

PROJECTS | TUTORIALS | COURSES | KITS

CONTACT US

YOU ARE HERE: [HOME](#) / [ARDUINO](#) / BLUETOOTH CONTROLLED ROBOT USING ARDUINO

Search this website

AUGUST 7, 2018 BY [RAVI](#) — 6 COMMENTS



ONLY \$5
FOR 10 PCBS



PCB PROTOTYPE **Order Now**

 **Lead time 2-3 days**
Free first order for new members

1. Introduction
2. Prerequisites for Bluetooth Controller Robot
 - 2.1. HC-05 Bluetooth Module
 - 2.2. L298N Motor Driver Module
3. Circuit Diagram of Bluetooth Controlled Robot
 - 3.1. Components Required
 - 3.2. Circuit Design
4. Code
5. Android App
6. Working
7. Limitations
8. Applications

Robots are always a fancy topic for students, hobbyists and DIYers. If you are beginner, then building a robot (like a car or an arm) is probably one of the important projects to do after

PROJECTS BY CATEGORY

- Arduino Projects (200+)
- Electronics Projects (250+)
- Mini Project Circuits (160+)
- Mini Project Ideas (150+)
- ECE Projects (150+)
- EEE Projects (150+)
- 8051 Projects (110+)
- Raspberry Pi Projects (101+)
- Electrical Project Ideas (100+)
- Embedded Projects (100+)
- Latest Electronics Ideas (100+)
- Microcontroller Mini Projects (100+)
- Robotics Projects (100+)
- VLSI Projects (100+)
- Solar Projects (100+)
- IOT Projects (100+)

- Communication Projects (70+)
- LED Projects (70+)
- Power Electronics Projects (60+)
- RFID Projects (60+)
- Home Automation Projects (50+)

learning about the basics.

If you remember the earlier tutorial, I have discussed about HC-05 Bluetooth Module and how to interface one with Arduino. Also, I have provided a simple Bluetooth Controller App, which can be installed on your Android Phone and start transmitting the data.

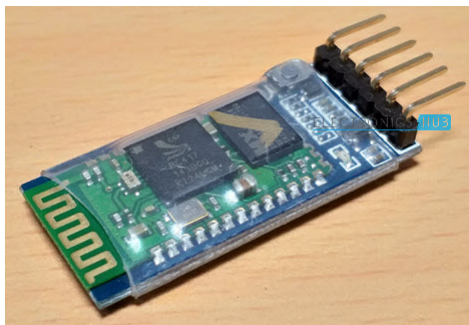
As a continuation to that project, I will be implementing Bluetooth Controlled Robot using Arduino and a few other components and build a simple robotic car that can be controlled using an Android Phone (through an App) over Bluetooth Communication.

Prerequisites for Bluetooth Controller Robot

Apart from Arduino, which is the main controlling module of the project, there are two other important modules that you have to be familiar with in order to implement the Bluetooth Controlled Robot project.

They are the HC-05 Bluetooth Module and the L298N Motor Driver Module.

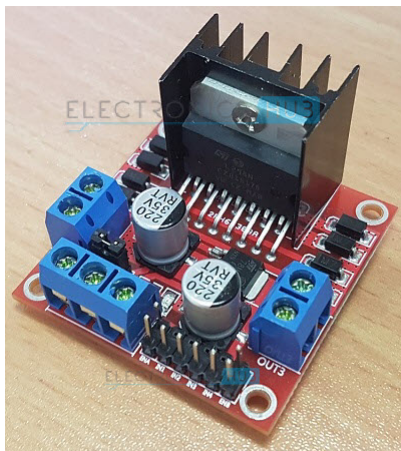
HC-05 Bluetooth Module



The HC-05 Bluetooth Module is responsible for enabling Bluetooth Communication between Arduino and Android Phone.

For more information on HC-05 Bluetooth Module, refer to [HC-05 BLUETOOTH MODULE](#).

L298N Motor Driver Module



The L298N Motor Driver Module is responsible for providing the necessary drive current to the motors of the robotic car. I have provided information about L298N Module in an earlier project called Arduino DC Motor Control using L298N.

So, refer to [ARDUINO DC MOTOR CONTROL USING L298N](#) for more information on interfacing L298N with Arduino.

NOTE: I strongly recommend you to refer to the above mentioned two projects before proceeding further.

Circuit Diagram of Bluetooth Controlled Robot

The following is the circuit diagram of Bluetooth Controlled Robot using Arduino, L298N and HC-05.

[Matlab Projects \(50+\)](#)
[EIE Projects \(50+\)](#)
[Wireless Projects \(50+\)](#)
[LabView Projects \(45+\)](#)
[Zigbee Projects \(45+\)](#)
[GSM Projects \(40+\)](#)
[555 Timer Circuits \(40+\)](#)
[Sensor Projects \(40+\)](#)
[ARM Projects \(60+\)](#)
[DTMF Projects \(30+\)](#)
[PIC Projects \(30+\)](#)
[Electrical Mini Projects \(25\)](#)
[ESP8266 Projects \(15\)](#)

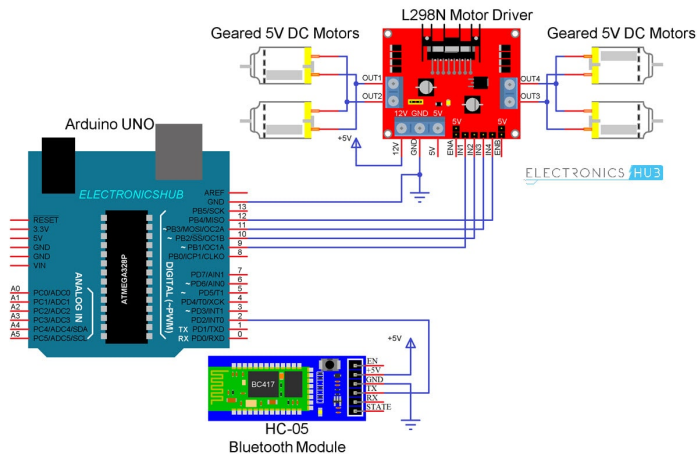
KITS

[Best Drone Kits \[12\]](#)
[3D Printer Kits \[12\]](#)
[Best Robot Vacuum Clears \[14\]](#)
[Best Waveform Generators \[12\]](#)
[RGB LED Strip Light Kits \[20\]](#)
[Best LED Christmas Light Kits \[13\]](#)

SUBSCRIBE FOR UPDATES

Enter your email address:

SUBSCRIBE



Components Required

- Arduino UNO [\[Buy Here\]](#)
- L298N Motor Driver Module [\[Buy Here\]](#)
- HC-05 Bluetooth Module [\[Buy Here\]](#)
- Robot Chassis [\[Buy Here\]](#)
- 4 x 5V Geared Motors [\[Buy Here\]](#)
- Connecting Wires [\[Buy Here\]](#)
- Battery Holder [\[Buy Here\]](#)
- Power Supply
- Android Phone
- Bluetooth Controller App

NOTE: I have used L298N Motor Driver Module to drive the motors of the robot. You can use either this one or L293D Motor Driver Module. If you are using L293D, then check out for the connections.

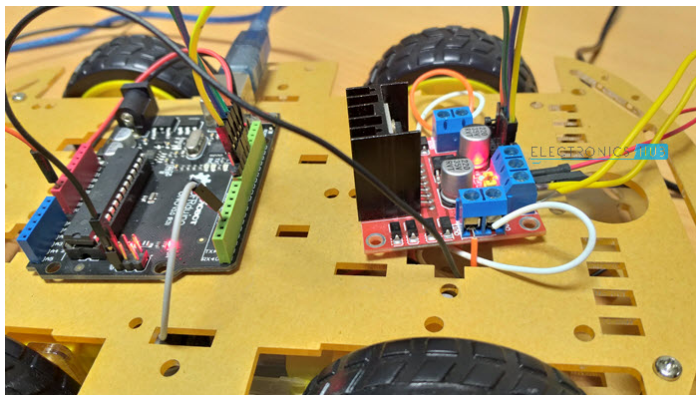
Circuit Design

I wouldn't go into the details of the construction of the robot as your robot chassis might be different from mine and you can easily figure it out how to build the robot from the available parts and possible cable management for making the robot more appealing.

Coming to the design of the circuit, first is the HC-05 Bluetooth Module. The +5V and GND pins of the Bluetooth Module are connected to +5V and GND of Arduino.

Since I will be only transmitting data related to the Robot's movement from Android Phone to Bluetooth Module and do not intend to receive any data from Arduino, I will connect only the TX pin of the Bluetooth Module to RX Pin of Arduino.

This RX pin of Arduino is based on SoftwareSerial library (Pin 2 and Pin 3 are configured as RX and TX on Arduino). The RX pin of the Bluetooth is left open.



Now, the L298N Motor Driver Module. Digital I/O Pins 9 through 12 of Arduino are configured as Input pins of the Motor Driver and are connected to IN1 through IN4 of the L298N Motor Driver Module. Both the Enable Pins are connected to 5V through provided jumper.

The robot chassis which I am using in this Bluetooth Controlled Robot Car project is supplied with 4 geared motors. Since L298N has slots for only two motors, I have joined the left side motors as one set and the right side motors as other set and connected both these sets to the output of L298N Module.

Code

The Arduino code for Bluetooth Controlled Robot project is given below.

```

1  #include<SoftwareSerial.h>
2
3  #define IN1 12
4  #define IN2 11
5  #define IN3 10
6  #define IN4 9
7  //define EN1 6
8  //define EN2 5
9
10 SoftwareSerial mySerial(2, 3); // RX, TX
11
12 String data;
13 int btVal;
14
15 void setup()
16 {
17     //Serial.begin(115200);
18     pinMode(IN1, OUTPUT);
19     pinMode(IN2, OUTPUT);
20     pinMode(IN3, OUTPUT);
21     pinMode(IN4, OUTPUT);
22     //pinMode(EN1, OUTPUT);
23     //pinMode(EN2, OUTPUT);
24     digitalWrite(IN1, LOW);
25     digitalWrite(IN2, LOW);
26     digitalWrite(IN3, LOW);
27     digitalWrite(IN4, LOW);
28     //analogWrite(EN1,63);
29     //analogWrite(EN2,63);
30     mySerial.begin(9600);
31 }
32
33 void loop()
34 {
35     while (mySerial.available())
36     {
37         {
38             data = mySerial.readStringUntil('\n');
39             //Serial.print(str);
40         }
41
42         btVal = (data.toInt());
43         //Serial.print("BlueTooth Value ");
44         //Serial.println(btVal);
45
46
47
48         switch (btVal)
49         {
50             case 1:
51                 //Serial.println("Forward");
52                 forward();
53                 break;
54
55             case 2:
56                 //Serial.println("Reverse");
57                 reverse();
58                 break;
59
60             case 3:
61                 //Serial.println("Left");
62                 left();
63                 break;
64
65             case 4:
66                 //Serial.println("Right");
67                 right();
68                 break;
69
70             case 5:
71                 //Serial.println("Stop");
72                 stoprobot();
73                 break;
74
75         }
76
77     }
78
79
80     if (mySerial.available() < 0)
81     {

```

```

82      //Serial.println("No Bluetooth Data ");
83    }
84  }
85 }
86
87 void forward()
88 {
89   digitalWrite(IN1, HIGH);
90   digitalWrite(IN2, LOW);
91   digitalWrite(IN3, HIGH);
92   digitalWrite(IN4, LOW);
93 }
94
95 void reverse()
96 {
97   digitalWrite(IN1, LOW);
98   digitalWrite(IN2, HIGH);
99   digitalWrite(IN3, LOW);
100  digitalWrite(IN4, HIGH);
101 }
102
103 void left()
104 {
105   digitalWrite(IN1, LOW);
106   digitalWrite(IN2, LOW);
107   digitalWrite(IN3, HIGH);
108   digitalWrite(IN4, LOW);
109 }
110
111 void right()
112 {
113   digitalWrite(IN1, HIGH);
114   digitalWrite(IN2, LOW);
115   digitalWrite(IN3, LOW);
116   digitalWrite(IN4, LOW);
117 }
118
119 void stoprobot()
120 {
121   digitalWrite(IN1, LOW);
122   digitalWrite(IN2, LOW);
123   digitalWrite(IN3, LOW);
124   digitalWrite(IN4, LOW);
125 }

```

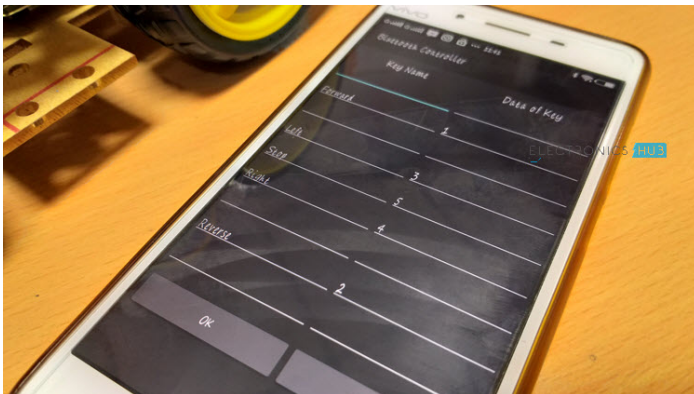
Bluetooth_Controlled_Robot_using_Arduino.ino hosted with ❤ by GitHub

[view raw](#)

Android App

If you remember the HC-05 Bluetooth Module tutorial, I have used a simple app called Bluetooth Controller, which is installed on an Android Phone to communicate with the Bluetooth Module.

In this project, I have used the same app with modifications in the data to be transmitted.



The above given Arduino code is written to synchronize with the data configured in the Bluetooth Controller App.

NOTE: The link to download the Bluetooth Controller App is provided in the HC-05 Bluetooth Module tutorial. To download the app, please refer to that tutorial.

Working

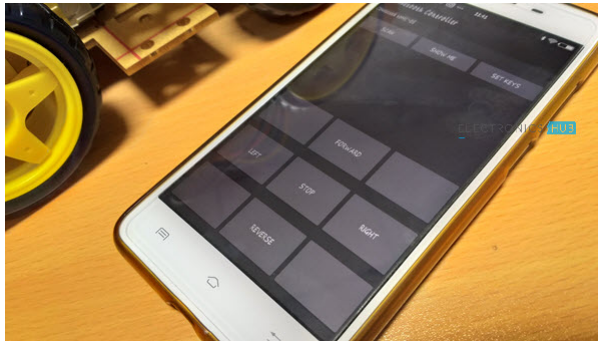
Assemble the robot, make the necessary connections and upload the code to Arduino. If you understood the HC-05 Bluetooth Module tutorial, then understanding the Bluetooth

Controlled Robot project is very easy.

First, in the Android App, I have used 5 keys as Forward, Reverse, Left, Right and Stop. The corresponding data associated with each key is as follows:

- Forward – 1
- Reverse – 2
- Left – 3
- Right – 4
- Stop – 5

When a key is pressed, the corresponding data is transmitted to the Bluetooth Module from the Phone over Bluetooth Communication.



In the Arduino code, the Arduino UNO receives any of this data from the Bluetooth Module (as per the key pressed) and performs a simple switch case operation, where each case associated with appropriate instructions to the Motor Driver Input Pins.

For example, if 'Forward' key is pressed in the Android Phone, then '1' is transmitted. Arduino will then make IN1 and IN3 as HIGH and IN2 and IN4 as LOW to achieve a forward motion.

Similarly, other keys correspond to appropriate setting of IN1 – IN4 pins.

Limitations

- As the range of the Bluetooth Communication is limited (a maximum of 10 meters for class 2 devices for example) the control range of Bluetooth Controlled Robot is also limited.
- Make sure that sufficient power is provided to all the modules especially the Bluetooth Module. If the power is not sufficient, even though the Bluetooth Module powers on, it cannot transmit data or cannot be paired with other Bluetooth devices.

Applications

- Low range Mobile Surveillance Devices
- Military Applications (no human intervention)
- Assistive devices (like wheelchairs)
- Home automation

FILED UNDER: [ARDUINO](#), [DIY PROJECTS](#)

Comments



pearl.says

AUGUST 9, 2018 AT 4:04 AM

Wow this is a great idea...

[Reply](#)**Harsh Vardhan says**

AUGUST 17, 2018 AT 3:48 AM

Hi..could you tell me the total cost of this project? And can a first year engineering student make this?

[Reply](#)**Ravi says**

AUGUST 18, 2018 AT 5:18 AM

It will cost you somewhere around Rs. 2000/-

[Reply](#)**Harsha says**

AUGUST 30, 2018 AT 4:26 AM

Diploma students make this project?..

[Reply](#)**Ravi says**

AUGUST 31, 2018 AT 3:00 AM

Yes,

[Reply](#)**vinay gautam says**

SEPTEMBER 10, 2018 AT 8:27 AM

how can i get robo kit and from where?

[Reply](#)

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment

Name *

Email *

Website

POST COMMENT

GENERAL

- Tutorials
- Symbols
- Courses
- Calculator
- Contact

PROJECTS

- Electrical
- Electronics
- Embedded
- Power
- Robotics
- ARM
- IOT

PROJECTS

- Mini projects
- Microcontroller
- Arduino
- Solar
- Free circuits
- Home Automation
- Seminar Topics
- Electronics Questions

TUTORIALS

- Capacitors
- Resistors
- Filters
- Diodes
- Transistors

TUTORIALS

- Amplifiers
- IO Devices
- Thyristors
- DC Circuits
- Number System
- TS EAMCET 2019

FOLLOW US

- Instagram
- Youtube
- Facebook
- Google Plus
- Twitter