

st7735-LCD 显示

版本：1.0

日期：2026-02-06

状态：受控/临时文件

文件保密级别：（请勾选 <input checked="" type="checkbox"/> ）		
绝密 <input type="checkbox"/>	保密 <input type="checkbox"/>	公开 <input checked="" type="checkbox"/>

文件管控表

文件变更记录			
修订日期	最新版本	修订内容	编制
2026-02-06	1	首次发行	Wells Li

上海移远通信技术股份有限公司（以下简称“移远通信”）始终以为客户提供最及时、最全面的服务为宗旨。如需任何帮助，请随时联系我司上海总部，联系方式如下：

上海移远通信技术股份有限公司
上海市闵行区田林路 1016 号科技绿洲 3 期（B 区）5 号楼 邮编：200233
电话：+86 21 5108 6236 邮箱：info@quectel.com

或联系我司当地办事处，详情请登录：<http://www.quectel.com/cn/support/sales.htm>。

如需技术支持或反馈我司技术文档中的问题，请随时登录网址：
<http://www.quectel.com/cn/support/technical.htm> 或发送邮件至：support@quectel.com。

前言

移远通信提供该文档内容以支持客户的产品设计。客户须按照文档中提供的规范、参数来设计产品。同时，您理解并同意，移远通信提供的参考设计仅作为示例。您同意在设计您目标产品时使用您独立的分析、评估和判断。在使用本文档所指导的任何硬软件或服务之前，请仔细阅读本声明。您在此承认并同意，尽管移远通信采取了商业范围内的合理努力来提供尽可能好的体验，但本文档和其所涉及服务是在“可用”基础上提供给您的。移远通信可在未事先通知的情况下，自行决定随时增加、修改或重述本文档。

使用和披露限制

许可协议

除非移远通信特别授权，否则我司所提供硬软件、材料和文档的接收方须对接收的内容保密，不得将其用于除本项目的实施与开展以外的任何其他目的。

版权声明

移远通信产品和本协议项下的第三方产品可能包含受移远通信或第三方材料、硬软件和文档版权保护的相关资料。除非事先得到书面同意，否则您不得获取、使用、向第三方披露我司所提供的文档和信息，或对此类受版权保护的资料进行复制、转载、抄袭、出版、展示、翻译、分发、合并、修改，或创造其衍生作品。移远通信或第三方对受版权保护的资料拥有专有权，不授予或转让任何专利、版权、商标或服务商标权的许可。为避免歧义，除了正常的非独家、免版税的产品使用许可，任何形式的购买都不可被视为授予许可。对于任何违反保密义务、未经授权使用或以其他非法形式恶意使用所述文档和信息的违法侵权行为，移远通信有权追究法律责任。

商标

除另行规定，本文档中的任何内容均不授予在广告、宣传或其他方面使用移远通信或第三方的任何商标、商号及名称，或其缩略语，或其仿冒品的权利。

第三方权利

您理解本文档可能涉及一个或多个属于第三方的硬软件和文档（“第三方材料”）。您对此类第三方材料的使用应受本文档的所有限制和义务约束。

移远通信针对第三方材料不做任何明示或暗示的保证或陈述，包括但不限于任何暗示或法定的适销性或特定用途的适用性、平静受益权、系统集成、信息准确性以及与许可技术或被许可人使用许可技术相关的不侵犯任何第三方知识产权的保证。本协议中的任何内容都不构成移远通信对任何移远通信产品或任何其他软硬件、设备、工具、信息或产品的开发、增强、修改、分销、营销、销售、提供销售或以其他方式维持生产的陈述或保证。此外，移远通信免除因交易过程、使用或贸易而产生的任何和所有保证。

隐私声明

为实现移远通信产品功能，特定设备数据将会上传至移远通信或第三方服务器（包括运营商、芯片供应商或您指定的服务器）。移远通信严格遵守相关法律法规，仅为实现产品功能之目的或在适用法律允许的情况下保留、使用、披露或以其他方式处理相关数据。当您与第三方进行数据交互前，请自行了解其隐私保护和数据安全政策。

免责声明

- 1) 移远通信不承担任何因未能遵守有关操作或设计规范而造成损害的责任。
- 2) 移远通信不承担因本文档中的任何因不准确、遗漏、或使用本文档中的信息而产生的任何责任。
- 3) 移远通信尽力确保开发中功能的完整性、准确性、及时性，但不排除上述功能错误或遗漏的可能。除非另有协议规定，否则移远通信对开发中功能的使用不做任何暗示或法定的保证。在适用法律允许的最大范围内，移远通信不对任何因使用开发中功能而遭受的损害承担责任，无论此类损害是否可以预见。
- 4) 移远通信对第三方网站及第三方资源的信息、内容、广告、商业报价、产品、服务和材料的可访问性、安全性、准确性、可用性、合法性和完整性不承担任何法律责任。

版权所有 ©上海移远通信技术股份有限公司 2024，保留一切权利。

Copyright © Quectel Wireless Solutions Co., Ltd. 2024.

目录

1	模块概述	5
2	类初始化	5
2.1.	LCD	5
2.1.1.	接口概述	5
2.1.2.	函数原型	5
2.1.3.	参数描述	5
2.1.4.	返回值描述	5
2.1.5.	示例	6
3	显示控制函数	6
3.1.	set_rotation	6
3.1.1.	接口概述	6
3.1.2.	函数原型	6
3.1.3.	参数描述	6
3.1.4.	返回值描述	6
3.1.5.	示例	6
3.2.	fill_screen	7
3.2.1.	接口概述	7
3.2.2.	函数原型	7
3.2.3.	参数描述	7
3.2.4.	返回值描述	7
3.2.5.	示例	7
3.3.	flush	7
3.3.1.	接口概述	7
3.3.2.	函数原型	7
3.3.3.	参数描述	7
3.3.4.	返回值描述	8
3.3.5.	示例	8
4	基本绘图函数	8
4.1.	draw_point	8
4.1.1.	接口概述	8
4.1.2.	函数原型	8
4.1.3.	参数描述	8
4.1.4.	返回值描述	8
4.1.5.	示例	8
4.2.	draw_line	9
4.2.1.	接口概述	9
4.2.2.	函数原型	9
4.2.3.	参数描述	9
4.2.4.	返回值描述	9
4.2.5.	示例	9

4.3.	draw_rectangle.....	9
4.3.1.	接口概述.....	9
4.3.2.	函数原型.....	9
4.3.3.	参数描述.....	9
4.3.4.	返回值描述.....	10
4.3.5.	示例.....	10
4.4.	draw_circle.....	10
4.4.1.	接口概述.....	10
4.4.2.	函数原型.....	10
4.4.3.	参数描述.....	10
4.4.4.	返回值描述.....	10
4.4.5.	示例.....	10
4.5.	fill_rectangle.....	11
4.5.1.	接口概述.....	11
4.5.2.	函数原型.....	11
4.5.3.	参数描述.....	11
4.5.4.	返回值描述.....	11
4.5.5.	示例.....	11
5	图像和文本显示函数.....	11
5.1.	show_image.....	11
5.1.1.	接口概述.....	11
5.1.2.	函数原型.....	12
5.1.3.	参数描述.....	12
5.1.4.	返回值描述.....	12
5.1.5.	示例.....	12
5.2.	show_string.....	12
5.2.1.	接口概述.....	12
5.2.2.	函数原型.....	12
5.2.3.	参数描述.....	12
5.2.4.	返回值描述.....	13
5.2.5.	示例.....	13
6	颜色相关功能.....	13
6.1.	COLOR565.....	13
6.1.1.	接口概述.....	13
6.1.2.	函数原型.....	13
6.1.3.	参数描述.....	13
6.1.4.	返回值描述.....	13
6.1.5.	示例.....	13
6.2.	预定义颜色常量.....	14
7	注意事项.....	14

1 模块概述

st7735 模块提供了对 ST7735S 显示驱动芯片的控制功能，支持 160x128 分辨率的 RGB565 显示。该模块包含基本的显示控制、图形绘制和文本显示功能。

2 类初始化

2.1. LCD

2.1.1. 接口概述

初始化 LCD 对象，配置 SPI 通信引脚并设置显示参数。

2.1.2. 函数原型

```
lcd = LCD(spi, dc_pin="F12", cs_pin="D14")
```

2.1.3. 参数描述

参数名	是否必填	数据类型	默认值	说明
spi	是	SPI 对象	-	machine.SPI 创建的对象，用于与 LCD 通信
dc_pin	否	string	"F12"	数据/命令选择引脚
cs_pin	否	string	"D14"	片选引脚

2.1.4. 返回值描述

返回 LCD 对象实例。

2.1.5. 示例

```
import machine
from st7735 import LCD

spi = machine.SPI(1, baudrate=20000000, polarity=0, phase=0)
lcd = LCD(spi, dc_pin="F12", cs_pin="D14")
```

3 显示控制函数

3.1. set_rotation

3.1.1. 接口概述

设置屏幕旋转方向。

3.1.2. 函数原型

```
lcd.set_rotation(rotation)
```

3.1.3. 参数描述

参数名	是否必填	数据类型	默认值	说明
rotation	是	int	1	旋转角度：0-0 度，1-90 度，2-180 度，3-270 度

3.1.4. 返回值描述

无返回值。

3.1.5. 示例

```
# 旋转 90 度
lcd.set_rotation(1)
```


3.2. fill_screen

3.2.1. 接口概述

用指定颜色填充整个屏幕。

3.2.2. 函数原型

```
lcd.fill_screen(color)
```

3.2.3. 参数描述

参数名	是否必填	数据类型	默认值	说明
color	是	int	-	RGB565 颜色值，可使用预定义颜色常量,见颜色常量定义

3.2.4. 返回值描述

无返回值。

3.2.5. 示例

```
# 填充白色
lcd.fill_screen(lcd.WHITE)
```

```
# 填充红色
lcd.fill_screen(lcd.RED)
lcd.flush()
```

3.3. flush

3.3.1. 接口概述

将缓冲区内容刷新到显示屏。

3.3.2. 函数原型

```
lcd.flush()
```

3.3.3. 参数描述

无。

3.3.4. 返回值描述

无返回值。

3.3.5. 示例

```
# 绘制内容后刷新显示
lcd.fill_screen(lcd.BLUE)
lcd.draw_circle(80, 64, 30, lcd.RED)
lcd.flush()
```

4 基本绘图函数

4.1. draw_point

4.1.1. 接口概述

在指定位置绘制一个像素点。

4.1.2. 函数原型

```
lcd.draw_point(x, y, color)
```

4.1.3. 参数描述

参数名	是否必填	数据类型	默认值	说明
x	是	int	-	X 坐标
y	是	int	-	Y 坐标
color	是	int	-	像素颜色（RGB565）

4.1.4. 返回值描述

无返回值。

4.1.5. 示例

```
#在(50, 50)位置绘制红色点
lcd.draw_point(50, 50, lcd.RED)
lcd.flush()
```

4.2. draw_line

4.2.1. 接口概述

绘制一条直线。

4.2.2. 函数原型

```
lcd.draw_line(x0, y0, x1, y1, color)
```

4.2.3. 参数描述

参数名	是否必填	数据类型	默认值	说明
x0	是	int	-	起点 X 坐标
y0	是	int	-	起点 Y 坐标
x1	是	int	-	终点 X 坐标
y1	是	int	-	终点 Y 坐标
color	是	int	-	线条颜色（RGB565）

4.2.4. 返回值描述

无返回值。

4.2.5. 示例

```
#从(10, 10)到(50, 100)绘制绿色直线
lcd.draw_line(10, 10, 50, 100, lcd.GREEN)
lcd.flush()
```

4.3. draw_rectangle

4.3.1. 接口概述

绘制矩形边框。

4.3.2. 函数原型

```
lcd.draw_rectangle(x0, y0, x1, y1, color)
```

4.3.3. 参数描述

参数名	是否必填	数据类型	默认值	说明
x0	是	int	-	左上角 X 坐标
y0	是	int	-	左上角 Y 坐标

x1	是	int	-	右下角 X 坐标
y1	是	int	-	右下角 Y 坐标
color	是	int	-	矩形边框颜色（RGB565）

4.3.4. 返回值描述

无返回值。

4.3.5. 示例

```
#绘制红色矩形边框
lcd.draw_rectangle(20, 20, 100, 80, lcd.RED)
lcd.flush()
```

4.4. draw_circle

4.4.1. 接口概述

绘制圆形边框。

4.4.2. 函数原型

```
lcd.draw_circle(x0, y0, r, color)
```

4.4.3. 参数描述

参数名	是否必填	数据类型	默认值	说明
x	是	int	-	圆心 X 坐标
y	是	int	-	圆心 Y 坐标
r	是	int	-	圆半径（像素）
color	是	int	-	圆形颜色（RGB565）

4.4.4. 返回值描述

无返回值。

4.4.5. 示例

```
#在屏幕中心绘制红色圆形
lcd.draw_circle(80, 64, 30, lcd.RED)
lcd.flush()
```

4.5. fill_rectangle

4.5.1. 接口概述

绘制填充矩形。

4.5.2. 函数原型

```
lcd.fill_rectangle(x, y, width, height, color)
```

4.5.3. 参数描述

参数名	是否必填	数据类型	默认值	说明
x	是	int	-	左上角 X 坐标
y	是	int	-	左上角 Y 坐标
width	是	int	-	矩形宽度
height	是	int	-	矩形高度
color	是	int	-	填充颜色（RGB565）

4.5.4. 返回值描述

无返回值。

4.5.5. 示例

```
#绘制蓝色填充矩形
lcd.fill_rectangle(20, 20, 100, 60, lcd.BLUE)
lcd.flush()
```

5 图像和文本显示函数

5.1. show_image

5.1.1. 接口概述

在指定位置显示图像。

5.1.2. 函数原型

```
lcd.show_image(x, y, width, height, image_data)
```

5.1.3. 参数描述

参数名	是否必填	数据类型	默认值	说明
x	是	int	-	图像左上角 X 坐标
y	是	int	-	图像左上角 Y 坐标
width	是	int	-	图像宽度
height	是	int	-	图像高度
image_data	是	bytearray	-	RGB565 格式的图像数据（每个像素 2 字节）

5.1.4. 返回值描述

无返回值。

5.1.5. 示例

```
import machine
from st7735 import LCD
#事先要准备好 RGB565 的图像数据
from images import QQ_ICON_40x40
spi = machine.SPI(1, baudrate=20000000, polarity=0, phase=0)
lcd = LCD(spi, dc_pin="F12", cs_pin="D14")
lcd.show_image(10, 10, 40, 40, QQ_ICON_40x40)
lcd.flush()
```

5.2. show_string

5.2.1. 接口概述

在指定位置显示字符串。

5.2.2. 函数原型

```
lcd.show_string(x, y, text, fc=WHITE, bc=BLACK)
```

5.2.3. 参数描述

参数名	是否必填	数据类型	默认值	说明
x	是	int	-	文本起始 X 坐标

y	是	int	-	文本起始 Y 坐标
text	是	string	-	要显示的文本
fc	否	int	WHITE	前景色（文字颜色）
bc	否	int	BLACK	背景色

5.2.4. 返回值描述

无返回值。

5.2.5. 示例

```
lcd.show_string(10, 10, "Hello World", lcd.WHITE)
lcd.flush()
```

6 颜色相关功能

6.1. COLOR565

6.1.1. 接口概述

将 24 位 RGB888 颜色值转换为 16 位 RGB565 格式。

6.1.2. 函数原型

```
color565 = lcd.COLOR565(r, g, b)
```

6.1.3. 参数描述

参数名	是否必填	数据类型	默认值	说明
r	是	int	-	红色分量（0-255）
g	是	int	-	绿色分量（0-255）
b	是	int	-	蓝色分量（0-255）

6.1.4. 返回值描述

返回 16 位 RGB565 颜色值。

6.1.5. 示例

```
pink = LCD.COLOR565(255, 192, 203) # 粉色
```

```
orange = LCD.COLOR565(255, 165, 0) # 橙色
purple = LCD.COLOR565(128, 0, 128) # 紫色
lcd.fill_screen(purple)
lcd.show_string(10, 10, "Hello", orange, lcd.WHITE)
lcd.flush()
```

6.2. 预定义颜色常量

模块预定义了以下常用颜色常量：

```
WHITE = 0xFFFF # 白色
BLACK = 0x0000 # 黑色
RED = 0xF800 # 红色
GREEN = 0x07E0 # 绿色
BLUE = 0x001F # 蓝色
YELLOW = 0xFFE0 # 黄色
CYAN = 0x7FFF # 青色
MAGENTA = 0xF81F # 品红色
GBLUE = 0X07FF # 青色
BROWN = 0XBC40 # 棕色
BRRED = 0XFC07 # 亮红/棕红色
GRAY = 0X8430 # 灰色
DARKBLUE = 0X01CF # 深蓝色
LIGHTBLUE = 0X7D7C # 浅蓝色
GRAYBLUE = 0X5458 # 灰蓝色
LIGHTGREEN = 0X841F # 浅绿色
LGRAY = 0XC618 # 浅灰色
LGRAYBLUE = 0XA651 # 浅灰蓝色
LBBLUE = 0X2B12 # 浅棕蓝色
```

7 注意事项

1. **显示刷新：**所有绘图操作都在缓冲区中进行，需要调用 `flush()` 方法才能显示到屏幕上
2. **坐标范围：**坐标必须保持在屏幕范围内 ($0 \leq x < \text{WIDTH}$, $0 \leq y < \text{HEIGHT}$)
3. **内存占用：**160x128 的 RGB565 缓冲区占用约 40KB 内存
4. **旋转影响：**屏幕旋转会影响坐标系统，请根据实际旋转方向调整坐标
5. **颜色格式：**所有颜色值均为 RGB565 格式，可使用预定义常量或 `COLOR565` 函数转换
6. **图像数据：**显示图像需要使用 RGB565 格式的数据，可通过提供的图像转换工具将 JPG/PNG 等格式图像转换为 RGB565 数据。

