



# RC Car

—

**SRB DEMO FILES**

**ROBOVR - THE OLYMPICS OF ROBOTS**

**[WWW.ROBOVR.WORLD](http://WWW.ROBOVR.WORLD)**

## Introduction

A four-wheel drive F1 car powered by minimal number of electrical components, featuring instantaneous response to driver's input. The car is compact and light, enabling it to traverse through any terrain. The driver can modify various features according to his will.

## Requirements

A list of requirements and goals were developed to guide you in the development of this project. The requirements are as follows.

- The car shall be controlled by a Bluetooth controller.
- The car shall be battery powered.

## Materials

Arduino Uno R3
HC-05 Bluetooth Module
12V Lithium-ion Battery
L298N Motor Driver Module
DC Motors (300 RPM)
Car Chassis
Servo Motors

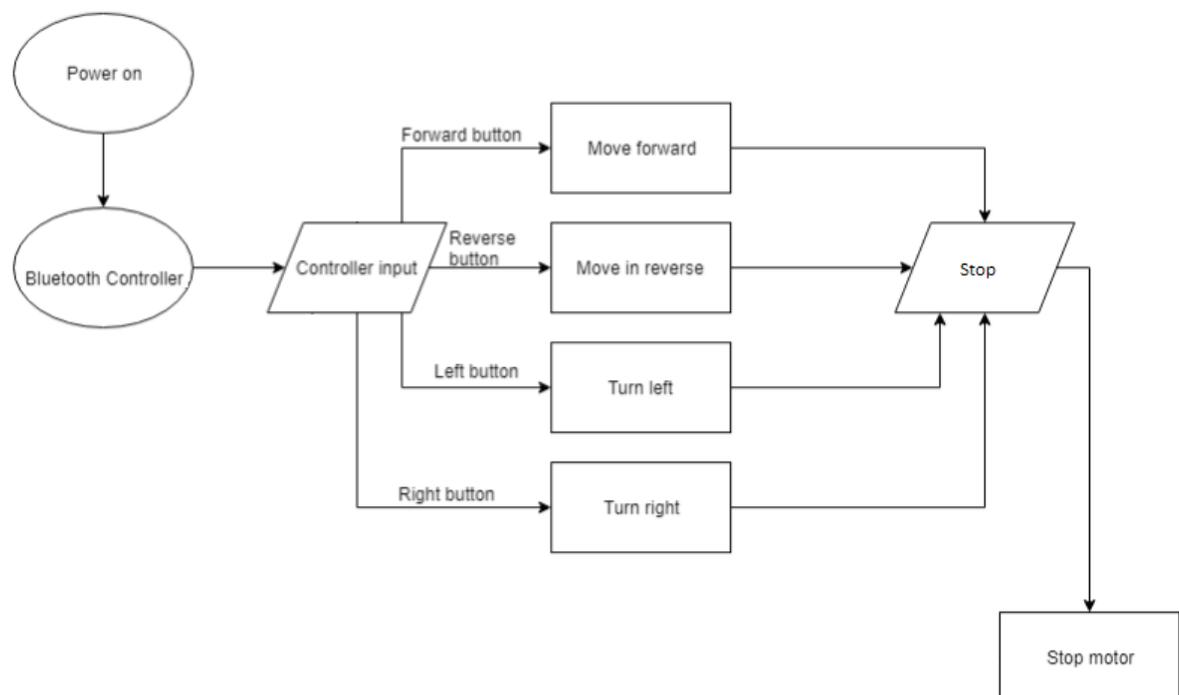
In addition to the hardware used on the car itself, there was some other hardware necessary to build and run the car. These additional tools included a soldering iron and some solder to use to disassemble the RC car and solder wires to the pins connected to the motor batteries. In addition an Android phone is required along with the app "**Arduino Bluetooth Control**" which can be found in the Google Play Store.

The app has prebuilt symbols, the user just needs to assign various characters (A-Z or a-z) to these symbols and press enter after assigning characters to each symbol.

Once all the required characters are assigned the user should select "switch to controller mode" for connecting the Bluetooth with the Arduino microcontroller.

If asked for a password, the password may be 0000 or 1234 and the name of the Bluetooth module will be HC-05

## Working

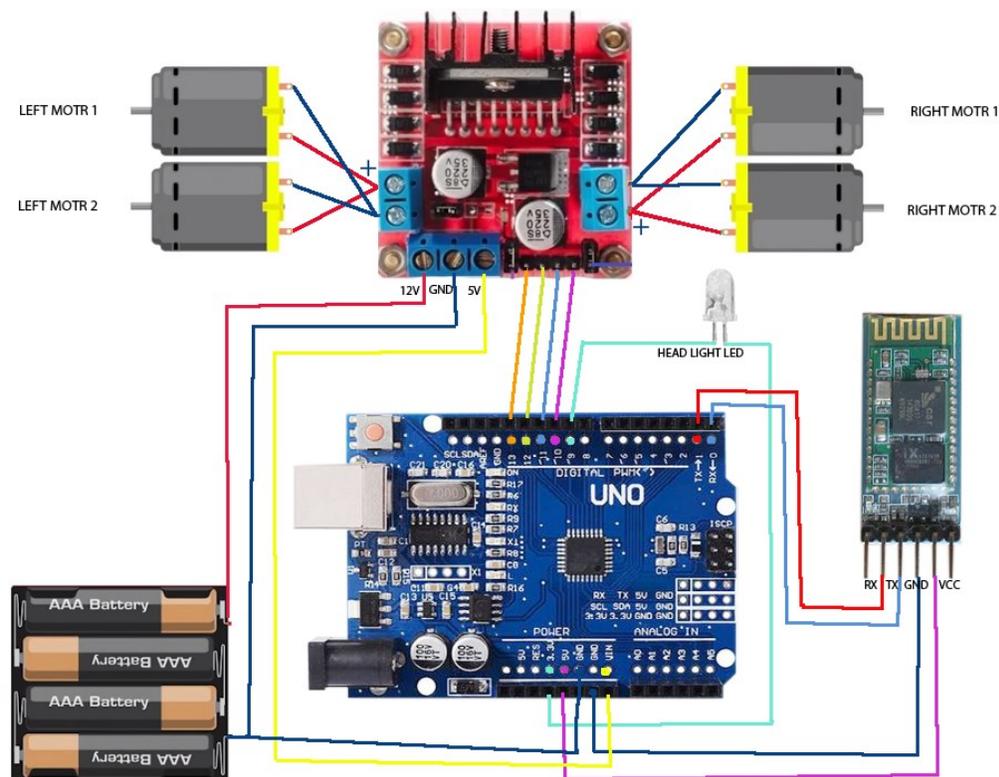


The working of this F1 car is based on Arduino UNO microcontroller. The controller is used for communication between various electrical components. Motor driver is connected to the controller and is used to provide sufficient power to the motors connected to it and acts as a medium for communication between Arduino and the motors. The Bluetooth module helps in communicating the desired motion via a mobile application. The predefined code helps to achieve forward, backward, right turn and left turn motion.

**NOTE:**

When uploading the program to the Arduino the connection to the GPIO pin D0 and D1 had to be temporarily removed until uploading. This became somewhat of a hassle after many times having to upload new code. This must be done since the D0 and D1 pin is used for serial communication between the Arduino and computer.

## Circuits and Connections

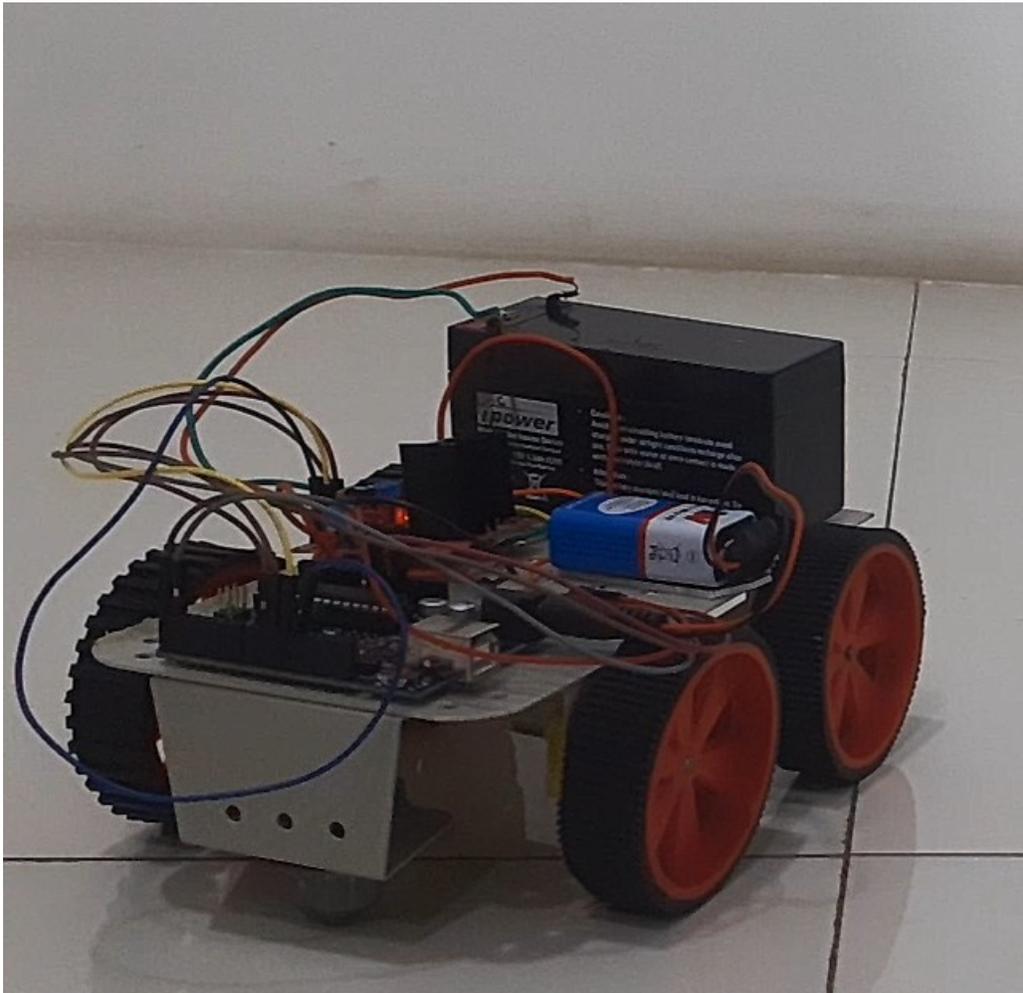


The Vcc and Ground pins of the servo motors should be connected to the Vcc and ground pins of the Arduino.

The four motors that are used to move the bot are connected to the L298N motor driver module as shown in the figure.

The servo motor is connected to pin 4 of the Arduino and moves in the clockwise direction when user presses the button on the Bluetooth app and thus the arrow is released due to this motion of the servo motor.

## RoboVR Design:



## Code

```
char ch='S'; //initially STOP condition
const int IN1=9;// Define RIGHT MOTOR
const int IN2=10;
const int IN3=11;// Define LEFT MOTOR
const int IN4=12;
```

```
void setup()
{
  pinMode(IN1,OUTPUT);
  pinMode(IN2,OUTPUT);
  pinMode(IN3,OUTPUT);
  pinMode(IN4,OUTPUT);
  Serial.begin(9600);
}
void loop()
{
  if(Serial.available()) // check if data is available or not
  {
    ch=Serial.read(); // read the incoming byte
  }
  Serial.println(ch);
  if (ch=='R')
  {
    RIGHT();
  }
  if (ch=='L')
  {
    LEFT();
  }
  if (ch=='F')
  {
    FORWARD();
  }
  if (ch=='B')
  {
    BACKWARD();
  }
  if (ch=='S')
  {
    STOP();
  }
  void FORWARD(){
    Serial.println("forward");
    digitalWrite(IN1,HIGH);
    digitalWrite(IN2,LOW);
    digitalWrite(IN3,LOW);
```

```
digitalWrite(IN4,HIGH);
}
//-----
void BACKWARD(){
Serial.println("backward");
digitalWrite(IN1,LOW);
digitalWrite(IN2,HIGH);
digitalWrite(IN3,HIGH);
digitalWrite(IN4,LOW);
}
//-----
void LEFT(){
Serial.println("LEFT");
digitalWrite(IN1,LOW);//MOVING RIGHT MOTOR IN CLOCKWISE DIRECTION
digitalWrite(IN2,HIGH);
digitalWrite(IN3,LOW);//MOVING LEFT MOTOR IN ANTICLOCKWISE DIRECTION
digitalWrite(IN4,HIGH);
}
//-----
void RIGHT(){
Serial.println("RIGHT");
digitalWrite(IN1,HIGH);//MOVING RIGHT MOTOR IN ANTICLOCKWISE
DIRECTION
digitalWrite(IN2,LOW);
digitalWrite(IN3,HIGH);//MOVING LEFT MOTOR IN CLOCKWISE DIRECTION
digitalWrite(IN4,LOW);
}
//-----
void STOP(){
Serial.println("stop");
digitalWrite(IN1,LOW);
digitalWrite(IN2,LOW);
digitalWrite(IN3,LOW);
digitalWrite(IN4,LOW);
delay(100);
}
```