

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：240128A

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
V3.0	2017-11-03	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

240128A 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，可以直接连接到8080系列MPU，允许选择不同的文本和图形模式组合，包括各种属性功能。

显示分辨率: 240 X 128dots

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: 1/128 **驱动偏压:** 1/12

控制芯片: RA6963

支持8080 8位MPU接口

内置256字字体ROM:基本ASCII码，日语，数字

支持最大64Kbyte外接显示SRAM显示模式:字符、图形和混合模式

字体大小:

- 水平:5-、6-、7-、8像素
- 垂直:频段

支持加粗字体和反向显示

尺寸 (Unit: mm)

外形尺寸: 144X104X12.1

可视区域: 111.4X64

字符尺寸:107.95X57.55

点尺寸:0.40X0.40

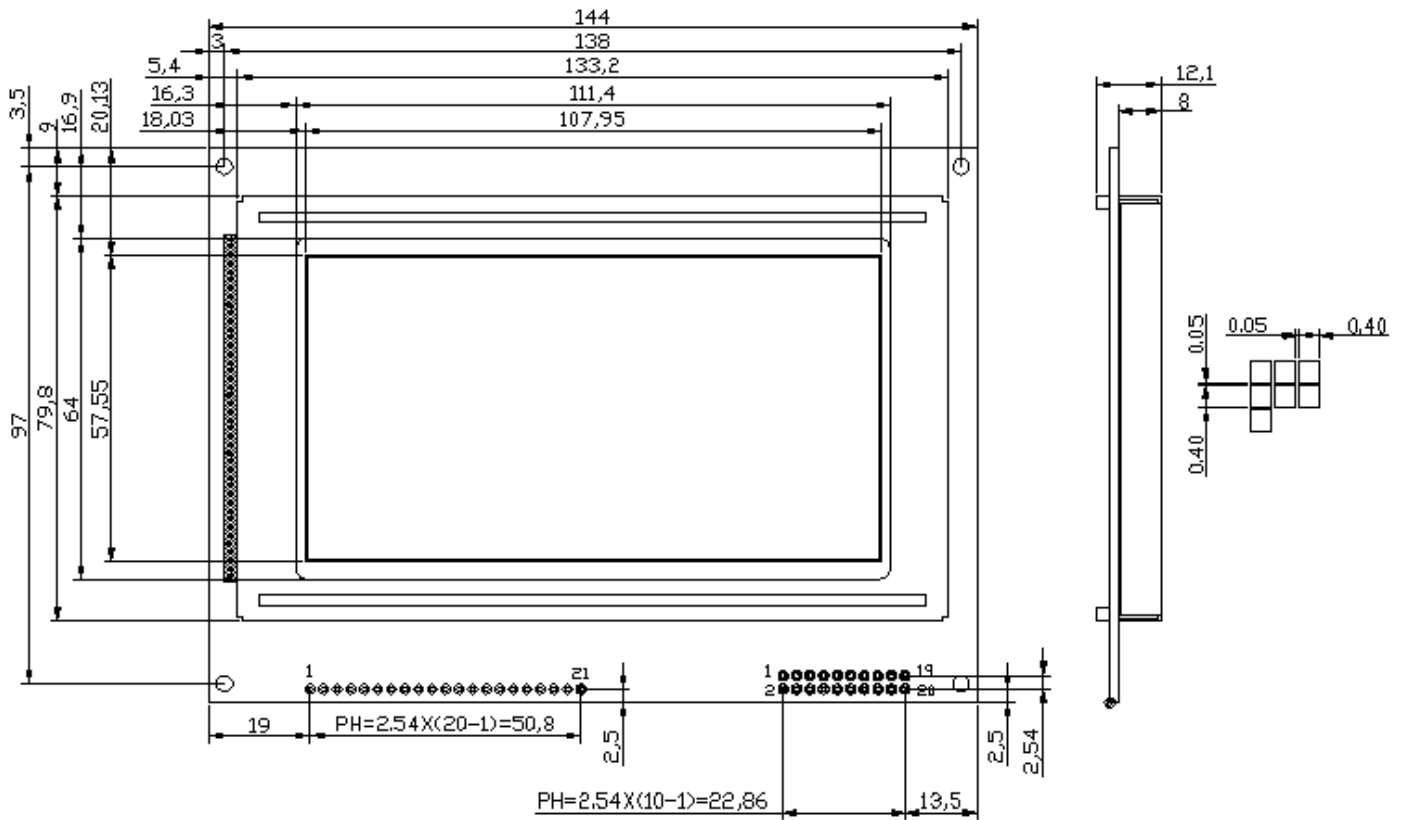
像素间距: 0.45X0.45

重量: g

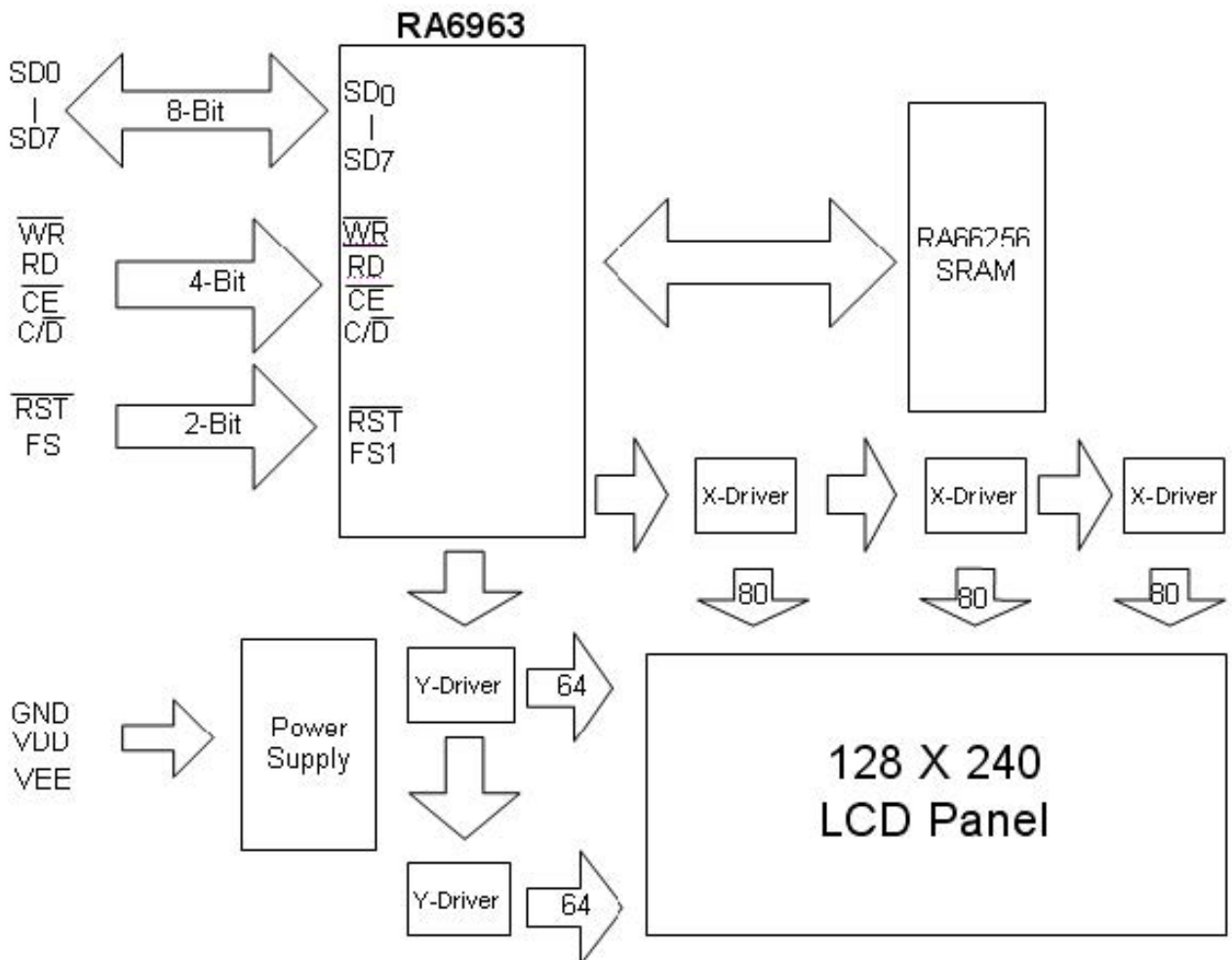
对比度: V0外部调节或内部固定对比度

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	°C
储存温度(T)	Tstg	-	-30	80	°C

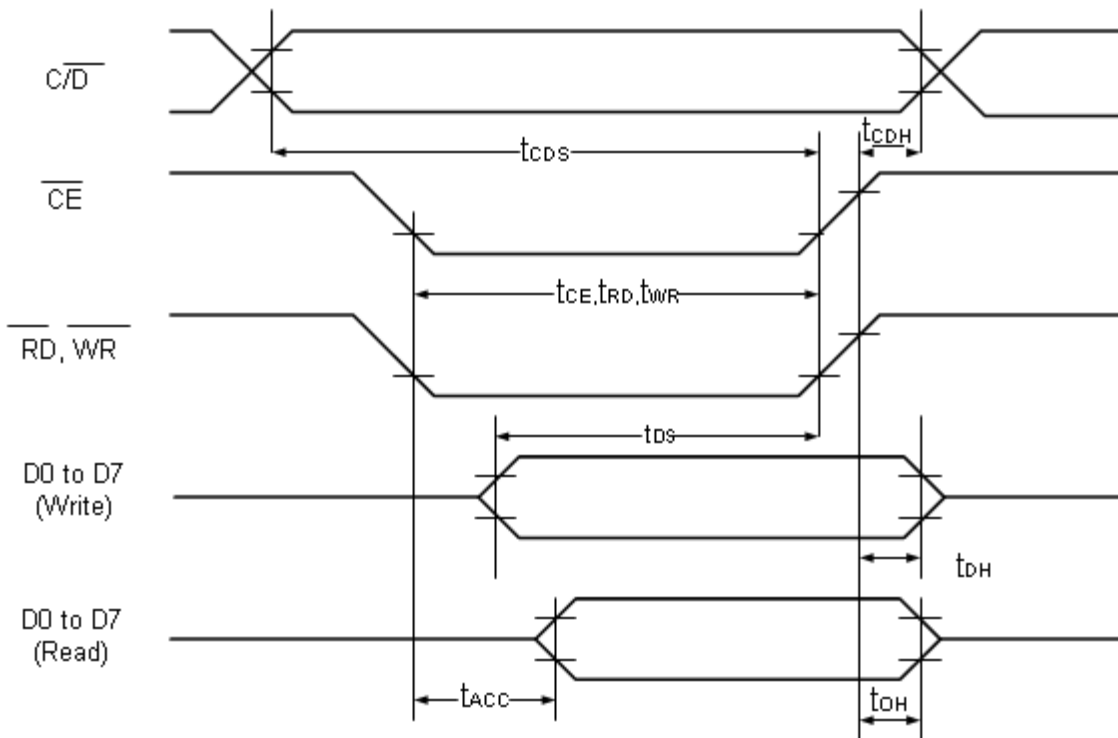
5.2 直流参数1(Ta=25°C, Vdd=3.0V~5.5V)

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	-	3.0	5.0	5.5	V
工作电流 (不包括背光)	Idd	Vdd=5V	3	6.5	12	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	136	144	160	mA
LED背光功耗	Pd		408	720	880	mW
输入高电平	Vih		0.8Vdd	-	Vdd	V
输入低电平	Vil		0	-	0.2Vdd	V
输出高电平	Voh		Vdd-0.3	-	Vdd	V
输出低电平	Vol		0	-	0.3	V

5.3 交流参数1(Ta=25°C, Vdd=3.0V~5.5V)MPU接口

参数名称	符号	条件	标称值			单位
			最小	典型	最大	
C/D 建立时间	tCDS		100	-	-	ns
C/D 保持时间	tCDH		10	-	-	ns
CE, RD, WR 脉冲宽度	tCE, tRD,		80	-	-	ns
数据建立时间	tDS		80	-	-	ns
数据保持时间	tDH		40	-	-	ns
存取时间	tACC		-	-	150	ns
输出保持时间	tOH		10	-	50	ns

测试波形图



6.接口说明

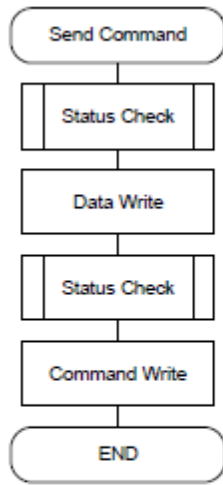
脚号	符号	功能	备注
1	FG	边框地	
2	VSS	供电电源负极	
3	VDD	供电电源正极	
4	V0	LCD偏压供电	可调节对比度
5	/WR	写使能	
6	/RD	读使能	
7	/CE	片选	
8	C/D	数据/指令选择(H:指令 L:数据)	
9	RST	系统复位脚	
10	D0	数据位0	
11	D1	数据位1	
12	D2	数据位2	
13	D3	数据位3	
14	D4	数据位4	
15	D5	数据位5	
16	D6	数据位6	
17	D7	数据位7	
18	FS	字体选择 (FS=1: 8*6点阵, FS=0: 8*8点阵)	
19	VEE	负压输出	
20	BLA	背光正极	
21	BLK	背光负极	

7.指令说明

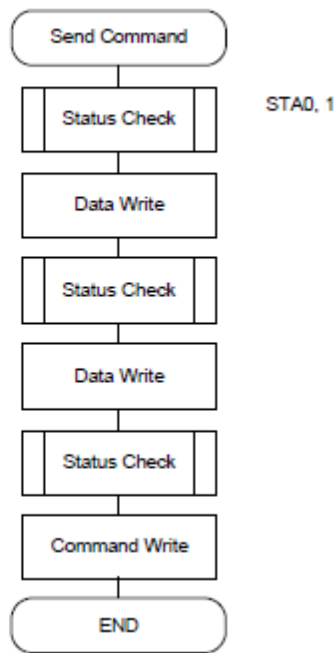
模块在设置数据时先发送数据，再发送指令。

发送指令的过程：

a) The case of 1 data



b) The case of 2 data



当发送超过两个数据时，最后一个（或两个数据）为有效数据。

7.1 指令描述

指令表

C/D: 1:指令/0:数据 -: 不考虑 #: 有效数值

指令	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	描述
设置光标指针	1	0	0	0	1	0	0	0	0	1	21H
	0	0	#	#	#	#	#	#	#	#	设置X地址
	0	0	#	#	#	#	#	#	#	#	设置Y地址
设置偏移寄存器	1	0	0	0	1	0	0	0	1	0	22H
	0	0	#	#	#	#	#	#	#	#	数值
	0	0	0	0	0	0	0	0	0	0	
设置地址指针	1	0	0	0	1	0	0	1	0	0	24H
	0	0	#	#	#	#	#	#	#	#	低字节
	0	0	#	#	#	#	#	#	#	#	高字节
设置文本首地址	1	0	0	1	0	0	0	0	0	0	40H
	0	0	#	#	#	#	#	#	#	#	低字节
	0	0	#	#	#	#	#	#	#	#	高字节
设置文本区域	1	0	0	1	0	0	0	0	0	1	41H
	0	0	#	#	#	#	#	#	#	#	列
	0	0	0	0	0	0	0	0	0	0	
设置图形首地址	1	0	0	1	0	0	0	0	1	0	42H
	0	0	#	#	#	#	#	#	#	#	低字节
	0	0	#	#	#	#	#	#	#	#	高字节
设置图形区域	1	0	0	1	0	0	0	0	1	1	43H
	0	0	#	#	#	#	#	#	#	#	列
	0	0	0	0	0	0	0	0	0	0	
OR模式	1	0	1	0	0	0	-	0	0	0	8XH
EXOR模式	1	0	1	0	0	0	-	0	0	1	
AND模式	1	0	1	0	0	0	-	0	1	1	
文本属性模式	1	0	1	0	0	0	-	1	0	0	
	0	0	-	-	-	-	#	#	#	#	

液晶显示模块使用说明书

内部CGROM模式	1	0	1	0	0	0	0	-	-	-	
外部CGRAM模式	1	0	1	0	0	0	1	-	-	-	
显示模式	1	0	1	0	0	1	#	#	#	#	图形/文本/光标/闪烁 开关
光标形状设置	1	0	1	0	1	0	0	#	#	#	设置光标: 1~8行
数据写入和ADP增加	1	0	1	1	0	0	0	0	0	0	C0H
	0	0	#	#	#	#	#	#	#	#	数据
数据读和ADP增加	1	1	1	1	0	0	0	0	0	1	C1H
数据写入和ADP减小	1	0	1	1	0	0	0	0	1	0	C2H
	0	0	#	#	#	#	#	#	#	#	数据
数据读和ADP减小	1	1	1	1	0	0	0	0	1	1	C3H
数据写入和ADP不变	1	0	1	1	0	0	0	1	0	0	C4H
	0	0	#	#	#	#	#	#	#	#	数据
数据读和ADP不变	1	1	1	1	0	0	0	1	0	1	C5H
数据自动写入	1	0	1	0	1	1	0	0	0	0	B0H
数据自动读出	1	0	1	0	1	1	0	0	0	1	B1H
自动读写复位	1	0	1	0	1	1	0	0	1	0	B2H
屏幕Peek	1	0	1	1	1	0	0	0	0	0	E0H
屏幕复制	1	0	1	1	1	0	1	0	0	0	E8H
位设置/重置	1	0	1	1	1	1	#	#	#	#	FXH
全屏反转	1	0	1	1	0	1	0	0	0	0	D0H
	0	0	-	-	-	-	-	-	-	#	0: 正常 1: 反转
	0	0	-	-	-	-	-	-	-	-	
闪烁时间	1	0	0	1	0	1	0	0	0	0	50H
	0	0	-	-	-	-	-	#	#	#	000b:0.066s 100b:1s 001b:0.25s 101b:1.25s 010b:0.5s 110b:1.5s 011b:1.75s 111b:2s
	0	0	-	-	-	-	-	-	-	-	
光标自动移动	1	0	0	1	1	0	0	0	0	0	60H
	0	0	-	-	-	-	-	-	-	#	0: 禁用 1: 启用
	0	0	-	-	-	-	-	-	-	-	
CGROM字体选择	1	0	0	1	1	1	0	0	0	0	70H
	0	0	-	-	-	-	-	-	#	#	10b:CGROM Font-01 11b:CGROM Font-02
	0	0	-	-	-	-	-	-	-	-	

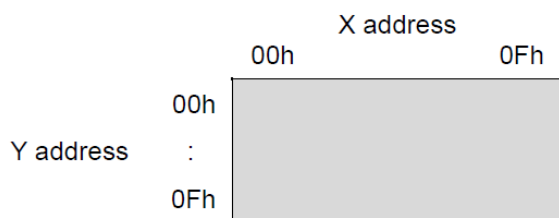
注:“-”为不考虑

7.1.1 设置光标指针

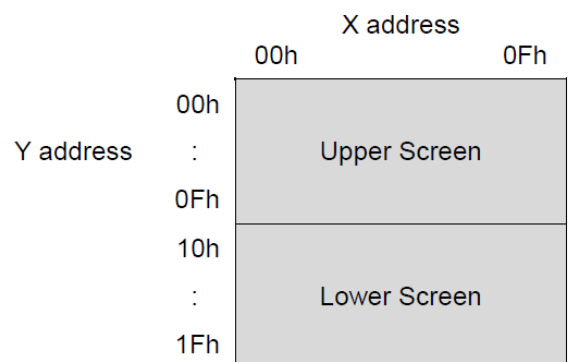
动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码 (21H)	1:C	0:W	0	0	1	0	0	0	0	1
设置X地址:00H~4FH	0	0	-	#	#	#	#	#	#	#
设置Y地址:00H~1FH	0	0	-	-	-	#	#	#	#	#

X地址和Y地址指定光标的位置。只能通过此命令移动光标位置。读取数据/从MPC写入不会改变光标指针。X地址和Y地址范围如下

Single Scan:



Dual Scan:



7.1.2 设置偏移寄存器

动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码 (22H)	1:C	0:W	0	0	1	0	0	0	1	0
	0	0	#	#	#	#	#	#	#	#
00H	0	0	0	0	0	0	0	0	0	0

模块有一个16位地址总线如下:

MSB											LSB				
A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
偏移寄存器的数据					字符代码							线扫描			
高5位定义了外部存储器中CGRAM区域的开始地址					在内部CG ROM模式下, 00h ~ 7Fh:预定义的“内部”CG ROM字符, 80h ~ FFh:用户自定义的“外部”字符。 在外部CG RAM模式下, 用户自定义字符。							这低3位表示定义字符形状的8行8点中的一个。			

显示RAM地址与偏移寄存器的关系:

偏移寄存器数据	CGRAM 开始到结束地址
00000	0000H~07FFH
00001	0800H~0FFFH
00010	1000H~17FFH
00011	1800H~1FFFH
-	-
-	-
11100	E000H~E7FFH
11101	E800H~EFFFH
11110	F000H~F7FFH
11111	F800H~FFFFH

7.1.3 设置地址指针

动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码 (24H)	1:C	0:W	0	0	1	0	0	1	0	0
低字节	0	0	#	#	#	#	#	#	#	#
高字节	0	0	#	#	#	#	#	#	#	#

命令用于指示向外部RAM写入(或从外部RAM读取)的起始地址。

流程:状态检查 → 设置地址数据(低8位) → 状态检查 → 设置地址数据(高8位) → 状态检查 → 设置地址指针

7.1.4 设置文本区首地址

动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码 (40H)	1:C	0:W	0	1	0	0	0	0	0	0
低字节	0	0	#	#	#	#	#	#	#	#
高字节	0	0	#	#	#	#	#	#	#	#

文本显示在外部显示RAM中的起始地址由这个命令定义。文本首地址表示最左边和最上面的位置。

显示RAM地址与显示位置的关系:

Display RAM Address		Display Position
TH	...	TH + CL
TH + TA	...	TH + TA + CL
(TH + 2TA) + TA	...	TH + 2TA + CL
(TH + 3TA) + TA	...	TH + 3TA + CL
⋮	⋮	⋮
TH + (n+1) TA	...	TH + (n-1) TA + CL

Remark: TH: Text Home address

TA: Text Area number (columns)

CL: Columns are fixed by hardware (pin-programmable).

7.1.5设置文本区域

动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码 (41H)	1:C	0:W	0	1	0	0	0	0	0	1
设置列数	0	0	#	#	#	#	#	#	#	#
00H	0	0	0	0	0	0	0	0	0	0

显示列由硬件设置定义。此命令可用于调整显示的列。

7.1.6设置图形区首地址

动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码 (42H)	1:C	0:W	0	1	0	0	0	0	1	0
低字节	0	0	#	#	#	#	#	#	#	#
高字节	0	0	#	#	#	#	#	#	#	#

图形显示在外部显示RAM中的起始地址由这个命令定义。图形区首地址表示最左边和最上面的位置。

外部显示RAM地址与显示位置的关系:

GH	GH + CL
GH + GA	GH + GA + CL
(GH + GA) + GA	GH + 2GA + CL
(GH + 2GA) + GA	GH + 3GA + CL
⋮	⋮	⋮
⋮	⋮	⋮
⋮	⋮	⋮
GH + (n-1) GA	GH + (n-1) GA + CL

Remark: GH: Graphic Home Address
 GA: Graphic Area Number (columns)
 CL: Columns are fixed by hardware (pin-programmable).

7.1.5 设置图形区域

动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码 (43H)	1:C	0:W	0	1	0	0	0	0	1	1
设置列数	0	0	#	#	#	#	#	#	#	#
00H	0	0	0	0	0	0	0	0	0	0

显示列由硬件设置定义。此命令可用于调整图形区显示的列。

模式设置命令:显示模式由以下命令定义。在发送下一个命令之前，显示模式不变。

7.1.6 OR模式

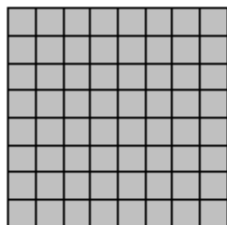
动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码(80H或88H)	1:C	0:W	1	0	0	0	-	0	0	0

7.1.7 EXOR模式

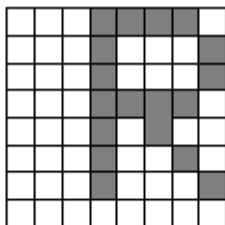
动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码(81H或89H)	1:C	0:W	1	0	0	0	-	0	0	1

7.1.8 AND模式

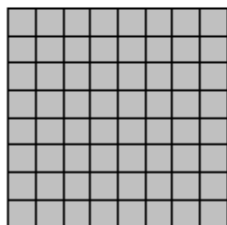
动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码(83H或8BH)	1:C	0:W	1	0	0	0	-	0	1	1



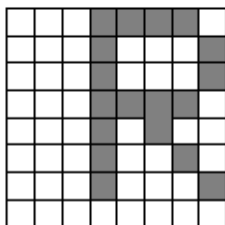
Graphic



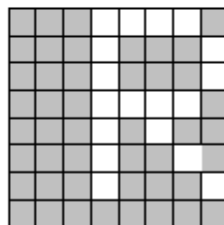
Text



“OR”



“AND”



“EXOR”

注：逻辑OR、EXOR和AND可应用于文本或图形显示

7.1.9 文本属性模式

动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码(84H或8CH)	1:C	0:W	1	0	0	0	-	1	0	0
设置禁止/反转/加粗/闪烁	0	0	-	-	-	-	#	#	#	#

属性操作是选择反向显示、字符闪烁、加粗和禁止的组合。属性功能模式只能显示文本;图形显示被自动禁用。但是，必须使用显示模式命令来打开文本和图形区域，以使属性函数可用。文本区域中每个字符的属性数据被写入图形区域中的相同地址。

D (3: 0)

0000b:正常显示

0011b:禁止显示

0101b:反转显示

0111b:加粗显示

1000b:正常闪烁显示

1011b:禁止闪烁显示

1101b:反转闪烁显示

1111b:加粗闪烁显示

注意:属性函数只能应用于文本显示，因为属性数据被放置在图形内存区域。

7.1.10 内部CGROM模式

动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码(80H~87H)	1:C	0:W	1	0	0	0	0	-	-	-

此命令选择内部CG ROM模式。在内部字符生成器模式，字符代码00h到7Fh被分配到内置字符生成器ROM。

00h ~ 7Fh:预定义的内部CG ROM字符

80h ~ FFh:用户自定义的“外部”字符

7.1.11 外部CGRAM模式

动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码(88H~8FH)	1:C	0:W	1	0	0	0	1	-	-	-

此命令选择外部CG RAM模式。在此模式下，字符码80h到FFh自动分配到外部字符发生器RAM中。

7.1.12 显示模式设置

动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码(90H~9FH)	1:C	0:W	1	0	0	1	#	#	#	#

液晶显示模块使用说明书

显示模式由该命令定义，在发送下一个命令之前不会更改。可以显示文本或图形显示的逻辑OR、EXOR、AND。

	D3:图形显示	D2: 文本显示	D1: 光标显示	D0: 光标闪烁
0b	OFF	OFF	OFF	OFF
1b	ON	ON	ON	ON

D(3:0):0000b:显示关闭 xx10b:光标开, 闪烁关 xx11b:光标、闪烁开
 01xxb:文本开, 图形关 10xxb:文本关, 图形开 11xxb:文本开、图形开

注意：在下列情况下，必须打开“文本显示”和“图形显示”：

- a) 文本/图形显示组合
- b) 属性功能

7.1.13 光标形状设置

动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码(A0H~A7H)	1:C	0:W	1	0	1	0	0	#	#	#

当光标显示处于开启状态时，此命令在1行到8行范围内选择光标模式。光标地址由光标指针设置命令定义。

7.1.14 数据写入和ADP增加

动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码(C0H)	1:C	0:W	1	1	0	0	0	0	0	0
写入数据	0	0	#	#	#	#	#	#	#	#

数据写操作后，地址指针（ADP）将增加。

7.1.15 数据读取和ADP增加

动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码(C1H)	1:C	0:W	1	1	0	0	0	0	0	1

数据将被自动读到一个特定的寄存器，数据读操作后地址指针（ADP）增加。

7.1.16 数据写入和ADP减小

动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码(C2H)	1:C	0:W	1	1	0	0	0	0	1	0
写入数据	0	0	#	#	#	#	#	#	#	#

数据写操作后，地址指针（ADP）将减小。

7.1.17 数据读取和ADP减小

动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码(C3H)	1:C	0:W	1	1	0	0	0	0	1	1

数据将被自动读到一个特定的寄存器，数据读操作后地址指针（ADP）减小。

7.1.18 数据写入和ADP不变

动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码(C4H)	1:C	0:W	1	1	0	0	0	1	0	0
写入数据	0	0	#	#	#	#	#	#	#	#

数据写操作后，地址指针（ADP）不变。

7.1.19 数据读取和ADP不变

动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码(C5H)	1:C	0:W	1	1	0	0	0	1	0	1

数据将被自动读到一个特定的寄存器，数据读操作后地址指针（ADP）不变。

7.1.20 数据自动写入

动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码(B0H)	1:C	0:W	1	0	1	1	0	0	0	0

流程:设置地址指针→ 状态检查1→ 自动写(B0h)→ 状态检查2→ 数据写→ 状态检查2→ 数据写→ 状态检查2→ 自动复位(B2h)

7.1.21 数据自动读

动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码(B1H)	1:C	0:W	1	0	1	1	0	0	0	1

流程:设置地址指针→ 状态检查1→ 自动读(B1h)→ 状态检查2→ 读数据→ 状态检查2→ 读数据→ 状态检查2→ 自动复位(B2h)

7.1.22 自动复位

动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码(B2H)	1:C	0:W	1	0	1	1	0	0	1	0

7.1.23 屏幕Peek

动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码(E0H)	1:C	0:W	1	1	1	0	0	0	0	0

此命令用于将1个字节的显示数据传输到数据栈;这个字节可以通过MPU数据访问读取。该命令可以读取LCD屏幕上文本和图形显示数据的逻辑组合。

状态(STA6)应该在屏幕Peek命令之后进行检查。如果设置地址指针命令设置的地址不在图形区域中，则忽略该命令并置位状态标志(STA6)。

流程:设置地址指针→□状态检查1→□屏幕Peek (E0h)→□状态检查→□状态检查1→□数据访问

注意:当硬件列号和软件列号相同时，此命令可用。硬件列号与MD2和MD3设置相关。软件列号与设置文本区和设置图形区命令有关。

7.1.24 屏幕复制

动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码(E8H)	1:C	0:W	1	1	1	0	1	0	0	0

此命令将一行点阵数据复制到图形区域。必须使用set Address指针设置起始点命令。

注1: 如果正在使用属性功能，则此命令不可用。（属性数据是图形区域数据。）

注2: 对于双扫描，不能使用此命令（因为此模式下不能分辨上屏幕数据和下屏数据）。

流程: 设置地址指针→ 状态检查1→ 屏幕拷贝(E8h)→ 状态检查(是否STA6 = 1)→ 状态检查(是否STA0/1 = 1)

注意:当硬件列号相同时，此命令可用。硬件列号与MD2和MD3设置有关。软件列号与

设置文本区域和设置图形区域命令有关。

7.1.25 位的置位和清零

动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码(FxH)	1:C	0:W	1	1	1	1	#	#	#	#

此命令置位或清零由地址指针指定的字节的一个位。一次只能置位或清零一位。

D(3): 0:清零 1:置位

D(2:0): 000b~111b 0~7位

流程: 设置地址指针 → 状态检查1 → 置位或清零

7.1.26 全屏反转

动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码 (D0H)	1:C	0:W	1	1	0	1	0	0	0	0
反转/正常	0	0	-	-	-	-	-	-	-	#
虚拟写	0	0	-	-	-	-	-	-	-	-

0:正常 1:反向

这个命令(D0h)用于反转整个屏幕显示的数据。当此功能被启用时,液晶屏上显示的数据将被反转以显示反转模式。

7.1.27 闪烁时间设置

动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码 (50H)	1:C	0:W	0	1	0	1	0	0	0	0
闪烁时间	0	0	-	-	-	-	-	#	#	#
虚拟写	0	0	-	-	-	-	-	-	-	-

通过此命令调整光烁功能的闪烁时间(50h)。例如,如果帧的频率为60Hz,眨眼时间可以通过软件选择从0.066秒调整为2秒。

000: 0.066秒 001: 0.25秒 010: 0.5秒 011: 0.75秒
 100: 1 101: 1.25秒 110: 1.5秒 111: 2 s

注意:在这个函数中,在发送命令之前必须发送两个数据,但是第二个数据(D2)的内容可以是任何值。

7.1.28 光标自动移动

动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码 (60H)	1:C	0:W	0	1	1	0	0	0	0	0
自动移动开/关	0	0	-	-	-	-	-	-	-	#
虚拟写	0	0	-	-	-	-	-	-	-	-

模块提供了一种独特的自动光标移动功能。在写入(读取)每个显示数据后,光标指针在光标自动移动模式下自动增加/减少1。

0:禁用 1:启用

注意:在这个函数中,在发送命令之前必须发送两个数据,但是第二个数据(D2)的内容可以是任何值。

7.1.29 CGROM字体选择

液晶显示模块使用说明书

动作	C/D	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写指令代码 (70H)	1:C	0:W	0	1	1	1	0	0	0	0
CGROM字体选择	0	0	-	-	-	-	-	-	#	#
虚拟写	0	0	-	-	-	-	-	-	-	-

这个命令(70h)是选择字符字体映射的一个方便的功能。用户可以从CGROM Font-01或CGROM Font-02获得更多内置字符，这是由软件选择决定的。

00: Don't care 01: Don't care 10: CGROM Font-01 11: CGROM Font-02

注意:在这个函数中，在发送命令之前必须发送两个数据，但是第二个数据(D2)的内容可以是任何值。

CGROM Font - 02

LSB \ MSB	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
1	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
2	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
3	P	Q	R	S	T	U	U	W	X	Y	Z	[\]	^	_
4	千	万	月	*	*	■	ヲ	ア	イ	ウ	エ	オ	カ	ク	ケ	コ
5	一	ア	イ	ウ	エ	オ	カ	キ	ク	ケ	コ	サ	シ	ス	セ	ソ
6	タ	チ	ツ	テ	ト	ナ	ニ	ヌ	ネ	ノ	ハ	ヒ	フ	ホ	ヘ	フ
7	ミ	ム	メ	モ	カ	キ	ク	ケ	コ	レ	ロ	ワ	ウ	エ	オ	カ

CGROM Font - 01

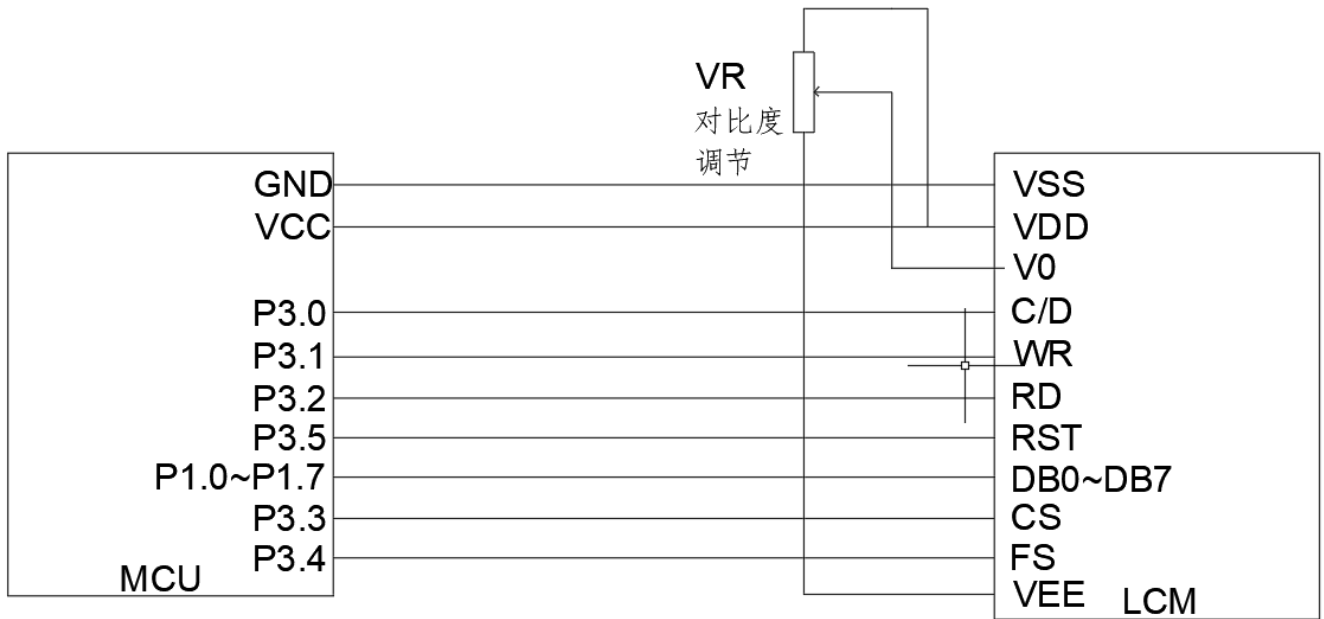
LSB \ MSB	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
1	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
2	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
3	P	Q	R	S	T	U	U	W	X	Y	Z	[\]	^	_
4	~	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
5	p	q	r	s	t	u	u	w	x	y	z	{		}	~	
6	ÿ	ü	ë	ä	ä	ä	ä	ö	ö	ö	é	ï	ï	ï	ä	ä
7	ÿ	ÿ	ÿ	ö	ö	ö	ö	ö	ö	ö	ö	ö	ö	ö	ö	ö

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
#define db P1
sbit cd=P3^0;
sbit wr=P3^1;
sbit rd=P3^2;
sbit cs=P3^3;
sbit fs=P3^4;
sbit rst=P3^5;
uchar code IC_DAT1 []={
" websitehttp://www.yxylcm.com "
" mobile: 18028784142 "
" 240*64 DOTS + Character LCM "
"abcdefghijklmnopqrstuvwxyz,./:'- "
};
////////////////////////////////////
```

```

void delay(unsigned int m) //延时程序
    {
        unsigned int i,j;
        for(i=0;i<m;i++)
            for(j=0;j<20;j++);
    }

////////////////////////////////////
/*****
*****/

void delayms(unsigned int n) //延时10×n毫秒程序
    {
        unsigned int i,j;
        for(i=0;i<n;i++)
            {
                for(j=0;j<800;j++);
            }
    }

void wr_cmd (uchar cmd)
{
    cd=1;
    rd=1;
    wr=0;
    db=cmd;
    delay(1);
    wr=1;
}

void wr_dat (uchar dat)
{
    cd=0;
    rd=1;
    wr=0;
    db=dat;
    delay(1);
    wr=1;
}

void wr_autowr (uchar dat)
{
    cd=0;
    rd=1;
    wr=0;
}

```

```
    db=dat;
    delay(1);
    wr=1;
}

void wr_cmd3 (uchar cmd,uchar cs1,uchar cs2)
{
    wr_dat (cs2);
    wr_dat (cs1);
    wr_cmd (cmd);
}

void cleabmp ()
{
    uchar i,j;
    wr_cmd3 (0x24,0,0);
    for(j=0;j<128;j++)
    {
        wr_cmd (0xb0);
        for(i=0;i<30;i++)
            wr_autowr (0x00);
        wr_cmd (0xb2);
    }
    wr_cmd3 (0x24,0x10,0);
    for(j=0;j<128;j++)
    {
        wr_cmd (0xb0);
        for(i=0;i<30;i++)
            wr_autowr (0x00);
        wr_cmd (0xb2);
    }
}

void initial ()
{
    fs=0;
    cs=0;
    wr=1;
    rd=1;
    wr_cmd3 (0x24,0,0);
    wr_cmd3 (0x22,0,0);
    wr_cmd3 (0x40,0x10,0);
    wr_cmd3 (0x41,0,0x1e);
    wr_cmd3 (0x42,0x00,0);
```

```
wr_cmd3 (0x43,0,0x1e);  
wr_cmd (0x80);  
wr_cmd3 (0x70,0,2);  
wr_cmd (0x9c);  
}
```

```
void disp_dat (uchar dat1,uchar dat2)  
{  
    uint page_cnt,col_cnt;  
    wr_cmd3 (0x24,0,0);  
    wr_cmd (0xb0);  
    for(page_cnt=0;page_cnt<64;page_cnt++)  
    {  
        for (col_cnt=0;col_cnt<30;col_cnt++)  
        {  
            wr_autowr (dat1);  
        }  
        for (col_cnt=0;col_cnt<30;col_cnt++)  
        {  
            wr_autowr (dat2);  
        }  
    }  
    wr_cmd (0xb2);  
}
```

```
void disp_str (uchar code *txtzz)  
{ uchar i;  
wr_cmd3 (0x24,8,0);  
wr_cmd (0xb0);  
for(i=0;i<120;i++)  
    {  
wr_autowr (i);//txtzz[i]-0x20);  
    //txtzz++;  
    }  
    for(i=0;i<120;i++)  
    {  
wr_autowr (txtzz[i]-0x20);  
    //txtzz++;  
    }  
    for(i=60;i<120;i++)  
    {  
wr_autowr (i);//txtzz[i]-0x20);  
    //txtzz++;  
    }  
wr_cmd (0xb2);
```

```
}

```

```
void disp_bmp (uchar code *bmparea)

```

```
{uint i;

```

```
wr_cmd3 (0x24,0,0);

```

```
wr_cmd (0xb0);

```

```
for(i=0;i<3840;i++)

```

```
{

```

```
wr_autowr (bmparea[i]);

```

```
//txtzz++;

```

```
}

```

```
wr_cmd (0xb2);

```

```
}//////////

```

```
/******

```

```
* 名称 : Main()

```

```
* 功能 : 主函数

```

```
*****/

```

```
void main(void)

```

```
{

```

```
rst = 0;

```

```
delays(10);

```

```
rst = 1;

```

```
delays(100);

```

```
initial();

```

```
while (1)

```

```
{

```

```
cleabmp();

```

```
disp_dat(0xff,0xff); //All Dots Display

```

```
delays(100);

```

```
disp_dat(0x00,0x00); //Clear Screen

```

```
delays(100);

```

```
disp_dat(0xaa,0xaa); //Seperate Rows I

```

```
delays(100);

```

```
disp_dat(0x55,0x55); //Seperate Rows II

```

```
delays(100);

```

```
disp_dat(0xff,0x00); //Seperate Columns II

```

```
delays(100);

```

```
disp_dat(0x00,0xff); //Seperate Columns II

```

```
delays(100);

```

```
disp_dat(0xaa,0x55); //Seperate Dots I

```

```
delays(100);

```

```
disp_dat(0x55,0xaa); //Separate Dots II

```

```
delays(100);

```

```
cleabmp();

```

```
disp_str(IC_DAT1);

```



```
delays(250);
cleabmp();
disp_bmp bmp1);
delays(250);

} }
```

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。
- (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。