S1D13742 Mobile Graphics Engine

Hardware Functional Specification

Document Number: X63A-A-001-06.6

SEIKO EPSON CORPORATION

Rev. 6.6

NOTICE

No part of this material may be reproduced or duplicated in any form or by any means without the written permission of Seiko Epson. Seiko Epson reserves the right to make changes to this material without notice. Seiko Epson does not assume any liability of any kind arising out of any inaccuracies contained in this material or due to its application or use in any product or circuit and, further, there is no representation that this material is applicable to products requiring high level reliability, such as, medical products. Moreover, no license to any intellectual property rights is granted by implication or otherwise, and there is no representation or warranty that anything made in accordance with this material will be free from any patent or copyright infringement of a third party. When exporting the products or technology described in this material, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You are requested not to use, to resell, to export and/or to otherwise dispose of the products (and any technical information furnished, if any) for the development and/or manufacture of weapon of mass destruction or for other military purposes.

All brands or product names mentioned herein are trademarks and/or registered trademarks of their respective companies.

Table of Contents

1 Introduction

1.1 Scope

This is the Hardware Functional Specification for the S1D13742 Embedded Memory LCD Controller. Included in this document are timing diagrams, AC and DC characteristics, register descriptions, and power management descriptions. This document is intended for two audiences: Video Subsystem Designers and Software Developers.

This document is updated as appropriate. Please check the Epson Research and Development Website at vdc.epson.com for the latest revision of this document before beginning any development.

We appreciate your comments on our documentation. Please contact us via email at vdc-documentation@ea.epson.com.

1.2 Overview Description

The S1D13742 is a color LCD graphics controller with an embedded 768K byte display buffer. The S1D13742 supports a 8/16-bit Intel 80 CPU architecture while providing high performance bandwidth into display memory allowing for fast screen updates.

Products requiring a rotated display image can take advantage of the SwivelView™ feature which provides hardware rotation of the display memory transparent to the software application. Resolutions supported include 800x480 single buffered and 352x416 double buffered.

The S1D13742 uses a double-buffer architecture to prevent any visual tearing during streaming video screen updates.

2 Features

2.1 Integrated Frame Buffer

- Embedded 768K byte SRAM display buffer.

2.2 CPU Interface

- 8/16-bit Intel 80 interface (used for display or register data).
- Chip select is used to select device. When inactive, any input data/command will be ignored.

2.3 Input Data Formats

- RGB: 8:8:8, 6:6:6, 5:6:5 (8:8:8 will be truncated to 16 or 18 bpp).
- YUV 4:2:2, 4:2:0 (Internal YUV to RGB Converter stored as 16 or 18 bpp).

Note
All input data must be internally converted to the same format before being stored in the display buffer. Different data types can not be mixed within a common display buffer.

2.4 Display Support

- Active Matrix TFT interface.
  - 18/36-bit interface.
  - Supports resolutions up to 800x480.

2.5 Display Modes

- 16/18 bit-per-pixel (bpp) color depths.
- 16 bpp to 18 bpp conversion: Input data can be converted from 16 bpp to 18 bpp in one of three ways.
  1. RGB (5:6:5) msb copying to create new lsb for the Red and Blue components. This conversion is done prior to storing in memory, as this allows for 16 bpp and 18 bpp input data to be mixed.
  2. Gamma Correction Look-Up-Tables: there are three, 64 position, 8-bit wide LUT's. The data stored in memory can be used as an index into these tables. The LUT's are placed on the display side and therefore do not affect the data stored in memory.
  3. RGB (5:6:5) stored in memory: LUT is by-passed. Copy msb to lsb for red and blue during the display read from memory.

2.6 Display Features

- All display writes will be handled by window apertures/position for complete or partial display updates. All window coordinates are referenced to top left corner of the displayed image (even in a rotated display, the top-left corner is maintained and no host side translation need take place).
- SwivelView™: 90°, 180°, 270° counter-clockwise hardware rotation of display image. All displayed windows can have independent rotation. No additional programming necessary when enabling these modes.
- Double-Buffer available to prevent image tearing during streaming input. Resolutions supported must fit inside 384K bytes (½ of total available display buffer). Typical resolution of 352x416.
- Pixel Doubling: Horizontal and Vertical averaging for smooth doubling of a single window.
- Pixel Halving: no limitation on number of windows.

2.7 Clock Source

- Internal programmable PLL.
- Single MHz clock input: CLKI.
- CLKI available as CLKOUT (separate CLKOUTEN pin associated with output).
  - output state = 0 when disabled.

2.8 Miscellaneous

- Hardware / Software Power Save mode.
- Input pin to Enable/Disable Power Save Mode.
- General Purpose Input/Output pins are available (GPIO[7:0]).
  - INT pin associated with selectable GPIO inputs.
- Package: QFP20 144-pin package

3 Block Diagram

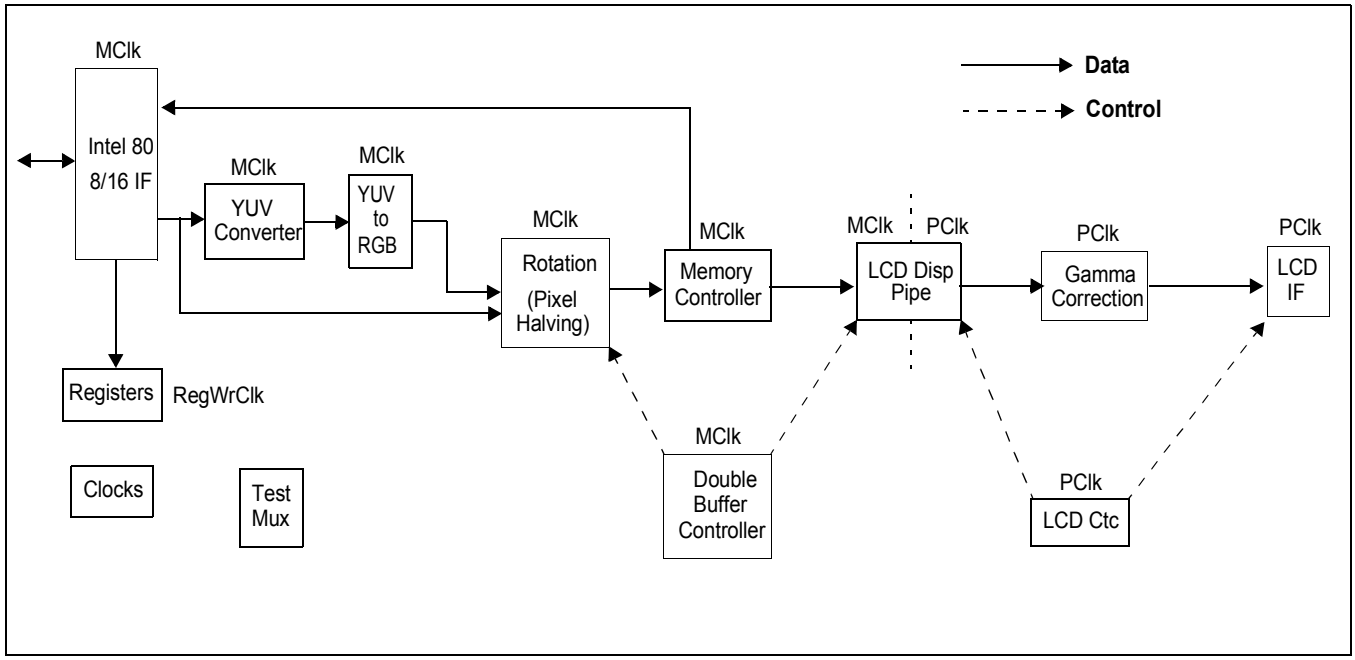


Figure 3-1: Block Diagram

4 Pinout Diagram

4.1 Pin-Out

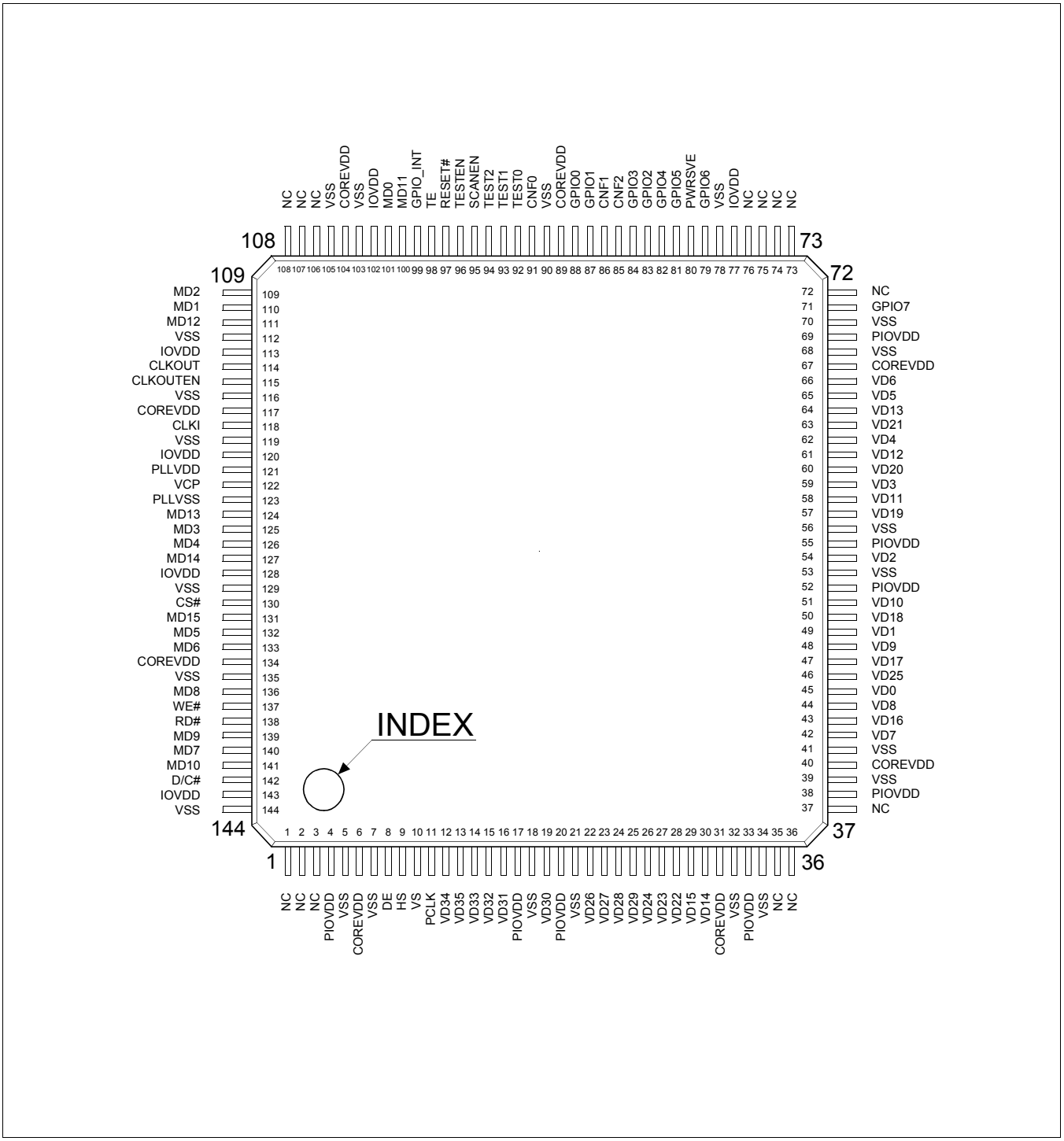


Figure 4-1: S1D13742 QFP20 Pinout (Top View)

4.2 Pin Descriptions

Key:

Pin Types

I	=	Input
O	=	Output
IO	=	Bi-Directional (Input/Output)
P	=	Power pin

RESET# / Power Save Status

H	=	High level output
L	=	Low level output
Hi-Z	=	High Impedance

Table 4-1: Cell Description

Item	Description
HI	H System[1] LVCMOS[3] Input Buffer
HIS	H System LVCMOS Schmitt Input Buffer
HID	H System LVCMOS Input Buffer with pull-down resistor
HO	H System LVCOMOS Output buffer
HB	H System LVCMOS Bidirectional Buffer
HBD	H System LVCMOS Bidirectional Buffer with pull-down resistor
HB_DSEL	H System LVCMOS Bidirectional Buffer with Drive Selector
LIDS	L System[2] LVCMOS Schmitt Input Buffer with pull-down resistor
LITR	L System Transparent Input Buffer

[1] H System is IOVDD and PIOVDD (see Section 6, “D.C. Characteristics”).
[2] L System is COREVDD (see Section 6, “D.C. Characteristics”).
[3] LVCMOS is Low Voltage CMOS (see Section 6, “D.C. Characteristics”).

4.2.1 Intel 80 Host Interface

Table 4-2: Host Interface Pin Descriptions

Pin Name	Type	QFP Pin #	Cell	IO Voltage	RESET# State	Power Save Status	Description
MD[15:0]	IO	131,127, 124,111, 100,141, 139,136, 140,133, 132,126, 125,109, 110,101	HB	IOVDD	Hi-Z	Hi-Z	Intel 80 Data lines. • For the S1D13742B00, when the 8-bit bus interface is selected by CNF1, MD[15:8] are pulled low by internal resistors. • For the S1D13742B01, when the 8-bit bus interface is selected by CNF1, MD[15:8] should be connected to VSS. **Note:** The Host Data lines can be swapped (i.e. MD15 = MD0) using the CNF0 pin. For details, see Section 4.3, “Summary of Configuration Options” on page 17.
WE#	I	137	HI	IOVDD	Input	Input	This input pin is the Write Enable signal.
RD#	I	138	HI	IOVDD	Input	Input	This input pin is the Read Enable signal.
CS#	I	130	HI	IOVDD	Input	Input	This input pin is the Chip Select signal.
D/C#	I	142	HI	IOVDD	Input	Input	This input pin is used to select between Intel 80 address and data
TE	O	98	HO	IOVDD	L	L	Tearing Effect: this pin will reflect the VSYNC, HSYNC or the OR’d combination status of the display.
GPIO_INT	O	99	HO	IOVDD	L	Output	This interrupt pin is associated with selected GPIO pins when configured as inputs or outputs. Interrupt functionality is not affected by Power Save. See Section 9.3.10, “General Purpose IO Pins Registers” on page 71 for operational description.
RESET#	I	97	HI	IOVDD	Input	Input	Active low input to set all internal registers to the default state and to force all signals to their inactive states.

4.2.2 LCD Interface

Table 4-3: LCD Interface Pin Descriptions

Pin Name	Type	QFP Pin #	Cell	IO Voltage	RESET# State	Power Save Status	Description
VD[35:0]	IO	13,12,14,15, 16,19,25,24, 23,22,46,26, 27,28,63,60, 57,50,47,43, 29,30,64,61, 58,51,48,44, 42,66,65,62, 59,54,49,45	HB_ DSEL	PIOVDD	L	L	Panel Data bits 35-0. VD[35:0] are used for all modes. In 2 pixels/clock mode, VD[17:0] represent the 1st pixel sent in a 2 pixel/clock operation. **Note:** The Panel Data Lines can be swapped (i.e. VD23 = VD0) using the VD Data Swap bit, REG[14h] bit 7. **Note:** The VD output drive is selectable between 2.5mA and 6.5mA using the CNF2 pin. For details, see Section 4.3, "Summary of Configuration Options" on page 17.
VS	O	10	HO	PIOVDD	H	L	This output pin is the Vertical Sync pulse
HS	O	9	HO	PIOVDD	H	L	This output is the Horizontal Sync pulse
PCLK	O	11	HO	PIOVDD	CLKI	L	This output pin is the Data Clock
DE	O	8	HO	PIOVDD	L	L	This output pin is the Data Enable

Note
The LCD interface requires a separate power rail (PIOVDD) to support the configurable IO drive. For details, see the CNF2 description in Section 4.3, "Summary of Configuration Options" on page 17.

Note
Input of VD[35:0] is used for production test only.

4.2.3 Clocks

Table 4-4: Clock Input Pin Descriptions

Pin Name	Type	QFP Pin #	Cell	IO Voltage	RESET# State	Power Save Status	Description
CLKI	I	118	HIS	IOVDD	Input	Input	MHz input for PLL operation or MHz input if PLL is bypassed Input frequency range: 1MHz ~ 33MHz
CLKOUT	O	114	HO	IOVDD	L	CLKI	This output pin represents the CLKI pin if enabled by CLKOUTEN. When disabled the output is low. **Note:** this output is not affected by the various power save modes
CLKOUTEN	I	115	HI	IOVDD	Input	Input	This pin enables/disables the CLKOUT pin.

4.2.4 Miscellaneous

Table 4-5: Miscellaneous Pin Descriptions

Pin Name	Type	QFP Pin #	Cell	IO Voltage	RESET# State	Power Save Status	Description
CNF[2:0]	I	85,86,91	HI	IOVDD	Input	Input	These inputs are used for power-up configuration. For details, see Section 4.3, “Summary of Configuration Options” on page 17. **Note: These pins must be connected directly to IOVDD or VSS.**
TESTEN	I	96	LIDS	IOVDD	—	—	Test Enable input used for production test only This pin should be left unconnected for normal use.
GPIO[7:0]	IO	71,79,81, 82,84,83, 87,88	HBD	IOVDD	L	Pull Down Active	These pins are general purpose input/output pins. These pins have internal pull-down resistors which can be controlled using REG[64h].
PWRSVE	I	80	HI	IOVDD	Input	Input	This pin enables/disables the Standby Power Save Mode When unused this pin must be connected to VSS.
TEST[2:0]	I	94,93,92	HID	IOVDD	—	—	These are Test Function pins and are used for production test only. These pins should be left unconnected for normal operation.
SCANEN	I	95	HID	IOVDD	—	—	This is the Test Scan Enable input and is used for production test only. This pin should be left unconnected for normal operation.
VCP	I	122	LITR	PLLVDD	—	—	This is the PLL VCP Test pin and is used for production test only. This pin should be left unconnected for normal operation.
NC	—	1,2,3, 35,36, 37,72, 73,74, 75,76, 106,107, 108	—	—	—	—	These pins are not connected.

4.2.5 Power And Ground

Table 4-6: Power And Ground Pin Descriptions

Pin Name	Type	QFP Pin #	Cell	Description
COREVDD	P	6,31,40,67,89, 104,117,134	P	Core power supply
IOVDD	P	77,102,113, 120,128,143	P	IO power supply for the host interface
PIOVDD	P	4,17,20,33,38, 52,55,69	P	IO power supply for the panel interface
PLLVDD	P	121	P	PLL power supply
PLLVSS	P	123	P	GND for PLL
VSS	P	5,7,18,21,32, 34,39,41,53, 56,68,70,78, 90,103,105, 112,116,119, 129, 135,144	P	GND

4.3 Summary of Configuration Options

These pins are used for power-up configuration and must be connected directly to IOVDD or VSS. The state of CNF[2:0] may be changed at any time.

Table 4-7: Summary of Power-On/Reset Options

Configuration Input	Power-On/Reset State	
	1 (connected to IOVDD)	0 (Connected to VSS)
CNF0	Host Data Lines are normal: If CNF1 = 1, then D15 = D15, etc. If CNF1 = 0, then D7 = D7, etc.	Host Data Lines are swapped: If CNF1 = 1, then D15 = D0, etc. If CNF1 = 0, then D7 = D0, etc.
CNF1	Host Data is 16-bit	Host Data is 8-bit
CNF2	PIOVDD output current (I_{OL2}) = 6.5mA	PIOVDD output current (I_{OL2}) = 2.5mA

Note

When CNF1=0, all Register access is 8-bit only.

When CNF1 =1 (16-bit): All Register access is 8-bit ONLY (the most significant byte on the data bus is ignored) except the Memory Data Port. Access to the Memory Data Port is 16-bit.

5 Pin Mapping

5.1 Intel 80 Data Pins

This function is controlled by CNF [1:0]

Table 5-1: S1D13742B00 Intel 80 Data Pin Mapping

Pin Name	16-Bit Data No Swap (CNF1=1, CNF0=1)	16-Bit Data Swapped (CNF1=1, CNF0=0)	8-Bit Data No Swap (CNF1=0, CNF0=1)	8-Bit Data Swapped (CNF1=0, CNF0=0)
MD15	MD15	MD0	Pulled Low by Internal Resistor	Pulled Low by Internal Resistor
• • •	• • •	• • •	• • •	• • •
MD8	MD8	MD7	Pulled Low by Internal Resistor	Pulled Low by Internal Resistor
MD7	MD7	MD8	MD7	MD0
• • •	• • •	• • •	• • •	• • •
MD0	MD0	MD15	MD0	MD7

Table 5-2: S1D13742B01 Intel 80 Data Pin Mapping

Pin Name	16-Bit Data No Swap (CNF1=1, CNF0=1)	16-Bit Data Swapped (CNF1=1, CNF0=0)	8-Bit Data No Swap (CNF1=0, CNF0=1)	8-Bit Data Swapped (CNF1=0, CNF0=0)
MD15	MD15	MD0	Hi-Z	Hi-Z
• • •	• • •	• • •	• • •	• • •
MD8	MD8	MD7	Hi-Z	Hi-Z
MD7	MD7	MD8	MD7	MD0
• • •	• • •	• • •	• • •	• • •
MD0	MD0	MD15	MD0	MD7

5.2 LCD Interface Pin Mapping

Table 5-3: LCD Interface Pin Mapping for Mode 1 and Mode 2

Pin Name	16bpp				18bpp			
	Single (18-bit)		Double (36-bit)		Single (18-bit)		Double (36-bit)	
	Normal	Swap	Normal	Swap	Normal	Swap	Normal	Swap
VS	Vertical Sync							
HS	Horizontal Sync							
PCLK	Pixel Clock							
DE	Data Enable							
VD0	B4	R4	B4	R4	B0	R5	B0	R5
VD1	B0	R3	B0	R3	B1	R4	B1	R4
VD2	B1	R2	B1	R2	B2	R3	B2	R3
VD3	B2	R1	B2	R1	B3	R2	B3	R2
VD4	B3	R0	B3	R0	B4	R1	B4	R1
VD5	B4	R4	B4	R4	B5	R0	B5	R0
VD6	G0	G5	G0	G5	G0	G5	G0	G5
VD7	G1	G4	G1	G4	G1	G4	G1	G4
VD8	G2	G3	G2	G3	G2	G3	G2	G3
VD9	G3	G2	G3	G2	G3	G2	G3	G2
VD10	G4	G1	G4	G1	G4	G1	G4	G1
VD11	G5	G0	G5	G0	G5	G0	G5	G0
VD12	R4	B4	R4	B4	R0	B5	R0	B5
VD13	R0	B3	R0	B3	R1	B4	R1	B4
VD14	R1	B2	R1	B2	R2	B3	R2	B3
VD15	R2	B1	R2	B1	R3	B2	R3	B2
VD16	R3	B0	R3	B0	R4	B1	R4	B1
VD17	R4	B4	R4	B4	R5	B0	R5	B0
VD18	driven 0	driven 0	B4	R4	driven 0	driven 0	B0	R5
VD19	driven 0	driven 0	B0	R3	driven 0	driven 0	B1	R4
VD20	driven 0	driven 0	B1	R2	driven 0	driven 0	B2	R3
VD21	driven 0	driven 0	B2	R1	driven 0	driven 0	B3	R2
VD22	driven 0	driven 0	B3	R0	driven 0	driven 0	B4	R1
VD23	driven 0	driven 0	B4	R4	driven 0	driven 0	B5	R0
VD24	driven 0	driven 0	G0	G5	driven 0	driven 0	G0	G5
VD25	driven 0	driven 0	G1	G4	driven 0	driven 0	G1	G4
VD26	driven 0	driven 0	G2	G3	driven 0	driven 0	G2	G3
VD27	driven 0	driven 0	G3	G2	driven 0	driven 0	G3	G2
VD28	driven 0	driven 0	G4	G1	driven 0	driven 0	G4	G1
VD29	driven 0	driven 0	G5	G0	driven 0	driven 0	G5	G0
VD30	driven 0	driven 0	R4	B4	driven 0	driven 0	R0	B5
VD31	driven 0	driven 0	R0	B3	driven 0	driven 0	R1	B4
VD32	driven 0	driven 0	R1	B2	driven 0	driven 0	R2	B3
VD33	driven 0	driven 0	R2	B1	driven 0	driven 0	R3	B2
VD34	driven 0	driven 0	R3	B0	driven 0	driven 0	R4	B1
VD35	driven 0	driven 0	R4	B4	driven 0	driven 0	R5	B0

5.3 LCD Interface Data Pins

This function is controlled by REG[14h] bit 7.

Table 5-4: LCD Interface Data Pin Mapping

Pin Name	36-Bit Data No Swap REG[14] b7=0	36-Bit Data Swapped REG[14] b7=1	18-Bit Data No Swap REG[14] b7=0	18-Bit Data Swapped REG[14] b7=1
VD35	VD35	VD0	Driven Low	Driven Low
• • •	• • •	• • •	• • •	• • •
VD18	VD18	VD17	Driven Low	Driven Low
VD17	VD17	VD18	VD17	VD0
• • •	• • •	• • •	• • •	• • •
VD0	VD0	VD35	VD0	VD17

6 D.C. Characteristics

6.1 Absolute Maximum Ratings

Table 6-1: Absolute Maximum Ratings

Symbol	Parameter	Rating	Units
Core V_{DD}	Core Supply Voltage	VSS - 0.3 ~ 2.0	V
PLL V_{DD}	PLL Supply Voltage	VSS - 0.3 ~ 2.0	V
IO V_{DD}	Host IO Supply Voltage	COREVDD ~ 4.0	V
PIO V_{DD}	Panel IO Supply Voltage	COREVDD ~ 4.0	V
V_{IN}	Input Signal Voltage	VSS - 0.3 ~ IOVDD + 0.3	V
V_{OUT}	Output Signal Voltage	VSS - 0.3 ~ IOVDD + 0.3	V
I_{OUT}	Output Signal Current	±10	mA

6.2 Recommended Operating Conditions

Table 6-2: Recommended Operating Conditions

Symbol	Parameter	Condition	Min	Typ	Max	Units
Core V_{DD}	Core Supply Voltage	VSS = 0 V	1.40	1.50	1.60	V
PLL V_{DD}	PLL Supply Voltage	VSS = 0 V	1.40	1.50	1.60	V
IO V_{DD}	Host IO Supply Voltage	VSS = 0 V	1.65	—	3.6	V
PIO V_{DD}	Panel IO Supply Voltage	VSS = 0 V	1.65	—	3.6	V
V_{IN}	Input Voltage	—	VSS	—	IOVDD	V
T_{OPR}	Operating Temperature	—	-40	+25	+85	°C
T_{stg}	Storage Temperature	—	-65		+150	°C

Note

There are no special Power On/Off requirements with respect to sequencing the various VDD pins. There are also no special requirements for the IO signals, however Inputs should not be floating. If the input signals were to power up in a valid cycle, the S1D13742 would decode the cycle.

6.3 Electrical Characteristics

The following characteristics are for: IOVDD. VSS = 0V, T_{OPR} = -40 to +85°C.

Table 6-3: Electrical Characteristics for IOVDD or PIOVDD = 1.8V ± 0.15V

Symbol	Parameter	Condition	Min	Typ	Max	Units
I_{QALL}	Quiescent Current	CLKI stopped (grounded), Sleep Mode enabled, all power supplies active	—	100	—	µA
I_{PLL}	PLL Current	f_{PLL} = 54MHz	—	500	1000	µA
I_{CORE}	Operation Peak Current	COREVDD Power Pin	—	—	62	mA
P_{CORE}	Core Typical Operating Power	see Note 1	—	9.15	—	mW
P_{PLL}	PLL Typical Operating Power		—	0.7	—	mW
P_{PIO}	PIO Typical Operating Power		—	2.8	—	mW
P_{HIO}	HIO Typical Operating Power		—	0.018	—	mW
P_{CORE}	Core Typical Operating Power	see Note 2	—	10.9	—	mW
P_{PLL}	PLL Typical Operating Power		—	0.77	—	mW
P_{PIO}	PIO Typical Operating Power		—	2.124	—	mW
P_{HIO}	HIO Typical Operating Power		—	0.001	—	mW
I_{IZ}	Input Leakage Current	—	-5	—	5	µA
I_{OZ}	Output Leakage Current	—	-5	—	5	µA
IOV_{OH2}	High Level Output Voltage	IOV_{DD} = min I_{OH2} = -2.5mA	IOVDD - 0.40	—	IOVDD	V
$PIOV_{OH2}$	High Level Output Voltage	PIOVDD = min I_{OH2} = -2.5mA	PIOVDD - 0.40	—	PIOVDD	V
$PIOV_{OH4}$	High Level Output Voltage	PIOVDD = min I_{OH2} = -6.5mA	PIOVDD - 0.40	—	PIOVDD	V
IOV_{OL2}	Low Level Output Voltage	IOVDD = min I_{OL2} = 2.5mA	VSS	—	0.40	V
$PIOV_{OL2}$	Low Level Output Voltage	PIOVDD = min I_{OL2} = 2.5mA	VSS	—	0.40	V
$PIOV_{OL4}$	Low Level Output Voltage	PIOVDD = min I_{OL2} = 6.5mA	VSS	—	0.40	V
IOV_{IH}	High Level Input Voltage	CMOS Input	1.27	—	—	V
$PIOV_{IH}$	High Level Input Voltage	CMOS Input	1.27	—	—	V
IOV_{IL}	Low Level Input Voltage	CMOS Input	—	—	0.57	V
$PIOV_{IL}$	Low Level Input Voltage	CMOS Input	—	—	0.57	V
IOV_{T+}	Positive Trigger Voltage	CMOS Schmitt	0.57	—	1.56	V
IOV_{T-}	Negative Trigger Voltage	CMOS Schmitt	0.33	—	1.27	V
IO V_H	Hysteresis Voltage	CMOS Schmitt	0.24	—	—	V
R_{PU1}	Pull-Up Resistance Type1	V_I = VSS	40	100	240	kΩ
R_{PD1}	Pull-Down Resistance Type1	V_I = VDD	40	100	240	kΩ
R_{PU2}	Pull-Up Resistance Type2	V_I = VSS	80	200	480	kΩ
R_{PD2}	Pull-Down Resistance Type2	V_I = VDD	80	200	480	kΩ
C_{IO}	Pin Capacitance	f = 1MHz, VDD = 0V	—	—	8	pF

Note

1. Typical Operating Current Environment:
   352x416 K2 TFT panel with PCLK divide by 4. SYSCLK=48.5MHz from PLL, PLL Source from 19.2MHz CLKI input. 18bpp memory storage.
   COREVDD and PLLVDD to 1.5V, HIOVDD, PIOVDD to 1.8V
2. Typical Operating Current Environment:
   800 x 480 TFT panel with PCLK divide by 3. SYSCLK= 59MHz from PLL, PLL Source from 12MHz CLKI input. 16bpp memory storage.
   COREVDD and PLLVDD to 1.5V, HIOVDD, PIOVDD to 1.8V

The following characteristics are for: IOVDD. VSS = 0V, T_{OPR} = -40 to +85°C.

Table 6-4: Electrical Characteristics for IOVDD or PIOVDD = 2.8V ± 0.14V

Symbol	Parameter	Condition	Min	Typ	Max	Units
I_{QALL}	Quiescent Current	CLKI stopped (grounded), Sleep Mode enabled, all power supplies active	—	120	—	µA
I_{PLL}	PLL Current	f_{PLL} = 54MHz	—	500	1000	µA
I_{CORE}	Operation Peak Current	COREVDD Power Pin	—	—	62	mA
I_{IZ}	Input Leakage Current	—	-5	—	5	µA
I_{OZ}	Output Leakage Current	—	-5	—	5	µA
IOV_{OH2}	High Level Output Voltage	IOV_{DD} = min I_{OH2} = -3.6mA	IOVDD - 0.40	—	IOVDD	V
$PIOV_{OH2}$	High Level Output Voltage	PIOVDD = min I_{OH2} = -3.6mA	PIOVDD - 0.40	—	PIOVDD	V
$PIOV_{OH4}$	High Level Output Voltage	PIOVDD = min I_{OH2} = -10.8mA	PIOVDD - 0.40	—	PIOVDD	V
IOV_{OL2}	Low Level Output Voltage	IOVDD = min I_{OL2} = 3.6mA	VSS	—	0.40	V
$PIOV_{OL2}$	Low Level Output Voltage	PIOVDD = min I_{OL2} = 3.6mA	VSS	—	0.40	V
$PIOV_{OL4}$	Low Level Output Voltage	PIOVDD = min I_{OL2} = 10.8mA	VSS	—	0.40	V
IOV_{IH}	High Level Input Voltage	CMOS Input	1.75	—	—	V
$PIOV_{IH}$	High Level Input Voltage	CMOS Input	1.75	—	—	V
IOV_{IL}	Low Level Input Voltage	CMOS Input	—	—	0.70	V
$PIOV_{IL}$	Low Level Input Voltage	CMOS Input	—	—	0.70	V
IOV_{T+}	Positive Trigger Voltage	CMOS Schmitt	0.93	—	2.36	V
IOV_{T-}	Negative Trigger Voltage	CMOS Schmitt	0.53	—	1.92	V
IO V_H	Hysteresis Voltage	CMOS Schmitt	0.40	—	—	V
R_{PU1}	Pull-Up Resistance Type1	V_I = VSS	24	60	144	kΩ
R_{PD1}	Pull-Down Resistance Type1	V_I = VDD	24	60	144	kΩ
R_{PU2}	Pull-Up Resistance Type2	V_I = VSS	48	120	288	kΩ
R_{PD2}	Pull-Down Resistance Type2	V_I = VDD	48	120	288	kΩ
C_{IO}	Pin Capacitance	f = 1MHz, VDD = 0V	—	—	8	pF

Note

1. Typical Operating Current Environment:
   352x416 K2 TFT panel with PCLK divide by 4. SYSCLK=48.5MHz from PLL, PLL Source from 19.2MHz CLKI input. 18bpp memory storage.
   COREVDD and PLLVDD to 1.5V, HIOVDD, PIOVDD to 2.8V
2. Typical Operating Current Environment:
   800 x 480 TFT panel with PCLK divide by 3. SYSCLK= 59MHz from PLL, PLL Source from 12MHz CLKI input. 16bpp memory storage.
   COREVDD and PLLVDD to 1.5V, HIOVDD, PIOVDD to 2.8V

The following characteristics are for: IOVDD, VSS = 0V, T_{OPR} = -40 to +85°C.

Table 6-5: Electrical Characteristics for IOVDD or PIOVDD = 3.3V ± 0.3V

Symbol	Parameter	Condition	Min	Typ	Max	Units
I_{QALL}	Quiescent Current	Quiescent Conditions	—	160	—	µA
I_{PLL}	PLL Current	f_{PLL} = 54MHz	—	500	1000	µA
I_{CORE}	Operation Peak Current	COREVDD Power Pin	—	—	62	mA
I_{IZ}	Input Leakage Current	—	-5	—	5	µA
I_{OZ}	Output Leakage Current	—	-5	—	5	µA
IOV_{OH2}	High Level Output Voltage	IOV_{DD} = min I_{OH2} = -4.0mA	IOVDD - 0.40	—	IOVDD	V
$PIOV_{OH2}$	High Level Output Voltage	PIOVDD = min I_{OH2} = -4.0mA	PIOVDD - 0.40	—	PIOVDD	V
$PIOV_{OH4}$	High Level Output Voltage	PIOVDD = min I_{OH2} = -12.0mA	PIOVDD - 0.40	—	PIOVDD	V
IOV_{OL2}	Low Level Output Voltage	IOVDD = min I_{OL2} = 4.0mA	VSS	—	0.40	V
$PIOV_{OL2}$	Low Level Output Voltage	PIOVDD = min I_{OL2} = 4.0mA	VSS	—	0.40	V
$PIOV_{OL4}$	Low Level Output Voltage	PIOVDD = min I_{OL2} = 12.0mA	VSS	—	0.40	V
IOV_{IH}	High Level Input Voltage	CMOS Input	2.20	—	—	V
$PIOV_{IH}$	High Level Input Voltage	CMOS Input	2.20	—	—	V
IOV_{IL}	Low Level Input Voltage	CMOS Input	—	—	0.80	V
$PIOV_{IL}$	Low Level Input Voltage	CMOS Input	—	—	0.80	V
IOV_{T+}	Positive Trigger Voltage	CMOS Schmitt	1.40	—	2.70	V
IOV_{T-}	Negative Trigger Voltage	CMOS Schmitt	0.60	—	1.80	V
IO V_H	Hysteresis Voltage	CMOS Schmitt	0.45	—	—	V
R_{PU1}	Pull-Up Resistance Type1	V_I = VSS	20	50	120	kΩ
R_{PD1}	Pull-Down Resistance Type1	V_I = VDD	20	50	120	kΩ
R_{PU2}	Pull-Up Resistance Type2	V_I = VSS	40	100	240	kΩ
R_{PD2}	Pull-Down Resistance Type2	V_I = VDD	40	100	240	kΩ
C_{IO}	Pin Capacitance	f = 1MHz, VDD = 0V	—	—	8	pF

7 A.C. Characteristics

Conditions:

IOVDD = PIOVDD = 1.8V ± 0.15V or 2.8V ± 0.14V
T_A = -40° C to 85° C
T_{rise} and T_{fall} for all inputs except Schmitt and CLKI must be ≤ 50 ns (10% ~ 90%)
T_{rise} and T_{fall} for all Schmitt must be ≤ 5 ms (10% ~ 90%)
C_L = 8pF ~ 30pF (MD[15:0])
C_L = 15pF (TE, GPIO_INT, CLKOUT)
C_L = 30pF (LCD Panel/GPIO Interface)

7.1 Clock Timing

7.1.1 Input Clocks

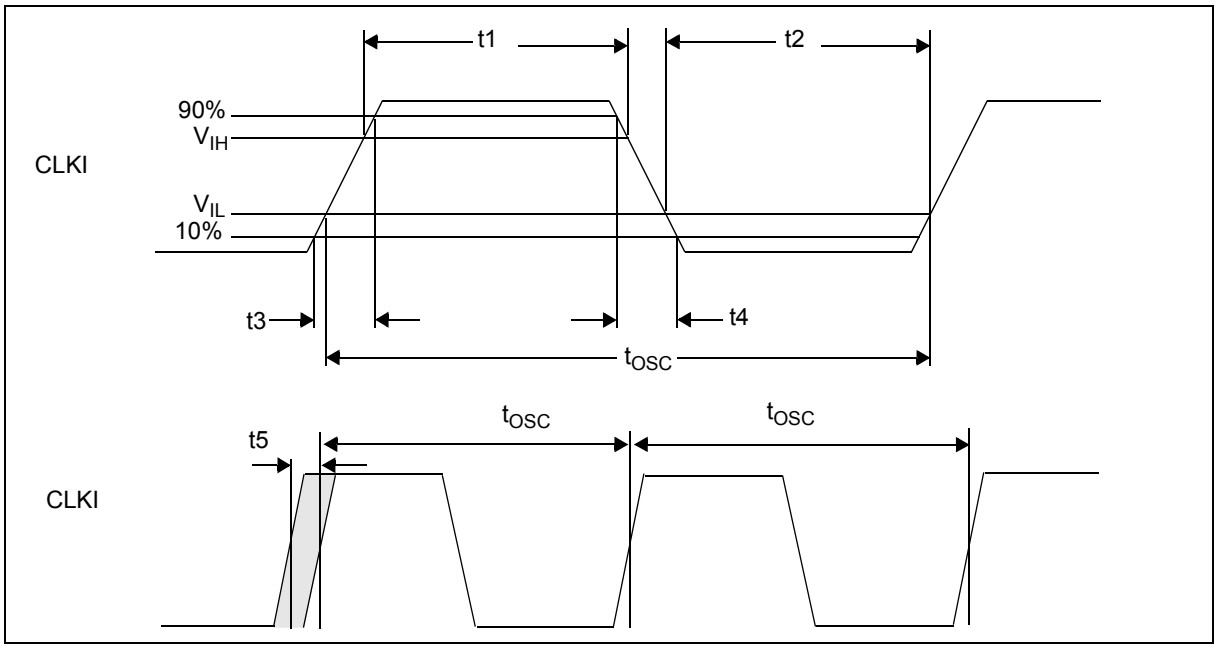


Figure 7-1 Clock Input Required (CLKI)

Table 7-1 Clock Input Requirements (CLKI)

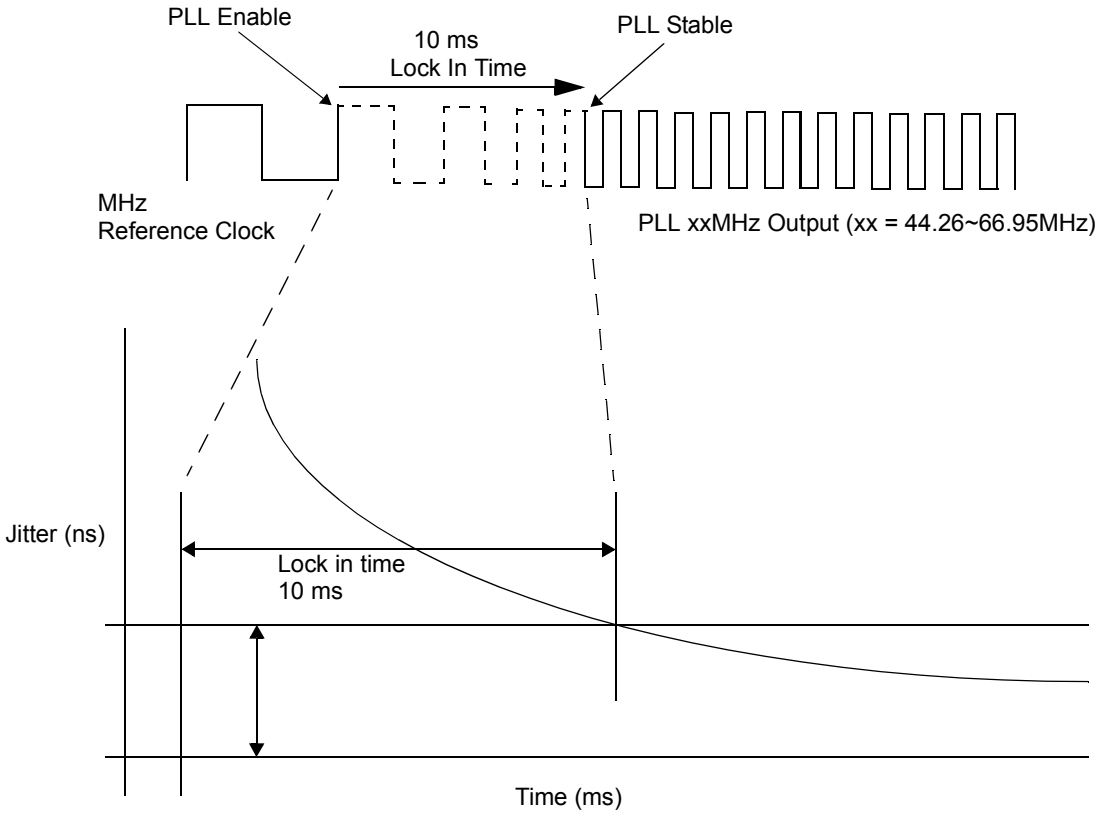
Symbol	Parameter	Min	Typ	Max	Units
f_{OSC} (see note 6)	Input clock frequency - PLL used for System Clock	1	—	66	MHz
	Input clock frequency - CLKI used for System Clock	0	—	68.90	MHz
t_{OSC}	Input clock period	—	$1/f_{OSC}$	—	µs
t1	Input clock pulse width high	$0.4t_{OSC}$	—	$0.6t_{OSC}$	µs
t2	Input clock pulse width low	$0.4t_{OSC}$	—	$0.6t_{OSC}$	µs
t3	Input clock rise time (10% - 90%)	—	—	5.0	ns
t4	Input clock fall time (90% - 10%)	—	—	5.0	ns
t5	Input clock period jitter (see notes 2 and 4)	-300		300	ps
t6 (see note 1)	Input clock cycle jitter (see notes 3 and 4)	-300		300	ps

1. $t6 = 2*t_{OSC}$
2. The input clock period jitter is the displacement relative to the center period (reciprocal of the center frequency).
3. The input clock cycle jitter is the difference in period between adjacent cycles.
4. The jitter characteristics must satisfy both the t5 and t6 characteristics
5. Input Duty cycle is not critical and can be 40/60
6. The minimum System Clock frequency required for correct operation depends on the cycle length of the Intel 80 interface. See Section 8.4, “Setting SYSCLK and PCLK” on page 42 for more details.

7.1.2 PLL Clock

The PLL circuit is an analog circuit and is very sensitive to noise on the input clock waveform or the power supply. Noise on the clock or the supplied power may cause the operation of the PLL circuit to become unstable or increase the jitter.

Due to these noise constraints, it is highly recommended that the power supply traces or the power plane for the PLL be isolated from those of other power supplies. Filtering should also be used to keep the power as clean as possible. The jitter of the input clock waveform should be as small as possible.



The PLL frequency will ramp between the OFF state and the programmed frequency.
To guarantee the lowest possible clock jitter, 10ms is required for stabilization.

Note: PLL minimum frequency = 44.26MHz
(Based on Intel 80 cycle length. Refer to Section 8.4 for more information)
PLL maximum frequency = 66.95MHz

Figure 7-2: PLL Start-Up Time

Table 7-2: PLL Clock Requirements

Symbol	Parameter	Min	Max	Units
f_{PLL}	PLL output clock frequency	44.26[1]	66.95	MHz
t_{PJref}	PLL output clock period jitter	-3	3	%
t_{PDuty}	PLL output clock duty cycle	40	60	%
t_{PStal}	PLL output stable time	—	10	ms

[1] Refer to Section 8.4, “Setting SYSCLK and PCLK” on page 42.

7.2 RESET# Timing

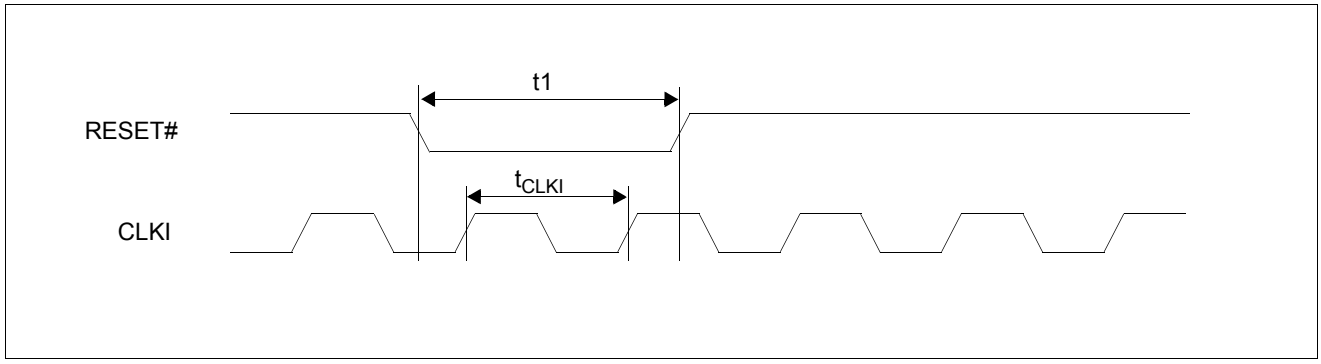


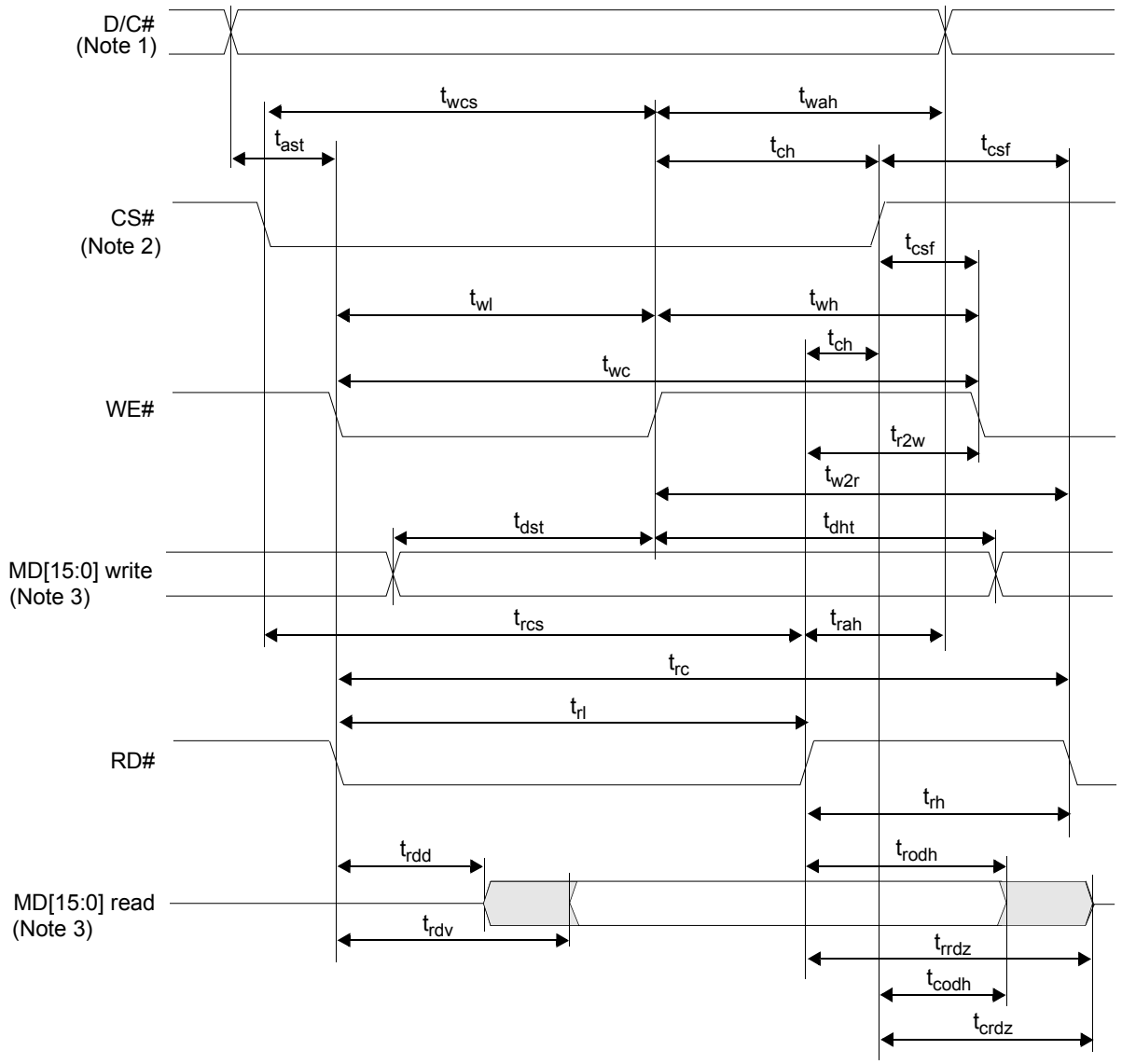
Figure 7-3 S1D13742 RESET# Timing

Table 7-3 S1D13742 RESET# Timing

Symbol	Parameter	Min	Max	Units
t1	Active Reset Pulse Width	1	—	CLKI

7.3 Host interface Timing

7.3.1 Intel 80 Interface Timing - 1.8 Volt



Note 1: The D/C# input pin is used to distinguish between Address and Data.

Note 2: The CS# pin can be kept low between write and read pulses as the register addresses will auto-increment. The register address will auto-increment in word increments for all register access except the Memory Data Port. Writes to the Memory Data Port will not increment the register address to support burst data writes to memory.

Note 3: When CNF1=0, only MD[7:0] are used.
When CNF1=1, MD[15:0] are used for accesses to the Memory Data Port. MD[7:0] are used for all other accesses.

Figure 7-4: Intel 80 Input A.C. Characteristics - 1.8 Volt

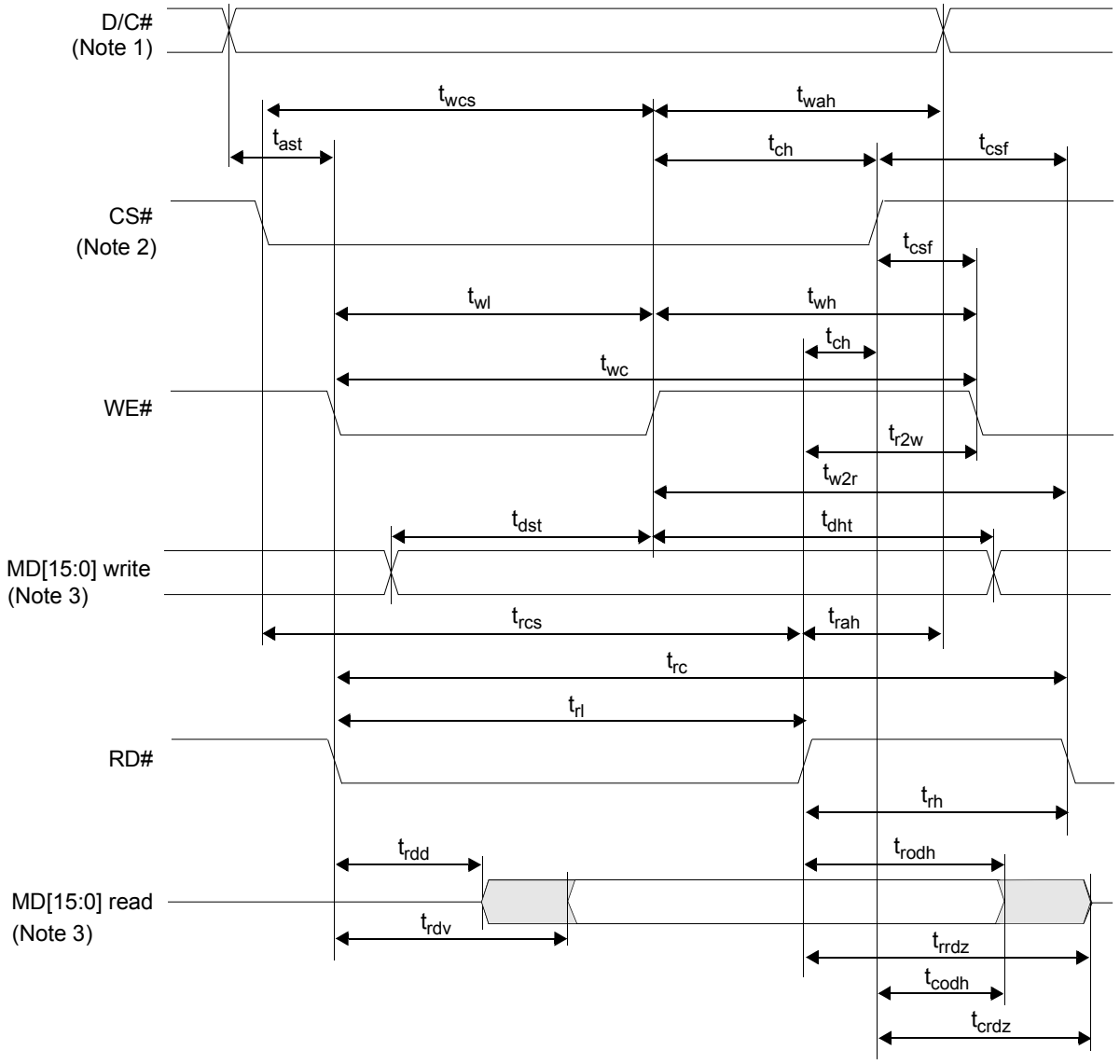
Table 7-4: Intel 80 Input A.C. Characteristics - 1.8 Volt

Signal	Symbol	Parameter	Min	Max	Unit	Description
D/C#	t_{ast}	Address setup time (read/write)	1	—	ns	
	t_{wah}	Address hold time (write)	5	—	ns	
	t_{rah}	Address hold time (read)	29	—	ns	
CS#	t_{wcs}	Chip Select setup time (write)	t_{wl}	—	ns	
	t_{rcs}	Chip Select setup time (read)	t_{rl}	—	ns	
	t_{ch}	Chip Select hold time (read/write)	0	—	ns	
	t_{csf}	Chip Select Wait time (read/write)	1	—	ns	
WE#	t_{wc}	Register Write cycle	12	—	ns	
		LUT write cycle	2SYSCLK + 1	—	ns	
		Memory write cycle	2SYSCLK + 1	—	ns	
	t_{wl}	Pulse low duration	5	—	ns	
	t_{wh}	Pulse high duration	t_{wc} - t_{wl}	—	ns	
	t_{w2r}	WR# rising edge to RD# falling edge	11	—	ns	Note 1
RD#	t_{r2w}	RD# rising edge to WR# falling edge	26	—	ns	Note 2
	t_{rc}	Read cycle	t_{rl} + t_{rh}	—	ns	
	t_{rl}	Pulse low duration	t_{rdv}	—	ns	
	t_{rh}	Pulse high duration for Registers	35	—	ns	
		Pulse high duration for Memory and LUT	1SYSCLK + 26	—	ns	
MD[15:0] (Note 3)	t_{dst}	Write data setup time	4	—	ns	
	t_{dht}	Write data hold time	5	—	ns	
	t_{rodh}	Read data hold time from RD# rising edge	11	—	ns	
	t_{rrdz}	RD# rising edge to MD Hi-Z	—	31	ns	
	t_{codh}	Read data hold time from CS# rising edge	1	—	ns	
	t_{crdz}	CS# rising edge to MD Hi-Z	—	8	ns	
	t_{rdv}	RD# falling edge to MD valid for Registers	—	16	ns	CL=30pF
		RD# falling edge to MD valid for LUT	—	4SYSCLK + 26	ns	
		RD# falling edge to MD valid for Memory	—	5SYSCLK + 19	ns	
		RD# falling edge to MD valid for Registers	—	11	ns	CL = 8pF
		RD# falling edge to MD valid for LUT	—	4SYSCLK + 21	ns	
		RD# falling edge to MD valid for Memory	—	5SYSCLK + 14	ns	
	t_{rdd}	RD# falling edge to MD driven	4	—	ns	CL=30pF
		RD# falling edge to MD driven	3	—	ns	CL = 8pF

Note

1. For a read cycle after a write cycle, MD[15:0] must be driven Hi-Z a maximum of t_{rdd} after the falling edge of RD#.
2. For a write cycle after a read cycle, MD[15:0] should not be driven by the host until t_{rrdz} after the rising edge of RD#.
3. When CNF1=0, only MD[7:0] are used. When CNF1=1, MD[7:0] are used for all accesses except for the Memory Data Port when MD[15:0] are used.

7.3.2 Intel 80 Interface Timing - 3.3 Volt



Note 1: The D/C# input pin is used to distinguish between Address and Data.

Note 2: The CS# pin can be kept low between write and read pulses as the register addresses will auto-increment. The register address will auto-increment in word increments for all register access except the Memory Data Port. Writes to the Memory Data Port will not increment the register address to support burst data writes to memory.

Note 3: When CNF1=0, only MD[7:0] are used.
When CNF1=1, MD[15:0] are used for accesses to the Memory Data Port. MD[7:0] are used for all other accesses.

Figure 7-5: Intel 80 Input A.C. Characteristics - 3.3 Volt

Table 7-5: Intel 80 Input A.C. Characteristics - 3.3 Volt

Signal	Symbol	Parameter	Min	Max	Unit	Description
D/C#	t_{ast}	Address setup time (read/write)	1	—	ns	
	t_{wah}	Address hold time (write)	5	—	ns	
	t_{rah}	Address hold time (read)	29	—	ns	
CS#	t_{wcs}	Chip Select setup time (write)	t_{wl}	—	ns	
	t_{rcs}	Chip Select setup time (read)	t_{rl}	—	ns	
	t_{ch}	Chip Select hold time (read/write)	0	—	ns	
	t_{csf}	Chip Select Wait time (read/write)	1	—	ns	
WE#	t_{wc}	Register Write cycle	12	—	ns	
		LUT write cycle	2SYSCLK + 1	—	ns	
		Memory write cycle	2SYSCLK + 1	—	ns	
	t_{wl}	Pulse low duration	5	—	ns	
	t_{wh}	Pulse high duration	t_{wc} - t_{wl}	—	ns	
	t_{w2r}	WR# rising edge to RD# falling edge	16	—	ns	Note 1
RD#	t_{r2w}	RD# rising edge to WR# falling edge	26	—	ns	Note 2
	t_{rc}	Read cycle	t_{rl} + t_{rh}	—	ns	
	t_{rl}	Pulse low duration	t_{rdv}	—	ns	
	t_{rh}	Pulse high duration for Registers	36	—	ns	
		Pulse high duration for Memory and LUT	1SYSCLK + 26	—	ns	
MD[15:0] (Note 3)	t_{dst}	Write data setup time	4	—	ns	
	t_{dht}	Write data hold time	5	—	ns	
	t_{rodh}	Read data hold time from RD# rising edge	11	—	ns	
	t_{rrdz}	RD# rising edge to MD Hi-Z	—	31	ns	
	t_{codh}	Read data hold time from CS# rising edge	1	—	ns	
	t_{crdz}	CS# rising edge to MD Hi-Z	—	8	ns	
	t_{rdv}	RD# falling edge to MD valid for Registers	—	11	ns	CL=30pF
		RD# falling edge to MD valid for LUT	—	4SYSCLK + 21	ns	
		RD# falling edge to MD valid for Memory	—	5SYSCLK + 14	ns	
		RD# falling edge to MD valid for Registers	—	9	ns	CL = 8pF
		RD# falling edge to MD valid for LUT	—	4SYSCLK + 18	ns	
		RD# falling edge to MD valid for Memory	—	5SYSCLK + 11	ns	
	t_{rdd}	RD# falling edge to MD driven	3	—	ns	CL=30pF
		RD# falling edge to MD driven	2	—	ns	CL = 8pF

Note

1. For a read cycle after a write cycle, MD[15:0] must be driven Hi-Z a maximum of t_{rdd} after the falling edge of RD#.
2. For a write cycle after a read cycle, MD[15:0] should not be driven by the host until t_{rrdz} after the rising edge of RD#.
3. When CNF1=0, only MD[7:0] are used. When CNF1=1, MD[7:0] are used for all accesses except for the Memory Data Port when MD[15:0] are used.

7.3.3 Definition of Transition Time to Hi-Z State

Due to the difficulty of Hi-Z impedance measurement for high speed signals, transition time from High/Low to Hi-Z specified as follows.

- High to Hi-Z delay time: t_{pHZ}, delay time when a gate voltage of final stage of the Pch-MOSFET turns to 0.8 x IOVDD (Pch-MOSFET is off). Total delay time to Hi-Z is calculated as follows:
  Internal logic delay + t_{pHZ} (from High to Hi-Z)
- Low to Hi-Z delay time: t_{pLZ}, delay time when a gate voltage of final stage of the Nch-MOSFET turns to 0.2 x IOVDD (Nch-MOSFET is off). Total delay time to Hi-Z is calculated as follows:
  Internal logic delay + t_{pHZ} (from High to Hi-Z)

The functional model of a final stage of the Tri state Output Cell is shown in Figure 7-6: "Definition of transition time to Hi-Z state".

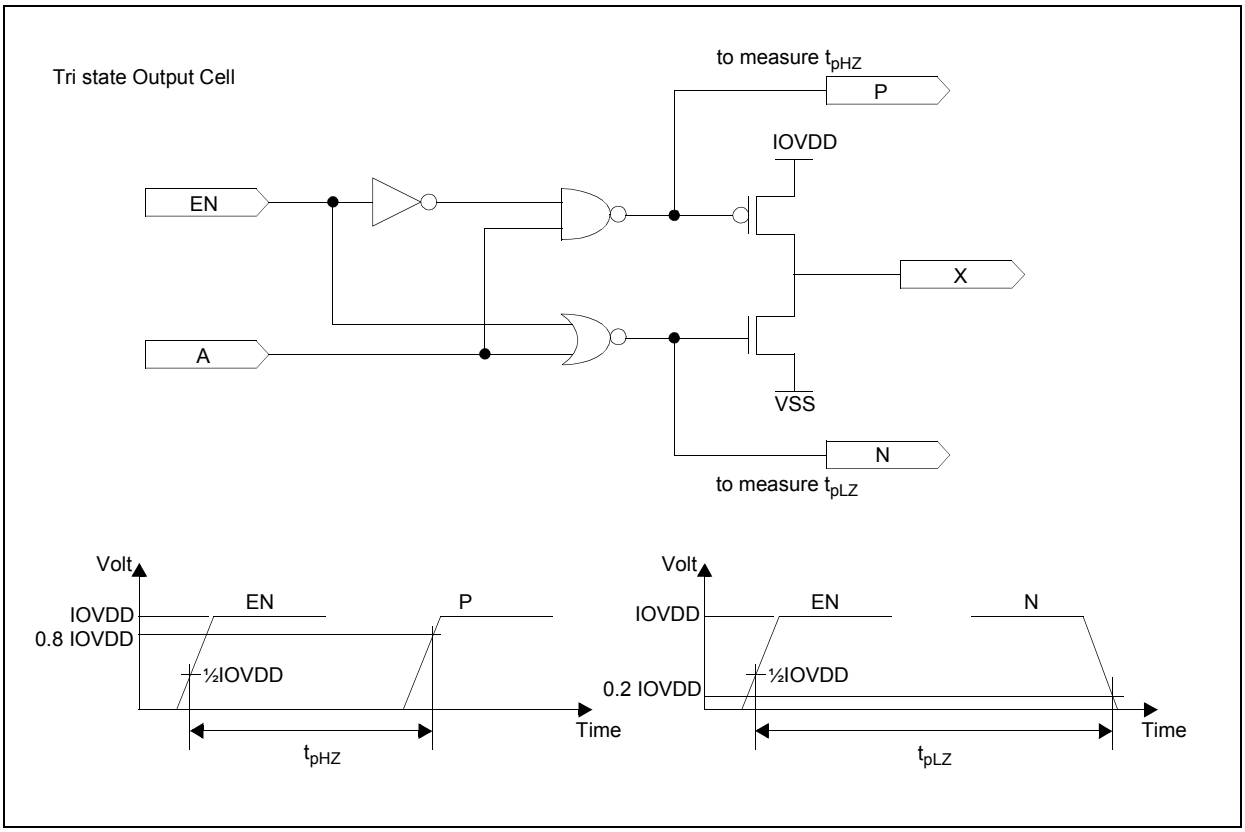


Figure 7-6: Definition of Transition Time to Hi-Z State

7.4 Display Interface

The timing parameters required to drive a flat panel display are shown below. Timing details for each supported panel type are provided in the remainder of this section.

Note
All timing measurements are taken to/from the ½PIOVDD level in the following Display Interface timing diagrams.

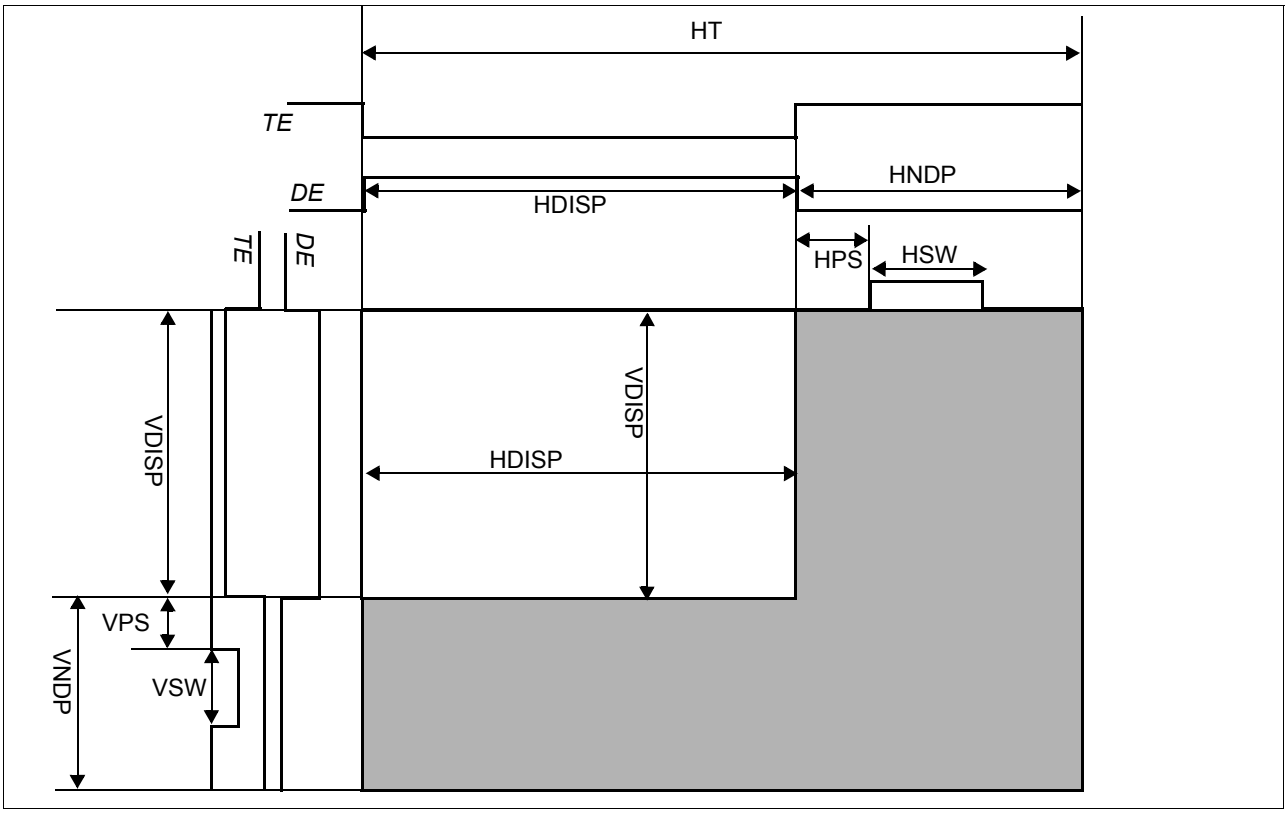


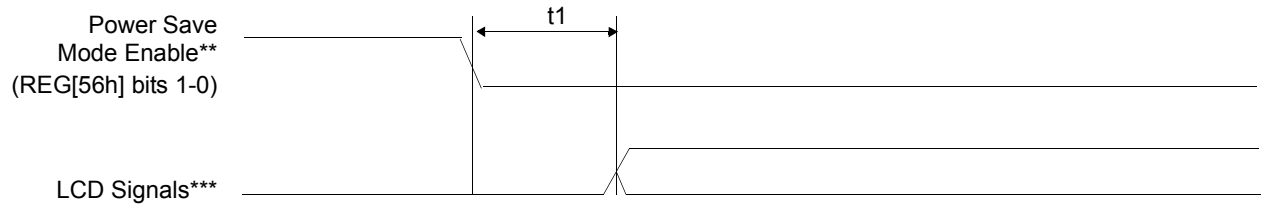
Figure 7-7: Panel Timing Parameters

Table 7-6: Panel Timing Parameter Definition and Register Summary

Symbol	Description	Derived From	Units
HDISP	Horizontal Display Width	(REG[16h] bits 6-0) x 8	Ts
HNDP	Horizontal Non-Display Period	(REG[18h] bits 6-0)	
HPS	HS Pulse Start Position	REG[22h] bits 6-0	
HSW	HS Pulse Width	(REG[20h] bits 6-0)	
VDISP	Vertical Display Height	(REG[1Ch] bits 1-0, REG[1Ah] bits 7-0)	Lines (HT)
VNDP	Vertical Non-Display Period	REG[1Eh] bits 7-0	
VPS	VS Pulse Start Position	REG[26h] bits 7-0	
VSW	VS Pulse Width	REG[24h] bits 6-0	

Note
TS = 1/PCLK

7.4.1 TFT Power-On Sequence



**The LCD power-on sequence is activated by programming the Power Save Register (REG[56h]) bit 1 or bit 0 to 0.
***LCD Signals include: VD[35:0], PCLK, HS, VS, and DE.

Figure 7-8: TFT Power-On Sequence Timing

Table 7-7: TFT Power-On Sequence Timing

Symbol	Parameter	Min	Max	Units
t1	Power Save Mode disabled to LCD signals active	0	20	ns

7.4.2 TFT Power-Off Sequence

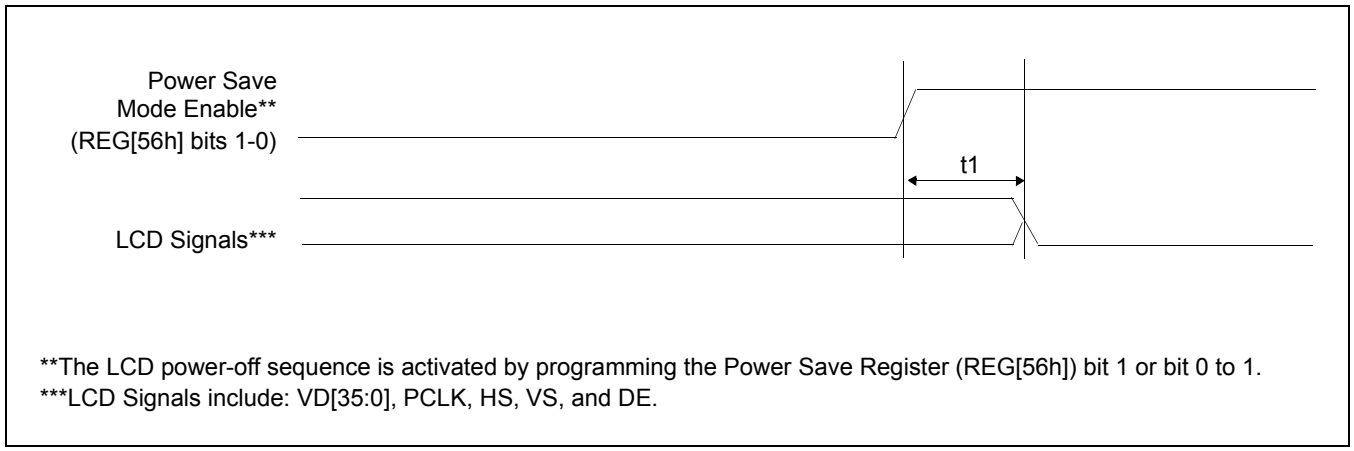


Figure 7-9: TFT Power-Off Sequence Timing

Table 7-8: TFT Power-Off Sequence Timing

Symbol	Parameter	Min	Max	Units
t1	Power Save Mode enabled to LCD signals low	0	20	ns

7.4.3 18/36-Bit TFT Panel Timing

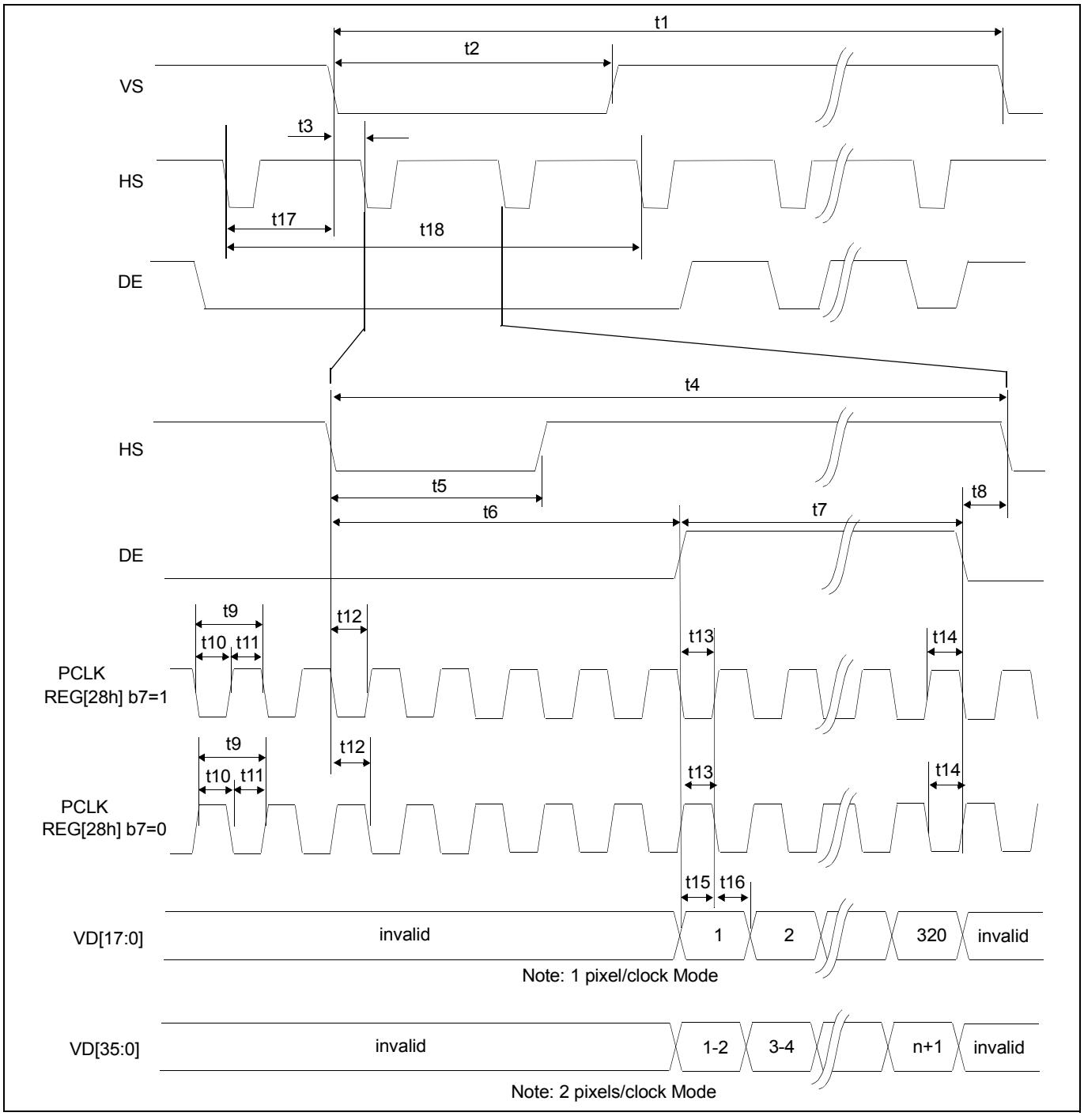


Figure 7-10: 18/36-Bit TFT A.C. Timing

Note

HS, VS, PCLK all have Polarity Select bits via registers

Table 7-9: 18/36-Bit TFT A.C. Timing

Symbol	Parameter	Min	Typ	Max	Units
t1	VS cycle time	—	VDISP + VNDP	—	Lines
t2	VS pulse width low	—	VSW	—	Lines
t3	VS falling edge to HS falling edge phase difference	—	HPS	—	Ts
t4	HS cycle time	—	HDISP + HNDP	—	Ts
t5	HS pulse width low	—	HSW	—	Ts
t6	HS Falling edge to DE active	—	HNDP-HPS	—	Ts
t7	DE pulse width	—	HDISP	—	Ts
t8	DE falling edge to HS falling edge	—	HPS	—	Ts
t9	PCLK period	1	—	—	Ts
t10	PCLK pulse width low	0.5	—	—	Ts
t11	PCLK pulse width high	0.5	—	—	Ts
t12	HS setup to PCLK active edge	0.5	—	—	Ts
t13	DE to PCLK rising edge setup time	0.5	—	—	Ts
t14	DE hold from PCLK active edge	0.5	—	—	Ts
t15	Data setup to PCLK active edge	0.5	—	—	Ts
t16	Data hold from PCLK active edge	0.5	—	—	Ts
t17	DE Stop setup to VS start	—	VPS	—	Ts
t18	Vertical Non-Display Period	—	VNDP	—	Ts

1. Ts = pixel clock period

Note

In 36-bit mode, the data is always guaranteed to be launched on the correct edge of PCLK. In this mode, the frequency of PCLK is ½ the programmed internal value. If it is desired that HS and VS are always launched on the same edge of PCLK as the data, then HNDP, HSW, and HSS should be programmed with even values.

8 Clocks

8.1 Clock Descriptions

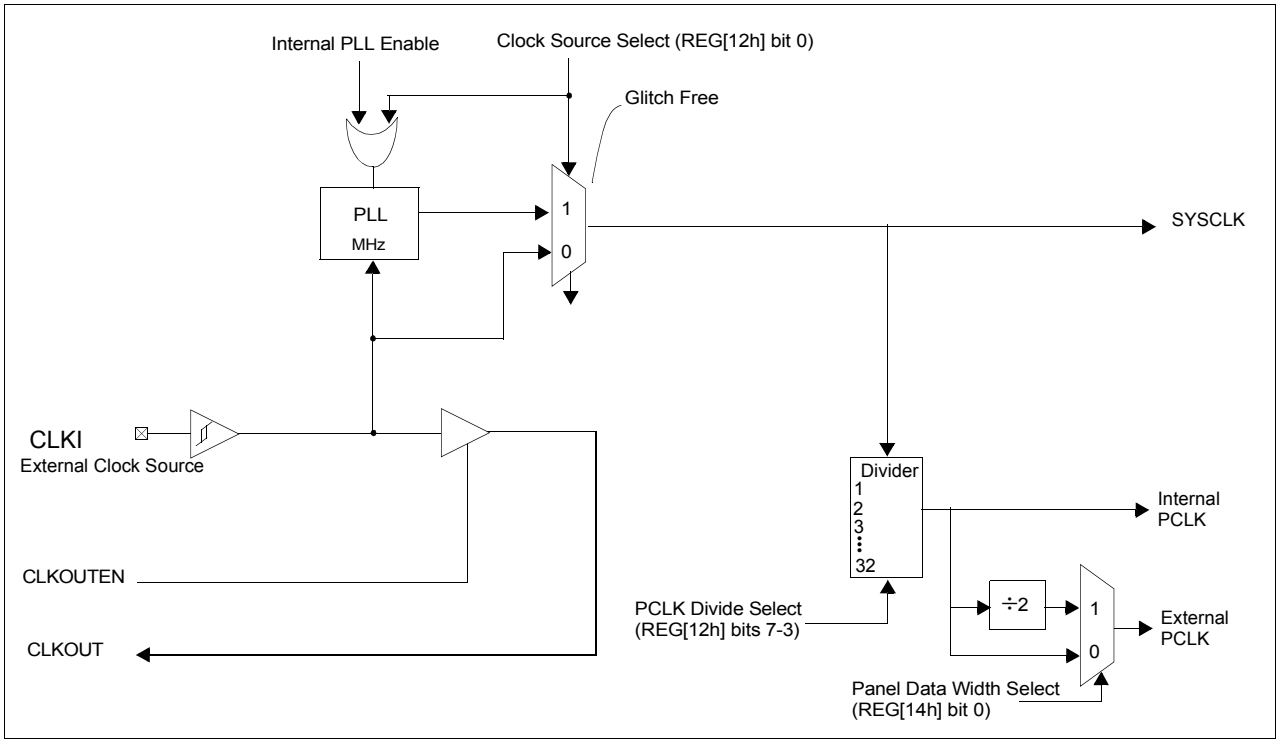


Figure 8-1: S1D13742 Clock Block Diagram

8.2 PLL Block Diagram

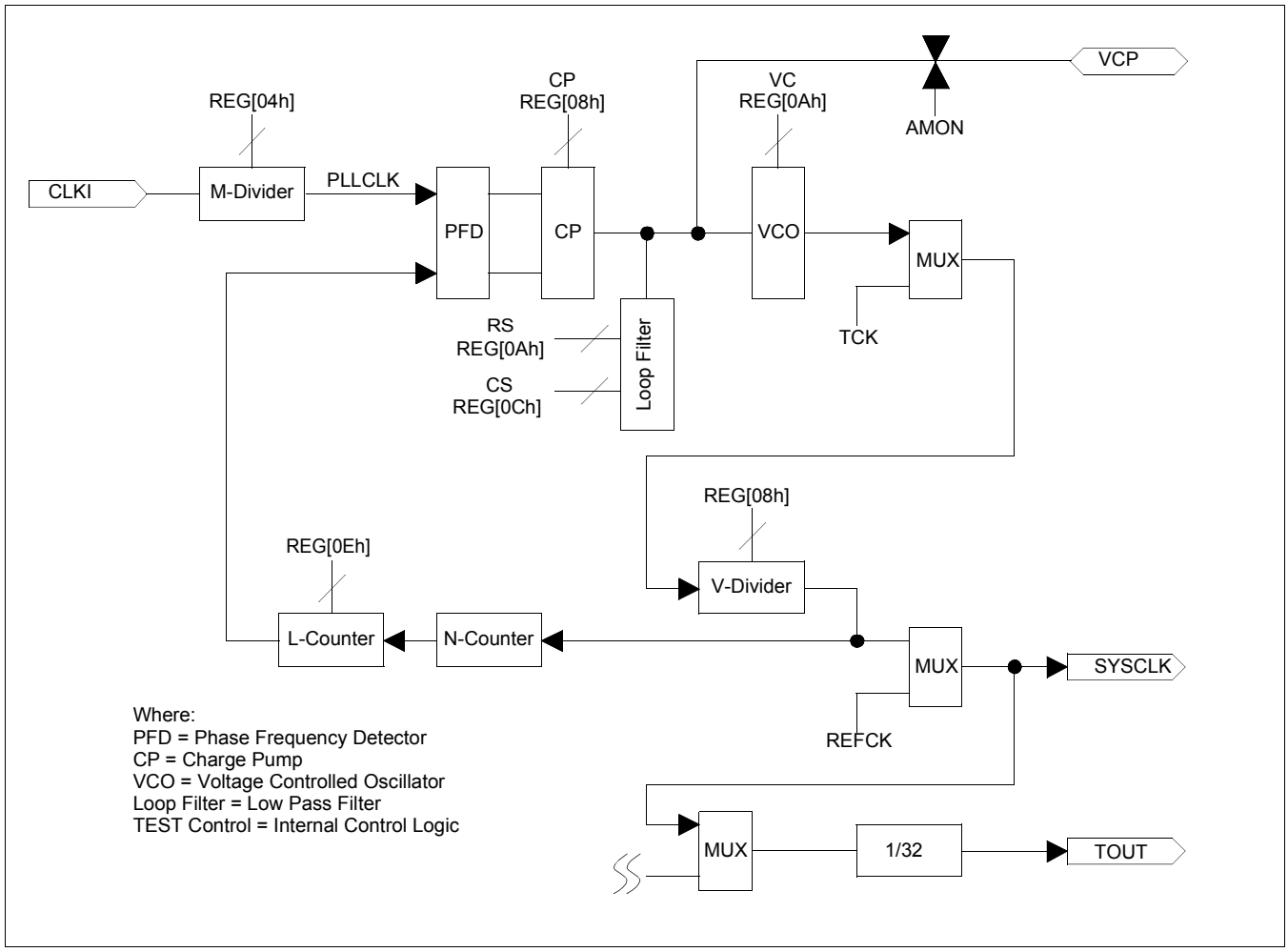


Figure 8-2: PLL Block Diagram

8.3 Clocks versus Functions

This table lists the internal clocks required for the following S1D13742 functions.

Internal Clock Requirements

Function	Internal SYSCLK	Internal PCLK
Register Read/Write	No	No
Memory Read/Write	Yes	No
Look-Up Table Register Read/Write	Yes	No
Power Save	No	No
LCD Output	Yes	Yes

Note

Register access does not require an internal clock as the S1D13742 creates a clock from the bus cycle alone.

8.4 Setting SYSCLK and PCLK

The period of the system clock, T_{SYSCLK}, must be set such that it falls within the following range:

For PLL: $14.94ns < T_{SYSCLK} < (T_{BBC} - 0.914) \times 0.485$ ns
For CLKI: $14.50ns < T_{SYSCLK} < (T_{BBC} - 0.914) \times 0.5ns$

where T_{BBC} is the minimum back-to-back cycle time of the Intel 80 Interface.

For example, if the minimum back-to-back cycle time of the Intel 80 Interface is 5 x 9.5 = 47.5ns, then:

For PLL: $14.94ns < T_{SYSCLK} < 22.594ns$
For CLKI: $14.50ns < T_{SYSCLK} < 23.293ns$

Therefore,

For PLL: $44.26MHz < f_{SYSCLK} < 66.95MHz$
For CLKI: $42.94MHz < f_{SYSCLK} < 68.96MHz$

Care should be taken when setting T_{SYSCLK} so that the desired PCLK frequency, f_{PCLK}, can be achieved. PCLK is an integer divided version of SYSCLK. The following graph shows the suggested setting for SYSCLK for a given value of PCLK for T_{BBC} = 47.5ns.

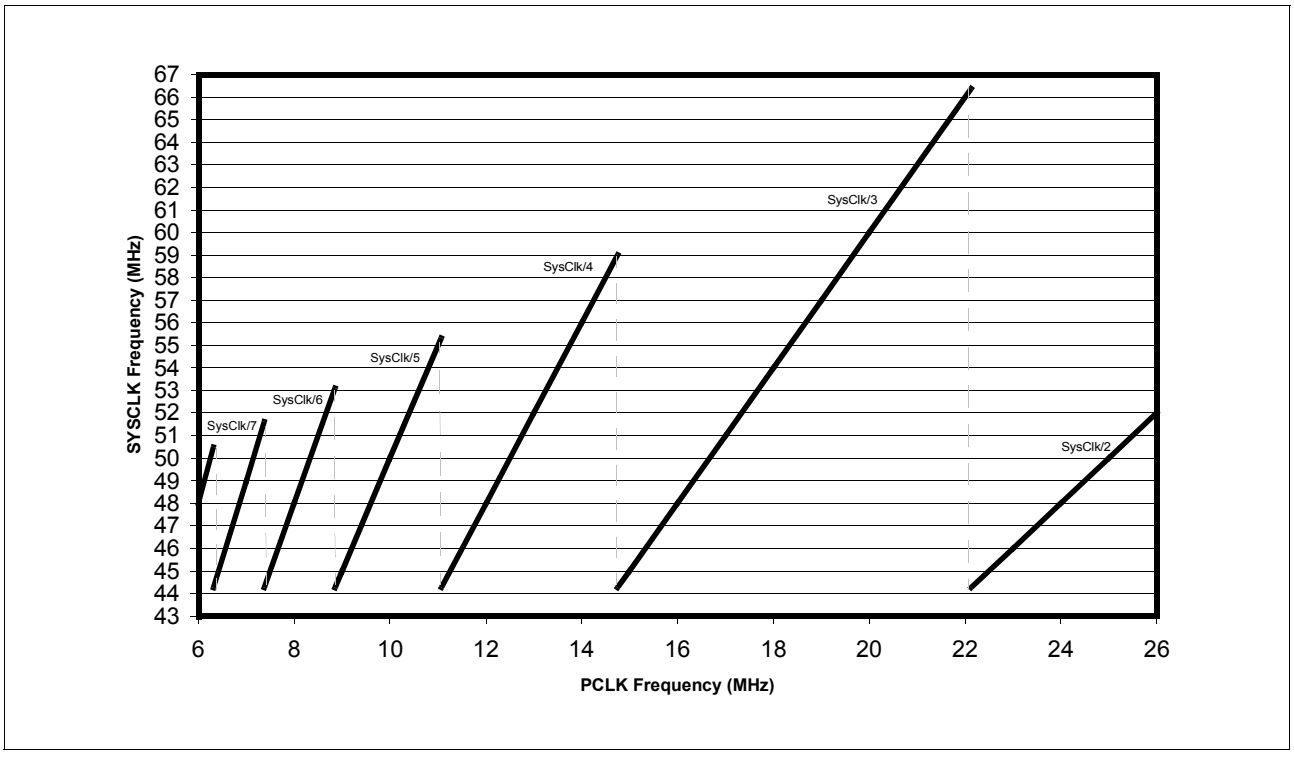


Figure 8-3: Setting of SYSCLK For a Desired PCLK

9 Registers

This section discusses how and where to access the S1D13742 registers. It also provides detailed information about the layout and usage of each register.

Burst data writes to the register space is supported. This applies to all register write access except the Memory Data Port (REG[48h - 49h]) and the Gamma Correction Table Data Register [REG[54h]). All writes to these two registers will auto-increment the internal memory address only.

9.1 Register Mapping

All registers and memory are accessed via the Intel 80 interface. All access is 8-bit only except for the Memory Data Port (REG[48h - 49h]) which is accessed as 16-bit (if CNF1=1) or 8-bit (if CNF1=0).

9.2 Register Set

The S1D13742 registers are listed in the following table.

Table 9-1: S1D13742 Register Set

Register	Pg	Register	Pg
Read-Only Configuration Registers			
REG[00h] Revision Code Register	45	REG[02h] Configuration Readback Register	45
Clock Configuration Registers			
REG[04h] PLL M-Divider Register	46	REG[06h] PLL Setting Register 0	47
REG[08h] PLL Setting Register 1	47	REG[0Ah] PLL Setting Register 2	47
REG[0Ch] PLL Setting Register 3	48	REG[0Eh] PLL Setting Register 4	48
REG[10h]	48	REG[12h] Clock Source Select Register	49
Panel Configuration Registers			
REG[14h] Panel Type Register	51	REG[16h] Horizontal Display Width Register (HDISP)	51
REG[18h] Horizontal Non-Display Period Register (HNDP)	51	REG[1Ah] Vertical Display Height Register 0 (VDISP)	52
REG[1Ch] Vertical Display Height Register 1 (VDISP)	52	REG[1Eh] Vertical Non-Display Period Register (VNDP)	52
REG[20h] HS Pulse Width Register (HSW)	52	REG[22h] HS Pulse Start Position Register 0 (HPS)	53
REG[24h] VS Pulse Width Register (VSW)	53	REG[26h] VS Pulse Start Position Register 0 (VPS)	53
REG[28h] PCLK Polarity Register	53		
Input Mode Register			
REG[2Ah] Input Mode Register	54	REG[2Ch] Input YUV/RGB Translate Mode Register 0	56
REG[2Eh] YUV/RGB Translate Mode Register 1	56	REG[30h] U Data Fix Register	58
REG[32h] V Data Fix Register	58		
Display Mode Registers			
REG[34h] Display Mode Register	59	REG[36h] Special Effects Register	60
Window Settings			
REG[38h] Window X Start Position Register 0	63	REG[3Ah] Window X Start Position Register 1	63
REG[3Ch] Window Y Start Position Register 0	63	REG[3Eh] Window Y Start Position Register 1	63
REG[40h] Window X End Position Register 0	64	REG[42h] Window X End Position Register 1	64
REG[44h] Window Y End Position Register 0	64	REG[46h] Window Y End Position Register 1	64
Memory Access			
REG[48h] Memory Data Port Register 0	65	REG[49h] Memory Data Port Register 1	65
REG[4Ah] Memory Read Address Register 0	66	REG[4Ch] Memory Read Address Register 1	66
REG[4Eh] Memory Read Address Register 2	66		
Gamma Correction Registers			
REG[50h] Gamma Correction Enable Register	67	REG[52h] Gamma Correction Table Index Register	68
REG[54h] Gamma Correction Table Data Register	68		
Miscellaneous Registers			
REG[56h] Power Save Register	69	REG[58h] Non-Display Period Control / Status Register	69
General Purpose IO Pins Registers			
REG[5Ah] General Purpose IO Pins Configuration Register 0	71	REG[5Ch] General Purpose IO Pins Status/Control Register 0	71
REG[5Eh] GPIO Positive Edge Interrupt Trigger Register	71	REG[60h] GPIO Negative Edge Interrupt Trigger Register	72
REG[62h] GPIO Interrupt Status Register	72	REG[64h] GPIO Pull Down Control Register 0	72

9.3 Register Descriptions

All reserved bits must be set to the default value. Writing a non-default value to a reserved bit may produce undefined results. Bits marked as n/a have no hardware effect. Unless specified otherwise, all register bits are set to 0 during power-on reset.

9.3.1 Read-Only Configuration Registers

REG[00h] Revision Code Register
Default = 80h for S1D13742B00 or 81h for S1D13742B01 — Read Only

Product Code bits 5-0						Revision Code bits 1-0	
7	6	5	4	3	2	1	0

bits 7-2 Product Code bits [5:0]
These are read-only bits that indicates the product code. The product code is 100000b.

bits 1-0 Revision Code bits [1:0]
These are read-only bits that indicates the revision code. The revision code for the S1D13742B00 is 00b, and for the S1D13742B01 is 01b.

REG[02h] Configuration Readback Register
Default = xxh — Read Only

n/a					CNF2 Status	CNF1 Status	CNF0 Status
7	6	5	4	3	2	1	0

bits 2-0 CNF[2:0] Status
These read-only status bits return the status of the configuration pins CNF[2:0].

9.3.2 Clock Configuration Registers

REG[04h] PLL M-Divider Register Default = 00h							Read/Write
PLL Lock Bit (RO)	n/a	M-Divider bits 5-0					
7	6	5	4	3	2	1	0

bit 7 PLL Lock Bit (read only)
When this bit = 0, the PLL output is not stable. In this state R/W access to the display buffer is prohibited.
When this bit = 1, the PLL output is stable.

bits 5-0 M-Divider bits [5:0]
These bits determine the divide ratio between CLKI and the actual input clock to the PLL

Note
The internal input clock to the PLL (PLLCLK) must be between 1 MHz and 2 MHz. Depending on CLKI, these bits will have to be set accordingly.

Note
Values higher then 20h are not allowed.

Table 9-2: PLL M-Divide Selection

REG[04h] bits 5-0	M-Divide Ratio
0h	1:1
01h	2:1
02h	3:1
03h	4:1
• • •	• • •
20h	33:1
21h to 3Fh	Reserved

REG[06h] PLL Setting Register 0

Default = 00h Read/Write

PLL Setting Register 0 bits 7-0							
7	6	5	4	3	2	1	0

This register must be programmed with the value F8h.

REG[08h] PLL Setting Register 1

Default = 00h Read/Write

PLL Setting Register 1 bits 7-0							
7	6	5	4	3	2	1	0

This register must be programmed with the value 80h.

REG[0Ah] PLL Setting Register 2

Default = 00h Read/Write

PLL Setting Register 2 bits 7-0							
7	6	5	4	3	2	1	0

This register must be programmed with the value 28h.

REG[0Ch] PLL Setting Register 3
Default = 00h — Read/Write

PLL Setting Register 3 bits 7-0							
7	6	5	4	3	2	1	0

This register must be programmed with the value 00h.

REG[0Eh] PLL Setting Register 4
Default = 00h — Read/Write

n/a	L-Counter bits 6-0						
7	6	5	4	3	2	1	0

bits 6-0 L-Counter bits [6:0]

These bits are used to configure the PLL Output (in MHz) and must be set according to the following formula.

PLL Output = (L-Counter +1) x PLLCLK
= LL x PLLCLK

Where:

PLL Output is the desired PLL output frequency (in MHz).
L-Counter is the value of this register (in decimal).
PLLCLK is the internal input clock to the PLL (in MHz).

Please refer to Section 8.4, “Setting SYSCLK and PCLK” on page 42 for restrictions on PLL Output frequencies.

Table 9-3 PLL Setting Example

Target Frequency (MHz)	LL	CLKI Input Clock (MHz)	M-Divider REG[04] bits 5-0	M-Divide Ratio	PLLCLK (MHz)	POUT (MHz)
53	53	12	0Bh	12:1	1.0	53
60	60	12	0Bh	12:1	1.0	60
•	•	•	•	•	•	•
53	53	19.2	12h	19:1	1.0105	53.53
60	60	19.2	12h	19:1	1.0105	60.63

REG[10h]
Default = 00h — Read/Write

n/a							
7	6	5	4	3	2	1	0

Writes to this register have no effect on hardware. During Auto Increment, a dummy write needs to be performed to this register.

REG[12h] Clock Source Select Register
Default = 00h — Read/Write

PCLK Divide Select bits 4-0					n/a		SYSCLK Source Select
7	6	5	4	3	2	1	0

bits 7-3 PCLK Divide Select bits [5:0]
These bits specify the divide ratio for the panel clock (PCLK).
The clock source for PCLK is SYSCLK.

All resulting clock frequencies will maintain a 50/50 duty cycle regardless of divide ratio.

Table 9-4 PCLK Divide Ratio Selection

REG[0012h] bits 7-3	PCLK Divide Ratio
00h	Reserved
01h	2:1
02h	3:1
03h	4:1
04h	5:1
05h	6:1
06h	7:1
07h	8:1
08h	9:1
09h	10:1
0Ah	11:1
0Bh	12:1
0Ch	13:1
0Dh	14:1
0Eh	15:1
0Fh	16:1
10h	17:1
11h	18:1
• • •	• • •
1Fh	32:1

bit 0 SYSCLK Source Select

This bit selects the system clock (SYSCLK) source for the controller.
When this bit = 0, the SYSCLK source is the external CLKI input.
When this bit = 1, the SYSCLK source is the internal PLL.

If the PLL is selected as the SYSCLK source (bit 0 = 1), the PLL must be configured using REG[06h], REG[08h], REG[0Ah], REG[0Ch], REG[0Eh] and REG[10h] before setting this bit.

Note

To use PLL as system clock source (SYSCLK), Sleep Mode needs to be first enabled, REG[56h] bit 1 = 1. Once in Sleep Mode, REG[04h] and REG[0Eh] can be changed to set the desired PLL frequency. Once REG[04h] and REG[0Eh] have been set, REG[12h] bit 0 can be set to 1b to select PLL as the system clock source. The PLL output will only be active after exiting the Sleep Mode (REG[56h] bit 1 = 0). The PLL output will become stable after 10msec. The display memory or the Gamma Correction Table must not be accessed before this time. REG[04h] bit 7, the PLL Lock Bit, can be used to determine if the PLL output is stable.

9.3.3 Panel Configuration Registers

REG[14h] Panel Type Register
Default = 00h — Read/Write

7	6	5	4	3	2	1	0
VD Data Swap	n/a						Panel Data Width

bit 7 VD Data Swap
When this bit = 0, data lines are normal (i.e.: output pin VD35 = VD35, etc.)
When this bit = 1, data lines are swapped (i.e.: output pin VD35 = VD0, etc.)

Note
The Data swap will always go from the msb to the lsb on the active output pins. See "LCD Interface Data Pins" on page 20.

bit 0 Panel Data Width
When this bit = 0, the LCD interface is configured as 18-bit.
When this bit = 1, the LCD interface is configured as 36-bit.

REG[16h] Horizontal Display Width Register (HDISP)
Default = 01h — Read/Write

7	6	5	4	3	2	1	0
n/a	Horizontal Display Period bits 6-0						

bits 6-0 Horizontal Display Width bits [6:0]
These bits specify the LCD panel Horizontal Display Width (HDISP), in 8 pixel resolution.

Horizontal Display Width in number of pixels = ((REG[16h] bits 6-0) × 8

Note
Minimum value of 8 pixels (register programmed to 1).

REG[18h] Horizontal Non-Display Period Register (HNDP)
Default = 00h — Read/Write

7	6	5	4	3	2	1	0
n/a	Horizontal Non-Display Period bits 6-0						

bits 6-0 Horizontal Non-Display Period bits [6:0]
These bits specify the horizontal non-display period in pixels. For 36-bit wide panels, there are 2 pixels per external PCLK.

HNDP is calculated using the following formula.
HNDP = (REG[18h] bits 6-0)

Note
The minimum Horizontal Non-Display Period is 3 Pixels (REG[18h] bits 6-0 = 03h).
HS Start + HS Width <= HNDP

REG[1Ah] Vertical Display Height Register 0 (VDISP)
Default = 01h — Read/Write

7	6	5	4	3	2	1	0
Vertical Display Height bits 7-0							

REG[1Ch] Vertical Display Height Register 1 (VDISP)
Default = 00h — Read/Write

7	6	5	4	3	2	1	0
n/a						Vertical Display Height bits 9-8	

REG[1Ch] bits 1-0
REG[1Ah] bits 7-0 Vertical Display Height bits [9:0]
These bits specify the LCD panel Vertical Display Height, in 1 line resolution.
Vertical Display Height in number of lines = (REG[1Ch] bits 1-0, REG[1Ah] bits 7-0)

Note
Minimum value = 1 line

REG[1Eh] Vertical Non-Display Period Register (VNDP)
Default = 01h — Read/Write

7	6	5	4	3	2	1	0
Vertical Non-Display Period bits 7-0							

bits 7-0 Vertical Non-Display Period bits [7:0]
These bits specify the Vertical Non-Display Period for panels in 1 line resolution.

Note
Minimum value = 2 lines

REG[20h] HS Pulse Width Register (HSW)
Default = 00h — Read/Write

7	6	5	4	3	2	1	0
HS Pulse Polarity	HS Pulse Width bits 6-0						

bit 7 HS Pulse Polarity
This bit selects the polarity of the horizontal sync signal. This bit is set according to the horizontal sync signal of the panel.
When this bit = 0, the horizontal sync signal is active low.
When this bit = 1, the horizontal sync signal is active high.

bits 6-0 HS Pulse Width bits [6:0]
These bits specify the width of the panel horizontal sync signal, in 1 pixel resolution. The horizontal sync signal is typically HS, depending on the panel type. The minimum value for these bits is 1.

HS Pulse Width in number of pixels = (REG[20h] bits 6-0)

For 36-bit wide panels, there are 2 pixels per external PCLK.

REG[22h] HS Pulse Start Position Register 0 (HPS)
Default = 00h — Read/Write

7	6	5	4	3	2	1	0
n/a	HS Pulse Start Position bits 6-0						

bits 6-0 HS Pulse Start Position bits [6:0]
These bits specify the start position of the horizontal sync signal with respect to the start of Horizontal Non-Display period, in 1 pixel resolution. For 36-bit wide panels, there are 2 pixels per external PCLK.

HPS = (REG[22h] bits 6-0)

REG[24h] VS Pulse Width Register (VSW)
Default = 00h — Read/Write

7	6	5	4	3	2	1	0
VS Pulse Polarity	n/a	VS Pulse Width bits 5-0					

bit 7 VS Pulse Polarity
This bit selects the polarity of the vertical sync signal. This bit is set according to the vertical sync signal of the panel.
When this bit = 0, the vertical sync signal is active low.
When this bit = 1, the vertical sync signal is active high.

bits 5-0 VS Pulse Width bits [5:0]
These bits specify the width of the panel vertical sync signal, in 1 line resolution. The vertical sync signal is typically VS, depending on the panel type.

VS Pulse Width in number of lines = REG[24h] bits 5-0

REG[26h] VS Pulse Start Position Register 0 (VPS)
Default = 00h — Read/Write

7	6	5	4	3	2	1	0
VS Pulse Start Position bits 7-0							

bits 7-0 VS Pulse Start Position bits [7:0]
These bits specify the start position of the vertical sync signal with respect to the start of Vertical Non-Display period, in 1 line resolution.

VPS is calculated using the following formula:
VPS = (REG[26h] bits 7-0)

REG[28h] PCLK Polarity Register
Default = 00h — Read/Write

7	6	5	4	3	2	1	0
PCLK Polarity	n/a						

bit 7 PCLK Polarity
When this bit = 0, the PCLK outputs data transitions on the rising edge
When this bit = 1, the PCLK outputs data transitions on the falling edge

9.3.4 Input Mode Register

REG[2Ah] Input Mode Register Default = 01h							Read/Write
Memory Data Format	n/a			Input Data Format			
7	6	5	4	3	2	1	0

bit 7 Memory Data Format
This bit determines how the data is stored in memory
When this bit = 0, the data stored in memory is 16 bpp. In this case 18 bpp input data will be truncated to 16 bpp
When this bit = 1, the data stored in memory is 18 bpp. In this case 16 bpp input data (as determined by bits 3-0) will be expanded to 18 bpp.

Note

In 18-bpp mode, memory above $A0000h is reserved for 2 bits of each 18 bit pixel. Therefore the maximum display resolution supported can be calculated as follows:

$X \times Y \times 2 \leq 640KB$

In 16-bpp mode the entire 768K Byte display buffer is available and therefore the maximum display resolution is $X \times Y \times 2 \leq 768KB$

bits 4-0 Input Data Format bits [4:0]

Table 9-5: Input Data Type Selection

REG[2Ah] bits 3-0	Input Data Type
0000	Reserved
0001	RGB 5:6:5
0010	RGB 6:6:6 Mode 1
0011	RGB 8:8:8 Mode 1 (LSBs will be truncated to 16 bpp or 18 bpp)
0100	Reserved
0101	Reserved
0110	RGB 6:6:6 Mode 2
0111	RGB 8:8:8 Mode 2 (LSBs will be truncated to 16 bpp or 18 bpp)
1000	YUV 4:2:2
1001	YUV 4:2:0
1010 • • • 1111	Reserved

Note

For YUV 4:2:2 and YUV 4:2:0 settings, the image width must be a multiple of 2 and 4 respectively. For YUV 4:2:0 the height must be a multiple of 2.
For RGB 6:6:6 and RGB 8:8:8 Mode 1, if the image width is odd, the red pixel data in the last word in each line will be ignored. The red pixel data will need to be re-written on the following transfer along with the green data. See Figure 12-2: “18 bpp Mode 1(R 6-bit, G 6-bit, B 6-bit), 262,144 colors,” on page 78 or Figure 12-4: “24 bpp Mode 1(R 8-bit, G 8-bit, B 8-bit), 16,777,216 colors,” on page 80.

Note

For further information on Input Data Format and Memory Data Format, see Section 11, “Intel 80, 8-bit Interface Color Formats” on page 74, Section 12, “Intel 80, 16-bit Interface Color Formats” on page 77 and Section 13, “YUV Timing” on page 82.

REG[2Ch] Input YUV/RGB Translate Mode Register 0
Default = 00h — Read/Write

Reserved	YUV/RGB Converter Reset	UV Fix bits 1-0		n/a			
7	6	5	4	3	2	1	0

bit 7 Reserved
The default value for this bit is 0.

bit 6 YUV/RGB Converter Reset
This bit performs a software reset of the YUV to RGB Converter (YRC). To perform a reset, write a 1 to enter reset, and then write a 0 to return from the reset state.
For Reads:
When this bit = 0, the YRC is not in a reset state.
When this bit = 1, the YRC is in a reset state.
For Writes:
Writing a 0 to this bit returns the YRC from the reset state.
Writing a 1 to this bit initiates a software reset of the YRC.

bits 5-4 UV Fix Select bits [1:0]
These bits control the UV input to the YUV/RGB Converter (YRC).

Table 9-6: UV Fix Selection

REG[2Ch] bits 5-4	UV Input to the YUV/RGB Converter
00	Original U data, original V data
01	U data = REG[30h] bits 7-0, original V data
10	Original U data, V data = REG[032h] bits 7-0
11	U data = REG[30h] bits 7-0, V data = REG[32h] bits 7-0

REG[2Eh] YUV/RGB Translate Mode Register 1
Default = 05h — Read/Write

Reserved		YUV Input Data Type Select bits 1-0		Reserved	YUV/RGB Transfer Mode bits 2-0		
7	6	5	4	3	2	1	0

bits 7-6 Reserved
The default value for these bits is 0.

bits 5-4 YUV Input Data Type Select bits [1:0]
These bits specify the data type of the YUV input to the YUV to RGB Converter (YRC).

Table 9-7: YUV Data Type Selection

REG[2Eh] bits 5-4	YRC Input Data Range
00	$0 \le Y \le 255$ $-128 \le U \le 127$ $-128 \le V \le 127$
01	$16 \le Y \le 235$ $-113 \le U \le 112$ $-113 \le V \le 112$
10	$0 \le Y \le 255$ $0 \le U \le 255$ $0 \le V \le 255$
11	$16 \le Y \le 235$ $16 \le U \le 240$ $16 \le V \le 240$

bit 3 Reserved
The default value for this bit is 0.

bits 2-0 YUV/RGB Transfer Mode bits [2:0]
These bits specify the YUV/RGB Transfer mode. Recommended settings are provided for various specifications.

Table 9-8: YUV/RGB Transfer Mode Selection

REG[2Eh] bits 2-0	YUV/RGB Specification
000	Reserved
001	Recommended for ITU-R BT.709
010	Reserved
011	Reserved
100	Recommended for ITU-R BT.470-6 System M
101 (Default)	Recommended for all other systems in ITU-R BT.470-6 (Recommended for ITU-R BT.601-5)
110	SMPTE 170M
111	SMPTE 240M(1987)

REG[30h] U Data Fix Register

Default = 00h — Read/Write

U Data Fix bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 U Data Fix bits [7:0]

These bits only have an effect when the UV Fix Select bits are set to 01 or 11 (REG[2Ch] bits 5-4 = 01 or 11). The U Data Input of the YUV/RGB Converter data is fixed to the value of these bits.

REG[32h] V Data Fix Register

Default = 00h — Read/Write

V Data Fix bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 V Data Fix bits [7:0]

These bits only have an effect when the UV Fix Select bits are set to 10 or 11 (REG[2Ch] bits 5-4 = 10 or 11). The V Data Input of YUV/RGB Converter data is fixed to the value of these bits.

9.3.5 Display Mode Registers

REG[34h] Display Mode Register
Default = 00h — Read/Write

7	6	5	4	3	2	1	0
Display Blank	n/a					SwivelView Mode Select bits 1-0	

bit 7 Display Blank
When this bit = 0, the LCD display pipeline is enabled.
When this bit = 1, the LCD display pipeline is disabled and all LCD data outputs are forced to zero (i.e., the screen is blanked).

bits 1-0 Window SwivelView Mode Select bits [1:0]
These bits select different SwivelView™ orientations:

Table 9-9: SwivelView™ Mode Select Options

REG[34h] bits 1-0	SwivelView Orientation
00	0° (Normal)
01	90°
10	180°
11	270°

Note
All windows written to the active display can have independent rotation as the rotation is performed prior to writing to the display buffer.

REG[36h] Special Effects Register
Default = 00h — Read/Write

Window Data Type	Double Buffer Enable	n/a				Window Pixel Sizing bits 1-0	
7	6	5	4	3	2	1	0

bit 7 Window Data Type

When this bit = 0, the data being written from the Host is intended for single buffer only.
When this bit = 1, the data being written from the Host is intended for double buffer operation.

If the Input Data Format is YUV 4:2:0 (REG[2Ah] bits 4-0 = 1001), the Window Data Type should not be changed while the YYC is busy (REG[58h] bit 4 = 1).

Note
This bit must be set before the window being written. The window coordinates will be latched internally to be used by the display pipe during display cycles.

Note
This bit setting is necessary for the Double-Buffer architecture when enabled (bit 6=1)

Note
While double buffering is enabled, the window coordinates should not be modified.

Table 9-10: Window Data Type/Buffer Selection

REG[36h] Bit 7	REG[36h] Bit 6	Use Case
0	0	Single buffered window with no double buffering anywhere on the display.
0	1	Use this to write a single buffered window while preventing tearing in a previously defined double buffered window.
1	0	Reserved
1	1	Use this to write data to be double buffered.

bit 6 Double Buffer Enable

This bit enables the Double Buffer architecture.
When this bit = 0, the double buffer is disabled.
When this bit =1, the double buffer is enabled. This feature is only available if the memory size resulting from the display size and color depth will fit within the 1/2 the allowable size for the display buffer.

When enabled, this feature is intended for streaming input sources to prevent visual tearing when updating the display.

Note
This bit must be set before the window being written. The window coordinates will be latched internally to be used by the display pipe during display cycles.

Note
While double buffering is enabled, the window coordinates should not be modified.

Note
Only one window can be double-buffered. All other windows are single buffered.

Table 9-11: Window Data Type Selection

REG[36h] Bit 7	REG[36h] Bit 6	Use Case
0	0	Single buffered window with no double buffering anywhere on the display.
0	1	Use this to write a single buffered window while preventing tearing in a previously defined double buffered window.
1	0	Reserved
1	1	Use this to write data to be double buffered.

bits 1-0 Window Pixel Sizing bits [1:0]

Table 9-12: Window Pixel Sizing

REG[36h] bits 1-0	Result
00	No Resizing
01	Pixel Doubling
10	Pixel Halving
11	Reserved

Note
These bits must be set before the window being written. The window coordinates will be latched internally to be used by the display pipe during display cycles.

Note
Only 1 active window can be pixel doubled. The pixel doubling design uses horizontal and vertical averaging for smooth doubling.

Note
The sizing is performed with respect to the top left corner

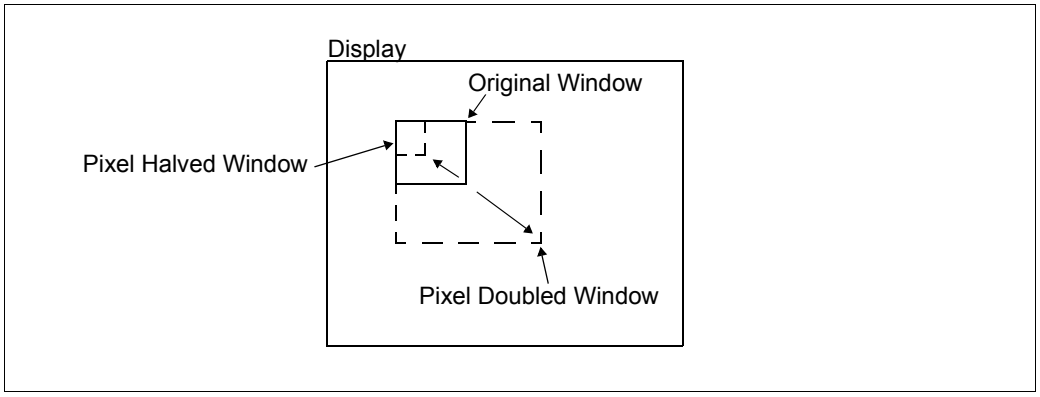


Figure 9-1: Sizing Example

Note
To turn off pixel doubling for a currently pixel doubled window, either:
1. Overwrite any part of the pixel doubled window with a new window.
2. Write a new pixel doubled window.

9.3.6 Window Settings

REG[38h] Window X Start Position Register 0
Default = 00h — Read/Write

7	6	5	4	3	2	1	0
Window X Start Position bits 7-0							

REG[3Ah] Window X Start Position Register 1
Default = 00h — Read/Write

7	6	5	4	3	2	1	0
n/a						Window X Start Position bits 9-8	

REG[3Ah] bits 1-0
REG[38h] bits 7-0 Window X Start Position bits [9:0]

These bits determine the X start position of the window in relation to the top left corner of the displayed image. Even in a rotated orientation, the top left corner is still relative to the displayed image.

Note
When pixel doubling or pixel halving is enabled, these registers should be programmed with the pre-resized coordinates.

REG[3Ch] Window Y Start Position Register 0
Default = 00h — Read/Write

7	6	5	4	3	2	1	0
Window Y Start Position bits 7-0							

REG[3Eh] Window Y Start Position Register 1
Default = 00h — Read/Write

7	6	5	4	3	2	1	0
n/a						Window Y Start Position bits 9-8	

REG[3Eh] bits 1-0
REG[3Ch] bits 7-0 Window Y Start Position bits [9:0]

These bits determine the Y start position of the window in relation to the top left corner of the displayed image. Even in a rotated orientation, the top left corner is still relative to the displayed image.

Note
When pixel doubling or pixel halving is enabled, these registers should be programmed with the pre-resized coordinates.

REG[40h] Window X End Position Register 0
Default = 00h — Read/Write

7	6	5	4	3	2	1	0
Window X End Position bits 7-0							

REG[42h] Window X End Position Register 1
Default = 00h — Read/Write

7	6	5	4	3	2	1	0
n/a						Window X End Position bits 9-8	

REG[42h] bits 1-0
REG[40h] bits 7-0 Window X End Position bits [9:0]

These bits determine the X end position of the window in relation to the top left corner of the displayed image. Even in a rotated orientation, the top left corner is still relative to the displayed image.

Note

When pixel doubling or pixel halving is enabled, these registers should be programmed with the pre-resized coordinates.

REG[44h] Window Y End Position Register 0
Default = 00h — Read/Write

7	6	5	4	3	2	1	0
Window Y End Position bits 7-0							

REG[46h] Window Y End Position Register 1
Default = 00h — Read/Write

7	6	5	4	3	2	1	0
n/a						Window Y End Position bits 9-8	

REG[46h] bits 1-0
REG[44h] bits 7-0 Window Y End Position bits [9:0]

These bits determine the Y end position of the window in relation to the top left corner of the displayed image. Even in a rotated orientation, the top left corner is still relative to the displayed image.

Note

When pixel doubling or pixel halving is enabled, these registers should be programmed with the pre-resized coordinates.

9.3.7 Memory Access

REG[48h] Memory Data Port Register 0							
Default = XXh							Read/Write
Memory Data Port bits [7:0]							
7	6	5	4	3	2	1	0

REG[49h] Memory Data Port Register 1							
Default = XXh							Read/Write
Memory Data Port bits [15:8]							
7	6	5	4	3	2	1	0

REG[48h] bits 7-0 Memory Data Port bits [7:0]
These specify the lsb for the data word

REG[49h] bits 7-0 Memory Data Port bits [15:8]
These bits specify the msb of the data word.

Note
If CNF1=0 (8-bit interface), REG[49h] is not used.
The data read back from memory will be byte swapped (i.e. if 12 34 56 78 is written to memory, data read back will be 34 12 78 56).

Note
Burst data writes are supported through this register. Register auto-increment is automatically disabled once reaching this address. All writes to this register will auto-increment the internal memory address only.

Note
Panel dimension registers must be set before writing any window data.

Note
Upon writing the last pixel in the defined window, this register will automatically point back to the first pixel in the window. Therefore there is no need to re-initialize the pointers.

REG[4Ah] Memory Read Address Register 0
Default = 00h — Read/Write

Memory Address bits 7-0							
7	6	5	4	3	2	1	0

REG[4Ch] Memory Read Address Register 1
Default = 00h — Read/Write

Memory Address bits 15-8							
7	6	5	4	3	2	1	0

REG[4Eh] Memory Read Address Register 2
Default = 00h — Read/Write

n/a				Memory Address bit 19-16			
7	6	5	4	3	2	1	0

REG[4Eh] bits 3-0
REG[4Ch] bits 7-0
REG[4Ah] bits 7-0 Memory Read Address bits [19:0]

This register is only used for individual memory location reads.
Individual memory location writes are not supported.
After a completed memory access, this register is incremented automatically.
To perform memory reads:

- perform a register address write to point to this register
- followed by 3 data writes to set-up the memory address
- read the Memory Data Port (REG[48h - 49h])

Note
All write data uses the Memory Data Port and the Window coordinates.

Note
For Intel 80, 16-bit interface, the least significant bit is not used (data is fetched on word boundaries).
For Intel 80, 8-bit interface, the least significant bit is used (data is fetched on byte boundaries)

9.3.8 Gamma Correction Registers

Note
Gamma correction is implemented as a look-up table. RGB input data (if the input is YUV, it will be converted to RGB first) is used to look-up the values from the programmed tables. The Gamma LUT's are placed on the display read path and the 18-bit (6 msb's from each channel) output goes to the LCD interface.

Note
The Gamma Correction Tables should not be accessed during display period as this will result in visual anomalies. All updates to the LUT's should be performed during non-display period or when the LUT's are disabled and not in use.

REG[50h] Gamma Correction Enable Register
Default = 00h — Read/Write

7	6	5	4	3	2	1	0
n/a					Look-Up Table Access Mode bits 1-0		Gamma Correction Enable

bits 2-1 Look-Up Table Access Mode bits [1:0]

Table 9-13: Look-Up Table Access Mode

REG[50h] bits 2-1	Description
00	Writing will be done to all Red, Green, & Blue tables. Reading will be done from Red table.
01	Reading and writing will be done to Red table.
10	Reading and writing will be done to Green table.
11	Reading and writing will be done to Blue table.

bit 0 Gamma Correction Enable

When this bit = 0, gamma correction is disabled and the input data will bypass the gamma correction look-up table. In this case, data stored as 16 bpp will automatically be converted to 18 bpp by copying the Red and Blue msb to create new lsb's. This will be performed on the display read therefore not requiring any additional memory.

When this bit = 1, gamma correction is enabled and the input data will go through the gamma correction look-up table.

Note
The Gamma Correction Tables should not be accessed during display period as this will result in visual anomalies. All updates to the LUT's should be performed during non-display period or when the LUT's are disabled and not in use.

REG[52h] Gamma Correction Table Index Register

Default = 00h — Read/Write

7	6	5	4	3	2	1	0
n/a		Gamma Correction Table Index bits 5-0					

bits 5-0 Gamma Correction Table Index bits [5:0]
These bits will specify the index of the gamma correction look-up table which subsequent read/write will start at.

REG[54h] Gamma Correction Table Data Register

Default = XXh — Read/Write

7	6	5	4	3	2	1	0
Gamma Correction Table Data bits 5-0							

bits 7-0 Gamma Correction Table Data bits [7:0]

When writing to Gamma Correction Table Data register, the index to the internal table will be automatically incremented. For continuous update to the table, the Gamma Correction Table Index register needs only to be written once. The index will incremented by 1 for every write to Gamma Correction Table Data register.

Note
Although bits 7 and 6 are programmed to the LUT, they are ignored in the final output from the LUT.

Note
All 64 positions of each LUT must be written when using auto-increment writes.In the 5:6:5 case, the first 32 positions of the Red and Blue LUT's will be used.

9.3.9 Miscellaneous Registers

REG[56h] Power Save Register
Default = 00h — Read/Write

7	6	5	4	3	2	1	0
PWRSVE Input Pin Function	n/a					Sleep Mode Enable/Disable	Standby Mode Enable/Disable

bit 7 PWRSVE Input Pin Function
When this bit = 0, the PWRSVE pin is OR'd with bit 1 (setting either to 1 will enable Sleep Mode)
When this bit = 1, the PWRSVE pin is OR'd with bit 0 (setting either to 1 will enable Standby Mode)

bit 1 Sleep Mode Enable/Disable
When this bit = 0, Sleep Mode is disabled (normal operation)
When this bit = 1, Sleep Mode is enabled.

Sleep Mode disables all internal blocks including the PLL. When Sleep Mode is disabled (low), the PLL requires approximately 10msec lock time before any memory access should be attempted. The PLL Lock bit, REG[04] bit 7, can be read to verify when the PLL becomes stable.

bit 0 Standby Mode Enable/Disable
When this bit = 0, Standby Mode is disabled (normal operation)
When this bit = 1, Standby Mode is enabled

Standby Mode disables all internal blocks except the PLL. Using this mode, the chip can be accessed immediately when Standby is disabled.

Note
Standby Mode can also be enabled/disabled using the PWRSVE input pin.

REG[58h] Non-Display Period Control / Status Register
Default = 00h — Read/Write

7	6	5	4	3	2	1	0
Vertical Non-Display Period Status (RO)	Horizontal Non-Display Period Status (RO)	VS OR'd with HS Status (RO)	YYC Last Line	n/a	TE Output Pin Enable	TE Output Pin Function Select bits 1-0	

bit 7 Vertical Non-Display Period Status
This is a read-only status bit.
When this bit = 0, the LCD panel output is in a Vertical Non-Display Period.
When this bit = 1, the LCD panel output is in a Vertical Display Period.

Note
VNDP is defined as time between the last pixel on the last line of one frame to the first pixel on the first line of the next frame.

bit 6 Horizontal Non-Display Period Status
This is a read only status bit
When this bit = 0, the LCD panel output is in a Horizontal Non-Display Period
When this bit = 1, the LCD panel output is in a Horizontal Display Period

Note
HNDP is defined as the time between the last pixel in line n to the first pixel in line n+1.

bit 5 VDP OR'd with HDP Status
This bit is a read only status bit.
When this bit = 0, the LCD panel output is in either the Horizontal or Vertical Non-Display period.
When this bit = 1, the LCD panel output is in a Display period.

bit 4 YYC Last Line
If the input data type is YUV 4:2:0, this bit will go high 5 MClk's after the Intel 80 interface has finished writing the last pixel of the current window.

This bit will go low once the YYC is idle. At this point, a new window can be written.

When doing back-to-back window writes with a different dimension or format, and the first window is YUV 4:2:0, before starting to write the second window, make sure this bit is low.

Note
It can take up to five SYSCLKs from the rising edge of WE# of the last byte/word of a frame before this bit is set.

bit 2 TE Output Pin Enable
When this bit = 0, the TE output pin is disabled
When this bit = 1, the TE output pin is enabled.

bits 1-0 TE Output Pin Function Select bits [1:0]

Table 9-14: TE Output Pin Function Select

REG[58h] bits 1-0	TE Output Pin Function
00	Reserved
01	Horizontal Non-Display Period
10	Vertical Non-Display Period
11	HS OR'd with VS

9.3.10 General Purpose IO Pins Registers

REG[5Ah] General Purpose IO Pins Configuration Register 0
Default =00h — Read/Write

GPIO7 Configuration	GPIO6 Configuration	GPIO5 Configuration	GPIO4 Configuration	GPIO3 Configuration	GPIO2 Configuration	GPIO1 Configuration	GPIO0 Configuration
7	6	5	4	3	2	1	0

bits 7-0 GPIO[7:0] Configuration
When this bit = 0 (normal operation), the associated GPIO is configured as an input pin.
When this bit = 1, the associated GPIO is configured as an output pin.

Note
When configured as an input or an output, the associated GPIO can also be configured to produce an interrupt (GPIO_INT) based on selectable Interrupt Trigger conditions (see REG[5E], [60])

REG[5Ch] General Purpose IO Pins Status/Control Register 0
Default = 00h — Read/Write

GPIO7 Status	GPIO6 Status	GPIO5 Status	GPIO4 Status	GPIO3 Status	GPIO2 Status	GPIO1 Status	GPIO0 Status
7	6	5	4	3	2	1	0

bits 7-0 GPIO[7:0] Status
When the associated GPIO is configured as an output, writing a 1 to this bit drives it high and writing a 0 to this bit drives it low.
When the associated GPIO is configured as an input, a read from this bit returns the raw status.

Note
When configured as an output, the GPIO_INT pin can still be toggled by writing the appropriate value to this register if enabled by REG[5E],[60].

REG[5Eh] GPIO Positive Edge Interrupt Trigger Register
Default = 00h — Read/Write

GPIO7 Positive Edge Interrupt Trigger	GPIO6 Positive Edge Interrupt Trigger	GPIO5 Positive Edge Interrupt Trigger	GPIO4 Positive Edge Interrupt Trigger	GPIO3 Positive Edge Interrupt Trigger	GPIO2 Positive Edge Interrupt Trigger	GPIO1 Positive Edge Interrupt Trigger	GPIO0 Positive Edge Interrupt Trigger
7	6	5	4	3	2	1	0

bits 7-0 GPIO[7:0] Positive Edge Interrupt Trigger
Setting these bits = 1, will enable the associated interrupt.
This bit determines whether the associated GPIO interrupt is triggered on the positive edge (when the GPIOx pin changes from 0 to 1).
When this bit = 0, the associated GPIO interrupt (GPIO_INT) is disabled.
When this bit = 1, the associated GPIO interrupt (GPIO_INT) is triggered on the positive edge.
Once triggered, the GPIO_INT pin will toggle from 0 to 1. The GPIO_INT pins is cleared (non-active state (0)) by clearing the associated GPIO Interrupt Status bit (REG[62])

REG[60h] GPIO Negative Edge Interrupt Trigger Register
Default = 00h — Read/Write

GPIO7 Negative Edge Interrupt Trigger	GPIO6 Negative Edge Interrupt Trigger	GPIO5 Negative Edge Interrupt Trigger	GPIO4 Negative Edge Interrupt Trigger	GPIO3 Negative Edge Interrupt Trigger	GPIO2 Negative Edge Interrupt Trigger	GPIO1 Negative Edge Interrupt Trigger	GPIO0 Negative Edge Interrupt Trigger
7	6	5	4	3	2	1	0

bits 7-0 GPIO[7:0] Negative Edge Interrupt Trigger

Setting these bits = 1, will enable the associated interrupt.

This bit determines whether the associated GPIO interrupt is triggered on the negative edge (when the GPIOx pin changes from 1 to 0).

When this bit = 0, the associated GPIOx interrupt (GPIO_INT) is disabled.

When this bit = 1, the associated GPIOx interrupt (GPIO_INT) is triggered on the negative edge.

Once triggered, the GPIO_INT pin will toggle from 0 to 1. The GPIO_INT pins is cleared (non-active state (0)) by clearing the associated GPIO Interrupt Status bit (REG[62])

REG[62h] GPIO Interrupt Status Register
Default = 00h — Read/Write

GPIO7 Interrupt Status	GPIO6 Interrupt Status	GPIO5 Interrupt Status	GPIO4 Interrupt Status	GPIO3 Interrupt Status	GPIO2 Interrupt Status	GPIO1 Interrupt Status	GPIO0 Interrupt Status
7	6	5	4	3	2	1	0

bits 7-0 GPIO[7:0] Interrupt Status

If configured to generate an Interrupt (GPIO_INT), this status bit will show which GPIO generated the interrupt. To clear this status bit, you must perform two writes to it: first write = 1, the second write = 0.

Note

The GPIO_INT pin will also toggle back to 0 upon clearing the status. However, if the original interrupt condition still exists on the GPIO input pin, the GPIO_INT will immediately set again.

REG[64h] GPIO Pull Down Control Register 0
Default = FFh — Read/Write

GPIO7 Pull-down Control	GPIO6 Pull-down Control	GPIO5 Pull-down Control	GPIO4 Pull-down Control	GPIO3 Pull-down Control	GPIO2 Pull-down Control	GPIO1 Pull-down Control	GPIO0 Pull-down Control
7	6	5	4	3	2	1	0

bits 7-0 GPIO[7:0] Pull-down Control

All GPIO pins have internal pull-down resistors. These bits individually control the state of the pull-down resistors.

When the bit = 0, the pull-down resistor for the associated GPIO pin is inactive.

When the bit = 1, the pull-down resistor for the associated GPIO pin is active.

10 Frame Rate Calculation

The following formula is used to calculate the display frame rate.

$$\text{FrameRate} = \frac{f_{PCLK}}{(HT) \times (VT)}$$

Where:

f_{PCLK} = PClk frequency (Hz)

HT = Horizontal Total
= Horizontal Display Width + Horizontal Non-Display Period

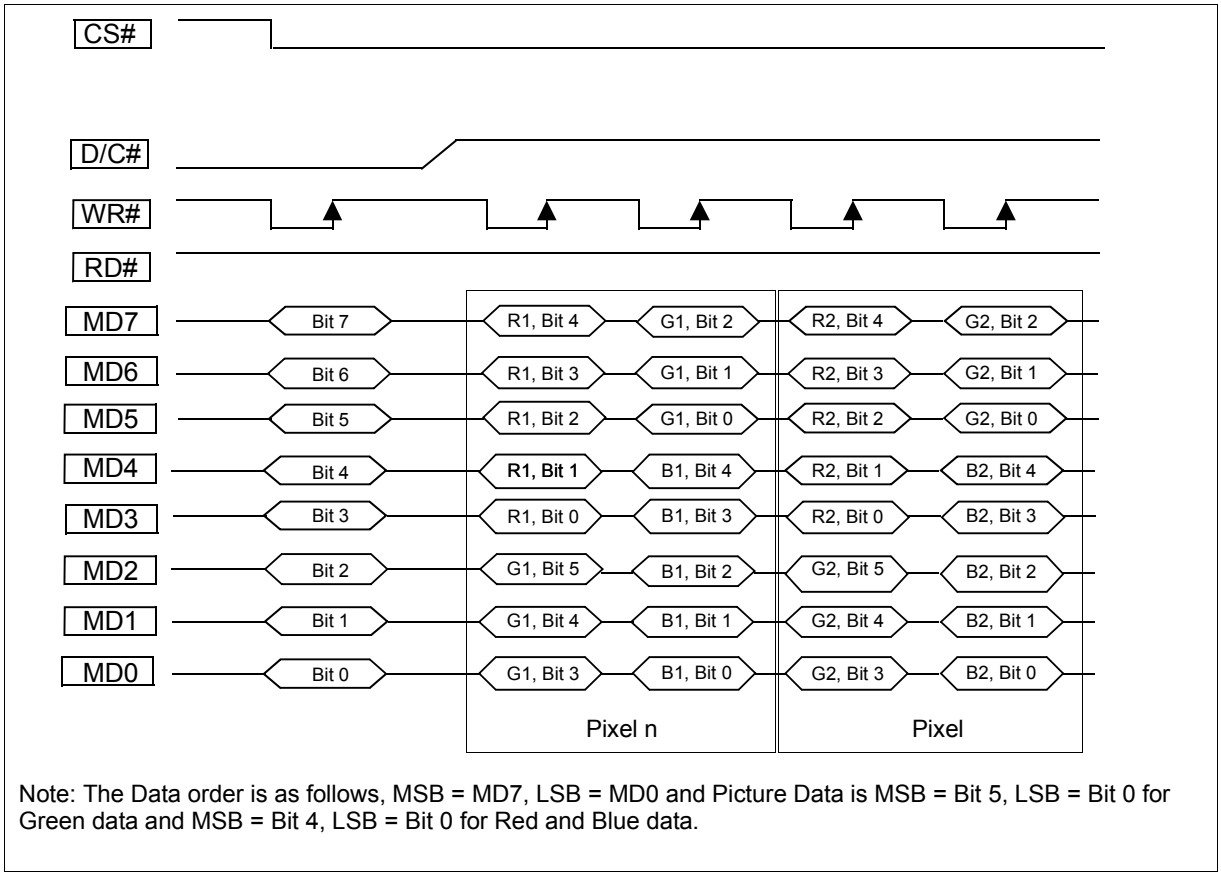
VT = Vertical Total
= Vertical Display Height + Vertical Non-Display Period

Note

For definitions of panel timing parameters, see Section 7.4, “Display Interface” on page 34.

11 Intel 80, 8-bit Interface Color Formats

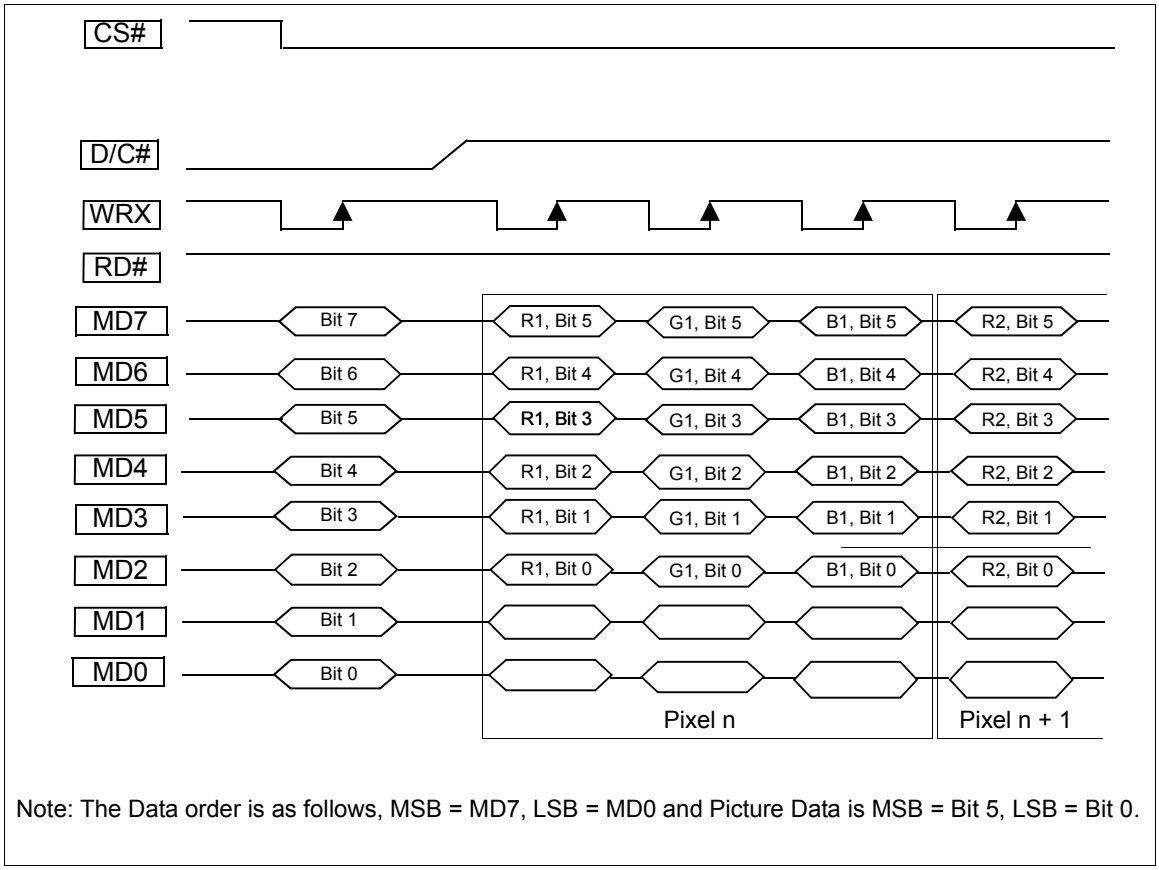
11.1 16 bpp Mode (R 5-bit, G 6-bit, B 5-bit), 65,536 colors



Note: The Data order is as follows, MSB = MD7, LSB = MD0 and Picture Data is MSB = Bit 5, LSB = Bit 0 for Green data and MSB = Bit 4, LSB = Bit 0 for Red and Blue data.

Figure 11-1: 16 bpp Mode (R 5-bit, G 6-bit, B 5-bit), 65,536 colors

11.2 18 bpp (R 6-bit, G 6-bit, B 6-bit), 262,144 colors



Note: The Data order is as follows, MSB = MD7, LSB = MD0 and Picture Data is MSB = Bit 5, LSB = Bit 0.

Figure 11-2: 18 bpp (R 6-bit, G 6-bit, B 6-bit), 262,144 colors

11.3 24 bpp (R 8-bit, G 8-bit, B 8-bit), 16,777,216 colors

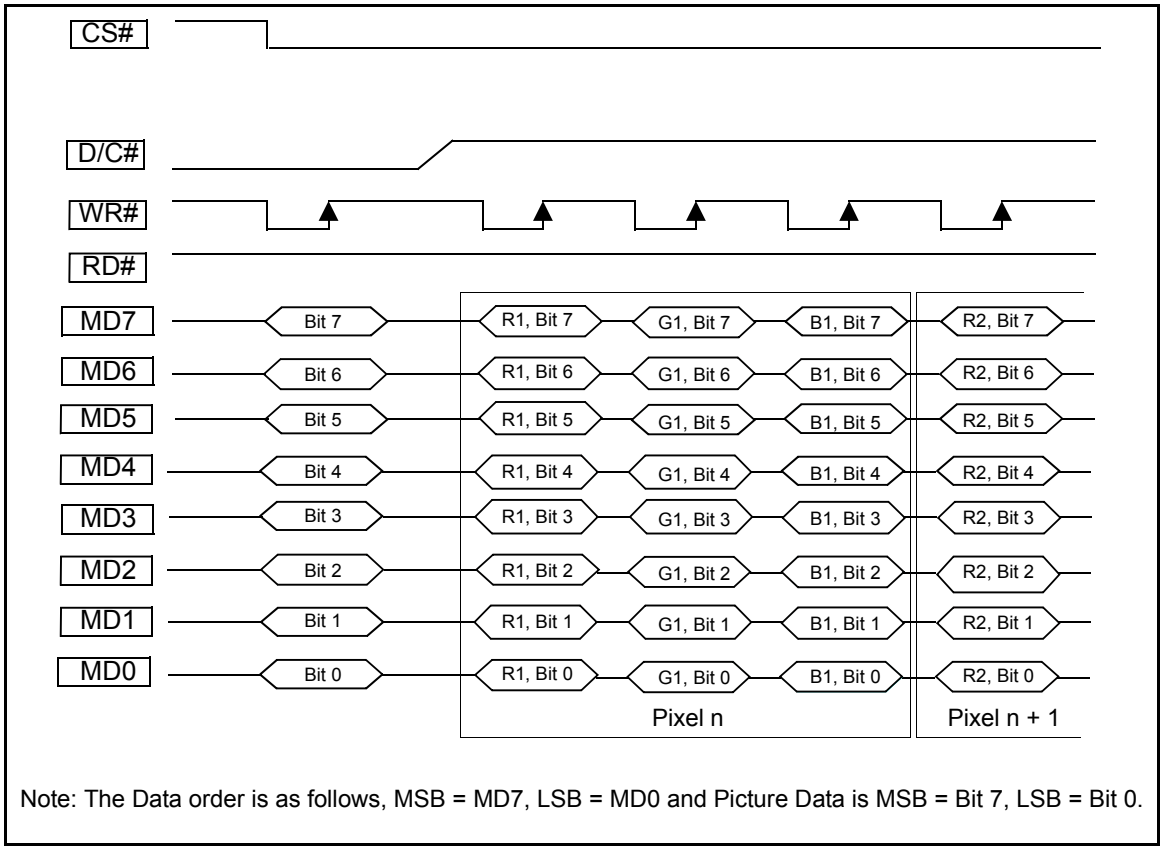
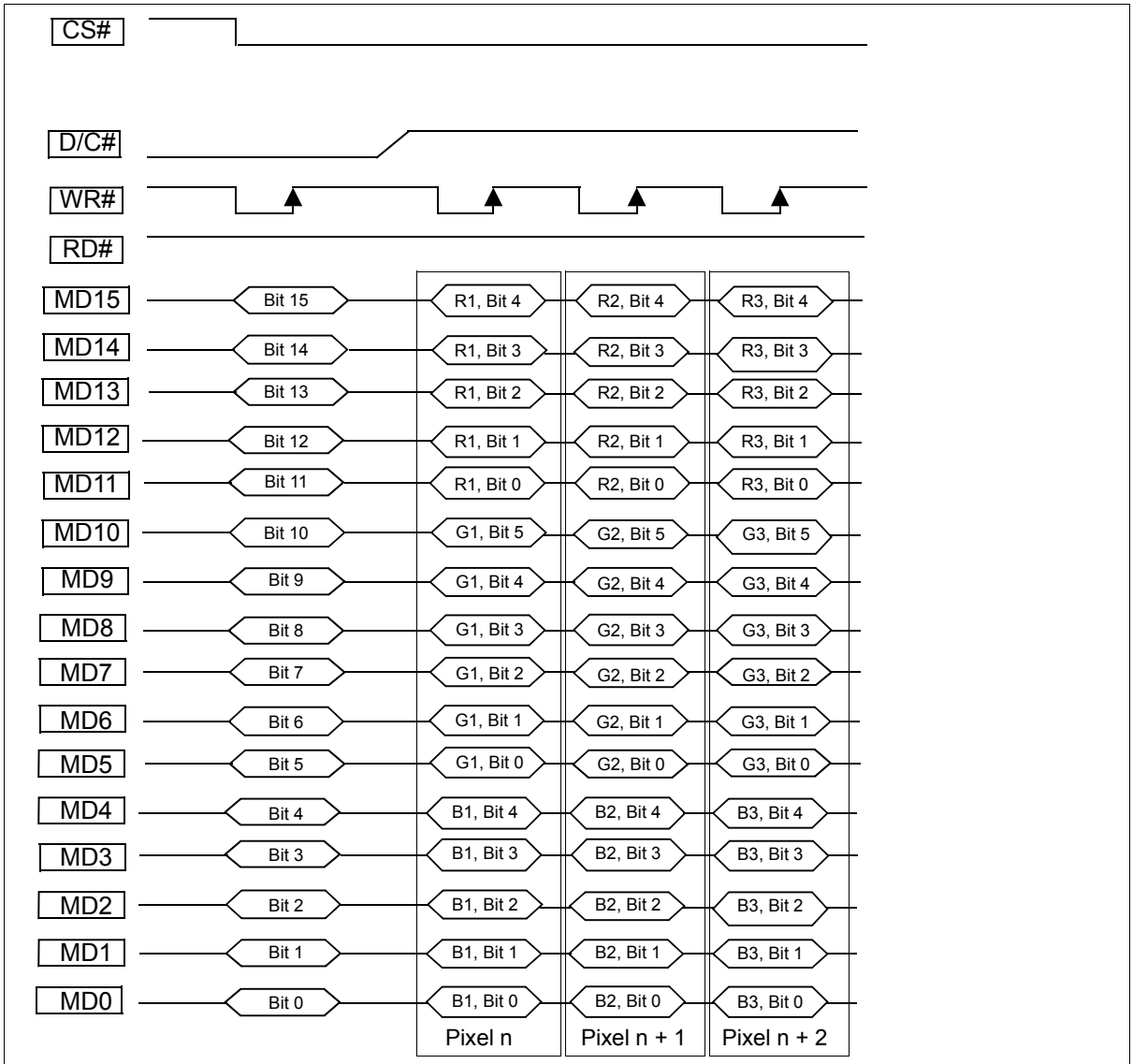


Figure 11-3: 24 bpp (R 8-bit, G 8-bit, B 8-bit), 16,777,216 colors

12 Intel 80, 16-bit Interface Color Formats

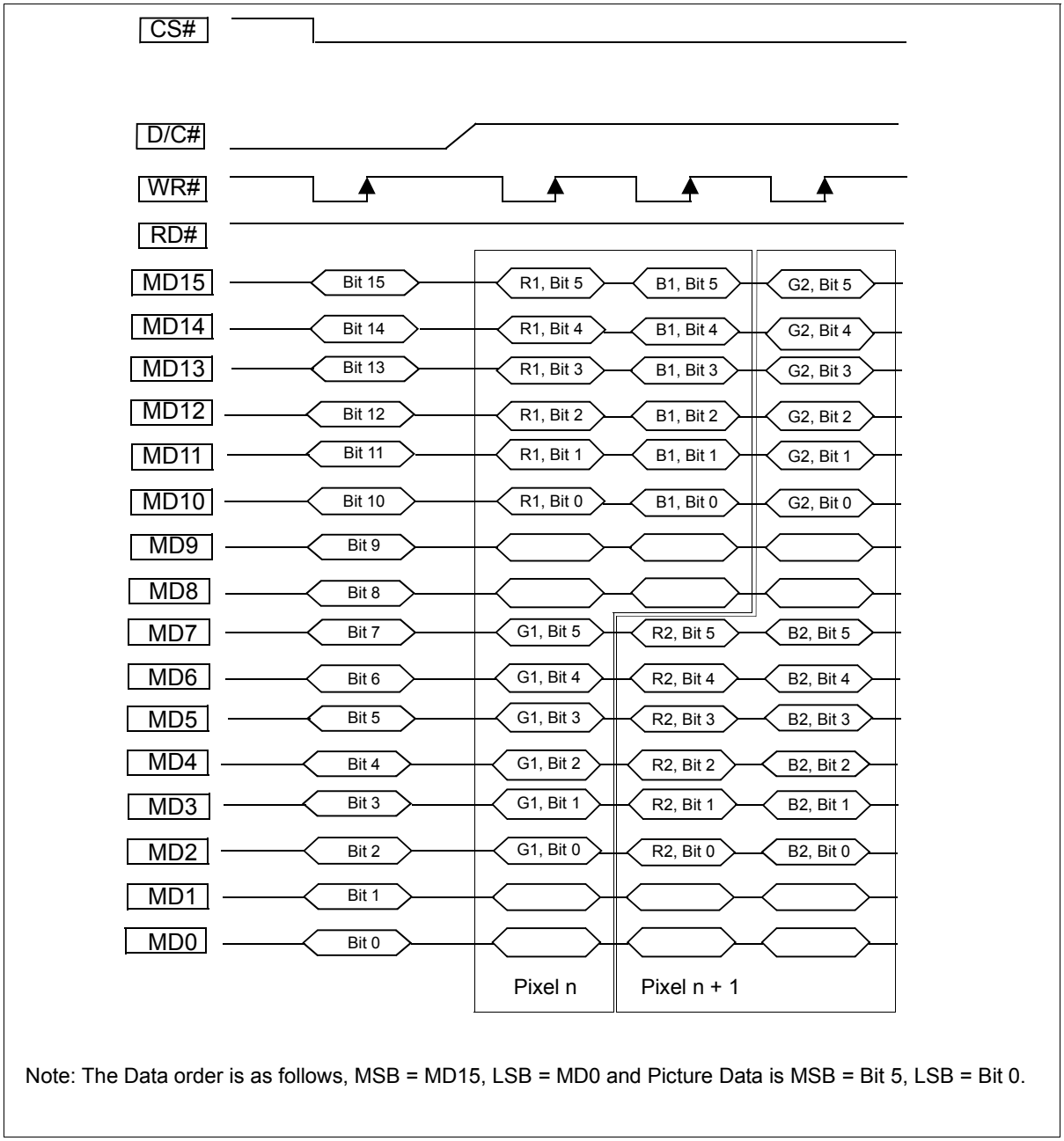
12.1 16 bpp (R 5-bit, G 6-bit, B 5-bit), 65,536 colors



Note: The Data order is as follows, MSB = MD15, LSB = MD0 and Picture Data is MSB = Bit 5, LSB = Bit 0 for Green data and MSB = Bit 4, LSB = Bit 0 for Red and Blue data.

Figure 12-1: 16 bpp (R 5-bit, G 6-bit, B 5-bit), 65,536 colors

12.2 18 bpp Mode 1 (R 6-bit, G 6-bit, B 6-bit), 262,144 colors



Note: The Data order is as follows, MSB = MD15, LSB = MD0 and Picture Data is MSB = Bit 5, LSB = Bit 0.

Figure 12-2: 18 bpp Mode 1(R 6-bit, G 6-bit, B 6-bit), 262,144 colors

12.3 18 bpp Mode 2 (R 6-bit, G 6-bit, B 6-bit), 262,144 colors

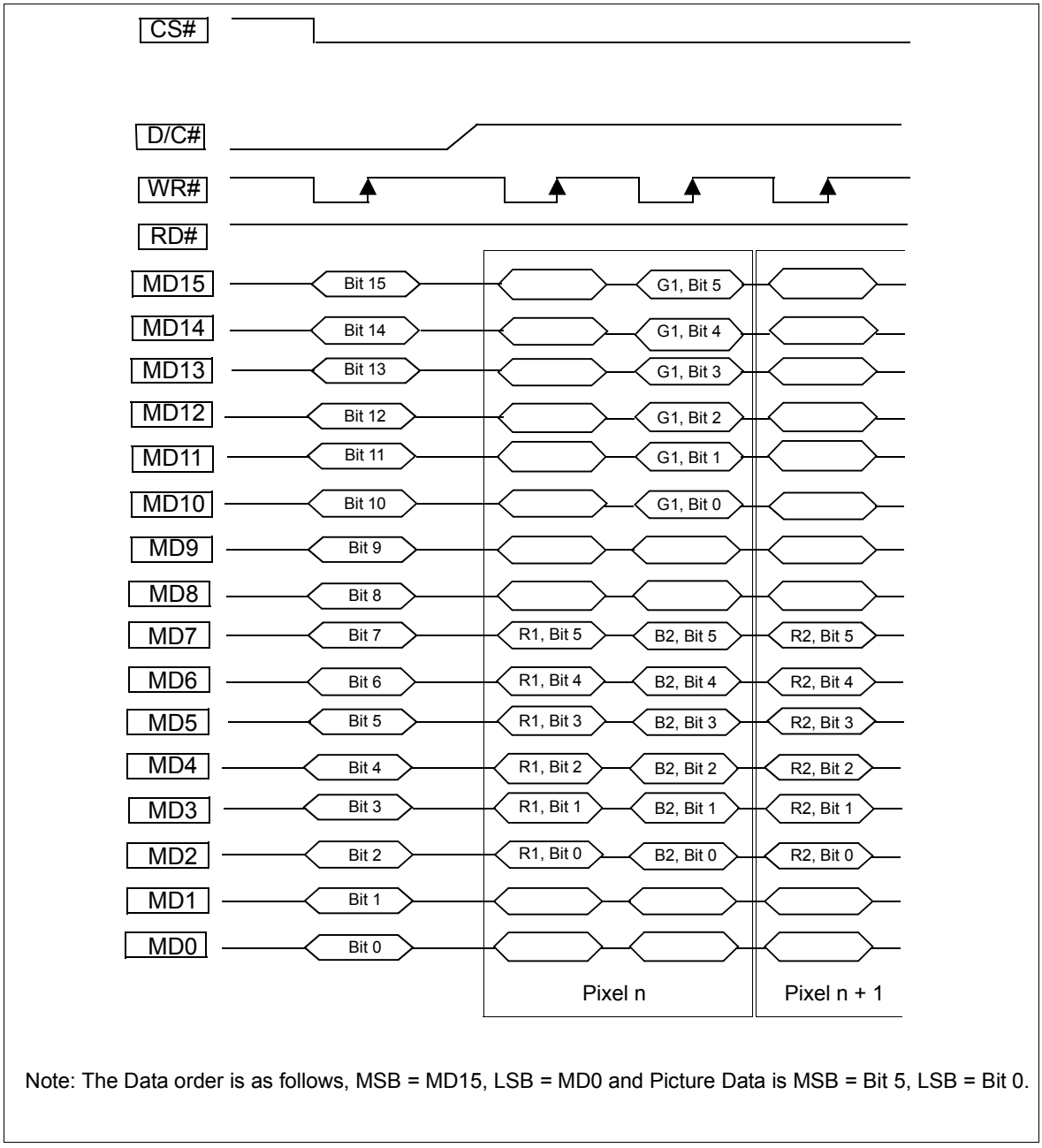
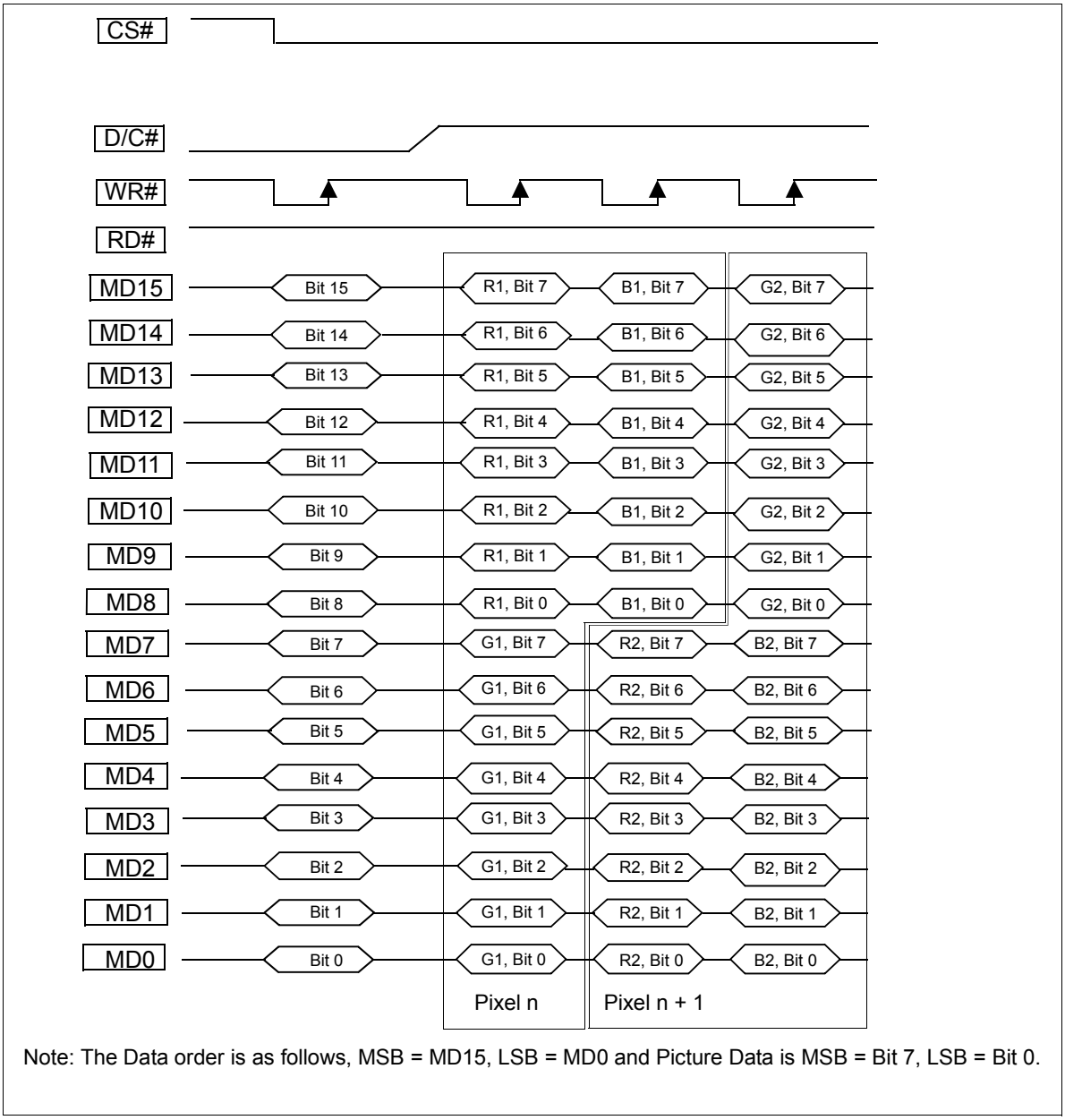


Figure 12-3: 18 bpp Mode 2 (R 6-bit, G 6-bit, B 6-bit), 262,144 colors

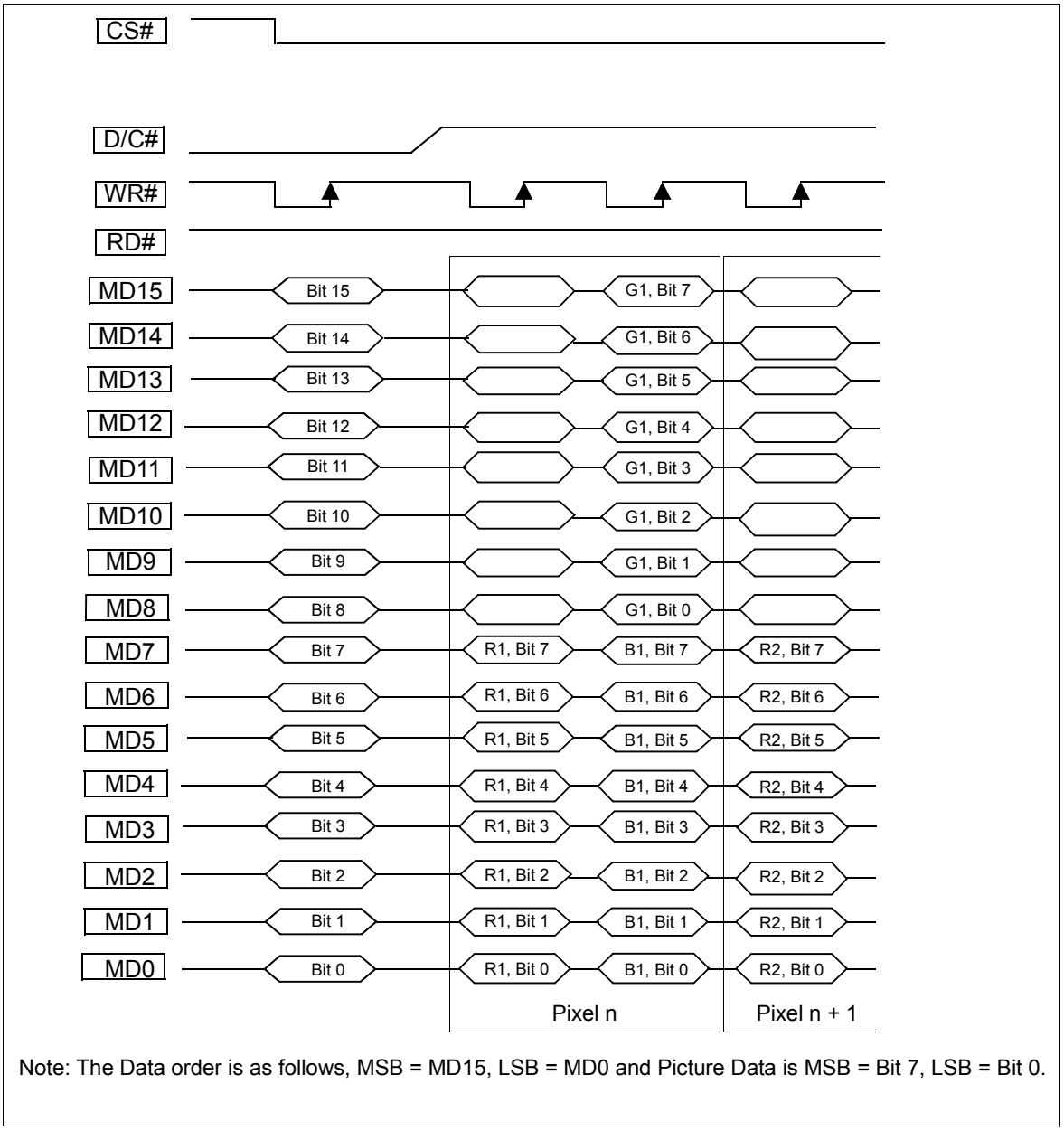
12.4 24 bpp Mode 1 (R 8-bit, G 8-bit, B 8-bit), 16,777,216 colors



Note: The Data order is as follows, MSB = MD15, LSB = MD0 and Picture Data is MSB = Bit 7, LSB = Bit 0.

Figure 12-4: 24 bpp Mode 1(R 8-bit, G 8-bit, B 8-bit), 16,777,216 colors

12.5 24 bpp Mode 2 (R 8-bit, G 8-bit, B 8-bit), 16,777,216 colors



Note: The Data order is as follows, MSB = MD15, LSB = MD0 and Picture Data is MSB = Bit 7, LSB = Bit 0.

Figure 12-5: 24 bpp Mode 2 (R 8-bit, G 8-bit, B 8-bit), 16,777,216 colors

13 YUV Timing

Format Definition

- The number of pixels per line is always even
- The YC_BC_R colorspace is defined in ITU-R BT601.4
- YUV 4:2:2 format
  $U_{11}Y_{11}V_{11}Y_{12}U_{13}Y_{13}V_{13}Y_{14}...$
- YUV 4:2:0 format
  Odd Line: $UY_{11}Y_{12}...$
  Even Line: $VY_{21}Y_{22}...$

Note
When a window is setup for YUV data, the data must always alternate between odd and even lines, starting with an odd line.

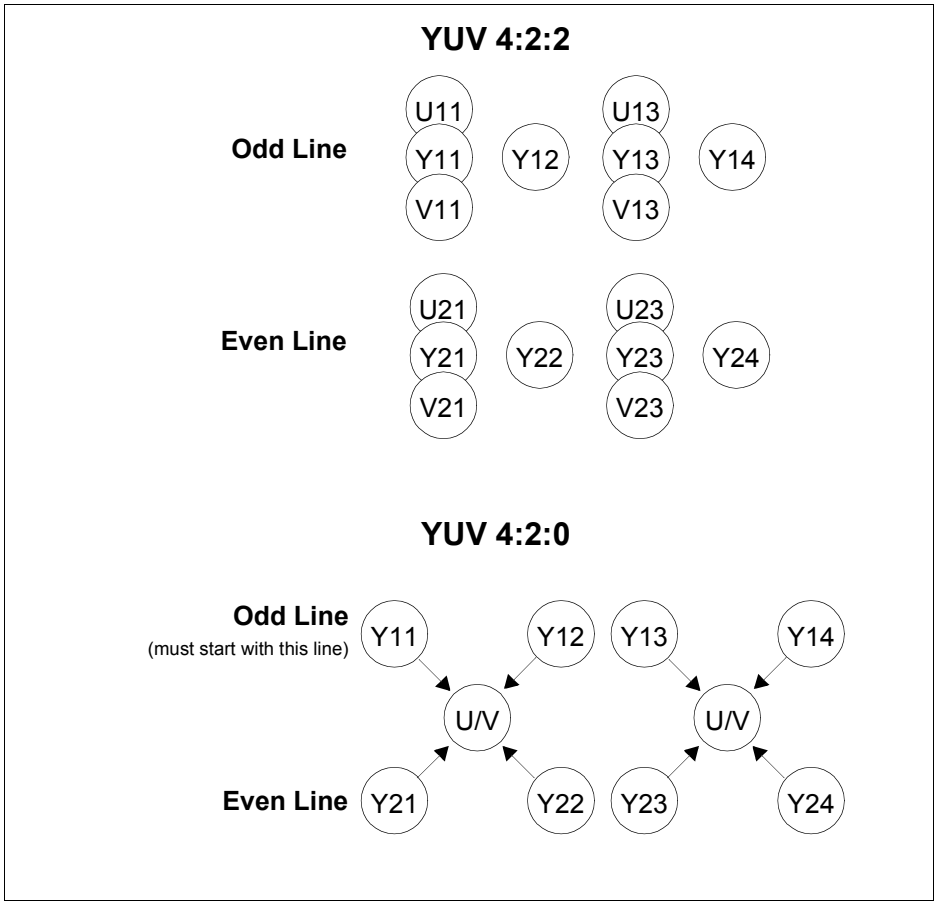


Figure 13-1: YUV Format Definition

13.1 YUV 4:2:2 with Intel 80, 8-bit Interface

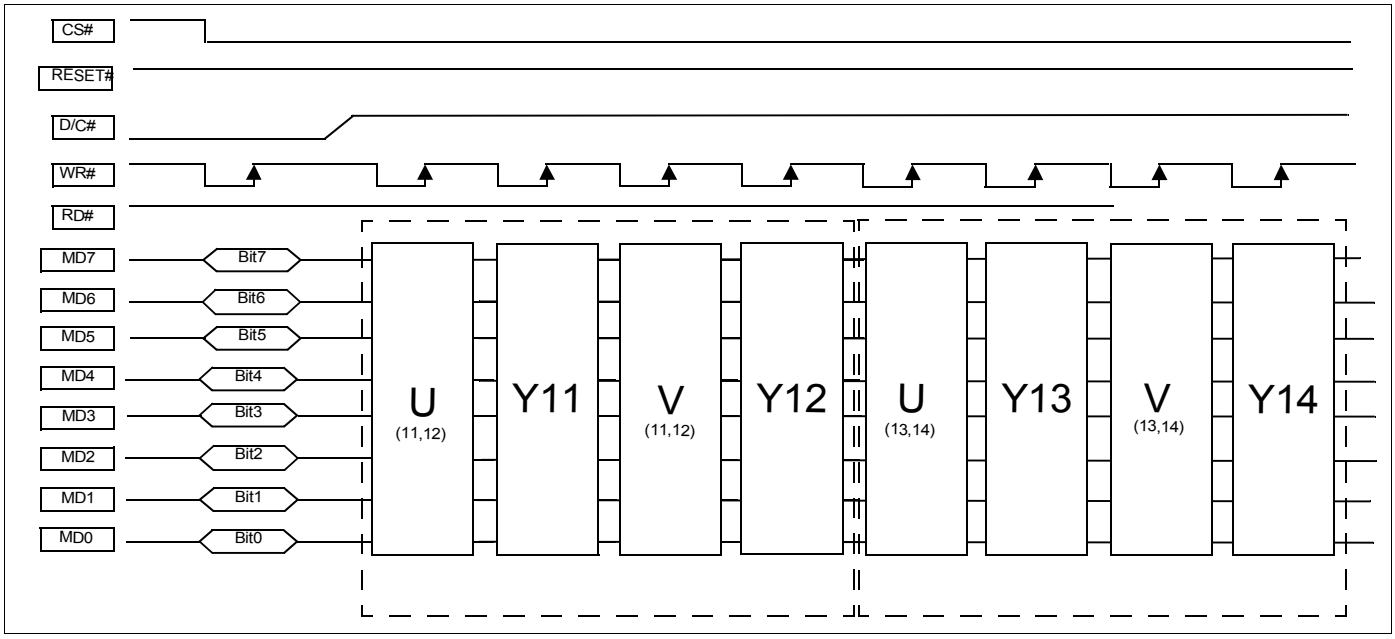


Figure 13-2: YUV 4:2:2 with Intel 80, 8-bit Interface

13.2 YUV 4:2:0 ODD Line with Intel 80, 8-bit Interface

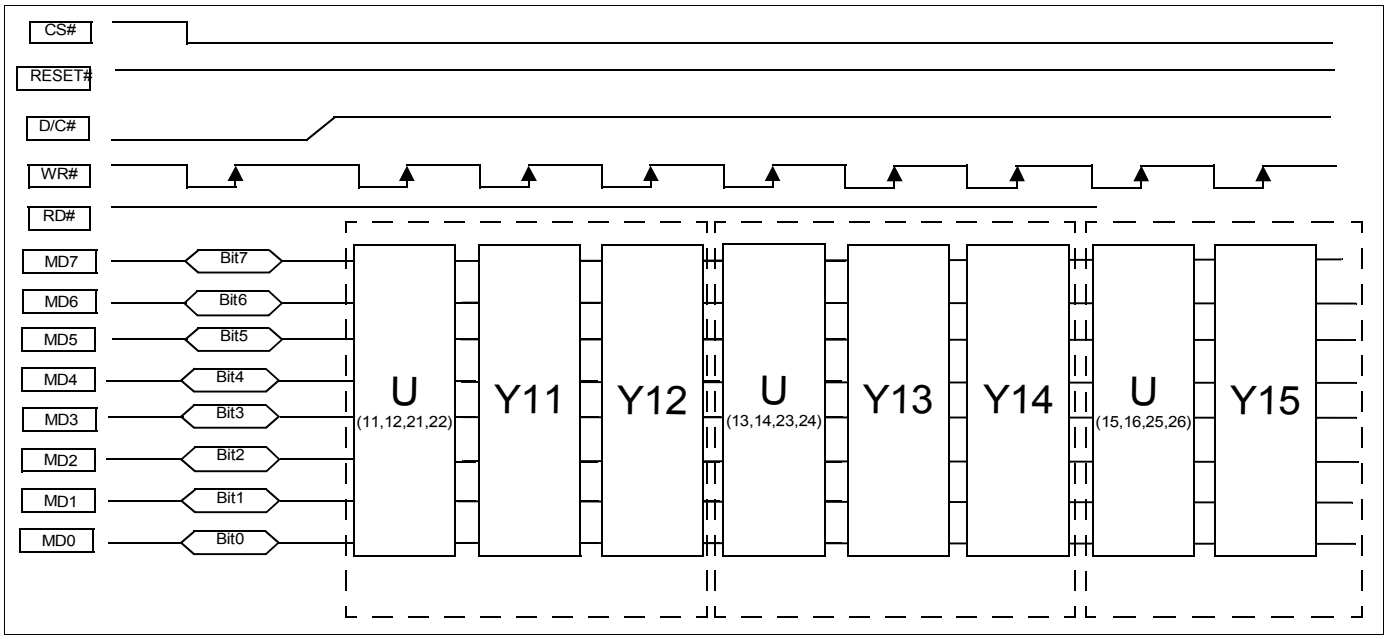


Figure 13-3: YUV 4:2:0 ODD Line with Intel 80, 8-bit Interface

13.3 YUV 4:2:0 EVEN Line with Intel 80, 8-bit Interface

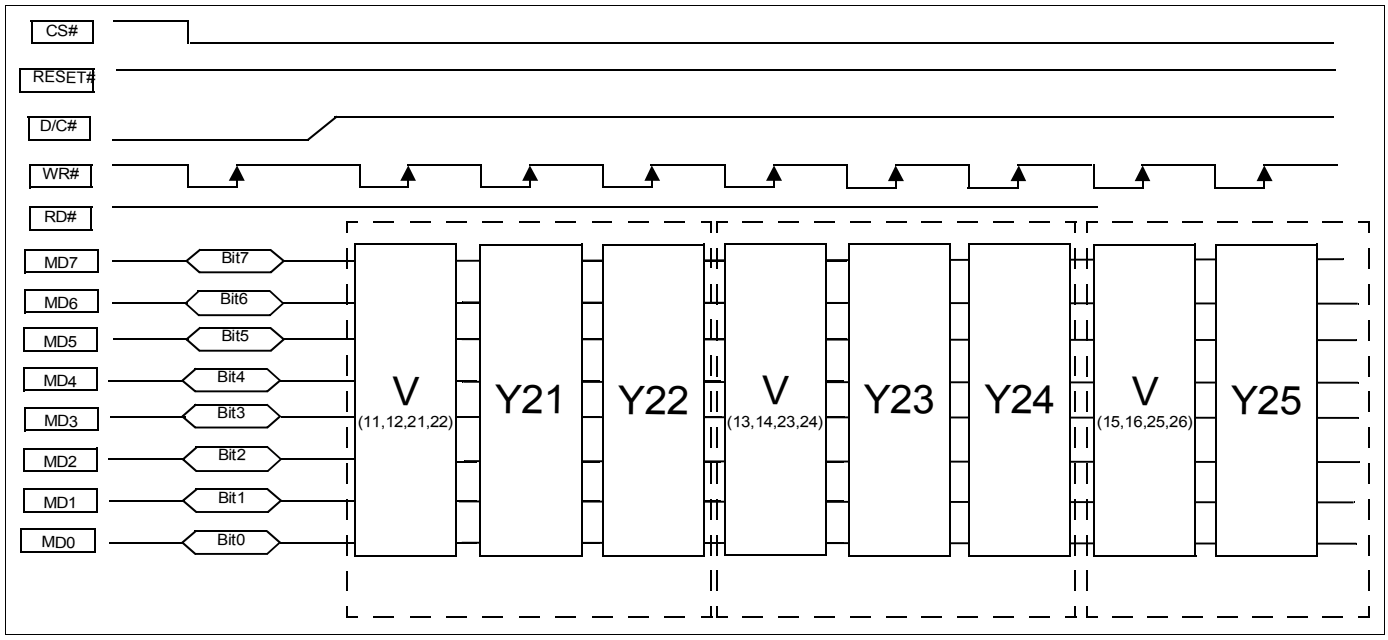


Figure 13-4: YUV 4:2:0 EVEN Line with Intel 80, 8-bit Interface

13.4 YUV 4:2:2 with Intel 80, 16-bit Interface

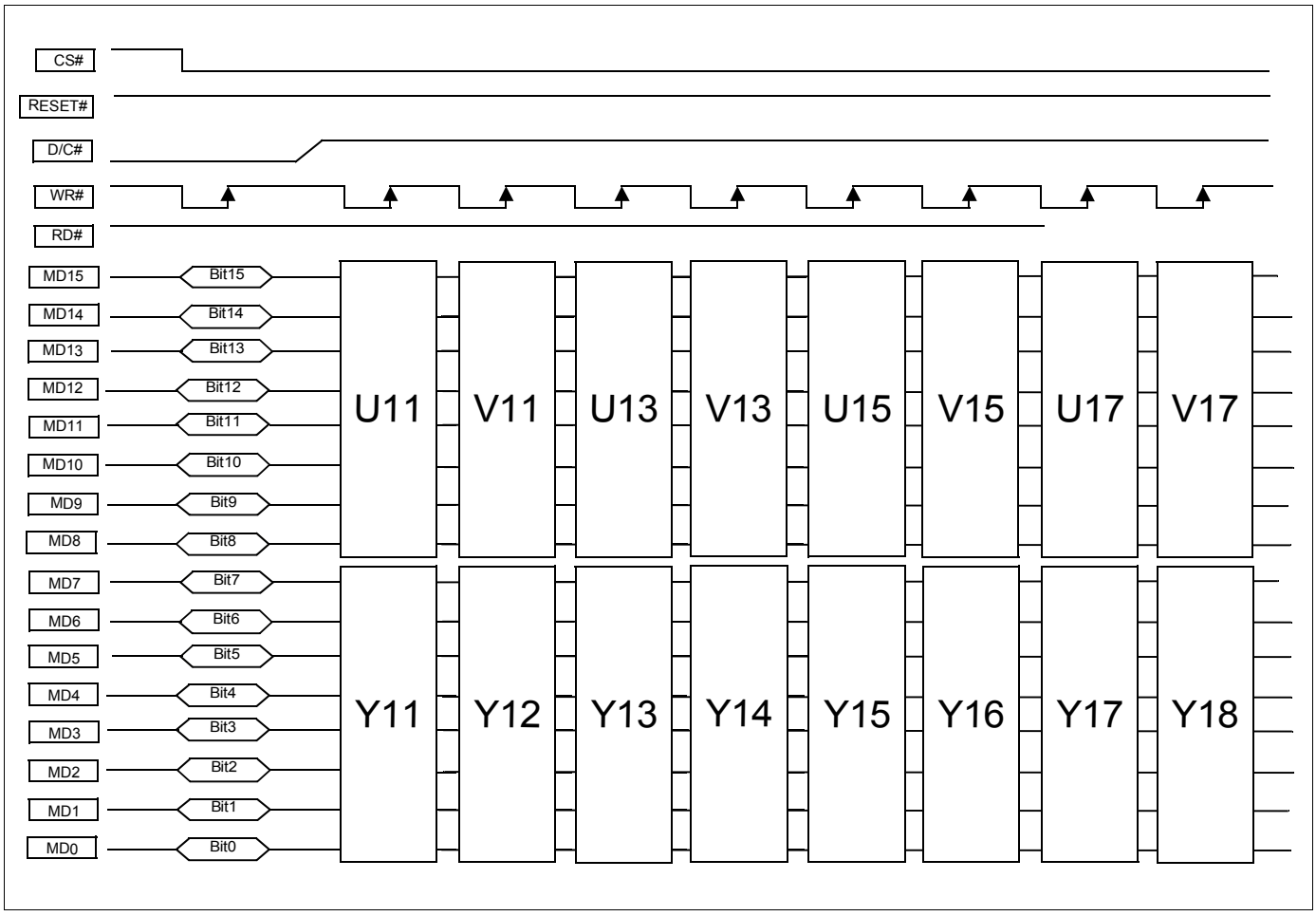


Figure 13-5: YUV 4:2:2 with Intel 80, 16-bit Interface

13.5 YUV 4:2:0 ODD Line with Intel 80, 16-bit Interface

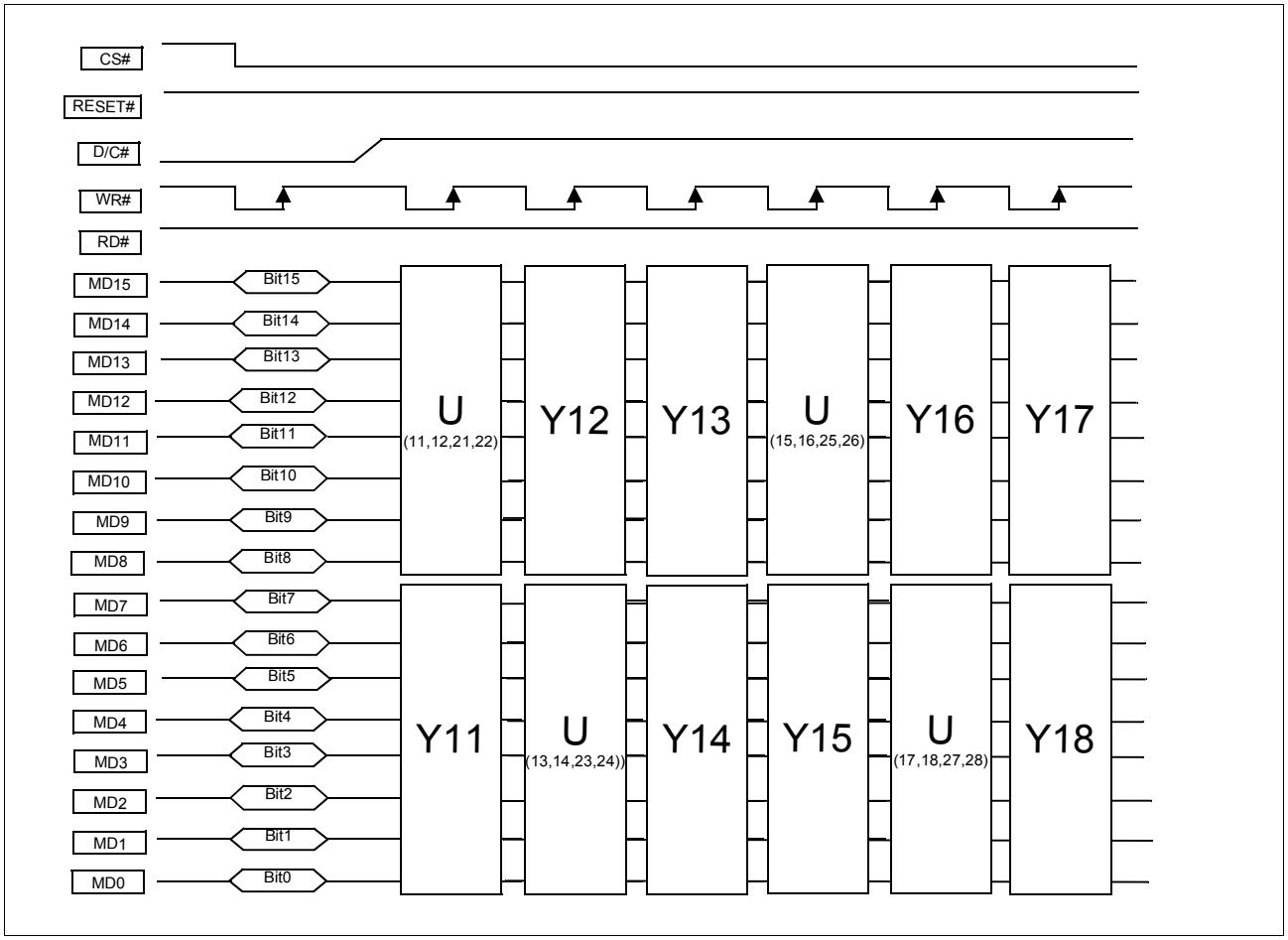


Figure 13-6: YUV 4:2:0 ODD Line with Intel 80, 16-bit Interface

13.6 YUV 4:2:0 EVEN Line with Intel 80, 16-bit Interface

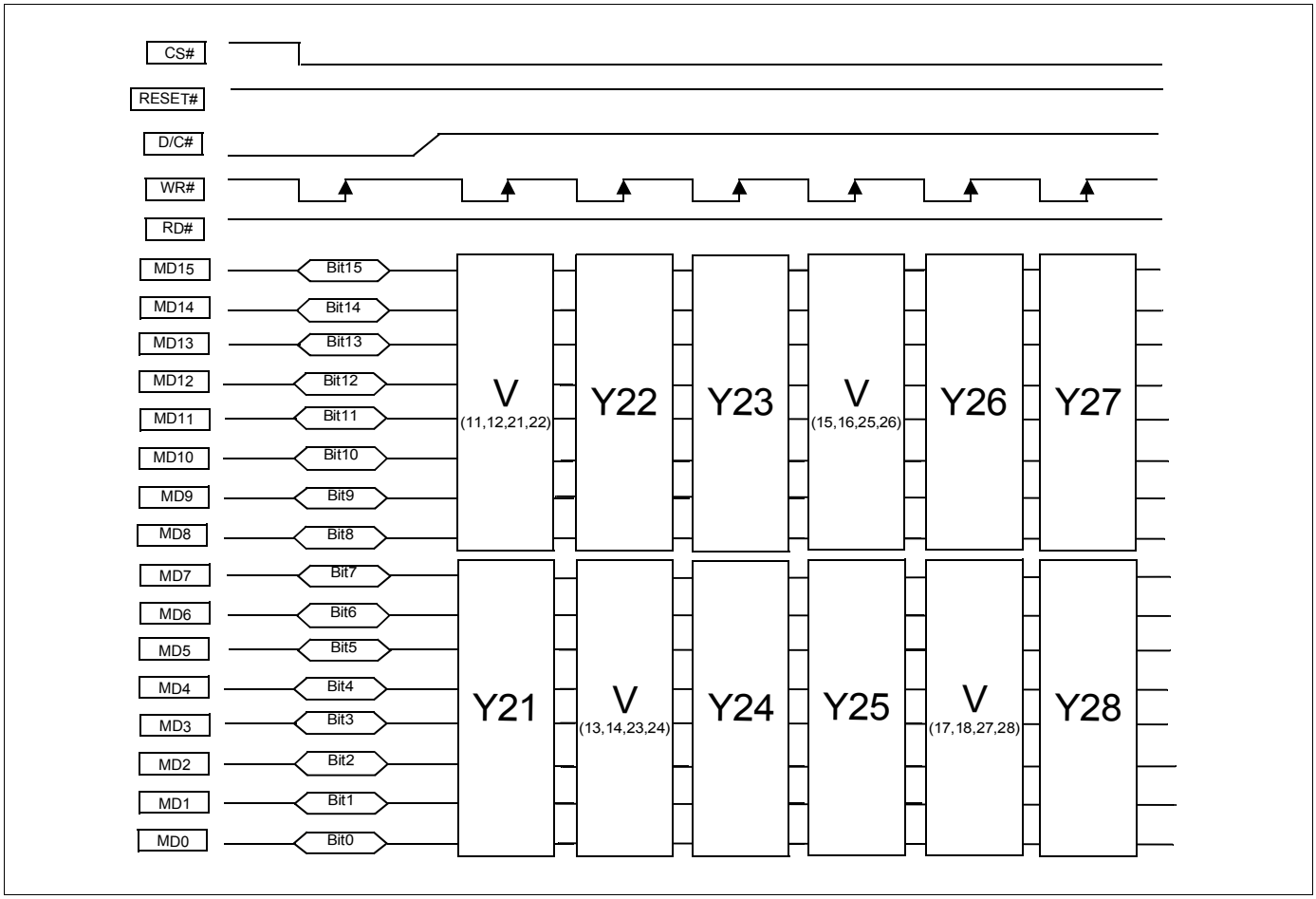


Figure 13-7: YUV 4:2:0 EVEN Line with Intel 80, 16-bit Interface

14 Gamma Correction Look-Up Table Architecture

The following figures are intended to show the display data output path only.

The following diagram shows the architecture for 18 bpp using LUT.

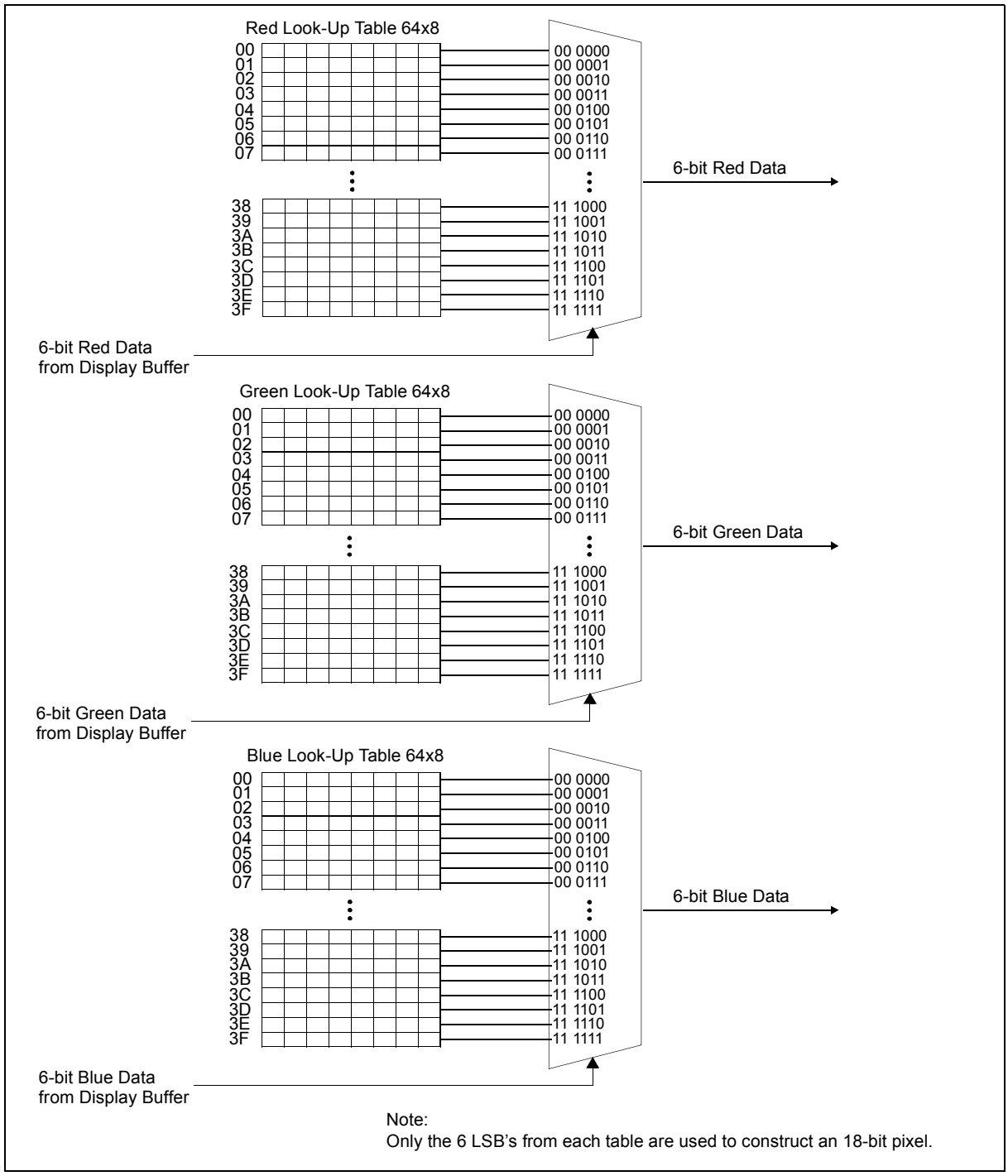


Figure 14-1: Look-Up Table Architecture

14.1 Gamma Correction Example Programming

- Disable the LUT's or ensure you are in a non-display period when accessing to avoid visual anomalies.
- Write register "address" for Gamma Correction Enable Register.
- Write data to set LUT Access Mode.
- Write data to set LUT Index to "x" (auto-increment is already enabled therefore the LUT Index Register address does not have to be written).
- Write data to Gamma Correction Data Register (data value for Index "x").
- Write data to Gamma Correction Data Register (data value for Index "x+1").
- Continue until complete (64 positions). Even in the case of 5:6:5, all 64 positions of each RGB LUT must be programmed when using the auto-increment method.
- Enable Gamma Correction.

15 Display Data Format

Table 15-1: 36-Bit Data Format (Non-Swapped)

	Cycle Count				
	1	2	3	...	n
VD35	R_1^5	R_3^5	R_5^5	...	R_{n+1}^5
VD34	R_1^4	R_3^4	R_5^4	...	R_{n+1}^4
VD33	R_1^3	R_3^3	R_5^3	...	R_{n+1}^3
VD32	R_1^2	R_3^2	R_5^2	...	R_{n+1}^2
VD31	R_1^1	R_3^1	R_5^1	...	R_{n+1}^1
VD30	R_1^0	R_3^0	R_5^0	...	R_{n+1}^0
VD29	G_1^5	G_3^5	G_5^5	...	G_{n+1}^5
VD28	G_1^4	G_3^4	G_5^4	...	G_{n+1}^4
VD27	G_1^3	G_3^3	G_5^3	...	G_{n+1}^3
VD26	G_1^2	G_3^2	G_5^2	...	G_{n+1}^2
VD25	G_1^1	G_3^1	G_5^1	...	G_{n+1}^1
VD24	G_1^0	G_3^0	G_5^0	...	G_{n+1}^0
VD23	B_1^5	B_3^5	B_5^5	...	B_{n+1}^5
VD22	B_1^4	B_3^4	B_5^4	...	B_{n+1}^4
VD21	B_1^3	B_3^3	B_5^3	...	B_{n+1}^3
VD20	B_1^2	B_3^2	B_5^2	...	B_{n+1}^2
VD19	B_1^1	B_3^1	B_5^1	...	B_{n+1}^1
VD18	B_1^0	B_3^0	B_5^0	...	B_{n+1}^0
VD17	R_0^5	R_2^5	R_4^5	...	R_n^5
VD16	R_0^4	R_2^4	R_4^4	...	R_n^4
VD15	R_0^3	R_2^3	R_4^3	...	R_n^3
VD14	R_0^2	R_2^2	R_4^2	...	R_n^2
VD13	R_0^1	R_2^1	R_4^1	...	R_n^1
VD12	R_0^0	R_2^0	R_4^0	...	R_n^0
VD11	G_0^5	G_2^5	G_4^5	...	G_n^5
VD10	G_0^4	G_2^4	G_4^4	...	G_n^4
VD9	G_0^3	G_2^3	G_4^3	...	G_n^3
VD8	G_0^2	G_2^2	G_4^2	...	G_n^2
VD7	G_0^1	G_2^1	G_4^1	...	G_n^1
VD6	G_0^0	G_2^0	G_4^0	...	G_n^0
VD5	B_0^5	B_2^5	B_4^5	...	B_n^5
VD4	B_0^4	B_2^4	B_4^4	...	B_n^4
VD3	B_0^3	B_2^3	B_4^3	...	B_n^3
VD2	B_0^2	B_2^2	B_4^2	...	B_n^2
VD1	B_0^1	B_2^1	B_4^1	...	B_n^1
VD0	B_0^0	B_2^0	B_4^0	...	B_n^0

Table 15-2: 36-Bit Data Format (Swapped)

	Cycle Count				
	1	**2**	**3**	**...**	**n**
VD35	B_0^0	B_2^0	B_4^0	...	B_n^0
VD34	B_0^1	B_2^1	B_4^1	...	B_n^1
VD33	B_0^2	B_2^2	B_4^2	...	B_n^2
VD32	B_0^3	B_2^3	B_4^3	...	B_n^3
VD31	B_0^4	B_2^4	B_4^4	...	B_n^4
VD30	B_0^5	B_2^5	B_4^5	...	B_n^5
VD29	G_0^0	G_2^0	G_4^0	...	G_n^0
VD28	G_0^1	G_2^1	G_4^1	...	G_n^1
VD27	G_0^2	G_2^2	G_4^2	...	G_n^2
VD26	G_0^3	G_2^3	G_4^3	...	G_n^3
VD25	G_0^4	G_2^4	G_4^4	...	G_n^4
VD24	G_0^5	G_2^5	G_4^5	...	G_n^5
VD23	R_0^0	R_2^0	R_4^0	...	R_n^0
VD22	R_0^1	R_2^1	R_4^1	...	R_n^1
VD21	R_0^2	R_2^2	R_4^2	...	R_n^2
VD20	R_0^3	R_2^3	R_4^3	...	R_n^3
VD19	R_0^4	R_2^4	R_4^4	...	R_n^4
VD18	R_0^5	R_2^5	R_4^5	...	R_n^5
VD17	B_1^0	B_3^0	B_5^0	...	B_{n+1}^0
VD16	B_1^1	B_3^1	B_5^1	...	B_{n+1}^1
VD15	B_1^2	B_3^2	B_5^2	...	B_{n+1}^2
VD14	B_1^3	B_3^3	B_5^3	...	B_{n+1}^3
VD13	B_1^4	B_3^4	B_5^4	...	B_{n+1}^4
VD12	B_1^5	B_3^5	B_5^5	...	B_{n+1}^5
VD11	G_1^0	G_3^0	G_5^0	...	G_{n+1}^0
VD10	G_1^1	G_3^1	G_5^1	...	G_{n+1}^1
VD9	G_1^2	G_3^2	G_5^2	...	G_{n+1}^2
VD8	G_1^3	G_3^3	G_5^3	...	G_{n+1}^3
VD7	G_1^4	G_3^4	G_5^4	...	G_{n+1}^4
VD6	G_1^5	G_3^5	G_5^5	...	G_{n+1}^5
VD5	R_1^0	R_3^0	R_5^0	...	R_{n+1}^0
VD4	R_1^1	R_3^1	R_5^1	...	R_{n+1}^1
VD3	R_1^2	R_3^2	R_5^2	...	R_{n+1}^2
VD2	R_1^3	R_3^3	R_5^3	...	R_{n+1}^3
VD1	R_1^4	R_3^4	R_5^4	...	R_{n+1}^4
VD0	R_1^5	R_3^5	R_5^5	...	R_{n+1}^5

Table 15-3: 18-Bit Data Format (Non-Swapped)

	Cycle Count				
	1	**2**	**3**	**...**	**n**
VD[35:18]	Driven Low				
VD17	R_0^5	R_1^5	R_2^5	...	R_n^5
VD16	R_0^4	R_1^4	R_2^4	...	R_n^4
VD15	R_0^3	R_1^3	R_2^3	...	R_n^3
VD14	R_0^2	R_1^2	R_2^2	...	R_n^2
VD13	R_0^1	R_1^1	R_2^1	...	R_n^1
VD12	R_0^0	R_1^0	R_2^0	...	R_n^0
VD11	G_0^5	G_1^5	G_2^5	...	G_n^5
VD10	G_0^4	G_1^4	G_2^4	...	G_n^4
VD9	G_0^3	G_1^3	G_2^3	...	G_n^3
VD8	G_0^2	G_1^2	G_2^2	...	G_n^2
VD7	G_0^1	G_1^1	G_2^1	...	G_n^1
VD6	G_0^0	G_1^0	G_2^0	...	G_n^0
VD5	B_0^5	B_1^5	B_2^5	...	B_n^5
VD4	B_0^4	B_1^4	B_2^4	...	B_n^4
VD3	B_0^3	B_1^3	B_2^3	...	B_n^3
VD2	B_0^2	B_1^2	B_2^2	...	B_n^2
VD1	B_0^1	B_1^1	B_2^1	...	B_n^1
VD0	B_0^0	B_1^0	B_2^0	...	B_n^0

Table 15-4: 18-Bit Data Format (Swapped)

	Cycle Count				
	1	**2**	**3**	**...**	**n**
VD[35:18]	Driven Low				
VD17	B_0^0	B_1^0	B_2^0	...	B_n^0
VD16	B_0^1	B_1^1	B_2^1	...	B_n^1
VD15	B_0^2	B_1^2	B_2^2	...	B_n^2
VD14	B_0^3	B_1^3	B_2^3	...	B_n^3
VD13	B_0^4	B_1^4	B_2^4	...	B_n^4
VD12	B_0^5	B_1^5	B_2^5	...	B_n^5
VD11	G_0^0	G_1^0	G_2^0	...	G_n^0
VD10	G_0^1	G_1^1	G_2^1	...	G_n^1
VD9	G_0^2	G_1^2	G_2^2	...	G_n^2
VD8	G_0^3	G_1^3	G_2^3	...	G_n^3
VD7	G_0^4	G_1^4	G_2^4	...	G_n^4
VD6	G_0^5	G_1^5	G_2^5	...	G_n^5
VD5	R_0^0	R_1^0	R_2^0	...	R_n^0
VD4	R_0^1	R_1^1	R_2^1	...	R_n^1
VD3	R_0^2	R_1^2	R_2^2	...	R_n^2
VD2	R_0^3	R_1^3	R_2^3	...	R_n^3
VD1	R_0^4	R_1^4	R_2^4	...	R_n^4
VD0	R_0^5	R_1^5	R_2^5	...	R_n^5

16 SwivelView™

16.1 Concept

Most computer displays are refreshed in landscape orientation – from left to right and top to bottom. Computer images are stored in the same manner. SwivelView™ is designed to rotate the displayed image on an LCD by 90°, 180°, or 270° in a counter-clockwise direction. The rotation is done in hardware and is transparent to the user for all display buffer writes. By processing the rotation in hardware, SwivelView™ offers a performance advantage over software rotation of the displayed image.

The actual address translation is performed during the Host Write and is therefore stored in memory as rotated. Because of where the rotation logic is, each Window written to the S1D13742 can be independently rotated with respect to each other.

16.2 90° SwivelView™

The following figure shows how the programmer sees a 320x480 portrait image and how the image is being displayed. The application image is written to the S1D13742 in the following sense: A–B–C–D. The display is refreshed in the following sense: B-D-A-C.

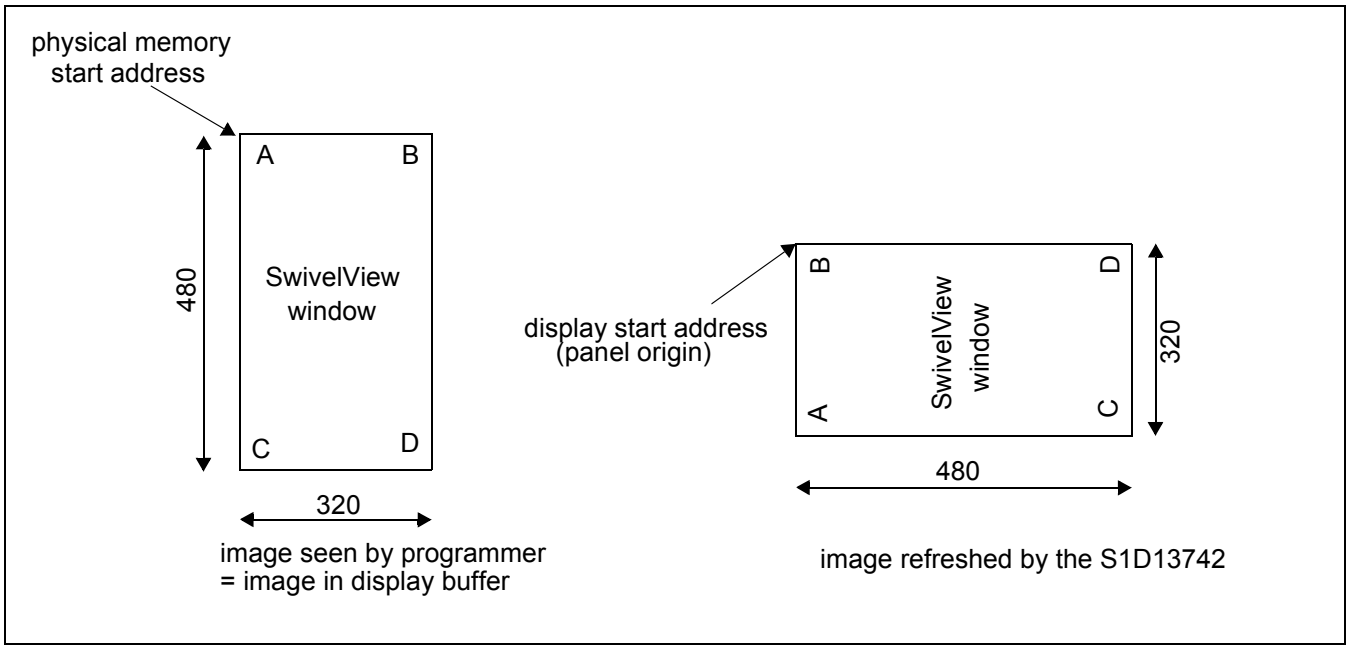


Figure 16-1: Relationship Between The Screen Image and the Image Refreshed in 90° SwivelView.

16.2.1 Register Programming

There is no special programming requirements other than simply enabling the rotation itself. All start addresses and Line Offset's are automatically calculated by hardware.

16.3 180° SwivelView™

The following figure shows how the programmer sees a 480x320 landscape image and how the image is being displayed. The application image is written to the S1D13742 in the following sense: A–B–C–D. The display is refreshed in the following sense: D-C-B-A.

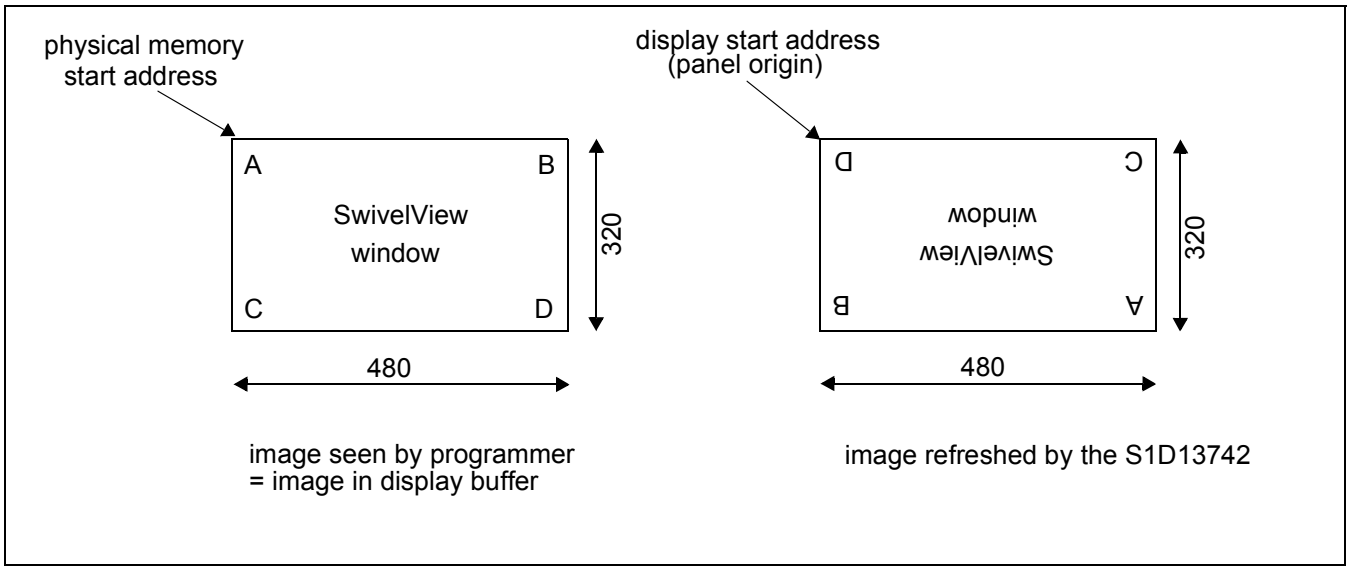


Figure 16-2: Relationship Between The Screen Image and the Image Refreshed in 180° SwivelView.

16.3.1 Register Programming

There is no special programming requirements other than simply enabling the rotation itself. All start addresses and Line Offset's are automatically calculated by hardware.

16.4 270° SwivelView™

The following figure shows how the programmer sees a 320x480 portrait image and how the image is being displayed. The application image is written to the S1D13742 in the following sense: A–B–C–D. The display is refreshed in the following sense: C-A-D-B.

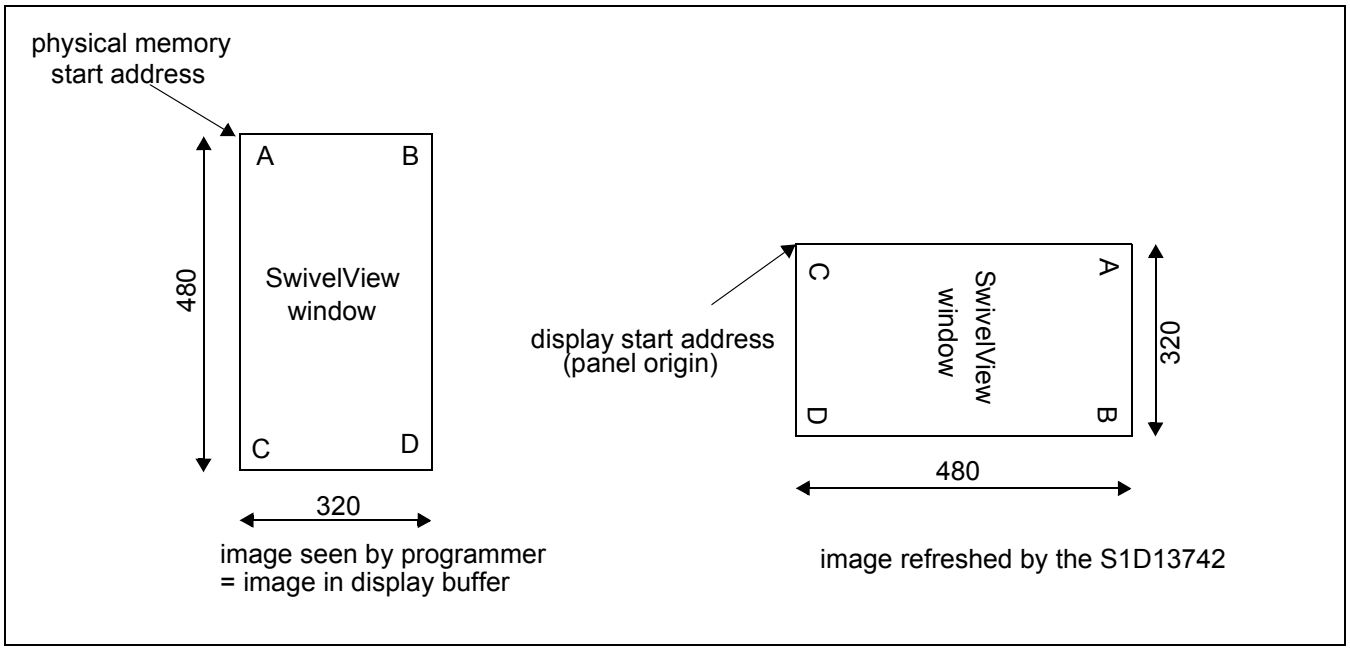


Figure 16-3: Relationship Between The Screen Image and the Image Refreshed in 270° SwivelView.

16.4.1 Register Programming

There is no special programming requirements other than simply enabling the rotation itself. All start addresses and Line Offset's are automatically calculated by hardware.

17 Host Interface

17.1 Using the Intel 80 Interface

Accessing the S1D13742 through the Intel 80 interface is a multiple step process. All Registers and Memory are accessed through register space.

Note

All Register accesses, except the Memory Data Port, are 8-bit only. If the Host interface is 16-bits wide, the lsb's (MD[7:0]) are used for all registers except the Memory Data Port.
The Memory Data Port (REG[48h, 49h]) is handled as 8-bit if CNF1 = 0 (REG[49h] not used) or 16-bit if CNF1 =1.

First, perform a single "Address Write" to setup the register address. Next a "Data Read/Write" is performed that specifies the data to be stored or read from the registers or memory specified in the "Address Write" cycle. Subsequent data Read/Writes without a Address Write to change the register address, will automatically "auto" increment the register address or the internal memory address if accessing the Memory Data Port.

To write display data to a Window Aperture, simply set-up the Window coordinates followed by the burst data writes to the Memory Data Port to fill the window. In this sequence, the internal memory addressing is automatic (see examples). The Memory Data Port is located directly following the Window coordinates to minimize the number of Address Writes.

To Read display data, perform an Address Write to the Memory Address Port (3 bytes) and then read data from the Memory Data Port. Sequential reads will auto-increment the internal memory address

17.1.1 Register write procedure

1. Perform address write to setup register address bits 7-0.
2. Perform data write to update the register.
3. Additional data writes are supported. In this case, the register addresses will be auto-incremented.

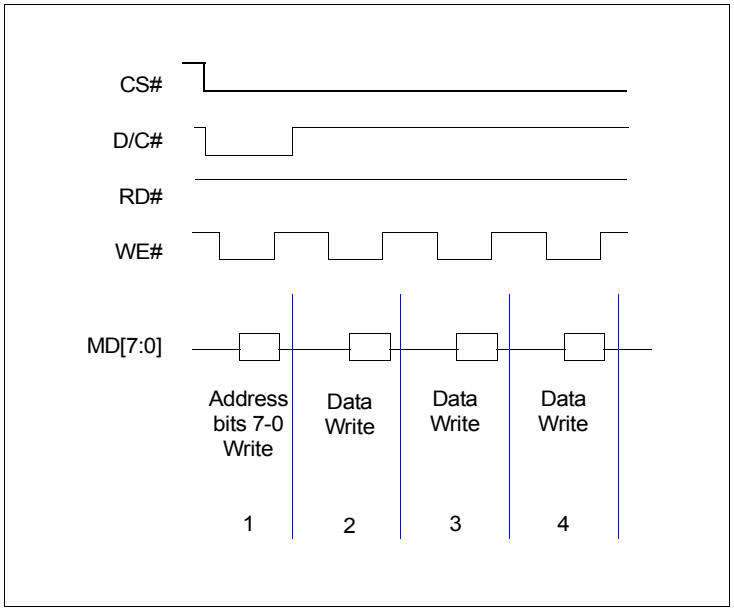


Figure 17-1: Register Write Example Sequence

17.1.2 Register read procedure

1. Perform address write to setup register address bits 7-0.
2. Perform data read to get the register value.
3. Additional data reads are supported. In this case, the register addresses will be auto-incremented.

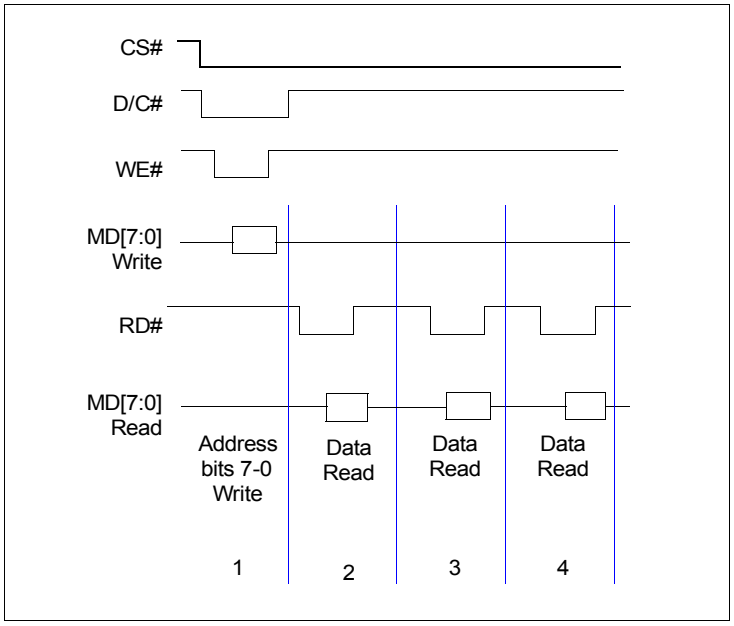


Figure 17-2: Register Read Example Sequence

17.1.3 New Window Aperture Write procedure

The S1D13742 has a special procedure to minimize set-up accesses when bursting window data.

1. The panel dimension registers must be set before writing any Window data.
2. Perform an Address Write to point to the first Window Register (Window X Start Position).
3. Perform eight "data" writes to the next eight, 8-bit registers (this will set-up all the Window coordinates.

Note
In this case the register addresses will be auto-incremented until you reach the Memory Data Port Register

4. Perform burst data writes to fill the window (the register address will already be pointing at the Memory Data Port)

The Memory Data Port Register is located in the 9th register address after the Window X Start Position. Every write to the Memory Data Port will auto-increment the internal memory address only.

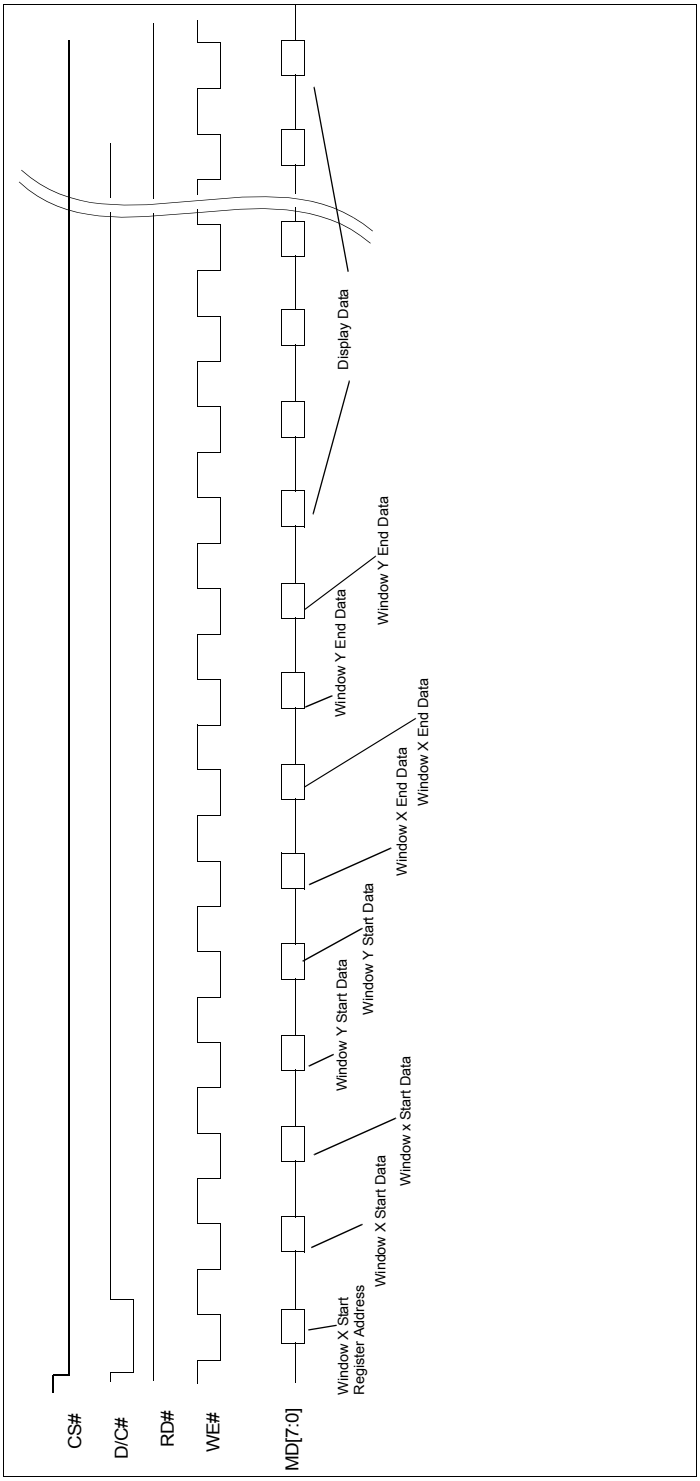


Figure 17-3: Sequential Memory Write Example Sequence

17.1.4 Opening Multiple Windows

1. Repeat steps above (New Window Aperture write procedure) with new window coordinates for each new window.
2. Non-pixel doubled windows can overlap with the last one being written considered the top.

Update Window using existing Window Coordinates:

1. Perform an Address Write to point to the Memory Data Port
2. Perform burst data writes to fill the window.

Note
In this case the previous coordinates of the Window Aperture will be used. Every write to the Memory Data Port will auto-increment the internal memory address only.

17.1.5 Individual Memory Location Reads

Note
This function is for test purposes only and serves no practical use in a system.

1. Set the Memory Data Format to 16bpp.
2. Write the physical address of the memory location to read from, to the Memory Read Address Registers (for a 16bit bus, the LSB of this address is ignored).
3. Perform a read from the Memory Data Port Register.
4. Continuous reads from the Memory Data Port Register will cause the address in the Memory Read Address Registers to increment, thereby supporting burst reads.

Note
To access the 2 msb's for each 18-bit value, you must know the physical address as they are stored at different locations as compared to the lower 16-bits.

18 Double Buffer Description

18.1 Double Buffer Controller

Double buffering is provided to prevent tearing of streaming video data. All static (non-video) image data will always be written to the upper half (Buffer 1) of the frame buffer. When video is being input, the first frame will be written to the lower half (Buffer 2) of the double buffer. The second frame will be written to Buffer 1. While video data is being input, the static part of the image going to the LCD will still always come from Buffer 1. The source of the video window will come from either Buffer 1 or Buffer 2, depending on which one was the last to be completely updated.

The switching of the buffer read/write pointers can only occur once per frame, at the beginning of the vertical non-display period. The pointers will only switch if: a video frame had completed being updated within the last output frame period, and no new video frame is currently being written. Because of this, each time the user finishes writing a frame of video data, they should wait until the next vertical non-display period before writing the next frame. This can be accomplished by using the TE pin or by polling the Vertical Display Period Status (REG[58h] bit 7). Alternatively, if the user can guarantee that the maximum input video frame rate is 1/2 the LCD frame rate and that the burst length for writing a video frame is less than one LCD frame period, then no checking for the vertical non-display period is required. If attention is not paid to allowing the pointers to switch, then frames may be dropped.

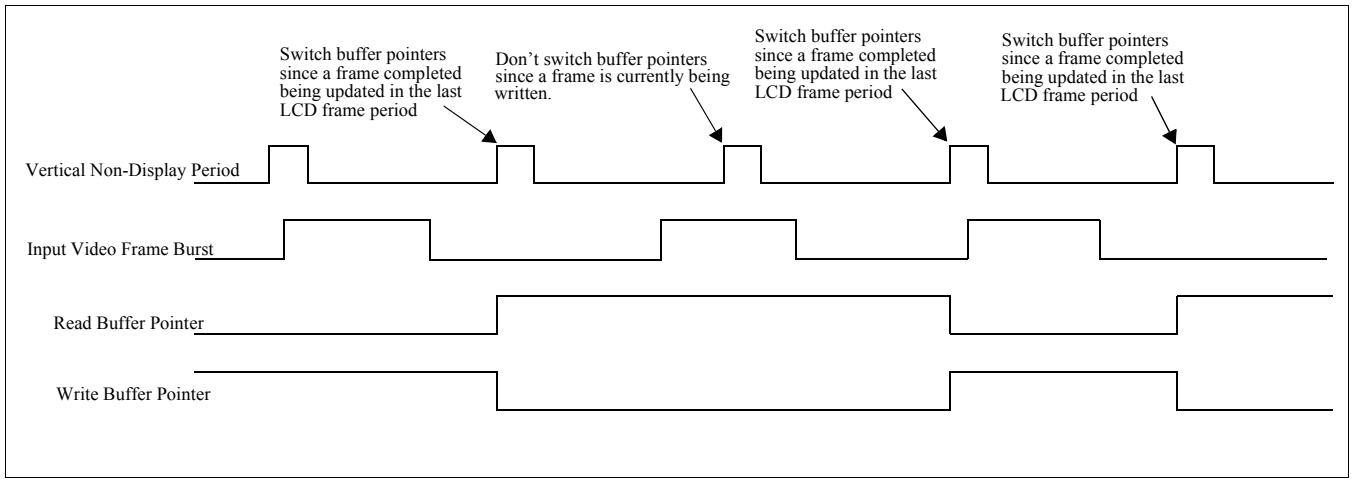


Figure 18-1: Switching of Buffer Pointers

To use the double buffer feature:

- Set the Special Effects Register REG[36h] bits 7-6 to 11.
- Setup the Window Position Registers REG[38h] - REG[46h].
- Write the video data to the Memory Data Port REG[48h] - REG[49h].

It is also possible to update a static window while double buffering is enabled, even in the middle of a video stream. To do this:

- Write the last pixel of the current frame of video data.
- Set the Special Effects Register REG[36h] bits 7-6 to 01.
- Setup the Window Position Registers REG[38h] - REG[46h].
- Write the static data to the Memory Data Port REG[48h] - REG[49h].

This allows a static image to be written at any time, while still preventing the double buffered window from tearing. Once the static window has been written, the user can go back to writing the streaming video data by following the steps described above for using the double buffer feature.

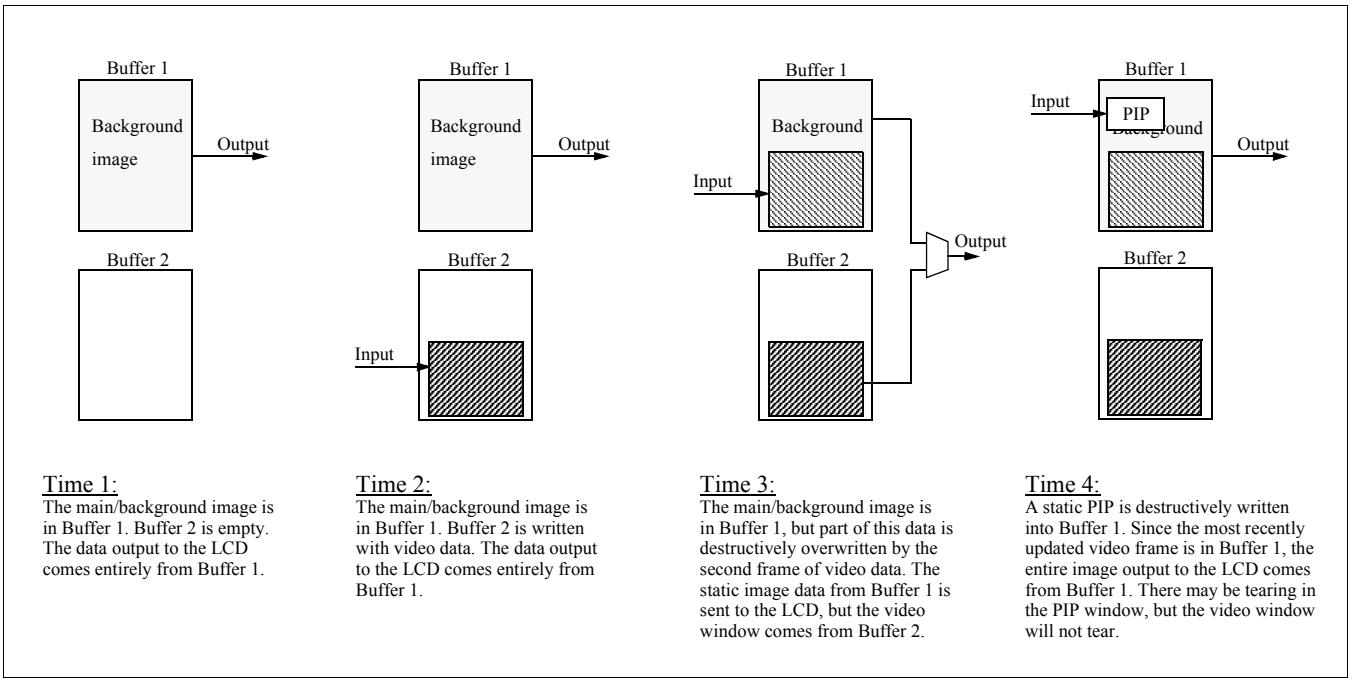


Figure 18-2: Double Buffer Example

There are some limitations to double buffering:

- Consider the case where there is a video stream being input and the user wants to place a static PIP over all or some part of the video window. The user can write the PIP, but when the video stream is continued, it will destructively overwrite the PIP, so that it will appear as though the PIP is under the video window.
- Consider the case where there is a video stream which stops after the last frame of video is sent. The final frame of video will continue to be displayed on the LCD. Assume that this last frame is stored in Buffer 2. Now, if the user disables double buffering, the buffer read pointer will immediately reset to Buffer 1. This means that the 2nd to last frame will now be displayed instead of the last frame.
- The user must either wait for a vertical non-display period between writing frames of video data, or guarantee that their maximum input frame rate is 1/2 the LCD frame rate and that the length of time it takes to burst write a frame of video data is less than one LCD frame period.
- Only one window can be double buffered at a time.

19 Interfacing the S1D13742 and a TFT Panel

This section describes the hardware and software environment required to interface the S1D13742 Mobile Graphics Engine and a 352x416 or 800x480 TFT Panel.

The designs described in this section are presented only as examples of how such interfaces might be implemented.

19.1 Overview

The S1D13742 was designed to directly support the Sanyo LC13015 and requires no additional hardware and minimal programming. The S1D13742 register settings and electrical interface is described below.

19.1.1 Electrical Interface

Table 19-1: Pin Mapping

S1D13742 Pin Name	S1D13742 Pin Number	LCD13015 Pin Name
HS	D9	HS
VS	D10	VS
PCLK	D11	PCLK
DE	C11	DE
VD[17:0]	J8,J9,J10,J11,K4,K5,K6,K7,K8,K9,K10,L3,L4,L5,L6,L7,L8,L9	R5,R4,R3,R2,R1,R0,G5,G4,G3,G2,G1,G0,B5,B4,B3,B2,B1,B0

19.1.2 S1D13742 Register Settings for 352x416 TFT Panel

Note
The registers listed below are only those associated with panel specific timing issues All other registers are not shown here.

Note
When a window is setup for YUV data, the data must always alternate between odd and even lines, starting with an odd line.

Table 19-2: Example Register Settings for 352x416 TFT Panel

Register	Value	Comment
All	default	Come out of reset - all registers set to default values
REG[56h]	02h	enter sleep mode (or use PWRSVE pin)
REG[04h]	12h	set PLL M-Divider. CLKI = 19.2MHz, PLL input clock = CLKI/19 = 1.01MHz.
REG[06h]	F8h	
REG[08h]	80h	
REG[0Ah]	28h	
REG[0Ch]	00h	
REG[0Eh]	2Fh	LL = 48, resulting SYSCLK = LL x PLL input clock = 48MHz
REG[12h]	19h	set PCLK divide, PCLK = 12.1MHz set SYSCLK source = PLL
REG[14h]	0h	no panel data swap, 18-bit panel
REG[16h]	2Ch	HDP = 352 pixels
REG[18h]	5Ah	HNDP = 90 pixels
REG[1Ah]	A0h	VDP = 416 lines
REG[1Ch]	01h	
REG[1Eh]	06h	VNDP = 6 lines
REG[20h]	14h	HS Pulse Width = 20 pixels
REG[22h]	2Dh	HS Start Position = 45 pixels
REG[24h]	02h	VS Width = 2 lines
REG[26h]	01h	VS Start Position (VFP) = 1 line
REG[28h]	80h	PCLK Polarity: data output on falling edge
REG[2Ah]	01h	set memory to 16 bpp, set input data mode to RGB 5:6:5
REG[56h]	00h	disable sleep mode
REG[04h] bit 7	—	wait for PLL to lock - poll REG[04h] bit 7
REG[38h]	00h	Window X Start Position = 0
REG[3Ah]	00h	
REG[3Ch]	00h	Window Y Start Position = 0
REG[3Eh]	00h	

Table 19-2: Example Register Settings for 352x416 TFT Panel (Continued)

Register	Value	Comment
REG[40h]	5Fh	Window X End Position = 351
REG[42h]	01h	
REG[44h]	9Fh	Window Y End Position = 415
REG[46h]	01h	
REG[48h]	Write the image data to the Memory Data Port, REG[48h] and REG[49h]. The image will immediately begin to appear on the LCD.	
REG[49h]		

Note

The above values are intended as examples. This example assumes that CLKI = 19.2MHz and that the PLL is used to generate SYSCLK. Actual settings can vary and still remain within the LCD panel timing requirements.

19.1.3 S1D13742 Register Settings for 800x480 TFT Panel

Note
The registers listed below are only those associated with panel specific timing issues All other registers are not shown here.

Note
When a window is setup for YUV data, the data must always alternate between odd and even lines, starting with an odd line.

Table 19-3: Example Register Settings for 800x480 TFT Panel

Register	Value	Comment
All	default	Come out of reset - all registers set to default values
REG[56h]	02h	enter sleep mode (or use PWRSVE pin)
REG[04h]	0Bh	set PLL M-Divider. CLKI = 12MHz, PLL input clock = CLKI/12 = 1.0MHz.
REG[06h]	F8h	
REG[08h]	80h	
REG[0Ah]	28h	
REG[0Ch]	00h	
REG[0Eh]	2Dh	LL = 45, resulting SYSCLK = LL x PLL input clock = 45MHz
REG[12h]	09h	set PCLK divide, PCLK = 22.5MHz set SYSCLK source = PLL
REG[14h]	0h	no panel data swap, 18-bit panel
REG[16h]	64h	HDP = 800 pixels
REG[18h]	14h	HNDP = 20 pixels
REG[1Ah]	E0h	VDP = 480 lines
REG[1Ch]	01h	
REG[1Eh]	06h	VNDP = 6 lines
REG[20h]	14h	HS Pulse Width = 20 pixels
REG[22h]	2Dh	HS Start Position = 45 pixels
REG[24h]	02h	VS Width = 2 lines
REG[26h]	01h	VS Start Position (VFP) = 1 line
REG[28h]	80h	PCLK Polarity: data output on falling edge
REG[2Ah]	01h	set memory to 16 bpp, set input data mode to RGB 5:6:5
REG[56h]	00h	disable sleep mode
REG[04h] bit 7	—	wait for PLL to lock - poll REG[04h] bit 7
REG[38h]	00h	Window X Start Position = 0
REG[3Ah]	00h	
REG[3Ch]	00h	Window Y Start Position = 0
REG[3Eh]	00h	

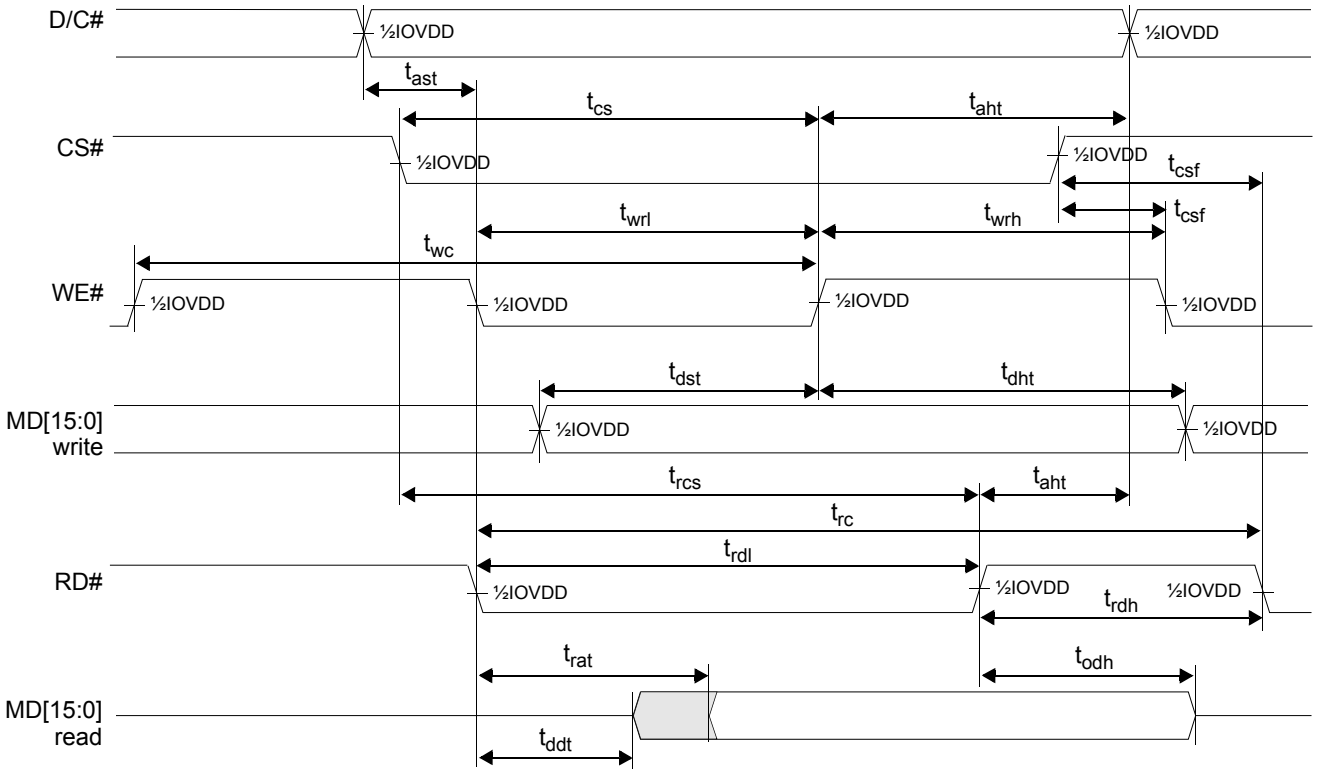
Table 19-3: Example Register Settings for 800x480 TFT Panel (Continued)

Register	Value	Comment
REG[40h]	1Fh	Window X End Position = 799
REG[42h]	03h	
REG[44h]	DFh	Window Y End Position = 479
REG[46h]	01h	
REG[48h]	Write the image data to the Memory Data Port, REG[48h] and REG[49h]. The image will immediately begin to appear on the LCD.	
REG[49h]		

Note

The above values are intended as examples. This example assumes that CLKI = 12MHz and that the PLL is used to generate SYSCLK. Actual settings can vary and still remain within the LCD panel timing requirements.

19.2 Host Bus Timing



Note: The D/C# input pin is used to distinguish between Address and Data.

Note: The register address will auto-increment in word increments for all register access except the Gamma Correction Table Data register and Memory Data Port. Writes to the Gamma Correction Table Data register and Memory Data Port will not increment the register address to support burst data writes to the gamma correction table and to memory.

Figure 19-1: Intel 80 Input A.C. Characteristics

19.2.1 Host Bus Timing for 352x416 TFT Panel

Table 19-4: Intel 80 Input A.C. Characteristics (352x416 Panel Timings)

Signal	Symbol	Parameter	Min	Max	Unit	Description
D/C#	t_{ast}	Address setup time	1.4	—	nsec	
	t_{aht}	Address hold time	0.3	—	nsec	
CS#	t_{cs}	Chip Select setup time (write)	0.6 + twrl	—	nsec	
	t_{rcs}	Chip Select setup time (read)	1.3 + trdl	—	nsec	
	t_{csf}	Chip Select Wait time	9.2	—	nsec	
WE#	t_{wc}	Write cycle (rising edge to next rising edge)	42.6	—	nsec	
	t_{wrh}	Pulse high duration	Note 1	—		
	t_{wrl}	Pulse low duration	0.1	—	nsec	
RD#	t_{rc}	Read cycle for Registers	42.6	—	nsec	
		Read cycle for Memory	122.1 + trdh	—	nsec	
		Read cycle for LUT	108.1 + trdh	—	nsec	
	t_{rdh}	Pulse high duration	Note 2	—		
	t_{rdl}	Pulse low duration for Registers	10.2	—	nsec	
		Pulse low duration for Memory	122.1	—	nsec	
		Pulse low duration for LUT	108.1	—	nsec	
MD[15:0]	t_{dst}	Data setup time	0.3	—	nsec	For maximum CL=30pF For minimum CL=8pF
	t_{dht}	Data hold time	6.4	—	nsec	
	t_{rat} (See note)	Read falling edge to Data valid for Registers	—	12.2	nsec	
		Read falling edge to Data valid for Memory	—	122.1	nsec	
		Read falling edge to Data valid for LUT	—	108.1	nsec	
	t_{odh} (See note)	Read hold time	10.7	32.1	nsec	
	t_{ddt} (See note)	Read falling edge to Data driven	3.0	12.3	nsec	

SYSCLK = 48MHz, PCLK = 12MHz, CLKI = 12MHz

1. t_{wrh} min = long enough to satisfy t_{wc}
2. t_{rdh} min = long enough to satisfy t_{rc}

19.2.2 Host Bus Timing for 800x480 TFT Panel

Table 19-5: Intel 80 Input A.C. Characteristics (800x480 Panel Timings)

Signal	Symbol	Parameter	Min	Max	Unit	Description
D/C#	t_{ast}	Address setup time	1.4	—	nsec	
	t_{aht}	Address hold time	0.3	—	nsec	
CS#	t_{cs}	Chip Select setup time (write)	0.6 + twrl	—	nsec	
	t_{rcs}	Chip Select setup time (read)	1.3 + trdl	—	nsec	
	t_{csf}	Chip Select Wait time	9.2	—	nsec	
WE#	t_{wc}	Write cycle (rising edge to next rising edge)	34.8	—	nsec	
	t_{wrh}	Pulse high duration	Note 1	—		
	t_{wrl}	Pulse low duration	0.1	—	nsec	
RD#	t_{rc}	Read cycle for Registers	34.8	—	nsec	
		Read cycle for Memory	102.7 + trdh	—	nsec	
		Read cycle for LUT	92.5 + trdh	—	nsec	
	t_{rdh}	Pulse high duration	Note 2	—		
	t_{rdl}	Pulse low duration for Registers	10.2	—	nsec	
		Pulse low duration for Memory	102.7	—	nsec	
		Pulse low duration for LUT	92.5	—	nsec	
MD[15:0]	t_{dst}	Data setup time	0.3	—	nsec	For maximum CL=30pF For minimum CL=8pF
	t_{dht}	Data hold time	6.4	—	nsec	
	t_{rat} (See note)	Read falling edge to Data valid for Registers	—	12.2	nsec	
		Read falling edge to Data valid for Memory	—	102.7	nsec	
		Read falling edge to Data valid for LUT	—	92.5	nsec	
	t_{odh} (See note)	Read hold time	10.7	32.1	nsec	
	t_{ddt} (See note)	Read falling edge to Data driven	3.0	12.3	nsec	

SYSCLK = 59 MHz, PCLK = 19.67 MHz, CLKI = 12MHz

1. t_{wrh} min = long enough to satisfy t_{wc}
2. t_{rdh} min = long enough to satisfy t_{rc}

19.3 Panel Timing

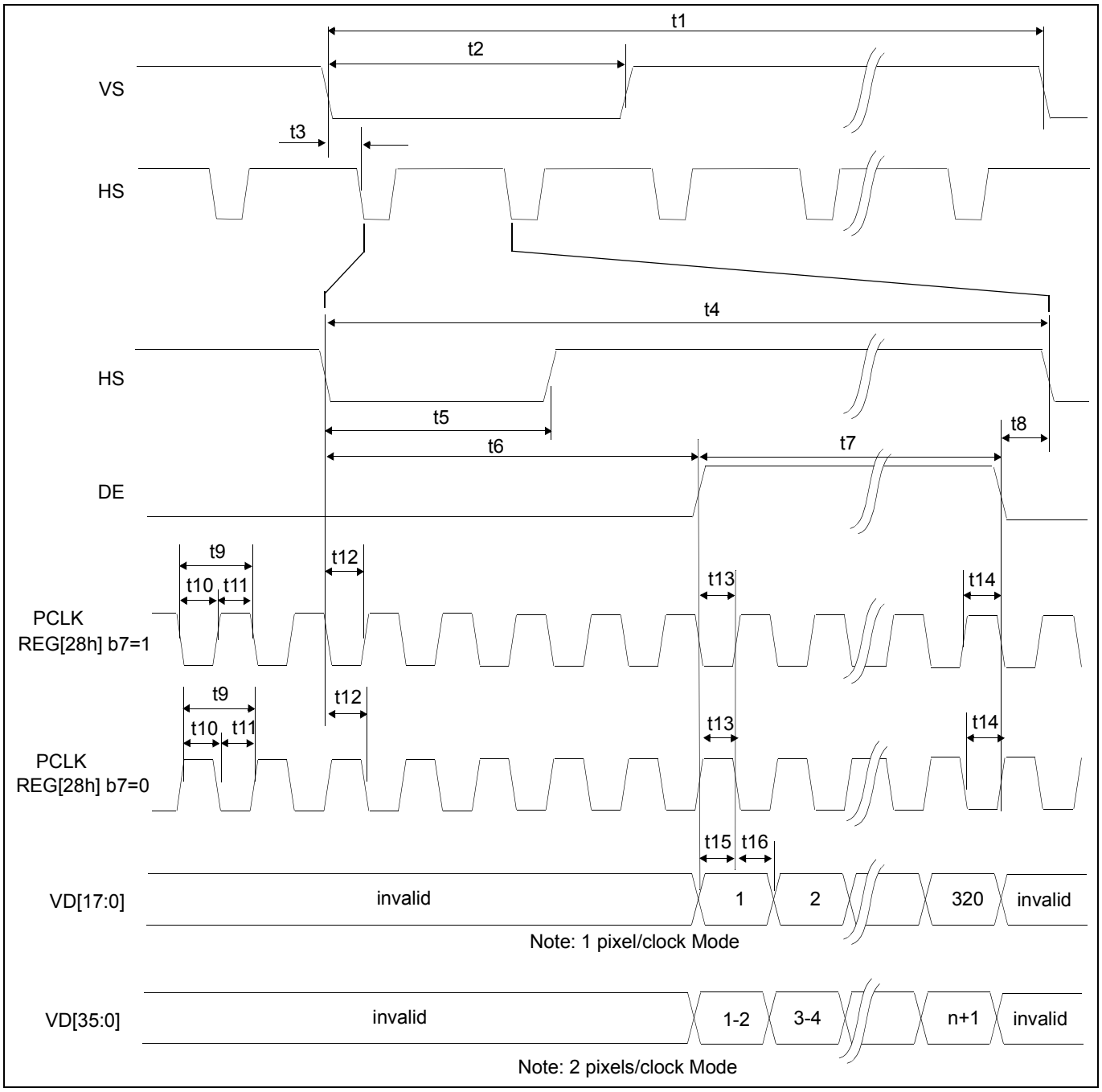


Figure 19-2: 18/36-Bit TFT A.C. Timing

19.3.1 Panel Timing for 352x416 Panel

Table 19-6: 18/36-Bit TFT A.C. Timing (352x416 Panel Timing)

Symbol	Parameter	Min	Typ	Max	Units
t1	VS cycle time		15.54	—	ms
t2	VS pulse width low		73.67	—	us
t3	VS falling edge to HS falling edge phase difference	0	—	36.75	us
t4	HS cycle time		36.83	—	us
t5	HS pulse width low		1.67	—	us
t6	HS Falling edge to DE active		3.75	—	us
t7	DE pulse width		29.3	—	us
t8	DE falling edge to HS falling edge		3.75	—	us
t9	PCLK period	83.3	—	—	ns
t10	PCLK pulse width low	41.7	—	—	ns
t11	PCLK pulse width high	41.7	—	—	ns
t12	HS setup to PCLK falling edge	41.7	—	—	ns
t13	DE to PCLK rising edge setup time	41.7	—	—	ns
t14	DE hold from PCLK rising edge	41.7	—	—	ns
t15	Data setup to PCLK rising edge	41.7	—	—	ns
t16	Data hold from PCLK rising edge	41.7	—	—	ns

1. Ts = pixel clock period = 83.3 ns (12MHz PCLK)

19.3.2 Panel Timing for 800x480 Panel

Table 19-3 18/36-Bit TFT A.C. Timing (800x480 Panel Timings)

Symbol	Parameter	Min	Typ	Max	Units
t1	VS cycle time		20.34	—	ms
t2	VS pulse width low		83.4	—	us
t3	VS falling edge to HS falling edge phase difference	0	—	41.63	us
t4	HS cycle time		41.68	—	us
t5	HS pulse width low		1.02	—	us
t6	HS Falling edge to DE active		966	—	ns
t7	DE pulse width		40.67	—	us
t8	DE falling edge to HS falling edge		50.84	—	ns
t9	PCLK period	50.84	—	—	ns
t10	PCLK pulse width low	25.42	—	—	ns
t11	PCLK pulse width high	25.42	—	—	ns
t12	HS setup to PCLK falling edge	25.42	—	—	ns
t13	DE to PCLK rising edge setup time	25.42	—	—	ns
t14	DE hold from PCLK rising edge	25.42	—	—	ns
t15	Data setup to PCLK rising edge	25.42	—	—	ns
t16	Data hold from PCLK rising edge	25.42	—	—	ns

1. Ts = pixel clock period = 50.84 (19.67 PCLK)

19.4 Example Play.exe Scripts

The following example scripts are written for the PLAY.EXE program. The script Demo.txt will initialize the S1D13742, then display horizontal bars at different rotations, and then display a PIP+ window.

Demo.txt

```
verbose cmd:off out:on set:off
halt 0

'================================================================================
' _DEMO_.txt - Play script for 13742 to demonstrate various features.
'
' This demonstration code is written in the Play.exe script language so that
' various steps can be easily observed. Some steps such as the initialization
' and the memory fills use Play intrinsic commands. These operation of these
' commands are easily determined.
'================================================================================


' Initialize the registers to the default state by
' running the register list generated by 13742CFG
'-------------------------------------------------------
init

' Set the window to the full screen and clear the display
'-------------------------------------------------------
SetWin.txt
f WIN 0


' ROTATE 0
'-------------------------------------------------------
print "Color bars at SwivelView 0\n"
x 34 0
DrawBarsA.txt
Pause.txt


' ROTATE 90
' NOTE:  There is a bug with the Fill WINdow command in
'        Play which causes the 90 and 270 degree fills
'        to be filled incorrectly. This will be corrected.
'-------------------------------------------------------
print "Color bars at SwivelView 90\n"
x 34 1
DrawBarsB.txt
Pause.txt


' ROTATE 180
'-------------------------------------------------------
print "Color bars at SwivelView 180\n"
x 34 2
```

```
DrawBarsA.txt
Pause.txt


' ROTATE 270
' NOTE:  There is a bug with the Fill WINdow command in
'        Play which causes the 90 and 270 degree fills
'        to be filled incorrectly. This will be corrected.
'-------------------------------------------------------
print "Color bars at SwivelView 270\n"
x 34 3
DrawBarsB.txt
Pause.txt


' PIP
'-------------------------------------------------------
print "Draw Color bars in a PIP (small window)\n"

x 34 0
SetWin.txt
f WIN 0

DrawBarsA.txt
DrawPIP.txt 50 50 100 128
Pause.txt


section END
```

DrawBarsA.txt

```
verbose cmd:off out:on set:off

'================================================================================
' DrawBars.txt - Play script for the 13742
'
' This script draws eight equally sized horizontal
' bars on the display.
'================================================================================

set $Height ((reg[1C] << 8) + (reg[1A]))
set $Lines  ($Height / 8)
set $StartX 0
set $StartY 0
set $EndX    width
set $EndY   $Lines

set $Color  0
set $Bars   8

section LOOP

SetWin.txt $StartX $StartY $EndX $EndY

f WIN $Color

set $StartY ($StartY + $Lines)
set $EndY   ($EndY   + $Lines)
set $Color  ($Color + 0821)

set $Bars ($Bars - 1)
if $Bars!=0 then goto LOOP
```

DrawBarsB.txt

```
verbose cmd:off out:on set:off

'================================================================================
' DrawBarsB.txt - Play script for the 13742
'
' This script draws horizontal bars in SwivelView 90 and SwivelView 270
' display modes.
'================================================================================

set $Height (reg[16] * 8)
set $Lines  ($Height / 8)
set $StartX 0
set $StartY 0
set $EndX    height
set $EndY   $Lines

set $Color  0
set $Bars   8

section LOOP

SetWin.txt $StartX $StartY $EndX $EndY

f WIN $Color

set $StartY ($StartY + $Lines)
set $EndY   ($EndY   + $Lines)
set $Color  ($Color + 0821)

set $Bars ($Bars - 1)
if $Bars!=0 then goto LOOP
```

DrawPIP.txt

```
verbose cmd:off out:on set:off

'================================================================================
' DrawPIP.txt - Play script for the 13742
'
' This script draws eight equally sized horizontal bars on the display.
'================================================================================

set $StartX arg[1].nt
set $StartY arg[2].nt
set $Width   arg[3].nt
set $Height arg[4].nt

set $Lines ($Height / 8)

set $Color  0
set $Bars   8


section LOOP

SetWin.txt $StartX  $StartY  $Width  $Lines

f WIN $Color

set $StartY ($StartY + $Lines)
set $Color  ($Color + 0821)

set $Bars ($Bars - 1)
if $Bars!=0 then goto LOOP
```

Pause.txt

```
verbose cmd:off out:on set:off
halt 0

print "Paused . . . press any key to continue\n"
input line
```

SetWin.txt

```
verbose cmd:off out:on set:off

'-------------------------------------------------------------------------------
' SetWin.txt - Play script for the 13742
'
' This script is functionally identical to the Play command 'win'. Call this
' script to set the 13742 window co-ordinates as specified by the arguments.
'
'   Syntax:  SetWin X Y W H
'   Where:   X  - Left edge window X position
'            Y  - Top edge window Y position
'            W  - Window width
'            H  - Window height
'
'   Example: SetWin 0 0 100 100
'            Sets the window to start at 0,0 and end at 100, 100
'
'            SetWin
'            Sets the window size to the size of the display
'
'   win SX:0 SY:0 EX:width EY:height
'-------------------------------------------------------------------------------

' Set the default window values to the display size.
set $SX 0
set $SY 0
set $EX (width  - 1)
SET $EY (height - 1)

' Use non-default values ONLY if all four arguments are given
if (argn!=5) then goto SETWINDOW

set $SX arg[1].n
set $SY arg[2].n
set $EX (arg[1].n + arg[3].n - 1)
set $EY (arg[2].n + arg[4].n - 1)

section SETWINDOW

' Change the register window settings

x 38  $SX
x 3A ($SX >> 8)

x 3C  $SY
x 3E ($SY >> 8)

x 40  $EX
x 42 ($EX >> 8)

x 44  $EY
x 46 ($EY >> 8)
```

19.5 References

19.5.1 Documents

- Sanyo Electric Co., Ltd. Display Company*, LC13015 Low Temperature P-Si TFT-LCD Specification*, Document Number *LC13015-040302*
- Epson Research and Development, Inc., *S1D13742 Hardware Functional Specification*, Document Number X63A-A-001-xx.

20 PLL Power Supply Considerations

The PLL circuit is an analog circuit which is very sensitive to noise on the input clock waveform or the power supply. Noise on the clock or the supplied power may cause the operation of the PLL circuit to become unstable or increase the jitter.

Due to these noise constraints, it is highly recommended that the power supply traces or the power plane for the PLL be isolated from those of other power supplies. Filtering should also be used to keep the power as clean as possible.

The following are guidelines which, if followed, will result in cleaner power to the PLL, this will result in a cleaner and more stable clock. Even a partial implementation of these guidelines will give results.

20.1 Guidelines for PLL Power Layout

The PLL circuit is an analog circuit and is very sensitive to noise on the input clock waveform or the power supply. Noise on the clock or the supplied power may cause the operation of the PLL circuit to become unstable or increase the jitter.

Due to these noise constraints, it is highly recommended that the power supply traces or the power plane for the PLL be isolated from those of other power supplies. Filtering should also be used to keep the power as clean as possible.

The following are guidelines which, if followed, will result in cleaner power to the PLL, resulting in a cleaner and more stable clock. Even a partial implementation of these guidelines will give results.

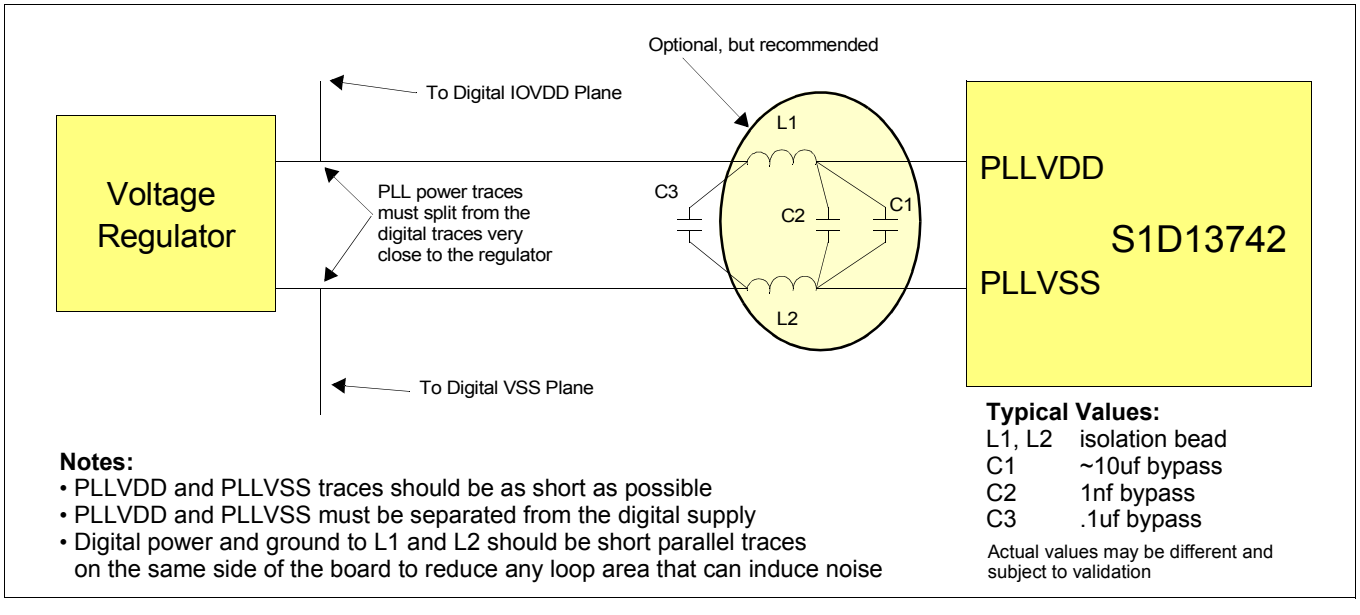
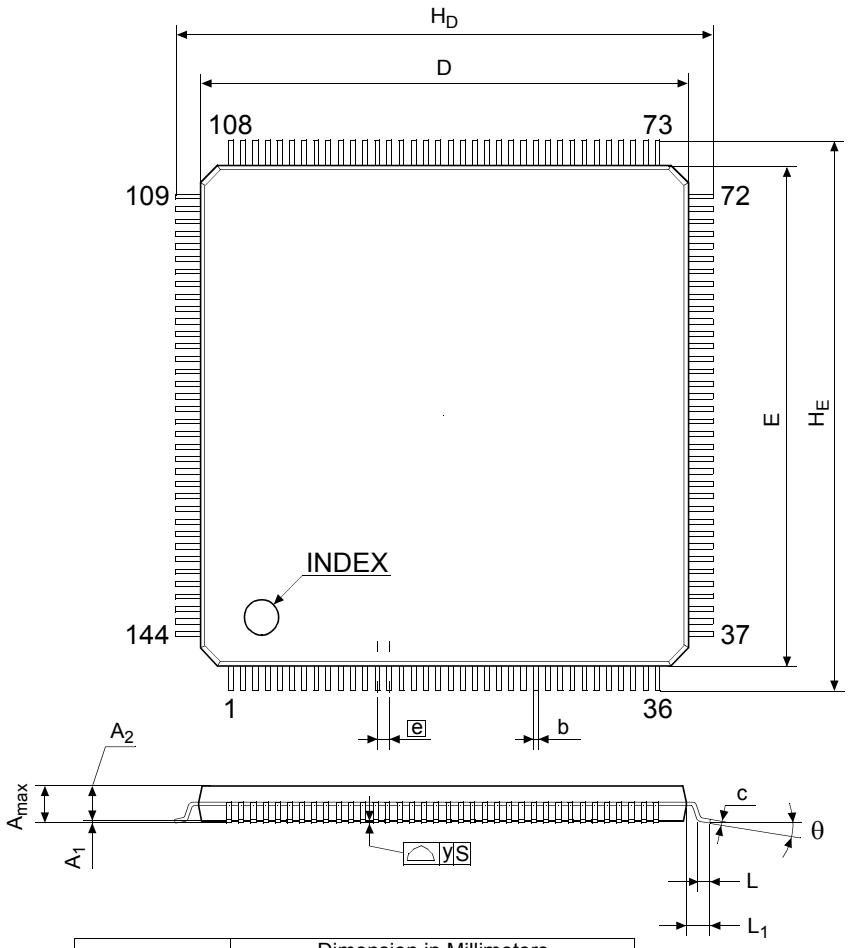


Figure 20-1: PLL Power Layout

- Place the ferrite beads (L1 and L2) parallel to each other with minimal clearance between them. Both bypass caps (C2 and C3) should be as close as possible to the inductors. The traces from C3 to the power planes should be short parallel traces on the same side of the board with just the normal small clearance between them. Any significant loop area here will induce noise. If there is a voltage regulator on the board, try to run these power traces directly to the regulator instead of dropping to the power planes (still follow above rules about parallel traces).
- The analog ground point where bypass cap (C2) connects to the ground isolation inductor (L2) becomes the analog ground central point for a ground star topology. None of the components connect directly to the analog ground pin of the MGE (PLLVSS) except for a single short trace from C2 to the PLLVSS pin. The ground side of the large bypass capacitor (C1) should also have a direct connection to the star point.
- The same star topology rules used for analog ground apply to the analog power connection where L2 connects to C2.
- All of the trace lengths should be as short as possible.
- If possible, have all the PLL traces on the same outside layer of the board. The only exception is C1, which can be put on the other side of the board if necessary. C1 does not have to be as close to the analog ground and power star points as the other components.
- If possible, include a partial plane under the PLL area only (area under PLL components and traces). The solid analog plane should be grounded to the C2 (bypass) pad. This plane won't help if it is too large. It is strictly an electrostatic shield against coupling from other layers' signals in the same board area. If such an analog plane is not possible, try to have the layer below the PLL components be a digital power plane instead of a signal layer.
- If possible, keep other board signals from running right next to PLL pin vias on any layer.
- Wherever possible use thick traces, especially with the analog ground and power star connections to either side of C2. Try to make them as wide as the component pads – thin traces are more inductive.

It is likely that manufacturing rules will prohibit routing the ground and power star connections as suggested. For instance, four wide traces converging on a single pad could have reflow problems during assembly because of the thermal effect of all the copper traces around the capacitor pad. One solution might be to have only a single trace connecting to the pad and then have all the other traces connecting to this wide trace a minimum distance away from the pad. Another solution might be to have the traces connect to the pad, but with thermal relief around the pad to break up the copper connection. Ultimately the board must also be manufacturable, so best effort is acceptable.

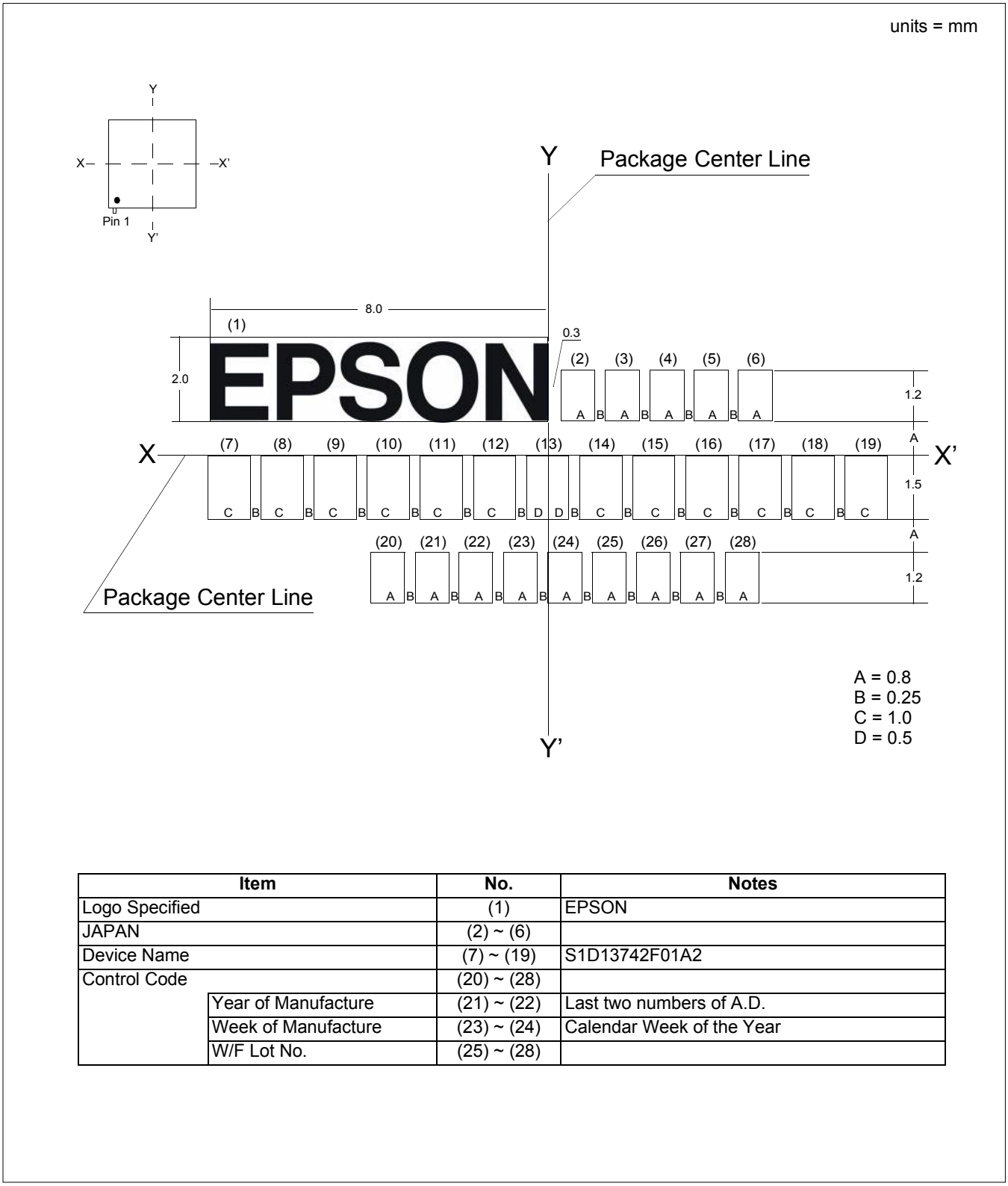
21 Mechanical Data



Symbol	Dimension in Millimeters		
	Min	Nom	Max
E	—	20	—
D	—	20	—
H_E	—	22	—
H_D	—	22	—
A_{max}	—	—	1.7
A_1	—	0.1	—
A_2	—	1.4	—
e	—	0.5	—
b	0.17	—	0.27
c	0.09	—	0.2
θ	0°	—	10°
L	0.3	—	0.75
L_1	—	1	—
y	—	—	0.08

units = mm

Figure 21-1: S1D13742 QFP20 144-pin Package



Item		No.	Notes
Logo Specified		(1)	EPSON
JAPAN		(2) ~ (6)	
Device Name		(7) ~ (19)	S1D13742F01A2
Control Code		(20) ~ (28)	
	Year of Manufacture	(21) ~ (22)	Last two numbers of A.D.
	Week of Manufacture	(23) ~ (24)	Calendar Week of the Year
	W/F Lot No.	(25) ~ (28)	

Figure 21-2: S1D13742 QFP 144-pin Package Marking

22 Change Record

X63A-A-001-06 Revision 6.6 - Issued: March 15, 2018

- updated Sales and Technical Support Section
- updated some formatting

X63A-A-001-06 Revision 6.5 - Issued: 2014/07/080

- Corrected Table 6.2 for Storage Temperature Min Value

X63A-A-001-06 Revision 6.4 - Issued: 2014/06/30

- Updated Table 6.2 for Storage Temperature values

X63A-A-001-06 Revision 6.3 - Issued: 2012/02/29

- globally delete FCBGA-121 ball package

X63A-A-001-06 Revision 6.2 - Issued: 2008/07/07

- all changes from the last revision of the spec are highlighted in Red
- Set revision to 6.2 to align with Japan revision numbering
- section 8.4 Setting SYSCLK and PCLK - add CLKI information to this section

X63A-A-001-06 Revision 6.01 - Issued: 2007/09/18

- all changes from the last revision of the spec are highlighted in Red
- section 5.1, for the Intel 80 Data Pin Mapping tables, swapped the MD[15:8] descriptions for CNF1=0, B00 should be “internal resistors” and B01 should be “Hi-Z”
- section 7.3.1 ~ 7.3.2, added note and clarified the usage of MD[15:8] pins in the Host Timing figures and tables
- section 17.1.3, updated the X/Y Start/End data order in the Sequential Memory Write Example Sequence figure and moved it to section 17.1.3
- section 22, added References
- section 23, added Sales and Technical Support addresses

X63A-A-001-06 Revision 6.0 (Issued 2007/05/29)

- all changes from the last revision of the spec are highlighted in Red
- section 14 Gamma Correction Look-Up Table Architecture - correct typos in Figure 14 - 1; change data from display buffers to 6 bit, change the multiplexers to 64 positions from 256
- section 19.1.3 S1D13742 Register Settings for 800x480 TFT Panel - correct typo in Table 19-3, change the REG[04h] value to 0Bh

X63A-A-001-05 Revision 5.02 (Issued 2006/08/23)

- all changes from the last revision of the spec are highlighted in Red
- globally add QFP20 144-pin package information

- section 5.3 LCD Interface Data Pins - correct typos in table, change Hi-Z to Driven Low
- section 6.3 Electrical Characteristics - add table 6-5 Electrical Characteristics for IOVDD or PIOVDD = 3.3V ± 0.3V
- section 7.2 RESET# Timing - add CLKI signal to figure
- section 7.3.1 Intel 80 Interface Timing - 1.8 Volt - rewrite section for 1.8 volts
- section 7.3.2 Intel 80 Interface Timing - 3.3 Volt - add this section
- REG[2Ah] bits 4-0 - add note “RGB 6:6:6 mode 2 and RGB 8:8:8 mode 2...”

X63A-A-001-05 Revision 5.01 (Issued 2006/04/28)

- updated EPSON tagline
- all changes from the last revision of the spec are highlighted in Red
- section 4.2.1 Intel 80 Host Interface - for MD[15:0] rewrite the note in pin description, for GPIO_INT add reference to General Purpose IO Pins Registers to pin description.
- section 4.2.2 LCD Interface - for VD[35:0] rewrite both notes in pin description
- section 4.2.4 Miscellaneous - for GPIO[7:0] rewrite pin description, for PWRSVE rewrite pin description for no pull-down resistor
- section 4.2.4, change SCANEN pin description IO Voltage from “VSS” to “IOVDD”
- section 7.2 RESET# Timing - add this section
- section 17.1.2 and 17.1.5, for the Host Interface section changed the references in the figures from “D[15:0]” to “MD[15:0]”
- fixed typo in change record, document numbers should be listed as “X63...” instead of “X59...”
- section 6.3 Electrical Characteristics - in tables 6-3 and 6-4, define the conditions for Quiescent Current

X63A-A-001-04 Revision 4.0 (Issued 2005/11/29)

- section 7.3.3 18/36-Bit TFT Panel Timing - correct typos in figure 7-8 18/36-Bit TFT A/C Timing - change references to REG[2Ah] to REG[28h], change t17 reference to falling edge of VS, and in table 7-7 18/36-Bit TFT A/C Timing change PCLK edge references to “active”

23 Sales and Technical Support

For more information on Epson Display Controllers, visit the Epson Global website.

https://global.epson.com/products_and_drivers/semicon/products/display_controllers/

For Sales and Technical Support, contact the Epson representative for your region.

https://global.epson.com/products_and_drivers/semicon/information/support.html