

## .....Table of contents.....

一、Introduction	2
二、Installing using Arduino IDE	2
三、sample program usage	11

# Getting Started

## Introduction

The objective of this post is to explain how to upload an Arduino program to the ESP32-4848S040 module.

The ESP32 WiFi and Bluetooth chip is the latest generation of Espressif products. It has a dual-core 32-bit MCU, which integrates WiFi HT40 and Bluetooth/BLE 4.2 technology inside.

ESP32-S3-wroom-1 has a significant performance improvement. It is equipped with a high-performance dual-core Tensilica LX7 MCU. One core handles high speed connection and the other for standalone application development. The dual-core MCU has a 240 MHz frequency and a computing power of 600 DMIPS.

In addition, it supports Wi-Fi HT40, Classic Bluetooth/BLE 4.2, and more GPIO resources.

## 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 1.8.19 of the Arduino IDE (or greater), for example, the Arduino installation was in "C/Programs(x86)/Arduino".

download release link:

<https://downloads.arduino.cc/arduino-1.8.19-windows.exe>

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.

```
// An example showing rainbow colours on a 1.8" TFT LCD screen
// and to show a basic example of font use.

Make sure all the display driver and pin connections are correct by
editing the User_Setup.h file in the TFT_eSPI library folder.

Note that yield() or delay(0) must be called in long duration for/while
loops to stop the ESP8266 watchdog triggering.

#####
##### DON'T FORGET TO UPDATE THE User_Setup.h FILE IN THE LIBRARY #####
#####

#include <TFT_eSPI.h> // Graphics and font library for ST7735 driver chip
#include <SPI.h>

TFT_eSPI tft = TFT_eSPI(); // Invoke library, pins defined in User_Setup.h

unsigned long targetTime = 0;
```

Invalid library found in C:\Users\zhang'pei\Documents\Arduino\libraries\Touch\_test: no headers files (.h) found in C:\U  
Invalid library found in C:\Users\zhang'pei\Documents\Arduino\libraries\Touch\_test: no headers files (.h) found in C:\U

```
ESP32 Dev Module, Disabled, Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS), 240MHz (WiFi/BT), DIO, 80MHz, 4MB (32Mb), 921600, Core 1, Core 1, None on COM6
```

- (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.

The screenshot shows the Arduino IDE interface with the title bar "3\_4\_TFT\_Rainbow | Arduino 1.8.19". The menu bar includes File, Edit, Sketch, Tools, and Help. A context menu is open over some code, with options like New, Open..., Open Recent, Sketchbook, Examples, Close, Save, Save As..., Page Setup, Print, Preferences (which is highlighted in blue), and Quit. The main code area contains several lines of TFT library code, including font definitions and drawCentreString() calls. At the bottom of the code area, there are two error messages: "Invalid library found in C:\Users\zhang'pei\Documents\Arduino\libraries\Touch\_test: no headers files (.h) found in C:\U" repeated twice. The status bar at the bottom right shows "ESP32 Dev Module, Disabled, Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS), 240MHz (WiFi/BT), DIO, 80MHz, 4MB (32Mb), 921600, Core 1, Core 1, None on COM6".

```
reen << 5 | blue;
font still works as before
);
suit font!");

do not use the .setCursor call, coords are embedded
CK, TFT_BLACK); // Do not plot the background colour

// Overlay the black text on top of the rainbow plot (the advantage of not drawing the backgorund colour!)
tft.drawCentreString("Font size 2", 80, 14, 2); // Draw text centre at position 80, 12 using font 2

//tft.drawCentreString("Font size 2",81,12,2); // Draw text centre at position 80, 12 using font 2

tft.drawCentreString("Font size 4", 80, 30, 4); // Draw text centre at position 80, 24 using font 4

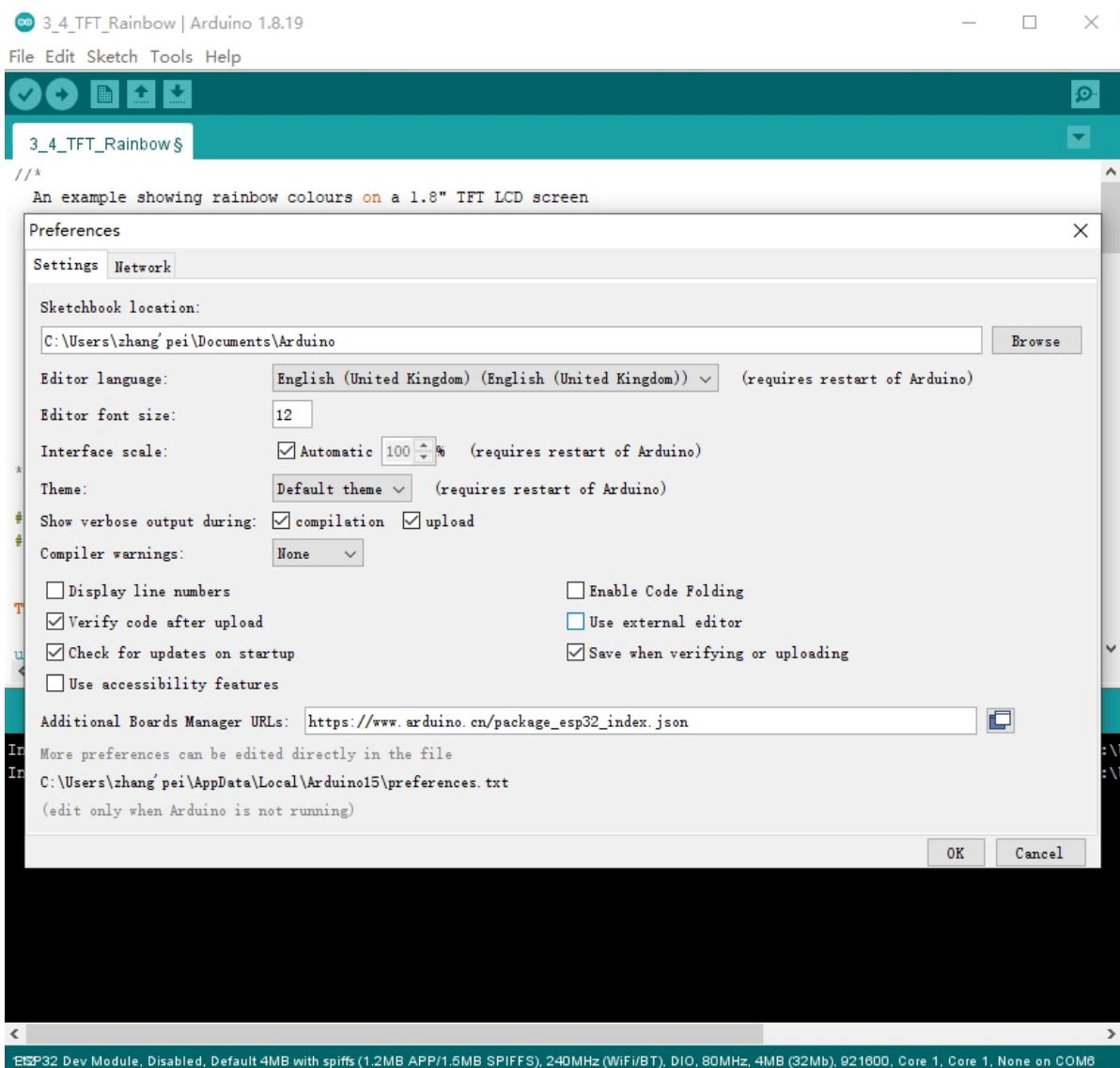
tft.drawCentreString("12.34", 80, 54, 6); // Draw text centre at position 80, 24 using font 6
```

- (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://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package\\_esp32\\_index.json](https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json)  
Development release link:

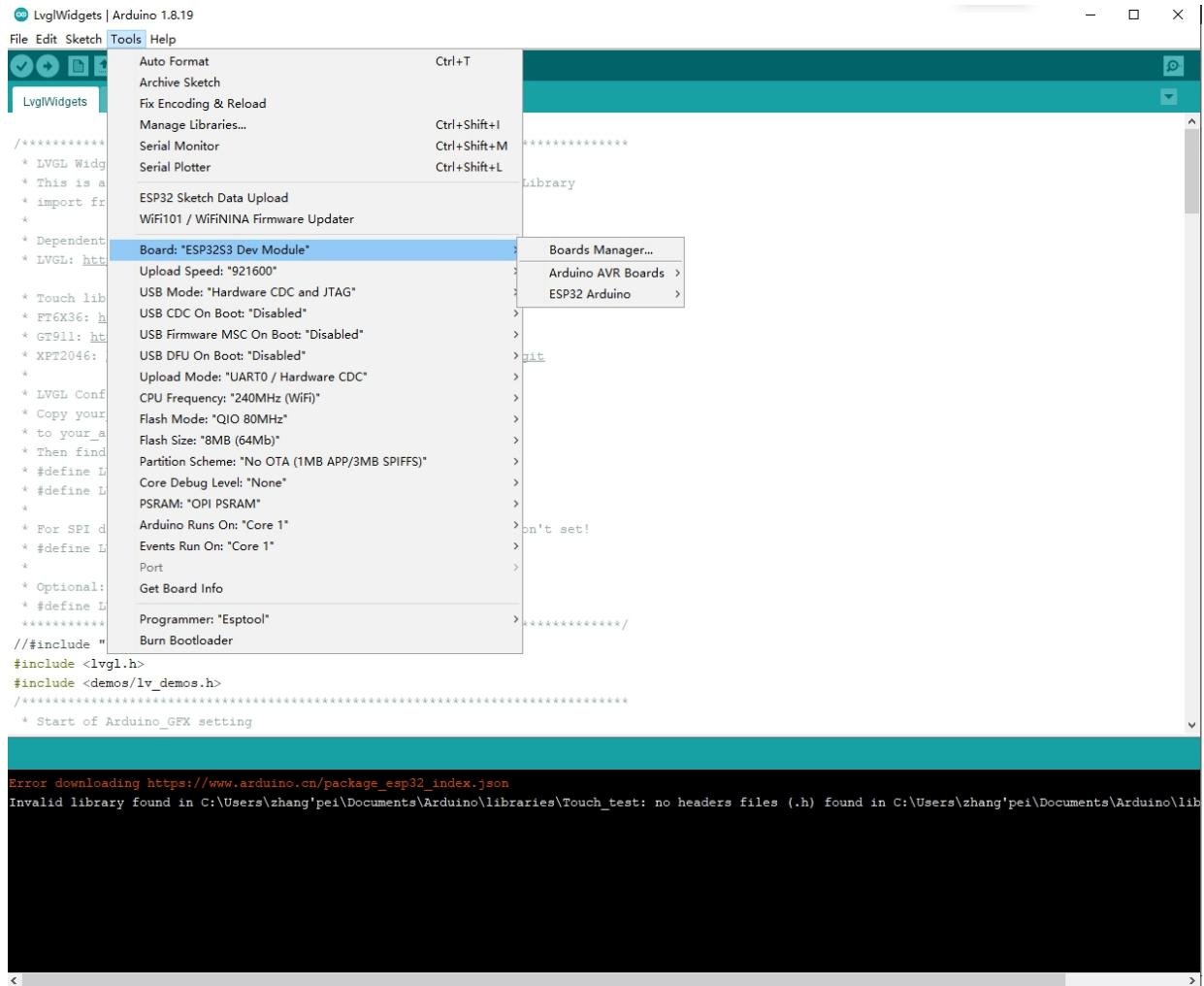
[https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package\\_esp32\\_dev\\_index.json](https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_dev_index.json)

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

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

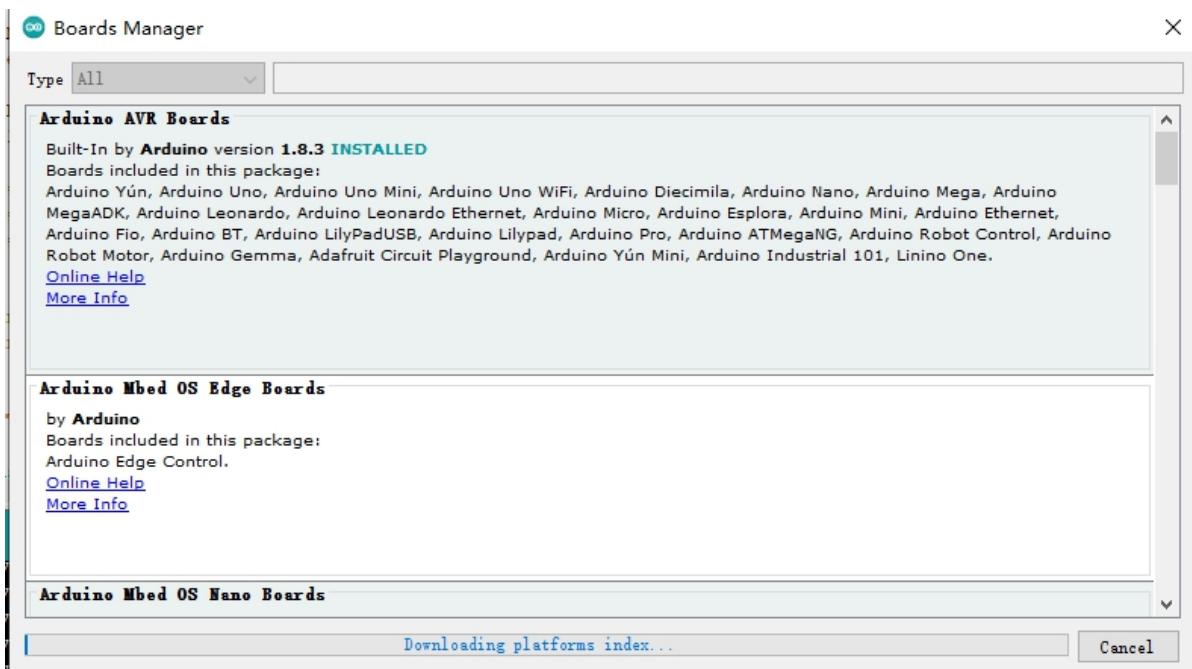


- (9) In the Arduino IDE click on the Tools menu on the top menu bar.
- (10) Scroll down to the Board: entry
- (11) A submenu will open when you highlight the Board: entry.
- (12) At the top of the submenu is Boards Manager. Click on it to open the Boards Manager dialog box.
- (13) In the search box in the Boards Manager enter "esp32".

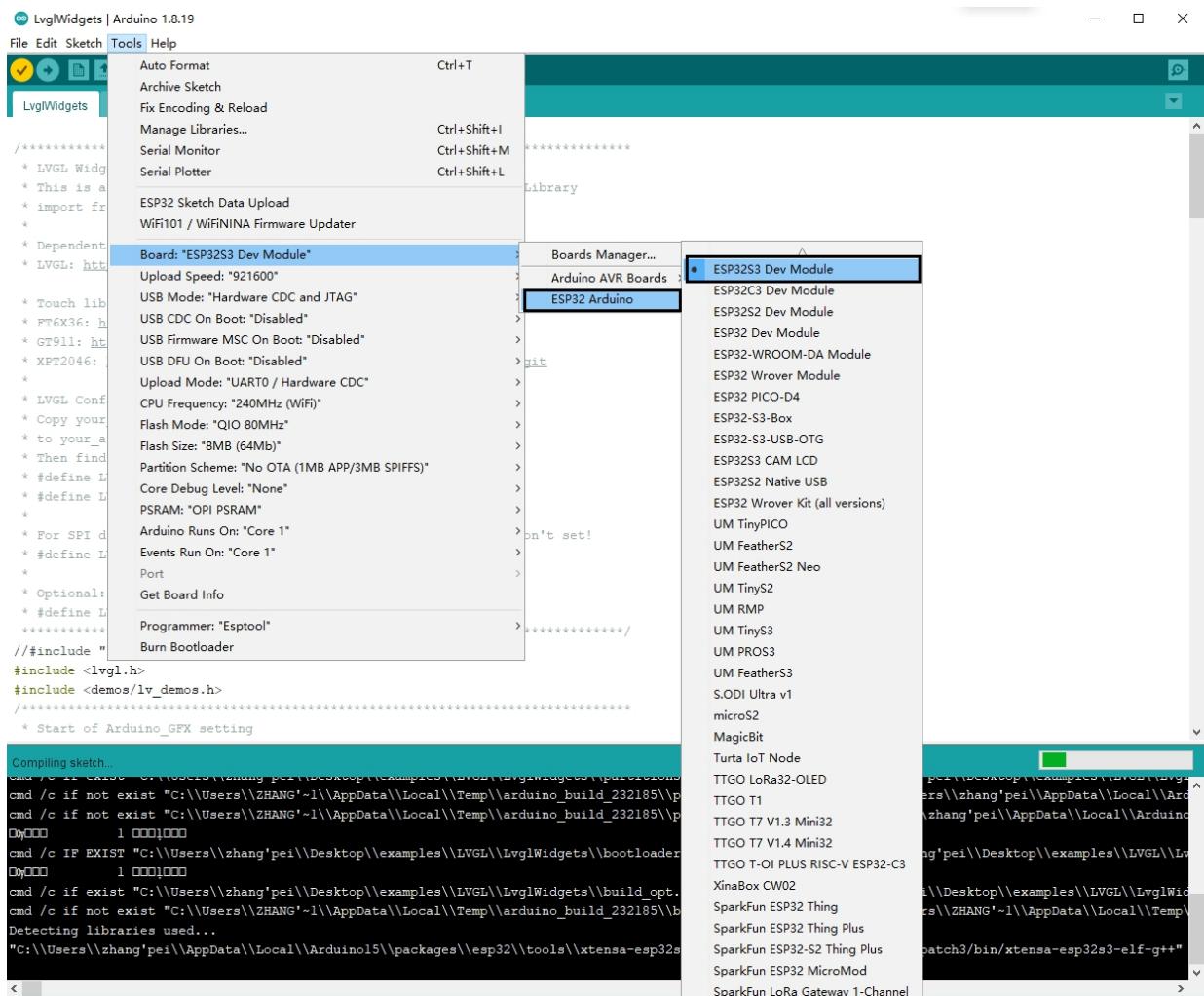


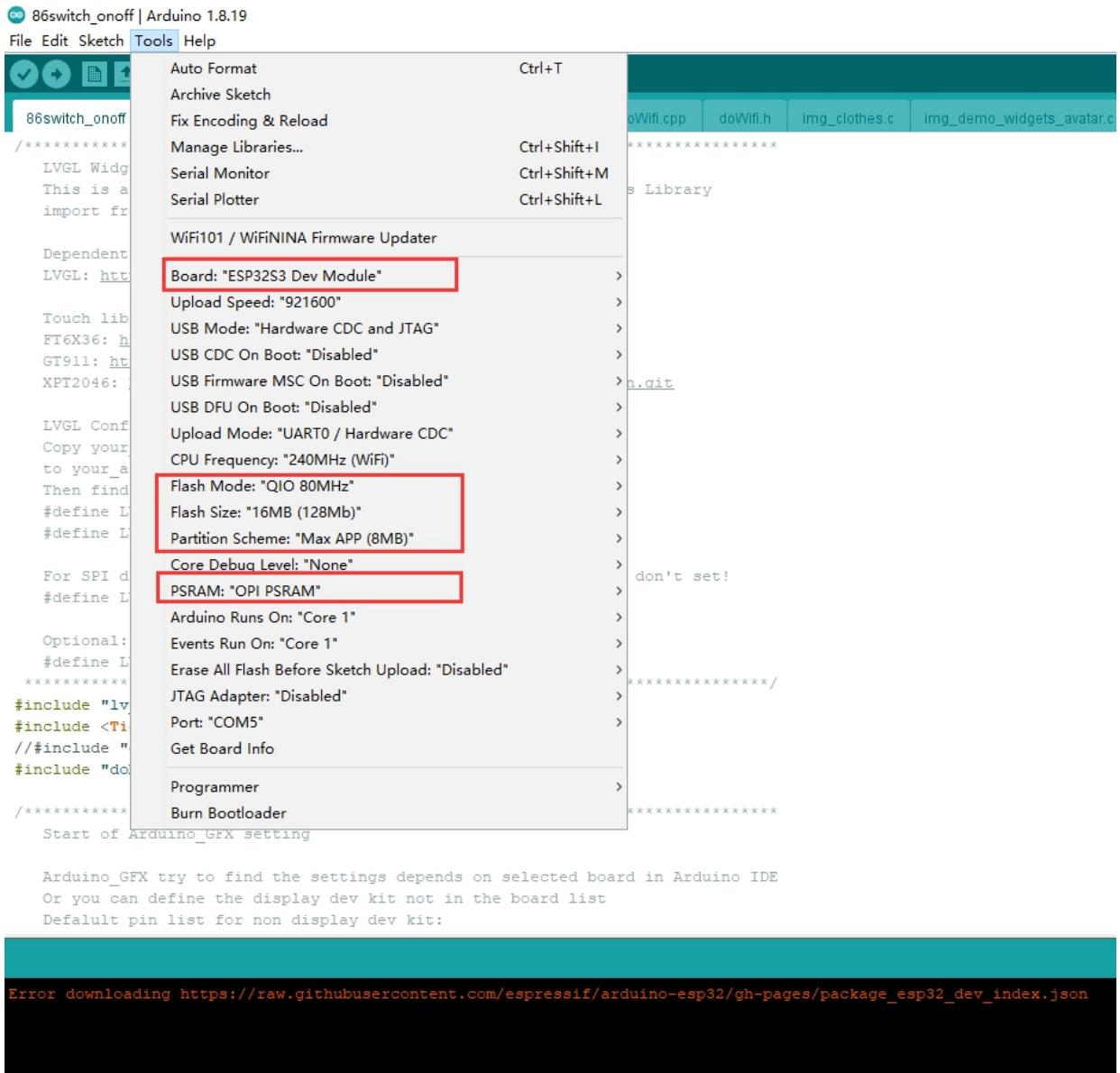
(14) You should see an entry for “esp32 by Espressif Systems”. Highlight this entry and click on the Install button.

This will install the ESP32 boards into your Arduino IDE

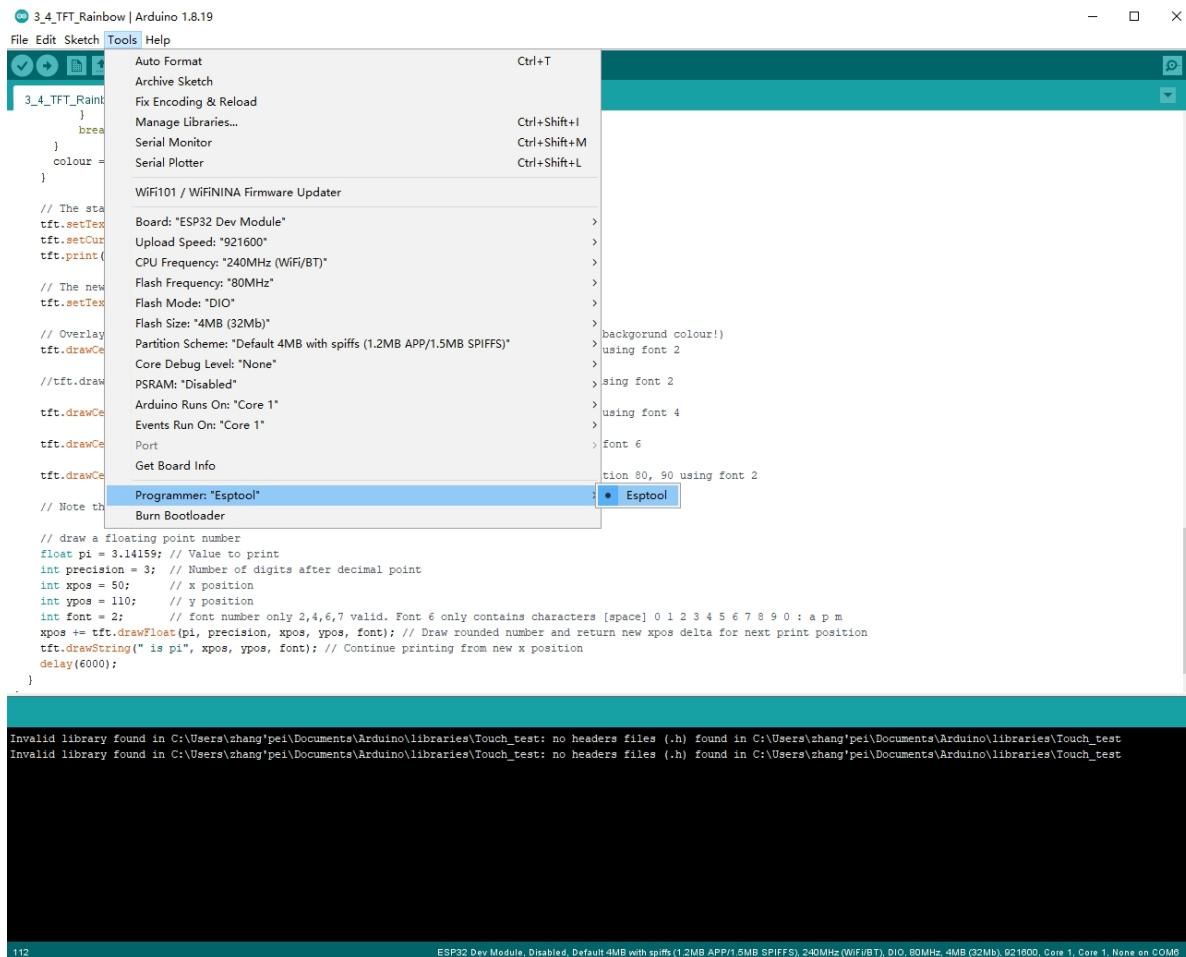


Once the installation completes, we need to select the correct board options for the "ESP32 Arduino" board. In the board type, in the tools tab, we choose "ESP32S3 Dev Module".





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, we 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 below starts the Serial port and prints "hello from ESP32" every second.

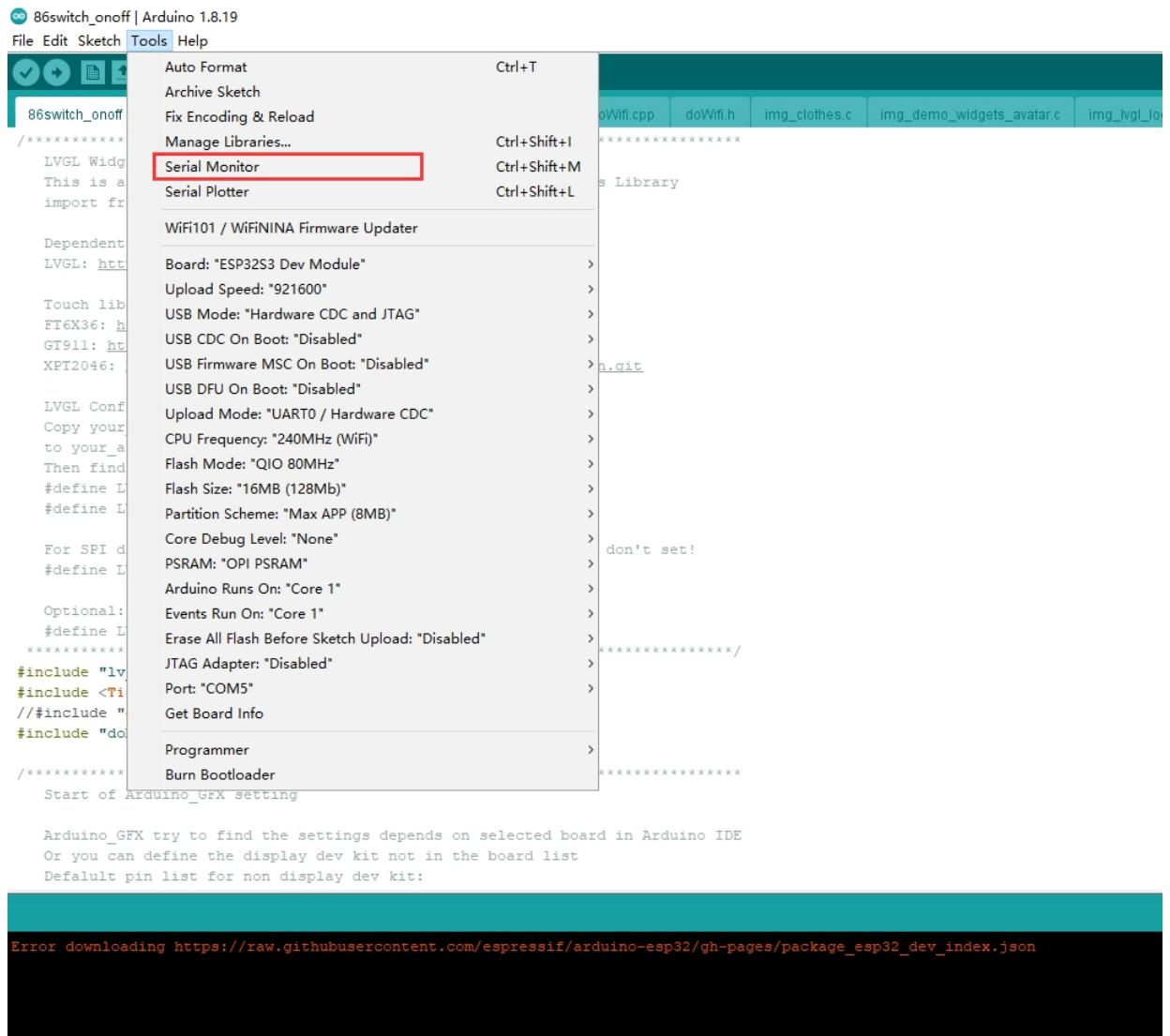
```

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

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

```

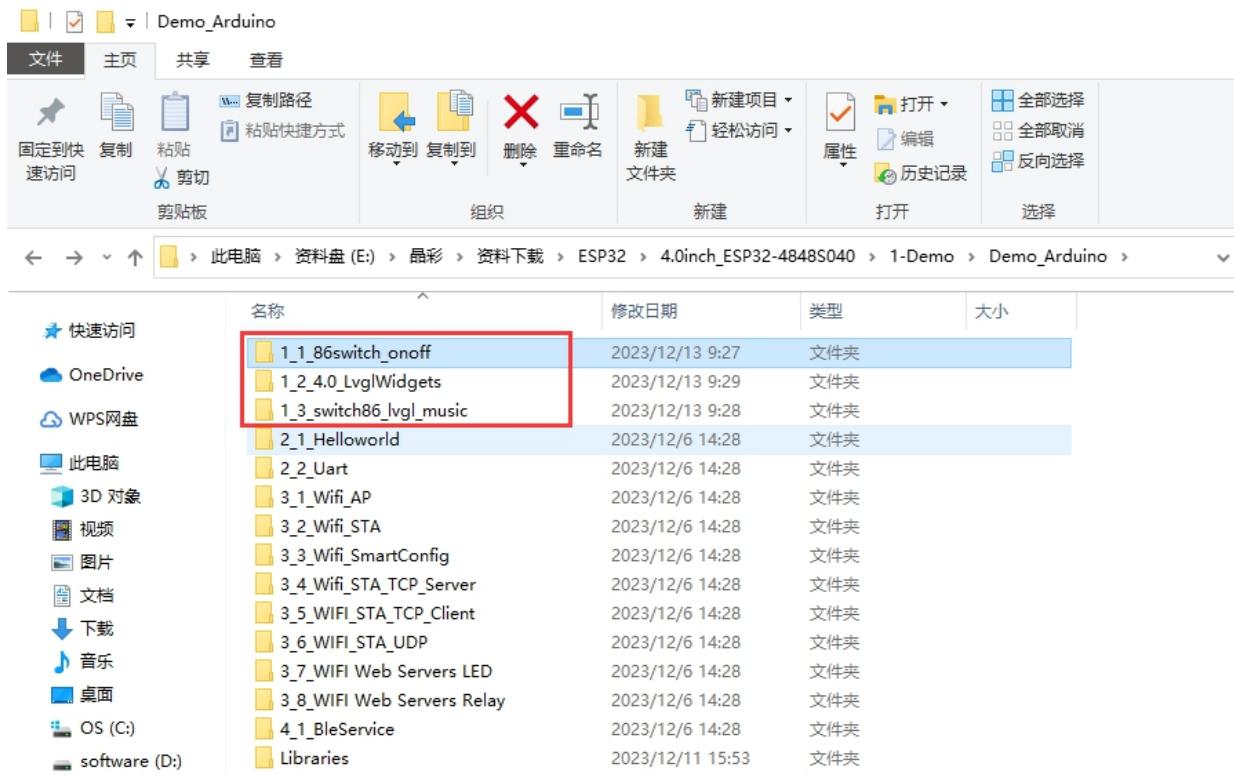
If everything is working fine, we will see the output in the serial console shown.



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

## Sample program usage

At present, only a preliminary explanation and introductory use are given to the samples displayed on the screen, and the corresponding examples in the data center are found, as shown in the figure:



The examples in the red circle are all based on the Arduino\_GFX library as the basic application. This library supports various commonly used driver chips, such as ST7735, ST7789, ILI9341, etc., and has good compatibility.

Arduino\_GFX library file installation:

Open the library manager in Arduino, search for Arduino\_GFX, and click instal .

The screenshot shows the Arduino IDE interface with the following details:

- Title Bar:** LVGL\_Arduino | Arduino 1.8.19
- Menu Bar:** File Edit Sketch Tools Help
- Tools Menu (Open):** Manage Libraries... (Ctrl+Shift+I)
- Sketch Content:** LVGL\_Arduino sketch code. Key parts include:
  - #include <lvgl.h>
  - #include <TFT\_eSPI.h>
  - #include <esp32-hal-wiFi.h>
  - /\*更改屏幕分辨率
  - static const uint16\_t lv\_color = #
  - static lv\_disp\_t \*lv\_disp = lv\_disp\_get\_default();
  - static lv\_colord\_t lv\_colord;
  - TFT\_eSPI tft = TFT\_eSPI();
  - #if LV\_USE\_LOG
  - /\* 日志调试 \*/
  - void my\_print()
  - {
  - Serial.print("LVGL-Arduino example");
  - }
  - #endif
  - //
  - void lv\_examp
  - {
  - /\*要转换的值
  - static lv\_style\_prop\_t props[] = {
  - {LV\_STYLE\_TRANSFORM\_WIDTH, LV\_STYLE\_TRANSFORM\_HEIGHT, LV\_STYLE\_TEXT LETTER\_SPACE},
  - };
  - /\*Transition descriptor when going back to the default state.
  - \*Add some delay to be sure the press transition is visible even if the press was very short\*/
  - static lv\_style\_transition\_dsc\_t transition\_dsc\_def;
  - lv\_style\_transition\_dsc\_init(&transition\_dsc\_def, props, lv\_anim\_path\_overshoot, 250, 100, NULL);
  - /\*Transition descriptor when going to pressed state.
  - \*No delay, go to presses state immediately\*/
- Output Panel:** Shows the upload progress and a warning about an invalid library found in the C:\Users\zhang\Documents\Arduino\libraries\Touch\_test folder.

86switch\_onoff | Arduino 1.8.19  
File Edit Sketch Tools Help

86switch\_onoff WeatherNow.cpp WeatherNow.h doMain.cpp doMain.h doWifi.cpp doWifi.h img\_clothes.c img\_demo\_widgets\_avatar.c img\_lvgl\_logo.c lv\_demo\_wid

```
/*
 * LVGL Widgets
 * This is a widgets demo for LVGL - Light and Versatile Graphics Library
 * import from: https://github.com/lvgl/lv\_demos.git
 *
 * Dependent libraries:
 * LVGL: https://github.com/lvgl/lvgl.git
 *
 * Touch libraries:
 * FT6X36: https://github.com/strange-v/r
 * GT911: https://github.com/TAMCTec/gr91
 * XPT2046: https://github.com/PaulStoffx
 *
 * LVGL Configuration file:
 * Copy your_arduino_path/libraries/lvgl/
 * to your_arduino_path/libraries/lv_conf
 * Then find and set:
 * #define LV_COLOR_DEPTH 16
 * #define LV_TICK_CUSTOM 1
 *
 * For SPI display set color swap can be
 * #define LV_COLOR_16_SWAP 1
 *
 * Optional: Show CPU usage and FPS count
 * #define LV_USE_PERF_MONITOR 1
 */
#include "lv_demo_widgets.h"
#include <Ticker.h>
//#include "demos/lv_demos.h"
#include "doMain.h"

/*
 * Start of Arduino_GFX setting
 */

Arduino_GFX try to find the settings depends on selected board in Arduino IDE
Or you can define the display dev kit not in the board list
Default pin list for non display dev kit:
```

**Library Manager**

Type: All Topic: All Arduino\_GFX

by Marc MERLIN **Arduino\_GFX and FastLED compatible library for ArduinoOnPC X11 TFT Emulator** Designed to work with <https://github.com/marcmerlin/ArduinoOnPc-FastLED-GFX-LEDMatrix>

More info **Install**

**GFX Library for Arduino**

by Moon On Our Nation **Version 1.2.9 INSTALLED**  
**Arduino\_GFX is a GFX library for various color displays with various data bus interfaces** Arduino\_GFX is a Arduino graphics library. Currently support GC9A01 round display, GC9106, GC9107, HX8347C, HX8347D, HX8352C, HX8357A, HX8357B, HX8369A, ILI6122, ILI9225, ILI9331, ILI9341, ILI9342(MSStack, ESP32-S3-BOX), ILI9481, ILI9486, ILI9488, ILI9806, JET6K71, NT35310, NT35510, NT39125, NV3041A, OTM8009A, R61529, RM67162, SEPS325, SSD1283A, SSD1331, SSD1351, ST7735, ST7789, ST7796 and virtually all Raspberry Pi DPI (RGB) display. Tested RGB display: GC9503V, ILI6485, ST7262, ST7701. Currently support software SPI (8-bit and 9-bit), hardware SPI (8-bit, ESP32 also support 9-bit), 8-bit parallel interface(AVR, ESP32, RPi Pico, RTL8720, STM32), 16-bit parallel interface(ESP32 and RPi Pico) and RGB Panel

**GFX4d**

by 4D Systems Pty Ltd **Graphics Library for the gen4-IoD by 4D Systems** This is a library which enables graphics to be easily added to the gen4-IoD modules using the Arduino IDE or Workshop4 IDE. gen4-IoD is powered by the ESP8266.

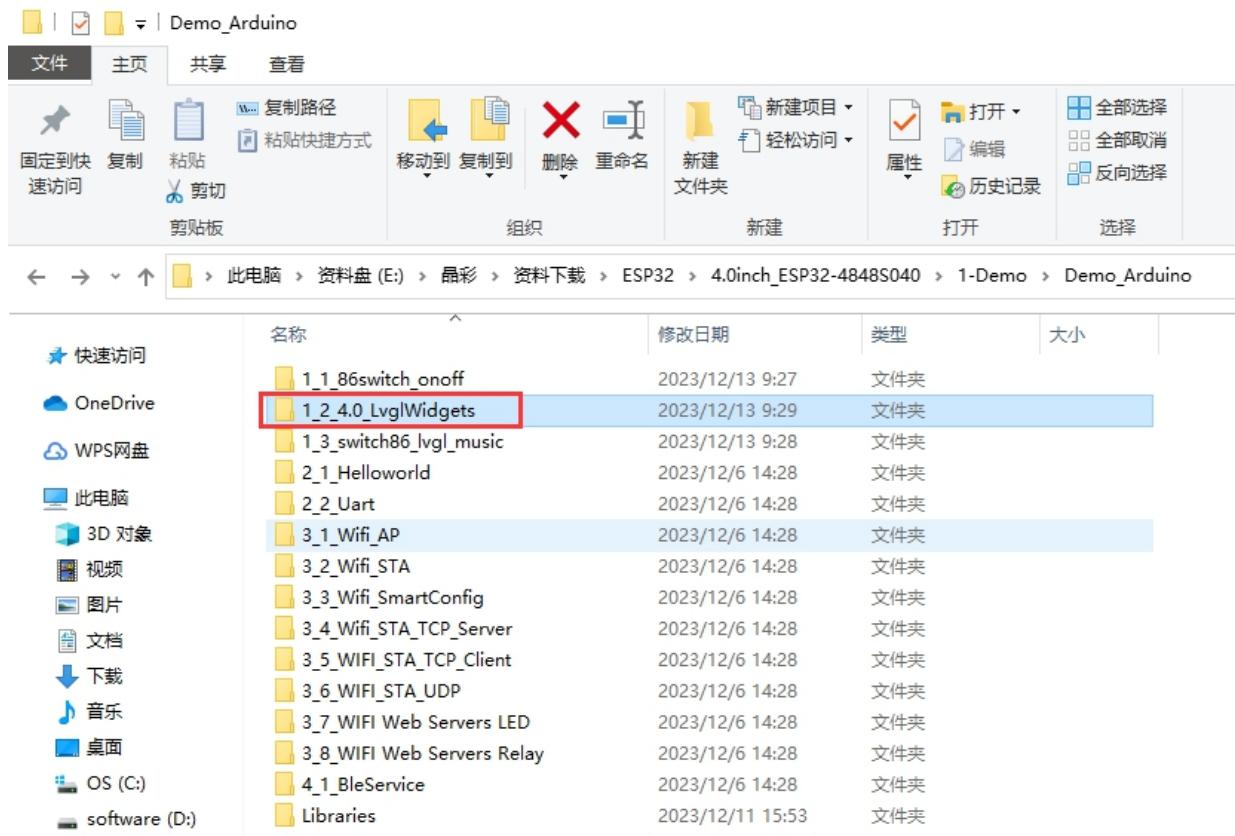
**Error downloading [https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package\\_esp32\\_dev\\_index.json](https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_dev_index.json)**

Although the Arduino\_GFX library has many advantages, it may also have a troublesome place for ordinary users, that is, after the installation

### About the use of touch and LVGL:

Find the data center 1\_2\_4.0\_LvglWidgets

As shown:



Download two library files .

One -Arduino\_GFX library

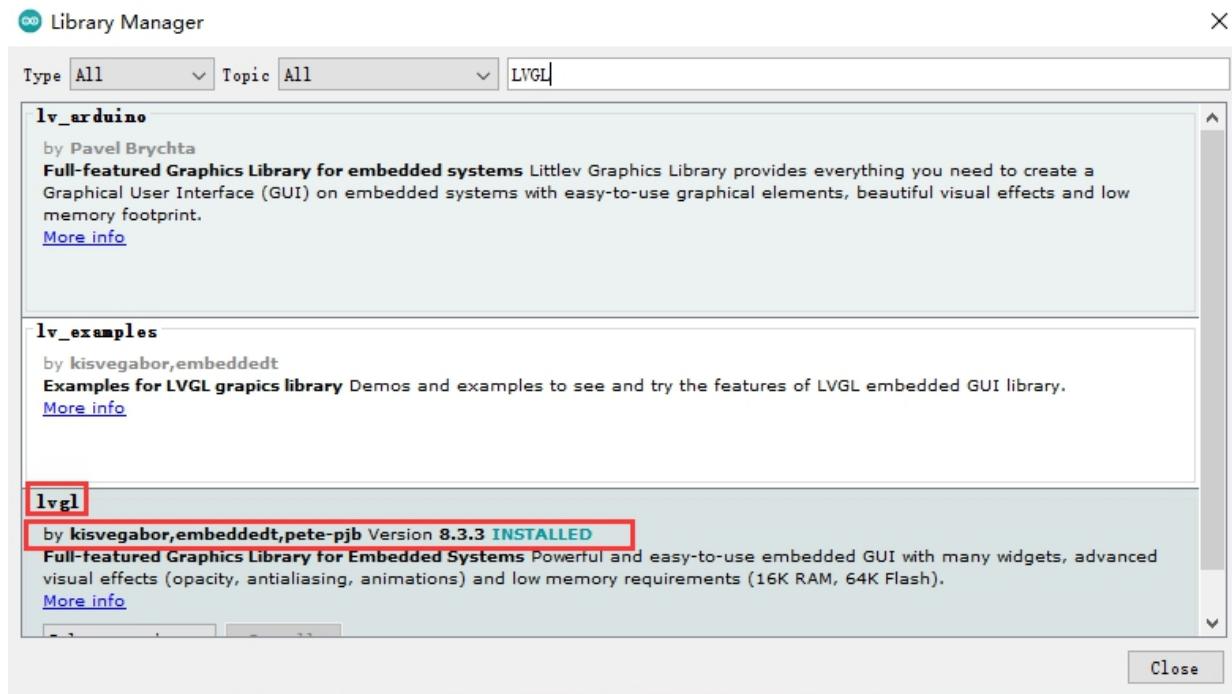
by Marc MERLIN  
Adafruit\_GFX and FastLED compatible library for ArduinoOnPC X11 TFT Emulator Designed to work with <https://github.com/marcm Merlin/ArduinoOnPc-FastLED-GFX-LEDMatrix>

More info Install

**GFX Library for Arduino**  
by Moon On Our Nation Version 1.2.9 INSTALLED  
Arduino\_GFX is a GFX library for various color displays with various data bus interfaces Arduino\_GFX is a Arduino graphics library. Currently support GC9A01 round display, GC9106, GC9107, HX8347C, HX8347D, HX8352C, HX8357A, HX8357B, HX8369A, ILI6122, ILI9225, ILI9331, ILI9341, ILI9342(M5Stack, ESP32-S3-BOX), ILI9481, ILI9486, ILI9488, ILI9806, JBT6K71, NT35310, NT35510, NT39125, NV3041A, OTM8009A, R61529, RM67162, SEPS525, SSD1283A, SSD1331, SSD1351, ST7735, ST7789, ST7796 and virtually all Raspberry Pi SPI (RGB) display. Tested RGB display: GC9503V, ILI6485, ST7262, ST7701. Currently support software SPI (8-bit and 9-bit), hardware SPI (8-bit, ESP32 also support 9-bit), 8-bit parallel interface(AVR, ESP32, RPi Pico, RTL8720, STM32), 16-bit parallel interface(ESP32 and RPi Pico) and RGB Panel

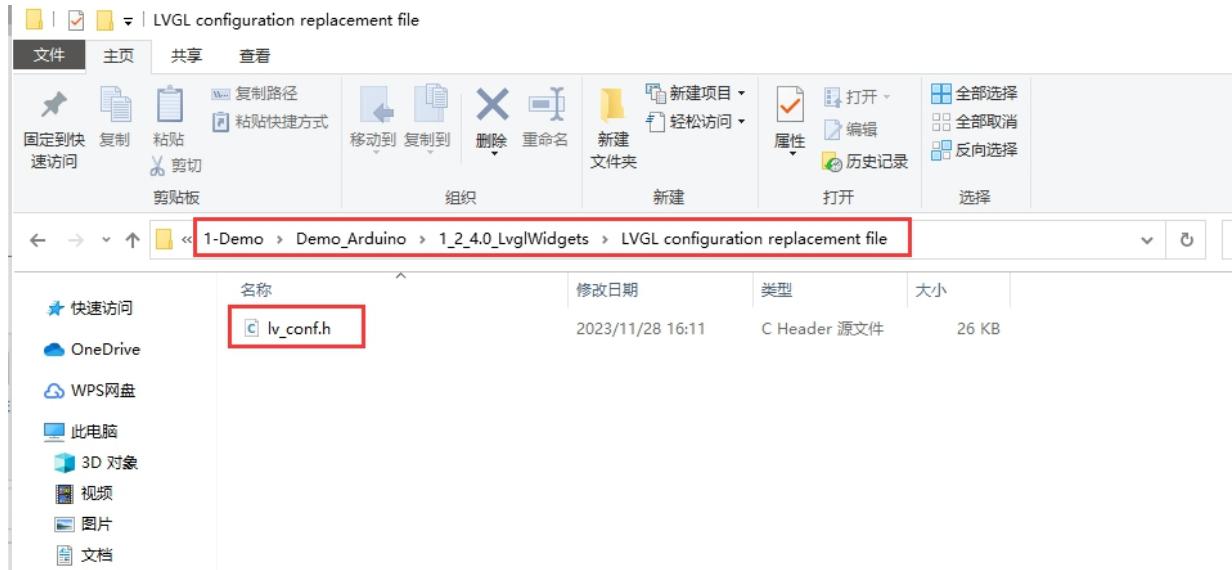
**GFX4d**  
by 4D Systems Pty Ltd  
Graphics Library for the gen4-IoD by 4D Systems This is a library which enables graphics to be easily added to the gen4-IoD modules using the Arduino IDE or Workshop4 IDE. gen4-IoD is powered by the ESP8266.

Two -Lvgl



Copy the lv\_conf.h of the data center .

As shown:



Put this file under the arduino library file, it must be in the same root directory as the library TFT\_eSPI .

As shown:

此电脑 > OS (C:) > 用户 > zhang'pei > 文档 > Arduino > libraries

在 libraries 中搜索

名称	修改日期	类型	大小
Adafruit_CCS811	2022/6/27 12:06	文件夹	
Adafruit_Unified_Sensor	2022/6/27 12:06	文件夹	
ArduinoJson	2022/7/6 9:23	文件夹	
AsyncTCP	2022/6/27 12:06	文件夹	
Audio	2022/6/28 17:44	文件夹	
DallasTemperature	2022/6/27 12:06	文件夹	
DHT_sensor_library	2022/6/27 12:06	文件夹	
DHT_sensor_library_for_ESPx	2022/6/25 10:23	文件夹	
ESP32Servo	2022/6/27 12:06	文件夹	
ESPAsyncWebServer	2022/6/27 12:06	文件夹	
FastLED	2022/7/6 9:23	文件夹	
GFX_Library_for_Arduino	2022/8/9 18:08	文件夹	
gt911-arduino-main	2022/8/17 10:21	文件夹	
GT911-master	2022/8/15 15:10	文件夹	
IRremote	2022/6/27 12:06	文件夹	
JPEGDecoder	2022/6/28 13:49	文件夹	
LiquidCrystal_I2C	2022/6/27 12:06	文件夹	
LovyanGFX	2022/7/31 14:05	文件夹	
lvgl	2022/3/4 10:31	文件夹	
MFRC522	2022/6/27 12:06	文件夹	
OneWire	2022/6/27 12:06	文件夹	
PNGdec	2022/6/28 10:48	文件夹	
Rtc_by_Makuna	2022/6/27 12:06	文件夹	
TFT_eSPI	2022/8/16 12:46	文件夹	
TFT_Touch-master	2022/8/1 12:16	文件夹	
Time	2022/7/6 9:23	文件夹	
TJpg_Decoder	2022/8/3 14:25	文件夹	
Touch_test	2022/8/1 12:12	文件夹	
TP_Arduino_DigitalRain_Anim-main	2022/7/31 13:13	文件夹	
XPT2046_Touchscreen	2022/7/17 18:09	文件夹	
XT_DAC_Audio	2022/7/2 17:12	文件夹	
lv_arduino.rar	2022/7/21 14:20	360压缩 RAR 文件	6,740 KB
lv_conf.h	2022/8/19 17:01	C Header 源文件	24 KB
readme.txt	2022/6/15 15:12	文本文档	1 KB

After compiling, you can run LVGL and touch normally.