ACKNOWLEDGEMENTS

It is my privilege to have been associated with **Prof.S.R.Jain**, Head of Department of Computer engineering, My guide **Prof.R.T.Kadam** during this project stage I work. We are thankful to her constants Inspiration and valuable guidance, Carefully reading and editing our work and always boosting my confidence to complete our work . We express our gratitude to **Prof.R.T.Kadam** and **Prof.S.R.Jain** for their constant encouragement , co-operation ,valuable guidance and support. We would like to all the staff members of the department of there continues support. We grateful thanks to our principle **Prof.V.S.Zadkhande**.

This kind of work cannot be finished without May others help, even some of them have not aware of their contribution and importance in developing this application. It is great pleasure for me to take this opportunity to express my gratefulness to all them.

Mr. Shubham Vaman Ahire.

Mr. Aditya Rajendra Nilange.

Mr. Yash Harshchandra Jagdhane.

Mr. Prasad Narendra Nikam.

ABSTRACT

Fire hazards remain a major concern in rural and industrial areas, where limited infrastructure often delays fire brigade response. This project introduces a semi-automatic fire-fighting robot tailored for ground-level fires in such environments. Designed to be mounted on small platforms like farm trolleys or industrial carts, the robot is manually driven to the site and then operates autonomously to detect and suppress fire.

Equipped with three flame sensors, it scans for fire in front and on both sides, using logic to prioritize and target the central flame. A servo motor-controlled nozzle directs water from a micro submersible pump, enabling precise and efficient fire suppression. The system is powered via a DC adapter for the Arduino Uno and a separate relay power source for motors and pump.

Once the fire is extinguished, the robot resumes its movement, continuing to monitor for re-ignition. The robot's compact design, low cost, and automatic response make it suitable for petrol pumps, godowns, solar farms, and narrow industrial areas where traditional fire engines struggle.

By integrating embedded systems, sensors, and automation, this robot enhances safety and reduces human risk during fire emergencies. Its scalable and adaptable structure promotes wider adoption in remote and high-risk zones, contributing to faster, smarter, and safer fire-fighting solutions.

KEYWORDS : Fire-fighting robot, Arduino Uno, flame sensor, servo motor, water pump, L298N motor driver, TIP122 transistor, semi-automatic fire system, embedded system, submersible pump, rural firefighting, low-cost fire extinguisher, trolley-based fire robot, Indian fire safety, smart automation, autonomous fire response, industrial safety, DC motor robot, relay module, flame detection.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

Sr. No.	Abbreviation	Full Form
01	IR	Infrared
02	DC	Direct Current
03	AC	Alternating Current
04	UNO	Universal Number One (Arduino Uno Board)
05	PWM	Pulse Width Modulation
06	TIP122	NPN Darlington Transistor (TIP122 Model)
07	IDE	Integrated Development Environment
08	DFD	Data Flow Diagram
09	UML	Unified Modeling Language
10	LED	Light Emitting Diode
11	ENA / ENB	Enable Pins on L298N Motor Driver
12	GPS	Global Positioning System
13	GSM	Global System for Mobile Communications
14	AI	Artificial Intelligence
15	KNN	K-Nearest Neighbor
16	SVM	Support Vector Machine

Table 1: Abbreviations

NOTATION

Sr. No.	Notation	Meaning
01	FIRE	Fire detected signal
02	STOP	Robot stops (motor halt)
03	WATER ON	Water pump activated
04	FORWARD	Robot moving forward
05	$\rightarrow$	Directional flow or connection (used in block/DFD diagrams)
06	LOW	Logic level from fire sensor indicating fire presence (active LOW)
07	HIGH	Logic level used to activate components (e.g., pump, motor)
08	A0–A2	Analog pins used to read flame sensors on Arduino Uno
09	D2–D13	Digital pins used to control motor drivers, pump, and servo
10	90°	Centered servo angle (default nozzle position)
11	28°	Right-aligned servo angle (for fire on right)
12	150°	Left-aligned servo angle (for fire on left)
13	`bool`	Boolean data type representing TRUE or FALSE values
14	`delay(ms)`	Pauses execution for specified milliseconds (Arduino code)

Table 2: Notations

CHAPTER 1

INTRODUCTION

1.1 Introduction

Fire is one of the most dangerous and destructive forces in our environment. It can cause massive damage to life, property, and the environment within minutes. Fire hazards are especially threatening in rural and industrial areas of India, where narrow roads, poor infrastructure, and long distances often prevent timely access by traditional fire brigades.

To tackle these challenges, our project presents a semi-automatic Arduino-based fire-fighting robot, designed specifically for ground-level fire response in areas where conventional firefighting methods fall short. This robotic system is designed to be mounted on any standard vehicle platform such as a farm trolley, industrial cart, or temporary carrier. The robot is semi-automatic—it requires a human only to drive the vehicle close to the fire zone, and from that point, the system takes over. It then automatically detects and extinguishes the fire, reducing human risk and enhancing safety.

The robot uses three flame sensors placed at the left, center, and right sides to detect the direction of the fire. Once fire is detected, an Arduino Uno microcontroller sends signals to a servo motor, which rotates a water nozzle connected to a Synronic submersible micro water pump. Water is then sprayed toward the fire. If all three sensors detect fire simultaneously, the robot identifies it as a major fire, and gives priority to extinguishing that area.

The system also includes:

• An L298N motor driver to control the movement of four DC motors (one for each wheel, connected in parallel).
• A TIP122 transistor to switch heavy current devices such as the pump.
• A relay module powered by four standard cells for powering the pump and servo motor.

- A AC to DC adapter to power the Arduino Uno.

Although the circuit diagram shows two motors for simplicity, the actual system uses four motors to ensure strong and stable motion. The robot does not use ultrasonic sensors, smartphone connectivity, or remote control, keeping the design cost-effective and simple for Indian conditions.

This project showcases the power of embedded systems, robotics, and automation, offering a life-saving, real-world application that not only solves critical fire hazards but also contributes to public and industrial safety at an affordable cost.

1.2 Need of project

In many parts of India—especially rural, remote, and industrial zones traditional firefighting systems are often delayed, ineffective, or unavailable due to poor infrastructure and accessibility.

These regions are particularly vulnerable due to:

- Presence of flammable substances.
- Limited fire-fighting resources.
- Long response times for emergency services.

There is a pressing need for an affordable, reliable, and rapid-response firefighting solution that can operate in such challenging environments. Our fire-fighting robot directly addresses this need.

This robot is ideal for: Chemical and gas plants, Solar farms, Industrial factories, Temporary military camps, Ordnance and ammunition storage, Ground-based fire safety at airports, Atomic power stations, Petrol and diesel stations, Remote and rural fire-sensitive zones,

Such areas typically involve ground-level operations rather than high-rise buildings, making this robot a perfect fit for small-scale industries and rural safety efforts. Its compact size, low cost, and semi-automatic operation make it accessible for anyone, anywhere in India. Even in countries with better infrastructure, this system can be adapted to suit different environments, proving its wide applicability and importance.

1.3 Objective of Project

The main objectives of this project are as follows:

- To design a robotic system capable of detecting and extinguishing fire automatically.
- To build a semi-automatic robot that requires human intervention only to reach the fire

zone.
• To use Arduino Uno, flame sensors, servo motors, and submersible water pumps for fire management.
• To ensure compatibility with ground-level vehicles such as carts or trolleys.
• To decrease human risk and improve fire response time.
• To provide a low-cost, scalable firefighting solution suitable for Indian rural and industrial use.
• To demonstrate a real-life embedded system application for public safety.
• To prioritize major fires using a combination of multiple sensor inputs.

1.4 Proposed System

The proposed solution is an Arduino Uno-based semi-automatic fire-fighting robot. It is mounted on a four-wheel platform and controlled using the L298N motor driver module.

The robot includes:
• 3 Flame Sensors (Left, Center, Right) for detecting fire direction.
• 1 Servo Motor to rotate the water nozzle in the detected direction.
• 1 Synronic Submersible Water Pump for spraying water.
• 1 TIP122 Transistor for switching high-current devices.
• 1 Relay Module powered by 4 standard batteries to power the pump and servo.
• 4 DC Motors (connected in parallel) for movement control.
• 1 AC to DC Adapter to power the Arduino Uno.

1.4.1 Advantages of Proposed System

The proposed system offers several key advantages, including:
• Reduces human risk in fire-prone zones.
• Fully automatic fire detection and extinguishing once in position.
• Can be mounted on any ground-level trolley, cart, or vehicle.
• Cost-effective and suitable for Indian industrial and rural conditions.
• Simple design without the need for remote or smartphone control.
• Prioritizes major fires using multiple flame sensors.
• Lightweight and portable for rapid deployment.
• Minimal power consumption and maintenance requirements.
• Ideal for dangerous zones like chemical plants, petrol stations, and ammo storage.

1.5 Working Principal

When any flame sensor detects fire, the robot automatically halts movement, rotates the nozzle toward the fire, and activates the water pump to extinguish it. If multiple sensors detect fire, the robot prioritizes the center zone for effective control.

1.6 Goal of the Project

The primary goal of this project is to design and develop a semi-automatic, Arduino-based fire-fighting robot that can:
• Detect and extinguish fire autonomously after reaching the fire-prone area,
• Minimize human risk in fire-fighting operations, especially in rural and industrial zones where traditional firefighting services are delayed or inaccessible.
• Provide a cost-effective and efficient alternative to large-scale fire-fighting systems by using easily available components and simple embedded technology.

This project aims to enhance public and industrial safety by creating a compact, reliable, and scalable fire-response system suitable for India's conditions and adaptable to global needs.

1.7 Scope of the project

The scope of a healthcare chatbot includes a wide range of functionalities and services related to healthcare information and support. Some key aspects of its scope include:

1. **Fire Detection and Directional Response: :** Utilizes three flame sensors to identify the presence and direction of the fire (left, center, right) and prioritizes areas with higher flame intensity.

2. **Semi-Automatic Operation: :** Requires manual transportation to the danger zone, after which the robot automatically takes over to detect and extinguish the fire

3. **Embedded System Integration: :** Incorporates Arduino Uno, servo motor, DC motors, L298N motor driver, TIP122 transistor, and a Synronic submersible pump for seamless automation and control.

4. **Vehicle Platform Compatibility: :** The system can be mounted on any standard mobile platform like a farm trolley, industrial cart, or temporary wheeled base, ensuring flexibility in deployment.

5. **Cost-Effective Design: :** Omits expensive features like ultrasonic sensors,

smartphone integration, or remote-control systems, making it affordable for rural and small-scale industrial use.

6. Application Suitability: : Suitable for use in a variety of high-risk zones such as chemical plants, solar farms, ammunition depots, fuel stations, temporary camps, and remote villages.

7. Scalability and Future Adaptation: : The robot design can be upgraded or adapted for enhanced applications in urban settings or larger industrial units with better infrastructure

1.8 Limitations

Despite its benefits, the system has the following limitations:

- Limited water capacity due to the size of the pump.
- Only suitable for ground-level fires (not for tall buildings).
- Requires a human to drive it to the fire zone (not fully autonomous).
- Doesn't detect or avoid obstacles (no ultrasonic sensors used).
- Using multiple servo motors for individual sensor control could improve precision but increases cost and complexity.
- Wind or water pressure conditions might affect pump performance.

1.9 Applications

This project finds its applications in the following areas:

- Solar and chemical plants
- Gas and petrol stations
- Industrial factories
- Military ammunition and ordnance areas
- Ground-level airport fire safety
- Atomic energy research facilities
- Remote villages and industrial zones
- Emergency fire response in temporary setups
- Farming areas and storage warehouses

CHAPTER 2

LITERATURE REVIEW

The concept of automated and semi-automated fire-fighting robots has evolved significantly over the past two decades with advancements in embedded systems, robotics, and sensor technology. Various research works and prototype systems have been developed to address fire emergencies, especially in hazardous and inaccessible environments. This chapter presents an overview of past work done in the field of fire-fighting robots, highlighting their features, technologies used, and limitations.

2.1 Previous Work and Technologies

-Researchers have developed fire-fighting robots using IR sensors and temperature sensors to detect flame and heat sources. However, IR sensors often fail in outdoor conditions due to interference from sunlight.
-In 2015, a project titled "Design and Implementation of a Fire Fighting Robot" used flame sensors, Arduino Uno, and DC motors but lacked directional water spraying mechanisms and was limited to a basic indoor prototype.
-An advanced project published in the International Journal of Engineering Research and Development (IJERD) used a wireless-controlled robot that required continuous manual operation, thus failing to reduce human involvement.
-Another design used ultrasonic sensors for obstacle avoidance and Bluetooth modules for remote control. While effective indoors, these systems were unsuitable for rugged, large outdoor terrains.

2.2 Gap in Existing Systems

-Most past projects are either fully manual or require continuous remote supervision using smartphones or computers.
-Very few systems focused on ground-level fire zones, especially in rural and industrial Indian contexts where fire trucks cannot easily reach.
-Prior models didn't support modular mounting on trolleys or carts, making them less versatile for real-world deployment.

2.3 Uniqueness of the Proposed System

-The current system combines the simplicity of manual vehicle transport with the intelligence of automatic fire detection and extinguishing, making it semi-automatic.
-The use of three flame sensors for directional fire detection and a servo-controlled water nozzle makes the system capable of handling fire from different directions.
-Unlike earlier systems, this robot uses a DC adapter, Transistors, and submersible pumps that are cost-effective and suitable for mass deployment.
-The system does not rely on smartphones, Bluetooth, or Wi-Fi, which ensures reliability in rural and offline regions.

2.4 Application-Based Review

Sr. No.	System	Technology Used	Limitations
01	IR-Based Fire Robot (2014)	IR sensor, Arduino Uno	Affected by sunlight; Not direction sensitive
02	Bluetooth control fire bot (2017)	Bluetooth, Flame sensor, pump	Needs constant smartphone control
03	Obstacle Avoidance Fire Robot (2019)	Ultrasonic sensors, IR flame detector	Complex logic; Costly hardware
04	Present Proposed System (2025)	Flame sensors, Servo motor, DC motors, Water pump, Arduino	Cost-Efficient, Directional, Semi-automatic

Table 2.1: Comparison of Fire-Fighting Robot Systems (2014–2025)

2.5 Conclusion of Literature Review

From the literature, it is evident that although various fire-fighting robots exist, most are either unsuitable for outdoor deployment or fail to reduce human risk. The proposed fire-fighting robot fills this gap by offering an affordable, modular, and reliable solution. It is particularly well-suited for chemical plants, atomic stations, solar farms, military camps, and Indian industrial zones, where quick response and minimal human risk are crucial

CHAPTER 3

PROBLEM IDENTIFICATION

In rural and industrial areas of India, fire accidents often cause severe damage due to delayed response from fire brigades. Narrow lanes, poor infrastructure, and the unavailability of high-tech fire safety systems make it difficult to control fires in time. Most fire emergencies require manual intervention, which is risky and inefficient. Additionally, there is a lack of low-cost, reliable, and mobile fire-fighting systems that can reach remote areas and work automatically. Fires in farms, chemical warehouses, and isolated factories often go unnoticed or uncontrolled due to these limitations. This calls for a solution that is affordable, portable, and semi-automatic, which can detect and extinguish fires quickly before they spread. There is a need for a system that can be integrated with basic vehicles and controlled with simple electronics, reducing the dependence on human efforts and improving safety in fire emergencies.

3.1 Existing System

The current fire-fighting systems heavily depend on manual operations by human firefighters who physically access the fire scene and operate hoses or fire extinguishers. In urban areas, fire brigades are equipped with advanced equipment and vehicles, but their response time is often delayed in rural or industrial zones due to poor road connectivity and infrastructure. Additionally, robotic fire-fighting systems do exist in some parts of the world, but they are costly, complex, and mostly designed for city buildings or large industries. These systems are not suitable for open, uneven terrain or for use on common vehicles like trollies or carts. As a result, many smaller regions and facilities remain unprotected due to the high cost and complexity of these solutions. There is a clear gap in providing affordable and practical fire-fighting technology for ground-level fires in challenging environments.

3.1.1 Disadvantages of Existing System

- The major drawback of existing systems is their inaccessibility to common people and small industries.
- The cost of robotic systems and the technical expertise required to operate them make them unfit for rural applications.
- Fire brigades also face difficulties reaching fire locations quickly due to narrow roads and traffic congestion, especially in remote towns and villages.
- Human firefighters often risk their lives in such operations, as there is no automation or remote-control mechanism involved.
- The use of bulky equipment and the need for trained professionals further adds to the challenges.
- Most of these systems are not portable or flexible enough to be mounted on regular vehicles like farm trollies.
- These limitations have created an urgent need for a better solution that is simple, low-cost, and quick to deploy.

3.2 Proposed System

The proposed system is a semi-automatic fire-fighting robot designed using an Arduino Uno microcontroller, three flame sensors, a servo motor, a mini water pump, and a nozzle for water spraying. It is made to be mounted on a commonly available cart or trolley, allowing easy transport to the fire location by a human driver. Once the system is brought near the fire, it activates automatically based on flame sensor input. The servo motor rotates the nozzle toward the direction of the detected flame, and the pump sprays water to extinguish it. If multiple sensors detect fire, the system gives priority to the central sensor to ensure the main flame is targeted first. After the fire is extinguished, the robot resumes movement to check for additional fire. This system is cost-effective, easy to build, and suitable for open ground-level fires. It is especially useful in farms, fuel stations, solar panels, and military or ordinance depots. It reduces human risk and increases efficiency by automating the dangerous task of extinguishing fires.

3.3 Feasibility Study

To ensure that the proposed system can be effectively implemented in the real world, a detailed feasibility study has been conducted. This study focuses on three main aspects: economical, technical, and social feasibility. Each aspect helps assess how suitable and practical the system is for use in real-time fire emergencies. The study confirms that this

robot is not only affordable and technically sound but also valuable in terms of social impact and safety improvement. The findings are discussed in the following sections.
1. Economical feasibility
2. Technical feasibility
3. Social feasibility

3.3.1 Economical Feasibility

The fire-fighting robot is built using low-cost and easily available electronic components. Devices like the Arduino Uno, flame sensors, DC motors, mini pumps, can be purchased from local electronics shops at affordable prices. Since the robot can be mounted on existing vehicles like carts or trolleys, there is no need to invest in a custom robotic base. This makes it suitable for low-budget setups, including rural farms, storage yards, and small factories. The system also requires minimal maintenance and can be repaired using basic tools and knowledge. Its low cost makes it a scalable and repeatable solution for communities or industries that cannot afford high-end robotic systems. This cost-effectiveness ensures wider adoption and long-term usage in critical fire safety applications.

3.3.2 Technical Feasibility

Technically, the robot is built using simple and reliable components that ensure smooth operation in fire emergency situations. The Arduino Uno serves as the brain of the system, reading sensor inputs and controlling the servo motor and water pump. Three flame sensors are placed at different angles to detect fire direction, allowing accurate targeting of the nozzle. A servo motor rotates the nozzle, and a TIP122 transistor, along with a relay module, controls the power to the water pump. The L298N motor driver runs 4 DC motors connected, providing strong movement across rough surfaces. The system is powered using a 12V ACDC adapter for the Arduino and a 4-cell battery pack for the motors and pump. Programming is done using Arduino IDE, and the logic is coded in embedded C. Overall, the system is technically simple, robust, and reliable for field use.

3.3.3 Social Feasibility

The fire-fighting robot provides a socially beneficial solution, especially for areas where fire brigades cannot reach on time. In remote villages and small industrial setups, it is difficult to control fires quickly, and people often rely on unsafe manual efforts. This robot offers a safe and quick way to respond to fire without risking human life. It is easy to use, can be operated by anyone with basic training, and reduces the need for professional firefighters. Since it can be mounted on any local vehicle, it becomes a highly portable and flexible solution. The system encourages the use of embedded technology for life-saving purposes and spreads awareness about affordable automation in fire emergencies. It brings safety within reach of every community, ensuring better preparedness and faster response during fire accidents

3.4 Requirements

3.4.1 Hardware specifications

Sr. No.	Hardware Name	Hardware Description
01	Arduino Uno	Microcontroller used to control the entire system
02	Flame Sensors (3 units)	Detect fire direction and send signals to the Arduino.
03	Servo Motor	Rotates the water nozzle toward the direction of the fire.
04	Water Pump	Sprays water when fire is detected.
05	Relay Module	Acts as a switch to control the water pump.
06	TIP122 Transistor	Helps in switching high-power devices like the pump.
07	L298N Motor Driver	Controls the direction and speed of the DC motors.
08	DC Motors (4 units)	Move the robot forward, backward, left, and right.
09	DC Adapter (9V)	Powers the Arduino and low-voltage components.
10	4-Cell Battery Pack	Supplies power to the DC motors and water pump.
11	Chassis / Trolley Base	The platform to hold and support all components.
12	Water Nozzle	Directs the water spray accurately at the fire.
13	Connecting Wires and Solder	Used to connect and fix all components together.

Table 3.1: Hardware Components Used in Fire-Fighting Robot Project

3.4.2 Software specifications

Sr. No.	Software Name	Software Description
01	Arduino IDE	Used to write, compile, and upload code to the Arduino Uno.
02	Embedded C	Programming language used to write the robot's control logic.
03	Windows OS	Operating system to run Arduino IDE and other tools on a PC or laptop.
04	USB Driver	Helps connect the Arduino board to the computer via USB cable.
05	Serial Monitor	Tool in Arduino IDE to test and debug the sensor readings and outputs.

Table 3.2: Software Tools Used in Fire-Fighting Robot Project

CHAPTER 4

SYSTEM DESIGN

4.1 Description

System design is a crucial phase in any embedded system project where the overall architecture and working model of the system are planned. In this fire-fighting robot project, the system is built around the Arduino Uno microcontroller and includes flame sensors, a servo motor, a submersible water pump, a relay module, a TIP122 transistor, and an L298N motor driver to control the movement of DC motors. The design focuses on ensuring that all components are connected efficiently and that the robot performs fire detection and extinguishing actions smoothly and reliably. This chapter presents a complete understanding of the circuit design, system block diagram, component selection, and Arduino Uno board layout. The robot's operation is semi-automatic, meaning that while it requires manual positioning at the fire site, all detection and fire extinguishing actions are performed automatically based on sensor inputs.

4.2 System Architecture & Circuit Diagram

The circuit diagram of our fire-fighting robot plays a vital role in understanding how various electronic components work together in coordination to detect and extinguish fire. It represents the complete electrical layout of the robot and shows how each hardware component is connected to the Arduino Uno. Three flame sensors are connected to the analog pins A0, A1, and A2 of the Arduino Uno. These sensors detect infrared light from fire and send analog signals based on the intensity of the flame. The servo motor, responsible for rotating the nozzle, is connected to digital pin 9. The submersible water pump is connected to a TIP122 transistor which acts as a switch, and is powered through a relay module, all controlled via digital pin 8 for efficient switching and silent operation.

To control the robot's movement, we use an L298N motor driver module. It receives control signals from pins 2, 3, 4, and 5 and powers four DC motors connected in parallel for movement in all directions. The enable pins ENA and ENB are used for speed control via PWM. Power to the Arduino is supplied by a 9V DC adapter, and a separate 4-cell battery pack powers the motor driver and pump. All connections are carefully soldered for durability and proper signal flow, with a shared common ground to avoid malfunctioning.

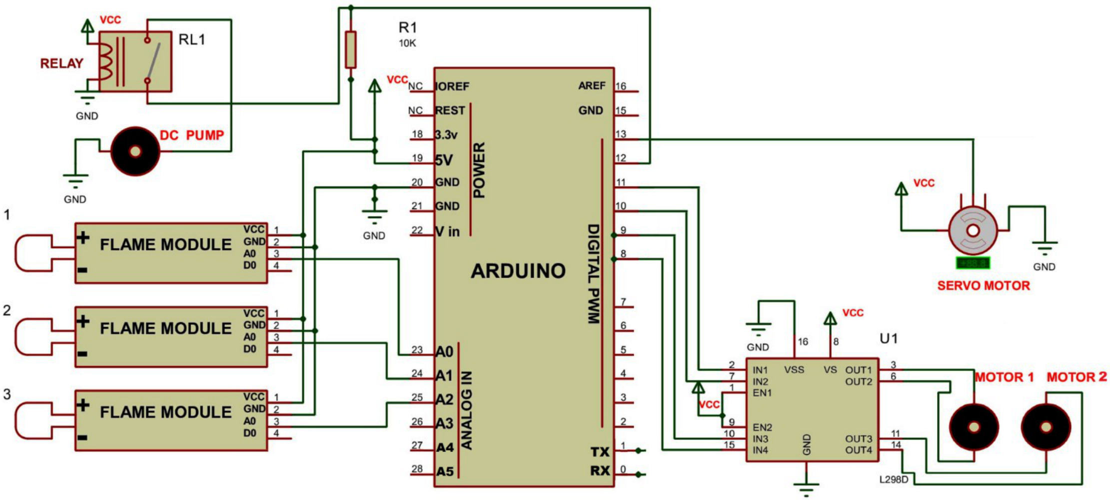


Fig. 4.1: System Architecture & Circuit Diagram

4.2.1 Description of All Connections

1. Flame Sensors: Three flame sensors are placed at the left, center, and right positions on the robot. These are connected to analog pins A0, A1, and A2 of the Arduino Uno. They continuously monitor for fire and send signals based on IR flame detection.

2. Servo Motor: The servo motor used for nozzle direction is connected to digital pin 9. When any flame sensor detects fire, the Arduino rotates the servo to aim the nozzle in the corresponding direction.

3. Water Pump with TIP122 Transistor and Relay Module: The water pump is powered by a 4-cell battery through a relay module, and switching is done using a TIP122 transistor. A flyback diode is added across the pump terminals to prevent voltage spikes.

4. DC Motors and Motor Driver (L298N): The L298N motor driver is connected to pins 2, 3, 4, and 5 to control four DC motors used for robot movement. ENA and ENB pins are used for PWM speed control from pins 10 and 11.

5. Power Supply: The Arduino is powered by a 9V DC adapter. A separate 4-cell battery pack provides power to the motors, relay, and water pump.

6. Ground Connections: All components share a common ground to ensure stable voltage reference and avoid erratic behavior.

4.2.2 Use of TIP122 Transistor and Relay Module

Both TIP122 and relay module are used for controlling high-power components like the water pump and servo motor, without affecting the Arduino board.

TIP122 Transistor:
TIP122 is an NPN Darlington pair transistor used to switch high-current loads using a small current from the Arduino.A 1k resistor connects the digital pin to the base of the transistor. When the pin is set HIGH, the pump circuit is completed and current flows.

Relay Module:
The relay module connects/disconnects the pump and servo motor from a separate power supply. It prevents Arduino from experiencing voltage drops due to high-current operations.

Working Together:
1. Arduino activates the relay to connect the pump circuit to the battery.
2. Then, it triggers TIP122 to start the pump.
3. This ensures safe power isolation and reliable switching.

Wiring Summary:

- Pump positive terminal → Relay module output → 9V battery
- Pump negative terminal → TIP122 collector
- TIP122 emitter → Ground
- TIP122 base → Arduino digital output via 1k resistor

4.2.3 Block Diagram

The block diagram offers a simplified logical overview of the fire-fighting robot's functioning. It illustrates how various components interact and how data flows from input detection to output actions.
Logical Flow:
1. Flame Sensors → Arduino Uno → Servo Motor
2. Arduino Uno → TIP122 Transistor → Relay Module → Water Pump
3. Arduino Uno → L298N Motor Driver → DC Motors

When a flame sensor detects fire, the Arduino Uno receives the signal, processes it, and performs three actions simultaneously: it rotates the nozzle in the direction of fire using the servo motor, turns ON the water pump through the TIP122 transistor and relay module, and pauses the robot's movement to allow precise spraying. Once the fire is extinguished, the Arduino resumes robot movement. This logical arrangement makes the system smart, responsive, and semi-autonomous.

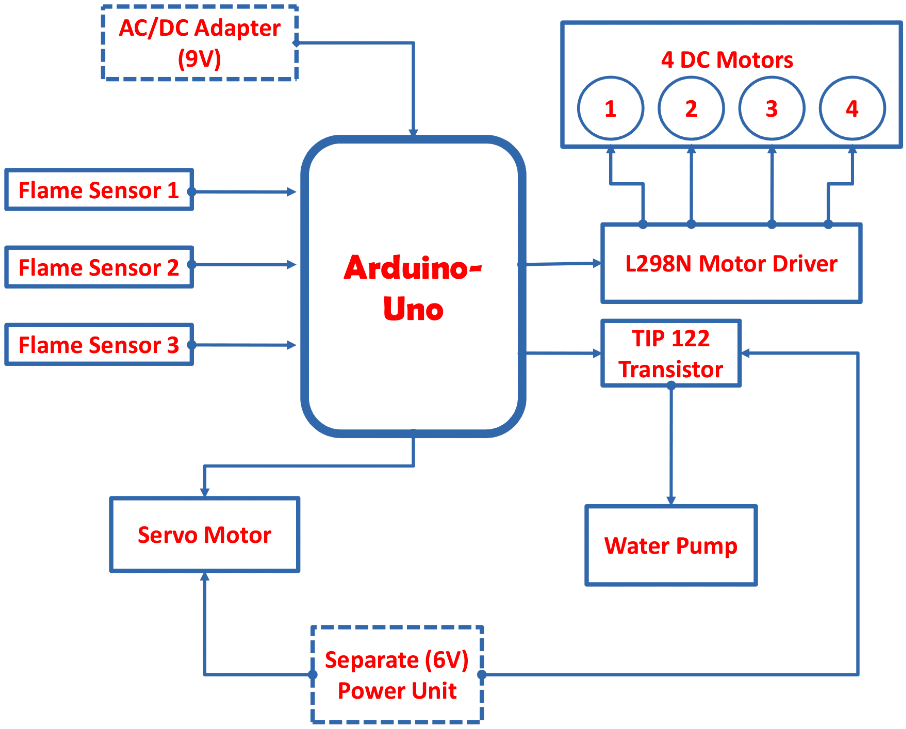


Fig. 4.2: Block Diagram

4.2.4 Arduino Uno Diagram & Details

At the heart of the system lies the Arduino Uno, a powerful and open-source development board based on the ATmega328P microcontroller. It serves as the brain of the fire-fighting robot by reading sensor inputs and executing logic to control outputs.

Specifications of Arduino Uno:

- Microcontroller: ATmega328P
- Operating Voltage: 5V
- Input Voltage (Recommended): 7–12V
- Input Voltage (Recommended): 7–12V

- Digital I/O Pins: 14 (of which 6 provide PWM output)
- Analog Input Pins: 6
- Flash Memory: 32 KB (0.5 KB used by bootloader)
- SRAM: 2 KB
- EEPROM: 1 KB
- Clock Speed: 16 MHz

The Arduino Uno receives input from the flame sensors, processes the data using embedded C code uploaded via the Arduino IDE, and sends output commands to the TIP122 transistor (for the water pump), the servo motor, and the L298N motor driver. It is user-friendly, reliable, and compact—making it an ideal choice for automation and embedded projects like this fire-fighting robot.

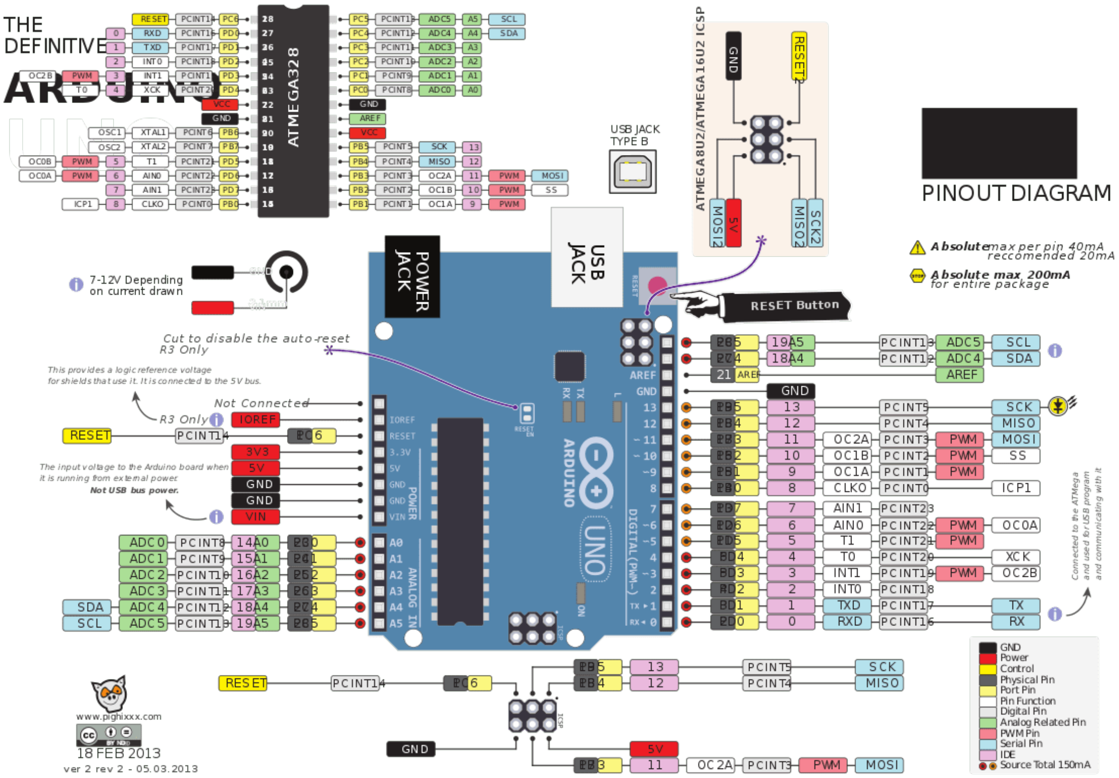


Fig. 4.3: Arduino Uno Pinout Diagram

CHAPTER 5

DESIGN

5.1 UML Diagrams

Unified Modeling Language (UML) diagrams are standardized visual representations used to model the structure and behavior of software or embedded systems. They provide a clear and organized way to document the components, interactions, and dynamic flows of a system. In the context of embedded systems, robotics, and automation, UML diagrams help both developers and stakeholders visualize how various modules of the system interact, communicate, and operate over time.

UML plays a crucial role during system design, especially for:

- Understanding system architecture before implementation
- Identifying hardware-software interfaces
- Documenting logical flow and control mechanisms
- Debugging and maintaining embedded or robotic systems

In an Arduino-based fire-fighting robot, UML helps illustrate:

- How sensors detect fire
- How decisions are made to move the robot or spray water
- How components like motors, pumps, and servos are coordinated

In this project, we have included the following UML diagrams:

- Use Case Diagram – to show interactions between user and system
- Class Diagram – to describe the structure and data handling
- Sequence Diagram – to represent the time-ordered operation flow
- Activity Diagram – to detail the dynamic behavior and system logic

5.1.1 Activity Diagram

An Activity Diagram is used to model the workflow of the system. It represents the dynamic behavior of the system by illustrating the flow of activities, decisions, parallel processes, and their interactions.

Key Components:

Purpose: Depicts workflows and the flow of control or data in a system.

Usage: Represents the logic behind the fire-fighting robot's operation, such as detecting a fire, determining the fire's position, and activating the water pump.

Components: Actions, decisions, start/end nodes, and transitions.

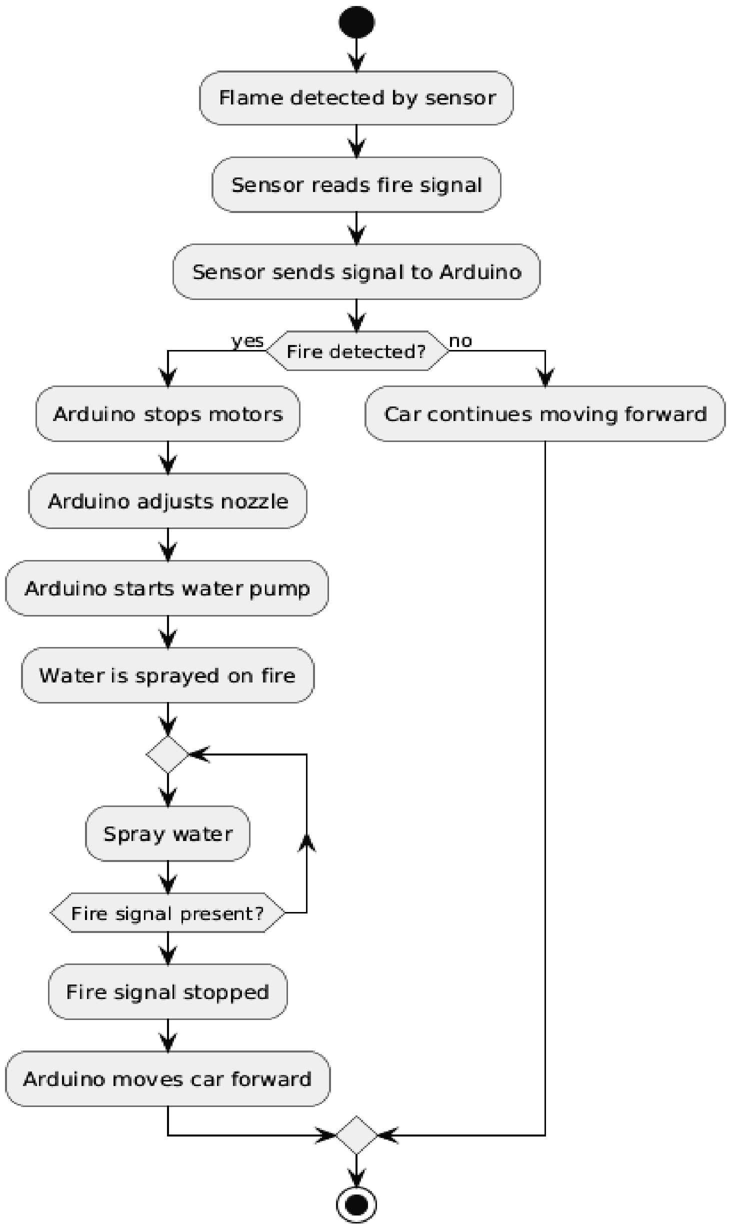


Fig. 5.1: Activity Diagram

5.1.2 Class Diagram

A Class Diagram is used to represent the static structure of a system by showing its classes, attributes, methods, and the relationships between them. It's an essential tool in object-oriented design.

Key Components:

Purpose: Defines the structure of classes and their relationships.

Usage: Helps in understanding the different software components (e.g., fire detection, robot movement, and fire-extinguishing modules) in the system.

Components: Classes, attributes, methods, and associations (like inheritance or aggregation).

Example in Your Project:

Classes: FireDetection, MovementControl, WaterPumpControl

Attributes: Flame sensor data, motor speed, servo angle.

Methods: detectFire(), moveForward(), activatePump()

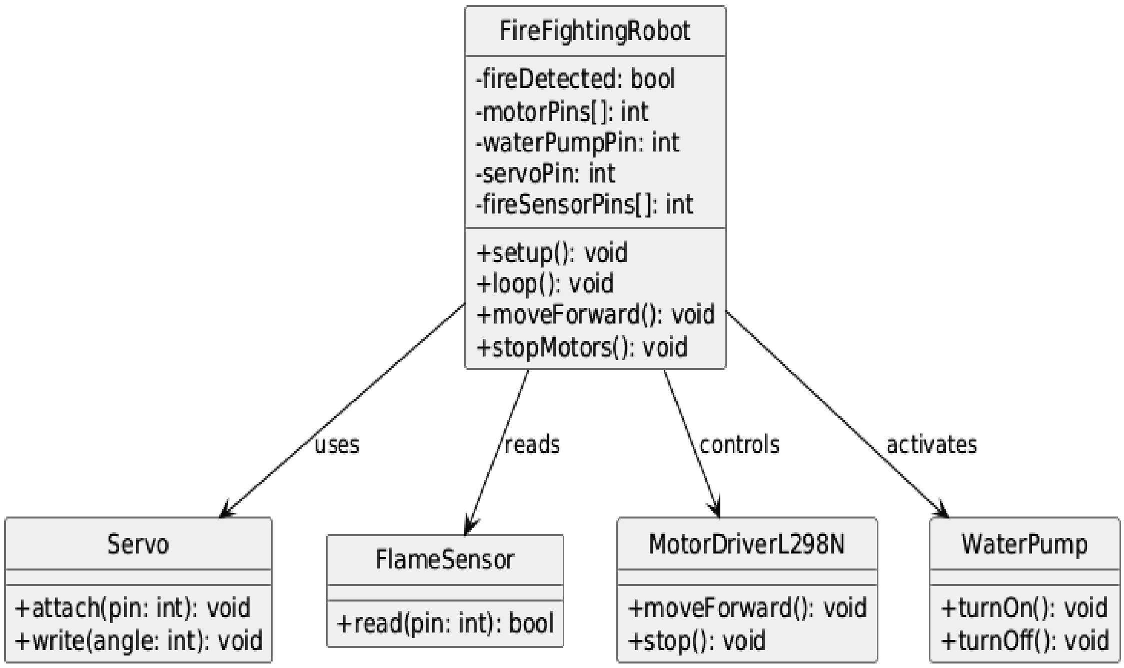


Fig. 5.2: Class Diagram

5.1.3 Sequence Diagram

A Sequence Diagram models the interaction between different components of a system over time. It shows the sequence of events or messages passed between objects or components in the system, often depicting how the system responds to external stimuli.

Key Components:

Purpose: Represents interactions in a time-ordered sequence.

Usage: Shows the communication between different parts of the fire-fighting robot, such as how the Arduino communicates with sensors, motors, and the water pump.

Components: Actors (e.g., sensors, user, or control system), messages (arrows), and lifelines (objects that exist over time).

Example in Your Project:

The Arduino Uno sends signals to flame sensors and receives inputs.

The Arduino sends control commands to the servo motor and water pump.

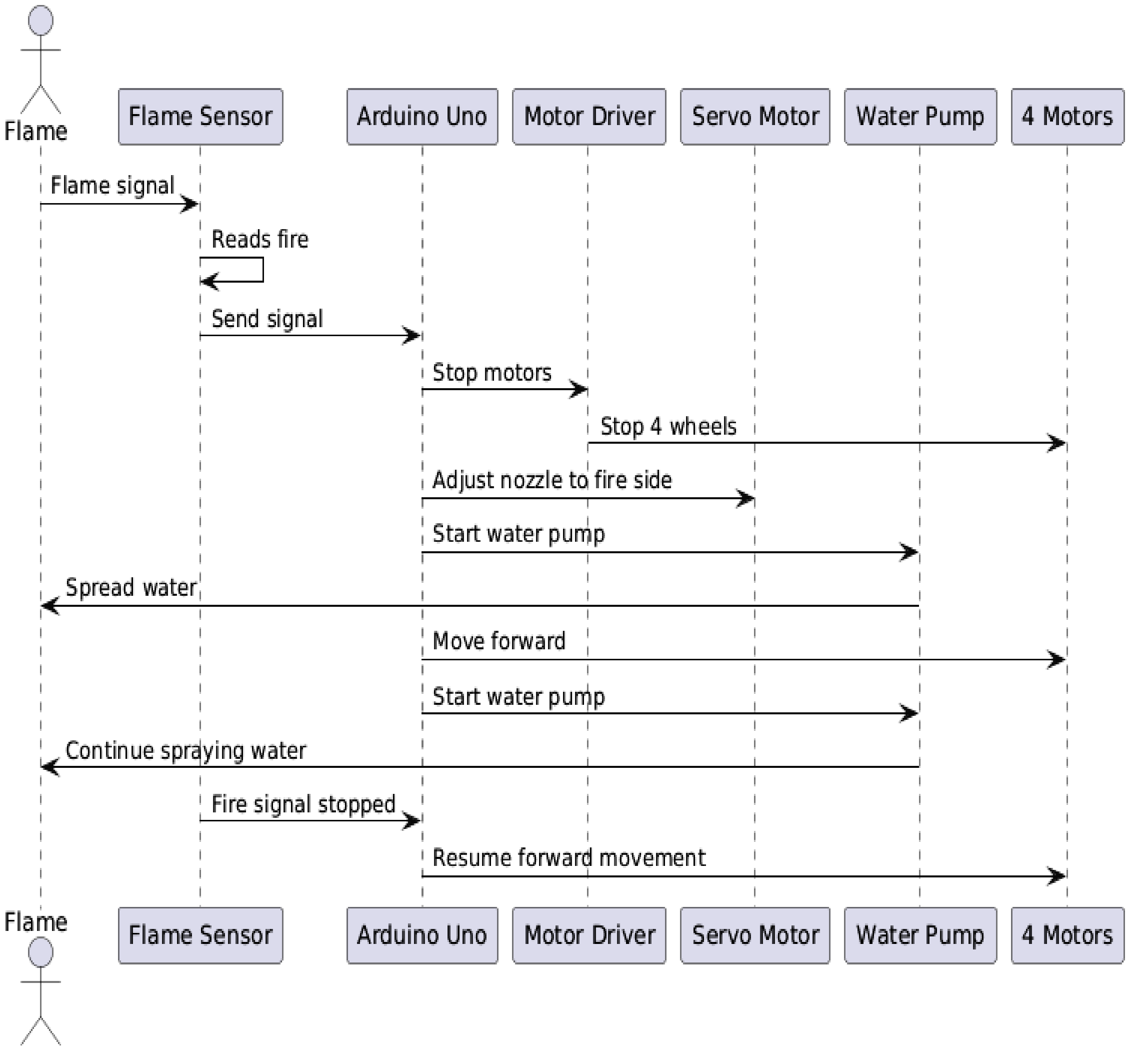


Fig. 5.3: Sequence Diagram

5.1.4 Use-Case Diagram

A Use Case Diagram is a behavioral UML diagram that shows the interactions between users (or actors) and the system. It identifies the main functionalities of the system from the user's perspective.

Key Components:

Actors: The users or external systems interacting with the system (e.g., human driver, sensors).

Use Cases: Functionalities or actions that the system provides (e.g., detect fire, rotate nozzle, activate pump).

Associations: Relationships between actors and use cases showing how they interact.

Application: In your project, actors like the human driver, fire sensors, and the Arduino controller interact to perform tasks like fire detection, robot movement, and fire extinguishing.

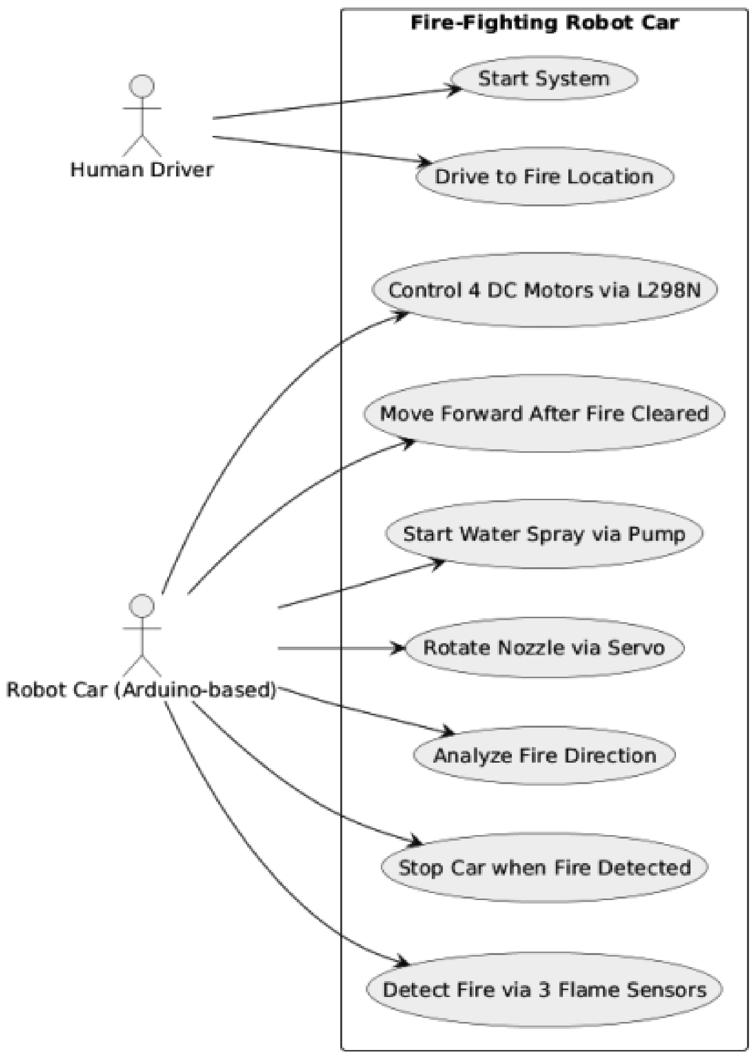


Fig. 5.4: Use-Case Diagram

5.2 Data Flow Diagrams

A Data Flow Diagram illustrates how data moves within the system, showing the flow of information between various components and processes. It helps to visualize the system's operations and the data exchange between entities.

Key Components:

Processes: Functions that transform inputs into outputs (e.g., fire detection, data processing).

Data Stores: Where data is stored in the system (e.g., sensor data).

External Entities: Sources or destinations of data (e.g., human driver, external environment).

Data Flows: Arrows representing the movement of data between processes, stores, and external entities.

Application: For the fire-fighting robot, the DFD would show how data from the fire sensors is processed by the Arduino, triggering the appropriate actions like moving the robot or activating the water pump.

5.2.1 DFD Level 0 – Context Level

The Level 0 Data Flow Diagram, also known as the Context Level Diagram, represents the entire fire-fighting robot system as a single process. It highlights the interaction between the system and external entities such as the Human Driver and the Fire Location. The Human Driver positions the robot near the suspected fire, and the system receives fire detection input from the environment. The robot processes this data and responds with movement and fire-extinguishing actions. This diagram provides a high-level overview of the system's boundaries and its communication with external elements.

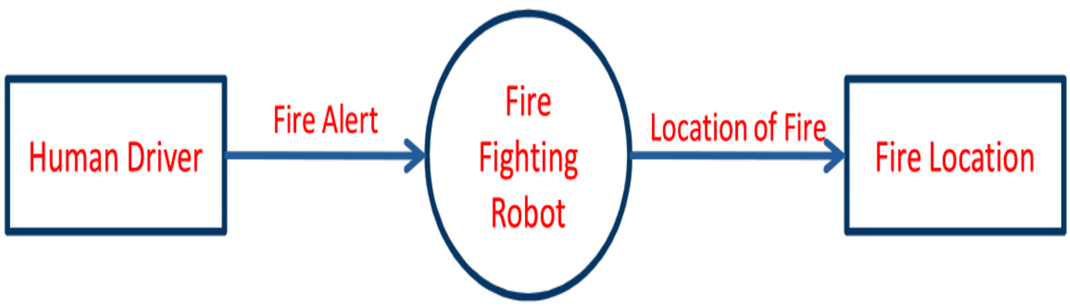


Fig. 5.5: 0 Level Data Flow Diagram

5.2.2 DFD Level 1 – System Overview

The Level 1 Data Flow Diagram expands the context-level DFD by dividing the overall fire-fighting system into main functional modules. It includes sub-processes such as flame detection, data processing, decision-making, and fire-extinguishing operations. The diagram shows how the Human Driver inputs the system, how flame sensors interact with the Arduino for data collection, and how control signals are sent to the motor driver, servo motor, and water pump. It also outlines data flow between these processes, emphasizing coordination and logical sequencing within the robot system.

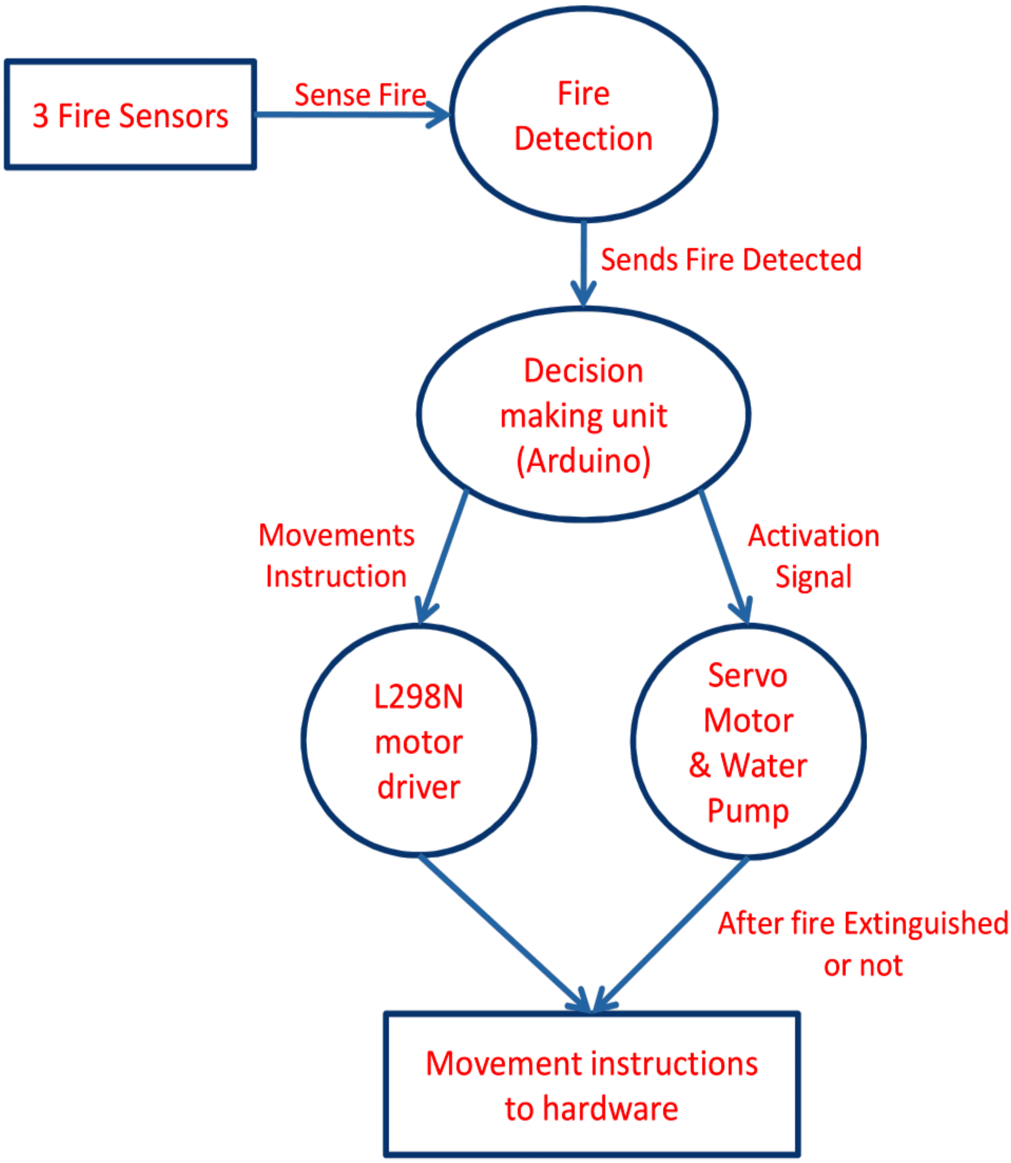


Fig. 5.6: 1 Level Data Flow Diagram

5.2.3 DFD Level 2 - Detailed Functional View

This diagram provides an in-depth view of individual sub-processes defined in Level 1. It elaborates on how flame sensors collect data, how the Arduino processes sensor inputs, and how commands are sent to specific modules like the motor driver, servo motor, and water pump. It also shows feedback loops, decision logic for prioritizing fire zones, and real-time control flow for extinguishing actions. This level helps in understanding the internal logic and data handling at a component level in the fire-fighting robot system.

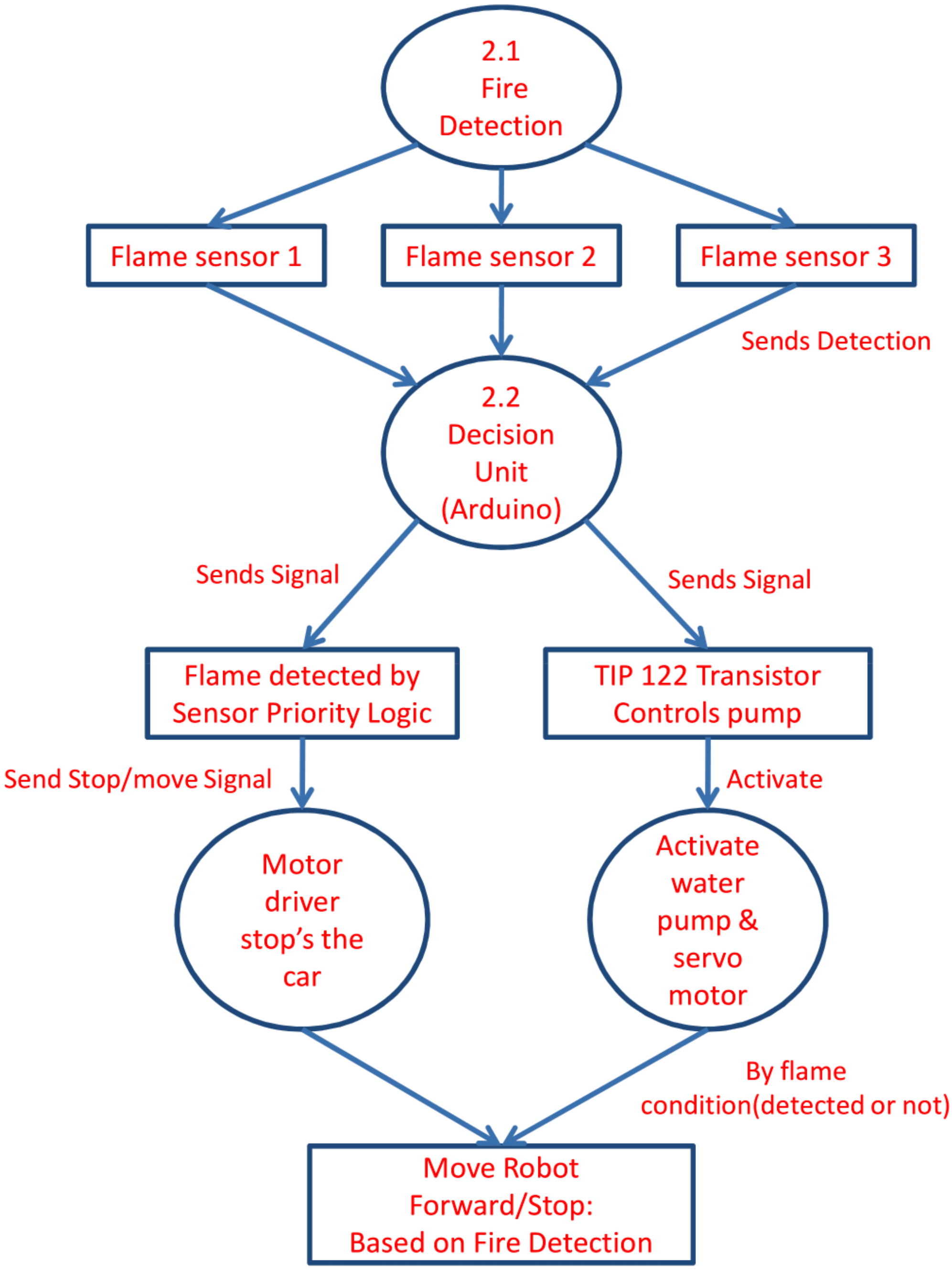


Fig. 5.7: 2 Level Data Flow Diagram

CHAPTER 6

IMPLEMENTATION

Implementation is the most crucial phase of the project where theoretical ideas are transformed into a working model. For our fire-fighting robot, implementation involved multiple tasks like component procurement, circuit design, Arduino programming, testing, and final assembly. Each task was tracked weekly with clear goals, actions taken, issues encountered, and their solutions. This chapter describes the entire process step-by-step along with submodules, technology used, and real-time progress.

6.1 System Development and Implementation

6.1.1 Components Gathering

Initially, all necessary electronic and mechanical components were identified.

We purchased the following:

- Arduino Uno
- DC Motors
- Servo Motor (for nozzle)
- 3 Fire Sensors (IR-based)
- Motor Driver L298N
- Chassis & Wheels
- Water Pump (Synronic 3–6V Submersible)
- Relay Module Of 4 batteries
- DC Power Adapter
- Jumpers, Resistors, Connectors, Pipes

6.1.2 Circuit Design

The full schematic was designed using basic prototyping tools.

It included:

- Motor driver for motion control
- Servo connection for rotating nozzle
- Three fire sensors at left, center, and right
- A TIP122 transistor circuit with relay for pump control
- Parallel motor wiring (actual 4 motors for 4 wheels)
- The circuit was revised after initial errors in motor wiring.

6.1.3 Arduino Programming

The Arduino was programmed using Embedded C (C with Arduino-specific libraries).

Technologies/libraries used:

- include <Servo.h> for servo motor control
- PWM logic via analogWrite() for motor speed
- digitalRead() to check sensor inputs
- digitalWrite() to activate motors and water pump

The code logic included:

- Default forward movement
- Fire detection using 3 sensors
- Servo-based direction for nozzle
- Pump activation when fire is detected
- Motor stop and resume behavior
- Language Used: Embedded C (Arduino IDE)

6.1.4 Testing Phase

First testing phase began in Week 5 using the initial code.

Issues faced:

- Delayed pump activation
- Motors not syncing with sensors

Fixed using:

- Updated delay logic, Rewiring motor connections
- Threshold calibration in code

6.1.5 Full Assembly

- In Week 7, complete structure was built with water tank mounted.
- Servo and pump were aligned for accurate targeting.
- Nozzle was placed using PVC pipe through servo-mounted mechanism.

6.1.6 Final Testing & Calibration

- After implementing Version 2 code in Week 9:
- Servo rotates precisely (Left = 150°, Center = 90°, Right = 28°)
- Fire detection is instant
- Pump sprays accurately on fire source Multiple tests were conducted to ensure:
- Robot halts during fire detection
- Continues forward only after fire is cleared
- Prioritizes central fire if multiple sensors are triggered

6.1.7 Installation

- Wiring was bundled and secured
- Chassis was locked using bolts and glue gun
- Electronics were packed inside a plastic casing to prevent water damage
- DC supply was connected to Arduino Uno

6.1.8 Documentation Maintenance

Started from Week 12.
Half of the Word file completed by Week 13 Contains:
- Component explanation
- Circuit logic
- Testing stages
- Error handling
- Maintenance suggestions:
- Clean fire sensors regularly
- Replace water pump after long use
- Recharge batteries before use

6.1.9 Additional Points

- Relay module controls high power to pump via Arduino trigger
- TIP122 transistor used to amplify control signal
- L298N motor driver supports two motors each side with separate forward/backward logic
- Servo controlled by 5V via Arduino with default neutral position
- Semi-automatic Design: User takes car to fire zone; robot handles detection and extinguishing

6.2 Modules Used

Sr. No.	Module Name	Description
01	Component Collection Module	Identified and purchased sensors, motors, Arduino, chassis, and water pump across multiple rounds due to design updates and stock issues.
02	Circuit Design Module	Designed wiring between Arduino, sensors, motor driver, relay, and servo; finalized power layout and pin mapping.
03	Mechanical Assembly Module	Assembled physical structure including chassis, motors, sensors, and Sprite bottle water tank.
04	Arduino Programming Module	Used Arduino IDE and Embedded C to control sensors, motors, servo, and pump.
05	Initial Testing & Debugging Module	Tested initial code and wiring; fixed motor directions, sensor logic, and pump triggering.
06	Final Assembly Module	Finalized all components and wiring with mounted pump and water tank for full functionality.
07	Code Enhancement & Synchronization Module	Improved servo angle logic, removed delays, and ensured real-time sensor-pump sync.
08	Complete Testing & Calibration Module	Calibrated servo angles, motor speed, fire detection, and replaced faulty components.
09	Final Model Verification Module	Verified fire detection, nozzle rotation, and robot behavior post-fire extinguish.
10	Documentation & Maintenance Module	Created detailed project report; suggested regular cleaning, pump replacement, and battery charging.

Table 6.1: Project Development Modules in Fire-Fighting Robot

6.3 Technology Used

Sr. No.	**Technology**	**Details**
01	Programming Language	Embedded C via Arduino IDE
02	Microcontroller	Arduino Uno R3
03	Motor Driver	L298N Dual H-Bridge
04	Sensors	IR Flame Sensors (Digital LOW = Fire)
05	Pump Control	Relay Module with TIP122 Transistor
06	Servo Library	`<Servo.h>` for nozzle rotation
07	Speed Control	PWM using `analogWrite()` to ENA/ENB pins
08	Debugging Tools	Serial Monitor using `Serial.begin()` and `Serial.print()`

Table 6.2: Technologies and Tools Used in the Fire-Fighting Robot

CHAPTER 7

TESTING

7.1 Types of Testing

7.1.1 Unit Testing

Unit testing focuses on checking individual components separately before combining them.

Tested Units:

1. Flame Sensors – checked for flame detection range and angle.
2. Servo Motor – tested for angle rotation based on different sensor input.
3. Water Pump – tested with manual relay switching.
4. L298N Motor Driver – tested for directional movement of wheels.
5. Relay Module – checked for water pump control.
6. Arduino Uno – tested for uploading code and correct signal processing.

Results:

All units worked independently without errors.

7.1.2 Integration Testing

Integration testing verifies how different components work together when connected.

Key Integrations:

- Sensor + Arduino + Servo Motor
- Arduino + Relay + Pump
- Arduino + Motor Driver + Wheels

Issues Found:

1. Servo delay when multiple sensors were triggered simultaneously.

2. Pump activation timing mismatch with sensor input.

Fixes:

1. Code optimized to prioritize central sensor.
2. Delay functions adjusted for smooth transitions.

7.1.3 Functional Testing

This test ensures that the robot performs all expected functionalities correctly.

Functional Scenarios Tested:

1. Detect fire from various directions.
2. Rotate nozzle towards active fire.
3. Activate pump and spray water.
4. Resume movement after extinguishing the fire.

Testing Tools:

1. Small flame (from lighter/candle) used to trigger sensors.
2. Water container used for real-time pump testing.
3. Outcome: All functionalities performed accurately in most test cases.

7.1.4 Functional Testing

System testing involves the complete setup being tested in real-time, simulating real-life fire scenarios.

Test Environment:

1. Open ground with minimal wind for controlled flame detection.
2. Robot placed on level ground with a small flame source at different angles.

System Behavior Observed:

1. Proper detection of fire.
2. Servo rotated towards fire direction.
3. Water pump activated and extinguished the fire.
4. Robot paused while extinguishing and resumed after success.

Conclusion:

System behaved as expected under different fire conditions.

7.1.5 White Box Testing

White box testing focuses on the internal working of the Arduino program. **Activities Performed:**

1. Reviewed Arduino code logic manually.

2. Checked loop execution, if-else conditions for flame direction.
3. Verified angle values and output signals for each sensor.

Advantage:

Helped in identifying redundant code, unnecessary delays, and improved execution time.

7.1.6 Black Box Testing

This test focuses only on input-output behavior without knowing internal code.

Performed By:

1. Team members who didn't view the code.
2. Observed Behaviors:
3. Fire detection triggered actions correctly.
4. Spraying mechanism operated when expected.
5. No wrong or delayed actions occurred during test.

Purpose:

To simulate a real user's experience and observe system response.

7.2 Integration Testing

Integration testing verifies that different software or Hardware modules or components work together correctly as a single system.

Integration testing was performed in multiple phases:

1. First, the flame sensors and Arduino were connected to verify signal accuracy.
2. Then, the servo was added to check if it moved correctly based on sensor input.
3. Lastly, the relay and pump were connected to ensure water was sprayed only when fire was detected.

Example Issue Faced:

When both left and right sensors were triggered, the servo got confused. This was solved by prioritizing the center sensor using nested if conditions in Arduino code.

Result:

Smooth communication between modules, with correct sequencing of actions.

7.3 Acceptance Testing

This testing was conducted at the final stage to ensure the robot meets user requirements.

Acceptance Criteria:

1. Should detect fire from at least 3 directions.
2. Should rotate the nozzle accurately.
3. Should spray water and turn off after flame disappears.
4. Should resume movement after fire is extinguished.

Testing Steps:

1. Set fire at different points.
2. Observe robot actions without intervention.
3. Repeat with slight code variations to test consistency.

Status:

Robot passed all acceptance scenarios and was ready for demonstration.

7.4 Manual Testing

Manual testing was performed throughout the development phase.

Checklist Used:

1. Flame sensors output verified on serial monitor.
2. Relay manually tested with ON/OFF commands.
3. Servo angle adjusted using code until correct.
4. Pump water flow checked visually.
5. Wiring checked for loose or wrong connections.

Advantages:

1. Early detection of minor issues.
2. Quick fixes before running complex tests.
3. Better understanding of component behavior.

CHAPTER 8

CODING

```
#include <Servo.h> // Include the Servo library to control the water nozzle

// Motor Driver Pins (L298N motor driver module)
const int motorLeftForwardPin = 6;   // IN1 - Controls left motor forward motion
const int motorLeftBackwardPin = 7;  // IN2 - Controls left motor backward motion
const int motorRightForwardPin = 8;  // IN3 - Controls right motor forward motion
const int motorRightBackwardPin = 9; // IN4 - Controls right motor backward motion

// PWM Pins for speed control (optional - ENA and ENB)
const int enableLeftMotor = 5;   // Speed control for left motor using PWM
const int enableRightMotor = 10; // Speed control for right motor using PWM

// Relay-connected water pump control pin
const int waterPumpPin = 2; // Turns ON/OFF the water pump

// Servo Motor Pin (rotates nozzle to aim at fire direction)
const int servoPin = 3;

// Fire Sensor Inputs (IR flame sensors)
const int fireSensor1Pin = 11; // Left flame sensor
const int fireSensor2Pin = 12; // Center flame sensor
const int fireSensor3Pin = 13; // Right flame sensor

// Servo object to control nozzle direction
```

```
Servo waterServo;

// Flag to track fire detection status
bool fireDetected = false;

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

  pinMode(motorLeftForwardPin, OUTPUT);
  pinMode(motorLeftBackwardPin, OUTPUT);
  pinMode(motorRightForwardPin, OUTPUT);
  pinMode(motorRightBackwardPin, OUTPUT);

  pinMode(enableLeftMotor, OUTPUT);
  pinMode(enableRightMotor, OUTPUT);
  analogWrite(enableLeftMotor, 255);
  analogWrite(enableRightMotor, 255);

  pinMode(waterPumpPin, OUTPUT);
  digitalWrite(waterPumpPin, LOW);

  waterServo.attach(servoPin);
  waterServo.write(90);

  pinMode(fireSensor1Pin, INPUT);
  pinMode(fireSensor2Pin, INPUT);
  pinMode(fireSensor3Pin, INPUT);

  Serial.println("Robot␣Starting...");
  moveForward();
}

void loop() {
  bool fire1 = digitalRead(fireSensor1Pin) == LOW;
  bool fire2 = digitalRead(fireSensor2Pin) == LOW;
  bool fire3 = digitalRead(fireSensor3Pin) == LOW;

  Serial.print("Fire␣Sensor␣1:␣"); Serial.print(fire1);
  Serial.print("␣|␣Fire␣Sensor␣2:␣"); Serial.print(fire2);
  Serial.print("␣|␣Fire␣Sensor␣3:␣"); Serial.println(fire3);

  if (fire1 || fire2 || fire3) {
    fireDetected = true;
    stopMotors();

    if (fire1) {
```

```
      Serial.println("␣␣Fire␣on␣the␣LEFT!␣Turning␣Nozzle␣Left.");
      waterServo.write(150);
    } else if (fire2) {
      Serial.println("␣␣Fire␣in␣the␣CENTER!␣Keeping␣Nozzle␣Centered.");
      waterServo.write(90);
    } else if (fire3) {
      Serial.println("␣␣Fire␣on␣the␣RIGHT!␣Turning␣Nozzle␣Right.");
      waterServo.write(28);
    }

    delay(500);
    digitalWrite(waterPumpPin, HIGH);
    Serial.println("␣␣Water␣Pump␣Activated!");
  } else {
    if (fireDetected) {
      Serial.println("   ␣Fire␣Cleared!␣Stopping␣Water␣Pump.");
      digitalWrite(waterPumpPin, LOW);
      fireDetected = false;
    }

    Serial.println("No␣Fire␣Detected.␣Continuing...");
    moveForward();
  }

  delay(100);
}

void moveForward() {
  Serial.println("␣␣Moving␣Forward...");
  digitalWrite(motorLeftForwardPin, HIGH);
  digitalWrite(motorLeftBackwardPin, LOW);
  digitalWrite(motorRightForwardPin, HIGH);
  digitalWrite(motorRightBackwardPin, LOW);
}

void stopMotors() {
  Serial.println("␣␣Stopping␣Motors...");
  digitalWrite(motorLeftForwardPin, LOW);
  digitalWrite(motorLeftBackwardPin, LOW);
  digitalWrite(motorRightForwardPin, LOW);
  digitalWrite(motorRightBackwardPin, LOW);
}
```

Listing 8.1: Arduino-based Fire Fighting Robot Code

CHAPTER 9

RESULT

9.1 Output of the Working Model

The fire-fighting robot was tested in real-time conditions using a small flame (like a candle or lighter).

The system responded successfully as follows:

1. When powered ON, the robot moved forward automatically and stayed alert for fire.
2. Upon detecting fire, it stopped, rotated the nozzle toward the fire direction using the servo motor, and activated the water pump to extinguish the flame.
3. Once the fire was cleared, the water pump turned OFF and the robot resumed moving forward.
4. The serial monitor displayed sensor status and actions in real time for testing and debugging.

Summary of Output:

1. Fire Detection: Accurate on left, center, and right sides.
2. Servo Response: Quick direction adjustment based on fire location.
3. Water Pump: Activated immediately on fire detection and stopped after fire cleared.
4. Autonomous Motion: Resumed automatically after extinguishing fire.
5. The robot showed reliable performance, fast response, and successful fire extinguishing behavior.

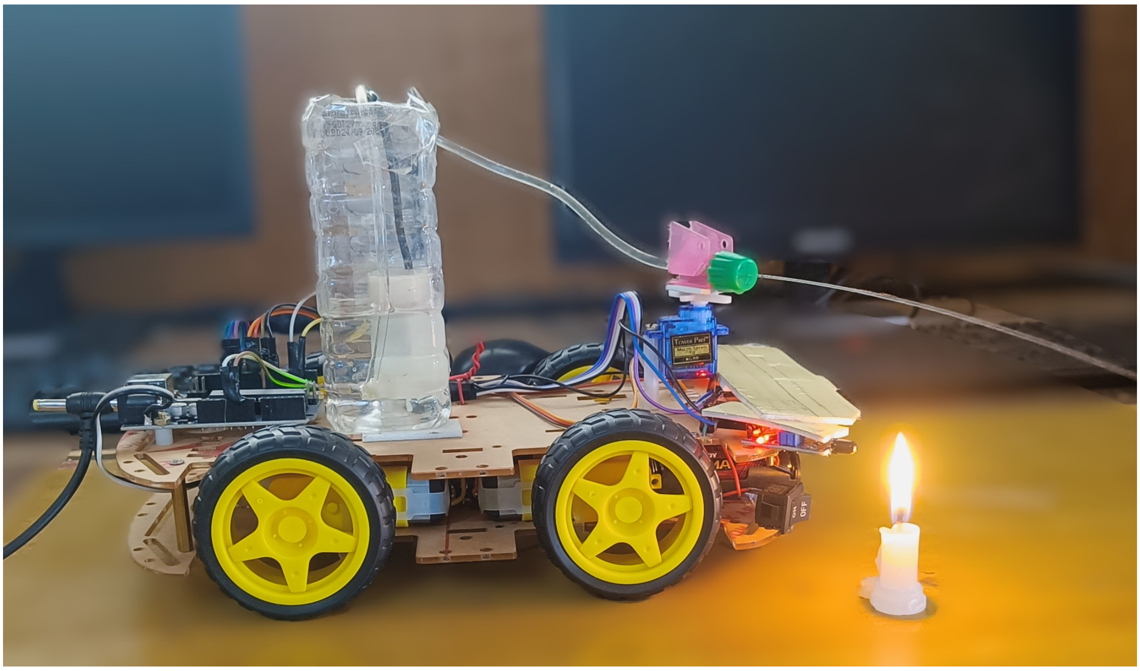


Fig. 9.1: Working Model

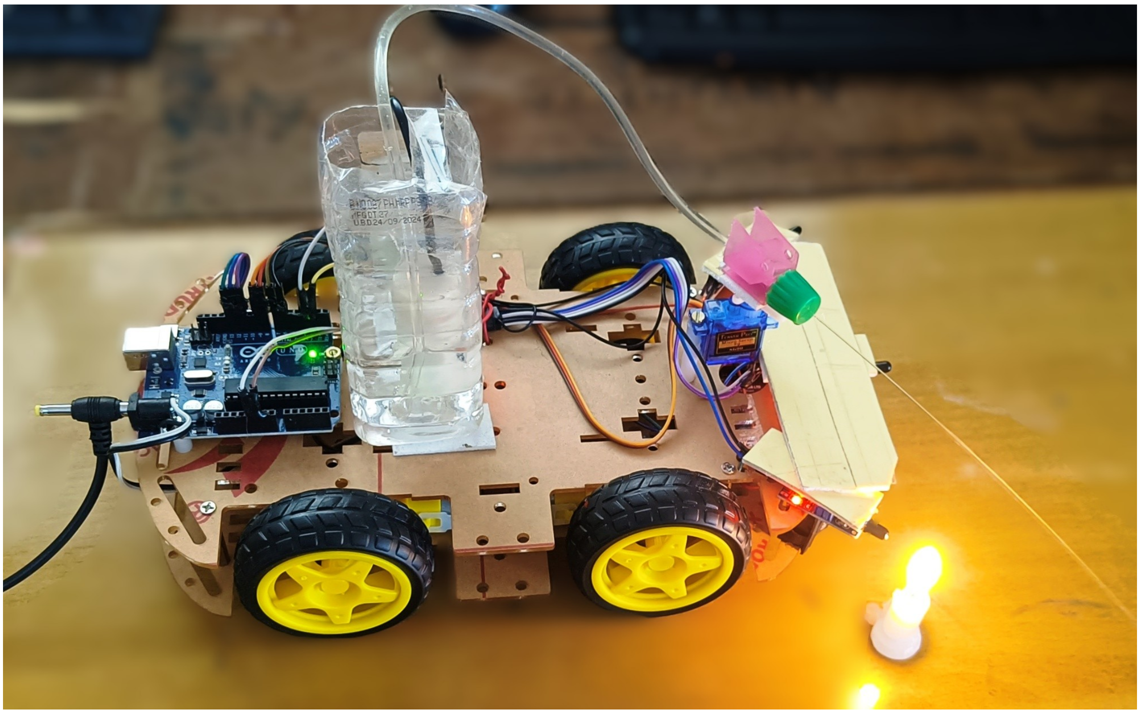


Fig. 9.2: Right Sensor Working

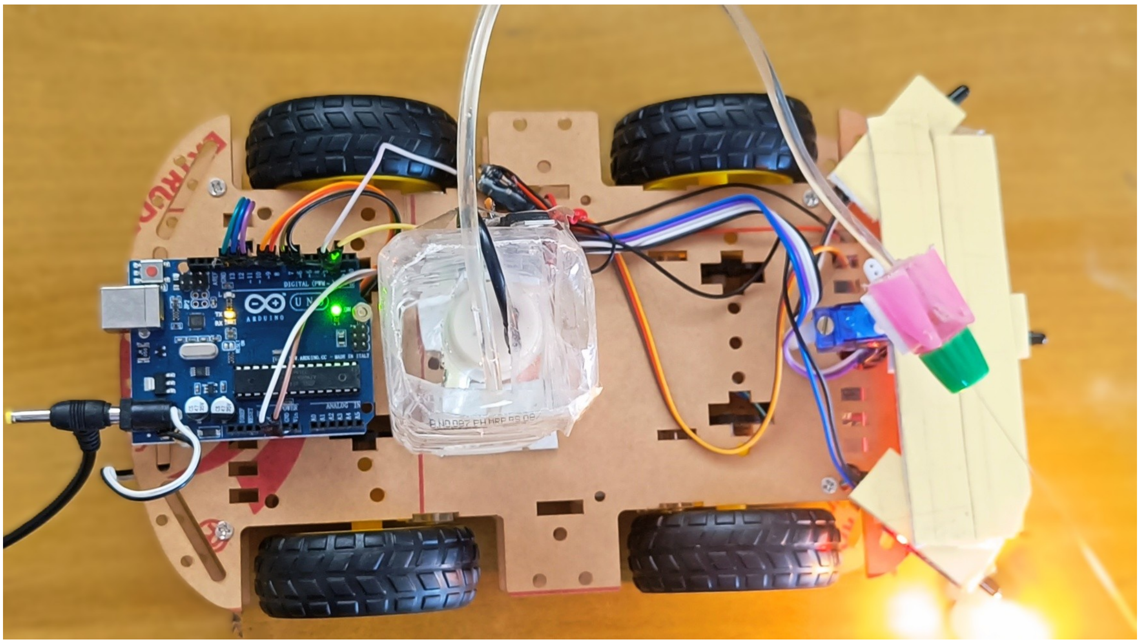


Fig. 9.3: Right Sensor Working (Top View)

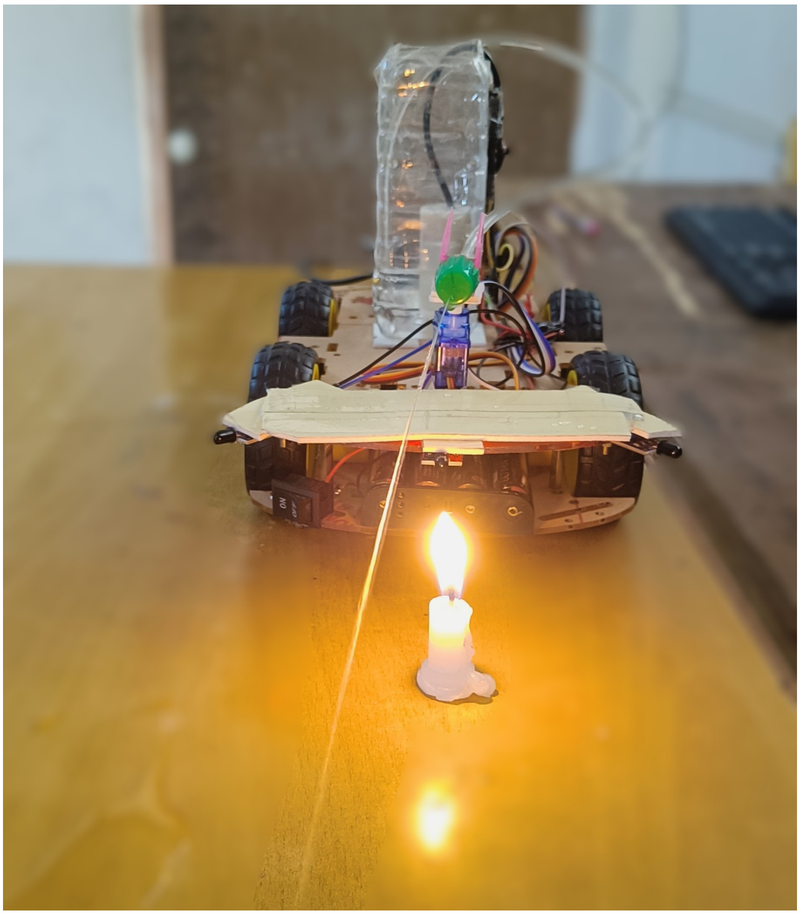


Fig. 9.4: Middle Sensor Active & Working

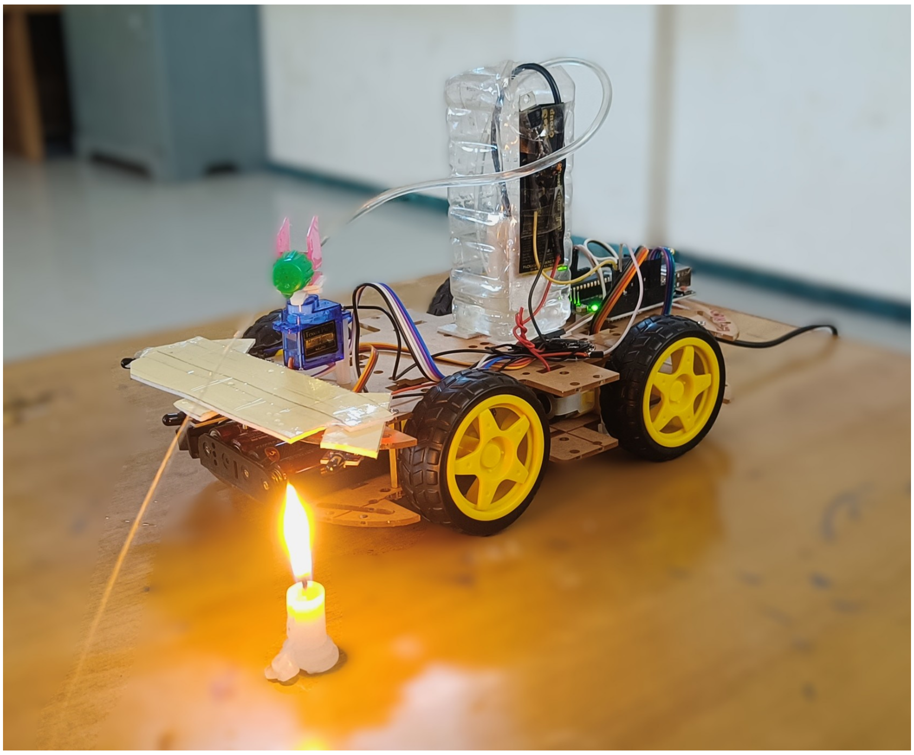


Fig. 9.5: Left Sensor Active & Working

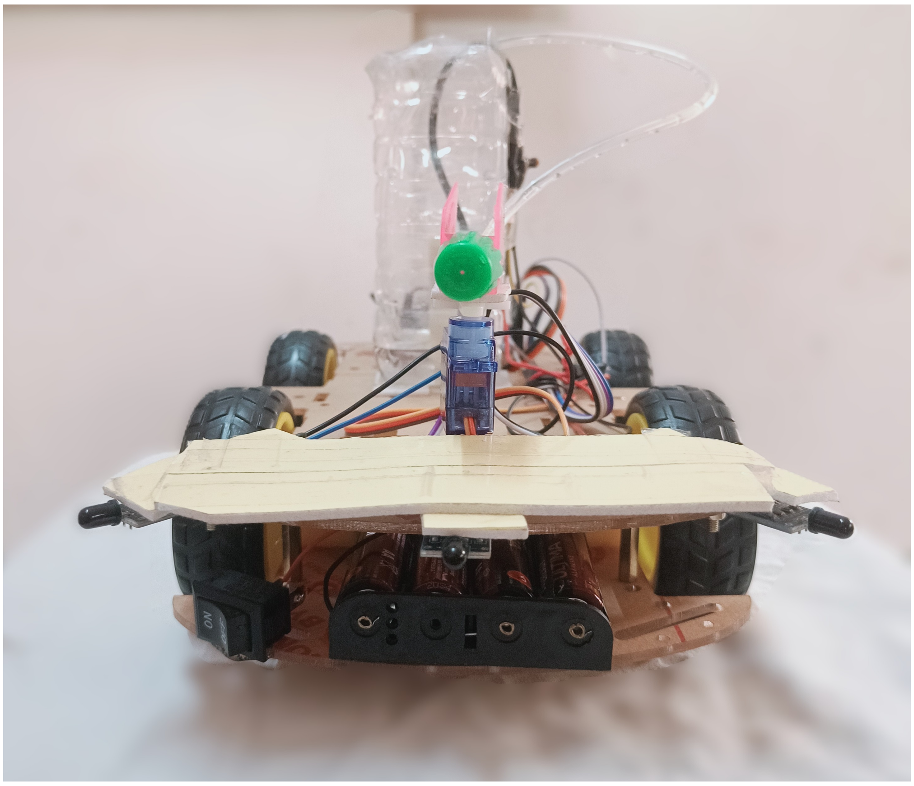


Fig. 9.6: Front View of Car with Relay of Batteries

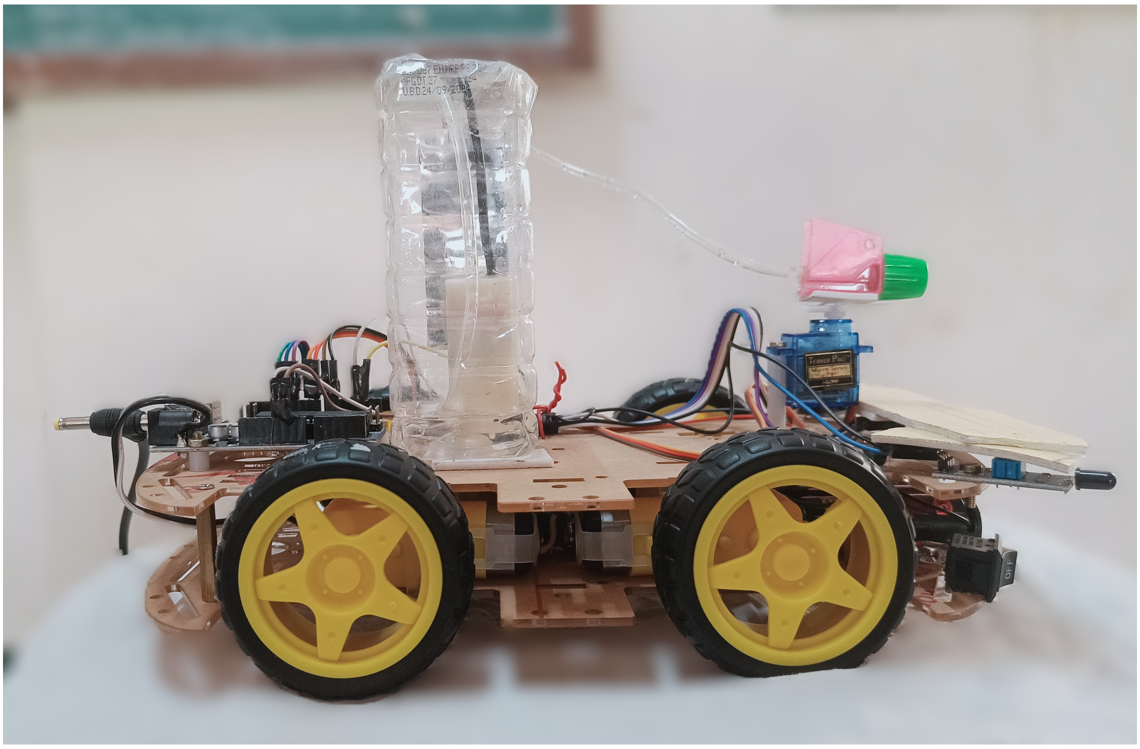


Fig. 9.7: Right Side of Car with 2 DC Motors Visible

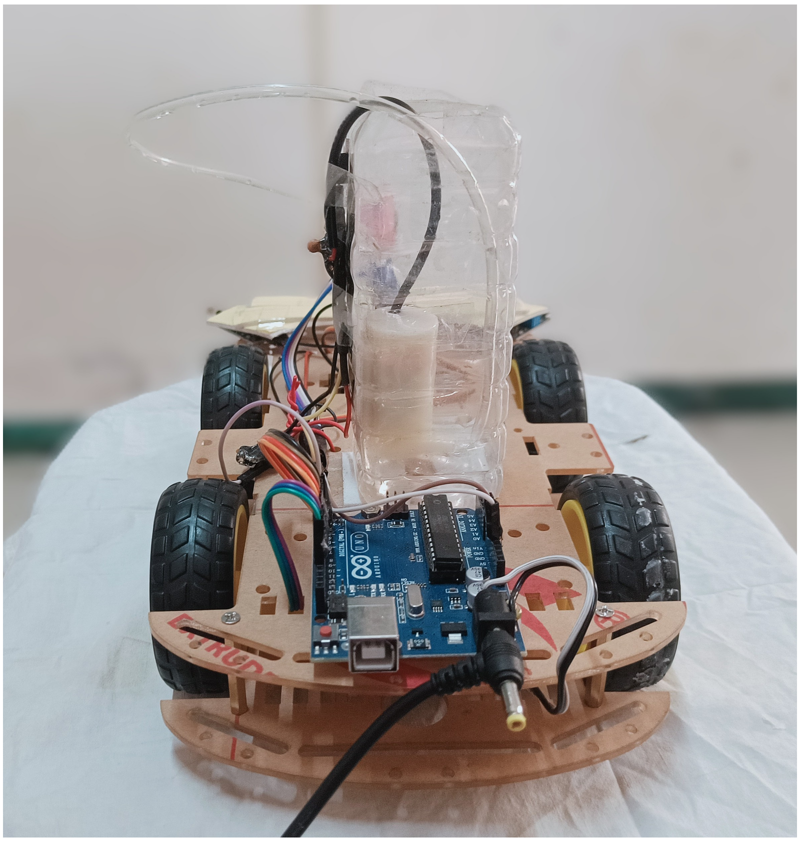


Fig. 9.8: Back Side of Car(Arduino Assembled)

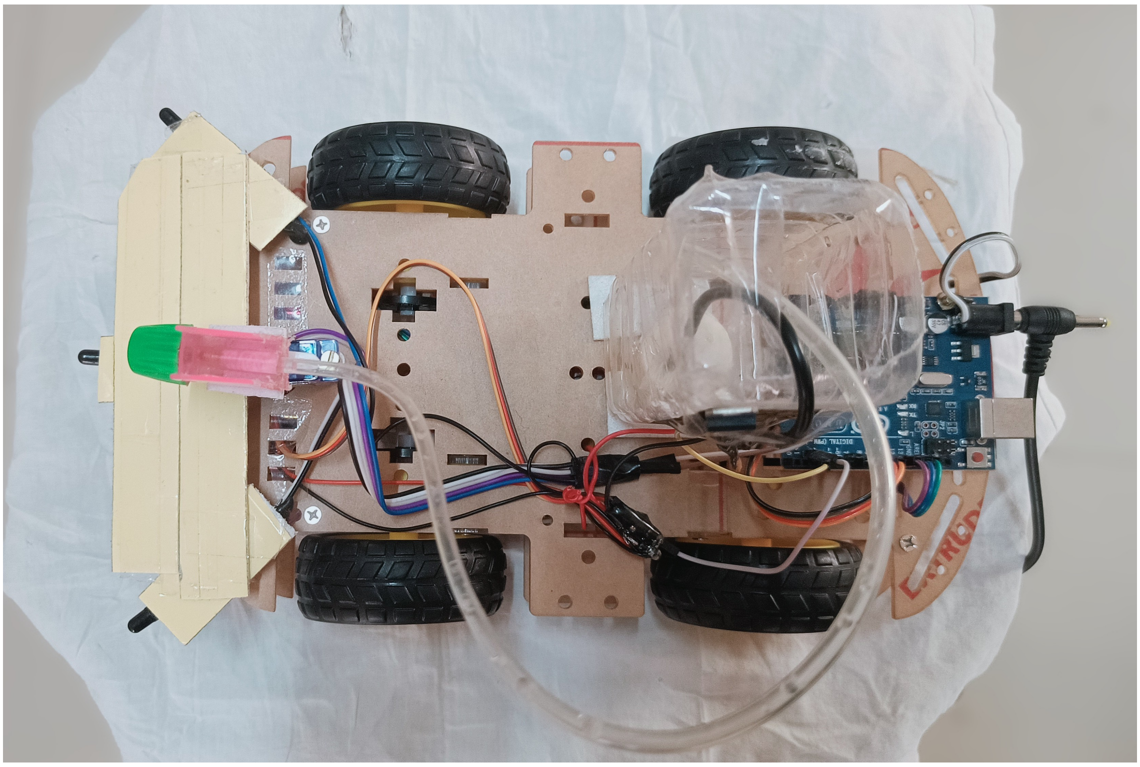


Fig. 9.9: Top View of Car with Water Pump & Tank Assembled

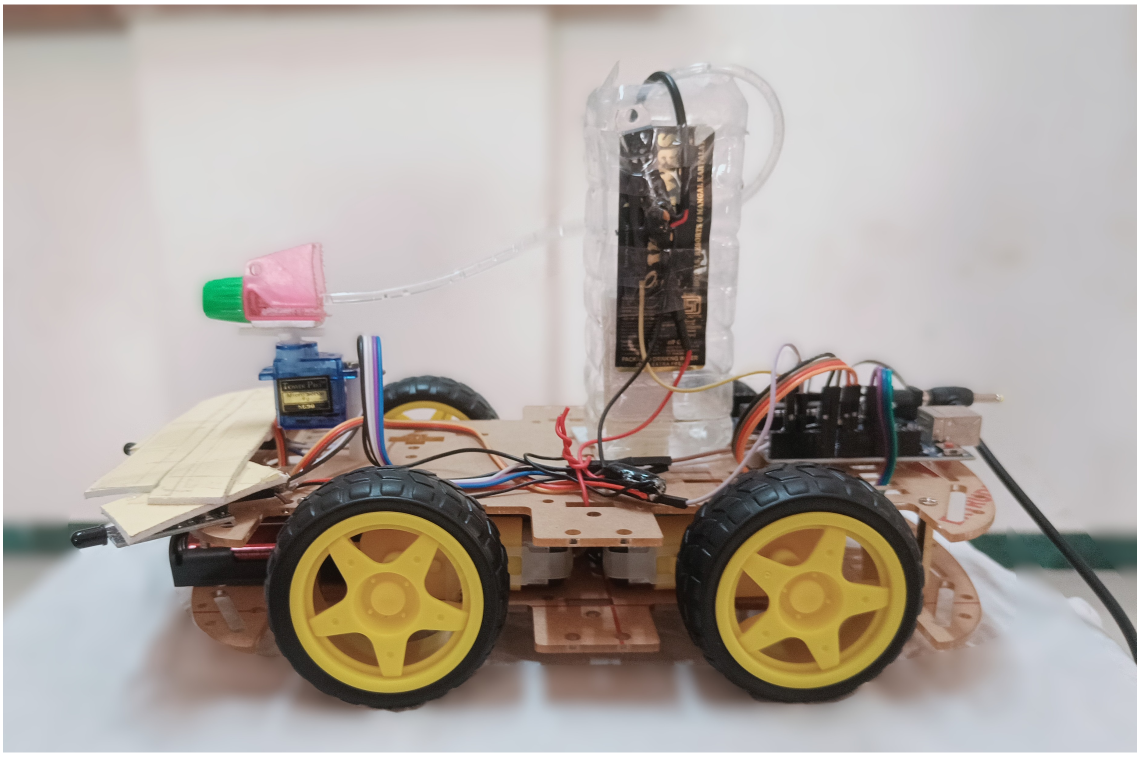


Fig. 9.10: Left Side of Car & Transistor Assembled

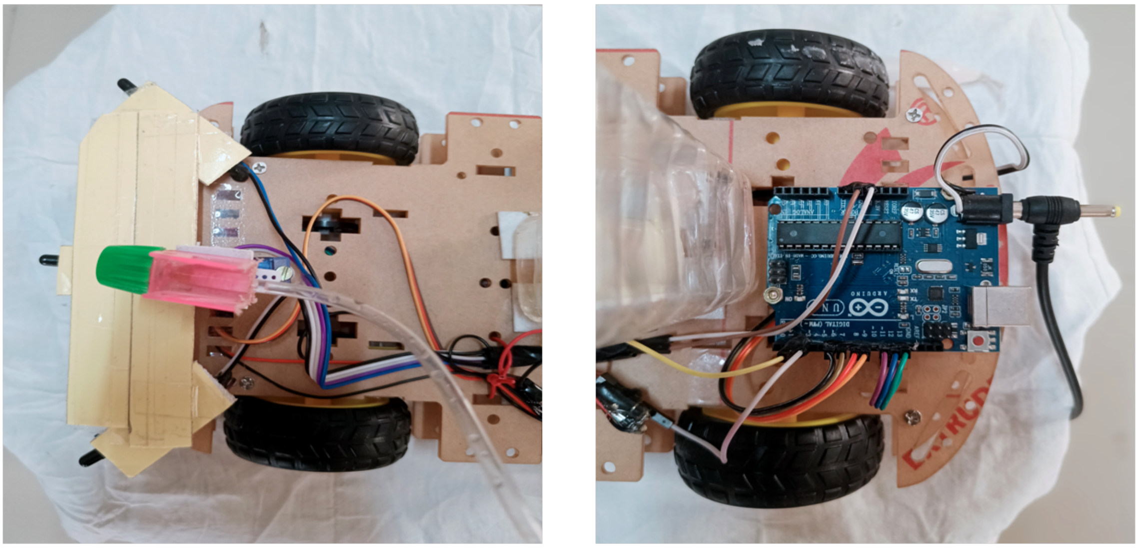


Fig. 9.11: Servo Motor and Arduino Assembled on Car

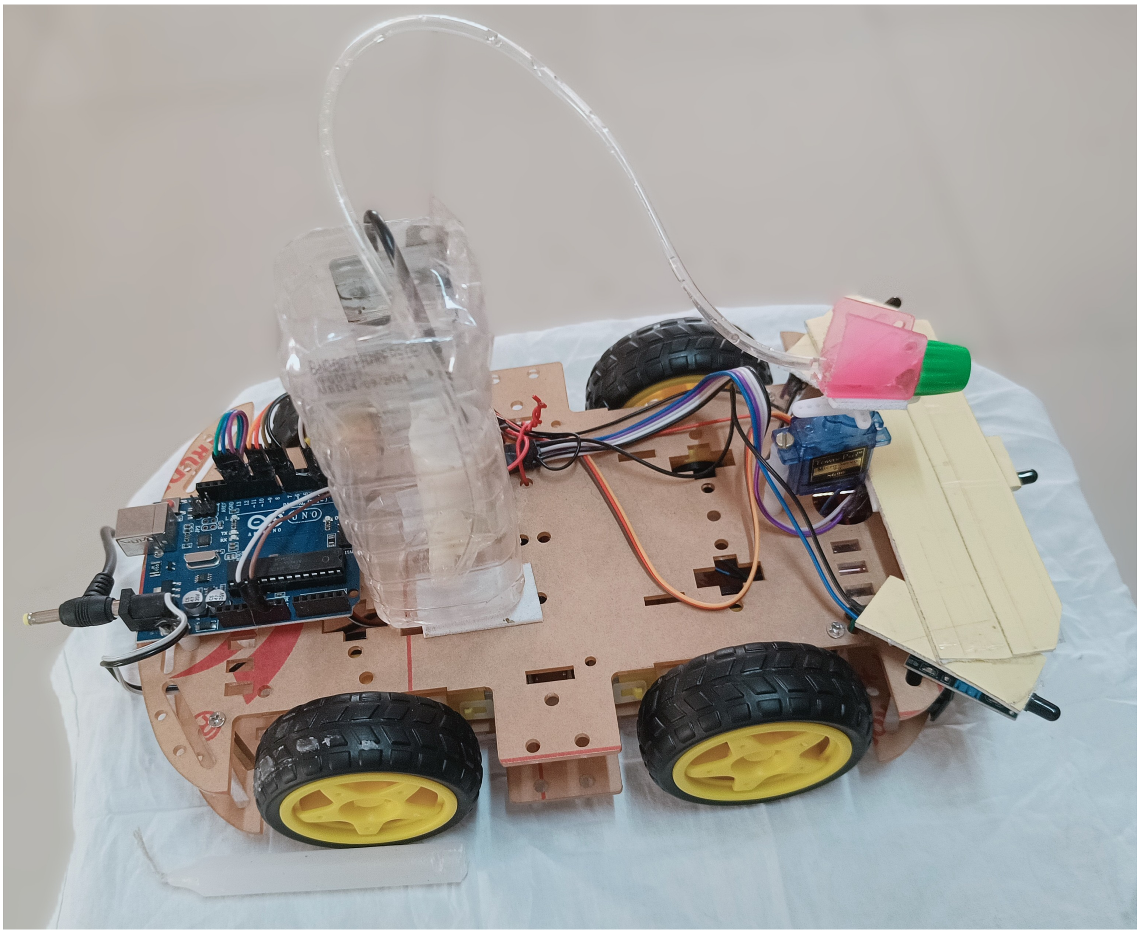


Fig. 9.12: Auto Fire Chaser & Fire Extinguisher

CHAPTER 10

CONCLUSION

The implementation of the fire-fighting robot project marks a successful blend of embedded systems, automation, and real-time environmental sensing. The system accurately detects fire using flame sensors, positions the nozzle using a servo motor, and activates the water pump to extinguish the flame, all without human control once activated on-site. This semi-automatic robot proves to be a reliable and cost-effective solution for responding to small-scale fire incidents, especially in areas where accessibility is limited. By automating a critical part of the fire response process, this project showcases how intelligent robotic systems can contribute meaningfully to public safety and emergency response. The project not only meets its technical objectives but also demonstrates a practical and impactful use of engineering knowledge in solving real-world problems.

CHAPTER 11

FUTURE WORK

While the current fire-fighting robot system performs semi-automated fire detection and extinguishing at the ground level, there is a wide scope for future development to improve efficiency, autonomy, and application areas. The project can evolve significantly with the integration of modern technologies and smarter components.

Below are the possible enhancements and future directions:

1. Integration of Ultrasonic Sensors

- **Purpose:** To detect obstacles and avoid collisions while navigating.
- **Benefit:** Ensures smoother autonomous movement without human intervention and protects the robot from damage in cluttered environments.

2. Fully Automatic Navigation using GPS

- **Purpose:** To guide the robot automatically to the fire location using GPS coordinates.
- **Benefit:** Can be deployed in large farms, industries, or outdoor areas where human access is limited. Helps in quick emergency response.

3. GSM or IoT-Based Alert System

- **Purpose:** To send alerts via SMS or cloud when fire is detected.
- **Benefit:** Allows remote monitoring and notification for immediate action, especially useful in industrial or agricultural settings.

4. Integration of Thermal Cameras or IR Sensors

- **Purpose:** To detect heat zones or flames more accurately.
- **Benefit:** Enhances the accuracy of fire detection, especially in low-light or smoky

conditions.

5. Wireless Remote Control and Mobile App Interface

- **Purpose:** To operate and monitor the robot via a smartphone or computer.
- **Benefit:** Improves usability and allows operators to control the robot from a safe distance.

6. AI-Based Decision Making

- **Purpose:** To analyze fire intensity, direction, and best path for movement.
- **Benefit:** Adds intelligence to the system and allows for faster, better response in complex fire scenarios.

7. Rechargeable Battery and Solar Charging Option

- **Purpose:** For uninterrupted and eco-friendly power supply.
- **Benefit:** Makes the robot energy-efficient and suitable for long-duration use in outdoor environments.

CHAPTER 12

REFERENCES

Key References and Resources Utilized in the Development of the Fire-Fighting Robot:

(a) Arduino Uno Pin Configuration and Microcontroller Basics –
www.robokits.co.in

(b) Analysis of components and pricing –
Maxim Electronics, Shalimar Electronics, and Hemant Electronics (Pandit Colony, Nashik, Maharashtra)

(c) Comprehensive tutorial covering assembly and code explanation –
www.youtube.com

(d) Understanding of Circuit Design and Flame Sensor Connection –
www.circuitdigest.com

(e) Coding Logic Reference and Explanation –
ChatGPT (OpenAI), Project Guide (Faculty Assistance)

(f) Servo Motor Working and PWM Control –
www.engineersgarage.com

(g) L298N Motor Driver Working and Connection –
www.electronicsforu.com

(h) Relay Module and Pump Activation Circuit Reference –
www.electronicshub.org

(i) LCD Display Commands and Pin Setup –
www.electronicwings.com

(j) Assembly Tools and Chassis Material –
Purchased locally from Nashik Electronics and Hardware Market

(k) Embedded System Basics and Application in Robotics –
www.techopedia.com

(l) Project Report Formatting Help and Circuit Editing –
Guidance by Project Guide & Mentors

(m) Flame Sensor Working Principle –
www.sciencedirect.com

(n) Submersible Pump Motor Usage in Embedded Projects –
www.hackster.io

IT QUEST 2K25

K K WAGH EDUCATION SOCIETY'S
K K WAGH POLYTECHNIC, NASHIK
DEPARTMENT OF INFORMATION TECHNOLOGY
(ACCREDITED BY NBA)

CERTIFICATE

THIS IS TO CERTIFY THAT

Mr./Ms. Prasad Nikam

From Gurukul Polytechnic, Nandgaon

has been awarded the 1st / 2nd / or has participated in Paper Presentation / Blind Coding / Project Competition / Poster Competition Under State Level Technical Event "IT Quest 2K25" held at K K Wagh Polytechnic, Nashik on 25th March 2025.

PROF. M. P. NAWARKAR
CO-ORDINATIOR

PROF. M. S. KARANDE
HEAD OF DEPT

PROF. P. T. KADAVE
PRINCIPAL

IT QUEST 2K25

K K WAGH EDUCATION SOCIETY'S
K K WAGH POLYTECHNIC, NASHIK
DEPARTMENT OF INFORMATION TECHNOLOGY
(ACCREDITED BY NBA)

CERTIFICATE

THIS IS TO CERTIFY THAT

Mr./Ms. Yash Jagdhane

From Gurukul Polytechnic, Nandgaon

has been awarded the 1st / 2nd / or has participated in Paper Presentation / Blind Coding / Project Competition / Poster Competition Under State Level Technical Event "IT Quest 2K25" held at K K Wagh Polytechnic, Nashik on 25th March 2025.

PROF. M. P. NAWARKAR
CO-ORDINATIOR

PROF. M. S. KARANDE
HEAD OF DEPT

PROF. P. T. KADAVE
PRINCIPAL

IT QUEST 2K25

K K WAGH EDUCATION SOCIETY'S
K K WAGH POLYTECHNIC, NASHIK
DEPARTMENT OF INFORMATION TECHNOLOGY
(ACCREDITED BY NBA)

CERTIFICATE

THIS IS TO CERTIFY THAT

Mr./Ms...Shubham...Ahire......

From ...Gurukul...Polytechnic...,Nandgaon......

has been awarded the 1st / 2nd / or has participated in Paper Presentation / Blind C coding / Project Competition / Poster Competition Under State Level Technical Event "IT Quest 2K25" held at K K Wagh Polytechnic, Nashik on 25th March 2025.

PROF. M. P. NAWARKAR, CO-ORDINATIOR
PROF. M. S. KARANDE, HEAD OF DEPT
PROF. P. T. KADAVE, PRINCIPAL

IT QUEST 2K25

K K WAGH EDUCATION SOCIETY'S
K K WAGH POLYTECHNIC, NASHIK
DEPARTMENT OF INFORMATION TECHNOLOGY
(ACCREDITED BY NBA)

CERTIFICATE

THIS IS TO CERTIFY THAT

Mr./Ms...Aaditya...Nilange......

From ...Gurukul...Polytechnic...,Nandgaon......

has been awarded the 1st / 2nd / or has participated in Paper Presentation / Blind C coding / Project Competition / Poster Competition Under State Level Technical Event "IT Quest 2K25" held at K K Wagh Polytechnic, Nashik on 25th March 2025.

PROF. M. P. NAWARKAR, CO-ORDINATIOR
PROF. M. S. KARANDE, HEAD OF DEPT
PROF. P. T. KADAVE, PRINCIPAL

Krantiveer Vasantrao Narayanrao Naik Shikshan Prasarak Sanstha's

LOKNETE GOPINATHJI MUNDE INSTITUTE OF ENGINEERING EDUCATION & RESEARCH, NASHIK

(Degree, Diploma Engineering & MBA)

Krantiveer Vasantrao Naik Marg, Canada Corner, Sharanpur Road, Nashik.

NAAC ACCREDITED

Certificate

This is to certify that

Mr./Ms. Prasad Nikam from Gurukul , poly Nandgaon has participated / worked as Volunteer in Project Competition State level competition during "LoGMIEER Techfest 2k25' organized by Loknete Gopinathji Munde Institute of Engineering Education and Research on 1st March, 2025.

Prof. H. P. Bhabad (Event Co-ordinator) | Prof. T. Y. Saindane (Event Co-ordinator) | Dr. K. V. Chandratre Principal

Krantiveer Vasantrao Narayanrao Naik Shikshan Prasarak Sanstha's

LOKNETE GOPINATHJI MUNDE INSTITUTE OF ENGINEERING EDUCATION & RESEARCH, NASHIK

(Degree, Diploma Engineering & MBA)

Krantiveer Vasantrao Naik Marg, Canada Corner, Sharanpur Road, Nashik.

NAAC ACCREDITED

Certificate

This is to certify that

Mr./Ms. Yash Jagdhane from Gurukul poly, Nandgaon has participated / worked as Volunteer in Project Competition State level competition during "LoGMIEER Techfest 2k25' organized by Loknete Gopinathji Munde Institute of Engineering Education and Research on 1st March, 2025.

Prof. H. P. Bhabad (Event Co-ordinator) | Prof. T. Y. Saindane (Event Co-ordinator) | Dr. K. V. Chandratre Principal

Krantiveer Vasantrao Narayanrao Naik Shikshan Prasarak Sanstha's

LOKNETE GOPINATHJI MUNDE INSTITUTE OF ENGINEERING EDUCATION & RESEARCH, NASHIK

(Degree, Diploma Engineering & MBA)

Krantiveer Vasantrao Naik Marg, Canada Corner, Sharanpur Road, Nashik.

NAAC ACCREDITED

Certificate

This is to certify that

Mr./Ms. Shubham Ahire from Gurukul poly, Nandgaon has participated / worked as Volunteer in Project Competition

State level competition during "LoGMIEER Techfest 2k25" organized by Loknete Gopinathji Munde Institute of Engineering Education and Research on 1st March, 2025.

Prof. H. P. Bhabad (Event Co-ordinator) | Prof. T. Y. Saindane (Event Co-ordinator) | Dr. K. V. Chandratre Principal

Krantiveer Vasantrao Narayanrao Naik Shikshan Prasarak Sanstha's

LOKNETE GOPINATHJI MUNDE INSTITUTE OF ENGINEERING EDUCATION & RESEARCH, NASHIK

(Degree, Diploma Engineering & MBA)

Krantiveer Vasantrao Naik Marg, Canada Corner, Sharanpur Road, Nashik.

NAAC ACCREDITED

Certificate

This is to certify that

Mr./Ms. Aditya Nilange from Gurukul poly, Nandgon has participated / worked as Volunteer in Project Competition

State level competition during "LoGMIEER Techfest 2k25" organized by Loknete Gopinathji Munde Institute of Engineering Education and Research on 1st March, 2025.

Prof. H. P. Bhabad (Event Co-ordinator) | Prof. T. Y. Saindane (Event Co-ordinator) | Dr. K. V. Chandratre Principal


Sanjivani K. B. P. Polytechnic, Kopargaon
(with NBA Accredited Programmes, Graded Excellent By MSBTE)
INDIAN SOCIETY FOR TECHNICAL EDUCATION STUDENTS' CHAPTER
AND INSTITUTE OF ENGINEERS (INDIA) STUDENTS' CHAPTER
National Level Technical Event
SANJIVANI TECH-MANTRA
Certificate

This is to certify that, Mr./ M/ss. Prasad Narendra Nikam
studying in class TY of Institute Gurukul Polytechnic, Nandgaon
has participated in Project competition of
National Level Event - Sanjivani Tech-Mantra organized on 18 / 03 /2025 . We
congratulate him/her for securing Winner / Runner-up / Consolation / Participation
/ Volunteer position.

Event Co-ordinator | Co-coordinator | Co-ordinator | Principal


Sanjivani K. B. P. Polytechnic, Kopargaon
(with NBA Accredited Programmes, Graded Excellent By MSBTE)
INDIAN SOCIETY FOR TECHNICAL EDUCATION STUDENTS' CHAPTER
AND INSTITUTE OF ENGINEERS (INDIA) STUDENTS' CHAPTER
National Level Technical Event
SANJIVANI TECH-MANTRA
Certificate

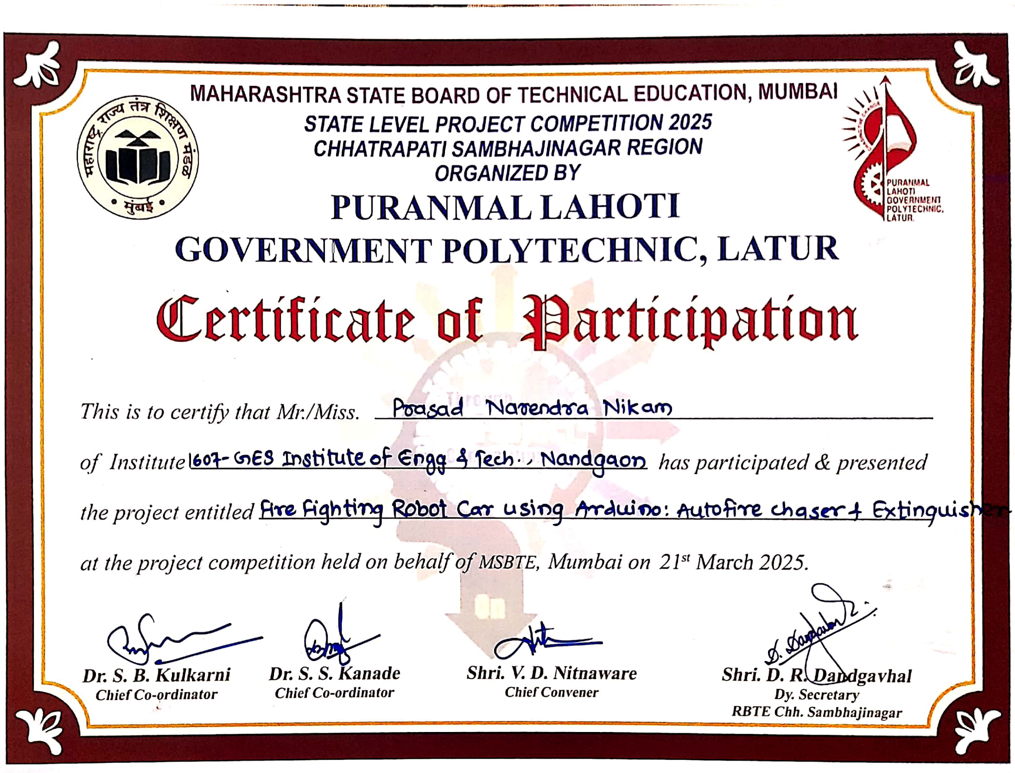
This is to certify that, Mr./ M/ss. Yash Harishchandra Jagdhane
studying in class TY of Institute Gurukul Polytechic, Nandgaon
has participated in Project competition of
National Level Event - Sanjivani Tech-Mantra organized on 18 / 03 /2025 . We
congratulate him/her for securing Winner / Runner-up / Consolation / Participation
/ Volunteer position.

Event Co-ordinator | Co-coordinator | Co-ordinator | Principal

Sanjivani K. B. P. Polytechnic, Kopargaon
(with NBA Accredited Programmes, Graded Excellent By MSBTE)
INDIAN SOCIETY FOR TECHNICAL EDUCATION STUDENTS' CHAPTER
AND INSTITUTE OF ENGINEERS (INDIA) STUDENTS' CHAPTER
National Level Technical Event
SANJIVANI TECH-MANTRA
Certificate

This is to certify that, Mr./ Miss. Aditya Rajendra Nilange studying in class TY of Institute Gurukul Polytechnic, Nandgaon has participated in Project competition of National Level Event - Sanjivani Tech-Mantra organized on 18 / 03 /2025 . We congratulate him/her for securing Winner / Runner-up / Consolation / Participation / Volunteer position.

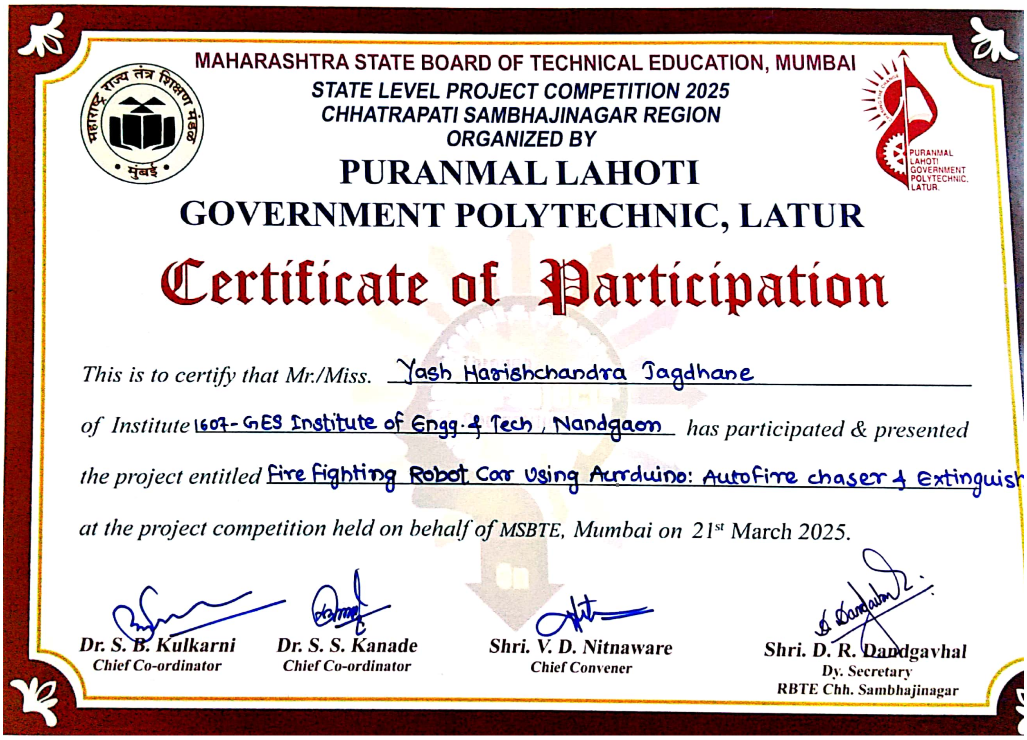
Event Co-ordinator | Co-coordinator | Co-ordinator | Principal



MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION, MUMBAI
STATE LEVEL PROJECT COMPETITION 2025
CHHATRAPATI SAMBHAJINAGAR REGION
ORGANIZED BY
PURANMAL LAHOTI
GOVERNMENT POLYTECHNIC, LATUR
Certificate of Participation

This is to certify that Mr./Miss. Prasad Narendra Nikam of Institute 1607- GES Institute of Engg & Tech., Nandgaon has participated & presented the project entitled Fire Fighting Robot Car using Arduino: Autofire chaser & Extinguisher at the project competition held on behalf of MSBTE, Mumbai on 21st March 2025.

Dr. S. B. Kulkarni, Chief Co-ordinator
Dr. S. S. Kanade, Chief Co-ordinator
Shri. V. D. Nitnaware, Chief Convener
Shri. D. R. Dandgavhal, Dy. Secretary, RBTE Chh. Sambhajinagar

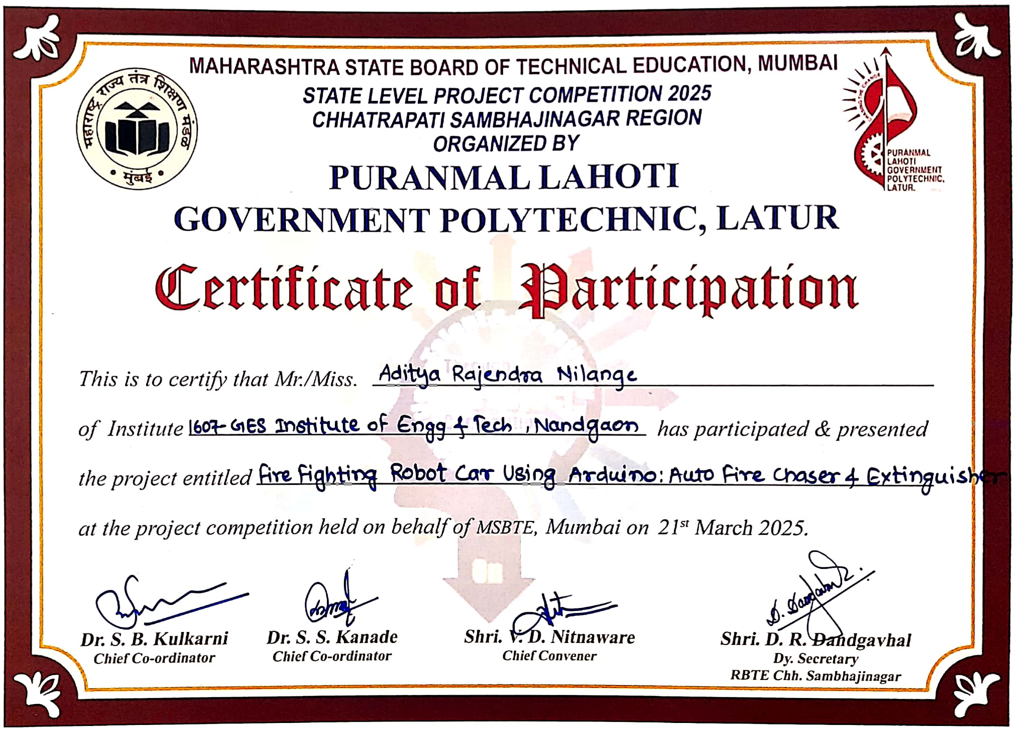


MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION, MUMBAI
STATE LEVEL PROJECT COMPETITION 2025
CHHATRAPATI SAMBHAJINAGAR REGION
ORGANIZED BY
PURANMAL LAHOTI
GOVERNMENT POLYTECHNIC, LATUR

Certificate of Participation

This is to certify that Mr./Miss. Yash Harishchandra Jagdhane
of Institute 1607- GES Institute of Engg. & Tech, Nandgaon has participated & presented
the project entitled Fire fighting Robot Car Using Aurduino: Autofire chaser & Extinguish
at the project competition held on behalf of MSBTE, Mumbai on 21st March 2025.

Dr. S. B. Kulkarni, Chief Co-ordinator
Dr. S. S. Kanade, Chief Co-ordinator
Shri. V. D. Nitnaware, Chief Convener
Shri. D. R. Dandgavhal, Dy. Secretary, RBTE Chh. Sambhajinagar



MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION, MUMBAI
STATE LEVEL PROJECT COMPETITION 2025
CHHATRAPATI SAMBHAJINAGAR REGION
ORGANIZED BY
PURANMAL LAHOTI
GOVERNMENT POLYTECHNIC, LATUR

Certificate of Participation

This is to certify that Mr./Miss. Aditya Rajendra Nilange
of Institute 1607- GES Institute of Engg & Tech, Nandgaon has participated & presented
the project entitled Fire fighting Robot Car Using Arduino: Auto Fire Chaser & Extinguisher
at the project competition held on behalf of MSBTE, Mumbai on 21st March 2025.

Dr. S. B. Kulkarni, Chief Co-ordinator
Dr. S. S. Kanade, Chief Co-ordinator
Shri. V. D. Nitnaware, Chief Convener
Shri. D. R. Dandgavhal, Dy. Secretary, RBTE Chh. Sambhajinagar