Dr. BABASAHEB AMBEDKAR MARATHWADA UNIVERSITY,

CHH. SAMBHAJNAGAR,431001

PROJECT REPORT ON

Line Following Rover

SUBMITTED BY

Pranav Madhukar Sirsufale

Pratibha Ashok Wagh

Shraddha Shahadev Gulajkar

Rohan Bhaktpralhad Magar

Sunil Ansiram Misal

MSc CS (1st - Year)

Subject: Microcontroller Programming Using Python

YEAR 2024-25

SUBMITTED TO

Dr. Pravin L. Yannawar

Department of Computer Science,

Dr. Babasaheb Ambedkar Marathwada University,

Chh. Sambhajinagar.

Dr. Babasaheb Ambedkar Marathwada University, Chhatrapati Sambhaji Nagar – 431001.

NAAC Accredited with 'A+'

Department of Computer Science and Information Technology

CERTIFICATE

This is to certify that Mr./Mrs./Ku. _____

He/ She has successfully carried out the all Practical and Assignments of the subject **Microcontroller Programming Using Python** in the class of **M.Sc. (Computer Science) Sem.-IInd** during the academic year 2024-2025.

Hence Certified.

Seat No.:_____

Date: __/__/____

_____
Dr. Pravin L. Yannawar
Course Co-ordinator

_____
Prof. Ramesh R. Manza
Head of The Department

_____
Ms. Diksha Pawar
Practical In-Charge

_____
Examiner

Black Line Following Car Robot

1. Introduction

A line-following robot is a self-driving robotic vehicle that can detect and follow a black line on a white surface (or the other way around). These robots are commonly used in industrial automation, warehouses, and robotics competitions to navigate predefined paths efficiently.

Project Overview

In this project, we will:

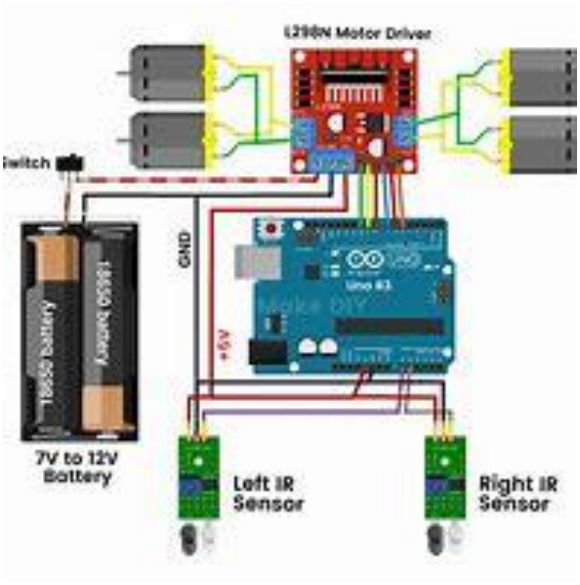
- Use infrared (IR) sensors to detect and track a black line.
- Control DC motors to move the robot forward, left, and right as needed.
- Build an Arduino-powered control system to automate movement.

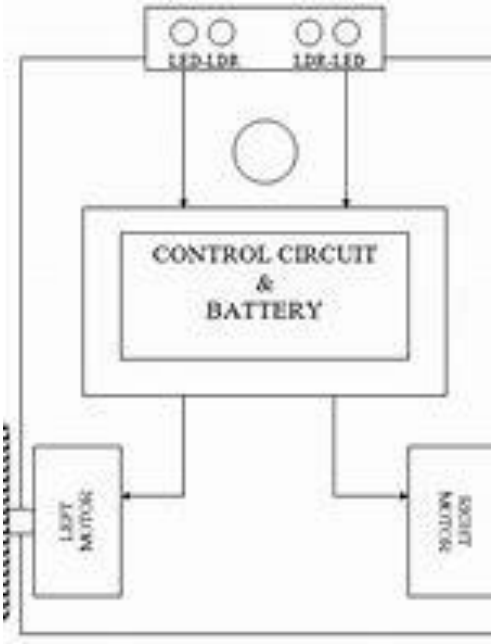
2. Objectives

The main goals of this project are:

- Designing and developing an autonomous robot that follows a predefined black line.
- Understanding how IR sensors work to distinguish between black and white surfaces.
- Using an Arduino microcontroller to process sensor input and control movements.
- Exploring motor driver circuits and their role in guiding robotic motion.
- Optimizing the navigation system through efficient coding logic for smoother movement.

Block Digram





3. Components Required

a) Hardware Components

Microcontroller Unit

- **Arduino Uno/Nano** – Acts as the brain of the robot, processing sensor inputs and controlling motors.

Sensors

- **IR Sensors (Infrared Sensors) – 2 units**
    - Used to detect the black line on a white surface.

- Works by emitting infrared light and detecting its reflection.
- Black absorbs IR light (no reflection), while white reflects it.

Motors & Driver

- **2x DC Motors** – Drive the robot's wheels forward and sideways.
- **Motor Driver Module (L298N)** – Allows the Arduino to control motor speed and direction.

Power Supply

- **9V or 12V Battery** – Provides power to the circuit.
- **Battery Holder** – Securely holds the battery.

Mechanical Structure

- **Robot Chassis** – The base frame where all components are mounted.
- **2x Wheels** – Connected to the motors to enable movement.
- **1x Caster Wheel** – Helps balance the car.

Connecting Components

- **Jumper Wires** – Used to connect all components.

4. Working Principle of the Line Following Robot

1. **IR Sensors detect the black line.**
2. **Arduino reads the sensor values.**
3. **Based on sensor input, Arduino controls the motors:**
    - If both sensors detect black → Move forward.
    - If only the left sensor detects black → Turn right.
    - If only the right sensor detects black → Turn left.
    - If no black detected → Stop.

5. Circuit Diagram & Connections

IR Sensor Connections

IR Sensor	Arduino Pin
Left Sensor Output	A0
Right Sensor Output	A1

VCC	5V
GND	GND

Motor Driver (L298N) Connections

Motor Driver (L298N)	Arduino Pin
IN1	9
IN2	10
IN3	11
IN4	12
Motor A (Left Motor)	L298N OUT1, OUT2
Motor B (Right Motor)	L298N OUT3, OUT4
VCC	9V-12V Battery
GND	GND

6. Step-by-Step Implementation

Step 1: Assembling the Robot Chassis

- Attach the two DC motors to the base.
- Fix two wheels onto the motor shafts.
- Place the caster wheel at the front or back for balance.

Step 2: Mounting the IR Sensors

- Attach two IR sensors to the front bottom of the chassis.

- Ensure one sensor is on the left side and the other on the right side.
- The sensors should be slightly above the surface (5mm - 10mm).

Step 3: Wiring the Components

- Connect the IR sensors to the Arduino (A0 & A1).
- Connect the motor driver to the Arduino and DC motors.
- Connect the battery to power the system.

Step 4: Uploading the Arduino Code

- Open Arduino IDE.
- Write the code to read sensor values and control motors.
- Upload the code to the Arduino board.

7. Arduino Code for Line Follower Robot

```
// Define IR sensor pins
#define leftSensor A0
#define rightSensor A1

// Define Motor Driver Pins
#define motor1A 9
#define motor1B 10
#define motor2A 11
#define motor2B 12

void setup() {
  pinMode(leftSensor, INPUT);
  pinMode(rightSensor, INPUT);
  pinMode(motor1A, OUTPUT);
  pinMode(motor1B, OUTPUT);
  pinMode(motor2A, OUTPUT);
```

```
  pinMode(motor2B, OUTPUT);
}



void loop() {
  int left = digitalRead(leftSensor);
  int right = digitalRead(rightSensor);

  if (left == 0 && right == 0) {
    moveForward();
  } else if (left == 1 && right == 0) {
    turnRight();
  } else if (left == 0 && right == 1) {
    turnLeft();
  } else {
    stopMotors();
  }
}

void moveForward() {
  digitalWrite(motor1A, HIGH);
  digitalWrite(motor1B, LOW);
  digitalWrite(motor2A, HIGH);
  digitalWrite(motor2B, LOW);
}
```

```
void turnRight() {
  digitalWrite(motor1A, LOW);
  digitalWrite(motor1B, LOW);
  digitalWrite(motor2A, HIGH);
  digitalWrite(motor2B, LOW);
}

void turnLeft() {
  digitalWrite(motor1A, HIGH);
  digitalWrite(motor1B, LOW);
  digitalWrite(motor2A, LOW);
  digitalWrite(motor2B, LOW);
}

void stopMotors() {
  digitalWrite(motor1A, LOW);
  digitalWrite(motor1B, LOW);
  digitalWrite(motor2A, LOW);
  digitalWrite(motor2B, LOW);
}
```

8. Expected Output

The robot moves forward along the black line.

- If the left sensor detects white, the car turns right.
- If the right sensor detects white, the car turns left.
- If no black line is detected, the robot stops.

9. Future Scope & Enhancements

Avoid obstacle - In order to detect objects or obstacle in the path use/add ultrasonic sensors

Control over speed – Implement PID control for smooth movement.
Wireless Control – Use Bluetooth/WiFi for remote operation.
AI & Computer Vision – Use OpenCV & Camera for smart navigation.
Industrial Applications – Automate warehouse robots for transportation.

10. Conclusion

This Black path following Rover is a simple but good project in order to understand robotics, controls , sensors and automation. It can be further developed into AI-Powered vehicle for smart applications.

11.Future scope

The **Black Line Following Car Robot** has a bright future in:

1. **Industrial Automation** – Used in factories and warehouses for material transport.
2. **Public Transport** – Can guide autonomous buses and trains.
3. **Healthcare** – Helps in hospitals for medicine and supply delivery.
4. **Smart Agriculture** – Assists in crop monitoring and automated farming.
5. **Military & Defense** – Used for unmanned transport in risky areas.
6. **Education & Research** – Enhances learning in robotics and AI.

With AI and IoT, these robots are evolving into **fully autonomous systems** for real-world applications.