ROBOTIC VEHICLE CONTROLLED BY VOICE COMMANDS

By

Arda Mantaş

Engineering Project Report

YEDİTEPE ÜNİVERSİTESİ

Faculty of Engineering

Department of Electrical and Electronics Engineering

Istanbul, 2019

ACKNOWLEDGEMENTS

In performing project, there are some people who helped author to complete his project. Author would like to show his appreciation to Ata Mert Erdihan, Turan Oğuzhan Koyunoğlu, Dila Alkır and Mehmet Yunus Doruk for their assistance and creative ideas. Likewise, a special thank you to Asst.Prof Gökhan Şahin for his help and help author to solve the problems from a different perspective. Finally, authors also thank the Yeditepe University for providing these facilities and opportunities.

ABSTRACT

The aim of this project is a robotic vehicle which is controlled by voice commands and bluetooth module through the application.

First of all, the application detects the voice commands and it converts these commands into text.

Thereafter, the commands which were translated into text, get transferred into the bluetooth module. It is important that there should be a connection between the commands and the code which was written in arduino sofware language application.

In conclusion, if the codes which were pre-defined via arduino programming language application, and the commands which were sent from the bluetooth module connected to the arduino card matches, the robot moves according to the command. On the other hand, if the code which was written in arduino software application and commands are incompatible, the robot cannot take action.

ÖZET

Uygulama üzerinden, bluetooth modülünün yardımı ile komutların anlaşılması ve robotun gönderilen komutları uygulaması bu projenin temel amacıdır.

İlk olarak, uygulama gönderilmiş olan ses komutlarını algılayarak bu komutları metin formatına dönüştürür.

Daha sonrasında, metin haline dönüştürülmüş olan komutlar bluetooth modülüne aktarılır. Gönderilen komut ile arduino programında yazılmış olan kodlar arasında ilişki olması önemlidir.

Sonuç olarak, gönderilen komut ile arduino ugulamasında önceden yazılmış olan kodun eşleşmesi durumunda robot harekete geçer. Öte yandan uyuşma olmaması durumunda robot hareket etmez.

Table of Contents

Table of Figures

LIST OF SYMBOLS/ABBREVIATIONS

AC	Alternative Current
DC	Direct Current
F/f	Farads
Ω	Ohm
R	Resistor
C	Capacitance
T.A.	Teaching Assistant
Q	Charge
A	Field/Area
d	Distance
V	Voltage
Z	Impedances
Hz	Frequency
m	Meter

1. INTRODUCTION

1.1 Wireless Communication

As mentioned in the summary section, the aim of the project is to activate the vehicle with voice commands. For this purpose, the importance of wireless communication should be mentioned.

Wireless communication plays an important role in everyday life. In addition to communication, wireless technology has become an significant part of daily activities. Wireless communication is transfer the data or information from one place to another. The information or data is transmitted from well-defined channels from several meters to hundreds of kilometers.

Wireless communication is diversified according to the transmission distance of information or data.

Wireless communication technologies are diversified as follows;

- Satellite communication
- Global Positioning System
- Radio Frequency Identification
- Cellular Communication
- Radio and Television Broadcasting
- Radar Communication
- Wi-Fi
- Bluetooth

1.2 Bluetooth Communication

In this project, communication with bluetooth module was chosen due to the fact that the transmission distance is within the desired value range and that it is cost-effective.

Bluetooth is a short-range wireless technology that enables devices such as computers and mobile phones to transmit data or voice wirelessly over a short distance. As with other types of wireless communication, bluetooth communication also uses the 2.4GHz band.

1.2.1 Advantages & Disadvantages of Bluetooth Communication

Bluetooth communication avoids interference from other wireless devices. Bluetooth is a technology that is open to development and easy to use. Also low energy consumption, very low-cost products according to the performance offers quite attractive options. Thanks to high encryption methods, it has an extremely robust security mechanism and network. Bluetooth brings independence in office and home solutions. Ends the cable dependence in case of travel or movement. Accessing the internet via radio frequencies with a bluetooth mobile phone and laptop away from home or the office is no longer a problem. These are the advantages of the Bluetooth communication.

On the other hand, the range is shorter than other types of wireless communications, up to about 30 feet, in addition to the room walls, floors and ceilings can cause a significant reduction in signal quality and range. Pairing can be difficult depending on the device type when attempting to connect.

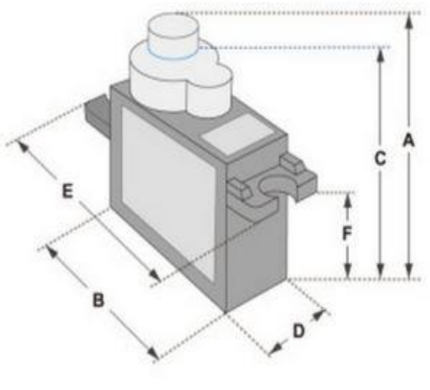
2. PROCEDURE

2.1 Components

Componenets were used in the construction of the project. These parts are ultrasonic distance sensor, bluetooth module, buzzer, arduino mega card, motor driver, motors, breadboard, jumpers and leds. The details of each part are given below in sections.

2.1.1 SG90 9G Mini Servo Motor

Servo motor allows the part to rotate at an angle of approximately 180 degrees(90 degree for each side). Although the voltage varies between 4.8 and 6, 5 volts are generally used. The following Figure 2.1[1] shows the specific features of this component.



Dimensions & Specifications
A (mm) : 32
B (mm) : 23
C (mm) : 28.5
D (mm) : 12
E (mm) : 32
F (mm) : 19.5
Speed (sec) : 0.1
Torque (kg-cm) : 2.5
Weight (g) : 14.7
Voltage : 4.8 - 6

Figure 2.1 dimensions & specifications of the servo motor

2.1.2 Buzzer

This circuit element can be considered as a speaker. For example, when turn right command is given, if there is an obstacle on the right side of the device, it will beep and notify the user that it cannot perform the command which is turn right. The following Figure 2.2[2] shows the specific features of this part.

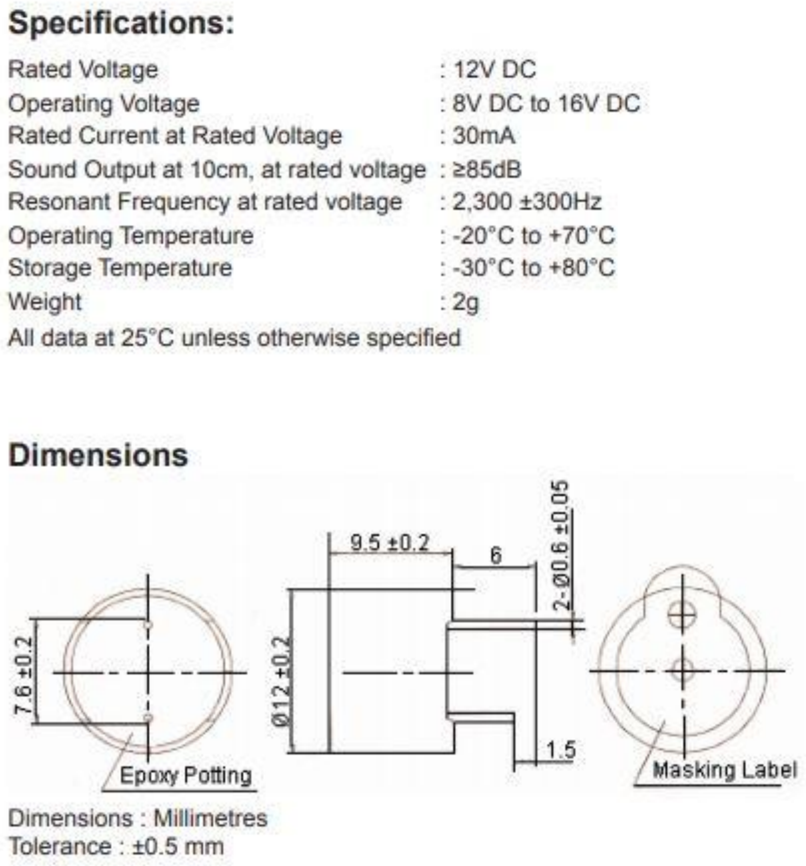


Figure 2.2 dimensions & specifications of the buzzer

2.1.3 Arduino Mega 2560 Card

The Arduino Mega 2560 is a microcontroller board which is contains 54 digital input / output pins (14 of them can be used as PWM output), 16 analog inputs, 4 UART (hardware serial port), 16 MHz crystal oscillator, one USB connection, a power input, an ICSP header and a reset button. All the parts it contains are shown in the Figure 2.3[3] and also specifications and dimensions are given in Figure 2.4[3] and Figure 2.5[3] respectively.

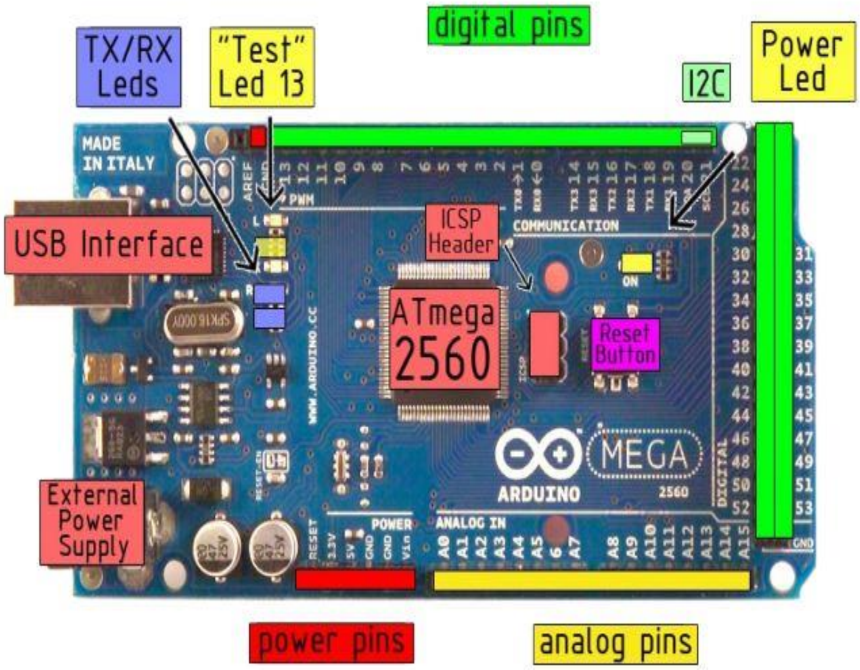


Figure 2.3 all parts of the arduino mega 2560 card

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (of which 14 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

Figure 2.4 specifications of the arduino mega board

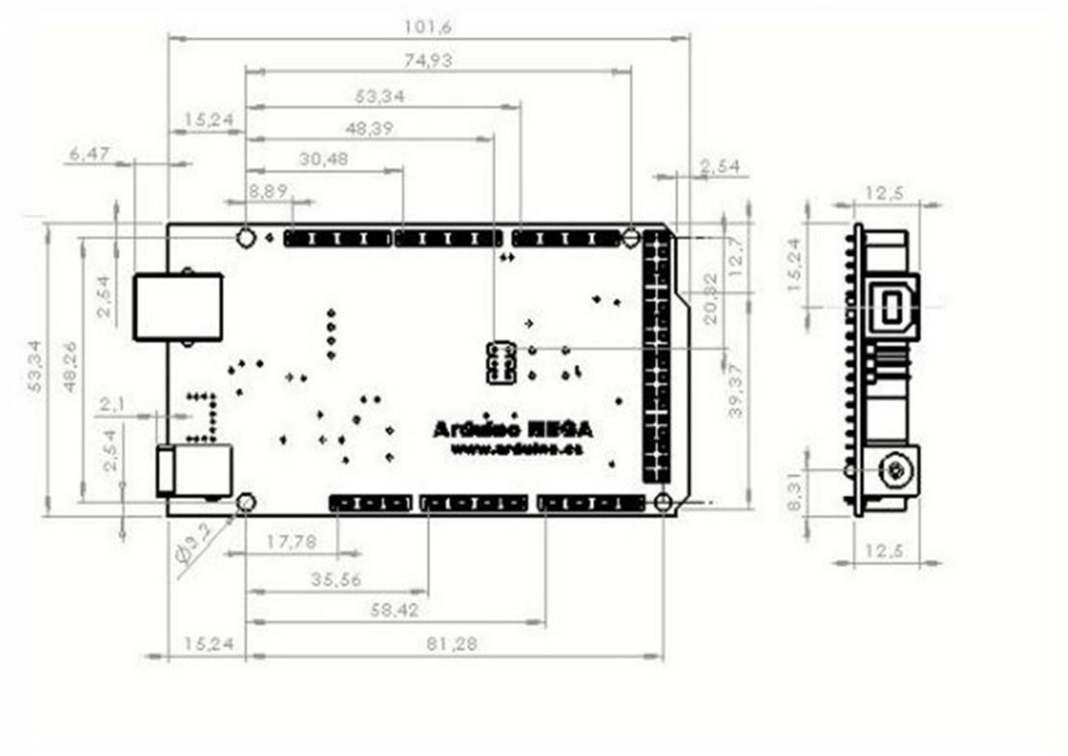


Figure 2.5 dimensions of the arduino mega 2560 card

2.1.4 Leds

LED, as a word - consists of the initials of the Light Emitting Diode. Led is a semi-conductive circuit element which is converting electrical energy into light. Dimensions and voltage values according to color types of led are given in Figure 2.6[4] and Table 2.7 respectively.

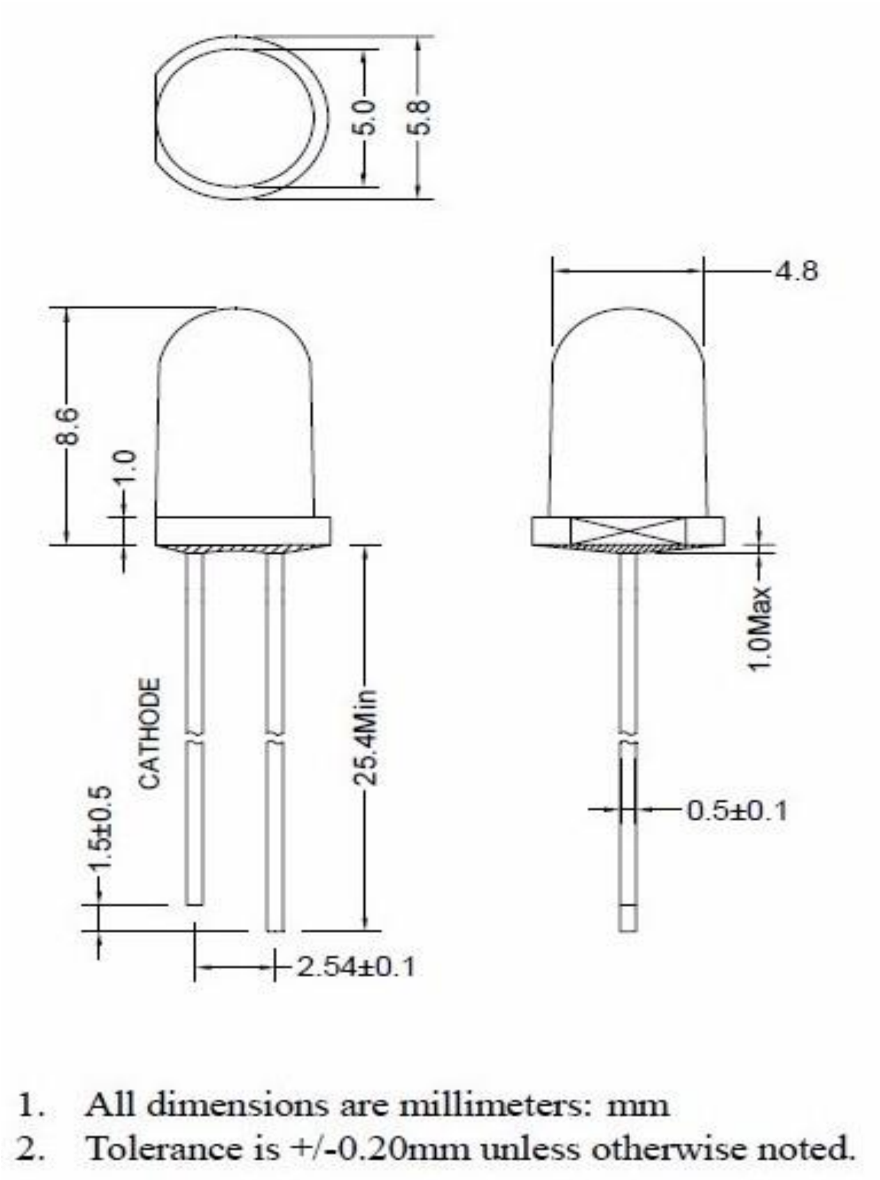


Figure 2.6 dimensions of led

Led Colour	**Forward Voltage**
Red	1.63-2.03V
Yellow	2.10-2.18V
Orange	2.03-2.10V
Blue	2.48-3.07V
Green	1.9-4.0V
Violet	2.76-4.0V
UV	3.1-4.4V
White	3.2-3.6V

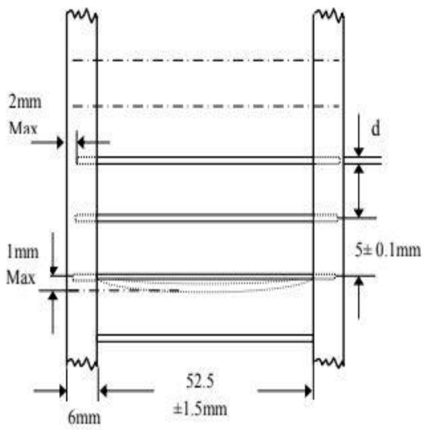
Table 2.7 voltage values according to colour types of led

2.1.5 Jumper Wire

In short, some kind of connection cables. To make a connection between breadboard and arduino is quite useful. According to the male and female inputs at the ends of it. Specifications are given in Figure 2.8[5]. Also there are 3 types of jumper cables as follows;

1) Male-Male
2) Male-Female
3) Female-Female

Jumper Wire Specification



Characteristic

Material	Tin plated soft Copper Wire
Conductor Resistance	0.54mΩ/cm
Conductivity	Min. 96%
Current Rating	6 Amps at 70°C for 0.50mm
	7 Amps at 70°C for 0.56mm
	7.5 Amps at 70°C for 0.60mm
	8.5 Amps at 70°C for 0.70mm
	10 Amps at 70°C for 0.80mm

Style

d ± 0.02mm	Standard Q'ty per Box or Reel
0.50	10,000
0.56	10,000
0.60	10,000
0.70	8,000
0.80	8,000

How to Order :

JW	56	T
Jumper Wire	50=0.50mm	T=Ammo Box
	56=0.60mm	R=Tape on Reel
	60=0.60mm	
	70=0.70mm	
	80=0.80mm	

Figure 2.8 specifications of the jumper wire

2.1.6 Breadboard

It is the tool that enables to establish circuit with parts without soldering. Breadboard allows the user to test easily. In this way, an error which is possible to observe whether or not by checking the circuit connections. The internal structure is as shown in Figure 2.9[6].

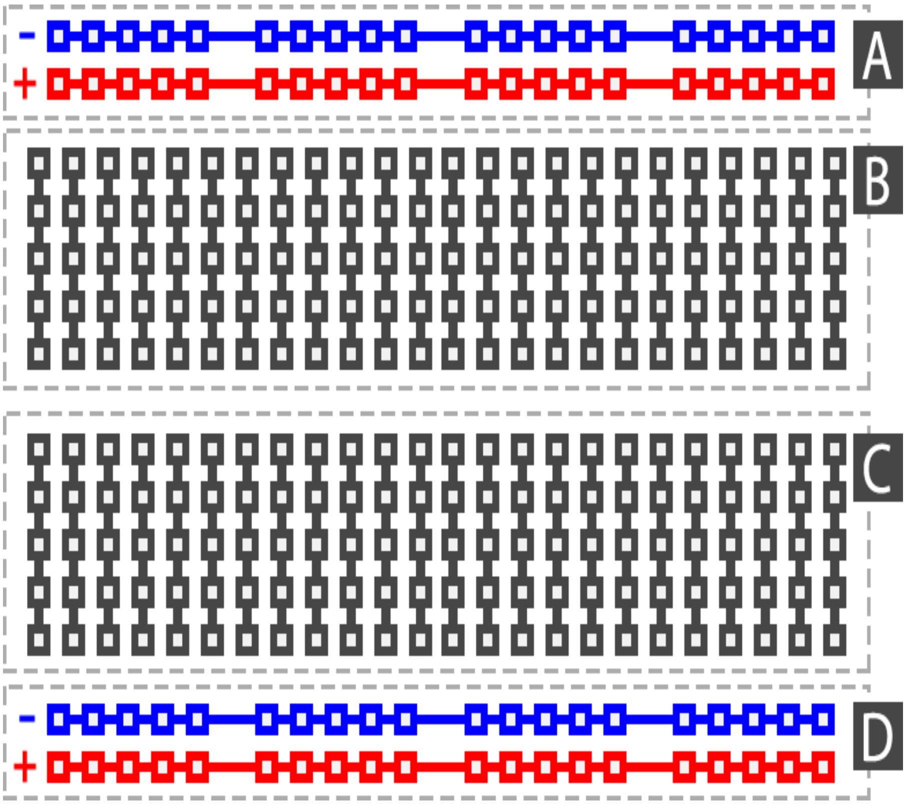


Figure 2.9 internal structure of the breadboard

2.1.7 HC-SR04 Ultrasonic Distance Sensor

HC-SR04 ultrasonic sensor is an input source that calculates the distance from the object to the another object by using sonar (Sound Navigation and Ranging). Such sensors are inspired by dolphins and bats. Product layout and specifications are given in Figure 2.10[7] and Figure 2.11[7] respectively.

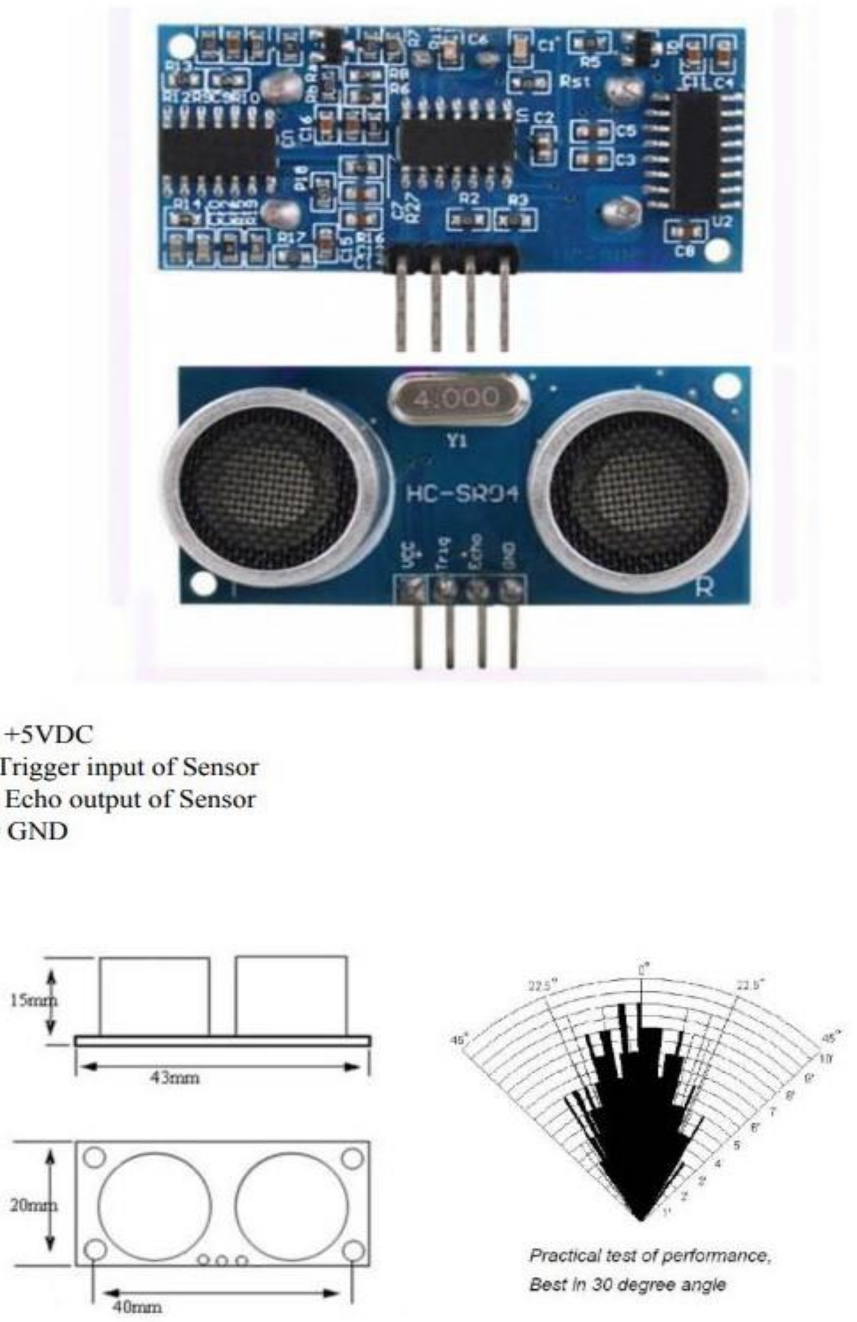


Figure 2.10 product layout of the ultrasonic distance sensor

Parameter	Min	Typ.	Max	Unit
Operating Voltage	4.50	5.0	5.5	V
Quiescent Current	1.5	2	2.5	mA
Working Current	10	15	20	mA
Ultrasonic Frequency	-	40	-	kHz

Figure 2.11 specifications of the ultrasonic distance sensor

2.1.8 HC-05 Bluetooth Module

HC-05 Bluetooth module can be used as master and slave. In other words, it can receive serial data and also transmit them. Communication is provided by the Rx (Receiver) and Tx (Transmitter) pins on it. Pin representation is given in Figure 2.12[8].

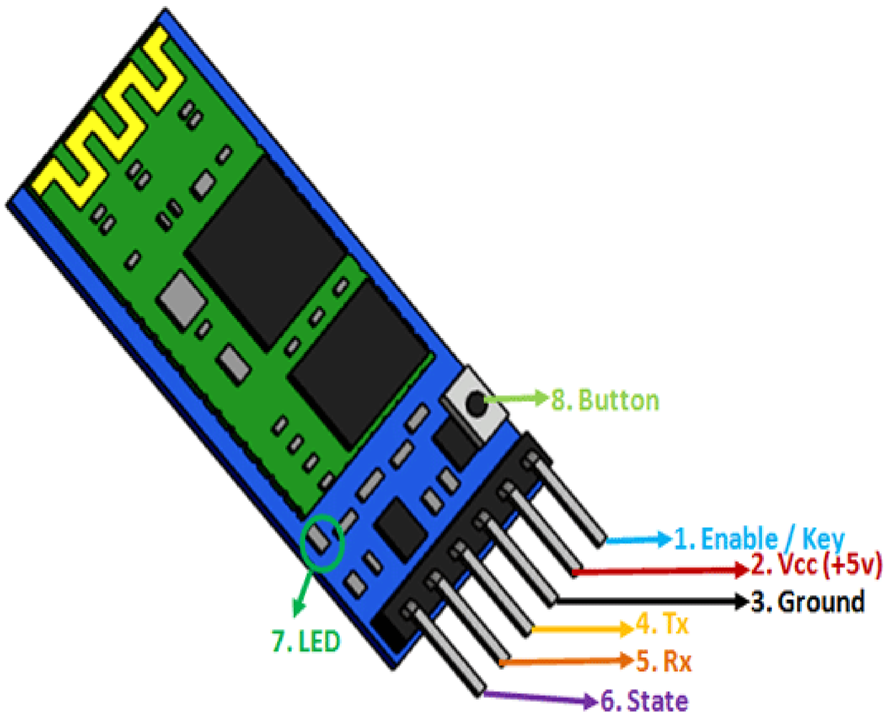


Figure 2.12 pin representation of the bluetooth module

2.1.9 DC Gearbox Motor

The dc motor is the part that gives the wheels the power required to arouse the vehicle. Technical details are shown in the Figure 2.13[9].

TECHNICAL DETAILS

- Rated Voltage: 3~6V
- Continuous No-Load Current: 150mA +/- 10%
- Min. Operating Speed (3V): 90+/- 10% RPM
- Min. Operating Speed (6V): 200+/- 10% RPM
- Torque: 0.15Nm ~0.60Nm
- Stall Torque (6V): 0.8kg.cm
- Gear Ratio: 1:48
- Body Dimensions: 70 x 22 x 18mm
- Wires Length: 200mm & 28 AWG
- Weight: 30.6g

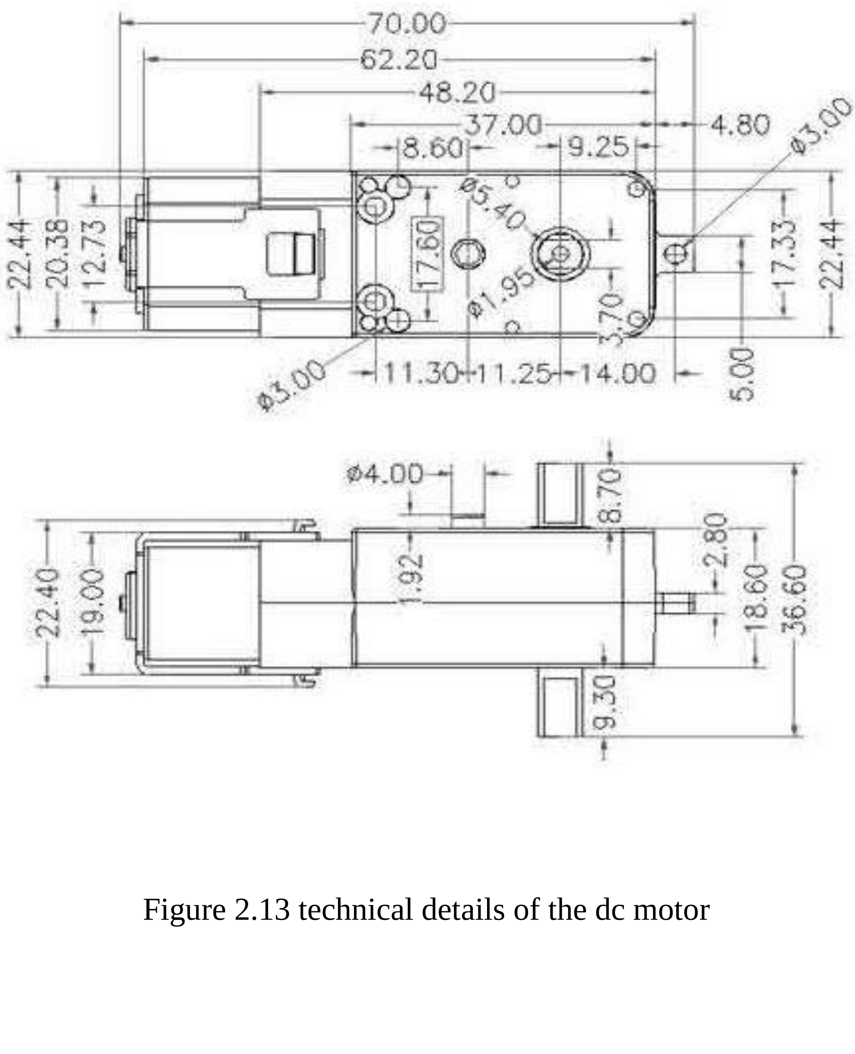


Figure 2.13 technical details of the dc motor

2.1.10 L298N Motor Driver

‘L298N Stepper Motor Driver’ is a DC motor driver board designed to drive two motors between 4.8V-46V. L298N motor driver integrated on the card is used for sumo, mini sumo, line following and variety robot motor control applications. It can supply 2A current per channel which are two separate motors. There are high-temperature and short circuit protection. The board has an internal cooler as well. Detailed representation is given in Figure 2.14[10].

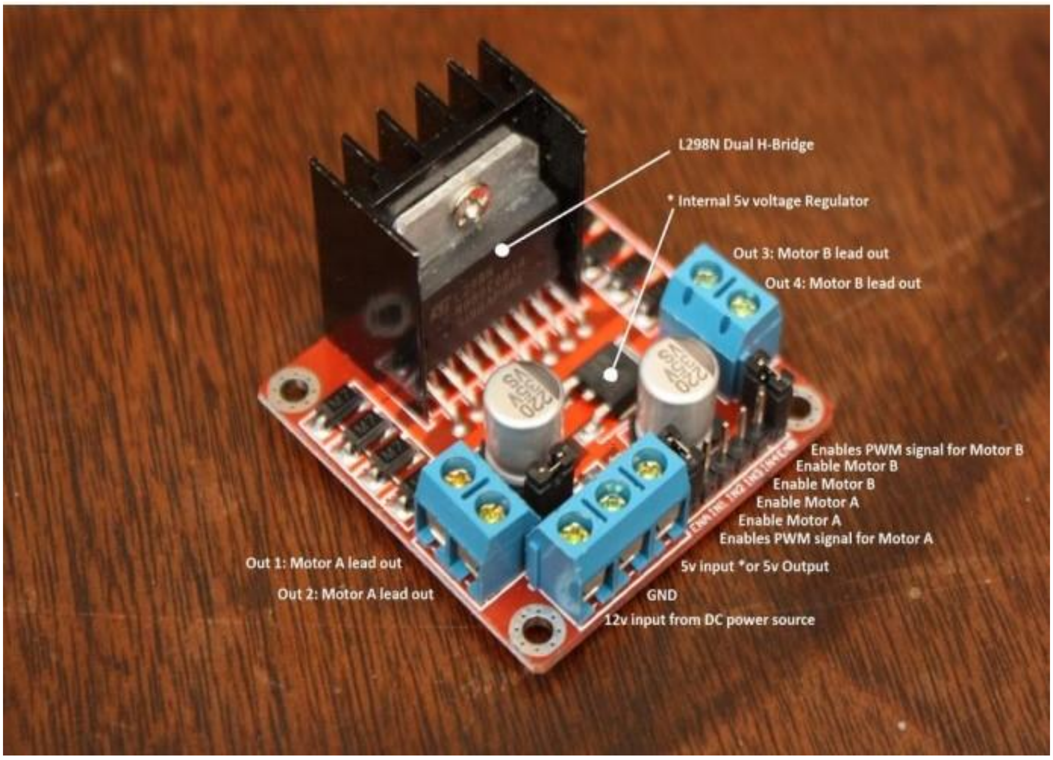


Figure 2.14 detailed representation of the L298N motor driver

2.2 Code

Before starting, low and high commands are used in certain parts of the code. This means that the devices plugged into the pins can be switched on or off by these commands. In digital pin LOW equals to zero and HIGH equals to one.

```
#include <Servo.h>   /* Servo terminal was included in the project. */
  Servo servo;

  // L289N motor drive input pins identified as follows;
  const int IN1  = 13;
  const int IN2  = 12;
  const int IN3  = 9;
  const int IN4  = 10;

  // With the enable pins, speed can be adjusted.
  int EN_A = 11;
  int EN_B = 7;

  // Buzzer indentified
  int buzzer = 3 ;

// LEDs were identified to digital pins as follows;
  int sag_on = 35;
  int sol_on = 36;
  int sag_arka = 37;
  int sol_arka = 38;
```

```
// pin identification of ultrasonic sensor.
#define echoPin 32
#define trigPin 30

// Identification of necessary variables to be used in the code.
long sure, uzaklik;
unsigned int on_uzaklik;
unsigned int sol_uzaklik = 100;
unsigned int sag_uzaklik = 100;
String voice;

void setup()
{

// The setup section defines whether it is input or output. Data reception represents input, and transmission represents output

Serial.begin(9600); //starts serial communication
servo.attach(8);  /* Servo motor connected to pin 8 */
  pinMode(IN1, OUTPUT);
  pinMode(IN2, OUTPUT);
  pinMode(IN3, OUTPUT);
  pinMode(IN4, OUTPUT);
  pinMode(EN_A, OUTPUT);
  pinMode(EN_B, OUTPUT);

  pinMode (buzzer, OUTPUT) ;

  pinMode(sag_on, OUTPUT);
  pinMode(sag_arka, OUTPUT);
```

```
  pinMode(sol_on, OUTPUT);
  pinMode(sol_arka, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(trigPin, OUTPUT);
}
```

// This part of the code is very important. It is the part that allows the robot to execute the command according to the commands given (forward, backward, right, left and stop).

```
void loop()
{
 while (Serial.available()){  //Check if there is an available byte to read
  delay(10); //Delay added to make thing stable
  char c = Serial.read(); //Conduct a serial read
   if (c == '#') {break;} //Exit the loop when the # is detected after the word
  voice += c; //Shorthand for voice = voice + c
 }

 digitalWrite (buzzer, HIGH);
 digitalWrite(trigPin, LOW);
 delayMicroseconds(5);
 digitalWrite(trigPin, HIGH);
 delayMicroseconds(10);
 digitalWrite(trigPin, LOW);

 sure = pulseIn(echoPin, HIGH); // measuring the time it takes for the wave to return
 uzaklik = sure / 29.1 / 2; // converting measured time to distance
 Serial.println("ön");//
 Serial.println(uzaklik);//
 delay(500);
```

// servo. write code checks if there is any obstacle in front of the robot. Ultrasonic sensor measures the distance to the obstacle.

```
servo.write(90);
on_uzaklik = uzaklik;
if (on_uzaklik > 20){ // Check the distance between robot and obstacle.
 if(voice == "*forward"){ // If the distance between the robot and the obstacle is more than 20 cm, the robot continues to execute the command given.

  digitalWrite(IN1, LOW);
  digitalWrite(IN2, HIGH);
  digitalWrite(IN3, LOW);
  digitalWrite(IN4, HIGH);
  analogWrite(EN_A, 100);
  analogWrite(EN_B, 100);

 }
 }

  if (uzaklik <= 20) //  If it is not more than 20 cm then stop.
  {
  digitalWrite(2,LOW);
  digitalWrite(IN1, LOW);
  digitalWrite(IN2, LOW);
  digitalWrite(IN3, LOW);
  digitalWrite(IN4, LOW);



  analogWrite(EN_A, 0);
```

```
analogWrite(EN_B, 0);

digitalWrite (buzzer, LOW);
delay(1000);
}
else if(voice == "*backward"){
digitalWrite(IN1, HIGH);
digitalWrite(IN2, LOW);
digitalWrite(IN3, HIGH);
digitalWrite(IN4, LOW);
analogWrite(EN_A, 100);
analogWrite(EN_B, 150);
```

// The LEDs flash 9 times within 3.6 seconds delay as shown below.(200*18=3600ms)

```
digitalWrite(sag_on,LOW);
digitalWrite(sag_arka,LOW);
digitalWrite(sol_on,LOW);
digitalWrite(sol_arka,LOW);
delay(200);
digitalWrite(sag_on,HIGH);
digitalWrite(sag_arka,HIGH);
digitalWrite(sol_on,HIGH);
digitalWrite(sol_arka,HIGH);
delay(200);
digitalWrite(sag_on,LOW);
digitalWrite(sag_arka,LOW);
digitalWrite(sol_on,LOW);
digitalWrite(sol_arka,LOW);
delay(200);
```

```
digitalWrite(sag_on,HIGH);
digitalWrite(sag_arka,HIGH);
digitalWrite(sol_on,HIGH);
digitalWrite(sol_arka,HIGH);
delay(200);
digitalWrite(sag_on,LOW);
digitalWrite(sag_arka,LOW);
digitalWrite(sol_on,LOW);
digitalWrite(sol_arka,LOW);
delay(200);
 digitalWrite(sag_on,HIGH);
digitalWrite(sag_arka,HIGH);
digitalWrite(sol_on,HIGH);
digitalWrite(sol_arka,HIGH);
delay(200);
digitalWrite(sag_on,LOW);
digitalWrite(sag_arka,LOW);
digitalWrite(sol_on,LOW);
digitalWrite(sol_arka,LOW);
delay(200);
digitalWrite(sag_on,HIGH);
digitalWrite(sag_arka,HIGH);
digitalWrite(sol_on,HIGH);
digitalWrite(sol_arka,HIGH);
delay(200);
digitalWrite(sag_on,LOW);
digitalWrite(sag_arka,LOW);
digitalWrite(sol_on,LOW);
digitalWrite(sol_arka,LOW);
delay(200);
```

```
digitalWrite(sag_on,HIGH);
digitalWrite(sag_arka,HIGH);
digitalWrite(sol_on,HIGH);
digitalWrite(sol_arka,HIGH);
delay(200);
digitalWrite(sag_on,LOW);
digitalWrite(sag_arka,LOW);
digitalWrite(sol_on,LOW);
digitalWrite(sol_arka,LOW);
delay(200);
digitalWrite(sag_on,HIGH);
digitalWrite(sag_arka,HIGH);
digitalWrite(sol_on,HIGH);
digitalWrite(sol_arka,HIGH);
delay(200);
digitalWrite(sag_on,LOW);
digitalWrite(sag_arka,LOW);
digitalWrite(sol_on,LOW);
digitalWrite(sol_arka,LOW);
delay(200);
digitalWrite(sag_on,HIGH);
digitalWrite(sag_arka,HIGH);
digitalWrite(sol_on,HIGH);
digitalWrite(sol_arka,HIGH);
delay(200);
digitalWrite(sag_on,LOW);
digitalWrite(sag_arka,LOW);
digitalWrite(sol_on,LOW);
digitalWrite(sol_arka,LOW);
delay(200);
```

```
digitalWrite(sag_on,HIGH);
digitalWrite(sag_arka,HIGH);
digitalWrite(sol_on,HIGH);
digitalWrite(sol_arka,HIGH);
delay(200);
digitalWrite(sag_on,LOW);
digitalWrite(sag_arka,LOW);
digitalWrite(sol_on,LOW);
digitalWrite(sol_arka,LOW);
delay(200);
digitalWrite(sag_on,HIGH);
digitalWrite(sag_arka,HIGH);
digitalWrite(sol_on,HIGH);
digitalWrite(sol_arka,HIGH);
delay(200);
analogWrite(EN_A, 0);
analogWrite(EN_B, 0);


}
else if(voice == "*right") {
servo.write(0); // The servo motor rotates the ultrasonic sensor right at an angle of 90 degrees.
delay(1000);


digitalWrite(trigPin, LOW); // Ultrasonic sensor disabled.
delayMicroseconds(5);
digitalWrite(trigPin, HIGH); // The ultrasonic sensor was activated to transmit the wave.
delayMicroseconds(10);
digitalWrite(trigPin, LOW);
```

sure = pulseIn(echoPin, HIGH); // Measuring the time it takes for the wave to return

uzaklik = sure / 29.1 / 2; // The measured time was converted to distance.

delay (1000);

sag_uzaklik = uzaklik;

Serial.println("sag");

Serial.println(sag_uzaklik);

servo.write(90); // Sensor returns to the main position of it.

if (sag_uzaklik > 30){

// If the distance is more than 30 cm, the robot executes the right turn command.

digitalWrite(IN1, HIGH);

digitalWrite(IN2, LOW);

digitalWrite(IN3, LOW);

digitalWrite(IN4, HIGH);

analogWrite(EN_A, 100);

analogWrite(EN_B, 100);

// The robot performs the turn right command within 1.8 seconds delay (200*9=1800ms) and simultaneously turns the right front and right rear LEDs on and off 5 times.

digitalWrite(sag_on,LOW);

digitalWrite(sag_arka,LOW);

delay(200);

digitalWrite(sag_on,HIGH);

digitalWrite(sag_arka,HIGH);

delay(200);

digitalWrite(sag_on,LOW);

digitalWrite(sag_arka,LOW);

```
    delay(200);
    digitalWrite(sag_on,HIGH);
    digitalWrite(sag_arka,HIGH);
    delay(200);
    digitalWrite(sag_on,LOW);
    digitalWrite(sag_arka,LOW);
    delay(200);
    digitalWrite(sag_on,HIGH);
    digitalWrite(sag_arka,HIGH);
    delay(200);
    digitalWrite(sag_on,LOW);
    digitalWrite(sag_arka,LOW);
    delay(200);
    digitalWrite(sag_on,HIGH);
    digitalWrite(sag_arka,HIGH);
    delay(200);
    digitalWrite(sag_on,LOW);
    digitalWrite(sag_arka,LOW);
    delay(200);
    digitalWrite(sag_on,LOW);
    digitalWrite(sag_arka,LOW);
    }

if (sag_uzaklik <= 30)  {
        digitalWrite (buzzer, LOW);
       delay(1000);
        digitalWrite(IN1, LOW);
        digitalWrite(IN2, LOW);
        digitalWrite(IN3, LOW);
        digitalWrite(IN4, LOW);
```

```
// If the distance is equal or less than 30 cm, the robot stops and the buzzer is activated.

    analogWrite(EN_A, 0);
    analogWrite(EN_B, 0);
  }
}
else if(voice == "*left") {

  servo.write(180); // The servo motor rotates the ultrasonic sensor left at an angle of 90 degrees.
    delay(1000);
    digitalWrite(trigPin, LOW); //
    delayMicroseconds(5);
    digitalWrite(trigPin, HIGH); // The ultrasonic sensor was activated to transmit the wave.
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW); // Ultrasonic sensor disabled.
    sure = pulseIn(echoPin, HIGH); // Measuring the time it takes for the wave to return.
    uzaklik = sure / 29.1 / 2; // The measured time was converted to distance.
    sol_uzaklik = uzaklik;
    Serial.println("sol");
    Serial.println(sol_uzaklik);
    servo.write(90); // Sensor returns to the main position of it.




  if (sol_uzaklik > 30){
```

// If the distance is more than 30 cm, the robot executes the left turn command.

```
digitalWrite(IN1, LOW);
digitalWrite(IN2, HIGH);
digitalWrite(IN3, HIGH);
digitalWrite(IN4, LOW);
analogWrite(EN_A, 150);
analogWrite(EN_B, 100);
```

// The robot performs the turn left command within 1.8 seconds delay (200*9=1800ms) and simultaneously turns the left front and left back LEDs on and off 5 times.

```
digitalWrite(sol_on,LOW);
digitalWrite(sol_arka,LOW);
delay(200);
digitalWrite(sol_on,HIGH);
digitalWrite(sol_arka,HIGH);
delay(200);
digitalWrite(sol_on,LOW);
digitalWrite(sol_arka,LOW);
delay(200);
digitalWrite(sol_on,HIGH);
digitalWrite(sol_arka,HIGH);
delay(200);
digitalWrite(sol_on,LOW);
digitalWrite(sol_arka,LOW);
delay(200);
digitalWrite(sol_on,HIGH);
digitalWrite(sol_arka,HIGH);
```

```
delay(200);
digitalWrite(sol_on,LOW);
digitalWrite(sol_arka,LOW);
delay(200);
digitalWrite(sol_on,HIGH);
digitalWrite(sol_arka,HIGH);
delay(200);
digitalWrite(sol_on,LOW);
digitalWrite(sol_arka,LOW);
delay(200);
digitalWrite(sol_on,HIGH);
digitalWrite(sol_arka,HIGH);

analogWrite(EN_A, 0);  // The robot stops after the given command is executed.
analogWrite(EN_B, 0);
}
if (sol_uzaklik <= 30){      // If the distance is equal or less than 30 cm, the robot stops and the buzzer is activated.
  digitalWrite (buzzer, LOW);
  delay(1000);
  digitalWrite(IN1, LOW);
  digitalWrite(IN2, LOW);
  digitalWrite(IN3, LOW);
  digitalWrite(IN4, LOW);
  analogWrite(EN_A, 0);
  analogWrite(EN_B, 0);
  }
}

else if(voice == "*stop") {
```

```
    digitalWrite(IN1, LOW);
    digitalWrite(IN2, LOW);
    digitalWrite(IN3, LOW);
    digitalWrite(IN4, LOW);

    analogWrite(EN_A, 0);
    analogWrite(EN_B, 0);
   }

  voice=""; //Reset the variable after initiating

}
```

2.3 Wiring

HC-05 Bluetooth Module	**Arduino Mega 2560 Card**
5V	5V
GND	GND
Tx	Digital PIN1
Rx	Digital PIN0

SG90 9G Mini Servo Motor	**Arduino Mega 2560 Card**
Vcc	5V
GND	GND
Input	Digital PIN8

Buzzer	**Arduino Mega 2560 Card**
Vcc	5V
GND	GND
Input	Digital PIN3

HC-SR04 Ultrasonic Distance Sensor	**Arduino Mega 2560 Card**
Vcc	5V
GND	GND
Echo	Digital PIN32
Trigger	Digital PIN30

Led1	**Arduino Mega 2560 Card**
Vcc	Digital PIN35
GND	GND

Led2	**Arduino Mega 2560 Card**
Vcc	Digital PIN36
GND	GND

Led3	**Arduino Mega 2560 Card**
Vcc	Digital PIN37
GND	GND

Led4	**Arduino Mega 2560 Card**
Vcc	Digital PIN38
GND	GND

L298N Motor Driver	**Arduino Mega 2560 Card**
ENA	Digital PIN11
ENB	Digital PIN7
IN1	Digital PIN13
IN2	Digital PIN12
IN3	Digital PIN9
IN4	Digital PIN10

L298N Motor Driver	**DC Motor**
INA1	Motor1
INB1	Motor1
INA2	Motor2
INB2	Motor2

L298N Motor Driver	**Battery**	**Arduino Mega 2560 Card**
12V Input Port	12V(+)	-
GND	GND(-)	GND
5V	-	Vin

The components connections shown in the tables above are schematized below.

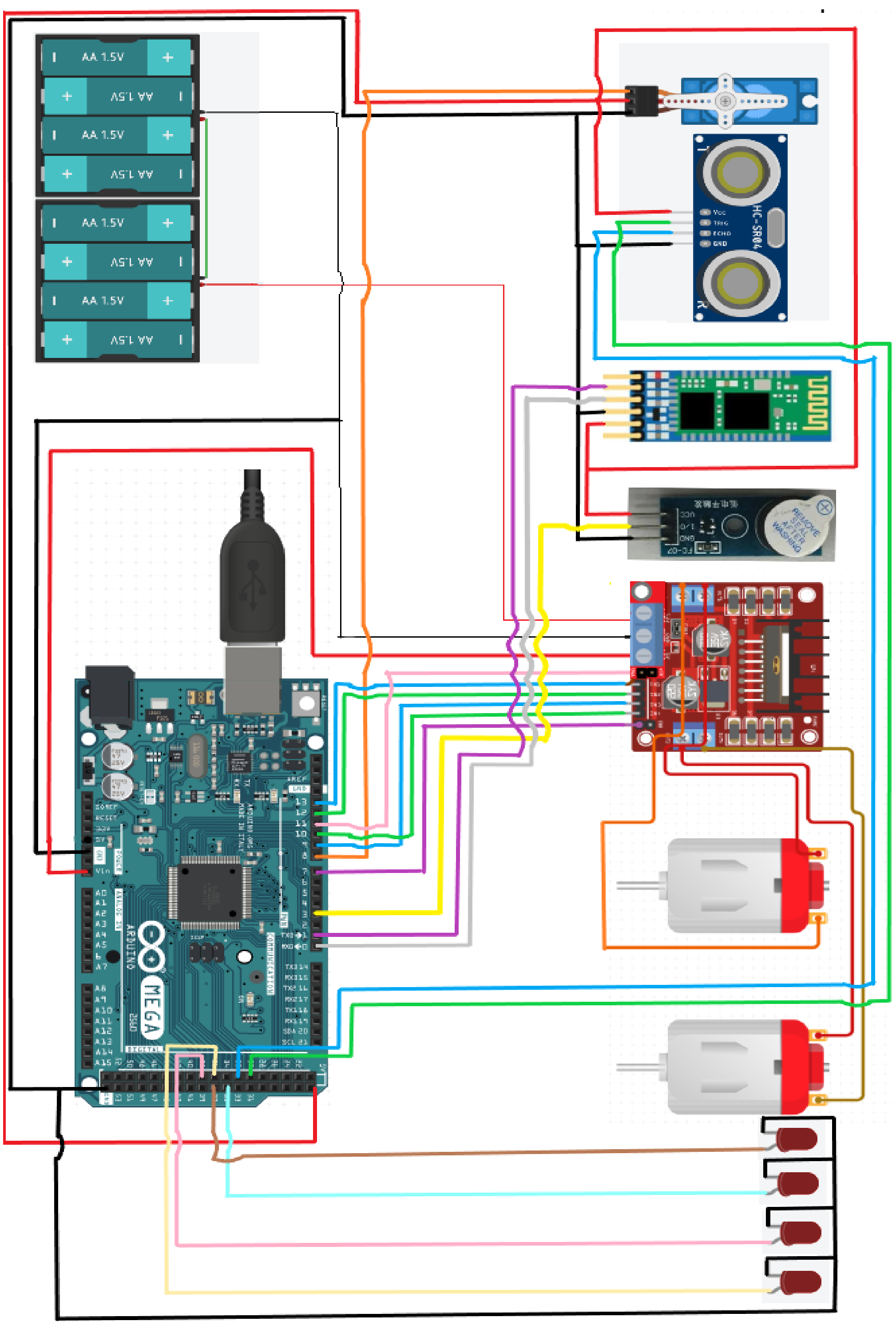


Figure 2.15 detailed illustration of wiring

3. CONCLUSION

In conclusion, as mentioned earlier the purpose of this project is to create a robot which is driven by voice commands. The project basically consists of two steps: coding and wiring.By the identification of the pins in the code, the commands sent are detected and executed by the robot.

In wiring, the input and output pins must be correctly connected. With enable pins, the speed of the robot's motors can be changed. The required values must be given to these pins in order to rotate the robot at the desired angle according to the command given. The sewing method can be used to avoid cable tangling. The required total voltage value for the parts to operate is provided by the voltage supply.

With the help of the application, voice commands are converted into text and transmitted to the bluetooth module. These text commands must match the commands in the arduino software language. In this project there are 5 voice commands. These are forward, backward, right, left and stop commands. The servo motor rotates the ultrasonic sensor according to the direction of the command, it detects and measures the distance to the obstacle. If the measured distance is more than the specified distance with the code, the robot executes the command. On the other hand, if the distance is less than the specified distance, the buzzer runs and the robot does not move.

4. REFERENCES

[1]Servo Motor SG90 Data Sheet, “ee.ic.ac.uk”. [Online]. Avaliable: http://www.ee.ic.ac.uk/pcheung/teaching/DE1_EE/stores/sg90_datasheet.pdf

[2]Piezo Buzzer Data Sheet, “farnell.com” 29 July 2016. [Online]. Avaliable: http://www.farnell.com/datasheets/2095438.pdf

[3]Arduino Mega 2560 Data Sheet, “mantech.co.za”. [Online]. Avaliable: http://www.mantech.co.za/datasheets/products/A000047.pdf

[4]5mm Diode Led Data Sheet, “tr.led-kento.net”. [Online]. Avaliable: http://tr.led-kento.net/lamp-led/5mm-round-lamp-led/led-5mm-red-diffused-red-led-round-top-led.html

[5]Jumper Wire Data Sheet, “flying1688.com”. [Online]. Avaliable: https://www.flying1688.com/download_file.php?id=272&f=1

[6] A. Kopuz, “maker.robotistan.com” 28 October 2018. [Online]. Avaliable: https://maker.robotistan.com/breadboard/

[7] Cytron Technologies Sdn. Bhd., “datasheet4u.com” May 2018. [Online]. Avaliable: https://datasheet4u.com/datasheet-parts/HC-SR04-datasheet.php?id=1291829

[8] HC-05 - Bluetooth Module Data Sheet, “components101.com” 10 March 2018. [Online]. Avaliable: https://components101.com/wireless/hc-05-bluetooth-module

[9]DC Gearbox Motor Data Sheet, “adafruit.com”. [Online]. Avaliable: https://www.adafruit.com/product/3777

[10]L298N Motor Driver Data Sheet, “robotpark.com.tr”. [Online]. Avaliable: http://www.robotpark.com.tr/L298N-H-Bridge-Cift-Motor-Surucu-Karti