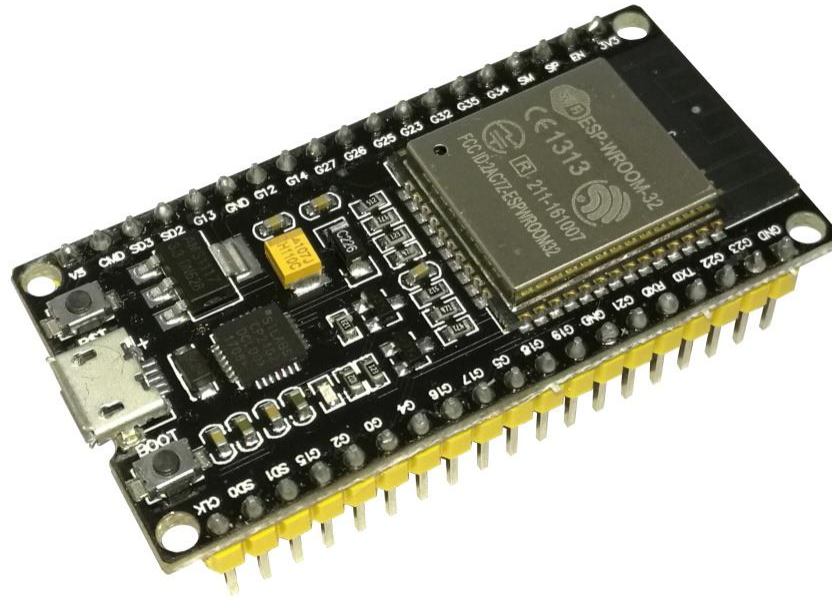


Micropython for D1 ESP8266/ESP32

Getting started Guide



By Subrata Sarker <picklumithu@yahoo.com>

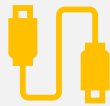
Setting up



1. Download and install Thony IDE for Micropython.



2. Download Mircopython firmware (*.bin file) for ESC32 generic.




3. Connect D1 ESP8266 or ESP32 to your PC via USB.




4. Open Micropython IDE Thony.

Setting up

5. Go to menu → Run → Select Interpreter: Set the interpreter as MicroPython (ESP8266) or MicroPython (ESP32) and Port or WebREPL as the port that your microcontroller is using.



6. From there click → Install or update firmware → A new window titled “ESP8266 firmware installer” or “ESP32 firmware installer” will pop up.



7. Set the port as before. Select the firmware file that you have already downloaded at step 2.

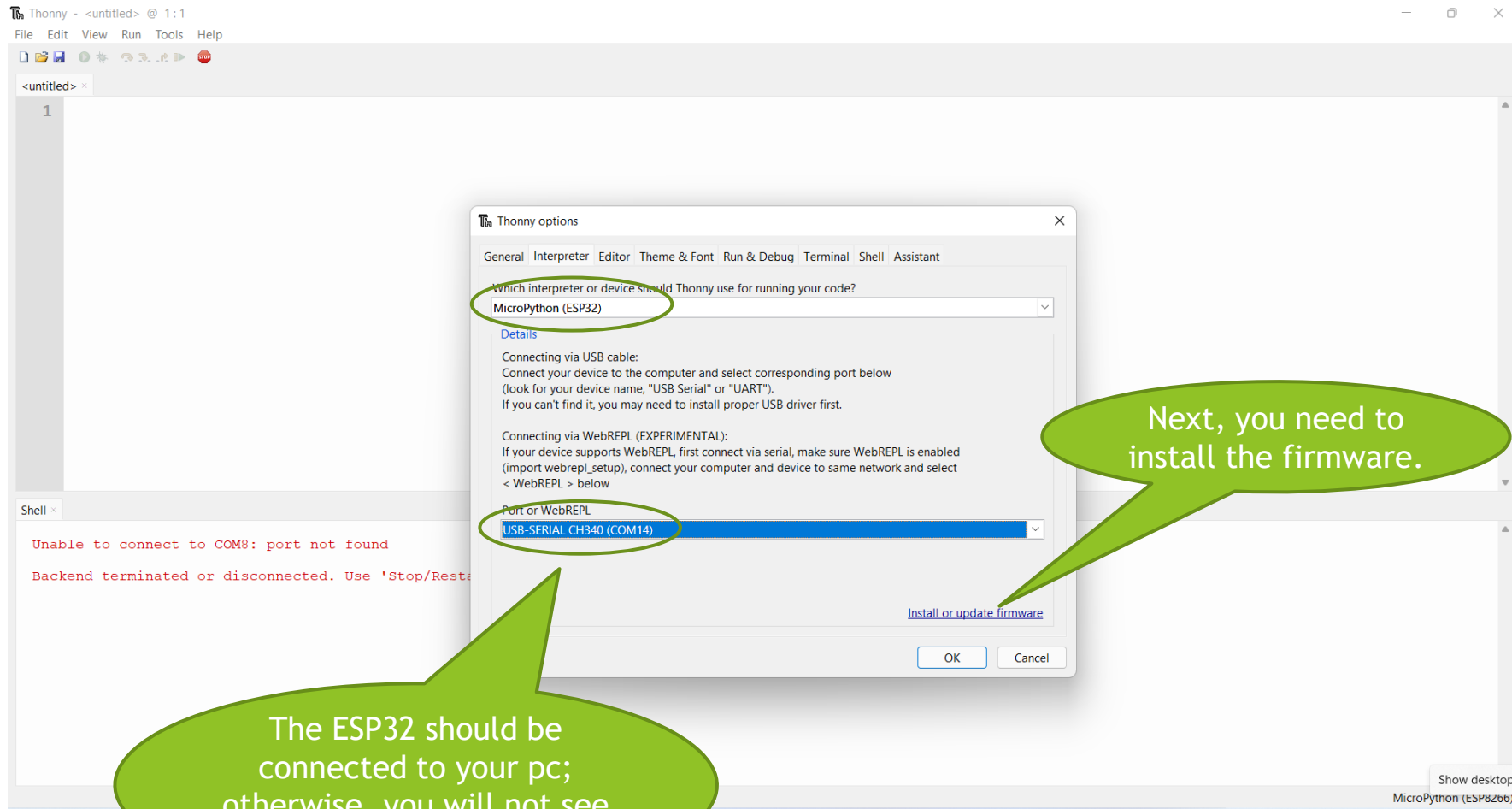
Setting up

8. Click install; it may take a few minutes.
Close the window when done.

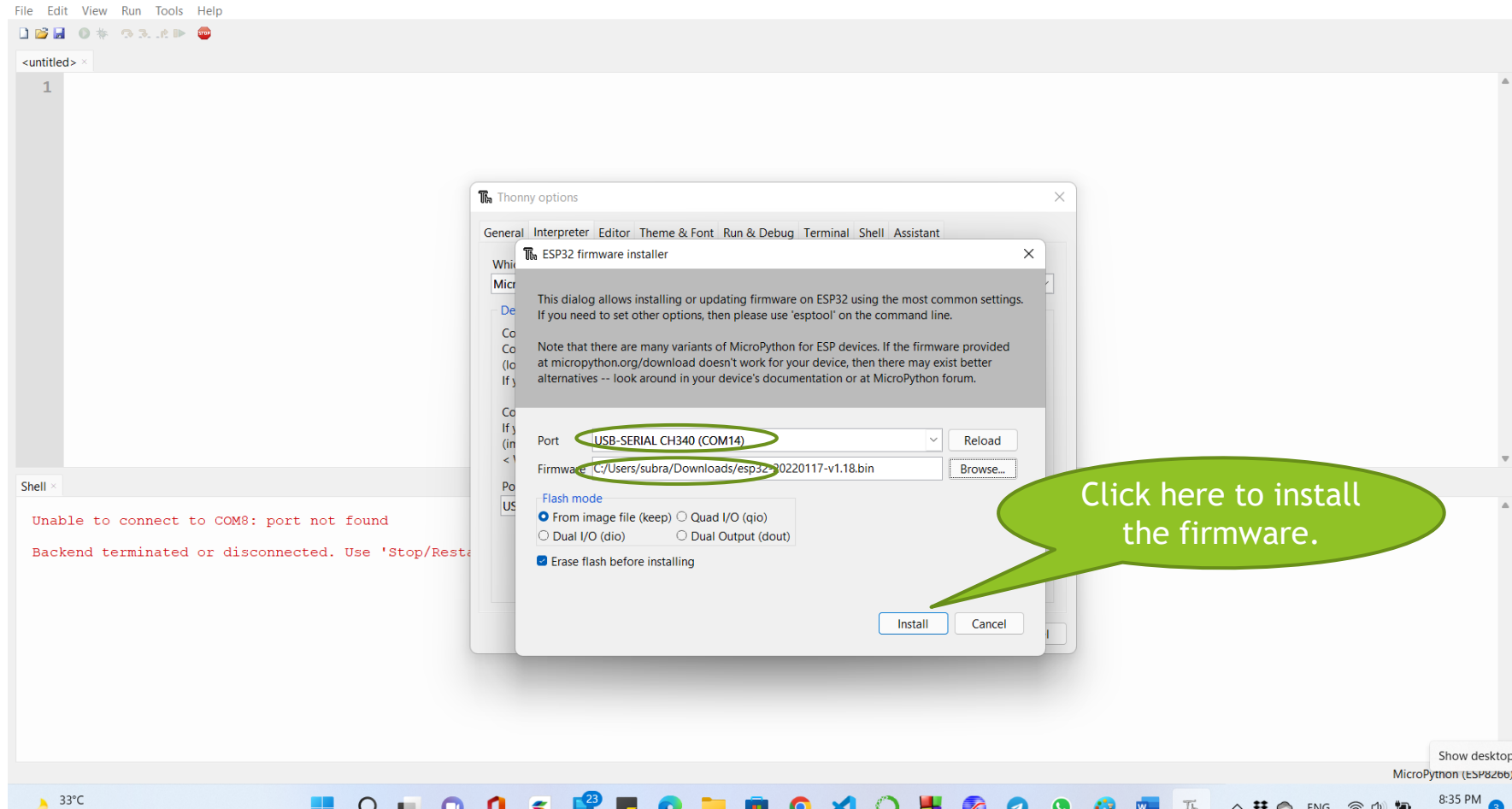


Now your microcontroller is ready to
be programmed using *Micropython*.

Setting up



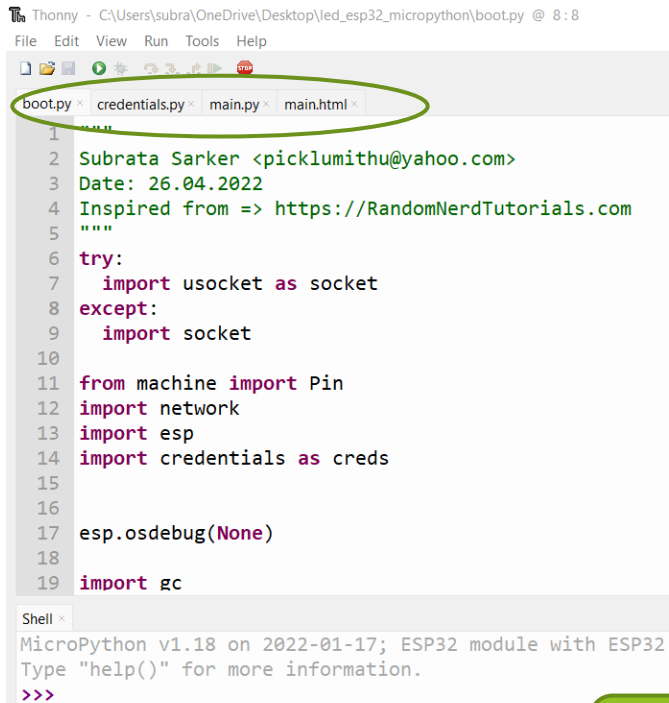
Setting up



Setting up



Program your D1 ESP8266 or ESP32 to control a LED from web browser



The screenshot shows the Thonny IDE interface. The top menu bar includes File, Edit, View, Run, Tools, and Help. Below the menu bar, there are four tabs: boot.py, credentials.py, main.py, and main.html. The boot.py tab is active, showing the following code:

```
1 """
2 Subrata Sarker <picklumithu@yahoo.com>
3 Date: 26.04.2022
4 Inspired from => https://RandomNerdTutorials.com
5 """
6 try:
7     import usocket as socket
8 except:
9     import socket
10
11 from machine import Pin
12 import network
13 import esp
14 import credentials as creds
15
16
17 esp.osdebug(None)
18
19 import gc
```

Below the code editor, there is a Shell window showing the MicroPython v1.18 on 2022-01-17; ESP32 module with ESP32. Type "help()" for more information. >>>

Open the required files that are available in the repo:

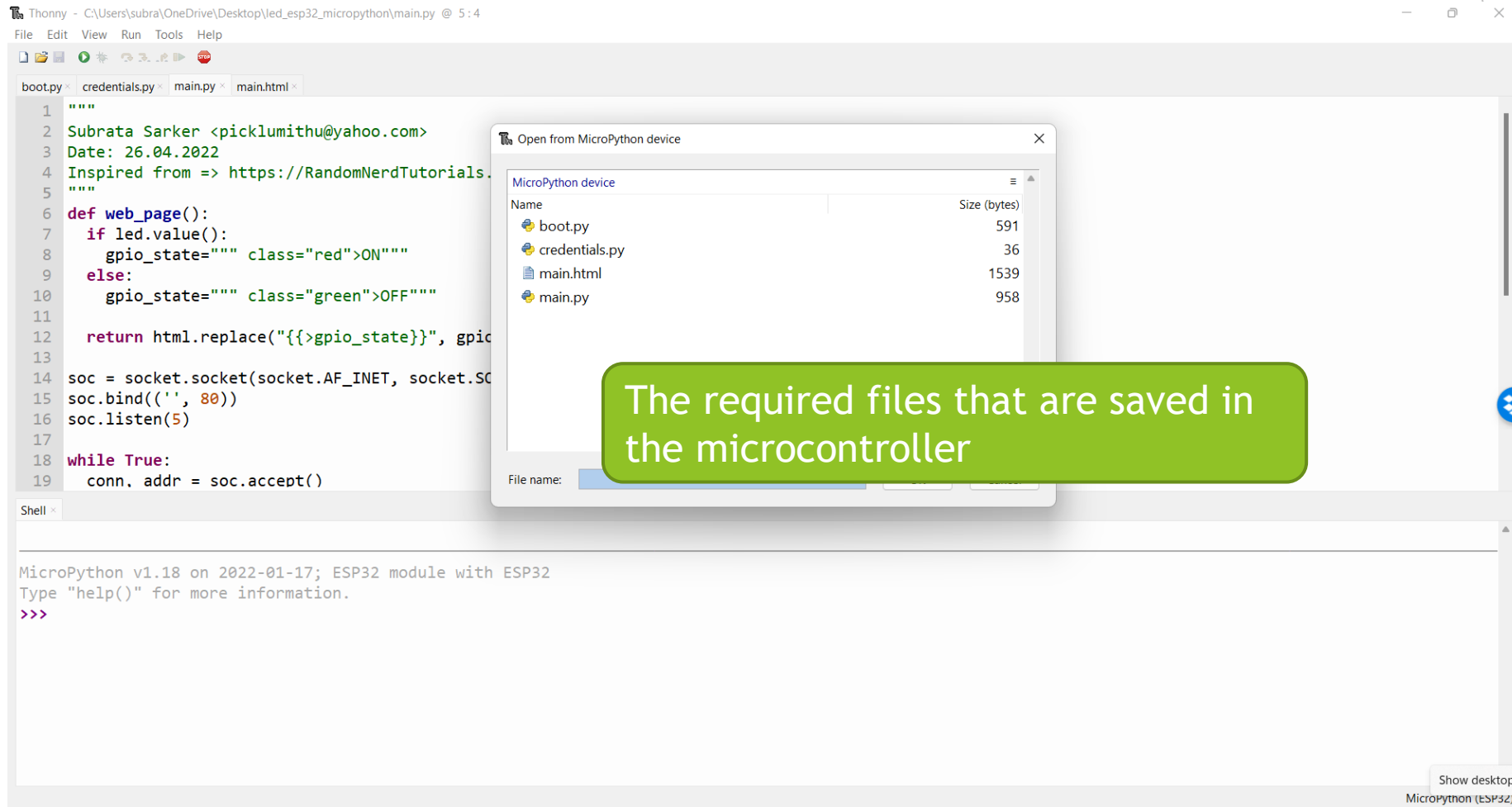
1. boot.py
2. credentials.py
3. main.py
4. main.html

Save a copy of the files to the microcontroller using the file menu.

You can see the files in the microcontroller by clicking here.

*** You must update the credentials.py file with your wifi “ssid” and “password”.

Program your D1 ESP8266 or ESP32 to control a LED from web browser



Program your D1 ESP8266 or ESP32 to control a LED from web browser

Thonny - C:\Users\subra\OneDrive\Desktop\led_esp32_micropython\main.py @ 5:4

File Edit View Run Tools Help

boot.py x credentials.py x main.py x main.html x

```
1 """
2 Subrata Sarker <picklumithu@yahoo.com>
3 Date: 26.04.2022
4 Inspired from => https://RandomNerdTutorials.com
5 """
6 def web_page():
7     if led.value():
8         gpio_state="" class="red">ON""
9     else:
10         gpio_state="" class="green">OFF""
11
12     return html.replace("{}>gpio_state{}", gpio_state)
13
14 soc = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
15 soc.bind(('', 80))
16 soc.listen(5)
17
18 while True:
19     conn, addr = soc.accept()
```

Shell

```
>>> import os
>>> os.listdir()
['boot.py', 'credentials.py', 'main.html', 'main.py']
>>>
```

MicroPython (ESP32)

You can list the files on ESP32 by running the python code as shown above.

Program your D1 ESP8266 or ESP32 to control a LED from web browser

```
Thonny - C:\Users\subra\OneDrive\Desktop\led_esp32_micropython\main.py @ 5:4
File Edit View Run Tools Help

boot.py | credentials.py | main.py | main.html |
1 """
2 Subrata Sarker <picklumithu@yahoo.com>
3 Date: 26.04.2022
4 Inspired from => https://RandomNerdTutorials.com
5 """
6 def web_page():

Shell |
>>> import os
>>> os.listdir()

['boot.py', 'credentials.py', 'main.html', 'main.py']
Hrst' (PUI=9) ESETJTI 0x13 * } | MT_FLASH * % == Q %
T } n.c S1j000
etsR 016 :22:MSH
ip: 0 A} A 0xeeET) I 3 (S Re|MQ) LASH% == Q
ckE 0x00, q 0x0bd_drv' 0, c. 0x00, hE 000, wpE 0x
mov "%= clock V 2
+ 0x s T len: S!! j
8 1 12 4
load' 00!! bb:n:12%!! S tai 1 m 4
+ 0x!! blen: L&SH ry 0!! j

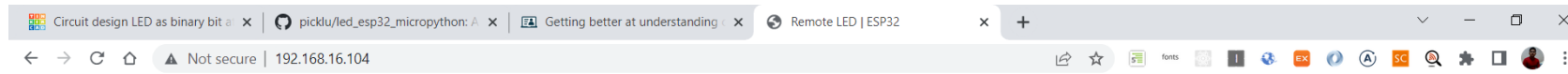
Brownout detector was triggered

ets Jun 8 2016 00:22:57

rst:0xc (SW_CPU_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:2
load:0x3fff0030,len:4540
ho 0 tail 12 room 4
load:0x40078000,len:12344
ho 0 tail 12 room 4
load:0x40080400,len:4124
entry 0x40080680
Connection successful
('192.168.16.104', '255.255.255.0', '192.168.16.1', '192.168.16.1')
```

Once the reset pin of the ESP32 is clicked, the Micropython “repl” will be populated with text like this indicating that the web server powered by ESP32 is running at ‘192.168.16.104’.

Program your D1 ESP8266 or ESP32 to control a LED from web browser



Remote LED Server Powered by ESP32

GPIO state: **OFF**

ON

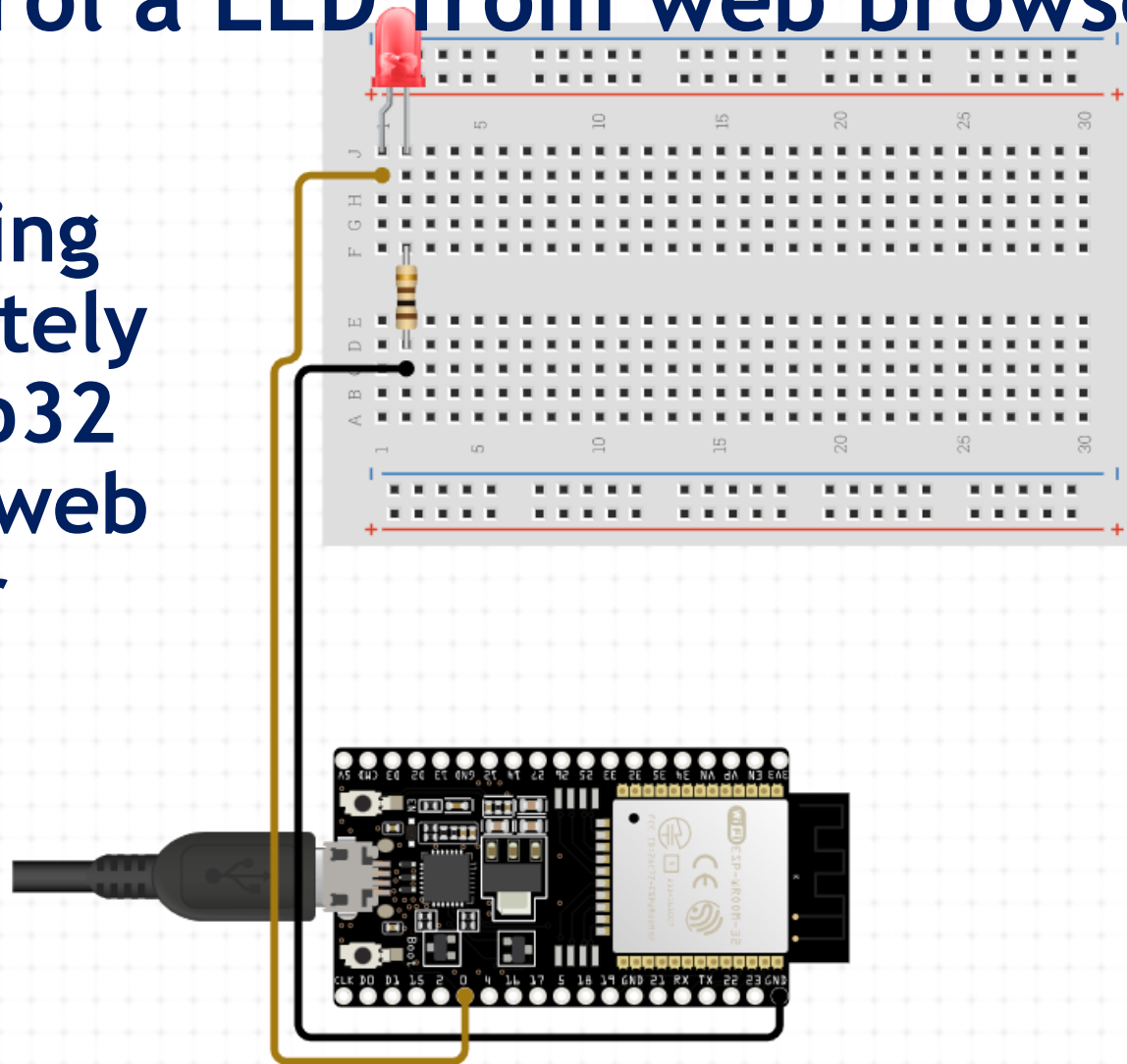
OFF

Open a browser and type the ip address that you found in the previous page; you will see a page like this. If you have followed along then you have created your first micro web server using Micropython on ESP32. Now play with button and see the action on the circuit as provide in the next page.

Cheers!

Program your D1 ESP8266 or ESP32 to control a LED from web browser

Controlling
LED remotely
using ESsp32
powered web
server





Useful Links

- <https://micropython.org/>
- <https://micropython.org/download/esp8266/>
- <https://micropython.org/download/esp32/>
- <https://micropython.org/resources/firmware/esp8266-20220117-v1.18.bin>
- <https://micropython.org/resources/firmware/esp32-20220117-v1.18.bin>
- <https://docs.micropython.org/en/latest/>
- <https://docs.micropython.org/en/latest/esp32/quickref.html>
- https://github.com/picklu/led_esp32_micropython
- <https://github.com/jczic/MicroWebSrv2>

Happy Coding!!!

