

SSD1963

Advance Information

1215KB Embedded Display SRAM LCD Display Controller

This document contains information on a new product. Specifications and information herein are subject to change without notice.

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

SSD1963

Rev 1.5

P 1/94

Apr 2012

Copyright © 2012 **Solomon Systech Limited**



Appendix: IC Revision history of SSD1963 Specification

| Version | Change Items | Effective Date |
|-------------------|---|----------------|
| 0.10 21-Nov-08 | 1 st Release | 24-Nov-08 |
| 0.10 08-Dec-08 | <ol style="list-style-type: none"> 1. Changed the set_pll_mnk to set_pll_mn in section 7.2 2. Change register name in section 8 3. Removed ABC 4. Revised description for REG 0x00, 0x01, 0x0C, 0x0D, 0x0E, 0x10, 0x11, 0x21, 0x26, 0x28, 0x2A, 0x2B, 0x2C, 0x2E, 0x33, 0x34, 0x35, 0x36, 0x37, 0x3A, 0x3C, 0x3E, 0x44, 0x45, 0xA1, 0xB0, 0xB1, 0xB4, 0xB5, 0xB6, 0xB7, 0xB8, 0xB9, 0xBE, 0xBF, 0xD0, 0xD1, 0xD4, 0xE5. 5. Added max VIH in Table 12-1 6. Added Table 9-1 7. Added Table 11-1 8. Revised Figure 9-19 9. Revised Figure 14-2 10. Revised Figure 13-5 11. Corrected typo for Table 7-2 12. Revised test condition for 12 and 13 | 10-Dec-08 |
| 1.0 07-May-09 | <ol style="list-style-type: none"> 1. Changed status to Advance Information 2. Update min/max rating of VDDD and VDDPLL in Table 11-1 3. Added tape and reel drawing of 128-pin LQFP package in Section 15.3 4. Revised Section 13.2 5. 5. Added 12 bits for Table 7-1 6. Removed TTL interface 7. Revised section 7.1.5 8. Change the title of section 7.2 9. Revised command description in section 8 10. Removed the command 0x0C and 0x3A 11. Added figures in section 13.4 12. Revised figures in section 13.3 13. Revise Table 6-1 | 18-May-09 |
| 1.1 23-Dec-09 | <ol style="list-style-type: none"> 1. Update Table 7-1 2. Revised section 9.72 | 18-Jan-10 |
| 1.2 31-May-10 | <ol style="list-style-type: none"> 1. Add Table 13-7 | 15-Jul-10 |
| 1.3 11-Nov-10 | <ol style="list-style-type: none"> 1. Update Section 7.2 reset timing 2. Correct Section 13.4 the serial RGB timing <p>Correct Table 6.1-6.5 Pin Mapping -> Pin description</p> | 07-Dec-10 |
| 1.4 25-Jul-11 | <ol style="list-style-type: none"> 1. Update Section 9.45 SET_PWM_CONF 2. Update Section 9.30 SET_TEAR_SCANLINE 3. Update Section 15.3 Tape and Reel Drawing | 25-Jul-11 |
| 1.5 16-Apr-12 | <ol style="list-style-type: none"> 1. Update Table 13-5 and 13-6 (t_{PWCSH} and t_{PWCSL} typo) | 25-Apr-12 |

CONTENTS

| | | |
|----------|---|-----------|
| 1 | GENERAL DESCRIPTION | 8 |
| 2 | FEATURES..... | 8 |
| 3 | ORDERING INFORMATION | 8 |
| 4 | BLOCK DIAGRAM | 9 |
| 5 | PIN ARRANGEMENT..... | 10 |
| 5.1 | 80 BALLS TFBGA..... | 10 |
| 5.2 | 128 PINS LQFP | 11 |
| 6 | PIN DESCRIPTIONS..... | 13 |
| 7 | FUNCTIONAL BLOCK DESCRIPTIONS..... | 16 |
| 7.1 | MCU INTERFACE..... | 16 |
| 7.1.1 | 6800 Mode | 16 |
| 7.1.2 | 8080 Mode | 16 |
| 7.1.3 | Register Pin Mapping | 16 |
| 7.1.4 | Pixel Data Format | 16 |
| 7.1.5 | Tearing Effect Signal (TE)..... | 17 |
| 7.2 | SYSTEM CLOCK GENERATION | 18 |
| 7.3 | FRAME BUFFER..... | 19 |
| 7.4 | SYSTEM CLOCK AND RESET MANAGER | 19 |
| 7.5 | LCD CONTROLLER..... | 20 |
| 7.5.1 | Display Format | 20 |
| 7.5.2 | General Purpose Input/Output (GPIO) | 20 |
| 8 | COMMAND TABLE | 21 |
| 9 | COMMAND DESCRIPTIONS..... | 24 |
| 9.1 | NOP..... | 24 |
| 9.2 | SOFT_RESET..... | 24 |
| 9.3 | GET_POWER_MODE..... | 24 |
| 9.4 | GET_ADDRESS_MODE | 25 |
| 9.5 | GET_DISPLAY_MODE | 25 |
| 9.6 | GET_TEAR_EFFECT_STATUS | 26 |
| 9.7 | ENTER_SLEEP_MODE | 27 |
| 9.8 | EXIT_SLEEP_MODE | 27 |
| 9.9 | ENTER_PARTIAL_MODE | 27 |
| 9.10 | ENTER_NORMAL_MODE..... | 27 |
| 9.11 | EXIT_INVERT_MODE..... | 28 |
| 9.12 | ENTER_INVERT_MODE | 28 |
| 9.13 | SET_GAMMA_CURVE | 29 |
| 9.14 | SET_DISPLAY_OFF | 29 |
| 9.15 | SET_DISPLAY_ON..... | 29 |
| 9.16 | SET_COLUMN_ADDRESS | 30 |
| 9.17 | SET_PAGE_ADDRESS..... | 30 |
| 9.18 | WRITE_MEMORY_START..... | 31 |
| 9.19 | READ_MEMORY_START | 32 |
| 9.20 | SET_PARTIAL_AREA..... | 32 |
| 9.21 | SET_SCROLL_AREA..... | 34 |
| 9.22 | SET_TEAR_OFF | 36 |
| 9.23 | SET_TEAR_ON..... | 36 |
| 9.24 | SET_ADDRESS_MODE..... | 36 |

| | | |
|-----------|---|-----------|
| 9.25 | SET_SCROLL_START | 39 |
| 9.26 | EXIT_IDLE_MODE | 40 |
| 9.27 | ENTER_IDLE_MODE | 40 |
| 9.28 | WRITE_MEMORY_CONTINUE..... | 41 |
| 9.29 | READ_MEMORY_CONTINUE | 42 |
| 9.30 | SET_TEAR_SCANLINE..... | 43 |
| 9.31 | GET_SCANLINE | 43 |
| 9.32 | READ_DDB | 44 |
| 9.33 | SET_LCD_MODE..... | 44 |
| 9.34 | GET_LCD_MODE | 46 |
| 9.35 | SET_HORI_PERIOD | 47 |
| 9.36 | GET_HORI_PERIOD..... | 48 |
| 9.37 | SET_VERT_PERIOD..... | 49 |
| 9.38 | GET_VERT_PERIOD | 49 |
| 9.39 | SET_GPIO_CONF..... | 50 |
| 9.40 | GET_GPIO_CONF | 51 |
| 9.41 | SET_GPIO_VALUE | 52 |
| 9.42 | GET_GPIO_STATUS..... | 52 |
| 9.43 | SET_POST_PROC..... | 53 |
| 9.44 | GET_POST_PROC | 53 |
| 9.45 | SET_PWM_CONF..... | 54 |
| 9.46 | GET_PWM_CONF | 55 |
| 9.47 | SET_LCD_GEN0..... | 57 |
| 9.48 | GET_LCD_GEN0 | 58 |
| 9.49 | SET_LCD_GEN1 | 59 |
| 9.50 | GET_LCD_GEN1 | 60 |
| 9.51 | SET_LCD_GEN2..... | 61 |
| 9.52 | GET_LCD_GEN2 | 62 |
| 9.53 | SET_LCD_GEN3..... | 63 |
| 9.54 | GET_LCD_GEN3 | 64 |
| 9.55 | SET_GPIO0_ROP..... | 65 |
| 9.56 | GET_GPIO0_ROP..... | 65 |
| 9.57 | SET_GPIO1_ROP..... | 66 |
| 9.58 | GET_GPIO1_ROP..... | 67 |
| 9.59 | SET_GPIO2_ROP..... | 67 |
| 9.60 | GET_GPIO2_ROP..... | 68 |
| 9.61 | SET_GPIO3_ROP..... | 69 |
| 9.62 | GET_GPIO3_ROP..... | 69 |
| 9.63 | SET_DBC_CONF..... | 70 |
| 9.64 | GET_DBC_CONF | 71 |
| 9.65 | SET_DBC_TH..... | 72 |
| 9.66 | GET_DBC_TH | 73 |
| 9.67 | SET_PLL | 73 |
| 9.68 | SET_PLL_MN..... | 74 |
| 9.69 | GET_PLL_MN | 75 |
| 9.70 | GET_PLL_STATUS | 75 |
| 9.71 | SET_DEEP_SLEEP | 75 |
| 9.72 | SET_LSHIFT_FREQ..... | 76 |
| 9.73 | GET_LSHIFT_FREQ | 76 |
| 9.74 | SET_PIXEL_DATA_INTERFACE | 78 |
| 9.75 | GET_PIXEL_DATA_INTERFACE..... | 78 |
| 10 | MAXIMUM RATINGS..... | 79 |
| 11 | RECOMMENDED OPERATING CONDITIONS | 79 |
| 11.1 | POWER-UP SEQUENCE | 79 |
| 12 | DC CHARACTERISTICS..... | 80 |
| 13 | AC CHARACTERISTICS..... | 80 |

| | | |
|-----------|---|-----------|
| 13.1 | CLOCK TIMING | 80 |
| 13.2 | MCU INTERFACE TIMING | 81 |
| 13.2.1 | <i>Parallel 6800-series Interface Timing</i> | 81 |
| 13.2.2 | <i>Parallel 8080-series Interface Timing</i> | 83 |
| 13.3 | PARALLEL LCD INTERFACE TIMING..... | 85 |
| 13.4 | SERIAL RGB INTERFACE TIMING | 86 |
| 14 | APPLICATION EXAMPLE..... | 89 |
| 15 | PACKAGE INFORMATION..... | 91 |
| 15.1 | PACKAGE MECHANICAL DRAWING FOR 80 BALLS TFBGA | 91 |
| 15.2 | PACKAGE MECHANICAL DRAWING FOR 128 PINS LQFP..... | 92 |
| 15.3 | TAPE & REEL DRAWING FOR 128 PINS LQFP | 93 |

TABLES

| | |
|---|----|
| TABLE 3-1: ORDERING INFORMATION | 8 |
| TABLE 5-1: TFBGA PIN ASSIGNMENT TABLE..... | 10 |
| TABLE 5-2 : LQFP PIN ASSIGNMENT TABLE | 12 |
| TABLE 6-1: MCU INTERFACE PIN DESCRIPTION | 13 |
| TABLE 6-2: LCD INTERFACE PIN DESCRIPTION..... | 14 |
| TABLE 6-3: CONTROL SIGNAL PIN DESCRIPTION..... | 14 |
| TABLE 6-4: POWER PIN DESCRIPTION | 14 |
| TABLE 6-5 : LCD INTERFACE PIN MAPPING | 15 |
| TABLE 7-1: PIXEL DATA FORMAT | 17 |
| TABLE 7-2: FRAME BUFFER SETTINGS REGARDING TO SET_ADDRESS_MODE COMMAND 0x36..... | 19 |
| TABLE 9-1 ENTER IDLE MODE MEMORY CONTENT VS DISPLAY COLOR | 41 |
| TABLE 10-1: MAXIMUM RATINGS (VOLTAGE REFERENCED TO V _{SS})..... | 79 |
| TABLE 11-1: RECOMMENDED OPERATING CONDITION..... | 79 |
| TABLE 12-1 : DC CHARACTERISTICS | 80 |
| TABLE 13-1: CLOCK INPUT REQUIREMENTS FOR CLK (PLL-BYPASS) | 80 |
| TABLE 13-2 : CLOCK INPUT REQUIREMENTS FOR CLK | 80 |
| TABLE 13-3 : CLOCK INPUT REQUIREMENTS FOR CRYSTAL OSCILLATOR XTAL..... | 80 |
| TABLE 13-4: PARALLEL 6800-SERIES INTERFACE TIMING CHARACTERISTICS (USE CS# AS CLOCK)..... | 81 |
| TABLE 13-5: PARALLEL 6800-SERIES INTERFACE TIMING CHARACTERISTICS (USE E AS CLOCK) | 82 |
| TABLE 13-6: PARALLEL 8080-SERIES INTERFACE TIMING CHARACTERISTICS..... | 83 |
| TABLE 13-7: QUICK REFERENCE TABLE FOR LCD PARAMETER SETTING | 88 |

FIGURES

| | |
|---|----|
| FIGURE 4-1: SSD1963 BLOCK DIAGRAM | 9 |
| FIGURE 5-1: PINOUT DIAGRAM –TFBGA (TOPVIEW)..... | 10 |
| FIGURE 5-2 : PINOUT DIAGRAM – LQFP (TOPVIEW) | 11 |
| FIGURE 7-1: RELATIONSHIP BETWEEN TEARING EFFECT SIGNAL AND MCU MEMORY WRITING..... | 18 |
| FIGURE 7-2: CLOCK CONTROL DIAGRAM | 19 |
| FIGURE 7-3: STATE DIAGRAM OF SSD1963 | 20 |
| FIGURE 9-1: EXIT INVERT MODE EXAMPLE..... | 28 |
| FIGURE 9-2: ENTER INVERT MODE EXAMPLE..... | 29 |
| FIGURE 9-3: SET COLUMN ADDRESS EXAMPLE | 30 |
| FIGURE 9-4: SET PAGE ADDRESS EXAMPLE | 31 |
| FIGURE 9-5: SET PARTIAL AREA WITH SET_ADDRESS_MODE (0x36) A[4] = 0 WHEN END ROW > START ROW | 33 |
| FIGURE 9-6: SET PARTIAL AREA WITH SET_ADDRESS_MODE (0x36) A[4] = 1 WHEN END ROW > START ROW..... | 33 |
| FIGURE 9-7: SET PARTIAL AREA WITH SET_ADDRESS_MODE (0x36) A[4] = 0 WHEN START ROW > END ROW..... | 33 |
| FIGURE 9-8: SET PARTIAL AREA WITH SET_ADDRESS_MODE (0x36) A[4] = 1 WHEN START ROW > END ROW..... | 33 |
| FIGURE 9-9: SET SCROLL AREA WITH SET_ADDRESS_MODE (0x36) A[4] = 0..... | 35 |
| FIGURE 9-10: SET SCROLL AREA WITH SET_ADDRESS_MODE (0x36) A[4] = 1..... | 35 |
| FIGURE 9-11: A[7] PAGE ADDRESS ORDER | 37 |
| FIGURE 9-12: A[6] COLUMN ADDRESS ORDER..... | 37 |
| FIGURE 9-13: A[5] PAGE / COLUMN ADDRESS ORDER | 37 |
| FIGURE 9-14: A[3] RGB ORDER..... | 38 |
| FIGURE 9-15: A[1] FLIP HORIZONTAL | 38 |
| FIGURE 9-16: A[0] FLIP VERTICAL | 39 |
| FIGURE 9-17: SET SCROLL START WITH SET_ADDRESS_MODE (0x36) A[4] = 0 | 39 |
| FIGURE 9-18: SET SCROLL START WITH SET_ADDRESS_MODE (0x36) A[4] = 1 | 40 |
| FIGURE 9-19: PWM SIGNAL | 54 |
| FIGURE 11-1: POWER-UP SEQUENCE | 79 |
| FIGURE 13-1: PARALLEL 6800-SERIES INTERFACE TIMING DIAGRAM (USE CS# AS CLOCK)..... | 81 |
| FIGURE 13-2: PARALLEL 6800-SERIES INTERFACE TIMING DIAGRAM (USE E AS CLOCK) | 82 |
| FIGURE 13-3: PARALLEL 8080-SERIES INTERFACE TIMING DIAGRAM (WRITE CYCLE) | 83 |
| FIGURE 13-4: PARALLEL 8080-SERIES INTERFACE TIMING DIAGRAM (READ CYCLE)..... | 84 |
| FIGURE 13-5: GENERIC TFT PANEL TIMING | 85 |
| FIGURE 13-6: SERIAL RGB INTERFACE TIMING (WITHOUT DUMMY MODE)..... | 86 |
| FIGURE 13-7: SERIAL RGB INTERFACE TIMING (WITH DUMMY MODE) | 87 |
| FIGURE 14-1 : APPLICATION CIRCUIT FOR SSD1963 (WITH DIRECT CLOCK INPUT)..... | 89 |
| FIGURE 14-2 : APPLICATION CIRCUIT FOR SSD1963 (WITH CRYSTAL OSCILLATOR INPUT) | 90 |

1 GENERAL DESCRIPTION

SSD1963 is a display controller of 1215K byte frame buffer to support up to 864 x 480 x 24bit graphics content. It also equips parallel MCU interfaces in different bus width to receive graphics data and command from MCU. Its display interface supports common RAM-less LCD driver of color depth up to 24 bit-per-pixel.

2 FEATURES

- Display feature
 - Built-in 1215K bytes frame buffer. Support up to 864 x 480 at 24bpp display
 - Support TFT 18/24-bit generic RGB interface panel
 - Support 8-bit serial RGB interface
 - Hardware rotation of 0, 90, 180, 270 degree
 - Hardware display mirroring
 - Hardware windowing
 - Programmable brightness, contrast and saturation control
 - Dynamic Backlight Control (DBC) via PWM signal
- MCU connectivity
 - 8/9/16/18/24-bit MCU interface
 - Tearing effect signal
- I/O Connectivity
 - 4 GPIO pins
- Built-in clock generator
- Deep sleep mode for power saving
- Core supply power (V_{DDPLL} and V_{DDD}): 1.2V \pm 0.1V
- I/O supply power(V_{DDIO}): 1.65V to 3.6V
- LCD interface supply power (V_{DDLCD}): 1.65V to 3.6V

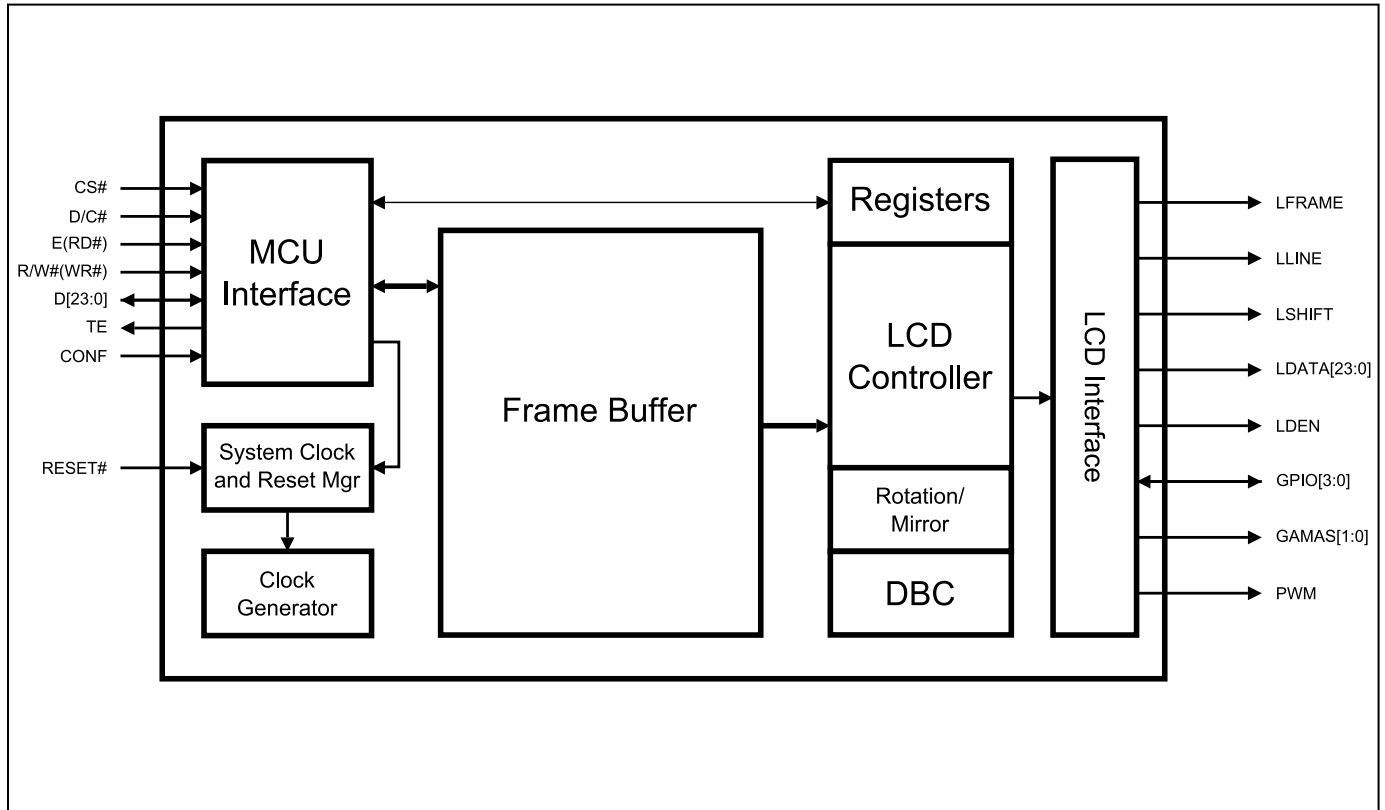
3 ORDERING INFORMATION

Table 3-1: Ordering Information

| Ordering Part Number | Package Form |
|----------------------|------------------------|
| SSD1963G41 | TFBGA-80 (Tray) |
| SSD1963QL9 | LQFP-128 (Tray) |
| SSD1963QL9R | LQFP-128 (Tape & Reel) |

4 BLOCK DIAGRAM

Figure 4-1: SSD1963 Block Diagram



5 PIN ARRANGEMENT

5.1 80 balls TFBGA

Figure 5-1: Pinout Diagram –TFBGA (Topview)

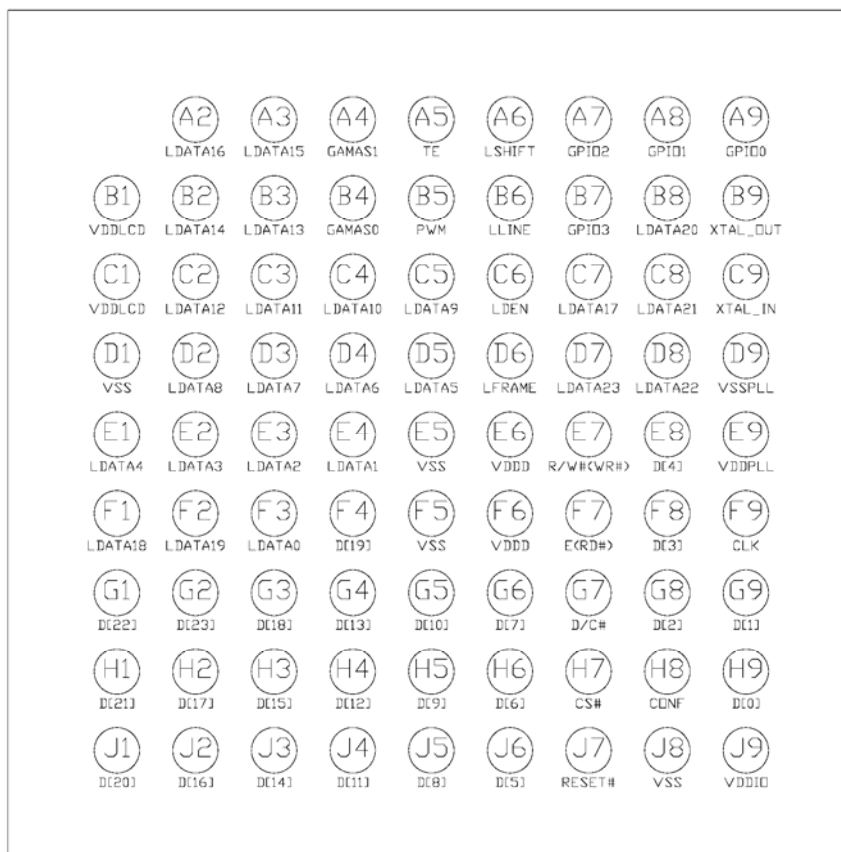


Table 5-1: TFBGA Pin Assignment Table

| Pin # | Signal Name | Pin # | Signal Name | Pin # | Signal Name | Pin # | Signal Name | Pin # | Signal Name |
|-------|-------------|-------|-------------|-------|-------------|-------|-------------|-------|-------------|
| A1 | - | C1 | VDDLCD | E1 | LDATA4 | G1 | D[22] | J1 | D[20] |
| A2 | LDATA16 | C2 | LDATA12 | E2 | LDATA3 | G2 | D[23] | J2 | D[16] |
| A3 | LDATA15 | C3 | LDATA11 | E3 | LDATA2 | G3 | D[18] | J3 | D[14] |
| A4 | GAMAS1 | C4 | LDATA10 | E4 | LDATA1 | G4 | D[13] | J4 | D[11] |
| A5 | TE | C5 | LDATA9 | E5 | VSS | G5 | D[10] | J5 | D[8] |
| A6 | LSHIFT | C6 | LDEN | E6 | VDDD | G6 | D[7] | J6 | D[5] |
| A7 | GPIO2 | C7 | LDATA17 | E7 | R/W# (WR#) | G7 | D/C# | J7 | RESET# |
| A8 | GPIO1 | C8 | LDATA21 | E8 | D[4] | G8 | D[2] | J8 | VSS |
| A9 | GPIO0 | C9 | XTAL_IN | E9 | VDDPLL | G9 | D[1] | J9 | VDDIO |
| B1 | VDDLCD | D1 | VSS | F1 | LDATA18 | H1 | D[21] | | |
| B2 | LDATA14 | D2 | LDATA8 | F2 | LDATA19 | H2 | D[17] | | |
| B3 | LDATA13 | D3 | LDATA7 | F3 | LDATA0 | H3 | D[15] | | |
| B4 | GAMAS0 | D4 | LDATA6 | F4 | D[19] | H4 | D[12] | | |
| B5 | PWM | D5 | LDATA5 | F5 | VSS | H5 | D[9] | | |
| B6 | LLINE | D6 | LFRAME | F6 | VDDD | H6 | D[6] | | |
| B7 | GPIO3 | D7 | LDATA23 | F7 | E(RD#) | H7 | CS# | | |
| B8 | LDATA20 | D8 | LDATA22 | F8 | D[3] | H8 | CONF | | |
| B9 | XTAL_OUT | D9 | VSSPLL | F9 | CLK | H9 | D[0] | | |

5.2 128 pins LQFP

Figure 5-2 : Pinout Diagram – LQFP (Topview)



Table 5-2 : LQFP Pin Assignment Table

| Pin # | Signal Name | Pin # | Signal Name | Pin # | Signal Name | Pin # | Signal Name |
|-------|-------------|-------|-------------|-------|-------------|-------|-------------|
| 1 | VDDD | 33 | VSS | 65 | VSS | 97 | VDDIO |
| 2 | VSS | 34 | VDDLCD | 66 | VDDLCD | 98 | D17 |
| 3 | VSS | 35 | LDATA17 | 67 | LDATA11 | 99 | D16 |
| 4 | VDDIO | 36 | GPIO0 | 68 | LDATA10 | 100 | VDDIO |
| 5 | VSS | 37 | GPIO1 | 69 | LDATA9 | 101 | VSS |
| 6 | VDDD | 38 | GPIO2 | 70 | LDATA8 | 102 | VDDD |
| 7 | D0 | 39 | GPIO3 | 71 | LDATA7 | 103 | D15 |
| 8 | D1 | 40 | VDDD | 72 | LDATA6 | 104 | D14 |
| 9 | D2 | 41 | VSS | 73 | VDDD | 105 | D13 |
| 10 | D3 | 42 | VDDLCD | 74 | VSS | 106 | D12 |
| 11 | D4 | 43 | LFRAME | 75 | VDDLCD | 107 | D11 |
| 12 | VDDIO | 44 | LLINE | 76 | LDATA5 | 108 | VDDIO |
| 13 | VSS | 45 | LSHIFT | 77 | LDATA4 | 109 | VSS |
| 14 | VDDD | 46 | VDDD | 78 | LDATA3 | 110 | VDDD |
| 15 | CLK | 47 | VSS | 79 | LDATA2 | 111 | D10 |
| 16 | VDDIO | 48 | VDDLCD | 80 | LDATA1 | 112 | D9 |
| 17 | VSS | 49 | LDEN | 81 | LDATA0 | 113 | D8 |
| 18 | VDDPLL | 50 | TE | 82 | VDDD | 114 | D7 |
| 19 | VSSPLL | 51 | PWM | 83 | VSS | 115 | D6 |
| 20 | VSS | 52 | GAMAS0 | 84 | VDDLCD | 116 | D5 |
| 21 | VDDD | 53 | GAMAS1 | 85 | LDATA19 | 117 | VDDIO |
| 22 | XTAL_IN | 54 | VDDLCD | 86 | LDATA18 | 118 | VSS |
| 23 | VSS | 55 | VSS | 87 | D19 | 119 | VDDD |
| 24 | XTAL_OUT | 56 | VDDD | 88 | D18 | 120 | R/W#(WR#) |
| 25 | VDDD | 57 | VSS | 89 | D23 | 121 | E(RD#) |
| 26 | VSS | 58 | VDDLCD | 90 | D22 | 122 | D/C# |
| 27 | VDDLCD | 59 | LDATA16 | 91 | D21 | 123 | CS# |
| 28 | LDATA23 | 60 | LDATA15 | 92 | D20 | 124 | VDDIO |
| 29 | LDATA22 | 61 | LDATA14 | 93 | VDDD | 125 | VSS |
| 30 | LDATA21 | 62 | LDATA13 | 94 | VSS | 126 | VDDD |
| 31 | LDATA20 | 63 | LDATA12 | 95 | VDDIO | 127 | RESET# |
| 32 | VDDD | 64 | VDDD | 96 | VSS | 128 | CONF |

6 PIN DESCRIPTIONS

Key:

I = Input

O =Output

IO = Bi-directional (input/output)

P = Power pin

Hi-Z = High impedance

Table 6-1: MCU Interface Pin Description

| Pin Name | Type | Reference Voltage Level | TFBGA Pin # | LQFP Pin # | Description |
|-----------|------|-------------------------|--|--|--|
| CLK | I | VDDIO | F9 | 15 | TTL clock input. This pin should be tied to VSS if TTL clock input is not used |
| XTAL_IN | I | - | C9 | 22 | Crystal oscillator input. This pin should be tied to VSS if not used |
| XTAL_OUT | O | - | B9 | 24 | Crystal oscillator output. This pin should be floating if not used |
| CS# | I | VDDIO | H7 | 123 | Chip select |
| D/C# | I | VDDIO | G7 | 122 | Data/Command select |
| E(RD#) | I | VDDIO | F7 | 121 | 6800 mode: E (enable signal) 8080 mode: RD# (read strobe signal) |
| R/W#(WR#) | I | VDDIO | E7 | 120 | 6800 mode: R/W# 0: Write cycle 1: Read cycle 8080 mode: WR# (write strobe signal) |
| D[23:0] | IO | VDDIO | E8, F4, F8, G1, G2, G3, G4, G5, G6, G8, G9, H1, H2, H3, H4, H5, H6, H9, J1, J2, J3, J4, J5, J6 | 7, 8, 9, 10, 11, 87, 88, 89, 90, 91, 92, 98, 99, 103, 104, 105, 106, 107, 111, 112, 113, 114, 115, 116 | Data bus. Pins not used should be floating |
| TE | O | VDDLCD | A5 | 50 | Tearing effect |

Table 6-2: LCD Interface Pin Description

| Pin Name | Type | Reference Voltage Level | TFBGA Pin # | LQFP Pin # | Description |
|-------------|------|-------------------------|--|--|--|
| LFRAME | O | VDDLCD | D6 | 43 | Vertical sync (Frame pulse) |
| LLINE | O | VDDLCD | B6 | 44 | Horizontal sync (Line pulse) |
| LSHIFT | O | VDDLCD | A6 | 45 | Pixel clock (Pixel shift signal) |
| LDEN | O | VDDLCD | C6 | 49 | Data valid |
| LDATA[23:0] | O | VDDLCD | A2, A3, B2, B3, B8, C2, C3, C4, C5, C7, C8, D2, D3, D4, D5, D7, D8, E1, E2, E3, E4, F1, F2, F3 | 28, 29, 30, 31, 35, 59, 60, 61, 62, 63, 67, 68, 69, 70, 71, 72, 76, 77, 78, 79, 80, 81, 85, 86 | RGB data |
| GPIO[3:0] | IO | VDDLCD | A7, A8, A9, B7 | 36, 37, 38, 39 | These pins can be configured for display miscellaneous signals or as general purpose I/O. Default as input |
| GAMAS [1:0] | O | VDDLCD | A4, B4 | 52, 53 | Gamma selection for panel |
| PWM | O | VDDLCD | B5 | 51 | PWM output for backlight driver |

Table 6-3: Control Signal Pin Description

| Pin Name | Type | Reference Voltage Level | TFBGA Pin # | LQFP Pin # | Description |
|----------|------|-------------------------|-------------|------------|---|
| RESET# | I | VDDIO | J7 | 127 | Master synchronize reset |
| CONF | I | VDDIO | H8 | 128 | MCU interface configuration 0: 6800 Interface 1: 8080 Interface |

Table 6-4: Power Pin Description

| Pin Name | Type | TFBGA Pin # | LQFP Pin # | Description |
|----------|------|----------------|---|--|
| VDDD | P | E6, F6 | 1, 6, 14, 21, 25, 32, 40, 46, 56, 64, 73, 82, 93, 102, 110, 119, 126 | Power supply for internal digital circuit |
| VDDLCD | P | B1, C1 | 27, 34, 42, 48, 54, 58, 66, 75, 84 | Power supply for LCD interface related pads |
| VDDPLL | P | E9 | 18 | Power supply for internal analog circuit and analog I/O pads |
| VDDIO | P | J9 | 4, 12, 16, 95, 97, 100, 108, 117, 124 | Power supply for digital I/O pads |
| VSS | P | D1, E5, F5, J8 | 2, 3, 5, 13, 17, 20, 23, 26, 33, 41, 47, 55, 57, 65, 74, 83, 94, 96, 101, 109, 118, 125 | Ground for internal digital circuit |
| VSSPLL | P | D9 | 19 | Ground for internal analog circuit and analog I/O pads |

Table 6-5 : LCD Interface Pin Mapping

| Pin Names | 24-bit | 18-bit | 8-bit serial |
|-----------|--------|---------|--------------|
| LFRAME | FRAME | | |
| LLINE | LINE | | |
| LSHIFT | SHIFT | | |
| LDEN | DEN | | |
| LDATA23 | R7 | Drive 0 | Drive 0 |
| LDATA22 | R6 | Drive 0 | Drive 0 |
| LDATA21 | R5 | Drive 0 | Drive 0 |
| LDATA20 | R4 | Drive 0 | Drive 0 |
| LDATA19 | R3 | Drive 0 | Drive 0 |
| LDATA18 | R2 | Drive 0 | Drive 0 |
| LDATA17 | R1 | R5 | Drive 0 |
| LDATA16 | R0 | R4 | Drive 0 |
| LDATA15 | G7 | R3 | Drive 0 |
| LDATA14 | G6 | R2 | Drive 0 |
| LDATA13 | G5 | R1 | Drive 0 |
| LDATA12 | G4 | R0 | Drive 0 |
| LDATA11 | G3 | G5 | Drive 0 |
| LDATA10 | G2 | G4 | Drive 0 |
| LDATA9 | G1 | G3 | Drive 0 |
| LDATA8 | G0 | G2 | Drive 0 |
| LDATA7 | B7 | G1 | D7 |
| LDATA6 | B6 | G0 | D6 |
| LDATA5 | B5 | B5 | D5 |
| LDATA4 | B4 | B4 | D4 |
| LDATA3 | B3 | B3 | D3 |
| LDATA2 | B2 | B2 | D2 |
| LDATA1 | B1 | B1 | D1 |
| LDATA0 | B0 | B0 | D0 |

Note

- (1) These pin mappings use signal names commonly used for each panel type, however signal names may differ between panel manufacturers.

7 FUNCTIONAL BLOCK DESCRIPTIONS

7.1 MCU Interface

The MCU interface connects the MCU and SSD1963 graphics controller. The MCU interface can be configured as 6800 mode and 8080 mode by the CONF pin. By pulling the CONF pin to VSSIO, the MCU interface will be configured as 6800 mode interface. If the CONF pin is connected to VDDIO, the MCU interface will be configured in 8080 mode.

7.1.1 6800 Mode

The 6800 mode MCU interface consist of CS#, D/C#, E, R/W#, D[23:0], and TE signals (Please refer to Table 6-1 for pin multiplexed with 8080 mode). This interface supports both fixed E and clock E scheme to define a read/write cycle. If the E signal is kept high and used as enable signal, the CS# signal acts as a bus clock, the data or command will be latched into the system at the rising edge of CS#. If the user wants to use the E pin as the clock pin, the CS# pin then need to be fixed to logic 0 to select the chip. Then the falling edge of the E signal will latch the data or command. For details, please refer to the timing diagram in chapter 13.2.1.

7.1.2 8080 Mode

The 8080 mode MCU interface consist of CS#, D/C#, RD#, WR#, D[23:0] and TE signals (Please refer to Table 6-1 for pin multiplexed with 6800 mode). This interface use WR# to define a write cycle and RD# for read cycle. If the WR# goes low when the CS# signal is low, the data or command will be latched into the system at the rising edge of WR#. Similarly, the read cycle will start when RD# goes low and end at the rising edge of RD#. The detailed timing will show in the chapter 13.2.2.

7.1.3 Register Pin Mapping

When user access the registers via the parallel MCU interface, only D[7:0] will be used regardless the width of the pixel data is. Therefore, D[23:8] will only be used to address the display data only. This provided the possibility that the pixel data format as shown in Table 7-1 can be configured by command 0xF0.

7.1.4 Pixel Data Format

Both 6800 and 8080 support 8-bit, 9-bit, 16-bit, 18-bit and 24-bit data bus. Depending on the width of the data bus, the display data are packed into the data bus in different ways.

Table 7-1: Pixel Data Format

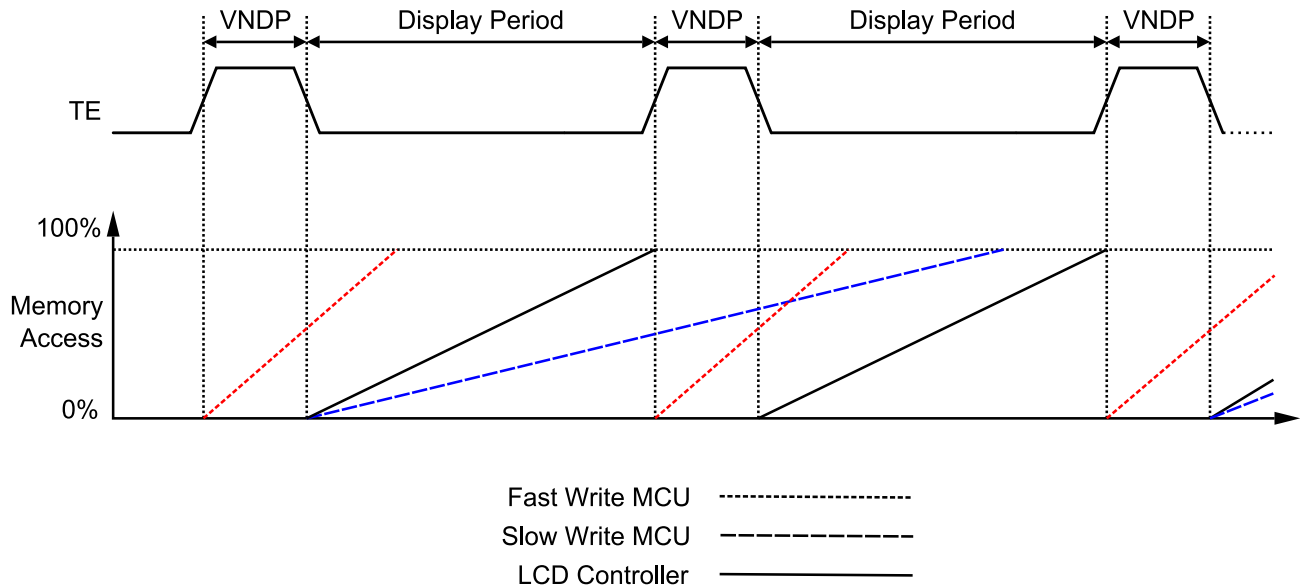
|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|

7.1.5 Tearing Effect Signal (TE)

The Tearing Effect Signal (TE) is a feedback signal from the LCD Controller to MCU. This signal reveals the display status of LCD controller. In the non-display period, the TE signal will go high. Therefore, this signal enables the MCU to send data by observing the non-display period to avoid tearing.

Figure 7-1 shows how the TE signal helps to avoid tearing. If the MCU writing speed is slower than the display speed, the display data should be updated after the LCD controller start to scan the frame buffer. Then the LCD controller will always display the old memory content until the next frame. However, if the MCU is faster than the LCD controller, it should start updating the display content in the vertical non-display period (VNDP) to enable the LCD controller will always get the newly updated data.

Figure 7-1: Relationship between Tearing Effect Signal and MCU Memory Writing



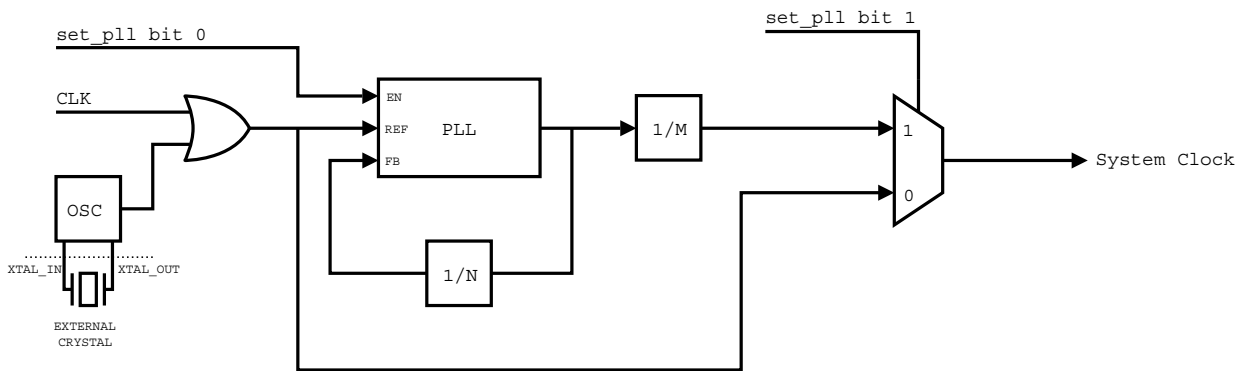
In SSD1963, users can configure the TE signal to reflect the vertical non-display period only or reflect both vertical and horizontal non-display period. With the additional horizontal non-display period information, the MCU can control the refresh action in more accurately by counting the horizontal line scanned by the LCD controller. Usually, a fast MCU will not need horizontal non-display period. But a slow MCU will need it to ensure the frame buffer update process always lags behind the LCD controller.

7.2 System Clock Generation

The system clock of SSD1963 is generated by the built-in PLL. The reference clock of the PLL can come from either the CLK pin or the external crystal oscillator. Since the CLK pin and the output of the oscillator was connected to PLL with an “OR” gate, the unused clock must be tied to VSS.

Before the PLL output is configured as the system clock by the bit 1 of “set_pll” command 0xE0, the system will be clocked by the reference clock. This enables the user to send the “set_pll_mn” command 0xE2 to the PLL for frequency configuration. When the PLL frequency is configured and the PLL was enabled with the bit 0 of “set_pll” command 0xE0, the user should still wait for 100us for the PLL to lock. Then the PLL is ready and can be configured as system clock with the bit 1 of “set_pll” command 0xE0.

Figure 7-2: Clock Control Diagram



7.3 Frame Buffer

There are 1215K bytes built-in SRAM inside SSD1963 to use as frame buffer. When the frame buffer is written or read, the “address counter” will automatically increase by one or decrease by one depends on the frame buffer settings.

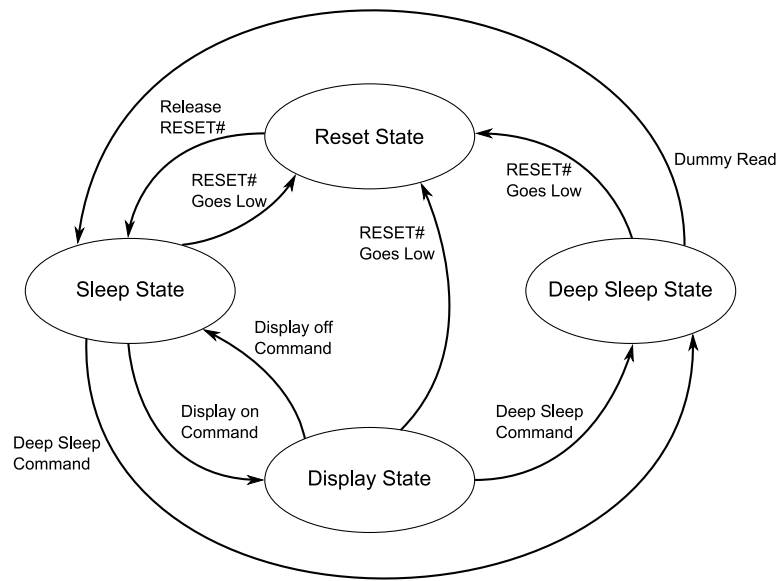
Table 7-2: Frame Buffer Settings regarding to set_address_mode command 0x36

| | Option 1 Horizontal: Increment Vertical: Increment B6 = 0; B7 = 0 | Option 2 Horizontal: Increment Vertical: Decrement B6 = 0; B7 = 1 | Option 3 Horizontal: Decrement Vertical: Increment B6 = 1; B7 = 0 | Option 4 Horizontal: Decrement Vertical: Decrement B6 = 1; B7 = 1 |
|---|--|--|--|--|
| Horizontal Frame Buffer Mode B5 = 0 | 00000h 5DC00h | 00000h 5DC00h | 00000h 5DC00h | 00000h 5DC00h |
| Vertical Frame Buffer Mode B5 = 1 | 00000h 5DC00h | 00000h 5DC00h | 00000h 5DC00h | 00000h 5DC00h |

7.4 System Clock and Reset Manager

The “System Clock and Reset Manager” distributes the reset signal and clock signal to the entire system. It controls the Clock Generator and contains clock gating circuitry to turn on and off the clock of each functional module. Also, it divides the root clock from Clock Generator to operation clocks for different module. The System Clock and Reset Manager also manage the reset signals to ensure all the module are reset to appropriate status when the system are in reset state, deep sleep state, sleep state and display state. Figure 7-3 shows a state diagram of four operation states of SSD1963.

Figure 7-3: State Diagram of SSD1963



Reset State:
 Clock Generator Stop
 Unable to Receive Command
 Unable to Update Frame Buffer
 Display Off
 All Settings Reset

Deep Sleep State:
 Clock Generator Stop
 Unable to Receive Command
 Unable to Update Frame Buffer
 Display Off
 All Settings Retain

Sleep State:
 Clock Generator On
 Able to Receive Command
 Able to Update Frame Buffer
 Display Off
 All Settings Retain

Display State:
 Clock Generator On
 Able to Receive Command
 Able to Update Frame Buffer
 Display On
 All Settings Retain

7.5 LCD Controller

7.5.1 Display Format

The LCD controller reads the frame buffer and generates display signals according to the selected display panel format. SSD1963 supports common RAM-less TFT driver using generic RGB data format.

7.5.2 General Purpose Input/Output (GPIO)

The GPIO pins can operate in 2 modes, GPIO mode and miscellaneous display signal mode. When the pins are configured as GPIOs, these pins can be controlled directly by MCU. Therefore, user can use these pins to emulate other interface such as SPI or I2C. If these pins are configured as display signals, they will toggle with display periodically according to the signal settings. They can be set to toggle once a frame, once a line or in arbitrary period. Therefore they can be configured as some common signal needed for different panels such as STH or LP.

8 COMMAND TABLE

| Hex Code | Command | Description |
|----------|------------------------|---|
| 0x00 | nop | No operation |
| 0x01 | soft_reset | Software Reset |
| 0x0A | get_power_mode | Get the current power mode |
| 0x0B | get_address_mode | Get the frame buffer to the display panel read order |
| 0x0C | Reserved | Reserved |
| 0x0D | get_display_mode | The SSD1963 returns the Display Image Mode. |
| 0x0E | get_tear_effect_status | Get the Tear Effect status |
| 0x0F | Reserved | Reserved |
| 0x10 | enter_sleep_mode | Turn off the panel. This command will pull low the GPIO0. If GPIO0 is configured as normal GPIO or LCD miscellaneous signal with command set_gpio_conf, this command will be ignored. |
| 0x11 | exit_sleep_mode | Turn on the panel. This command will pull high the GPIO0. If GPIO0 is configured as normal GPIO or LCD miscellaneous signal with command set_gpio_conf, this command will be ignored. |
| 0x12 | enter_partial_mode | Part of the display area is used for image display. |
| 0x13 | enter_normal_mode | The whole display area is used for image display. |
| 0x20 | exit_invert_mode | Displayed image colors are not inverted. |
| 0x21 | enter_invert_mode | Displayed image colors are inverted. |
| 0x26 | set_gamma_curve | Selects the gamma curve used by the display panel. |
| 0x28 | set_display_off | Blanks the display panel |
| 0x29 | set_display_on | Show the image on the display panel |
| 0x2A | set_column_address | Set the column address |
| 0x2B | set_page_address | Set the page address |
| 0x2C | write_memory_start | Transfer image information from the host processor interface to the SSD1963 starting at the location provided by set_column_address and set_page_address |
| 0x2E | read_memory_start | Transfer image data from the SSD1963 to the host processor interface starting at the location provided by set_column_address and set_page_address |
| 0x30 | set_partial_area | Defines the partial display area on the display panel |
| 0x33 | set_scroll_area | Defines the vertical scrolling and fixed area on display area |
| 0x34 | set_tear_off | Synchronization information is not sent from the SSD1963 to the host processor |
| 0x35 | set_tear_on | Synchronization information is sent from the SSD1963 to the host processor at the start of VFP |
| 0x36 | set_address_mode | Set the read order from frame buffer to the display panel |
| 0x37 | set_scroll_start | Defines the vertical scrolling starting point |
| 0x38 | exit_idle_mode | Full color depth is used for the display panel |
| 0x39 | enter_idle_mode | Reduce color depth is used on the display panel. |
| 0x3A | Reserved | Reserved |
| 0x3C | write_memory_continue | Transfer image information from the host processor interface to the SSD1963 from the last written location |
| 0x3E | read_memory_continue | Read image data from the SSD1963 continuing after the last read_memory_continue or read_memory_start |

| Hex Code | Command | Description |
|----------|-------------------|--|
| 0x44 | set_tear_scanline | Synchronization information is sent from the SSD1963 to the host processor when the display panel refresh reaches the provided scanline |
| 0x45 | get_scanline | Get the current scan line |
| 0xA1 | read_ddb | Read the DDB from the provided location |
| 0xA8 | Reserved | Reserved |
| 0xB0 | set_lcd_mode_ | Set the LCD panel mode and resolution |
| 0xB1 | get_lcd_mode | Get the current LCD panel mode, pad strength and resolution |
| 0xB4 | set_hori_period | Set front porch |
| 0xB5 | get_hori_period | Get current front porch settings |
| 0xB6 | set_vert_period | Set the vertical blanking interval between last scan line and next LFRAME pulse |
| 0xB7 | get_vert_period | Set the vertical blanking interval between last scan line and next LFRAME pulse |
| 0xB8 | set_gpio_conf | Set the GPIO configuration. If the GPIO is not used for LCD, set the direction. Otherwise, they are toggled with LCD signals. |
| 0xB9 | get_gpio_conf | Get the current GPIO configuration |
| 0xBA | set_gpio_value | Set GPIO value for GPIO configured as output |
| 0xBB | get_gpio_status | Read current GPIO status. If the individual GPIO was configured as input, the value is the status of the corresponding pin. Otherwise, it is the programmed value. |
| 0xBC | set_post_proc | Set the image post processor |
| 0xBD | get_post_proc | Set the image post processor |
| 0xBE | set_pwm_conf | Set the image post processor |
| 0xBF | get_pwm_conf | Set the image post processor |
| 0xC0 | set_lcd_gen0 | Set the rise, fall, period and toggling properties of LCD signal generator 0 |
| 0xC1 | get_lcd_gen0 | Get the current settings of LCD signal generator 0 |
| 0xC2 | set_lcd_gen1 | Set the rise, fall, period and toggling properties of LCD signal generator 1 |
| 0xC3 | get_lcd_gen1 | Get the current settings of LCD signal generator 1 |
| 0xC4 | set_lcd_gen2 | Set the rise, fall, period and toggling properties of LCD signal generator 2 |
| 0xC5 | get_lcd_gen2 | Get the current settings of LCD signal generator 2 |
| 0xC6 | set_lcd_gen3 | Set the rise, fall, period and toggling properties of LCD signal generator 3 |
| 0xC7 | get_lcd_gen3 | Get the current settings of LCD signal generator 3 |
| 0xC8 | set_gpio0_rop | Set the GPIO0 with respect to the LCD signal generators using ROP operation. No effect if the GPIO0 is configured as general GPIO. |
| 0xC9 | get_gpio0_rop | Get the GPIO0 properties with respect to the LCD signal generators. |
| 0xCA | set_gpio1_rop | Set the GPIO1 with respect to the LCD signal generators using ROP operation. No effect if the GPIO1 is configured as general GPIO. |
| 0xCB | get_gpio1_rop | Get the GPIO1 properties with respect to the LCD signal generators. |
| 0xCC | set_gpio2_rop | Set the GPIO2 with respect to the LCD signal generators using ROP operation. No effect if the GPIO2 is configured as general GPIO. |

| Hex Code | Command | Description |
|----------|--------------------------|--|
| 0xCD | get_gpio2_rop | Get the GPIO2 properties with respect to the LCD signal generators. |
| 0xCE | set_gpio3_rop | Set the GPIO3 with respect to the LCD signal generators using ROP operation. No effect if the GPIO3 is configured as general GPIO. |
| 0xCF | get_gpio3_rop | Get the GPIO3 properties with respect to the LCD signal generators. |
| 0xD0 | set_dbc_conf | Set the dynamic back light configuration |
| 0xD1 | get_dbc_conf | Get the current dynamic back light configuration |
| 0xD4 | set_dbc_th | Set the threshold for each level of power saving |
| 0xD5 | get_dbc_th | Get the threshold for each level of power saving |
| 0xE0 | set_pll | Start the PLL. Before the start, the system was operated with the crystal oscillator or clock input |
| 0xE2 | set_pll_mn | Set the PLL |
| 0xE3 | get_pll_mn | Get the PLL settings |
| 0xE4 | get_pll_status | Get the current PLL status |
| 0xE5 | set_deep_sleep | Set deep sleep mode |
| 0xE6 | set_lshift_freq | Set the LSHIFT (pixel clock) frequency |
| 0xE7 | get_lshift_freq | Get current LSHIFT (pixel clock) frequency setting |
| 0xE8 | Reserved | Reserved |
| 0xE9 | Reserved | Reserved |
| 0xF0 | set_pixel_data_interface | Set the pixel data format of the parallel host processor interface |
| 0xF1 | get_pixel_data_interface | Get the current pixel data format settings |
| 0xFF | Reserved | Reserved |

9 COMMAND DESCRIPTIONS

9.1 nop

Command 0x00
Parameters None

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|---------|-----|----|----|----|----|----|----|----|----|-----|
| Command | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 00 |

Description
No operation.

9.2 soft_reset

Command 0x01
Parameters None

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|---------|-----|----|----|----|----|----|----|----|----|-----|
| Command | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 01 |

Description
The SSD1963 performs a software reset. All the configuration register will be reset except command 0xE0 to 0xE5.

Note :

The host processor must wait 5ms before sending any new commands to a SSD1963 following this command.

9.3 get_power_mode

Command 0x0A
Parameters 1

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----|----------------|----------------|----------------|----------------|----------------|----|----|-----|
| Command | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0A |
| Parameter 1 | 1 | 0 | A ₆ | A ₅ | A ₄ | A ₃ | A ₂ | 0 | 0 | xx |

Description
Get the current power mode

A[6] : Idle mode on/off (POR = 0)
0 Idle mode off
1 Idle mode on

A[5] : Partial mode on/off (POR = 0)
0 Partial mode off
1 Partial mode on

A[4] : Sleep mode on/off (POR = 0)
0 Sleep mode on
1 Sleep mode off

A[3] : Display normal mode on/off (POR = 1)
0 Display normal mode off

- 1 Display normal mode on (partial mode and vertical scroll off)

A[2] : Display on/off (POR = 0)

- 0 Display is off
1 Display is on

9.4 get_address_mode

Command 0x0B

Parameters 1

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----------------|----------------|----------------|----------------|----------------|----------------|----|----|-----|
| Command | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0B |
| Parameter 1 | 1 | A ₇ | A ₆ | A ₅ | A ₄ | A ₃ | A ₂ | 0 | 0 | xx |

Description

Get the frame buffer to the display panel read order

A[7] : Page address order (POR = 0)

- 0 Top to bottom
1 Bottom to top

A[6] : Column address order (POR = 0)

- 0 Left to right
1 Right to left

A[5] : Page / Column order (POR = 0)

- 0 Normal mode
1 Reverse mode

A[4] : Line address order (POR = 0)

- 0 LCD refresh top to bottom
1 LCD refresh bottom to top

A[3] : RGB / BGR order (POR = 0)

- 0 RGB
1 BGR

A[2] : Display data latch data (POR = 0)

- 0 LCD refresh left to right
1 LCD refresh right to left

9.5 get_display_mode

Command 0x0D

Parameters 1

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----------------|----|----------------|----|----|----------------|----------------|----------------|-----|
| Command | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0D |
| Parameter 1 | 1 | A ₇ | 0 | A ₅ | 0 | 0 | A ₂ | A ₁ | A ₀ | xx |

Description

Get the Display Image Mode status.

A[7] : Vertical scrolling on/off (POR = 0)

- 0 Vertical scrolling is off

1 Vertical scrolling is on

A[5] : Invert mode on/off (POR = 0)

0 Inversion is off
1 Inversion is on

A[2:0] : Gamma curve selection (POR = 011)

000 Gamma curve 0
001 Gamma curve 1
010 Gamma curve 2
011 Gamma curve 3
100 Reserved
101 Reserved
110 Reserved
111 Reserved

9.6 get_tear_effect_status

Command 0x0E

Parameters 1

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----------------|----|----|----|----|----|----|----|-----|
| Command | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0E |
| Parameter 1 | 1 | A ₇ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | xx |

Description

Get the current Tear Effect mode from the SSD1963

A[7] : Tearing effect line mode (POR = 0)

0 Tearing effect off
1 Tearing effect on

9.7 enter_sleep_mode

Command 0x10
Parameters None

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|---------|-----|----|----|----|----|----|----|----|----|-----|
| Command | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 10 |

Description

Turn off the panel. This command causes the SSD1963 to enter sleep mode and pull low the GPIO[0] if set_gpio_conf (0xB8) B0 = 0

If GPIO[0] is configured as normal GPIO or LCD miscellaneous signal with command set_gpio_conf (0xB8), this command will not affect the GPIO[0].

Note :

The host processor must wait 5ms before sending any new commands to a SSD1963 following this command.

9.8 exit_sleep_mode

Command 0x11
Parameters None

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|---------|-----|----|----|----|----|----|----|----|----|-----|
| Command | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 11 |

Description

Turn on the panel. This command causes the SSD1963 to exit sleep mode and will pull high the GPIO[0] if set_gpio_conf (0xB8) B0 = 0.

If GPIO[0] is configured as normal GPIO or LCD miscellaneous signal with command set_gpio_conf (0xB8), this command will not affect the GPIO[0].

Note :

The host processor must wait 5ms after sending this command before sending another command.

**This command will automatic trigger set_display_on (0x29)

9.9 enter_partial_mode

Command 0x12
Parameters None

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|---------|-----|----|----|----|----|----|----|----|----|-----|
| Command | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 12 |

Description

Once enter_partial_mode is triggered, the Partial Display Mode window is described by the set_partial_area (0x30).

Once enter_normal_mode (0x13) is triggered, partial display mode will end.

9.10 enter_normal_mode

Command 0x13
Parameters None

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|---------|-----|----|----|----|----|----|----|----|----|-----|
| Command | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 13 |

Description

This command causes the SSD1963 to enter the normal mode. Normal mode is defined as partial display and vertical scroll mode are off. That means the whole display area is used for image display.

9.11 exit_invert_mode

Command 0x20

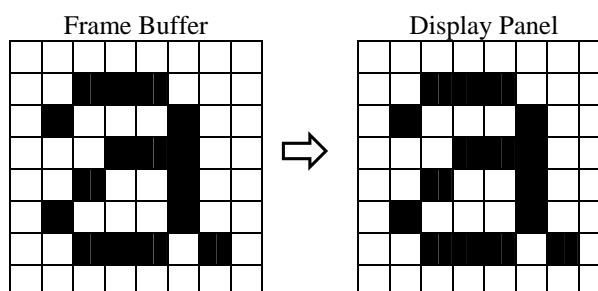
Parameters None

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|---------|-----|----|----|----|----|----|----|----|----|-----|
| Command | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 20 |

Description

This command causes the SSD1963 to stop inverting the image data on the display panel. The frame buffer contents remain unchanged.

Figure 9-1: Exit Invert mode example



9.12 enter_invert_mode

Command 0x21

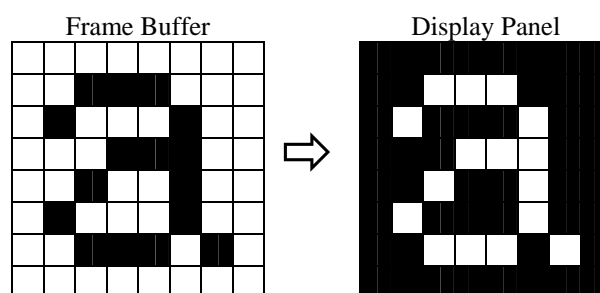
Parameters None

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|---------|-----|----|----|----|----|----|----|----|----|-----|
| Command | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 21 |

Description

This command causes the SSD1963 to invert the image data only on the display panel. The frame buffer contents remain unchanged.

Figure 9-2: Enter Invert mode example



9.13 set_gamma_curve

Command 0x26
Parameters 1

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----|----|----|----|----------------|----------------|----------------|----------------|-----|
| Command | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 26 |
| Parameter 1 | 1 | 0 | 0 | 0 | 0 | A ₃ | A ₂ | A ₁ | A ₀ | xx |

Description

Selects the gamma curve used by the display panel.

| A[3:0] | Gamma curve selection (POR = 1000) | GAMAS[1] | GAMAS[0] |
|--------|---|----------|----------|
| 0000 | No gamma curve selected (Same as 0001b) | 0 | 0 |
| 0001 | Gamma curve 0 | 0 | 0 |
| 0010 | Gamma curve 1 | 0 | 1 |
| 0100 | Gamma curve 2 | 1 | 0 |
| 1000 | Gamma curve 3 | 1 | 1 |
| Others | Reserved | | |

9.14 set_display_off

Command 0x28
Parameters None

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|---------|-----|----|----|----|----|----|----|----|----|-----|
| Command | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 28 |

Description

Blanks the display panel. The frame buffer contents remain unchanged.

9.15 set_display_on

Command 0x29
Parameters None

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|---------|-----|----|----|----|----|----|----|----|----|-----|
| Command | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 29 |

Description

Show the image on the display panel

9.16 set_column_address

Command 0x2A
Parameters 4

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|------------------|------------------|------------------|------------------|------------------|------------------|-----------------|-----------------|-----|
| Command | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 2A |
| Parameter 1 | 1 | SC ₁₅ | SC ₁₄ | SC ₁₃ | SC ₁₂ | SC ₁₁ | SC ₁₀ | SC ₉ | SC ₈ | xx |
| Parameter 2 | 1 | SC ₇ | SC ₆ | SC ₅ | SC ₄ | SC ₃ | SC ₂ | SC ₁ | SC ₀ | xx |
| Parameter 3 | 1 | EC ₁₅ | EC ₁₄ | EC ₁₃ | EC ₁₂ | EC ₁₁ | EC ₁₀ | EC ₉ | EC ₈ | xx |
| Parameter 4 | 1 | EC ₇ | EC ₆ | EC ₅ | EC ₄ | EC ₃ | EC ₂ | EC ₁ | EC ₀ | xx |

Description

Set the column address of frame buffer accessed by the host processor with the read_memory_continue (0x3E) and write_memory_continue (0x3C)..

SC[15:8] : Start column number high byte (POR = 00000000)

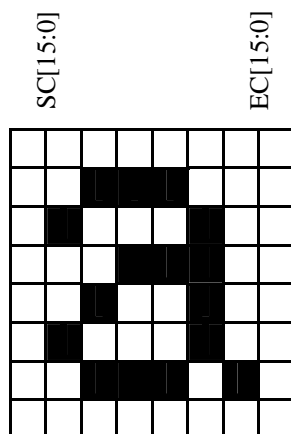
SC[7:0] : Start column number low byte (POR = 00000000)

EC[15:8] : End column number high byte (POR = 00000000)

EC[7:0] : End column number low byte (POR = 00000000)

Note : SC[15:0] must always be equal to or less than EC[15:0]

Figure 9-3: Set Column Address example



9.17 set_page_address

Command 0x2B
Parameters 4

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|------------------|------------------|------------------|------------------|------------------|------------------|-----------------|-----------------|-----|
| Command | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 2B |
| Parameter 1 | 1 | SP ₁₅ | SP ₁₄ | SP ₁₃ | SP ₁₂ | SP ₁₁ | SP ₁₀ | SP ₉ | SP ₈ | xx |
| Parameter 2 | 1 | SP ₇ | SP ₆ | SP ₅ | SP ₄ | SP ₃ | SP ₂ | SP ₁ | SP ₀ | xx |
| Parameter 3 | 1 | EP ₁₅ | EP ₁₄ | EP ₁₃ | EP ₁₂ | EP ₁₁ | EP ₁₀ | EP ₉ | EP ₈ | xx |
| Parameter 4 | 1 | EP ₇ | EP ₆ | EP ₅ | EP ₄ | EP ₃ | EP ₂ | EP ₁ | EP ₀ | xx |

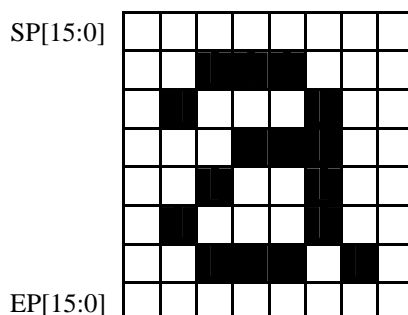
Description

Set the page address of the frame buffer accessed by the host processor with the read_memory_start (0x2C), write_memory_start (0x2E), read_memory_continue (0x3E) and write_memory_continue (0x3C)..

SP[15:8] : Start page (row) number high byte (POR = 00000000)
 SP[7:0] : Start page (row) number low byte (POR = 00000000)
 EP[15:8] : End page (row) number high byte (POR = 00000000)
 EP[7:0] : End page (row) number low byte (POR = 00000000)

Note : SP[15:0] must always be equal to or less than EP[15:0]

Figure 9-4: Set Page Address example



9.18 write_memory_start

Command 0x2C
Parameters None

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|---------|-----|----|----|----|----|----|----|----|----|-----|
| Command | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 2C |

Description

Transfer image information from the host processor interface to the SSD1963 starting at the location provided by set_column_address (0x2A) and set_page_address (0x2B).

If set_address_mode (0x36) A[5] = 0:

The column and page address are reset to the Start Column (SC) and Start Page (SP), respectively.

Pixel Data 1 is stored in frame buffer at (SC, SP). The column address is then incremented and pixels are written to the frame buffer until the column address equals the End Column (EC) value. The column address is then reset to SC and the page address is incremented. Pixels are written to the frame buffer until the page address equals the End Page (EP) value and the column address equals the EC value, or the host processor sends another command. If the number of pixels exceeds $(EC - SC + 1) * (EP - SP + 1)$ the extra pixels are ignored.

If set_address_mode (0x36) A[5] = 1:

The column and page address are reset to the Start Column (SC) and Start Page (SP), respectively.

Pixel Data 1 is stored in frame buffer at (SC, SP). The page address is then incremented and pixels are written to the frame buffer until the page address equals the End Page (EP) value. The page address is then reset to SP and the column address is incremented. Pixels are written to the frame buffer until the column address equals the End column (EC) value and the page address equals the EP value, or the host processor sends another command. If the number of pixels exceeds $(EC - SC + 1) * (EP - SP + 1)$ the extra pixels are ignored.

9.19 read_memory_start

Command 0x2E
Parameters None

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|---------|-----|----|----|----|----|----|----|----|----|-----|
| Command | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 2E |

Description

Transfer image data from the SSD1963 to the host processor interface starting at the location provided by set_column_address (0x2A) and set_page_address (0x2B).

If set_address_mode A[5] = 0:

The column and page address are reset to the Start Column (SC) and Start Page (SP), respectively.

Pixels Data 1 are read from frame buffer at (SC, SP). The column address is then incremented and pixels read from the frame buffer until the column address equals the End Column (EC) value. The column address is then reset to SC and the page address is incremented. Pixels are read from the frame buffer until the page address equals the End Page (EP) value and the column address equals the EC value, or the host processor sends another command.

If set_address_mode (0x36) A[5] = 1:

The column and page address are reset to the Start Column (SC) and Start Page (SP), respectively.

Pixels Data 1 are read from frame buffer at (SC, SP). The page address is then incremented and pixels read from the frame buffer until the page address equals the End Page (EP) value. The page address is then reset to SP and the column address is incremented. Pixels are read from the frame buffer until the column address equals the End Column (EC) value and the page address equals the EP value, or the host processor sends another command.

9.20 set_partial_area

Command 0x30
Parameters 4

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|------------------|------------------|------------------|------------------|------------------|------------------|-----------------|-----------------|-----|
| Command | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 30 |
| Parameter 1 | 1 | SR ₁₅ | SR ₁₄ | SR ₁₃ | SR ₁₂ | SR ₁₁ | SR ₁₀ | SR ₉ | SR ₈ | xx |
| Parameter 2 | 1 | SR ₇ | SR ₆ | SR ₅ | SR ₄ | SR ₃ | SR ₂ | SR ₁ | SR ₀ | xx |
| Parameter 3 | 1 | ER ₁₅ | ER ₁₄ | ER ₁₃ | ER ₁₂ | ER ₁₁ | ER ₁₀ | ER ₉ | ER ₈ | xx |
| Parameter 4 | 1 | ER ₇ | ER ₆ | ER ₅ | ER ₄ | ER ₃ | ER ₂ | ER ₁ | ER ₀ | xx |

Description

This command defines the Partial Display mode's display area. There are two parameters associated with this command, the first defines the Start Row (SR) and the second the End Row (ER). SR and ER refer to the Frame Buffer Line Pointer.

SR[15:8] : Start display row number high byte (POR = 00000000)

SR[7:0] : Start display row number low byte (POR = 00000000)

ER[15:8] : End display row number high byte (POR = 00000000)

ER[7:0] : End display row number low byte (POR = 00000000)

Note : SR[15:0] and ER[15:0] cannot be 0000h nor exceed the last vertical line number.

If End Row > Start Row

Figure 9-5: Set Partial Area with set_address_mode (0x36) A[4] = 0 when End Row > Start Row

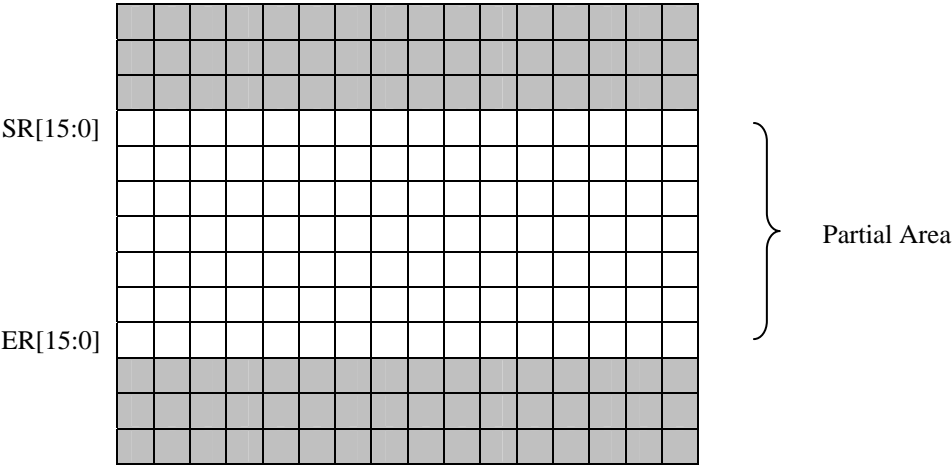
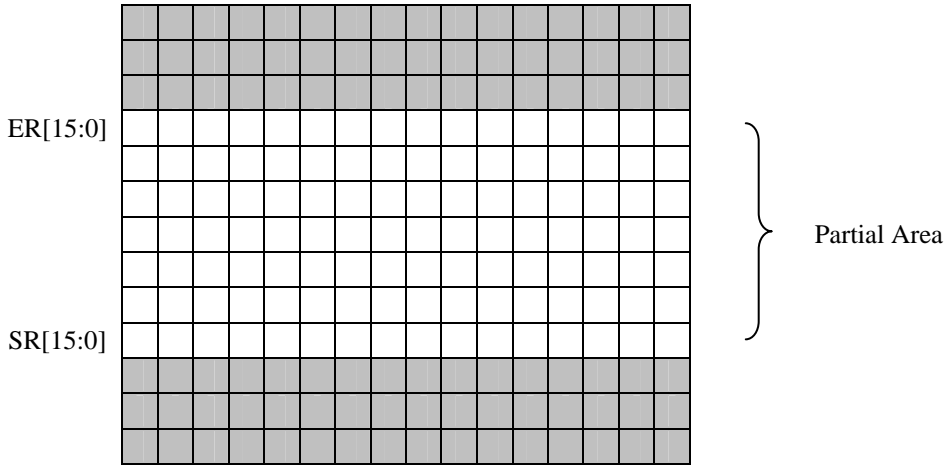


Figure 9-6: Set Partial Area with set_address_mode (0x36) A[4] = 1 when End Row > Start Row



If Start Row > End Row

Figure 9-7: Set Partial Area with set_address_mode (0x36) A[4] = 0 when Start Row > End Row

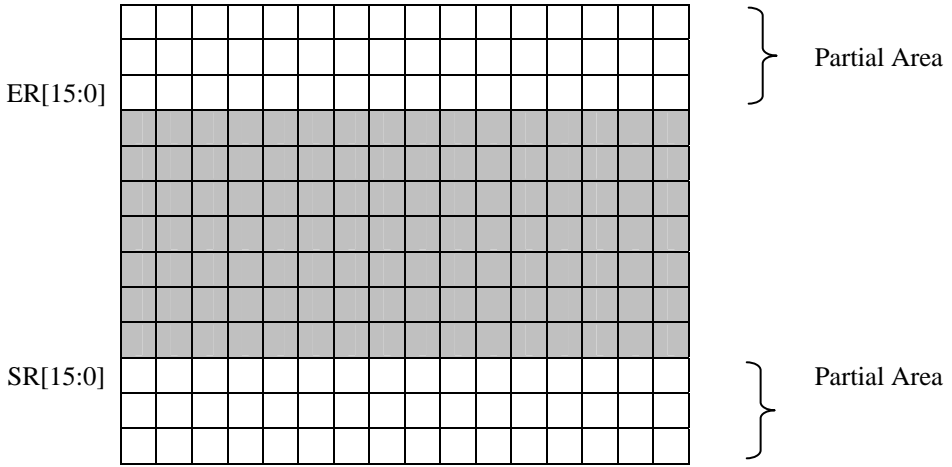
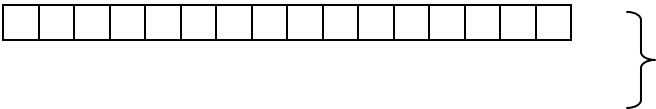
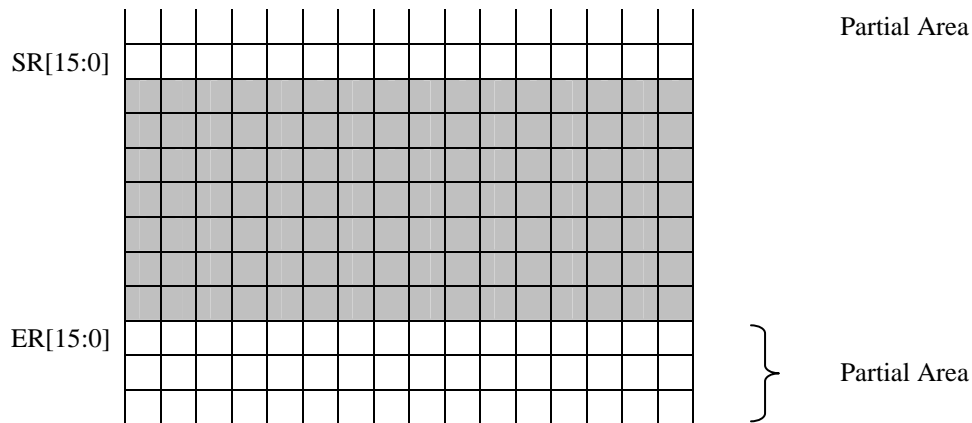


Figure 9-8: Set Partial Area with set_address_mode (0x36) A[4] = 1 when Start Row > End Row





9.21 set_scroll_area

Command 0x33
Parameters 6

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|-----|
| Command | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 33 |
| Parameter 1 | 1 | TFA ₁₅ | TFA ₁₄ | TFA ₁₃ | TFA ₁₂ | TFA ₁₁ | TFA ₁₀ | TFA ₉ | TFA ₈ | xx |
| Parameter 2 | 1 | TFA ₇ | TFA ₆ | TFA ₅ | TFA ₄ | TFA ₃ | TFA ₂ | TFA ₁ | TFA ₀ | xx |
| Parameter 3 | 1 | VSA ₁₅ | VSA ₁₄ | VSA ₁₃ | VSA ₁₂ | VSA ₁₁ | VSA ₁₀ | VSA ₉ | VSA ₈ | xx |
| Parameter 4 | 1 | VSA ₇ | VSA ₆ | VSA ₅ | VSA ₄ | VSA ₃ | VSA ₂ | VSA ₁ | VSA ₀ | xx |
| Parameter 5 | 1 | BFA ₁₅ | BFA ₁₄ | BFA ₁₃ | BFA ₁₂ | BFA ₁₁ | BFA ₁₀ | BFA ₉ | BFA ₈ | xx |
| Parameter 6 | 1 | BFA ₇ | BFA ₆ | BFA ₅ | BFA ₄ | BFA ₃ | BFA ₂ | BFA ₁ | BFA ₀ | xx |

Description

Defines the vertical scrolling and fixed area on display area

TFA[15:8] : High byte of Top Fixed Area number in lines from the top of the frame buffer (POR = 00000000)

TFA[7:0] : Low byte of Top Fixed Area number in lines from the top of the frame buffer (POR = 00000000)

VSA[15:8] : High byte of Vertical scrolling area in number of lines of the frame buffer (POR = 00000000)

VSA[7:0] : Low byte of Vertical scrolling area in number of lines of the frame buffer (POR = 00000000)

BFA[15:8] : High byte of Bottom Fixed Area in number of lines from the bottom of the frame buffer (POR = 00000000)

BFA[7:0] : Low byte of Bottom Fixed Area in number of lines from the bottom of the frame buffer (POR = 00000000)

If set_address_mode (0x36) A[4] = 0 :

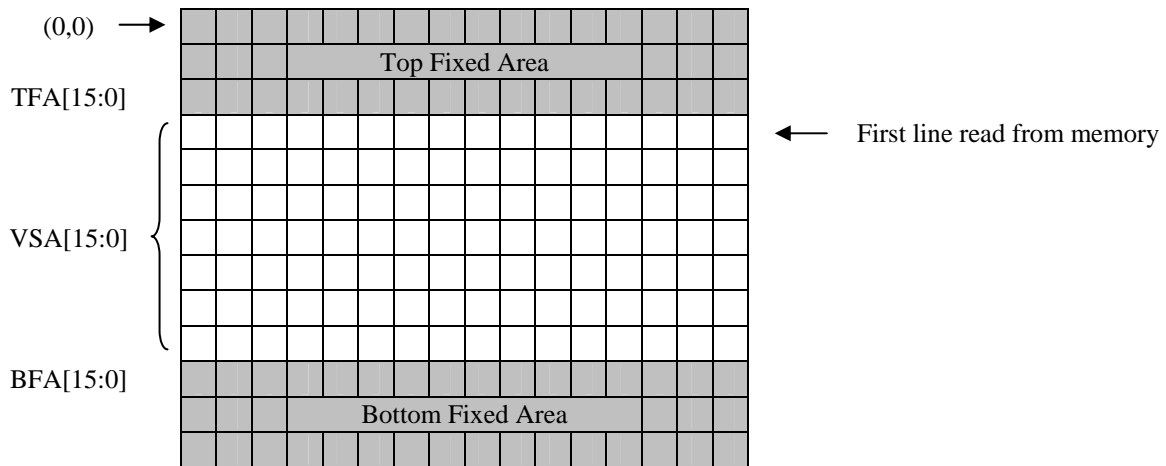
The TFA[15:0] describes the Top Fixed Area in number of lines from the top of the frame buffer. The top of the frame buffer and top of the display panel are aligned.

The VSA[15:0] describes the height of the Vertical Scrolling Area in number of lines of frame buffer from the Vertical Scrolling Start Address. The first line of the Vertical Scrolling Area starts immediately after the bottom most line of the Top Fixed Area. The last line of the Vertical Scrolling Area ends immediately before the top most line of the Bottom Fixed Area.

The BFA[15:0] describes the Bottom Fixed Area in number of lines from the bottom of the frame buffer. The bottom of the frame buffer and bottom of the display panel are aligned.

TFA, VSA and BFA refer to the Frame Buffer Line Pointer.

Figure 9-9: Set Scroll Area with set_address_mode (0x36) A[4] = 0



If set_address_mode (0x36) A[4] = 1 :

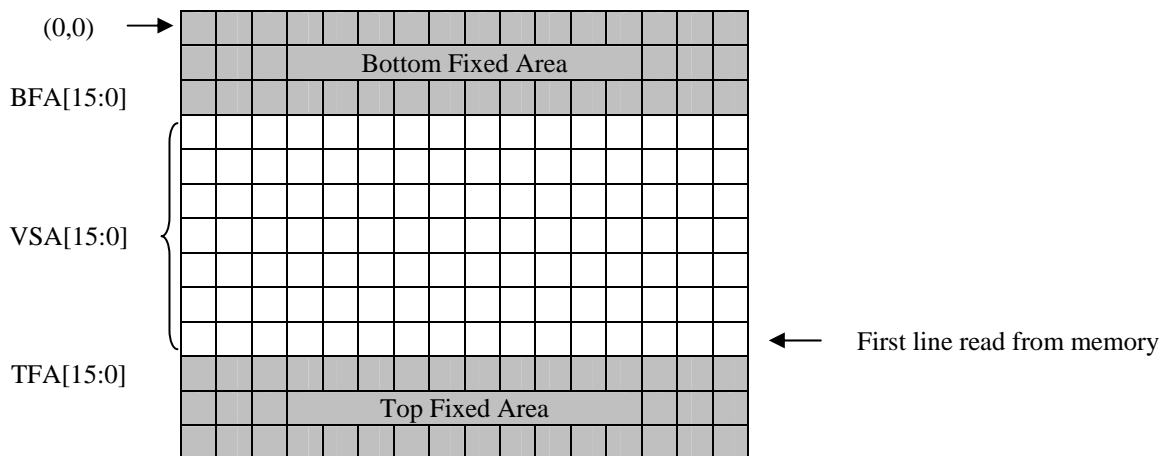
The TFA[15:0], describes the Top Fixed Area in number of lines from the bottom of the frame buffer. The bottom of the frame buffer and bottom of the display panel are aligned.

The VSA[15:0] describes the height of the Vertical Scrolling Area in number of lines of frame buffer from the Vertical Scrolling Start Address. The first line of the Vertical Scrolling Area starts immediately after the top most line of the Top Fixed Area. The last line of the Vertical Scrolling Area ends immediately before the bottom most line of the Bottom Fixed Area.

The BFA[15:0] describes the Bottom Fixed Area in number of lines from the top of the frame buffer. The top of the frame buffer and top of the display panel are aligned.

TFA, VSA and BFA refer to the Frame Buffer Line Pointer.

Figure 9-10: Set Scroll Area with set_address_mode (0x36) A[4] = 1



Note :

The sum of TFA, VSA and BFA must equal the number of the display panel's horizontal lines (pages), otherwise Scrolling mode is undefined.

In Vertical Scroll Mode, set_address_mode (0x36) A[5] should be set to '0' – this only affects the Frame Buffer Write.

9.22 set_tear_off

Command 0x34
Parameters None

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|---------|-----|----|----|----|----|----|----|----|----|-----|
| Command | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 34 |

Description

TE signal is not sent from the SSD1963 to the host processor.

9.23 set_tear_on

Command 0x35
Parameters 1

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----|----|----|----|----|----|----|----------------|-----|
| Command | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 35 |
| Parameter 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | A ₀ | xx |

Description

TE signal is sent from the SSD1963 to the host processor at the start of VFP.

A[0] : Tearing effect line mode (POR = 0)

- 0 The tearing effect output line consists of V-blanking information only.
- 1 The tearing effect output line consists of both V-blanking and H-blanking information by set_tear_scanline (0x44).

The TE signal shall be active low when the display panel is in Sleep mode.

9.24 set_address_mode

Command 0x36
Parameters 1

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|-----|
| Command | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 36 |
| Parameter 1 | 1 | A ₇ | A ₆ | A ₅ | A ₄ | A ₃ | A ₂ | A ₁ | A ₀ | xx |

Description

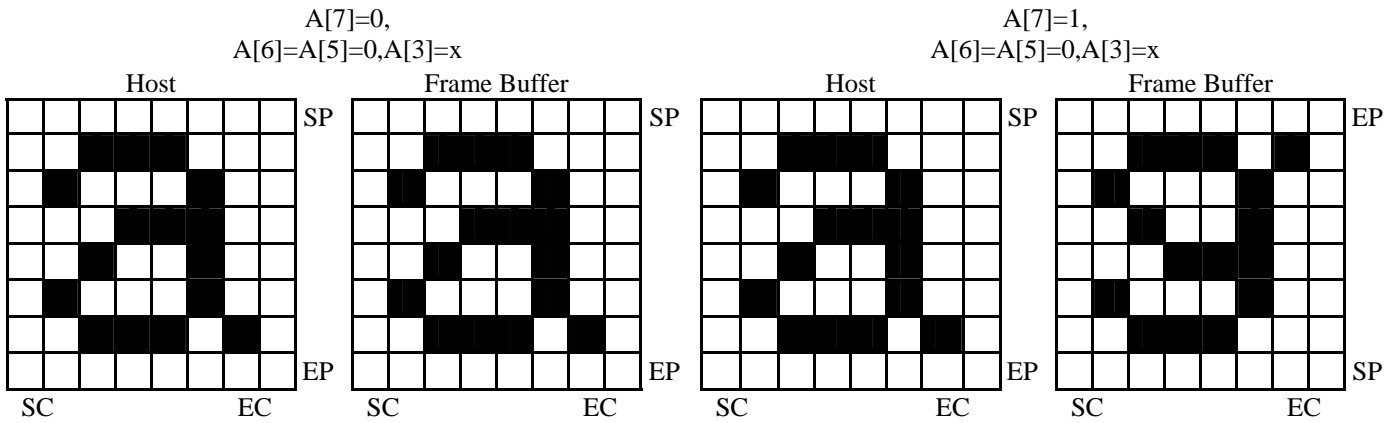
Set the read order from host processor to frame buffer by A[7:5] and A[3] and from frame buffer to the display panel by A[2:0] and A[4].

A[7] : Page address order (POR = 0)

This bit controls the order that pages of data are transferred from the host processor to the SSD1963's frame buffer.

- 0 Top to bottom, pages transferred from SP (Start Page) to EP (End Page).
- 1 Bottom to top, pages transferred from EP (End Page) to SP (Start Page).

Figure 9-11: A[7] Page Address Order

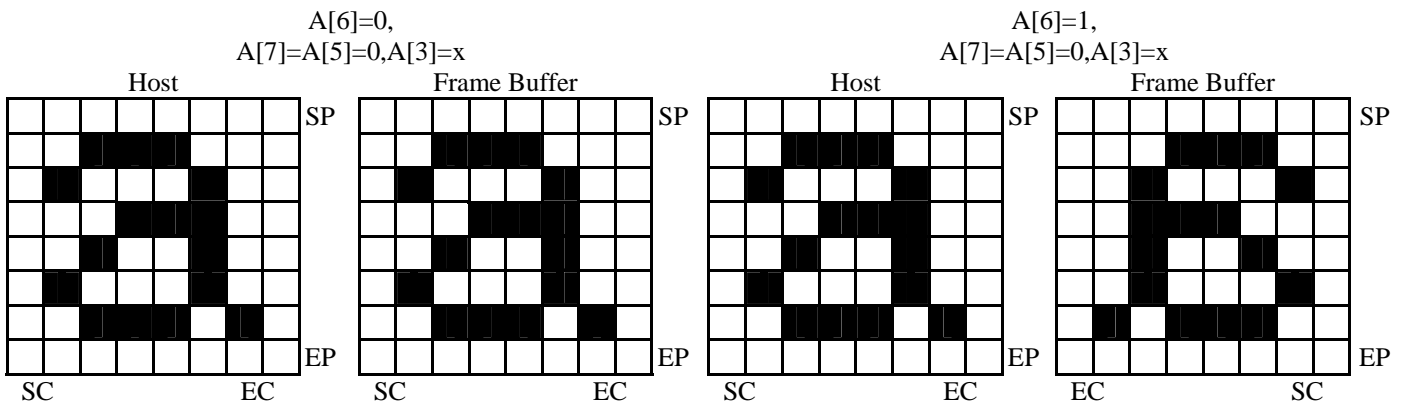


A[6] : Column address order (POR = 0)

This bit controls the order that columns of data are transferred from the host processor to the SSD1963's frame buffer.

- 0 Left to right, columns transferred from SC (Start Column) to EC (End Column).
- 1 Right to left, columns transferred from EC (End Column) to SC (Start Column).

Figure 9-12: A[6] Column Address Order

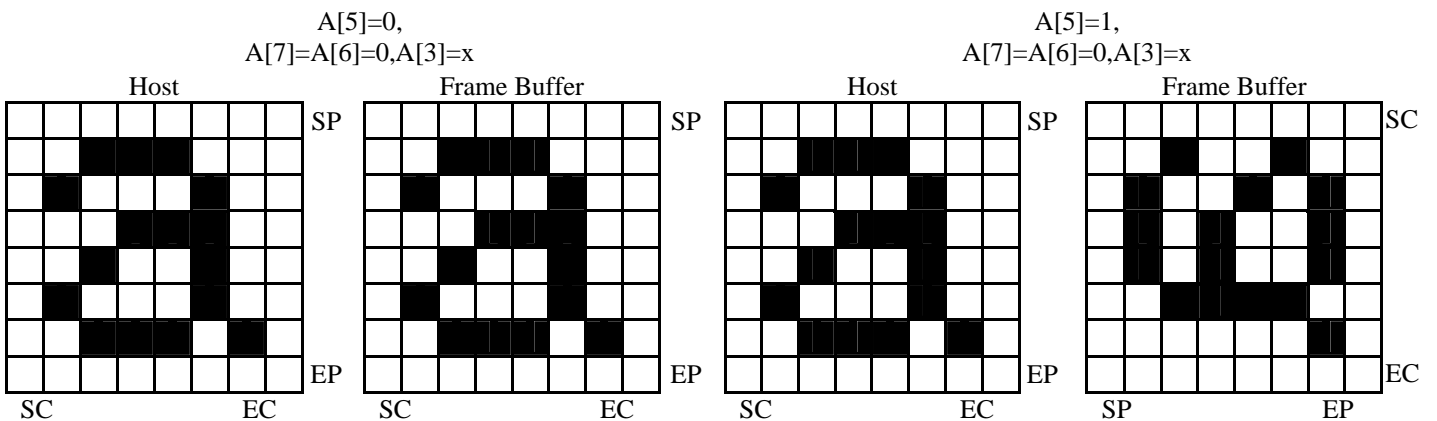


A[5] : Page / Column order (POR = 0)

This bit controls the order that columns of data are transferred from the host processor to the SSD1963's frame buffer.

- 0 Normal mode
- 1 Reverse mode

Figure 9-13: A[5] Page / Column Address Order



A[4] : Line address order (POR = 0)

This bit controls the display panel's horizontal line refresh order. The image shown on the display panel is unaffected, regardless of the bit setting.

- 0 LCD refresh from top line to bottom line.
- 1 LCD refresh from bottom line to top line.

A[3] : RGB / BGR order (POR = 0)

This bit controls the RGB data order transferred from the SSD1963's frame buffer to the display panel.

- 0 RGB
- 1 BGR

Figure 9-14: A[3] RGB Order



A[2] : Display data latch data (POR = 0)

This bit controls the display panel's vertical line data latch order. The image shown on the display panel is unaffected, regardless of the bit setting.

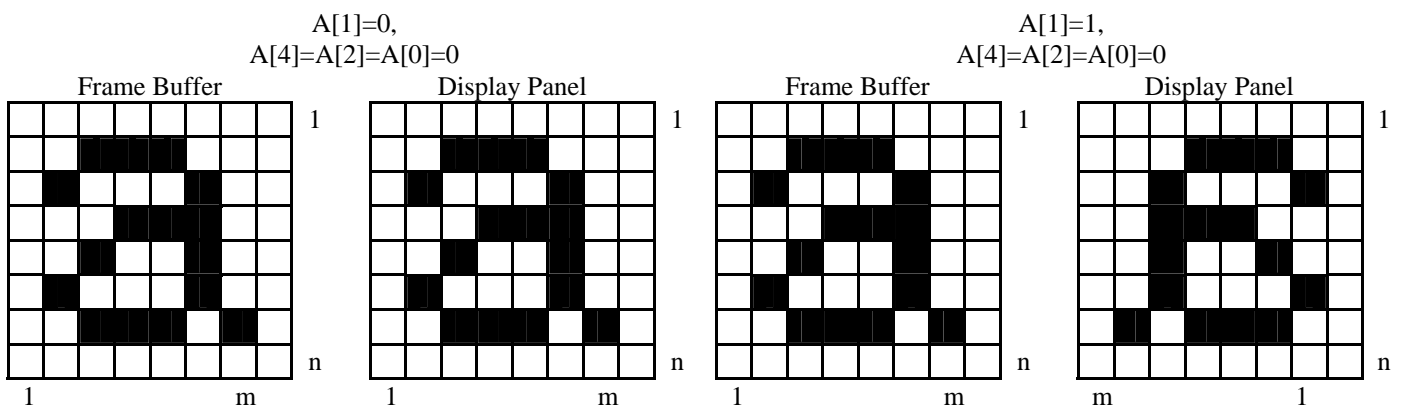
- 0 LCD refresh from left side to right side
- 1 LCD refresh from right side to left side

A[1] : Flip Horizontal (POR = 0)

This bit flips the image shown on the display panel left to right. No change is made to the frame buffer.

- 0 Normal
- 1 Flipped

Figure 9-15: A[1] Flip Horizontal

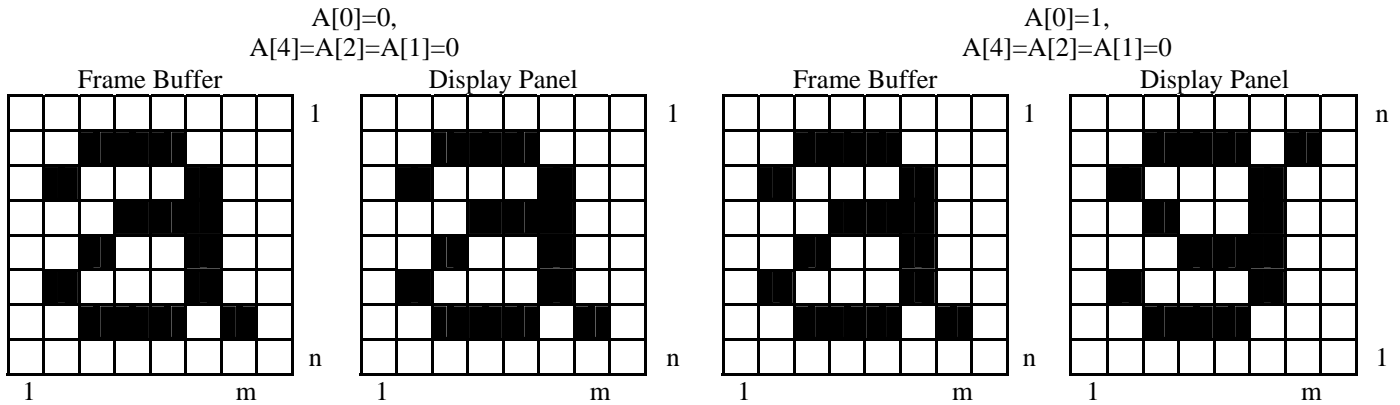


A[0] : Flip Vertical (POR = 0)

This bit flips the image shown on the display panel top to bottom. No change is made to the frame buffer.

0 Normal
1 Flipped

Figure 9-16: A[0] Flip Vertical



9.25 set_scroll_start

Command 0x37
Parameters 2

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|-----|
| Command | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 37 |
| Parameter 1 | 1 | VSP ₁₅ | VSP ₁₄ | VSP ₁₃ | VSP ₁₂ | VSP ₁₁ | VSP ₁₀ | VSP ₉ | VSP ₈ | xx |
| Parameter 2 | 1 | VSP ₇ | VSP ₆ | VSP ₅ | VSP ₄ | VSP ₃ | VSP ₂ | VSP ₁ | VSP ₀ | xx |

Description

This command sets the start of the vertical scrolling area in the frame buffer. The vertical scrolling area is fully defined when this command is used with the set_scroll_area (0x33).

VSP[15:8] : High byte of the line number in frame buffer that is written to the display as the first line of the vertical scrolling area (POR = 00000000)

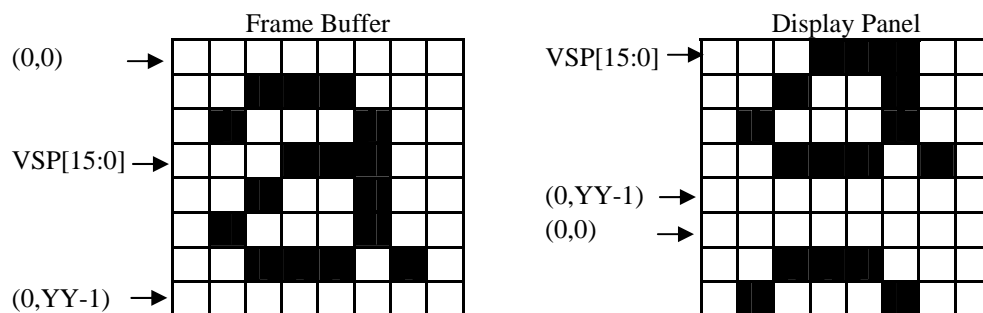
VSP[7:0] : Low byte of the line number in frame buffer that is written to the display as the first line of the vertical scrolling area (POR = 00000000)

If set_address_mode (0x36) A[4] = 0:

Example:

When Top Fixed Area = Bottom Fixed Area = 0, Vertical Scrolling Area = YY and VSP = 3.

Figure 9-17: Set Scroll Start with set_address_mode (0x36) A[4] = 0

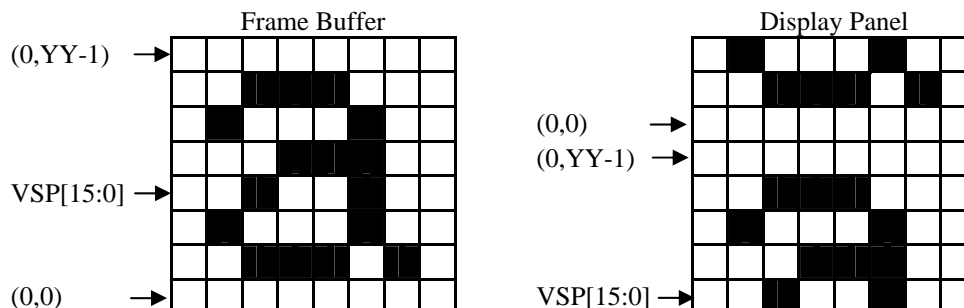


If set_address_mode (0x36) A[4] = 1:

Example:

When Top Fixed Area = Bottom Fixed Area = 0, Vertical Scrolling Area = YY and VSP = 3.

Figure 9-18: Set Scroll Start with set_address_mode (0x36) A[4] = 1



Note :

If set_address_mode, (0x36) A[4] = 0, TFA[15:0] - 1 < VSP[15:0] < # of lines in frame buffer - BFA[15:0]

If set_address_mode, (0x36) A[4] = 1, BFA[15:0] - 1 < VSP[15:0] < # of lines in frame buffer - TFA[15:0]

9.26 exit_idle_mode

Command 0x38

Parameters None

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|---------|-----|----|----|----|----|----|----|----|----|-----|
| Command | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 38 |

Description

This command causes the SSD1963 to exit Idle Mode.

Full color depth is used for the display panel.

9.27 enter_idle_mode

Command 0x39

Parameters None

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|---------|-----|----|----|----|----|----|----|----|----|-----|
| Command | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 39 |

Description

This command causes the SSD1963 to enter Idle Mode.

In Idle Mode, color depth is reduced. Colors are shown on the display panel using the MSB of each of the R, G and B color components in the frame buffer.

Table 9-1 Enter Idle Mode memory content vs display color

| Color | R ₇ R ₆ R ₅ R ₄ R ₃ R ₂ R ₁ R ₀ | G ₇ G ₆ G ₅ G ₄ G ₃ G ₂ G ₁ G ₀ | B ₇ B ₆ B ₅ B ₄ B ₃ B ₂ B ₁ B ₀ |
|---------|---|---|---|
| Black | 0XXXXXXXX | 0XXXXXXXX | 0XXXXXXXX |
| Blue | 0XXXXXXXX | 0XXXXXXXX | 1XXXXXXXX |
| Red | 1XXXXXXXX | 0XXXXXXXX | 0XXXXXXXX |
| Magenta | 1XXXXXXXX | 0XXXXXXXX | 1XXXXXXXX |
| Green | 0XXXXXXXX | 1XXXXXXXX | 0XXXXXXXX |
| Cyan | 0XXXXXXXX | 1XXXXXXXX | 1XXXXXXXX |
| Yellow | 1XXXXXXXX | 1XXXXXXXX | 0XXXXXXXX |
| White | 1XXXXXXXX | 1XXXXXXXX | 1XXXXXXXX |

9.28 write_memory_continue

Command 0x3C
Parameters None

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|---------|-----|----|----|----|----|----|----|----|----|-----|
| Command | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 3C |

Description

Transfer image information from the host processor interface to the SSD1963 from the last write_memory_continue (0x3C) or write_memory_start (0x2C).

If set_address_mode (0x36) A[5] = 0:

Data is written continuing from the pixel location after the write range of the previous write_memory_start (0x2C) or write_memory_continue (0x3C). The column address is then incremented and pixels are written to the frame buffer until the column address equals the End Column (EC) value. The column address is then reset to SC and the page address is incremented. Pixels are written to the frame buffer until the page address equals the End Page (EP) value and the column address equals the EC value, or the host processor sends another command. If the number of pixels exceeds (EC – SC + 1) * (EP – SP + 1) the extra pixels are ignored.

If set_address_mode (0x36) A[5] = 1:

Data is written continuing from the pixel location after the write range of the previous write_memory_start (0x2C) or write_memory_continue (0x3C). The page address is then incremented and pixels are written to the frame buffer until the page address equals the End Page (EP) value. The page address is then reset to SP and the column address is incremented. Pixels are written to the frame buffer until the column register equals the End column (EC) value and the page address equals the EP value, or the host processor sends another command. If the number of pixels exceeds (EC – SC + 1) * (EP – SP + 1) the extra pixels are ignored.

9.29 read_memory_continue

Command 0x3E
Parameters None

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|---------|------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|------------|
| Command | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 3E |

Description

Read image data from the SSD1963 to host processor continuing after the last read_memory_continue (0x3E) or read_memory_start (0x2E).

If set_address_mode (0x36) A[5] = 0:

Pixels are read continuing from the pixel location after the read range of the previous read_memory_start (0x2E) or read_memory_continue (0x3E). The column address is then incremented and pixels are read from the frame buffer until the column address equals the End Column (EC) value. The column address is then reset to SC and the page address is incremented. Pixels are read from the frame buffer until the page address equals the End Page (EP) value and the column address equals the EC value, or the host processor sends another command.

If set_address_mode (0x36) A[5] = 1:

Pixels are read continuing from the pixel location after the read range of the previous read_memory_start (0x2E) or read_memory_continue (0x3E). The page address is then incremented and pixels are read from the frame buffer until the page address equals the End Page (EP) value. The page address is then reset to SP and the column address is incremented. Pixels are read from the frame buffer until the column address equals the End Column (EC) value and the page address equals the EP value, or the host processor sends another command.

9.30 set_tear_scanline

Command 0x44
Parameters 2

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|----------------|----------------|-----|
| Command | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 44 |
| Parameter 1 | 1 | N ₁₅ | N ₁₄ | N ₁₃ | N ₁₂ | N ₁₁ | N ₁₀ | N ₉ | N ₈ | xx |
| Parameter 2 | 1 | N ₇ | N ₆ | N ₅ | N ₄ | N ₃ | N ₂ | N ₁ | N ₀ | xx |

Description

TE signal is sent from the SSD1963 to the host processor when the display panel refresh reaches the provided scanline, N.

N[15:8] : High byte of the scanline (POR = 00000000)

N[7:0] : Low byte of the scanline (POR = 00000000)

Note :

Valid setting for TE signal: 0x0000, 0x0002 to 0xFFFF.

The number of Tear Scanline = N[15:0] +1, except N = 0.

Set Tear Scanline with N = 0 is equivalent to set_tear_on (0x35) A[0] = 0.

When Tear Scanline, N >= Vertical panel size, TE signal will always pull high.

Program set_tear_scanline will automatic change the operating mode of set_tear_on (0x35) A[0] = 1.

This command takes affect on the frame following the current frame. Therefore, if the Tear Effect (TE) signal is already ON, the TE output shall continue to operate as programmed by the previous set_tear_on (0x35) or set_tear_scanline (0x44) until the end of the frame.

9.31 get_scanline

Command 0x45
Parameters 2

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|----------------|----------------|-----|
| Command | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 45 |
| Parameter 1 | 1 | N ₁₅ | N ₁₄ | N ₁₃ | N ₁₂ | N ₁₁ | N ₁₀ | N ₉ | N ₈ | xx |
| Parameter 2 | 1 | N ₇ | N ₆ | N ₅ | N ₄ | N ₃ | N ₂ | N ₁ | N ₀ | xx |

Description

Get the current scan line, N.

N[15:8] : High byte of the current scanline (POR = 00000000)

N[7:0] : Low byte of the current scanline (POR = 00000000)

9.32 read_ddb

Command 0xA1
Parameters 5

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-----|
| Command | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | A1 |
| Parameter 1 | 1 | SSL ₁₅ | SSL ₁₄ | SSL ₁₃ | SSL ₁₂ | SSL ₁₁ | SSL ₁₀ | SSL ₉ | SSL ₈ | xx |
| Parameter 2 | 1 | SSL ₇ | SSL ₆ | SSL ₅ | SSL ₄ | SSL ₃ | SSL ₂ | SSL ₁ | SSL ₀ | xx |
| Parameter 3 | 1 | PROD ₇ | PROD ₆ | PROD ₅ | PROD ₄ | PROD ₃ | PROD ₂ | PROD ₁ | PROD ₀ | xx |
| Parameter 4 | 1 | 0 | 0 | 0 | 0 | 0 | REV ₂ | REV ₁ | REV ₀ | xx |
| Parameter 5 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | FF |

Description

Read the DDB (Device Descriptor Block) information of SSD1963.

SSL[15:8] : Supplier ID of Solomon Systech Limited high byte, always 01h (POR = 00000001)

SSL[7:0] : Supplier ID of Solomon Systech Limited low byte, always 57h (POR = 01010111)

PROD[7:0] : Product ID, always 61h (POR = 01100001)

REV[2:0] : Revision code, always 01h (POR = 001)

Exit code, always FFh (POR = 11111111)

9.33 set_lcd_mode

Command 0xB0
Parameters 7

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|------------------|------------------|------------------|------------------|------------------|-------------------|------------------|------------------|-----|
| Command | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | B0 |
| Parameter 1 | 1 | 0 | 0 | A ₅ | A ₄ | A ₃ | A ₂ | A ₁ | A ₀ | xx |
| Parameter 2 | 1 | 0 | B ₆ | B ₅ | 0 | 0 | 0 | 0 | 0 | xx |
| Parameter 3 | 1 | 0 | 0 | 0 | 0 | 0 | HDP ₁₀ | HDP ₉ | HDP ₈ | xx |
| Parameter 4 | 1 | HDP ₇ | HDP ₆ | HDP ₅ | HDP ₄ | HDP ₃ | HDP ₂ | HDP ₁ | HDP ₀ | xx |
| Parameter 5 | 1 | 0 | 0 | 0 | 0 | 0 | VDP ₁₀ | VDP ₉ | VDP ₈ | xx |
| Parameter 6 | 1 | VDP ₇ | VDP ₆ | VDP ₅ | VDP ₄ | VDP ₃ | VDP ₂ | VDP ₁ | VDP ₀ | xx |
| Parameter 7 | 1 | 0 | 0 | G ₅ | G ₄ | G ₃ | G ₂ | G ₁ | G ₀ | xx |

Description

Set the LCD panel mode and resolution

A[5] : TFT panel data width (POR = 0)

0 18-bit

1 24-bit

A[4] : TFT color depth enhancement enable (POR = 0)

0 Disable FRC or dithering

1 Enable FRC or dithering for color depth enhancement

If the panel data width was set to 24-bit, FRC and dithering feature will be disabled automatic regardless the value of this register.

A[3] : TFT FRC enable (POR = 0)

0 TFT dithering enable

1 TFT FRC enable

| A[5] | A[4] | A[3] | TFT FRC | TFT dithering |
|------|------|------|---------|---------------|
| 0 | 0 | X | Disable | Disable |
| 0 | 1 | 0 | Disable | Enable |
| 0 | 1 | 1 | Enable | Disable |
| 1 | X | X | Disable | Disable |

A[2] : LSHIFT polarity (POR = 0)

Set the dot clock pulse polarity.

- 0 Data latch in falling edge
- 1 Data latch in rising edge

A[1] : LLINE polarity (POR = 0)

Set the horizontal sync pulse polarity.

- 0 Active low
- 1 Active high

A[0] : LFRAME polarity (POR = 0)

Set the vertical sync pulse polarity.

- 0 Active low
- 1 Active high

B[6:5] : TFT type (POR = 01)

- 00, 01 TFT mode
- 10 Serial RGB mode
- 11 Serial RGB+dummy mode

HDP [10:8] : High byte of the horizontal panel size (POR = 010)

HDP [7:0] : Low byte of the horizontal panel size (POR = 01111111)

Horizontal panel size = (HDP + 1) pixels

VDP [10:8] : High byte of the vertical panel size (POR = 001)

VDP [7:0] : Low byte of the vertical panel size (POR = 11011111)

Vertical panel size = (VDP + 1) lines

G[5:3] : Even line RGB sequence for serial TFT interface (POR = 000)

- 000 RGB
- 001 RBG
- 010 GRB
- 011 GBR
- 100 BRG
- 101 BGR
- 11x Reserved

G[2:0] : Odd line RGB sequence for serial TFT interface (POR = 000)

- 000 RGB
- 001 RBG
- 010 GRB
- 011 GBR
- 100 BRG
- 101 BGR
- 11x Reserved

9.34 get_lcd_mode

Command 0xB1

Parameters 7

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|------------------|------------------|------------------|------------------|------------------|-------------------|------------------|------------------|-----|
| Command | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | B1 |
| Parameter 1 | 1 | 0 | 0 | A ₅ | A ₄ | A ₃ | A ₂ | A ₁ | A ₀ | xx |
| Parameter 2 | 1 | 0 | B ₆ | B ₅ | 0 | 0 | 0 | 0 | 0 | xx |
| Parameter 3 | 1 | 0 | 0 | 0 | 0 | 0 | HDP ₁₀ | HDP ₉ | HDP ₈ | xx |
| Parameter 4 | 1 | HDP ₇ | HDP ₆ | HDP ₅ | HDP ₄ | HDP ₃ | HDP ₂ | HDP ₁ | HDP ₀ | xx |
| Parameter 5 | 1 | 0 | 0 | 0 | 0 | 0 | VDP ₁₀ | VDP ₉ | VDP ₈ | xx |
| Parameter 6 | 1 | VDP ₇ | VDP ₆ | VDP ₅ | VDP ₄ | VDP ₃ | VDP ₂ | VDP ₁ | VDP ₀ | xx |
| Parameter 7 | 1 | 0 | 0 | G ₅ | G ₄ | G ₃ | G ₂ | G ₁ | G ₀ | xx |

Description

Get the current LCD panel mode and resolution

A[5] : TFT panel data width(POR = 0)

0 18-bit
1 24-bit

A[4] : TFT color depth enhancement enable(POR = 0)

0 Disable FRC or dithering
1 Enable FRC or dithering for color depth enhancement

If the panel data width was set to 24-bit, FRC and dithering feature will be disabled automatic regardless the value of this register.

A[3] : TFT FRC enable (POR = 0)

0 TFT dithering enable
1 TFT FRC enable

A[2] : LSHIFT polarity (POR = 0)

The dot clock pulse polarity.

0 Data latch in falling edge
1 Data latch in rising edge

A[1] : LLINE polarity (POR = 0)

The horizontal sync pulse polarity.

0 Active low
1 Active high

A[0] : LFRAME polarity (POR = 0)

The vertical sync pulse polarity.

0 Active low
1 Active high

B[6:5] : TFT type(POR = 01)

00, 01 TFT mode
10 Serial RGB mode
11 Serial RGB+dummy mode

HDP[10:8] : High byte of the horizontal panel size (POR = 010)

HDP[7:0] : Low byte of the horizontal panel size (POR = 01111111)

VDP[10:8] : High byte of the vertical panel size (POR = 001)

VDP[7:0] : Low byte of the vertical panel size (POR = 11011111)

G[5:3] : Even line RGB sequence (POR = 000)

| | |
|-----|----------|
| 000 | RGB |
| 001 | RBG |
| 010 | GRB |
| 011 | GBR |
| 100 | BRG |
| 101 | BGR |
| 11x | Reserved |

G[2:0] : Odd line RGB sequence (POR = 000)

| | |
|-----|----------|
| 000 | RGB |
| 001 | RBG |
| 010 | GRB |
| 011 | GBR |
| 100 | BRG |
| 101 | BGR |
| 11x | Reserved |

9.35 set_hori_period

Command 0xB4

Parameters 8

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|------------------|------------------|------------------|------------------|------------------|-------------------|-------------------|-------------------|-----|
| Command | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | B4 |
| Parameter 1 | 1 | 0 | 0 | 0 | 0 | 0 | HT ₁₀ | HT ₉ | HT ₈ | xx |
| Parameter 2 | 1 | HT ₇ | HT ₆ | HT ₅ | HT ₄ | HT ₃ | HT ₂ | HT ₁ | HT ₀ | xx |
| Parameter 3 | 1 | 0 | 0 | 0 | 0 | 0 | HPS ₁₀ | HPS ₉ | HPS ₈ | xx |
| Parameter 4 | 1 | HPS ₇ | HPS ₆ | HPS ₅ | HPS ₄ | HPS ₃ | HPS ₂ | HPS ₁ | HPS ₀ | xx |
| Parameter 5 | 1 | 0 | HPW ₆ | HPW ₅ | HPW ₄ | HPW ₃ | HPW ₂ | HPW ₁ | HPW ₀ | xx |
| Parameter 6 | 1 | 0 | 0 | 0 | 0 | 0 | LPS ₁₀ | LPS ₉ | LPS ₈ | xx |
| Parameter 7 | 1 | LPS ₇ | LPS ₆ | LPS ₅ | LPS ₄ | LPS ₃ | LPS ₂ | LPS ₁ | LPS ₀ | xx |
| Parameter 8 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | LPSP ₁ | LPSP ₀ | xx |

Description

Set front porch and back porch

HT[10:8] : High byte of horizontal total period (display + non-display) in pixel clock (POR = 010)

HT[7:0] : Low byte of the horizontal total period (display + non-display) in pixel clock (POR = 10101111)
Horizontal total period = (HT + 1) pixels

HPS[10:8] : High byte of the non-display period between the start of the horizontal sync (LLINE) signal and the first display data. (POR = 000)

HPS[7:0] : Low byte of the non-display period between the start of the horizontal sync (LLINE) signal and the first display data. (POR = 00100000)

For TFT : Horizontal Sync Pulse Start Position = HPS pixels

For Serial TFT : Horizontal Sync Pulse Start Position = HPS pixels + LPSP subpixels

HPW[6:0] : Set the horizontal sync pulse width (LLINE) in pixel clock. (POR = 0000111)

Horizontal Sync Pulse Width = (HPW + 1) pixels

LPS[10:8] : Set the horizontal sync pulse (LLINE) start location in pixel clock. (POR = 000)
 LPS[7:0] : Set the horizontal sync pulse width (LLINE) in start. (POR = 00000000)
 Horizontal Display Period Start Position = LPS pixels

LPSPP[1:0] : Set the horizontal sync pulse subpixel start position for serial TFT interface (POR = 00)

Timing refer to Figure 13-5.

9.36 get_hori_period

Command 0xB5
Parameters 8

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|------------------|------------------|------------------|------------------|------------------|-------------------|--------------------|--------------------|-----|
| Command | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | B5 |
| Parameter 1 | 1 | 0 | 0 | 0 | 0 | 0 | HT ₁₀ | HT ₉ | HT ₈ | xx |
| Parameter 2 | 1 | HT ₇ | HT ₆ | HT ₅ | HT ₄ | HT ₃ | HT ₂ | HT ₁ | HT ₀ | xx |
| Parameter 3 | 1 | 0 | 0 | 0 | 0 | 0 | HPS ₁₀ | HPS ₉ | HPS ₈ | xx |
| Parameter 4 | 1 | HPS ₇ | HPS ₆ | HPS ₅ | HPS ₄ | HPS ₃ | HPS ₂ | HPS ₁ | HPS ₀ | xx |
| Parameter 5 | 1 | 0 | HPW ₆ | HPW ₅ | HPW ₄ | HPW ₃ | HPW ₂ | HPW ₁ | HPW ₀ | xx |
| Parameter 6 | 1 | 0 | 0 | 0 | 0 | 0 | LPS ₁₀ | LPS ₉ | LPS ₈ | xx |
| Parameter 7 | 1 | LPS ₇ | LPS ₆ | LPS ₅ | LPS ₄ | LPS ₃ | LPS ₂ | LPS ₁ | LPS ₀ | xx |
| Parameter 8 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | LPSPP ₁ | LPSPP ₀ | xx |

Description

Get current front porch and back porch settings

HT[10:8] : High byte of the horizontal total period (display + non-display) in pixel clock (POR = 010)
 HT[7:0] : Low byte of the horizontal total period (display + non-display) in pixel clock (POR = 10101111)

HPS[10:8] : High byte of the non-display period between the start of the horizontal sync (LLINE) signal and the first display data. (POR = 000)
 HPS[7:0] : Low byte of the non-display period between the start of the horizontal sync (LLINE) signal and the first display data. (POR = 00100000)

HPW[6:0] : The horizontal sync pulse width (LLINE) in pixel clock. (POR = 0000111)

LPS[10:8] : High byte of the horizontal sync pulse (LLINE) start location in pixel clock. (POR = 000)
 LPS[7:0] : Low byte of the horizontal sync pulse width (LLINE) in start. (POR = 00000000)

LPSPP[1:0] : The horizontal sync pulse subpixel start position (POR = 00)

9.37 set_vert_period

Command 0xB6
Parameters 7

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|------------------|------------------|------------------|------------------|------------------|-------------------|------------------|------------------|-----|
| Command | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | B6 |
| Parameter 1 | 1 | 0 | 0 | 0 | 0 | 0 | VT ₁₀ | VT ₉ | VT ₈ | xx |
| Parameter 2 | 1 | VT ₇ | VT ₆ | VT ₅ | VT ₄ | VT ₃ | VT ₂ | VT ₁ | VT ₀ | xx |
| Parameter 3 | 1 | 0 | 0 | 0 | 0 | 0 | VPS ₁₀ | VPS ₉ | VPS ₈ | xx |
| Parameter 4 | 1 | VPS ₇ | VPS ₆ | VPS ₅ | VPS ₄ | VPS ₃ | VPS ₂ | VPS ₁ | VPS ₀ | xx |
| Parameter 5 | 1 | 0 | VPW ₆ | VPW ₅ | VPW ₄ | VPW ₃ | VPW ₂ | VPW ₁ | VPW ₀ | xx |
| Parameter 6 | 1 | 0 | 0 | 0 | 0 | 0 | FPS ₁₀ | FPS ₉ | FPS ₈ | xx |
| Parameter 7 | 1 | FPS ₇ | FPS ₆ | FPS ₅ | FPS ₄ | FPS ₃ | FPS ₂ | FPS ₁ | FPS ₀ | xx |

Description

Set the vertical blanking interval between last scan line and next LFRAME pulse

VT[10:8] : High byte of the vertical total (display + non-display) period in lines (POR = 001)

VT[7:0] : Low byte of the vertical total (display + non-display) period in lines (POR = 11101111)
 Vertical Total = (VT + 1) lines

VPS[10:8] : High byte the non-display period in lines between the start of the frame and the first display data in line. (POR = 000)

VPS[7:0] : The non-display period in lines between the start of the frame and the first display data in line. (POR = 00000100)
 Vertical Sync Pulse Start Position = VPS lines

VPW[6:0] : Set the vertical sync pulse width (LFRAME) in lines. (POR = 000001)
 Vertical Sync Pulse Width = (VPW + 1) lines

FPS[10:8] : High byte of the vertical sync pulse (LFRAME) start location in lines. (POR = 000)

FPS[7:0] : Low byte of the vertical sync pulse (LFRAME) start location in lines. (POR = 00000000)
 Vertical Display Period Start Position = FPS lines

Timing refer to Figure 13-5.

9.38 get_vert_period

Command 0xB7
Parameters 7

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|------------------|------------------|------------------|------------------|------------------|-------------------|------------------|------------------|-----|
| Command | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | B7 |
| Parameter 1 | 1 | 0 | 0 | 0 | 0 | 0 | VT ₁₀ | VT ₉ | VT ₈ | xx |
| Parameter 2 | 1 | VT ₇ | VT ₆ | VT ₅ | VT ₄ | VT ₃ | VT ₂ | VT ₁ | VT ₀ | xx |
| Parameter 3 | 1 | 0 | 0 | 0 | 0 | 0 | VPS ₁₀ | VPS ₉ | VPS ₈ | xx |
| Parameter 4 | 1 | VPS ₇ | VPS ₆ | VPS ₅ | VPS ₄ | VPS ₃ | VPS ₂ | VPS ₁ | VPS ₀ | xx |
| Parameter 5 | 1 | 0 | VPW ₆ | VPW ₅ | VPW ₄ | VPW ₃ | VPW ₂ | VPW ₁ | VPW ₀ | xx |
| Parameter 6 | 1 | 0 | 0 | 0 | 0 | 0 | FPS ₁₀ | FPS ₉ | FPS ₈ | xx |
| Parameter 7 | 1 | FPS ₇ | FPS ₆ | FPS ₅ | FPS ₄ | FPS ₃ | FPS ₂ | FPS ₁ | FPS ₀ | xx |

Description

Get the vertical blanking interval between last scan line and next LFRAME pulse

VT[10:8] : High byte of the vertical total (display + non-display) period in lines (POR = 001)
 VT[7:0] : Low byte of the vertical total (display + non-display) period in lines (POR = 01111111)
 VPS[10:8] : High byte of the non-display period in lines between the start of the frame and the first display data in line. (POR = 000)
 VPS[7:0] : Low byte of the non-display period in lines between the start of the frame and the first display data in line. (POR = 00000100)
 VPW[6:0] : The vertical sync pulse width (LFRAME) in lines. (POR = 000001)
 FPS[10:8] : High byte of the vertical sync pulse (LFRAME) start location in lines. (POR = 000)
 FPS[7:0] : Low byte of the vertical sync pulse (LFRAME) start location in lines. (POR = 00000000)

9.39 set_gpio_conf

Command 0xB8

Parameters 2

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|-----|
| Command | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | B8 |
| Parameter 1 | 1 | A ₇ | A ₆ | A ₅ | A ₄ | A ₃ | A ₂ | A ₁ | A ₀ | xx |
| Parameter 2 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | B ₀ | xx |

Description

Set the GPIOs configuration. If the GPIOs are not used for LCD, set the direction. Otherwise, they are toggled with LCD signals by 0xC0 – 0xCF.

A[7] : GPIO3 configuration (POR = 0)

0 GPIO3 is controlled by host
 1 GPIO3 is controlled by LCDC

A[6] : GPIO2 configuration (POR = 0)

0 GPIO2 is controlled by host
 1 GPIO2 is controlled by LCDC

A[5] : GPIO1 configuration (POR = 0)

0 GPIO1 is controlled by host
 1 GPIO1 is controlled by LCDC

A[4] : GPIO0 configuration (POR = 0)

0 GPIO0 is controlled by host
 1 GPIO0 is controlled by LCDC

A[3] : GPIO3 direction (POR = 0)

0 GPIO3 is input
 1 GPIO3 is output

A[2] : GPIO2 direction (POR = 0)

0 GPIO2 is input
 1 GPIO2 is output

A[1] : GPIO1 direction (POR = 0)

0 GPIO1 is input
 1 GPIO1 is output

A[0] : GPIO0 direction (POR = 0)

0 GPIO0 is input
 1 GPIO0 is output

B[0] : GPIO0 direction (POR = 0)

- 0 GPIO0 is used to control the panel power with enter_sleep_mode (0x10) or exit_sleep_mode (0x11).
- 1 GPIO0 is used as normal GPIO

9.40 get_gpio_conf

Command 0xB9
Parameters 2

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|-----|
| Command | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | B9 |
| Parameter 1 | 1 | A ₇ | A ₆ | A ₅ | A ₄ | A ₃ | A ₂ | A ₁ | A ₀ | xx |
| Parameter 2 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | B ₀ | xx |

Description

Get the current GPIOs configuration

A[7] : GPIO3 configuration (POR = 0)

- 0 GPIO3 is controlled by host
- 1 GPIO3 is controlled by LCDC

A[6] : GPIO2 configuration (POR = 0)

- 0 GPIO2 is controlled by host
- 1 GPIO2 is controlled by LCDC

A[5] : GPIO1 configuration (POR = 0)

- 0 GPIO1 is controlled by host
- 1 GPIO1 is controlled by LCDC

A[4] : GPIO0 configuration (POR = 0)

- 0 GPIO0 is controlled by host
- 1 GPIO0 is controlled by LCDC

A[3] : GPIO3 direction (POR = 0)

- 0 GPIO3 is input
- 1 GPIO3 is output

A[2] : GPIO2 direction (POR = 0)

- 0 GPIO2 is input
- 1 GPIO2 is output

A[1] : GPIO1 direction (POR = 0)

- 0 GPIO1 is input
- 1 GPIO1 is output

A[0] : GPIO0 direction (POR = 0)

- 0 GPIO0 is input
- 1 GPIO0 is output

B[0] : GPIO0 direction (POR = 0)

- 0 GPIO0 is used to control the panel power with enter_sleep_mode (0x10) or exit_sleep_mode (0x11)
- 1 GPIO0 is used as normal GPIO

9.41 set_gpio_value

Command 0xBA

Parameters 1

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----|----|----|----|----------------|----------------|----------------|----------------|-----|
| Command | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | BA |
| Parameter 1 | 1 | 0 | 0 | 0 | 0 | A ₃ | A ₂ | A ₁ | A ₀ | xx |

Description

Set GPIO value for GPIO configured as output

A[3] : GPIO3 value (POR = 0)

0 GPIO3 outputs 0

1 GPIO3 outputs 1

A[2] : GPIO2 value (POR = 0)

0 GPIO2 outputs 0

1 GPIO2 outputs 1

A[1] : GPIO1 value (POR = 0)

0 GPIO1 outputs 0

1 GPIO1 outputs 1

A[0] : GPIO0 value (POR = 0)

0 GPIO0 outputs 0

1 GPIO0 outputs 1

9.42 get_gpio_status

Command 0xBB

Parameters 1

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----|----|----|----|----------------|----------------|----------------|----------------|-----|
| Command | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | BB |
| Parameter 1 | 1 | 0 | 0 | 0 | 0 | A ₃ | A ₂ | A ₁ | A ₀ | xx |

Description

Read current GPIO status. If the individual GPIO was configured as input, the value is the status of the corresponding pin. Otherwise, it is the programmed value.

A[3] : GPIO3 value (POR : depends on pad value)

0 GPIO3 is pulled low

1 GPIO3 is pulled high

A[2] : GPIO2 value (POR : depends on pad value)

0 GPIO2 is pulled low

1 GPIO2 is pulled high

A[1] : GPIO1 value (POR : depends on pad value)

0 GPIO1 is pulled low

1 GPIO1 is pulled high

A[0] : GPIO0 value (POR : depends on pad value)

0 GPIO0 is pulled low

1 GPIO0 is pulled high

9.43 set_post_proc

Command 0xBC
Parameters 4

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|-----|
| Command | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | BC |
| Parameter 1 | 1 | A ₇ | A ₆ | A ₅ | A ₄ | A ₃ | A ₂ | A ₁ | A ₀ | xx |
| Parameter 2 | 1 | B ₇ | B ₆ | B ₅ | B ₄ | B ₃ | B ₂ | B ₁ | B ₀ | xx |
| Parameter 3 | 1 | C ₇ | C ₆ | C ₅ | C ₄ | C ₃ | C ₂ | C ₁ | C ₀ | xx |
| Parameter 4 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | D ₀ | xx |

Description

Set the image post processor

A[7:0] : Set the contrast value (POR = 01000000)

B[7:0] : Set the brightness value (POR = 10000000)

C[7:0] : Set the saturation value (POR = 01000000)

D[0] : Post Processor Enable (POR = 0)

0 Disable the postprocessor
1 Enable the postprocessor

9.44 get_post_proc

Command 0xBD
Parameters 4

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|-----|
| Command | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | BD |
| Parameter 1 | 1 | A ₇ | A ₆ | A ₅ | A ₄ | A ₃ | A ₂ | A ₁ | A ₀ | xx |
| Parameter 2 | 1 | B ₇ | B ₆ | B ₅ | B ₄ | B ₃ | B ₂ | B ₁ | B ₀ | xx |
| Parameter 3 | 1 | C ₇ | C ₆ | C ₅ | C ₄ | C ₃ | C ₂ | C ₁ | C ₀ | xx |
| Parameter 4 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | D ₀ | xx |

Description

Get the image post processor

A[7:0] : Get the contrast value (POR = 01000000)

B[7:0] : Get the brightness value (POR = 10000000)

C[7:0] : Get the saturation value (POR = 01000000)

D[0] : Post Processor Enable (POR = 0)

0 Disable the postprocessor
1 Enable the postprocessor

9.45 set_pwm_conf

Command 0xBE

Parameters 6

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-----|
| Command | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | BE |
| Parameter 1 | 1 | PWMF ₇ | PWMF ₆ | PWMF ₅ | PWMF ₄ | PWMF ₃ | PWMF ₂ | PWMF ₁ | PWMF ₀ | xx |
| Parameter 2 | 1 | PWM ₇ | PWM ₆ | PWM ₅ | PWM ₄ | PWM ₃ | PWM ₂ | PWM ₁ | PWM ₀ | xx |
| Parameter 3 | 1 | 0 | 0 | 0 | 0 | C ₃ | 0 | 0 | C ₀ | xx |
| Parameter 4 | 1 | D ₇ | D ₆ | D ₅ | D ₄ | D ₃ | D ₂ | D ₁ | D ₀ | xx |
| Parameter 5 | 1 | E ₇ | E ₆ | E ₅ | E ₄ | E ₃ | E ₂ | E ₁ | E ₀ | xx |
| Parameter 6 | 1 | 0 | 0 | 0 | 0 | F ₃ | F ₂ | F ₁ | F ₀ | xx |

Description

Set the PWM configuration

PWMF[7:0] : Set the PWM frequency in system clock (POR = 00000000)

PWM signal frequency = PLL clock / (256 * (PWMF[7:0] + 1)) / 256

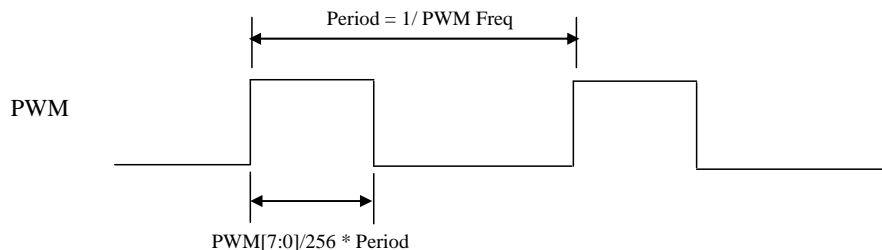
PWM[7:0] : Set the PWM duty cycle (POR = 00000000)

PWM duty cycle = PWM[7:0] / 256 for DBC disable (0xD0) A0 = 0

If DBC enable (0xD0) A0 = 1, these parameter will be ignored

Note : PWM always 0 if PWM[7:0] = 00h

Figure 9-19: PWM signal



C[3] : PWM configuration (POR = 0)

0 PWM controlled by host

1 PWM controlled by DBC

C[0] : PWM enable (POR = 0)

0 PWM disable

1 PWM enable

D[7:0] : DBC manual brightness (POR = 00000000)

Set the manual brightness level. When Manual Brightness Mode (0xD0) A[6] is enabled, the final DBC duty cycle output will be multiplied by this value / 255.

PWM duty cycle = DBC output * D[7:0] / 255

00 Dimmest

FF brightest

E[7:0] : DBC minimum brightness (POR = 00000000)

Set the minimum brightness level. When Manual Brightness Mode (0xD0) A[6] is enabled, DBC duty cycle output will be limited by this value. This will prevent from backlight being too dark or off.

00 Dimmest

FF Brightest

F[3:0] : Brightness prescaler (POR = 0000)

Set the brightness prescaler to control how gradually the manual brightness is changed between different levels. There is a filter will undergo a number of iterations before the manual brightness saturated. This parameter is valid when

Transition Effect enable (0xD0) A5 = 1

The iteration ration = system frequency / Divcode / 32768

| F[3:0] | Divcode |
|--------|---------|
| 0000 | off |
| 0001 | 1 |
| 0010 | 2 |
| 0011 | 3 |
| 0100 | 4 |
| 0101 | 6 |
| 0110 | 8 |
| 0111 | 12 |
| 1000 | 16 |
| 1001 | 24 |
| 1010 | 32 |
| 1011 | 48 |
| 1100 | 64 |
| 1101 | 96 |
| 1110 | 128 |
| 1111 | 192 |

9.46 get_pwm_conf

Command 0xBF

Parameters 7

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-----|
| Command | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | BF |
| Parameter 1 | 1 | PWMF ₇ | PWMF ₆ | PWMF ₅ | PWMF ₄ | PWMF ₃ | PWMF ₂ | PWMF ₁ | PWMF ₀ | xx |
| Parameter 2 | 1 | PWM ₇ | PWM ₆ | PWM ₅ | PWM ₄ | PWM ₃ | PWM ₂ | PWM ₁ | PWM ₀ | xx |
| Parameter 3 | 1 | 0 | 0 | 0 | 0 | C ₃ | 0 | 0 | C ₀ | xx |
| Parameter 4 | 1 | D ₇ | D ₆ | D ₅ | D ₄ | D ₃ | D ₂ | D ₁ | D ₀ | xx |
| Parameter 5 | 1 | E ₇ | E ₆ | E ₅ | E ₄ | E ₃ | E ₂ | E ₁ | E ₀ | xx |
| Parameter 6 | 1 | 0 | 0 | 0 | 0 | F ₃ | F ₂ | F ₁ | F ₀ | xx |
| Parameter 7 | 1 | G ₇ | G ₆ | G ₅ | G ₄ | G ₃ | G ₂ | G ₁ | G ₀ | xx |

Description

Get the PWM configuration

PWMF[7:0] : Get the PWM frequency in system clock (POR = 00000000)

PWM[7:0] : Get the PWM duty cycle (POR = 00000000)

C[3] : PWM configuration (POR = 0)

| | |
|---|------------------------|
| 0 | PWM controlled by host |
| 1 | PWM controlled by DBC |

C[0] : PWM enable (POR = 0)

| | |
|---|-------------|
| 0 | PWM disable |
| 1 | PWM enable |

D[7:0] : DBC manual brightness (POR = 00000000)

Get the brightness level

| | |
|----|-----------|
| 00 | Dimmest |
| FF | brightest |

E[7:0] : DBC minimum brightness (POR = 00000000)

Get the minimum brightness level.

| | |
|----|-----------|
| 00 | Dimmest |
| FF | Brightest |

F[3:0] : Brightness prescaler (POR = 0000)

Get the brightness prescaler

G[7:0] : Dynamic backlight duty cycle : Get the current PWM duty cycle controlled by PWM (POR = 00000000)

9.47 set_lcd_gen0

Command 0xC0

Parameters 7

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|------------------|------------------|------------------|------------------|------------------|-------------------|------------------|------------------|-----|
| Command | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | C0 |
| Parameter 1 | 1 | A ₇ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | xx |
| Parameter 2 | 1 | 0 | 0 | 0 | 0 | 0 | GF0 ₁₀ | GF0 ₉ | GF0 ₈ | xx |
| Parameter 3 | 1 | GF0 ₇ | GF0 ₆ | GF0 ₅ | GF0 ₄ | GF0 ₃ | GF0 ₂ | GF0 ₁ | GF0 ₀ | xx |
| Parameter 4 | 1 | 0 | 0 | 0 | 0 | 0 | GR0 ₁₀ | GR0 ₉ | GR0 ₈ | xx |
| Parameter 5 | 1 | GR0 ₇ | GR0 ₆ | GR0 ₅ | GR0 ₄ | GR0 ₃ | GR0 ₂ | GR0 ₁ | GR0 ₀ | xx |
| Parameter 6 | 1 | F ₇ | F ₆ | F ₅ | F ₄ | F ₃ | GP0 ₁₀ | GP0 ₉ | GP0 ₈ | xx |
| Parameter 7 | 1 | GP0 ₇ | GP0 ₆ | GP0 ₅ | GP0 ₄ | GP0 ₃ | GP0 ₂ | GP0 ₁ | GP0 ₀ | xx |

Description

Set the rise, fall, period and toggling properties of LCD signal generator 0

A[7] : Reset LCD generator 0 at every frame start

0 The generator 0 will not reset in the starting point of a frame

1 The generator 0 will reset in the starting point of a frame

GF0[10:8] : The highest 3 bits of the generator 0 falling position (POR = 000)

GF0[7:0] : The lower byte of the generator 0 falling position (POR = 00000001)

GR0[10:8] : The highest 3 bits of the generator 0 rising position (POR = 000)

GR0[7:0] : The lower byte of the generator 0 rising position (POR = 00000000)

F[7] : Force the generator 0 output to 0 in non-display period

0 generator 0 is normal

1 generator 0 output is forced to 0 in non-display period

F[6:5] : Force the generator 0 output to 0 in odd or even lines

00 generator 0 is normal in both odd and even lines

01 generator 0 output is force to 0 in odd lines

10 generator 0 output is force to 0 in even lines

11 generator 0 is normal in both odd and even line

F[4:3] : Generator 0 toggle mode

00 Disable

01 Toggle by pixel clock (LSHIFT)

10 Toggle by Line (LLINE)

11 Toggle by Frame (LFRAME)

GP0[10:8] : The highest 3 bits of the generator 0 period (POR = 100)

GP0[7:0] : The lower byte of the generator 0 period (POR = 00000000)

9.48 get_lcd_gen0

Command 0xC1

Parameters 7

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|------------------|------------------|------------------|------------------|------------------|-------------------|------------------|------------------|-----|
| Command | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | C1 |
| Parameter 1 | 1 | A ₇ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | xx |
| Parameter 2 | 1 | 0 | 0 | 0 | 0 | 0 | GF0 ₁₀ | GF0 ₉ | GF0 ₈ | xx |
| Parameter 3 | 1 | GF0 ₇ | GF0 ₆ | GF0 ₅ | GF0 ₄ | GF0 ₃ | GF0 ₂ | GF0 ₁ | GF0 ₀ | xx |
| Parameter 4 | 1 | 0 | 0 | 0 | 0 | 0 | GR0 ₁₀ | GR0 ₉ | GR0 ₈ | xx |
| Parameter 5 | 1 | GR0 ₇ | GR0 ₆ | GR0 ₅ | GR0 ₄ | GR0 ₃ | GR0 ₂ | GR0 ₁ | GR0 ₀ | xx |
| Parameter 6 | 1 | F ₇ | F ₆ | F ₅ | F ₄ | F ₃ | GP0 ₁₀ | GP0 ₉ | GP0 ₈ | xx |
| Parameter 7 | 1 | GP0 ₇ | GP0 ₆ | GP0 ₅ | GP0 ₄ | GP0 ₃ | GP0 ₂ | GP0 ₁ | GP0 ₀ | xx |

Description

Get the rise, fall, period and toggling properties of LCD signal generator 0

A[7] : Reset LCD generator 0 at every frame start

0 The generator 0 will not reset in the starting point of a frame

1 The generator 0 will reset in the starting point of a frame

GF0[10:8] : The highest 3 bits of the generator 0 falling position (POR = 000)

GF0[7:0] : The lower byte of the generator 0 falling position (POR = 00000001)

GR0[10:8] : The highest 3 bits of the generator 0 rising position (POR = 000)

GR0[7:0] : The lower byte of the generator 0 rising position (POR = 00000000)

F[7] : Force the generator 0 output to 0 in non-display period

0 generator 0 is normal

1 generator 0 output is forced to 0 in non-display period

F[6:5] : Force the generator 0 output to 0 in odd or even lines

00 generator 0 is normal in both odd and even lines

01 generator 0 output is force to 0 in odd lines

10 generator 0 output is force to 0 in even lines

11 generator 0 is normal in both odd and even line

F[4:3] : Generator 0 toggle mode

00 Disable

01 Toggle by pixel clock (LSHIFT)

10 Toggle by Line (LLINE)

11 Toggle by Frame (LFRAME)

GP0[10:8] : The highest 3 bits of the generator 0 period (POR = 100)

GP0[7:0] : The lower byte of the generator 0 period (POR = 00000000)

9.49 set_lcd_gen1

Command 0xC2

Parameters 7

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|------------------|------------------|------------------|------------------|------------------|-------------------|------------------|------------------|-----|
| Command | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | C2 |
| Parameter 1 | 1 | A ₇ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | xx |
| Parameter 2 | 1 | 0 | 0 | 0 | 0 | 0 | GF1 ₁₀ | GF1 ₉ | GF1 ₈ | xx |
| Parameter 3 | 1 | GF1 ₇ | GF1 ₆ | GF1 ₅ | GF1 ₄ | GF1 ₃ | GF1 ₂ | GF1 ₁ | GF1 ₀ | xx |
| Parameter 4 | 1 | 0 | 0 | 0 | 0 | 0 | GR1 ₁₀ | GR1 ₉ | GR1 ₈ | xx |
| Parameter 5 | 1 | GR1 ₇ | GR1 ₆ | GR1 ₅ | GR1 ₄ | GR1 ₃ | GR1 ₂ | GR1 ₁ | GR1 ₀ | xx |
| Parameter 6 | 1 | F ₇ | F ₆ | F ₅ | F ₄ | F ₃ | GP1 ₁₀ | GP1 ₉ | GP1 ₈ | xx |
| Parameter 7 | 1 | GP1 ₇ | GP1 ₆ | GP1 ₅ | GP1 ₄ | GP1 ₃ | GP1 ₂ | GP1 ₁ | GP1 ₀ | xx |

Description

Set the rise, fall, period and toggling properties of LCD signal generator 1

A[7] : Reset LCD generator 1 at every frame start

0 The generator 1 will not reset in the starting point of a frame

1 The generator 1 will reset in the starting point of a frame

GF1[10:8] : The highest 3 bits of the generator 1 falling position (POR = 000)

GF1[7:0] : The lower byte of the generator 1 falling position (POR = 00000001)

GR1[10:8] : The highest 3 bits of the generator 1 rising position (POR = 000)

GR1[7:0] : The lower byte of the generator 1 rising position (POR = 00000000)

F[7] : Force the generator 1 output to 0 in non-display period

0 generator 1 is normal

1 generator 1 output is forced to 0 in non-display period

F[6:5] : Force the generator 1 output to 0 in odd or even lines

00 generator 1 is normal in both odd and even lines

01 generator 1 output is force to 0 in odd lines

10 generator 1 output is force to 0 in even lines

11 generator 1 is normal in both odd and even line

F[4:3] : Generator 1 toggle mode

00 Disable

01 Toggle by pixel clock (LSHIFT)

10 Toggle by Line (LLINE)

11 Toggle by Frame (LFRAME)

GP1[10:8] : The highest 3 bits of the generator 1 period (POR = 100)

GP1[7:0] : The lower byte of the generator 1 period (POR = 00000000)

9.50 get_lcd_gen1

Command 0xC3

Parameters 7

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|------------------|------------------|------------------|------------------|------------------|-------------------|------------------|------------------|-----|
| Command | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | C3 |
| Parameter 1 | 1 | A ₇ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | xx |
| Parameter 2 | 1 | 0 | 0 | 0 | 0 | 0 | GF1 ₁₀ | GF1 ₉ | GF1 ₈ | xx |
| Parameter 3 | 1 | GF1 ₇ | GF1 ₆ | GF1 ₅ | GF1 ₄ | GF1 ₃ | GF1 ₂ | GF1 ₁ | GF1 ₀ | xx |
| Parameter 4 | 1 | 0 | 0 | 0 | 0 | 0 | GR1 ₁₀ | GR1 ₉ | GR1 ₈ | xx |
| Parameter 5 | 1 | GR1 ₇ | GR1 ₆ | GR1 ₅ | GR1 ₄ | GR1 ₃ | GR1 ₂ | GR1 ₁ | GR1 ₀ | xx |
| Parameter 6 | 1 | F ₇ | F ₆ | F ₅ | F ₄ | F ₃ | GP1 ₁₀ | GP1 ₉ | GP1 ₈ | xx |
| Parameter 7 | 1 | GP1 ₇ | GP1 ₆ | GP1 ₅ | GP1 ₄ | GP1 ₃ | GP1 ₂ | GP1 ₁ | GP1 ₀ | xx |

Description

Get the rise, fall, period and toggling properties of LCD signal generator 1

A[7] : Reset LCD generator 1 at every frame start

- 0 The generator 1 will not reset in the starting point of a frame
- 1 The generator 1 will reset in the starting point of a frame

GF1[10:8] : The highest 3 bits of the generator 1 falling position (POR = 000)

GF1[7:0] : The lower byte of the generator 1 falling position (POR = 00000001)

GR1[10:8] : The highest 3 bits of the generator 1 rising position (POR = 000)

GR1[7:0] : The lower byte of the generator 1 rising position (POR = 00000000)

F[7] : Force the generator 1 output to 0 in non-display period

- 0 generator 1 is normal
- 1 generator 1 output is forced to 0 in non-display period

F[6:5] : Force the generator 1 output to 0 in odd or even lines

- 00 generator 1 is normal in both odd and even lines
- 01 generator 1 output is force to 0 in odd lines
- 10 generator 1 output is force to 0 in even lines
- 11 generator 1 is normal in both odd and even line

F[4:3] : Generator 1 toggle mode

- 00 Disable
- 01 Toggle by pixel clock (LSHIFT)
- 10 Toggle by Line (LLINE)
- 11 Toggle by Frame (LFRAME)

GP1[10:8] : The highest 3 bits of the generator 1 period (POR = 100)

GP1[7:0] : The lower byte of the generator 1 period (POR = 00000000)

9.51 set_lcd_gen2

Command 0xC4

Parameters 7

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|------------------|------------------|------------------|------------------|------------------|-------------------|------------------|------------------|-----|
| Command | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | C4 |
| Parameter 1 | 1 | A ₇ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | xx |
| Parameter 2 | 1 | 0 | 0 | 0 | 0 | 0 | GF2 ₁₀ | GF2 ₉ | GF2 ₈ | xx |
| Parameter 3 | 1 | GF2 ₇ | GF2 ₆ | GF2 ₅ | GF2 ₄ | GF2 ₃ | GF2 ₂ | GF2 ₁ | GF2 ₀ | xx |
| Parameter 4 | 1 | 0 | 0 | 0 | 0 | 0 | GR2 ₁₀ | GR2 ₉ | GR2 ₈ | xx |
| Parameter 5 | 1 | GR2 ₇ | GR2 ₆ | GR2 ₅ | GR2 ₄ | GR2 ₃ | GR2 ₂ | GR2 ₁ | GR2 ₀ | xx |
| Parameter 6 | 1 | F ₇ | F ₆ | F ₅ | F ₄ | F ₃ | GP2 ₁₀ | GP2 ₉ | GP2 ₈ | xx |
| Parameter 7 | 1 | GP2 ₇ | GP2 ₆ | GP2 ₅ | GP2 ₄ | GP2 ₃ | GP2 ₂ | GP2 ₁ | GP2 ₀ | xx |

Description

Set the rise, fall, period and toggling properties of LCD signal generator 2

A[7] : Reset LCD generator 2 at every frame start

0 The generator 2 will not reset in the starting point of a frame

1 The generator 2 will reset in the starting point of a frame

GF2[10:8] : The highest 3 bits of the generator 2 falling position (POR = 000)

GF2[7:0] : The lower byte of the generator 2 falling position (POR = 00000001)

GR2[10:8] : The highest 3 bits of the generator 2 rising position (POR = 000)

GR2[7:0] : The lower byte of the generator 2 rising position (POR = 00000000)

F[7] : Force the generator 2 output to 0 in non-display period

0 generator 2 is normal

1 generator 2 output is forced to 0 in non-display period

F[6:5] : Force the generator 2 output to 0 in odd or even lines

00 generator 2 is normal in both odd and even lines

01 generator 2 output is force to 0 in odd lines

10 generator 2 output is force to 0 in even lines

11 generator 2 is normal in both odd and even line

F[4:3] : Generator 2 toggle mode

00 Disable

01 Toggle by pixel clock (LSHIFT)

10 Toggle by Line (LLINE)

11 Toggle by Frame (LFRAME)

GP2[10:8] : The highest 3 bits of the generator 2 period (POR = 100)

GP2[7:0] : The lower byte of the generator 2 period (POR = 00000000)

9.52 get_lcd_gen2

Command 0xC5

Parameters 7

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|------------------|------------------|------------------|------------------|------------------|-------------------|------------------|------------------|-----|
| Command | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | C5 |
| Parameter 1 | 1 | A ₇ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | xx |
| Parameter 2 | 1 | 0 | 0 | 0 | 0 | 0 | GF2 ₁₀ | GF2 ₉ | GF2 ₈ | xx |
| Parameter 3 | 1 | GF2 ₇ | GF2 ₆ | GF2 ₅ | GF2 ₄ | GF2 ₃ | GF2 ₂ | GF2 ₁ | GF2 ₀ | xx |
| Parameter 4 | 1 | 0 | 0 | 0 | 0 | 0 | GR2 ₁₀ | GR2 ₉ | GR2 ₈ | xx |
| Parameter 5 | 1 | GR2 ₇ | GR2 ₆ | GR2 ₅ | GR2 ₄ | GR2 ₃ | GR2 ₂ | GR2 ₁ | GR2 ₀ | xx |
| Parameter 6 | 1 | F ₇ | F ₆ | F ₅ | F ₄ | F ₃ | GP2 ₁₀ | GP2 ₉ | GP2 ₈ | xx |
| Parameter 7 | 1 | GP2 ₇ | GP2 ₆ | GP2 ₅ | GP2 ₄ | GP2 ₃ | GP2 ₂ | GP2 ₁ | GP2 ₀ | xx |

Description

Get the rise, fall, period and toggling properties of LCD signal generator 2

A[7] : Reset LCD generator 2 at every frame start

- 0 The generator 2 will not reset in the starting point of a frame
- 1 The generator 2 will reset in the starting point of a frame

GF2[10:8] : The highest 3 bits of the generator 2 falling position (POR = 000)

GF2[7:0] : The lower byte of the generator 2 falling position (POR = 00000001)

GR2[10:8] : The highest 3 bits of the generator 2 rising position (POR = 000)

GR2[7:0] : The lower byte of the generator 2 rising position (POR = 00000000)

F[7] : Force the generator 2 output to 0 in non-display period

- 0 generator 2 is normal
- 1 generator 2 output is forced to 0 in non-display period

F[6:5] : Force the generator 2 output to 0 in odd or even lines

- 00 generator 2 is normal in both odd and even lines
- 01 generator 2 output is force to 0 in odd lines
- 10 generator 2 output is force to 0 in even lines
- 11 generator 2 is normal in both odd and even line

F[4:3] : Generator 2 toggle mode

- 00 Disable
- 01 Toggle by pixel clock (LSHIFT)
- 10 Toggle by Line (LLINE)
- 11 Toggle by Frame (LFRAME)

GP2[10:8] : The highest 3 bits of the generator 2 period (POR = 100)

GP2[7:0] : The lower byte of the generator 2 period (POR = 00000000)

9.53 set_lcd_gen3

Command 0xC6

Parameters 7

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|------------------|------------------|------------------|------------------|------------------|-------------------|------------------|------------------|-----|
| Command | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | C6 |
| Parameter 1 | 1 | A ₇ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | xx |
| Parameter 2 | 1 | 0 | 0 | 0 | 0 | 0 | GF3 ₁₀ | GF3 ₉ | GF3 ₈ | xx |
| Parameter 3 | 1 | GF3 ₇ | GF3 ₆ | GF3 ₅ | GF3 ₄ | GF3 ₃ | GF3 ₂ | GF3 ₁ | GF3 ₀ | xx |
| Parameter 4 | 1 | 0 | 0 | 0 | 0 | 0 | GR3 ₁₀ | GR3 ₉ | GR3 ₈ | xx |
| Parameter 5 | 1 | GR3 ₇ | GR3 ₆ | GR3 ₅ | GR3 ₄ | GR3 ₃ | GR3 ₂ | GR3 ₁ | GR3 ₀ | xx |
| Parameter 6 | 1 | F ₇ | F ₆ | F ₅ | F ₄ | F ₃ | GP3 ₁₀ | GP3 ₉ | GP3 ₈ | xx |
| Parameter 7 | 1 | GP3 ₇ | GP3 ₆ | GP3 ₅ | GP3 ₄ | GP3 ₃ | GP3 ₂ | GP3 ₁ | GP3 ₀ | xx |

Description

Set the rise, fall, period and toggling properties of LCD signal generator 3

A[7] : Reset LCD generator 3 at every frame start

0 The generator 3 will not reset in the starting point of a frame

1 The generator 3 will reset in the starting point of a frame

GF3[10:8] : The highest 3 bits of the generator 3 falling position (POR = 000)

GF3[7:0] : The lower byte of the generator 3 falling position (POR = 00000001)

GR3[10:8] : The highest 3 bits of the generator 3 rising position (POR = 000)

GR3[7:0] : The lower byte of the generator 3 rising position (POR = 00000000)

F[7] : Force the generator 3 output to 0 in non-display period

0 generator 3 is normal

1 generator 3 output is forced to 0 in non-display period

F[6:5] : Force the generator 3 output to 0 in odd or even lines

00 generator 3 is normal in both odd and even lines

01 generator 3 output is force to 0 in odd lines

10 generator 3 output is force to 0 in even lines

11 generator 3 is normal in both odd and even line

F[4:3] : Generator 3 toggle mode

00 Disable

01 Toggle by pixel clock (LSHIFT)

10 Toggle by Line (LLINE)

11 Toggle by Frame (LFRAME)

GP3[10:8] : The highest 3 bits of the generator 3 period (POR = 100)

GP3[7:0] : The lower byte of the generator 3 period (POR = 00000000)

9.54 get_lcd_gen3

Command 0xC7

Parameters 7

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|------------------|------------------|------------------|------------------|------------------|-------------------|------------------|------------------|-----|
| Command | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | C7 |
| Parameter 1 | 1 | A ₇ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | xx |
| Parameter 2 | 1 | 0 | 0 | 0 | 0 | 0 | GF3 ₁₀ | GF3 ₉ | GF3 ₈ | xx |
| Parameter 3 | 1 | GF3 ₇ | GF3 ₆ | GF3 ₅ | GF3 ₄ | GF3 ₃ | GF3 ₂ | GF3 ₁ | GF3 ₀ | xx |
| Parameter 4 | 1 | 0 | 0 | 0 | 0 | 0 | GR3 ₁₀ | GR3 ₉ | GR3 ₈ | xx |
| Parameter 5 | 1 | GR3 ₇ | GR3 ₆ | GR3 ₅ | GR3 ₄ | GR3 ₃ | GR3 ₂ | GR3 ₁ | GR3 ₀ | xx |
| Parameter 6 | 1 | F ₇ | F ₆ | F ₅ | F ₄ | F ₃ | GP3 ₁₀ | GP3 ₉ | GP3 ₈ | xx |
| Parameter 7 | 1 | GP3 ₇ | GP3 ₆ | GP3 ₅ | GP3 ₄ | GP3 ₃ | GP3 ₂ | GP3 ₁ | GP3 ₀ | xx |

Description

Get the rise, fall, period and toggling properties of LCD signal generator 3

A[7] : Reset LCD generator 3 at every frame start

0 The generator 3 will not reset in the starting point of a frame

1 The generator 3 will reset in the starting point of a frame

GF3[10:8] : The highest 3 bits of the generator 3 falling position (POR = 000)

GF3[7:0] : The lower byte of the generator 3 falling position (POR = 00000001)

GR3[10:8] : The highest 3 bits of the generator 3 rising position (POR = 000)

GR3[7:0] : The lower byte of the generator 3 rising position (POR = 00000000)

F[7] : Force the generator 3 output to 0 in non-display period

0 generator 3 is normal

1 generator 3 output is forced to 0 in non-display period

F[6:5] : Force the generator 3 output to 0 in odd or even lines

00 generator 3 is normal in both odd and even lines

01 generator 3 output is force to 0 in odd lines

10 generator 3 output is force to 0 in even lines

11 generator 3 is normal in both odd and even line

F[4:3] : Generator 3 toggle mode

00 Disable

01 Toggle by pixel clock (LSHIFT)

10 Toggle by Line (LLINE)

11 Toggle by Frame (LFRAME)

GP3[10:8] : The highest 3 bits of the generator 3 period (POR = 100)

GP3[7:0] : The lower byte of the generator 3 period (POR = 00000000)

9.55 set_gpio0_rop

Command 0xC8

Parameters 2

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|-----|
| Command | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | C8 |
| Parameter 1 | 1 | 0 | A ₆ | A ₅ | 0 | A ₃ | A ₂ | A ₁ | A ₀ | xx |
| Parameter 2 | 1 | B ₇ | B ₆ | B ₅ | B ₄ | B ₃ | B ₂ | B ₁ | B ₀ | xx |

Description

Set the GPIO0 with respect to the LCD signal generators using ROP operation. No effect if the GPIO0 is configured as general GPIO.

A[6:5] : Source 1 for GPIO0 when controlled by LCDC (POR = 00)

00 Generator 0
01 Generator 1
10 Generator 2
11 Generator 3

A[3:2] : Source 2 for GPIO0 when controlled by LCDC (POR = 00)

00 Generator 0
01 Generator 1
10 Generator 2
11 Generator 3

A[1:0] : Source 3 for GPIO0 when controlled by LCDC (POR = 00)

00 Generator 0
01 Generator 1
10 Generator 2
11 Generator 3

B[7:0] : ROP operation to mux the source 1, 2 and 3 for GPIO0 (POR = 00000000)

Please refer to the Application note for the ROP operation

9.56 get_gpio0_rop

Command 0xC9

Parameters 2

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|-----|
| Command | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | C9 |
| Parameter 1 | 1 | 0 | A ₆ | A ₅ | 0 | A ₃ | A ₂ | A ₁ | A ₀ | xx |
| Parameter 2 | 1 | B ₇ | B ₆ | B ₅ | B ₄ | B ₃ | B ₂ | B ₁ | B ₀ | xx |

Description

Get the GPIO0 properties with respect to the LCD signal generators.

A[6:5] : Source 1 for GPIO0 when controlled by LCDC (POR = 00)

00 Generator 0
01 Generator 1
10 Generator 2
11 Generator 3

A[3:2] : Source 2 for GPIO0 when controlled by LCD (POR = 00)

| | |
|----|-------------|
| 00 | Generator 0 |
| 01 | Generator 1 |
| 10 | Generator 2 |
| 11 | Generator 3 |

A[1:0] : Source 3 for GPIO0 when controlled by LCD (POR = 00)

| | |
|----|-------------|
| 00 | Generator 0 |
| 01 | Generator 1 |
| 10 | Generator 2 |
| 11 | Generator 3 |

B[7:0] : ROP operation to mux the source 1, 2 and 3 for GPIO0 (POR = 00000000)

Please refer to the Application note for ROP operation

9.57 set_gpio1_rop

Command 0xCA

Parameters 2

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|-----|
| Command | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | CA |
| Parameter 1 | 1 | 0 | A ₆ | A ₅ | 0 | A ₃ | A ₂ | A ₁ | A ₀ | xx |
| Parameter 2 | 1 | B ₇ | B ₆ | B ₅ | B ₄ | B ₃ | B ₂ | B ₁ | B ₀ | xx |

Description

Set the GPIO1 with respect to the LCD signal generators using ROP operation. No effect if the GPIO1 is configured as general GPIO.

A[6:5] : Source 1 for GPIO1 when controlled by LCD (POR = 00)

| | |
|----|-------------|
| 00 | Generator 0 |
| 01 | Generator 1 |
| 10 | Generator 2 |
| 11 | Generator 3 |

A[3:2] : Source 2 for GPIO1 when controlled by LCD (POR = 00)

| | |
|----|-------------|
| 00 | Generator 0 |
| 01 | Generator 1 |
| 10 | Generator 2 |
| 11 | Generator 3 |

A[1:0] : Source 3 for GPIO1 when controlled by LCD (POR = 00)

| | |
|----|-------------|
| 00 | Generator 0 |
| 01 | Generator 1 |
| 10 | Generator 2 |
| 11 | Generator 3 |

B[7:0] : ROP operation to mux the source 1, 2 and 3 for GPIO1 (POR = 00000000)

Please refer to the Application note for the ROP operation

9.58 get_gpio1_rop

Command 0xCB

Parameters 2

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|-----|
| Command | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | CB |
| Parameter 1 | 1 | 0 | A ₆ | A ₅ | 0 | A ₃ | A ₂ | A ₁ | A ₀ | xx |
| Parameter 2 | 1 | B ₇ | B ₆ | B ₅ | B ₄ | B ₃ | B ₂ | B ₁ | B ₀ | xx |

Description

Get the GPIO1 properties with respect to the LCD signal generators.

A[6:5] : Source 1 for GPIO1 when controlled by LCDC (POR = 00)

00 Generator 0
01 Generator 1
10 Generator 2
11 Generator 3

A[3:2] : Source 2 for GPIO1 when controlled by LCDC (POR = 00)

00 Generator 0
01 Generator 1
10 Generator 2
11 Generator 3

A[1:0] : Source 3 for GPIO1 when controlled by LCDC (POR = 00)

00 Generator 0
01 Generator 1
10 Generator 2
11 Generator 3

B[7:0] : ROP operation to mux the source 1, 2 and 3 for GPIO1 (POR = 00000000)

Please refer to the Application note for the ROP operation

9.59 set_gpio2_rop

Command 0xCC

Parameters 2

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|-----|
| Command | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | CC |
| Parameter 1 | 1 | 0 | A ₆ | A ₅ | 0 | A ₃ | A ₂ | A ₁ | A ₀ | xx |
| Parameter 2 | 1 | B ₇ | B ₆ | B ₅ | B ₄ | B ₃ | B ₂ | B ₁ | B ₀ | xx |

Description

Set the GPIO2 with respect to the LCD signal generators using ROP operation. No effect if the GPIO2 is configured as general GPIO.

A[6:5] : Source 1 for GPIO2 when controlled by LCDC (POR = 00)

00 Generator 0
01 Generator 1
10 Generator 2
11 Generator 3

A[3:2] : Source 2 for GPIO2 when controlled by LCD (POR = 00)
 00 Generator 0
 01 Generator 1
 10 Generator 2
 11 Generator 3

A[1:0] : Source 3 for GPIO2 when controlled by LCD (POR = 00)
 00 Generator 0
 01 Generator 1
 10 Generator 2
 11 Generator 3

B[7:0] : ROP operation to mux the source 1, 2 and 3 for GPIO2 (POR = 00000000)
 Please refer to the Application note for the ROP operation

9.60 get_gpio2_rop

Command 0xCD
Parameters 2

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|-----|
| Command | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | CD |
| Parameter 1 | 1 | 0 | A ₆ | A ₅ | 0 | A ₃ | A ₂ | A ₁ | A ₀ | xx |
| Parameter 2 | 1 | B ₇ | B ₆ | B ₅ | B ₄ | B ₃ | B ₂ | B ₁ | B ₀ | xx |

Description

Get the GPIO2 properties with respect to the LCD signal generators.

A[6:5] : Source 1 for GPIO2 when controlled by LCD (POR = 00)
 00 Generator 0
 01 Generator 1
 10 Generator 2
 11 Generator 3

A[3:2] : Source 2 for GPIO2 when controlled by LCD (POR = 00)
 00 Generator 0
 01 Generator 1
 10 Generator 2
 11 Generator 3

A[1:0] : Source 3 for GPIO2 when controlled by LCD (POR = 00)
 00 Generator 0
 01 Generator 1
 10 Generator 2
 11 Generator 3

B[7:0] : ROP operation to mux the source 1, 2 and 3 for GPIO2 (POR = 00000000)
 Please refer to the Application note for the ROP operation

9.61 set_gpio3_rop

Command 0xCE

Parameters 2

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|-----|
| Command | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | CE |
| Parameter 1 | 1 | 0 | A ₆ | A ₅ | 0 | A ₃ | A ₂ | A ₁ | A ₀ | xx |
| Parameter 2 | 1 | B ₇ | B ₆ | B ₅ | B ₄ | B ₃ | B ₂ | B ₁ | B ₀ | xx |

Description

Set the GPIO3 with respect to the LCD signal generators using ROP operation. No effect if the GPIO3 is configured as general GPIO.

A[6:5] : Source 1 for GPIO3 when controlled by LCDC (POR = 00)

00 Generator 0
01 Generator 1
10 Generator 2
11 Generator 3

A[3:2] : Source 2 for GPIO3 when controlled by LCDC (POR = 00)

00 Generator 0
01 Generator 1
10 Generator 2
11 Generator 3

A[1:0] : Source 3 for GPIO3 when controlled by LCDC (POR = 00)

00 Generator 0
01 Generator 1
10 Generator 2
11 Generator 3

B[7:0] : ROP operation to mux the source 1, 2 and 3 for GPIO3 (POR = 00000000)

Please refer to the Application note for the ROP operation

9.62 get_gpio3_rop

Command 0xCF

Parameters 2

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|-----|
| Command | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | CF |
| Parameter 1 | 1 | 0 | A ₆ | A ₅ | 0 | A ₃ | A ₂ | A ₁ | A ₀ | xx |
| Parameter 2 | 1 | B ₇ | B ₆ | B ₅ | B ₄ | B ₃ | B ₂ | B ₁ | B ₀ | xx |

Description

Get the GPIO3 properties with respect to the LCD signal generators.

A[6:5] : Source 1 for GPIO3 when controlled by LCDC (POR = 00)

00 Generator 0
01 Generator 1
10 Generator 2
11 Generator 3

A[3:2] : Source 2 for GPIO3 when controlled by LCD (POR = 00)

| | |
|----|-------------|
| 00 | Generator 0 |
| 01 | Generator 1 |
| 10 | Generator 2 |
| 11 | Generator 3 |

A[1:0] : Source 3 for GPIO3 when controlled by LCD (POR = 00)

| | |
|----|-------------|
| 00 | Generator 0 |
| 01 | Generator 1 |
| 10 | Generator 2 |
| 11 | Generator 3 |

B[7:0] : ROP operation to mux the source 1, 2 and 3 for GPIO3 (POR = 00000000)

Please refer to the Application note for the ROP operation

9.63 set_dbc_conf

Command 0xD0

Parameters 1

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----|----------------|----------------|----|----------------|----------------|----|----------------|-----|
| Command | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | D0 |
| Parameter 1 | 1 | 0 | A ₆ | A ₅ | 0 | A ₃ | A ₂ | 0 | A ₀ | xx |

Description

Set the Dynamic Backlight Control configuration.

A[6] : DBC Manual Brightness enable (POR = 1)

| | |
|---|---------|
| 0 | Enable |
| 1 | Disable |

A[5] : Transition effect (POR = 0)

| | |
|---|---------------------------|
| 0 | Transition effect disable |
| 1 | Transition effect enable |

Transition effect is used to remove visible backlight flickering. If rapid brightness change is required, it is recommended to enable this bit.

A[3:2] : Energy saving selection for DBC (POR = 00)

| | |
|----|-------------------|
| 00 | DBC is disable |
| 01 | Conservative mode |
| 10 | Normal mode |
| 11 | Aggressive mode |

A[0] : Master enable of DBC (POR = 0)

| | |
|---|-------------|
| 0 | DBC disable |
| 1 | DBC enable |

The hardware pin, PWM is the output signal from SSD1963 to the system backlight driver. So it should configure PWM module before enable DBC.

```

WRITE COMMAND "0xBE"
WRITE DATA "0x0E" (set PWM frequency)
WRITE DATA "0xFF" (dummy value if DBC is used)
WRITE DATA "0x09" (enable PWM controlled by DBC)
WRITE DATA "0xFF"
WRITE DATA "0x00"
WRITE DATA "0x00"

WRITE COMMAND "0xD4" (Define the threshold value)
WRITE DATA .....

WRITE COMMAND "0xD0"
WRITE DATA "0x0D" (Enable DBC with Aggressive mode)

```

9.64 get_dbc_conf

Command 0xD1
Parameters 1

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----|----------------|----------------|----|----------------|----------------|----|----------------|-----|
| Command | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | D1 |
| Parameter 1 | 1 | 0 | A ₆ | A ₅ | 0 | A ₃ | A ₂ | 1 | A ₀ | xx |

Description

Get the current dynamic back light configuration.

A[6] : DBC Manual Brightness enable (POR = 1)

0 Enable
1 Disable

A[5] : Transition effect (POR = 0)

0 Transition effect disable
1 Transition effect enable

A[3:2] : Energy saving selection for DBC (POR = 00)

00 DBC is disable
01 Conservative mode
10 Normal mode
11 Aggressive mode

A[0] : Master enable DBC (POR = 0)

0 DBC disable
1 DBC enable

9.65 set_dbc_th

Command 0xD4
Parameters 9

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|----------------------|-----------------------|-----|
| Command | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | D4 |
| Parameter 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | DBC_TH1 ₁₆ | xx |
| Parameter 2 | 1 | DBC_TH1 ₁₅ | DBC_TH1 ₁₄ | DBC_TH1 ₁₃ | DBC_TH1 ₁₂ | DBC_TH1 ₁₁ | DBC_TH1 ₁₀ | DBC_TH1 ₉ | DBC_TH1 ₈ | xx |
| Parameter 3 | 1 | DBC_TH1 ₇ | DBC_TH1 ₆ | DBC_TH1 ₅ | DBC_TH1 ₄ | DBC_TH1 ₃ | DBC_TH1 ₂ | DBC_TH1 ₁ | DBC_TH1 ₀ | xx |
| Parameter 4 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | DBC_TH2 ₁₆ | xx |
| Parameter 5 | 1 | DBC_TH2 ₁₅ | DBC_TH2 ₁₄ | DBC_TH2 ₁₃ | DBC_TH2 ₁₂ | DBC_TH2 ₁₁ | DBC_TH2 ₁₀ | DBC_TH2 ₉ | DBC_TH2 ₈ | xx |
| Parameter 6 | 1 | DBC_TH2 ₇ | DBC_TH2 ₆ | DBC_TH2 ₅ | DBC_TH2 ₄ | DBC_TH2 ₃ | DBC_TH2 ₂ | DBC_TH2 ₁ | DBC_TH2 ₀ | xx |
| Parameter 7 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | DBC_TH3 ₁₆ | xx |
| Parameter 8 | 1 | DBC_TH3 ₁₅ | DBC_TH3 ₁₄ | DBC_TH3 ₁₃ | DBC_TH3 ₁₂ | DBC_TH3 ₁₁ | DBC_TH3 ₁₀ | DBC_TH3 ₉ | DBC_TH3 ₈ | xx |
| Parameter 9 | 1 | DBC_TH3 ₇ | DBC_TH3 ₆ | DBC_TH3 ₅ | DBC_TH3 ₄ | DBC_TH3 ₃ | DBC_TH3 ₂ | DBC_TH3 ₁ | DBC_TH3 ₀ | xx |

Description

Set the threshold for each level of power saving.

DBC_TH1[16] : High byte of the threshold setting for the Conservative mode of DBC. (POR = 0)
 DBC_TH1[15:8] : 2nd byte of the threshold setting for the Conservative mode of DBC. (POR = 00000000)
 DBC_TH1[7:0] : Low byte of the threshold setting for the Conservative mode of DBC. (POR = 00000000)

$TH1 = \text{display width} * \text{display height} * 3 * 0.1 / 16$

DBC_TH2[16] : High byte of the threshold setting for the Normal mode of DBC. (POR = 0)
 DBC_TH2[15:8] : 2nd byte of the threshold setting for the Normal mode of DBC. (POR = 00000000)
 DBC_TH2[7:0] : Low byte of the threshold setting for the Normal mode of DBC. (POR = 00000000)

$TH2 = \text{display width} * \text{display height} * 3 * 0.25 / 16$

DBC_TH3[16] : High byte of the threshold setting for the Aggressive mode of DBC. (POR = 0)
 DBC_TH3[15:8] : 2nd byte of the threshold setting for the Aggressive mode of DBC. (POR = 00000000)
 DBC_TH3[7:0] : Low byte of the threshold setting for the Aggressive mode of DBC. (POR = 00000000)

$TH3 = \text{display width} * \text{display height} * 3 * 0.6 / 16$

9.66 get_dbc_th

Command 0xD5
Parameters 9

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|----------------------|-----------------------|-----|
| Command | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | D5 |
| Parameter 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | DBC_TH1 ₁₆ | xx |
| Parameter 2 | 1 | DBC_TH1 ₁₅ | DBC_TH1 ₁₄ | DBC_TH1 ₁₃ | DBC_TH1 ₁₂ | DBC_TH1 ₁₁ | DBC_TH1 ₁₀ | DBC_TH1 ₉ | DBC_TH1 ₈ | xx |
| Parameter 3 | 1 | DBC_TH1 ₇ | DBC_TH1 ₆ | DBC_TH1 ₅ | DBC_TH1 ₄ | DBC_TH1 ₃ | DBC_TH1 ₂ | DBC_TH1 ₁ | DBC_TH1 ₀ | xx |
| Parameter 4 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | DBC_TH2 ₁₆ | xx |
| Parameter 5 | 1 | DBC_TH2 ₁₅ | DBC_TH2 ₁₄ | DBC_TH2 ₁₃ | DBC_TH2 ₁₂ | DBC_TH2 ₁₁ | DBC_TH2 ₁₀ | DBC_TH2 ₉ | DBC_TH2 ₈ | xx |
| Parameter 6 | 1 | DBC_TH2 ₇ | DBC_TH2 ₆ | DBC_TH2 ₅ | DBC_TH2 ₄ | DBC_TH2 ₃ | DBC_TH2 ₂ | DBC_TH2 ₁ | DBC_TH2 ₀ | xx |
| Parameter 7 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | DBC_TH3 ₁₆ | xx |
| Parameter 8 | 1 | DBC_TH3 ₁₅ | DBC_TH3 ₁₄ | DBC_TH3 ₁₃ | DBC_TH3 ₁₂ | DBC_TH3 ₁₁ | DBC_TH3 ₁₀ | DBC_TH3 ₉ | DBC_TH3 ₈ | xx |
| Parameter 9 | 1 | DBC_TH3 ₇ | DBC_TH3 ₆ | DBC_TH3 ₅ | DBC_TH3 ₄ | DBC_TH3 ₃ | DBC_TH3 ₂ | DBC_TH3 ₁ | DBC_TH3 ₀ | xx |

Description

Get the threshold for each level of power saving.

DBC_TH1[16] : High byte of the threshold setting for the Conservative mode of DBC. (POR = 0)
 DBC_TH1[15:8] : 2nd byte of the threshold setting for the Conservative mode of DBC. (POR = 00000000)
 DBC_TH1[7:0] : Low byte of the threshold setting for the Conservative mode of DBC. (POR = 00000000)

DBC_TH2[16] : High byte of the threshold setting for the Normal mode of DBC. (POR = 0)
 DBC_TH2[15:8] : 2nd byte of the threshold setting for the Normal mode of DBC. (POR = 00000000)
 DBC_TH2[7:0] : Low byte of the threshold setting for the Normal mode of DBC. (POR = 00000000)

DBC_TH3[16] : High byte of the threshold setting for the Aggressive mode of DBC. (POR = 0)
 DBC_TH3[15:8] : 2nd byte of the threshold setting for the Aggressive mode of DBC. (POR = 00000000)
 DBC_TH3[7:0] : Low byte of the threshold setting for the Aggressive mode of DBC. (POR = 00000000)

9.67 set_pll

Command 0xE0
Parameters 1

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----|----|----|----|----|----|----------------|----------------|-----|
| Command | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | E0 |
| Parameter 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | A ₁ | A ₀ | xx |

Description

Start the PLL. Before the start, the system was operated with the crystal oscillator or clock input.

A[1] : Lock PLL (POR = 0)
 After PLL enabled for 100us, can start to lock PLL
 0 Use reference clock as system clock
 1 Use PLL output as system clock

A[0] : Enable PLL (POR = 0)
 0 Disable PLL
 1 Enable PLL

Before enabling PLL, the PLL setting (“0xE2”) have to be configured first. After PLL enabled for 100us, can start to lock PLL. SSD1963 needed to switch to PLL output as system clock after PLL is locked. The following is the program sequence.

```
WRITE COMMAND “0xE0”
WRITE DATA “0x01”
Wait 100us to let the PLL stable
WRITE COMMAND “0xE0”
WRITE DATA “0x03”
WRITE COMMAND “0x01”
```

* Note : SSD1963 is operating under reference clock before PLL is locked, registers cannot be set faster than half of the reference clock frequency. For instance, SSD1963 with a 10MHz reference clock is not allowed to be programmed higher than 5M words/s.

9.68 set_pll_mn

Command 0xE2
Parameters 3

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|-----|
| Command | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | E2 |
| Parameter 1 | 1 | M ₇ | M ₆ | M ₅ | M ₄ | M ₃ | M ₂ | M ₁ | M ₀ | xx |
| Parameter 2 | 1 | 0 | 0 | 1 | N ₄ | N ₃ | N ₂ | N ₁ | N ₀ | xx |
| Parameter 3 | 1 | 0 | 0 | 0 | 0 | 0 | C ₂ | 0 | 0 | xx |

Description

Set the MN of PLL

M[7:0] : Multiplier (M) of PLL. (POR = 00101101)

N[4:0] : Divider (N) of PLL. (POR = 00011)

C[2] : Effectuate MN value (POR = 0)

- 0 Ignore the multiplier (N) and divider (N) values
- 1 Effectuate the multiplier and divider value

$VCO = \text{Reference input clock} \times (M + 1)$

$\text{PLL frequency} = VCO / (N + 1)$

*** Note : 250MHz < VCO < 800MHz**

For a 10MHz reference clock to obtain 100MHz PLL frequency, user cannot program M = 19 and N = 1. The setting in this situation is setting M=29 and N=2, where $10 \times 30 / 3 = 100\text{MHz}$.

```
WRITE COMMAND “0xE2”
WRITE DATA “0x1D” (M=29)
WRITE DATA “0x02” (N=2)
WRITE DATA “0x54” (Dummy Byte)
```

9.69 get_pll_mn

Command 0xE3
Parameters 3

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|-----|
| Command | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | E3 |
| Parameter 1 | 1 | M ₇ | M ₆ | M ₅ | M ₄ | M ₃ | M ₂ | M ₁ | M ₀ | xx |
| Parameter 2 | 1 | 0 | 0 | 1 | N ₄ | N ₃ | N ₂ | N ₁ | N ₀ | xx |
| Parameter 3 | 1 | 0 | 0 | 0 | 0 | 0 | C ₂ | 0 | 0 | xx |

Description

Get the MN setting of PLL

M[7:0] : Multiplier (M) of PLL. (POR = 00101101)

N[4:0] : Divider (N) of PLL. (POR = 00011)

C[2] : Effectuate MN value (POR = 0)

- 0 Ignore the multiplier (M) and divider (N) values.
- 1 Effectuate the multiplier and divider value

9.70 get_pll_status

Command 0xE4
Parameters 1

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----|----|----|----|----|----------------|----|----|-----|
| Command | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | E4 |
| Parameter 1 | 1 | 0 | 0 | 0 | 0 | 0 | A ₂ | 0 | 0 | xx |

Description

Get the PLL status

A[2] : PLL Lock

- 0 Not locked
- 1 Locked

9.71 set_deep_sleep

Command 0xE5
Parameters None

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|---------|-----|----|----|----|----|----|----|----|----|-----|
| Command | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | E5 |

Description

Set deep sleep mode. PLL would be stopped.

It needs to issue 2 dummy read to exit Deep Sleep mode.

9.72 set_lshift_freq

Command 0xE6
Parameters 3

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|-----|
| Command | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | E6 |
| Parameter 1 | 1 | 0 | 0 | 0 | 0 | LCDC_FPR ₁₉ | LCDC_FPR ₁₈ | LCDC_FPR ₁₇ | LCDC_FPR ₁₆ | xx |
| Parameter 2 | 1 | LCDC_FPR ₁₅ | LCDC_FPR ₁₄ | LCDC_FPR ₁₃ | LCDC_FPR ₁₂ | LCDC_FPR ₁₁ | LCDC_FPR ₁₀ | LCDC_FPR ₉ | LCDC_FPR ₈ | xx |
| Parameter 3 | 1 | LCDC_FPR ₇ | LCDC_FPR ₆ | LCDC_FPR ₅ | LCDC_FPR ₄ | LCDC_FPR ₃ | LCDC_FPR ₂ | LCDC_FPR ₁ | LCDC_FPR ₀ | xx |

Description

Set the LSHIFT (pixel clock) frequency

LCDC_FPR[19:16] : The highest 4 bits for the pixel clock frequency settings. (POR = 0111)

LCDC_FPR[15:8] : The higher byte for the pixel clock frequency settings. (POR = 11111111)

LCDC_FPR[7:0] : The low byte for the pixel clock frequency settings. (POR = 11111111)

For parallel LCD interface:

Configure the pixel clock to PLL freq x ((LCDC_FPR + 1) / 2²⁰)

To obtain PCLK = 5.3MHz with PLL Frequency = 100MHz,

5.3MHz = 100MHz * (LCDC_FPR+ 1) / 2²⁰

LCDC_FPR = 55574

WRITE COMMAND "0xE6"

WRITE DATA "0x00" (LCDC_FPR = 55574)

WRITE DATA "0xD9"

WRITE DATA "0x16"

For serial LCD interface:

Configure the pixel clock to PLL freq x ((LCDC_FPR + 1) / 2²⁰) *4

To obtain PCLK = 5.3MHz with PLL Frequency = 100MHz,

5.3MHz = 100MHz * ((LCDC_FPR+ 1) / 2²⁰) *4

LCDC_FPR = 13892

WRITE COMMAND "0xE6"

WRITE DATA "0x00" (LCDC_FPR = 13892)

WRITE DATA "0x36"

WRITE DATA "0x44"

9.73 get_lshift_freq

Command 0xE7
Parameters 3

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|-----|
| Command | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | E7 |
| Parameter 1 | 1 | 0 | 0 | 0 | 0 | LCDC_FPR ₁₉ | LCDC_FPR ₁₈ | LCDC_FPR ₁₇ | LCDC_FPR ₁₆ | xx |
| Parameter 2 | 1 | LCDC_FPR ₁₅ | LCDC_FPR ₁₄ | LCDC_FPR ₁₃ | LCDC_FPR ₁₂ | LCDC_FPR ₁₁ | LCDC_FPR ₁₀ | LCDC_FPR ₉ | LCDC_FPR ₈ | xx |
| Parameter 3 | 1 | LCDC_FPR ₇ | LCDC_FPR ₆ | LCDC_FPR ₅ | LCDC_FPR ₄ | LCDC_FPR ₃ | LCDC_FPR ₂ | LCDC_FPR ₁ | LCDC_FPR ₀ | xx |

Description

Get the current LSHIFT (pixel clock) frequency setting

LCDC_FPR[19:16] : The highest 4 bits for the pixel clock frequency settings. (POR = 0111)
LCDC_FPR[15:8] : The higher byte for the pixel clock frequency settings. (POR = 11111111)
LCDC_FPR[7:0] : The low byte for the pixel clock frequency settings. (POR = 11111111)

9.74 set_pixel_data_interface

Command 0xF0
Parameters 1

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----|----|----|----|----|----------------|----------------|----------------|-----|
| Command | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | F0 |
| Parameter 1 | 1 | 0 | 0 | 0 | 0 | 0 | A ₂ | A ₁ | A ₀ | xx |

Description

Set the pixel data format to 8-bit / 9-bit / 12-bit / 16-bit / 16-bit(565) / 18-bit / 24-bit in the parallel host processor interface. This command is used for display data only, the command format is always 8 bit.

A[2:0] : Pixel Data Interface Format (POR = 101)

000 8-bit
001 12-bit
010 16-bit packed
011 16-bit (565 format)
100 18-bit
101 24-bit
110 9-bit
Others Reserved

* Note : The un-used data bus will be driven to ground by SSD1963, so don't connect the un-used data bus to MCU.

9.75 get_pixel_data_interface

Command 0xF1
Parameters 1

| | D/C | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex |
|-------------|-----|----|----|----|----|----|----------------|----------------|----------------|-----|
| Command | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | F1 |
| Parameter 1 | 1 | 0 | 0 | 0 | 0 | 0 | A ₂ | A ₁ | A ₀ | xx |

Description

Get the current pixel data format settings in the parallel host processor interface.

A[2:0] : Pixel Data Interface Format (POR = 101)

000 8-bit
001 12-bit
010 16-bit packed
011 16-bit (565 format)
100 18-bit
101 24-bit
110 9-bit
Others Reserved

10 MAXIMUM RATINGS

Table 10-1: Maximum Ratings (Voltage Referenced to V_{SS})

| Symbol | Parameter | Value | Unit |
|-------------|----------------------------|-----------------------------------|------|
| V_{DDD} | Digital Core power supply | -0.5 to 1.8 | V |
| V_{DDPLL} | PLL power supply | -0.5 to 1.8 | V |
| V_{DDLCD} | LCD Interface power supply | -0.5 to 4.6 | V |
| V_{DDIO} | I/O power supply | -0.5 to 4.6 | V |
| V_{IN} | Input Voltage | -0.5 to 4.6 | V |
| V_{OUT} | Output Voltage | -0.5 to 4.6 | V |
| T_{SOL} | Solder Temperature / Time | 225 for 40 sec max at solder ball | °C |
| T_{STG} | Storage temperature | -45 to 125 | °C |
| T_A | Operating temperature | -30 to 85 | °C |

Maximum ratings are those values beyond which damages to the device may occur. Functional operation should be restricted to the limits in the Electrical Characteristics tables or Pin Description section

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high impedance circuit. For proper operation it is recommended that V_{IN} and V_{OUT} be constrained to the range $V_{SS} < (V_{IN} \text{ or } V_{OUT}) < V_{DDIO}$. Reliability of operation is enhanced if unused input is connected to an appropriate logic voltage level (e.g., either V_{SS} or V_{DDIO}). Unused outputs must be left open. This device may be light sensitive. Caution should be taken to avoid exposure of this device to any light source during normal operation. This device is not radiation protected.

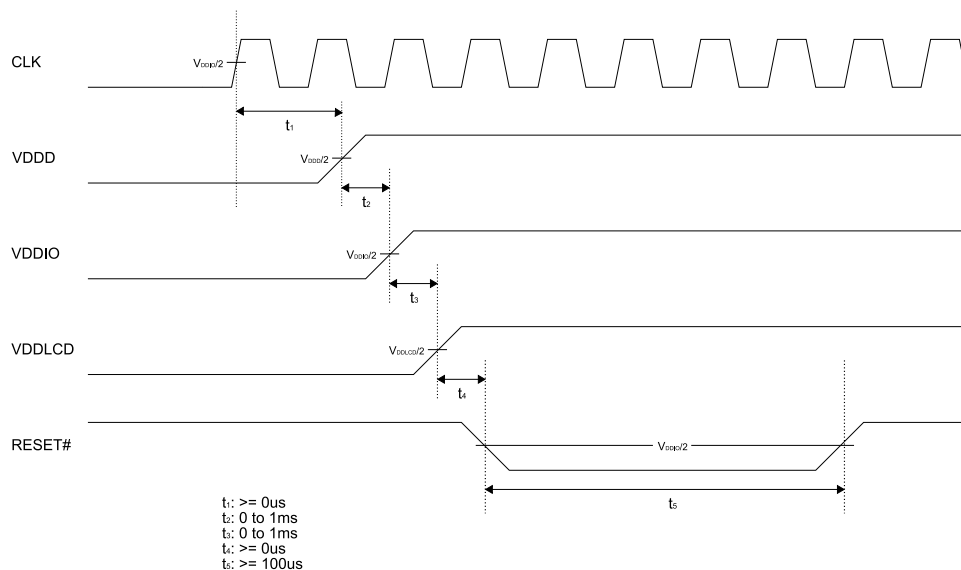
11 RECOMMENDED OPERATING CONDITIONS

Table 11-1: Recommended Operating Condition

| Symbol | Parameter | Min | Typ | Max | Unit |
|-------------|----------------------------|------|-----|------|------|
| V_{DDD} | Digital Core power supply | 1.10 | 1.2 | 1.30 | V |
| V_{DDPLL} | PLL power supply | 1.10 | 1.2 | 1.30 | V |
| V_{DDLCD} | LCD Interface power supply | 1.65 | 3.3 | 3.6 | V |
| V_{DDIO} | I/O power supply | 1.65 | 3.3 | 3.6 | V |

11.1 Power-up sequence

Figure 11-1: Power-up Sequence



Note

Clock reference is only applicable when CLK is used.

12 DC CHARACTERISTICS

Conditions:

Voltage referenced to V_{SS}

$V_{DD}, V_{DDPLL} = 1.2V$

$V_{DDIO}, V_{DDLCD} = 3.3V$

$T_A = 25^{\circ}C$

Table 12-1 : DC Characteristics

| Symbol | Parameter | Test Condition | Min | Typ | Max | Unit |
|------------------|------------------------|----------------|----------------------|-----|-------------------------|------|
| P _{STY} | Quiescent Power | | | 300 | 500 | uW |
| I _{IZ} | Input leakage current | | -1 | | 1 | uA |
| I _{OZ} | Output leakage current | | -1 | | 1 | uA |
| V _{OH} | Output high voltage | | 0.8V _{DDIO} | | | V |
| V _{OL} | Output low voltage | | | | 0.2V _{DDIO} | V |
| V _{IH} | Input high voltage | | 0.8V _{DDIO} | | V _{DDIO} + 0.5 | V |
| V _{IL} | Input low voltage | | | | 0.2V _{DDIO} | V |

13 AC CHARACTERISTICS

Conditions:

Voltage referenced to V_{SS}

$V_{DD}, V_{DDPLL} = 1.2V$

$V_{DDIO}, V_{DDLCD} = 3.3V$

$T_A = 25^{\circ}C$

$C_L = 50pF$ (Bus/CPU Interface)

$C_L = 0pF$ (LCD Panel Interface)

13.1 Clock Timing

Table 13-1: Clock Input Requirements for CLK (PLL-bypass)

| Symbol | Parameter | Min | Max | Units |
|------------------|-----------------------------|--------------------|-----|-------|
| F _{CLK} | Input Clock Frequency (CLK) | | 110 | MHz |
| T _{CLK} | Input Clock period (CLK) | 1/f _{CLK} | | ns |

Table 13-2 : Clock Input Requirements for CLK

| Symbol | Parameter | Min | Max | Units |
|------------------|-----------------------------|--------------------|-----|-------|
| F _{CLK} | Input Clock Frequency (CLK) | 2.5 | 50 | MHz |
| T _{CLK} | Input Clock period (CLK) | 1/f _{CLK} | | ns |

Table 13-3 : Clock Input Requirements for crystal oscillator XTAL

| Symbol | Parameter | Min | Max | Units |
|-------------------|-----------------------|---------------------|-----|-------|
| F _{XTAL} | Input Clock Frequency | 2.5 | 10 | MHz |
| T _{XTAL} | Input Clock period | 1/f _{XTAL} | | ns |

13.2 MCU Interface Timing

13.2.1 Parallel 6800-series Interface Timing

Table 13-4: Parallel 6800-series Interface Timing Characteristics (Use CS# as clock)

| Symbol | Parameter | Min | Typ | Max | Unit |
|-------------|---|----------------|--|-----|------|
| f_{MCLK} | System Clock Frequency* | 1 | - | 110 | MHz |
| t_{MCLK} | System Clock Period* | $1/f_{MCLK}$ | - | - | ns |
| t_{PWCSH} | Control Pulse High Width Write Read | 13 30 | $1.5 * t_{MCLK}$ $3.5 * t_{MCLK}$ | - | ns |
| t_{PWCSL} | Control Pulse Low Width Write (next write cycle) Write (next read cycle) Read | 13 80 80 | $1.5 * t_{MCLK}$ $9 * t_{MCLK}$ $9 * t_{MCLK}$ | - | ns |
| t_{AS} | Address Setup Time | 2 | - | - | ns |
| t_{AH} | Address Hold Time | 2 | - | - | ns |
| t_{DSW} | Data Setup Time | 4 | - | - | ns |
| t_{DHW} | Data Hold Time | 1 | - | - | ns |
| t_{PLW} | Write Low Time | 14 | - | - | ns |
| t_{PHW} | Write High Time | 14 | - | - | ns |
| t_{PLWR} | Read Low Time | 38 | - | - | ns |
| t_{ACC} | Data Access Time | 32 | - | - | ns |
| t_{DHR} | Output Hold time | 1 | - | - | ns |
| t_R | Rise Time | - | - | 0.5 | ns |
| t_F | Fall Time | - | - | 0.5 | ns |

* System Clock denotes external input clock (PLL-bypass) or internal generated clock (PLL-enabled)

Figure 13-1: Parallel 6800-series Interface Timing Diagram (Use CS# as Clock)

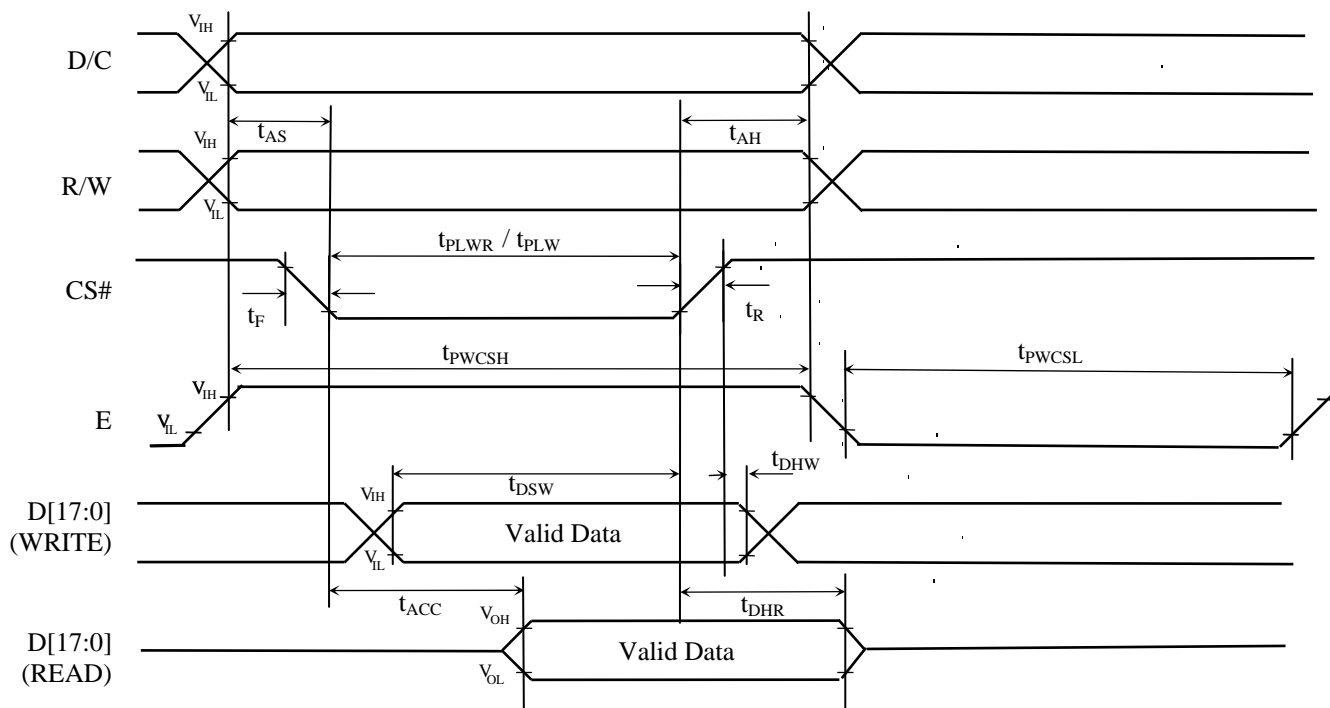
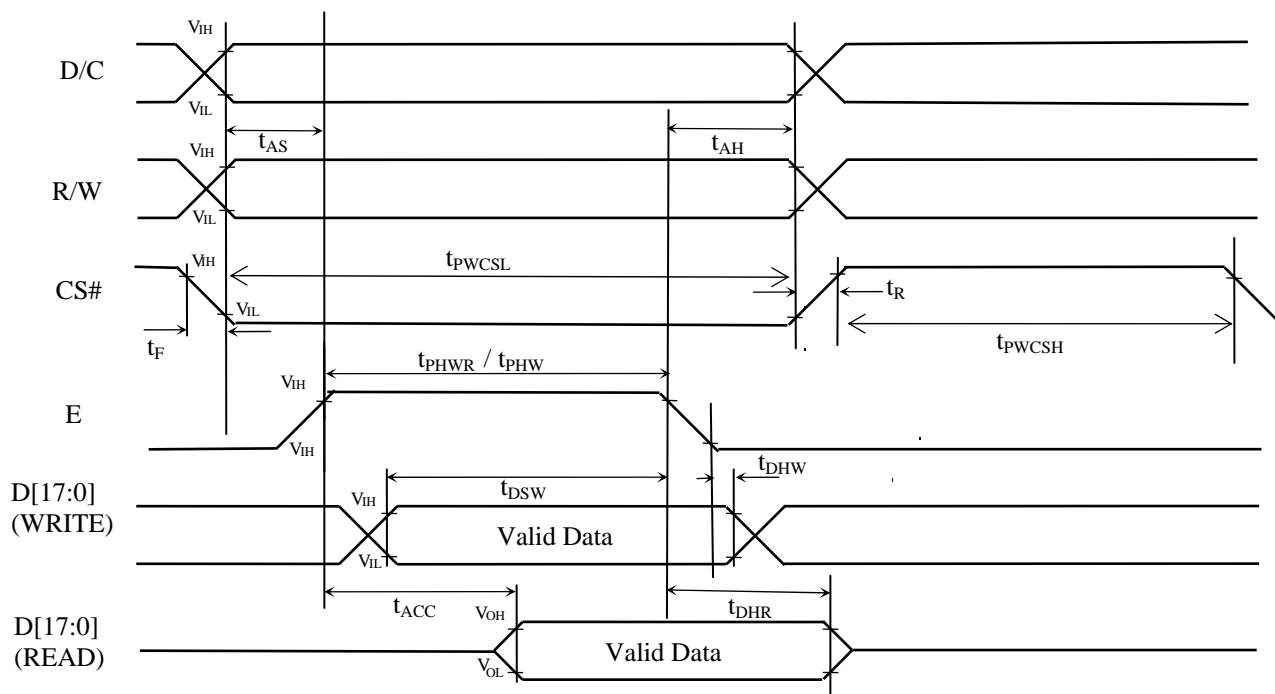


Table 13-5: Parallel 6800-series Interface Timing Characteristics (Use E as clock)

| Symbol | Parameter | Min | Typ | Max | Unit |
|-------------|--|----------------|--|-----|------|
| f_{MCLK} | System Clock Frequency* | 1 | - | 110 | MHz |
| t_{MCLK} | System Clock Period* | $1/f_{MCLK}$ | - | - | ns |
| t_{PWCSH} | Control Pulse High Width Write Read | 13 30 | $1.5^* t_{MCLK}$ $3.5^* t_{MCLK}$ | - | ns |
| t_{PWCSL} | Control Pulse Low Width Write (next write cycle) Write (next read cycle) Read | 13 80 80 | $1.5^* t_{MCLK}$ $9^* t_{MCLK}$ $9^* t_{MCLK}$ | - | ns |
| t_{AS} | Address Setup Time | 2 | - | - | ns |
| t_{AH} | Address Hold Time | 2 | - | - | ns |
| t_{DSW} | Data Setup Time | 4 | - | - | ns |
| t_{DHW} | Data Hold Time | 1 | - | - | ns |
| t_{PLW} | Write Low Time | 14 | - | - | ns |
| t_{PHW} | Write High Time | 14 | - | - | ns |
| t_{PLWR} | Read Low Time | 38 | - | - | ns |
| t_{ACC} | Data Access Time | 32 | - | - | ns |
| t_{DHR} | Output Hold Time | 1 | - | - | ns |
| t_R | Rise Time | - | - | 0.5 | ns |
| t_F | Fall Time | - | - | 0.5 | ns |

* System Clock denotes external input clock (PLL-bypass) or internal generated clock (PLL-enabled)

Figure 13-2: Parallel 6800-series Interface Timing Diagram (Use E as Clock)



13.2.2 Parallel 8080-series Interface Timing

Table 13-6: Parallel 8080-series Interface Timing Characteristics

| Symbol | Parameter | Min | Typ | Max | Unit |
|-------------|--|----------------|--|-----|------|
| f_{MCLK} | System Clock Frequency* | 1 | - | 110 | MHz |
| t_{MCLK} | System Clock Period* | $1/f_{MCLK}$ | - | - | ns |
| t_{PWCSH} | Control Pulse High Width Write Read | 13 30 | $1.5 * t_{MCLK}$ $3.5 * t_{MCLK}$ | - | ns |
| t_{PWCSL} | Control Pulse Low Width Write (next write cycle) Write (next read cycle) Read | 13 80 80 | $1.5 * t_{MCLK}$ $9 * t_{MCLK}$ $9 * t_{MCLK}$ | - | ns |
| t_{AS} | Address Setup Time | 1 | - | - | ns |
| t_{AH} | Address Hold Time | 2 | - | - | ns |
| t_{DSW} | Write Data Setup Time | 4 | - | - | ns |
| t_{DHW} | Write Data Hold Time | 1 | - | - | ns |
| t_{PWLW} | Write Low Time | 12 | - | - | ns |
| t_{DHR} | Read Data Hold Time | 1 | - | - | ns |
| t_{ACC} | Access Time | 32 | - | - | ns |
| t_{PWLW} | Read Low Time | 36 | - | - | ns |
| t_R | Rise Time | - | - | 0.5 | ns |
| t_F | Fall Time | - | - | 0.5 | ns |
| t_{CS} | Chip select setup time | 2 | - | - | ns |
| t_{CSH} | Chip select hold time to read signal | 3 | - | - | ns |

* System Clock denotes external input clock (PLL-bypass) or internal generated clock (PLL-enabled)

Figure 13-3: Parallel 8080-series Interface Timing Diagram (Write Cycle)

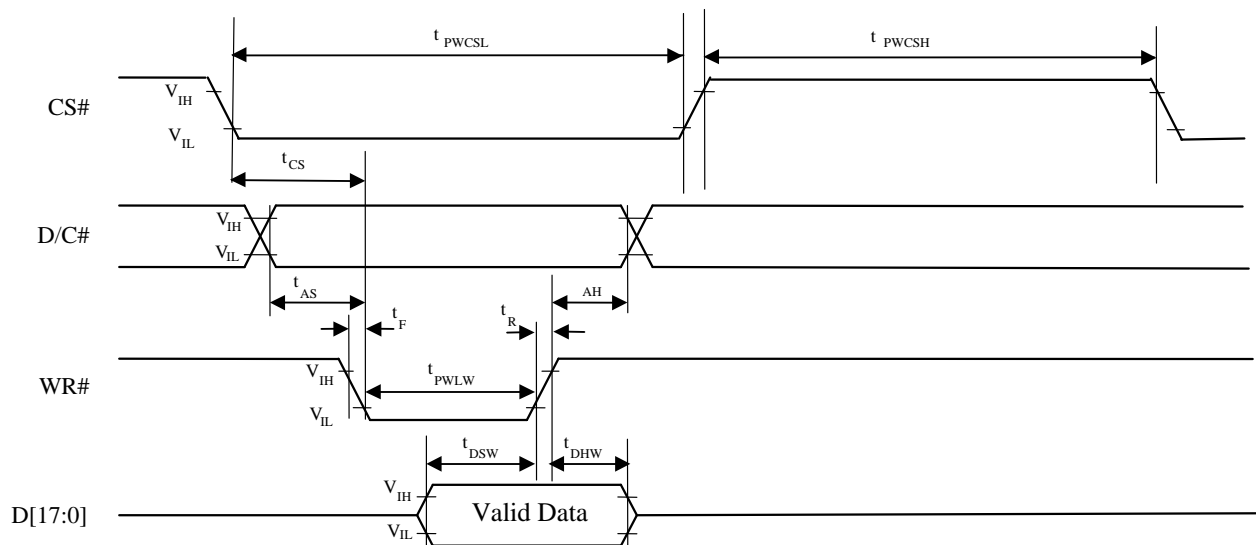
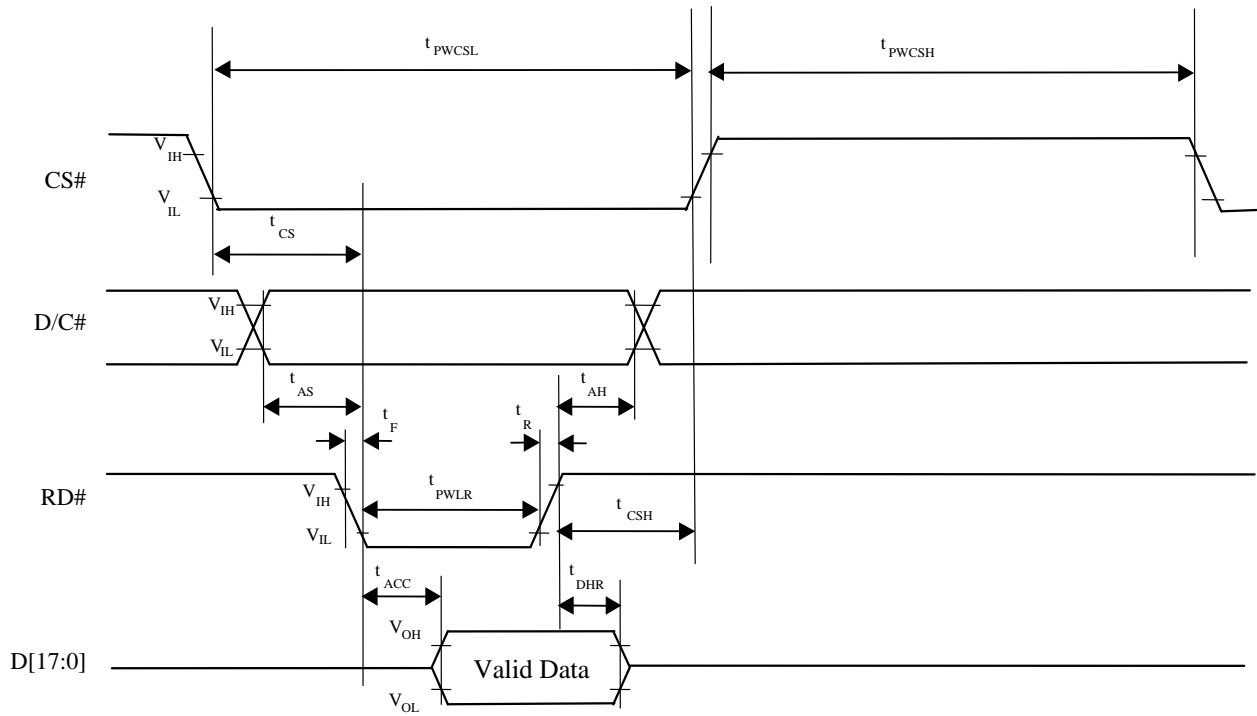
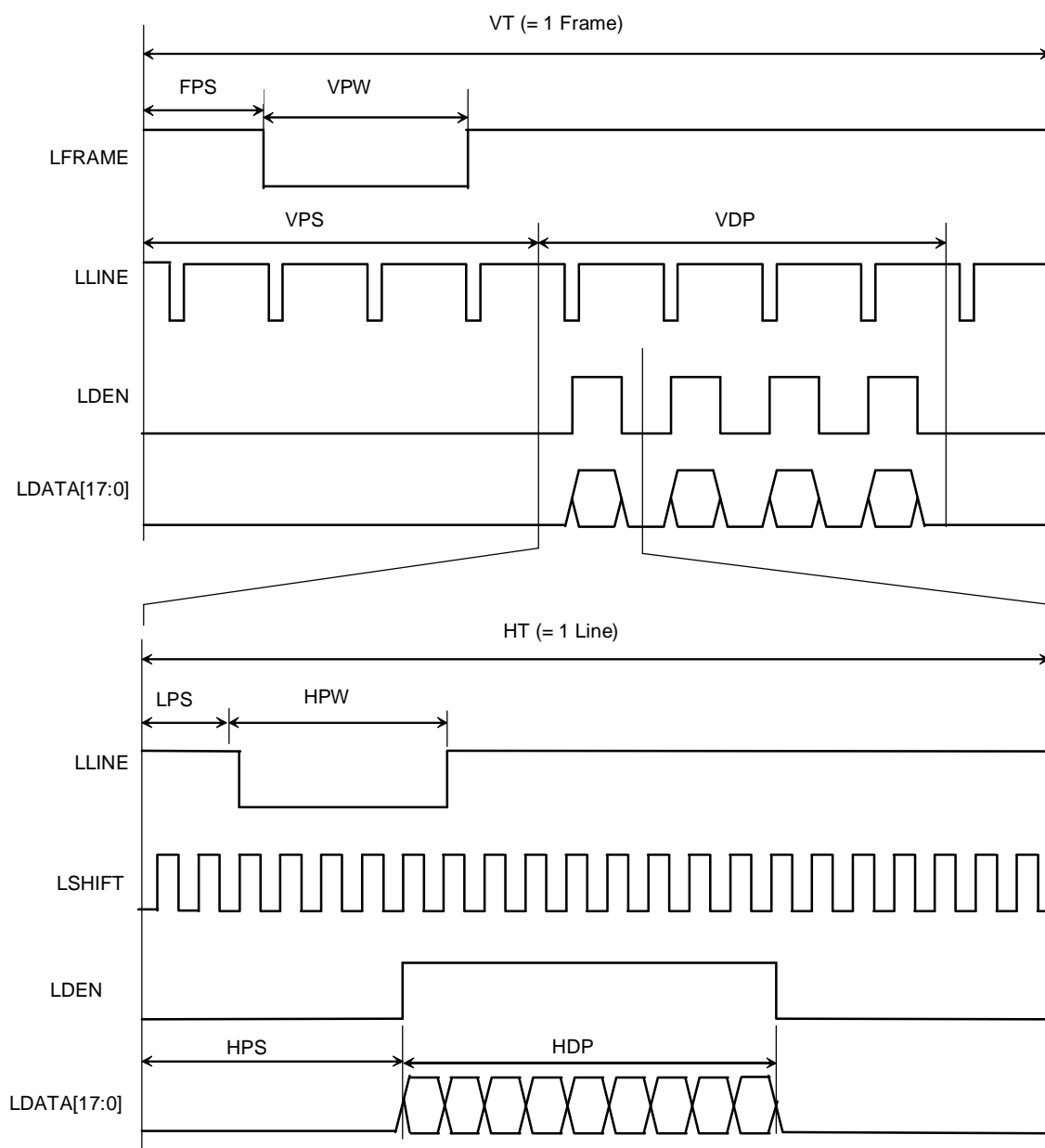


Figure 13-4: Parallel 8080-series Interface Timing Diagram (Read Cycle)



13.3 Parallel LCD Interface Timing

Figure 13-5: Generic TFT Panel Timing



13.4 Serial RGB Interface Timing

Figure 13-6: Serial RGB Interface Timing (without dummy mode)

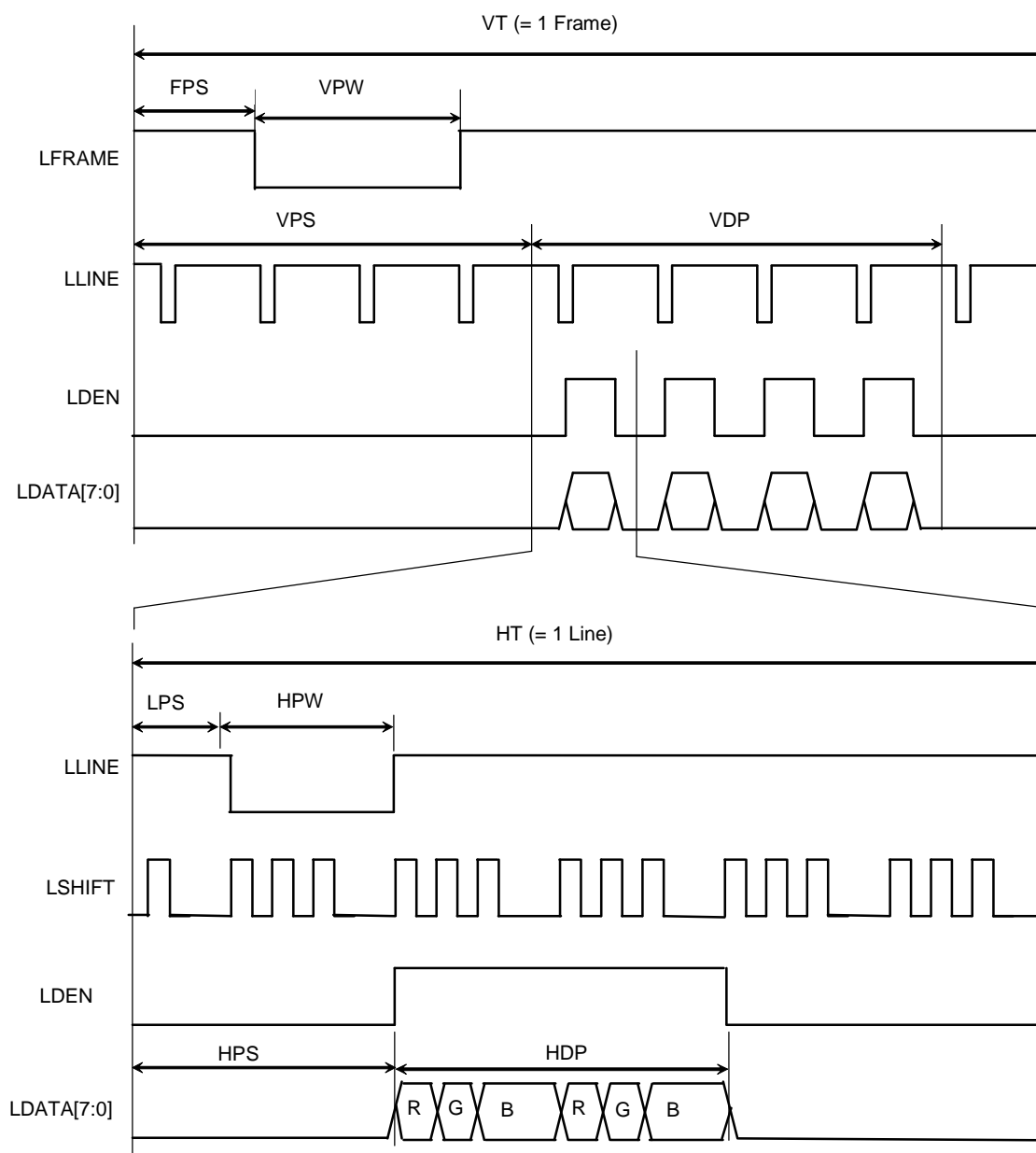


Figure 13-7: Serial RGB Interface Timing (with dummy mode)

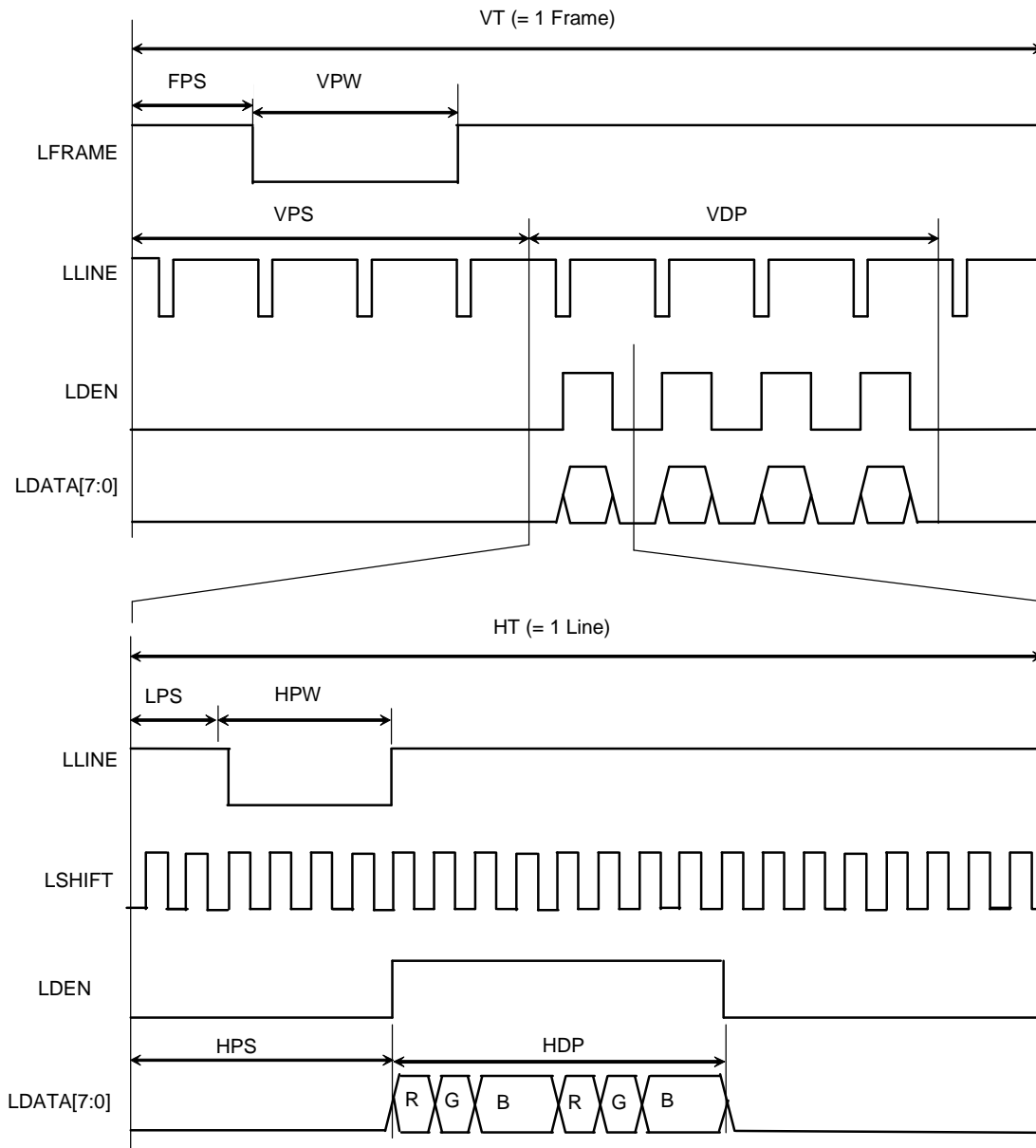


Table 13-7: Quick reference table for LCD parameter setting

| LCD Parameter setting | Corresponding Command | Register | Maximum Setting | |
|-----------------------|-----------------------|-----------|--|--------------------|
| | | | Parallel mode | Serial mode |
| VDP | 0xB0 | VDP[10:0] | 2048 Vertical panel size = (VDP + 1) pixels | |
| HDP | | HDP[10:0] | 2048 Horizontal panel size = (HDP + 1) pixels | |
| LPS | 0xB4 | LPS[10:0] | 2047 | 8188 (2047 x 4) |
| HPW | | HPW[6:0] | 128 (HPW + 1) pixels | 512 (128 x 4) |
| HPS | | HPS[10:0] | 2047 | 8188 (2047 x 4) |
| HT | | HT[10:0] | 2048 Horizontal Total = (HT + 1) lines | |
| FPS | 0xB6 | FPS[10:0] | 2047 | |
| VPW | | VPW[6:0] | 128 Vertical Sync Pulse Width = (VPW+1) lines | |
| VPS | | VPS[10:0] | 2047 | |
| VT | | VT[10:0] | 2048 Vertical Total = (VT + 1) lines | |

14 APPLICATION EXAMPLE

Figure 14-1 : Application circuit for SSD1963 (With Direct clock input)

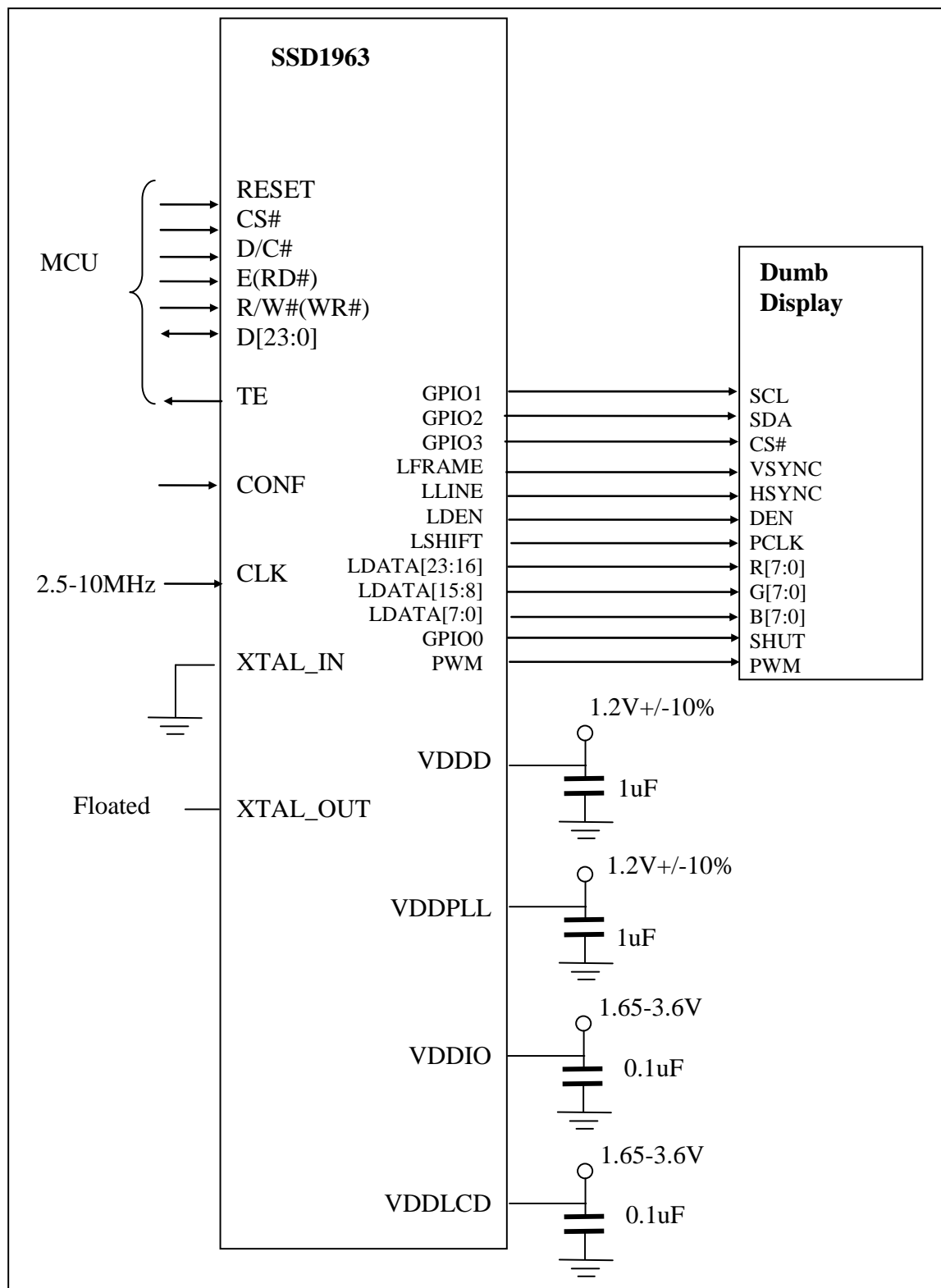
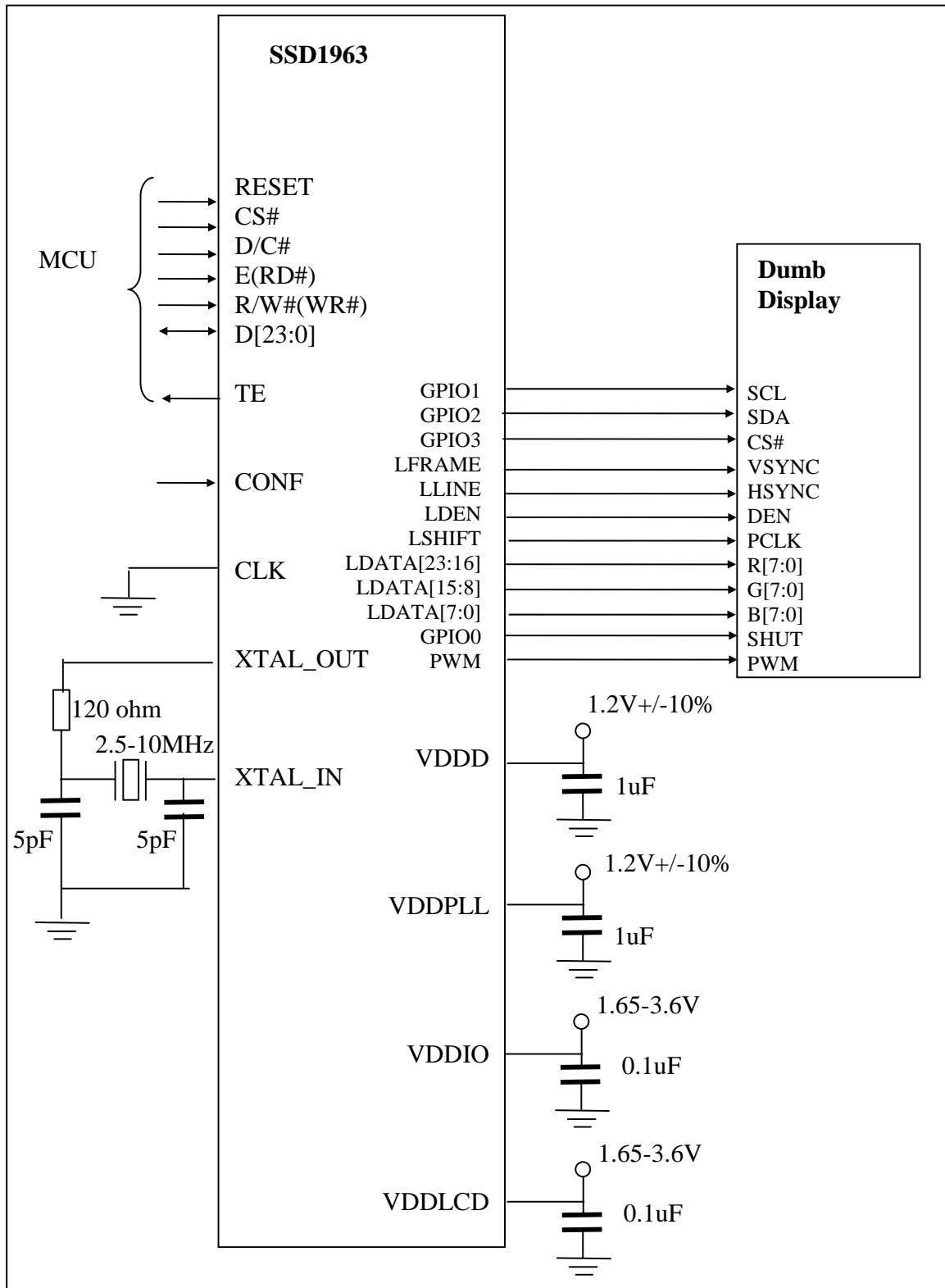
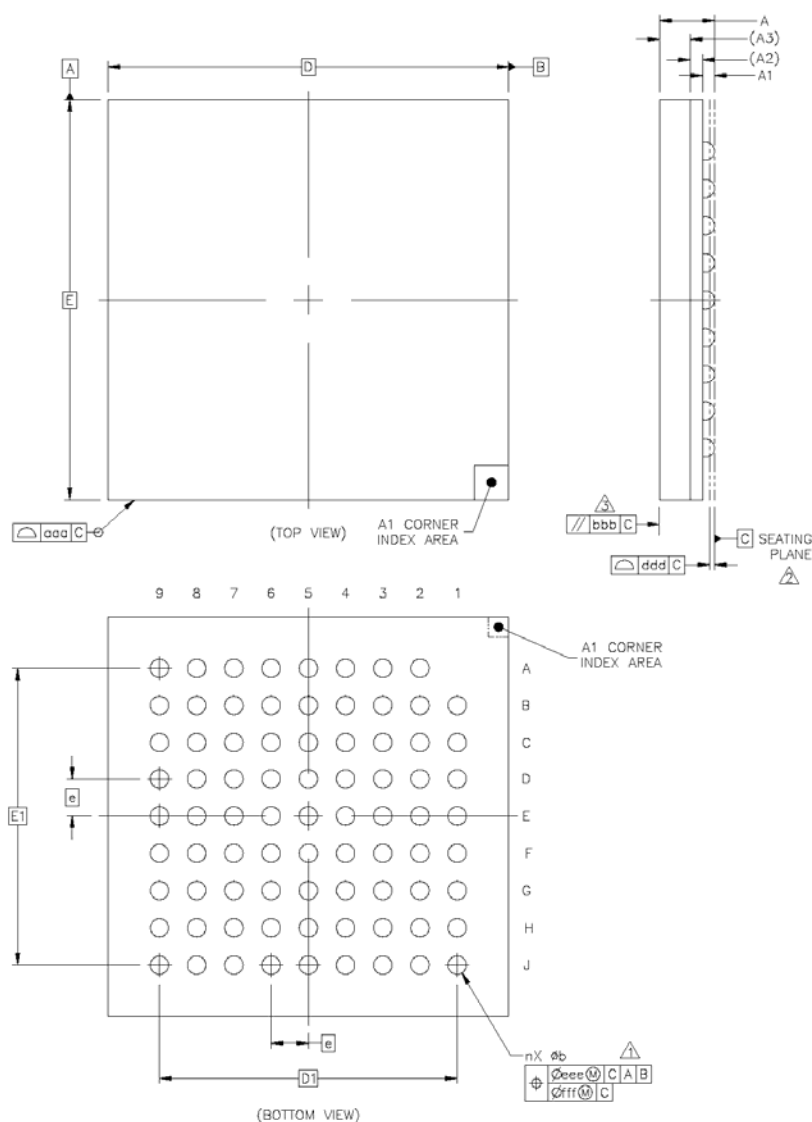


Figure 14-2 : Application circuit for SSD1963 (With crystal oscillator input)



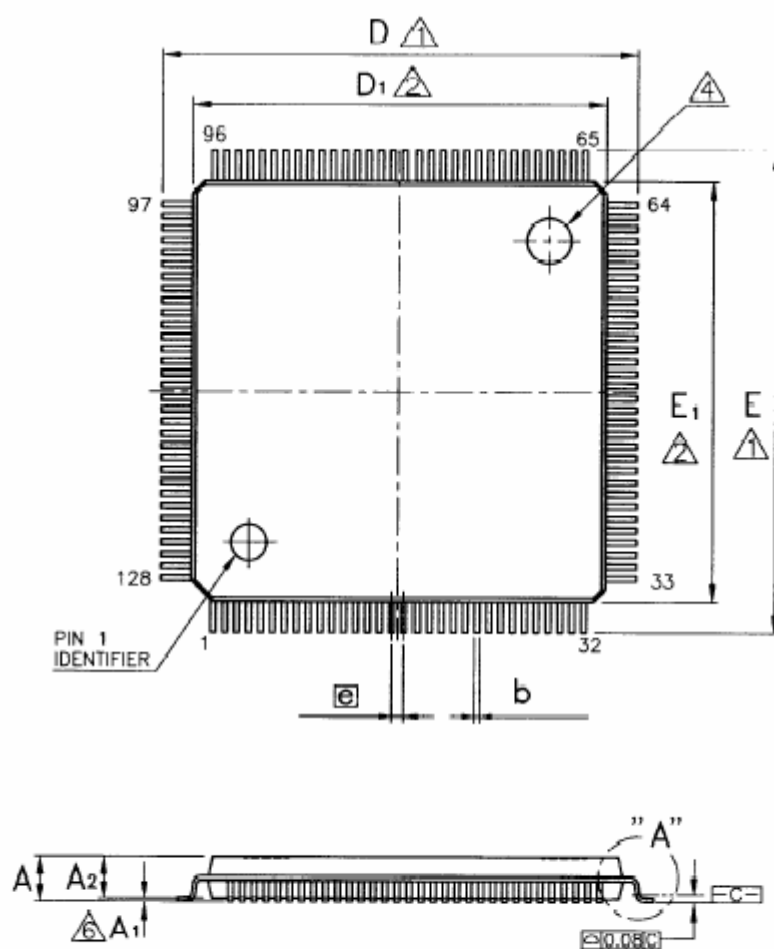
15 PACKAGE INFORMATION

15.1 Package Mechanical Drawing for 80 balls TFBGA



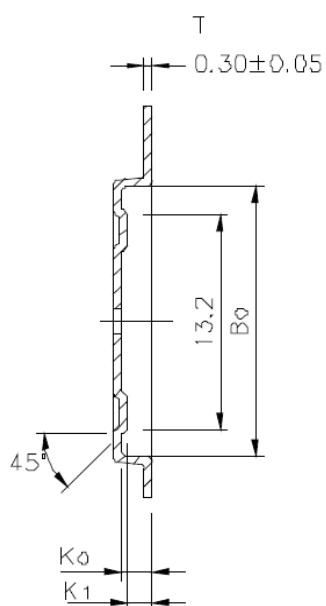
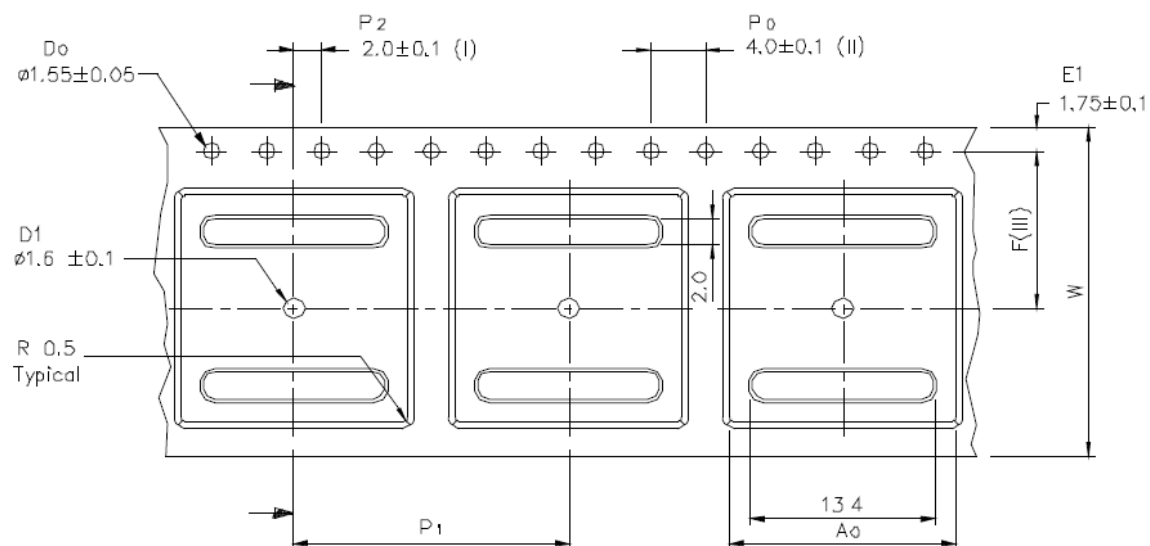
| Dimension in mm | | | |
|-----------------|------|----------|------|
| Symbol | Min | Typical | Max |
| A | --- | --- | 1.1 |
| A1 | 0.16 | --- | 0.26 |
| A2 | --- | 0.21 | --- |
| A3 | --- | 0.54 | --- |
| b | 0.27 | --- | 0.37 |
| D | --- | 7.00 BSC | --- |
| E | --- | 7.00 BSC | --- |
| e | --- | 0.65 BSC | --- |
| D1 | --- | 5.2 BSC | --- |
| E1 | --- | 5.2 BSC | --- |
| aaa | --- | 0.1 | --- |
| bbb | --- | 0.2 | --- |
| ddd | --- | 0.08 | --- |
| eee | --- | 0.15 | --- |
| fff | --- | 0.08 | --- |

15.2 Package Mechanical Drawing for 128 pins LQFP



| Symbol | Dimension in mm | | |
|--------|-----------------|----------|------|
| | Min | Nom | Max |
| A | | | 1.60 |
| A1 | 0.05 | | |
| A2 | | 1.40 | |
| D | | 16.00 | |
| D1 | | 14.00 | |
| E | | 16.00 | |
| E1 | | 14.00 | |
| e | | 0.40 BSC | |
| b | | 0.18 | |

15.3 Tape & Reel Drawing for 128 pins LQFP



| | |
|----------------|------------------|
| A ₀ | 16.50 +/− 0.1 |
| B ₀ | 16.50 +/− 0.1 |
| K ₀ | 1.90 +/− 0.1 |
| K ₁ | 1.55 +/− 0.1 |
| F | 11.50 +/− 0.1 |
| P ₁ | 20.00 +/− 0.1 |
| W | 24.00 +0.3/−0.00 |

- (I) Measured from centreline of sprocket hole to centreline of pocket.
 (II) Cumulative tolerance of 10 sprocket holes is ± 0.20 .
 (III) Measured from centreline of sprocket hole to centreline of pocket.
 (IV) Other material available.

ALL DIMENSIONS IN MILLIMETRES UNLESS OTHERWISE STATED.

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.



The product(s) listed in this datasheet comply with Directive 2002/95/EC of the European Parliament and of the council of 27 January 2004 on the restriction of the use of certain hazardous substances in electrical and electronic equipment and People's Republic of China Electronic Industry Standard SJ/T 11363-2006 "Requirements for concentration limits for certain hazardous substances in electronic information products (电子信息产品中有毒有害物质的限量要求)". Hazardous Substances test report is available upon request.

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