

Application Note
APN_GC_002

Application note

SSD1905/SSD1906 Graphics Controllers
Quick Start Guide

This document contains information on Solomon Systech product's application use case. Actual application use need consulting the final hardware specification of various parts.

<http://www.solomon-systech.com>

APN_GC_002 | Rev 0.10 | P 1/8 | Mar 2007

Copyright © 2007 Solomon Systech Limited



SOLOMON
SYSTECH

CONTENTS

1	INTRODUCTION.....	3
2	HARDWARE SYSTEM CONFIGURATION	3
2.1	SELECTION OF HOST MCU INTERFACE TYPE.....	3
2.2	SELECTION OF LCD INTERFACE	3
3	COMMUNICATE WITH GRAPHIC CONTROLLER.....	4
3.1	TROUBLE-SHOOTING THE FIRST PCB USING SSD190X.....	4
3.2	TROUBLE-SHOOTING HOST MCU INTERFACE.....	4
3.3	LCD INTERFACE TROUBLE-SHOOTING.....	4
3.3.1	<i>Pixel timing</i>	4
3.3.2	<i>Line timing</i>	5
3.3.3	<i>Frame timing</i>	5
3.4	FRAME BUFFER MEMORY	5
4	TURNING ON THE PANEL.....	6
4.1	INITIALIZE THE FRAME BUFFER AND CHECK SSD190X LCD OUTPUT SIGNALS.....	6
4.2	PANEL TIMING FINE TUNING.....	6
5	PLAYING WITH SSD190X FUNCTIONS	6
5.1	OTHER THAN 16BPP MODES	6
5.2	MISCELLANEOUS	6
6	SAMPLE PANEL CONFIGURATION USING TRULY MODULE (TFT-G320240DTSW-50W-E).....	7
6.1	CHECK THE LCD OUTPUT OF HARDWARE.....	7
6.2	TRY FRAME BUFFER UPDATE AFTER PANEL CONNECTED.....	7

1 INTRODUCTION

Solomon Systech Graphic Controllers family consists of SSD1905, SSD1906 and SSD1908. These act as frame buffer devices for graphical image to LCD displays, host microcontroller can access to them as if they are normal SRAM memory. All members of Graphic Controllers family feature STN, color STN, TFT panels driving support.

Using of LCD controller to interface any kind of host MCU to LCD displays often need some basic terminology of LCDs and video. Of course, the MCU external memories interface configurations need to be familiar with first.

2 HARDWARE SYSTEM CONFIGURATION

2.1 Selection of host MCU interface type

- Different MCUs have different standard interface to external devices. SSD190x Graphic Controller chips have different kind of interfaces to choose from. The details can refer to the interface type selection table and its related states table. 8-bit and/or 16-bit accesses are possible when interfacing SSD190x. Usually, for SRAM typed interface, “Generic #2” can be used. Detailed timing information can refer to “CPU Interface Timing” section of SSD190x datasheet.
- There are hardware configuration pins of SSD190x, these are called CF[7:0]. For the interface selection, CF[2:0] are used. If incorrect interface type was selected, SSD190x cannot talk to host MCU, since the timing requirements are totally different.
- “Big Endian” or “Little Endian” should also have to be select here, this is to match your software system’s byte order.
- DTACK# or WAIT# signal is optional for the system. Without using wait signal, just set the MCU host wait-state to match the maximum number of cycles required for any registers and memory accesses. This number is depending on the CLKI to BCLK divide selection (using CF[7:6]). If there is WAIT signal in MCU. It is recommended using it as it will fasten most registers and memory access. CF5 is used to select if the WAIT signal is active high or active low.
- “Host Bus Interface Pin Mapping” section of SSD1906 datasheet can be used as a quick reference.
- M/R# signal input pin of SSD190x is used to select memory frame buffer when in high state, and select SSD190x registers when in low state. It can use another higher order address bus to connect to this M/R#.

2.2 Selection of LCD interface

- Different LCD technology uses different interface type to host systems. SSD190x acts as the display’s frame buffer storing and refreshing the display with its content. Not requiring host to refresh the display every frame to free up host MCU’s power and MCU time processing.
- There are 2 types of panel interfaces which are commonly used, but the driving structures are totally different. The difference is mainly due to LCD driving difference:
 1. STN types of panels
 - Gray levels are generated by frame-rate control (FRC) or pulse-width modulation (PWM). This is a digital method of generating gray levels not so depending of display driver.
 - 4-bit or 8-bit STN interface are used for color or mono panel modules. 1 bit for each pixel of mono panel, or 1 bit for each sub-pixel of color panel.
 - The default memory arrangement is for single-scan type of STN panels. Dual-scan panel will need special handling of memory buffer.
 2. TFT types of panels
 - Gray levels are generated by DAC values or voltage levels of pixel data. Each pixel data is clocked (using LSHIFT) from SSD190x into display driver.

- For color TFT panel modules, parallel RGB or serial RGB are the main streams. While parallel RGB means shifting RGB sub-pixels to panel in a single clock, serial RGB usually means shifting each RGB sub-pixels in 3 separated clocks.

3 COMMUNICATE WITH GRAPHIC CONTROLLER

3.1 Trouble-shooting the first PCB using SSD190x

- Check all power supplies to SSD190x. Only single 3.3V is required for the operation of system. Note that COREVDD is connecting to a capacitor only which is used to stabilized the internal generated 2.5V power for SSD190x internal core.
- Make sure the logic level clock input is available to SSD190x through the CLKI pin. This is required for SSD190x and is the operating speed of SSD190x. If different clock speed is used for LCD pixel / shift clock, AUXCLK input of SSD190x can be used.
- To keep leakage current low, tie unused input ports of SSD1906 to GND or VDD when not used. Not used output ports can keep open.

3.2 Trouble-shooting host MCU interface

- Check CF pins status and feed SSD190x with memory access signals from MCU host. Make sure WAIT# signal is checked by MCU host when connected. If WAIT# not used, the MCU host's access time to SSD190x can be set to longest first to guarantee access operations.
- When correct read or write signal send to SSD190x and can be recognized, the WAIT# signal should have output from SSD190x. A pull-up or pull-down resistor is required by the WAIT# pin depends on active low or active high selection of this pin. Even if WAIT# signal is not used by MCU host, it is advised to have test point with this pin to easily check its output during debugging.
- With the MCU interface signals correct, the first test can be checking on the read-only registers of SSD190x. E.g. registers REG[01h], REG[02h] and some others registers detailed in datasheet can be used in checking.
- Next can check the write operation to some read/write registers and frame buffer memories. After that, check for odd and even addresses operations.
- Up to this stage, the interface between MCU host and SSD190x should be working. Some memory test programs can be used to verify access to SSD190x frame buffer memories. Then the interface timing can be optimized according to CF status and specification access timing minimal requirements.

3.3 LCD interface trouble-shooting

- Power on state of SSD190x was in power saving mode where there is NO LCD driving signal output.
- Try to write data 00 to register REG[A0h], this will set SSD190x into operating mode where LSHIFT, LLINE, LFRAME has output signals when checking with a scope. This can be a good test if the MCU interface does not have read-back capability. Note that this will not be your panel modules' signal as the panel related parameters have not been set yet. This part is only to verify if SSD190x's LCD interface can output correctly.
- After seeing signals output from above pins, we can set "Panel Type Register" of SSD190x according to the panel module to be configured.
- This part can refer to "Display Interface" section of datasheet for detailed information.

3.3.1 Pixel timing

- Graphics controller use pixels as registers units in various timing calculations. Pixel clock was a common reference to all parts of datasheet.
- Note that for STN panel, 1 shift clock is not equivalent 1 pixel clock in SSD190x terminology.
 - In mono mode, 4-bit or 8-bit parallel STN interface are commonly used. In this case, duration of 1 shift clock was driving 4 pixels or 8 pixels respectively.

- In color mode, RGB totally 3 sub-pixels contribute one pixel. For 8-bit parallel STN interface, 3 shift clocks can drive $3 \times 8 = 24$ sub-pixel data. i.e. 3 shift clocks period is equal to $24/3 = 8$ pixel clocks period.
- For other type of panels, 1 shift clock is equal to 1 pixel clock.
- For pixel clock (PCLK) frequency setting, first select the input clock source, either from system input clock (CLKI) or auxiliary clock (AUXCLK). And then select the ratio of PCLK to input clock. Make sure this frequency is close to the target pixel clock frequency which should appear in panel module specification.

3.3.2 Line timing

- For SSD190x, each display line consist of pixels, there is a total number of equivalent pixels per line value in panel specifications usually (named as “Horizontal Total / HT” in SSD190x). This value should be larger then the number of display pixels per line “Horizontal Display Period / HDP”. E.g. a 320RGBx240 panel, the display pixels per line is 320.
- With LSHIFT timing matching the specification, line timing can be adjusted with the “Horizontal Total” registers setting. Set this according to the panel line timing requirement.
- Select LLINE width and polarity with SSD190x register according to panel module specification.
- Note that for STN panel, 1 shift clock is not equivalent 1 pixel clock in SSD190x terminology.
 - In mono mode, 4-bit or 8-bit parallel STN interface are commonly used. In this case, duration of 1 shift clock was driving 4 pixels or 8 pixels respectively.
 - In color mode, RGB totally 3 sub-pixels contribute one pixel. For 8-bit parallel STN interface, 3 shift clocks can drive $3 \times 8 = 24$ sub-pixel data. i.e. 3 shift clocks period is equal to $24/3 = 8$ pixel clocks period.
- For other type of panels, 1 shift clock is equal to 1 pixel clock.

3.3.3 Frame timing

- The frame frequency of panels varies not so big for TFT panels, vary around 60Hz. However, for STN or color STN, the frame frequency can be much higher (as fast as 200Hz). This is because some STN panels use frame rate control to generate the gray levels, slower refresh will cause display to flicker.
- In this part the total number of equivalent lines per frame timing needs to be set correctly. This setting is adjusted using SSD190x’s “Vertical Total / VT”. Similar to line timing, the “Vertical Display Period /VDP” has to be set as well.
- Similar to LLINE setting, select LFRAME width and polarity with SSD190x register according to panel module specification.

3.4 Frame buffer memory

- Graphics controller SSD1905 has 80K bytes (KB) of frame buffer memory whereas SSD1906/08 has 256KB.
- Virtual display is supported as the display buffer length is not necessary the same as storage buffer length per horizontal line.
- E.g. in a 320RGB x 240 panel, each line should have 320 RGB pixels.
 - If displayed in SSD1906 16bpp mode, each pixel occupies 16 bits or 2 bytes.
 - $320 \times 2\text{bytes} = 640$ bytes required per line of display.
 - Assume the first line buffer address offset start from zero. If storage buffer line length is 660 bytes, the second line buffer address will be starting from 660. Then $(660-640)/2 = 10$ pixels of virtual display will not be shown on screen.
 - When the first line buffer address offset change from zero to, say 5, and the image will shift left 5 pixels horizontally.
- Note that except 16bpp mode, all other modes (1bpp, 2bpp, 4bpp or 8bpp modes) need the help of Look-up-table (LUT) initialization to have correct color output. Since LUT values after power-up is not initialized, there is possibility of no display content shown in these modes. So we usually tune panels with 16bpp mode first with “Display Mode Register” of SSD190x configured.

- As we usually use the “Main Window” of SSD190x for display, we need this window correctly configured before the content can be show on display. “Main Window Display Start Address Register” of SSD190x is 0 when power up. So we can put 16bpp image data into frame buffer starting from frame buffer address 0.
- The next important register to set is the “Main Window Line Address Offset Register”. SSD190x use this value to decide where to fetch the next line of image from frame buffer. If this register use power on default value of 0, the display will repeating the first line of frame buffer content for the rest of the lines.
- If possible, try using some memory test software routine to test the frame buffer memory of SSD190x. This is to make sure the MCU timing or wait-state setting is not too critical for SSD190x.

4 TURNING ON THE PANEL

4.1 Initialize the frame buffer and check SSD190x LCD output signals

- Up to this stage, the rough panel timing is fixed. Set display to 16 bit per pixel mode with “Display Mode Register” and fill frame buffer memory with content of 0xffff. This should set screen all white (or black depends on panel normal black or normal white).
- Use a scope to check the LCD output signals of SSD190x. Those pins are grouped into “LCD Interface Pin Descriptions” in SSD190x datasheet. These should be all logic level signals and the frequencies match the panel module specification.

4.2 Panel Timing Fine Tuning

- The next parameters to be set are the “Pulse Start Position Register” and “Pulse Start Offset Register” for both LLINE and LFRAME.
- Panel can be connected for the test afterwards. If the start position or offset is wrong, the display image will be shifted left or right or even scrolling depends on the panel driver used.
- Use display pattern with boundaries to the check if the image correctly on the panel without any lost. Tune the start position and offset if not correct.

5 PLAYING WITH SSD190X FUNCTIONS

5.1 Other than 16bpp Modes

- Set LUT according to SSD190x datasheet. And then change “Display Mode Register”.
- Fill frame buffer with the content of that color depth.

5.2 Miscellaneous

- Hardware rotation, floating window, cursors with or without blinking.
- Color invert, blank screen, dithering.
- Byte swap in fetching display frame buffer content is a very useful feature to match the software system of host MCU.
- The SSD190x datasheet contains more in-depth information on how these features can be used.
- SSD190x can act as a timing generator, the content mapped to display interface pins are content from frame buffer directly. Some panel types even not mentioned in datasheet can be configured. E.g. delta typed panel modules, TFT panel modules with serial RGB interface, etc.

6 SAMPLE PANEL CONFIGURATION USING TRULY MODULE (TFT-G320240DTSW-50W-E)

In this example we assume the MCU to SSD1906 interface is working after procedure described in section 3.2. And a 48MHz oscillator was used at SSD1906's CLKI pin input, and it is used as clock reference to the TFT panel. Other than 48MHz inputs need to change registers 04h and/or 05h accordingly. CF[7:6] both set to logic 0 (low level signal); i.e. BCLK = CLKI = 48MHz.

6.1 Check the LCD output of hardware

- Setup pixel clock to LSHIFT pin of SSD1906, but do not connect to panel yet:
 - REG [04h] to 10h (MCLK = $\frac{1}{2}$ BCLK = 24MHz)
 - REG [05h] to 30h (MCLK used, PCLK = $\frac{1}{4}$ MCLK = 6MHz)
 - REG [10h] to 61h (18bit output TFT color panel selected)
 - REG [A0h] to 00h (enable LCD signals output)
 - Measure LSHIFT with a scope and a 6MHz PCLK output signal will be seen
- Configuring panel parameters:
 - Horizontal:
 - REG [12h] to 32h (Horizontal Total, HT setting; 408 equivalent total pixels)
 - REG [14h] to 27h (HDP setting, 320 horizontal display pixels)
 - REG [16h][17h] to 45h and 00h respectively (HDPS setting)
 - REG [20h] to 1dh (HPW setting, active low line pulse in LLINE)
 - REG [22h][23h] to 05h and 00h respectively (HPS setting)
 - Vertical:
 - REG [18h][19h] to 05h and 01h respectively (Vertical Total, VT setting; 262 equivalent total lines)
 - REG [1ch][1dh] to 0efh and 00h respectively (VDP setting, 240 vertical display lines)
 - REG [1eh][1fh] to 12h and 00h respectively (VDPS setting)
 - REG [24h] to 03h (VPW setting, active low frame pulse in LFRAME)
- The panel output signal should be ready after this step. Please move the REG [A0h] to enable LCD output after setting REG [24h] above. Then the next test can be checked with panel connected to SSD1906 after power down the system.

6.2 Try frame buffer update after panel connected

- Configuring frame buffer memory buffer parameters, assuming using 16 bits-per-pixel (bpp) mode displaying on the main window of SSD1906 without virtual display:
 - REG [70h] set to 04h (16bpp color mode)
 - REG [74h][75h][76h] set to all 0 (assume frame buffer start from address 0 of 256KB SSD1906 frame buffer memory)
 - REG [78h][79h] set to 0a0h and 00h respectively (Main window line address offset = $320/(32/16\text{bpp}) = 160$ or 0a0h)
- Now, the SSD1906 memory buffer start from 0h to $320 \times 240 \times 2 = 153600$ or 25800h in hexadecimal will be the display frame buffer of 320x240 TFT panel.
- Filling upper left corner pixel with white color by setting buffer addresses 0h and 1h to 0ffh. The adjacent pixel on the same horizontal line (the second pixel) can be set in buffer address 2h and 3h and so on. Similarly, the lower right corner pixel was controlled by memory address 257feh and 257ffh.
- RGB bits are in 5-6-5 format, i.e. 5 bits red, 6 bits green and 5 bits blue. The data order can be changed using "word swap" or "byte swap" in register control of REG [71h] bit 7 and bit 6. Adjust it according to your host MCU or MCU firmware endian convention.
- Now you can test the panel by filling whole screen with black, white, red, green, blue and other colors as well.

Solomon Systech Ltd.

Solomon Systech reserves the right to make changes without notice to any products herein. Solomon Systech makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Solomon Systech assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any, and all, liability, including without limitation consequential or incidental damages. "Typical" parameters can and do vary in different applications. All operating parameters, including "Typical" must be validated for each customer application by the customer's technical experts. Solomon Systech does not convey any license under its patent rights nor the rights of others. Solomon Systech products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Solomon Systech product could create a situation where personal injury or death may occur. Should Buyer purchase or use Solomon Systech products for any such unintended or unauthorized application, Buyer shall indemnify and hold Solomon Systech and its offices, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Solomon Systech was negligent regarding the design or manufacture of the part.

<http://www.solomon-systech.com>