© 2018 IJSRST | Volume 5 | Issue 3 | Print ISSN: 2395-6011 | Online ISSN: 2395-602X
Themed Section: Science and Technology

Real Time Analysis of Pollutants inVehicles [R.T.A.P.V]

Yash Bhardwaj, Shakti Ratan
Information Science and Engineering, PES Institute of Technology, Banglore South Campus, Banglore, Karnataka, India

ABSTRACT

Air pollution has had devastating impacts on public health and environment, hence it has become a major concern especially in urban areas. A real-time system is essential because conventional systems are not scalable resulting in limited data of pollution levels available to us for detailed research. The need is for a gadget which can sense the level of pollutants at any time instant and help us know about the condition of the vehicles. This project of ours aims to automate the process of checking quantity of pollutants in cars and send data to a cloud system using local Wi-Fi for regular analysis of pollution levels. The real time analysis is the need of the hour and we strive to achieve that aim through this project. The device has been created to be of aid to the environment and the society.

Keywords: Research Paper, Technical Writing, Science, Engineering and Technology, real time analysis, pollution.

I. INTRODUCTION

Research has begun on air pollution all over the world due its hazardous impacts on human health. The concern on air pollution has increased significantly due to the serious hazards caused by it to the public health. The toxic gases emitted by the vehicles are also a chief source of pollution. The condition of the engine majorly decides the gas emissions pollution rate of the vehicle.

The improper combustion of the gases inside the combustion chamber is one of the chief cause of the pollution as these emissions include hazardous gases like carbon monoxide, volatile organic compounds, compounds of sulphur etc. The traditional way of measuring pollution are highly reliable and accurate in measuring wide range of pollutants by gas chromatography - masss spectrometers. The drawback of these conventional monitoring instruments, however, is that they are large in size, heavy and quite expensive.

Our project aims to overcome these drawbacks. The project uses cheap, easily available and portable sensors connected to a module that can access the local Wi-Fi and send the data on to a cloud-based web host. This data can then be analysed as desired. According to a survey in the national capital, 99% of the vehicles got their Pollution Under Control(PUC) certificates but in the same year this data was analysed, car pollution had caused major health issues in the national capital, hence making these PUC certificates unreliable.

In order to overcome this problem, we have made a device that performs Real Time Analysis of Pollutants in Vehicles [R.T.A.P.V], which is cost effective and portable, measuring pollution at regular intervals, as opposed to the stationary machines that vehicles are taken to for measuring the pollution levels every three months, because that’s how long the PUC certificate is valid for.

II. EXPERIMENT

The paper elucidates the measurement of Carbon Monoxide concentration from the exhaust of vehicles and send the measured data to a website over the local Wi-Fi at regular intervals.

The working model, currently, only measures the concentration of carbon monoxide, and sends data to a simple web host and not the cloud-based web host. It is a basic model that simulates the working of the complete device on a smaller scale.

Construction:
The device consists of:
Arduino UNO R3
MQ – 7 CO sensor
esp8266 Wi-Fi module
Jumper wires
Breadboard
N-channel MOSFET

Description of Important components:
ARDUINO UNO microcontroller:
The Arduino UNO R3 acts as the base and brain of our device. This helps us to program the different modules that we have used for our purpose in this project effectively and easily, allowing us access to a wide range of applications. The Arduino IDE, that is required for writing the code and uploading it to the microcontroller, is effortless to use.

It is a microcontroller board based on the Atmega328P. It has 14 digital input/output pins, 6 analog input pins, a 16 MHz quartz crystal, a USB connection jack, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller.

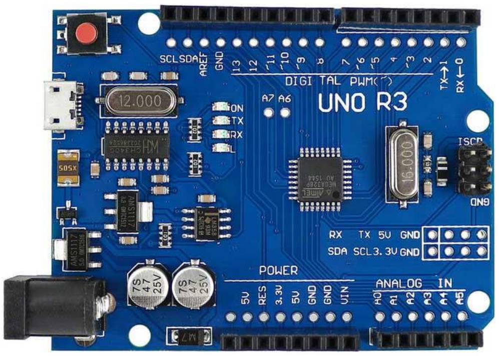


Figure 1

MQ – 7 Carbon Monoxide Sensor:
This is a simple and easy to use Carbon Monoxide (CO) sensor, suitable for sensing CO concentrations in the air. The MQ-7 can detect CO-gas concentrations anywhere from 20 to 2000ppm.

This is a semiconductor gas sensor tuned to detect carbon monoxide. It measures the change in surface conductivity of tin dioxide in the presence of carbon monoxide. This sensor has a high sensitivity and fast response time. The sensor's output is an analog resistance.

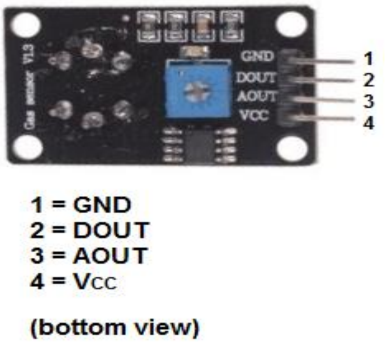
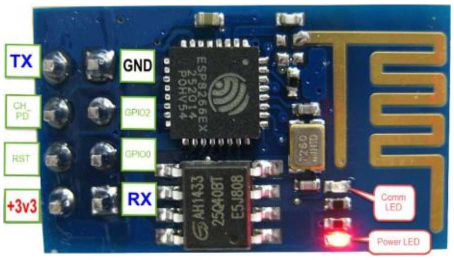


Figure 2

ESP8266 Wi-Fi Module:
ESP8266 provides a complete solution to Wi-Fi networking, which allows it to either host the application or to offload all Wi-Fi functions from another application processor. The module has an integrated cache which improve the performance of

the system, and minimizes the memory requirements.



Figuer 3

The Vcc(input voltage pin) of the MQ – 7 sensor is connected to the 5v pin of the power port of an Arduino board. Analog pin of CO sensor is connected to the analog port of Arduino board. Digital pin 13 of the Arduino board is connected to the Gate pin of the N-channel MOSFET. The Drain pin of the MOSFET is connected to the GND pin of the CO sensor. Source pin of the MOSFET is connected to the GND of the Arduino UNO board. The Vcc of the esp8266 Wi-Fi module is connected to the 3.3v pin of the power section on the Arduino board. GND pin is connected to the GND pin of both the devices. CH_PD pin of esp8266 is connected to the Vcc of esp8266 module. The UTXD pin of esp8266 is connected to the TX (pin 1) on the digital section of the Arduino UNO and the URXD of the Wi-Fi module is connected to the RX (pin 0) of the microcontroller.

III. WORKING

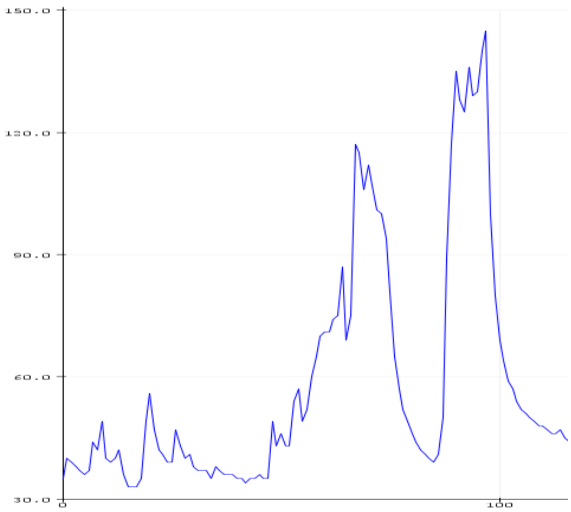
The Arduino UNO is connected to the computer through a USB cable, which also energizes the Arduino board. The MQ-7 sensor works most accurately on a 60-90 second cycle. The coil is heated for 60 seconds and then allowed to cool for 90 seconds. This 60-90 cycle is controlled by the MOSFET which basically acts as a switch. It switches the voltage between 5v and 1.4v. At the end of the 90 second cycle the sensor is supplied with 5v and the data is read through the sensor.

This data can be displayed using the serial monitor that comes in-built with the Arduino IDE. The values sent by the sensor are voltage values and not in ppm format. We can change these values to the ppm format using appropriate code that we'll write in the Arduino ide. Alternately, we can calculate the Ro value of the sensor and determine the ppm level; 1:28 is the ratio of CO in fresh air to that of 400 ppm of CO, i.e.

The values sent by the sensor in clean air is 1/28 times the value sent by the sensor at 400 ppm.

The code is:

```
---------------------------------
const int PinA0=A0
const int Pin13=13
//int limit;
float value;
float R0;
void setup() {
Serial.begin(9600);
pinMode(PinA0, INPUT);
pinMode(Pin13, OUTPUT);
}
void loop() {
digitalWrite(Pin13,HIGH);
delay(60000);
digitalWrite(Pin13,LOW);
delay(90000);
digitalWrite(Pin13,HIGH);
delay(100);
value = analogRead(PinA0);
//Serial.print("CO value :");
Serial.println(value);
R0 = 10*(1023-value)/value;
Serial.print("R0=");
Serial.println(R0);
}
```



Figuer 4

Pollution data from a Hyundai Xcent
(the peak value is when the sensor was closest to the exhaust)

Cloud computing many abilities like flexibility, disaster recovery, automatic software updates and many others, it is being preferred more than normal web hosting.

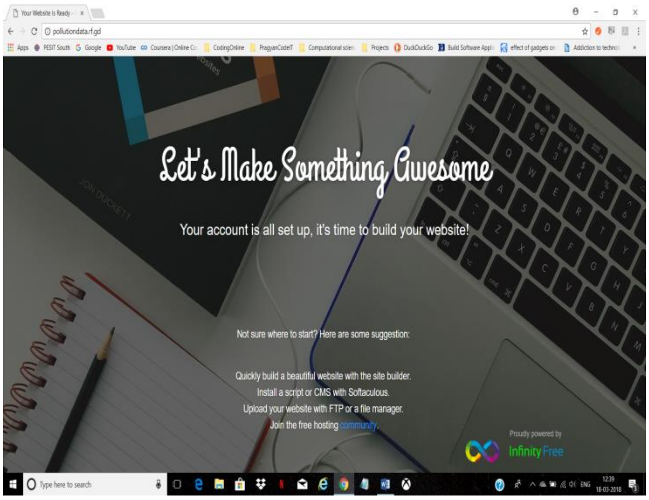
According to mckinsey.com," Yet while automotive digital technology has traditionally focused on optimizing the vehicle's internal functions, attention is now turning to developing the car's ability to connect with the outside world and enhance the in-car experience. This is the connected car—a vehicle able to optimize its own operation and maintenance as well as the convenience and comfort of passengers using onboard sensors and Internet connectivity."

This project tries to incorporate both 'cloud computing' and 'connected cars' to make the device relevant in the current era and easy to manufacture and low priced so that it can be preferred by the general public. Sending real time data of pollution of cars to a server (currently, a normal web server), would enable further research and better understand the problem at hand, i.e., the ever-increasing pollution and find better ways to solve it.

The ESP8266 Wi-Fi Module has integrated TCP/IP protocol stack that can give any microcontroller (Arduino UNO R3, in this case) access to a local Wi-Fi network. The device will use the features of esp8266 to connect to the in-car Wi-Fi and send data to our website.

We have created our own website using one of the many free domain and hosting websites. This project uses https://infinityfree.com for that job, which is convenient for our work. We send data to the server by directly uploading it through MonstaFTP, a MFTP provider.

The website where the data will be posted:



Let's Make Something Awesome

Your account is all set up, it's time to build your website!

Not sure where to start? Here are some suggestion:

Quickly build a beautiful website with the site builder.
Install a script or CMS with Softaculous.
Upload your website with FTP or a file manager.
Join the free hosting community.

https://pollutiondata.rf.gd

Figure 5

HTTP works as a request-response protocol between a client and server. A web browser may be the client, and an application on a computer that hosts a web site may be the server. The response basically contains the status information of the request and might also contain the requested content. In this case, the esp8266 is the client and server is the one that is hosting the above website.

The sever-side script can be written in a number of languages, one of them being PHP, which is very good for handling medium - sized websites. The PHP script is uploaded to the using the file manager(filezilla). When the esp8266 module requests the server for the php file, the server will process the request by performing all the functions specified in the script and then send the response(if required) back. We can then view that data on the website.

Esp8266 Wi-Fi module also helps us determine the location of the person through the help of the geo-location API. In order to get the geo-location API, a person needs to get a key from the Google API to use the geo-location API. The key provides us the access to know the location of the given module we have attached to the gadget, i.e., the esp8266 module. What it actually does is send the location of the device connected to the internet and providing Wi-Fi to the module. The API gives us the latitude as well as longitude thus providing accurate location of the module. Hence, the gadget is able to perform spatio-temporal analysis of pollutants in vehicles.

The code that we upload into Arduino for our device to work is:

```
#include "SoftwareSerial.h"
String ssid ="thessid";
String password="thepassword";
SoftwareSerial esp(6, 7);// RX, TX
const int AOUTpin=A0;
//const int DOUTpin=13;
const int ledPin=13;
String data;
String server = "www.pollutiondata.rf.gd"; // www.example.com
String uri = "www.pollutiondata.rf.gd/datasend.php";// our example is /esppost.php
void setup() {
pinMode(AOUTpin, INPUT);//sets the pin as an input to the arduino
pinMode(ledPin, OUTPUT);//sets the pin as an output of the arduino
esp.begin(9600);
Serial.begin(9600);
reset();
connectWifi();
}
//reset the esp8266 module
void reset() {
esp.println("AT+RST");
delay(1000);
if(esp.find("OK") ) Serial.println("Module Reset");
}
//connect to your wifi network
void connectWifi() {
String cmd = "AT+CWJAP=\"" +ssid+"\",\"" + password + "\"";
esp.println(cmd);
delay(4000);
if(esp.find("OK")) {
Serial.println("Connected!");
}
else {
connectWifi();
Serial.println("Cannot connect to wifi"); }
}
String read_data () {
String value;
digitalWrite(ledPin,HIGH);
//delay(60000);
delay(2000);
digitalWrite(ledPin,LOW);
//delay(90000);
delay(3000);
digitalWrite(ledPin,HIGH);
delay(100);
value = analogRead(AOUTpin);
//Serial.print("CO value :");
Serial.println(value);
 return value;
 }
void loop () {
String val1 = "tom";
//val1=read_data ();
```

```
data = "CO_level="+val1;
httppost();
delay(1000);
}
void httppost () {
esp.println("AT+CIPSTART=\"TCP\",\"" +
server + "\",80");//start a TCP connection.
if( esp.find("OK")) {
Serial.println("TCP connection ready");
} delay(1000);
String postRequest =
"POST " + uri + " HTTP/1.0\r\n" +
"Host: " + server + "\r\n" +
"Accept: *" + "/" + "*\r\n" +
"Content-Length: " + data.length() + "\r\n" +
"Content-Type: application/x-www-form-
urlencoded\r\n" +
"\r\n" + data;

String sendCmd =
"AT+CIPSEND=";//determine the number of
caracters to be sent.
esp.print(sendCmd);
esp.println(postRequest.length() );
delay(500);
if(esp.find(">")) { Serial.println("Sending..");
esp.print(postRequest);
if( esp.find("SEND OK")) {
Serial.println("Packet sent");
while (esp.available()) {
String tmpResp = esp.readString();
Serial.println(tmpResp);
}
// close the connection
esp.println("AT+CIPCLOSE");
}
}
```

If everything goes successfully, we will be able to see the data on to our websit

age because of the ever-increasing pollution. The efforts taken to reduce pollution has not had large enough impact due to world governments being unable to enforce necessary policies in their respective countries. There is also a technological gap between developed and underdeveloped countries that adds into the problem. The device helps to automate the process of checking pollution level in cars and sending data to a cloud system using local Wi-Fi for regular analysis of pollution level. This project helps to fill that technological gap, providing a cheap, reliable and real-time method of measuring carbon monoxide(currently) in the car exhaust.

The This project has helped us to gain knowledge about a lot of new subjects that we never explored before. It has changed our way of thinking and helped us see the surroundings in a different light. This project would have not been possible without the help and guidance of the professors of our college and our family, who were very generous and helpful throughout the making of our project.

We would like to show our gratitude to Prof. Revannasiddappa M for his guidance and mentoring throughout the endeavour. We would like to thank Prof. Prashanth Reddy for his guidance and funding for the project. We would also like to thank Prof. Muhammad Faisal for providing us with novel ideas and support.

IV. REFERENCES

[1]. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4721779/#!po=0.261780
[2]. https://www.areresearch.net
[3]. https://www.instructables.com