

E32R28T&E32N28T

2.8 寸 ESP-IDF

示例程序说明

目 录

1. 软件和硬件平台说明.....	3
2. 引脚分配说明.....	3
3. 示例程序说明.....	5
3.1. 搭建 ESP32 MicroPython 开发环境.....	5
3.2. 示例程序使用说明.....	5

1. 软件和硬件平台说明

- 模块：2.8寸ESP32-32E显示模块，拥有240x320分辨率，采用ILI9341屏驱IC。
- 模块主控：ESP32-WROOM-32E模组，最高主频240MHz，支持2.4G WIFI+蓝牙。
- ESP-IDF版本：5.3.1。
- LVGL版本：8.3.11。

2. 引脚分配说明

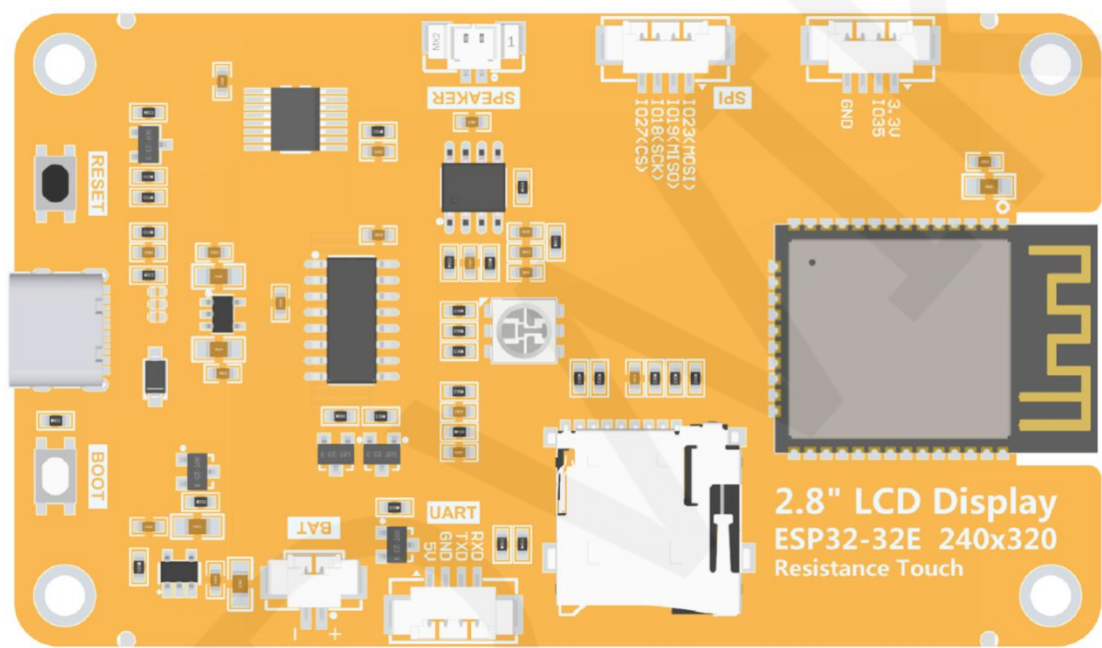


图2.1 2.8寸ESP32-32E显示模块背面图

2.8寸ESP32显示模块主控为ESP32-32E，其连接板载外设的GPIO分配如下表格所示：

ESP32-32E引脚分配说明			
板载设备	板载设备引脚	ESP32-32E连接引脚	说明
LCD	TFT_CS	IO15	液晶屏片选控制信号，低电平有效
	TFT_RS	IO2	液晶屏命令/数据选择控制信号 高电平：数据，低电平：命令
	TFT_SCK	IO14	液晶屏SPI总线时钟信号
	TFT_MOSI	IO13	液晶屏SPI总线写数据信号
	TFT_MISO	IO12	液晶屏SPI总线读数据信号

	TFT_RST	EN	液晶屏复位控制信号，低电平复位（和ESP32-32E主控共用复位引脚）	
	TFT_BL	IO21	液晶屏背光控制信号（高电平点亮背光，低电平关闭背光）	
RTP	TP_SCK	IO25	电阻触摸屏SPI总线时钟信号	
	TP_DIN	IO32	电阻触摸屏SPI总线写数据信号	
	TP_DOUT	IO39	电阻触摸屏SPI总线读数据信号	
	TP_CS	IO33	电阻触摸屏片选控制信号，低电平有效	
	TP_IRQ	IO36	电阻触摸屏触摸中断信号，产生触摸时，输入低电平到主控	
LED	LED_RED	IO22	红色LED灯	RGB三色LED灯，共阳极，低电平点亮，高电平关闭。
	LED_GREEN	IO16	绿色LED灯	
	LED_BLUE	IO17	蓝色LED灯	
SDCARD	SD_CS	IO5	SD卡片选信号，低电平有效	
	SD_MOSI	IO23	SD卡SPI总线写数据信号	
	SD_SCK	IO18	SD卡SPI总线时钟信号	
	SD_MISO	IO19	SD卡SPI总线读数据信号	
BATTERY	BAT_ADC	IO34	电池电压ADC值获取信号（输入）	
Audio	Audio_ENABLE	IO4	音频使能信号，低电平使能，高电平禁止	
	Audio_DAC	IO26	音频信号DAC输出信号	
KEY	BOOT_KEY	IO0	下载模式选择按键（按住该按键上电，然后松开就会进入下载模式）	
	RESET_KEY	EN	ESP32-23E复位按键，低电平复位（和液晶屏复位共用）	
Serial Port	RX0	RXD0	ESP32-32E串口接收信号	
	TX0	TXD0	ESP32-32E串口发送信号	
POWER	TYPE-C_POWER	/	Type-C电源接口，接入5V电压。	

表2.1 ESP32-32E板载外设引脚分配说明

3. 示例程序说明

3.1. 搭建ESP32 IDF开发环境

ESP32 IDF开发环境搭建详细说明见资料包里“使用VS Code搭建ESP-IDF环境”说明文档。

3.2. 示例程序使用说明

示例程序位于资料包的“1-示例程序_Demo\ESP32-IDF”目录下，如下图所示：



图 3.1 示例程序

该示例程序可已经将 LVGL 移植，且相关的程序文件也已经修改好，直接可以使用。关于 LVGL 移植说明请参考资料包里“ESP-IDF_LVGL 移植说明”文档。示例程序的使用步骤如下：

- 将示例程序整个文件夹“2.8inch_ESP32_LVGL”拷贝到全英文命名的路径下。否则编译时会因为找不到路径而报错。
- 打开 VS Code 软件，点击“文件”->“打开文件夹”，如下图所示：

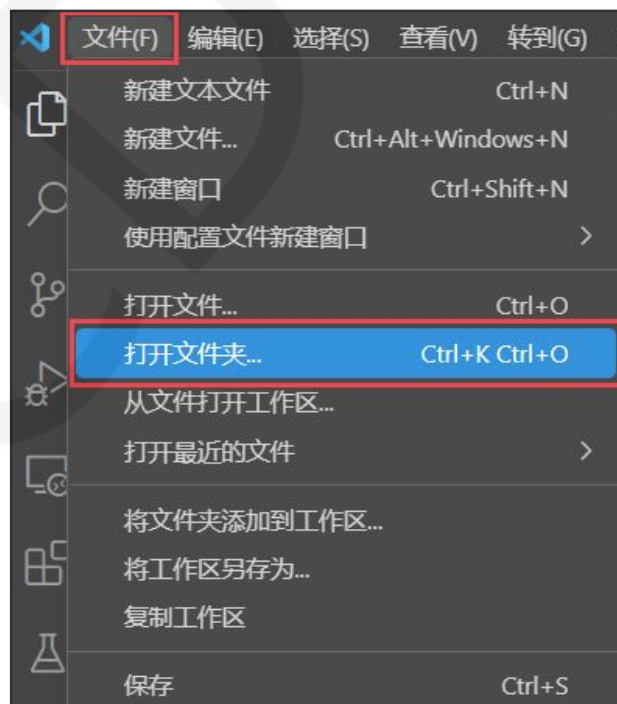


图 3.2 打开文件夹

- C. 找到示例程序文件夹，点击选中，然后点击“**选择文件夹**”按钮，这样就打开了示例程序，如下图所示：



图 3.3 找到示例程序文件夹

- D. 将 ESP32 设备和电脑连接，在 VS Code 底部工具栏选择正确的串口号、芯片、下载方式，然后点击  按钮，进行编译和烧录。
- E. 烧录完成后，就可以看到显示模块有内容显示了。