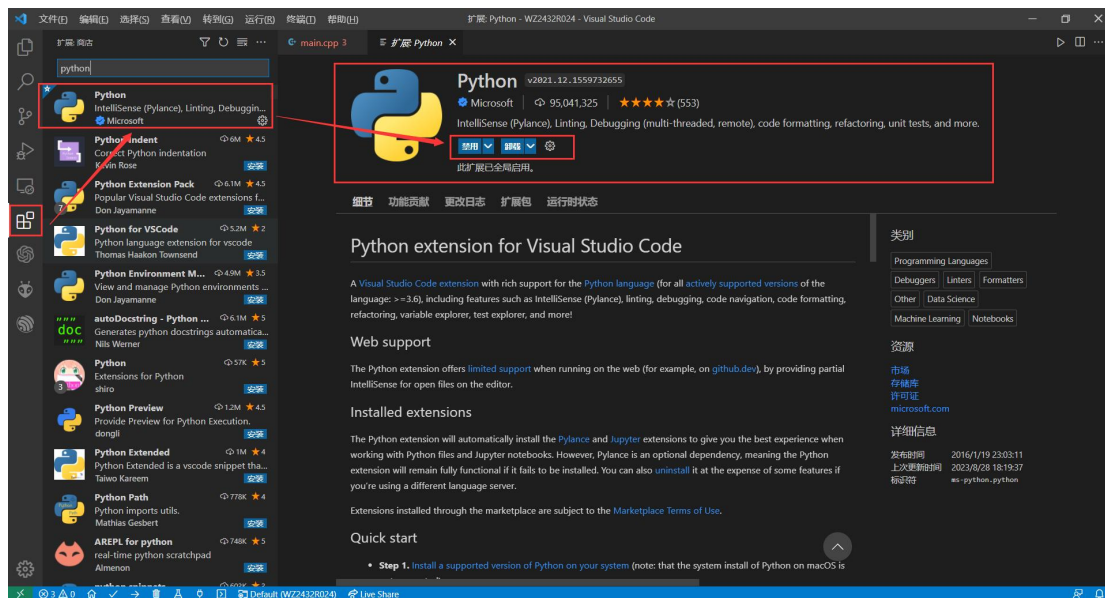


PlatformIO

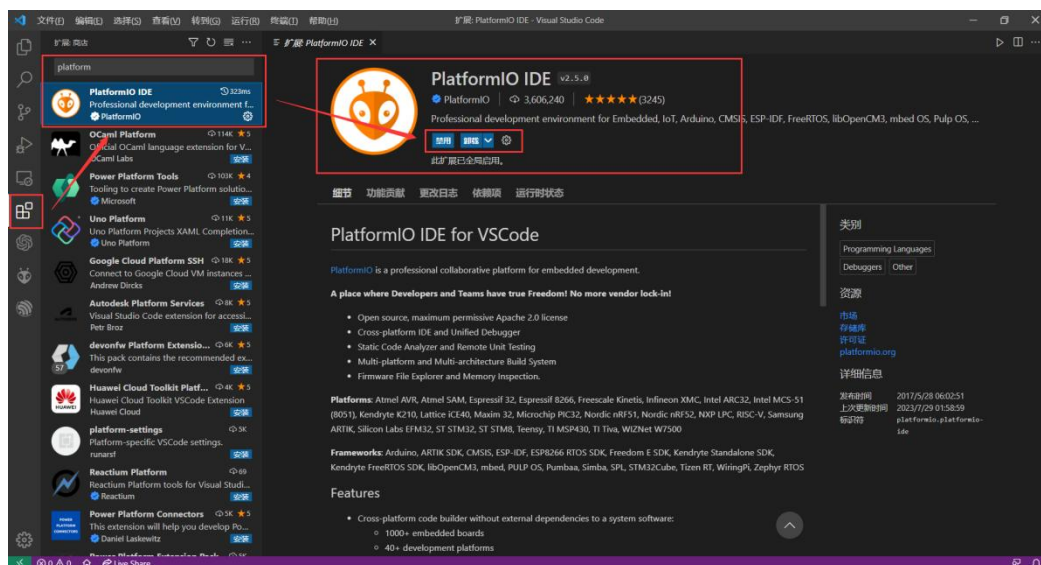
Terminal-spi or Terminal-rgb

Take the Terminal-spi as an example

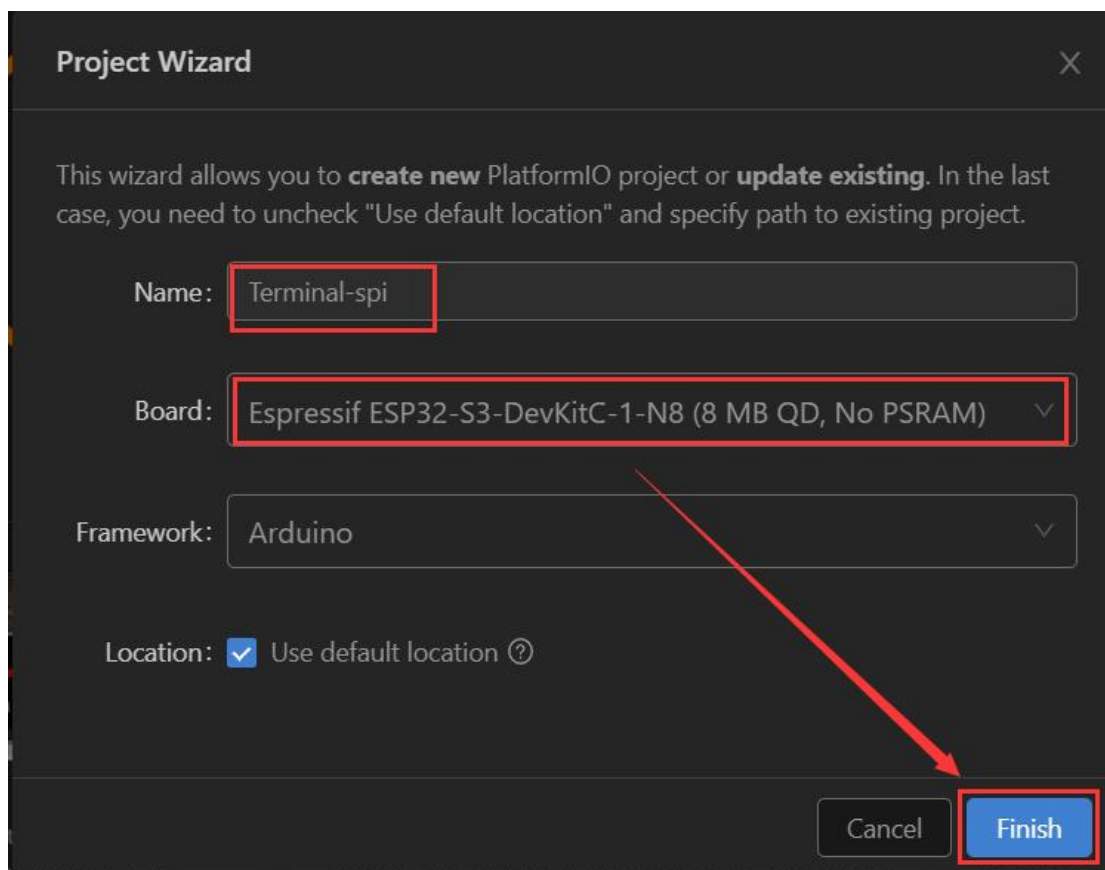
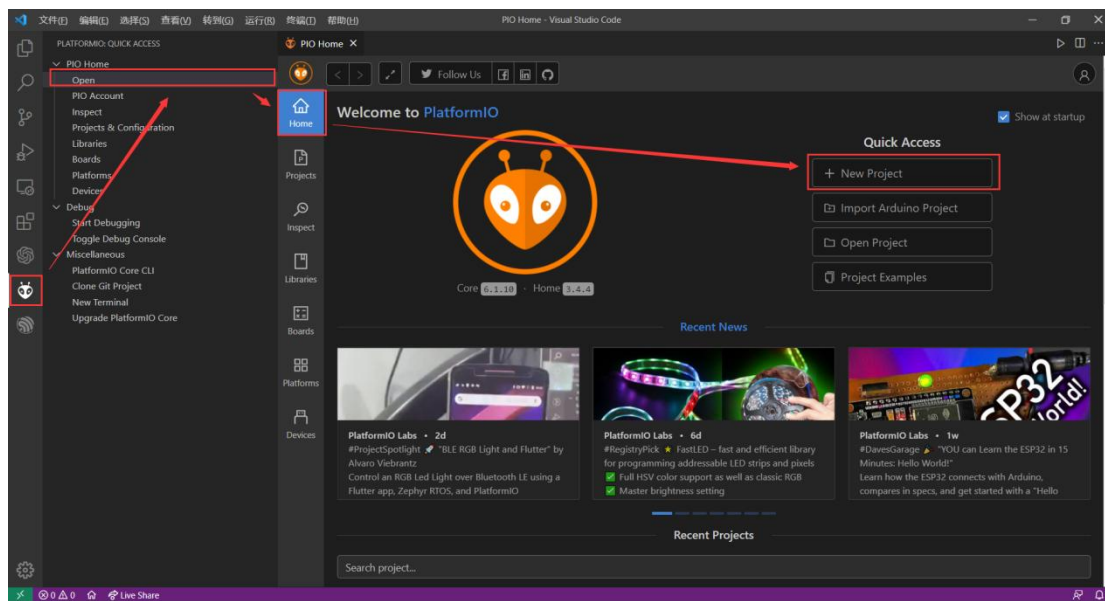
First open the VScode to check if the python is installed

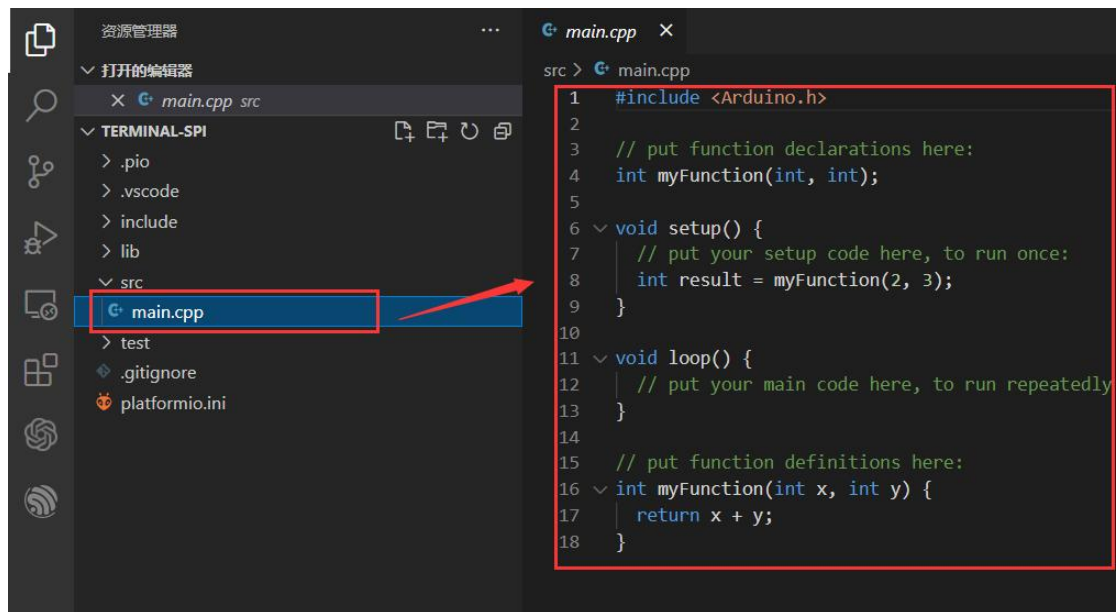


Open the VScode to download the PlatformIO

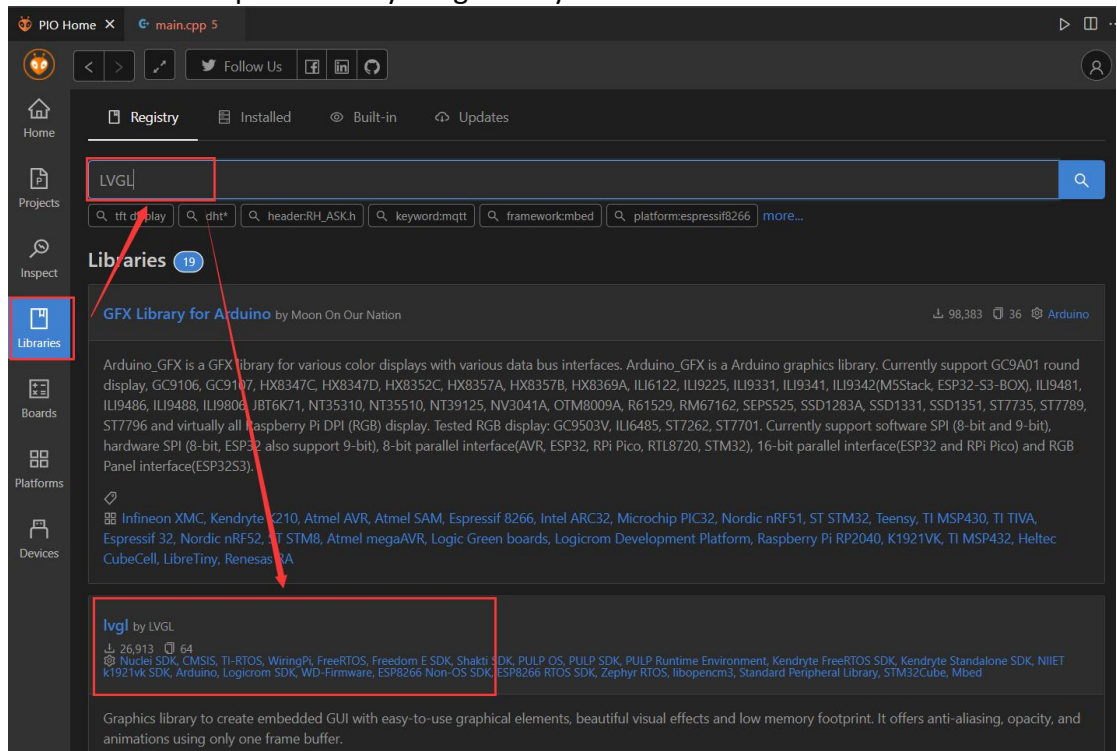


Create new projects





Download the required library (lvgl、LovyanGFX)

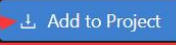


Registry Installed Built-in Updates

lvgl by LVGL

Graphics library to create embedded GUI with easy-to-use graphical elements, beautiful visual effects and low memory footprint, and animations using only one frame buffer.

Installation

8.3.6 released 3 months ago  | [More info](#)

Examples Installation Headers Changelog

lv_example_chart_1

lv_example_chart_1

```
#include "../lv_examples.h"
#if LV_USE_CHART && LV_BUILD_EXAMPLES

void lv_example_chart_1(void)
{
    /*Create a chart*/
    lv_obj_t * chart;
    chart = lv_chart_create(lv_scr_act());
```

Add project dependency

lvgl/lvgl@8.3.6

Projects\Terminal-spi

You can manage your projects in the "Projects" section: create a new or add existing.

Information

- > Registry and Specification
- > External resources

Cancel Add

Home

Projects

Inspect

Libraries

Boards

Platforms

Devices

Registry

Installed

Built-in

Updates

LovyanGFX

by lovyan03

TFT LCD Graphics driver with touch for ESP32, ESP8266, SAMD21, SAMD51, RP2040

1.1.8 released 26 days ago

Add to Project

More info

Examples

Installation

Headers

Changelog

RotateDial

RotateDial

```
#define LGFX_USE_V1
#include <LovyanGFX.hpp>

static LGFX lcd;
static LGFX_Sprite sp;
static LGFX_Sprite sprites[2];
static bool flip;
static int sprite_height;

inline uint16_t getBackColor(int x, int y)
{
    return lcd.swap565(abs((x&31)-16)<<3, 0, abs((y&31)-16)<<3);
    //return lcd.swap565(x, 0, y);
}

void setup(void)
```

Add project dependency

X

lovyan03/LovyanGFX@^1.1.8

Projects\Terminal-spi

You can manage your projects in the "Projects" section: create a new or add existing.

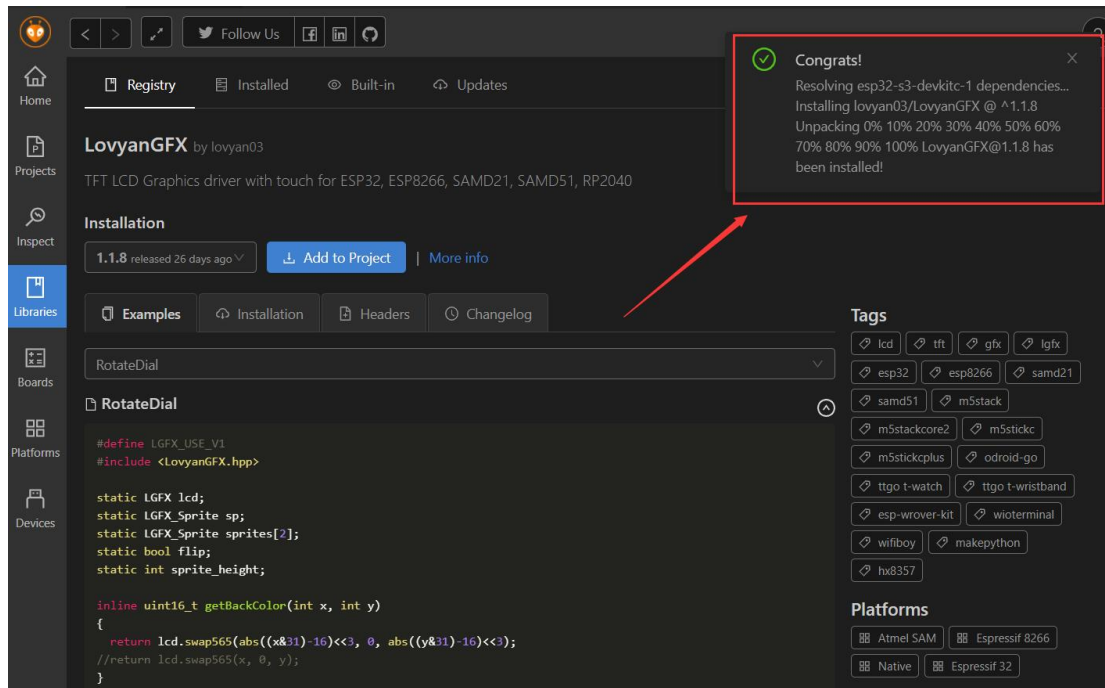
Information

> Registry and Specification

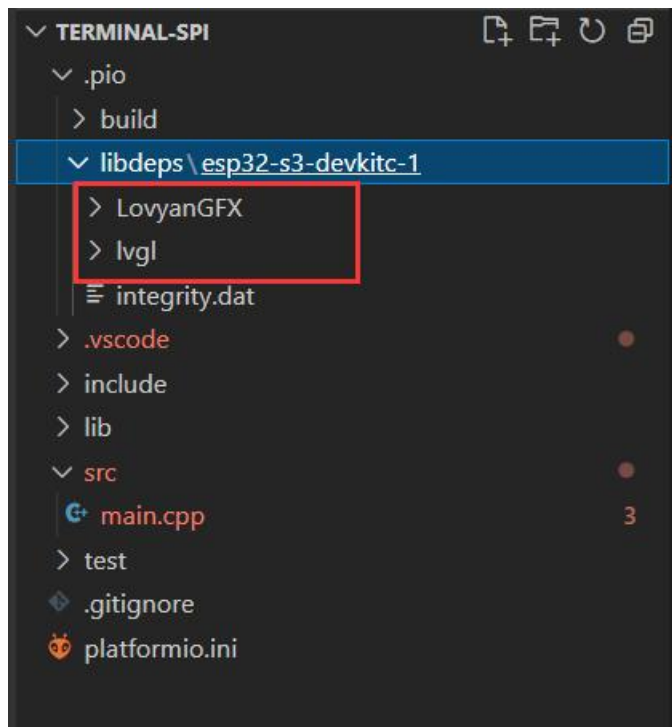
> External resources

Cancel

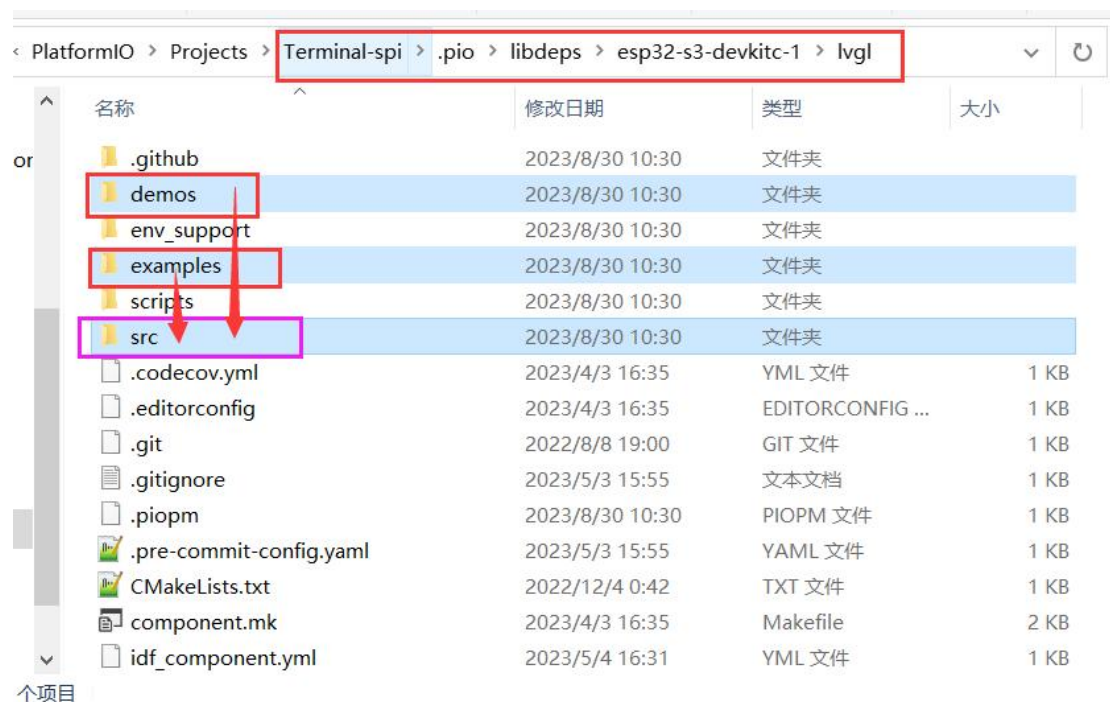
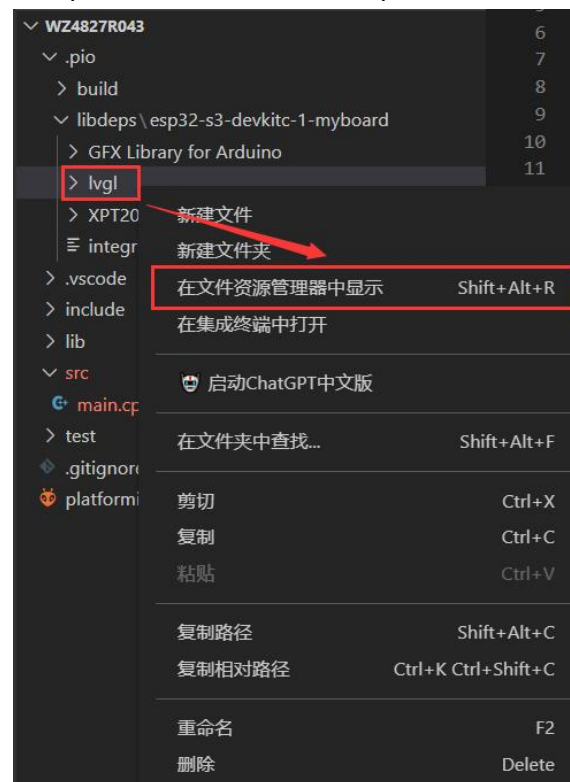
Add

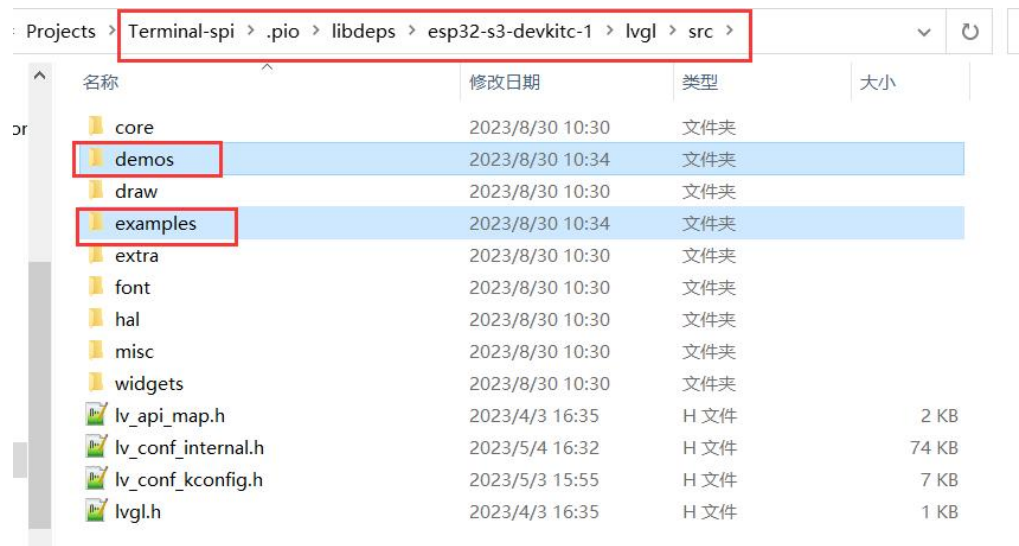


We can see that the library has been added successfully!

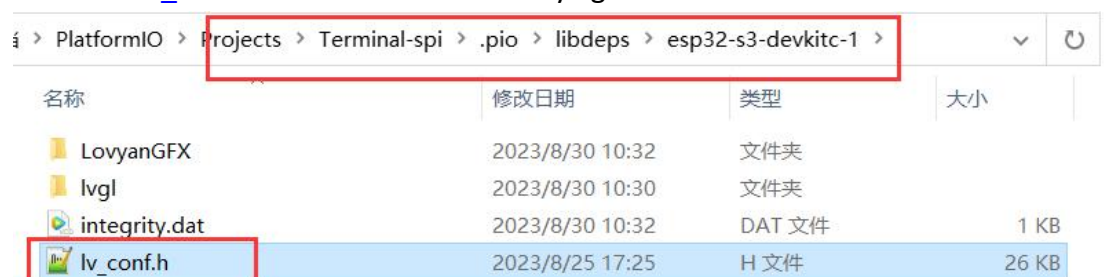


Next, we want to configure the lvgl library, right-click to open the folder directory, and put the demo and examples folders into the src folder!

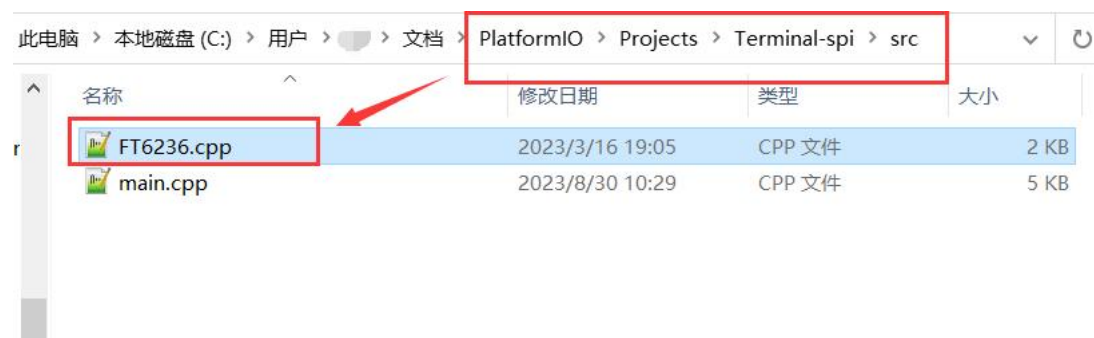




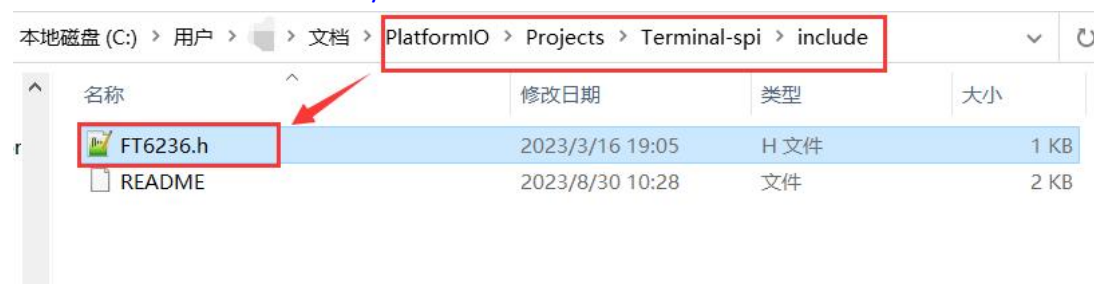
Place the [lv_conf.h](#) file under this directory again



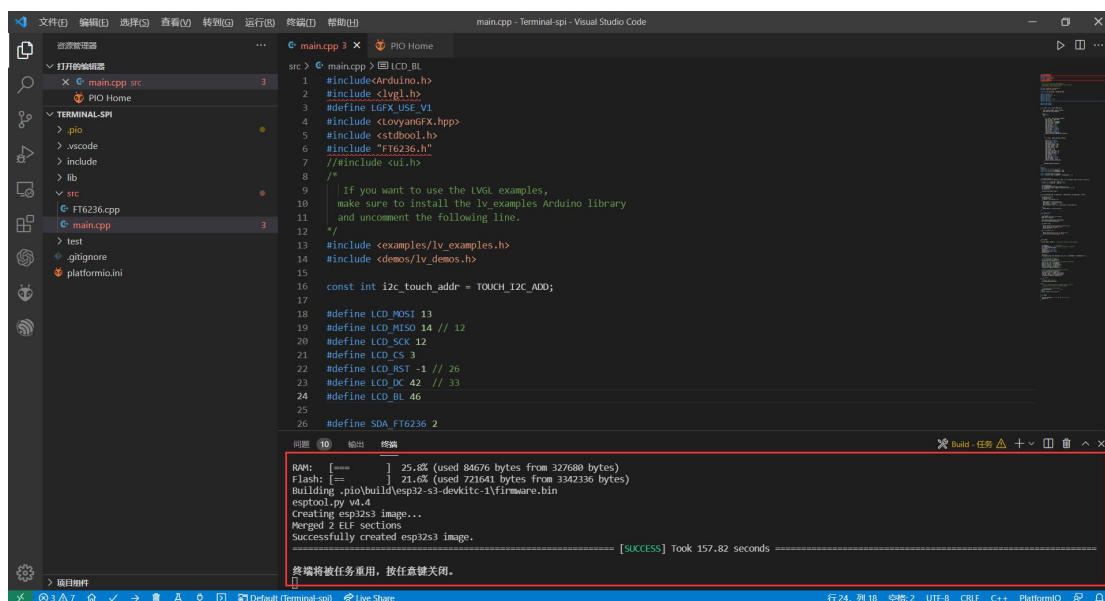
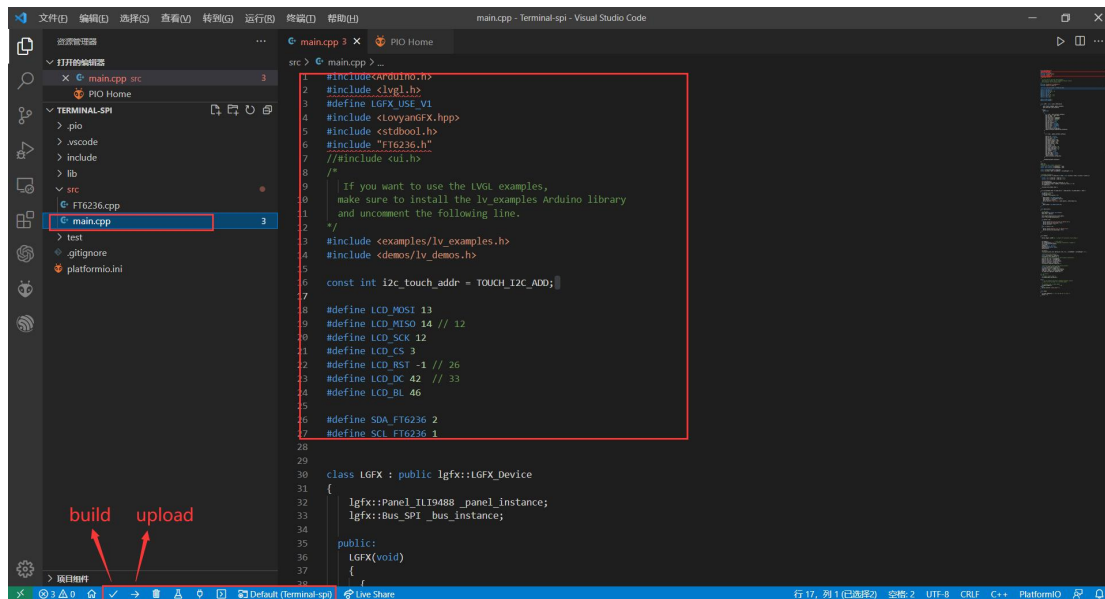
Put the [FT6236.cpp](#) file into the `/src` folder



Put the [FT6236.h](#) file in the `/include` folder



At this time, we will complete all the configuration, write the code and start compiling the program



Next we began to burn the program, finished!

The screenshot shows the Visual Studio Code interface with the 'main.cpp' file open. The code is for an FT6236 microcontroller and includes headers for LCD, LVGL, and LGFX. The terminal window at the bottom shows the output of the burning process, which was successful.

```
src > C:\main.cpp > | LCD_BL
1 #include<Arduino.h>
2 #include <lvgl.h>
3 #define LGFX_USE_V1
4 #include <LovyanGFX.hpp>
5 #include <stdbool.h>
6 #include "FT6236.h"
7 // #include <ui.h>
8 /*
9  * If you want to use the LVGL examples,
10  * make sure to install the lv_examples Arduino library
11  * and uncomment the following line.
12  */
13 #include <examples/lv_examples.h>
14 #include <demos/lv_demos.h>
15
16 const int i2c_touch_addr = TOUCH_I2C_ADDR;
17
18 #define LCD_MOSI 13
19 #define LCD_MISO 14 // 12
20 #define LCD_SCK 12
21 #define LCD_CS 3
22 #define LCD_RST -1 // 26
23 #define LCD_DC 42 // 33
24 #define LCD_BL 46
25
26 #define SOA_FT6236 2
```

Writing at 0x0000ffff... (96 %)
Writing at 0x0000a77b... (100 %)
Wrote 722832 bytes (409116 compressed) at 0x00010000 in 9.5 seconds (effective 606.4 kbit/s)...
Hash of data verified.
Leaving...
Hard resetting via RTS pin...
[SUCCESS] Took 39.24 seconds