ESP32-C6-Zero

From Waveshare Wiki
Jump to: navigation, search

Overview

The ESP32-C6-Zero is a low-cost, high-performance microcontroller development board with a compact size and rich peripheral interfaces.
Adopts ESP32-C6FH4 as the main chip, with RISC-V 32-bit single-core processor, support up to 160 MHz, and built-in 320KB ROM, 512KB HP SRAM and 16KB LP SRAM. It can be compatible with multiple peripheral devices and is easier to use in different application scenarios.
You can choose ESP-IDF development environment or Arduino IED for development so that you can easily and quickly get started and apply it to the product.

ESP32-C6-Zero
Without header

(https://www.waveshare.com/esp32-c6-zero.htm)

ESP32-C6, USB Type-C

Features

- Adopts ESP32-C6FH4 chip, with RISC-V 32-bit single-core processor, support up to 160 MHz.
- Integrated 320KB ROM, 512KB HP SRAM, 16KB LP SRAM and 4MB Flash memory.

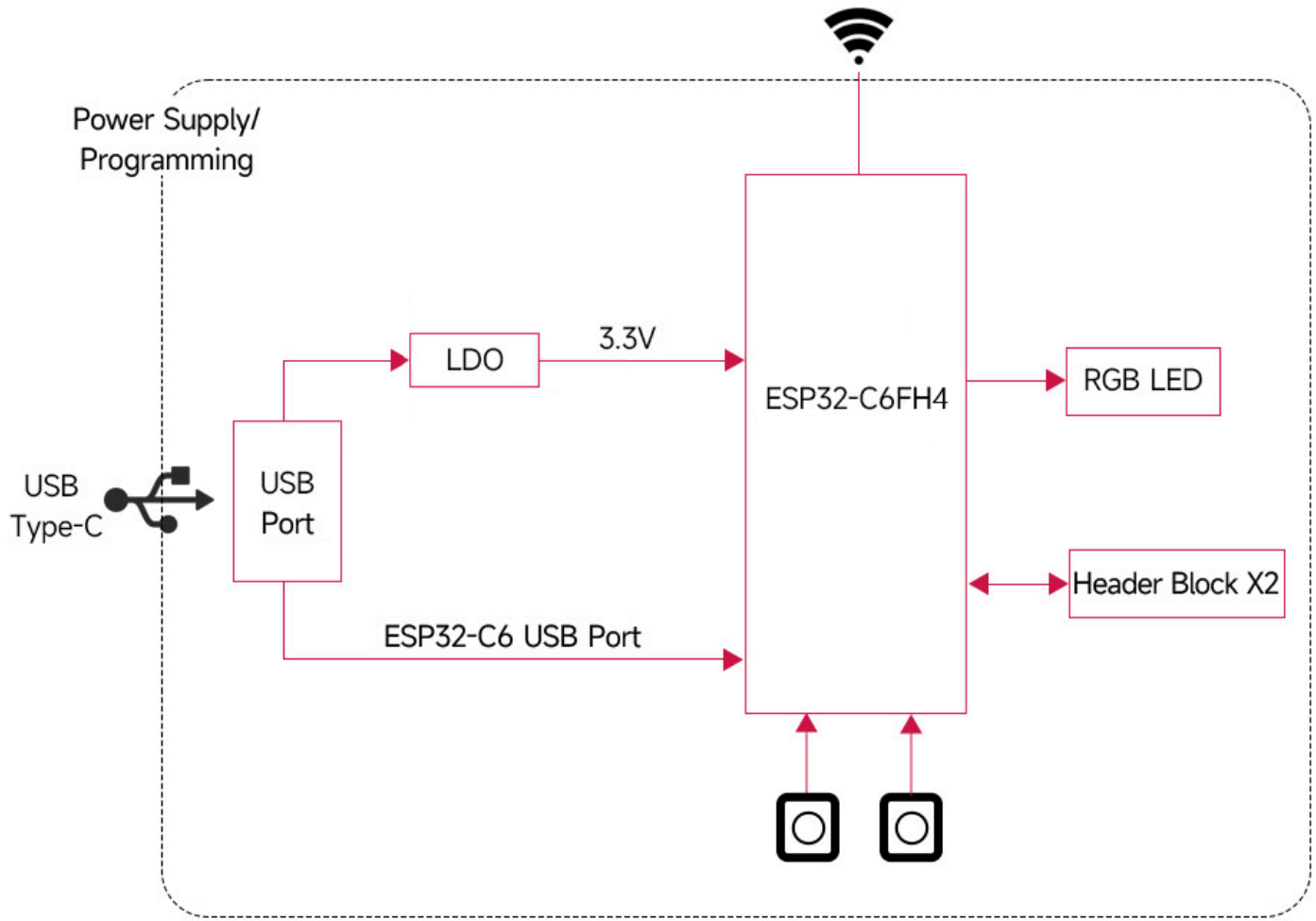
- Integrated WiFi 6, Bluetooth 5 and IEEE 802.15.4 (Zigbee 3.0 and Thread) wireless communication, with superior RF performance.
- Type-C connector, easier to use.
- Rich peripheral interfaces, better compatibility and expandability.
- Castellated module allows soldering directly to carrier boards.
- Support multiple low-power modes, adjustable balance between communication distance, data rate and power consumption to meet the power requirements of various application scenarios.

Function Diagram

ESP32-C6-Zero
With pre-soldered header

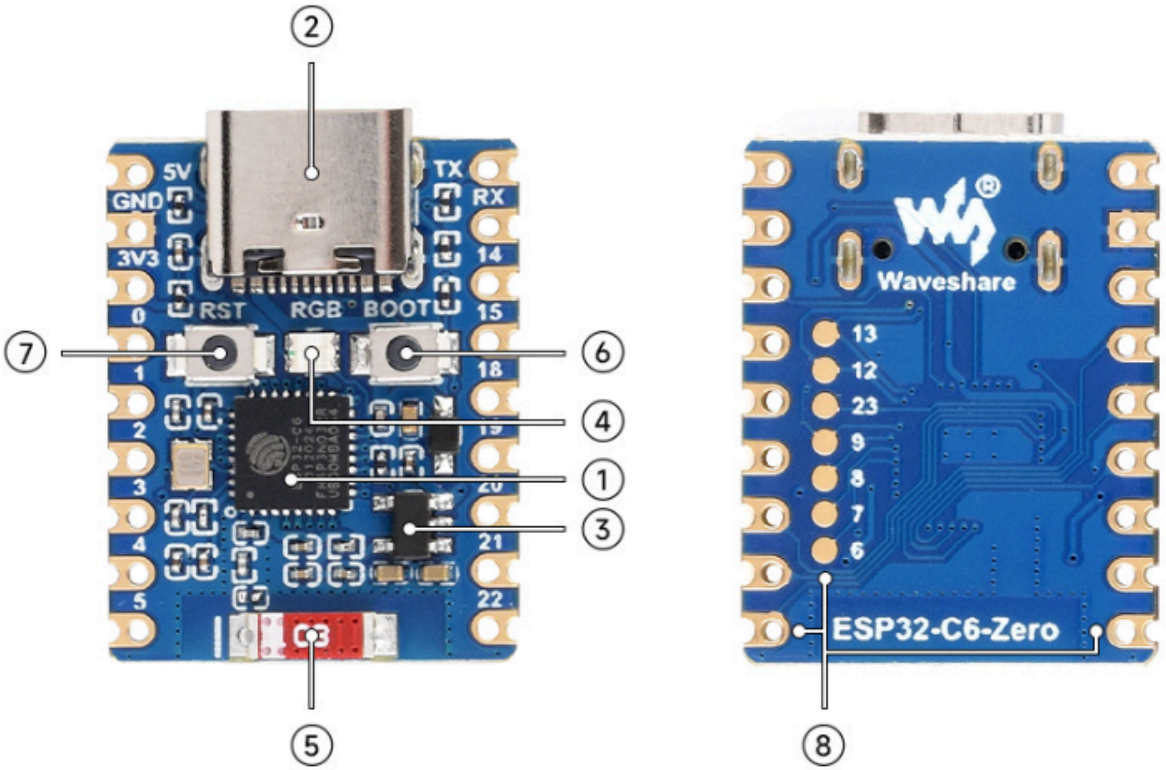
(https://www.waveshare.com/esp32-c6-zero.htm?sku=26976)

ESP32-C6, USB Type-C



(/wiki/File:ESP32-C6-Zero-Diagram.png)

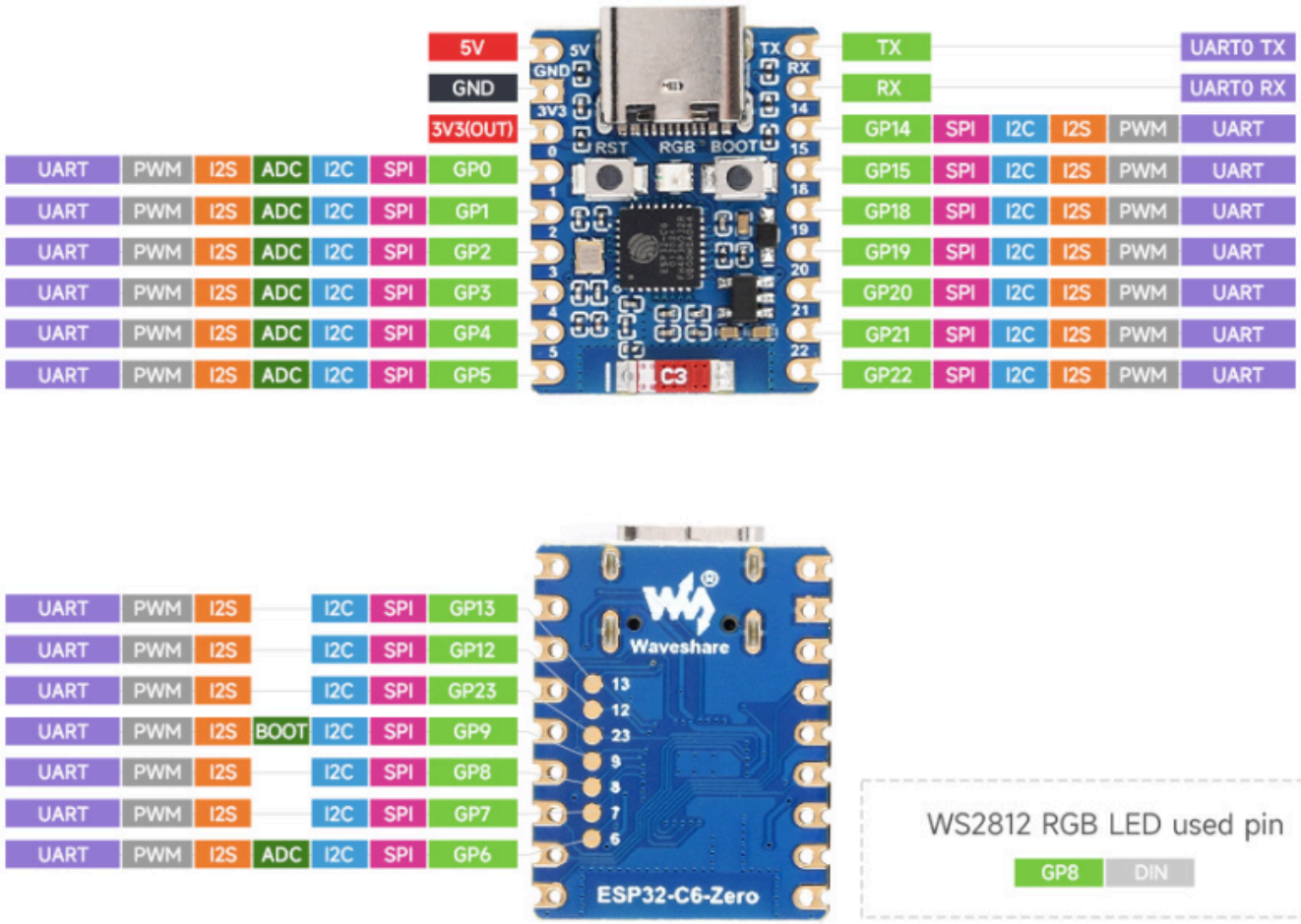
Onboard Interfaces



1. **ESP32-C6FH4**
   Built-in dual processors, main frequency is up to 160 MHz
2. **USB Type-C Port**
   for downloading programme and debugging
3. **ME6217C33M5G**
   low dropout LDO, 800mA (Max)
4. **WS2812 RGB LED**
5. **2.4G ceramic antenna**
6. **BOOT button**
   Press it and then press the RESET button to enter download mode
7. **RESET button**
8. **ESP32-C6FH4 pins**

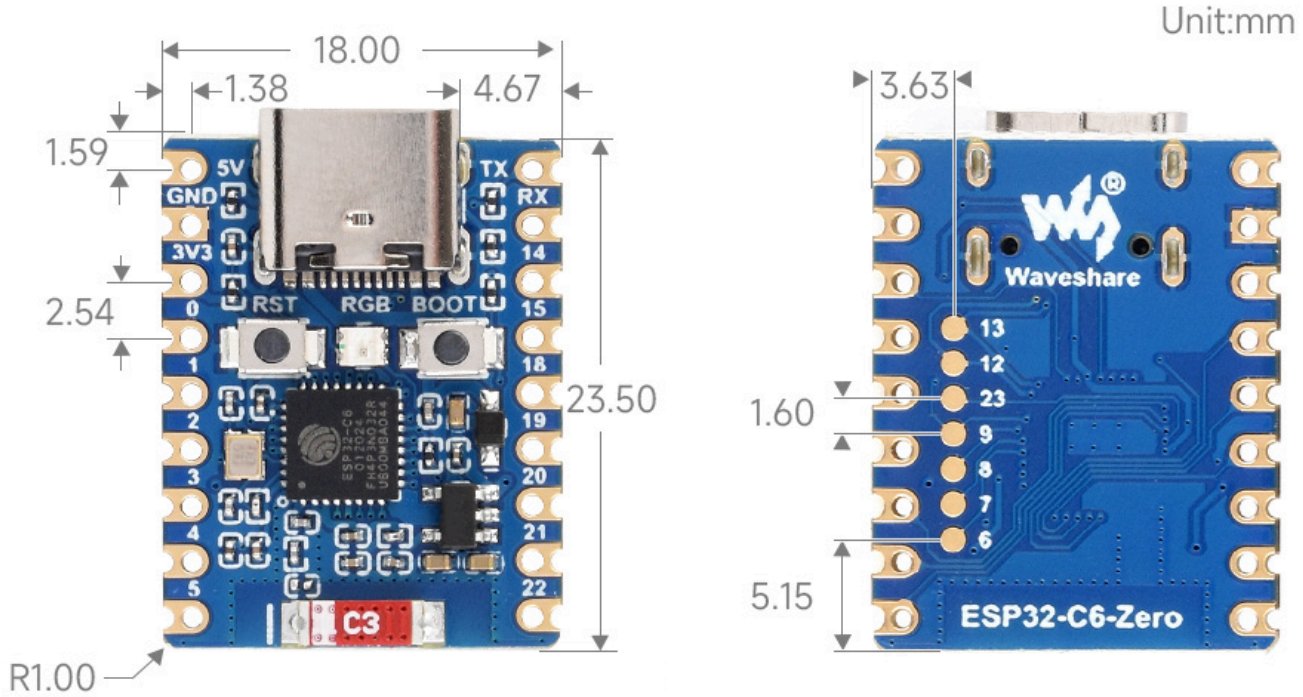
(/wiki/File:ESP32-C6-Zero_Inter.png)

Pinout



(/wiki/File:ESP32-C6-Zero_Pin.png)

Dimensions



(/wiki/File:ESP32-C6-Zero-Diagram-2.png)

Working with ESP-IDF

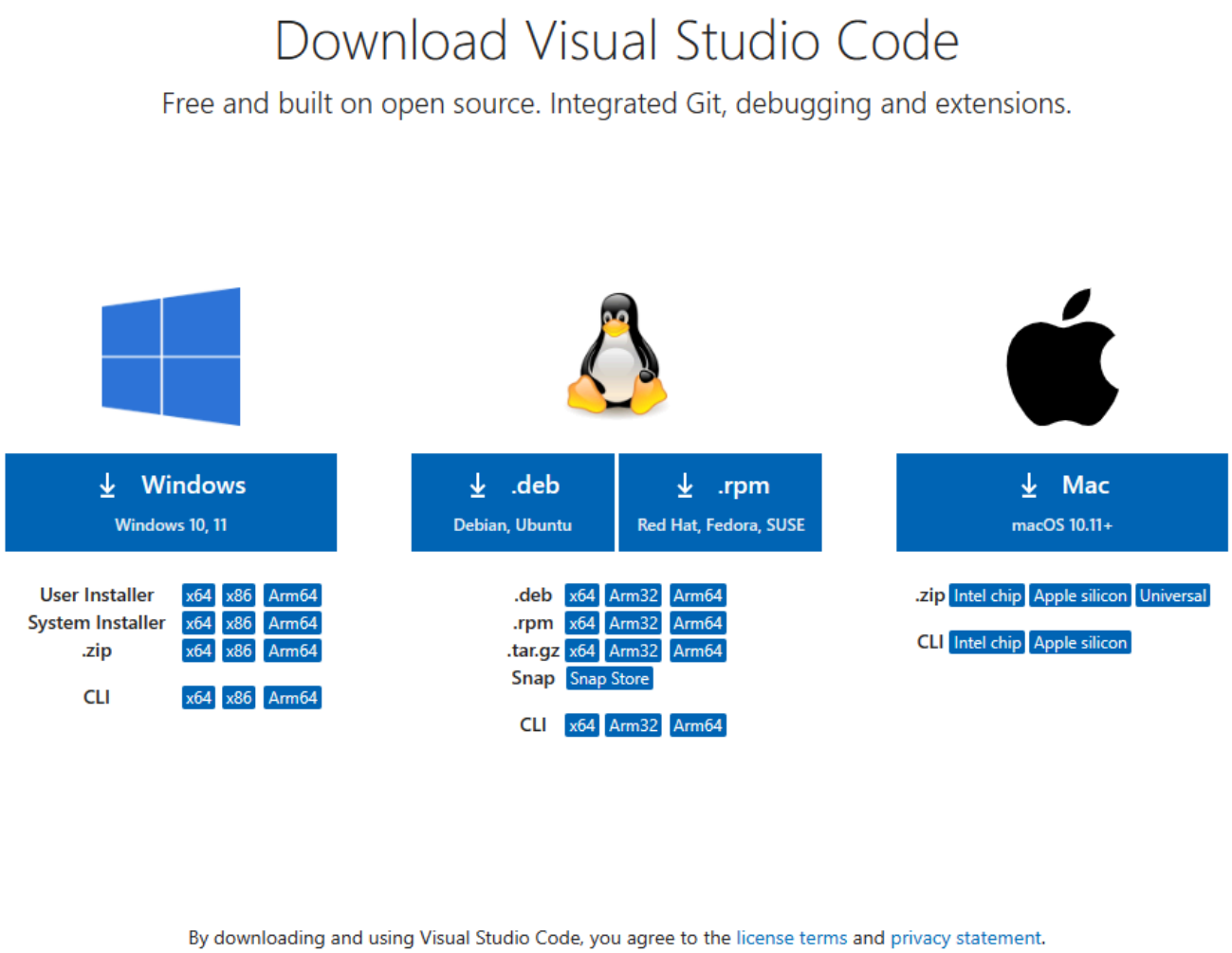
The following development system defaults to Windows, and it is recommended to use

the VSCode plug-in for development.

Develop with VSCode

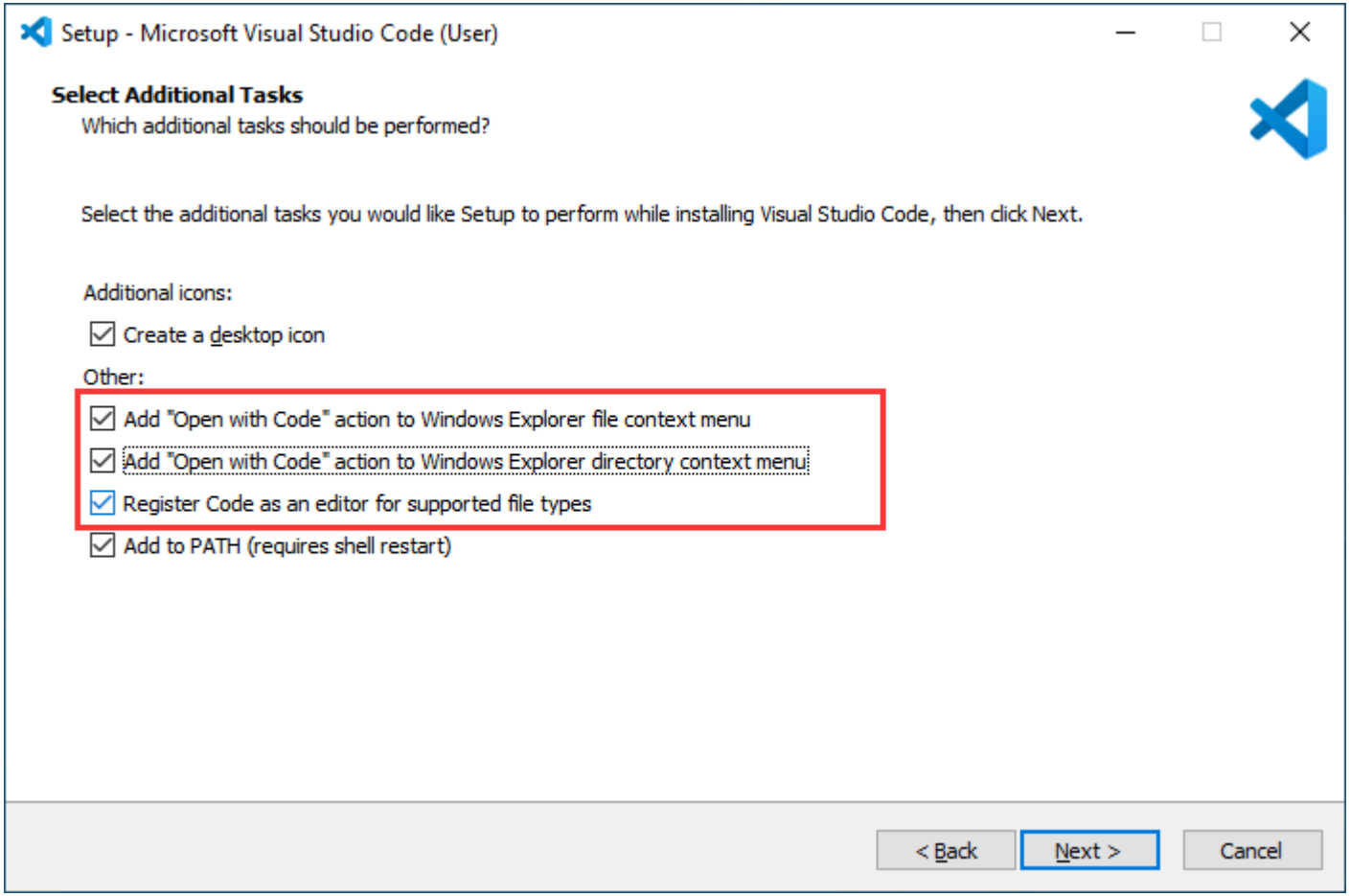
Install VSCode

- Open the VSCode website (https://code.visualstudio.com/download) to download according to the corresponding system and system bits.



(/wiki/File:ESP32-C6-DEV-KIT-N8-01.png)

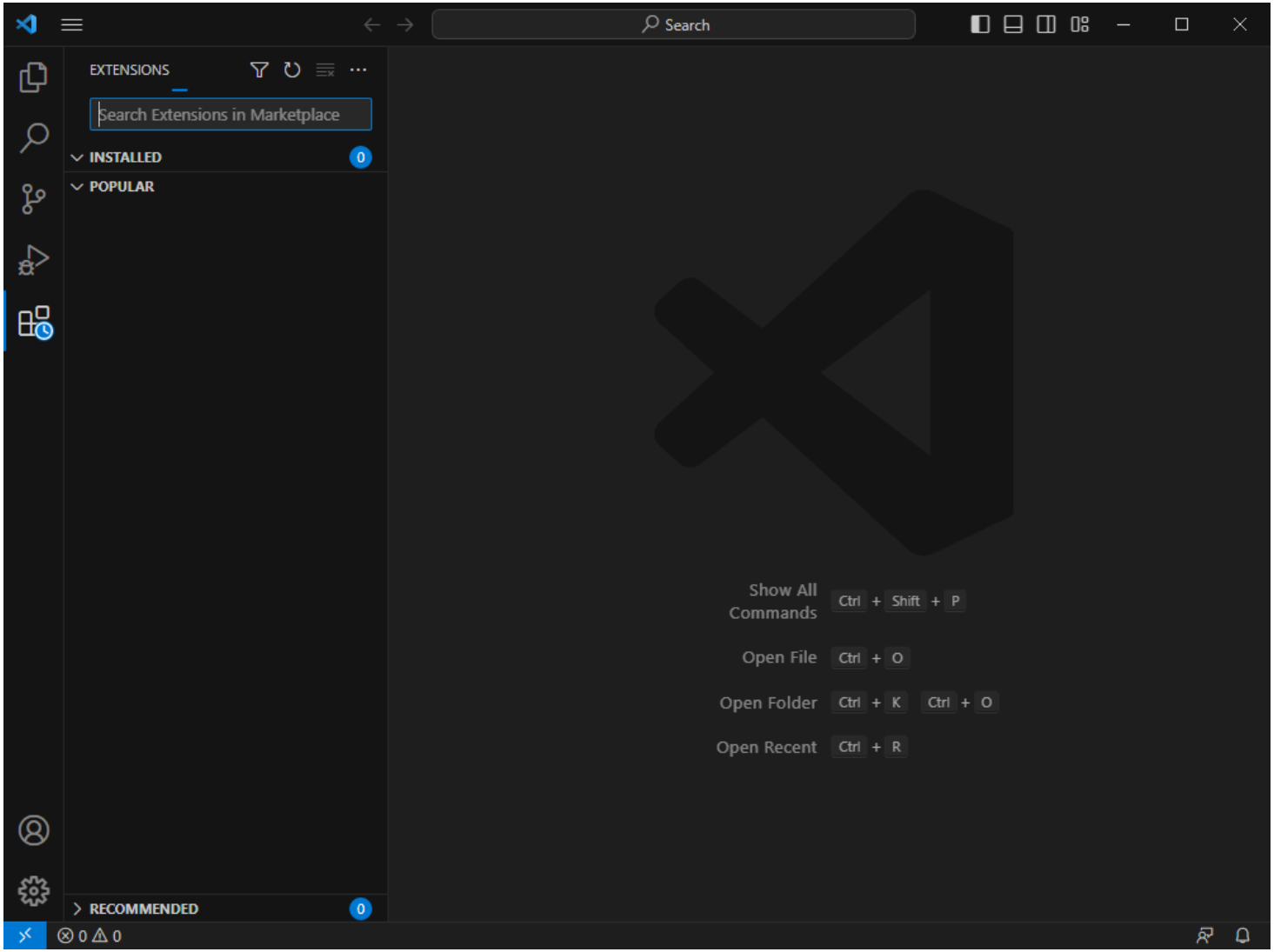
- After running the installation package, the rest can be installed by default, but here for a better experience, it is recommended to check the box in the 1, 2 and 3 items.
  - After enabling the 1st and 2rd items, you can directly open the VScode by right-clicking the file or the directory to improve your experience.
  - After enabling the 3nd items, you can directly select VSCode when choosing how to open.



(/wiki/File:ESP32-C6-DEV-KIT-N8-02.png)

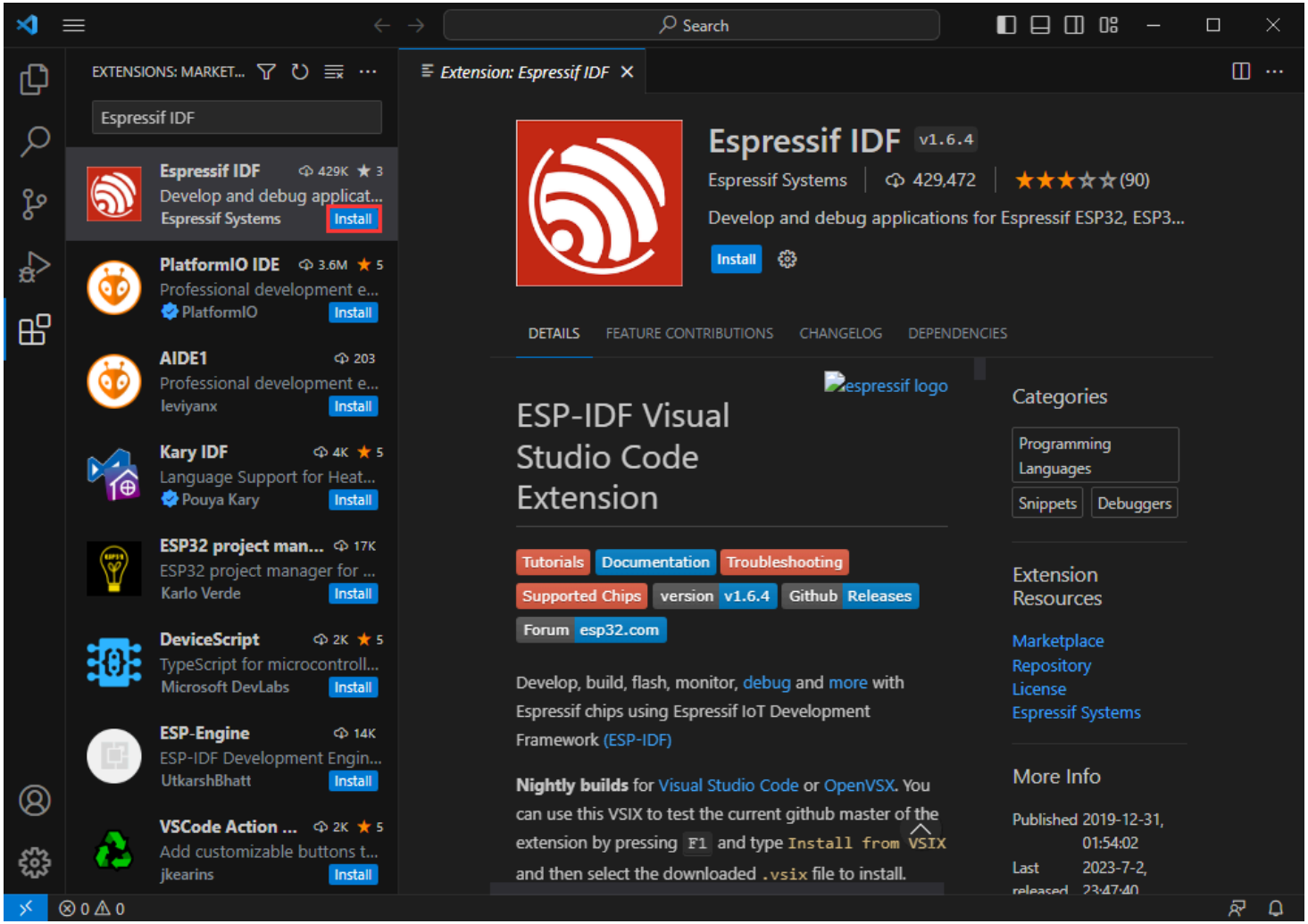
Install Espressif IDF Plug-in

- **Note: Currently the latest version of the plugin is V1.6.4, users can choose the same version as us for a consistent experience!**
- Open VSCode, use **Shift+Ctrl+X** to enter the plug-in manager.



(/wiki/File:ESP32-C6-DEV-KIT-N8-03.png)

- In the search bar, enter **Espressif IDF** to select the corresponding plug-in and click "Install".

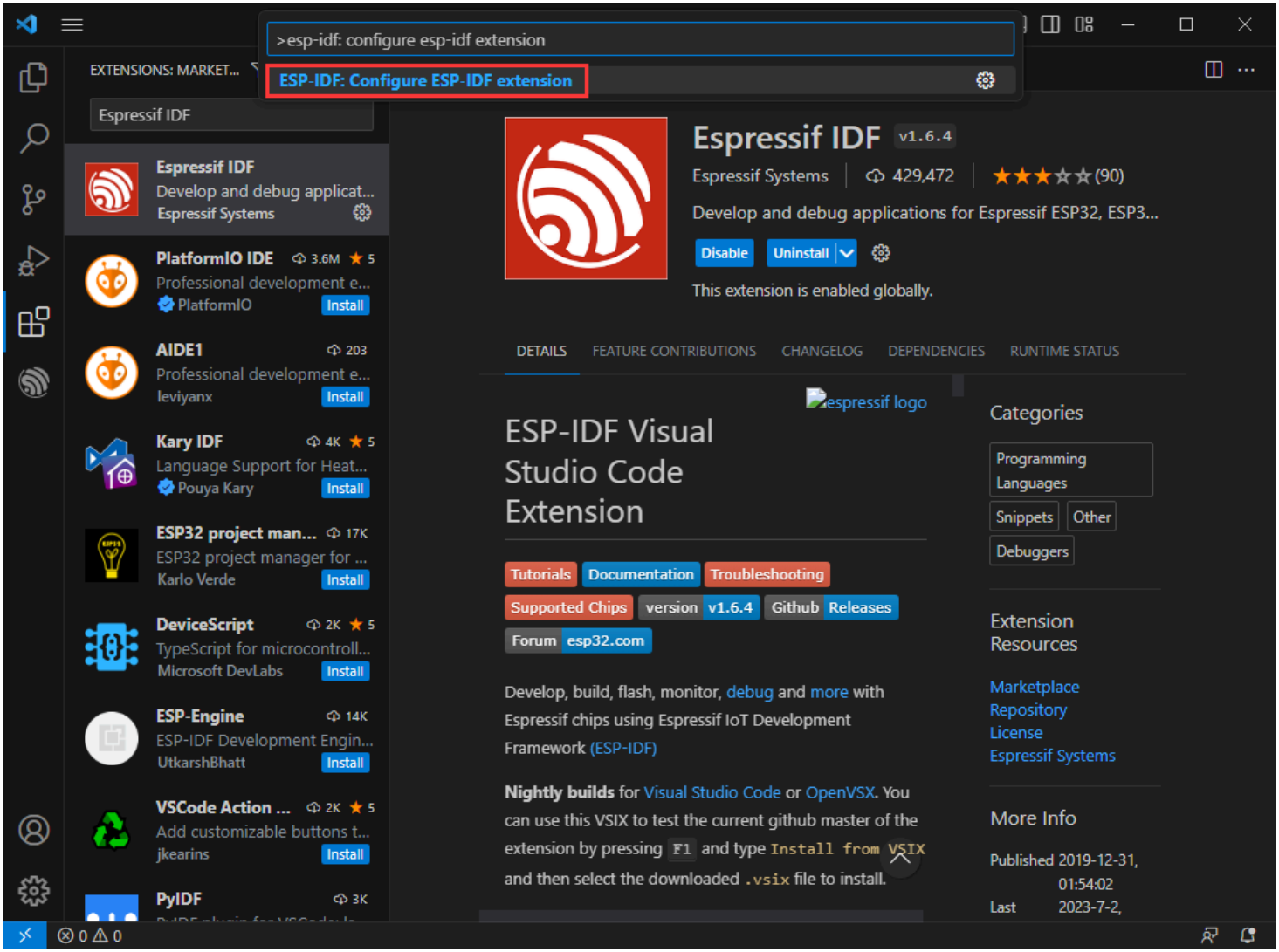


(/wiki/File:ESP32-C6-DEV-KIT-N8-04.png)

(/wiki/File:ESP32-C6-DEV-KIT-N8-05.png)

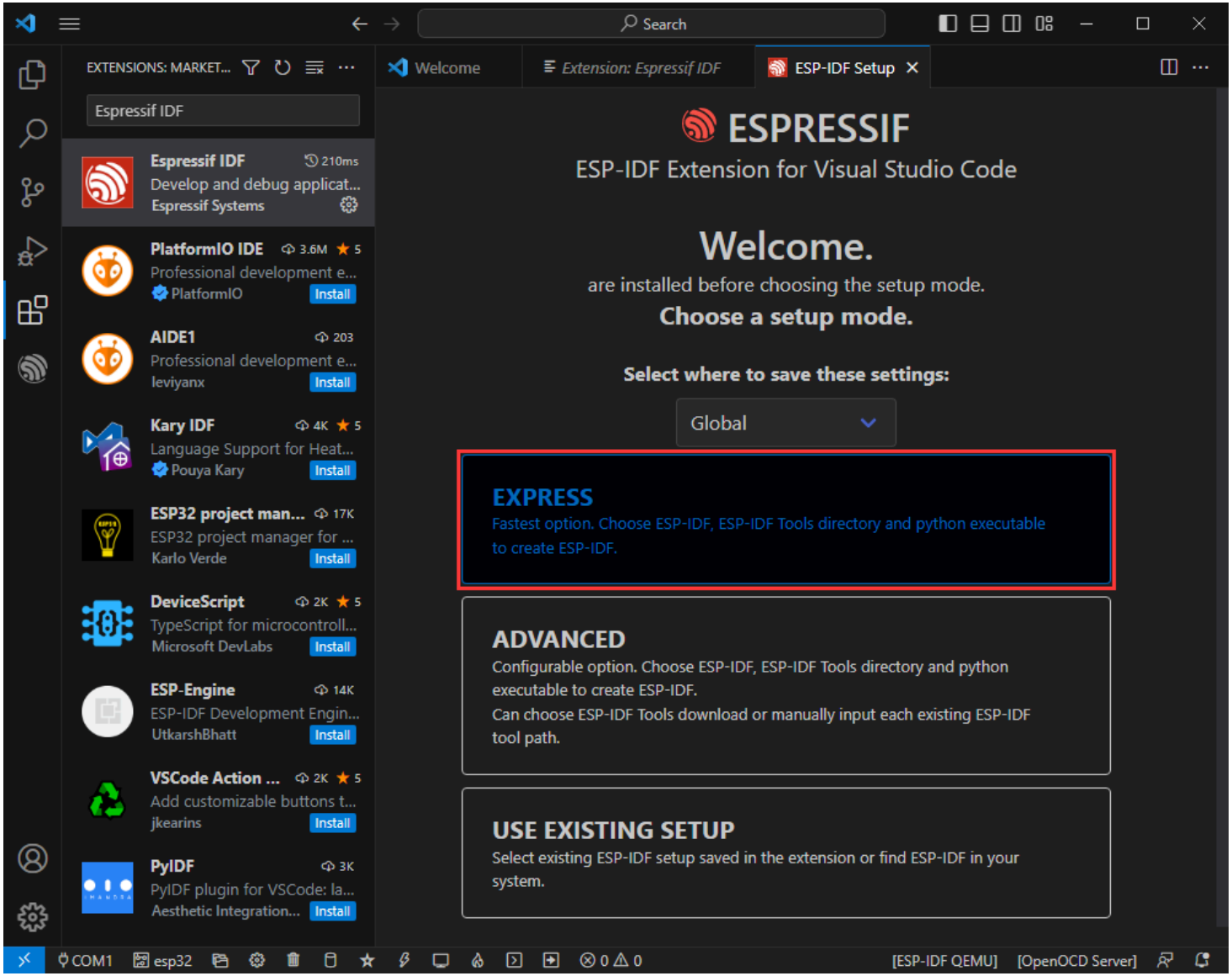
- Press **F1** to input:

```
esp-idf: configure esp-idf extension
```



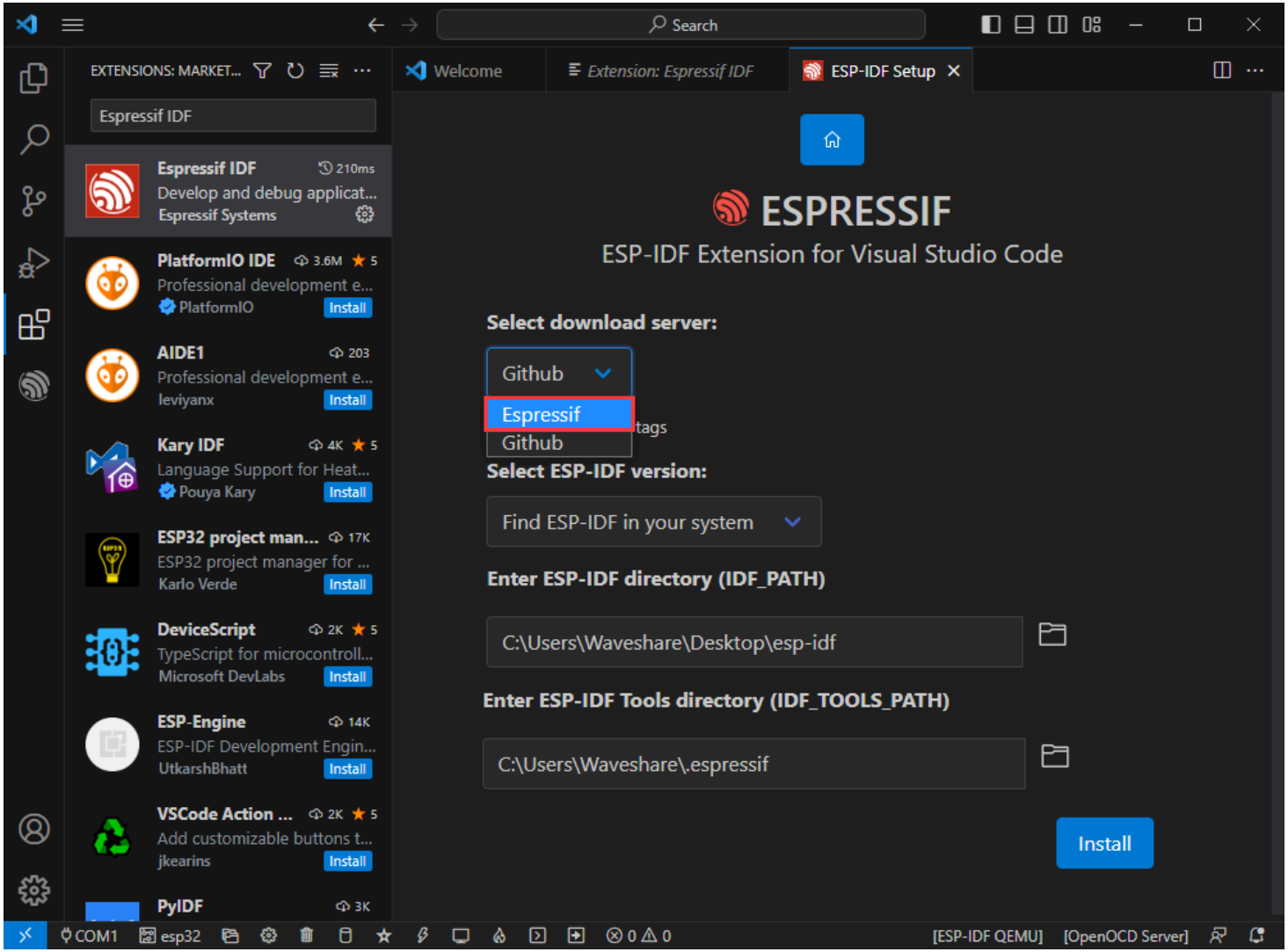
(/wiki/File:ESP32-C6-DEV-KIT-N8-06.png)

- Select express (this guide is for users who install it for the first time).



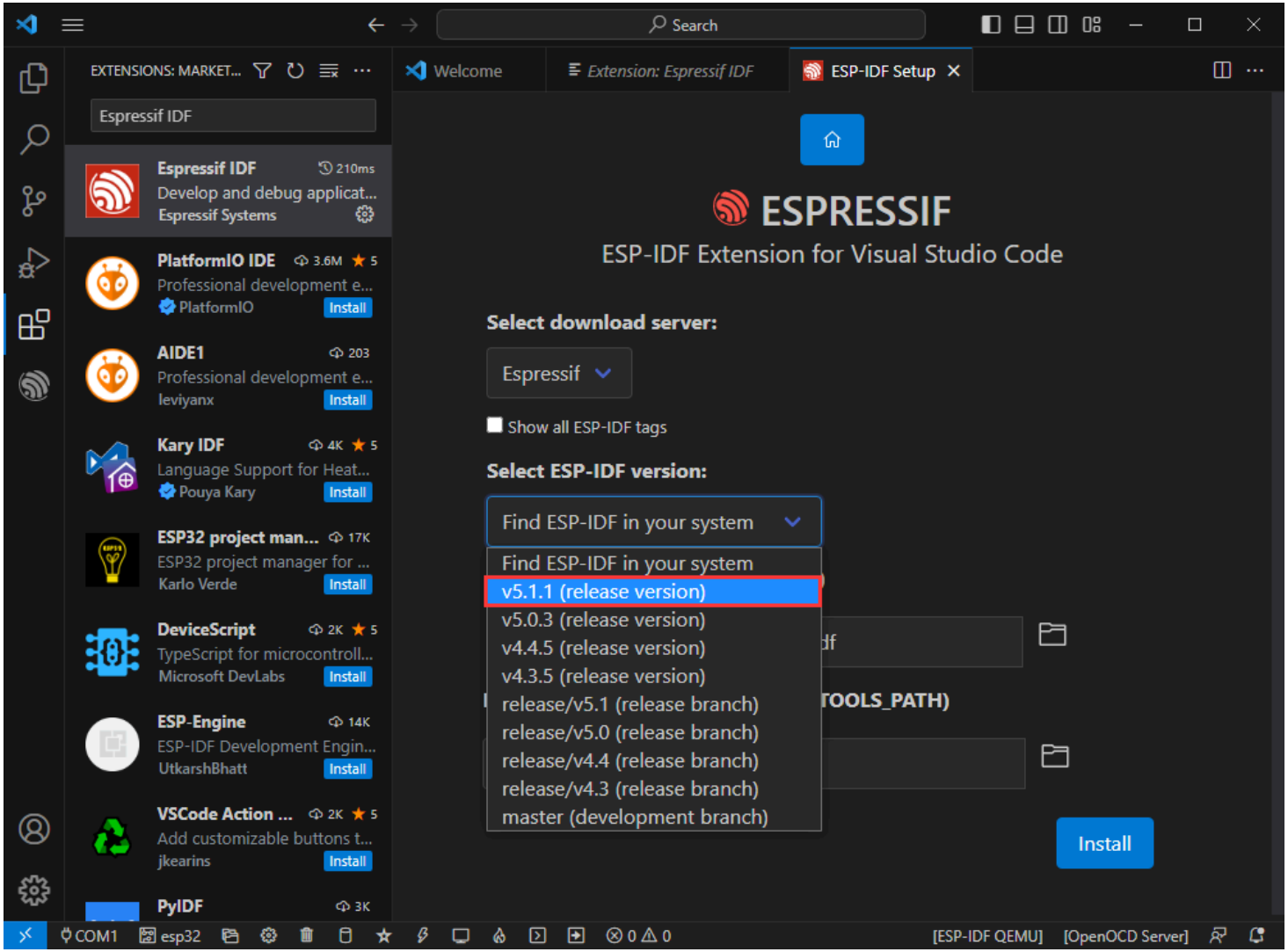
(/wiki/File:ESP32-C6-DEV-KIT-N8-07.png)

- Select download sever.



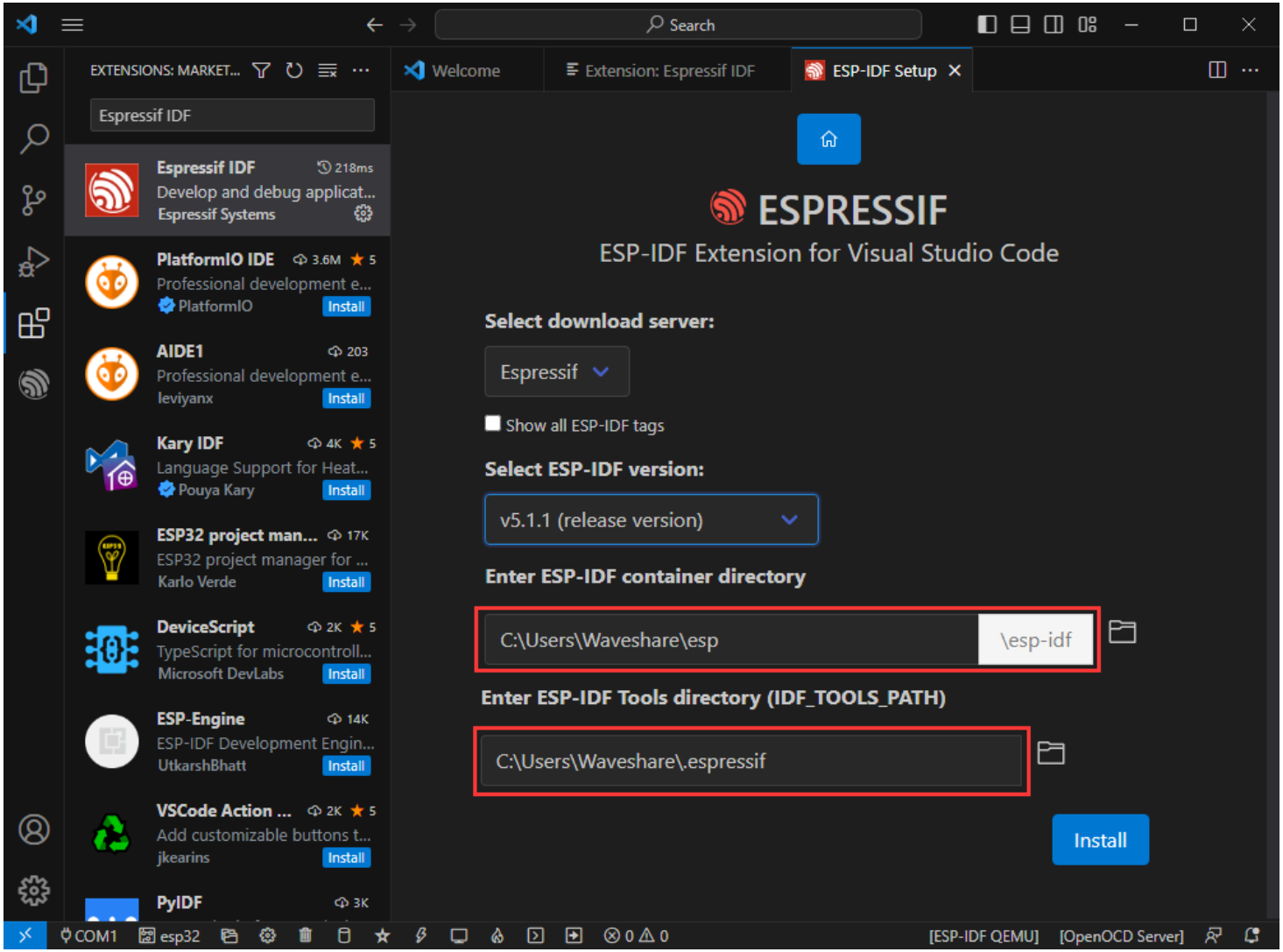
(/wiki/File:ESP32-C6-DEV-KIT-N8-08.png)

- Select the version of ESP-IDF you want to use now, we choose the latest V5.1.1 (note that only after V5.1 did ESP-IDF start to support ESP32-C6).



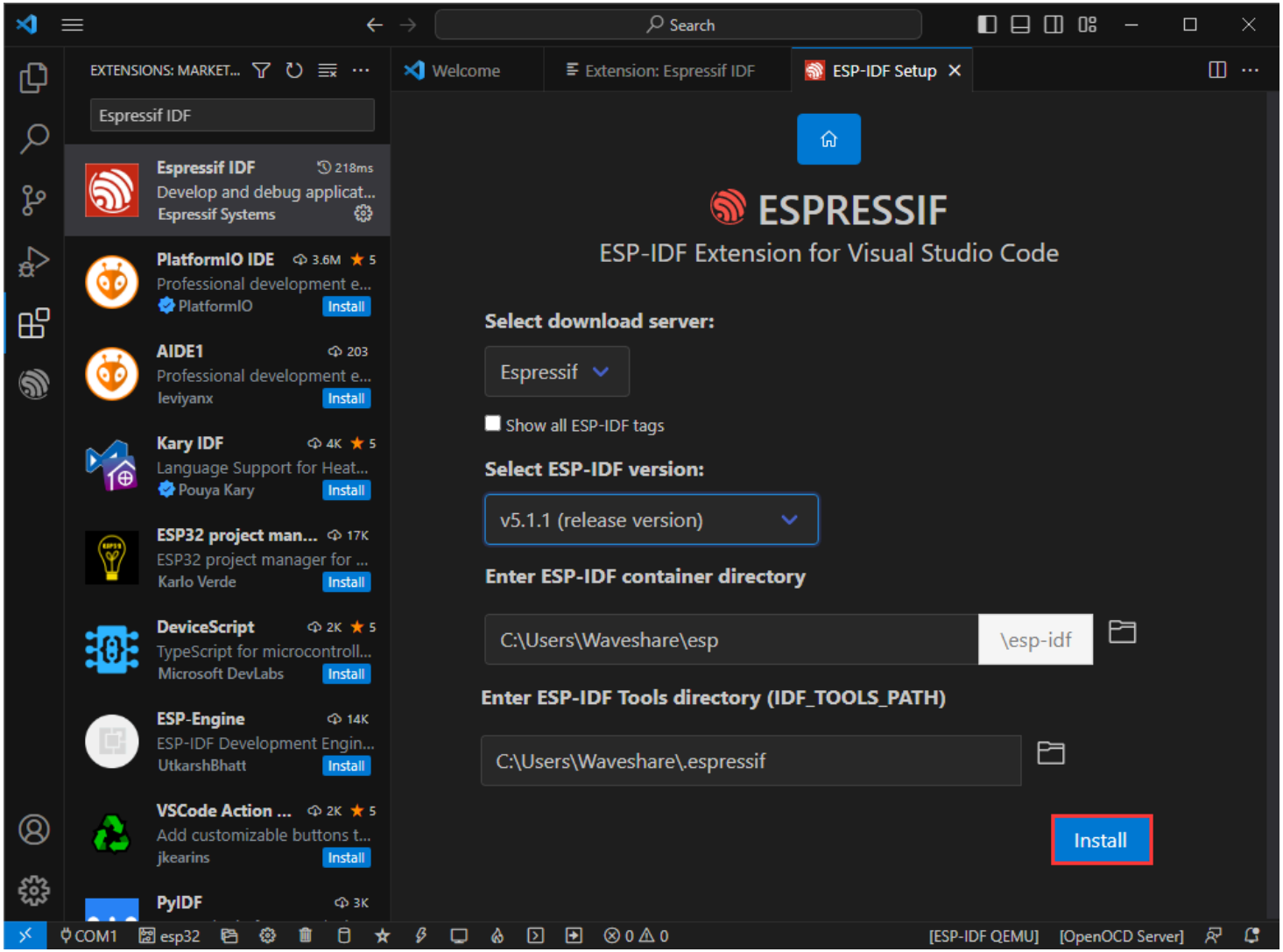
(/wiki/File:ESP32-C6-DEV-KIT-N8-09.png)

- The following two are the installation paths respectively for the ESP-IDF container directory and the ESP-IDF Tools directory.



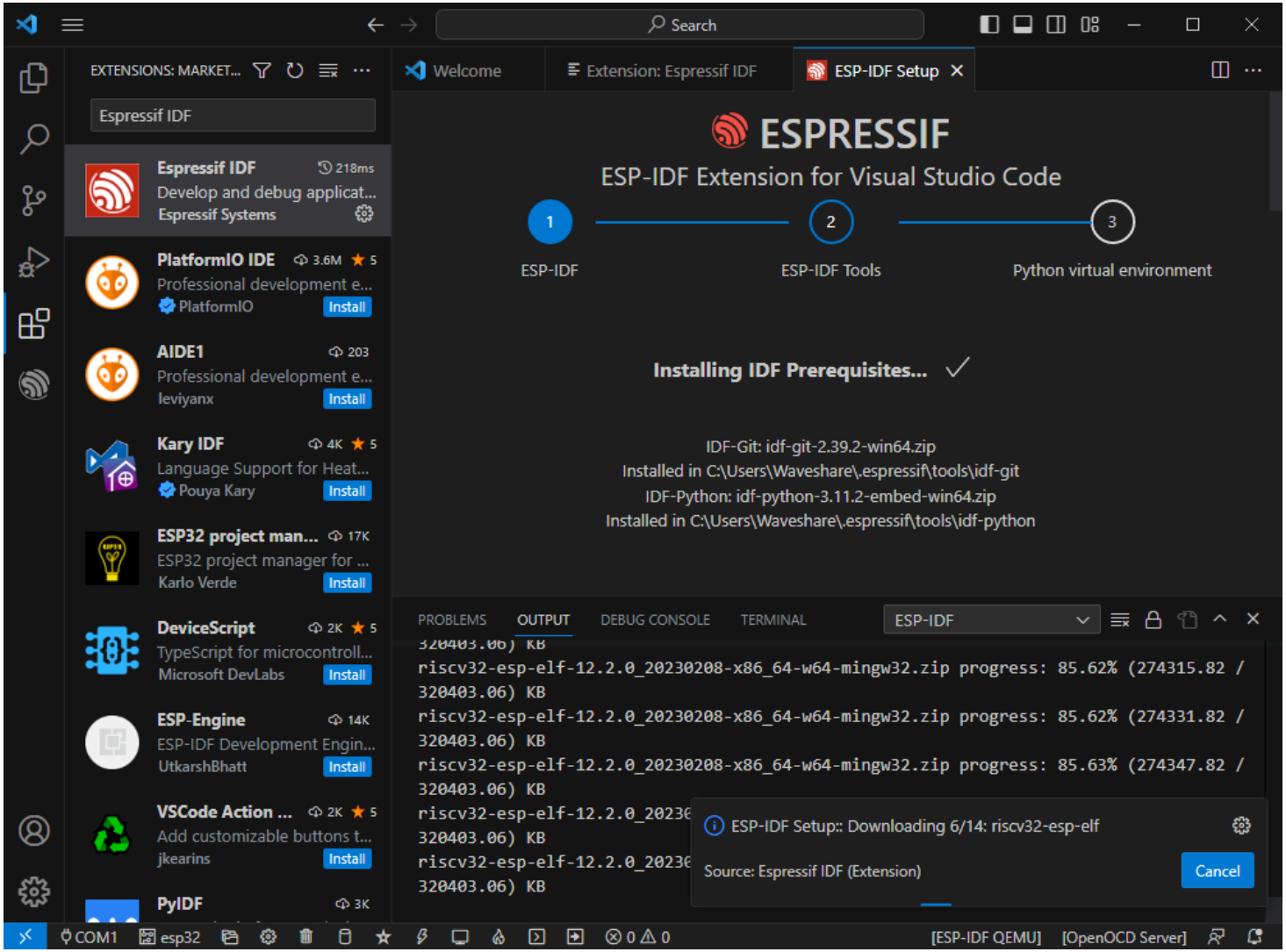
(/wiki/File:ESP32-C6-DEV-KIT-N8-10.png)

- **Note: If you have installed ESP-IDF before, or failed to do so, please be sure to delete the file completely .**
- After configuring, click "Install" to download.



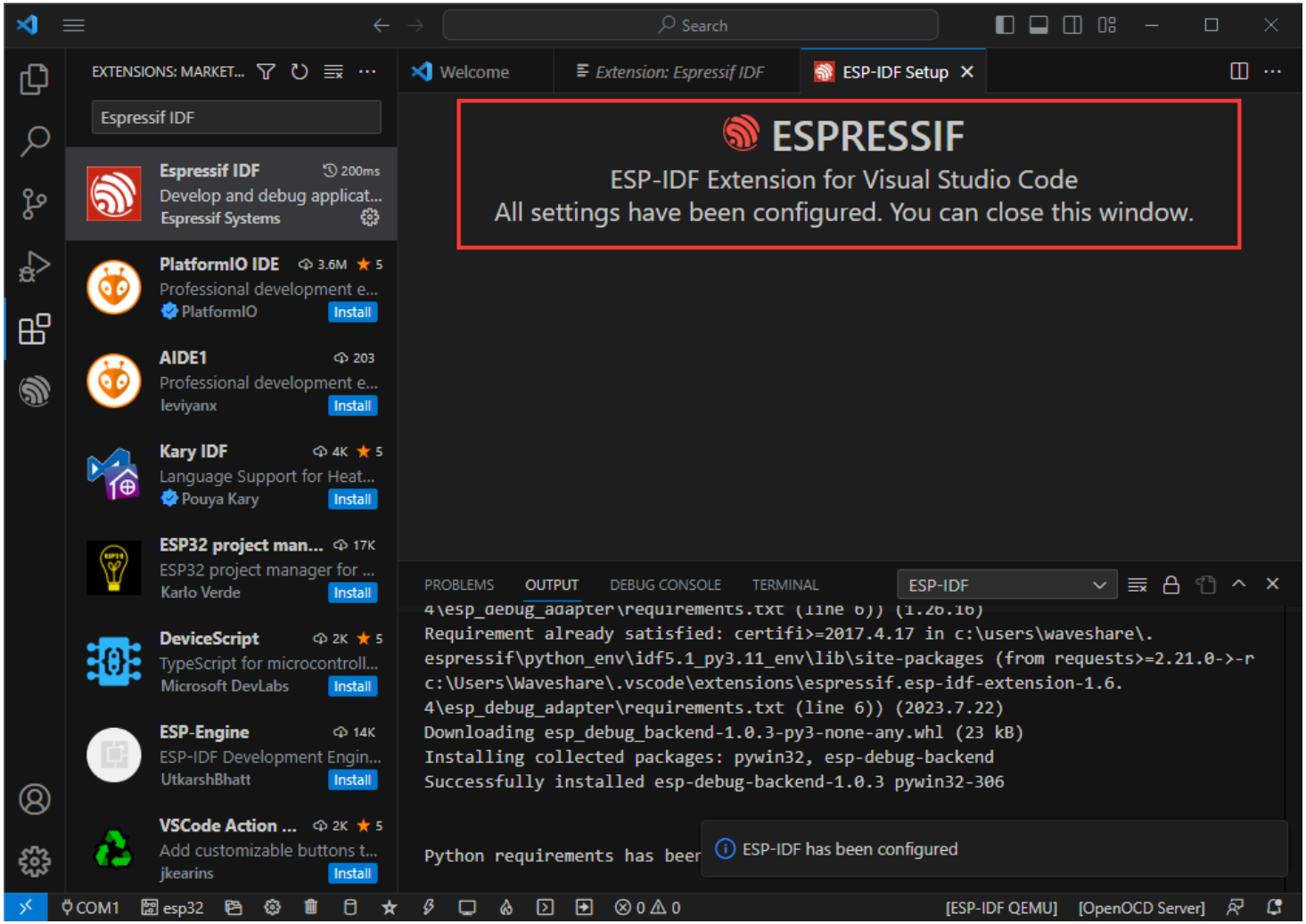
(/wiki/File:ESP32-C6-DEV-KIT-N8-19.png)

- Enter the download interface, and then it will automatically install the corresponding tools and environment, just wait for a second.



(/wiki/File:ESP32-C6-DEV-KIT-N8-11.png)

- After the installation is complete, you will enter the following interface, indicating that the installation is finished.



(/wiki/File:ESP32-C6-DEV-KIT-N8-12.png)

Official Demo Usage GUIDE

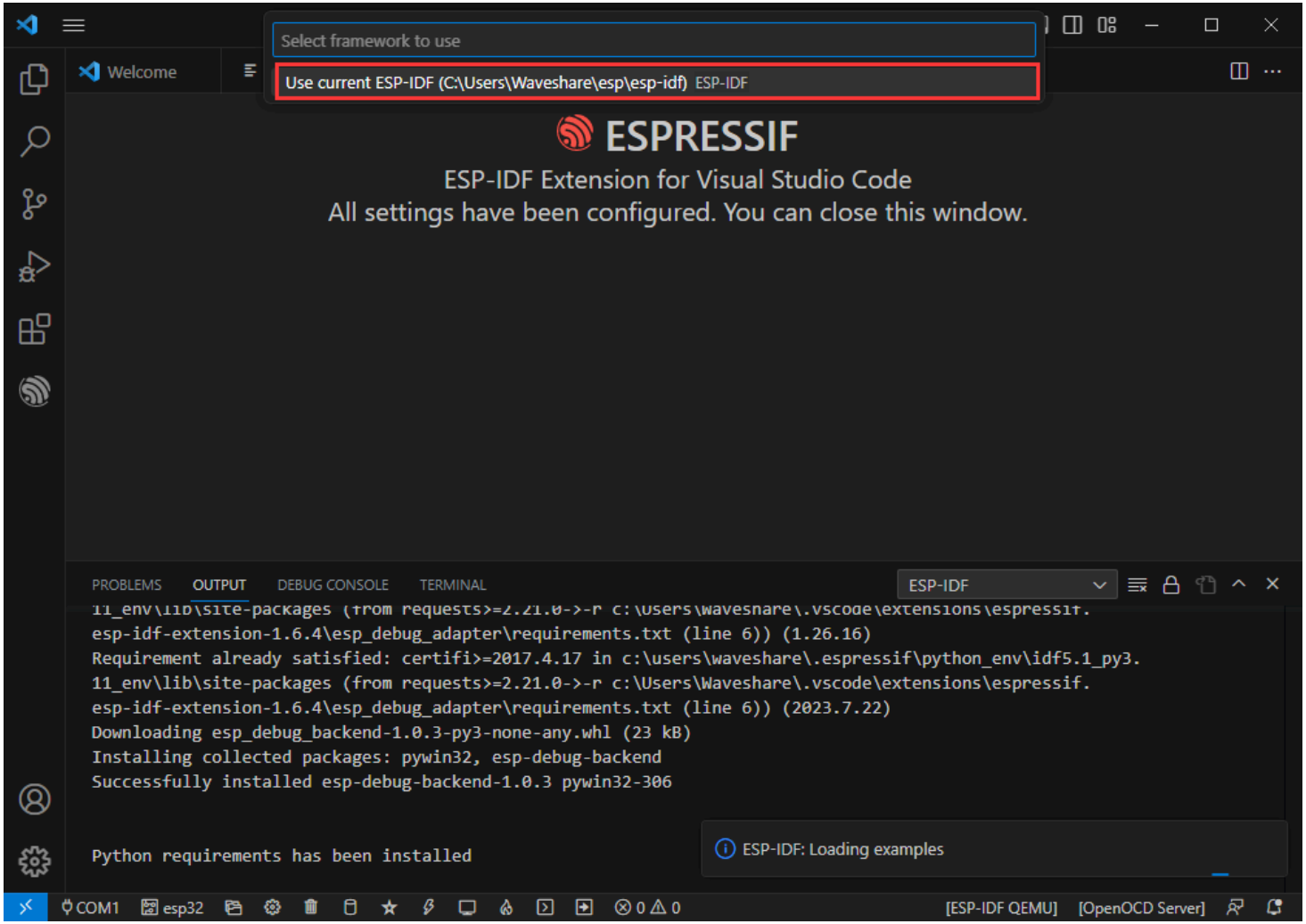
Create Demo (#Demo Example)

- Press **F1** to enter:

```
esp-idf:show examples projects
```

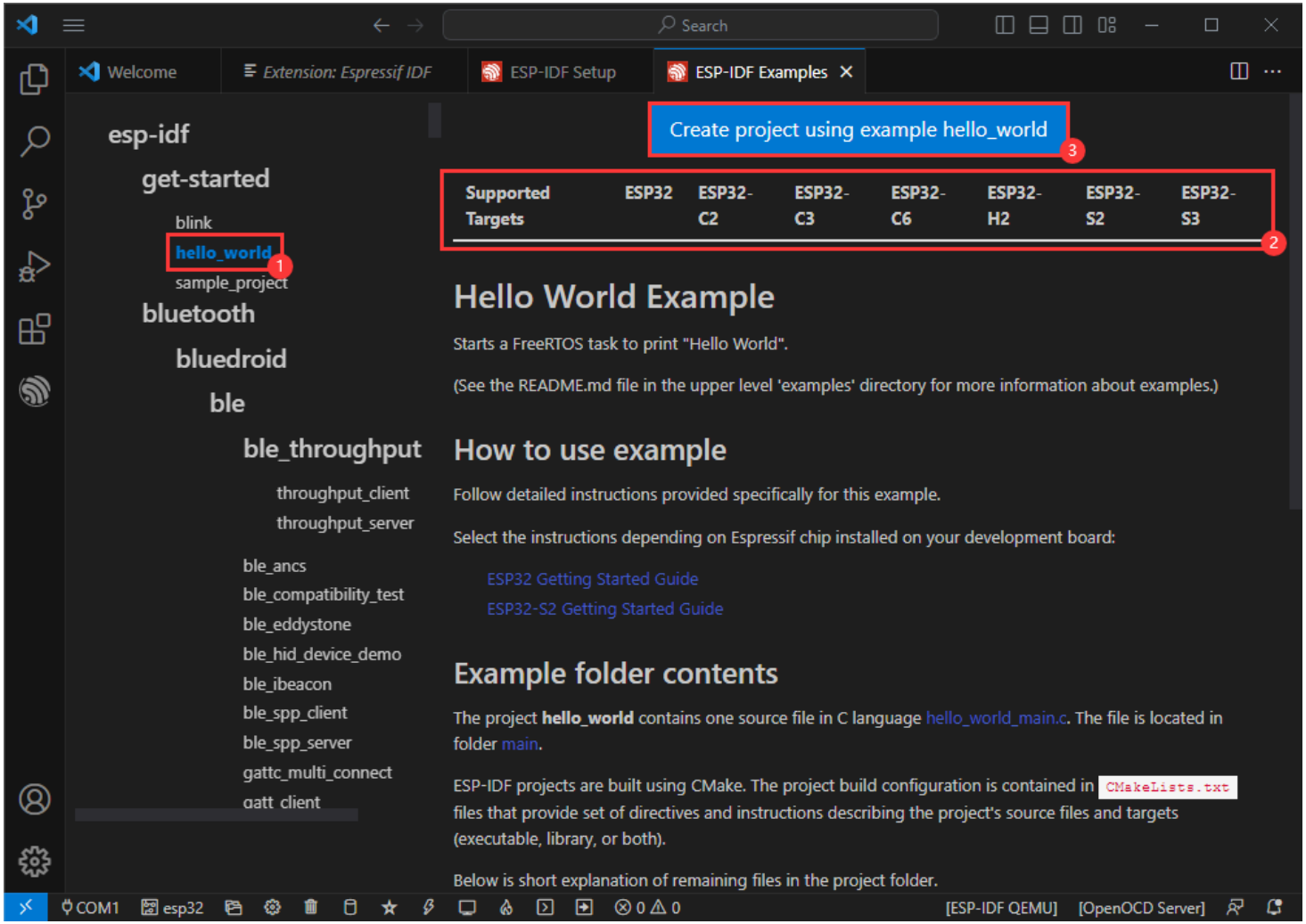
(/wiki/File:ESP32-C6-DEV-KIT-N8-13.png)

- Select the corresponding IDF version.



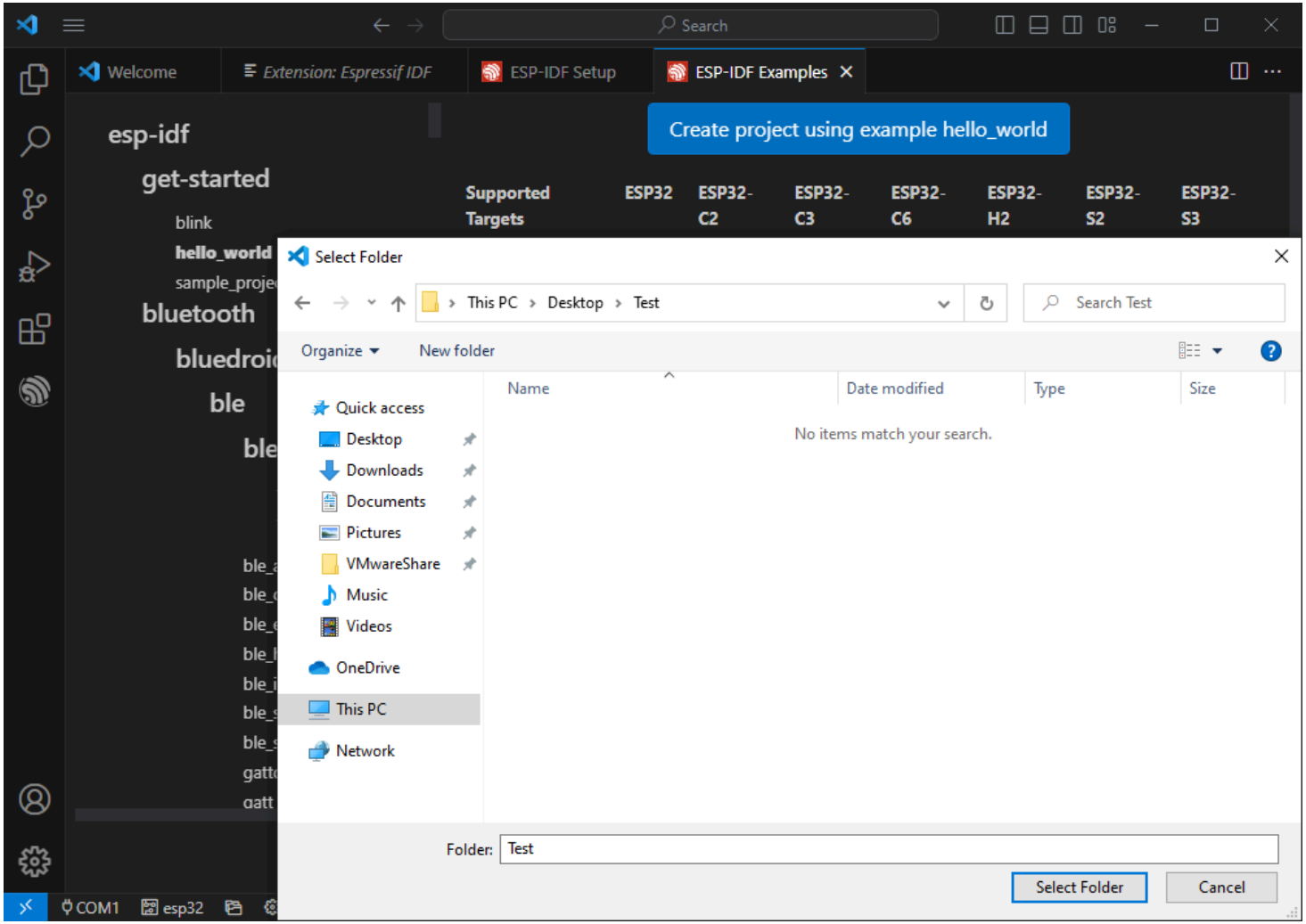
(/wiki/File:ESP32-C6-DEV-KIT-N8-14.png)

- Take the Hello World demo as an example:
  - ①Select the corresponding demo.
  - ②Its readme will state what chip the demo applies to (how the demo is used and the file structure are described below, omitted here).
  - ③Click to create the demo.



(/wiki/File:ESP32-C6-DEV-KIT-N8-15.png)

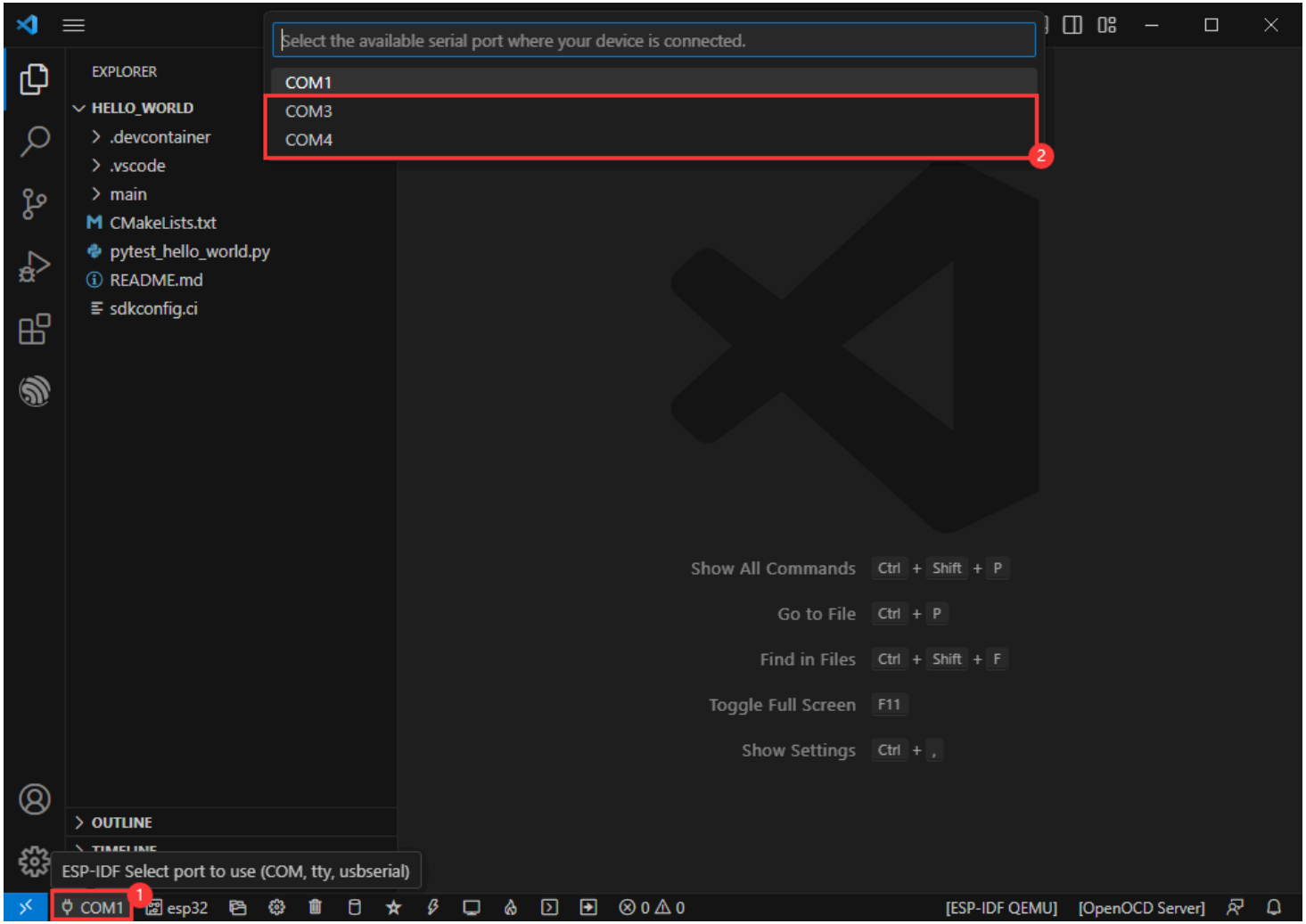
Select the path to place the demo, and no folder with the same name as the demo is required.



(/wiki/File:ESP32-C6-DEV-KIT-N8-16.png)

Modify COM Port

- The corresponding COM ports are shown here, click to modify them.
- Please select the COM ports according to your device. **It is recommended to prioritize the use of the COM port corresponding to USB (can be viewed through the device manager).**
- In case of a download failure, please press the reset button for more than 1 second and wait for the PC to recognize the device again before downloading once more.



(/wiki/File:ESP32-C6-DEV-KIT-N8-17.png)

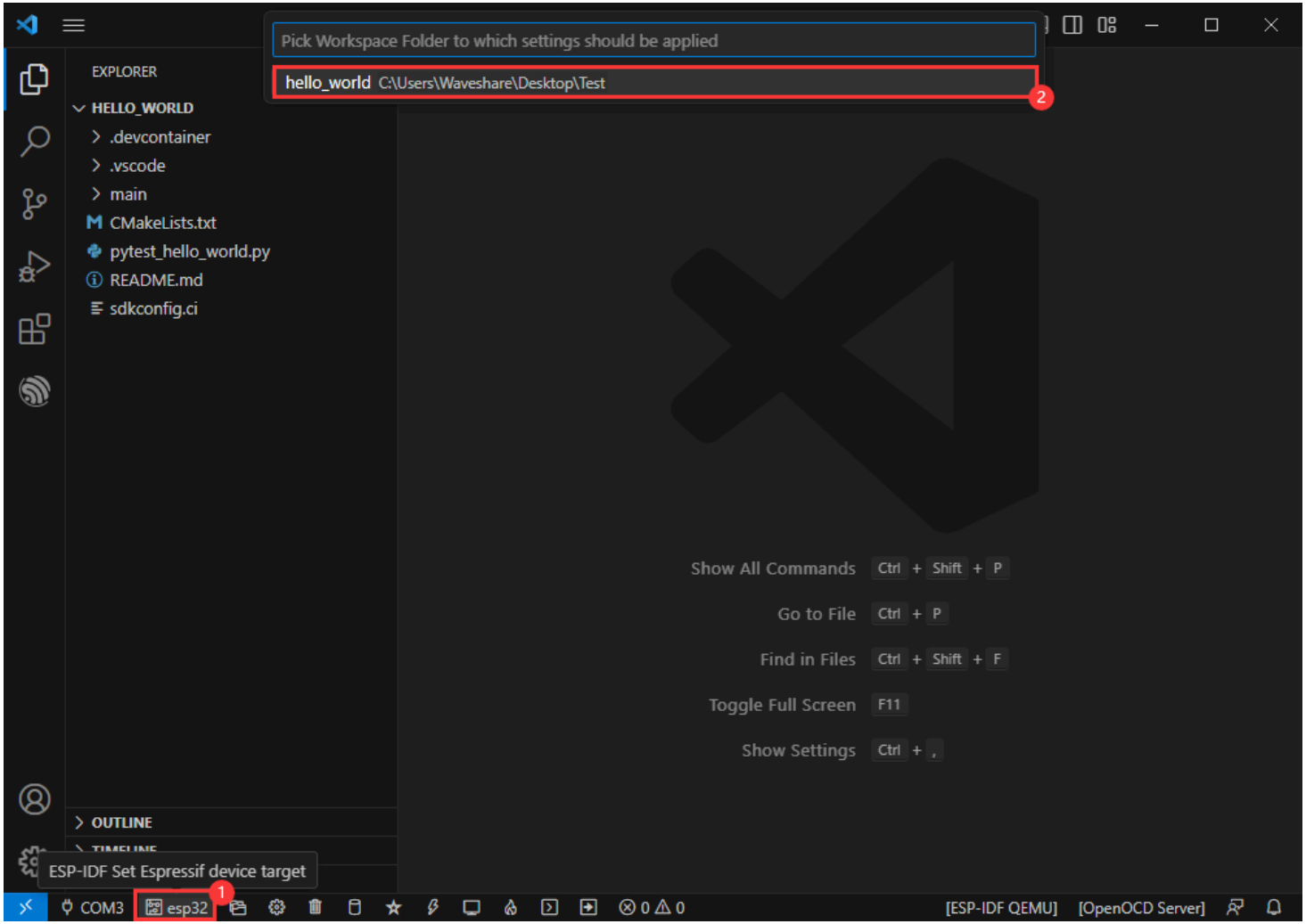
- Select the project or demo to use.

(/wiki/File:ESP32-C6-DEV-KIT-N8-18.png)

- Then we finish the modification of the COM ports.

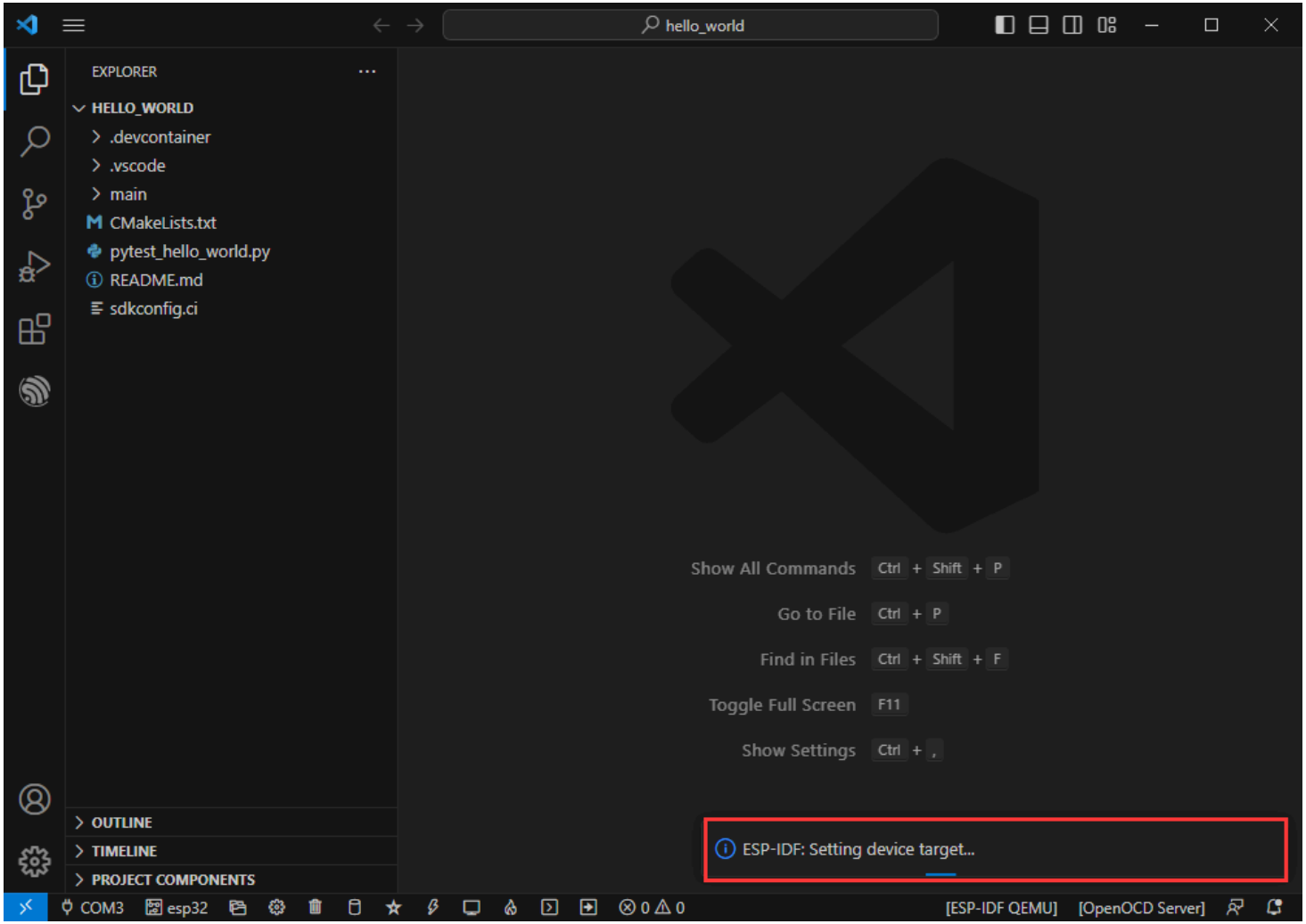
Modify the Driver Object

- The driver object is displayed here, and you can modify it by clicking on it.
- Select the project or demo to use.



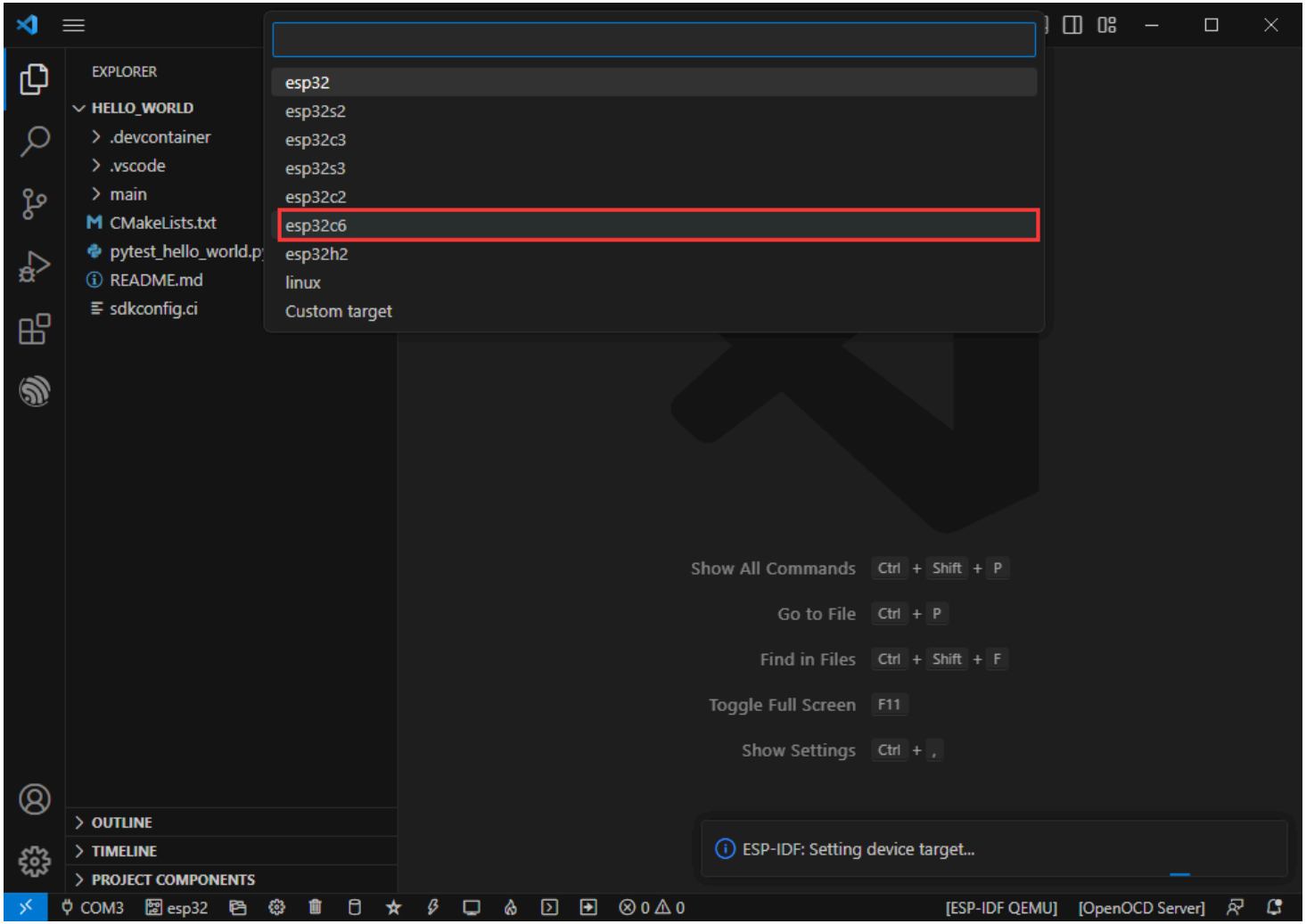
(/wiki/File:ESP32-C6-DEV-KIT-N8-20.png)

- Wait for a minute after clicking.



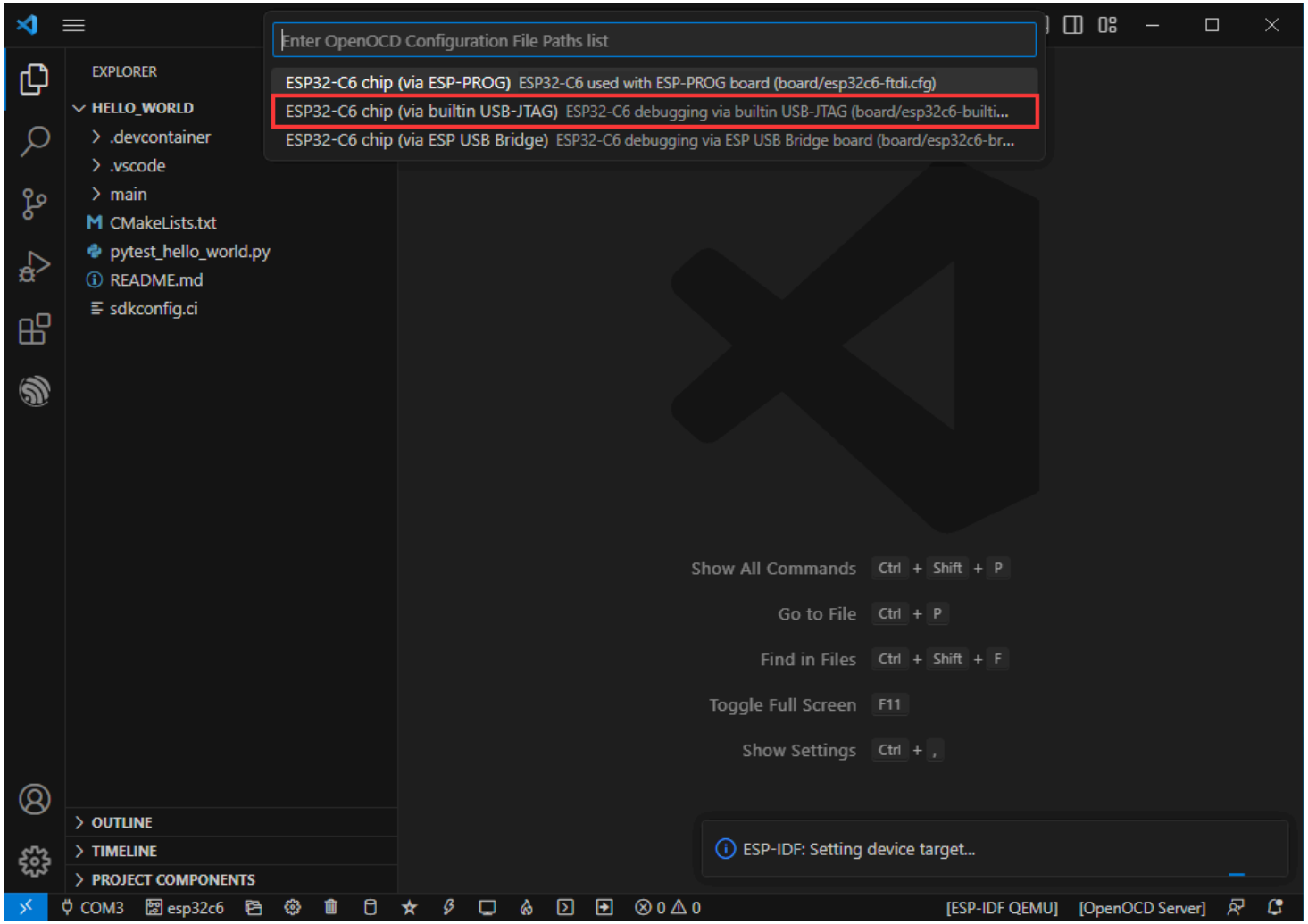
(/wiki/File:ESP32-C6-DEV-KIT-N8-21.png)

- Select the object we need to drive, which is our main chip ESP32C6.



(/wiki/File:ESP32-C6-DEV-KIT-N8-22.png)

- Choose the path to openocd, it doesn't affect us here, so let's just choose one at random.

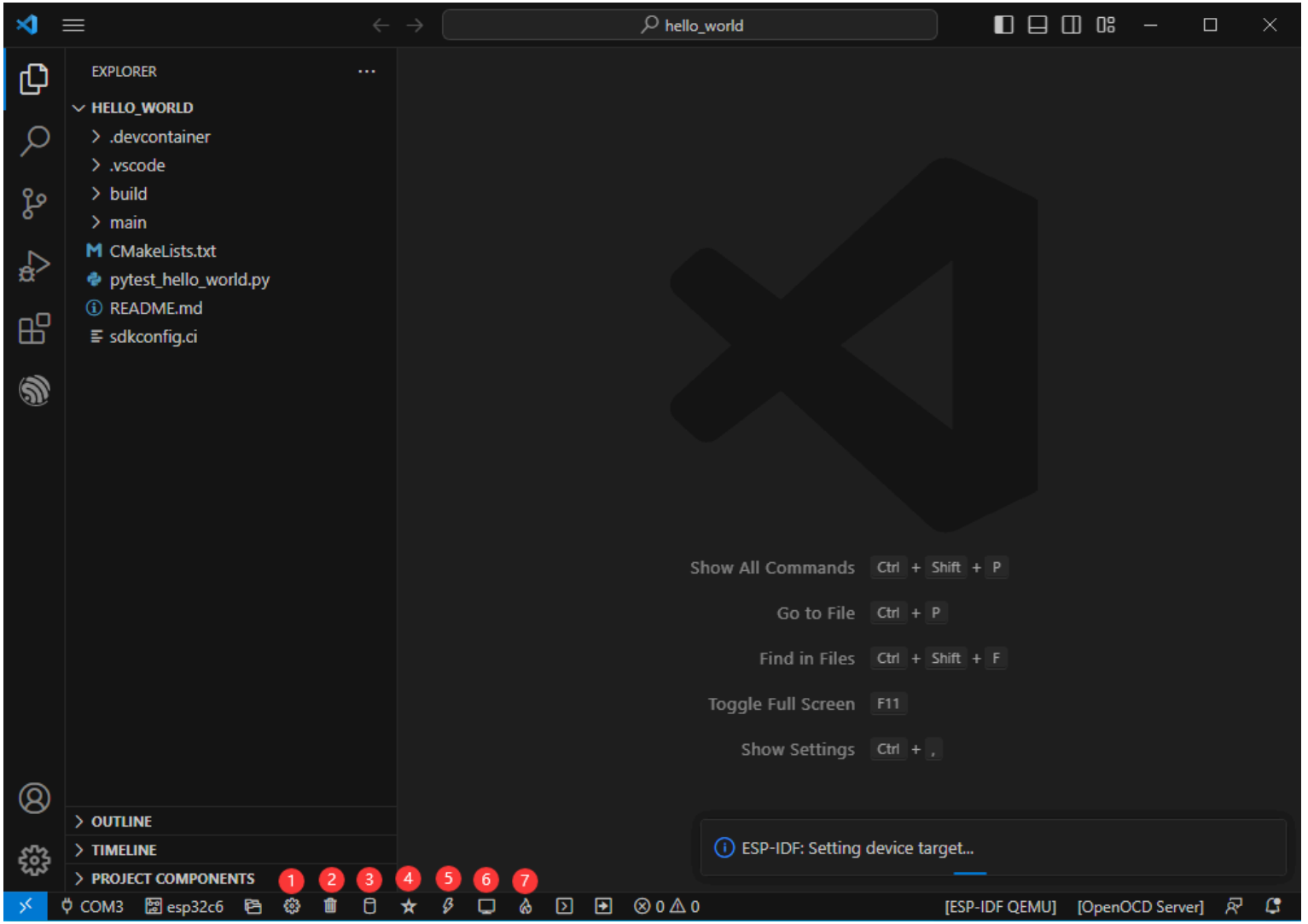


(/wiki/File:ESP32-C6-DEV-KIT-N8-23.png)

The Rest of the Status Bar

- ①SDK configuration editor, supports modifying most functions of ESP-IDF.
- ②All cleanup, and clear all compiled files.
- ③Compile.
- ④Current download mode, default is UART.
- ⑤Flash the current firmware, please do it after compiling.

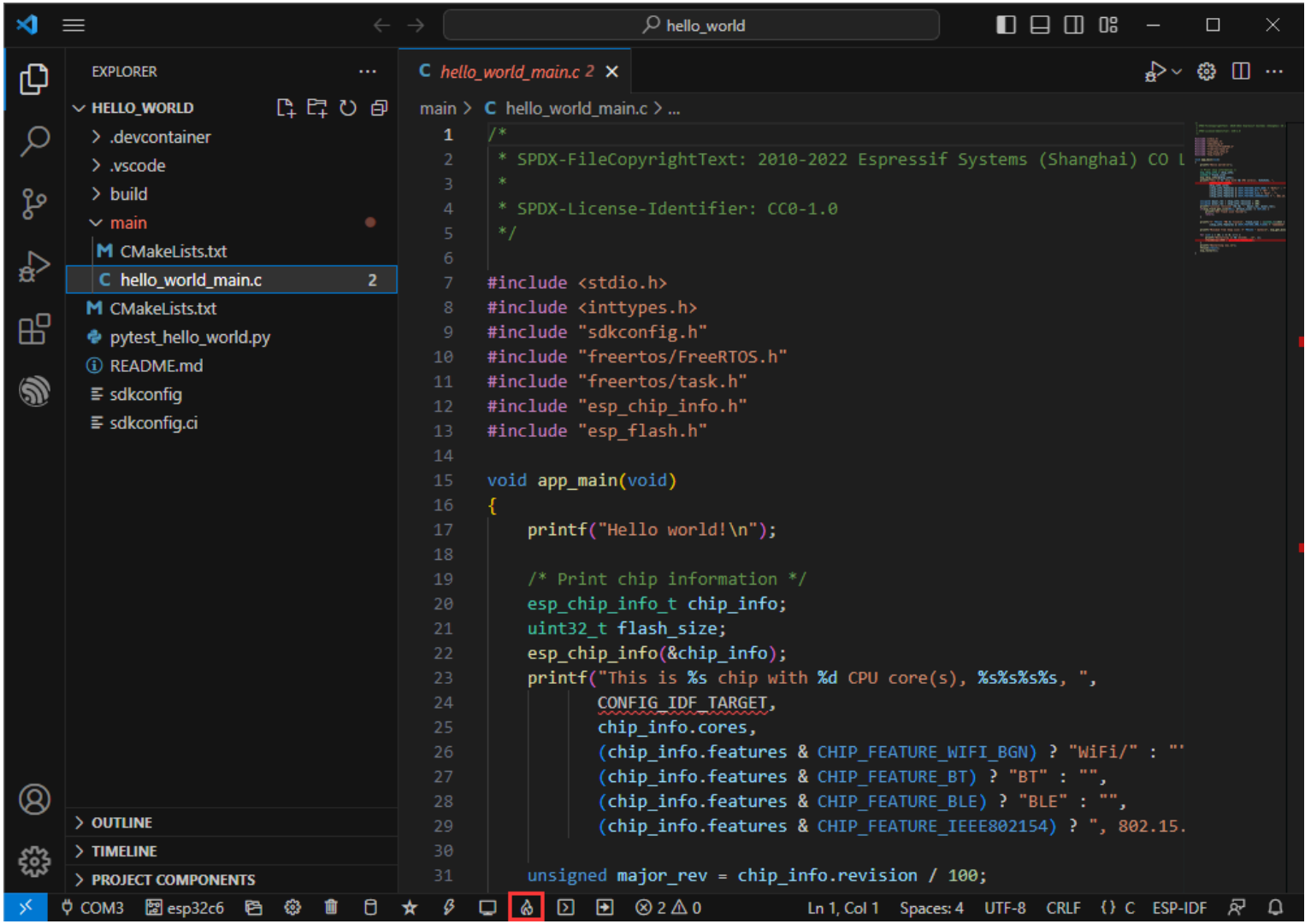
- ⑥Open the serial port monitor, which is used to view the serial port information.
- ⑦All-in-one button, compile, flash and open the serial monitor (most commonly used for debugging).



(/wiki/File:ESP32-C6-DEV-KIT-N8-24.png)

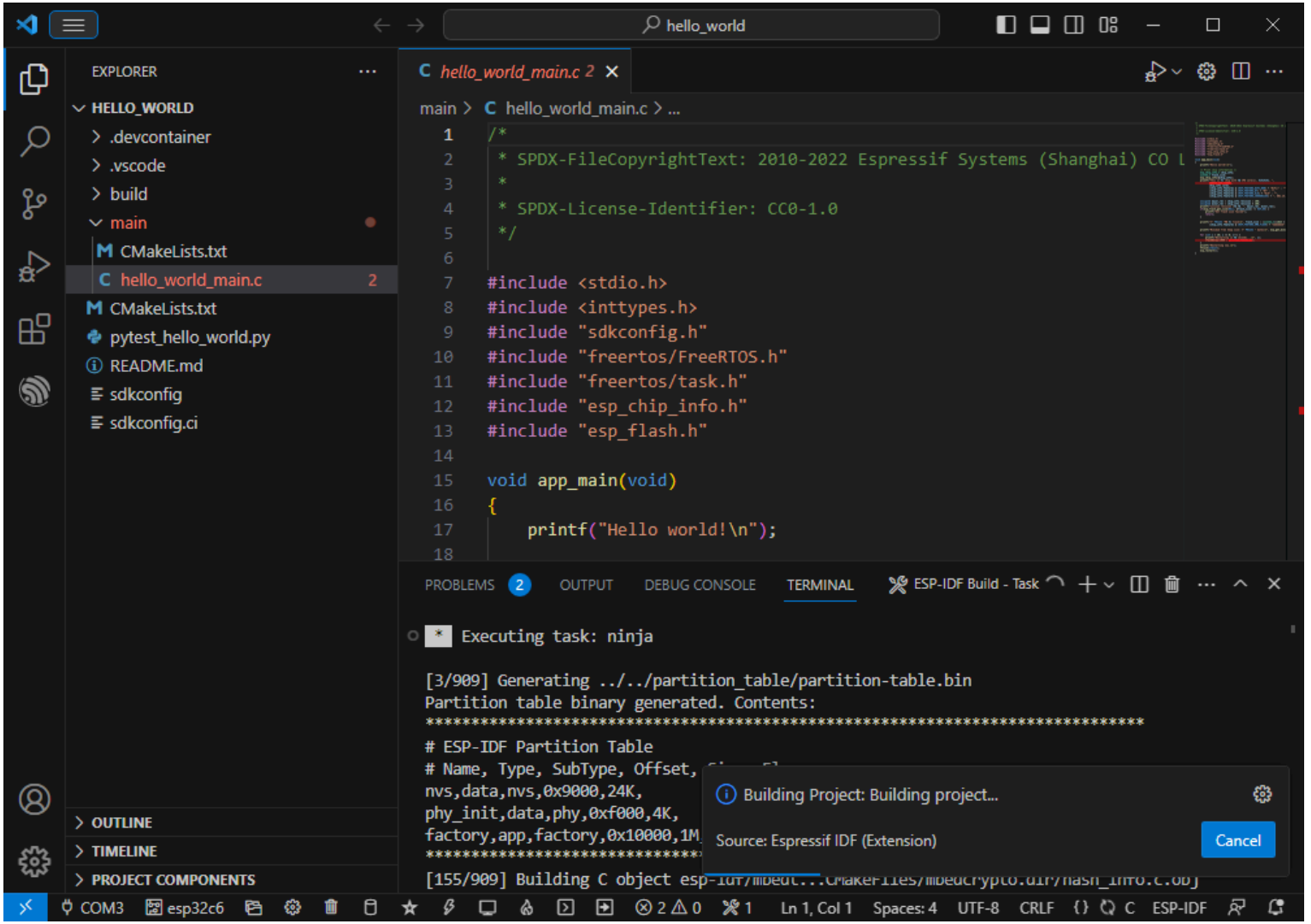
Compile, Program, Serial Port Monitoring

- Click on the all-in-one button we described before to compile, program and open the serial port monitor.



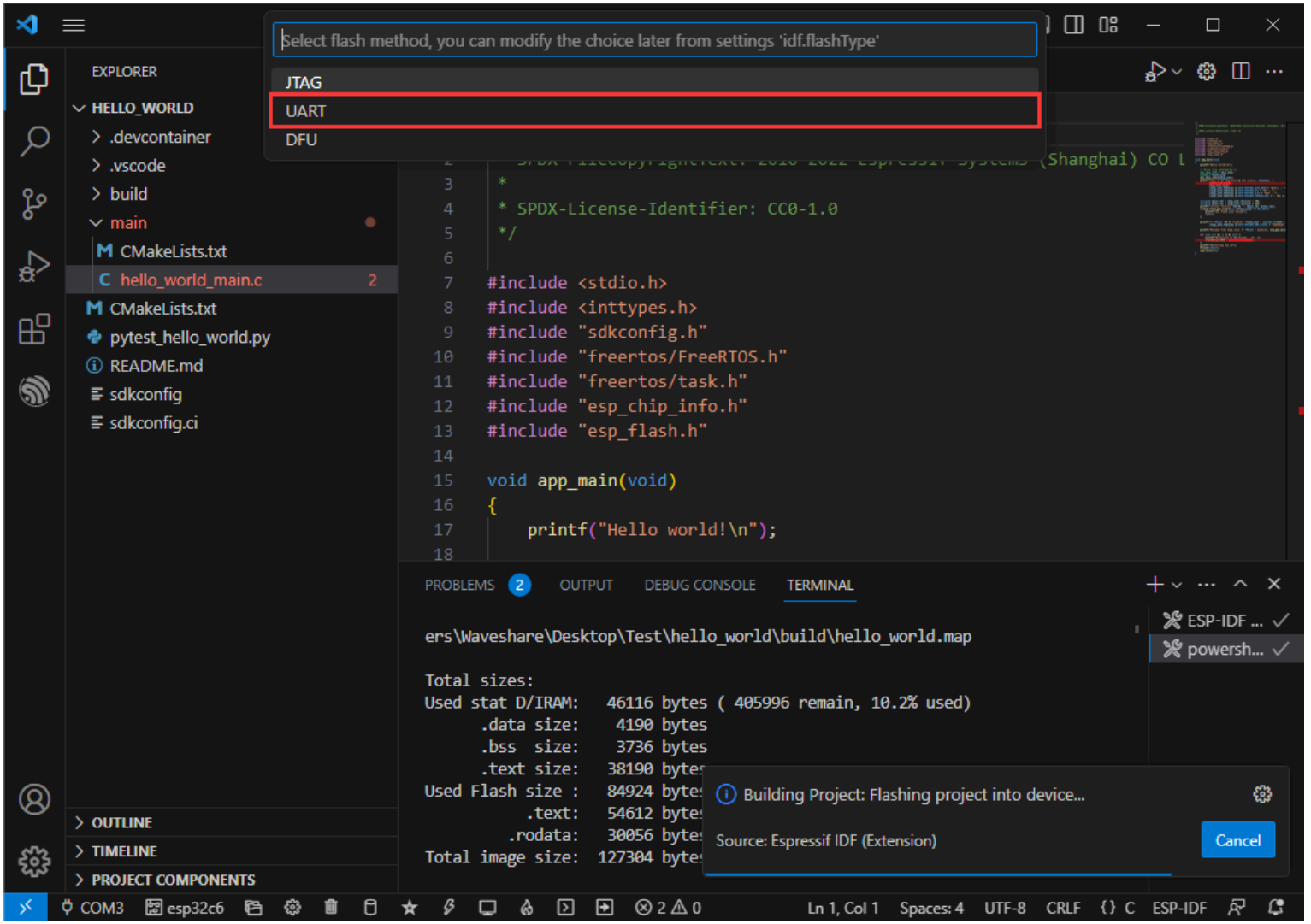
(/wiki/File:ESP32-C6-DEV-KIT-N8-25.png)

- It may take a long time to compile especially for the first time.



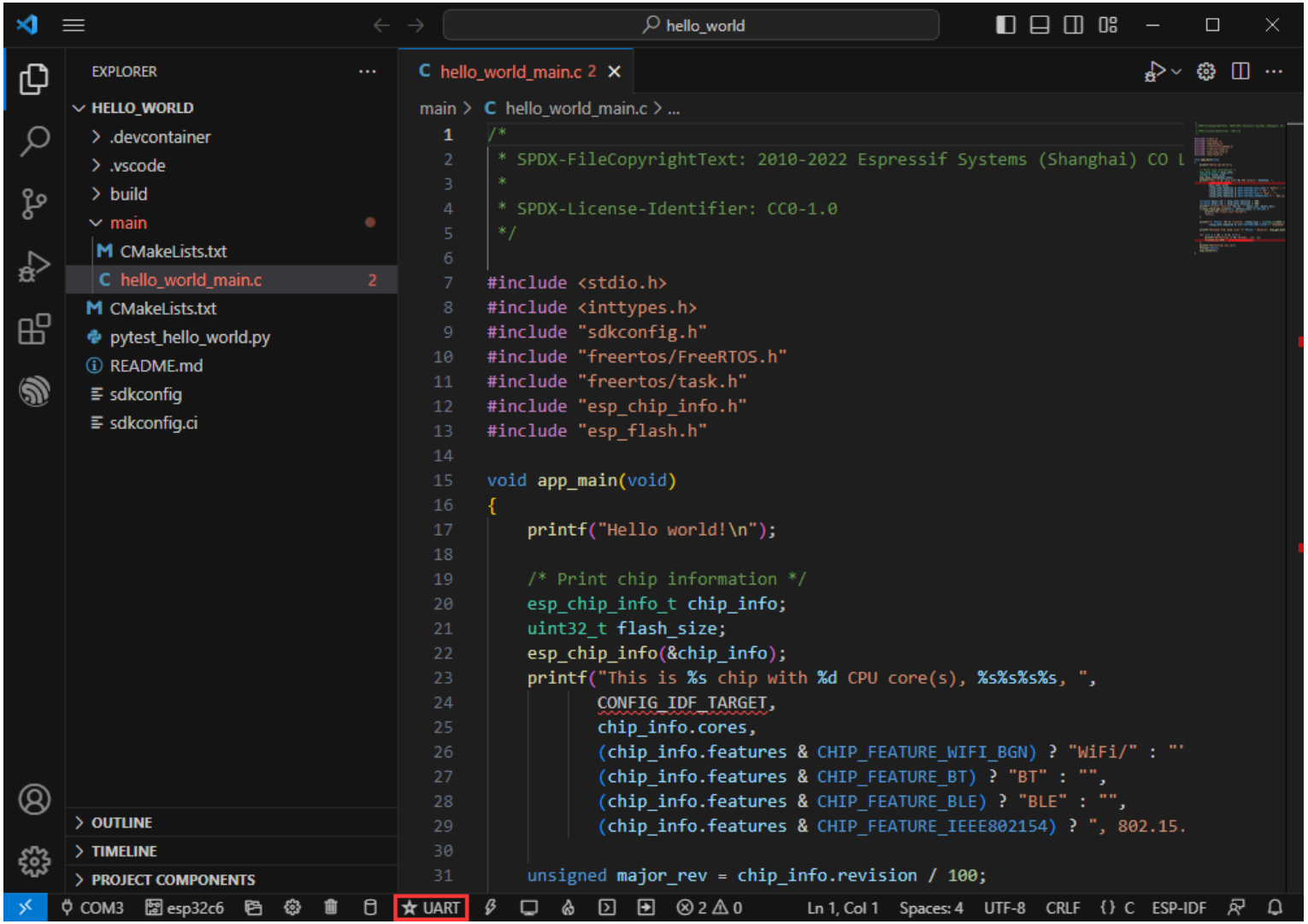
(/wiki/File:ESP32-C6-DEV-KIT-N8-26.png)

- **During this process, the ESP-IDF may take up a lot of CPU resources, so it may cause the system to lag.**
- Flashing the demo for a new project for the first time, you will need to select the download method, and select **UART**.



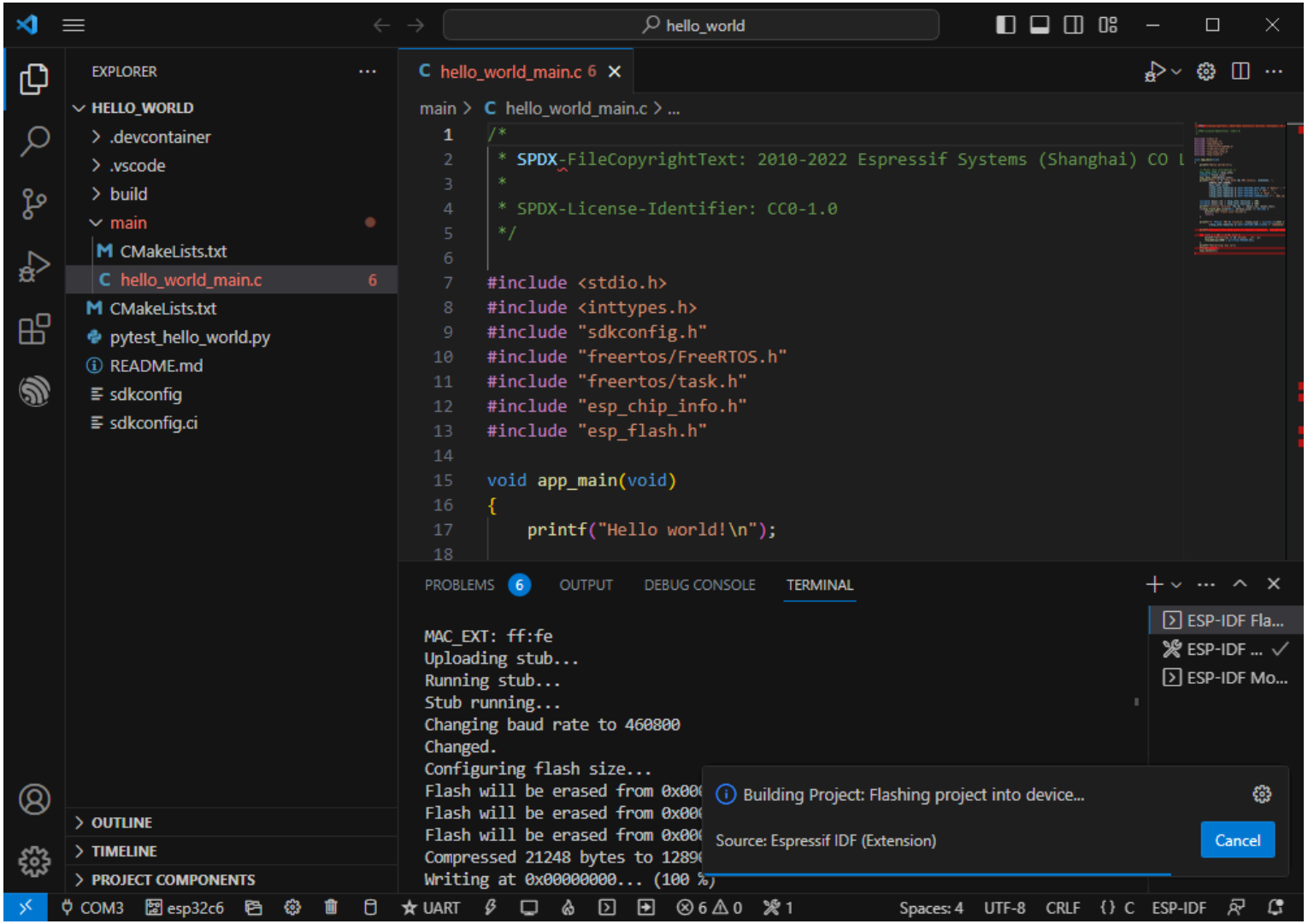
(/wiki/File:ESP32-C6-DEV-KIT-N8-27.png)

- This can also be changed later in the **Download Methods** section (click on it to bring up the options).



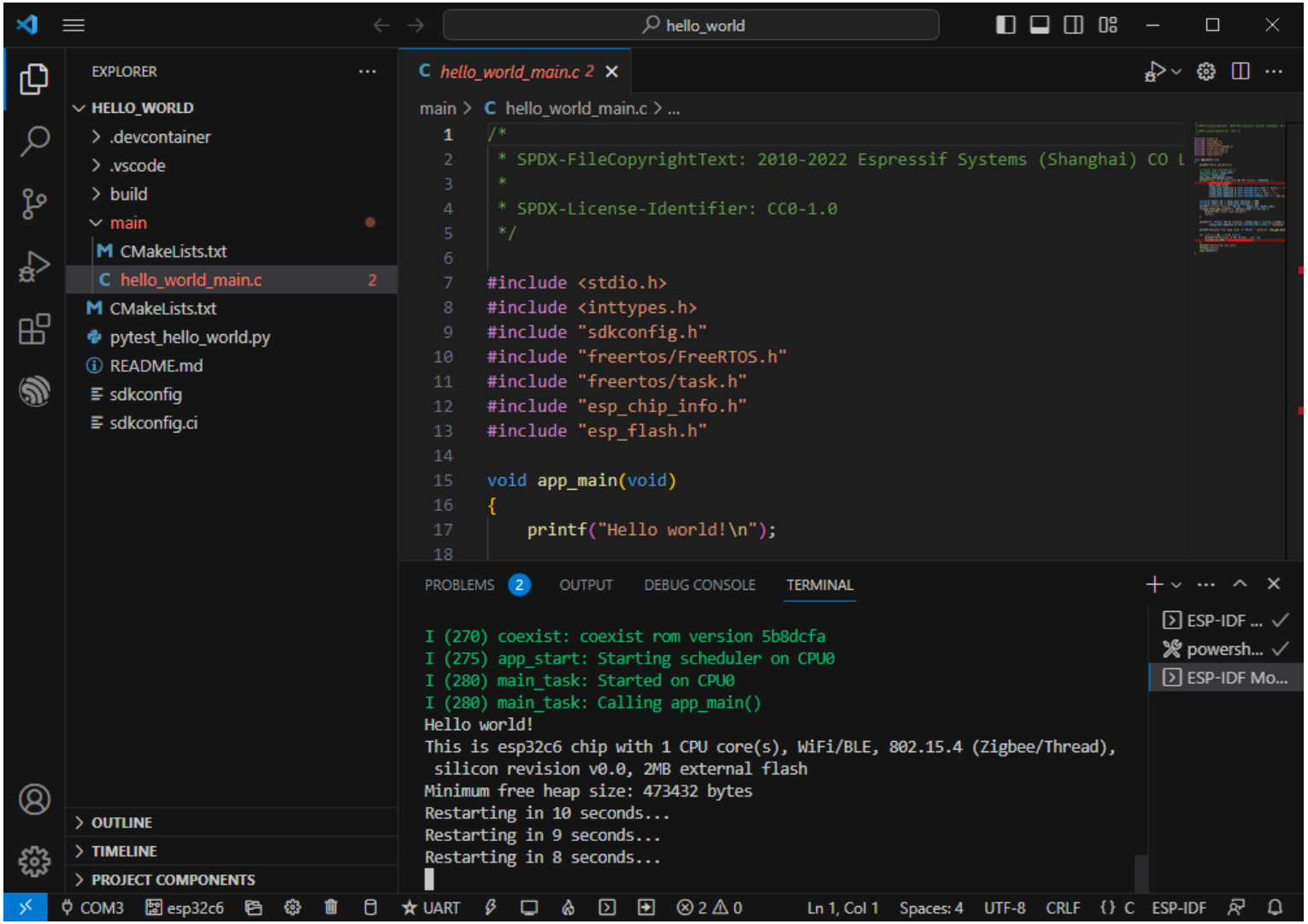
(/wiki/File:ESP32-C6-DEV-KIT-N8-28.png)

- As it comes with the onboard automatic download circuit, there is no need for manual operation to download automatically.



(/wiki/File:ESP32-C6-DEV-KIT-N8-29.png)

- After successful download, automatically enter the serial monitor, you can see the chip output the corresponding information and be prompted to restart after 10S.



(/wiki/File:ESP32-C6-DEV-KIT-N8-30.png)

Demo Demonstration

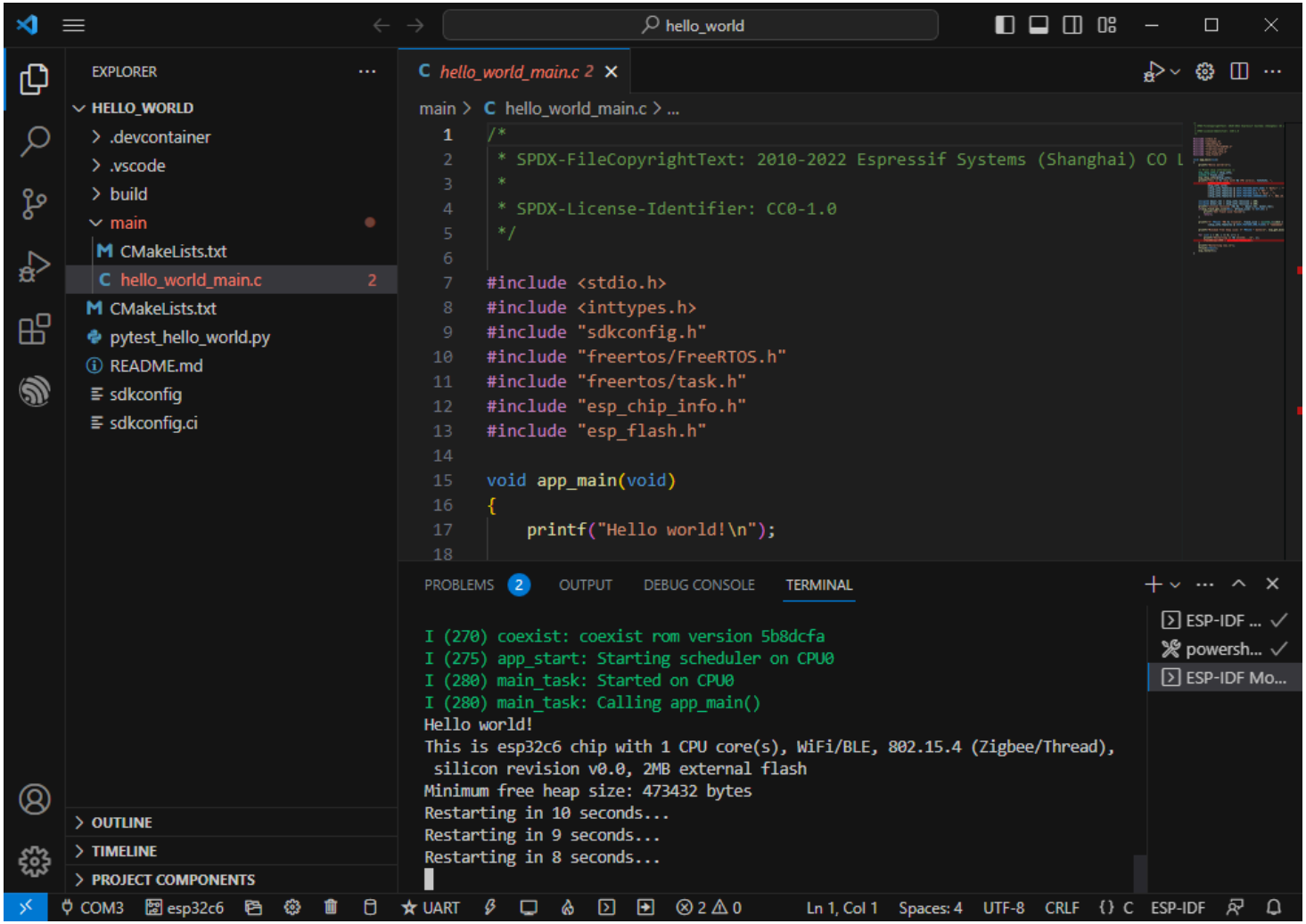
Hello World

The official example path: get-started -> hello_world.

The example effect: Output **Hello World!** on the **TERMINAL** window every 10s.

Software Operation

- Create the official example "hello_world" according to the above tutorial. (Create Example)
- The demo is compatible with ESP32-C6, and you can directly use it with no need to modify the demo.
- Modify the COM port and the driver object, click on the compile and flash to run the demo.



(/wiki/File:ESP32-C6-DEV-KIT-N8-30.png)

RGB

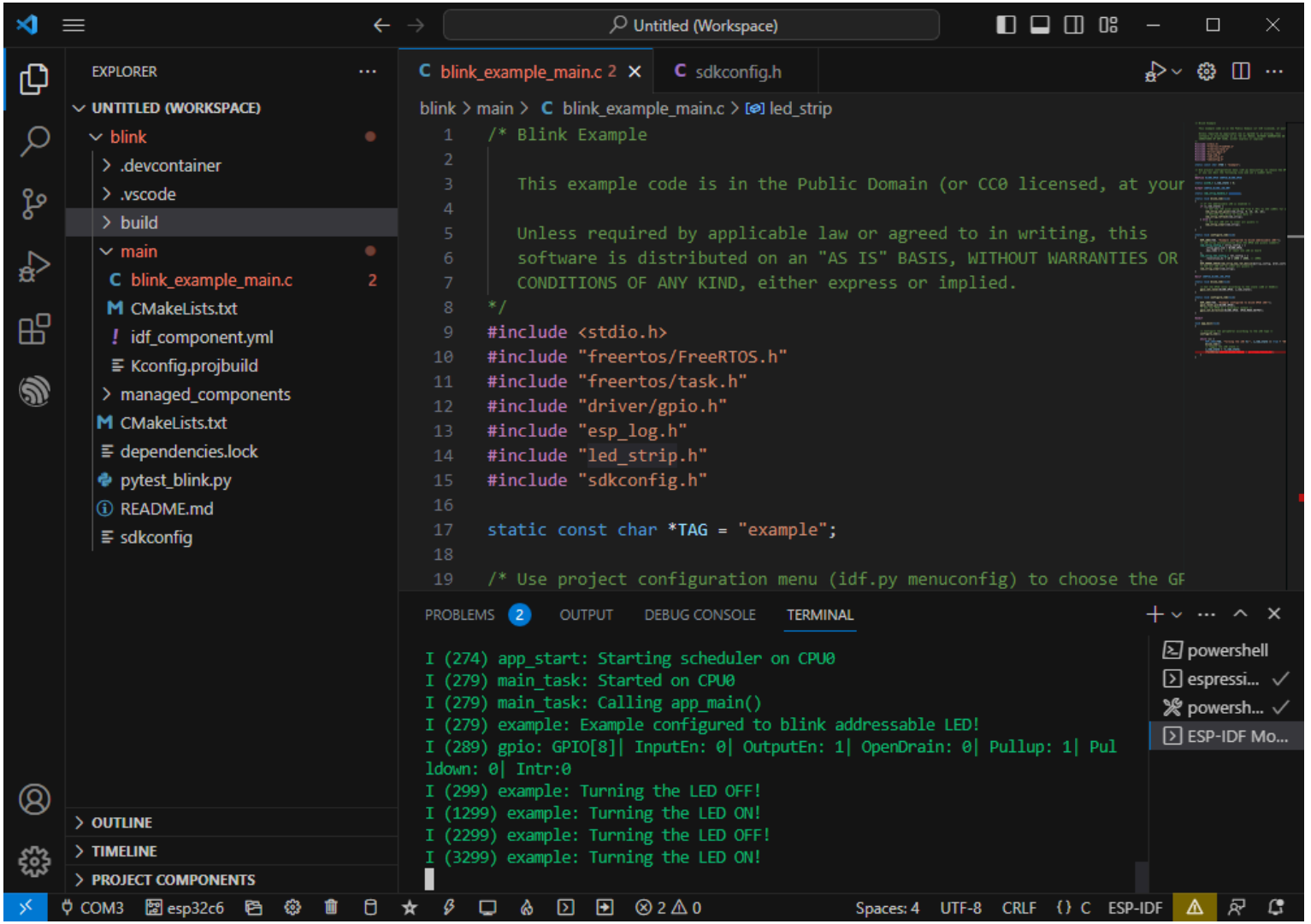
Official example path: get-started -> blink.

Example effect: onboard RGB beads blink at 1-second intervals.

Software Operation

- Follow the tutorial above to create the official example blink. (Create Example)

- The demo is compatible with ESP32-C6 and can be used without modifying the demo content.
- Modify the COM port and the driver object **(It is recommended to prioritize the use of the COM port corresponding to USB (can be viewed through the device manager))**, click compile and flash to run the demo.



(/wiki/File:ESP32-C6-DEV-KIT-N8-41.png)

UART

Official example path: peripherals -> uart-> uart_async_rxtxtasks.

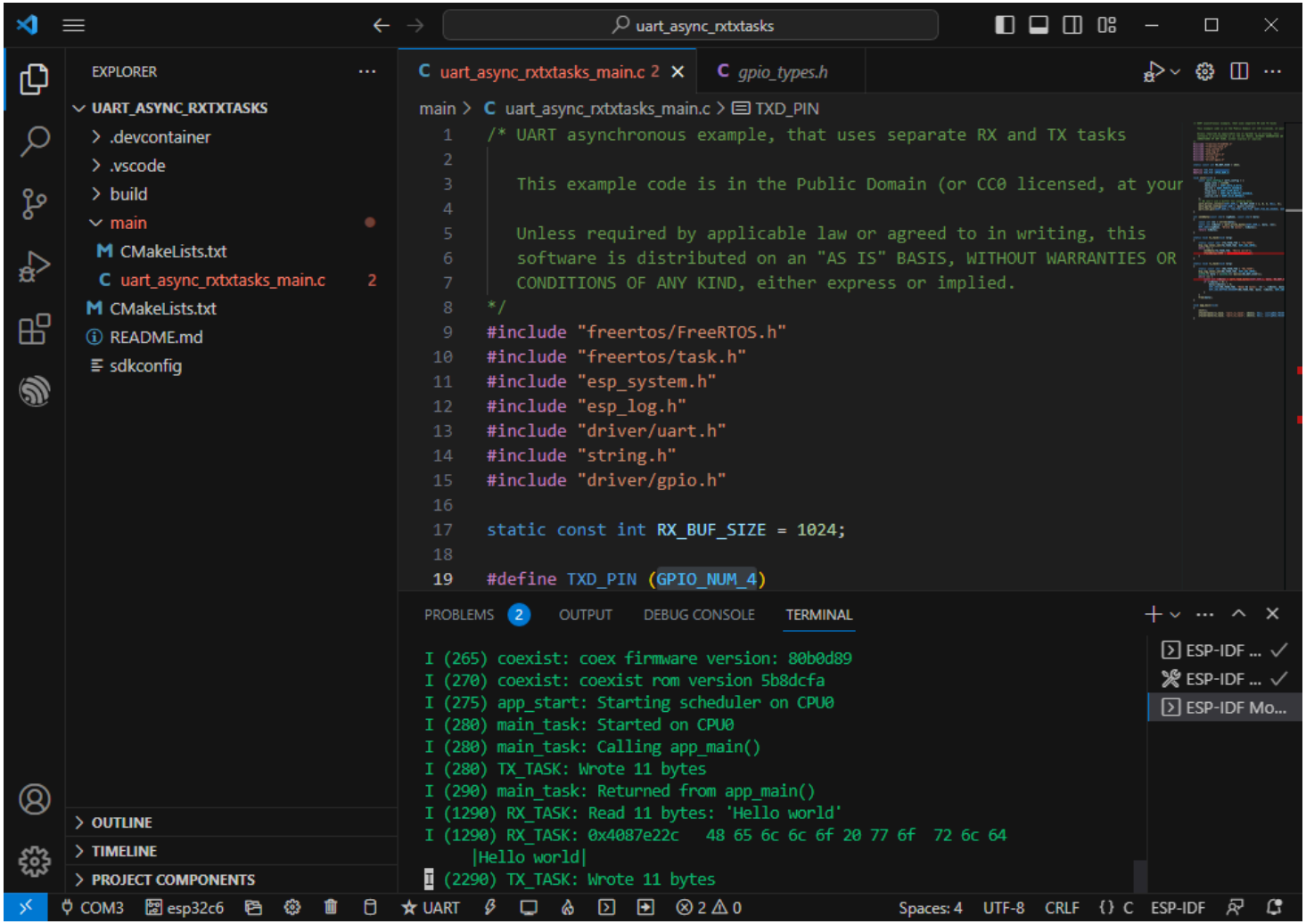
Example effect: shorting GPIO4 and GPIO5 to send/receive UART data.

Hardware Connection

ESP32-C6	ESP32-C6 (the same one)
GPIO4	GPIO5

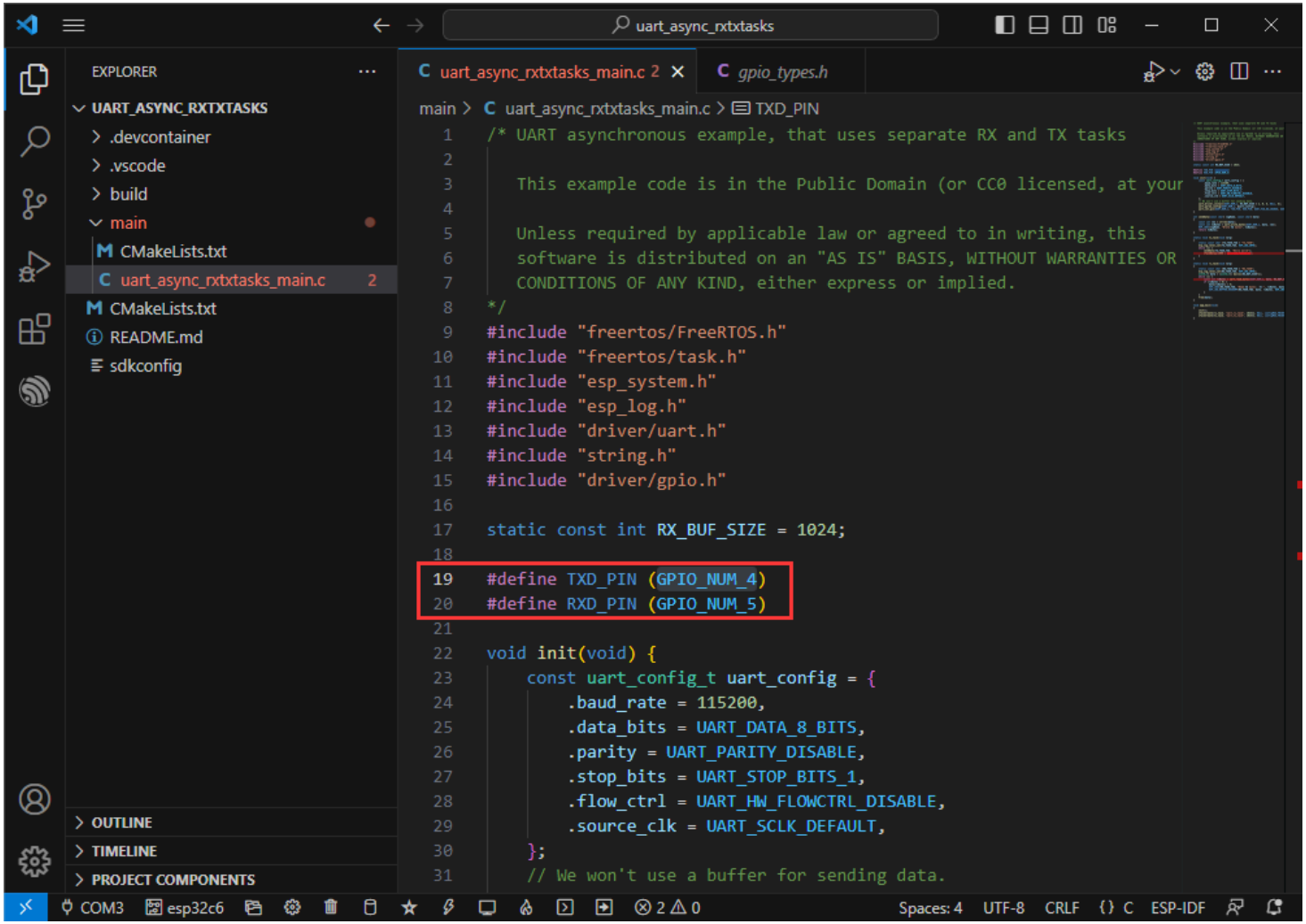
Software Operation

- Create the official example uart_async_rxtxtasks according to the tutorial above. (Create Example).
- The demo is compatible with ESP32-C6 and can be used without modifying the demo content.
- Modify the COM port and driver object, **(It is recommended to prioritize the use of the COM port corresponding to USB (can be viewed through the device manager)).** click compile and flash to run the demo.



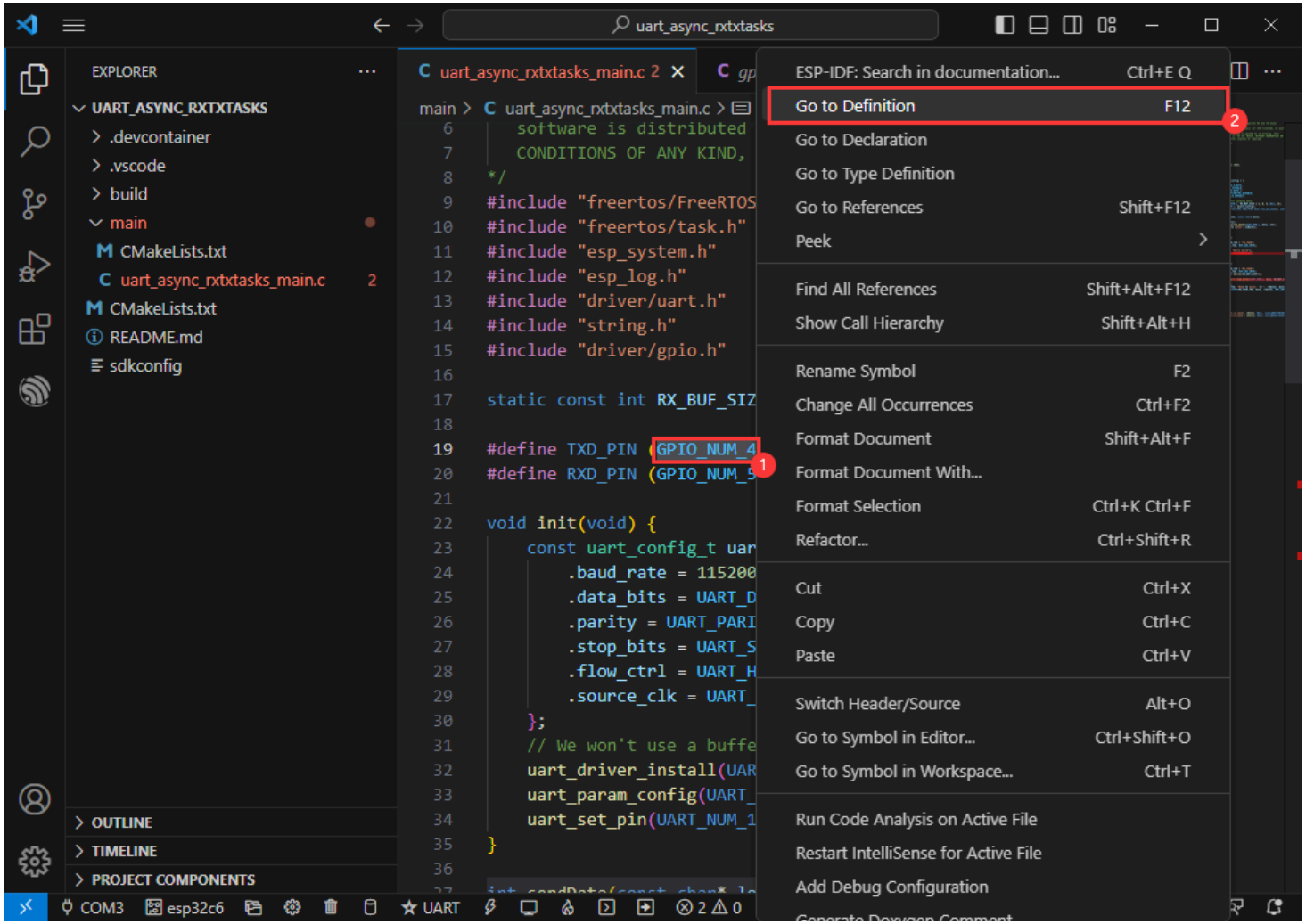
(/wiki/File:ESP32-C6-DEV-KIT-N8-77.png)

- Hardware connection according to the GPIO used.



(/wiki/File:ESP32-C6-DEV-KIT-N8-78.png)

- You can go to the definition file to see the actual GPIOs used (check **GPIO_NUM_4** -> Right click -> **Go to Definition**).



(/wiki/File:ESP32-C6-DEV-KIT-N8-79.png)

Bluetooth

Official sample path: bluetooth -> bluedroid -> ble -> gatt_server.

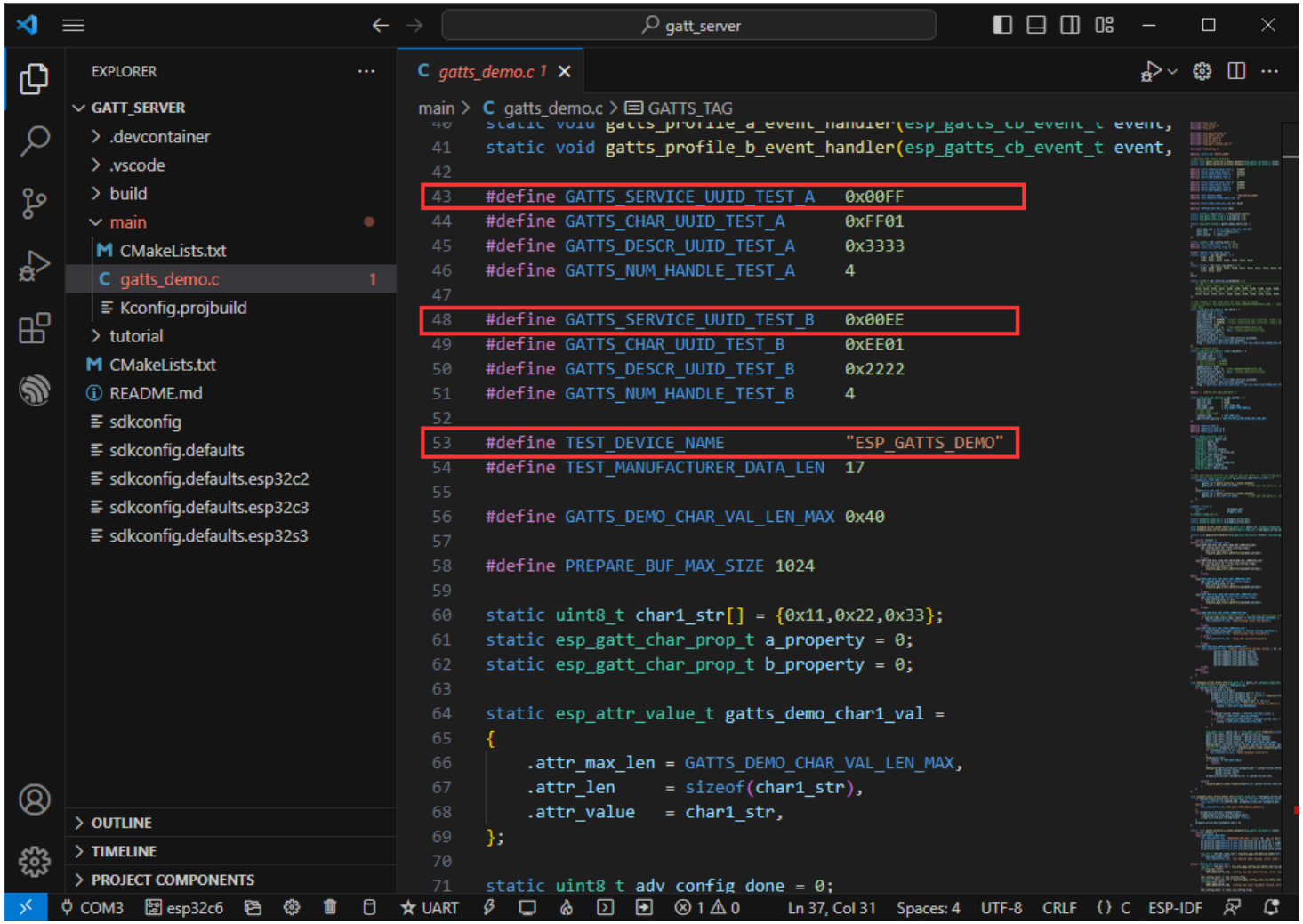
Example effect: ESP32-C6 and cell phone Bluetooth debugging assistant for data transmission.

Software Operation

- Install the Bluetooth debugging assistant (https://files.waveshare.com/wiki/ESP32-C6-DEV-KIT

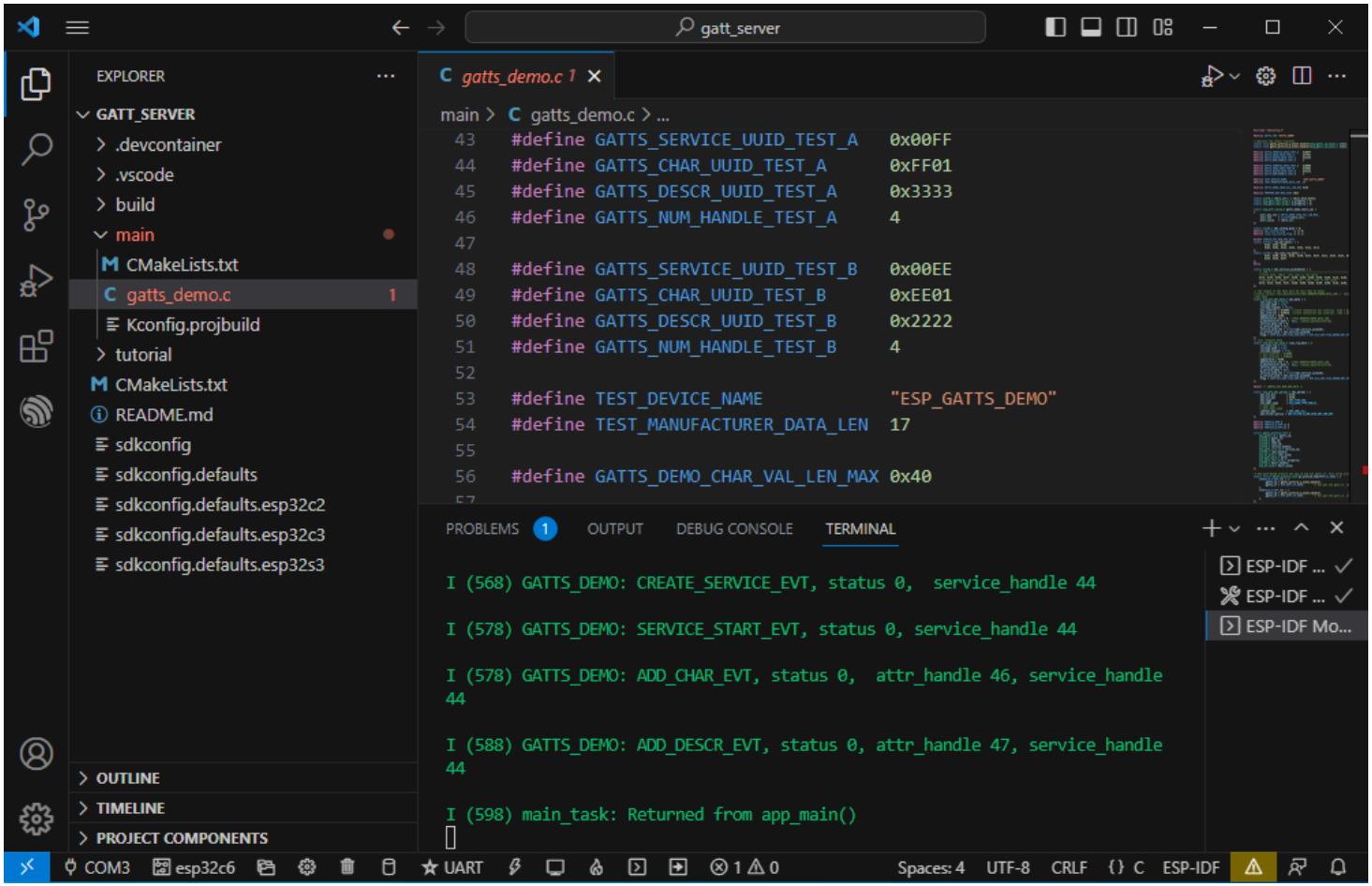
-N8/ESP32-C6_TO_BLEAssist.ZIP) on your phone.

- Follow the tutorial above to create the official example gatt_server. (Create Example)
- The demo is compatible with ESP32-C6 and can be used without modifying the demo content.
- Bluetooth name and UUID, Bluetooth name is **ESP_GATTS_DEMO**.



(/wiki/File:ESP32-C6-DEV-KIT-N8-50.png)

- Modify the COM port and the driver object, **(It is recommended to prioritize the use of the COM port corresponding to USB (can be viewed through the device manager)).** click on the compile and flash to run the demo.



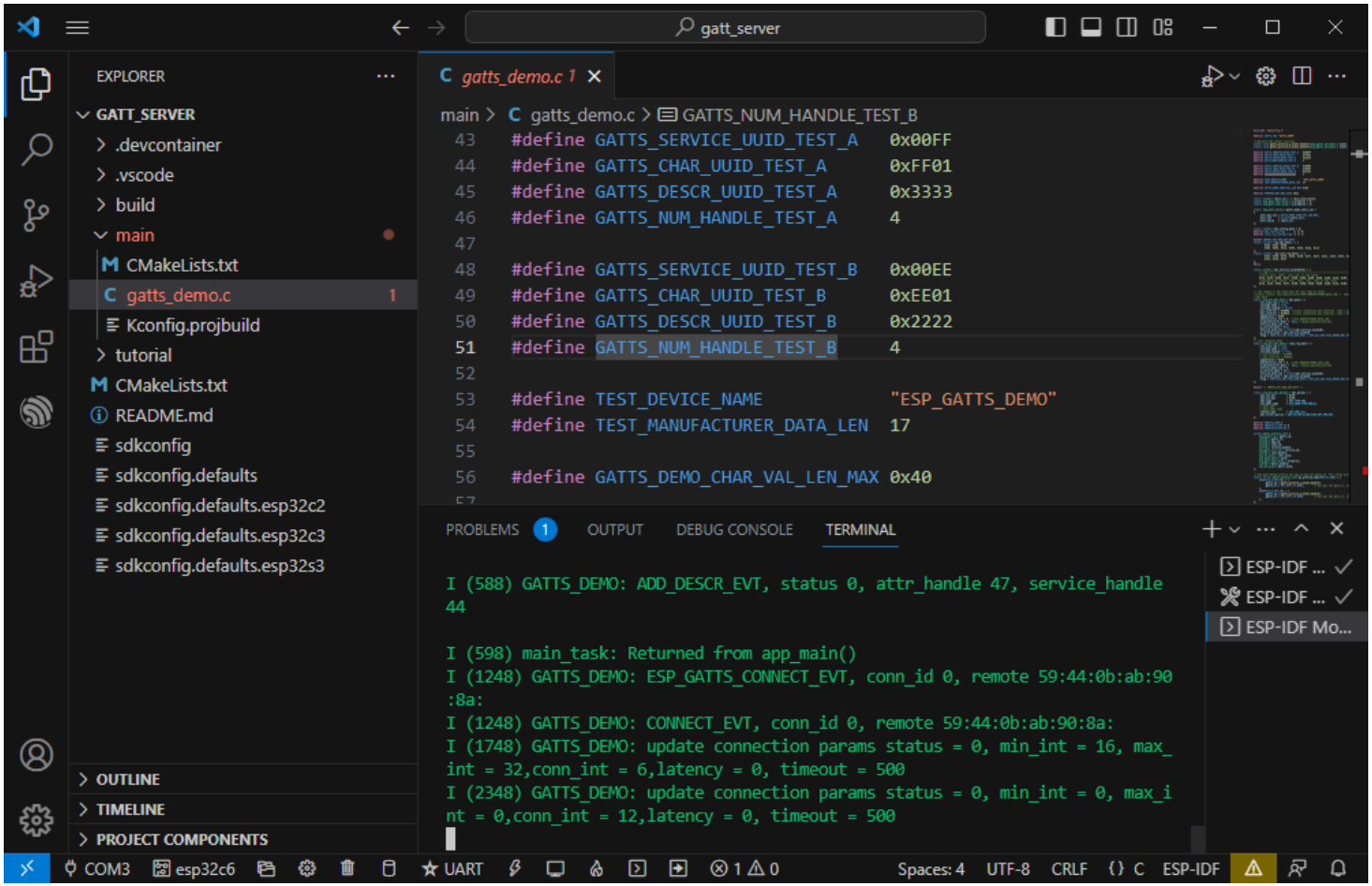
(/wiki/File:ESP32-C6-DEV-KIT-N8-51.png)

- Connecting the ESP_GATTS_DEMO Bluetooth device on the phone.


BLE调试助手
Scanner
Bonded
N/A
39:54:CE:BA:AA:F9
NOT BONDED -97 dBm
ESP_GATTS_DEMO
40:4C:CA:45:FE:46
NOT BONDED -27 dBm
CONNECT
N/A
58:D5:3E:6B:B4:D2
NOT BONDED -93 dBm
N/A
31:8F:08:40:2C:8D
NOT BONDED -71 dBm
N/A
32:8C:86:B9:40:3F
NOT BONDED -91 dBm
N/A
23:77:42:FC:F4:DC
NOT BONDED -94 dBm
N/A
35:45:3D:36:F7:3B
NOT BONDED -84 dBm
ESP_GATTS_DE...
断开连接
Services
Log
Generic Attribute
UUID: 00001801-0000-1000-8000-00805f9b34fb
PRIMARY SERVICE
Generic Access
UUID: 00001800-0000-1000-8000-00805f9b34fb
PRIMARY SERVICE
Unknown Service
UUID: 000000ff-0000-1000-8000-00805f9b34fb
PRIMARY SERVICE
Unknown Service
UUID: 000000ee-0000-1000-8000-00805f9b34fb
PRIMARY SERVICE
蓝牙连接成功

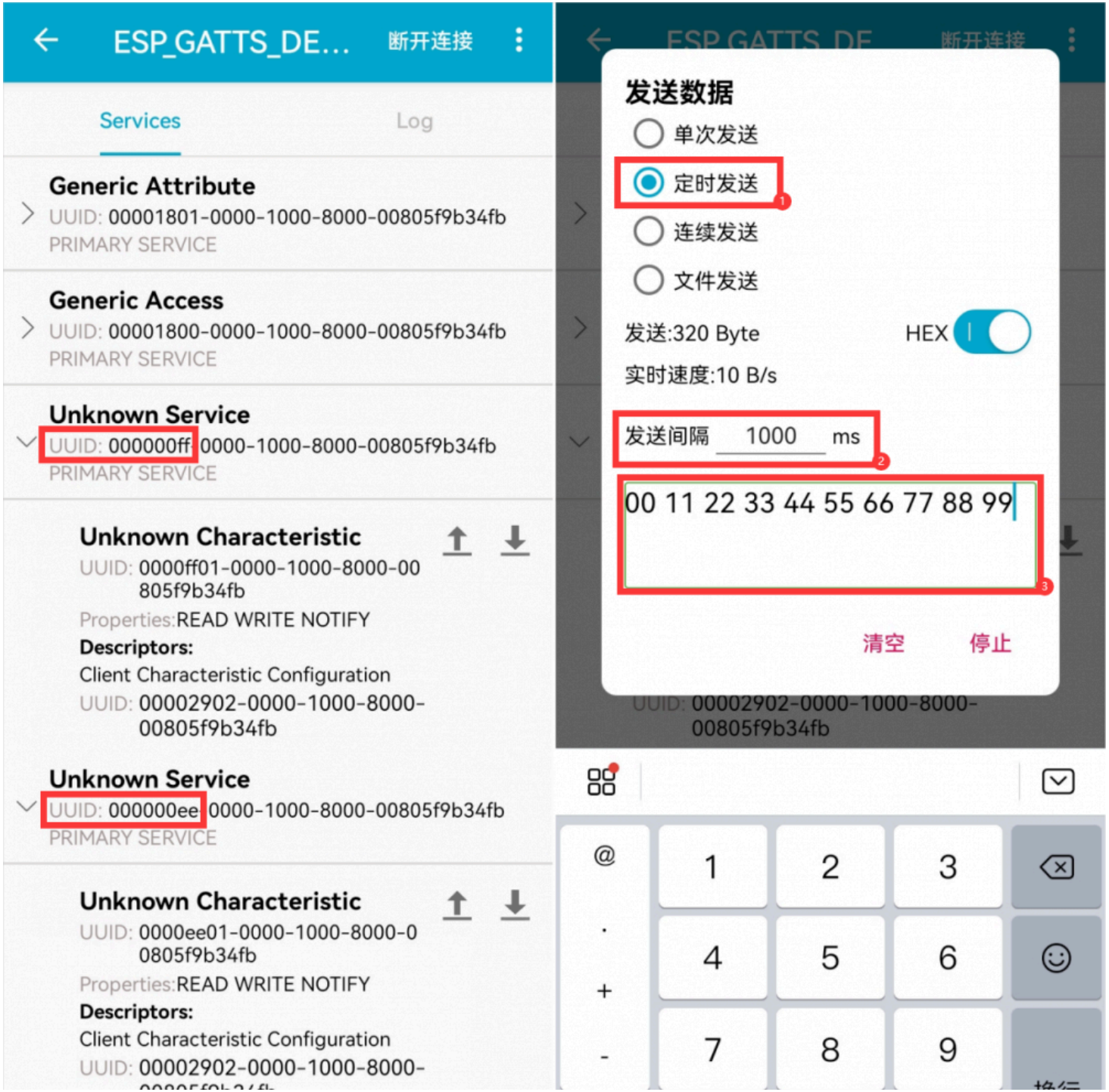
(/wiki/File:1000px-ESP32-C6_TO_Sample_27.png)

- The effect of a successful connection is shown below:



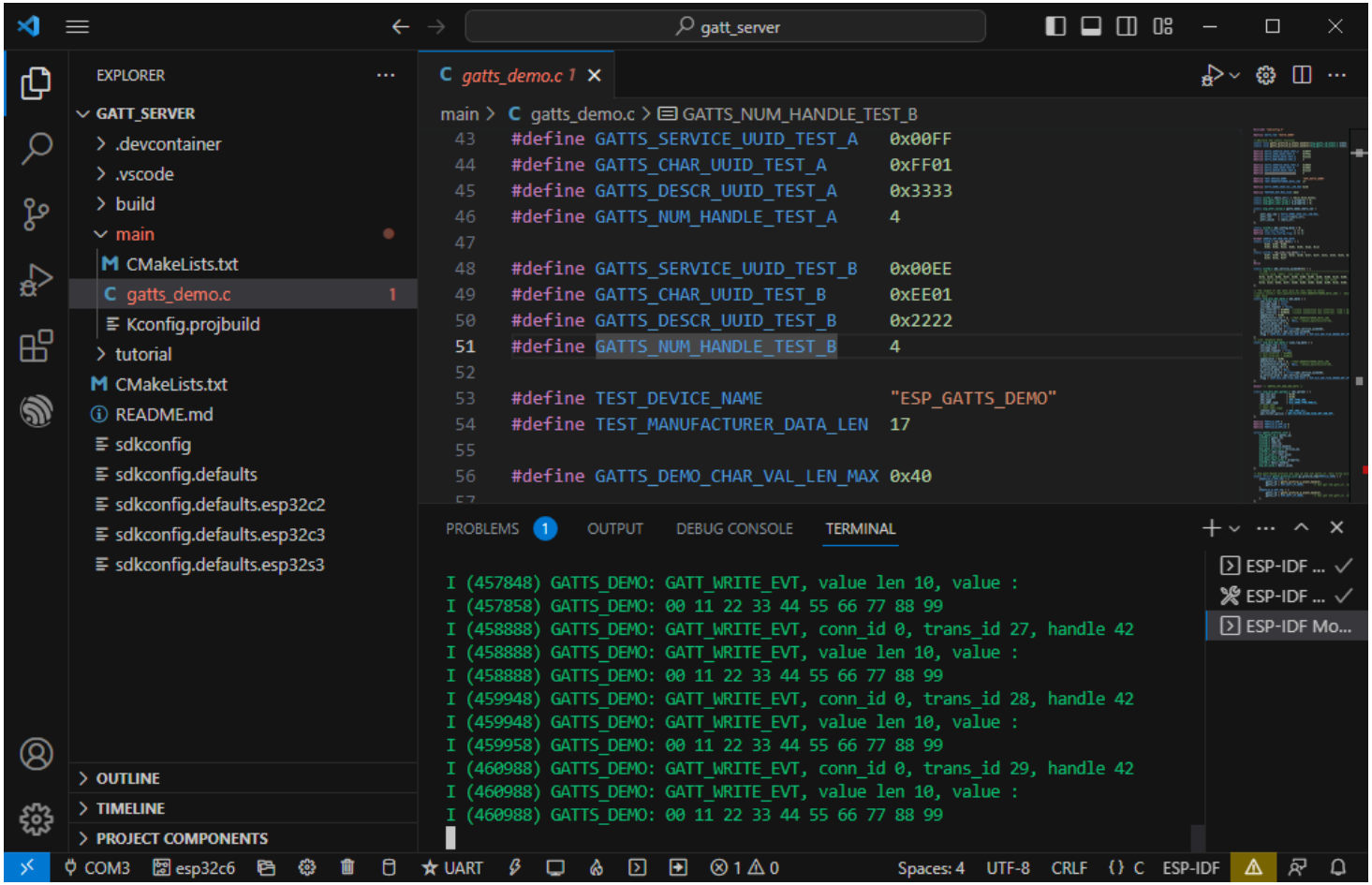
(/wiki/File:ESP32-C6-DEV-KIT-N8-52.png)

- Based on the UUID value in the demo, select one of the two servers for upstream transmission.


ESP_GATTS_DE...
断开连接
Services
Log
Generic Attribute
UUID: 00001801-0000-1000-8000-00805f9b34fb
PRIMARY SERVICE
Generic Access
UUID: 00001800-0000-1000-8000-00805f9b34fb
PRIMARY SERVICE
Unknown Service
UUID: 000000ff-0000-1000-8000-00805f9b34fb
PRIMARY SERVICE
Unknown Characteristic
UUID: 0000ff01-0000-1000-8000-00805f9b34fb
Properties:READ WRITE NOTIFY
Descriptors:
Client Characteristic Configuration
UUID: 00002902-0000-1000-8000-00805f9b34fb
Unknown Service
UUID: 000000ee-0000-1000-8000-00805f9b34fb
PRIMARY SERVICE
Unknown Characteristic
UUID: 0000ee01-0000-1000-8000-00805f9b34fb
Properties:READ WRITE NOTIFY
Descriptors:
Client Characteristic Configuration
UUID: 00002902-0000-1000-8000-
发送数据
单次发送
定时发送
连续发送
文件发送
发送:320 Byte
HEX
实时速度:10 B/s
发送间隔 1000 ms
00 11 22 33 44 55 66 77 88 99
清空
停止
UUID: 00002902-0000-1000-8000-00805f9b34fb
@
1
2
3
.
4
5
6
+
-
7
8
9

(/wiki/File:1000px-ESP32-C6_TO_Sample_29.png)

- The ESP32-C6 receives data.



(/wiki/File:ESP32-C6-DEV-KIT-N8-54.png)

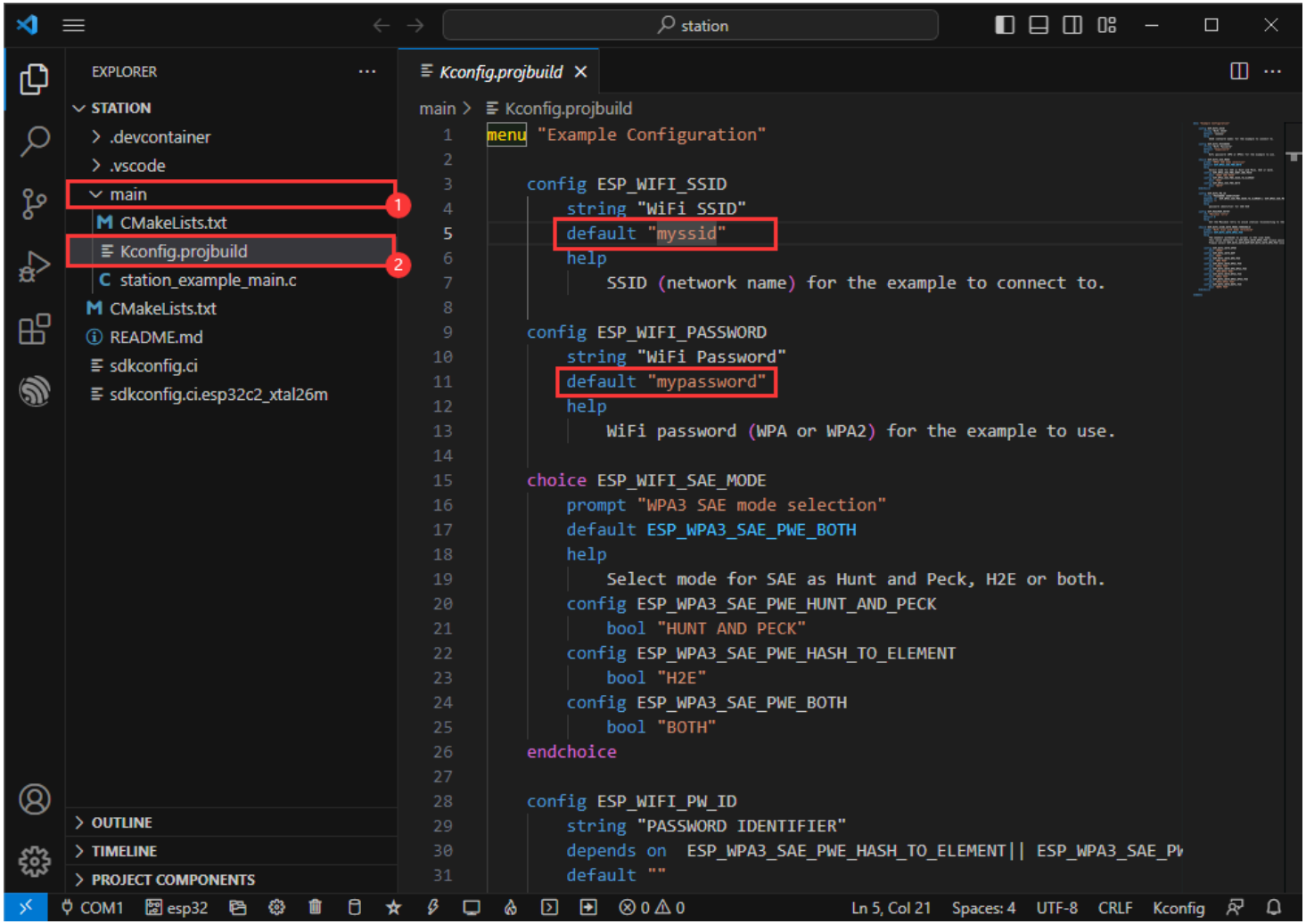
WIFI

Official example path: wifi -> getting_started -> station.

Example effect: ESP32-C6 connects to WIFI.

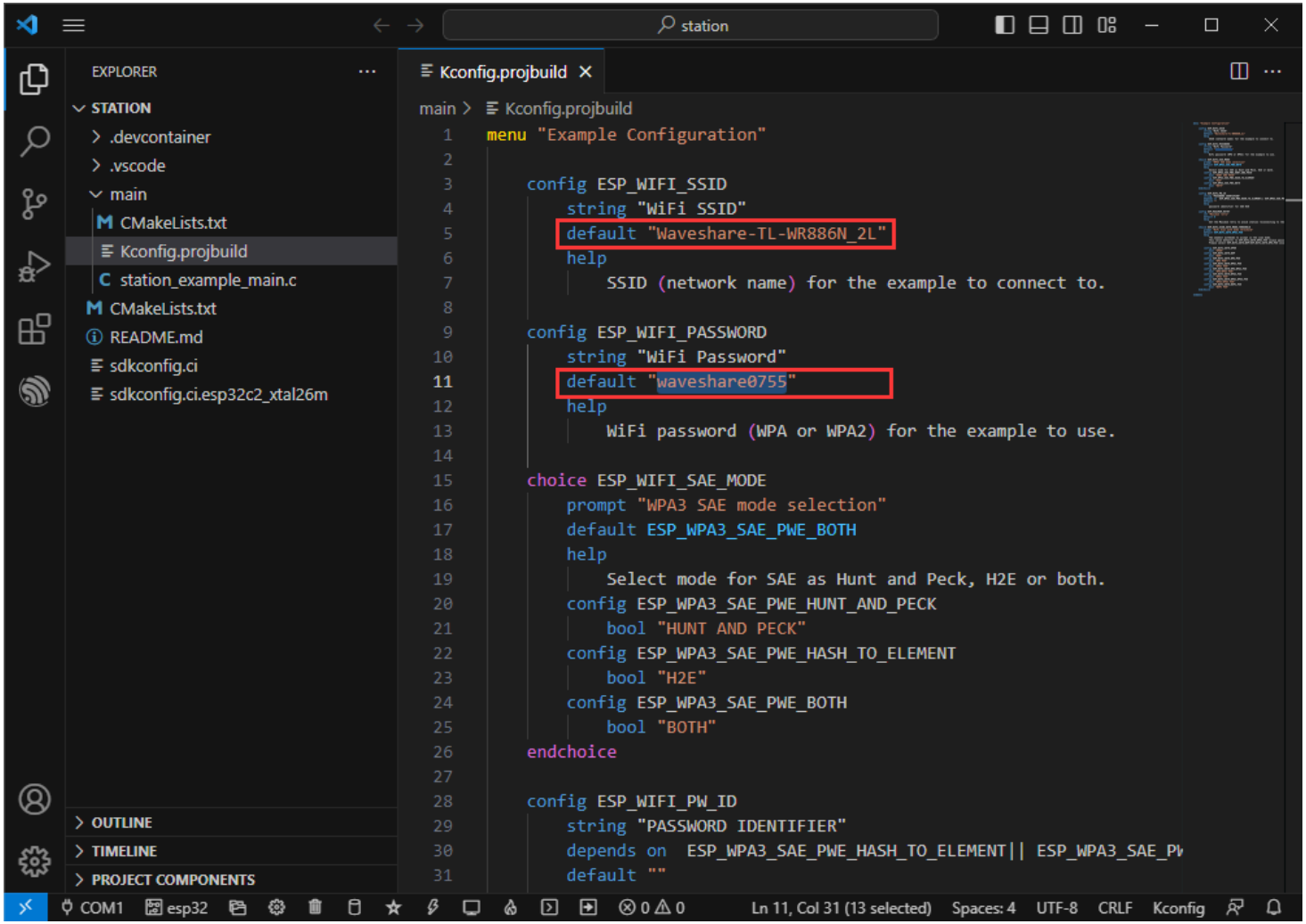
Software Operation

- Create the official example station according to the tutorial above. (Create Example)
- Modify the contents of the demo to connect to the required WiFi.
- Go to the **Kconfig.projbuild** file.



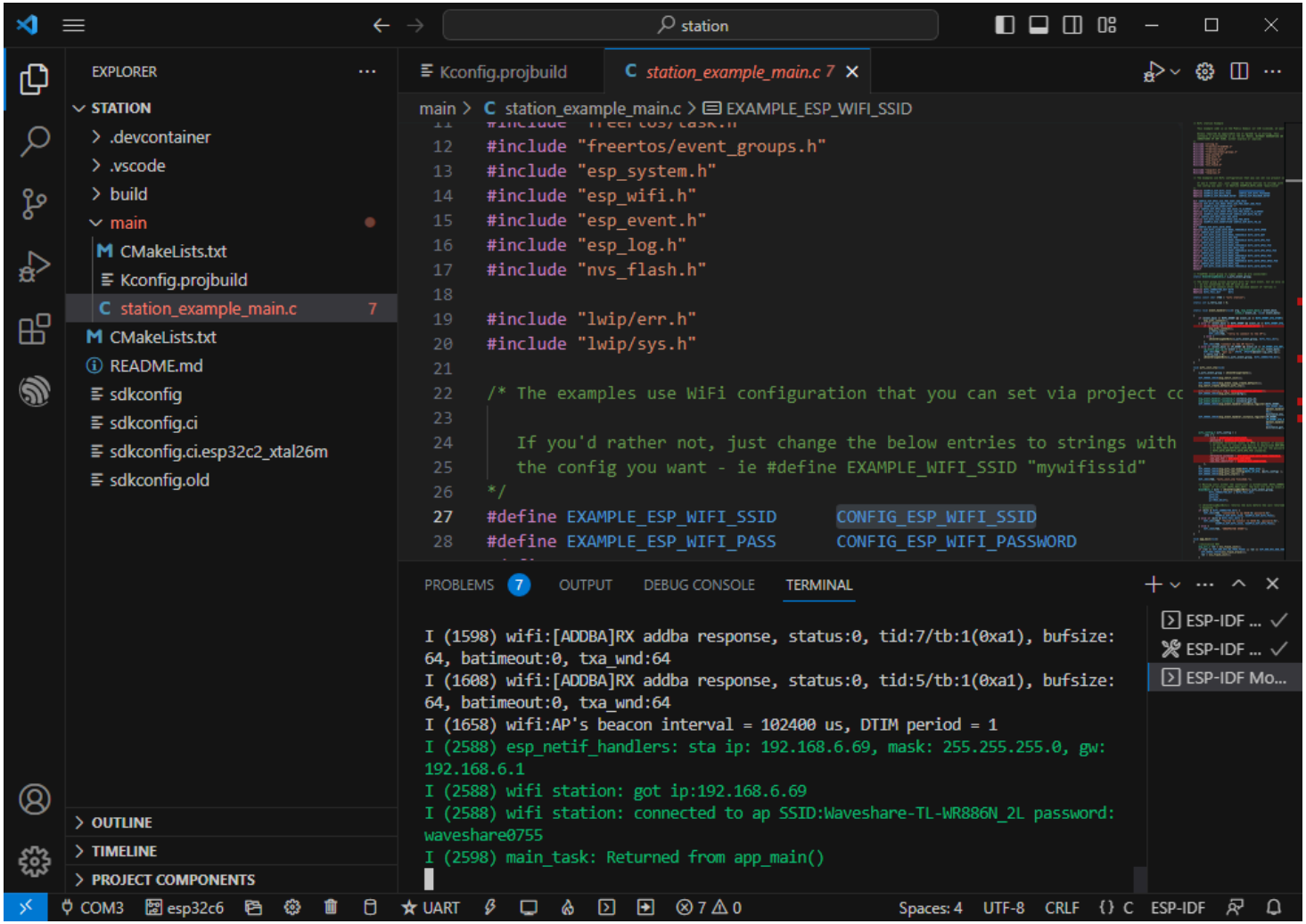
(/wiki/File:ESP32-C6-DEV-KIT-N8-60.png)

- Change the original **WiFi SSID** and **WiFi Password** to the WiFi information you want to connect to.



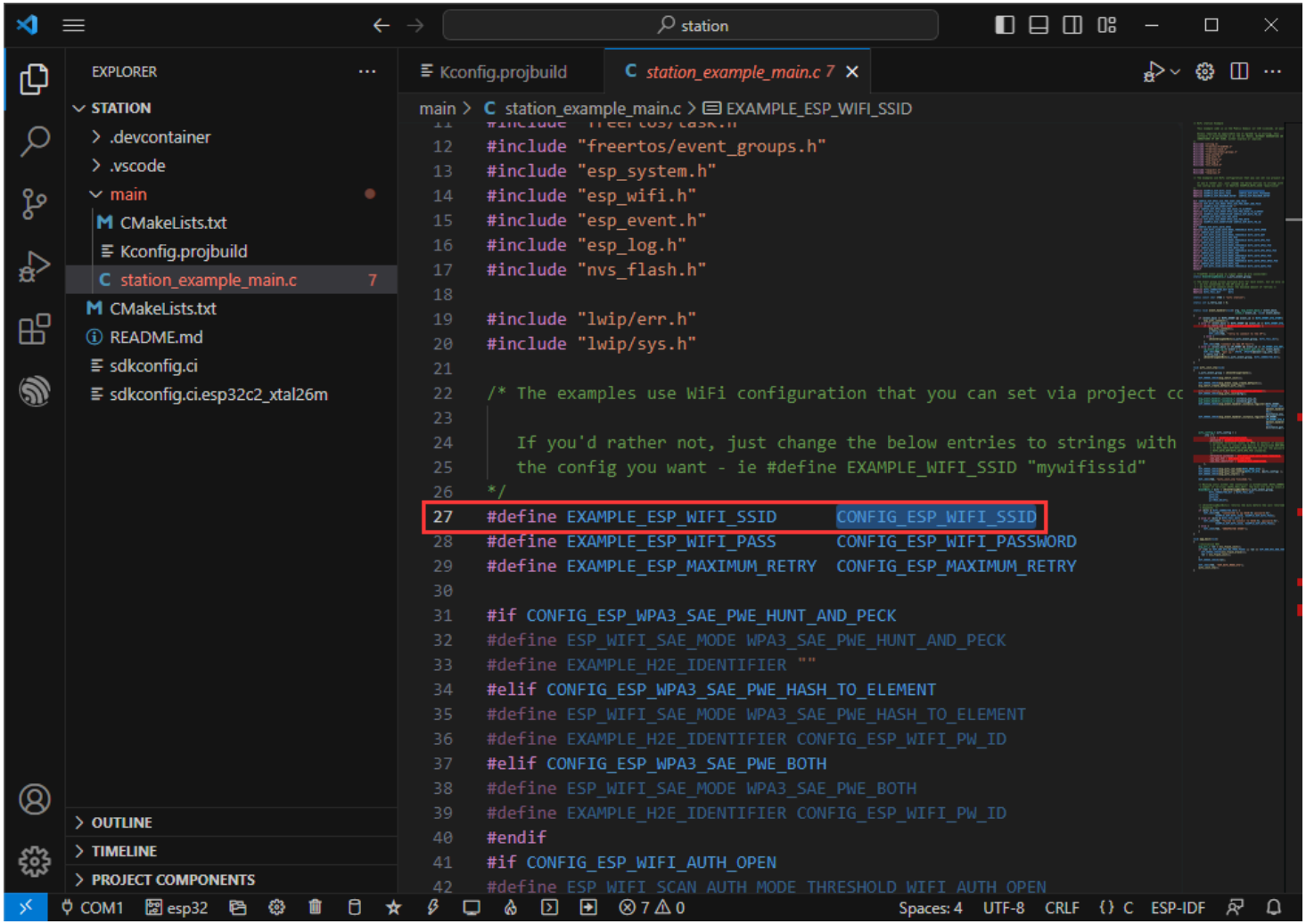
(/wiki/File:ESP32-C6-DEV-KIT-N8-61.png)

- Modify the COM port and the driver object. **It is recommended to prioritize the use of the COM port corresponding to USB (can be viewed through the device manager)).** click on the compile and flash to run the demo.



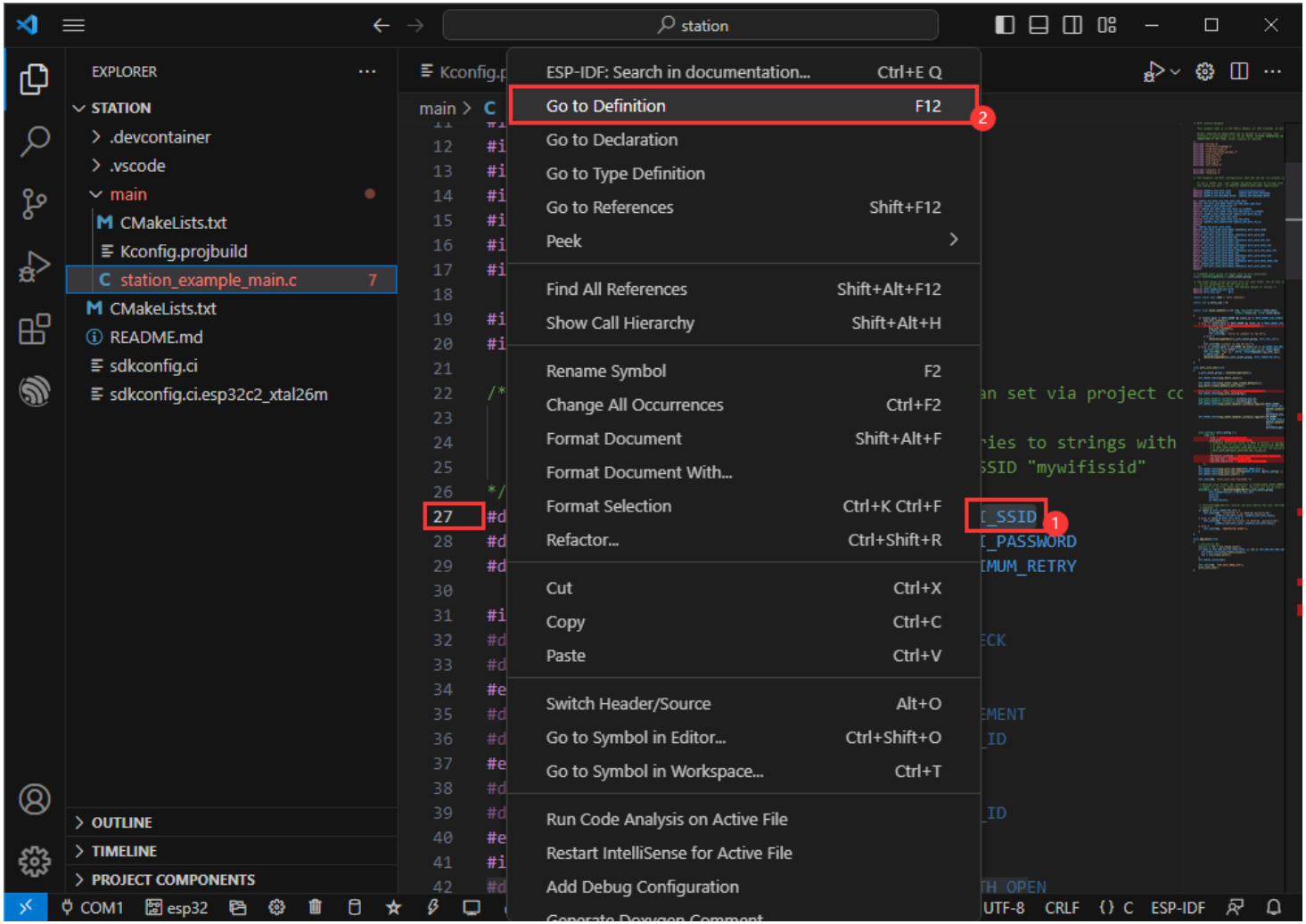
(/wiki/File:ESP32-C6-DEV-KIT-N8-62.png)

- You can check the value of **CONFIG_ESP_WIFI_SSID**.
- Go to the **station_ example_ main.c** file.



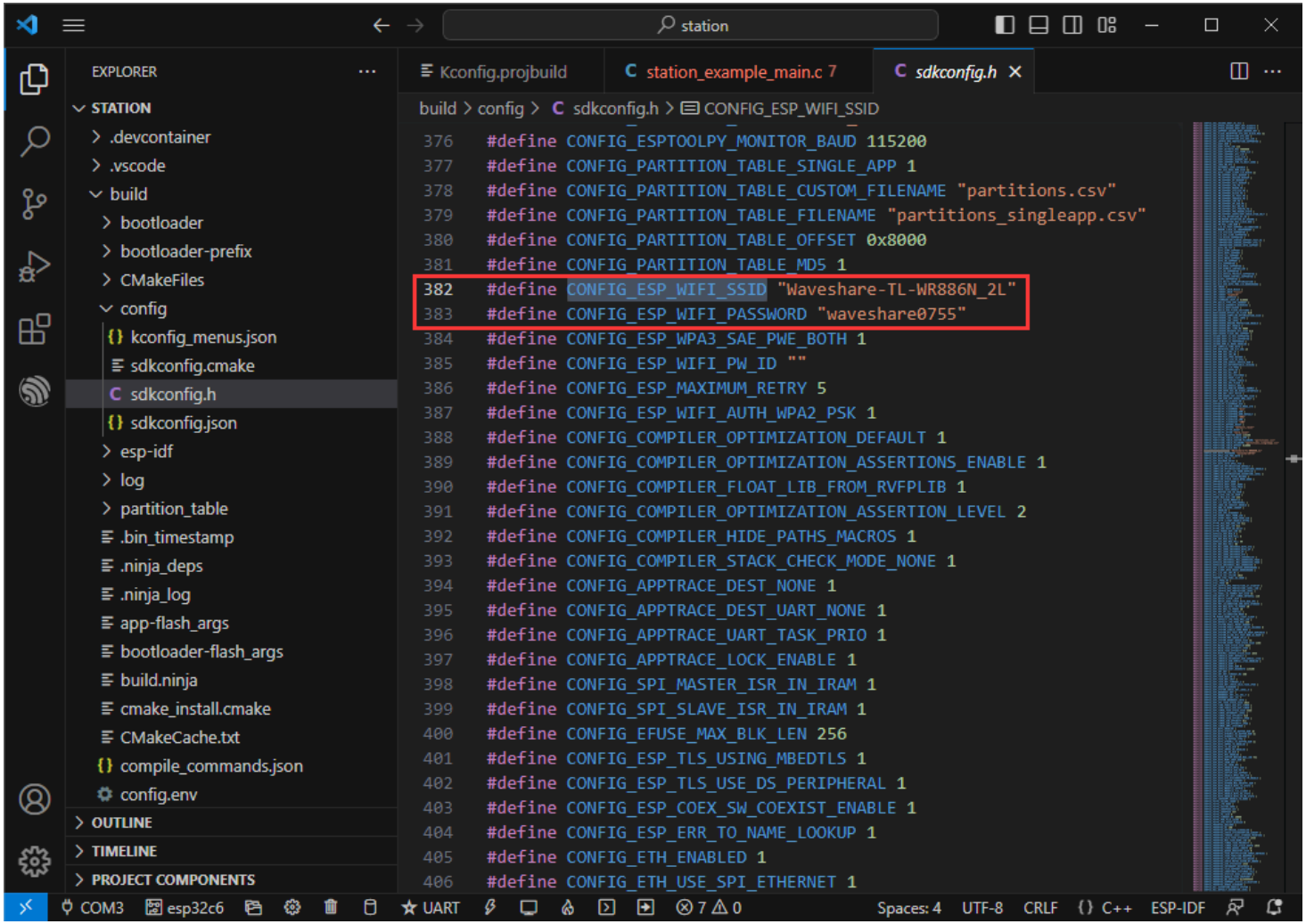
(/wiki/File:ESP32-C6-DEV-KIT-N8-63.png)

- Right-click to Go to Definition.



(/wiki/File:ESP32-C6-DEV-KIT-N8-64.png)

- The previously set value can be seen as:



(/wiki/File:ESP32-C6-DEV-KIT-N8-65.png)

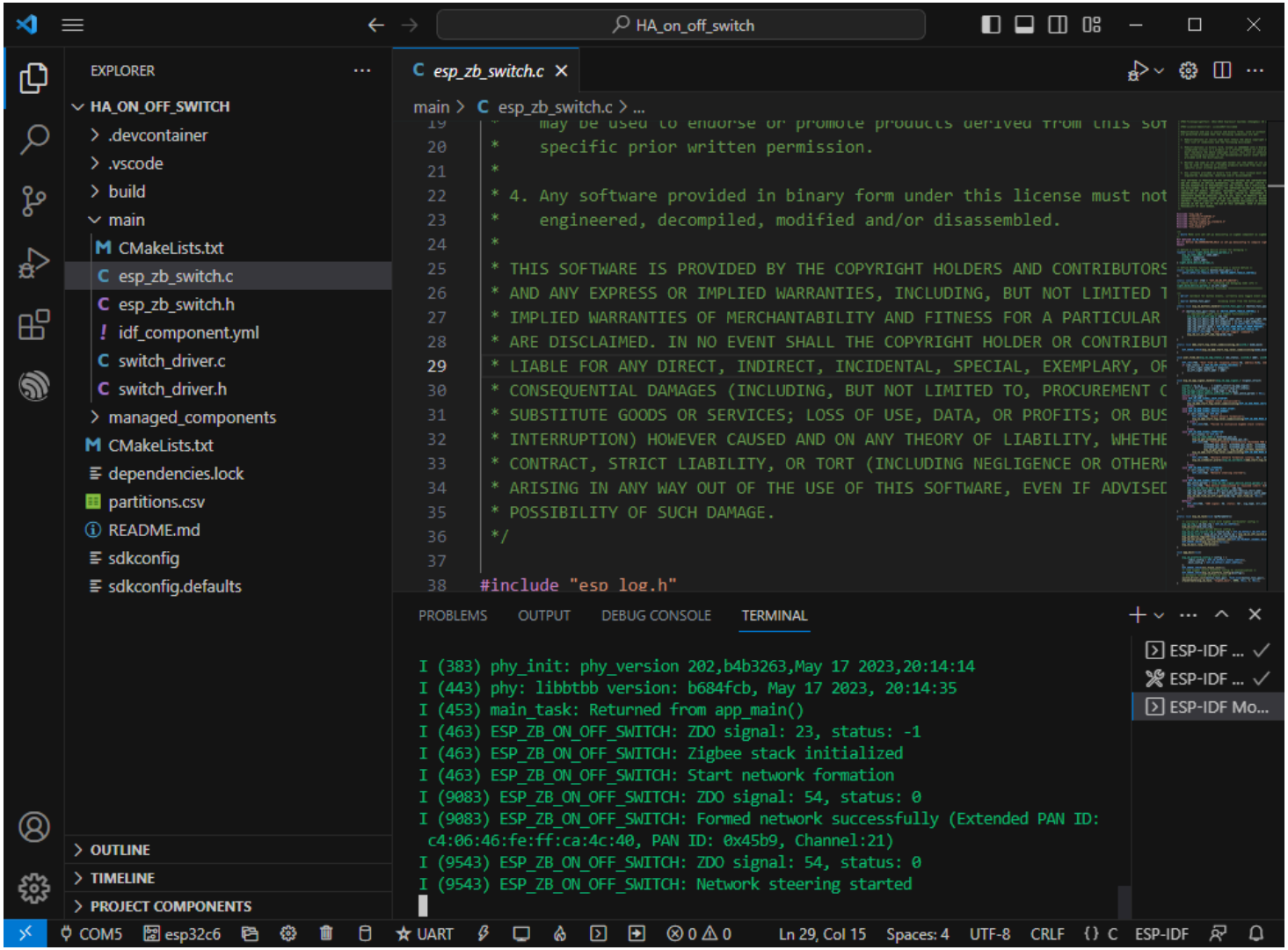
Zigbee

- Official example 1 path: Zigbee -> light_sample -> HA_on_off_switch.
- Official example 2 path: Zigbee -> light_sample -> HA_on_off_light.
- Example effect: use 2 ESP32-C6 boards, use the BOOT key of one ESP32-C6 board (flash HA_on_off_switch demo) to control the RGB LED ON/OFF on the other one.

- **Note: Please flash the HA_on_off_switch demo to one board first, and then flash the HA_on_off_light demo to the other board.**

Software Operation 1

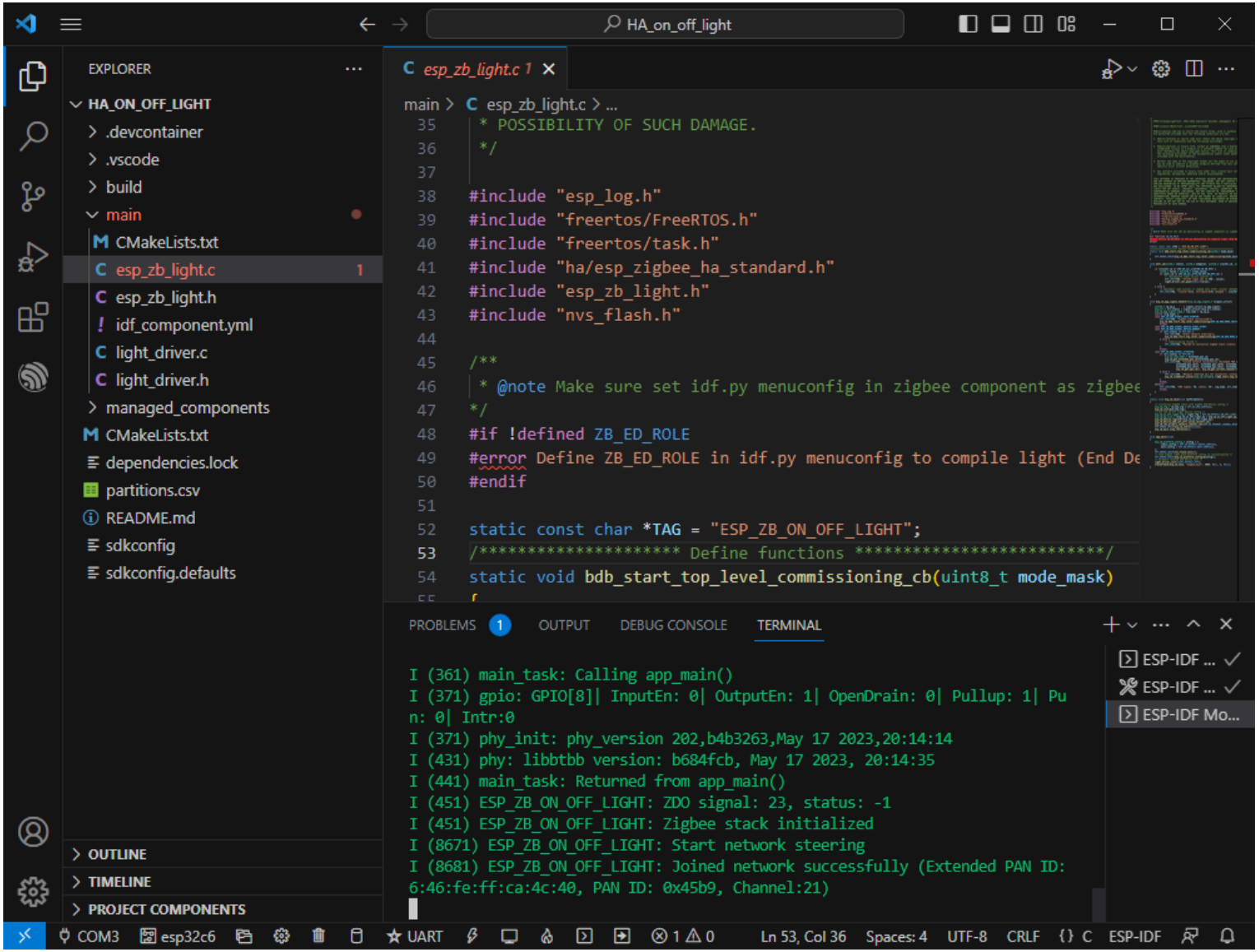
- Create the official example HA_on_off_switch according to the tutorial above. (Create Example)
- The demo is compatible with ESP32-C6 and can be used without modifying the demo content.
- Modify the COM port and the driver object. **It is recommended to prioritize the use of the COM port corresponding to USB (can be viewed through the device manager.))** click compile and flash to run the demo.



(/wiki/File:ESP32-C6-DEV-KIT-N8-66.png)

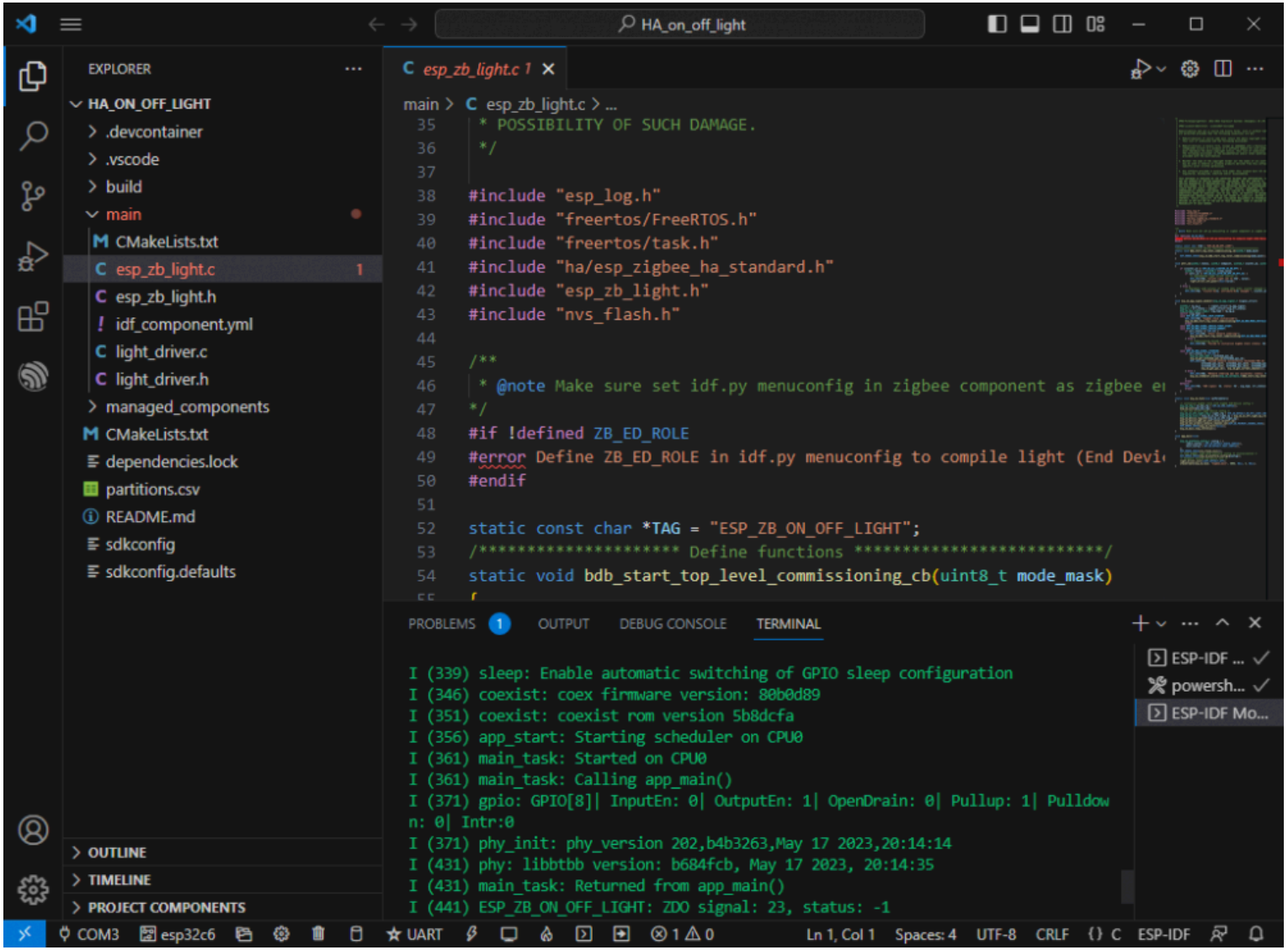
Software Operation 2

- Follow the tutorial above to create the official example HA_on_off_light. (Create Example)
- The demo is compatible with ESP32-C6 and can be used without modifying the demo.
- Modify the COM port and driver object, click compile and flash to run the demo (**you need to wait for a moment for the two chips to establish a connection**).



(/wiki/File:ESP32-C6-DEV-KIT-N8-70.png)

- **If the device remains unconnected**, it may be due to residual network information on the device, so you can erase the device information (Erase Tutorial) and reorganize the network.



(/wiki/File:ESP32-C6-DEV-KIT-N8-71.png)

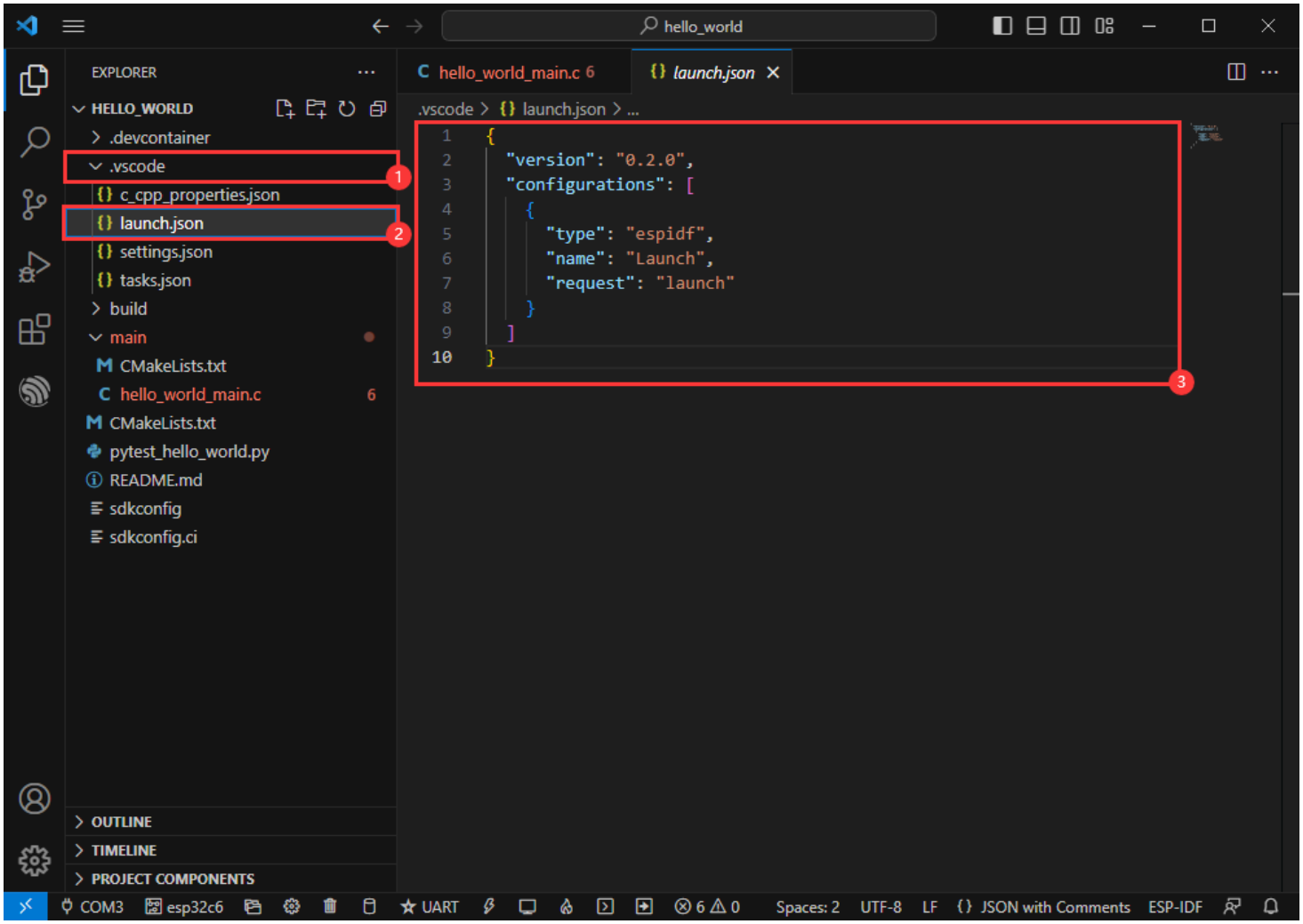
JTAG Debug

Software Operation

- Create a debugging example, this example uses the official example hello_world. (Create

Example)

- Modify the **launch.json** file.



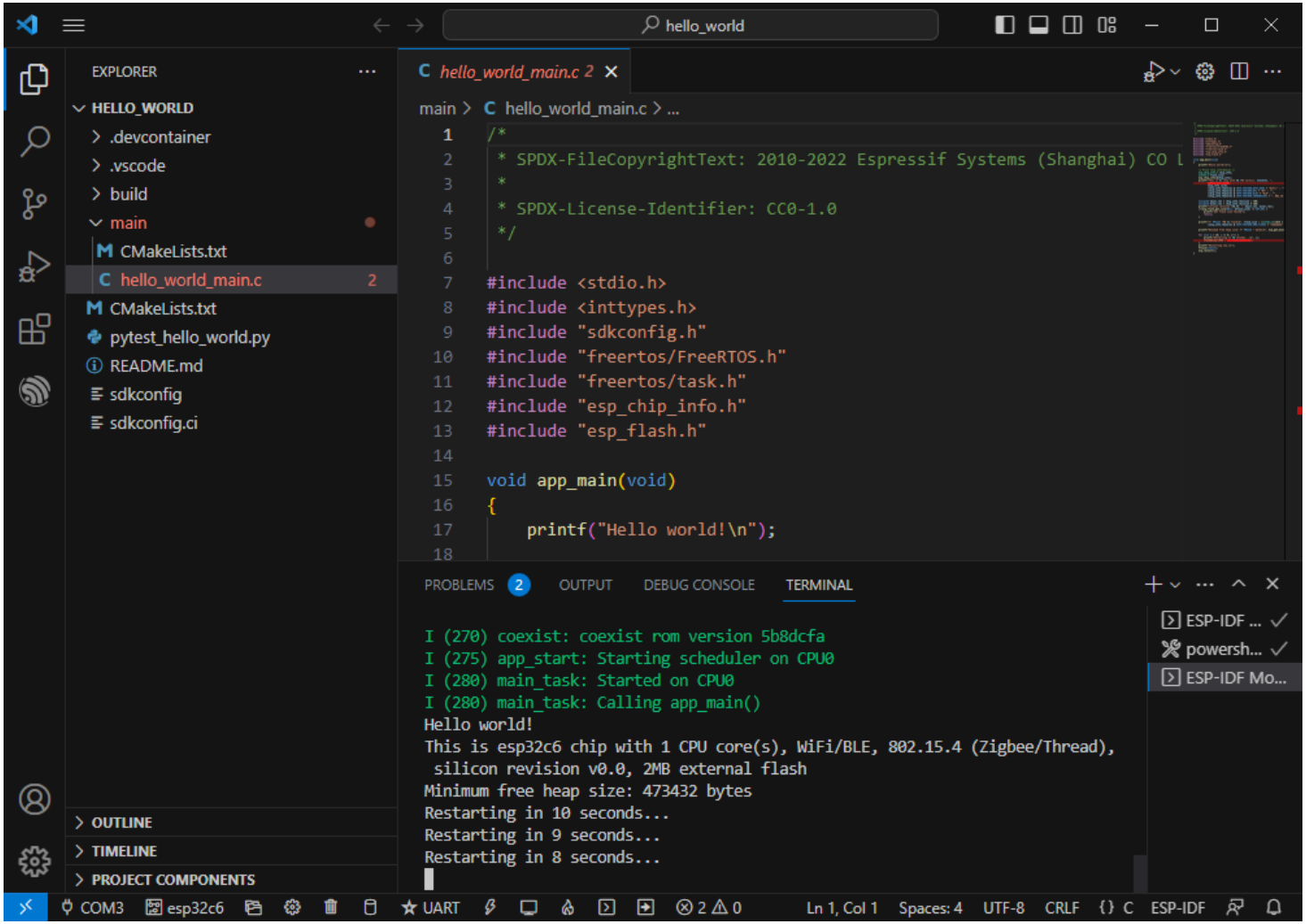
(/wiki/File:ESP32-C6-DEV-KIT-N8-72.png)

- Input the following content:

```
{
 "version": "0.2.0",
 "configurations": [
   {
     "name": "GDB",
     "type": "cppdbg",
     "request": "launch",
     "MIMode": "gdb",
     "miDebuggerPath": "${command:espIdf.getXtensaGdb}",
     "program": "${workspaceFolder}/build/${command:espIdf.getProjectName}.elf",
     "windows": {
       "program": "${workspaceFolder}\\build\\${command:espIdf.getProjectName}.elf"
     },
     "cwd": "${workspaceFolder}",
     "environment": [{ "name": "PATH", "value": "${config:idf.customExtraPaths}" }],
     "setupCommands": [
       { "text": "target remote :3333" },
       { "text": "set remote hardware-watchpoint-limit 2"},
       { "text": "mon reset halt" },
       { "text": "thb app_main" },
       { "text": "flushregs" }
     ],
     "externalConsole": false,
     "logging": {
       "engineLogging": true
     }
   }
 ]
}
```

(/wiki/File:ESP32-C6-DEV-KIT-N8-73.png)

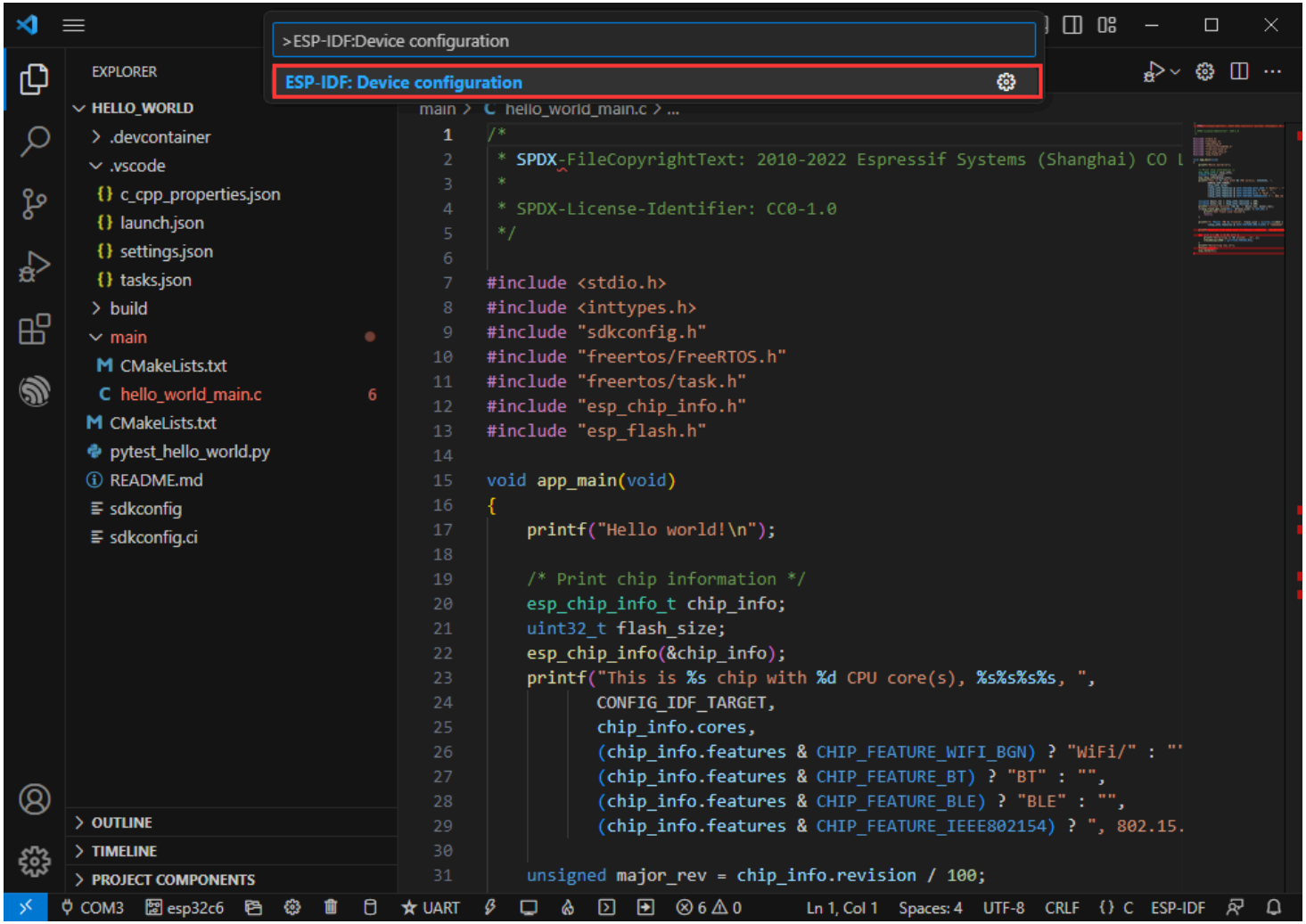
- The demo is compatible with ESP32-C6 and can be used without modifying the demo content.
- Modify the COM port and the driver object (**Please use the USB interface; the UART interface does not support JTAG debugging. The corresponding COM port can be checked through the Device Manager.**), click compile and flash to run the demo.



(/wiki/File:ESP32-C6-DEV-KIT-N8-30.png)

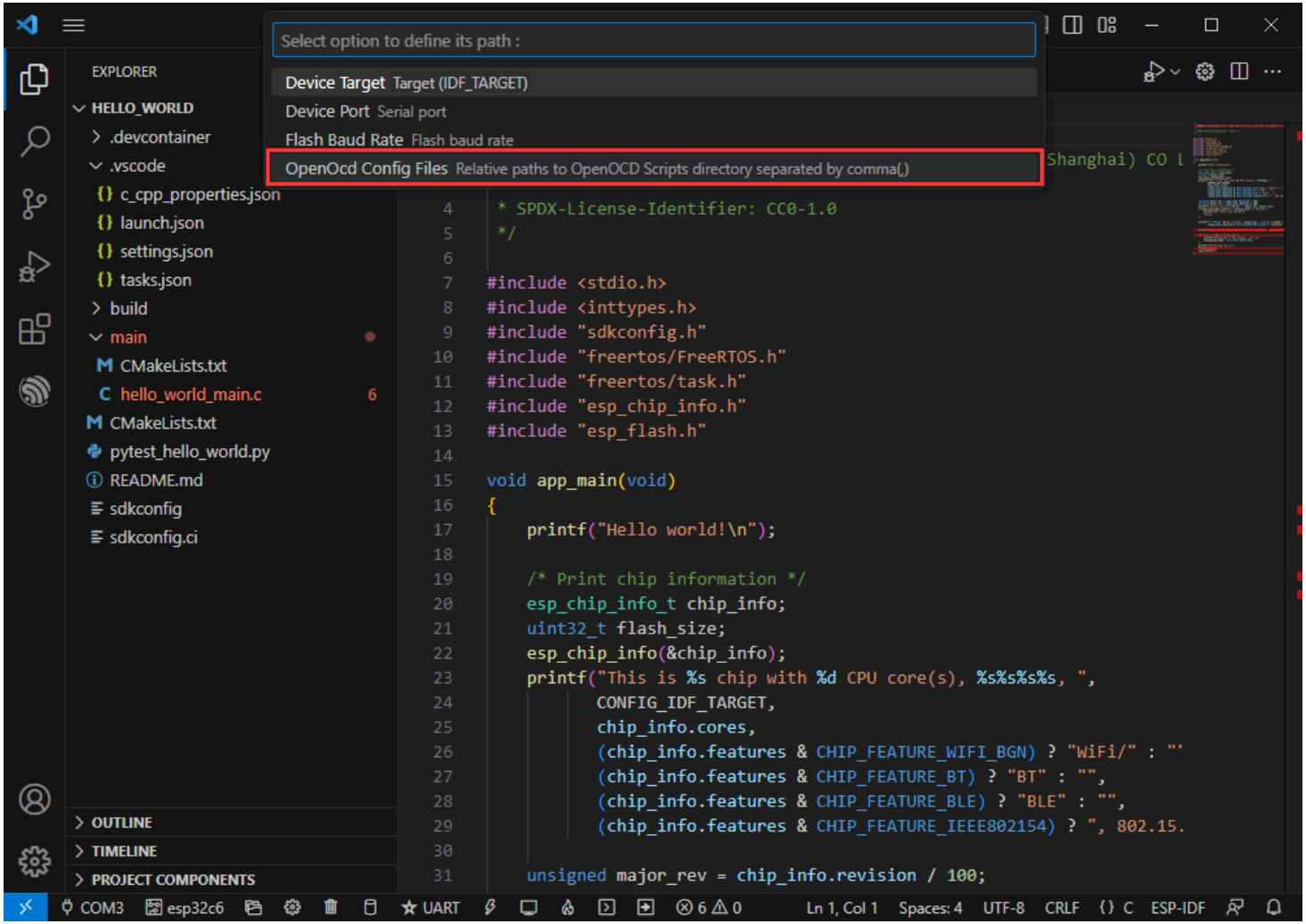
- Press F1 and input:

```
ESP-IDF:Device configuration
```



(/wiki/File:ESP32-C6-DEV-KIT-N8-75.png)

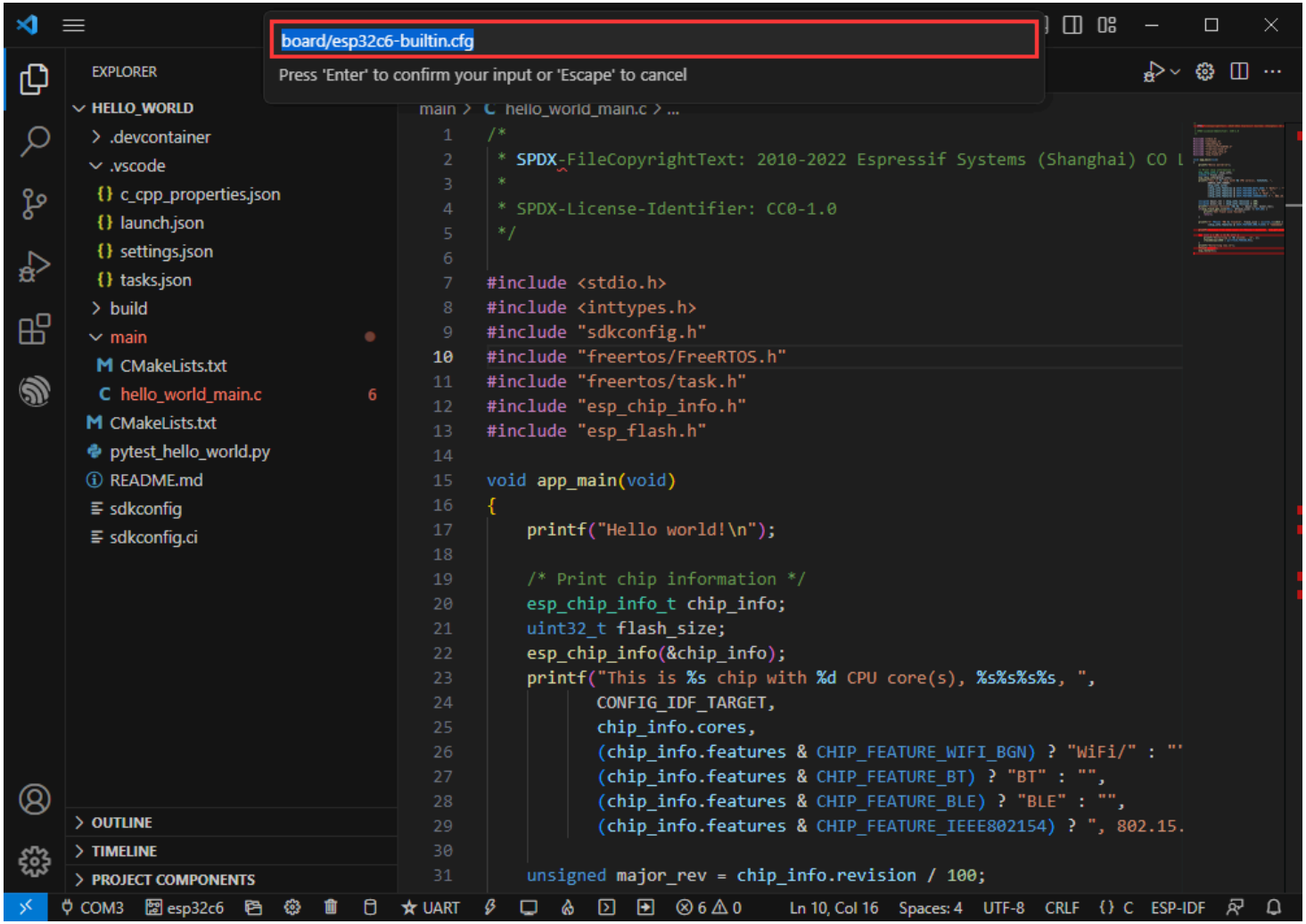
- Select **OpenOcd Config Files**.



(/wiki/File:ESP32-C6-DEV-KIT-N8-76.png)

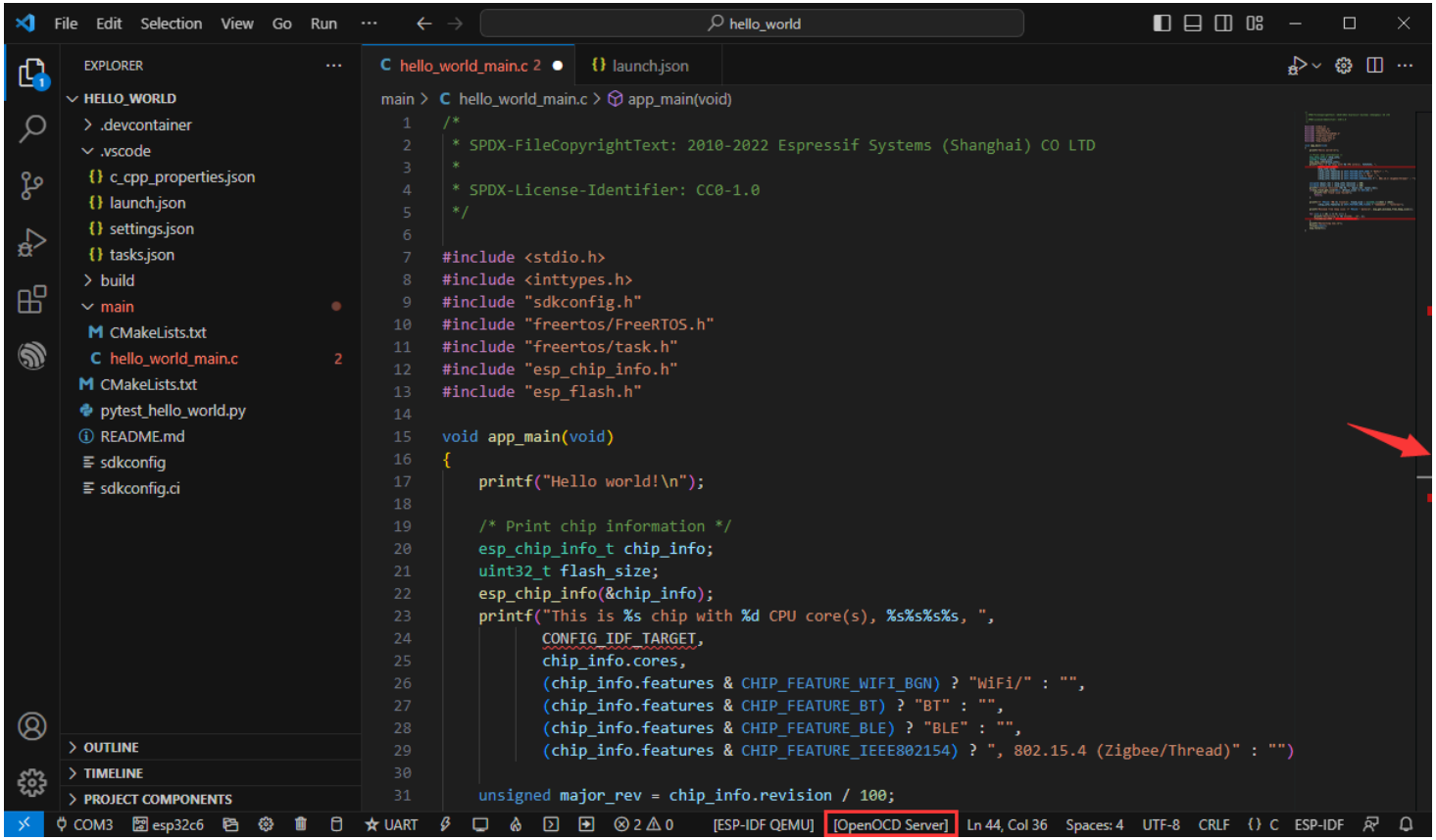
- Type **board/esp32c6-builtin.cfg** (if this is the default, just enter)

```
board/esp32c6-builtin.cfg
```



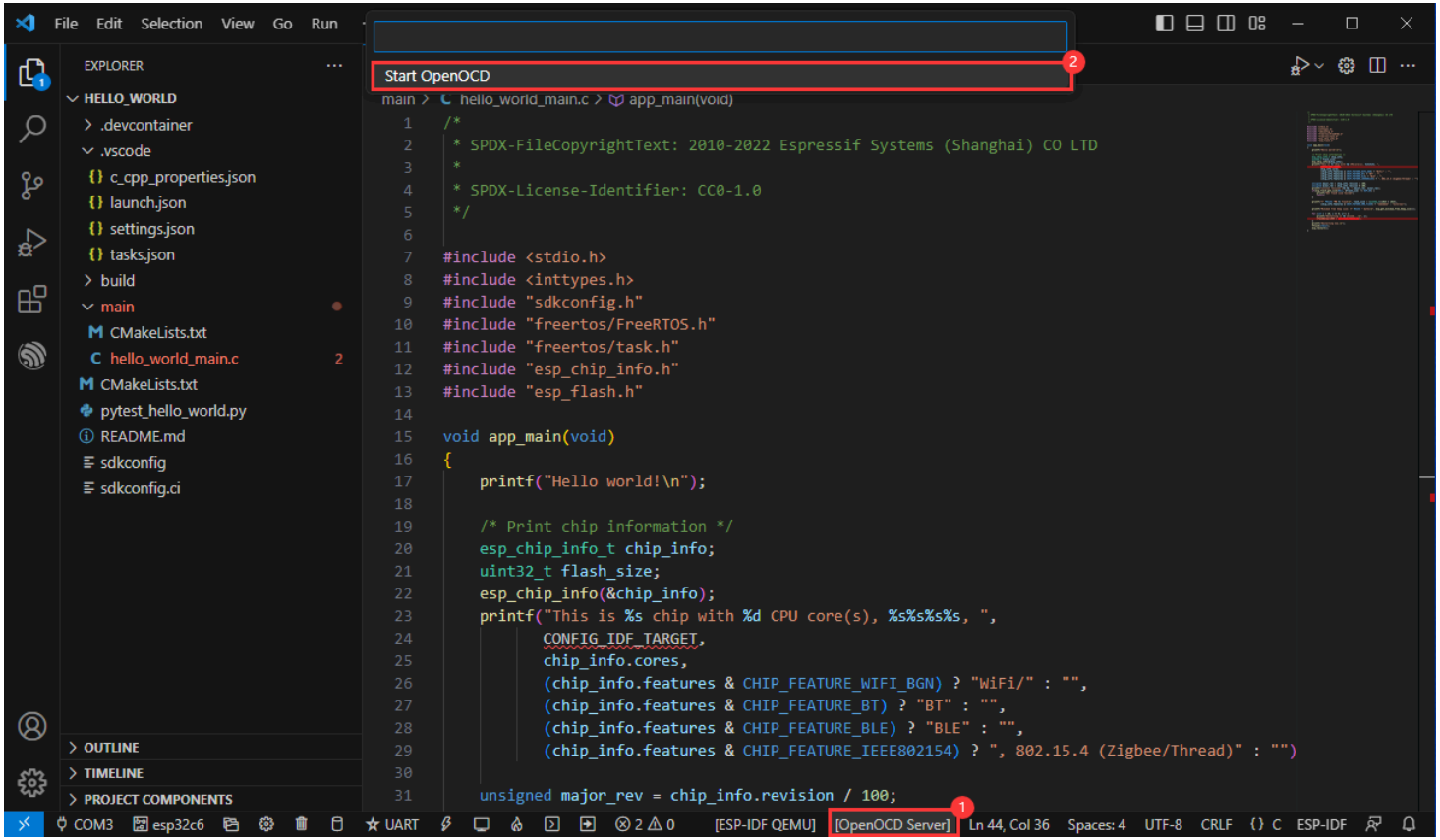
(/wiki/File:ESP32-C6-DEV-KIT-N8077.png)

- Stretch the width of the window until **[OpenOCD Server]** is displayed at the bottom.



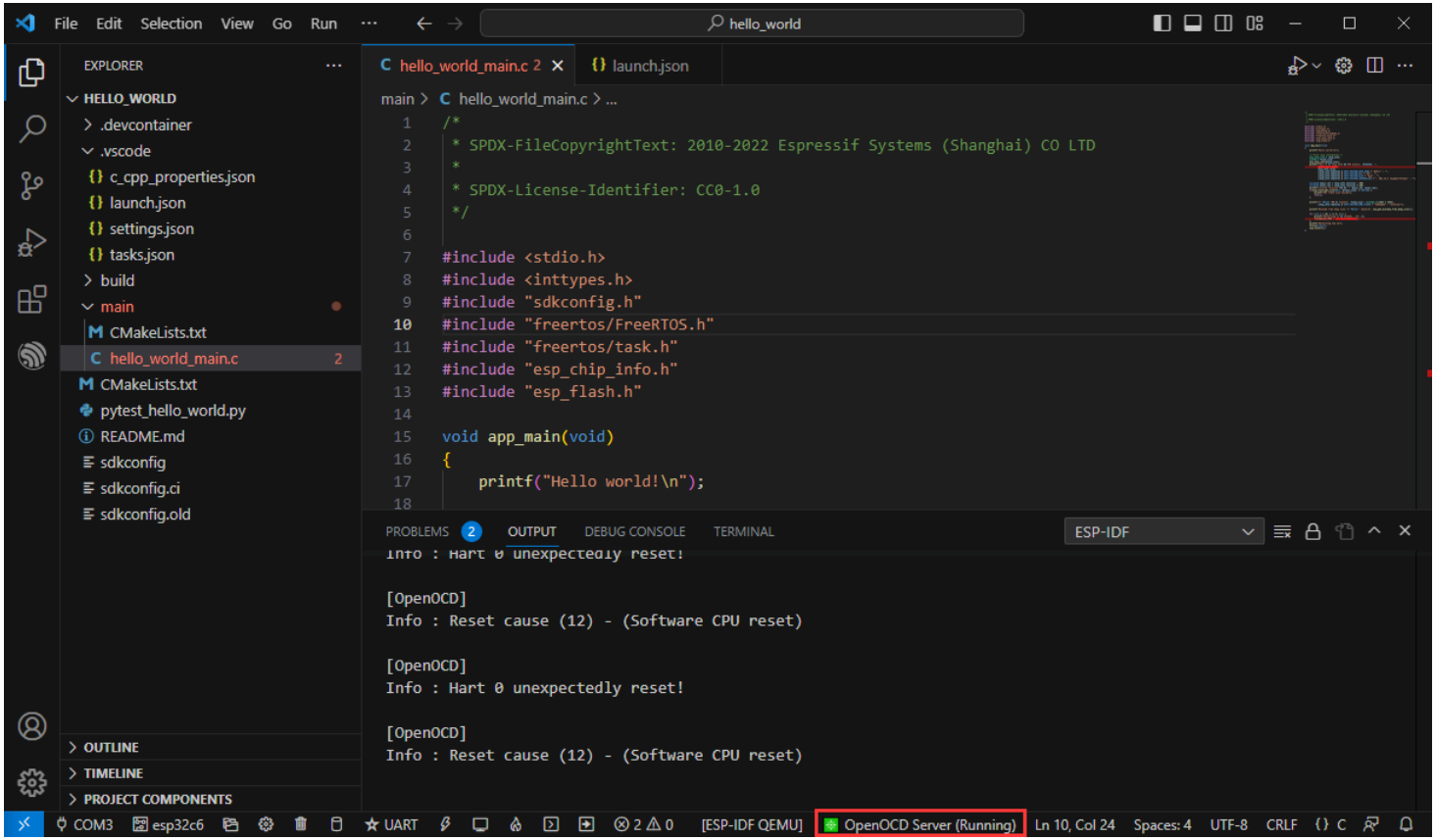
(/wiki/File:ESP32-C6-DEV-KIT-N8078.png)

- Click **[OpenOCD Server]** and select **Start OpenOCD**.



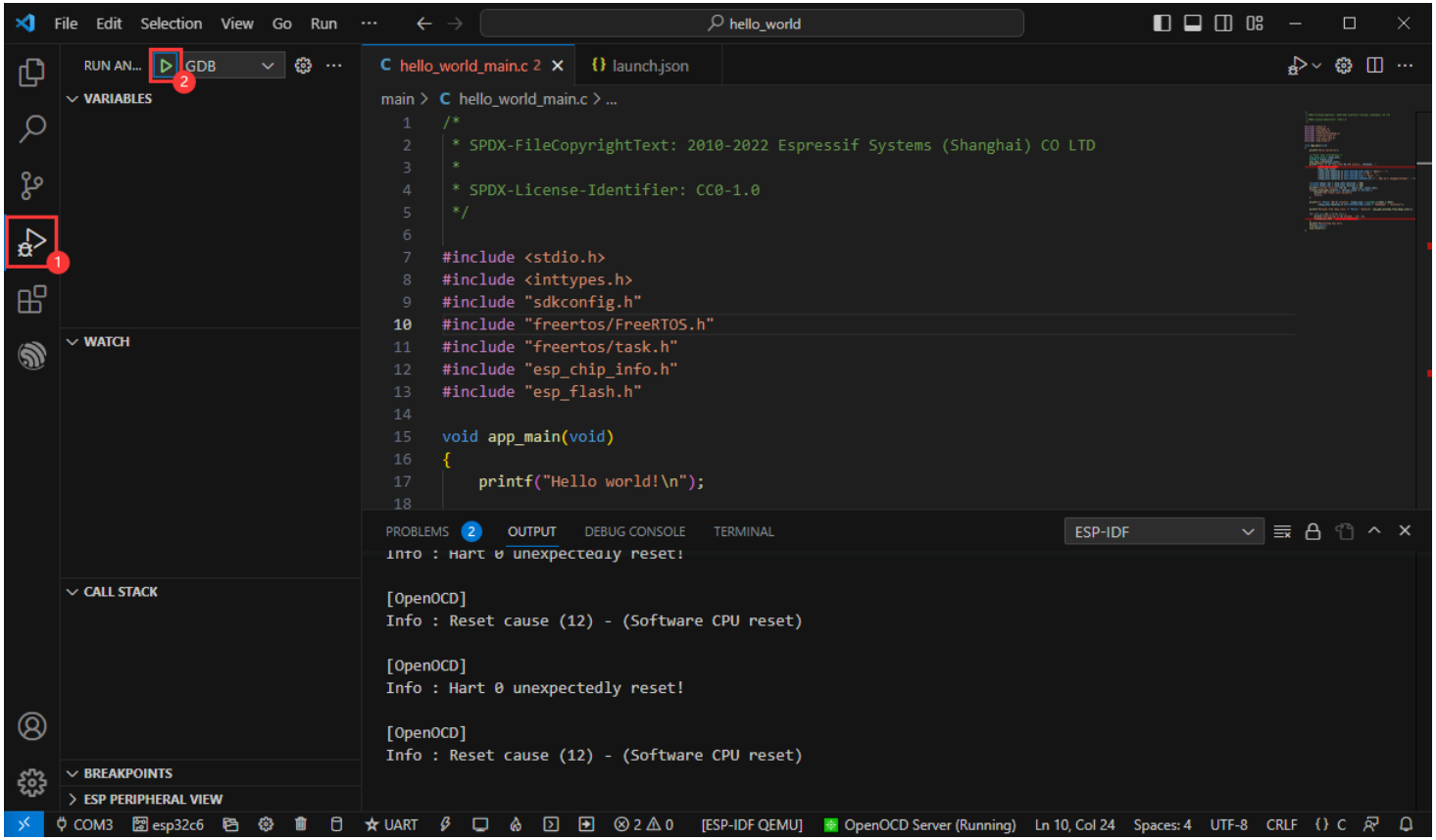
(/wiki/File:ESP32-C6-DEV-KIT-N8079.png)

- Successfully opened as follows:



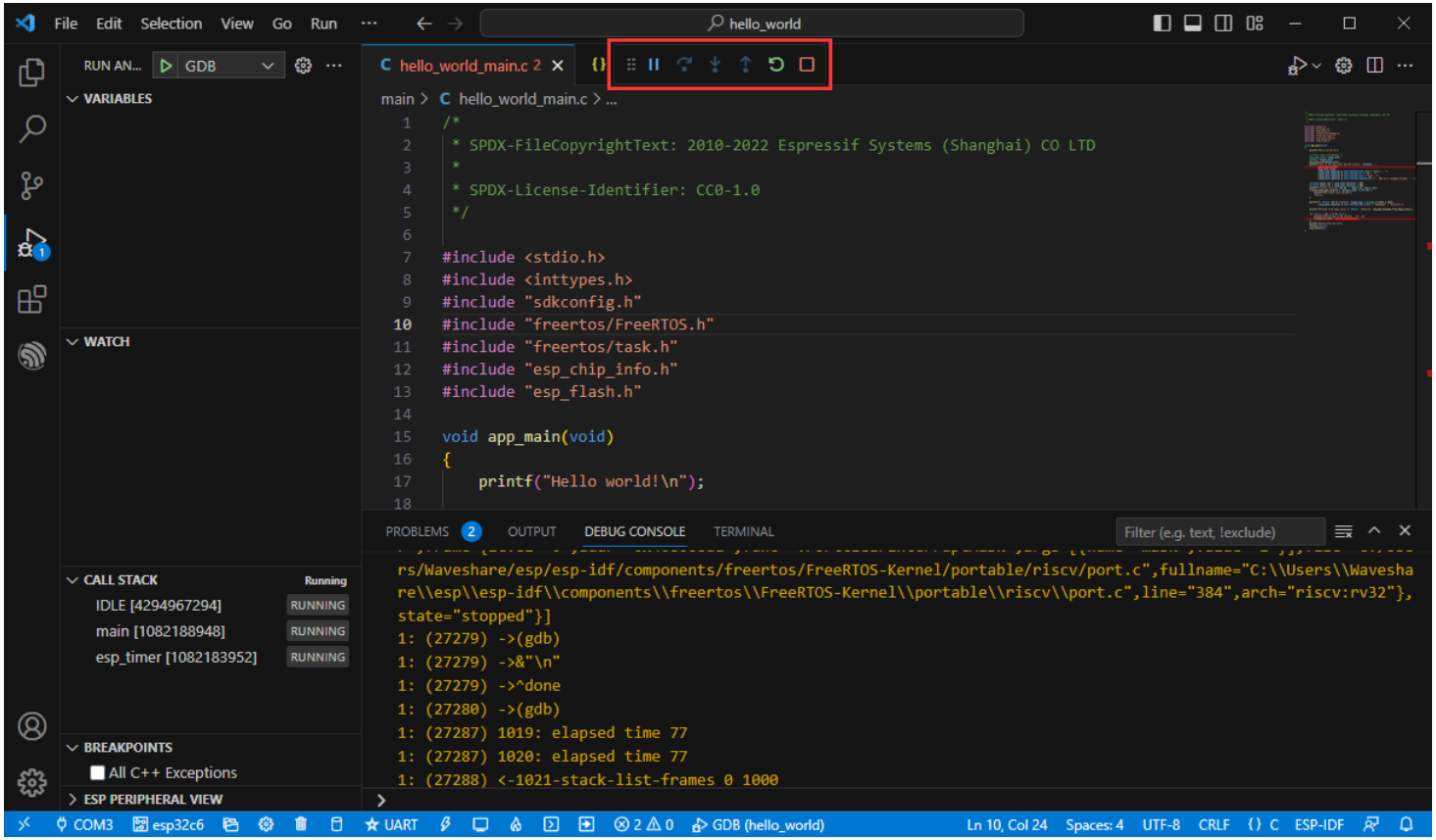
(/wiki/File:ESP32-C6-DEV-KIT-N8-80.png)

- Go to the debug function and click Debug.



(/wiki/File:ESP32-C6-DEV-KIT-N8-81.png)

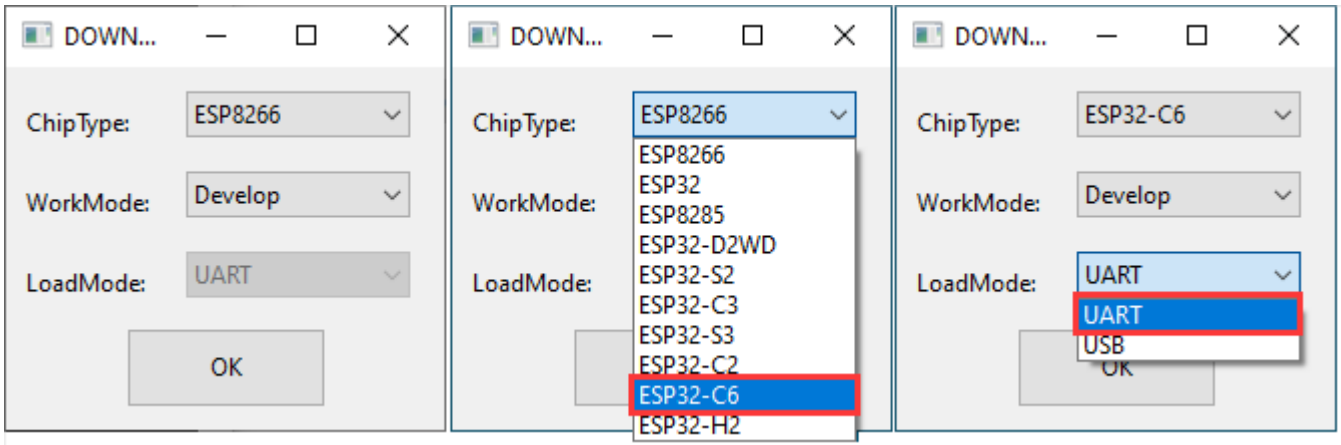
- Successfully enter the debugging interface.



(/wiki/File:ESP32-C6-DEV-KIT-N8-82.png)

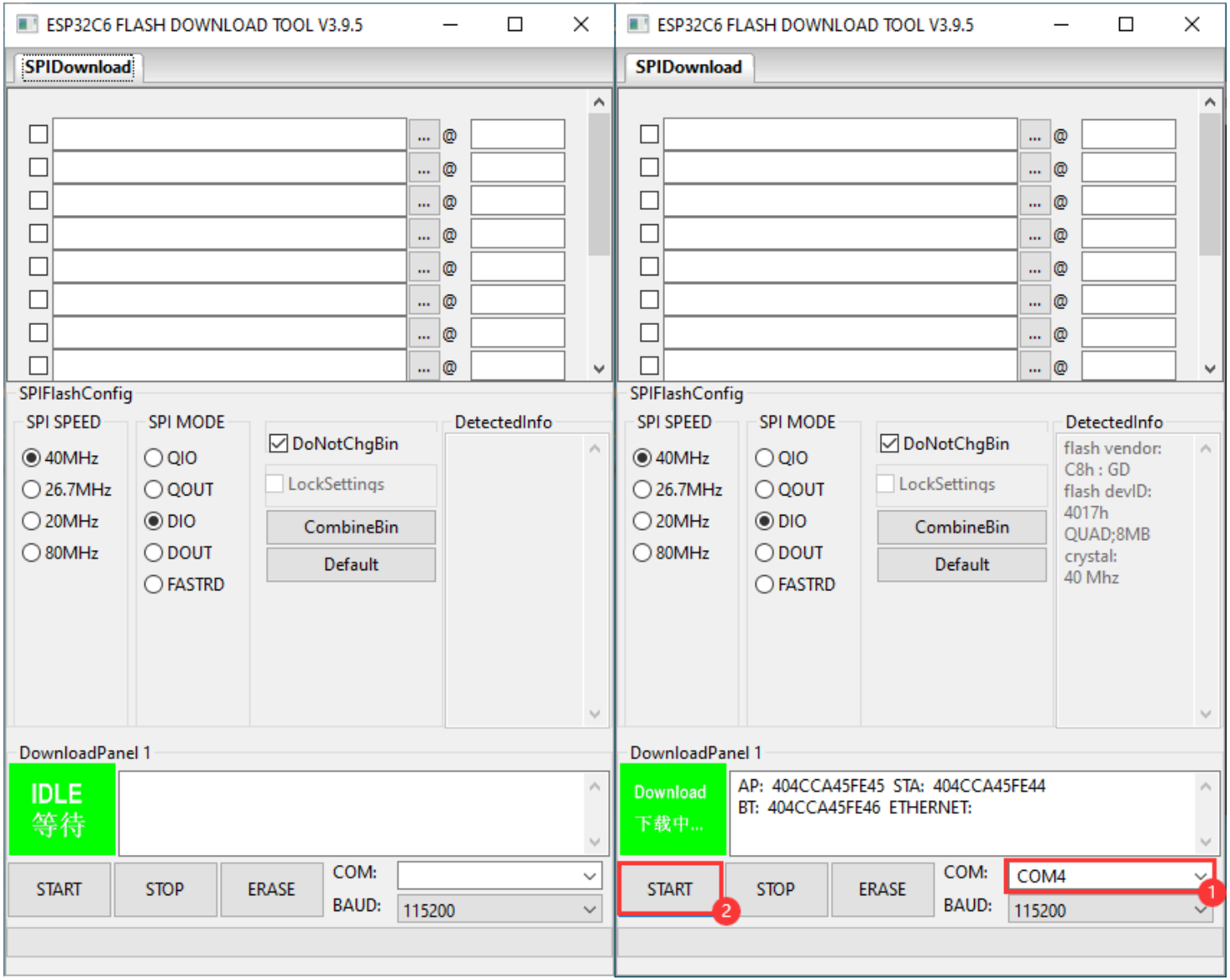
Erase Device Flash

- Unpack the software resource package (Flash debugging software (https://files.waveshare.com/wiki/ESP32-C6-DEV-KIT-N8/Flash_download_tool_3.9.5_0.zip)).
- Open **flash_download_tool_3.9.5.exe** software, select ESP32-C6 and UART.



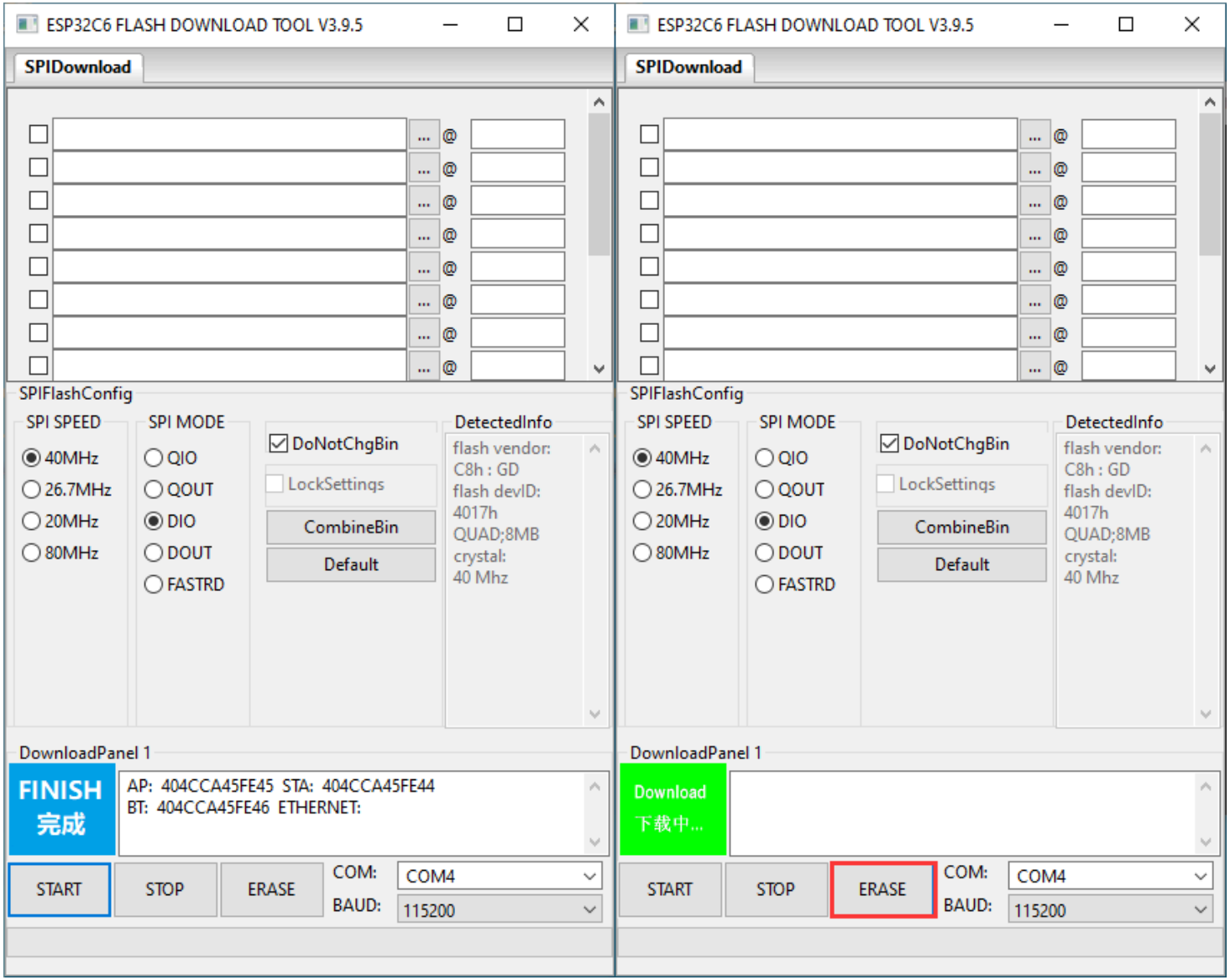
(/wiki/File:ESP32-C6-DEV-KIT-N8-83.png)

- Select the UART port number, and click **START** (not select any bin file).



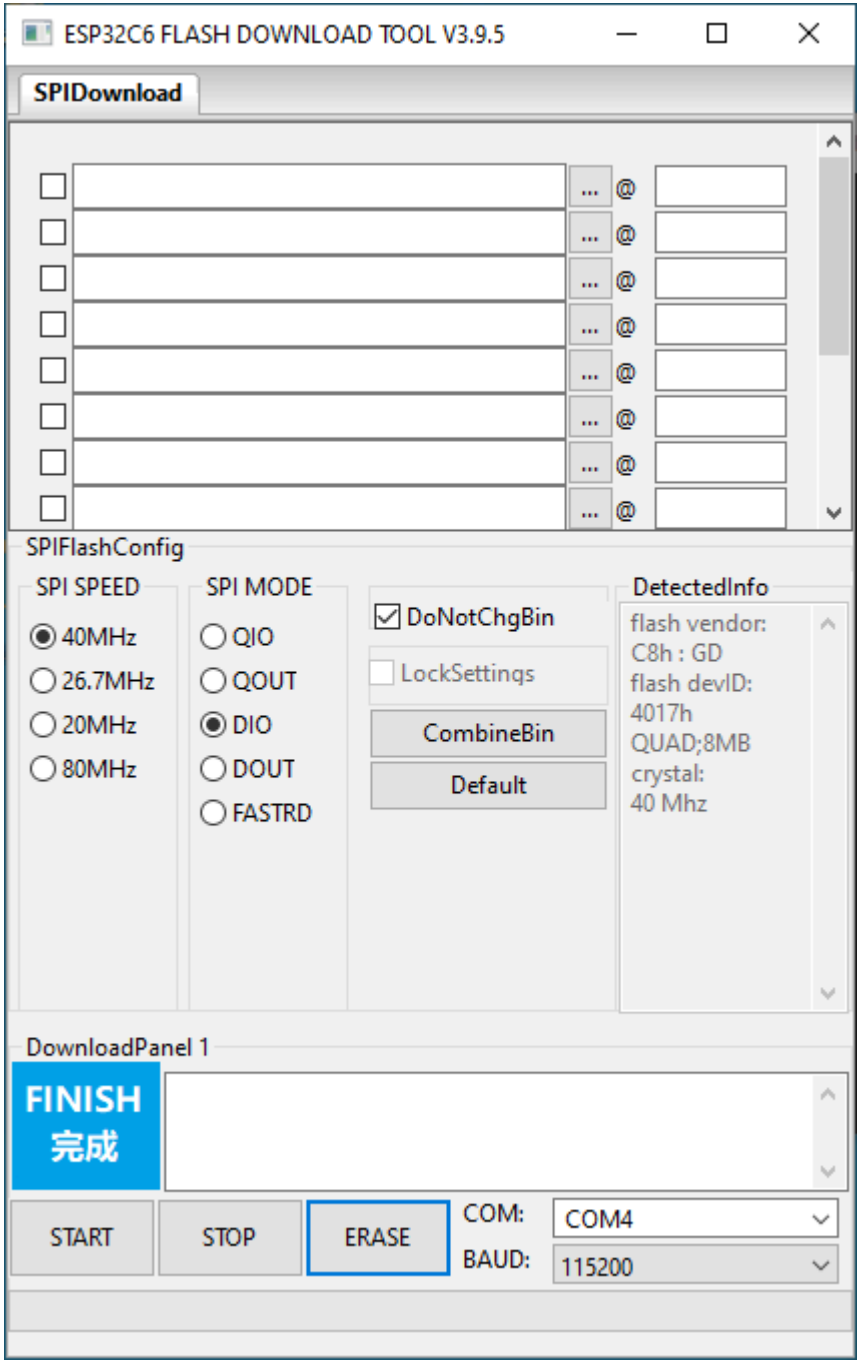
(/wiki/File:ESP32-C6-DEV-KIT-N8-84.png)

- After programming, click on "ERASE".



(/wiki/File:ESP32-C6-DEV-KIT-N8-85.png)

- Waiting for Erase to Finish.



(/wiki/File:ESP32-C6-DEV-KIT-N8-86.png)

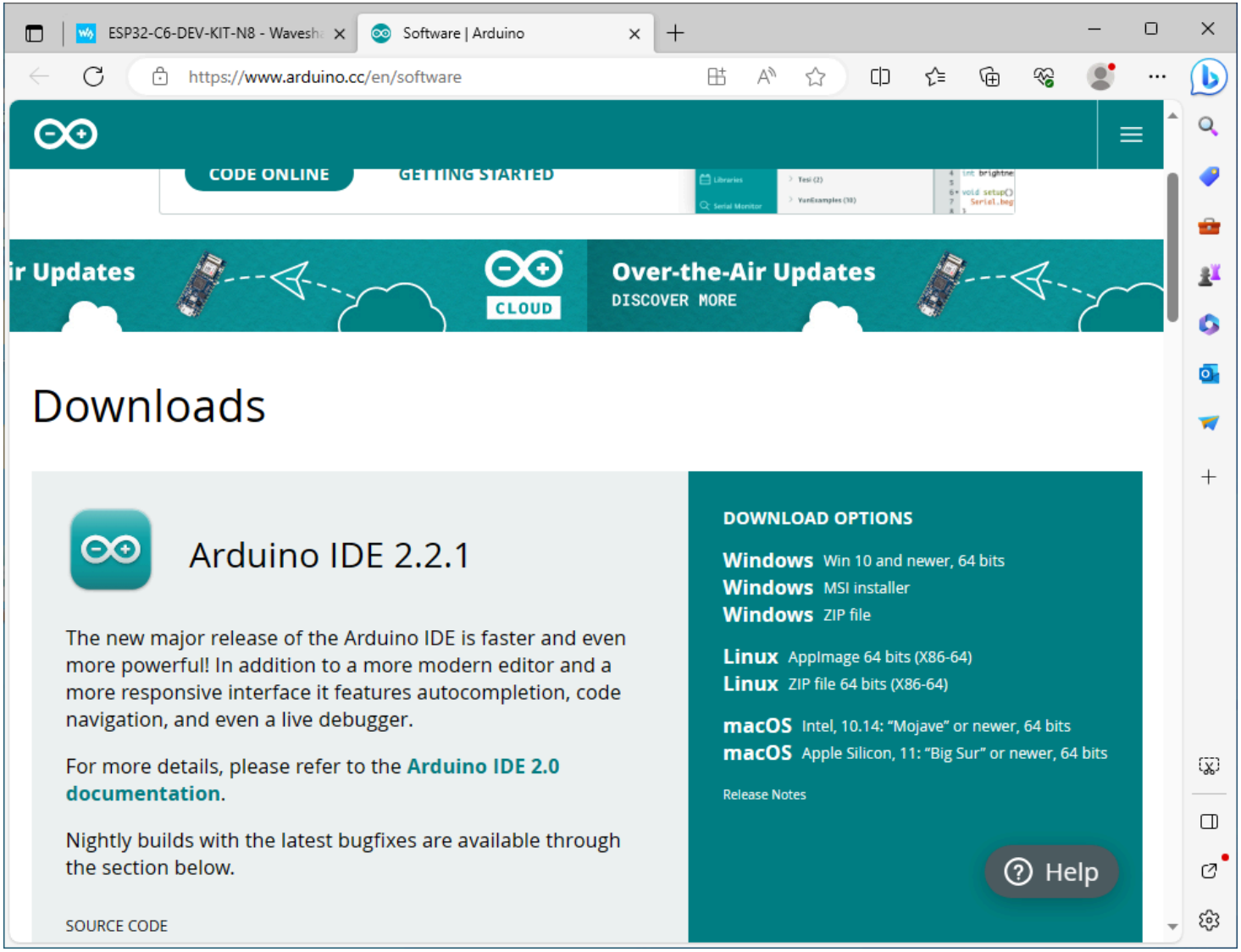
Working with Arduino

Please note that Arduino 3.0.0-alpha is based on ESP-IDF v5.1, which is quite different from the previous ESP-IDF V4.X. The original program may need to be adjusted after the following operations.

- Please do not use the computer username in Chinese as it will cause compilation errors.

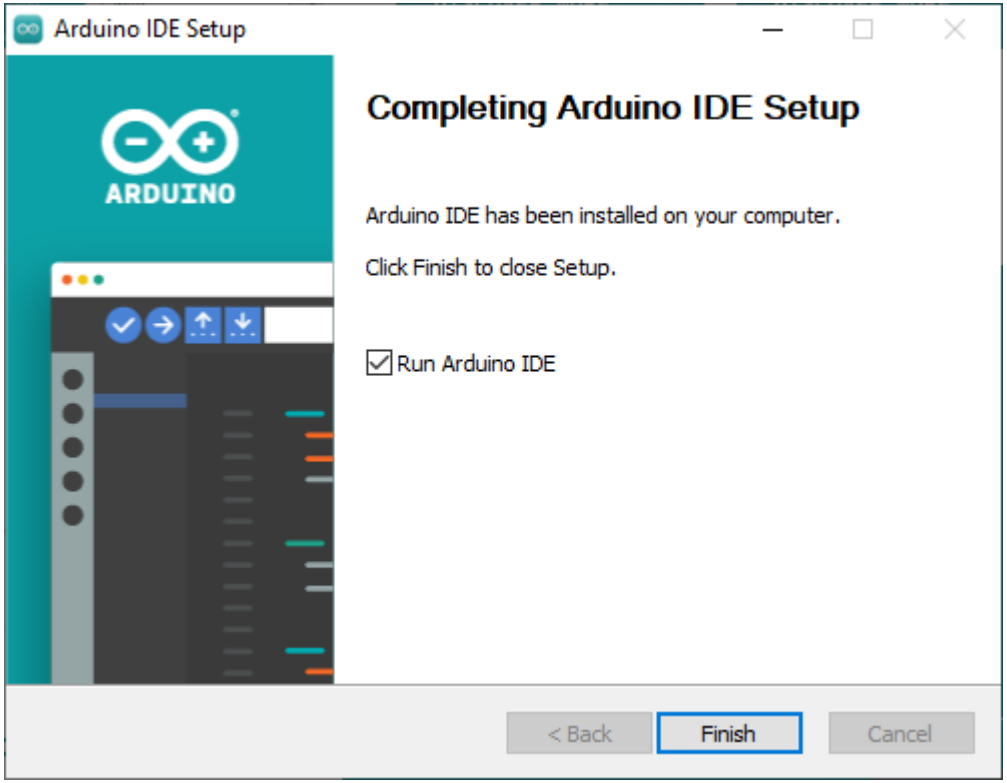
Environment Set-up

- Install Arduino IDE (https://www.arduino.cc/en/software/).



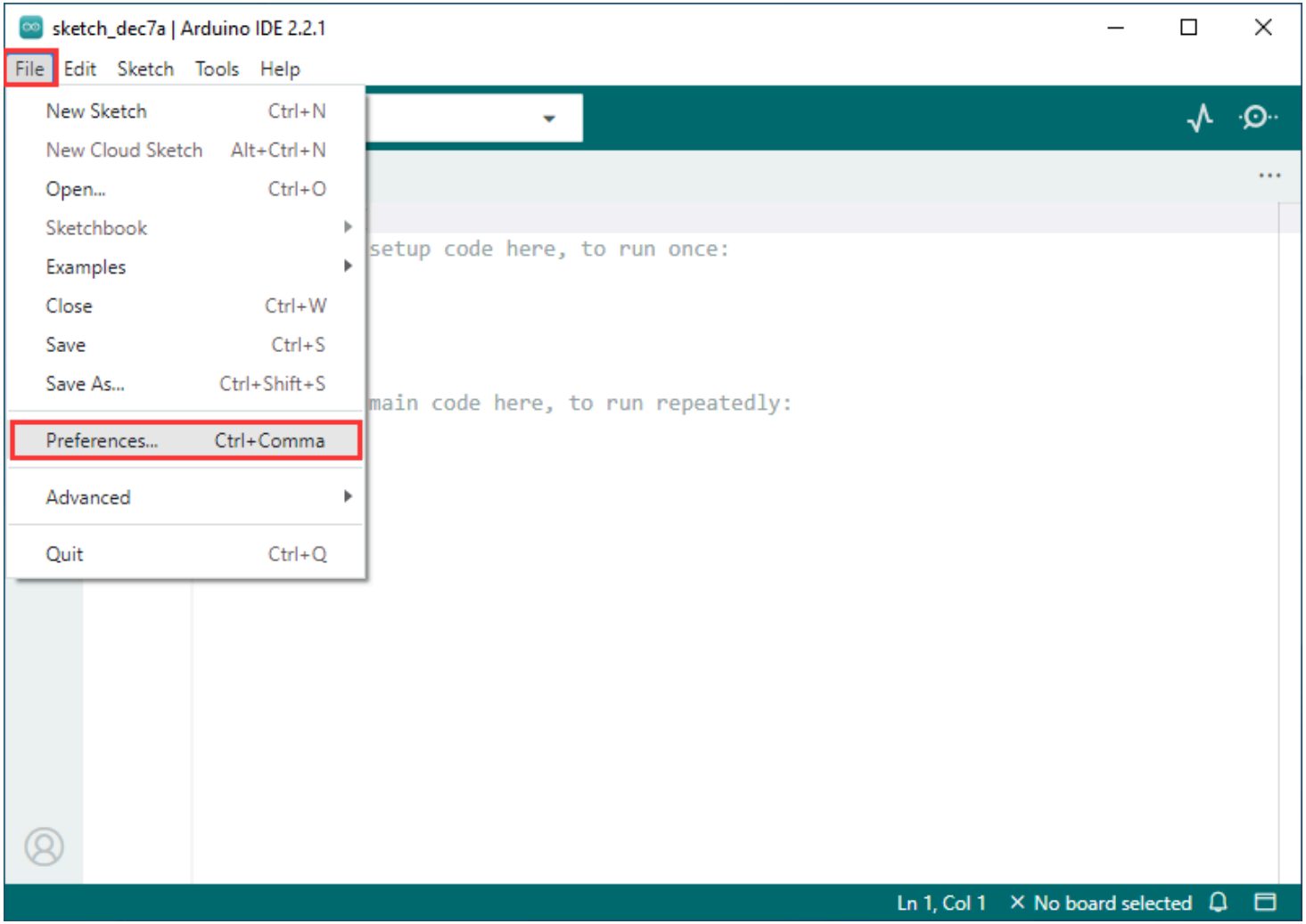
(/wiki/File:ESP32-C6-DEV-KIT-N8-Arduino01.png)

- Enter Arduino IDE after installation.



(/wiki/File:ESP32-C6-DEV-KIT-N8-Arduino02.png)

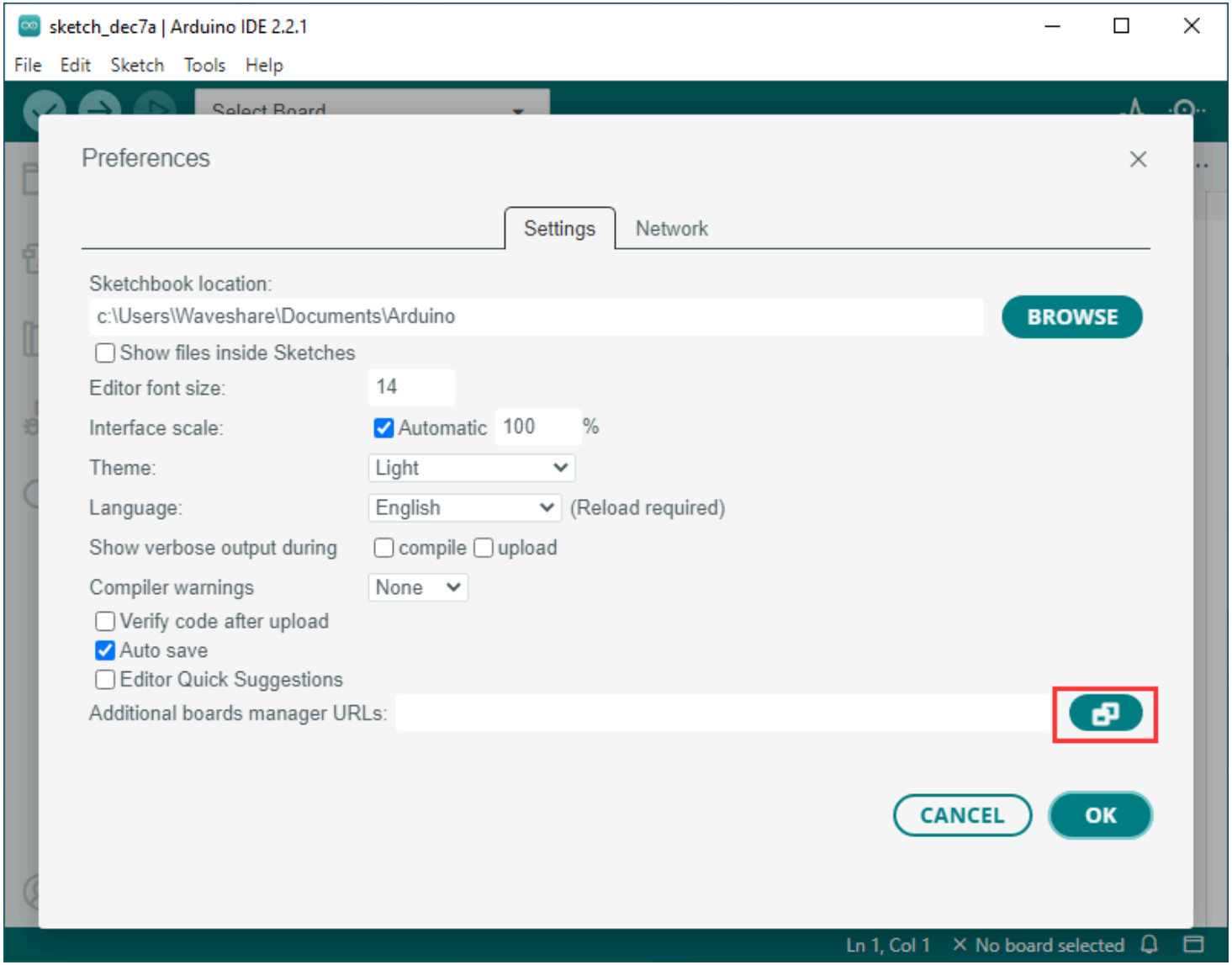
- Enter Preferences.



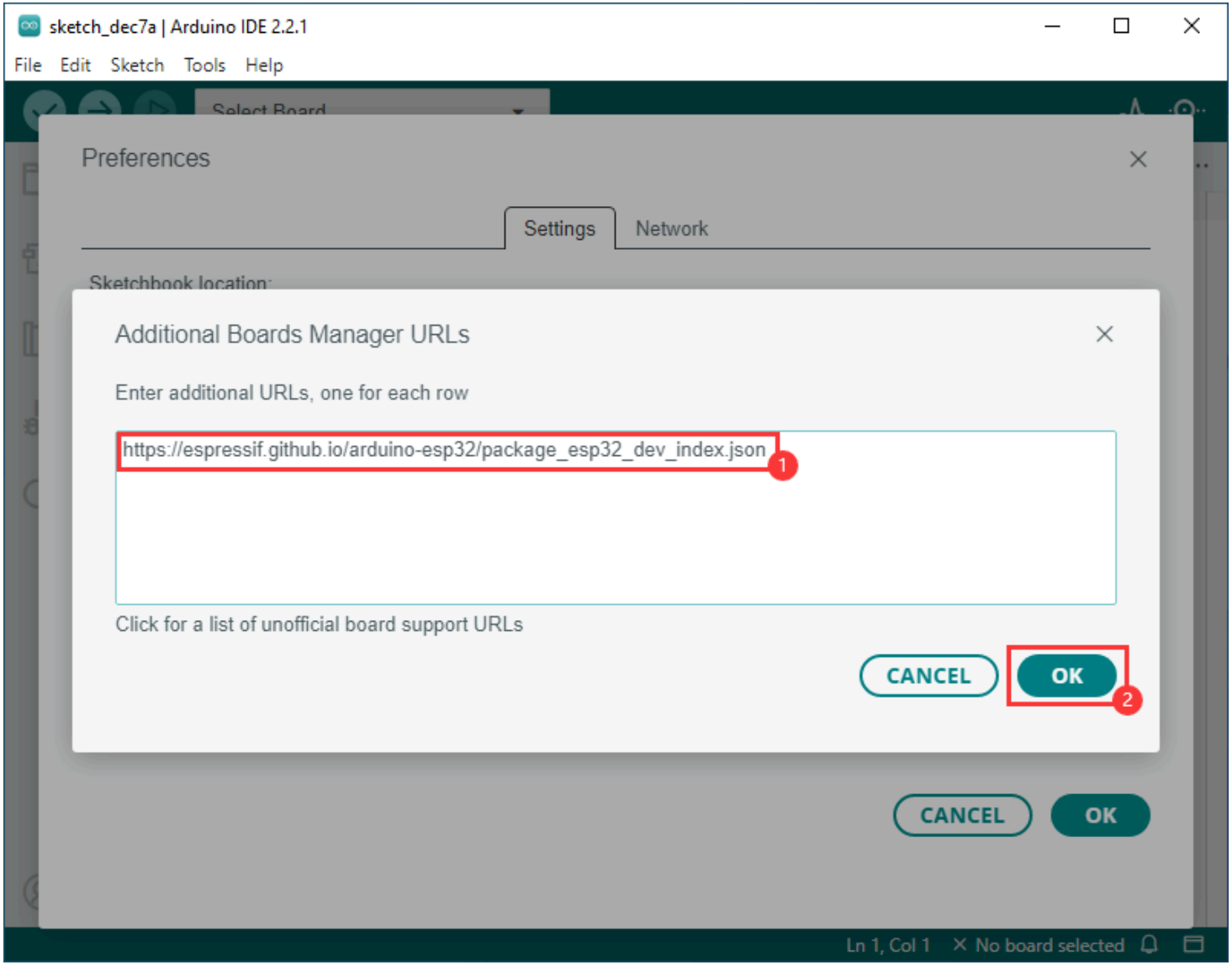
(/wiki/File:ESP32-C6-DEV-KIT-N8-Arduino03.png)

- Add JSON link:

```
https://espressif.github.io/arduino-esp32/package_esp32_dev_index.json (https://espressif.github.io/arduino-esp32/package_esp32_dev_index.json)
```

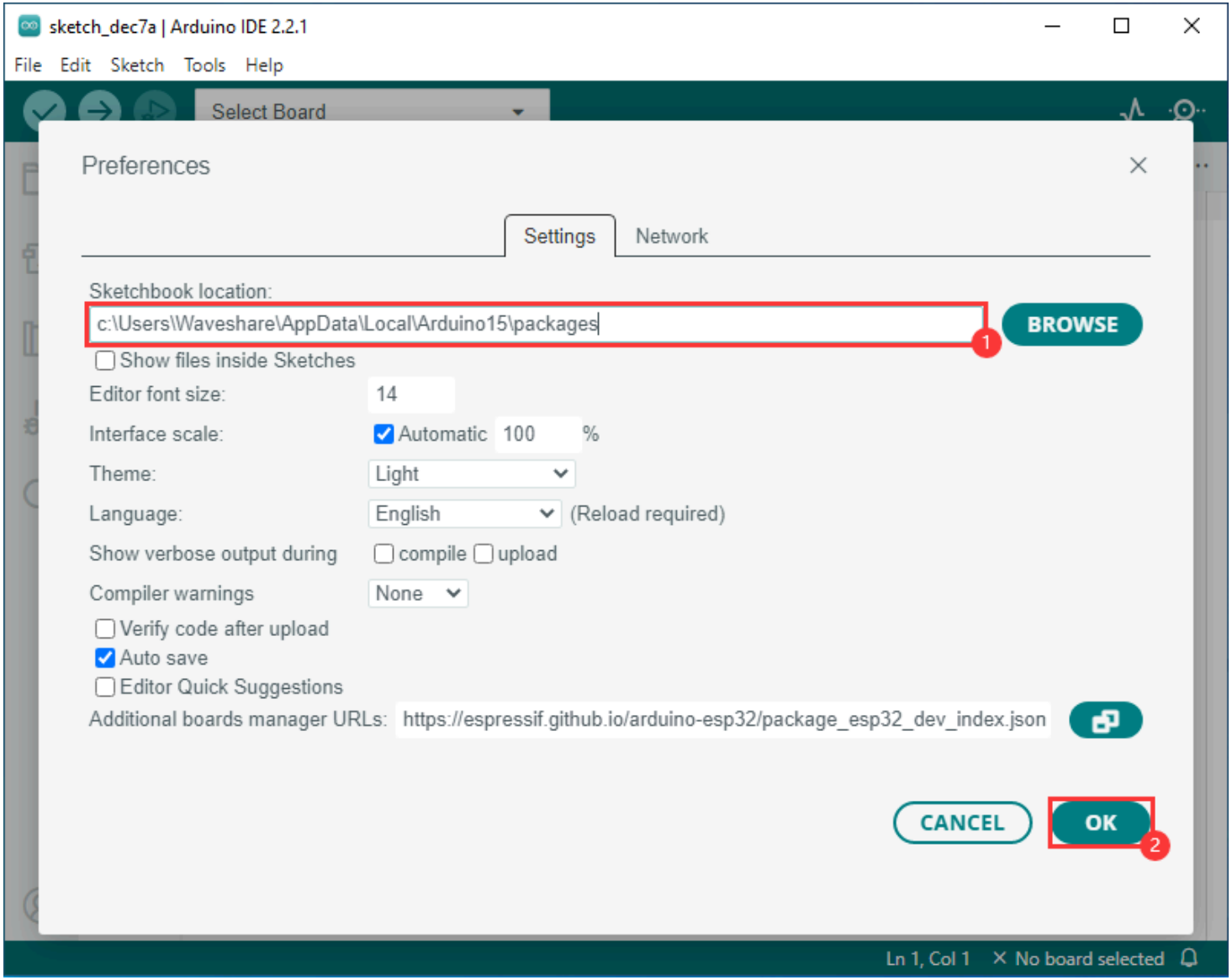


(/wiki/File:ESP32-C6-DEV-KIT-N8-Arduino04.png)



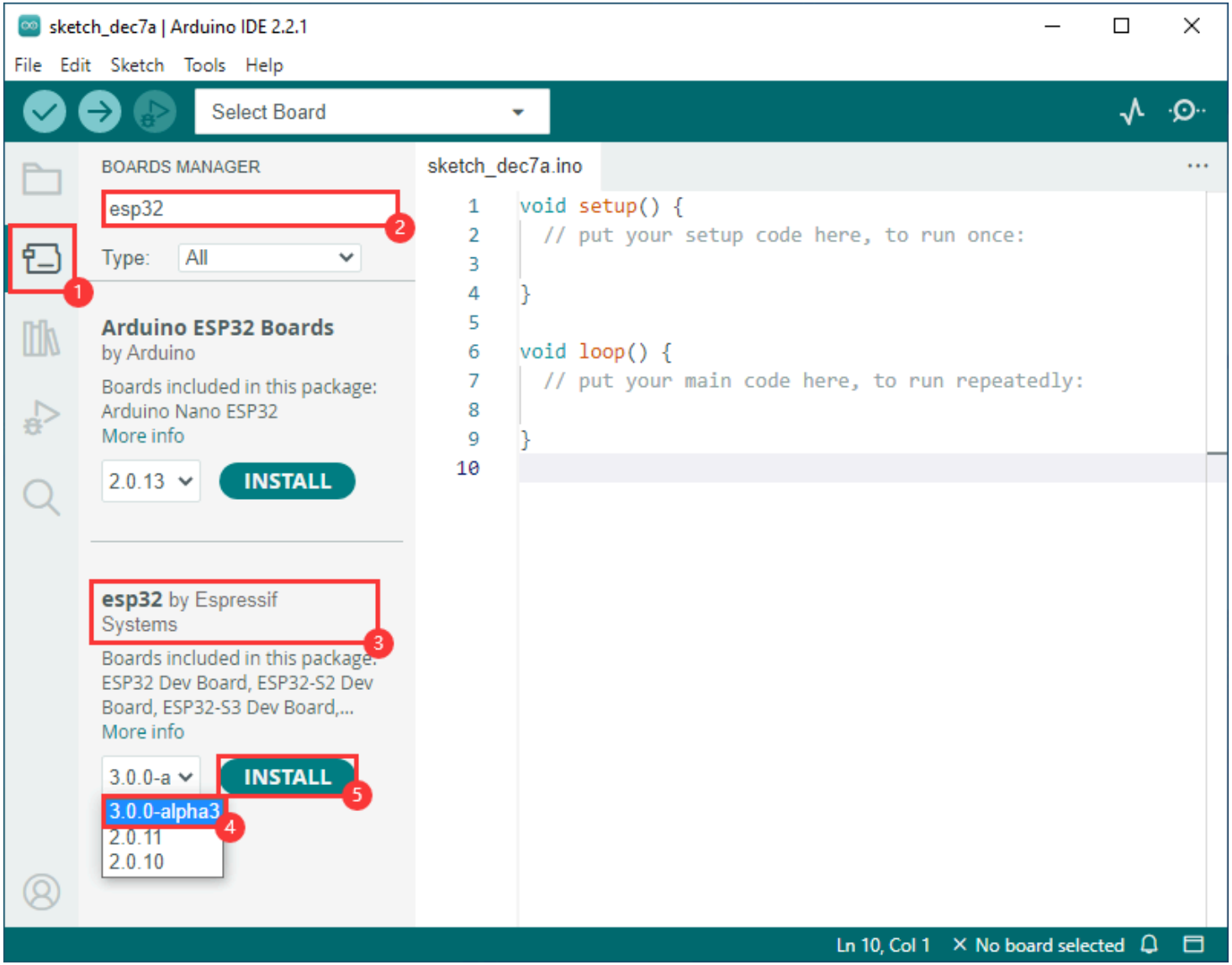
(/wiki/File:ESP32-C6-DEV-KIT-N8-Arduino05.png)

- Modify the project file folder as **C:\Users\Waveshare\AppData\Local\Arduino15\packages** (Waveshare is the username).

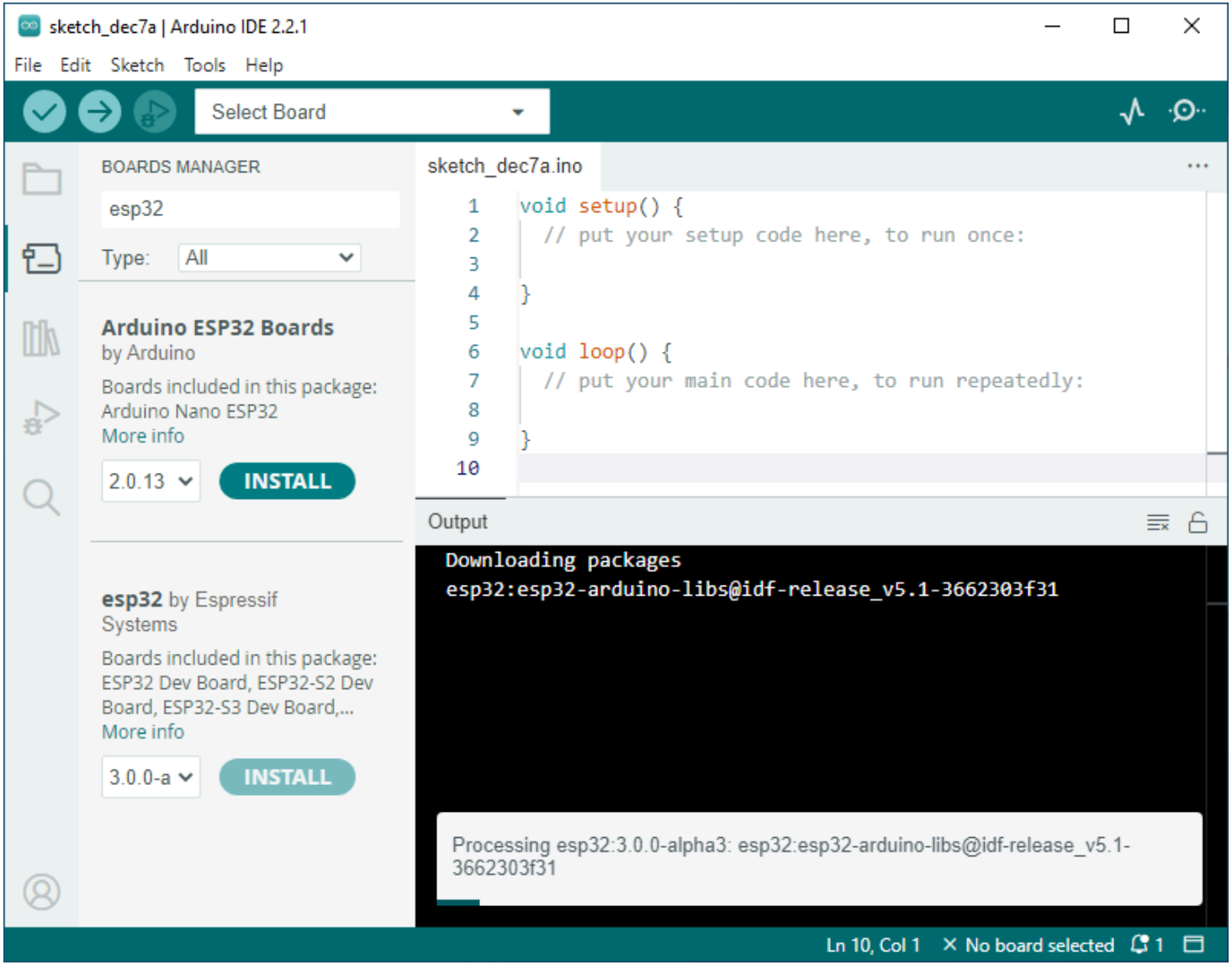


(/wiki/File:ESP32-C6-DEV-KIT-N8-Arduino06.png)

- Enter the development board manager, search for "esp32", select version 3.0.0-alpha3 under "esp32 by Espressif Systems" below, and click to install. (If installation fails, try using a mobile hotspot.)

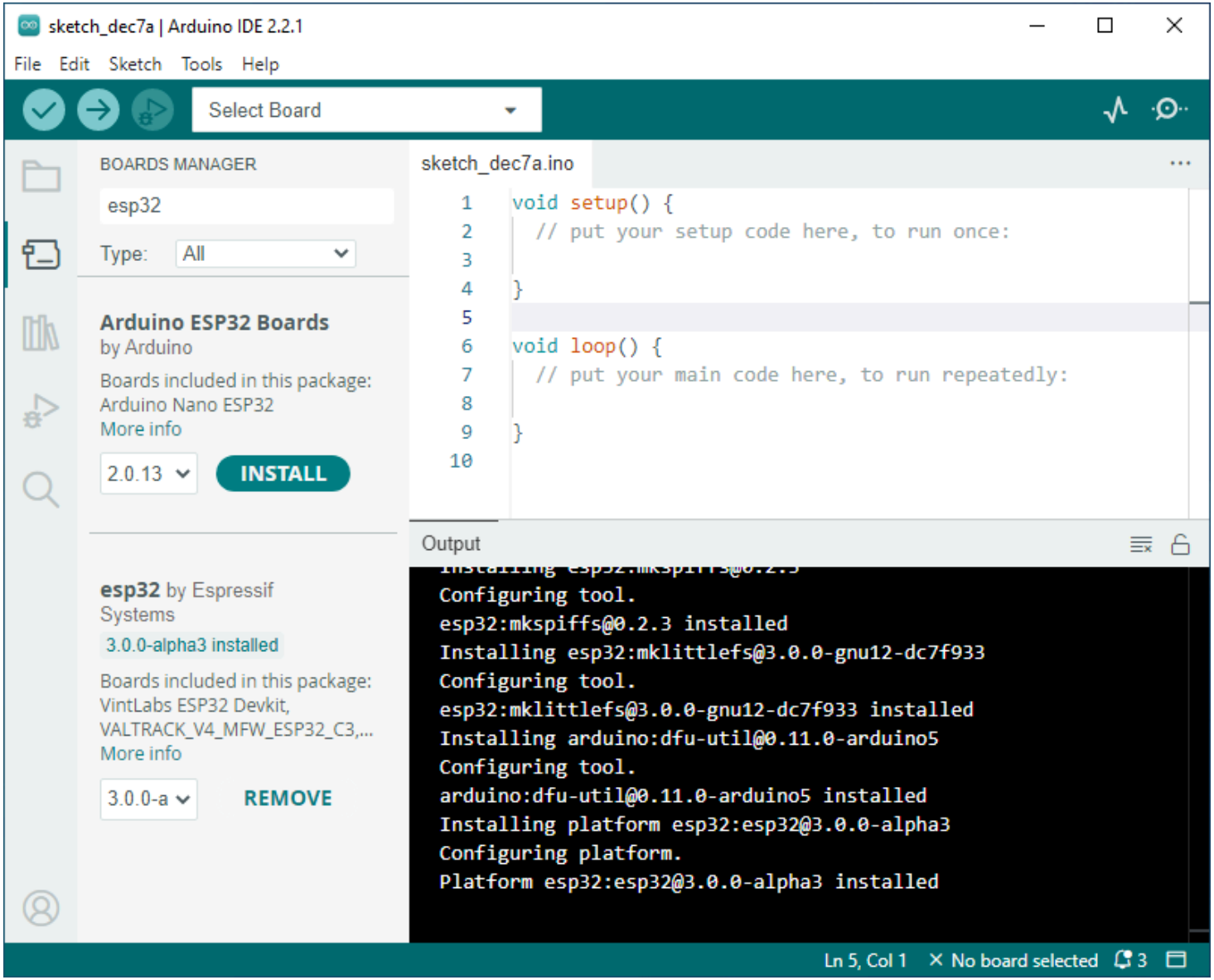


(/wiki/File:ESP32-C6-DEV-KIT-N8-Arduino07.png)



(/wiki/File:ESP32-C6-DEV-KIT-N8-Arduino08.png)

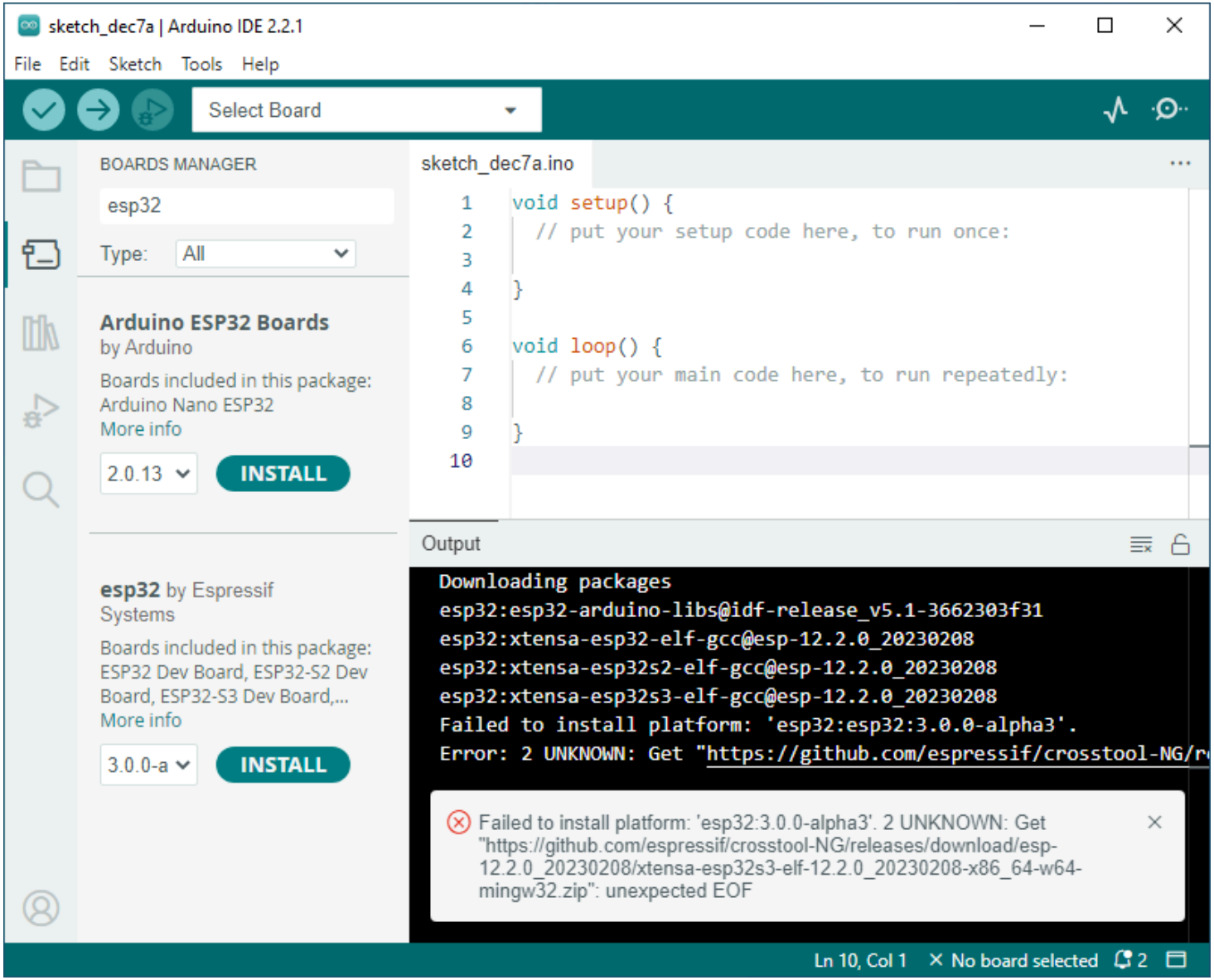
- Restart the Arduino IDE after installation, and then you can use it now.



(/wiki/File:ESP32-C6-DEV-KIT-N8-Arduino09.png)

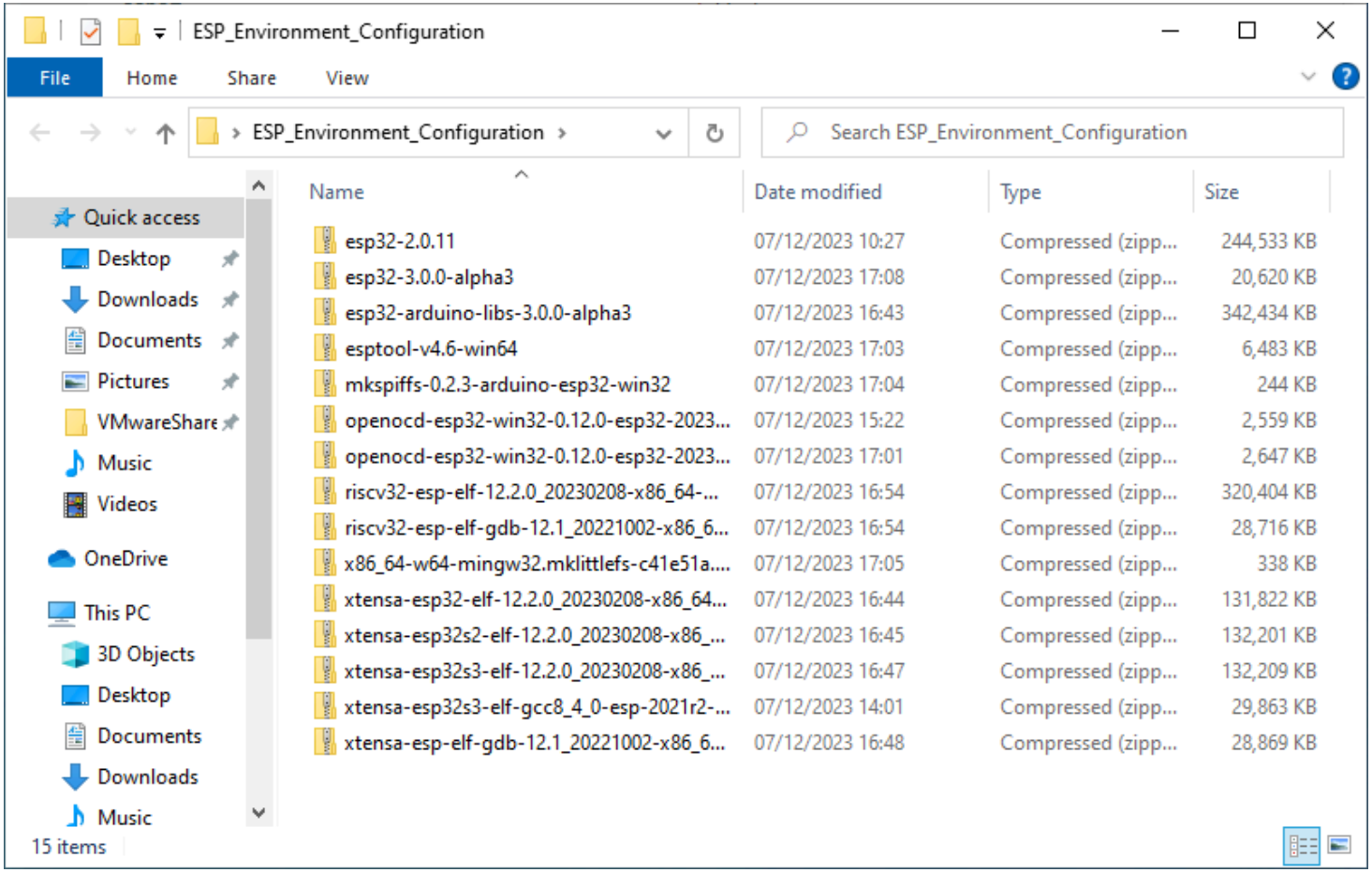
If the Installation Fails

- Failed to install version 3.0.0-alpha3:



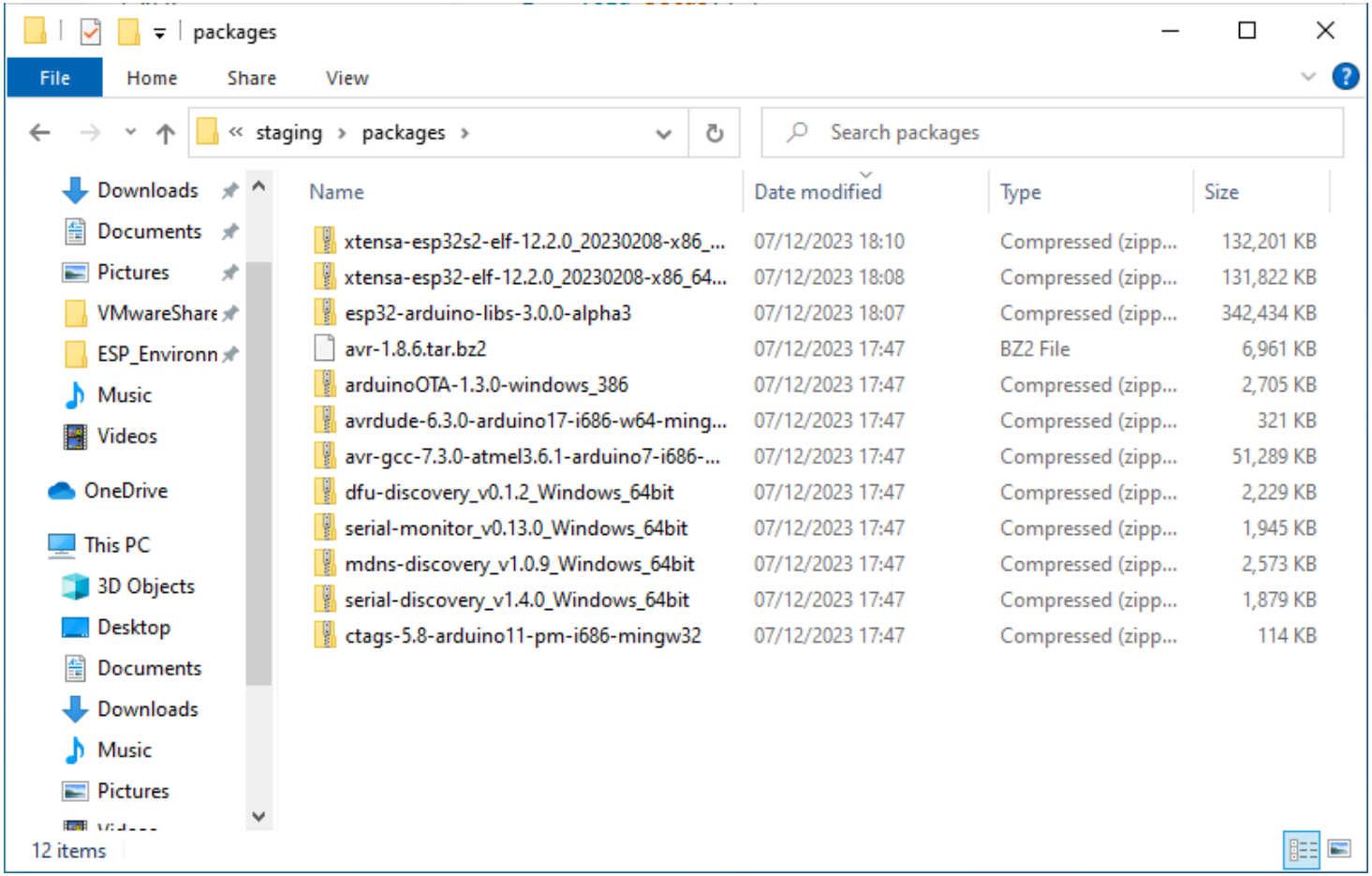
(/wiki/File:ESP32-C6-DEV-KIT-N8-install.png)

- Download the resource file (https://drive.google.com/file/d/19gMF3WHr4OreN26u2jYtGXKotRn8K16l/view?usp=sharing).



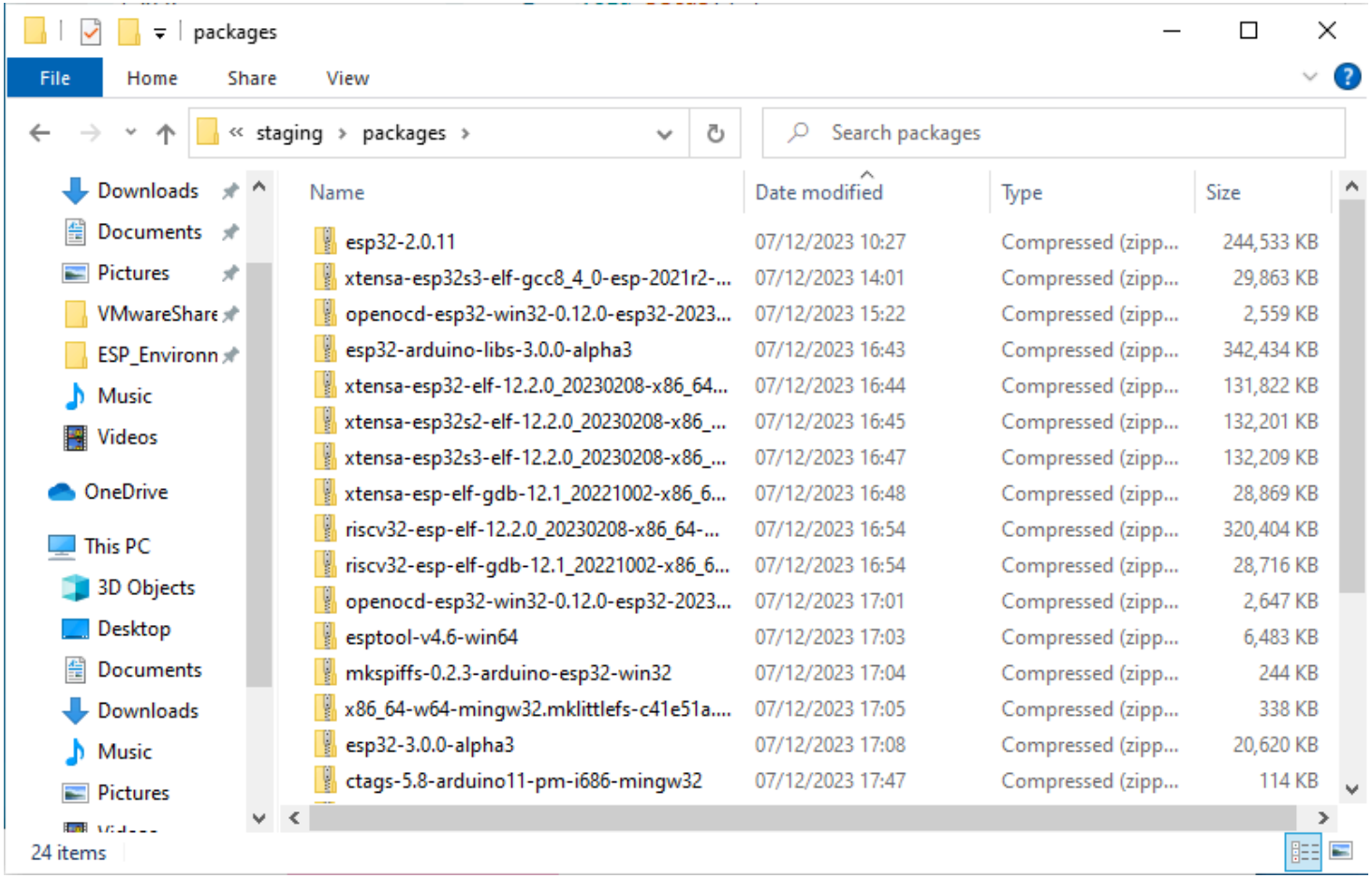
(/wiki/File:ESP32-C6-DEV-KIT-N8-install02.png)

- Click on the path "c:\Users\Waveshare\AppData\Local\Arduino15\packages" (where Waveshare is the user name of the computer, and you need to turn on Show Hidden Files).



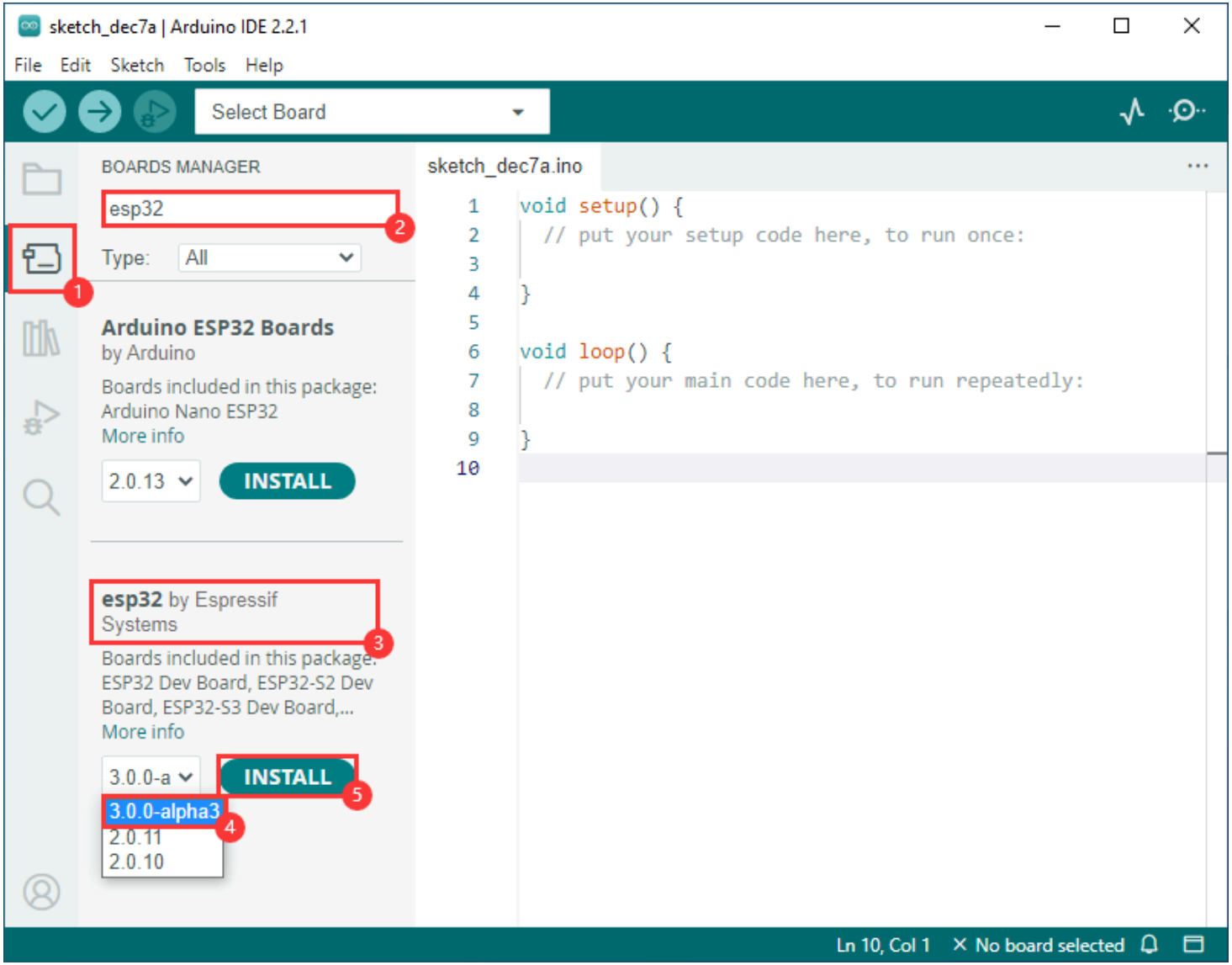
(/wiki/File:ESP32-C6-DEV-KIT-N8-install03.png)

- Unzip the downloaded files to the packages file folder.



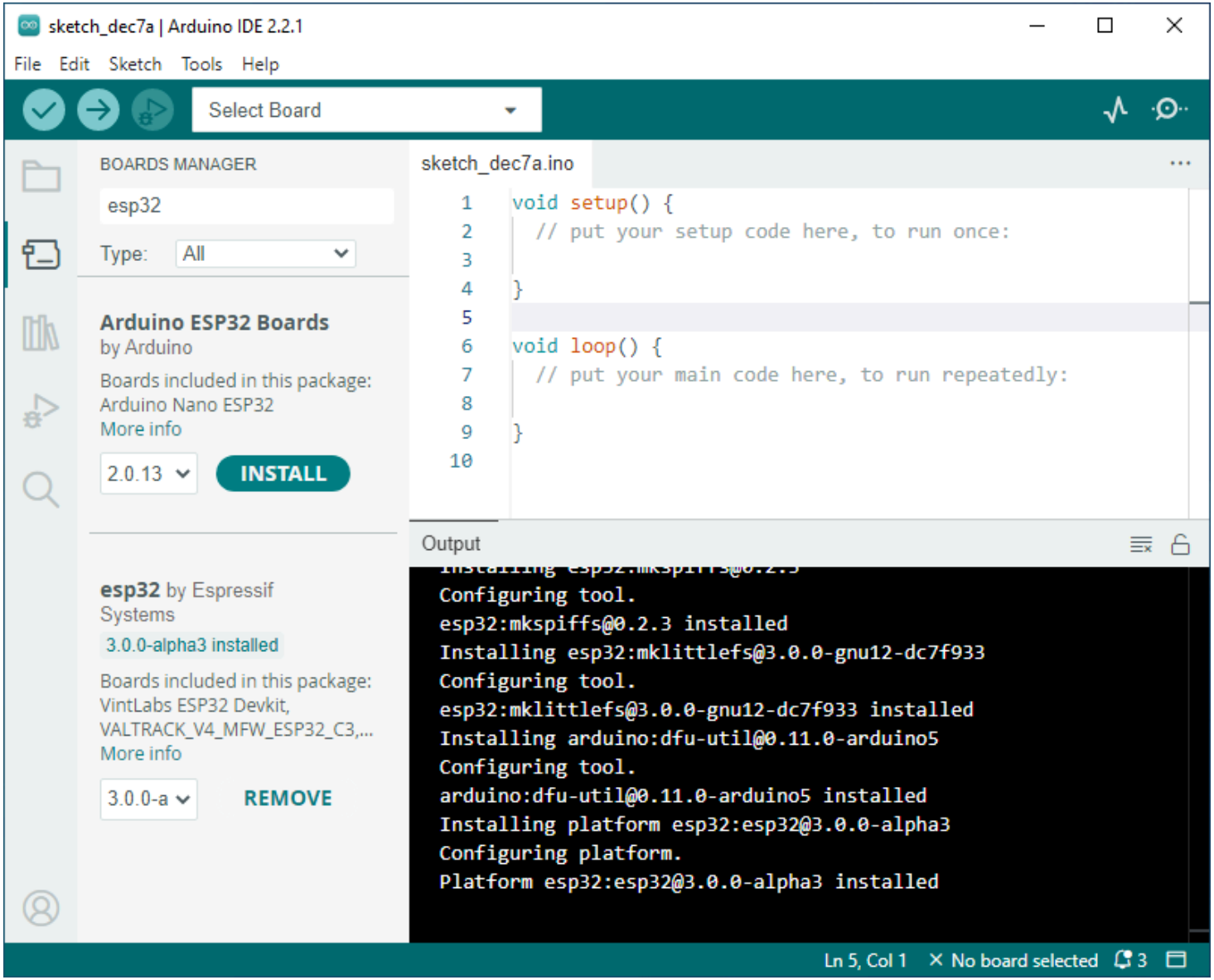
(/wiki/File:ESP32-C6-DEV-KIT-N8-install04.png)

- Install it again.



(/wiki/File:ESP32-C6-DEV-KIT-N8-Arduino07.png)

- Restart the Arduino IDE after installation and you're ready to go!

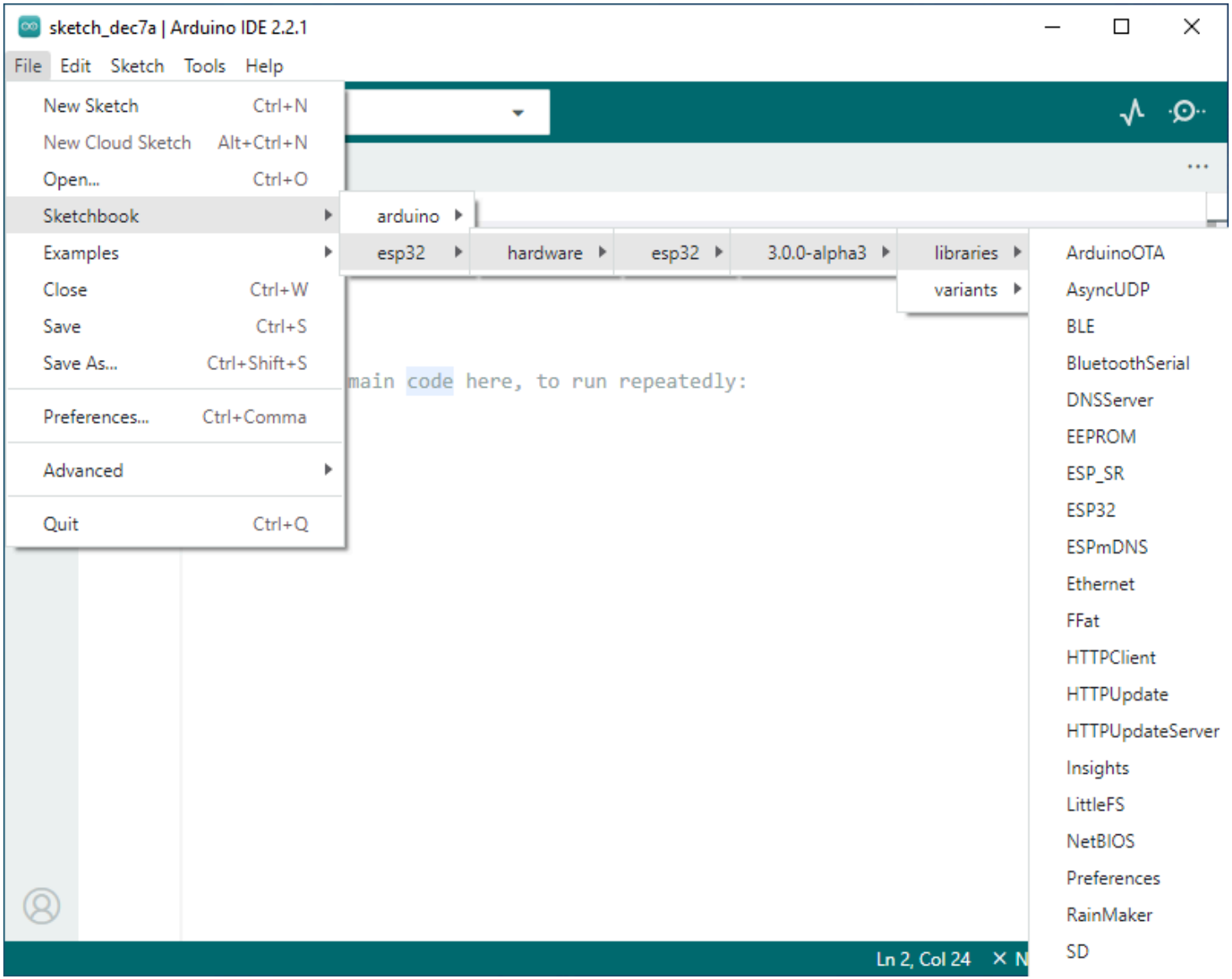


(/wiki/File:ESP32-C6-DEV-KIT-N8-Arduino09.png)

Create Example

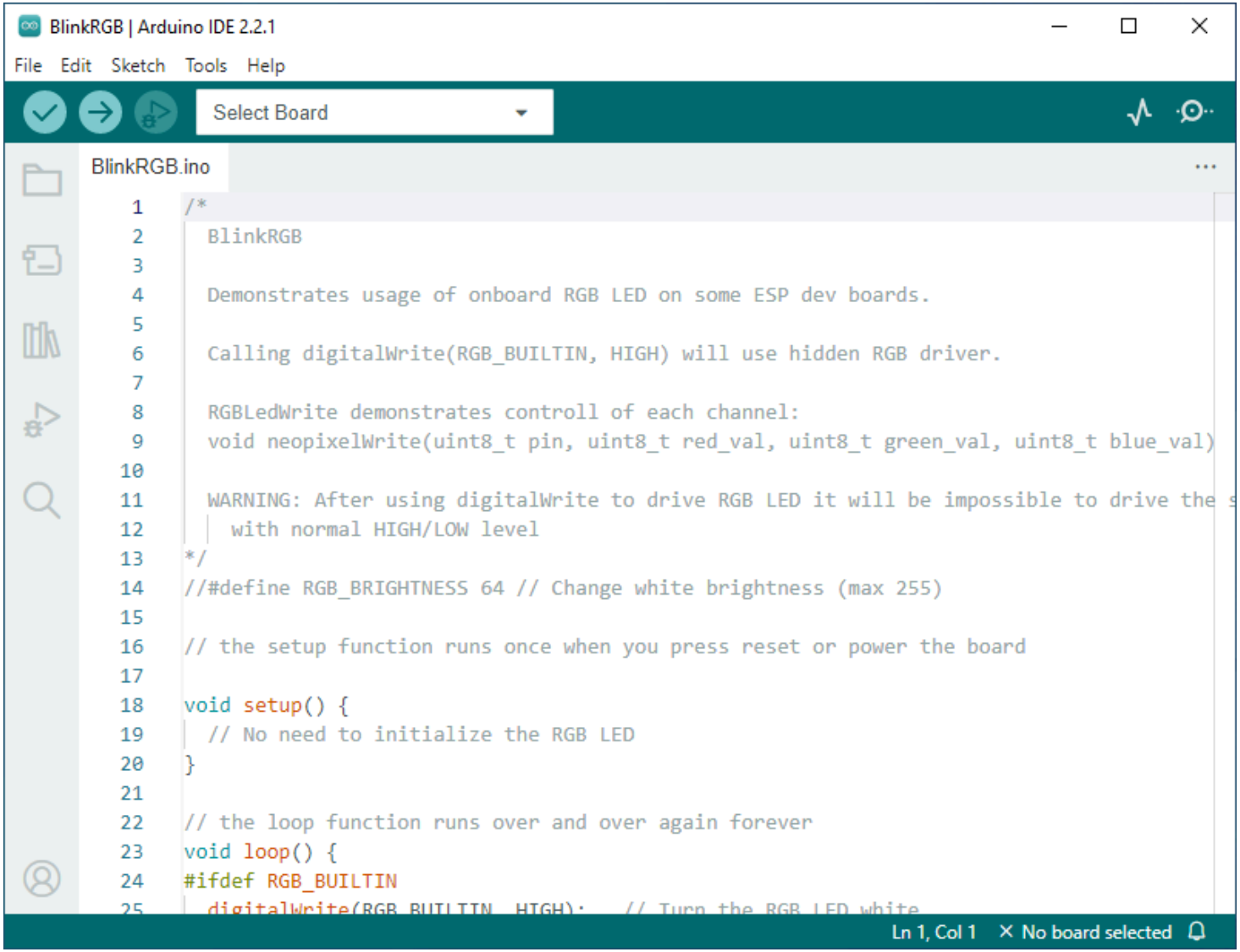
- After changing the project folder above to

c:\Users\Waveshare\AppData\Local\Arduino15\packages ("Waveshare" is the computer username), you can create demos using the examples in the project folder under the files.



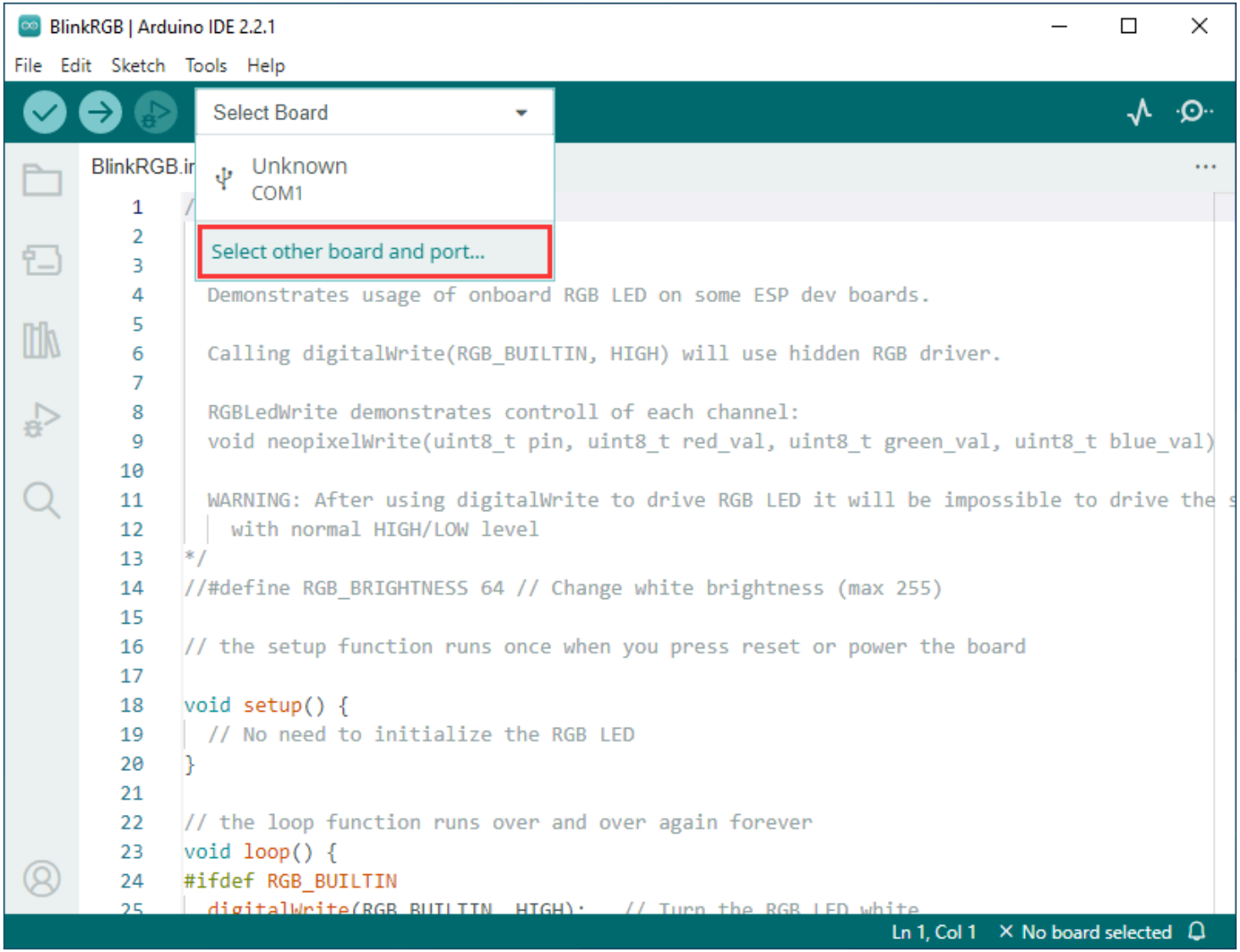
(/wiki/File:ESP32-C6-DEV-KIT-N8-Arduino10.png)

- The following is the RGB flashing example (File -> Sketchbook -> esp32 -> hardware -> esp32 -> 3.0.0-alpha3 -> libraries -> ESP32 -> examples -> BlinkRGB under GPIO).



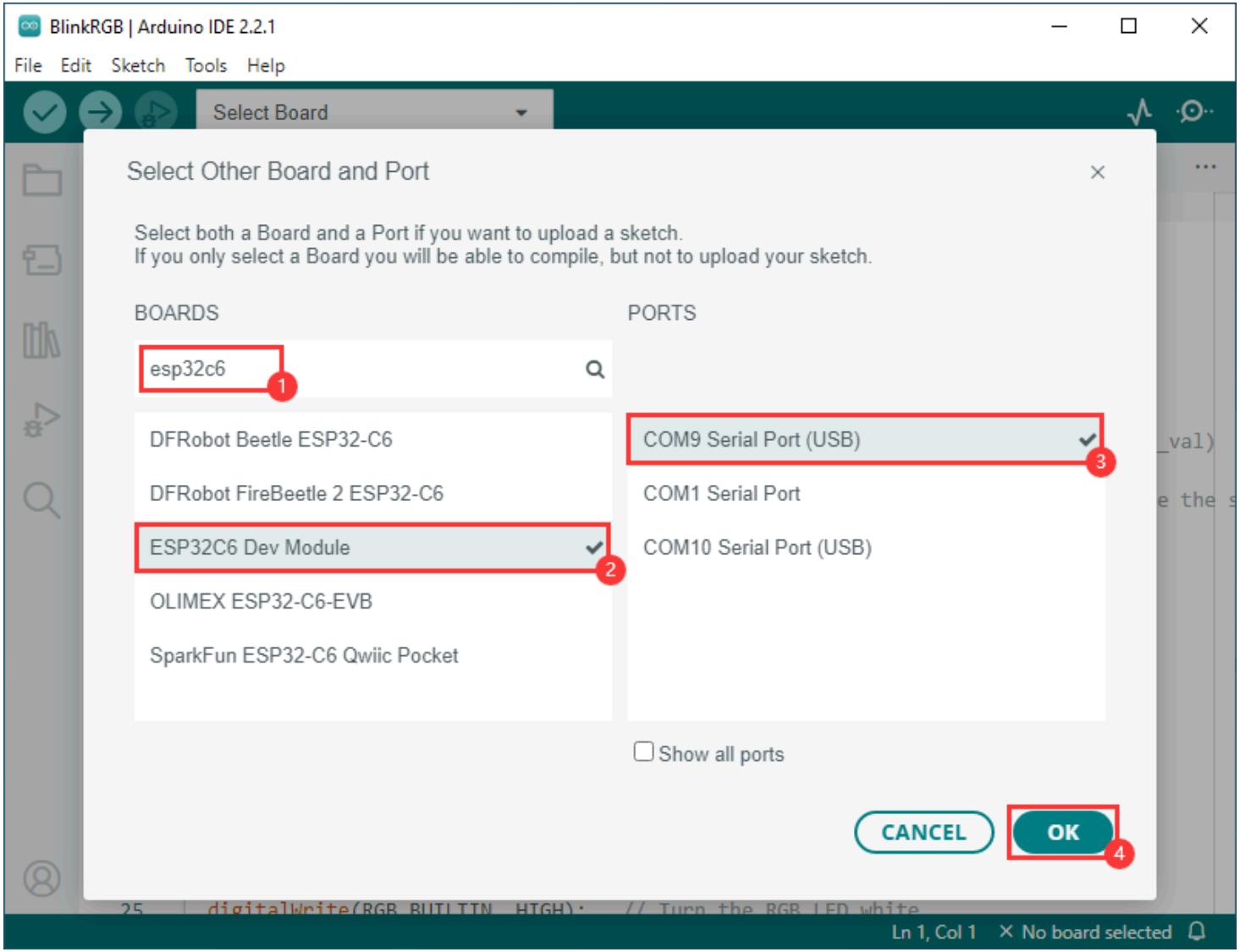
(/wiki/File:ESP32-C6-DEV-KIT-N8-Arduino11.png)

- Select the development board and port.



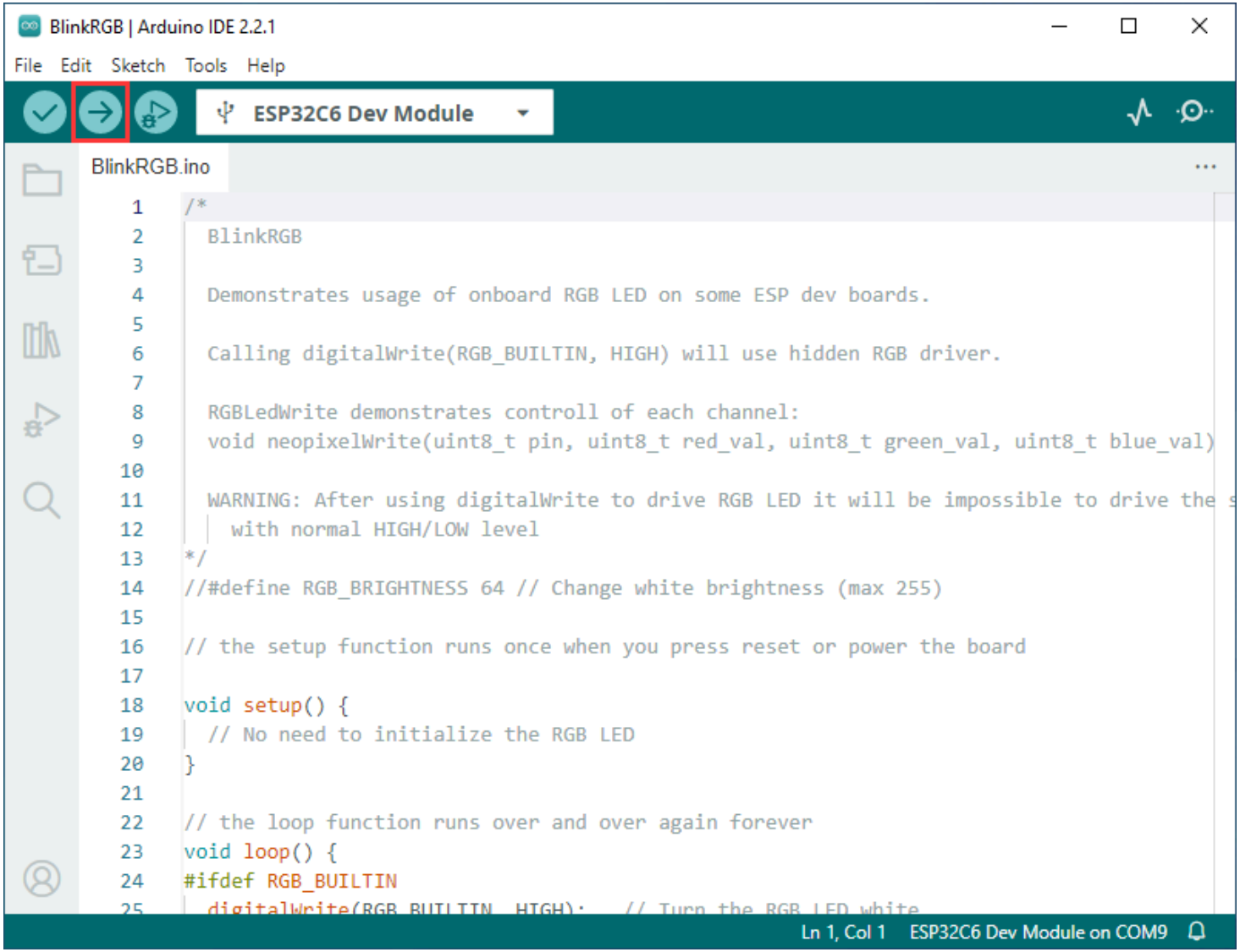
(/wiki/File:ESP32-C6-DEV-KIT-N8-Arduino12.png)

- Search esp32c6, select ESP32C6 Dev Module and the port to download.



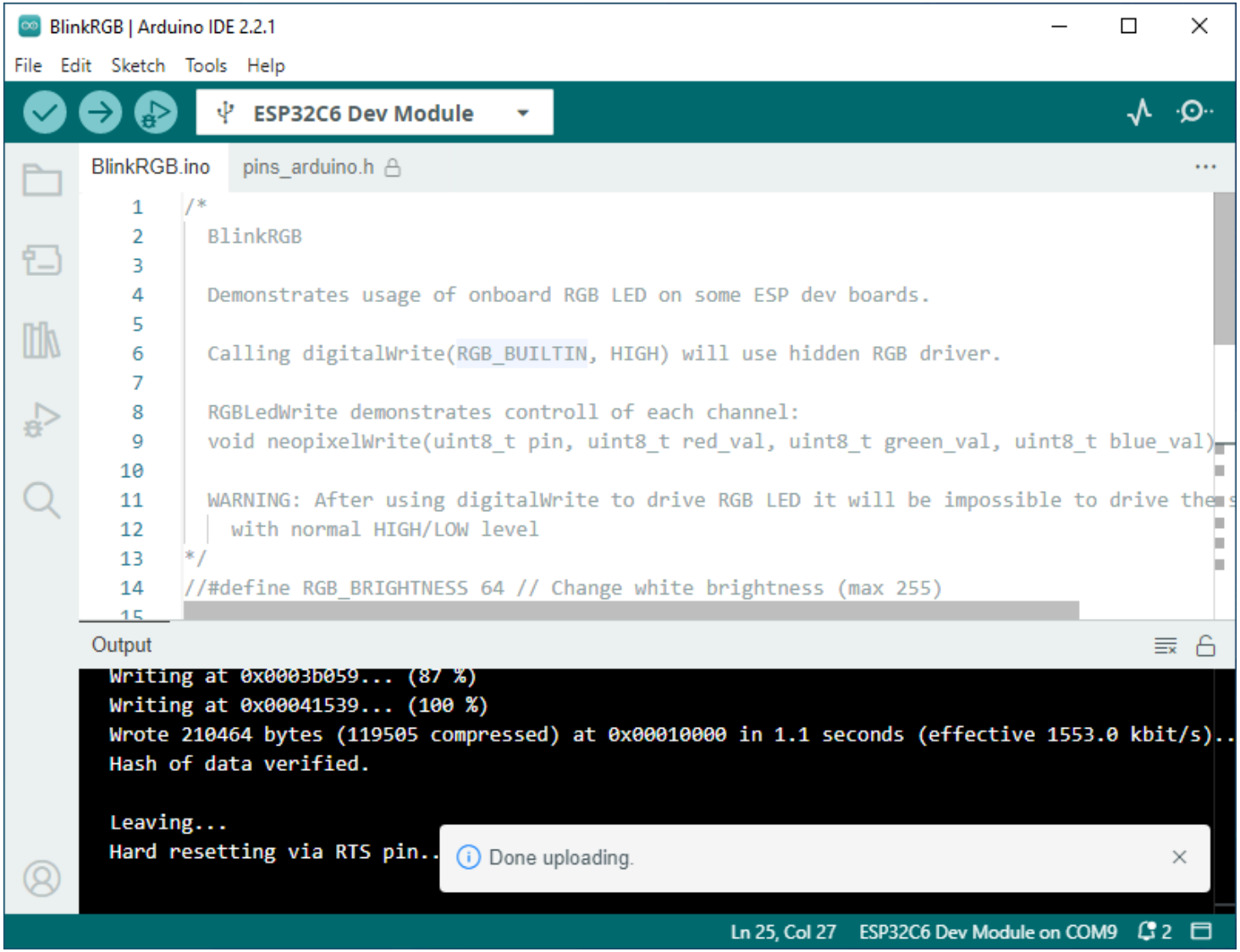
(/wiki/File:ESP32-C6-DEV-KIT-N8-Arduino13.png)

- After selecting, click to upload and Arduino IDE will start to compile and flash the demo.



(/wiki/File:ESP32-C6-DEV-KIT-N8-Arduino14.png)

- After uploading, you can see the effect on the development board.



(/wiki/File:ESP32-C6-DEV-KIT-N8-Arduino15.png)

Resources

Software

Compile

- VScode (https://code.visualstudio.com/download)
- Arduino IDE (https://www.arduino.cc/en/software)

UART

- SSCOM5.13.1 (https://files.waveshare.com/wiki/LC29H(XX)-GPS-RTK-HAT/Sscom5.13.1.zip)

Flash

- Flash (https://files.waveshare.com/wiki/ESP32-C6-DEV-KIT-N8/Flash_download_tool_3.9.5_0.zip)

Bluetooth

- Bluetooth debugging assistant (https://files.waveshare.com/wiki/ESP32-C6-DEV-KIT-N8/ESP32-C6_TO_BLEAssist.ZIP)

Schematic

- Schematic (https://files.waveshare.com/wiki/ESP32-C6-Zero/ESP32-C6-Zero-Sch.pdf)

3D/2D Files

- 3D File (https://files.waveshare.com/wiki/ESP32-C6-Zero/ESP32-C6-Zero-3D.zip)

- 2D File (https://files.waveshare.com/wiki/ESP32-C6-Zero/ESP32-C6-Zero-2D.zip)

Datasheets

- ESP32-C6 Technical Reference Manual (https://files.waveshare.com/wiki/ESP32-C6-DEV-KIT-N8/ESP32-C6_Technical_Reference_Manual.pdf)
- ESP32-C6 Series Datasheet (https://files.waveshare.com/wiki/ESP32-C6-Pico/ESP32-C6_Series_Datasheet.pdf)

Official Datasheet

- ESP-IDF Datasheet (https://docs.espressif.com/projects/esp-idf/en/latest/esp32c6/index.html)

FAQ

Question:The module appears to be reset all the time, and the recognition status will be flashing when viewed from the device manager?

Answer:

This situation may be caused by the Flash blank causing the USB port to be unstable, you can long-press the BOOT button, press RESET at the same time, and then release RESET, and then release the BOOT button, at this time the module can enter the download mode to flash the firmware (program) to solve the situation.

Question:After the module downloads the demo and re-downloads it,

sometimes it fails to connect to the serial port, or the flashing fails?

Answer:

- Method 1: Press the reset button for more than one second, wait for the PC to recognize the device again, and then download it again.
- Method 2: You can long-press the BOOT button, simultaneously press the RESET button, then release the RESET button, and finally release the BOOT button. This will put the module into download mode and can resolve most download issues.

Question:No ESP option below when setting up an environment or building a project?

Answer:

In VSCode, click the shortcut **F1**, and search for **Espressif IDF**, you will find that it is designated as an untrusted extension, set it as trusted.

Question:Switch to the same ESP model and encounter issues with program flashing and program execution?

Answer:

Please select the COM port and driver object again after switching ESP, then compile and flash.

Question:After powering up the module, the recognized serial devices and USB ports keep resetting and restarting?

Answer:

Check whether the power supply voltage for the USB port is less than 5V, in general, if it is 4.9V or more, the module's two USB ports can be used normally. If it is lower than 4.9V, the power supply may be insufficient and the USB port may disconnect. In this case, you should replace it with a USB port with sufficient voltage.

Support

Technical Support

If you need technical support or have any feedback/review, please click the **Submit Now** button to submit a ticket, Our support team will check and reply to you within 1 to 2 working days. Please be patient as we make every effort to help you to resolve the

Submit Now (https://service.waveshare.com/)

issue.

Working Time: 9 AM - 6 PM GMT+8

(Monday to Friday)