Line Following Robot Using Arduino

Project Overview

A **Line Following Robot** is an autonomous robot that follows a **predefined path** using **infrared (IR) sensors**. It detects **black and white surfaces** and adjusts its movement accordingly. This type of robot is widely used in **automation, industrial transport, and robotics competitions**.

Objectives

✅ **Detect and follow a line using IR sensors.**
✅ **Control motor speed and direction using an Arduino.**
✅ **Use a PID or simple logic-based algorithm for smooth movement.**
✅ **Optimize for accuracy and efficiency.**

Components Required

1. **Arduino Uno** – Main microcontroller.
2. **IR Sensor Module (2 or more)** – Detects the black/white line.
3. **Motor Driver Module (L298N or L293D)** – Controls DC motors.
4. **DC Motors (2x)** – Drives the robot.
5. **Wheels & Chassis** – Robot body structure.
6. **Battery (7.4V or 12V Li-ion)** – Powers the system.
7. **Jumper Wires & Breadboard** – For connections.

How the System Works

1. **IR Sensors** detect the black/white surface.
2. The **Arduino processes the sensor data** and determines movement direction.
3. The **Motor Driver (L298N)** controls the **DC motors** accordingly:
    - If the robot is **centered on the line**, it moves **forward**.
    - If the **left sensor detects black**, the robot **turns left**.
    - If the **right sensor detects black**, the robot **turns right**.
    - If **both sensors detect white**, the robot **stops**.

Circuit Diagram

IR Sensors to Arduino

IR Sensor Pin	Arduino Pin
VCC	5V
GND	GND
Left Sensor OUT	D2
Right Sensor OUT	D3

Motor Driver (L298N) to Arduino

L298N Pin	Arduino Pin	Function
IN1	D4	Left motor forward
IN2	D5	Left motor backward
IN3	D6	Right motor forward
IN4	D7	Right motor backward
ENA (PWM)	D9	Left motor speed
ENB (PWM)	D10	Right motor speed
VCC	12V Battery	Motor power
GND	GND	Ground connection

Arduino Code for Line Following Robot


```
#define leftSensor 2  // Left IR sensor
#define rightSensor 3 // Right IR sensor

#define motorA1 4
#define motorA2 5
#define motorB1 6
#define motorB2 7

void setup() {
  pinMode(leftSensor, INPUT);
  pinMode(rightSensor, INPUT);

  pinMode(motorA1, OUTPUT);
  pinMode(motorA2, OUTPUT);
  pinMode(motorB1, OUTPUT);
  pinMode(motorB2, OUTPUT);
}

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

```
  if (left == LOW && right == LOW) { // Both sensors on black
    moveForward();
  }
  else if (left == LOW && right == HIGH) { // Left sensor on black, right on
white
    turnLeft();
  }
  else if (left == HIGH && right == LOW) { // Right sensor on black, left on
white
    turnRight();
  }
  else { // Both sensors on white
    stopRobot();
  }
}

void moveForward() {
  digitalWrite(motorA1, HIGH);
  digitalWrite(motorA2, LOW);
  digitalWrite(motorB1, HIGH);
  digitalWrite(motorB2, LOW);
}

void turnLeft() {
  digitalWrite(motorA1, LOW);
  digitalWrite(motorA2, LOW);
  digitalWrite(motorB1, HIGH);
  digitalWrite(motorB2, LOW);
}

void turnRight() {
  digitalWrite(motorA1, HIGH);
  digitalWrite(motorA2, LOW);
  digitalWrite(motorB1, LOW);
  digitalWrite(motorB2, LOW);
}

void stopRobot() {
  digitalWrite(motorA1, LOW);
  digitalWrite(motorA2, LOW);
  digitalWrite(motorB1, LOW);
  digitalWrite(motorB2, LOW);
}
```

How to Use the Line Following Robot

1. **Assemble the robot** with **IR sensors facing downward** near the ground.
2. **Upload the code** to the Arduino using the Arduino IDE.
3. **Place the robot on a black line** with a white background.
4. **Power the system** using a 12V battery.
5. The robot will **automatically detect and follow the line**.
6. Adjust the **IR sensor sensitivity** if needed.

Features & Benefits

✅ **Fully autonomous** movement based on sensor input.
✅ **Precise and fast line following** using dual IR sensors.
✅ **Expandable system** – Add more sensors for complex paths.
✅ **Easy to build and modify** for robotics learning.

Future Enhancements

Use 3 or more IR sensors for better accuracy.
Implement PID control for smoother movement.
Add ultrasonic sensors for obstacle detection.
Integrate Bluetooth/Wi-Fi for remote control.
Use a camera with AI for advanced path recognition.