Line Follower & RC Robot

October 6 University

Faculty of Engineering, Mechatronics Department

Introduction to Mechatronics - MTE104

Prepared by:

Abdallah Gunaidy El-Sayed Hemeidah	23014373
Abdallah Mohammed Fathy Mohammed Mousa	23013075
Abdelrahman Ahmed El-Basiony	23014815
Abdelrahman Mohamed Fawzy	23017418
Abdel-Salam Mohamed Abdel-Aziz	23017606
Adel Mostafa Mohamed Ahmed	23015671
Ahmed Mostafa Ibrahim Nada	23015006
Ezz Eldin Gamil Gamal	23016741
Ezz Eldin Mostafa Ahmed	23015168

Supervisor: Dr. Ahmed Newir

Assistant Supervisor: Eng. Moemen Mohammed (TA)

Date: December 24, 2024

Abstract

This project presents the design and implementation of a dual-mode robotic car capable of operating autonomously using line-following sensors and manually via Bluetooth control. Built using Arduino, IR sensors, an HC-05 Bluetooth module, and a custom chassis, the robot demonstrates flexible navigation and modular functionality. This report outlines the mechanical design, wiring, software logic, challenges faced, and future improvements

Disclaimer

This project was developed for academic purposes at October 6 University, Faculty of Engineering – Mechatronics Department. All components, designs, and source code are intended for educational use only. The authors are not liable for any misuse or replication without proper citation.

Signed,

Project Team

[12-24-2024]

Acknowledgments

We would like to express our sincere gratitude to Dr. Ahmed Newir for his guidance and valuable feedback throughout the development of this project.

We also thank Eng. Moemen Mohammed for his continuous support, and all colleagues who contributed ideas and suggestions during testing and design

List of Contents

List of Figures

List of Tables

1 Introduction

In recent years, autonomous mobile robots have become increasingly popular in various fields such as industrial automation, transportation, and domestic assistance. These systems combine sensing, actuation, and control to perform tasks with minimal human intervention.

This project presents the design and implementation of a multi-mode robotic vehicle capable of operating in both autonomous line-following mode and manual Bluetooth-controlled mode. The robot is designed to follow a black line using infrared sensors and can also be remotely directed using a smartphone via Bluetooth commands.

The motivation behind this project is to explore basic autonomous navigation concepts while integrating wireless control systems and mechatronic principles. This system provides a practical foundation for learning embedded systems, robotics, and motor control techniques.

2 System Overview

The developed robotic system is a mobile vehicle capable of operating in two main modes: autonomous line-following and manual control via Bluetooth. The system is built around an Arduino Uno microcontroller, which serves as the central unit responsible for processing sensor inputs and controlling the motors.

The robot uses two infrared (IR) sensors placed at the front to detect and follow a black. In line-following mode, the sensor readings determine the robot's direction by adjusting the speed and direction of the motors accordingly.

In manual mode, a Bluetooth module (HC-05) is used to receive directional commands from a smartphone application. Based on the received character commands (e.g., 'f' for forward, 'r' for right), the Arduino controls the motors to move the robot in the desired direction.

The motion is achieved using two DC motors connected through an L298N motor driver, which allows for speed and direction control using PWM signals. The robot's frame was designed in CAD software and manufactured using CNC cutting for precision.

This dual-mode functionality allows flexibility in operation and provides a platform for testing both autonomous behavior and remote control integration in embedded systems.

The robot is powered by two 3.7V lithium-ion batteries connected in series, delivering a total of 7.4V. The power is routed through a physical power switch that allows the user to safely turn the system on and off. This voltage is sufficient to drive the DC motors via the L298N motor driver and also power the Arduino microcontroller.

The use of rechargeable lithium batteries enables the robot to operate wirelessly and makes it suitable for extended testing and mobile applications.

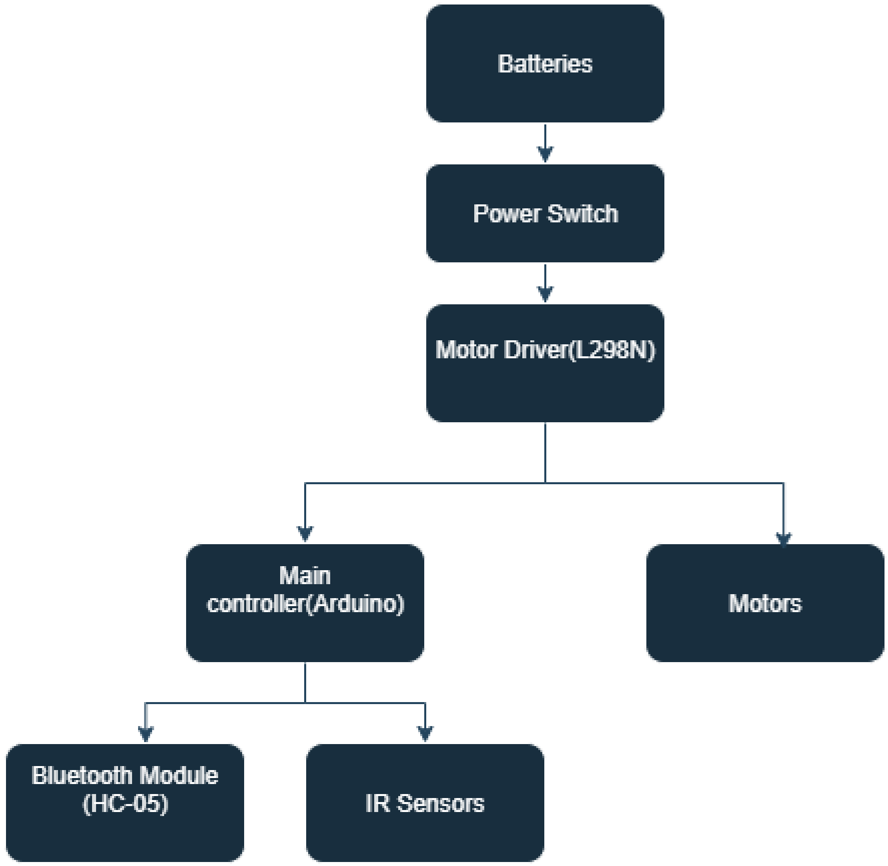


Figure 1 System Block Diagram

3 Hardware Components

The robotic system consists of several hardware components working together to enable movement, control, and sensing capabilities. Below is a brief description of each component:

3.1 Power Supply Unit

The robot is powered by two 3.7V lithium-ion batteries connected in series to deliver 7.4V. This voltage powers both the Arduino and the motor driver. A physical power switch is included to safely turn the robot on and off.

3.2 Arduino UNO

The Arduino UNO microcontroller(aTmega328p) acts as the brain of the robot. It receives sensor data, processes incoming Bluetooth commands, and controls the motor driver accordingly.

3.3 Motor Driver (L298N)

The L298N motor driver is used to control the speed and direction of two DC motors. It receives PWM signals from the Arduino and drives the motors accordingly. It also provides power regulation to the motors.

3.4 DC Motors

Two 6V DC motors are used for locomotion. They are connected to the motor driver and provide forward and backward movement, as well as turning capability.

3.5 IR Sensors (TCRT5000)

Two TCRT5000 infrared sensors are used for line detection. Each sensor consists of an infrared emitter and receiver, and is capable of detecting the difference between dark and light surfaces. The sensors are connected to the analog pins A0 and A1 of the Arduino for continuous feedback during line-following operation.

3.6 Bluetooth Module (HC-05)

The HC-05 module provides wireless serial communication between the robot and a smartphone. It operates using the UART protocol through the Arduino's TX and RX pins, allowing real-time command transmission.

3.7 Chassis and Frame

The chassis was custom-designed and manufactured using CNC cutting. It provides mounting space for all components and ensures mechanical stability.

3.8 Wheels and Mounts

The motors are attached to plastic wheels using motor mounts. The robot uses two rear wheels for drive and a front caster wheel for support and stability.

3.9 Power Switch

A toggle power switch is used to control the flow of electricity from the batteries to the rest of the system.

Table 1 List of Hardware Components Used

No.	Component	Quantity
1	Arduino UNO R3	1
2	L298N Motor Driver	1
3	DC Motors	2
4	IR Sensor 5-channel (TCRT5000)	1
5	Bluetooth Module (HC-05)	1
6	3.7 Li-ion Batteries	2
7	Power Switch	1
8	Wheels	2
9	Caster Wheel	1
10	Chassis Plate	1
11	Jumper Wires	~20
12	Battery Holder	2

4 Mechanical Design

This section discusses the mechanical aspects of the robot, including the chassis layout, material selection, component placement, and physical assembly process.

4.1 Chassis Overview

The robot chassis was designed to be compact and structurally stable, providing a solid foundation for all electronic and mechanical components. A symmetrical design was adopted to maintain balance during both autonomous (line following) and manual (Bluetooth-controlled) movement. The base included mounting points for motors, sensors, power modules, and control electronics.

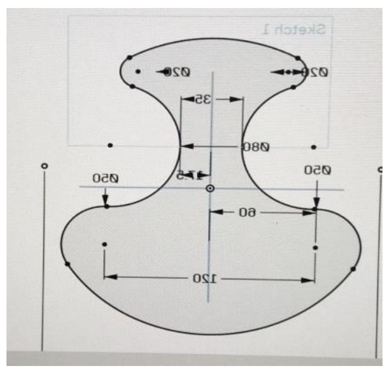


Figure 2 Frame_V1

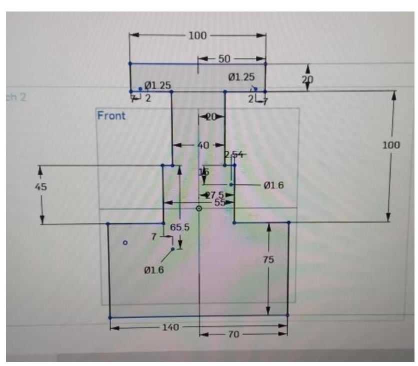


Figure 3 Frame_V2

4.2 Material & Fabrication

The chassis was fabricated using a lightweight wooden sheet MDF with an estimated thickness of 2.5 mm. This material was chosen due to its availability, ease of processing, and adequate mechanical strength for the application. The initial design was hand-sketched and later transferred to CNC for precision cutting, ensuring accurate positioning of component slots and mounting holes.

4.3 Component Placement

The layout was designed to ensure optimal weight distribution and wiring efficiency. The two DC motors were mounted on the rear sides of the chassis and directly connected to the rear wheels. A caster wheel was installed at the front to provide smooth turning. The IR sensors were mounted on the front edge, close to the ground, to effectively detect line contrast. The Arduino board, motor driver, and Bluetooth module were centrally located to allow organized cable management. The battery pack was placed toward the rear to balance the structure.

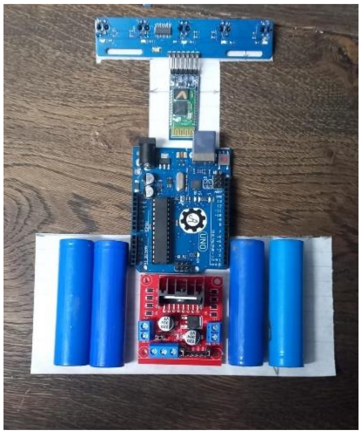


Figure 4 Component Placement

4.4 Final Mechanical Assembly

The final structure assembled all mechanical and electronic components securely onto the wooden chassis. Mounting was done using screws, glue, or adhesive tapes where necessary. The result was a compact, durable platform suitable for testing and deployment in both line-following and Bluetooth-controlled modes.

Figure 5 Assembly_V1

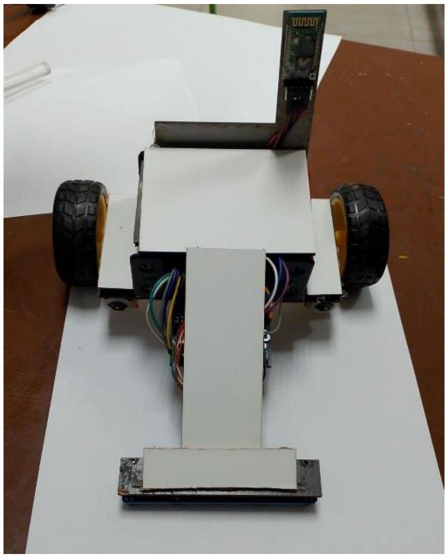


Figure 6 Assembly_V2

5 Wiring & Circuit

This section outlines how the electronic components were connected within the robot, including power distribution and communication between modules. The robot features a simple yet functional circuit that integrates sensors, actuators, a power source, and a microcontroller. All components were connected using jumper wires on the chassis, with the wiring layout designed to minimize electrical noise, ensure stability, and maintain easy access for testing and debugging.

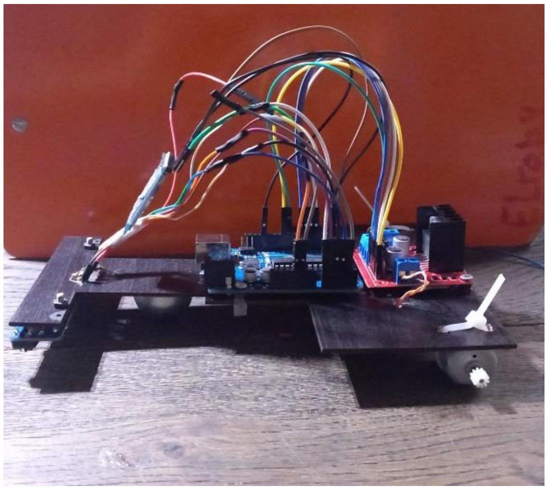


Figure 7 Wiring_1

5.1 Power Distribution

Power was supplied through two 3.7V lithium-ion batteries connected in series to provide 7.4V. The battery pack powered the Arduino board through its VIN pin, which also distributed power to the motor driver and sensors. A power switch was added between the battery and the circuit to safely toggle the system on and off.

5.2 Signal Connections

The Arduino UNO served as the central control unit. It received analog input from the IR sensors through pins A0 and A1. Motor control signals were sent to the L298N motor driver via digital pins 6, 7, 8, and 9, while PWM signals were provided through pins 5 and 10 for speed regulation. The HC-05 Bluetooth module was connected via the hardware UART interface (TX/RX), enabling serial communication for manual control.

5.3 Circuit Diagram

A full wiring diagram is included below to visualize the connections between the components.

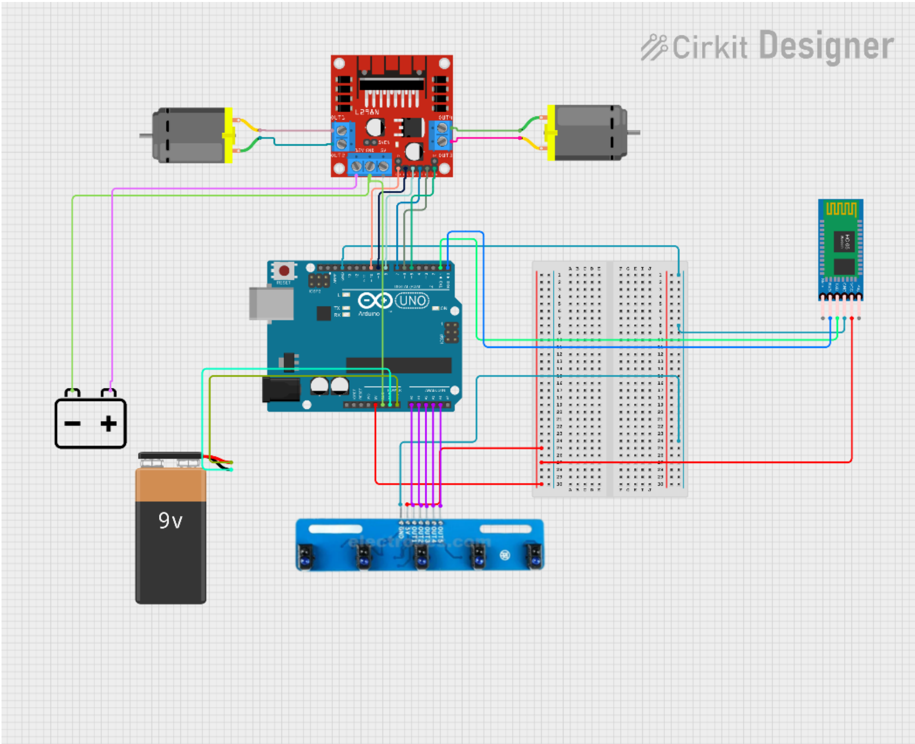


Figure 8 Circuit Diagram

5.4 Wiring Challenges

During assembly, attention was given to correct polarity, avoiding short circuits, and managing overlapping wires. Some early challenges included unstable motor power due to loose connections and minor interference between the Bluetooth module and motor driver, which were mitigated by rearranging wire paths and securing all headers properly.

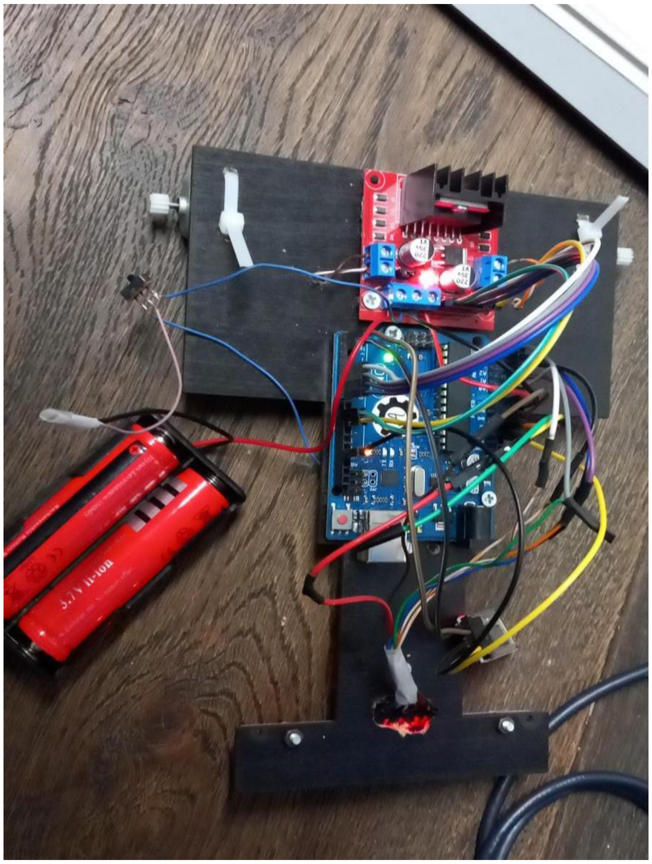
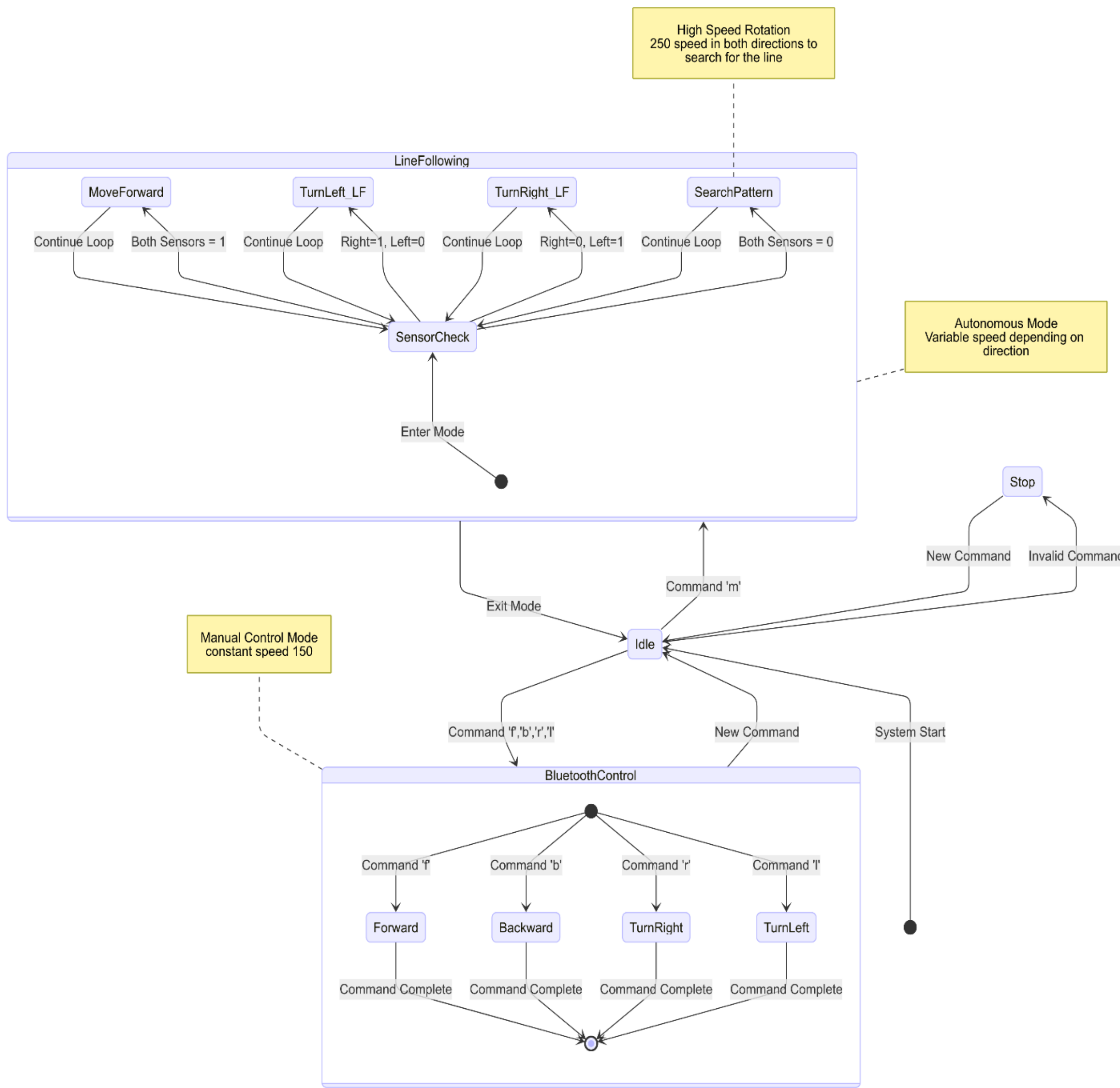


Figure 9 Wiring_2

6 Software

The robot's behavior was controlled through an Arduino-based embedded program that handled two main operation modes: Bluetooth remote control and autonomous line following. The system was programmed using the Arduino IDE in C/C++, and structured into modular functions for each movement type.



6.1 Mode Handling

The robot supported two modes:

- Bluetooth Mode: Controlled by a mobile phone or external device via an HC-05 Bluetooth module. Specific characters (e.g., 'f', 'b', 'l', 'r') were sent over serial communication and parsed by the Arduino to trigger corresponding movement functions.

- Line Following Mode: Activated by sending the character 'm'. The robot used two infrared sensors to track a black line on a white surface.

6.2 Sensor Logic

Two TCRT5000 IR sensors were connected to analog pins A0 and A1. The logic was simple but functional:

- Both sensors on white → move forward.

- One sensor on black, the other on white → steer accordingly.

- Both sensors off the line → perform a turning/search pattern to relocate the path.

Although a PID (Proportional-Integral-Derivative) controller was initially planned for more accurate and smooth line following, time constraints limited the implementation. Preliminary tests with PID logic showed potential, but the team relied on threshold-based logic in the final submission due to the approaching deadline.

6.3 Motor Control

Motor control was handled using an L298N H-Bridge. Four digital pins were used for direction control (in1–in4), while two PWM pins (enA and enB) managed motor speeds. Separate functions were implemented for forward, backward, left, and right movements in both manual and autonomous modes. Speed values were tuned manually after several trial runs to ensure balance and symmetry between the motors.

6.4 Bluetooth Communication

The HC-05 module was connected to the Arduino's UART (TX/RX) interface. A simple character-based protocol was used to receive commands and control the robot in real time. The Arduino continuously monitored the serial buffer and mapped incoming characters to movement commands.

6.5 Limitations and Future Improvements

While the implemented logic was sufficient for basic control, there were some limitations:

- The lack of a PID controller limited the line-following performance, especially on sharp turns or noisy backgrounds.

- The Bluetooth control relied on fixed-speed values, making it difficult to fine-tune the robot's behavior in real time.

Future improvements would include integrating a full PID loop for the line follower, adding real-time speed adjustment via Bluetooth, and possibly introducing sensor calibration routines for better accuracy.

7 Testing & Results

The robot was tested in both manual Bluetooth control and autonomous line-following modes. Individual hardware components such as motors, sensors, and the Bluetooth module were first tested independently. Full system testing was then carried out by integrating all modules and executing sequences of commands. The robot was operated over various line patterns to assess its stability, responsiveness, and adaptability.

7.1 Test Environment

All tests were conducted indoors on a smooth white surface with a black track made of electrical insulation tape. Lighting was kept constant to reduce the interference of ambient infrared sources. The robot was powered using two 3.7V lithium-ion batteries, and test sessions lasted approximately 10–15 minutes each.

7.2 Observations & Issues

- The robot accurately responded to Bluetooth commands including movement and directional control.

- Manual control offered smooth performance at constant speed but lacked feedback.

- In autonomous mode, the robot successfully followed straight and slightly curved paths.

- The line sensors were occasionally affected by ambient light, especially direct sunlight.

- Mode switching between manual and autonomous required a short delay (~0.5 seconds) to stabilize.

7.3 Results Summary

Table 2 Results Summary

Test Case	Result	Remarks
Bluetooth forward/backward	Success	Smooth movement and instant response
Bluetooth turning (left/right)	Success	Minor overshoot on sharp turns
Line following - straight path	Success	Maintained stable tracking
Line following - gentle curves	Success	Adjusted direction smoothly
Line following - sharp curves	Partial	Needs further tuning
Sensor response in dim light	Success	Accurate line detection
Sensor response in bright light	Partial	Occasional misreadings
Mode switching (manual - auto)	Success	Required short transition delay

8 Challenges

Throughout the development of the robot, the team encountered several challenges that shaped the project both technically and mentally.

8.1 Planning & Scope Management

One of the most significant challenges was the initial lack of focused planning. At the beginning, we aimed to build an extremely advanced robot — something unique and ambitious that would stand out from other projects. Ideas such as a dual-mode system with high-end features, complex mechanical structures. However, this ambition quickly led to confusion, overthinking, and a lack of clear direction. The scope kept expanding, while the remaining time kept shrinking. Eventually, we recognized the need to pause, reset, and focus on a minimal viable solution a working robot that fulfills the required functionalities. We made the conscious decision to rebuild from scratch and then improve step-by-step as time allowed.

This shift in mindset was one of the most valuable lessons of the entire project: Good planning and realistic goal setting are far more important than just big ideas.

8.2 Technical Difficulties

- The line-following algorithm was originally meant to be based on PID control. However, due to time constraints and limited tuning opportunities, we had to fall back on a simpler threshold-based approach.

- IR sensors occasionally misread the track under strong lighting conditions, requiring careful setup of the testing environment.

- Mechanical alignment and chassis stability were affected by slight inaccuracies in material cutting and sensor positioning.

8.3 Team & Time Pressure

The project timeline shifted unexpectedly. Coordination among team members was also challenging at certain stages due to overlapping responsibilities and external academic pressure. Nevertheless, communication improved significantly in the second half of the project.

9 Conclusion

This project aimed to develop a dual-mode robotic vehicle capable of operating via both manual Bluetooth control and autonomous line following. Despite the challenges faced in planning, time constraints, and hardware limitations, the team successfully delivered a functional prototype that meets the basic project requirements.

The robot was able to navigate tracks using infrared sensors and respond accurately to Bluetooth commands. Although some of the initially intended features such as PID control were not fully implemented due to time limitations, the system proved stable, responsive, and adaptable to further development.

Beyond the technical outcomes, this project provided valuable lessons in teamwork, realistic planning, and project execution. The shift from overly ambitious designs to a focused and functional system taught us the importance of prioritizing core goals before pursuing enhancements.

Overall, the project was a successful learning experience and serves as a solid foundation for future improvements and more advanced robotic systems.

10 Bibliography

- Arduino. (n.d.). Getting Started with Arduino. Retrieved from https://www.arduino.cc/en/Guide

- HowToMechatronics. (n.d.). HC-05 Bluetooth Module Tutorial. Retrieved from https://howtomechatronics.com/?s=HC-05

- Paul Horowitz & Winfield Hill. (2015). *The Art of Electronics* (3rd ed.). Cambridge University Press.

- Simon Monk. (2016). *Programming Arduino: Getting Started with Sketches*. McGraw-Hill Education.

- Michael Margolis. (2012). *Arduino Cookbook: Recipes to Begin, Expand, and Enhance Your Projects*. O'Reilly Media.

- John Boxall. (2013). *Arduino Workshop: A Hands-On Introduction with 65 Projects*. No Starch Press.

- instructables. (n.d.). TCRT5000 IR Sensor Tutorial. Retrieved from https://www.instructables.com/TCRT5000-Infrared- Reflective-Sensor-How-It-Works-a/

- Howtomechatronics. (n.d.). Interfacing L298N Motor Driver with Arduino. Retrieved from https://howtomechatronics.com/tutorials/arduino/arduino-dc-motor-control-tutorial-l298n-pwm-h-bridge/