

全动电子 0.96 寸 OLED 显示模块使用说明



深圳市全动电子技术有限公司

2018.06.09

目录

1 OLED 简介	3
2 产品特点	3
3 接口说明	3
3.1 裸屏接口说明	3
3.2 OLED 模块接口说明	4
3.3 OLED 驱动 IC 介绍	4
4 OLED 模块使用说明	5
5 软件平台移植注意事项	6
5.1 C51 代码说明	6
5.2 STM32 代码说明	7
5.3 Arduino 代码说明	8
6 取模软件使用说明	9
6.1 字符取模	9
6.2 图形取模	11

1、OLED 简介

OLED 即有机发光二极管 (Organic Light-Emitting Diode, OLED)。OLED 显示技术具有自发光、广视角、几乎无穷高的对比度、较低耗电、极高反应速度、可用于挠曲性面板、使用温度范围广、构造及制程较简单等优点，被认为是下一代的平面显示器新兴应用技术。

OLED 显示和传统的 LCD 显示不同，其可以自发光，所以不需要背光灯，这使得 OLED 显示屏相对于 LCD 显示屏尺寸更薄，同时显示效果更优。但是由于目前技术限制，OLED 使用寿命比 LCD 短，尺寸难以大型化，生产成本比较高，良品率低，难以大规模量产。

2、产品特点

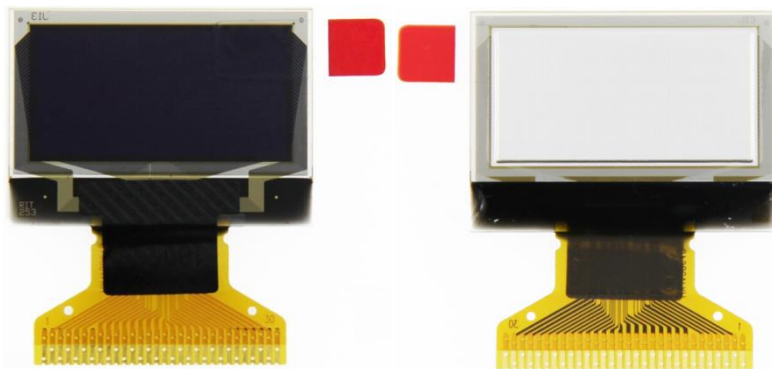
全动电子 0.96 寸的 OLED 显示模块具有以下特点：

- 1、支持多种单片机平台（包含 C51、STM32 以及 Arduino）；
- 2、黄蓝、白、蓝三种颜色显示方案可选，固定区域显示固定颜色，没点亮区域为黑色，点亮区域则为黄蓝、白、蓝三种颜色方案之一；
- 3、显示内容清晰、对比度高、视场角广；
- 4、分辨率为 128x64；
- 5、支持多种接口方式：6800、8080 两种并口、3 线或 4 线 SPI 接口以及 IIC 接口（本模块使用 IIC 接口），这五种接口是通过屏上的 BS0~BS2 来配置的；
- 6、提供丰富的测试程序，方便客户使用（包含 C51、STM32 以及 Arduino 平台）；

3、接口说明

3.1 裸屏接口说明

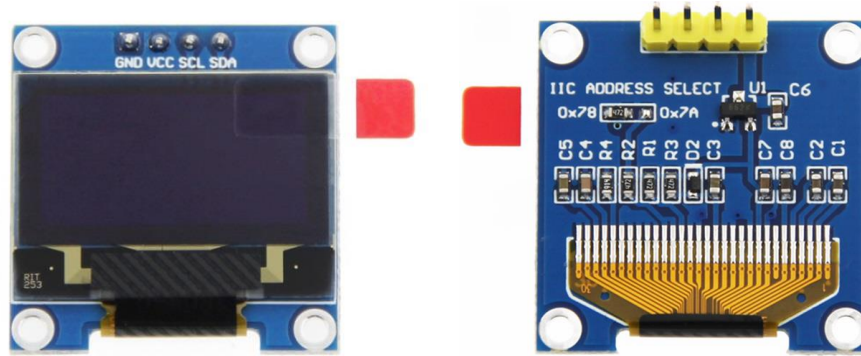
全动电子 0.96 寸 OLED 裸屏外观图如下：



裸屏为 30pin，从屏正面看左下角为 1，右下角为 30，在设计的时候一定要注意不要搞反了。具体的接口定义请大家查看 0.96 寸 OLED 官方数据手册；里面有详细介绍。

3.2 OLED 模块接口说明

本模块使用 IIC 接口，模块图如下：



GND: 电源负极（接地）

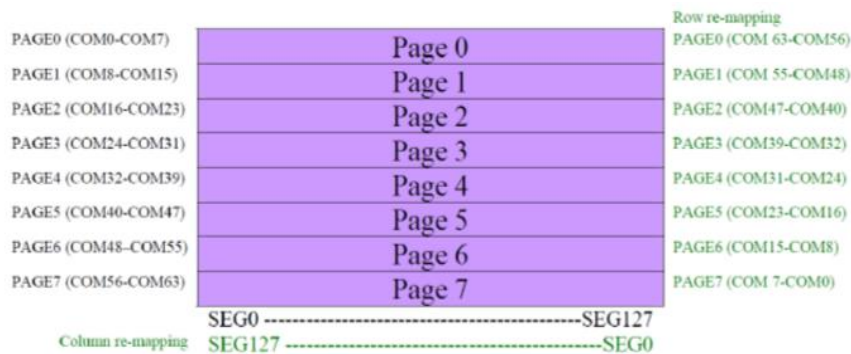
VCC: 电源正极（接 5V 电源）

SCL: IIC 时钟信号线，对应 OLED 的 D0 脚（不同的平台，接口定义不一样，需要根据定义接线）

SDA: IIC 数据信号线，对应 OLED 的 D1 脚（不同的平台，接口定义不一样，需要根据定义接线）

3.3 OLED 驱动 IC 介绍

本 OLED 显示屏使用 SSD1306 驱动 IC。SSD1306 是一个单片 CMOS OLED/PLED 驱动芯片，可以驱动有机/聚合发光二极管点阵图形显示系统。该芯片由 128segments 和 64Commons 组成，专为共阴极 OLED 面板设计。SSD1306 的图形显示数据 RAM 大小为 128x64，刚好对应屏幕分辨率的大小。RAM 大小被分成 8 页（PAGE0~PAGE7），每一页有 8 行，128 列。当往 RAM 里面写数据时，先要设置页地址，然后分别设置数据的高 4 位和低 4 位地址，每次都是写入一列数据，一页写完时，需要写入 128x8bit 数据。关于 SSD1306 其他方面的详细介绍，请查阅 SSD1306 的数据手册。



SSD1306 的图形显示数据 RAM 的页架构图

4、OLED 模块使用说明

通过查看 SSD1306 的数据手册可以得知该 IC 支持四线和三线 SPI、IIC 接口以及 8bit 的并口连接。由于并口连接方式占用的数据线较多，接线麻烦，所以不太常用。关于通信接口方式怎么选择，这是通过 BS0、BS1 以及 BS2 这三个管脚来配置的，具体定义如下图：

10	BS0	I	Communicating Protocol Select These pins are MCU interface selection input. See the following table:			
11	BS1			BS0	BS1	BS2
12	BS2		I ² C	0	1	0
			3-wire SPI	1	0	0
			4-wire SPI	0	0	0
			8-bit 68XX Parallel	0	0	1
			8-bit 80XX Parallel	0	1	1

SSD1306 通信接口定义说明图

本模块使用 IIC 通信接口，只需要接 4 根线就可以完成 OLED 屏数据通信。



GND：电源负极（接地）

VCC：电源正极（接 5V 电源）

SCL：IIC 时钟信号线

SDA：IIC 数据信号线

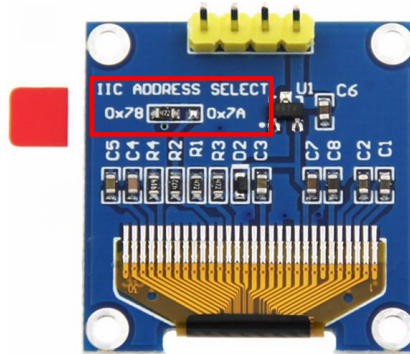
VCC 接到单片机的 5V 电源引脚上，GND 接到 GND 引脚上，SCL 和 SDA 需要根据不同的平台引脚定义来接线：

C51 和 STM32 一般使用模拟 IIC，就是自己定义两个引脚作为 SCL 和 SDA。

Arduino 使用硬件 IIC，就是芯片本来就定义好了 SCL 和 SDA 引脚，不需要我们自己再重新定义。

接好线上电后，可以运行我们提供的测试程序进行测试。

IIC 的从设备地址选择需要模块电路上进行修改，如图：



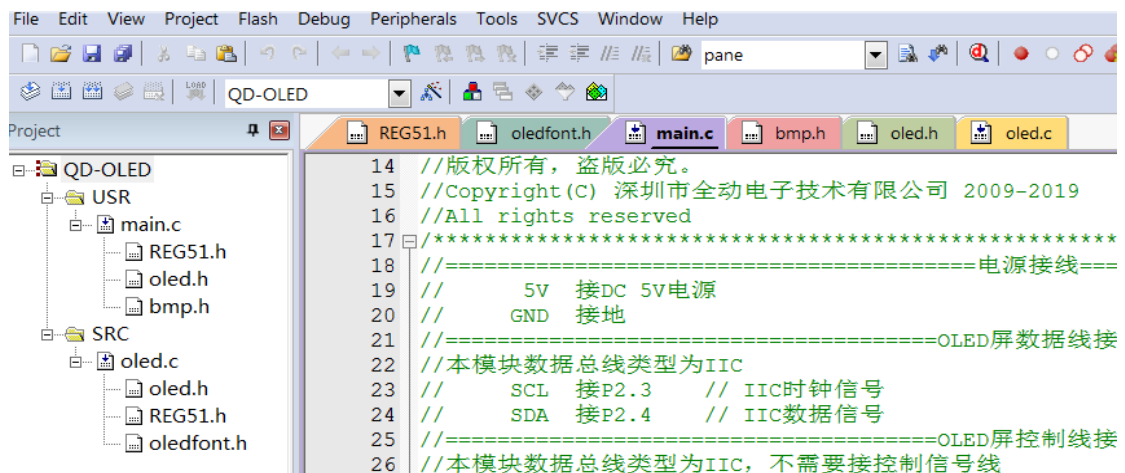
如图红色矩形框内，如果电阻接在 0x78 这边，则表示 IIC 从设备地址为 0x78, 如果接在 0x7A 这边，则表示 IIC 从设备地址为 0x7A。如果两边电阻都接上，则表示 IIC 有两个从设备，相当于接了两个 OLED 屏。

5、软件平台移植注意事项

本模块提供了三个平台的测试程序，分别是 C51、STM32 和 Arduino，每个平台的测试程序大致由三部分组成：平台相关的代码、OLED 操作相关的代码以及 IIC 操作相关的代码。除了平台相关的代码因平台而异，其他两部分的代码原理和实现方法都差不多，和所用的平台没有关系。在同一平台上，代码可以直接移植，只需要修改相关的引脚定义就可以正常运行了。

5.1 C51 代码说明

打开 C51 代码工程，发现有如下文件：



REG51.h 是和平台相关的代码，里面定义了一些平台相关的参数。

oledfont.h、bmp.h 都存放图片和汉字取模后的点阵数组。

oled.h 存放的是和 OLED 屏相关的一些参数，包括引脚定义。

oled.c 存放的是和 OLED 屏操作相关的一些函数，包括 IIC 的读写，OLED 屏数据写入等
main.c 则是主程序操作了。

通过以上描述我们可以知道，只需要修改 oled.h 文件里的引脚定义即可。

打开 oled.h 文件，找到以下内容：

```
sbit OLED_SCL=P2^3;//时钟 D0 (SCLK)
sbit OLED_SDIN=P2^4;//D1 (MOSI) 数据
```

只要修改 OLED_SCL 和 OLED_SDIN 的值就可以了。

这里是将 P23 引脚定义为时钟信号引脚，将 P24 引脚定义为数据信号引脚。

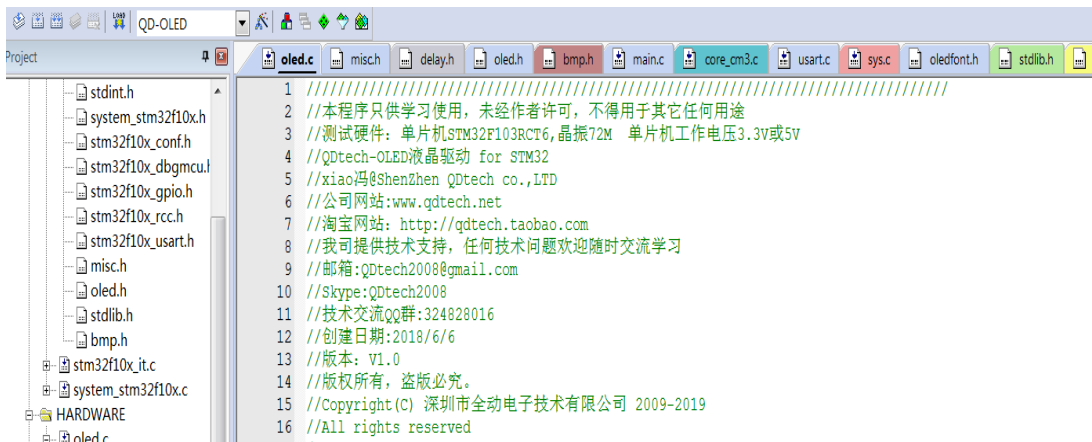
再找到如下内容，可以修改 IIC 从设备地址：

```
#define IIC_SLAVE_ADDR 0x78 //IIC slave device address
```

该 OLED 模块 IIC 地址为 0x78。

5.2 STM32 代码说明

打开 STM32 代码，发现有以下文件：



除了 main.c、oled.c、oled.h、bmp.h、oledfont.h，其他都是和平台相关的代码。

oledfont.h、bmp.h 都存放图片和汉字取模后的点阵数组。

oled.h 存放的是和 OLED 屏相关的一些参数，包括引脚定义。

oled.c 存放的是和 OLED 屏操作相关的一些函数，包括 IIC 的读写，OLED 屏数据写入等
main.c 则是主程序操作了。

由于 STM32 比 C51 多了 GPIO 口初始化操作，所以移植代码需要修改 oled.h 和 oled.c 两个文件。

打开 oled.h 文件，找到如下内容：


```
//-----OLED IIC端口定义-----

#define OLED_SCLK_Clr() GPIO_ResetBits(GPIOB,GPIO_Pin_13)//CLK
#define OLED_SCLK_Set() GPIO_SetBits(GPIOB,GPIO_Pin_13)

#define OLED_SDIN_Clr() GPIO_ResetBits(GPIOB,GPIO_Pin_14)//DIN
#define OLED_SDIN_Set() GPIO_SetBits(GPIOB,GPIO_Pin_14)
```

根据自己定义，修改 GPIO_ResetBits 和 GPIO_SetBits 的 GPIO 组和 pin 脚。

这里将 GPIOB 的 13 号引脚定义为时钟信号引脚，将 GPIOB 的 14 号引脚定义为数据信号引脚。

再找到如下内容，可以修改 IIC 从设备地址：

```
#define IIC_SLAVE_ADDR 0x78 //IIC slave device address
```

该 OLED 模块 IIC 地址为 0x78。

打开 oled.c 文件，找到如下内容：

```
GPIO_InitTypeDef GPIO_InitStructure;

RCC_APB2PeriphClockCmd(RCC_APB2Periph_GPIOB, ENABLE); //使能B端口时钟
GPIO_InitStructure.GPIO_Pin = GPIO_Pin_13|GPIO_Pin_14;
GPIO_InitStructure.GPIO_Mode = GPIO_Mode_Out_PP; //推挽输出
GPIO_InitStructure.GPIO_Speed = GPIO_Speed_50MHz; //速度50MHz
GPIO_Init(GPIOB, &GPIO_InitStructure); //初始化GPIOB3,4
GPIO_SetBits(GPIOB,GPIO_Pin_13|GPIO_Pin_14);
```

分别对定义的引脚进行输出模式、速率以及置位设置。

5.3 Arduino 代码说明

Arduino 使用硬件 IIC，芯片内部已经对 IIC 引脚做了定义，因此软件上不需要再对 IIC 引脚进行定义，只是不同型号的单片机，IIC 引脚定义不一样，需要在接线上做调整。

由于软件使用了 U8glib 开源库，所以初始化时，需要根据 OLED 屏驱动 IC 的型号选择正确的类。

例如本模块使用了 U8GLIB_SSD1306_128X64 u8g(U8G_I2C_OPT_NONE)进行初始化。

如果要修改 IIC 从设备地址，则需要打开 u8g_com_arduino_ssd_i2c.c 文件，找到如下内容：

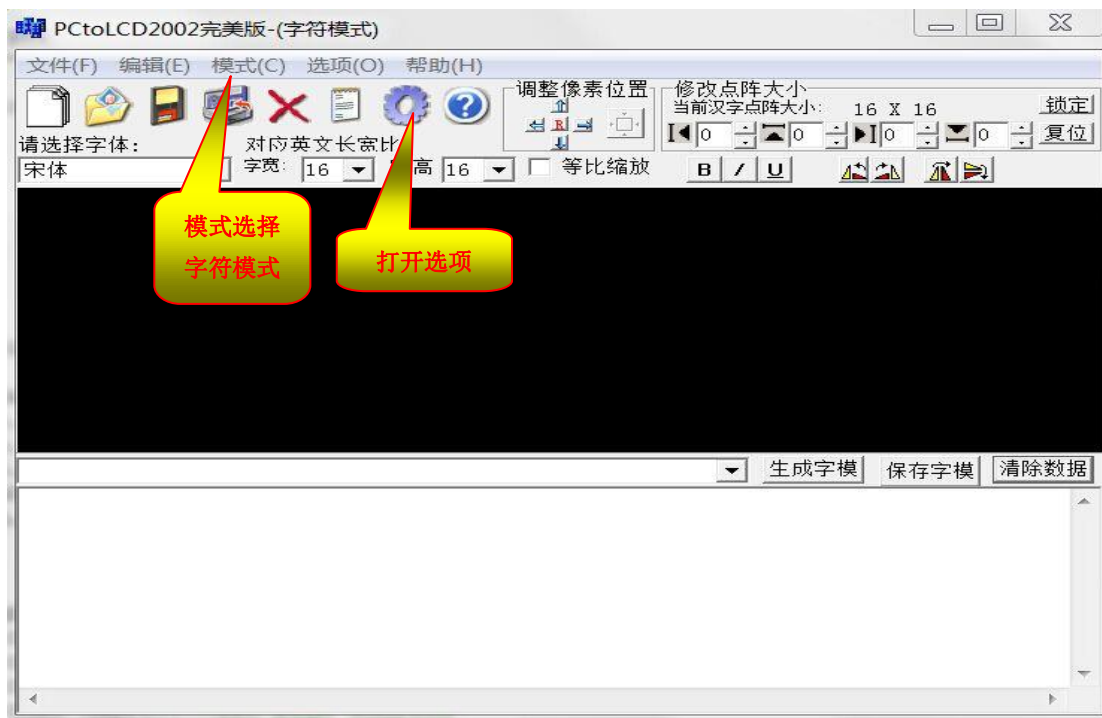
```
#define I2C_SLA (0x3c*2)
```

该 OLED 模块的 IIC 地址为 0x3c*2(0x78)

6、取模软件使用说明

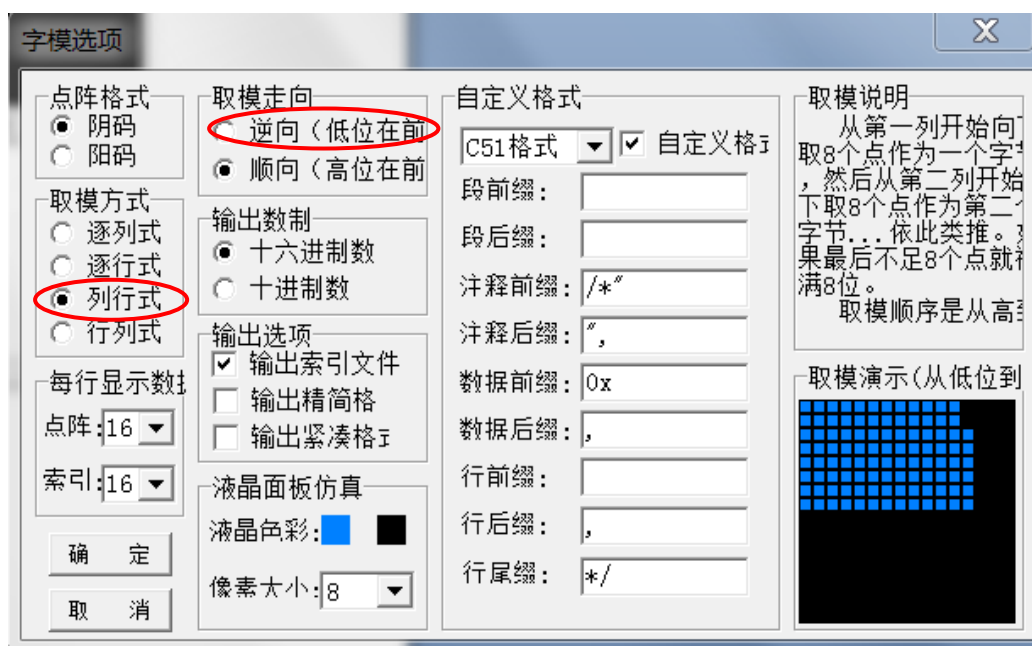
6.1 字符取模

打开 PCtoLCD2002 取模软件->选择模式->字符模式，如下图所示

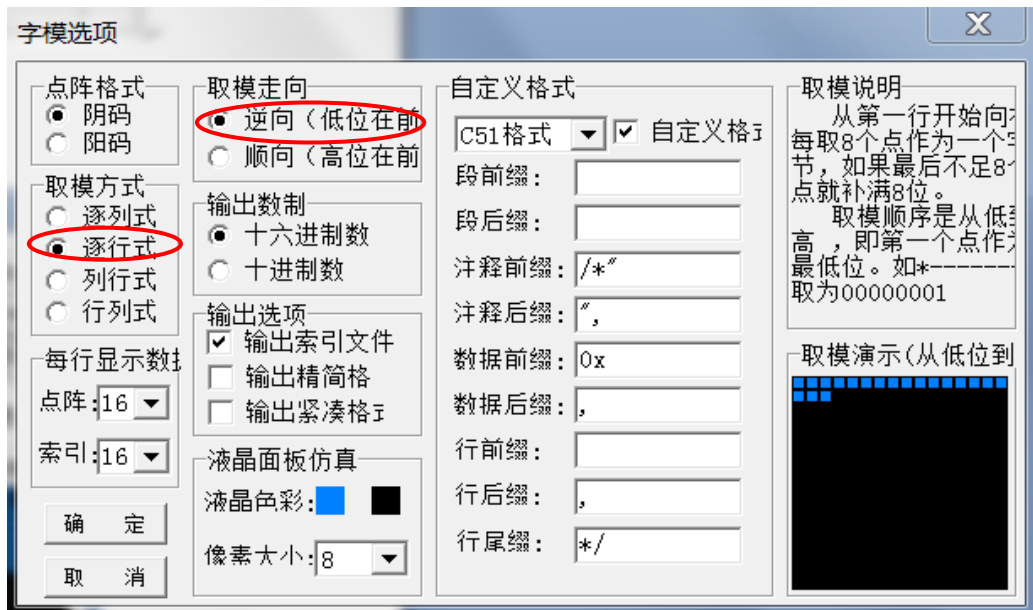


接下来选择选项按钮，配置好了，选择确认

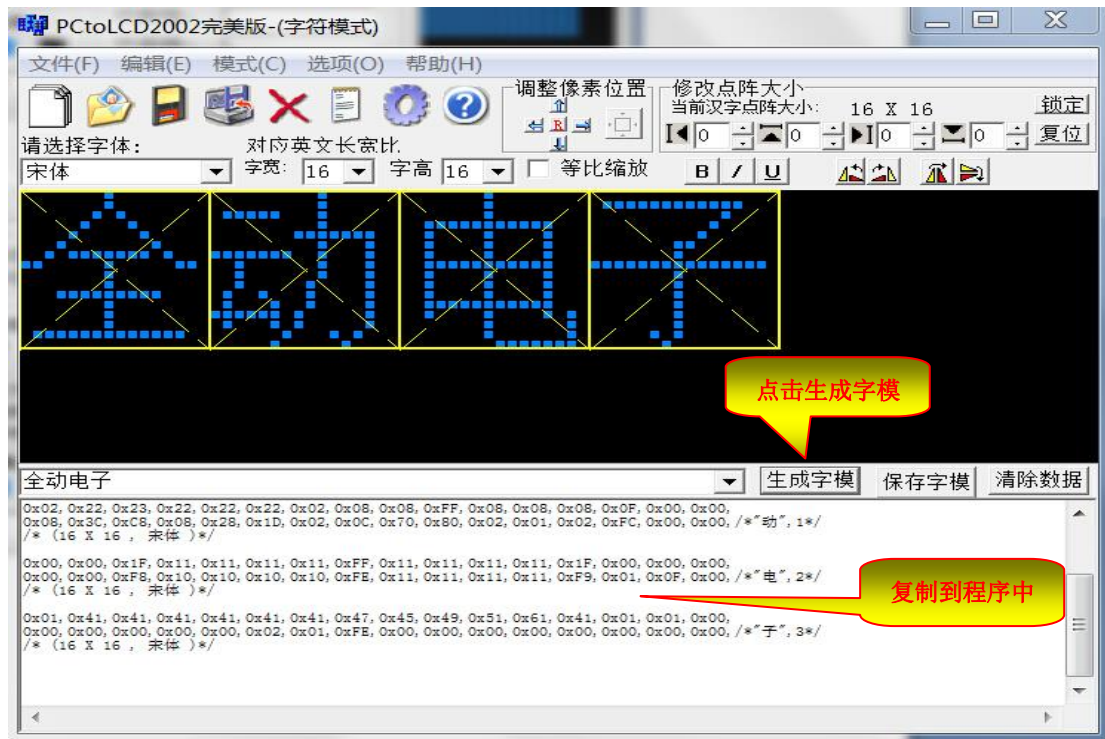
C51 和 STM32 平台按如下设置（选择逆向和行列式）：



Arduino 平台按如下设置（选择逆向和逐行式）：

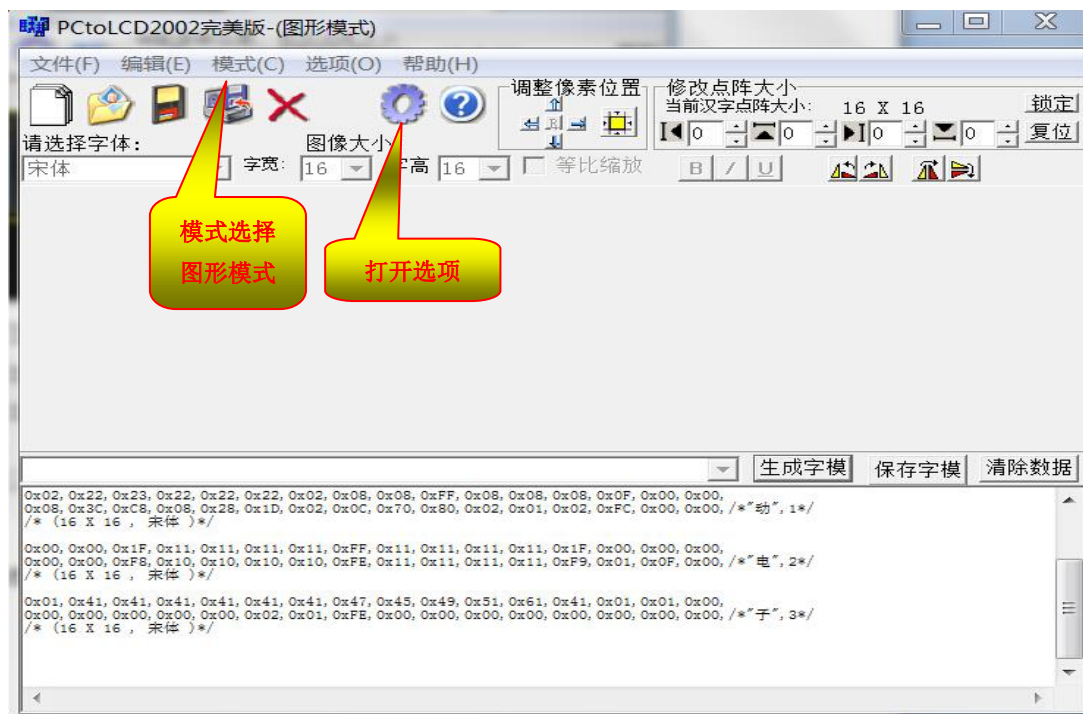


接下来输入你想要显示的文字，例如输入“全动电子”，然后点击生成字模，将生成的字模数据复制到程序中，写入 OLED 就可以显示汉字：



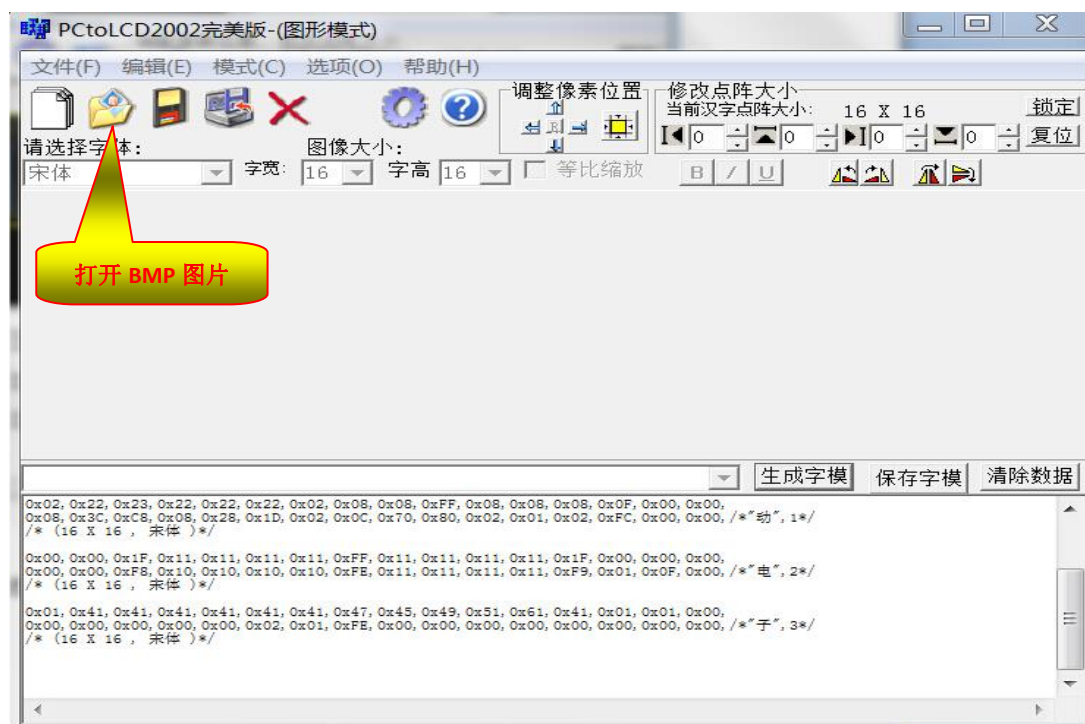
6.2 图形取模

打开 PCtoLCD2002 取模软件->选择模式->图形模式，如下图所示：



打开选项，配置设置和文字取模一致。

然后点击文件->打开或者直接点击文件按钮，打开一个 BMP 单色图片



软件会显示一个 OLED 屏幕点阵图，再点击生成字模，将生成好的数据拷贝到程序中，写入 OLED 屏，就可以显示图片了：

