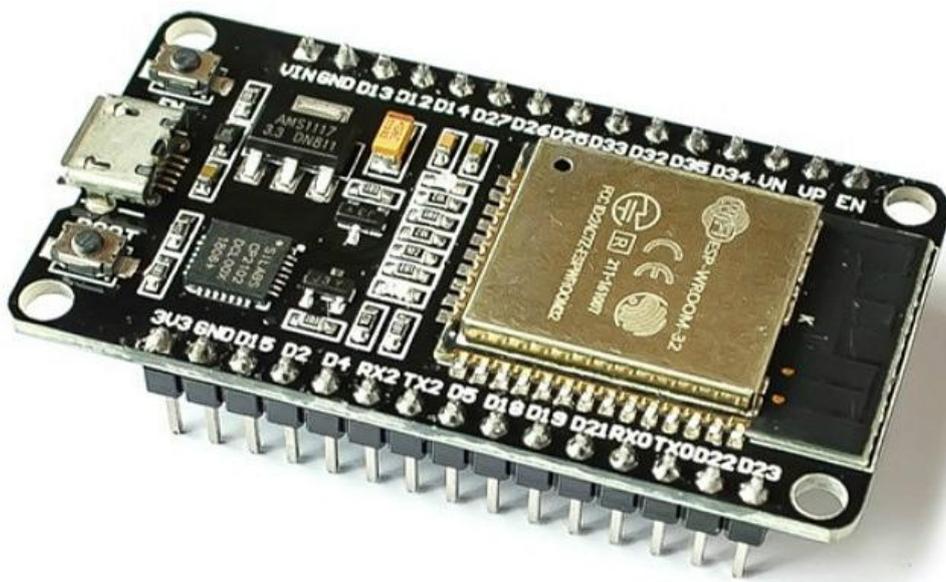# NodeMCU IoT ESP32 ESP-32 Wifi & Bluetooth Development Board

Introduction:

The NodeMCU ESP-32S is an IoT (Internet of Things) development board created by NodeMcu to evaluate the ESP-WROOM-32 module. It is based on the ESP32 microcontroller that supports WiFi, Bluetooth, Ethernet, and Low Power support all in a single chip.

The NodeMCU-32S board is also an open source IoT platform. It uses Lua script and a lot of open source projects, such as lua-cjson, and spiffs. It contains firmware that can run on ESP32 WiFi SoC chips, and its hardware is based on ESP-32S modules. The ESP32 is a WiFi + Bluetooth chip developed by Shanghai Espressif, which aims to provide access network for embedded products.

The development board maintains the classic NodeMCU 1.0 design, with the I / O pins placed on either side of the board, which provides a more straightforward setup and breadboard use. Using a breadboard for development can simplify the use of this module by making use of both sides of the standard pin. Open source, interactive, programmable, intelligent & straightforward, plus containing WIFI + Bluetooth hardware makes the NodeMCU ESP-32S a great little board to keep in your toolkit.

Components:

- NodeMCU IoT ESP32 ESP-32 Wifi & Bluetooth Development BoardBreadboard

- Blynk App in Android or iOS apps

- Breadboard

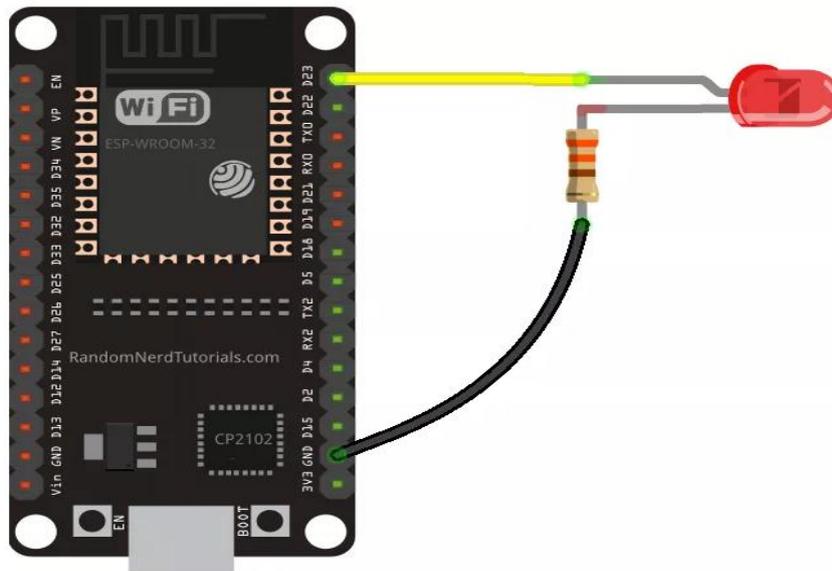- 330 ohm resistor

Objective:

To Control LED With Blynk via WiFi

Procedure:

Step 1: Build a circuit.

Below are the connection between NodeMCU IoT ESP32 ESP-32 Wifi & Bluetooth Development Board and LED.

| NodeMCU IoT ESP32 ESP-32 Wifi & Bluetooth Development Board	LED
GND	NEGATIVE (-)
D1	POSITIVE (+)

Step 2: Download Blynk app in your phone from Google play store or IOS and install it.

**Step 3:** After installing, you need to create an account in this app; you may use your current Gmail account.
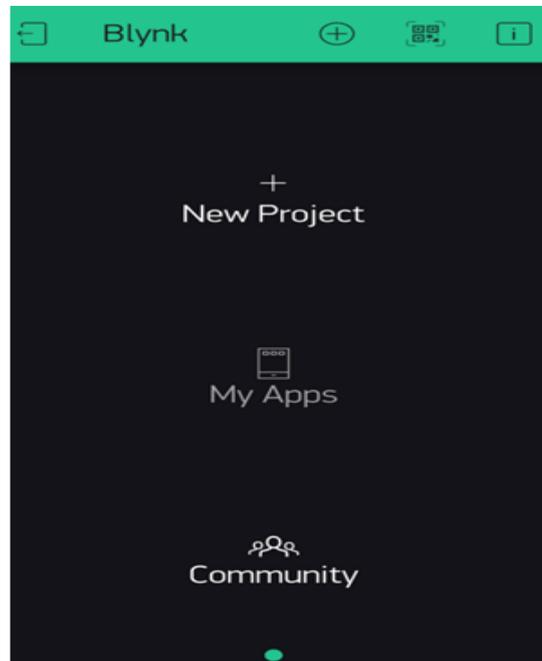

Step 4: After creating account a window will open, in this click on New Project

Step 5: Give project a name according to your choice and in device choose ESP32 Dev Board and in Connection type choose Wi-Fi and then click on Create.
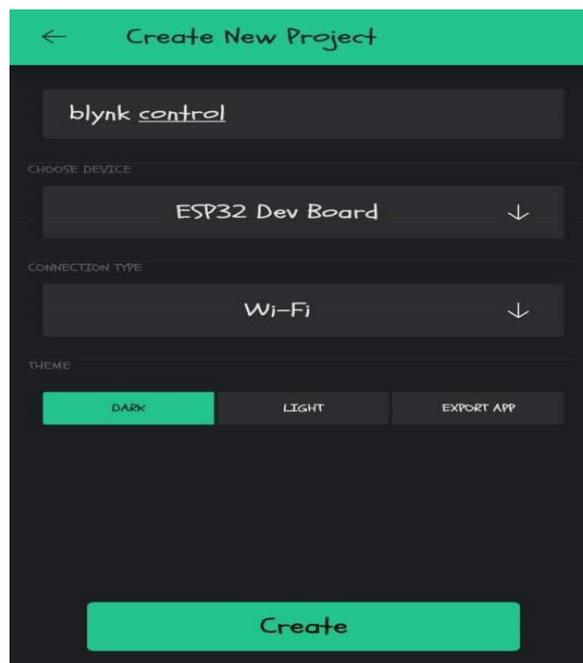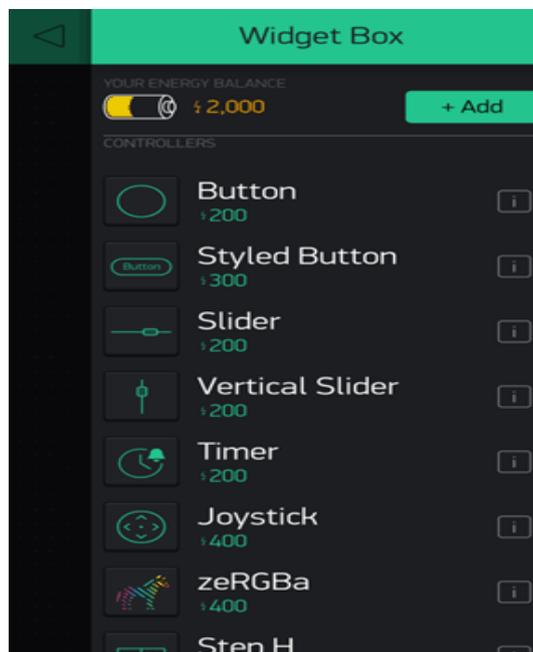
Step 6: Now a window will come which shows that your authentication token which you will need later sent to your concerned mail ID. You can open your email to check authentication key.
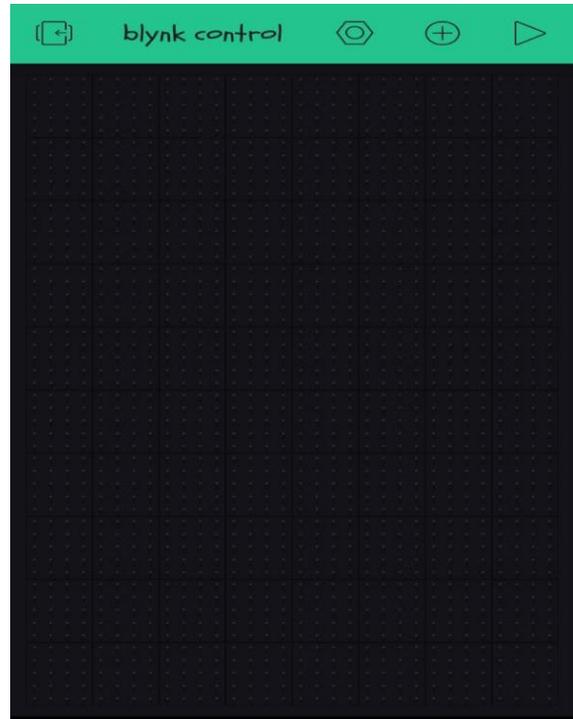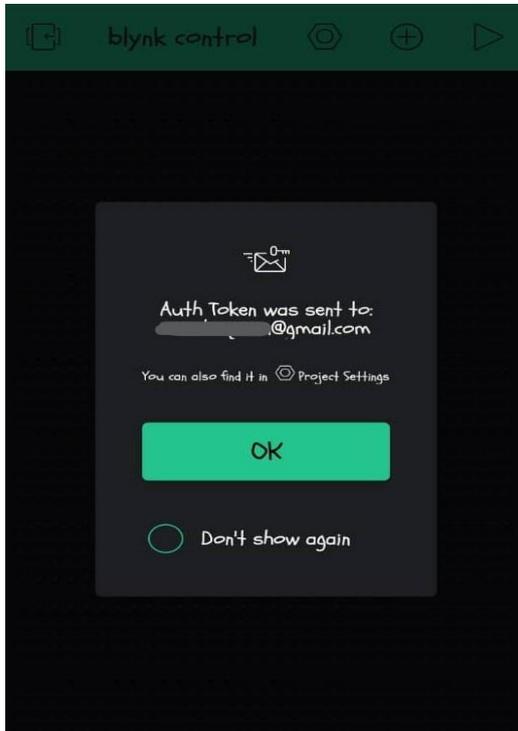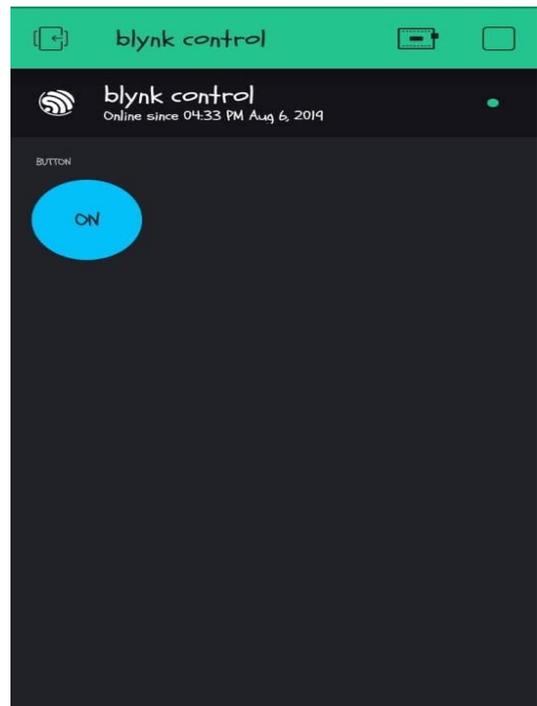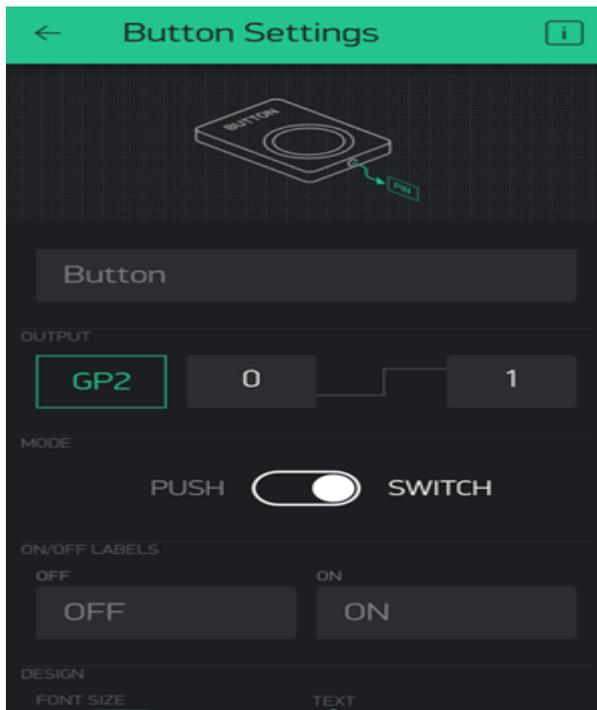
Step 7: After clicking on OK, you will find canvas window.

Step 8: Tap anywhere on the canvas to open the widget box. All the available widgets are located here. Now choose a button.

Step 9: Click on the Button widget to change the setting.

Step 10: Set output pin to gp2 as I am taking output here from GPIO2 pin, you can change according to you. In mode, select to switch.

Step 11: When you are done with this setting, you are ready to work with this app. On pressing Play button it will switch you from EDIT mode to PLAY mode where you can interact with the hardware. In PLAY mode, you will not be able to drag or set new widgets, press STOP to get back to STOP mode.

Step 12: Insert the sample programming provided below by copy and paste it into Arduino IDE.

```
#define BLYNK_PRINT Serial
#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>

int pin = 23;

char auth[] = "wJhYryrFM4bkUnuDVQyUNsDg44_yNXrK";      // You should get Auth
Token in the Blynk App.
char ssid[] = "Synacorp-2.4G";               // Your Wi-Fi Credentials
char pass[] = "**********";

void setup() {
 pinMode (pin, OUTPUT);
 pinMode (pin, HIGH);
 Serial.begin(115200);
 delay(10);
 Serial.print("Connecting to ");
 Serial.println(ssid);
 WiFi.begin(ssid, pass);
 int wifi_ctr = 0;
 while (WiFi.status() != WL_CONNECTED) {
 delay(500);
 Serial.print(".");
 }
 Serial.println("WiFi connected");
 Blynk.begin("wJhYryrFM4bkUnuDVQyUNsDg44_yNXrK ", ssid, pass);

}
void loop(){
   Blynk.run();
}
```
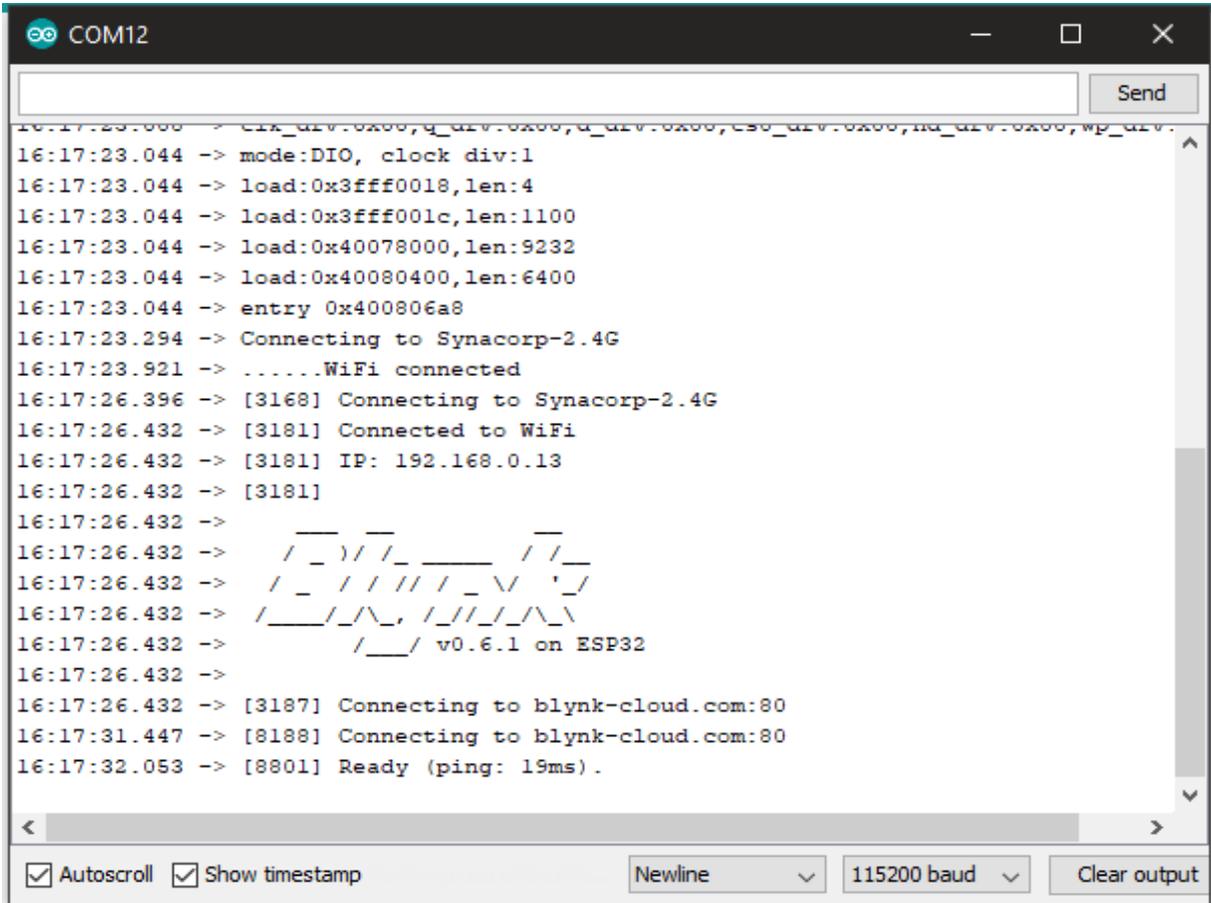
Step 13: Open the serial monitor to observe the result as shown below.

```
COM12                                                        —    □    ×

                                                                    Send

16:17:23.000 -> clk_div:0x00,q_div:0x00,d_div:0x00,cs0_div:0x00,hd_div:0x00,wp_div:
16:17:23.044 -> mode:DIO, clock div:1
16:17:23.044 -> load:0x3fff0018,len:4
16:17:23.044 -> load:0x3fff001c,len:1100
16:17:23.044 -> load:0x40078000,len:9232
16:17:23.044 -> load:0x40080400,len:6400
16:17:23.044 -> entry 0x400806a8
16:17:23.294 -> Connecting to Synacorp-2.4G
16:17:23.921 -> ......WiFi connected
16:17:26.396 -> [3168] Connecting to Synacorp-2.4G
16:17:26.432 -> [3181] Connected to WiFi
16:17:26.432 -> [3181] IP: 192.168.0.13
16:17:26.432 -> [3181]
16:17:26.432 ->       ___  __              __
16:17:26.432 ->      / _ )/ /_ _____  ___ / /__
16:17:26.432 ->     / _  / / // / _ \/ '_/
16:17:26.432 ->    /____/_/\_, /_//_/_/\_\
16:17:26.432 ->          /___/ v0.6.1 on ESP32
16:17:26.432 ->
16:17:26.432 -> [3187] Connecting to blynk-cloud.com:80
16:17:31.447 -> [8188] Connecting to blynk-cloud.com:80
16:17:32.053 -> [8801] Ready (ping: 19ms).

<                                                                     >

☑ Autoscroll ☑ Show timestamp    Newline  ∨   115200 baud ∨   Clear output
```