TABLE OF CONTENTS		
Sl.No	TOPIC	PAGE No
1.	ABSTRACT OF THE PROJECT	2
2.	CIRCUIT DIAGRAM	5
3.	ARDUINO CODE	6
4.	SCREENSHOTS OF THE OUTPUT	7
5.	REFERENCES	8

ABSTRACT OF THE PROJECT:

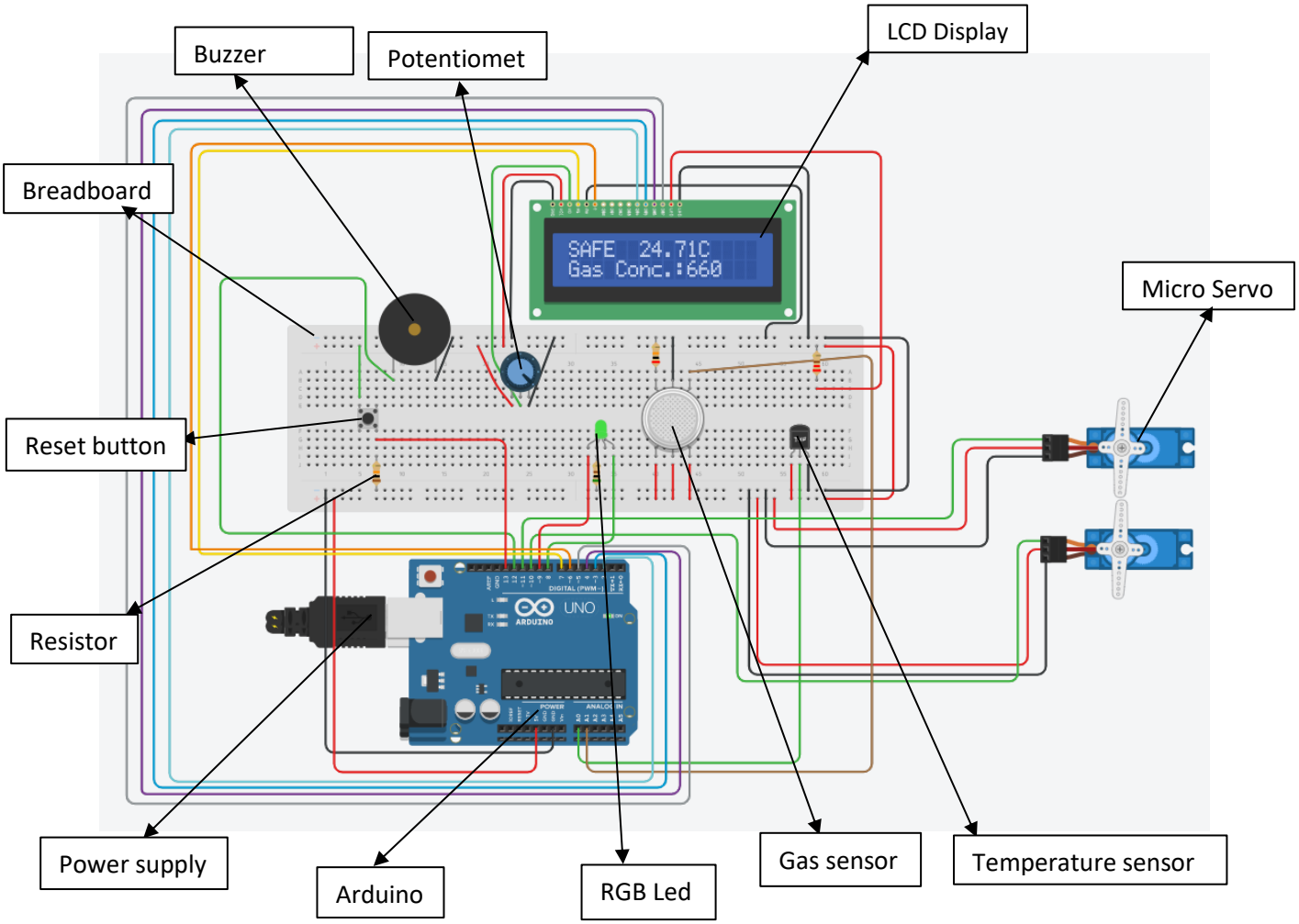
Fire alarm detection project demonstrates the use of Arduino to build a fire detection system using the sensors which measure temperature of the environment in which it is present and also measure concentration of gas that might cause fire (methane, alcohol) or is caused because of fire (carbon monoxide, carbon dioxide etc.).

The fire alarm detection system remains idle unless there is a fire outbreak. The system in idle condition glows green LED, keeps the micro servos(doors) closed, displays SAFE message on the LCD display along with the current temperature and gas concentration in the room/hall where the sensors are installed.

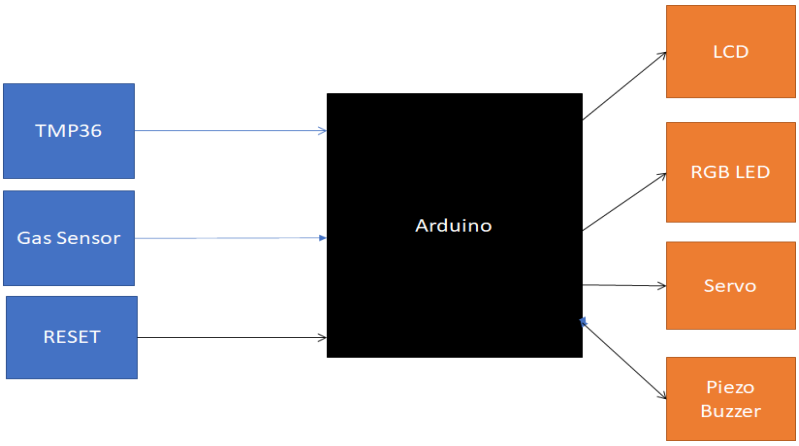
With a gas leak or a fire outbreak in a given room, halls or any part of the building where the sensors are installed, the Arduino will start the blinking red LED, shows warnings on the LCD display, turns on the siren, in our case a buzzer and opens the micro servos(doors) to indicate people to move out of the building into a safe area to avoid any kind of burns or injuries.

The system remains in alert mode until the fire is extinguished leading to reduction in temperature and gas concentration in the room. The system only switches back to idle mode after the fire is extinguished and the reset button is long pressed.

CIRCUIT DIAGRAM:



Block Diagram:



ARDUINO CODE:

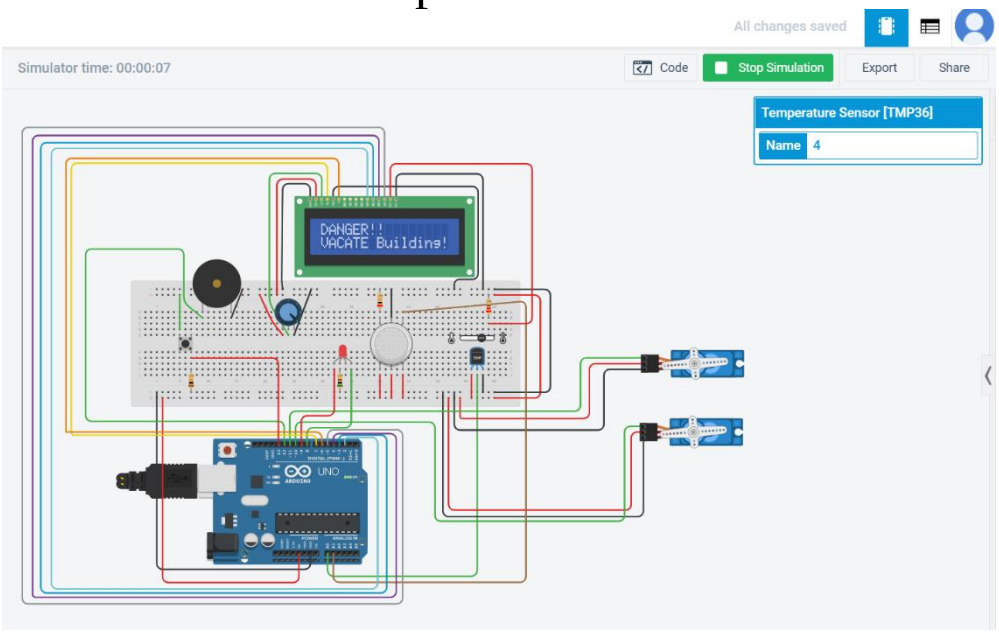
```
#include<Servo.h>//header file for servo
#include <LiquidCrystal.h>//header file for LCD

//first of all, we will use the TMP36 which is a temperature sensor that outputs
//a voltage that's proportional to the ambient temperature.

// We'll use analog input 0 to measure the temperature sensor's signal pin.
//Temparature Sensor
const int temperaturePin = 0; //The output of tmp36 is connected to A0 of arduino
const int buzzer = 12; //buzzer is connected to D12 on the arduino
const int resetbtn = 13;

//Gas Sensor
int gasSensorPin=A1;//Gas sensor output is connected to A1 of Arduino
int sensorval;//For storing the value sensed by gas sensor
int repeat = 0;

//Doors
Servo servo1, servo2;
int servo1Pin=11;
int servo2Pin=10;

//RGB LED
int red_led=9;//Red terminal of RGB LED is connected to D9 of Arduino
int green_led=8;//Green terminal of RGB LED is connected to D8 of Arduino

//LCD
LiquidCrystal lcd(7, 6, 2, 3, 4, 5);
//Sets the interfacing pins on Arduino that are connected to LCD
//7-Rs,6-E(Enable), 5,4,3,2 are the inputs->4 bit mode

//reset button
int buttonstate = 0;
```

```
void setup()
{
   pinMode(buzzer, OUTPUT);//set the pin connected to the buzzer as
an output

   servo1.attach(servo1Pin);
   servo2.attach(servo2Pin);
   servo1.write(90);//Intially both doors are closed(i.e, 90 degrees)
   servo2.write(90);
   delay(2000);

   pinMode(red_led,OUTPUT);
   pinMode(green_led,OUTPUT);
   pinMode(resetbtn,INPUT);

   lcd.begin(16,2);//initialisation of 16*2 LCD
}

void loop()
{
   //for buzzer and tmp36 temp sensor
   float voltage, degreesC;
   voltage = getVoltage(temperaturePin);
   degreesC = (voltage - 0.5) * 100.0;

   sensorval=analogRead(gasSensorPin);
   buttonstate = digitalRead(resetbtn);

   //check if reset button is being pressed
   if(buttonstate == HIGH) {
      repeat = 0;//to repeat in alert mode or not
   }


   if(degreesC>37 || sensorval>700 || repeat == 1)
   {
      repeat = 1;
      digitalWrite(buzzer, LOW);
      tone(buzzer, 800, 800);
      delay(200);
      tone(buzzer,600,800);

      servo1.write(0);
```

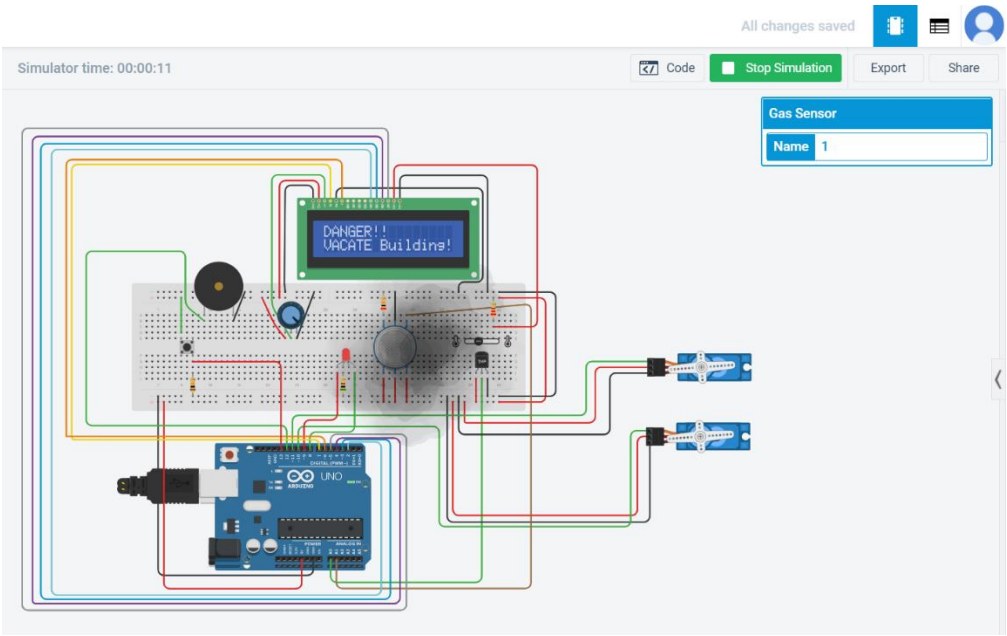
```
        servo2.write(0);
        delay(1000);

        digitalWrite(red_led,HIGH);
        digitalWrite(green_led,LOW);
        delay(1000);
        digitalWrite(red_led,LOW);

        lcd.clear();
        lcd.setCursor(0,0);//row 0 column 0
        lcd.print("DANGER!!");
        lcd.setCursor(0,1);//row 1 column 0
        lcd.print("VACATE Building!");
    }
    else{
        servo1.write(90);
        servo2.write(90);
        delay(1000);

        digitalWrite(green_led,HIGH);
        digitalWrite(red_led,LOW);

        lcd.clear();
        lcd.setCursor(0,0);//column 0 row 0
        lcd.print("SAFE");
        lcd.setCursor(6,0);//column 6 row 0
        lcd.print(degreesC);
        lcd.print("C");
        lcd.setCursor(0,1);
        lcd.print("Gas Conc.:");
        lcd.print(sensorval);
    }
}
float getVoltage(int pin)
{
    return (analogRead(pin) * 0.004882814);
}
```

SCREENSHOTS OF THE OUTPUT:

Increase in temperature:



Increase in Gas concentration level:



REFERENCES

- Importance: https://www.unifourfire.com/blog/importance-of-fire-alarms
- LCD 16*2: https://lastminuteengineers.com/arduino-1602-character-lcd-tutorial/
- Gas Sensor: https://components101.com/articles/introduction-to-gas-sensors-types-working-and-applications
- TMP36 temperature Sensor: https://bc-robotics.com/tutorials/using-a-tmp36-temperature-sensor-with-arduino/