Esp32+VSC+Esp-IDF+Lvgl+ST7789详细点亮过程

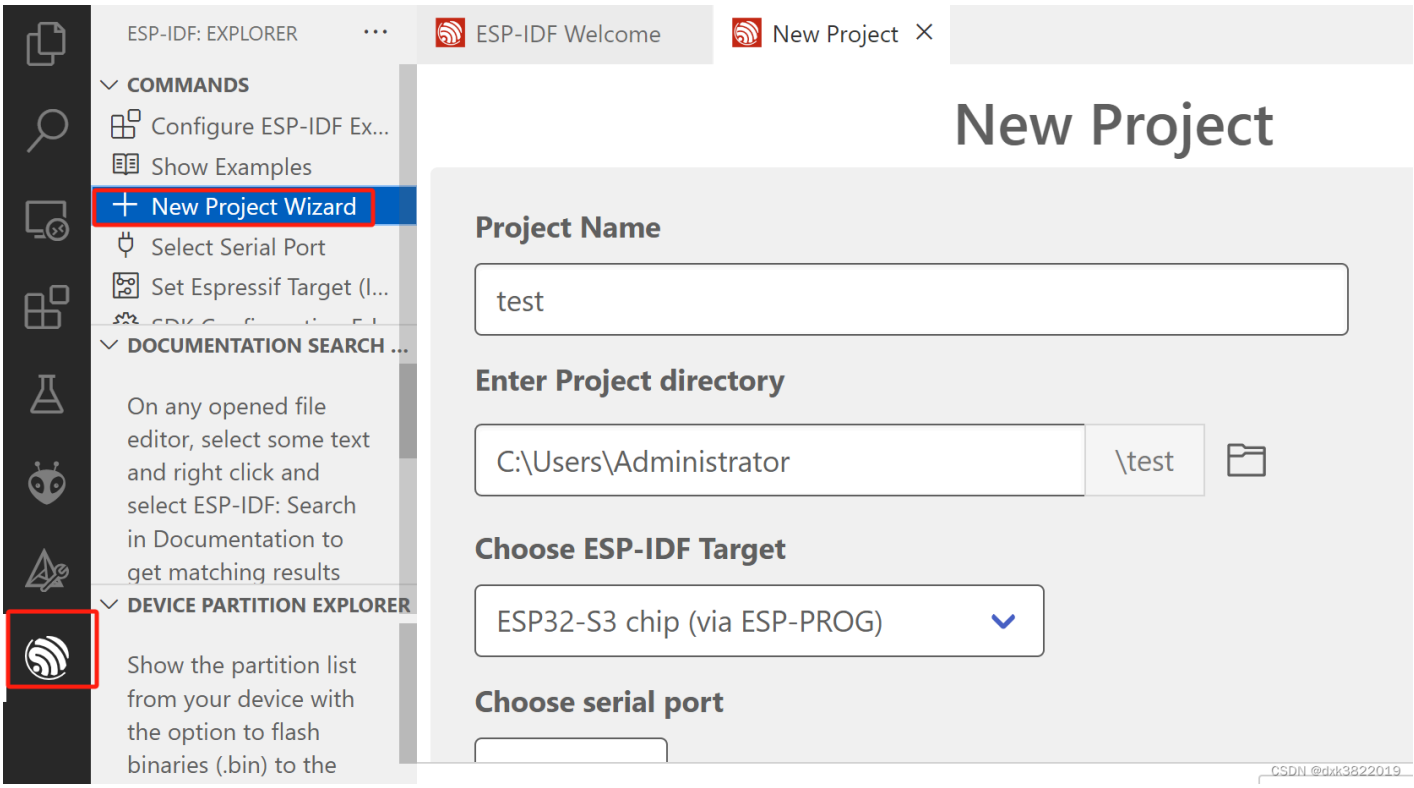
原创 dxk3822019 于 2024-02-21 18:14:36 发布 阅读量1.4k 收藏 23 点赞数 12 版权

分类专栏： ESP32学习 文章标签： 驱动开发 单片机 物联网 开源 mcu

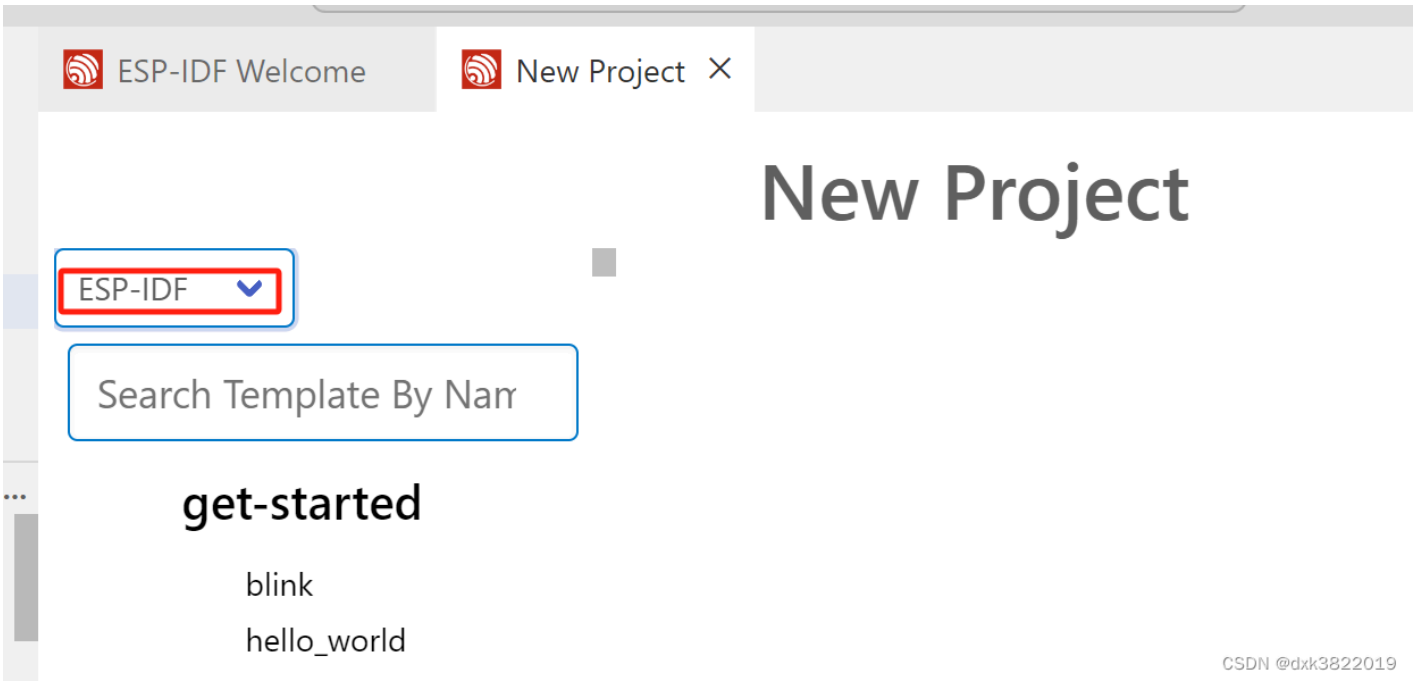
摘要 本文详细指导如何在ESP32项目中将LCD控制器从GC9A01替换为ST7789，包括创建工程、修改配置文件、添加驱动接口和配置LVGL，最终实现编译和下载程序到硬件。

摘要由CSDN通过智能技术生成

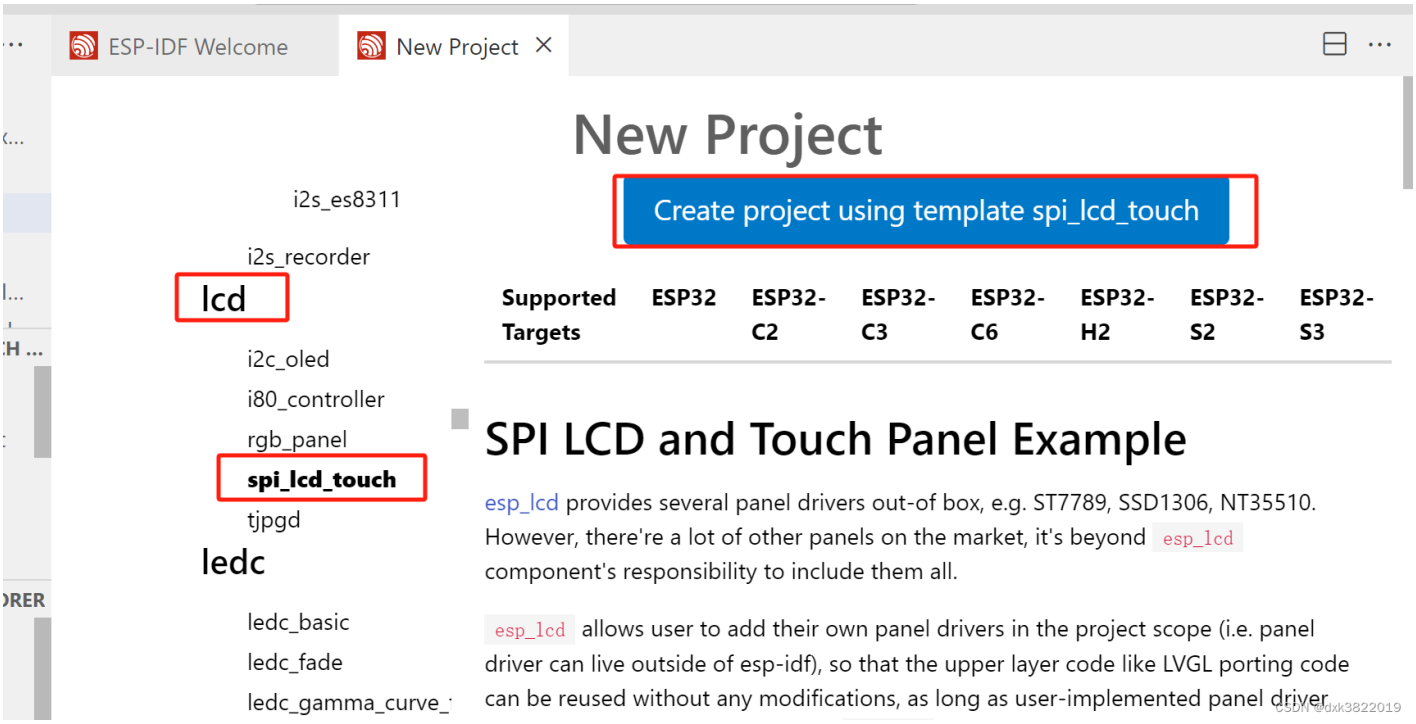
1.先创建一个工程



这里选择ESD-IDF

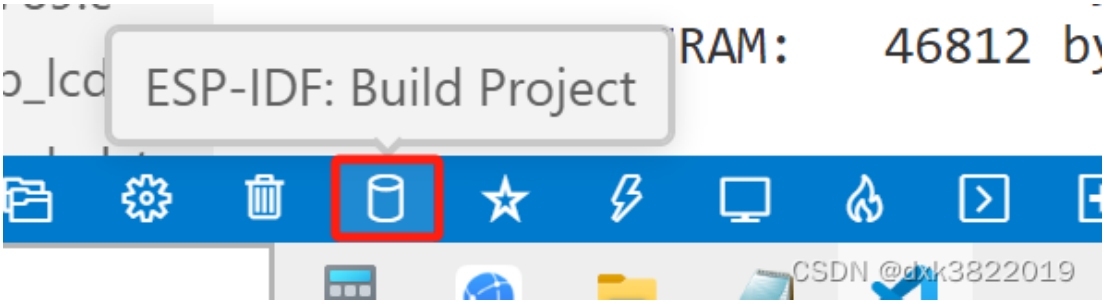


找到LCD ，spi_lcd_touch，点击创建工程。

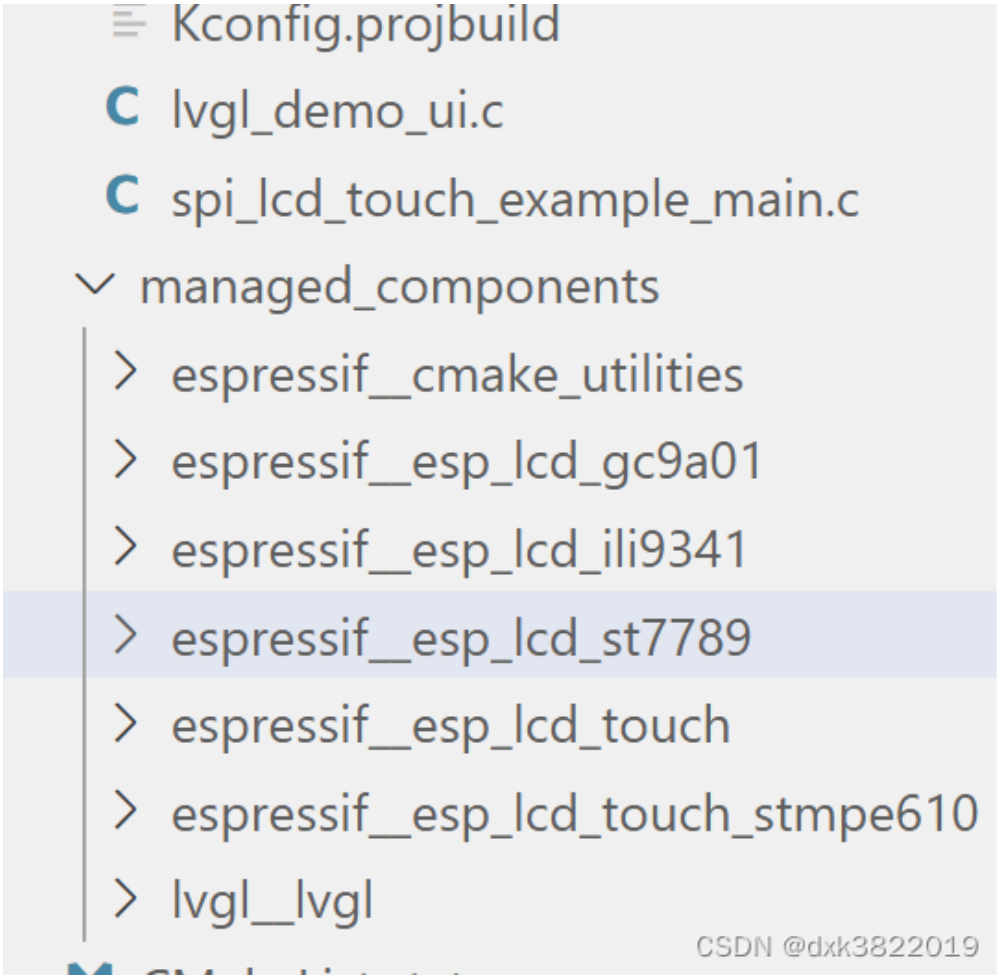


2.添加ST7789 文件夹

先编译一下，编译成功后，工程目录里面会出现一个文件夹，managed_components，

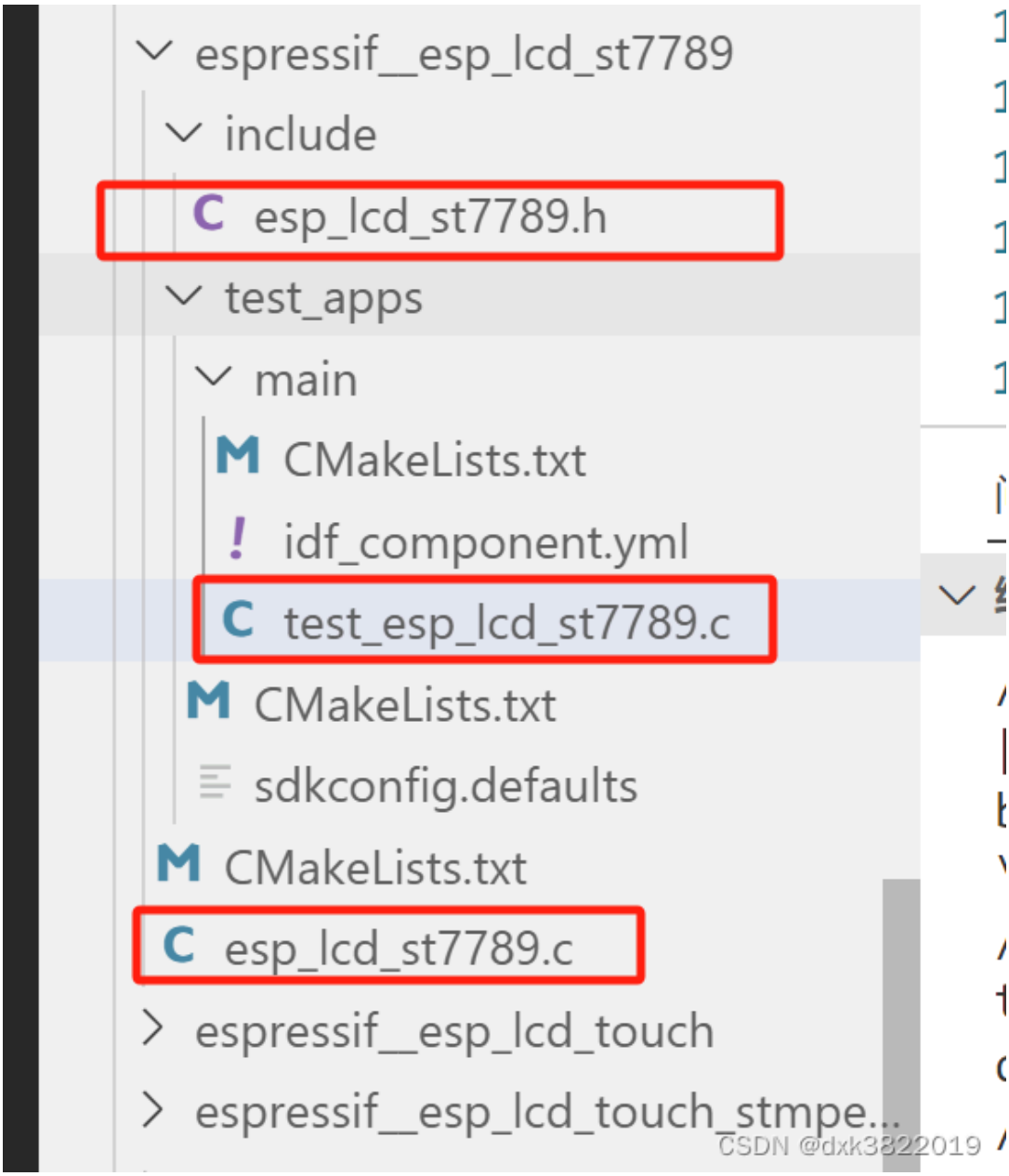


复制espressif_esp_lcd_gc9a01文件夹在同级目录，然后改一下文件夹名称，改成espressif_esp_lcd_st7789。



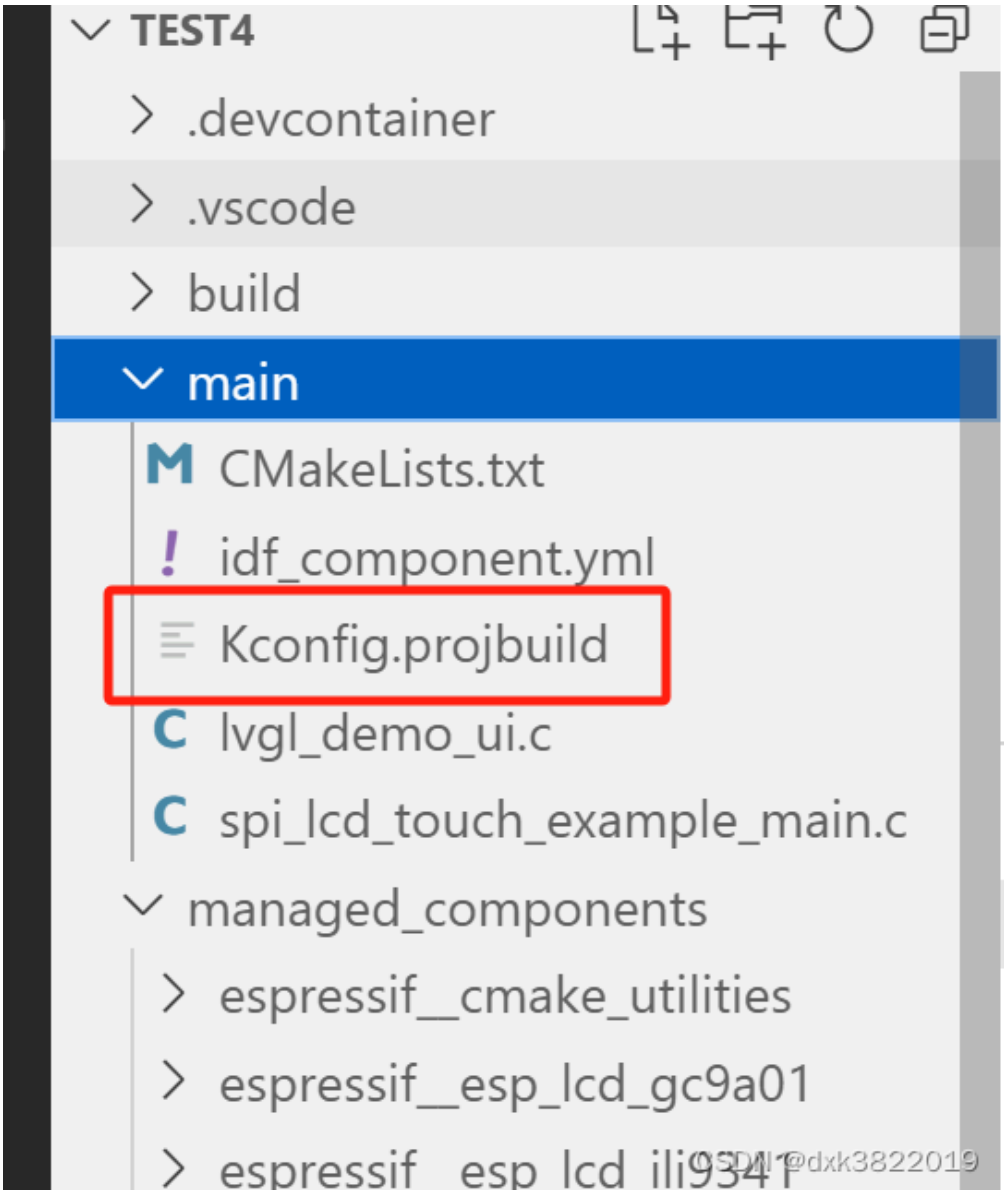
将espressif__esp_lcd_st7789文件夹里面的所有所有C文件和头文件里面的GC9A01全部换成ST7789，gc9a01换成st7789，名称和内容里面的全都换。

然后清理，再编译

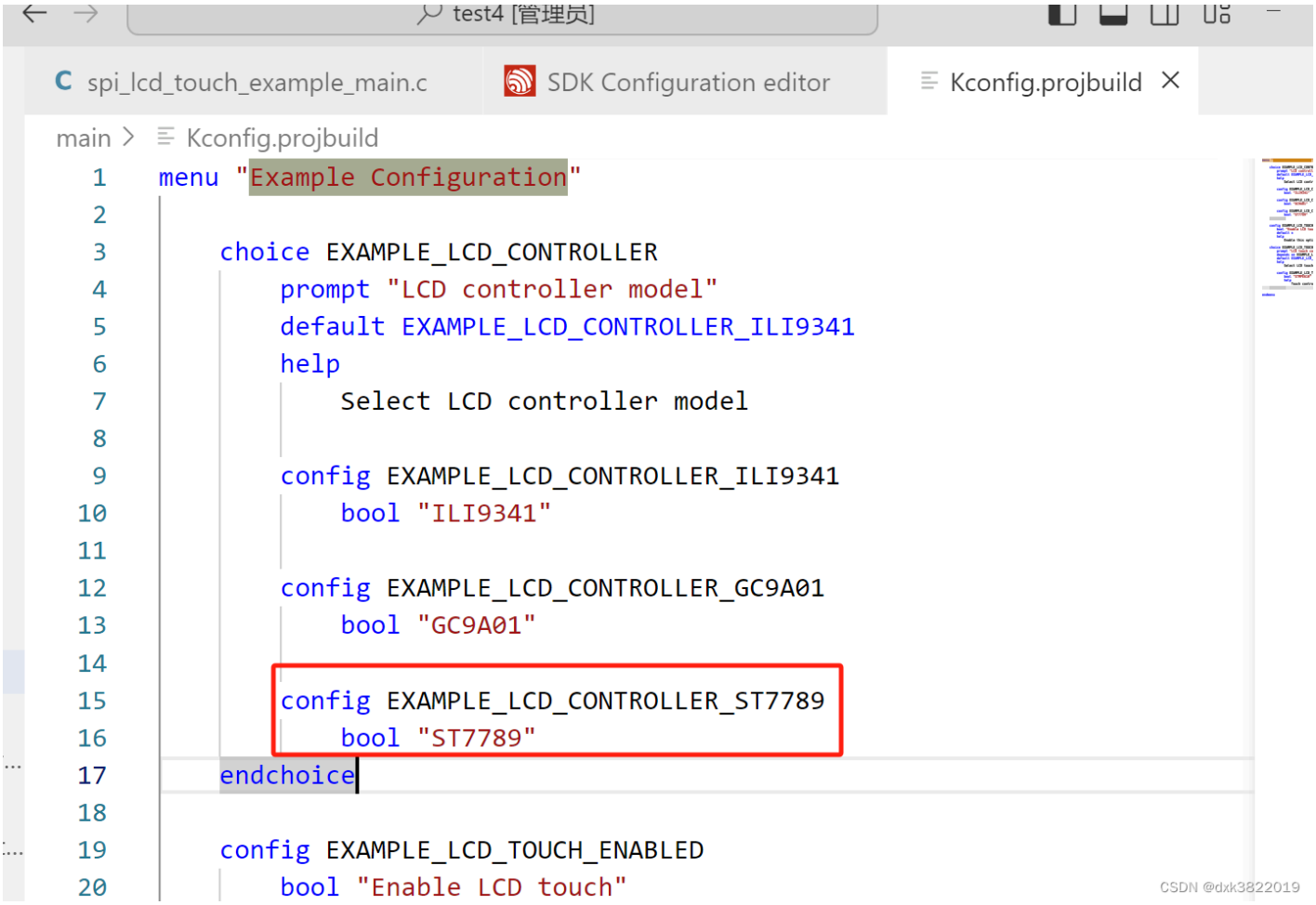


3.修改Kconfig，添加ST7789选项

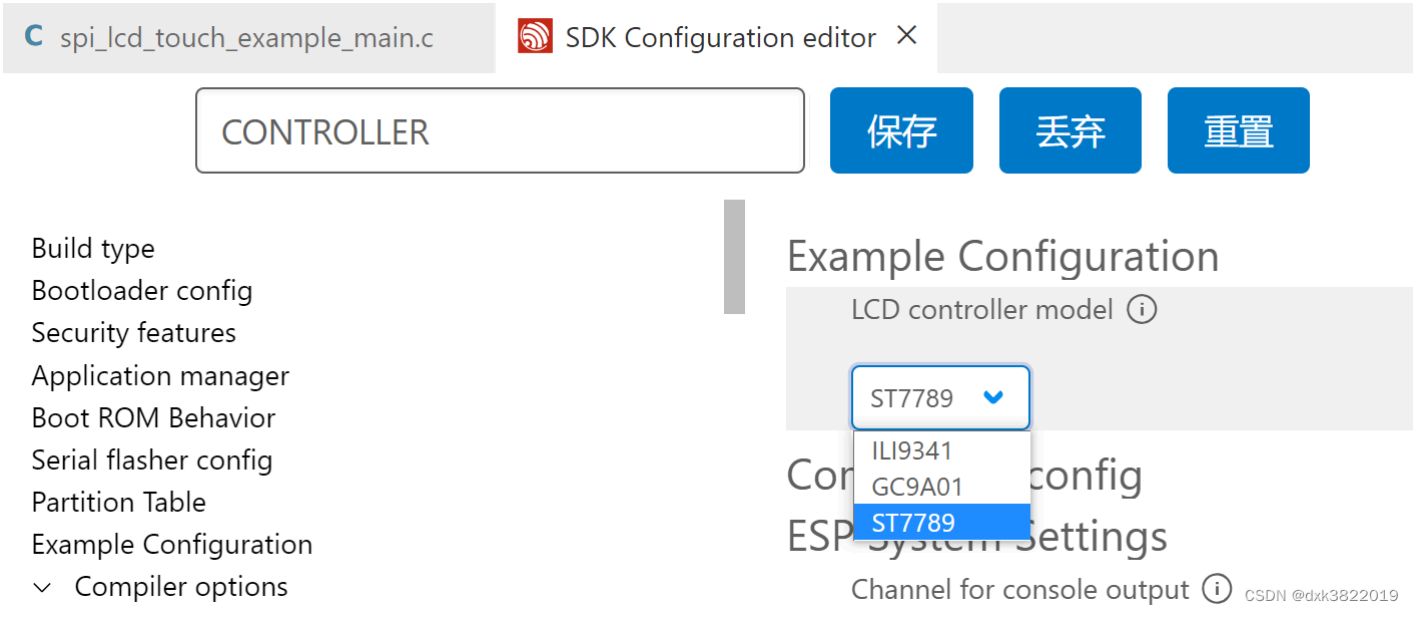
找到main文件夹下的Kconfig.projbuild文件并打开。



在下图位置添加ST7789选项，然后保存并关闭文件。



重启一下VSC，再次进入SDK menuconfig配置，搜CONTROLLER，发现LCD_controller里面应该有了ST7789选项，选择ST7789之后，点击保存。



4.配置LCD驱动接口

在main文件夹下的spi_lcd_touch_example_main.c文件中添加ST7789的屏幕配置宏定义。

spi_lcd_touch_example_main.c 3

main > spi_lcd_touch_example_main.c > EXAMPLE_LCD_PARAM_BITS

```
47  #define EXAMPLE_PIN_NUM_LCD_CS          4
48  #define EXAMPLE_PIN_NUM_BK_LIGHT        2
49  #define EXAMPLE_PIN_NUM_TOUCH_CS        15
50
51  // The pixel number in horizontal and vertical
52  #if CONFIG_EXAMPLE_LCD_CONTROLLER_ILI9341
53  #define EXAMPLE_LCD_H_RES               240
54  #define EXAMPLE_LCD_V_RES               320
55  #elif CONFIG_EXAMPLE_LCD_CONTROLLER_GC9A01
56  #define EXAMPLE_LCD_H_RES               240
57  #define EXAMPLE_LCD_V_RES               240
58  #elif CONFIG_EXAMPLE_LCD_CONTROLLER_ST7789
59  #define EXAMPLE_LCD_H_RES               240
60  #define EXAMPLE_LCD_V_RES               240
61  #endif
62  // Bit number used to represent command and parameter
63  #define EXAMPLE_LCD_CMD_BITS            8
```


添加一下ST7789的接口配置。

spi_lcd_touch_example_main.c

main > spi_lcd_touch_example_main.c > app_main(void)

```
219 #elif CONFIG_EXAMPLE_LCD_CONTROLLER_GC9A01
220     ESP_LOGI(TAG, "Install GC9A01 panel driver");
221     ESP_ERROR_CHECK(esp_lcd_new_panel_gc9a01(io_handle, &panel_config, &panel_ha
222 #elif CONFIG_EXAMPLE_LCD_CONTROLLER_ST7789
223     ESP_LOGI(TAG, "Install GC9A01 panel driver");
224     ESP_ERROR_CHECK(esp_lcd_new_panel_st7789(io_handle, &panel_config, &panel_ha
225 #endif
226
227     ESP_ERROR_CHECK(esp_lcd_panel_reset(panel_handle));
228     ESP_ERROR_CHECK(esp_lcd_panel_init(panel_handle));
229 #if CONFIG_EXAMPLE_LCD_CONTROLLER_GC9A01
230     ESP_ERROR_CHECK(esp_lcd_panel_invert_color(panel_handle, true));
231     ESP_ERROR_CHECK(esp_lcd_panel_mirror(panel_handle, true, false));
232 #elif CONFIG_EXAMPLE_LCD_CONTROLLER_ST7789
233     ESP_ERROR_CHECK(esp_lcd_panel_invert_color(panel_handle, true));
234     ESP_ERROR_CHECK(esp_lcd_panel_mirror(panel_handle, false, false));
235 #endif
236     // user can flush pre-defined pattern to the screen before we turn on the sc
237     ESP_ERROR_CHECK(esp_lcd_panel_disp_on_off(panel_handle, true));
238
```

此处也要改一下RGB设置。

spi_lcd_touch_example_main.c | lcd_types.h | SDK Configuration editor

main > spi_lcd_touch_example_main.c > app_main(void)

```
198         .cs_gpio_num = EXAMPLE_PIN_NUM_LCD_CS,
199         .pclk_hz = EXAMPLE_LCD_PIXEL_CLOCK_HZ,
200         .lcd_cmd_bits = EXAMPLE_LCD_CMD_BITS,
201         .lcd_param_bits = EXAMPLE_LCD_PARAM_BITS,
202         .spi_mode = 0,
203         .trans_queue_depth = 10,
204         .on_color_trans_done = example_notify_lvgl_flush_ready,
205         .user_ctx = &disp_drv,
206     };
207     // Attach the LCD to the SPI bus
208     ESP_ERROR_CHECK(esp_lcd_new_panel_io_spi((esp_lcd_spi_bus_handle_t)LCD_HO
209
210     esp_lcd_panel_handle_t panel_handle = NULL;
211     esp_lcd_panel_dev_config_t panel_config = {
212         .reset_gpio_num = EXAMPLE_PIN_NUM_LCD_RST,
213         .rgb_endian = LCD_RGB_ENDIAN_RGB,
214         .bits_per_pixel = 16,
215     };
216 #if CONFIG_EXAMPLE_LCD_CONTROLLER_ILI9341
217     ESP_LOGI(TAG, "Install ILI9341 panel driver");
218     ESP_ERROR_CHECK(esp_lcd_new_panel_ili9341(io_handle, &panel_config, &pane
```


设置下引脚宏定义。

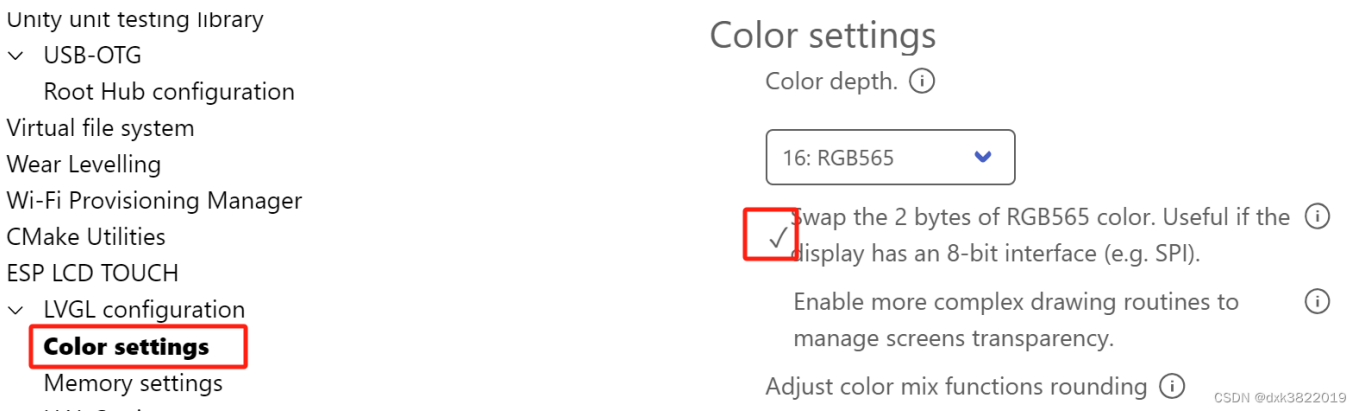
spi_lcd_touch_example_main.c | lcd_types.h | SDK Configuration editor

main > spi_lcd_touch_example_main.c > ...

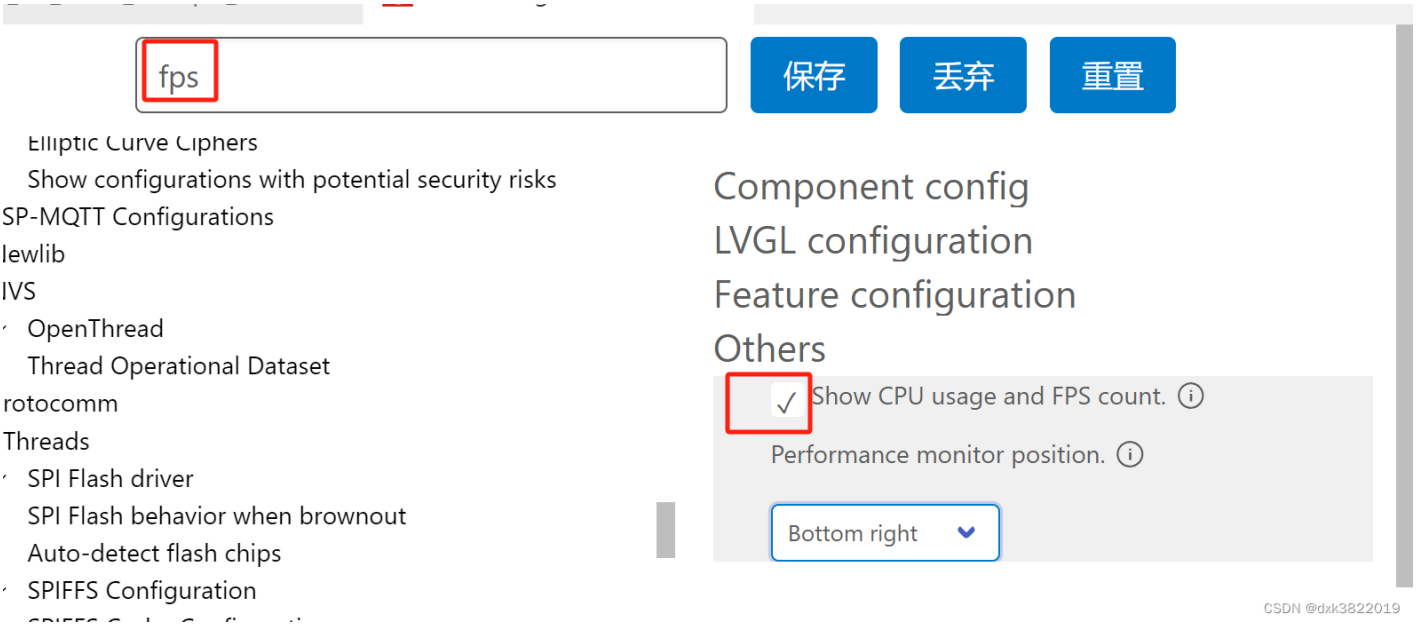
```
33  // Using SPI2 in the example
34  #define LCD_HOST  SPI2_HOST
35
36  ////////////////////////////////////////////////////////////////////////////
37  ////////////////////// Please update the following configuration according to your
38  ////////////////////////////////////////////////////////////////////////////
39  #define EXAMPLE_LCD_PIXEL_CLOCK_HZ     (20 * 1000 * 1000)
40  #define EXAMPLE_LCD_BK_LIGHT_ON_LEVEL  1
41  #define EXAMPLE_LCD_BK_LIGHT_OFF_LEVEL !EXAMPLE_LCD_BK_LIGHT_ON_LEVEL
42  #define EXAMPLE_PIN_NUM_SCLK           18
43  #define EXAMPLE_PIN_NUM_MOSI           19
44  #define EXAMPLE_PIN_NUM_MISO           21
45  #define EXAMPLE_PIN_NUM_LCD_DC         5
46  #define EXAMPLE_PIN_NUM_LCD_RST        3
47  #define EXAMPLE_PIN_NUM_LCD_CS         4
48  #define EXAMPLE_PIN_NUM_BK_LIGHT       2
49  #define EXAMPLE_PIN_NUM_TOUCH_CS       15
50
```


5.配置LVGL

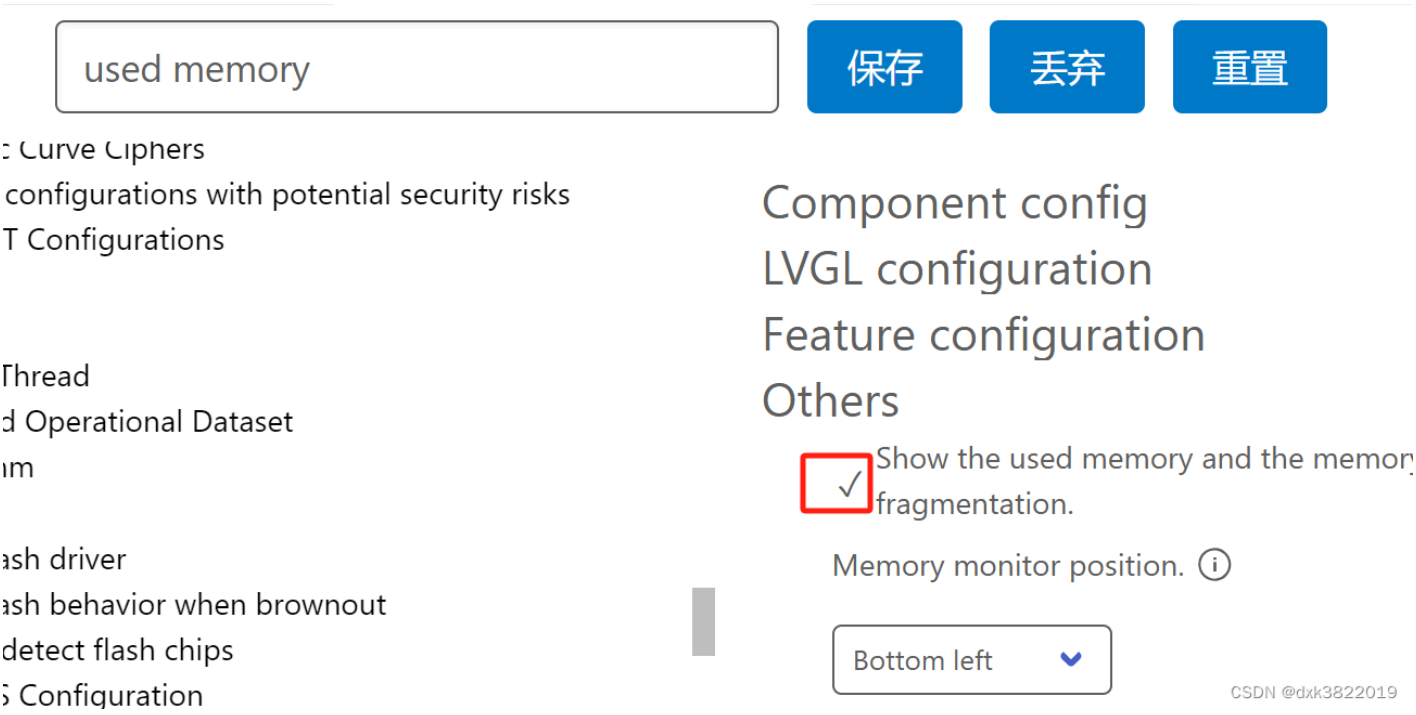
打开这个交换字节的选项。



打开显示FPS显示

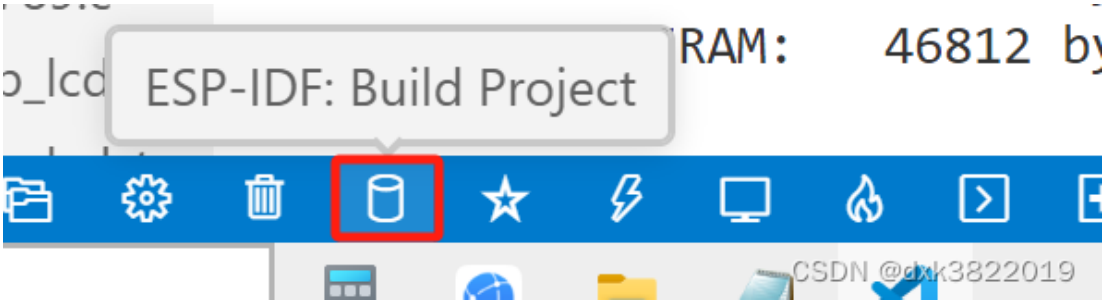


打开显示memory显示

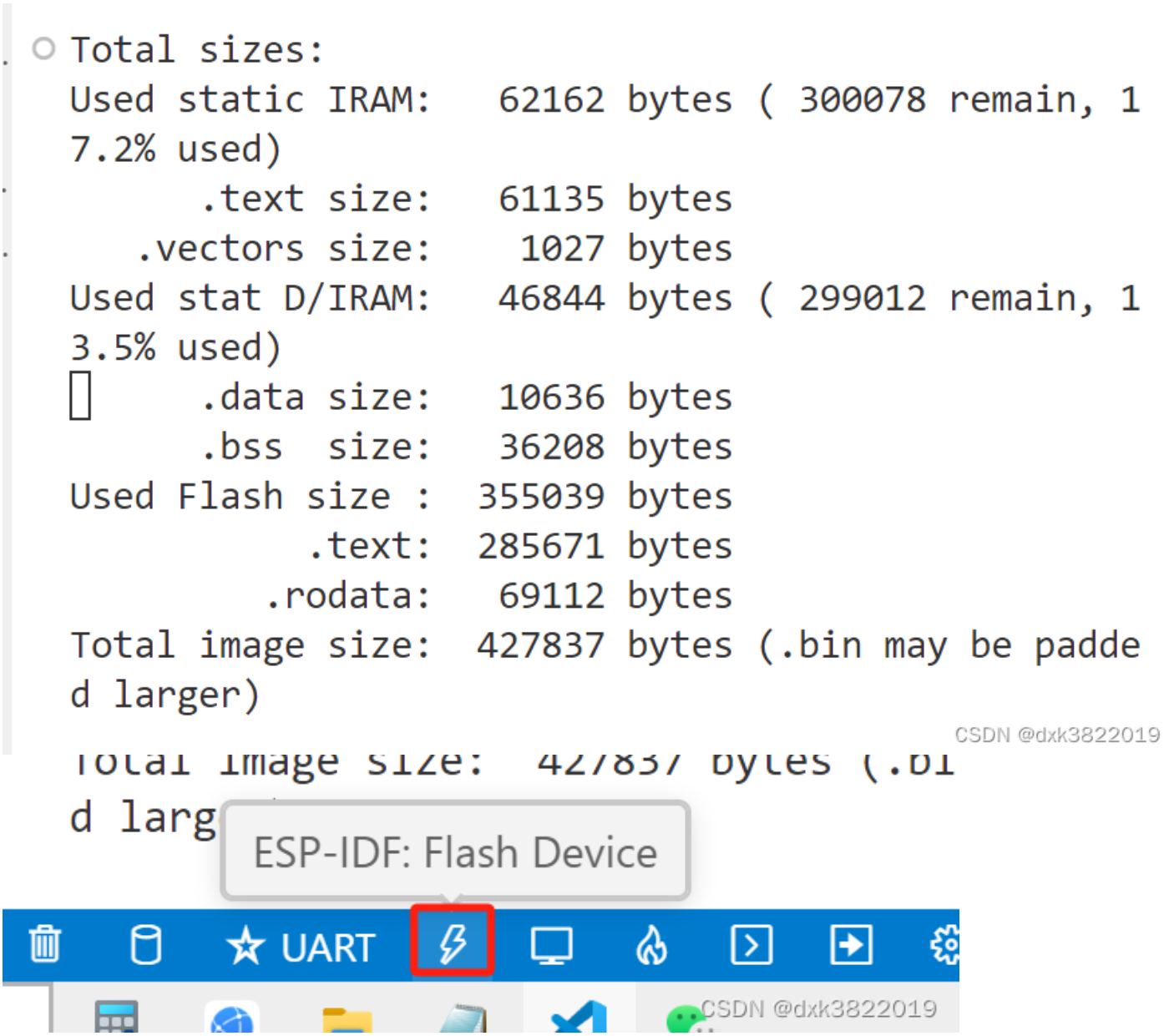


6.编译下载程序

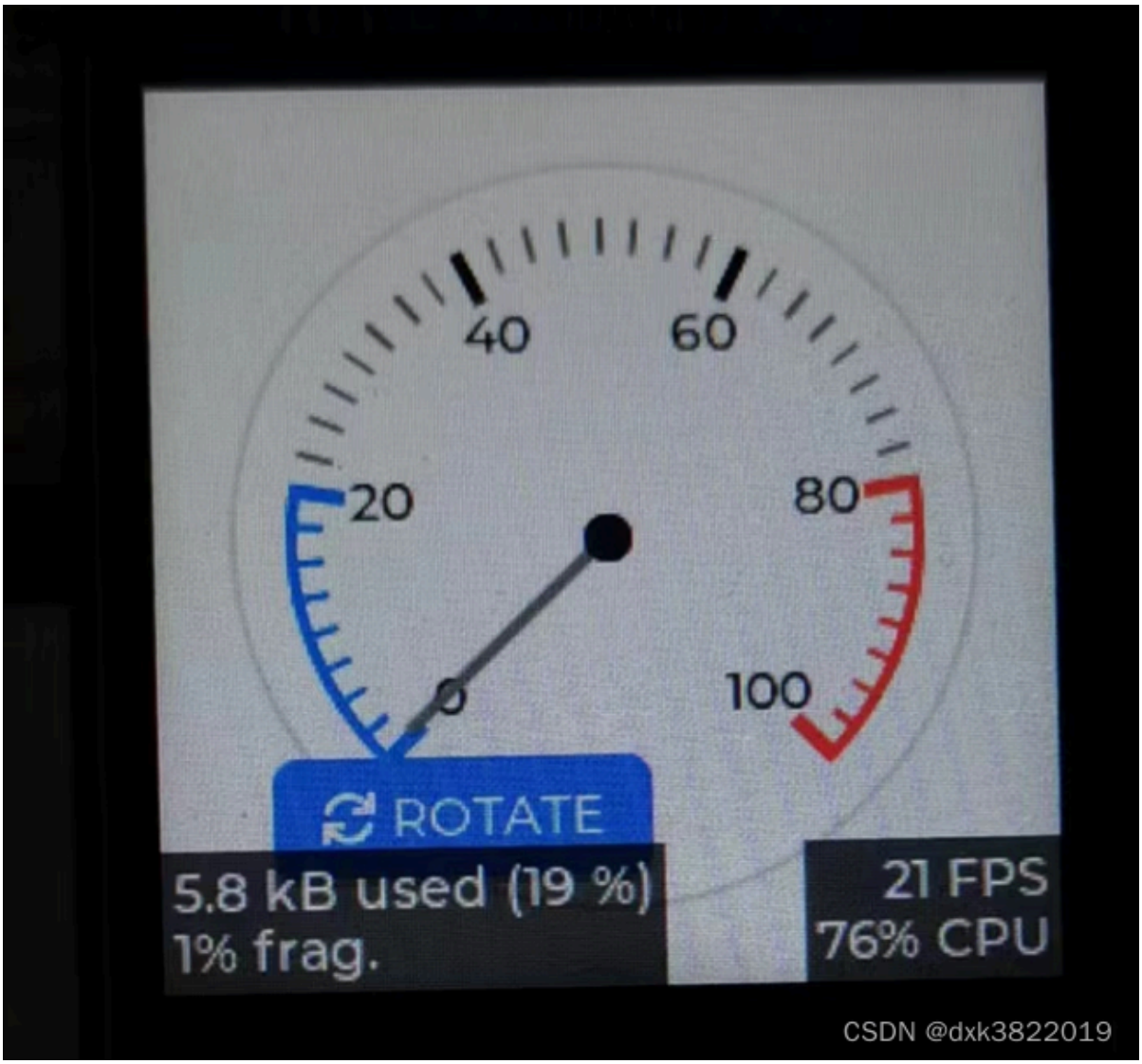
编译一下工程。



显示编译成功。



连接好硬件，通电，搞定!


40
60
20
80
0
100
ROTATE
5.8 kB used (19 %)
1% frag.
21 FPS
76% CPU
CSDN @dxk3822019

感谢大家观看！