

WLT07Q 液晶显示控制板技术手册 (0901)

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 EPM240 编程实现，性能稳定可靠。WLT07Q 可控制可控制 5.7 吋真彩 TFT 液晶显示屏；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位置。由于显示缓存采用高速 SRAM，对于 320×240 点阵彩色 LCD，可实现 4 页内容叠加显示，读写任意一页不会其他页内容。

控制板中的多点和 8 点写入方式在写字符和清屏操作上比一次写一个点的方式速度快十几倍。

应用：

- 1、在做人机界面时，可同时弹出式三层菜单或页面，擦去弹出式菜单或页面也变得非常容易。
- 2、方便各种仪器多路波形、移动游标、菜单显示。

一、接口定义：

1、CPU 侧接口（双排压线）

引脚	符号	功能	备注
1	VCC	液晶屏逻辑电源	5V
2	VCC	液晶屏逻辑电源	5V
3	NC		
4	DATA0	数据总线	(1)
5	DATA1	数据总线	(1)
6	DATA2	数据总线	(1)
7	DATA3	数据总线	(1)
8	DATA4	数据总线	(1)
9	DATA5	数据总线	(1)
10	DATA6	数据总线	(1)
11	DATA7	数据总线	(1)
12	CS	片选信号，低电平对屏操作有效	(1)
13	WR	写操作信号，低电平有效。	(1)
14	RD	读操作信号，低电平有效。	(1)
15	NC		
16	A0	寄存器地址	(1)
17	A1	寄存器地址	(1)
18	A2	寄存器地址	(1)
19	GND	电源地	
20	GND	电源地	

注(1)：3.3V 或 5V CMOS 电平

(1) CS 与 A0、A1、A2 组合功能如下：

CS	A2A1A0	WR	RD	功能
0	000	0	1	X 低 8 位寄存器
0	001	0	1	X 高 8 位寄存器
0	010	0	1	Y 低 8 位寄存器

0	011	0	1	(无)
0	100	0	1	前景色寄存器
0	101	0	1	背景色寄存器
0	110	0	1	写显示数据寄存器
0	111	0	1	状态控制寄存器
0	×	1	0	读显示数据
1	×	×	×	不选通

说明:

所有寄存器只能写，不能读。

(1) 当 RD 为低电平时，读的是显示屏当前 XY 坐标下的颜色值。

a) X 坐标寄存器: XREG 地址: A2A1A0=000(低), 001(高), 低字节在前, 高字节在后

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
×	×	×	×	×	×	×	X8	X7	X6	X5	X4	X3	X2	X1	X0

b) Y 坐标寄存器: YREG 地址: A=010

D7	D6	D5	D4	D3	D2	D1	D0
L7	L6	L5	L4	L3	L2	L1	L0

c) 状态字寄存器 PSW: 地址 A2A1A0=111

D7	D6	D5	D4	D3	D2	D1	D0
×	×	Prw1	Prw0	YINC	XINC	Wrcon1	Wrcon0

×: 建议为 0

Prw1, Prw0 为读写页号: =00, 顶层; =01, 第二层; =10, 第三层; =11, 底层。

D3 和 D2 位根据用户需要可设置成以下两种功能:

Xinc 为 X 坐标自动增加控制位, =1 时允许 X 自动增加, 写满一行后自动换行, =0 时则禁止增加。单点写屏时, X 自动加 1, 多点 (或 8 点) 写屏时自动加 8;

YINC 为行 (Y 坐标) 自动加 1 控制位, =1 时允许自动加 1, =0 时则禁止加 1;

Wrcon1, Wrcon0 为写入方式:

Wrcon1, Wrcon0=00 时为单点写入方式, 直接将颜色值写入数据寄存器, 而与前景色、背景色寄存器内容无关;

Wrcon1, Wrcon0=01 时为多点写入方式, 将点位信息写入数据寄存器, 如写入数据寄存器为 '01010101b' 则显示 '原色、前景色、原色、前景色、原色、前景色、原色、前景色'。

Wrcon1, Wrcon0=10 时为 8 点写入方式, 将点位信息写入数据寄存器, 如写入数据寄存器为 '01010101b' 则显示 '背景色、前景色、背景色、前景色、背景色、前景色、背景色、前景色'。

****原色:** 显示屏原有颜色, 前景色、背景色是事先存入前景、背景寄存器中的颜色值。

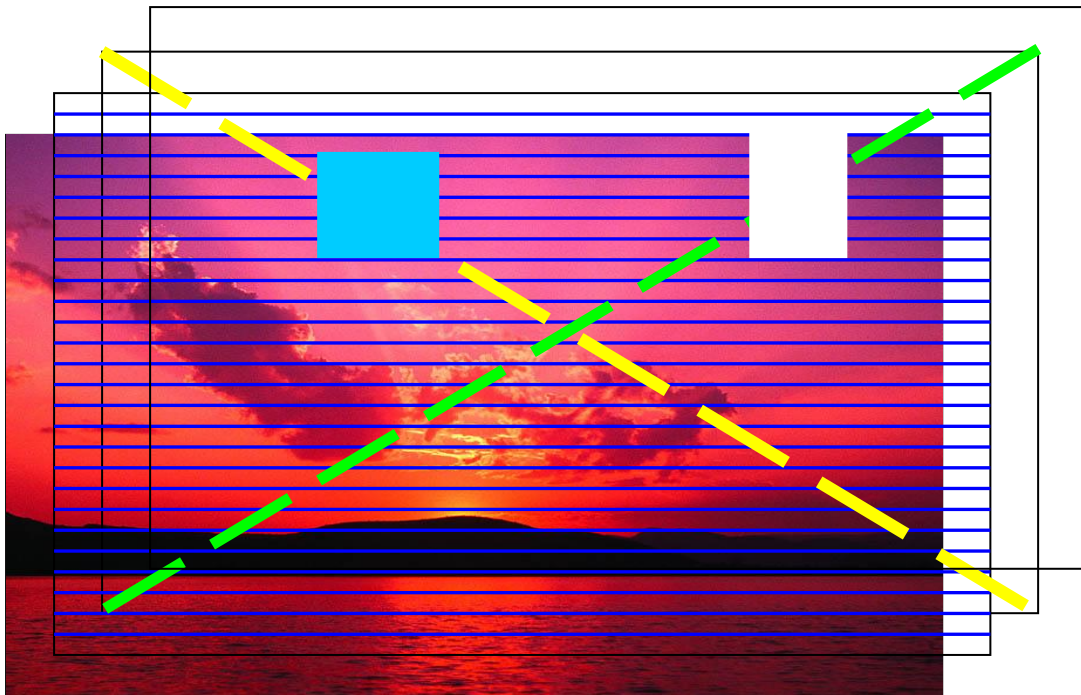
单点写屏时, 将颜色值写入该寄存器, 多点写屏时, 将点位信息写入该寄存器, 如字符点阵信息, 字符颜色为前景寄存器值。8 点写屏时, 字符颜色为前景寄存器值, 字符背景颜色为背景寄存器值。

如下图, 显示屏原有颜色是一幅照片, “多点” 两个字是用多点写入方式写入的, 只写前景色蓝色, 不写背景色; 而 “8 点” 两字是用 8 点写入方式写入的, 前景色为绿色, 背景色为白色, 前景和背景同时写入。



d) 四页叠加方式

控制板将显存分为四页，读写可分别操作，在显示时是叠加在一起显示的。对四页我们分别称之为顶层页、第二层页、第三层页、底层页，底层好比一块画布，上面三层好比三块玻璃，当顶层玻璃擦干净之后，可看见第二层上的图案，第二层也干净的话，可看见第三层的图案，如三层都干净的话，可看见底层图案。在此，我们定义一个新的概念——“透明色”，用透明色清屏，可将玻璃擦干净了。上面三层玻璃可写入 255 种颜色+1 个透明色，底层可写入 256 种颜色。00H 值写入上面三层为透明色，写入底层为黑色。如下图显示，底层为一幅照片，上面三层用透明色‘擦’干净后，分别画上网格、交叉虚线、方块，四层叠加后第一层方块挡住第二层虚线，第二层虚线挡住第三层网格，网格挡住照片，无遮挡部分露出照片。



e) 数据寄存器 DATA: 地址 A2A1A0=110,

前景色寄存器 FRONT:A2A1A0=100,

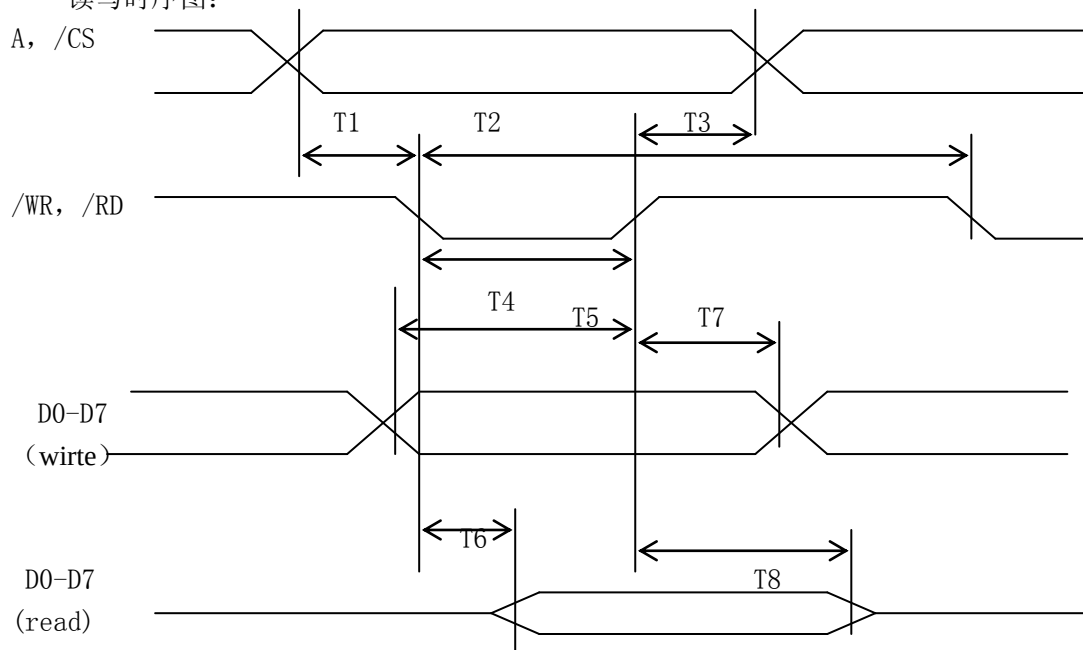
背景色寄存器 BACK:A2A1A0=101

	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

二、电气信号参数：

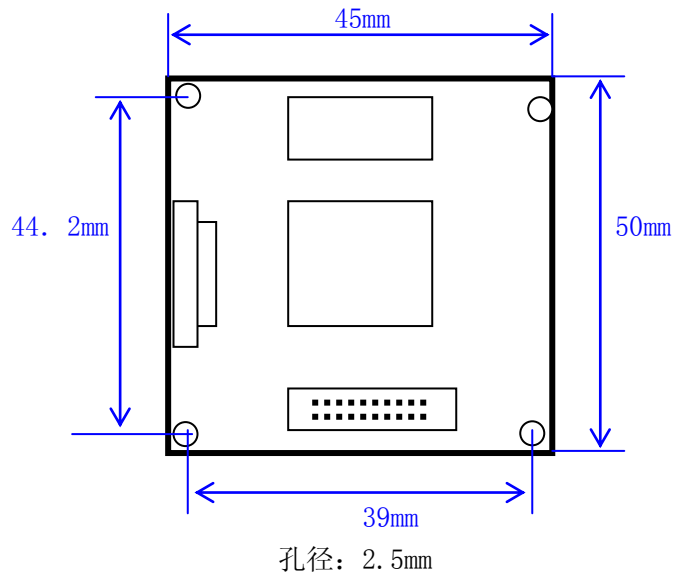
读写时序图：



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

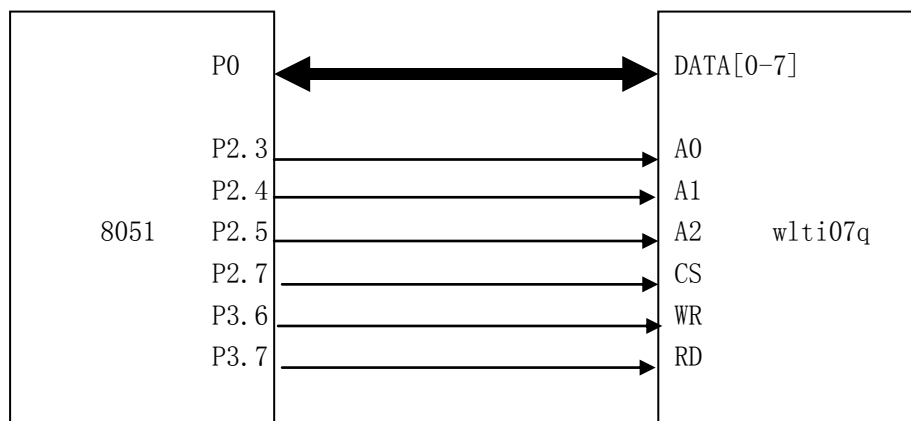
***对数据寄存器 (A2A1A0=110) 写入操作后一段时间内 (单点写入 300ns、多点或 8 点写入 450ns) 不得对控制板进行任何操作, 以便控制板将 8 个点位颜色值写入显存, 如 CPU 时钟很高, 可用插入空指令的办法实现等待。

三、控制板尺寸



四、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 *xlreg=(uchar *)0x0400;           //A=000
uchar xdata *xhreg=(uchar *)0x0c00;           //A=001
uchar xdata *ylreg=(uchar *)0x1400;           //A=010
uchar xdata *frontreg=(uchar *)0x2400;        //A=100
uchar xdata *backreg=(uchar *)0x2c00;         //A=101
uchar xdata *dreg=(uchar *)0x3400;            //A=110
uchar xdata *pswreg=(uchar *)0x3c00;          //A=111
//画实心矩形（宽为 8 的倍数）
void drawrectangle(uchar color,uint,x,uchar y,uchar l,uchar h,uchar psw)
{
    //page 层号
    //矩形宽=l*8
    uchar i,j;
    *frontreg=color;
    *pswreg=psw;           //0x05|page;多点写屏方式，层号 page=0x00,=0x10,=0x20,=0x30
    for(i=0;i<h;i++)
    {
        *ylreg =y;
        *xlreg = x;
        *xhreg=x>>8;
        for(j=0;j<l;j++)
        {

```

```
*dreg=0xff;          //一次写入 8 点, 8 点颜色均为 color
//delay 450ns
}
}
}

//画一条直线____单点写入方式, 一次写 1 点,
void writeline(uchar color,uint x,uchar y,uint l,uchar psw)
//color 线色
//x,y 左边起点坐标
//l 直线长度
{
    uint i
    *pswreg=psw;          // Xin=1:水平直线, 或者 Yin=1:垂直直线,
                        //Xin=1 且 Yin=1: 45 度斜线

    *xlreg=x;
    *xhreg=x>>8;
    *ylreg=y;
    for(i = 0; i < l; i ++)          //
    {
        *dreg=color;
    }
}

//写汉字程序____一次写 8 点
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;
    uint n=0;
    *pswreg=psw;          //psw=0x05|page;多点写屏, psw=0x06|page;8 点写屏。Page 页号
    *frontreg =frontcolor;          //front color
    *backreg=backcolor;          //back color
    for(i = 0; i < yNum; i ++)
    {

        *yreg = y;
```

```
y++;
*xlreg = x;
*xhreg=x>>8;
for(j = 0; j < xNum; j ++)
{
    *dreg = pData[n++];
    //_nop_();          //delay 500ns
}
}
}
//主程序（总线）
void main()
{
    uint i;
    drawrectangle (0xff, 0, 0, 40, 240, 0x35);          //底层清白色
    drawrectangle (0x00, 0, 0, 40, 240, 0x25);          //第三层清透明色
    drawrectangle (0x00, 0, 0, 40, 240, 0x15);          //第二层清透明色
    drawrectangle (0x00, 0, 0, 40, 240, 0x05);          //顶层清透明色
    for(i=0;i<256;i++)
    {writeline(i,i+30,10,100,0x38);}    //在底层用 256 种颜色画 256 条垂直线
    drawrectangle (0xe0, 80, 120, 15, 100, 0x35);      //底层画红色矩形
    drawrectangle (0x1c, 100, 125, 15, 100, 0x25);      //第三层画绿色矩形
    drawrectangle (0xe3, 120, 130, 15, 100, 0x15);      //第二层画紫色矩形
    drawrectangle (0xfc, 140, 135, 15, 100, 0x05);      //顶层画黄色矩形
    //分别顶层上用多点和 8 点写屏方式在显示“世龙电子”
    ColorWriteMultiWord(140, 60, 2, 16, shi, 0x03, 0x00, 0x05);    //多点写入“世龙电子”
    ColorWriteMultiWord(156, 60, 2, 16, lon, 0x03, 0x00, 0x05);
    ColorWriteMultiWord(172, 60, 2, 16, dian, 0x03, 0x00, 0x05);
    ColorWriteMultiWord(188, 60, 2, 16, zi, 0x03, 0x00, 0x05);
    ColorWriteMultiWord(90, 160, 2, 16, shi, 0x03, 0xff, 0x06);    //8 点写入“世龙电子”
    ColorWriteMultiWord(106, 160, 2, 16, lon, 0x03, 0xff, 0x06);
    ColorWriteMultiWord(122, 160, 2, 16, dian, 0x03, 0xff, 0x06);
    ColorWriteMultiWord(138, 160, 2, 16, zi, 0x03, 0xff, 0x06);
    while(1);
}
```

2、C51 程序（I/O 口方式）

```
#include <reg52.h>
#include <zi.h>
#define uchar unsigned char
#define uint unsigned int
#define xlreg 0x00          //A=000
```



```
#define xhreg 0x08          //A=001
#define ylreg 0x10          //A=010
#define frontreg 0x20       //A=100
#define backtreg 0x28       //A=101
#define dreg 0x30           //A=110
#define pswreg 0x38         //A=111
sbit a0=P2^3;
sbit a1=P2^4;
sbit a2=P2^5;
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（宽为 8 的倍数）
void drawrectangle(uchar color,uint,x,uchar y,uchar l,uchar h,uchar psw)
{
    //page 层号
    //矩形宽=l*8
    uchar i,j;
    writedata(frontreg,color);
    writedata(xlreg,x);
    writedata(xhreg,x>>8);
    writedata(ylreg,y);
```

```
writedata(pswreg, psw); //设层号及写屏方式
P2=dreg;
P0=0xff;
for(i=0;i<h;i++)
{
    for(j=0;j<l;j++)
    {
        wr=0;wr=1;          //一次写入 8 点, 8 点颜色均为 color
        //delay 450ns
    }
}
cs=1;
}
画一条直线____单点写入方式, 一次写 1 点
void writeline(uchar color,uint x,uint y,uint l,uchar,uchar page)
//color 线色
//x,y 垂直线顶端坐标
//l 线长
{
    uint i;
    writedata(pswreg, psw);          // Xin=1:水平直线, 或者 Yin=1:垂直直线,
                                     //Xin=1 且 Yin=1: 45 度斜线

    writedata(xlreg, x);
    writedata(xhreg, x>>8);
    writedata(ylreg, y);
    P2=dreg;
    P0=color;
    for(i = 0; i < l; i ++)          //
    {
        wr=0;wr=1;          //一次写入 1 个点
    }
    cs=1;
}
//写汉字程序____一次写 8 点
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
//frontcolor 字体色,backcolor 字体背景色
```

```
{
    uchar i, j, n=0;
    uchar xl, xh;
    xl=x%256;
    xh=x/256;
    writedata(pswreg, psw); //psw=0x05|page;多点写屏, =0x06|page;8点写屏。Page 页号
    writedata(frontreg, frontcolor); //front color
    writedata(backreg, backcolor); //back color
    for(i = 0; i < yNum; i ++)
    {
        writedata(ylreg, y);
        y++;
        writedata(xlreg, xl);
        writedata(xhreg, xh);
        for(j = 0; j < xNum; j ++)
        {
            writedata(dreg, pData[n++]);
            //_nop_(); //500ns
        }
    }
}
//主程序 (I/O 方式)
void main()
{
    uint i;
    cs=1;wr=1;rd=1; //初始化 wr, rd, cs 全位高电平, 以免端口争议
    drawrectangle (0xff, 0, 0, 40, 240, 0x35); //底层清白色
    drawrectangle (0x00, 0, 0, 40, 240, 0x25); //第三层清透明色
    drawrectangle (0x00, 0, 0, 40, 240, 0x15); //第二层清透明色
    drawrectangle (0x00, 0, 0, 40, 240, 0x05); //顶层清透明色
    for(i=0; i<256; i++)
    {writeline(i, i+30, 10, 100, 0x38);} //在底层用 256 种颜色画 256 条垂直线
    drawrectangle (0xe0, 80, 120, 15, 100, 0x35); //底层画红色矩形
    drawrectangle (0x1c, 100, 125, 15, 100, 0x25); //第三层画绿色矩形
    drawrectangle (0xe3, 120, 130, 15, 100, 0x15); //第二层画紫色矩形
    drawrectangle (0xfc, 140, 135, 15, 100, 0x05); //顶层画黄色矩形
    //分别顶层上用多点和 8 点写屏方式在显示“世龙电子”
    ColorWriteMultiWord(140, 60, 2, 16, shi, 0x03, 0x00, 0x05); //多点写入“世龙电子”
    ColorWriteMultiWord(156, 60, 2, 16, lon, 0x03, 0x00, 0x05);
    ColorWriteMultiWord(172, 60, 2, 16, dian, 0x03, 0x00, 0x05);
    ColorWriteMultiWord(188, 60, 2, 16, zi, 0x03, 0x00, 0x05);
    ColorWriteMultiWord(90, 160, 2, 16, shi, 0x03, 0xff, 0x06); //8 点写入“世龙电子”
}
```

```
ColorWriteMultiWord(106, 160, 2, 16, lon, 0x03, 0xff, 0x06);
ColorWriteMultiWord(122, 160, 2, 16, dian, 0x03, 0xff, 0x06);
ColorWriteMultiWord(138, 160, 2, 16, zi, 0x03, 0xff, 0x06);
while(1);
}
//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
};
```