

51单片机轻松入门

—基于STC15W4K系列

(C语言版)

李友全 编著

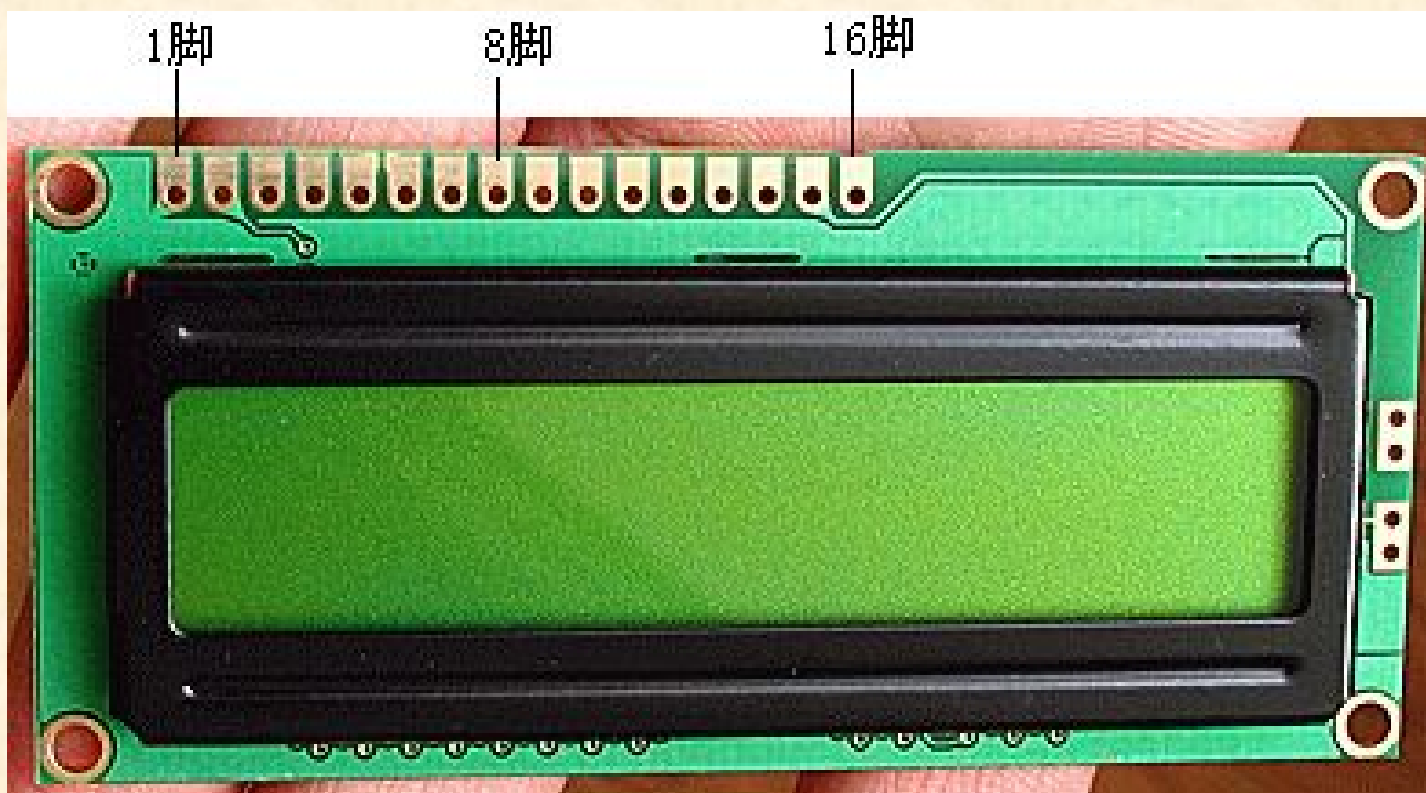
联系方式: <https://shop117387413.taobao.com>

QQ群: 515624099 (验证: STC15单片机)

2016年2月编辑整理 (第13章)

13章 1602液晶

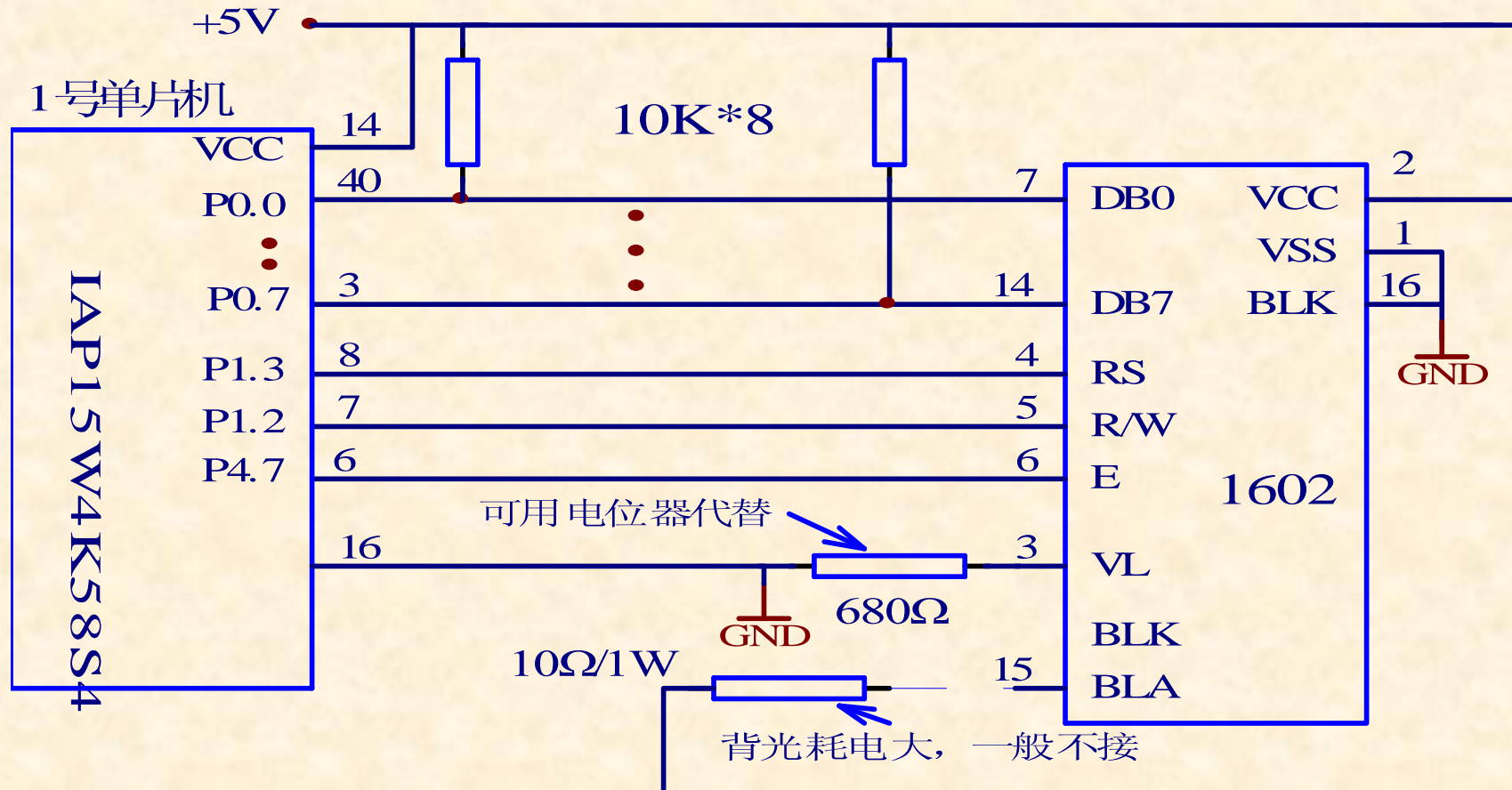
- 1 1602液晶电路图
- 2 LCD1602显示字符串与整数
- 3 1602液晶显示汉字与特殊符号



1 1602液晶电路图

1602液晶可显示2行英文字符，每行最多16个字符，市面上买来的1602液晶一般是可以直接互换使用的，与数码管显示最大的区别在于1602液晶显示字符更多，但亮度比不上数码管显示

1号单片机与D/A输出芯片TLC5615连接电路如图所示。



2 LCD1602显示字符串与整数

例13.1 LCD1602显示字符串与整数，要求1602上电立即在第一行显示字符串“LCD1602-TEST-OK”，在第二行显示一个字符串与一个固定不变的整数“Data = 123456789”。

说明：本例是最简单的字符串与整数显示程序，程序移植时只需要根据实际硬件连接调整LCD1602.H中定义的引脚即可。

```
//***** LCD1602.H *****  
#include "STC15W4K.H"  
#include "intrins.h"          // _nop_()函数需要  
sbit RS = P1^3;              // 根据实际硬件连接修改  
sbit RW = P1^2;              // 根据实际硬件连接修改  
sbit E = P4^7;               // 根据实际硬件连接修改  
#define DPORT P0             // 根据实际硬件连接修改  
void LCD1602_Init();         // 初始化  
void SetCur(unsigned char Para); // 设置光标  
void WriteString(unsigned char xPos, unsigned char yPos, unsigned char *s);  
                                // 写字符串
```

主程序代码如下：

```
#include "LCD1602.H"  
#include "myfun.H"
```

```

void main()
{
    unsigned char xPos, yPos;           // X坐标、Y坐标
    unsigned char *s="Data=";
    unsigned char DispBuf[9];           // 存放4个待发送ASCII码
    unsigned long TestDat=123456789;    // 临时变量
    port_mode();                        // 所有IO口设为准双向弱上拉方式。
    delay100ms();                       // 等待LCD1602上电时内部复位，很重要
    LCD1602_Init();
    SetCur(CurFlash); // 开光标显示、闪烁，NoCur—有显示无光标, NoDisp—无显示,
                        // CurNoFlash—有光标但不闪烁, CurFlash—有光标且闪烁
    xPos=0;                      // xPos表示水平右移字符数(0-15)
    yPos=1;                      // yPos表示垂直下移字符数(0-1)
    WriteString(0, 0, "LCD1602-TEST-OK");
    // X坐标、Y坐标、字符串，屏幕左上角为坐标原点，水平：0-15，垂直：0-1
    Long_Str(TestDat, DispBuf); // 同第4章8节“单片机串口向计算机串口发送2进制...”
    WriteString(xPos, yPos, s); // X坐标、Y坐标、字符串，屏幕左上角为坐标原点
    xPos=5;
    yPos=1;
    WriteString(xPos, yPos, DispBuf);
    while(1);
}

```


实验效果如下图：



3 1602液晶显示汉字与特殊符号

1602液晶显示汉字与特殊符号需要使用自定义字符的方式实现，实际非常简单，我们在前面例子的基础上添加很少的代码就可以实现，下面先看一个最简单的基本实例。

例13.3 1602第1行显示“一二三四五六日℃”。

```
//***** main.C *****  
#include "LCD1602.H"  
#include "myfun.H"  
void main()  
{  
    port_mode();           // 所有IO口设为双向弱上拉方式。  
    delay100ms();          // 等待LCD1602上电时内部复位  
    LCD1602_Init();  
    WriteROM();             // 将自定义字形码写入LCD1602内部存储器  
    WriteChar(0, 0, 0);     // 显示“一”，x坐标、y坐标，ASCII码  
    WriteChar(1, 0, 1);     // 显示“二”  
    WriteChar(2, 0, 2);     // 显示“三”  
    WriteChar(3, 0, 3);     // 显示“四”  
    WriteChar(4, 0, 4);     // 显示“五”  
    WriteChar(5, 0, 5);     // 显示“六”  
    WriteChar(6, 0, 6);     // 显示“日”  
    WriteChar(7, 0, 7);     // 显示“°C”  
    while(1);  
}
```

```

//***** LCD1602.C *****
unsigned char code table[]=
{
    0x00, 0x00, 0x00, 0x00, 0xff, 0x00, 0x00, 0x00,      // 一，显示时的ASCII码 0x00
    0x00, 0x00, 0x00, 0x0e, 0x00, 0xff, 0x00, 0x00,      // 二，显示时的ASCII码 0x01
    0x00, 0x00, 0xff, 0x00, 0x0e, 0x00, 0xff, 0x00,      // 三，显示时的ASCII码 0x02
    0x00, 0x00, 0xff, 0xf5, 0xfb, 0xf1, 0xff, 0x00,      // 四，显示时的ASCII码 0x03
    0x00, 0xfe, 0x08, 0xfe, 0x0a, 0x0a, 0xff, 0x00,      // 五，显示时的ASCII码 0x04
    0x00, 0x04, 0x00, 0xff, 0x00, 0x0a, 0x11, 0x00,      // 六，显示时的ASCII码 0x05
    0x00, 0x1f, 0x11, 0x1f, 0x11, 0x11, 0x1f, 0x00,      // 日，显示时的ASCII码 0x06
    0x18, 0x18, 0x07, 0x08, 0x08, 0x08, 0x07, 0x00,      // °C，显示时的ASCII码 0x07
};

// 将自定义字形码写入1602内部存储器（固定代码，不要改动）
void WriteROM()
{
    unsigned char i;
    LcdWriteCmd(0x40);      // 操作CGRAM的命令码
    for(i=0;i<64;i++)      // 写入数组中数据
    {
        LcdWriteDat(table[i]);
    }
}

```


对于LCD1602.C中字形码数组的获取，可以采用手工方式，也可以采用字模软件实现，由于1602液晶每个字符显示位置是一个 5×8 点阵(5列8行)，如图13-2所示，将点阵的某一行中要显示的点用1表示，不显示的点用0表示，采用一个字节存放一行的方式，字节高3位是无关的，8个字节表示一个字形码，允许最多自定义8个字形码。

×	×	×	4	3	2	1	0												
			1	1	1	1	1	0	0	0	1	1	1	1	1		0x1F		
			1	0	0	0	0	0	0	0	1	0	0	0	1		0x11		
			1	0	0	0	0	0	0	0	1	0	0	0	1		0x11		
			1	1	1	1	1	0	0	0	1	1	1	1	1		0x1F		
			1	0	0	0	0	0	0	0	1	0	0	0	1		0x11		
			1	0	0	0	0	0	0	0	1	0	0	0	1		0x11		
			1	1	1	1	1	0	0	0	1	1	1	1	1		0x1F		
			0	0	0	0	0	0	0	0	0	0	0	0	0		0x00		

当处理字形码较多时，可采用字模软件来操作会更加方便，我们以“字模提取V2.2”为例，进入软件后按图13-3所示步骤操作，虽然我们可以选取宽度5，但等效于8，所示得到的是 8×8 点阵。



我们再点左侧的“模拟动画”菜单，点击“放大格点”选项，一直放大到最大，然后我们就可以在 8×8 的点阵图形中用鼠标点击填充黑点，注意前三列空着不填充，字模软件是把黑色取为1，白色取为0，与手工取模方式正好相同，图形画好后，点一下“参数设置”菜单里的“其他选项”，我们选择“横向取模”，然后选择“取模方式”菜单，点一下“C51 格式”后，在“点阵生成区”自动产生了8个字节的数据，这8个字节的数据就是对应取出来的“模”，对于复杂点的汉字，可以使用紧挨着的2个或4个 5×8 点阵图形进行拼接显示。

例13.4 1602第1行显示“2014年11月30日”，主程序代码如下。

```
#include "LCD1602.H"
#include "myfun.H"
unsigned char str[16];           // 最长字符16字节,
void main()
{
    unsigned char str[]={ '2','0','1','4',0x00,'1','1',0x01,'3','0',0x02};
                        // 显示“2014年11月30日”， 0x00=年， 0x01=月， 0x02=日。
    port_mode();          // 所有IO口设为准双向弱上拉方式。
    delay100ms();         // 等待LCD1602上电时内部复位
    LCD1602_Init();
    WriteROM();           // 将自定义字形码写入LCD1602内部存储器
    WriteStringCN(0,0,str,11);
    while(1);
}
```


实验效果如下图

