Name : SAIDEEP RAHUL KULKARNI

College : PRAVARA RURAL ENGINEERING COLLEGE, LONI

Project : LINE FOLLOWER ROBOT USING ARDUINO

Components :

Line Follower Robot

Component List

Name	Quantity	Component
U1	1	Arduino Uno R3
M1 M2	2	Hobby Gearmotor
BAT1	1	9V Battery
U2	1	H-bridge Motor Driver
U3 U4 U5 U6 U7	5	IR sensor
U9	1	5V Regulator [LM7805]

Description :

A **Line Follower Robot** is an autonomous mobile robot designed to follow a predefined path, typically a black line on a white surface or vice versa. It utilizes **infrared (IR) sensors** to detect the line and adjust its movement accordingly. The robot is controlled by an **Arduino Uno**, which processes the

sensor data and drives the motors through an **L293D motor driver**.

Working Principle :

1. IR Sensor Functionality

- **Each IR sensor** consists of an **IR LED (emitter)** and a **photodiode (receiver)**.
- The IR LED emits **infrared light**, which reflects off **white surfaces** and is **absorbed by black surfaces**.
- The photodiode detects the reflected IR light and provides an **output signal**:
    - **White surface (reflection) → HIGH signal (1)**
    - **Black line (absorption) → LOW signal (0)**

The robot typically uses **5 IR sensors**, arranged in a straight line at the front:

- **Left-most sensor (S1)**
- **Left sensor (S2)**
- **Center sensor (S3)**
- **Right sensor (S4)**
- **Right-most sensor (S5)**

---

2. Decision-Making Process (Sensor Readings → Movement)

Based on the sensor readings, the Arduino decides the movement of the robot:

A. Moving Forward (Straight Line)

- If **only the center sensor (S3) detects the black line**, the robot moves forward.
- Motors:
  - Left motor → **Forward** ◦ Right motor → **Forward**

B. Slight Right Turn (Correcting Deviation)

- If **S3 and S4 detect the black line**, the robot slightly turns right.
- Motors:
  - Left motor → **Forward (Normal Speed)** ◦ Right motor → **Slow Forward**

C. Sharp Right Turn

- If **only S5 detects the black line**, the robot makes a **sharp right turn**.
- Motors:
  - Left motor → **Forward**
  - Right motor → **Reverse** (for faster turning)

D. Slight Left Turn (Correcting Deviation)

- If **S3 and S2 detect the black line**, the robot slightly turns left.
- Motors:
  - Left motor → **Slow Forward**
  - Right motor → **Forward (Normal Speed)**

E. Sharp Left Turn

- If **only S1 detects the black line**, the robot makes a **sharp left turn**.
- Motors:
  - Left motor → **Reverse** (for faster turning)
  - Right motor → **Forward**

F. No Line Detected (Stop)

- If **all sensors read HIGH (1)**, it means the robot is off the track.
- The robot stops until it detects the line again.

---

3. Motor Control via L293D

- The **Arduino cannot drive motors directly**, so it sends control signals to the **L293D motor driver**.
- The **L293D driver** receives these signals and supplies the necessary power from the **9V battery** to run the **DC motors** in the correct direction.

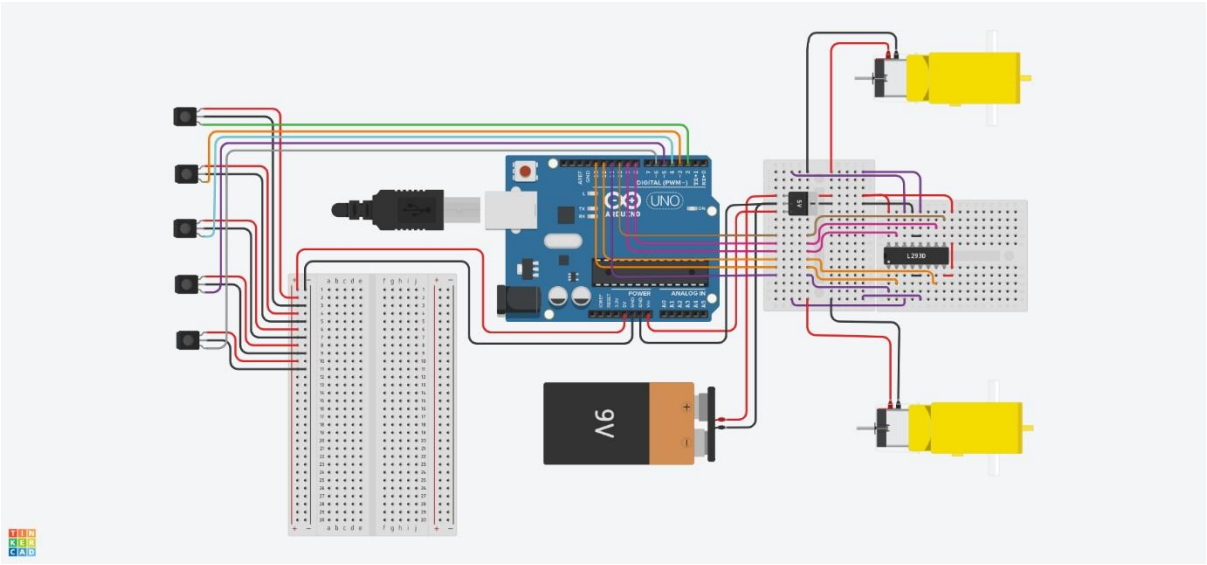
- Speed can be controlled using **PWM signals** on the **Enable (EN1 & EN2) pins** of the L293D.

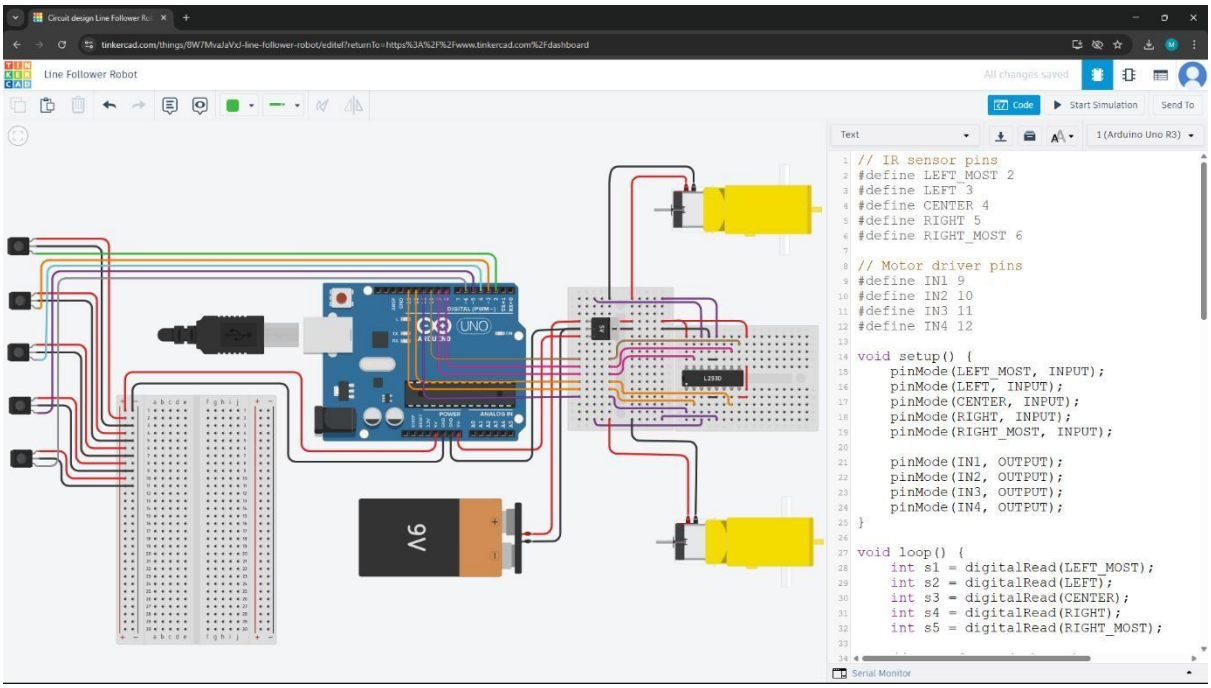
---

4. Real-Time Corrections & Adjustments

- The robot continuously reads the sensor values in a **loop** and updates motor speeds accordingly.
- If a deviation occurs (e.g., turning too much), the robot **quickly corrects itself** based on sensor input.
- The performance depends on **sensor calibration**, **motor speed**, and **track design**.

Schematic Diagram :





Code Explaination :

1. Defining Pin Connections

// IR sensor pins

#define LEFT_MOST 2

#define LEFT 3

```
#define CENTER 4
#define RIGHT 5
#define RIGHT_MOST 6
```

These define the 5 IR sensor pins connected to the Arduino. Each sensor detects the line (black/white contrast).

```
// Motor driver pins
#define IN1 9
#define IN2 10
#define IN3 11
#define IN4 12
```

These define the motor driver control pins connected to L293D.

- IN1, IN2 → Left motor
- IN3, IN4 → Right motor

Each motor moves forward or backward based on HIGH/LOW signals.

---

2. Setting Up the Pins

```
void setup() {
  pinMode(LEFT_MOST, INPUT);
pinMode(LEFT, INPUT);    pinMode(CENTER,
```

```
INPUT);     pinMode(RIGHT, INPUT);
pinMode(RIGHT_MOST, INPUT);

   pinMode(IN1, OUTPUT);
pinMode(IN2, OUTPUT);     pinMode(IN3,
OUTPUT);     pinMode(IN4, OUTPUT);
}
```

- IR sensors are set as INPUT because they provide data to Arduino.
- Motor driver pins are set as OUTPUT because Arduino controls the motors.

---

3. Reading Sensor Values and Making Decisions

```
void loop() {
   int s1 = digitalRead(LEFT_MOST);
int s2 = digitalRead(LEFT);     int s3 =
digitalRead(CENTER);     int s4 =
digitalRead(RIGHT);     int s5 =
digitalRead(RIGHT_MOST);
```

The Arduino reads sensor values (1 = line detected, 0 = no line).

4. Movement Logic Based on Sensor Readings

Moving Forward

```
if (s3 == 1) {
moveForward();
}
```

- The center sensor detects the line, so the robot moves straight.

Turning Right (Slight and Sharp)

```
else if (s4 == 1) {
   turnRight();
}
else if (s5 == 1) {
sharpRight();
}
```

- If the right sensor detects the line, the robot makes a slight right turn.
- If the extreme right sensor detects the line, the robot makes a sharp right turn.

Turning Left (Slight and Sharp)

```
else if (s2 == 1) {     turnLeft();
}
else if (s1 == 1) {
sharpLeft();
}
```

- If the left sensor detects the line, the robot makes a slight left turn.
- If the extreme left sensor detects the line, the robot makes a sharp left turn. Stopping the Robot cpp

```
CopyEdit else {
stopMotors();
}
```

- If no sensors detect the line, the robot stops.

---

5. Motor Control Functions

Each function sends signals to L293D motor driver to control the motors. Move Forward cpp CopyEdit void

```
moveForward() {     digitalWrite(IN1, HIGH);
digitalWrite(IN2, LOW);     digitalWrite(IN3, HIGH);
digitalWrite(IN4, LOW);
}
```

- Left motor moves forward (IN1 = HIGH, IN2 = LOW).
- Right motor moves forward (IN3 = HIGH, IN4 = LOW).

Turn Right (Slight) void

turnRight() {

digitalWrite(IN1, HIGH);

digitalWrite(IN2, LOW);

digitalWrite(IN3, LOW);

digitalWrite(IN4, LOW);

}

- Left motor moves forward, while the right motor stops, causing a slight right turn.

Sharp Right Turn void

sharpRight() {

digitalWrite(IN1, LOW);

digitalWrite(IN2, HIGH);

digitalWrite(IN3, HIGH);

digitalWrite(IN4, LOW);

}

- Left motor moves backward, while the right motor moves forward, making a sharp right turn.

Turn Left (Slight) void

turnLeft() {

digitalWrite(IN1, LOW);

digitalWrite(IN2, LOW);

digitalWrite(IN3, HIGH);

digitalWrite(IN4, LOW);

}

- Right motor moves forward, while the left motor stops, causing a slight left turn.

Sharp Left Turn void

sharpLeft() {

digitalWrite(IN1, HIGH);

digitalWrite(IN2, LOW);

digitalWrite(IN3, LOW);

digitalWrite(IN4, HIGH);

}

- Right motor moves backward, while the left motor moves forward, making a sharp left turn.

Stopping the Motors void

stopMotors() {

digitalWrite(IN1, LOW);

digitalWrite(IN2, LOW);

digitalWrite(IN3, LOW);

digitalWrite(IN4, LOW);

}

- All motors stop when there is no valid sensor input.

Arduino code :

line_follower_robot
1.ino

Schematic View :

Line Follower
Robot.pdf

Tinkercad Link :

https://www.tinkercad.com/things/8W7MvaJaVxJ-line-followerrobot?sharecode=-tSc2elX9QHIyhndzuPlM0ov05z1mnAFzyInwVdsbH4