Keyestudio ESP32 Core Board

(Black and Eco-friendly)

Guide Content

Description:

This keyestudio ESP32 core board is a Mini development board based on the ESP-WROOM-32 module.

The board has brought out most I/O ports to pin headers of 2.54mm pitch. These provide an easy way of connecting peripherals according to your own needs.

When it comes to developing and debugging with the development board, the both side standard pin headers can make your operation more simple and handy.

The ESP-WROOM-32 module is the industry's leading integrated WiFi + Bluetooth solution with less than 10 external components.

It integrates antenna switch, RF balun, power amplifiers, low noise amplifiers, filters and power management modules.

At the same time, it also integrates with TSMC's low-power 40nm technology, so that power performance and RF performance are safe and reliable, easy to expand to a variety of applications.
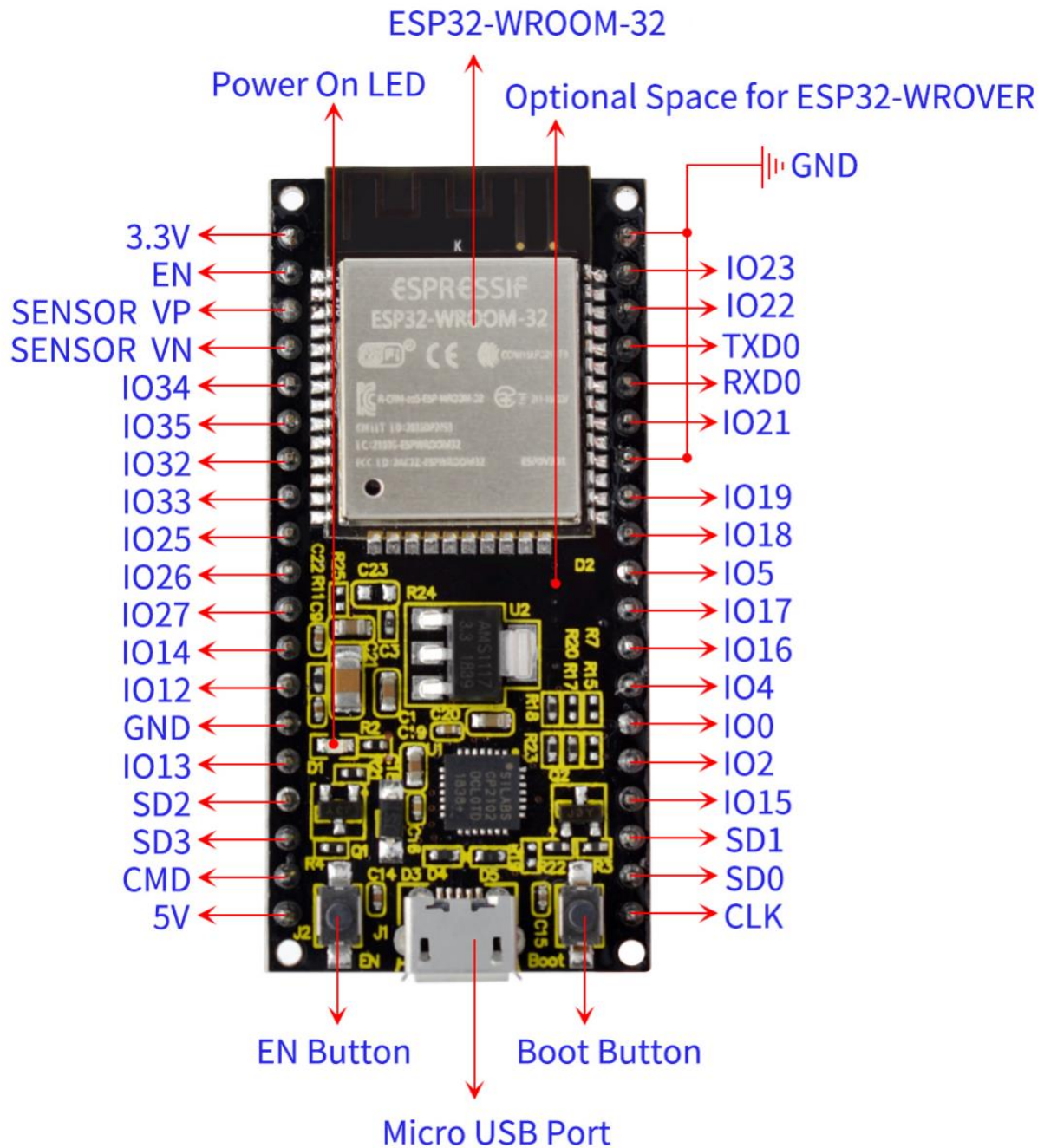
Technical Details:

- Microcontroller: ESP-WROOM-32 module

- USB to Serial Port Chip: CP2102-GMR

- Operating Voltage: DC 5V

- Operating Current: 80mA (average)

- Current Supply: 500mA (Minimum)

- Operating Temperature Range: -40℃ ~ +85℃

- WiFi mode: Station/SoftAP/SoftAP+Station/P2P

- WiFi protocol: 802.11 b/g/n/e/i (802.11n, speed up to 150 Mbps

- WiFi frequency range: 2.4 GHz ~ 2.5 GHz

- Bluetooth protocol: conform to Bluetooth v4.2 BR/EDR and BLE standards

- Dimensions: 55mm*26mm*13mm

- Weight: 9.3g
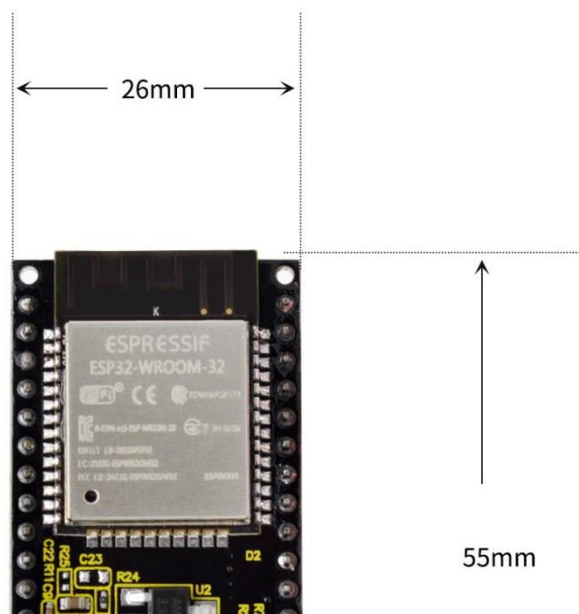
Element and Interfaces:

Here is an explanation of what every element and interface of the board has:

Power On LED   ESP32-WROOM-32   Optional Space for ESP32-WROVER

GND

3.3V	IO23
EN	IO22
SENSOR VP	TXD0
SENSOR VN	RXD0
IO34	IO21
IO35	
IO32	IO19
IO33	IO18
IO25	IO5
IO26	IO17
IO27	IO16
IO14	IO4
IO12	IO0
GND	IO2
IO13	IO15
SD2	SD1
SD3	SD0
CMD	CLK
5V	

EN Button   Boot Button

Micro USB Port

Specialized Functions of Some Pins:

PINS	EXPLANATIONS
IO23	VSPI MOSI/SPI MOSI
IO22	Wire SCL
TXD0	IO1/Serial TX
RXD0	IO3/Serial RX
IO21	Wire SDA
IO19	VSPI MISO/SPI MISO
IO18	VSPI SCK/SPI SCK
IO5	VSPI SS/SPI SS
IO4	ADC10/TOUCH0
IO0	ADC11/TOUCH1
IO2	ADC12/TOUCH2
IO15	HSPI SS/ADC13/TOUCH3/TDO
SD1	IO8/FLASH D1

SD0	IO7/FLASH D0
CLK	IO6/FLASH SCK
CMD	IO11/FLASH CMD
SD3	IO10/FLASH D3
SD2	IO9/FLASH D2
IO13	HSPI MOSI/ADC14/TOUCH4/TCK
IO12	HSPI MISO/ADC15/TOUCH5/TDI
IO14	HSPI SCK/ADC16/TOUCH6/TMS
IO27	ADC17/TOUCH7
IO26	ADC19/DAC2
IO25	ADC18/DAC1
IO33	ADC5/TOUCH8
IO32	ADC4/TOUCH9
IO35	ADC7
IO34	ADC6
SENSOR VN	IO39/ADC3
SENSOR VP	IO36/ADC0

EN	RESET

Detailed Using Method as follows:

Step1| Install the Arduino IDE

When programming the control board, first you should install the Arduino software and driver.

You can download the different versions for different systems from the link below:

https://www.arduino.cc/en/Main/OldSoftwareReleases#1.5.x

This control board is compatible with the Arduino 1.8.7 or latest version.

So next we will download the Arduino 1.8.7 software to test the keyestudio ESP32 core board.

In this Windows system page, there are two options. One is Windows version, the other is Windows Installer.

For Windows Installer, you can download the installation file, this way you need to install the arduino IDE.

For simple Windows version, you can download the software directly, do not need to install, just directly use the software after unzip the package.

Next, we click the **Windows**, pop up the interface as below.

Click **JUST DOWNLOAD**.

Downloaded well the **arduino-1.8.7-windows.zip** package to your computer, you can direct to unzip the package. Open the Arduino-1.8.7 folder, you should get it as follows.

arduino-1.8.7-windows

Name	Date modified	Type	Size
drivers	9/11/2018 5:33 PM	File folder	
examples	9/11/2018 5:35 PM	File folder	
hardware	9/11/2018 5:35 PM	File folder	
java	9/11/2018 5:35 PM	File folder	
lib	9/11/2018 5:35 PM	File folder	
libraries	9/11/2018 5:35 PM	File folder	
reference	9/11/2018 5:35 PM	File folder	
tools	9/11/2018 5:35 PM	File folder	
tools-builder	9/11/2018 5:34 PM	File folder	
arduino.exe	9/11/2018 5:35 PM	Application	395 KB
arduino.l4j	9/11/2018 5:35 PM	Configuration setti…	1 KB
arduino_debug.exe	9/11/2018 5:35 PM	Application	393 KB
arduino_debug.l4j	9/11/2018 5:35 PM	Configuration setti…	1 KB
arduino-builder.exe	9/11/2018 5:34 PM	Application	11,745 KB
libusb0.dll	9/11/2018 5:33 PM	Application extens…	43 KB
msvcp100.dll	9/11/2018 5:33 PM	Application extens…	412 KB
msvcr100.dll	9/11/2018 5:33 PM	Application extens…	753 KB
revisions	9/11/2018 5:33 PM	Text Document	87 KB

Click the icon of ARDUINO software to open. This is your Arduino.

sketch_may13a | Arduino 1.8.7

File Edit Sketch Tools Help

sketch_may13a

```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

Arduino/Genuino Uno on
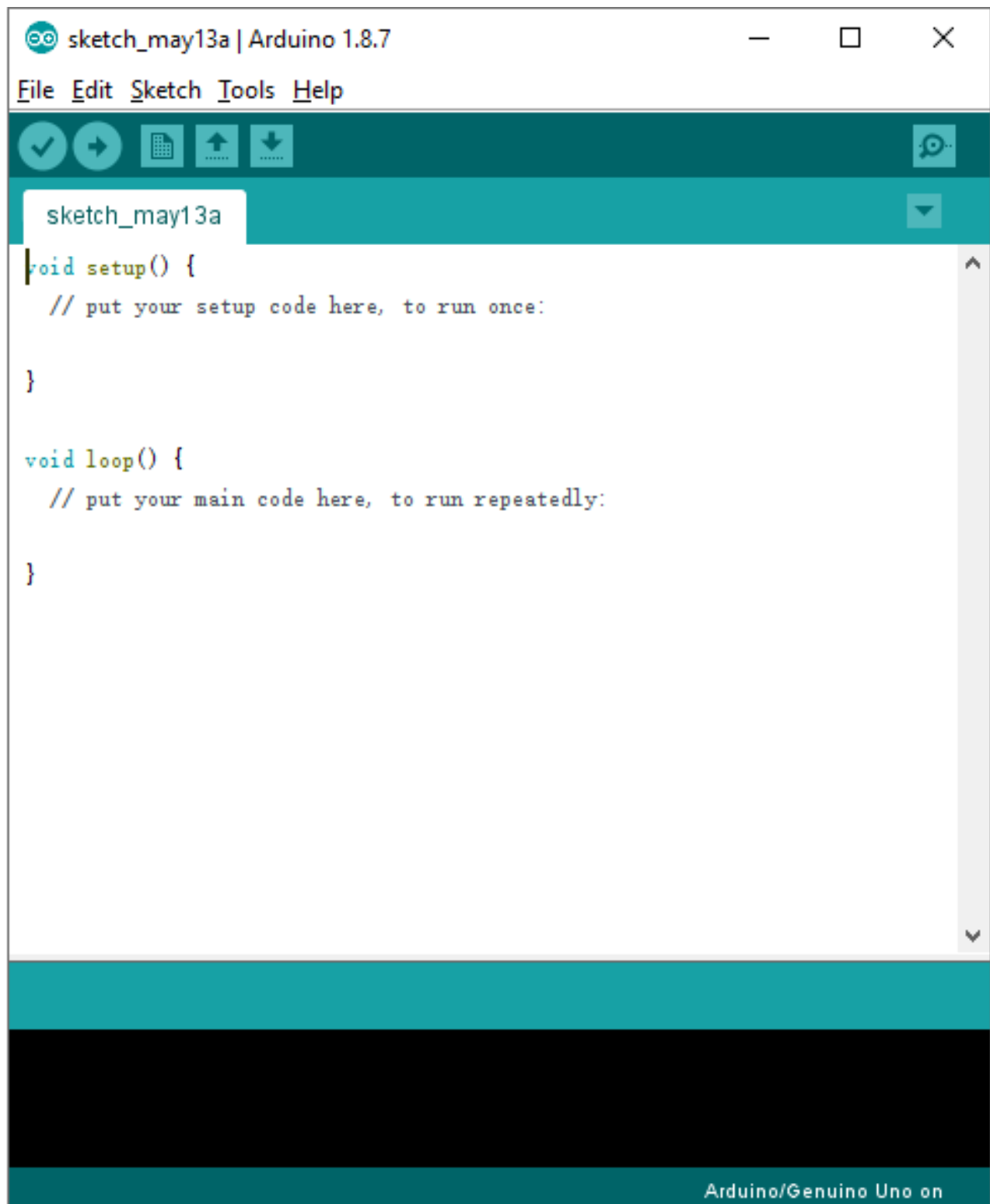
Step2| Installing the Driver

The USB to serial port chip of this control board is CP2102-GMR. So you need to install the driver for the chip.

You can click the driver tool download link:

https://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers

Download Software

The CP210x Manufacturing DLL and Runtime DLL have been updated and must be used with v6.0 and later of the CP210x Windows VCP Driver. Application Note Software downloads affected are AN144SW.zip, AN205SW.zip and AN223SW.zip. If you are using a 5.x driver and need support you can download archived Application Note Software.

Legacy OS software and driver package download links and support information ›

Download for Windows 10 Universal (v10.1.7)

Platform	Software	Release Notes
Windows 10 Universal	Download VCP (2.3 MB)	Download VCP Revision History

Download for Windows 7/8/8.1 (v6.7.6)

Platform	Software	Release Notes
Windows 7/8/8.1	Download VCP (5.3 MB) (Default)	Download VCP Revision History
Windows 7/8/8.1	Download VCP with Serial Enumeration (5.3 MB) Learn More »	Download VCP Revision History

Download for Windows XP/Server 2003/Vista/7/8/8.1 (v6.7)

Platform	Software	Release Notes
Windows XP/Server 2003/Vista/7/8/8.1	Download VCP (3.66 MB)	Download VCP Revision History

It includes different drivers for different computer's systems. Download and install

the driver according to your computer's system.

For example, we download the driver for Windows 7. Get the compression package of CP210x_Windows_Drivers

Download for Windows 7/8/8.1 (v6.7.6)

Platform	Software	Release Notes
Windows 7/8/8.1	Download VCP (5.3 MB) (Default)	Download VCP Revision History
Windows 7/8/8.1	Download VCP with Serial Enumeration (5.3 MB) Learn More »	Download VCP Revision History

CP210x_Windows_Drivers

Then extract the compression package; you should see the application to install.

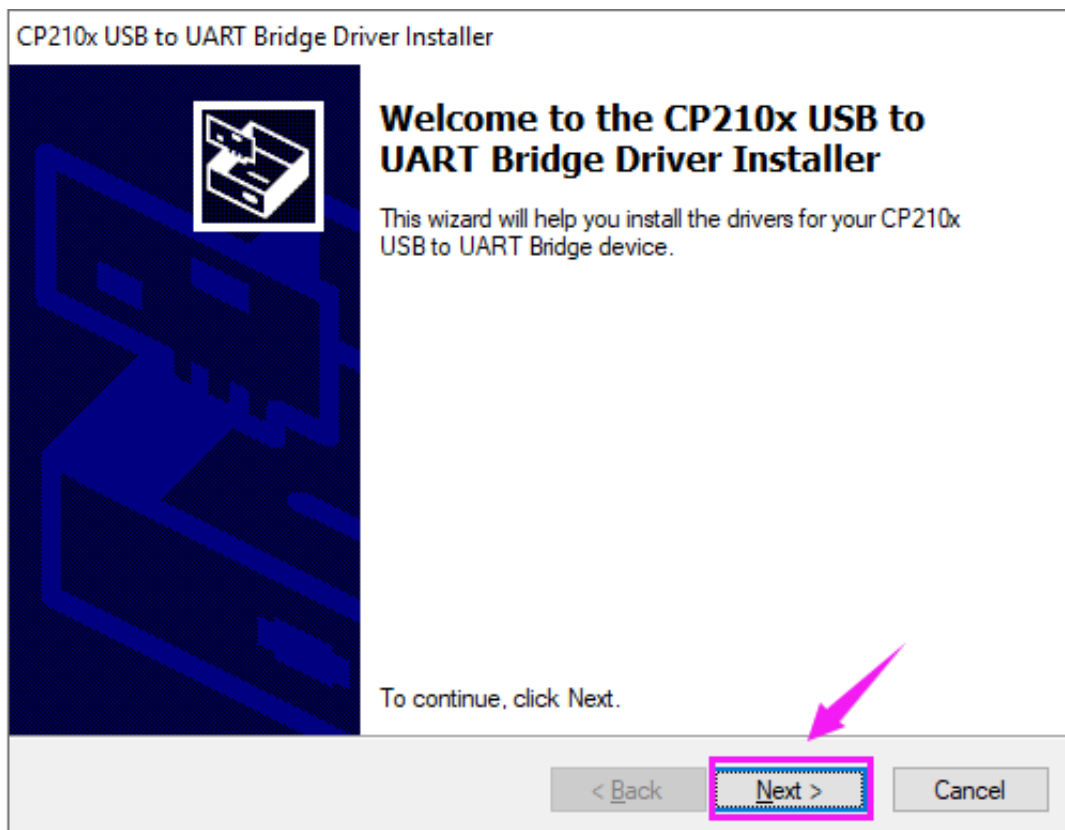| Name	Date modified	Type	Size
x64	5/14/2019 8:18 AM	File folder	
x86	5/14/2019 8:18 AM	File folder	
CP210xVCPInstaller_x64.exe	9/28/2017 1:58 AM	Application	1,026 KB
CP210xVCPInstaller_x86.exe	9/28/2017 1:58 AM	Application	903 KB
dpinst	9/28/2017 1:45 AM	XML Document	12 KB
SLAB_License_Agreement_VCP_Windows	9/28/2017 1:46 AM	Text Document	9 KB
slabvcp	6/2/2018 4:35 AM	Security Catalog	11 KB
slabvcp	6/2/2018 4:35 AM	Setup Information	8 KB
v6-7-6-driver-release-notes	6/16/2018 2:51 AM	Text Document	16 KB

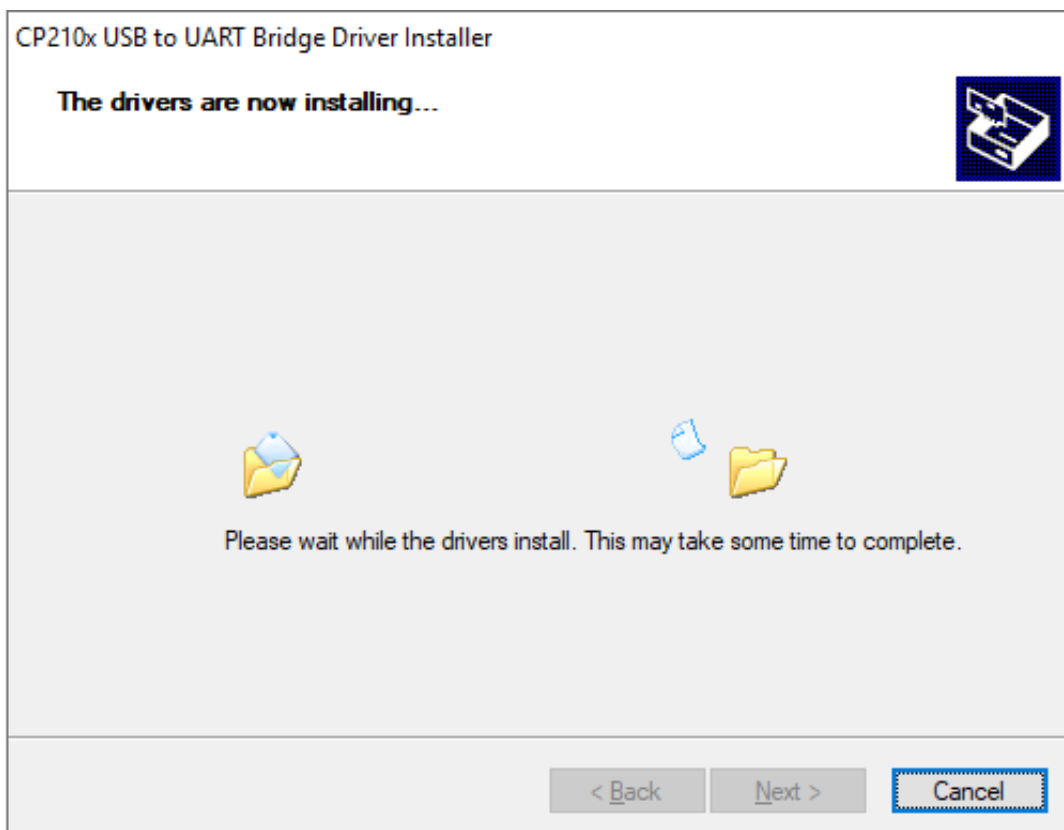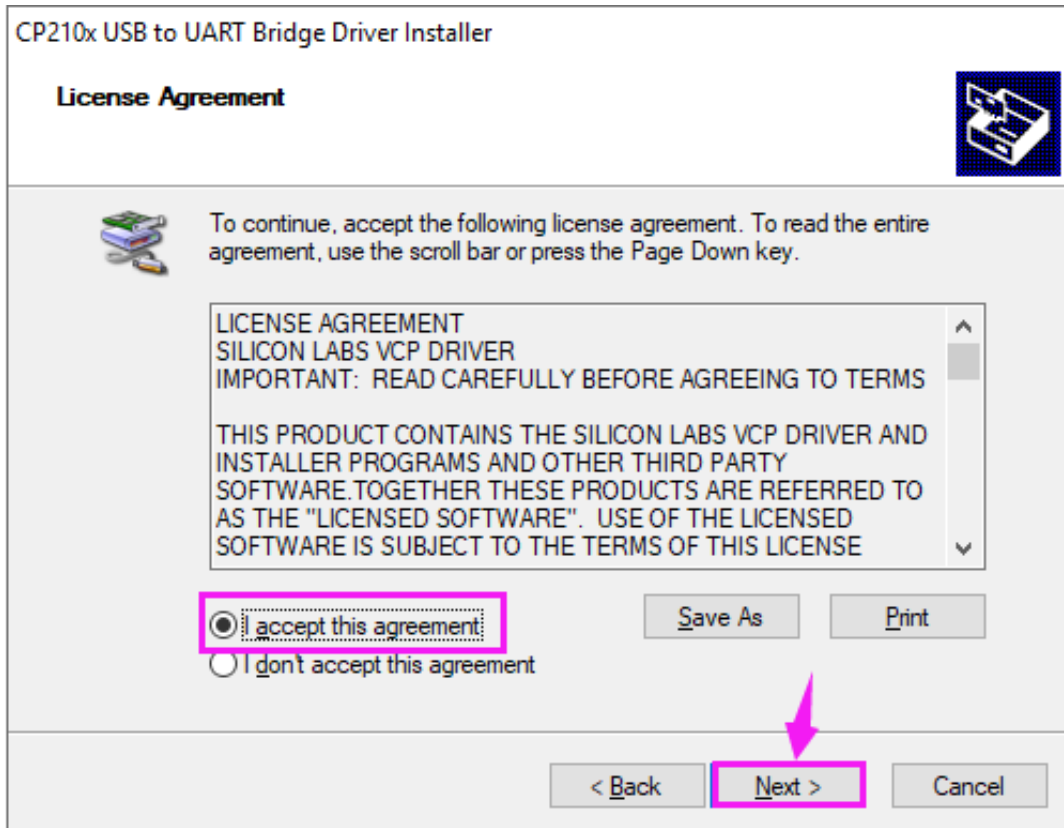The driver software installation is very simple. Just select the driver application as

you like.

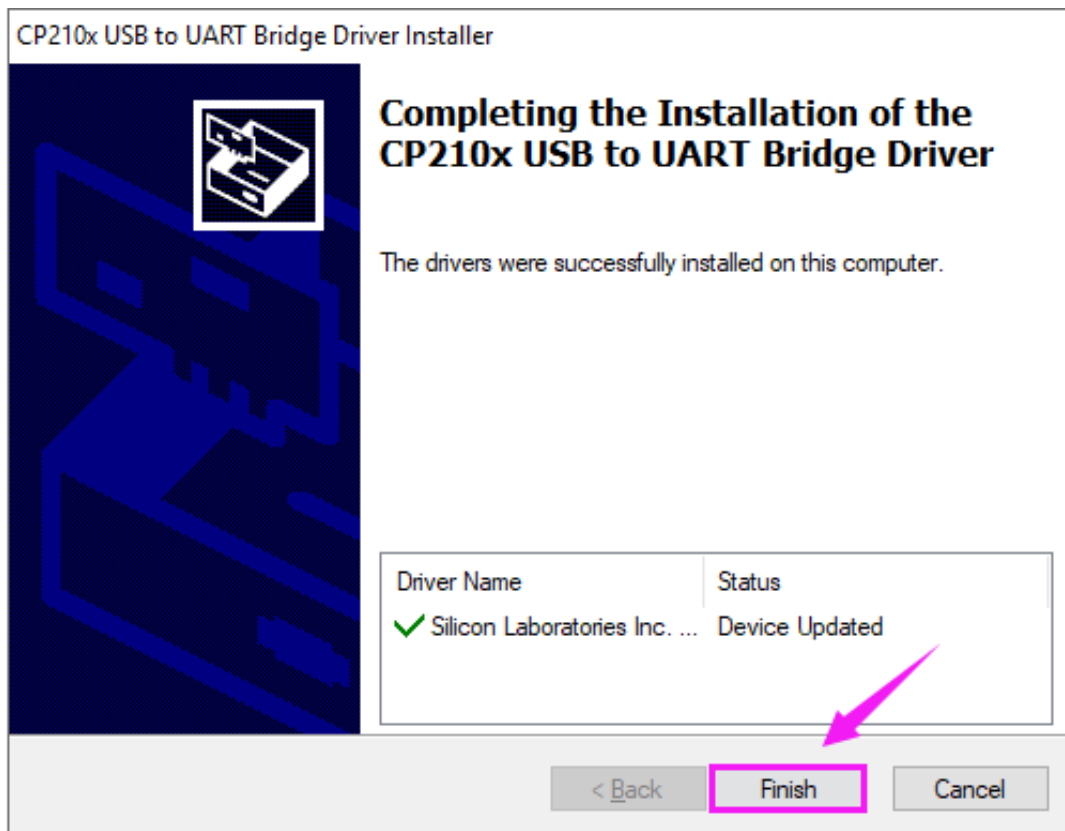Click to **.exe** package to install the driver. Click "Next".

Click to select "I accept this agreement" and click "Next".

CP210x USB to UART Bridge Driver Installer

License Agreement

To continue, accept the following license agreement. To read the entire agreement, use the scroll bar or press the Page Down key.

> LICENSE AGREEMENT
> SILICON LABS VCP DRIVER
> IMPORTANT: READ CAREFULLY BEFORE AGREEING TO TERMS
>
> THIS PRODUCT CONTAINS THE SILICON LABS VCP DRIVER AND
> INSTALLER PROGRAMS AND OTHER THIRD PARTY
> SOFTWARE.TOGETHER THESE PRODUCTS ARE REFERRED TO
> AS THE "LICENSED SOFTWARE". USE OF THE LICENSED
> SOFTWARE IS SUBJECT TO THE TERMS OF THIS LICENSE

⦿ I accept this agreement   Save As   Print
◯ I don't accept this agreement

< Back   Next >   Cancel

CP210x USB to UART Bridge Driver Installer

The drivers are now installing...

Please wait while the drivers install. This may take some time to complete.
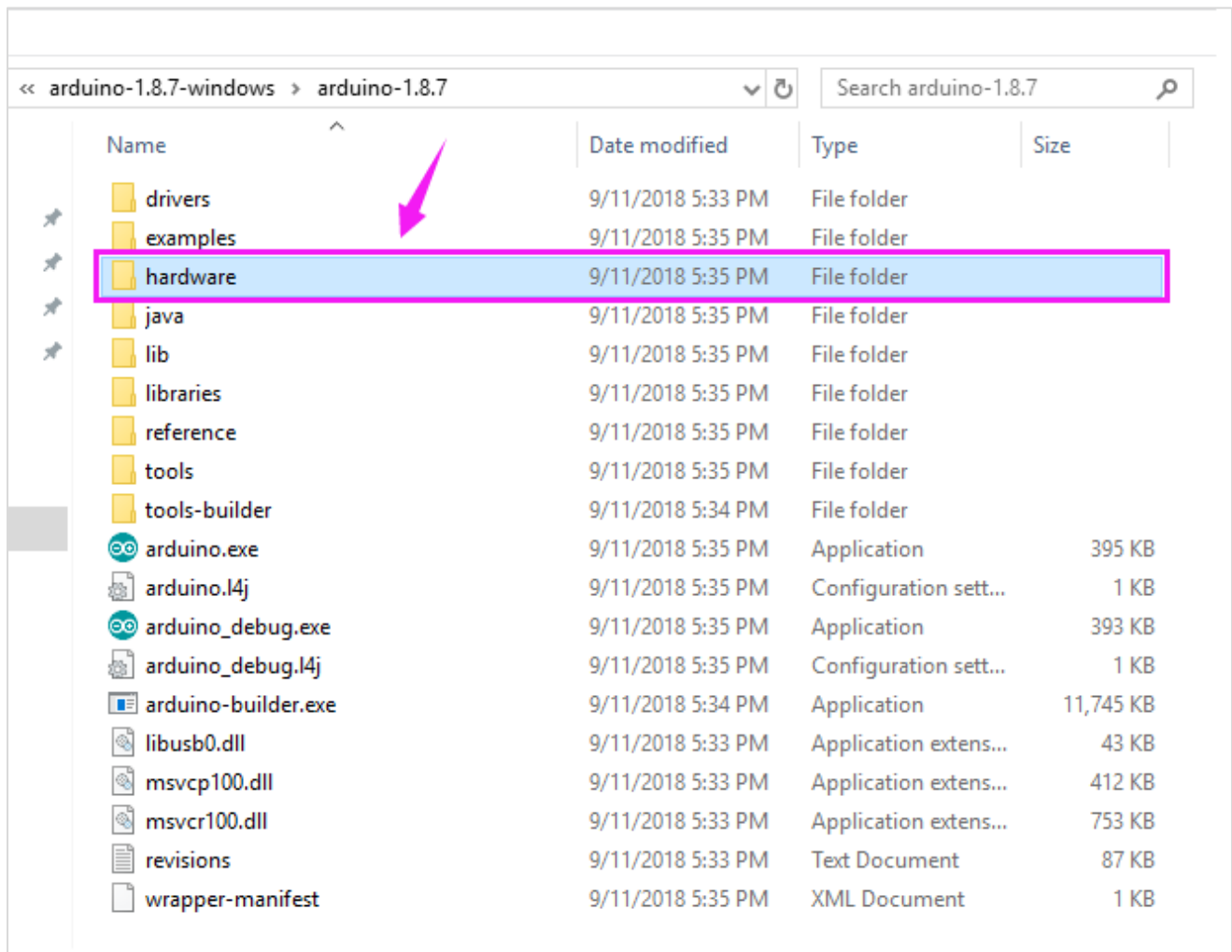
< Back   Next >   Cancel

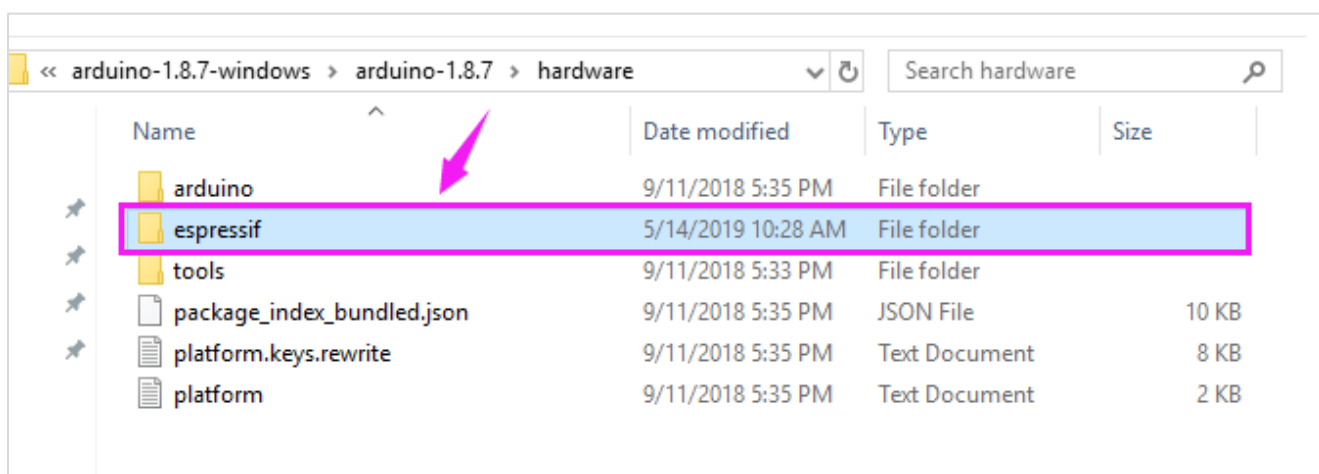Wait for the installation complete. Finally click "Finish" to close the window.

Step3| Building ESP32 Environment

At first, open the Arduino-1.8.7 folder, you will see the hardware folder;

Then open the hardware folder and add a new folder, remember to name it **espressif** shown below.
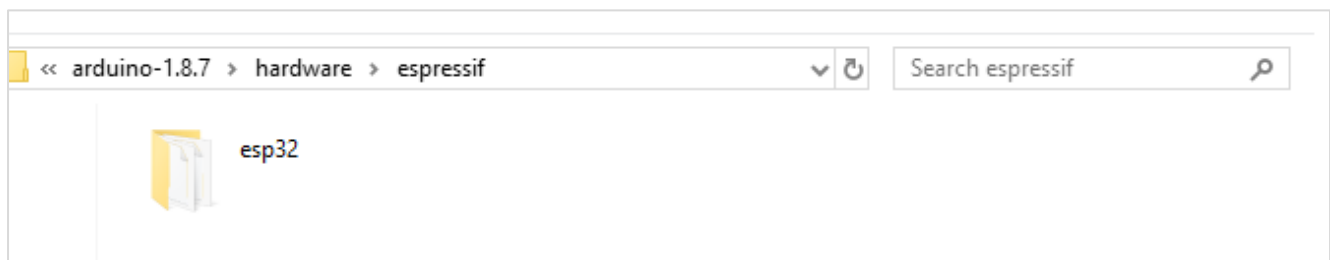
After that, unzip the esp32 compression package we provided, and copy to the

espressif folder.

esp32

So inside the **espressif** folder should see the esp32 folder as below. Note that the

folder should not name a type.

Now, click to enter the esp32 folder and you can see the **tools** folder below.

Name	Date modified	Type	Size
cores	5/14/2019 8:45 AM	File folder	
docs	5/14/2019 8:45 AM	File folder	
libraries	5/14/2019 8:45 AM	File folder	
package	5/14/2019 8:45 AM	File folder	
tools	5/14/2019 8:45 AM	File folder	
variants	5/14/2019 8:45 AM	File folder	
.gitignore	9/27/2018 5:29 AM	GITIGNORE File	1 KB
.gitmodules	9/27/2018 5:29 AM	GITMODULES File	1 KB
.travis.yml	9/27/2018 5:29 AM	YML File	2 KB
appveyor.yml	9/27/2018 5:29 AM	YML File	1 KB
boards	9/27/2018 5:29 AM	Text Document	88 KB
CMakeLists	9/27/2018 5:29 AM	Text Document	8 KB
component.mk	9/27/2018 5:29 AM	MK File	1 KB
Kconfig.projbuild	9/27/2018 5:29 AM	PROJBUILD File	7 KB
Makefile.projbuild	9/27/2018 5:29 AM	PROJBUILD File	1 KB
package.json	9/27/2018 5:29 AM	JSON File	1 KB
platform	9/27/2018 5:29 AM	Text Document	10 KB
programmers	9/27/2018 5:29 AM	Text Document	0 KB
README.md	9/27/2018 5:29 AM	MD File	4 KB

Enter the **tools** folder and click to run the **get.exe** application as an administrator.
(But the precondition is that you have already installed the Python)
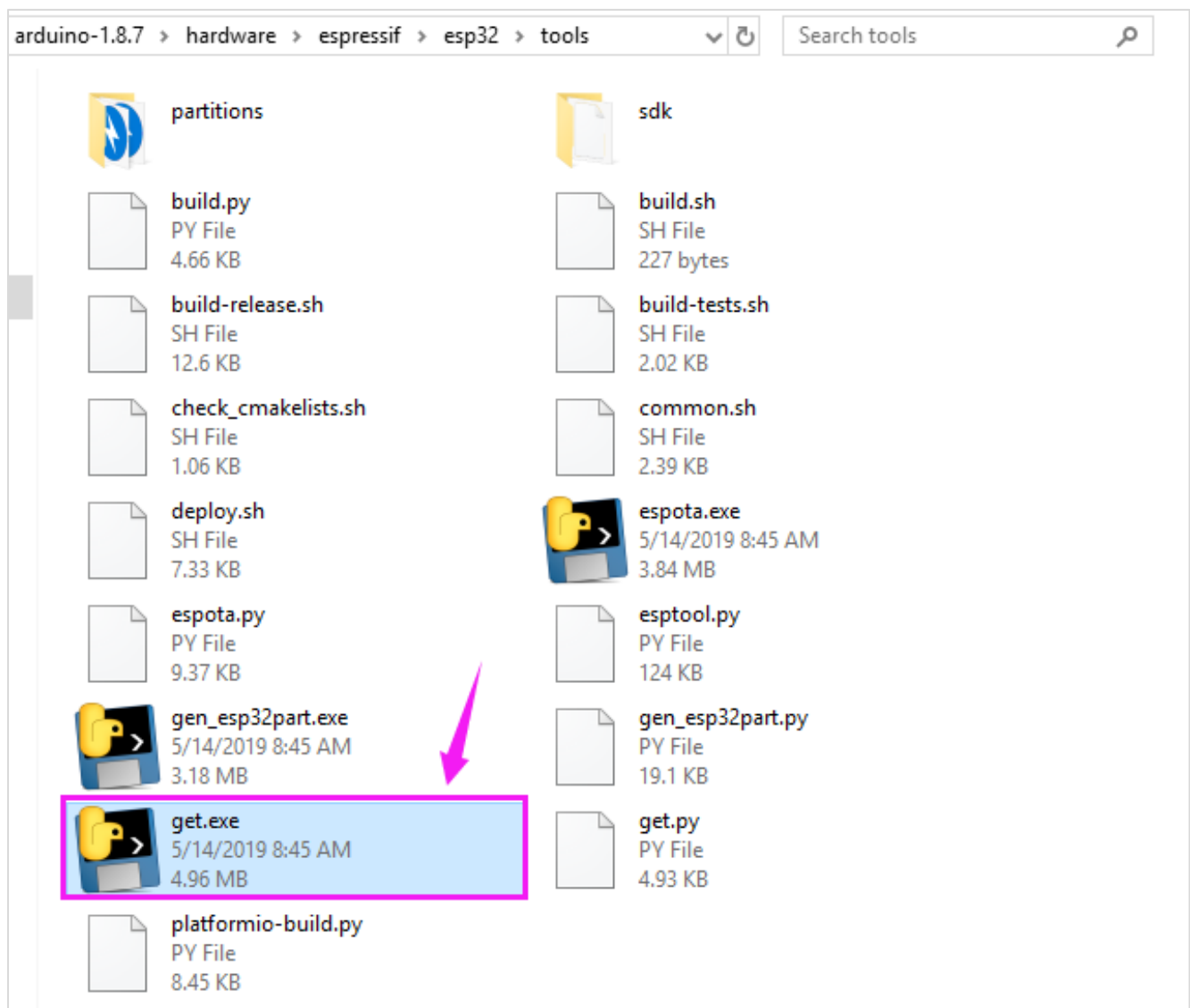
When run the **get.exe** application, ensure that your network is unblocked and wait for the program download. Done downloading, the following window will automatically close.

```
System: Windows, Info: Windows-10-10.0.17763
Platform: i686-mingw32
Downloading xtensa-esp32-elf-win32-1.22.0-80-g6c4433a-5.2.0.zip
Done
Extracting xtensa-esp32-elf-win32-1.22.0-80-g6c4433a-5.2.0.zip
Downloading esptool-2.5.0-windows.zip
Done
Extracting esptool-2.5.0-windows.zip
Downloading mkspiffs-0.2.3-arduino-esp32-win32.zip
Done
Extracting mkspiffs-0.2.3-arduino-esp32-win32.zip
```

```
F:\ARDUINO software\arduino-1.8.7-windows\arduino-1.8.7\hardware\espressif\esp32\tools\get.exe        —    □    ×
System: Windows, Info: Windows-10-10.0.17763
Platform: i686-mingw32
Tool xtensa-esp32-elf-win32-1.22.0-80-g6c4433a-5.2.0.zip already downloaded
Extracting xtensa-esp32-elf-win32-1.22.0-80-g6c4433a-5.2.0.zip
```

Step4| Arduino IDE Setting and Toolbar

Double-click the icon of Arduino software downloaded to open the IDE.

This is your Arduino 1.8.7 interface.

arduino.exe
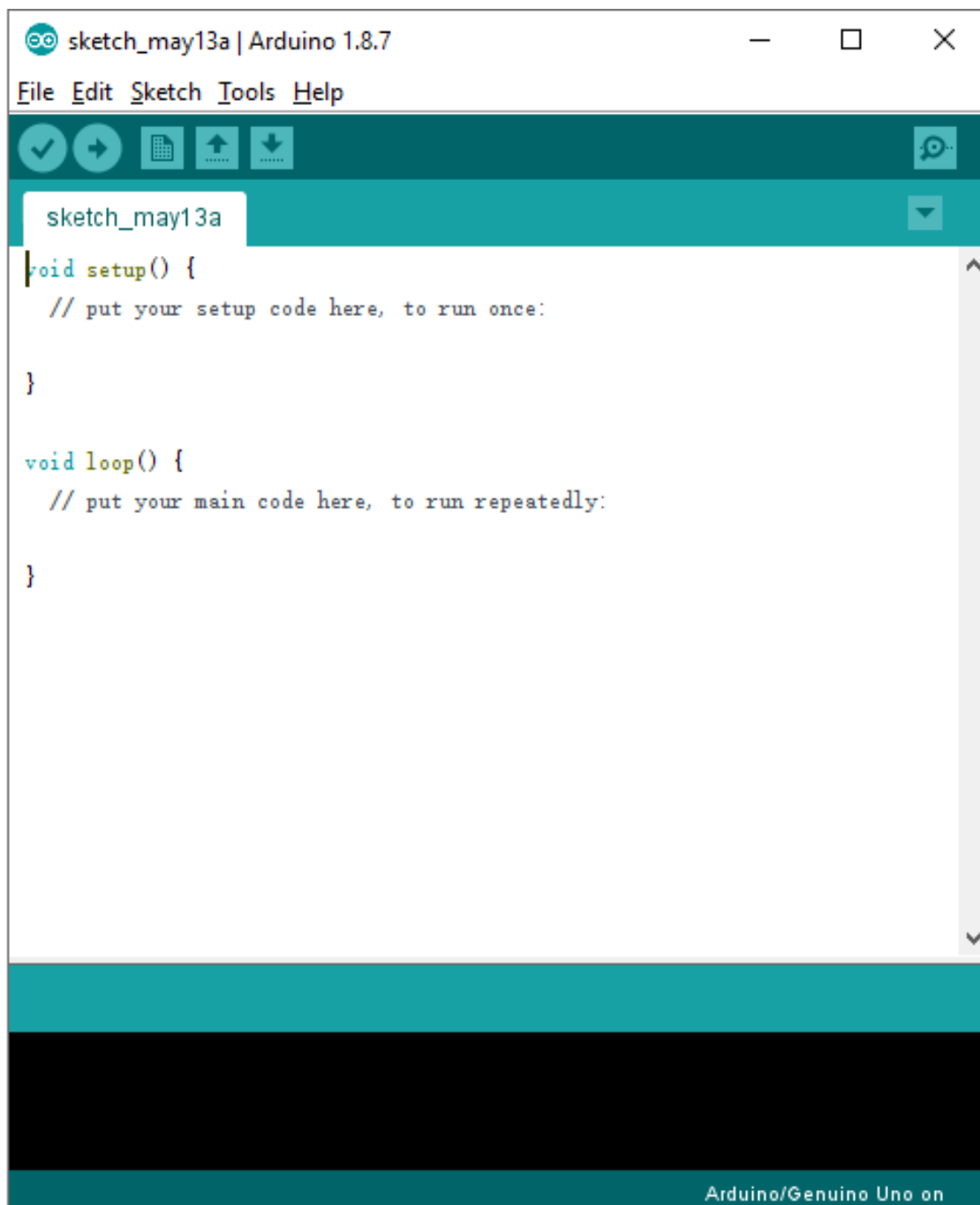
(**Note:** if the Arduino software loads in the wrong language, you can change it in the preferences dialog. See the environment page for details.)

The functions of each button on the Toolbar are listed below:

Verify/Compile	Check the code for errors
Upload	Upload the current Sketch to the Arduino
New	Create a new blank Sketch
Open	Show a list of Sketches
Save	Save the current Sketch
Serial Monitor	Display the serial data being sent from the Arduino

Attach your ESP32 core board to your computer with the USB cable.
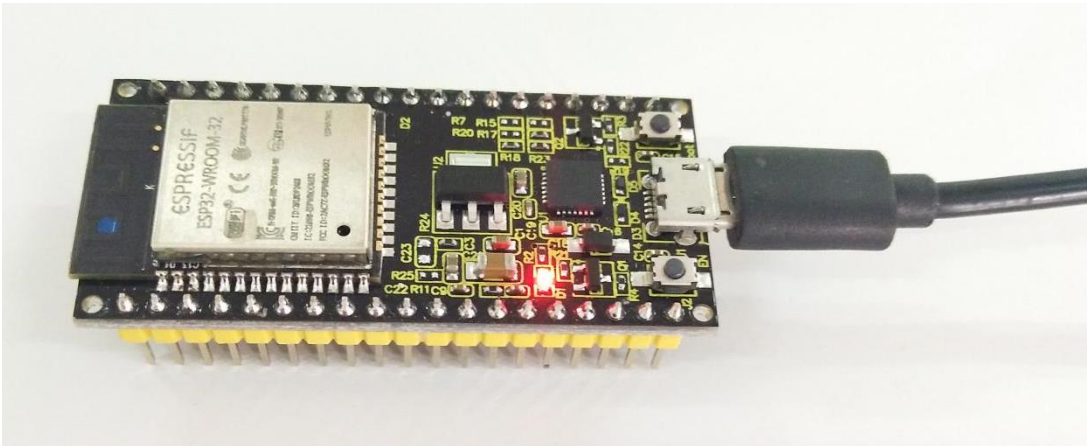
Check that the "Board Type"and "Serial Port" are set correctly.

Click to open the "**Tools"**, for "**Board"**, scroll to select the ESP32 Dev Module.
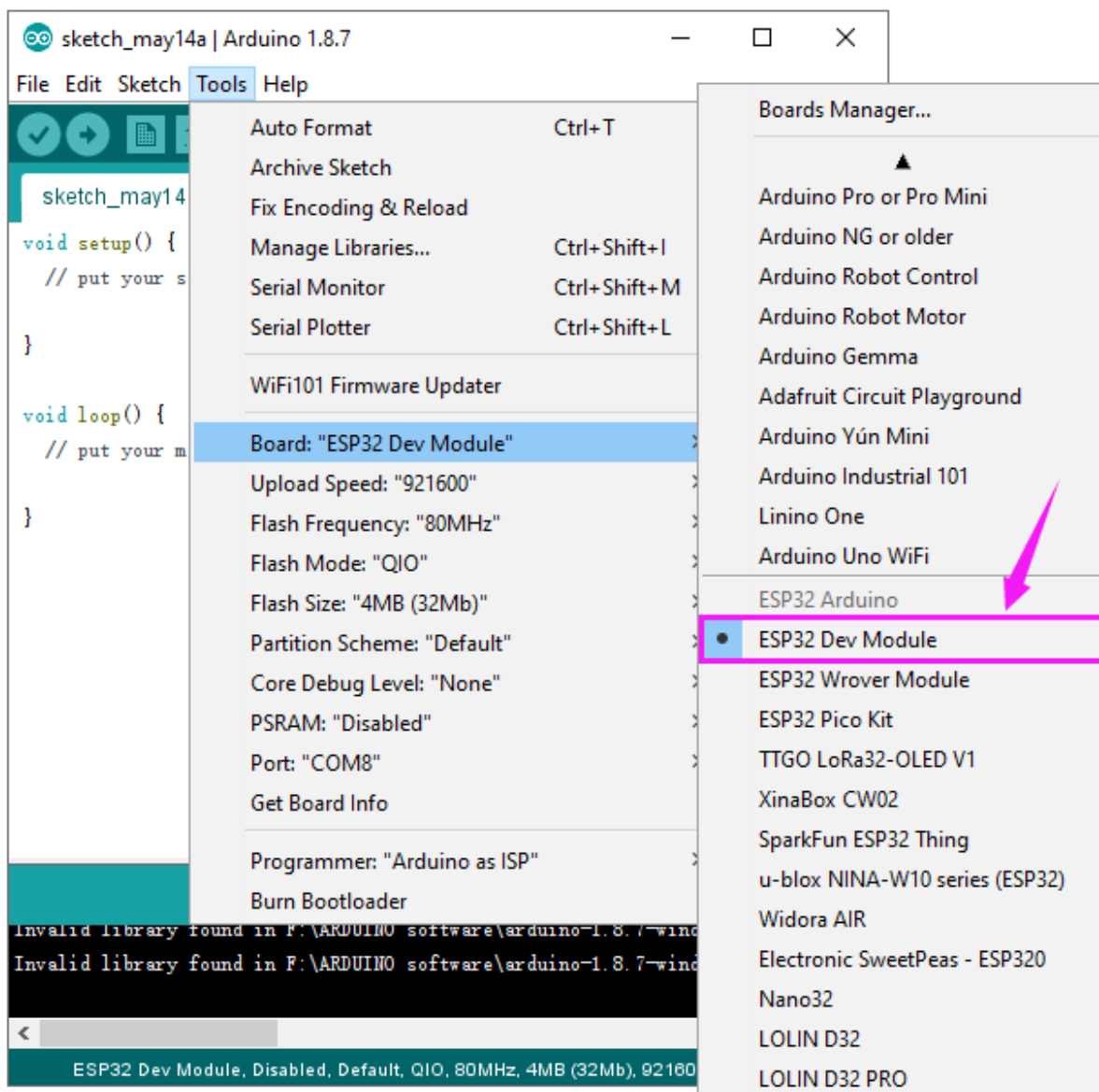
Select well the correct board and then should set the detailed information as shown below.
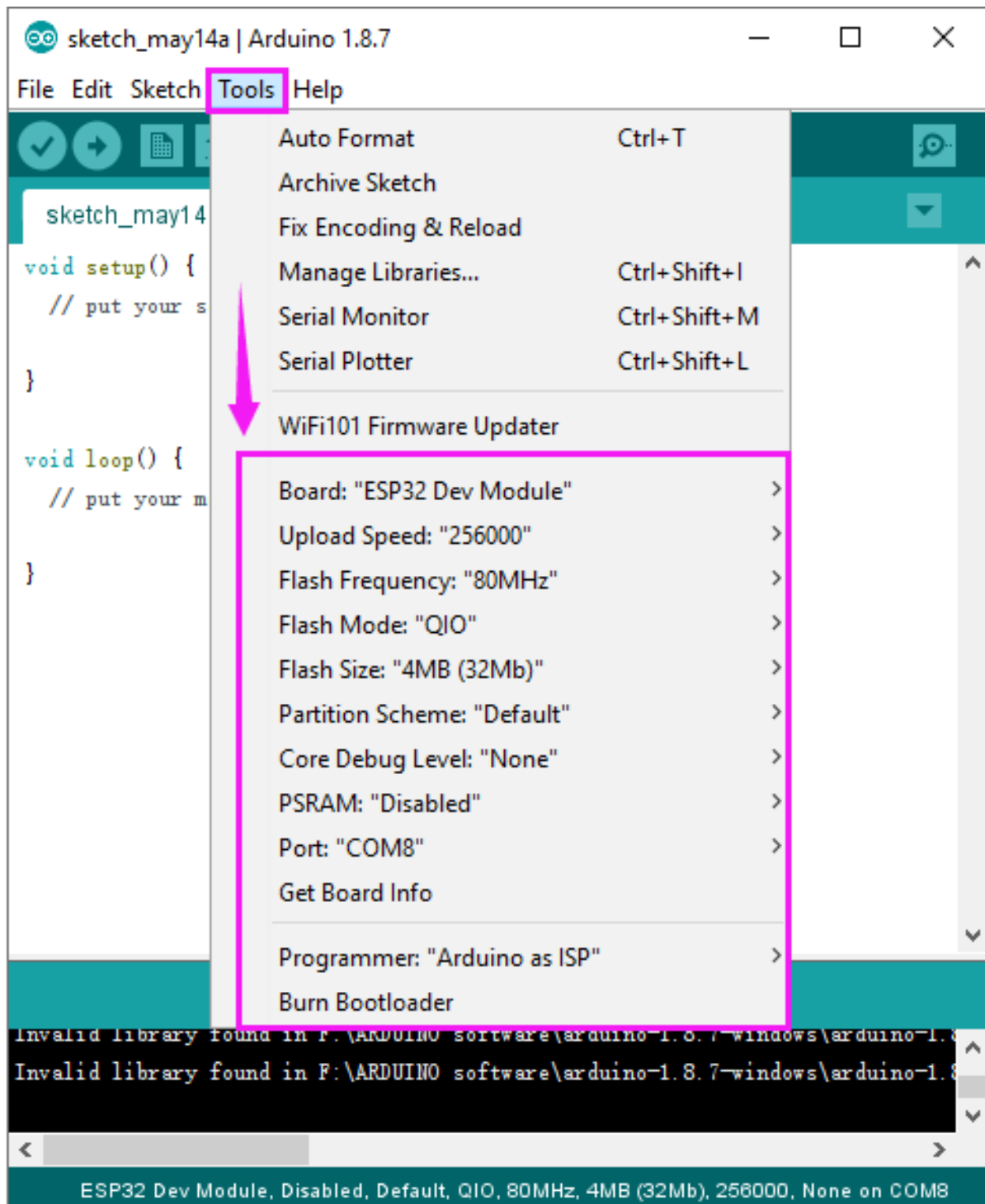
Pay close attention to select the proper **COM** port. (Arduino driver installed well, you are supposed to see the corresponding port.)

Check out the COM port in the Device Manager of your computer's control panel.

Here we can know the COM port is COM 8. Then select the Port COM 8 in the Arduino Tools.

Step5| Upload the Code

Paste and copy the source code below to Arduino IDE.

Special Note: when compile and upload the source code, hold the BOOT button on the ESP32 board until upload well the code.

....
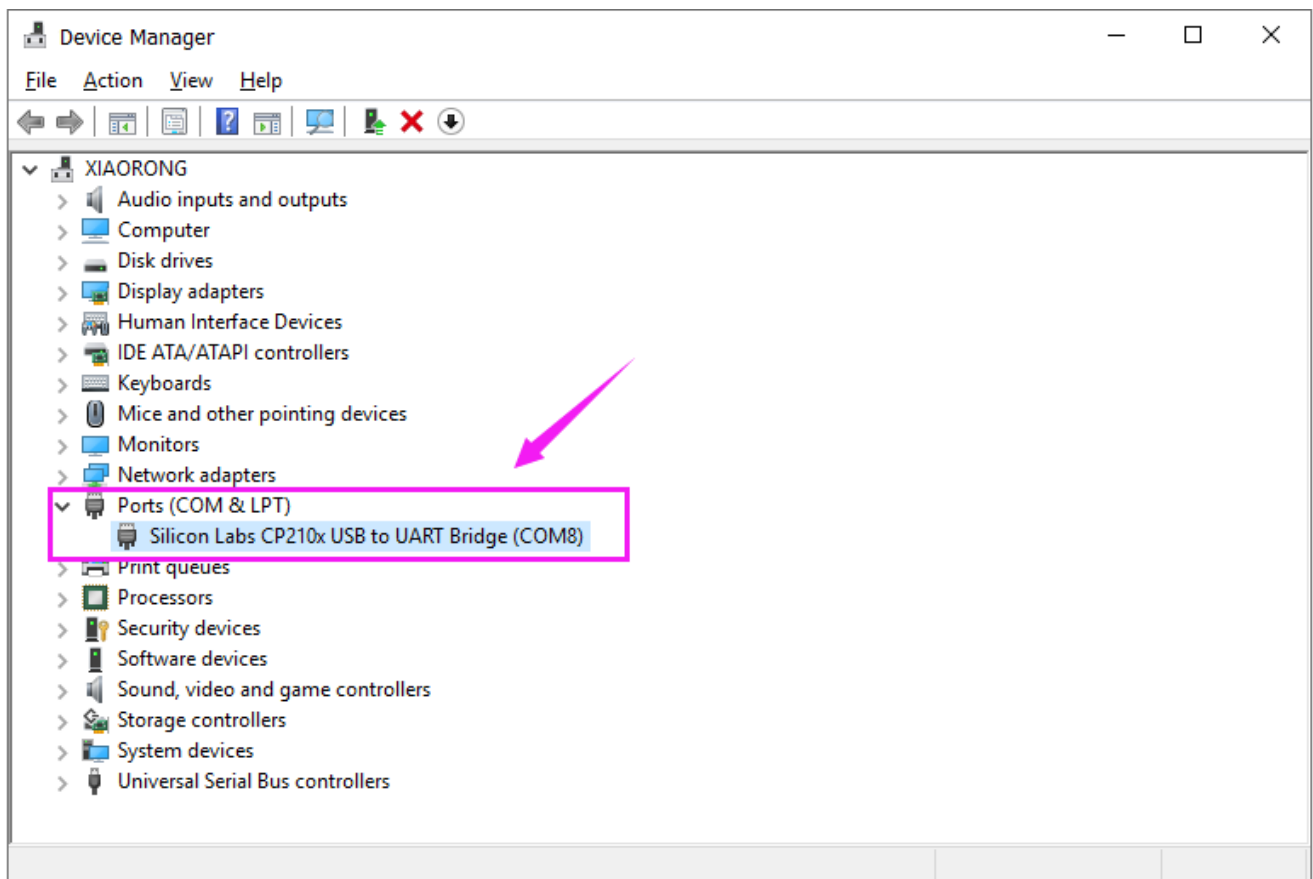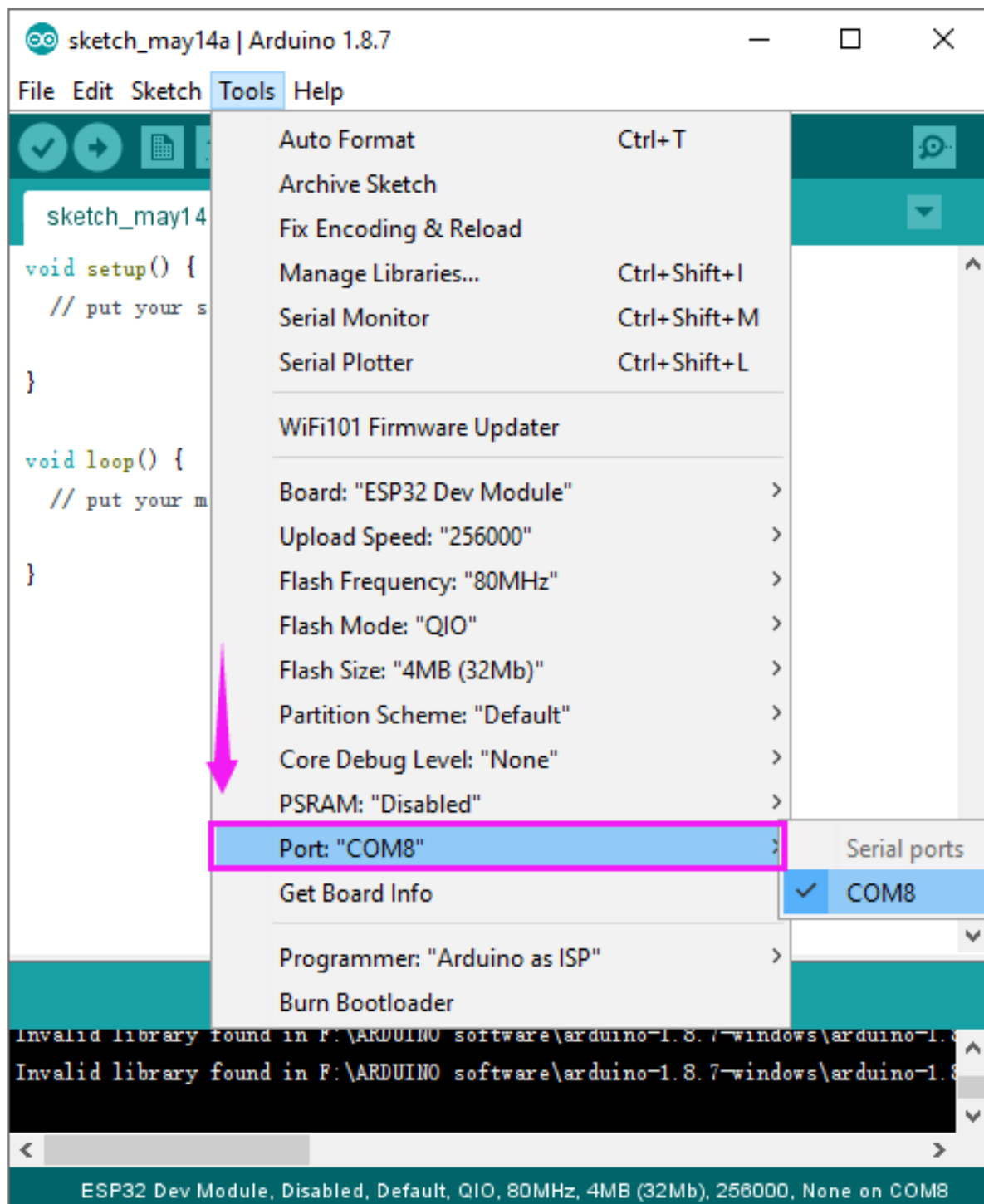
```
/* *   This sketch demonstrates how to scan WiFi networks. *   The API is almost the same as with the WiFi Shield library, *   the most obvious difference being the different file you need to include: */#include "WiFi.h"void setup(){    Serial.begin(115200);    // Set WiFi to station mode and disconnect from an AP if it was previously connected    WiFi.mode(WIFI_STA);  WiFi.disconnect();    delay(100);    Serial.println("Setup done");}void loop(){    Serial.println("scan start");    // WiFi.scanNetworks will return the number of networks found    int n = WiFi.scanNetworks();  Serial.println("scan done");    if (n == 0) {    Serial.println("no networks found");    } else {    Serial.print(n);    Serial.println(" networks found");      for (int i = 0; i < n; ++i) {    // Print SSID and RSSI for each network found    Serial.print(i + 1);    Serial.print(": ");    Serial.print(WiFi.SSID(i));    Serial.print(" (");  Serial.print(WiFi.RSSI(i));    Serial.print(")");  Serial.println((WiFi.encryptionType(i) == WIFI_AUTH_OPEN)?" ":"*");  delay(10);    }    }    Serial.println("");    // Wait a bit before scanning again    delay(5000);}
```

....

Click verify button to check the errors. If compiling successfully, the message "Done compiling." will appear in the status bar.
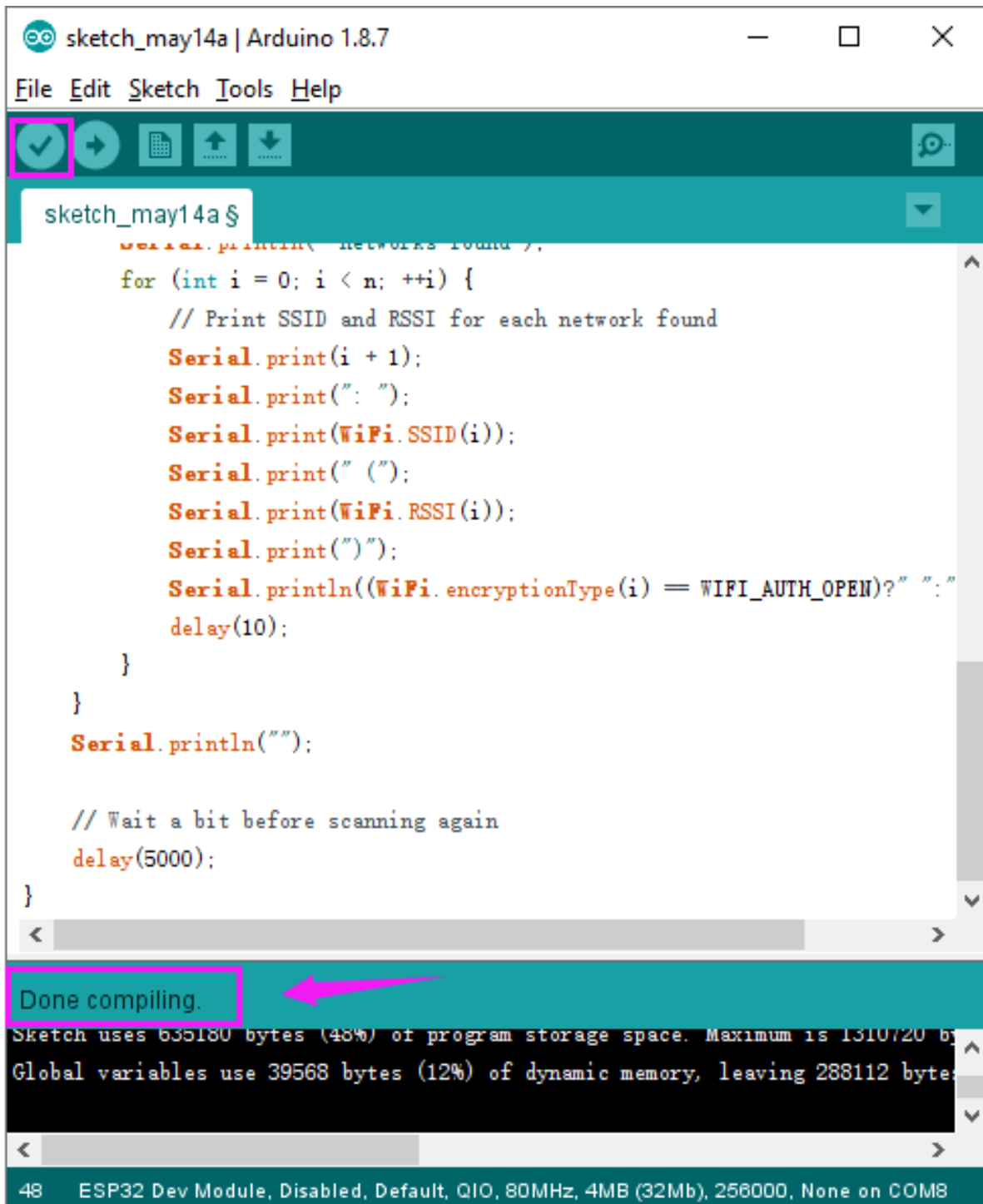
After that, click the "Upload" button to upload the code. If the upload is successful, the message "Done uploading." will appear in the status bar.

Special Note: if fail to upload, when upload the source code, hold the BOOT button on the ESP32 board until upload well the code.

```
Serial.println("Networks found");
        for (int i = 0; i < n; ++i) {
            // Print SSID and RSSI for each network found
            Serial.print(i + 1);
            Serial.print(": ");
            Serial.print(WiFi.SSID(i));
            Serial.print(" (");
            Serial.print(WiFi.RSSI(i));
            Serial.print(")");
            Serial.println((WiFi.encryptionType(i) == WIFI_AUTH_OPEN)?" ":"*
            delay(10);
        }
    }
    Serial.println("");

    // Wait a bit before scanning again
    delay(5000);
}
```
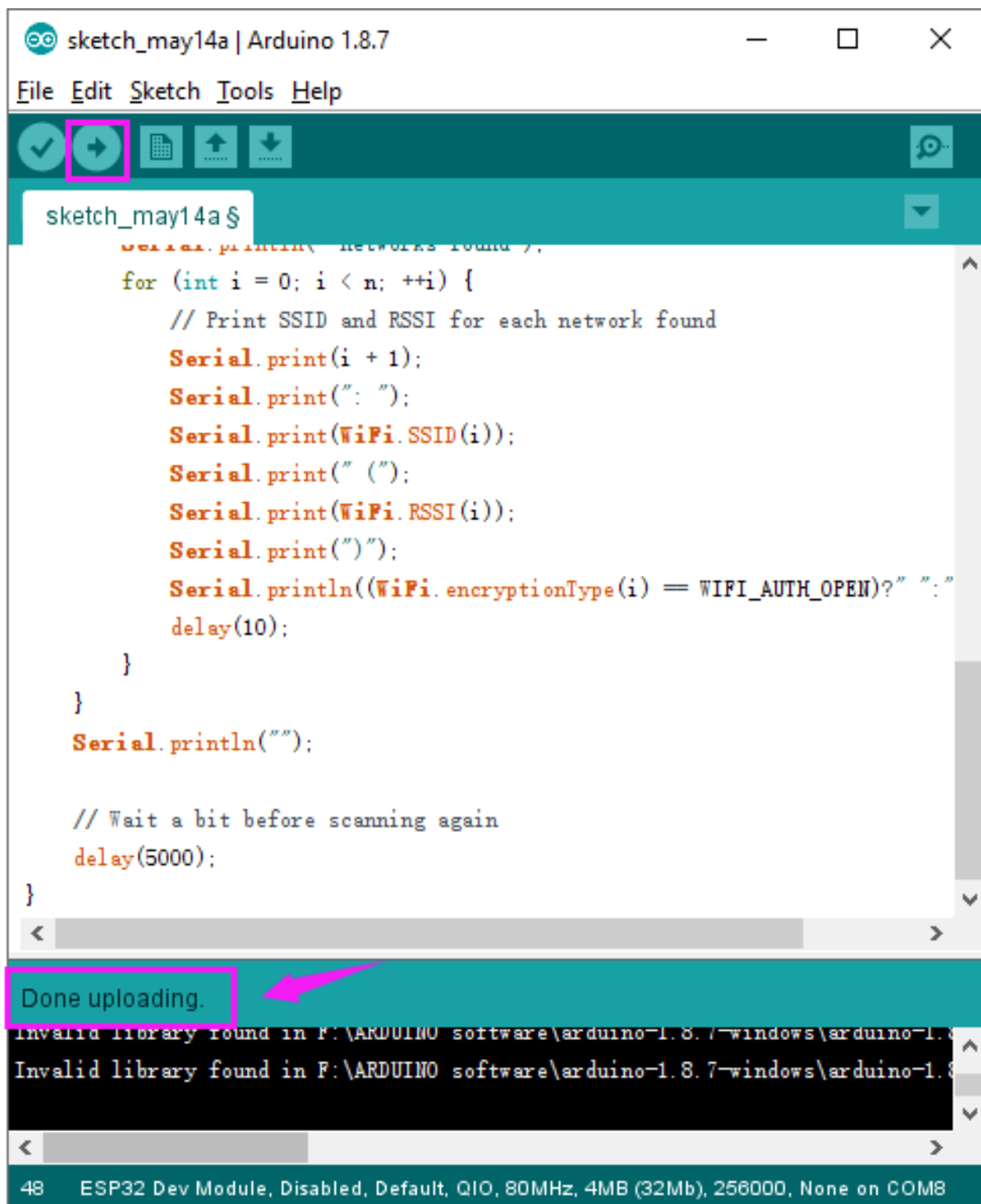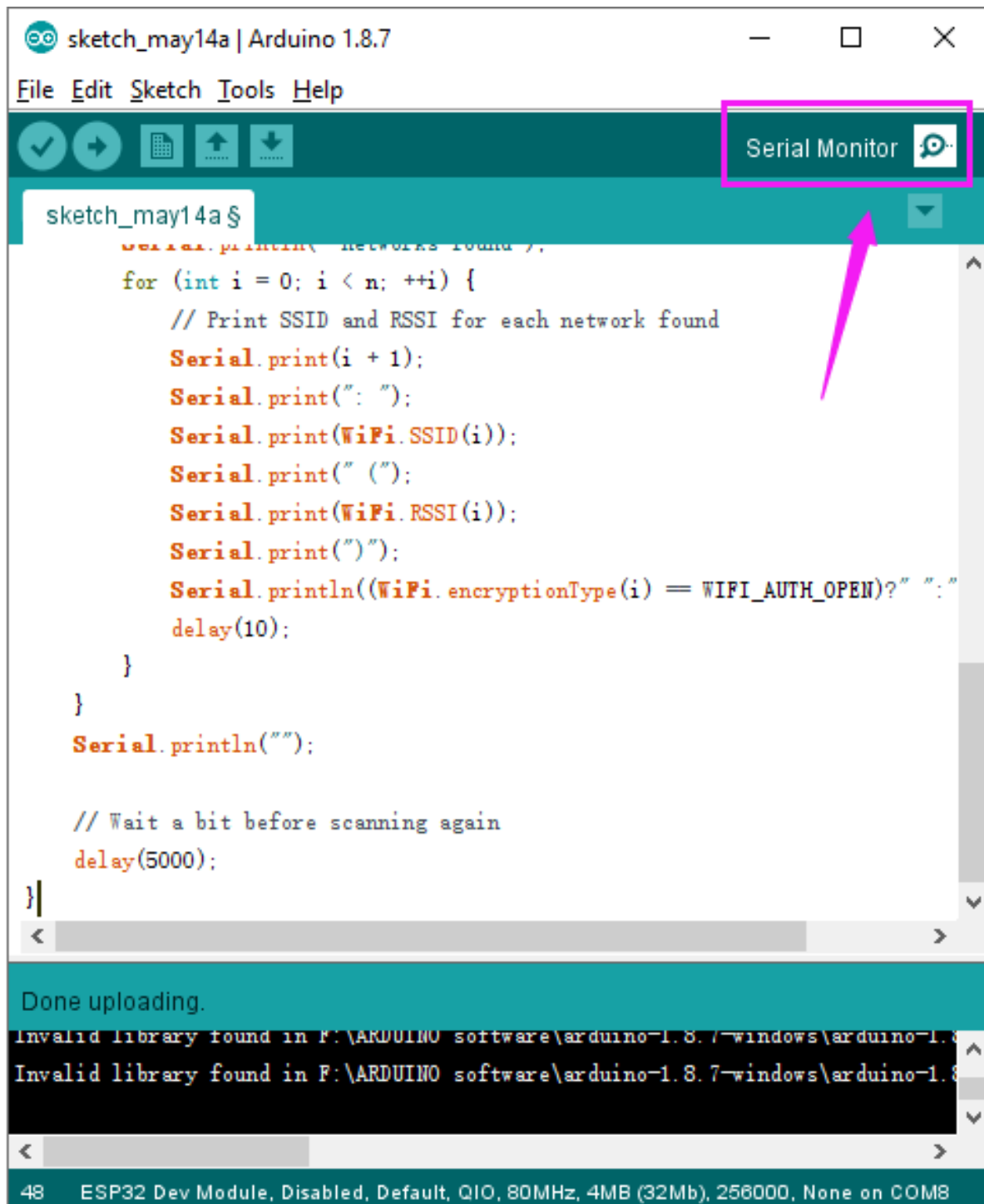
Done uploading.

```
Invalid library found in F:\ARDUINO software\arduino-1.8.7-windows\arduino-1.8
Invalid library found in F:\ARDUINO software\arduino-1.8.7-windows\arduino-1.8
```

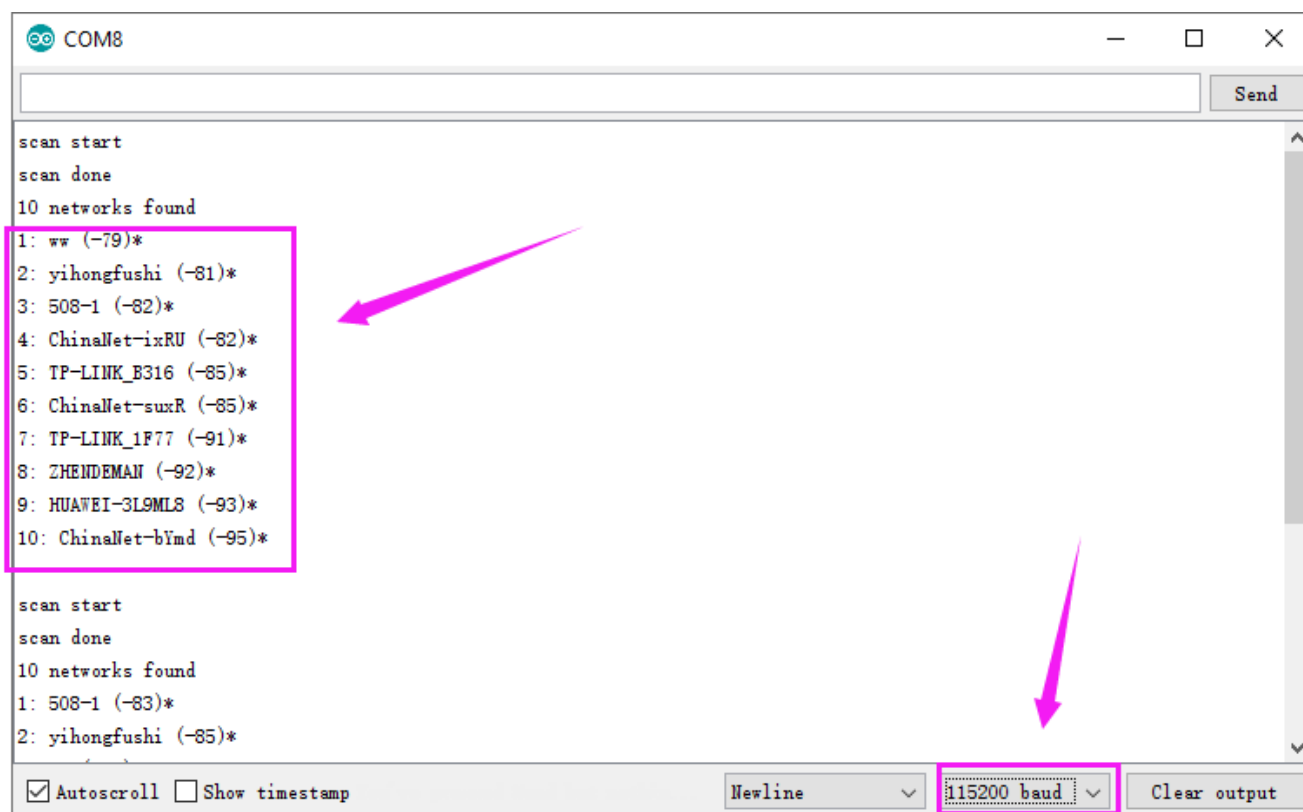48   ESP32 Dev Module, Disabled, Default, QIO, 80MHz, 4MB (32Mb), 256000, None on COM8

Done uploading the code to your board, open the serial monitor and set the baud rate to 115200. You should be able to see the WIFI information on the pop-up

window.

Resource Download:

You can download all the data package from the link:

https://drive.google.com/open?id=1qZ8MGRd-KwlD4wXACALr3P6Vc-4Xib2N

Download the ARDUINO Software:

https://www.arduino.cc/en/Main/OldSoftwareReleases#1.5.x

Download the Driver:

https://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers