

# 产品使用说明书

产品类型：点阵型LCD液晶显示模组

产品型号：19264A-V6.0

客户：

客户编号：

日期：

确认（盖章）

制造商	客户

# 目 录

- 1 . 修订记录
- 2 . 概述
- 3 . 外形尺寸
- 4 . 硬件方框图
- 5 . 电气特性
- 6 . 接口说明
- 7 . 指令说明
- 8 . 操作时序说明
- 9 . 应用例程
- 10 . 注意事项

1. 修订记录

版本	发行日期	新制/修订内容
V6.0	2020-6-17	新制

注：升级版本向下兼容，不做另行通知，如遇兼容问题影响性能请联系本公司解决

## 2.概述

19264A是一款点阵型液晶显示模块，可用于显示文字和图形。8位并行数据传输方式，提供显示4096位RAM。

显示分辨率: 192X 64dots

显示颜色及背光颜色: STN 蓝,黄绿,灰; 背光 黑,白,黄绿

偏光膜:全透/半透

观察角度: 6:00

显示占空比: 1/64 驱动偏压: 1/9

显示数据 RAM :4096 bits\*3

尺寸 (Unit: mm)

外形尺寸: 127X62X (高框12.5; 矮框11.5)

可视区域 : 104X39

字符尺寸: 95.576X31.823

点尺寸:0.458X0.458

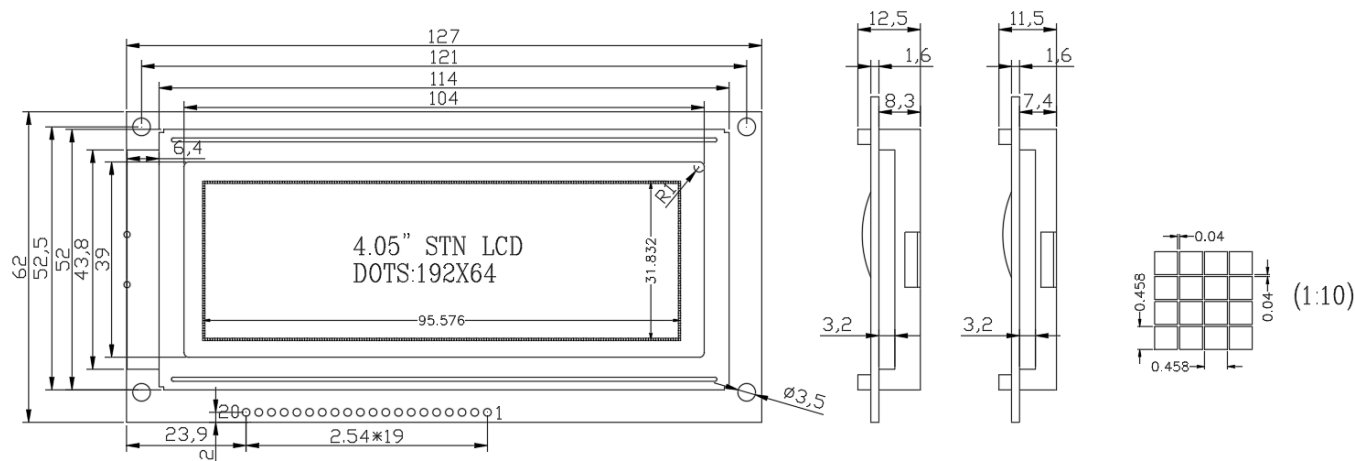
像素间距: 0.498X0.498

重量:           g

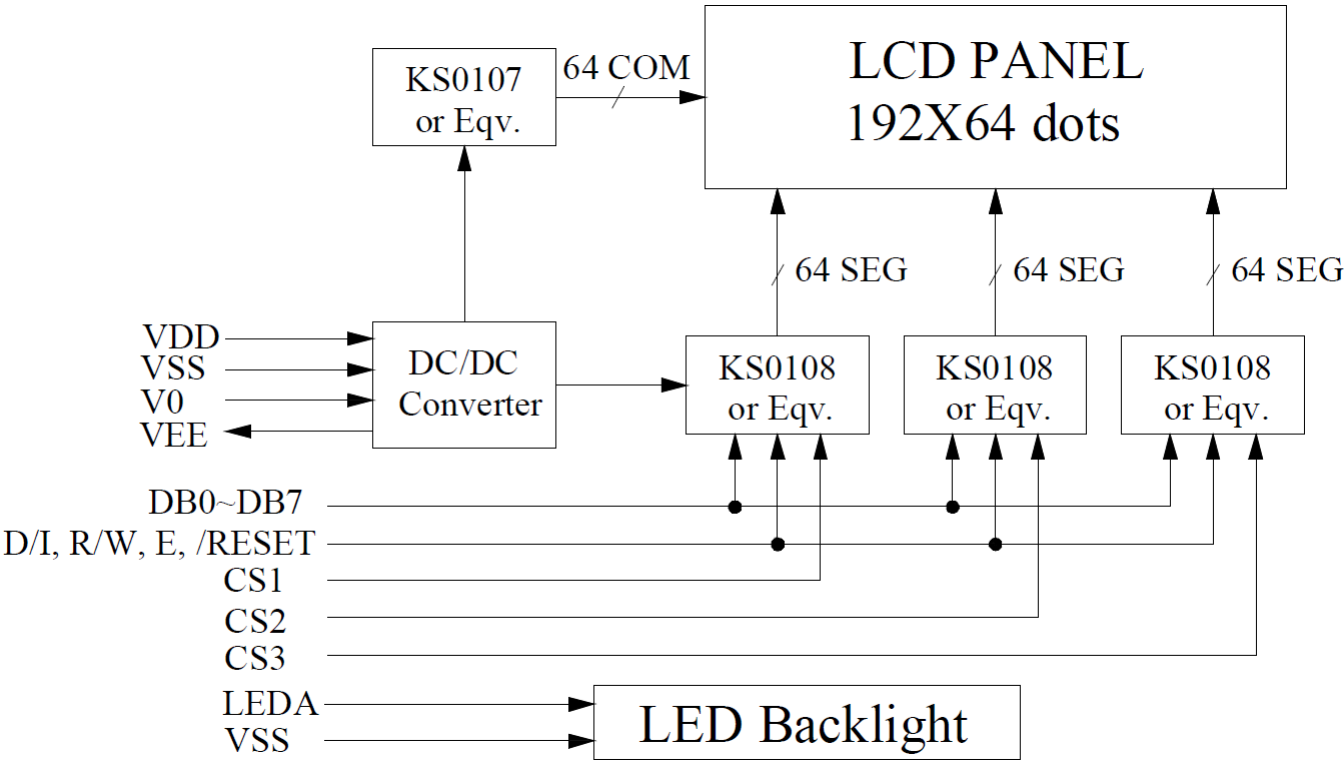
对比度: V0外部调节或内部固定对比度

工作电压: +3.3V或+5V 默认5V

3. 外形尺寸:



4. 硬件方框图:



## 5.电气特性

### 5.1 极限参数

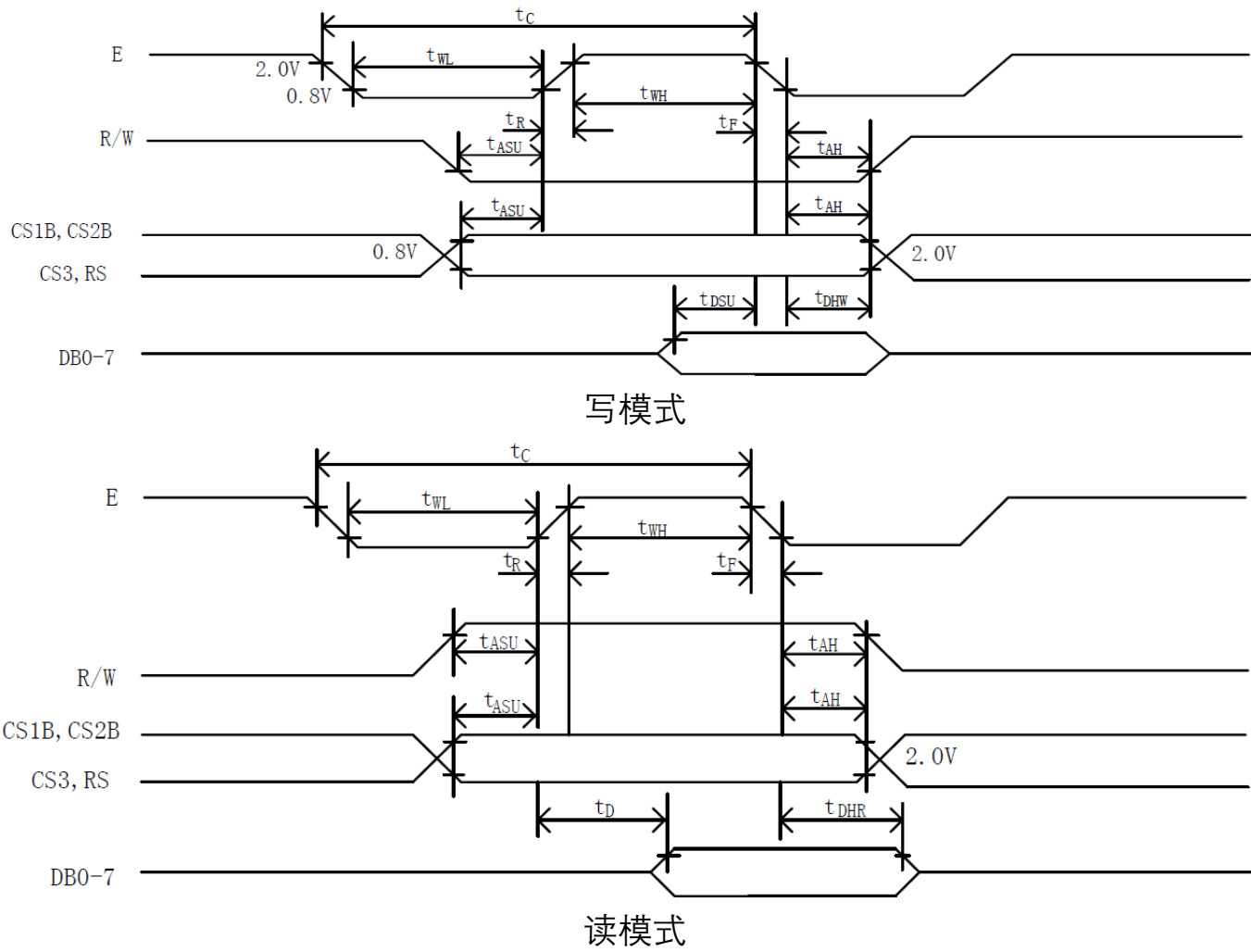
参数名称	符号	条件	典型值		单位
			最小值	最大值	
工作电压	Vdd		-0.3	+7.0	V
LCD驱动电压	Vlcd		Vee-0.3	Vdd+0.3	V
电源电压	Vee		Vdd-19	Vdd+0.3	V
工作温度(T)	Top	-	-20	70	℃
储存温度(T)	Tstg	-	-30	80	℃

### 5.2 直流参数1(Ta=25℃,Vdd=4.5V~5.5V,VDD-VEE=8~17V)

参数名称	符号	条件	标称值			单位
			最小	典型	最大	
电源电压	Vdd-GND	-	4.5	5.0	5.5	V
工作电流 (不包括背光)	Idd	Vdd=5V	12	16	18	mA
LCD驱动电压	Vdd-V5		3.0	-	Vdd	V
LED背光工作电流	If	Vf=2.8~3.0V	68	72	80	mA
LED背光功耗	Pd		306	360	440	mW
输入高电平	Vih		2	-	Vdd	V
输入低电平	Vil		-0.3	-	0.8	V
输出高电平	Voh	Ioh=-200uA	2.4	-	Vdd	V
输出低电平	Vol	Iol=1.6mA	0	-	0.4	V

### 5.3 交流参数1(Ta=25℃,Vdd=4.5V~5.5V)MPU接口

特 性	符 号	最 小	典 型	最 大	单 位
E 周期	t <sub>C</sub>	1000	—	—	ns
E 高电平宽度	t <sub>WH</sub>	450	—	—	ns
E 低电平宽度	t <sub>WL</sub>	450	—	—	ns
E 上升时间	t <sub>R</sub>	—	—	25	ns
E 下降时间	t <sub>F</sub>	—	—	25	ns
地址设置时间	t <sub>ASU</sub>	140	—	—	ns
地址保持时间	t <sub>AH</sub>	10	—	—	ns
数据设置时间	t <sub>DSU</sub>	200	—	—	ns
数据延迟时间	t <sub>D</sub>	—	—	320	ns
数据保持时间 (写)	t <sub>DHW</sub>	10	—	—	ns
数据保持时间 (读)	t <sub>DHR</sub>	20	—	—	ns



6.接口说明

脚号	符号	功能	备注
1	VSS	供电电源负极	
2	VDD	供电电源正极	
3	V0	LCD偏压供电	可调节对比度
4	RS	数据/指令选择(H:数据 L: 指令)	
5	R/W	读/写选择(H:读 L:写)	
6	E	使能信号	
7	D0	数据位 0	
8	D1	数据位1	
9	D2	数据位2	
10	D3	数据位3	
11	D4	数据位4	
12	D5	数据位5	
13	D6	数据位6	
14	D7	数据位7	
15	CS1	片选1（左64列）低电平有效	
16	RST	系统复位脚	
17	CS2	片选2（中64列）低电平有效	
18	CS3	片选3（右64列）低电平有效	
19	Vout	LCD负压输出	
20	LEDA	LED 背光正	

7.指令说明

I/O缓冲器

输入缓冲器控制着模块是否处于有效状态。只有当CS1、CS2或CS3为有效模式，否则数据的输入/输出和指令不会被执行，内部状态不变。RST的运行与CS1、CS2或CS3的状态无关。

输入寄存器

输入寄存器提供了一个模块与具有不同工作频率的MPU的接口。输入寄存器临时存储用来暂存写进显示RAM数据，当CS1、CS2或CS3有效，R/W和RS选择输入寄存器。从MPU来的数据被写入到输入寄存器，然后再写入到显示存储器中。数据在E信号的下降沿被锁存并由内部操作自动写入显示RAM。

输出寄存器

当CS1、CS2或CS3处于有效且R/W=RS=H时，输出寄存器暂存显示寄存器中的数据。当CS1、CS2或CS3处于有效且R/W=H，RS=L时，状态数据（忙检查）可以被读出。读显示RAM中的数据，需要两次读操作指令访问，第一次，显示数据RAM被锁存在输出寄存器中，第二次，MPU读出锁存的数据。当读忙标志时不需要两次。

通过设置RS/RW位的各种操作：

RS	RW	操作
L	L	写指令操作
L	H	读状态（忙检测）
H	L	写数据操作（从输入寄存器到显示数据存储器）
H	H	读数据操作（从显示数据存储器到输出寄存器）

复 位

系统可以被以下两种方式初始化：当上电时RST端保持低电平；接收MPU的指令；

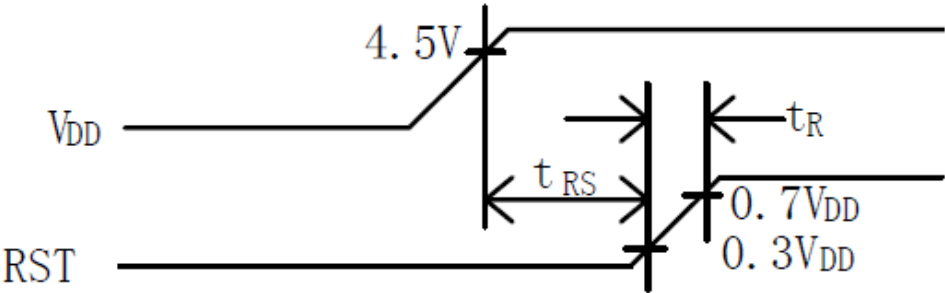
当RST端置低电平，执行下列步骤：

— 显示关

显示开始位置寄存器置0（Z计数器）

当RST为低时，除了读状态指令外其它的指令不被接收。因此，在确认DB4=0（清除RSTB）和DB7=0（准备好）后执行其它指令

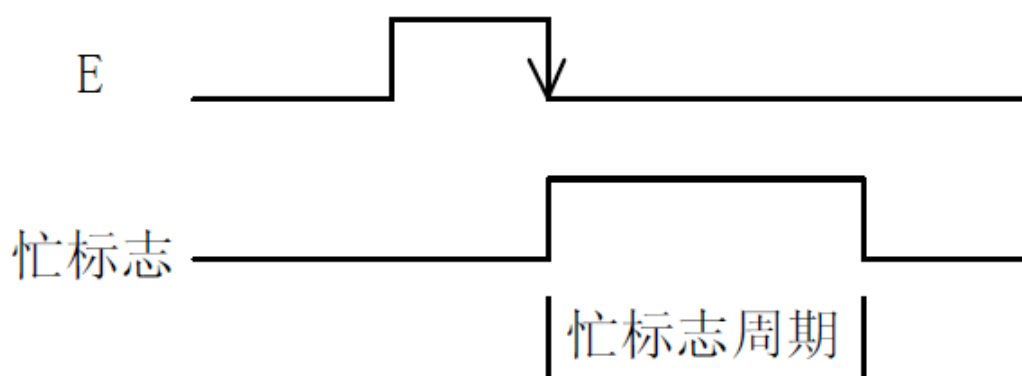
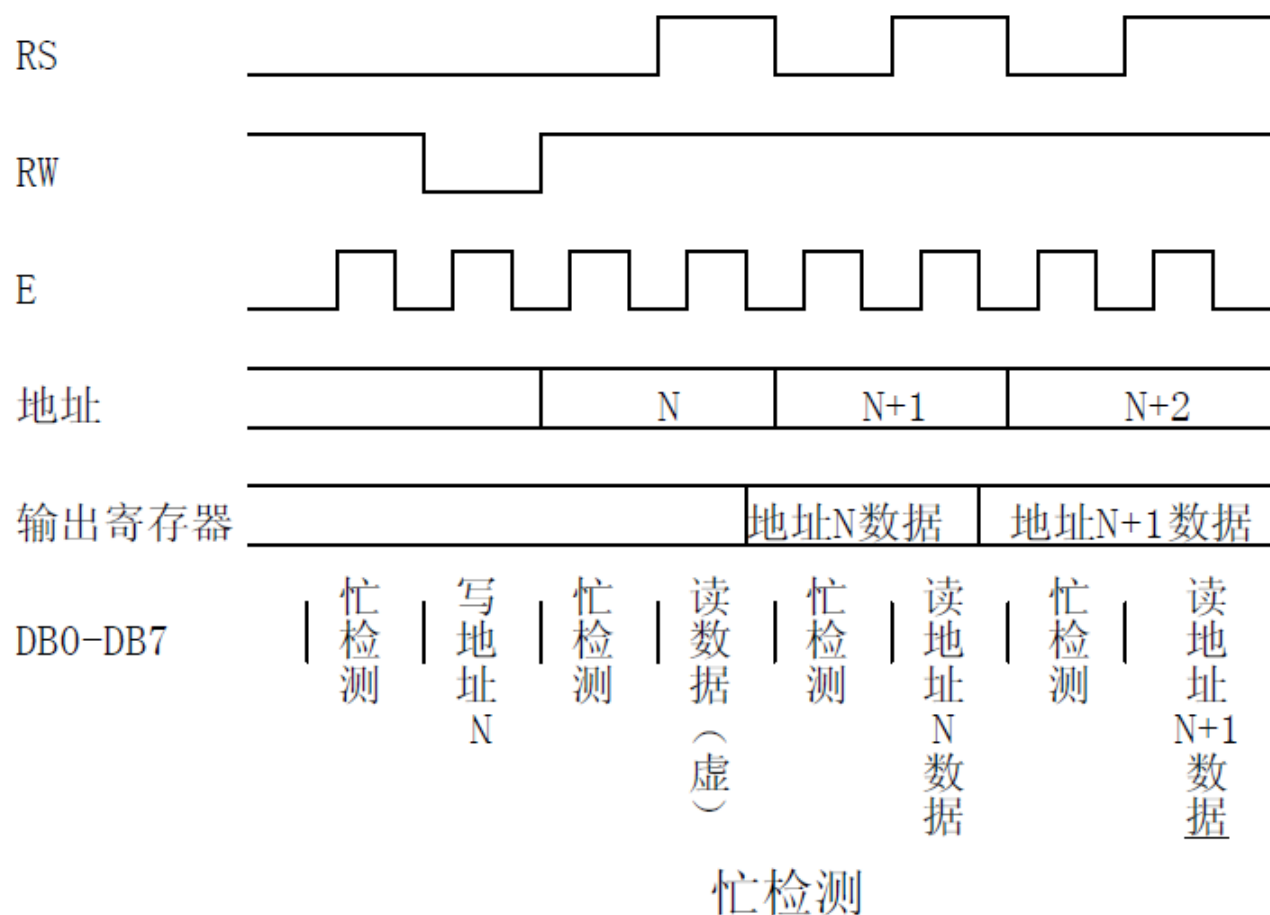
初始供电条件 如下表 参 数	符号	最小	典型	最大	单位
复位时间	tRS	1.0	—	—	us
上升时间	tR	—	—	200	ns



忙标志（BF）

忙标志表示模块的忙/闲状态，当忙标志为高时，表示模块正在执行内部操作，当忙标志为低时，模块可以接受外部命令或数据，DB7表示忙标志





$1/f_{clk} \leq T_{busy} \leq 3/f_{clk}$   
 $f_{clk}$  为 CLK1, CLK2 的频率  
 忙标志

7.1 指令描述

指令表

指令	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	描述
显示开/关	0	0	0	0	1	1	1	1	1	D	设置显示开关
设置Y地址	0	0	0	1	Y地址（0-63）					设置Y地址计数器中Y地址	
设置页X地址	0	0	1	0	1	1	1	页（0-7）		设置X地址计数器中X地址	
设置起始行Z地址	0	0	1	1	显示起始行（0~63）					设置显示起始行寄存器内容	
读状态	0	1	忙	0	开关	复位	0	0	0	0	忙：0空闲，1忙；开关：0开，1关 复位：0为正常，1为复位中
写数据	1	0	D7	D6	D5	D4	D3	D2	D1	D0	写数据到显示储存器（写指令后，Y地址自动加1）
读数据	1	1	D7	D6	D5	D4	D3	D2	D1	D0	从显示储存器中读取数据

注：“-”为不考虑

7.1.1 显示开关

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	1	1	1	1	1	D

显示数据在D=1时显示，在D=0时消失。尽管当D=0时显示数据不在屏幕上显示，该数据依然保存在存储器中，因此可以将D=0改变到D=1使其显示。

7.1.2 设置Y地址

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	1	AC5	AC4	AC3	AC2	AC1	AC0

显示数据存储器的Y地址（AC0-AC5）在Y计数器中设置。地址由指令设置并在对显示RAM读或写时自动增1。

7.1.3 设置页X地址

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	1	0	1	1	1	AC2	AC1	AC0

显示存储器的X地址（AC0-AC2）在X地址寄存器在X地址寄存器中设置。MPU中读/写操作在这一页面执行，直到下一个页被设置。

7.1.4 显示开始行Z地址

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	1	1	AC5	AC4	AC3	AC2	AC1	AC0

显示存储器的Z地址在显示开始行寄存器中被设置并显示在屏幕顶端。当显示占空比为1/64或其它（1/32-1/64），在LCD显示屏从显示开始指令指定的行开始显示。

7.1.5 读状态

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	1	忙	0	开关	复位	0	0	0	0

- 忙=1,芯片执行内部操作，不接受指令。
- 忙=0，芯片准备好接收指令。
- 开/关=1 显示开
- 开/关=0 显示关
- 复位=1 系统正在被初始化，在这个状态下，除状态读指令外，其余不接收
- 复位=0 系统初始化结束，系统可以正常工作

7.1.6 写数据到RAM

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
1	0	D7	D6	D5	D4	D3	D2	D1	D0

写数据（D0-D7）至显示存储器，写指令结束后，Y地址自动增1

7.1.7 从RAM中读数据

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
1	1	D7	D6	D5	D4	D3	D2	D1	D0

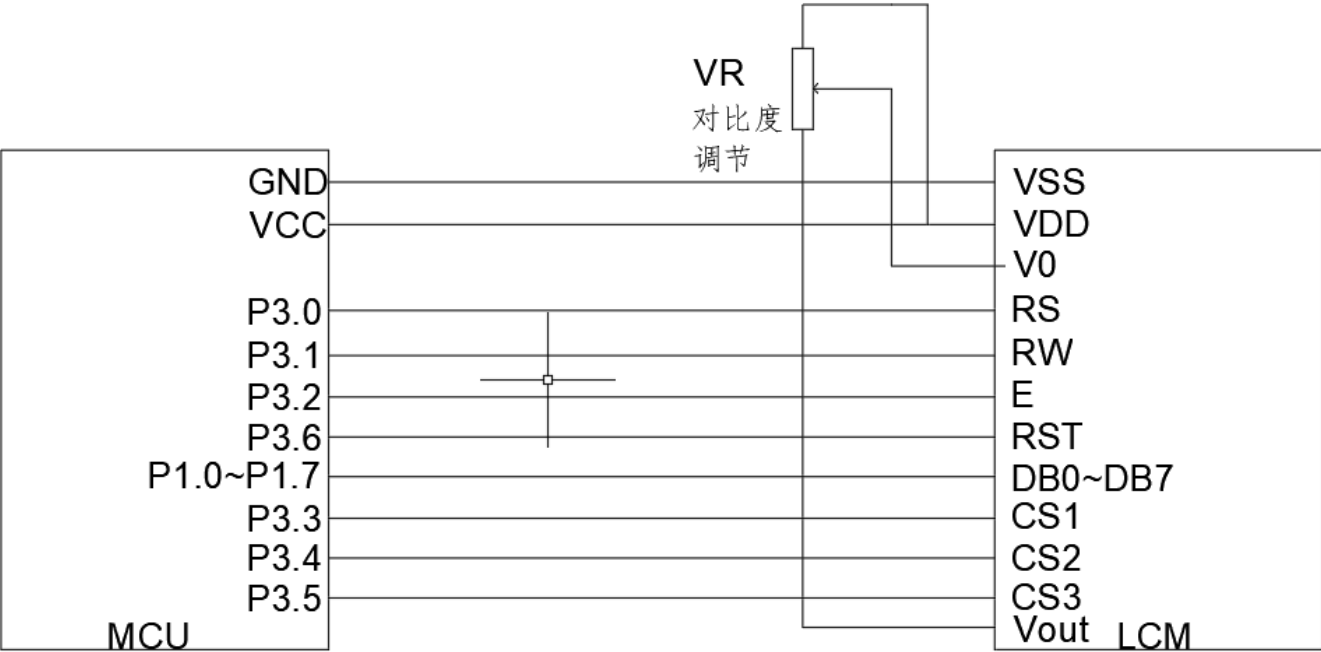
从显示存储器读数据（D0-D7），读指令结束后，Y地址自动增1

8.操作时序

参考检测波形图、

9.应用例程

9.1 并行传输接线图



9.2 并行C51例程

```
#include <STC15.H>
#include <string.h>
#include <INTRINS.H>
#define uchar unsigned char
#define uint unsigned int
sbit RS=P3^0;
sbit RW=P3^1;
sbit E=P3^2;
```

```
sbit CS1=P3^3;
sbit CS2=P3^4;
sbit CS3=P3^5;
sbit RST=P3^6;
```

```
//Defination Border Dots Matrix
```

```
uchar_code border_inf[8] = {
    0x01,0x00,0x00,0x00,0x00,0x00,0x00,0x80 };
```

```
uchar code cathode_txt[1024]=
```

```
{
```

```
////////////////////////////////////////////////////////////////
```

```
// Bitmap点阵数据表 //
```

```
// 图片: D:\. 19264-1.bmp,纵向取模下高位,数据排列:从左到右从上到下 //
```

```
// 图片尺寸: 192 * 64 //
```

```
////////////////////////////////////////////////////////////////
```

```
    0xFF,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,
    0x01,0x01,0x01,0x01,0x11,0x21,0x01,0x21,0xF9,0x21,0x01,0xF1,0x91,0x91,0xF1,0x01,
    0x01,0x11,0xD1,0x51,0xD1,0x51,0x79,0x51,0xD1,0x51,0xD1,0x11,0x01,0x01,0x41,0xF9,
    0x41,0xC1,0x01,0xF9,0x89,0x89,0x89,0xF9,0x01,0x01,0x01,0xF1,0x11,0x91,0x71,0x01,
    0xF1,0x11,0x11,0x11,0x11,0xF1,0x01,0x01,0x01,0xF1,0x51,0x51,0x51,0x51,0x51,
    0x51,0xF1,0x01,0x01,0x01,0x81,0x81,0x91,0x91,0x91,0x91,0x91,0x91,0x81,0x81,
    0x01,0x01,0x91,0x91,0xF1,0x89,0x81,0x91,0x21,0x01,0xF9,0x01,0x01,0x01,0x01,0x41,
    0x41,0xF9,0x41,0x21,0x21,0x21,0xF9,0x21,0x21,0x21,0x01,0x01,0x21,0x21,0xA1,0xE1,
    0xB9,0xA1,0xA1,0xA1,0xA1,0xA1,0x21,0x01,0x01,0xF9,0x09,0xC9,0x39,0x01,0xF9,0x49,
    0x49,0x49,0xF9,0x01,0x01,0x01,0x01,0x01,0xC1,0x39,0x01,0x81,0x01,0x39,0xC1,0x01,
    0x01,0x01,0x01,0x41,0x51,0x51,0x51,0x51,0x51,0x51,0x11,0xF1,0x01,0x01,0x01,
    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,
    0x00,0x00,0x00,0x00,0x21,0x12,0x00,0x3F,0x11,0x1F,0x40,0x3F,0x04,0x44,0x7F,0x00,
    0x00,0x00,0x7F,0x08,0x0A,0x0B,0x3E,0x0B,0x4A,0x48,0x7F,0x00,0x00,0x00,0x40,0x3F,
    0x00,0x3F,0x40,0x4F,0x48,0x48,0x48,0x4F,0x60,0x00,0x00,0x7F,0x10,0x11,0x0E,0x00,
    0x7F,0x21,0x21,0x21,0x21,0x7F,0x00,0x00,0x44,0x49,0x51,0x7F,0x41,0x41,0x41,0x7F,
    0x51,0x49,0x44,0x00,0x00,0x20,0x10,0x0C,0x00,0x40,0x7F,0x00,0x00,0x04,0x08,0x30,
    0x00,0x00,0x08,0x06,0x7F,0x02,0x04,0x08,0x09,0x08,0x7F,0x04,0x04,0x00,0x00,0x04,
    0x44,0x7F,0x02,0x41,0x47,0x29,0x11,0x29,0x47,0x40,0x00,0x00,0x02,0x01,0x7F,0x0A,
    0x0A,0x0A,0x0A,0x4A,0x7F,0x00,0x00,0x00,0x7F,0x10,0x10,0x0F,0x00,0x7F,0x22,
    0x0E,0x12,0x2B,0x44,0x00,0x00,0x02,0x21,0x30,0x28,0x26,0x21,0x20,0x28,0x30,0x61,
    0x02,0x00,0x00,0x00,0x3F,0x11,0x11,0x11,0x11,0x5F,0x40,0x40,0x3F,0x00,0x00,0x00,
    0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFF,
    0xFF,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
    0x00,0x00,0x00,0x00,0x88,0x10,0x18,0x48,0x28,0x18,0xC8,0x18,0x28,0x48,0x18,0x00,
    0x00,0x20,0xFC,0x20,0x00,0xFC,0x00,0x00,0xF8,0x00,0x00,0xFC,0x00,0x00,0x90,0xA0,
    0xFC,0xA0,0x90,0x28,0xD4,0xBC,0xA8,0xD4,0xBC,0x00,0x00,0x00,0x08,0x30,0x00,0x04,
    0x18,0x00,0x40,0x30,0x0C,0x00,0x00,0x00,0xF8,0x08,0xC8,0x38,0x00,0xF8,0x88,0x88,
```

0x88,0x88,0xF8,0x00,0x00,0x18,0x08,0x88,0x28,0x48,0x0C,0xE8,0x08,0x08,0x08,0x18,  
 0x00,0x00,0x40,0x80,0x00,0xFC,0x00,0x00,0x00,0xFC,0x00,0x80,0x40,0x00,0x00,0x10,  
 0x90,0xD0,0x70,0x5C,0x50,0x50,0x50,0x50,0xD0,0x10,0x00,0x00,0xFC,0x04,0x64,0x9C,  
 0x00,0xFC,0x24,0x24,0x24,0xFC,0x00,0x00,0x00,0x00,0x80,0x60,0x1C,0x00,0xC0,0x00,  
 0x1C,0x60,0x80,0x00,0x00,0x00,0x20,0xA8,0xA8,0xA8,0xA8,0xA8,0xA8,0x28,0x08,0xF8,  
 0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,  
 0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFF,  
 0xFF,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,  
 0x00,0x00,0x00,0x00,0x10,0x09,0x10,0x11,0x09,0x05,0x3F,0x05,0x09,0x11,0x11,0x00,  
 0x00,0x08,0x0F,0x24,0x10,0x0F,0x00,0x00,0x1F,0x00,0x00,0x3F,0x00,0x00,0x20,0x1F,  
 0x00,0x3F,0x10,0x01,0x3F,0x2A,0x3F,0x2A,0x2A,0x00,0x00,0x21,0x11,0x09,0x05,0x01,  
 0x01,0x01,0x05,0x09,0x11,0x21,0x00,0x00,0x3F,0x08,0x08,0x07,0x00,0x3F,0x10,0x10,  
 0x10,0x10,0x3F,0x00,0x00,0x22,0x22,0x12,0x13,0x0A,0x06,0x03,0x0A,0x0A,0x12,0x22,  
 0x00,0x00,0x20,0x20,0x21,0x3F,0x20,0x20,0x20,0x3F,0x21,0x20,0x20,0x00,0x00,0x01,  
 0x00,0x3F,0x05,0x05,0x05,0x05,0x05,0x25,0x3F,0x00,0x00,0x00,0x3F,0x08,0x08,0x07,  
 0x00,0x3F,0x11,0x07,0x09,0x15,0x22,0x00,0x00,0x01,0x10,0x18,0x14,0x13,0x10,0x10,  
 0x14,0x18,0x30,0x01,0x00,0x00,0x00,0x1F,0x08,0x08,0x08,0x08,0x2F,0x20,0x20,0x1F,  
 0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,  
 0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFF,  
 0xFF,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,  
 0x00,0x00,0x00,0x20,0x10,0xF8,0x00,0x00,0x00,0x60,0x90,0x08,0x90,0xE0,0x00,0x00,  
 0x30,0x08,0x08,0x90,0x60,0x00,0x08,0x10,0x60,0x80,0x60,0x10,0x08,0xE0,0x90,0x48,  
 0x88,0x10,0x00,0x00,0xC0,0x30,0xF8,0x00,0x00,0x00,0x00,0x00,0x00,0xF8,0x08,0x08,  
 0x08,0x10,0xE0,0x00,0x00,0xE0,0x10,0x08,0x08,0x10,0xE0,0x00,0x00,0x08,0x08,0xF8,  
 0x08,0x08,0x00,0x00,0x00,0x70,0x88,0x88,0x88,0x30,0x00,0x00,0x00,0x00,0x00,0x00,  
 0x00,0x00,0xFE,0x42,0x52,0x5A,0xAE,0xAA,0x2A,0x5A,0x42,0x42,0xFE,0x00,0x00,0x44,  
 0xFC,0x44,0x44,0xFC,0x44,0x00,0x10,0x88,0x46,0x30,0x00,0x00,0x44,0x88,0x00,0xC8,  
 0x38,0x8A,0x7C,0xA8,0x28,0xE8,0x08,0x00,0x00,0x80,0x80,0xBE,0xAA,0xAA,0x2A,0xAA,  
 0xAA,0xBE,0x80,0x80,0x00,0x00,0x10,0xD0,0xFE,0x90,0x04,0xF4,0x5E,0x54,0x5E,0xF4,  
 0x04,0x00,0x00,0x30,0xA8,0x66,0x10,0x00,0xFC,0x24,0x24,0x24,0xFC,0x00,0x00,0x00,  
 0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFF,  
 0xFF,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,  
 0x00,0x00,0x00,0x00,0x00,0x0F,0x00,0x00,0x00,0x04,0x08,0x09,0x04,0x03,0x00,0x00,  
 0x0C,0x0A,0x09,0x08,0x08,0x00,0x08,0x04,0x03,0x00,0x03,0x04,0x08,0x03,0x04,0x08,  
 0x04,0x03,0x00,0x03,0x02,0x02,0x0F,0x02,0x00,0x00,0x00,0x00,0x00,0x0F,0x08,0x08,  
 0x08,0x04,0x03,0x00,0x00,0x03,0x04,0x08,0x08,0x04,0x03,0x00,0x00,0x00,0x00,0x0F,  
 0x00,0x00,0x00,0x00,0x00,0x06,0x08,0x08,0x08,0x07,0x00,0x00,0x00,0x00,0x00,0x00,  
 0x00,0x00,0x1F,0x08,0x08,0x0A,0x0A,0x0C,0x0D,0x08,0x08,0x08,0x1F,0x00,0x00,0x10,  
 0x0F,0x00,0x00,0x1F,0x00,0x10,0x11,0x08,0x04,0x02,0x00,0x00,0x08,0x04,0x01,0x1F,  
 0x02,0x11,0x0A,0x04,0x0B,0x10,0x10,0x00,0x00,0x1F,0x0A,0x0A,0x0A,0x1F,0x00,0x1F,  
 0x0A,0x0A,0x0A,0x1F,0x00,0x00,0x01,0x00,0x1F,0x00,0x14,0x15,0x0D,0x07,0x0D,0x15,  
 0x14,0x00,0x00,0x09,0x09,0x05,0x05,0x10,0x1F,0x11,0x11,0x11,0x1F,0x10,0x00,0x00,  
 0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFF,  
 0xFF,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,  
 0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

```

0x00,0x00,0x00,0x00,0x00,0x00,0x30,0xC0,0x00,0xC0,0x30,0xC0,0x00,0xC0,0x30,0x00,
0x30,0xC0,0x00,0xC0,0x30,0xC0,0x00,0xC0,0x30,0x30,0xC0,0x00,0xC0,0x30,0xC0,0x00,
0xC0,0x30,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x30,0xC0,0x00,0xC0,0x30,0x00,0x00,
0x10,0x60,0x80,0x60,0x10,0x00,0x00,0x00,0x30,0xC0,0x00,0xC0,0x30,0x00,0x00,0x00,
0xFE,0x00,0x00,0xE0,0x10,0x10,0x10,0x20,0x00,0x00,0x00,0xF0,0x20,0x10,0x10,0xE0,
0x20,0x10,0x10,0xE0,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xE0,0x10,0x10,0x10,0x20,
0x00,0x00,0x00,0xE0,0x10,0x10,0x10,0xE0,0x00,0x00,0x00,0xF0,0x20,0x10,0x10,0xE0,
0x20,0x10,0x10,0xE0,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFF,
0xFF,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,
0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,
0x80,0x80,0x80,0x80,0x80,0x80,0x81,0x86,0x81,0x80,0x81,0x86,0x81,0x80,0x80,
0x80,0x81,0x86,0x81,0x80,0x81,0x86,0x81,0x80,0x80,0x81,0x86,0x81,0x80,0x81,0x86,
0x81,0x80,0x80,0x80,0x84,0x80,0x80,0x80,0x80,0xA0,0xA1,0x9E,0x81,0x80,0x80,0x80,
0x84,0x83,0x80,0x83,0x84,0x80,0x80,0x80,0xA0,0xA1,0x9E,0x81,0x80,0x80,0x80,0x80,
0x87,0x80,0x80,0x83,0x84,0x84,0x84,0x82,0x80,0x80,0x80,0x87,0x80,0x80,0x80,0x87,
0x80,0x80,0x80,0x87,0x80,0x80,0x80,0x84,0x80,0x80,0x80,0x83,0x84,0x84,0x84,0x82,
0x80,0x80,0x80,0x83,0x84,0x84,0x84,0x83,0x80,0x80,0x80,0x87,0x80,0x80,0x80,0x87,
0x80,0x80,0x80,0x87,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,
0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,
0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0xFF
};
unsigned char code bmp2[] =          // 数据表
{
    0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x40,0x40,0x40,0xC0,0x70,0x40,0x40,0x40,0x40,
    0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xF0,0x00,0x00,0x00,0xF0,0x00,0x00,0x00,0x00,
    0x00,0xE0,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x80,0x40,0x20,0x20,0x20,0x40,0x80,
    0x00,0x00,0x00,0xE0,0x20,0x20,0x20,0x20,0x40,0x80,0x00,0x00,0x00,0x00,0x00,0x00,
    0x00,0xE0,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x80,0x40,0x20,0x20,0x20,0x40,0x80,
    0x00,0x00,0xE0,0xC0,0x00,0x00,0x00,0x00,0x00,0xC0,0xE0,0x00,0x00,0x00,0x20,0x40,
    0x00,0x40,0xC0,0x50,0xE0,0x40,0x40,0x40,0x40,0x00,0x00,0x00,0x00,0x00,0xF0,0x50,
    0x50,0x50,0x50,0xF0,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
    0x00,0xFF,0xFF,0x03,0xFB,0xFB,0xFB,0xFB,0xFB,0xFB,0x03,0xFF,0xFF,0x00,0x2F,0x3F,
    0x7F,0xFF,0xFC,0xF8,0x78,0x78,0x78,0xCC,0xEF,0xCF,0xFC,0x7C,0x73,0x77,0x4F,0xFF,
    0xFF,0x63,0x43,0x03,0x3C,0x1C,0x0F,0x4F,0xE3,0x43,0x63,0x60,0xE8,0xFC,0xEC,0x4C,
    0x0F,0x0F,0x00,0xFF,0xFF,0x03,0xFB,0xFB,0xFB,0xFB,0xFB,0xFB,0x03,0xFF,0xFF,0x00,
    0x00,0x00,0x00,0x00,0x00,0x02,0x02,0x2A,0x2E,0x2B,0x4A,0x4A,0x6A,0x9A,0x02,
    0x02,0x00,0x00,0x00,0x81,0x82,0x84,0xFF,0x80,0x80,0x80,0xFF,0x84,0x82,0x81,0x00,
    0x00,0x7F,0x40,0x40,0x40,0x40,0x40,0x00,0x00,0x1F,0x20,0x40,0x40,0x40,0x20,0x10,
    0x00,0x00,0x00,0x7F,0x40,0x40,0x40,0x40,0x20,0x1F,0x00,0x00,0x00,0x00,0x00,0x00,
    0x00,0x7F,0x40,0x40,0x40,0x40,0x40,0x00,0x00,0x1F,0x20,0x40,0x40,0x40,0x20,0x10,
    0x00,0x00,0x7F,0x00,0x03,0x1C,0x60,0x1C,0x03,0x00,0x7F,0x00,0x00,0x00,0x42,0x24,
    0x08,0xFE,0x11,0x8C,0x53,0x25,0x59,0x87,0x80,0x00,0x00,0x00,0xFC,0x54,0x55,0x55,
    0xFD,0x01,0xFD,0x55,0x55,0x54,0xFC,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

```

0x00,0x8F,0x8F,0x0C,0xCD,0xCD,0xCD,0xCD,0xCD,0xCD,0x0C,0xCF,0xCF,0x00,0x7E,0xFE,  
 0xD0,0xC0,0x09,0x5B,0xF6,0x3E,0x3E,0xC1,0xF1,0xFD,0xF0,0xF0,0xCC,0xE8,0xB0,0xCF,  
 0xCF,0x36,0x36,0x0E,0x34,0x30,0x3C,0xA8,0x00,0xFE,0xFE,0xC6,0xC2,0xCD,0x86,0x86,  
 0x3E,0xBE,0xC0,0xCF,0xCF,0xCC,0xCD,0x0D,0x8D,0x8D,0xCD,0xCD,0xCC,0xCF,0xCF,0x00,  
 0x00,0x00,0x00,0x00,0x00,0x00,0x40,0x40,0xF8,0x40,0x10,0xD0,0x78,0x50,0x78,0xD0,  
 0x10,0x00,0x00,0x00,0xC0,0xA0,0x98,0x40,0x00,0xF0,0x90,0x90,0x90,0xF0,0x00,0x00,  
 0x00,0x10,0x90,0x70,0x10,0x00,0x10,0xF0,0x10,0x10,0xF0,0x10,0x00,0x00,0x00,0x70,  
 0x40,0x40,0xC0,0x78,0x40,0x40,0x48,0x50,0x40,0x40,0x00,0x00,0x00,0x00,0x10,0x10,  
 0xF0,0x10,0x10,0x10,0x90,0x50,0x30,0x00,0x00,0x00,0x00,0x80,0x70,0x40,0x40,0xF8,  
 0x40,0x40,0x40,0x40,0x00,0x00,0x00,0x00,0x00,0x20,0x20,0xA0,0x28,0x30,0x20,0x20,  
 0xA0,0x20,0x20,0x00,0x00,0x00,0x80,0x40,0x20,0x20,0x20,0xC0,0x00,0x00,0x80,0x40,  
 0x20,0x20,0x40,0x80,0x00,0x00,0x00,0xE0,0x20,0x20,0x20,0xC0,0x00,0x00,0x00,0x00,  
 0x80,0x39,0x39,0x38,0x1E,0x1E,0xFE,0xFE,0xE0,0xD9,0xD9,0x66,0x26,0x00,0xA4,0xF6,  
 0xDB,0x99,0x18,0x0A,0x06,0xD8,0xD8,0x66,0x26,0x07,0x63,0x71,0xC7,0x45,0x01,0x66,  
 0xF6,0x78,0x75,0x67,0x1C,0x18,0x1F,0xDF,0xE1,0x20,0x00,0xF7,0xF7,0xFE,0x1E,0x1F,  
 0x39,0x79,0xE0,0x7F,0x3F,0xDF,0x9B,0x18,0x9D,0x9F,0xF7,0xF7,0x3C,0x1F,0x1F,0x00,  
 0x00,0x00,0x00,0x00,0x00,0x00,0x04,0x03,0x7F,0x02,0x50,0x57,0x35,0x1D,0x35,0x57,  
 0x50,0x00,0x00,0x00,0x24,0x26,0x15,0x14,0x40,0x7F,0x44,0x44,0x44,0x7F,0x40,0x00,  
 0x00,0x02,0x3F,0x11,0x3F,0x00,0x41,0x3F,0x01,0x01,0x7F,0x01,0x00,0x00,0x00,0x20,  
 0x10,0x4C,0x43,0x26,0x2A,0x12,0x2A,0x26,0x40,0x40,0x00,0x00,0x00,0x40,0x30,0x0E,  
 0x41,0x43,0x24,0x28,0x11,0x29,0x45,0x43,0x00,0x00,0x41,0x40,0x44,0x44,0x44,0x7F,  
 0x44,0x44,0x44,0x40,0x40,0x00,0x00,0x00,0x40,0x3E,0x02,0x02,0x03,0x02,0x02,0x03,  
 0x02,0x02,0x02,0x00,0x00,0x00,0x0F,0x10,0x20,0x20,0x20,0x18,0x00,0x00,0x0F,0x10,  
 0x20,0x20,0x10,0x0F,0x00,0x00,0x00,0x3F,0x22,0x22,0x22,0x1D,0x00,0x00,0x00,0x00,  
 0x00,0x8F,0xC7,0x6C,0xE3,0xE3,0x63,0x63,0xE3,0xB1,0x18,0x9B,0xD3,0xE3,0xC3,0x03,  
 0x1F,0x1F,0x80,0x02,0x03,0x78,0x78,0x03,0x01,0x00,0x12,0x14,0x20,0x12,0x04,0x12,  
 0x84,0x10,0x10,0x26,0x40,0x44,0x04,0x00,0xE7,0xBE,0x1C,0x9B,0x9B,0x9F,0x9B,0x9B,  
 0x60,0x74,0x7C,0xDF,0x9F,0x60,0x71,0x1B,0x03,0x03,0x18,0x9A,0x9F,0x9B,0x9B,0x00,  
 0x00,0x00,0x00,0x00,0x00,0x00,0xC0,0x20,0x10,0x10,0x10,0x60,0x00,0x00,0x00,0xC0,  
 0x20,0x10,0x10,0x20,0xC0,0x00,0x00,0xC0,0x20,0x10,0x10,0x10,0x60,0x00,0x00,0x00,  
 0x40,0x40,0xA0,0x90,0x88,0x84,0x88,0x90,0xA0,0x40,0x40,0x00,0x00,0x00,0x08,0x88,  
 0x78,0x48,0x48,0xC8,0x00,0xF0,0x00,0x00,0xFC,0x00,0x00,0x00,0x10,0x10,0x50,0x94,  
 0x18,0x10,0x90,0x50,0x10,0x10,0x00,0x00,0x00,0x00,0x00,0x7C,0x44,0x44,0x44,0x44,  
 0x44,0x7C,0x00,0x00,0x00,0x00,0x00,0x60,0x50,0xCC,0x20,0x00,0x20,0x20,0xFC,0x90,  
 0x94,0x98,0x00,0x00,0x10,0x10,0x30,0x50,0x90,0x94,0x98,0x50,0x30,0x10,0x10,0x00,  
 0x00,0x00,0x40,0x40,0xA0,0x90,0x88,0x84,0x88,0x90,0xA0,0x40,0x40,0x00,0x00,0x00,  
 0x00,0x87,0x8F,0x8C,0xAC,0xB8,0x03,0x43,0xF0,0xFD,0xFF,0x6C,0x4C,0x00,0x83,0x83,  
 0x1F,0x1F,0x81,0xCD,0xEE,0x8E,0x9C,0x1C,0x00,0x00,0x00,0x00,0x02,0x08,0x60,0x7A,  
 0x7D,0x7B,0xF8,0x30,0x40,0x60,0x78,0x80,0xCC,0xFF,0xFF,0x79,0x71,0x6D,0x64,0x61,  
 0x02,0x2E,0x7C,0xEC,0xCC,0xBC,0x9E,0x0F,0x1E,0x3C,0x7C,0xFC,0xFD,0xFB,0x73,0x00,  
 0x00,0x00,0x00,0x00,0x00,0x00,0x07,0x08,0x10,0x10,0x10,0x0C,0x00,0x00,0x00,0x07,  
 0x08,0x10,0x10,0x08,0x07,0x00,0x00,0x07,0x08,0x10,0x11,0x11,0x0F,0x00,0x00,0x00,  
 0x20,0x24,0x24,0x24,0x24,0x3F,0x24,0x24,0x24,0x24,0x20,0x00,0x00,0x00,0x01,0x20,  
 0x11,0x0A,0x04,0x03,0x00,0x07,0x20,0x20,0x3F,0x00,0x00,0x20,0x1F,0x01,0x01,0x01,  
 0x01,0x01,0x01,0x01,0x01,0x00,0x00,0x00,0x00,0x3F,0x11,0x11,0x11,0x3F,0x00,0x3F,

```
0x11,0x11,0x11,0x3F,0x00,0x00,0x00,0x12,0x13,0x0A,0x0A,0x20,0x21,0x11,0x0F,0x14,
0x22,0x38,0x00,0x00,0x02,0x22,0x11,0x0F,0x00,0x00,0x00,0x3F,0x01,0x02,0x02,0x00,
0x00,0x00,0x20,0x24,0x24,0x24,0x24,0x3F,0x24,0x24,0x24,0x24,0x20,0x00,0x00,0x00,
0x00,0xFD,0xFD,0x07,0x05,0x0D,0xCC,0x6C,0x31,0x28,0x0C,0x36,0xB6,0x00,0xD7,0xFF,
0xCC,0x6C,0x3F,0x3F,0x0F,0x31,0x31,0x0C,0xC8,0xC1,0x8C,0x8C,0xC6,0xB7,0x3F,0x8E,
0x8F,0xFF,0xF7,0xC7,0x8E,0x8F,0xCF,0xCB,0xC1,0x0F,0x0F,0x00,0x52,0xF6,0xFA,0xFC,
0xC8,0xCE,0xC2,0xBF,0xBF,0x89,0x89,0xC0,0xC0,0x00,0x06,0x22,0xF1,0xBF,0x3E,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x88,0x48,0xE8,0xB8,0xAE,0xA8,0xA8,0xA8,0xA8,0xE8,
0x08,0x00,0x00,0x00,0x18,0x4A,0x5A,0x5A,0x0A,0x7E,0x0A,0x5A,0x5A,0x4A,0x18,0x00,
0x00,0x04,0x74,0x54,0x54,0xFC,0x54,0x7C,0x54,0x54,0x74,0x04,0x00,0x00,0x00,0x10,
0x10,0xFE,0x90,0x90,0x04,0x44,0x44,0x44,0x44,0xFC,0x00,0x00,0x00,0x08,0x08,0x08,
0xC8,0xB8,0x8E,0x88,0x88,0x88,0x88,0x08,0x00,0x00,0x20,0xF8,0x06,0xFC,0x84,0x04,
0xFC,0x00,0xF8,0x00,0xFE,0x00,0x00,0x00,0x00,0x08,0x08,0x08,0x08,0x08,0x08,0x08,
0x08,0x08,0x00,0x00,0x00,0x00,0x30,0xA8,0x66,0x10,0x20,0xF8,0x4E,0x48,0xFA,0x4C,
0x48,0x00,0x00,0x84,0xE4,0x5C,0xC4,0x02,0x7A,0x42,0x42,0x42,0x7E,0xC0,0x00,0x00,
0x00,0xF9,0xF9,0x18,0xD8,0xD8,0xD8,0xD8,0xD8,0x18,0xF9,0xF9,0x03,0xC3,0xC7,
0x06,0x26,0x3E,0xDE,0xC6,0x18,0x18,0xF8,0xF1,0xE1,0xF7,0xF7,0xC8,0xD9,0xF1,0x39,
0x39,0xFF,0xF6,0x26,0xDD,0xD9,0xD8,0xCD,0xF7,0xFE,0xFE,0xC7,0xD7,0x78,0x98,0xD8,
0x7F,0xFF,0xC1,0xD9,0xD9,0xC1,0xFF,0xFF,0x9A,0x18,0xC0,0xC5,0xDF,0x9A,0x18,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x1F,0x02,0x02,0x02,0x02,0x02,0x12,0x1F,
0x00,0x00,0x00,0x00,0x01,0x1D,0x05,0x05,0x1D,0x07,0x1D,0x05,0x15,0x1D,0x01,0x00,
0x00,0x11,0x11,0x15,0x17,0x09,0x09,0x09,0x15,0x13,0x11,0x01,0x00,0x00,0x00,0x01,
0x11,0x1F,0x00,0x00,0x08,0x08,0x08,0x08,0x08,0x1F,0x00,0x00,0x00,0x04,0x02,0x01,
0x1F,0x08,0x08,0x08,0x08,0x08,0x1F,0x00,0x00,0x00,0x00,0x1F,0x10,0x09,0x07,0x08,
0x11,0x00,0x03,0x10,0x1F,0x00,0x00,0x00,0x08,0x08,0x08,0x08,0x08,0x08,0x08,0x08,
0x08,0x08,0x08,0x00,0x00,0x00,0x09,0x09,0x05,0x05,0x00,0x1F,0x12,0x12,0x1F,0x12,
0x12,0x00,0x00,0x00,0x0F,0x04,0x0F,0x02,0x02,0x02,0x02,0x12,0x10,0x0F,0x00,0x00,
0x00,0x7F,0x7F,0x60,0x6F,0x6F,0x6F,0x6F,0x6F,0x6F,0x60,0x7F,0x7F,0x00,0x06,0x0F,
0x01,0x00,0x6C,0x2E,0x07,0x0F,0x07,0x18,0x3D,0x67,0x18,0x18,0x0E,0x54,0x78,0x1B,
0x1B,0x00,0x10,0x1B,0x7B,0x79,0x70,0x7A,0x7B,0x1C,0x1C,0x67,0x6E,0x7C,0x7B,0x7B,
0x18,0x1C,0x1C,0x33,0x73,0x6F,0x6F,0x00,0x61,0x63,0x6C,0x64,0x60,0x7C,0x7E,0x00
};
```

//Check Busy Flag

void check\_busy(void)

```
{
    RS=0;
    RW=1;
    P1=0Xff;
    while (1)
    {
        E=1;
        if (P1<0x80) break;
        E=0;
    }
}
```



```
E=0;
}
//End Function check_busy

//Delay Times(s Level)
#pragma disable
void delays(uint n)
{
    uint s;
    while (n-- >0)
        for (s=0;s<500;s++)
            {}
}

//Write Instruction Code
#pragma disable
void wr_cmd(uchar cmd)
{
    //check_busy();
    RS=0;
    RW=0;
    P1=cmd;
    _nop_();
    _nop_();
    E=1;
    _nop_();
    _nop_();
    E=0;
    _nop_();
    _nop_();
}

//Write Display RAM Data
#pragma disable
void wr_dat(uchar dat)
{
    //check_busy();
    RS=1;

    RW=0;
    _nop_();
    _nop_();
    P1=dat;
    _nop_();
    _nop_();
```

```
E=1;
_nop_();
_nop_();
E=0;
_nop_();
_nop_();
}

//S6B0108 Initial
#pragma disable
void initial()
{
    E=0;
    CS1=0;CS2=1;CS3=1;
    wr_cmd(0x3f); //Display On
    wr_cmd(0xc0); //Set Display Start Line(Z Address)

    CS1=1;CS2=0;CS3=1;
    wr_cmd(0x3f); //Display On
    wr_cmd(0xc0); //Set Display Start Line(Z Address)

    CS1=1;CS2=1;CS3=0;
    wr_cmd(0x3f); //Display On
    wr_cmd(0xc0); //Set Display Start Line(Z Address)
}

void disp_dat(uchar dat1,uchar dat2)
{
    uint page_cnt,col_cnt;
    uchar pg_address;
    pg_address = 0xb8;
    for (page_cnt=0;page_cnt<8;page_cnt++)
    {
        CS1=0;CS2=1;CS3=1;
        wr_cmd(pg_address); //Set Page Address(X Address)
        wr_cmd(0x40); //Set Address(Y Address)
        for (col_cnt=0;col_cnt<32;col_cnt++)
        {
            wr_dat(dat1);
            wr_dat(dat2);
        }
        CS1=1;CS2=0;CS3=1;
        wr_cmd(pg_address); //Set Page Address(X Address)
        wr_cmd(0x40); //Set Address(Y Address)
    }
}
```

```
    for (col_cnt=0;col_cnt<32;col_cnt++)
    {
        wr_dat(dat1);
        wr_dat(dat2);
    }
    CS1=1;CS2=1;CS3=0;
    wr_cmd(pg_address); //Set Page Address(X Address)
    wr_cmd(0x40); //Set Address(Y Address)
    for (col_cnt=0;col_cnt<32;col_cnt++)
    {
        wr_dat(dat1);
        wr_dat(dat2);
    }

    pg_address ++;
}
}
```

#pragma disable

```
void wr_border(void)
{
    uint page_cnt,col_cnt;
    uchar pg_address;
    pg_address = 0xb8;
    for (page_cnt=0;page_cnt<8;page_cnt++)
    {
        wr_cmd(pg_address); //Set Page Address(X Address)
        wr_cmd(0x40); //Set Address(Y Address)
        wr_dat(0xff);

        for (col_cnt=1;col_cnt<63;col_cnt++)
        {
            wr_dat(border_inf[page_cnt]);
        }
        wr_dat(0xff);
        pg_address ++;
    }
}
```

#pragma disable

```
void disp_border(void)
{
    CS1=0;CS2=1;CS3=1;
```

```
    wr_border();
    CS1=1;CS2=0;CS3=1;
    wr_border();
    CS1=1;CS2=1;CS3=0;
    wr_border();
}

#pragma disable
void disp_bmp(uchar_code *bmparea)
{
    uint i=0,j=0;
    uchar temp;
    CS1=0;CS2=1;CS3=1;
    temp=0xb8;
    wr_cmd(0xc0);
    for(j=0;j<8;j++)
    {
        wr_cmd(temp);
        wr_cmd(0x40);
        for(i=0;i<64;i++)
        { wr_dat(bmparea[i+j*192]);}
        temp++;
    }

    CS1=1;CS2=0;CS3=1;
    temp=0xb8;
    wr_cmd(0xc0);
    for(j=0;j<8;j++)
    {
        wr_cmd(temp);
        wr_cmd(0x40);
        for(i=0;i<64;i++)
        { wr_dat(bmparea[i+j*192+64]);}
        temp++;
    }

    CS1=1;CS2=1;CS3=0;
    temp=0xb8;
    wr_cmd(0xc0);
    for(j=0;j<8;j++)
    {
        wr_cmd(temp);
        wr_cmd(0x40);
        for(i=0;i<64;i++)
        { wr_dat(bmparea[i+j*192+128]);}
```

```
temp++;
}
}

void main()
{
    //S6B0108 initial
    RST = 0;
    delays(10);
    RST = 1;
    delays(10);
    initial();
    while (1)
    {
        disp_dat(0xff,0xff); //All Dots Display
        delays(100);
        disp_dat(0xaa,0xaa); //Seperate Rows I
        delays(100);
        disp_dat(0xff,0x00); //Seperate Columns II
        delays(100);
        disp_dat(0xaa,0x55); //Seperate Dots I
        delays(100);
        disp_dat(0x55,0xaa); //Separate Dots II
        delays(100);
        disp_bmp(cathode_txt);
        delays(100);
        disp_bmp bmp2);
        delays(100);
    }
}
```

## 10.注意事项

### 1.液晶显示器（LCD）

液晶显示器是由玻璃，有机密封胶，有机流体，和聚合物基偏光片。搬运时应注意以下事项：

- (1) .保持温度在使用和储存范围内。过高的温度和湿度会导致偏振退化、偏振器剥离或气泡。
- (2) .不要用比HB铅笔芯更硬的东西接触暴露的偏光镜。清除显示器表面的灰尘，用棉花轻轻擦拭，鹿皮巾或其他软材料浸泡在清洁油中。
- (3) 立即擦掉唾液或水滴。ITO与水接触时间过长，会导致液晶显示器表面变形或变色。
- (4) 玻璃很容易因粗暴的操作而碎裂。尤其是在角落和边缘。
- (5) .不要用直流电压驱动液晶显示器。

### 2.液晶显示模块

#### 2.1机械注意事项

LCM的装配和调整具有高精度。避免过度震动，不要做任何改动或修改。应注意以下几点。

- (1) .不要以任何方式改变金属框架上的凸耳。
- (2) .请勿通过钻额外的孔、更改其轮廓、移动其组件或修改其图案来修改PCB。

## 液晶显示模块使用说明书

---

(3) .请勿触摸弹性体连接器，尤其是插入背光面板（例如，EL）。

(4) .安装LCM时，请确保PCB板不受任何压力，如弯曲或扭曲。弹性体接触非常精细，任何元素的轻微错位都可能导致像素缺失。

(5) .避免压在金属挡板上，否则弹性体连接器可能会变形和失去接触，从而导致丢失像素。

### 2.2.静电

LCM包含CMOS LSI，对此类设备应采取相同的预防措施，即

(1) .当操作员与模块接触时，应将其接地。切勿用人体任何部位接触任何导电部件，如LSI焊盘、PCB上的铜导线和接口端子。

(2) .模块应保存在防静电袋或其他防静电容器中储存。

(3) .只能使用正确接地的烙铁。

(4) .如果使用电动螺丝刀，应良好接地，并防止换向器火花。

(5) .工作服和工作台应遵守正常的防静电措施；对于后者，建议使用导电（橡胶）垫。

(6) .由于干燥空气会感应静电，建议相对湿度为50-60%。

### 2.3.焊接

(1) .仅焊接至I/O端子。

(2) .只能使用接地正确且无漏电的烙铁。

(3) .焊接温度： $280^{\circ}\text{C} \pm 10^{\circ}\text{C}$

(4) .焊接时间：3到4秒。

(5) .使用树脂助焊剂填充的低温焊锡。

(6) .如果使用助焊剂，应覆盖LCD表面，以避免焊剂飞溅。助焊剂残留物应在防护后清除。

### 2.4.操作

(1) 观察角度可以通过改变LCD驱动电压V0来调节。

(2) 驱动电压应保持在规定的范围内，过高的电压会缩短显示器的寿命。(3) 响应时间随着温度的降低而增加。

(4) .在高于其工作范围的温度下，显示器可能会变成黑色或深蓝色；这（但是不要按压显示区域）可能会导致部分显示线段“断裂”。

(5) .操作过程中的机械损害（如按压显示区域）可能会导致线段出现“断裂”。

### 2.5.储存

如果有液体从损坏的玻璃电池中漏出，将任何接触的人体部分用肥皂和水冲洗干净。切勿吞下液体。毒性极低，但应始终小心。

### 2.6.有限保修

除非与客户另有约定，从装运日期起一年内，当根据验收标准检查时发现其电气和外观缺陷，将维修或修理其任何LCD和IC，该日期的确认应以货运单据为依据，保修责任仅限于根据上述条款进行的维修和/或更换。不会对任何后续或后果性事件负责。