Embedded System for Hazardous Gas Detection and Alerting

Course: EL-213

Course Instructor: Dr. Rutu Parekh

Group Members:	ID	Email
1. Kishan Patel	201401012	kishanpatel031096@gmail.com
2. Maulik Trapasiya	201401051	mauliktrapasiya@gmail.com
3. Brij Desai	201401119	brijdesai6@gmail.com
4. Dhruv Koshiyar	201401136	dhruv9586@gmail.com
5. Hitesh Parate	201401142	hiteshparate77@gmail.com
6. Rushi Divecha	201401211	rushidivecha88@gmail.com
7. Kunal Gohel	201401402	kkg176@gmail.com
8. Nihar Lalcheta	201401405	nihar.lalcheta@gmail.com
9. Dhruv Sachdev	201401418	dhruvnsachdev@gmail.com
10.Uttam Chaudhari	201401457	chaudharyuttam36@gmail.com

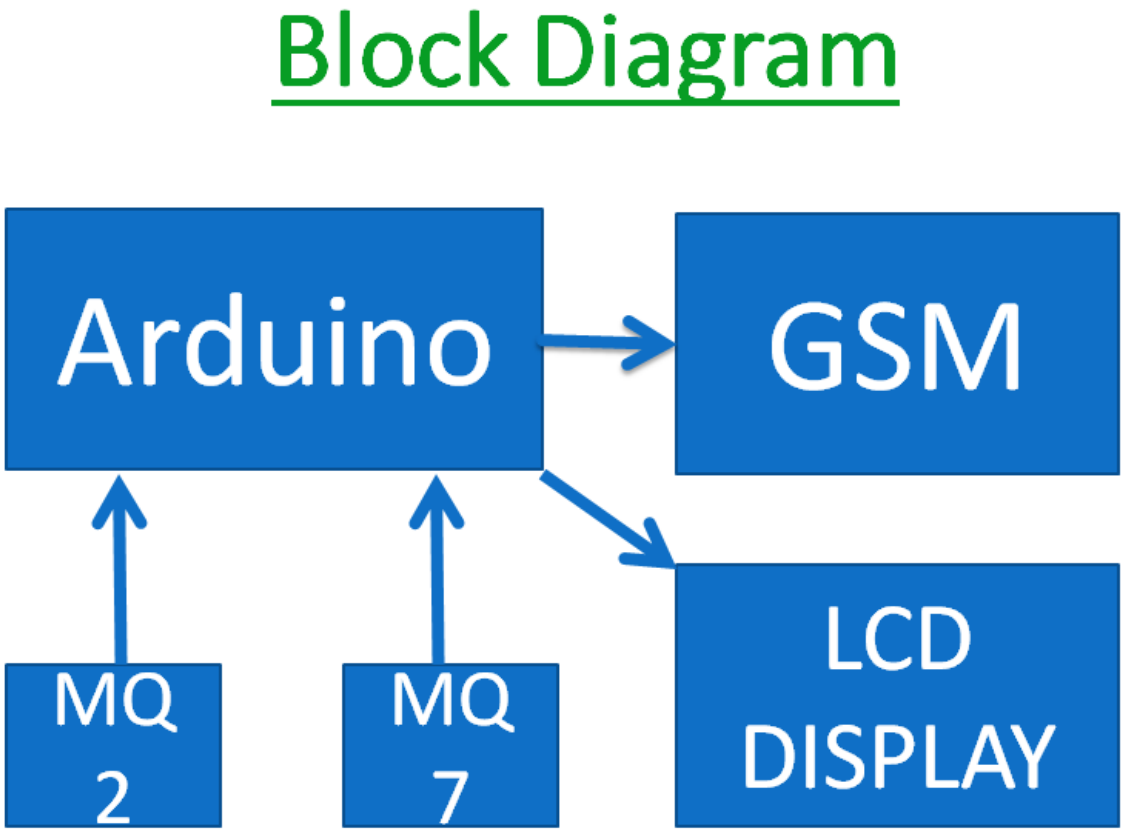
Introduction:

This project deals with the implementation of an embedded system to monitor the hazardous gases. This is useful in industries, homes and offices for the safety of personnel against gas leaks and alert users before the concentration of harmful gases increases above a dangerous value. Here, LPG and CO gas sensors are connected to the microcontroller in order to alert user in the event of any gas leaks.

Implementation:

Components :

- Microcontroller (Arduino UNO R3)
- GSM modem(SIM900A)
- MQ7(CO sensor)
- MQ2(LPG, Propane, Butane and natural gas)
- LCD display (16x2)



Sensors :

We have used MQ2 and MQ7 sensors to detect LPG gas and CO respectively. Supply voltage for both the sensors is 5V (Arduino's voltage). Analog output of sensor is value of voltage (0-5V). This value is normalized into 0 to 1023 by arduino. In the code, we have multiplied 5/1024 to sensor output in order get voltage associated with it (0-5). Ro is sensor resistance at 1000 ppm of H_2 in clean air.

Concept:

Rs_air is sensor resistance in clean air and Rs_gas1 and Rs_gas2 are sensor resistance at various concentration of gases used for LPG and CO respectively. Ratio of Rs_air and Ro is constant and it is 9.8. We can measure Rs_air through sensor and then using ratio (Rs_air/Ro) we can get Ro. Ratio for both the sensors is displayed in LCD display. For LPG, when ratio (Rs_gas1/Ro) goes below 1.75 and for CO, when ratio (Rs_gas2/Ro) goes below 1.7, microcontroller sends a signal to GSM modem which will in turn call to the mobile number as instructed.

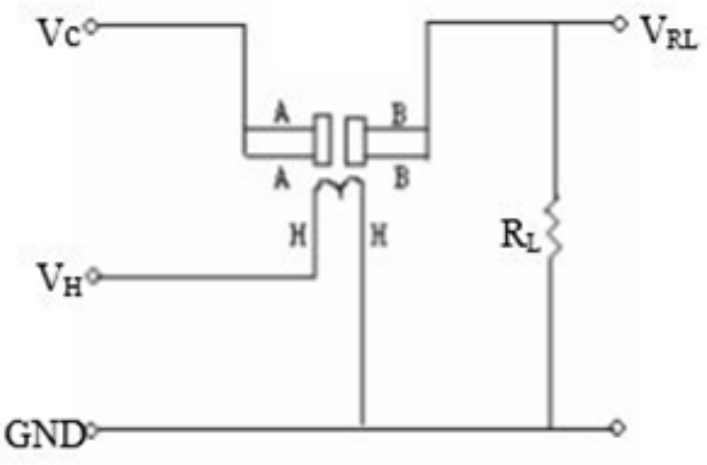
Carbon Monoxide (CO) Detection:

We have used MQ-7 sensor to detect the presence of CO in the surroundings.

Carbon monoxide is a colourless, odourless and tasteless gas. It can be fatal when there is high concentration of CO in the surroundings which makes it important to detect its presence. Therefore, a person would not be aware of the fact that they are breathing a dangerous gas like CO. The person would feel terrible after breathing of CO gas above a certain concentration (above 150-200 ppm). The common symptoms seen are headache, nausea, vomiting, weakness and neurological signs like disorientation and confusion also usually occur.

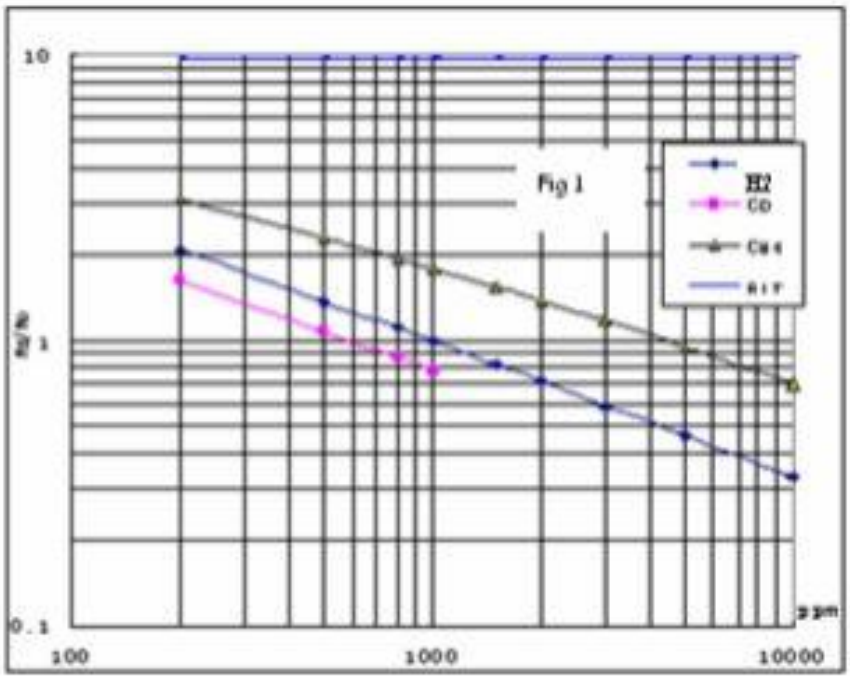
Carbon monoxide is produced when there is partial oxidation of carbon containing compounds. It is formed when there is not enough oxygen for the complete oxidation for the compound. This is generally the case when oxidation happens in small enclosed space. CO is also formed in manholes and in such a closed space, its concentration above a certain value can be deadly. Carbon monoxide enters the body through breathing and it combines with hemoglobin and it forms a deadly compound called carboxyhemoglobin which can lead to failure of hemoglobin to carry oxygen to body parts and a person can succumb to death due to shortage of oxygen.

The circuit which we have implemented is as follows:



Sensitive material of MQ-7 is SnO_2, which has lower conductivity in clean air. The conductivity of the sensor increases due to oxidation of SnO_2 when there is increase in the concentration of CO gas in the atmosphere. We can convert this change of conductivity to corresponding output signal of gas concentration through the circuit given above. The heater provides necessary work conditions for work of the sensitive components. Enveloped MQ-7 has 6 pins, 4 of them are used to fetch signals and other 2 are used to provide heating current.

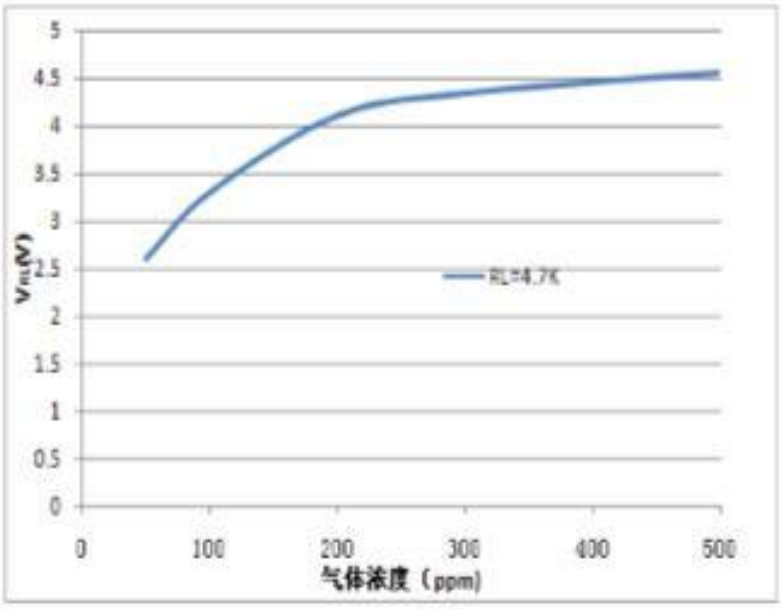
We have taken V_H and V_c equals to 5V D.C. and R_L equals to 10k ohm. We read the voltage across the load resistor R_L through the use of arduino's analog pin A1. Arduino will read that voltage and divide that voltage into 1024 parts. In the code, we multiply it by 5/1024 in order to convert in between 0-5 volts. When the concentration of gas increases and exceeds threshold value which is 200 ppm, arduino sends a signal to GSM to call the respective number according to the code.



Ro=Resistance of H_2 at 1000 ppm in clean air

The surface resistance of the sensor Rs is obtained through effected voltage signal output of the load resistance RL which is series-wound. The relationship between them is described:

Rs/RL = (Vc-VRL) / VRL



V_{RL} vs concentration of CO in ppm

Liquified Petroleum Gas (LPG) Detection:

We have used MQ-2 sensor to detect the presence of LPG in the surroundings.

Application:

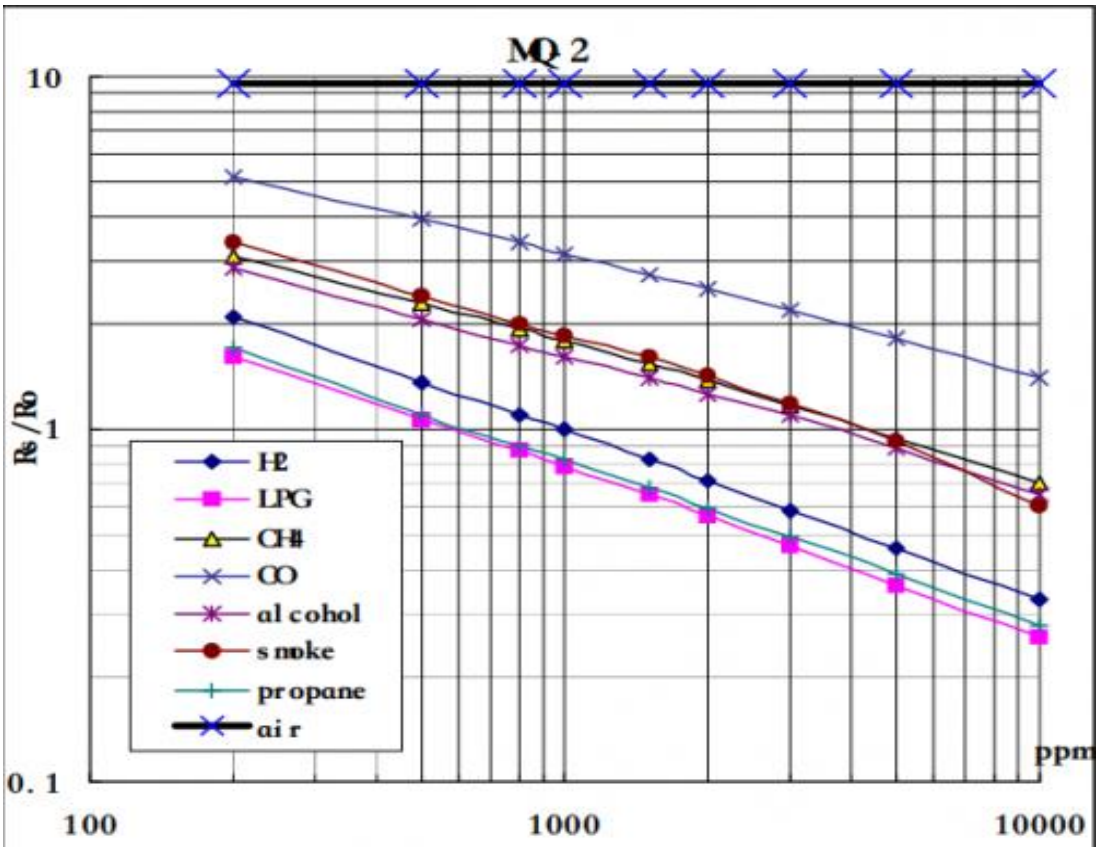
It is used in gas leakage detecting equipment at household and industry level. It can also detect LPG, propane, methane, alcohol, Hydrogen, smoke.

Working:

This sensor module has 4 pins. Those pins are used for Vcc, GND, analog output and digital output. Here concentration of gas is detected based on sensor's resistance values. Here sensitivity is defined as ratio of resistance “Rs” and “Ro”

Graph for Rs/Ro vs concentration of different gases (in ppm) is given below.

When concentration of LPG gas is in between 200 ppm to 10000 ppm, the ratio of Rs/Ro varies from 1.75 to 0.15 because as concentration of gas is increased, conductivity in sensor increases, so resistance decreases and thus ratio of Rs/Ro decreases.



Implementing MQ2 Sensor in our project:

We connected analog pin of sensor to analog pin A0. Analog pin of sensor provides voltage across Rl to the scale of 0 to 1023.

By using this values we first calculate “Ro” which is reference resistance and it is fixed. Then we are calculating value of Rs, which depicts the concentration of LPG. LPG is harmful to human health if its concentration increases above 200 ppm. So we have set 200ppm to be threshold value for Arduino. Hence Whenever sensor value cross threshold value then we can get alert call through Arduino and GSM. We are also displaying sensitivity i.e. “Rs/Ro” in 16x2 LCD Display.

Circuit connection:

Connection between LCD display and arduino:

Pin 1 to Arduino GND

Pin 2 to Arduino 5V

Pin 3 to the middle pin of the 10k potentiometer. One of the potentiometer pin is connected to 5V and another pin is connected to GND.

Pin 4 to Arduino pin 12

Pin 5 to Arduino GND

Pin 6 to Arduino pin 11

Pin 11 to Arduino pin 5

Pin 12 to Arduino pin 4

Pin 13 to Arduino pin 3

Pin 14 to Arduino pin 2

Pin 15 to Arduino 5V

Pin 16 to Arduino GND

Here potentiometer is used to adjust contrast of LCD and pin 15 and pin 16 are used to enable backlight of LCD.

Pin 5 can be used to read/write. If it is LOW or grounded then it is used to write to LCD display and it is HIGH then it is used to read from LCD display. We want to write to LCD display so we have grounded it.

Connection of MQ2 with arduino:

Analog output of sensor is connected with analog pin A0 of arduino. 5V supply is given to sensor to enable it and ground is common.

Connection of MQ7 with arduino:

Here the sensor consists of 6 terminals in which 2 are heating inputs and 4 are to fetch signals. One of the heating terminals is given 5V and other heating terminal is connected to ground. Two A pins are given a voltage of 5V and B pins are connected to load resistance. We take sensor output across load resistance (V_{RL}) which is connected to A1 pin of arduino.

Connection between GSM and arduino:

Rx pin of arduino => Tx pin of GSM

Tx pin of arduino => Rx pin of GSM

Ground is common.

Arduino Code:

The following code is for the arduino for both the sensors MQ-2 and MQ-7:

```
#include <SoftwareSerial.h>
#include <LiquidCrystal.h>

// initialize the library with the numbers of the interface pins
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
SoftwareSerial mySerial(0, 1);
const int gaspin_mq2 = A0;
const int gaspin_mq7 = A1;
float
RO1,RO2,sensorValue1=0,sensorValue2=0,sensorVoltage1,sensorVoltage2;
```

```
int RL1=5000, RL2=10000;
void setup()
{
  // put your setup code here, to run once:
  Serial.begin(9600);
  lcd.begin(16, 2);
  mySerial.begin(9600);

  float Rs_air1,Rs_air2;
  //Finding RO2 for MQ 2
  for(int i=1;i<=100;i++)
    sensorValue1+=analogRead(gaspin_mq2);
  sensorValue1=sensorValue1/100;      // Finding the average value of 100 sensorValues of sensor
  Serial.print("Pure air reading for MQ 2 = ");
  Serial.println(sensorValue1);
  Serial.println();
  sensorVoltage1 = sensorValue1*5.0/1024;
  Rs_air1 = RL1*(5-sensorVoltage1)/sensorVoltage1;
  RO1 = (Rs_air1/9.8);   //Ratio of Rs/Ro is 9.8 for air as mentioned in the graph in the datasheet

  //Finding RO2 for MQ 7
  for(int i=1;i<=100;i++)
       sensorValue2+=analogRead(gaspin_mq7);
  sensorValue2=sensorValue2/100;
  Serial.print("Pure air reading for MQ 7 = ");
  Serial.println(sensorValue2);
  Serial.println();
  sensorVoltage2=(sensorValue2*5.0)/1024;
  Rs_air2 = RL2*(5-sensorVoltage2)/sensorVoltage2;
  RO2 = (Rs_air2/9.8);   //Ratio of Rs/Ro is 9.8 for air as mentioned in the graph in the datasheet
}

void loop()
{
  lcd.begin(16,2);
  float Rs_gas1,Rs_gas2,ratio1,ratio2;

  //Code for MQ2
  sensorValue1=analogRead(gaspin_mq2);
```

```
  Serial.print("LPG gas reading = ");
  Serial.println(sensorValue1);
  sensorVoltage1=(sensorValue1*5.0)/1024;
  Rs_gas1=RL1*(5-sensorVoltage1)/(sensorVoltage1);
  ratio1=Rs_gas1/RO1;
  Serial.print("Rs/Ro = ");
  Serial.println(ratio1);
  //Adjusting the cursor of LCD and printing to it
  lcd.setCursor(0, 0);
  lcd.print("LPG            ");
  lcd.setCursor(4,0);
  lcd.print(ratio1);
  if(ratio1>=0.15 && ratio1<=1.75 ){    // Between 200ppm to 10000ppm
delay(2000);
    mySerial.println("ATD9714827200;");
    Serial.println("ATD9714827200;");
    lcd.setCursor(0,1);
    lcd.print("Calling        ");
delay(5000);
  }

  //Code for MQ7
  sensorValue2 = analogRead(gaspin_mq7);
  Serial.print("CO gas reading = ");
  Serial.println(sensorValue2);
  sensorVoltage2 = sensorValue2*5.0/1024;  //Voltage
  Rs_gas2 = RL2*(5.0-sensorVoltage2)/sensorVoltage2; //Ohm
  ratio2 = Rs_gas2/RO2;
  Serial.print("Rs/Ro = ");
  Serial.println(ratio2);
  //Adjusting the cursor of LCD and printing to it
  lcd.setCursor(0,1);
  lcd.print("CO ");
  lcd.setCursor(4,1);
  lcd.print(ratio2);
  if(ratio2>=0.7 && ratio2<=1.7)   //Between 200ppm to 1000ppm
  {
delay(2000);
    mySerial.println("ATD9099028100;");
    Serial.println("ATD9099028100;");
    lcd.setCursor(0,0);
    lcd.print("Calling        ");
```

```
delay(5000);
  }
  Serial.println();
  delay(7000);
}
```

References:

- https://www.seeedstudio.com/depot/datasheet/MQ-2.pdf
- https://www.pololu.com/file/download/MQ7.pdf?file_id=0J313
- http://www.instructables.com/id/My-First-Project-Arduino-LCD-16x2-Display/?ALLSTEPS
- http://www.circuitstoday.com/interface-gsm-module-with-arduino
- http://www.seeedstudio.com/wiki/Grove_-_Gas_Sensor%28MQ2%29