

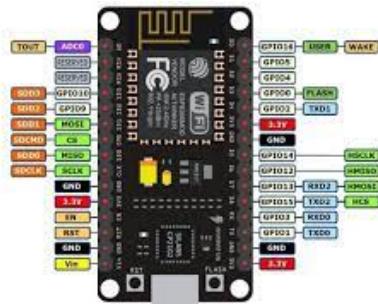
Roborace Bot

Components required

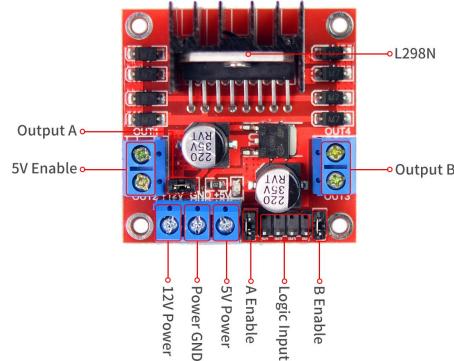
Components Name	Quantity
ESP8266 (NodeMCU)	1
Motor Driver (L298N)	1
Motors	4
Wheels	4
12 V Li-po battery/ 12V Li-Ion battery	1
XT 60 / T connector / DC jack	2
Chassis	1
Motor Holder	4
Jumper Wires (MM/MF/FF)	10 each
Wires	0.5 meter

Components

1. ESP8266



2. Motor Driver



3. Moto, Wheel and Motor Holder



4. Battery



Li-Ion Battery

or



Lipo Battery

5. Jumper wire, Connecting wire, DC jack or XT-connector



Jumper Wires



Wires

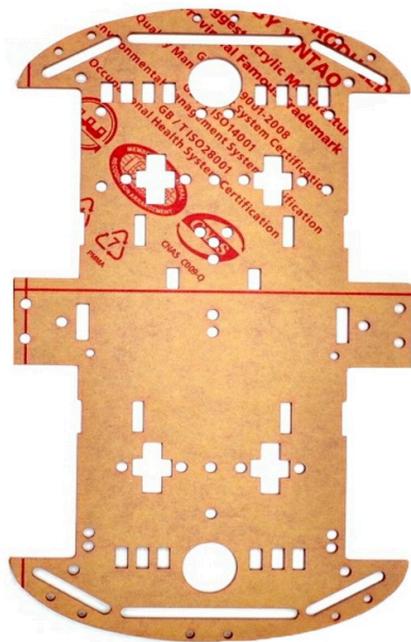


DC jack



XT connector

6. Chassis



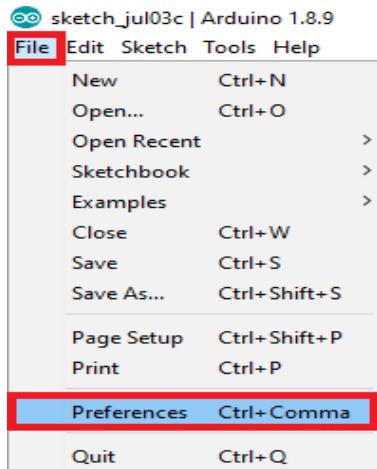
Procedure

1. Install the Arduino IDE on your Laptop/ Computer
2. Also Install the CH340 Driver
3. Connect ESP8266 to your Laptop/ Computer

If You have ESP8266 then follow the following method. Install ESP8266 Add-on in Arduino IDE

To install the ESP8266 board in your Arduino IDE, follow these next instructions: In your Arduino IDE,

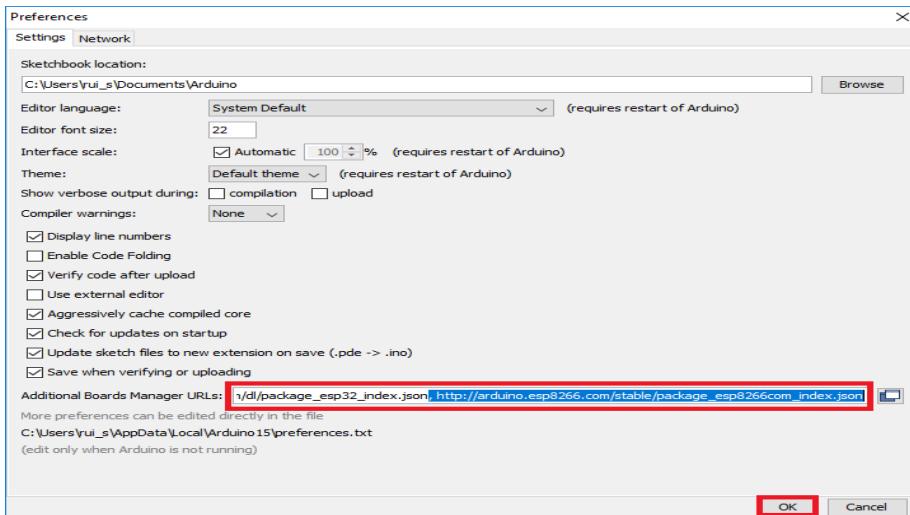
1. go to File > Preferences



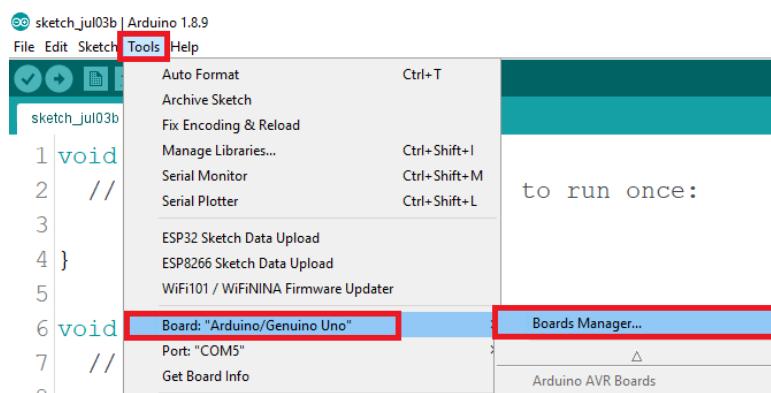
2. Enter

https://dl.espressif.com/dl/package_esp32_index.json,
http://arduino.esp8266.com/stable/package_esp8266com_index.json

into the “Additional Boards Manager URLs” field as shown in the figure below. Then, click the “OK” button:



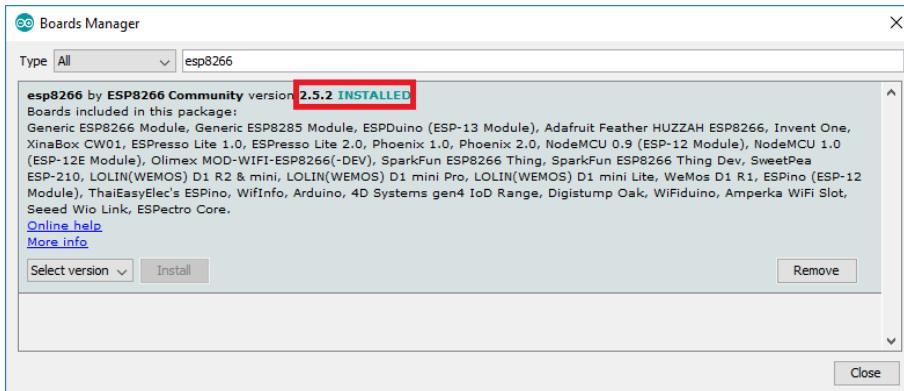
3. Open the Boards Manager. Go to Tools > Board > Boards Manager...



4. Search for ESP8266 and press install button for the “ESP8266 by ESP8266 Community“:



5. That's it. It should be installed after a few seconds.



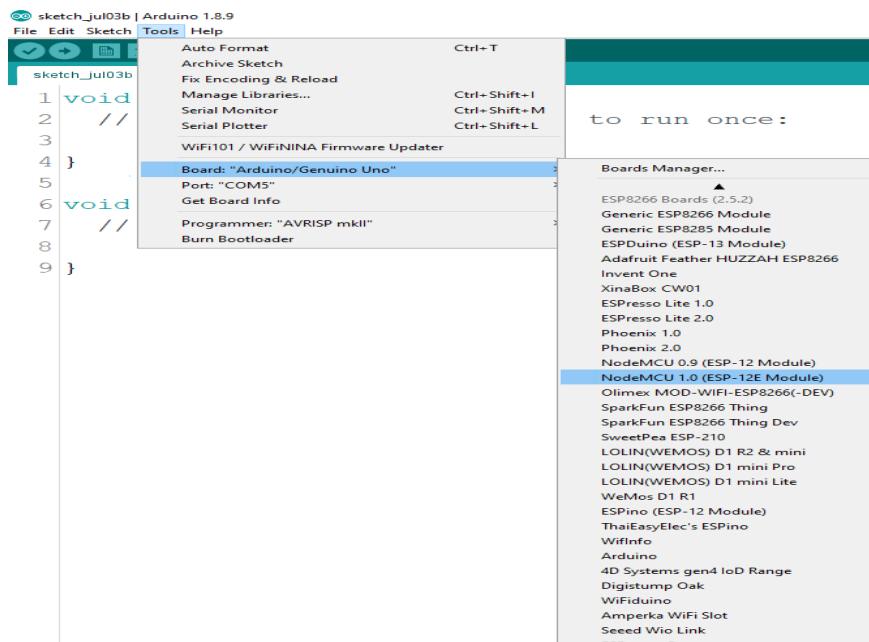
Testing the Installation

To test the ESP8266 add-on installation, let's see if we can blink an LED with the ESP8266 using the Arduino programming language.

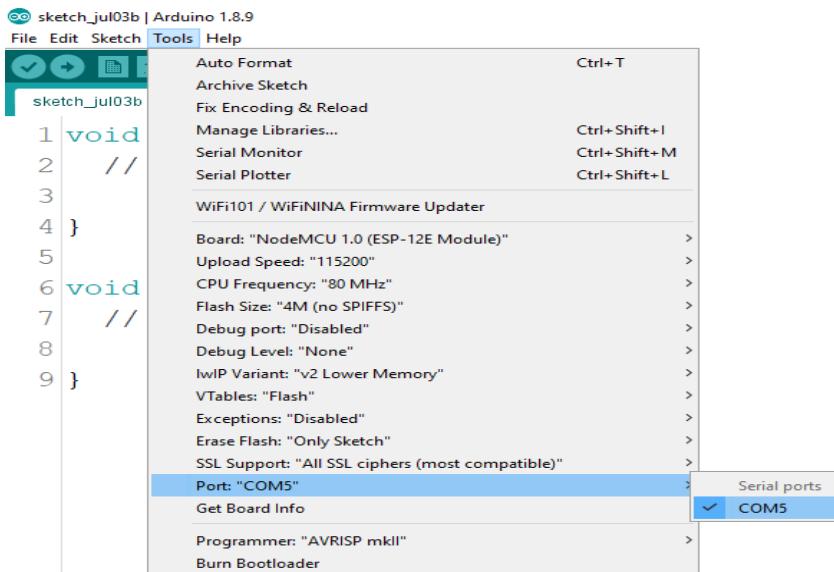
Uploading the Sketch

Uploading the Sketch to the ESP-12E

If you're using an ESP-12E NodeMCU Kit, uploading the sketch is very simple, since it has a built-in programmer. Plug your board to your computer. Make sure you have the right board selected and connect the LED to pin 2



You also need to select the Port:



Then, copy the code provided:

```
int pin = 2;
void setup() {
    pinMode(pin, OUTPUT);
}
void loop()
{
    digitalWrite(pin, HIGH); // turn the LED on (HIGH is the voltage level)
    delay(1000);           // wait for a second
    digitalWrite(pin, LOW); // turn the LED off by making the
    delay(1000);           // wait for a second
}
```

Click the “Upload” button in the Arduino IDE and wait a few seconds until you see the message “Done uploading.” in the bottom left corner.

ESP8266_Blinking_an_LED | Arduino 1.8.5

File Edit Sketch Tools Help

ESP8266_Blinking_an_LED

```
// Rui Santos
// Complete project details at http://randomnerdtutorials.com
// *****

int pin = 2;

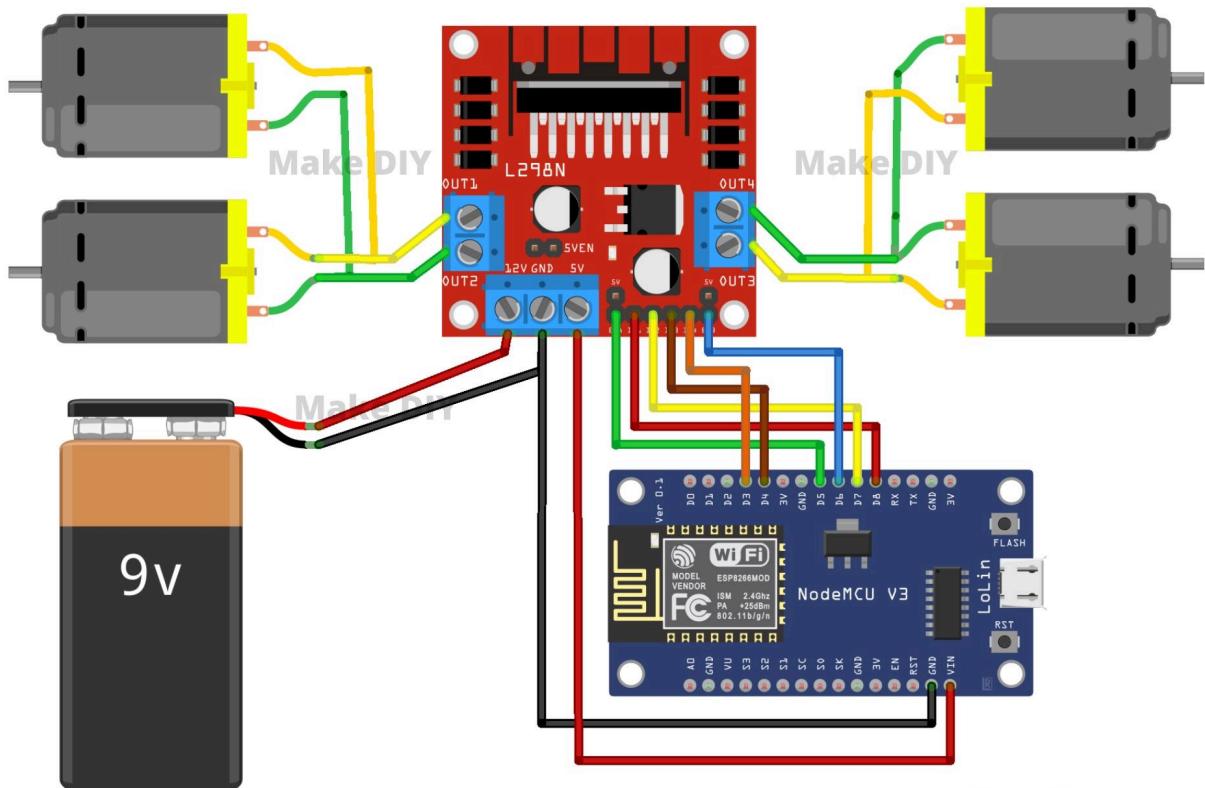
void setup() {
    // initialize GPIO 2 as an output.
    pinMode(pin, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
    digitalWrite(pin, HIGH);    // turn the LED on (HIGH is the voltage level)
    delay(1000);               // wait for a second
    digitalWrite(pin, LOW);     // turn the LED off by making the voltage LOW
    delay(1000);               // wait for a second
}
```

Done uploading.

After uploading you should see the LED blinking

Circuit Diagram



Code:

```
int m1 = 16; //Left forward GPIO D0
int m2 = 5; // left backward GPIO D1
int m3 = 4; // right forward GPIO D2
int m4 = 0; // right backward GPIO D3

#include <ESP8266WiFi.h> // Library for Wifi in Esp
#include <WiFiClient.h> // library for wifi client
#include <ESP8266WebServer.h> // library for server

String command; // String to store app command state.

const char* ssid = "Roborace"; // Wifi name
const char* pass1 = "car12345"; //Wifi Password

ESP8266WebServer server(80);

void setup() {
    // put your setup code here, to run once:
    pinMode(m1, OUTPUT);
    pinMode(m2, OUTPUT);
    pinMode(m3, OUTPUT);
    pinMode(m4, OUTPUT);

    Serial.begin(9600); // Serial monitor

    digitalWrite(m1, LOW); // motor off
    digitalWrite(m2, LOW);
    digitalWrite(m3, LOW);
    digitalWrite(m4, LOW);

    // Connecting WiFi

    WiFi.mode(WIFI_AP);
    WiFi.softAP(ssid,pass1);

    IPAddress myIP = WiFi.softAPIP();
    Serial.print("AP IP address: ");
    Serial.println(myIP);

    // Starting WEB-server
    server.on( "/", HTTP_handleRoot );
    server.onNotFound( HTTP_handleRoot );
```

```
server.begin();
}

void goAhead()
{
digitalWrite(m1, HIGH);
digitalWrite(m2, LOW);
digitalWrite(m3, HIGH);
digitalWrite(m4, LOW);
}

void goBack()
{
digitalWrite(m1, LOW);
digitalWrite(m2, HIGH);
digitalWrite(m3, LOW);
digitalWrite(m4, HIGH);
}

void goRight()
{
digitalWrite(m1, HIGH);
digitalWrite(m2, LOW);
digitalWrite(m3, LOW);
digitalWrite(m4, HIGH);
}

void goLeft()
{
digitalWrite(m1, LOW);
digitalWrite(m2, HIGH);
digitalWrite(m3, HIGH);
digitalWrite(m4, LOW);
}

void stopRobot()
{
digitalWrite(m1, LOW);
digitalWrite(m2, LOW);
digitalWrite(m3, LOW);
digitalWrite(m4, LOW);
}

void loop() {
```

```
server.handleClient();

command = server.arg("State");

if (command == "F")
    goAhead();

else if (command == "B")
    goBack();

else if (command == "L")
    goLeft();

else if (command == "R")
    goRight();

else if (command == "S")
    stopRobot();

}

void HTTP_handleRoot(void)
{
    if (server.hasArg("State"))
    {
        Serial.println(server.arg("State"));
    }
    server.send(200, "text/html", "");
    delay(1);
}
```

Final Look

