Robotic car controlled by voice

Ditipriya Seal[1,*], *Himeli Pandit*[1], *Lagnajita Shome*[1], *and Niyali Mukherjee*[4]

[1] *Department of Basic Science and Humanities, Institute of Engineering and Management, Kolkata-700091.*

Email: i.5.ditipriya@gmail.com

Abstract

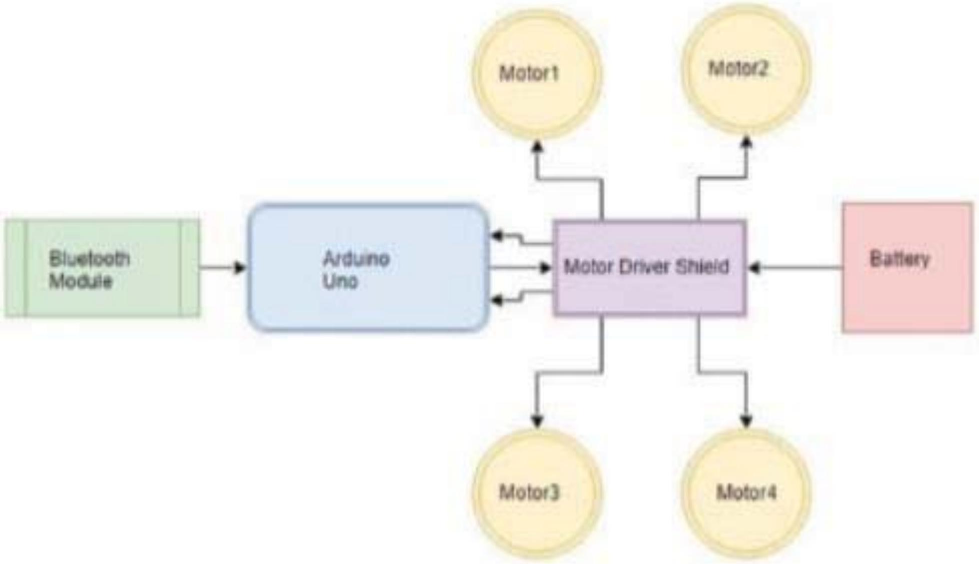
Robot reduces the efforts being put by humans in their day to day works. This paper is about to develop voice-controlled robotic car. This robotic vehicle helps to control robot through voice commands which is being received via android application. In the receiver part there would be a Bluetooth transceiver module which will receive the commands and will pass it on to the robotic car through Arduino. The robotic car will move in all directions according to voice commands given by the user. After processing the speech into text it will convert into text which will led to the necessary motion instructions are given to mobile platform via RF link. Ultrasonic sensors interfaced with Arduino will help in automatic slowdown whenever it sees an obstacle. In future technology this technique may prove to be a milestone in vehicle and for long distance range, Dual tone Multi Frequency (DTMF).

Keywords: *Bluetooth transceiver, Arduino, RFlink, vehicle automation, DTMF*

1. Introduction

The voice-controlled robot is a mobile robot, which listens and acts on the specific voice commands given by the user. In this project, we aim to control the movements of the vehicle wirelessly via voice commands given by the user. The objective of this project is to make the robotic car move forward, backward, left and right as per the commands given by the user. An Android with an speech recognition app is used in the transmitter end, and the app is to be connected with a Bluetooth HC-05/HC-06.The voice commands are detected by the phone microphone and the application running in the android, converts the speech to text, which is then sent to the receiver side via Bluetooth. Character by character, it is accepted from the serial buffer sent by the app and the string is formed by combining them. Then it is to be compared to the command, if it matches, the command is to be carried out. In this embedded systems project, we make a 4-WD robotic car which we can control using the mobile application. Application listens and sends the instruction to the Arduino using Bluetooth and thereafter the Arduino performs the specified operation. Voice recognition application is not 100% correct. The application is sensitive the surrounding noises. But we design our own application which can ignore the surroundings and can receive our own voice only. The basic knowledge used to do this project is robotics. Android speech recognition app used here was developed using MIT App Inventor. The basic components required in this project are- Arduino Uno, HC-05 Bluetooth module, L293D motor driver. Text received via Bluetooth is forwarded to the Arduino Uno Board using UART serial communication protocol. Arduino Program voice_ctrl.ino checks the text received and if it is a matching string, Arduino controls the movements of the robot accordingly. HC-05 Bluetooth module, used in our project is an easy to use Bluetooth Serial Port Protocol (SPP) module, designed for transparent wireless serial connected setup.

Arduino IDE 1.6.5 is used for programming in this project. Steps to program Arduino are as follows- 1.) Select the proper COM Port and board from Tools menu in the IDE. 2.) Upload source code voice_ctrl.ino to the board.



2. Description of Robot

2.1 Components

- Arduino Uno
- Breadboard
- Motors x2
- Wheels x2
- Chassis (of appropriate size)
- Voltage Regulator LM7805
- L293D
- 12V battery (power source)
- Jumper wires
- Bluetooth Module HC-05

2.2 Arduino uno

Arduino Uno is used here. It is cheap, easy to use and acquires less space as we have to place everything on the chassis.

The Uno is a micro-controller board based on the ATmega328P. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the micro-controller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The ATmega328 has 32 KB (with 0.5 KB occupied by the boot loader). It also has 2 KB of SRAM and 1 KB of EEPROM. The Uno has a number of facilities for communicating with a computer, another Uno board, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The 16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, an .inf file is required. The Arduino Software (IDE) includes a serial monitor which allows simple textual data to be sent to and from the board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

2.3 Transmitter and Receiver

On the transmitter section, various commands are given to the mobile app through the microphone of the mobile. This mobile is connected to the robotic car via Bluetooth. The mobile app used is programmed in such a manner that the voice commands given to the mobile are received by the microphone and these analog voice commands are converted to digital word sequences (A to D conversion). These stored sequences are then transmitted to the robot via Bluetooth transceiver module and consequently sent to the transceiver controller (MAX 232).

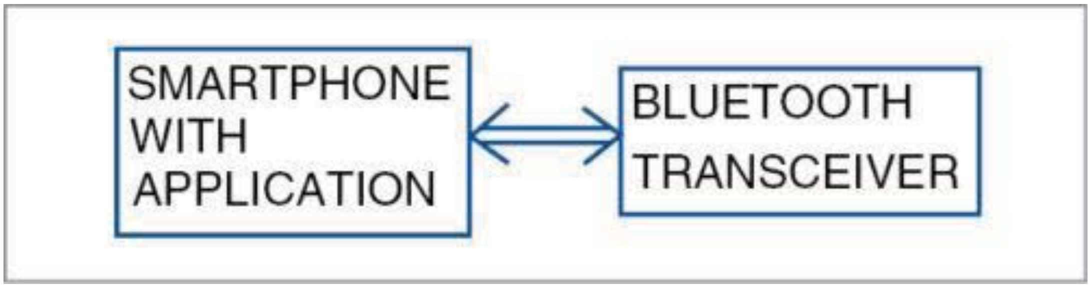


Fig 1. Block diagram of transmitter side.

The received signal is decoded by MAX 232 transceiver. It is also used for serial communication with the Bluetooth module. The controller compares these digital signals with the stored programme commands inside and convert the digital signal into voice strings. The voice strings are thereafter used to run the servo motors for the required interval of time.

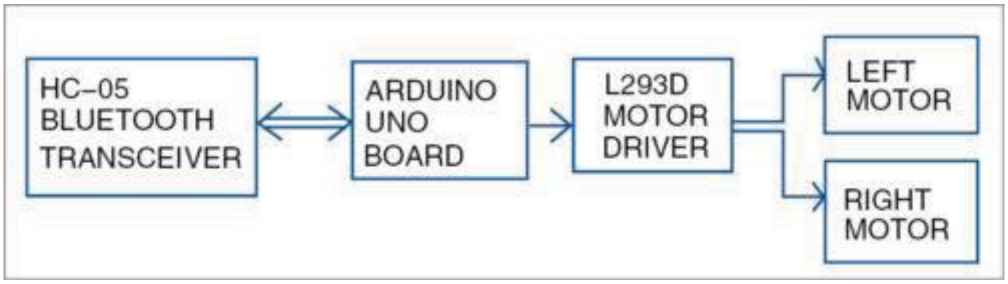


Fig. 2. Block diagram of the receiver side

2.4 L239D motor driver

It is a dual H-Bridge high current motor driver IC .This is used in the project since digitals pins of arduino cannot source enough current to drive the motors of the robotic car. H-bridges are also very useful in controlling the direction of rotation of the motor. It enable pins 1 and 9 of the IC, being active high, are connected to 5V.Four output pins of L239D IC are connected to motors M1 and M2 on the receiver side.

2.5 Circuit and working

The circuit diagram of the receiver side of the robot is shown in this figure-

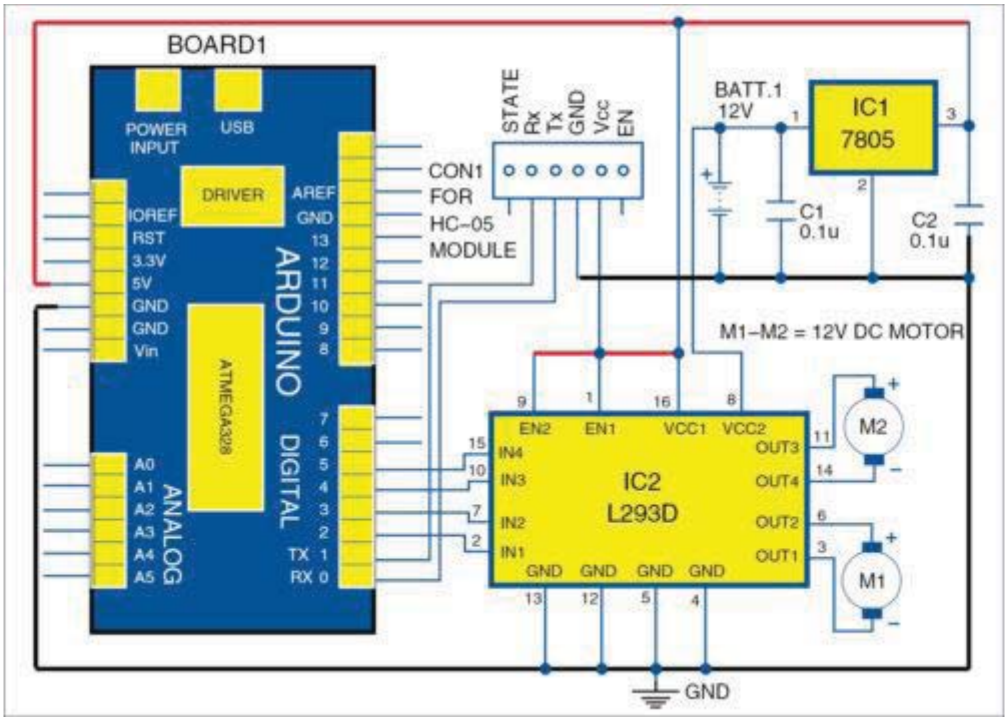
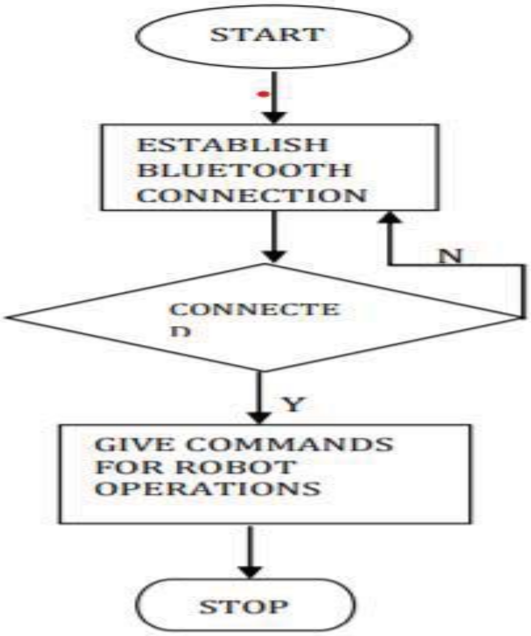


Fig. 3. Circuit diagram of receiver circuit

2.6 Flowchart


START
ESTABLISH BLUETOOTH CONNECTION
N
CONNECTE D
Y
GIVE COMMANDS FOR ROBOT OPERATIONS
STOP

2.7 **Algorithm**

1. Start
2. Establishing Bluetooth connectivity between Android Application and the Bluetooth module on the robot.
3. ToCheck whether the device is connected.
4. If connected, provide the pre-defined instructions or commands to the micro-phone of the mobile handset.
5. The voice commands should be trained to the EasyVR module.
6. Then the stored voice commands are represented in the form of binary numbers such as move forward – 001, move backward – 010 etc.
7. These binary values are transmitted via zigbee module which is a transceiver.
8. The transmitted binary values are then received by another zigbee module which is present on the receiver side.
9. Microcontroller will take those binary values and performs action (servo motors) according to the binary values.
10. If it is failed to connect at step 3 than again go to step 2.
11. Stop.

2.8 Code

```
int motor_input1=11;
int motor_input2=10;
int motor_input3=5;
int motor_input4=6;
String voice;
void setup()
{
  Serial.begin(9600);
  pinMode(motor_input1, OUTPUT);   //RIGHT MOTOR
  pinMode(motor_input2, OUTPUT);   //RIGHT MOTOR
  pinMode(motor_input3, OUTPUT);   //LEFT MOTOR
  pinMode(motor_input4, OUTPUT);   //LEFT MOTOR
}
void loop()
{
  while(Serial.available()>0)
  {
    delay(10);
    char c=Serial.Read();
    if(c=='#')
    {
      break;
    }
    voice+=c;
    }
   if(voice=="forward"){
    digitalWrite(motor_input1, LOW);
    digitalWrite(motor_input2, HIGH);
    digitalWrite(motor_input3, LOW);
    digitalWrite(motor_input4, HIGH);
     delay(5000);
    }
  else
    if(voice=="back"){
    digitalWrite(motor_input1, HIGH);
    digitalWrite(motor_input2, LOW);
    digitalWrite(motor_input3, HIGH);
    digitalWrite(motor_input4, LOW);
    delay(5000);}
  else
   if(voice=="left"){
    digitalWrite(motor_input1, LOW);
    digitalWrite(motor_input2, HIGH);
    digitalWrite(motor_input3, HIGH);
    digitalWrite(motor_input4, LOW);
    delay(800);
    }
  else
   if(voice=="right")
{
    digitalWrite(motor_input1, HIGH);
    digitalWrite(motor_input2, LOW);
    digitalWrite(motor_input3, LOW);
    digitalWrite(motor_input4, HIGH);
    delay(800);  }
   if(voice.length()>0)
    {
      Serial.println(voice);
       voice="";
    digitalWrite(motor_input1, LOW);
    digitalWrite(motor_input2, LOW);
    digitalWrite(motor_input3, LOW);
    digitalWrite(motor_input4, LOW);
    }
```

3. Conclusion

The project "Voice Control Robotic Car" that is developed in this paper completely reforms the robotic vehicle and gives it a new dimension. The voice commands given are easily recognised and processed in real time using an offline sever. Actions taken by the robotic car are very smooth and swift. The speech signal commands are directly communicated to the server over a wired connection. The accent of the speaker does not affect the operation of the robotic car as the voice commands are processed using a cloud server which functions irrespective of the accent of the speaker.

The robotic car is based on a microcontroller-based platform and it can be aware of its current location. Performance evaluation is carried out with encouraging results of the initial experiments. The areas to be further explored are the effects of the distance between the mouth and the microphone, the performance of the robot, the effect of the noise on the speech to text conversion.

We have used both the Arduino Due and the Arduino Uno and leant a bit of basics of it, in due time we will learn more and develop more new applications. We learnt a bit about how to make use of the IR Sensors in robots and also got an idea that the radiation is emitted in order to sense some aspects of the surrounding.

The robotic car developed has potential applications starting from chemical industries to our modern homes. Further enhancement in our project can be also used for home security and military purposes where commands can be given to the robotic car without risk by increasing the range and by installing cameras. Possible improvements are also discussed towards potential application of the "Voice Control Robotic Car". We will be upgrading ourselves to the latest technologies in near future and are also encouraged to learn more through our project" Voice Control Robotic Car".

4. Acknowledgement

We would like to express special thanks and gratitude to our Physics teachers who gave us the golden opportunity to do this wonderful project on the topic “VOICE CONTROLLED ROBOTIC CAR”.

Secondly, we would like to thank our parents and friends who helped us a lot in finalizing this project within the limited time frame.

References

[1] https://www.electronicsforu.com

[2] https://microcontrollerslab.com

[3] “Programming Arduino Getting Started with sketches.MC Graw Hill

[4] IEEE Conference Publication

[5] https://www.robotshop.com

[6] DavidOrenstein, “People with paralysis control robotic arms using brain”, http://news.born.edu/articles/2012/05/braingate2.