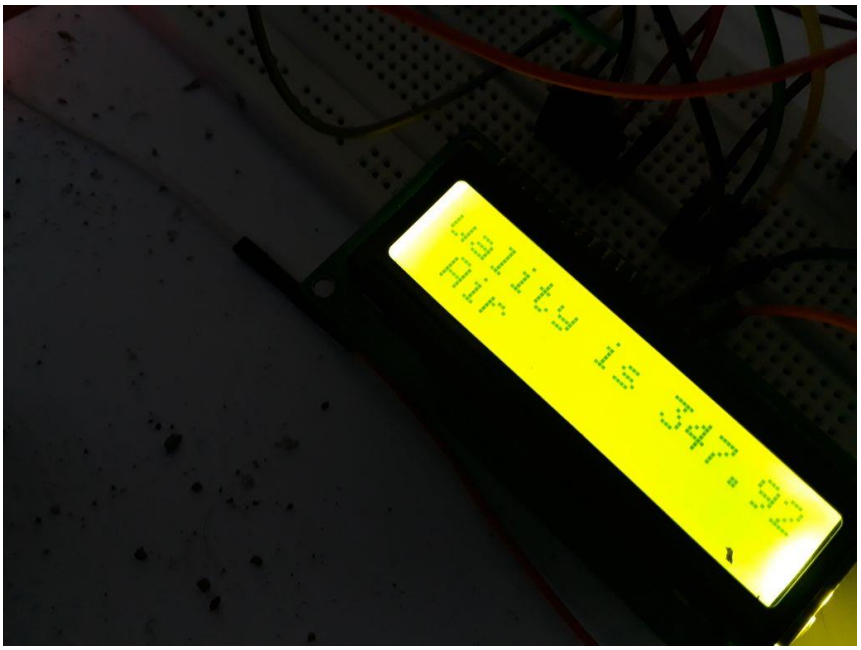

Fig 5.1.4-Air Quality 347.92 PPM

Fig 5.1.5- Air Quality 490.26 PPM

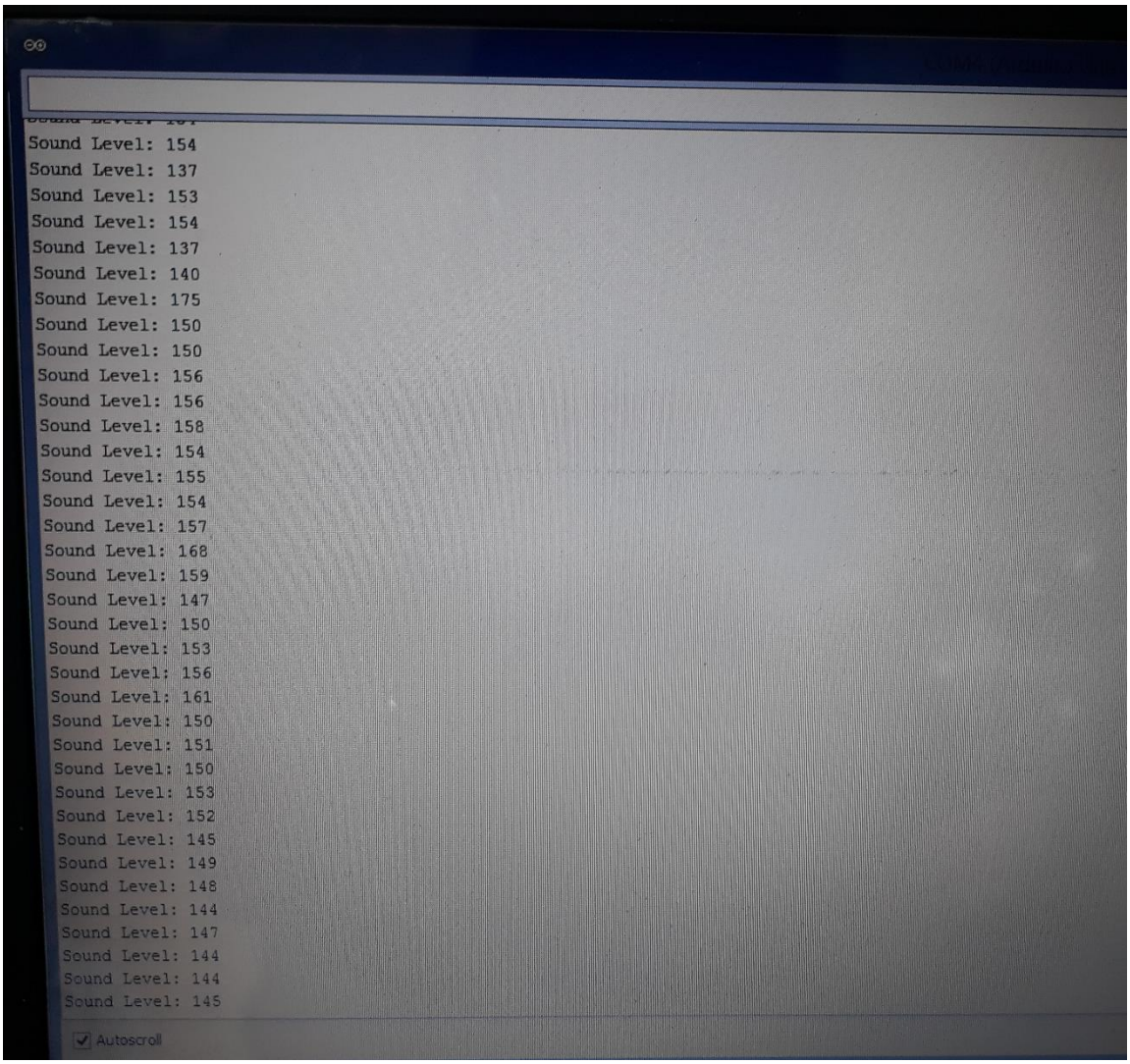

Fig 5.1.6- Sound Pollution in decible(db)

Appendix

(A) Code for Calibrating MQ135

```
#include "MQ135.h" // calibrating sensor for getting atmospheric level of CO2
void setup (){ // initialize void
Serial.begin (9600); // baud rate = 9600
}
void loop() { // it will calibrate MQ135 gas sensor
MQ135 gasSensor = MQ135 (A0); // Attach sensor to pin A0 of Arduino Uno
floatrzero = gasSensor.getRZero(); // getting value in float in serial monitor
Serial.println (rzero); // calibrate till getting value 494.63
delay(1000); // giving delay=1000 i.e after 100ms value changes in serial monitor
}
```

(B) Code for MQ135.h File

```
#ifndef MQ135_H
#define MQ135_H
#if ARDUINO >= 100
 #include "Arduino.h"
#else
 #include "WProgram.h"
#endif
/// The load resistance on the board
#define RLOAD 10.0
/// calibrating air sensor 24 hrs MQ135 and atmospheric level of CO2
#define RZERO 494.63// value after calibrating sensor MQ135 till 23 hours
```

```
/// Parameters for calculating ppm of CO2 from sensor resistance
#define PARA 116.6020682 // define parameter A
#define PARB 2.769034857 // define parameter B

/// Parameters to model temperature and humidity dependence
#define CORA 0.00035
#define CORB 0.02718
#define CORC 1.39538
#define CORD 0.0018


#define ATMOCO2 397.13 // CO2 atmospheric level

class MQ135 { // define a class MQ135
private: // Intialize it as a private class
  uint8_t _pin;

public: // all the members are public by default
MQ135(uint8_t pin);
floatgetCorrectionFactor(float t, float h); // correction factor initialize by data type float
floatgetResistance(); // getting value of resistance
floatgetCorrectedResistance(float t, float h); // getting corrected resistance
floatgetPPM(); // getting value in ppm
floatgetCorrectedPPM(float t, float h); // vorrected value in ppm
floatgetRZero();
floatgetCorrectedRZero(float t, float h);
};
#endif
```

(C) Code for Arduino Board

```
#include "MQ135.h"
#include <SoftwareSerial.h>
#define DEBUG true
SoftwareSerial esp8266(9,10); // This makes pin 9 of Arduino as RX pin and pin 10 of Arduino as the TX pin
const int sensorPin= 0;
int air_quality;
#include <LiquidCrystal.h>
LiquidCrystal lcd(12,11, 5, 4, 3, 2);
int num_Measure = 128 ; // Set the number of measurements
int pinSignal = A0; // pin connected to pin O module sound sensor
long Sound_signal; // Store the value read Sound Sensor

long sum = 0 ; // Store the total value of n
measurements long level = 0 ; // Store the average value

void setup() {
pinMode(8, OUTPUT);
lcd.begin(16,2);
lcd.setCursor (0,0);
lcd.print ("Pollution Monitoring ");
lcd.setCursor (0,1);
lcd.print ("Sensor Warming ");
delay(1000);
Serial.begin(115200);
esp8266.begin(115200); // your esp's baud rate might be different
sendData("AT+RST\r\n",2000,DEBUG); // reset module
sendData("AT+CWMODE=2\r\n",1000,DEBUG); // configure as access point
sendData("AT+CIFSR\r\n",1000,DEBUG); // get ip address
```

```
sendData("AT+CIPMUair_quality=1\r\n",1000,DEBUG); // configure for multiple connections
sendData("AT+CIPSERVER=1,80\r\n",1000,DEBUG); // turn on server on port 80
pinMode(sensorPin, INPUT); //Gas sensor will be an input to the arduino
lcd.clear();

pinMode (pinSignal, INPUT); // Set the signal pin as input
Serial.begin (9600);

}
void loop() {

 // Performs 128 signal readings

  for ( int i = 0 ; i <num_Measure; i ++)

  {

   Sound_signal = analogRead (pinSignal);
    sum =sum + Sound_signal;
  }
  level = sum / num_Measure; // Calculate the average
  value Serial.print("Sound Level: "); Serial.println (level);

  sum = 0 ; // Reset the sum of the measurement
  values delay(100);

MQ135 gasSensor = MQ135(A0);
float air_quality = gasSensor.getPPM();
if(esp8266.available()) // check if the esp is sending a message
  {
    if(esp8266.find("+IPD,"))
    {
     delay(1000);
```

```
    int connectionId = esp8266.read()-48; /* We are subtracting 48 from the output because the
read() function returns the ASCII decimal value and the first decimal number which is 0 starts
at 48*/


    String webpage = "<h1>IOT Air Pollution Monitoring System</h1>";
      webpage += "<p><h2>";
      webpage+= " Air Quality is ";
      webpage+= air_quality;
      webpage+=" PPM";
      webpage += "<p>";
    if (air_quality<=1000)
{
  webpage+= "Fresh Air";
}
else if(air_quality<=2000 && air_quality>=1000)
{
  webpage+= "Poor Air";
}
else if (air_quality>=2000 )
{
webpage+= "Danger! Move to Fresh Air"
webpage += "</h2></p></body>";


    String cipSend = "AT+CIPSEND=";
    cipSend += connectionId;
    cipSend += ",";
    cipSend +=webpage.length();
    cipSend +="\r\n";
    sendData(cipSend,1000,DEBUG);
    sendData(webpage,1000,DEBUG);
    cipSend = "AT+CIPSEND=";
```

```
    cipSend += connectionId;
    cipSend += ",";

cipSend +=webpage.length();
cipSend +="\r\n";
    String closeCommand = "AT+CIPCLOSE=";
    closeCommand+=connectionId; // append connection
    id closeCommand+="\r\n";
sendData(closeCommand,3000,DEBUG);
   }
  }
lcd.setCursor (0, 0);
lcd.print ("Air Quality is ");
lcd.print (air_quality);
lcd.print (" PPM ");
lcd.setCursor (0,1);
if (air_quality<=1000)
{
lcd.print("Fresh Air");
digitalWrite(8, LOW);
}
else if( air_quality>=1000 && air_quality<=2000 )
{
lcd.print("Poor Air, Open Windows");
digitalWrite(8, HIGH );
}
else if (air_quality>=2000 )
{
```

```
lcd.print("Danger! Move to Fresh Air");
digitalWrite(8, HIGH);   // turn the LED on
}
lcd.scrollDisplayLeft();
delay(1000);
}
String sendData(String command, const int timeout, boolean debug)
{
   String response = "";
   esp8266.print(command); // send the read character to the
   esp8266 long int time = millis();

   while( (time+timeout) > millis())
   {
     while(esp8266.available())
     {
       // The esp has data so display its output to the serial
       window char c = esp8266.read(); // read the next character.
       response+=c;

     }
   }
   if(debug)
   {
     Serial.print(response);
   }
   return response;
```

References

[1] https://www.lifewire.com/introduction-to-the-internet-of-things-817766

[2] https://circuitdigest.com/microcontroller-projects/iot-air-pollution-monitoring-using-arduino

[3] Somayya Madakam, R. Ramaswamy, "Internet of Things", Journal of Computer and Communications, 2015, 3, 164-173

[4] Anjaiah Guthi, "Implementation of an Efficient Noise and Air Pollution Monitoring System Using Internet of Things (IOT)", Vol. 5, Issue 7, July 2016 ISO 3297:2007 Certified.

[5] Palaghat Yaswanth Sai "An IOT Based Automated Noise and Air Pollution Monitoring System", Vol. 6, Issue 3, March 2017

[6] Uppugunduru Anil Kumar , G Keerthi et-al "IOT BASED NOISE AND AIR POLLUTION MONITORING SYSTEM USING RASPBERRY PI" ,Vol. 5, Issue 3,March 2017

[7] SRM.ArthiShri, NB.Keerthana, "Noise and Air Pollution Monitoring System Using IOT"

[8] L.Ezhilarasi, K.Sripriya, A .Suganya , K.Vinodhini "A SYSTEM FOR MONITORING AIR AND SOUND POLLUTION USING ARDUINO CONTROLLER WITH IOT TECHNOLOGY" , VOL 3 ISSUE 2 (2017), March 23, 2017

[9] Dr. Siva yellampalli, "IOT Based Air and Noise Pollution Monitoring in Urban and Rural Areas, Important Zones like Schools and Hospitals in Real Time" , Volume 1, Issue 5, May 2017

[10] https://www.tutorialspoint.com/arduino/arduino_board_description.htm

[11] http://m.iotgecko.com

[12] https://www.elprocus.com/mq-135-alcohol-sensor-circuit-and-working