

关于 7pin 0.96 寸 OLED 模块使用 IIC 接口的操作说明

7pin 0.96 寸 OLED 模块支持 SPI 和 IIC 接口；默认是 SPI 接口；如果想用 IIC 接口；操作如下几步骤：

1. 将模块背面的电阻 R3 换到 R1 位置，此时将模块切换为 IIC 接口；电阻 R8 可以用 0 欧姆电阻或是用焊锡短接电阻两端。
2. CS 脚接地。
3. DC 脚的处理：在 IIC 通信中 DC 的高低电平是用来选择 IIC 通信地址的；当 DC 接地时 IIC 从机地址为:0x78,当 DC 接高电平时 IIC 地址为 0x7A；测试程序中所用的为 0x78；也就是说大家需要将 DC 接地
4. 关于 RES 的处理。RES 这个脚是 OLED 屏的复位脚；大家在用 OLED 屏的时候会发现；所有 OLED 本身都会有一个复位脚；因为 OLED 在被操作之前需要在将寄存作一次复位；然后才能对屏进行初始货操作；否则 OLED 可能会出现水稳定的情况。

RES 处理办法：

- 1> 简单的验证办法：将 RES 接电源正；这样可以把屏点亮；但是会不稳定，在快速测试时可以这么操作
- 2> 将 RES 脚与开发板的复位脚连接；通过开发板的复位来对 OLED 进行复位
- 3> 通过一个 IO 脚来对 OLED 进行复位，这个操作放在对屏初始化之前；先将 RES 拉低延迟 200ms 左右；然后再拉高一直处于高电平状态
- 4> 通过一个 RC 复位电路来控制 RES

5.DO 为 IIC 时钟线，D1 为 IIC 数据线

## 5. 软件测试

软件这块 51，stm32 在资料包里已经有了；就不再说。说一下 arduino 的测试在这里给大家提供一下 U8glib 这个库的使用；这个库通用性比较好；一般不会有问题。具体操作方法：

1 将库解压后放到库目录下

2 打开解压后的文件在 U8glib\examples 目录下有很多程序；这些程序我们都可以使用；下面以 U8glib\examples\U8gLogo 下的 U8gLogo.ino 来说明；其它的操作一样。打开 U8gLogo.ino 文件，选择 U8GLIB\_SSD1306\_128X64 u8g(U8G\_I2C\_OPT\_NONE); // I2C / TWI

```
//U8GLIB_DOGXL160_2X_GR u8g(13, 11, 10, 9); // SPI Com: SCK = 13, MOSI = 11, CS = 10, AO = 9
//U8GLIB_PCD8544 u8g(13, 11, 10, 9, 8); // SPI Com: SCK = 13, MOSI = 11, CS = 10, AO = 9, Reset = 8
//U8GLIB_PCF8812 u8g(13, 11, 10, 9, 8); // SPI Com: SCK = 13, MOSI = 11, CS = 10, AO = 9, Reset = 8
//U8GLIB_KS0108_128 u8g(8, 9, 10, 11, 4, 5, 6, 7, 18, 14, 15, 17, 16); // 8Bit Com: DO..D7: 8,9,10,11,4,5,6,7 en=18, c
//U8GLIB_LC7981_160X80 u8g(8, 9, 10, 11, 4, 5, 6, 7, 18, 14, 15, 17, 16); // 8Bit Com: DO..D7: 8,9,10,11,4,5,6,7 en=18, c
//U8GLIB_LC7981_240X64 u8g(8, 9, 10, 11, 4, 5, 6, 7, 18, 14, 15, 17, 16); // 8Bit Com: DO..D7: 8,9,10,11,4,5,6,7 en=18, c
//U8GLIB_LC7981_240X128 u8g(8, 9, 10, 11, 4, 5, 6, 7, 18, 14, 15, 17, 16); // 8Bit Com: DO..D7: 8,9,10,11,4,5,6,7 en=18, c
//U8GLIB_ILI9325D_320x240 u8g(18,17,19,U8G_PIN_NONE,16 ); // 8Bit Com: DO..D7: 0,1,2,3,4,5,6,7 en=wr=18,
//U8GLIB_SBN1661_122X32 u8g(8,9,10,11,4,5,6,7,14,15, 17, U8G_PIN_NONE, 16); // 8Bit Com: DO..D7: 8,9,10,11,4,5,6,7 cs1=14,
//U8GLIB_SSD1306_128X64 u8g(13, 11, 10, 9); // SW SPI Com: SCK = 13, MOSI = 11, CS = 10, AO = 9
//U8GLIB_SSD1306_128X64 u8g(10, 9); // HW SPI Com: SCK = 13 and MOSI = 11
//U8GLIB_SSD1306_128X64 u8g(U8G_I2C_OPT_NONE); // I2C / TWI
//U8GLIB_SSD1306_128X64 u8g(U8G_I2C_OPT_NO_ACK); // Display which does not send ACK
//U8GLIB_SSD1306_ADAFRUIT_128X64 u8g(13, 11, 10, 9); // SW SPI Com: SCK = 13, MOSI = 11, CS = 10, AO = 9
//U8GLIB_SSD1306_ADAFRUIT_128X64 u8g(10, 9); // HW SPI Com: CS = 10, AO = 9 (Hardware Pins are SCK = 13 and MOSI = 11)
//U8GLIB_SSD1306_128X32 u8g(13, 11, 10, 9); // SW SPI Com: SCK = 13, MOSI = 11, CS = 10, AO = 9
//U8GLIB_SSD1306_128X32 u8g(10, 9); // HW SPI Com: CS = 10, AO = 9 (Hardware Pins are SCK = 13 and MOSI = 11)
//U8GLIB_SSD1306_128X32 u8g(U8G_I2C_OPT_NONE); // I2C / TWI
//U8GLIB_SH1106_128X64 u8g(13, 11, 10, 9); // SW SPI Com: SCK = 13, MOSI = 11, CS = 10, AO = 9
//U8GLIB_SH1106_128X64 u8g(U8G_I2C_OPT_NONE); // I2C / TWI
//U8GLIB_SH1106_128X64 u8g(U8G_I2C_OPT_NO_ACK); // Display which does not send ACK
```

将前面的//去了，一次只能选一个

将前面的//去了；如果用是其它控制 IC 的屏；直接选上前面的型号就可以了；这里我们

选 SSD1306 的；因为 0.96 是 SSD1306IC

然后用 arduino 开发板的 iic 接口与 OLED 对接就可以了

**以 Arduino UNO R3 为例 A4 接 D1（数据线），A5 接 D0（时钟线）**