**AIR POLLUTION MONITORING IN VEHICLES**

**USING IOT**

**A PROJECT REPORT**

***Submitted by***

**P.NIRESH GOMES (210916104018)**

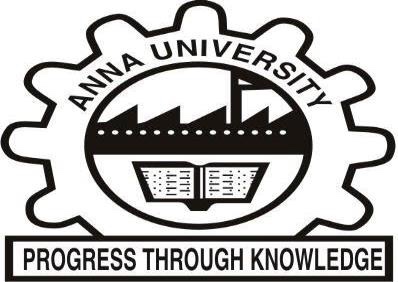
**V.ROHITH (210916104024)**

**J.FEBIAN (210916104007)**

***In partial fulfilment for the award of the degree of***

**BACHELOR OF ENGINEERING IN**

**COMPUTER SCIENCE AND ENGINEERING**



**LOYOLA INSTITUTE OF TECHNOLOGY, CHENNAI**

**ANNA UNIVERSITY: CHENNAI 600 025**

**MARCH 2020**

**ANNA UNIVERSITY::CHENNAI 600 025**

**BONAFIDE CERTIFICATE**

Certified that this project report “**AIR POLLUTION MONITORING IN VEHICLES USING IOT** ” is the bonafide work of “ **P.NIRESH GOMES (210916104018)**”, “**V.ROHITH(210916104024)**”,“**J.FEBIAN (210916104007)**”, who carried out the project work under my supervision.

**SIGNATURE SIGNATURE**

**DR.G.BHUVANESWARI.,Ph.D.**, **Ms.K.GEETHA RANI,M.TECH.,**

**HEAD OF THE DEPARTMENT, ASSISTANT PROFESSOR,**

Computer Science and Engineering,Computer Science and Engineering,

Loyola Institute of Technology, Loyola Institute of Technology

Palanchur, Palanchur,

Chennai – 600 123. Chennai – 600 123.

**Submitted for the ANNA UNIVERSITY examination held on \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.**

**INTERNAL EXAMINER EXTERNAL EXAMINER**

**ACKNOWLEDGEMENT**

A project of this magnitude and nature requires kind cooperation and support

from many for successful completion. We wish to express sincere thanks to all of

those who were involved in the completion of the project.

It is our immense pleasure to express our deep sense of gratitude to our honourable chairman **Rev. Fr. Dr. J.E. ARULRAJ,** for providing such an excellent lab facility, our correspondent **Rev. Sr. I. AROCKIA MARY,**  for continuous motivation to complete the project.

We are extremely thankful to our beloved and dynamic principal **Dr. SUJATHA ANAND, M.TECH., Ph.D.**, for having given us an opportunity todemonstrate our innovative abilities.

We would like to thank our head, Department of Computer Science and

Engineering, **Dr.G.BHUVANESWARI, Ph.D.**, for the necessary help to complete the

project successfully.

We take this moment as a great opportunity to thank our project coordinator

**Mrs.P.RAMA,M.TECH.,(Ph.D).**, Assistant Professor, Department of Computer Science and

Engineering and our guide **Ms.K. GEETHA RANI, M.TECH.,** Assistant Professor,

Department ofComputer Science and Engineering, for giving us their constant

encouragement,support and valuable guidance to us during the project.

Last but not least, we are very grateful to our parents, friends and well-

wishers for their constant encouragement during the project.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TABLE OF CONTENTS** | | | | |
| **CHAPTER NO.** | **TITLE** | | | **PAGE NO.** |
|  | ABSTRACT | | | 6 |
|  | LIST OF FIGURES | | | 7 |
|  | LIST OF ABBREVIATIONS | | | 8 |
| **1** | **INTRODUCTION** | | | 9 |
| **2** | **LITERATURE REVIEW** | | | 10 |
| **3** | **SYSTEM DESIGN** | | |  |
|  | 3.1 | EXISTING SYSTEM | | 16 |
|  | 3.2 | PROPOSED SYSTEM | | 17 |
|  | 3.3 | HARDWARE SOFTWARE  SPECIFICATION | | 18 |
| **4** |  | **SYSTEM DESCRIPTION** | |  |
|  | **4.1** | **MODULES LIST** | |  |
|  |  | 4.1.1 | INTRODUCTION TO ARDUINO | 19 |
|  |  | 4.1.2 | IDE FOR DEVELOPMENT | 21 |
|  |  | 4.1.3 | MQ135 GAS SENSOR | 24 |
|  |  | 4.1.4 | BLUETOOTH | 26 |
|  |  | 4.1.5 | SWITCH | 27 |
|  |  | 4.1.6 | ESP-BASED NODEMCU | 28 |
|  |  | 4.1.7 | GSM | 30 |
|  |  | 4.1.8 | MOTOR DRIVER IC | 31 |
|  |  | 4.1.9 | LCD | 31 |
|  |  | 4.1.10 | EMBEDDED C. | 32 |
|  | **4.2** | **BLOCK DIAGRAM** | | 33 |
| **5** | **RESULT ANALYSIS** | | |  |
|  | 5.1 | COMPARISON CHART | | 35 |
| **6** | **CONCLUSION AND FUTURE WORK** | | | 38 |
| **7** | **APPENDICES** | | |  |
|  | 7.1 | CODING | | 39 |
|  | 7.2 | SCREENSHOTS | | 49 |
|  | **REFERENCES** | | | 52 |

**ABSTRACT**

The growing air pollution in vehicles is one of the serious issues these days. As the pollution is increasing it is giving rise to number of diseases. So, it is essential to control the pollution to ensure healthy living and better future. The Air Pollution Monitoring device in the vehicle can be accessed by the each and every people curious about the pollution level. The device can be installed through a vehicle which will show the pollution level and can take necessary actions on situation and can control

|  |  |  |
| --- | --- | --- |
| **Fig.No** | **Figure Name** | **Page No** |
| 4.1.1 | ARDUINO UNO DIAGRAM | 20 |
| 4.1.2 | ARDUINO IDE | 23 |
| 4.1.3 | MQ135 GAS SENSOR | 24 |
| 4.1.6 | ESP-BASED NODEMCU | 29 |
| 4.1.7 | GSM | 30 |
| 4.1.9 | LCD | 32 |
| 4.2 | BLOCK DIAGRAM | 33 |

**LIST OF ABBREVIATIONS**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **WORD** | **ABBREVATIONS** |
| 1 | LCD | LIQUID CRYSTAL DISPLAY |
| 2 | GSM | GLOBAL SYSTEM FOR MOBILE |
| 3 | IDE | INTEGRATED DEVELOPMENT ENVIRONMENT |
| 4 | PPM | PARTS PER MILLION |

**CHAPTER-1**

**INTRODUCTION**

* 1. **OBJECTIVE**

Air pollution emerged in many parts of the world as a result of explosive industrial growth. Road transport is also one of the major contributors of air pollution which contribute to climate change that has perilous domestic and global consequences. Generation and transport of pollutant materials are governed not only by the distributions of their sources but also by the dynamics of the atmosphere. Pollutant clouds are sometimes observed traveling along the wind directions. To understand the involved processes in more detail we need more thorough data on the spreads of fine-grain pollutants and their variations with time. An air pollution monitoring system that is comprehensive in terms of spatial and pollutant coverage and is relatively inexpensive and autonomous is the priority.

Vehicles are a major source of pollution in urban areas. The drastic increase in number of vehicles has also resulted in a significant increase in the emission load of various pollutants. Our proposed system provides the solution to this problem with the help of internet of things. The Internet of Things is the internetworking of physical devices, buildings, vehicles and other items embedded with electronics, sensors, software, actuators and network connectivity that enable each object to collect and exchange Data. In this paper, the semiconductor sensors have been used to detect the pollutant level.

CHAPTER-2

LITERATURE REVIEW

**[2.1] Siva Shankar Chandrasekaran, Sudharshan Muthukumar and Sabeshkumar Rajendran,” Automated Control System for Air Pollution Detection in Vehicles”, IEEE conference, 2013,** **DOI 10.1109/ISMS.2013.94**

This paper aims at using those semi-conductor sensors at the emission outlet of vehicle which defect the level of pollutants and also indicate this level with a meter.

Vehicles have become an integral part of every one's life. Situations and circumstances demand the usage of vehicles in this fast paced urban life. As a coin has two sides, this has its own effects, one of the main side effects being air pollution. Every vehicle will have emission but the problem occurs when it is beyond the standardized values. The primary reason for this breach of emission level being the incomplete combustion of fuel supplied to engine, which is due to the improper maintenance of vehicles. This emission from vehicles cannot be completely avoided but, it definitely can be controlled.

When the pollution/ emission level shoots beyond the already set threshold level, there will be a buzz in the vehicle to indicate that the limit has been breached and the vehicle will stop after a certain period of time, a cushion time given for the driver to park his/her vehicle. During this time period, the GPS starts locating the nearest service stations.

After the timer runs out, the fuel supplied to the engine will be cut-off and the vehicle has to be towed to the mechanic or to the nearest service station. The synchronization and execution of the entire process is monitored and controlled by a micro controller**.**

This paper, when augmented as a real time project, will benefit the society and help in reducing the air pollution.

**[2.2] V.S.Harilakshmi, P.Arockia Jansi Rani”Intelligent Vehicle Counter – A Road to Sustainable Development and Pollution Prevention(P2) “,IEEE Conference,2016**

This paper proposes an effective system to detect and count dynamic vehicles. The proposed vehicle counter can adapt to the complicated road traffic of non-lane and chaotic network of many developing countries.

Increased vehicle traffic results in many problems such as traffic accidents, traffic congestion and so on. Amongst all, traffic induced air pollution is the most ravaging illness in most cities. This traffic induced air pollution is damaging the environment at an alarming rate. Vehicle exhaust is a major source of air pollution.

Vehicle detection and counting plays an important role in effectively estimating the amount of traffic. Automatic vehicle counter gives an estimation of the amount of traffic. This estimate enables the pollution control board to take relative measures to curb the harmful effects of traffic induced air pollution on the environment and on human community. Thus, paves a road to sustainable development and pollution prevention (P2).

**[2.3] Mahesh A. Rakhonde,Prof. Dr. S. A. Khoj,Prof. R. D. Komati.”Vehicle Collision Detection and Avoidance with Pollution Monitoring System Using IoT”,IEEE conference, 2018**

The increase in population leads to increment in pollution and accident as well. This paper is based on enhancement in the smart vehicle system. The main objective is to detect accidents in real time and minimize the response time of medical help.. The proposed system is useful in reducing the vehicular accidents and pollution monitoring will help to know the environmental status.

Different units implemented in this paper which enhance the vehicular system. The main objective is to detect accidents in real time and minimize the response time of medical help. For accident avoidance, tire pressure is measured whereas in accident detection is implemented with the help of node MCU.MQ7 is used in order to monitor the pollution.

The proposed system is useful in reducing the vehicular accidents and pollution monitoring will help to know the environmental status.

**[2.4] Afrah Mohammad,” GSM Based Air Quality Monitoring and Analysis Using Wireless Sensor Node “,IJIRSET, Vol. 5, Issue 6, June 2016, DOI:10.15680/IJIRSET.2015.0506131**

The main objective of this project is to devise a simple low cost air pollution monitoring system based on microcontroller using wireless technology which detects presence of various gases like CO2, SO, NO, etc., and parameters like humidity, temperature, etc., displays it on LCD and forwards it to remote user. This project is developed by using PIC 16F877A Microcontroller, SIM 900 GSM Module, JHD204A LCD display and gas sensors. The advantage of using GSM based technology is that GSM based communication network is distributed over a large area and have almost reached to every part of the world.

GSM technology also does provide users with high quality signal and channels, giving them access to high quality digital communication at affordable cost.

This embedded system can be useful for anyone who wish to monitor the quality of air at a location without being physically present there. The main advantages of the research are that the system may be able to collect the pollution levels throughout the day and also the data so collected may serve as a data base which can be used for various analysis as and when required.

The system may offer pollutant levels of a particular industry and this estimation may serve as an ready reckoner to the government for allowing or disallowing a particular industry to be set up in a particular area.

**[2.5] V.S.Revathy, K.Ganesan, K.Rohini, S.Tamil Chindhu, T.Boobalan, “Air Pollution Monitoring System” , Volume 11, Issue 2, IOSR-JECE, MARCH-2016 , DOI: 10.9790/2834-1102022740.**

Air pollution has significant influence on the concentration of constituents in the atmosphere leading to effects like global warming and acid rains.

This paper attempts to develop an effective solution for pollution monitoring using wireless sensor networks (WSN) on a real time basis namely real time wireless air pollution monitoring system.

Commercially available discrete gas sensors for sensing concentration of gases like CO and CO2 are calibrated using appropriate calibration technologies.These pre-calibrated gas sensors are then integrated with the wireless sensor motes for field deployment at the campus and the Coimbatore city using multi hop data aggregation algorithm

A light weight middleware and a web interface to view the live pollution data in the form of numbers and charts from the test beds was developed and made available from anywhere on the internet Other parameters like temperature and humidity were also sensed along with gas concentrations to enable data analysis through data fusion techniques

Experimentation carried out using the developed wireless air pollution monitoring system under different physical conditions show that the system collects reliable source of real time fine-grain pollution data.

**[2.6] Piyush Patil,” Smart IoT Based System For Vehicle Noise And Pollution Monitoring “,IEEE Conference,2017, 978-1-5090-4257-9/17**

According to population Reference Bureau, the current world population is 7.4 Billion. At present, a total number of the vehicle is 1.2 Billion according to survey and it will be 2 Billion up to 2035.Transportation contributed more than half nitrogen oxide and carbon monoxide and a quarter of the hydrocarbon emitted in air.

Due to heavier traffic and powerful engine, noise level in cities is rapidly increasing. Our proposed system provides the solution to these problems with the help of internet of things. Our system Monitor the pollution and noise created by vehicle and if any vehicle crosses its threshold value then it will get reported to the traffic department and agencies of national environment.

CHAPTER-3

SYSTEM DESIGN

* 1. EXISTING METHOD
* In existing system, manually checked the gas emission of the vehicle using vehicle emission test machine in the service center.
* People do pollution test only when the vehicle is given for service.

**LIMITATION**

* People use their vehicles without testing pollution.
* Chance of engine failure due to non-maintenance

* 1. PROPOSAL METHOD
* In proposed system, sensor is used to check the gas emission by the vehicle.
* Automatically check every time when the vehicle is used

**ADVANTAGE**

* Vehicle cannot be operated if gas from vehicle exhaust is high.
* Message is sent to the vehicle recovery station
  1. **HARDWARE AND SOFTWARE REQUIREMENTS**

**HARDWARE REQUIREMENT**

* ARDUINO UNO
* MQ135- AIR QUALITY
* BLUETOOTH
* SWITCH
* ROBOT CHASE
* IOT
* GSM
* MOTOR DRIVER
* DC MOTORS (2 Nos)
* LCD
* POWER SUPPLY

**SOFTWARE REQUIREMENTS**

* ARDUINO IDE
* EMBEDDED C

**CHAPTER-4**

**SYSTEM DESCRIPTION**

**4.1 MODULE DESCRIPTION**

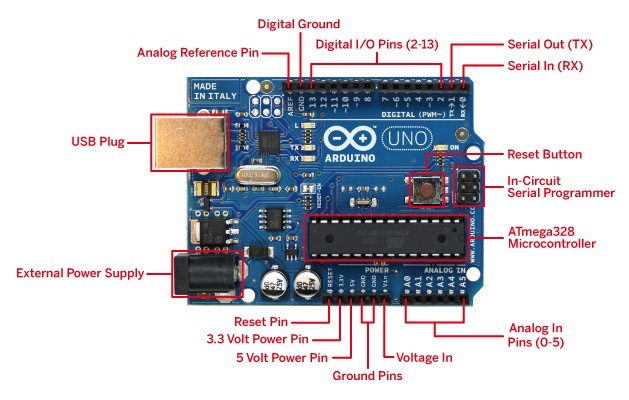
**4.1.1 INTRODUCTION TO ARDUINO**

Arduino is an open-source electronics platform based on easy-to-use hardware and software. [Arduino boards](https://www.arduino.cc/en/Main/Products) are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board.

Over the years Arduino has been the brain of thousands of projects, from everyday objects to complex scientific instruments. A worldwide community of makers - students, hobbyists, artists, programmers, and professionals - has gathered around this open-source platform, their contributions have added up to an incredible amount of [accessible knowledge](http://forum.arduino.cc/) that can be of great help to novices and experts alike.

Arduino was born at the Ivrea Interaction Design Institute as an easy tool for fast prototyping, aimed at students without a background in electronics and programming. As soon as it reached a wider community, the Arduino board started changing to adapt to new needs and challenges, differentiating its offer from simple 8-bit boards to products for IoT applications, wearable, 3D printing, and embedded environments. All Arduino boards are completely open-source, empowering users to build them independently and eventually adapt them to their particular needs. The [software](https://www.arduino.cc/en/Main/Software), too, is open-source, and it is growing through the contributions of users worldwide.

Fig. No. 4.1.1: ARDUINO UNO – SPECIFICATIONS



4.1.2 IDE FOR DEVELOPMENT

We are using Arduino IDE version 1.5x which is used to Program and manipulating Arduino Board. We can develop our program in Arduino IDE and upload to the Arduino board.

The Arduino **integrated development environment** is a cross-platform application (for Windows, MacOS, and Linux) that is written in the programming language Java. It is used to write and upload programs to Arduino compatible boards, but also, with the help of 3rd party cores, other vendor development boards.

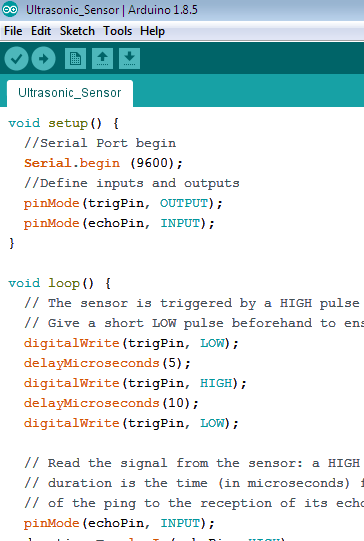
The source code for the IDE is released under the GNU General Public License version2. The Arduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub main( ) into an executable cyclic executive program with the GNU toolchain, also included with the IDE distribution.

The Arduino IDE employs the program avrdude to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware.

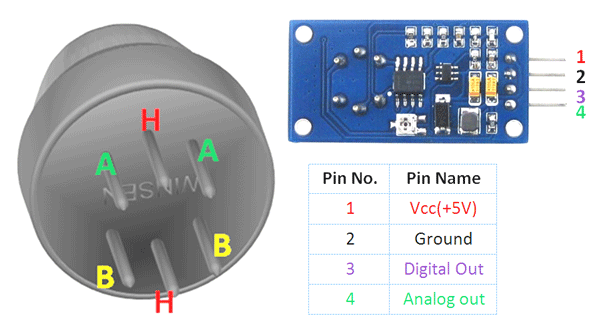
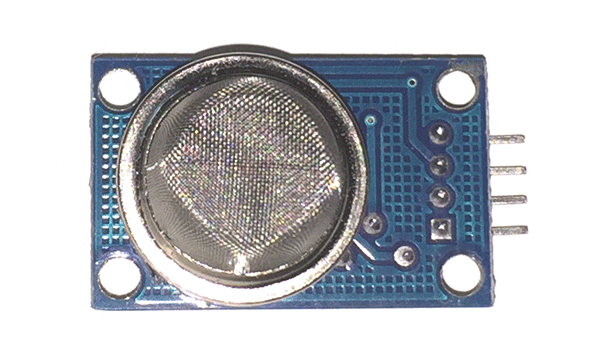
The Arduino is a fantastic single-board microcontroller solution for many DIY projects, and, in this blog, we will look at the Integrated Development Environment, or IDE, that is used to program it.

First, you must download the IDE and install it. Start by visiting Arduino's software page. The IDE is available for most common operating systems, including Windows, Mac OS X, and Linux, so be sure to download the correct version for your OS. If you are using Windows 7 or older, do not download the Windows app version, as this requires Windows 8.1 or Windows 10.

Once the installer has downloaded, go ahead and install the IDE. Chances are you will want to enable all options on the installer, including any USB drivers and libraries, but do make sure to read the EULA!. The Arduino IDE is incredibly minimalistic, yet it provides a near-complete environment for most Arduino-based projects. The top menu bar has the standard options, including “File” (new, load save, etc.), “Edit” (font, copy, paste, etc.), “Sketch” (for compiling and programming), “Tools” (useful options for testing projects), and “Help”. The middle section of the IDE is a simple text editor that where you can enter the program code. The bottom section of the IDE is dedicated to an output window that is used to see the status of the compilation, how much memory has been used, any errors that whe a, found in the program, and various other useful messages



**4.1.3 MQ135- GAS SENSOR FOR AIR QUALITY**



**Fig.no.4.1.3:MQ-135**

**Pin Configuration:**

|  |  |  |
| --- | --- | --- |
| **Pin No:** | **Pin Name:** | **Description** |
| **For Module** | | |
| 1 | **Vcc** | Used to power the sensor, Generally the operating voltage is +5V. |
| 2 | **Ground** | Used to connect the module to system ground. |
| 3 | **Digital Out** | You can also use this sensor to get digital output from this pin, by setting a threshold value using the potentiometer. |
| 4 | **Analog Out** | This pin outputs 0-5V analog voltage based on the intensity of the gas. |
| **For Sensor** | | |
| 1 | **H -Pins** | Out of the two H pins, one pin is connected to supply and the other to ground |
| 2 | **A-Pins** | The A pins and B pins are interchangeable. These pins will be tied to the Supply voltage. |
| 3 | **B-Pins** | 1. A pins and B pins are interchangeable. One pin will act as output while the other will be pulled to ground. |

The MQ-135 Gas sensors are used in air quality control equipment and are suitable for detecting or measuring of NH3, NOx, Alcohol, Benzene, Smoke, CO2. The MQ-135 sensor module comes with a Digital Pin which makes this sensor to operate even without a microcontroller and that comes in handy when you are only trying to detect one particular gas.  If you need to measure the gases in PPM the analog pin need to be used. The analog pin is TTL driven and works on 5V and so can be used with most common microcontrollers.

If you are looking for a sensor to detect or measure common air quality gases such as CO2, Smoke, NH3, NOx, Alcohol, Benzene then this sensor might be the right choice for you.

  Simply power the module with 5V and you should notice the power LED on the module to glow and when no gas it detected the output LED will remain turned off meaning the digital output pin will be 0V. Remember that these sensors have to be kept on for pre-heating time (mentioned in features above) before you can actually work with it. Now, introduce the sensor to the gas you want to detect and you should see the output LED to go high along with the digital pin, if not use the potentiometer until the output gets high. Now every time your sensor gets introduced to this gas at this particular concentration the digital pin will go high (5V) else will remain low (0V).

MQ-135 gas sensor applies SnO2 which has a higher resistance in the clear air as a gas-sensing material. When there is an increase in polluting gases, the resistance of the gas sensor decreases along with that. To measure PPM using MQ-135 sensor we need to look into the (Rs/Ro) v/s PPM graph taken from the MQ135 datasheet.

**4.1.4 BLUETOOTH**

Bluetooth is a de facto standard and specification for small-form factor, low-cost, short range radio links between mobile PCs, mobile phones and other portable devices. The technology allows users to form wireless connections between various communication devices, in order to transmit real-time voice and data communications. The Bluetooth radio is built into a small microchip and operates in the 2.4Ghz band, a globally available frequency band ensuring communication compatibility worldwide. It uses frequency hopping spread spectrum, which changes its signal 1600 times per second which helps to avoid interception by unauthorized parties. In addition, software controls and identity coding built into each microchip ensure that only those units preset by their owners can communicate.

The specification has two power levels defined; a lower power level that covers the shorter personal area within a room, and a higher power level that can cover a medium range, such as within a home. It supports both point-to-point and point-to-multipoint connections and provides up to 720 Kbps data transfer within a range of 10 meters (up to 100 meters with a power boost). The technology uses omni directional radio waves that can transmit through walls and other non-metal barriers. If there is interference from other devices, the transmission speed decreases but does not stop.

With the current specification, up to seven slave devices can be set to communicate with a master radio in one device. This connection of devices (slaves and master) is called a piconet. Several piconets can be linked together to form scatternet which allow communication between other device configurations.

**4.1.5 SWITCH**

Switch is an electrical component which can make or break electrical circuit automatically or manually. Switch is mainly works with ON (open) and OFF (closed) mechanism. Numerous circuits hold [switches that control](https://www.edgefxkits.com/dtmf-based-load-control-system) how the circuit works or actuate different characteristics of the circuit. The classification of switches depends on the connection they make. Two vital components that confirm what sorts of connections a switch makes are pole and throw.

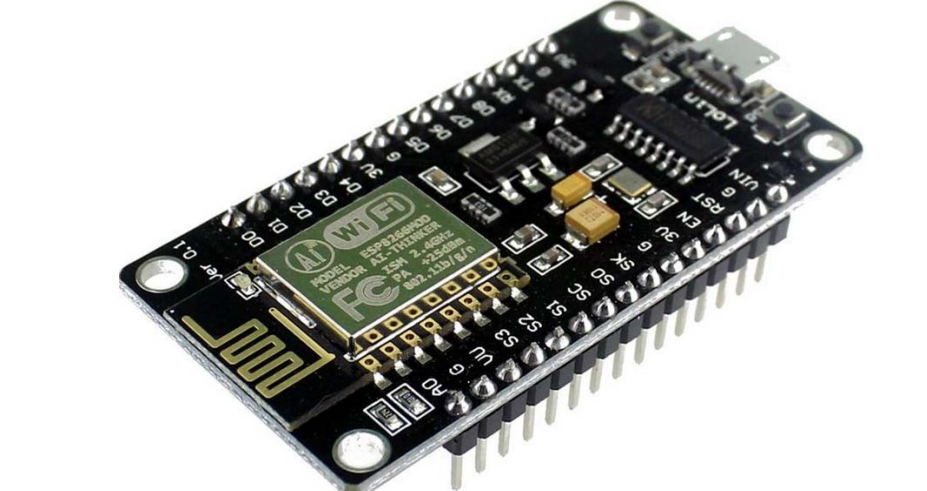
These are classified on based the connections they make. If you were under the impression that switches simply turn circuits on and off, guess again.

The terms pole and throw are also used to describe switch contact variations. The number of “poles” is the number of separate circuits which are controlled by a switch. The number of “throws” is the number of separate positions that the switch can adopt. A single-throw switch has one pair of contacts that can either be closed or open. A double-throw switch has a contact that can be connected to either of two other contacts; a triple-throw has a contact which can be connected to one of three other contacts, etc.

* **Pole:** The amount of circuits controlled by the switch is indicated by poles. Single pole (SP) switch controls only one electrical circuit. Double pole (DP) switch controls two independent circuits.
* **Throw:** The number of throws indicates how many different output connections every switch pole can connect its input. A single throw (ST) switch is a simple on/off switch. When the switch is ON, the two terminals of switch are connected and current flows between them. When the switch is OFF the terminals are not connected, so current does not flow.

**4.1.6 ESP-12E BASED** **NODEMCU**

The ESP8266 is the name of a micro controller designed by Espressif Systems. The ESP8266 itself is a self-contained Wi-Fi networking solution offering as a bridge from existing micro controller to Wi-Fi and is also capable of running self-contained applications. This module comes with a built in USB connector and a rich assortment of pin-outs. With a micro USB cable, you can connect NodeMCU devkit to your laptop and flash it without any trouble, just like Arduino. It is also immediately breadboard friendly.



**Fig.no.4.1.6:ESP-12E**

ESP-12E Wi-Fi module is developed by Ai-thinker Team. core processor ESP8266 in smaller sizes of the module encapsulates Tensilica L106 integrates industry-leading ultra-low power 32-bit MCU micro, with the 16-bit short mode, Clock speed support 80 MHz, 160 MHz, supports the RTOS, integrated Wi-Fi MAC/BB/RF/PA/LNA, on-board antenna. The module supports standard IEEE802.11 b/g/n agreement, complete TCP/IP protocol stack. Users can use the add modules to an existing device networking, or building a separate network controller. ESP8266 is high integration wireless SOCs, designed for space and power constrained mobile platform designers. It provides unsurpassed ability to embed Wi-Fi capabilities within other systems, or to function as a standalone application, with the lowest cost, and minimal space requirement.

**4.1.7 GSM**

**SIM 900 GSM/GPRS MODULE**

****

**Fig.no.4.1.7:GSM**

GSM/GPRS Modem-RS232 is built with Dual Band GSM/GPRS engine- SIM900, works on frequencies 900/ 1800 MHz. The Modem is coming with RS232 interface, which allows you connect PC as well as microcontroller with RS232 Chip(MAX232). The baud rate is configurable from 9600-115200 through AT command. The GSM/GPRS Modem is having internal TCP/IP stack to enable you to connect with internet via GPRS. It is suitable for SMS, Voice as well as DATA transfer application in M2M interface. The onboard Regulated Power supply allows you to connect wide range unregulated power supply. Using this modem, you can make audio calls, SMS, Read SMS, attend the incoming calls and internet through simple AT commands.

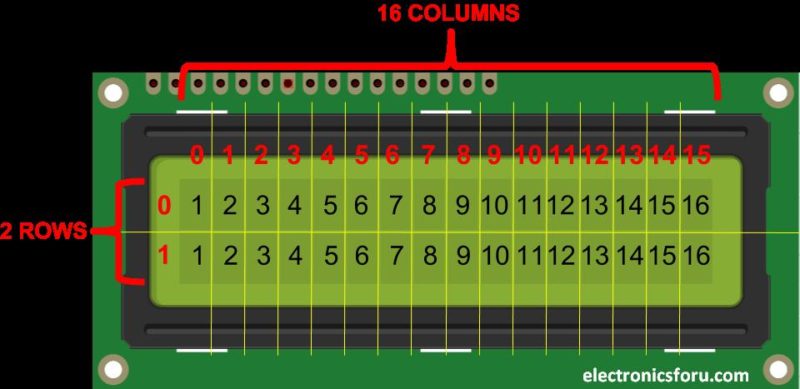
**4.1.8 MOTOR DRIVER IC**

Common DC gear head motors need current above 250mA. There are many integrated circuits like ATmega16 [Microcontroller](https://www.elprocus.com/8051-microcontroller-architecture-and-applications/), [555 timers IC](https://www.elprocus.com/555-timer-circuits-for-engineering-students/). But, IC 74 series cannot supply this amount of current. When the motor is directly connected to the o/p of the above ICs then, they might damage. To overcome this problem, a motor control circuit is required, which can act as a bridge between the above motors and ICs ([integrated circuits](https://www.elprocus.com/different-types-of-integrated-circuits/)). There are various ways of making H-bridge motor control circuit such as using transistor, relays and using L293D/L298.

**4.1.9 LIQUID CRYSTAL DISPLAY**

LCD screen is an electronic display module and find a wide range of applications. A 16x2 LCD display is very basic module and is very commonly used in various devices and circuits. These modules are preferred over [seven segments](http://www.engineersgarage.com/content/seven-segment-display) and other multi segment [LED](http://www.engineersgarage.com/content/led)s. The reasons being: LCDs are economical; easily programmable; have no limitation of displaying special & even [custom characters](http://www.engineersgarage.com/microcontroller/8051projects/create-custom-characters-LCD-AT89C51) (unlike in seven segments), [animations](http://www.engineersgarage.com/microcontroller/8051projects/display-custom-animations-LCD-AT89C51) and so on.

A **16x2 LCD** means it can display 16 characters per line and there are 2 such lines. In this LCD each character is displayed in 5x7 pixel matrix. This LCD has two registers, namely, Command and Data. The command register stores the command instructions given to the LCD. A command is an instruction given to LCD to do a predefined task like initializing it, clearing its screen, setting the cursor position, controlling display etc. The data register stores the data to be displayed on the LCD. The data is the ASCII value of the character to be displayed on the LCD. Click to learn more about internal structure of a [LCD](http://www.engineersgarage.com/insight/how-lcd-works).



**Fig.no.4.1.9: LCD**

**4.1.10** **EMBEDDED C**

Embedded C is most popular programming language in software field for developing electronic gadgets. Each processor used in electronic system is associated with embedded software.

Embedded C programming plays a key role in performing specific function by the processor. In day-to-day life we used many electronic devices such as mobile phone, washing machine, digital camera, etc. These all device working is based on microcontroller that are programmed by embedded C.

Let's see the block diagram representation of embedded system programming:

**4.2 BLOCK DIAGRAM**

**ARDUINO UNO**

**POWER SUPPLY**

**MQ135**

**(AIR QUALITY)**

**BLUETOOTH**

**ROBOT CHASE**

**MOTOR DRIVER**

**DC MOTORS (2)**

**LCD**

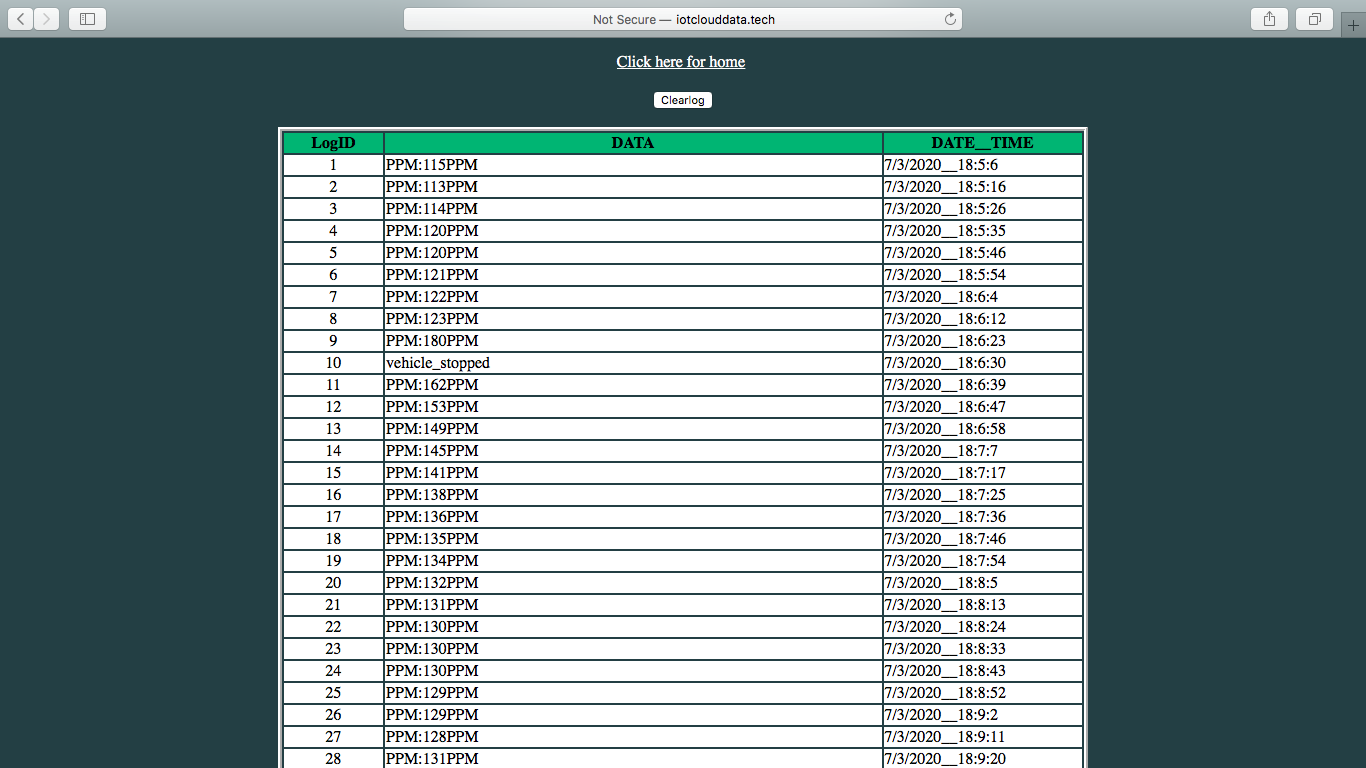
**SWITCH**

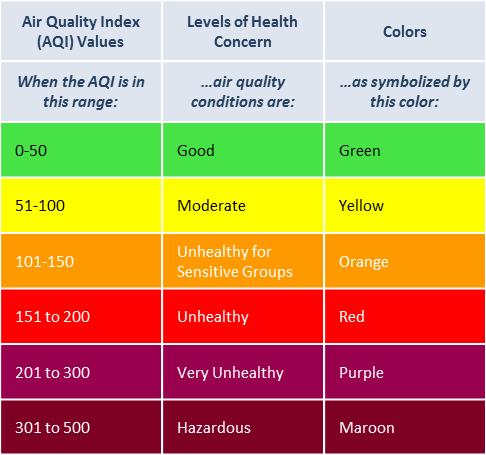
**WORKING PRINCIPLE**

In this system, we use ARDUINO UNO (ATmega328P) microcontroller which acts as brain of the system, because the entire system program instruction stored in it. MQ-135 Air quality sensor is used to check the vehicle exhaust gas level. If the vehicle is maintained well, exhaust gas release from vehicle is low. So person can operate their vehicle. If the vehicle is not maintained means, exhaust gas release from vehicle will be high, so the person cannot operate their vehicle. GSM is used to share the location of the vehicle to the recovery vehicle for help. Bluetooth is used to control the robot chase. Switch is used as ON/OFF of the vehicle. All the status of the process is displayed in the LCD and also updated in IOT

CHAPTER-5

RESULT ANALYSIS





CHAPTER-6

CONCLUSION AND FUTURE WORK

We can monitor air pollution in any lite vehicle by installing this system in it. The system also sends the pollutant level data to the IOT server. The air pollution agencies can able to analyze the data and also detect the vehicle registration numbers that causes more pollution in the atmosphere. The developed system is a low cost, simple to operate and is easily inserted in any locations. The kit has been integrated with the mobile application that helps the user in predicting the pollution level of their entire route. Further, data logging can be used to predict AQI levels. This proposed air pollution monitoring kit along with the integrated mobile application can be helpful to people suffering from respiratory diseases. we intend in the future to deploy our developed system in heavy vehicles with minimal cost

Î

CHAPTER-7

APPENDIX

**7.1 CODING**

#include<SoftwareSerial.h>

SoftwareSerial iot\_data(6, 7);

#include<LiquidCrystal.h>

LiquidCrystal lcd(8, 9, 10, 11, 12, 13);

#define g\_sensor A0

int count = 0;

int ic = 0;

int gas\_read = 0;

#define m1 2

#define m2 3

#define m3 4

#define m4 5

String sdata = "";

bool serial\_flag = 0, g\_flag = 0, i\_flag = 0;

void iot(String data);

void setup() {

Serial.begin(9600);

iot\_data.begin(9600);

lcd.begin(16, 2);

pinMode(g\_sensor, INPUT);

pinMode(m1, OUTPUT);

pinMode(m2, OUTPUT);

pinMode(m3, OUTPUT);

pinMode(m4, OUTPUT);

analogWrite(m1, LOW);

digitalWrite(m2, LOW);

analogWrite(m4, LOW);

digitalWrite(m4, LOW);

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("AIR MONITOR");

delay(1000);

lcd.clear();

iot("\*ENGINE STARTED#");

delay(1000);

send\_sms("ENGINE STARTED");

}

void loop()

{

gas\_func();

}

void gas\_func()

{

gas\_read = analogRead(g\_sensor);

lcd.clear();

lcd.setCursor(9, 0);

lcd.print(gas\_read);

lcd.setCursor(0, 0);

lcd.print("AIR LVL:");

lcd.setCursor(13, 0);

lcd.print("PPM");

lcd.setCursor(11, 1);

lcd.print("D:");

if ((gas\_read >= 0) && (gas\_read <= 50))

{

g\_flag = 0;

lcd.setCursor(0, 1);

lcd.print("HEALTHY");

delay(700);

}

if ((gas\_read >= 51) && (gas\_read <= 100))

{

g\_flag = 0;

lcd.setCursor(0, 1);

lcd.print("MODERATE");

delay(700);

}

if ((gas\_read >= 101) && (gas\_read <= 150))

{

g\_flag = 0;

lcd.setCursor(0, 1);

lcd.print("UNHLTHY");

delay(700);

}

if ((gas\_read >= 151) && (gas\_read <= 200))

{

g\_flag = 0;

lcd.setCursor(0, 1);

lcd.print("UNHLTHY ");

delay(700);

}

if ((gas\_read >= 201) && (gas\_read <= 300))

{

g\_flag = 1;

i\_flag = 1;

lcd.setCursor(0, 1);

lcd.print("V UNHLTHY ");

stop\_func();

send\_sms("VEHICLE STOPPED");

iot("\*vehicle stopped#");

delay(1000);

}

if (gas\_read >= 300)

{

g\_flag = 1;

i\_flag = 1;

lcd.setCursor(0, 1);

lcd.print("HAZARDOUS ");

stop\_func();

send\_sms("VEHICLE STOPPED");

iot("\*vehicle stopped#");

delay(700);

}

if ((ic == 5) && (g\_flag == 0))

{

ic = 0;

iot("\*PPM:" + String(gas\_read) + "PPM#");

}

delay(1000);

ic++;

if (i\_flag) {

stop\_func();

}

}

void stop\_func()

{

digitalWrite(m1, LOW);

digitalWrite(m2, LOW);

digitalWrite(m3, LOW);

digitalWrite(m4, LOW);

lcd.setCursor(13, 1);

lcd.print("S");

delay(700);

}

void iot(String data)

{

for (int i = 0; i < data.length(); i++)

iot\_data.write(data[i]);

delay(1000);

}

void send\_sms(String data)

{

Serial.println("AT+CMGF=1"); //Sets the GSM Module in Text Mode

delay(1000); // Delay of 1000 milli seconds or 1 second

Serial.println("AT+CMGS=\"+919003295885\"\r"); // Replace x with mobile number

delay(1000);

Serial.println(String(data) + '\0'); // The SMS text you want to send

delay(100);

Serial.println("LT:13.0250966");// The SMS text you want to send

delay(100);

Serial.println("LG:80.0316916" ); // The SMS text you want to send

delay(100);

Serial.println((char)26);// ASCII code of CTRL+Z

delay(1000);

}

void serial\_data(void)

{

while (serial\_flag)

{

if (sdata.length() > 0)

{

Serial.print(sdata);

Serial.print(g\_flag);

Serial.print(serial\_flag);

Serial.println();

gas\_func();

if ((sdata == "A") && (g\_flag == 0) && (!i\_flag))

{

stop\_func();

analogWrite(m1, 255);

digitalWrite(m2, LOW);

analogWrite(m3, 255);

digitalWrite(m4, LOW);

lcd.setCursor(11, 1);

lcd.print("D:");

lcd.setCursor(13, 1);

lcd.print("F");

delay(700);

serial\_flag = 0;

Serial.print(serial\_flag);

//break;

}

else if ((sdata == "B") && (g\_flag == 0) && (!i\_flag))

{

stop\_func();

analogWrite(m1, 255);

digitalWrite(m2, LOW);

analogWrite(m4, 255);

digitalWrite(m3, LOW);

lcd.setCursor(13, 1);

lcd.print("L");

delay(700);

serial\_flag = 0;

Serial.print(serial\_flag);

break;

}

else if ((sdata == "C") && (g\_flag == 0) && (!i\_flag))

{

stop\_func();

analogWrite(m2, 255);

digitalWrite(m1, LOW);

analogWrite(m3, 255);

digitalWrite(m4, LOW);

lcd.setCursor(13, 1);

lcd.print("R");

delay(700);

serial\_flag = 0;

Serial.print(serial\_flag);

break;

}

else if ((sdata == "D") && (g\_flag == 0) && (!i\_flag))

{

stop\_func();

analogWrite(m2, 255);

digitalWrite(m1, LOW);

analogWrite(m4, 255);

digitalWrite(m3, LOW);

lcd.setCursor(13, 1);

lcd.print("r");

delay(700);

serial\_flag = 0;

Serial.println(serial\_flag);

break;

}

else if ((sdata == "E") && (g\_flag == 0) && (!i\_flag))

{

digitalWrite(m1, LOW);

digitalWrite(m2, LOW);

digitalWrite(m3, LOW);

digitalWrite(m4, LOW);

lcd.setCursor(13, 1);

lcd.print("S");

delay(700);

serial\_flag = 0;

Serial.println(serial\_flag);

break;

}

}

}

sdata = "";

serial\_flag = 0;

g\_flag = 0;

}

void serialEvent()

{

while (Serial.available() > 0)

{

char c = (char)Serial.read();

if (c == '#')

{

serial\_flag = 1;

break;

}

sdata += c;

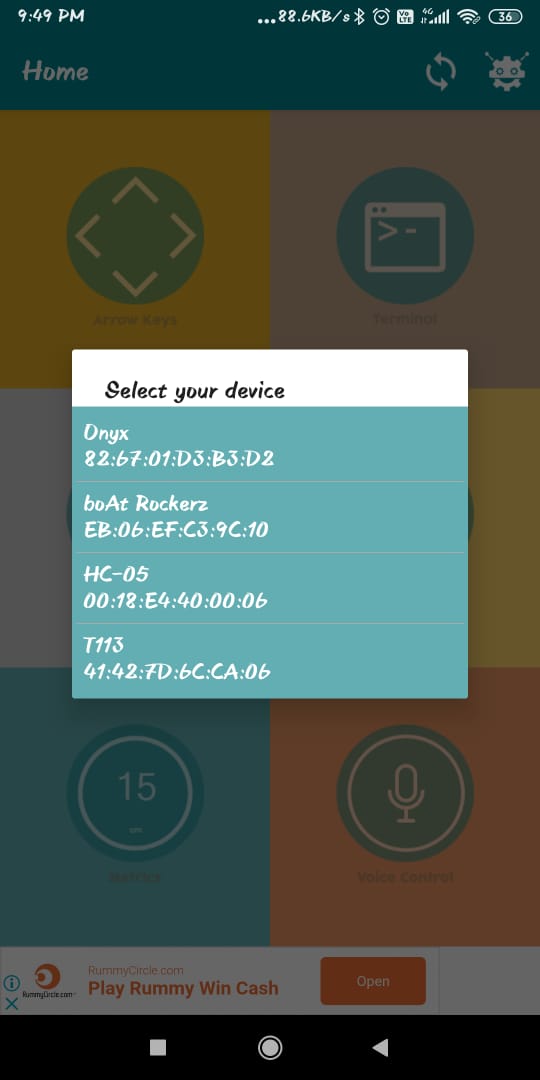
}

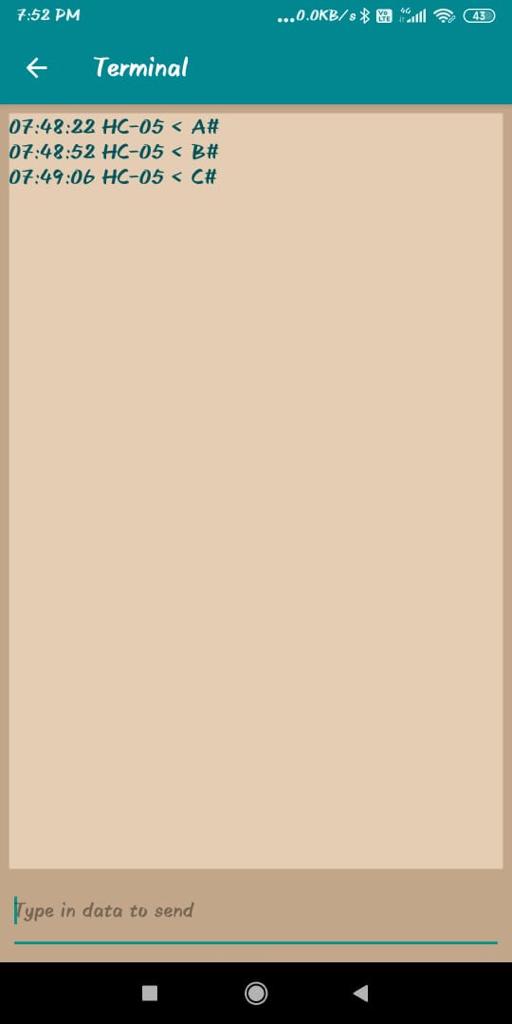
serial\_data();

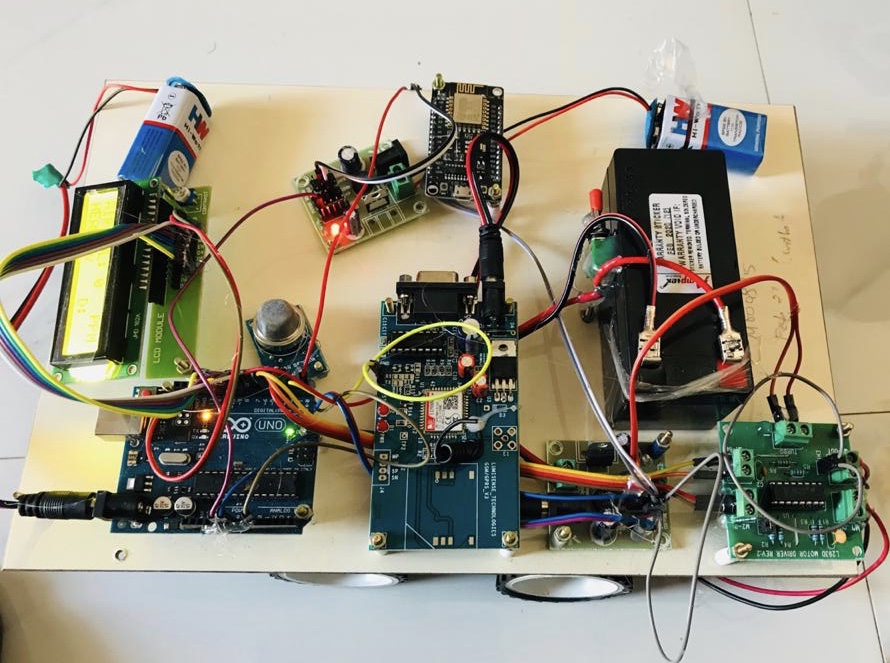
}

**7.2 SCREENSHOTS**

**ANDROID APPLICATION**

****

****

****

****

**REFERENCES**

1. **Siva Shankar Chandrasekaran, Sudharshan Muthukumar and Sabeshkumar Rajendran,” Automated Control System for Air Pollution Detection in Vehicles”, IEEE conference, 2013, DOI 10.1109/ISMS.2013.94**
2. **V.S.Harilakshmi, P.Arockia Jansi Rani”Intelligent Vehicle Counter – A Road to Sustainable Development and Pollution Prevention(P2) “,IEEE Conference,2016**
3. **Mahesh A. Rakhonde,Prof. Dr. S. A. Khoj,Prof. R. D. Komati.”Vehicle Collision Detection and Avoidance with Pollution Monitoring System Using IoT”,IEEE conference, 2018**
4. **Afrah Mohammad,” GSM Based Air Quality Monitoring and Analysis Using Wireless Sensor Node “,IJIRSET, Vol. 5, Issue 6, June 2016, DOI:10.15680/IJIRSET.2015.0506131**
5. **V.S.Revathy, K.Ganesan, K.Rohini, S.Tamil Chindhu, T.Boobalan, “Air Pollution Monitoring System” , Volume 11, Issue 2, IOSR-JECE, MARCH-2016 , DOI: 10.9790/2834-1102022740**
6. **Piyush Patil,” Smart IoT Based System For Vehicle Noise And Pollution Monitoring “,IEEE Conference,2017, 978-1-5090-4257-9/17**
7. **Aijaz Ahmad Reshi, Shabana Shafi, Dr. A. Kumaravel,” Wireless Sensor Network platform for automobile pollution control”, IEEE Conference, 2013**
8. **Vijayakumar Sajjan, Pramod Sharma ,’’Research on an Iot Based Air Pollution Monitoring System”, IJITEE Conference, ISSN: 2278-3075, Volume-8, Issue- 9S2, July 2019**
9. **Godbless Swagarya, Shubi Kaijage , Ramadhani S. Sinde.”Air Pollution Monitoring System based on Wireless Networks - Simulation “, Innovative Systems Design and Engineering Conference ,ISSN 2222-1727 (Paper) ISSN 2222-2871 Vol.5, No.8, 2014**