# 简介

SSD1306是一个单片CMOS OLED/PLED驱动芯片可以驱动有机/聚合发光二极管点阵图形显示系统。由128 segments 和64 Commons组成。该芯片专为共阴极OLED面板设计。

SSD1306中嵌入了对比度控制器、显示RAM和晶振，并因此减少了外部器件和功耗。有256级亮度控制。数据/命令的发送有三种接口可选择：6800/8000串口，I2C接口或SPI接口。适用于多数简介的应用，注入移动电话的屏显，MP3播放器和计算器等。

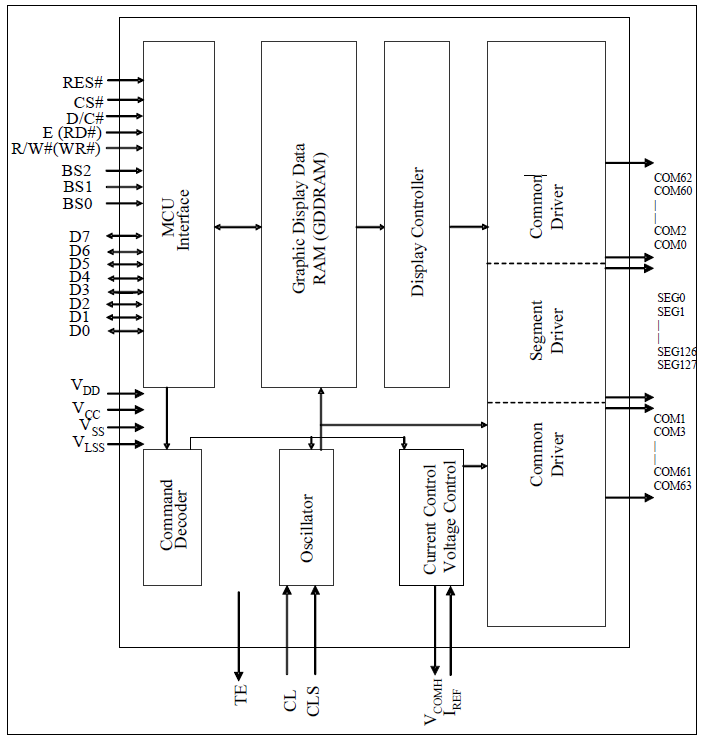
# 特性

1. 分辨率：128 \* 64 点阵面板
2. 电源：
   1. VDD = 1.65V to 3.3V 用于IC逻辑
   2. VCC = 7V to 15V 用于面板驱动
3. 点阵显示
   1. OLED驱动输出电压，最大15V
   2. Segment最大电流：100uA
   3. 常见最大反向电流：15mA
   4. 256级对比亮度电流控制
4. 嵌入式128 \* 64位SRAM显示缓存
5. 引脚选择MCU接口
   1. 8位6800/8000串口
   2. 3/4线SPI接口
   3. I2C接口
6. 水平和垂直两个方向的屏幕保存连续滚动功能。
7. RAM写同步信号
8. 可编程的帧率和多重比率
9. 行重映射和列冲映射
10. 片上晶振
11. 两种封装 COG和COF
12. 工作温度范围广：-40℃ to 85℃

# 订购信息

暂不翻译

# 结构方框图



# 功能块描述

## MCU接口选择

SSD1306MCU接口由8个数据引脚和5个控制引脚组成。引脚分配由不同的接口选择决定，详情如下表。不同的MCU模块可以通过BS[2:0]引脚的硬件选择设置。

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 引脚  总线接口 | Data/Command 接口 | | | | | | | | | | 控制信号 | | | | |
| D7 | D6 | D5 | D4 | D3 | D2 | | D1 | D0 | | E | R/W# | CS# | D/C# | RES# |
| 8-bit 8080 | D[7:0] | | | | | | | | | | RD# | WR# | CS# | D/C# | RES# |
| 8-bit 6800 | D[7:0] | | | | | | | | | | E | R/W# | CS# | D/C# | RES# |
| 3-wire SPI | Tie LOW | | | | | | NC | SDIN | | SCLK | Tie LOW | | CS# | Tie LOW | RES# |
| 4-wire Spi | Tie LOW | | | | | | NC | SDIN | | SCLK | Tie LOW | | CS# | D/C# | RES# |
| I2C | Tie LOW | | | | | | SDAout | SDAin | | SCl | Tie LOW | | | SA0 | RES# |

### MCU 并口 6800系列接口

不翻译

### MCU 并口8080系列接口

不翻译

### MCU串口（4-wire SPI）

不翻译

### MCU串口（3-wire SPI）

不翻译

### MCU I2C 接口

I2C通讯接口由从机地址为SA0，I2C总线数据信号（SDAout/D2输出和SDAin/D1输入）和I2C总线时钟信号SCL（D0）组成。数据和时钟信号线都必须接上上拉电阻。RES#用来初始化设备。

1. 从机地址位（SA0）

SSD1306在发送或接受任何信息之前必须识别从机地址。设备将会响应从机地址，后面跟随着从机地址位（SA0位）和读写选择位（R/W#位），格式如下：

b7 b6 b5 b4 b3 b2 b1 b0

0 1 1 1 1 0 SA0 R/W#

SA0位为从机地址提供了一个位的拓展。0111100或0111101都可以做为SSD1306的从机地址。D/C#引脚作为SA0用于从机地址选择。R/W#为用来决定I2C总线接口的操作模式。R/W# = 1，读模式。R/W# = 0 写模式

1. I2C总线数据信号SDA

SDA作为发送者和接受者之间的通讯通道。数据和应答都是通过SDA发送。

应该注意的是ITO轨道电阻和SDA引脚上的上拉电阻会变成一个潜在的电压分压器。结果就是SDA上的应答可能不能达到有效的逻辑0。

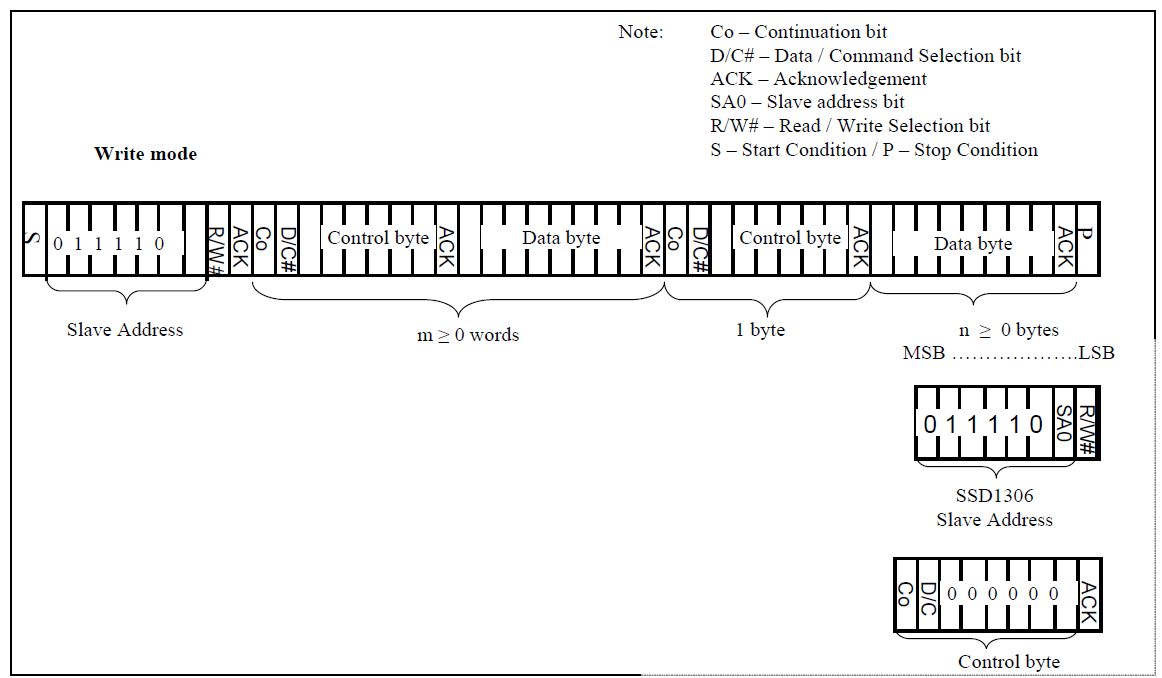
SDAin和SDAout绑定到了一起作为SDA。SDAin引脚必须连接到SDA。SDAout引脚可以不连接。当SDAout引脚不连接，应答信号将会被I2C总线忽略。

1. I2C总线时钟信号SCL

I2C总线上的信息传输任务是随着一个时钟信号的，SCL。每个数据位的传输任务发生在SCL的单个的时钟周期中。

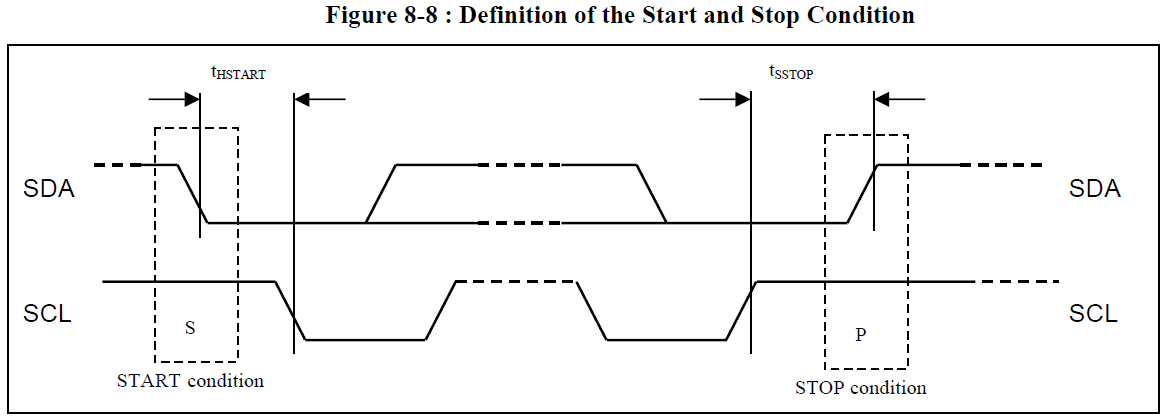
#### I2C总线写数据

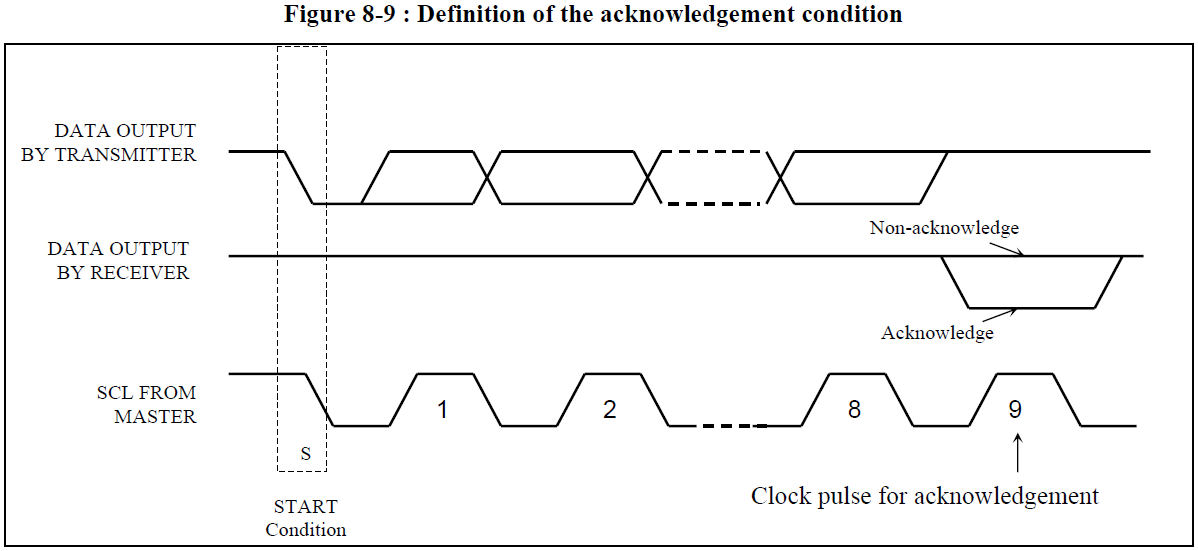
I2C总线接口提供了写数据和命令到设备的接口。关于I2C总线的写模式的时序请参考下图：



##### I2c的写模式

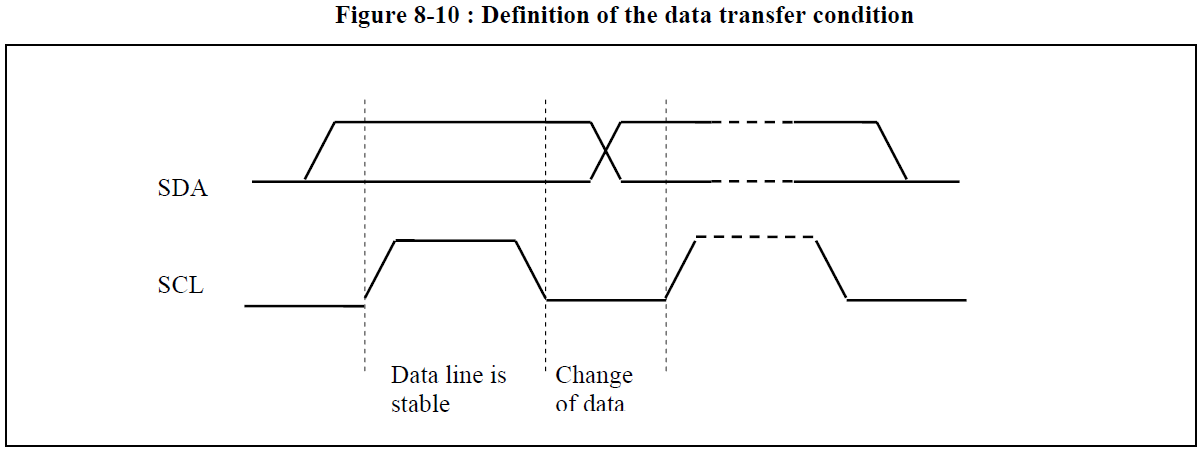
1. 主机设备通过开始条件初始化数据通讯。开始条件的定义在下面的图中展示。开始条件通过将SDA从高拉低而SCL保持高建立。
2. 从机地址紧跟着开始条件作为标志用。对于SSD1306，从机地址可以是“b0111100”或“b0111101”通过改变SA0到LOW或HIGH（D/C引脚作为SA0）。
3. 写模式的建立是通过设置R/W#为到逻辑0实现的。
4. 应答位信号将在接收到一个字节的数据后发生包括从机地址和R/W#位。可以参考下面的应答位信号的图示。应答位的定义是在应答位相关的时钟脉冲的高周期中把SDA线拉低。
5. 传输完从机地址之后，控制字节或数据字节开始通过SDA传输。一个控制字节主要由CO和D/C#位后面再加上六个0组成的。
   1. 如果Co为设为逻辑0，，后面传输的信息就只包含数据字节。
   2. D/C为决定了下个数据字节是作为命令还是数据。如果D/C#位设为0，下一个数据就是命令。如果D/C#位设为1，下一个数据就是数据，将会存在GDDRAM中。GDDRAM列地址指针将会在每次数据写之后自动加1.
6. 应答位将会在接收到每个控制字节或数据字节之后发生
7. 写模式会在用了停止条件之后结束。停止条件：将SDA从低拉高，而SCL保持高。





请注意数据位的传输是有限制的

1. 数据位，在每个SCL脉冲阶段传输，必须在时钟脉冲高周期内保持稳定状态。可以参考下面的图示。除了开始或停止条件，数据线只能在SCL为LOW的时候转换状态。
2. SDA和SCL都必须接外接上拉电阻



## 命令解码器

这个模块决定了输入数据被解读为数据还是命令。数据的解读是基于D/C#引脚的输入。

如果D/C#引脚是高，D[7:0]就被解读为写到图像显示数据RAM（GDDRAM）中的显示数据。

如果是低，D[7:0]的输入就被解读为一个命令。然后数据输入就会被解码并写到相关的命令寄存器中。

## 晶振电路和显示时间发生器

这是一个片上低功耗RC震荡电路。操作时钟（CLK）可以由内部震荡器或外部CL引脚生成。这个选择是通过CLS引脚完成的。当CLS引脚拉高，内部振荡器开启，CL应该保留开启。将CLS拉低可以静止内部振荡器，外部时钟必须连接到CL引脚上。当选择了内部振荡器之后，输出频率Fosc可以通过命令D5h A[7:4]修改。

用于显示时钟发生器的显示时钟（DCLK）是源于CLK的。分频因子“D”可以通过命令D5h编程范围为1到16

DCLK = Fosc/D

显示的帧率是由下面的公式决定

D 代表时钟分频。由D5h A[3:0]设置。范围是1到16

K是每行显示时钟的数量。值的计算如下：

K = 相1周期 + 相2周期 + BANK0脉冲宽度 = 2 + 2 + 50 = 54 在上电重启时

Number of multiplex ratio 通过command A8h设置。上电重启值为63 。

Fosc 是振荡器频率，可以通过命令D5h A[7:4]修改。值越大频率越大。

## FR同步

不翻译

## 重启电路

当RES# 输入为LOW时，芯片初始化为下面的状态：

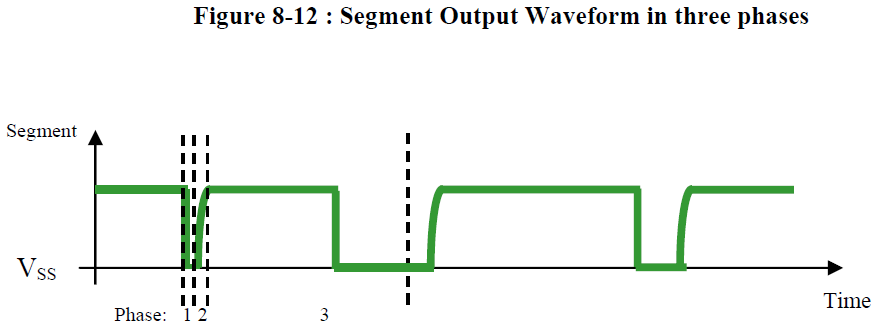
1. 显示关
2. 128 \* 64显示模式
3. 正常的segment和显示数据列地址和行地址映射（SEG0映射到address 00h，COM0映射到address 00h）
4. 在串口中清除移位寄存器数据
5. 显示开始行设置为显示RAM地址0
6. 列地址计数设为0
7. 正常扫描COM输出方向
8. 对比度控制寄存器设为7Fh
9. 正常显示模式（等同于A4h命令）

## Segment 驱动/Common驱动

Segment驱动实现了128个电路源来驱动OLED面板。驱动电流的范围可以在0到100uA以内调节，有256个等级。Common驱动产生电压脉冲。

segment驱动波形分为3个阶段：

1. 在阶段1，OLED用于显示上一个图片的像素停止充电目的是为了准备显示下一张图片的内容。
2. 在阶段2，OLED像素被充到目标电压。像素被从Vss驱动到相关电压水平。阶段2时间可以通过编程在1到15个DCLK内变化。如果OLED面板的像素的电容值比较大，那么就需要更长的时间周期来给电容充电以达到想要的电压。
3. 在阶段3，OLED驱动转换为使用电流源来驱动OLED像素，这就是电流源驱动阶段了。

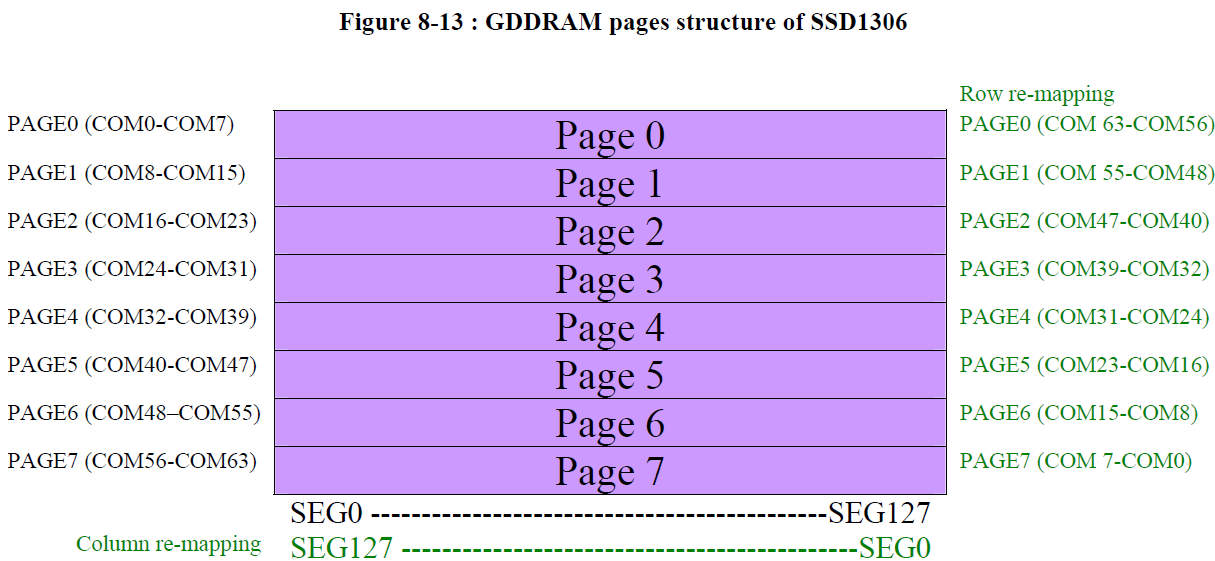


当完成阶段3之后，驱动芯片会回到阶段1来显示下一行图像信息。这三部循环持续运行来刷新OLED面板上的图像显示。

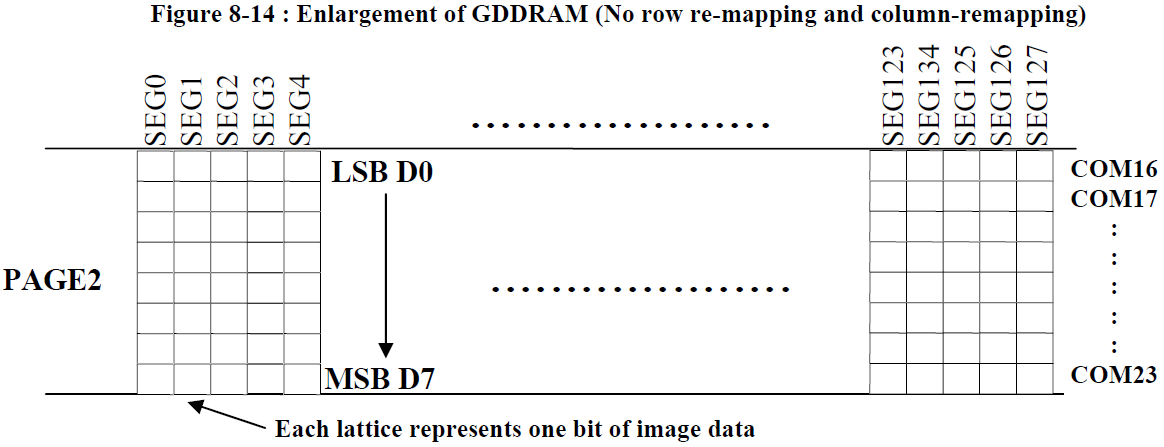
在阶段3，如果电流驱动脉冲长度设为50，在电流驱动阶段完成50个DCLK后，驱动芯片回到阶段1准备下一行的显示。

## 图形显示数据RAM（GDDRAM）

GDDRAM是一个为映射静态RAM保存位模式来显示。该RAM的大小为128 \* 64为，RAM分为8页，从PAFE0到PAGE7，用于单色128 \* 64点阵显示，如下图所示



当一个数据字节写到GDDRAM中，所有当前列的同一页的行图像数据都会被被填充（比如，被列地址指针指向的整列（8位）都会被填充）。数据位D0写到顶行，而数据位D7写到底行，如下图所示。



为了灵活性，segment和common上的重映射都可以通过软件选择。

对于显示的垂直移动，一个内部的寄存器保存了显示开始行可以用来设置控制RAM数据部分来映射显示（命令D3h）。

## SEG/COM驱动块

不翻译

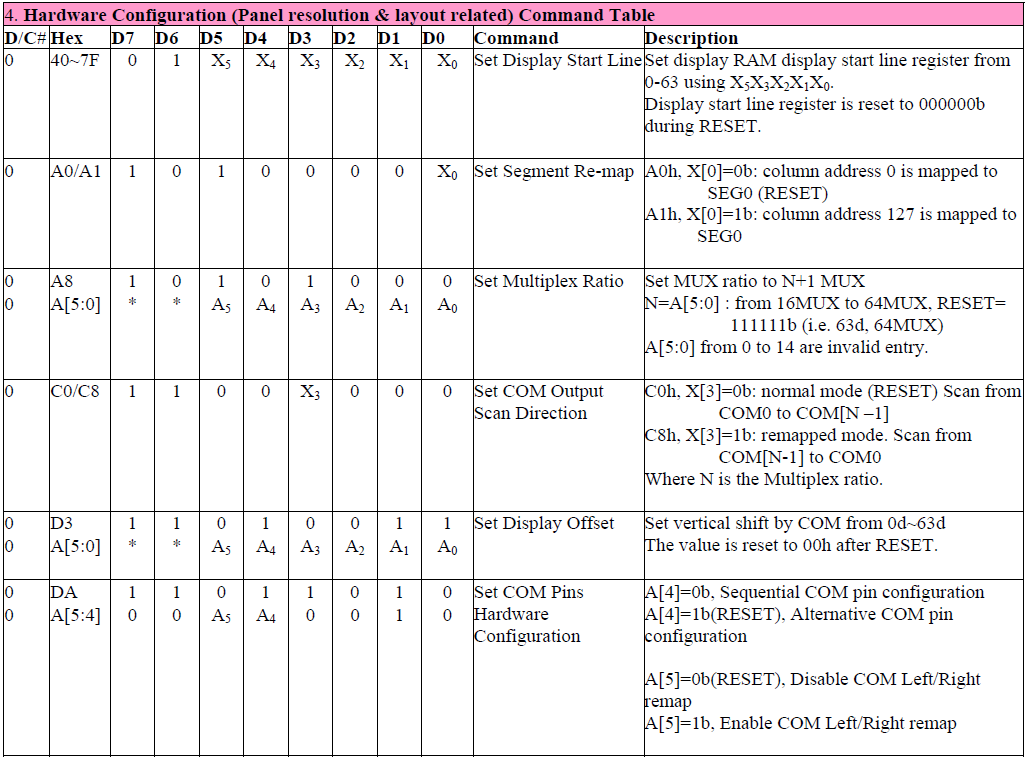
## 上电和断电顺序

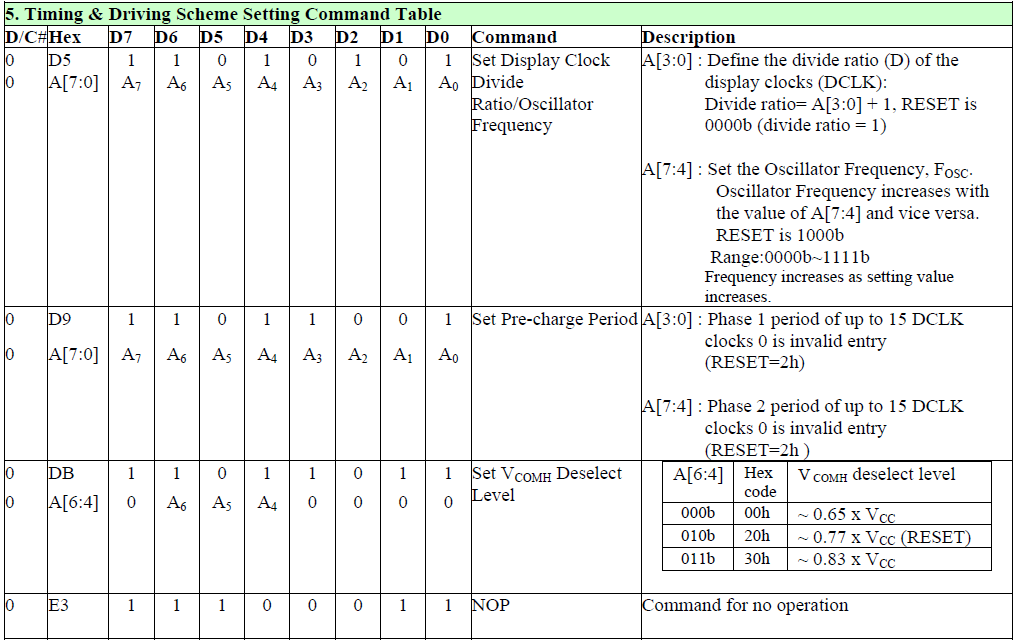
不翻译

# 命令表

（D/C# = 0， R/W#(WR#) = 0, E(RD# = 1)，除非有别的状态声明）

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1. 基本命令表 | | | | | | | | | | | |
| D/C# | Hex | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | 命令 | 描述 |
| 0  0 | 81  A[7:0] | 1  A7 | 0  A6 | 0  A5 | 0  A4 | 0  A3 | 0  A2 | 0  A1 | 1  A0 | 设置对比度 | 双字节命令选择256级对比度中的一种。对比度随着值的增加而增加  （RESET = 7Fh） |
| 0 | A4/A5 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | X0 | 整体显示开启状态 | A4h，X0 =0b：恢复RAM内容的显示（RESET）  输出跟随RAM  A5h， X0 =1b：进入显示开启状态，输出不管RAM内容 |
| 0 | A6/A7 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | X0 | 设置正常显示或反相显示 | A6h, X[0]=0b:正常显示（RESET）  在RAM中的0：在显示面板上为关  在RAM中的1：在显示面板上为开  A7h, X[0]=1b: 反相显示  在RAM中的0：在显示面板上为开  在RAM中的1：在显示面板上为关 |
| 0 | AE  AF | 1 | 0 | 1 | 0 | 1 | 1 | 1 | X0 | 设置显示开或关 | AEh, X[0]=0b:显示关（睡眠模式）  (RESET)  AFh X[0]=1b:显示开，正常模式 |
| 1. 滚屏命令 | | | | | | | | | | | |
| D/C# | Hex | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | 命令 | 描述 |
| 0  0  0  0  0 | 26/27  A[7:0]  B[2:0]  C[2:0]  D[2:0] | 0  0  \*  \*  \* | 0  0  \*  \*  \* | 1  0  \*  \*  \* | 0  0  \*  \*  \* | 0  0  \*  \*  \* | 1  0  B2  C2  D2 | 1  0  B1  C1  D1 | X0  0  B0  C0  D0 | 持续水平滚动设置 | 26h, X[0]=0, 向右水平滚动  27h, X[0]=1, 向左水平滚动  (水平平移1列)  A[7:0] : 空字节  B[2:0] : 定义开始页地址   |  |  |  | | --- | --- | --- | | 000b – PAGE0 | 011b – PAGE3 | 110b – PAGE6 | | 001b – PAGE1 | 100b – PAGE4 | 111b – PAGE7 | | 010b – PAGE2 | 101b – PAGE5 |  |   C[2:0] : 在帧率范围内设置每次滚屏的时间间隔   |  |  | | --- | --- | | 000b – 5 frames | 100b – 3 frames | | 001b – 64 frames | 101b – 4 frames | | 010b – 128 frames | 110b – 25 frame | | 011b – 256 frames | 111b – 2 frame |   D[2:0] : 定义结束页地址   |  |  |  | | --- | --- | --- | | 000b – PAGE0 | 011b – PAGE3 | 110b – PAGE6 | | 001b – PAGE1 | 100b – PAGE4 | 111b – PAGE7 | | 010b – PAGE2 | 101b – PAGE5 |  |   D[2:0]的值必须大于或等于B[2:0] |
| D/C# | Hex | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | 命令 | 描述 |
| 0  0  0  0  0  0 | 29/2A  A[2:0]  B[2:0]  C[2:0]  D[2:0]  E[5:0] | 0  0  \*  \*  \*  \* | 0  0  \*  \*  \*  \* | 1  0  \*  \*  \*  E5 | 0  0  \*  \*  \*  E4 | 1  0  \*  \*  \*  E3 | 0  0  B2  C2  D2  E2 | X1  0  B1  C1  D1  E1 | X0  0  B0  C0  D0  E0 | 持续垂直和水平滚屏设置 | 29h, X1X0=01b : 垂直和右水平滚屏  2Ah, X1X0=10b : 垂直和左水平滚屏  (水平滚动一列)  A[7:0] : 空字节  B[2:0] : 定义开始页地址   |  |  |  | | --- | --- | --- | | 000b – PAGE0 | 011b – PAGE3 | 110b – PAGE6 | | 001b – PAGE1 | 100b – PAGE4 | 111b – PAGE7 | | 010b – PAGE2 | 101b – PAGE5 |  |   C[2:0] :在帧频范围内设置每次滚屏的时间间隔   |  |  | | --- | --- | | 000b – 5 frames | 100b – 3 frames | | 001b – 64 frames | 101b – 4 frames | | 010b – 128 frames | 110b – 25 frame | | 011b – 256 frames | 111b – 2 frame |   D[2:0] :定义结束页地址   |  |  |  | | --- | --- | --- | | 000b – PAGE0 | 011b – PAGE3 | 110b – PAGE6 | | 001b – PAGE1 | 100b – PAGE4 | 111b – PAGE7 | | 010b – PAGE2 | 101b – PAGE5 |  |   D[2:0] 的值必须大于或等于B[2:0]  E[5:0] :垂直滚屏的位移  比如  E[5:0]= 01h offset =1 row  E[5:0] =3Fh offset =63 rows  **注意**  (1) 没有持续垂直滚屏功能 |
| D/C# | Hex | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | 命令 | 描述 |
| 0 | 2E | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 关闭滚屏 | 关闭因命令  26h/27h/29h/2Ah开启的滚屏功能  注意：  在使用2Eh命令来关闭滚屏动作后，RAM的数据需要重写 |
| D/C# | Hex | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | 命令 | 描述 |
| 0 | 2F | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 激活滚屏 | 开始滚屏，滚屏命令26h/27h/29h/2Ah配置的，有效的顺序是：  有效命令顺序1:26h；2Fh  有效命令顺序2:27h；2Fh  有效命令顺序3:29h；2Fh  有效命令顺序4:2Ah；2Fh  举例说明：如果发布命令26h；2Ah；2Fh，那么最后的一个滚屏命令会被执行，在这里是2Ah。换句话说最后的滚屏命令会重写以前的滚屏设置命令。 |
| D/C# | Hex | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | 命令 | 描述 |
| 0  0  0 | A3  A[5:0]  B[6:0] | 1  \*  \* | 0  \*  B6 | 1  A5  B5 | 0  A4  B4 | 0  A3  B3 | 0  A2  B2 | 1  A1  B1 | 1  A0  B0 | 设置垂直滚动区域 | A[5:0]设置顶层固定的行数。顶层固定区域的行数参考GDDRAM的顶部（比如row0）重置为0  B[6:0]设置滚动区域的行数。这个行的数量用于垂直滚动滚动区域。滚动区域开始于顶层固定区域的下一行。重置为64.  Note   1. A[5:0]+B[6:0] <=MUX ratio 2. B[6:0] <= MUX ratio 3. 垂直滚动偏移（29h/2Ah命令中的E[5:0]）< B[6:0] 4. 设置显示开始线（40h~7Fh中的X5X4X3X2X1X0）<B[6:0] 5. 滚动区域的最后一行移动到滚动区域的第一行 6. 对于64d最大显示   A[5:0]= 0，B[6:0]= 64整个区域滚动  A[5:0]=0，B[6:0]<64顶层区域滚动  A[5:0]+B[6:0]<64中心区域滚动  A[5:0]+B[6:0]= 64底部区域滚动 |
| 1. 地址设置命令表 | | | | | | | | | | | |
| D/C# | Hex | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | 命令 | 描述 |
| 0 | 00~0f | 0 | 0 | 0 | 0 | X3 | X2 | X1 | X0 | 设置低一点的列的开始地址作为页地址模式 | Set the lower nibble of the column start address  register for Page Addressing Mode using X[3:0]  as data bits. The initial display line register is  reset to 0000b after RESET. |
| 0 | 10~1F | 0 | 0 | 0 | 2 | X3 | X2 | X1 | X0 | 设置列的高地址作为页的开始地址 | Set the higher nibble of the column start address  register for Page Addressing Mode using X[3:0]  as data bits. The initial display line register is  reset to 0000b after RESET. |
| 0  0 | 20  A[1:0] | 0  \* | 0  \* | 1  \* | 0  \* | 0  \* | 0  \* | 0  A1 | 0  A0 | 设置内存地址模式 | A[1:0] = 00b, Horizontal Addressing Mode  A[1:0] = 01b, Vertical Addressing Mode  A[1:0] = 10b, Page Addressing Mode (RESET)  A[1:0] = 11b, Invalid |
| 0  0  0 | 21  A[6:0]  B[6:0] | 0  \*  \* | 0  A6  B6 | 1  A5  B5 | 0  A4  B4 | 0  A3  B3 | 0  A2  B2 | 0  A1  B1 | 1  A0  B0 | 设置列地址 | 设置列的开始和结束地址  A[6:0] : Column start address, range : 0-127d,  (RESET=0d)  B[6:0]: Column end address, range : 0-127d,  (RESET =127d) |
| D/C# | Hex | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | 命令 | 描述 |
| 0  0  0 | 22  A[2:0]  B[2:0] | 0  \*  \* | 0  \*  \* | 1  \*  \* | 0  \*  \* | 0  \*  \* | 0  A2  B2 | 1  A1  B1 | 0  A0  B0 | 设置页地址 | 设置页的开始结束地址  A[2:0] : Page start Address, range : 0-7d,  (RESET = 0d)  B[2:0] : Page end Address, range : 0-7d,  (RESET = 7d) |
| 0 | B0~B7 | 1 | 0 | 1 | 1 | 0 | X2 | X1 | X0 | 设置页开始地址作为页地址模式 | Set GDDRAM Page Start Address  (PAGE0~PAGE7) for Page Addressing Mode  using X[2:0]. |
| 硬件配置 | | | | | | | | | | | |
| D/C# | Hex | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | 命令 | 描述 |
|  |  |  |  |  |  |  |  |  |  |  |  |





* 1. 数据读写

为了往GDDRAM中写数据，在6800系列并口和8080系列并口模式中将R/W#（WR#）引脚置低，D/C#引脚置高。串口模式总是处于写模式。GDDRAM列地址指针在写完一个数据之后会自动加一。

10 命令描述

10.1 基本命令

10.1.1 设置低列开始地址作为页地址模式（00h~0Fh）

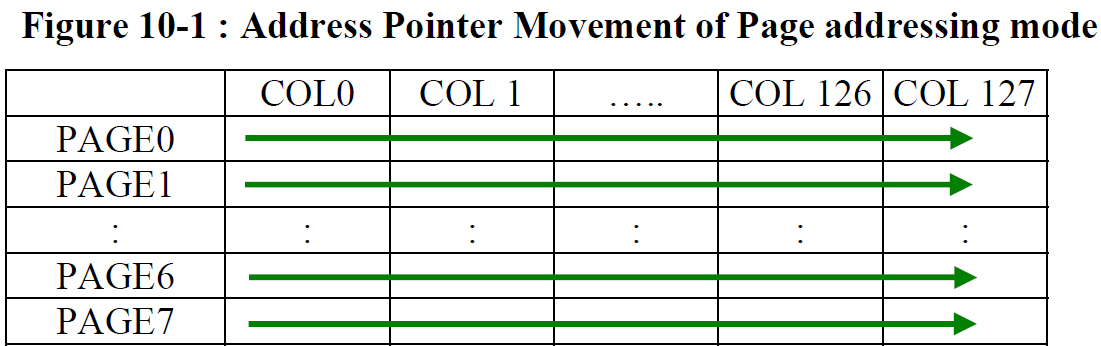
10.1.2 设置高列开始地址作为页地址模式（10h~1Fh）

10.1.3 设置内存地址模式（20h）

SSD1306中有三种不同的内存地址模式：页地址模式，水平地址模式，垂直地址模式。这个命令将内存地址模式设置成这三种中的一种。在这里COL的意思是图形显示数据RAM列。

页地址模式（A[1:0] =10xb）

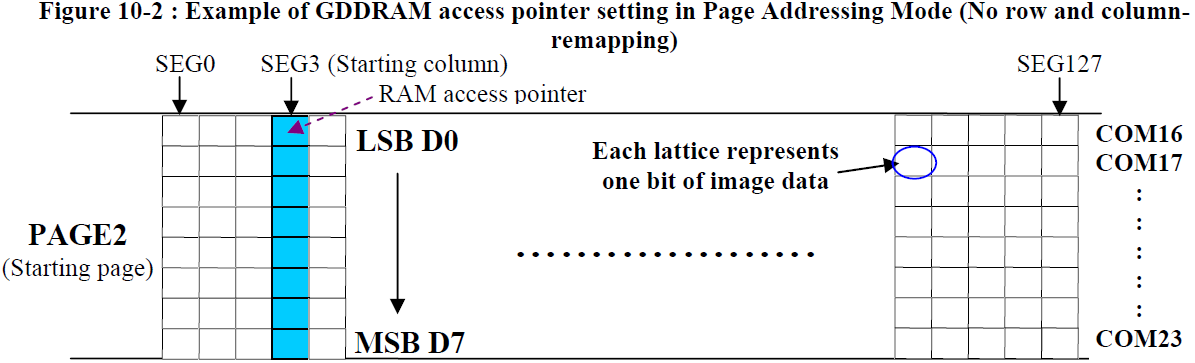
在页地址模式下，在显示RAM读写之后，列地址指针自动加一。如果列地址指针达到了列的结束地址，列地址指针重置为列开始地址并且也地址指针不会改变。用户需要设置新的页和列地址来访问下一页RAM内从。页地址模式下PAGE和列地址指针的移动模式参考下图



在正常显示数据RAM读或写和页地址模式，要求使用下面的步骤来定义开始RAM访问的位置：

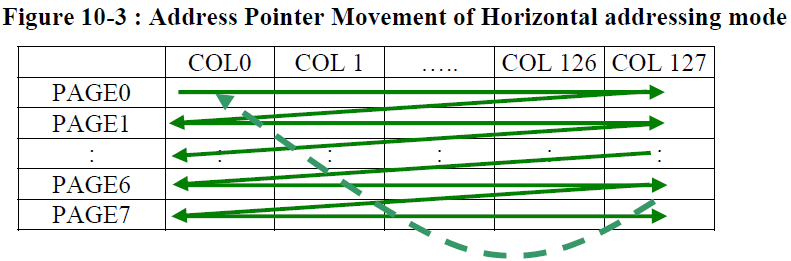
1. 通过命令B0h到B7h来设置目标显示位置的页开始地址
2. 通过00h~0Fh来设置低开始列地址的指针 )16\*16=256
3. 通过命令ENUM={10,11,12,13~~1Fh}//来设置高开始列地址

比如说，如果页地址设置为B2h，低列地址是03h高列地址为10h，那么就意味着开始列是PAGE2的SEG3.RAM访问指针的位置如下图所示。输出数据字节将写到RAM列3的位置。



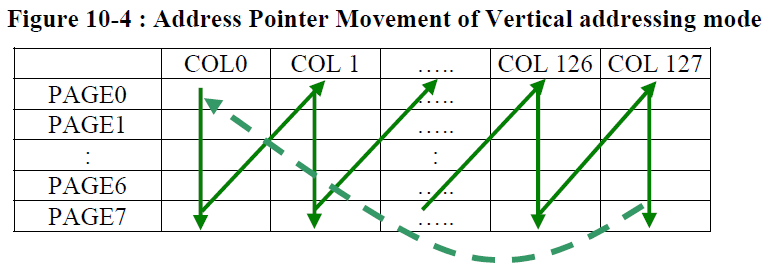
水平寻址模式（A[1:0]= 00b）

在水平寻址模式下，当显示RAM被读写之后，列地址指针自动加一。如果列地址指针达到列的结束地址，列地址指针重置为列的开始地址，并且页地址指针自动加1。水平寻址模式下页和列地址的移动顺序如下图所示。当列地址和页地址都达到了结束地址，指针重设为列地址和页地址的开始地址。



垂直寻址模式（A[1:0]=01b）

在垂直寻址模式下，当显示RAM被读写之后，页地址指针自动加一。如果页地址达到了页的结束地址，页地址自动重置为页的开始地址，列地址自动加一。页地址和列地址的移动顺序如下图所示。当列地址和页地址都达到结束地址后，指针自动重置为开始地址。



在正常显示RAM读或写，水平/垂直寻址模式下，要求用下面的步骤来定义RAM访问指针位置：

1. 用21h命令设置目标显示位置的列的开始和结束地址；
2. 用命令22h设置目标显示位置的页的开始和结束地址

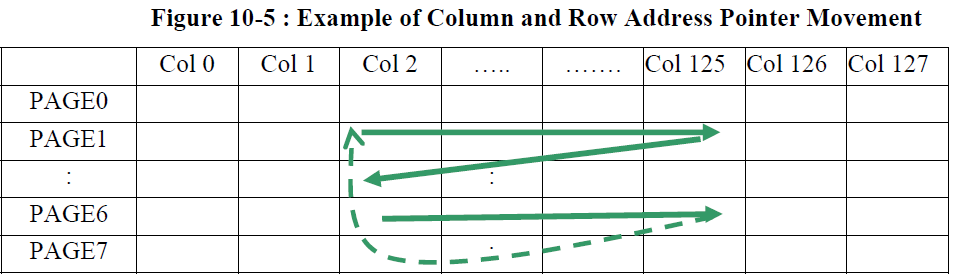
10.1.4 设置列地址（21h）

这个三字节命令指定了显示数据RAM列开始地址和结束地址。这个命令也会设置列地址指针到列开始地址。这个指针用于定义当前读写的显示数据RAM列地址。如果用20h命令开启了水平寻址命令，在完成读写一个列数据之后，该指针就会自动指向下一列地址。当什么时候列地址指针完成了访问列结束地址，它就会返回到列开始地址，和行地址增加到下一行。

10.1.5 设置页地址（22h）

这个三字节的命令指定了显示数据RAM页的开始和结束地址。这个命令同时也设置了页地址指针到页开始地址。这个指针用来定义在图像显示数据RAM中当前读写的地址。如果当前是垂直寻址模式，在读写完一页数据之后，会自动增加到下一页地址。当页地址指针完成访问结束页地址之后，会重置到开始页地址。

下面的插图展示了列和页地址指针的移动方式：列开始地址设置为2，列结束地址设置为125，页开始地址设置为1，页结束地址设置为6；通过命令20h使能水平地址增加模式。在这个案例中，图形显示数据RAM可访问的地址就只有从列2到列125，页1到页6。另外，列地址指针被设为2页地址指针被设为1.在完成读写数据中的一个像素后，列地址自动加1到下一个RAM位置进行下一次读写操作。当列地址指针完成访问结束列地址125之后，重新回到列2，页地址自动加1。当结束页6和结束列125RAM位置被访问过之后，页地址设回1，列地址设回2。



10.1.6 设置显示开始行（40h~7Fh）

这个命令设置显示开始行寄存器来决定显示RAM的开始地址，通过选择0到63的值。当值为0是，RAM行0映射到COM0，当值为1时，RAM行1映射到COM0，以此类推。

10.1.7 为BANK0设置对比度控制（81h）

这个命令用来设置显示的对比度，该芯片有256级对比度，从00h到FFh。屏显输出电流随着对比度增大而增大。

10.1.8 设置Segment 重映射（A0h/A1h）

这个命令修改显示数据列地址和segment驱动器之间的映射，允许在OLED模块设置上的灵活性。

这个命令只影响后续的数据输出。早已存储在GDDRAM中的数据不会改变。

10.1.9 全部显示开启（A4h/A5h）

A4h命令打开显示，输出依据GDDRAM中的内容。

如果使用A5h命令，然后再使用A4h命令，显示就会重覆盖为GDDRAM的内容。

换句话说，A4h命令从整体显示开启状态覆盖显示。

A5h命令轻质整体显示状态位开启，不管显示数据RAM中的内容。

10.1.10 设置正常/反相显示（A6h、A7h）

这个命令将显示设置成正常或反相模式。在正常模式1就是开，而在反相模式下1就是关

。

10.1.11 设置复用率（A8h）

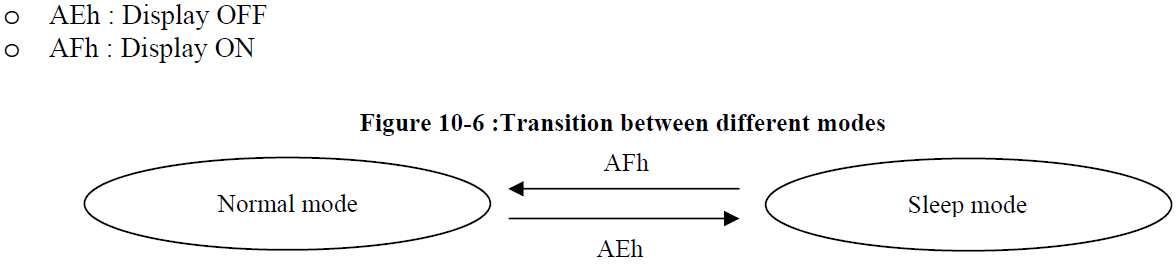
这个命令转换默认的63复用模式到任何复用率，范围从16到63。输出pads COM0~COM63将会转换为相关的COM信号。

10.1.12 设置显示开、关（Aeh、AFh）

这个单字节命令用来打开或关闭OLED面板的显示。

当显示是ON，通过设置主机配置命令选择的电路会打开；

当显示为OFF时，这些电路会关闭，segment 和 common输出将处于高阻状态。这些命令设置显示的状态在开关之间切换：



10.1.13 设置页开始地址作为页寻址模式（B0h~B7h）

……

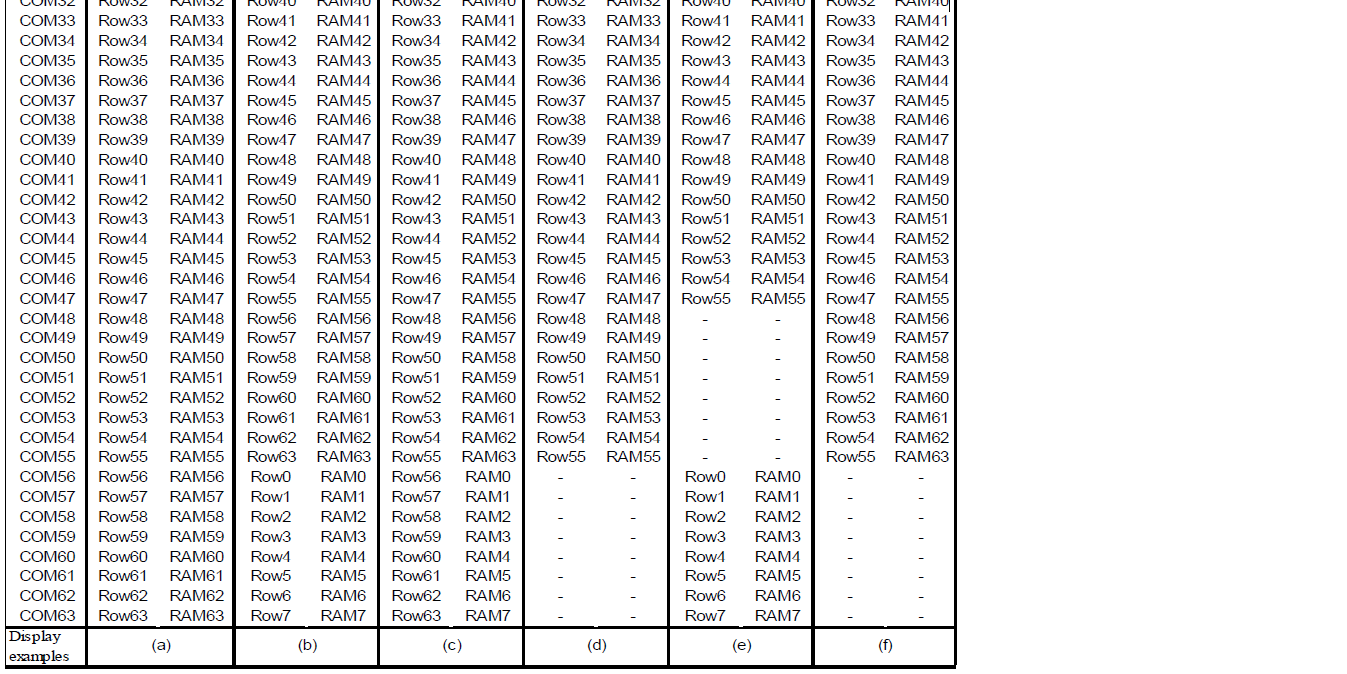
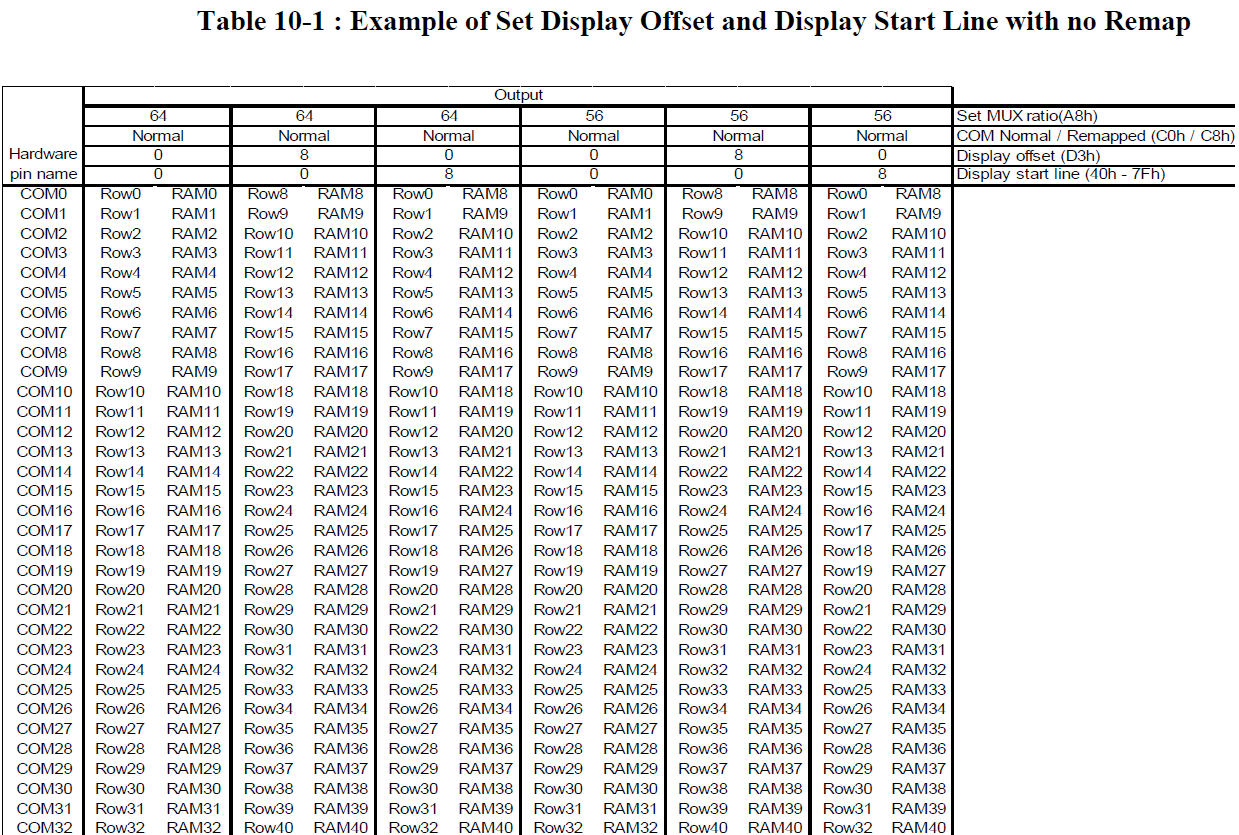
10.1.14 设置COM输出扫描方向（C0h、C8h）

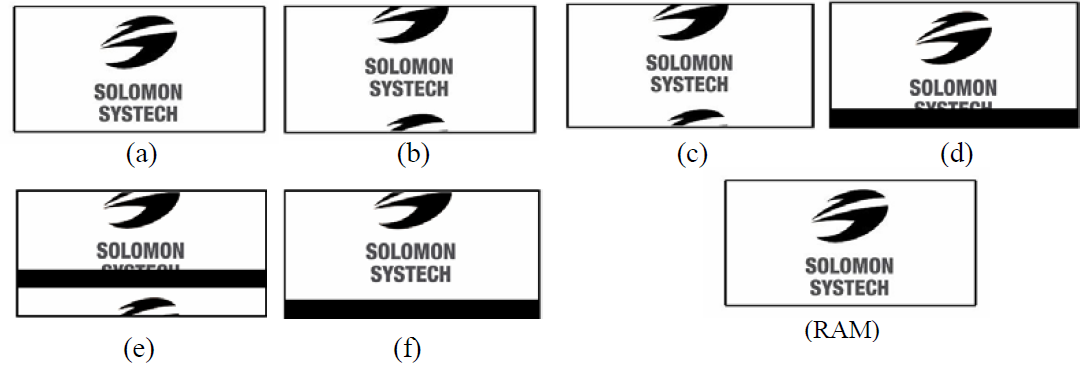
……

10.1.15 设置显示偏移（D3h）

这是一个两字节的命令。第二个命令指定显示映射的开始行到COM0和COM63中的一个（假设COM0是显示的开始行，那么显示开始寄存器就等于0）。

比如为了将COM16向COM0方向移动16行，第二个字节六位数据就应该写成010000b。为了向相反方向移动16行，这个六位数据就应该是64-16，所谓第二个字节应该是100000b。下面两个表展示了命令C0h/C8h和D3h的设置效果





10.1.16 设置显示时钟分频率、振荡器频率（D5h）

这个命令由两个功能组成：

显示时钟分频率D（A[3:0]）

设置分频率从CLK来生成DCKL（display clock）。这个分频率的范围为1到16，重置值为1.请参考8.3，获取DCLK和CLK关系的跟多细节

振荡器频率（A[7:4]）

如果CLS引脚置高编程后的振荡器频率Fosc就是CLK的源。这个4位的值设置16种不同的频率，默认设置为1000b。

10.1.17 设置重充电周期（D9h）

这个命令用于设置充充电周期的时间长度。间隔以计算DCLK的数量，重置值为2DCLK。

10.1.18 设置COM引脚硬件配置（DAh）。

这个命令设置COM信号引脚配置来匹配OLED面板硬件层。下面的表展示了不同条件下的COM引脚配置（服用率为64）

|  |  |
| --- | --- |
| 条件 | COM引脚配置 |
| 1 连续COM引脚配置（Dah A[4] = 0）  COM输出扫描方面，从COM0到COM63（C0h）  禁止COM左右重映射（Dah A[5] = 0） |  |
| 2 顺序COM引脚配置（Dah A[4]= 0）  COM输出扫描顺序：从COM0到COM63（C0h）  使能COM左右重映射（Dah A[5]= 1） |  |
| 3 连续COM引脚配置（Dah A[4]= 0）  COM输出扫描方向：从COM63到COM0（C8h）  使能COM左右重映射（Dah A[5]= 1） |  |
| 4 连续COM引脚配置（Dah A[4] = 0）COM输出扫描方向：从COM63 到COM0（C8h）使能COM左右重映射（Dah A[5] = 1） |  |
| 5 备选COM引脚配置（Dah A[4]= 1） COM输出扫描方向：从COM0到COM63（C0h）  禁能COM左右重映射（Dah A[5]= 0） |  |
| 6备选COM引脚配置（Dah A[4] = 1）COM输出扫描方向：从COM0 到COM63（C0h） 使能COM左右重映射（Dah A[5]= 1） |  |
| 7 备用COM引脚配置（Dah A[4] = 1）  COM 输出扫描方向：从COM63 到COM0（C8h）  禁能COM左右重映射（Dah A[5]= 0） |  |
| 8 备用COM引脚配置（Dah A[4] = 1）  COM输出扫描方向：从COM63 到COM0（C8h）  使能COM左右重影射 |  |

10.1.19 设置Vcomh 取消选择水平（DBh）

这个命令调整Vcomh调整器输出

10.1.20 NOP（E3h）

空指令

10.1.21 状态寄存器读

……

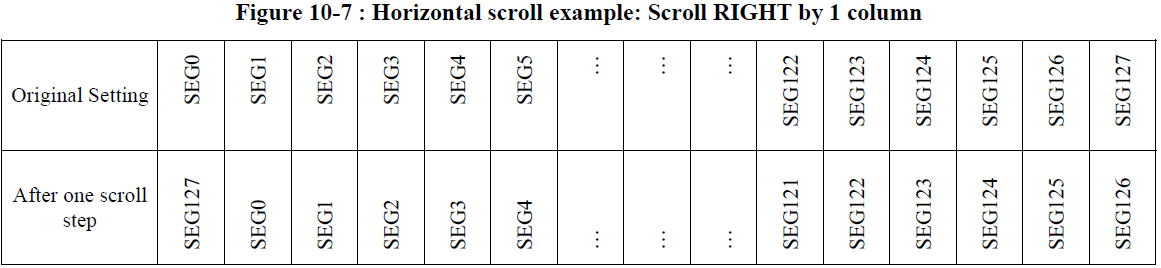
10.2 图形加速命令

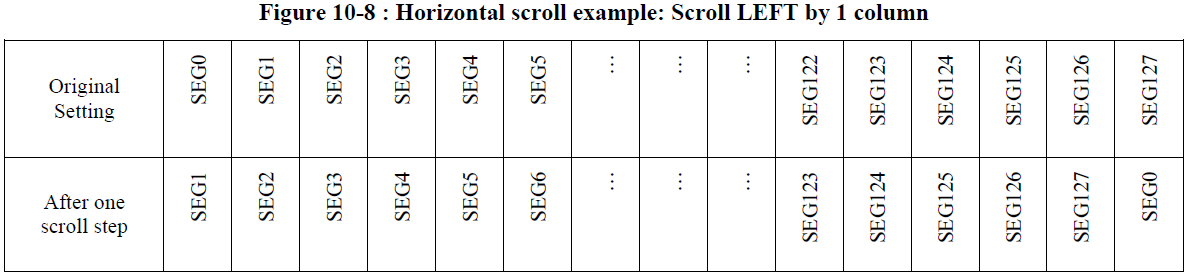
10.2.1 水平滚动设置（26h 27h）

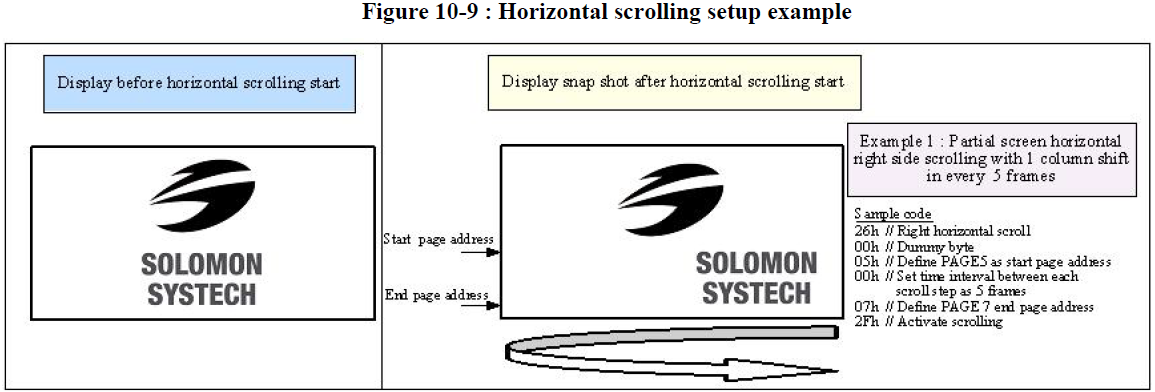
这个命令是由五个连续字节来设置水平滚动参数和决定滚动开始页，结束页和滚动速度的。

在声明这个命令前，水平滚动必须用命令（2Eh）关闭，否则，RAM中的内容可能被损坏。

SSD1306 水平滚动是为128列滚动设计的。下面的两张插图演示了滚动的效果：





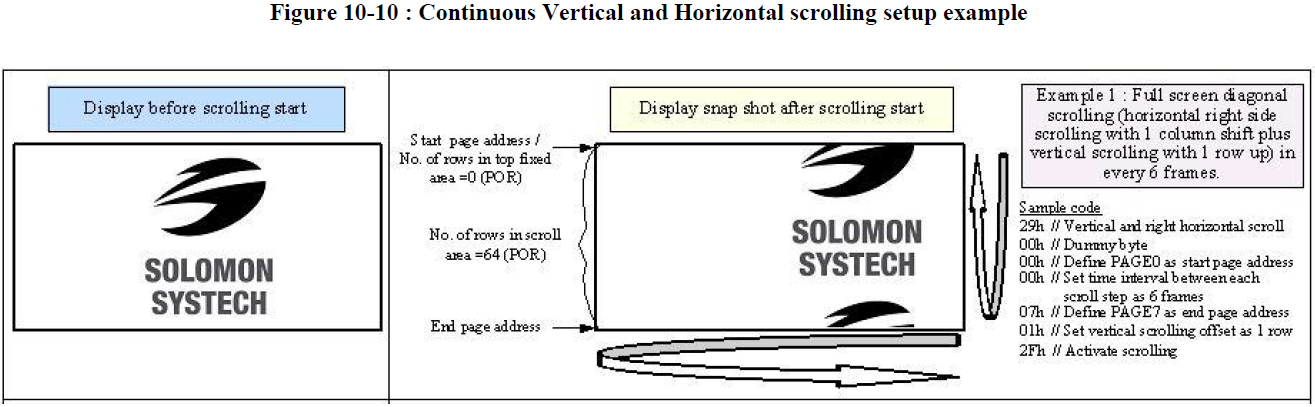


10.2.2 持续垂直和水平滚动设置（29h 2Ah）

这是一个由六个连续字节组成的命令，用来设置持续水平滚动参数和决定滚动开始页，结束页，滚动速度和垂直滚动偏移的。

命令29h、2Ah中的字节B[2:0],C[2:0]和D[2:0]用来设置持续水平滚动。字节E[5:0]用来设置持续垂直滚动的偏移量。所有这些字节一起设置持续倾斜（垂直+水平）滚动。如果垂直滚动偏移字节E[5:0]设为0，那么效果就只是水平滚动。

在发布此命令前必须用命令（2Eh）关闭滚动。否则，RAM内容会损坏。下面的插图展示了滚动效果：



10.2.3 关闭滚动（2Eh）

这个命令停止滚动的动作。在发布命令2Eh来关闭滚动动作之后，RAM内容需要重写。

10.2.4 激活滚动（2Fh）

这个命令开始滚动的懂做，并且只有在声明了滚动设置参数之后使用。只对最后的设置命令有效

下面的命令在滚动激活后是禁止的：

1. RAM 访问（数据读写）
2. 改变水平滚动设置参数

10.2.5 设置垂直滚动区域（A3h）

这个命令由三个连续的字节组成用来设置垂直滚动区域。对于持续垂直滚动功能（命令29h

2Ah），垂直滚动的行数可以设置的更小或等于最大复用率。

**I2C接口时间参数**

