



# 0.91inch OLED Module

## 用户手册

### 产品概述

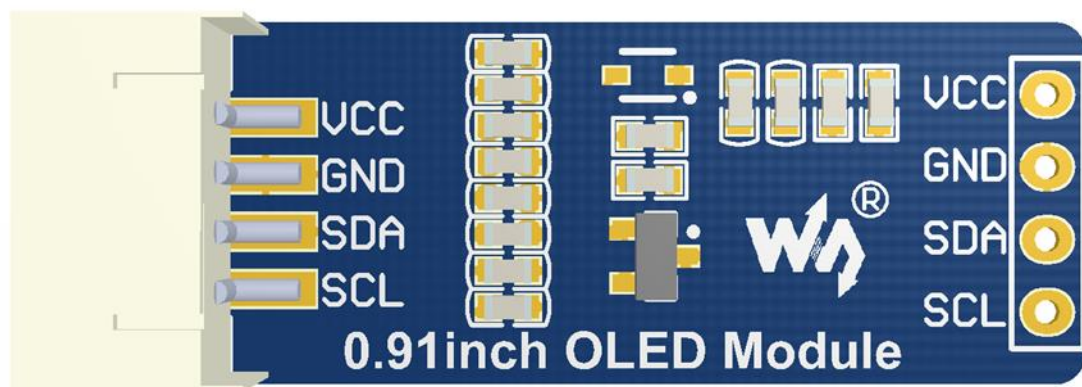
本产品是 0.91 寸 OLED 显示屏，分辨率为 128\*32，带有内部控制器，可使用 I2C 进行控制。

### 产品特性

驱动芯片：SSD1306  
支持接口：I2C  
分辨率：128 \* 32  
显示尺寸：1.5inch  
显示颜色：白色  
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
SDA	数据输入
SCL	时钟输入

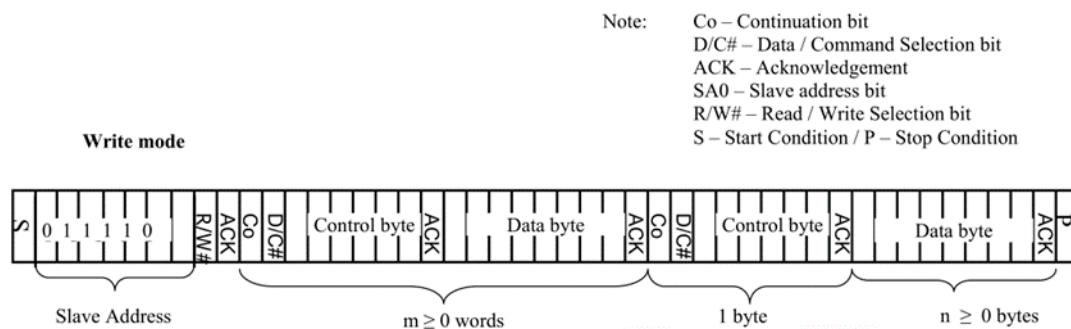


## 工作原理

SSD1306 是一款 128\*64 像素的 OLED 控制器，但该 OLED 只有 128\*32 个像素点，所以该屏只用到了 SSD1306 缓存区的前半部分。

该 OLED 支持 8bit 8080 并行、8bit 6800 并行、三线 SPI、四线 SPI 和 I2C 等通信方式，但考虑到该模块的尺寸，以及节省微控制器宝贵的 IO 资源，故舍弃了前四种通信方式，只支持 I2C 的通信方式。

## I2C 通信协议



在 I2C 通信时，先发送一个 7bit 的从设备地址+1bit 的读写位，等待设备的响应。

在从设备应答后，接着发送一个控制字节，该字节决定了后面发送的字节是命令还是数据，然后再等待从设备应答。

在从设备再次应答之后，若发送命令，则只发送一字节的命令。若发送数据，可以只发送一个字节，也可以多个字节的数据连着发送，视情况而定。

详见 Datasheet Page20 Figure 8-7。

## 使用说明

该模块提供 STM32、Arduino、树莓派例程，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，可以显示字符串。

## 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
SDA	SDA /D14
SCL	SCL /D15

## 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：

Adafruit\_SSD1306.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。

Adafruit\_GFX.cpp: 为 OLED 的上层应用，提供了显示，画线，圆，显示文字等函数。

glcdfont.h: 为显示字库，提供了 6\*8、8\*16 大小的英文字库

## 3. 现象

将 OLED 模块和开发板连接好之后，上电后烧录到开发板，即可看到上述的功能。

## 树莓派使用

### 1. 硬件连接

功能引脚	开发板
VCC	3V3/5V
GND	GND
SDA	SDA
SCL	SCL

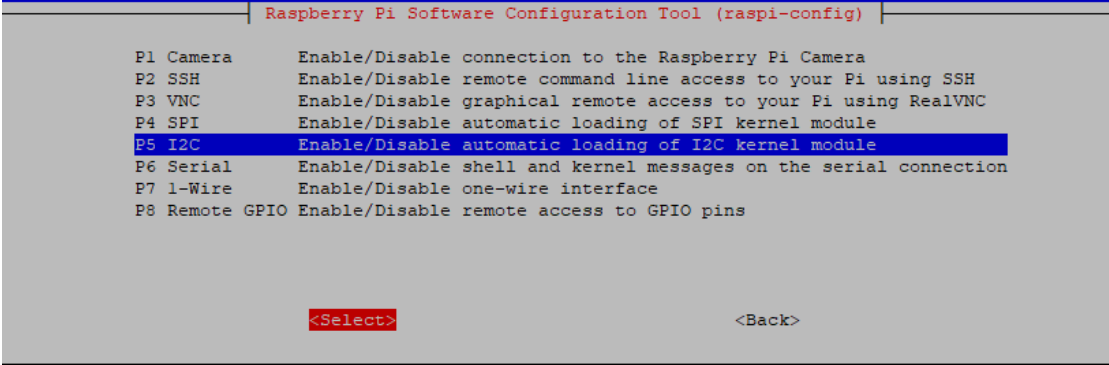
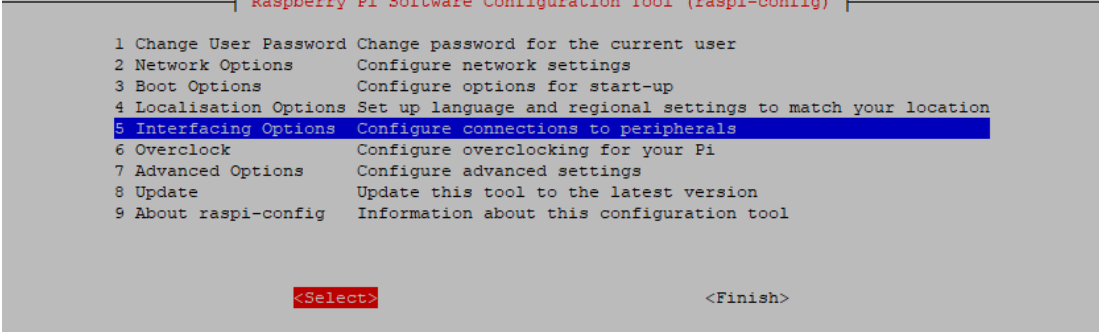


## 2. 开启树莓派外设功能

`sudo raspi-config`

选择 Interfacing Options -> I2C -> yes 启动 I2C 内核驱动:

```
pi@raspberrypi: ~  
pi@raspberrypi:~ $ sudo raspi-config
```



## 3. 库的安装

关于树莓派函数库的安装详见微雪课堂:

<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。

### 4.1 BCM2835 使用

(1) 安装 bcm2835 函数库

(2) 运行 ls 命令，可见如下文件

```
pi@raspberrypi:~/0in91/bcm2835 $ ls
bin  Fonts  Makefile  obj  oled_0in91
```

其中:

bin 文件夹中: 为项目生成的.o 文件, 通常情况下我们是不需要管的;

Fonts 文件夹: 为 5 中常见的字体;

Obj 文件夹: 为项目文件, 其中有 main.c, OLED\_Driver.c.及.h , OLED\_Config.c 及.h, 以及 OLED\_GUI.c 及.h, 还有一个定义图片的头文件 Show\_Pic.h;

main.c: 为主函数, 不用过多说明。

OLED\_Config.c: 为硬件底层接口, 定义了树莓派的管脚及通信方式;

OLED\_Driver.c: 为硬件驱动, 即 OLED 的驱动, 通常情况下是不要修改的;

OLED\_GUI.c: 为上层应用, 提供常用的画点、线、图、显示字、显示图片等函数。

Show\_Pic.h: 为显示的图片, 图片需使用数据水平, 字节垂直扫描生成的数组才能正常显示, 在后面会具体介绍;

Makefile 为工程的编译规则, 您若更改了代码则先执行 `make clear`, 清除全部文件依赖以及生成的可执行文件, 然后再执行 `make`, 这样 `makefile` 就会自动化编译整个项目从而生成可执行文件;

oled\_0in91 为可执行文件, 通过 `make` 命令生成。

介绍完了文件及目录, 就该说明使用, 一如既往:

您只需要执行 `sudo ./oled_0in91` 即可。

## 4.2 WiringPi 使用

(1) 安装 wiringPi 函数库

(2) 运行 ls 命令, 可见如下文件

```
pi@raspberrypi:~/0in91/wiringpi $ ls
bin  Fonts  Makefile  obj  oled_0in91
```

WiringPi 与 BCM2835 文件目录相同, 两种方法的区别在于两点:

第一: WiringPi 是通过读写 linux 系统的设备文件操作, 而 bcm2835 则是树莓派 cpu 芯片的库函数, 操作的为寄存器, 因此若先使用了 BCM2835 库, WiringPi 则会使用失败, 此时重启系统可以运行;

第二：由于第一个区别的原因，他们底层的配置就不一样，在 DEV\_Config.c 中使用的为 WiringPi 及其相应的 wiringPiSPI 来提供底层接口。

同样，只需运行 `sudo ./oled_0in91` 即可。

### 4.3 Python

运行 ls 命令，可见如下文件

```
pi@raspberrypi:~/0in91/python $ ls
Adafruit_Python_SSD1306 stats.py
```

本 python 例程使用 Adafruit 库。

为了保证正常使用，安装以下库：

```
sudo apt-get update
```

```
sudo apt-get install build-essential python-dev python-pip
```

```
sudo pip install RPi.GPIO
```

```
sudo apt-get install python-imaging python-smbus
```

进入 python 示例程序目录，安装并执行：

```
sudo python Adafruit_Python_SSD1306/setup.py install
```

```
sudo python stats.py
```

### 4.4 设置开机自动启动运行

通过配置/etc/rc.local，使代码在树莓派启动时运行。

运行：

```
sudo vim /etc/rc.local
```

在 exit 0 前加上：

```
sudo python /home/pi/0in91/python/stats.py &
```

需注意的是：/home/pi/python/demo.py 为您放置例程所在目录位置，可以通过命令：pwd 来获取。

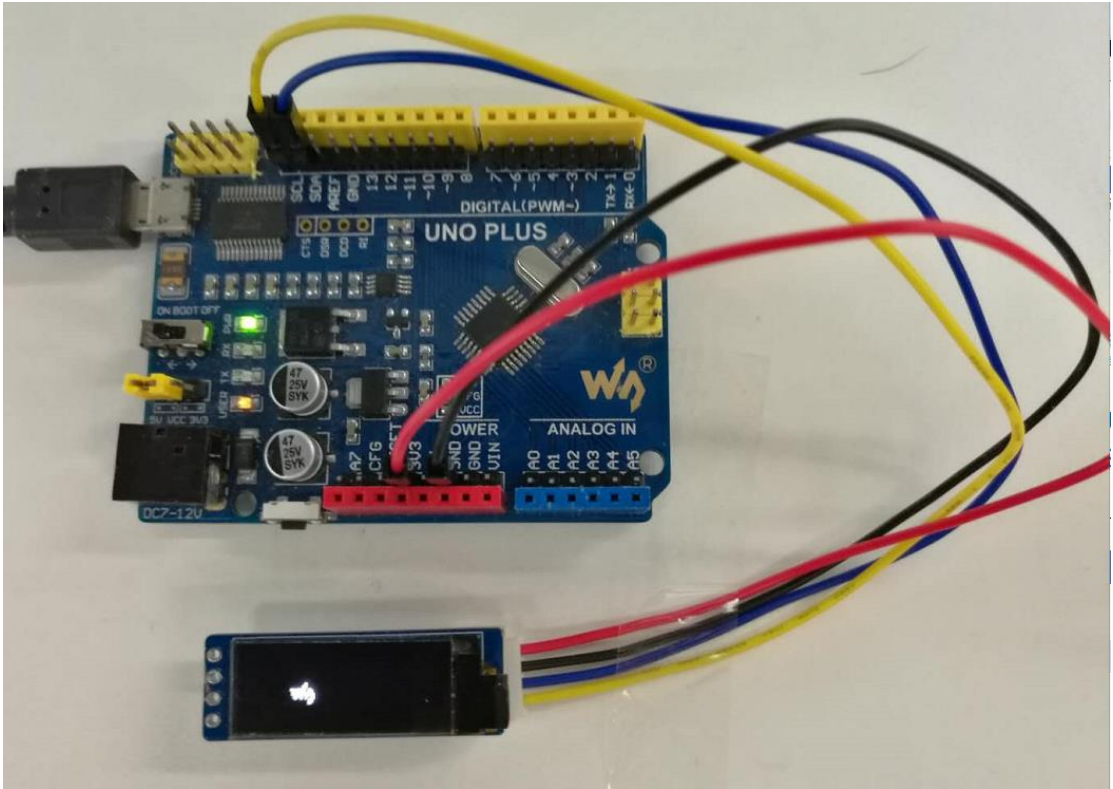
还有务必在结尾加上&，否则您可能需要重装系统（无法通过 ctrl+c 终止进程，无法登陆到树莓派用户下）。

## ARDUINO 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
SDA	SDA
SCL	SCL



### 2. 工程文件说明

工程目录/oled:

oled.ino : Arduino 工程文件。直接双击可以打开工程

工程目录:

Adafruit\_SSD1306.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。

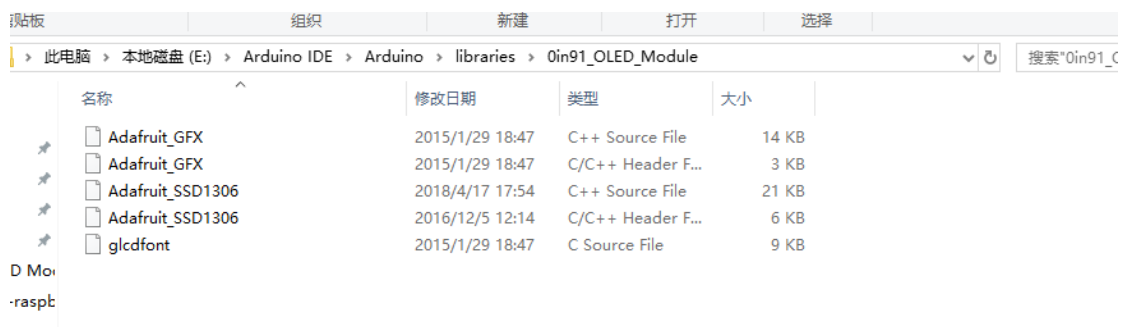
Adafruit\_GFX.cpp: 为 OLED 的上层应用, 提供显示画线、圆, 显示文字等函数。

Adafruit\_SSD1306.h、Adafruit\_GFX.h: 头文件

glcdfont.h: 为显示字库, 提供了 6\*8、8\*16 大小的英文字库。

### 3. 运行示例程序

运行示例程序之前, 需要先将工程目录/下的文件, 复制到 Arduino IDE 安装目录下的 libraries 文件夹里面。注意: 不能直接放置在 libraries 里面, 需要新建一个文件放置, 比如新建文件夹 0in91\_OLED\_Module:



然后打开工程文件 oled.ino, 选择正确开发板和串口, 将程序下载到开发板运行

### 数组制作

请使用 Image2Lcd 打开黑白图片, 并配置如下:

输出数据类型: C 语言数据(\*.c)

扫描方式: 数据水平, 字节垂直

输出灰度: 单色

最大宽度和高度: 128 32 (这两个为 OLED 的最大分辨率)

然后再勾选下面的字节内像素数据反序即可。



