

# 晟矽微电 应用笔记

MC51F003A4

LCD 驱动操作

AN22020

V1.1





## 目 录

1	适用范围 .....	1
2	功能简介 .....	1
3	段码 LCD 简介 .....	1
3.1	LCD 结构 .....	1
3.2	LCD 驱动原理 .....	1
4	代码实现 .....	3
5	修订记录 .....	5
6	免责声明 .....	6

Sinomcu.com



## 1 适用范围

本文档适用于使用 MC51F003A4、MC51F8124 驱动 LCD 操作。

## 2 功能简介

本例程提供了一种通过软件 LCD 模块驱动 LCD 屏显示的方法。本例程仅实现 4 位 8 段、1/4duty、1/2bias LCD 驱动，用户可依据具体需求修改完善。

## 3 段码 LCD 简介

### 3.1 LCD 结构

本文 LCD 为 4 位 8 段屏，可以显示 4 位数字，每位包括 ABCDEFGP 8 个笔段。其引脚说明图 1 所示。PIN1-4 为 COM 选择端，选中 COM 端时，PIN5-PIN12 控制 4 个位的其中两个笔段。

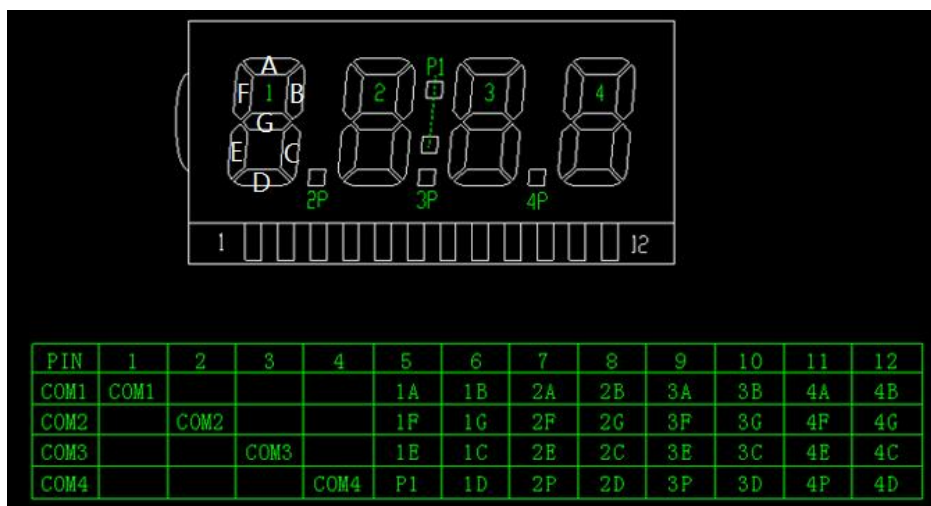


图 1 LCD 引脚说明

### 3.2 LCD 驱动原理

段码 LCD 与 LED 数码管驱动方式有相似之处，每位显示包括 ABCDEFGP 8 个笔段，通过点亮每一个笔段来显示数字。

不同之处：

LED 直接使用直流驱动每段，定时扫描；选中一个 COM 端口时，控制一个位的 8 个笔段。

LCD 需要使用“交流”方式驱动，定时扫描；选中一个 COM 端口时，控制 4 个位的某 2 个笔段。

段码 LCD 驱动帧频率（四个 COM 端扫描一次的时间）：通常在 25Hz~250Hz，一般设置为 60Hz。其 COM 端模拟驱动波形如图 2 所示。

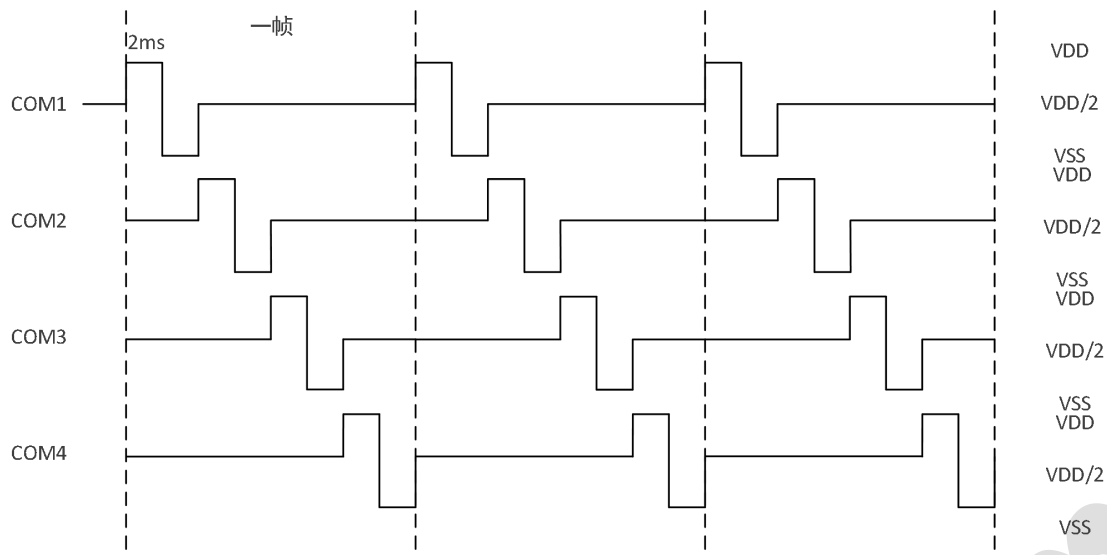


图 2 COM 端模拟驱动波形

- 1) 显示一个笔段: COM、PIN5-12 的压差为 VDD 时, 对应的笔段点亮, 引脚对应关系如图 1 所示。例如: PIN1 为 VDD, PIN5 为 GND 或者 PIN1 为 GND, PIN5 为 VDD 时, 1A 段点亮。
- 2) 显示一位数字: 即控制 A-G 段的亮灭。

显示一位数字, 值为 0-9 时, 各笔段的二进制值如图 3 所示。以 COM 端为低电平时为例, ABCDEFGP 为 1 表示亮, 0 表示灭。

笔段	A	B	C	D	E	F	G	P
数字	0	1	1	1	1	1	1	0
1	0	1	1	1	0	0	0	0
2	1	1	0	1	1	1	0	0
3	1	1	1	1	0	0	1	0
4	0	1	1	0	0	1	1	0
5	1	0	1	1	1	0	1	0
6	1	0	1	1	1	1	1	0
7	1	1	1	0	0	0	0	0
8	1	1	1	1	1	1	1	0
9	1	1	1	1	0	1	1	0
不显示	0	0	0	0	0	0	0	0

图 3 数字 0-9 与 ABCDEFGP 的关系

如图 1 所示, PIN1 引脚与 PIN5-12 引脚控制 4 位数字的 AB 两段, FG、EC、PD 依次由 PIN2、PIN3、PIN4 与 PIN5-12 引脚控制。为编写程序做准备, 将数字 0-9 的 ABCDEFGP 的关系按如下顺序调整。



笔段 数字	P	D	E	C	F	G	A	B
0	0	1	1	1	1	0	1	1
1	0	0	0	1	0	0	0	1
2	0	1	1	0	0	1	1	1
3	0	1	0	1	0	1	1	1
4	0	0	0	1	1	1	0	1
5	0	1	0	1	1	1	1	0
6	0	1	1	1	1	1	1	0
7	0	0	0	1	0	0	1	1
8	0	1	1	1	1	1	1	1
9	0	1	0	1	1	1	1	1
不显示	0	0	0	0	0	0	0	0

图 4 例程中数字 0-9 与 ABCDEFGP 的关系

### 3) 显示 4 位

因 LCD 硬件选择 COM 端时，可同时控制 4 位数字的其中 2 个笔段，为了提高程序效率，同时驱动 4 位显示需要译码。以选择 COM1 时（4 个数字的 A、B 笔段），设 4 位数值分别显示 0 1 2 3，如下图 5 所示。

先将 0 的 AB 段的值赋值给 PIN5 与 PIN6，将 1 的 AB 段的值赋值给 PIN7 与 PIN8，将 2 的 AB 段的值赋值给 PIN9 与 PIN10，将 3 的 AB 段的数值赋值给 PIN11 与 PIN12。则 AB 段译码推算公式为（其他段类似）：

$$\text{show\_num} = \text{num1} \& 0x03 \ll 6 \mid \text{num2} \& 0x03 \ll 4 \mid \text{num3} \& 0x03 \ll 2 \mid \text{num4} \& 0x03$$

注：show\_num:8 个 bit 分别对应 LCD 的 PIN5-12 脚，num1-4: 表示 LCD 屏上显示的 4 位数字。

按LCD显示顺序排列								
笔段 数字	P	D	E	C	F	G	A	B
0	0	1	1	1	1	0	1	1
1	0	0	0	1	0	0	0	1
2	0	1	1	0	0	1	1	1
3	0	1	0	1	0	1	1	1
4	0	0	0	1	1	1	0	1
5	0	1	0	1	1	1	1	0
6	0	1	1	1	1	1	1	0
7	0	0	0	1	0	0	1	1
8	0	1	1	1	1	1	1	1
9	0	1	0	1	1	1	1	1
不显示	0	0	0	0	0	0	0	0

LCD显示的译码推算								
笔段 引脚	PIN5	PIN6	PIN7	PIN8	PIN9	PIN10	PIN11	PIN12
AB	1	1	0	1	1	1	1	1
FG	1	0	0	0	0	1	0	1
EC	1	1	0	1	1	0	0	1
PD	0	1	0	0	0	1	0	1
显示数字	0	1	2	3				

图 5 以 AB 段为例译码示意图

## 4 代码实现

1) 第一步初始化 LCD 模块，使能 LCD 模块并选择合适的分压电阻。



```
LCDCRO = 0x80;           //打开LCD模块显示
LCDSW = 0xcf;             //内部分压300k, 使能COM0-COM1
```

- 2) 第二步模拟 COM 端输出驱动波形, 其波形如图 2 所示。

```
switch(position)
{
    case 0: { COM1_VDD(); P1 = ~COM1_NUM; P0 = ( ~COM1_NUM&0x07)>>1; position = 1;break; } //COM1
    case 1: { COM1_GND(); P1 = COM1_NUM; P0 = ( COM1_NUM&0x07)>>1; position = 2;break; }
    case 2: { COM2_VDD(); P1 = ~COM2_NUM; P0 = ( ~COM2_NUM&0x07)>>1; position = 3;break; } //COM2
    case 3: { COM2_GND(); P1 = COM2_NUM; P0 = ( COM2_NUM&0x07)>>1; position = 4;break; }
    case 4: { COM3_VDD(); P1 = ~COM3_NUM; P0 = ( ~COM3_NUM&0x07)>>1; position = 5;break; } //COM3
    case 5: { COM3_GND(); P1 = COM3_NUM; P0 = ( COM3_NUM&0x07)>>1; position = 6;break; }
    case 6: { COM4_VDD(); P1 = ~COM4_NUM; P0 = ( ~COM4_NUM&0x07)>>1; position = 7;break; } //COM4
    case 7: { COM4_GND(); P1 = COM4_NUM; P0 = ( COM4_NUM&0x07)>>1; position = 0;break; }
    default :break;
}
```

注: MC51F003A4 COM 端口输出  $1/2V_{DD}$  电平时, IO 需设置为保留模式 (00 或者 11 都可以)。

- 3) 第三步计算 LCD 显示的 4 位数字的值, 即根据图 5 所示的规则进行译码, 得到 4 位显示的数字的值。

```
#define COM1_NUM ( (buf[num[0]]&0x03)<<6 | (buf[num[1]]&0x03)<<4 | (buf[num[2]]&0x03)<<2 | buf[num[3]]&0x03 ) //译码
#define COM2_NUM ( (buf[num[0]]&0x0c)<<4 | (buf[num[1]]&0x0c)<<2 | buf[num[2]]&0x0c | (buf[num[3]]&0x0c)>>2 )
#define COM3_NUM ( (buf[num[0]]&0x30)<<2 | buf[num[1]]&0x30 | (buf[num[2]]&0x30)>>2 | (buf[num[3]]&0x30)>>4 )
#define COM4_NUM ( (buf[num[0]]&0xc0 | (buf[num[1]]&0xc0)>>2 | (buf[num[2]]&0xc0)>>4 | (buf[num[3]]&0xc0)>>6 ) | x)
```



## 5 修订记录

版本	修订日期	修订内容
V1.0	2022-10-13	1325, 初版。
V1.1	2023-03-30	1325, 修改笔误。

Sinomcu.com



## 6 免责声明

本资料为晟矽微电子（以下简称“我司”）版权所有。

我司将力求资料内容准确无误，同时保留在不通知用户的情况下，对本资料内容的修改权。  
如您需要获得最新的资料，请及时联系我司。

Sinomcu.com