

OCMJ4X8B-3 中文液晶显示模块 使用说明书

感谢您关注和使用我们的中文系列液晶显示器产品，欢迎您提出意见和建议，我们将竭诚为您服务、让您满意。您可以浏览 <http://www.shsixian.com> 了解最新的产品与应用信息，或拨打热线电话 **021-53083613** 及向 sx@shsixian.com 邮箱发 E-mail 获取具体的技术咨询与服务。

上海思先电子有限公司

Shanghai Sixian Electronics Co; Ltd.

目录

| | |
|-------------------|----|
| 一、产品简介 | 3 |
| 二、模块外型尺寸图..... | 3 |
| 三、引脚描述 | 4 |
| 四、硬件接口 | 4 |
| 五、电性能参数 | 5 |
| 六、用户命令 | 5 |
| 七、外部接口连接 | 8 |
| 八、其他说明 | 8 |
| 九、应用程序示例 | 9 |
| 十、显示窗口坐标关系..... | 14 |
| 十一、ASCII 码表 | 14 |

一、产品简介

B 系列（改进型）中文液晶显示模块采用本公司研发的新型控制器 8133，该新型控制器是在原 8130 控制器的基础上进行改进，与原 8130 控制器的指令系统完全兼容，并增加了一些新的功能，原有用户可在原硬件接口不变的情况下，对软件程序稍做改动就可实现以下新增的功能（不做更改可以照常使用）：

- (1) 反白功能：写入反白功能指令后，以后写入的图形或字符、汉字的显示会出现反显效果（指显示内容与输入显示数据为取反关系。例：输入显示数据为 00001111：显示为 11110000）
- (2) 光标功能：写入光标功能指令后，光标会在下一个输入的字符或汉字的下一地址位显示（不在最后显示的字符或汉字下方）。光标可设置闪烁效果，并可设置光标长度（8 点或 16 点）。
- (3) 显示移动功能：写入显示移动功能指令后：整屏显示可按设定移动参数移位：移动速度可设置为 1、2、8、16 个点行/列移动。
- (4) 汉字字模为 16×16 点阵，字体改为 15×15 点阵，内置字库增加为 GB2312 国标一、二级简体汉字字库及前 16 区字符（16×16 点阵）

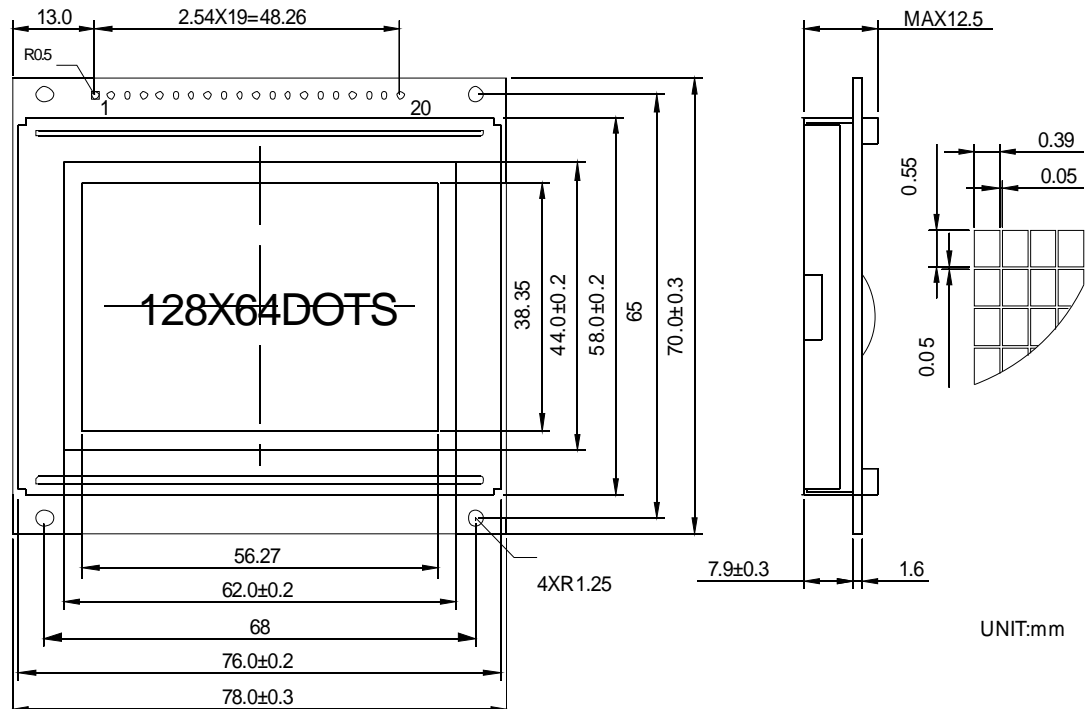
B 系列（改进型）中文液晶显示模块内含 GB2312 15×15 点阵国标一、二级简体汉字和 8×8 点阵及 8×16 点阵 ASCII 字符，用户输入 GB2312 区位码或 ASCII 码即可实现文本显示。

B 系列（改进型）中文液晶显示模块也可实现一般的点阵图形液晶显示模块功能。提供有位点阵和字节点阵两种图形液晶显示模块的功能。

本系列模块具有上/下/左/右整屏移动显示屏幕及整屏清除屏幕、光标显示、反白等操作命令。模块本身自带上电低电平复位的阻容复位回路，上电复位后可自动进行初始化设置，同时在接口提供一复位引脚，可提供用户进行软件复位控制或硬件复位控制。简单的 13 个用户接口命令代码，非常容易记忆。标准用户硬件接口采用 REQ/BUSY 握手通讯协议，简单可靠。

本系列模块工作温度：-10—55℃（常温）/-20+70℃（宽温）；存储温度：-20+65℃（常温型）

二、模块外型尺寸图



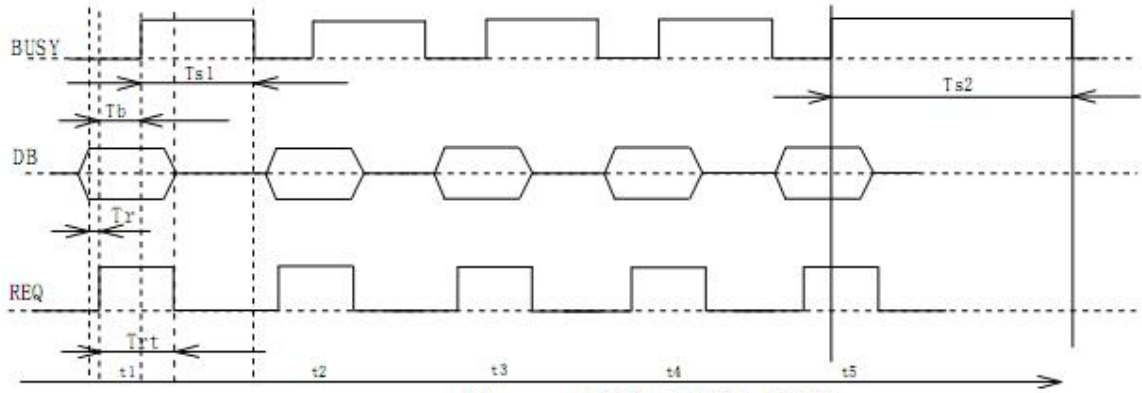
UNIT:mm

三、引脚描述

| 引脚 | 名称 | 方向 | 说明 |
|----|------|----|-------------------------------------|
| 1 | LED- | I | 背光电源负 (-0V) |
| 2 | LED+ | I | 背光电源正 (+5V) |
| 3 | VSS | I | 电源地(-0V) |
| 4 | VDD | I | 电源正 (+5V) |
| 5 | DB0 | I | 数据 0 |
| 6 | DB1 | I | 数据 1 |
| 7 | DB2 | I | 数据 2 |
| 8 | DB3 | I | 数据 3 |
| 9 | DB4 | I | 数据 4 |
| 10 | DB5 | I | 数据 5 |
| 11 | DB6 | I | 数据 6 |
| 12 | DB7 | I | 数据 7 |
| 13 | BUSY | I | 应答信号=1:已收到数据并正在处理中 =0:模块空闲,可接收数据 |
| 14 | REQ | O | 请求信号,高电平有效 |
| 15 | /RST | I | 复位信号,低电平有效. |
| 16 | FG | | 悬空 |
| 17 | VEE | | LCD 驱动负压输出端 |
| 18 | VO | | LCD 驱动电压输入端 |
| 19 | LED+ | I | 背光电源正(+5V) |
| 20 | LED- | I | 背光电源负(-0V) |

四、硬件接口

接口协议为请求/应答 (REQ/BUSY) 握手方式。应答 BUSY 高电平 (BUSY=1) 表示 OCMJ 忙于内部处理, 不能接收用户命令; BUSY 低电平 (BUSY=0) 表示 OCMJ 空闲, 等待接收用户命令。发送命令到 OCMJ 可在 BUSY=0 后的任意时该开始, 先把用户命令的当前字节放到数据线上, 接着发高电平 REQ 信号 (REQ=1) 通知 OCMJ 请求处理当前数据线上的命令或数据。OCMJ 模块在收到 REQ 高电平信号后立即读取数据线上的命令或数据, 同时将应答线 BUSY 变为高电平, 表明模块已收到数据并正在忙于对此数据的内部处理, 此时, 用户对模块的写操作已经完成, 用户可以撤消数据线上的信号并可作模块显示以外的其他工作, 也可不断地查询应答线 BUSY 是否为低 (BUSY=0), 如果 BUSY=0, 表明模块对用户的写操作已经执行完毕。可以再送下一个数据。如向模块发出一个完整的显示汉字的命令, 包括坐标及汉字代码在内共需 5 个字节, 模块在接收到最后一个字节后才开始执行整个命令的内部操作, 因此, 最后一个字节的应答 BUSY 高电平 (BUSY=1) 持续时间较长, 具体的时序图参见图一, 对应的具体时间参数说明参见表一4。



图—1：对模块写汉字时序图

五、电性能参数

1) 表—4：模块时间参数表

| 编号 | 名称 | 单位 | 值 | | 说明 |
|----|-----|----|-----|---------|-------------|
| | | | 最小值 | 最大值 | |
| 1 | Tr | uS | 0.4 | - | 数据线上数据稳定时间 |
| 2 | Tb | uS | 2 | 20 | 最大模块响应时间 |
| 3 | Trt | uS | 11 | - | 最小 REQ 保持时间 |
| 4 | Ts1 | uS | 20 | 45 | 最大数据接收时间 |
| 5 | Ts2 | mS | -- | 0.1~30* | 最大命令指令处理时间 |

*：不同命令所占用的时间各不相同，具体时间由命令表中给出

2) 表—5：模块主要电气参数表

| 编号 | 名称 | 单位 | 值 | | | |
|----|------------|----|-----|-----|------|------------|
| | | | 2X8 | 4X8 | 5X10 | |
| 1 | 电源工作电压 | V | 5 | 5 | 5 | |
| 2 | 电源电流 | mA | 2 | 3 | 5 | |
| 3 | 输入引脚电压 | V | 5 | 5 | 5 | |
| 4 | 最大输入电压建立时间 | uS | 1 | 1 | 1 | |
| 5 | 最大复位电压持续时间 | uS | 6 | 6 | 6 | RES 端低电平时间 |
| 6 | 复位内部处理时间 | mS | 15 | 15 | 15 | 此时模块不接收命令 |
| 7 | 背光电压 | V | 5 | 5 | 5 | |
| 8 | 标称背光电流 | mA | 120 | 180 | 250 | |
| 9 | 外接灰度调节电位器 | Ω | 10K | 10K | 10K | |

六、用户命令

用户通过命令调用 OCMJ 系列液晶显示器的各种功能。命令分为操作码及操作数两部分，操作数为十六进制。共分为 3 类 13 条。分别是：

一)、字符显示命令：1、显示国标汉字；2、显示 8×8 ASCII 字符；3、显示 8×16 ASCII 字符；

二)、图形显示命令：4、显示位点阵；5、显示字节点阵

三)、屏幕控制命令：6、清屏；7、上移；8、下移；

9、左移；10、右移；11、反白

12、光标显示 13、移动速度调整

(注意：以下所示取值范围分别为：2X8、4X8、4X12、5X10、8X10、8X15 的取值范围)

1) 显示国标汉字

命令格式：F0 XX YY QQ WW

该命令为 5 字节命令 (最大执行时间为 1.2 毫秒, $T_{s2}=1.2mS$) ,其中

XX: 为以汉字为单位的屏幕行坐标值, 取值范围 00H 到 70H、02H 到 09H、00H 到 0BH、00H 到 09H、00H 到 09H、00H 到 0EH

YY: 为以汉字为单位的屏幕列坐标值, 取值范围 00H 到 01H、00H 到 03H、00H 到 03H、00H 到 04H、00H 到 07H、00H 到 07H

QQ WW: 坐标位置上要显示的 GB2321 汉字区码, 位码

2) 显示 8×8 ASCII 字符

命令格式：F1 XX YY AS

该命令为 4 字节命令 (最大执行时间为 0.8 毫秒, $T_{s2}=0.8mS$) ,其中

XX: 为以 ASCII 码为单位的屏幕行坐标值, 取值范围 00H 到 0FH、04H 到 13H、00H 到 17H、00H 到 13H、00H 到 13H、00H 到 1DH

YY: 为以 ASCII 码为单位的屏幕列坐标值, 取值范围 00H 到 1FH、00H 到 3FH、00H 到 3FH、00H 到 4FH、00H 到 7FH、00H 到 7FH

AS: 坐标位置上要显示的 ASCII 字符码

3) 显示 8×16 ASCII 字符

命令格式：F9 XX YY AS

该命令为 4 字节命令 (最大执行时间为 1.0 毫秒, $T_{s2}=1.0mS$) ,其中

XX: 为以 ASCII 码为单位的屏幕列坐标值, 取值范围 00H 到 0FH、04H 到 13H、00H 到 17H、00H 到 13H、00H 到 13H、00H 到 7FH

AS: 坐标位置上要显示的 ASCII 字符码

4) 显示位点阵

命令格式：F2 XX YY

该命令为 3 字节命令 (最大执行时间为 0.1 毫秒, $T_{s2}=0.1mS$) ,其中

XX: 为以 1*1 点阵为单位的屏幕行坐标值, 取值范围 00H 到 7FH、20H 到 9FH、00H 到 0BFH、00H 到 9FH、00H 到 09FH、00H 到 0EFH

YY: 为以 1*1 点阵为单位的屏幕列坐标值, 取值范围 00H 到 1FH、00H 到 3FH、00H 到 4FH、00H 到 7FH、00H 到 7FH

5) 显示字节点阵

命令格式：F3 XX YY BT

该命令为 4 字节命令 (最大执行时间为 0.1 毫秒, $T_{s2}=0.1mS$) ,其中

XX: 为以 1*8 点阵为单位的屏幕行坐标值, 取值范围 00H 到 1FH、04H 到 13H、00H 到 17H、00H 到 13H、00H 到 13H、00H 到 1DH

YY: 为以 1*1 点阵为单位的屏幕列坐标值, 取值范围 00H 到 1FH、00H 到 3FH、00H 到 3FH、00H 到 4FH、00H 到 7FH、00H 到 7FH

BT: 字节像素值, 0 显示白点, 1 显示黑点 (显示字节为横向)

6) 清屏

命令格式：F4

该命令为单字节命令 (最大执行时间为 11 毫秒, $T_{s2}=11mS$) ,其功能为将屏幕向上移动一个点阵行。

7) 上移

命令格式：F5

该命令为单字节命令（最大执行时间为 25 毫秒， $T_{s2}=25mS$ ），其功能为将屏幕向上移动一个点阵行。

8) 下移

命令格式：F6

该命令为单字节命令（最大执行时间为 30 毫秒， $T_{s2}=30mS$ ），其功能为将屏幕向下移动一点阵行。

9) 左移

命令格式：F7

该命令为单字节命令（最大执行时间为 12 毫秒， $T_{s2}=mS$ ），其功能为屏幕向左移动一个点阵列。

10) 右移

命令格式：F8

该命令为单字节命令（最大执行时间为 12 毫秒， $T_{s2}=12mS$ ），其功能为将屏幕向右移动一个点阵列。

11) 反白（新增指令）

该命令为单字节命令（最大执行时间为 0.1 毫秒， $T_{s2}=mS$ ），当写入指令#FAH：以后写入的图形或字符、汉字的显示会出现反显效果，就是把本应显示的数据取后再显示。例如：写入字节：“00010011b”：显示的是：“11101100b”。当再次写入指令#FAH：取消反白功能，恢复原来显示效果。

12) 光标显示（新增指令）

命令格式：FBH+00H/07H/0FH

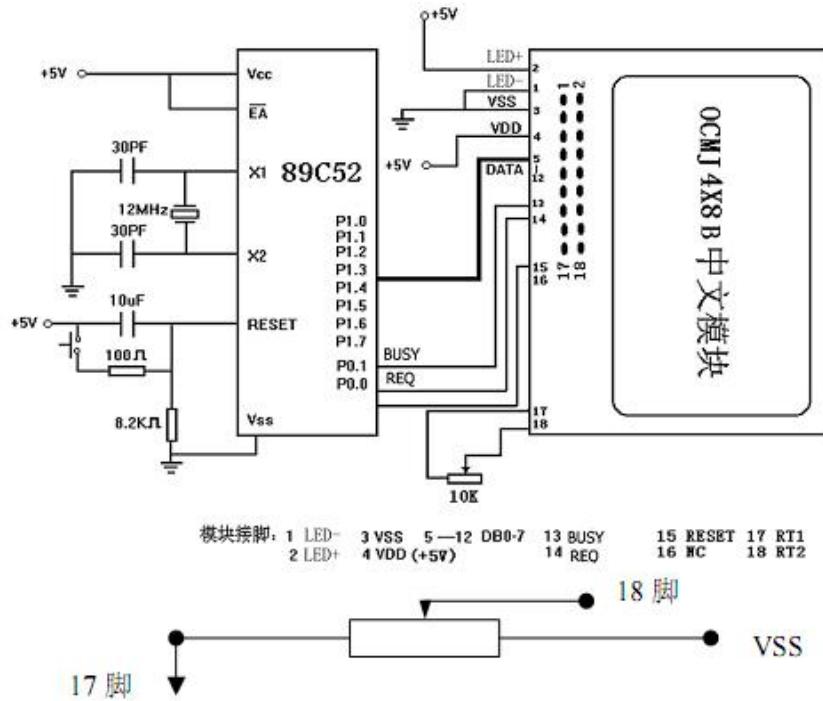
该命令为双字节命令（最大执行时间为 0.11 毫秒， $T_{s2}=0.1mS$ ），该指令含有一个参数，参数=00H 时关光标显示；参数=07H 时显示光标为 8 点（一个字节）长度；参数=0FH 时显示光标为 16 点（两个字节）长度；光标的移动是按照参数确定跳到下一地址显示，不在最后显示的字符或汉字下方。光标具有闪烁效果。

13) 调整移动速度（新增指令）

命令格式：FCH+00H/01H/07H/0FH

该指令含有一个参数，参数“00H”表示移动时按 1 个点行/列移动，这为默认值，一般不设置上电后以此为参数；参数“01H”表示移动时按 2 个点行/列移动；参数“07H”表示移动时按 8 个点行/列移动；参数“0FH”表示移动时按 16 个点行/列移动。

七、外部接口连接



八、其他说明

1) 复位

OCMJ 中文模块所有的设置初始化工作都是在上电时自动完成的，因此，在大多数情况下，复位端可经一电阻（10K）接在电源上或悬空处理。在确实需要复位操作的应用中，将该线拉低（RES=0）并保持10 微秒即可使模块复位。正常的复位功能包括清屏在内，占用时间不大于15毫秒，用户在此期间应禁止对模块进行操作，以免数据丢失，复位后的操作应在确保BUSY=0之后开始。

2) 背光

模块电源VDD与LED 背光电源最好取两组电源分开供电，以免背光源功耗相对大而影响模块显示。

另外，

LED+/LED-为背光源引脚，在模块背面，PCB板上的电路连接线途经两焊盘（断开），是空开两个贴片电阻位置，由用户接上相应的电阻调整LED 背光亮度，电阻阻值在10 欧-30 欧之间。该电阻不可短路，以免烧坏背光源或PCB板过热而影响模块IC。

3) 对比度调节

该系列模块有两种调整方式：1、OCMJ2X8A 和OCMJ4X8B 可在引脚RT1、RT2 之间接电位器（10K）调整，一般模块在生产时已经调整好固定电阻在PCB上（R6），用户不用外接电位器调整，如有需要可以先把R6 电阻去掉再接电位器调整显示对比度。2、OCMJ5X10B、OCMJ4X12B、OCMJ8X10B

和OCMJ8X15B需要外接电位器（10K-20K）调整显示对比度，电位器一端接地（VSS），一端接17脚（VEE/RT1），可调端接18脚（V0/RT2），调整电位器可以调整显示对比度。

九、典型应用程序示例 -----用 8051 为 MPU

```
//OCM4x8B 参考程序
```

```
#include <reg52.h>
```

```
#include <string.h>
```

```
#include <stdlib.h>
```

```
#include <stdio.h>
```

```
#define data_ora    P1                //MCU P1<-----> LCM
```

```
#define uchar      unsigned char
```

```
#define uint       unsigned int
```

```
sbit req =P3^0; //请求信号,H 有效
```

```
sbit busy=P3^1; //H:已收到数据并在处理中,L:空闲可接收数据
```

```
void send (uchar dat);
```

```
uchar code tab2[]={
```

```
0x24,0x03,0x1f,0x49,0x24,0x3a,0x36,0x23,0x19,0x5a,0x11,0x4a,0x1c,0x52,0x2c,0x45,
0x36,0x30,0x2e,0x24,0x32,0x1b,0x1c,0x16,0x37,0x36,0x1f,0x42,0x32,0x34,0x1c,0x10,
0x19,0x06,0x24,0x3c,0x17,0x41,0x18,0x1b,0x15,0x24,0x33,0x23,0x1b,0x07,0x36,0x18,
0x21,0x4e,0x2a,0x19,0x33,0x23,0x18,0x5c,0x17,0x1d,0x11,0x43,0x21,0x49,0x1b,0x4e};
```

```
uchar code tab4[]={
```

```
"Golden Palm TECH"
```

```
"TEL:0758-2263858"
```

```
" No.31 Jianshe "
```

```
"Road3 Zhaoqing "
```

```
"P.R.China      "
```

```
"Postcode: 526040"};
```

```
uchar code tab6[]={
```

```
/*-- 调入了一幅图像： F:\梁\画图\COCK.bmp  --*/
```

```
/*-- 宽度 x 高度=128x64  --*/
```

```
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xE0,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x07,
0xC0,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x03,
0x80,0x00,0x00,0x00,0x00,0x00,0x20,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,
0x80,0x00,0x00,0x00,0x20,0x81,0x00,0x60,0x00,0x00,0x00,0x00,0x00,0x00,0x01,
0x80,0x00,0x00,0x00,0x61,0x83,0x00,0xE0,0x00,0x00,0x0F,0xFF,0xE0,0x00,0x00,0x01,
0x80,0x00,0x00,0x00,0xE3,0x87,0x03,0xE0,0x00,0x00,0xFF,0xFF,0xFF,0x00,0x00,0x01,
```

0x80,0x00,0x00,0x01,0xC7,0x8F,0x07,0xC0,0x00,0x1F,0xFF,0xFF,0xFF,0xF0,0x00,0x01,
 0x80,0x00,0x00,0x23,0xCF,0x9E,0x0F,0x80,0x00,0xFF,0xFF,0xFF,0xFF,0xFE,0x00,0x01,
 0x80,0x00,0x00,0x3B,0xFF,0xFF,0xFE,0x00,0x03,0xFF,0xFC,0x00,0x0F,0xFF,0x00,0x01,
 0x80,0x00,0x00,0x3F,0xFF,0xFF,0xF8,0x00,0x0F,0xFF,0x80,0x00,0x00,0xFF,0x80,0x01,
 0x80,0x00,0x00,0x3F,0xFF,0xFF,0xF0,0x00,0x3F,0xFF,0xFF,0xFF,0x00,0x0F,0xC0,0x01,
 0x80,0x00,0x00,0x7C,0xFF,0xFF,0xC0,0x00,0xFF,0xFF,0xFF,0xFF,0xF0,0x07,0xC0,0x01,
 0x80,0x00,0x00,0xFC,0xFF,0xFF,0x00,0x01,0xFF,0xFF,0xFF,0xFF,0xFE,0x01,0xE0,0x01,
 0x80,0x00,0x01,0xFF,0xFF,0xF0,0x00,0x03,0xFF,0xFF,0xF0,0x00,0xFF,0x80,0xF0,0x01,
 0x80,0x00,0x01,0xFF,0xFF,0x80,0x00,0x07,0xFF,0xFF,0xFF,0xE0,0x0F,0xE0,0x70,0x01,
 0x80,0x00,0x01,0x87,0xFF,0xE0,0x00,0x0F,0xFF,0xFF,0xFF,0xFE,0x01,0xF0,0x30,0x01,
 0x80,0x00,0x01,0x0F,0xFF,0xF0,0x00,0x1F,0xFF,0xFF,0xFF,0xFF,0xC0,0xF8,0x30,0x01,
 0x80,0x00,0x00,0x7F,0xFF,0xF8,0x00,0x3F,0xFF,0xFF,0xFF,0xFF,0xF0,0x78,0x30,0x01,
 0x80,0x00,0x01,0xFF,0xFF,0xFC,0x00,0x3F,0xFF,0xFF,0xFF,0xFF,0xF0,0x1C,0x30,0x01,
 0x80,0x00,0x07,0xFF,0xFF,0xFE,0x00,0x7F,0xFF,0xFF,0xFF,0xFF,0xFC,0x1C,0x30,0x01,
 0x80,0x00,0x1F,0xFF,0xFF,0xFE,0x00,0x7F,0xFF,0xFF,0xFF,0xFF,0xFE,0x0C,0x30,0x01,
 0x80,0x00,0x3F,0xFF,0xFF,0xFE,0x00,0xFF,0xFF,0xFF,0xFF,0xFF,0xFE,0x0C,0x30,0x01,
 0x80,0x00,0x7F,0xFF,0xFF,0xFE,0x01,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x0C,0x30,0x01,
 0x80,0x00,0xFF,0xFF,0xFF,0xFE,0x01,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x18,0x20,0x01,
 0x80,0x00,0xFF,0xFF,0xFF,0xFE,0x03,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x10,0x40,0x01,
 0x80,0x00,0xFF,0xFF,0xFF,0xFF,0x07,0xFF,0xFF,0xFF,0xFF,0xFE,0x3F,0x00,0x80,0x01,
 0x80,0x00,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x9F,0x00,0x00,0x01,
 0x80,0x00,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xCF,0x00,0x00,0x01,
 0x80,0x00,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFE,0xFF,0xEF,0x00,0x00,0x01,
 0x80,0x00,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x7F,0xE7,0x00,0x00,0x01,
 0x80,0x00,0x7F,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xBF,0xE6,0x00,0x00,0x01,
 0x80,0x00,0x7F,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xBF,0xE6,0x00,0x00,0x01,
 0x80,0x00,0x3F,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xBF,0xC6,0x00,0x00,0x01,
 0x80,0x00,0x1F,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x9F,0xC6,0x00,0x00,0x01,
 0x80,0x00,0x0F,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x7F,0x9F,0x0C,0x00,0x00,0x01,
 0x80,0x00,0x07,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x7F,0xFE,0x1E,0x0C,0x00,0x00,0x01,
 0x80,0x00,0x03,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFC,0x7F,0xFE,0x1E,0x0C,0x00,0x00,0x01,
 0x80,0x00,0x01,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFC,0x7F,0xFE,0x1C,0x0C,0x00,0x00,0x01,
 0x80,0x00,0x00,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xF0,0xFE,0xFC,0x18,0x0C,0x00,0x00,0x01,
 0x80,0x00,0x00,0x7F,0xFF,0xFF,0xFF,0xFF,0xFF,0xE0,0xF8,0xF8,0x10,0x08,0x00,0x00,0x01,
 0x80,0x00,0x00,0x3F,0xFF,0xFF,0xFF,0xFF,0xFF,0xE0,0xF9,0xF0,0x10,0x00,0x00,0x00,0x01,
 0x80,0x00,0x00,0x1F,0xFF,0xFF,0xFF,0xFF,0xFF,0x21,0xF1,0xC0,0x00,0x00,0x00,0x00,0x01,
 0x80,0x00,0x00,0x0F,0xFF,0xFF,0xFF,0xFE,0x21,0x81,0x00,0x00,0x00,0x00,0x00,0x01,
 0x80,0x00,0x00,0x07,0xFF,0xFF,0xFF,0xFE,0x11,0x01,0x00,0x00,0x00,0x00,0x00,0x01,
 0x80,0x00,0x00,0x01,0xFF,0xFE,0x7F,0xFE,0x01,0x00,0x00,0x00,0x00,0x00,0x00,0x01,
 0x80,0x00,0x00,0x00,0x7F,0xF8,0x3F,0xFE,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,
 0x80,0x00,0x00,0x00,0x1F,0xF0,0x0F,0xFC,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,
 0x80,0x00,0x00,0x00,0x07,0xE6,0x03,0xFC,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,
 0x80,0x00,0x00,0x00,0x03,0xE0,0x00,0x1C,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,
 0x80,0x00,0x00,0x00,0x01,0xC0,0x00,0x07,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,
 0x80,0x00,0x00,0x00,0x01,0xC0,0x00,0x64,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,

```

0x80,0x00,0x00,0x00,0x03,0xE0,0x00,0x3A,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,
0x80,0x00,0x00,0x00,0x06,0x20,0x00,0x20,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,
0x80,0x00,0x00,0x00,0x0C,0x20,0x00,0x40,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,
0x80,0x00,0x00,0x00,0x18,0x18,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,
0x80,0x00,0x00,0x00,0x31,0x80,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,
0x80,0x00,0x00,0x00,0xE1,0x90,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,
0x80,0x00,0x01,0xFF,0xC1,0x81,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,
0x80,0x00,0x03,0xFA,0x10,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,
0x80,0x00,0x00,0x04,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,
0xC0,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x03,
0xE0,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x07,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF
};
/*-----延时子程序-----*/
void delay (uint us)
{
    while(us--);
}
void delay1 (uint ms)
{
    uint i,j;
    for(i=0;i<ms;i++)
        for(j=0;j<1000;j++)
            ;
}
/*-----写命令或数据到 LCD-----*/
void wr_zb (uchar comm,uchar addr_x,uchar addr_y)
{
    send(comm);
    send(addr_x);
    send(addr_y);
}
void send (uchar dat) //写命令或数据
{
    while(busy==1)
        ;
    data_or=dat;
    req=1;
    while(busy==0)
        ;
    req=0;
}
/*-----显示点阵-----*/
void disp_lat (uchar data1,uchar data2)
{

```

```

uchar i,j,k;
for(k=0;k<64;k=k+4)
{
    for(j=0;j<2;j++)
    {
        for(i=4;i<20;i++)
        {
            wr_zb(0xf3,i,k+j);
            send(data1);
        }
    }
    for(j=2;j<4;j++)
    {
        for(i=4;i<20;i++)
        {
            wr_zb(0xf3,i,k+j);
            send(data2);
        }
    }
}
}
/*-----显示图形-----*/
void disp_img (uchar code *img)
{
    uchar i,j;
    for(j=0;j<64;j++)
    {
        for(i=4;i<20;i++)
        {
            wr_zb(0xf3,i,j);
            send(img[j*16+i-4]);
        }
    }
}
/*-----显示中文-----*/
void disp_chn (uchar code *chn)
{
    uchar i,j;
    for(j=0;j<4;j++)
    {
        for(i=2;i<10;i++)
        {
            wr_zb(0xf0,i,j);
            send(chn[j*16+(i-2)*2]);
            send(chn[j*16+(i-2)*2+1]);
        }
    }
}

```

```

    }
}
}
/*-----显示字符-----*/
void disp_eng (uchar code *eng)
{
    uchar i,j;
    for(j=0;j<2;j++)
    {
        for(i=4;i<20;i++)
        {
            wr_zb(0xf9,i,j*16);
            send(eng[j*16+i-4]);
        }
    }
    for(j=2;j<6;j++)
    {
        for(i=4;i<20;i++)
        {
            wr_zb(0xf1,i,(j+2)*8);
            send(eng[j*16+i-4]);
        }
    }
}
/*-----主程序-----*/
void main ()
{
    SP=0x5f;
    req=0;
    busy=1;
    while(1)
    {
        send(0xf4);
        disp_lat(0xff,0xff);
        delay1(800);
        send(0xf4);
        disp_chn(tab2);
        delay1(800);
        send(0xf4);
        disp_lat(0xff,0x00);
        delay1(800);
        send(0xf4);
        disp_eng(tab4);
        delay1(800);
        send(0xf4);
    }
}

```

```

disp_lat(0xcc,0xcc);
delay1(800);
send(0xf4);
disp_img(tab6);
delay1(800);
}
}

```

十、显示窗口坐标关系

OCMJ5X10B-2屏幕显示坐标关系表

| | | X=00H(L1-L16)..... | | | | X=09H | | | | |
|------------------------------|--------|--------------------|-----|-----|-----|-------|-----|-----|-----|-----|
| | | L0 | | | | L159 | | | | |
| Y=00 (Line0 --Line15) | Line0 | 1/0 | 1/0 | 1/0 | 1/0 | ... | 1/0 | 1/0 | 1/0 | 1/0 |
| | Line1 | 1/0 | 1/0 | 1/0 | 1/0 | ... | 1/0 | 1/0 | 1/0 | 1/0 |
| | Line2 | 1/0 | 1/0 | 1/0 | 1/0 | ... | 1/0 | 1/0 | 1/0 | 1/0 |
| | Line3 | 1/0 | 1/0 | 1/0 | 1/0 | ... | 1/0 | 1/0 | 1/0 | 1/0 |
| | | ... | | | | ... | | | | |
| Y=04 (Line64 --Line79) | Line76 | 1/0 | 1/0 | 1/0 | 1/0 | ... | 1/0 | 1/0 | 1/0 | 1/0 |
| | Line77 | 1/0 | 1/0 | 1/0 | 1/0 | ... | 1/0 | 1/0 | 1/0 | 1/0 |
| | Line78 | 1/0 | 1/0 | 1/0 | 1/0 | ... | 1/0 | 1/0 | 1/0 | 1/0 |
| | Line79 | 1/0 | 1/0 | 1/0 | 1/0 | ... | 1/0 | 1/0 | 1/0 | 1/0 |

注：ASC II 码坐标：X=00H(L1-L8).....X=13H

.以上列表为汉字、ASC II 码显示屏幕坐标（ASC II 码Y坐标以点阵坐标为准）

如显示图形点阵，则以128*64（OCMJ4X8）或128*32（OCMJ2X8）点阵坐标为准，可在屏幕任意位置显示。

十一、ASCII 码表

| 符号 | 十进制 | 十六进制 | 符号 | 十进制 | 十六进制 |
|----|-----|------|----|-----|------|
| A | 65 | 41H | a | 97 | 61H |
| B | 66 | 42H | b | 98 | 62H |
| C | 67 | 43H | c | 99 | 63H |
| D | 68 | 44H | d | 100 | 64H |
| E | 69 | 45H | e | 101 | 65H |
| F | 70 | 46H | f | 102 | 66H |

| | | | | | |
|-----|----|-----|-----|-----|-----|
| G | 71 | 47H | g | 103 | 67H |
| H | 72 | 48H | h | 104 | 68H |
| I | 73 | 49H | i | 105 | 69H |
| (J) | 74 | 4AH | (j) | 106 | 6AH |
| K | 75 | 4BH | k | 107 | 6BH |
| L | 76 | 4CH | l | 108 | 6CH |
| M | 77 | 4DH | m | 109 | 6DH |
| N | 78 | 4EH | n | 110 | 6EH |
| O | 79 | 4FH | o | 111 | 6FH |
| P | 80 | 50H | p | 112 | 70H |
| Q | 81 | 51H | q | 113 | 71H |
| R | 82 | 52H | r | 114 | 72H |
| S | 83 | 53H | s | 115 | 73H |
| T | 84 | 54H | t | 116 | 74H |
| U | 85 | 55H | u | 117 | 75H |
| V | 86 | 56H | v | 118 | 76H |
| X | 87 | 57H | x | 119 | 77H |
| Y | 88 | 58H | y | 120 | 78H |
| Z | 89 | 59H | z | 121 | 79H |