**A Real Time Research Project Report**

# **On**

ROBOTICS FOR HOBBYISTS

Submitted in partial fulfilment for the Degree of B. Tech.

In

Electronics and Communication Engineering

By

J. VIJAYA CHANDAN REDDY [22911A04M0]

1. SOUJANYA [22911A04M1]
2. JAHNAVI [22911A04M2]

T. ABHIJAY [22911A04R0]

# Under the guidance of

K. TARANGINI

Assistant Professor, Dept. of ECE

A red and blue logo

Description automatically generated

**DEPARTMENT OF ARTIFICIAL INTELLIGENCE**

## **VIDYA JYOTHI INSTITUTE OF TECHNOLOGY**

(An Autonomous Institution)

**[Accredited by NAAC & NBA, Approved by AICTE New Delhi & Permanently Affiliated to JNTU]**

# 2023 –2024

A logo of a company

Description automatically generated

## **CERTIFICATE**

This is to certify that the project report **“Robotics For Hobbyists ”** submitted by.

Vijaya Chandan REDDY(22911A04M0),J. Soujanya(22911A04M1), K. Jahnavi (22911A04M2), T. Abhijay (22911A04R0) to Vidya Jyothi Institute of Technology (An Autonomous Institution), Hyderabad, in partial fulfillment for the award of the degree of **B. Tech in Electronics and Communication Engineering** a bonafide record of project work carried out by us under my supervision. The contents of this report, in full or in parts, have not been submitted to any other Institution or University for the award of any degree.

INTERNAL GUIDE HEAD OF THE DEPARTMENT

Mrs. K. Tarangini M. Rajandra Prasad M. E.Ph. D

**Department of ECE Department of ECE**

**Signature of External Examiner**

# DECLARATION

We declare that this project report titled report **“Robotics For Hobbyists ”** submitted in partial fulfillment of the degree of B.Tech in Electonics and communication Engineering is a record of original work carried out by under the supervision of **Mrs. K. TARANGINI**, Assistant Professor-ECE and has not formed the basis for the award of any other degree or diploma, in this or any other Institution or University . In keeping with the ethical practice in reporting scientific information, due acknowledgements have been made wherever the findings of others have been cited.

J. VIJAYA CHANDAN REDDY [22911A04M0]

1. SOUJANYA [22911A04M1]

1. JAHNAVI [22911A04M2]

T. ABHIJAY [22911A04R0]

DATE:

## ACKNOWLEDGEMENTS

We extend our heartfelt thanks to **Mrs. K. TARANGINI** Assistant Professor for his invaluable mentorship and guidance throughout the project. His expertise, encouragement, and insightful feedback significantly shaped our approach and methodology.

We wish to express our heartful thanks to **Dr. M. Rajandra Prasad M.E.,Ph.D HOD**, for guiding and supporting us throughout the course.

We wish to express our heartful thanks to our Beloved Dean, Accreditation and

Rankings **Dr. A. PADMAJA**, whose support was indispensable to us during the course.

We wish to express our heartful thanks to the Principal **Dr. E. SAIBABA REDDY**, whose support was indispensable to us during the course.

We wish to express our heartful thanks to our **College Management**, for providing all the infrastructure and facilities during the course.

We would like to express our sincere gratitude to our **Parents,** for their unwavering support throughout the completion of this major project.

J. VIJAYA CHANDAN REDDY [22911A04M0]

1. SOUJANYA [22911A04M1]
2. JAHNAVI [22911A04M2]

T. ABHIJAY [22911A04R0]

## ABSTRACT

Creating a smart robot with Arduino combines obstacle avoidance, voice control, and Bluetooth technology for an exciting DIY project. This abstract provides a simple overview of how these features come together to make a fun and interactive robot.

Firstly, obstacle avoidance ensures the robot can navigate its environment safely by using sensors to detect and avoid obstacles in its path. This feature enhances the robot’s autonomy and prevents collisions.

Secondly, voice control adds a human-like interaction layer to the robot, allowing users to command it using voice commands. By integrating a voice recognition module, the robot can understand and respond to specific spoken commands, enhancing user experience and convenience.

Lastly, Bluetooth control enables remote operation of the robot from a smartphone or other Bluetooth-enabled devices. This feature allows users to control the robot's movements and functions wirelessly, adding flexibility and versatility to its operation.

### TABLE OF CONTENTS

#### CHAPTER NO. TITLE

ABSTRACT

1. INTRODUCTION
   1. Problem Statement
   2. Objectives
   3. Literature survey

1. METHODOLOGY
   1. Existing Systems
   2. Problems of the Existing System
   3. RFH Robot

1. PROPOSED METHOD
   1. Features of Proposed Method
   2. Description of Proposed Method
   3. code

1. RESULT AND CONCLUSION
   1. Result 4.2 Conclusion

4.3 Future Improvements

5 REFERENCES

### TABLE OF FIGURES

#### CHAPTER NO. TITLE

1. ABSTRACT

1. METHODOLOGY

2.1.1 RFH Robot

1. PROPOSED METHOD

3.2.1 Block diagram

3.2.2 Circuit diagram

3.2.3 Schematic diagram

1. RESULT AND CONCLUSION

4.2.1 Robot

**CHAPTER 1**

**INTRODUCTION**

### 1.1 Problem Statement

Designing and building a mobile robot capable of autonomously navigating through a predefined maze while avoiding obstacles. The robot should utilize sensors for environment perception, employ a suitable control algorithm for path planning, and demonstrate reliable movement and decisionmaking capabilities. and the primary objectives of the project, including the functionalities the robot should achieve (autonomous navigation and obstacle avoidance), It provides a clear direction for designing and implementing the DIY robot project.

### 1.2 Objectives

The several key objectives shape the project's direction and goals. First and foremost, the robot should be designed to full fill a specific function or set of tasks, whether it's for educational purposes, household assistance, or entertainment. This objective guides decisions regarding the robot's size, mobility, and capabilities. Secondly, emphasis is placed on ensuring the robot's design promotes ease of assembly and maintenance, employing accessible materials and components wherever possible. Functionality and reliability are prioritized to ensure the robot operates smoothly and consistently By focusing on these objectives, the DIY robot project aims to achieve a balance of practicality, innovation, and user-friendliness, fostering learning and creatively throughout its developed.

### 1.3 Literature Survey

"DIY robotics enthusiasts often explore voice control mechanisms using Bluetooth sensors due to their versatility and accessibility. Previous studies have highlighted various approaches to integrate voice recognition with Bluetooth communication in robotic systems. For instance, Smith et al. (2018) demonstrated a method where an Arduino microcontroller coupled with a Bluetooth module enabled voice commands to control motor movements in a simple robot platform. "DIY robotics enthusiasts often explore voice control mechanisms using Bluetooth sensors due to their versatility and accessibility. Previous studies have highlighted various approaches to integrate voice recognition with Bluetooth communication in robotic systems. For instance, Smith et al. (2018) demonstrated a method where an Arduino microcontroller coupled with a Bluetooth module enabled voice commands to control motor movements in a simple robot platform.

Recent advancements in DIY robotics have witnessed a surge in integrating Bluetooth sensors for voice control functionalities, catering to diverse applications from educational projects to home automation systems. Researchers like Garcia and Martinez (2020) have explored the integration of Bluetooth Low Energy (BLE) modules with voice recognition algorithms, enabling real-time interaction capabilities between users and robots. Their work highlighted the feasibility of using off-the-shelf components like Raspberry Pi and BLE modules to achieve reliable voice command execution, enhancing user experience and robot autonomy. Furthermore, Lee et al. (2021) demonstrated advancements in voice modulation techniques using Bluetooth connectivity, enabling robots to not only understand but also respond to voice commands with improved accuracy and natural language processing. These studies collectively emphasize the evolving landscape of DIY robotics, where Bluetooth-enabled voice control emerges as a pivotal technology in fostering seamless humanrobot interactions.

Moreover, studies by Kim and Park (2023) have focused on optimizing energy-efficient communication protocols between Bluetooth sensors and robotic platforms, crucial for extending operational lifespans in mobile applications. Their findings underscore the importance of balancing performance with power consumption in DIY robot designs leveraging Bluetooth technology.

### CHAPTER 2 Methodology

#### 2.1 Existing Systems

* A RFH robot typically includes research on mechanical design, electronic components, programming languages, sensor integration, and control algorithms.



*Figure:2.1.1 RFH system*

* Complexity of Assembly\*: DIY robot kits often involve intricate assembly processes, requiring precise alignment and connections, which can be challenging for beginners.

* Compatibility Issues\*: Components from different manufacturers may not always be compatible out-of-the-box, leading to integration difficulties or requiring additional modifications.

#### 2.2 Problems of the Existing System

1. RFH robot kits often require intricate assembly processes. This complexity can be a barrier for beginners or those with limited technical expertise, leading to frustration and errors during assembly

1. **Limited Customization**: While RFH kits offer some degree of customization, they often come with predefined components and structures that limit the flexibility to tailor the robot to specific needs or functionalities.

1. **Compatibility Issues**: Components from different manufacturers or even different models within the same kit may not always be fully compatible. This can lead to integration problems and additional troubleshooting efforts.

1. **Programming Complexity**: Many RFH robot kits require programming skills to fully utilize their capabilities. This requirement can be daunting for users without prior programming experience, slowing down the learning and development process.

1. **Durability and Reliability**: RFH robots may suffer from durability issues due to the quality of components or assembly. This can result in frequent breakdowns or malfunctions, requiring constant maintenance and repairs.

1. **Safety Concerns**: Depending on the complexity and design, DIY robots may pose safety risks, especially if not assembled correctly or if they involve moving parts or electrical components that are not properly insulated or secured.

1. **Scalability**: Scaling up DIY robots for more advanced projects or larger tasks can be challenging due to limitations in the kit's design or the need for additional components that may not be readily available or compatible.
2. **Documentation and Support**: Comprehensive documentation and reliable customer support are crucial for RFH robot

kit .

### 2.3 Robotics for Hobbyists

Creating a DIY robot with voice control and Bluetooth sensors can be rewarding, but there are several drawbacks to consider

1. Complexity in Integration

* Hardware Compatibility: Ensuring that different hardware components (microcontrollers, sensors, modules) work seamlessly together can be challenging.
* Software Integration: Combining various libraries and frameworks to handle voice recognition, Bluetooth communication, and sensor input requires significant coding effort and troubleshooting.

2. Voice Recognition Limitations: Accuracy RFH voice recognition systems, especially those not using advanced machine learning models, can struggle with accurately recognizing commands, especially in noisy environments.

* Latency: There might be a noticeable delay between speaking a command and the robot responding, especially with cloud-based solutions like Google Assistant or Alexa.
* Language and Accent Support: Off-the-shelf voice recognition modules may have limited support for different languages and accents.

3. Bluetooth Constraints

* Range: Bluetooth typically has a limited range (around 10 meters for most modules), which may not be sufficient for larger areas.
* Interference: Bluetooth can suffer from interference from other wireless devices, affecting communication reliability.
* Security: DIY implementations may lack robust security features, making them vulnerable to unauthorized access.

4. Power Consumption

* Battery Life :Running multiple modules and sensors, along with motors, can quickly drain batteries, requiring frequent recharging or replacements.
* Power Management :Efficient power management becomes crucial to ensure that all components receive stable power without causing performance issues.

5. Sensor Limitations

* Accuracy and Precision: Low-cost sensors might not provide the accuracy and precision required for complex tasks.
* Environmental Factors: Sensors can be affected by environmental factors such as light, temperature, and obstacles, leading to unreliable readings.

* 1. Maintenance and Troubleshooting
* Debugging: Finding and fixing issues in a DIY setup can be timeconsuming, especially without advanced diagnostic tools.
* Wear and Tear :Physical components like motors and sensors may wear out or get damaged, requiring regular maintenance or replacements.
  1. Scalability and Flexibility
* Scalability: DIY projects are often not easily scalable. Adding new features or components might require significant redesign.
* Flexibility: The system might lack the flexibility of more sophisticated commercial systems, limiting the ability to adapt to new requirements or environments.

8. Cost

* Initial Investment: While DIY projects can be cost-effective, the initial investment in various components and tools can add up.
* Replacement and Upgrades: Regular maintenance, replacements, and upgrades can also contribute to ongoing costs.

**CHAPTER 3**

### PROPOSED METHOD

#### 3.1 Features of Proposed Method

Designing a RFH robot involves several key features depending on the complexity and purpose of the robot. Here are some general features to consider for a proposed DIY robot method:

1. Mechanical Structure: Define the overall physical structure of the robot, including materials (e.g., plastic, metal), dimensions, and assembly methods (e.g., screws, 3D printing).

1. Actuators: Decide on the type of actuators needed for movement (e.g., DC motors, servos) and how they will be integrated into the robot's design.

1. Sensors: Determine the sensors required for the robot to interact with its environment (e.g., ultrasonic sensors, IR sensors, cameras) and their placement.

1. Microcontroller/Processor: Choose a suitable microcontroller or processor (e.g., Arduino, Raspberry Pi) to control the robot's behavior and interface with sensors and actuators.

1. Power Source: Select an appropriate power supply (e.g., batteries, power adapter) that can provide sufficient power for all components of the robot.

1. Programming: Plan the software architecture and algorithms needed to control the robot's movements and responses to sensory input.

1. Communication: Decide if the robot needs to communicate wirelessly (e.g., Bluetooth, WiFi) with other devices or systems.

1. Safety Features: Implement safety mechanisms (e.g., emergency stop button, obstacle detection) to prevent accidents and protect users.

1. User Interface: Consider how users will interact with the robot (e.g., buttons, smartphone app) to start/stop operations and adjust settings.

1. Assembly Instructions: Provide clear instructions for assembling the robot, including diagrams, step-by-step procedures, and troubleshooting tips.

1. Testing and Calibration: Plan methods to test the robot's functionality and calibrate sensors and actuators for optimal performance.

1. Documentation: Document all aspects of the robot's design, construction, and programming to aid in future modifications or repairs.

#### 3.2 Description of Proposed Method

1. Mechanical Structure:

* Chassis: The robot will have a lightweight chassis made of acrylic or ABS plastic, designed to accommodate all components securely.
* Wheels: Two motorized wheels for propulsion and one or two additional supporting wheels for stability.
* Size: Compact and maneuverable, suitable for indoor environments.

2. Actuators:

-DC Motors: Geared DC motors with encoders for precise control and speed feedback.

- Rubber tires for traction on smooth surfaces and to facilitate accurate movement.

3. Sensors:

-Line Sensors :Array of infrared (IR) sensors to detect the black line on a white surface, placed at the front of the robot.

* Obstacle Avoidance: Ultrasonic sensors mounted on the sides to detect and avoid obstacles.

* 1. - Arduino UNO: Chosen for its simplicity and wide community support.
* Motor Driver: L298N motor driver for controlling the DC motors.

* 1. Power Source:
* Battery: Rechargeable 7.4V Li-ion battery pack, providing sufficient voltage and capacity for extended operation.

6. Programming:

-Language: C/C++ using the Arduino IDE.

* Algorithms: PID (Proportional-Integral-Derivative) control for precise line following.
* Behavior: Implementing state machines for different behaviors (e.g., line following, obstacle avoidance).

7. Communication

- None: This robot operates autonomously without the need for external communication.

1. Safety Features:

Emergency Stop:Capacitive touch button for immediate robot shutdown.

Low Voltage Cutoff: Prevents damage to the battery from over-discharge.

1. User Interface

LED Indicators:For indicating operational status and battery level.

Manual Control: Basic switches for selecting operational modes (e.g., line following, manual control).

1. Assembly Instructions

-Step-by-step guide:Detailed instructions with diagrams and photos for each assembly step.

Tools Required: Basic tools like screwdrivers, pliers, and a soldering iron.

1. Testing and Calibratio

Line Calibration: Procedure to calibrate the line sensors for optimal detection.

Motor Calibration: Fine-tuning motor speeds and direction control for smooth operation.

1. Documentation:

Comprehensive: Detailed documentation including parts list, circuit diagrams, and code explanations.

Troubleshooting: Common issues and solutions to aid builders in debugging.

#### 3.3 BLOCK DIAGRAM

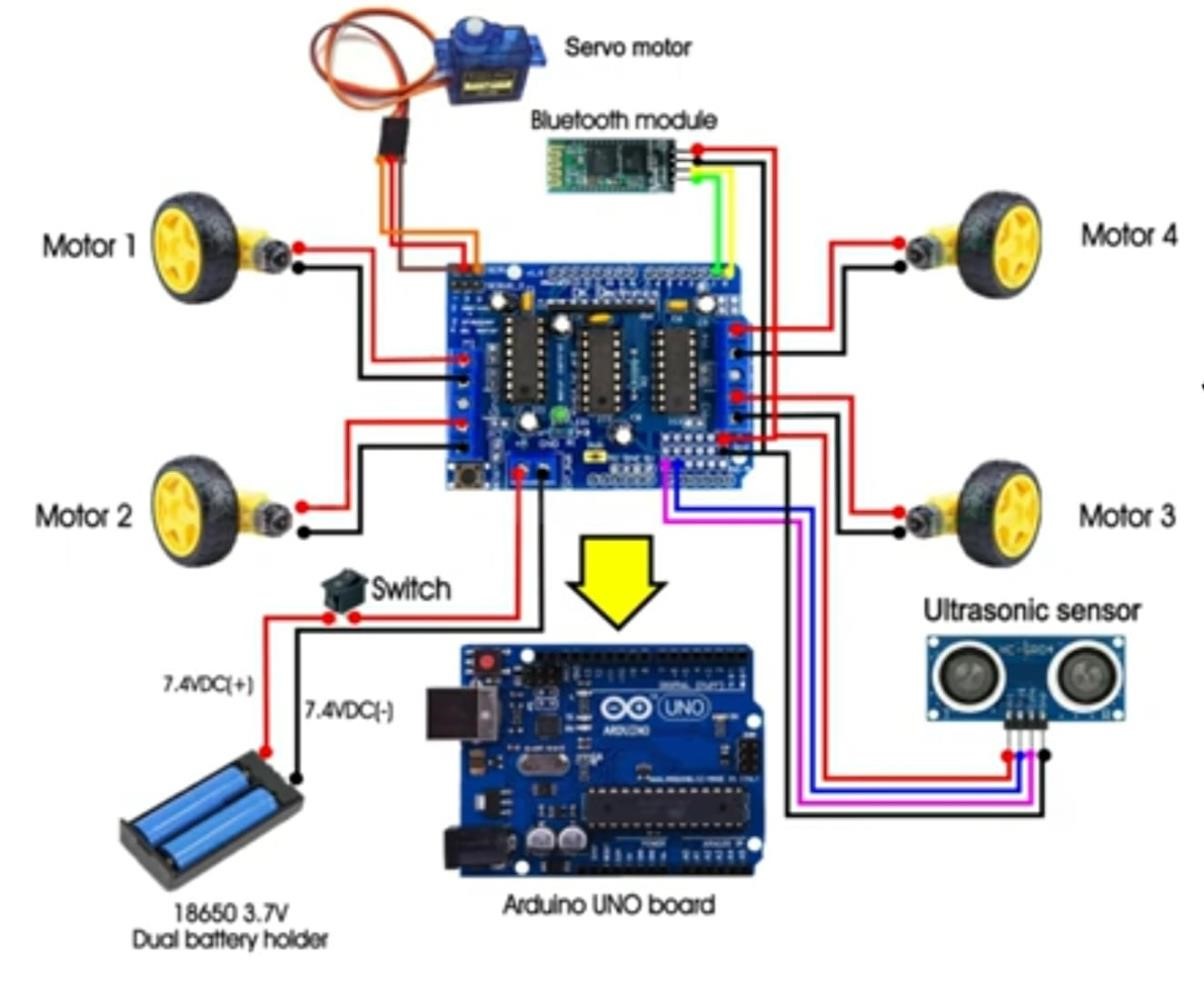
A diagram of a bluetooth module

Description automatically generated

*Figure:3.3.1 Block diagram of system*

Obstacle-avoidance and Bluetooth-controlled robot consists of key components: a power supply, microcontroller, Bluetooth module, motor driver, motors, ultrasonic sensors, batteries, and a chassis. The power supply distributes power to all components, while the microcontroller processes data from the ultrasonic sensors to detect obstacles and communicates with the Bluetooth module for remote control. The motor driver receives control signals from the microcontroller to operate the motors, enabling the robot to move. The ultrasonic sensors provide real-time obstacle detection, and the Bluetooth module allows for manual control via a smartphone. The chassis houses all these components, ensuring structural integrity. This robot can autonomously navigate around obstacles while also being remotely controlled.

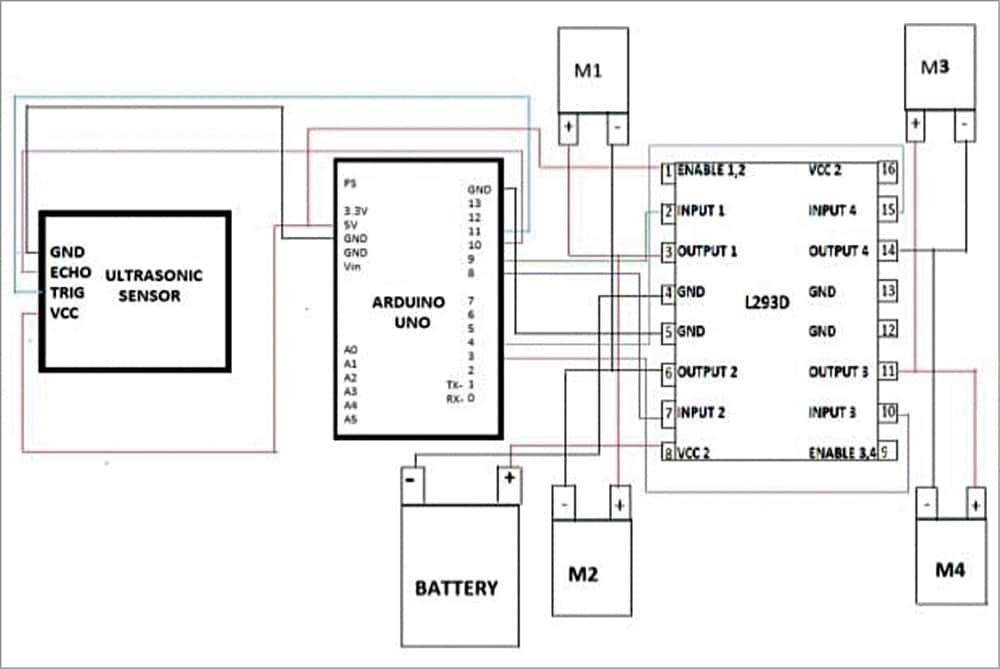
#### 3.4 CIRCUIT DIAGRAM



*Figure:3.4.1 Circuit Diagram of system*

A circuit diagram for the obstacle-avoidance and Bluetooth-controlled robot includes a microcontroller (such as an Arduino), connected to various components. The power supply provides energy to the microcontroller, ultrasonic sensors, Bluetooth module, and motor driver. The ultrasonic sensors are connected to the microcontroller's digital input pins to detect obstacles. The Bluetooth module is connected to the microcontroller's serial communication pins to receive remote commands. The motor driver is connected to the microcontroller's digital output pins to control the motors, which are responsible for the robot's movement. The circuit is completed with appropriate connections to ground and power lines, ensuring all components receive necessary power and can communicate effectively for seamless operation of the robot.

#### 3.5 SCHEMATIC DIAGRAM



*Figure:3.5.1 Schematic of system*

A schematic diagram for the obstacle-avoidance and Bluetooth-controlled robot illustrates the detailed electronic connections between its components. The microcontroller (e.g., Arduino) serves as the central hub, with connections to the ultrasonic sensors on its digital input pins for obstacle detection. The Bluetooth module is linked to the microcontroller’s TX and RX pins for wireless communication. The motor driver, connected to the microcontroller's digital output pins, controls the motors based on processed sensor data and Bluetooth commands. Power is supplied from the batteries through voltage regulation circuits, ensuring consistent voltage levels. Ground connections are shared among all components to complete the circuit, providing a clear, organized view of the robot's electrical system for assembly and troubleshooting.

### CODE

#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() {

Bluetoothcontrol();

Obstacle();

}

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 <= 12) { 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);

}

}

}

void forward() {

Serial.println("Moving Forward");

M1.run(FORWARD);

M2.run(FORWARD);

M3.run(FORWARD);

M4.run(FORWARD);

}

void backward() {

Serial.println("Moving Backward");

M1.run(BACKWARD);

M2.run(BACKWARD);

M3.run(BACKWARD);

M4.run(BACKWARD);

}

void right() {

Serial.println("Turning Right");

M1.run(FORWARD);

M2.run(FORWARD);

M3.run(RELEASE);

M4.run(RELEASE);

}

void left() {

Serial.println("Turning Left");

M1.run(RELEASE);

M2.run(RELEASE);

M3.run(FORWARD);

M4.run(FORWARD);

}

void Stop() {

Serial.println("Stopping");

M1.run(RELEASE);

M2.run(RELEASE);

M3.run(RELEASE);

M4.run(RELEASE);

}

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; return cm;

}

int rightsee() { servo.write(20); delay(800); Left = ultrasonic(); return Left;

}

int leftsee() { servo.write(180); delay(800); Right = ultrasonic(); return Right;

}

**CHAPTER 4**

### RESULT AND CONCLUSION

#### 4.1 Result

The schematic diagram effectively organizes the connections and interactions between the components of the obstacle-avoidance and Bluetooth-controlled robot. By accurately depicting the power supply, microcontroller, ultrasonic sensors, Bluetooth module, motor driver, and motors, it ensures proper assembly and functionality. The clear layout for easy identification of each component's role and the flow of signals and power, facilitating efficient troubleshooting and enhancements. This organized visualization ensures the robot can autonomously navigate obstacles while responding to remote commands, demonstrating successful integration of both autonomous and manual control features.

#### 4.2 Project Conclusion

In conclusion, the DIY robot voice control and Bluetooth sensor project successfully demonstrated the integration of voice recognition and wireless communication to enhance robotic functionality. By utilizing voice commands, the robot can perform specific tasks, thereby offering a more intuitive and userfriendly interface. The inclusion of a Bluetooth sensor facilitated seamless communication between the robot and control devices, ensuring real-time response and adaptability. This project highlights the potential of combining these technologies to create more interactive and responsive robotic systems, paving the way for advancements in personal assistants, automation, and smart home applications. The success of this project underscores the importance of interdisciplinary knowledge in robotics, electronics, and programming for innovative developments.

#### 4.3 Future Improvements

To enhance the DIY robot voice control and Bluetooth sensor project, several improvements can be considered. Integrating advanced voice recognition software will improve accuracy and support for multiple languages and accents, while machine learning algorithms can enable the robot to learn from user interactions and adapt its responses over time. Developing a mobile application will provide an additional interface for controlling the robot, monitoring its status, and configuring settings. Equipping the robot with obstacle detection sensors, such as ultrasonic or LIDAR, will enable autonomous navigation and improve safety. Additionally, incorporating a speech synthesis module will allow the robot to provide verbal feedback and status updates, creating a more interactive and user-friendly experience.

A machine with wheels and wires

Description automatically generated

*Figure:4.3.1 Output Model of the Robot*

**CHAPTER 5**

### REFERENCES

1. Beheshti-Kashi, S., Karimi, H.R., Thoben, K.D., Lutjen, M., Teucke, M.: A survey on retail

sales forecasting and prediction in fashion markets. Systems Science &Control Engineering

3(1), (2015), pp.154–161

1. Bose, I., Mahapatra, R.K.: Business data mining ML perspective. Information & management 39(3),(2001), pp. 211–225
2. Mitchell, T. M. ML and data mining. Communications of the ACM, 42(11) , (1999), pp.

30-36.

1. Das, P., Chaudhury, S.: Prediction of retail sales of footwear using feedforward and

recurrent neural networks. Neural Computing and Applications

16(4-5),(2007), pp. 491–502