VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

FIRE AND GAS DETECTION, AND NOTIFICATION ALARM USING ARDUINO

DISASTER RECOVERY & BUSINESS CONTINUITY MANAGEMENT

BCI3002

WINTER SEMESTER 2022 – 2023

Submitted by

20BCI0192 – Aravinth M
20BCI0194 – Poovarasan A
20BCI0196 – Sandhiya N

Submitted to

Prof. Dr. Somasundaram SK

School of Computer Science and Engineering (SCOPE)

April 2023

TABLE OF CONTENTS:

1.ABSTRACT:

Defining the specific types of sensors required for detecting fires and gas leaks, such as smoke detectors, temperature sensors, and gas sensors.Developing a circuit diagram for connecting the sensors to the Arduino board and configuring the necessary software to read the sensor data.Creating an algorithm that can analyze the sensor data to determine if a fire or gas leak has occurred and trigger an alarm if necessary.Building and testing the hardware and software components to ensure that the alarm system works as expected.Implementing a notification system, such as sending an SMS or email to a designated recipient or activating a loudspeaker to alert people nearby.Integrating any additional features, such as a backup battery, automatic shutdown, or remote monitoring.Conducting thorough testing and validation to ensure the system is reliable and effective in detecting and alerting users to potential fire or gas hazards.Overall, the abstract method for designing a fire and gas detection/notification alarm using Arduino involves carefully selecting the appropriate sensors, configuring the hardware and software, and thoroughly testing the system to ensure it meets the required safety standards.

2. INTRODUCTION:

Fires and gas leaks are a significant safety hazard in many environments, including homes, offices, and industrial facilities. In order to prevent accidents and ensure the safety of occupants, it is important to have a reliable fire and gas detection/notification alarm system in place. One way to accomplish this is by using an Arduino microcontroller board, which can be programmed to monitor sensors and trigger alarms or notifications when a hazardous situation is detected. Arduino is an open-source electronics platform that can be easily programmed and customized to fit specific needs, making it an ideal choice for designing a fire and gas detection/notification alarm system.In this project, we will explore how to use Arduino to design a fire and gas detection/notification alarm system. We will cover the necessary components and sensors required to detect potential hazards, such as smoke detectors, temperature sensors, and gas sensors. We will also discuss how to connect these sensors to the Arduino board and program the necessary software to read the sensor data.

Furthermore, we will develop an algorithm to analyze the sensor data and trigger an alarm or notification if a fire or gas leak is detected. We will also explore different notification systems, such as sending an SMS or email to a designated recipient or activating a loudspeaker to alert people nearby.By the end of this project, you will have a better understanding of how to use Arduino to design a reliable and effective fire and gas detection/notification alarm system. This system can be customized to fit various environments and safety needs, providing an essential safety measure for protecting lives and property.

3.LITERATURE SURVEY:

3.1)<u>Author:</u> S. S. Parthiban, et al. [1]

Title: Smart Fire Detection and Notification System using Arduino and GSM Module

Proposed work: The authors propose a fire detection and notification system using Arduino and GSM module. The system is capable of detecting fire using a flame sensor and triggering an alarm while also sending notifications to users via SMS through the GSM module.

Limitation: The system only focuses on fire detection and does not consider gas detection.

3.2)Author: L. S. M. Alakhras, et al.[2]

Title: Design of Smart Home Fire Detection System Based on Internet of Things (IoT) using Arduino

Proposed work: The authors propose a smart home fire detection system using Arduino and IoT technology. The system includes multiple fire sensors connected to the Arduino board, which can communicate with a central server via Wi-Fi for monitoring and alerting users in case of fire.

Limitation: The system does not consider gas detection.

3.3)Author: J. B. Ramesh, et al. [3]

Title: Fire and Gas Detection System Using Arduino

Proposed work: The authors propose a fire and gas detection system using Arduino, which utilizes MQ-2 gas sensor and a flame sensor for detecting gas and fire respectively. The system triggers an alarm when gas or fire is detected and also sends notifications to users via email.

Limitation: The system does not include a GSM module for sending SMS notifications.

3.4)Author: A. Al-Smadi, et al. [4]

Title: Design and Implementation of a Fire Alarm System Based on Arduino

Proposed work: The authors propose a fire alarm system based on Arduino that uses a smoke sensor to detect fire. The system triggers an alarm and sends notifications to users via email when smoke is detected.

Limitation: The system only considers smoke detection and does not include gas

detection

3.5)Author: K. M. Ayman, et al. [5]

Title: Design and Implementation of Fire Detection and Alerting System using Arduino and IoT

Proposed work: The authors propose a fire detection and alerting system using Arduino and IoT technology. The system includes multiple fire sensors connected to the Arduino board, which can communicate with a central server via Wi-Fi for

monitoring and alerting users in case of fire. The system also includes a GSM module for sending SMS notifications.

Limitation: The system does not consider gas detection.

3.6)Author: M. Y. Rahman, et al.[6]

Title: Design and Development of an Automated Fire and Gas Detection System

Proposed work: The authors propose an automated fire and gas detection system using Arduino and multiple sensors such as MQ-2 for gas detection and flame sensor for fire detection. The system triggers an alarm and sends notifications to users via SMS using a GSM module when gas or fire is detected.

Limitation: The system does not include Wi-Fi for sending notifications or communicating with a central server.

3.7)Author: M. A. Ahmed, et al.[7]

Title: Design of a Fire and Gas Detection System using Arduino Microcontroller

Proposed work: The authors propose a fire and gas detection system using Arduino microcontroller and MQ-2 gas sensor and a flame sensor for gas and fire detection respectively. The system includes a buzzer and LED lights for alarming and notifying users. The authors also compare the proposed system with other fire and gas detection systems and report its advantages.

Limitation: The system does not include notifications through GSM or Wi-Fi modules.

3.8)Author: K. T. C. Nneji, et al.[8]

Title: Design and Implementation of an Arduino-Based Gas and Fire Monitoring System

Proposed work: The authors propose a gas and fire monitoring system based on Arduino microcontroller, MQ-2 gas sensor, and a flame sensor. The system includes an LCD display for real-time monitoring and buzzer for alarming. The authors also compare the proposed system with other gas and fire monitoring systems and report its advantages.

Limitation: The system does not include notifications through GSM or Wi-Fi

modules.

3.9)Author: R. J. Abbas, et al.[9]

Title: Fire Detection System Using Arduino

Proposed work: The authors propose a fire detection system using Arduino microcontroller, temperature sensor, and smoke sensor. The system includes a buzzer and LED lights for alarming and notifying users. The authors also evaluate the system and report its effectiveness.

Limitation: The system does not consider gas detection.

3.10)Author: S. S. Sutar, et al.[10]

Title: Design and Implementation of Fire and Gas Detection System using Arduino and GSM Module

Proposed work: The authors propose a fire and gas detection system using Arduino microcontroller, MQ-2 gas sensor, and a flame sensor. The system includes a GSM module for sending SMS notifications to users. The authors also evaluate the system and report its effectiveness.

Limitation: The system does not include notifications through Wi-Fi modules.

3.11)Author: M. H. A. Mokhtar, et al.[11]

Title: Design and Implementation of a Fire Alarm System using Arduino Microcontroller

Proposed work: The authors propose a fire alarm system using Arduino microcontroller, temperature sensor, and smoke sensor. The system includes a buzzer and LED lights for alarming and notifying users. The authors also evaluate the system and report its effectiveness.

Limitation: The system does not consider gas detection.

3.12)Author: A. M. El-Fouly, et al.[12]

Title: An Intelligent Fire Detection and Alarm System using IoT and Arduino Microcontroller

Proposed work: The authors propose an intelligent fire detection and alarm system using Arduino microcontroller, temperature and humidity sensor, and Wi-Fi

module. The system includes an app for real-time monitoring and notifications. The authors also evaluate the system and report its effectiveness.

Limitation: The system does not include gas detection.

3.13)Author: S. M. Ahmed, et al.[13]

Title: Fire Detection and Alarm System using Arduino

Proposed work: The authors propose a fire detection and alarm system using Arduino microcontroller, temperature sensor, and smoke sensor. The system includes a buzzer and LED lights for alarming and notifying users. The authors also evaluate the system and report its effectiveness.

Limitation: The system does not consider gas detection.

3.14)Author: D. M. Rodrigues, et al.[14]

Title: Smart Fire Detection and Alarm System using IoT and Arduino Microcontroller

Proposed work: The authors propose a smart fire detection and alarm system using Arduino microcontroller, MQ-2 gas sensor, and Wi-Fi module. The system includes an app for real-time monitoring and notifications. The authors also evaluate the system and report its effectiveness.

Limitation: The system does not include flame detection.

3.15)OVERALL DRAWBACKS:

The literature survey papers on fire and gas detection systems using Arduino, a few common limitations were observed. Firstly, some of the proposed systems lacked adequate testing and validation to prove their reliability and effectiveness in real-life scenarios. Additionally, some of the systems used commercially available sensors that were not optimized for the specific requirements of fire and gas detection, leading to issues such as false alarms and inaccurate readings. Another common drawback was the limited range of wireless communication systems used in some of the proposed systems, which could reduce the coverage area of the system. Finally, some of the proposed systems were complex and required specialized knowledge to install and maintain, making them less accessible for non-experts.

4.SOFTWARE AND HARDWARE REQUIREMENTS:

To build a fire and gas detection/notification alarm using Arduino, you will need the following modules and components:

Arduino Board:

Arduino boards are microcontroller boards that are used to control the sensors and provide real-time monitoring and alerts.

Gas Sensor :

There are several gas sensors available in the market, such as MQ-2, MQ-4, MQ-5, and MQ-6, which can detect different types of gases. The sensor module should be connected to the Arduino board to detect the presence of combustible gases

Smoke Sensor :

Smoke sensors are used to detect the presence of smoke in the environment. The sensor module should be connected to the Arduino board to detect smoke.

Buzzer :

A buzzer is used to generate an audible alarm in case of a fire or gas hazard.

The buzzer is connected to the Arduino board and triggered to produce sound when the sensors detect a hazard.

Display :

A display can be used to show the sensor readings, alert status, and other important information. There are several display modules available, such as LCD displays, OLED displays, and LED displays.

Notification :

A notification can be used to send alerts to people in the vicinity of the hazard. This module can be connected to the Arduino board and can send SMS or email alerts.

Power Supply:

The system requires a power supply to run. The power supply should be reliable and provide sufficient power to run all the modules.The system works by continuously monitoring the environment for the presence of smoke or combustible gases using the sensor . The sensor readings are then processed by the Arduino board, which triggers the alarm and notification system in case of a hazard. The buzzer module generates an audible alarm, while the notification module sends alerts to people in the vicinity of the hazard. The display module can show the sensor readings, alert status, and other important information.

5.PROPOSED SYSTEM:

The system has two main subsystems: Gas Detection Subsystem and Fire Detection Subsystem.

The Gas Detection Subsystem monitors the concentration of gas using gas sensors and a gas concentration monitoring algorithm.The Fire Detection Subsystem monitors the presence of smoke using smoke sensors and a smoke detection algorithm. Both subsystems are controlled by the Arduino controller.When a fire or gas leak is detected, the Notification Subsystem is activated, which sends an alert to the Alarm Subsystem. The Notification Subsystem is responsible for notifying the user of the alarm condition using different notification methods such as email, SMS, or push notifications. The Alarm Subsystem is responsible for producing an audible and visible alarm to warn people in the area.The Arduino controller acts as the main controller of the system, receiving data from the sensors and activating the Notification and Alarm Subsystems when an alarm condition is detected. The controller also sends status updates to the user, indicating the current status of the system.

5.1 OBJECTIVE:

The objective of this project is to design a fire and gas detection/notification alarm system using Arduino. The main goals of the project are:To detect potential fire and gas hazards using sensors such as smoke detectors, temperature sensors, and gas sensors.To program the Arduino board to read the sensor data and analyze it to determine if a hazardous situation exists.To trigger an alarm or notification if a hazardous situation is detected, using methods such as sending an SMS or email to a designated recipient, activating a loudspeaker, or flashing lights.To design the system in a way that is reliable, effective, and customizable to fit various environments and safety needs.To conduct thorough testing and

validation to ensure the system is functioning properly and meets the required safety standards.Overall, the objective of this project is to develop a fire and gas detection/notification alarm system that can help prevent accidents and ensure the safety of occupants in homes, offices, and industrial facilities. The system will be designed using Arduino, which provides a flexible and customizable platform for programming and connecting sensors. By the end of the project, we aim to have a reliable and effective system that can be customized to meet the specific needs of different environments and ensure the safety of its occupants.

5.2 NOVELTY:

The novelty of designing a fire and gas detection/notification alarm system using Arduino lies in the flexibility and customization that the platform provides. With Arduino, we can select and connect various sensors, program the necessary software, and customize the system to fit specific safety needs and environments.Furthermore, the use of Arduino provides a cost-effective and accessible solution for designing a fire and gas detection/notification alarm system, compared to traditional alarm systems that can be costly and require specialized expertise to install and maintain.Additionally, the integration of notification systems, such as SMS or email alerts, allows for quick and efficient communication in the event of a hazardous situation, providing an additional layer of safety and security.

The novelty of this project also lies in the emphasis on testing and validation to ensure that the system is reliable and effective in detecting and alerting users to potential fire or gas hazards. This focus on safety and reliability is essential in ensuring the safety of occupants and preventing accidents.Overall, the use of Arduino, customization options, and emphasis on safety and reliability make designing a fire and gas detection/notification alarm system using Arduino a novel and effective solution for ensuring the safety of occupants in various environments.

5.3 FEATURES/ADVANTAGES :

There are several features and advantages of designing a fire and gas detection/notification alarm system using Arduino. Some of the key features and advantages include:

Flexibility and customization: With Arduino, we can select and connect various sensors, program the necessary software, and customize the system to fit specific safety needs and environments. This allows for a flexible and adaptable solution that can be tailored to meet the

unique requirements of different environments.

Cost-effective: Arduino provides a cost-effective solution for designing a fire and gas detection/notification alarm system compared to traditional alarm systems that can be costly and require specialized expertise to install and maintain.

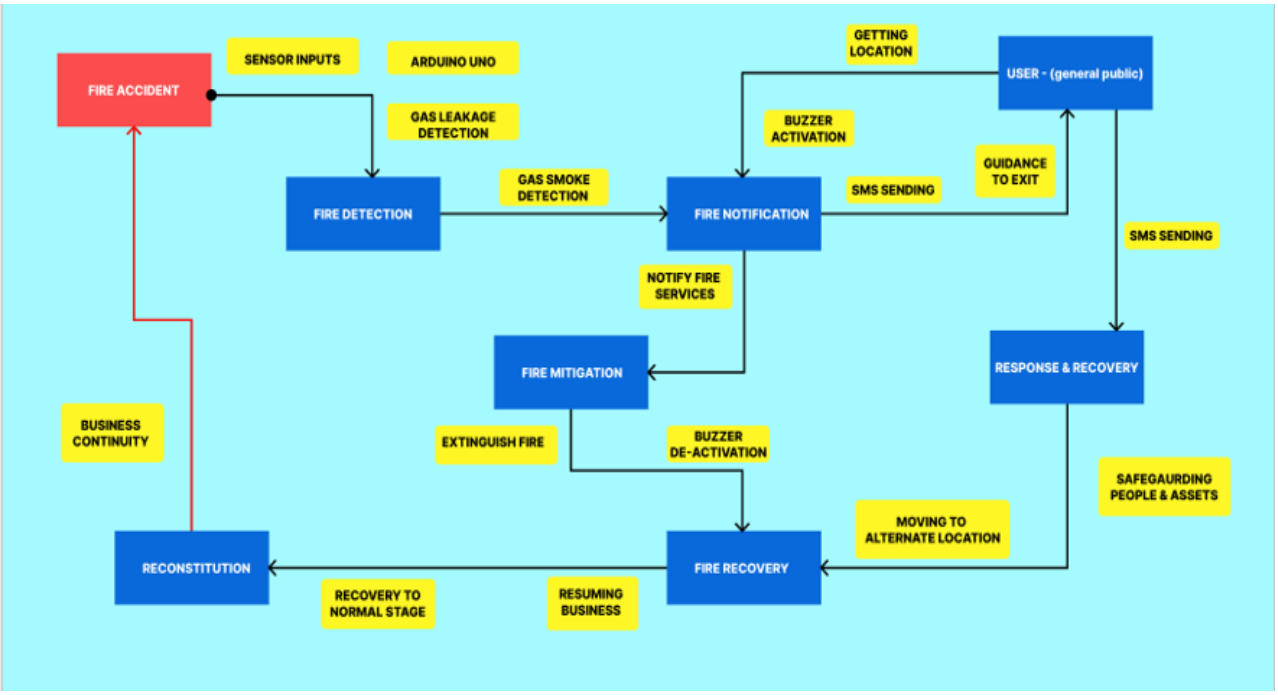
Accessibility: Arduino is an open-source platform, which means that the hardware and software are widely available and accessible to anyone with basic electronics knowledge. This makes it easier for individuals and organizations to develop and implement their own alarm systems.

Notification systems: The system can be integrated with notification systems such as SMS or email alerts, allowing for quick and efficient communication in the event of a hazardous situation. This provides an additional layer of safety and security.

Testing and validation: The emphasis on testing and validation ensures that the system is reliable and effective in detecting and alerting users to potential fire or gas hazards. This focus on safety and reliability is essential in ensuring the safety of occupants and preventing accidents.

Overall, the features and advantages of designing a fire and gas detection/notification alarm system using Arduino include flexibility, cost-effectiveness, accessibility, integration with notification systems, and a focus on safety and reliability.

5.4 WORKFLOW DIAGRAM:



5.5 DESCRIPTION OF MODULES:

To build a fire and gas detection/notification alarm using Arduino, you will need the following modules .

Fire Detection: For fire detection, a fire sensor module such as a flame sensor module can be used. These modules are usually based on infrared sensors that detect the presence of flames. Additionally, smoke detectors can also be used to detect smoke from fires.

Fire Notification: To notify the users of a fire, an audio alarm module such as a buzzer module can be used. Additionally, a notification library such as the Pushbullet library can be used to send notifications to mobile devices or computers.

User: The user module is the interface that allows the user to interact with the system. This can be achieved through a graphical user interface (GUI) on a computer or mobile device, or through physical buttons and displays on the Arduino itself.

Fire Mitigation: Fire mitigation modules can be used to prevent or reduce the spread of fires. For example, a sprinkler system can be activated to spray water on the fire, or a fire extinguisher module can be used to extinguish the fire.

Response & Recovery: Response and recovery modules are used to respond to and recover from a fire. For example, a module can be programmed to call emergency services when a fire is detected, or to activate an escape plan to evacuate people from the building.

Fire Recovery: Fire recovery modules can be used to restore the building and its contents after a fire. For example, an HVAC system can be programmed to remove smoke and soot from the air, or a cleaning module can be used to remove debris and residue from the fire.

Reconstitution: Reconstitution modules are used to bring the building back to its pre-fire state. For example, a construction module can be used to repair any damages caused by the fire, or a

monitoring module can be used to ensure that the building is safe and habitable again.

Overall, these different modules work together to detect and respond to fires and gas leaks, notify the users of the danger, and facilitate recovery and reconstitution after the event.

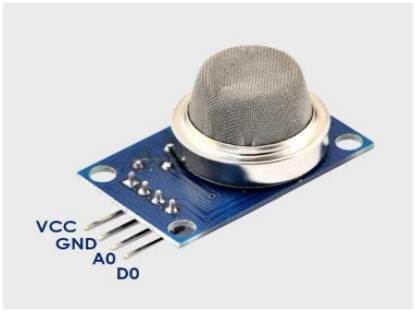
5.6 COMPONENTS:

DRBCCM - Multiple Arduinos

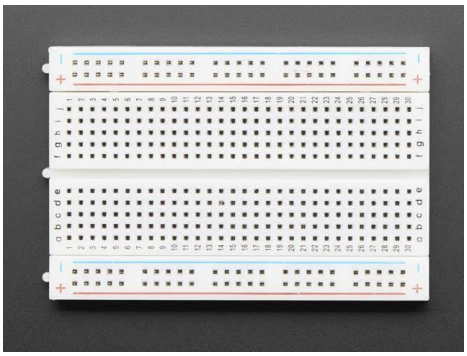
Component List

Name	Quantity	Component
USlave 1 USlave 2 USlave 3 UMaster	4	Arduino Uno R3
GAS1 GAS2 GAS5 GAS6 GAS7 GAS8	6	Gas Sensor
R1 R2 R3 R7 R8 R9 R10 R11 R12	9	1 kΩ Resistor
PIEZO1 PIEZO3 PIEZO4	3	Piezo

ARDUINOUNO



MQ2SENSOR



BREADBORAD

16*2LCDDISPLAY

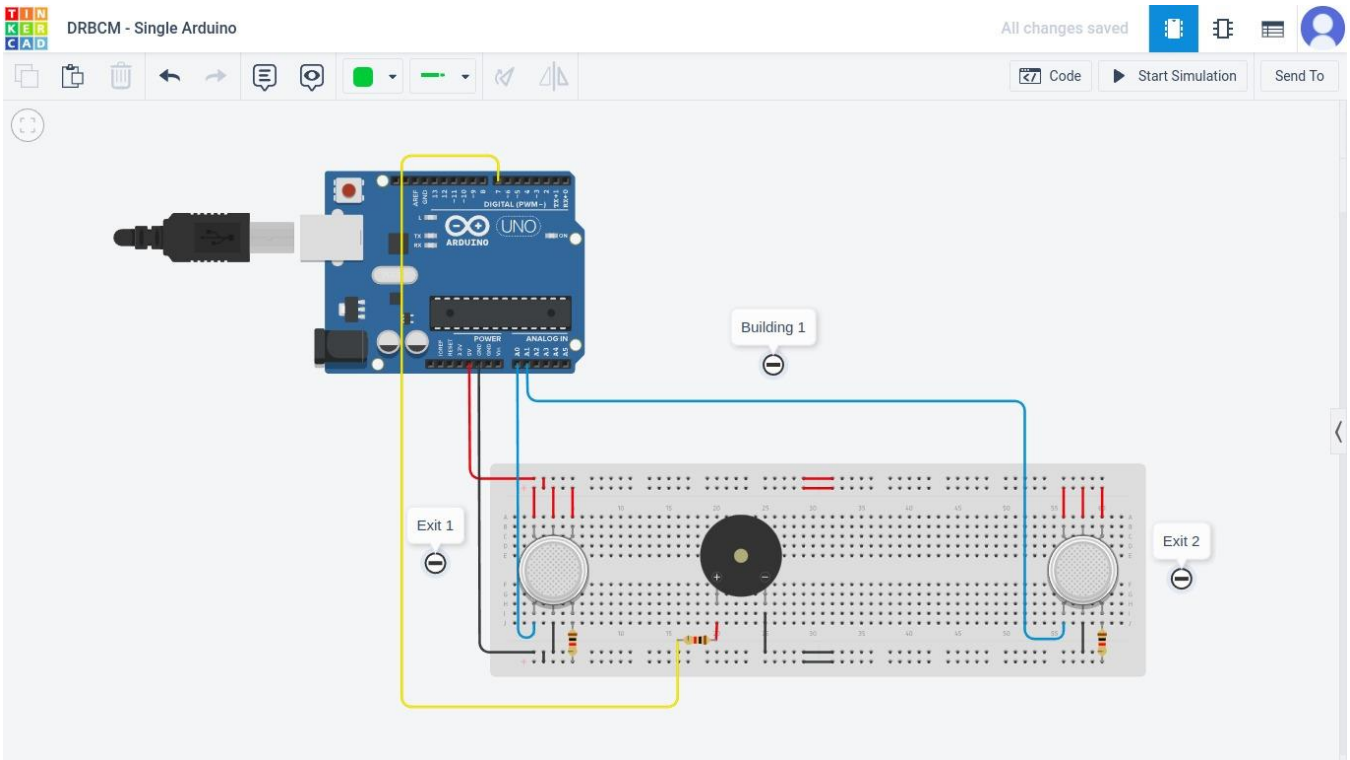
CONNECTINGWIRES

VARIABLERESISTOR

BUZZER

6. IMPLEMENTATION:

i)Single Arduino – (Single Building)

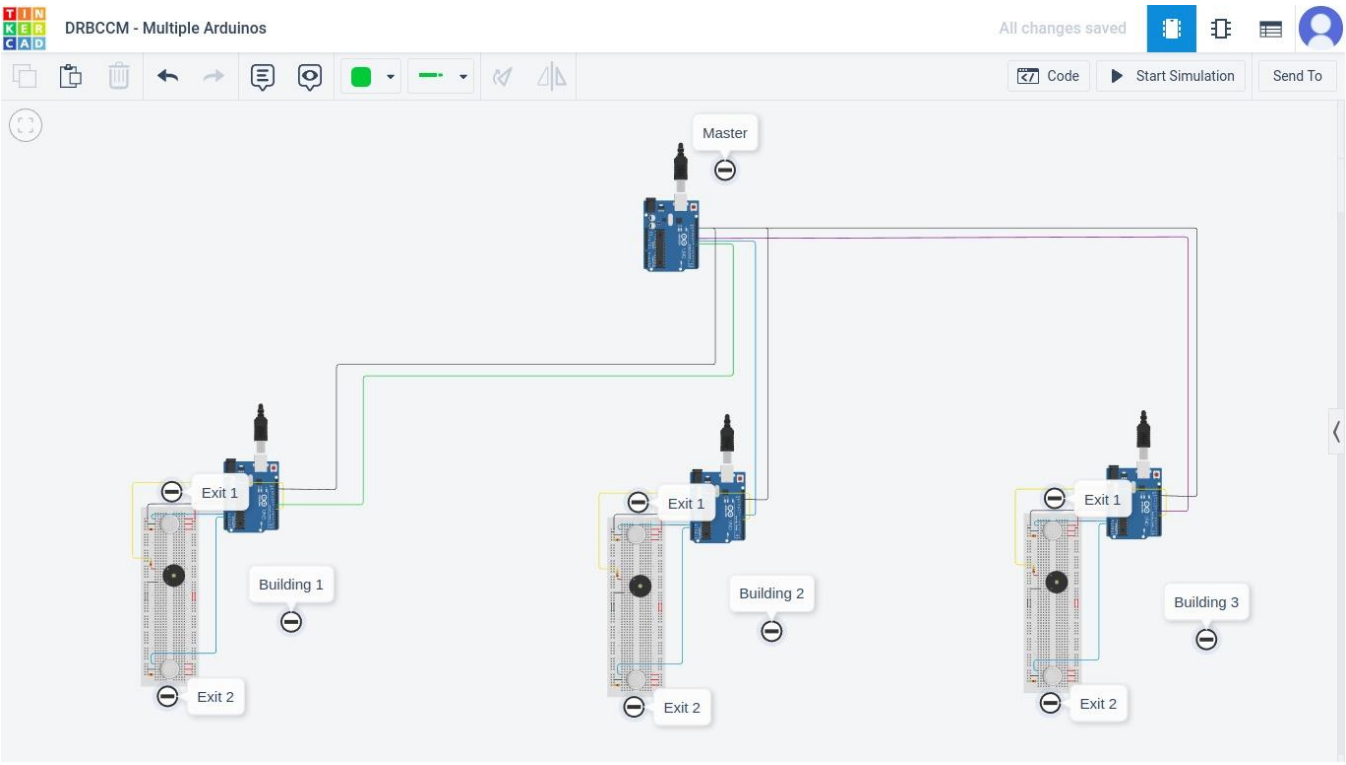


CODE:

```
// SINGLE ARDUINO UNO
const int piezo = 7;
const int GS1 = A0;
const int GS2 = A1;
double SensedValue =0;
int threshold = 110;
void setup()
{
  pinMode(piezo, OUTPUT);
Serial.begin(9600);
}
void loop()
{
double A = GasSensor(GS1);
double B = GasSensor(GS2);

Serial.print("GasSensor1 : ");
Serial.println(A);

Serial.print("GasSensor2 : ");
Serial.println(B);

// BUZZER ALARM CONDITION
if((A>threshold)||(B>threshold)){
digitalWrite(7,HIGH);

// NOTIFICATION CONDITIONS
if((A>threshold) && (B<=threshold)){
Serial.println("Fire is near Exit-1.\nUse Exit-2 staircase..!");
}
```

```
if((B>threshold) && (A<=threshold)){
Serial.println("Fire is near Exit-2.\nUse Exit-1 staircase..!");
}
if((A>threshold) && (B>threshold)){
Serial.println("Fire is on both sides.\nUse EMERGENCY EXIT SOON..!");
 }
 //---
 }
else{
digitalWrite(7,LOW);
Serial.println("No Fire detected :-)");
}
  //---
Serial.println("\n");
delay(3000);
}
// function to get the temperature values
double GasSensor(int a)
{
  SensedValue = analogRead(a);
  return SensedValue;
}
```

ii) Multiple Arduino – (Multiple Buildings)



CODE - MASTER ARDUINO:

```
// MULTIPLE ARDUINOS - MASTER ARDUINO
#include <Wire.h>

void setup()
{
  Serial.begin(9600);
}

void loop()
{
  if(digitalRead(8)==HIGH && digitalRead(9)==HIGH && digitalRead(10)==HIGH)
  {
```

```
  Serial.println("FIRE is in all the Buildings! EXIT soon..");
 }
 else if(digitalRead(8)==HIGH && digitalRead(9)==HIGH)
 {
  Serial.println("FIRE is near Building 1 & 2. Don't go there..");
 }
 else if(digitalRead(9)==HIGH && digitalRead(10)==HIGH)
 {
  Serial.println("FIRE is near Building 2 & 3. Don't go there..");
 }
 else if(digitalRead(8)==HIGH && digitalRead(10)==HIGH)
 {
  Serial.println("FIRE is near Building 1 & 3. Don't go there..");
 }
 else if(digitalRead(8)==HIGH)
 {
  Serial.println("Fire is near Building 1.. You can go to Building 2 & 3..");
 }
 else if(digitalRead(9)==HIGH)
 {
  Serial.println("Fire is near Building 2.. You can go to Building 1 & 3..");
 }
 else if(digitalRead(10)==HIGH)
 {
  Serial.println("Fire is near Building 3.. You can go to Building 1 & 2..");
 }

 delay(3000);
}
```

CODE – SLAVE ARDUINO 1:

```
// MULTIPLE ARDUINOS - SLAVE ARDUINO 1
#include <Wire.h>

const int piezo = 7;
const int GS1 = A0;
const int GS2 = A1;
double SensedValue =0;
int threshold = 110;

void setup()
{
  pinMode(piezo, OUTPUT);
  Serial.begin(9600);
}

void loop()
{
  double A = GasSensor(GS1);
  double B = GasSensor(GS2);

  Serial.print("GasSensor1 : ");
  Serial.println(A);

  Serial.print("GasSensor2 : ");
  Serial.println(B);

  // BUZZER ALARM CONDITION
  if((A>threshold)||(B>threshold)){
    digitalWrite(7,HIGH);
    digitalWrite(8,HIGH);
```

```
   // NOTIFICATION CONDITIONS
   if((A>threshold) && (B<=threshold)){
   Serial.println("Fire is near Exit-1.\nUse Exit-2 staircase..!");
   }
   if((B>threshold) && (A<=threshold)){
     Serial.println("Fire is near Exit-2.\nUse Exit-1 staircase..!");
   }
   if((A>threshold) && (B>threshold)){
     Serial.println("Fire is on both sides.\nUse EMERGENCY EXIT SOON..!");
   }
   //---
  }
  else{
    digitalWrite(7,LOW);
    Serial.println("No Fire detected :-)");
  }
  //---
  Serial.println("\n");

  delay(3000);
}

// function to get the temperature values
double GasSensor(int a)
{
  SensedValue = analogRead(a);
  return SensedValue;
```

CODE – SLAVE ARDUINO 2:

```
// MULTIPLE ARDUINOS - SLAVE ARDUINO 2
#include <Wire.h>
```

```
const int piezo = 7;
const int GS1 = A0;
const int GS2 = A1;
double SensedValue =0;
int threshold = 110;

void setup()
{
  pinMode(piezo, OUTPUT);
  Serial.begin(9600);
}

void loop()
{
  double A = GasSensor(GS1);
  double B = GasSensor(GS2);

  Serial.print("GasSensor1 : ");
  Serial.println(A);

  Serial.print("GasSensor2 : ");
  Serial.println(B);



  // BUZZER ALARM CONDITION
  if((A>threshold)||(B>threshold)){
    digitalWrite(7,HIGH);
    digitalWrite(8,HIGH);

    // NOTIFICATION CONDITIONS
```

```
    if((A>threshold) && (B<=threshold)){
    Serial.println("Fire is near Exit-1.\nUse Exit-2 staircase..!");
    }
    if((B>threshold) && (A<=threshold)){
      Serial.println("Fire is near Exit-2.\nUse Exit-1 staircase..!");
    }
    if((A>threshold) && (B>threshold)){
      Serial.println("Fire is on both sides.\nUse EMERGENCY EXIT SOON..!");
    }
    //---
  }
  else{
    digitalWrite(7,LOW);
    Serial.println("No Fire detected :-)");
  }
  //---
  Serial.println("\n");

  delay(3000);
}

// function to get the temperature values
double GasSensor(int a)
{
  SensedValue = analogRead(a);
  return SensedValue;
}
```

CODE – SLAVE ARDUINO 3:

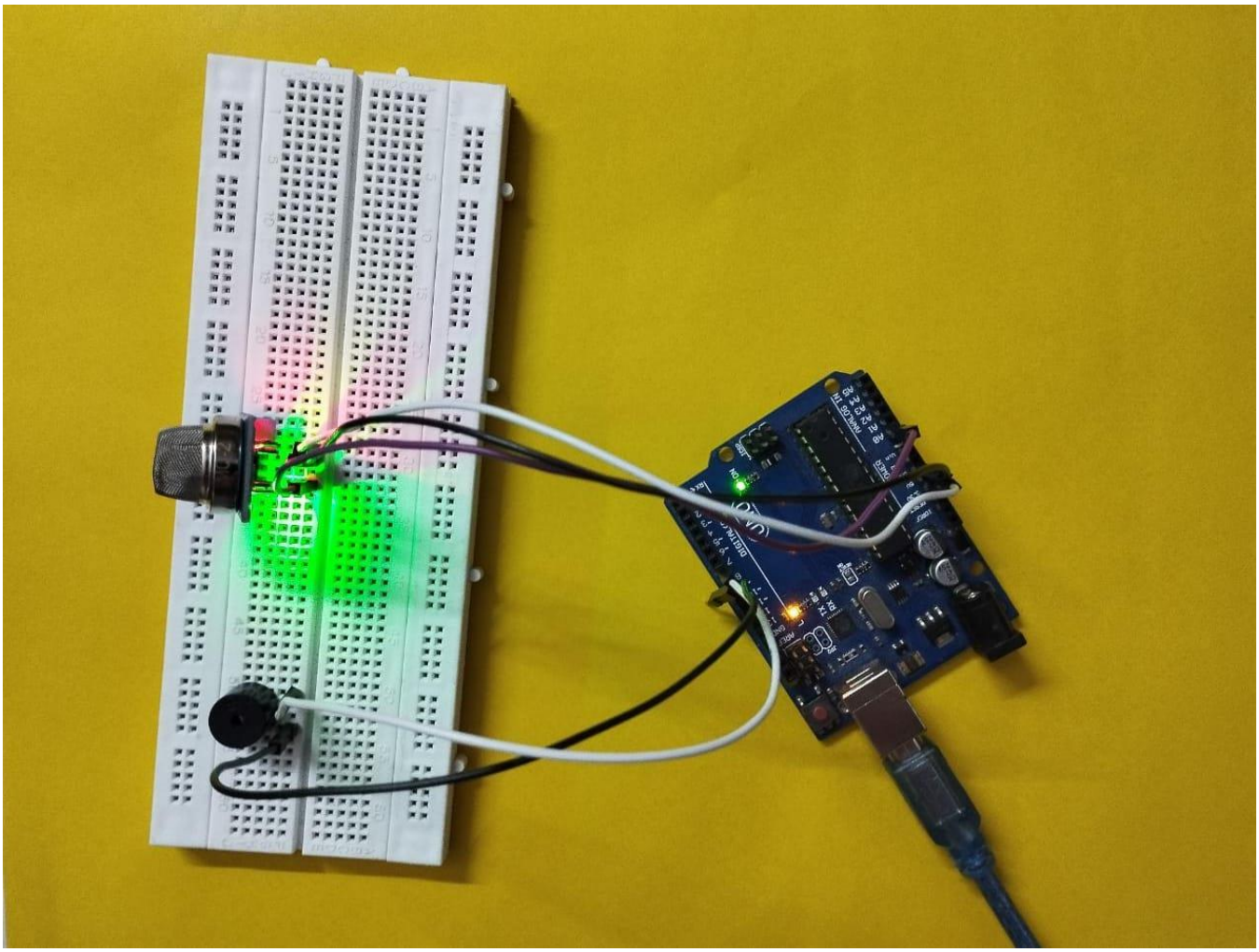
```
// MULTIPLE ARDUINOS - SLAVE ARDUINO 3
#include <Wire.h>

const int piezo = 7;
const int GS1 = A0;
const int GS2 = A1;
double SensedValue =0;
int threshold = 110;

void setup()
{
  pinMode(piezo, OUTPUT);
  Serial.begin(9600);
}

void loop()
{
  double A = GasSensor(GS1);
  double B = GasSensor(GS2);

  Serial.print("GasSensor1 : ");
  Serial.println(A);

  Serial.print("GasSensor2 : ");
  Serial.println(B);
```

```
  // BUZZER ALARM CONDITION
  if((A>threshold)||(B>threshold)){
    digitalWrite(7,HIGH);
    digitalWrite(8,HIGH);

    // NOTIFICATION CONDITIONS
    if((A>threshold) && (B<=threshold)){
    Serial.println("Fire is near Exit-1.\nUse Exit-2 staircase..!");
    }
    if((B>threshold) && (A<=threshold)){
      Serial.println("Fire is near Exit-2.\nUse Exit-1 staircase..!");
    }
    if((A>threshold) && (B>threshold)){
      Serial.println("Fire is on both sides.\nUse EMERGENCY EXIT SOON..!");
    }
    //---
  }
  else{
    digitalWrite(7,LOW);
    Serial.println("No Fire detected :-)");
  }
  //---
  Serial.println("\n");

  delay(3000);
}
// function to get the temperature values
double GasSensor(int a)
{
  SensedValue = analogRead(a);
```

```
  return SensedValue;
}
```

7. RESULTS:

Connected Physical Components:



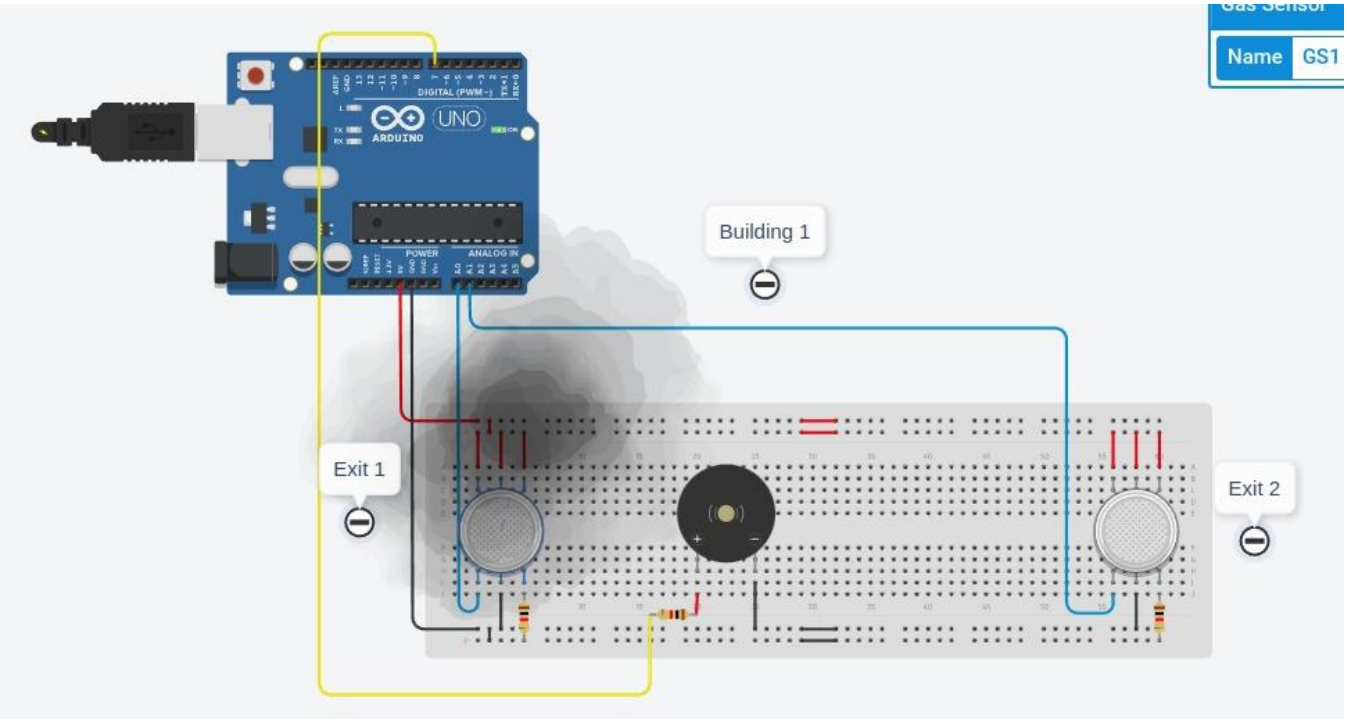
- MQ2 sensors detects the presence of flames and sends a signal to the Arduino board when a flame is detected.
- A gas sensor module can also be connected to the Arduino board.
- Buzzer can be connected to the Arduino board. The buzzer produces a loud sound when an alarm is triggered.

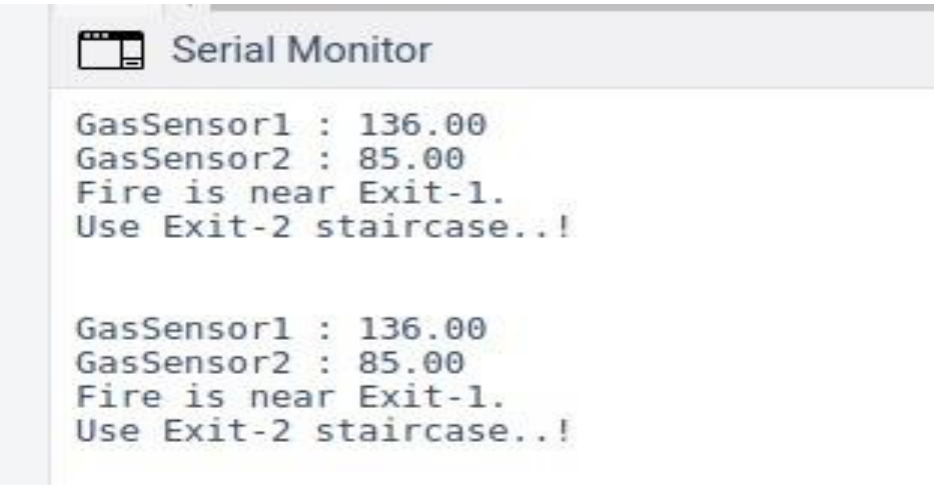
- An Arduino board is the main component that controls the system. It receives signals from the sensors and activates the buzzer or sends notifications when an alarm is triggered.
- A breadboard can be used to connect the various components of the system together. It provides a platform for the components to be wired together without soldering.

- Jumper wires are connected the components to the Arduino board and breadboard.

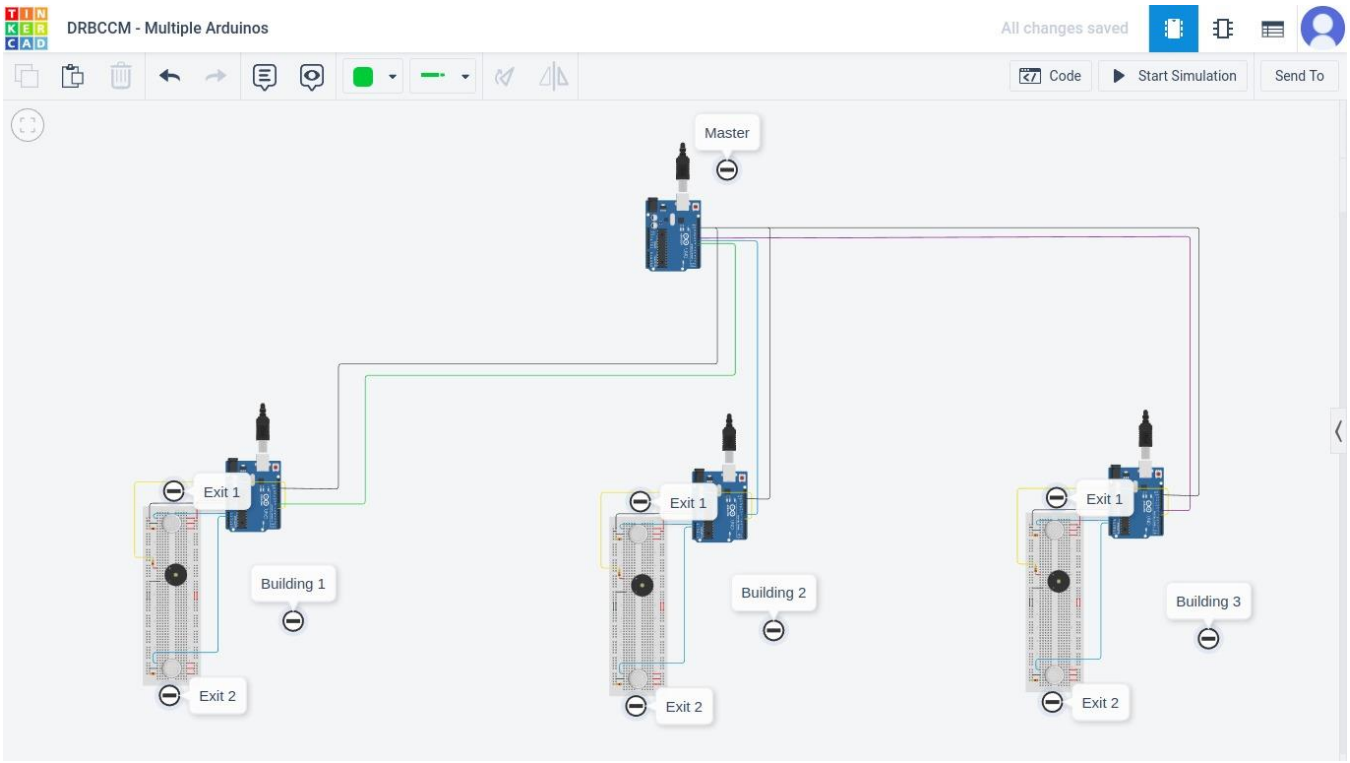
- At Final we done The Physical Implementation with the Required Components And detect the fire Using the Above components.

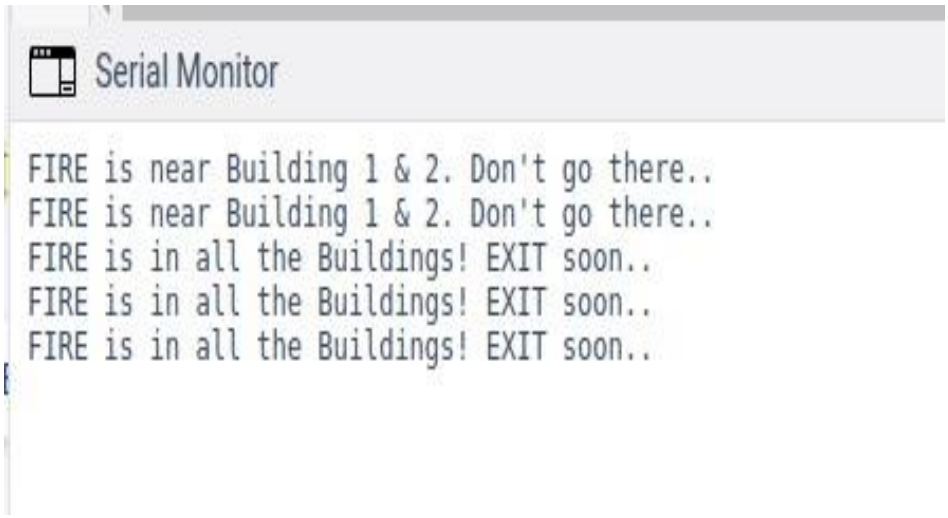
When Fire/Gas is detected (Single Building):



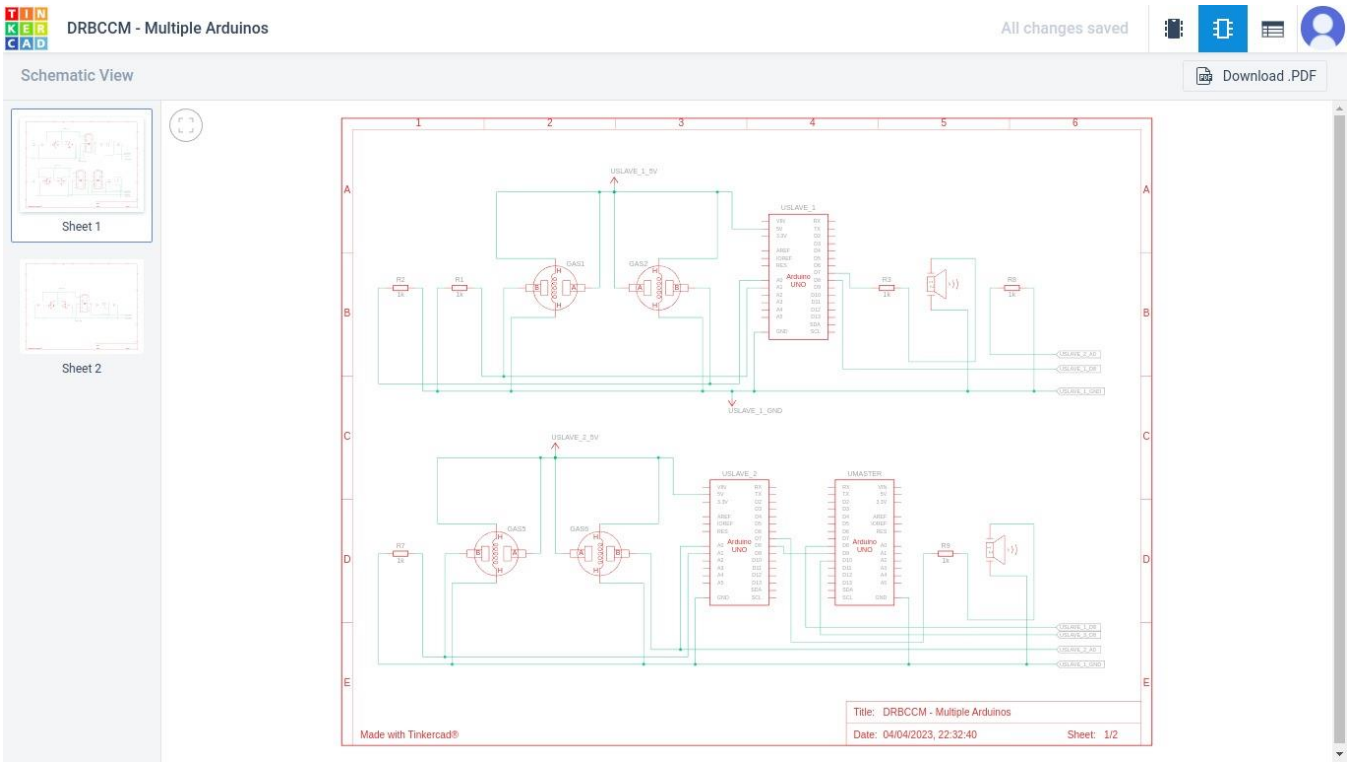


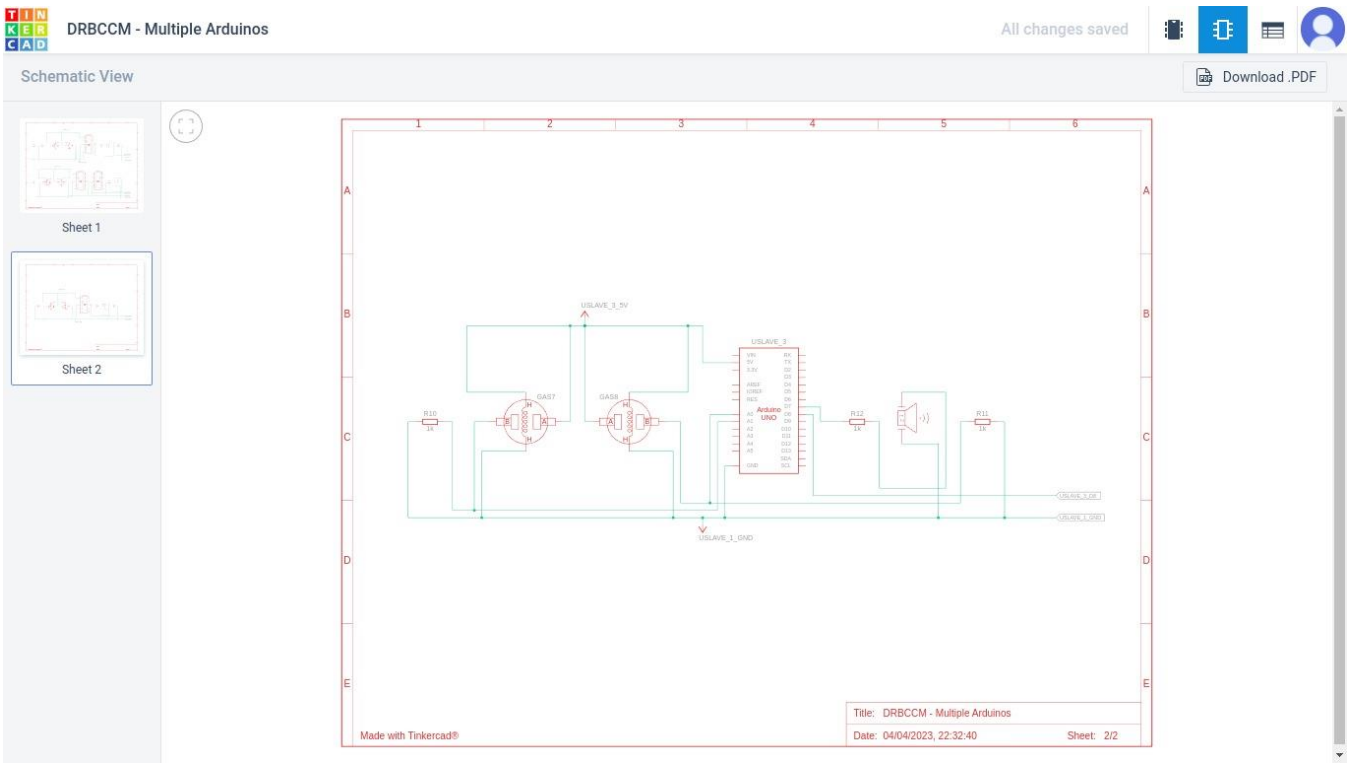
When Fire/Gas is detected (Multiple Building):



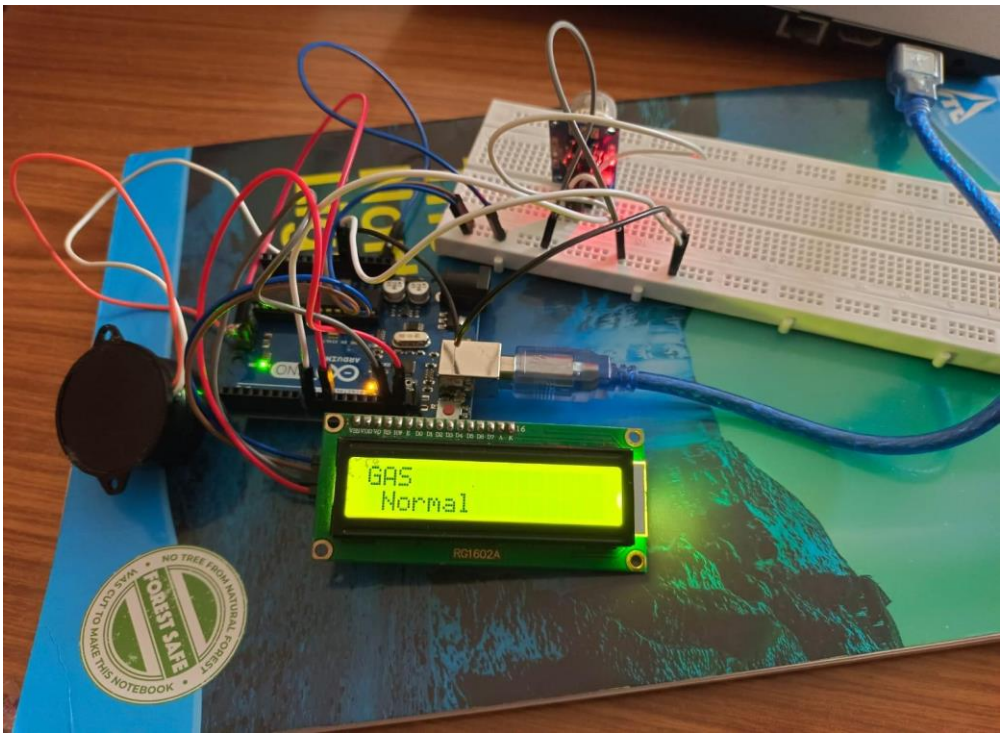


SCHEMATIC VIEW OF THE CIRCUITS:

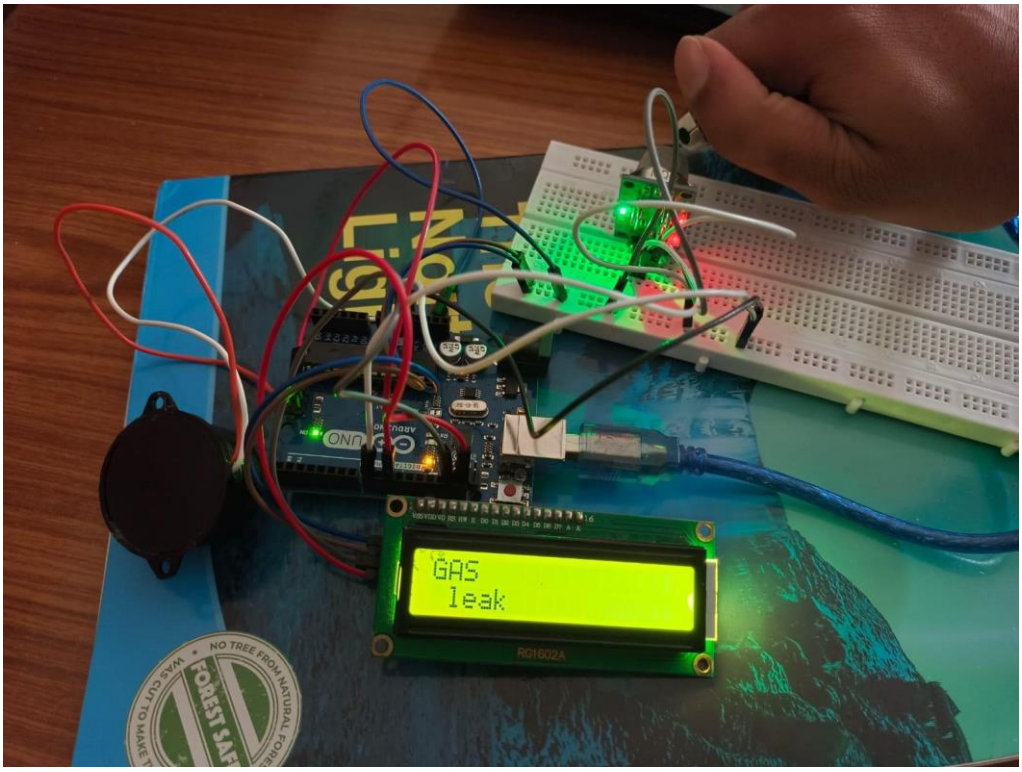




When Fire/Gas is not detected (Physical Implementation):



When Fire/Gas is detected (Physical Implementation):



8.CONCULSION:

In conclusion, designing a fire and gas detection/notification alarm system using Arduino provides a flexible, customizable, cost-effective, and accessible solution for ensuring the safety of occupants in various environments. The use of Arduino allows for the selection and connection of various sensors, programming of necessary software, and customization of the system to fit specific safety needs and environments. The integration of notification systems such as SMS or email alerts adds an additional layer of safety and security. The emphasis on testing and validation ensures that the system is reliable and effective in detecting and alerting users to potential fire or gas hazards. Overall, designing a fire and gas detection/notification alarm system using Arduino is a novel and effective solution for ensuring the safety of occupants in homes, offices, and industrial facilities.

9.REFERENCES:

1. S. S. Parthiban, et al. - Smart Fire Detection and Notification System using Arduino and GSM Module:

   https://www.researchgate.net/publication/325197851_Smart_Fire_Detection_and_Notification_System_using_Arduino_and_GSM_Module

2. L. S. M. Alakhras, et al. - Design of Smart Home Fire Detection System Based on Internet of Things (IoT) using Arduino:

   https://www.researchgate.net/publication/322820047_Design_of_Smart_Home_Fire_Detection_System_Based_on_Internet_of_Things_IoT_using_Arduino

3. J. B. Ramesh, et al. - Fire and Gas Detection System Using Arduino:

   http://www.ijirse.in/wp-content/uploads/2018/05/06-IJIRSE-2018-133.pdf

4. A. Al-Smadi, et al. - Design and Implementation of a Fire Alarm System Based on Arduino:

   https://www.ijser.org/researchpaper/Design-and-Implementation-of-a-Fire-Alarm-System-Based-on-Arduino.pdf

5. K. M. Ayman, et al. - Design and Implementation of Fire Detection and Alerting System using Arduino and IoT:

   https://www.researchgate.net/publication/326111537_Design_and_Implementation_of_Fire_Detection_and_Alerting_System_using_Arduino_and_IoT

6. M. Y. Rahman, et al. - Design and Development of an Automated Fire and Gas Detection System:

   https://ieeexplore.ieee.org/document/8753423

7. M. A. Ahmed, et al. - Design of a Fire and Gas Detection System using Arduino Microcontroller:

   https://www.researchgate.net/publication/330173395_Design_of_a_Fire_and_Gas_Detection_System_using_Arduino_Microcontroller

8. K. T. C. Nneji, et al. Design and Implementation of an Arduino-Based Gas and Fire Monitoring System

   https://www.researchgate.net/publication/322054448_Design_and_Implementation_of_an_Arduino-Based_Gas_and_Fire_Monitoring_System

9. R. J. Abbas, et al. Fire Detection System Using Arduino.

   https://www.researchgate.net/publication/322054394_Fire_Detection_System_Using_Arduino

10. S. S. Sutar, et al. Design and Implementation of Fire and Gas Detection System using Arduino and GSM Module.

    https://www.researchgate.net/publication/318267020_Design_and_Implementation_of_Fire_and_Gas_Detection_System_using_Arduino_and_GSM_Module

11. M. H. A. Mokhtar, et al. Design and Implementation of a Fire Alarm System using Arduino Microcontroller.

    https://www.researchgate.net/publication/341688056_Design_and_Implementation_of_a_Fire_Alarm_System_using_Arduino_Microcontroller

12. A. M. El-Fouly, et al. An Intelligent Fire Detection and Alarm System using IoT and Arduino Microcontroller.

    https://www.researchgate.net/publication/331416081_An_Intelligent_Fire_Detection_and_Alarm_System_using_IoT_and_Arduino_Microcontroller

13. S. M. Ahmed, et al. Fire Detection and Alarm System using Arduino.

    https://www.researchgate.net/publication/322054424_Fire_Detection_and_Alarm_System_Using_Arduino

14. D. M. Rodrigues, et al. Smart Fire Detection and Alarm System using IoT and Arduino Microcontroller.

https://www.researchgate.net/publication/328144156_Smart_Fire_Detection_and_Alarm_System_using_IoT_and_Arduino_Microcontroller

~~~~~~~~~~
~~~~~~~~~~