

3510I 液晶简易文档 v1.5

wang_di@hit.edu.cn

1 描述

S1D15G14 是附带有 LCD 电源驱动电路的 LCD 驱动器，可在单片上实现彩色显示。

S1D15G14 可以直接连接到微处理器上，显示数据存储于“片上显示数据 RAM”中 (=DDRAM)。

并且 312 个段输出和 82 个公共输出可以驱动 LCD。它的 DDRAM 有 312×4 (16 级灰度) $\times 82$ 的容量。

LCD 上一个点与 DDRAM 的 4 个位对应，在单片上能够显示 104 (RGB) $\times 82$ 点。

而且可以用内建的电源供给电路生成精确的 LCD 驱动电压。

2 特性

LCD 驱动

- 312 段及 82 公共 LCD 驱动输出
- 4096 色，或 4096 色中的 256 色（正常模式），8 色（空闲模式）可以被显示
- PWM 灰度驱动以传统驱动方式驱动
- LCD 驱动 DUTY 可选：1/82DUTY，1/67DUTY
- 局部显示
- DDRAM 与 LCD 之间对应
DDRAM 的位数据 "0,0,0,0" --- 关
"1,1,1,1" --- 开
- *开/关表明在正常显示时，电压是否施加给 LCD
- DDRAM 容量： $312 \times 4 \times 84 = 104,832$ 位

MPU 界面

- 2 种串行接口可用：8 位，9 位 (D/C+8 位数据)
- 并行接口可用

内建电路

- LCD 电源供给电路
升压器，分压器及电压跟随器：占空比 1/9,1/8,1/7,1/6,1/5 可选
电压调整器：高精度
- 内建 CR 振荡器

电源供给

- 电源供给电压

输入/输出电源供给: VDDI-GND=1.6V 到 3.6V

内部电源供给: VDD-GND=2.35V 到 3.6V

LCD 驱动电源供给: V2-MV2=10.0V 到 25.5V

·电流消耗: 400uA(1/6 占空比, 85Hz 帧频, Vseg=3.3V,正常模式)

其他

◎封装模式: Au bump bare chip

◎操作温度范围宽: -40 到 +85°C

注意

◎本 IC 不能抵抗强烈的无线电/光学活动

3 功能描述

3.1 串行数据输入

串行输入输出命令和数据.然而,显示 RAM 中的内容不能被读取.

对于串行接口,下列两个模式可用.当 S89 针被设为高或低(高是 8 位,低是 9 位),这些模式可以被选择。

(1) 9 位接口

_CS,SCL 和 SIO 三针被使用.数据格式变成 D/_C(数据/命令识别位)+8 位,或 9 位单元

(2) 8 位接口

_CS,SCL,SIO 及 A0 四针被使用.数据格式为一个 8 位,数据/命令标识位由 A0 信号的高低决定

3.1.1 串行接口

写数据模式

(1) 9 位接口

当 _CS 变为低后,SCL 上升,SIO 的数据被接收。当 SCL 降低时,数据从 MPU 同步传送到 SIO 脚,S1D15G14 在 SCL 上升时从 SIO 脚接收信号.首先是数据/命令标识位的 D/C 信号,然后是 8 位数据,其中高位(D7)先传送到 SIO 脚。

数据传送完毕,设置 _CS 为高电平。

当命令发送完毕紧接着发送参数 P1,此时 _CS 脚变为高,而参数 P1 未发送完毕,产生传输延迟,这是参数 P1 并不被 S1D15G14 接收。

当设 _CS 脚为低后重始数据传送,之后的数据才被接收,并当作 P1。

◎1 在数据传送中显示 _CS 变为高

◎2 延迟的*1 被忽略,传输恢复后的参数*2 被认做参数 1。

(2)8 位接口,

_CS 变为低后,当 SCL 上升,SIO 数据被接收.

在 SCL 下降时,8 位数据首先是高位从 MPU 同步传送到 SIO 脚,

当 SCL 上升时,S1D15G14 从 SIO 脚接收信号,

此时,数据/命令由 A0 脚作标识.数据发送后,设置_CS 脚为高电平.

在数据传输中.当命令发送后,_CS 脚变为高,而参数 p1 正被发送,发送因此延迟,参数 P1 不被 S1D15G14 接收.设置_CS 脚为低后,重始数据发送,这时新数据被接收,作为 P1。

4 软件安装例程

下面展示软件安装例程.对于那些复位后默认值可以被使用而无需任何改变的命令,在下列例子里,不必要输入命令.

(1)电源打开的命令输入过程

输入 VDD 及 VDDI

↓

确保执行开电源复位(_RES=低)

↓

设_RES 为高并等 5ms(内部操作稳定化时间)

↓

软件复位(01h):开始复位操作,对 IC 内部设置.

↓

等 5ms

↓

INIESC(C6h):初始化 IC 内部设置

↓

<显示设置 1>

◎ REFSET(B9H)

设置 IC 内部状态

◎ 显示控制(B6H)

◎ 灰度位置设置 (B3H,B4H)

设置灰度规范

◎ 伽马曲线设置 (B5H)

选择灰度

◎ 公共驱动输出选择 (BDH)

设置输出位置及公共驱动输出次序

<电源供给安装>

◎ 电源控制 (BEH)

◎SLEEP OUT(11H)
◎电压控制 (BAH)
◎写对比 (25H)
◎温度梯度设置 (B7H)
◎BOOSTER 电压开 (03H)
<显示设置 2>

◎反转开 (21H) 或反转关 (20H)
◎部分区域 (30H)
◎垂直滚动定义 (33H)
◎垂直滚动开始地址 (37H)

<显示设置 3>
◎界面点格式 (3AH)
◎颜色设置 (2DH)
◎内存访问控制 (36H)
◎页地址设置 (2BH)
◎列地址设置 (2AH)
◎内存写 (2CH)
写数据到显存
↓
显示开 (29H)
显示开

(2)关电源的命令输入流程

@1 _RES 信号未使用时

显示关 (28H): 显示关
↓
睡眠 (10H): 睡眠
↓
关闭 VDD-VDDI 电源

注意: 为了给连接 LCD 电源供给电路的电容放电, 在关闭电源之前执行 SLEEP IN 命令设置 IC 睡眠。

当 LCD 电源供给电路的输出足够低, VDD-VDDI 电源关闭。

@2 _RES 信号使用时

执行电源开复位 (_RES=低)
↓
关闭 VDD-VDDI 电源

注意：当 LCD 电源供给电路的输出足够低，VDD-VDDI 电源关闭。

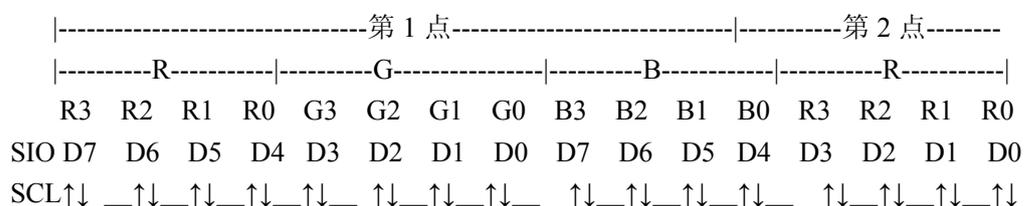
本 IC 使用 VDD-GND 及 VDDI-GND 电源供给的逻辑电压控制 LCD 输出驱动，因此，如果电源供给 VDD-GND,VDDI-GND 被关闭，此时 LCD 电源供给电路电压仍然维持，LCD 输出驱动(COM,SEG) 可产生无法控制的输出。

当 LCD 电源供给电路输出变得足够低，VDD-VDDI 电源关闭。

5 MPU 接口

本节阐述 MPU 读取 S1D15G14 显存的数据与实际存储在显存中数据的关系。

444 模式（显示 4096 色）



6 命令

(1) 空操作 (NOP)

00H

该命令能结束对显存的读写序列

该命令可以退出测试模式

所以推荐定期输入本指令

(2) 软复位 (SWRESET)

01H

本指令与硬复位（将_RES 脚置低）功能一致

(3) 调压关 (BSTOFF)

该指令关闭内建电源

02H

本命令执行时，LCD 驱动电源供给电路关闭

SLEEP IN/OUT 命令也可以打开/关闭 LCD 驱动电源供给电路，

但本指令是独立执行的。

03H 能够改变本指令的状态

◎复位后，调压开/关为关。

(4) 电源供给打开指令 (BSTON)

本命令执行时，LCD 驱动电源供给电路关闭

SLEEP IN/OUT 命令也可以打开/关闭 LCD 驱动电源供给电路，
但本指令是独立执行的。

◎复位后，调压开/关为关。

(5) 测试模式

IC 测试模式指令

04H

(6)读显示状态 (RDDST)

09H

(7)SLEEP IN (SLPIN)

睡眠状态

本指令使所有 LCD 输出脚为 VC 电平，LCD 驱动电源供给电路及内建振荡器关闭
因此，所有 LCD 驱动输出脚变为 GND 电平并静止。

在本指令之前的显示开关决定电路如何进入睡眠状态

显示开时：

命令后的 2 到 3 帧内睡眠

显示关时：

立即进入睡眠

(8) SLEEP OUT(SLPOUT)

取消睡眠状态

11H

命令之前的显示开关状态能够决定本电路如何退出睡眠状态

显示开时：

指令后 3 帧，显示开。

显示关时，

40MS 或更久才开显示

(9)

部分模式开 (PTLON)

12H

13H 时，本指令退出，进入正常显示模式

使用部分区域命令设置部分区域

当然，本命令也不可以访问显存

◎复位后，正常显示模式状态开。

(10) 正常模式开(NORON)

13H

◎复位后，默认为开

(11) 显示反转关

20H

意味着加到 LCD 的电压为最大值（当 RAM 数据为“1111”）

该命令无需改变显存

◎复位后默认关

（12）显示反转开

21H

◎复位后默认关

（13）所有点关

22H

该命令关闭所有 LCD 显示

意味着加在 LCD 的电压最大

执行后，RAM 访问停止，LCD 输出固定为低电平，不管其他设置。

下列命令可以解除该状态

部分模式开 12H

正常模式开 13H

◎复位后，所有点状态为 OFF，显示 OFF 状态

（14）

所有点开

23H

（15）写对比度（WRCNTR）

执行本指令改变 LCD 驱动电压输出到段/公共驱动器

25H

命令后跟参数

0 到 127（最高位 0）

中心值为 63，参数大时，LCD 电压升高（黑），反之降低（亮）

◎复位后默认值 63

（16）显示关

关闭 LCD 显示

执行后，RAM 访问停止，输出变为：

段：低电平不管 RAM 数据与其他设置

公共：与显示开状态一致

28H

显示开能够取消本状态

◎复位后，显示状态为关

（17）显示开

29H

执行后，显示显存内数据，显示设置也会输出到 LCD 驱动器

◎复位后，显示状态为关

(18)列地址设置 (CASET)

当显示数据从 MPU 传输显存，该指令用来设置一个写区域
列地址扫描时，地址从起始地址到终止地址递增，然后页地址增 1
之后列地址返回起始地址

当执行本指令，同时设置起始列与终止列，以使起始列小于终止列

当然，如果列地址设置到显存区域以外，数据写操作将被忽略，读操作将是错误数据

2AH

输入指令后，根据 2 个参数来设置 8 位起始与终止地址

XXH

XXH

◎复位后默认值：

起始：0

终止：103 (RAMDIV 脚=LOW)，97 (RAMDIV 脚=HIGH)

(19) 页地址设置

2BH

XXH

XXH

◎复位后默认值

起始页：0

结束页：83 (RAMDIV 脚=LOW)，66(RAMDIV 脚=HIGH)

(20) 内存写 (RAMWR)

2CH

之后开始写数据

任意其他命令可以结束本指令状态

(21)

2DH 颜色设置

(RGBSET)

指令设置颜色检索表

该表在 256 色时使用 (界面颜色点格式)

发送格式如下

2DH

XXH----XXH

(8R)

XXH----XXH

(8G)

XXH----XXH

(4B)

总共发送 20 个字节

使用时 1 个字节表示 1 点颜色

RRRGGGBB

其中 RRR,GGG,BB 是检索表序号

(22) RAM 数据读

2EH

(23)部分区域 (PLTAR)

命令设置显示区域

30H

开始行 8 位

结束行 8 位

对于非可视区域: S1D15G14 显示输出关闭, 内存访问禁止

当起始行小于结束行, 显示区域是屏幕中一个矩形

反之, 不可视区域是屏幕中一个矩形

(24)

垂直卷轴定义 (VSCRDEF)

33H

有 3 个参数

第 1 个, 顶固定行数

第 2 个, 卷轴行数

第 3 个, 底固定行数

其中卷轴区域是可滚动的。

注意: 顶固定区域根据内存访问控制命令的 B4 位可变

◎默认值复位后的:

0

0

53H

(25) 测试模式

34H

(26)测试模式

35H

35H

(27)内存访问控制 (MADCTL)

36H

设置 MPU 访问内存方式

1 个参数

B7 B6 B5 B4 B3 0 0 B0 XXH

B7: 0 从顶算 0 页 (0 行)

1 从底算 0 页

B6: 0 列从左到右递增

1 反之

B5: 0 正常模式一行行写

1 反转模式一列列写

B4: 设置扫描方向 0 从顶到底 1 从底到顶

B3: 写 RGB 的次序: 0 RGB 1 BGR

B0: 0 正常 1 垂直反转

注意: DISCTL 的参数 P32 定义了顶和底的值

(28) 垂直卷轴始地址(VSCRSADD)

37H

该指令用于垂直卷轴时。设置的卷轴地址下一帧有效。

(29) 空闲模式关

38H

(30) 空闲模式开

39H

用于 8 色显示 (RGB 的最高 4 位)

(31) 接口点格式

3AH

参数为 2—256 色

参数为 3—4096 色

(32) 测试模式

DEH

引起 4 种变化

◎有选择的极性输出到不止一个公共端

◎LCD 参考电压的色温变化

◎震荡停止

◎正常振荡频率变化, LCD 参考电压变化

(33) NOP2

AAH

空操作, 可以退出测试模式

(34) 初始化退出 C6H

(35) 测试模式

DAH

(36) 测试模式

DBH

(37) 测试模式

DCH

(38)测试模式

B2H

(39)(40) 灰度位置设置

B3H

后面跟 15 个 8 位 GCP0

标明 0001——1111 所代表的灰度

B4H

后面跟 15 个 8 位 GCP1

因为 IC 提供 2 个序列的寄存器。

注意：

1 在执行 SLEEP OUT 之前设置该寄存器，显示时不要改变

2

3 确保 15 个参数递增

4 输出 0000 是固定的

©复位后寄存器值不恒定，需执行初始化

(41)

伽马曲线设定

B5H

后面跟 1 个参数

1: GCP0

2:GCP1

3:保留上次设置

(42)

B6H 显示控制