



1.5inch RGB OLED Module

用户手册

产品概述

本产品是 1.5 寸彩色 OLED 显示屏，分辨率为 128*128，带有内部控制器，支持 4-wireSPI 和 3-wire SPI 进行控制。

产品特性

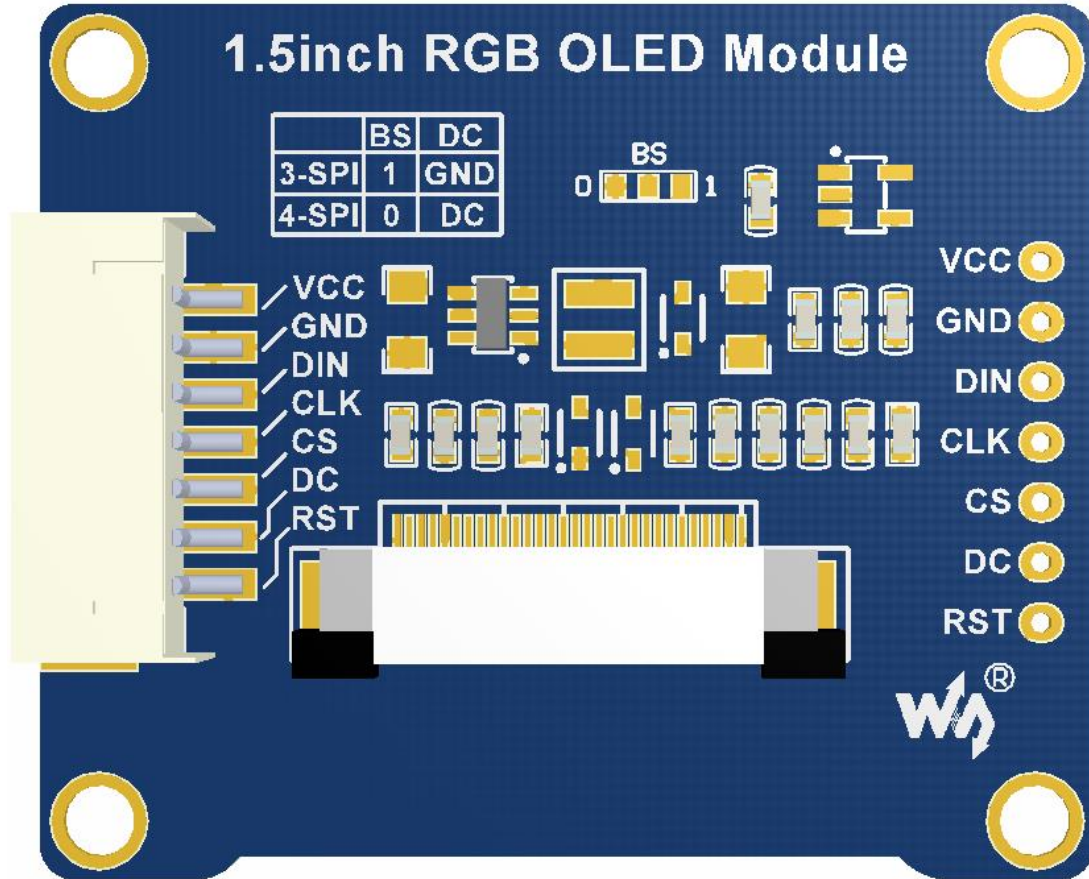
驱动芯片：SSD1351
支持接口：4-wire SPI、3-wire SPI
分辨率：128 * 128
显示尺寸：1.5inch
显示颜色：65K 彩色
工作电压：3.3V / 5V

管脚配置

功能引脚	描述
VCC	电源正
GND	电源地
DIN	数据输入
CLK	时钟输入
CS	片选信号
DC	数据/命令信号选择
RST	复位信号

硬件配置

本 OLED 模块提供两种通信方式：4-wire SPI 和 3wire-SPI，在模块的背面有一个可选择焊接的电阻，通过该电阻来选择通信方式。如下图：



模块出厂默认使用 4 线 SPI 通信，即 BS0 默认接 0

注：下表为接口连接。

	BS	CS	D/C	DIN	CLK
4-wire SPI	0	片选	D/C	MOSI	SCK
3-wire SPI	1	片选	需接地	MOSI	SCL

具体硬件配置，如下：

使用 4 线 SPI:即出厂设置,BS 接 0 连接到 GND, DIN 接控制引脚 MOSI, CLK 接控制引脚 SCK;

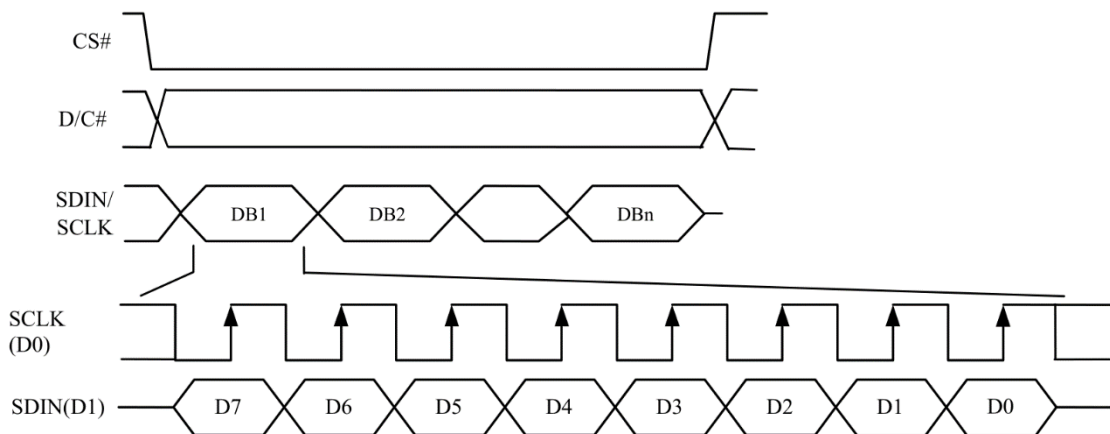
使用 3 线 SPI: BS 接 1 连接到 VCC, D/C 需接地, DIN 接控制引脚 MOSI, CLK 接控制引脚 SCK;

工作原理

SSD1351 是一款 128RGB * 128 Dot Matrix OLED/PLED 控制器，其内部有一个 128*128*18bit 的 SRAM 作为显示缓存区，支持 262k 和 65k 两种颜色深度。并支持 8bit 8080 并行、8bit 6800 并行、3wire-SPI 和 4wire-SPI 等通信方式。

本模块选择了 4wire-SPI 和 3wire-SPI 作为通信方式，减小模块的面积，同时节省了控制器的 IO 资源。

4WIRE-SPI 通信协议



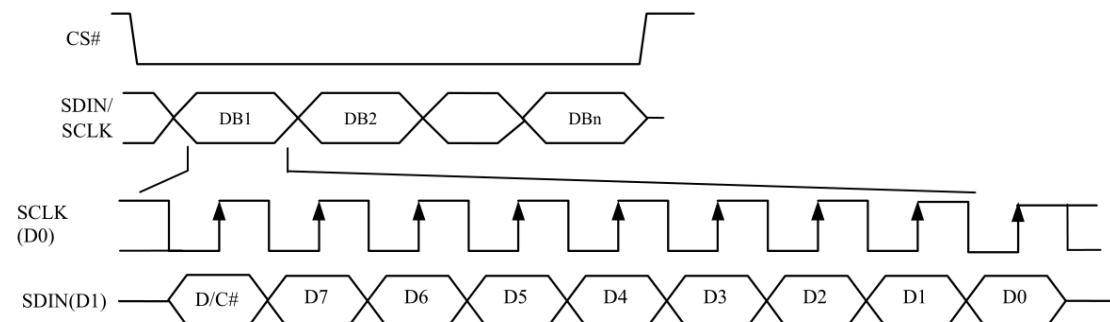
在 4wire-SPI 通信中，先将 DC 置 1 或置 0，再发送一个或多个字节的数据。

当 DC 置 1 时，发送的字节将作为显示数据存入 SSD1351 的 SRAM 中，作为显示数据。在发送数据时，可以连续发送多个字节。

当 DC 置 0 时，发送的字节将作为控制 OLED 的命令，发送命令时，一般一次只发送一个字节。

详见 SSD1351 Datasheet Figure 8-5。

3WIRE-SPI 通信协议



3wire-SPI 和 4wire-SPI 唯一的区别在于，它去掉了控制发送命令和数据的 DC 引脚，在每次 SPI 传输的字节前加一个位来识别该字节是命令还是数据。

故在 3wire-SPI 中，DC 引脚需要接地，此外，每次传输的数据不是 8bit，而是 9bit。

使用说明

该模块提供 STM32、Arduino 和 RaspberryPi 等例程，并实现常用屏幕操作功能：画点、画线、画矩形、画圆，可以控制它们的大小、宽度、填充。可以显示中英文，提供 2 种大小不同的字体。

STM32 的使用

1. 硬件配置

该例程使用的开发板为：XNUCLEO-F103RB

功能引脚	开发板
VCC	3V3/5V
GND	GND
DIN	D11(PA7)
CLK	D13(PA5)
CS	D10(PB6)
DC	D7(PA8)
RST	D8(PA9)

2. 工程文件说明

示例工程基于 MDK-ARM V5，使用 STM32CubeMX 配置生成。

工程目录/Src 下：

OLED_Driver.cpp：为 OLED 底层接口，主要提供了 OLED 的初始化，以及一些基本的显示和显示配置功能。

OLED_GFX.cpp：为 OLED 的上层应用，提供显示画线、圆，显示文字等函数。

工程目录/Inc 下：

ASCII_Font.h：为显示库，提供了 5*8、8*16 大小的英文字库，以及显示图片数据。

3. 现象

将 OLED 模块和开发板连接好之后，上电后烧录即可看到上述的功能。

ARDUINOD 的使用

1. 硬件配置

该例程使用的开发板为：UNO_PLUS

功能引脚	开发板
VCC	3V3/5V
GND	GND
DIN	D11
CLK	D13
CS	D10
DC	D7
RST	D8

2. 文件说明

工程目录下：

OLED_Driver.cpp：为 OLED 底层接口，主要提供了 OLED 的初始化，以及一些基本的显示和显示配置功能。

OLED_GFX.cpp：为 OLED 的上层应用，提供显示画线、圆，显示文字等函数。

ASCII_Font.h：为显示字库，提供了一组 5*7 大小的 OLED 显示数据。

3. 使用说明

将 OLED 模块和开发板连接好之后，上电后烧录即可看到上述的功能。

树莓派的使用

1. 硬件配置

该例程使用的开发板为：Raspberry Pi 3B

功能引脚	开发板
VCC	3V3/5V
GND	GND
DIN	MOSI
CLK	SCK
CS	CE0
DC	24(BCM)
RST	25(BCM)

2. 开启树莓派的 SPI

```
sudo raspi-config
```

选择 Interface Options -> SPI -> yes 启动 SPI 内核驱动

3. 库的安装

关于树莓派库的安装详细见微雪课堂：

<http://www.waveshare.net/study/article-742-1.html>

此处详细介绍了 wiringPi、bcm2835、python 的安装。

4. 使用

把对应的例程拷贝进树莓派中（可通过 samba 或者直接复制到 SD）即可，以下示例均复制到了树莓派 pi 用户目录下。

python 程序

运行 ls 命令，可见如下文件

```
pi@raspberrypi:~/oled_test $ ls
cambriab.ttf main.py OLED_Driver.py picture1.jpg picture2.jpg picture3.jpg
```

其中：

cambriab.ttf：为字体文件，用于显示字体。

mian.py：为主函数，使用 Image 库实现各种画图和显示文本功能。

运行程序：sudo python main.py

**注：有些树莓派可能没有安装 image 库，运行：sudo apt-get install python-imaging
安装 python-imaging 库**

Image 是 python 图像处理库，任何一副图像都是用一个 Image 对象表示，因此可以通过 new 方法来创建一张空白图片，大小与 OLED 的最大显示范围一致，通过 Draw 库来进行画图，最后把这个图片传递到 OLED 上，需要注意的是请创建一个 RGB 图：image = Image.new("RGB", (128, 128), "BLACK")

对于 python 而言：不需要像其他的控制一样，把图片转换成数组，直接可以使用 open()方法：打开图片，即可直接显示，同时注意图片大小不超过屏幕最大显示范围。