

JLX12864G-99-2 使用说明书

目 录

序号	内 容 标 题	页码
1	概述	2
2	字符型模块的特点	2
3	外形及接口引脚功能	2~3
4	基本原理	4
5	技术参数	4
6	时序特性	5~6
7	指令功能及硬件接口与编程案例	6~13

1. 概述

晶联讯电子专注于液晶屏及液晶模块的研发、制造。所生产 JLX12864G-99-2 型液晶模块由于使用方便、显示清晰，广泛应用于各种人机交流面板。

JLX12864G-99-2 可以显示 128 列*64 行点阵单色图片,或显示 8 个/行*4 行 16*16 点阵的汉字,或显示 16 个/行*8 行 8*8 点阵的英文、数字、符号。

2. JLX12864G-99-2 图像型点阵液晶模块的特性

- 1.1 重量轻: ≤65g;
- 1.2 视窗大: 65.5*38.0mm 厚;
- 1.3 结构牢: 带 PCB、背光、铁框
- 1.4 IC 采用矽创公司 ST7565R, 功能强大, 稳定性好
- 1.5 功耗低: 10 - 100mW (不带背光 10mW, 带背光不大于 100mW) ;
- 1.6 显示内容:
 - 128*64 点阵单色图片;
 - 可选用 16*16 点阵或其他点阵的图片来自编汉字, 按照 16*16 点阵汉字来计算可显示 8 字/行*4 行。按照 12*12 点阵汉字来计算可显示 10 字/行*4 行。
- 1.7 指令功能强: 可组合成各种输入、显示、移位方式以满足不同的要求;
- 1.8 接口简单方便: 采用 4 线 SPI 串行接口, 可只需 5 位 MPU 的端口 (4 线 SPI 接口加上复位信号线<RESET>)。如需并行接口, 可选用我厂的 JLX12864G-100 等产品代换。
- 1.9 工作温度宽: -20℃ - 70℃ ;
- 1.10 可靠性高: 寿命为 50,000 小时 (25℃)。

3. 外形尺寸及接口引脚功能

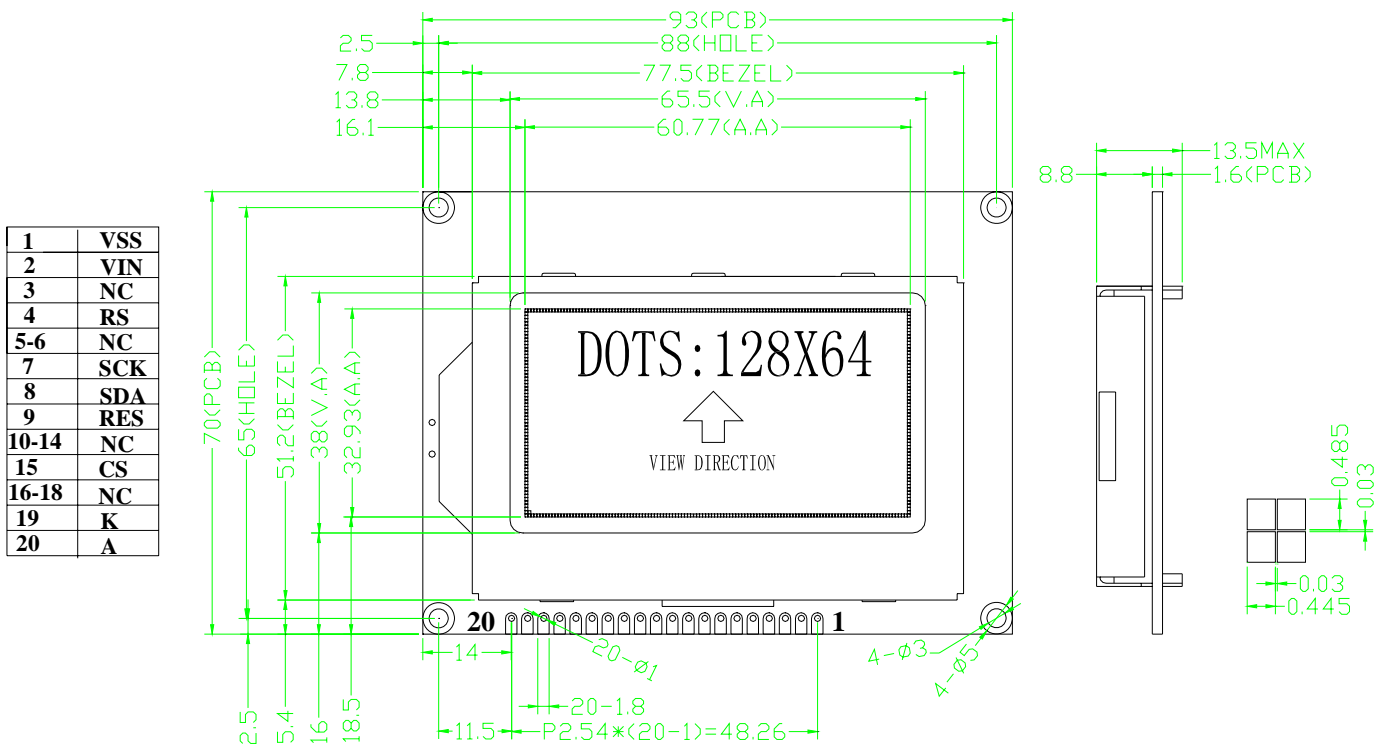


图 1. 外形尺寸

模块的接口引脚功能

引线号	符号	名称	功能
1	VSS	接地	0V
2	VIN	电路电源	5V, 或 3.3V 可选
3	NC	空脚	
4	RS	寄存器选择信号	H:数据寄存器 0:指令寄存器 (IC 资料上所写为" A0")
5~6	NC	空脚	
7	SCK	串行时钟	串行时钟 (IC 资料上所写为" SCL")
8	SDA	串行数据	数据传输 (IC 资料上所写为" SI")
9	RES	复位	低电平复位, 复位完成后, 回到高电平, 液晶模块开始工作
10~14	NC	空脚	
15	CS	片选	低电平片选
16~18	NC	空脚	
19	K	LED 背光负极	0V
20	A	LED 背光正极	5V, 或 3.3V 可选

表 1: 模块的接口引脚功能

4. 基本原理

4.1 液晶屏 (LCD)

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

4.2 工作电图:

图 1 是 JLX12864G-99-2 图像点阵型模块的电路框图, 它由驱动 IC ST7565R 及几个电阻电容组成。

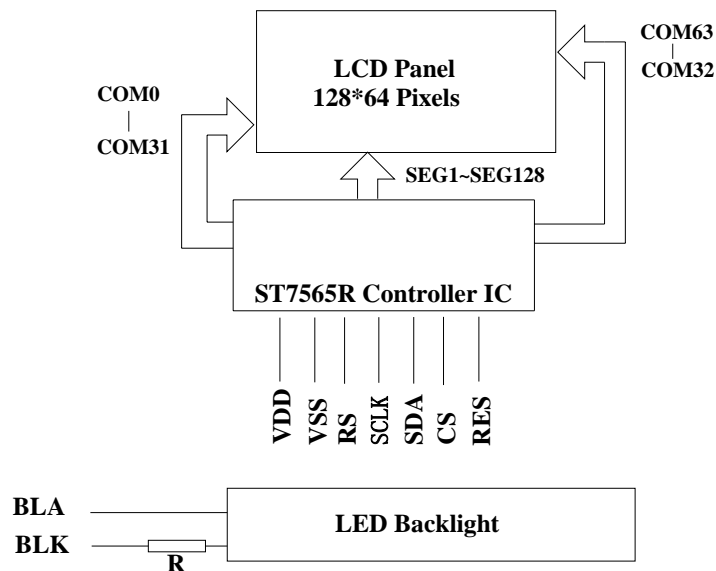


图 2: JLX12864G-99-2 图像点阵型液晶模块的电路框图

接口方面, 有 5 条数据及控制线。可与微处理器或微控制器相连, 通过送入数据和指令, 就可使模

块正常工作.

4.3 背光参数

该型号液晶模块带 LED 背光源。它的性能参数如下:

工作温度: $-20\sim+70^{\circ}\text{C}$;

存储温度: $-30\sim+80^{\circ}\text{C}$;

背光板可显示绿色, 黄绿色, 兰色和白色。背光一般为绿色, 也可为客户设计为其他颜色, 但价格较绿色贵一点。

正常工作电流为: $10\sim 20\text{mA}$ (若 LED 灯数不止一颗, 则乘以相应数量);

工作电压: 3.0V ;

正常工作条件下, LED 可连续点亮 5 万小时;

5. 技术参数

5.1 最大极限参数 (超过极限参数则会损坏液晶模块)

名称	符号	标准值			单位
		最小	典型	最大	
电路电源	VDD - VSS	-0.3		7.0	V
LCD 驱动电压	VDD - VO	VDD - 13.5		VDD + 0.3	V
静电电压		-	-	100	V
工作温度		-20		+70	$^{\circ}\text{C}$
储存温度		-30		+80	$^{\circ}\text{C}$

表 2: 最大极限参数

5.2 直流 (DC) 参数

名称	符号	测试条件	标准值			单位
			MIN	TYPE	MAX	
工作电压	VIN	3.3V 供电	2.4	3.3	3.6	V
		5.0V 供电	4.0	5.0	5.2	V
输入高电平	VIH	-	2.2		VDD	V
输入低电平	VIO	-	-0.3		0.6	V
输出高电平	VOH	IOH = 0.2mA	2.4		-	V
输出低电平	VOO	I00 = 1.2mA	-		0.4	V
工作电流	IDD	VDD = 5.0V		2.0		mA

表 3: 直流 (DC) 参数

6. 读写时序特性

6.1 从 CPU 写到 ST7565R (Writing Data from CPU to ST7565R)

The 4-line SPI Interface

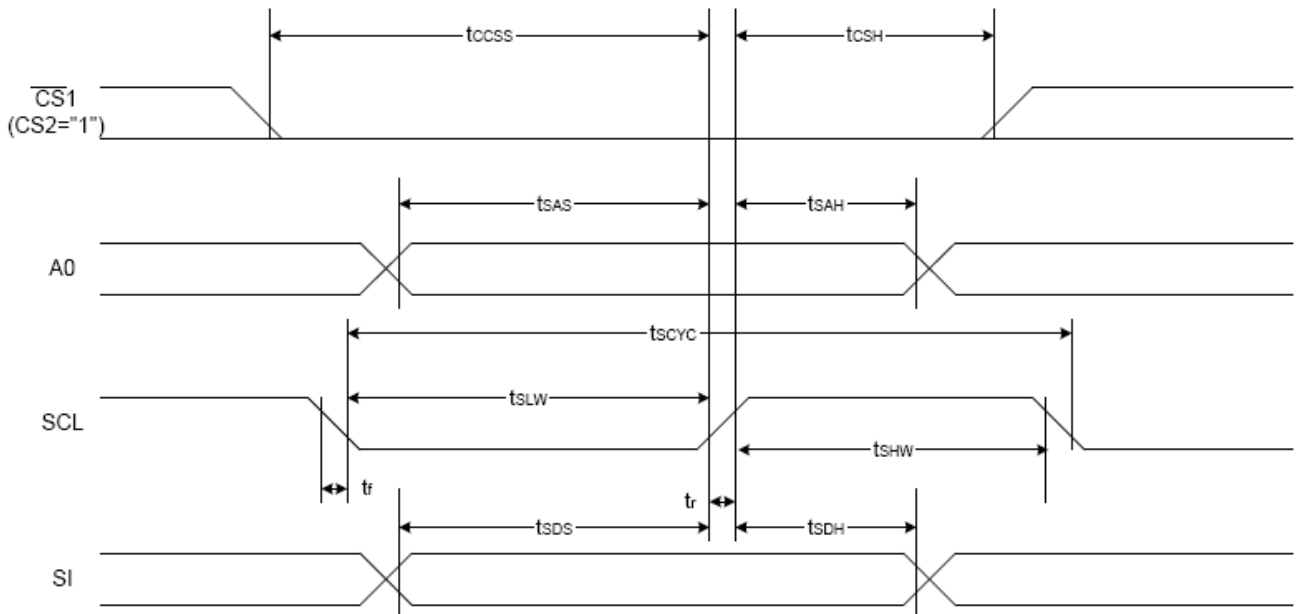


图 3. 从 CPU 写到 ST7565R (Writing Data from CPU to ST7565R)

6.2 时序要求 (AC 参数):

写数据到 ST7565R 的时序要求:

表 4.

项目	符号	测试条件	极限值			单位
			MIN	TYPE	MAX	
4线 SPI串口时钟周期 (4-line SPI Clock Period)	T_{scyc}	引脚: SCK	50	--	25	ns
保持SCK高电平脉宽 (SCK "H" pulse width)	T_{shw}	引脚: SCK	25			ns
保持SCK低电平脉宽 (SCK "L" pulse width)	T_{slw}	引脚: SCK	25			ns
地址建立时间 (Address setup time)	T_{sas}	引脚: RS	20	--	--	ns
地址保持时间 (Address hold time)	T_{sah}	引脚: RS	10	--	--	ns
数据建立时间 (Data setup time)	T_{sds}	引脚: SI	20	--	--	ns
数据保持时间 (Data hold time)	T_{sdh}	引脚: SI	10	--	--	ns
片选信号建立时间 (CS-SCL time)	T_{css}	引脚: CS	20			ns
片选信号保持时间 (CS-SCL time)	T_{csh}	引脚: CS	40			ns

VDD = 3.0V ± 5%, Ta = 25°C

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

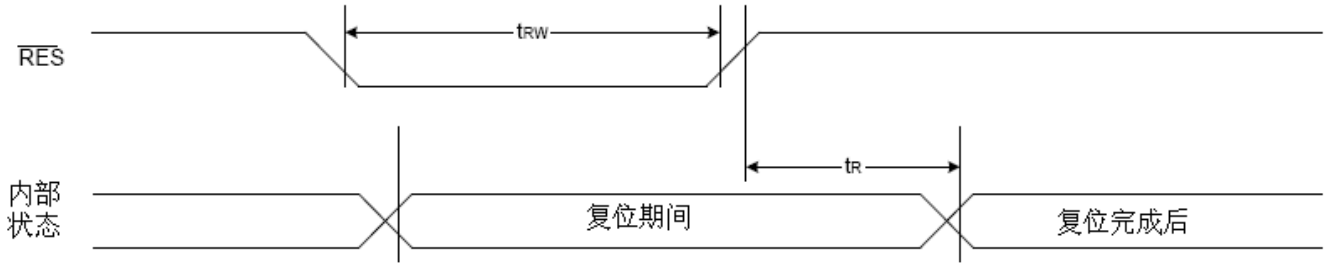


图 5: 电源启动后复位的时序

表 6: 电源启动后复位的时序要求

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

7. 指令功能:

7.1 指令表

格式:

RS	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
----	-----	-----	-----	-----	-----	-----	-----	-----

共 11 种指令: 1. 清除, 2. 返回, 3. 输入方式设置, 4. 显示开关, 5. 控制, 移位, 6. 功能设置, 7. CGRAM 地址设置, 8. DDRAM 地址设置, 9. 读忙标志, 10. 写数据到 CG/DDRAM, 11. 读数据由 CG/DDRAM。

指令表

表 8.

指令名称	指令码									说明
	RS	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	
(1) 显示开/关 (display on/off)	0	1	0	1	0	1	1	1	0 1	显示开/关: 0: 关, 1: 开
(2) 显示初始行设置 (Display start line set)	0	0	1	显示初始行地址, 共 5 位						设置显示存储器的显示初始行
(3) 页地址设置 (Page address set)	0	1	0	1	1	显示页地址, 共 4 位				设置显示页地址 (注: 每 8 行为一个页, 64 行分为 8 个页, 例 0000 为第一页, 0001 为第二页)
(4) 列地址高 4 位设置	0	0	0	0	1	列地址的高 4 位				高 4 位与低 4 位共同组成列地址, 分别指定 128 列中任对列。本液

	列地址低4位设置		0	0	0	0	列地址的低 4 位			晶模块的第一列的地址为00000001, 所以此指令表达为: 0x10, 0x01	
(5)	读状态 (Status read)	0	状态				0	0	0	0	在本型号液晶模块不用此指令
(6)	写数据 (Display data write)	1	8 位显示数据								从 CPU 写数据到液晶模块
(7)	读数据 (Display data read)	1	8 位显示数据								在本型号液晶模块不用此指令
(8)	显示列地址增减 (ADC select)		1	0	1	0	0	0	0	0 1	显示列地址增减: 0: 常规: 从左到右, 1: 反转: 从右到左
(9)	显示正显/反显 (Display normal/reverse)	0	1	0	1	0	0	1	1	0 1	显示正显/反显: 0: 常规: 正显 1: 反显
(10)	显示全部点阵 (Display all points)	0	1	0	1	0	0	1	0	0 1	显示全部点阵: 0: 常规 1: 显示全部点阵
(11)	LCD 偏压比设置 (LCD bias set)	0	1	0	1	0	0	0	1	0 1	设置偏压比: 0: 1/9 BIAS 1: 1/7BIAS
(12)	Read-modify-write	0	1	1	1	0	0	0	0	0	Column address increment At write: +1 At read: 0
13)	退出上述指令 (End)	0	1	1	1	0	1	1	1	0	退出上述 “read/modify/write” 指令
(14)	软件复位 (Reset)	0	1	1	1	0	0	0	1	0	软件复位。
(15)	行扫描顺序选择 (Common output mode select)		1	1	0	0	0	0	0	0	行扫描顺序选择: 0: 普通顺序 1: 反向扫描
(16)	电源控制 (Power control set)		0	0	1	0	1	电压操作模式选择, 共 3 位			选择内部电压供应操作模式
(17)	选择内部电阻比例	0	0	0	1	0	0	内部电压值电阻设置			选择内部电阻比例 (Rb/Ra), 本液晶模块通过外置电阻设置, 此指令失效
(18)	内部设置液晶电压模式	0	1	0	0	0	0	0	0	1	设置内部电阻微调, 以设置液晶电压, 此两个指令需紧接着使用
	设置的电压值		0	0	6 位电压值数据, 0~63 共 64 级						
(19)	静态图标显示: 开/关	0	1	0	1	0	1	1	0	0 1	0: 关, 1: 开。本液晶屏无此图标, 所以此指令无效

(20) 升压倍数选择 (Booster ratio set)	0	1	1	1	1	1	0	0	0	2位数设置 升压倍数	选择升压倍数: 00: 2倍, 3倍, 4倍 01: 5倍 11: 6倍。本模块外部已设置升压倍数为4倍, 不必使用此指令
		0	0	0	0	0	0	0	0		
(21) 省电模式 (Power save)											省电模式, 此非一条指令, 是由“(10)显示全部点阵”、(19)静态图标显示: 开/关等指令合成一个“省电功能”。
(22) 空指令 (NOP)	0	1	1	1	0	0	0	1	1		空操作
(23) 测试 (Test)	0	1	1	1	1	*	*	*	*		内部测试用, 千万别用!

请详细参考 IC 资料”ST7564R_V15.PDF”的第 42~49 页。

7.3 点阵与 DD RAM 地址的对应关系

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

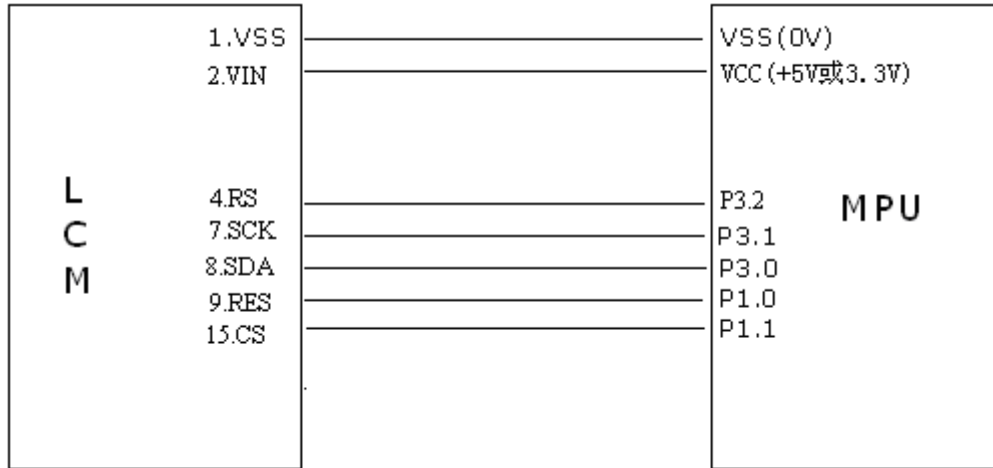
DB7--DB0 的排列方向: 数据是从下向上排列的。最低位 D0 是在最上面, 最高位 D7 是在最下面。下图摘自 ST7565R IC 资料, 可通过“ST7565R_V15. PDF”之第 27 页获取最佳效果。

7.4 初始化方法

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

7.5 程序举例:

液晶模块与 MPU(以 8051 系列单片机为例)接口图如下:



```

INITIAL:                ;初始化子程序
    MOV    A, #0E2H      ;软件复位
    LCALL  WCOM          ;调用写指令子程序 (call Write command sub-program), 以下相同
    LCALL  DELAY
;-----
    MOV    A, #02CH      ;设置内部电压模式, X0, X1, X2=1, 0, 0
    LCALL  WCOM          ;
    LCALL  DELAY
;-----
    MOV    A, #02EH      ;设置内部电压模式, X0, X1, X2=1, 1, 0
    LCALL  WCOM          ;
    LCALL  DELAY
;-----
    MOV    A, #02FH      ;设置内部电压模式, X0, X1, X2=1, 1, 1, 以上#02CH\#02EH\#02FH;
                        ;是内部升压电路的三步曲
    LCALL  WCOM
    LCALL  DELAY
;-----
    MOV    A, #023H      ;设置内部调压电阻值, 通常用#023H
    LCALL  WCOM
;-----
    MOV    A, #0A2H      ;设置 LCD 偏压比 (Bias) : 1/9
    LCALL  WCOM
;-----
    MOV    A, #081H      ;设置液晶驱动电压微调 V0 值
    LCALL  WCOM
    
```

```

MOV      A, #02DH      ;设置液晶驱动电压微调 V0 值, 与上述#081H 指令共同使用。但电
                        ;压值由此指令决定
LCALL    WCOM
;-----
MOV      A, #0C8h      ;设置行扫描顺序为从上到下扫描
LCALL    WCOM      ;
;-----
MOV      A, #0A0H      ;设置列扫描顺序为从左到右
LCALL    WCOM
;-----
MOV      A, #060H      ;设置起始行位置, 从第 1 行开始
LCALL    WCOM
;-----
MOV      A, #0AFH      ;打开显示
LCALL    WCOM
RET
;***写指令子程序*****
WCOM:
    CLR      CS      ;片选
    CLR      RS      ;寄存器选择为: 指令
    JMP      TRANSMIT
;***写数据子程序*****
WDATA:
    CLR      CS      ;片选
    SETB     RS      ;寄存器选择为: 数据
;-----
TRANSMIT:
    CLR      SCLK     ;时钟
    MOV      44H, #08
TRAN1:
    RLC      A      ;左移一位数据
    MOV      SDI, C      ;左移到进位标志位 CY, 被提取到 SDI 引线
    SETB     SCLK     ;时钟上升沿锁存数据
    CLR      SCLK     ;时钟回复到低电平, 为下一次时钟作准备
    NOP
    NOP
    NOP
    NOP
    DJNZ     44H, TRAN1
    RET
;-----

```

C 语言的程序:

```

void Initial_Lcd()
{
    reset=0;                //Reset the chip when reset=0
}

```

```
    Delay(200);
    reset=1;
    Delay(200);
    Transfer_command(0xe2); //软件复位
    Delay(10);
    Transfer_command(0x2c); //设置内部电压模式, X0, X1, X2=1, 0, 0
    Delay(10);
    Transfer_command(0x2e); //设置内部电压模式, X0, X1, X2=1, 1, 0
    Delay(10);
    Transfer_command(0x2f); //设置内部电压模式, X0, X1, X2=1, 1, 1, 以上
                            // #02CH\#02EH\#02FH 是内部升压电路的三步曲

    Delay(10);
    Transfer_command(0x23); //设置内部调压电阻值, 通常用#023H
    Transfer_command(0x81); //设置液晶驱动电压微调 V0 值
    Transfer_command(0x2d); //;设置液晶驱动电压微调 V0 值, 与上述#081H 指令共同使用。
                            //但电压值由此指令决定

    Transfer_command(0xa2); //设置 LCD 偏压比 (Bias) : 1/9
    Transfer_command(0xc8); //设置行扫描顺序为从上到下扫描
    Transfer_command(0xa0); //设置列扫描顺序为从左到右
    Transfer_command(0x60); //设置起始行位置, 从第 1 行开始
    Transfer_command(0xaf); //打开显示
}

//=====传送指令到液晶模块=====
void Transfer_command(int data1)
{
    char i;
    cs1=0;
    rs=0;
    for(i=0;i<8;i++)
    {
        sclk=0;
        if(data1&0x80) sid=1;
        else sid=0;
        Delay1(5);
        sclk=1;
        Delay1(5);
        data1=data1<<=1;
    }
}
```

```
//-----传送数据到液晶模块-----
```

```
void Transfer_data(int data1)
{
    char i;
    cs1=0;
    rs=1;
    for(i=0;i<8;i++)
        {
            sclk=0;
            if(data1&0x80) sid=1;
            else sid=0;
            sclk=1;
            data1=data1<<=1;
        }
}
```