

```

#include <Wire.h>
#include <LCD.h>
// #include <dht.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE);
#include <SoftwareSerial.h>
SoftwareSerial gprsSerial(2,3);
SoftwareSerial SIM900(2, 3); // gsm module connected here
String textForSMS;

#include <String.h>
#include <DHT.h>
#define DHTPIN 7
DHT dht(DHTPIN, DHT11);

int gas_sensor = A0; //Sensor pin... mq 135
float m = -0.3376; //Slope
float b = 0.7165; //Y-Intercept
float R0 = 15.52; //Sensor Resistance in fresh air from previous code

int CO_sensor = A1; //Sensor pin... mq 7
float m1 = -0.6527; //Slope
float b1 = 1.30; //Y-Intercept
float R01 = 6.42; //Sensor Resistance

int LPG_sensor = A2; //Sensor pin ... mq 6
float m2 = -0.6527; //Slope
float b2 = 1.30; //Y-Intercept
float R02 = 6.42; //Sensor Resistance

int ch4 = A3; //Sensor pin ... mq 4
float m3 = -0.6527; //Slope
float b3 = 1.30; //Y-Intercept
float R03 = 6.42; //Sensor Resistance

void setup() {

    // PC to Arduino Serial Monitor
    // Serial.begin(115200); // Arduino to ESP01 Communication
    lcd.begin(16, 2);
    lcd.backlight();
    lcd.setCursor(0, 0);
    lcd.print(" Air Quality & ");
    lcd.setCursor(0, 1);
    lcd.print("Weather Station");
    delay(2000);
    lcd.clear();
}

```

```

pinMode(gas_sensor, INPUT);
pinMode(CO_sensor, INPUT);
pinMode(LPG_sensor, INPUT);
pinMode(ch4, INPUT);

gprsSerial.begin(9600);          // the GPRS baud rate
Serial.begin(9600); // the GPRS baud rate
dht.begin();
delay(1000);
randomSeed(analogRead(0));
SIM900.begin(9600); // for sim900D 19200. while enter 9600 for sim900A
}

void loop() {
  // put your main code here, to run repeatedly:
  float sensor_volt; //Define variable for sensor voltage
  float RS_gas; //Define variable for sensor resistance
  float ratio; //Define variable for ratio
  float sensorValue = analogRead(gas_sensor); //Read analog values of sensor
  sensor_volt = sensorValue*(5.0/1023.0); //Convert analog values to voltage
  RS_gas = ((5.0*10.0)/sensor_volt)-10.0; //Get value of RS in a gas
  ratio = RS_gas/R0; // Get ratio RS_gas/RS_air
  double ppm_log = (log10(ratio)-b)/m; //Get ppm value in linear scale according to the the ratio value
  double ppm = pow(10, ppm_log); //Convert ppm value to log scale
  Serial.print("Air Quality = ");
  Serial.println(ppm);
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("AQ:");
  lcd.print(ppm);
  lcd.print(" PPM ");
  delay(2000);
  lcd.clear();

  float sensor_volt1; //Define variable for sensor voltage
  float RS_gas1; //Define variable for sensor resistance
  float ratio1; //Define variable for ratio
  float sensorValue1 = analogRead(CO_sensor); //Read analog values of sensor
  sensor_volt1 = sensorValue1*(5.0/1023.0); //Convert analog values to voltage
  RS_gas1 = ((5.0*10.0)/sensor_volt1)-10.0; //Get value of RS in a gas
  ratio1 = RS_gas1/R01; // Get ratio RS_gas/RS_air
  double ppm_log1 = (log10(ratio1)-b1)/m1; //Get ppm value in linear scale according to the the ratio value
  double ppm1 = pow(10, ppm_log1); //Convert ppm value to log scale
  Serial.print("CO PPM = ");
  Serial.println(ppm1);

  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("CO:");

```

```

    lcd.print(ppm1);
    lcd.print(" PPM ");
    delay(2000);
    lcd.clear();

float sensor_volt2; //Define variable for sensor voltage
float RS_gas2; //Define variable for sensor resistance
float ratio2; //Define variable for ratio
float sensorValue2 = analogRead(LPG_sensor); //Read analog values of sensor
sensor_volt2 = sensorValue2*(5.0/1023.0); //Convert analog values to voltage
RS_gas2 = ((5.0*10.0)/sensor_volt2)-10.0; //Get value of RS in a gas
ratio2 = RS_gas2/R02; // Get ratio RS_gas/RS_air
double ppm_log2 = (log10(ratio2)-b2)/m2; //Get ppm value in linear scale according to the the ratio value
double ppm2 = pow(10, ppm_log2); //Convert ppm value to log scale
Serial.print("LPG PPM = ");
Serial.println(ppm2);

    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("LPG:");
    lcd.print(ppm2);
    lcd.print(" PPM ");
    delay(2000);
    lcd.clear();

float sensor_volt3; //Define variable for sensor voltage
float RS_gas3; //Define variable for sensor resistance
float ratio3; //Define variable for ratio
float sensorValue3 = analogRead(ch4); //Read analog values of sensor
sensor_volt3 = sensorValue3*(5.0/1023.0); //Convert analog values to voltage
RS_gas3 = ((5.0*10.0)/sensor_volt3)-10.0; //Get value of RS in a gas
ratio3 = RS_gas3/R03; // Get ratio RS_gas/RS_air
double ppm_log3 = (log10(ratio3)-b3)/m3; //Get ppm value in linear scale according to the the ratio value
double ppm3 = pow(10, ppm_log3); //Convert ppm value to log scale
Serial.print("CH4 = ");
Serial.println(ppm3);

    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("CH4:");
    lcd.print(ppm3);
    lcd.print(" PPM ");
    delay(2000);
    lcd.clear();

float h = dht.readHumidity();
float t = dht.readTemperature();
delay(100);

```

```

Serial.print("Temperature = ");
Serial.print(t);
Serial.println(" °C");
Serial.print("Humidity = ");
Serial.print(h);
Serial.println(" %");

lcd.setCursor(0,0);
lcd.print("Temperature:");
lcd.print(t);
lcd.print(" Celcius ");
lcd.setCursor(0,1);
lcd.print("Humidity:");
lcd.print(h);
lcd.print(" % ");
delay(1000);
  lcd.clear();

if (gprsSerial.available())
  Serial.write(gprsSerial.read());

gprsSerial.println("AT");
delay(1000);

gprsSerial.println("AT+CPIN?");
delay(1000);

gprsSerial.println("AT+CREG?");
delay(1000);

gprsSerial.println("AT+CGATT?");
delay(1000);

gprsSerial.println("AT+CIPSHUT");
delay(1000);

gprsSerial.println("AT+CIPSTATUS");
delay(2000);

gprsSerial.println("AT+CIPMUX=0");
delay(2000);

ShowSerialData();

gprsSerial.println("AT+CSTT=\"robigprs.com\"");//start task and setting the APN,
delay(1000);

ShowSerialData();

```

```

gprsSerial.println("AT+CIICR");//bring up wireless connection
delay(3000);

ShowSerialData();

gprsSerial.println("AT+CIFSR");//get local IP adress
delay(2000);

ShowSerialData();

gprsSerial.println("AT+CIPSPRT=0");
delay(3000);

ShowSerialData();

gprsSerial.println("AT+CIPSTART=\"TCP\", \"api.thingspeak.com\", \"80\");//start up the connection
delay(6000);

ShowSerialData();

gprsSerial.println("AT+CIPSEND");//begin send data to remote server
delay(4000);
ShowSerialData();

String str="GET https://api.thingspeak.com/update?api_key=ZBH8TUM52HDGIQ51&field1=" + String(t)
+"&field2="+String(h)+"&field3="+String(ppm)+"&field4="+String(ppm1)+"&field5="+String(ppm2)+"&f
ield6="+String(ppm3);
Serial.println(str);
gprsSerial.println(str);//begin send data to remote server

delay(4000);
ShowSerialData();

gprsSerial.println((char)26);//sending
delay(5000);//waitting for reply, important! the time is base on the condition of internet
gprsSerial.println();

ShowSerialData();

gprsSerial.println("AT+CIPSHUT");//close the connection
delay(100);
ShowSerialData();

if (( ppm > 8)|| ( ppm1 > 2)|| ( ppm2 > 5)|| ( ppm3 > 6)|| ( t > 30)|| ( h > 50))
{
    textForSMS = "\nWarning! Gas Detected";
    sendSMS(textForSMS);
    Serial.println(textForSMS);
}

```

```

    Serial.println("message sent.");
    delay(5000);
    //while(1)
    //{
    //
    //}
    }
}

void ShowSerialData()
{
    while(gprsSerial.available()!=0)
    Serial.write(gprsSerial.read());
    delay(5000);
}

void sendSMS(String message)
{
    SIM900.print("AT+CMGF=1\r");           // AT command to send SMS message
    delay(1000);
    SIM900.println("AT + CMGS = \""+8801839689600"\"); // recipient's mobile number, in international
    forma8t

    delay(1000);
    SIM900.println(message);                // message to send
    delay(1000);
    SIM900.println((char)26);               // End AT command with a ^Z, ASCII code 26
    delay(1000);
    SIM900.println();
    delay(100);                             // give module time to send SMS
    // SIM900power();                       // turn off module
}

```