Voice Control VBOT

Shweta Prakash Aswani  
Computer Science  
State University of New York at  
Binghamton  
Binghamton New York United States  
saswani1@binghamton.com

Varun Katragadda  
Computer Science  
State University of New York at  
Binghamton  
Binghamton New York United States  
vkatrag2@binghamton.edu

ABSTRACT
We present VBot, a voice control robot which can be controlled using specific voice commands. Speech to Text functionality is used to convert voice commands to text which are then sent to Arduino through Bluetooth communication. Voice commands have become a primary way to interact with devices after the development of devices like Alexa and Google Home. Through this project, control of the robots becomes handy. The voice commands are perceived using an Android application which converts speech to text. This text is in the form of a string. It is then sent to Arduino via the Bluetooth module. Then the code compares it to the command. If it matches, the command is carried out. For example, if the string we received is "ahead", the robot turns ahead. Also, it measures the speed, distance and RPM when the car is being operated. Here a 12C 16x2 LCD module is used to display the values of the speed, distance and rpm. This LCD module is integrated with the Motor Driver Shield of the Arduino.

1 Hardware Specifications
We have the following hardware in our project:

1.1 Arduino UNO
Arduino Uno is a microcontroller board based on the ATMega328P (datasheet). 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 microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

1.2 Motor Driver Shield L293D
L293D is a monolithic integrated, high voltage, high current, 4-channel driver Can drive 4 DC motors or 2 stepper motors or 2 Servo. Up to 4 bi-directional DC motors with individual 8-bit speed selection. Up to 2 stepper motors (unipolar or bipolar) with single coil, double coil or interleaved stepping. 4 H-Bridges: per bridge provides 0.6A (1.2A peak current) with thermal protection, can run motors on 4.5V to 10V DC.

1.3 Bluetooth Module HC-05
This module which can add two-way (full-duplex) wireless functionality to your projects. You can use this module to communicate between two microcontrollers like Arduino or communicate with any device with Bluetooth functionality like a Phone or Laptop.

1.4 Servo motor
A servo motor is a rotary actuator that allows for precise control of angular position. It consists of a motor coupled to a sensor for position feedback. It also requires a servo drive to complete the system. The drive uses the feedback sensor to precisely control the rotary position of the motor.
1.5 **HC-SR04 Ultrasonic sensor**
It works by sending sound waves from the transmitter, which then bounce off of an object and then return to the receiver. You can determine how far away something is by the time it takes for the sound waves to get back to the sensor.

![HC-SR04 Ultrasonic sensor](image)

1.6 **I2C LCD 16x2**
This type of LCD is ideal for displaying text and numbers. The I2C LCDs come with a small add-on circuit mounted on the back of the module. This module features a PCF8574 chip (for I2C communication) and a potentiometer to adjust the LED backlight. The advantage of an I2C LCD is that the wiring is very simple. You only need two data pins to control the LCD.

![I2C LCD 16x2](image)

1.7 **LM393 Speed sensor**
Widely used in motor speed detection, pulse count, the position limit, etc. The DO output interface can be directly connected to a micro-controller IO port, if there is a block detection sensor, such as the speed of the motor encoder can detect.

![LM393 Speed sensor](image)

1.8 **Plastic Tire Wheel with DC 3-6v Gear Motor**
These parts were used to build the customized car with 4 DC motors on which the remaining parts were mounted for the voice control and distance calculation functionality.

![Plastic Tire Wheel with DC 3-6v Gear Motor](image)

1.9 **18650 Batteries for the continuous power supply**
The Voltage supplied by the 18650 Battery is 3.6V. It has a capacity to hold 3400 mAh. The charging voltage is 4.2V and the time taken for charging each battery is approximately 2 hours. We are using three batteries for this project so that there is a continuous power supply provided to all the components in the car.

![18650 Batteries](image)

2 **Integrating All together**
Here is the final picture of the car after mounting all the parts and connecting them to Arduino Uno with a motor shield mounted on the top of the Arduino Uno.
3 Circuit Diagram

3.1 Integrating the four DC Motors
All the four DC motors were connected to the L293D motor shield to their respective M1, M2, M3 and M4 ports. These ports will supply the power to the DC motors from the L293D motor shield.

3.2 Integrating the HC-05 Bluetooth Module
The HC-05 Bluetooth module has four pins that were connected to the Arduino Uno via the L293D motor shield. The GND and VCC are connected respectively to the power slots on the L293D motor shield. TX and RX pins of the Bluetooth module are connected to the RX and TX pins on the motor shield for transmitting and receiving of the serial data.

3.3 Integrating the LM393 Speed Sensor Module
The LM393 Speed Sensor have three pins that were connected to the motor shield. The GND and VCC are connected to the respective power supply pins on the motor shield. The output pin of the LM393 is connected to the PIN2 on the Arduino for getting the data calculated by the LM393 sensor.

3.4 Integrating the I2C_16X2 LCD display
The I2C_16X2 LCD display have four pins to be connected to the L293D motor shield. The GND and VCC pins are connected respectively to the power supply pins over the motor shield. The SDA pin is connected to the Analog pin A4 and the SCL pin is connected to the Analog A5 pin respectively over the motor shield.

3.5 Integrating the Servo Motor and HC-04 Sensor
The HC-04 sensor is mounted over the servo motor and the servo motor is connected to the SER1 pin output respectively over the L293D motor shield. The servo motor comes into action when the car is rotating left or right.

3.6 Integrating the 18650 Batteries
The 18650 batteries are mounted in the three holder battery case and the positive and negative terminals are connected respectively to the L293D motor shield for the continuous power supply.

4 Logic behind measuring speed with LM393 Sensor
From the sensor mounting set-up you should be aware that the LM393 speed sensor measures only the gaps present in the grid plate. While mounting, it should be sure that the wheel and the grid plate rotates in the same speed. Every time the gap in the grid plate is detected an interrupt will be triggered and the code in the ISR (Interrupt service Routine ) is executed. The speed is mainly calculated by calculating the time interval between two such triggers.

\[
\text{Time taken} = \text{current time} - \text{previous time}
\]

\[
\text{Timetaken} = \text{millis()} - \text{pevtime}; // \text{time taken in millsec}
\]

Once we have calculated the time taken, we can calculate the value of RPM using the formulae below, where \((1000/\text{timetaken})\) gives the RPS (revolutions per second) and it is multiplied by 60 to convert the RPS to RPM (revolutions per minute).

\[
\text{Rpm} = (1000/\text{timetaken}) \times 60
\]

There are 20 slots in the grid plate hence for measuring the time between two slot gaps will overload the microcontroller. Hence we measure the speed only at a full rotation of the wheel. Since two interrupts will be generated for every gap(one at the start and other at the end of the gap) we will get a total of 40 interrupts for the wheel to make one complete rotation. So we wait for 40 interrupts before we actually calculate the speed of the wheel. The code is shown below.

\[
\text{If(rotation}>=40) \{ \text{timetaken} = \text{millis()} - \text{pevtime}; \text{rpm} = (1000/\text{timetaken}) \times 60; \text{pevtime} = \text{millis()}; \text{rotation} = 0; \}
\]

5 Logic behind measuring the distance travelled by the car
We Know that the Arduino will sense 40 interrupts when the wheel makes one complete rotation. So for every one rotation made by the wheel, it is evident that the distance travelled by the wheel is equal to the circumference of the wheel. Since we already know the radius of the wheel, we
can easily calculate the distance covered using the below formula.

\[ \text{Distance} = 2\pi r \times \text{number of rotations} \]

\[ \text{distance} = (2 \times 3.141 \times \text{radius of wheel}) \times (\text{left intr/40}) \]

where the circumference of the wheel is calculated using the formula and it is multiplied by the number of rotations made by the wheel.

CONCLUSION

VBot is the car which is built by integrating all the different sensors perfectly to calculate the speed and distance travelled by the car and is controlled by the voice commands. There are a lot of challenges faced by the team in order to make it work perfectly. There was problems of insufficient power supply and also damaging the motor shield and different sensors by giving the high voltage. It took a while to calculate the exact power supply needed by the all the integrated components. There are also instances where we have damaged the motor shield while soldering it and have to get a new one. Coming to the future developments, we can attach the camera module to the VBot and can capture all the footage and then we can integrate the car with a Raspberry Pi and store all the data over the cloud and retrieve whenever we need.

ACKNOWLEDGMENTS

We would like to thank our Professor Mo Sha for giving us this chance to do research and implement the project successfully. We would also thank our Teaching Assistant Di Mu for helping us regarding the soldering part and the suggestions given by him. Finally we would like to thank the CS Department of Binghamton University for providing us the infrastructure and scope to build this project with ease.