International Journal on Science and Technology (IJSAT)

E-ISSN: 2229-7677 ● Website: www.ijsat.org ● Email: editor@ijsat.org

Obstacle Avoiding, Bluetooth and Voice Control Robot Car

Baggu Saikumar, G Sree Vikas, K Chandra Sheaker, K Subramanyam

Abstract

This project introduces an abstract avoidance system controlled via Bluetooth and voice commands using an Arduino board. Ultrasonic sensors detect obstacles, providing real-time data for the system. Users can remotely control the system's movements and navigate through complex environments using a mobile device connected via Bluetooth. Additionally, voice commands enhance usability and convenience. The integration of hardware components, including ultrasonic sensors, an Arduino board, and a Bluetooth module, along with algorithm development for obstacle detection and communication protocols, enables the system's functionality. This versatile system finds applications in robotics, automation, and smart environments where obstacle avoidance is crucial. By combining Bluetooth and voice control, this project offers an efficient and user-friendly solution for enhancing control and safety in various real-world scenarios. Keywords: Avoidance, Bluetooth-Controlled, Voice-Controlled, Arduino Board, Obstacle Detection.

1. Introduction

Building an obstacle avoidance, Bluetooth, and voice control robot car using Arduino is an educational and captivating project that merges robotics, programming, and electronics. By assembling a robot car, integrating sensors, connecting them to an Arduino board, and developing code, users can create a versatile vehicle capable of autonomously evading obstacles while also being remotely controlled through Bluetooth and voice commands. This project offers several advantages, including fostering an understanding of robotics, honing programming skills, and nurturing problem-solving abilities. It also provides hands-on experience with hardware assembly, wiring, and troubleshooting, encouraging practical knowledge in electronics. The flexibility of an Arduino-based robot car opens up numerous applications. It can be utilized for surveillance, industrial automation, search and rescue missions, and home automation tasks. In educational settings, it serves as a valuable tool for teaching robotics and electronics to students, promoting engagement and curiosity. Furthermore, it provides practical experience for aspiring engineers in fields such as autonomous vehicles, smart cities, and environmental monitoring. In summary, an obstacle avoidance, Bluetooth, and voice control robot car using Arduino offers an immersive exploration of the realm of robotics and automation. It nurtures creativity, innovation, and problem-solving skills while delivering practical proficiencies in programming and electronics. Whether for educational purposes, real-world applications, or fostering innovation in emerging fields, this project provides an enjoyable and rewarding experience for individuals interested in STEM fields and the fascinating world of robotics.

2. Existing System And Proposed System

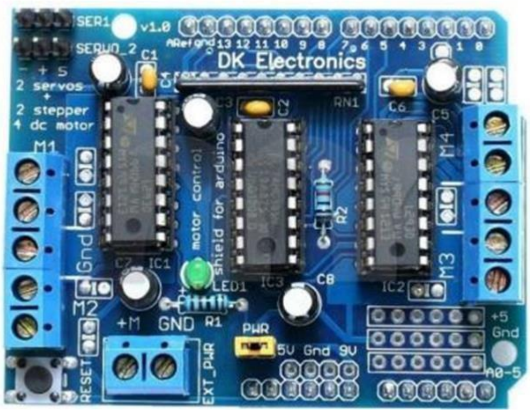
The existing system for the obstacle avoidance robotic car focuses solely on obstacle detection and avoidance capabilities. It lacks versatility and interactivity. In contrast, the proposed system enhances the existing obstacle avoidance by incorporating Bluetooth and voice control. With Bluetooth integration, the car can be remotely controlled using a smartphone or computer, providing a convenient and wireless mode of operation. Voice control adds interactivity, allowing users to issue commands for movement. This comprehensive system combines obstacle avoidance, Bluetooth, and voice control, making the project more engaging and versatile for educational, practical, and recreational purposes.

3. Hardware Requirements

1. Arduino UnoATmega328P

The Arduino Uno is a popular microcontroller board based on the ATmega328P chip. It offers a simple and user-friendly platform for electronics and programming projects. With its built-in I/O pins, it allows easy connection and control of various sensors, actuators, and other components. The Arduino Uno is widely used for prototyping, education, and creating interactive projects.

2. Motor Driver Shield



A motor driver shield is an electronic circuit board that connects to an Arduino microcontroller to control the speed and direction of DC motors. It utilizes H-bridge circuits for variable speed and bidirectional control. Motor driver shields offer convenience and additional features like current sensing and over-current protection. They are commonly used in robotics and automation projects for precise motor control. Popular motor driver shields include L293D, L298N, and TB6612FNG.

3. SG90 Servo Motor

A servomotor is a precise actuator that allows for control of angular or linear position, velocity, and acceleration. It operates in a closed-loop system using position feedback to control its motion. The input signal represents the desired position, and the motor adjusts its rotation accordingly. Servomotors commonly use encoders for position and speed feedback, enabling quick and accurate positioning. More advanced servomotors incorporate PID control algorithms for improved performance and reduced overshooting. They find applications in various fields, including industrial motion control and radio-controlled models.

1. DC motors

You will need four DC motors to drive the robot car.

2. Wheels

Four wheels are required to connect to the DC motors.

3. Ultrasonic sensor

Ultrasonic sensor is used to detect obstacles in front of the robot car. You can use HC-SR04 or JSN-SR04T ultrasonic sensor

4. Jumper wires

Jumper wires are used to make the electrical connections between the components.

5. Battery

A battery is required to power the robot car. You can use a 9V battery or a 6V rechargeable battery.

6. Chassis

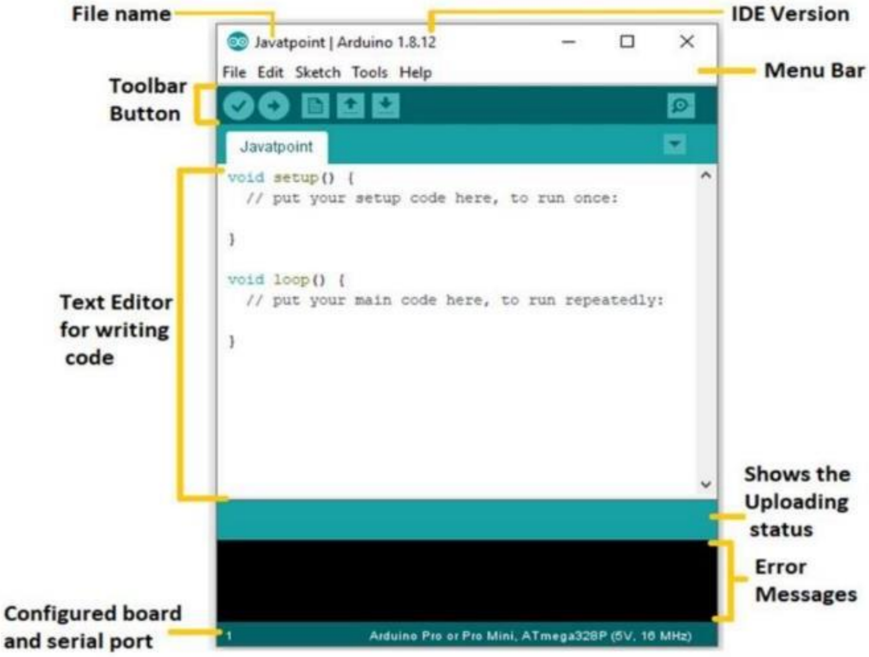
A chassis is used to hold all the components together and provide stability to the robot car.

7. Wheel

A wheel is used for balancing the robot car.

4. Software Requirements

1. ARDUINO IDE

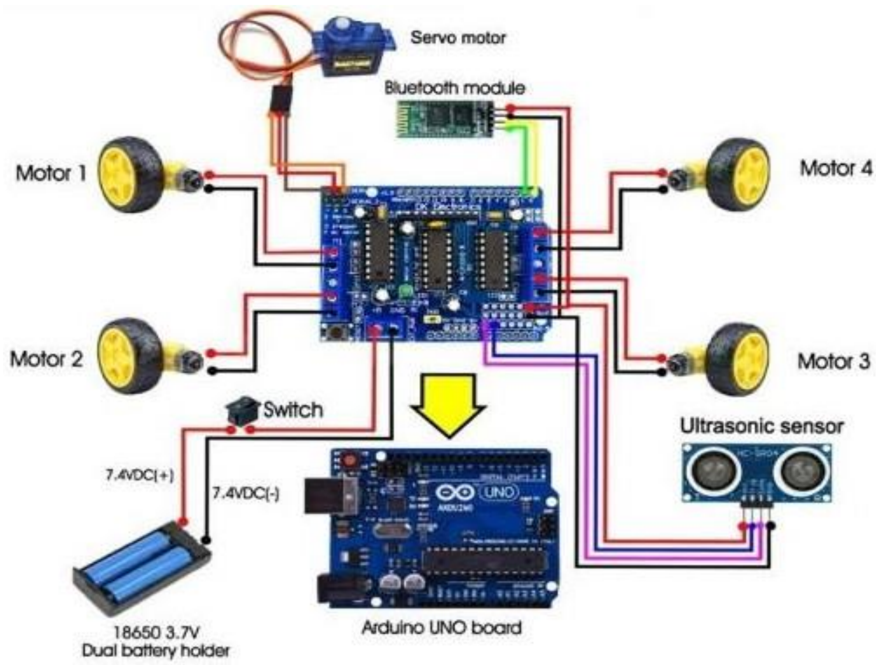


The Arduino IDE (Integrated Development Environment) is a software platform that provides a user-friendly environment for programming Arduino microcontrollers. It is a crucial tool for beginners and experienced developers alike, as it simplifies the process of writing, compiling, and uploading code to Arduino boards. The Arduino IDE offers a simple and intuitive interface, making it accessible to individuals with minimal programming experience. It provides a code editor with syntax highlighting and auto-completion, enabling users to write code efficiently. The IDE supports the Arduino programming language, which is based on C and C++, making it easy to learn and understand. One of the key features of the Arduino IDE is its vast library of pre- written code, known as "sketches."

These sketches provide ready-to-use code snippets for various hardware components and functions, saving time and effort in coding. Additionally, the IDE has a built-in serial monitor, which allows developers to debug and monitor their Arduino projects in real-time. The Arduino IDE supports a wide range of Arduino boards, from the basic Uno to more advanced models like the Mega and Due. It is compatible with multiple operating systems, including Windows, Mac OS X, and Linux, ensuring broad platform accessibility. In summary, the Arduino IDE serves as a fundamental tool for programming Arduino boards, offering a user-friendly interface, a rich library of pre-written code, and compatibility with various hardware components. It empowers individuals to unleash their creativity and bring their electronic projects to life.

International Journal on Science and Technology (IJSAT)
E-ISSN: 2229-7677 ● Website: www.ijsat.org ● Email: editor@ijsat.org

2. FRITZING



Fritzing is a popular open-source software tool designed for electronics enthusiasts, hobbyists, and designers. It provides a user-friendly platform for creating and documenting electronic circuits, prototyping projects, and designing custom PCBs (Printed Circuit Boards). With its intuitive interface and extensive component library, Fritzing simplifies the process of visualizing and sharing electronic projects. The key feature of Fritzing is its breadboard view, which allows users to assemble virtual circuits by dragging and dropping electronic components onto a breadboard.

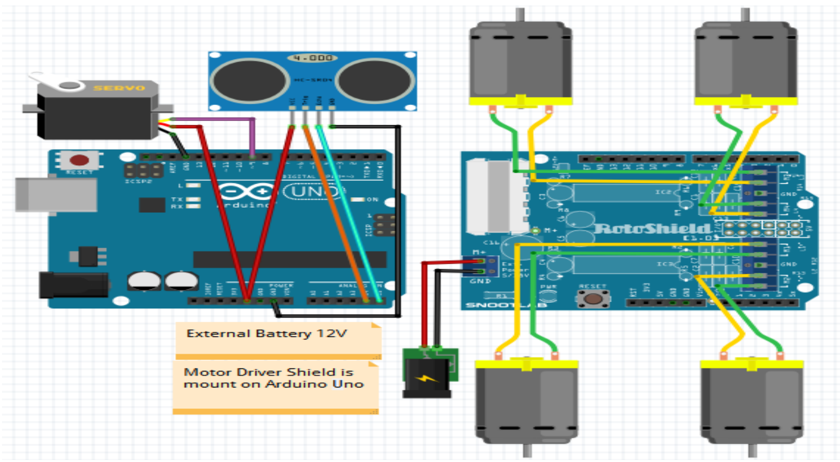
This feature helps beginners understand the physical connections between components and simulate the behavior of their circuits before building them in real life. Fritzing also provides schematic and PCB views, enabling users to create professional-level circuit diagrams and design custom PCB layouts. Fritzing's component library contains a vast collection of commonly used electronic components, including microcontrollers, sensors, actuators, and various other modules. It also allows users to create their own custom components, providing flexibility for unique project requirements. Moreover, Fritzing supports importing and exporting designs in various formats, making it compatible with other popular CAD tools. Another valuable aspect of Fritzing is its project documentation capabilities. Users can create interactive documentation for their projects, including images, schematics, and step-by-step instructions. This makes it easier to share projects with others, collaborate on designs, and even showcase projects on websites or blogs. Overall, Fritzing is a versatile software tool that empowers electronics enthusiasts and designers to create, visualize, and document their electronic projects effectively. Its intuitive interface, extensive component library, and documentation features make it an essential tool in the electronics community.

3. MATLAB

MATLAB is a powerful software tool widely used in various fields, including engineering, mathematics, and scientific research. It provides a high-level programming environment for numerical computation, data analysis, and visualization. With its extensive library of built-in functions and toolboxes, MATLAB simplifies complex mathematical operations and enables rapid prototyping of algorithms. Its interactive interface allows users to perform tasks such as data manipulation, simulation, and image processing with ease. MATLAB also supports the creation of graphical user interfaces (GUIs), making it convenient for developing user-friendly applications. Overall, MATLAB is a versatile software that facilitates efficient data analysis, algorithm development, and problem-solving in diverse disciplines.

6. Implemenation and Design

1. Assemble the hardware components: Gather the chassis, DC gear motors, wheels, motor driver shield, ultrasonic sensor, Arduino board, and battery. Follow the provided instructions to assemble them.
2. Connect the motor driver shield: Connect the DC gear motors to the motor driver shield and then connect the motor driver shield to the Arduino board.
3. Connect the ultrasonic sensor: Attach and connect the ultrasonic sensor to the Arduino board.
4. Write the code: Develop the Arduino code to handle obstacle detection, avoidance, and movement control.
5. Test the robot car: Power on the robot car and observe its movement in response to the ultrasonic sensor and user commands.
6. Fine-tune the code: Make necessary adjustments to improve performance, accuracy, and reliability.
7. Add a user interface: Integrate a user interface like an LCD display or Bluetooth module for control and distance readings.
8. Enclose the robot car: Provide a protective casing to safeguard the components and enhance durability.
9. Test and debug: Thoroughly test and debug the robot car to ensure proper operation and fix any issues that may arise.



7. Conclusion

An obstacle avoidance robot car using Arduino is an intriguing and practical venture with diverse applications. By merging components like DC gear motors, ultrasonic sensors, and motor driver shields with Arduino's software code, a self-navigating robot car capable of detecting and evading obstacles can be constructed. This project encompasses motor control, sensor integration, and programming, providing an interactive opportunity for individuals to grasp concepts in electronics, robotics, and coding. As a result, it serves as an exceptional educational tool for students and enthusiasts. In essence, an obstacle avoidance robot car using Arduino offers an exciting and challenging undertaking, fostering the development of skills in robotics, coding, and problem- solving. Its real-world utility extends to areas like security, surveillance, and exploration.

Reference

1. Oloka Reagan Otieno, “SOLAR TRACKER FOR SOLAR PANEL”, University of Nairobi, Dept. of Electrical and Electronic Engineering, 24th August 2015.
2. Mayank Kumar Lokhande, “Automatic Solar Tracking System,” International Journal of Core Engineering & Management, October, 2014.
3. Scott J Hamilton, “Sun-Tracking Solar Cell Array System,” University of Queensland, Department of Electrical Engineering, 1999.
4. C. Chang, Chapter 5 Tracking solar collection technologies for solar heating and cooling systems, “Advances in Solar Heating and Cooling”, Woodhead Publishing, 2016
5. R.Z. Wang, T.S. Ge, “Advances in Solar Heating and Cooling,” Woodhead Publishing, 2016, Pages 81-93, ISBN 9780081003015.
6. Levent Bas, “Thin Film vs. Crystalline Silicon PV Modules,” December, 2011.
7. X. Sun and N. Ansari, "EdgeIoT: Mobile Edge Computing for The Internet Things", IEEE Communi- cations Magazine, vol.54, no.12, pp.22-29, Dec. 2016.