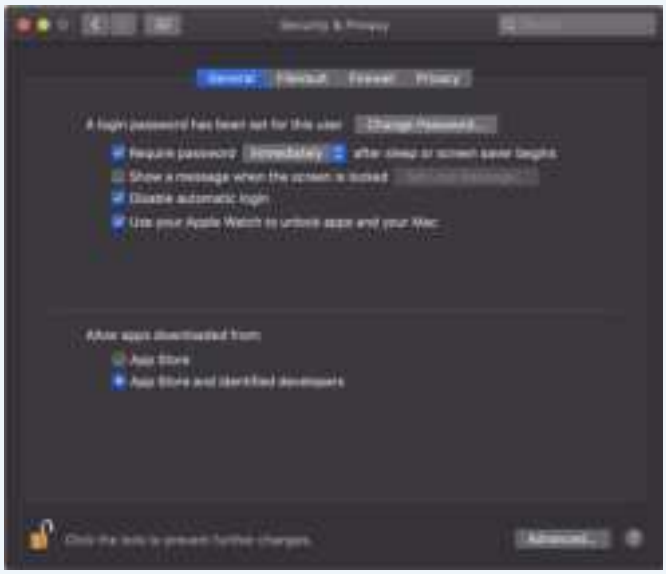
TOUGH Quick Start

USB Driver

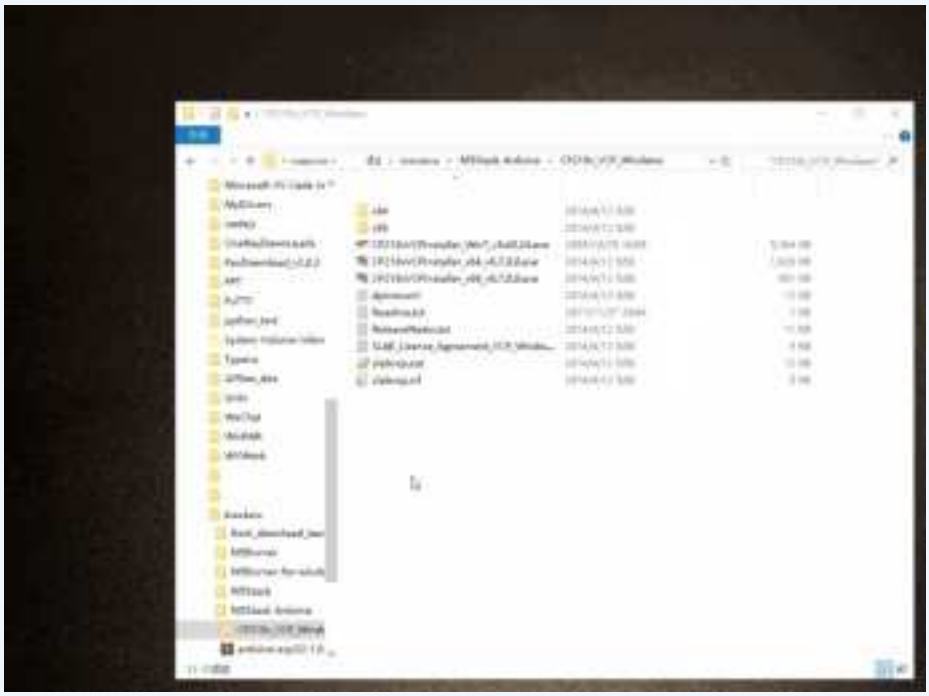
Before the program is burned, users please download the corresponding CP210X driver package according to the operating system you are using, click the button below. After decompressing the compressed package, select the installation package corresponding to the operating system value for installation.

For Mac OS, make sure System preferences -> Security & Privacy -> General before installing, and allow the apps to be installed from the App Store and identified developers



- **Download CP2104 driver**

Windows10 MacOS Linux

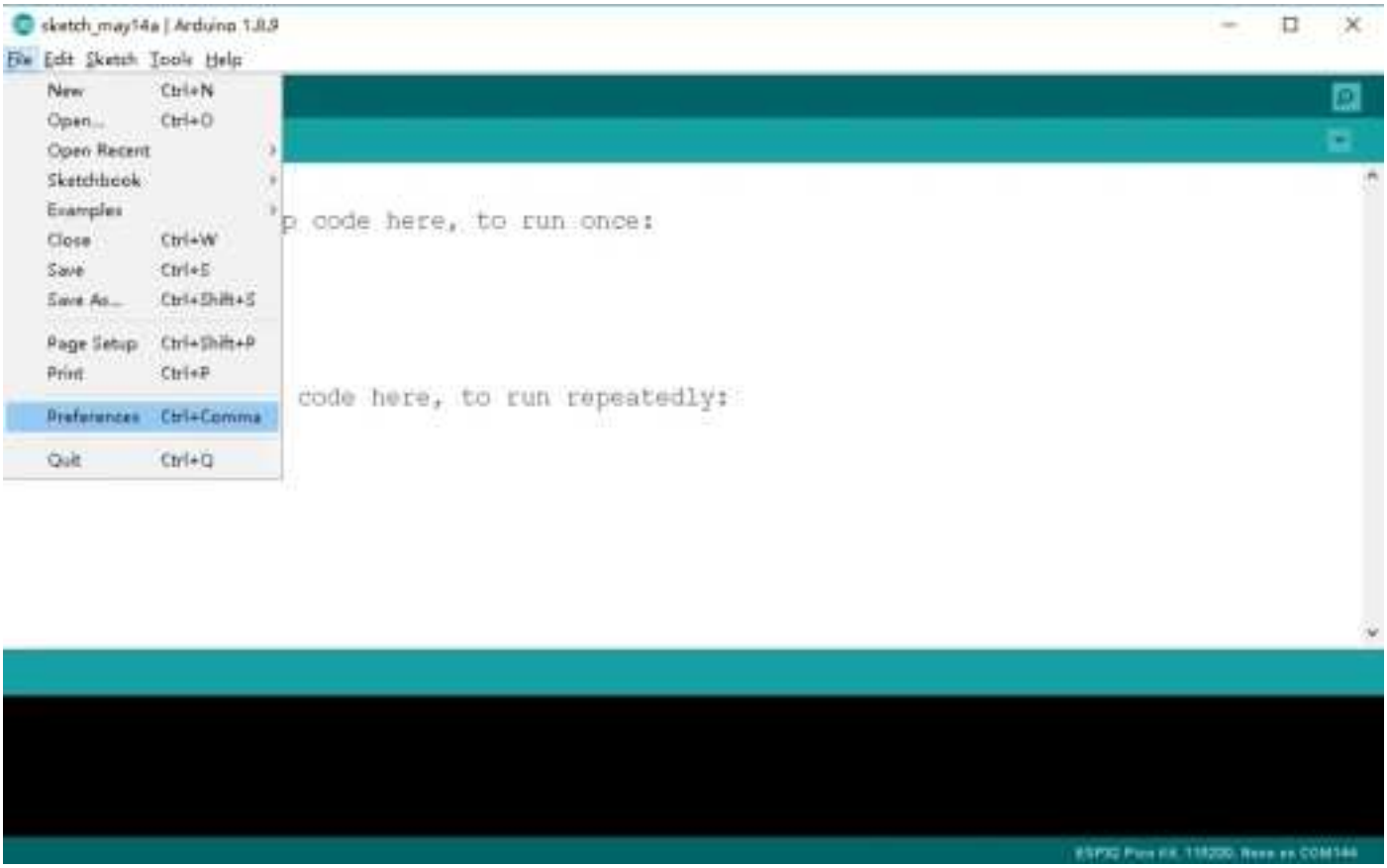


Arduino-IDE

Click here to visit Arduino's official website,Select the installation package for your own operating system to download.

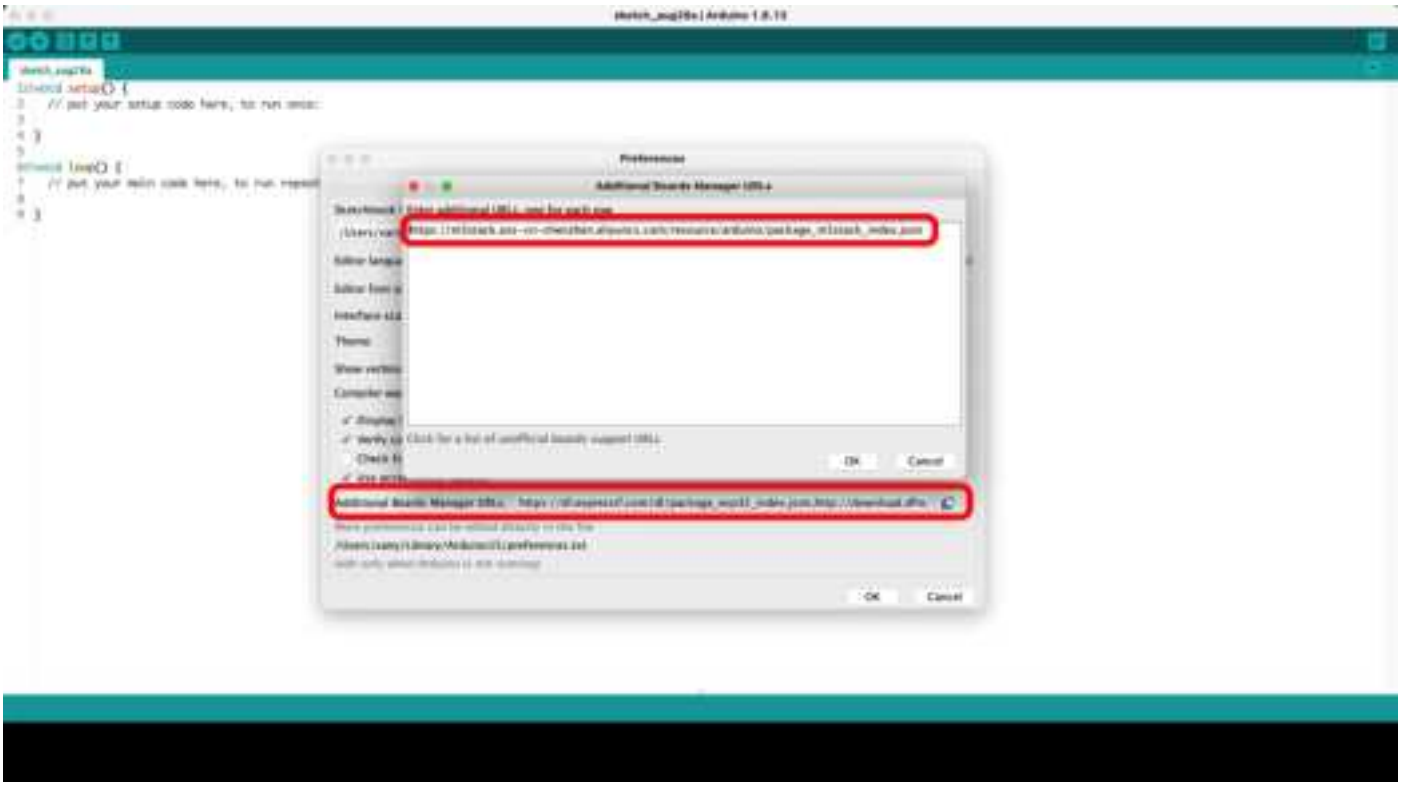
M5Stack Boards Manager

1.Open up Arduino IDE, navigate to `File` -> `Peferences` -> `Settings`

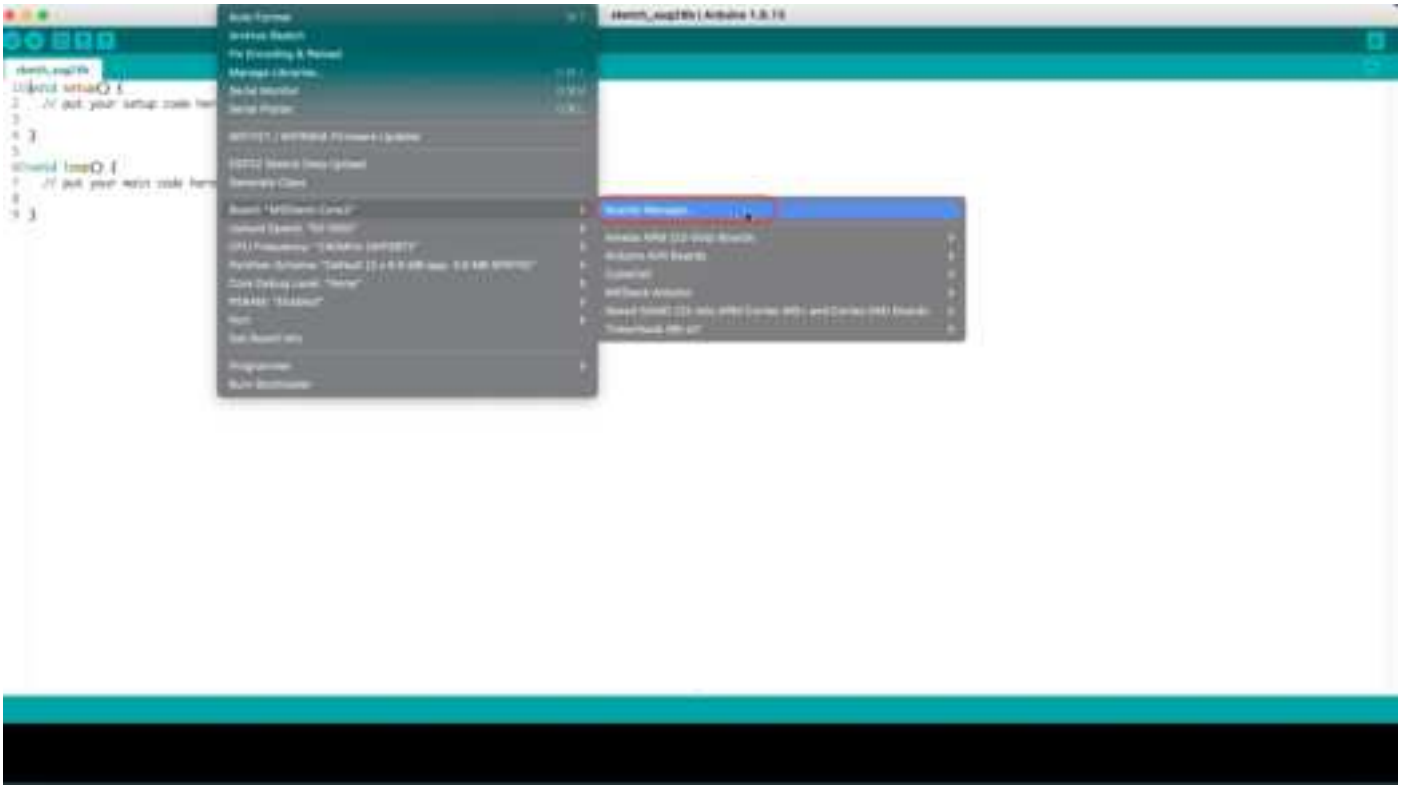


2.Copy the following M5Stack Boards Manager url to `Additional Boards Manager URLs:`

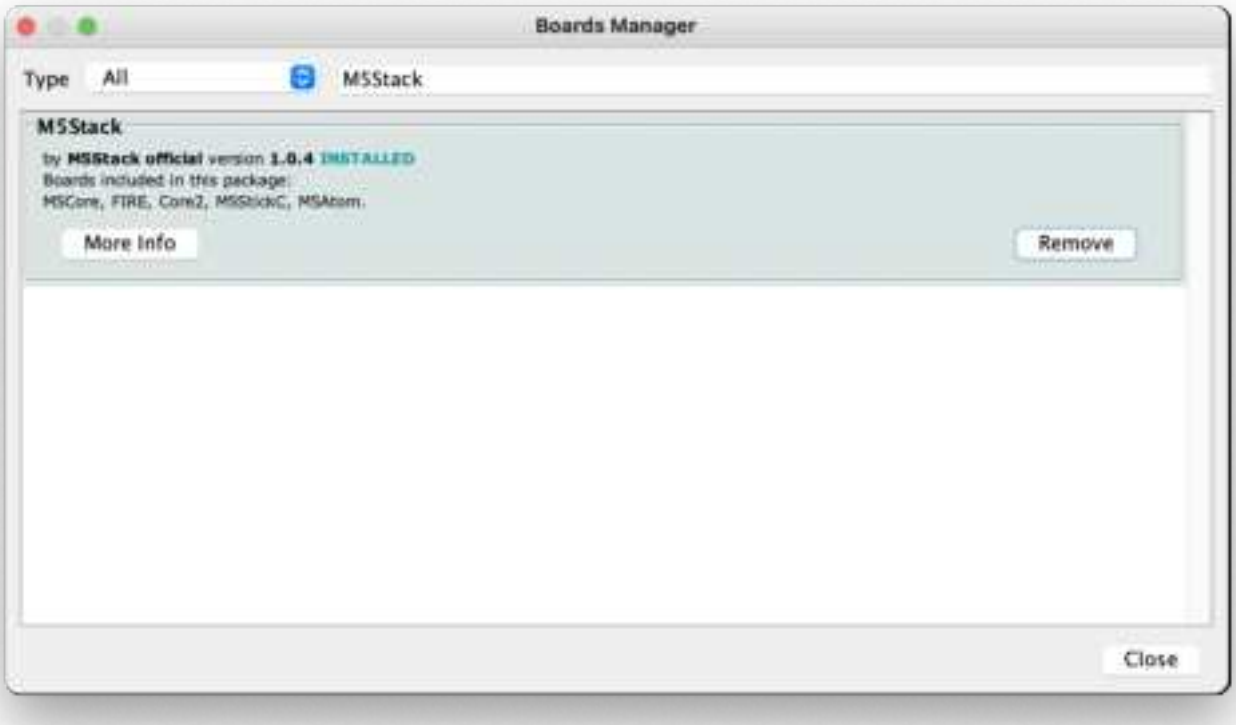
https://m5stack.oss-cn-shenzhen.aliyuncs.com/resource/arduino/package_m5stack_index.json



3.Navigate to `Tools` -> `Board:` -> `Boards Manager...`



4.Search `M5Stack` in the pop-up window, find it and click `Install`



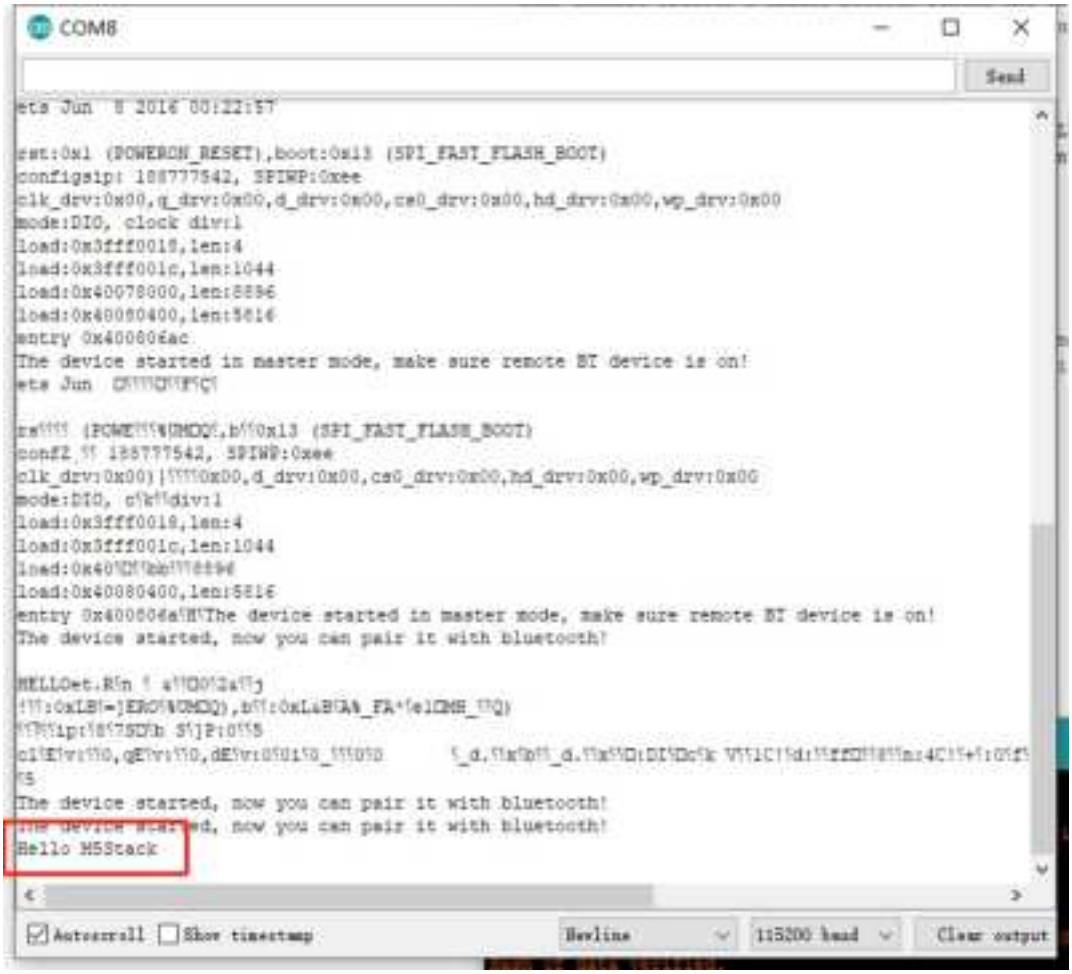
5.select `Tools` -> `Board:` -> `M5Stack-Tough`

6. `Visit Github` (https://github.com/m5stack/M5Tough), download the M5Tough library, and place it in the

> Arduino library file path `C:\Users\UserName\Documents\Arduino\libraries`

Bluetooth serial port function

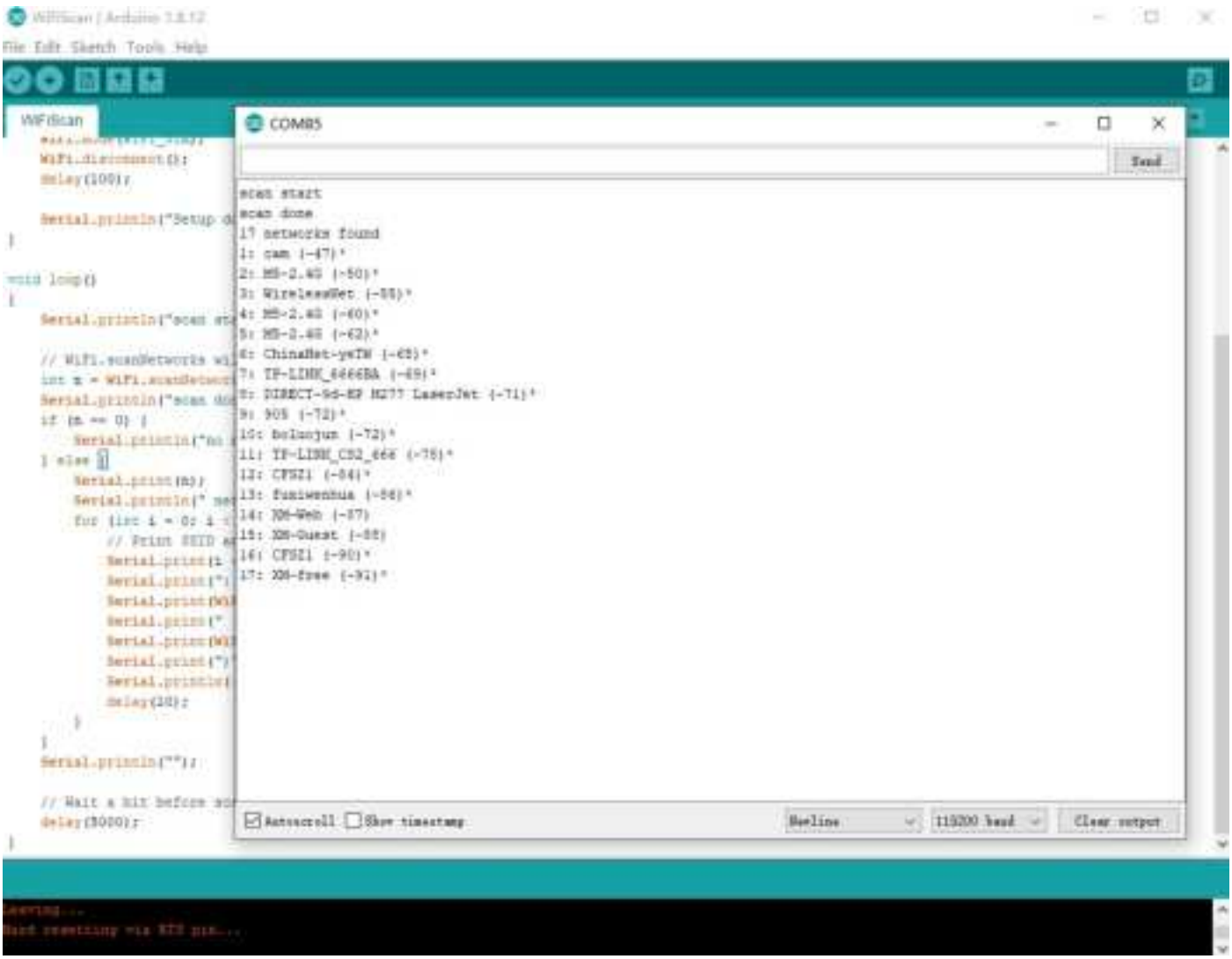
> Open the Arduino IDE and open the example program `File` -> `Examples` -> `BluetoothSerial` -> `SerialToSerialBT` . Connect the device to the computer and select the corresponding port to burn. After completion, the device will automatically run Bluetooth, and the device name is `ESP32test` . At this time, use the Bluetooth serial port sending tool on the PC to realize the transparent transmission of Bluetooth serial data.




```
#include "BluetoothSerial.h"

#if !defined(CONFIG_BT_ENABLED) || !defined(CONFIG_BLUEDROID_ENABLED)
#error Bluetooth is not enabled! Please run `make menuconfig` to and enable it
#endif

BluetoothSerial SerialBT;

void setup() {
  Serial.begin(115200);
  SerialBT.begin("ESP32test"); //Bluetooth device name
  Serial.println("The device started, now you can pair it with bluetooth!");
}

void loop() {
  if (Serial.available()) {
    SerialBT.write(Serial.read());
  }
  if (SerialBT.available()) {
    Serial.write(SerialBT.read());
  }
  delay(20);
}
```

WiFi scanning function

> Open the Arduino IDE and open the example program `File` -> `Examples` -> `WiFi` -> `WiFiScan` . Connect the device to the computer and select the corresponding port to burn. After completion, the device will automatically run the WiFi scan, and the current WiFi scan result can be obtained through the serial port monitor that comes with the Arduino.




```
#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);
}
```

FCC Statement:

Any Changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.

This device complies with part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) This device may not cause harmful interference, and (2) This device must accept any interference received, including interference that may cause undesired operation.

FCC Radiation Exposure Statement:

This equipment complies with FCC radiation exposure limits set forth for an uncontrolled environment .This equipment should be installed and operated with minimum distance 20cm between the radiator& your body.

Note: This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However,there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

—Reorient or relocate the receiving antenna.

—Increase the separation between the equipment and receiver.

—Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.

—Consult the dealer or an experienced radio/TV technician for help.