

ST7920 控制器液晶模块使用说明书

感谢您关注和使用我们的液晶产品。如果您在使用中有任何疑问，请拨打我们的客户服务热线 **075527351404** 寻求技术支持和获取相关资料，我们竭诚为您服务。您可以登录我们的网站了解最新产品信息。或者您可以在我公司网站的留言簿栏目留下您宝贵的意见。

深圳市金马电子有限公司

邮 编： 518108

业务联系： 13686451718

技术支持： 13510654866

传 真： 0755-27351404

地址：深圳市宝安区石岩水田长城工业区兰姜十区

中文字库液晶显示模块使用手册

JN12864J

一、液晶显示模块概述

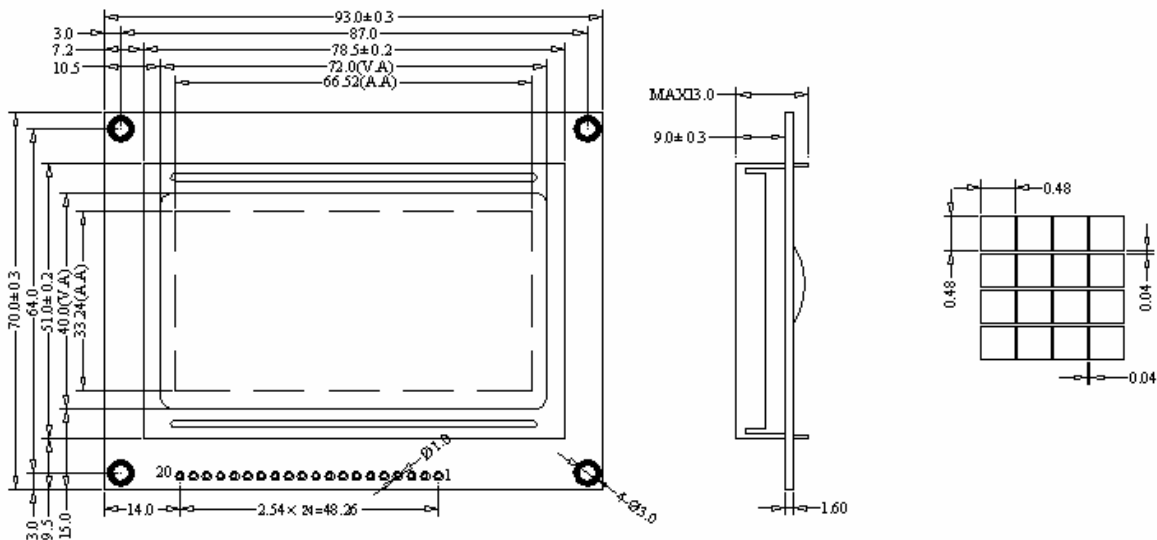
1. 硬件特性如下:

- 提供 8 位, 4 位并行接口及串行接口可选
- 并行接口适配 M6800 时序
- 自动电源启动复位功能
- 内部自建振荡源
- 64 × 16 位字符显示 RAM (DDRAM 最多 16 字符 × 4 行, LCD 显示范围 16 × 2 行)
- 2M 位中文字型 ROM (CGROM), 总共提供 8192 个中文字型 (16 × 16 点阵)
- 16K 位半宽字型 ROM (HCGROM), 总共提供 126 个西文字型 (16 × 8 点阵)
- 64 × 16 位字符产生 RAM (CGRAM)
- 15 × 16 位总共 240 点的 ICON RAM (ICONRAM)

2. 软件特性如下:

- 文字与图形混合显示功能
- 画面清除功能
- 光标归位功能
- 显示开/关功能
- 光标显示/隐藏功能
- 显示字体闪烁功能
- 光标移位功能功能
- 显示移位功能
- 垂直画面旋转功能
- 反白显示功能
- 休眠模式

3. 外形尺寸图:



对上图外形尺寸说明:

ITEM	NOMINAL DIMEN	UNIT
模块体积	93.0×70.0×13.0	mm
视域 (VA)	72.0×40.0	mm
实际显示区 (AA)	66.52×33.24	mm
行列点阵数	128×64	dots
点距离	0.52×0.52	mm
点大小	0.48×0.48	mm
定位孔	87×64 (4-◎3)	mm

二、模块引脚

JN128X64J 引脚定义:

管脚序	名称	型态	电平	功能描述	
				并口	串口
1	VSS	I	—	电源地	
2	VDD	I	—	电源输入 (+5V)	
3	VO	I	—	空脚 (NC)	
4	RS (CS)	I	H/L	寄存器选择端: H 数据; L 指令	片选, 低有效
5	R/W (SID)	I	H/L	H: 读 L: 写	串行数据线
6	E (SCLK)	I	H/L		串行时钟输入
7-10	DB0~DB3	I/O	H/L	数据总线低四位	
11-14	DB4~DB7	I/O	H/L	数据总线高四位, 4 位并口时空接	
15	PSB	I	H/L	并口/串口选择: H 并口	L 串口
16	NC	I		空脚 (NC)	
17	/RST	I		复位信号, 低电平有效	
18	VEE (Vout)	I	—	此模块此脚为空脚 (NC)	
19	BLA		—	背光负极 (LEDA、BLA)	
20	BLK		—	A/K可调换 背光正极 (LEDK、BLK)	

三、液晶硬件接口

- 1、逻辑工作电压 (VDD): 4.5V~5.5V 或 3.0V~3.3V
- 2、电源地 (GND): 0V
- 3、工作温度 60°C (常温) / -20~75°C (宽温)

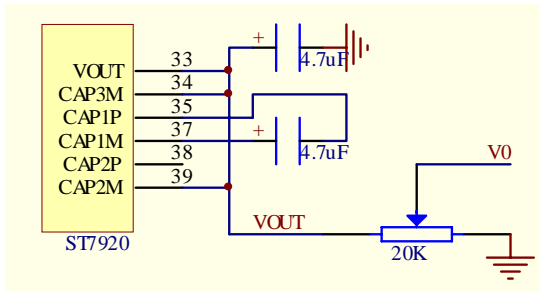
电气特性: 测试条件 Ta=25, Vdd=5.0±10%

- 1) 输入高电平 (Vih): 0.7Vdd~Vdd
- 2) 输入低电平 (Vil): 0.6Vmax
- 3) 输出高电平 (Voh): 0.8Vdd~Vdd
- 4) 输出低电平 (Vol): 0.4Vmax
- 5) 模块工作电流: 1.14~1.18mA (不含背光)
- 6) 侧白光工作电流: MAX≥80mA
- 7) 底黄绿光工作电流: MAX≥300Ma
- 8) 模块背光脚正负极可以任意选择

LCD 驱动电压即对比度调节电路

ST7920 内带倍压电路，生成 2 倍于 VCC 的电压。倍压通过 Vout 脚引出，通过电位器调节后，从 V0 引回模块用来驱动 LCD。直接驱动 LCD 的是 V0，V0 电压越高，对比度越深。通过调节电位器来调节 V0 值以改变对比度。

JN12864J 内置电位器，倍压后的电压由 VOUT 端引出，通过下图 VR (20K) 调节适合 LCD 驱动电压到 V0 端。此时，LCM 接口处 VO 和 VEE 为空脚，如果需要调节对比度调节电位器就可以达到所需对比度效果



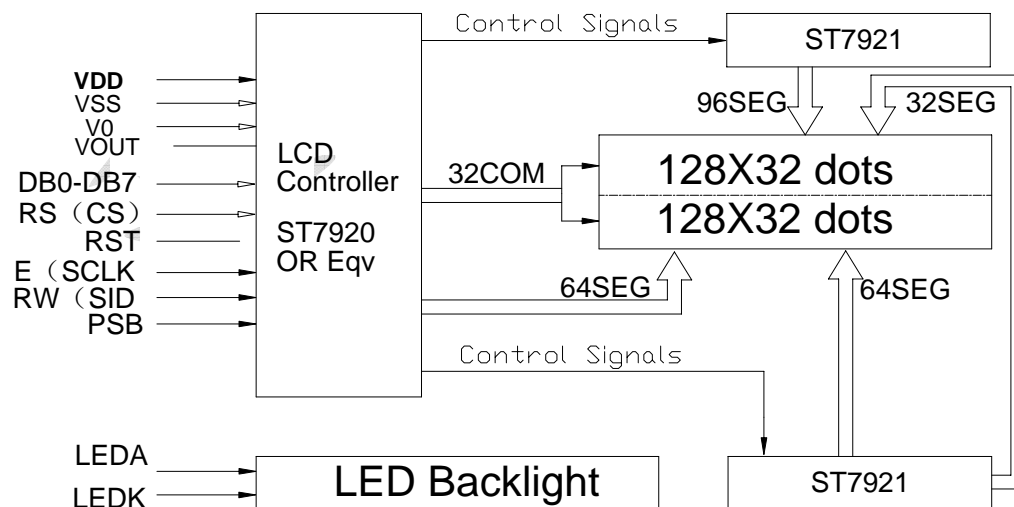
温度补偿

JN12864J 可匹配温度补偿电路，温度补偿范围为-7°C~+7°C。

PSB 电路

PSB 接高时选择并口，接低时选择串口。模块上一般都有跳线方式将 PSB 接高或接低，用户可以不再处理 PSB 脚（用户事先需声明是用并口还是串口）。也可以选择不在模块上处理，而由客户自己选择并口还是串口。

原理简图



四. ST7920 内置硬件说明

中文字型产生 ROM (CGROM) 及半宽字型 ROM(HCGROM)

ST7920 的字型产生 ROM 通过 8192 个 16×16 点阵的中文字型，以及 126 个 16×8 点阵的西文字符，它用 2 个字节来提供编码选择，将要显示的字符的编码写到 DDRAM 上，硬件将依照编码自动从 CGROM 中选择将要显示的字型显示再屏幕上。

字型产生 RAM(CGRAM)

ST7920 的字型产生 RAM 提供用户自定义字符生成（造字）功能，可提供 4 组 16×16 点阵的空间，用户可以将 CGROM 中没有的字符定义到 CGRAM 中。

显示 RAM(DDRAM)

显示 RAM 提供 64×2 字节的空间，最多可以控制 4 行 16 字的中文字型显示。当写入显示资料 RAM 时，可以分别显示 CGROM，HCGROM 及 CGRAM 的字型。

三种字型的选择：

- 1) 显示半宽字型 将一个字节的编码写入 DDRAM 中，范围是 02~7FH
- 2) 显示 CGRAM 字型 将 2 个字节的编码写入 DDRAM 中，共有 0000H，0002H，0004H 及 0006H 四种编码
- 3) 显示中文字型 将 2 字节的编码写入 DDRAM 中，先写高 8 位，后写低 8 位范围是 A140H~D75FH(BIG5), A1A0H~F7FFH(GB)

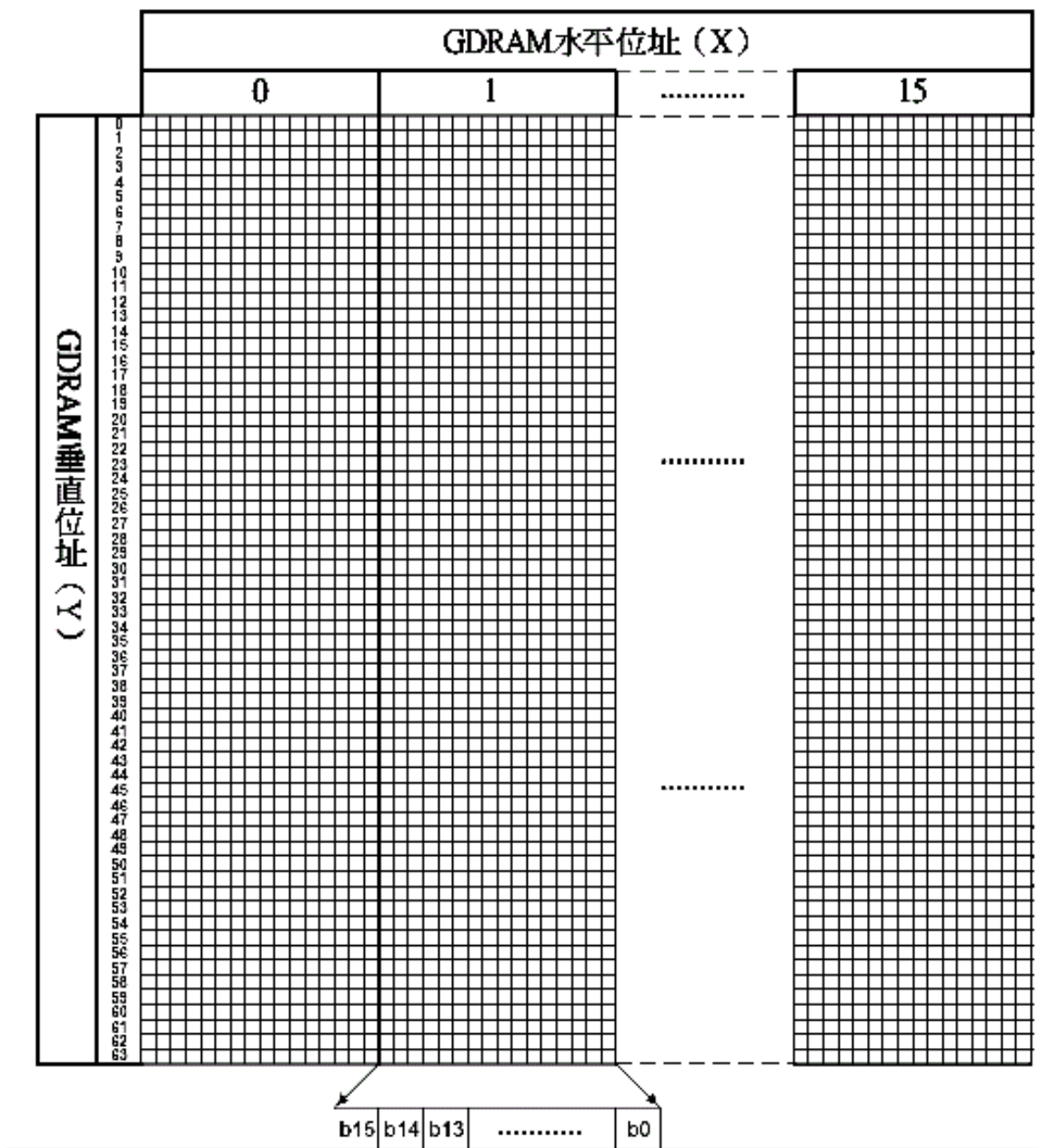
ICON RAM 位址 在擴充指令集將 SR 設 為“0”,再利用設定 IRAM 位 址指令來設定 AC3...AC0				ICON RAM 資料															
				高位元組								低位元組							
AC3	AC2	AC1	AC0	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	SEG0	SEG1	SEG2	SEG3	SEG4	SEG5	SEG6	SEG7	SEG8	SEG9	SEG10	SEG11	SEG12	SEG13	SEG14	SEG15
0	0	0	1	SEG16	SEG17	SEG18	SEG19	SEG20	SEG21	SEG22	SEG23	SEG24	SEG25	SEG26	SEG27	SEG28	SEG29	SEG30	SEG31
0	0	1	0	SEG32	SEG33	SEG34	SEG35	SEG36	SEG37	SEG38	SEG39	SEG40	SEG41	SEG42	SEG43	SEG44	SEG45	SEG46	SEG47
0	0	1	1	SEG48	SEG49	SEG50	SEG51	SEG52	SEG53	SEG54	SEG55	SEG56	SEG57	SEG58	SEG59	SEG60	SEG61	SEG62	SEG63
0	1	0	0	SEG64	SEG65	SEG66	SEG67	SEG68	SEG69	SEG70	SEG71	SEG72	SEG73	SEG74	SEG75	SEG76	SEG77	SEG78	SEG79
0	1	0	1	SEG80	SEG81	SEG82	SEG83	SEG84	SEG85	SEG86	SEG87	SEG88	SEG89	SEG90	SEG91	SEG92	SEG93	SEG94	SEG95
0	1	1	0	SEG96	SEG97	SEG98	SEG99	SEG100	SEG101	SEG102	SEG103	SEG104	SEG105	SEG106	SEG107	SEG108	SEG109	SEG110	SEG111
0	1	1	1	SEG112	SEG113	SEG114	SEG115	SEG116	SEG117	SEG118	SEG119	SEG120	SEG121	SEG122	SEG123	SEG124	SEG125	SEG126	SEG127
1	0	0	0	SEG128	SEG129	SEG130	SEG131	SEG132	SEG133	SEG134	SEG135	SEG136	SEG137	SEG138	SEG139	SEG140	SEG141	SEG142	SEG143
1	0	0	1	SEG144	SEG145	SEG146	SEG147	SEG148	SEG149	SEG150	SEG151	SEG152	SEG153	SEG154	SEG155	SEG156	SEG157	SEG158	SEG159
1	0	1	0	SEG160	SEG161	SEG162	SEG163	SEG164	SEG165	SEG166	SEG167	SEG168	SEG169	SEG170	SEG171	SEG172	SEG173	SEG174	SEG175
1	0	1	1	SEG176	SEG177	SEG178	SEG179	SEG180	SEG181	SEG182	SEG183	SEG184	SEG185	SEG186	SEG187	SEG188	SEG189	SEG190	SEG191
1	1	0	0	SEG192	SEG193	SEG194	SEG195	SEG196	SEG197	SEG198	SEG199	SEG200	SEG201	SEG202	SEG203	SEG204	SEG205	SEG206	SEG207
1	1	0	1	SEG208	SEG209	SEG210	SEG211	SEG212	SEG213	SEG214	SEG215	SEG216	SEG217	SEG218	SEG219	SEG220	SEG221	SEG222	SEG223
1	1	1	0	SEG224	SEG225	SEG226	SEG227	SEG228	SEG229	SEG230	SEG231	SEG232	SEG233	SEG234	SEG235	SEG236	SEG237	SEG238	SEG239
1	1	1	1	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

ICON RAM(IRAM)

ST7920 提供 240 点的 ICON 显示，它由 15 个 IRAM 单元组成，每个单元有 16 位，每写入一组 IRAM 时，需先写入 IRAM 地址，然后连续送入 2 个字节的数据，先高 8 位 (D15~D8)，后低 8 位 (D7~D0)。

绘图 RAM

提供 64×32 个字节的存储空间（由扩充指令设定绘图 RAM 地址），最多可以控制 256×64 点阵的二维绘图缓冲空间，在更改绘图 RAM 是，由扩充指令设置 GDRAM 地址先垂直地址后水平地址（连续 2 个字节的数据来定义垂直和水平地址），再 2 个字节的数据给绘图 RAM（先高 8 位后低 8 位）。



DDRAM 内容, CGRAM 地址以及 CGRAM 内容的对照关系

DDRAM 資料 (字元代碼)				CGRAM 位址				CGRAM 資料 (高位元組)				CGRAM 資料 (低位元組)																			
B15~ B4				B	B	B	B	D	D	D	D	D	D	D	D																
	3	2	1	0	5	4	3	2	1	0	1	1	1	1	1	1	9	8	7	6	5	4	3	2	1	0					
0	X	00	X	00	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0		
					0	0	0	1	1	1	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
					0	0	1	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	0
					0	0	1	1	0	0	0	1	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0
					0	1	0	0	0	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0	0	1	0	0	0	0
					0	1	0	1	0	0	1	1	1	1	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0
					0	1	1	0	0	1	1	0	0	1	0	1	0	1	0	1	0	1	0	0	1	0	0	0	0	0	0
					0	1	1	1	1	0	1	0	0	1	1	0	0	1	1	0	0	1	0	0	1	0	0	1	0	0	0
					1	0	0	0	0	0	1	0	0	1	0	0	0	0	1	0	1	0	0	0	1	0	0	0	0	0	0
					1	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
					1	0	1	0	0	0	1	0	0	1	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0
					1	0	1	1	0	0	1	1	1	1	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0
					1	1	0	0	0	0	1	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0
					1	1	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0
					1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
					0	X	01	X	01	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	1	1
0	0	0	1	0						0	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
0	0	1	0	0						0	1	0	0	0	0	0	0	1	0	0	1	1	0	1	0	0	1	0	0	0	
0	0	1	1	0						1	0	1	1	1	0	1	1	0	1	1	0	1	0	0	1	0	0	1	0	0	0
0	1	0	0	1						0	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	0	1	0	0	0	
0	1	0	1	0						1	1	1	1	1	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0	0	0
0	1	1	0	0						1	0	0	0	0	0	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0
0	1	1	1	0						1	1	1	1	1	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0
1	0	0	0	0						1	0	0	0	0	0	0	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0
1	0	0	1	0						1	1	1	1	1	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0
1	0	1	0	0						1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
1	0	1	1	0						1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	0	0	1						0	1	0	0	0	0	0	0	1	0	1	0	0	1	0	0	1	0	0	1	0	0
1	1	0	1	1						0	1	1	1	1	1	1	0	0	1	1	1	0	0	1	1	1	0	0	0	0	0
1	1	1	0	1						0	1	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0	0
1	1	1	1	0						0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

CGRAM 与中文字型的编码只能出现在 adress counter 的起始位置 (见下表)

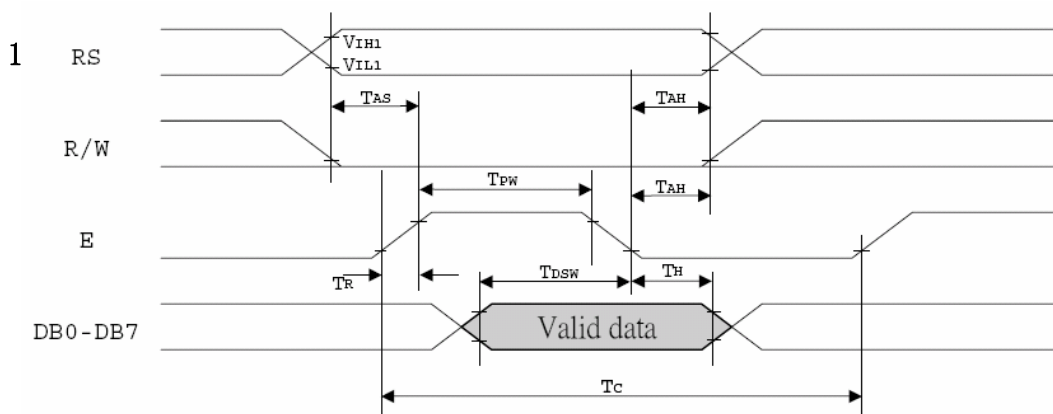
80	81	82	83	84	85	86	87	88	89	8A	8B	8C
----	----	----	----	----	----	----	----	----	----	----	----	----

H	L	H	L	H	L	H	L	H	L	H	L	H	L	H	L	H	L	H	L	H	L	H	L	
		1	2	2	3	2	Z			1	2	8	6	4	-	Z	A		1	6	0	3	2	A
金	马	液	晶	A	B	1	2	中	文	(正	确)										
金	马	液	晶		...AB12中	文	(错	误)														

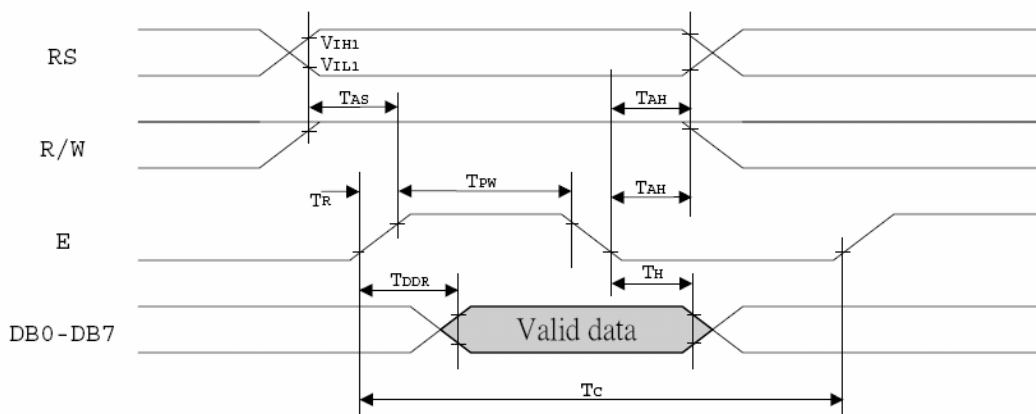
16×8 半宽字型表

☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	♂	♀	♫	♫	✳
▶	◀	↑	!!	☒	☒	☒	↑	↑	↓	→	←	↵	↵	▲	▼
☐	!	"	#	\$	%	&	'	()	*+	,	-	.	/	
0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
Q	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
p	q	r	s	t	u	v	w	x	y	z	{	!	}	~	△

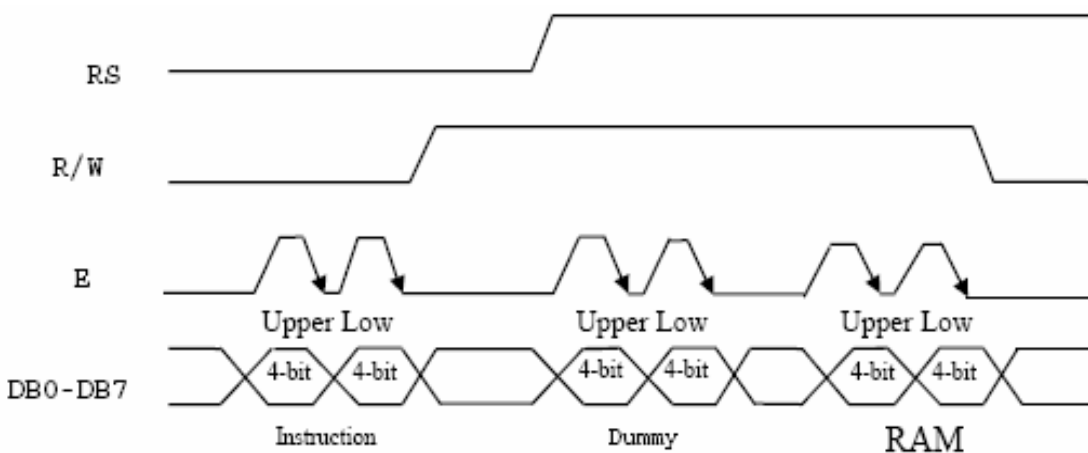
时序



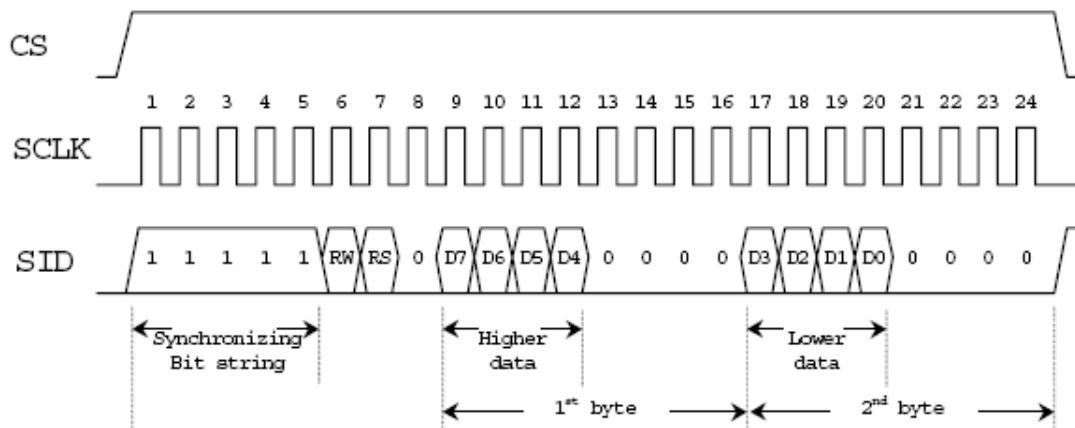
2. 8 位并口读操作时序图



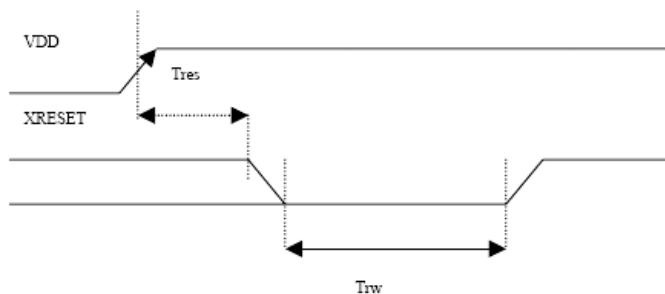
3. 4 位并口时序图图



4. 串口时序图



5. 外部复位时序图



XRESET pulse width	Trw	10us
RESET start time	Tres	50ns

指令说明

1.指令表 1 (RE=0,基本指令集)

指令名称	控制信号		控制代码								执行时间	
	RS	R/W	D7	D6	D5	D4	D3	D2	D1	D0		
清除显示	0	0	0	0	0	0	0	0	0	1	1.6 ms	
地址归0	0	0	0	0	0	0	0	0	0	1	72us	
进入设定点	0	0	0	0	0	0	0	0	1	I/D	72us	
显示开关设置	0	0	0	0	0	0	0	1	D	C	B	72us
移位控制	0	0	0	0	0	0	1	S/C	R/L	X	X	72us
功能设定	0	0	0	0	0	1	DL	X	0/RE	X	X	72us
设定 CGRAM 地址	0	0	0	1	A5	A4	A3	A2	A1	A0	72us	
设定 DDRAM 地址	0	0	1	0	A5	A4	A3	A2	A1	A0	72us	
读忙标志和地址	0	1	BF	A6	A5	A4	A3	A2	A1	A0	72us	
写显示数据	1	0	显示数据								72us	
读显示数据	1	1	显示数据								72us	

1. 指令表 2 (RE=1, 扩充指令集)

指令名称	控制信号		控制代码								执行时间
	RS	R/W	D7	D6	D5	D4	D3	D2	D1	D0	
待命模式	0	0	0	0	0	0	0	0	0	1	72 μ s
滚动地址或 RAM 地址选择	0	0	0	0	0	0	0	0	1	SR	72 μ s
反白显示	0	0	0	0	0	0	0	1	R1	R0	72 μ s
睡眠模式	0	0	0	0	0	0	1	SL	X	X	72 μ s
扩充功能设定	0	0	0	0	1	DL	X	1/RE	G	0	72 μ s
设定 IRAM 地址或滚动地址	0	0	0	1	A5	A4	A3	A2	A1	A0	72 μ s
设定绘图 RAM 地址	0	0	1	0	0	0	A3	A2	A1	A0	72 μ s
				A6	A5	A4	A3	A2	A1	A0	

备注：当 ST7920 在接受指令前，MCU 必须先确认 ST7920 处于非忙状态。即读取 BF = 0，才能接受新的指令；如果在送出一条指令前不检查 BF 状态，则需要延时一段时间，以确保上一条指令执行完毕，具体指令执行时间参照指令表。

“RE” 是基本指令集与扩充指令集的选择控制位，当变更 “RE” 的状态后，以后的指令维持在最后的状态。除非再次变更 “RE” 的状态，否则使用相同的指令集时，不需要重新设置 “RE”。

2. 基本指令详细说明表

1) 清除显示 (CLEAR)

格式

0	0	0	0	0	0	0	1
---	---	---	---	---	---	---	---

将 DDRAM 填满 “20H” (空格) 代码，并且设定 DDRAM 的地址计数器 (AC) 为 00H；更新设置进入设定点将 I/D 设为 1，游标右移 AC 加 1。

2) 地址归 0 (HOME)

格式

0	0	0	0	0	0	1	X
---	---	---	---	---	---	---	---

设定 DDRAM 的地址寄存器为 00H，并且将游标移到开头原点位置；这个指令并不改变 DDRAM 的内容。

3) 进入设定点 (ENTRY MODE SET) 初始值：06H

格式

0	0	0	0	0	1	I/D	S
---	---	---	---	---	---	-----	---

指定在显示数据的读取与写入时，设定游标的移动方向及指定显示的移位

I/D=1，游标右移，DDRAM 地址计数器 (AC) 加 1

I/D=0，游标左移，DDRAM 地址计数器 (AC) 减 1

S: 显示画面整体位移

S	I/D	功能描述
H	H	画面整体左移
H	L	画面整体右移

4) 显示开关设置 (DISPLAY STATUS) 初始值: 08H

格式	0	0	0	0	1	D	C	B
----	---	---	---	---	---	---	---	---

控制整体显示开关, 游标开关, 游标位置显示反白开关

D=1, 整体显示开; D=0, 整体显示关, 但是不改变 DDRAM 内容

C=1, 游标显示开; C=0, 游标显示关

B=1, 游标位置显示反白开, 将游标所在地址上的内容反白显示; B=0, 正常显示

5) 游标或显示移位控制(CURSOR AND DISPLAY SHIFT CONTROL) 初始值: 0001 XXXX B (X=0,1)

格式	0	0	0	1	S/C	R/L	X	X
----	---	---	---	---	-----	-----	---	---

这条指令不改变 DDRAM 的内容

S/C	R/L	方向	AC 的值
L	L	游标向左移动	AC=AC-1
L	H	游标向右移动	AC=AC+1
H	L	显示向左移动, 游标跟着移动	AC=AC
H	H	显示向右移动, 游标跟着移动	AC=AC

6) 功能设定(FUNCTION SET) 初始值: 0011 X0XX B (X=0,1)

格式	0	0	1	DL	X	0/RE	X	X
----	---	---	---	----	---	------	---	---

DL: 8/4 位接口控制位

DL=1, 8 位 MPU 接口; DL=0, 4 位 MPU 接口

RE: 指令集选择控制位

RE=1, 扩充指令集; RE=0, 基本指令集

同一指令的动作不能同时改变 DL 和 RE, 需先改变 DL 再改变 RE 才能确保设置正确

7) 设定 CGRAM 地址

格式	0	1	A5	A4	A3	A2	A1	A0
----	---	---	----	----	----	----	----	----

设定 CGRAM 地址到地址计数器 (AC), AC 范围为 00H~3FH 需确认扩充指令中 SR=0 (卷动位置或 RAM 地址选择)

8) 设定 DDRAM

格式	1	0	A5	A4	A3	A2	A1	A0
----	---	---	----	----	----	----	----	----

设定 DDRAM 地址到地址计数器 (AC)

第一行 AC 范围 80H~8FH

第二行 AC 范围 90H~9FH

备注: ST7920 控制器的 128 × 64 点阵液晶其实原理上等同 256 × 32 点阵, 第三行对应的 DDRAM 地址紧接第一行; 第四行对应的 DDRAM 地址紧接第二行。

用户在使用行反白功能时, 如果第一行反白, 第三行必然反白。第二行反白, 第四行必然反白。这是正常现象。

9) 读取忙标志和地址(RS=0,R/W=1)

格式	BF	A6	A5	A4	A3	A2	A1	A0
----	----	----	----	----	----	----	----	----

读取忙标志以确定内部动作是否完成, 同时可以读出地址计数器 (AC) 的值

10) 写显示数据到 RAM(RS=1,R/W=0)

格式

D7	D6	D5	D4	D3	D2	D1	D0
----	----	----	----	----	----	----	----

当显示数据写入后会使得 AC 改变，每个 RAM (CGRAM, DDRAM, IRAM) 地址都可以连续写入 2 个字节的显示数据，当写入第二个字节时，地址计数器 (AC) 的值自动加一。

11) 读取显示 RAM 数据 (RS=1, R/W=1)

格式

D7	D6	D5	D4	D3	D2	D1	D0
----	----	----	----	----	----	----	----

读取后会使得 AC 改变

设定 RAM (CGRAM, DDRAM, IRAM) 地址后，先要 Dummy read 一次后才能读取到正确的显示数据，第二次读取不需要 Dummy read，除非重新设置了 RAM 地址

3. 扩充指令详细说明表

1) 待命模式

格式

0	0	0	0	0	0	0	1
---	---	---	---	---	---	---	---

进入待命模式，执行如何其它指令都可以结束待命模式；该指令不能改变 RAM 的内容。

2) 卷动位置或者 RAM 地址选择 初始值: 02H

格式

0	0	0	0	0	0	1	SR
---	---	---	---	---	---	---	----

当 SR=1 时，允许输入垂直卷动地址

当 SR=0 时，允许输入 IRAM 地址 (扩充指令) 及允许设定 CGRAM 地址 (基本指令)

3) 反白显示 初始值: 04H

格式

0	0	0	0	0	1	0	R0
---	---	---	---	---	---	---	----

选择 2 行中的任意一行作反白显示，并可决定反白与否。R0 初始值为 0，第一次执行时为反白显示，再次执行时为正常显示

通过 R0 选择要作反白处理的行：

R0=0 第一行， R0=1 第二行

说明：参考基本指令详细说明中的 DDRAM 地址说明

128 × 64 点阵的液晶执行反白功能时实用意义不大，因为一三行连在一起，二四行连在一起，用户对第一行执行反白显示操作时，第三行必然也反白显示。

4) 睡眠模式 初始值: 0000 10XXB(X=0,1)

格式

0	0	0	0	1	SL	0	0
---	---	---	---	---	----	---	---

SL=1，脱离睡眠模式

SL=0，进入睡眠模式

5) 扩充功能设定

初始值: 001 DL X100 B (DL=1,8BIT 并口;DL=0,4BIT 并口 X=0,1)

格式

0	0	1	DL	X	RE	G	X
---	---	---	----	---	----	---	---

DL: 8/4 位接口控制位

DL=1, 8 位 MPU 接口; DL=0, 4 位 MPU 接口

RE: 指令集选择控制位

RE=1, 扩充指令集; RE=0, 基本指令集

G: 绘图显示控制位

G=1, 绘图显示开; G=0, 绘图显示关

同一指令的动作不能同时改变 RE 及 DL、G, 需先改变 DL 或 G 再改变 RE 才能确保设置正确

6) 设定 IRAM 地址或卷动地址

格式

0	1	A5	A4	A3	A2	A1	A0
---	---	----	----	----	----	----	----

SR=1, A5~A0 为垂直卷动地址; SR=0, A3~A0 为 IRAM 地址

7) 设定绘图 RAM 地址

格式

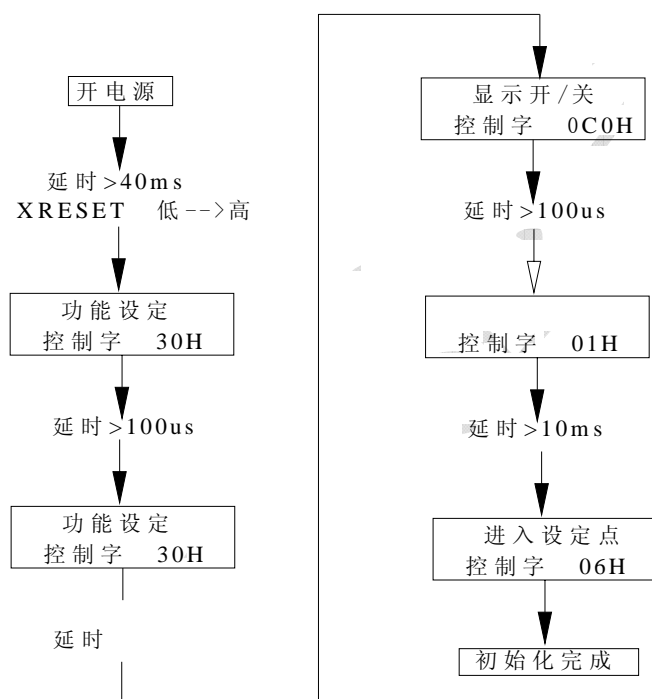
1	0	0	0	A3	A2	A1	A0
	A6	A5	A4	A3	A2	A1	A0

设定 GDRAM 地址到地址计数器 (AC), 先设置垂直位置再设置水平位置 (连续写入 2 字节数据来完成垂直与水平坐标的设置)。

垂直地址范围: AC6~AC0

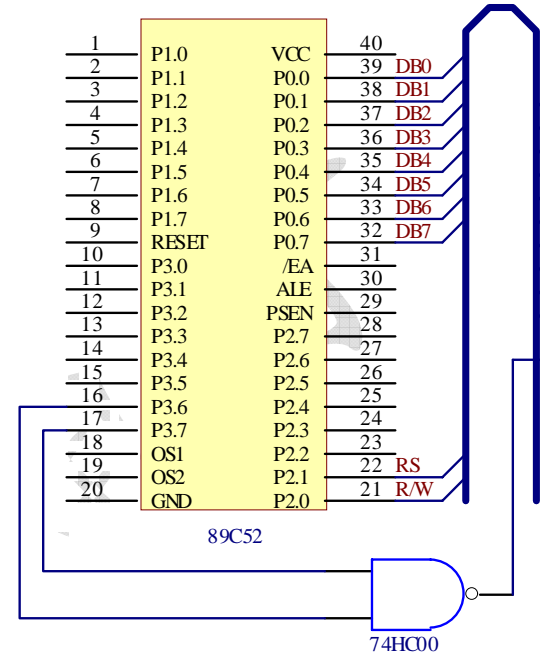
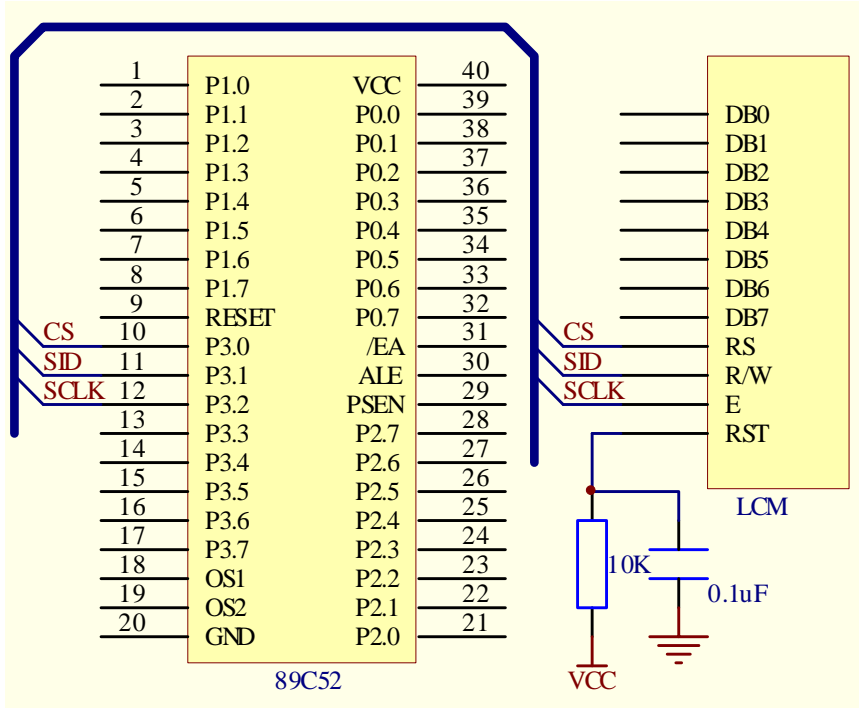
水平地址范围: AC3~AC0

4. 初始化流程

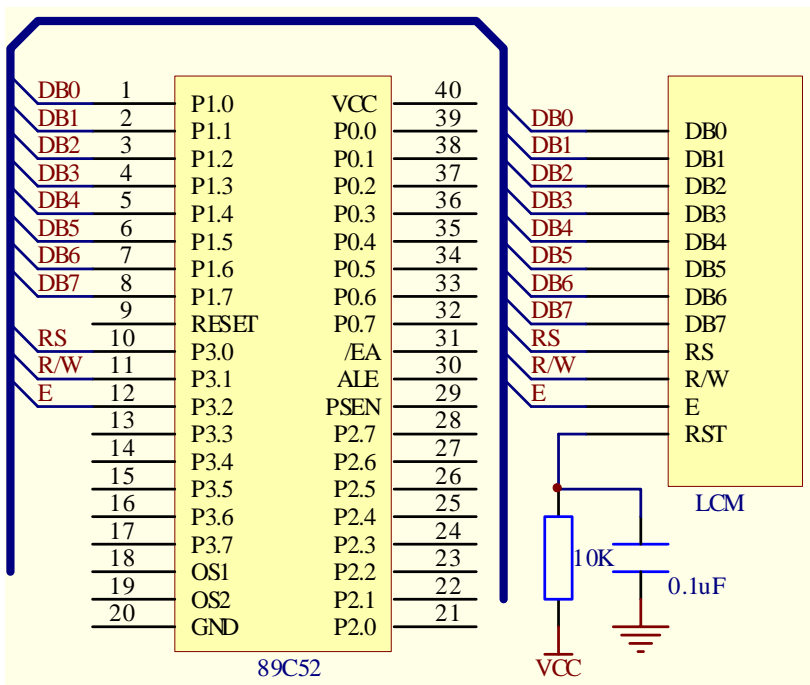


五. MCU 接口图

1. 并口直接访问方式



2. 并口间接访问方式



3. 串口接线图

示例程序

1. 并口直接访问方式

```

;*****
;CONTROLLER:ST7920(8BIT INTERFACE) *
;EDITOR:ZY *
;MCU=89C52
;CRYSTAL:12MHz *
;RS=P2.0 RW=P2.1 E=/(WR*RD) *
;DB0-DB7=P0 *
;*****

DATWP EQU 0100H ;写数据通道
DATRP EQU 0300H ;读数据通道
COMWP EQU 0000H ;写数据通道
COMRP EQU 0200H ;读忙通道

COM EQU 30H ;控制字暂存单元
DAT EQU 31H ;显示数据暂存单元
CODER EQU 32H ;字符代码暂存单元
ADDR EQU 33H ;地址暂存单元

ORG 0000H
AJMP MAIN
ORG 0030H

MAIN:
MOV SP,#60H
LCALL DEL_20MS
LCALL DEL_20MS
LCALL INI ;进入初始化子程序

MOV ADDR,#80H
MOV DPTR,#WEL_1
LCALL W_LINE
MOV ADDR,#90H
MOV DPTR,#WEL_2
LCALL W_LINE
MOV ADDR,#88H
MOV DPTR,#WEL_3
LCALL W_LINE
MOV ADDR,#98H
MOV DPTR,#WEL_4
LCALL W_LINE
LCALL DEL_1500MS

```



```

MOV  ADDR,#80H
MOV  DPTR,#WEL_5
LCALL W_LINE
MOV  ADDR,#90H
MOV  DPTR,#WEL_6
LCALL W_LINE
MOV  ADDR,#88H
MOV  DPTR,#WEL_7
LCALL W_LINE
MOV  ADDR,#98H
MOV  DPTR,#WEL_8
LCALL W_LINE
LCALL DEL_1500MS

MOV  ADDR,#80H      ;
MOV  DPTR,#HZ_1    ;
LCALL W_LINE
MOV  ADDR,#90H      ;
MOV  DPTR,#HZ_2    ;
LCALL W_LINE
MOV  ADDR,#88H      ;
MOV  DPTR,#HZ_3    ;
LCALL W_LINE
MOV  ADDR,#98H      ;
MOV  DPTR,#HZ_4    ;
LCALL W_LINE
LCALL DEL_1500MS   ;延时后显示下一页

MOV  ADDR,#80H      ;
MOV  DPTR,#HZ_5    ;
LCALL W_LINE
MOV  ADDR,#90H      ;
MOV  DPTR,#HZ_6    ;
LCALL W_LINE
MOV  ADDR,#88H      ;
MOV  DPTR,#HZ_7    ;
LCALL W_LINE
MOV  ADDR,#98H      ;
MOV  DPTR,#HZ_8    ;
LCALL W_LINE
LCALL DEL_1500MS

LJMP MAIN

```

INI:

```
MOV  COM,#30H      ;功能设定,基本指令
```

```

LCALL WCOM
MOV  COM,#30H      ;基本指令,8-bit 模式,基本指令
LCALL WCOM
MOV  COM,#0CH      ;显示开,游标关,反白关
LCALL WCOM
MOV  COM,#01H      ;清除显示
LCALL WCOM
MOV  COM,#06H      ;进入设定点,游标 7 右移,画面不移动
LCALL WCOM
RET

```

W_LINE:

```

MOV  COM,ADDR      ;
LCALL WCOM          ;
MOV  R2,#16        ;16 个循环,连续写入 8 个汉字或是 16 个西文字符

```

W_L1:

```

MOV  A,#00H
MOVC A,@A+DPTR
MOV  CODER,A
LCALL WCODE
INC  DPTR
DJNZ R2,W_L1
RET

```

WCOM:

```

PUSH DPH
PUSH DPL
MOV  DPTR,#COMRP

```

WCOM1:

```

MOVX A,@DPTR
JB  ACC.7,WCOM1
MOV  DPTR,#COMWP
MOV  A,COM
MOVX @DPTR,A
POP  DPL
POP  DPH
RET

```

WCODE:

```

PUSH DPH
PUSH DPL
MOV  DPTR,#COMRP

```

WCODE1:

```

MOVX A,@DPTR
JB  ACC.7,WCODE1
MOV  DPTR,#DATWP
MOV  A,CODER

```

```
MOVX @DPTR,A
POP DPL
POP DPH
RET
```

DEL_20MS:

```
MOV R0,#20
```

D2:

```
MOV R1,#200
```

D1:

```
NOP
```

```
NOP
```

```
NOP
```

```
DJNZ R1,D1
```

```
DJNZ R0,D2
```

```
RET
```

DEL_100MS:

```
LCALL DEL_20MS
```

```
LCALL DEL_20MS
```

```
LCALL DEL_20MS
```

```
LCALL DEL_20MS
```

```
LCALL DEL_20MS
```

```
RET
```

DEL_500MS:

```
LCALL DEL_100MS
```

```
LCALL DEL_100MS
```

```
LCALL DEL_100MS
```

```
LCALL DEL_100MS
```

```
LCALL DEL_100MS
```

```
RET
```

DEL_1500MS:

```
LCALL DEL_500MS
```

```
LCALL DEL_500MS
```

```
LCALL DEL_500MS
```

```
RET
```

WEL_1:

```
DB " "
```

WEL_2:

```
DB " _____ "
```

WEL_3:

WEL_4:

```
DB "128X64 全点阵显示"
```

```

WEL_5:
    DB "图形以及字符显示"
WEL_6:
    DB "可显示 8X4 个中文"
WEL_7:
    DB "内带标准中文字库"
WEL_8:
    DB "含 8192 个中文字型"

HZ_1:
    DB "临别殷勤重寄词，"
HZ_2:
    DB "词中有誓两心知。"
HZ_3:
    DB "七月七日长生殿，"
HZ_4:
    DB "夜半无人私语时。"
HZ_5:
    DB "在天愿作比翼鸟，"
HZ_6:
    DB "在地愿为连理枝。"
HZ_7:
    DB "天长地久有时尽，"
HZ_8:
    DB "此恨绵绵无绝期。"

END

```

2. 并口间接访问方式

```

;*****
;CONTROLLER:ST7920(8BIT INTERFACE)
;128×64 点阵
;EDITOR:ZY
;CRYSTAL:12MHz
;RS=P3.0 RW=P3.1 E=P3.2 D0-D7=P1
;*****

RS    EQU  P3.0    ;H=DATA,L=COM
RW    EQU  P3.1    ;H=READ,L=WRITE
E     EQU  P3.2    ;

COM   EQU  30H    ;控制字暂存单元
DAT   EQU  31H    ;显示数据暂存单元
CODER EQU  32H    ;字符代码暂存单元
ADDR  EQU  33H    ;地址暂存单元

```

```
ORG 0000H
AJMP MAIN
ORG 0030H
```

MAIN:

```
MOV SP,#60H
MOV P3,#0F8H
LCALL DEL_20MS
LCALL DEL_20MS
LCALL INI ;进入初始化子程序
```

```
MOV ADDR,#80H
MOV DPTR,#WEL_1
LCALL W_LINE
MOV ADDR,#90H
MOV DPTR,#WEL_2
LCALL W_LINE
MOV ADDR,#88H
MOV DPTR,#WEL_3
LCALL W_LINE
MOV ADDR,#98H
MOV DPTR,#WEL_4
LCALL W_LINE
LCALL DEL_1500MS
```

```
MOV ADDR,#80H
MOV DPTR,#WEL_5
LCALL W_LINE
MOV ADDR,#90H
MOV DPTR,#WEL_6
LCALL W_LINE
MOV ADDR,#88H
MOV DPTR,#WEL_7
LCALL W_LINE
MOV ADDR,#98H
MOV DPTR,#WEL_8
LCALL W_LINE
LCALL DEL_1500MS
```

```
MOV ADDR,#80H ;
MOV DPTR,#HZ_1 ;
LCALL W_LINE
MOV ADDR,#90H ;
MOV DPTR,#HZ_2 ;
LCALL W_LINE
MOV ADDR,#88H ;
```

```

MOV  DPTR,#HZ_3    ;
LCALL W_LINE
MOV  ADDR,#98H     ;
MOV  DPTR,#HZ_4    ;
LCALL W_LINE
LCALL DEL_1500MS   ;延时后显示下一页

MOV  ADDR,#80H     ;
MOV  DPTR,#HZ_5    ;
LCALL W_LINE
MOV  ADDR,#90H     ;
MOV  DPTR,#HZ_6    ;
LCALL W_LINE
MOV  ADDR,#88H     ;
MOV  DPTR,#HZ_7    ;
LCALL W_LINE
MOV  ADDR,#98H     ;
MOV  DPTR,#HZ_8    ;
LCALL W_LINE
LCALL DEL_1500MS

```

```
LJMP MAIN
```

INI:

```

MOV  COM,#30H      ;功能设定,基本指令
LCALL WCOM
MOV  COM,#30H      ;基本指令,8-bit 模式 基本指令
LCALL WCOM
MOV  COM,#0CH      ;      游标关,反白关
LCALL WCOM
MOV  COM,#01H      ;清除显示
LCALL WCOM
MOV  COM,#06H      ;进入设定点,游标 7 右移,画面不移动
LCALL WCOM
RET

```

W_LINE: ;写入一行 8 汉字子程序

```

MOV  COM,ADDR     ;
LCALL WCOM        ;
MOV  R2,#16       ;16 个循环,连续写入 8 个汉字或是 16 个西文字符

```

W_L1:

```

MOV  A,#00H
MOVC A,@A+DPTR
MOV  CODER,A
LCALL WCODE
INC  DPTR
DJNZ R2,W_L1

```

RET

WCOM: ;写指令子程序

CLR RS
SETB RW

WC1:

MOV P1,#0FFH
SETB E
MOV A,P1
CLR E
JB ACC.7,WC1 ;查忙
CLR RW
MOV P1,COM
SETB E
CLR E
RET

WCODE: ;写代码子程序

CLR RS
SETB RW

WCO1:

MOV P1,#0FFH
SETB E
MOV A,P1
CLR E
JB ACC.7,WCO1
SETB RS
CLR RW
MOV P1,CODER
SETB E
CLR E
RET

DEL_20MS:

MOV R0,#20

D2:

MOV R1,#200

D1:

NOP
NOP
NOP
DJNZ R1,D1
DJNZ R0,D2
RET

DEL_100MS:

LCALL DEL_20MS

```
LCALL DEL_20MS
LCALL DEL_20MS
LCALL DEL_20MS
LCALL DEL_20MS
RET
```

DEL_500MS:

```
LCALL DEL_100MS
LCALL DEL_100MS
LCALL DEL_100MS
LCALL DEL_100MS
LCALL DEL_100MS
RET
```

DEL_1500MS:

```
LCALL DEL_500MS
LCALL DEL_500MS
LCALL DEL_500MS
RET
```

WEL_1:

```
DB " "
```

WEL_2:

```
DB "_____"
```

WEL_3:

```
DB "123456789"
```

WEL_4:

```
DB "128X64 全点阵显示"
```

WEL_5:

```
DB "图形以及字符显示"
```

WEL_6:

```
DB "可显示 个中文"
```

WEL_7:

```
DB "内带标准中文字库"
```

WEL_8:

```
DB " 8192 个中文字型"
```

HZ_1:

```
DB "昨夜星辰昨夜风, "
```

HZ_2:

```
DB "画楼西畔桂堂东。"
```

HZ_3:

```
DB "身无彩凤双飞翼, "
```

HZ_4:

```
DB "心有灵犀一点通。"
```

HZ_5:

```
DB "隔座送钩春酒暖, "
```


HZ_6:

DB "分曹射覆蜡灯红。"

HZ_7:

DB "嗟余听鼓应官去，"

HZ_8:

DB "走马兰台类转蓬。"

END

3. 4 位并口模式

请参照 8 位并口间接访问方式以及 4 位并口时序图。不同之处只是将每个字节分两次送入，第一次送入高四位，第二次送入低四位。

4. 串口程序

PSB 接低时,串口模式被选择。在该模式下，只用两根线(SID 与 SCLK)来完成数据传输。当同时使用多颗 ST7920 时，CS 线被配合使用，CS 是高有效。

ST7920 的他不时钟 SCLK 有独立的操作时序，当多个连续的指令需要被送入时，指令执行时间需要被考虑。必须等待上一个指令执行完毕才送入下一个指令，因为 ST7920 内部没有传送/接收缓冲区。

一个完整的串行传输周期由一下部分组成：

首先送入启动字节，送入 5 个连续的“1”用来启动一个周期，此时传输计数被重置，并且串行传输被同步。紧接的两个位指定传输方向（RW，确定读还是写）和传输性质（RS，确定是命令寄存器还是数据寄存器），最后的第八位是一个“0”

送完启动字节之后，可以送入指令或是显示数据（或是字型代码）。指令或者代码是以字节为单位的，每个字节的内容（指令或数据）在被送入时分为两个字节来处理：高四位放在第一个字节的高四位，低四位放在第二个字节的高四位。无关位都补“0”。请参照第四章串行通讯时序图。

示例程序如下：

```

;*****
;CONTROLLER:ST7920(串行接口
;128×64 点阵
;EDITOR:ZY
;CRYSTAL:12MHz
;CS=P3.0 SID=P3.1 SCLK=P3.2
;*****
CS EQU P3.0 ;H=DATA,L=COM
SID EQU P3.1 ;H=READ,L=WRITE
SCLK EQU P3.2 ;

COM EQU 30H ;控制字暂存单元
DAT EQU 31H ;显示数据暂存单元
CODER EQU 32H ;字符代码暂存单元
ADDR EQU 33H ;地址暂存单元

ORG 0000H

```

```
AJMP MAIN
ORG 0030H
```

MAIN:

```
MOV SP,#60H
LCALL DEL_40MS
LCALL INI

MOV ADDR,#80H
MOV DPTR,#WEL_1
LCALL W_LINE
MOV ADDR,#90H
MOV DPTR,#WEL_2
LCALL W_LINE
MOV ADDR,#88H
MOV DPTR,#WEL_3
LCALL W_LINE
MOV ADDR,#98H
MOV DPTR,#WEL_4
LCALL W_LINE
LCALL DEL_1500MS
```

```
MOV ADDR,#80H
MOV DPTR,#WEL_5
LCALL W_LINE
MOV ADDR,#90H
MOV DPTR,#WEL_6
LCALL W_LINE
MOV ADDR,#88H
MOV DPTR,#WEL_7
LCALL W_LINE
MOV ADDR,#98H
MOV DPTR,#WEL_8
LCALL W_LINE
LCALL DEL_1500MS
```

```
MOV ADDR,#80H ;
MOV DPTR,#HZ_1 ;
LCALL W_LINE
MOV ADDR,#90H ;
MOV DPTR,#HZ_2 ;
LCALL W_LINE
MOV ADDR,#88H ;
MOV DPTR,#HZ_3 ;
LCALL W_LINE
MOV ADDR,#98H ;
MOV DPTR,#HZ_4 ;
```

```
LCALL W_LINE
LCALL DEL_1500MS ;延时后显示下一页
```

```
MOV ADDR,#80H ;
MOV DPTR,#HZ_5 ;
```

```
LCALL W_LINE
```

```
MOV ADDR,#90H ;
MOV DPTR,#HZ_6 ;
```

```
LCALL W_LINE
```

```
MOV ADDR,#88H ;
MOV DPTR,#HZ_7 ;
```

```
LCALL W_LINE
```

```
MOV ADDR,#98H ;
MOV DPTR,#HZ_8 ;
```

```
LCALL W_LINE
```

```
LCALL DEL_1500MS
```

```
LJMP MAIN
```

INI:

```
MOV COM,#30H ;功能设定,基本指令
LCALL WCOM
MOV COM,#30H ;基本指令,8-bit 模式,基本指令
LCALL WCOM
MOV COM,#0CH ;显示开 光标关 反白关
LCALL WCOM
MOV COM,#01H ;清除显示
LCALL WCOM
MOV COM,#06H ;进入设定点 光标 7 右移 画面不移动
LCALL WCOM
RET
```

W_LINE:

```
MOV COM,ADDR
LCALL WCOM
MOV R4,#16 ;连续写入 8 个中文或者 16 个西文字符
```

W_L1:

```
MOV A,#00H
MOVC A,@A+DPTR
MOV CODER,A
LCALL WCODE
INC DPTR
DJNZ R4,W_L1
RET
```

WCOM:

```
LCALL STWC
```

```

MOV  A,COM
LCALL W4_D      ;送入高四位指令
LCALL W4_0      ;连续送入四个 0
LCALL W4_D      ;送入高四位指令
LCALL W4_0      ;连续送入四个 0
CLR   CS
LCALL DEL_2MS
RET

```

WCODE:

```

LCALL STWD
MOV  A,CODER
LCALL W4_D
LCALL W4_0
LCALL W4_D
LCALL W4_0
CLR   CS
LCALL DEL_2MS
RET

```

STWC:

```

SETB  CS
SETB  SID
MOV   R3,#5      ;连续送入 5 个"1",起始

```

STWC1:

```

SETB  SCLK
CLR   SCLK
DJNZ  R3,STWC1
CLR   SID
MOV   R3,#3

```

STWC2:

```

SETB  SCLK      ;RW=0,RS=0,第八位"0"
CLR   SCLK
DJNZ  R3,STWC2
RET

```

STWD:

```

SETB  CS
SETB  SID
MOV   R3,#5      ;连续送入 5 个"1",起始

```

STWD1:

```

SETB  SCLK
CLR   SCLK
DJNZ  R3,STWD1
CLR   SID      ;RW=0
SETB  SCLK
CLR   SCLK

```

```
SETB SID ;RS=1
SETB SCLK
CLR SCLK
CLR SID ;第八位"0"
SETB SCLK
CLR SCLK
RET
```

W4_D:

```
MOV R3,#4
```

W4_D1:

```
RLC A
MOV SID,C
SETB SCLK
CLR SCLK
DJNZ R3,W4_D1
RET
```

W4_0:

```
MOV R3,#4
```

W4_01:

```
CLR SID
SETB SCLK
CLR SCLK
DJNZ R3,W4_01
RET
```

DEL_2MS:

```
MOV R0,#2
```

D1:

```
MOV R1,#200
```

D2:

```
NOP
NOP
NOP
DJNZ R1,D2
DJNZ R0,D1
RET
```

DEL_40MS:

```
MOV R2,#20
```

D3:

```
LCALL DEL_2MS
DJNZ R2,D3
RET
```

DEL_200MS:

```
MOV R2,#100
D4:
LCALL DEL_2MS
DJNZ R2,D4
RET

DEL_500MS:
MOV R2,#250
D5:
LCALL DEL_2MS
DJNZ R2,D5
RET

DEL_1500MS:
LCALL DEL_500MS
LCALL DEL_500MS
LCALL DEL_500MS
RET

WEL_1:

WEL_2:
DB " "
WEL_3:

WEL_4:
DB " "
WEL_5:
DB "示, 可显示 8X4"
WEL_6:
DB "共 32 个中文, 内带"
WEL_7:
DB "标准中文字库, 内"
WEL_8:
DB " "

HZ_1:
DB " 桃 夭 "
HZ_2:
DB "诗经·国风·周南"
HZ_3:
DB "桃之夭夭灼灼其华"
HZ_4:
DB "之子于归宜其室家"
HZ_5:
DB "桃之夭夭有蕢其实"
HZ_6:
```

DB "之子于归宜其家室"

HZ_7:

DB "桃之夭夭其叶蓁蓁"

HZ_8:

DB "之子于归宜其家人"

END

六. 客户选型指南

供电电压	<input type="checkbox"/> 5.0V <input type="checkbox"/> 3.3V
LCD 模式	<input type="checkbox"/> STN <input type="checkbox"/> FSTN <input type="checkbox"/> FFSTN
LCD 颜色	<input type="checkbox"/> 黄绿屏 <input type="checkbox"/> 蓝屏 <input type="checkbox"/> 黑白屏
透射模式	<input type="checkbox"/> 反射 <input type="checkbox"/> 半透 <input type="checkbox"/> 全透
背光源模式	<input type="checkbox"/> 黄绿色 <input type="checkbox"/> 白色 <input type="checkbox"/> 蓝色 <input type="checkbox"/> 翡翠绿 <input type="checkbox"/> 红色 <input type="checkbox"/> 琥珀色

----- **OVER** -----