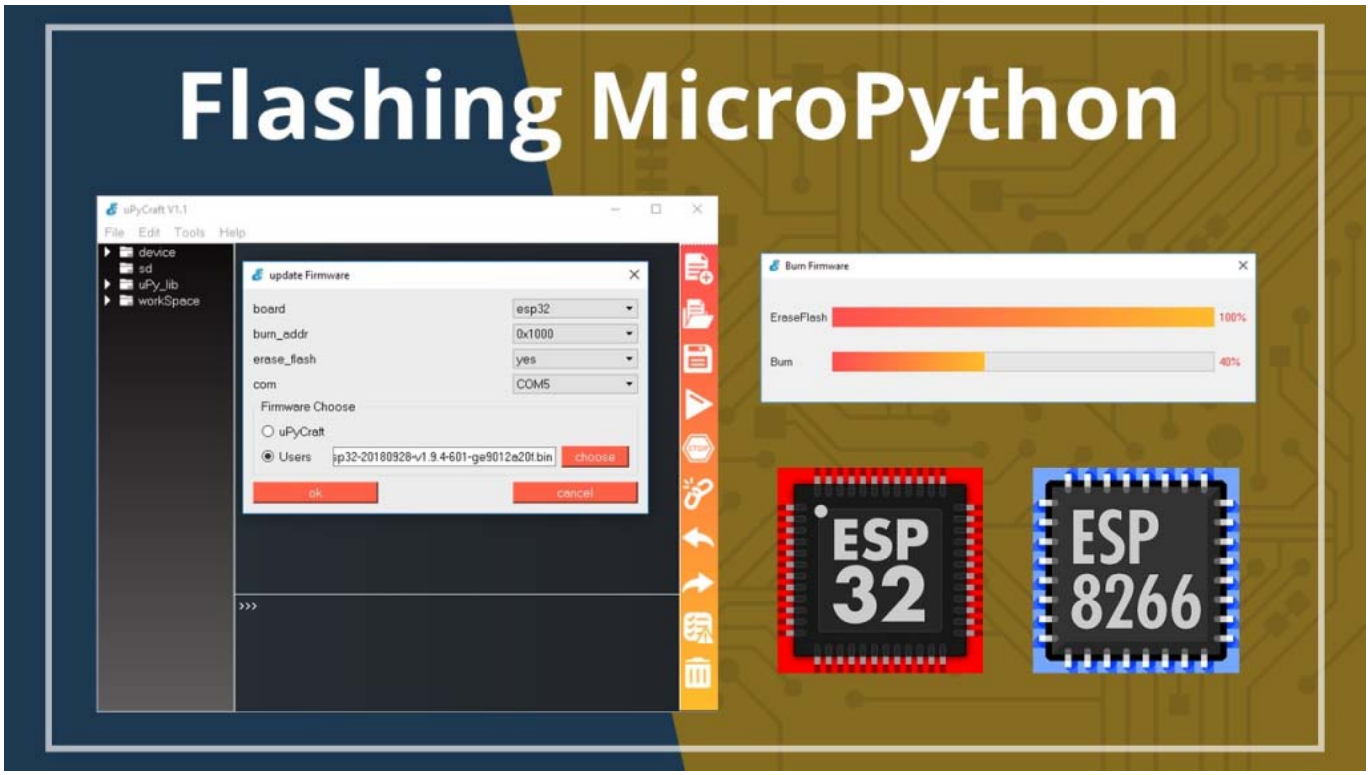
Flash/Upload MicroPython Firmware to ESP32 and ESP8266

This posts shows how to flash MicroPython firmware to ESP32/ESP8266 boards using the uPyCraft IDE software. It works on Windows, Linux, and Mac OS X.



Before continuing with this tutorial, make sure you follow one of these guides to install uPyCraft IDE on your computer:

- Windows PC – Install uPyCraft IDE
- Mac OS X – Install uPyCraft IDE
- Linux Ubuntu – Install uPyCraft IDE

With uPyCraft IDE installed in your computer, you can easily flash your ESP32 or ESP8266 boards with the MicroPython firmware. This post is divided in two parts, read Part 1 or Part 2 depending on your board:

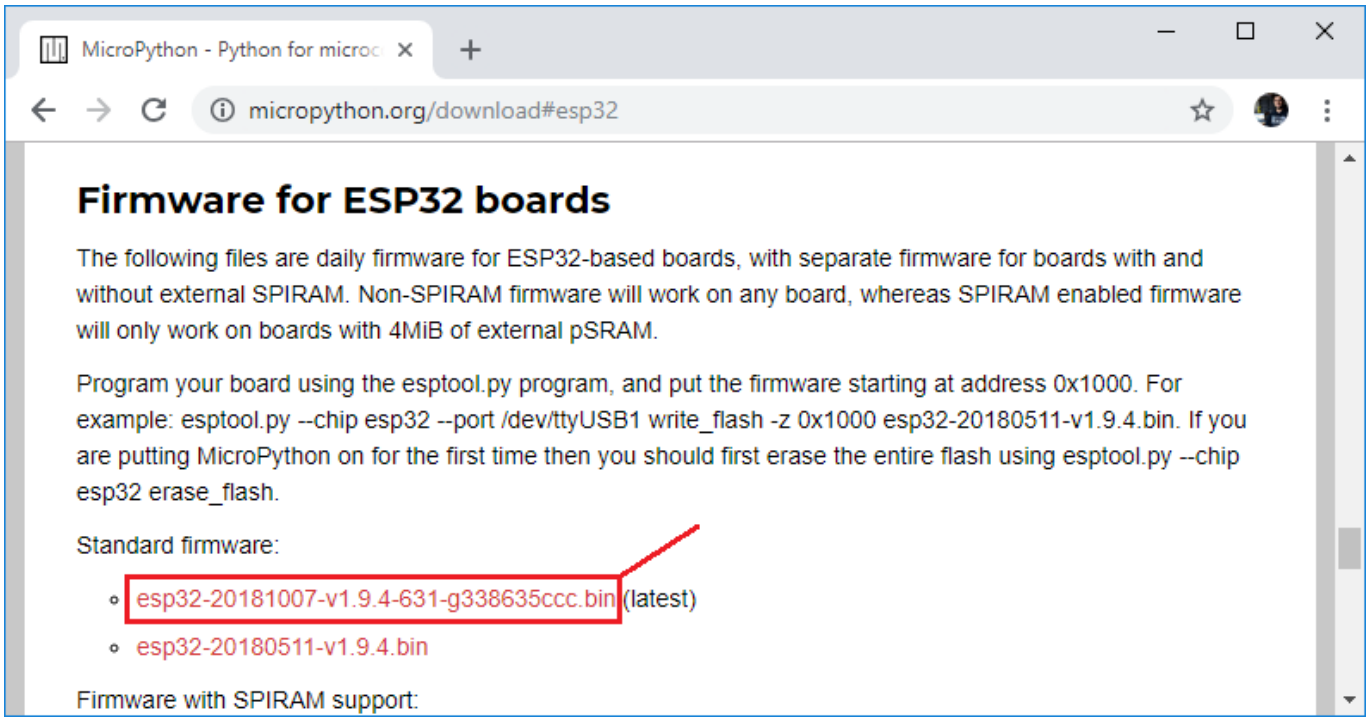
- Part 1 – ESP32
- Part 2 – ESP8266

Note: After installing MicroPython firmware on your ESP32 or ESP8266, you can go back and use Arduino IDE again. You just need to upload code using Arduino IDE. Then, if you want to use MicroPython again, you need to flash MicroPython firmware.

[Part 1 – ESP32] Downloading and Flashing the MicroPython Firmware on ESP32

To download the latest version of MicroPython firmware for the ESP32, go to the MicroPython Downloads page and scroll all the way down to the ESP32 section.

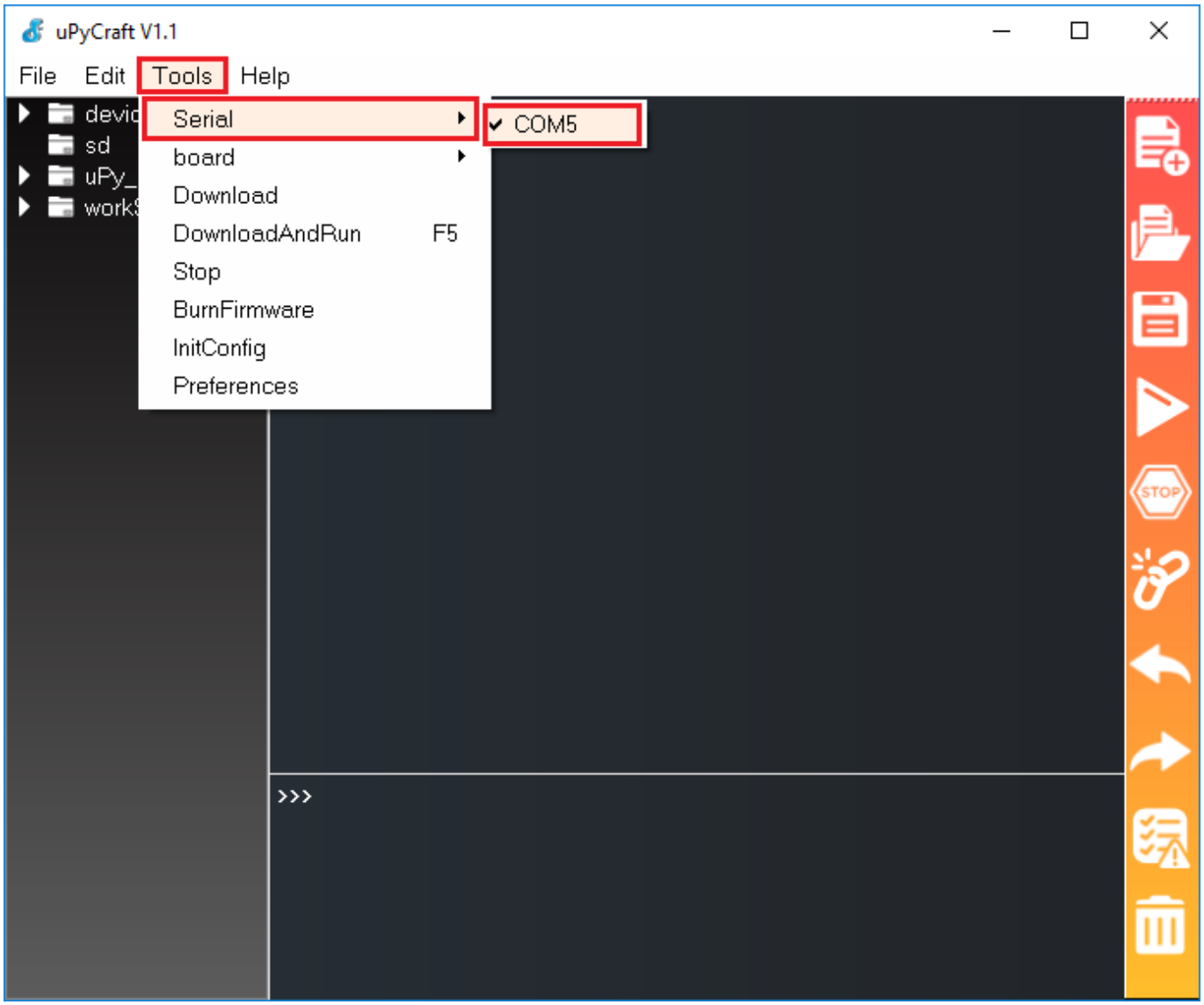
You should see a similar web page (see figure below) with the latest link to download the ESP32 *.bin* file – for example: *esp32-20181007-v1.9.4-631-g338635ccc.bin*.



Note: if you're using a different board (like a PyBoard, WiPy, or other), go to MicroPython Downloads page and download the right firmware for your board.

Selecting Serial Port

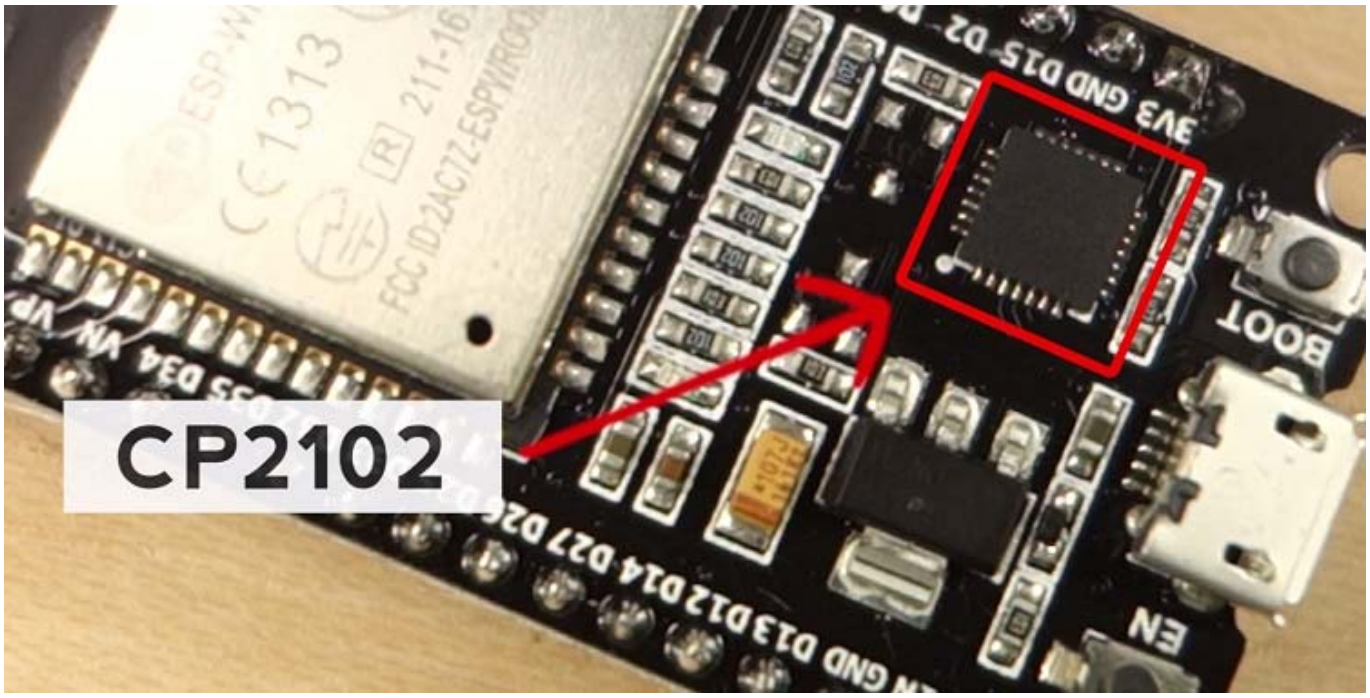
Go to **Tools** > **Serial** and select your ESP32 COM port (in our case it's COM5).



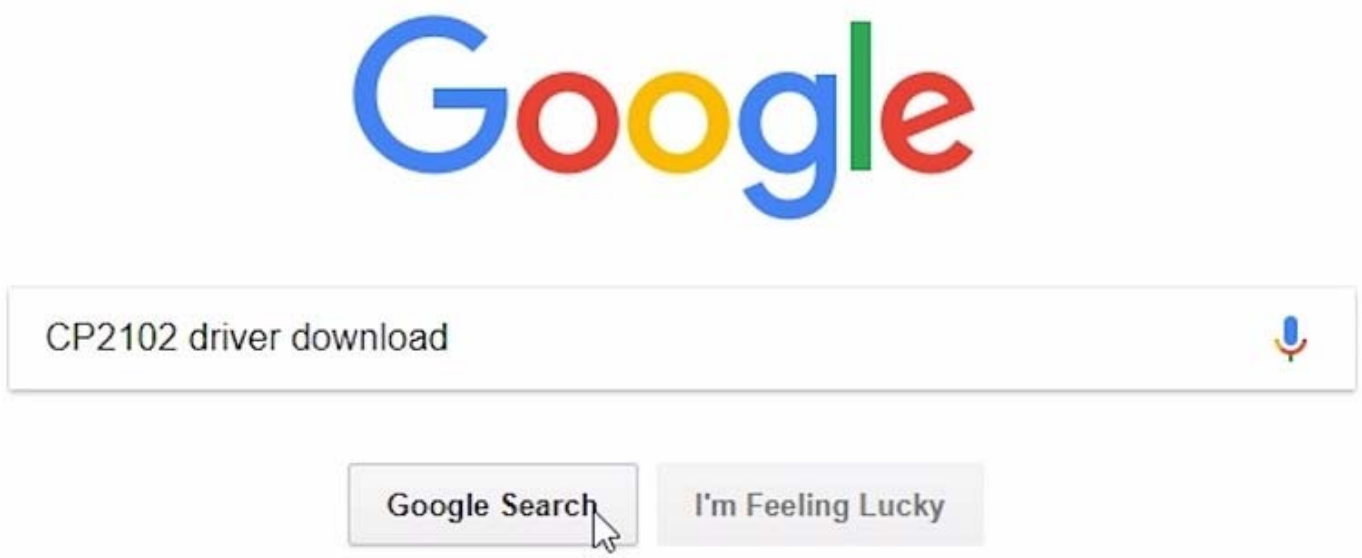
Important: if you plug your ESP32 board to your computer, but you can't find the ESP32 Port available in your uPyCraft IDE, it might be one of these two problems: **1.** USB drivers missing or **2.** USB cable without data wires.

1. If you don't see your ESP's COM port available, this often means you don't have the USB drivers installed. Take a closer look at the chip next to the voltage regulator on board and check its name.

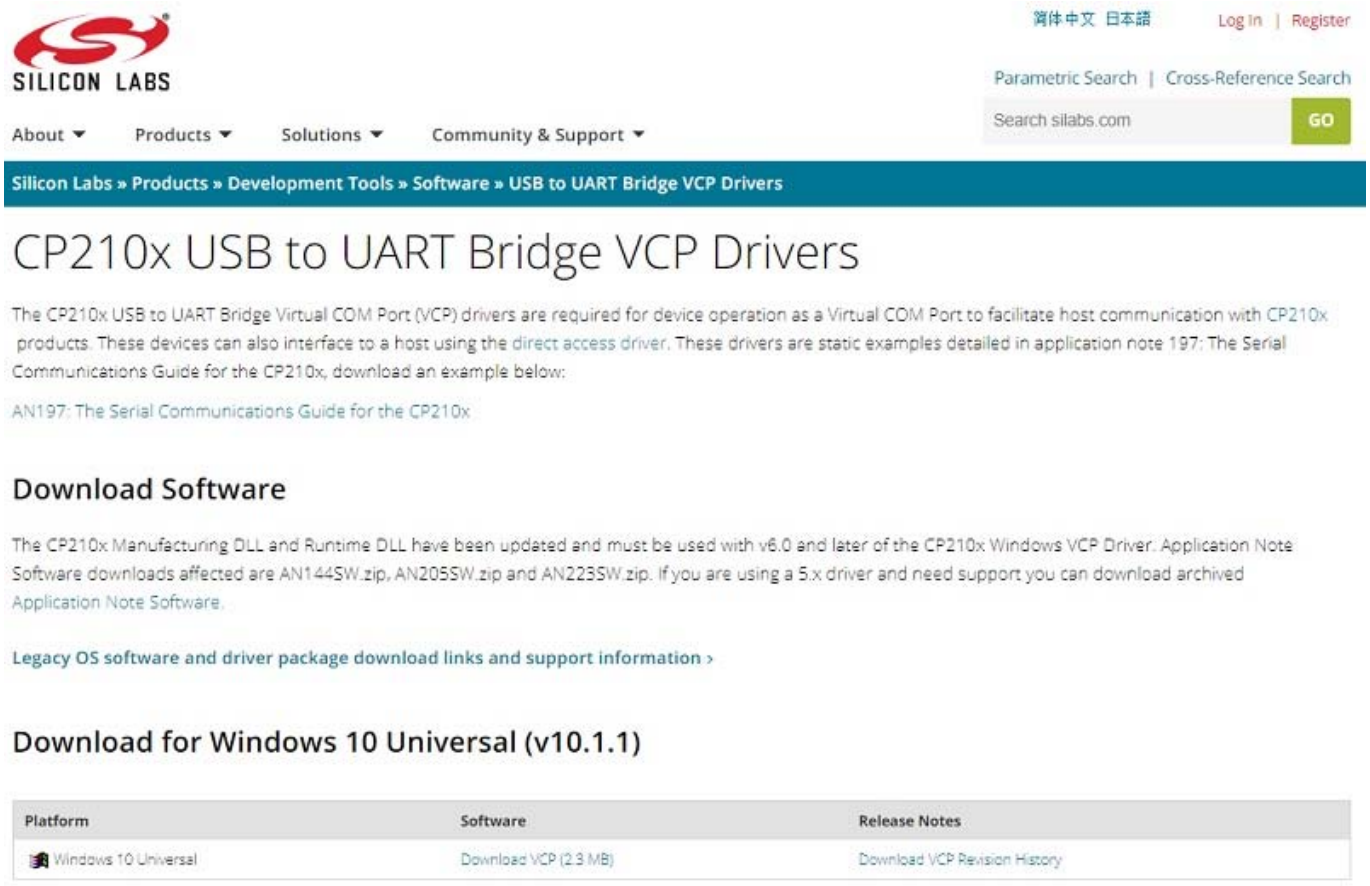
The ESP32 DEVKIT V1 DOIT board uses the **CP2102** chip.



Go to Google and search for your specific chip to find the drivers and install them in your operating system.



You can download the CP2102 drivers on the Silicon Labs website.



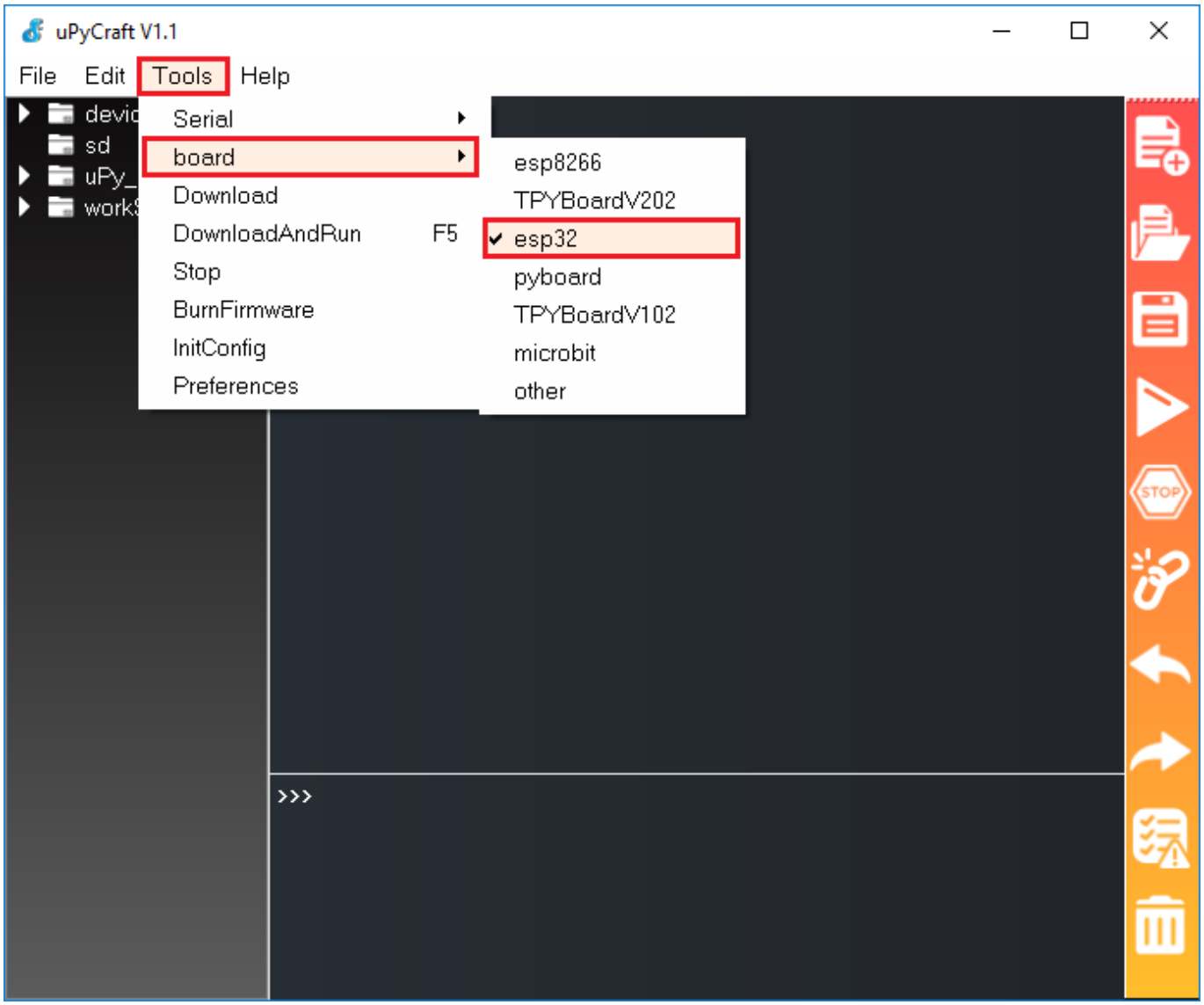
After they are installed, restart the uPyCraft IDE and you should see the COM port in the **Tools** menu.

2. If you have the drivers installed, but you can't see your device, double-check that you're using a USB cable with data wires.

USB cables from powerbanks often don't have data wires (they are charge only). So, your computer will never establish a serial communication with your ESP32. Using a a proper USB cable should solve your problem.

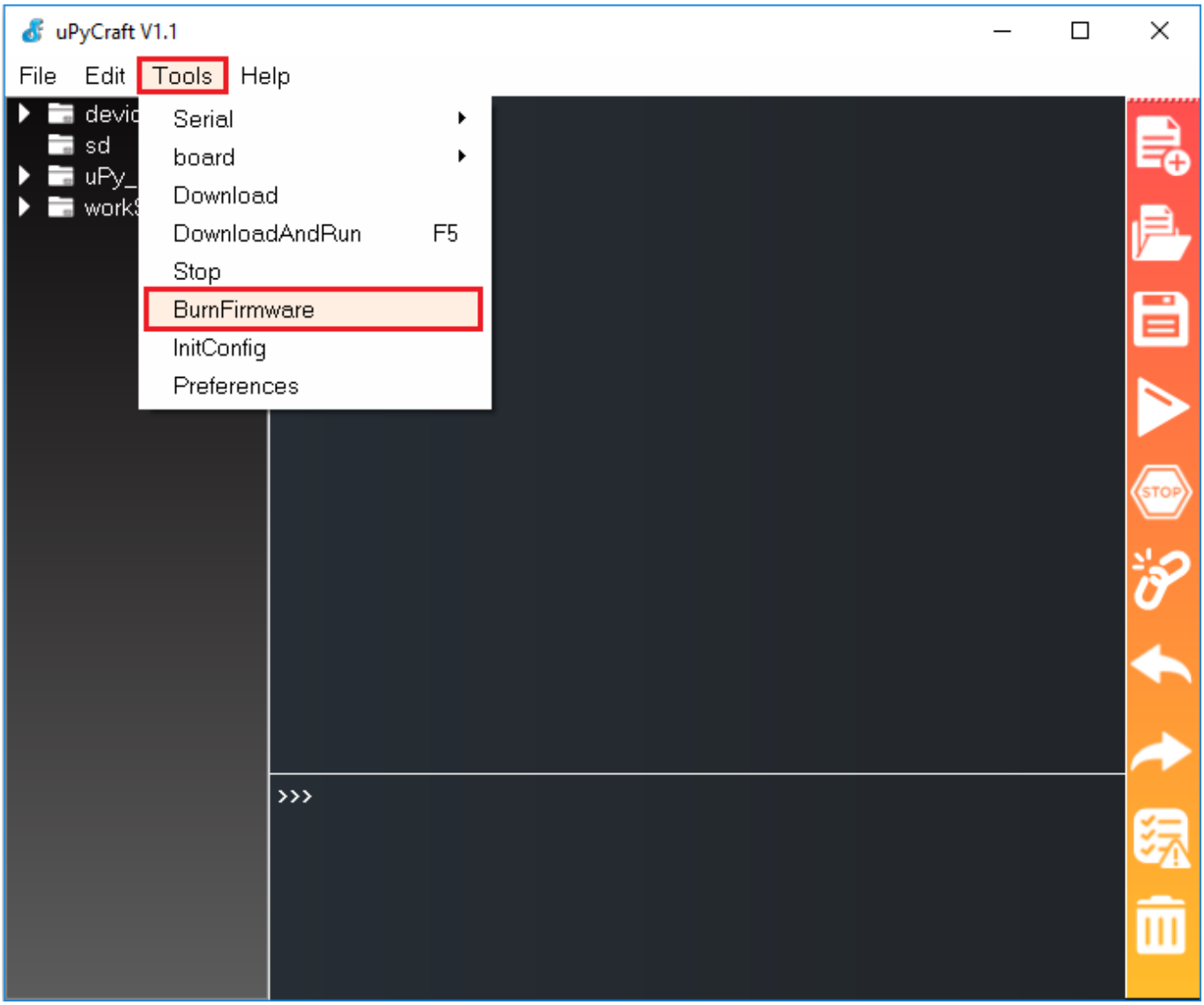
Selecting the Board

Go to **Tools** > **Board**. For this tutorial, we assume that you're using the ESP32, so make sure you select the "**esp32**" option:



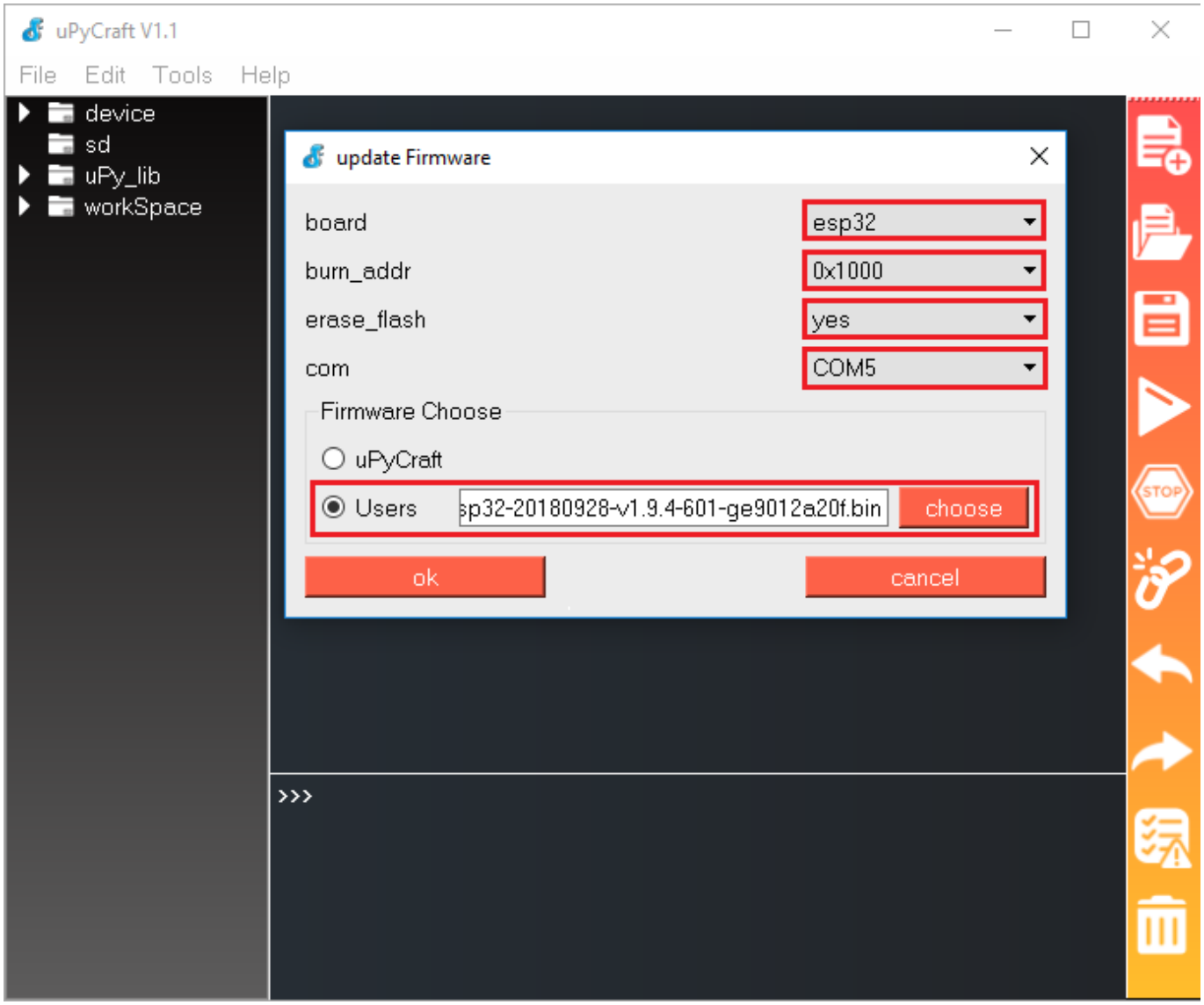
Flashing/Uploading MicroPython Firmware

Finally, go to **Tools** > **BurnFirmware** menu to flash your ESP32 with MicroPython.

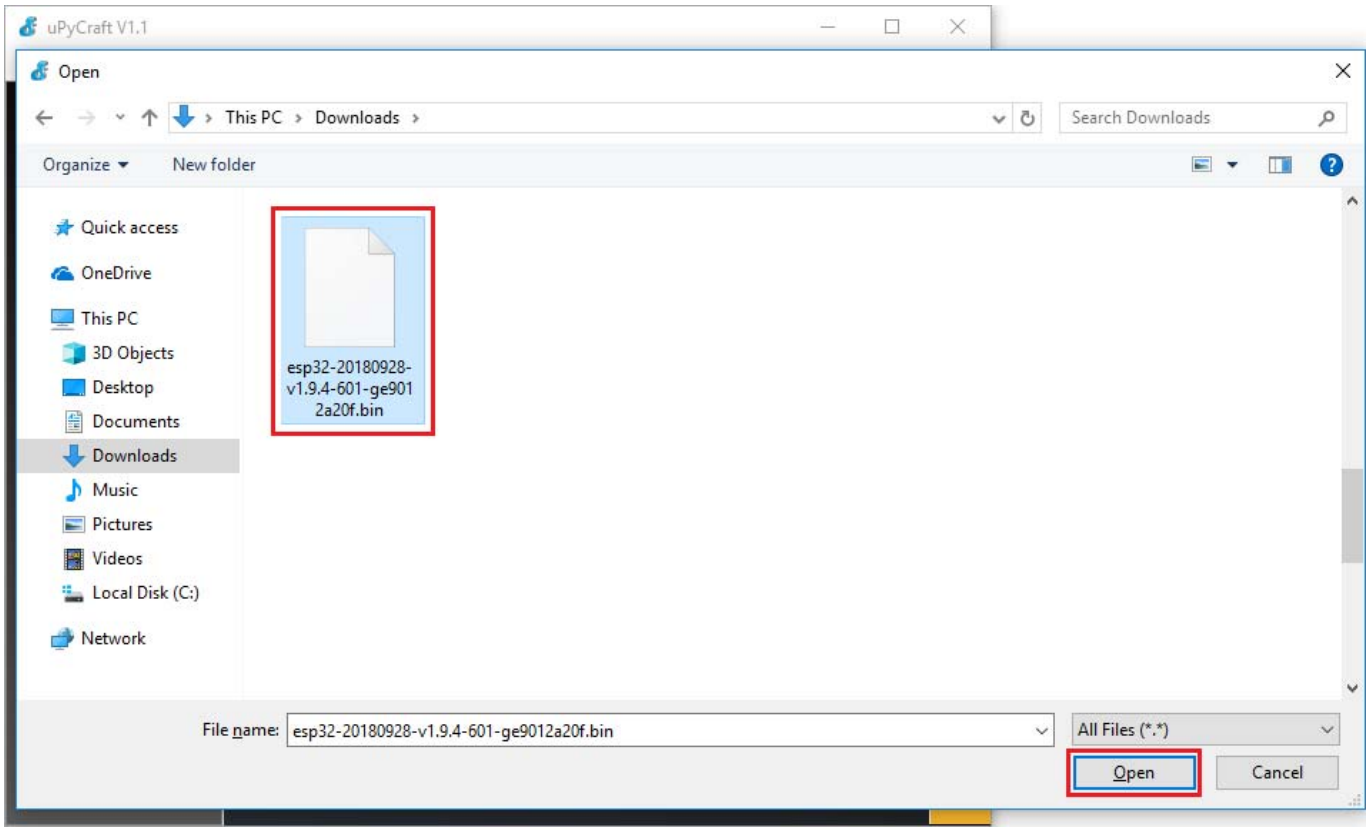


Select all these options to flash the ESP32 board:

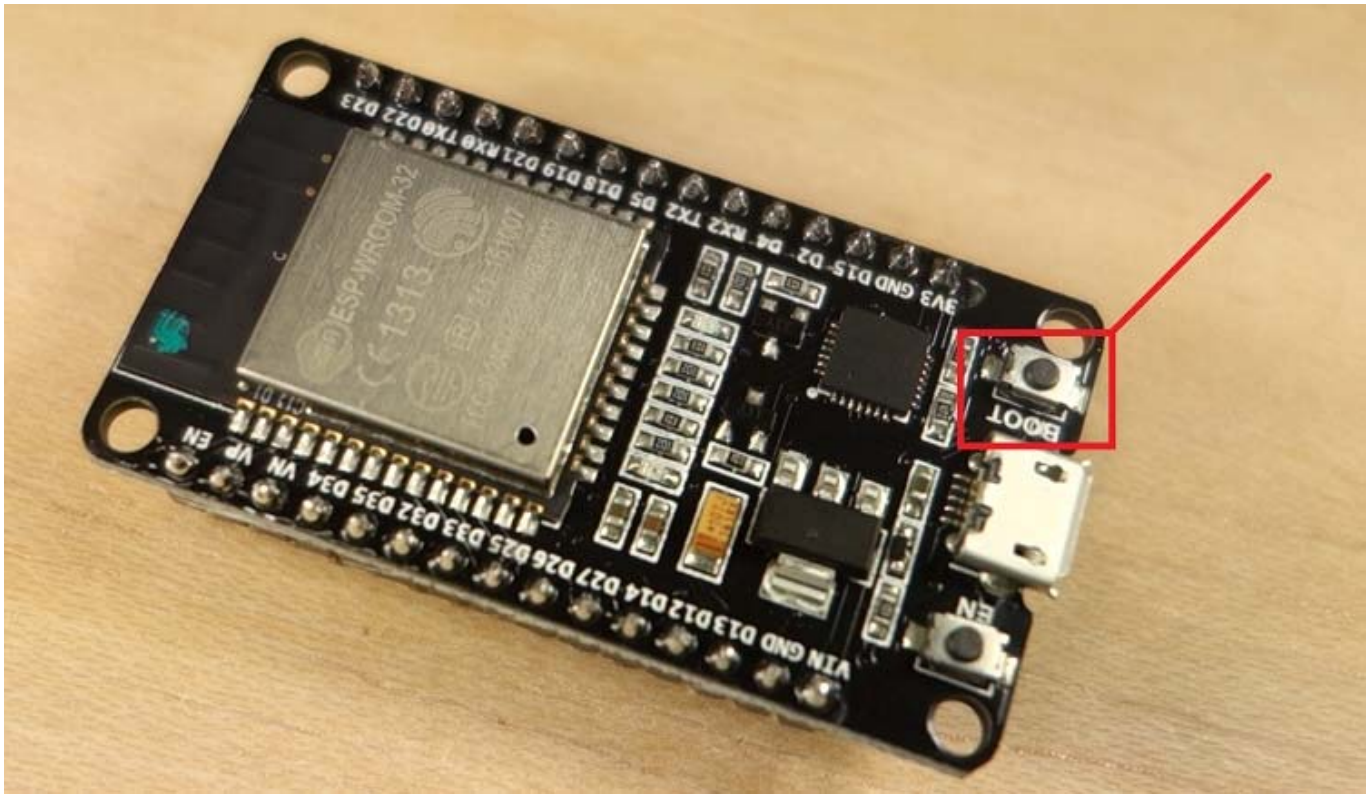
- board: **esp32**
- burn_addr: **0x1000**
- erase_flash: **yes**
- com: **COMX** (in our case it's COM5)
- Firmware: Select "**Users**" and choose the **ESP32 *.bin*** file downloaded earlier



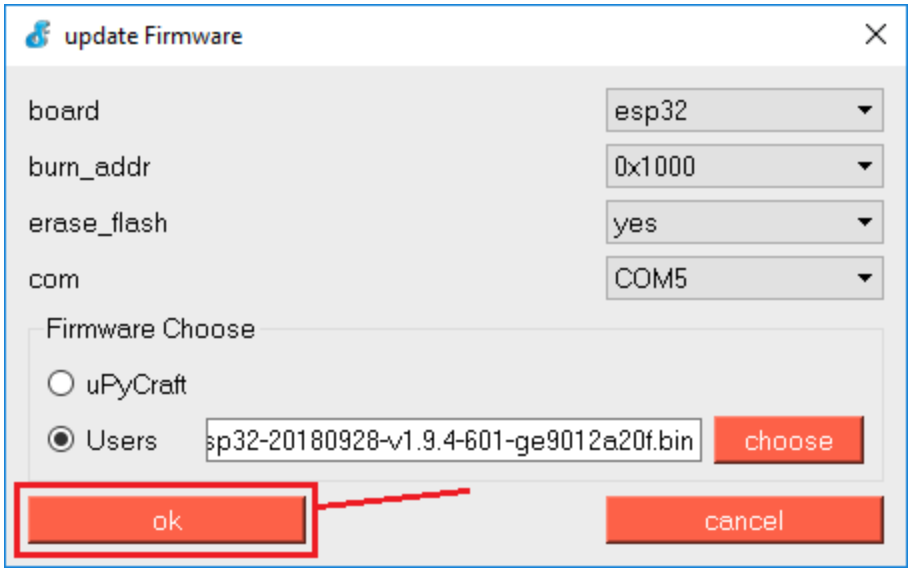
After pressing the "**Choose**" button, navigate to your Downloads folder and select the ESP32 *.bin* file:



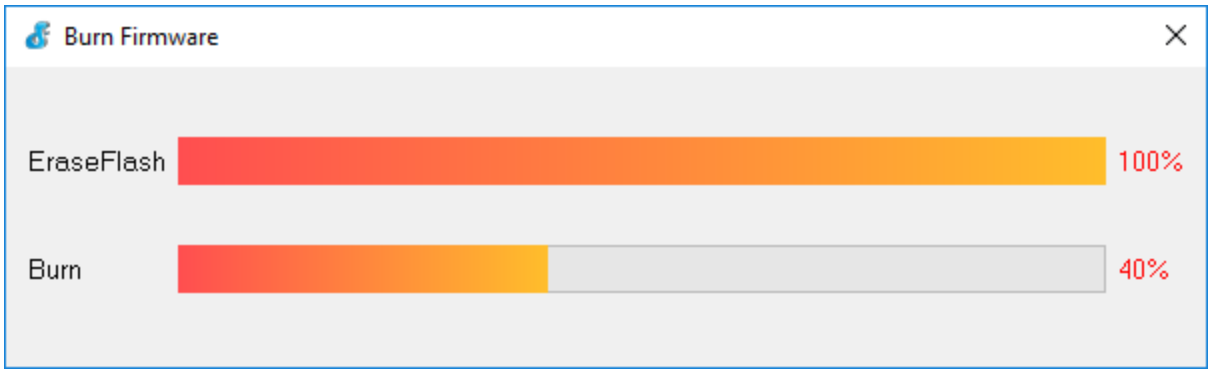
Having all the settings selected, hold-down the "**BOOT/FLASH**" button in your ESP32 board:



While holding down the “**BOOT/FLASH**“, click the “**ok**” button in the burn firmware window:



When the “**EraseFlash**” process begins, you can release the “**BOOT/FLASH**” button. After a few seconds, the firmware will be flashed into your ESP32 board.

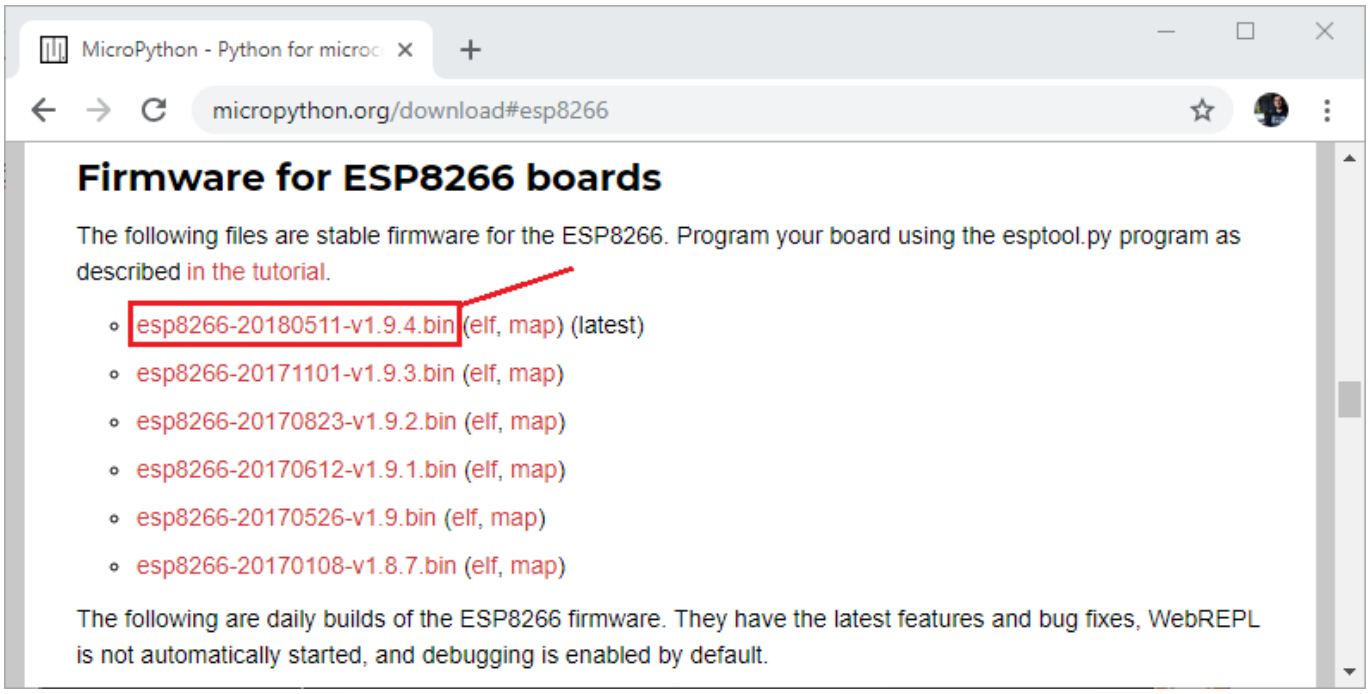


Note: if the “**EraseFlash**” bar doesn’t move and you see an error message saying “**erase false.**“, it means that your ESP32 wasn’t in flashing mode. You need to repeat all the steps described earlier and hold the “**BOOT/FLASH**” button again to ensure that your ESP32 goes into flashing mode.

[Part 2 – ESP8266] Downloading and Flashing the MicroPython Firmware on ESP8266

To download the latest version of MicroPython firmware for the ESP8266, go to the MicroPython Downloads page and scroll all the way down to the ESP8266 section.

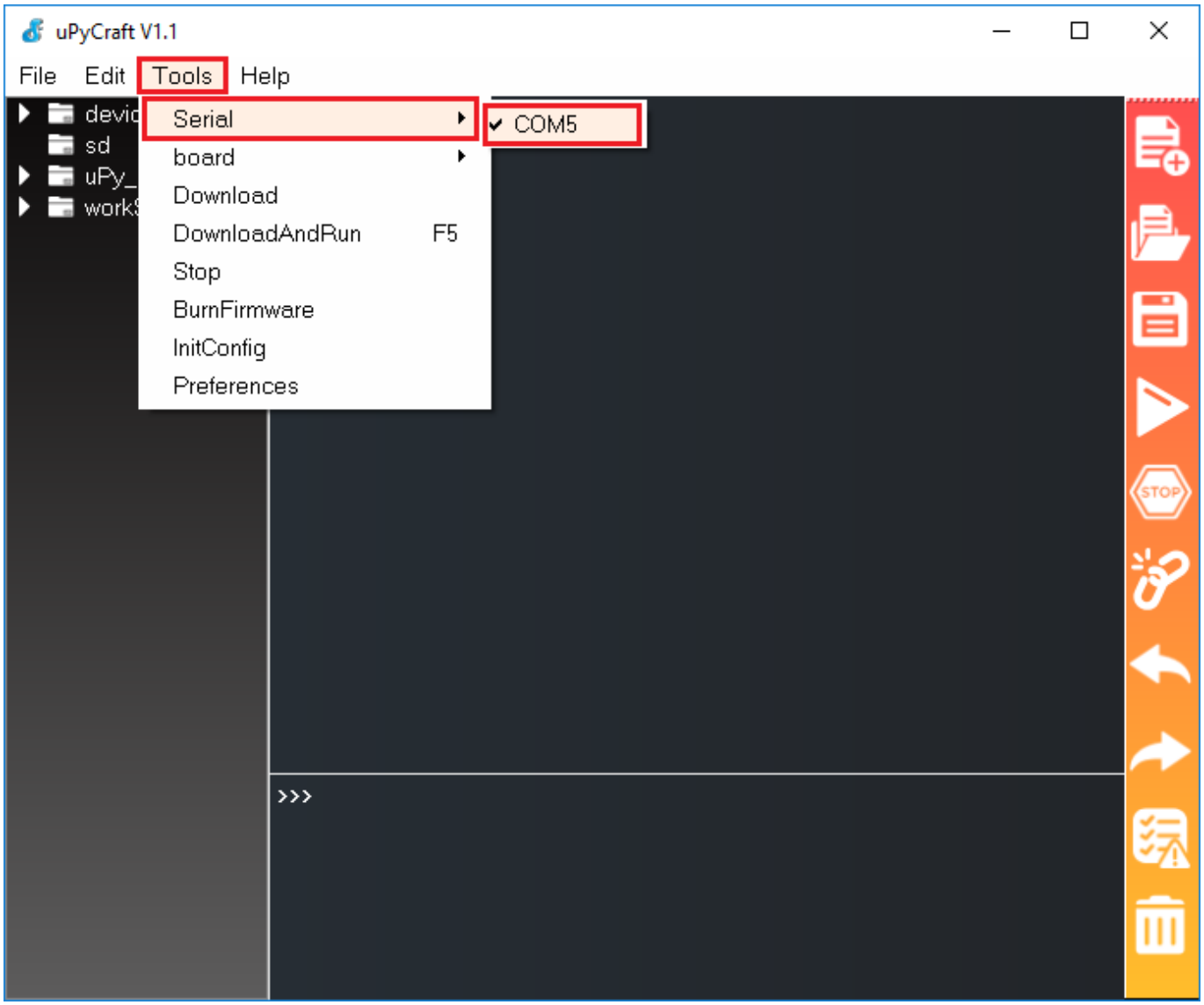
You should see a similar web page (see figure below) with the latest link to download the ESP8266 *.bin* file – for example: *esp8266-20180511-v1.9.4.bin*.



Note: if you're using a different board (like a PyBoard, WiPy, or other), go to MicroPython Downloads page and download the right firmware for your board.

Selecting Serial Port

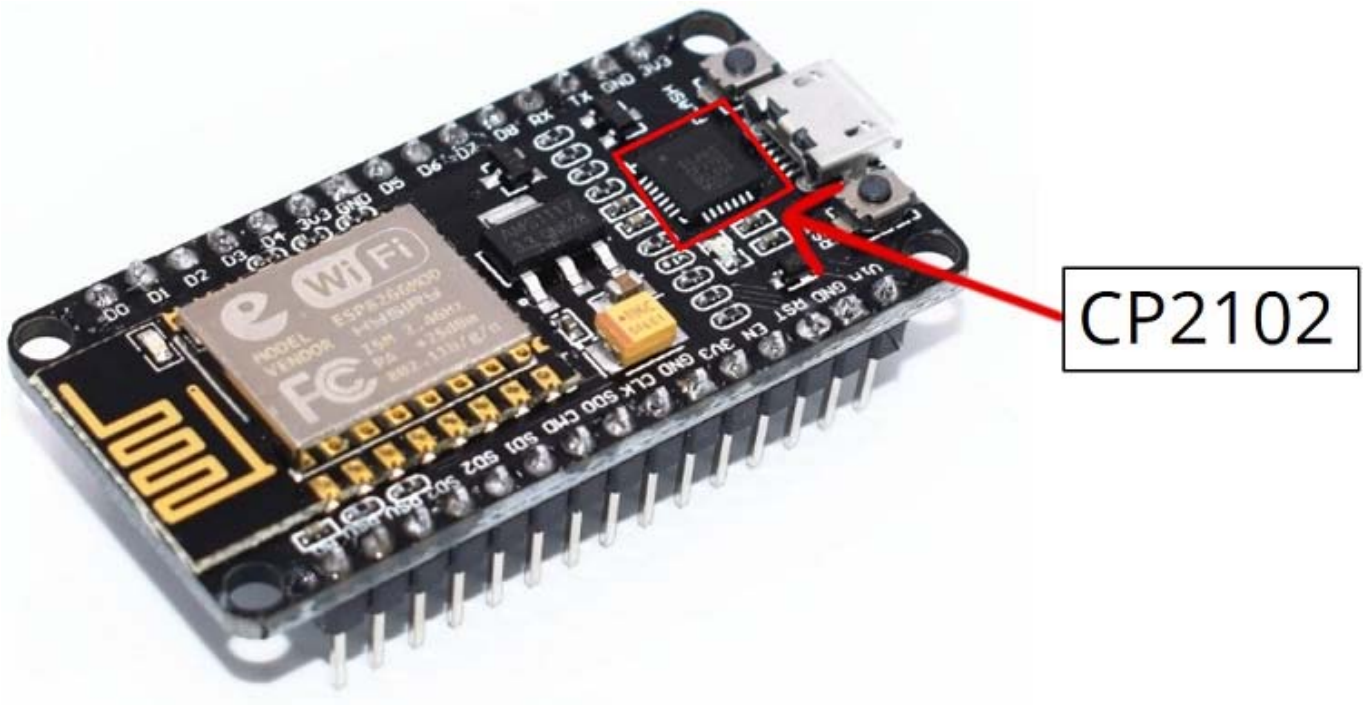
Go to **Tools** > **Serial** and select your ESP8266 COM port (in our case it's COM5).



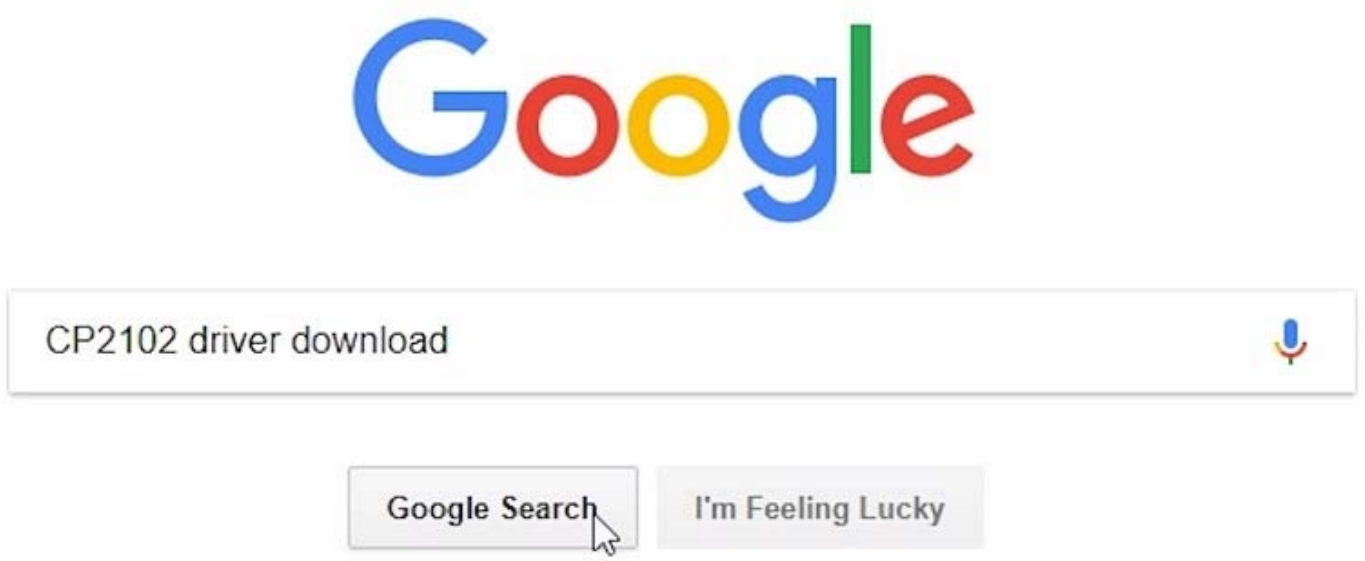
Important: if you plug your ESP32 board to your computer, but you can't find the ESP32 Port available in your uPyCraft IDE, it might be one of these two problems: **1.** USB drivers missing or **2.** USB cable without data wires.

1. If you don't see your ESP's COM port available, this often means you don't have the USB drivers installed. Take a closer look at the chip next to the voltage regulator on board and check its name.

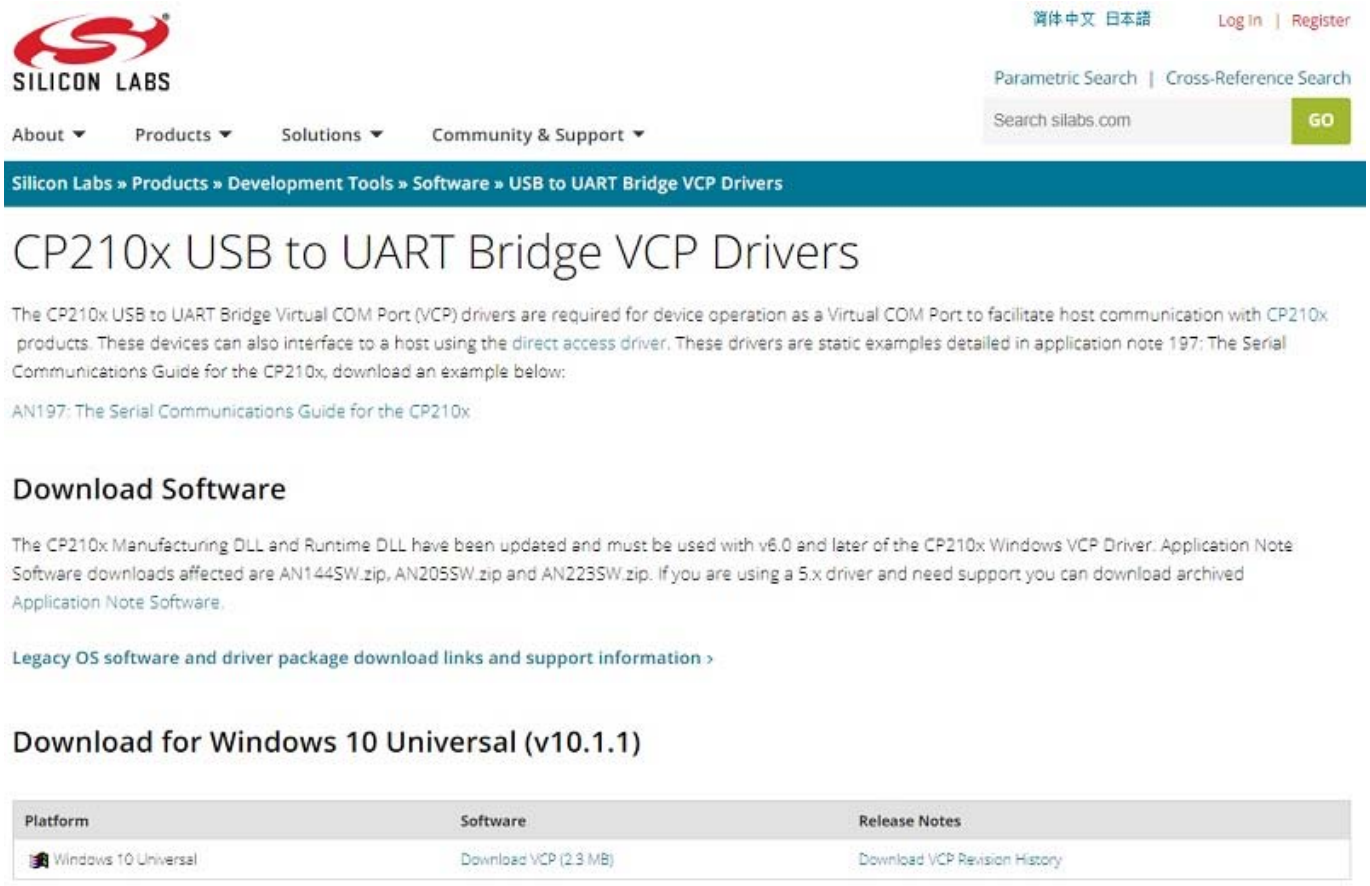
The ESP8266 ESP-12E NodeMCU board uses the **CP2102** chip.



Go to Google and search for your specific chip to find the drivers and install them in your operating system.



You can download the CP2102 drivers on the Silicon Labs website.



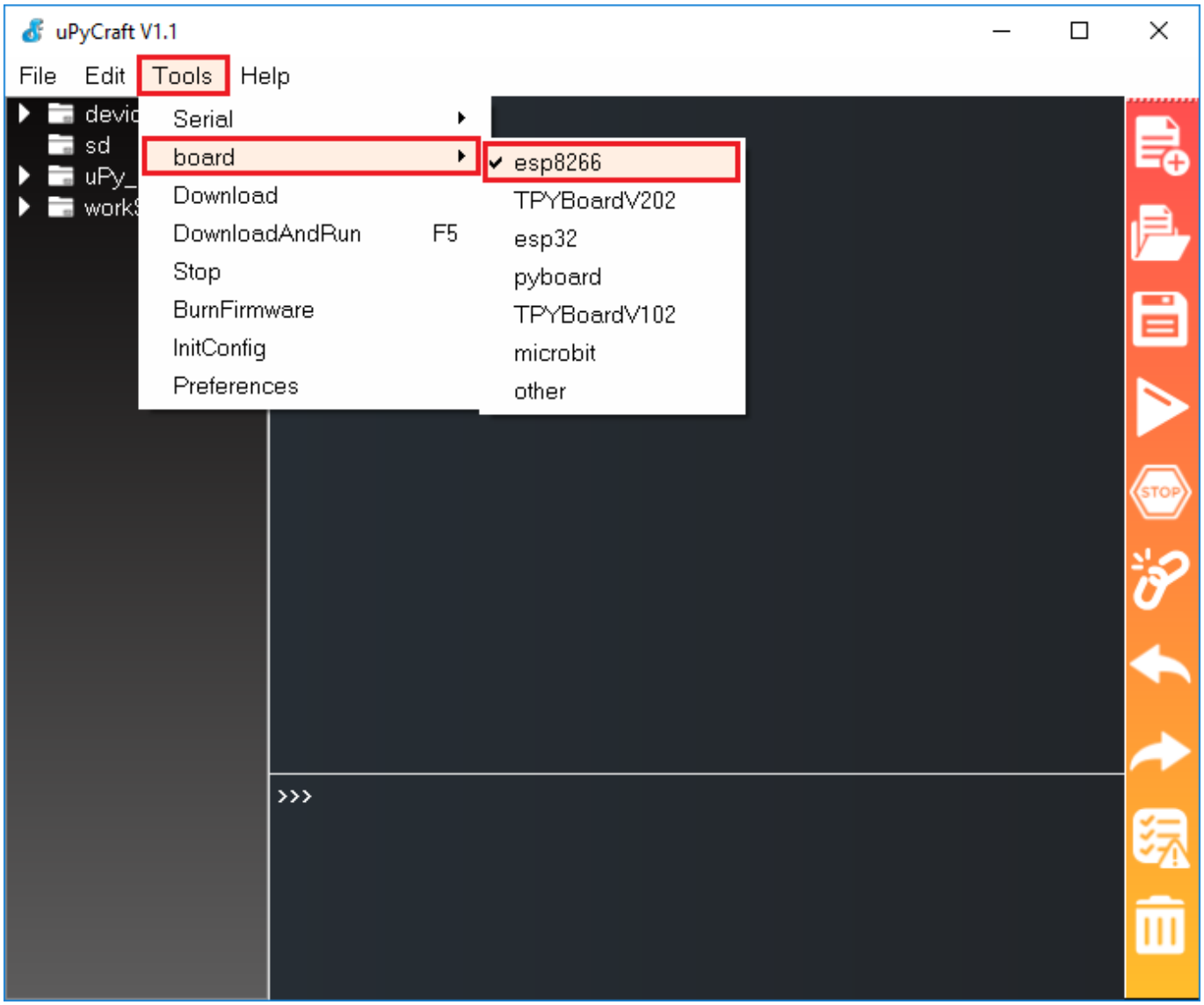
After they are installed, restart the uPyCraft IDE and you should see the COM port in the **Tools** menu.

2. If you have the drivers installed, but you can't see your device, double-check that you're using a USB cable with data wires.

USB cables from powerbanks often don't have data wires (they are charge only). So, your computer will never establish a serial communication with your ESP8266. Using a a proper USB cable should solve your problem.

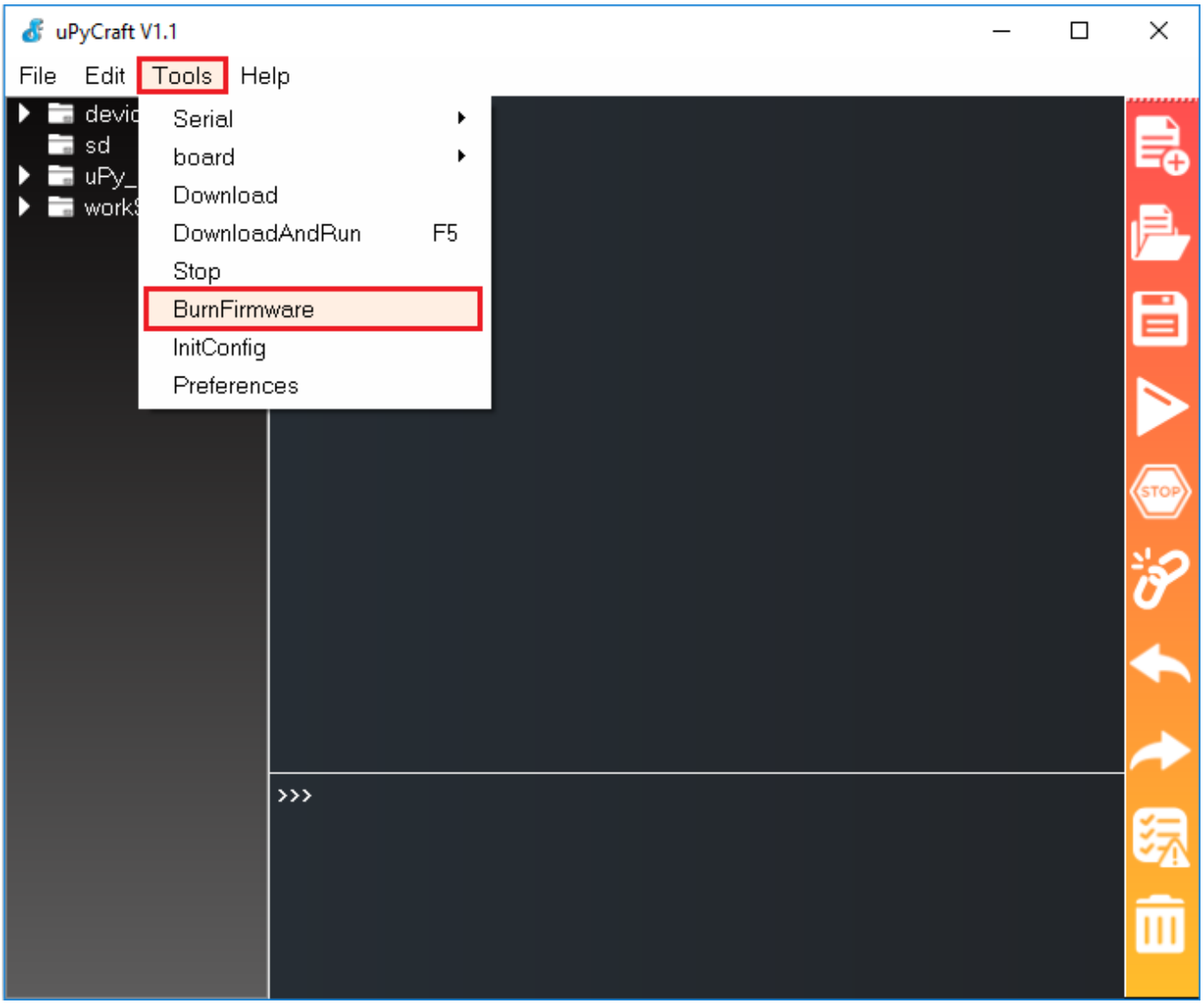
Selecting the Board

Go to **Tools** > **Board**. For this tutorial, we assume that you're using the ESP8266, so make sure you select the "**esp8266**" option:



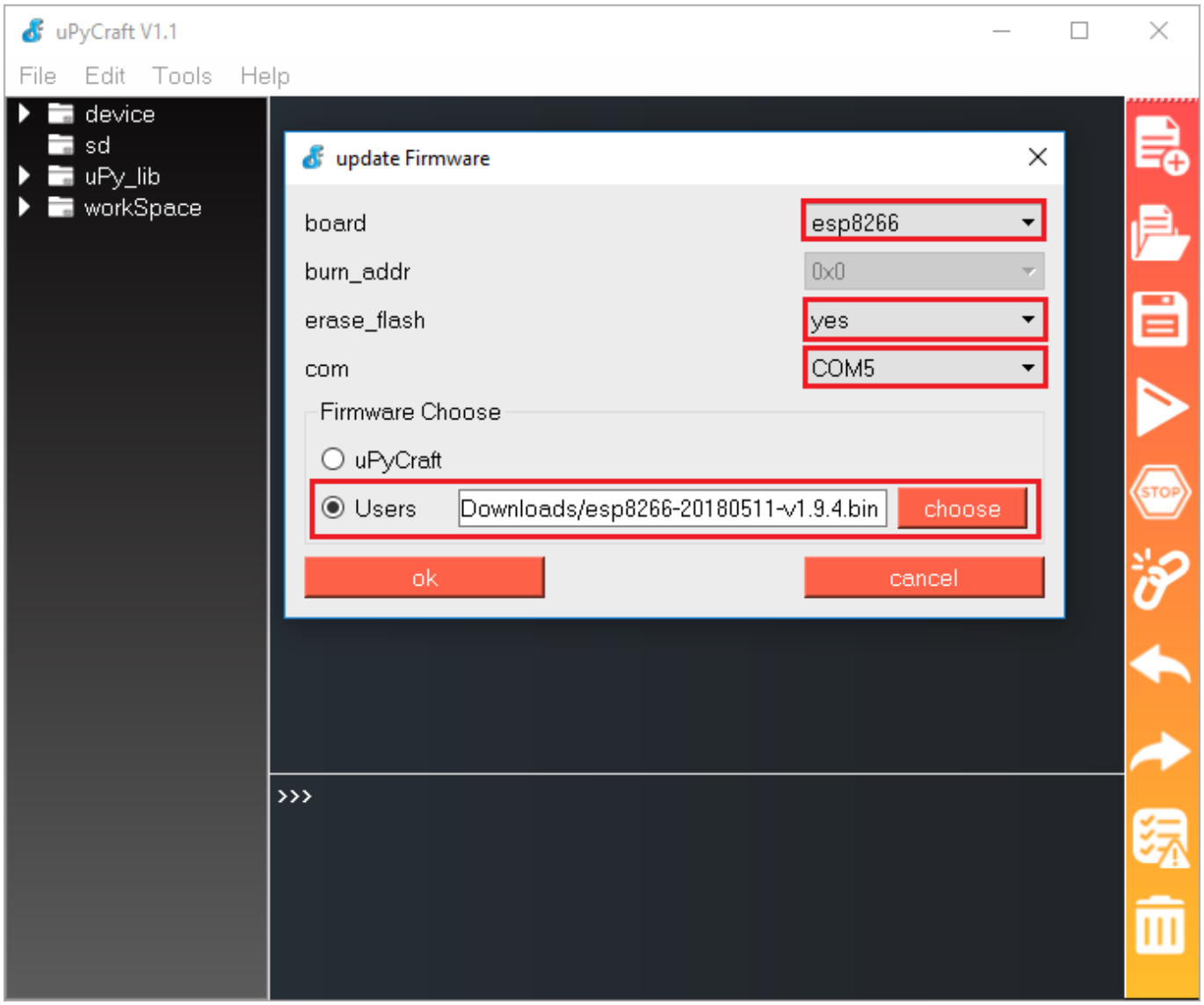
Flashing/Uploading MicroPython Firmware

Finally, go to **Tools** > **BurnFirmware** menu to flash your ESP32 with MicroPython.

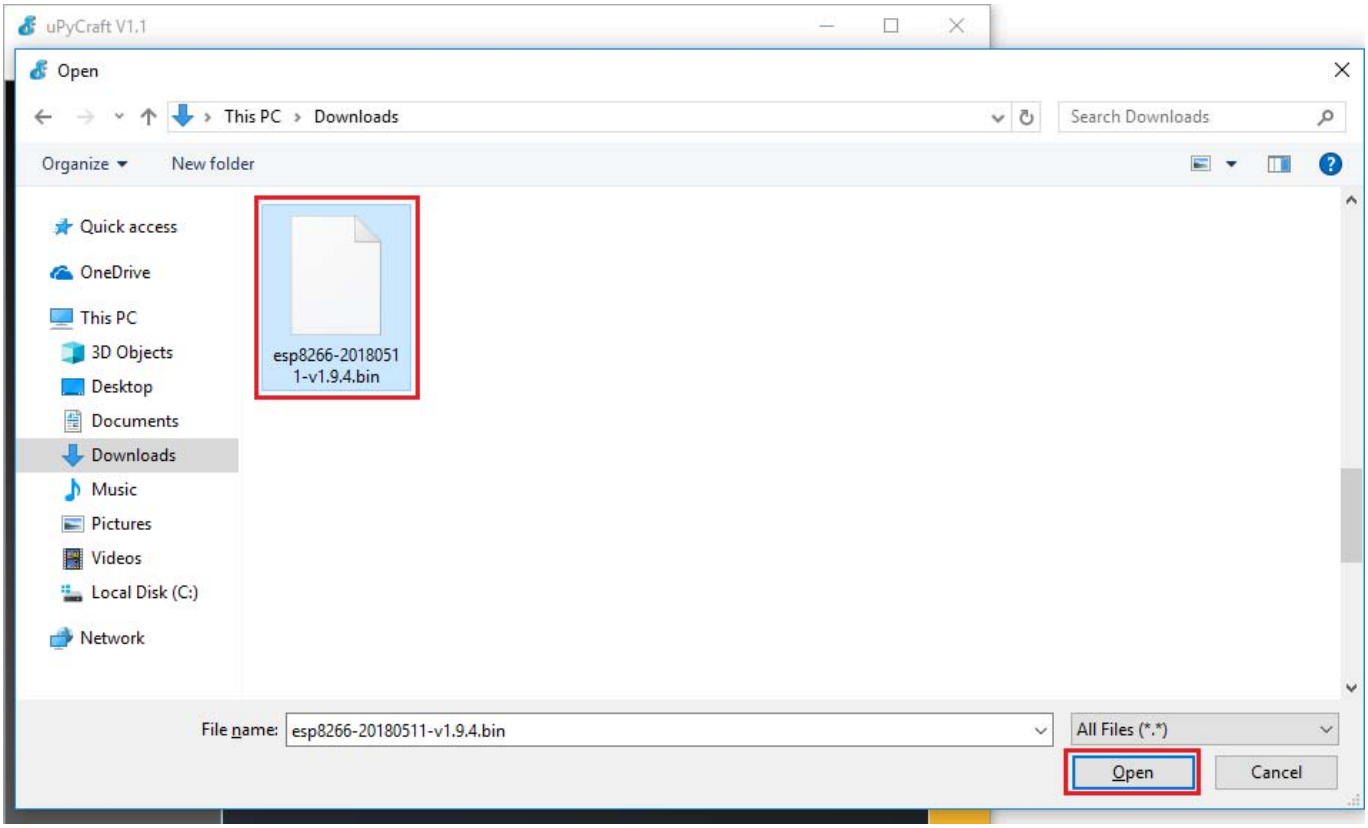


Select all these options to flash the ESP8266 board:

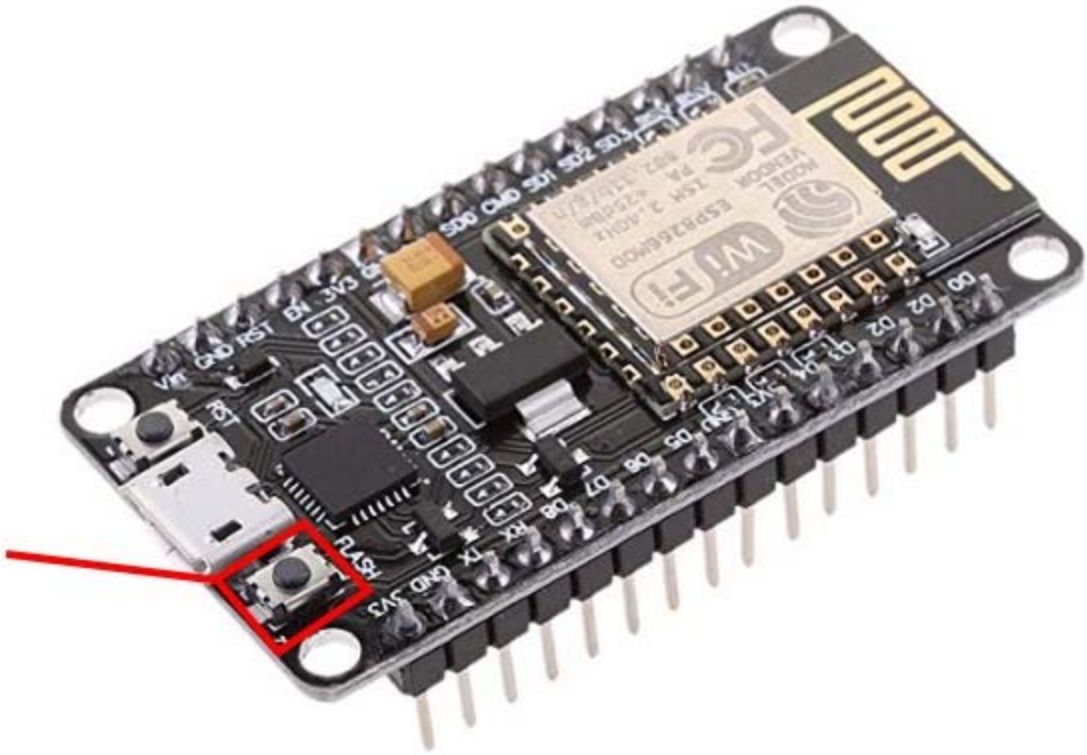
- board: **esp8266**
- burn_addr: **0x0**
- erase_flash: **yes**
- com: **COMX** (in our case it's COM5)
- Firmware: Select "**Users**" and choose the **ESP8266** ***.bin*** file downloaded earlier



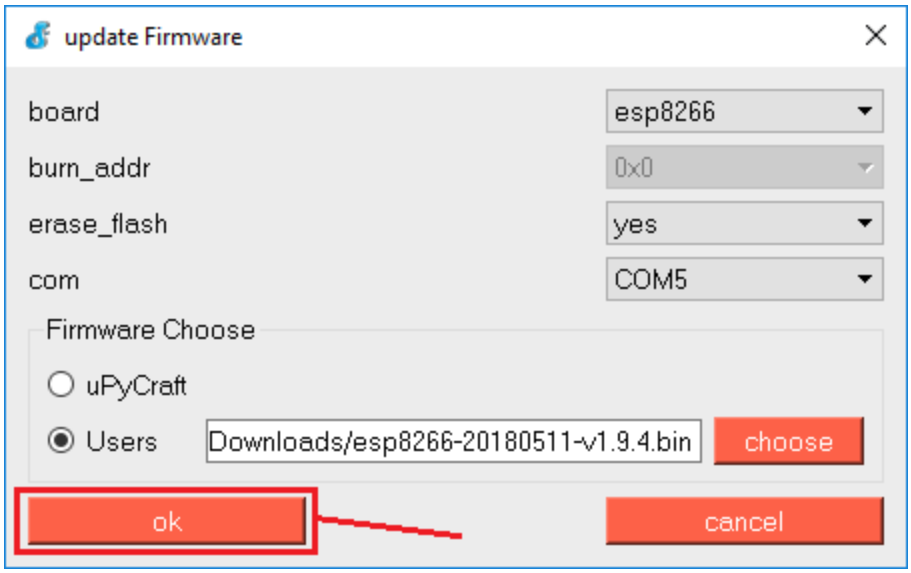
After pressing the "**Choose**" button, navigate to your Downloads folder and select the ESP8266 *.bin* file:



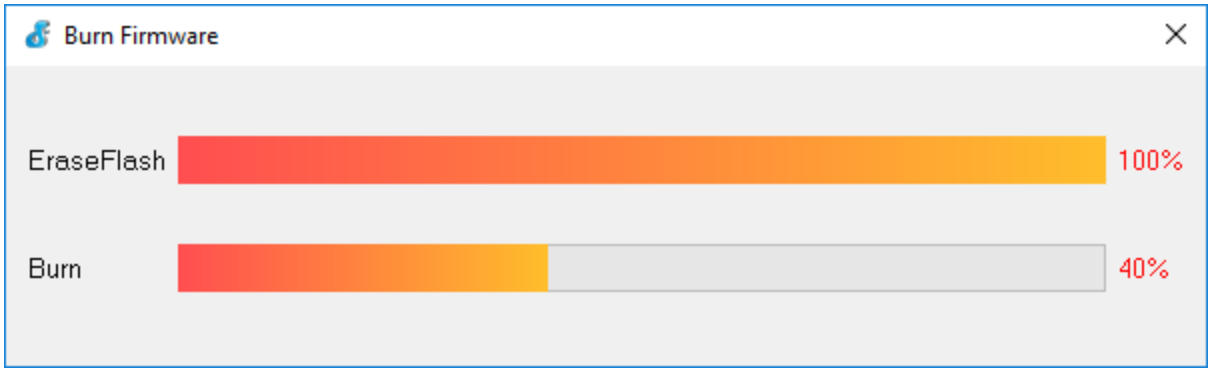
Having all the settings selected, hold-down the "**BOOT/FLASH**" button in your ESP8266 board:



While holding down the "**BOOT/FLASH**", click the "**ok**" button in the burn firmware window:



When the "**EraseFlash**" process begins, you can release the "**BOOT/FLASH**" button. After a few seconds, the firmware will be flashed into your ESP8266 board.



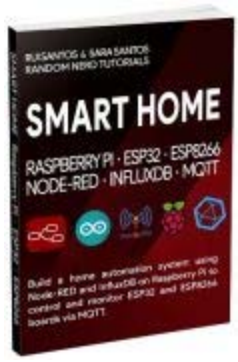
Note: if the "**EraseFlash**" bar doesn't move and you see an error message saying "**erase false.**", it means that your ESP8266 wasn't in flashing mode. You need to repeat all the steps described earlier and hold the "**BOOT/FLASH**" button again to ensure that your ESP8266 goes into flashing mode.

Wrapping Up

We hope you've found this tutorial useful. Your ESP32/ESP8266 should now be flashed with MicroPython firmware. To learn more about MicroPython read: Getting

Started with MicroPython on ESP32 and ESP8266.

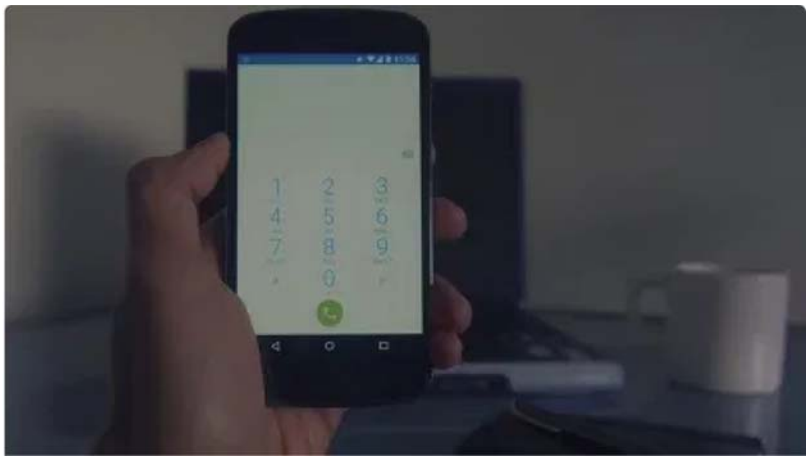
If you liked this post, you might like our next ones, so make sure you subscribe to the RNT blog and download our free electronics eBooks.



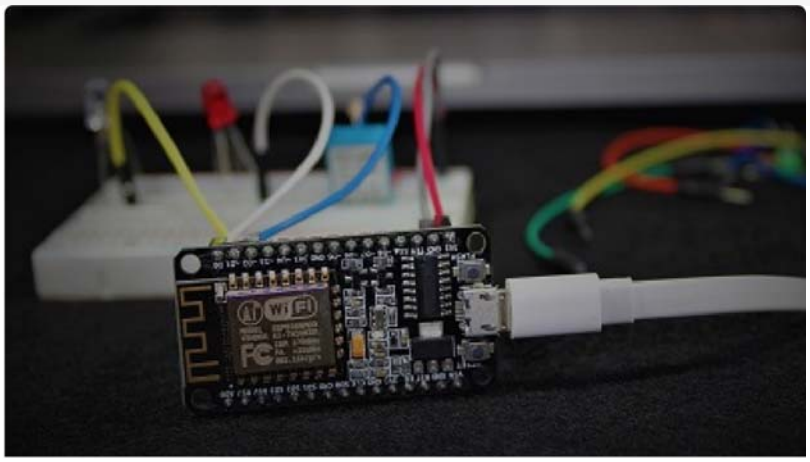
SMART HOME with Raspberry Pi, ESP32, ESP8266 [eBook]

Learn how to build a home automation system and we'll cover the following main subjects: Node-RED, Node-RED Dashboard, Raspberry Pi, ESP32, ESP8266, MQTT, and InfluxDB database **DOWNLOAD »**

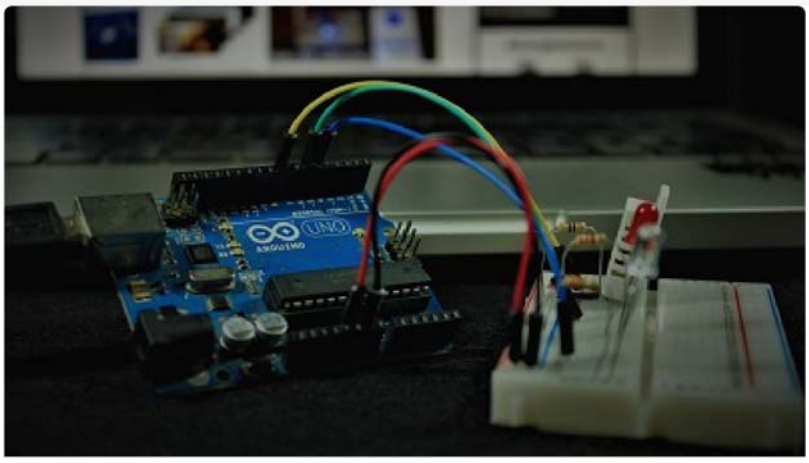
Recommended Resources



Build a Home Automation System from Scratch » With Raspberry Pi, ESP8266, Arduino, and Node-RED.



Home Automation using ESP8266 eBook and video course » Build IoT and home automation projects.



Arduino Step-by-Step Projects » Build 25 Arduino projects with our course, even with no prior experience!

What to Read Next…

Build an ESP8266 Web Server – Code and Schematics (NodeMCU)

ESP8266 NodeMCU Door Status Monitor with Email Notifications (IFTTT)

ESP32/ESP8266: Firebase Data Logging Web App (Gauges, Charts, and Table)

45 thoughts on “Flash/Upload MicroPython Firmware to ESP32 and ESP8266”

Jelle Aarnoudse

October 16, 2018 at 6:03 pm

Nice tutorial! One question: after flashing the Micropython firmware to the ESP32, can I still use the ESP32 with the Arduino-IDE?

Reply

Sara Santos

October 17, 2018 at 8:36 am

Hi Jelle.
Yes. You just need to upload code using Arduino IDE again.
Regards,
Sara 🙂

Reply

Andrew Wilson

August 2, 2020 at 3:02 pm

One thing worth bearing in mind is that Micropython will need to be re-flashed again onced used with arduino. The ESP32 is now so cheap it may be worth having one/some with micropython, and use others with arduino.

Reply

viswanatha v

September 14, 2022 at 1:58 pm

Good question Even i had same question in my mind.thanks

Reply

Chris Parsons

October 17, 2018 at 8:28 am

If I flash an ESP32 or ESP8266 board with MicroPython is this permanent?
Or can I go back to using the Aduino IDE and code?

Reply

Sara Santos

October 17, 2018 at 8:35 am

Hi Chris.
You can go back using Arduino IDE.
You just need to upload a code using Arduino IDE.
REgards,
Sara 🙂

Reply

Tommy

October 15, 2020 at 12:08 pm

Thanks!

Reply

viswanatha v

September 14, 2022 at 2:00 pm

Thank you for the reply .Your books are very informative and easy to understand. Thank you so much

Reply

Sara Santos

September 15, 2022 at 9:59 am

Thanks 🙂

Reply

Shehu Bello

October 18, 2018 at 3:29 pm

Very interesting and helpful post! But I have this error message any time I tried to upload a script:

download false
exec(open('main.py').read(),globals())
Traceback (most recent call last):

File “”, line 1, in
OSError: [Errno 1] EPERM

Reply

Sara Santos
October 19, 2018 at 1:44 pm

Hi.
Try to reconnect the board to uPyCraft IDE or click the STOP button before uploading a new script.
I think that error means that your ESP is not ready to receive a new script.

Note: make sure your file name is correct (boot.py/main.py) and that your script indentation is also correct (no spaces for the first instructions and two spaces for indentation).
Finally, it looks like the firmware was flashed correctly. But in case you keep having problems, we recommend reflashing the firmware.
I hope this helps.
Let me know how it went.
Regards,
Sara 🙂

Reply

Shehu Bello
October 20, 2018 at 10:50 am

Hi Sara,
I flashed the firmware several time and my script is no having any syntax

error. But I observed that there is no boot.py file in the workSpace folder. Could that be the problem? If yes, how do I go about it? Thanks a lot

Reply

Sara Santos

October 21, 2018 at 3:48 pm

Hi.
You need to have the boot.py file in the *device* folder. Do you have that file? As shown in this image?
https://i0.wp.com/randomnerdtutorials.com/wp-content/uploads/2018/10/main-py-file-created.png?w=744&ssl=1
Have you tried to follow these tutorial: https://randomnerdtutorials.com/getting-started-micropython-esp32-esp8266/ and uploading the blinkind LED sketch?

Reply

Shehu Bello

October 23, 2018 at 5:46 am

Hi Sara.
The device folder does not open and it keeps showing error message as follows:
current version only open py txt json ini file.
In fact, apart from workspace, all the folders do not open.
Thanks

Sara Santos

October 23, 2018 at 8:27 am

Hi again.
That's very weird, it means it can't open the files or they don't exist. I have no idea why that is happening and I can't replicate that error.
Can you reflash the firmware again? Don't forget to check the "Erase Flash" option.
If that doesn't work, upload a blank sketch in Arduino IDE, and then, reflash MicroPython again.
If none of those options work, your ESP may be malfunctioning for some reason.
Regards,
Sara

Maurizio Cozzetto

November 2, 2018 at 1:31 pm

Hi Ruis! I flashed my ESP32 board with Micropython and all seemed to work. After turning the PC off and then on again, I've found my ESP32 board contained nothing. I've repeated the process many times according your instructions, but whenever the situation was the same, the board did not contain anything. Do I have to throw the card? Many thanks.

Reply

Sara Santos

November 2, 2018 at 2:35 pm

Hi Maurizio.
After flashing the ESP with MicroPython, when you select the Serial port, do you see a boot.py file under the device folder?
https://i0.wp.com/randomnerdtutorials.com/wp-content/uploads/2018/10/main-py-file-created.png?w=744&ssl=1

Reply

Shukri
November 3, 2018 at 2:14 pm

Hi,
Anybody try to flash Yunshan wifi relay with Micropython firmware
ucexperiment.wordpress.com/2016/12/18/yunshan-esp8266-250v-15a-acdc-network-wifi-relay-module/
Really appreciate your help, thanks in advance

Reply

Sara Santos
November 4, 2018 at 11:33 am

Hi Shukri. I've never experimented with Yunshan wifi relay. It uses an ESP8266, so it should be possible to flash micropython.
Probably you need to follow the instructions on that page to connect the board to your computer using TTL, and then follow the steps to upload firmware:
"To program the ESP8266 module, I connected the TX, RX and ground pins

of connector P6 to a SparkFun USB FTDI programmer, and jumped the two pins of connector P5 together when I was ready to upload. Connector P5 grounds GPIO0 and GPIO15, sending the device into bootloader mode. If you have trouble programming the ESP8266 like I did on the first attempt, ensure you also ground your FTDI device through the P6 connector."

Reply

Paul C Johnson

January 6, 2019 at 3:11 am

I have been trying to install micropython over and over getting nowhere. I just happened to see you site after a google. I was able to load it onto adafruit HUZZAH feather and two esp32 dev kit v1. Your soft is great. Thank You Very Much

Reply

Sara Santos

January 8, 2019 at 5:14 pm

Thanks 🙂

Reply

Ong Kheok Chin

January 30, 2019 at 1:49 am

The ESP8266 can be flash successfully but the ESP32 is not able to flash even though the display indicates that EraseFalsh is 100% and Burn is 100%. DO you know what happen ?

Reply

Sara Santos

February 5, 2019 at 12:38 pm

Hi.
Can you provide more information to better understand what is going on?
Regards,
Sara

Reply

Roma

April 15, 2019 at 1:05 pm

Under windows I can upload and run only one time, my program, but in next time I only see error on uPyCraft – open serial error. Please help.

Reply

Sara Santos

April 16, 2019 at 2:56 pm

Hi.
Please refer to this article https://randomnerdtutorials.com/getting-started-micropython-esp32-esp8266/ scroll down to the “troubleshooting tips” section, error #1 at the end of the post. You should see how to solve that problem.
Alternatively, I recommend using Thonny IDE. It is another IDE but more intuitive to use and you won’t have those kind of problems. See our tutorial here: https://randomnerdtutorials.com/getting-started-thonny-micropython-python-ide-esp32-esp8266/
I hope this helps.
Regards,
Sara

Reply

S.Mary
May 2, 2019 at 5:24 pm

Hi
I’m french and i don’t speak very well english
i have a problem with my microbit and uPycraft…

my program

```
from microbit import *
from machine import time_pulse_us
pin16.write_digital(1)
sleep(0.010)
pin16.write_digital(0)
t = time_pulse_us(pin12, 1)
h=round((340*t*1e-6/2)*100,1)
```

print(“Hauteur:”,h,”cm”)
V=round(24.8*15*(16.6-h)/1000,2)
print (“le volume de fuel restant est :”, V, “L”)
UV=round(V*(((0.2/t)**2+(4/340)**2)**0.5),2)
print(“l’incertitude sur le volume de fuel restant :”,UV,”L”)

error
>>>
Ready to download this file,please wait!
…….
download ok
exec(open(‘Cuve complet.py’).read(),globals())
Traceback (most recent call last):
File “”, line 1, in
File “”, line 2, in
ImportError: no module named ‘machine’
>>>

Can you help me …
Thanks
S.Mary

Reply

Sara Santos
May 2, 2019 at 9:32 pm

Hi Mary.
We don’t have tutorials using microbit and I’m not familiar with it.
But, it seems that the micropython firmware in your microbit doesn’t contain the machine module.
Regards,
Sara

Reply

M2M

May 4, 2019 at 9:15 pm

Why uPycraft often ask me to burn a firmware even though ESP32 have already burned? I often lost my files in ESP32 if I burn it again. It often detects my ESP32 as a new device and ask to burn again.

Reply

Sara Santos

May 5, 2019 at 9:05 am

Hi.
I don’t know why that happens.
If that happens to you very often, I recommend using another IDE like Thonny IDE. Usually, it doesn’t give this kind of problems.
Here is our tutorial to Thonny IDE: https://randomnerdtutorials.com/getting-started-thonny-micropython-python-ide-esp32-esp8266/
Regards,
Sara

Reply

Joseph Jakpa

August 22, 2019 at 9:54 am

Can micropython be burnt on the esp-01? get update false error.

Reply

Sara Santos

August 22, 2019 at 10:07 am

Yes.
But you need to flash the ESP8266 firmware tailored for modules with only 512kbytes of flash. See here: https://micropython.org/download
Also, the ESP8266 needs to be in flashing mode so that you can upload the firmware.
Regards,
Sara

Reply

Joseph Jakpa

August 22, 2019 at 10:11 am

Hi Sara. in fact i have the esp-01s version with 1mb of flash and gpio 0 connected to gnd But i'm still getting the update false error and sometimes erase false. It could be bad jumper cables or something else so i'll keep debugging until i fix it. Thanks.

Reply

Sara Santos

August 24, 2019 at 11:33 am

Hi.
alternatively, you can try flashing firmware using esptool.py
https://randomnerdtutorials.com/flashing-micropython-firmware-esptool-py-esp32-esp8266/
Regards,
Sara

Reply

Mihael

September 11, 2019 at 10:11 pm

Hi Sara,
I flashed my esp-01 with firmware for 512k, and flash goes ok, but after restart I try to set uPyCraft to COM Port and then comes Burn Firmware Window – the app doesn't recognise my device. What's wrong with it?

Reply

Sara Santos

September 13, 2019 at 9:55 am

Hi.
Is is probably something with the connections between your esp-01 and

your computer?
It may also be an issue with uPyCraft IDE.
Try taking a look at Thonny IDE: https://randomnerdtutorials.com/getting-started-thonny-micropython-python-ide-esp32-esp8266/
Regards,
Sara

Reply

Friedhelm
February 17, 2020 at 8:30 am

Hi Sara,
MicroPython Downloads page shows no stable firmware for ESP32!
What can I do?

Reply

Patrick Fitzgerald
April 25, 2020 at 12:06 pm

This is great! Following this tutorial I used uPyCraft on windows with a TTGO T-Display

1 burn Firmware with uPyCraft choosing gave me a MicroPython OS on the device that worked with both uPyCraft IDE and with Mu IDE and provided REPL interface

2 LilyGO also has a bin file at https://github.com/Xinyuan-LilyGO/TTGO-T-Display. Flashing this with uPyCraft returned to the demo software that was on the device when I bought it

3 Arduino IDE still worked. I was able to load a demo program I had put together based on the examples that loaded when I selected ESP32 Dev Module with the Arduino Boards Manager

Reply

Philip Hart

September 9, 2020 at 11:26 am

Great tutorial – thanks. I'm fairly up to speed with Arduino, but now my Year 9 son is just starting to learn Python, so this has helped me to help him get into the embedded world. We have our first microPython Blink programme working on a NodeMCU V3 – onwards and upwards! Thanks again.

Reply

WILLIAM E WEBB

December 12, 2020 at 9:10 pm

Great tutorial. I was generally successful in installing micro Python on an ESP8266 and ESP32. I installed both IDEs on by WIN10 machine and both work well. uPyCraft IDE makes erasing and flashing the OS onto the microcontrollers easy.

Now down to my question:

In both IDES and on both controllers, %lsdevice returns an an error.

MicroPython v1.13 on 2020-09-11; ESP module with ESP8266

Type “help()” for more information.

> > *%lsdevice*
> > *Unknown command: lsdevice*

Help() works fine, so I know that I am ‘talking’ Thoughts? fixes?

One other thing that may or maynot be abnormal boot.py on both controllers contains only one comment line. Is that normal?

Is there a list of commands, like lsdevice?

Reply

Sara Santos
December 15, 2020 at 4:33 pm

Hi.
Unfortunately, that command is not longer supported.
Regards,
Sara

Reply

Alan Knight

February 5, 2021 at 8:29 pm

I am unable to create a boot.py file. PC keeps asking me for a path to store the file.

Reply

Alan Knight

February 6, 2021 at 3:04 pm

After flashing the ESP32, there is no boot file in ‘device’. No error shows at end of the flash, but no positive msg either.

Please help.

Thanks

Reply

Gil Greenwood

February 18, 2021 at 3:41 pm

Trying to flash micropython to my ESP8266. Cleared the board using esptool.py erase_flash. Found my serial port as /dev/ttyUSB0. I then rshell to finish getting REPL. The program starts but then REPL searches and hangs. Is there something I am doing wrong? Ideas?

Reply

Riley

July 30, 2022 at 1:26 am

uPyCraft no longer works on MacOS Monterrey.

It requires PyQt4 (which is no longer available) and QScintilla which doesn't seem to work.

Reply

José Gómez

September 16, 2023 at 2:47 pm

Hello Sara,
I am José Gómez from Spain and I tying to use Deep Sleep with ESP8266 (d1 mini) with Micropython and it works for some cicles, but after a while stop working. Could you help me?
If yes I will send the modules that I am using.

Thanks in advance,
José

Reply

Leave a Comment

Name *

Email *

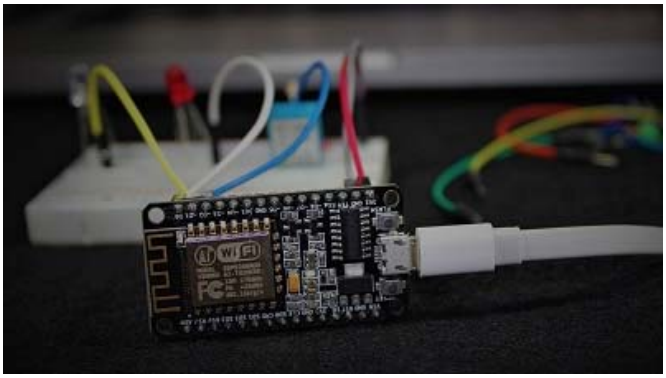
Website

Notify me of follow-up comments by email.

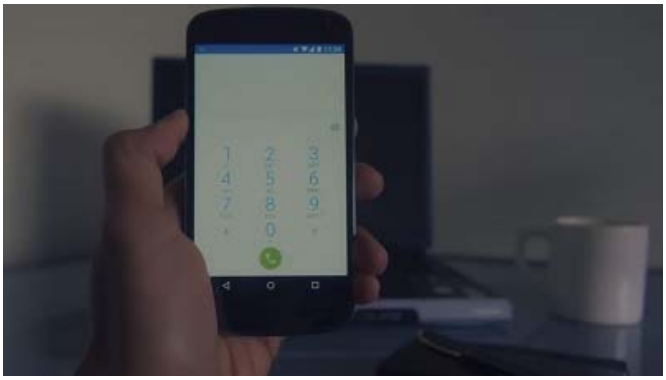
Notify me of new posts by email.

Post Comment

Visit Maker Advisor – Tools and Gear for makers, hobbyists and DIYers »



Home Automation using ESP8266 eBook and video course » Build IoT and home automation projects.



Build Web Servers with ESP32 and ESP8266 » boards to control outputs and monitor sensors remotely.