Design and Simulation of Voice Controlled Robotic Car

M. Vamshi Kiran, S.K. Santhosh Kumar,
Naman M Naik, M.R. Beluraiah and S. Divya
Department of Electrical and Electronics Engineering
BMS College of Engineering, Bangalore, Karnataka, India

Abstract
The project aims to design a voice controlled robotic car, which can be operated through an app built using MIT App Inventor. Arduino Mega is used to control the differential drive robotic car. An Arduino code is developed to take voice commands from the Bluetooth module. This Arduino robotic car can do multiple tasks like obstacle avoidance and edge detection. Ultrasonic sensors are used to detect obstacles and IR sensors to detect edges. A push button is also used in the robot which acts as a bump sensor. A model was created in coppeliasim to show how model works in real world. The robotic car operations and functions are designed and simulated.
Keywords
Arduino Mega 2560, Bluetooth Module HC – 05, Ultrasonic Sensor, IR sensor, Motor driver L298

I. INTRODUCTION

An Arduino Mega based voice controlled robotic car uses a smartphone application to control the car using voice commands. This voice-controlled car is controlled by using an android mobile phone through voice commands instead of any other method like buttons, gestures, etc. Voice commands through the android app such as "forward", "backward", "left', "right", "rotate clockwise", "rotate counter clockwise" are given to control the robotic car. The android app in phone is used as a transmitter and Bluetooth module placed in the robotic car is used as a receiver. This robotic car can perform other functions like, obstacle avoidance, edge detection and avoidance. The robot turns in the counter clockwise direction as soon as an obstacle is encountered. This is obstacle avoidance. Ultrasonic sensors are used to perform this function. When the robot encounters an edge in forward direction, it moves backward and vice versa. IR sensors are used to perform this function. The push button placed at the front and back of the car acts as a bumper sensor. A small amount of force when applied to this button can turn it on and the car starts to move in the opposite direction of the active bumper sensor. The components used include: Arduino mega 2560, Ultrasonic sensors, IR sensors, Motor driver L298, Bluetooth module HC-05, Gear motor, Bump switch.

Arduino Mega 2560

The Arduino Mega 2560 is microcontroller board which has 54 digital I/O pins (contains 15 PWM outputs), 16 analog input pins, 4 UARTs. It also has 16 MHz crystal oscillator. It has EPROM of 4 KB, Flash memory of 256 KB (bootloader uses a total of 8KB from 256KB flash memory), and SRAM of 8KB. Many shield designed for UNO is compatible with Arduino Mega 2560 board. Fig. 1 shows a typical Arduino Mega 2560 board.

Ultrasonic Sensors

An ultrasonic sensor is a sensor that senses ultrasonic waves, using which the distance can be measured. Ultrasonic sensors produce a sound wave of frequency higher than range of human hearing, this wave is sent through transducer and received after being hit by the object. The time taken for the ultrasonic wave to reach transducer is considered and distance is measured. Fig. 2 shows typical ultrasonic sensor, HC – SRO.

IR Sensors

An IR sensor emits infrared light in order to sense any object in the surroundings. It consists of an IR transmitter (Infrared LED) and an IR receiver (Infrared Photodiode). IR transmitter emits infrared light to the surrounding, some of the light gets reflected and photodiode senses the reflected IR radiation. The output voltage received from the sensor is directly proportional to the IR light received by photodiode. Fig. 3 shows a typical IR sensor

Motor Driver L298

L298 motor driver is a high-powered version of the L293 motor driver IC. L298 motor driver can control two dc motors. It consists of two enable pins to enable the motor. It consists of 4 input pins, each motor is connected to 2 input pins. Using this motor driver, the speed and direction of two dc motors can be controlled. Fig.4 shows a typical motor driver IC L298.

Bluetooth Module HC-05

HC - 05 is used for wireless Bluetooth communication. It uses UART communication to communicate with microcontrollers like Arduino. It can also be used in master/slave configuration. HC - 05 has built-in antenna for communication. It uses 9600 as baud rate. Fig. 5 shows a typical Bluetooth module HC–05.

Gear Motor

Geared DC Motor is just a DC motor to which gear assembly is attached. The gear assembly used to reduce speed and increase torque. Using different combination of gears in a gear motor, its speed can be varied to desired value. Fig. 6 shows a typical DC geared motor.

Bump Switch

Bump Switch Bump switches are similar to push button, it helps to avoid obstacles and to function as collision detector. Pressing the push-button implies ON and releasing the button implies OFF. Fig. 7 shows a typical bump switch.

Figure 1: Arduino Mega 2560

Figure 2: Ultrasonic Sensor

Figure 3: IR Sensor

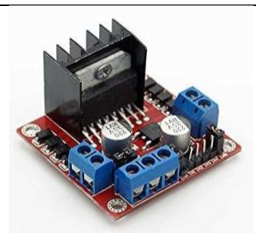


Figure 4: Motor Driver L298

Figure 5: Bluetooth Module HC–05

Figure 6: DC Geared Motor

Figure 7: Bump Switch

II. BLOCK DIAGRAM

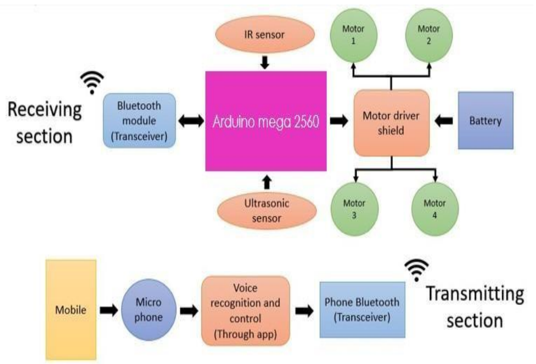


Figure 8: Block Diagram

Fig. 8 shows the block diagram of a voice controlled robotic car. There are two sections in the block diagram representation: Transmitting section and the Receiving section.

Transmitting section includes a mobile phone which is Bluetooth enabled. An app is built through MIT app inventor to control the robot through voice commands. The receiving section has the voice controlled robotic car and has Bluetooth module in it. The Bluetooth module receives the instructions from mobile app and sends these instructions to Arduino Mega. An Arduino code is developed to take voice commands from the Bluetooth module. Arduino Mega is connected to L298 motor driver, Ultrasonic sensors and IR sensors. Motor driver is connected with four gear motors. Ultrasonic sensor is used for obstacle avoidance and IR sensor is used for edge detection

III. METHODOLOGY

a) App Design

An App was built using the MIT App Inventor, this android app connects to the circuit through Bluetooth and can send voice commands to the robotic car. There are two steps to build an app in MIT App Inventor, Backend app design and the front end app design. a) Backend Program for the App: Backend programming is where the functionality of the app is described through a programming language (Blocks in MIT app inventor). Fig. 9 shows the

E-ISSN: 2584-0657

backend code of the voice controlled android app. Through this code the app can perform the following functions:

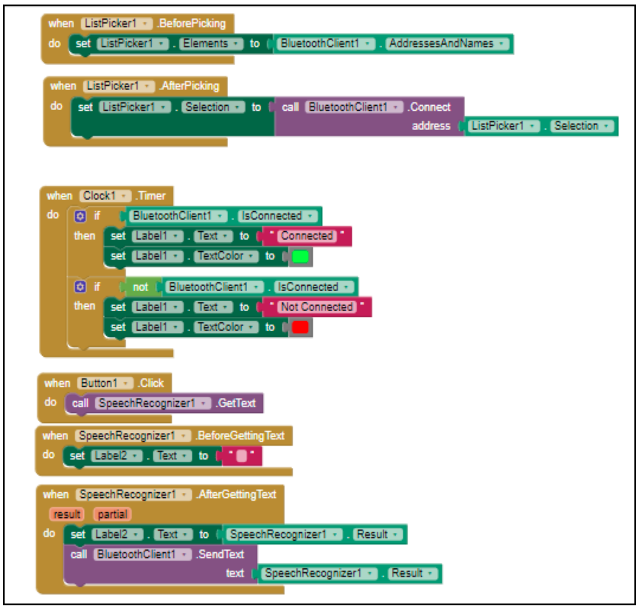


Figure 9: Backend Program of the App

1. Show nearby Bluetooth devices.
2. Connect to nearby Bluetooth device.
3. Show whether it is "Connected" or "Not Connected".
4. Take voice commands and transfer it to Bluetooth module.
5. Show the command it has received from voice as text using speech to text function.

b) App Frontend

The App frontend is the outer look of an app. When a user installs the app, the interface he sees is called front end.

Figure 10(a)

Fig. 10 (a) shows how the app looks like for a user when Bluetooth is not connected. The Bluetooth button should be clicked to connect to nearby Bluetooth device.

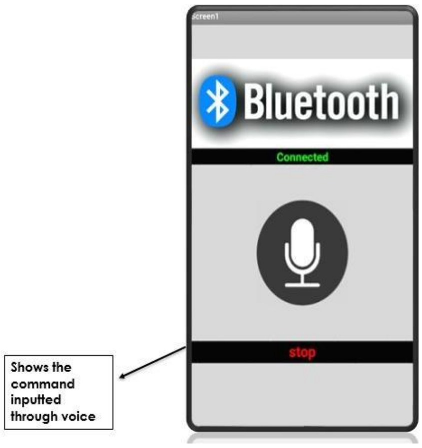


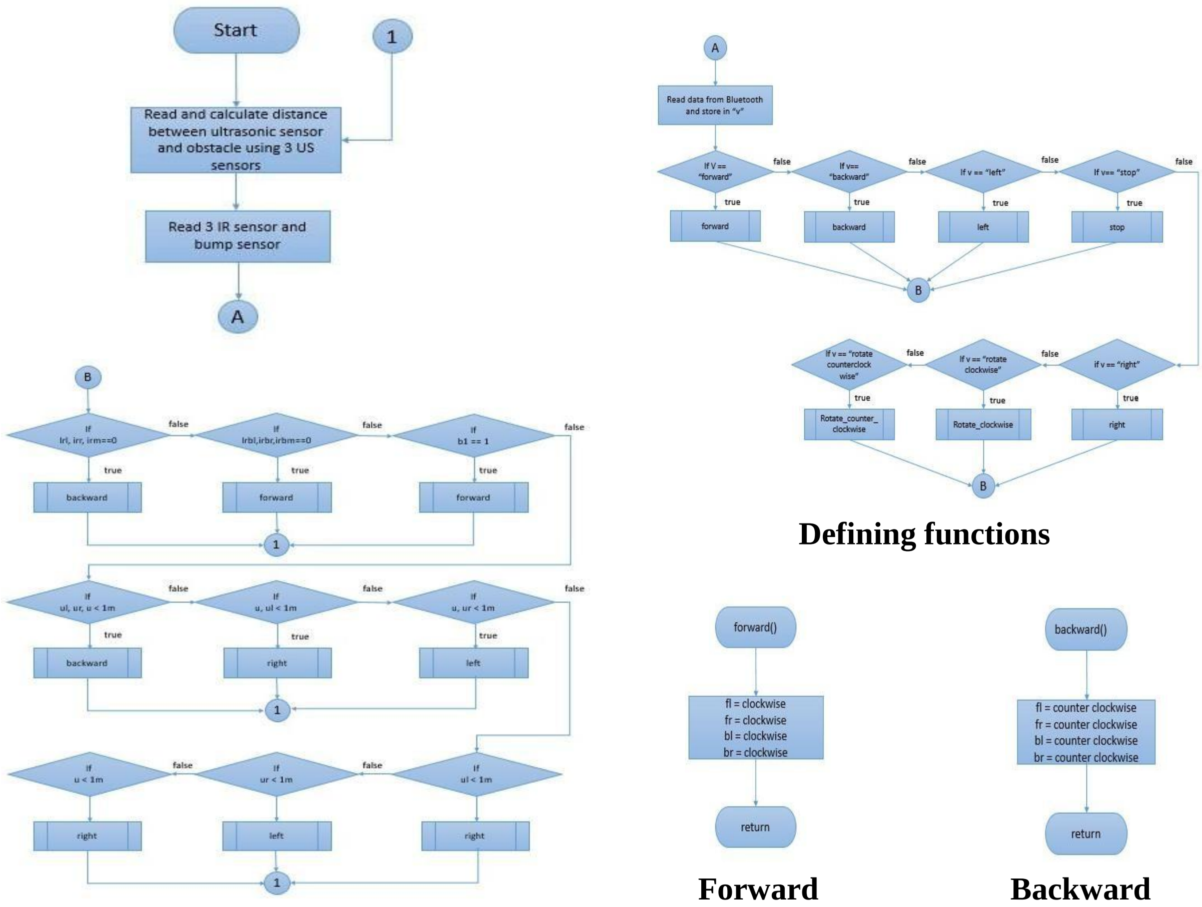
Figure 10(b)

Fig. 10 (b) shows how the app looks like for a user when Bluetooth is connected. After connecting, voice commands can be given by pressing the microphone button. The voice command given is then sent to connected Bluetooth module. The voice is converted into text by speech to text function and is displayed

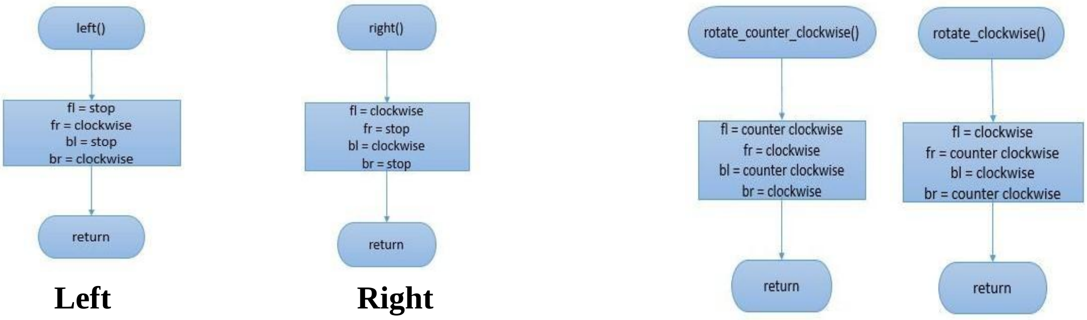
IV. Flowchart and Simulation Model

A. Flowchart of Model

The complete flowchart of the model is as shown in the Fig. 11. Same flowchart is followed for both Arduino Mega code and Lua code used for Coppeliasim.



E-ISSN: 2584-0657



Left **Right** **Clockwise** **Counter Clockwise**

Figure 11: Flowchart of the model

B. Proteus Model

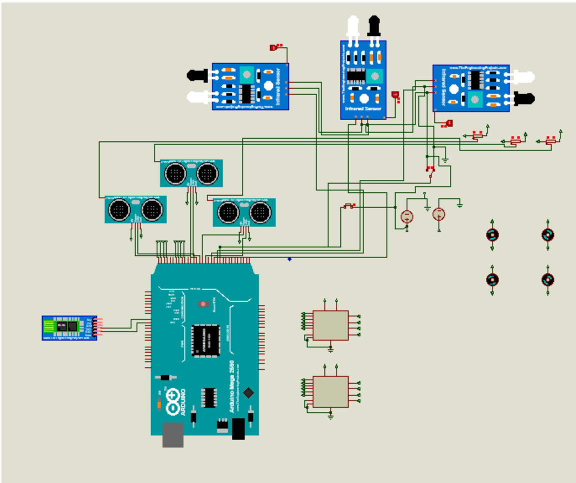


Figure 12: Circuit in Proteus Software

Fig. 12 shows the circuit in Proteus software. The circuit consists of the following components.

1. Arduino mega 2560 – 1 unit
2. Ultrasonic sensor – 3 units
3. IR senor – 3 units
4. Motor driver L298 – 2 units
5. Bluetooth module HC-05 – 1 unit
6. Logic state – 3 units
7. Potentiometer (1kΩ) – 3 units
8. Voltage supply – 2 units
9. DC geared motor – 3 units
10. Push button – 1 unit

Edge cannot be shown in simulation so logic states are used, which is connected to test pin. The test pin will be at logic high by default which in turn tells us that edge is not detected and when edge is detected, it will turn to logic zero. Similarly, distance of the obstacle varies depending on resistance of potentiometer. So that robotic car can feel that has encountered an obstacle.

Two motor drivers are used to control four DC geared motors. Three Ultrasonic sensors (Left, Middle and Right) are used to detect the obstacle. Three IR sensors (two in the front, one at the back) are used to detect edge in front and back of the robotic car. Bluetooth module is used to receive all the voice commands from the android app and send all instructions to Arduino Mega.

C. Coppeliasim Model

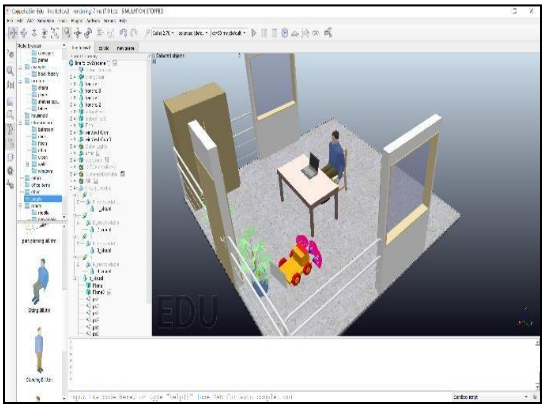


Figure 13: Robotic Car Model in Coppeliasim

Coppeliasim is a very efficient robot simulator. It has wide range of elements, sensors, modules and it supports ROS (Robot Operating System). The tool was used to test algorithm of the robot car. The module was created in solid works and then converted it into Universal Robot Description Format (URDF). It was then imported to Coppeliasim. The module uses three proximity sensors (cone type with max distance of 0.2 m and offset of 0.05 m, which act as ultrasonic senor) used to avoid obstacle and there are six proximity sensors, three at the front and three at the back (ray type with max distance of 0.4 m and offset of 0.01 m, which act as IR sensor) and finally a plane back is placed in robotic car which functions as a bump sensor. A child script (non threaded) is created and code is written for it. Fig. 13 shows the robotic car creation using Coppeliasim environment

V. RESULTS AND DISCUSSION

Once the circuit is built, the Bluetooth module is connected to the android app through Bluetooth. Then instructions such as left, right, forward, backward, turn clockwise, turn counter clockwise are given as voice commands through mobile app. The Bluetooth module senses the signals from mobile app and pass instructions to Arduino Mega to perform respective tasks. Table I indicate the working of differential drive with DC Geared Motor.

Table 1: Working of Differential Drive FED DC Geared Motors

Instructions	Fore left motor	Fore right motor	Rear left motor	Rear right motor
Forward	CW	CW	CW	CW
Backward	CCW	CCW	CCW	CCW
Left	Stop	CW	Stop	CW
Right	CW	Stop	CW	Stop
Rotate clockwise	CW	CCW	CW	CCW
Rotate counter clockwise	CCW	CW	CCW	CW

Here CW – Clockwise, CCW – Counter Clockwise

Table II indicate the working of IR sensor and bump sensor.

Table 2: Working of IR Sensor and Bump Sensor

IR sensor	Left sensor	Right sensor	Back sensor	Push button	Function
1	Low	Low	High	Off	Backward
2	High	High	Low	Off	Forward
3	High	High	High	On	Forward

Three IR sensors are used in the model which checks for any edge and turns accordingly. There is a push button in the circuit which function as bumper circuit. When a small amount of force is applied to this bumper switch, robotic car starts moving in the direction opposite to the active bumper switch.

Table III indicate the working of Ultrasonic Sensor There are three ultrasonic sensors in the model which checks if any obstacle is front of car. Ultrasonic sensors used in robotic car function only when distance between obstacle and sensor is less than 100cm.

Table 3: Working of Ultrasonic Sensor

US sensor	Left sensor	Mid sensor	Right sensor	Function
1	<100cm	<100cm	<100cm	Backward
2	<100cm	<100cm	>100cm	Right
3	>100cm	<100cm	<100cm	Left
4	<100cm	>100cm	>100cm	Right
5	>100cm	<100cm	>100cm	Right
6	>100cm	>100cm	<100cm	Left

VI. CONCLUSION

This paper presents the design and simulation of voice controlled robotic car. IoT is a rapidly growing field. This project is a proof of what IoT can create. An android app was built to give voice commands to the robotic car. A model was built in proteus, and results were verified for correctness. Another model was built in Coppeliasim which showed the model in the real world. This robotic car not only follows voice instructions but also avoids obstacles and avoids edges. This model has a variety of applications in the present market.

VII. FUTURE ENHANCEMENTS

The future scope of the work includes introduction of LIDAR (Light Detection and Ranging) in the circuit for automatic functioning and build the hardware prototype of the model for applications such as wheel chairs, and automatic vacuum cleaners, etc.

IX. REFERENCES

[1] Meghana, M., et al. "Hand Gesture Recognition and Voice Controlled Robot." *Materials Today: Proceedings*, vol. 33, 2020.

[2] Paul, Souvik, et al. "Arduino based, Bluetooth Controlled Rc Car." *International Journal of Computer Science & Communication*, vol. 11, no. 1, 2019, pp. 7-13.

[3] Mahesh Reddy, P., et al. "Intuitive Voice Controlled Robot for Obstacle, Smoke and Fire Detection for Physically Challenged People." *International Conference on Trends in Electronics and Informatics*, 2020.

[4] Ramjan Begum, M., et al. “Design and Development of Dual Axis Control Robot for Writing Robot through Speech Recognition.” *International Research Journal of Modernization in Engineering Technology and Science*, 2020.

[5] John, Linda Mary, et al. “Voice Control Human Assistance Robot.” *National Conference on Technical Advancements for Social Upliftment*, 2020.

[6] Mishra, Anurag, et al. “A Voice-Controlled Personal Assistant Robot.” *International Conference on Industrial Instrumentation and Control*, 2015.

[7] Louis, Leo. “Working Principle of Arduino and using it as a Tool for Study and Research.” *International Journal of Control, Automation, Communication and Systems*, vol. 1, no. 2, 2016.

[8] Thivagar, T., and A. Sriram. “Hand Gesture and Voice Controlled Smart Vehicle.” *International Journal of Modern Science and Technology*, vol. 5, no. 6, 2020.

[9] Saravanan, M., et al. “Arduino Based Voice Controlled Robot Vehicle.” *IOP Conference Series: Materials Science and Engineering*, 2020.

[10] Rajput, Harshada, et al. “Voice- Based Home Automation System Using Raspberry Pi.” *International Research Journal of Engineering and Technology*, vol. 5, no. 4, 2018.

[11] Malav, Vaibhav, et al. “Research paper on Bluetooth based Home Automation using Arduino.” 2019.

[12] Desai, Shrikar, et al. “Voice Controlled Robotic Car.” *International Journal of Advanced Research in Science, Communication and Technology*, vol. 4, no. 3, 2021.

[13] Verma, Anjali, et al. “Voice Control Robot Using Arduino.” *International Research Journal of Modernization in Engineering Technology and Science*, vol. 2, no. 4, 2020.

[14] Timsina, Abikshit, et al. “Voice Command Robotics Car.” *International Research Journal of Engineering and Technology*, vol. 8, no. 4, 2021.