

MSP430 的液晶显示接口设计

南京航空航天大学 魏小龙

MSP430 单片机的很多系列都有液晶驱动的能力，比如：41 系列、42 系列、43 系列、44 系列、31 系列、32 系列、33 系列等都可以直接驱动液晶。数码管显示可能读者非常熟悉，直接使用单片机的引脚或使用功率型接口芯片即可驱动；而液晶显示需要特殊的电压，一般单片机的口线（I/O）不能直接驱动液晶。而上述所列举的 430 系列单片机能直接驱动液晶，本文将以 430F43X、MSP430F44X 为例（具体型号为 MSP430F435）说明在 MSP430 中如何使用液晶显示。

液晶屏使用笔者自己设计（开模定做）的一款，该液晶可涵盖众多的液晶应用，同时该液晶淋漓尽致地体现了 MSP430 的强大液晶显示能力。

一、MSP430 液晶显示原理

MSP430 的液晶显示有静态、2MUX、3MUX、4MUX 四种显示模式，而最常用还是 4MUX 模式。通俗讲，就是有四个公共端（相当于数码管扫描显示的位选端），若干个驱动端的模式。这种模式的最大优点就是能使用最少的引脚提供最多的液晶显示段。图 1 表示了 4MUX 的显示原理。其中（a）说明了一个“8”字的四个公共端、（b）说明了两个驱动端，当分别给公共端与驱动端液晶信号时，就显示对应的数码。

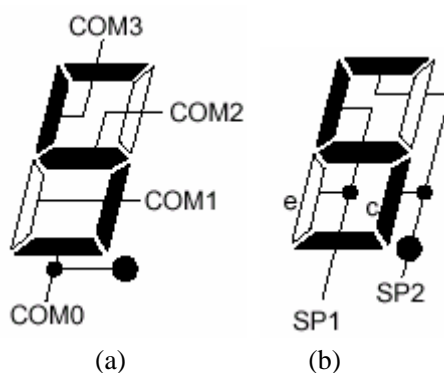


图 1 4MUX 显示模式下的公共端与驱动端

本文将以 4MUX 方式为例讲述液晶应用。在这种方式下，一个“8”字的显示只需要给“COM0、COM1、COM2、COM3、SP1、SP2”对应的信号。而 COM0、COM1、COM2、COM3 公共信号，所有的“8”字都需要与它们相连接。故真正表示一个“8”字的显示的就只与 SP1、SP2 有关系了。所以在 4MUX 方式下，只需要两个驱动端即可表示一个“8”字的显示。

在 MSP430 系列能驱动液晶显示的单片机中，专门开辟了一片存储空间（LCDMEM1~LCDMEM20）存放要显示的信息，被称为液晶显示缓存，简称液晶显存。MSP430F435 一共有 20 字节单元液晶显存，如果使用 4MUX 方式显示，可以显示 160 段液晶笔画。这时，每个显存将对应两个驱动端。图 2 表示了 4MUX 方式下的液晶显存、液晶显示、液晶驱动端之间的对应关系。

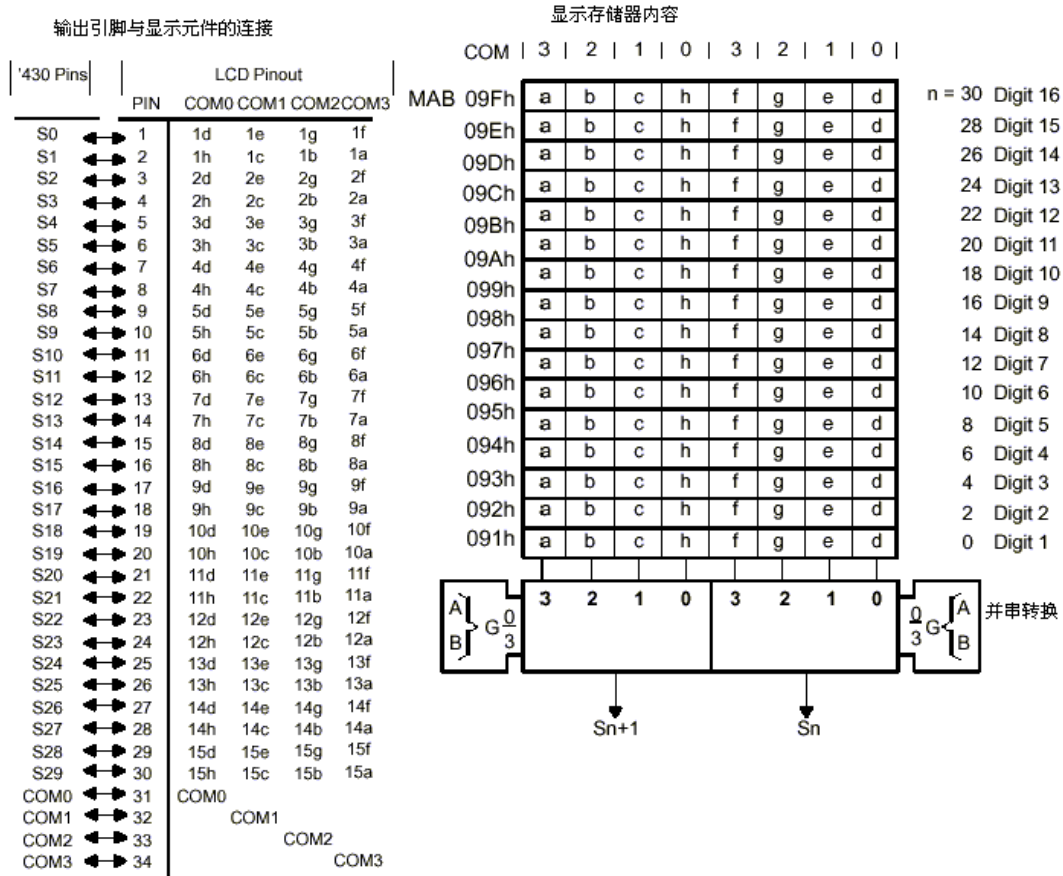
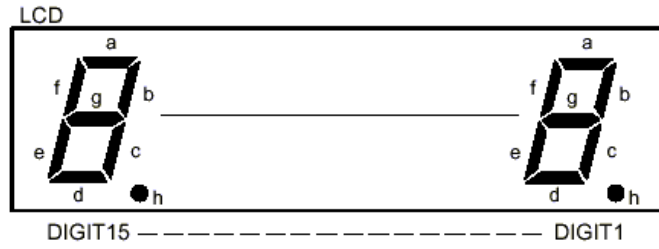


图 2 4MUX 方式下的液晶显示原理

这时要显示“1234567890”则送液晶显存数据为“0x3f, 0x06, 0x5b, 0x4f, 0x66, 0x6d, 0x7d, 0x07, 0x7f, 0x6f,”即可。如果液晶引脚的 S0、S1、S2、S3 分别连接在单片机的 S0、S1、S2、S3，则给 LCDMEM1, LCDMEM2 送数据“0x3f、0x06”液晶上就显示“0、1”字符。

二、液晶简介

此液晶一共 160 段，见图 3。上面为图标与 6 个“米”字，可以显示数字与英文字母，用来表达一些指示性含义的符号。下面一排为 7 个“8”字与符号，显示数字，可以有具体的量纲含义数据。

图 4 为此液晶“8”字、“米”字各笔段的定义。

表 1 为此液晶引脚真值表。

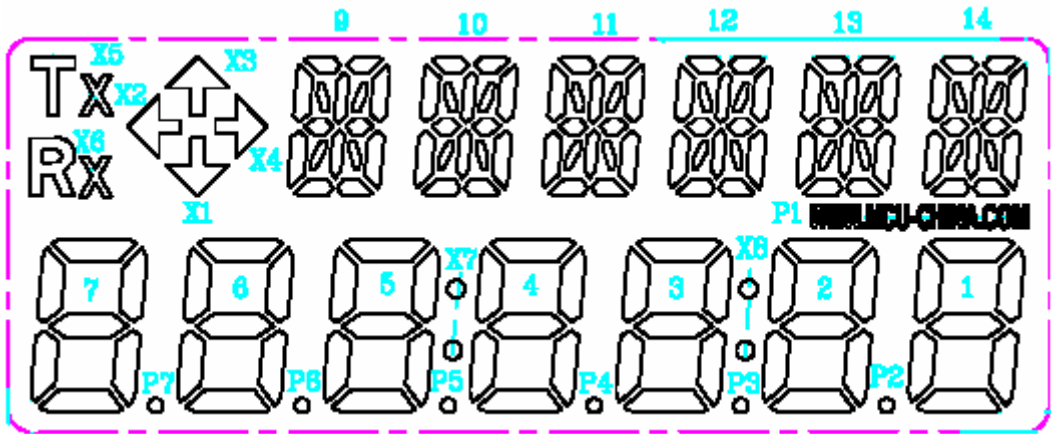


图3 液晶段码布局

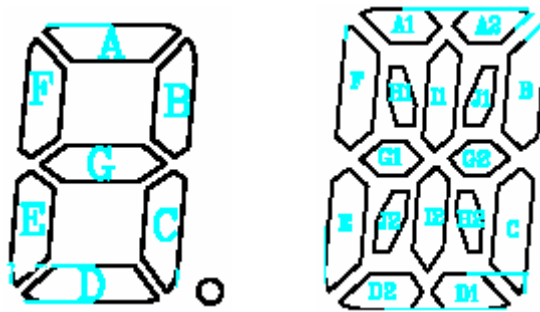


图4 “8”字、“米”字笔段定义

PIN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
COM1	1D	P1	2D	P2	3D	P3	4D	P4	5D	P5	6D	P6	7D	P7	X6
COM2	1C	1E	2C	2E	3C	3E	4C	4E	5C	5E	6C	6E	7C	7E	X7
COM3	1B	1G	2B	2G	3B	3G	4B	4G	5B	5G	6B	6G	7B	7G	X5
COM4	1A	1F	2A	2F	3A	3F	4A	4F	5A	5F	6A	6F	7A	7F	X8
PIN	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
COM1	X2	COM1	/	/	/	9F	9A1	9J1	9A2	10F	10A1	10J1	10A2	11F	11A1
COM2	X3	/	COM2	/	/	9E	9H1	9I1	9B	10E	10H1	10I1	10B	11E	11H1
COM3	X4	/	/	COM3	/	9J2	9G1	9G2	9C	10J2	10G1	10G2	10C	11J2	11G1
COM4	X1	/	/	/	COM4	9D2	9I2	9H2	9D1	10D2	10I2	10H2	10D1	11D2	11I2
PIN	31	32	33	34	35	36	37	38	39	40	41	42	43	44	
COM1	11J1	11A2	12F	12A1	12J1	12A2	13F	13A1	13J1	13A2	14F	14A1	14J1	14A2	
COM2	11I1	11B	12E	12H1	12I1	12B	13E	13H1	13I1	13B	14E	14H1	14I1	14B	
COM3	11G2	11C	12J2	12G1	12G2	12C	13J2	13G1	13G2	13C	14J2	14G1	14G2	14C	
COM4	11H2	11D1	12D2	12I2	12H2	12D1	13D2	13I2	13H2	13D1	14D2	14I2	14H2	14D1	

表1 引脚真值表

三、 硬件连接。

硬件连接最简单，只需要3只电阻、一只晶体。然后将液晶的公共端与单片机的公共端；液晶的各驱动段与单片机对应连接就可以了。具体电路如图5所示（MSP430F435）。


```
unsigned char lcd_seg[]={0x3f,0x06,0x5b,0x4f, 0x66 , 0x6d, 0x7d , 0x07, 0x7f,0x6f};
```

而显示最上面一排符号时，如果要经常使用很多符号也可以使用段码表，如果显示符号不是频繁，也可以不使用段码表，而直接在需要的地方临时翻译段码即可。这时也可以使用上面的方法：比如要在第九个符号的位置（左边第一个“米”字）显示“T”，则修改LCDMEM9=90H、LCDMEM10= 12H ，则显示“T”。其他依此类推。

五、 程序举例

此示例程序的运行，液晶显示将如图 6 所示。

首先要设置液晶显示需要的参数：刷新时钟频率、液晶显示打开、液晶显示模式等。

C 语言语句： LCDCTL = 0XFD;

汇编语句： MOV.B #0FDH, &LCDCTL

然后将各显示数据写入显存即可，这里对数字的写入使用循环语句。

```
#include <msp430x43x.h>
```

```
unsigned char lcd_seg[]={0x3f,0x06,0x5b,0x4f, 0x66,0x6d,0x7d,0x07, 0x7f,0x6f};
```

```
unsigned char lcd_data[ ]={ 0, 8, 1, 2, 3, 9};
```

```
void main(void)
```

```
{
    int I;
    WDTCTL = WDTPW + WDTHOLD;    // 必须的，因为默认为打开
    LCDCTL = 0xfd;
    LCDMEM[7]=0xa;    //显示两个表示时钟的”冒号”
    LCDMEM[8]=0x90;    //以下 8 句显示 TIME
    LCDMEM[9]=0x12;
    LCDMEM[10]=0x80;
    LCDMEM[11]=0x2;
    LCDMEM[12]=0x93;
    LCDMEM[13]=0x72;
    LCDMEM[14]=0x5b;
    LCDMEM[15]=0x94;
    For ( I=0;I<7;I++)    //以下语句显示数组 lcd_data 中的数字
        LCDMEM[I] = lcd_seg[lcd_data[I]];
    for (;;)
    {
        _BIS_SR(SCG1+SCG0+CPUOFF);    // 低功耗
        _NOP();
    }
}
```