END SEMESTER ASSESSMENT (ESA) B.TECH.
(CSE)
IV SEMESTER

UE18CS256 – MICROPROCESSOR AND COMPUTER ARCHITECTURE LABORATORY

MINI PROJECT REPORT

ON

Fire Detection and Alarming System

SUBMITTED BY

NAME	SRN
1) Chandradhar Rao	PES1UG19CS123
2) Dandu Satvik	PES1UG19CS128

JANUARY – MAY 2021

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

RR CAMPUS,

BENGALURU – 560100, KARNATAKA, INDIA

TABLE OF CONTENTS

Sl.No	TOPIC	PAGE NO
1.	ABSTRACT OF THE PROJECT	3
2.	CIRCUIT DIAGRAM	4
3.	ARDUINO CODE	5
4.	SCREEN SHOTS OF THE OUTPUT	6
5.	REFERENCES	7

ABSTRACT OF THE PROJECT:

This circuit can be used to simulate an actual fire detection device before implementing it on a large scale.

The temperature sensor and gas sensors would always be listening for a spike in temperature or increase in concentration of dangerous gases released during a fire outbreak.

The temperature sensor measures analog potential change caused due to change in temperature in A0.The value in analog can range from 0 to 2^10.Hence to get a value between 0 and 1,we divide by 1023 and since our power supply in 5mV we multiply it with 500 and convert it to degree celsius.

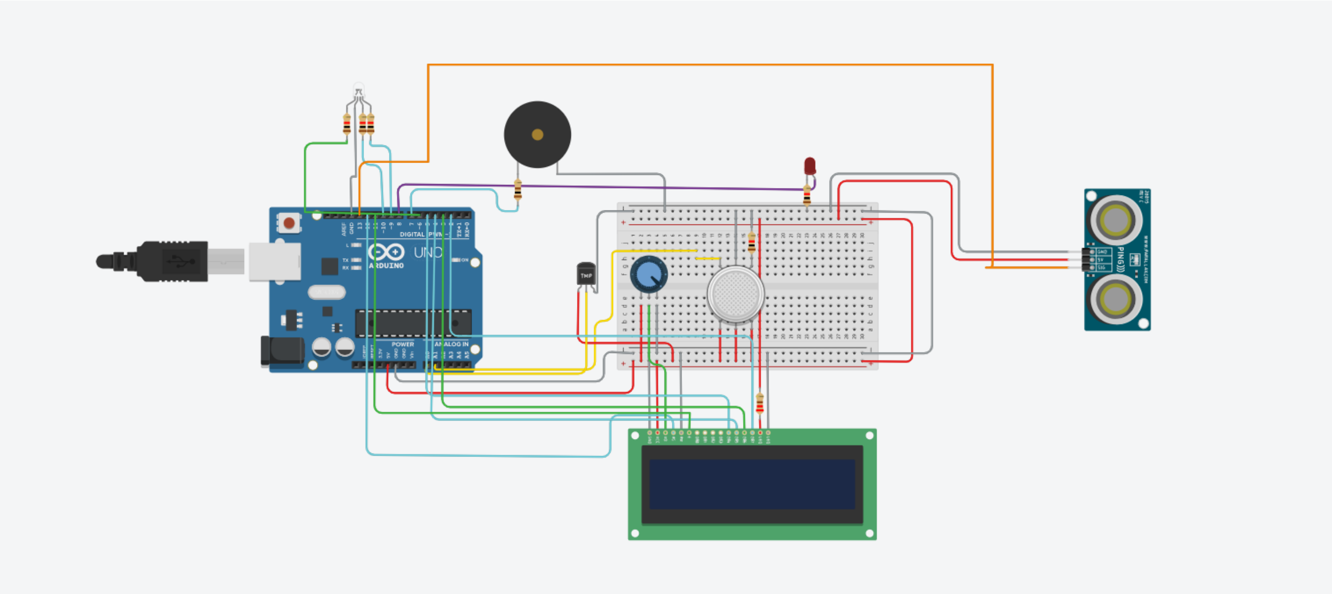
If the temperature is less than 65,it's normal.

If temperature is between 65 and 100 degree celsius,its level 1 of fire outbreak that could still be contained but if the temperature is greater than 100 degrees then it is dangerous and people would not be allowed to enter the room.

Similarly if the gas concentration of methane crosses 250 parts per million then the users are alerted. With a beep sound from the piezo electric buzzer,LCD display message.

If either increase in temperature or gas concentration goes above the mentioned threshold,the ultrasonic sensor will detect the distance of the user from the region of fire and warn the incoming user of the fire and tell him/her to stay away.

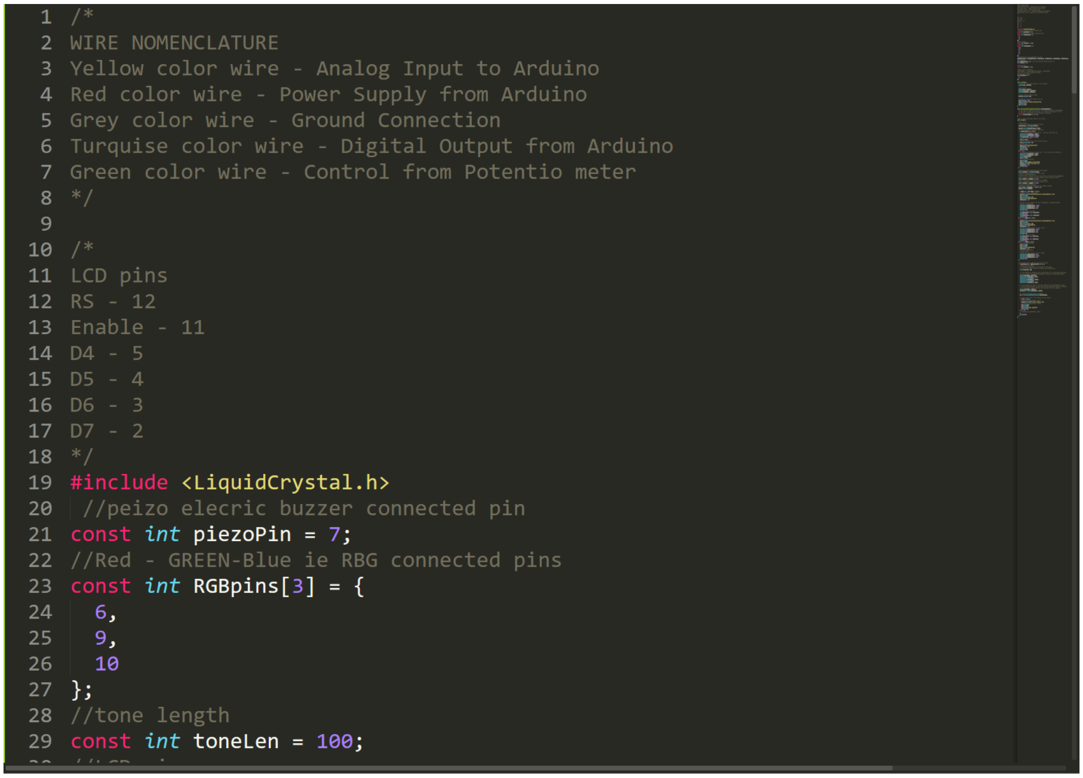
CIRCUIT DIAGRAM:



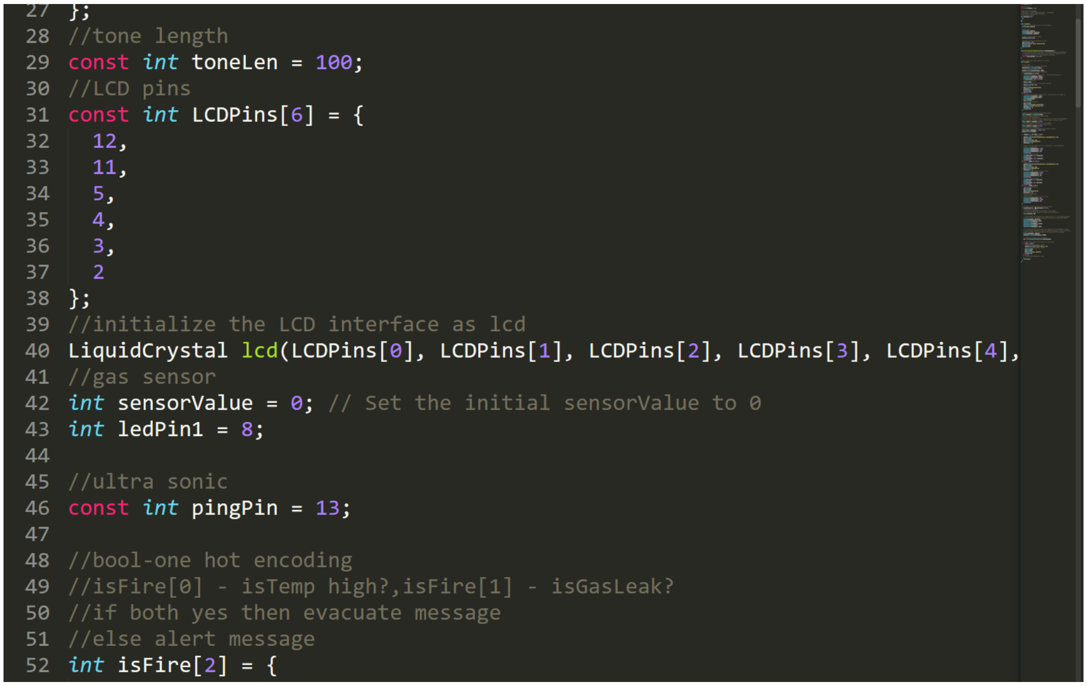
Components Used :

1) Arduino UNO Board
2) LM35 Temperature Sensor
3) Gas Sensor
4) Resistor
5) BreadBoard
6) LEDs
7) Piezo Buzzer
8) Jumper Wires

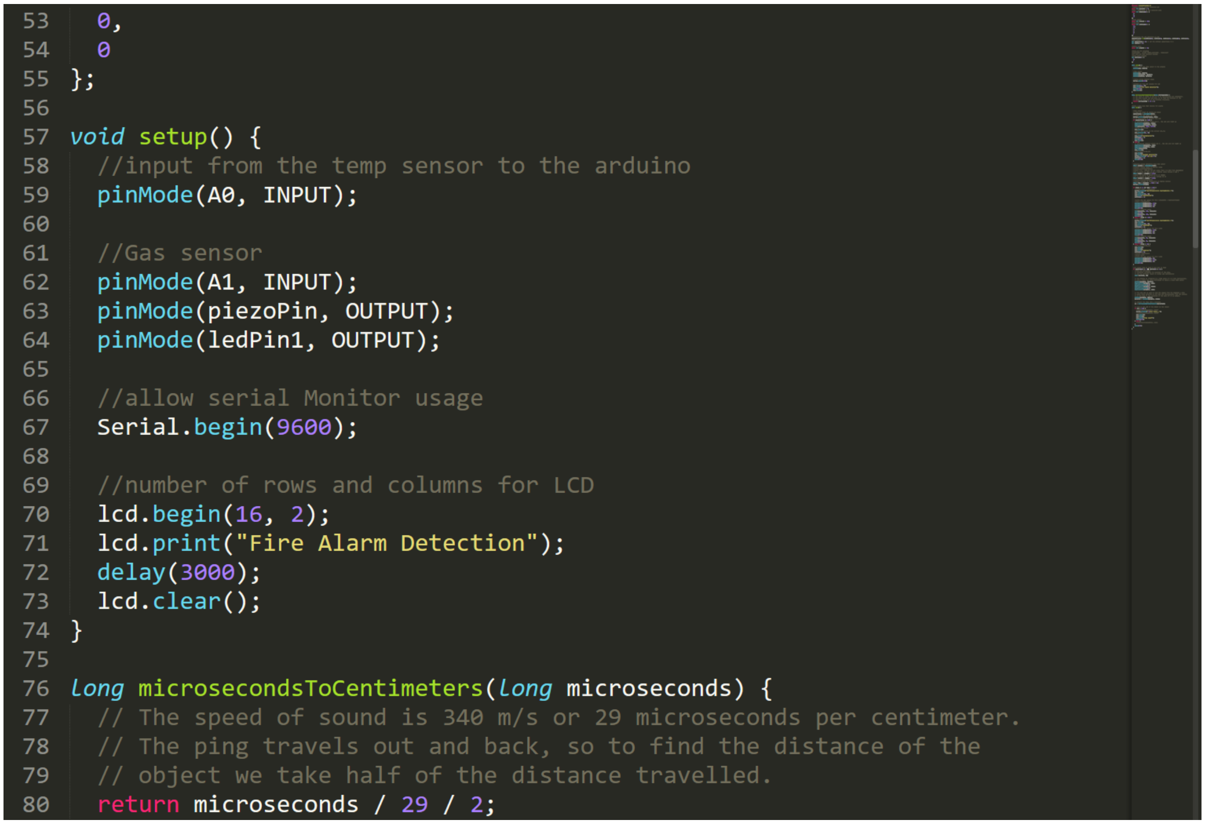
ARDUINO CODE:



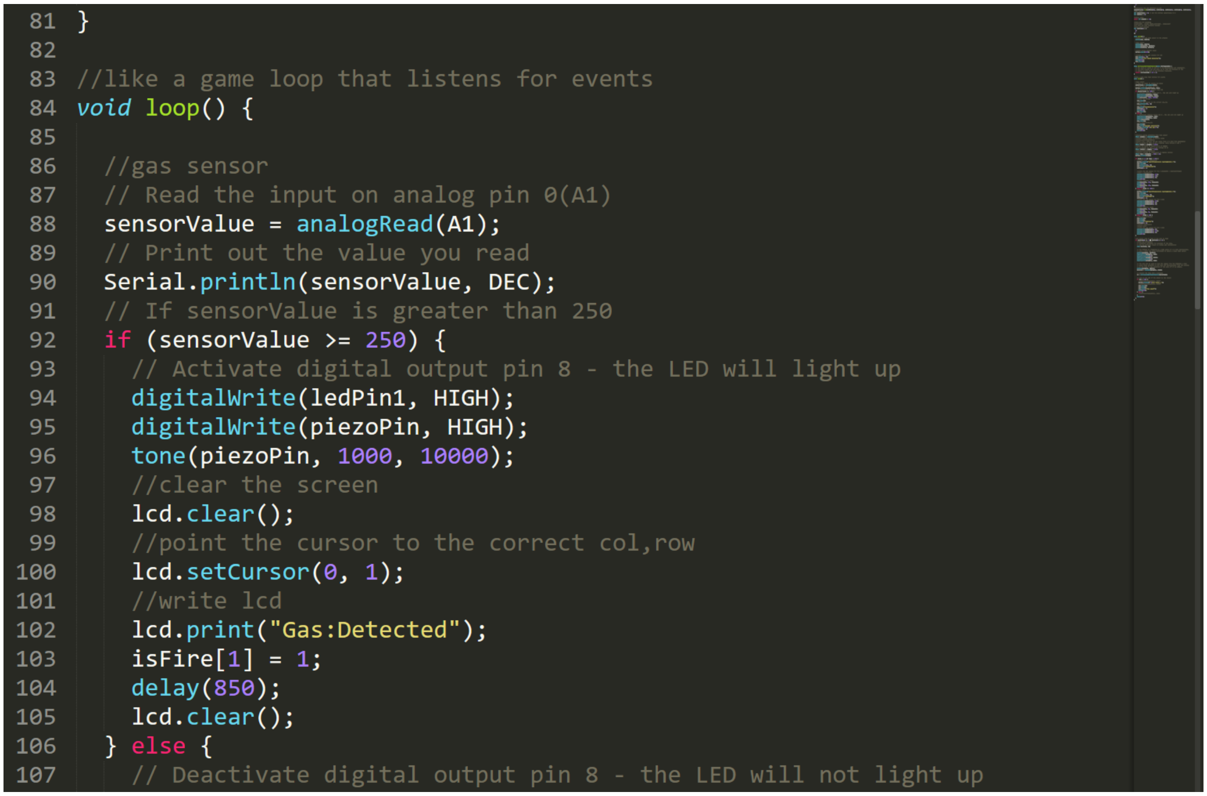
```
/*
WIRE NOMENCLATURE
Yellow color wire - Analog Input to Arduino
Red color wire - Power Supply from Arduino
Grey color wire - Ground Connection
Turquise color wire - Digital Output from Arduino
Green color wire - Control from Potentio meter
*/

/*
LCD pins
RS - 12
Enable - 11
D4 - 5
D5 - 4
D6 - 3
D7 - 2
*/
#include <LiquidCrystal.h>
 //peizo elecric buzzer connected pin
const int piezoPin = 7;
//Red - GREEN-Blue ie RBG connected pins
const int RGBpins[3] = {
  6,
  9,
  10
};
//tone length
const int toneLen = 100;
```



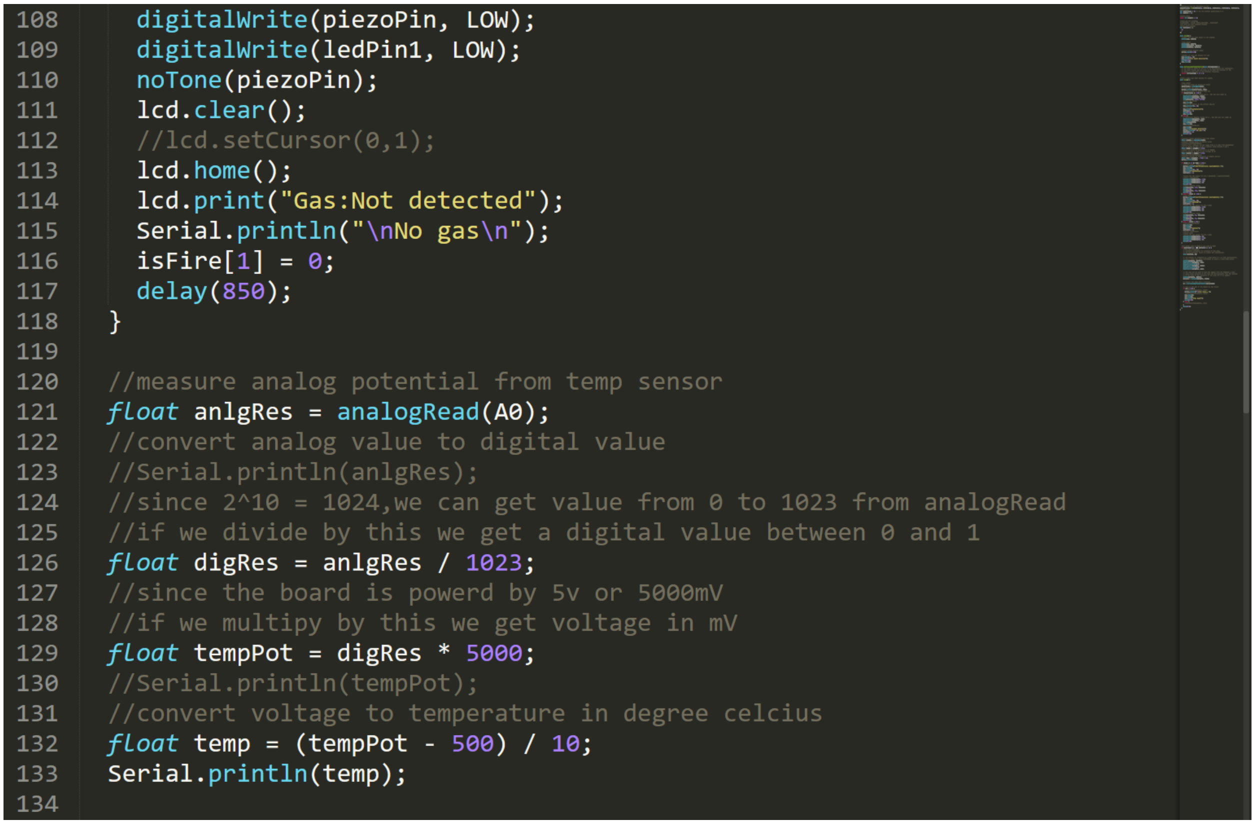
```
};
//tone length
const int toneLen = 100;
//LCD pins
const int LCDPins[6] = {
  12,
  11,
  5,
  4,
  3,
  2
};
//initialize the LCD interface as lcd
LiquidCrystal lcd(LCDPins[0], LCDPins[1], LCDPins[2], LCDPins[3], LCDPins[4],
//gas sensor
int sensorValue = 0; // Set the initial sensorValue to 0
int ledPin1 = 8;

//ultra sonic
const int pingPin = 13;

//bool-one hot encoding
//isFire[0] - isTemp high?,isFire[1] - isGasLeak?
//if both yes then evacuate message
//else alert message
int isFire[2] = {
```



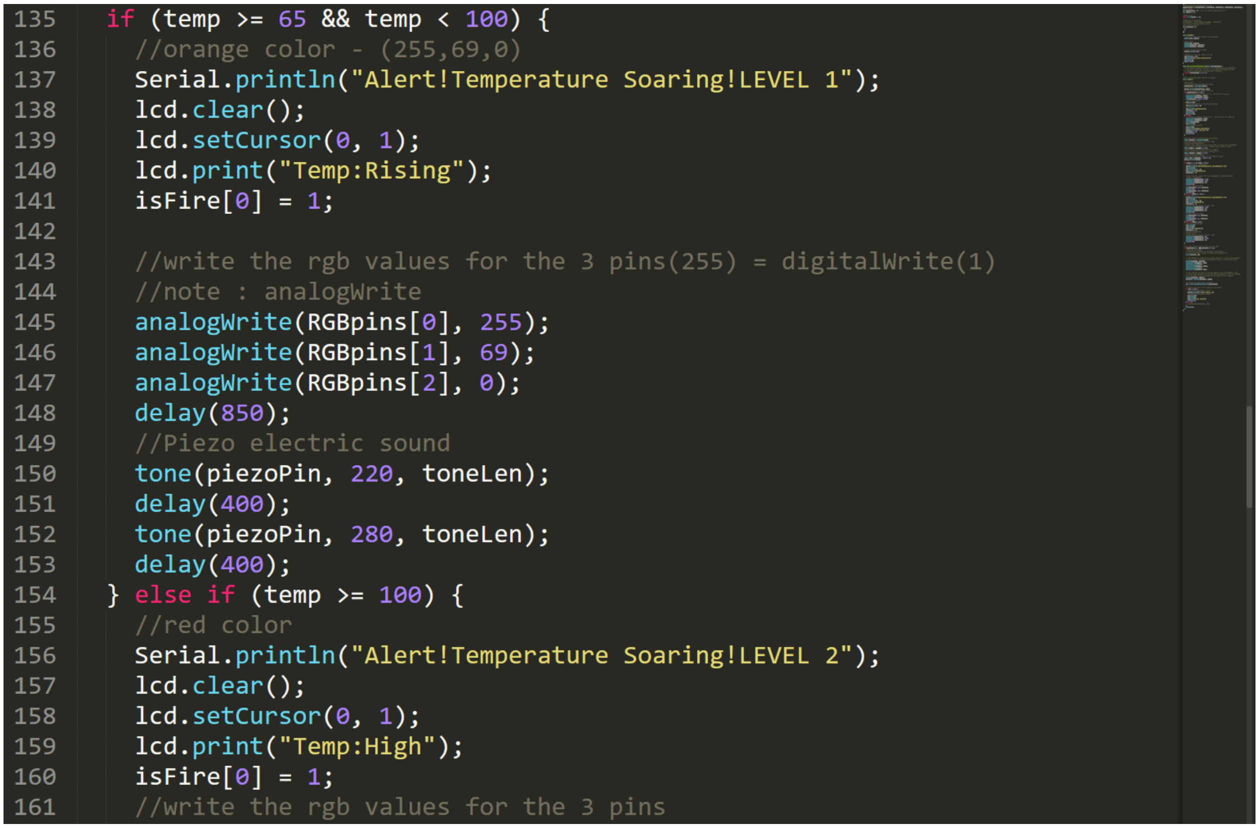
```
  0,
  0
};

void setup() {
  //input from the temp sensor to the arduino
  pinMode(A0, INPUT);

  //Gas sensor
  pinMode(A1, INPUT);
  pinMode(piezoPin, OUTPUT);
  pinMode(ledPin1, OUTPUT);

  //allow serial Monitor usage
  Serial.begin(9600);

  //number of rows and columns for LCD
  lcd.begin(16, 2);
  lcd.print("Fire Alarm Detection");
  delay(3000);
  lcd.clear();
}

long microsecondsToCentimeters(long microseconds) {
  // The speed of sound is 340 m/s or 29 microseconds per centimeter.
  // The ping travels out and back, so to find the distance of the
  // object we take half of the distance travelled.
  return microseconds / 29 / 2;
```



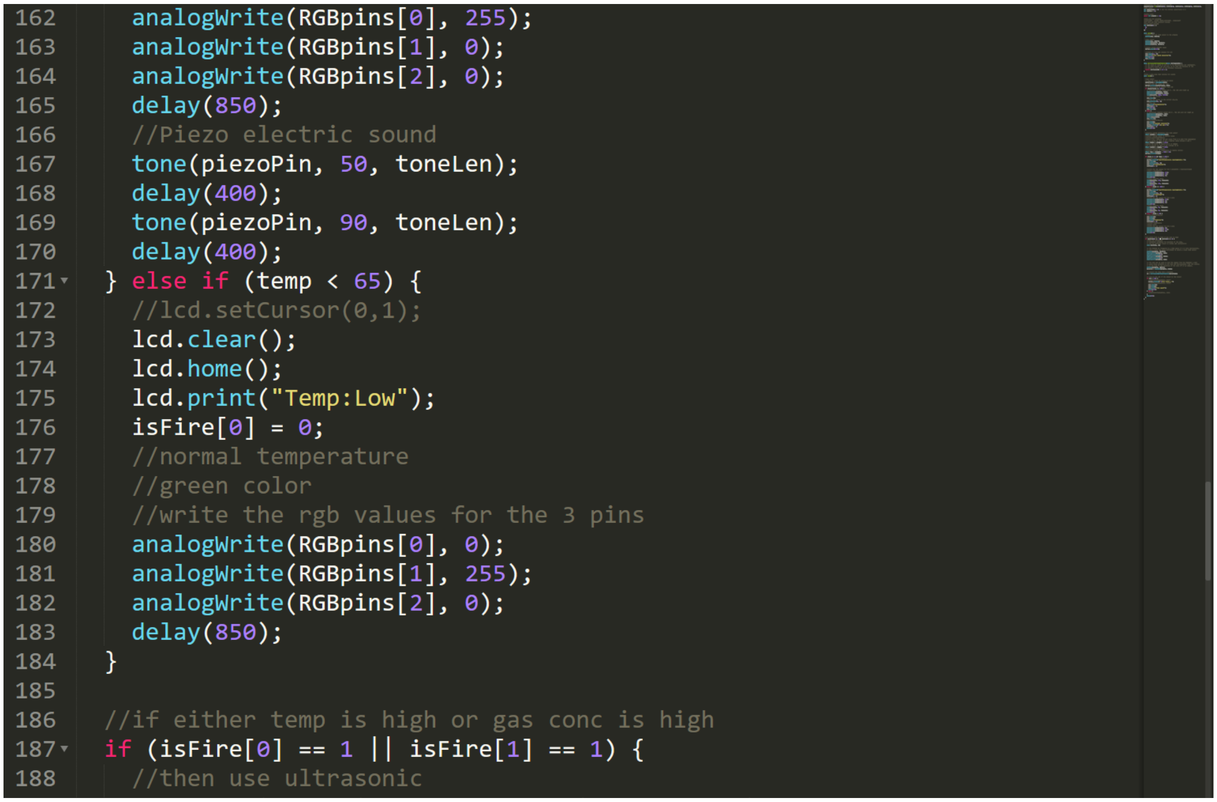
```
}

//like a game loop that listens for events
void loop() {

  //gas sensor
  // Read the input on analog pin 0(A1)
  sensorValue = analogRead(A1);
  // Print out the value you read
  Serial.println(sensorValue, DEC);
  // If sensorValue is greater than 250
  if (sensorValue >= 250) {
    // Activate digital output pin 8 - the LED will light up
    digitalWrite(ledPin1, HIGH);
    digitalWrite(piezoPin, HIGH);
    tone(piezoPin, 1000, 10000);
    //clear the screen
    lcd.clear();
    //point the cursor to the correct col,row
    lcd.setCursor(0, 1);
    //write lcd
    lcd.print("Gas:Detected");
    isFire[1] = 1;
    delay(850);
    lcd.clear();
  } else {
    // Deactivate digital output pin 8 - the LED will not light up
```



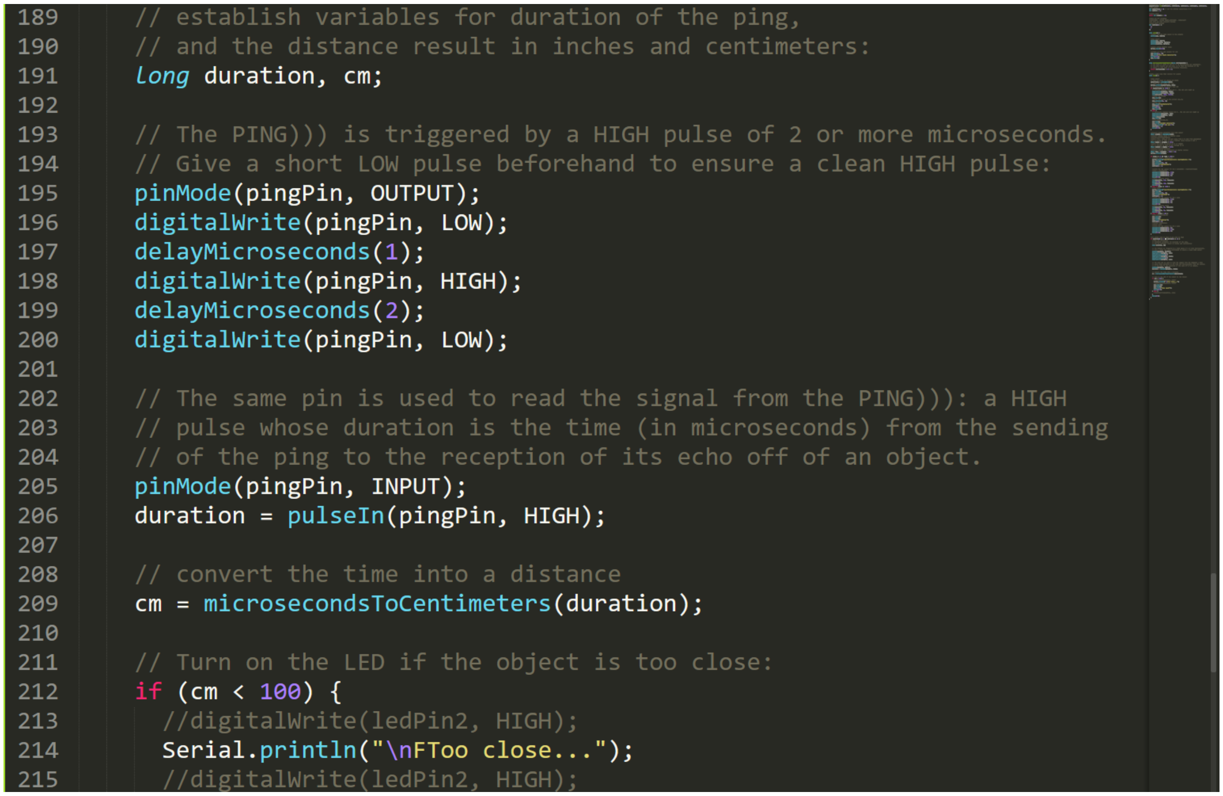
```
    digitalWrite(piezoPin, LOW);
    digitalWrite(ledPin1, LOW);
    noTone(piezoPin);
    lcd.clear();
    //lcd.setCursor(0,1);
    lcd.home();
    lcd.print("Gas:Not detected");
    Serial.println("\nNo gas\n");
    isFire[1] = 0;
    delay(850);
  }

  //measure analog potential from temp sensor
  float anlgRes = analogRead(A0);
  //convert analog value to digital value
  //Serial.println(anlgRes);
  //since 2^10 = 1024,we can get value from 0 to 1023 from analogRead
  //if we divide by this we get a digital value between 0 and 1
  float digRes = anlgRes / 1023;
  //since the board is powerd by 5v or 5000mV
  //if we multipy by this we get voltage in mV
  float tempPot = digRes * 5000;
  //Serial.println(tempPot);
  //convert voltage to temperature in degree celcius
  float temp = (tempPot - 500) / 10;
  Serial.println(temp);

```



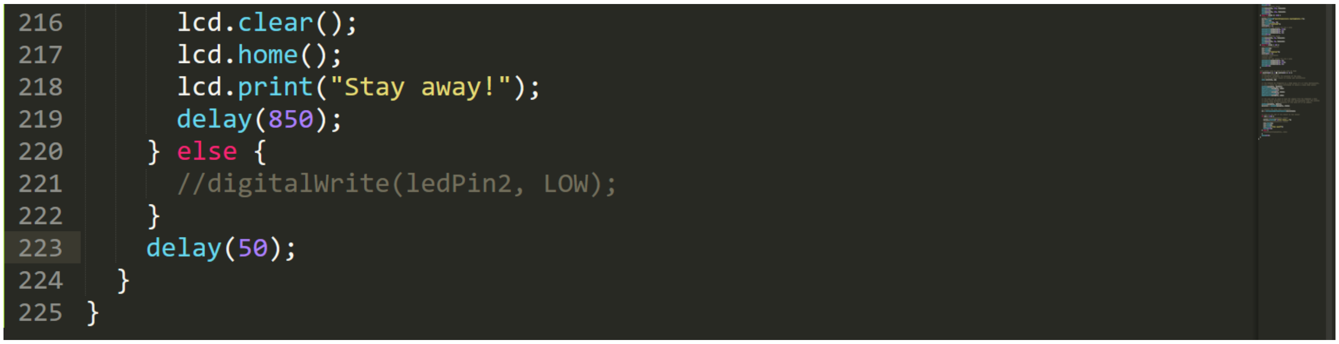
```
  if (temp >= 65 && temp < 100) {
    //orange color - (255,69,0)
    Serial.println("Alert!Temperature Soaring!LEVEL 1");
    lcd.clear();
    lcd.setCursor(0, 1);
    lcd.print("Temp:Rising");
    isFire[0] = 1;

    //write the rgb values for the 3 pins(255) = digitalWrite(1)
    //note : analogWrite
    analogWrite(RGBpins[0], 255);
    analogWrite(RGBpins[1], 69);
    analogWrite(RGBpins[2], 0);
    delay(850);
    //Piezo electric sound
    tone(piezoPin, 220, toneLen);
    delay(400);
    tone(piezoPin, 280, toneLen);
    delay(400);
  } else if (temp >= 100) {
    //red color
    Serial.println("Alert!Temperature Soaring!LEVEL 2");
    lcd.clear();
    lcd.setCursor(0, 1);
    lcd.print("Temp:High");
    isFire[0] = 1;
    //write the rgb values for the 3 pins
```



```
    analogWrite(RGBpins[0], 255);
    analogWrite(RGBpins[1], 0);
    analogWrite(RGBpins[2], 0);
    delay(850);
    //Piezo electric sound
    tone(piezoPin, 50, toneLen);
    delay(400);
    tone(piezoPin, 90, toneLen);
    delay(400);
  } else if (temp < 65) {
    //lcd.setCursor(0,1);
    lcd.clear();
    lcd.home();
    lcd.print("Temp:Low");
    isFire[0] = 0;
    //normal temperature
    //green color
    //write the rgb values for the 3 pins
    analogWrite(RGBpins[0], 0);
    analogWrite(RGBpins[1], 255);
    analogWrite(RGBpins[2], 0);
    delay(850);
  }

  //if either temp is high or gas conc is high
  if (isFire[0] == 1 || isFire[1] == 1) {
    //then use ultrasonic
```



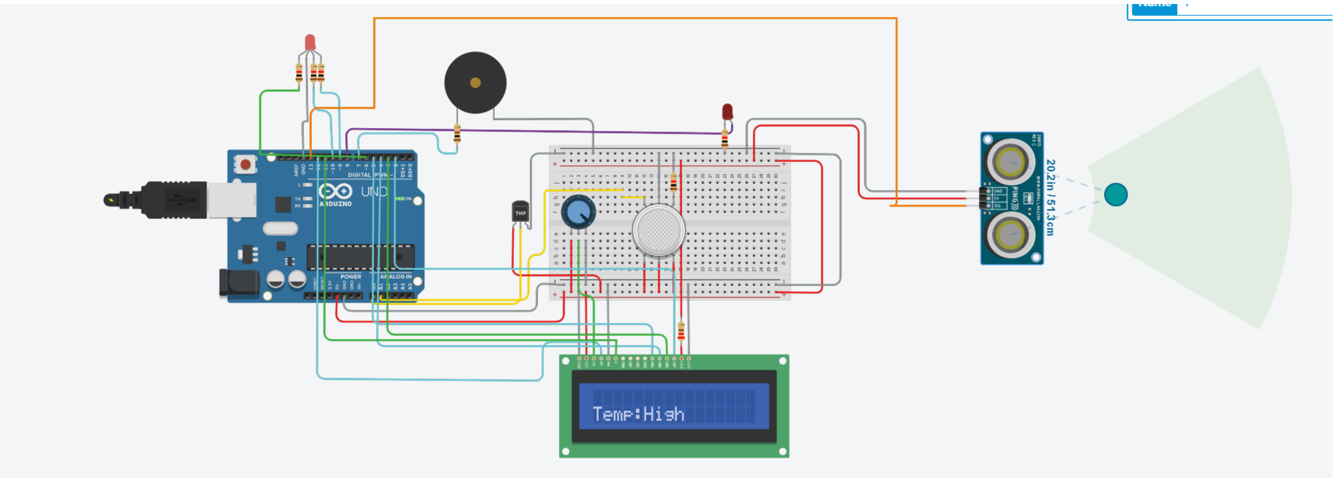
```
    // establish variables for duration of the ping,
    // and the distance result in inches and centimeters:
    long duration, cm;

    // The PING))) is triggered by a HIGH pulse of 2 or more microseconds.
    // Give a short LOW pulse beforehand to ensure a clean HIGH pulse:
    pinMode(pingPin, OUTPUT);
    digitalWrite(pingPin, LOW);
    delayMicroseconds(1);
    digitalWrite(pingPin, HIGH);
    delayMicroseconds(2);
    digitalWrite(pingPin, LOW);

    // The same pin is used to read the signal from the PING))): a HIGH
    // pulse whose duration is the time (in microseconds) from the sending
    // of the ping to the reception of its echo off of an object.
    pinMode(pingPin, INPUT);
    duration = pulseIn(pingPin, HIGH);

    // convert the time into a distance
    cm = microsecondsToCentimeters(duration);

    // Turn on the LED if the object is too close:
    if (cm < 100) {
      //digitalWrite(ledPin2, HIGH);
      Serial.println("\nFToo close...");
      //digitalWrite(ledPin2, HIGH);
```



```
      lcd.clear();
      lcd.home();
      lcd.print("Stay away!");
      delay(850);
    } else {
      //digitalWrite(ledPin2, LOW);
    }
    delay(50);
  }
}
```

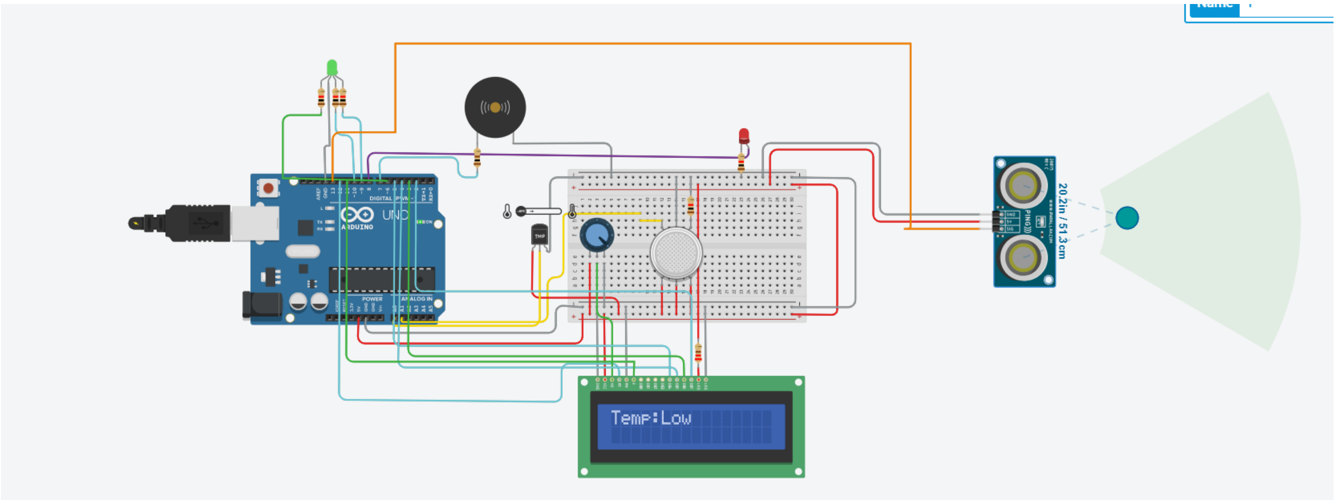
SCREENSHOTS OF THE OUTPUT:

1) **When Temperature is high and user is near room:**



```
Temp:High
```

2) **When Gas Concentration is high and user is near room:**



```
Temp:Low
```

REFERENCES:

Temperature Sensor - https://bc-robotics.com/tutorials/using-a-tmp36-temperature-sensor-with-arduino/#:~:text=The%20TMP36%20temperature%20sensor%20is,making%20it%20a%20popular%20choice.

Gas Sensor - https://www.instructables.com/How-to-use-MQ2-Gas-Sensor-Arduino-Tutorial/#:~:text=Introduction%3A%20How%20to%20Use%20MQ2%20Gas%20Sensor%20%2D%20Arduino%20Tutorial&text=The%20output%20is%20an%20analog,%2Calcohol%2C%20hydrogen%20and%20smoke.

LCD - https://www.arduino.cc/en/Tutorial/LibraryExamples/HelloWorld

Buzzer - https://www.youtube.com/watch?v=xBLYrbYIxLA