# LCD1602简述

通信1601

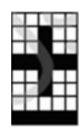
常锡媛

# 主要内容

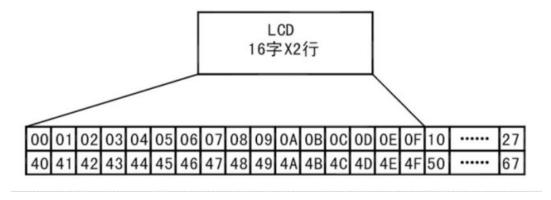
- 1602简介
- · 与STC连接
- 重要指令
- 操作时序
- 代码讲解



- 1、1602液晶是一种专门用来显示数字,字母,符号的点阵型液晶模块(不可显示汉字)
- 2、名字由来:叫1602是因为,该液晶每行可显示16个字符,一共可显示2行
- 3、它是由若干个5\*8或者5\*10的点阵字符位组成,每个点阵字符位都可以用于显示一个字符,每位之间有一个点距的间隔,每行之间也有间隔,这样使得显示时具有一定的字符间距和行间距。
- 4、优点: 抗干扰能力强, 简单编程即可驱动。







1602内部RAM结构图

| 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 0A | 0B | 00 | OD | 0E | 0F |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 4A | 4B | 4C | 4D | 4E | 4F |

每 行 存 储 器 地 址 范 围

# 与STC的连接

| STC学习板J12插座引脚号 | 信号名字 | 1602LCD引脚号 | 信号名字 |
|----------------|------|------------|------|
| 1              | GND  | 1          | VSS  |
| 2              | VCC  | 2          | VCC  |
| 3              | V0   | 3          | V0   |
| 4              | RS   | 4          | RS   |
| 5              | R/W  | 5          | R/W  |
| 6              | Е    | 6          | E    |
| 7              | DB0  | 7          | DB0  |
| 8              | DB1  | 8          | DB1  |
| 9              | DB2  | 9          | DB2  |
| 10             | DB3  | 10         | DB3  |
| 11             | DB4  | 11         | DB4  |
| 12             | DB5  | 12         | DB5  |
| 13             | DB6  | 13         | DB6  |
| 14             | DB7  | 14         | DB7  |
| 15             | PSB  | 15         | LEDA |
| 16             | N.C  | 16         | LEDK |
| 17             | /RST |            |      |
| 18             | VOUT |            |      |
| 19             | Α    |            |      |
| 20             | K    |            |      |







### 1、清屏指令

| 长人叶松 |    | 指令编码 |     |     |     |     |     |     |     |     |  |  |
|------|----|------|-----|-----|-----|-----|-----|-----|-----|-----|--|--|
| 指令功能 | RS | R/W  | DB7 | DB6 | DB5 | DB4 | DB3 | DB2 | DB1 | DB0 |  |  |
| 清屏   | 0  | 0    | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 1   |  |  |

## 功能:

- 1、清除液晶屏显示器,将20H(表示"空白"的ASCII码)写入DDRAM区。
- 2、光标撤回到液晶显示屏的左上方。
- 3、将地址寄存器 (AC) 的值设置为0。



### 2、输入模式设置

| 45 A TH 46 | 指令编码 |     |     |     |     |     |     |     |     |     |  |  |
|------------|------|-----|-----|-----|-----|-----|-----|-----|-----|-----|--|--|
| 指令功能       | RS   | R/W | DB7 | DB6 | DB5 | DB4 | DB3 | DB2 | DB1 | DBO |  |  |
| 进入模式设置     | 0    | 0   | 9   | 9   | 0   | 0   | 0   | 1   | I/D | S   |  |  |

功能: 分配光标移动的方向

1、I=0: 写入新数据后光标左移, I=1: 写入数据后光标右移

2、S=0: 关闭整个移动, S=1: 写入数据后显示屏整体右移一个字符



### 3、显示打开/关闭控制

| 北人大社   | 指令编码 |     |     |     |     |     |     |     |     |     |  |
|--------|------|-----|-----|-----|-----|-----|-----|-----|-----|-----|--|
| 指令功能   | RS   | R/W | DB7 | DB6 | DB5 | DB4 | DB3 | DB2 | DB1 | DBO |  |
| 显示开关控制 | 0    | 0.  | 0   | 0   | 9   | 0   | 1   | D   | С   | В   |  |

功能:设置显示(D)、光标(C)和光标闪烁(B)打开/关闭控制。

1、D=0:显示关闭; D=1: 打开显示

2、C=0: 关闭光标; C=1: 打开光标

3、B=0: 关闭闪烁; B=1: 打开闪烁



#### 4、功能设置

| 七人叶华 |    | 指令编码 |     |     |     |     |     |     |     |     |  |  |
|------|----|------|-----|-----|-----|-----|-----|-----|-----|-----|--|--|
| 指令功能 | RS | R/W  | DB7 | DB6 | DB5 | DB4 | DB3 | DB2 | DB1 | DB9 |  |  |
| 功能设定 | 0  | 9    | 9   | 0   | 1   | DL  | )H  | F   | x   | х   |  |  |

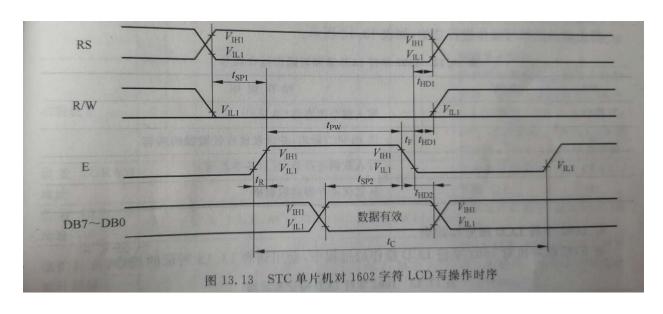
功能: 设置接口数据宽度和显示行的个数

1、DL=1: 8位宽度; DL=0: 4位宽度

2、N=0:显示1行;N=1:显示2行

3、F=0: 5\*8字符字体; F=1: 5\*10字符字体





教材367页图13.13写操作时序

- 1、将R/W信号拉低。同时,给出RS信号,该信号为1或者0,用于区分数据(1)和指令(0)。
- 2、将E信号拉高。当E信号拉高后,STC单片机将写入1602字符LCD的数据放在DB7~DB0数据线上。当数据有效一段时间后,首先将E信号拉低。然后,数据继续维持一段时间 $t_{H2}$ 。这样,数据就写到1602字符LCD中。
- 3、撤出R/W信号。



| p.+ rtr. 40, 86 | 77.0   |     | 单位  | 测量复杂 |    |                 |  |
|-----------------|--------|-----|-----|------|----|-----------------|--|
| 时序参数            | 符号     | 最小值 | 典型值 | 最大值  | 中江 | 测试条件            |  |
| E 信号周期          | to     | 400 | -   | -    | ns |                 |  |
| E 脉冲宽度          | tpw    | 150 | -   | -    | ns | 引脚E             |  |
| E 上升沿/下降沿时间     | tR, tF | -   | -   | 25   | ns |                 |  |
| 地址建立时间          | tsP1   | 30  | -   | -    | ns | PIRME DO DA     |  |
| 地址保持时间          | THD1   | 10  | -   | -    | ns | 引脚 E、RS、R/W     |  |
| 数据建立时间(读操作)     | to     | -   | -   | 100  | ns |                 |  |
| 数据保持时间(读操作)     | tHD2   | 20  | -   | -    | ns | 引脚 DBO~DB7      |  |
| 数据建立时间(写操作)     | tsp2   | 40  | -   | -    | ns | ו פטי שמו לישור |  |
| 数据保持时间(写操作)     | tHD2   | 10  | -   | -    | ns |                 |  |

以上时间的单位都是ns,说明操作是很迅速的



#### 涉及的主要自定义子函数

```
void lcdwait();
void lcdwritecmd(unsigned char cmd);
void lcdwritedata(unsigned char dat);
void lcdinit();
void lcdsetcursor(unsigned char x, unsigned char y);
void lcdshowstr(unsigned char x, unsigned char y, unsigned char x, unsigned char x,
```



#### 写指令函数

```
//写指令到1602函数
void lcdwritecmd(unsigned char cmd)
                                  //等待LCD不忙
 lcdwait();
                                  //空操作延迟
 _nop_();
 _nop_();
 _nop_();
 nop_();
                                  //将LCD1602的RS信号拉低, 写指令
 LCD1602 RS=0;
                                  //将LCD1602的RW信号拉低
 LCD1602 RW=0;
                                  //将指令控制码的cmd放到PO端口
 LCD1602 DB=cmd;
                                  //将LCD1602的使能信号E拉高
 LCD1602 E=1;
 _nop_();
 nop_();
 _nop_();
 nop ();
                                  //将LCD1602的E信号拉低
 LCD1602 E=0;
```



#### 写数据函数

```
// 写数据到1602函数
void lcdwritedata(unsigned char dat)
                                   //等待LCD不忙
 lcdwait();
                                   //空操作延迟
 _nop_();
 _nop_();
 nop_();
 _nop_();
                                   //将LCD1602的RS信号拉高, 写数据
 LCD1602 RS=1;
                                   //将LCD1602的RW信号拉低
 LCD1602 RW=0;
                                   //将数据dat放到PO端口
 LCD1602 DB=dat;
                                   //将LCD1602的E信号拉高
 LCD1602 E=1;
 nop_();
 nop_();
 nop_();
  _nop_();
                                   //将LCD1602的E信号拉低
 LCD1602 E=0;
```



#### 数据存取函数

```
void lcdsetcursor(unsigned char x, unsigned char y) // //设置显示RAM的地址, x和y表示在1602的列和行参数
                                 //声明无符号char型变量address
 unsigned char address;
 if (y==0)
                                 //存续平其地址以0x00开始
  address=0x00+x;
 else
                                 //存储器地址以0x40开始
   address=0x40+x;
                                 //写存储器地址指令
 lcdwritecmd(address|0x80);
void lcdshowstr (unsigned char x, unsigned char y,
                                //在液晶制定的x和v位置,显示字符
            unsigned char *str)
                                 //设置显示RAM的地址
 lcdsetcursor(x, y);
                                 //如果不是字符串的结尾. 则继续
 while((*str)!='\0')
                                 //发写数据指令,在LCD上显示数据
   lcdwritedata(*str);
                                 //指针加1, 指向下一个地址
   str++;
```

# (05) 代码讲解 void lcdwait()

```
LCD1602_DB=0xFF;
_nop_();
_nop_();
_nop_();
_nop_();
LCD1602_RS=0;
LCD1602_RW=1;
LCD1602_E=1;
while(LCD1602_DB & 0x80);
LCD1602_E=0;
```

```
void lcdinit()

{
    lcdwritecmd(0x38);
    lcdwritecmd(0x0c);
    lcdwritecmd(0x06);
    lcdwritecmd(0x01);
}
```

#### 等待函数

```
//等待函数,判断LCD是否忙,是否可读写
//先将PO端口设置为FF
//空操作延时

//将LCD1602的RS信号拉低
//将LCD1602的RW信号拉高
//将LCD1602的E信号拉高
//等待标志位BF为低时
//将LCD1602的E信号拉低

初始化函数

//发指令0x38,2行模式,5*8点阵,8位宽度
//发指令0x06,打开显示,关闭光标
//发指令0x06,文字不移动,地址自动加1
```

//发指令0x01, 清屏