

字库串口液晶模块控制板 FOR 320x240 点阵

SLCM320240

功能简介:

1. 驱动 320x240 点阵液晶显示模块, 单 5V 供电, LED 背光电源共享液晶电源(保证供电电流 300mA)
2. 国标 GB_2312 二级字库, 标准 ASCII 码, 编码调用
3. 可显示 16*16/32*32 点阵字体
4. 两种接口形式: UART 串行接口(默认 TTL 电平)和普通 IO 三线通信(TTL 电平)

电源及控制接口:

PWR (DC5V)

+	-
+5V	0V

接口一: UART 串口

1	2	3
GND	RX	TX
公共端	接收端	发送端

接口二: 普通 IO 串口

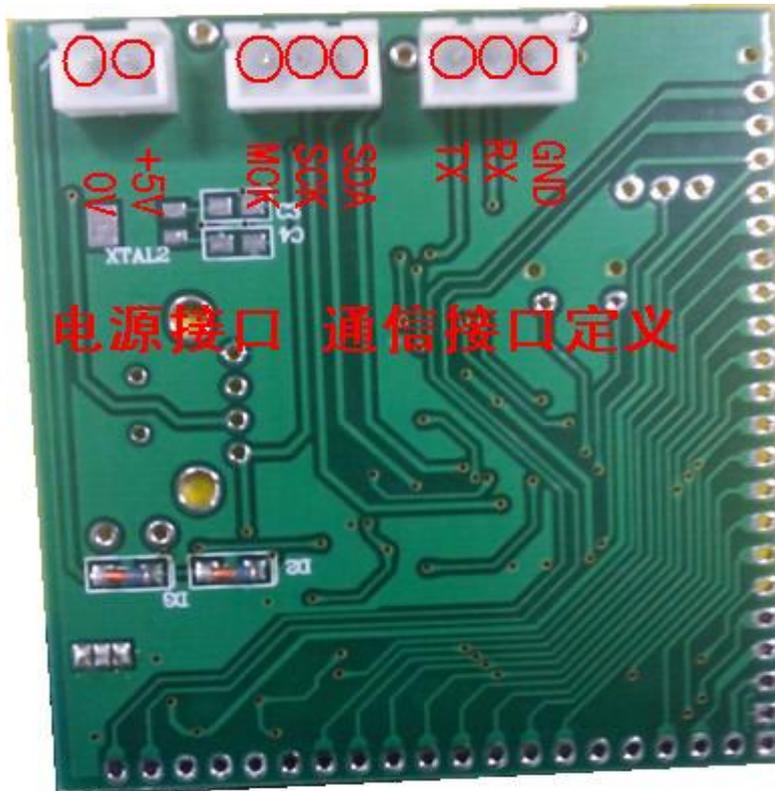
1	2	3
SDA	SCK	MCK
串行数 据出入	液晶握 手信号	用户握 手信号

说明: UART 接口和普通 IO 串口在控制板上是两个相同的三芯座, 在使用的时候可选择其中之一。其中的 UART 可选 RS232 电平和 TTL 电平, 出厂之后这种电平不能再更改!

见下图说明:

SLCM320240 串行接口及电源接口

SLCM 的电源及控制接口:



SLCM 选购型号说明:

SLCM xxxxx AA BB C DD

SLCM-串口液晶标志

xxxxxx-5~6 位数字,表示液晶的分辨率 12864/19264/24064/240128/320240

AA-两位字母表示屏和背光颜色 蓝屏 B 黄绿屏 G 黑屏 F /白光 W 蓝光 B 黄光 Y

BB-两位数字表示字库点阵 01 表示 12*12/24*24 点阵,02 表示 16*16/32*32 点阵

C-接口方式 A-UART B-USB C-三 IO

DD-简繁体字库工作电压: 简体 S 繁体 T 3V: 3.0~3.6 5: 5V

例: SLCM320240BW02CS5 表示 320*240 分辨率蓝屏 16*16/32*32 三线 IO 接口简体 5V

注: 320x240 点阵只提供 16*16 点阵汉字库

SLCM 指令:

表 1 SLCM 命令说明: (下表命令在某些分辨率下可能无效, 请注意后面的说明)

控制命令	参数	功能说明
???	无	查询模块的状态,模块准备好就回传字符"OK"到主机(UART),模块收到此命令后,在显示屏最后一行显示波特率等提示
asd	英文串	在当前位置起显示一串英文字符,最多显示到行尾(地址不改变)
bls	a	无参数时为开或关背光(即发送一次改变状态) 带参数 a=0 时为打开背光,a=1 时关背光
brt	a	重新设置波特率(a=0-8),设置新的波特率后,主机也要重新设置 a=0 2400 a=1 4800 a=2 7200 a=3 9600 a=4 14400 a=5 19200 a=6 38400 a=7 57600 a=8 115200 ;UART/USB 接口有效 ;如果 a>8 则不做处理
chr	abc	显示内码为 ab 的字符,地址自动加 1.c=0 显示普通字,c=1 显示大字(对于英文字母 a=0)
cir	abcd	画圆.原点(a,b),半径 c.d=0 消圆, d=1 画圆
	abcde	画圆.原点(a,b,c),半径 d.e=0 消圆, e=1 画圆(320*240)
cln	abcd	清除从点(a,b)到点(c,d)的一直线
	abcdef	清除从点(ab,c)到点(de,f)的一直线(320*240)
clr	无	清屏.清屏后,显示坐标也回复到起点(左上角)
clw	ablh	清除以(a,b)为起点的 l 个字符宽,h 个字符高的窗口内容
cpl	ablh	反显以(a,b)为起点的 l 个字符宽,h 个字符高的窗口内容
dat	abc	在坐标(a,b)处显示 c(主要用来输入图形或用户自定义文字数据)
fhz	abcde	在点坐标(b,c)处显示内码为 de 的文字,a=0 显示普通字,a=1 显示大字
	abcdef	在点坐标(bc,d)处显示内码为 ef 的文字,a=0 显示普通字,a=1 显示大字
dot	abc	在(a,b)处显示一个点 c,c=0 则为清除点,c=1 则是画点
	abcd	在(ab,c)处显示一个点 d,d=0 则为清除点,d=1 则是画点
lin	abcd	从点(a,b)到点(c,d)画一直线
	abcdef	从点(ab,c)到点(de,f) 画一直线(320*240)
loc	abc	设置(a,b)为显示坐标,c=1 则显示光标块,在进行 str,chr 命令前需先发送此命令
res	无	液晶复位。复位不影响显示内容!
sad	a	选择显示区,a=0~7 共 8 个显示区,第 8 个显示区为文本区,可显示 40*16 个 5*8 点阵的英文字符。默认 0 区, 每个显示区 240*128 个点 (此指令仅 T6963 控制的 240*128 有效)
str	a 串	显示字符串,a=0 显示正常字,a=1 显示大字;串长度<=100 字符(50 汉字),字符串以'\0'结束.

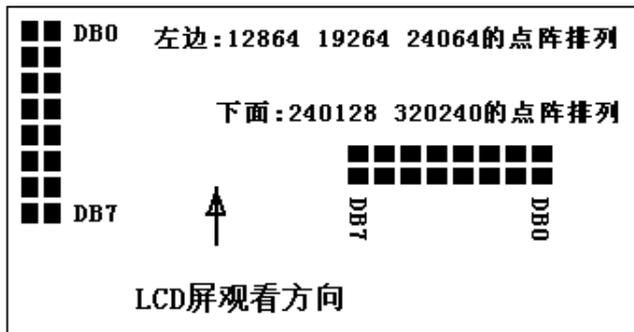
注: 显示大字符时

如果当前为 8*16/16*16 点阵标准字形,则大字符就是 16*32/32*32

注意事项:

1. RS232/UART 串行口模式设为模式 1 (1 个起始位, 8 个数据位, 1 个停止位);

2. RS232/UART 波特率设为 19200（默认是 19200, 可通过命令修改）
3. 串口发送的必须是完整命令的 16 进制格式；
4. 在每条命令末尾还要加上命令结束码“0x0D”，以表明该命令结束。
5. 液晶接收到一个正确的指令并处理完毕后，会返回一个固定值：0xff。
液晶处理指令需要一定时间, 所以请在接收到返回的 0xff 后再发下一条指令！
6. 请注意 320*240 点阵的屏暂无 12*12/24*24 点阵字库
7. 关于 dat 指令输入图形数据的说明:
12864 19264 24064 三种的 dat 数据排列是按点列字节行进行排列的, 数据高位在下, 低位在上; 而 240128 320240 的排列则是按点行字节列进行排列的, 数据的高位在左, 低位在右, 如下图:



附上测试程序及三线 IO 串口通讯协议

```
#include <reg51.h>
#include <intrins.h>
sbit MCLK=P1^0;
sbit SCLK=P1^1;
sbit SDA =P1^2;

unsigned char bdata DATA;
sbit BIT0=DATA^0;
// 发送一个数据到模块
send_byte(unsigned char ch) //三线 IO 通讯协议:发送一字节数据
{
    unsigned char i;
    if(!MCLK)return; //MCLK=0 时不发送数据直接返回,对此有可能会造成此次发送数据的丢失!
    //如果发送不成功,请自行处理数据重发
    MCLK=0; //申请通讯
    DATA=ch;
    for(i=4;i!=0;i--)
    {
        while(SCLK); //检测从机响应
        SDA=BIT0; //放一个 bit 到 SDA
        MCLK=1;
        DATA=DATA>>1; //准备下一个 bit
    }
}
```

```

        while(!SCLK); //检测从机响应
            SDA=BIT0; //放一个 bit 到 SDA
            MCLK=0;
            DATA=DATA>>1; //准备下一个 bit
        }
        while(SCLK); //检测从机收完
        MCLK=1;
        while(!SCLK); //等待从机准备好
        SDA=1;
    }

//显示字符串
lcd_string(bit char_bit, unsigned char *pst)
{
    send_byte('s');
    send_byte('t');
    send_byte('r');
    send_byte(char_bit);
    while(*pst!=0)
        send_byte(*pst++);
    send_byte(0x0d); //串口发送结束标志
}

//显示直线
dis_line(unsigned int x1,unsigned char y1,unsigned int x2,unsigned char y2)
{
    send_byte('l');
    send_byte('i');
    send_byte('n');
    send_byte(x1>>8);
    send_byte(x1);
    send_byte(y1);
    send_byte(x2>>8);
    send_byte(x2);
    send_byte(y2);
    send_byte(0x0d);
}

//清除直线
del_line(unsigned int x1,unsigned char y1,unsigned int x2,unsigned char y2)
{
    send_byte('c');
    send_byte('l');
    send_byte('n');
    send_byte(x1>>8);
    send_byte(x1);
    send_byte(y1);
    send_byte(x2>>8);

```

```

    send_byte(x2);
    send_byte(y2);
    send_byte(0x0d);
}
clr_lcd()    //
{
    //清屏-清空所有显示 RAM 的内容,如果仅清空当前显示窗口的请使用 clw 命令
    send_byte('c');
    send_byte('l');
    send_byte('r');
    send_byte(0x0d);
}
////////////////////////////////////
main()
{
    unsigned int n;
    unsigned char i;
    for(n=0;n<1000;n++); //wait...
    while(!MCLK); //MCLK=0 wait...
    while(1)
    {
        clr_lcd(); //清屏
        for(n=0;n<10000;n++);
        //查状态    建议只在调试的时候用
        send_byte('?');
        send_byte('?');
        send_byte('?');
        send_byte(0x0d);
        for(n=0;n<60000;n++);

        clr_lcd(); //清屏
        for(i=0;i<96;i++) //显示所有的 ascii 字符
        {
            send_byte('c');
            send_byte('h');
            send_byte('r');
            send_byte(0x00);
            send_byte(i+0x20);
            send_byte(0x00); //小字体
            send_byte(0x0d);
        }
        for(n=0;n<60000;n++);

        clr_lcd();

        //显示一个串 1

```

```

send_byte('l');
send_byte('o');
send_byte('c');
send_byte(2);
Send_byte(0);
send_byte(0);
send_byte(0x0d); //定位字符串起始地址
lcd_string(0,"Wayeah LCM ABC123 中文 a 混 B 排 D 液晶显示器。");
for(n=0;n<60000;n++);

//显示一个串 2
lcd_string(0,"三串口线,UART 串口,带 USB 转换!");
for(n=0;n<60000;n++);

//显示一个串 2
lcd_string(0,"三 0 线串口,UART");
for(n=0;n<60000;n++);

clr_lcd();
//显示一个串 3
send_byte('l'); //没指定显示地址则从上次的结束地址开始
send_byte('o');
send_byte('c');
send_byte(0);
Send_byte(0);
send_byte(1);
send_byte(0x0d); //定位字符串起始地址
lcd_string(1,"深圳炜烨");
for(n=0;n<60000;n++);
for(n=0;n<60000;n++);
//显示一线段
dis_line(0,10,117,33);
for(n=0;n<60000;n++);

//清除一线段
del_line(0,10,117,33);
for(n=0;n<60000;n++);

//close backlight
// send_byte('b');
// send_byte('l');
// send_byte('s');
// send_byte(0x0d);
// for(n=0;n<30000;n++);

//open backlight

```

```

//      send_byte('b');
//      send_byte('l');
//      send_byte('s');
//      send_byte(0x0d);
//      for(n=0;n<30000;n++);

//反显一个区域:0 列开始反显 12 个字符位置
//          1 行开始反显两行
//注意:
send_byte('c');
send_byte('p');
send_byte('l');
send_byte(0);
send_byte(1);
send_byte(12);
send_byte(2);
send_byte(0x0d);
for(n=0;n<60000;n++);

//清除一个区域:1 列开始反显 12 个字符位置
//          1 行开始清除两行
send_byte('c');
send_byte('l');
send_byte('w'); //clr 指令
send_byte(1); //开始清除的坐标
send_byte(1);
send_byte(12); //清除的宽度-12 个字符位置
send_byte(2); //清除的高度-两行
send_byte(0x0d);
for(n=0;n<60000;n++);
}
}

```

以下为使用 C51 UART 串口通信的设置:

//串口初始化 晶振为 11.0592M 方式 1 波特率 19200, THTL=253

```

void InitCom(unsigned char BaudRate)
{
    unsigned char THTL=253;
    SCON = 0x50; //串口方式 1,允许接收
    TMOD = 0x20; //定时器 1 定时方式 2
    TCON = 0x40; //设定定时器 1 开始计数
    TH1 = THTL;
    TL1 = THTL;
    PCON = 0x80; //波特率加倍控制,SMOD 位
    RI = 0; //清收发标志
    TI = 0;
}

```

```

TR1 = 1; //启动定时器
}
//UART 串口输出一个字节（非中断方式）
void ComOutChar(unsigned char OutData) //用此代替 send_byte(unsigned char ch)可使用 UART 串口
{
    SBUF = OutData;
    while(!TI); //空语句判断字符是否发完
    TI = 0; //清 TI
}

```

320*240 上显示 16*16/32*32 点阵文字和显示圆



液晶模块的尺寸图:

