Fig. 1 Shakey the robot

1980s: Microcontroller Introduction: The introduction of programmable microcontrollers allowed robots to perceive and respond to visual inputs more effectively.

2000s–Present: Advanced Autonomy: The integration of AI, machine learning, and advanced sensor arrays has enabled, high-precision navigation in complex industrial environments, such as autonomous warehouse, vehicles.

Design Progression

Through effective analysis of previous line following robots models, the modern design of line-following robots has evolved and been modified to improve speed, reliability, and adaptability.

Sensing Technology

Early: Simple photocells or rudimentary optical sensors.

Intermediate: Two-sensor systems that could detect a black line on a white surface.

Modern: Arrays of 5 to 16+ infrared (IR) sensors, or CCD cameras, allowing for rapid detection, line-inversion handling, and complex path tracking.

Chapter 2

Literary Review

2.1 Introduction

This project focuses on the Design and Implementation of a Line Following Robot, a fundamental autonomous system in the field of robotics and embedded control. In modern industrial settings, autonomous vehicles are essential for material transportation and path-based navigation without human intervention.

This specific project integrates an Arduino microcontroller, Infrared (IR) sensors, and an L293D motor driver to create a robot capable of detecting and tracking a predefined line on a contrasting surface.

2.2 Review of Existing Robots

Researchers and students have extensively explored autonomous navigation using various configurations. Below are some common implementations of line-following systems:

1. **Basic Two-Sensor IR Robots**: The most common educational model, utilizing two infrared sensors to detect line edges and perform simple binary (on/off) steering.
2. **Multi-Sensor Array Robots**: Systems using five or more sensors to achieve higher resolution in path detection, allowing for smoother navigation through sharp curves.
3. **PID-Controlled Line Followers**: Advanced versions that use Proportional-Integral-Derivative algorithms to minimize oscillations and maintain a constant path center.
4. **Industrial Automated Guided Vehicles (AGVs)**: Large-scale robots used in warehouses to transport heavy materials along predefined floor tracks.
5. **High-Speed Racing Line Followers**: Specialized robots optimized with high-torque motors and lightweight chassis designed for competitive speed on technical tracks.

Fig. 2 A High-Speed Racing Line Following Robot

Chapter 3

Methodology

3.1 System Block Design

System Architecture

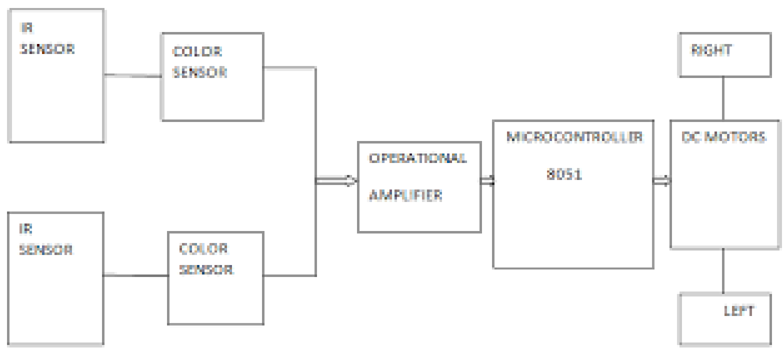

Fig. 3 System Block Design

The line-following robot was designed as an embedded control system consisting of three major subsystems:

Input Unit: Infrared (IR) sensors used for detecting the line path.

Processing Unit: Microcontroller board based on Arduino architecture for signal interpretation and decision making.

Output Unit: Motor driver circuit and DC motors responsible for movement execution.

These subsystems interact such that sensor signals are read by the microcontroller, processed using programmed logic, and translated into motor control actions.

3.2 Hardware Design

The hardware design followed these steps:

Component Selection

- Two IR sensors for path detection
- Arduino microcontroller board for control logic
- L293D motor driver IC for motor interfacing
- Two DC motors for locomotion

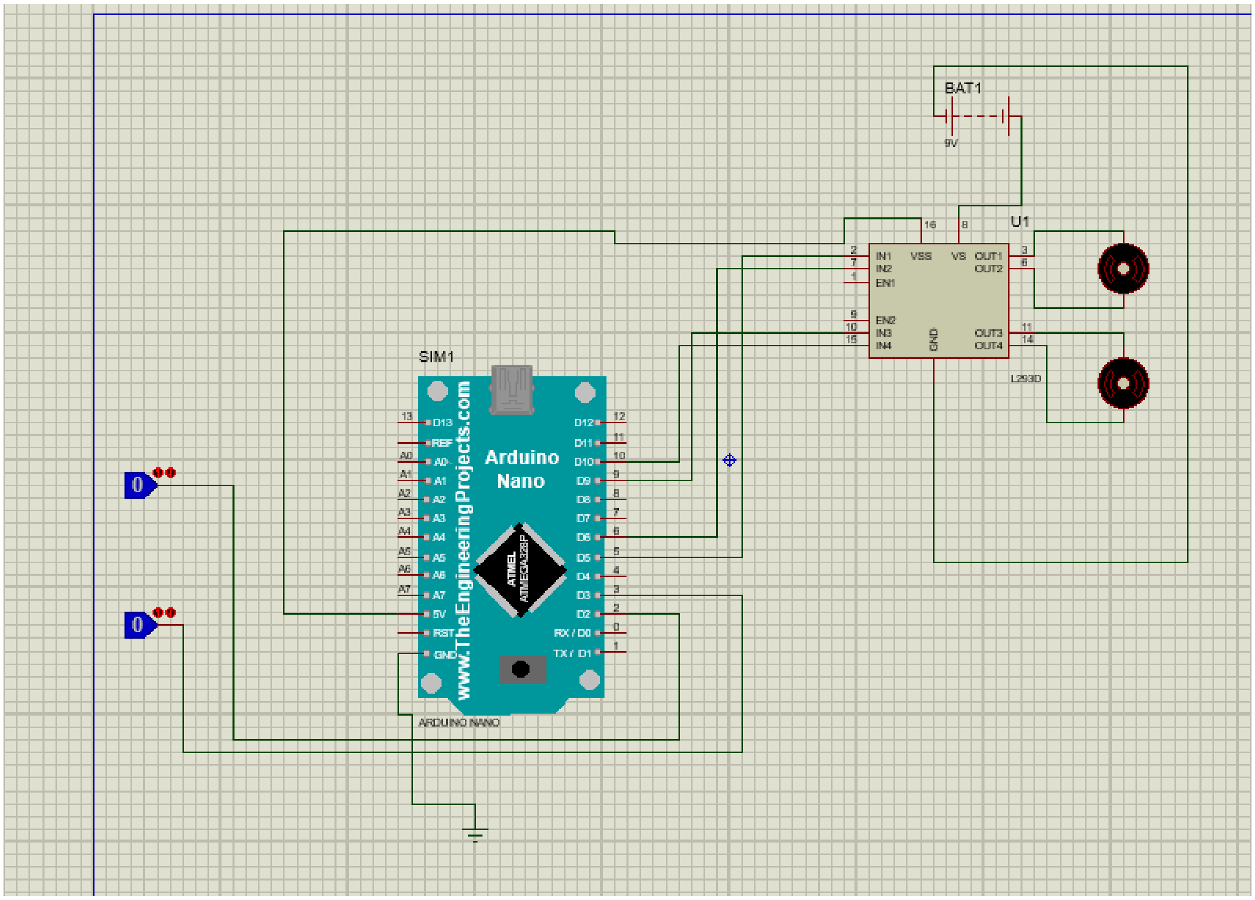

Fig. 4-Circuit Design in Proteus

The sensor pins were connected to Arduino input ports configured as digital inputs, while the motor driver control pins were connected to Arduino digital output pins. PWM-capable pins were selected for the enable terminals of the motor driver to allow speed regulation.

To protect the microcontroller from voltage fluctuations caused by the motors, common ground referencing and stable power routing were implemented across the circuit.

3.3 PCB Design

To improve system reliability and reduce wiring errors, a custom printed circuit board was developed using EasyEDA.

Component	Footprint Used	Reason
Arduino Interface	2.54 mm Female Headers	Matches Arduino pin spacing
Motor Driver IC	DIP-16 Through-Hole	Fits IC package exactly
IR Sensors	3-Pin Header	Allows detachable sensors
DC Motors	2-Pin Screw Terminal	Secure high-current connection
Battery Input	2-Pin Screw Terminal	Safe power connection
Resistors/LEDs	Axial TH / 5 mm LED	Standard PCB parts

Table 1. – Component Showing Their Compatibility and Advantages

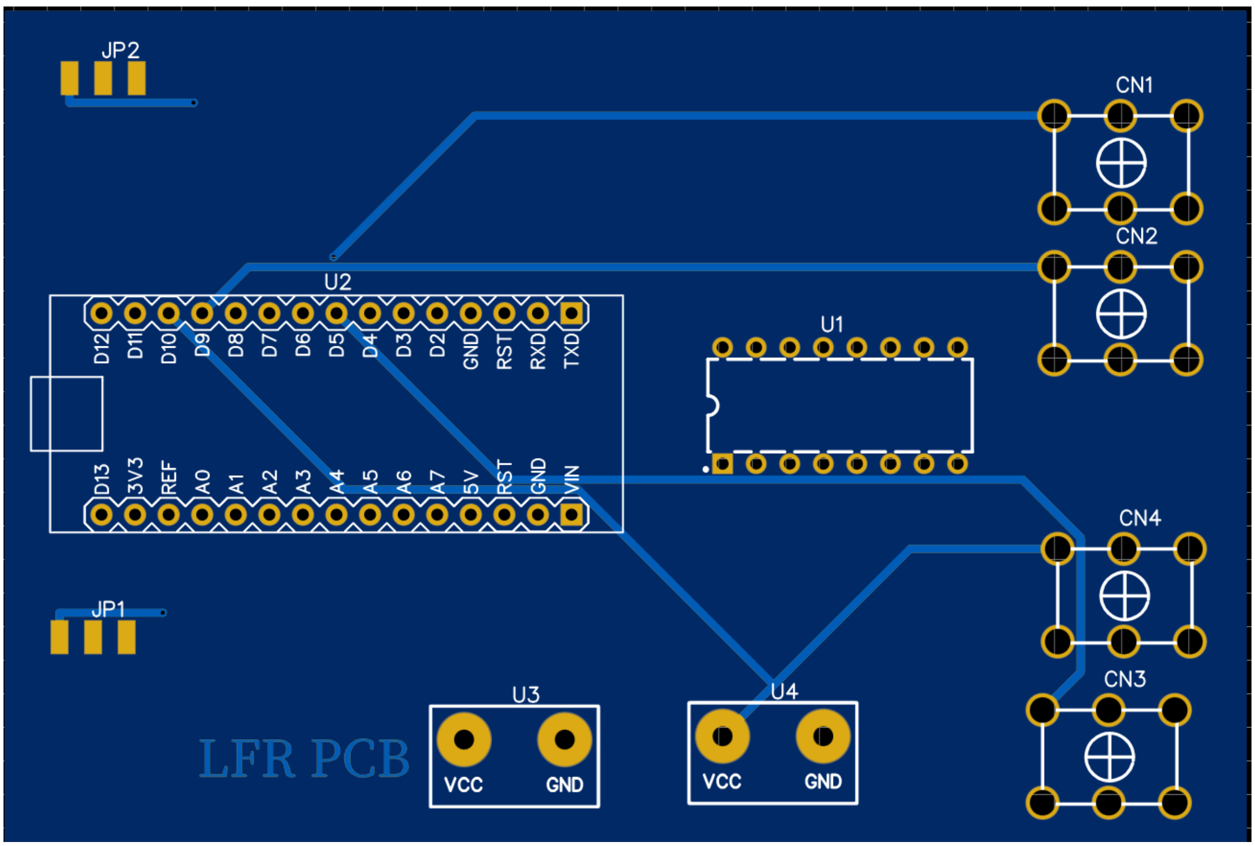

Fig 5-2D Design of the PCB Board

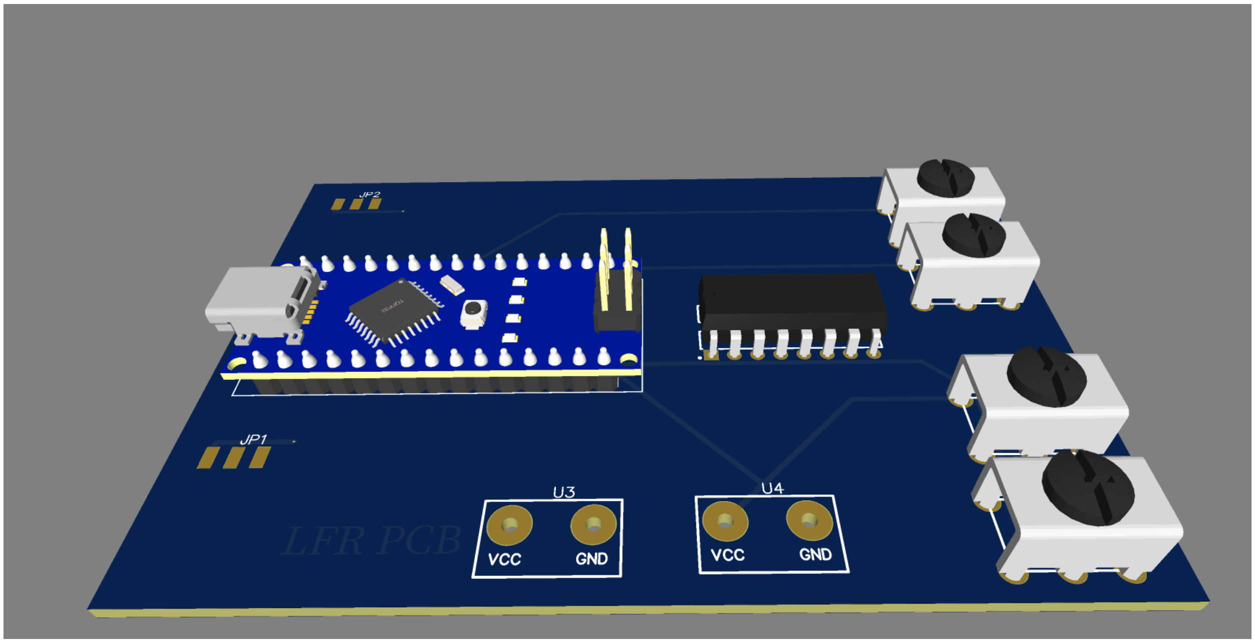

Fig.6-3D Design of PCB

3.4 Software Development

The control software was written in the Arduino programming environment and uploaded to the microcontroller.

Decision Table:

Left HIGH, Right HIGH → Forward
Left LOW, Right HIGH → Turn Left
Left HIGH, Right LOW → Turn Right
Left LOW, Right LOW → Stop

Flow chart of the code for a line following robot

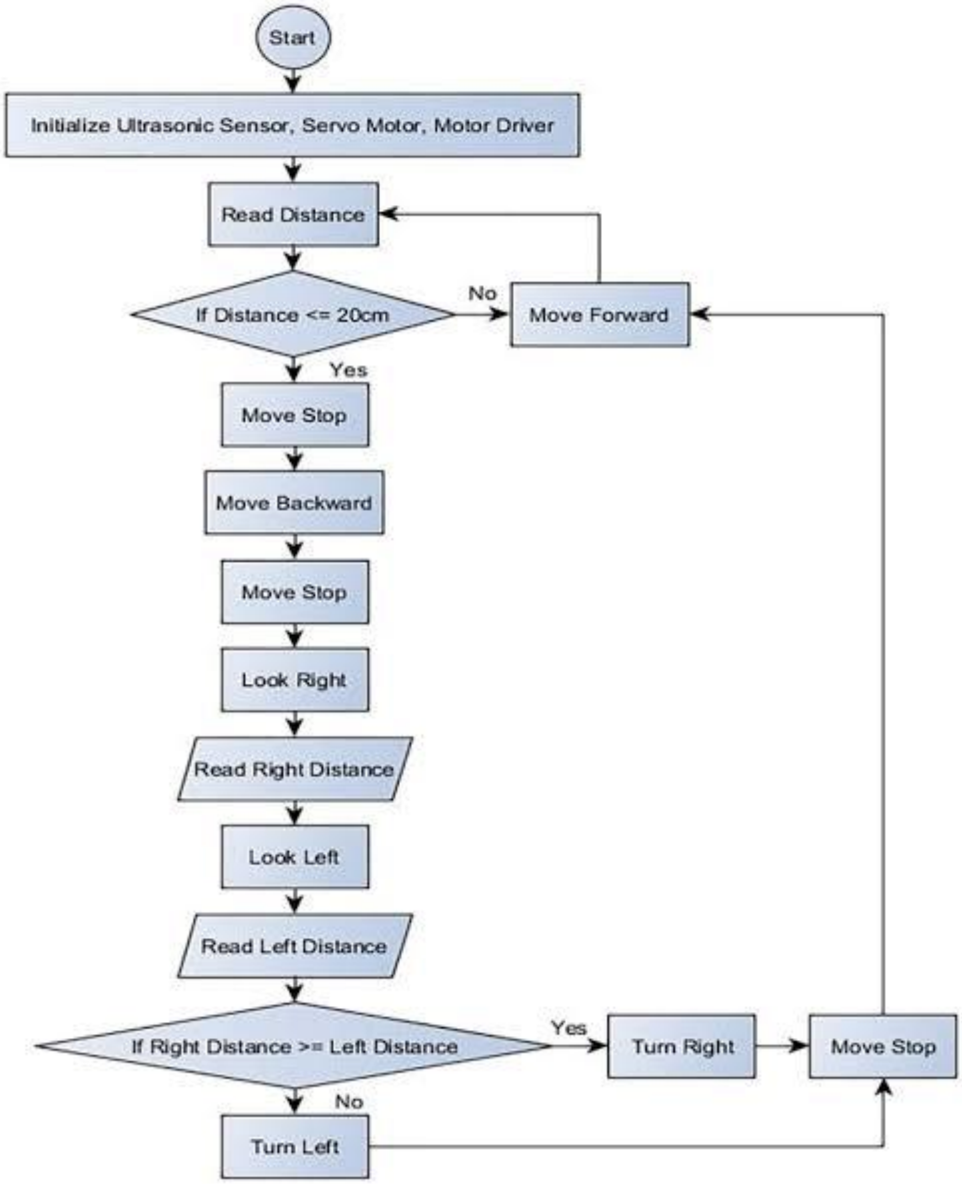

Fig.7 flow chart

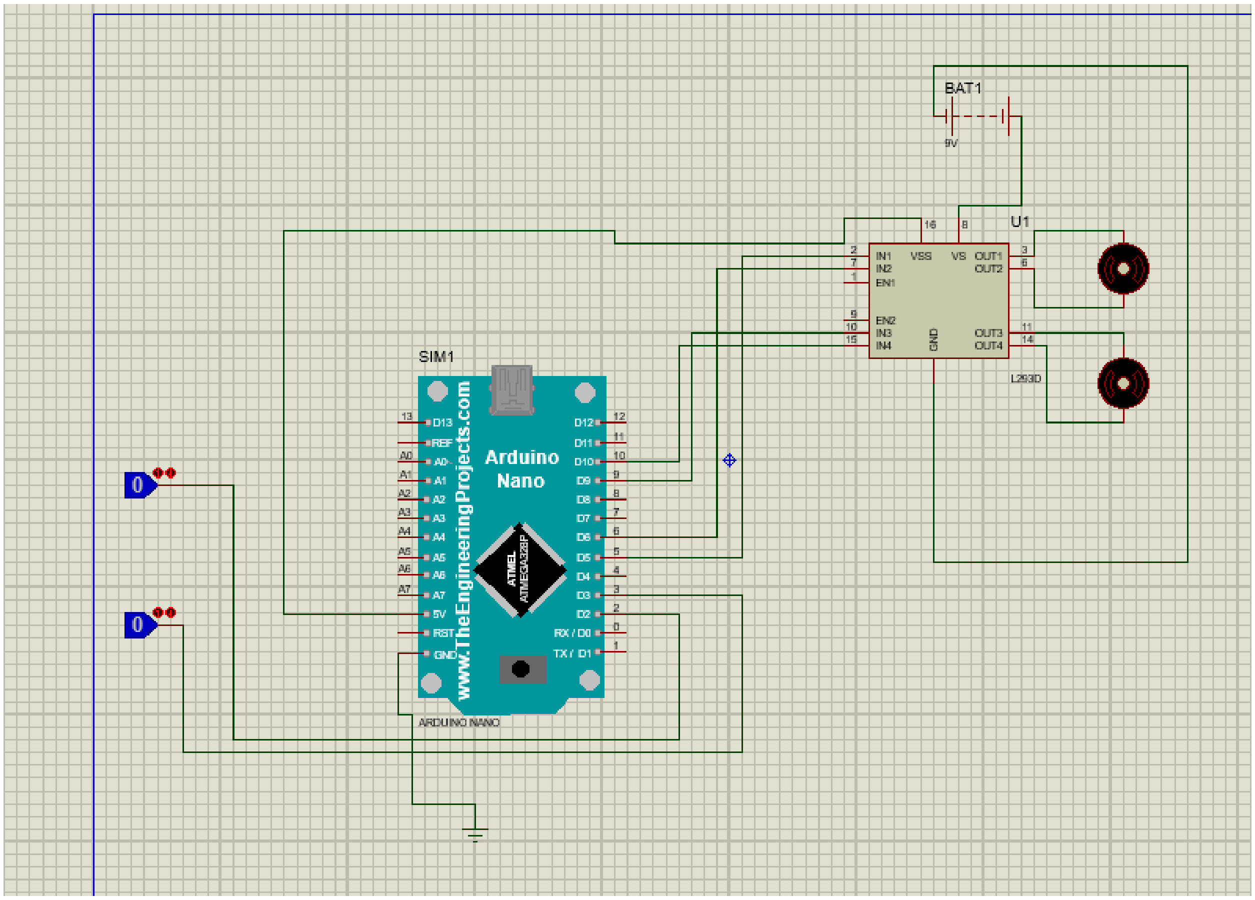

Fig. 4-Circuit Design in Proteus

4.2 Hardware Results

Although implementation of hardware was not done, the anticipated behavior of the system was obtained through simulation. Logical state outputs obtained from the proteus environment were systematically recorded and analyzed to evaluate overall system performance.

Time(s)	Left IR sensor	Right IR sensor	Motor action
0	0	0	Stop
2	1	0	Turn right
4	1	1	Move forward
6	0	1	Turn left
8	0	0	Stop

Table 2-Simulation of Sensors

The recorded data were plotted on Microsoft excel in order to generate a graphical representation of the line following robot's behavior.

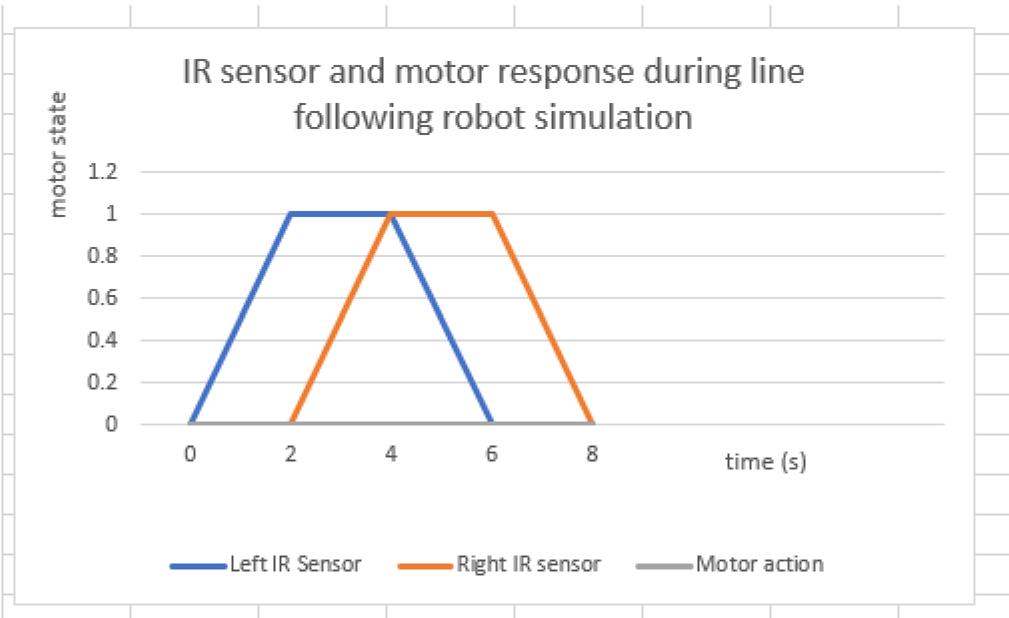

Fig.8-Graph of Simulation Results

4.3 PCB Performance Comparison

The circuit design was realized as a custom printed circuit board (PCB) using EasyEDA and its performance was subsequently evaluated in comparison with a conventional prototype writing (breadboard-based) approach.

Parameter	Prototype writing	Custom PCB
Signal stability	Moderate	High
Noise susceptibility	High	Low
Ease of troubleshooting	Difficult	Easy
Mechanical strength	Low	High
Overall stability	Moderate	High

Table 3-PCB Parameters and Performance

The PCB designed using EasyEDA showed enhanced reliability and improved signal integrity. Careful optimization of component placement and routing helped to reduce electrical noise and minimize power losses.In comparison with prototype writing, the PCB offered a more compact,robust and professional solution,makng it more suitable for practical,real-world deployment.

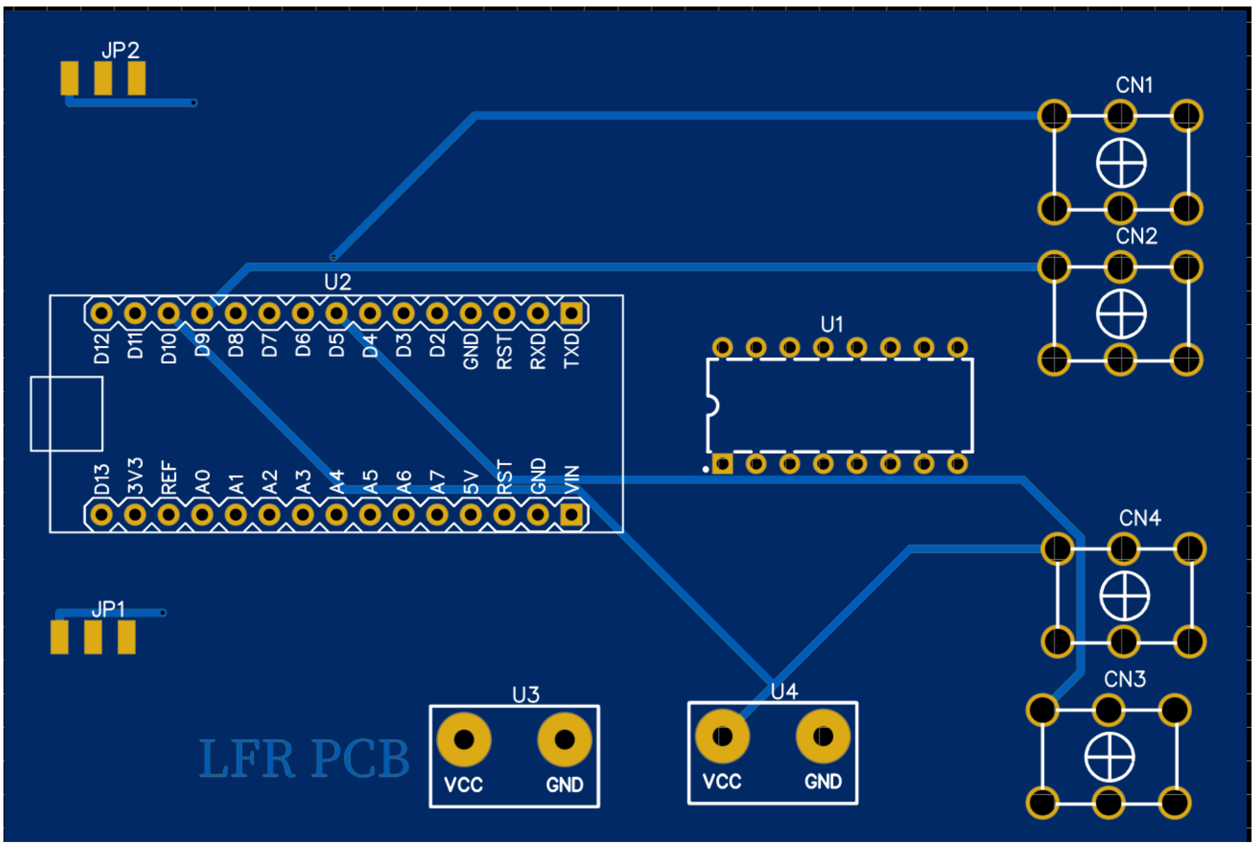

Fig. 5

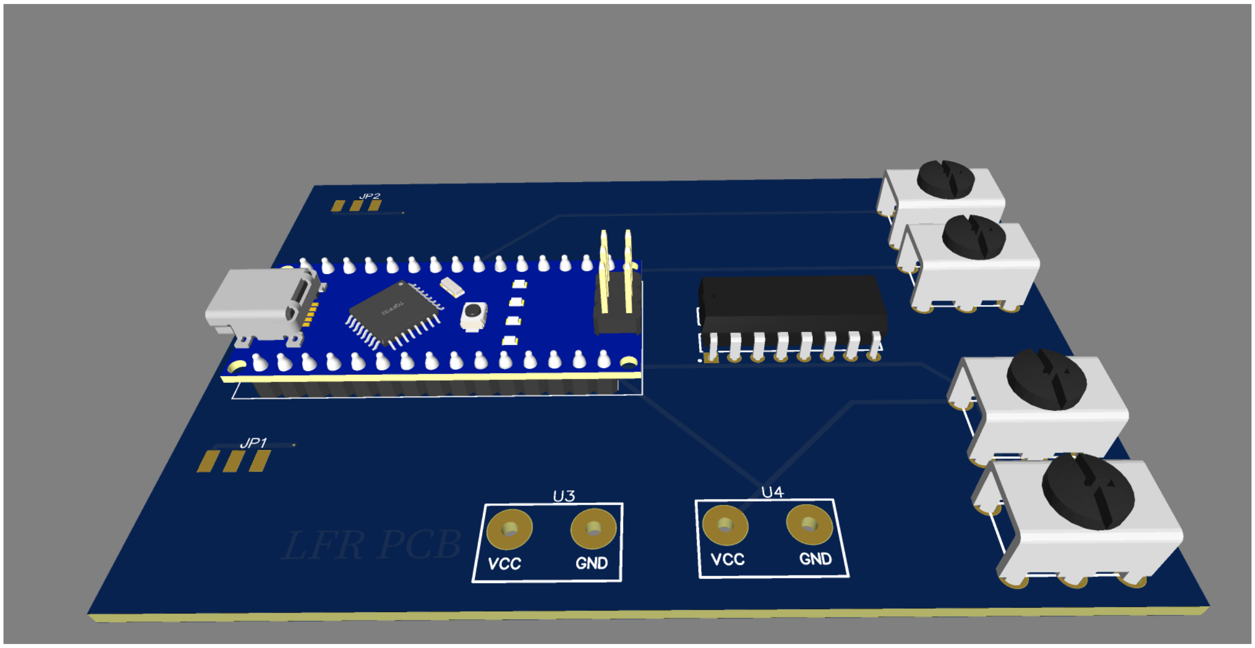

Fig. 6

4.4 Discussion of Findings

The outcomes of simulation, PCB design, and data analysis are consistent with the literature on line-following robotic systems. Research has demonstrated that IR sensor-based line-following robots provide dependable navigation when paired with efficient motor control algorithms; the observed behaviors—straight movement, left and right turns, and stopping—are consistent with standard line-following control strategies documented in robotics research; the use of an Arduino microcontroller and L293D motor driver is also widely supported in literature due to their affordability, simplicity, and ease of implementation; additionally, adopting a custom PCB design, as advised in many engineering studies, greatly increases system reliability and decreases wiring-related faults.

Appendix B

PCB Layout & Schematic

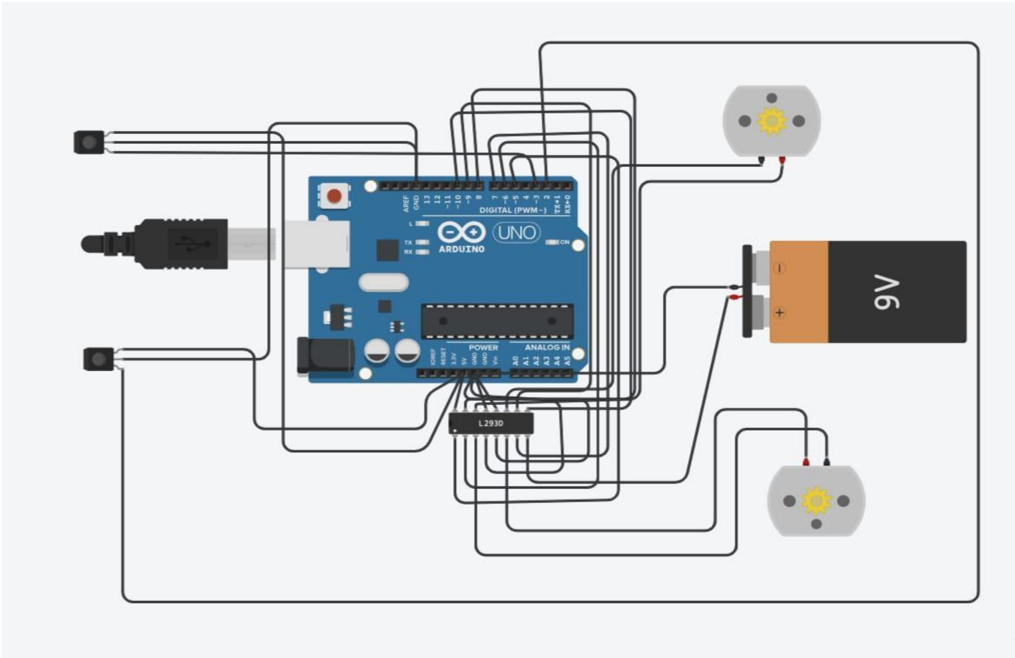

Fig. 8-PCB Layout

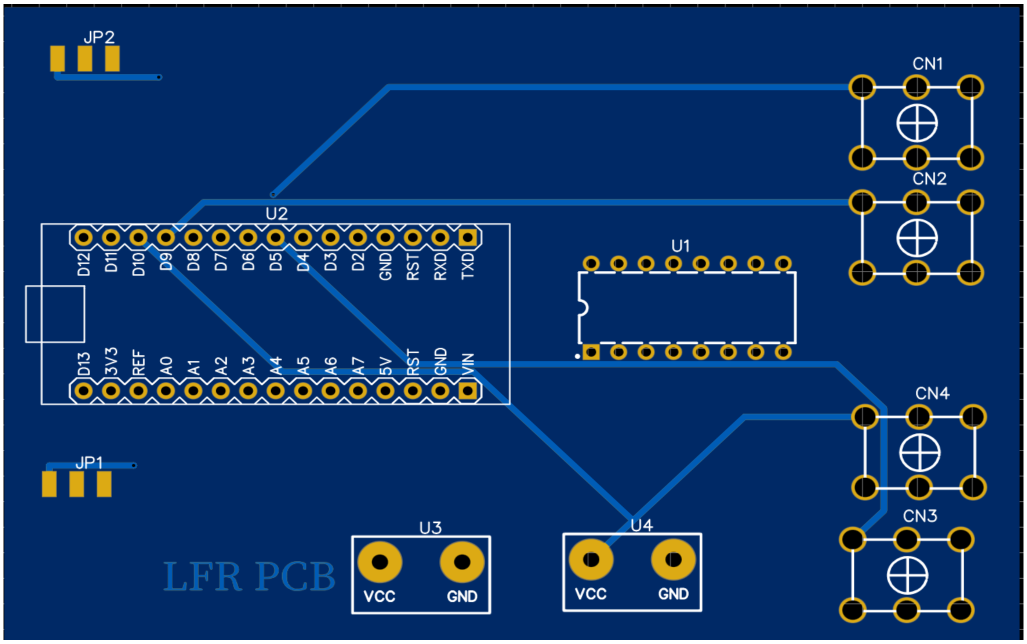

Fig. 5

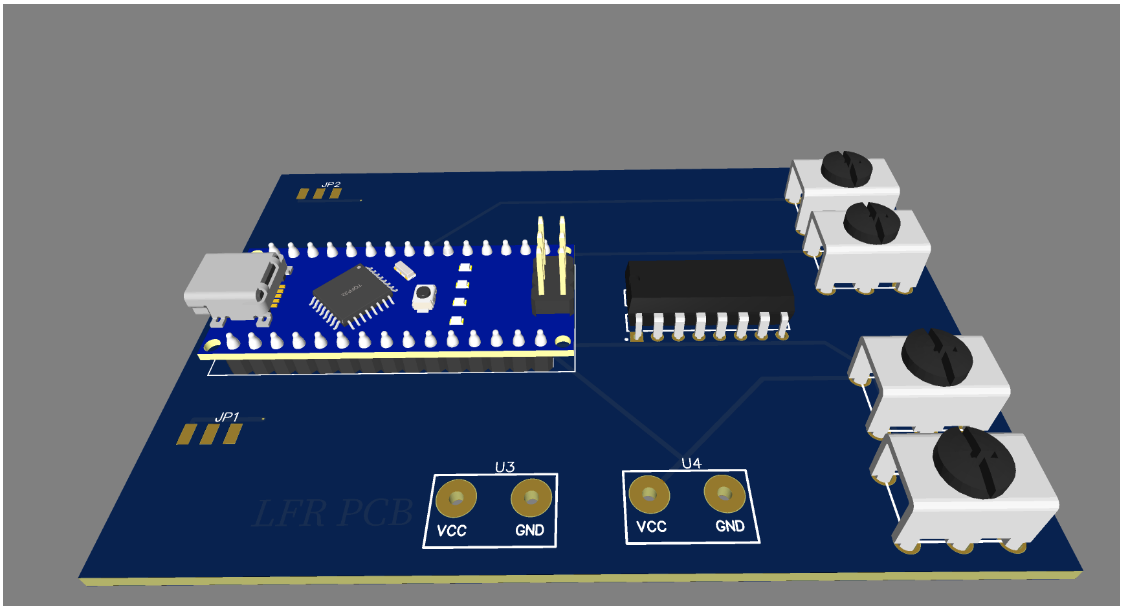

Fig. 6

Appendix C

Project Photos

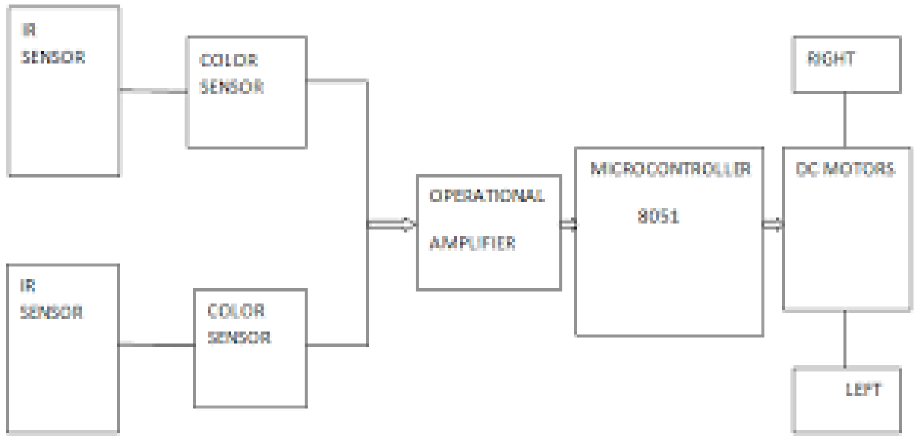

Fig. 3

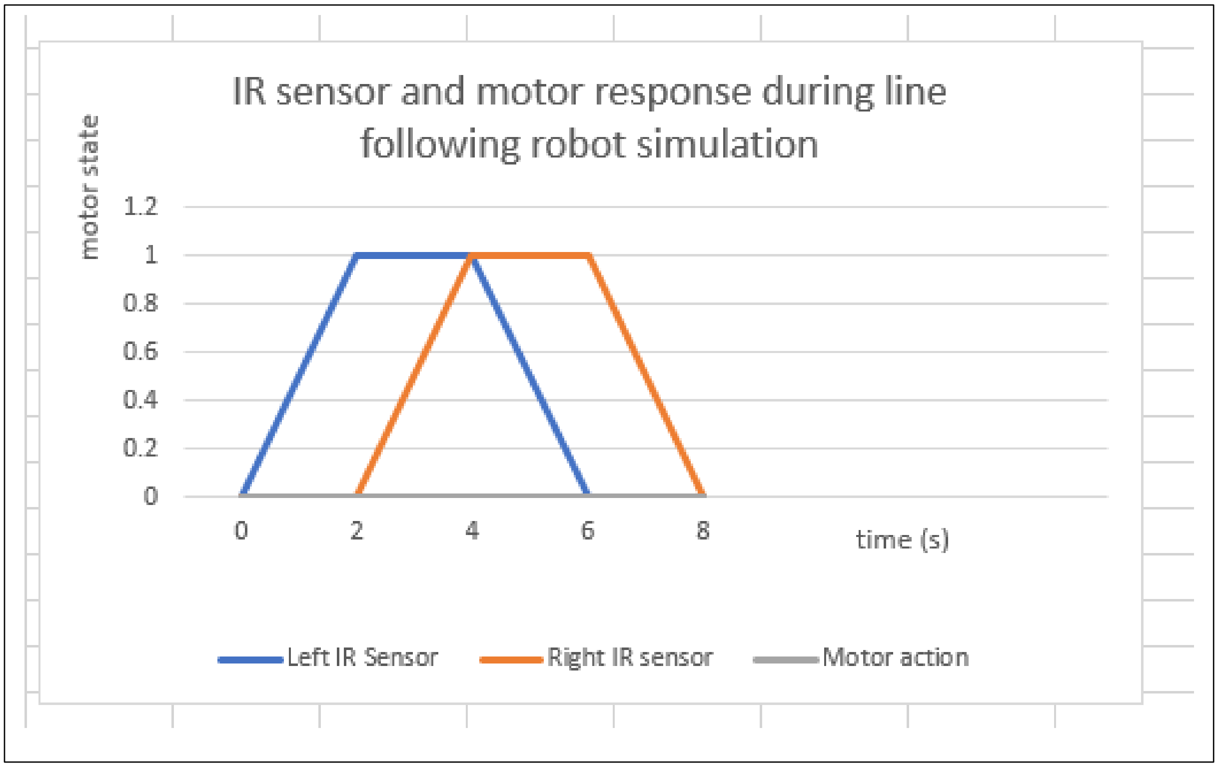
IR sensor and motor response during line following robot simulation
motor state
1.2
1
0.8
0.6
0.4
0.2
0
0
2
4
6
8
time (s)
Left IR Sensor
Right IR sensor
Motor action

Fig. 7