



目 录

第一章、	显示器外形结构尺寸图	1
第二章、	显示器基本功能介绍	2
第三章、	显示器接口定义说明	3
第一节、	引出管脚定义说明	3
第二节、	背光引出脚说明	4
第四章、	显示器的电性参数	4
第一节、	直流供电参数	4
第二节、	级限参数	4
第三节、	液晶屏功耗	5
第五章、	显示器的显示结构原理	7
第一节、	显示器控制器方框图	7
第二节、	玻璃走线映射图	8
第三节、	显示内存映射图	9
第四节、	写入数据流程图	10
第六章、	驱动程序时序图说明	11
第一节、	模块串行接口时序图	11
第二节、	写串口时序的流程图	12
第三节、	写六八零零并口时序	13
第四节、	写八零八零并口时序	14
第五节、	写复位时序	15
第七章、	驱动程序的指令说明	15
第一节、	显示模块指令表：	16
第二节、	显示模块指令详细说明：	17
一、	显示开关设置	17
二、	起始行设置	17
三、	页地址设置	17
四、	列地址设置	17
五、	读状态设置	18
六、	写数据设置	19
七、	读数据设置	19
八、	列方向设置	19
九、	显示方向设置	19
十、	打开所以点阵	19
十一、	偏压选择设置	19
十二、	读到写设置	19
十三、	结束设置	19
十四、	复位设置	19
十五、	水平方向设置	19
十六、	电源控制设置	19
十七、	调节对比度比	20

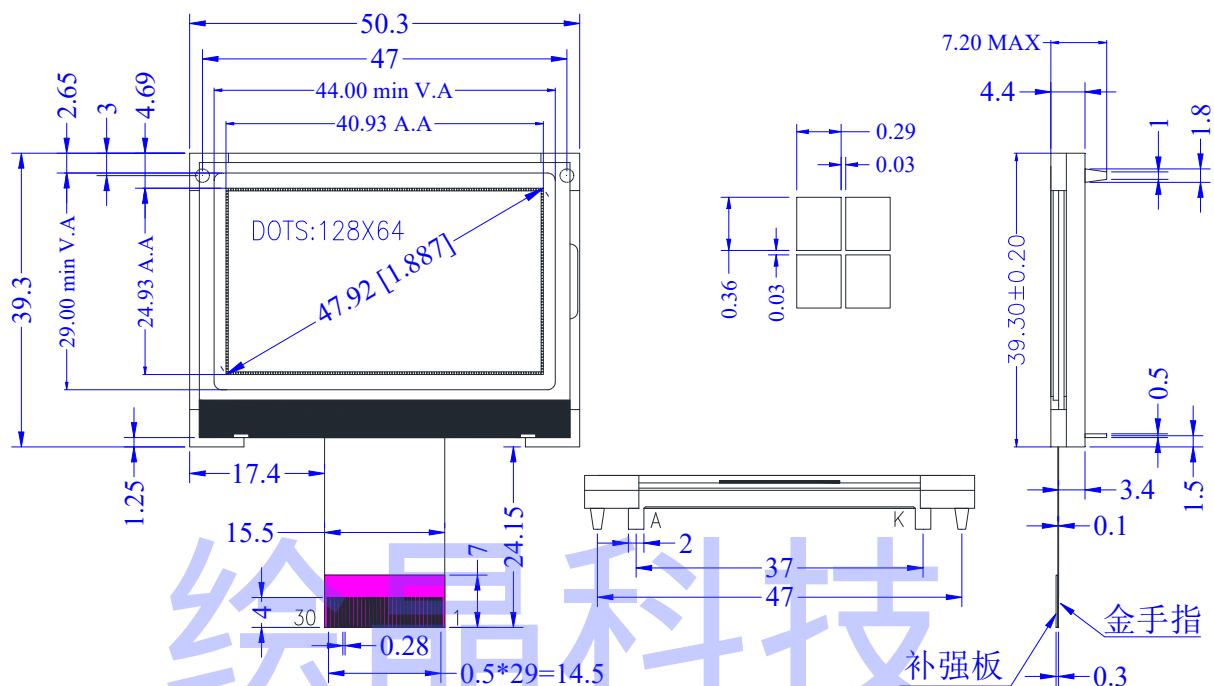


十八.	玻璃电压微调对比度设置	20
十九.	睡眠模式设置	21
二十.	省电设置	21
二十一.	升压倍数设置	21
二十二.	空操作设置	21
二十三.	测试设置	21
第八章、	单片机与显示器连接说明	22
第一节、	并口应用原理图	22
第二节、	串口应用原理图	23
第三节、	单片机串口连接图	24
第四节、	单片机并口连接图	24
第九章、	单片机驱动程序源代码	25
第一节、	源代码解释定义声明	25
第二节、	接口时序函数	26
第三节、	液晶模块初始化	28
第四节、	应用函数	28
第五节、	主调用函数	33
第六节、	动态显示函数	34
第七节、	标准字符数据表	36
第十章、	版本信息	36

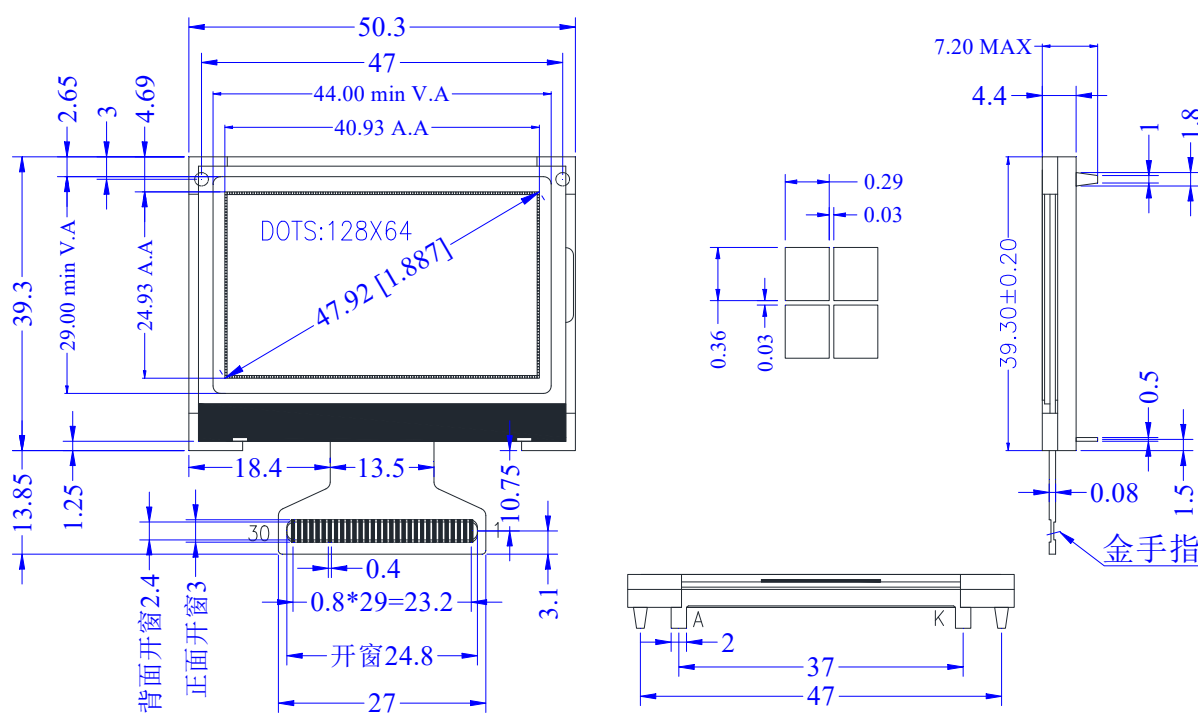
[返回目录 Ctrl+Home 或者 wep 里的返回箭头](#)

第一章、显示器外形结构尺寸图

(插拔式)



(焊接式)



背光由 3 支 LED 并联，串限流电阻

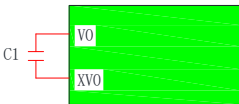
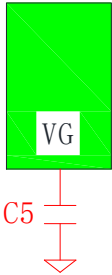
项目	参考值
单位 (英寸)	1.9 (1.887)
LCM 尺寸 (长×宽×厚)	50.3×39.3×4.4
可视区域 (长×宽)	44.0×29.0
点尺寸 (长×宽)	0.29×0.36
点间距 (长×宽)	0.32×0.39

第二章、显示器基本功能介绍

- ✧ COG: 将驱动芯片 (例如 ST7565R)、FPC 软板 (例如金手指) 直接压贴在玻璃上的封装形式叫做 COG, 有性价比高, 低功耗, 体积小, 软件指令丰富的优势
- ✧ 工作电压 3.3V, 背光电压 3.0V 45mA
- ✧ 通讯方式: 8 位并行 (6800 和 8080 时序), 4 线串行通讯
- ✧ 128*64 点阵显示器内存
 - 可自由编写图形或者汉字
- ✧ 功能指令
 - 行列扫描顺序设置 起始行 页 列地址设置
 - 显示开/关 反白显示
 - 电源控制 待机模式
 - 偏压比设置 LCD 倍压设置
 - 显示对比度设置 软件复位
- ✧ 低功耗省电设计 (除背光 45mA)
 - 正常模式 (147uA typ VDD=3.3V)
 - 待机模式 (4uA max VDD=3.3V)
- ✧ 显示对比度, 内部寄存器调节 (软件调对比度)
- ✧ 占空比 1/64 偏压比 1/9
- ✧ 工作温度 -20 到+70 度
- ✧ 视角 6 点钟, 可以根据需要 180, 270, 90 使用, 但是视角相对就变, 软件相对也要不一样, 根据 DDRAM 的映射结构编写函数,
- ✧ 显示颜色可以黄绿, 蓝白, 灰白, 也可以订制。采购时请和业务说明颜色

第三章、显示器接口定义说明

第一节、引出管脚定义说明

引脚	名称	方向	说明
1	NC	空脚	空脚
2	CSB	I	使能信号，低电平有效。
3	RSTB	I	低电平复位，复位完成后，回到高电平，液晶模块开始工作
4	A0 (RS)	I	数据/指令寄存器选择（高为数据，低为指令）
5	RWR	I	6800 时序时：R/W=0 写操作， R/W=1 读操作 8080 时序时：写数据端口，低电平有效
6	ERD	I	6800 时序时：使能端，高电平有效 8080 时序时：读数据端口，低电平有效
7-12	DB0 ~ DB5	I/O	MPU 与模块之间并口的数据传送通道，
13	DB6/SCLK	I	并口数据/串口时显示模块串行时钟
14	DB7/SDA	I/O	并口数据/串口时显示模块串行数据
15	VDD	--	电源正端(+3.3V)
16	VSS	--	电源负端(0V)
17	NC	倍压电路	升压电路  如图接 0.1UF-10uf 电容（常用 1Uf）
18	NC		
19	V0		
20	XV0		
21	NC		
22	NC		
23	VG	偏置电压	 如图接 0.1uf-10uf 电容（常用 1Uf）
24	NC		
25	NC		
26	NC		
27	NC		
28	C86	6800/8080	选择时序 6800(高电平) /8080（低电平）
29	P/S	并口/串口	通讯方式并口（高电平）/串口（低电平）

30	NC	空脚	空脚
----	----	----	----

第二节、背光引出脚说明

A	A	--	背光电源正极 (+3.0V) 45MA
K	K	--	背光电源负极 (0V)

第四章、显示器的电性参数

第一节、直流供电参数

名称	符号	测试条件	参数范围			单位
			最小	标准	最大	
模块工作电压	VDD	—	3.2	3.3	3.4	V
背光工作电压	VLED	—	2.8	3.0	3.05	V
I/O 输入高电平	VIH	—	2.2	—	VDD	V
I/O 输入低电平	VIL	—	-0.3	—	0.6	V
LCM 输出高电平	VOH	—	2.4	—	—	V
LCM 输出低电平	VOL	—	—	—	0.4	V
模块工作电流	IDD	=VDD	—	—	1.0	MA
模块待机电流	ID0	=VDD	—	—	10	uA
背光工作电流	ILED	=VLED	30	60	80	MA

第二节、级限参数

Parameter	Sysbol	Conditions	Uint
Digital power supply voltage	Vdd	-0.3~3.6	V
Analog power supply voltage	Vdd2	-0.3~3.6	V
LCD power supply voltage	Vout,v0	-0.3~13.5	V

LCD power supply voltage	V1,v2,v3,v4	-0.3~v0	V
Operating temperature	Topr	-25 to +80	0c
Storage temperature	Tstr	-55 to +125	0c

Notes

1. Stresses above those listed under Limiting Values may cause permanent damage to the device.
2. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to VSS unless otherwise noted.
3. Insure the voltage levels of VOUT, V0, V1, V2, V3, V4 and VSS always match the correct relation:
 $VOUT \geq V0 \geq V1 \geq V2 \geq V3 \geq V4 \geq VSS$

第三节、液晶屏功耗

Current consumption: During Display, without internal power system, current consumed by whole IC (bare die).

Test Pattern	Symbol	Condition	Rating			Unit	Note
			Min.	Typ.	Max.		
Display Pattern: SNOW (Static)	ISS	VDD=VDD2 =3.0V, V0=11.0V, Ta=25°C	—	19	32	μA	
Display OFF	ISS	VDD=VDD2 =3.0V, V0=11.0V, Ta=25°C	—	16	27	uA	

Current consumption: During Display, with internal power system, current consumed by whole IC (bare die).

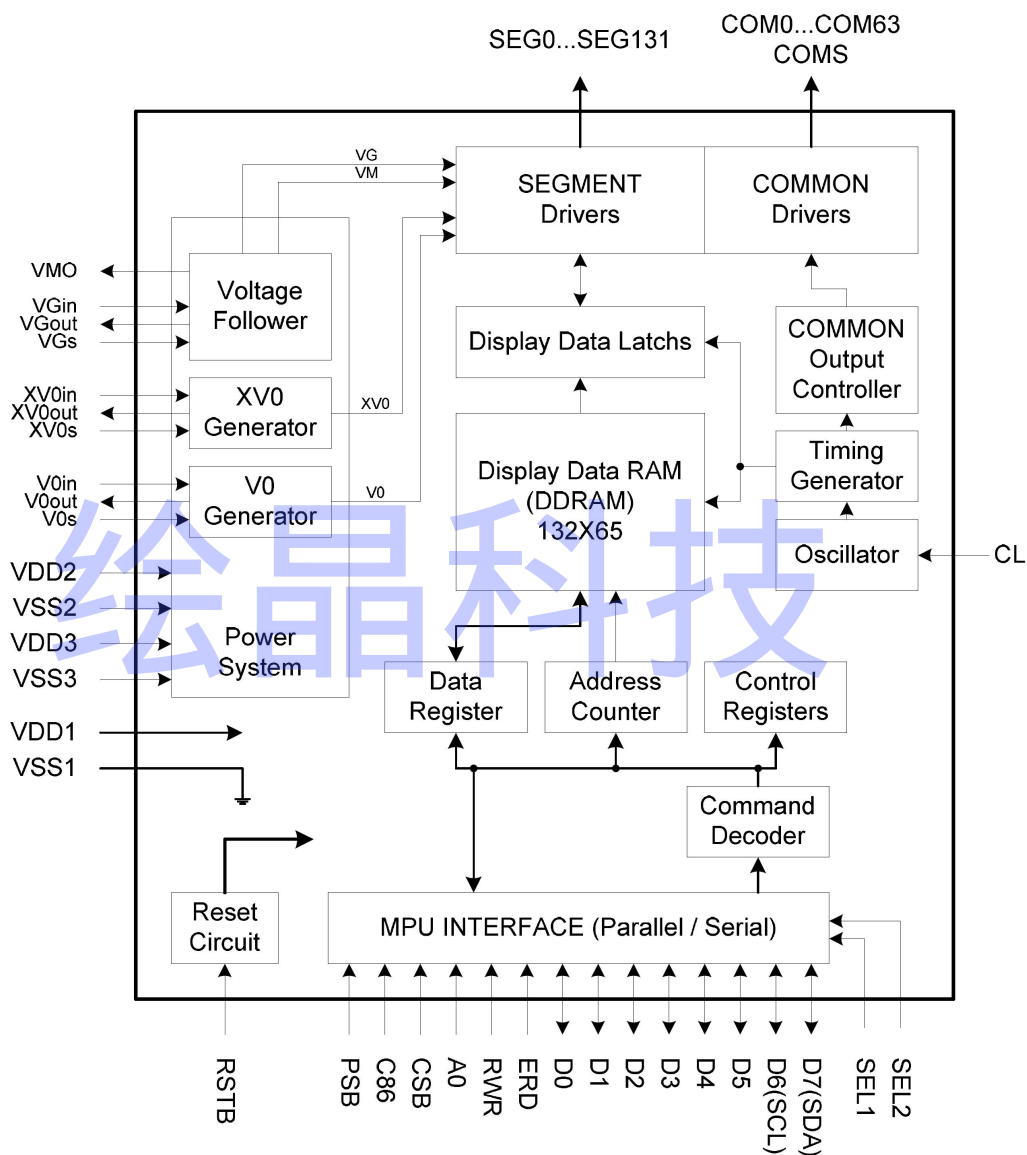
Test Pattern	Symbol	Condition	Rating			Unit	Note
			Min.	Typ.	Max.		
Display Pattern: SNOW	ISS	VDD=VDD2 =3.0V, V0=11.0V, Booster=x4, Ta=25°C	—	100	147	uA	Normal Mode
			—	135	205	uA	High Power Mode
Display OFF	ISS	VDD=VDD2 =3.0V, V0=11.0V, Booster=x4, Ta=25°C	—	90	130	uA	Normal Mode
			—	128	193	uA	High Power Mode
Sleep Mode	ISS	VDD=VDD2 =3.0V, Ta=25°C	—	0.4	4	uA	



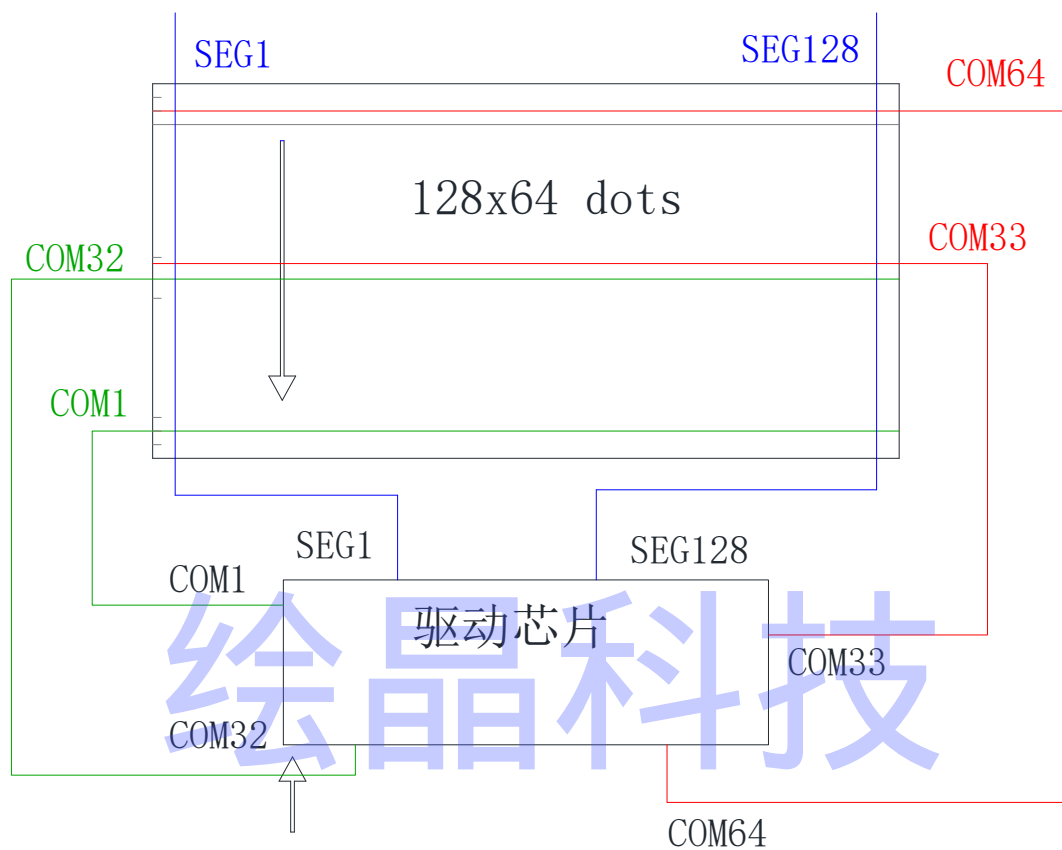
绘晶科技

第五章、显示器的显示结构原理

第一节、显示器控制器方框图



第二节、玻璃走线映射图

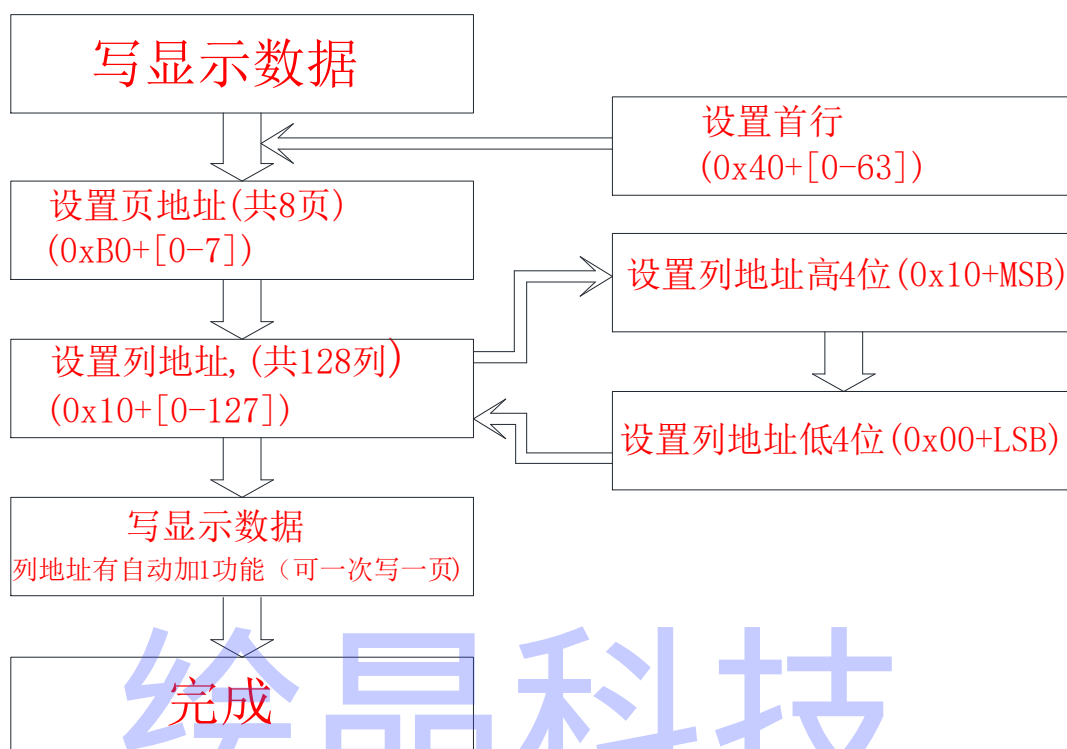


第三节、显示内存映射图

Y =	0-127列										行号
	0	1	...	62	63	64	65	...	126	127	
X=0	DB0	DB0	DB0	DB0	DB0	DB0	DB0	DB0	DB0	DB0	0
	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
	DB7	DB7	DB7	DB7	DB7	DB7	DB7	DB7	DB7	DB7	7
↓	DB0	DB0	DB0	DB0	DB0	DB0	DB0	DB0	DB0	DB0	8
	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
	DB7	DB7	DB7	DB7	DB7	DB7	DB7	DB7	DB7	DB7	55
X=7	DB0	DB0	DB0	DB0	DB0	DB0	DB0	DB0	DB0	DB0	56
	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
	DB7	DB7	DB7	DB7	DB7	DB7	DB7	DB7	DB7	DB7	63

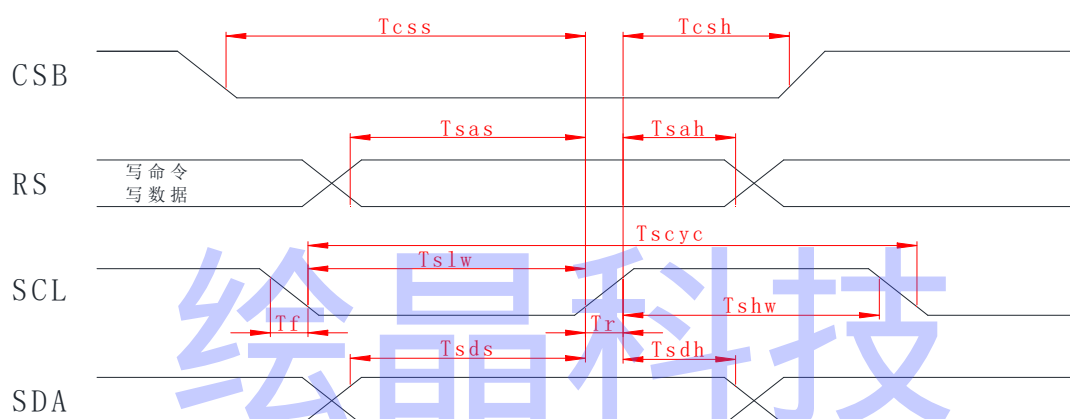
这是一款 128*64 的纯图形点阵屏，地址结构如图，列地址范围以 1 个像素单位定义为（0-127）列，行地址范围以 8 个像素为一个单位定义为（0-7）页，首行地址范围为一个像素单位，设置范围为（0-63）

第四节、写入数据流程图



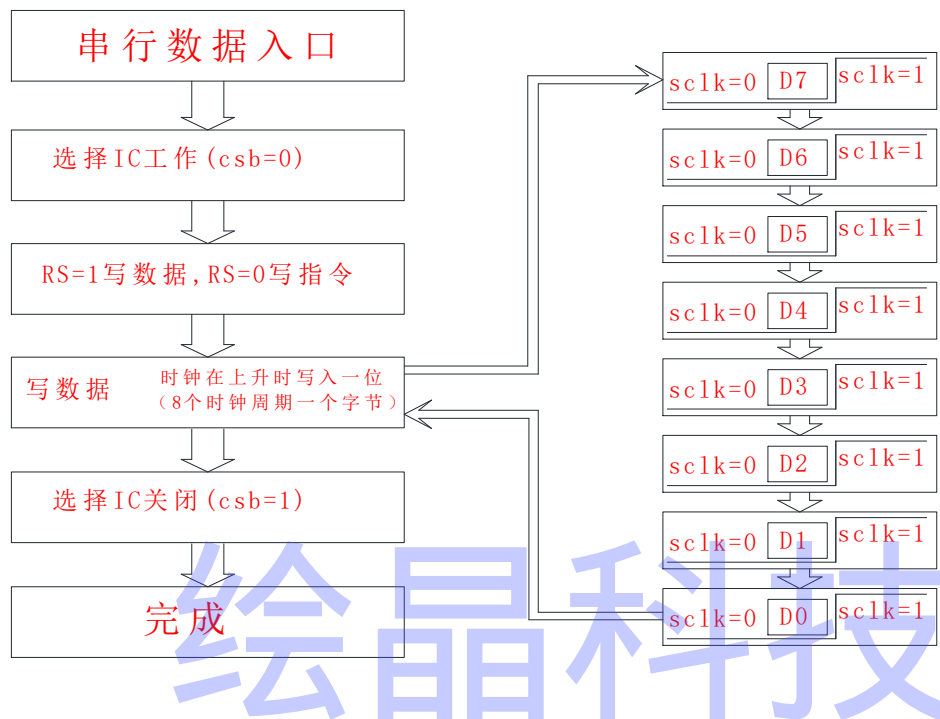
第六章、 驱动程序时序图说明

第一节、模块串行接口时序图



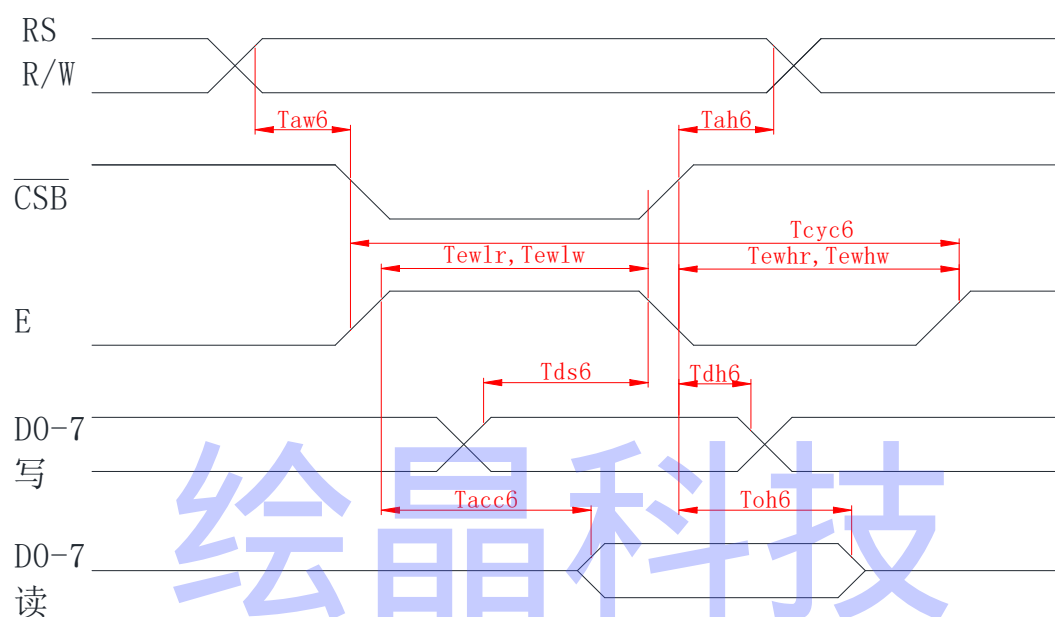
项目	信号	标识	条件	最小	最大	单位
串行时钟	SCL	Tscyc		50	-	纳秒
时钟高电平宽度		Tshw		25	-	
时钟高电平宽度		Tslw		25	-	
指令建立时间	RS	Tsas		20	-	
指令宽度		Tsah		10	-	
数据建立时间	SDA	Tsds		20	-	
数据宽度		Tsdh		10	-	
片选建立时间	CSB	Tcss		20	-	
片选保持时间		Tcsh		40	-	

第二节、写串口时序的流程图



第三节、写六八零零并口时序

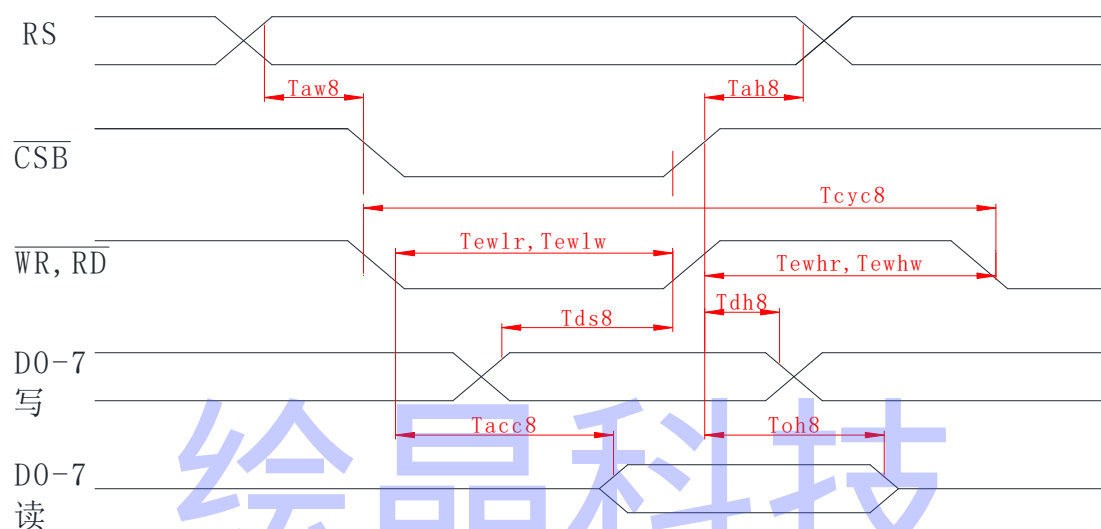
6800并口时序



项目	信号	标识	条件	最小	最大	单位
指令建立时间	RS	Taw6		0	-	纳秒
指令宽度		Tah6		0	-	
系统循环时间	E	Tcyc6		400	-	
写使能低电平宽度		Tewlw		220	-	
写使能高电平宽度		Tewhw		180	-	
读使能低电平宽度		Tewlr		220	-	
读使能高电平宽度		Tewhr		180	-	
写数据建立时间	D[7:0]	Tds6		40	-	
写数据数据宽度		Tdh6		0	-	
读数据时间		Tacc6	CL=100pF	-	140	
读数据禁用时间		Toh6	CL=100pF	10	100	

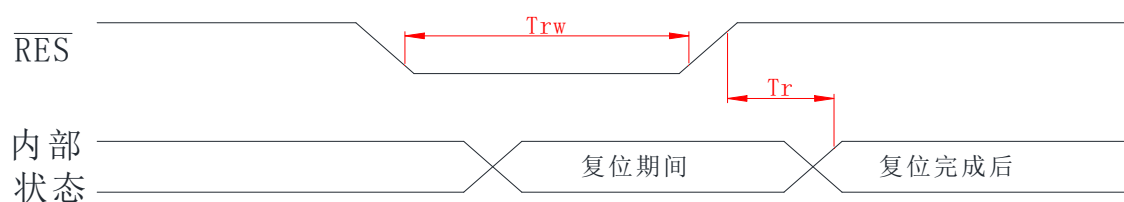
第四节、写八零八零并口时序

8080并口时序



项目	信号	标识	条件	最小	最大	单位
指令建立时间	RS	Taw8		0	-	纳秒
指令宽度		Tah8		0	-	
系统循环时间	/WR	Tcyc8		400	-	
写使能低电平宽度		Tewlw		220	-	
写使能高电平宽度		Tewhw		180	-	
读使能低电平宽度	RD	Tewlr		220	-	
读使能高电平宽度		Tewhr		180	-	
写数据建立时间	D[7:0]	Tds8		40	-	
写数据数据宽度		Tdh8		0	-	
读数据时间		Tacc8	CL=100pF	-	140	
读数据禁用时间		Toh8	CL=100pF	10	100	

第五节、写复位时序



项目	符号	测试条件	范围			单位
			最小	最佳	最大	
复位保持低电平时间	T_{rw}	引脚: RES	3.0	-	-	us
复位到内部状态延时	T_r	引脚和 IC 内部状态	6.0	-	-	ms

第七章、 驱动程序的指令说明

第一节、显示模块指令表：

N O	指令	指令码										HEX	说明
		A0	RWR	D7	D6	D5	D4	D3	D2	D1	D0		
1	显示开/关	0	0	1	0	1	0	1	1	1	D	AF	D=1, 显示开 D=0, 显示关
2	起始行设置 (COMO)	0	0	0	1	S5	S4	S3	S2	S1	S0	40	设置显示屏第一行的位置
3	页地址设置	0	0	1	0	1	1	Y3	Y2	Y1	Y0	B0	设置页地址
4	列地址设置	0	0	0	0	0	1	X7	X6	X5	X4	10	列地址高4位 (MSB)
		0	0	0	0	0	0	X3	X2	X1	X0	00	列地址低4位 (LSB)
5	读状态	0	1	BUS Y	MX	D	RST	0	0	0	0		读出模块的IC内部工作状态
6	写数据	1	0	D7	D6	D5	D4	D3	D2	D1	D0	00	向模块内写显示数据
7	读数据	1	1	D7	D6	D5	D4	D3	D2	D1	D0	00	读出模块IC的数据
8	列方向设置 (SEG)	0	0	1	0	1	0	0	0	0	MX	A0	设置列的扫描方向 MX=1, 反方向 (右到左) MX=0, 正方向 (左到右)
9	显示方向	0	0	1	0	1	0	0	1	1	INV	A6	INV=1, 反向显示 INV=0, 正常显示
10	全部点阵打开	0	0	1	0	1	0	0	1	0	AP	A4	AP=1, 点阵全部打开 AP=0, 正常
11	偏压选择	0	0	1	0	1	0	0	0	1	BS	A2	偏压比设置 0=1/9; 1=1/7 (1/64duty)
12	Read-modify-Write	0	0	1	1	1	0	0	0	0	0	E0	Columnaddress increment: Read:+0, Write:+1
13	END	0	0	1	1	1	0	1	1	1	0	EE	Exit Read-modify-Write mode
14	复位	0	0	1	1	1	0	0	0	1	0	E2	软件复位
15	行方向 (COM)	0	0	1	1	0	0	MY	-	-	-	C0	设置行方向 MY=1, 反方向 (下到上) MY=0, 正方向 (上到下)
16	LCD电源控制	0	0	0	0	1	0	1	VB	VR	VF	28	供应LCD电压的电路控制 =1, 打开; =0, 关闭
17	Regulation Ratio	0	0	0	0	1	0	0	RR2	RR1	RR0	20	选择调节电阻比
18	LCD电压设置 (对比度)	0	0	1	0	0	0	0	0	0	1	81	双指令模式, 6位精度调节LCD 电压 (显示双比度)
		0	0	0	0	EV5	EV4	EV3	EV2	EV1	EV0	00	
19	睡眠模式设置	0	0	1	0	1	0	1	1	0	MD	AC	MD=0, 睡眠模式 MD=1, 正常模式
		0	0	0	0	0	0	0	0	0	0	00	
20	省电	0	0	复合指令									显示关+全部点阵开
21	设置升压倍数 (LCD 电压)	0	0	1	1	1	1	1	0	0	0	F8	双指令模式 设置升压倍数 BL (0 0) 2倍, 3倍, 4倍 BL (0 1) 5倍 BL (1 0) 6倍
		0	0	0	0	0	0	0	0	BL1	BL0	00	
22	空指令	0	0	1	1	1	0	0	0	1	1	E3	空操作
23	测试	0	0	1	1	1	1	-	-	-	-	F0	不要使用, 保留测试

第二节、显示模块指令详细说明：

一. 显示开关设置

D 标志选择显示模式

	RS	RW	D7	D6	D5	D4	D3	D2	D1	D0	HEX
代码	0	0	1	0	1	0	1	1	1	D	AE

D=1；正常显示模式

D=0;显示关闭，所以SEG/COM输出低电平

二. 起始行设置

显示内存的第一行对应显示屏顶部的第一行地址设置

	RS	RW	D7	D6	D5	D4	D3	D2	D1	D0	HEX
代码	0	0	0	1	S5	S4	S3	S2	S1	S0	40

三. 页地址设置

128*8点阵为一页， $12864 = (1286 * 8) * 8$ 页，与起始行决定行在显示屏上的位置

	RS	RW	D7	D6	D5	D4	D3	D2	D1	D0	HEX
代码	0	0	1	0	1	1	Y3	Y2	Y1	Y0	B0

四. 列地址设置

设置列地址

	RS	RW	D7	D6	D5	D4	D3	D2	D1	D0	HEX
代码	0	0	0	0	0	1	X7	X6	X5	X4	10

	RS	RW	D7	D6	D5	D4	D3	D2	D1	D0	HEX
代码	0	0	0	0	0	0	X3	X2	X1	X0	00

第1列为 0x10(高4位) 0x00(低4位) 列地址需要写两次
第2列为 0x10(高4位) 0x01(低4位)
第128列为 0X17(高4位) 0x0F(低4位)

五. 读状态设置

读显示器内部状态。串行接口模式读取功能不可用。

	RS	RW	D7	D6	D5	D4	D3	D2	D1	D0	HEX
代码	0	0	BUSY	MX	D	RST	0	0	0	0	

标志	描述
BUSY	BUSY= 0: 命令可以接受 BUSY= 1: 命令或复位程序执行
MX	MX=0; 反向 (SEG131--SEG0) MX=1; 正常 (SEG0--SEG131)
D	D=0; 显示开 D=1; 显示关
RST	RST=1在复位 (硬件或软件复位) RES=0; 正常运行

绘晶科技

六. 写数据设置

七. 读数据设置

八. 列方向设置

九. 显示方向设置

十. 打开所以点阵

十一. 偏压选择设置

十二. 读到写设置

十三. 结束设置

十四. 复位设置

十五. 水平方向设置

十六. 电源控制设置

一般3个标志位同时打开

	RS	RW	D7	D6	D5	D4	D3	D2	D1	D0	HEX
代码	0	0	0	0	1	0	1	VB	VR	VF	28

标志	描述
VB	VB=0; 内置升压关闭 VB=1; 内置升压打开
VR	VR=0; 内置稳压关闭 VR=1; 内置稳压打开

VF	VF=0; 内置降压关闭 VF=1; 内置降压打开
----	------------------------------

十七. 调节对比度比

LCD 对比度档位选择

	RS	RW	D7	D6	D5	D4	D3	D2	D1	D0	HEX
代码	0	0	0	0	1	0	0	RR2	RR1	RR0	20
	标志		调节比								
	0x20		3.0								
	0x21		3.5								
	0x22		4.0								
	0x23		4.5								
	0x24		5.0								
	0x25		5.5								
	0x26		6.0								
	0x27		6.5								

工作电压（V0）的计算公式如下：（RR 为调节比，EV 为 LCD 电压设置）

$V0 = RR \times [1 - (63 - EV) / 162] \times 2.1$ ，或 $V0 = RR \times [(99 + EV) / 162] \times 2.1$

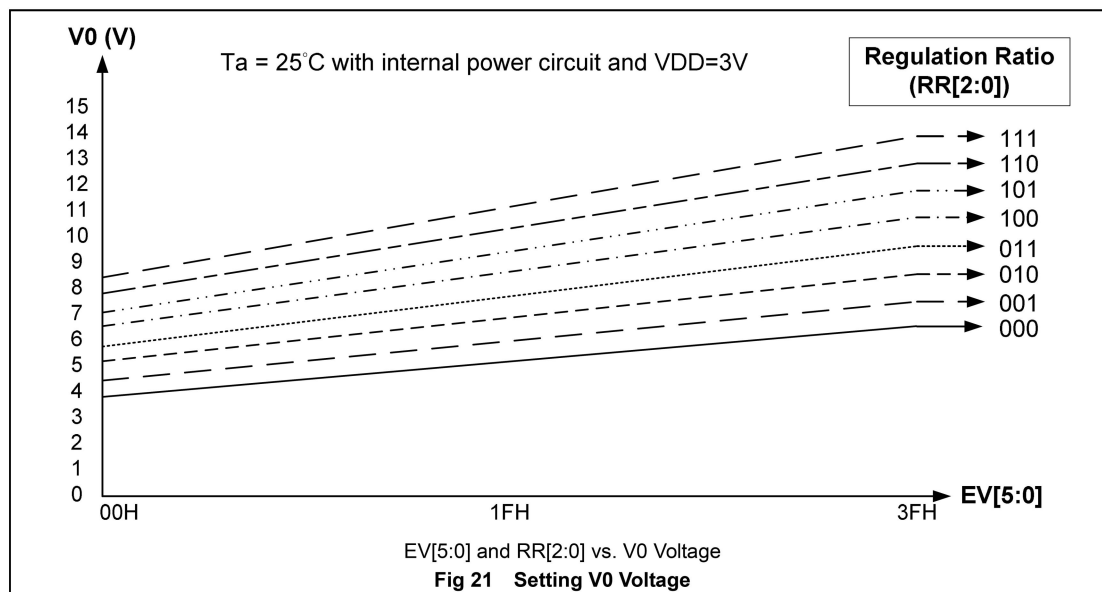
标志	寄存器	数值
RR	RR[2: 0]	3.0,3.5,4.0,4.5,5.0,5.5,6.0,6.5
EV	EV[5: 0]	0-63

十八. 玻璃电压微调对比度设置

双字节指令，两条指令要一起使用可以调节LCD的深浅显示

	RS	RW	D7	D6	D5	D4	D3	D2	D1	D0	HEX
代码	0	0	1	0	0	0	0	0	0	1	81
	0	0	0	0	EV5	EV4	EV3	EV2	EV1	EV0	0

LCD 电压与 EV 和 RR 之前的关系图



十九. 睡眠模式设置

二十. 省电设置

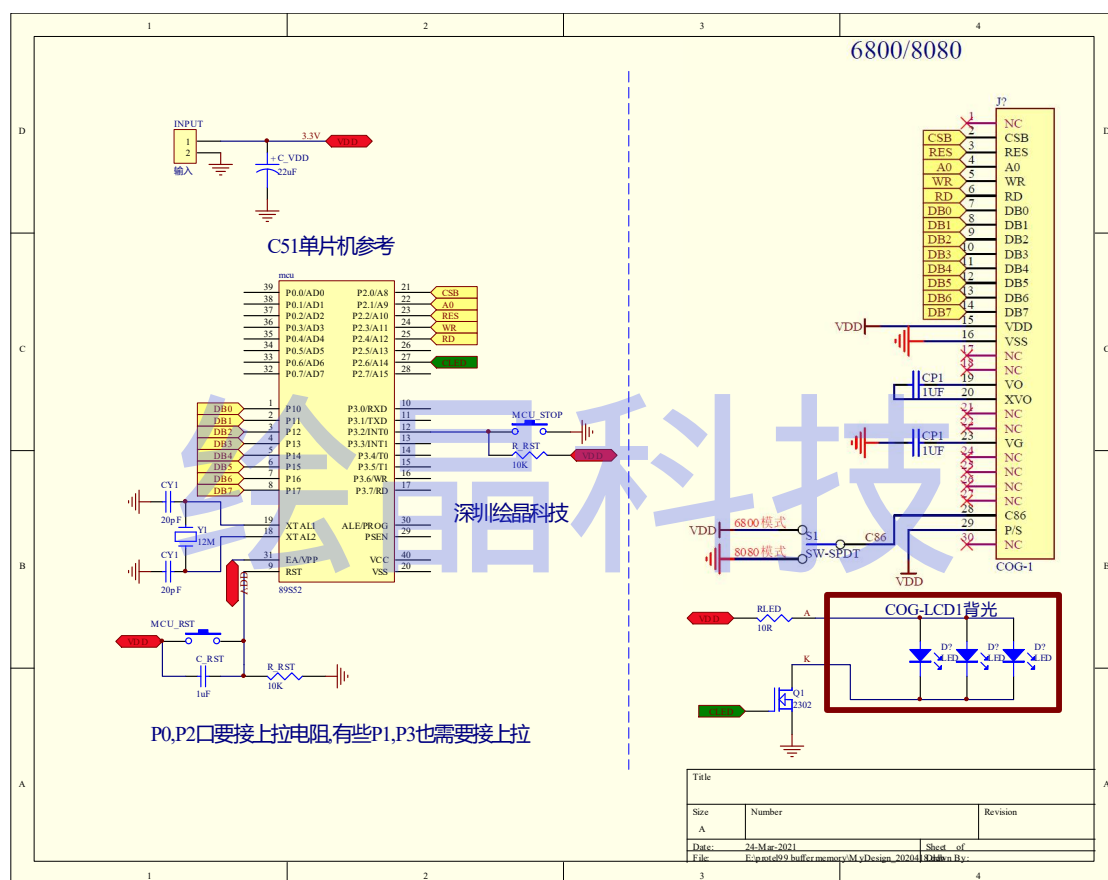
二十一. 升压倍数设置

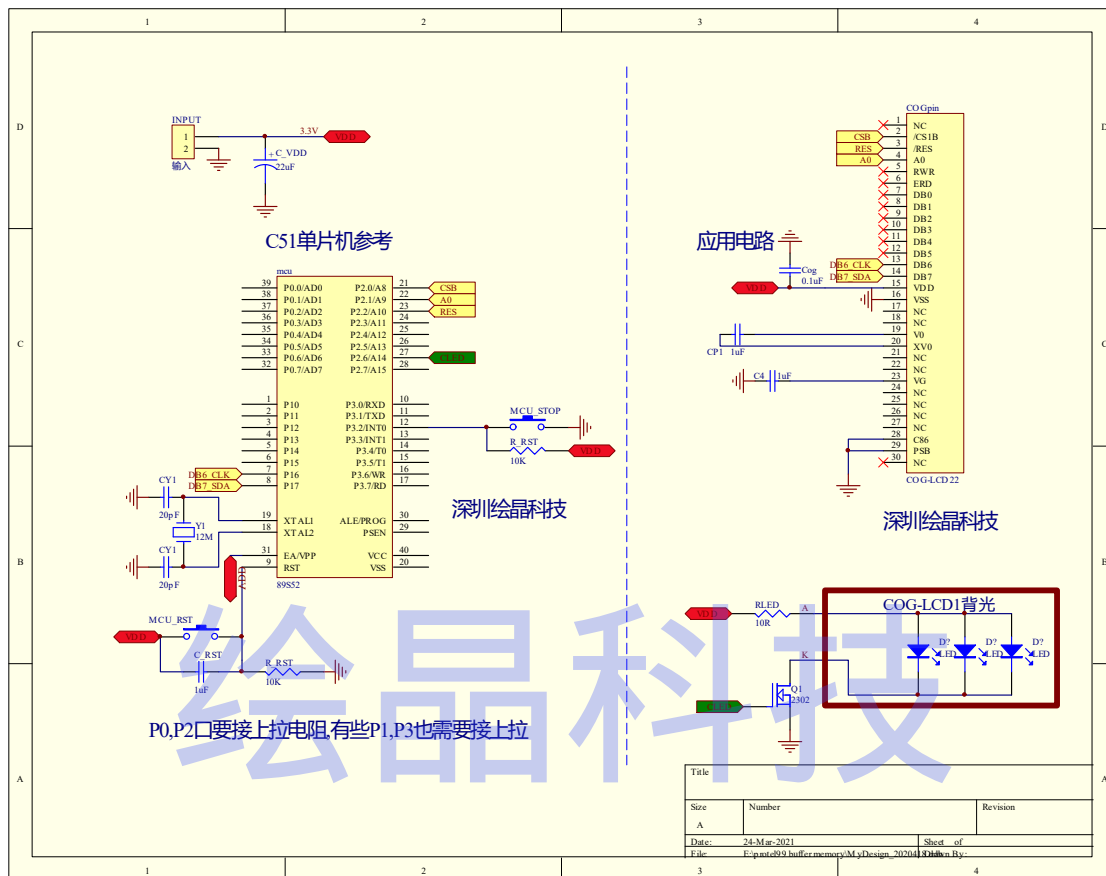
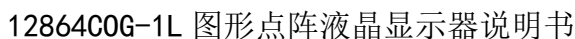
二十二. 空操作设置

二十三. 测试设置

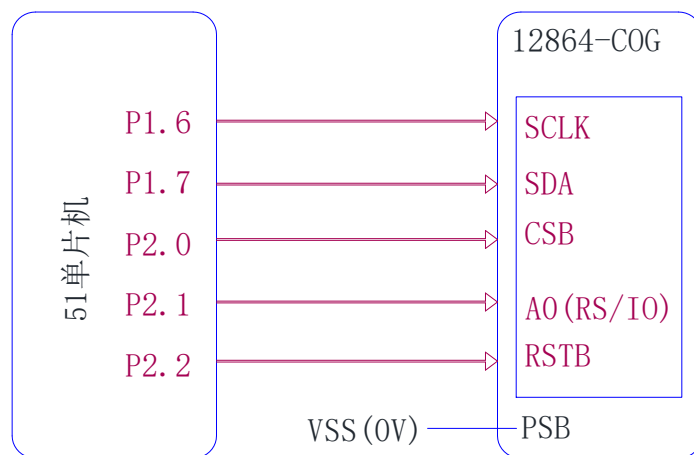
第八章、单片机与显示器连接说明

第一节、并口应用原理图

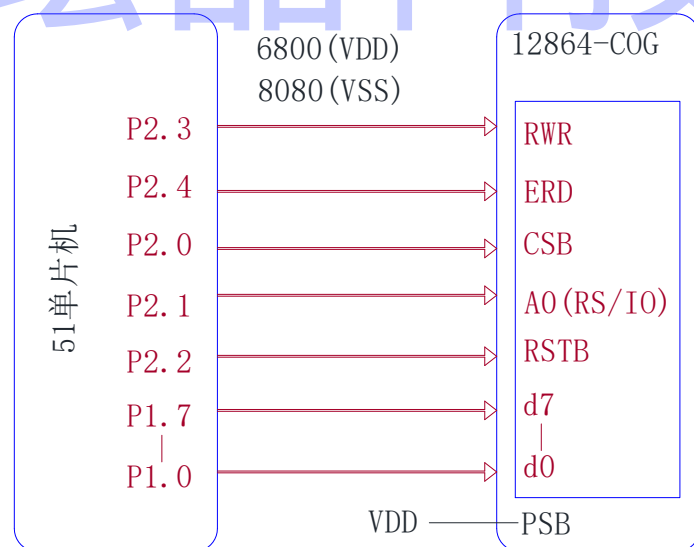




第三节、单片机串口连接图



第四节、单片机并口连接图



第九章、 单片机驱动程序源代码

第一节、源代码解释定义声明

//接口：串行/并行8080通用。

//产品内核ST7567R

//单片机ATMEL 89S52

//STC单片机完全兼容

#include <reg52.H>

#include <intrins.h>

sbit cs1=P2²;

sbit reset=P2¹;

sbit rs=P2⁰;

sbit rd=P2³;

sbit wr=P2⁴;

sbit lcd_sclk=P1⁶;

sbit lcd_sid=P1⁷;

#define uchar unsigned char

#define uint unsigned int

uchar m1, z, z1, d, d1, s, s1, s10, s100;

uchar code ascii_table_8x16[95][16];

uchar code ascii_table_5x8[95][5];

uchar code YEI1[];

uchar code jing1[];

uchar code ke1[];

uchar code ji1[];

void ini_int1(void)

{

EA=1;

EX0=1;//允许外部 INT0 的中断

IT0=1;// 允许中断

}

int scankey1() interrupt 0 using 1 //使用外部中断1, 寄存器组3

{

```
while(P3^2==0) {for(;;);}
    IE1=0;//中断标志清零

}
```

第二节、接口时序函数

```
//=====transfer command to LCM=====并口
```

```
void transfer_command(int data1)
{
    cs1=0;
    rs=0;
    //rd=0;
    wr=0;
    P1=data1;
    //rd=1;
    wr=1;
    rs=1;
    cs1=1;
    //rd=0;
}
```

```
//-----transfer data to LCM-----
```

```
void transfer_data(int data1)
{
    cs1=0;
    rs=1;
    //rd=0;
    wr=0;
    P1=data1;
    //rd=1;
    wr=1;
    rs=0;
    cs1=1;
    //rd=0;
}
```

```
//=====transfer command to LCM=====串口
```

```
/*
```

```
void transfer_command(int data1)
```

```
{  
char i;  
cs1=0;  
rs=0;  
for(i=0;i<8;i++)  
{  
lcd_sclk=0;  
//delay_us(10); //加少量延时  
if(data1&0x80) lcd_sid=1;  
else lcd_sid=0;  
lcd_sclk=1;  
//delay_us(10); //加少量延时  
data1=data1<<=1;  
}  
cs1=1;  
}
```

```
//-----transfer data to LCM-----
```

```
void transfer_data(int data1)
```

```
{  
char i;  
  
cs1=0;  
rs=1;  
for(i=0;i<8;i++)  
{  
lcd_sclk=0;  
if(data1&0x80) lcd_sid=1;  
else lcd_sid=0;  
lcd_sclk=1;  
data1=data1<<=1;  
}  
cs1=1;  
}
```

```
*/
```

```
//=====transfer command to LCM=====
```

```
/*延时*/  
void delay(int i)  
{  
    int j,k;  
    for(j=0;j<i;j++)  
        for(k=0;k<110;k++);  
}
```

第三节、液晶模块初始化

```
/*LCD 模块初始化*/  
void initial_lcd()  
{  
    csl=0;  
    reset=0; /*低电平复位*/  
    delay(20);  
    reset=1; /*复位完毕*/  
    delay(20);  
    transfer_command(0xe2); /*软复位*/  
    delay(50);  
    transfer_command(0x2c); /*升压步骤1*/  
    delay(50);  
    transfer_command(0x2e); /*升压步骤2*/  
    delay(50);  
    transfer_command(0x2f); /*升压步骤3*/  
    delay(50);  
    transfer_command(0x24); /*粗调对比度，可设置范围0x20~0x27*/  
    transfer_command(0x81); /*微调对比度*/  
    transfer_command(0x1b); /*0x1a, 微调对比度的值，可设置范围0x00~0x3f 1f*/  
    transfer_command(0xa2); /*1/9 偏压比 (bias) */  
    transfer_command(0xc8); /*行扫描顺序：从上到下*/  
    transfer_command(0xa0); /*列扫描顺序：从左到右*/  
    transfer_command(0x60); /*起始行：第一行开始*/  
    transfer_command(0xaf); /*开显示*/  
    csl=1;  
}
```

第四节、应用函数

```
void lcd_address(uchar page,uchar column)
{

    cs1=0;
    column=column; //
    page=page-1;
    transfer_command(0xb0+page);
    transfer_command(((column>>4)&0x0f)+0x10); //设置列地址的高4 位
    transfer_command(column&0x0f); //设置列地址的低4 位
}
```

/*全屏清屏*/

```
void clear_screen()
{
    unsigned char i,j;
    cs1=0;
    for(i=0;i<9;i++)
    {
        lcd_address(1+i,1);
        for(j=0;j<132;j++)
        {
            transfer_data(0x00);
        }
    }
    cs1=1;
}
```

```
void test_display(uchar data1,uchar data2)
{
    int i,j;
    for(j=0;j<8;j++)
    {
        cs1=0;
        lcd_address(j+1,1); //页，列地址
        for(i=0;i<64;i++)
        {
            transfer_data(data1);
            transfer_data(data2);
        }
    }
}
```

```
}
```

```
void display_string_8x16(uchar fb,uint page,uint column,uchar *text)
```

```
{
uint i=0,j,k,n;
csl=0;
while(text[i]>0x00)
{
if((text[i]>=0x20)&&(text[i]<=0x7e))
{
j=text[i]-0x20;

for(n=0;n<2;n++)
{
lcd_address(page+n,column);
for(k=0;k<8;k++)
{
if (fb==1) transfer_data(ascii_table_8x16[j][k+8*n]);
else transfer_data(~(ascii_table_8x16[j][k+8*n]));
}
}
i++;
column+=8;
}
else
i++;
}
}
```

```
void display_string_8x16_t(uchar fb,uint page,uint column,uchar text)
```

```
{
uint j,k,n;
j=text+16;
for(n=0;n<2;n++)
{
lcd_address(page+n,column);
for(k=0;k<8;k++)
{
if (fb==1) transfer_data(ascii_table_8x16[j][k+8*n]);
else transfer_data(~(ascii_table_8x16[j][k+8*n]));
}
```

```
    }  
  }  
  
}
```

```
void display_string_5x8(uint page,uint column,uchar *text)
```

```
{  
  uint i=0,j,k;  
  cs1=0;  
  while(text[i]>0x00)  
  {  
    if((text[i]>=0x20)&&(text[i]<0x7e))  
    {  
      j=text[i]-0x20;  
      lcd_address(page,column);  
      for(k=0;k<5;k++)  
      {  
        transfer_data(ascii_table_5x8[j][k]);  
      }  
      i++;  
      column+=6;  
    }  
    else  
      i++;  
  }  
}
```

```
void display_graphic_16x16(uchar fb,uchar page,uchar column,uchar *dp)
```

```
{  
  uint i,j;  
  for(j=0;j<2;j++)  
  {  
    lcd_address(page+j,column);  
    for (i=0;i<16;i++)  
    {  
      if(fb==1) transfer_data(*dp);  
      else transfer_data(~(*dp));  
    }  
  }  
}
```

```
dp++;  
}  
}  
}  
  
//显示128x64 点阵边框  
  
void display_bk()  
{  
uint i, j;  
  
//左框  
for(j=0; j<8; j++)  
{  
lcd_address(j+1, 1);  
transfer_data(0xff);  
}  
  
//右框  
for(j=0; j<8; j++)  
{  
lcd_address(j+1, 128);  
transfer_data(0xff);  
}  
  
//上框  
lcd_address(1, 2);  
for (i=0; i<126; i++)  
{  
transfer_data(0x01);  
}  
  
//下框  
lcd_address(8, 2);  
for (i=0; i<126; i++)  
{  
transfer_data(0x80);  
}  
}
```

绘晶科技

第五节、主调用函数

```
//~~~~~@~~~~&~~~~@~~~~程序开始
void main(void)
{

    ini_int1(); //开中断

    for(m1=0;m1<50;m1++)
    {

        initial_lcd();

        clear_screen(); //clear all dots

        test_display(0xaa, 0xaa);
        delay(700);
        //~~~~~@~~~~&~~~~@~~~~1屏, 横
        test_display(0x55, 0x55);
        delay(700);
        //~~~~~@~~~~&~~~~@~~~~2屏, 横
        test_display(0xff, 0x00);
        delay(700);
        //~~~~~@~~~~&~~~~@~~~~3屏, 竖
        test_display(0x00, 0xff);
        delay(700);
        //~~~~~@~~~~&~~~~@~~~~4屏, 竖
        test_display(0x55, 0xaa);
        delay(700);
        //~~~~~@~~~~&~~~~@~~~~5屏, 点
        test_display(0xaa, 0x55);
        delay(700);
        //~~~~~@~~~~&~~~~@~~~~6屏, 点
        test_display(0xff, 0xff);
        delay(700);
        //~~~~~@~~~~&~~~~@~~~~7屏, 黑

        clear_screen(); //clear all dots
        display_string_8x16(0, 1, 8, " YEIJINGKEJI");
        display_string_8x16(1, 3, 1, "WELCOME TO YEIJI");
```

```
display_string_5x8(5,1,"TEL: 0755-12346001");
display_string_5x8(6,1,"FAX: 0755-12346002");
display_string_5x8(7,1,"www.YEIJinglcm.com");
display_string_5x8(8,1,"YEIJinglcm@sina.com");
delay(700);
//~~~~~@~~~~~&~~~~~@~~~~~8屏，自建字符演示

clear_screen(); //clear all dots
display_bk();
display_graphic_16x16(1,3,33,YEI1);
display_graphic_16x16(1,3,49,jing1);
display_graphic_16x16(1,3,65,ke1);
display_graphic_16x16(1,3,81,ji1);
display_string_8x16(1,5,40,"128*64");
delay(700);
//~~~~~@~~~~~&~~~~~@~~~~~9屏，自建汉字图形
clear_screen(); //clear all dots

transfer_command(0xa7); //反显
display_graphic_16x16(1,3,33,YEI1);
display_graphic_16x16(1,3,49,jing1);
display_graphic_16x16(1,3,65,ke1);
display_graphic_16x16(1,3,81,ji1);
display_string_8x16(1,5,40,"128*64");
delay(700);
//~~~~~@~~~~~&~~~~~@~~~~~10屏，反白演示
clear_screen(); //clear all dots
transfer_command(0xa6);
}
initial_lcd();
clear_screen(); //clear all dots
display_graphic_16x16(0,1,0,YEI1);
display_graphic_16x16(0,3,0,jing1);
display_string_8x16(0,6,0,"TEL:755-12346001");
display_string_5x8(8,4,". WWW. YEIJINGLCM. COM. ");
display_string_5x8(1,40,"~~~~~");
display_string_5x8(4,40,"~~~~~");
```

第六节、动态显示函数

```
for(z=0;z<10;z++)
```

```
{
    display_string_8x16_t(0, 2, 40, z);
    for(z1=0; z1<10; z1++)
    {
        display_string_8x16_t(0, 2, 48, z1);
        for(d=0; d<6; d++)
        {
            display_string_8x16_t(0, 2, 56, 10);
            display_string_8x16_t(0, 2, 64, d);
            for(d1=0; d1<10; d1++)
            {
                display_string_8x16_t(0, 2, 72, d1);
                for(s=0; s<6; s++)
                {
                    display_string_8x16_t(0, 2, 80, 10);
                    display_string_8x16_t(0, 2, 88, s);
                    for(s1=0; s1<10; s1++)
                    {
                        display_string_8x16_t(0, 2, 96, s1);
                        delay(10);////延时 x ms
                        display_string_8x16_t(0, 2, 104, 10);
                        for(s10=0; s10<10; s10++)
                        {
                            display_string_8x16_t(0, 2, 112, s10);
                            delay(9);////延时 x ms
                            for(s100=0; s100<9; s100++)
                            {
                                display_string_8x16_t(0, 2, 120, s100);
                                delay(9);////延时 x ms
                            }
                        }
                    }
                }
            }
        }
    }
}

//~~~~~@~~~~~&~~~~~@~~~~~11屏，老化测试
}
```

第七节、标准字符数据表

uchar code ascii_table_8x16[95][16]={

篇幅较大，用户需要时请和业务联系

第十章、版本信息

更新说明

更新日期	更新内容说明	更新前日期
2017/2/11 10:45	优化	2015-8-12
2021/3/24	更换ST7567驱动芯片	

绘晶科技