

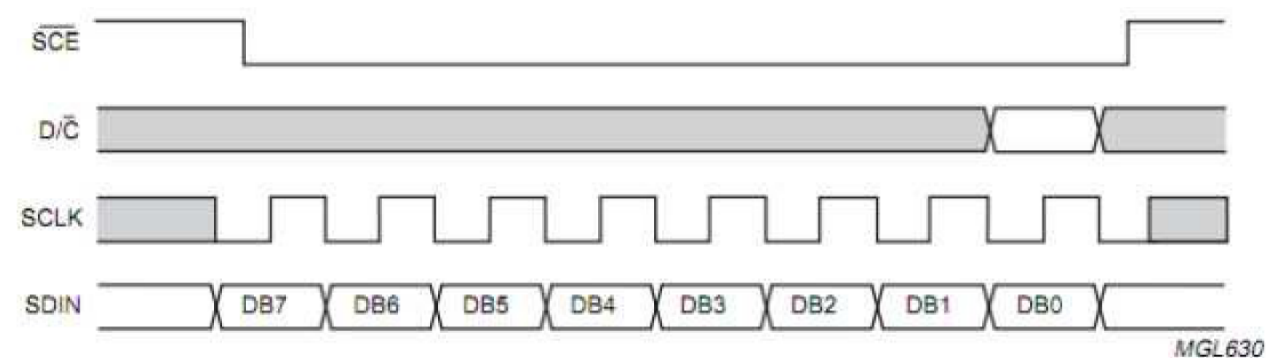
Nokia5110 液晶屏完全新手学习笔记(二)

2 人收藏此文章, [我要收藏](#) 发表于 1 年前, 已有 2690 次阅读 共 1 个评论

四、时序图

没错, 要向液晶屏写入数据, 我们需要通过模拟一个串行总线协议来写入数据。

先来看看传送一个字节 (指令) 的时序图:



时序图1 串行总线协议—传送1个字节

如果能看懂时这些序图的兄弟, 就不要听我废话啦。请飘过~~

从时序图 1 看出:

1. SCE 片选为 0 时开始发送数据。
2. 时钟信号 SCLK 仅在 SCE 片选为 0 时有效。
3. SDIN 数据输入, 需要在时钟信号 SCLK 有效时开始发送数据, 且在 SCLK 的正边缘取样, 注意, 数据是从高位开始发送的。

知道了传送数据的时序，我们需要使用程序来模拟这个时序，从而发送指令控制液晶屏。

接下来，看看实例。

五、程序实例

代码一：

```
/*-----  
LCD_write_byte: 使用 SPI 接口写数据到 LCD  
输入参数: dt: 写入的数据;  
command : 写数据/命令选择;  
编写日期: 20080918  
-----*/  
  
void LCD_write_byte(unsigned char dt, unsigned char command)  
{  
    unsigned char i;  
    sce=0;  
    dc=command;  
    for(i=0;i<8;i++)  
    {  
        if(dt&0x80)  
            sdin=1;  
        else
```

```

        sdin=0;

        dt=dt<<1;

        sclk=0;

        sclk=1;

    }

    dc=1;

    sce=1;

    sdin=1;

}

```

代码分析：

这个是一个写入一个字节数据的模拟时序函数，具有两个参数，`dt` 为要写入的字节，`command` 为数据/命令选择。

首先，需要 `sce` 拉低电平，开始发送数据。

`i` 循环 8 次，写入 8 位数据，现以“设置扩展指令集”（即写入 `0x21`）

全过程如下：

- `i = 0 : dt = 0x21 = 0010 0001 b`

`dt & 0x80 :`

`0010 0000`

`1000 0000 &`

`0000 0000 = 0` , 则 `sdin = 0 (DB7)` ,

`dt` 左移 1 位 , `dt<<1 = 0100 0010 b` ,

模拟时钟信号 `sclk = 0` , `sclk = 1` ,

以使采样

- $i = 1: dt = 0100\ 0010b$

$dt \ \& \ 0x80:$

0100 0010

1000 0000 &

0000 0000 = 0, 则 $sdin = 0$ (DB6),

dt 左移 1 位, $dt \ll 1 = 1000\ 0100\ b$,

模拟时钟信号 $sclk = 0$, $sclk = 1$, 以使

采样

- $i = 2: dt = 1000\ 0100b$

$dt \ \& \ 0x80:$

1000 0100

1000 0000 &

1000 0000 $\neq 0$, 则 $sdin = 1$ (DB5),

dt 左移 1 位, $dt \ll 1 = 0000\ 1000\ b$,

模拟时钟信号 $sclk = 0$, $sclk = 1$, 以使

采样

- $i = 3: dt = 0000\ 1000b$

$dt \ \& \ 0x80:$

0000 1000

1000 0000 &

0000 0000 = 0, 则 $sdin = 0$ (DB4),

dt 左移 1 位, $dt \ll 1 = 0001\ 0000\ b$,

模拟时钟信号 $sclk = 0$, $sclk = 1$, 以使

采样

- $i = 4 : dt = 0001\ 0000b$

dt & 0x80:

0001 0000

1000 0000 &

0000 0000 = 0, 则 $sdin = 0$ (DB3),

dt 左移 1 位, $dt \ll 1 = 0010\ 0000\ b$,

模拟时钟信号 $sclk = 0$, $sclk = 1$, 以使

采样

- $i = 5 : dt = 0010\ 0000b$

dt & 0x80:

0010 0000

1000 0000 &

0000 0000 = 0, 则 $sdin = 0$ (DB2),

dt 左移 1 位, $dt \ll 1 = 0100\ 0000\ b$,

模拟时钟信号 $sclk = 0$, $sclk = 1$, 以使

采样

- $i = 6 : dt = 0100\ 0000b$

$dt \ \& \ 0x80 :$

$0100\ 0000$

$1000\ 0000 \ \&$

$0000\ 0000 = 0$, 则 $sdin = 0$ (DB1),

dt 左移 1 位 , $dt \ll 1 = 1000\ 0000\ b$,

模拟时钟信号 $sclk = 0$, $sclk = 1$, 以使

采样

- $i = 7 : dt = 1000\ 0000b$

$dt \ \& \ 0x80 :$

$1000\ 0000$

$1000\ 0000 \ \&$

$1000\ 0000 \neq 0$, 则 $sdin = 1$ (DB0),

dt 左移 1 位 , $dt \ll 1 = 0000\ 0000\ b$,

模拟时钟信号 $sclk = 0$, $sclk = 1$, 以使

采样

过程很罗嗦 , 不过很简单 , 呵呵~~

可见,这样循环 8 次,写入了 8 位数, $DB7 \sim DB0 = 0010\ 0001b = 0x20$,正是“扩展指令集”的指令值。

代码二：

```
/*-----  
LCD_init: 初始化 LCD_init  
输入参数: 无  
编写日期: 20080918  
-----*/  
  
void LCD_init(void)  
{  
    res=0;  
    delayms(10);  
    res=1;  
    LCD_write_byte(0x21,0); //1、初始化 Lcd, 功能设定使用扩充指令  
    LCD_write_byte(0xd0,0); //2、设定液晶偏置电压  
    LCD_write_byte(0x20,0); //3、使用基本指令  
    LCD_write_byte(0x0C,0); //4、设定显示模式, 正常显示  
}
```

代码分析：

初始化步骤可以参照芯片手册中的编程示例：

表6 编程示例

步骤	串行总线字节									显示 Y	操作技巧
	D/ \overline{C}	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0		
1	开始										SCE 变低 (LOW)
2	0	0	0	1	0	0	0	0	1		功能设置 PD = 0 和 V = 0 , 选择扩展指令集 (H = 1)
3	0	1	0	0	1	0	0	0	0		设置 V _{OP} ; V _{OP} 设为 a+16 × b [V]
4	0	0	0	1	0	0	0	0	0		功能设置 PD = 0 和 V = 0 , 选择标准指令集 (H = 0)
5	0	0	0	0	0	1	1	0	0		显示控制, 设置标准模式 (D = 1和E = 0)

代码三：

```

/*-----
LCD_write_char    : 显示英文字符
输入参数: c       : 显示的字符;
编写日期         : 2004-8-10
最后修改日期     : 2004-8-10
-----*/

void LCD_write_char_my(unsigned char *font)
{
    unsigned char i;

    for (i=0; i<6; i++)

        LCD_write_byte(font[i], 1);
}

```


代码分析：

我们可以使用字模软件，求出一个 6*8 大小的字符“A”，该组值为：

```
unsigned char A[6] = {0x00, 0x7C, 0x12, 0x11, 0x12, 0x7C};
```

这些很容易理解，还原看一下：

```
000100
001010
010001
010001
011111
010001
010001
000000
```

上面的矩阵，就是一个字母“A”，可见只要把这组值写入即可以显示“A”。

最后，到这里，我们已经学会了 Nokia 5110 液晶屏的原理和基本控制。是不是觉

得很简单。呵呵~~~希望帮到大家，写得不好，请见谅，写的不对，望指正！

小 Jay 累了。