单片机驱动控制 TFT 真彩色液晶屏控制程序
摘要:武汉谷鑫科技有限公司专业从事单片机驱动控制 TFT 真彩色液晶屏,www.mcutech.cn,

```
sfr AUXR = 0x8e;//////////////////////////////////注意增加该语句
#include <string.h>    //这个头文件是为外部存储数据加的
#include <reg51.h>
#include <absacc.h>

//touch screen
#include <intrins.h>
sbit DCLK=P3^5;
sbit CS=P3^4;
sbit DIN=P3^3;
sbit DOUT=P1^7;
sbit BUSY=P1^6;
sbit IRQ=P3^2;
//touch screen

sbit P13=0x93;
#define X_ADDR XBYTE[0x0000]
#define Y_ADDR XBYTE[0x0100]
#define CMD XBYTE[0x0200]
#define DAT XBYTE[0x0300]

#define PORT0 XBYTE[0x0000]
#define PORT1 XBYTE[0x0100]
#define PORT2 XBYTE[0x0200]
#define PORT3 XBYTE[0x0300]


//touch screen
long X=0,Y=0;
long X2=0,X3=0,Y2=0,Y3=0;
char jz0=0,jz1=0,jz2=0,key1,key2;

int xxx,yyy,X1=0,Y1=0,Xlast=0,Ylast=0;


unsigned char code hzk[2000] =
{
/*--  文字:  色  --*/
/*--  楷体_GB231222;  此字体下对应的点阵为：宽 x 高=31x29    --*/
```

/*-- 宽度不是 8 的倍数，现调整为：宽度 x 高度=32x29 --*/
0x00,0x18,0x00,0x00,0x00,0x1E,0x00,0x00,0x00,0x1E,0x00,0x00,0x00,0x3C,0xC0,0x00,
0x00,0x7B,0xF0,0x00,0x00,0x7F,0xF0,0x00,0x00,0xFF,0xE0,0x00,0x01,0xE3,0xC0,0x00,
0x03,0x87,0x80,0x00,0x07,0x0F,0x60,0x00,0x00,0x1F,0xF8,0x00,0x06,0xFF,0xFC,0x00,
0x07,0xFC,0x78,0x00,0x07,0x9C,0xF0,0x00,0x07,0x1C,0xF0,0x00,0x07,0x1F,0xF0,0x00,
0x07,0x3F,0xF0,0x00,0x07,0xFE,0x00,0x60,0x07,0xC0,0x00,0xE0,0x07,0x00,0x00,0xE0,
0x07,0x00,0x00,0xE0,0x07,0x80,0x00,0xE0,0x07,0x80,0x01,0xE0,0x03,0xC0,0x03,0xF0,
0x03,0xFF,0xFF,0xF0,0x01,0xFF,0xFF,0xC0,0x00,0x7F,0xFE,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,

/*-- 文字: 彩 --*/
/*-- 楷体_GB231222; 此字体下对应的点阵为：宽 x 高=31x29 --*/
/*-- 宽度不是 8 的倍数，现调整为：宽度 x 高度=32x29 --*/
0x00,0x00,0x00,0x00,0x00,0x0C,0x00,0x00,0x00,0x0E,0x00,0x00,0x00,0x1F,0x00,0xE0,
0x00,0x7E,0x01,0xF0,0x01,0xF9,0xC1,0xE0,0x01,0xF1,0xE3,0xC0,0x00,0x7B,0xC7,0x80,
0x07,0xBF,0x8F,0x00,0x07,0x98,0x1E,0x00,0x03,0xF8,0x3D,0x80,0x00,0x3F,0xF1,0xC0,
0x00,0x3F,0xC3,0xE0,0x0F,0xFC,0x07,0xC0,0x7F,0xF8,0x0F,0x00,0x3C,0xF8,0x1E,0x00,
0x00,0xFF,0x79,0xC0,0x01,0xFF,0xE1,0xE0,0x03,0xF9,0x01,0xE0,0x07,0xB8,0x01,0xE0,
0x0F,0x38,0x03,0xC0,0x1C,0x38,0x07,0x80,0x38,0x78,0x0F,0x00,0x00,0x78,0x1E,0x00,
0x00,0x78,0x3C,0x00,0x00,0x30,0xF8,0x00,0x00,0x03,0xE0,0x00,0x00,0x07,0x80,0x00,
0x00,0x00,0x00,0x00,

/*-- 文字: 测 --*/
/*-- 楷体_GB231222; 此字体下对应的点阵为：宽 x 高=31x29 --*/
/*-- 宽度不是 8 的倍数，现调整为：宽度 x 高度=32x29 --*/
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,0xE0,
0x00,0x00,0x01,0xE0,0x0F,0x01,0x81,0xE0,0x0F,0xFF,0xC1,0xE0,0x07,0xFF,0xE0,0xE0,
0x03,0xF1,0xD8,0xE0,0x00,0x7D,0xDC,0xE0,0x00,0x7F,0xDE,0xE0,0x3C,0x7F,0xDE,0xE0,
0x3C,0x7F,0xDE,0xE0,0x1F,0xFF,0xCE,0xE0,0x0D,0xFF,0xCE,0xE0,0x03,0xFF,0xCE,0xE0,
0x03,0xFF,0xCE,0xE0,0x07,0x7F,0xCC,0xE0,0x07,0x7E,0xCC,0xE0,0x0F,0x1F,0x00,0xE0,
0x3E,0x3F,0x80,0xE0,0x3E,0x3B,0xC0,0xE0,0x1C,0x79,0xE6,0xE0,0x0C,0xF0,0xE7,0xE0,
0x01,0xC0,0x03,0xE0,0x03,0x00,0x01,0xE0,0x00,0x00,0x01,0xC0,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,

/*-- 文字: 试 --*/
/*-- 楷体_GB231222; 此字体下对应的点阵为：宽 x 高=31x29 --*/
/*-- 宽度不是 8 的倍数，现调整为：宽度 x 高度=32x29 --*/
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,0xE0,0x00,0x00,0x00,0xF0,0x00,
0x03,0x80,0xF0,0x00,0x03,0xE0,0xF7,0xC0,0x01,0xF0,0xF7,0xC0,0x00,0xF0,0x71,0xC0,
0x00,0x00,0x70,0x00,0x00,0x00,0x77,0x00,0x00,0x00,0x7F,0x80,0x03,0x83,0xFE,0x00,
0x0F,0xDF,0xF8,0x00,0x7F,0xCE,0x38,0x00,0x3F,0xC0,0x38,0x00,0x03,0x83,0xFC,0x00,
0x03,0x9F,0xDC,0x00,0x03,0x9F,0x9C,0x00,0x03,0x83,0x9E,0x00,0x03,0xB3,0xBE,0x00,
0x07,0xF3,0xFF,0x38,0x07,0xEF,0xC7,0xB8,0x07,0xFF,0x07,0xF8,0x0F,0xBC,0x03,0xF8,

0x07,0x00,0x01,0xF8,0x06,0x00,0x00,0xF8,0x00,0x00,0x00,0x78,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,

/*-- 文字: 红 --*/
/*-- 楷体_GB231222; 此字体下对应的点阵为：宽 x 高=31x29 --*/
/*-- 宽度不是 8 的倍数，现调整为：宽度 x 高度=32x29 --*/
0x00,0x00,0x00,0x00,0x01,0xC0,0x00,0x00,0x01,0xE0,0x00,0x00,0x01,0xF0,0x00,0x00,
0x01,0xE0,0x00,0x00,0x03,0xC0,0x00,0x00,0x03,0xC0,0x00,0x00,0x07,0xBC,0x00,0x00,
0x0F,0x3E,0x3F,0xE0,0x0E,0x3F,0xFF,0xC0,0x1E,0x7B,0xFC,0x00,0x3F,0xF0,0x3C,0x00,
0x3F,0xE0,0x3C,0x00,0x01,0xC0,0x3C,0x00,0x03,0xCC,0x38,0x00,0x0F,0xFC,0x38,0x00,
0x1F,0xF0,0x38,0x00,0x1F,0x80,0x38,0x00,0x0E,0x0C,0x38,0x00,0x00,0x3C,0x38,0x00,
0x01,0xF0,0x3F,0xF8,0x07,0xEF,0xFF,0xF8,0x3F,0xBF,0xF8,0x00,0x3F,0x0E,0x00,0x00,
0x1C,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,

/*-- 文字: 绿 --*/
/*-- 楷体_GB231222; 此字体下对应的点阵为：宽 x 高=31x29 --*/
/*-- 宽度不是 8 的倍数，现调整为：宽度 x 高度=32x29 --*/
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x03,0x80,0x0C,0x00,0x03,0xC0,0xFE,0x00,
0x03,0xC7,0xFF,0x00,0x07,0x83,0x0E,0x00,0x07,0x30,0x1E,0x00,0x0F,0x7F,0xFC,0x00,
0x0E,0x7B,0xDC,0x00,0x3C,0xF0,0x1F,0xC0,0x3F,0xE0,0x7F,0xE0,0x3F,0xFF,0xFF,0xE0,
0x03,0xBF,0xE0,0x00,0x07,0x00,0xE7,0x00,0x0F,0xF0,0xEF,0x80,0x1F,0xDE,0xFF,0x00,
0x1F,0x0F,0xFC,0x00,0x0C,0x07,0xF0,0x00,0x00,0xE3,0xFC,0x00,0x03,0xCF,0xFF,0x00,
0x1F,0x9E,0xEF,0xC0,0x7F,0x7C,0xE3,0xFC,0x7C,0x78,0xE1,0xF8,0x30,0x03,0xE0,0x00,
0x00,0x03,0xE0,0x00,0x00,0x01,0xE0,0x00,0x00,0x00,0xE0,0x00,0x00,0x00,0xC0,0x00,
0x00,0x00,0x00,0x00,

/*-- 文字: 蓝 --*/
/*-- 楷体_GB231222; 此字体下对应的点阵为：宽 x 高=31x29 --*/
/*-- 宽度不是 8 的倍数，现调整为：宽度 x 高度=32x29 --*/
0x00,0x00,0x00,0x00,0x00,0x00,0xF0,0x00,0x00,0xF0,0x70,0x00,0x00,0x78,0xF0,0x00,
0x00,0x78,0xFF,0x80,0x00,0x7F,0xFF,0xC0,0x1F,0xFF,0xE0,0x00,0x0F,0xF9,0xC0,0x00,
0x00,0x31,0xE0,0x00,0x00,0x38,0xF0,0x00,0x03,0x38,0xF0,0x00,0x03,0xB8,0xFF,0x00,
0x03,0xB9,0xFF,0x00,0x03,0xB9,0xC0,0x00,0x03,0xBB,0xF8,0x00,0x03,0xBB,0x3C,0x00,
0x03,0xBE,0x1C,0x00,0x01,0xB8,0x0C,0x00,0x00,0x19,0xFE,0x00,0x07,0xFF,0xFF,0x00,
0x03,0xFD,0xDF,0x00,0x03,0xDD,0xDE,0x00,0x01,0xDD,0xDC,0x00,0x01,0xDD,0xDF,0xC0,
0x01,0xDF,0xFF,0xF0,0x7F,0xFF,0xFF,0xF0,0x7F,0xC0,0x00,0x00,0x18,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,

/*-- 文字: 第 --*/
/*-- 楷体_GB231222; 此字体下对应的点阵为：宽 x 高=31x29 --*/
/*-- 宽度不是 8 的倍数，现调整为：宽度 x 高度=32x29 --*/
0x00,0x00,0x60,0x00,0x00,0xE0,0x78,0x00,0x00,0xF0,0x7F,0x80,0x00,0xFF,0xFF,0x80,

0x01,0xFF,0xF8,0x00,0x01,0xC1,0xF8,0x00,0x03,0xFB,0xBC,0x00,0x07,0xFF,0x1C,0x00,
0x0F,0x18,0x70,0x00,0x1C,0x1F,0xFC,0x00,0x00,0xFF,0xBC,0x00,0x00,0x07,0x78,0x00,
0x00,0x07,0xF0,0x00,0x03,0xFF,0xF0,0x00,0x03,0xF7,0x00,0x00,0x03,0x87,0x1F,0x80,
0x03,0x9F,0xFF,0xC0,0x07,0xFF,0xC7,0x80,0x07,0xFF,0x07,0x80,0x07,0x3F,0x07,0x80,
0x00,0x7F,0x3F,0x00,0x01,0xF7,0x3F,0x00,0x03,0xE7,0x0F,0x00,0x0F,0x87,0x0E,0x00,
0x3E,0x07,0x00,0x00,0x30,0x07,0x00,0x00,0x00,0x07,0x00,0x00,0x00,0x07,0x00,0x00,
0x00,0x06,0x00,0x00,

/*-- 文字: 级 --*/
/*-- 楷体_GB231222; 此字体下对应的点阵为：宽 x 高=31x29 --*/
/*-- 宽度不是 8 的倍数，现调整为：宽度 x 高度=32x29 --*/
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,0xC0,0x00,0x00,0x01,0xE0,0x00,0x00,
0x01,0xE0,0x1F,0x00,0x03,0xC1,0xFF,0x80,0x03,0xCF,0xFF,0x00,0x03,0xBF,0xCE,0x00,
0x07,0x3D,0xEE,0x00,0x0F,0x79,0xDC,0x00,0x0E,0xF1,0xD8,0x00,0x1F,0xE3,0xFE,0x00,
0x1F,0xC3,0x9F,0x00,0x03,0xC3,0x8F,0x00,0x07,0xFF,0xCF,0x00,0x0F,0xF7,0xEE,0x00,
0x0F,0x87,0x7E,0x00,0x0E,0x1E,0x3E,0x00,0x00,0xFE,0x3E,0x00,0x03,0xFC,0x7F,0x00,
0x0F,0xF8,0xFF,0xC0,0x7F,0x79,0xE7,0xF0,0x3C,0xF7,0x83,0xFC,0x00,0xC6,0x01,0xFC,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,

/*-- 文字: 亮 --*/
/*-- 楷体_GB231222; 此字体下对应的点阵为：宽 x 高=31x29 --*/
/*-- 宽度不是 8 的倍数，现调整为：宽度 x 高度=32x29 --*/
0x00,0x00,0x00,0x00,0x00,0x0F,0x00,0x00,0x00,0x07,0x80,0x00,0x00,0x07,0x80,0x00,
0x00,0x01,0xFF,0x00,0x00,0xFF,0xFF,0x00,0x07,0xFE,0x00,0x00,0x00,0x01,0xE0,0x00,
0x00,0x7F,0xF0,0x00,0x00,0x7C,0xF0,0x00,0x00,0x38,0xE0,0x00,0x00,0x3F,0xE0,0x00,
0x00,0x3E,0x03,0x00,0x06,0x01,0xFF,0xC0,0x0F,0xFF,0xFF,0xE0,0x0F,0xF8,0x07,0xE0,
0x0E,0x03,0x8F,0x00,0x1E,0x7F,0xC0,0x00,0x1C,0x3F,0xC0,0x00,0x1C,0x3B,0xC0,0x00,
0x00,0x3B,0x80,0xC0,0x00,0x7B,0x80,0xE0,0x00,0x73,0x80,0xE0,0x00,0xF3,0xC1,0xE0,
0x01,0xE1,0xE3,0xE0,0x07,0xC1,0xFF,0xE0,0x0F,0x00,0xFF,0xC0,0x18,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,

/*-- 文字: 度 --*/
/*-- 楷体_GB231222; 此字体下对应的点阵为：宽 x 高=31x29 --*/
/*-- 宽度不是 8 的倍数，现调整为：宽度 x 高度=32x29 --*/
0x00,0x00,0x00,0x00,0x00,0x07,0x80,0x00,0x00,0x07,0xC0,0x00,0x00,0x03,0xC0,0x00,
0x00,0x01,0xFF,0x00,0x00,0x0F,0xFE,0x00,0x03,0xFF,0xF0,0x00,0x01,0xF0,0x78,0x00,
0x01,0xDE,0x70,0x00,0x03,0xCE,0x7F,0x80,0x03,0xCF,0xFF,0x80,0x03,0xFF,0xF0,0x00,
0x03,0xFE,0x70,0x00,0x03,0x8F,0xF8,0x00,0x03,0x8F,0xF0,0x00,0x07,0x8E,0x30,0x00,
0x07,0x83,0xF8,0x00,0x07,0x1F,0xFC,0x00,0x0F,0x0E,0x70,0x00,0x0E,0x3C,0xF0,0x00,
0x0E,0x1F,0xE0,0x00,0x1E,0x07,0xE0,0x00,0x1C,0x07,0xF0,0x00,0x38,0x1F,0xFE,0x00,
0x70,0x7E,0x7F,0xE0,0xE3,0xF8,0x1F,0xFC,0xC0,0x00,0x0F,0xF0,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,

```
/*--  文字:  :  --*/
/*--  楷体_GB231222;  此字体下对应的点阵为：宽 x 高=31x29  --*/
/*--  宽度不是 8 的倍数，现调整为：宽度 x 高度=32x29  --*/
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x0C,0x00,0x00,0x00,
0x1F,0x00,0x00,0x00,0x1F,0x00,0x00,0x00,0x1F,0x00,0x00,0x00,0x0C,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x1E,0x00,0x00,0x00,0x1F,0x00,0x00,0x00,
0x1F,0x00,0x00,0x00,0x1E,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,
};

unsigned char code szk[1000] =
{
/*--  文字:  0  --*/
/*--  楷体_GB231222;  此字体下对应的点阵为：宽 x 高=16x29  --*/
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x03,0xE0,0x07,0xF0,0x0F,0x70,0x0E,0x38,
0x1E,0x38,0x1E,0x3C,0x1E,0x3C,0x1E,0x3C,0x1E,0x3C,0x1E,0x3C,0x1E,0x3C,0x1E,0x3C,
0x1E,0x3C,0x1E,0x3C,0x1E,0x3C,0x1E,0x3C,0x0E,0x38,0x0F,0x78,0x07,0x70,0x07,0xE0,
0x01,0xC0,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

/*--  文字:  1  --*/
/*--  楷体_GB231222;  此字体下对应的点阵为：宽 x 高=16x29  --*/
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,0x80,0x07,0x80,0x1F,0x80,0x1F,0x80,
0x03,0x80,0x03,0x80,0x03,0x80,0x03,0x80,0x03,0x80,0x03,0x80,0x03,0x80,0x03,0x80,
0x03,0x80,0x03,0x80,0x03,0x80,0x03,0x80,0x03,0x80,0x03,0x80,0x07,0xC0,0x1F,0xF0,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

/*--  文字:  2  --*/
/*--  楷体_GB231222;  此字体下对应的点阵为：宽 x 高=16x29  --*/
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x07,0xC0,0x0F,0xE0,0x1F,0xF0,0x1C,0xF0,
0x38,0xF0,0x38,0x70,0x00,0x70,0x00,0x70,0x00,0xF0,0x00,0xF0,0x00,0xE0,0x01,0xE0,
0x03,0xC0,0x03,0x80,0x07,0x80,0x0F,0x00,0x0E,0x18,0x1E,0x78,0x3F,0xF8,0x3F,0xF0,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

/*--  文字:  3  --*/
/*--  楷体_GB231222;  此字体下对应的点阵为：宽 x 高=16x29  --*/
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x03,0xE0,0x07,0xF0,0x0E,0xF8,0x0C,0x78,
0x0C,0x38,0x00,0x38,0x00,0x70,0x00,0x70,0x00,0xE0,0x03,0xF0,0x03,0xF8,0x00,0x78,
0x00,0x3C,0x00,0x3C,0x00,0x3C,0x00,0x3C,0x00,0x38,0x1E,0x38,0x1F,0xF0,0x0F,0xE0,
0x07,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
```

/*-- 文字: 4 --*/
/*-- 楷体_GB231222; 此字体下对应的点阵为：宽 x 高=16x29 --*/
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x70,0x00,0xF0,0x00,0xF0,0x01,0xF0,
0x03,0xF0,0x03,0xF0,0x07,0xF0,0x07,0xF0,0x0E,0xF0,0x1C,0xF0,0x1C,0xF0,0x38,0xF0,
0x3F,0xF8,0x3F,0xF8,0x3F,0xF8,0x00,0xF0,0x00,0xF0,0x00,0xF0,0x00,0xF0,0x00,0xF0,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

/*-- 文字: 5 --*/
/*-- 楷体_GB231222; 此字体下对应的点阵为：宽 x 高=16x29 --*/
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x03,0xF8,0x07,0xF8,0x07,0xF0,0x06,0x00,
0x0E,0x00,0x0E,0x00,0x1F,0x00,0x1F,0xC0,0x0F,0xE0,0x01,0xF0,0x00,0x70,0x00,0x78,
0x00,0x38,0x00,0x38,0x00,0x38,0x00,0x30,0x00,0x70,0x1C,0x70,0x3F,0xE0,0x1F,0xC0,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

/*-- 文字: 6 --*/
/*-- 楷体_GB231222; 此字体下对应的点阵为：宽 x 高=16x29 --*/
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x78,0x01,0xF0,0x03,0xC0,0x07,0x80,
0x0F,0x00,0x0E,0x00,0x1E,0x00,0x1E,0x00,0x1F,0xE0,0x3F,0xF0,0x3C,0x78,0x3C,0x38,
0x3C,0x38,0x3C,0x38,0x3C,0x38,0x3C,0x38,0x1C,0x78,0x1E,0x78,0x0F,0xF0,0x0F,0xE0,
0x03,0x80,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

/*-- 文字: 7 --*/
/*-- 楷体_GB231222; 此字体下对应的点阵为：宽 x 高=16x29 --*/
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x0F,0xF8,0x1F,0xF8,0x1F,0xF8,0x18,0x78,
0x38,0x70,0x00,0x70,0x00,0x70,0x00,0xF0,0x00,0xE0,0x00,0xE0,0x00,0xE0,0x01,0xC0,
0x01,0xC0,0x01,0xC0,0x03,0xC0,0x03,0x80,0x03,0x80,0x03,0x80,0x07,0x80,0x07,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

/*-- 文字: 8 --*/
/*-- 楷体_GB231222; 此字体下对应的点阵为：宽 x 高=16x29 --*/
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x07,0xC0,0x0F,0xF0,0x1C,0x70,0x3C,0x78,
0x3C,0x78,0x3C,0x78,0x1E,0x78,0x1E,0xF0,0x0F,0xE0,0x07,0xC0,0x07,0xC0,0x0F,0xE0,
0x1E,0xF0,0x1C,0x78,0x3C,0x78,0x3C,0x38,0x3C,0x38,0x3C,0x78,0x1E,0xF0,0x0F,0xE0,
0x03,0x80,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

/*-- 文字: 9 --*/
/*-- 楷体_GB231222; 此字体下对应的点阵为：宽 x 高=16x29 --*/
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x07,0xC0,0x0F,0xE0,0x1C,0xF0,0x1C,0x78,
0x3C,0x78,0x3C,0x78,0x3C,0x38,0x3C,0x38,0x3C,0x38,0x1C,0x78,0x1E,0xF8,0x0F,0xF8,
0x07,0xF8,0x00,0xF0,0x00,0xF0,0x00,0xE0,0x01,0xE0,0x03,0xC0,0x07,0x80,0x3E,0x00,
0x10,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
};

```
unsigned char code plus[300] =
{
/*--  文字:  +  --*/
/*--  楷体_GB231220;  此字体下对应的点阵为：宽 x 高=14x27   --*/
/*--  宽度不是 8 的倍数，现调整为：宽度 x 高度=16x16 --*/
0x00,0x00,0x01,0x80,0x01,0x80,0x01,0x80,0x01,0x80,0x01,0x80,0x01,0x80,0x7F,0xFE,
0x7F,0xFE,0x01,0x80,0x01,0x80,0x01,0x80,0x01,0x80,0x01,0x80,0x01,0x80,0x00,0x00,
};

//TFT8060-8A
unsigned char code picture[800]={//zhongda
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0xC0, 0x00, 0x60, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x01, 0x80, 0x00, 0xC0, 0x03, 0xFF, 0x03, 0xFF, 0xC0,
0x00, 0x01, 0x80, 0x00, 0xC0, 0x06, 0x06, 0x00, 0x00, 0x00,
0x00, 0x7F, 0xFC, 0x01, 0x80, 0x06, 0x06, 0x00, 0x00, 0x00,
0x00, 0x63, 0x0C, 0x7F, 0xFF, 0x07, 0xFC, 0x00, 0x00, 0x00,
0x00, 0x63, 0x18, 0x01, 0x80, 0x0C, 0x0C, 0x3F, 0xFF, 0xC0,
0x00, 0xC6, 0x18, 0x03, 0x80, 0x0F, 0xFC, 0x00, 0x60, 0x00,
0x00, 0xC6, 0x30, 0x03, 0x80, 0x03, 0x60, 0x07, 0xD8, 0x00,
0x00, 0xFF, 0xF0, 0x07, 0x80, 0x36, 0xCC, 0x06, 0xD8, 0x00,
0x00, 0x0C, 0x00, 0x0D, 0x80, 0x36, 0xDC, 0x0C, 0xCC, 0x00,
0x00, 0x0C, 0x00, 0x0C, 0xC0, 0x1E, 0xF0, 0x39, 0x86, 0x00,
```

0x00, 0x0C, 0x00, 0x38, 0xE0, 0x1D, 0xE0, 0x61, 0x86, 0x00,
0x00, 0x18, 0x00, 0x60, 0x70, 0x0D, 0xC0, 0xC1, 0x80, 0x00,
0x00, 0x18, 0x01, 0xC0, 0x31, 0xFF, 0xFC, 0x07, 0x00, 0x00,
0x00, 0x00, 0x07, 0x00, 0x00, 0x00, 0x00, 0x06, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x03, 0xCF, 0xFE, 0x38, 0x20, 0x47, 0xB8, 0x00, 0x00,
0x00, 0x01, 0x87, 0x32, 0x60, 0x60, 0xCC, 0xD8, 0x00, 0x00,
0x00, 0x00, 0xC4, 0x30, 0xC0, 0xE1, 0xCC, 0xD8, 0x00, 0x00,
0x00, 0x00, 0xC4, 0x31, 0xF0, 0xE1, 0xCE, 0x9E, 0x00, 0x00,
0x00, 0x00, 0x68, 0x31, 0x99, 0x62, 0xC7, 0x9B, 0x00, 0x00,
0x00, 0x00, 0x68, 0x31, 0x9A, 0x64, 0xCD, 0xDB, 0x00, 0x00,
0x00, 0x00, 0x30, 0x31, 0x9B, 0xF7, 0xEC, 0xDB, 0x00, 0x00,
0x00, 0x00, 0x30, 0x31, 0x98, 0x60, 0xCC, 0xDB, 0x00, 0x00,
0x00, 0x00, 0x30, 0x78, 0xF0, 0x60, 0xC7, 0x96, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

```
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00
  };
//TFT8060-8A



delay(unsigned char i)
{
      while(i--);
}

void start() //SPI 开始
{
      unsigned int z;
      DCLK=0;
      CS=1;for(z=0;z<50;z++) _nop_();
      DIN=1;for(z=0;z<50;z++) _nop_();
      DCLK=1;
      CS=0;
}

WriteCharTo7843(unsigned char num) //SPI 写数据
{
      unsigned int z;
      unsigned char count=0;
      DCLK=0;
      for(count=0;count<8;count++)
      {
            num<<=1;
            DIN=CY;
            DCLK=0;for(z=0;z<50;z++) _nop_();//上升沿有效
            DCLK=1;for(z=0;z<50;z++) _nop_();
      }
}


ReadFromCharFrom7843() //SPI  读数据
{
```

```
    unsigned int z;
    unsigned char count=0;
    unsigned int Num=0;
    for(count=0;count<12;count++)
    {
    //    Num<<=1;     //LY 这是原程序的
        DCLK=0;for(z=0;z<50;z++) _nop_();//下降沿有效   LY 认为是上升沿
        DCLK=1;for(z=0;z<50;z++) _nop_();
        if(DOUT) Num++;
        Num<<=1;     //LY 我觉得应该是先读后自加
    }
    return(Num);
}



void ZhongDuan() interrupt 0 //外部中断 0  用来接受键盘发来的数据
{
    unsigned int z;
    for(z=0;z<5000;z++) delay(25);
    //delay(20000);//中断后延时以消除抖动，使得采样数据更准确
    //for(y=0;y<5000;y++);//延时 16.6 毫秒

    IE=0;

    start(); //启动 SPI
    //while(BUSY); //如果 BUSY 信号不好使可以删除不用
    delay(35);
    WriteCharTo7843(0x90); //送控制字  10010000  即用差分方式读 X 坐标  详细请见有关资
料
    //while(BUSY); //如果 BUSY 信号不好使可以删除不用
    delay(35);
    DCLK=1;for(z=0;z<50;z++) _nop_();
    DCLK=0;for(z=0;z<50;z++) _nop_();
    X=ReadFromCharFrom7843();
    delay(260);

    WriteCharTo7843(0xD0);  //送控制字  11010000  即用差分方式读  Y  坐标  详细请见有关
资料
    delay(35);
    DCLK=1;for(z=0;z<50;z++) _nop_();
    DCLK=0;for(z=0;z<50;z++) _nop_();
    Y=ReadFromCharFrom7843();
```

```
    delay(150);
    CS=1;

    for(z=0;z<3000;z++) delay(25);

    Y1=(X-X2)*yyy/(X3-X2)+20;       //不改硬件改软件就改这
    X1=(Y-Y2)*xxx/(Y3-Y2)+20;       //不改硬件改软件就改这

//  Y1=(X2-X)*yyy/(X2-X3)+20; //其实取点的数值和我想象的正好相反
//  X1=(Y2-Y)*xxx/(Y2-Y3)+20; //其实取点的数值和我想象的正好相反

//  Y1=(X-X3)*yyy/(X2-X3)+20; //其实取点的数值和我想象的正好相反
//  X1=(Y-Y3)*xxx/(Y2-Y3)+20; //其实取点的数值和我想象的正好相反

//  Y1=(4096-X)*yyy/4096;       //直接用的实参，但不知道为这么 X 取值是反的，所以就用 4096-X
//  X1=(4096-Y)*xxx/4096;       //直接用的实参，但不知道为这么 Y 取值是反的，所以就用 4096-Y


    if(!jz0)
    {
        X2=X;
        Y2=Y;
    }
    if(!jz1)
    {
        X3=X;
        Y3=Y;
    }


    if(IRQ==0&&jz2) //触摸结束瞬间，提笔后，最后个数据是错误的，就是那跑偏的点，好像是这的问题吧!!!
    {
        if(X1>7&&X1<xxx+33&&Y1>7&&Y1<yyy+33)          //0<=X1-8<=xxx+4+20
        {                                                   //8<=X1<=632
            //消除上一个点
            unsigned char wn,wy;
            //单点写模式下刷某区域
```

```
                CMD = 0x00;
                for (wy=0; wy<16; wy++)
                {
                    X_ADDR = Xlast;
                    X_ADDR = Xlast/256;
                    Y_ADDR = wy+Ylast;
                    if( ((key2==1)|(key2==2))==0 ) Y_ADDR = (wy+Ylast)/256;
                    for ( wn=0; wn<16; wn++) DAT = 0;
                }
                //显示当前触摸点
                //8 点写汉字
                CMD = 0x02;DAT=0xe0;
                CMD = 0x04;DAT=0;
                CMD = 0x01;
                for (wy=0; wy<16; wy++)
                {
                    X_ADDR = X1-8;
                    X_ADDR = (X1-8)/256;
                    Y_ADDR = Y1-8+wy;
                    if( ((key2==1)|(key2==2))==0 ) Y_ADDR = (Y1-8+wy)/256;
                    for ( wn=0; wn<2; wn++) DAT = plus[wy*2+wn];
                }
                Xlast=X1-8;
                Ylast=Y1-8;
            }
            for(z=0;z<2000;z++) delay(200);
        }

        jz0++;
        jz1++;

        IE=0x81;
}
//touch screen


word_display(unsigned int stx,sty,unsigned char width,height,fcolor,bcolor,zkwz,zkk,wk)
//                          起点位置，              字符宽高，      前景背景色，字库位置
    ,字库块
{
    unsigned char wn;
    unsigned int yhbl,wy;       //优化变量
    yhbl=zkwz*height*width;
```

```
    //8 点写汉字
    CMD = 0x02;DAT=fcolor;
    CMD = 0x04;DAT=bcolor;
    CMD = 0x01;
    for (wy=sty; wy<sty+height; wy++)     //优化
    {
        X_ADDR = stx;
        X_ADDR = stx/256;
        Y_ADDR = wy;
        if( ((wk==1)|(wk==2))==0 ) Y_ADDR = wy/256;
        for ( wn=0; wn<width; wn++)
        {
            if(zkk==0)    DAT = hzk[yhbl+(wy-sty)*width+wn];
            if(zkk==1)    DAT = szk[yhbl+(wy-sty)*width+wn];
            if(zkk==2)    DAT = plus[yhbl+(wy-sty)*width+wn];
        }
    }
}

area_display(unsigned int ax,ay,unsigned int aw,ah,acolor,ak)
//                  单点刷块区域起点坐标，块宽高，块颜色
{
    unsigned int wn,wy;
    unsigned char axx;
    axx=ax/256;
    //单点写模式下刷某区域
    CMD = 0x00;
    for (wy=ay; wy<ah+ay; wy++)
    {
        X_ADDR = ax;
        X_ADDR = axx;
        Y_ADDR = wy;
        if( ((ak==1)|(ak==2))==0 ) Y_ADDR = wy/256;
        for ( wn=0; wn<aw; wn++) DAT = acolor;
    }
}

edot_display(unsigned int estx,esty,ex,ey,unsigned char ecolor,ek)
//                  块区域起点坐标，屏类型参数传递，前景色背景色，跳线参数传递
{
    //8 点写模式下黑色清屏 DAT=0;
    unsigned int x,y;
    unsigned char exx,estxx;       //优化变量
```

```
    exx=ex/8;estxx=estx/256;          //优化
    CMD = 0x02;DAT=0xff;//前景色(白色)
    CMD = 0x04;DAT=0x00;//背景色(黑色)
    CMD = 0x01;
    for( y=esty;y<ey+esty;y++ )//用前景色清屏     //优化
    {
        X_ADDR = estx;
        X_ADDR = estxx;                    //优化
        Y_ADDR = y;
        if( ((ek==1)|(ek==2))==0 ) Y_ADDR = y/256;
        for( x=0;x<exx;x++ ) DAT=ecolor;
    }
}


a8area_display(unsigned int ax,ay,unsigned int aw,ah,acolor,ak)
//              单点刷块区域起点坐标，块宽高，块颜色
{
    unsigned int wn,wy;
    unsigned char axx;
    axx=ax/256;
    //单点写模式下刷某区域
    CMD = 0x00;
    for (wy=ay; wy<ah+ay; wy++)
    {
        Y_ADDR = axx;
        Y_ADDR = ax;
        CMD = wy/256;
        if( ((ak==1)|(ak==2))==0 ) CMD = wy;
        for ( wn=0; wn<aw; wn++) X_ADDR = acolor;
    }
}


main()
{
    unsigned char j,m,n;
    unsigned int x,y,x2,y2,y1;
    unsigned char key,bright;
    unsigned int bkx,bky,byck;                  //刷蓝色边框参数和百叶窗块宽
    unsigned char skk,skg,skx,sky,sjj,red,green,blue; //刷色块的参数，为色块宽和色块高，红色块起点，数字间距
    unsigned char ldx,ldy,bycg,Tjj,bycjj;   //刷触摸亮度界面的参数，为红色块位置，百叶窗
```

块高，T 字区间距，百叶窗间距

AUXR = 0x02;//////////////////////////////////注意增加该语句。

```
key = 1;//0:240320,1:320240,2:480234,3:480272,4:640480,5:800480,6:800600,7:272480

key2=key;
switch(key)
{
    case 0:x2=240;y2=320;
    skk=15;skg=50;skx=95;sky=50;sjj=35;          //色块宽*8 等于 x2 的一半，也即色块条是屏宽的一半
    ldx=30;ldy=35;byck=80;bycg=120;Tjj=10,bycjj=20;
    break;
    case 1:x2=320;y2=240;
    skk=20;skg=30;skx=115;sky=50;sjj=30;
    ldx=35;ldy=30;byck=120;bycg=80;Tjj=20,bycjj=10;
    break;
    case 2:x2=480;y2=234;
    skk=35;skg=30;skx=135;sky=50;sjj=30;
    ldx=65;ldy=30;byck=160;bycg=80;Tjj=20,bycjj=30;
    break;
    case 3:x2=480;y2=272;
    skk=35;skg=35;skx=135;sky=50;sjj=35;
    ldx=65;ldy=35;byck=160;bycg=90;Tjj=20,bycjj=30;
    break;
    case 4:x2=640;y2=480;
    skk=50;skg=90;skx=155;sky=50;sjj=50;
    ldx=70;ldy=50;byck=240;bycg=170;Tjj=40,bycjj=20;
    break;
    case 5:x2=800;y2=480;
    skk=60;skg=90;skx=195;sky=50;sjj=50;
    ldx=65;ldy=50;byck=320;bycg=170;Tjj=40,bycjj=30;
    break;
    case 6:x2=800;y2=600;
    skk=60;skg=110;skx=195;sky=60;sjj=50;
    ldx=65;ldy=70;byck=320;bycg=200;Tjj=50,bycjj=30;
    break;
    case 7:x2=272;y2=480;
    skk=20;skg=90;skx=90;sky=50;sjj=50;
    ldx=35;ldy=65;byck=90;bycg=160;Tjj=30,bycjj=20;
    break;
}
```

```
xxx=x2-40;yyy=y2-40;key1=key;



//如果跳线为 8 则进入纯色测屏带点
while(key==8)
{
    //用黑色清屏
    CMD = 0x04;DAT = 0;//黑色
    CMD = 0x08;//启动填充操作
    for(y=0;y<200*200;y++) delay(10);//延时 16.6 毫秒
    CMD = 0x00;//退出填充操作
    for(y=0;y<80;y++) {x=1;while(x!=0)x++;}

    //用红色清屏
    CMD = 0x04;DAT = 0xe0;//红色
    CMD = 0x08;//启动填充操作
    for(y=0;y<200*200;y++) delay(10);//延时 16.6 毫秒
    CMD = 0x00;//退出填充操作
    for(y=0;y<80;y++) {x=1;while(x!=0)x++;}

    //用绿色清屏
    CMD = 0x04;DAT = 0x1c;//绿色
    CMD = 0x08;//启动填充操作
    for(y=0;y<200*200;y++) delay(10);//延时 16.6 毫秒
    CMD = 0x00;//退出填充操作
    for(y=0;y<80;y++) {x=1;while(x!=0)x++;}

    //用蓝色清屏
    CMD = 0x04;DAT = 0x03;//蓝色
    CMD = 0x08;//启动填充操作
    for(y=0;y<200*200;y++) delay(10);//延时 16.6 毫秒
    CMD = 0x00;//退出填充操作
    for(y=0;y<80;y++) {x=1;while(x!=0)x++;}

    //用白色清屏
    CMD = 0x04;DAT = 0xff;//白色
    CMD = 0x08;//启动填充操作
    for(y=0;y<200*200;y++) delay(10);//延时 16.6 毫秒
    CMD = 0x00;//退出填充操作
    for(y=0;y<80;y++) {x=1;while(x!=0)x++;}
}
//如果跳线为 8 则进入纯色测屏带点
```

```
//TFT8060-8A
if(key==9)
{
    //单点清屏
    CMD = 0x00;
    for (y=0;y<600;y++)
    {
        Y_ADDR = 0;
        Y_ADDR = 0;
        CMD = y/256;
        CMD = y%256;
        for (x=0;x<800;x++)    X_ADDR = 0x13;
    }
    ////////////////////////加红色单像素边框//////////////////////////
    //横向边框
    CMD = 0x00;
    for (y=0;y<2;y++)
    {
        y1=y*599;
        Y_ADDR = 0;
        Y_ADDR = 0;
        CMD = y1/256;
        CMD = y1%256;
        for (x=0;x<800;x++)    X_ADDR = 0xe0;
    }
    //横向边框
    //纵向边框
    CMD = 0x00;
    for (y=0;y<600;y++)
    {
        Y_ADDR = 0;
        Y_ADDR = 0;
        CMD = y/256;
        CMD = y%256;
        X_ADDR = 0xe0;
    }
    CMD = 0x00;
    for (y=0;y<600;y++)
    {
        Y_ADDR = 799;
        Y_ADDR = 799;
```

```
        CMD = y/256;
        CMD = y%256;
        X_ADDR = 0xe0;
}
//纵向边框
//////////////////////////加红色单像素边框//////////////////////////
//刷出色阶
for( red=0;red<8;red++ )
        a8area_display( 160+60*red,105,60,130,(7-red)*32,6 );          //刷出红色灰阶

for( green=0;green<8;green++ )
        a8area_display(  160+60*green,235,60,130,(7-green)*4,6  );     //刷出绿色灰阶


for( blue=0;blue<4;blue++ )
        a8area_display( 280+60*blue,365,60,130,3-blue,6 );             //刷出蓝色灰阶
//刷出色阶
//单点写模式下写单色图片
CMD = 0x00;
for (y=0; y<80; y++)
{
        Y_ADDR = 0;
        Y_ADDR = 1;
        CMD = (y+1)/256;
        CMD = y+1;
        for ( n=0; n<10; n++)
        {
            j = picture[y*10+n];
            for ( m=0;m<8;m++ )
            {
                if( j>127 ) X_ADDR = 0x00;
                else X_ADDR = 0xff;
                j=(j<<1);
            }
        }
}
//单点写模式下写单色图片
CMD = 0x00;
for (y=0; y<80; y++)
{
        Y_ADDR = 719/256;
        Y_ADDR = 719;
        CMD = (y+519)/256;
```

```
            CMD = y+519;
            for ( n=0; n<10; n++)
            {
                j = picture[y*10+n];
                for ( m=0;m<8;m++ )
                {
                    if( j>127 ) X_ADDR = 0xff;
                    else X_ADDR = 0x1c;
                    j=(j<<1);
                }
            }
        }
        while(1);
}
//TFT8060-8A


//TFT6448-BGEN
if(key==10)
{

 //用红色清屏
 //定义 A 点坐标（0，0）
 CMD = 0x00;   //bit[3:2]是 A 点的 X[8], bit[0]是 A 点的 Y[8],
 X_ADDR = 0;
 Y_ADDR = 0;
 //定义 B 点坐标（639，479）及填充颜色
 CMD = 0x20;
 X_ADDR = 0x7f;   //639=0x27f
 Y_ADDR = 0xdf;   //479=0x1df
 DAT      = 0xe0;
 //启动填充操作及 B 点高位操作
 CMD = 0x39;   // bit[3:2]是 B 点的 X[8], bit[0]是 B 点的 Y[8],
 //延时 16.6 毫秒
 for(y=0;y<250*150;y++);
 //退出填充操作
 CMD = 0x00;

 X_ADDR = 50;
 Y_ADDR = 50;
 //定义 B 点坐标（589，429）及填充颜色
 CMD = 0x20;
 X_ADDR = 0x4d;   //589=0x24D
```

```
Y_ADDR = 0xad;   //429=0x1AD
DAT      = 0x1c;
//启动填充操作及 B 点高位操作
CMD = 0x39;   // bit[3:2]是 B 点的 X[8], bit[0]是 B 点的 Y[8],
//延时 16.6 毫秒
for(y=0;y<250*150;y++);
//退出填充操作
CMD = 0x00;

X_ADDR = 100;
Y_ADDR = 100;
//定义 B 点坐标（589，429）及填充颜色
CMD = 0x20;
X_ADDR = 0x1b;   //539=0x21b
Y_ADDR = 0x7b;   //379=0x17b
DAT      = 0x03;
//启动填充操作及 B 点高位操作
CMD = 0x39;   // bit[3:2]是 B 点的 X[8], bit[0]是 B 点的 Y[8],
//延时 16.6 毫秒
for(y=0;y<250*150;y++);
//退出填充操作
CMD = 0x00;

X_ADDR = 150;
Y_ADDR = 150;
//定义 B 点坐标（589，429）及填充颜色
CMD = 0x20;
X_ADDR = 0xe9;   //489=0x1e9
Y_ADDR = 0x49;   //329=0x149
DAT      = 0x1f;
//启动填充操作及 B 点高位操作
CMD = 0x35;   // bit[3:2]是 B 点的 X[8], bit[0]是 B 点的 Y[8],
//延时 16.6 毫秒
for(y=0;y<250*150;y++);
//退出填充操作
CMD = 0x00;

X_ADDR = 200;
Y_ADDR = 200;
//定义 B 点坐标（589，429）及填充颜色
CMD = 0x20;
X_ADDR = 0xb7;   //439=0x1b7
Y_ADDR = 0x17;   //279=0x117
```

```
DAT       = 0xfc;
//启动填充操作及 B 点高位操作
CMD = 0x35;    // bit[3:2]是 B 点的 X[8], bit[0]是 B 点的 Y[8],
//延时 16.6 毫秒
for(y=0;y<250*150;y++);
//退出填充操作
CMD = 0x00;

//单点写模式下写单色图片
CMD = 0x00;
for (y=0; y<80; y++)
    {
     X_ADDR = 1;
     Y_ADDR = y+1;
     for ( n=0; n<10; n++)
          {j = picture[y*10+n];
            for ( m=0;m<8;m++ )
                 {
                   if( j>127 ) DAT = 0xff; else DAT = 0x00;
                   j=(j<<1);
                 }
          }
      }
//单点写模式下写单色图片
CMD = 0x09;
for (y=0; y<80; y++)
    {
     X_ADDR = 559;
     Y_ADDR = y+399;
     for ( n=0; n<10; n++)
          {j = picture[y*10+n];
           for ( m=0;m<8;m++ )
                 {
                   if( j>127 ) DAT = 0x1f; else DAT = 0x03;
                   j=(j<<1);
                 }
          }
      }
while(1);
}


//如果跳线为 15 则进入纯红色测电流
```

```
if(key==15)
{
    CMD = 0x04;DAT = 0xe0;//背景色
    CMD = 0x08;//启动填充操作
    for(y=0;y<200*200;y++) delay(10);//延时 16.6 毫秒
    CMD = 0x00;//退出填充操作
    while(1);
}
//如果跳线为 15 则进入纯红色测电流


//touch screen
TMOD=0x11; //  记数器 0  计数器 1  都以  16  位  记数
TCON=0x00;
IE=0x81; //1000 0001 EA=1 中断允许,
IP=0x01;
//touch screen
bright = 7;
CMD = 0x06;
DAT = bright;                          //设置亮度


//////////////////////红色清屏//////////////////////////////////
CMD = 0x04;DAT = 0xe0;//背景色
CMD = 0x08;//启动填充操作
for(y=0;y<200*200;y++) delay(10);//延时 16.6 毫秒
CMD = 0x00;//退出填充操作
for(y=0;y<50;y++){x=1;while(x!=0)x++;}
//////////////////////红色清屏//////////////////////////////////

//////////////////////清屏//////////////////////////////////////
edot_display( 0,0,x2,y2,0,key );                //8 点黑色清屏

for(y=0;y<30;y++){x=1;while(x!=0)x++;}
//////////////////////清屏//////////////////////////////////////

//////////////////////加蓝色单像素边框//////////////////////////
for(bkx=0;bkx<x2;bkx++)
{
    area_display( bkx,0,1,1,0x03,key );
    area_display( bkx,y2-1,1,1,0x03,key );
}
```

```
    if(key==1)
    {
        for(bkx=0;bkx<x2;bkx++)
            area_display( bkx,233,1,1,0x03,key );
    }

    for(bky=0;bky<y2;bky++)
    {
        area_display( 0,bky,1,1,0x03,key );
        area_display( x2-1,bky,1,1,0x03,key );
    }
    ////////////////////////加蓝色单像素边框//////////////////////////

    ////////////////////色彩测试////////////////////////////////////
    word_display( 10,10,4,29,0xff,0,0,0,key );          //显示“色”
    word_display( 42,10,4,29,0xe0,0,1,0,key );          //显示“彩”
    word_display( 74,10,4,29,0x1c,0,2,0,key );          //显示“测”
    word_display( 106,10,4,29,0x03,0,3,0,key );         //显示“试”
    word_display( 138,10,4,29,0x7f,0,11,0,key );        //显示“：”

    //刷出红色灰阶
    word_display( skx-70,sky+skg/2-15,4,29,0xe0,0,4,0,key );
//显示“红”
    word_display( skx-38,sky+skg/2-15,4,29,0xe0,0,0,0,key );
//显示“色”
    for( red=0;red<8;red++ )
    {
        area_display(                skx+(red*skk),sky,skk,skg,(7-red)*32,key                );
//刷出红色灰阶
        word_display( skx+skk/2-8+(red*skk),sky+skg,2,29,0xe0,0,7-red,1,key );
//刷出对应数字
    }
    //刷出红色灰阶

    //刷出绿色灰阶
    word_display( skx-70,sky+skg/2-15+(skg+sjj),4,29,0x1c,0,5,0,key );
//显示“绿”
    word_display( skx-38,sky+skg/2-15+(skg+sjj),4,29,0x1c,0,0,0,key );
//显示“色”
    for( green=0;green<8;green++ )
    {
        area_display(          skx+(green*skk),sky+(skg+sjj),skk,skg,(7-green)*4,key          );
//刷出绿色灰阶
```

```
        word_display( skx+skk/2-8+(green*skk),sky+skg+(skg+sjj),2,29,0x1c,0,7-green,1,key );
//刷出对应数字
    }
    //刷出绿色灰阶

    //刷出蓝色灰阶
    word_display( skx-70,sky+skg/2-15+(skg+sjj)*2,4,29,0x03,0,6,0,key );
//显示“蓝”
    word_display( skx-38,sky+skg/2-15+(skg+sjj)*2,4,29,0x03,0,0,0,key );
//显示“色”
    for( blue=0;blue<4;blue++ )
    {
        area_display(          skx+(blue*skk),sky+(skg+sjj)*2,skk,skg,3-blue,key          );
//刷出蓝色灰阶
        word_display( skx+skk/2-8+(blue*skk),sky+skg+(skg+sjj)*2,2,29,0x03,0,3-blue,1,key );
//刷出对应数字
    }
    //刷出蓝色灰阶
    for(y=0;y<90;y++){x=1;while(x!=0)x++;}
    ////////////////////////色彩测试/////////////////////////////////////

    ////////////////////////亮度调节和触摸测试/////////////////////////////
    edot_display(                    0,0,x2,y2,0,key                    );
//8 点黑色清屏



    for(y=0;y<3;y++){x=1;while(x!=0)x++;}

    word_display( 12,12,2,16,0xe0,0,0,2,key );                    //
显示第一个触摸校准十字

    area_display( ldx,ldy,byck*2+bycjj,bycg,0xe0,key );                    //
单点刷出红色块区域
    edot_display(  ldx,ldy+bycg+Tjj,byck,bycg,0x55,key  );                    //8
点黑色清屏
    edot_display( ldx+byck+bycjj,ldy+bycg+Tjj,byck,bycg,0x0f,key );                    //8 点
黑色清屏

    word_display( x2/2-80,ldy+bycg/2-15,4,29,0x7f,0xe0,7,0,key );                    // 显 示
“第”
    word_display( x2/2-80+32*2,ldy+bycg/2-15,4,29,0x7f,0xe0,8,0,key );                    //
显示“级”
```

```
    word_display( x2/2-80+32*3,ldy+bycg/2-15,4,29,0x7f,0xe0,9,0,key );                    //
显示“亮”
    word_display( x2/2-80+32*4,ldy+bycg/2-15,4,29,0x7f,0xe0,10,0,key );                   //
显示“度?

    jz0=0;        //取第一个校准点坐标标志位
    while(!jz0)
    {
        for(y=0;y<30;y++){x=1;while(x!=0)x++;}
        for(bright=7;bright!=255;bright--)
        {
            CMD = 0x06;
            DAT = bright;
            word_display( x2/2-80+32+8,ldy+bycg/2-15,2,29,0x7f,0xe0,bright,1,key );    //
显示“N”
            for(y=0;y<30;y++){x=1;while(x!=0)x++;}
            if(jz0) break;
        }
        CMD                                =                                0x20;
//关断显示，poweroff
    }

    bright = 7;
    CMD = 0x06;
    DAT = bright;                          //设置亮度

    //用黑色清屏消除第一个十字
    edot_display(                          0,0,x2,y2,0,key                          );
//8 点黑色清屏
    //用黑色清屏消除第一个十字

    jz1=0;        //取第二个校准点坐标标志位
    word_display( x2-28,y2-28,2,16,0xe0,0,0,2,key );                                      //
显示第二个触摸校准十字

    while(!jz1);

    for( x=0;x<5000;x++ );
    jz2=1;        //和以上延时配套，防止校准最后取点左边的误差,如果这个标志未置位则
无法显示触摸点

    area_display( x2-28,y2-28,16,16,0,key );
//消除第二个十字
```

```
//////////////////////亮度调节和触摸测试//////////////////////////

    while(1);

}
```

文章来源：武汉谷鑫科技有限公司　www.mcutech.cn