ES3C28P&ES3N28P 2.8 寸 MicroPython 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-S3 MicroPython 开发环境.....5

3.2. 上传文件.....5

3.3. 示例程序使用说明.....8

1. 软件和硬件平台说明

模块：2.8寸IPS ESP32-S3显示模块，拥有240x320分辨率，采用ILI9341V屏驱IC。

模块主控：ESP32-S3芯片，最高主频240MHz，支持2.4G WIFI+蓝牙。

Thonny 版本：4.1.6。

ESP32 MicroPython固件版本：1.25.0。

2. 引脚分配说明

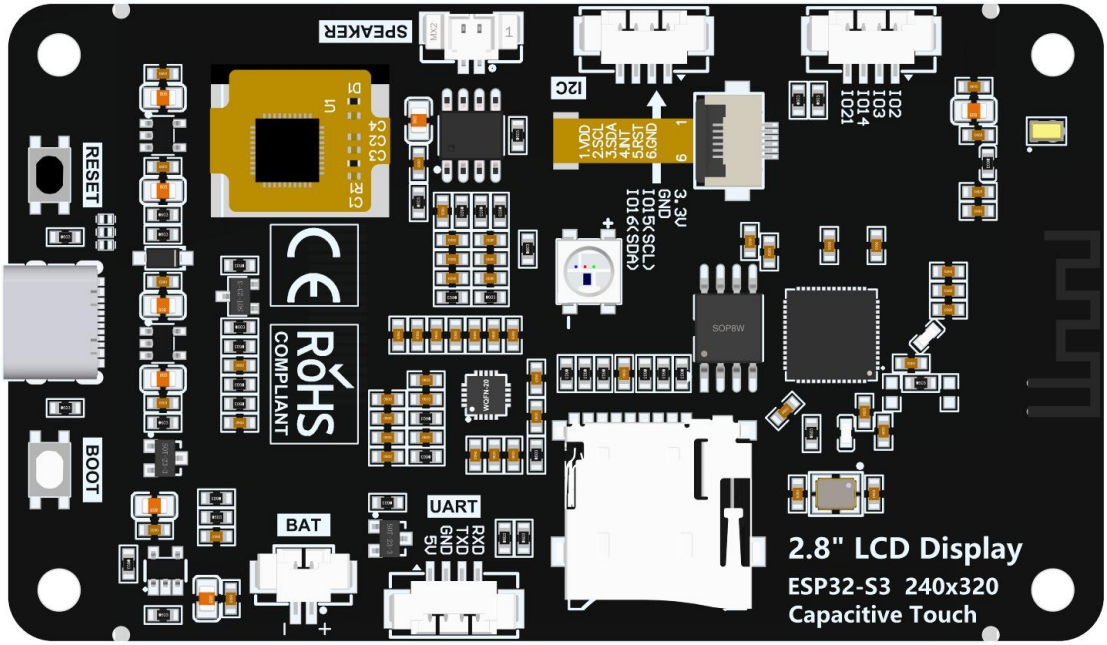

图2.1 2.8寸ESP32-S3显示模块背面图

2.8寸ESP32-S3显示模块主控为ESP32-S3芯片，其连接板载外设的GPIO分配如下表所示：

ESP32-S3芯片引脚分配说明			
板载设备	**板载设备引脚**	**ESP32-S3连接引脚**	**说明**
LCD	**TFT_CS**	IO10	液晶屏片选控制信号，低电平有效
	TFT_RS	IO46	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	IO12	液晶屏SPI总线时钟信号
	TFT_MOSI	IO11	液晶屏SPI总线写数据信号
	TFT_MISO	IO13	液晶屏SPI总线读数据信号
	TFT_RST	CHIP_PU	液晶屏复位控制信号，低电平复位（和

			ESP32-S3主控共用复位引脚）
	TFT_BL	IO45	液晶屏背光控制信号（高电平点亮背光，低电平关闭背光）
CTP	TP_SDA	IO16	电容触摸屏I2C总线数据信号
	TP_SCL	IO15	电容触摸屏I2C总线时钟信号
	TP_RST	IO18	电容触摸屏复位控制信号，低电平复位
	TP_INT	IO17	电容触摸屏中断输入信号，发生触摸事件时，输入低电平。
LED	RGB_INT	IO42	单线RGB三色LED灯，可以根据不同信号分别点亮内部的红绿蓝三种灯珠
SDCARD	SD_CLK	IO38	SD卡SDIO总线时钟信号
	SD_CMD	IO40	SD卡SDIO总线命令信号
	SD_D0	IO39	SD卡SDIO总线数据信号（DATA0~DATA3四根数据线）
	SD_D1	IO41	
	SD_D2	IO48	
	SD_D3	IO47	
BATTERY	BAT_ADC	IO9	电池电压ADC值获取输入信号
Audio	Audio_EN	IO1	音频输出使能信号，低电平使能，高电平禁止
	I2S_MCK	IO4	音频I2S总线主时钟信号
	I2S_SCK	IO5	音频I2S总线位时钟信号
	I2S_DO	IO8	音频I2S总线位输出数据信号
	I2S_LRC	IO7	音频I2S总线左右声道选择信号。高电平：右声道；低电平：左声道
	I2S_DI	IO6	音频I2S总线位输入数据信号
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-s3复位按键，低电平复位（和液晶屏复位共用）
USB	USB_N	IO19	USB总线差分信号数据线负极
	USB_P	IO20	USB总线差分信号数据线正极
Serial Port	TX0	IO43	ESP32-S3串口0发送信号
	RX0	IO44	ESP32-S3串口0接收信号
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-S3 MicroPython开发环境

ESP32-S3 MicroPython开发环境搭建详细说明见资料包里“**ESP32-S3_MicroPython开发环境搭建**”说明文档。

3.2. 上传文件

开发环境搭建好之后，需将相关的文件上传到ESP32-S3设备，才能运行测试程序。

上传文件之前，先了解一下MicroPython示例程序的目录内容。打开资料包里“**1-示例程序_Demo\MicroPython**”目录，如下图所示：

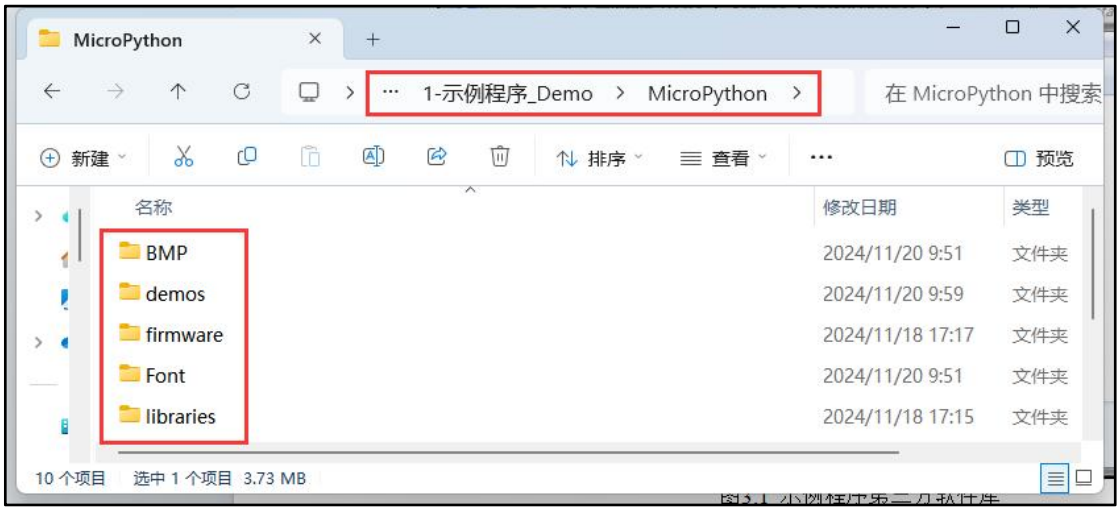

图3.1 MicroPython示例程序目录

各文件夹内容介绍如下：

BMP：存放有示例程序需要使用的BMP格式图片。

demos：存放有示例程序

firmware：存放有MicroPython固件（搭建开发环境时需要烧录）

Font：存放有示例程序需要使用的中英文字符取模数据。

libraries：存放有示例程序需要使用的 MicroPython 库文件

了解完 MicroPython 示例程序的目录内容后，接下来上传程序文件到 ESP32-S3 设备，步骤如下：

A、使用USB线将ESP32-S3显示模块连接电脑上电。

B、打开Thonny软件，配置ESP32-S3的MicroPython解释器，如下图所示：

(如果已经配置，该步骤可省略)

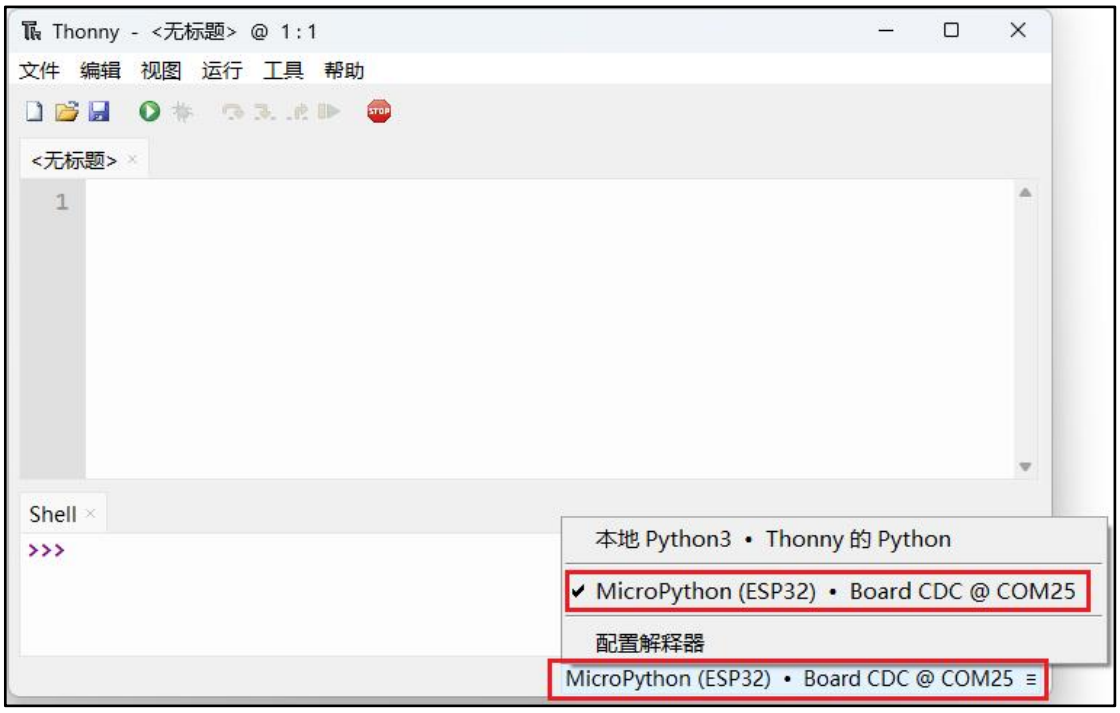

图3.2 选择MicroPython解释器

C、点击工具栏 按钮连接ESP32-S3设备，待shell信息栏出现如下提示，则说明设备连接成功。

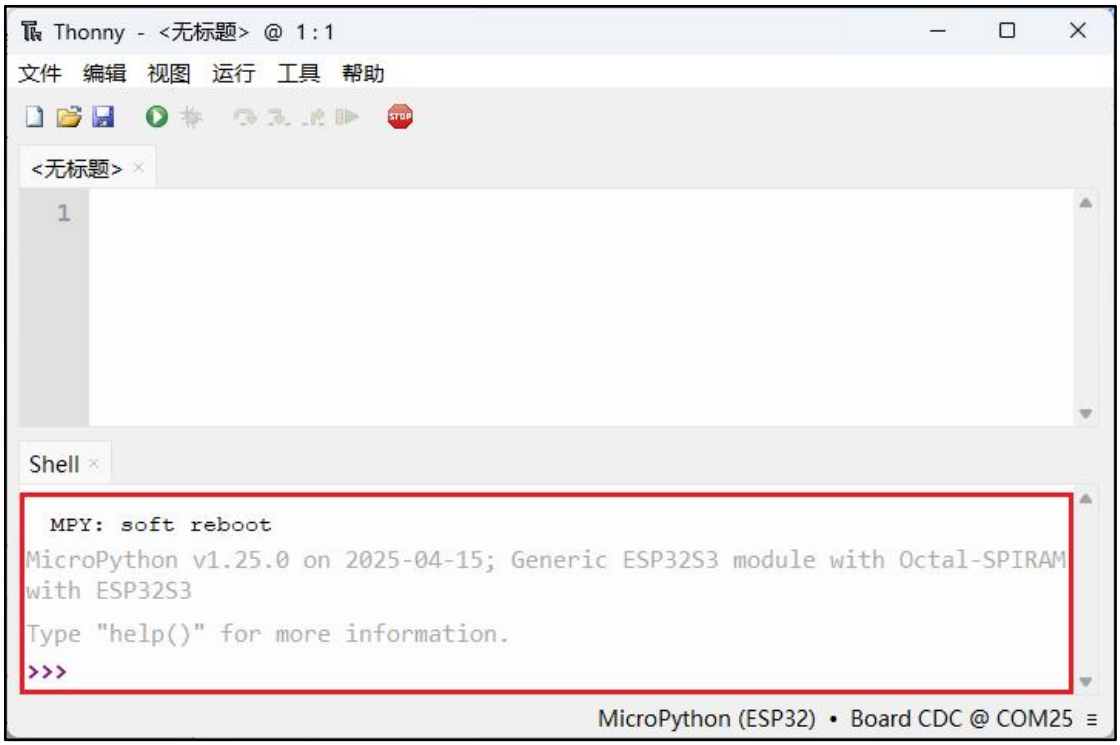

图3.3 连接ESP32-S3设备

D、点击“**视图->文件**”按钮，打开文件窗口（如已打开，则忽略该操作），在窗口中找到资料包里的“**1-示例程序_Demo\MicroPython**”目录，单击鼠标左键

选择目录里目标文件，在单击鼠标右键选择“**上传到/**”，就可以将目标文件上传了。如下图所示：

需要注意，上传文件时，ESP32-S3不能运行任何程序，否则会上传失败

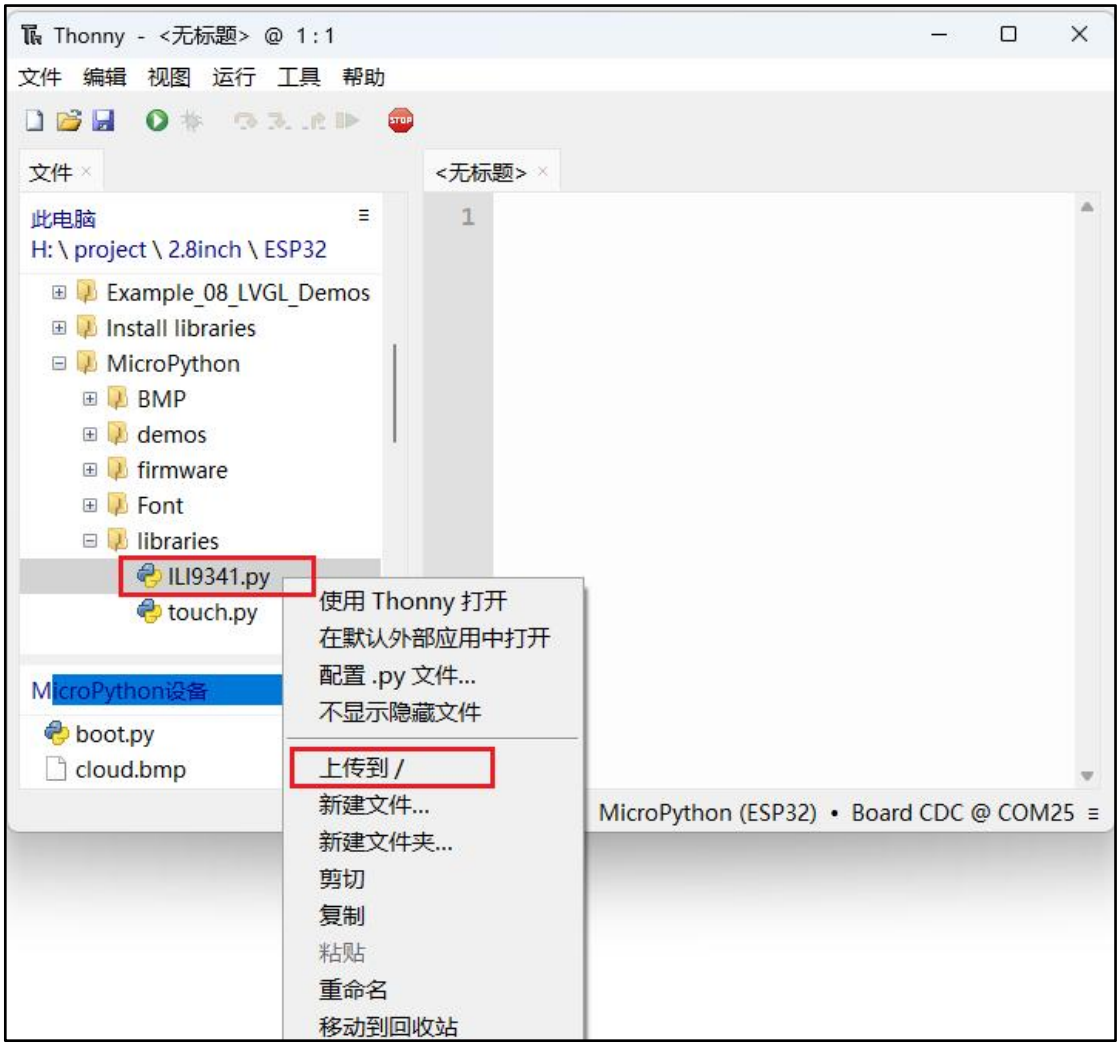

图3.4 上传文件到ESP32-S3设备

E、按上述方法分别将“BMP”、“Font”、“libraries”目录下的文件上传到ESP32-S3设备。“demos”目录下的文件可传可不传。如下图所示：

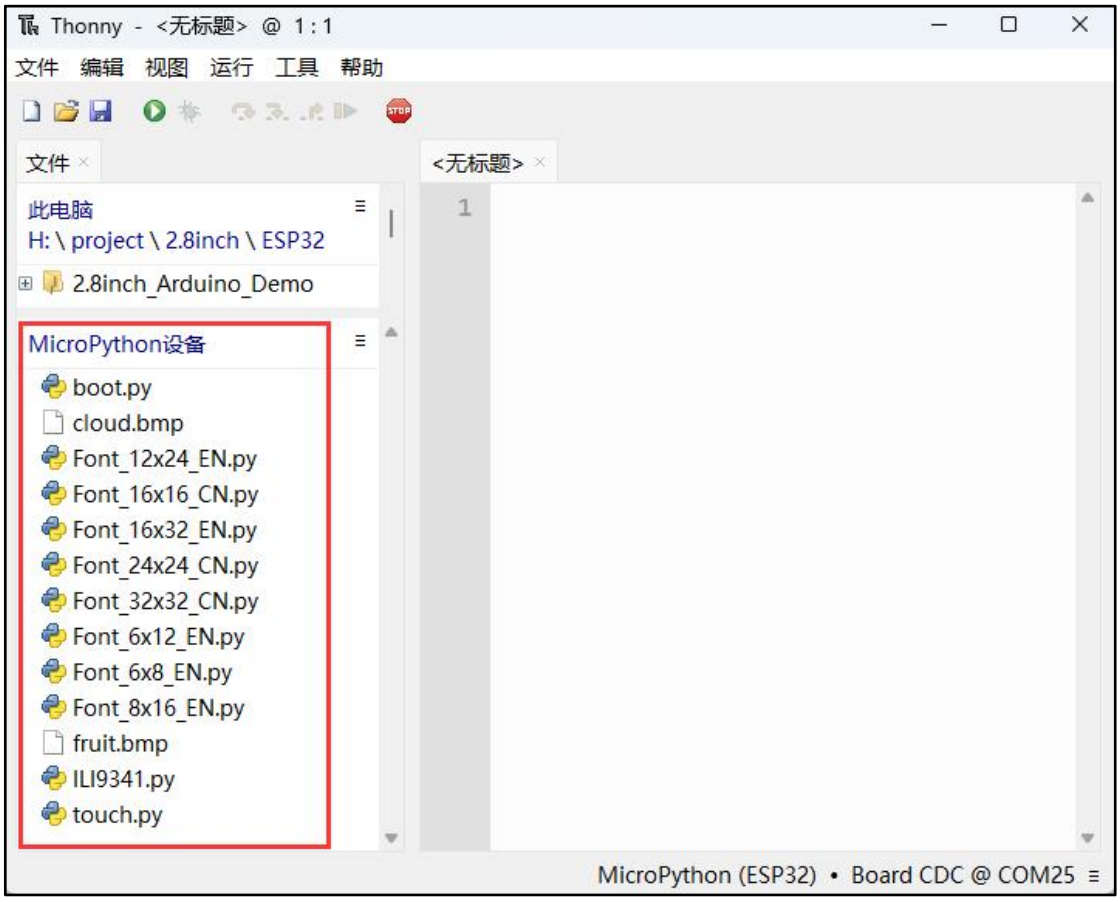

图3.5 完成文件上传

3.3. 示例程序使用说明

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

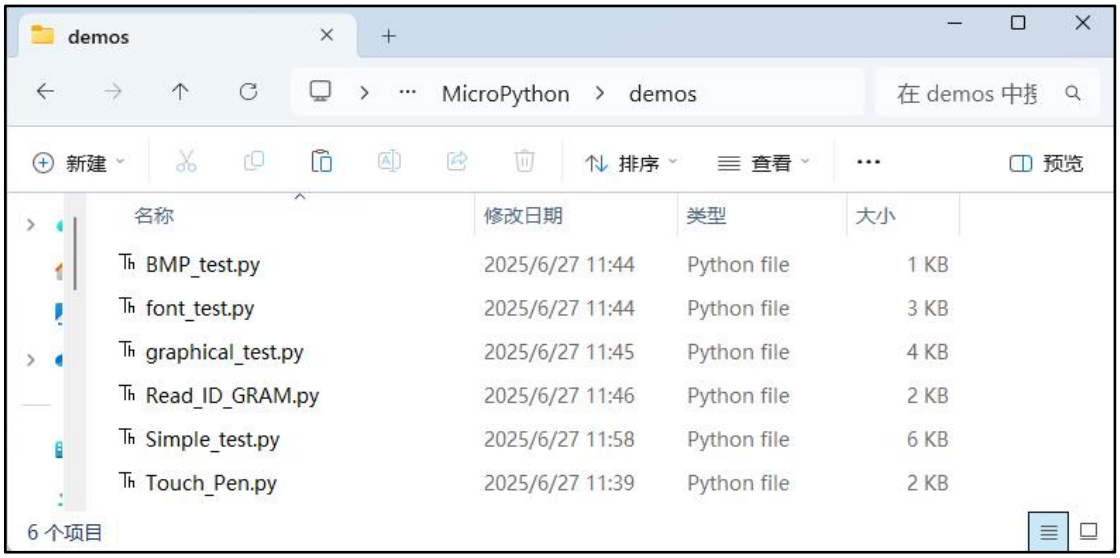

图 3.6 示例程序

示例程序可以上传到 ESP32-S3 设备打开运行，也可以在本地电脑里打开运行。如

果需要在 ESP32-S3 显示模块里上电自动运行，需要将示例程序名称改为“**main.py**”，然后上传到 ESP32-S3 显示模块。

在 thonny 软件里，打开目标示例程序，点击菜单栏 按钮，就可以运行了。如果运行失败，需要重新连接 ESP32-S3 设备。

各示例程序介绍如下：

BMP_test.py

此示例程序需要依赖 ILI9341.py 库，显示内容为 BMP 格式的图片

font_test.py

此示例程序需要依赖 ILI9341.py 库，显示内容为各种尺寸的中英文字符。字体取模数据需要按照相关的格式保存在字体文件里。关于字符取模说明，请查阅如下网址：

http://www.lcdwiki.com/zh/%E3%80%90%E6%95%99%E7%A8%8B%E3%80%91%E4%B8%AD%E8%8B%B1%E6%96%87%E6%98%BE%E7%A4%BA%E5%8F%96%E6%A8%A1%E8%AE%BE%E7%BD%AE

graphical_test.py

此示例程序需要依赖 ILI9341.py 库，显示内容为点、线、矩形、圆角矩形、三角形、圆形，椭圆形等图形绘制和填充以及显示方向设置。

Read_ID_GRAM.py

此示例程序需要依赖 ILI9341.py 库，显示内容为 LCD 的 ID 和 RGAM 颜色值读取。

Simple_test.py

此示例不依赖任何软件库，显示内容为简单的刷屏。

Touch_Pen.py

此示例需要依赖 ILI9341.py 库和 touch.py 库，显示内容为触摸屏画点画线操作。