

LCD12232系列点阵型液晶显示模块 使用说明书

一、液晶驱动IC基本特性

- 1、具有低功耗、供应电压范围宽等特点。
- 2、具有16common和61segment输出，并可外接驱动IC扩展驱动。
- 3、具有2560位显示RAM (DD RAM)，即 $80 \times 8 \times 4$ 位
- 4、具有与68系列或80系列相适配的MPU接口功能，并有专用的指令集，可完成文本显示或图形显示的功能设置

二、模块基本特性

视域尺寸：60.5×18.0mm (12232-1/-2)，54.8×18.3mm (12232-3)

显示类型：黄底黑字

LCD显示角度：6点钟直观

驱动方式：1/32 duty，1/6 bias

连接方式：导电胶条，铁框

- 补充说明：模块外观尺寸可根据用户的要求进行适度调整。

三、外形尺寸图

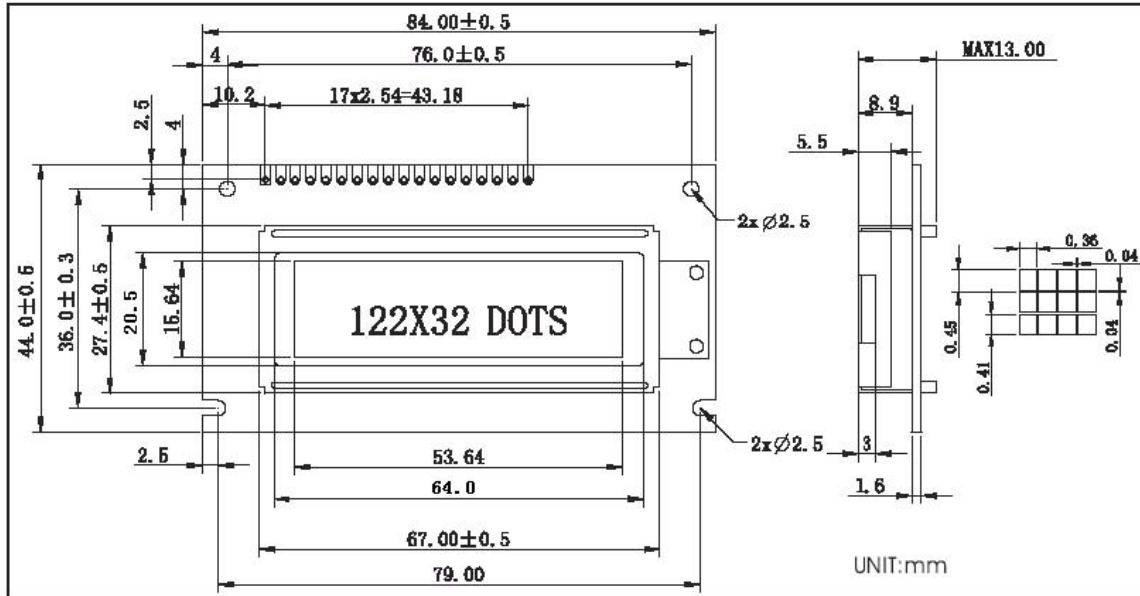


图1 12232-1尺寸图

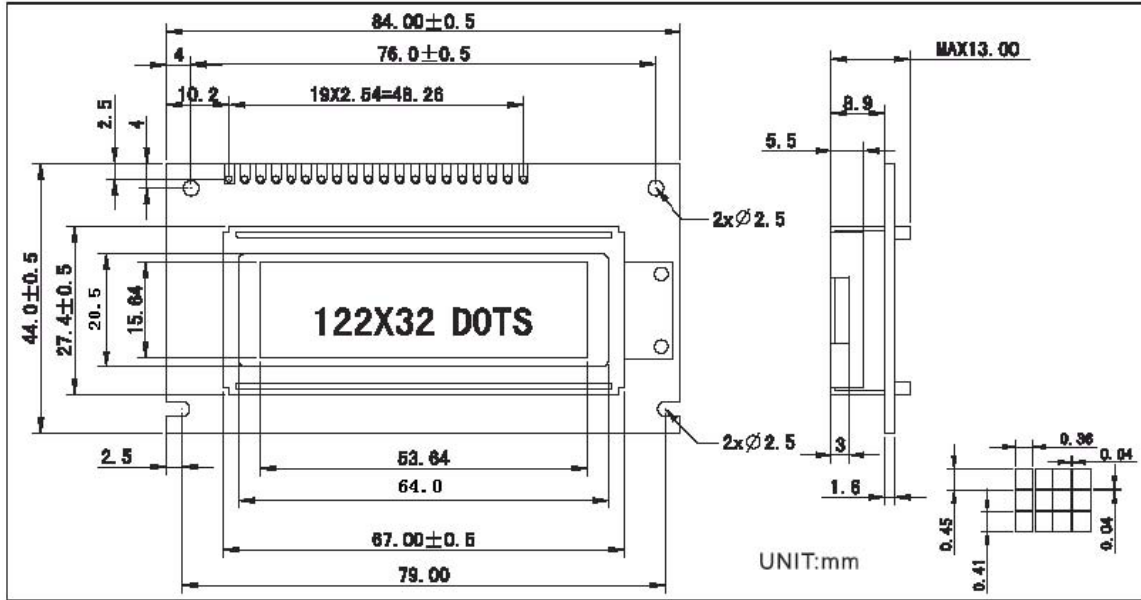


图2 12232-2尺寸图

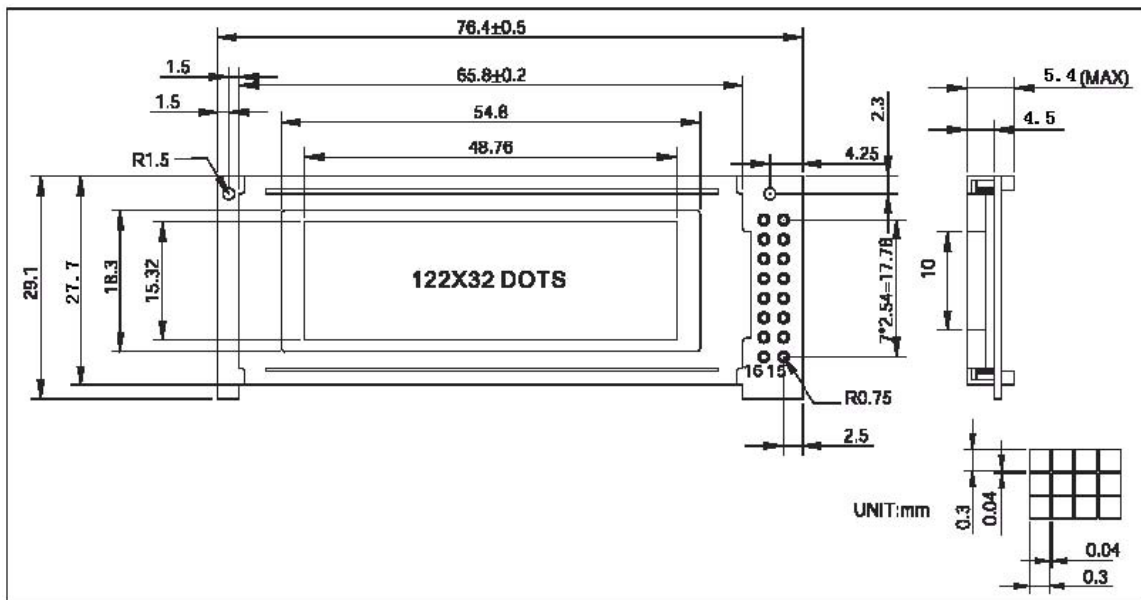


图3 12232-3尺寸图

四、工作参数

- 1、逻辑工作电压 (VDD-VSS) : 2.4 ~ 6.0V
- 2、LCD驱动电压(Vdd-Vlcd) : 3.0 ~ 13.5V
- 3、工作温度(Ta) : 0 ~ 55 (常温) / -20 ~ 70 (宽温)
- 4、保存温度(Tstg) : -10 ~ 70

五、电气特性(测试条件 Ta=25, Vdd=5.0 ± 0.25V)

- 1、输入高电平(Vih) : 3.5Vmin
- 2、输入低电平(Vil) : 0.55Vmax
- 3、输出高电平(Voh) : 3.75Vmin
- 4、输出低电平(Vol) : 1.0Vmax
- 5、工作电流 : 2.0Amax

六、接口说明

PCB板引脚号	12232/1	12232/2	12232/3
1	VDD	VSS	同12232-1引脚
2	VSS	VDD	
3	Vo	Vo	
4	/RESET	Ao	
5	E1	E1	
6	E2	E2	
7	R/W	CL(外振时钟)	
8	Ao	/RD	
9	D0	/WR	
10	D1	D0	
11	D2	D1	
12	D3	D2	
13	D4	D3	
14	D5	D4	
15	D6	D5	
16	D7	D6	
17	LED+	D7	
18	LED-	/RESET	
19		LED+	
20		LED-	

管脚说明:

VDD : 逻辑电源正

GND(VSS) : 逻辑电源地

VO(VEE) : LCD驱动电源

RESET : 复位端,对于68系列MPU:上升沿(L-H)复位,且复位后电平须保持为高电(H);

对于80系列MPU:下降沿(H-L)复位,且复位后电平须保持为低电平(L)。

E1 : 读写使能。对于68系列MPU,连接使能信号引脚,高电平有效;

对于80系列MPU,连接/RD引脚,低电平有效。

E2 : 同E1引脚。

/RD : 读允许,低电平有效。

/WR : 写允许,低电平有效。

R/W：读写选择，对于68系列MPU，高电平时读数据，低电平时写数据；
对于80系列MPU，低电平时允许数据传输，上升沿时锁定数据。
A0：数据/指令选择 高电平：数据D0-D7将送入显示RAM；
低电平：数据D0-D7将送入指令执行器执行。
D0-D7：数据输入输出引脚。

七、指令描述

1、显示模式设置（显示开关指令）

R/W											
CODE：	A0	/RD	/WR	D7	D6	D5	D4	D3	D2	D1	D0
	L	H	L	H	L	H	L	H	H	H	D

功能：开/关屏幕显示，不改变显示RAM(DD RAM)中的内容，也不影响内部状态。D=0，开显示；D=1，关显示。如果在显示关闭的状态下选择静态驱动模式，那么内部电路将处于安全模式。
该指令不影响显示RAM的内容。

2、设置显示起始行

R/W											
CODE：	A0	/RD	/WR	D7	D6	D5	D4	D3	D2	D1	D0
	L	H	L	H	H	L	显示起始行(1~31)				

功能：执行该命令后，所设置的行将显示在屏幕的第一行。起始地址可以是0-31范围内任意一行。
行地址计数器具有循环计数功能，用于显示行扫描同步，当扫描完一行后自动加一。

3、页地址设置

R/W											
CODE：	A0	/RD	/WR	D7	D6	D5	D4	D3	D2	D1	D0
	L	H	L	H	L	H	H	H	L	A1	A0

功能：设置页地址。当MPU要对DD RAM进行读写操作时，首先要设置页地址和列地址。本指令不影响显示。

A1	A0	页地址
0	0	0
0	1	1
1	0	2
1	1	3

4、列地址设置

R/W											
CODE：	A0	/RD	/WR	D7	D6	D5	D4	D3	D2	D1	D0
	L	H	L	L	A6	A5	A4	A3	A2	A1	A0

功能：设置DD RAM中的列地址。当MPU要对DD RAM进行读写操作前，首先要设置页地址和列地址。执行读写命令后，列地址会自动加1，直到达到50H才会停止，但页地址不变。

A6	A5	A4	A3	A2	A1	A0	列地址
0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1
1	0	0	1	1	1	0	4E
1	0	0	1	1	1	1	4F

5、 读状态指令

R/W											
CODE :	A0	/RD	/WR	D7	D6	D5	D4	D3	D2	D1	D0
	L	L	H	BUSY	ADC	ON/OFF	RESET	L	L	L	L

功能：检测内部状态。

BUSY为忙信号位，BUSY =1：内部正在执行操作；BUSY =0：空闲状态。

ADC为显示方向位，ADC=0：反向显示；ADC=1：正向显示。

ON/OFF显示开关状态，ON/OFF=0：显示打开，ON/OFF=1：显示关闭。

RESET复位状态，RESET=0：正常，RESET=1：内部正处于复位初始化状态。

6、 写显示数据

R/W											
CODE :	A0	/RD	/WR	D7	D6	D5	D4	D3	D2	D1	D0
	H	H	L	Write Data							

功能：将8位数据写入DD RAM，该指令执行后，列地址自动加1，所以可以连续将数据写入DD RAM而不用重新设置列地址。

7、 读显示数据

R/W											
CODE :	A0	/RD	/WR	D7	D6	D5	D4	D3	D2	D1	D0
	H	L	H	Read Data							

功能：读出页地址和列地址限定的DD RAM地址内的数据。当“读-修改-写模式”关闭时，每执行一次读指令，列地址自动加1，所以可以连续从DD RAM读出数据而不用设置列地址。

注意：再设置完列地址后，首次读显示数据前必须执行一次空的“读显示数据”。这是因为设置完列地址后，第一次读数据时，出现在数据总线上的数据是列地址而不是所要读出的数据。

8、 设置显示方向

R/W											
CODE :	A0	/RD	/WR	D7	D6	D5	D4	D3	D2	D1	D0
	L	H	L	H	L	H	L	L	L	L	D

功能：该指令设置DD RAM中的列地址与段驱动输出的对应关系

显示当设置D=0时，反向；D=1时，正向。

9、开/关静态驱动模式设置

R/W											
CODE :	A0	/RD	/WR	D7	D6	D5	D4	D3	D2	D1	D0
	L	H	L	H	L	H	L	L	H	L	D

功能：D=0表正常驱动，D=1表示打开静态显示。

如果在打开静态显示时，执行关闭显示指令，内部电路将被置为安全模式。

10、DUTY选择

R/W											
CODE :	A0	/RD	/WR	D7	D6	D5	D4	D3	D2	D1	D0
	L	H	L	H	L	H	L	H	L	L	D

功能：设置D=0 表示1/16DUTY，D=1表示1/32DUTY。

11、“读-修改-写”模式设置

R/W											
CODE :	A0	/RD	/WR	D7	D6	D5	D4	D3	D2	D1	D0
	L	H	L	H	H	H	L	L	L	L	L

功能：执行该指令以后，每执行一次写数据指令列地址自动加1；但执行读数据指令时列地址不会改变。这个状态一直持续到执行“END”指令。

注意：在“读-修改-写”模式下，除列地址设置指令之外，其他指令照常执行。

12、END指令

R/W											
CODE :	A0	/RD	/WR	D7	D6	D5	D4	D3	D2	D1	D0
	L	H	L	H	H	H	L	H	H	H	L

功能：关闭“读-修改-写”模式，并把列地址指针恢复到打开“读-修改-写”模式前的位置。

13、复位指令

R/W											
CODE :	A0	/RD	/WR	D7	D6	D5	D4	D3	D2	D1	D0
	L	H	L	H	H	H	L	L	L	H	L

功能：使模块内部初始化。

初始化内容： 设置显示初始行为第一行；

页地址设置为第三页。

复位指令对显示RAM没有影响。

14、设置安全模式

通过关闭显示并打开静态显示的方法，可以设置安全模式，以减小功耗。

安全模式下的内部状态：

停止LCD驱动。Segment和Common输出VDD电平。

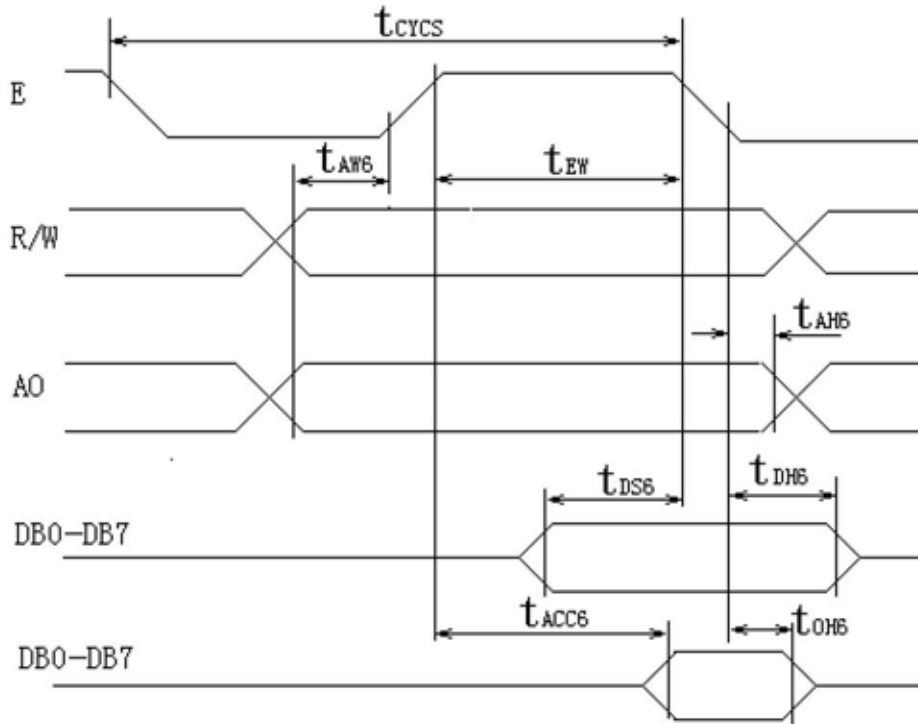
停止晶体震荡并禁止外部时钟输入，晶振输入OSC₂引脚处于不确定状态。

显示数据和内部模式不变。

可通过打开显示或关闭静态显示的方法关闭安全模式。

八、接口时序

与68系列MPU接口读写操作时序图

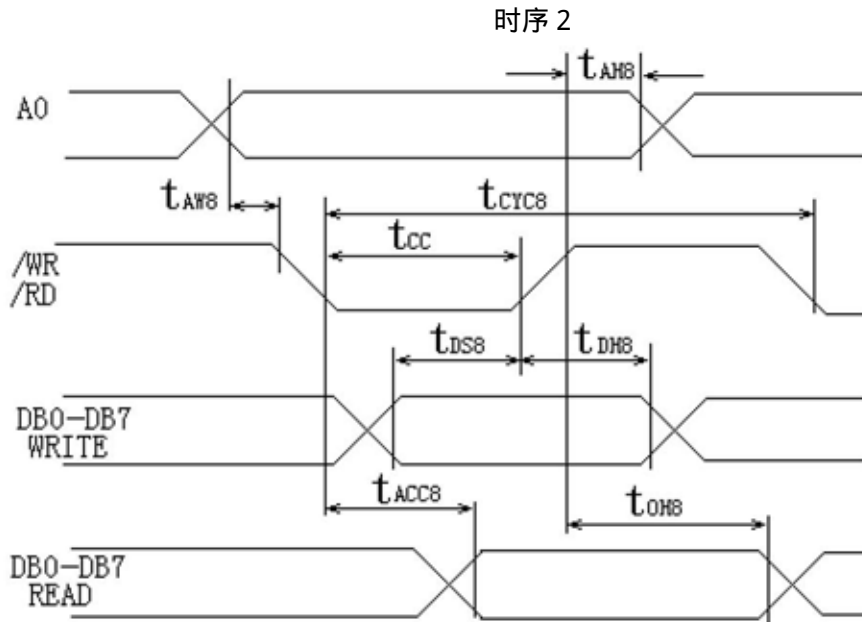


时序 1

时序参数表 (VDD=5.0 ± 10%, VSS=0V, Ta=-20 ~ +75)

名称	符号	最小值	最大值	单位
地址建立时间	Taw6	20		ns
地址保持时间	Tah6	10		ns
系统时钟周期	Tcycs	1000		ns
E脉冲宽度	Tew	100		ns
		80		ns
数据建立时间	Tds6	80		ns
写数据保持时间	Tdh6	10		ns
存取时间	Tacc6		90	ns
读数据保持时间	Tch6	10	60	ns

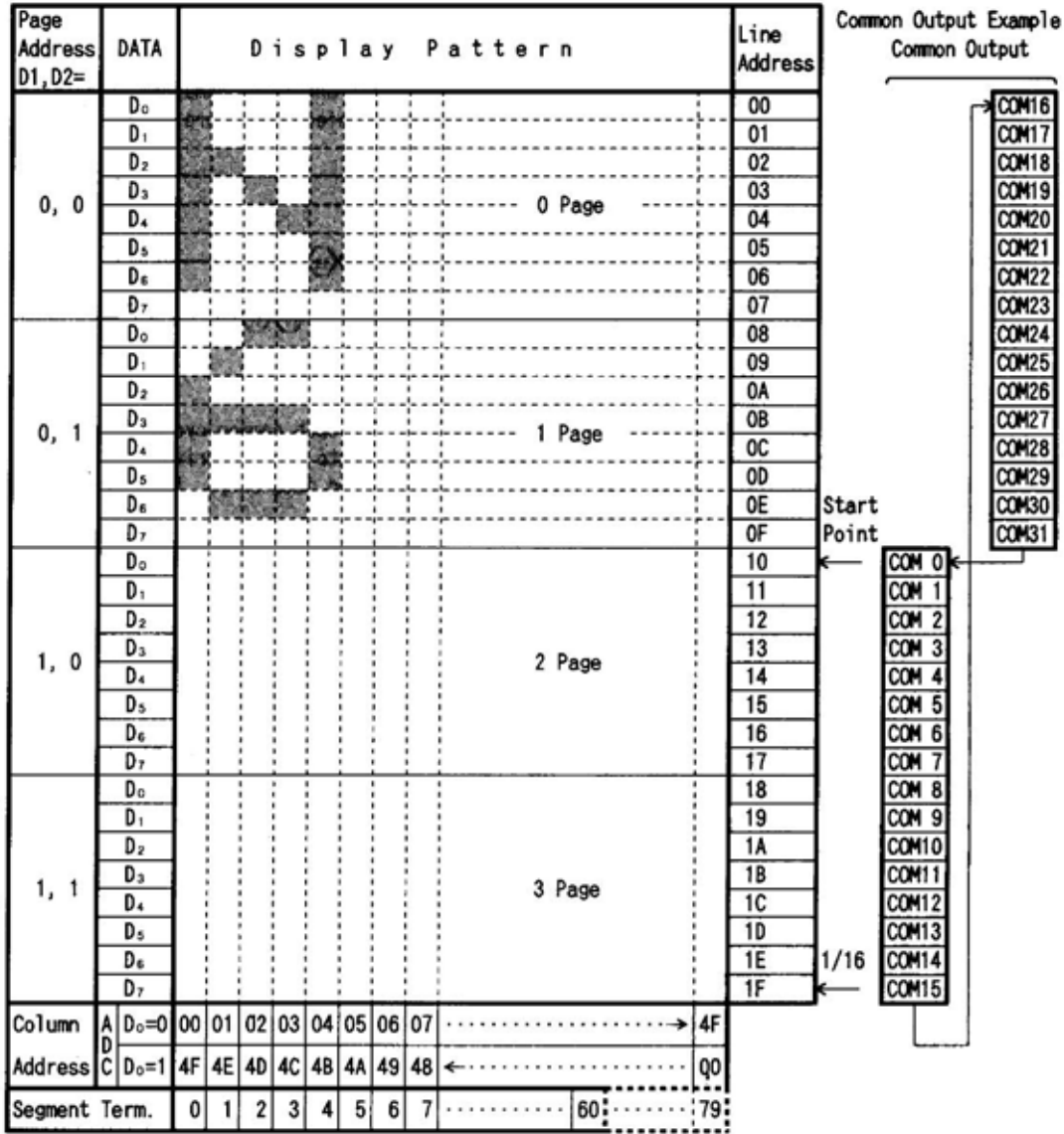
与80系列MPU接口读写操作时序图



时序参数表 ($V_{DD}=5.0 \pm 10\%$, $V_{SS}=0V$, $T_a=-20 \sim +70$)

名称	符号	最小值	最大值	单位
地址建立时间	T_{aw8}	20		ns
地址保持时间	T_{ah8}	10		ns
/RW, /WR周期	T_{cyc8}	1000		ns
/RW, /WR脉冲宽度	T_{cc}	200		ns
数据建立时间	T_{ds8}	80		ns
写数据保持时间	T_{dh8}	10		ns
读存取时间	T_{acc8}		90	ns
读数据保持时间	T_{ch8}	10	60	ns

九、显示数据存储器 (DDRAM) 与地址的对应关系
(显示设定为 1/32 DUTY , 显示起始行为10th)



十、应用举例

1、 模块接口与8031单片机测架连接表 (以OCM12232-1模块为例) :

模块引脚	符号	8031单片机引脚
3	V _o	与GND间接一可调电阻
4	RET	V _{cc}
5	E1	TXD
6	E2	RXD
7	R/W	GND

8	A0	INT1
9 ~ 16	D0 ~ D7	P1口

2、简单的测试程序(模拟68MPU接口时序)

```

; *****给主IC发送控制指令子程序*****
S_INS1:SETB    TXD
        CLR     INT1
        MOV     P1,A
        CLR     TXD
        LCALL  DELAY
        RET
; *****给主IC发送显示数据子程序*****
S_DAT1:SETB    TXD
        SETB    INT1
        MOV     P1,A
        CLR     TXD
        LCALL  DELAY
        RET
; *****给辅IC发送控制指令子程序*****
S_INS2:SETB    RXD
        CLR     INT1
        MOV     P1,A
        CLR     RXD
        LCALL  DELAY
        RET
; *****给辅IC发送显示数据子程序*****
S_DAT2:SETB    RXD
        SETB    INT1
        MOV     P1,A
        CLR     RXD
        LCALL  DELAY
        RET
; *****显示错位点程序*****
MOV     R3,    #55H
MOV     R4,    #0AAH
LDOT:  MOV     R2,#0B8H
LDOT1: MOV     A,R2    ;页地址设置,0页
        LCALL  S_INS1
        LCALL  S_INS2
MOV     A,#00H        ;列地址设置,0列
        LCALL  S_INS1
        LCALL  S_INS2
MOV     R1,#1EH      ;发送显示数据
LDOT2: MOV     A,      R3
        LCALL  S_DAT1
        LCALL  S_DAT2
MOV     A,      R4
        LCALL  S_DAT1
    
```

```
LCALL S_DAT2
DJNZ R1, LDOT2
MOV A, R3
LCALL S_DAT1
LCALL S_DAT2
INC R2
CJNE R2,#0BCH,LDOT1
LCALL DEALY
RET
```

3、12232点阵液晶基本驱动程序

```
/*-----
【说明】12232点阵液晶基本驱动程序，国内很多122×32的液晶与之完全兼容，
该液晶左右分MASTER和SLAVE，上下共分4页，左边列地址从0-61，
右边列地址从0-61，由于芯片内部线路的落后，所以使用较麻烦。
```

```
【文件】12232lcd.c @ 2001/03/23
【作者】龙啸九天 c51@yeah.net http://mcs51.yeah.net
【修改】修改建议请到论坛公布 http://c51bbs.yeah.net
【版本】V1.00A Build 0323
```

```
-----*/
#define E2 P3_5
#define E1 P3_6

#define A0 P3_7
#define DATA P2

/*-----
调用方式：void OutMI(uchar i)
函数说明：发指令 i 到主窗口。（内函数，私有，用户不直接调用）
-----*/
void OutMI(uchar i)
{
    E1=1;_nop();_nop();
    A0=0;_nop();_nop();
    DATA=i;_nop();_nop();
    E1=0;_nop();_nop();
}
/*-----
调用方式：void OutMD(uchar i)
函数说明：发数据 i 到主窗口。（内函数，私有，用户不直接调用）
-----*/
void OutMD(uchar i)
{
    E1=1;_nop();_nop();
    A0=1;_nop();_nop();
    DATA=i;_nop();_nop();
    E1=0;_nop();_nop();
}
/*-----
调用方式：void OutSI(uchar i)
函数说明：发指令 i 到从窗口。（内函数，私有，用户不直接调用）
-----*/
void OutSI(uchar i)
```

```
{
  A0=0;_nop();_nop();
  E2=1;_nop();_nop();
  DATA=i;_nop();_nop();
  E2=0;_nop();_nop();
}
/*-----
调用方式：void OutSD(uchar i)
函数说明：发数据 i 到从窗口。（内函数，私有，用户不直接调用）
-----*/

void OutSD(uchar i)
{
  A0=1;_nop();_nop();
  E2=1;_nop();_nop();
  DATA=i;_nop();_nop();
  E2=0;_nop();_nop();
}
/*-----
调用方式：void LcdIni(void)
函数说明：12232 点阵液晶初始化，开机后仅调用一次。
-----*/

void LcdIni(void)
{
  OutMI(0XE2);OutSI(0XE2);//复位
  OutMI(0XAE);OutSI(0XAE);//POWER SAVE
  OutMI(0XA4);OutSI(0XA4);//动态驱动
  OutMI(0XA9);OutSI(0XA9);//1/32 占空比
  OutMI(0XA0);OutSI(0XA0);//时钟线输出
  OutMI(0XEE);OutSI(0XEE);//写模式
  OutMI(0X00);OutMI(0XC0);
  OutSI(0X00);OutSI(0XC0);
  OutMI(0XAF);OutSI(0XAF);
}
/*-----
调用方式：void SetPage(uchar page0,uchar page1)
函数说明：同时设置主从显示页为 0 - 3 页。（内函数，私有，用户不直接调用）
-----*/

void SetPage(uchar page0,uchar page1)
{
  OutMI(0xB8|page1);OutSI(0xB8|page0);
}
/*-----
调用方式：void SetAddress(uchar address0,uchar address1)
函数说明：同时设置主从列地址为 0 - 121。（内函数，私有，用户不直接调用）
-----*/

void SetAddress(uchar address0,uchar address1)
{
  OutMI(address1&0x7F);OutSI(address0&0x7F);
}
/*-----
调用方式：void PutChar0(uchar ch)
函数说明：在左页当前地址画一个字节 8 个点。（内函数，私有，用户不直接调用）
-----*/

void PutChar0(uchar ch)
{
```

```
OutSD(ch);
}
/*-----
调用方式：void PutChar1(uchar ch)
函数说明：在右页当前地址画一个字节 8 个点。（内函数，私有，用户不直接调用）
-----*/
void PutChar1(uchar ch)
{
    OutMD(ch);
}
/*-----
调用方式：void DrawBmp(uchar x,bit layer,uchar width,uchar *bmp)
函数说明：画一个图，横坐标是 x，layer 表示上下层，width 是图形的宽，高都是 16，bmp 是
图形指针
使用 zimo21 软件，采用纵向取模得到 bmp 数据。
-----*/
void DrawBmp(uchar x0,bit layer,uchar width,uchar *bmp)
{
    uchar x,address,i=0; //address 表示显存的物理地址
    uchar page=0;
    bit window=0; //page 表示上下两页，window 表示左右两页
    if (layer) page=2;
    for (x=x0;x<x0+width;x++)
    {
        if (x>60) {window=1;address=x%61;}
        else address=x;
        SetPage(page,page);
        SetAddress(address,address);
        if (window) PutChar1(bmp[i]);
        else PutChar0(bmp[i]);//画上层
        SetPage(page+1,page+1);
        SetAddress(address,address);
        if (window) PutChar1(bmp[i+width]);
        else PutChar0(bmp[i+width]);//画下层
        i++;
    }
}
/*-----
调用方式：void clrscr(void)
函数说明：清屏
-----*/
void clrscr(void)
{
    uchar i;
    uchar page;
    for (page=0;page<4;page++)
    {
        SetPage(page,page);
        SetAddress(0,0);
        for (i=0;i<61;i++){PutChar0(0);PutChar1(0);}
    }
}
```

4、使用12232液晶显示图片演示

```
/*=====
```

功能：使用12232液晶显示图片演示

```
=====
```

硬件：TG12232B（122*32）模拟口线接线方式

连接线图：

```
-----  
|DB0-----P0.0 | DB4-----P0.4 | RW-----P1.3 | A0-----P1.4 |  
|DB1-----P0.1 | DB5-----P0.5 | RS-----P2.1 | V0接1K电阻到GND |  
|DB2-----P0.2 | DB6-----P0.6 | E1-----P1.1 |  
|DB3-----P0.3 | DB7-----P0.7 | E2-----P1.2 |  
-----
```

[注：AT89x52使用22.1184M或11.0592M晶振，实测使用22.1184M或11.0592都可以]

编译平台：Keil uV2 7.08

```
=====*/
```

```
#include <AT89X52.H>
```

```
#define uchar unsigned char
```

```
#define uint unsigned int
```

```
#define E1 P1_1 //块1 左边
```

```
#define E2 P1_2 //块2 右边
```

```
#define RW P1_3
```

```
#define A0 P1_4 //A0为1时是数据，0时为指令数据
```

```
#define DATA P0 //数据
```

```
extern unsigned char code Bmpt1[];  
extern unsigned char code Bmpt2[];  
extern unsigned char code Bmpt3[];  
extern unsigned char code Bmpt4[];  
extern unsigned char code Bmpt5[];  
extern unsigned char code Bmpt6[];  
extern unsigned char code Bmpt7[];  
extern unsigned char code Bmptc[];  
extern unsigned char code bmp001[];  
extern unsigned char code Bmp08[];  
extern unsigned char code Bmp07[];  
extern unsigned char code Bmp06[];  
extern unsigned char code Bmp05[];  
extern unsigned char code Bmp04[];  
extern unsigned char code Bmp03[];  
extern unsigned char code Bmp02[];  
extern unsigned char code Bmp01[];  
extern unsigned char code Bmp012[];  
extern unsigned char code Bmp003[];  
extern unsigned char code Bmp002[];
```

```
//延时
void delay(unsigned int i)
{
    unsigned char k=200;
    while(i>0)
    {
        i--;
    }
    while(k>1)k--;
}

//左页
//发送数据
void OUTMD(unsigned char i)
{
    A0=1;    //写数据
    //delay(5);
    E1=1;
    //delay(10);
    DATA=i;
    //delay(5);
    E1=0;
}

//左页
//发送命令
void OUTMI(unsigned char i)
{
    A0=0;    //写指令
    //delay(5);
    E1=1;
    //delay(10);
    DATA=i;
    //delay(5);
    E1=0;
}

//右页
//发送数据
void OUTSD(unsigned char i)
{
    A0=1;    //写数据
    //delay(5);
    E2=1;
    //delay(10);
    DATA=i;
    //delay(5);
    E2=0;
}
```

```
}

//右页
//发送命令
void OUTSI(unsigned char i)
{
    A0=0;    //写指令
    //delay(5);
    E2=1;
    //delay(10);
    DATA=i;
    //delay(5);
    E2=0;
}

//初始化
void Icdini(void)
{
    RW=0;
    OUTMI(0XE2);
    OUTSI(0XE2);//复位
    OUTMI(0XAE);
    OUTSI(0XAE);//POWER SAVE
    OUTMI(0XA4);
    OUTSI(0XA4);//动态驱动
    OUTMI(0XA9);
    OUTSI(0XA9);//1/32占空比
    OUTMI(0XA0);
    OUTSI(0XA0);//时钟线输出
    OUTMI(0XEE);
    OUTSI(0XEE);//写模式

    OUTMI(0X00);
    OUTMI(0XC0);
    OUTSI(0X00);
    OUTSI(0XC0);

    OUTMI(0XAF);
    OUTSI(0XAF);
}

/*-----
函数说明：同时设置主从显示页为0 - 3页。（内函数，私有，用户不直接调用）
-----*/
void SetPage(uchar page0,uchar page1)
{
    OUTMI(0xB8|page1);OUTSI(0xB8|page0);
}
}
```



```
/*-----  
函数说明：同时设置主从列地址为0 - 121。（内函数，私有，用户不直接调用）  
-----*/  
void SetAddress(uchar address0,uchar address1)  
{  
    OUTMI(address1);OUTSI(address0);  
}  
  
/*-----  
调用方式：void PutChar0(uchar ch)  
函数说明：在右页当前地址画一个字节8个点。（内函数，私有，用户不直接调用）  
-----*/  
void PutCharR(uchar ch)  
{  
    OUTSD(ch);  
}  
  
/*-----  
调用方式：void PutChar1(uchar ch)  
函数说明：在左页当前地址画一个字节8个点。（内函数，私有，用户不直接调用）  
-----*/  
void PutCharL(uchar ch)  
{  
    OUTMD(ch);  
}  
  
/*-----  
调用方式：void clrscr(void)  
函数说明：清屏  
-----*/  
void clrscr(void)  
{  
    uchar i;  
    uchar page;  
    for (page=0;page<4;page++)  
    {  
        SetPage(page,page);  
        SetAddress(0,0);  
        for (i=0;i<61;i++)  
        {  
            PutCharR(0);  
            PutCharL(0);  
        }  
    }  
}
```

```
/*-----  
调用方式：void DrawBmp(bit layer,uchar width,uchar *bmp)  
函数说明：画一个图,layer表示上下层,width是图形的宽,高都是16,bmp是图形指针  
            使用zimo3软件,纵向取模,字节倒序/240字节  
-----*/  
void DrawBmp(bit layer,uchar width,uchar *bmp)  
{  
    uchar x,address,i=0;    //address表示显存的物理地址  
    uchar page=0; //page表示上下两页  
    bit window=0; //window表示左右两页  
    //putcharR    //右边  
    //putcharL    //左边  
    for (x=width;x>1;x--)  
    {  
        if (i>60) {window=1;address=i%61;}  
        else address=i;  
  
        if(layer==0) //显示一行八个字  
        {  
            SetPage(0,0);  
            SetAddress(address,address);  
            if(window==1)PutCharR(bmp[i]);  
            else PutCharL(bmp[i]);  
            SetPage(1,1);  
            SetAddress(address,address);  
            if(window==1)PutCharR(bmp[i+width]);  
            else PutCharL(bmp[i+width]);  
        }  
    else  
        { //显示第二行八个汉字  
            SetPage(2,2);  
            SetAddress(address,address);  
            if(window==1)PutCharR(bmp[i]);  
            else PutCharL(bmp[i]);  
            SetPage(3,3);  
            SetAddress(address,address);  
            if(window==1)PutCharR(bmp[i+width]);  
            else PutCharL(bmp[i+width]);  
        }  
        i++;  
    }  
}
```

//-----
//图片显示采用字模2生成或字模3生成的都可以。
//-----
void DrawBmp1(uint x_add, uchar width,uchar *bmp)
{

```
uchar x,address,i=0; //address表示显存的物理地址
uchar page=0; //page表示上下两页
bit window=0; //window表示左右两页
//putcharR //右边
//putcharL //左边
for (x=width;x>1;x--)
{
if (x_add>60) {window=1;address=x_add%61;}
else address=x_add;

SetPage(0,0);
SetAddress(address,address);
if(window==1)PutCharR bmp[i];
else PutCharL bmp[i];
SetPage(1,1);
SetAddress(address,address);
if(window==1)PutCharR bmp[i+width];
else PutCharL bmp[i+width];

SetPage(2,2);
SetAddress(address,address);
if(window==1)PutCharR bmp[i+width+width];
else PutCharL bmp[i+width+width];
SetPage(3,3);
SetAddress(address,address);
if(window==1)PutCharR bmp[i+width+width+width];
else PutCharL bmp[i+width+width+width];
i++;
x_add++;
}
}

//-----
//-----
//采用zimo2 显示汉字,以汉字的显示方式
//void Draw_word(uchar d_where,uint x_add,bit layer,uchar width)
//d_where表示在码表中第几个汉字,x_add横坐标位置,layer显示的层,width显示的宽度。
//-----
void Draw_word(uchar d_where,uint x_add,bit layer,uchar width)
{
uchar x,i=0,address; //address表示显存的物理地址
uchar page=0; //page表示上下两页
bit window=0; //window表示左右两页
//putcharR //右边
//putcharL //左边
d_where=d_where*32;
```

```
for (x=width;x>1;x--)  
{  
if (x_add>60) {window=1;address=x_add%61;}  
else address=x_add;  
  
if(layer==0) //显示一行八个字  
{  
  
SetPage(0,0);  
SetAddress(address,address);  
if(window==1)PutCharR bmp001[d_where]; //右边  
else PutCharL bmp001[d_where]; //左边  
SetPage(1,1);  
SetAddress(address,address);  
if(window==1)PutCharR bmp001[d_where+width];  
else PutCharL bmp001[d_where+width];  
  
}  
else  
{ //显示第二行八个汉字  
SetPage(2,2);  
SetAddress(address,address);  
if(window==1)PutCharR bmp001[d_where]; //右边  
else PutCharL bmp001[d_where]; //左边  
SetPage(3,3);  
SetAddress(address,address);  
if(window==1)PutCharR bmp001[d_where+width];  
else PutCharL bmp001[d_where+width];  
  
}  
x_add++;  
d_where++;  
}  
}  
/*  
//-----  
//字模2显示一个汉字  
//纵向取模,字节倒序  
void disp_one(bit top_low,bit widows,uchar address,uchar width,uchar*bmp)  
{  
if(top_low==0)  
{  
SetPage(0,0);  
SetAddress(address,address);  
if(window==1)PutCharR bmp001[next][i];  
else PutCharL bmp001[next][i];  
SetPage(1,1);
```

```
        SetAddress(address, address);
        if(window==1)PutCharR bmp001[next][i+width]);
        else PutCharL bmp001[next][i+width]);
    }
else
    {
    SetPage(0,0);
    SetAddress(address, address);
    if(window==1)PutCharR bmp001[next][i]);
    else PutCharL bmp001[next][i]);
    SetPage(1,1);
    SetAddress(address, address);
    if(window==1)PutCharR bmp001[next][i+width]);
    else PutCharL bmp001[next][i+width]);
    }
}
*/
```

//液晶显示规则

```
//      M(左)          S(右)
//      page          page
//      0      |      0
//      1      |      1
//      2      |      2
//      3      |      3
```

```
void delay1s(unsigned char i)
{
    while(i>1)
    {
        i--;

        delay(65530);
    }
}
```

//显示动态的等待图标

```
void wait1(unsigned char i)
{
    for(;i>1;i--)
    {delay1s(2);
    clrscr(); //
    DrawBmp1(0,60,Bmptc); //
    DrawBmp1(76,19,Bmpt1); //
    delay1s(2);
    clrscr(); //
    DrawBmp1(0,60,Bmptc); //
    DrawBmp1(76,19,Bmpt2); //
```

```
    delay1s(2);
    clrscr(); //
    DrawBmp1(0,60,Bmptc); //
    DrawBmp1(76,19,Bmpt3); //
    delay1s(2);
    clrscr(); //
    DrawBmp1(0,60,Bmptc); //
    DrawBmp1(76,19,Bmpt4); //
    delay1s(2);
    clrscr(); //
    DrawBmp1(0,60,Bmptc); //
    DrawBmp1(76,19,Bmpt5); //
    delay1s(2);
    clrscr(); //
    DrawBmp1(0,60,Bmptc); //
    DrawBmp1(76,19,Bmpt6); //
    delay1s(2);
    clrscr(); //
    DrawBmp1(0,60,Bmptc); //
    DrawBmp1(76,19,Bmpt7); //
}

}
//演示程序
void main()
{

    lcdini(); //reset
    clrscr(); //clr

    Draw_word(0,0,0,16); //
    Draw_word(1,16,0,16); //
    Draw_word(2,32,0,16); //
    Draw_word(3,48,0,16); //
    Draw_word(0,64,0,16); //
    Draw_word(1,80,0,16); //
    Draw_word(2,96,0,16); //

    DrawBmp(1,120,Bmp002); //
    clrscr(); //
    delay1s(3);
    DrawBmp1(0,122,Bmp012); //LOGO
    while(1)
    {
        delay1s(3);

        clrscr(); //
```

```
DrawBmp1(10,101,Bmp901); //

delay1s(8);
clrscr(); //
wait1(3);
DrawBmp1(0,122,Bmp04); //
delay1s(8);
clrscr(); //
wait1(3);
DrawBmp1(0,122,Bmp03); //超前科技
delay1s(8);
clrscr(); //
wait1(3);
clrscr(); //
DrawBmp1(10,101,Bmp05); //
delay1s(8);
clrscr(); //
wait1(3);
clrscr(); //
DrawBmp1(10,101,Bmp06); //
delay1s(8);
clrscr(); //
wait1(3);
clrscr(); //
DrawBmp1(10,101,Bmp08); //
delay1s(8);
clrscr(); //
wait1(3);
clrscr(); //
DrawBmp1(0,122,Bmp901);
wait1(8);
}
}
```

ok.c图符文件

```
unsigned char code Bmpt1[]=
{
/*-- 调入了一幅图像：E:\新建文件夹\Wait_1.bmp --*/
/*-- 宽度x高度=19x30 --*/
/*-- 宽度不是8的倍数，现调整为：宽度x高度=19x32 --*/
0x00,0x00,0x00,0x60,0xE0,0xE0,0xE0,0xE0,0xA0,0xA0,0xA0,0xA0,0xA0,0xA0,0xA0,0xA0,
0x60,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x03,0x0C,0x13,0x27,0xDF,0x3F,0x3F,0xDF,0x27,
0x13,0x0C,0x03,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x80,0xF0,0xCC,0xC2,0xC1,0x40,0x40,
0x40,0x40,0x41,0x42,0x4C,0x70,0x80,0x00,0x00,0xC0,0xC0,0xC0,0xC1,0xC1,0xC1,0xC1,
0xC1,0xC1,0xC1,0xC1,0xC1,0xC1,0xC1,0xC1,0xC1,0xC1,0xC0,0xC0
};
unsigned char code Bmpt2[]=
```

迈讯科技

网址：<http://www.mycent.net>

Email：yfz_431@hotmail.com

项目开发 芯片解密 零件配单 TEL:15013652265 QQ:38537442

本文档的原始资料来源于网络

Dodo整理

QQ：6859620

```
{
/*-- 调入了一幅图像：E:\新建文件夹\Wait_2.bmp --*/
/*-- 宽度x高度=19x30 --*/
/*-- 宽度不是8的倍数，现调整为：宽度x高度=19x32 --*/
0x00,0x00,0x00,0x60,0xA0,0xA0,0xA0,0xA0,0xA0,0xA0,0xA0,0xA0,0xA0,0xE0,0xE0,0xE0,0xE0,
0x60,0x00,0x00,0x00,0x00,0x00,0x00,0x03,0x0C,0x12,0x2E,0xDE,0xFE,0xBE,0xDE,0x26,
0x12,0x0C,0x03,0x00,0x00,0x00,0x00,0x00,0x00,0x80,0x70,0x4C,0x42,0x61,0x60,0x61,
0x75,0x70,0xC1,0xC2,0xCC,0xF0,0x80,0x00,0x00,0xC0,0xC0,0xC0,0xC1,0xC1,0xC1,0xC1,
0xC1,0xC1,0xC1,0xC1,0xC1,0xC1,0xC1,0xC1,0xC0,0xC0

};
unsigned char code Bmpt3[]=
{
/*-- 调入了一幅图像：E:\新建文件夹\Wait_3.bmp --*/
/*-- 宽度x高度=19x30 --*/
/*-- 宽度不是8的倍数，现调整为：宽度x高度=19x32 --*/
0x00,0x00,0x00,0x60,0xA0,0xA0,0xA0,0xA0,0xA0,0xA0,0xA0,0xA0,0xE0,0xE0,0xE0,0xE0,
0x60,0x00,0x00,0x00,0x00,0x00,0x00,0x03,0x0C,0x10,0x20,0xD0,0xF0,0xB0,0xD0,0x20,
0x10,0x0C,0x03,0x00,0x00,0x00,0x00,0x00,0x00,0x80,0x70,0x4C,0x42,0x41,0x60,0x75,
0x7F,0x60,0xC1,0xC2,0xCC,0xF0,0x80,0x00,0x00,0xC0,0xC0,0xC0,0xC1,0xC1,0xC1,0xC1,
0xC1,0xC1,0xC1,0xC1,0xC1,0xC1,0xC1,0xC1,0xC0,0xC0

};
unsigned char code Bmpt4[]=
{
/*-- 调入了一幅图像：E:\新建文件夹\Wait_4.bmp --*/
/*-- 宽度x高度=19x30 --*/
/*-- 宽度不是8的倍数，现调整为：宽度x高度=19x32 --*/
0x00,0x00,0x00,0x60,0xA0,0xA0,0xA0,0xA0,0xA0,0xA0,0xA0,0xA0,0xE0,0xE0,0xE0,0xE0,
0x60,0x00,0x00,0x00,0x00,0x00,0x00,0x03,0x0C,0x10,0x20,0xC0,0x00,0x00,0xC0,0x20,
0x10,0x0C,0x03,0x00,0x00,0x00,0x00,0x00,0x00,0x80,0x70,0x6C,0x72,0x79,0x7E,0x7F,
0x7F,0x7E,0xF9,0xF2,0xEC,0xF0,0x80,0x00,0x00,0xC0,0xC0,0xC0,0xC1,0xC1,0xC1,0xC1,
0xC1,0xC1,0xC1,0xC1,0xC1,0xC1,0xC1,0xC1,0xC0,0xC0

};
unsigned char code Bmpt5[]=
{
/*-- 调入了一幅图像：E:\新建文件夹\Wait_5.bmp --*/
/*-- 宽度x高度=19x30 --*/
/*-- 宽度不是8的倍数，现调整为：宽度x高度=19x32 --*/
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x40,0x60,0xA0,0xA0,0xC0,0x80,
0x80,0x00,0x00,0x00,0x00,0x80,0xC0,0xC0,0x60,0xA0,0xA0,0xE0,0xBF,0x00,0x80,0x80,
0x81,0x86,0x44,0x4D,0x7E,0x3C,0x00,0x06,0x0B,0x1B,0x6F,0x4F,0xDF,0xFF,0xFF,0xC3,
0x3F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xC0,0xC0,0xC0,0xC0,0xC0,0xC0,0xC0,
0xC1,0xC1,0xC0,0xC0,0xC0,0xC0,0xC0,0xC0,0xC0,0xC0,0xC0,0xC0,0xC0,0xC0,0xC0,0xC0

};
unsigned char code Bmpt6[]=
```



```
/*-- 宋体12; 此字体下对应的点阵为：宽x高=16x16 --*/
0x00,0x40,0x42,0x42,0x42,0x42,0x42,0x42,0xFE,0x42,0x42,0x42,0x42,0x42,0x40,0x00,
0x00,0x80,0x40,0x20,0x10,0x08,0x06,0x01,0x02,0x04,0x08,0x10,0x30,0x60,0x20,0x00,

/*-- 文字： 天 --*/
/*-- 宋体12; 此字体下对应的点阵为：宽x高=16x16 --*/
0x00,0x40,0x42,0x42,0x42,0x42,0x42,0xFE,0x42,0x42,0x42,0x42,0x42,0x40,0x00,
0x00,0x80,0x40,0x20,0x10,0x08,0x06,0x01,0x02,0x04,0x08,0x10,0x30,0x60,0x20,0x00,

/*-- 文字： 向 --*/
/*-- 宋体12; 此字体下对应的点阵为：宽x高=16x16 --*/
0x00,0x00,0xFC,0x04,0x04,0xE6,0x25,0x24,0x24,0x24,0xE4,0x04,0x04,0xFC,0x00,0x00,
0x00,0x00,0x7F,0x00,0x00,0x0F,0x04,0x04,0x04,0x04,0x0F,0x20,0x40,0x3F,0x00,0x00,

/*-- 文字： 上 --*/
/*-- 宋体12; 此字体下对应的点阵为：宽x高=16x16 --*/
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFE,0x40,0x40,0x40,0x40,0x40,0x00,0x00,0x00,
0x00,0x40,0x40,0x40,0x40,0x40,0x40,0x7F,0x40,0x40,0x40,0x40,0x40,0x60,0x40,0x00
};

//-----
unsigned char code Bmp002[]=
{
/*-----
; 源文件 / 文字：超前科技有限公司
; 宽x高(像素)：120x15
; 字模格式/大小：单色点阵液晶字模，纵向取模，字节倒序/240字节
; 数据转换日期：04-2-27 14:10:02
-----*/
0xA0,0xA4,0xA4,0xFF,0xA4,0xA4,0x80,0xA2,0x92,0x8E,0xA2,0xA2,0x9F,0x82,0x80,0x88,
0x88,0xE8,0xA9,0xAA,0xAC,0xE8,0x88,0x8C,0xEA,0x89,0xE8,0x8C,0x88,0x80,0x94,0x94,
0x94,0xFE,0x93,0x92,0x80,0xA0,0xC4,0x98,0x80,0xFF,0x80,0x80,0x80,0x88,0x88,0x88,
0xFF,0xC8,0x80,0xC8,0xC8,0xC8,0xFF,0xC8,0xC8,0xCC,0x88,0x80,0x84,0x84,0xC4,0xA4,
0xF4,0xAC,0xA7,0xA4,0xA4,0xA4,0xF4,0xA4,0x86,0x84,0x80,0x80,0xFE,0x82,0xB2,0xCE,
0x80,0xFE,0xCA,0xCA,0xCA,0xCA,0xFF,0x82,0x80,0x80,0x80,0xC0,0xA0,0x90,0x88,0x86,
0xC0,0x80,0x87,0x98,0xA0,0xC0,0xC0,0xC0,0x80,0x90,0x92,0xD2,0xD2,0xD2,0xD2,0xD2,
0xD2,0x9A,0x92,0x82,0xFF,0x82,0x80,0x80,0x30,0x0F,0x04,0x0F,0x11,0x11,0x21,0x20,
0x2F,0x24,0x24,0x24,0x2F,0x20,0x00,0x00,0x00,0x3F,0x02,0x12,0x22,0x1F,0x00,0x00,
0x17,0x20,0x1F,0x00,0x00,0x00,0x04,0x03,0x00,0x3F,0x00,0x01,0x02,0x02,0x02,0x02,
0x02,0x3F,0x01,0x01,0x00,0x01,0x11,0x20,0x1F,0x00,0x20,0x20,0x11,0x0A,0x04,0x0A,
0x11,0x20,0x20,0x00,0x01,0x00,0x00,0x00,0x3F,0x02,0x02,0x02,0x02,0x22,0x3F,0x00,
0x00,0x00,0x00,0x00,0x3F,0x02,0x04,0x03,0x00,0x3F,0x10,0x13,0x04,0x0A,0x11,0x30,
0x10,0x00,0x00,0x00,0x10,0x38,0x14,0x13,0x10,0x10,0x12,0x14,0x18,0x30,0x00,0x00,
0x00,0x00,0x00,0x07,0x02,0x02,0x02,0x02,0x07,0x00,0x10,0x20,0x1F,0x00,0x00,0x00
};
```



```
0x28,0x28,0x24,0x23,0x24,0xE8,0xE0,0x00,0x00,0x00,0x80,0xC0,0x40,0x40,0x40,0x40,
0x40,0x40,0x60,0x60,0x40,0x40,0x48,0x54,0x43,0x54,0xC8,0x88,0x14,0x23,0x14,0x08,
0x00,0x80,0x80,0xC0,0x60,0x30,0x30,0x60,0xC0,0x80,0x00,0x00,0x00,0x00,0x00,0x00,
0x08,0x14,0xC3,0xD4,0x08,0x08,0x14,0x03,0xE4,0xE8,0x00,0x00,0x40,0x00,0x00,0x00,
0x00,0x40,0x80,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x42,0x73,0x7B,0x5E,0x77,0xE5,0xC4,0x65,0x25,0xB5,0xFD,0x6D,0xC5,0x87,0xC7,
0x74,0x34,0x84,0x40,0x07,0x07,0xE0,0xF4,0x94,0x04,0x04,0x04,0xFC,0xFC,0x84,0x84,
0x84,0x84,0x84,0x87,0x03,0x00,0x04,0x16,0x12,0x13,0x51,0x90,0x50,0x10,0x13,
0x57,0x94,0xD0,0x11,0x11,0x13,0xF2,0xF6,0x04,0x04,0x18,0x5E,0x47,0x41,0x41,0x41,
0x41,0x41,0xFF,0xFF,0x41,0x41,0x4D,0x51,0x4D,0x4D,0x51,0x0C,0xD2,0x61,0x80,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x80,0x88,0x88,0xF8,0xF8,0x88,0x88,0xF8,0xF8,
0x8C,0x8C,0x88,0x00,0x00,0x00,0x00,0x00,0x00,0x01,0x00,0x1E,0x1E,0x06,0x1E,0x5B,
0x9F,0xDE,0x1B,0xA9,0x2D,0x26,0xB3,0x3F,0x8F,0x0E,0x1C,0x10,0x10,0x00,0x30,0x3E,
0x0F,0x01,0x07,0x0E,0x08,0x18,0x1F,0x1F,0x30,0x20,0xA0,0xA0,0x20,0xA0,0x20,0x20,
0x80,0x40,0x80,0x20,0x20,0x20,0x20,0x20,0x20,0x30,0x10,0x10,0x18,0x09,0x0C,0x07,
0x13,0x10,0x3E,0x00,0xA0,0xA0,0x20,0xA0,0x20,0x20,0xA0,0x20,0xBF,0x3F,0x20,0x20,
0x20,0x20,0x20,0x20,0x00,0x01,0x02,0x1F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0xC0,0xC0,0x10,0xF0,0xE0,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x10,0x10,0x18,0x0F,0x07,0x00,0x00,0x1F,0x1F,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x21,0x52,0x8C,0x52,0x21,0x21,
0x52,0x8C,0x52,0x21,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x21,0x52,0x8C,0x52,0x21,0x21,0x52,0x8C,0x52,0x21,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x21,0x52,
0x8C,0x52,0x21,0x21,0x52,0x8C,0x52,0x21,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x08,0x0F,0x07,0x7F,0x7F,
0x40,0x40,0x40,0x78,0x7B,0x0F,0x0C,0x00
```

```
};
unsigned char code Bmp01[]=
{
///这仿真器真不错,用过都说好.
/*-- 调入了一幅图像:C:\My Documents\仿真.bmp --*/
/*-- 宽度x高度=122x32 --*/
0x00,0x00,0x80,0x90,0xB0,0xB0,0x00,0xC0,0xE8,0xE8,0xE0,0x20,0x20,0x00,0x00,0x00,
0x00,0x20,0x20,0xB8,0xF8,0x60,0xC0,0xE0,0x78,0xD8,0xC0,0x40,0x00,0x00,0x00,0x00,
0x80,0xC0,0xE0,0xB8,0x18,0x40,0xC8,0xD8,0x50,0x40,0x40,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0xD0,0xF0,0xFC,0xEC,0xE8,0x48,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x10,
0xB0,0xB8,0xF8,0xC8,0xB8,0xB8,0x38,0x58,0x48,0x00,0x00,0x00,0x00,0x00,0x00,0xC0,
0xE0,0x20,0xA0,0x80,0xF8,0xF8,0xC0,0xC0,0x00,0x00,0x00,0x00,0x80,0xE0,0x78,0x58,
0x60,0x60,0x78,0xF8,0xB0,0xD0,0xC0,0x00,0x00,0x00,0x00,0xC0,0xE0,0xB8,0xF8,0xC0,
0xC0,0xE0,0xFC,0xFC,0xC0,0x40,0x40,0x00,0x00,0x00,0x00,0x00,0x08,0x0F,0x0F,0x08,
0x1C,0x16,0x13,0x13,0x16,0x14,0x10,0x10,0x10,0x00,0x04,0x1C,0x1D,0x3F,0x3F,0x1B,
0x1A,0x0F,0x07,0x0C,0x18,0x10,0x10,0x00,0x00,0x01,0x01,0x00,0x0F,0x0F,0x0C,0x06,
0x33,0x39,0x1F,0x07,0x00,0x00,0x00,0x00,0x00,0x88,0x88,0x28,0x3F,0x1F,0x0E,0x0F,
0x3F,0x34,0x04,0x04,0x84,0x80,0x00,0x00,0x04,0x1C,0x3E,0x3B,0x19,0x08,0x39,0x3B,
```



```
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,  
};
```

```
unsigned char code Bmp04[]=
```

```
{  
/*-- 调入了一幅图像：C:\My Documents\My Pictures\ok.bmp --*/  
/*-- 宽度x高度=122x32 --*/  
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFF,0x81,0x81,0x81,0xC1,0xE1,0xE1,0xF1,  
0xF9,0x79,0x3D,0x1F,0x1F,0x1F,0x07,0x8B,0x4D,0x4F,0x27,0xA5,0xA7,0xAF,0xAF,0xA7,  
0xAB,0xAF,0x23,0x4F,0x41,0x47,0x2F,0xA7,0xA5,0xA7,0x2F,0x4F,0x87,0x07,0x83,0x73,  
0x09,0xEB,0x0B,0x33,0x93,0x55,0x53,0x53,0x91,0x23,0x15,0xD3,0x13,0x97,0x55,0x97,  
0x57,0x53,0x53,0x93,0x23,0xCF,0x05,0x05,0x03,0x09,0x0F,0x85,0x53,0x51,0x2D,0xA3,  
0xA1,0xA1,0xA7,0xA7,0xAF,0xA7,0x27,0x43,0x41,0x41,0x81,0x03,0x0F,0x1D,0x37,0x6D,  
0xD5,0xD7,0x9B,0xF5,0x39,0x73,0x77,0x25,0xDB,0x13,0x77,0x65,0x29,0xF9,0xFF,0x00,  
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,  
0x00,0x00,0xFF,0x5F,0x1F,0xEF,0x67,0x0B,0x0D,0x82,0x14,0x00,0x00,0xC2,0x20,0x9E,  
0x41,0x3C,0x02,0x41,0x41,0x23,0x23,0x8F,0x1F,0x3F,0xFF,0xFF,0xFF,0xFF,0xF3,0xEE,  
0xEF,0xFF,0x07,0xF7,0x33,0x98,0x57,0x50,0x9F,0x50,0x57,0x94,0x34,0xF0,0x13,0x54,  
0x94,0x14,0x93,0x58,0x18,0x33,0xB6,0xB1,0xB8,0x33,0x35,0x35,0xB5,0x31,0x2C,0xA3,  
0x20,0xC0,0xF0,0x0C,0xC3,0x38,0x06,0x01,0xE1,0x53,0x13,0xE7,0x07,0x0F,0xFF,0xFF,  
0x9F,0x0F,0x9F,0xFE,0xFC,0xF9,0x02,0xFC,0x00,0x00,0x00,0x00,0x01,0x00,0x05,0x34,  
0xC8,0xC1,0x43,0x87,0x36,0x4C,0x98,0x03,0xFF,0x00,0x00,0x00,0x00,0x00,0x00,0x00,  
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFF,0x80,0x61,0x38,  
0x0C,0x06,0x01,0x00,0x00,0x00,0x00,0x03,0x04,0x08,0xD3,0x34,0x84,0x48,0x28,0x10,  
0x31,0x52,0x91,0x90,0x73,0x37,0xCF,0x0F,0x67,0x93,0x09,0x04,0x02,0x03,0xC4,0x29,  
0xB2,0xA4,0xA8,0x24,0x72,0x79,0x7C,0xFF,0x3F,0xA0,0xAF,0x21,0x7C,0x6E,0x60,0x27,  
0xA8,0x28,0x28,0xA7,0x20,0xA6,0xA9,0xA8,0x2E,0xA1,0x3C,0xA7,0x27,0xC8,0x13,0x14,  
0x24,0x28,0xC8,0x11,0xF1,0xD8,0x5C,0x5E,0xDF,0xFF,0x0F,0xEF,0x27,0x13,0x09,0x04,  
0x02,0x01,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,0x07,0x0A,0x33,0x5E,  
0xE0,0x7E,0xFF,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,  
0x00,0x00,0x00,0x00,0x00,0x00,0x7F,0x40,0x40,0x40,0x40,0x40,0x40,0x40,0x40,  
0x40,0x40,0x40,0x5F,0x60,0x4A,0x5B,0x5A,0x42,0x42,0x5A,0x5A,0x42,0x42,0x5A,0x5A,  
0x4A,0x63,0x58,0x47,0x40,0x40,0x40,0x7F,0x41,0x5C,0x43,0x7A,0x60,0x4E,0x51,0x51,  
0x4E,0x60,0x6E,0x61,0x60,0x4E,0x55,0x55,0x53,0x40,0x67,0x6C,0x63,0x70,0x67,0x6A,  
0x6A,0x69,0x60,0x6F,0x61,0x5C,0x42,0x41,0x40,0x5C,0x62,0x49,0x5D,0x5C,0x4B,0x43,  
0x5A,0x5A,0x5B,0x41,0x5C,0x5D,0x49,0x62,0x5C,0x40,0x40,0x40,0x40,0x40,0x40,0x40,  
0x40,0x40,0x40,0x40,0x40,0x40,0x40,0x40,0x40,0x40,0x43,0x7F,0x00,0x00,0x00,  
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00
```

```
};  
  
unsigned char code Bmp05[]=
```

```
{  
/*-- 调入了一幅图像：C:\My Documents\My Pictures\Tom_1048488042.bmp --*/  
/*-- 宽度x高度=101x29 --*/  
/*-- 宽度不是8的倍数，现调整为：宽度x高度=101x32 --*/  
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x80,0x80,0x40,0x40,  
0x20,0x20,0x20,0x20,0x20,0x10,0x10,0x10,0x10,0x10,0x10,0x10,0x10,0x10,0x10,0x10,
```



```
0x06,0x06,0x06,0x06,0x06,0x06,0x06,0x06,0xF7,0x04,0x04,0x04,0xF7,0x06,0x06,0x06,0x06,  
0x06,0x06,0x06,0xF7,0x04,0x04,0x04,0xF7,0x06,0x06,0x06,0x06,0x06,0x06,0x06,0xF7,  
0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x04,  
0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x04,  
0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x04,  
0x04,0x04,0x04,0x04,0x01,0x01,0x01,0x01,0x0F,0x03,0x0F,0x01,0x03,0x09,0x0F,0x01,  
0x01,0x01,0x0F,0x01,0x01,0x0F,0x01,0x01,0x09,0x0F,0x01,0x01,0x09,0x17,0x01,0x0F,  
0x09,0x09,0x0B,0x09,0x0F,0x01,0x07,0x01,0x01,0x07,0x05,0x05,0x05,0x05,0x05,0x05,  
0x05,0x07,0x01,0x01,0x01,0x07,0x05,0x05,0x05,0x05,0x05,0x05,0x05,0x07,0x01,0x01,  
0x01,0x07,0x05,0x05,0x05,0x05,0x05,0x05,0x05,0x07,0x01,0x01,0x01,0x01,0x01,0x01,  
0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,  
0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,  
0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,  
};
```