BANGLADESH UNIVERSITY OF ENGINEERING AND TECHNOLOGY
DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING

EEE 318 (January 2024)
Control System I Laboratory

Final Project Report

Section: C1 Group: 02

Autonomous Fire Extinguishing Robot

Course Instructors:

1. **Dr. Pran Kanai Saha**
   Professor, Dept of EEE, BUET
2. **Md. Jawad Ul Islam**
   Lecturer, Dept of EEE, BUET

Signature of Instructor: ____

Academic Honesty Statement:

IMPORTANT! Please carefully read and sign the Academic Honesty Statement, below. Type the student ID and name, and put your signature. *You will not receive credit for this project experiment unless this statement is signed in the presence of your lab instructor.*

"In signing this statement, We hereby certify that the work on this project is our own and that we have not copied the work of any other students (past or present), and cited all relevant sources while completing this project. We understand that if we fail to honor this agreement, We will each receive a score of ZERO for this project and be subject to failure of this course."

Signature: ____________________ **Full Name: S. M. Razibur Rahman** **Student ID: 2006134**	**Signature:** ____________________ **Full Name: Samia Hossain** **Student ID: 2006138**
Signature: ____________________ **Full Name: Bashir Shahriyer Rafi** **Student ID: 2006136**	**Signature:** ____________________ **Full Name: Md. Amir Hamza** **Student ID: 2006145**
Signature: ____________________ **Full Name: Md. Jaied Hasan** **Student ID: 2006146**	

Table of Contents

1 Abstract 1
2 Introduction 1
3 Design 2
3.1 Problem Formulation 2
3.1.1 Identification of Scope 2
3.1.2 Literature Review 2
3.1.3 Formulation of Problem 2
3.1.4 Analysis 2
3.2 Design Method (PO(a)) 3
3.2.1 System Overview 3
3.2.2 Fire Detection Mechanism 4
3.2.3 Wireless Communication 4
3.2.4 Control System 4
3.2.5 Fire Extinguishing Mechanism 5
3.2.6 Movement and Navigation 5
4 Implementation 5
4.1 Hardware Setup 6
4.2 Wireless Communication Between ESP32 and ESP8266 7
4.3 Client Side (ESP8266 Control) 8
4.4 Robot Navigation (Arduino Control) 10
4.5 Fire Extinguishing Mechanism (Servo and Water Pump) 12
5 Design Analysis and Evaluation 14
5.1 Novelty 14
5.2 Design Considerations (PO(c)) 14
5.2.1 Considerations to public health and safety 14
5.2.2 Considerations to environment 14
5.2.3 Considerations to cultural and societal needs 15
5.3 Investigations (PO(d)) 15
5.3.1 Design of Experiment 15
5.3.2 Data Collection 15
5.3.3 Results and Analysis 16
5.3.4 Interpretation and Conclusions on Data 17

5.4 Limitations of Tools (PO(e)) 18
5.5 Impact Assessment (PO(f)) 18
5.6 Sustainability Evaluation (PO(g)) 18
5.7 Ethical Issues (PO(h)) 19
6 Reflection on Individual and Team work (PO(i)) 20
6.1 Individual Contribution of Each Member 20
6.2 Mode of teamwork 20
7 Project Management and Cost Analysis (PO(k)) 20
8 Component 20
9 Future Work (PO(l)) 21
10 References 22

1 Abstract

This project presents the design and implementation of a fire extinguisher robot that autonomously detects and extinguishes fires in a two-room environment. The system integrates flame sensors, ESP32 and ESP8266 Wi-Fi modules, and an Arduino-based control system for efficient fire detection, navigation, and suppression. Flame sensors in each room continuously monitor fires and communicate fire events wirelessly to the robot. Upon receiving a fire signal, the robot navigates autonomously to the affected room using pre-programmed directions, confirms the presence of fire with its onboard sensors, and activates a water pump mechanism to extinguish the flames. The system employs a highly coordinated hardware setup, including motors for movement, servos for precise nozzle positioning, and a water pump for fire suppression. Wireless communication between the ESP modules ensures real-time, reliable signal transmission. This autonomous solution minimizes human intervention in fire emergencies, enhancing safety and response efficiency. The modular design allows for adaptability in various controlled environments, demonstrating the potential for scalable fire management applications.

2 Introduction

Fire outbreaks are one of the most devastating emergencies that can lead to significant loss of lives, property, and resources. In recent years, Bangladesh, particularly its densely populated capital, Dhaka, has witnessed frequent fire incidents in industrial plants, residential areas, and commercial establishments. The problem is exacerbated by the city's congested layout, which often makes it challenging for fire service personnel to reach the affected areas promptly. According to the latest WHO data published in 2020, fire-related deaths in Bangladesh reached 1,271, accounting for 0.18% of total deaths. These statistics underscore the pressing need for efficient and reliable fire mitigation solutions to address this grave threat to everyday life.

Addressing fire emergencies is complex, and various factors such as accessibility constraints, the unpredictable nature of fire spread, and the need for rapid response contribute to this. These challenges classify the issue as a Complex Engineering Problem (CEP) requiring innovative, multidisciplinary solutions. Conventional fire extinguishing methods rely heavily on human intervention, which can be delayed or hindered due to physical or logistical barriers. Alternative solutions, such as automated fire suppression systems, have been explored but often lack mobility and adaptability to dynamic fire scenarios.

In this context, the "Automatic Fire Extinguishing Robot" offers a promising alternative by integrating robotics, automation, and advanced sensing technologies. The proposed robot is designed to detect flames, navigate autonomously or via a preplanned route using Bluetooth communication, and extinguish the fire using a built-in water pump. This system aims to minimize response time and reduce potential damage by eliminating the dependency on human intervention in high-risk situations.

Implementing such a robot is particularly relevant in Dhaka, where fire emergencies demand swift action despite infrastructural and urban constraints. By leveraging cutting-edge engineering principles, the Automatic Fire Extinguishing Robot has the potential to revolutionize fire safety measures, significantly reducing the risk to human lives and property in both urban and rural settings

3 Design

3.1 Problem Formulation

3.1.1 Identification of Scope

Fire hazards in confined environments, such as residential or commercial buildings, pose significant threats to safety and often require quick intervention. Traditional fire management systems rely on alarms and human involvement, which can delay response times and increase the risks to individuals in hazardous areas. The scope of this project is to design and implement an autonomous fire extinguisher robot capable of detecting, navigating to, and suppressing fires in a two-room environment. The robot integrates real-time fire detection, wireless communication, and automated extinguishing mechanisms to minimize human intervention and enhance safety. It is particularly beneficial for areas that are difficult to access during emergencies, such as chemical laboratories or warehouses.

3.1.2 Literature Review

Existing studies in fire safety automation emphasize the importance of integrating robotics and sensor-based systems for efficient fire management. Several robotic systems have been developed to detect fires using flame or heat sensors; however, many lack reliable navigation or efficient extinguishing mechanisms. Wireless communication, such as the use of Wi-Fi modules, has shown promise in real-time signaling between fire detection systems and autonomous robots. Research also highlights the need for energy-efficient systems that remain dormant during non-critical periods. Building upon these insights, this project addresses gaps in existing designs by incorporating robust wireless communication, precise navigation, and an energy-efficient extinguishing system.

3.1.3 Formulation of Problem

The core problem addressed in this project is the development of a reliable and autonomous robotic system to detect and extinguish fires in confined spaces. Specifically, the robot must overcome challenges in accurate fire detection, real-time communication, and precise navigation in a two-room environment. The system must also ensure effective fire suppression through a water pump mechanism while conserving energy during idle periods. Additionally, the communication between sensors and the robot should be robust, ensuring timely and accurate signaling.

3.1.4 Analysis

The problem is analyzed by breaking it into key functional components: fire detection, wireless communication, navigation, and extinguishing mechanisms. The analysis includes studying the operational thresholds of flame sensors for accurate detection, evaluating the reliability of Wi-Fi-based communication (ESP32 and ESP8266 modules), and designing pre-programmed navigation paths for efficient movement between rooms. Energy consumption patterns are also analyzed to optimize the robot's power usage, ensuring longer operational periods. The robot's extinguishing mechanism is tested for directional precision and water flow efficiency to achieve effective fire suppression.

3.2 Design Method (PO(a))

The design of our fire extinguisher robot aims to provide an autonomous solution for fire detection and suppression in a two-room environment. The system integrates flame sensors, wireless communication modules, and an Arduino-based control system, enabling the robot to detect fire events, navigate to the affected room, and extinguish the fire without human intervention.

3.2.1 System Overview

The system comprises three major components: the fire detection system, the wireless communication system, and the robot itself. The fire detection system consists of flame sensors placed in each room, which monitor the presence of fire. Upon detection, the flame sensor sends a signal to the robot via wireless communication. The robot is then guided to the fire's location, where it uses its flame sensor to confirm the presence of fire and activates the extinguishing mechanism.

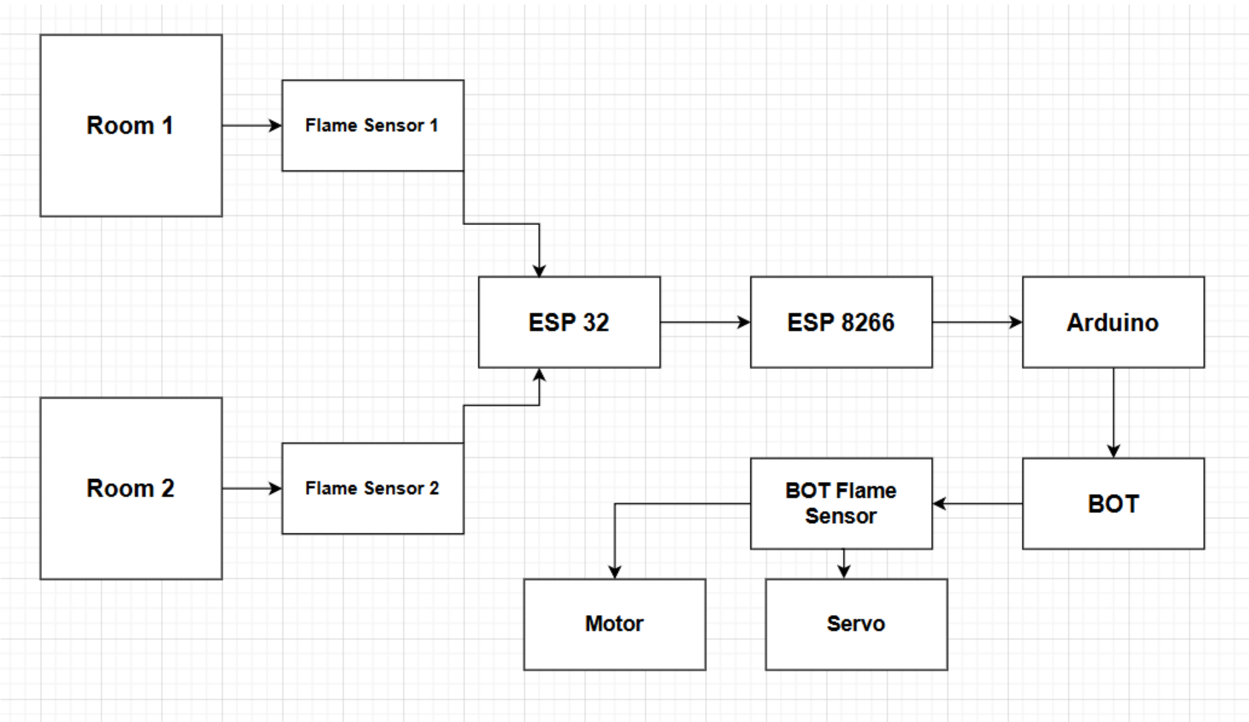

Figure 1 System design overview

3.2.2 Fire Detection Mechanism

Each room is equipped with a flame sensor that continuously monitors the presence of fire. The flame sensor detects the infrared radiation emitted by flames and generates an output signal when a fire is detected. These sensors are connected to an ESP32 microcontroller, which functions as the server in the wireless communication network. The ESP32 monitors the sensors for any fire-related signals.

3.2.3 Wireless Communication

The wireless communication between the flame sensors and the robot is established using two ESP modules: the ESP32 and the ESP8266. The ESP32 acts as the server, while the ESP8266 functions as the client. These ESP modules communicate via a Wi-Fi network, which is set up using a mobile phone as the router. This mobile network simply facilitates communication between the ESP modules, with no involvement in the control of the robot itself.

Communication between the ESP modules follows these steps:

- If no fire is detected in either room, the flame sensor sends a signal labeled 's' (safe) to the ESP8266.
- If fire is detected in room 1, the sensor sends a signal '1' to the ESP8266.
- If fire is detected in room 2, the sensor sends a signal '2' to the ESP8266.

The ESP8266 then forwards the received signal to the Arduino, which decides the robot's course of action based on the signal received.

3.2.4 Control System

The heart of the robot's operation lies in the Arduino, which receives the signals from the ESP8266 and directs the robot's movements accordingly. The Arduino processes the received signal and triggers appropriate actions based on whether the signal indicates safety or a fire event in room 1 or room 2.

- If the signal indicates safety ('s'), the robot remains stationary, doing nothing.
- If the signal indicates a fire in room 1 ('1') or room 2 ('2'), the Arduino uses pre-programmed directions to guide the robot to the corresponding room.

The robot's navigation is managed through motors and servo mechanisms, which control its movement and direction.

3.2.5 Fire Extinguishing Mechanism

Upon arriving at the designated room, the robot uses its own flame sensor to verify the presence of fire. The robot performs a 360-degree rotation to scan the environment for flames. If fire is detected, the robot halts its movement, and the extinguishing mechanism is activated.

- The robot is equipped with a water pump connected to a servo motor, which directs the water flow toward the fire. When fire is detected, the servo adjusts the nozzle's position, and the water pump is activated to spray water onto the fire.
- If no fire is detected after the scan, the robot stops its motor and servo to conserve energy.

3.2.6 Movement and Navigation

The robot's movement is powered by motors that drive the robot forward, and backward, and enable turns. The servo is used to position the water spray nozzle during the extinguishing process. The directions for the robot's navigation are pre-programmed based on the room layout, ensuring the robot reaches the appropriate room efficiently.

4 Implementation

The Fire Extinguisher Robot is a highly automated system designed to detect and extinguish fires in a controlled environment. It uses wireless communication between two ESP modules (ESP32 as the server and ESP8266 as the client), flame sensors to detect fire, and an Arduino-based control system to navigate and activate the fire extinguishing mechanism.

The system is divided into three main modules: Flame Detection, Robot Navigation, and Fire Extinguishing Mechanism. Each part works together to ensure the robot can detect fires, navigate to the appropriate room, and extinguish the fire.

The following table represents the workflow of the code.

Step	Description	Function/Action
1. Setup Phase	Initialize serial communication, set pin modes	Serial.begin(960e) - pinmode() for motor and flame sensor pins
2. Main Loop	Check if fire is detected in the rooms	analogRead() for room1 and room2, check conditions for fire

2a. Fire Detected in Room 1	If fire is detected in Room 1 , update variables and call gotoRoom1()	detectedRoom - 1, fireDetected - true, call gotoRoom1 ()
2b. Fire Detected in Room 2	If fire is detected in Room 2. update variables and call gotoRoom2()	detectedRoom - 2, fireDetected - true, call gotoRoom2 ()
3. Goto Room 1	Move forward and rotate 360 degrees	Call forward() 3 times, then rotate36e()
4. Goto Room 2	Rotate, move forward, and rotate 360 degrees	Call rotateleft(450), forward() 2 times, rotateright(45e), forward() 3 times, then rotate36e()
5. Move Forward	Move motors forward for 1 second, then stop motors	Set notor1Fonward - HIGH , motor2Forward HIGH, then stopmotors()
6. Stop Motors	Stop all motors	Set notor1Forward, notor18achward, motor2Forward, motor2Bacloward to LOW
7. Rotate 360 Degrees	Rotate the motors in 90 -degree steps until fire is detected	Call rotateright(9e) in a loop, check checkForfire() until firefound - true
8. Rotate Right	Rotate right for specified milliseconds	Set notor1forward - HIGH, motor2Bacloward HIGH, stop motors after the specified time
9. Rotate Left	Rotate left for specified milliseconds	Set notor1Backward - HIGH, rotor2Forward HIGH , stop motors after the specified time
10. Check for Fire	Check if flame sensor detects fire	If flamesensor1 is LOW, set firefound - true

4.1 Hardware Setup

The hardware components of the robot include:

- **Flame Sensors**: Two flame sensors (connected to the ESP32) are placed in Room 1 and Room 2. These sensors detect the presence of fire by measuring infrared radiation.

- **Motors**: The robot uses motors for movement. It can move forward, backward, and rotate.
- **Water Pump and Servo Motor**: These are used to extinguish fire by spraying water. The servo controls the direction of the nozzle, while the water pump activates to spray water.
- **Wi-Fi Modules (ESP32 and ESP8266)**: The ESP32 communicates with the flame sensors and sends fire alerts to the ESP8266. The ESP8266 receives these signals and controls the robot's movement.
- **Arduino Microcontroller**: It handles the overall control logic, interpreting the fire detection signals and controlling the motors and fire extinguishing mechanisms.

4.2 Wireless Communication Between ESP32 and ESP8266

The ESP32 (server) communicates wirelessly with the ESP8266 (client) via UDP. The ESP32 is responsible for detecting fire and sending the fire signal to the ESP8266. The ESP8266 then triggers the robot's actions, guiding it to the appropriate room and activating the fire extinguishing mechanism.

ESP32 (Server) Code: The ESP32 continuously checks the flame sensors connected to Room 1 and Room 2 using analog readings. If the sensor detects a fire (when the sensor's value exceeds a threshold), the ESP32 sends the corresponding signal (`'R1'`, `'R2'`, or `'S'`) to the ESP8266.

```
#include <WiFi.h>
#include <WiFiUdp.h>

const char* ssid = "OPPO A15";
const char* password = "jahid002";

const char * udpAddress = "192.168.43.23";
WiFiUDP udp;
unsigned int udpPort=1234;
int sensor1 = 13;
int sensor2 = 12;

char pktbuf[10];
char rx_val;

void setup() {
  pinMode(sensor1, INPUT);
  pinMode(sensor2, INPUT);
  Serial.begin(115200);
  WiFi.begin(ssid, password);
  while (WiFi.waitForConnectResult() != WL_CONNECTED) {
    Serial.println("Connection Failed! Rebooting...");
    delay(5000);
    ESP.restart();
  }
  Serial.println(WiFi.localIP());
  Serial.println("Status: Connected");

  udp.begin(udpPort);
  Serial.println(udpPort);
```

```
}

void loop() {
  int rp = udp.parsePacket();
  if (!rp) {
    if (digitalRead(sensor1) == LOW) {
      rx_val = 'R1';  // Fire detected in Room 1
      Serial.print("udp_send: ");
      Serial.println(rx_val);
      udp.beginPacket(udpAddress, udpPort);
      udp.write(rx_val);
      udp.endPacket();
    }
    else if (digitalRead(sensor2) == LOW) {
      rx_val = 'R2';  // Fire detected in Room 2
      Serial.print("udp_send: ");
      Serial.println(rx_val);
      udp.beginPacket(udpAddress, udpPort);
      udp.write(rx_val);
      udp.endPacket();
    }
    else {
      rx_val = 'S';   // No fire detected (Safety)
      Serial.print("udp_send: ");
      Serial.println(rx_val);
      udp.beginPacket(udpAddress, udpPort);
      udp.write(rx_val);
      udp.endPacket();
    }
  } else {
    udp.read(pktbuf, 1);
    Serial.print("Packet from "+String(udpAddress)+": ");
    Serial.println(pktbuf);
    delay(1000);
  }
  delay(1000);
}
```

This code continuously monitors the flame sensors, sending the appropriate signals to the ESP8266 based on whether a fire is detected in Room 1, Room 2, or no fire is detected.

4.3 Client Side (ESP8266 Control)

The **ESP8266** receives the fire detection signals (`'R1'` for Room 1, `'R2'` for Room 2) from the **ESP32** and controls the corresponding LEDs to indicate which room has the fire.

```
cpp
Copy code
#include <ESP8266WiFi.h>
#include <WiFiUdp.h>
#include<Servo.h>
Servo mys;
const char* ssid = "OPPO A15";
const char* password = "jahid002";

const char * udpAddress = "192.168.43.250";
WiFiUDP udp;
unsigned int udpPort = 1234;
int ledR1 = D2;
```

```
int ledR2 = D4;

char pktbuf1[10];
char rx_val;

void setup() {
  mys.attach(D5);
  mys.write(90);
  pinMode(ledR1, OUTPUT);
  pinMode(ledR2, OUTPUT);
  pinMode(D6, INPUT);
  Serial.begin(9600);
  WiFi.begin(ssid, password);

  while (WiFi.waitForConnectResult() != WL_CONNECTED) {
    Serial.println("Connection Failed! Rebooting...");
    delay(5000);
    ESP.restart();
  }

  Serial.println(WiFi.localIP());
  Serial.println("Status: Connected");

  udp.begin(udpPort);
  Serial.println("UDP Port: " + String(udpPort));
}

void loop() {
  int packetSize = udp.parsePacket();

  if (!packetSize) {
    if (Serial.available() > 0) {
      rx_val = Serial.read();
      Serial.print("udp_send: ");
      Serial.println(rx_val);
      udp.beginPacket(udpAddress, udpPort);
      udp.write(rx_val);
      udp.endPacket();
    }
  } else {
    udp.read(pktbuf1, sizeof(pktbuf1) - 1);
    pktbuf1[packetSize] = '\0';

    Serial.print("Packet from " + String(udpAddress) + ": ");
    Serial.println(pktbuf1);

    if (pktbuf1[0] == '1') {
      digitalWrite(ledR1, HIGH);
      digitalWrite(ledR2, LOW);
    }
    else if(pktbuf1[0] == '2') {
      digitalWrite(ledR1, LOW);
      digitalWrite(ledR2, HIGH);
    }
    else {
      digitalWrite(ledR1, LOW);
      digitalWrite(ledR2, LOW);
    }

  }

  if(digitalRead(D6) == LOW){
```

```
    for (int pos = 0; pos <= 180; pos += 5) {
      mys.write(pos);
      delay(30);
    }
    for (int pos = 180; pos >= 0; pos -= 5) {
      mys.write(pos);
      delay(30);
    }
  }
  delay(100);
}
```

This code checks for incoming fire detection signals from the ESP32 and controls the LEDs and servo based on the received packet.

4.4 Robot Navigation (Arduino Control)

The **Arduino** receives fire detection signals from the ESP8266 and determines the robot's movement. Depending on the signal (`'R1'` or `'R2'`), it directs the robot to the correct room using motor controls. The robot's movement includes moving forward, rotating, and stopping at specific locations to perform fire detection with its own flame sensor.

Here's how the robot moves to the respective room based on the fire signal:

```
cpp
Copy code
const int motor1Forward = 2;
const int motor1Backward = 3;
const int motor2Forward = 4;
const int motor2Backward = 5;
const int flameSensor1 = 12;  // Room 1 flame sensor
int detectedRoom = 0;
bool fireDetected = false;

void setup() {
  Serial.begin(9600);
  pinMode(motor1Forward, OUTPUT);
  pinMode(motor1Backward, OUTPUT);
  pinMode(motor2Forward, OUTPUT);
  pinMode(motor2Backward, OUTPUT);
  pinMode(flameSensor1, INPUT);
}

void loop() {
  if (!fireDetected) {
    int fire1 = analogRead(A0);  // Read Room 1 sensor
    int fire2 = analogRead(A1);  // Read Room 2 sensor
    if (fire1 > 650) {
      Serial.println("Fire detected in Room 1!");
      detectedRoom = 1;
      fireDetected = true;
      gotoRoom1();
    } else if (fire2 > 650) {
      Serial.println("Fire detected in Room 2!");
      detectedRoom = 2;
      fireDetected = true;
      gotoRoom2();
    }
```

```
    delay(500);
  }
}

void gotoRoom1() {
  forward();
  rotate360();
}

void gotoRoom2() {
  rotateLeft(450);
  forward();
  rotateRight(450);
  forward();
  rotate360();
}

void forward() {
  digitalWrite(motor1Forward, HIGH);
  digitalWrite(motor1Backward, LOW);
  digitalWrite(motor2Forward, HIGH);
  digitalWrite(motor2Backward, LOW);
  delay(1000);
  stopMotors();
  delay(500);
}

void stopMotors() {
  digitalWrite(motor1Forward, LOW);
  digitalWrite(motor1Backward, LOW);
  digitalWrite(motor2Forward, LOW);
  digitalWrite(motor2Backward, LOW);
}

void rotate360() {
  while (fireDetected == false) {
    rotateRight(90);
    checkForFire();
    delay(300);
  }
  if (fireDetected) {
    stopMotors();
    Serial.println("Fire found");
  }
}

void rotateRight(int time) {
  digitalWrite(motor1Forward, HIGH);
  digitalWrite(motor1Backward, LOW);
  digitalWrite(motor2Forward, LOW);
  digitalWrite(motor2Backward, HIGH);
  delay(time);
  stopMotors();
}

void rotateLeft(int time) {
  digitalWrite(motor1Forward, LOW);
  digitalWrite(motor1Backward, HIGH);
  digitalWrite(motor2Forward, HIGH);
  digitalWrite(motor2Backward, LOW);
  delay(time);
  stopMotors();
```

```
}

void checkForFire() {
  if (digitalRead(flameSensor1) == LOW) {
    fireDetected = true;
  }
}
```

In this code:

- **`gotoRoom1`** and **`gotoRoom2`** functions direct the robot to the respective rooms. The robot moves forward and rotates 360° to detect the fire using its own flame sensor.
- The robot stops and checks for fire when it reaches the designated room.

4.5 Fire Extinguishing Mechanism (Servo and Water Pump)

Once the robot detects fire, the **Servo** motor is used to position the water nozzle, and the **water pump** is activated to spray water. The servo rotates back and forth to ensure the fire is extinguished from all directions.

The servo movement is programmed as follows:

```
cpp
Copy code
#include <Servo.h>
Servo mys;
int ledR1 = D2;
int ledR2 = D4;

void setup() {
  mys.attach(D5);
  mys.write(90);  // Position the servo at 90 degrees (neutral)
  pinMode(ledR1, OUTPUT);
  pinMode(ledR2, OUTPUT);
}

void loop() {
  if (digitalRead(D6) == LOW) {  // If the flame sensor detects fire
    for (int pos = 0; pos <= 180; pos += 5) {
      mys.write(pos);  // Move the servo to position the water nozzle
      delay(30);
    }
    for (int pos = 180; pos >= 0; pos -= 5) {
      mys.write(pos);  // Return the servo to its original position
      delay(30);
    }
    // Turn on the water pump to extinguish fire
    // Code to activate the pump can be added here.
  }
  delay(100);  // Delay between checks
}
```

This code moves the servo from 0 to 180 degrees and back to cover the fire with water. The hardware implementation parts images are attached in the following figure.

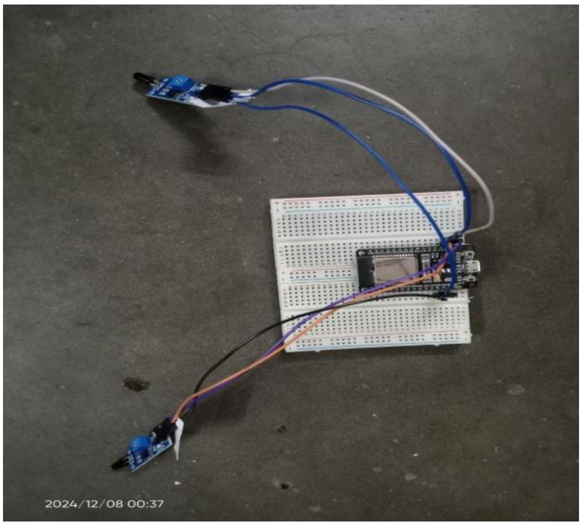

Figure 2a. Room Flame Sensor and ESP 32

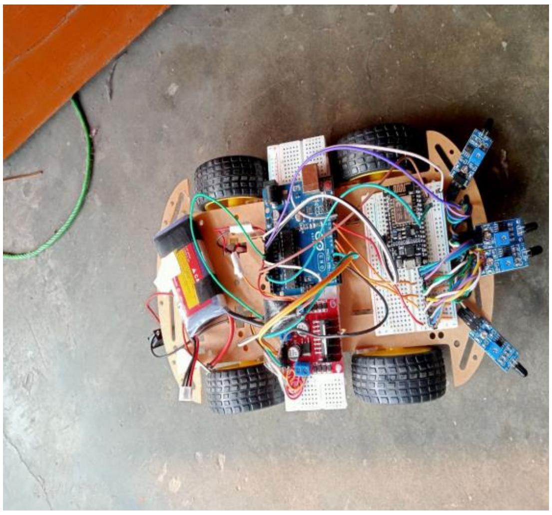

Figure 2b. ESP 8266 and Arduino

Figure 2c. Fire extinguishing using pump and servo

5 Design Analysis and Evaluation

5.1Novelty

The "Automatic Fire Extinguishing Robot" stands out for its innovative features, including:

- **Adaptability in Residential Settings**: The robot's compact design and precise navigation capabilities make it ideal for maneuvering within confined spaces in households.
- **Wifi-Based Preplanned Navigation**: Integration of wireless communication ensures efficient and accurate movement toward fire locations within a home, reducing delays and reliance on external controls.
- **Fire Detection Mechanism**: Equipped with flame sensor, the robot ensures reliable detection of visible flames, enhancing responsiveness.
- **Cost-Effectiveness and Portability**: By utilizing readily available components such as Arduino boards, mini water pumps, and simple chassis designs, the system remains affordable and easy to deploy in various household scenarios.
- **Autonomous Operation**: The robot is capable of independently detecting fires, extinguishing them, and returning to its base, minimizing the need for human intervention in hazardous conditions.
- **Scalability for Future Enhancements**: The design allows for integration of additional features, such as obstacle avoidance, AI-based decision-making, and higher-capacity fire suppression systems, ensuring long-term usability and adaptability.

These features collectively make the robot a cutting-edge solution to modern fire safety challenges, particularly in residential environments.

5.2 Design Considerations (PO(c))

The design of the "Automatic Fire Extinguishing Robot" aligns with Program Outcome (PO(c)) by addressing the need to develop solutions for complex engineering problems with a focus on public health, safety, and societal impact. Key considerations include:

5.2.1 Considerations to public health and safety

The robot provides a reliable and rapid response to fire emergencies in households, reducing risks to human life and minimizing potential injuries during hazardous situations.

5.2.2 Considerations to environment

The system is designed to minimize resource usage by using efficient components like mini water pumps and sensors, reducing water wastage and energy consumption during fire suppression. The use of affordable and accessible materials promotes widespread adoption, while the modular design supports future upgrades to incorporate advanced technologies for enhanced performance.

5.2.3 Considerations to cultural and societal needs

In densely populated regions like Dhaka, where accessibility is a major concern, the robot’s compact design ensures effective functionality within confined residential spaces.

5.3 Investigations (PO(d))

5.3.1 Design of Experiment

The experiment was designed to test the effectiveness of the "Automatic Fire Extinguishing Robot" in identifying and extinguishing fires in a multi-room household environment. The workflow involves two rooms, each equipped with a flame sensor connected to an ESP32 microcontroller. The ESP32 sends data to an ESP8266 module, which communicates with an Arduino for decision-making. Scenarios included simulating flames in each room and testing the robot's response in terms of navigation, fire detection, and extinguishing.

5.3.2 Data Collection

Data was gathered through multiple trials involving different scenarios:
Signal Communication: Monitoring the communication between flame sensors, ESP32, ESP8266, and Arduino to ensure accurate data relay.

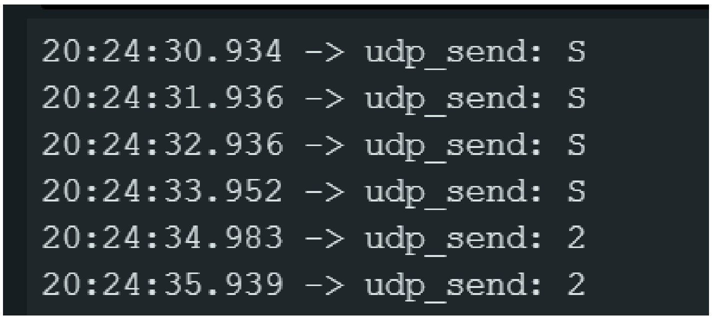

Figure 3a. ESP32 sending data to ESP8266 through UDP connection

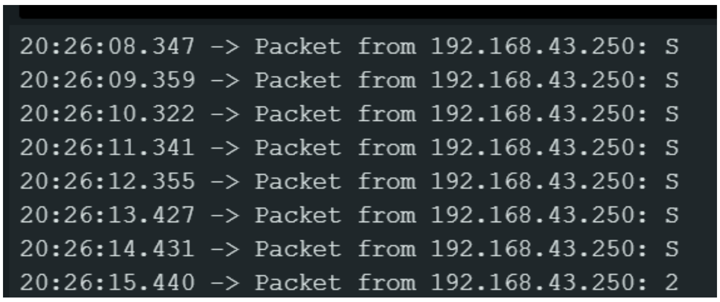

Figure 3b. ESP8266 sending data to Arduino

Room Navigation: Observing the robot's ability to identify the correct room based on sensor

input.
360-Degree Fire Detection: Assessing the robot's rotation to locate the fire within the room.

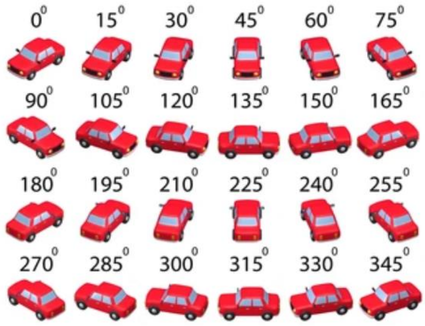

Figure 4 Rotation to identify fire

Extinguishing Mechanism: Evaluating the activation of the water pump and servo motor when the fire source is detected.

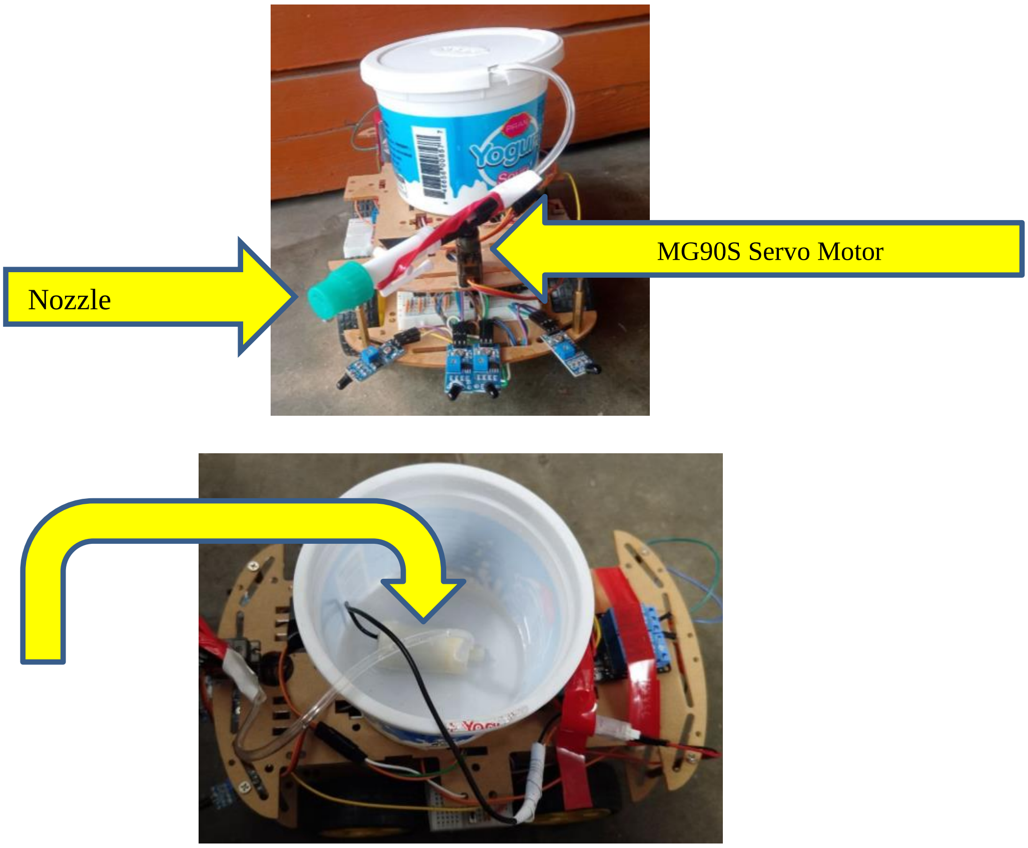

Figure 5 Submersible mini DC water pump

System Reset: Ensuring the robot returns to a standby state once the fire is extinguished.

5.3.3 Results and Analysis

Signal Communication: The system demonstrated reliable data transfer between components, with the ESP8266 effectively relaying data to the Arduino for decision-making.

Room Navigation: The robot consistently navigated to the correct room based on sensor input, confirming the system's reliability in interpreting flame sensor signals.

Figure 6 Flame sensor is activated in a room

Fire Detection: The 360-degree rotation allowed the robot to accurately locate the fire source within the room, demonstrating its adaptability to varying fire locations.
Extinguishing Mechanism: The water pump and servo motor activated promptly upon detecting fire, extinguishing the flame effectively without excessive resource use.

Figure 7 Water is being sprayed by pump

System Reset: Post-extinguishing, the robot successfully returned to its initial state, ready for the next task.

5.3.4 Interpretation and Conclusions on Data

The experimental results confirm the "Automatic Fire Extinguishing Robot" as a robust and effective solution for household fire emergencies. The system's communication, navigation, and fire-extinguishing capabilities ensure reliable performance in real-world scenarios. Minor improvements, such as enhancing the precision of 360-degree rotation and optimizing the flame sensor's sensitivity, could further refine its efficiency. Overall, the robot meets its objectives of rapid fire detection and suppression, ensuring enhanced safety in residential settings.

5.4 Limitations of Tools (PO(e))

The tools used in this project, while effective, have certain limitations:

1. **Flame Sensors**: The sensors have limited detection range and may fail to detect small or distant flames. They also show reduced sensitivity in high ambient light conditions.
2. **ESP32 and ESP8266 Modules**: Both modules rely on a stable Wi-Fi network. Signal interference or weak connections can disrupt data communication.
3. **Arduino Microcontroller**: Processing multiple sensor inputs simultaneously can cause delays due to its limited computational power.
4. **Water Pump and Servo Motor**: The water pump has limited pressure and flow rate, which may not be sufficient for larger fires. The servo motor's precision could also be impacted by heavy loads.
5. **Power Supply**: The robot's reliance on batteries limits its operational time, requiring frequent recharging or replacement.

These limitations highlight the need for careful consideration of the tools' constraints during design and implementation.

5.5 Impact Assessment (PO(f))

5.5.1 Assessment of Societal and Cultural Issues

The robot addresses societal challenges by providing a cost-effective solution for household fire safety, particularly in densely populated regions. Its accessibility ensures that even low-income households can benefit from enhanced safety measures. The design respects cultural norms by maintaining compact and unobtrusive functionality within homes.

5.5.2 Assessment of Health and Safety Issues

The robot significantly improves health and safety by reducing the risks associated with fire emergencies. Its autonomous operation minimizes human involvement in hazardous situations, protecting lives while ensuring rapid response to fires.

5.5.3 Assessment of Legal Issues

The project complies with relevant safety and electronic device regulations, ensuring safe operation within households. Future iterations could include certifications to meet international standards, enhancing its legal acceptability.

5.6 Sustainability Evaluation (PO(g))

The project promotes sustainability through:

1. **Material Use**: The use of affordable and widely available components minimizes waste and promotes accessibility.
2. **Energy Efficiency**: Efficient components such as low-power microcontrollers and water pumps reduce energy consumption.
3. **Longevity**: Modular design enables easy upgrades and maintenance, reducing the need for complete replacement.
4. **Environmental Impact**: By mitigating fire damage, the robot indirectly contributes to environmental preservation by reducing the release of harmful emissions during fires.

5.7 Ethical Issues (PO(h))

The project adhered to ethical principles throughout its development. Data privacy was respected by ensuring no sensitive information was transmitted during testing. Ethical considerations also guided the design, ensuring no harm to users or the environment. Challenges included sourcing reliable components without compromising on quality or affordability, which were mitigated through careful supplier selection. The team remained committed to professional responsibilities and norms of engineering practice, ensuring transparency and integrity in all aspects of the project.

6 Reflection on Individual and Team work (PO(i))

6.1 Individual Contribution of Each Member

2006134	**Wireless Communication client side using ESP 8266**
2006136	**Driving the Bots motors, going to rooms**
2006138	**Servo and Pump**
2006145	**Flame Sensor and Hardware**
2006146	**Wireless communication server end using ESP 32**

6.2 Mode of teamwork

Our project was completed through collaborative teamwork, where each member contributed their expertise to specific components of the system while providing mutual assistance and feedback to ensure the project's success. Individual responsibilities included tasks like wireless communication (client-side with ESP8266 and server-side with ESP32), motor control for navigation, servo and pump integration, and flame sensor hardware setup. Despite these defined roles, the team worked cohesively by sharing knowledge, troubleshooting challenges together, and integrating individual contributions into a seamless and functional system. This cooperative approach ensured a high-quality and efficient outcome, reflecting our shared commitment to the project.

7 Project Management and Cost Analysis (PO(k))

8 Component	Quantity	Price
Jumper （m2m）	2 set	120
Arduino uno	1	700

Flame sensor	3	210
switch	1	10
Water pump	1	50
breadboard	2	180
Motor driver	1	150
motor	4	300
wheel	4	260
relay	1	30
ESP	2	780
Li-po battery	1	1350
Charger	1	450
Servo motor	1	210
Chassis	1	540
Total		5340

9 Future Work (PO(l))

The "Automatic Fire Extinguishing Robot" project demonstrates a robust approach to addressing household fire emergencies, but further advancements can enhance its capabilities:

1. **Integration of Advanced Sensors**: Future iterations could include high-precision flame and gas sensors to improve detection reliability and range.
2. **AI-Powered Navigation**: Incorporating artificial intelligence for decision-making and path optimization would enhance the robot's adaptability to complex layouts.
3. **Image Processing for Fire Detection**: Utilizing image processing techniques with cameras to visually identify fire sources, improving accuracy and reducing false positives.
4. **Obstacle Avoidance**: Adding advanced obstacle detection and avoidance mechanisms to ensure smooth navigation in cluttered environments.
5. **Extended Connectivity**: Exploring IoT-based solutions for real-time monitoring and control via mobile applications or cloud platforms.
6. **Alternative Fire Suppression Methods**: Investigating the use of materials other than water, such as CO2 or dry chemical agents, for extinguishing fires safely and effectively in various environments.
7. **Tunable Water Direction**: Implementing precise control of the water jet direction to optimize fire suppression and minimize resource wastage.
8. **Improved Power Management**: Using more efficient power systems, such as solar panels or higher-capacity rechargeable batteries, to extend operational duration.
9. **Scalability for Larger Spaces**: Enhancing the robot's water tank capacity and fire suppression mechanisms for use in larger residential or commercial spaces.
10. **Community Integration**: Educating users on the robot's operation and incorporating feedback loops for continuous improvement.

These advancements emphasize the importance of life-long learning and keeping pace with contemporary developments in electrical and electronic engineering to maintain the robot's relevance and effectiveness.

10 References

D. Eridani, A. F. Rochim and F. N. Cesara, "Comparative Performance Study of ESP-NOW, Wi-Fi, Bluetooth Protocols based on Range, Transmission Speed, Latency, Energy Usage and Barrier Resistance," *2021 International Seminar on Application for Technology of Information and Communication (iSemantic),* Semarangin, Indonesia, 2021, pp. 322-328, doi: 10.1109/iSemantic52711.2021.9573246.

T. N. Hoang, S. -T. Van and B. D. Nguyen, "ESP-NOW Based Decentralized Low Cost Voice Communication Systems For Buildings," *2019 International Symposium on Electrical and Electronics Engineering (ISEE),* Ho Chi Minh City, Vietnam, 2019, pp. 108-112, doi: 10.1109/ISEE2.2019.8921062