JH-A56256L　VRAM 液晶显示控制板技术手册

JH-A56256L 可控制 5.6 寸/7 寸模拟彩色 TFT 模拟液晶显示屏。工作电压为 5V，电流为 80mA(仅就本控制板而言，不包含屏和背光的电流).本控制板采用 I/O 连接方式，可显示 256 色；为提高读写速度、简化程序，显示屏中每个点影射显示缓存中的一个字节，显示屏中的行列号与缓存器的行列号一一对应,用户只需把数据连续写进 SRAM 中即可,对用户来讲,直接输入的是 LCD 上的点的坐标.详细的程序编写请参考 DEMO 程序。由于显示缓存采用 SRAM，显示容量大，对于 320×234 点阵彩色 LCD，可储存 1 页显示内容。如果用户的数据比较大，我们可以提供原理图和 PCB 图，让用户自己根据自己的结构去布局。我们只提供编好程序的 CPLD 即相关的技术支持服务。适配 CPU：51,96,X86,8088,Z80,DSP,ARM.　本控制板的响应速度很快。能达到 50ns 的读写时间.本控制板的 7 寸和 5 寸是一样的。

一、接口定义：

1、CPU 侧接口（双排压线）

引脚	符号	功能	备注
1	NC		
2	GND	液晶屏逻辑地	0V
3	VCC	液晶屏逻辑电源	5V
4	RD	读操作信号，低电平有效	IN
5	WR	写操作信号，低电平有效	IN
6	CS	片选信号，低电平对屏操作有效	IN
7	RS	1---对数据通道操作，0---写命令寄存器	IN
8	DATA0	数据总线	INOUT
9	DATA1	数据总线	INOUT
10	DATA2	数据总线	INOUT
11	DATA3	数据总线	INOUT
12	DATA4	数据总线	INOUT
13	DATA5	数据总线	INOUT
14	DATA6	数据总线	INOUT
15	DATA7	数据总线	INOUT
16	NC	保留	

本控制板有如下性能：

1、 本控制板可工作于 8 位数据总线模式

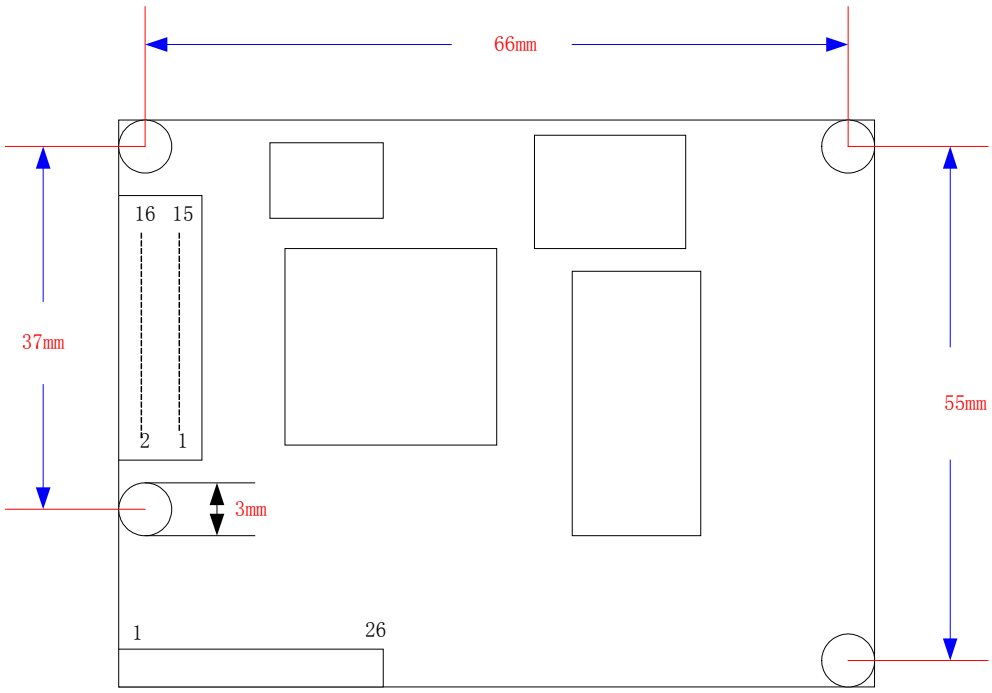
2、 用户的 CPU 不需要计算行列与 SRAM 地址的对应关系，直接输入坐标地址即可。

3、 本控制板的读写周期可做到 50ns,无需任何等待。请详细参考时序图

4、 连续读或写入时，地址自动加 1，遇到行末自动跳到下一行。

5、 H 型控制板支持块填充和块移动的命令，适合做动画等.如果用户自己的 CPU 是速度快的 ARM/DSP 等，建议不必选用这个型号。如果用户的 CPU 是 51 等指令周期比较长的 CPU，建议选用这种型号以实现比较好的显示效果。

2　TTL 接口控制板尺寸



16 pin 双排座的 PIN 间距为 2.54mm

本板适合于下面接口顺序的 5.6 或 7 寸的模拟屏。其他的不适合。

Pin no	Symbol	I/O	Description
1	GND	-	Ground for logic circuit
2	VCC	I	Supply voltage of logic control circuit for scan driver
3	VGL	I	Negative power for scan driver
4	VGH	I	Positive power for scan driver
5	STVR	I/O	Vertical start pulse
6	STVL	I/O	Vertical start pulse
7	CKV	I	Shift clock input for scan driver
8	U/D	I	Up/Down scan control input
9	OEV	I	Output enable input for scan driver
10	VCOM	I	Common electrode driving signal

11	VCOM	I	Common electrode driving signal
12	L/R	I	Left/Right scan control input
13	Q1H	I	Analog signal rotate input
14	OEH	I	Output enable input for data driver
15	STHL	I/O	Start pulse for horizontal scan line
16	STHR	I/O	Start pulse for horizontal scan line
17	CPH3	I	Sampling and shifting clock pulse for data driver
18	CPH2	I	Sampling and shifting clock pulse for data driver
19	CPH1	I	Sampling and shifting clock pulse for data driver
20	VCC	I	Supply voltage of logic control circuit for data driver
21	GND	-	Ground for logic circuit
22	VR	I	Alternated video signal input(Red)
23	VG	I	Alternated video signal input(Green)
24	VB	I	Alternated video signal input(Blue)
25	AVDD	I	Supply voltage for analog circuit
26	AVSS	-	Ground for analog circuit

3 CS 与 RS,WR,RD,DATA 组合功能如下：

CS	RS	DATA[7:0]	WR	RD	功能
0	0	0x00	0	1	保留
0	1	00xXX	0	1	保留
0	0	0x01	0	1	地址指向列地址低 8 位寄存器
0	1	00xXX	0	1	写入列地址的低 8 位
0	0	0x02	0	1	地址指向列地址的高位寄存器

0	1	X	0	1	写入列地址的高位
0	0	0x03	0	1	地址指向行地址寄存器
0	1	X	0	1	写入行地址的
0	0	0x07	0	1	地址指向数据通道
0	1	X	0	1	向控制板写入数据
0	1	X	1	0	从控制板读出显示数据
1		×	×	×	不选通

说明：在读写显示数据时,要保证指令寄存器的值设为 7,选择数据通道，不能被修改为其他值，否则，会读写不正常。因为本方案针对的模拟屏的行数是 234 行，小于 256，所以，行地址只需要一个字节。

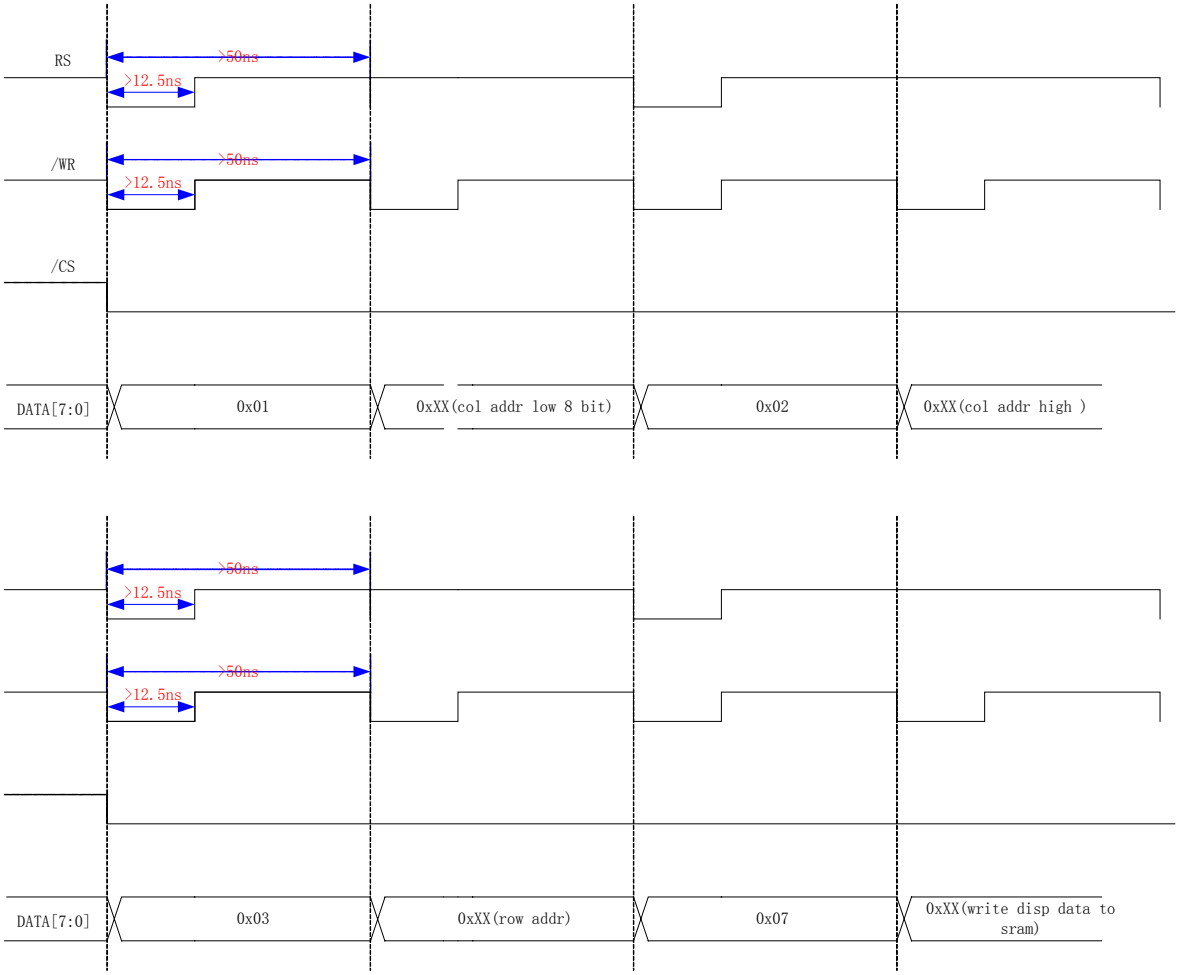
256 色的设置方法:

	D7	D6	D5	D4	D3	D2	D1	D0
256 色	R2	R1	R0	G2	G1	G0	B1	B0

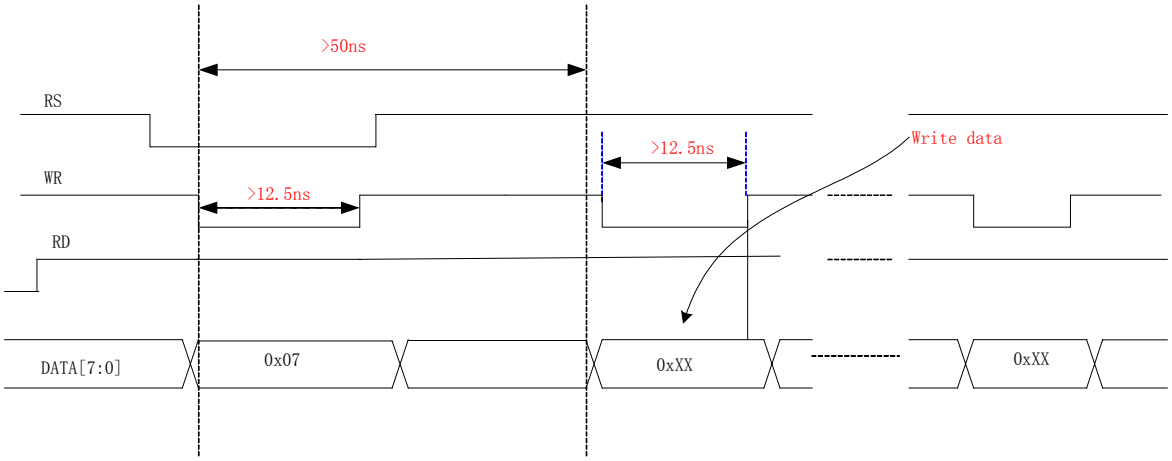
256 色

	颜色灰度	R2、R1、R0	G2、G1、G0	B1、B0
基本颜色	最黑	000	000	00
	亮蓝	000	000	11
	亮绿	000	111	00
	亮青	000	111	11
	亮红	111	000	00
	亮紫	111	000	11
	亮黄	111	111	00
	亮白	111	111	11
蓝色灰度	最黑	000	000	00
	较暗	000	000	01
	较亮	000	000	10
	最亮	000	000	11
绿色灰度	最黑	000	000	00
	较暗	000	001	00
	…	…	…	…
	较亮	000	110	00
	最亮	000	111	00
红色灰度	最黑	000	000	00
	较暗	001	000	00
	…	…	…	…
	较亮	110	000	00
	最亮	111	000	00

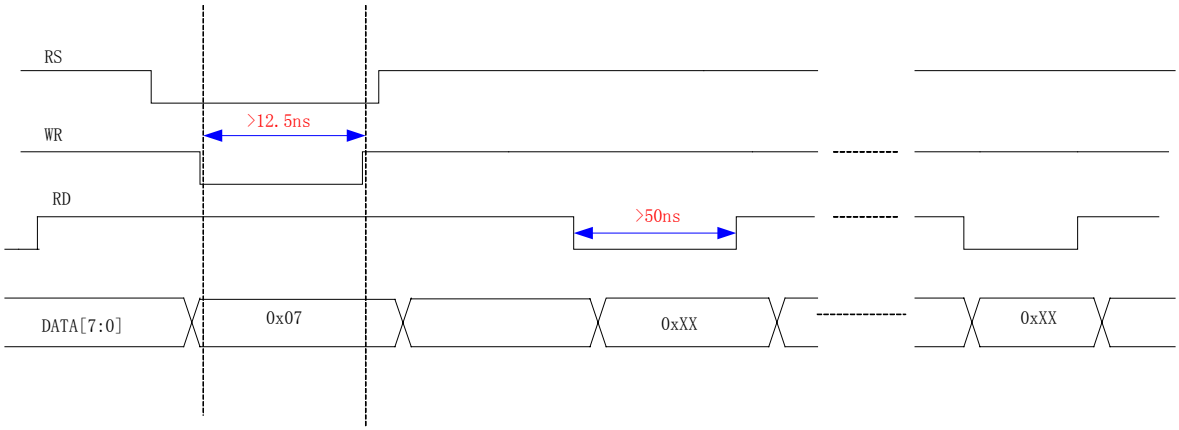
1 写列地址位的时序



5 写显示数据进显存中的时序

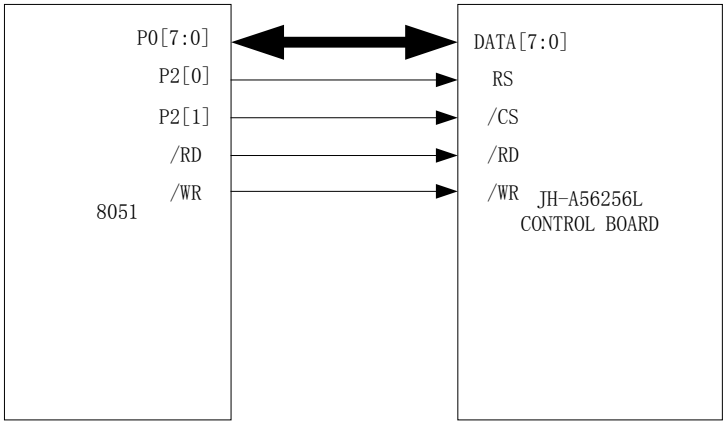


6 从显存中读出显示数据的时序



注意：读操作时，低电平要保证有 50ns 以上的空间，CPU 才能在 RD 上升沿时得到正确的数据，否则，CPU 在 RD 信号的上升沿采集到的数据是不正确的。

四、简单的单片机接口应用框图



8051 控制方式

16×16 点阵汉字字符输入程序(51)：

汉字左上角为汉字在显示屏中坐标，

；汉字字模有两种方式，如下表，以下演示程序是以第二种字模编写的。

左半字模	右半字模
第 1 字节	第 2 字节
第 3 字节	第 4 字节
…	…
第 31 字节	第 32 字节

左半字模	右半字模
第 1 字节	第 17 字节
第 2 字节	第 18 字节

…	…
第 16 字节	第 32 字节

根据客户的取字模式来把字符中的点阵信息显示出来。显示的时候，要用前景色去修改点阵数据中为 1 的 BIT 对应的点的显示数据。用背景色去修改点阵数据中为 0 的 BIT 的数据。
假如某个字符的第 1 个字节的点阵数据如下，而我们想把这个字符显示在 33 行，100 列上（33，100）。我们把前景色设为白色（0xffh），背景色设为兰色（0x03），我们用 FontData 来保存字模的点阵数据.

0	0	1	0	1	1	0	0

当用户设置好工作模式寄存器 1，前景色寄存器，被景色寄存器以及该字符显示位置的起始点后，就可以把这个点阵数据送给控制板了。

本手册上的 51 单片机的连接方式,命令地址为
FontColo 保存前景色的寄存器
BackColo 保存背景色的寄存器
WCMD 0xfc00 命令通道地址
WDAT 0xfd00 数据通道地址

```
  WCMD=0x01; //地址指针指向列地址低位寄存器
  WDAT=0x00; //送列地址的低 8 位的地址，如果从 0 行开始写，则这个值送 0

  WCMD=0x02; //送列的高 2 位地址的前缀命令
  WDAT=0x00; //送列的高 2 位地址,

  WCMD=0x03; //送行的地址的前缀命令  相当于 mov dptr,#CMDAddr,mov a,#00h,movx
             //@dptr,a,这三条汇编指令
  WDAT=0x00; //送行的地址 如果是从 0 列开始写，

  WCMD=0x07; //送完行列的初始地址后，要把指针指向数据通道。

 WDAT=0x03; //往控制板上写入一个点的显示数据---蓝色
```

注意： 行方向可以在送完初始地址后，可以连续地写入数据。而如果想画一条竖线，必须对每个点的地址都要设定。

```
//////////////////////////////////////////////////////////////
//画点的函数----坐标和颜色
void DrawDot(unsigned int y,unsigned int x,unsigned char color)
{
   DrawInitail(y,x);
   WDAT=color;
}
//////////////////////////////////////////////////////////////
/画线的函数  入口参数---线的起点和终点----
//  DrawLine(300,0,300,233,0xe0)
```

```
void DrawLine(unsigned int ysta,unsigned int xsta,unsigned int yend,
          unsigned int xend,unsigned char color)
{  unsigned int  DeltaX;
   unsigned int  DeltaY;
   unsigned int  X;  //列号，列间距
   unsigned int  Y; //两行之间的间距
   unsigned int  i;
       DeltaX=xend-xsta; //列间距
       DeltaY=yend-ysta;//行间距

   if(DeltaX==0) //这是纵向画的线,在同一列上画线
    { //纵向画线，每次都需要送一次点的坐标

       X=xsta;
        for (i=0;i<DeltaY;i++)
       {
          Y=ysta+i;//要实现画纵向的点,每次必须跨一行
          DrawDot(Y,X,color);

       }
    }
   else if(DeltaY==0) //这是须横向画的线,行间距为 0，在同一行上画线
    {
     Y=ysta;
     X=xsta;
     DrawInitail(Y,X);
     for (i=0;i<DeltaX;i++)//在列方向上跨步前进
       {
         WDAT=color;

       }

    }
  // else  //这是斜线


}
//////////////////////////////////////////////////////////////////////
//显示一个 16X16 点阵汉字的程序，有 32 个字节，列方向两个字节，行方向有 16 个字节
    j=0;
  while(j<16)
  {
     DrawInitail(y+j,x);//把字符将要显示的左上角坐标设置好
    FontDat=FontArr[CntFontData];//得到 8 个点的显示信息
```

```
        //字符被放在一个数组里面
    for(i=0;i<8;i++)  //处理字模的左半部分
      {
       c=(bit)(FontDat&0x80);
       FontDat=FontDat<<1;
       if (c==0)
        WDAT=BackColor;
       else
        WDAT=FontColor;
      }
    CntFontData=CntFontData+1;
       FontDat=FontArr[CntFontData];
        for(i=0;i<8;i++)  //处理字模的右半部分
      {
       c=(bit)(FontDat&0x80);
       FontDat=FontDat<<1;
       if (c==0)
        WDAT=BackColor;
       else
        WDAT=FontColor;
      }
       CntFontData=CntFontData+1;
       j++; //j 是行的计数器
    }
```

本控制板提供有可提供参考源程序。如果你有什么意见和建议，请email给我们，我们将尽量满足您的要求。联系：lijie789@sohu.com 电话：0755-81953366 李工