

JLX12864G-200-PN 使用说明书

目 录

序号	内 容 标 题	页码
1	概述	2
2	字符型模块的特点	2
3	外形及接口引脚功能	3~4
4	基本原理	4~5
5	技术参数	5
6	时序特性	6~7
7	指令功能及硬件接口与编程案例	8~末页

1. 概述

晶联讯电子专注于液晶屏及液晶模块的研发、制造。所生产 JLX12864G-200 型液晶模块由于使用方便、显示清晰，广泛应用于各种人机交流面板。

JLX12864G-200 可以显示 128 列*64 行点阵单色图片，或显示 8 个/行*4 行 16*16 点阵的汉字，或显示 16 个/行*8 行 8*8 点阵的英文、数字、符号。

2. JLX12864G-200 图像型点阵液晶模块的特性

2.1 结构牢：背光带有挡墙，焊接式 FPC。

2.2 IC 采用矽创公司 ST7588T, 功能强大，稳定性好

2.3 显示内容：

- 128*64 点阵单色图片；

- 可選用 16*16 点阵或其他点阵的图片来自编汉字，按照 16*16 点阵汉字来计算可显示 8 字*4 行。

- 按照 12*12 点阵汉字来计算可显示 10 字/行*4 行。

- 可显示 16 个*4 行 8*16 点阵的英文、数字、符号。

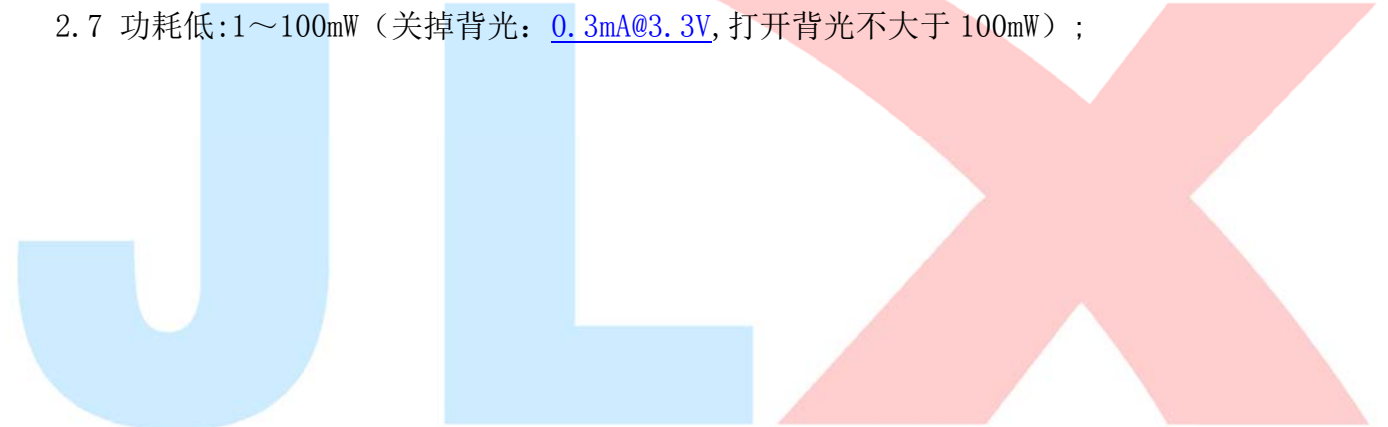
- 可显示 21 个*8 行 5*8 点阵的英文、数字、符号。

2.4 指令功能强：可软件调对比度、正显/反显转换、行列扫描方向可改（可旋转 180 度使用）。

2.5 接口简单方便：I²C 接口。

2.6 工作温度宽：-20℃~+70℃；

2.7 功耗低：1~100mW（关掉背光：[0.3mA@3.3V](#), 打开背光不大于 100mW）；



3. 外形尺寸及接口引脚功能

3.1 外形尺寸图

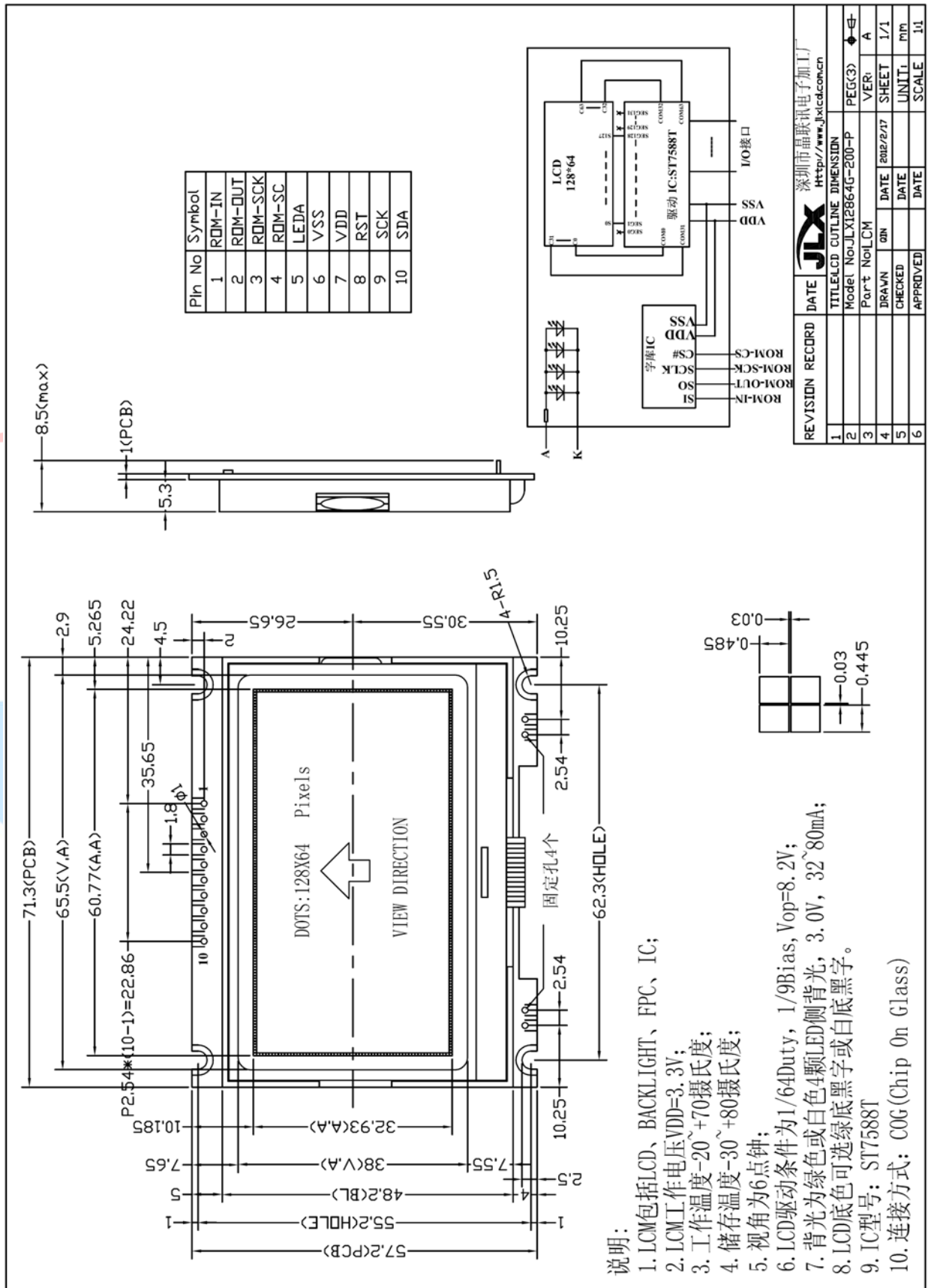


图 1. 外形尺寸

说明:

1. LCM包括LCD、BACKLIGHT、FPC、IC;
2. LCM工作电压VDD=3.3V;
3. 工作温度-20~+70摄氏度;
4. 储存温度-30~+80摄氏度;
5. 视角为6点钟;
6. LCD驱动条件为1/64Duty, 1/9Bias, Vop=8.2V;
7. 背光为绿色或白色4颗LED侧背光, 3.0V, 32~80mA;
8. LCD底色可选绿底黑字或白底黑字。
9. IC型号: ST7588T
10. 连接方式: COG (Chip On Glass)

3.2 模块的接口引脚功能

3.2.1 I²C 接口引脚功能

引线号	符号	名称	功能
1	ROM_IN (NC)	即字库 IC 接口 (SI)	串行数据输入
2	ROM_OUT (NC)	即字库 IC 接口 (SO)	串行数据输出
3	ROM_SCK (NC)	即字库 IC 接口 (sck)	串行时钟输入
4	ROM_CS (NC)	字库 IC 接口 (CS#)	片选输入
1. 当选择带字库的产品, 请参阅: (1) 字库 IC: JLX-GB2312 说明书 (2) JLX12864G-200-PC 的中文字库编程说明书 2. 当不用字库时为空			
5	LEDA	背光电源	背光电源正极, 同 VDD 电压 (5V 或 3.3V)
6	VSS	接地	0V
7	VDD	电路电源	5V, 或 3.3V 可选
8	RST	复位	低电平复位, 复位完成后, 回到高电平, 液晶模块开始工作
9	SCK (D6)	I/O	串行时钟
10	SDA (D7)	I/O	串行数据

表 1: 模块 I²C 接口引脚功能

4. 基本原理

4.1 液晶屏 (LCD)

在 LCD 上排列着 128×64 点阵, 128 个列信号与驱动 IC 相连, 64 个行信号也与驱动 IC 相连, IC 邦定在 LCD 玻璃上 (这种加工工艺叫 COG)。

4.2 工作电路图:

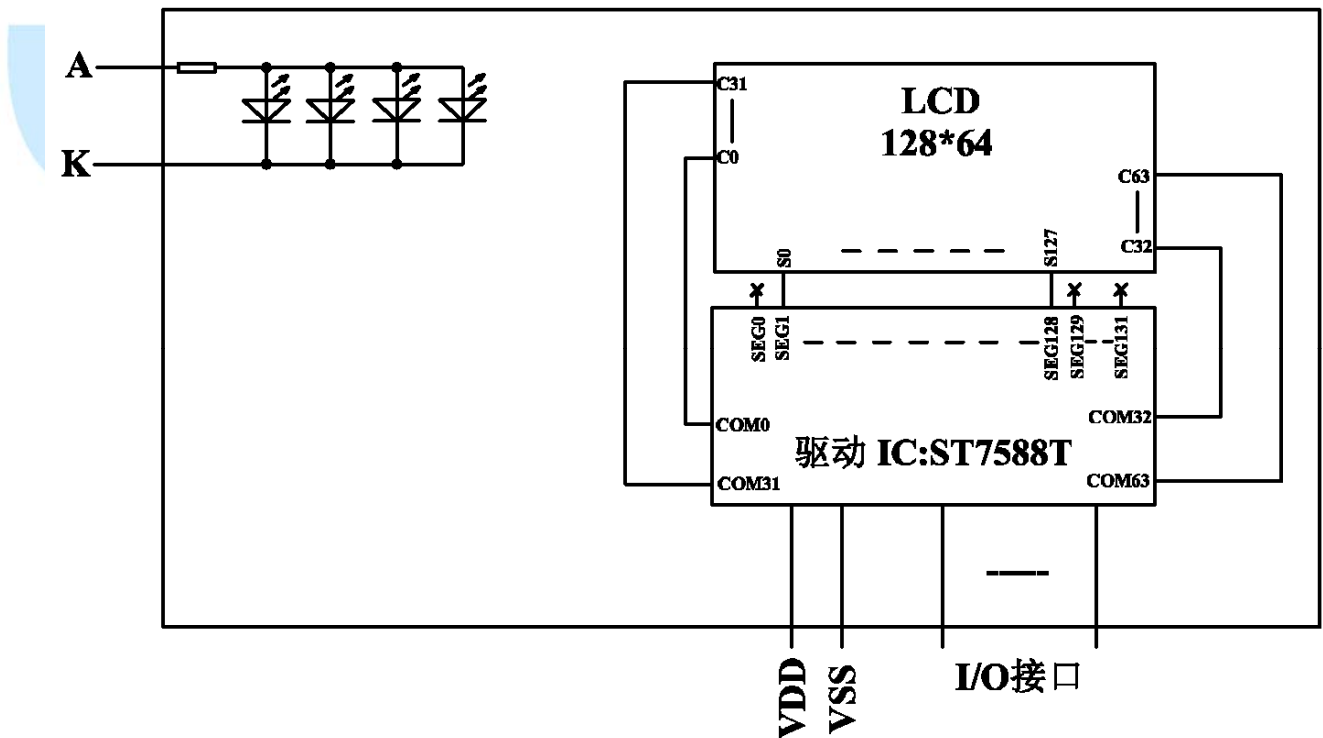


图 2: JLX12864G-200 图像点阵型液晶模块的电路框图

4.3 背光参数

该型号液晶模块带 LED 背光源。它的性能参数如下:

工作温度: 20~+70° C;

存储温度: -30~+80° C;

背光板可选择绿色和白色。

正常工作电流为: 32~80mA;

工作电压: 3.0V;

5. 技术参数

5.1 最大极限参数 (超过极限参数则会损坏液晶模块)

名称	符号	标准值			单位
		最小	典型	最大	
电路电源	VDD - VSS	-0.3		7.0	V
LCD 驱动电压	V0、VOUT	-0.3		13.5	V
静电电压		-	-	100	V
工作温度		-20		+70	°C
储存温度		-30		+80	°C

表 2: 最大极限参数

5.2 直流 (DC) 参数

名称	符号	测试条件	标准值			单位
			MIN	TYPE	MAX	
工作电压	VDD		2.4	-	3.6	V
背光工作电压	VLED		2.9	3.0	3.1	V
输入高电平	V _{IHC}	-	0.8xVDD	-	VDD	V
输入低电平	V _{ILC}	-	VSS	-	0.2xVDD	V
输出高电平	V _{OHC}	I _{OH} = -0.5mA	0.8xVDD	-	VDD	V
输出低电平	V _{OHC}	I _{OL} = -0.5mA	VSS	-	0.2xVDD	V
模块工作电流	I _{DD}	VDD = 3.3V	-		0.3	mA
背光工作电流	I _{LED}	V _{LED} =3.0V	32	40	80	mA

表 3: 直流 (DC) 参数

6. 读写时序特性

6.1 从 CPU 写到 ST7588T (Writing Data from CPU to ST7588T)

- SERIAL INTERFACE (IIC Interface)

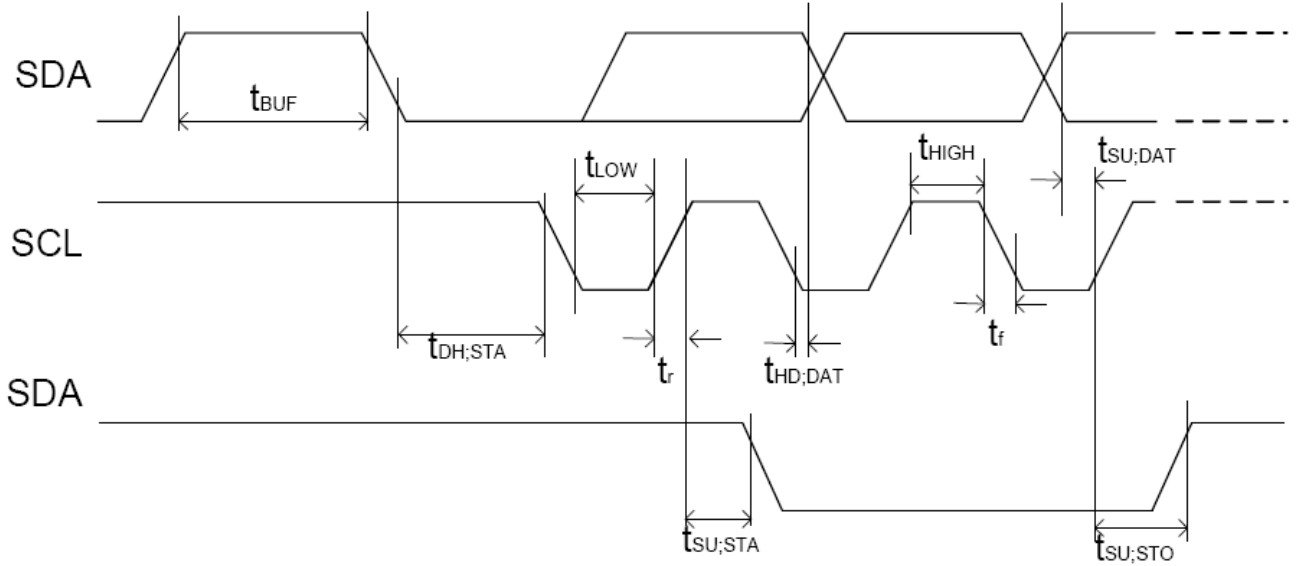


图 3. 从 CPU 写到 ST7588T (Writing Data from CPU to ST7588T)

6.2 时序要求 (AC 参数): 写数据到 ST7588 的时序要求:

表 4.

($V_{DD}=3.3V, T_a=25^{\circ}C$)

Item	Signal	Symbol	Condition	Rating		Units
				Min.	Max.	
SCL clock frequency	SCL	f_{SCLK}		DC	400	kHz
SCL clock low period	SCL	t_{LOW}		150	--	
SCL clock high period	SCL	t_{HIGH}		100	--	
Data set-up time	SDA	$t_{SU;Dat}$		90	--	
Data hold time	SDA	$t_{HD;Dat}$		40	--	
Setup time for a repeated START condition	SDA	$t_{SU;STA}$		70	--	
Start condition hold time	SDA	$t_{HD;STA}$		170	--	
Setup time for STOP condition		$t_{SU;STO}$		90	--	
BUS free time between a STOP and START condition	SCL	t_{BUF}		70	--	

6.3 电源启动后复位的时序要求 (RESET CONDITION AFTER POWER UP):

■ RESET TIMING

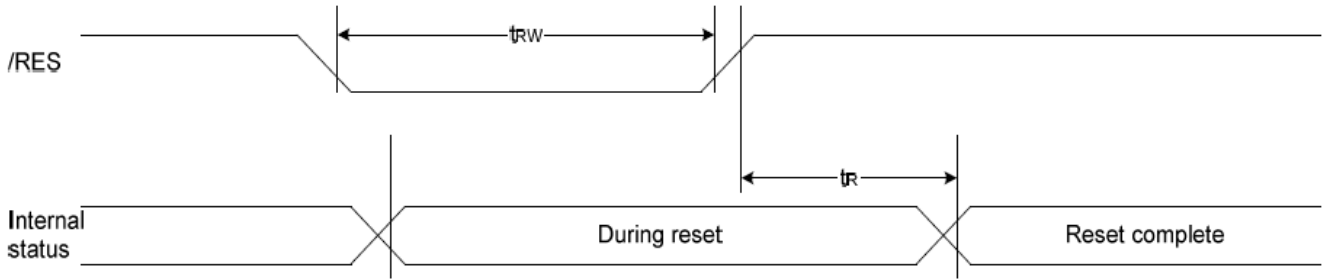
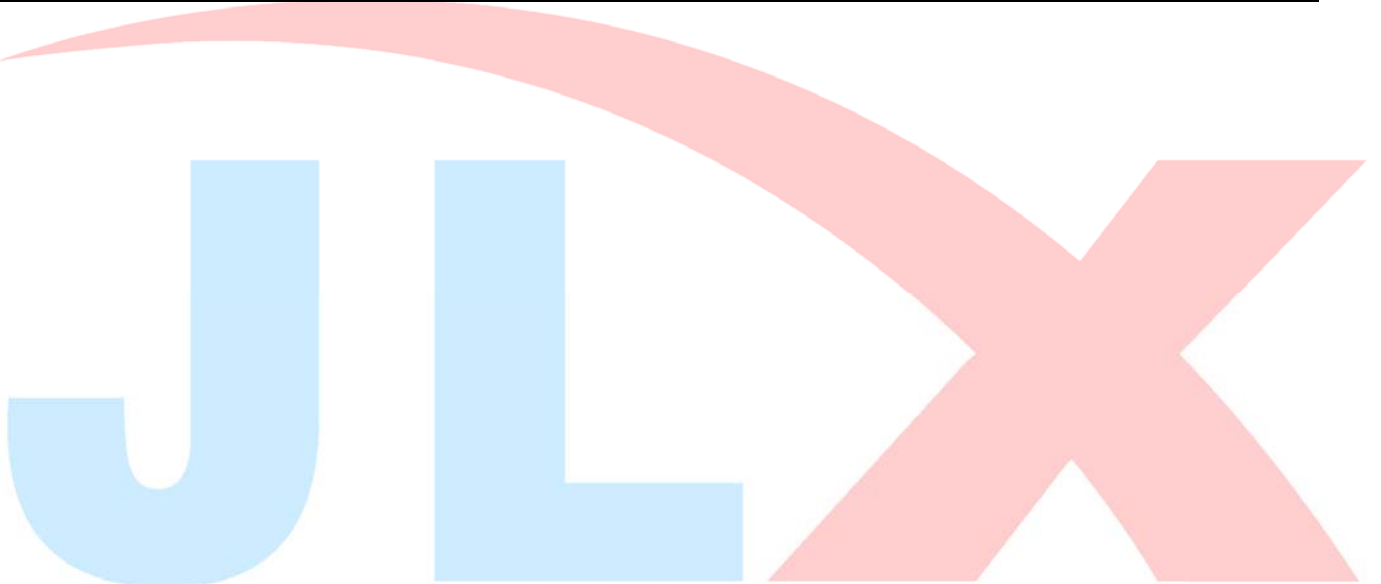


图 5: 电源启动后复位的时序

表 6: 电源启动后复位的时序要求

项目	符号	测试条件	极限值			单位
			MIN	TYPE	MAX	
复位时间	tr		--	--	400	ns
复位保持低电平的时间	trw	引脚: RES	1.0	--	--	us



7. 指令功能:

7.1 指令表

指令表

表 8.

INSTRUCTION TABLE

INSTRUCTION	A0	WR (R/W)	COMMAND BYTE								DESCRIPTION
			D7	D6	D5	D4	D3	D2	D1	D0	
H independent instruction											
Write data	1	0	D7	D6	D5	D4	D3	D2	D1	D0	Write data to RAM
Read data	1	1	D7	D6	D5	D4	D3	D2	D1	D0	Read data to RAM
Read status byte	0	1	PD	0	V	D	E	MX	MY	DO	Read status byte
Function Set	0	0	0	0	1	MX	MY	PD	H1	H0	Mirror X, mirror Y, Power down, Extended table

INSTRUCTION	A0	WR (R/W)	COMMAND BYTE								DESCRIPTION
			D7	D6	D5	D4	D3	D2	D1	D0	
H[1:0]=[0:0]											
Set V _{LCD} range	0	0	0	0	0	0	0	1	0	PRS	V _{LCD} range L/H select
END	0	0	0	0	0	0	0	1	1	0	Release read/modify/write
Read/modify/write	0	0	0	0	0	0	0	1	1	1	RAM address at R:+0, W:+1
Display control	0	0	0	0	0	0	1	D	0	E	Sets display configuration
SI3-8bit data(L)&start	0	0	0	1	0	1	DA3	DA2	DA1	DA0	Specify the number of data bytes (8 bit 3-line SPI)
SI3-8bit data(L)	0	0	0	1	1	0	DA7	DA6	DA5	DA4	Specify the number of data bytes (8 bit 3-line SPI)
SI3-8bit data (M)	0	0	0	1	1	1	0	DA10	DA9	DA8	Specify the number of data bytes (8 bit 3-line SPI)
Set Y address of RAM	0	0	0	1	0	0	Y3	Y2	Y1	Y0	Sets Y address of RAM 0 ≤ Y ≤ 9
Set Address (Low)	0	0	1	1	1	0	X3	X2	X1	X0	Set X address of RAM 0 ≤ X ≤ 131
Set Address (High)	0	0	1	1	1	1	X7	X6	X5	X4	Set X address of RAM 0 ≤ X ≤ 131
H[1:0]=[0:1]											
Display configuration	0	0	0	0	0	0	1	DO	0	V	Top/bottom row mode set data order
Bias system	0	0	0	0	0	1	0	BS ₂	BS ₁	BS ₀	Sets bias system (BSx)
Set V _{OP}	0	0	1	V _{OP6}	V _{OP5}	V _{OP4}	V _{OP3}	V _{OP2}	V _{OP1}	V _{OP0}	Write V _{OP} to register

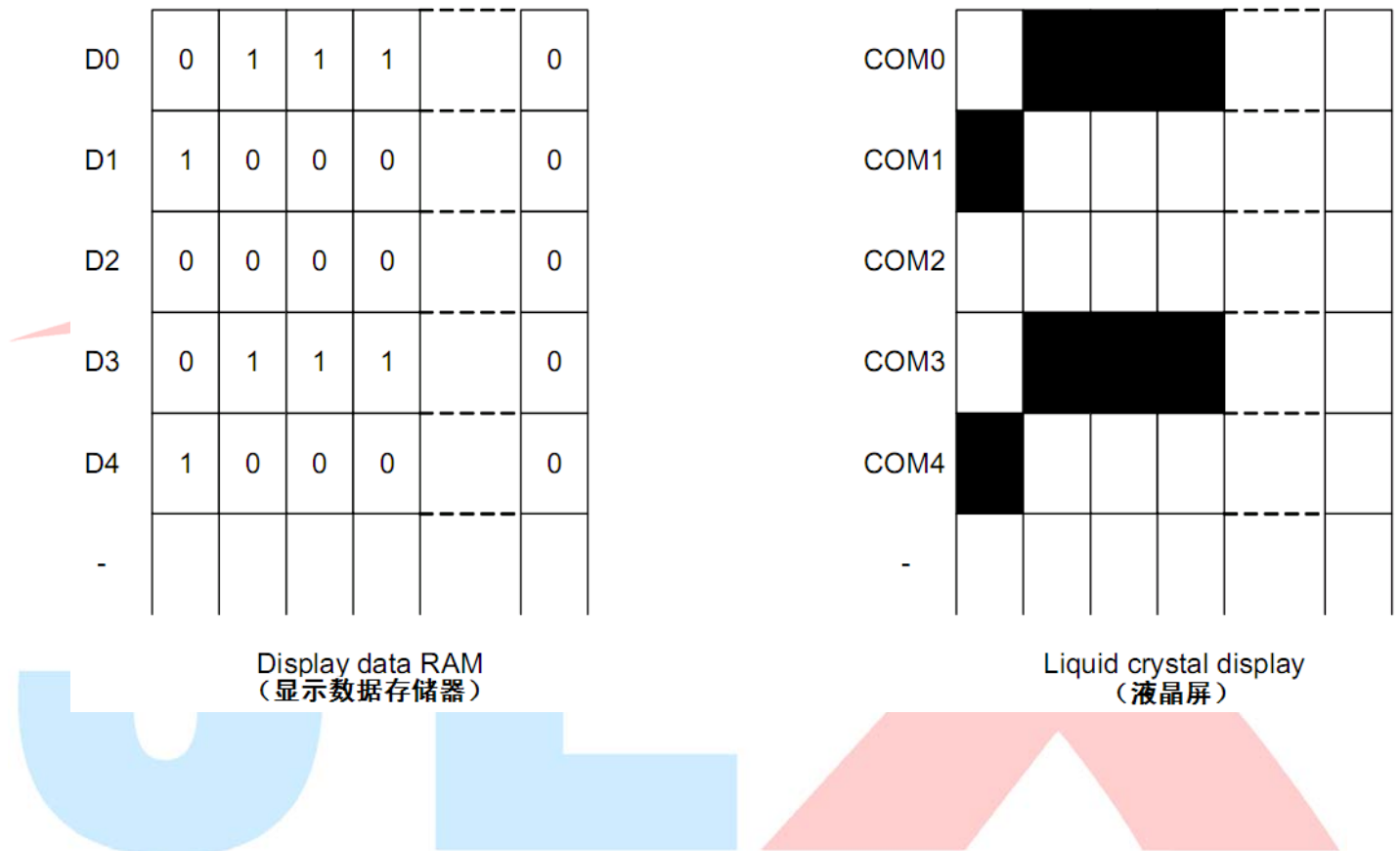
INSTRUCTION	A0	WR (R/W)	COMMAND BYTE								DESCRIPTION
			D7	D6	D5	D4	D3	D2	D1	D0	
H[1:0]=[1:0]											
Partial screen mode	0	0	0	0	0	0	0	1	0	PS	Partial screen enable
Partial Display	0	0	0	0	0	0	1	0	0	WS	Set partial screen size
Partial Display part	0	0	0	0	0	1	DP3	DP2	DP1	DP0	Set display part for partial screen mode
Set Start line	0	0	1	S6	S5	S4	S3	S2	S1	S0	Specify the initial display line to realize vertical scrolling
H[1:0]=[1:1]											
RESET	0	0	0	0	0	0	0	0	1	1	Software reset
High Power Mode	0	0	1	0	1	1	0	HP	0	0	High Power Mode SET
Frame	0	0	0	0	0	0	1	FR2	FR1	FR0	Frame rate control
N line inversion	0	0	0	1	0	NL4	NL3	NL2	NL1	NL0	Sets N line inversion

请详细参考 IC 资料”ST7588T.PDF”。

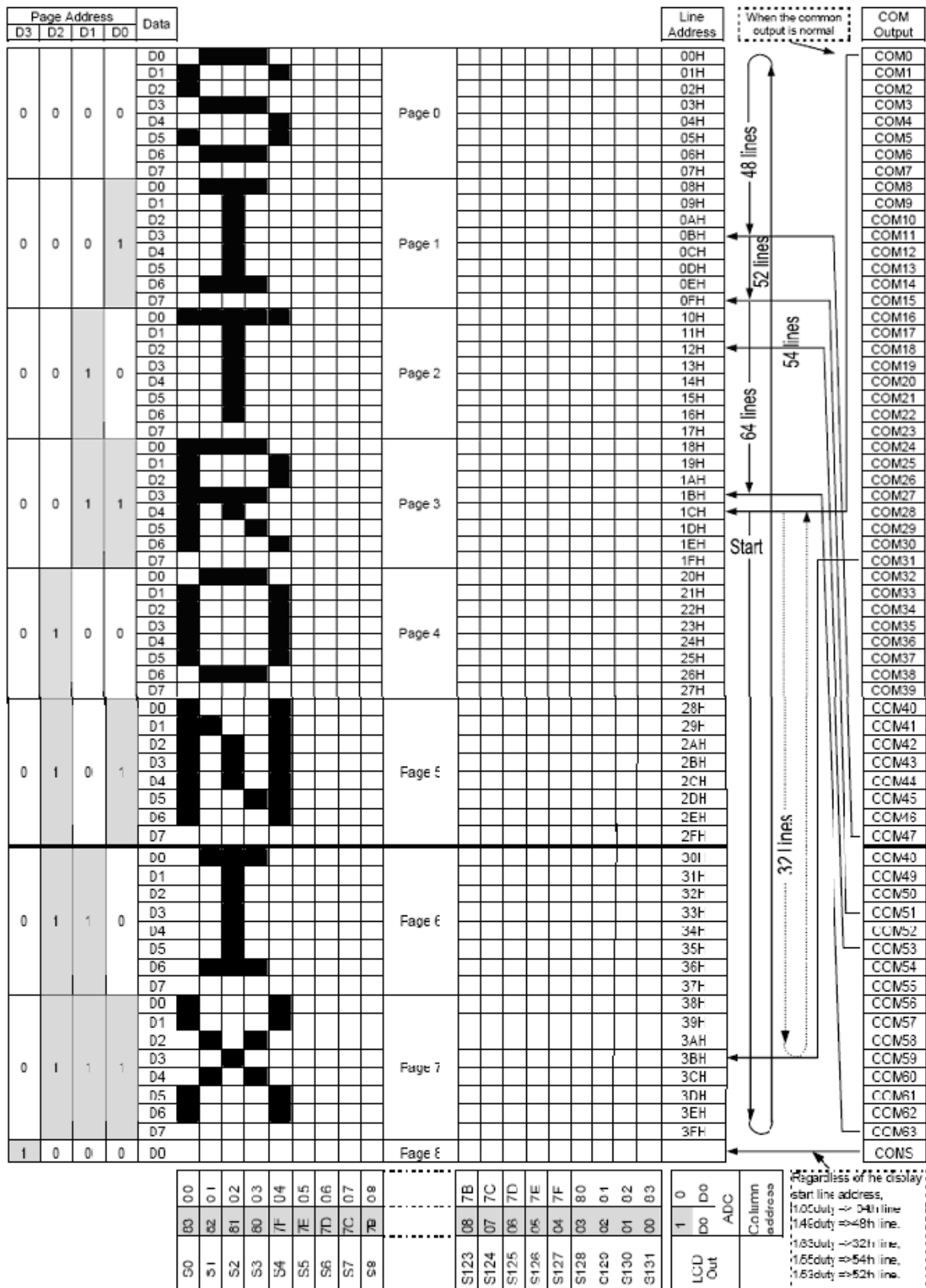
7.3 点阵与 DD RAM 地址的对应关系

请留意页的定义：PAGE, 与平时所讲的“页”并不是一个意思，在此表示 8 个行就是一个“页”，一个 128*64 点阵的屏分为 8 个“页”，从第 0“页”到第 7“页”。

DB7--DB0 的排列方向：数据是从下向上排列的。最低位 D0 是在最上面，最高位 D7 是在最下面。每一位 (bit) 数据对应一个点阵，通常“1”代表点亮该点阵，“0”代表关掉该点阵。如下图所示：



下图摘自 ST7567 IC 资料，可通过“ST7588T.PDF”获取最佳效果。



Data structure

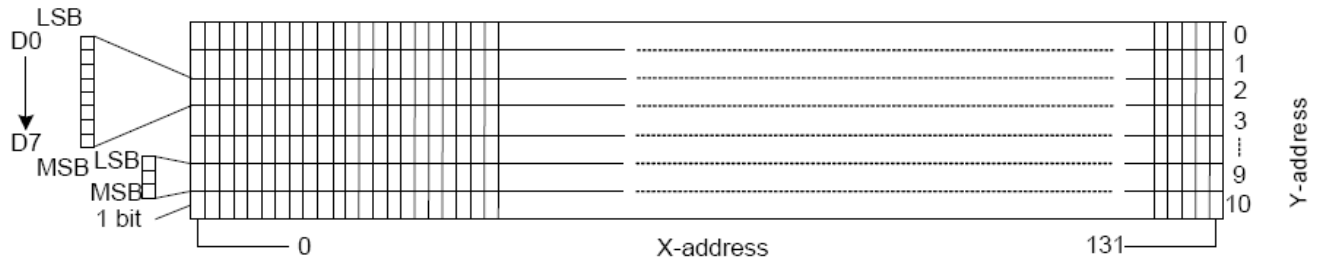


Figure 14 RAM format, addressing , if DO=0

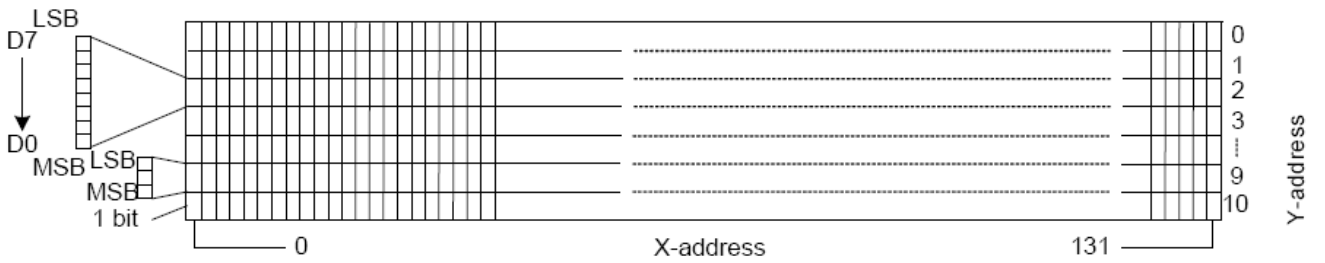
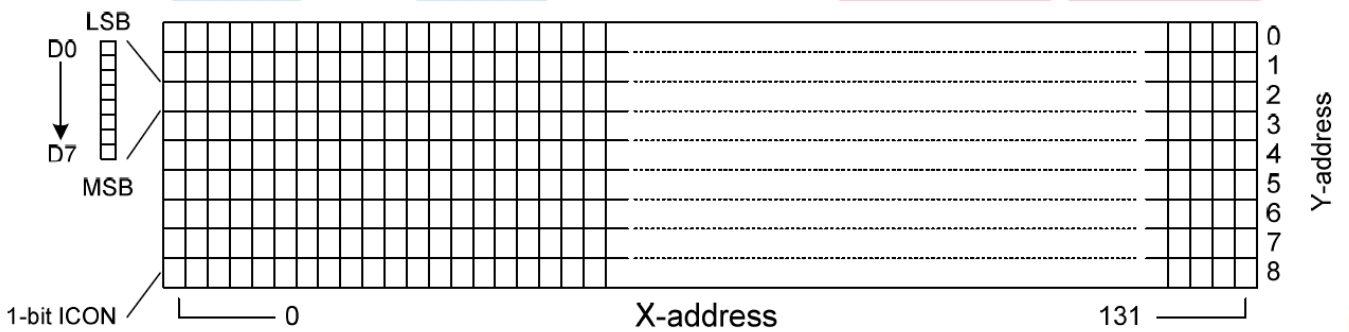


Figure 15 RAM format, addressing, if DO=1



7.4 初始化方法

用户所编的显示程序, 开始必须进行初始化, 否则模块无法正常显示, 过程请参考程序

点亮液晶模块的步骤

硬件准备:
开发板 (或专门设计的主板)、单片机、电源、连接线、仿真器或程序下载器 (又名烧录器)

正确地接线
根据说明书正确地与开发板连接, 连接的线包括: 液晶模块电源线、背光电源线、I/O 端口 (接口)
I/O 端口包括: 并口时: CS、RESET、RW、E、RS、D0--D7, 串口时: CS、SCLK、SDA、RESET、RS

编写软件
背光给合适的直流电可以点亮, 但液晶屏里面没有程序, 只给电不能让液晶屏显示 (我们通常说“点亮”), 程序须另外编写, 并烧录 (下载) 到单片机里液晶模块才能工作。

7.5 程序举例:

7.5.1 并行接口

液晶模块与 MPU (以 8051 系列单片机为例) 接口图如下:

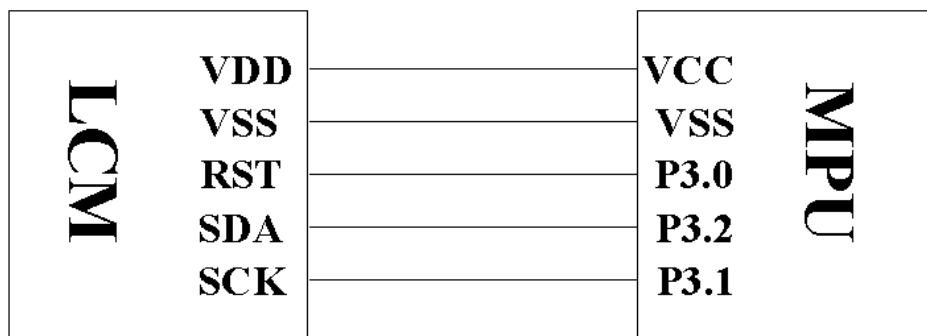


图 8. I²C 接口

```
//Test program for JLX12864G-200, I2C 接口
//驱动 IC 是:ST7588T
//晶联讯电子: 网址 http://www.jlxlcd.cn:
```

```
#include <reg51.h>
sbit reset=P3^0;
sbit scl=P3^1;
sbit sda=P3^2;
```

```
#define uchar unsigned char
#define uint unsigned int
#define ulong unsigned long
```

```
char code bmp_12864_1[];
char code bmp_12864_2[];
char code bmp_12864_3[];
char code bmp_12864_4[];
char code bmp_12864_5[];
char code bmp_12864_6[];
```

```
char code yun1[];
char code xing1[];
char code zhong1[];
char code gel[];
char code qul[];
char code ming1[];
char code chuan1[];
char code qil[];
```

```
char code yan1[];
char code chang1[];
char code zhel[];
char code wang1[];
char code feil[];
char code mao_hao[];
```

```
//=====delay time=====
```

```
void delay(int i)
{
    int j,k;
    for(j=0;j<i;j++)
    for(k=0;k<990;k++);
}
```

```
//-----wait a switch, jump out if P2.0 get a signal"0"-----
```

```
void waitkey()
{
    repeat:
        if (P2&0x01) goto repeat;
        else;
        delay(60);
}
```

```
void transfer(int data1)
{
    int i;
    for(i=0;i<8;i++)
    {
        scl=0;
        if(data1&0x80) sda=1;
        else sda=0;
        scl=1;
        scl=0;
        data1=data1<<1;
    }
    sda=0;
    scl=1;
    scl=0;
```

```

}

void start_flag()
{
    scl=1;      /*START FLAG*/
    sda=1;      /*START FLAG*/
    sda=0;      /*START FLAG*/
}

void stop_flag()
{
    scl=1;      /*STOP FLAG*/
    sda=0;      /*STOP FLAG*/
    sda=1;      /*STOP FLAG*/
}

display_graphic_128x64(uint page,uint column,uchar *dp)
{
    int i, j;

    uchar page_address;
    uchar column_address_L, column_address_H;
    page=page-1;
    page_address = 0x40+page;
    column=column+3;
    column_address_L =0xe0+(column&0x0f);
    column_address_H =0xf0+((column>>4)&0x0f);

    for(j=0; j<8; j++)
    {
        start_flag();          //开始标志
        transfer(0x7e);        //选择 SLAVE ADDRESS
        transfer(0x00);        //控制字：表示以下传输的 N 个字节是指令
        transfer(0x38);        //功能设置：MX=0, MY=0, PD=0, H1=0, H0=0
        transfer(page_address+j); //设置页地址
        transfer(column_address_L); //设置页地址低位
        transfer(column_address_H); //设置页地址高位
        stop_flag();          //停止标志
        start_flag();          //开始标志
        transfer(0x7e);        //选择 SLAVE ADDRESS
        transfer(0x40);        //控制字：表示以下传输的 N 个字节都是显示数据
        for(i=0; i<128; i++)
        {
            transfer(*dp);
            dp++;
        }
        transfer(0x00);
        stop_flag();
    }
}

display_graphic_32x32(uint page,uint column,uchar *dp)
{
    int i, j;

    uchar page_address;
    uchar column_address_L, column_address_H;
    page=page-1;
    page_address = 0x40+page;
    column=column+3;
    column_address_L =0xe0+(column&0x0f);
    column_address_H =0xf0+((column>>4)&0x0f);

    for(j=0; j<4; j++)
    {
        start_flag();          /*开始标志*/
        transfer(0x7e);        /*选择 SLAVE ADDRESS*/
        transfer(0x00);        /*控制字：表示以下传输的 N 个字节是指令*/
    }
}

```

```

transfer(0x38);          /*功能设置：MX=0,MY=0,PD=0,H1=0,H0=0,*/
transfer(page_address+j); /*设置页地址*/
transfer(column_address_L); /*设置页地址低位*/
transfer(column_address_H); /*设置页地址高位*/
stop_flag();           /*停止标志*/
start_flag();          /*开始标志*/
transfer(0x7e);         /*选择 SLAVE ADDRESS*/
transfer(0x40);         /*控制字：表示以下传输的 N 个字节都是显示数据*/
for(i=0;i<31;i++)
{
    transfer(*dp);
    dp++;
}
transfer(0x00);
stop_flag();
}
}

```

```
display_graphic_16x16(uint page,uint column,uchar *dp)
```

```

{
    int i,j;

    uchar page_address;
    uchar column_address_L,column_address_H;
    page=page-1;
    page_address = 0x40+page;
    column=column+3;
    column_address_L =0xe0+(column&0x0f);
    column_address_H =0xf0+((column>>4)&0x0f);

```

```

for(j=0;j<2;j++)
{
    start_flag();          /*开始标志*/
    transfer(0x7e);        /*选择 SLAVE ADDRESS*/
    transfer(0x00);        /*控制字：表示以下传输的 N 个字节是指令*/
    transfer(0x38);        /*功能设置：MX=0,MY=0,PD=0,H1=0,H0=0,*/
    transfer(page_address+j); /*设置页地址*/
    transfer(column_address_L); /*设置页地址低位*/
    transfer(column_address_H); /*设置页地址高位*/
    stop_flag();
    start_flag();
    transfer(0x7e);
    transfer(0x40);        /*控制字：表示以下传输的 N 个字节都是显示数据*/
    for(i=0;i<16;i++)
    {
        transfer(*dp);
        dp++;
    }
    stop_flag();
}
}

```

```
display_graphic_8x16(uint page,uint column,uchar *dp)
```

```

{
    int i,j;

    uchar page_address;
    uchar column_address_L,column_address_H;
    page=page-1;
    page_address = 0x40+page;
    column=column+3;
    column_address_L =0xe0+(column&0x0f);
    column_address_H =0xf0+((column>>4)&0x0f);

```

```

for(j=0;j<2;j++)
{
    start_flag();          /*开始标志*/
    transfer(0x7e);        /*选择 SLAVE ADDRESS*/
    transfer(0x00);        /*控制字：表示以下传输的 N 个字节是指令,直到 STOP_FLAG(停止标志为止)*/

```

```

transfer(0x38);          /*功能设置：MX=0,MY=0,PD=0,H1=0,H0=0,*/
transfer(page_address+j); /*设置页地址*/
transfer(column_address_L); /*设置页地址低位*/
transfer(column_address_H); /*设置页地址高位*/
stop_flag();
start_flag();
transfer(0x7e);
transfer(0x40);          /*控制字：表示以下传输的N个字节都是显示数据*/
for(i=0;i<8;i++)
{
    transfer(*dp);
    dp++;
}
stop_flag();
}
}

void clear_screen()
{
    int i, j;
    for(j=0;j<10;j++)
    {
        start_flag();          /*开始标志*/
        transfer(0x7e);        /*选择 SLAVE ADDRESS*/
        transfer(0x00);        /*控制字：表示以下传输的N个字节是指令*/
        transfer(0x38);        /*功能设置：MX=0,MY=0,PD=0,H1=0,H0=0,*/
        transfer(0x40+j);      /*设置页地址*/
        transfer(0xe0);        /*设置页地址低位*/
        transfer(0xf0);        /*设置页地址高位*/
        stop_flag();
        start_flag();
        transfer(0x7e);
        transfer(0x40);        /*控制字：表示以下传输的N个字节都是显示数据*/
        for(i=0;i<132;i++)
        {
            transfer(0x00);
        }
        stop_flag();
    }
}

void initial_lcd()
{
    start_flag();          //开始标志

    transfer(0x7e);        //选择 SLAVE ADDRESS
    transfer(0x00);        //表示以下传输的字节是指令*/
    transfer(0x3b);        //功能设置：MX=1,MY=1,PD=0,H1=1,H0=1,
    transfer(0x03);        //软件复位*/
    transfer(0x38);        //功能设置：MX=1,MY=1,PD=0,H1=0,H0=0,
    transfer(0x05);        //设置 VLCD 的范围：0X05 表示高电压 0X04 表示低电压
    transfer(0x0c);        //打开显示
    transfer(0x39);        //功能设置：MX=1,MY=1,PD=0,H1=0,H0=1,
    transfer(0x08);        //显示配置：D0=0,V=0,(Top/bottom row mode set data order)
    transfer(0x10);        //BIAS 设置为：0x10:1/11,0x11:1/10,0x12:1/9
    transfer(0x8f);        //对比度设置：最低是 0x80, 最高是 0xff, 数值越大对比度就越高

    stop_flag();          //结束标志
}

void main()
{
    reset=0;
    delay(100);
    reset=1;
    initial_lcd();
    while(1)
    {
        clear_screen();          //clear all dots
    }
}

```



```

display_graphic_128x64(1, 1, bmp_12864_3); //在第 1 页, 第 1 列显示 128x64 点阵的图片
waitkey();
display_graphic_128x64(1, 1, bmp_12864_2); //在第 1 页, 第 1 列显示 128x64 点阵的图片
waitkey();
display_graphic_128x64(1, 1, bmp_12864_1); //在第 1 页, 第 1 列显示 128x64 点阵的图片
waitkey();
display_graphic_128x64(1, 1, bmp_12864_4); //在第 1 页, 第 1 列显示 128x64 点阵的图片
waitkey();
display_graphic_128x64(1, 1, bmp_12864_5); //在第 1 页, 第 1 列显示 128x64 点阵的图片
waitkey();
display_graphic_128x64(1, 1, bmp_12864_6); //在第 1 页, 第 1 列显示 128x64 点阵的图片
waitkey();

clear_screen(); //clear all dots
display_graphic_32x32(1, 1, yun1); //在第 1 页, 第 1 列显示单个汉字“运”
display_graphic_32x32(1, (1+32+16), xing1); //在第 1 页, 第 49 列显示单个汉字“行”
display_graphic_32x32(1, (1+32*3), zhong1); //在第 1 页, 第 97 列显示单个汉字“中”

display_graphic_16x16(5, 1, gel); //“歌”
display_graphic_16x16(5, 17, qul); //“曲”
display_graphic_16x16(5, 33, ming1); //“名”
display_graphic_8x16(5, 49, mao_hao); //“:”
display_graphic_16x16(5, 57, chuan1); //“传”
display_graphic_16x16(5, 73, qi1); //“奇”

display_graphic_16x16(7, 1, yan1); //“演”
display_graphic_16x16(7, 17, chang1); //“唱”
display_graphic_16x16(7, 33, zhe1); //“者”
display_graphic_8x16(7, 49, mao_hao); //“:”
display_graphic_16x16(7, 57, wang1); //“王”
display_graphic_16x16(7, 73, fei1); //“菲”

waitkey();
}
}

char code yun1[]={
/*-- 文字: 运 --*/
/*-- 宋体 23; 此字体下对应的点阵为: 宽 x 高=31x31 --*/
/*-- 高度不是 8 的倍数, 现调整为: 宽度 x 高度=31x32 --*/
0x00, 0x00, 0x00, 0x00, 0x0C, 0x1C, 0xF8, 0xF0, 0x60, 0x00, 0x00, 0x00, 0x30, 0x30, 0x30, 0x30,
0x30, 0x30, 0x30, 0x30, 0x30, 0x30, 0x30, 0x30, 0x38, 0x38, 0x30, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x10, 0x30, 0x30, 0x10, 0x10, 0xF8, 0xF8, 0x30, 0x00, 0x08, 0x08, 0x08, 0x08, 0x88, 0xE8,
0xF8, 0x78, 0x38, 0x08, 0x08, 0x08, 0x08, 0x08, 0x0C, 0x0E, 0x0C, 0x08, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x80, 0x80, 0xFF, 0xFF, 0x80, 0x00, 0x00, 0x30, 0x78, 0x7C, 0x7E, 0x37, 0x31, 0x30,
0x30, 0x30, 0x30, 0x33, 0x17, 0x1E, 0x7C, 0xF8, 0x70, 0x00, 0x00, 0x00, 0x00, 0x00, 0x06, 0x0F,
0x07, 0x03, 0x01, 0x00, 0x01, 0x03, 0x07, 0x06, 0x0E, 0x0C, 0x0C, 0x1C, 0x1C, 0x1C, 0x1C, 0x18,
0x18, 0x18, 0x18, 0x1C, 0x1C, 0x1C, 0x1C, 0x1C, 0x0C, 0x04, 0x00,
};

char code xing1[]={
/*-- 文字: 行 --*/
/*-- 宋体 23; 此字体下对应的点阵为: 宽 x 高=31x31 --*/
/*-- 高度不是 8 的倍数, 现调整为: 宽度 x 高度=31x32 --*/
0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xE0, 0x70, 0x3C, 0x1C, 0x18, 0x00, 0x00, 0x00, 0x30, 0x30,
0x10, 0x10, 0x10, 0x10, 0x10, 0x10, 0x10, 0x10, 0x18, 0x18, 0x18, 0x10, 0x00, 0x00, 0x00, 0x00,
0x00, 0x06, 0x02, 0x03, 0x81, 0xE0, 0xF0, 0xFC, 0xDF, 0x0F, 0x02, 0x18, 0x18, 0x18, 0x18, 0x18,
0x18, 0x18, 0x18, 0xF8, 0xF8, 0xF8, 0x18, 0x18, 0x18, 0x1C, 0x1C, 0x18, 0x10, 0x00, 0x00, 0x08,
0x0C, 0x06, 0x03, 0x03, 0x01, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x3F, 0x1F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x08, 0x18, 0x38,
0x38, 0x3F, 0x3F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
};

char code zhong1[]={
/*-- 文字: 中 --*/
/*-- 宋体 23; 此字体下对应的点阵为: 宽 x 高=31x31 --*/
/*-- 高度不是 8 的倍数, 现调整为: 宽度 x 高度=31x32 --*/
0x00, 0x00, 0x00, 0x00, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFE, 0xFC,
0x0C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0xFF, 0xFF, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xFF, 0xFF, 0x01,

```

```

0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x0F, 0x0F, 0x02, 0x02, 0x02, 0x02, 0x02, 0x02, 0x02, 0x02, 0xFF, 0xFF, 0x02, 0x02,
0x02, 0x02, 0x02, 0x02, 0x02, 0x02, 0x0F, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x3F, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
};

char code gel[]={
/*-- 文字: 歌 --*/
/*-- 宋体 12; 此字体下对应的点阵为: 宽 x 高=16x16 --*/
0x80, 0xBA, 0xAA, 0xAA, 0xBA, 0x82, 0xFE, 0xA2, 0x90, 0x0C, 0xEB, 0x08, 0x28, 0x18, 0x08, 0x00,
0x00, 0x1E, 0x12, 0x12, 0x5E, 0x80, 0x7F, 0x40, 0x20, 0x18, 0x07, 0x08, 0x30, 0xE0, 0x40, 0x00
};

char code qu1[]={
/*-- 文字: 曲 --*/
/*-- 宋体 12; 此字体下对应的点阵为: 宽 x 高=16x16 --*/
0x00, 0x00, 0xF8, 0x08, 0x08, 0xFF, 0x08, 0x08, 0x08, 0xFF, 0x08, 0x08, 0x08, 0xF8, 0x00, 0x00,
0x00, 0x00, 0x7F, 0x21, 0x21, 0x3F, 0x21, 0x21, 0x21, 0x3F, 0x21, 0x21, 0x21, 0x7F, 0x00, 0x00
};

char code ming1[]={
/*-- 文字: 名 --*/
/*-- 宋体 12; 此字体下对应的点阵为: 宽 x 高=16x16 --*/
0x00, 0x40, 0x20, 0x10, 0x08, 0x27, 0x44, 0x84, 0x44, 0x24, 0x14, 0x0C, 0x04, 0x00, 0x00, 0x00,
0x04, 0x04, 0x04, 0x02, 0x7E, 0x23, 0x23, 0x22, 0x22, 0x22, 0x22, 0x22, 0x7E, 0x00, 0x00, 0x00
};

char code chuan1[]={
/*-- 文字: 传 --*/
/*-- 宋体 12; 此字体下对应的点阵为: 宽 x 高=16x16 --*/
0x40, 0x20, 0xF8, 0x07, 0x42, 0x44, 0x44, 0x44, 0xF4, 0x4F, 0x44, 0x44, 0x46, 0x64, 0x40, 0x00,
0x00, 0x00, 0x7F, 0x00, 0x00, 0x00, 0x02, 0x0B, 0x12, 0x22, 0x52, 0x0A, 0x07, 0x02, 0x00, 0x00
};

char code qil[]={
/*-- 文字: 奇 --*/
/*-- 宋体 12; 此字体下对应的点阵为: 宽 x 高=16x16 --*/
0x80, 0x80, 0xC4, 0xC4, 0xA4, 0xA4, 0x94, 0x8F, 0x94, 0x94, 0xA4, 0xE4, 0xA4, 0x80, 0x80, 0x00,
0x00, 0x00, 0x00, 0x3E, 0x12, 0x12, 0x12, 0x3E, 0x00, 0x40, 0x80, 0x7F, 0x00, 0x00, 0x00, 0x00
};

char code yan1[]={
/*-- 文字: 演 --*/
/*-- 宋体 12; 此字体下对应的点阵为: 宽 x 高=16x16 --*/
0x10, 0x21, 0x86, 0x60, 0x00, 0x0C, 0xD4, 0x54, 0x55, 0xF6, 0x54, 0x54, 0xD4, 0x0C, 0x04, 0x00,
0x04, 0xFC, 0x03, 0x00, 0x00, 0x00, 0x8F, 0x69, 0x09, 0x0F, 0x09, 0x29, 0xCF, 0x00, 0x00, 0x00};

char code chang1[]={
/*-- 文字: 唱 --*/
/*-- 宋体 12; 此字体下对应的点阵为: 宽 x 高=16x16 --*/
0xFE, 0x02, 0x02, 0xFE, 0x00, 0x80, 0xBE, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xBE, 0x80, 0x00, 0x00,
0x07, 0x02, 0x02, 0x07, 0x00, 0x7F, 0x24, 0x24, 0x24, 0x24, 0x24, 0x24, 0x24, 0x7F, 0x00, 0x00};

char code zhe1[]={
/*-- 文字: 者 --*/
/*-- 宋体 12; 此字体下对应的点阵为: 宽 x 高=16x16 --*/
0x00, 0x20, 0x24, 0x24, 0x24, 0x24, 0xA4, 0xBF, 0x64, 0x24, 0x34, 0x28, 0x26, 0x24, 0x20, 0x00,
0x04, 0x04, 0x04, 0x02, 0x02, 0xFF, 0x49, 0x49, 0x49, 0x49, 0x49, 0x49, 0xFF, 0x00, 0x00, 0x00};

char code wang1[]={
/*-- 文字: 王 --*/
/*-- 宋体 12; 此字体下对应的点阵为: 宽 x 高=16x16 --*/
0x00, 0x02, 0x82, 0x82, 0x82, 0x82, 0x82, 0xFE, 0x82, 0x82, 0x82, 0x82, 0x82, 0xC3, 0x82, 0x00, 0x00,
0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x3F, 0x20, 0x20, 0x20, 0x20, 0x20, 0x30, 0x20, 0x00};

char code fei1[]={
/*-- 文字: 菲 --*/
/*-- 宋体 12; 此字体下对应的点阵为: 宽 x 高=16x16 --*/

```


};

```

char code bmp_12864_3[]={
/*-- 调入了一幅图像: D:\e\新开发部\显示图案收藏\12864G-200.bmp  --*/
/*-- 宽度 x 高度=128x64  --*/
0xFF,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,
0x01,0xF9,0xA9,0xA9,0xA9,0xA9,0xA9,0xA9,0xF9,0x01,0x01,0x01,0x01,0x09,0xF9,0x49,
0x49,0x49,0xF9,0x49,0x45,0x49,0x71,0xC1,0x61,0x5D,0x49,0x41,0x01,0x81,0x85,0xB9,
0x91,0x01,0x09,0x09,0xF9,0x09,0x09,0x09,0x09,0xF9,0x01,0x01,0x01,0x21,0x21,0xE1,
0x21,0x21,0x01,0x01,0x19,0x85,0x45,0x25,0x19,0x01,0x01,0x01,0x01,0xC1,0x21,0x21,
0x21,0x21,0xE1,0x01,0x01,0x01,0x01,0xE1,0x25,0x39,0x21,0x21,0x21,0x31,0x2D,0xE1,
0x01,0x01,0x01,0x01,0x01,0x81,0x61,0x1D,0x89,0x01,0x41,0x41,0x41,0xFD,0x41,0x49,
0x71,0x51,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0xFF,
0xFF,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFC,0x94,
0x94,0x95,0x94,0xFC,0x00,0x00,0xFC,0x94,0x95,0x94,0x94,0xFC,0x00,0x20,0x7F,0x22,
0x22,0x12,0xFF,0x14,0x04,0x04,0xC4,0x3F,0x44,0x84,0x04,0x04,0x00,0x00,0x00,0x00,
0xFF,0x80,0x41,0x01,0xFF,0x01,0x01,0x01,0x00,0xFF,0x00,0xE0,0x00,0x80,0x80,0xFF,
0x80,0x80,0x00,0x00,0x03,0x02,0x02,0x02,0x02,0x00,0x00,0x00,0x1F,0x60,0x80,0x80,
0x80,0x40,0x20,0x00,0x00,0x80,0xE0,0x03,0xF2,0x02,0x02,0x0A,0x32,0x02,0x02,0xC3,
0x10,0x60,0xC0,0x00,0x81,0x89,0x8D,0x4B,0x49,0x48,0x82,0x82,0x42,0x4F,0x32,0x52,
0x8A,0x02,0xE2,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFF,
0xFF,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x08,0x08,0xF9,0x08,
0x08,0x08,0xF9,0x08,0x08,0x01,0x00,0x00,0x00,0x08,0x09,0x18,0x28,0xC8,0x00,
0xC0,0x28,0x1B,0x08,0x0A,0x01,0x00,0x10,0x10,0xF8,0x03,0x01,0x00,0x00,0x30,0x08,
0x09,0x08,0x10,0xE0,0x01,0x00,0x70,0x88,0x08,0x08,0x89,0x70,0x00,0x00,0x80,0x60,
0x90,0x88,0x08,0x08,0x00,0x00,0x80,0x80,0x60,0x10,0xF8,0x00,0x00,0x00,0xC0,0x20,
0x10,0x08,0x08,0x08,0x08,0x10,0x38,0x00,0x00,0x01,0x01,0x01,0x01,0x01,0x01,0x01,
0x00,0x30,0x08,0x08,0x08,0x10,0xE0,0x00,0x00,0xC0,0x20,0x10,0x10,0x20,0xC0,0x00,
0xC0,0x21,0x13,0x10,0x20,0xC0,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFF,
0xFF,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x30,0x20,0x20,0x1F,0x00,
0x00,0x20,0x20,0x3F,0x20,0x20,0x20,0x20,0x30,0x08,0x20,0x20,0x30,0x28,0x06,0x01,
0x06,0x28,0x30,0x20,0x20,0x00,0x00,0x00,0x20,0x3F,0x20,0x00,0x00,0x00,0x00,0x28,
0x24,0x22,0x21,0x30,0x00,0x00,0x1C,0x22,0x21,0x21,0x22,0x1C,0x00,0x00,0x0F,0x11,
0x20,0x20,0x11,0x0E,0x00,0x06,0x05,0x04,0x04,0x04,0x3F,0x04,0x00,0xC0,0x07,0x08,
0x10,0x20,0x20,0x20,0x21,0x21,0x9F,0x81,0x80,0x84,0x84,0x04,0x04,0x00,0xC0,0x00,
0x00,0x30,0x28,0x24,0x22,0x21,0x30,0x00,0x00,0x0F,0x10,0x20,0x20,0x10,0x0F,0x00,
0x0F,0x10,0x20,0x20,0x10,0x0F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFF,
0xFF,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x04,0x04,0xFE,0x00,0x00,0x00,0x00,0x00,0x1C,
0x02,0x02,0x82,0x62,0x1C,0x00,0x00,0x1C,0xA2,0x42,0x42,0xA2,0x1C,0x00,0x02,0x06,
0x1A,0xE0,0xE0,0x1A,0x06,0x02,0x00,0xF8,0x44,0x22,0x22,0x46,0x80,0x00,0x00,0xC0,
0x30,0x08,0x04,0xFE,0x00,0x00,0x00,0x00,0x00,0xF8,0x88,0x88,0x88,0x8F,0x89,0x89,
0x89,0xFD,0x09,0x00,0x00,0x00,0xFF,0x00,0x84,0x0A,0xF1,0x22,0x32,0x2E,0x23,0xFA,
0x22,0x22,0x22,0x22,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFF,
0xFF,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x40,0x80,0x80,0x00,0x00,0x80,
0x80,0x80,0x80,0x80,0x80,0x00,0x08,0x08,0x8F,0x88,0x88,0x80,0x80,0x80,0x8C,
0x8A,0x89,0x88,0x88,0x8C,0x80,0x00,0x07,0x08,0x08,0x08,0x08,0x07,0x00,0x08,0x0C,
0x0B,0x00,0x00,0x0B,0x0C,0x08,0x00,0x03,0x04,0x08,0x08,0x04,0x03,0x00,0x00,0x01,
0x01,0x09,0x09,0x0F,0x09,0x00,0x00,0x10,0x0C,0x01,0x04,0x18,0x00,0x02,0x04,0x18,
0x00,0x03,0x04,0x18,0x00,0x00,0x3F,0x00,0x00,0x01,0x00,0x01,0x01,0x01,0x01,0x3F,
0x01,0x01,0x01,0x01,0x01,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFF,
0xFF,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x82,0x42,0x22,0xD3,0x2A,0xC6,0x80,0x7F,
0x00,0x80,0x7E,0x80,0x00,0xFF,0x00,0x00,0x00,0xFF,0x00,0x00,0x04,0x08,0x88,0x50,
0x20,0x58,0x87,0x02,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x30,0x30,0x00,0x00,0x00,
0x00,0xF8,0x44,0x22,0x22,0x46,0x80,0x00,0x00,0x7E,0x42,0x22,0x22,0x42,0x82,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x7E,0x42,0x22,0x22,0x42,0x82,0x00,
0x00,0x20,0x60,0x80,0xA0,0x60,0x20,0x00,0x00,0x0C,0x02,0x22,0x22,0x52,0x8C,0x00,
0x00,0x1C,0xA2,0x42,0x42,0xA2,0x1C,0x00,0x20,0xE0,0x20,0x20,0xE0,0x20,0x20,0xC0,
0x20,0xE0,0x20,0x20,0xE0,0x20,0x20,0xC0,0x00,0x00,0x00,0x00,0x00,0x00,0xFF,
0xFF,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x9F,0x80,0x81,0x90,0x88,
0x86,0x81,0x80,0x8F,0x90,0x90,0x9C,0x80,0x80,0x8F,0x88,0x8C,0x8A,0x89,0x88,0x88,
0x88,0x88,0x89,0x8F,0x8A,0x88,0x80,0x80,0x80,0x80,0x80,0x80,0x8C,0x8C,0x80,0x80,0x80,
0x80,0x83,0x84,0x88,0x88,0x84,0x83,0x80,0x80,0x86,0x88,0x88,0x88,0x84,0x83,0x80,
0x80,0x8C,0x8C,0x80,0x80,0x80,0x80,0x80,0x86,0x88,0x88,0x88,0x84,0x83,0x80,
0x80,0x88,0x8C,0x8B,0x83,0x8C,0x88,0x80,0x80,0x86,0x88,0x88,0x88,0x84,0x83,0x80,
0x80,0x87,0x88,0x88,0x88,0x88,0x87,0x80,0x88,0x88,0x8F,0x88,0x80,0x8F,0x88,0x80,0x8F,
0x88,0x8F,0x88,0x80,0x8F,0x88,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0xFF,
};

```

```

char code bmp_12864_4[]={

```