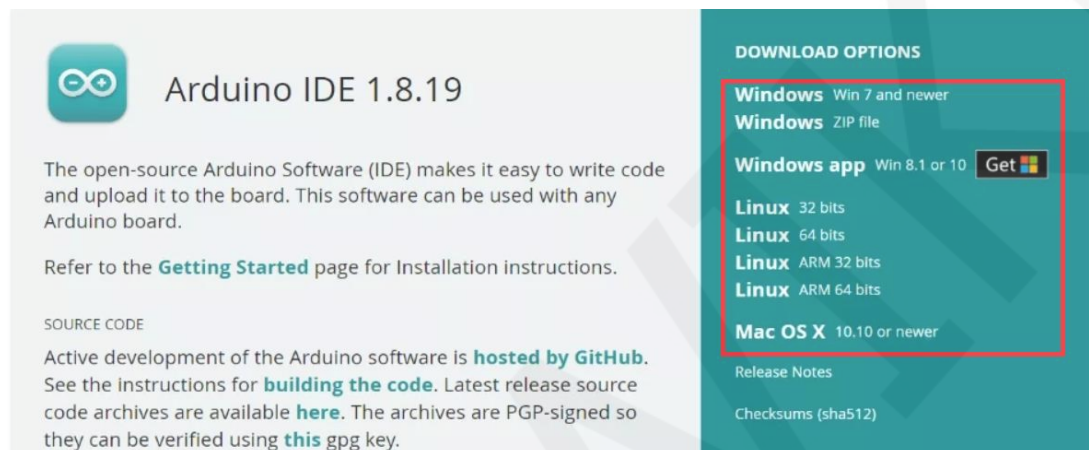


## 1. Installing the Arduino IDE

Download the installation package from the Arduino official website.

Download address:<https://www.arduino.cc/en/software>

Download the corresponding installation package according to your PC system, as shown in the following figure (the version in the picture may not be the latest version, and the download interface may not be the latest):



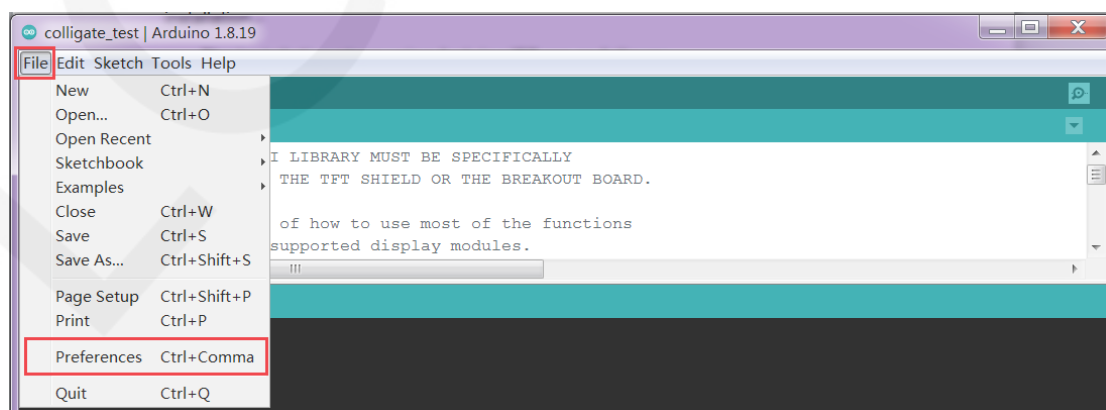
After the download is completed, unzip and click Install.

## 2. Install Arduino-ESP32 support library

There are two installation methods: using the Arduino IDE for installation and manual installation.

**The installation steps using Arduino IED are as follows:**

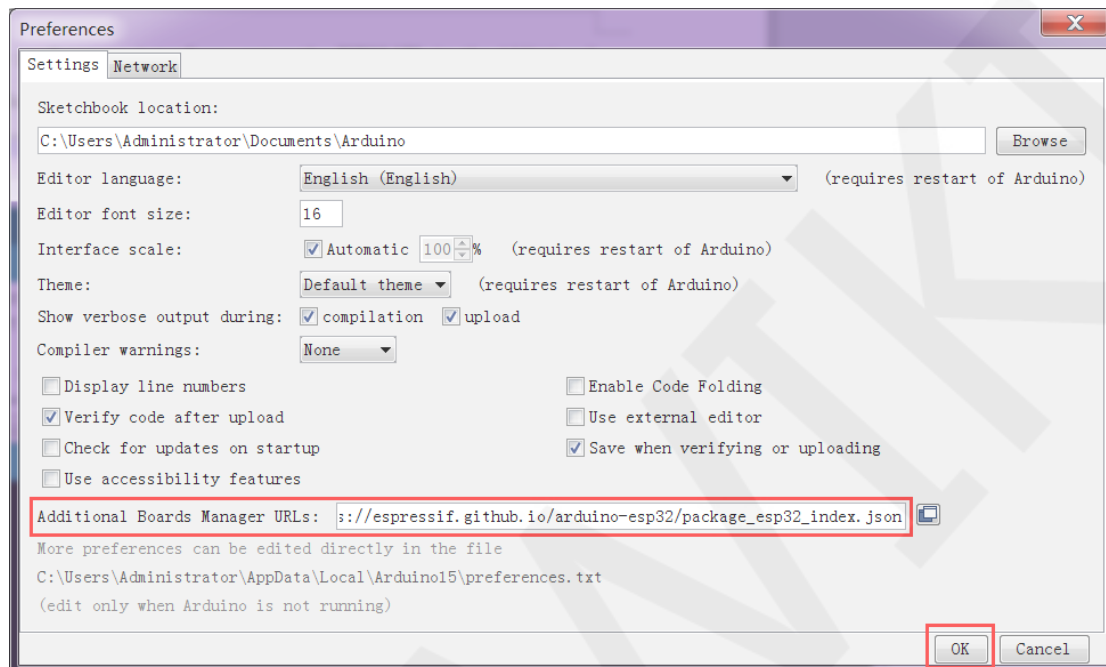
- A. Open the Arduino IED software, click on the **File -> Preferences** button, as shown in the following figure:



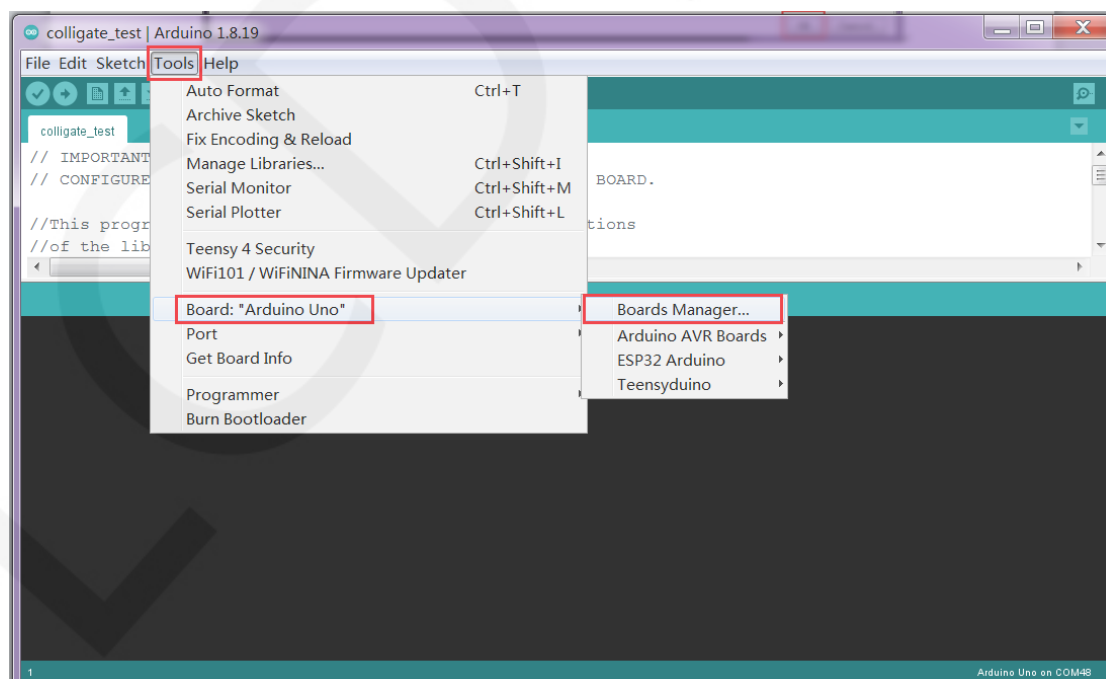
- B. In the pop-up preferences interface, enter the ESP32 URL in the **Additional Boards Manager URLs**:

[https://espressif.github.io/arduino-esp32/package\\_esp32\\_index.json](https://espressif.github.io/arduino-esp32/package_esp32_index.json)

Then click the "OK" button to save and exit, as shown in the following figure:

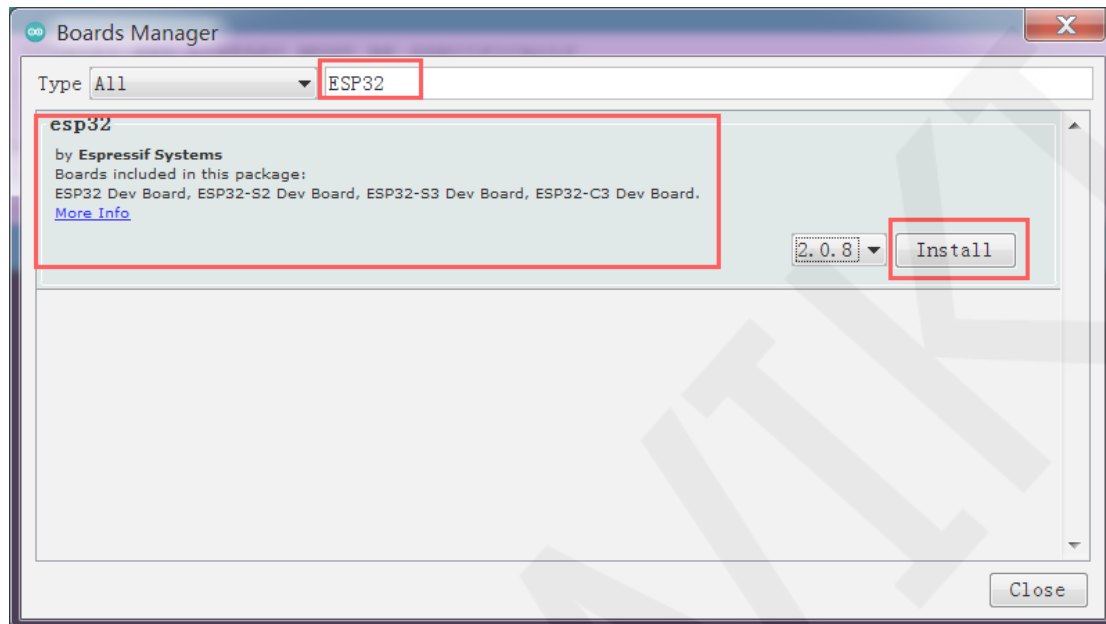


- C. Click the **Tools** button, then select the **Board->Boards Manager** option, as shown in the following figure:

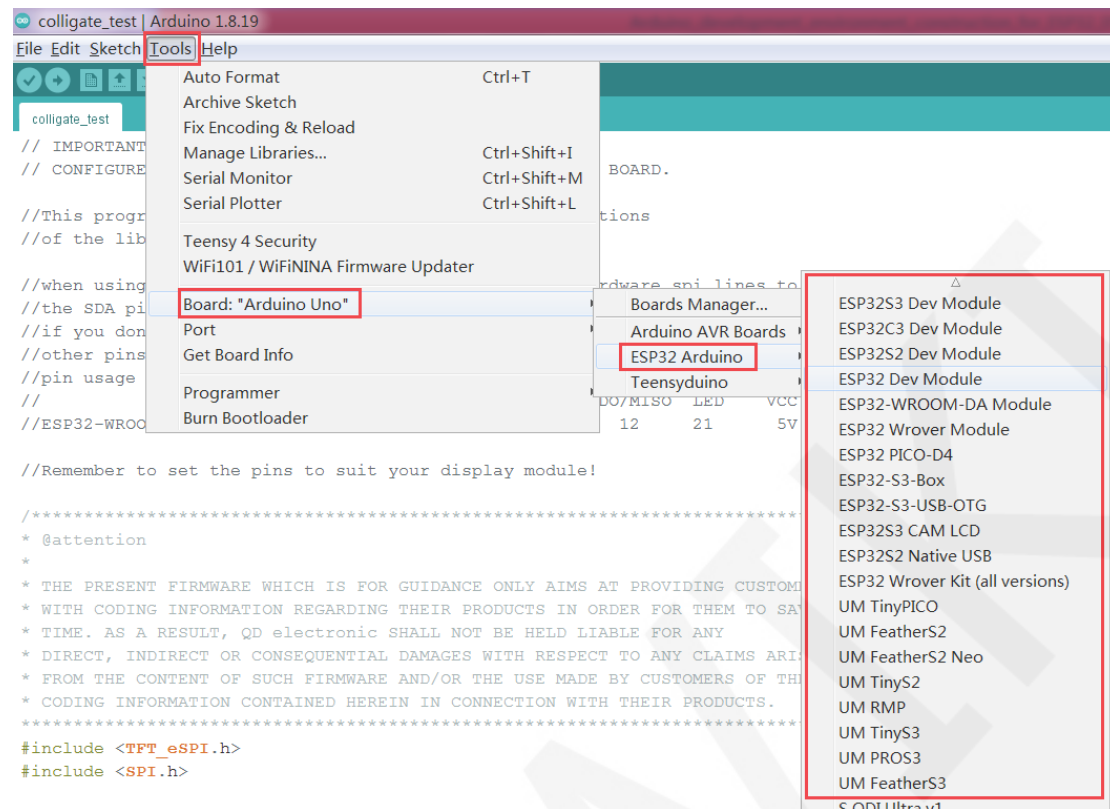


- D. Click the **Boards Manager** button to enter the Boards Manager interface. After

the interface is loaded, enter "**ESP32**" in the search bar to search for the Arduino-ESP32 support library. After the search is completed, click the "**Install**" button to install, as shown in the following figure:



E. The installation time is relatively long. If there is an error during the installation process, try it multiple times. After the installation is completed, click the "**Tools**" button and select the **Board** option. You will see the **ESP32 Arduino** option. After selecting this option, various ESP32 development board models will appear, as shown in the following figure:



If Arduino IDEs cannot be successfully installed (possibly due to poor network environment, resulting in continuous download errors), then manual installation is necessary.

#### The steps for manual installation are as follows:

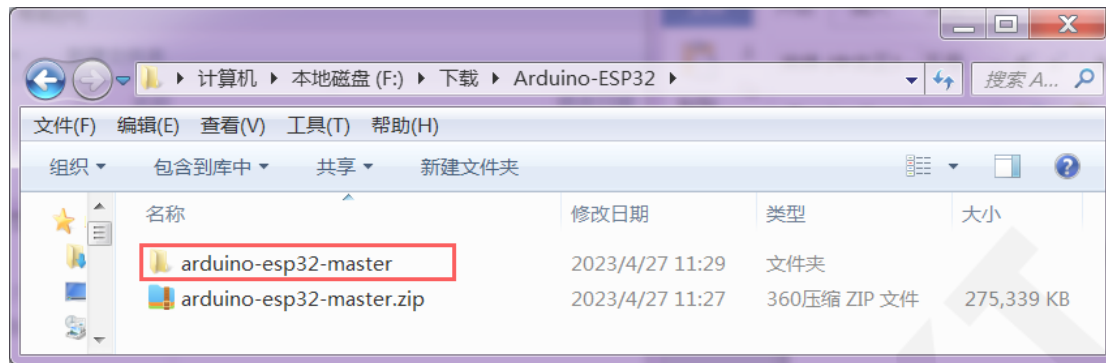
- A. Download the Arduino-ESP32 support library at the following address:

<https://github.com/espressif/arduino-esp32>

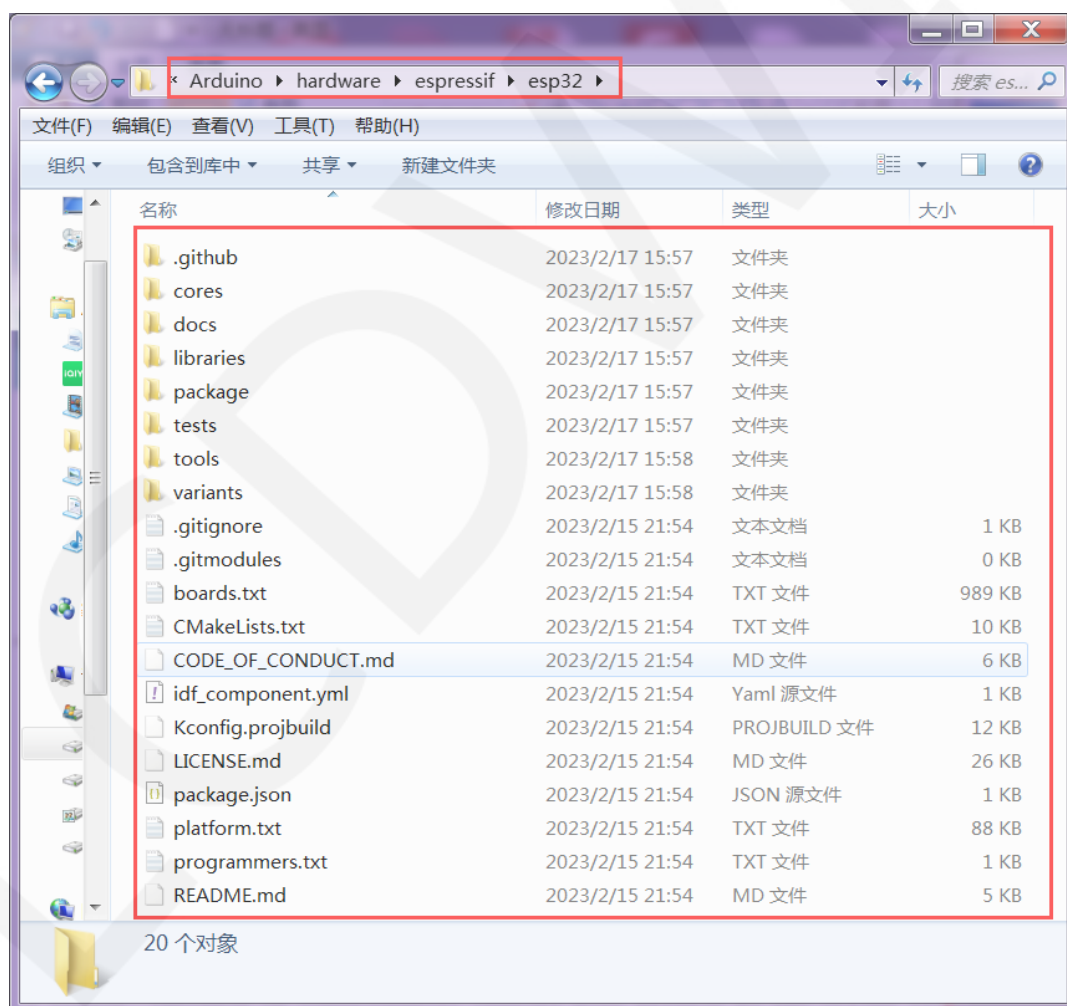
If the above address cannot be downloaded, it is recommended to use the following download address:

<https://gitcode.net/mirrors/espressif/arduino-esp32>

- B. After successfully downloading the support library, unzip the file as shown in the following figure:

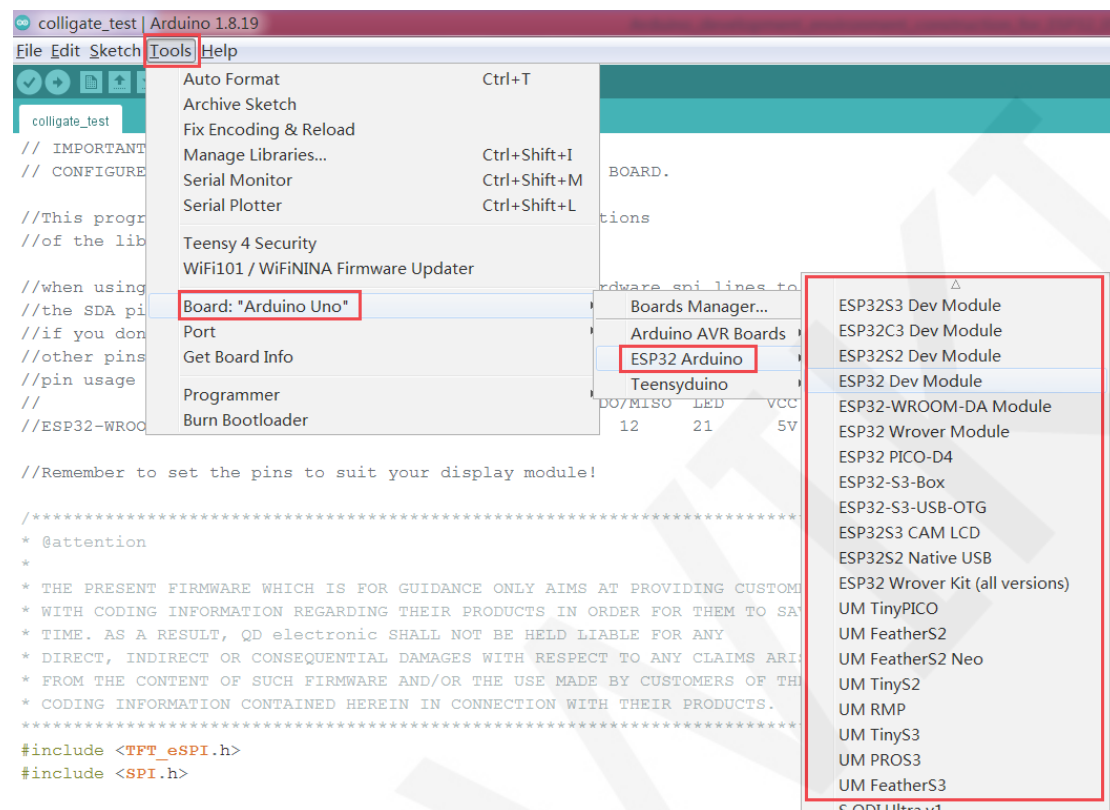


- C. Open the Arduino IED installation directory, create a new **espressif** directory in the **hardware** directory, and then create an **esp32** directory in the **espressif** directory. Finally, copy all the support library files extracted in the previous step (i.e. all files in the **arduino-esp32-master** directory) to the **esp32** directory, as shown in the following figure:



- D. Open the **Arduino IED** software, click the **"Tools"** button, select the **board** option,

and you will see the **ESP32 Arduino** option. After selecting this option, various ESP32 development board models will appear, as shown in the following figure:



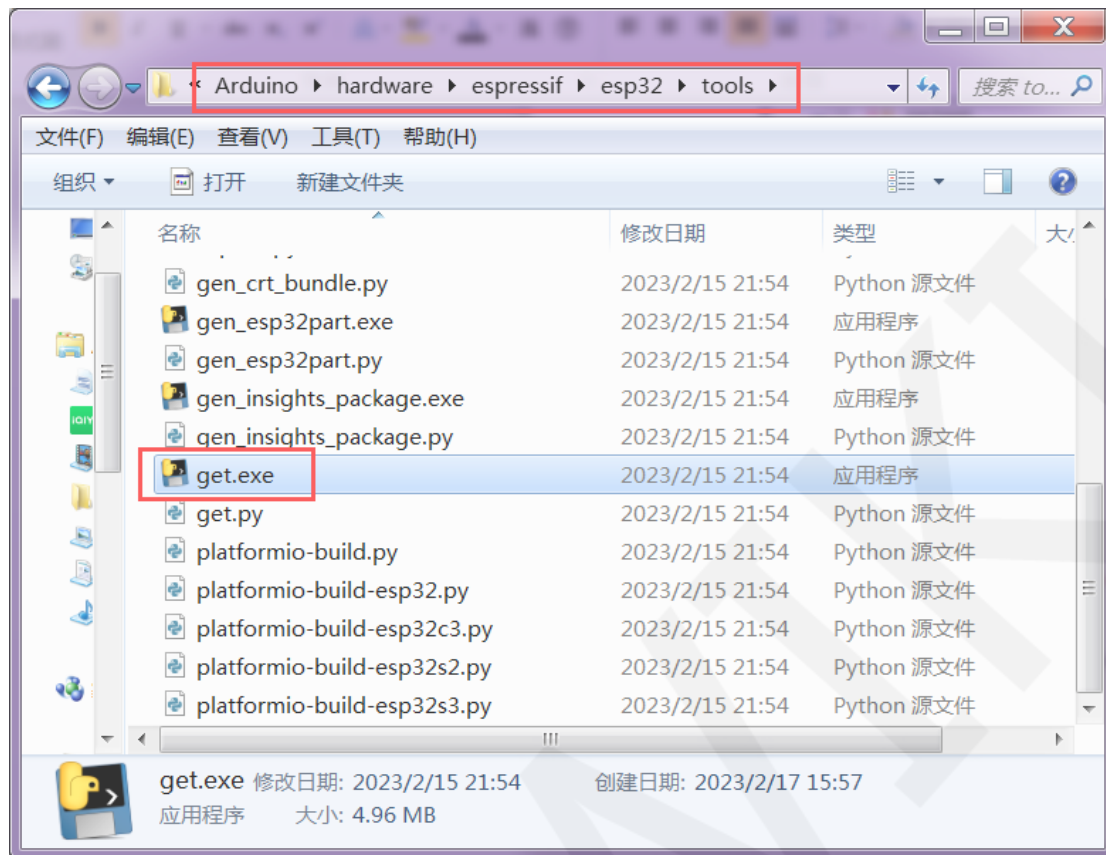
### 3. Install Compilation Tools

This step is only for installing the Arduino-ESP32 support library using manual methods. If it is installed using the Arduino IDE method, this step is not necessary.

There are two methods for installing compilation tools: **executing script installation** and **manual installation**.

The installation steps for executing the script are as follows:

- Open the "**hardware\espressif\esp32\tools**" directory under the Arduino IDE installation directory and locate the "**get.exe**" file, as shown in the following figure:



- B. Double click on the '**get.exe**' file to download and install the compilation tool, which takes a bit longer to execute.

If the get.exe program cannot install the compilation tool, then manual installation is required.

**The manual installation steps are as follows:**

- A. Open the "**hardware\espressif\esp32\package**" directory under the Arduino IED installation directory and locate the "**package\_esp32\_index.template.json**" file, as shown in the following figure:





- B. Open "package\_esp32\_index.template.json" file, Find the download links for the three files: "xtensa-esp32-elf-gcc", "esptool.py", and "mkspiffs". Wherein:

**xtensa-esp32-elf-gcc** is the ESP32 cross compilation toolchain

**esptool.py** is a tool for burning (downloading) ESP32 software

**mkspiffs** is the file system used on SPI NOR flash devices

Each file corresponds to download links for multiple platforms, such as Linux, Apple, etc. Here, only the i686-mingw32 platform, which is the 32-bit Windows platform, is required. Of course, if you are using a 64 bit Windows system, you can also use x86\_ Link to the 64 mingw32 platform. The three file links are obtained as shown in the following figure:

**xtensa-esp32-elf-gcc:**

```
"name": "xtensa-esp32-elf-gcc",
"version": "gcc8_4_0-esp-2021r2-patch5",
"systems": [
  {
    "host": "x86_64-pc-linux-gnu",
    "url": "https://github.com/espressif/crosstool-NG/releases/download/esp-2021r2-patch5/xtensa-esp32-elf-gcc8_4_0-esp-2021r2-patch5-linux-amd64.tar.gz",
    "archiveFileName": "xtensa-esp32-elf-gcc8_4_0-esp-2021r2-patch5-linux-amd64.tar.gz",
    "checksum": "SHA-256:8ef14e0409c2011b41e504a30f70d3e35287313a795d1f2462ad2cd0e2052d37",
    "size": "94397702"
  },
  {
    "host": "x86_64-apple-darwin",
    "url": "https://github.com/espressif/crosstool-NG/releases/download/esp-2021r2-patch5/xtensa-esp32-elf-gcc8_4_0-esp-2021r2-patch5-macos.tar.gz",
    "archiveFileName": "xtensa-esp32-elf-gcc8_4_0-esp-2021r2-patch5-macos.tar.gz",
    "checksum": "SHA-256:19af109fda024a3a4c989f7ccaa104f9b1b74cfddc9363e730bb8cb9b50d5dc4",
    "size": "101712946"
  },
  {
    "host": "i686-mingw32",
    "url": "https://github.com/espressif/crosstool-NG/releases/download/esp-2021r2-patch5/xtensa-esp32-elf-gcc8_4_0-esp-2021r2-patch5-win32.zip",
    "archiveFileName": "xtensa-esp32-elf-gcc8_4_0-esp-2021r2-patch5-win32.zip",
    "checksum": "SHA-256:9851c2cfa355e1fad8abfb643a1c945d27385b1851f3ae468915ea78fcbec940",
    "size": "118610020"
  },
  {
    "host": "x86_64-mingw32",
    "url": "https://github.com/espressif/crosstool-NG/releases/download/esp-2021r2-patch5/xtensa-esp32-elf-gcc8_4_0-esp-2021r2-patch5-win64.zip",
    "archiveFileName": "xtensa-esp32-elf-gcc8_4_0-esp-2021r2-patch5-win64.zip",
    "checksum": "SHA-256:a328b3c55631846241bbe7999a309b20b797c8dc50b6e8dcf463e66a2da5fb4",
    "size": "121846722"
  }
]
```

**esptool.py:**



```

"name": "esptool_py",
"version": "4.2.1",
"systems": [
  {
    "host": "i686-mingw32",
    "url": "https://github.com/espressif/arduino-esp32/releases/download/2.0.4/esptool-4.2.1-windows.zip",
    "archiveFileName": "esptool-4.2.1-windows.zip",
    "checksum": "SHA-256:582560067bfb9895f4862eb5fd87558dde5d4d30e7575c9b8bcb0dd60fd94",
    "size": "6368279"
  },
  {
    "host": "x86_64-mingw32",
    "url": "https://github.com/espressif/arduino-esp32/releases/download/2.0.4/esptool-4.2.1-windows.zip",
    "archiveFileName": "esptool-4.2.1-windows.zip",
    "checksum": "SHA-256:582560067bfb9895f4862eb5fd87558dde5d4d30e7575c9b8bcb0dd60fd94",
    "size": "6368279"
  },
  {
    "host": "x86_64-apple-darwin",
    "url": "https://github.com/espressif/arduino-esp32/releases/download/2.0.4/esptool-4.2.1-macos.tar.gz",
    "archiveFileName": "esptool-4.2.1-macos.tar.gz",
    "checksum": "SHA-256:a984f7ad8bdb40c42d0d368bf4bb21b69a9587aed46b7b6d7de23ca58a3f150d",
    "size": "5816598"
  },
  {
    "host": "x86_64-pc-linux-gnu",
    "url": "https://github.com/espressif/arduino-esp32/releases/download/2.0.4/esptool-4.2.1-linux.tar.gz",
    "archiveFileName": "esptool-4.2.1-linux.tar.gz",
    "checksum": "SHA-256:5a45fb77eb6574554ec2f45230d0b350f26f9c24ab3b6c13c4031ebdf72a34ab",
    "size": "90123"
  }
]

```

### mkspiffs:

```

"name": "mkspiffs",
"version": "0.2.3",
"systems": [
  {
    "host": "i686-mingw32",
    "url": "https://github.com/igrr/mkspiffs/releases/download/0.2.3/mkspiffs-0.2.3-arduino-esp32-win32.zip",
    "archiveFileName": "mkspiffs-0.2.3-arduino-esp32-win32.zip",
    "checksum": "SHA-256:b647f2c2ef6949819c85ea9404271b55c7c9c25bcb98d3b98a1d0ba771adf56",
    "size": "249809"
  },
  {
    "host": "x86_64-apple-darwin",
    "url": "https://github.com/igrr/mkspiffs/releases/download/0.2.3/mkspiffs-0.2.3-arduino-esp32-osx.tar.gz",
    "archiveFileName": "mkspiffs-0.2.3-arduino-esp32-osx.tar.gz",
    "checksum": "SHA-256:9f43fc74a858cf564966b5035322c3e5e61c31a647c5a1d71b388ed6efc48423",
    "size": "130270"
  },
  {
    "host": "i386-apple-darwin",
    "url": "https://github.com/igrr/mkspiffs/releases/download/0.2.3/mkspiffs-0.2.3-arduino-esp32-osx.tar.gz",
    "archiveFileName": "mkspiffs-0.2.3-arduino-esp32-osx.tar.gz",
    "checksum": "SHA-256:9f43fc74a858cf564966b5035322c3e5e61c31a647c5a1d71b388ed6efc48423",
    "size": "130270"
  },
  {
    "host": "x86_64-pc-linux-gnu",
    "url": "https://github.com/igrr/mkspiffs/releases/download/0.2.3/mkspiffs-0.2.3-arduino-esp32-linux64.tar.gz",
    "archiveFileName": "mkspiffs-0.2.3-arduino-esp32-linux64.tar.gz",
    "checksum": "SHA-256:5e1a4ff41385e842f389f6b5254102a547e566a06b49babeffa93ef37115cb5d",
    "size": "50646"
  }
]

```

The three extracted links are as follows:

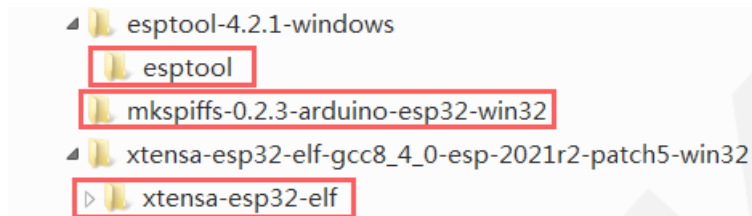
[https://github.com/espressif/crosstool-NG/releases/download/esp-2021r2-patch5/xtensa-esp32-elf-gcc8\\_4\\_0-esp-2021r2-patch5-win32.zip](https://github.com/espressif/crosstool-NG/releases/download/esp-2021r2-patch5/xtensa-esp32-elf-gcc8_4_0-esp-2021r2-patch5-win32.zip)

<https://github.com/espressif/arduino-esp32/releases/download/2.0.4/esptool-4.2.1-windows.zip>

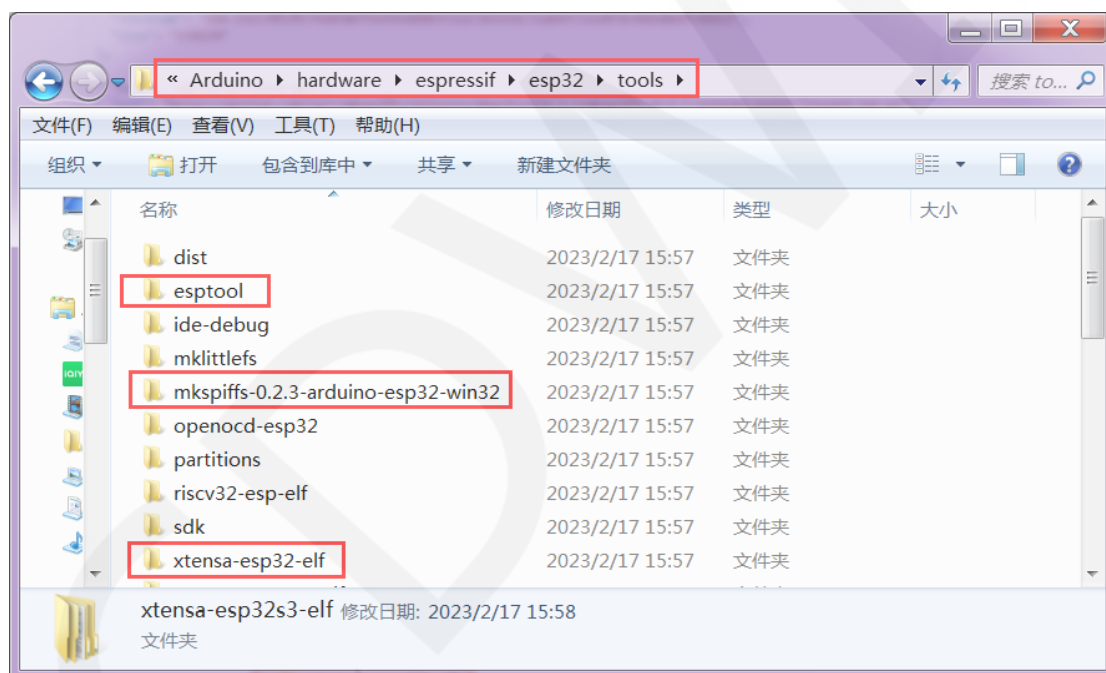
<https://github.com/igrr/mkspiffs/releases/download/0.2.3/mkspiffs-0.2.3-arduino-esp32-win32.zip>

You can directly click to download, or if the speed is slow, you can copy it to a browser or downloader to download.

- C. After downloading three files, unzip them, and the unzipped folder is shown in the following figure:



- D. Then copy the extracted folder to the "**hardware\espressifesp32\tools**" directory under the Arduino IED installation directory, as shown in the following figure:

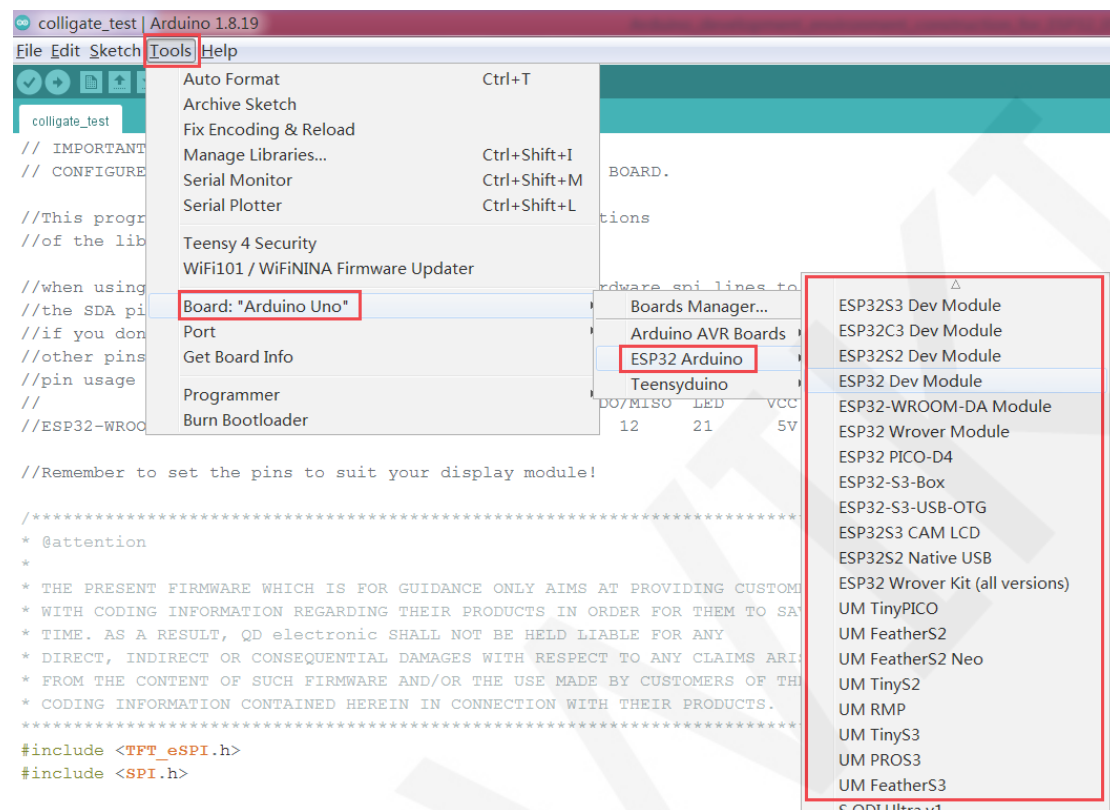


## 4. Compile and download the program

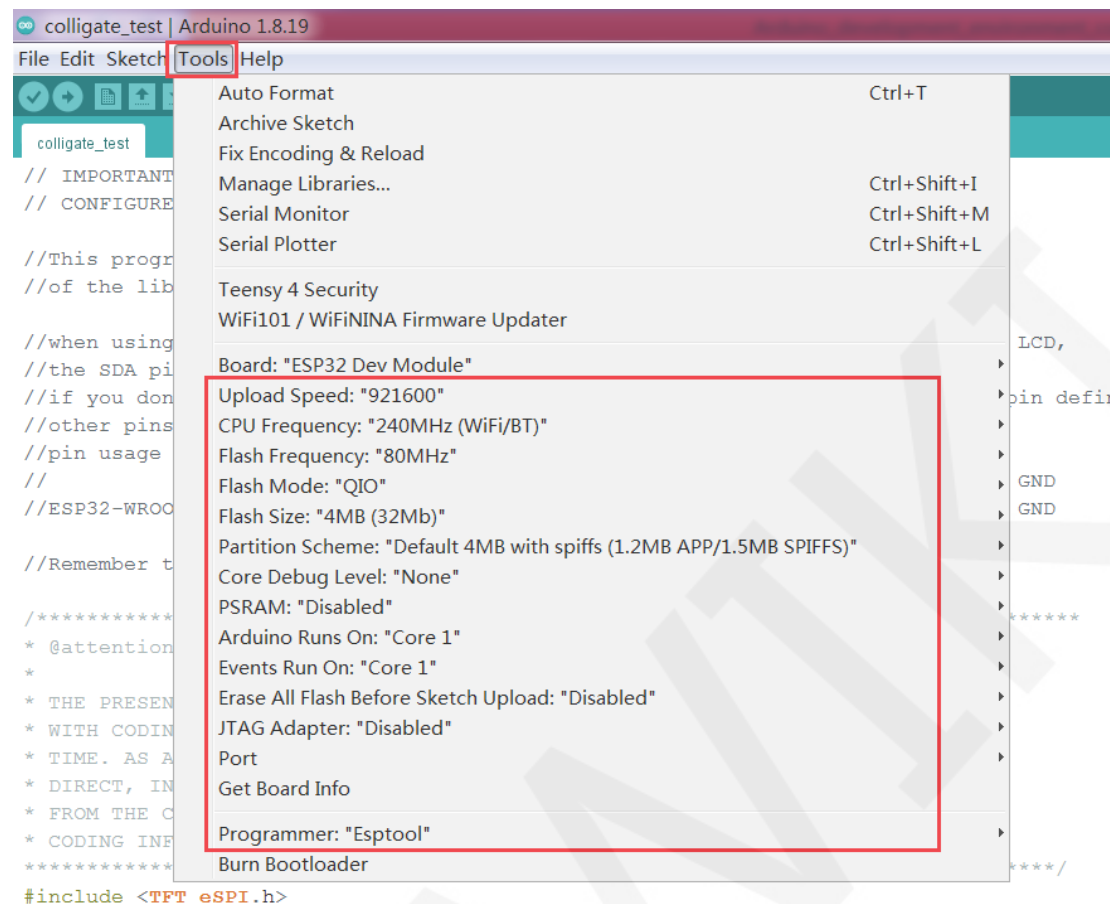
After the development environment is set up, it is necessary to compile and download the program to verify whether the environment is set up correctly.

- A. Connect the ESP32 development board to the PC and power up the development board.

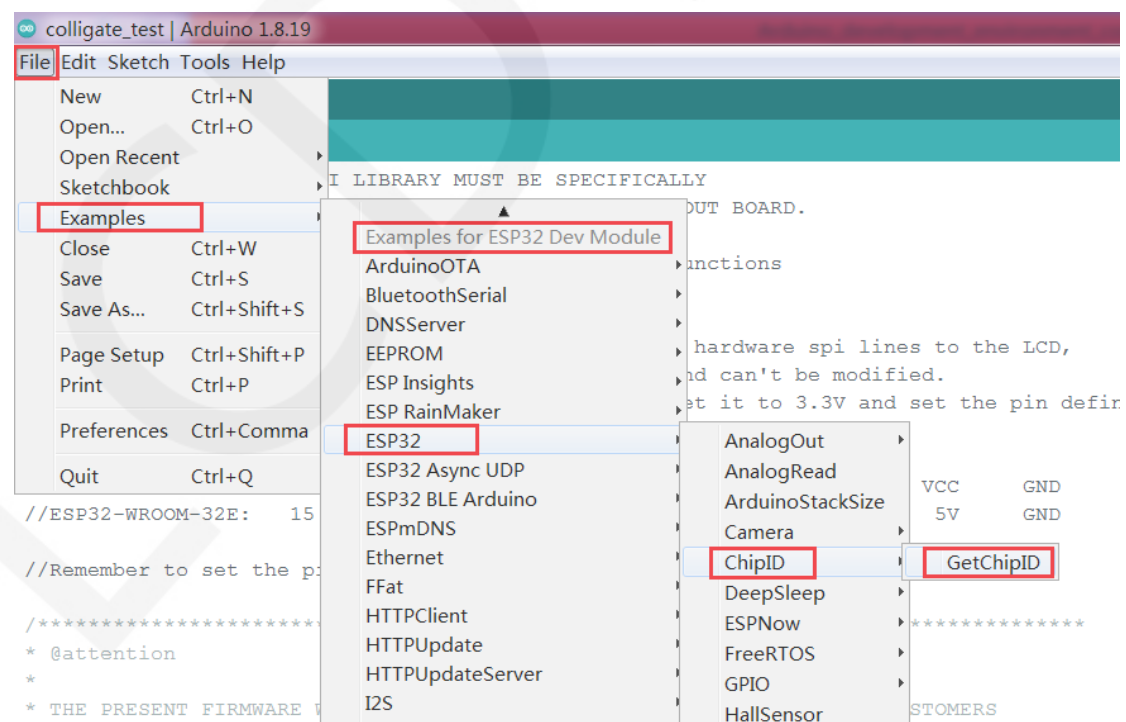
- B. Open the Arduino IDE software, click the **Tools** button, and then select **Board**  
 ->**ESP32 Arduino** ->**ESP32 Dev Module**, as shown in the following figure:



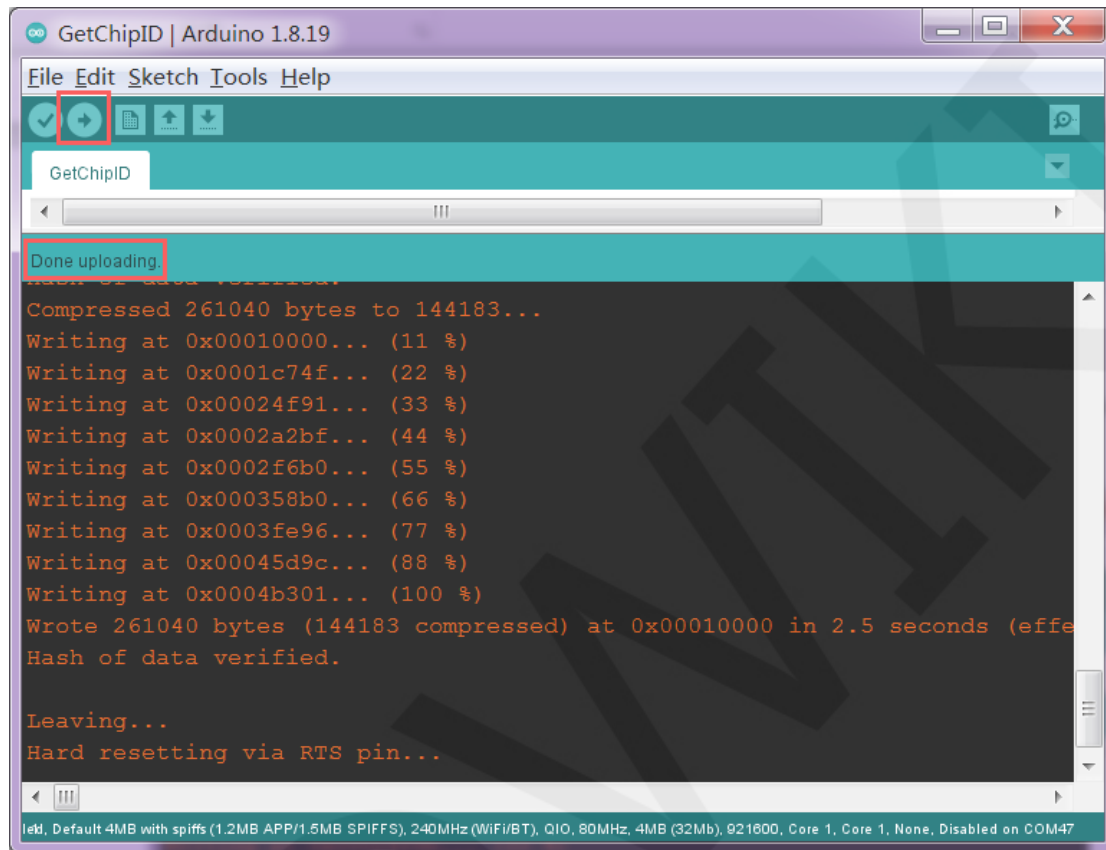
- C. Click the **Tools** button again, and you can see that the **Board** has become **ESP32**. Next, according to the configuration of the ESP32 module you are using, set the **Upload Speed**, **Flash**, **PSRAM**, **Port**, and **other option parameters**, as shown in the following figure:



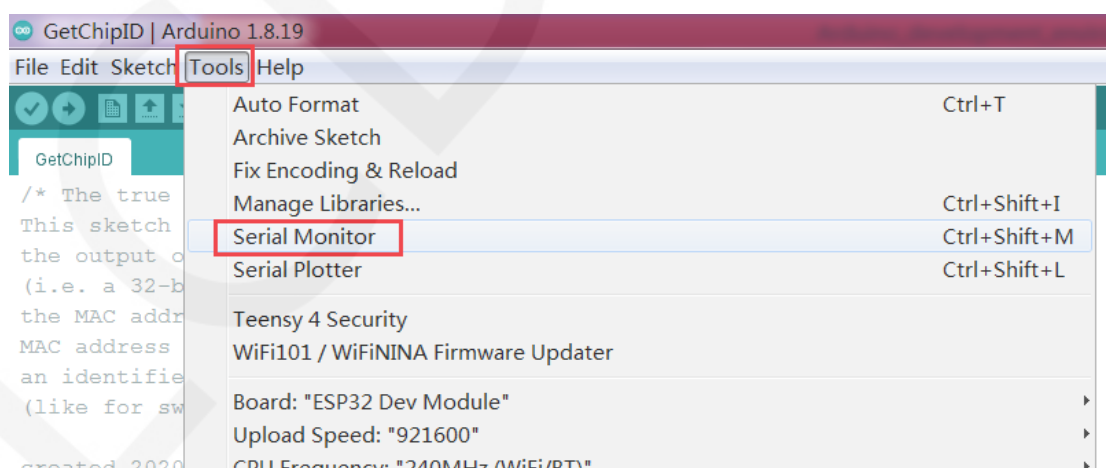
- D. Click the **File** button, then select the **Examples** option, and then select an ESP32 example, as shown in the following figure (select **GetChipID** example here):



- E. Click the **Upload** button, and the program will be compiled and burned. When the **"Done uploading"** prompt appears, it indicates that the program has been compiled and burned successfully, as shown in the following figure:



- F. Click on **Tools ->Serial Monitor**, and the serial port information interface will appear, as shown in the following figure:



The serial information interface displays relevant information such as Chip ID, indicating successful program operation and successful environment setup, as shown in the following figure:

