

WLA70 液晶显示控制板技术手册

世龙显示控制产品核心电路采用 ALTERA 公司的大规模可编程集成电路 (CPLD) EPM240 编程实现, 性能稳定可靠。WLA70 可控制彩色 TFT 液晶显示屏, 采用总线连接方式, 占用 CPU 外部 4 字节存储空间, 可显示 256 种颜色; 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 XY 坐标号与缓存器的行列号一一对应, 因此, 只需输入 XY 坐标, 便可直接读写相应点数据, 不用计算点在显示缓存中的地址, 写入数据后 X 坐标自动加 1, 写满一行后自动换行。由于显示缓存采用高速 SRAM, 显示容量大, 对于 480X234 点阵彩色 LCD, 可储存 4 页显示内容。对当前页或任一页读写不影响当前页的显示, 无雪花现象, 页之间的内容可方便地实现交替显示。

适配 CPU: 51, 96, X86, 8088, Z80, DSP

一、控制板 CPU 端接口定义

控制板 CPU 侧接口

表 1 双排压线

引脚	符号	功能	备注
1	VCC	液晶屏逻辑电源	3.3V
2	VCC	液晶屏逻辑电源	3.3V
3	NC		
4	DATA0	数据总线	
5	DATA1	数据总线	
6	DATA2	数据总线	
7	DATA3	数据总线	
8	DATA4	数据总线	
9	DATA5	数据总线	
10	DATA6	数据总线	
11	DATA7	数据总线	
12	/CS	片选信号, 低电平对屏操作有效	
13	/WR	写操作信号, 低电平有效。	
14	/RD	读操作信号, 低电平有效。	
15	NC		
16	A0	寄存器地址	
17	A1	寄存器地址	
18	NC		
19	GND	电源地	

20	GND	电源地	
----	-----	-----	--

二、控制板操作指令系统

WLA70 控制板中有 4 个寄存器，它们是显示屏 Y 坐标、X 坐标、数据、状态控制寄存器，通过对相关寄存器操作，可实现**单点写屏**。

所有寄存器只能写，不能读。

表 5

CS 与 A0、A1 组合功能如

CS	A1A0	WR	RD	功能
0	00	0	1	写列低 8 位地址
0	01	0	1	写行低 8 位地址
0	10	0	1	写显示数据(颜色值)
0	11	0	1	写状态字
0	×××	1	0	读显示数据(颜色值)
1	×××	×	×	不选通

A)、状态字寄存器 PSW：地址 AA1A0=11

D7	D6	D5	D4	D3	D2	D1	D0
Pd1	Pd0	Prw1	Prw0	Yinc	Xinc	×	X8

Pd1, Pd0 为显示页号；

Prw1, Prw0 为读写页号, 对当前显示页或其他页读写，不影响当前页的显示，无雪花现象。

Yinc 为 Y 坐标自动加 1 控制位，=1 时允许；=0 禁止。

Xinc 为列号（X 坐标）地址自动增 1、自动换行控制位：

=1 时允许，=0 时禁止；

写入数据后，X 坐标自动加 1，写满一行时自动换行，即 X=0，Y+1。

X8 为 X 坐标高字节位。

×：待开发位，建议设置为 0。

B)、数据寄存器：地址 A1A0=10，其颜色设置为

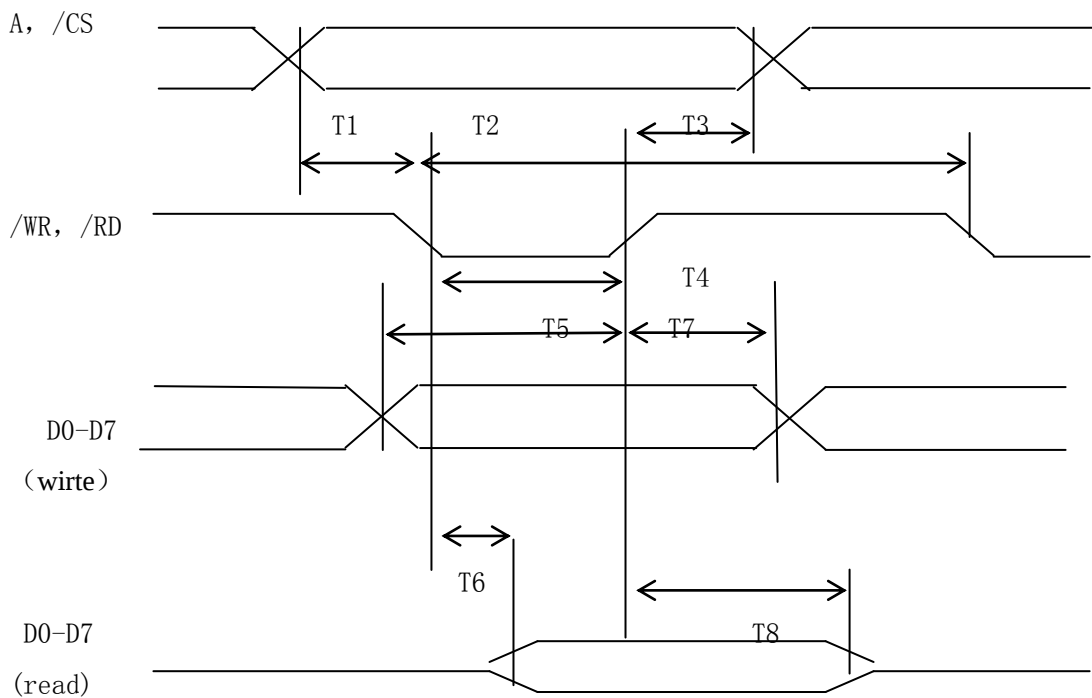
256 色

	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

三、电气信号参数：

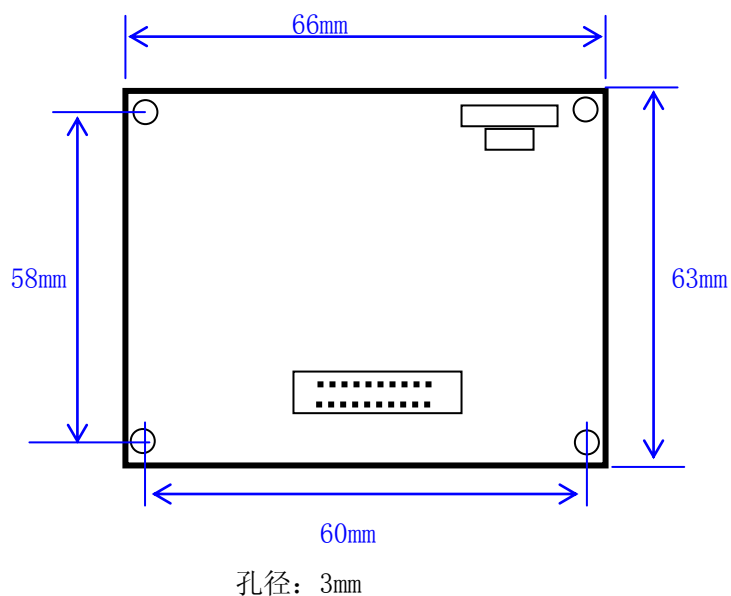
读写时序图：



符号	参数说明	最小	最大	单位
T1	地址建立时间	0	—	ns
T3	地址保持时间	5	—	ns
T2	读写同期	20	—	ns
T4	读写脉冲宽度	10	—	ns
T5	写数据建立时间	15	—	ns
T7	写数据保持时间	5	—	ns
T6	读数据建立时间	5	—	ns
T8	读数据保持时间	10	—	ns

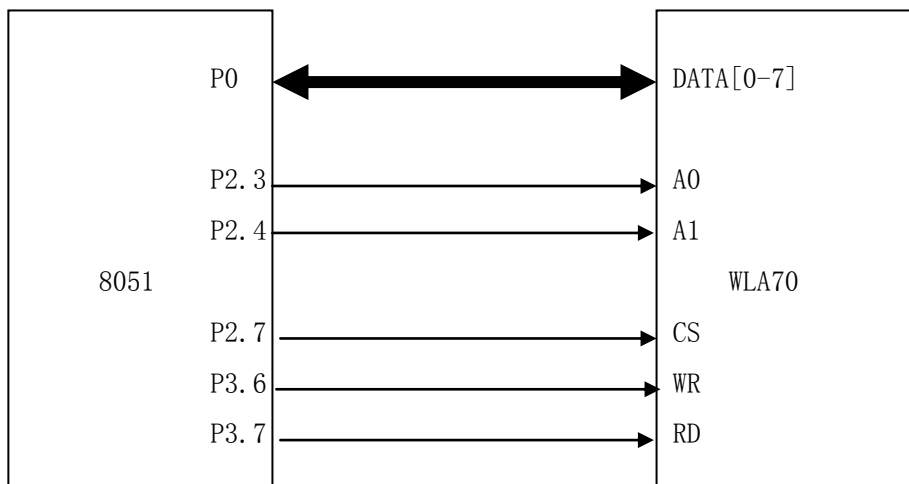
***对数据寄存器 (A1A0=110) 写入操作后一段时间内 (200ns) 不得对控制板进行任何操作。

四、控制板尺寸



五、应用程序

下面提供 51 系列单片机总线和 I/O 口控制方式 C51 演示程序



控制板与 CPU 连接示意图

总线方式比 I/O 方式速度要快很多，建议用户用总线方式

1、C51 程序（总线方式）:

```

#include <reg52.h>
#include <zi.h>
#define uchar unsigned char
#define uint unsigned int
uchar xdata *xreg=(uchar *)0x0000;
uchar xdata *yreg=(uchar *)0x0800;
uchar xdata *dreg=(uchar *)0x1000;
uchar xdata *pswreg=(uchar *)0x1800;

//画实心矩形
void drawrectangle (uchar color,uint x,uchar y,uint l,uchar h,uchar psw)
//psw=0x04|page(页号)
{
    uchar i;
    uint j;
    *yreg=y;
    *xreg=x;
    *pswreg=psw|(x>>8) ;
    for(i=0;i<h;i++)
    {

```

```
        for(j=0;j<1;j++)
        {
            *dreg=color;
        }
    }
}

//画一条直线
void writeline(uchar color,uint x,uchar y,uint l,uchar psw)
    //color 线色
    // x,y 起始坐标
    //l 直线长度
    {
        uint i;
        *pswreg=psw|(x>>8);          // Xin=1:水平直线, 或者 Yin=1:垂直直线,
                                     //Xin=1 且 Yin=1: 45 度斜线

        *xreg=x;
        *yreg=y;
        for(i=0;i<l;i++)
        {
            *dreg=color;
        }
    }

//写汉字程序
void ColorWriteMultiWord(uint x, uchar y, uchar xNum, uchar yNum, uchar *pData,uchar
frontcolor,uchar backcolor,uchar psw)
//write word at Screen
//x,y:字符左上角位置
//xNum=字符一行的点阵数/8, 如 16 点阵字=2
//yNum=字符行数, 如 16 点阵字符=16
//pData 字符首地址
//color 字体色
{
    uchar i, j, dat, bool, k;
    uint n=0;
    for(i=0;i<yNum;i++)
```

```
{
    *yreg=y;
    y++;
    *xreg=x;
    *pswreg=psw|(x>>8);
    for(j = 0;j<xNum;j++)
    {
        dat=pData[n++];
        for(k=0;k<8;k++)
        {
            bool=dat&0x80;           //判断 BIT7, =0, 写入 BACK
            if(bool==0x00) *dreg=backcolor; //!=1, 写入 FRONT
            else *dreg=frontcolor;
            dat<<=1;                 //左移一位
        }
    }
}

//主程序（总线）
void main()
{
    uint i;
    drawrectangle (0xff, 0, 0, 480, 240, 0x54)           //操作页号为 01 显示页号为 01, 清白色
    for(i=0;i<234;i++)
    {writeline(i, 50, i, 100, 0x54);}                    //画 234 个横线, 234 种颜色(00h-e9h)
    ColorWriteMultiWord(196, 117, 2, 16, shi, 0x03, 0xff, 0x54);    //显示 “世龙电子”
    ColorWriteMultiWord(196+16, 117, 2, 16, lon, 0x03, 0xff, 0x54);
    ColorWriteMultiWord(196+32, 117, 2, 16, dian, 0x03, 0xff, 0x54);
    ColorWriteMultiWord(196+48, 117, 2, 16, zi, 0x03, 0xff, 0x54);
}
```

2、C51 程序（I/O 口方式）

```
#include <reg52.h>
#include <zi.h>
#define uchar unsigned char
#define uint unsigned int
```

```
#define xreg 0x00
#define yreg 0x08
#define dreg 0x10
#define pswreg 0x18
sbit a0=P2^3;
sbit a1=P2^4;
sbit cs=P2^7;
sbit wr=P3^6;
sbit rd=P3^7;
void writedata(uchar a,uchar dat) //对控制板写数据
{
    P2=a;          //CS=0, WR=1, a 寄存器地址,
    P0=dat;
    wr=0;
    wr=1;
    cs=1;
}
uchar readdata() //读显示屏当前页号、XY 坐标下的颜色值
{
    uchar dd;
    cs=0;    //cs=0, wr=1, rd=1, a=任意值
    wr=1;
    rd=0;
    dd=P0;
    rd=1;
    cs=1;
    return dd;
}
//画实心矩形
void drawrectangle (uchar color,uint x,uchar y,uint l,uchar h,uchar psw)
    //psw=0x04|page(页号)
    {
        uint i=0,j=0;
        writedata(xreg,x);
        writedata(yreg,y);
```

```
writedata(pswreg, psw|(x>>8));
P0=color;P2=0x10;cs=0;
for(i=0;i<h;i++)
{
    for(j=0;j<l;j++)
    {
        wr=0;wr=1;
    }
}
cs=1;
}
//画一条直线
void writeline(uchar color,uint x,uchar y,uint l,uchar page)
//color 线色
//x,y 起始坐标
// l 线长
{
    uint i;
    writedata(pswreg, psw|(x>>8)); // Xin=1:水平直线, 或者 Yin=1:垂直直线,
                                   //Xin=1 且 Yin=1: 45 度斜线

    writedata(xreg, x);
    writedata(yreg, y);
    P0=color;P2=0x10;cs=0;
    for(i=0;i<l;i++)
    {
        wr=0;wr=1;
    }
    cs=1;
}
//控制板写汉字程序
void ColorWriteMultiWord(uint x, uchar y, uchar xNum, uchar yNum, uchar *pData, uchar
frontcolor, uchar backcolor, uchar psw)
//x,y:字符左上角位置
//xNum=字符一行的点阵数/8, 如 16 点阵字=2
//yNum=字符行数, 如 16 点阵字符=16
```

```
//pData 字符首地址
//color 字体色
{
    uchar i, j, dat, bool, k;
    uint n=0;
    for(i =0;i<yNum;i++)
    {
        writedata(yreg, y);
        y++;
        writedata(xreg, x);
        writedata(pswreg, psw|(x>>8));
        for(j=0;j<xNum;j++)
        {
            dat= pData[n++];
            for(k=0;k<8;k++)
            {
                bool=dat&0x80;           //判断 BIT7, =0, 写入 BACK
                if(bool==0x00) writedata(dreg, backcolor);    //=1, 写入 FRONT
                else writedata(dreg, frontcolor);
                dat<<=1;           //左移一位
            }
        }
    }
}

//主程序
void main()
{
    uint i=0;
    cs=1;wr=1;rd=1;           //初始化 wr, rd, cs 全位高电平
    drawrectangle (0xff, 0, 0, 480, 240, 0x54);    //操作页号为 01 显示页号为 01, 清白色
    for(i=0;i<256;i++)
    { writeline(i, 32+i, 10, 100, 0x58);}    //画 256 个竖线, 256 种颜色(00h~ffh)
    ColorWriteMultiWord(128, 148, 2, 16, shi, 0x03, 0xff, 0x54);    //显示 “世龙电子”
    ColorWriteMultiWord(144, 148, 2, 16, lon, 0x03, 0xff, 0x54);
    ColorWriteMultiWord(160, 148, 2, 16, dian, 0x03, 0xff, 0x54);
}
```

```
ColorWriteMultiWord(178, 148, 2, 16, zi, 0x03, 0xff, 0x50);  
  
}  
  
//字库”世龙电子”16x16点阵(zi.h)  
unsigned char code shi[]={          // “世”16x16点阵  
    0x00, 0x00, 0x00, 0xC0, 0x00, 0xE0, 0x03, 0x60,  
    0x0B, 0x60, 0x0F, 0xFC, 0x3F, 0x44, 0x7B, 0x40,  
    0x2B, 0xC0, 0x09, 0x40, 0x08, 0x00, 0x0F, 0xC0,  
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00  
};  
unsigned char code lon[]={          // “龙”16x16点阵  
    0x00, 0x00, 0x03, 0x80, 0x01, 0xF0, 0x01, 0x00,  
    0x03, 0xE0, 0x07, 0x80, 0x1F, 0x00, 0x0F, 0xA0,  
    0x05, 0x30, 0x09, 0x60, 0x1A, 0xC4, 0x13, 0x84,  
    0x23, 0x0C, 0x01, 0xFC, 0x00, 0x00, 0x00, 0x00  
};  
unsigned char code dian[]={          // “电”16x16点阵  
    0x00, 0x00, 0x06, 0x00, 0x07, 0x00, 0x03, 0x00,  
    0x03, 0xE0, 0x07, 0x10, 0x1B, 0xB0, 0x1F, 0xA0,  
    0x1B, 0x60, 0x0F, 0xC0, 0x0F, 0xC0, 0x02, 0x00,  
    0x01, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00  
};  
unsigned char code zi[]={          // “子”16x16点阵  
    0x00, 0x00, 0x00, 0xE0, 0x03, 0xB0, 0x07, 0x60,  
    0x07, 0xC0, 0x01, 0xC0, 0x00, 0xF0, 0x03, 0xF8,  
    0x0F, 0x20, 0x0C, 0x20, 0x08, 0x20, 0x06, 0x20,  
    0x03, 0xC0, 0x00, 0xC0, 0x00, 0x00, 0x00, 0x00  
};
```