islington college
(इस्लिङ्टन कलेज)

LONDON METROPOLITAN UNIVERSITY

# CS5054NI Advanced Programming and Technologies

## 50% Group Coursework
## Submission: Milestone 1

## Academic Semester: Spring Semester 2025
## Obstacle Avoiding Robot
## Credit: 15 credit semester long module

| Group Name: | | | |
|---|---|---|---|
| SN | Student Name | College ID | University ID |
| 1 | Mandeep Chhetri (N7) | Np01nt4a230145 | 23047466 |
| 2 | Prasun Lamichhane (N7) | Np01nt4a230139 | 23047465 |
| 3 | Unisha Khadgi (N7) | Np01nt4a230155 | 23047475 |
| 4 | Tenji Sherpa (N7) | Np01nt4a230174 | 23048523 |
| 5 | Samarpan Khadka (N7) | Np01nt4a230214 | 23047551 |

**GitHub Link:** https://srituhobby.com/how-to-make-a-multi-function-arduino-robot/

**Assignment Due Date: May 15, 2025**
**Assignment Submission Date: May 15, 2025**

***I confirm that I understand my coursework needs to be submitted online via MST Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded.***

# 23047466 MandeepChhetri L2N7 Team2.docx

Islington College,Nepal

## Document Details

Submission ID

trn:oid:::3618:96000877

Submission Date

May 15, 2025, 12:30 PM GMT+5:45

Download Date

May 15, 2025, 12:31 PM GMT+5:45

File Name

23047466 MandeepChhetri L2N7 Team2.docx

File Size

26.0 KB

24 Pages

3,827 Words

22,347 Characters



# 6% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

## Match Groups

- 15 Not Cited or Quoted 5%
  Matches with neither in-text citation nor quotation marks
- 2 Missing Quotations 1%
  Matches that are still very similar to source material
- 0 Missing Citation 0%
  Matches that have quotation marks, but no in-text citation
- 0 Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

## Top Sources

3% Internet sources

2% Publications

5% Submitted works (Student Papers)

## Integrity Flags

**0 Integrity Flags for Review**

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

## Acknowledgement

We would like to extend our gratitude to our teachers- Mr. Sugat Man Sakhya and Mr. Bishnu Pandey for the priceless guidance, support and encouragement provided for the course of this project. Their knowledge and positive suggestions played a critical role in developing our perception and carrying out the IoT-based Obstacle Avoidance Car as well.

We also appreciate the resource department of Islington College catering with essential components and tools required in the project. What is more, we are also grateful to our group members for cooperation, commitment, and labor, which were crucial for the project's goal realization.

Finally, we recognize the support of different online resources, open-source platforms, and Arduino community, whose materials and tutorials have greatly helped us in learning and implementing our project.

## Abstract

The IoT-based Obstacle Avoidance Car project envisages the development and deployment of an autonomous system that will be able to navigate environments while avoiding obstacles. Using an Arduino Uno microcontroller, ultrasonic sensors, servo motors, and gear motors, the car senses obstacles at real-time and changes its route accordingly. The system comprises of hardware components including L293D motor driver shield and the lithium-ion batteries and software tools such as Arduino IDE, Cirkit Designer. Under strenuous testing, the car proved to detect and avoid obstacles at the range between 2 to 100 cm, with response time smaller than 0.5 seconds. This work is an educational tool used to connect the theories as it relates to use in robotics and IoT technologies. Services that can be implemented in the future are advanced sensors and machine learning algorithms that increase the performance in dynamic environments.

# Table of Contents

## TABLE OF FIGURE

## Table of tables

# 1 Introduction

The term IoT represents the interconnected device framework alongside the communication infrastructure that ties devices with the Internet and other devices and the cloud platform. Beneath the combination of inexpensive computer chips and quick telecommunications technology billions of devices now maintain internet connectivity. These everyday items including toothbrushes and vacuums and cars and machines can obtain information by using sensors that also provide smart responses to users. (AWS, 2020)

Regular things achieve connection with the internet through the Internet of Things. Over the last two decades of the 1990s Computer Engineers started embedding processors along with sensors inside regular household items. The slow development stemmed from their bulky and large nature during the early stages of advancement. Initially RFID tags were used as low-power computer chips to track valuable equipment. (AWS, 2020)

Businesses use IoT devices for monitoring key indicators that include temperature, humidity, air quality and both energy consumption and equipment operation status. Companies immediately access examined data to identify operational patterns and irregularities which help them boost their profitability and operational improvements. (IBM, May 2023)

## 1.1 Current Scenario

In 2025, contemporary robotics and automation systems significantly depend on autonomous navigation for their functionality. Autonomous navigation systems with detection and obstacle avoidance capabilities function in manufacturing settings, support delivery of drones and vehicles, and provide robotic education. Devices that feature obstacle avoidance capabilities function independently in changing environments, thereby improving safety and increasing the dependability of the system.

Advancements in modern sensor technology enable robots to achieve improved environmental awareness through three key developments: ultrasonic devices, LiDAR systems, infrared sensors, and computer vision capabilities. Current technology poses challenges in adopting these platforms because of their limited-scale needs without compromising affordability, which primarily impacts students, researchers, and hobbyists. The growing need for cost-effective obstacle-detection vehicles utilizing simple sensors alongside Arduino and Raspberry Pi microcontrollers has become evident.

The car project that avoids educational obstacles has become increasingly important as students grasp essential principles of embedded systems alongside automation and artificial intelligence programming concepts through this initiative. These projects replicate real system applications, encompassing autonomous delivery robots and smart navigation techniques, all while offering both academic and practical benefits. (Kuhn, 2023)

## 1.2 Problem Statement and Project as a solution

### 1.2.1 Problem Statement

The operational efficiency of autonomous machines relies significantly on their ability to avoid obstacles, as robotics and automation are becoming increasingly prominent in daily life. Fundamental robotic systems encounter difficulties due to challenges in identifying reliable and affordable methods for obstacle detection. Most autonomous systems available require costly and sophisticated sensors such as LiDAR, computer vision, and radar to function. Educational organizations and basic automation fans encounter a challenge due to restricted access to technological resources.

The issue of equipment damage rises when educational or experimental robotic vehicles collide with objects due to their absence of integrated navigation and detection systems. The robots that move through Amazon warehouse shelves employ systems that prompt them to steer clear of both cramped shelf spaces and active machinery. Contemporary systems utilize advanced sensors in conjunction with algorithms for functionality. These solutions continue to be unfeasible for school and DIY projects as they demand skills that most students or DIY enthusiasts cannot afford or obtain.

Creating a budget-friendly obstacle avoidance system necessitates a sensor-driven approach utilizing Arduino and ultrasonic sensors. This approach would make the link between theoretical studies and practical application feasible by enabling beginners to observe the real-time decision-making processes of intelligent systems. (Taner Ince, 2022)

### 1.2.2 Project as a Solution

An Obstacle Avoiding Car acts as the primary solution, providing autonomous navigation around obstacles through its cost-effective smart robotic design. The automobile utilizes ultrasonic sensors located at the front to carry out ongoing object detection. The uno microcontroller analyzes data from detecting nearby objects within the defined distance range before guiding the car in a new direction for safety purposes.

The system is made up of essential parts that connect Arduino boards with ultrasonic sensors, steering servo motors, and gear motors for movement. A real-time path modification function has been integrated into the system for efficient decision-making with distance detection abilities.

This project is like current robotic devices such as vacuum cleaners (like Roomba) that move through homes independently, avoiding furniture, walls, and other different obstacles. Through this obstacle-avoiding car project, students can grasp the basics of robotics and automation, as it replicates the underlying operational principles on a smaller scale and at lower costs. (.Kuhn, 2023)

# 2 Aim

This project is aimed at building and implementing an IoT-based system **(Obstacle Avoidance Car)** that will support effective monitoring and automatic control of environmental and operational factors with the perspective of increasing productivity, safety, and quality of decision-making.

# 3 Objective

- To design an IoT framework that can retrieve and follow data in real-time
- To integrate sensors and microcontrollers for satisfying requirements of both environmental as well as operational data acquisition.
- To build a secure and efficient wireless protocol for the transport of data.
- To build cloud storage infrastructure for logging and data retrieval.
- To build an intuitive interface – for convenient data visualization and analysis.
- For assessment of system reliability, scalability and cost effectiveness during performance evaluation.
- The protection of the data and the user's privacy by imposing the necessary security control

# 4 Background

## 4.1 System Overview

The obstacle avoiding robot is an autonomous system which detects obstacles using an ultrasonic sensor and avoids collisions by adjusting its movement according to its surroundings. To implement this system, several tools and devices were used and arranged in a specific way to allow the robot to navigate by itself without human intervention. This system helps users understand the basic principles of autonomous robotics and intelligent motion control.

Various components were used to properly build the robot, It includes an Arduino Uno that acts as the main microcontroller board to control all the functionality of the system, a L293D Motor Driver Shield mounted on the Arduino to control the robot's motors, a HC-SR04 Ultrasonic Sensor detects obstacles and measures distance, a Servo Motor to rotate the ultrasonic sensor for scanning the environment around it, Four Gear Motors attached to Robot Wheels to allow movement in all directions, Lithium-Ion Batteries to provide power to the entire system, and finally, Jumper Wires to connect all the components together without the need for soldering.

## 4.2 Design Diagrams

Several diagrams were created to properly understand the working principle of the project which were a Hardware architecture diagram, a schematic diagram, a circuit diagram and a flowchart diagram which are all provided below.

### 4.2.1 Hardware architecture diagram

In this diagram the basic hardware configuration for the obstacle avoiding robot is shown with all its main components. At the center of the system is the Arduino Uno which acts as the primary controller and is responsible for processing data and sending signals throughout the robot. It receives input from the Ultrasonic Sensor which detects obstacles in the environment and uses that information to make decisions for movement. Based on the sensor readings, the Arduino sends commands to the motor driver shield, which then controls the movement of the Gear Motors attached to the robot wheels, enabling it to avoid any detected obstacles.

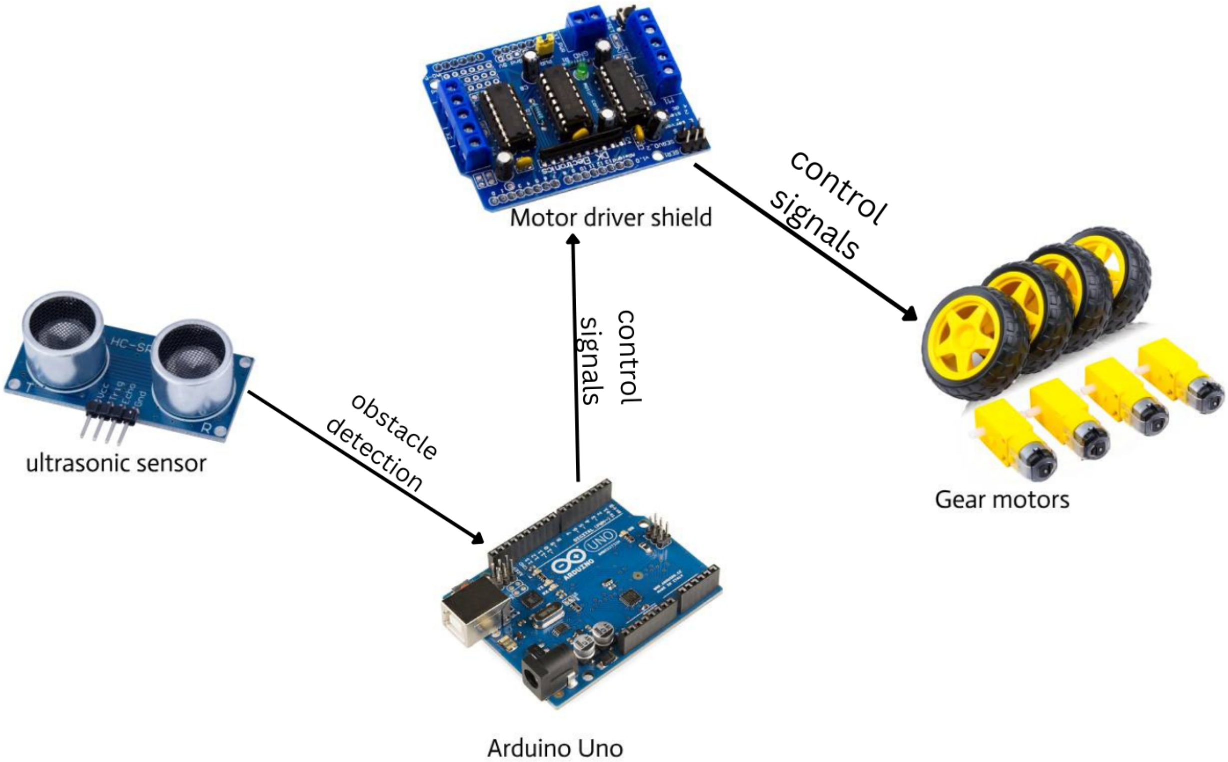


*Figure 1 Hardware architecture diagram*

### 4.2.2 Block Diagram

A block diagram is a visual representation of a system, project, or scenario which provides a functional view of a system and shows how the different elements of that system connect to each other. These types of diagrams are used by engineers to understand how components of a system are connected. (Miro, 2025)

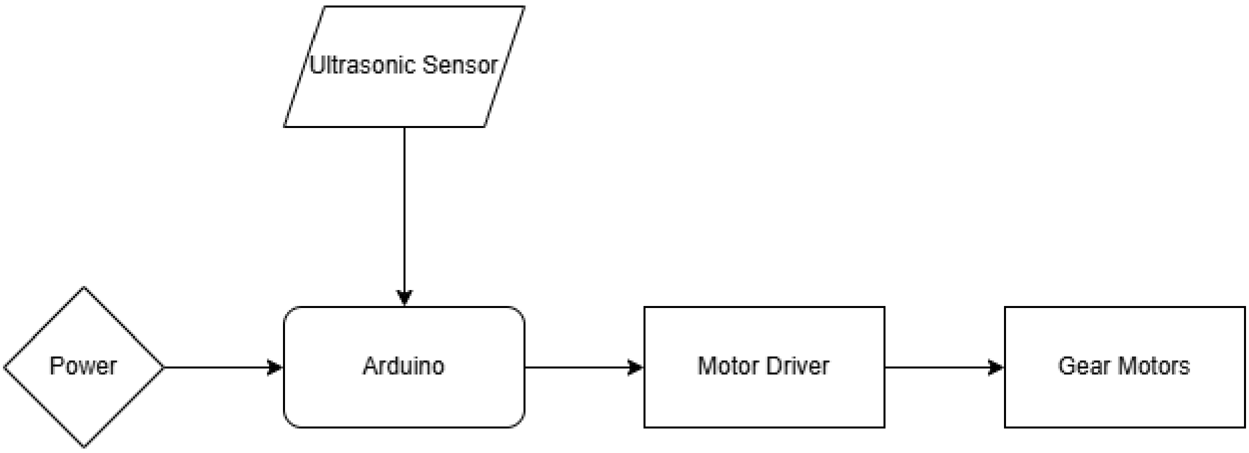


*Figure 2 Block Diagram*

### 4.2.3 Circuit diagram

A circuit diagram is a visual representation of the various parts of an electrical circuit. It is widely utilized in the design, maintenance, and repair of electrical and electronic equipment, assisting in understanding how various components are interrelated. These diagrams are used to troubleshoot and develop both basic and complicated electrical systems. (TestBook, 2025)

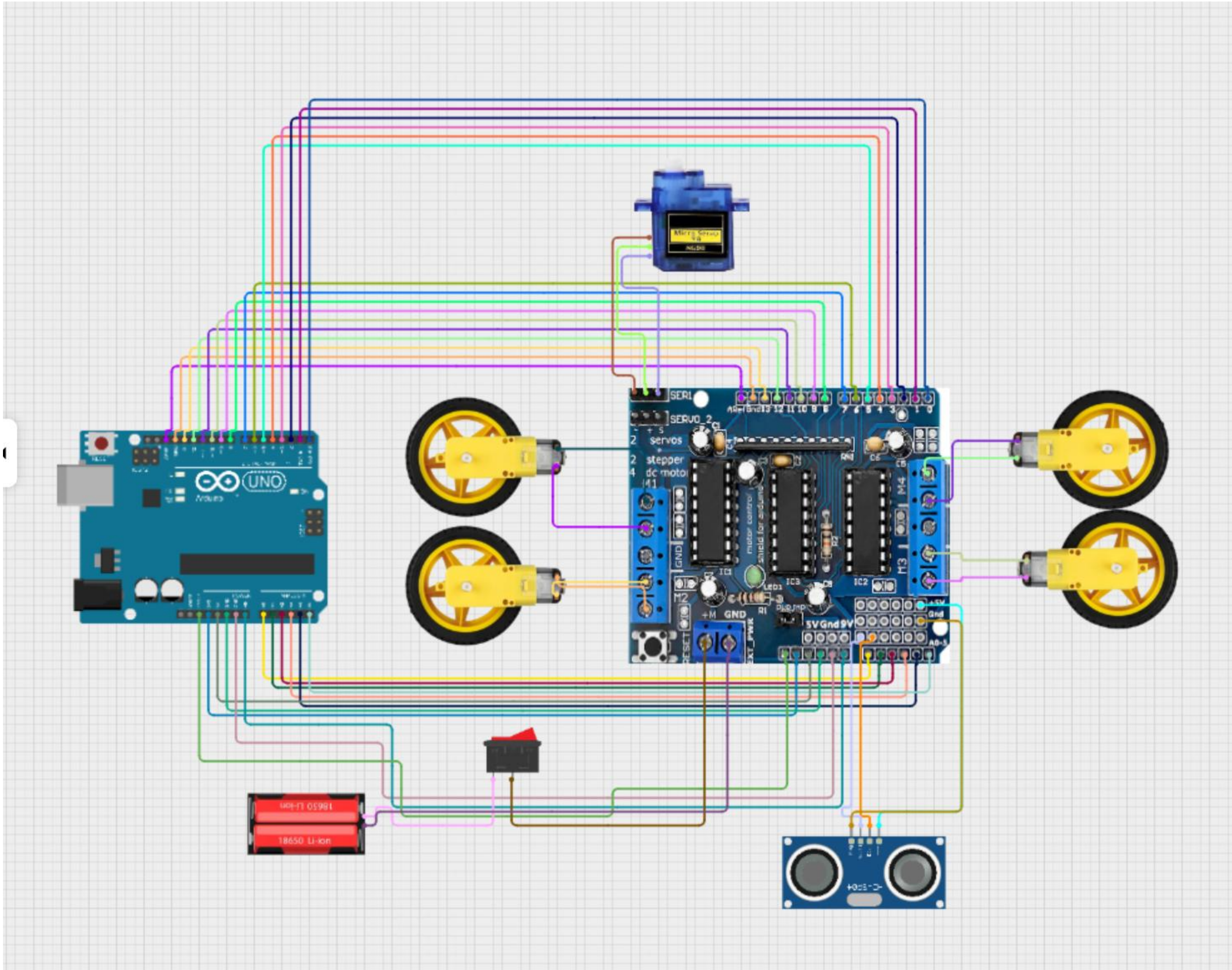

*Figure 3 Circuit diagram*

### 4.2.4 Schematic diagram

A schematic diagram is a diagram that uses symbols to represent the essential components of a system, device, or process. It shows critical components and their connections by using standardized symbols and lines. While it may also include added details for better understanding. (Lim, 2024)

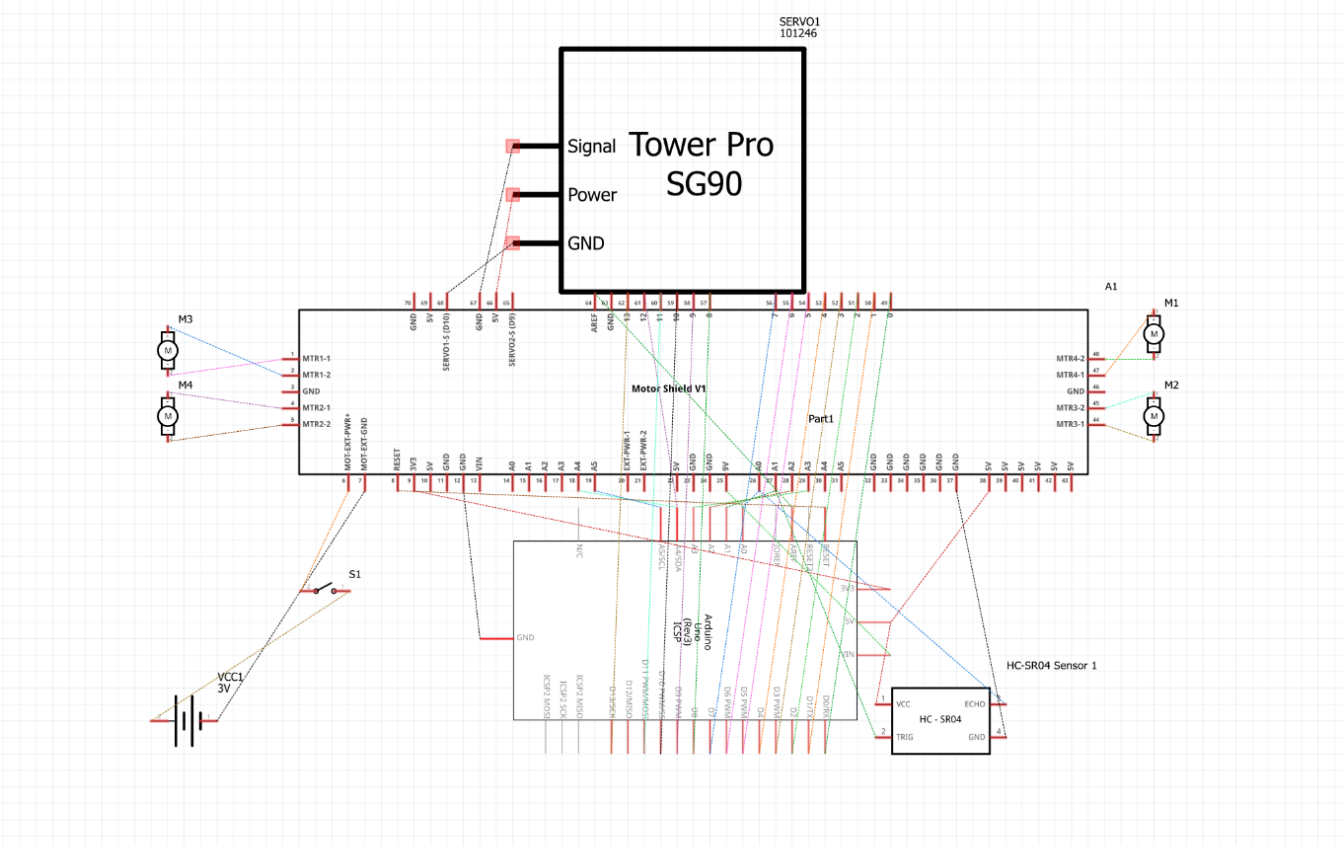


*Figure 4 Schematic diagram*

### 4.2.5 Flowchart diagram

A flowchart is a diagram that shows how a process, system, or computer algorithm works in a visual format. It is frequently used in numerous fields to document, examine, plan, develop, and convey complicated processes in a clear and easy manner. (Lucidchart, 2025)

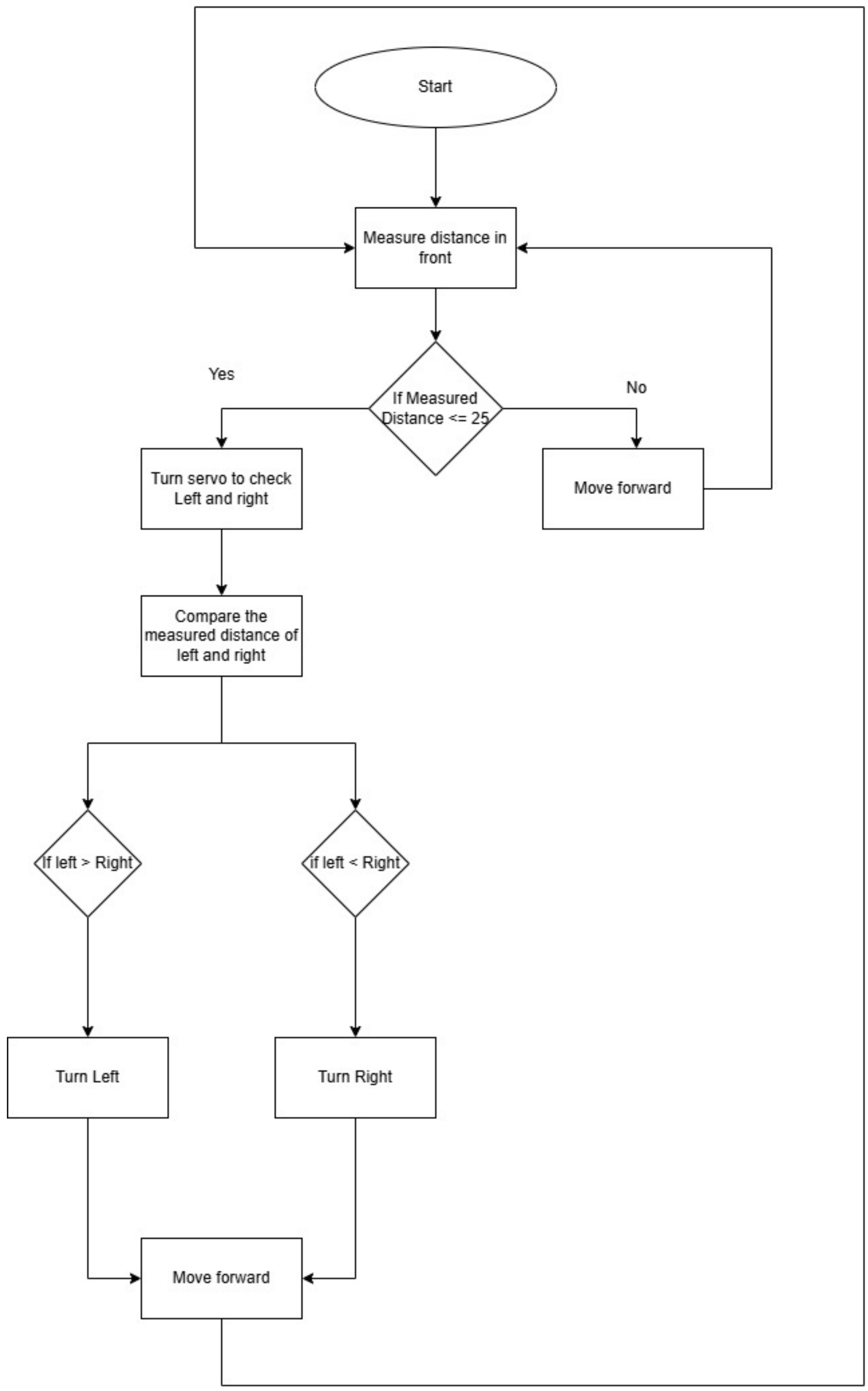


*Figure 5 Flowchart of the Obstacle avoiding car*

## 4.3 Requirement analysis

### 4.3.1 Hardware

**i. Arduino UNO**:

The Arduino Uno is an open-source microcontroller board that runs on the ATmega328P processor. There are 14 digital I/O pins, 6 analog inputs, a USB connector, a power jack, an ICSP header, and a reset button. It contains all the modules required to support the microcontroller (flybro, 2019)

*Figure 6 Arduino uno*

**ii. L239D motor driver shield**:

It is an add-on board for Arduino that allows you to control the direction and speed of DC motors and stepper motors. (Last Minute engineers, 2025)

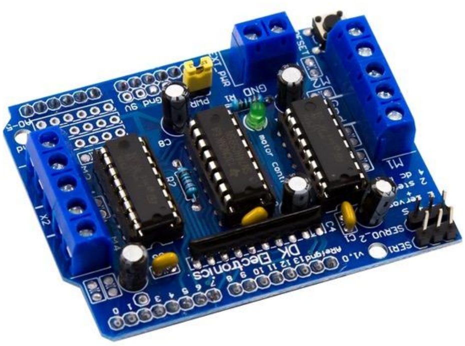

*Figure 7 L239D Motor Driver Shield*

**iii. HC SR04**:

The HC-SR04 ultrasonic sensor serves as a popular solution when measuring distances. The device sends out an ultrasonic pulse before measuring how long the echo takes to return so it can compute distance. This device can detect objects accurately within ranges of up to 4 meters and finds applications in parking sensors together with robotics obstacle avoidance systems and interactive displays. (AP Monitor, 2024)

*Figure 8 HC SR04*

**iv. Servo Motor**:

A servo motor is a motor that can precisely regulate its angular position. It contains a feedback control circuit for correctly monitoring and adjusting the position of the motor shaft. Servo motors are often utilized when precise angle or distance movement is required. They use a servo mechanism and are often classed as DC or AC servo motors depending on their power source. (circuit digest, 2025)

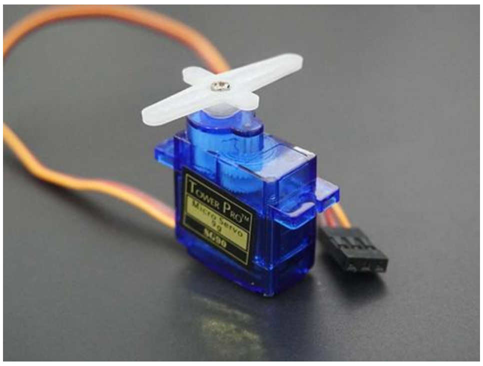

*Figure 9 Servo Motor*

**v. Gear Motors**:

A gear motor combines an electric motor with a gearbox to slow down operation speed while boosting torque which increases its performance at necessary speeds. The gearbox modifies the motor output to satisfy application specifications. The design coupled with its adjustable configuration features enables it to function widely throughout various sectors. (micromotors, 2022)

*Figure 10 Gear motor*

**vi. Robot Wheels**:

It allows the robot to move when attached to the motor.

*Figure 11 Robot wheels*

**vii. Lithium Ion Battery**:

A lithium-ion battery is a rechargeable battery that works by transferring lithium ions between the anode (negative) and cathode (positive) when charging and discharging. Unlike primary batteries, lithium-ion batteries may be used again, making them suitable for portable gadgets and electric vehicles. (Toshiba, 2025)

*Figure 12 Lithium Ion Battery*

### viii. Lithium-Ion Battery Holder:

The main purpose of battery holders is to secure electrical cells inside their compartments while supplying energy to applications. Developers can embed these units directly into electrical devices, but they also provide external housing or appendages to the original product. They offer great value to consumer devices because they simplify battery replacement. (SAFT, 2023)

*Figure 13 Lithium-Ion Battery Holder*

**ix. Jumper Wires**:

Jumper wires are electrical wires with connector pins at both ends which enable them to connect two points through their ports without applying heat for soldering. The purpose of jumper wires in breadboard and other prototyping setups is to create flexible circuit connections through their pins. (Hemmings, 2018)

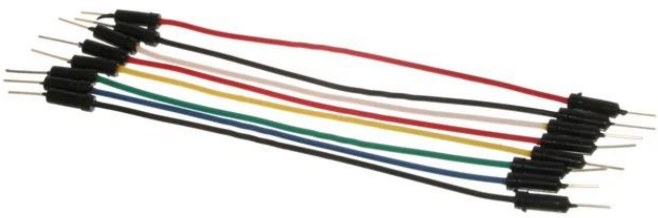

*Figure 14 Jumper wires*

### 4.3.2 Software:

**i. Arduino IDE**:

The Arduino IDE serves as a software platform which permits users to create modify or upload programming code to control Arduino boards. The IDE provides users with a text editor and chat section and terminal interface alongside a toolbar containing shortcut buttons and display menus. Through the IDE users establish a connection with Arduino hardware to both transfer code and operate their board. (Arduino Docs, 2024)

*Figure 15 Arduino IDE*

**ii. Cirkit Designer**:

Cirkit Designer is a development platform that includes an Arduino IDE (based on VS Code), allowing users to create, build, and upload code to Arduino boards right in the environment. Its user-friendly design and guided lessons make it suitable for students, enthusiasts, and professionals. (Cirkit Designer, 2025)

*Figure 16 Cirkit Designer*

### iii. Draw.io

Draw.io is software that allows you to easily create diagrams and charts. It provides both automated layout functionality and the ability to create bespoke layouts, all backed by a big library of shapes and hundreds of graphic components. (Computer Hope, 2024)

*Figure 17 Draw.io*

### iv. Fritzing

Fritzing is an open-source platform that provides a software tool, community website, and related services to make electronic applications accessible for creative use. It enables users to design, document, and share electronic prototypes. It is used to teach electronics in educational settings, and create professional PCBs. (Fritzing, 2025)

*Figure 18 Fritzing*

# 5 Development

## 5.1 Planning and Design

To complete the prototype of obstacle avoidance, car major planning and design was conducted. Ultrasonic sensor is used in front of the car to detect object on the way and servomotor helps ultrasonic sensor to rotate and change the route if any object is detected and after finding an alternative route the cars continue to move, 3.7v Battery provide power to operate and gear motors used to make the wheels rotate. L293D motor driver shield was installed on the Arduino UNO and all the components were connected to the driver shield.

## 5.2 Resource Collection

Many components were used to complete this project. Two-thirds of the components were provided by the resource department of Islington College, and the remaining components were bought by our team.

Components provided by the Resource department are:

- Arduino Uno
- Jumper Wires
- Ultrasonic Sensor
- Gear motor
- Servo motor
- Robotic wheels

Components bought by the team members are:

- L293D motor driver shield
- Li-ion Battery holder
- Li-ion battery

## 5.3 System Development

### 5.3.1 Phase 1:

In the first phase Arduino was connected to l293d driver sheild and the connected to the laptop and it was checked if it was operating correctly.

Figure 19 Connecting Arduino uno to laptop.

### 5.3.2 Phase 2:

In the second phase the micro controller Arduino Uno was attached to the middle of the frame and L293D driver motor shield was attached to the top of Arduino UNO and connections were done for the ultrasonic sensor.

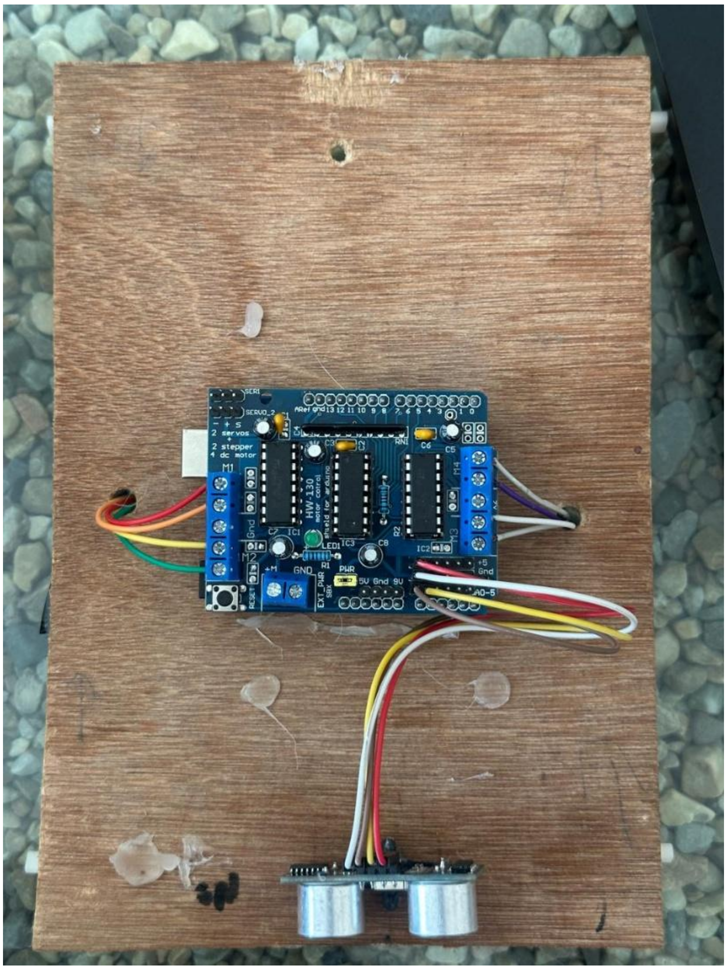

*Figure 20 Connecting ultrasonic sensor to driver motor.*

### 5.3.3 Phase 3:

At the third phase gear motors were attached in the bottom of frame and robotic wheels were connected in the gear motors. Server motor was installed at the front of the car and ultrasonic sensor was mounted at the top of the servo motor. Battery holder was attached which can hold two 3.7 volts battery that can provide power to operate the car. A switch was connected to toggle the power supply.

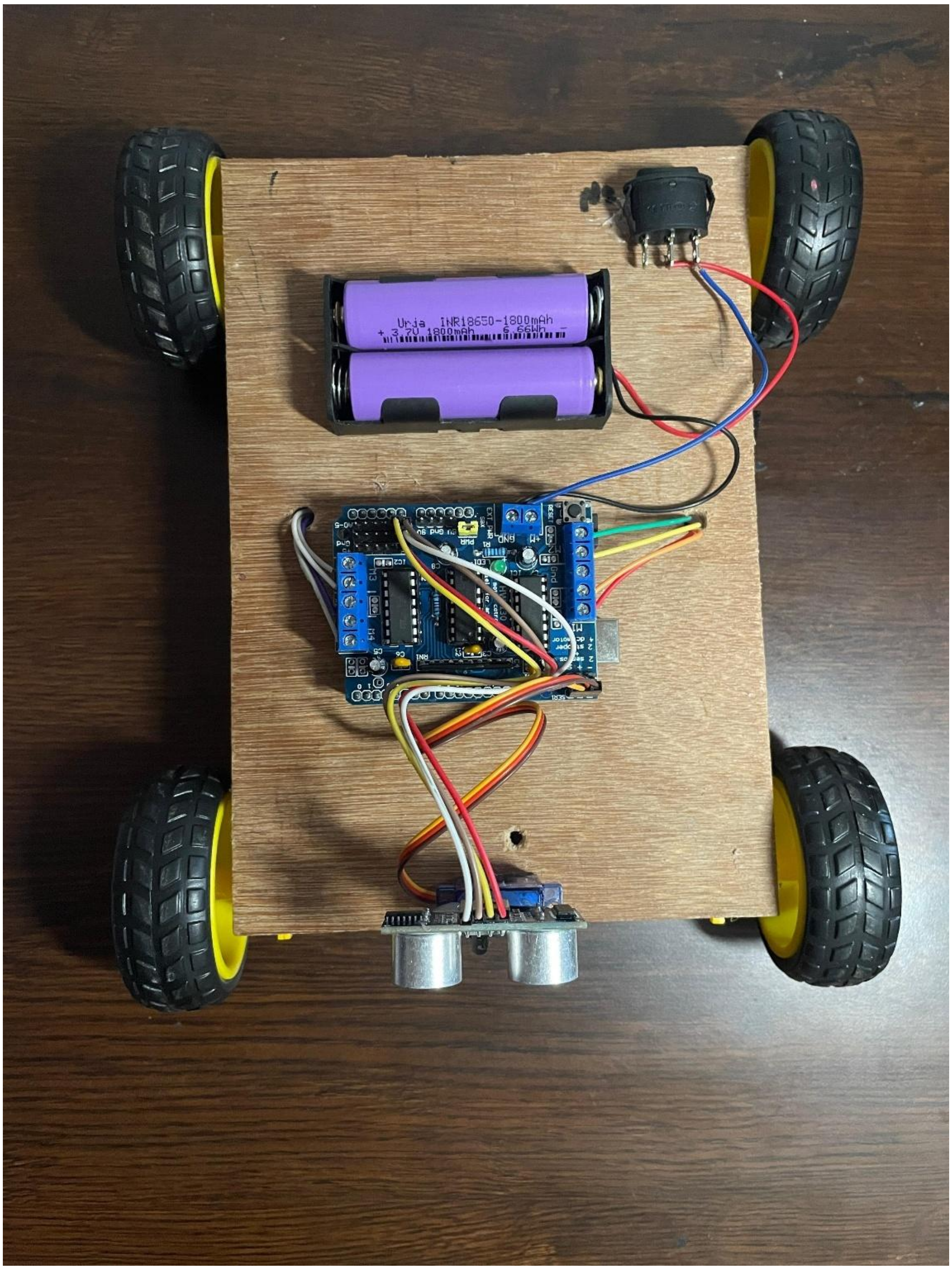


*Figure 21 Obstacle avoidance car after all hardware connections.*

### 5.3.4 Phase 4:

At the last phase program was uploaded to Arduino after all hardware were connected.

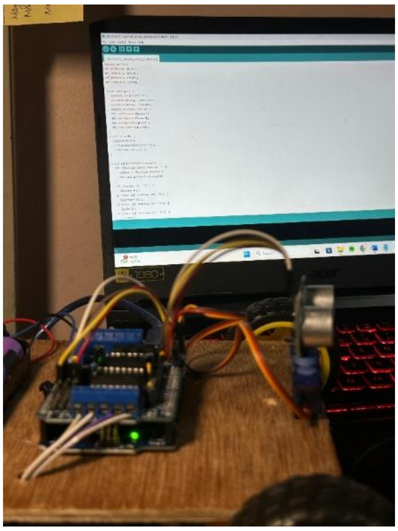

*Figure 22 Uploading program to the device.*

### Connections made in the driver motor shield.

#### Ultrasonic sensor

| | |
|---|---|
| ECHO | A1 |
| TRIG | A0 |
| GND | GND |
| VCC | VCC |

*Table 1: Connection of ultrasonic sensor.*

#### Servo motor

| | |
|---|---|
| GND (black) | GND |
| VCC (red) | VCC |
| SIG (yellow) | 10s |

*Table 2: Connection of servomotor.*

#### Motors and Batteries

Motors and batteries are connected to driver shields terminals.

# 6 Results and Findings

Avoiding the obstacles, the Arduino robot's ability to identify and avoid obstacles was successfully tested in a variety of controlled conditions. The robot used an Arduino Uno-connected ultrasonic sensor (HC-SR04) to assess distances in real time and modify its course as necessary. The robot successfully identified impediments between 2 and 100 cm during testing, and it responded by changing course to prevent collisions. Navigation was seamless and timely since the detection to action reaction time was continuously less than 0.5 seconds. The robot went uninterrupted in wide areas and could make precise directional judgments to avoid several obstacles in congested surroundings. With steady sensor readings and steady movement, the system operated dependably in typical interior settings. These outcomes validate the effectiveness of the deployed system by showing that the Arduino-based robot can function independently and effectively in settings with possible difficulties.

## 6.1 Testing

### 6.1.1 Test 1: To compile and upload

| | |
|---|---|
| **Test** | 1 |
| **Objective** | To verify successful code compilation and uploading to Arduino Uno. |
| **Activity** | Compiled and uploaded code using Arduino IDE. |
| **Expected Result** | Code compiles and uploads without errors. |
| **Actual Result** | Code compiled and uploaded successfully. |
| **Conclusion** | Compilation and upload process works correctly. |

*Table 3: Test 1.*

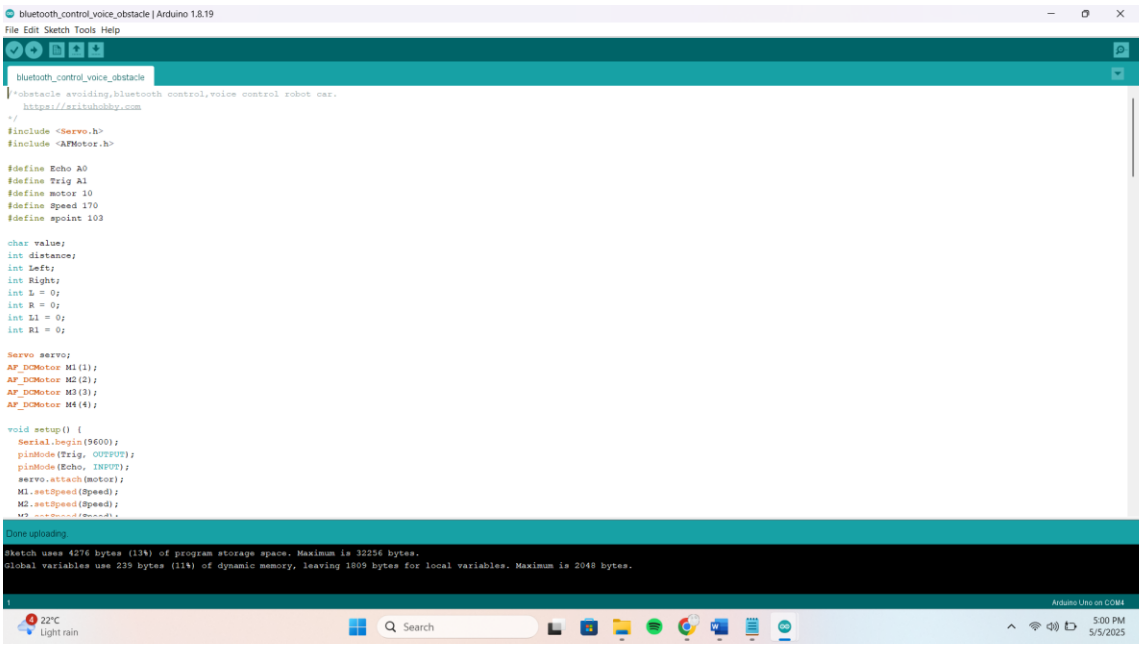


*Figure 23 Screenshot of Arduino IDE uploading the code successfully.*

### 6.1.2 Test 2: Ultrasonic Sensor Functionality

| | |
|---|---|
| **Test** | 2 |
| **Objective** | To check if the ultrasonic sensor detects obstacles accurately. |
| **Activity** | Placed objects at various distances and observed readings. |
| **Expected Result** | Sensor detects and displays distance accurately. |
| **Actual Result** | Accurate distance readings observed. |
| **Conclusion** | Ultrasonic sensor works as expected. |

*Table 4: Test 2.*

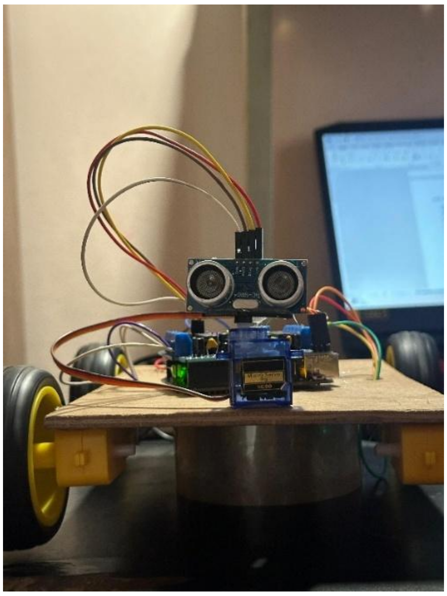

*Figure 24 Wheels rotating before ultrasonic sensor detecting an object.*

*Figure 25 Wheels stopped after ultrasonic sensor detected an object.*

### 6.1.3 Test 3: Servo Motor Movement

| | |
|---|---|
| **Test** | 3 |
| **Objective** | To test servo motor rotation for directional changes. |
| **Activity** | Sent angle commands to servo motor (0°, 90°, 180°). |
| **Expected Result** | Servo rotates to given angles. |
| **Actual Result** | Servo rotated smoothly and correctly. |
| **Conclusion** | Servo motor functions properly. |

*Table 5: Test 3.*

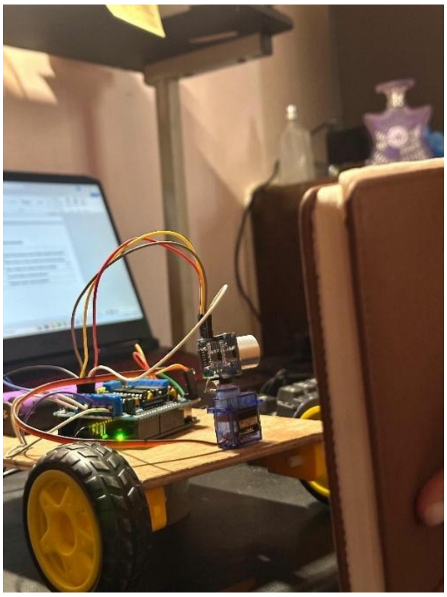

*Figure 26 Servo motor rotating to right.*

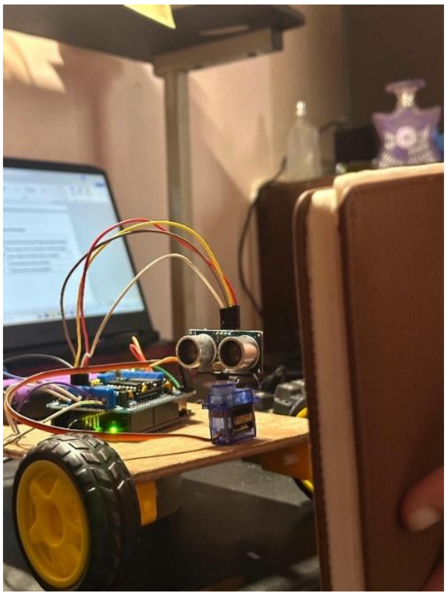

Figure 27 Servo motor rotating to left

### 6.1.4 Test 4: Car Stops After Detecting Obstacle

| | |
|---|---|
| **Test** | 4 |
| **Objective** | To verify robot stops when obstacles are close. |
| **Activity** | Placed object 25 cm in front of moving robot. |
| **Expected Result** | The robot stops immediately. |
| **Actual Result** | Robot stopped upon detecting obstacle. |
| **Conclusion** | Stop function is working correctly. |

Table 6: Test 4

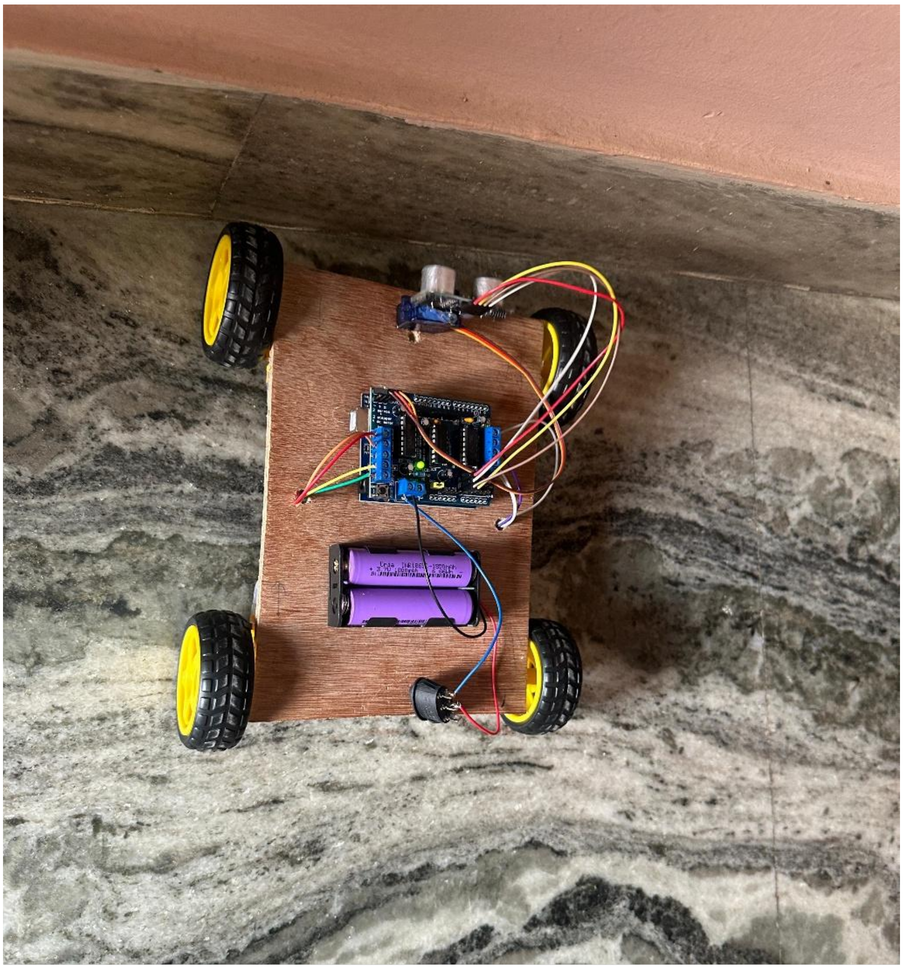

Figure 28 Picture of the car stopping after detecting an object.

### 6.1.5 Test 5: Car Changes Direction After Detecting Obstacle

| | |
|---|---|
| **Test** | 5 |
| **Objective** | To check if the robot changes direction after detecting obstacle. |
| **Activity** | Robots moved toward obstacles and scanned surroundings. |
| **Expected Result** | The robot stops and turns in a clear direction. |
| **Actual Result** | Robot turned to avoid obstacle. |
| **Conclusion** | Direction changes logic works correctly. |

*Table 7: Test 5*

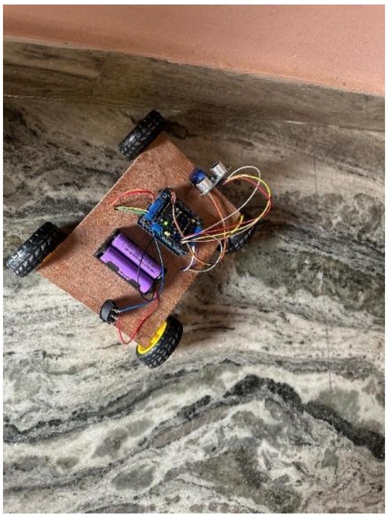

*Figure 29 Car changing its route after detecting an object.*

*Figure 30 Car changing its route after detecting an object continue.*

### 6.1.6 Test 6: Inaccuracy of Ultrasonic Sensors in Narrow Space Detection

| | |
|---|---|
| **Test** | 6 |
| **Objective** | To see if the robot can use the ultrasonic sensor to identify and avoid small obstacles. |
| **Activity** | The robot was instructed to move through spaces where there were narrow spaces between obstructions. |
| **Expected Result** | The robot should recognize the little area and choose a different route to avoid collision. |
| **Actual Result** | The robot ran into barriers because it was unable to detect small areas. |
| **Conclusion** | Due to the ultrasonic sensor's inability to identify small spaces, the robot may overestimate safe routes and collision. It could be necessary to make further logic or sensor changes. |

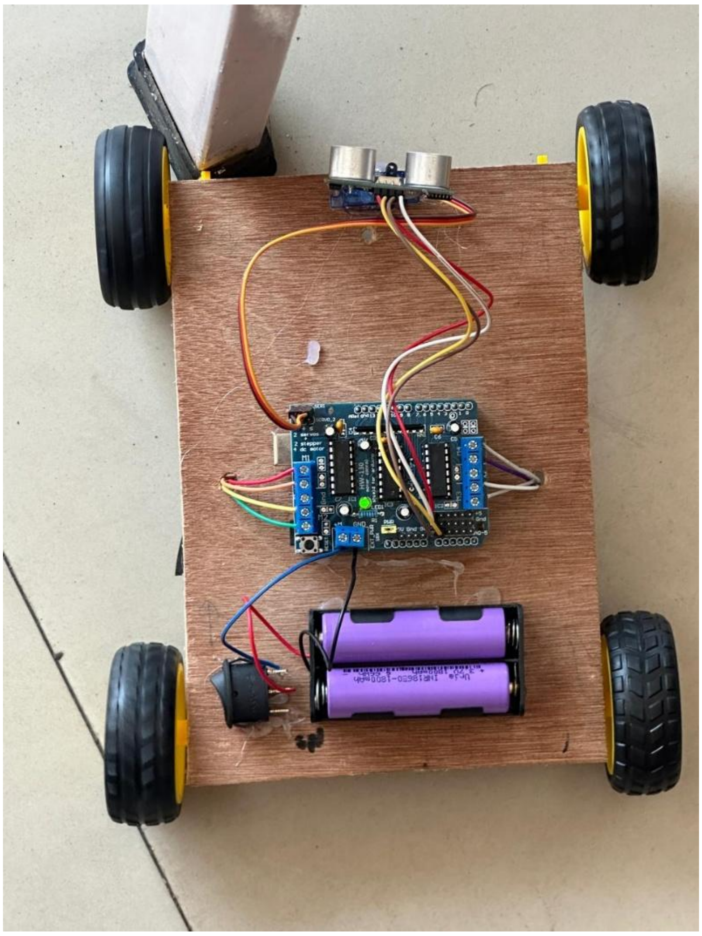

*Figure 31: Wheels crashed.*

# 7 Future Works

The obstacle avoidance system offers a lot of expansion opportunities into real-world application to different industries. The autonomous navigational abilities of the platform can make it evolve to become a specialized robotic solution to urgent needs in healthcare, logistics, and public services.

In an environment of hospitals, the system can develop into an automatic cleaning and sanitization robot. The robot would be able to autonomously traverse patient wards, operation theaters and high traffic areas with UV sterilization modules or disinfectant spraying mechanisms integrated. The current barrier detection scheme would guarantee normal operation within the vicinity of medical equipment, beds, and working around moving personnel. Such implementation can always drastically lower hospital-acquired infections with little exposure of people to contaminated areas. The modular design of the platform would make it possible to customize the platform for various healthcare settings, including large hospitals, as well as outpatient clinics.

Common segments might find value in modified applications that would be used within warehouses and storefronts. In distribution centers, augmented with inventory tracking abilities, the robots would be able to autonomously move the goods while it would dynamically avoid other obstacles including workers, forklifts, and racks for storage. Retail space can integrate the technology as smart shopping helpers that could assist shoppers while lugging the supplies in a crowded shop. The exact navigation of the system would avoid collisions with shoppers, as well as display fixtures, thus enhancing customer and operational efficiency.

The platform could be tailored to agricultural applications through greenhouse monitoring and precision spraying. Armed with specialized sensors, it was able to move within the pant rows but avoid irrigation systems and workers. The obstacle avoidance ability would be especially useful in rough terrain where the standard automation would fail. Other attachments would allow options such as crop health monitoring, application of pesticides in a selective manner, or hands-free harvesting in tightly-controlled locations.

Another promising field of applications is the public spaces. Based on such technology, airports, shopping malls, and junctions transport could implement sanitization robots. These units could work around the clock to sanitise high touch surfaces whilst not disrupting permanent structures, and temporary obstacles, such as cleaning equipment or maintenance workers. The scheduling capacity of the system could be improved to accommodate operating times of facilities and times of peak travel. Future technical improvements would target the following three areas: sensor fusion, intelligent navigation and connectivity. Using LiDAR alongside ultrasonic sensors would offer more comprehensive environmental mapping, especially in complex environments where there are glass surfaces, or moving obstacles. Machine learning algorithms may allow the system to perform predictive path planning andFeatures to avoid dynamic obstacles more efficiently. Through cloud integration, fleet management abilities would be supported, allowing remote monitoring and capturing of data, and orchestrated control of many units in large facilities. The modular nature of the platform allows for different payload configuration configurations to target different sectors. The swappable modules could convert the base unit from cleaning, delivery, inspection or monitoring functions as required. This flexibility in combination with the demonstrated foundation for obstacle avoiding makes the technology a flexible solution to many automation problems. With the advances, the prototype may grow to become an economically viable product to mitigate efficiency, safety and hygiene concerns in other industries.

# 8 Conclusion

The successful design of the IoT based Obstacle Avoidance Car system shows that smart technologies can be integrated to generate enhanced environmental awareness and high operational efficiency. By bringing together real-time gathering data, effective sensor, and microcontroller synchronization, and protected wireless interaction, the system enables autonomy in navigation, maintaining safety and productivity. Cloud-based data logging and easy to use interface support the seamless data recovery, analysis and following well-informed decision-making.

This technology has great opportunity applications in various fields, such as healthcare, domestic use, industrial automation and help with mobility. Such devices, like the Xiaomi vacuum cleaner, themselves demonstrate how obstacle avoidance increases convenience and efficiency within home automation by mapping as well as cleaning relevant spaces autonomously. Similarly, the IoT-based system will be used for such tasks as the transportation of medical supplies, the aid to the visually compromised, and changing components in the industrial environment. As the methods for path-planning and sensors improve, and machine learning is developed, this system will become a sensible and flexible tool for addressing real-world issues in various environments.

# 9 References


.Kuhn, M. B., 2023. A numerical study of an atomizing jet in a resonant acoustic field. *International Journal of Multiphase Flow,* Volume 167.

AP Monitor, 2024. [Online] Available at: https://apmonitor.com/dde/index.php/Main/UltrasonicDistanceSensor

Arduino Docs, 2024. [Online] Available at: https://docs.arduino.cc/software/ide-v1/tutorials/arduino-ide-v1-basics/

AWS, 2020. *AWS.* [Online] Available at: https://aws.amazon.com/what-is/iot/ [Accessed 4 05 2025].

circuit digest, 2025. [Online] Available at: https://circuitdigest.com/article/servo-motor-working-and-basics

Cirkit Designer, 2025. [Online] Available at: https://www.cirkitstudio.com/index.html

Computer Hope, 2024. [Online] Available at: https://www.computerhope.com/jargon/d/drawio.htm

flybro, 2019. [Online] Available at: https://www.flyrobo.in/blog/what-is-arduino-uno?srsltid=AfmBOopE5NNO0kvAUOLP4z3v5ffnUT8j6eDIyLmVFP87RBB7bU0R2kHI

Fritzing, 2025. [Online] Available at: https://fritzing.org

Hemmings, M., 2018. [Online] Available at: https://blog.sparkfuneducation.com/what-is-jumper-wire

IBM, May 2023. *IBM.* [Online] Available at: https://www.ibm.com/think/topics/internet-of-things [Accessed 4 05 2025].

Kuhn, M. B., 2023. A numerical study of an atomizing jet in a resonant acoustic field. *International Journal of Multiphase Flow,* 167(11), p. 176.

Last Minute engineers, 2025. [Online] Available at: https://lastminuteengineers.com/l293d-motor-driver-shield-arduino-tutorial/#:~:text=The%20L293D%20is%20a%20dual,motors%20or%20two%20stepper%20motors.

Lim, A., 2024. [Online] Available at: https://www.thoughtco.com/what-is-a-schematic-diagram-4584811

Lucidchart, 2025. [Online] Available at: https://www.lucidchart.com/pages/what-is-a-flowchart-tutorial

micromotors, 2022. [Online] Available at: https://www.micromotors.eu/en/what-is-a-gear-motor/

Miro, 2025. [Online] Available at: https://miro.com/diagramming/what-is-a-block-diagram/

SAFT, 2023. [Online] Available at: https://saft.com/en/energizing-iot/battery-holders-what-you-need-know

Taner Ince, N. D., 2022. Weighted Residual NMF With Spatial Regularization for Hyperspectral Unmixing. *IEEE Geoscience and Remote Sensing Letters,* Volume 19.

TestBook, 2025. [Online]
Available at: https://testbook.com/physics/circuit-diagram

Toshiba, 2025. [Online]
Available at: https://www.global.toshiba/ww/products-solutions/battery/scib/about-scib/more/batteryschool/episode1.html#:~:text=A%20lithium-ion%20battery%20is,a%20wide%20effective%20SOC%20range.

# 10 Appendix

## 10.1 Source code

```
/*obstacle avoiding, Bluetooth control, voice control robot car.
*/
#include <Servo.h>
#include <AFMotor.h>
#define Echo A0
#define Trig A1
#define motor 10
#define Speed 170
#define spoint 103
char value;
int distance;
int Left;
int Right;
int L = 0;
int R = 0;
int L1 = 0;
int R1 = 0;
Servo servo;
AF_DCMotor M1(1);
AF_DCMotor M2(2);
AF_DCMotor M3(3);
AF_DCMotor M4(4);
void setup() {
  Serial.begin(9600);
  pinMode(Trig, OUTPUT);
  pinMode(Echo, INPUT);
  servo.attach(motor);
  M1.setSpeed(Speed);
  M2.setSpeed(Speed);
```

```
  M3.setSpeed(Speed);
  M4.setSpeed(Speed);
}
void loop() {
  //Obstacle();
  //Bluetoothcontrol();
  //voicecontrol();
}
void Bluetoothcontrol() {
  if (Serial.available() > 0) {
    value = Serial.read();
    Serial.println(value);
  }
  if (value == 'F') {
    forward();
  } else if (value == 'B') {
    backward();
  } else if (value == 'L') {
    left();
  } else if (value == 'R') {
    right();
  } else if (value == 'S') {
    Stop();
  }
}
void Obstacle() {
  distance = ultrasonic();
  if (distance <= 25) {
    Stop();
    backward();
    delay(100);
```

```
    Stop();
    L = leftsee();
    servo.write(spoint);
    delay(800);
    R = rightsee();
    servo.write(spoint);
    if (L < R) {
      right();
      delay(500);
      Stop();
      delay(200);
    } else if (L > R) {
      left();
      delay(500);
      Stop();
      delay(200);
    }
  } else {
    forward();
  }
}
void voicecontrol() {
  if (Serial.available() > 0) {
    value = Serial.read();
    Serial.println(value);
    if (value == '^') {
      forward();
    } else if (value == '-') {
      backward();
    } else if (value == '<') {
      L = leftsee();
```

```
      servo.write(spoint);
      if (L >= 10 ) {
        left();
        delay(500);
        Stop();
      } else if (L < 10) {
        Stop();
      }
    } else if (value == '>') {
      R = rightsee();
      servo.write(spoint);
      if (R >= 10 ) {
        right();
        delay(500);
        Stop();
      } else if (R < 10) {
        Stop();
      }
    } else if (value == '*') {
      Stop();
    }
  }
}
// Ultrasonic sensor distance reading function
int ultrasonic() {
  digitalWrite(Trig, LOW);
  delayMicroseconds(4);
  digitalWrite(Trig, HIGH);
  delayMicroseconds(10);
  digitalWrite(Trig, LOW);
  long t = pulseIn(Echo, HIGH);
```

```
  long cm = t / 29 / 2; //time convert distance
  return cm;
}
void forward() {
  M1.run(FORWARD);
  M2.run(FORWARD);
  M3.run(FORWARD);
  M4.run(FORWARD);
}
void backward() {
  M1.run(BACKWARD);
  M2.run(BACKWARD);
  M3.run(BACKWARD);
  M4.run(BACKWARD);
}
void right() {
  M1.run(BACKWARD);
  M2.run(BACKWARD);
  M3.run(FORWARD);
  M4.run(FORWARD);
}
void left() {
  M1.run(FORWARD);
  M2.run(FORWARD);
  M3.run(BACKWARD);
  M4.run(BACKWARD);
}
void Stop() {
  M1.run(RELEASE);
  M2.run(RELEASE);
  M3.run(RELEASE);
```

```
  M4.run(RELEASE);
}
int rightsee() {
  servo.write(20);
  delay(800);
  Left = ultrasonic();
  return Left;
}
int leftsee() {
  servo.write(180);
  delay(800);
  Right = ultrasonic();
  return Right;
}
```

## 10.2 Final overview of the system

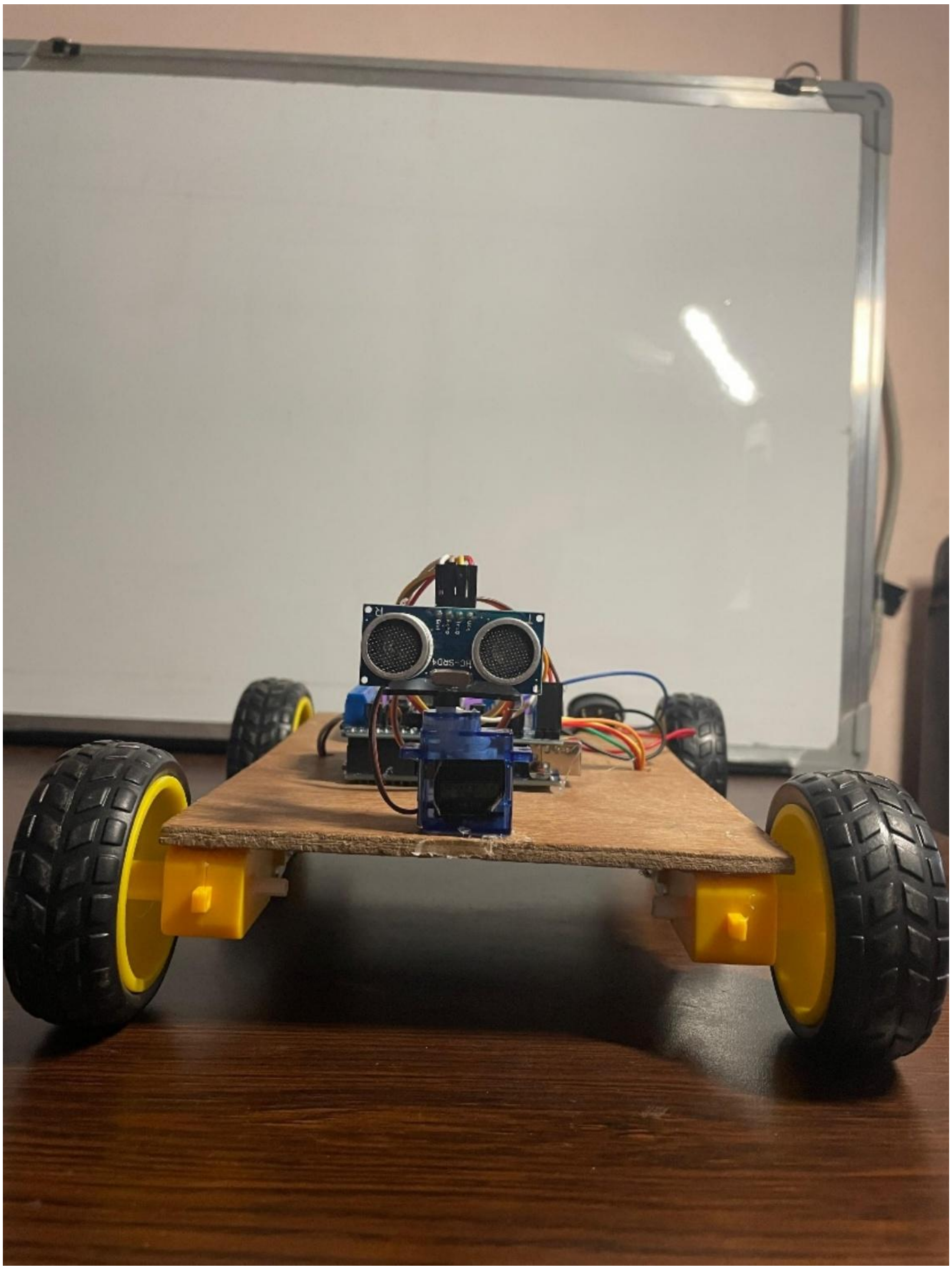

Figure 32: Front view of the system.

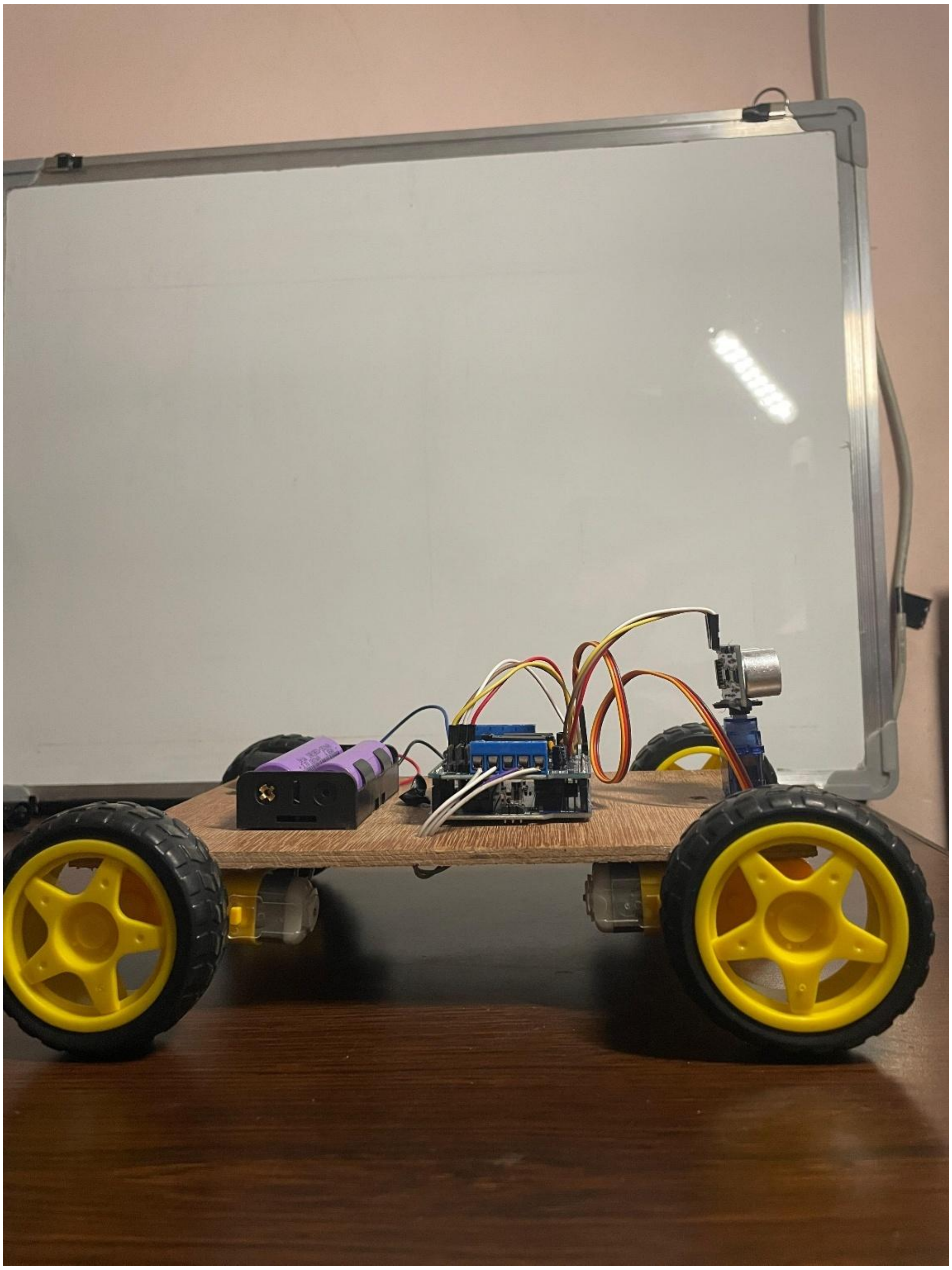

Figure 33: Right view of the system.

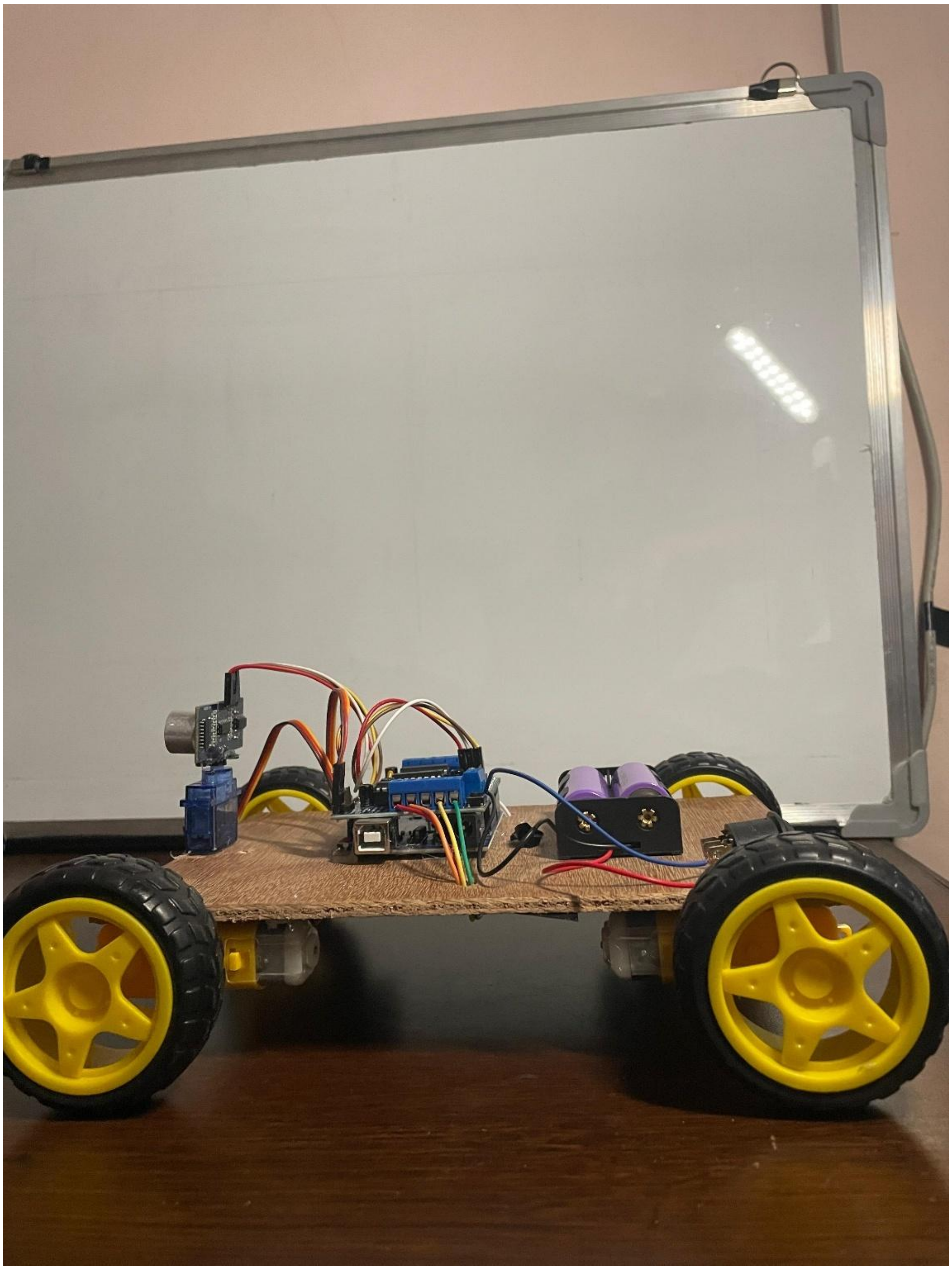

Figure 34: Left view of the System.

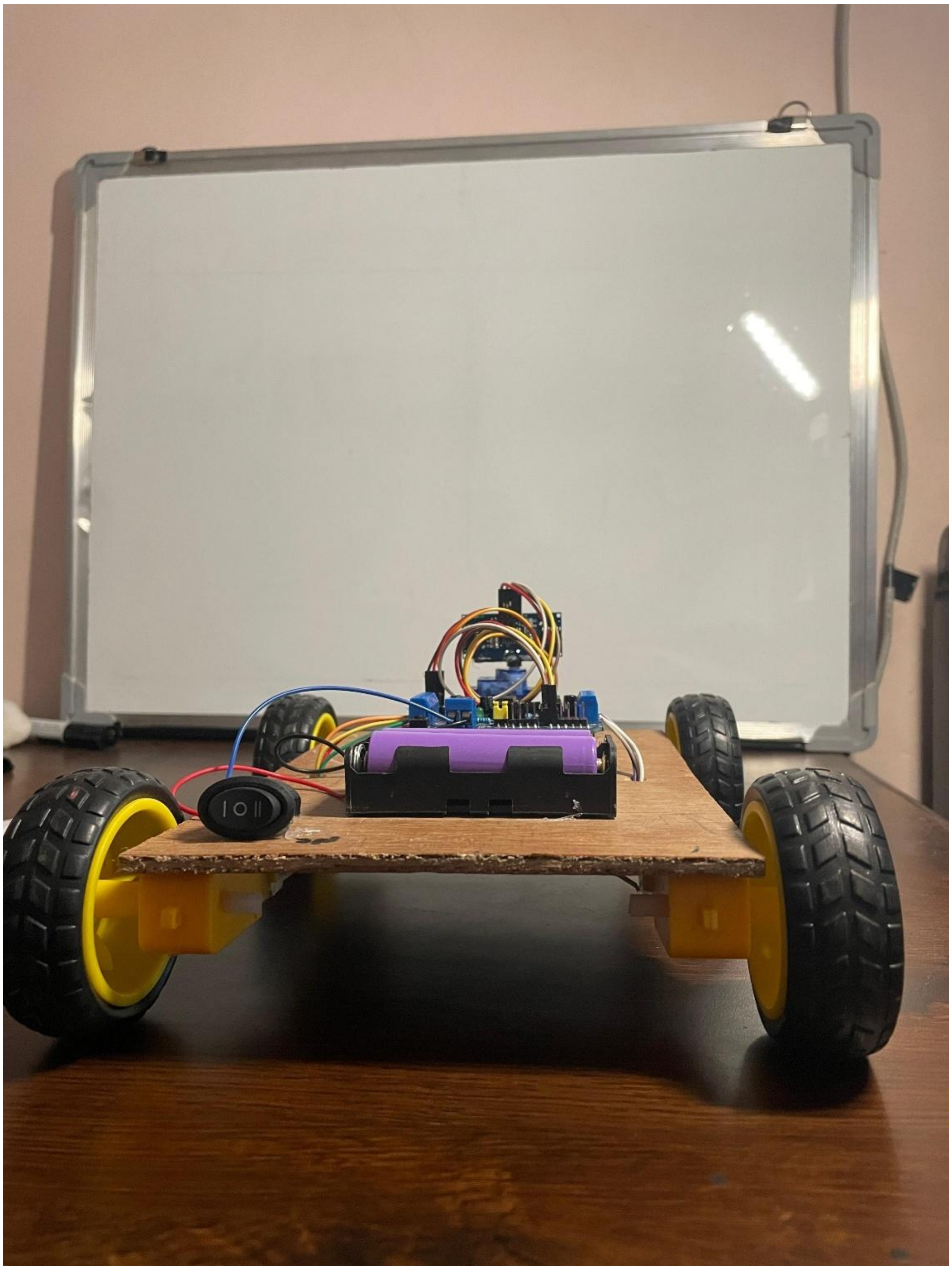

*Figure 35: Back view of the system.*

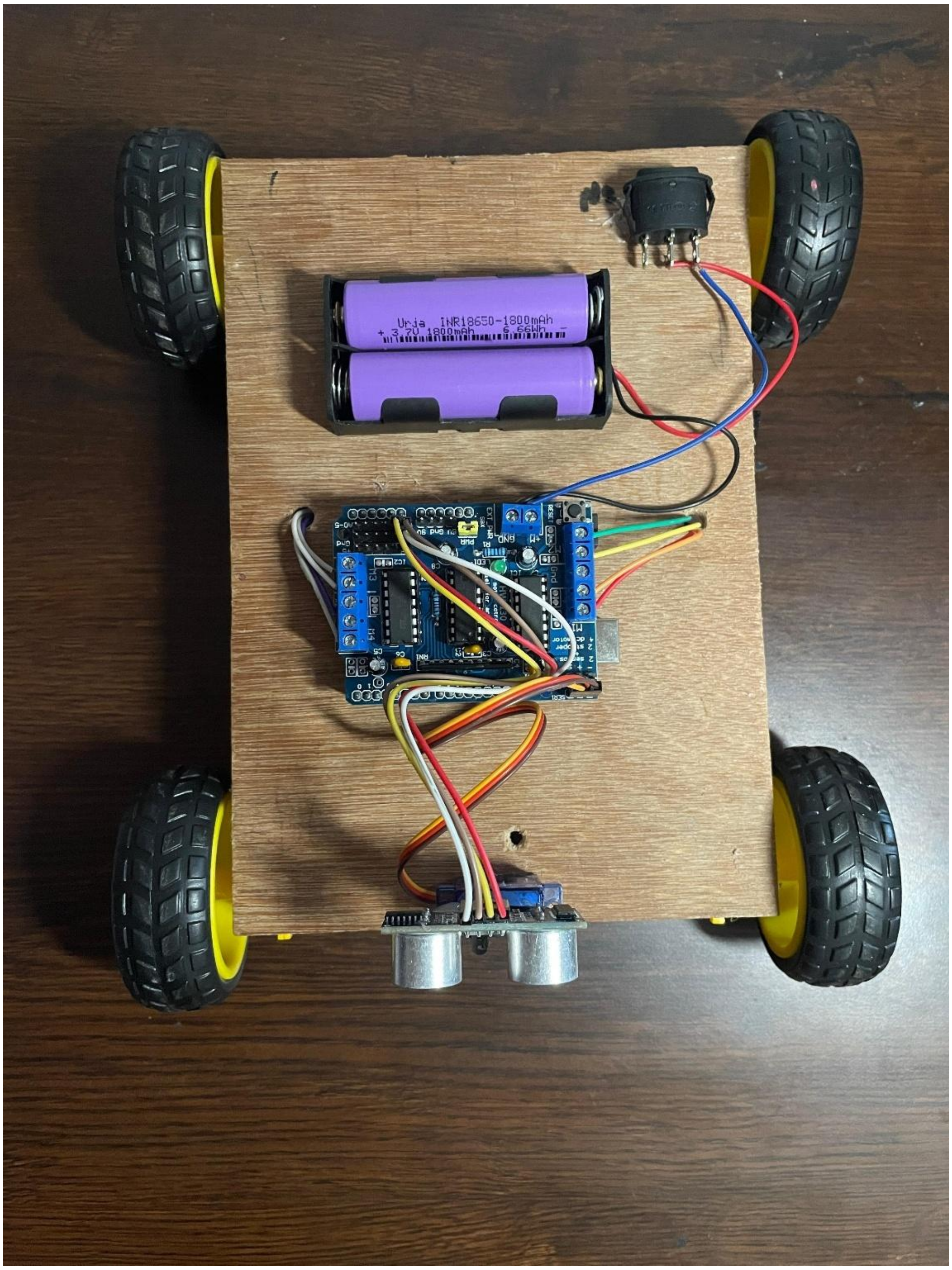


Figure 36: Top view of the system.

## 10.3 Individual contribution plan

| Student name | Role | Contribution |
|---|---|---|
| Mandeep Chhetri | The hardware design for obstacle avoidance car required precise physical buildup of chassis, with particular attention paid to mounting all motors and wheels. Effective circuit connections were made between components, especially the motor driver shield and Arduino integration. In the “System Development” section, each build phase begins from the setup phase to final integration is documented and hardware architecture diagrams provides visual representation of system design. Practical exposure to physical parts and well-defined documentation of the assembling process became key to the project’s success. | 20% |
| Samarpan Khadka | Combining the servo motor system and performance of extensive field-testing procedures authenticated the prototype performance in varying environments. Testing methodologies were established to test the accuracy for obstacle detection and response times. Context of applied IoT in robotics was addressed in the introduction of the project, whilst “Current Scenario” analysis evaluated current obstacle avoidance technology. Research skills made proper compilation and formatting of all references possible with all results of the tests being systematically presented in the results. | 20% |

| | | |
|---|---|---|
| Tenji Sherpa | Power management solutions comprised of effective battery designing and provision of stable power to all components. Reality-based enhancements of battery mounting increased reliability in hours-long testing. Documentation contributions were, for instance, generation of the "Future Works" section to establish possible system improvements. Generation of visual documentation, such as test videos and demonstration photos, produced a demonstration of car’s operational capabilities in various circumstances. | 20% |
| Prashun Lamichanne | Implementation of logical design concerned translating concepts conveyed in flow charts into codes used to institute functionalities, leading debugging efforts to solve motor control and sensor integration issues. System troubleshooting isolated and got rid wiring faults and signal interference issues. System diagrams that is block, circuit and flowcharts explained the design architecture visually. Co-writing the section “Results and Findings” offered advanced analysis of test results, focusing on debugging solutions, as well as their influence on the performance of the system. | 20% |
| Unisha Khagdi | Core software development concerned Arduino’s implementation of obstacle detection and avoidance algorithms, specifically a calibration of ultrasonic sensor thresholds for the best performance. Real- | 20% |

| | | |
|---|---|---|
| | time intelligent navigation decisions were made possible from the Arduino code. Extensive documentation included “Software Requirements” section and in-depth test cases for sensor’s functionality. Software troubleshooting with technical specifications well described guaranteed effective operation as well as in-depth documentation of the car’s command system. | |