Vehicular Pollution Monitoring Using IoT

Pramodh Narayan L[1], Shankha Mukhopadhyay[2], Utkarsh Kumar[3],
Pratyush Prasanna Sahu[4] and D Vishnu Simha[5]

[1]*Student, B. Tech, ECE, SRMIST, Chennai-89, Tamil Nadu, India*
[2]*Student, B. Tech, ECE, SRMIST, Chennai-89, Tamil Nadu, India*
[3]*Student, B. Tech, ECE, SRMIST, Chennai-89, Tamil Nadu, India*
[4]*Student, B. Tech, ECE, SRMIST, Chennai-89, Tamil Nadu, India*
[5]*Student, B. Tech, ECE, SRMIST, Chennai-89, Tamil Nadu, India*

ABSTRACT

The quality of air is degrading because of the usage of large amount of vehicles [e.g. motor, buses, cars etc.]. Air pollution increases rapidly which causes diseases and trigger, asthma attacks, cancer etc. So monitoring and controlling air quality is most important for healthy life. So we have designed a system which monitors vehicle's pollution. For that we are using IoT technique. IoT(Internet of Things) is an emerging technology in IT and Embedded system. We have used our system in vehicles. When a pollution gets detected, system warns to Regional transport office(RTO). By using mailing technique, we can send vehicle's owner details and location which is unique for every vehicle and senders mail address is based on vehicle number plate.

Keywords: *Iot, Gas sensor, Arduino, Wi-Fi module*

1. Introduction

The environmental problems are growing now days. Air pollutants from cars, buses and trucks can worsen respiratory diseases and trigger asthma attacks. Transportation is responsible for more than 50 percent of air pollution. The air pollution may lead to Chronic obstructive pulmonary disease (COPD) and escalates risk of cancer. The public health is affected due to pollution from vehicles. One of the major reasons of air pollution is emission of polluting gases from vehicles which is responsible for almost 70% of the total air pollution. To control the air pollution, the amount of air pollution needs to be monitored and vehicles causing the pollution should be identified. Internet of Things can be helpful in cities for monitoring air pollution from vehicles and also pollution data on different roads of a city can be gathered and analysed. The Internet of Things (loT) is a new concept which has attracted the attention of both academic and industry. The Internet of Things (loT) has a unique addressable devices and their virtual illustration on an internet like structure. These unique devices are connected to the web, and the data can be sent using the protocol that communicates computers to the internet. The devices can sense the environment and communicate with each other and may generally enable automatic reply to challenging scheme without human interference. This paper proposes use of loT for air pollution monitoring and identification of vehicles causing pollution.

2. Literature Survey

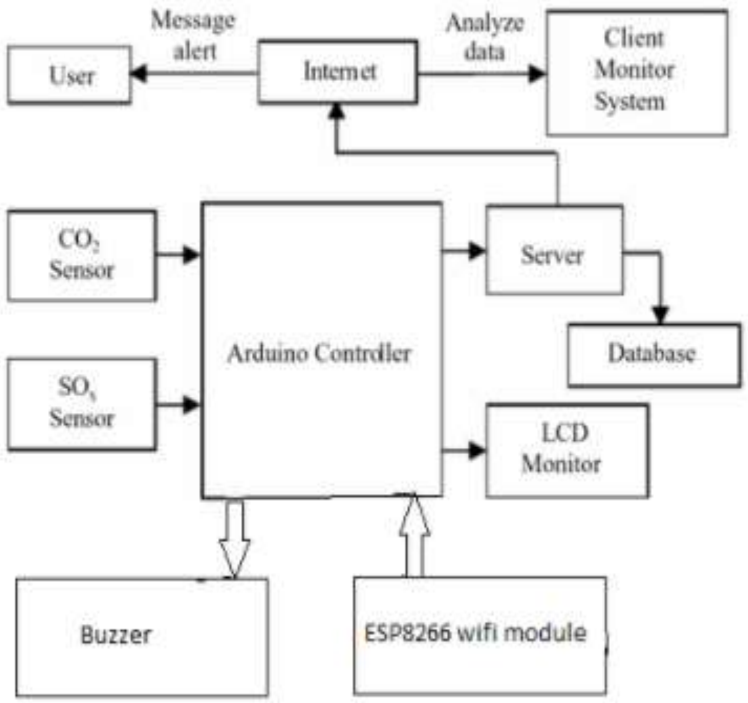
The drawbacks of the conventional monitoring instruments are their large size, heavy weight and extraordinary expensiveness. These lead to sparse deployment of the monitoring stations. In order to be effective, the locations of the monitoring stations need careful placement because the air pollution situation in urban areas is highly related to human activities (e.g. construction activities) and location-dependent (e.g., the traffic choke-points have much worse air quality than average). 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 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.

3. Proposed System

In the current scenario, one of the greatest problems that the world is facing today is pollution, increasing every year and causing grave and also irreparable damage to the Earth. In order to minimise these issues, smart emission monitoring system has been used. In this system sensors such as CO, hydrogen and gas sensors are connected to the input pins of Arduino and the values are sensed. If the sensed value goes beyond threshold value set in the program then automatically an alert message will be sent to the vehicle owner by using ESP8266 Wi-Fi enabled module in addition a buzzer is used for drivers notice. At the same time the emitted level will be monitored by the agencies of national environment by uploading the data in the web page. By using this system the owner can monitor the details about the emission level of his own vehicle.

4. Block Diagram



5. Hardware Used

The following hardware part is used in this project is explained as follows

5.1 Arduino UNO

The Arduino Uno is a microcontroller board based on the ATmega328 (datasheet). It has 14 digital input/output pins (of which can be used as PWM outputs), analog inputs, A 16MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Uno differs from all preceding boards in that it does not use the FTDI USB –to-serial driver chip. Instead, it features the Atmega8U2 programmed as a USB –to-serial converter.

5.2 Wi-Fi Module

The ESP8266 Wi-Fi Module is self-contained SOC with integrated TCP/IP protocol stack that can give any microcontroller access to your Wi-Fi network. The ESP8266 is capable of either hosting an application or offloading all Wi-Fi networking functions from another application processor. This module has a powerful enough on-board processing and storage capability that allows it to be integrated with the sensors. Its high degree of on the chip integration allows for minimal external circuitry, including the front-end module, is designed to occupy minimal PCB area

Fig 5.2 Wi-Fi Module

5.3 Gas sensors

5.3.1 CO Sensors (Carbon Monoxide)

MQ-7 gas sensor composed by micro AL2O3 ceramic tube, Tin Dioxide (SnO2) sensitive layer, measuring electrode and heater are fixed into a crust made by plastic and stainless steel net. High sensitivity to carbon monoxide. Stable and long life. They are used in gas detecting equipment for carbon monoxide (CO)in family and industry or car.

Fig 5.3.2 MQ135

5.4 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 Air and Humidity in PPM. Fig shows LCD (16*2).

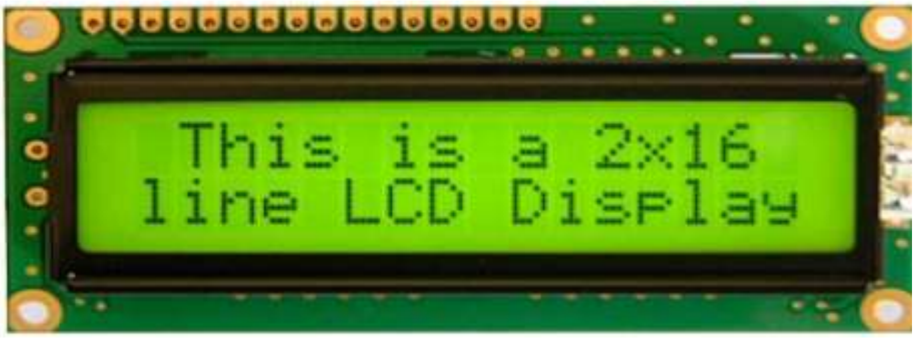


Fig 5.4 LCD

6. Software used

The following software part is used in this project is explained as follows.

6.1 Arduino IDE

Arduino is an open-source platform used for building electronics projects. Arduino consists of both a physical programmable circuit board (often referred to as a microcontroller) and a piece of software, or IDE (Integrated Development Environment) that runs on your computer, used to write and upload computer code to the physical board.

The Arduino platform has become quite popular with people just starting out with electronics, and for good reason. Unlike most previous programmable circuit boards, the Arduino does not need a separate piece of hardware (called a programmer) in order to load new code onto the board – you can simply use a USB cable. Additionally, the Arduino IDE uses a simplified version of C++, making it easier to learn to program.

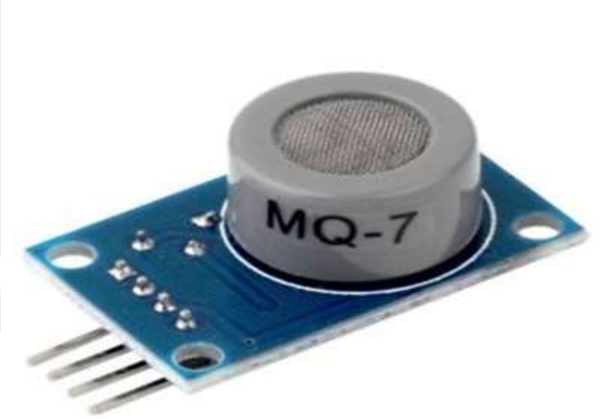
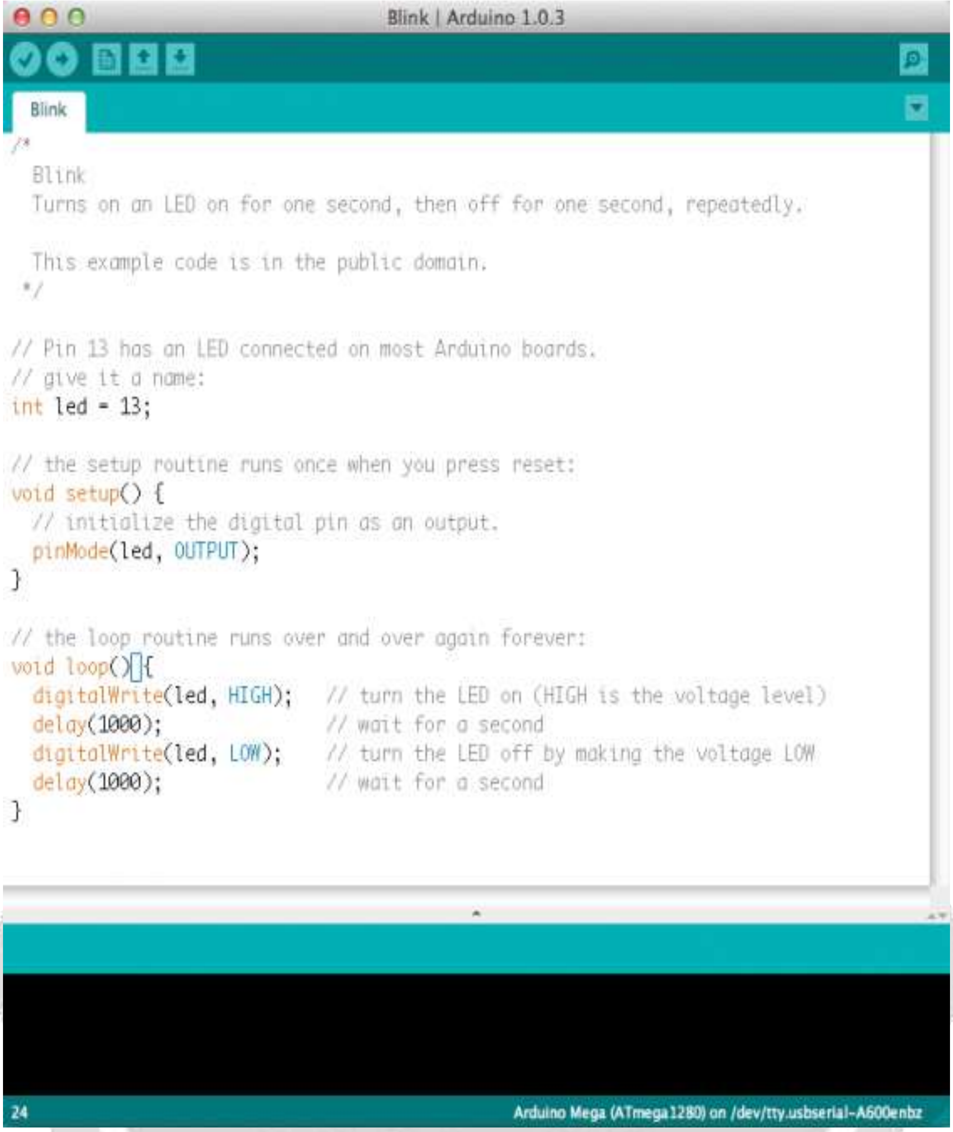


Fig 5.3.1 MQ-7

5.3.2 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.



```
/*
  Blink
  Turns on an LED on for one second, then off for one second, repeatedly.

  This example code is in the public domain.
 */

// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);               // wait for a second
  digitalWrite(led, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);               // wait for a second
}
```

6.1 Arduino IDE

7. Working Explanation

The MQ135 sensor can sense NH3, NOx, alcohol, Benzene, smoke, CO2 and some other gases, so it is perfect gas sensor for our Air Quality Monitoring Project. When we will connect it to Arduino then it will sense the gases, and we will get the Pollution level in PPM (parts per million). MQ135 gas sensor gives the output in form of voltage levels and we need to convert it into PPM. So for converting the output in PPM, here we have used a library for MQ135 sensor, it is explained in detail in “Code Explanation” section below.

Sensor was giving us value of 90 when there was no gas near it and the safe level of air quality is 350 PPM and it should not exceed 1000 PPM. When it exceeds the limit of 1000 PPM, then it starts cause Headaches, sleepiness and stagnant, stale, stuffy air and if exceeds beyond 2000 PPM then it can cause increased heart rate and many other diseases.

When the value will be less than 1000 PPM, then the LCD and webpage will display “Fresh Air”. Whenever the value will increase 1000 PPM, then the buzzer will start beeping and the LCD and webpage will display “Poor Air, Open Windows”. If it will increase 2000 then the buzzer will keep beeping and the LCD and webpage will display “Danger! Move to fresh Air”.

8. Code Explanation

Here we are using the library of MQ135, Using the library you can directly get the PPM values, by just using the below two lines:

```
MQ135 gasSensor = MQ135(A0);

float air_quality = gasSensor.getPPM();
```

But before that we need to calibrate the MQ135 sensor, for calibrating the sensor upload the below given code and let it run for 12 to 24 hours and then get the *RZERO* value.

```
#include "MQ135.h"

void setup (){

Serial.begin (9600);

}

void loop() {

MQ135 gasSensor = MQ135(A0); // Attach sensor to pin A0

float rzero = gasSensor.getRZero();

Serial.println (rzero);

delay(1000);

}
```

In the code, first of all we have defined the libraries and the variables for the Gas sensor and the LCD. By using the Software Serial Library, we can make any digital pin as TX and RX pin. In this code, we have made Pin 9 as the RX pin and the pin 10 as the TX pin for the ESP8266. Then we have included the library for the LCD and have defined the pins for the same. We have also defined two more variables: one for the sensor analog pin and other for storing *air_quality* value.

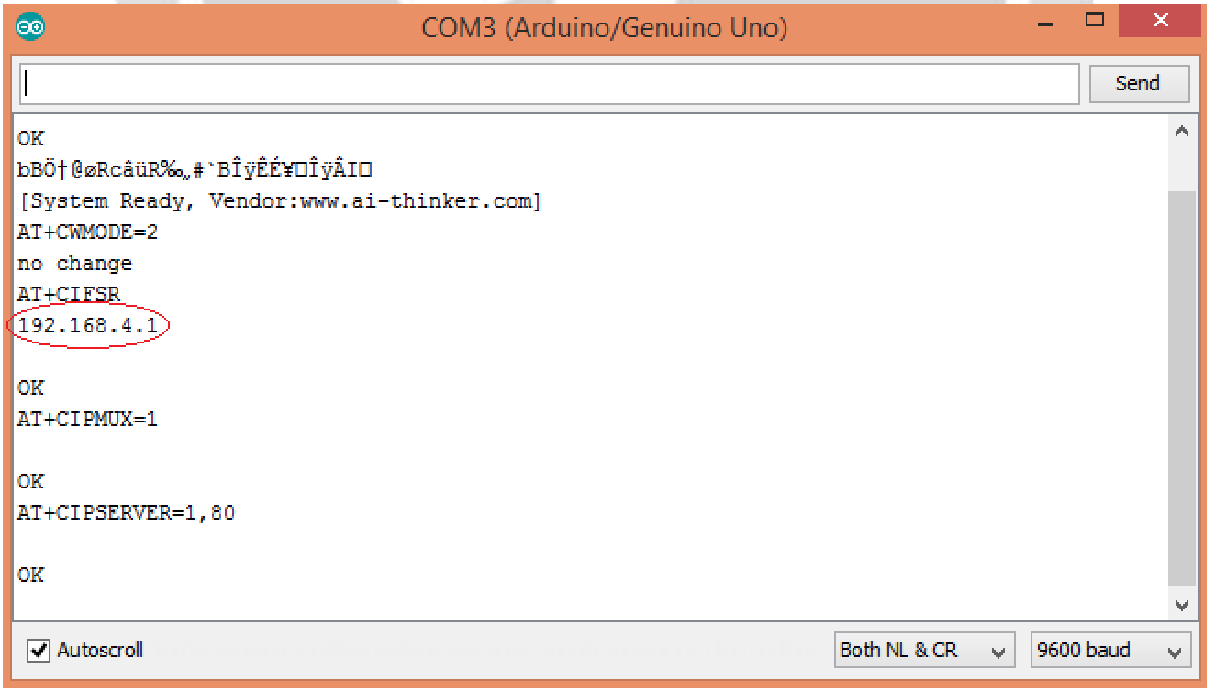
```
#include <SoftwareSerial.h>

#define DEBUG true

SoftwareSerial esp8266(9,10);

#include <LiquidCrystal.h>

LiquidCrystal lcd(12,11, 5, 4, 3, 2);

const int sensorPin= 0;

int air quality;
```

The following code will print the data on the LCD. We have applied various conditions for checking air quality, and LCD will print the messages according to conditions and buzzer will also beep if the pollution goes beyond 1000 PPM.

```
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);
```

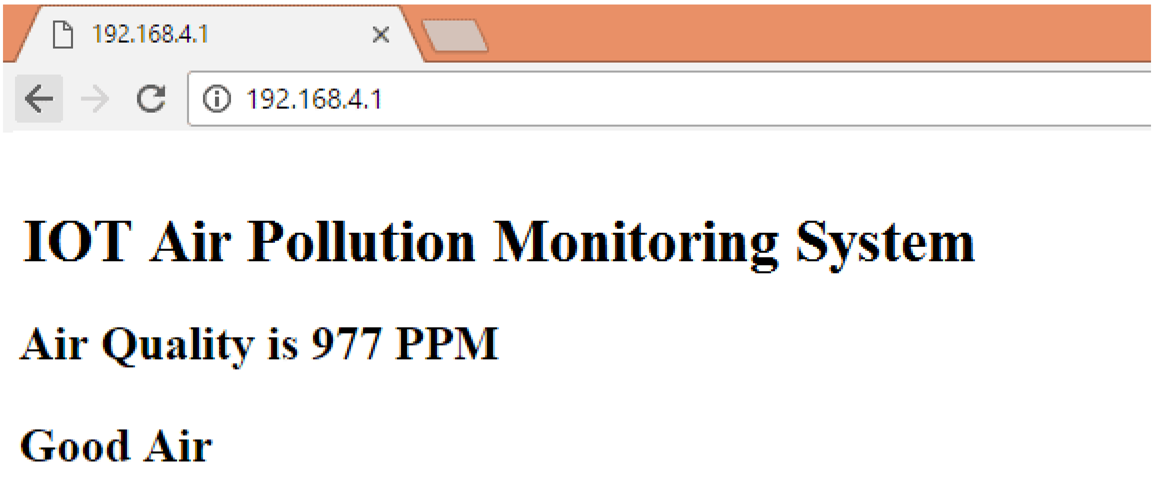
9. Testing and Output of the Project

Before uploading the code, make sure that you are connected to the Wi-Fi of your ESP8266 device. After uploading, open the serial monitor and it will show the IP address like shown below.



```
OK
bBÖ†@øRcâüR‰„#`BÎÿÊÉ¥□ÎÿÂI□
[System Ready, Vendor:www.ai-thinker.com]
AT+CWMODE=2
no change
AT+CIFSR
192.168.4.1

OK
AT+CIPMUX=1

OK
AT+CIPSERVER=1,80

OK
```

Type this IP address in your browser, it will show you the output as shown below. You will have to refresh the page again if you want to see the current Air Quality Value in PPM.



IOT Air Pollution Monitoring System

Air Quality is 977 PPM

Good Air

We have setup a local server to demonstrate its working. But to monitor the air quality from anywhere in the world you need to forward the port 80(used for HTTP or internet) to your local or private IP address (192.168*) of your device .After port forwarding all the incoming connections will be forwarded to this local address and you can open above shown webpage by just entering the public IP address of your internet from anywhere . You can forward the port by logging into your router (192.168.1.1) and find the option to setup the port forwarding.

10. Reference

1. A. M. Gibb, New Media Art, design, and the Arduino microcontroller. A malleable tool. PhD thesis, Pratt Institute 2010
2. Evaluating Critical Security issues of the IoT world: Present and Future Challenges Mario Fructaci: Pasquale Pace ; Gianluca Aloi ; Giancarlo Fortino, Publication Year: 2018
3. A survey on Internet Of Things: Architecture Enabling Technologies, Security and Privacy and applications – Jie Lin; Wei Yu; Nan Zhang; Xinyu Yang; Hanlin Zhang; Wei Zhao, Publication Year-2017
4. http://www.pollutionissues.com/Ve-Z/Vehicular-Pollution.html
5. https://tutorialspoint.com/internet_of_things/index.html