

SSD1926

Advance Information

JPEG Coder
SD interface
256K Embedded Display SRAM
Image Processor
CMOS

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



CONTENTS

| | | |
|-----------|--|-----------|
| 1 | GENERAL DESCRIPTION | 6 |
| 2 | FEATURES..... | 6 |
| 2.1 | HARDWARE JPEG DECODER..... | 6 |
| 2.2 | 2D GRAPHIC ENGINE..... | 6 |
| 2.3 | LCD GRAPHIC CONTROLLER..... | 7 |
| 2.4 | LCD PANEL INTERFACE | 7 |
| 2.5 | HOST MCU INTERFACE | 7 |
| 2.6 | MMC/SD INTERFACE..... | 7 |
| 2.7 | I/O INTERFACE | 8 |
| 2.8 | MISCELLANEOUS | 8 |
| 2.9 | PACKAGE..... | 8 |
| 3 | ORDERING INFORMATION | 8 |
| 4 | BLOCK DIAGRAM | 9 |
| 5 | PIN ARRANGEMENT | 10 |
| 5.1 | 128 PIN LQFP | 10 |
| 6 | PIN DESCRIPTIONS | 12 |
| 6.1 | GLOBAL SIGNAL..... | 12 |
| 6.2 | MCU INTERFACE..... | 13 |
| 6.3 | DISPLAY INTERFACE..... | 14 |
| 6.4 | MMC/SD/SDIO INTERFACE | 15 |
| 6.5 | CONFIGURATION..... | 16 |
| 6.6 | MISCELLANEOUS | 16 |
| 6.7 | POWER AND GROUND | 16 |
| 6.8 | SUMMARY OF CONFIGURATION | 17 |
| 6.9 | HOST BUS INTERFACE PIN MAPPING | 18 |
| 6.10 | LCD INTERFACE PIN MAPPING..... | 18 |
| 6.11 | DATA BUS ORGANIZATION..... | 19 |
| 7 | FUNCTIONAL BLOCK DESCRIPTIONS..... | 20 |
| 7.1 | PHASE LOCK LOOP (PLL)..... | 20 |
| 7.2 | EMBEDDED MEMORY | 20 |
| 7.3 | MCU INTERFACE..... | 21 |
| 7.3.1 | <i>Generic #1 addressing Mode</i> | 21 |
| 7.3.2 | <i>Generic #2 addressing Mode</i> | 22 |
| 7.3.3 | <i>8080 Indirect addressing Mode</i> | 23 |
| 7.4 | REGISTERS | 28 |
| 7.5 | JPEG DECODER..... | 28 |
| 7.6 | 2D ENGINE | 28 |
| 7.7 | DISPLAY INTERFACE..... | 28 |
| 7.8 | MMC/SD/SDIO INTERFACE | 28 |
| 7.9 | GENERAL PURPOSE INPUT/OUTPUT (GPIO) | 28 |
| 8 | MAXIMUM RATINGS | 29 |
| 9 | DC CHARACTERISTICS | 30 |
| 10 | AC CHARACTERISTICS | 31 |
| 10.1 | CLOCK TIMING | 31 |
| 10.1.1 | <i>Input Clocks</i> | 31 |
| 10.2 | CPU INTERFACE TIMING | 32 |

| | | |
|-----------|--|-----------|
| 10.2.1 | Generic #1 Interface Timing..... | 32 |
| 10.2.2 | Generic #2 Interface Timing (e.g. ISA)..... | 34 |
| 10.2.3 | 8080 Indirect Interface Timing..... | 36 |
| 11 | APPLICATION EXAMPLES | 37 |
| 11.1 | APPLICATION DIAGRAM | 37 |
| 12 | PSEUDO-CODE EXAMPLES FOR INDIRECT ADDRESS MODE..... | 41 |
| 12.1 | 8080 INDIRECT ADDRESS MODE | 41 |
| 13 | PACKAGE INFORMATION..... | 46 |
| 13.1 | PACKAGE MECHANICAL DRAWING FOR 128 PINS LQFP..... | 46 |

TABLES

| | |
|--|----|
| TABLE 3-1 : ORDERING INFORMATION | 8 |
| TABLE 5-1 : LQFP PIN ASSIGNMENT TABLE | 11 |
| TABLE 6-1 : HOST INTERFACE PIN DESCRIPTIONS | 12 |
| TABLE 6-2 : MCU INTERFACE PIN DESCRIPTIONS | 13 |
| TABLE 6-3 : DISPLAY INTERFACE PIN DESCRIPTIONS | 14 |
| TABLE 6-4 : MMC/SD/SDIO INTERFACE PIN DESCRIPTIONS | 15 |
| TABLE 6-5 : CONFIGURATION PIN DESCRIPTIONS | 16 |
| TABLE 6-6 : MISCELLANEOUS PIN DESCRIPTIONS | 16 |
| TABLE 6-7 : POWER AND GROUND PIN DESCRIPTIONS | 16 |
| TABLE 6-8 : SUMMARY OF CONFIGURATION PINS | 17 |
| TABLE 6-9 : HOST BUS INTERFACE PIN MAPPING | 18 |
| TABLE 6-10 : LCD INTERFACE PIN MAPPING | 18 |
| TABLE 6-11 : DATA BUS ORGANIZATION | 19 |
| TABLE 6-12 : PIN STATE SUMMARY | 19 |
| TABLE 8-1: ABSOLUTE MAXIMUM RATINGS | 29 |
| TABLE 8-2 : RECOMMENDED OPERATING CONDITIONS | 29 |
| TABLE 9-1 : ELECTRICAL CHARACTERISTICS FOR $IOV_{DD} = 3.3V$ TYPICAL | 30 |
| TABLE 10-1 : CLOCK INPUT REQUIREMENTS FOR CLKI | 31 |
| TABLE 10-2 : OSCILLATOR CLOCK INPUT REQUIREMENTS FOR CLKI2 | 31 |
| TABLE 10-3 : GENERIC #1 INTERFACE TIMING | 33 |
| TABLE 10-4 : GENERIC #2 INTERFACE TIMING | 35 |
| TABLE 10-5 : 8080 INTERFACE TIMING | 36 |

FIGURES

| | |
|--|----|
| FIGURE 4-1 : SSD1926 BLOCK DIAGRAM..... | 9 |
| FIGURE 5-1 : PINOUT DIAGRAM – 128 PIN LQFP (TOPVIEW)..... | 10 |
| FIGURE 7-1 : CIRCUIT FOR PLL ENABLE | 20 |
| FIGURE 7-2 : GENERIC #1 INTERFACE TIMING | 21 |
| FIGURE 7-3 : GENERIC #2 INTERFACE TIMING | 22 |
| FIGURE 7-4 : 8080 16 BIT INTERFACE TIMING (WRITE CYCLE)..... | 24 |
| FIGURE 7-5 : 8080 16 BIT INTERFACE TIMING (READ CYCLE) | 25 |
| FIGURE 7-6 : 8080 8 BIT INTERFACE TIMING (WRITE CYCLE)..... | 26 |
| FIGURE 7-7 : 8080 8 BIT INTERFACE TIMING (READ CYCLE) | 27 |
| FIGURE 10-1 : GENERIC #1 INTERFACE TIMING | 32 |
| FIGURE 10-2 : GENERIC #2 INTERFACE TIMING | 34 |
| FIGURE 10-3 : 8080 INTERFACE TIMING | 36 |
| FIGURE 11-1 : TYPICAL SYSTEM DIAGRAM (GENERIC #1 BUS) | 37 |
| FIGURE 11-2 : TYPICAL SYSTEM DIAGRAM (GENERIC #2 BUS) | 38 |
| FIGURE 11-3 : TYPICAL SYSTEM DIAGRAM (INDIRECT 8080 16 BIT BUS)..... | 39 |
| FIGURE 11-4: TYPICAL SYSTEM DIAGRAM (GENERIC #2 BUS)..... | 40 |

1 GENERAL DESCRIPTION

SSD1926 is an image processor designed for advanced car AV device with image capture and process features.

The image files can be saved into SD/MMC card through SD interface. The JPEG file is retrieved back from SD/MMC card, decoded and displayed on LCD panel through LCD interface. This interface supports various kinds of LCD panel like STN, CSTN and TFT.

The LCD controller of SSD1926 supports LCD panel for mobile phone with size, for example, 176x220 and 240x160 resolution at color depth 1, 2, 4, 8, 16 and 32 bit-per-pixel (bpp). For 16 and 32 bpp, SSD1926 provides 2D graphics acceleration features like virtual display, image rotation, cursor display, line drawing, BitBLT with raster operation, color fill, color expansion etc.

SSD1926 is able to interface different type of generic microcontrollers that are popular in handheld devices market. It also support indirect addressing mode which can minimize the pin count of control signals.

Internal PLLs is built such that only single clock is required for SSD1926 to generate clocks for blocks with various clock speed requirement.

With advanced power management design, SSD1926 is suitable for low power consumption and advanced image applications etc. The SSD1926 is available in LQFP package.

2 FEATURES

The main features of the SSD1926 are as follows:

2.1 Hardware JPEG decoder

- Hardware decoder to decode JPEG image with variable size up to 1280 x 1024.
- JPEG decoder is consisted of the following hardware module
 - a. Discrete Cosine Transform (DCT) and Inverse Discrete Cosine Transform (iDCT)
 - b. Quantization calculation with table downloadable by software
 - c. Zigzag and run-length coding
 - d. Huffman decoding with table downloadable by software
- For viewing JPEG image on LCD panel, the JPEG decoder can decimate and crop the image such that the length is in multiple of 8.

2.2 2D Graphic Engine

- Screen panning and scrolling – virtual display mode
- Image rotation including 0, 90, 180, 270 degree
- Two cursors with three colors and transparency selection. Cursor blinking is available
- Line drawing
- Rectangle drawing
- Ellipse drawing

- Bit block transfer (BitBLT)
 - a. Host to frame buffer
 - b. Frame buffer to frame buffer
 - c. Total 256 three-operand raster operations (ROP3) working with BitBLT
 - d. Pattern BitBLT: Source image is repeatedly filled up destination block
 - e. Stretch BitBLT: Stretch the source image to a destination larger or smaller than the source
 - f. Color Expansion: Monochrome color is expanded to either background or foreground color.
 - g. Color Fill: Fill a rectangular block with a single color.

2.3 LCD Graphic Controller

- Support 1, 2, 4, 8, 16 and 32 bit-per-pixel (bpp) color depth
- In 32bpp mode, each pixel is consisted of 8-bit red, 8-bit green, 8-bit blue and 8-bit alpha channel for controlling the transparency of the image.
- In 1, 2, 4, 8bpp mode, it can display still image and has no 2D graphic engine feature available.
- Arbitrary image size supported up to horizontal resolution of 512

2.4 LCD Panel Interface

- Support the following type of LCD panels:
 - a. Monochrome and color STN 4/8/12/16 bit interface
 - b. TFT 9/12/18/24 bit interface
 - c. 18 bit HR-TFT interface
 - d. 8 bit Serial TFT interface
 - e. 8 bit Delta panel with sub-pixel accuracy algorithm
 - f. Support Smart LCD panels through SPI and 8-bit MCU (8080, 6800) interface
- For STN and CSTN panel, spatial and dynamic dithering is available to increase color depth.
 - a. 16 gray shades for each color component when applying frame rate control only
 - b. 64 gray shades for each color component when applying frame rate control and dithering
- LCD panel power on and off sequencing

2.5 Host MCU interface

- Support the following MCU interface
 - a. SRAM interface (e.g. generic ARM core type MCU)
 - b. ISA interface for MCU like NEC MIPS
 - c. 8/16 bits 8080 indirect addressing mode
- Support synchronous and asynchronous interface communication
- Memory mapped I/O
- Big/Little endian support

2.6 MMC/SD Interface

- Compatible with “The MultiMedia Card System Specification version 3.0”
- Compatible with “SD Memory Card Specification version 1.0” and “SDIO Card Specification version 1.0”
- Block transfer from/to external host

- Block transfer from/to internal memory
- Supports many SD functions including multiple I/O and combined I/O and memory

2.7 I/O Interface

- 13 GPIOs

2.8 Miscellaneous

- Embedded 256K bytes SRAM
- Single clock input
- Integrated PLL
- Advanced power management to cut off the power for modules that are idle.

2.9 Package

- 128-pin LQFP package

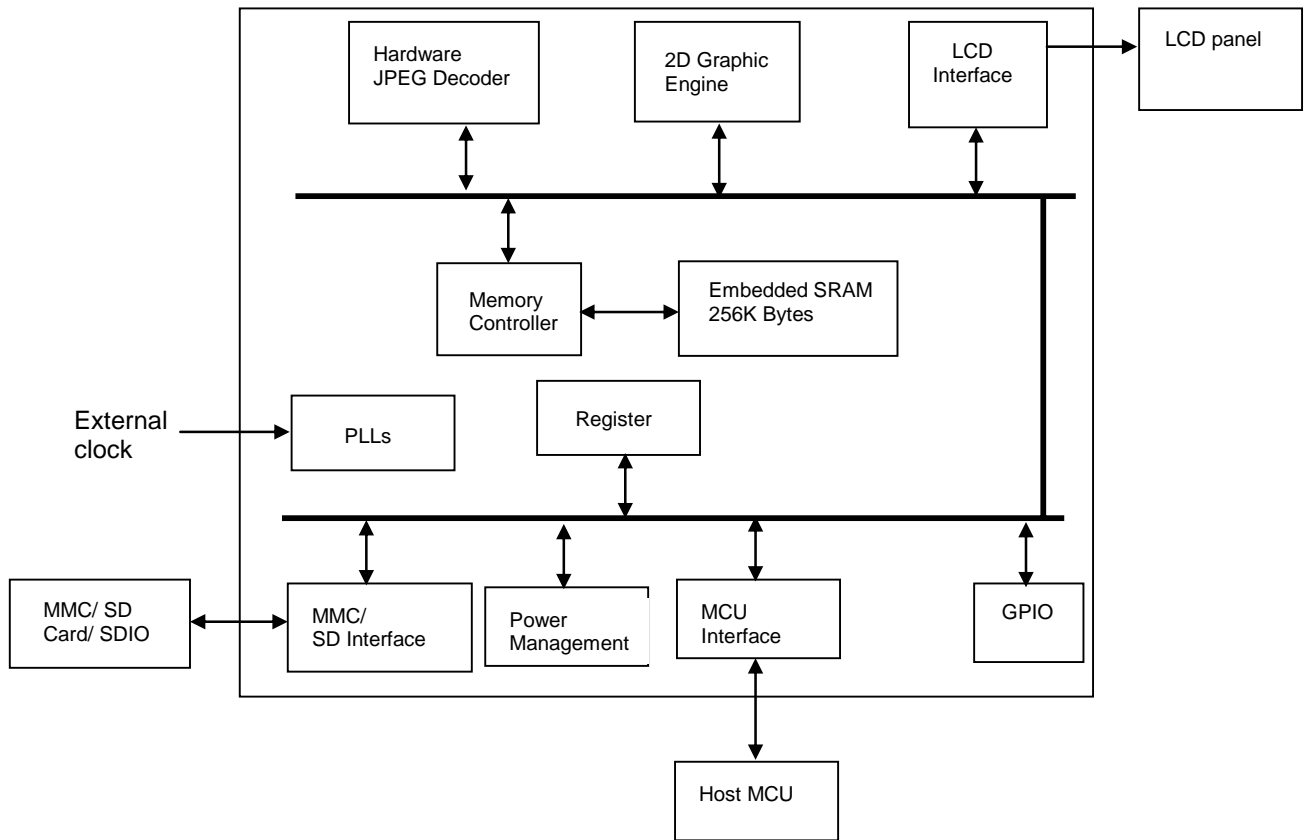
3 ORDERING INFORMATION

Table 3-1 : Ordering Information

| Ordering Part Number | Package Form |
|-----------------------------|---------------------|
| SSD1926QL9 | 128 LQFP |

4 BLOCK DIAGRAM

Figure 4-1 : SSD1926 Block Diagram



5 PIN ARRANGEMENT

5.1 128 pin LQFP

Figure 5-1 : Pinout Diagram – 128 pin LQFP (Topview)

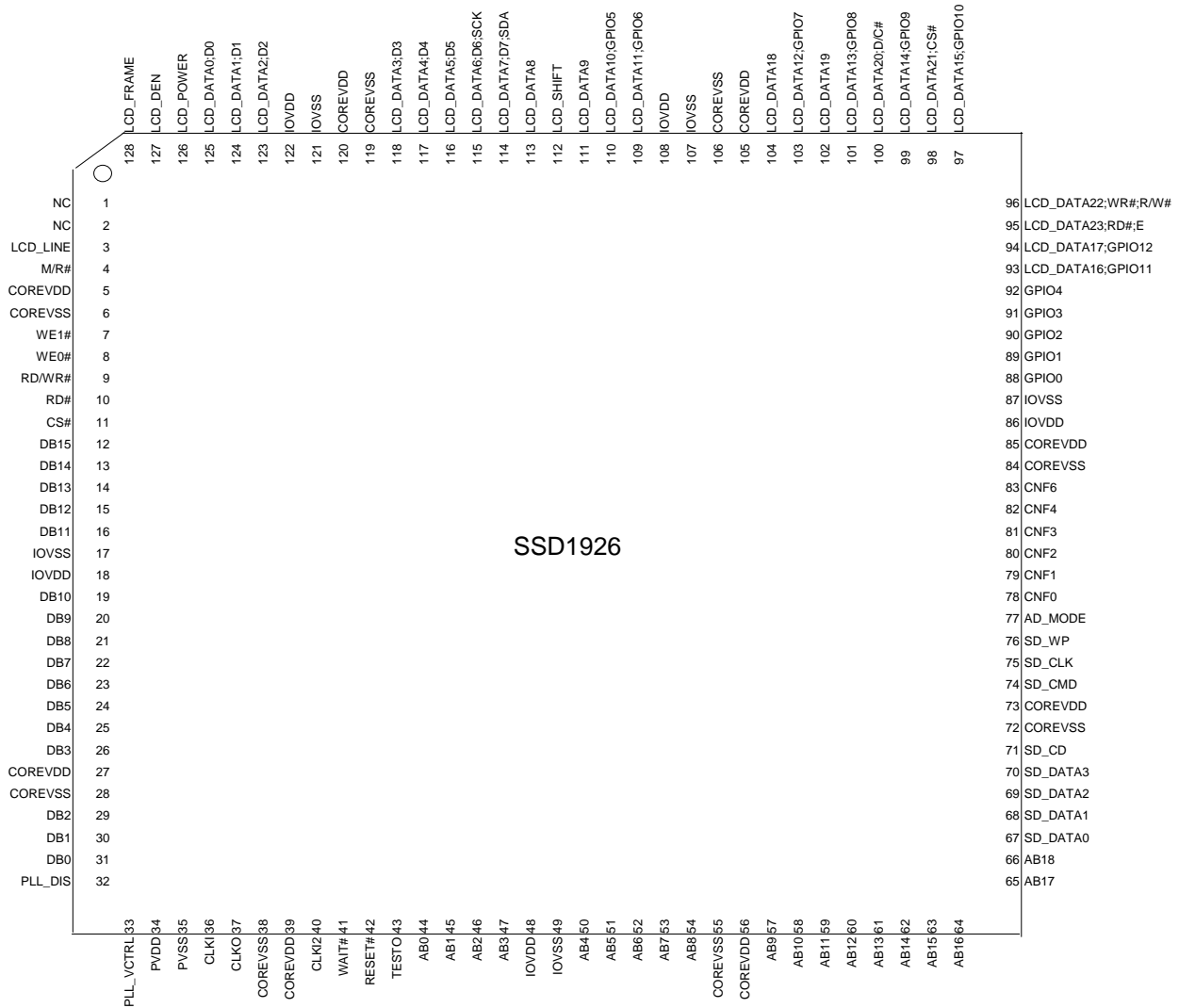


Table 5-1 : LQFP Pin Assignment Table

| Pin # | Signal Name | Pin # | Signal Name | Pin # | Signal Name | Pin # | Signal Name |
|-------|---------------------|-------|---------------------|-------|------------------------|-------|-----------------------|
| 1 | NC | 33 | PLL_VCTRL | 65 | AB17 | 97 | LCD_DATA15; GPIO10 |
| 2 | NC | 34 | PV _{DD} | 66 | AB18 | 98 | LCD_DATA21; CS# |
| 3 | LCD_LINE | 35 | PV _{SS} | 67 | SD_DATA0 | 99 | LCD_DATA14; GPIO9 |
| 4 | M/R# | 36 | CLKI | 68 | SD_DATA1 | 100 | LCD_DATA20; D/C# |
| 5 | COREV _{DD} | 37 | CLKO | 69 | SD_DATA2 | 101 | LCD_DATA13; GPIO8 |
| 6 | COREV _{SS} | 38 | COREV _{SS} | 70 | SD_DATA3 | 102 | LCD_DATA19 |
| 7 | WE1# | 39 | COREV _{DD} | 71 | SD_CD | 103 | LCD_DATA12; GPIO7 |
| 8 | WE0# | 40 | CLKI2 | 72 | COREV _{SS} | 104 | LCD_DATA18 |
| 9 | RD/WR# | 41 | WAIT# | 73 | COREV _{DD} | 105 | COREV _{DD} |
| 10 | RD# | 42 | RESET# | 74 | SD_CMD | 106 | COREV _{SS} |
| 11 | CS# | 43 | TESTO | 75 | SD_CLK | 107 | IOV _{SS} |
| 12 | DB15 | 44 | AB0 | 76 | SD_WP | 108 | IOV _{DD} |
| 13 | DB14 | 45 | AB1 | 77 | AD_MODE | 109 | LC_DATA11; GPIO6 |
| 14 | DB13 | 46 | AB2 | 78 | CNF0 | 110 | LCD_DATA10; GPIO5 |
| 15 | DB12 | 47 | AB3 | 79 | CNF1 | 111 | LCD_DATA9 |
| 16 | DB11 | 48 | IOV _{DD} | 80 | CNF2 | 112 | LCD_SHIFT |
| 17 | IOV _{SS} | 49 | IOV _{SS} | 81 | CNF3 | 113 | LCD_DATA8 |
| 18 | IOV _{DD} | 50 | AB4 | 82 | CNF4 | 114 | LCD_DATA7;D7; SDA |
| 19 | DB10 | 51 | AB5 | 83 | CNF6 | 115 | LCD_DATA6;D6; SCK |
| 20 | DB9 | 52 | AB6 | 84 | COREV _{SS} | 116 | LCD_DATA5;D5 |
| 21 | DB8 | 53 | AB7 | 85 | COREV _{DD} | 117 | LCD_DATA4;D4 |
| 22 | DB7 | 54 | AB8 | 86 | IOV _{DD} | 118 | LCD_DATA3;D3 |
| 23 | DB6 | 55 | COREV _{SS} | 87 | IOV _{SS} | 119 | COREV _{SS} |
| 24 | DB5 | 56 | COREV _{DD} | 88 | GPIO0 | 120 | COREV _{DD} |
| 25 | DB4 | 57 | AB9 | 89 | GPIO1 | 121 | IOV _{SS} |
| 26 | DB3 | 58 | AB10 | 90 | GPIO2 | 122 | IOV _{DD} |
| 27 | COREV _{DD} | 59 | AB11 | 91 | GPIO3 | 123 | LCD_DATA2;D2 |
| 28 | COREV _{SS} | 60 | AB12 | 92 | GPIO4 | 124 | LCD_DATA1;D1 |
| 29 | DB2 | 61 | AB13 | 93 | LCD_DATA16; GPIO11 | 125 | LCD_DATA0;D0 |
| 30 | DB1 | 62 | AB14 | 94 | LCD_DATA17; GPIO12 | 126 | LCD_POWER |
| 31 | DB0 | 63 | AB15 | 95 | LCD_DATA23; RD#;E | 127 | LCD_DEN |
| 32 | PLL_DIS | 64 | AB16 | 96 | LC_DATA22; WR#;R/W# | 128 | LCD_FRAME |

6 PIN DESCRIPTIONS

Key:

I = Input
 O = Output
 IO = Bi-directional (input / output)
 P = Power pin
 AN = Analog
 LIS = LVCMOS Schmitt input
 LB2 = LVCMOS IO buffer (8mA/-8mA at 3.3V)
 LB3 = LVCMOS IO buffer (16mA/-16mA at 3.3V)
 LO1 = LVCMOS output buffer (2mA/-2mA at 3.3V)
 LO2 = LVCMOS output buffer (4mA/-4mA at 3.3V)
 LO3 = LVCMOS output buffer (16mA/-16mA at 3.3V)
 LT2 = Tri-state output buffer (8mA/-8mA at 3.3V)
 Hi-Z = High impedance

6.1 Global Signal

Table 6-1 : Host Interface Pin Descriptions

| Pin Name | Type | LQFP Pin # | Cell | RESET# State | Description |
|------------|------|------------|------|--------------|---|
| CLKI2 | I | 40 | LIS | - | This pin can be used as clock source input when PLL is disabled. When synchronized MCU interface is selected, connect the bus clock to CLKI2 and disable the PLL by connecting the PLL_DIS pin to IOV _{DD} . If the PLL is enabled, CLKI2 has to be pull-up or pull-down. |
| CLKI, CLKO | IO | 36, 37 | AN | - | These two pins are the source clock of internal PLL. It accepts clock frequency from 2MHz to 4MHz. The clock source can be either oscillator or crystal. If 4-pin oscillator or a slow clock source is available, please connect the output of oscillator or the clock source to CLKI and leave the CLKO pin floating. If CLKI2 is used as clock source, CLKI has to be pull-up or pull-down and leave CLKO floating. |
| PLL_DIS | I | 32 | LIS | - | PLL disable control pin. PLL_DIS = IOV _{DD} , PLL disabled (When PLL is disabled, the master clock is directly fed from CLKI2 pin) PLL_DIS = IOV _{SS} , PLL enabled |
| PLL_VCTRL | I | 33 | AN | - | Control for PLL. If internal PLL is selected, RC circuit should be connected. Refer to 7.1. Leave this pin floating if PLL is disabled. |
| RESET# | I | 42 | LIS | - | Master chip reset. Active low input to set all internal registers to the default state and to force all signals to their inactive states. It is recommended to place a 0.1µF capacitor to V _{SS} . Note (1) When reset state is released (RESET# = "H"), normal operation can be started after 3 MCLK period. |

6.2 MCU Interface

Table 6-2 : MCU Interface Pin Descriptions

| Pin Name | Type | LQFP Pin # | Cell | RESET# State | Description |
|----------------|------|---------------------|------|--------------|--|
| AB0 | I | 44 | LIS | 0 | This input pin has multiple functions. <ul style="list-style-type: none"> For Generic #1, this pin is not used and should be connected to V_{SS}. For Generic #2, this is an input of system address bit 0 (A0). For 8080, this pin is not used and should be connected to V_{SS}. |
| AB[18:4, 2, 1] | I | 45-46, 50-54, 57-66 | LIS | 0 | System address bus bits 18-4, 2, 1 for direct address mode. For 8080, those pins are not used and should be connected to V _{SS} . |
| AB[3] | I | 47 | LIS | 0 | This input pin has multiple functions. <ul style="list-style-type: none"> System address bus bit 3 for direct address mode For 8080, this pin is used as data / command select, D/C#. |
| DB[15:0] | IO | 12-16, 19-26, 29-31 | LB2 | Hi-Z | Bi-directional system data bus 15:0. |
| WE0# | I | 8 | LIS | 1 | This input pin has multiple functions. <ul style="list-style-type: none"> For Generic #1, this is an input of the write enable signal for the lower data byte (WE0#). For Generic #2, this is an input of the write enable signal (WE#). For 8080, this is an input of write enable signal, WR# |
| WE1# | I | 7 | LIS | 1 | This input pin has multiple functions. <ul style="list-style-type: none"> For Generic #1, this is an input of the write enable signal for the upper data byte (WE1#). For Generic #2, this is an input of the byte enable signal for the high data byte (BHE#). For 8080, this pin is not used and should be connected to V_{SS}. |
| CS# | I | 11 | LIS | 1 | Chip select input. |
| M/R# | I | 4 | LIS | 0 | <ul style="list-style-type: none"> For 8080, this pin is not used and should be connected to V_{SS}. For other interfaces, this input pin is used to select the display buffer or internal registers of the SSD1926. M/R# is set high to access the display buffer and low to access the registers. |
| RD/WR# | I | 9 | LIS | 1 | This input pin has multiple functions. <ul style="list-style-type: none"> For Generic #1, this is an input of the read signal for the upper data byte (RD1#). For Generic #2, this pin must be tied to IOV_{DD}. For 8080, this pin is not used and should be connected to V_{SS}. |

| Pin Name | Type | LQFP Pin # | Cell | RESET# State | Description |
|----------|------|------------|------|--------------|---|
| RD# | I | 10 | LIS | 1 | This input pin has multiple functions. <ul style="list-style-type: none"> For Generic #1, this is an input of the read signal for the lower data byte (RD0#). For Generic #2, this is an input of the read command (RD#). For 8080, this is an input of read enable signal, RD# |
| WAIT# | O | 41 | LT2 | Hi-Z | During a data transfer, this output pin is driven active to force the system to insert wait states. It is driven inactive to indicate the completion of a data transfer. WAIT# is released to the high impedance state after the data transfer is complete. Its active polarity is configurable. A pull-up or pull-down resistor should be used to resolve any data contention issues. See Table 6-8 : Summary of Configuration . <ul style="list-style-type: none"> For Generic #1, this pin outputs the wait signal (WAIT#). For Generic #2, this pin outputs the wait signal (WAIT#). For 8080, this pin outputs the wait signal (WAIT#). |

See Table 6-9 : Host Bus Interface Pin Mapping for summary.

6.3 Display Interface

Table 6-3 : Display Interface Pin Descriptions

| Pin Name | Type | LQFP Pin # | Cell | RESET# State | Description |
|----------------------|------|-----------------|------|--------------|---|
| LCD_DATA[5:0];D[5:0] | O | 116-118,123-125 | LO3 | 0 | If RGB dump panel is selected, those pins are RGB dump data bits 5-0. If MCU smart parallel panel is selected, those pins are data bits 5-0 If MCU smart serial panel is selected, those signals are not in used. |
| LCD_DATA6;D6;SCK | O | 115 | LO3 | 0 | If RGB dump panel is selected, this pin is RGB dump data bit 6. If MCU smart parallel panel is selected, this pin is data bit 6 If MCU smart serial panel is selected, this pin becomes serial clock signal, SCK. |
| LCD_DATA7;D7;SDA | O | 114 | LO3 | 0 | If RGB dump panel is selected, this pin is RGB dump data bit 7. If MCU smart parallel panel is selected, this pin is data bit 7 If MCU smart serial panel is selected, this pin becomes serial data signal, SDA. |
| LCD_DATA8;D8 | O | 113 | LO3 | 0 | If RGB dump panel is selected, RGB dump data bits 8 If MCU smart parallel panel is selected, data bits 8 If MCU smart serial panel is selected, this signal is not in used. |
| LCD_DATA9 | O | 111 | LO3 | 0 | If RGB dump panel is selected, RGB dump data bit 9 If MCU smart panel is selected, those signals were not in used. |

| | | | | | |
|--------------------------------|----|-------------------------------------|-----|---|---|
| LCD_DATA[17:10]; GPIO[12:5] | IO | 93,94,97,99, 101,103,109, 110 | LB3 | 0 | If 18/24-bit dump TFT panel is selected, RGB dump data bits 17:10 If other dump and MCU smart panel is selected, these pins become GPIO control pins. |
| LCD_DATA[19:18] | O | 102,104 | LO3 | 0 | If RGB dump panel is selected, RGB dump data bits 19,18 If MCU smart panel is selected, those signals were not in used. |
| LCD_DATA20; D/C# | O | 100 | LO3 | 0 | If RGB dump panel is selected, RGB dump data bits 20 If MCU smart parallel or serial panel is selected, this signal is data/command select, D/C#. |
| LCD_DATA21;CS# | O | 98 | LO3 | 0 | If RGB dump panel is selected, RGB dump data bits 21 If MCU smart parallel or serial panel is selected, this signal is chip select, CS#. |
| LCD_DATA22;WR#; R/W# | O | 96 | LO3 | 0 | If RGB dump panel is selected, RGB dump data bits 22 If MCU smart parallel 8080 panel is selected, this signal is WR#. If MCU smart parallel 6800 panel is selected, this signal is R/W#. If MCU smart serial panel is selected, this signal is not in used. |
| LCD_DATA23;RD#; E | O | 95 | LO3 | 0 | If RGB dump panel is selected, RGB dump data bits 23 If MCU smart parallel 8080 panel is selected, this signal is RD#. If MCU smart parallel 6800 panel is selected, this signal is E. If MCU smart serial panel is selected, this signal is not in used. |
| LCD_FRAME | O | 128 | LO3 | 0 | Frame Pulse (vertical sync) |
| LCD_LINE | O | 3 | LO3 | 0 | Line Pulse (horizontal sync) |
| LCD_SHIFT | O | 112 | LO3 | 0 | Shift Clock |
| LCD_DEN | O | 127 | LO3 | 0 | This output pin has multiple functions. <ul style="list-style-type: none"> • Display enable (LDEN) for TFT panels • LCD backplane bias signal (MOD) for all other LCD panels |
| LPOWER | O | 126 | LO2 | 0 | Power control for LCD panel. |

See Table 6-10 : LCD Interface Pin Mapping for summary.

6.4 MMC/SD/SDIO Interface

Table 6-4 : MMC/SD/SDIO Interface Pin Descriptions

| Pin Name | Type | LQFP Pin # | Cell | RESET# State | Description |
|--------------|------|------------|------|--------------|---|
| SD_CLK | O | 75 | LO3 | 0 | SD clock |
| SD_CMD | IO | 74 | LB2 | - | SD command |
| SD_DATA[3:0] | IO | 67-70 | LB2 | - | SD data[3:0] SD_DATA[3:1] are not used for 1 bit SD or MMC |
| SD_CD | I | 71 | LIS | - | SD card inserted |
| SD_WP | I | 76 | LIS | - | SD card write protected |

6.5 Configuration

Table 6-5 : Configuration Pin Descriptions

| Pin Name | Type | LQFP Pin # | Cell | RESET # State | Description |
|----------------------------|------|------------|------|---------------|--|
| CNF[6,4:0], AD_MOD E | I | 77-83 | LIS | — | These inputs are used to configure the SSD1926 – see Table 6-8 : Summary of Configuration pins. Note ⁽¹⁾ These pins are used for configuration of the SSD1926 and must be connected directly to IOV _{DD} or V _{SS} . |

6.6 Miscellaneous

Table 6-6 : Miscellaneous Pin Descriptions

| Pin Name | Type | LQFP Pin # | Cell | RESET # State | Description |
|-----------|------|------------|------|---------------|--|
| GPIO[4:0] | IO | 88-92 | LB3 | 0 | General Purpose IO. Those GPIO signals can be programmed as LCD control which sync with LCD signals. |
| TESTO | O | 43 | LO3 | 0 | Test output pin. Floated this pin in normal operation. |

6.7 Power and Ground

Table 6-7 : Power and Ground Pin Descriptions

| Pin Name | Type | LQFP Pin # | Cell | RESET # State | Description |
|---------------------|------|--------------------------------------|------|---------------|--|
| IOV _{DD} | P | 18,48, 86,108, 122 | P | — | 3.3V Power supply pins for I/O pads. It is recommended to place a 0.1μF bypass capacitor close to each of these pins. |
| IOV _{SS} | P | 17,49, 87,107, 121 | P | — | Ground pins for I/O pads |
| COREV _{DD} | P | 5,27,39, 56,73, 85,105, 120 | P | — | 1.8V Power supply pins for core. It is recommended to place a 0.1μF bypass capacitor close to each of these pins. |
| COREV _{SS} | P | 6,28,38, 55,72, 84,106, 119 | P | — | Ground pins for core |
| PV _{DD} | P | 34 | P | — | 1.8V Power supply pins for PLL. It is recommended to place a 0.1μF bypass capacitor close to each of these pins. |
| PV _{SS} | P | 35 | P | — | Ground pins for PLL |

6.8 Summary of Configuration

These pins are used for configuration of the SSD1926 and must be connected directly to IOV_{DD} or IOV_{SS}. The state of AD_MODE, CNF[6, 4:0] is latched on the rising edge of RESET# or after the software reset function is activated (REG[A2h] bit 0). Changing state at any other time has no effect.

Table 6-8 : Summary of Configuration pins

| SSD1926 Configuration Input | Power-On/Reset State | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------------------------------|--|--|---------|----------|---|------|----------|---|--|---|---|---|-----------|---|--|---|---|---|-----------|---|--|---|---|---|---|---|--|---|---|---|----------------------|
| | 1 (Connected to IOV _{DD}) | 0 (Connected to IOV _{SS}) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| CNF[2:0], AD_MODE | Select host bus interface as follows: <table border="1"> <thead> <tr> <th>AD_MODE</th> <th>CNF2</th> <th>CNF1</th> <th>CNF0</th> <th>Host Bus</th> </tr> </thead> <tbody> <tr> <td>0</td> <td></td> <td>0</td> <td>1</td> <td>1</td> <td>Generic#1</td> </tr> <tr> <td>0</td> <td></td> <td>1</td> <td>0</td> <td>0</td> <td>Generic#2</td> </tr> <tr> <td>1</td> <td></td> <td>0</td> <td>1</td> <td>1</td> <td>Indirect 8 bit 8080 (For Big Endian only)</td> </tr> <tr> <td>1</td> <td></td> <td>1</td> <td>0</td> <td>0</td> <td>Indirect 16 bit 8080</td> </tr> </tbody> </table> Note : The host bus interface is 18-bit only. | | AD_MODE | CNF2 | CNF1 | CNF0 | Host Bus | 0 | | 0 | 1 | 1 | Generic#1 | 0 | | 1 | 0 | 0 | Generic#2 | 1 | | 0 | 1 | 1 | Indirect 8 bit 8080 (For Big Endian only) | 1 | | 1 | 0 | 0 | Indirect 16 bit 8080 |
| AD_MODE | CNF2 | CNF1 | CNF0 | Host Bus | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | | 0 | 1 | 1 | Generic#1 | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | | 1 | 0 | 0 | Generic#2 | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | | 0 | 1 | 1 | Indirect 8 bit 8080 (For Big Endian only) | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | | 1 | 0 | 0 | Indirect 16 bit 8080 | | | | | | | | | | | | | | | | | | | | | | | | | | |
| CNF3 | Configure GPIO pins as inputs at power-on | Configure GPIO pins as outputs at power-on | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| CNF4 | Big Endian bus interface | Little Endian bus interface | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| CNF6 | MCLK = PLL_CLK / 4 | MCLK = PLL_CLK | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Note : Recommended to use CNF6 = 0 for Indirect addressing mode | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

6.9 Host Bus Interface Pin Mapping

Table 6-9 : Host Bus Interface Pin Mapping

| SSD1926 Pin Name | Generic #1 | Generic #2 | Indirect 8080 |
|------------------|--------------------------------|--------------------------------|--------------------------------|
| AB0 | Connected to IOV _{SS} | A0 | Connected to IOV _{SS} |
| AB[18:4, 2, 1] | A[18:4, 2, 1] | | Connected to IOV _{SS} |
| AB[3] | A[3] | | D/C# |
| DB[15:0] | D[15:0] | | D[15:0] |
| CS# | External Decode | | |
| M/R# | External Decode | | Connected to IOV _{SS} |
| CLKI2 (optional) | BUSCLK | | |
| RD/WR# | RD1# | Connected to IOV _{DD} | Connected to V _{SS} |
| RD# | RD0# | RD# | RD# |
| WE0# | WE0# | WE# | WR# |
| WE1# | WE1# | BHE# | Connected to IOV _{SS} |
| RESET# | RESET# | | |

6.10 LCD Interface Pin Mapping

Table 6-10 : LCD Interface Pin Mapping

| Pin Names | DUMB DRIVER | | | | | | | | | | SMART DRIVER | | | |
|------------|-------------|---------|---------------------|-----------------------|---------|---------|---------|---------|--------|--------------|-----------------|---------|---------|--------------|
| | Mono STN | | CSTN | | | TFT | | | | | TFT | CSTN | OLED | TFT(Hitachi) |
| | 4-bit | 8-bit | 4-bit | 8-bit (format stripe) | 16-bit | 9-bit | 12-bit | 18-bit | 24-bit | Serial 8 bit | 8/9-bit /3 wire | | | |
| LCD_FRAME | FRAME | | | | | | | | | | Drive 0 | Drive 0 | Drive 0 | Drive 0 |
| LCD_LINE | LINE | | | | | | | | | | Drive 0 | Drive 0 | Drive 0 | Drive 0 |
| LCD_SHIFT | SHIFT | | | | | | | | | | Drive 0 | Drive 0 | Drive 0 | Drive 0 |
| LCD_DEN | MOD | | | | | DEN | | | | | Drive 0 | Drive 0 | Drive 0 | Drive 0 |
| LCD_DATA0 | Drive 0 | D0 | Drive 0 | D0(G3) ¹ | R6 | R2 | R3 | R5 | R7 | D0 | D0 | | D0/SDA | |
| LCD_DATA1 | Drive 0 | D1 | Drive 0 | D1(R3) ¹ | G5 | R1 | R2 | R4 | R6 | D1 | D1 | | | |
| LCD_DATA2 | Drive 0 | D2 | Drive 0 | D2(B2) ¹ | B4 | R0 | R1 | R3 | R5 | D2 | D2 | | | |
| LCD_DATA3 | Drive 0 | D3 | Drive 0 | D3(G2) ¹ | R4 | G2 | G3 | G5 | G7 | D3 | D3 | | | |
| LCD_DATA4 | D0 | D4 | D0(R2) ¹ | D4(R2) ¹ | B5 | G1 | G2 | G4 | G6 | D4 | D4 | | | |
| LCD_DATA5 | D1 | D5 | D1(B1) ¹ | D5(B1) ¹ | R5 | G0 | G1 | G3 | G5 | D5 | D5 | | | |
| LCD_DATA6 | D2 | D6 | D2(G1) ¹ | D6(G1) ¹ | G4 | B2 | B3 | B5 | B7 | D6 | D6/SCK | | D6 | |
| LCD_DATA7 | D3 | D7 | D3(R1) ¹ | D7(R1) ¹ | B3 | B1 | B2 | B4 | B6 | D7 | D7/SDA | | D7 | |
| LCD_DATA8 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | G3 | B0 | B1 | B3 | B5 | Drive 0 | D8 | | | |
| LCD_DATA9 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | B2 | Drive 0 | R0 | R2 | R4 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 |
| LCD_DATA10 | GPI O | GPI O | GPI O | GPI O | R2 | GPI O | GPI O | R1 | R3 | GPI O | GPI O | GPI O | GPI O | GPI O |
| LCD_DATA11 | GPI O | GPI O | GPI O | GPI O | G1 | GPI O | GPI O | R0 | R2 | GPI O | GPI O | GPI O | GPI O | GPI O |
| LCD_DATA12 | GPI O | GPI O | GPI O | GPI O | R3 | GPI O | G0 | G2 | G4 | GPI O | GPI O | GPI O | GPI O | GPI O |
| LCD_DATA13 | GPI O | GPI O | GPI O | GPI O | G2 | GPI O | GPI O | G1 | G3 | GPI O | GPI O | GPI O | GPI O | GPI O |
| LCD_DATA14 | GPI O | GPI O | GPI O | GPI O | B1 | GPI O | GPI O | G0 | G2 | GPI O | GPI O | GPI O | GPI O | GPI O |
| LCD_DATA15 | GPI O | GPI O | GPI O | GPI O | R1 | GPI O | B0 | B2 | B4 | GPI O | GPI O | GPI O | GPI O | GPI O |
| LCD_DATA16 | GPI O | GPI O | GPI O | GPI O | GPI O | GPI O | GPI O | B1 | B3 | GPI O | GPI O | GPI O | GPI O | GPI O |
| LCD_DATA17 | GPI O | GPI O | GPI O | GPI O | GPI O | GPI O | GPI O | B0 | B2 | GPI O | GPI O | GPI O | GPI O | GPI O |
| LCD_DATA18 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | R1 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 |

| | | | | | | | | | | | | | | |
|------------|---------|---------|---------|---------|---------|---------|---------|---------|----|---------|----------|---------|-----------|---------|
| LCD_DATA19 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | R0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 |
| LCD_DATA20 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | G1 | Drive 0 | D/C# | | | |
| LCD_DATA21 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | G0 | Drive 0 | CS# | | | |
| LCD_DATA22 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | B1 | Drive 0 | WR#;R/W# | | WR#;E;SCK | |
| LCD_DATA23 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | Drive 0 | B0 | Drive 0 | RD#;E | | RD#;R/W# | |

Note

⁽¹⁾ These pin mappings use signal names commonly used for each panel type, however signal names may differ between panel manufacturers. The values shown in brackets represent the color components as mapped to the corresponding LCD_DATAxx signals at the first valid edge of LCD_SHIFT.

6.11 Data Bus Organization

There are two data bus architectures, little endian and big endian. Little endian means the bytes at lower addresses have lower significance. Big endian means the most significant byte has the lowest address.

Table 6-11 : Data Bus Organization

| | D[15:8] | D[7:0] |
|----------------------|----------------|---------------|
| Big endian | 2N | 2N + 1 |
| Little endian | 2N + 1 | 2N |

Note

⁽¹⁾ N : Byte Address

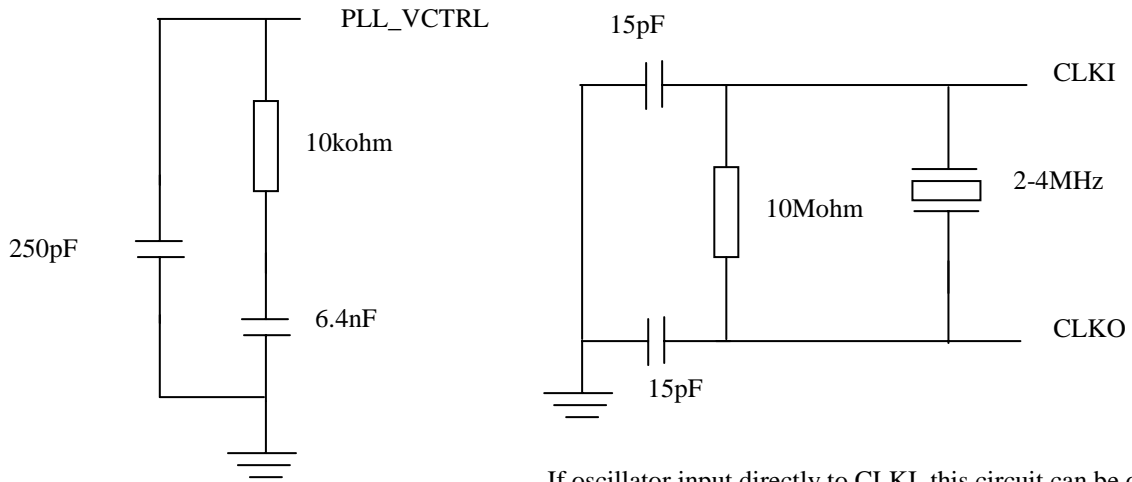
Table 6-12 : Pin State Summary

| MCU Mode (Endian) | A0 | RD/WR# | RD# | WE1# | WE0# | Operation |
|--------------------------|-----------|---------------|------------|-------------|-------------|----------------------|
| Generic#1 (Big) | X | 0 | 0 | 1 | 1 | Word read |
| | X | 0 | 1 | 1 | 1 | High byte read 2N |
| | X | 1 | 0 | 1 | 1 | Low byte read 2N+1 |
| | X | 1 | 1 | 0 | 0 | Word write |
| | X | 1 | 1 | 0 | 1 | High byte write 2N |
| | X | 1 | 1 | 1 | 0 | Low byte write 2N+1 |
| Generic#1 (Little) | X | 0 | 0 | 1 | 1 | Word read |
| | X | 0 | 1 | 1 | 1 | High byte read 2N+1 |
| | X | 1 | 0 | 1 | 1 | Low byte read 2N |
| | X | 1 | 1 | 0 | 0 | Word write |
| | X | 1 | 1 | 0 | 1 | High byte write 2N+1 |
| | X | 1 | 1 | 1 | 0 | Low byte write 2N |
| Generic#2 (Big) | 0 | X | 0 | 0 | 1 | Word read |
| | 0 | X | 0 | 1 | 1 | High byte read 2N |
| | 1 | X | 0 | 0 | 1 | Low byte read 2N+1 |
| | 0 | X | 1 | 0 | 0 | Word write |
| | 0 | X | 1 | 1 | 0 | High byte write 2N |
| | 1 | X | 1 | 0 | 0 | Low byte write 2N+1 |
| Generic#2 (Little) | 0 | X | 0 | 0 | 1 | Word read |
| | 1 | X | 0 | 0 | 1 | High byte read 2N+1 |
| | 0 | X | 0 | 1 | 1 | Low byte read 2N |
| | 0 | X | 1 | 0 | 0 | Word write |
| | 1 | X | 1 | 0 | 0 | High byte write 2N+1 |
| | 0 | X | 1 | 1 | 0 | Low byte write 2N |

7 FUNCTIONAL BLOCK DESCRIPTIONS

7.1 Phase Lock Loop (PLL)

The built-in PLL synthesizes the internal clock by an external 2MHz – 4MHz clock through the CLKI and CLKO. RC circuit should be connected if internal PLL is selected. The input of clock source can be either oscillator or crystal. If oscillator is used, the clock source is input directly to CLKI and leave CLKO floating. The target maximum output frequency of the PLL is 85MHz. If sync MCU interface, PLL should be disabled and use direct clock input to CLKI2.



RC circuit for PLL_VCTRL

If oscillator input directly to CLKI, this circuit can be omitted

Figure 7-1 : Circuit for PLL enable

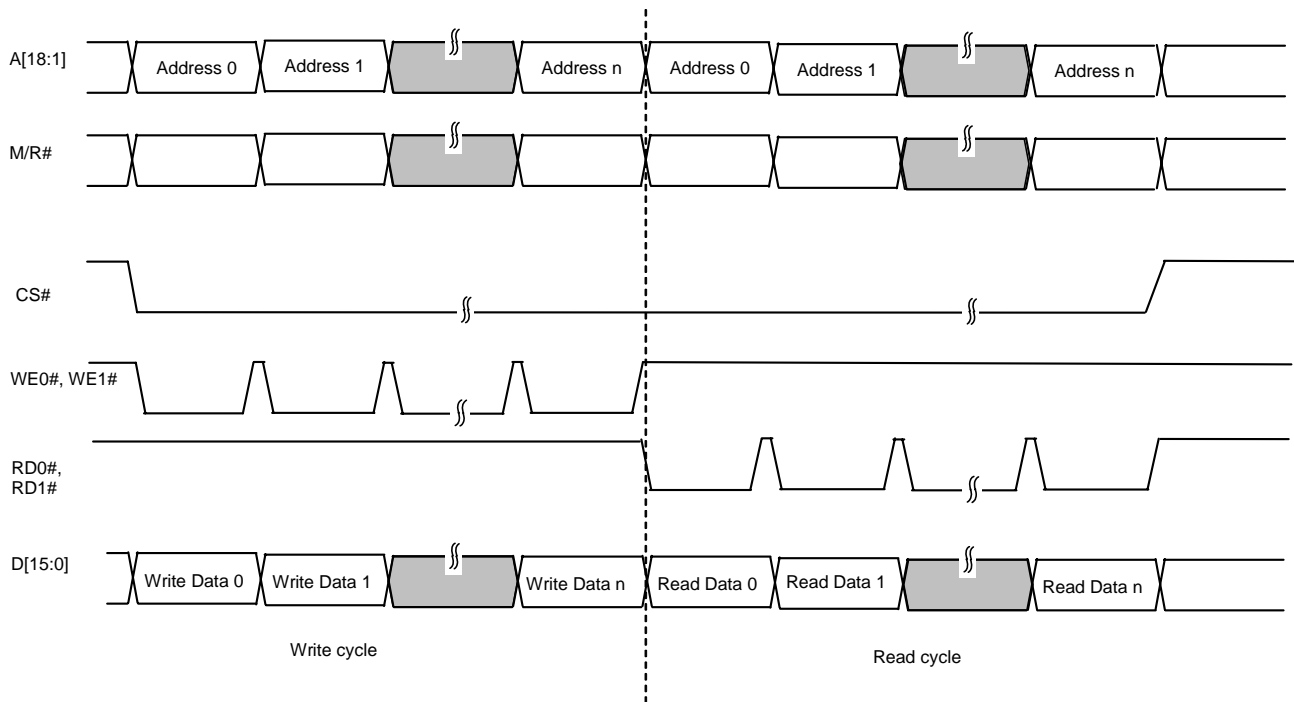
7.2 Embedded Memory

The 256kByte embedded memory can be access by the modules for different functions. For example, frame buffer, SD read/write buffer, internal buffer for JPEG encoding/decoding, encoded JPEG image output and so on.

7.3 MCU Interface

Responds to bus request for various kinds of MCU and translates to internal interface signals. SSD1926 can support direct and indirect addressing mode.

7.3.1 Generic #1 addressing Mode

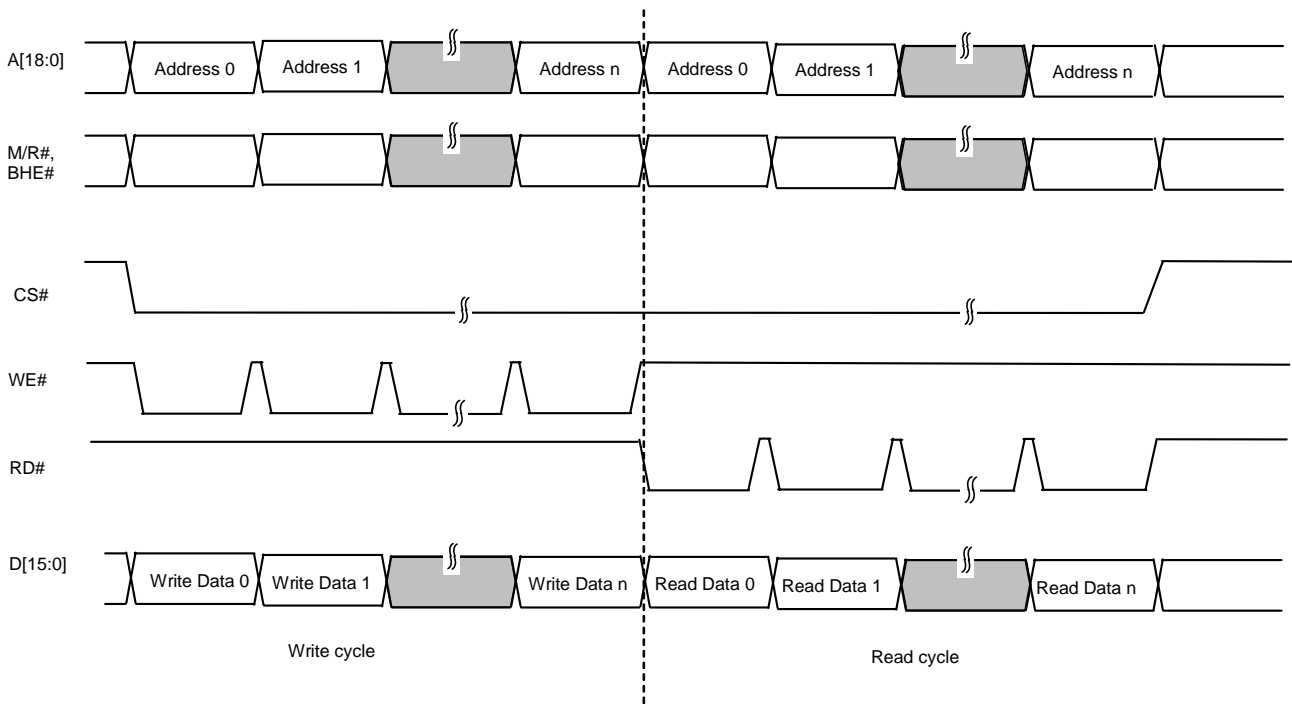


Note :

* 13 MCLK is needed for each cycle if WAIT# is not used for interface.

Figure 7-2 : Generic #1 Interface Timing

7.3.2 Generic #2 addressing Mode



Note :

* 13 MCLK is needed for each cycle if WAIT# is not used for interface.

Figure 7-3 : Generic #2 Interface Timing

7.3.3 8080 Indirect addressing Mode

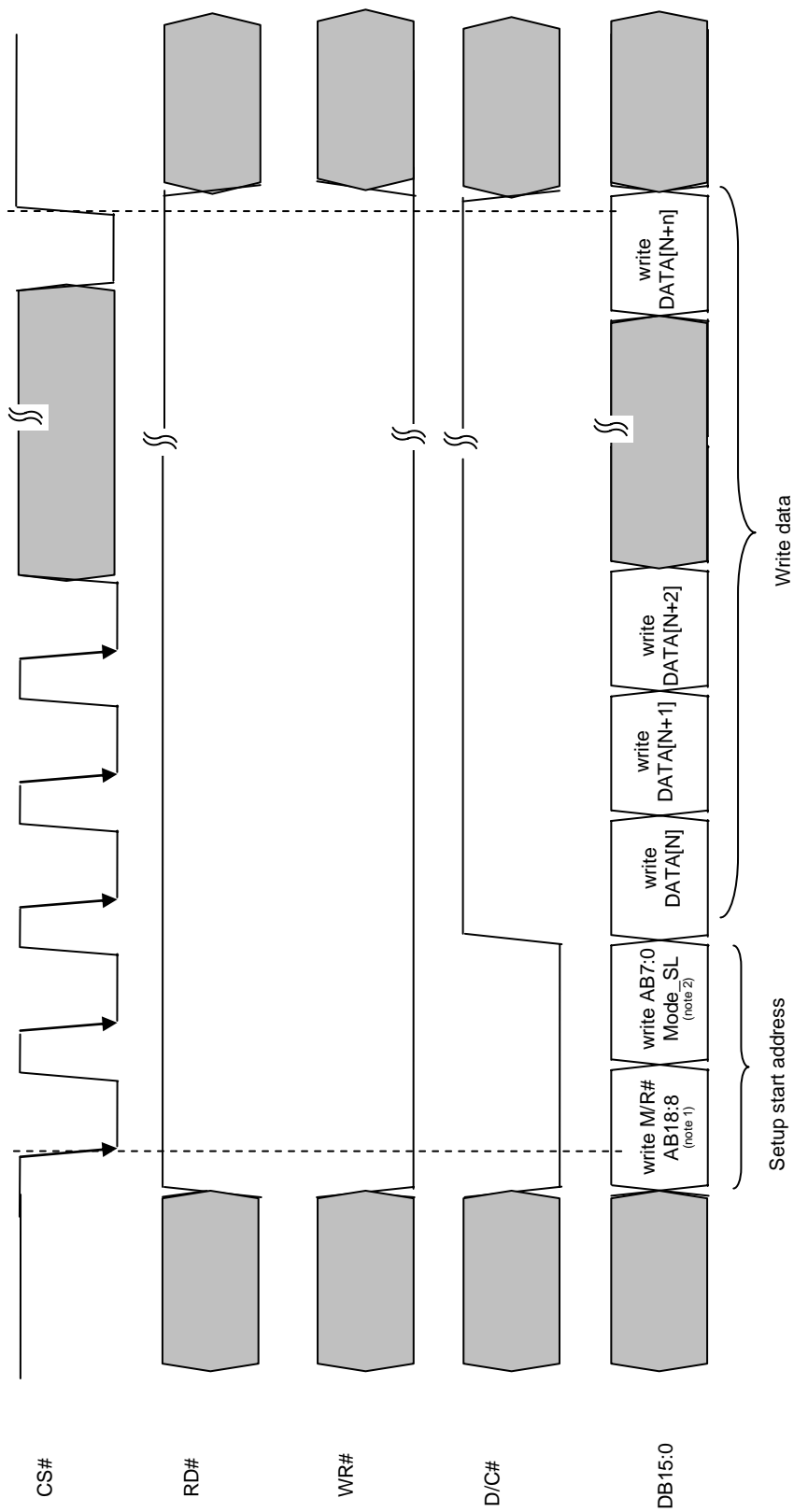
8080 Indirect addressing mode consists of 16 or 8 bi-directional data pins (DB15:0), CS#, RD#, WR# and D/C#. CS# is the chip select; RD# is the read strobe; WR# is the write strobe; and D/C# is the data/command select. They can be used for either 8 bit (DB7:0) or 16 bit (DB15:0) bus protocol. CS# falling edge input serves as data read latch signal when RD# is low. D/C# controls whether reading the data or reading the command (i.e. status). CS# falling edge input serves as data write latch signal when WR# is low. D/C# controls whether writing the data or writing the command (i.e. address). In read operation, dummy invalid data read is required after start address.

The start address counter should be assigned before the data is written or read. The most significant bit of the start address is used to select the memory or register (M/R#). During the byte mode access, the address counter is automatically incremented by 1 byte after writing or reading the data. During the word mode access, the address counter is automatically incremented by 1 word after writing or reading the data. The address counter of memory will be returned to 0x00000 if counter = 0x3FFFF in byte mode or 0x3FFFE in word mode.

For 16 bit bus protocol, it can interface with byte or word mode access. The last byte of start address in 16-bit access will be used to select byte or word mode (MODE_SL). MODE_SL = 0x00 means byte access and MODE_SL = 0x01 means word access.

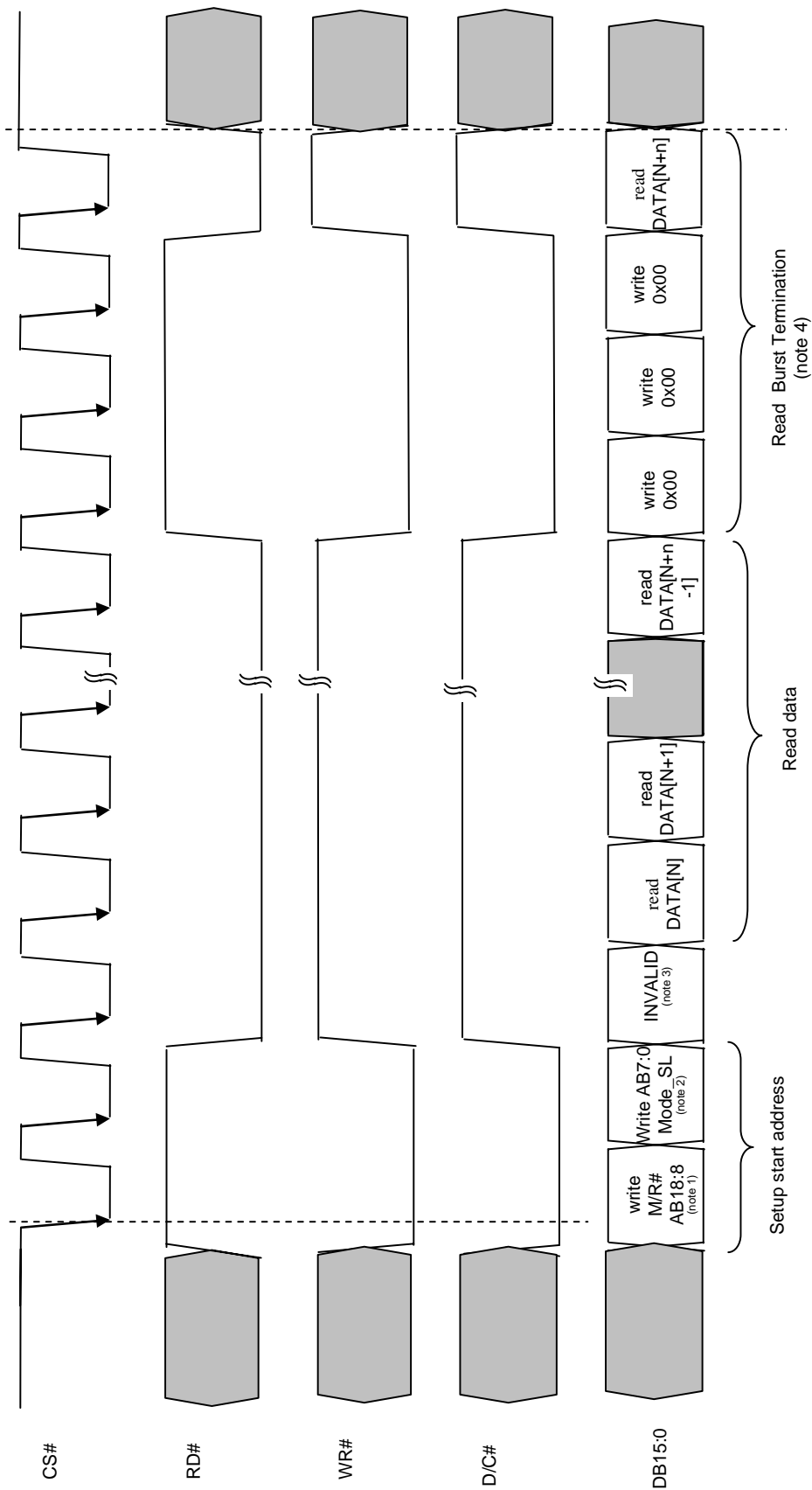
Read Burst Termination must be asserted for all JPEG related memory access. If the burst length is as small as 1, the read data stage may be reduced to a single dummy read.

Refer to section 12 for Pseudo-code examples.



- Note :**
- 1 : Bit15 represent the M/R#, Bit15 = 1 means memory access, Bit15 = 0 means register access. Bit14;11 = 0.
 - 2 : Bit10:0 represent the the address AB18:8. Bit15:8 represent the address AB7:0 and Bit7:0 represent Mode_SL. Mode_SL to select byte or word access during 16 bit mode. 0x00 means Byte access, 0x01 means word access.
- * 7 MCLK is needed for each cycle if WAIT# is not used for interface.

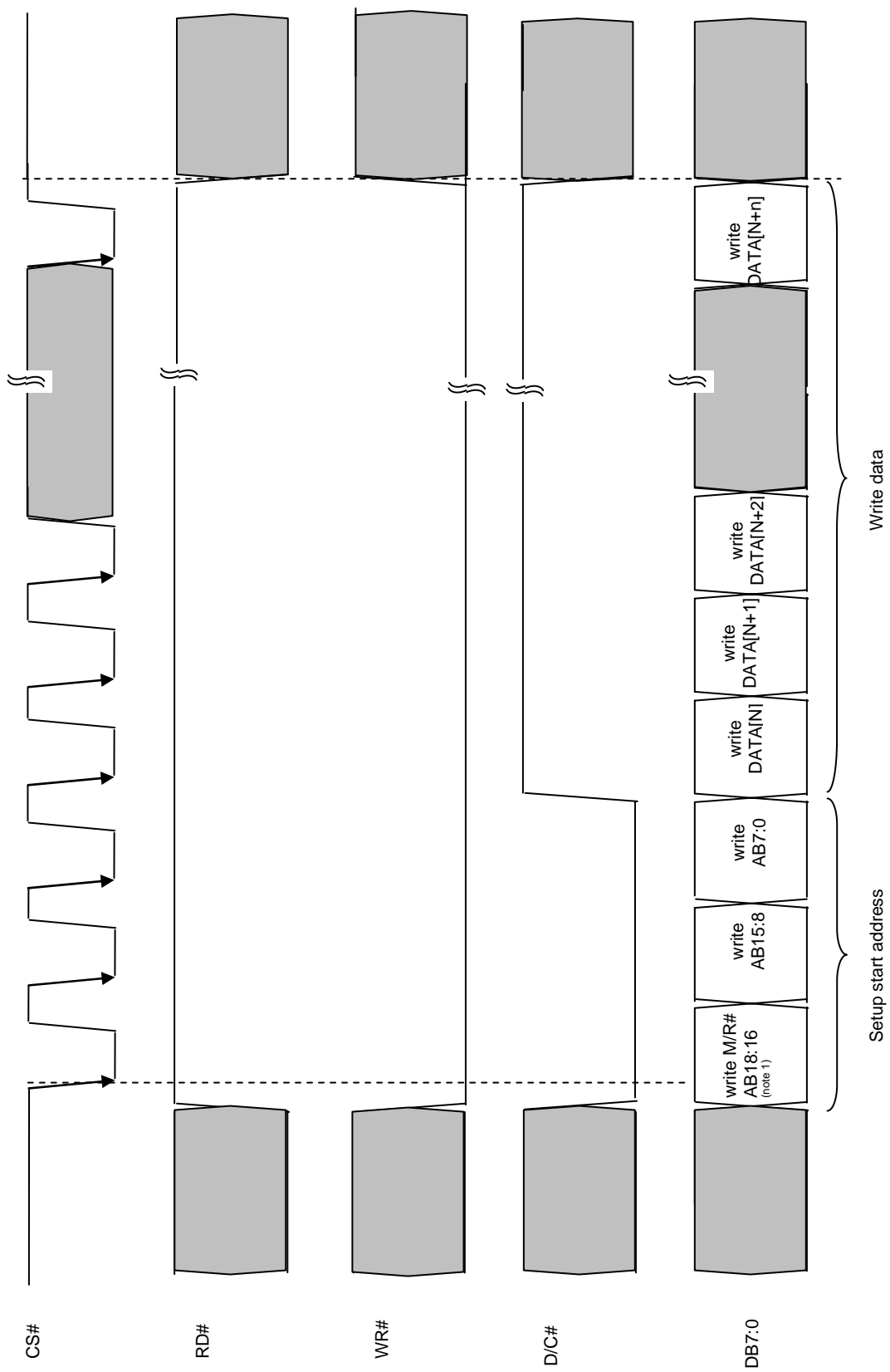
Figure 7-4 : 8080 16 bit Interface Timing (write cycle)



- Note :**
- 1 : Bit15 represent the M/R#, Bit15 = 1 means memory access, Bit15 = 0 means register access. Bit14:11 = 0.
 - 2 : Bit10:0 represent the address AB18:8.
 - 3 : Bit15:8 represent the address AB7:0 and Bit7:0 represent Mode_SL. Mode_SL to select byte or word access during 16 bit mode. 0x00 means Byte access, 0x01 means word access. Invalid dummy data cycle is needed after address is written.
 - 4 : Read Burst Termination must be asserted for all JPEG related memory access.

* 7 MCLK is needed for each cycle if WAIT# is not used for interface.

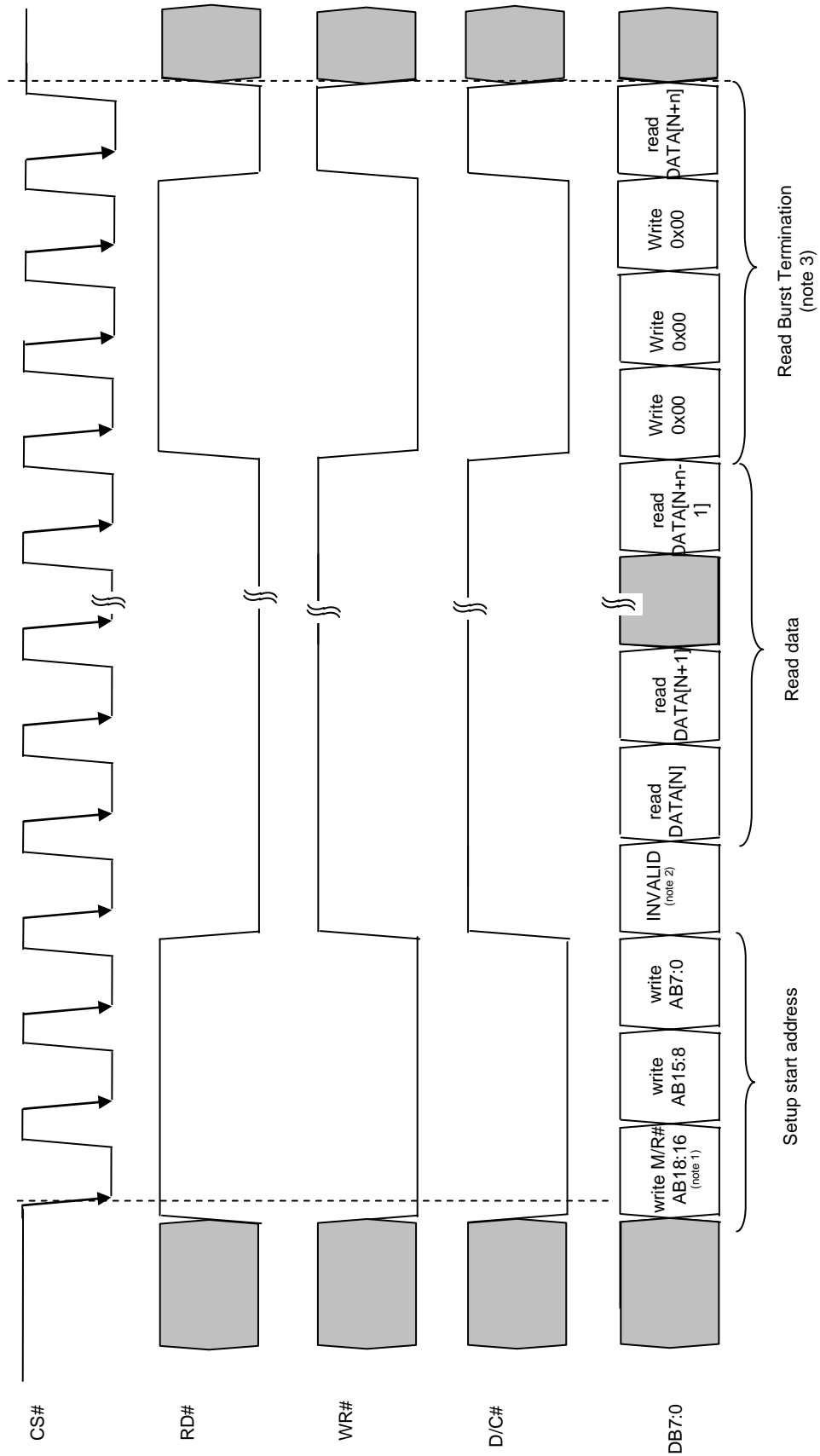
Figure 7-5 : 8080 16 bit Interface Timing (read cycle)



Note :

- 1 : Bit7 represent the M/R#, Bit7 = 1 means memory access, Bit7 = 0 means register access.
 Bit6:3 = 0.
 Bit2:0 represent the the address AB18:16.
- * 7 MCLK is needed for each cycle if WAIT# is not used for interface.

Figure 7-6 : 8080 8 bit Interface Timing (write cycle)



- Note :**
- 1 : Bit7 represent the M/R#, Bit7 = 1 means memory access, Bit7 = 0 means register access. Bit6:3 = 0.
 - 2 : Bit2:0 represent the the address AB18:16.
 - 3 : Invalid dummy read cycle is needed after address is written.
- * 7 MCLK is needed for each cycle if WAIT# is not used for interface.

Figure 7-7 : 8080 8 bit Interface Timing (read cycle)

7.4 Registers

It stores all the register settings for different functional modules. Refer to Application Note for Register Table.

7.5 JPEG Decoder

With MMC/SD card interface, the JPEG data can be stored in external MMC/SD card. The JPEG decoder can decompress the JPEG image from embedded memory to display⁽¹⁾. If the image stored in MMC/SD card is copied to embedded memory, the JPEG decoder can decompress it to display also.

Note

⁽¹⁾ If the output memory address is the same as the overlay window, the decompressed image will be display immediately.

7.6 2D Engine

The 2D engine is designed on the basis of Microsoft Windows GDI. It support straight line drawing, rectangle drawing, rectangle color fill, rectangle pattern fill, BitBLT, color expansion, StretchBLT and alpha blending.

7.7 Display Interface

This is LCD interface for the main display. The maximum resolution of the LCD depends on the size of frame buffer located in the embedded memory. This display interface supports most panel type, including dump STN, CSTN, TFT. The smart STN, CSTN, TFT, OLED panel of parallel and serial interface are also supported.

7.8 MMC/SD/SDIO Interface

This interface act as a bridge between the MCU and the external memory card. This interface can also used as a bridge between the internal functional blocks such as JPEG encoder/decoder and external memory card. Since this interface also supports SDIO, the MCU can use this interface as an expansion slot.

7.9 General Purpose Input/Output (GPIO)

This is a collection of 13 GPIOs with can be used for LCD, keypad, LED backlight control and so on.

8 MAXIMUM RATINGS

Table 8-1: Absolute Maximum Ratings

| Symbol | Parameter | Rating | Units |
|------------|-------------------------|------------------------------------|-------|
| IOV_{DD} | Supply Voltage | $V_{SS} - 0.3$ to 4.0 | V |
| V_{IN} | Input Voltage | $V_{SS} - 0.3$ to 5.0 | V |
| V_{OUT} | Output Voltage | $V_{SS} - 0.3$ to $IOV_{DD} + 0.5$ | V |
| T_{STG} | Storage Temperature | -65 to 150 | °C |
| T_{SOL} | Solder Temperature/Time | 260 for 10 sec. max at lead | °C |

Maximum Ratings are those values beyond which damage 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 to 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 $V_{SS} \leq (V_{IN}$ or $V_{OUT}) \leq IOV_{DD}$. Reliability of operation is enhanced if unused input are connected to an appropriate logic voltage level (e.g., either V_{SS} or IOV_{DD}). This device is not radiation protected.

Table 8-2 : Recommended Operating Conditions

| Symbol | Parameter | Condition | Min | Typ | Max | Units |
|--------------|-----------------------|---------------|----------|-----|------------|-------|
| IOV_{DD} | Supply Voltage | $V_{SS} = 0V$ | 3.0 | 3.3 | 3.6 | V |
| $COREV_{DD}$ | Supply Voltage | $V_{SS} = 0V$ | 1.62 | 1.8 | 1.98 | V |
| PV_{DD} | Supply Voltage | $V_{SS} = 0V$ | 1.62 | 1.8 | 1.98 | V |
| V_{IN} | Input Voltage | | V_{SS} | | IOV_{DD} | V |
| T_{OPR} | Operating Temperature | | -30 | 25 | 85 | °C |

9 DC CHARACTERISTICS

Table 9-1 : Electrical Characteristics for IOV_{DD} = 3.3V typical

| Symbol | Parameter | Condition | Min | Typ | Max | Units |
|------------------|---------------------------|---|----------------------------|-----|----------------------------|-------|
| I _{DDs} | Quiescent Current | Quiescent Conditions PLL_DIS = VSS | | | 150 | μA |
| I _{IZ} | Input Leakage Current | | -1 | | 1 | μA |
| I _{OZ} | Output Leakage Current | | -1 | | 1 | μA |
| V _{OH} | High Level Output Voltage | IOV _{DD} = min I _{OH} = -2mA (Type 1) -4mA (Type 2) -16mA (Type 3) | 70% * IOV _{DD} | | | V |
| V _{OL} | Low Level Output Voltage | IOV _{DD} = min I _{OL} = 2mA (Type 1) 4mA (Type 2) 16mA (Type 3) | | | 30% * IOV _{DD} | V |
| V _{IH} | High Level Input Voltage | LVTTL Level, IOV _{DD} = max | 90% * IOV _{DD} | | | V |
| V _{IL} | Low Level Input Voltage | LVTTL Level, IOV _{DD} = min | | | 10% * IOV _{DD} | V |
| V _{T+} | High Level Input Voltage | LVTTL Schmitt | 1.1 | | | V |

10 AC CHARACTERISTICS

Conditions: $IOV_{DD} = 3.3V \pm 10\%$
 $T_A = -30^{\circ}C$ to $85^{\circ}C$
 T_{rise} and T_{fall} for all inputs must be < 5 ns (10% ~ 90%)
 $C_L = 50pF$ (Bus/CPU Interface)
 $C_L = 0pF$ (LCD Panel Interface)

10.1 Clock Timing

10.1.1 Input Clocks

Table 10-1 : Clock Input Requirements for CLKI

| Symbol | Parameter | Min | Max | Units |
|------------|------------------------------|--------------|-----|-------|
| F_{CLKI} | Input Clock Frequency (CLKI) | 2 | 4 | MHz |
| T_{CLKI} | Input Clock period (CLKI) | $1/f_{CLKI}$ | | ns |

Table 10-2 : Oscillator Clock Input Requirements for CLKI2

| Symbol | Parameter | Min | Max | Units |
|-------------|-------------------------------|---------------|-----|-------|
| F_{CLKI2} | Input Clock Frequency (CLKI2) | | 85 | MHz |
| T_{CLKI2} | Input Clock period (CLKI2) | $1/f_{CLKI2}$ | | ns |

10.2 CPU Interface Timing

The following section are CPU interface AC Timing based on $IOV_{DD} = 3.3V$.

10.2.1 Generic #1 Interface Timing

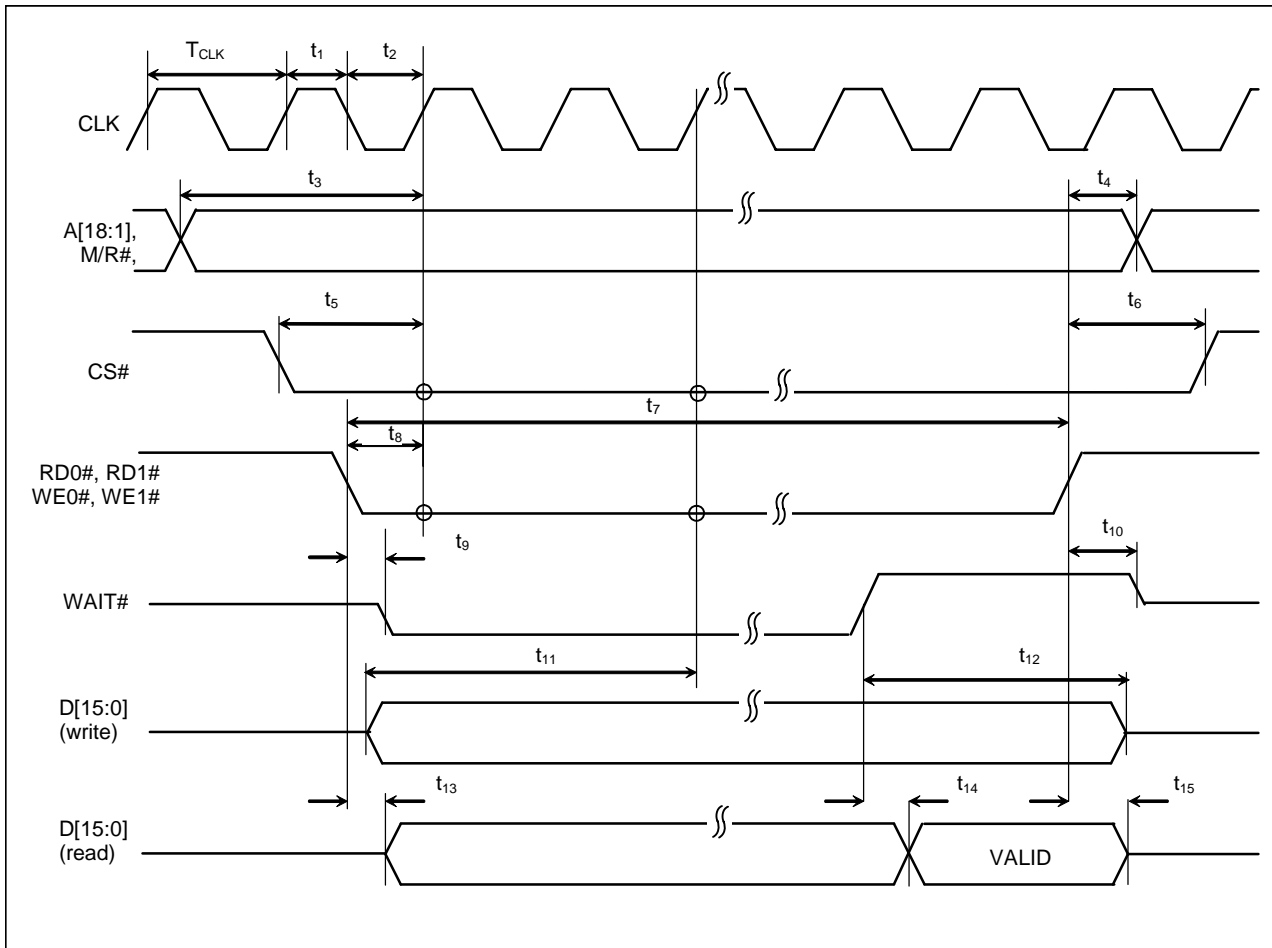


Figure 10-1 : Generic #1 Interface Timing

Table 10-3 : Generic #1 Interface Timing

| Symbol | Parameter | Min | Max | Units |
|-----------|--|-------------|-----|-------|
| f_{CLK} | Bus Clock frequency | | 85 | MHz |
| T_{CLK} | Bus Clock period | $1/f_{CLK}$ | | ns |
| t_1 | Clock pulse width high | 6 | | ns |
| t_2 | Clock pulse width low | 6 | | ns |
| t_3 | A[18:1], M/R# setup to first CLK rising edge where CS# = 0 and either RD0#, RD1# = 0 or WE0#, WE1# = 0 | 1 | | ns |
| t_4 | A[18:1], M/R# hold from either RD0#, RD1# or WE0#, WE1# rising edge | 0 | | ns |
| t_5 | CS# setup to CLK rising edge | 1 | | ns |
| t_6 | CS# hold from either RD0#, RD1# or WE0#, WE1# rising edge | 1 | | ns |
| t_{7a} | RD0#, RD1#, WE0#, WE1# asserted for MCLK = PLL_CLK | | 13 | TCLK |
| t_{7b} | RD0#, RD1#, WE0#, WE1# asserted for MCLK = PLL_CLK ÷2 | | 18 | TCLK |
| t_{7c} | RD0#, RD1#, WE0#, WE1# asserted for MCLK = PLL_CLK ÷3 | | 23 | TCLK |
| t_{7d} | RD0#, RD1#, WE0#, WE1# asserted for MCLK = PLL_CLK ÷4 | | 28 | TCLK |
| t_8 | RD0#, RD1#, WE0#, WE1# setup to CLK rising edge | 1 | | ns |
| t_9 | Falling edge of either RD0#, RD1# or WE0#, WE1# to WAIT# driven low | 3 | 15 | ns |
| t_{10} | Rising edge of either RD0#, RD1# or WE0#, WE1# to WAIT# high impedance | 3 | 13 | ns |
| t_{11} | D[15:0] setup to third CLK rising edge where CS# = 0 and WE0#, WE1#=0 (write cycle)(see note 1) | 0 | | ns |
| t_{12} | D[15:0] hold from WAIT# rising edge (write cycle) | 0 | | ns |
| t_{13} | RD0#, RD1# falling edge to D[15:0] driven (read cycle) | 3 | 14 | ns |
| t_{14} | WAIT# rising edge to D[15:0] valid (read cycle) | | 2 | ns |
| t_{15} | RD0#, RD1# rising edge to D[15:0] high impedance (read cycle) | 3 | 11 | ns |

1. t_{11} is the delay from when data is placed on the bus until the data is latched into the write buffer.

10.2.2 Generic #2 Interface Timing (e.g. ISA)

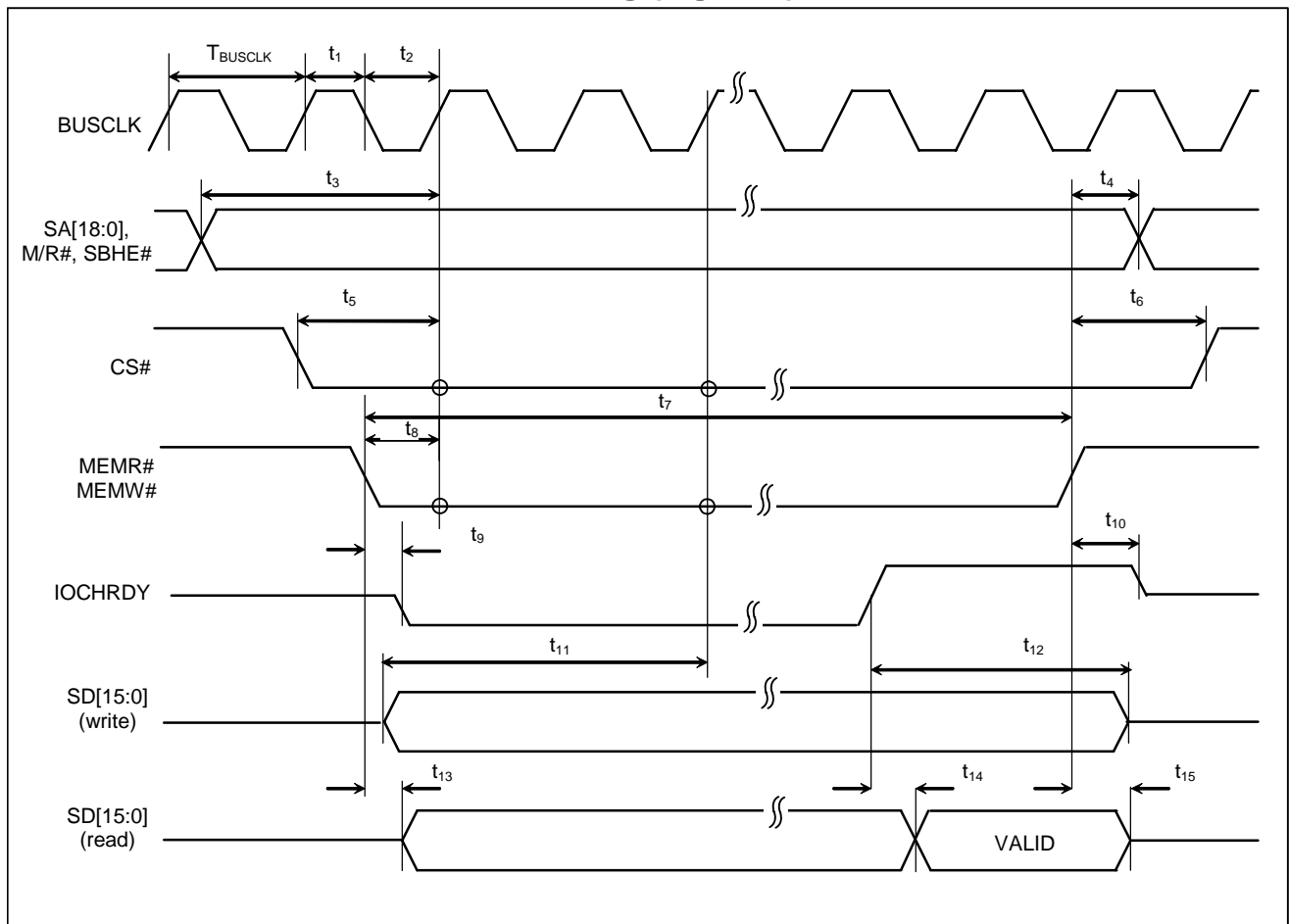


Figure 10-2 : Generic #2 Interface Timing

Table 10-4 : Generic #2 Interface Timing

| Symbol | Parameter | Min | Max | Units |
|---------------------|---|-----------------------|-----|---------------------|
| f_{BUSCLK} | Bus Clock frequency | | 85 | MHz |
| T_{BUSCLK} | Bus Clock period | $1/f_{\text{BUSCLK}}$ | | ns |
| t_1 | Clock pulse width high | 6 | | ns |
| t_2 | Clock pulse width low | 6 | | ns |
| t_3 | SA[18:0], M/R#, SBHE# setup to first BUSCLK rising edge where CS# = 0 and either MEMR# = 0 or MEMW# = 0 | 1 | | ns |
| t_4 | SA[18:0], M/R#, SBHE# hold from either MEMR# or MEMW# rising edge | 0 | | ns |
| t_5 | CS# setup to BUSCLK rising edge | 1 | | ns |
| t_6 | CS# hold from either MEMR# or MEMW# rising edge | 0 | | ns |
| t_{7a} | MEMR# or MEMW# asserted for MCLK = PLL_CLK | | 13 | T_{BUSCLK} |
| t_{7b} | MEMR# or MEMW# asserted for MCLK = PLL_CLK ÷2 | | 18 | T_{BUSCLK} |
| t_{7c} | MEMR# or MEMW# asserted for MCLK = PLL_CLK ÷3 | | 23 | T_{BUSCLK} |
| t_{7d} | MEMR# or MEMW# asserted for MCLK = PLL_CLK ÷4 | | 28 | T_{BUSCLK} |
| t_8 | MEMR# or MEMW# setup to BUSCLK rising edge | 1 | | ns |
| t_9 | Falling edge of either MEMR# or MEMW# to IOCHRDY driven low | 3 | 15 | ns |
| t_{10} | Rising edge of either MEMR# or MEMW# to IOCHRDY high impedance | 3 | 13 | ns |
| t_{11} | SD[15:0] setup to third BUSCLK rising edge where CS# = 0 and MEMW#=0 (write cycle)(see note 1) | 0 | | ns |
| t_{12} | SD[15:0] hold from IOCHRDY rising edge (write cycle) | 0 | | ns |
| t_{13} | MEMR# falling edge to SD[15:0] driven (read cycle) | 3 | 13 | ns |
| t_{14} | IOCHRDY rising edge to SD[15:0] valid (read cycle) | | 2 | ns |
| t_{15} | Rising edge of MEMR# to SD[15:0] high impedance (read cycle) | 3 | 12 | ns |

1. t_{11} is the delay from when data is placed on the bus until the data is latched into the write buffer.

10.2.3 8080 Indirect Interface Timing

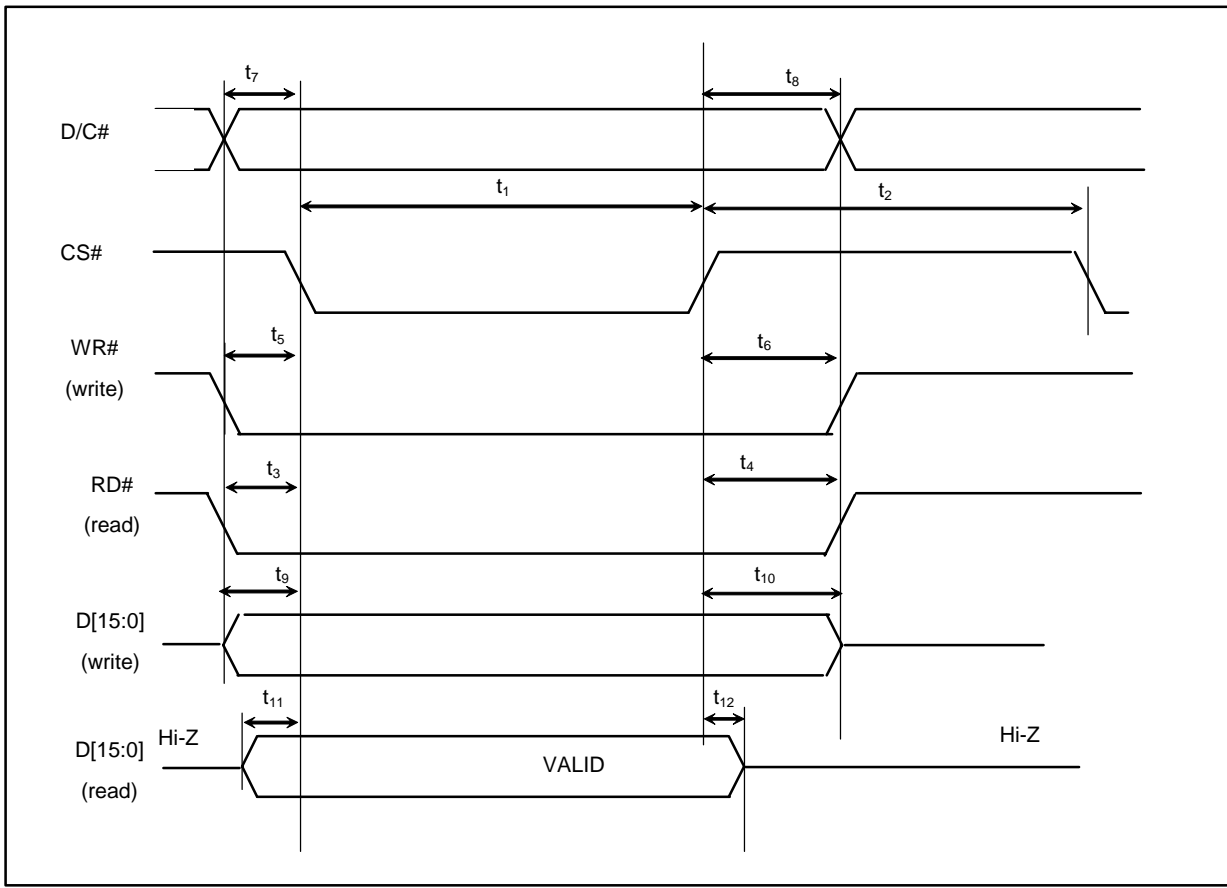


Figure 10-3 : 8080 Interface Timing

Table 10-5 : 8080 Interface Timing

| Symbol | Parameter | Min | Max | Units |
|----------|-------------------------|-----|-----|-------|
| t_1 | CS# pulse width low | 82 | | ns |
| t_2 | CS# pulse width high | 82 | | ns |
| t_3 | RD# setup | 18 | | ns |
| t_4 | RD# hold | 0 | | ns |
| t_5 | WR# setup | 18 | | ns |
| t_6 | WR# hold | 0 | | ns |
| t_7 | D/C# setup | 18 | | ns |
| t_8 | D/C# hold | 0 | | ns |
| t_9 | D[15:0] setup for write | 18 | | ns |
| t_{10} | D[15:0] hold for write | 0 | | ns |
| t_{11} | D[15:0] delay for read | 55 | | ns |
| t_{12} | D[15:0] hold for read | 0 | | ns |

Note : Above timing is based on MCLK = 85MHz

11 APPLICATION EXAMPLES

11.1 Application Diagram

Figure 11-1 : Typical System Diagram (Generic #1 Bus)

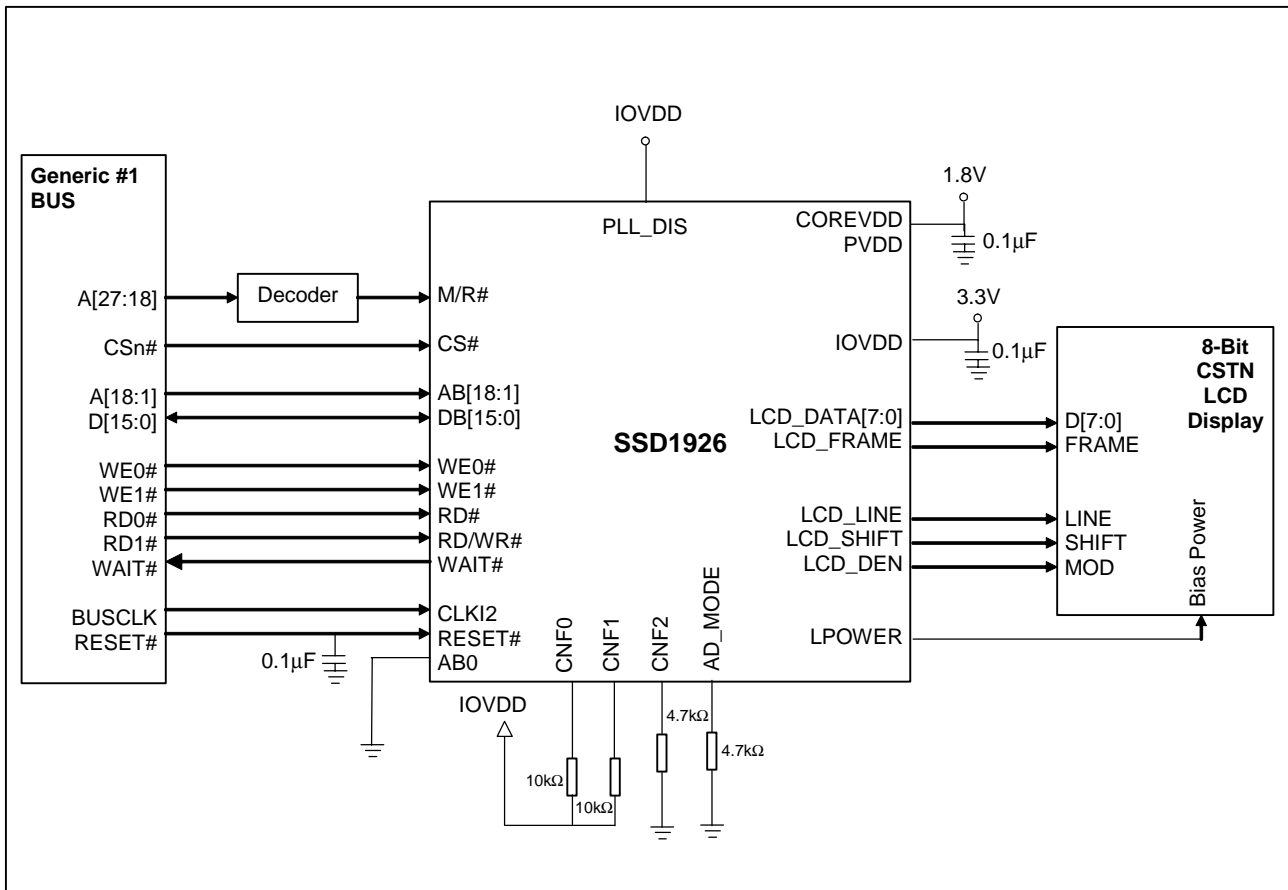


Figure 11-2 : Typical System Diagram (Generic #2 Bus)

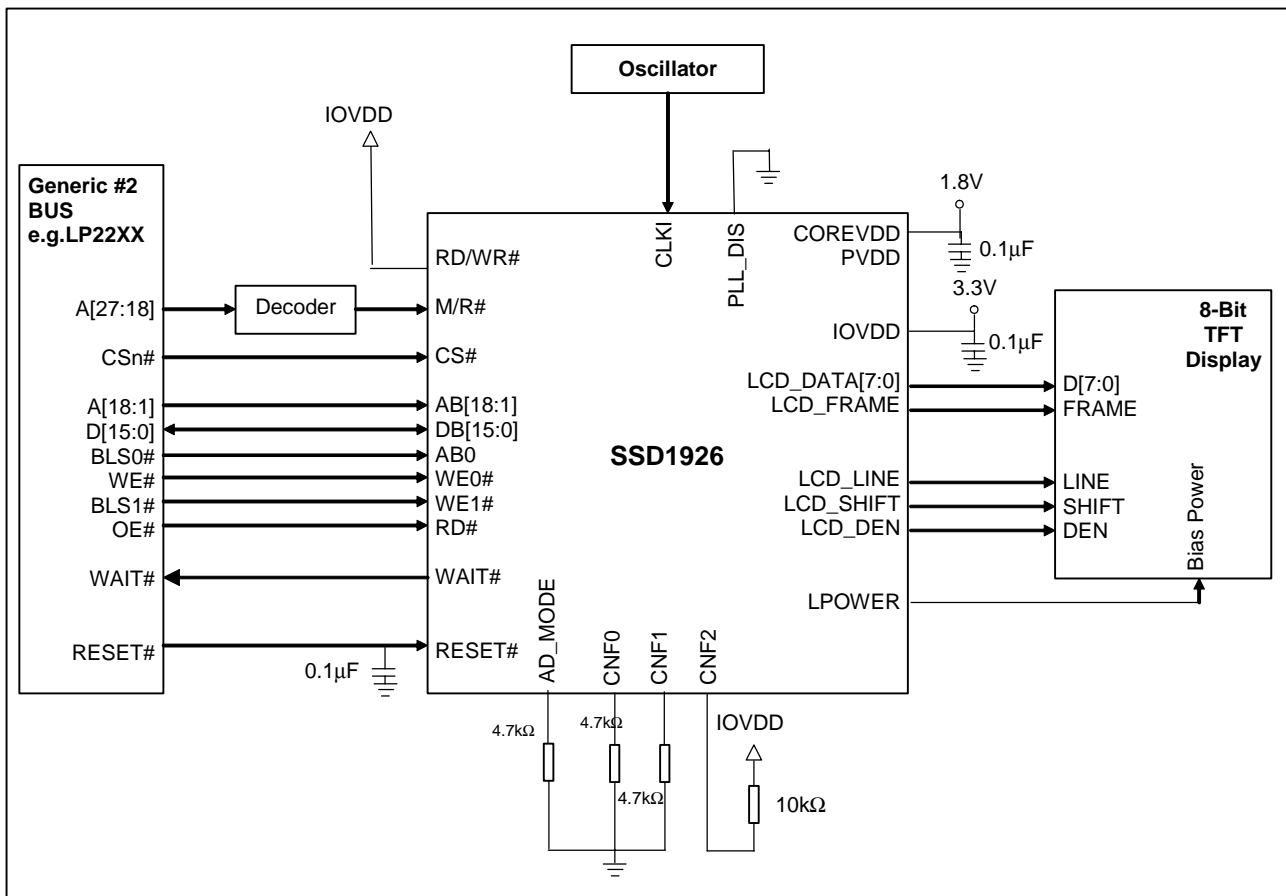


Figure 11-3 : Typical System Diagram (Indirect 8080 16 bit Bus)

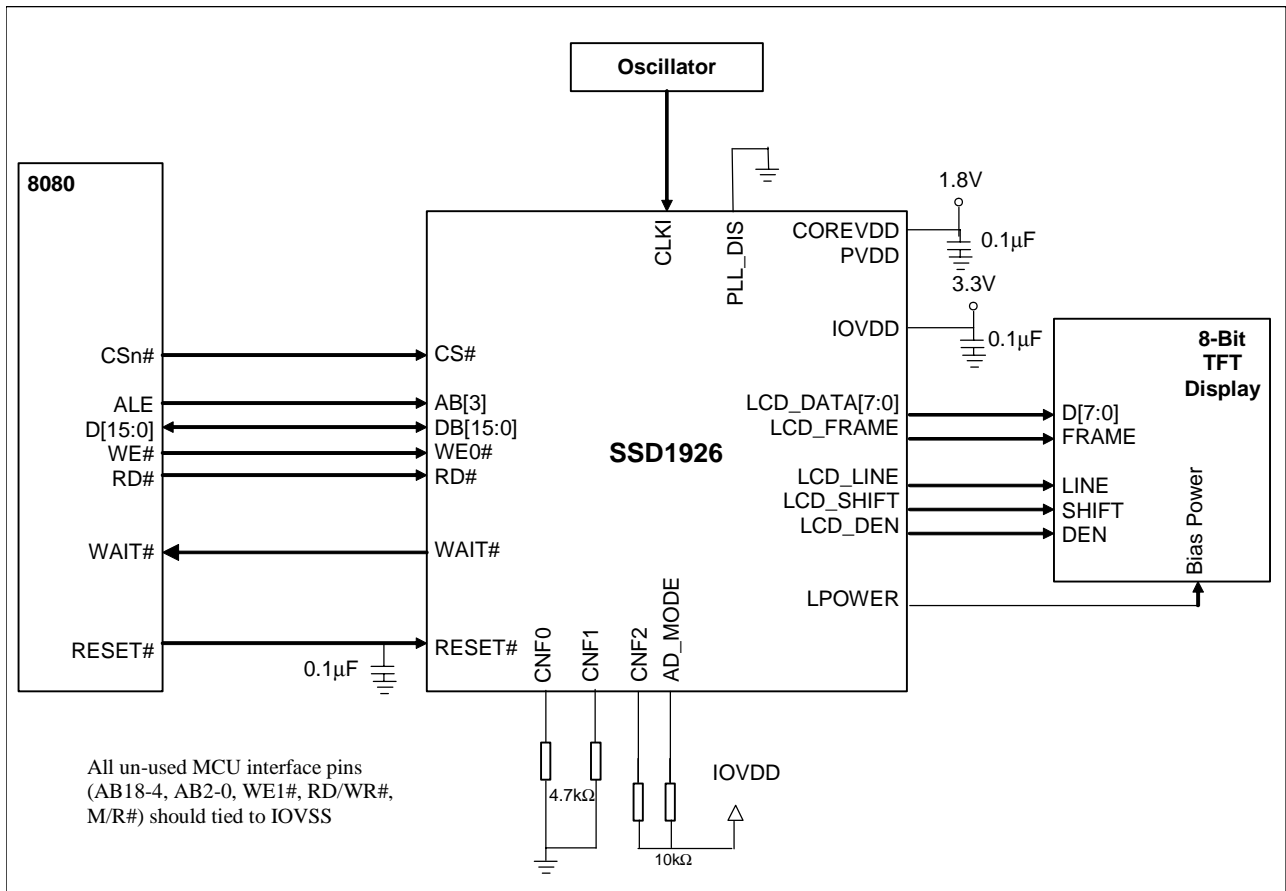
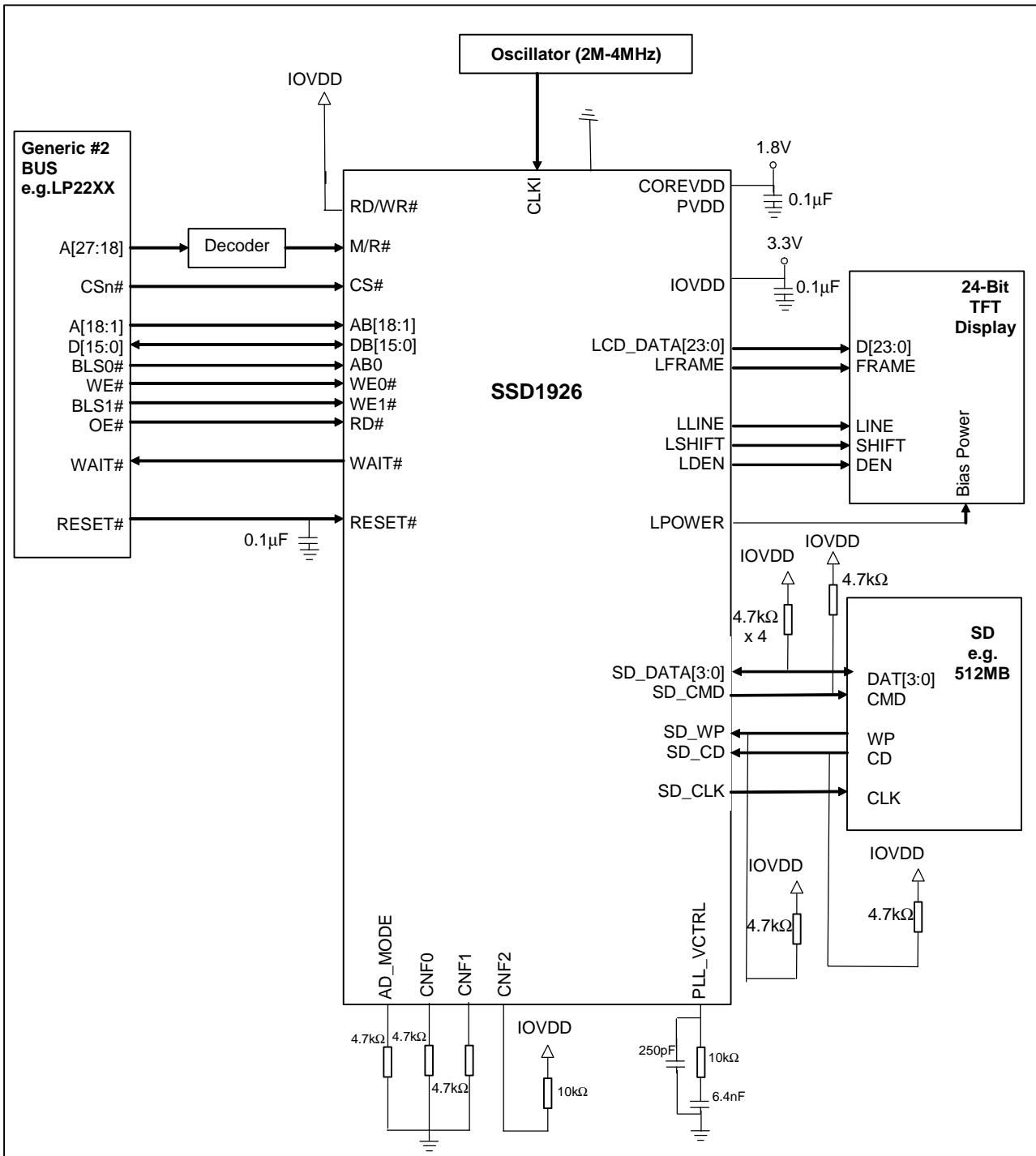


Figure 11-4: Typical System Diagram (Generic #2 Bus)



12 Pseudo-code Examples for Indirect address mode

12.1 8080 Indirect address mode

For example, PORTA is used for control signals and PORTB is used for data signals.

PORTA[3:0] are the control signals

| PORTA[7] | PORTA[6] | PORTA[5] | PORTA[4] | PORTA[3] | PORTA[2] | PORTA[1] | PORTA[0] |
|----------|----------|----------|----------|----------|----------|----------|----------|
| X | X | X | X | CS# | RD# | WR# | D/C# |

X : Don't care

PORTB[15:0] are the data signals for 16 bit mode

PORTB[7:0] are the data signals for 8 bit mode

| | | |
|---------------------|--|-------------------------------|
| read_command (cmd) | | |
| write PORTA, 0x0A | | [CS#=1, RD#=0, WR#=1, D/C#=0] |
| write PORTA, 0x02 | | [CS#=0, RD#=0, WR#=1, D/C#=0] |
| read PORTB, cmd | | |
| write PORTA, 0x0E | | [CS#=1, RD#=1, WR#=1, D/C#=0] |
| read_data (data) | | |
| write PORTA, 0x0B | | [CS#=1, RD#=0, WR#=1, D/C#=1] |
| write PORTA, 0x03 | | [CS#=0, RD#=0, WR#=1, D/C#=1] |
| read PORTB, data | | |
| write PORTA, 0x0F | | [CS#=1, RD#=1, WR#=1, D/C#=1] |
| write_command (cmd) | | |
| write PORTB, cmd | | |
| write PORTA, 0x0C | | [CS#=1, RD#=1, WR#=0, D/C#=0] |
| write PORTA, 0x04 | | [CS#=0, RD#=1, WR#=0, D/C#=0] |
| write PORTA, 0x0E | | [CS#=1, RD#=1, WR#=1, D/C#=0] |
| write_data (data) | | |
| write PORTB, data | | |
| write PORTA, 0x0D | | [CS#=1, RD#=1, WR#=0, D/C#=1] |
| write PORTA, 0x05 | | [CS#=0, RD#=1, WR#=0, D/C#=1] |
| write PORTA, 0x0F | | [CS#=1, RD#=1, WR#=1, D/C#=1] |

Example 1 : Register Access with 16 bit indirect address mode (Big Endian, CNF4=1) with Word Mode

(Write **Register** REG[0x010h]=0x11, REG[0x011h]=0x22, REG[0x012h]=0x33, REG[0x013h]=0x04 and read back the contents)

```

Step 1: Set the start address to 0x00010 and write DATA0
  write_command 0x0000      [M/R#=0 => Register]
  write_command 0x1001      [start address = 0x00010, MODE_SL=0x01 => Word]
  write_data 0x1122         [write word 0x1122 to REG[0x010]]
Step 2 : Write DATA1
  write_data 0x3304         [write word 0x3304 to REG[0x012]]
Step 3 : Set the start Address to 0x00010 and dummy read
  write_command 0x0000      [M/R#=0 => Register]
  write_command 0x1001      [start address = 0x00010,MODE_SL=0x01 => Word]
  read_data DUMMY          [first read cycle is dummy]
Step 4 : Read Back the Data [Total = 4 bytes]
  read_data DATA0         [read back value is 0x1122 => REG[0x010]=0x1122]
  read_data DATA1         [read back value is 0x3304 => REG[0x012]=0x3304]

```

Example 2 : Register Access with 16 bit indirect address mode (Big Endian, CNF4=1) with Byte Mode

(Write **Register** REG[0x010h]=0x11, REG[0x011h]=0x22, REG[0x012h]=0x33, REG[0x013h]=0x04 and read back the contents)

```

Step 1: Set the start address to 0x00010 and write DATA0
  write_command 0x0000      [M/R#=0 => Register]
  write_command 0x1000      [start address = 0x00010, MODE_SL=0x00 => Byte]
  write_data 0x1122         [write word 0x11 to REG[0x010]]
Step 2 : Write the DATA1-3
  write_data 0x3344         [write word 0x44 to REG[0x011]]
  write_data 0xAABB         [write word 0xAA to REG[0x012]]
  write_data 0xCC07         [write word 0x07 to REG[0x013]]
Step 3 : Set the start Address to 0x00010 and dummy read
  write_command 0x0000      [M/R#=0 => Register]
  write_command 0x1000      [start address = 0x00010,MODE_SL=0x00 => Byte]
  read_data DUMMY          [first read cycle is dummy]
Step 4 : Read Back the Data [Total = 4 bytes]
  read_data DATA0         [read back value is 0x1144 => REG[0x010]=0x11]
  read_data DATA1         [read back value is 0x1144 => REG[0x011]=0x44]
  read_data DATA2         [read back value is 0xAA07 => REG[0x012]=0xAA]
  read_data DATA3         [read back value is 0xAA07 => REG[0x013]=0x07]

```

Example 3 : Memory Access with 16 bit indirect address mode (Big Endian, CNF4=1) with Word Mode

(Write **Memory** AB[0x00910h]=0x11, AB[0x00911h]=0x22, AB[0x00912h]=0x33, AB[0x00913h]=0x44 and read back the contents)

Step 1: Set the start address to 0x00910 and write DATA0
write_command 0x8009 [M/R#=1 => Memory]
write_command 0x1001 [start address = 0x00910, MODE_SL=0x01 => Word]
write_data 0x1122 [write word 0x1122 to 0x00910]

Step 2 : Write the DATA1
write_data 0x3344 [write word 0x3344 to 0x00912]

Step 3 : Set the start Address to 0x00910 and dummy read
write_command 0x8009 [M/R#=1 => Memory]
write_command 0x1001 [start address = 0x00910,MODE_SL=0x01 => Word]
read_data DUMMY [first read cycle is dummy]

Step 4 : Read Back the Data [Total = 4 bytes]
read_data DATA0 [read back value is 0x1122 => AB[0x00910]=0x1122]
read_data DATA1 [read back value is 0x3344 => AB[0x00912]=0x3344]

Example 4 : Memory Access with 16 bit indirect address mode (Big Endian, CNF4=1) with Byte Mode

(Write **Memory** AB[0x00910h]=0x11, AB[0x00911h]=0x22, AB[0x00912h]=0x33, AB[0x00913h]=0x44 and read back the contents)

Step 1: Set the start address to 0x00910 and write DATA0
write_command 0x8009 [M/R#=1 => Memory]
write_command 0x1000 [start address = 0x00910, MODE_SL=0x00 => Byte]
write_data 0x1122 [write word 0x11 to AB[0x00910]]

Step 2 : Write the DATA1-3
write_data 0x3344 [write word 0x44 to AB[0x00911]]
write_data 0xAABB [write word 0xAA to AB[0x00912]]
write_data 0xCCDD [write word 0xDD to AB[0x00913]]

Step 3 : Set the start Address to 0x00910 and dummy read
write_command 0x8009 [M/R#=1 => Memory]
write_command 0x1000 [start address = 0x00910,MODE_SL=0x00 => Byte]
read_data DUMMY [first read cycle is dummy]

Step 4 : Read Back the Data [Total = 4 bytes]
read_data DATA0 [read back value is 0x1144 => AB[0x00910]=0x11]
read_data DATA1 [read back value is 0x1144 => AB[0x00911]=0x44]
read_data DATA2 [read back value is 0xAADD => AB[0x00912]=0xAA]
read_data DATA3 [read back value is 0xAADD => AB[0x00913]=0xDD]

Example 5 : Register Access with 8 bit indirect address mode (Big Endian, CNF4=1)

(Write **Register** REG[0x010h]=0x11, REG[0x011h]=0x22, REG[0x012h]=0x33, REG[0x013h]=0x04 and read back the contents)

Step 1: Set the start address to 0x00010 and write DATA0
write_command 0x00 [M/R#=0 => Register]
write_command 0x00
write_command 0x10 [start address = 0x00010]
write_data 0x11 [write word 0x11 to REG[0x010]]

Step 2 : Write the DATA1-3
write_data 0x22 [write word 0x22 to REG[0x011]]
write_data 0x33 [write word 0x33 to REG[0x012]]
write_data 0x04 [write word 0x04 to REG[0x013]]

Step 3 : Set the start Address to 0x00010 and dummy read
write_command 0x00 [M/R#=0 => Register]
write_command 0x00
write_command 0x10 [start address = 0x00010]
read_data DUMMY [first read cycle is dummy]

Step 4 : Read Back the Data [Total = 4 bytes]
read_data DATA0 [read back value is 0x11 => REG[0x010]=0x11]
read_data DATA1 [read back value is 0x22 => REG[0x011]=0x22]
read_data DATA2 [read back value is 0x33 => REG[0x012]=0x33]
read_data DATA3 [read back value is 0x04 => REG[0x013]=0x04]

Example 6 : Memory Access with 8 bit indirect address mode (Big Endian, CNF4=1)

(Write **Memory** AB[0x00910h]=0x11, AB[0x00911h]=0x22, AB[0x00912h]=0x33, AB[0x00913h]=0x44 and read back the contents)

Step 1: Set the start address to 0x00910 and write DATA0
write_command 0x80 [M/R#=1 => Memory]
write_command 0x09
write_command 0x10 [start address = 0x00910]
write_data 0x11 [write word 0x11 to AB[0x00910]]

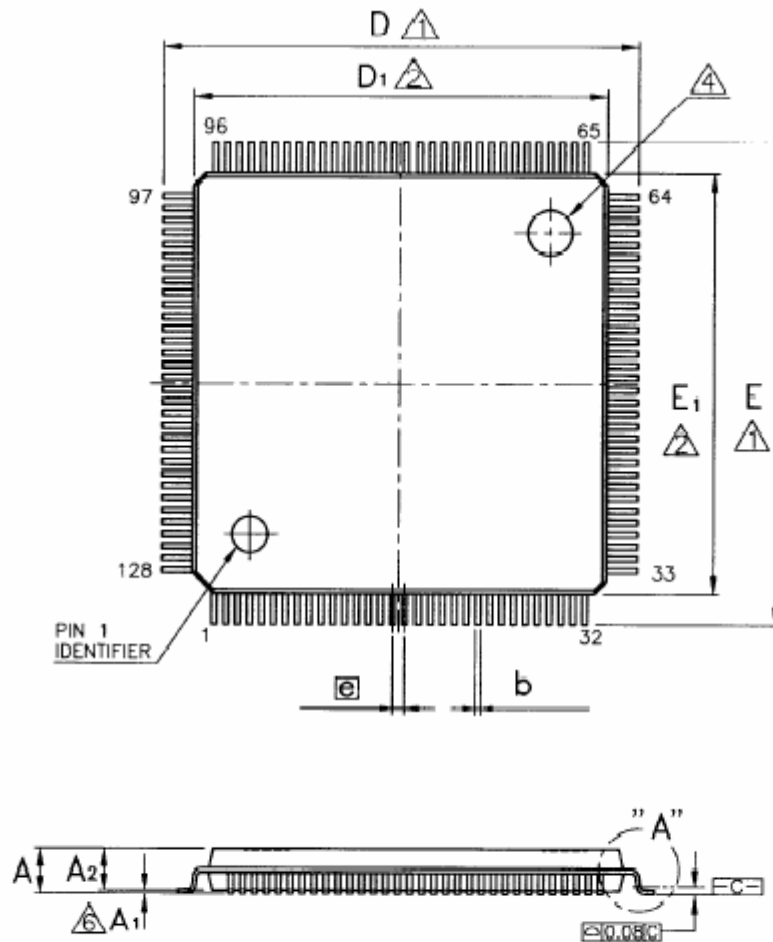
Step 2 : Write the DATA1-3
write_data 0x22 [write word 0x22 to AB[0x00911]]
write_data 0x33 [write word 0x33 to AB[0x00912]]
write_data 0x44 [write word 0x44 to AB[0x00913]]

Step 3 : Set the start Address to 0x00910 and dummy read
write_command 0x80 [M/R#=1 => Memory]
write_command 0x09
write_command 0x10 [start address = 0x00910]
read_data DUMMY [first read cycle is dummy]

Step 4 : Read Back the Data [Total = 4 bytes]
read_data DATA0 [read back value is 0x11 => AB[0x00910]=0x11]
read_data DATA1 [read back value is 0x22 => AB[0x00911]=0x22]
read_data DATA2 [read back value is 0x33 => AB[0x00912]=0x33]
read_data DATA3 [read back value is 0x44 => AB[0x00913]=0x44]


13 PACKAGE INFORMATION

13.1 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 | |

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.

All Solomon Systech Products complied with six (6) hazardous substances limitation requirement per European Union (EU) “Restriction of Hazardous Substance (RoHS) Directive (2002/95/EC)” and China standard “电子信息产品污染控制标识要求 (SJ/T11364-2006)” with control Marking Symbol . Hazardous Substances test report is available upon requested.

With collaboration of <https://www.displayfuture.com>