# Welcome to the JCZN Workshop!

## ·····Table of contents·····

# Getting Started

## Introduction

The objective of this post is to explain how to upload an Arduino program to the JC1060P470 module, from JCZN .

http://www.jczn1688.com/zlxz

The ESP32-P4 is powered by a dual-core RISC-V CPU featuring AI instruction extensions, an advanced memory subsystem, and integrated high-speed peripherals.
Powered by a dual-core RISC-V CPU running at speeds up to 400 MHz, the ESP32-P4 also boasts support for single-precision FPU and AI extensions, providing essential
ESP32-P4 itself does not have WiFi and Bluetooth functions. Use ESP-Hosted to connect to the ESP32-C6 wireless SOC through the SDIO/SPI/UART interface.

## Installing using Arduino IDE

Programming the ESP32

An easy way to get started is by using the familiar Arduino IDE. While this is not necessarily the best environment for working with the ESP32, it has the advantage of being a familiar application, so the learning curve is flattened.

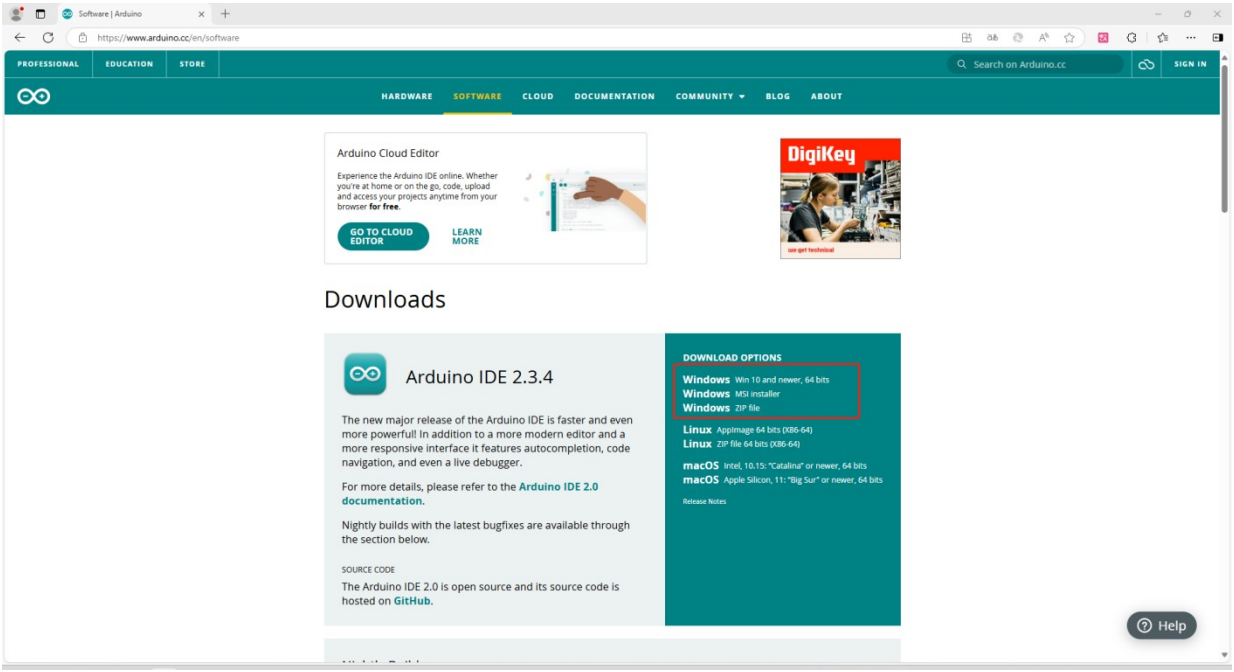
We will be using the Arduino IDE for our experiments.

1, Installing using Arduino IDE

we first need to install version 2.3.4 of the Arduino IDE (or greater),for example, the Arduino installation was in “C/Programs(x86)/Arduino”.

download release link:

https://www.arduino.cc/en/software

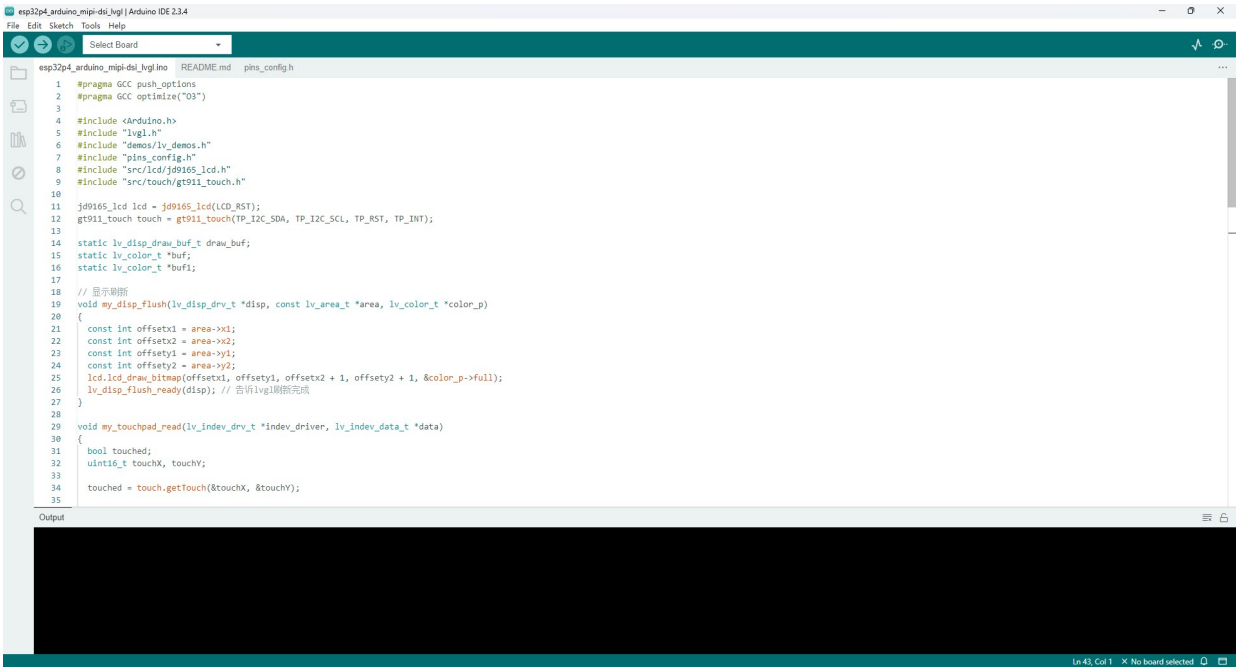



2, This is the way to install Arduino-ESP32 directly from the Arduino IDE.

Add Boards Manager Entry

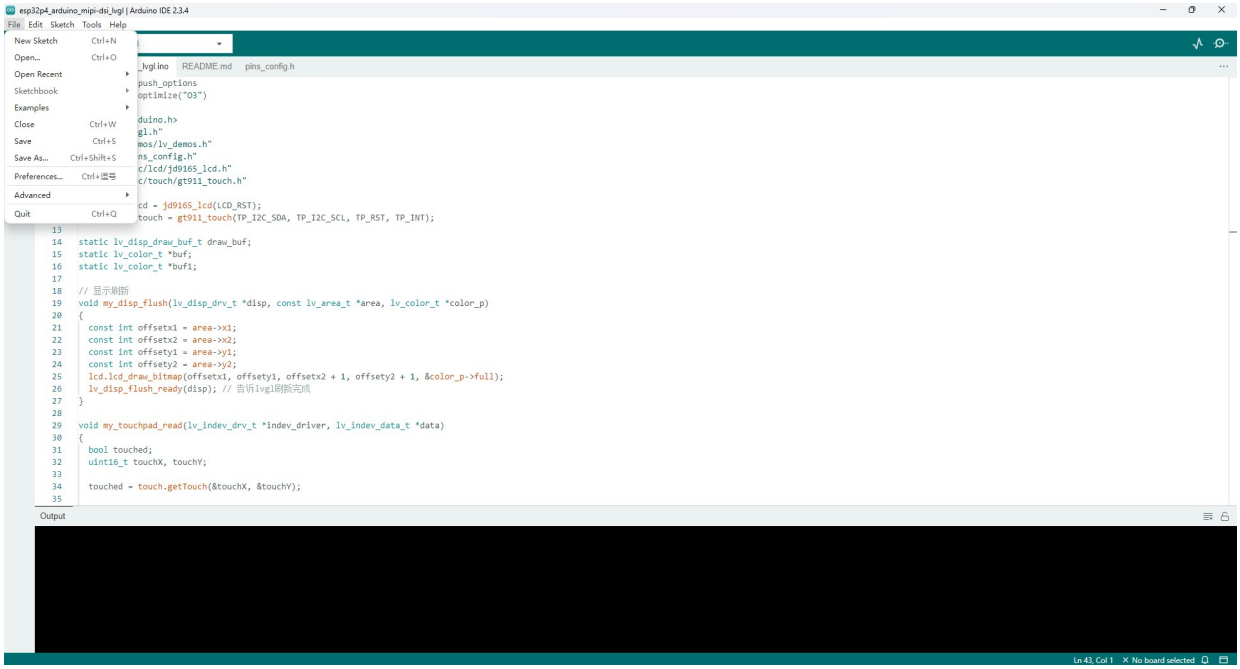
Here is what you need to do to install the ESP32 boards into the Arduino IDE:

(1) Open the Arduino IDE.




(2) Click on the File menu on the top menu bar.

(3) Click on the Preferences menu item. This will open a Preferences dialog box.



(4) You should be on the Settings tab in the Preferences dialog box by default.

(5) Look for the textbox labeled “Additional Boards Manager URLs”.

(6) If there is already text in this box add a coma at the end of it, then follow the next step.

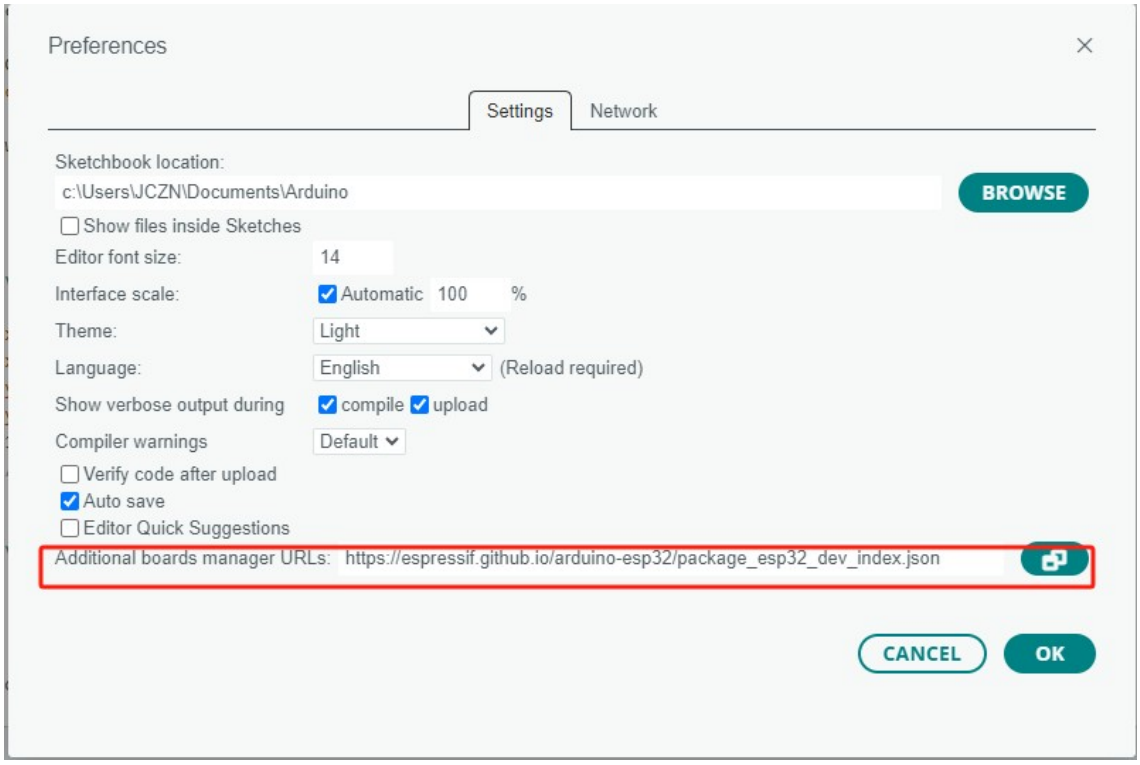
(7) Paste the following link into the text box ：

Stable release link:

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

(8) Click the OK button to save the setting.

The textbox with the JSON link in it is illustrated here:



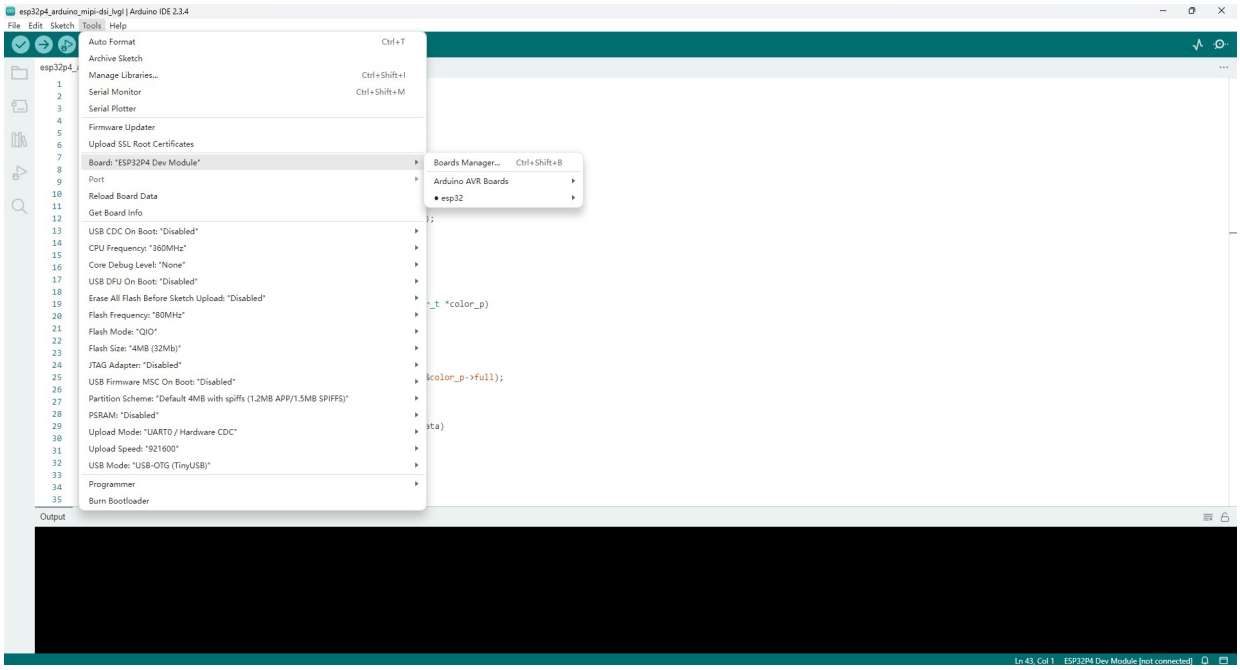
(9) In the Arduino IDE click on the Tools menu on the top menu bar.

(10) Scroll down to the Board: entry

(11) A submenu will open when you highlight the Board: entry.

(12) At the top of the submenu is Boards Manager. Click on it to open the Boards Manager dialog box.

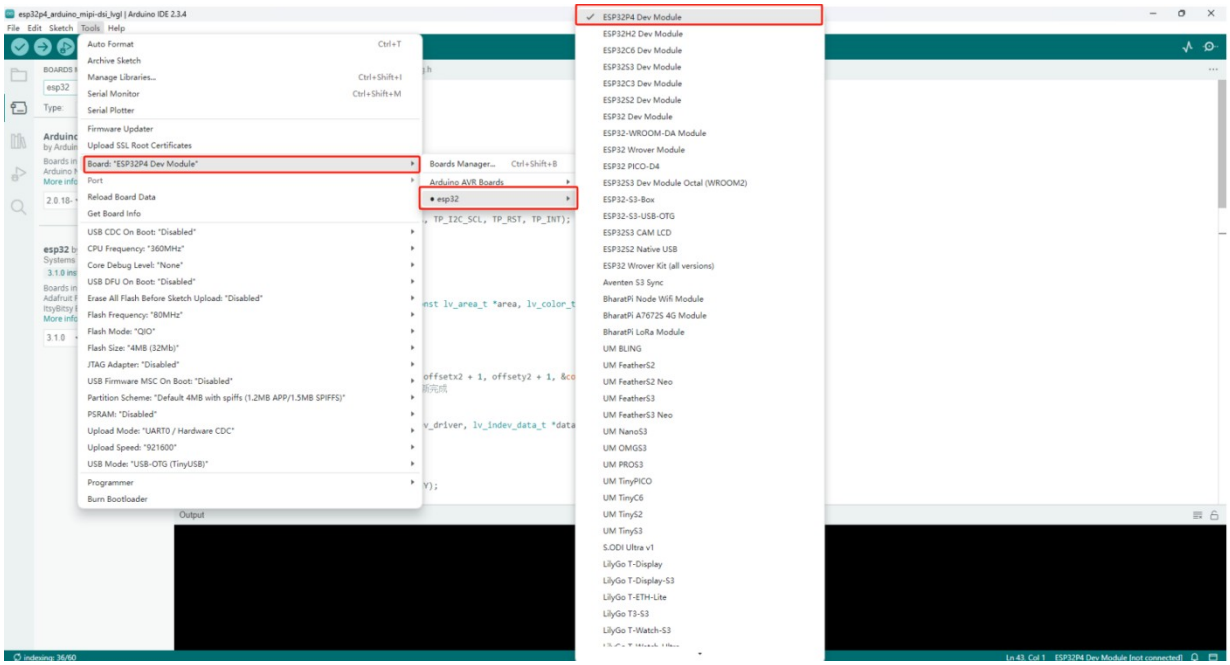
(13)I n the search box in the Boards Manager enter “esp32”.

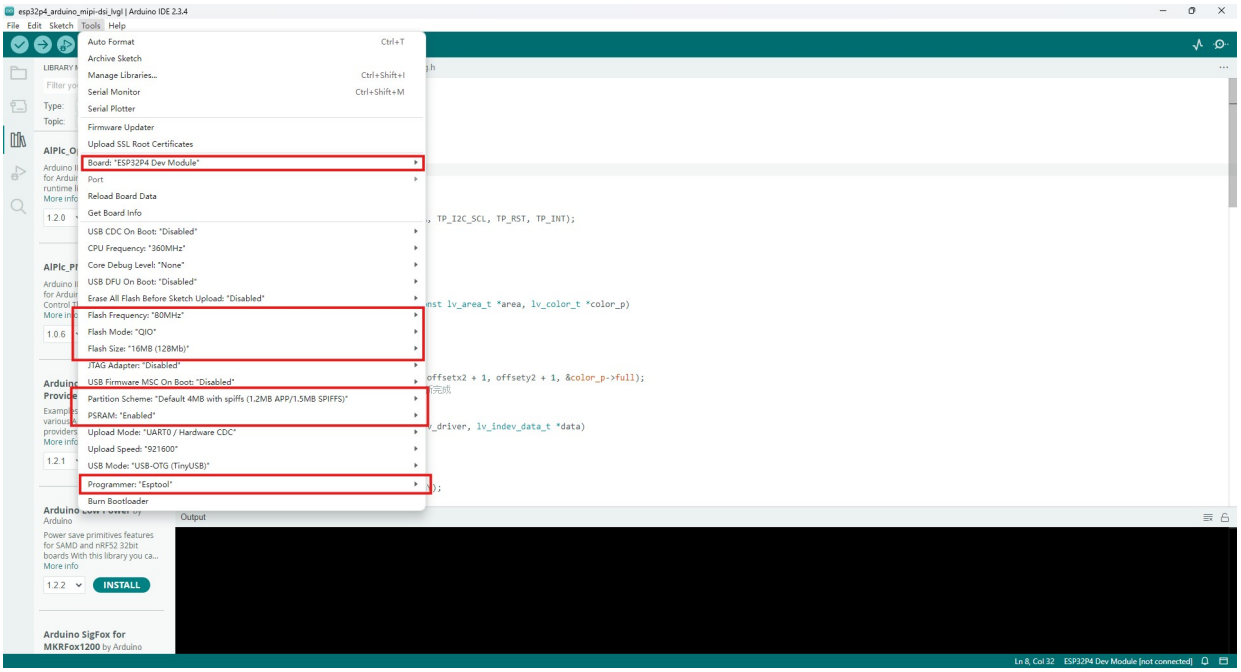



(14) You should see an entry for “esp32 by Espressif Systems”. Highlight this entry and click on the Install button.

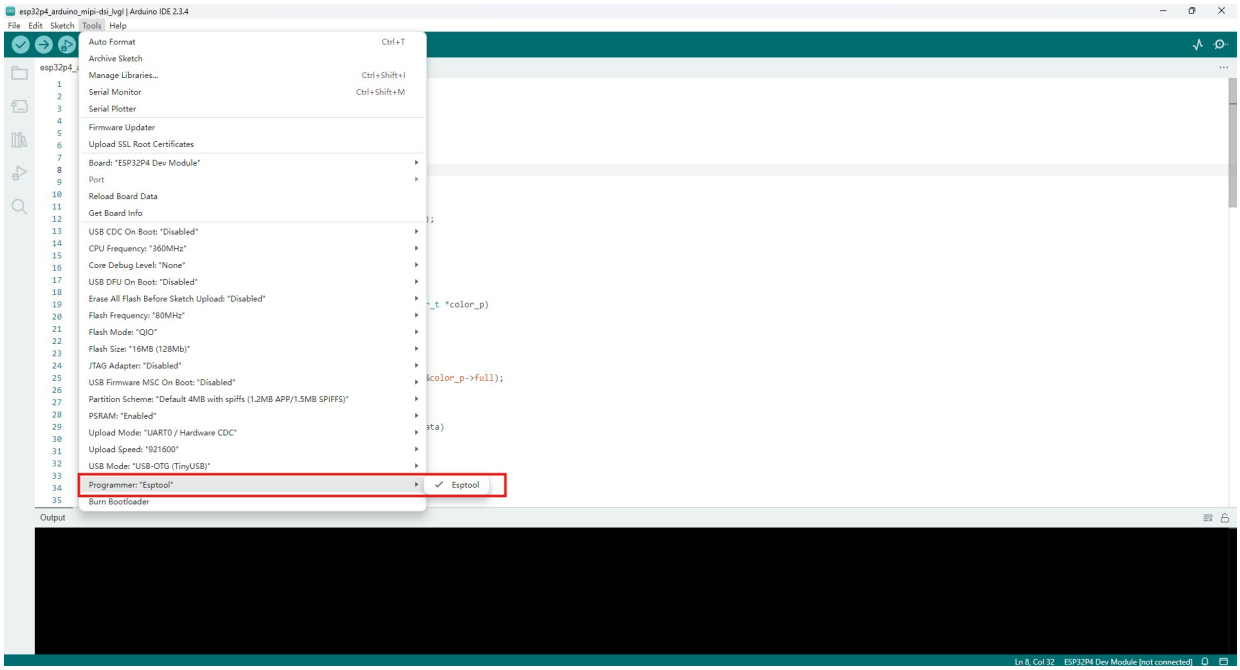
This will install the ESP32 boards into your Arduino IDE

Once the installation completes, we need to select the correct board options for the "ESP32 Arduino" board. In the board type, in the tools tab, we choose “ESP32P4 Dev Module”.




Set and In the programmer entry of the same tab, we choose “esptool”.




It's important to note that after the code is uploaded, the device will start to run it. So, if we want to upload a new program, wee need to reset the power of the device, in order to guarantee that it enters flashing mode again.

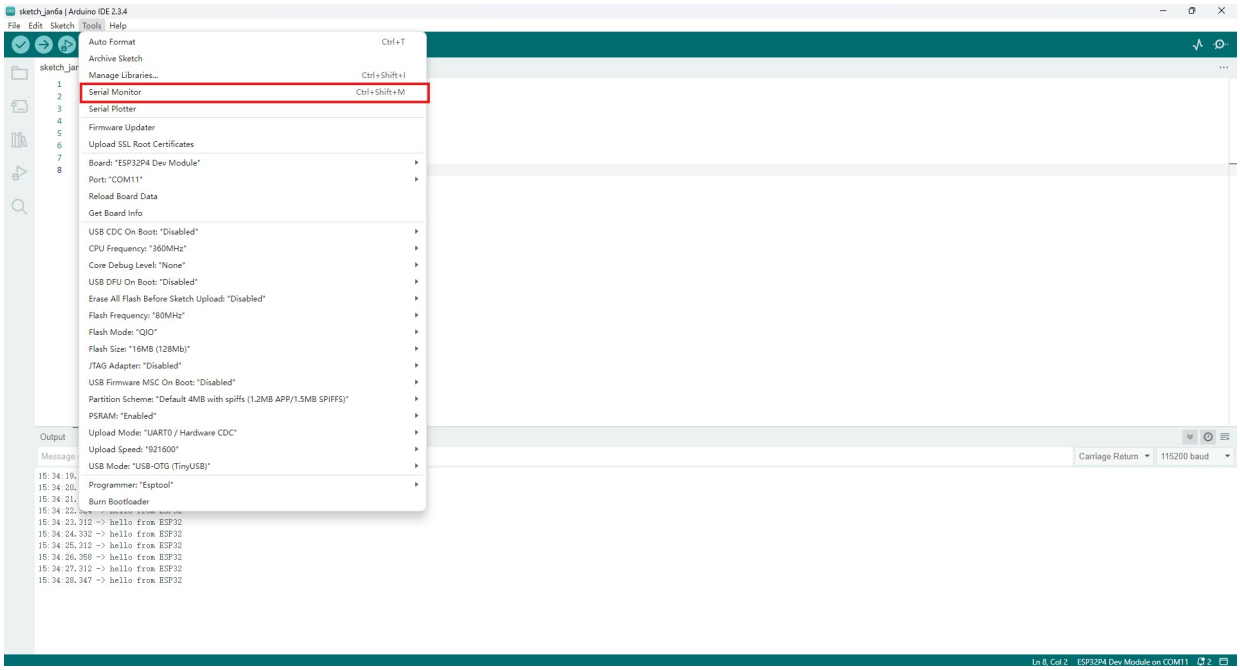
First program

Since this platform is based on Arduino, we can use many of the usual functions. As an example for the first program, the code bellow starts the Serial port and prints "hello from ESP32" every second.

```
void setup() {
  Serial.begin(115200);
}

void loop() {
  Serial.println("hello from ESP32");
  delay(1000);
}
```

If everything is working fine, we will see the output in the serial console shown.

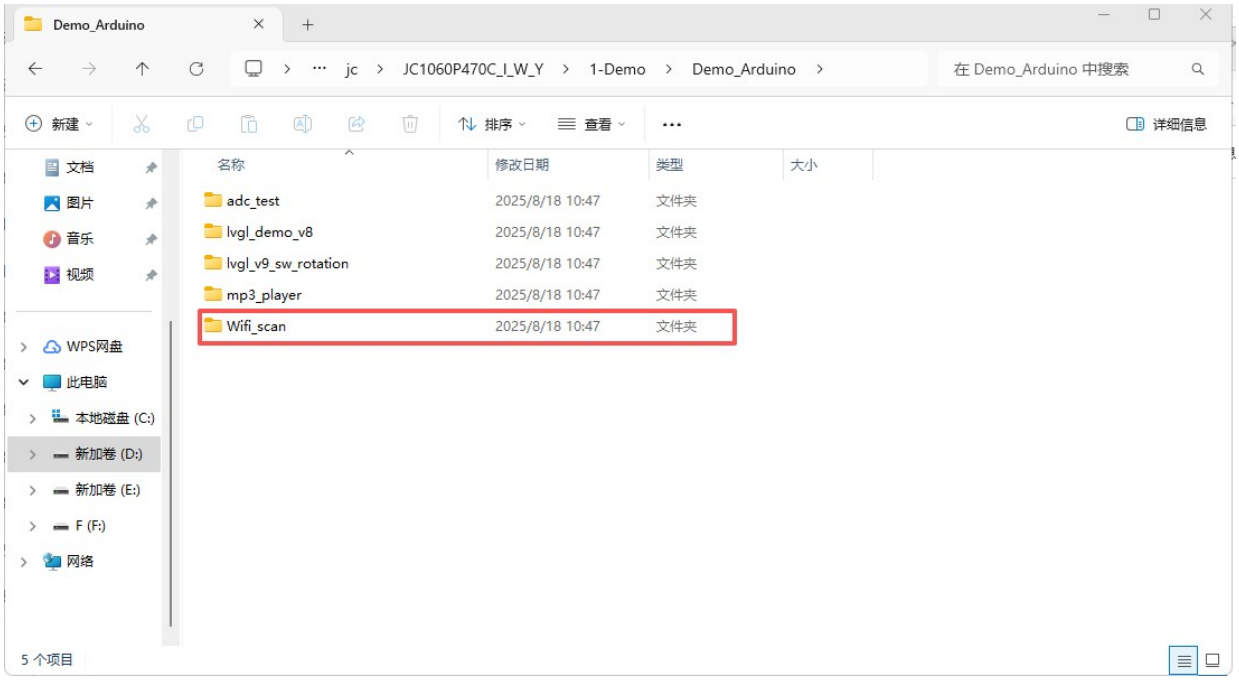



Again thank you for so much concern.. Hopefully, it's the beginning of a wonderful relationship!

**About the wifi_scan Demo:**

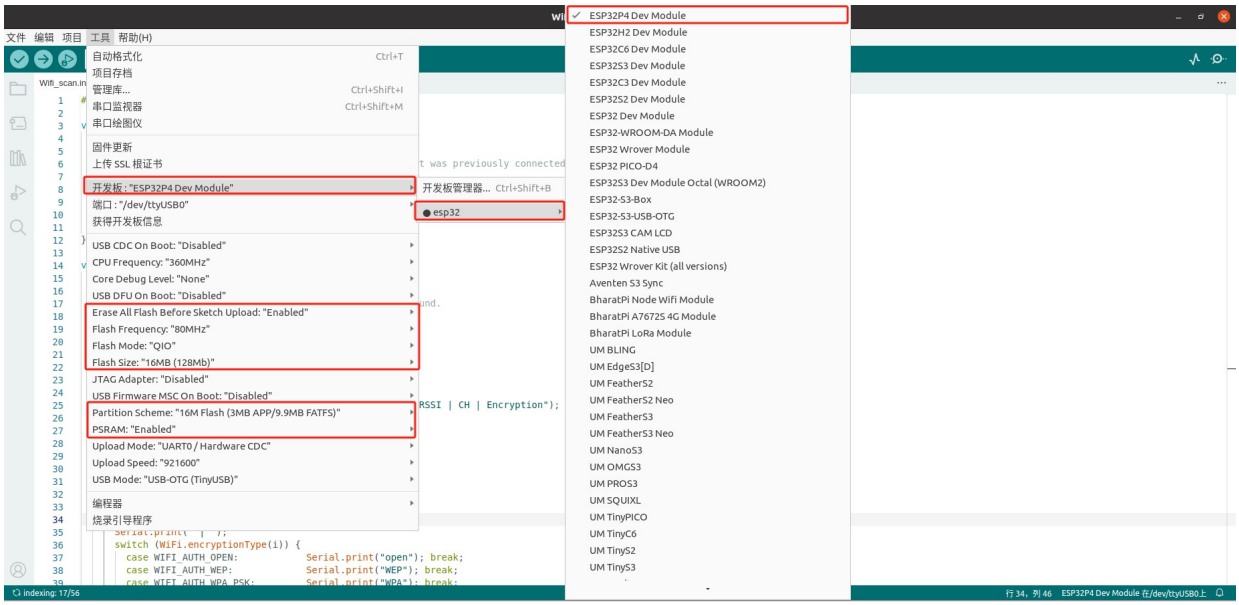
Find the data center wifi_scan:

As shown:

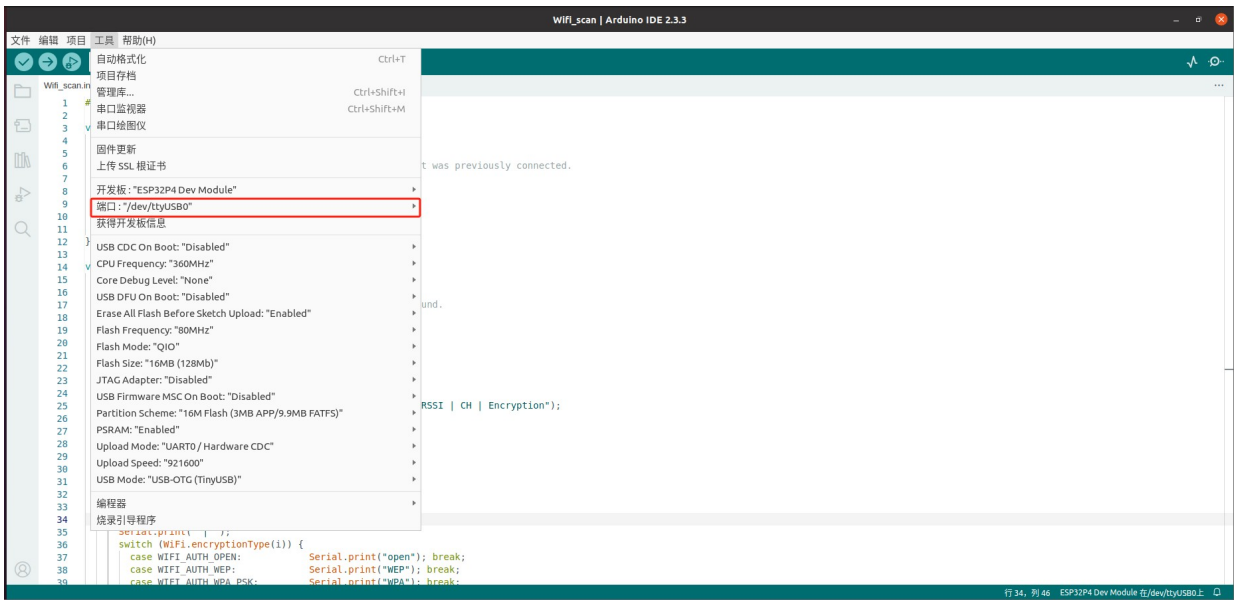



Step 2: select the development board configuration

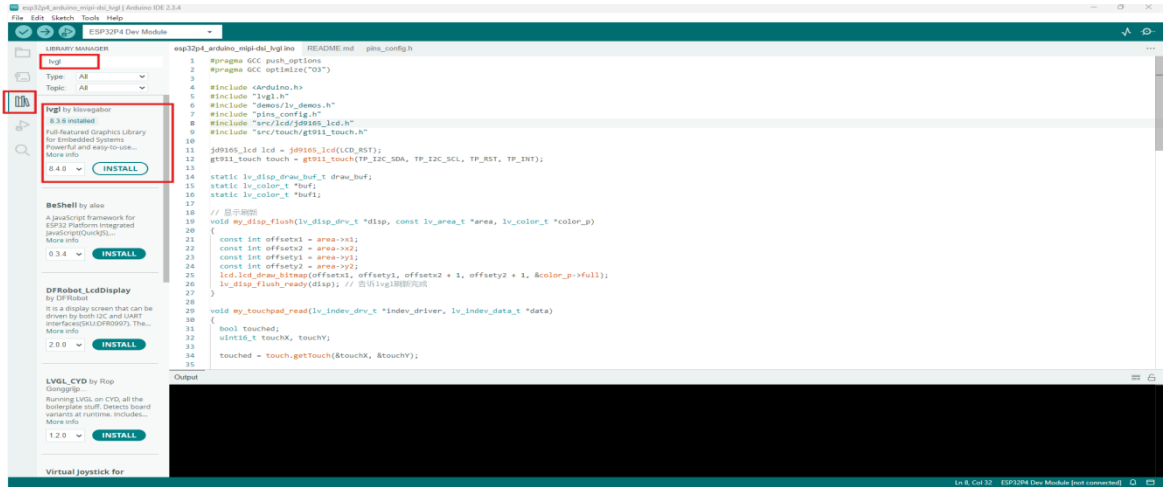
As show



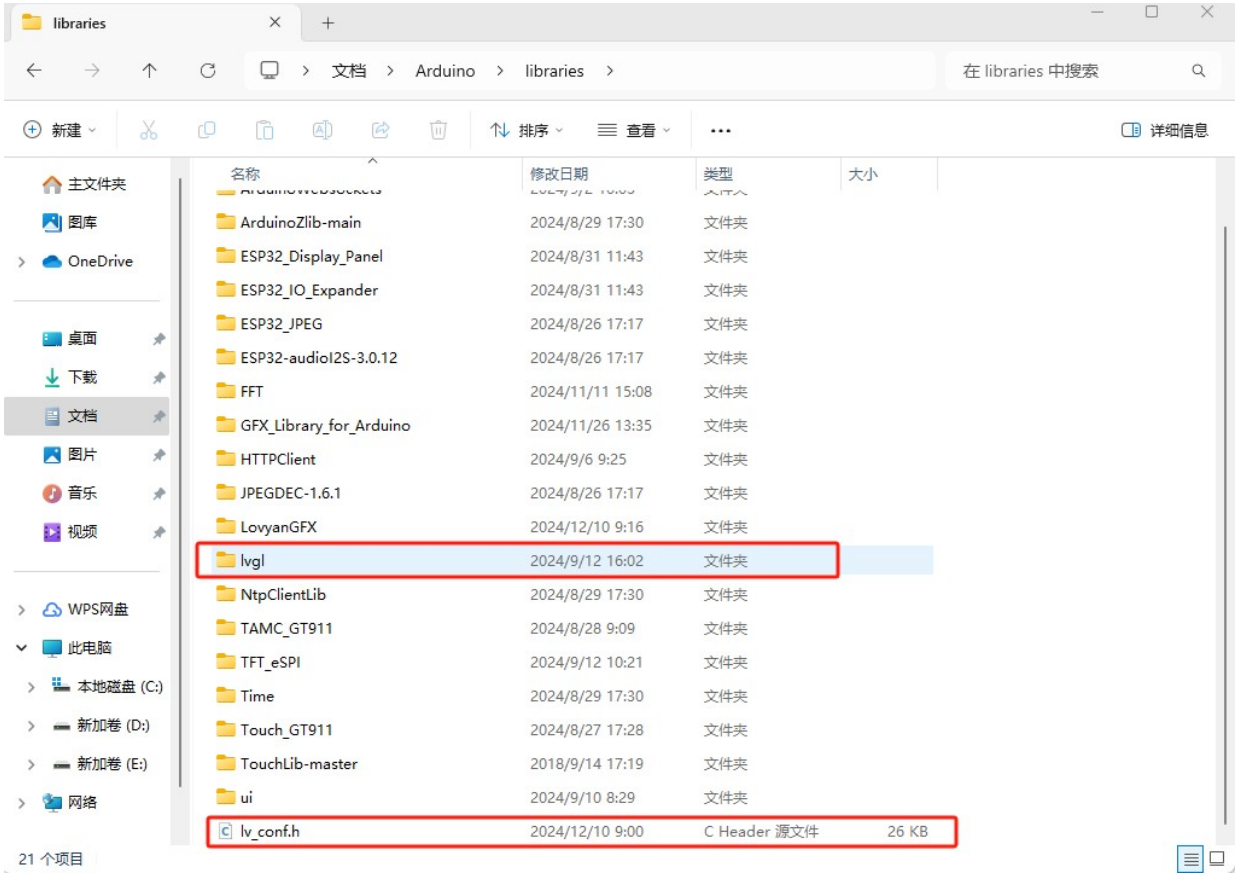

Step 3: Select the corresponding port



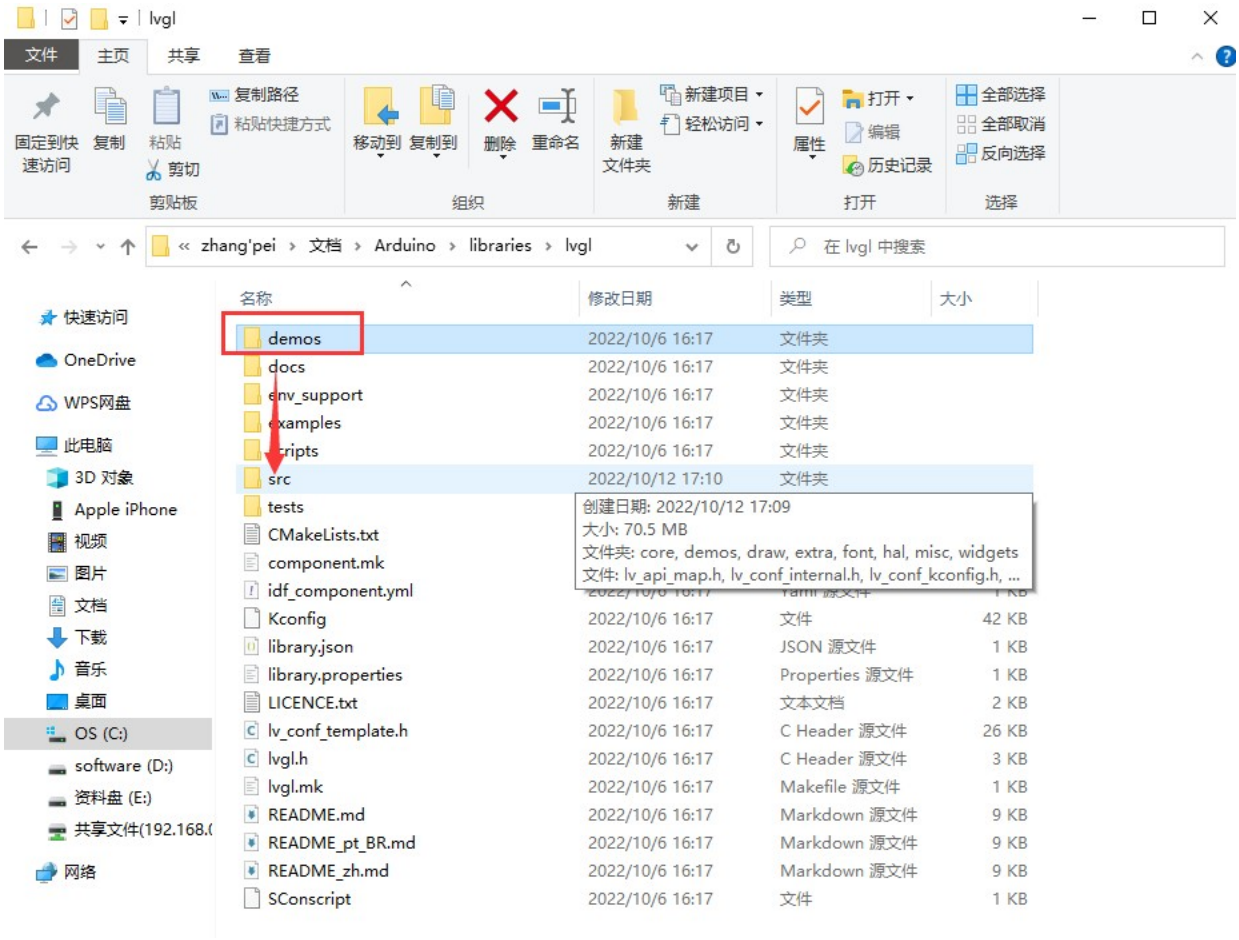

Step 4:Download LVGL library files .



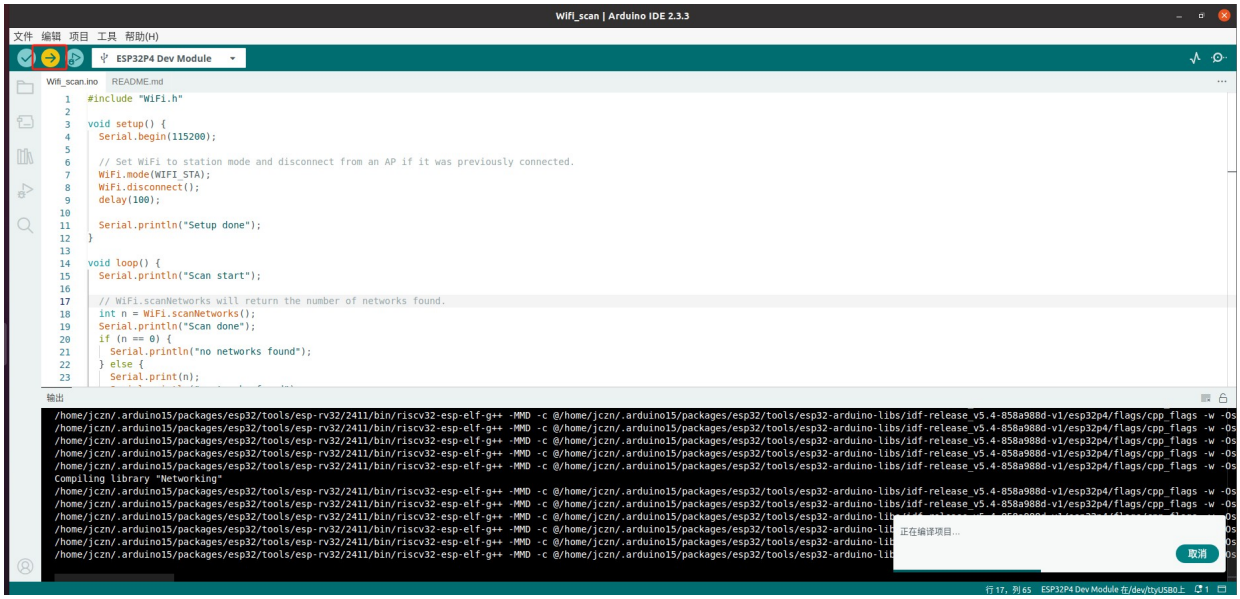
Step 5: Put this file under the arduino library file, it must be in the same root directory as the library lvgl .



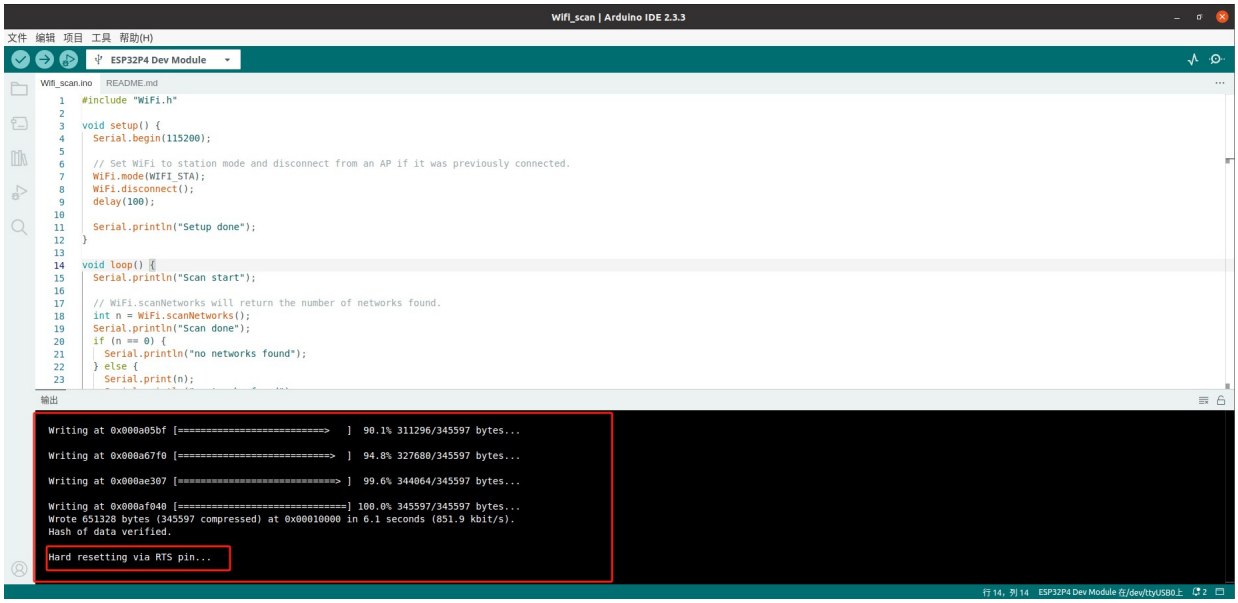

Step 6: Three-Lvgl demos The file is copied to the SRC folder




Step 7: begin the programming.




Wait for the burning to complete, as shown in the following figure




Step 8:Check the serial port

If the serial port has output wifi information, it means that the program is running perfectly