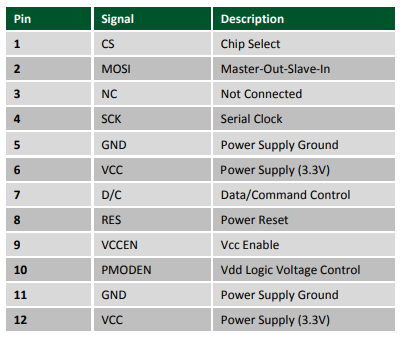
**SSD1331 OLED Display Controller Interface Notes**

**Overview**

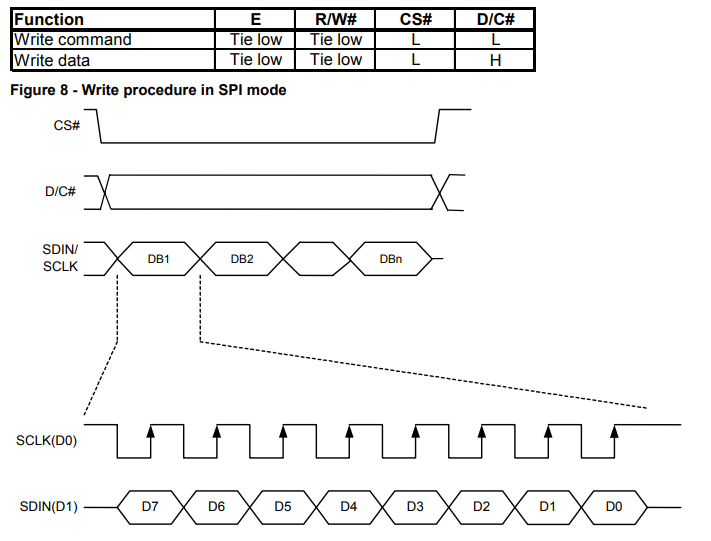
* Pixel Dimensions: 96x64
* 16 bit color resolution
* Uses SPI to communicate from FPGA to PMOD
* Only write operations allowed in Serial mode

**PMOD Pinout**

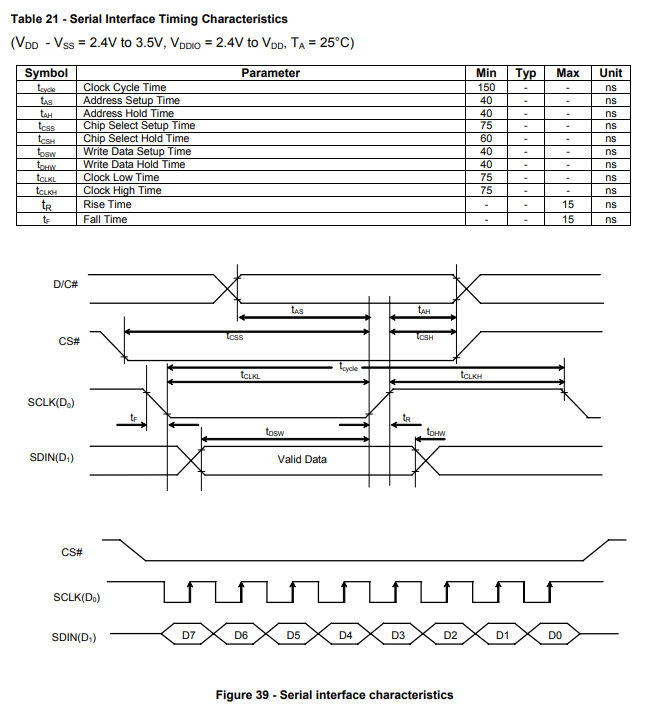
* CS: Chip select, active high reset
* MOSI: Master out slave in
* SCK: serial clock
* D/C: Data/command control
  + If high, data is written to graphic display data RAM
  + If low, inputs at D0-D15 are interpreted as a command and will be decoded and written to corresponding command register
* RES: Power reset
* VCCEN: VCC Enable
* PMODEN: Vdd logic voltage control



**SPI Interface (Section 7.1.3, Page 17)**



**Clock Timing**



Minimum SCLK cycle time = 150ns

* Maximum 6.66 MHz
* With 100MHz clock, can use divider 100 for 1 MHz clock

**Reset Circuit**

When RES pulled low, chip is initialized with the following:

1. Display is OFF
2. 64 MUX display mode
3. Display start line is set at display RAM address 0
4. Display offset set to 0
5. Normal segment and display data column address and row address mapping
6. Column address counter is set at 0
7. Master contrast control register is set at 0FH
8. Individual contrast control registers of color A, B, and C are set to 80H
9. Shift register data clear in serial interface
10. Normal display mode (equivalent to A4 command)

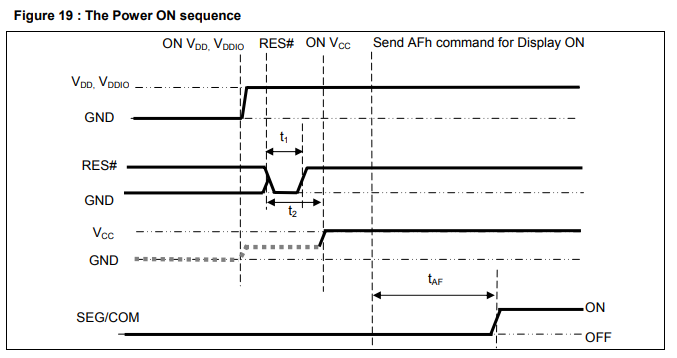
**Power on sequence**

Bytes provided are in format of (command, data)

1. Bring D/C control to logic low
2. Bring the reset pin logic high
3. Bring the VCC Enable logic low
4. Bring PMODEN to high, wait 20 ms
5. Bring RES logic low, wait at least 3 us, then bring back to logic high to reset display controller
6. Wait for reset to complete, max of 3 us
7. Enable driver IC to accept command by sending unlock command over SPI (0xFD, 0x12)
8. Send display off command (0xAE)

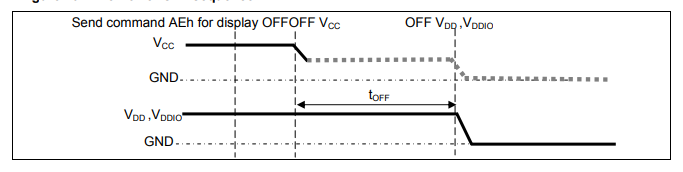
Power on from SSD1331 Datasheet:

1. Power ON VDD, VDDIO
2. After VDD stable, set RES pin LOW for atleast 3us, and then HIGH
3. Wait atleast 3 uS to power on VCC
4. After VCC is stable, send command 0xAF to power display ON, will be on after 100ms



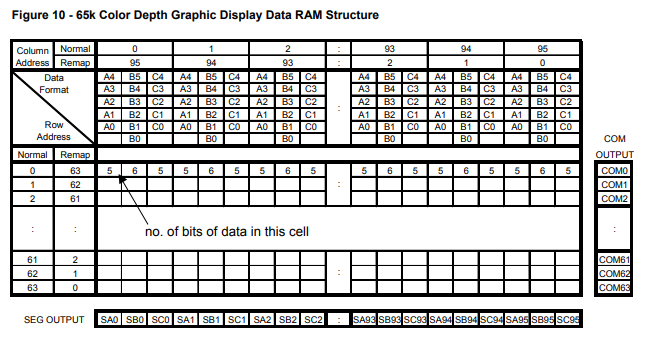
**Power off sequence**

1. Turn the display off (0xAE)
2. Bring VCCEN low
3. Delay 400 ms
4. Power off VDD

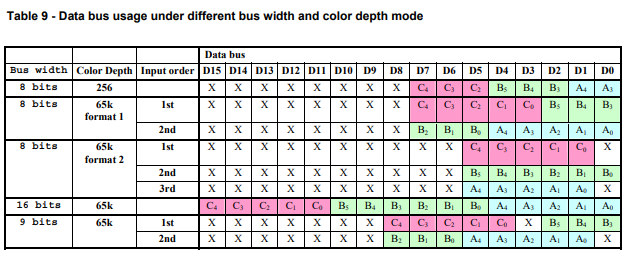


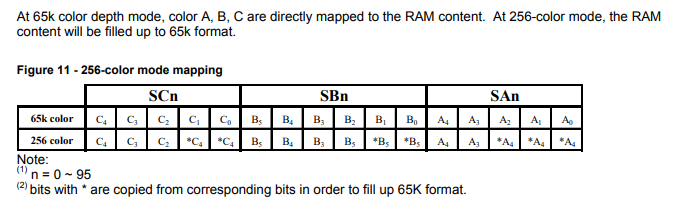
**Graphic Display Data RAM (GDDRAM)**

* Bit mapped static RAM holding pattern to be displayed
* RAM size is 96x64x16 bits
* Each pixel has 16-bit data
* Three sub pixels for color A, B, and C have 5 bits, 6 bits, and 5 bits respectively



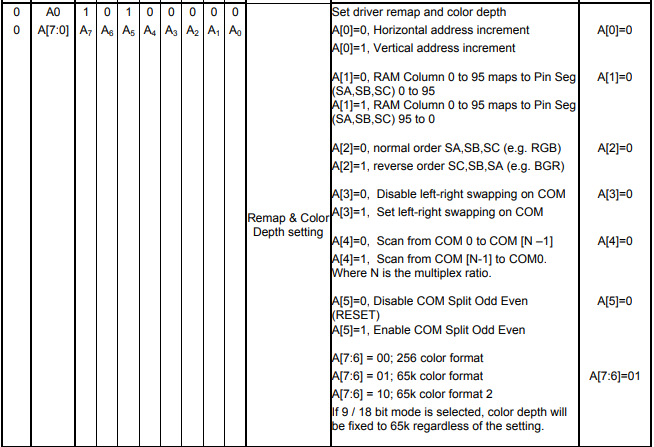
* Can also choose different color depth to save on transmit size



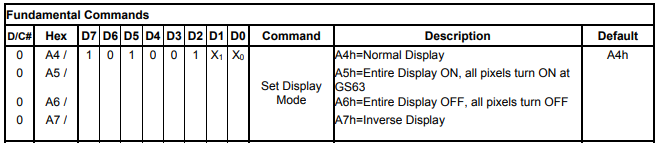


**Remap and Color Depth Setting**

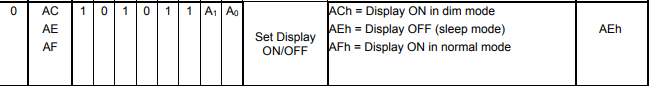




**Set Display Mode**



**Set Display ON/OFF (0xAD)**



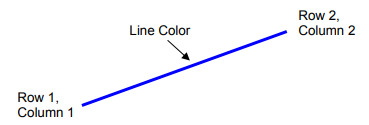
**Display Functions**

* Pages 28 through 33 of datasheet
* Additional functions on page 35

**Drawing a Line**

Create a line between two sets of points with the command 0x21

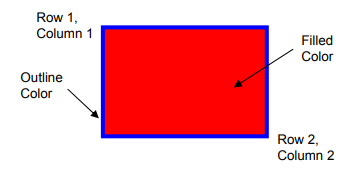
1. Enter into draw line mode with command 0x21
2. Send column start address of line, column1, for example 0x01
3. Send row start address of line, row1, for example 0x10
4. Send column end address of line, column 2, for example 0x28
5. Send row end address of line, row 2, for example 0x04
6. Send color C, B, and A of line, for example 35d, 0d, 0d for blue color



**Drawing a rectangle**

Create a rectangle with a specific command 0x22

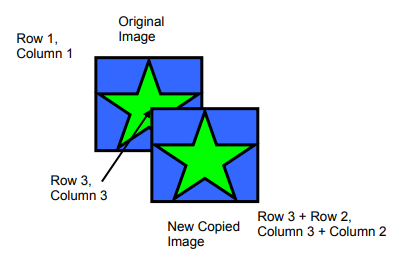
1. Enter “Draw rectangle mode” by execute command 0x22
2. Set starting column coordinates, column 1, for example 0x03
3. Set starting row coordinates, Row 1, for example 0x02
4. Set finishing column coordinates, column 2, for example 0x12
5. Set finishing row coordinates, Row 2, for example 0x15
6. Set outline color C, B, and A, for example 28, 0, and 0 for blue color
7. Set filled color C, B, and A, for example 0, 0, and 40 for red color



**Copy**

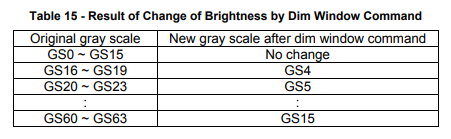
Copy the rectangular region defined by starting point (row 1, col 1) and the ending point (row 2, col 2) to location (row 3, col 3). If new coordinates are smaller than the ending points, the new image will overlap with original one.

1. Enter copy mode with command 0x23
2. Set starting column coordinates, Column 1, for example 0x00
3. Set starting row coordinates, row 1, for example 0x00
4. Set finishing column coordinates, Column 2, for example 0x055
5. Set finishing row coordinates, row 2, for example 0x05
6. Set new column coordinates, column 3, for example 0x03
7. Set new row coordinates, row 3, for example 0x03



**Dim Window**

Command will dim window area specify by starting point (row 1, col 1) and the ending point (row 2, col 2). After execution of this command, the selected window area will become darker as follows:

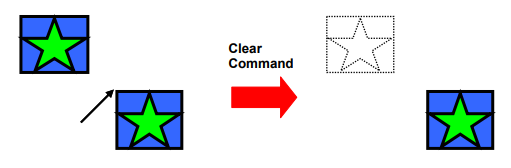


Additional execution of this command over the same window area will not change the data content

**Clear Window**

This command sets the window area specify by starting point (row 1, col 1) and the ending point (row 2, col 2) to clear the window display. The graphic display data RAM content of the specified window will be set to 0

1. Set clear mode with command 0x25
2. Set starting col 1
3. Set starting row 1
4. Set finish col 2
5. Set finish row 2



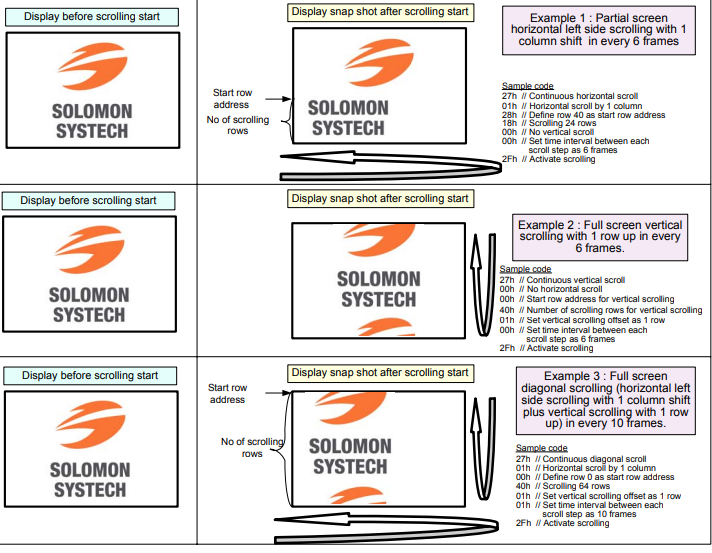
**Fill Enable/Disable**

This command has two functions:

1. Enable/Disable fill (A[0])
   1. 0: disable filling of color into rectangle in draw rectangle command (RESET)
   2. 1: enable filling of color into rectangle in draw rectangle command
2. Enable/Disable reverse copy(A[4])
   1. 0: Disable reverse copy (RESET)
   2. 1: during copy command, the new image colors are swapped

**Scrolling**

Setup command is 0x27



Should not change setup parameters after scrolling has been activated (deactivate first!)

Deactivate scrolling: 0x2E

Activate scrolling: 0x2F