SMART SENSING ROVER

MINI PROJECT REPORT

Submitted by

SOWMYA R	2116230701328
SOUNDARYALAKSHMI S	2116230701326

In partial fulfillment for the award of the degree

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE

ANNA UNIVERSITY: CHENNAI 600 025

MAY 2025

BONAFIDE CERTIFICATE

Certified that this project **“SMART SENSING ROVER**” is the bonafide work of “**SOWMYA R (2116230701328) and SOUNDARYALAKSHMI S (2116230701326)”** who carried out the project work under my supervision.

SIGNATURE
Dr.N.Duraimurugan, M.Tech., Ph.D.

Associate Professor,

Computer Science & Engineering

Rajalakshmi Engineering College
(Autonomous)
Thandalam, Chennai -602105.

Submitted for the **ANNA UNIVERSITY** practical examination Mini-Project work viva voice held on__________________

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

Initially we thank the Almighty for being with us through every walk of our life and showering his blessings through the endeavor to put forth this report. Our sincere thanks to our Chairman **Mr. S.MEGANATHAN, B.E, F.I.E.,** our Vice Chairman **Mr. ABHAY SHANKAR MEGANATHAN, B.E., M.S.,** and our respected Chairperson **Dr. (Mrs.) THANGAM MEGANATHAN, Ph.D.,** for providing us with the requisite infrastructure and sincere endeavoring in educating us in their premier institution.

Our sincere thanks to **Dr. S.N. MURUGESAN, M.E., Ph.D.,** our beloved Principal for his kind support and facilities provided to complete our work in time.

We express our sincere thanks to **Dr. P. KUMAR, M.E., Ph.D.,** Professor and Head of the Department of Computer Science and Engineering for his guidance and encouragement throughout the project work.

We also extend our sincere and hearty thanks to our Internal Guide **Dr.N.Duraimurugan, M.Tech., Ph.D.** Associate Professor, Department of Computer Science and Engineering for his valuable guidance and motivation during the completion of this project. Our sincere thanks to our family members, friends and other staff members of information technology.

SOWMYA R	**2116230701328**
SOUNDARYALAKSHMI S	**2116230701326**

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATION

ABBREVIATION	ACRONYM
IOT	Internet of Things
IR	Infrared
ESP32	Espresso System Processor 32

ABSTRACT

In disaster-prone or hazardous environments, timely and safe search-and-rescue operations are critical but often risky for human responders. This project presents an **IoT-based autonomous rescue rover** designed to assist in locating trapped or injured individuals without direct human intervention. The system is built around the **ESP32 microcontroller**, which serves as the core controller and communication hub, thanks to its integrated Wi-Fi capability.

The rover is equipped with essential sensors: an **ultrasonic sensor** for obstacle detection and avoidance, an **IR sensor** for proximity and edge detection, and a **sound detector** to identify potential victim signals such as cries for help. These sensors enable the rover to navigate autonomously through debris or confined areas while actively scanning for human presence.

Collected sensor data is processed in real-time by the ESP32 and can be transmitted to a remote **web-based dashboard** for visualization and monitoring. The system includes an **alerting mechanism** that notifies operators immediately when sound signals indicative of a victim are detected. This enhances the responsiveness of rescue teams and reduces risk to human life.

The rover's modular design, wireless communication, and autonomous navigation make it a valuable tool for efficient and safer search-and-rescue operations in inaccessible or life-threatening zones.

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

Search-and-rescue operations in disaster-affected areas often involve significant risk to human responders, especially in environments with unstable structures or limited visibility. To address this challenge, this project introduces an IoT-based Rescue Rover that can autonomously navigate and assist in locating victims without endangering rescue personnel.

The rover is powered by the ESP32 microcontroller, which integrates Wi-Fi connectivity and sufficient processing power to handle real-time sensor data and communication. It is equipped with an ultrasonic sensor for obstacle detection, an IR sensor for proximity or edge detection, and a sound detector to identify potential human distress signals such as cries or knocks.

As the rover navigates, it continuously monitors its surroundings and responds to sensor inputs. Detected information is processed and transmitted to a web-based interface, allowing operators to track the rover's movements and receive alerts when possible victims are found. This approach enhances the speed, safety, and effectiveness of rescue missions, making it a valuable tool in emergency response scenarios.

1.2 SCOPE OF THE WORK

This project focuses on developing an **IoT-based autonomous rescue rover** capable of assisting in search-and-rescue operations, particularly in disaster-affected or hazardous environments. By utilizing the **ESP32 microcontroller** along with sensors such as **ultrasonic**, **IR**, and **sound detectors**, the rover is designed to navigate through debris or confined spaces, detect potential victims, and communicate real-time data to a remote operator.

1.3 PROBLEM STATEMENT

Rescue operations in disaster zones are often delayed and dangerous due to limited access and high risk to human responders. There is a need for an autonomous system that can safely navigate hazardous areas, detect trapped victims, and share real-time information. This project aims to solve this problem by developing an IoT-based rescue rover using ESP32 and sensors to improve the speed and safety of search-and-rescue missions.

1.4 AIM AND OBJECTIVES OF THE PROJECT

The aim of this project is to design and develop an **IoT-based autonomous rescue rover** using the **ESP32 microcontroller** to assist in search-and-rescue operations by navigating hazardous environments, detecting potential victims, and providing real-time data to rescue teams. The rover will integrate sensors such as **ultrasonic**, **IR**, and **sound detectors** to enable obstacle avoidance, victim detection, and autonomous movement. The project aims to establish a communication system that allows the rover to transmit data and alerts to a centralized web interface, facilitating remote monitoring and control by rescue operators. Ultimately, the goal is to improve the efficiency, speed, and safety of rescue operations through an intelligent, autonomous system that responds in real-time to distress signals and environmental conditions.

CHAPTER 2

SYSTEM SPECIFICATIONS

2.1 IOT DEVICES

1. Sound Sensor
2. IR Sensor
3. Ultrasonic Sensor

2.2 SYSTEM HARDWARE SPECIFICATIONS

PROCESSOR	Intel i7 13th Gen
MEMORY SIZE	8 GB (Minimum)
HDD	40 GB (Minimum)

2.3 SOFTWARE SPECIFICATIONS

Operating System	Windows 11
Front – End	React JS
Back – End	PostgreSQL, Express and Node JS
Browser	Google Chrome
IDE	Visual Studio Code

CHAPTER 3
SYSTEM DESIGN

3.1 ARCHITECTURE DIAGRAM

An architecture diagram is a graphical representation of a set of concepts, that are part of an architecture, including their principles, elements and components

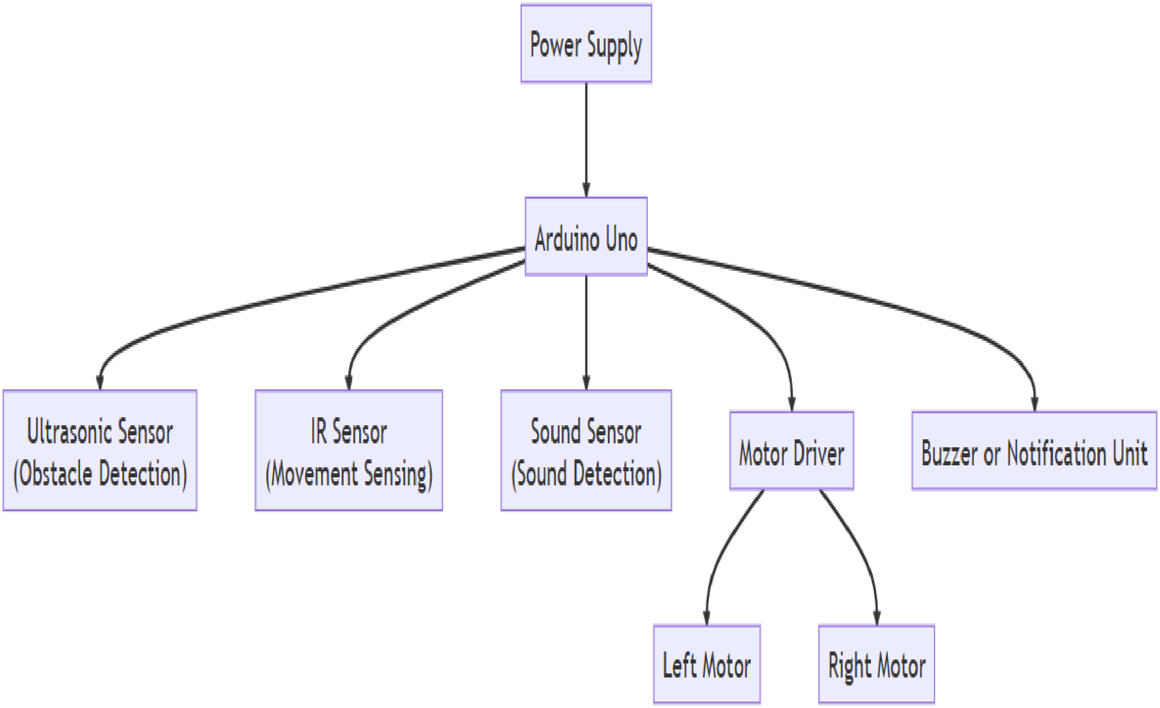

Power Supply
Arduino Uno
Ultrasonic Sensor (Obstacle Detection)
IR Sensor (Movement Sensing)
Sound Sensor (Sound Detection)
Motor Driver
Buzzer or Notification Unit
Left Motor
Right Motor

Figure 3.1 Architecture Diagram

From the above Figure 3.1, the architecture of the system is well understood.

3.2 USE CASE DIAGRAM

A use case is a list of actions or event steps typically defining the interactions between a role (known in the Unified Modelling Language as an actor) and a system to achieve a goal. The actor can be a human or other external system.

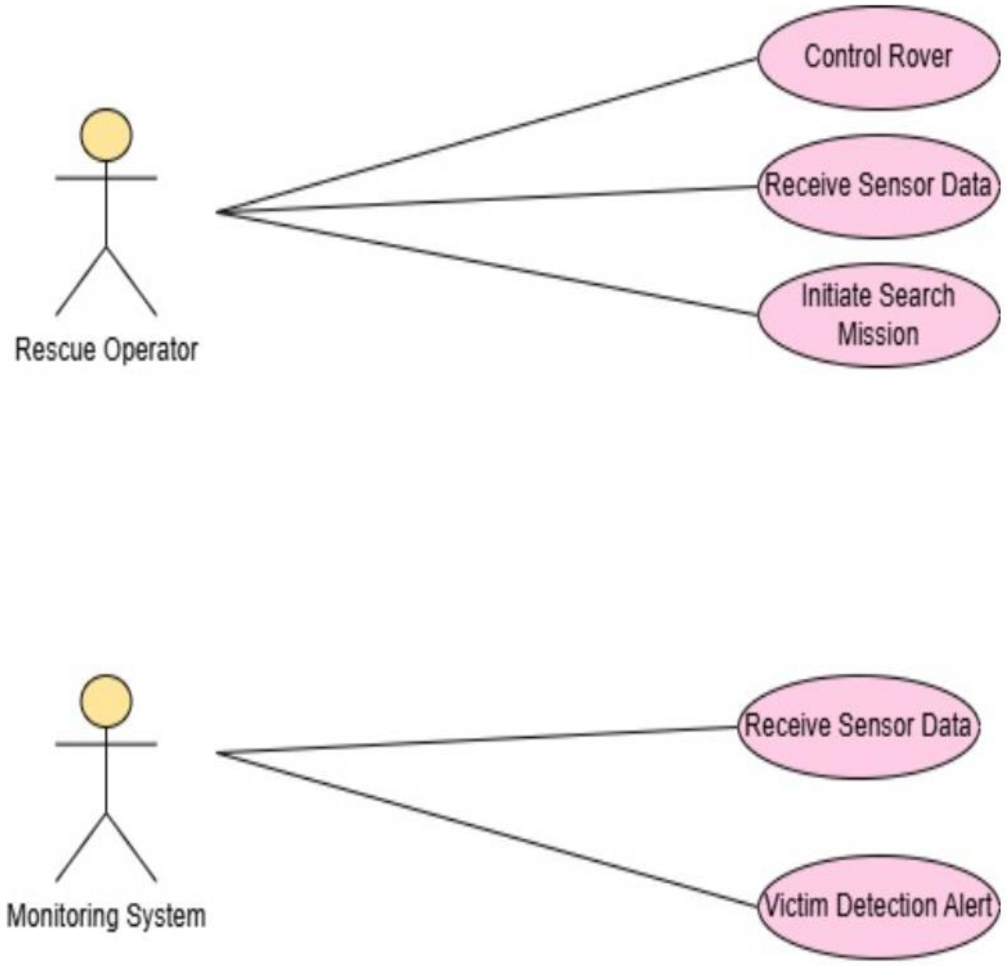


Figure 3.2 Use case diagram

From the above figure 3.2, the interactions between a role in the system is shown

3.3 ACTIVITY DIAGRAM

An activity in Unified Modelling Language (UML) is a major task that must take place in order to fulfill an operation contract. Activities can be represented inactivity diagrams. An activity can represent: The invocation of an operation. A step in a business process.

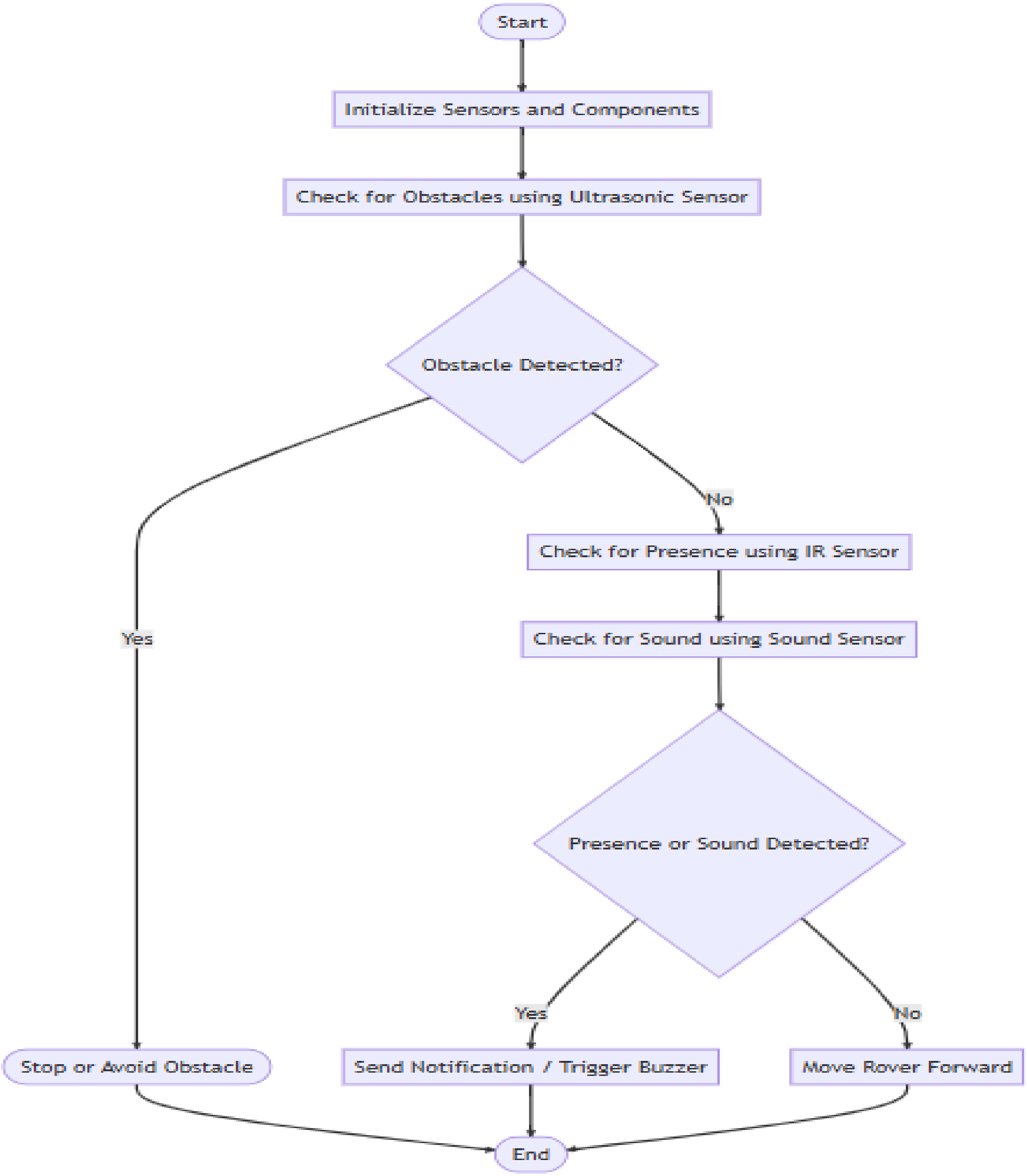

Start
Initialize Sensors and Components
Check for Obstacles using Ultrasonic Sensor
Obstacle Detected?
No
Check for Presence using IR Sensor
Yes
Check for Sound using Sound Sensor
Presence or Sound Detected?
Yes
No
Stop or Avoid Obstacle
Send Notification / Trigger Buzzer
Move Rover Forward
End

Figure 3.3 Activity Diagram

From the above figure 3.3, the activities of the system are shown

3.4 CLASS DIAGRAM

A class diagram is an illustration of the relationships and source code dependencies among classes in the Unified Modelling Language (UML). In this context, a class defines the methods and variables in an object, which is a specific entity in a program or the unit of code representing that entity.

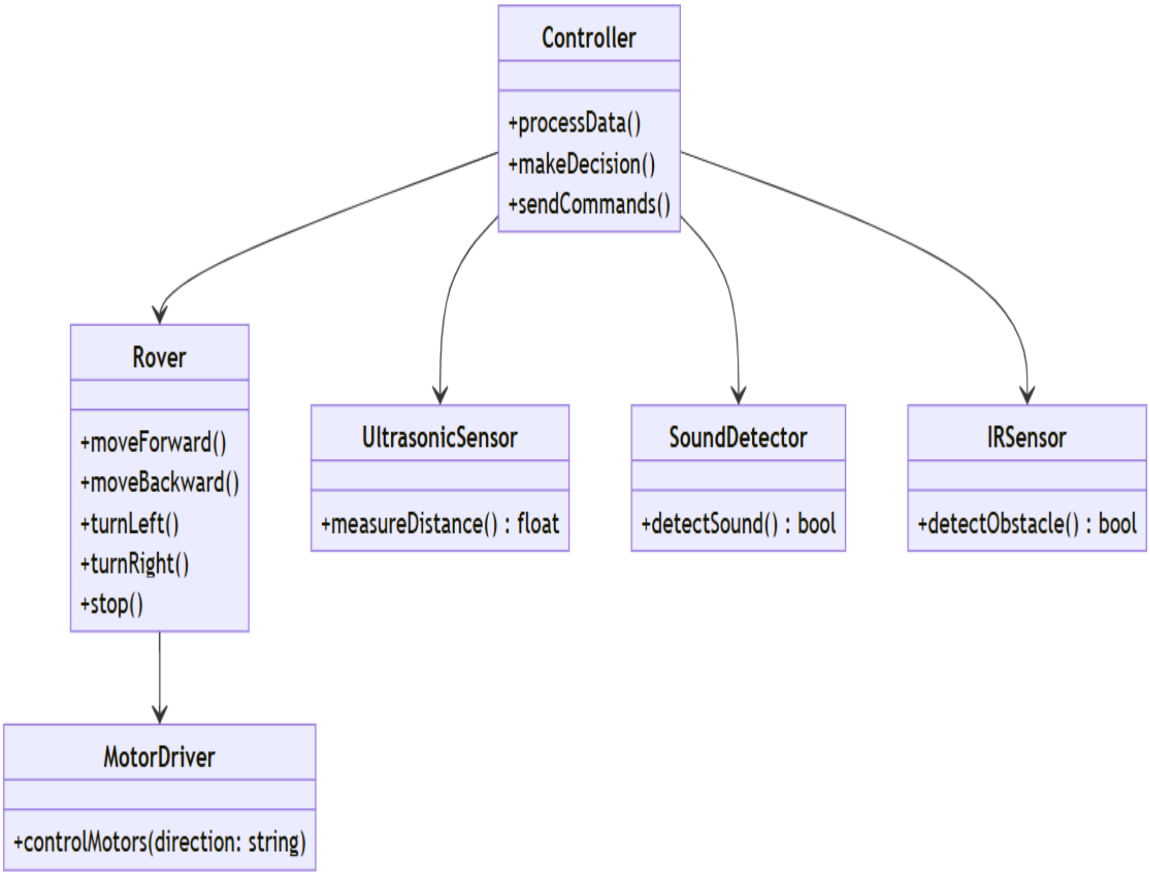


Figure 3.4 Class Diagram

The above Figure 3.4 is the class diagram for the system.

CHAPTER 4

MODULE DESCRIPTION

4.1 HARDWARE MODULE:

This module includes the ESP32 microcontroller along with the ultrasonic sensor, IR sensor, and sound detector. The ESP32 collects real-time data from the sensors and controls the motor driver based on environmental feedback. Its dual-core processing and built-in Wi-Fi capabilities make it ideal for handling sensor inputs and connectivity simultaneously. This module enables the rover to autonomously navigate through rescue environments while detecting obstacles and sound sources.

4.2 DATA COLLECTION PROCESSING MODULE:

This module interfaces with the hardware to collect data from all sensors. The ESP32 processes this data locally to determine movement, obstacle avoidance, and victim detection logic. It also analyzes sensor values and can trigger decision-making algorithms based on programmed thresholds. The module may transmit data via Wi-Fi using HTTP or MQTT protocols to a server or dashboard for remote monitoring.

4.3 ALERTING MODULE:

Continuously monitors processed data from the sound and distance sensors. When a potential victim is detected (based on sound intensity or pattern), or an abnormal environment is encountered, this module triggers an alert. Alerts can be sent via push notifications, emails, or updated in a web/mobile dashboard using ESP32's Wi-Fi capability to ensure immediate operator awareness.

4.4 WEB INTERFACE MODULE:

A web-based dashboard (or mobile app) that receives live data from the rover through Wi-Fi. Provides real-time visualization of the rover's status, movement, sensor readings, and alerts. Allows rescue operators to monitor the rover remotely, view detection logs, and intervene manually if needed.

4.5 INTEGRATION MODULE:

This module ensures seamless communication between the **ESP32**, **sensors**, and the **web interface**. It coordinates data flow from the ESP32 to the cloud or local server and synchronizes status updates and alerts to the dashboard. This tight integration allows for a real-time closed-loop system where sensor data, decision logic, and operator visibility are tightly linked.

CHAPTER 5

SAMPLE CODING

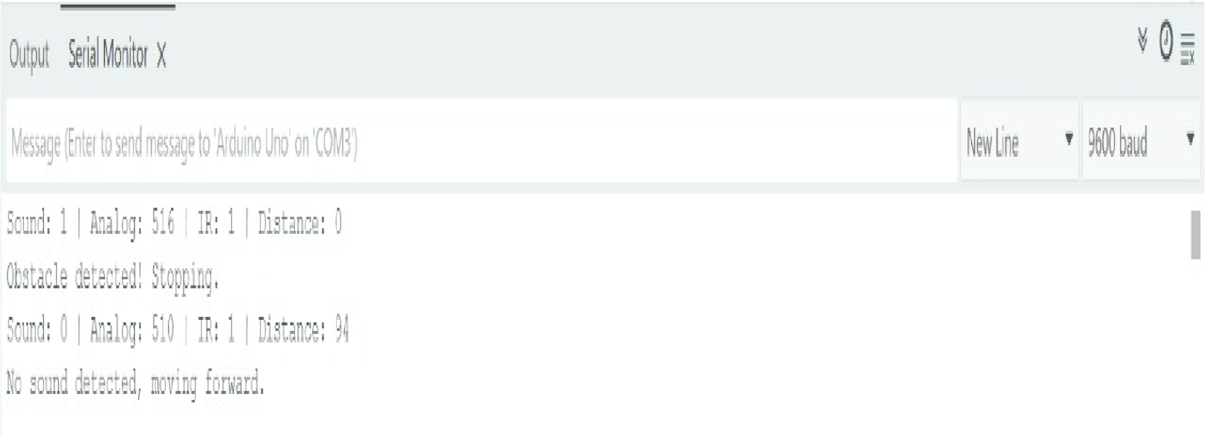
Program

```
#include <Wire.h>
#include <Adafruit_MLX90614.h>

#define TRIG_PIN 9
#define ECHO_PIN 10
#define LED_PIN 7
#define SOUND_ANALOG A0
#define SOUND_DIGITAL 2

Adafruit_MLX90614 mlx = Adafruit_MLX90614();

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

  pinMode(TRIG_PIN, OUTPUT);
  pinMode(ECHO_PIN, INPUT);
  pinMode(LED_PIN, OUTPUT);
  pinMode(SOUND_DIGITAL, INPUT);

  Serial.println("Smart Sensor System Initialized");
}

void loop() {
  // --- Temperature ---
  float temperature = mlx.readObjectTempC();

  // --- Ultrasonic distance ---
  digitalWrite(TRIG_PIN, LOW);
  delayMicroseconds(2);
  digitalWrite(TRIG_PIN, HIGH);
  delayMicroseconds(10);
  digitalWrite(TRIG_PIN, LOW);

  long duration = pulseIn(ECHO_PIN, HIGH);
  float distanceMM = duration * 0.34 / 2; // in mm

  // --- Sound ---
  int soundAnalog = analogRead(SOUND_ANALOG);
  int soundDigital = digitalRead(SOUND_DIGITAL);

  // --- LED only ON if object within 50mm ---
  if (distanceMM > 0 && distanceMM <= 50) {
    digitalWrite(LED_PIN, HIGH);
    Serial.println("LED: GLOWING (Object within 50mm)");
  } else {
    digitalWrite(LED_PIN, LOW);
    Serial.println("LED: OFF (No object within 50mm)");
  }

  // --- Other sensor status ---
  Serial.print("Temperature: ");
  Serial.print(temperature);
  Serial.print(" °C | Sound Level (Analog): ");
  Serial.print(soundAnalog);

  if (soundDigital == HIGH) {
    Serial.println(" | Loud Sound Detected!");
  } else {
    Serial.println(" | No Loud Sound");
  }

  delay(500);
}
```

CHAPTER 7

SCREEN SHOTS

1. Dashboard Page



New Line

9600 baud

```
Sound: 1 | Analog: 516 | IR: 1 | Distance: 0
Obstacle detected! Stopping.
Sound: 0 | Analog: 510 | IR: 1 | Distance: 94
No sound detected, moving forward.
```

Figure 7.1 Responsive Dashboard

2. Smart Sensing Rover

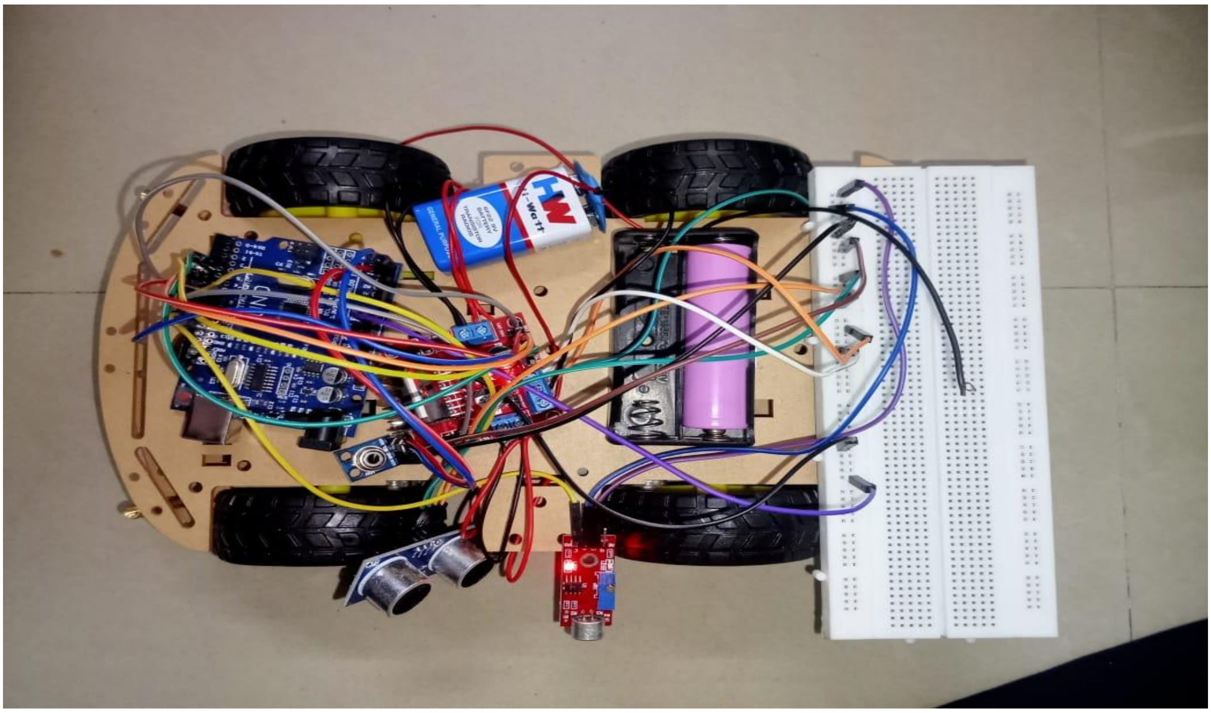


Figure 7.2 Smart Sensing Rover with its Detecting Features

CHAPTER 8

CONCLUSION AND FUTURE ENHANCEMENT

The **Smart Sensing Rover** provides an effective solution for detecting human presence and monitoring hazardous or difficult-to-reach areas. By utilizing sensors such as ultrasonic, sound, and IR, the rover can autonomously navigate its environment, detect obstacles, and identify signs of human activity. The system transmits real-time alerts and visuals to a remote device, enhancing safety and facilitating efficient response in emergency situations. The rover’s low-cost and scalable design make it a practical tool for a variety of applications, particularly in disaster management or search-and-rescue operations.

Future improvements to the rover could include integrating **GPS** for precise location tracking, enabling faster deployment in emergency scenarios. Voice command functionality could be added for more intuitive control, while a mobile app interface would allow remote monitoring and operation. Additionally, upgrading the rover with a **night vision** camera would enhance its performance in low-light environments, and incorporating **machine learning** algorithms could further improve human detection accuracy and enable the rover to recognize specific behaviors, making it more reliable in complex rescue operations.

REFERENCES

1. [1] S. K. Gupta, A. Bansal, and A. S. Malik, "IoT based multi-sensor smart rescue robot using ESP32," *Proceedings of the 2020 International Conference on Smart Electronics and Communication (ICOSEC)*, Trichy, India, 2020, pp. 253–258.
2. [2] R. Singh and N. Sharma, "Design and development of an autonomous rescue robot for disaster management," *2019 5th International Conference on Computing, Communication, Control and Automation (ICCUBEA)*, Pune, India, 2019, pp. 1–5.
3. [3] S. J. Prasad and D. R. Reddy, "IoT-based human detection robot using ESP32 and sensors," *2021 IEEE International Conference on Robotics and Smart Manufacturing (RoSMa)*, Chennai, India, 2021, pp. 134–138.
4. [4] M. Jain, V. Sharma, and P. Patil, "Development of a real-time environment monitoring rover using IoT," *2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE)*, Vellore, India, 2020, pp. 1–5.
5. [5] A. S. Pawar and K. R. More, "Survivor detection using wireless sensor network for rescue operation," *2018 International Conference on Communication and Signal Processing (ICCSP)*, Chennai, India, 2018, pp. 694–698.