Design and Implementation of a Line Following and Obstacle Avoidance Robot

MD Rafiul Islam*, MD Azadul Islam*, Umut Faruk Karakaya*
*Systems Engineering and Prototyping, BSc Electronic Engineering
Hochschule Hamm-Lippstadt, Lippstadt, Germany
{rafiul.islam, azadul.islam, umut.karakaya}@stud.hshl.de

***Abstract*—This paper presents the design, development, and testing of an autonomous robot that can follow a black line on a white surface and avoid obstacles detected within its path. The robot is built using Arduino and multiple sensors including infrared (IR), ultrasonic, and color sensors. The project integrates hardware prototyping, systems engineering principles, and timed behavior modeling using UPPAAL. The robot adjusts its behavior based on environmental feedback, such as light variation and battery voltage, enhancing robustness and real-world usability.**

***Index Terms*—Line follower robot, obstacle avoidance, Arduino, sensor fusion, autonomous system, prototyping, UPPAAL, systems engineering.**

I. Introduction

Autonomous navigation systems are rapidly becoming critical in robotics. This project aims to implement a small-scale, autonomous mobile robot capable of line following and intelligent obstacle avoidance. The development follows a structured engineering approach, combining system modeling, electronic design, simulation, and real-world testing.

II. System Design and Engineering

The foundation of the robot was based on systems engineering principles. Initially, use case and requirement diagrams were created using SysML to define the functional and non-functional aspects of the robot. These models were refined into block definition and internal block diagrams which provided a modular structure. Each module—sensing, control, and actuation—was individually designed for clear interface and responsibility.

III. Behavioral Modeling using UPPAAL

To validate the robot's timing and safety behaviors, formal models were created using UPPAAL. These timed automata diagrams describe the interaction of sensors and actuators over time, including decision delays and critical thresholds for obstacle detection.

IV. Prototyping and Design

A. Electronic Components

The robot uses the following key components:

- Arduino Uno microcontroller
- L298N motor driver
- 2 IR sensors for line detection
- Ultrasonic sensor for distance measurement
- Servo motor for scanning obstacles

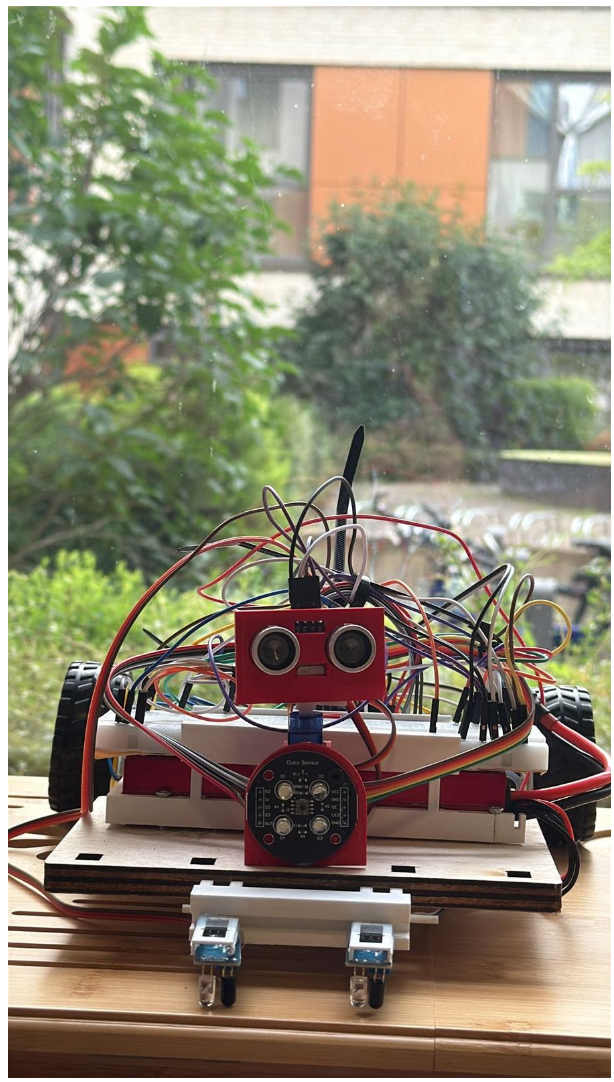

Fig. 1. Front view of the autonomous robot during testing

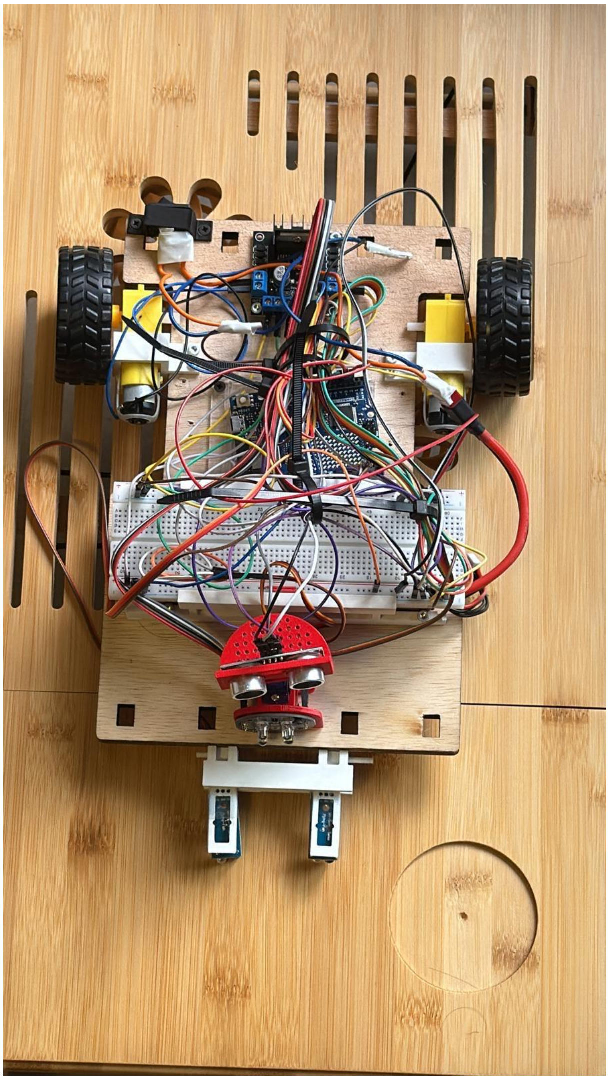

Fig. 2. Side angle showing full robot assembly

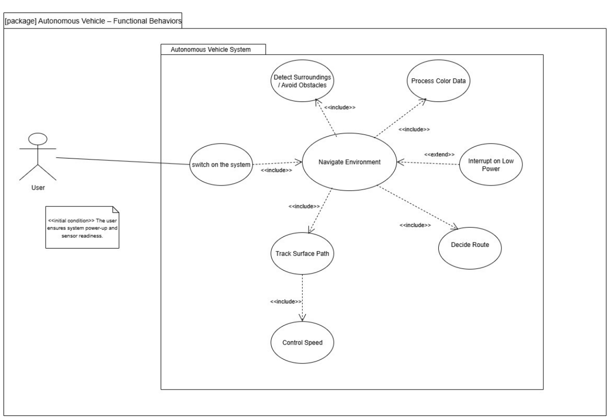

Fig. 3. SysML Use Case Diagram

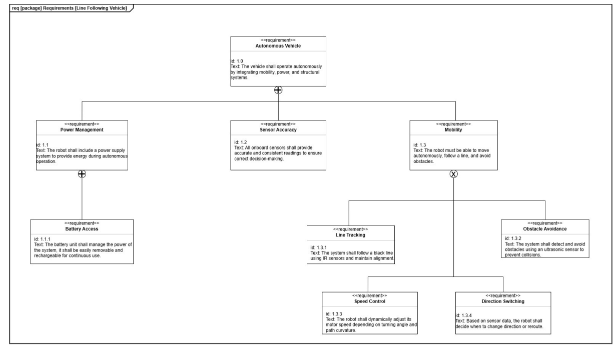

Fig. 4. SysML Requirements Diagram

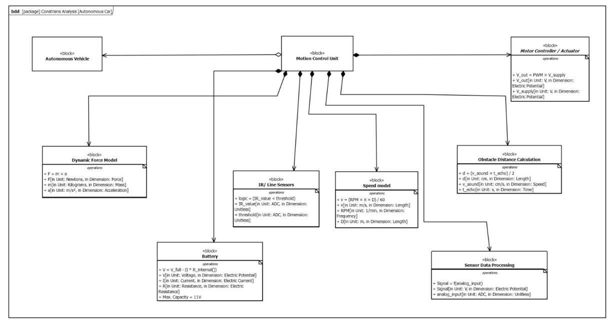

Fig. 5. SysML Constraints Diagram

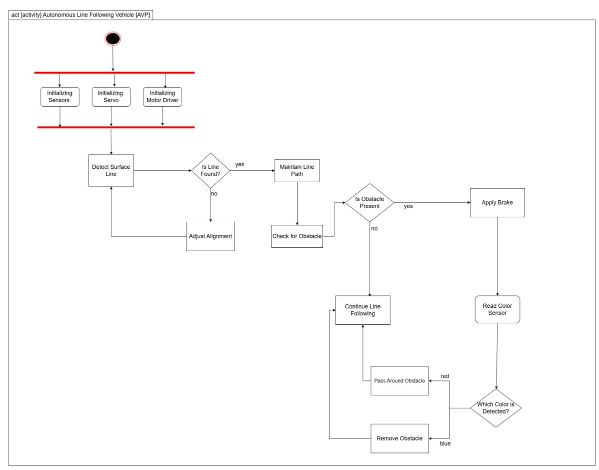

Fig. 6. SysML Activity Diagram

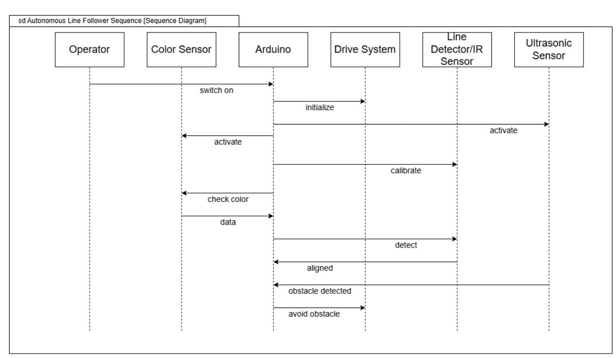

Fig. 7. SysML State Machine Diagram

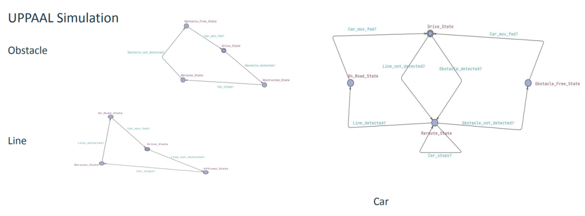

Fig. 8. UPPAAL model of line-following and state transition

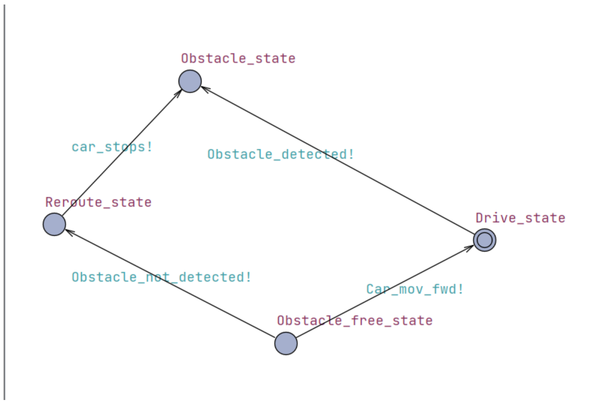

Fig. 9. UPPAAL obstacle avoidance decision automaton

- TCS3200 color sensor for identifying specific zones
- 2 DC motors for driving wheels
- 12V battery (with 5V regulation for logic circuits)

B. Virtual and Physical Prototypes

Initial logic and wiring were tested using Tinkercad simulation. A physical prototype was developed on a custom chassis. The system was incrementally built and validated using Multisim for circuit design, followed by breadboard implementation and final soldering.

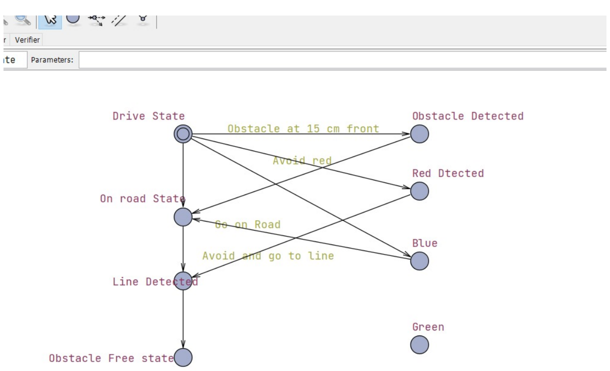

Fig. 10. Updated UPPAAL transition structure

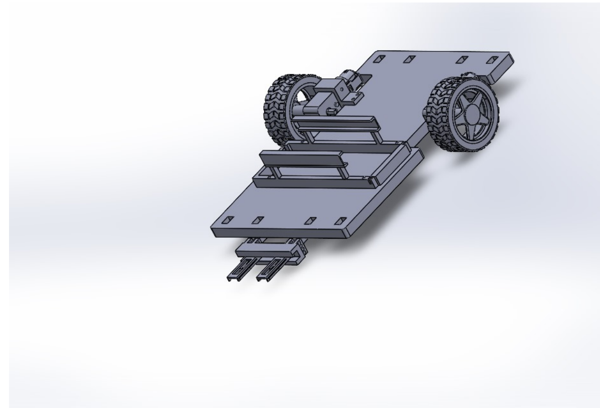

Fig. 11. Physical prototype of the assembled robot

C. Hardware Assembly

All components were mounted securely on the chassis. Connections were made as per schematic, considering voltage compatibility. A switch and power regulation module were added to safely deliver 12V to motors and 5V to logic.

V. Circuit Design

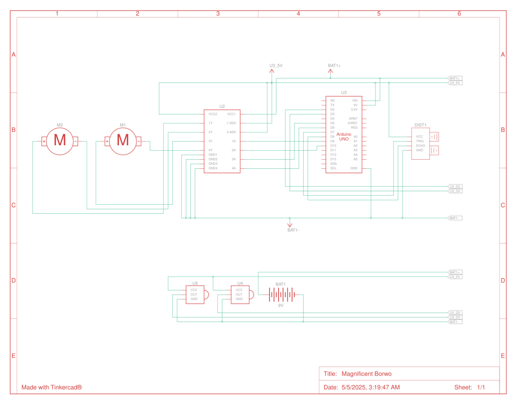

Fig. 12. Complete circuit layout of the robot

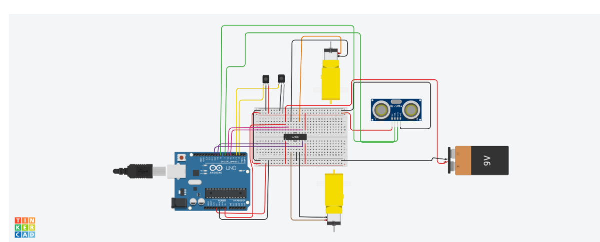

Fig. 13. Tinkercad simulation-based wiring diagram

VI. Software Implementation

The Arduino code integrates line following, obstacle detection, and color recognition. It uses a non-blocking loop, servo

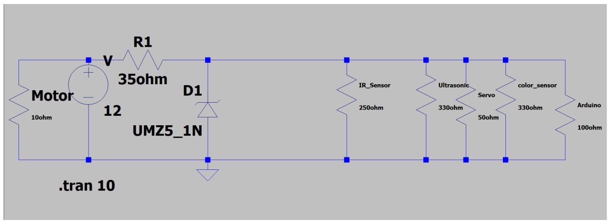

Fig. 14. Voltage regulation and power distribution simulated in LTspice

sweep logic for obstacle scanning, and conditional branching based on sensor input to control motion. Real-time feedback ensures responsiveness and robustness.

VII. Git Usage and Collaboration

Project progress was tracked using GitHub. Branches were created per feature. Each member contributed based on assigned tasks, and all commits were documented with messages.

Project Repository: https://github.com/UmutKarakayaHSHL/D3--prototyping

Team Member	**Number of Commits**
MD Rafiul Islam	53
MD Azadul Islam	53
Umut Faruk Karakaya	53
Total	**159**

TABLE I
GitHub Commits by Team Members

VIII. Key Achievements

- Developed a fully functional autonomous robot
- Validated timing behavior using UPPAAL timed automata
- Created reusable, modular code with real-time sensor feedback
- Demonstrated robust performance under various voltage and light conditions
- Successfully collaborated using GitHub with task-wise contributions

Acknowledgment

We thank Prof. Dr. Stefan Henkler and the Prototyping staff at HSHL for guidance throughout the project.

References

References

[1] Arduino Documentation. https://www.arduino.cc/en/Guide
[2] Tinkercad Simulation. https://www.tinkercad.com
[3] Uppaal Model Checker. http://www.uppaal.org/
[4] LTspice - Analog Circuit Simulator. https://www.analog.com/en/design-center/design-tools-and-calculators/ltspice-simulator.html

Declaration of Originality

We, the undersigned, hereby declare that this report titled *Design and Implementation of a Line Following and Obstacle Avoidance Robot* is entirely our own work. All sources and external contributions have been properly cited and acknowledged. No part of this document has been plagiarized.

- MD Rafiul Islam
- MD Azadul Islam
- Umut Faruk Karakaya