

LM9033A

液晶显示模块应用参考

深圳市拓普微科技开发有限公司

版本	描述	日期	编者
0.1	新版本	2005-11-09	淮俊霞
0.2	修改 2.1: 单片机的型号和 V_{CC} 电压值 修改参考程序中的错字	2006-01-18	淮俊霞



目 录

1 简介.....	3
2 应用.....	3
2.1 接口.....	3
2.2 指令表.....	4
2.3 显示内存映射图.....	6
2.4 流程图.....	6
参考程序.....	7

1 简介

本公司产品 LM9033A 为 128×96 全图形点阵的液晶显示模块，能够产生四阶灰度的显示效果，此外还具有负向显示，显示视窗移动、图像移动，软件调节对比度等功能。模块采用 TAB 结构，最大外形尺寸为：63.8×47.4×7.3 (mm)，提供串/并行接口模式，可以满足用户对不同接口的需求。模块的特点有：

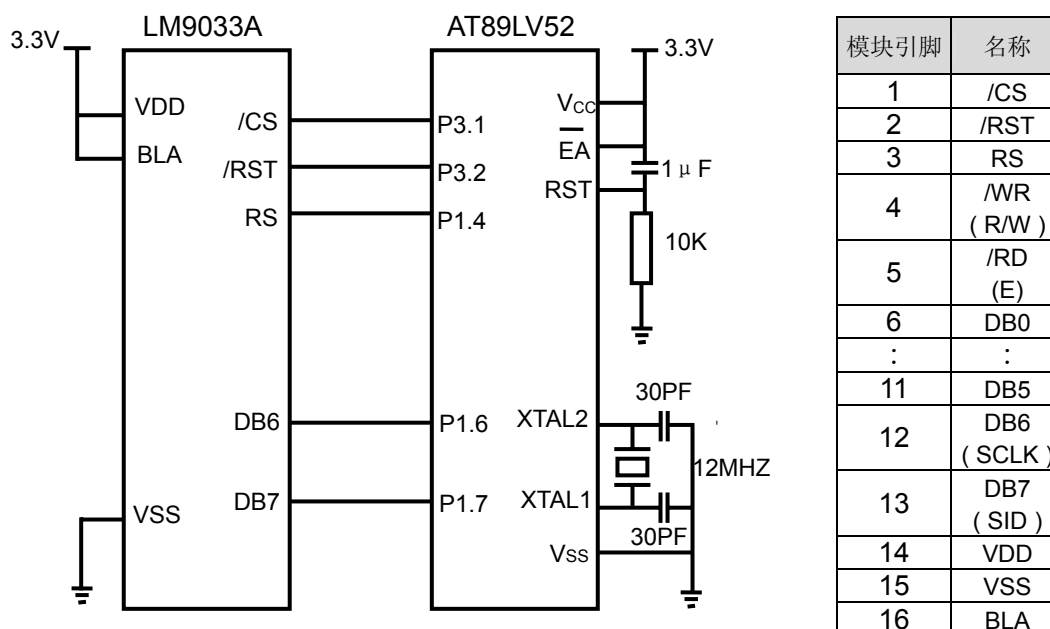
- ✧ 单电源 3.3V 供电，低电压、低功耗
- ✧ 内建升压电路
- ✧ 高对比度，FSTN 型 LCD 屏
- ✧ 白色 LED 背光

2 应用

2.1 接口

短接点设置				功能	描述
JP1	JP2	JP3	JP4		
开路	闭路	闭路	开路	8080 模式	设置为 8080 模式 (8 位并行)
闭路	闭路	开路	开路	6800 模式	设置为 6800 接口模式 (8 位并行)
闭路	开路	开路	闭路	4 线 SPI	设置为 4 线的 SPI (串行) <模块的出厂设置>
开路	开路	闭路	闭路	3 线 SPI	设置为 3 线的 SPI (串行)

模块的背面有 4 个短接点：JP1、JP2、JP3、JP4，用户可根据需要自行设定，见上表。以 4 线的 SPI 应用为例，模块与单片机 AT89LV52 的接口如下图所示。串行接口模式下，显示数据只能写不能读。



2.2 指令表

指令	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
读显示数据	1	1	显示 RAM 数据							
写显示数据	1	0	显示 RAM 数据							
读状态	0	1	BUSY 位 0: 准备好 1: 忙	显示开关 0: 关 1: 开	RES 位 0: 结束 1: 复位	产品 ID MF2 MF1 MF0=[0, 0, 0]			显示尺寸 DS1 DS0=[1, 0]	
页地址	0	0	1	0	1	1	页地址 P3~P0			
列地址 MSB	0	0	0	0	0	1	0	列地址高三位: Y7~Y5		
列地址 LSB	0	0	0	0	0	0	列地址低四位 Y4~Y1			
启动读修改方式	0	0	1	1	1	0	0	0	0	0
结束读修改方式	0	0	1	1	1	0	1	1	1	0
显示开关	0	0	1	0	1	0	1	1	1	显示开关 0: 关 1: 开
显示起始行设置	0	0	0	1	0	0	0	0	×	×
			×	显示起始行: S6~S0						
起始 COM0 寄存器	0	0	0	1	0	0	0	1	×	×
			×	起始 COM0: C6~C0						
部分显示占空比设置	0	0	0	1	0	0	1	0	×	×
			占空比: D7~D0							
N 行反转	0	0	0	1	0	0	1	1	×	×
			×	×	×	N 行反转: N4~N0				
退出 N 行反转	0	0	1	1	1	0	0	1	0	0
负显开关	0	0	1	0	1	0	0	1	1	负向显示 0: 否 1: 是
全显开关	0	0	1	0	1	0	0	1	0	EON 选择 0: 关 1: 开
电源控制	0	0	0	0	1	0	1	升压电路 VC 选择 0 关 1 开	调节电路 VR 选择 0 关 1 开	电压输出 VF 选择 0 关 1 开
选择 DC-DC 升压	0	0	0	1	1	0	0	1	内部电压转换选择 00: 3 倍 01: 4 倍 10: 5 倍 11: 6 倍	
调节电阻设置	0	0	0	0	1	0	0	内部电阻率选择: 1+(Rb/Ra) 000: 2.3 001: 3.0 010: 3.7 011: 4.4 100: 5.1 101: 5.8 110: 6.5 111: 7.2		



接上表

指令	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
设置 EV 寄存器	0	0	1	0	0	0	0	0	0	1
			×	×	参考电压: EV5~EV0					
选择 LCD 偏压	0	0	0	1	0	1	0	偏压选择: 000: 1/5 偏压 001: 1/6 偏压 010: 1/7 偏压 011: 1/8 偏压 100: 1/9 偏压 101: 1/10 偏压 110: 1/11 偏压 111: 1/12 偏压		
SHL 选择	0	0	1	1	0	0	SHL 方向 0: 正常 1: 反向	×	×	×
ADC 选择	0	0	1	0	1	0	0	0	0	ADC 方向 0: 正常 1: 反向
开振荡器	0	0	1	0	1	0	1	0	1	1
设置省电模式	0	0	1	0	1	0	1	0	0	电源模式 0: 正常 1: 休眠
退出省电模式	0	0	1	1	1	0	0	0	0	1
软件复位	0	0	1	1	1	0	0	0	1	0
设置数据方向和显示数据长度	×	×	1	1	1	0	1	0	0	0
			数据的字节数 (仅用于 3 线的 SPI): DB7~DB0							
设置 FRC 和 PWM 模式	0	0	1	0	0	1	0	FRC 选择 0: 4FRC 1: 3FRC	PWM 选择 00: 9PWM 01: 9PWM 10: 12PWM 11: 15PWM	
设置白模式 1 st /2 nd 帧脉冲宽度	0	0	1	0	0	0	1	0	0	0
			WB3~WB0				WA3~WA0			
设置白模式 3 rd /4 th 帧脉冲宽度	0	0	1	0	0	0	1	0	0	1
			WD3~WD0				WC3~WC0			
设置淡灰模式 1 st /2 nd 帧脉冲宽度	0	0	1	0	0	0	1	0	1	0
			LB3~LB0				LA3~LA0			
设置淡灰模式 3 rd /4 th 帧脉冲宽度	0	0	1	0	0	0	1	0	1	1
			LD3~LD0				LC3~LC0			
设置深灰模式 1 st /2 nd 帧脉冲宽度	0	0	1	0	0	0	1	1	0	0
			DB3~DB0				DA3~DA0			
设置深灰模式 3 rd /4 th 帧脉冲宽度	0	0	1	0	0	0	1	1	0	1
			DD3~DD0				DC3~DC0			
设置黑模式 1 st /2 nd 帧脉冲宽度	0	0	1	0	0	0	1	1	1	0
			BB3~BB0				BA3~BA0			
设置黑模式 3 rd /4 th 帧脉冲宽度	0	0	1	0	0	0	1	1	1	1
			BD3~BD0				BC3~BC0			

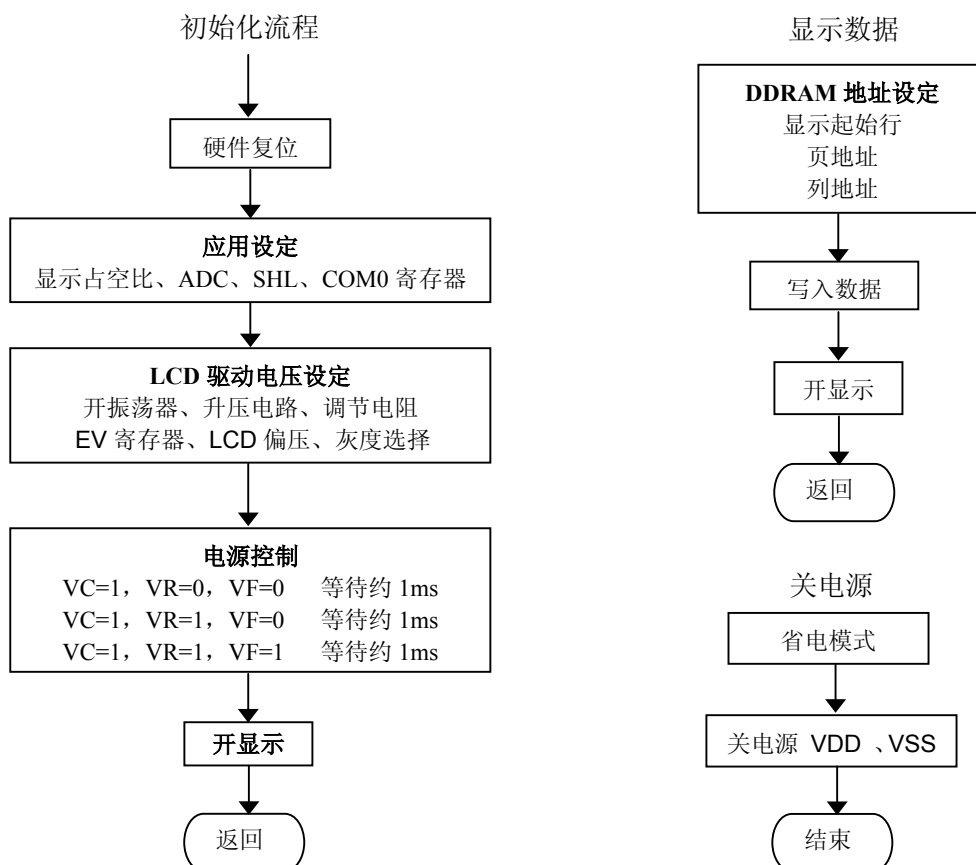
注: 详细的说明请参见 LM9033A 的技术手册或者三星 S6B0741 控制器手册

2.3 显示内存映射图

页地址	数据	LCD 模块顶视图														
0	D0 ⋮ D7	<div style="text-align: center;"> <p>128×96 像素</p> <p>四级灰度</p> <p>数据位值 (2n, 2n+1)</p> </div>														
1	D0 ⋮ D7															
⋮	⋮															
⋮	⋮															
10	D0 ⋮ D7															
11	D0 ⋮ D7															
列地址	Y7 ⋮ Y1	00H	01H	02H	...				7DH	7EH	7FH					
	Y0	0	1	0	1	0	1	...				0	1	0	1	0
内部显示 RAM 地址																

注：在进行读写操作时，列地址指针具有自加 1 的功能，但用户只能定义列地址指针的高 7 位 Y7~Y1；页地址指针不具有自动加 1 的功能，需要用户设定页地址指针 P3~P0。

2.4 流程图



参考程序

```

//-----
//本例程的演示结果为：“TOPWAY LM9033A”
//          + 四级灰度条
//-----
#include <reg52.h>
sbit _CS=P3^1;          //片选信号
sbit _RST=P3^2;        //复位信号
sbit RS=P1^4;          //寄存器选择
sbit SCLK=P1^6;        //串行通信数据线的时钟信号
sbit SID=P1^7;         //串行通信数据线

unsigned char code bmp[]=
{ 0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
  0x18,0x08,0x08,0xF8,0x08,0x08,0x18,0x00,
  0xE0,0x10,0x08,0x08,0x08,0x10,0xE0,0x00,
  0x08,0xF8,0x08,0x08,0x08,0x08,0xF0,0x00,
  0xF8,0x08,0x00,0xF8,0x00,0x08,0xF8,0x00,
  0x00,0x00,0xC0,0x38,0xE0,0x00,0x00,0x00,
  0x08,0x38,0xC8,0x00,0xC8,0x38,0x08,0x00,
  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
  0x08,0xF8,0x08,0x00,0x00,0x00,0x00,0x00,
  0x08,0xF8,0xF8,0x00,0xF8,0xF8,0x08,0x00,
  0x00,0xE0,0x10,0x08,0x08,0x10,0xE0,0x00,
  0x00,0xE0,0x10,0x08,0x08,0x10,0xE0,0x00,
  0x00,0x30,0x08,0x88,0x88,0x48,0x30,0x00,
  0x00,0x30,0x08,0x88,0x88,0x48,0x30,0x00,
  0x00,0x00,0xC0,0x38,0xE0,0x00,0x00,0x00,
  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
  0x00,0x00,0x20,0x3F,0x20,0x00,0x00,0x00,
  0x0F,0x10,0x20,0x20,0x20,0x10,0x0F,0x00,
  0x20,0x3F,0x21,0x01,0x01,0x01,0x00,0x00,
  0x03,0x3C,0x07,0x00,0x07,0x3C,0x03,0x00,
  0x20,0x3C,0x23,0x02,0x02,0x27,0x38,0x20,
  0x00,0x00,0x20,0x3F,0x20,0x00,0x00,0x00,
  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
  0x20,0x3F,0x20,0x20,0x20,0x20,0x30,0x00,
  0x20,0x3F,0x00,0x3F,0x00,0x3F,0x20,0x00,
  0x00,0x00,0x31,0x22,0x22,0x11,0x0F,0x00,
  0x00,0x0F,0x10,0x20,0x20,0x10,0x0F,0x00,
  0x00,0x18,0x20,0x20,0x20,0x11,0x0E,0x00,
  0x00,0x18,0x20,0x20,0x20,0x11,0x0E,0x00,
  0x20,0x3C,0x23,0x02,0x02,0x27,0x38,0x20,
  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00};
//-----
//延迟子程序
//-----
void Delay(unsigned int t)
{
    unsigned int i;
    for(i=0;i<t;i++);
}
//-----
//串行口送一个字节的的数据
//-----
void SendByte(unsigned char Dbyte)
{
    unsigned char i;
    for(i=0;i<8;i++)
    {
        SCLK=0;
        Dbyte=Dbyte<<1; //左移一位
        SID=CX;          //移出的位送 SID
        SCLK=1;
    }
}
//-----
//写指令
//-----
void CmdWrite(unsigned char cmdcode)
{
    Delay(0);
    _CS=0;
    RS=0;
    SendByte(cmdcode);
    RS=1;
    _CS=1;
}

//-----
//数据写操作
//-----
void DataWrite(unsigned char wrdata)
{
    Delay(0);
    _CS=0;
    RS=1;
    SendByte(wrdata);
    RS=0;
    _CS=1;
}

//-----
//整屏幕写入数据
//-----
void FullScreenFill(unsigned char fill_data)
{
    unsigned char i,j;
    CmdWrite(0x00); //列地址低 4 位为 0
    CmdWrite(0x10); //列地址高 3 位为 0
    for(i=0;i<128;i++)
    {
        CmdWrite(0xb0+i); //设置页地址
        for(j=0;j<128;j++)
        {
            DataWrite(fill_data);
            DataWrite(fill_data);
        }
    }
}

//-----
//显示图片
//-----
void ShowBmp()
{
    unsigned char i,j;
    CmdWrite(0x00); //列地址低 4 位为 0
    CmdWrite(0x10); //列地址高 3 位为 0
    for(j=0;j<2;j++)
    {
        CmdWrite(0xb2+j); //设置页地址
        for(i=0;i<128;i++)
        {
            DataWrite(bmp[i+128*j]);
            DataWrite(bmp[i+128*j]);
        }
    }
}

//-----
//四级灰度条
//-----
void GrayScale()
{
    unsigned char i,j,k,l;
    unsigned char mm=0xff;
    unsigned char nn=0xff;
    CmdWrite(0x00); //列地址低 4 位为 0
    CmdWrite(0x10); //列地址高 3 位为 0
    for(k=0;k<6;k++)
    {
        CmdWrite(0xb6+k); //设置页地址
        for(j=0;j<2;j++)
        {
            mm=~mm;
            for(i=0;i<2;i++)
            {
                nn=~nn;
                for(l=0;l<32;l++)
                {
                    DataWrite(mm);
                    DataWrite(nn);
                }
            }
        }
    }
}

```



```
//-----  
//模块复位与初始化  
//-----  
void LCD_Initial()  
{  
    _RST=0;           //复位开始  
    Delay(100);  
    _RST=1;  
    Delay(100);      //复位结束  
  
    CmdWrite(0x48);CmdWrite(0x60);  
    CmdWrite(0xa1);  
    CmdWrite(0xc8);  
    CmdWrite(0x44);CmdWrite(0x10);  
    CmdWrite(0x40);CmdWrite(0x00);  
    CmdWrite(0xab);  
    CmdWrite(0x66);  
    CmdWrite(0x27);  
    CmdWrite(0x81);CmdWrite(0x20);  
    CmdWrite(0x54);  
    CmdWrite(0x94);  
    CmdWrite(0x88);CmdWrite(0x00);  
    CmdWrite(0x89);CmdWrite(0x00);  
    CmdWrite(0x8a);CmdWrite(0x09);  
  
    CmdWrite(0x8b);CmdWrite(0x00);  
    CmdWrite(0x8c);CmdWrite(0x99);  
    CmdWrite(0x8d);CmdWrite(0x00);  
    CmdWrite(0x8e);CmdWrite(0x99);  
    CmdWrite(0x8f);CmdWrite(0x09);  
    CmdWrite(0x2c);Delay(100);  
    CmdWrite(0x2e);Delay(100);  
    CmdWrite(0x2f);Delay(100);  
    CmdWrite(0xaf);  
}  
//-----  
//主程序  
//-----  
void main()  
{  
    RS=1;  
    _CS=1;  
    LCD_Initial();  
    FullScreenFill(0x00); //清屏  
    ShowBmp();  
    GrayScale();  
    Delay(60000);  
    CmdWrite(0xa9);  
}  
//end of program
```