EPSON
EXCEED YOUR VISION

S1D13515 / S2D13515 Display Controller

Hardware Functional Specification

Document Number: X83A-A-001-01.8

SEIKO EPSON CORPORATION

Rev. 1.8

NOTICE

Table Of Contents

Chapter 1 Introduction

1.1 Scope

This is the Hardware Functional Specification for the S1D13515/S2D13515 Display 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 for the latest revision of this document before beginning any development. The latest revision can be downloaded at vdc.epson.com.

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

1.2 Overview Description

The S1D13515/S2D13515 is a highly integrated color LCD graphics controller with external memory interface. The architecture is designed to meet the needs of automotive and embedded markets requiring a flexible LCD solution. For automotive applications, the S2D13515 has three primary target placements within a vehicle.

1. Heads-Up Display
2. Instrument Cluster
3. Center Console

The S1D13515/S2D13515 advances on the successes of other Epson LCD controllers by embedding a proprietary 32-bit RISC CPU and associated accelerator blocks to achieve an increase in flexibility and functionality over previous designs. Routines are provided allowing for audio playback, 2D BitBLT operations, warp and filtering before display operations, and the ability to offer OpenGL-ES 1.1 support. In particular, the warp functions make this an ideal solution for the automotive Heads-Up Display (HUD) market, or pseudo 3D navigation displays.

The S1D13515/S2D13515 is an affordable, low power device which uses a flexible external SDRAM memory interface to provide its frame buffer. It supports a wide variety of CPU interfaces and LCD panel types, including Double Display panels, which makes it an excellent choice for instrumentation or center cluster applications. While focusing on the automotive market, the S1D13515/S2D13515's impartiality to CPU type or operating system makes it an ideal display solution for a wide variety of other markets.

The S1D13515/S2D13515 design includes some of the following key features:

1. Warp engine for HUD projection correction
2. Embedded 32-bit proprietary RISC CPU
3. Support for two TFT Displays simultaneously
4. Support for Double Display LCD panels from Epson and Sharp
5. The ability to provide OpenGL-ES library functionality
6. The ability to playback audio

7. The ability to reset and display an image without the Host CPU involvement

Chapter 2 Features

2.1 Memory

- Uses external SDRAM which is:
  - Accessible by both the internal and Host CPUs
  - Used for executable code, data, and the frame buffer
  - Addressable through direct or indirect access modes
  - Accessible linearly in configurable 4M byte paging windows (direct access mode)
- SDRAM Interface:
  - SDRAM Clock Frequency: 100Mhz (typical)
  - Supports x16 and x32 SDRAM interfaces (x32 is strongly recommended in most cases)
  - Supports 8/16/32/64M bytes of 4 bank SDRAM
  - Low power design

2.2 CPU Interfaces

Note
The S1D/S2D13515 supports Little Endian interface only.

- Direct and indirect interface support for the following CPU interfaces:
  - Intel 80 Types 1 and 2 (8/16-bit)
  - Renesas SH-4 (8/16-bit)
  - FreeScale MPC555 PowerPC bus interface with burst and non-burst modes (16-bit Little Endian configuration only)
  - NEC V850 Types 1 and 2 (8/16-bit)
  - Texas Instruments TMS470 with burst mode (16-bit only)
  - Marvell PXA3xx (16-bit Direct only)
- Serial Host Interface
  - SPI
  - I2C

2.3 Panel Interface Support

- Single or Dual panels (dual panel implementations can have independent images)
  - LCD1 supports:
    - 12/16/18-bit interface for Generic TFT/TFD
    - Optionally, LCD1 pins can be used for a second Camera / RGB data stream
  - LCD2 supports:
    - 12/16/18/24-bit interface for Generic TFT/TFD
    - EID Double Screen panel
    - Sharp DualView panel
  - Optional Serial Command interface supports:
    - a-Si TFT interface (8-bit)
    - TFT w/u-Wire interface (16-bit)
    - EPSON ND-TFD 4 pin interface (8-bit)
    - EPSON ND-TFD 3 pin interface (9-bit)
    - 24-bit serial
- Panel Resolution Examples:
  - 800x480 + 320x240 @ 32 bpp, 60Hz
  - 1024x768 @ 32 bpp, 60Hz
- TV-Out can be achieved by connecting an external TV encoder, such as the S1D13746, to the LCD outputs

2.4 Display Features

- Four input window sources can be stored in SDRAM (Main/Aux/OSD/LCD Fetcher) and support:
  - 8/16/24 bpp color depths
  - Hardware / Software Double Buffer Frame Control
  - Horizontal Flip
  - Virtual Width
  - Alpha Blending for the OSD
- Blending Engine can combine various input window sources for output
  - Three input sources
  - Input sources can be blended in four different ways
- Warp logic for HUD projection correction or other distortion compensation
  - Processed image can be sent back to SDRAM
- Camera1 or Camera2 image can be stored in SDRAM and used for Main/Aux/OSD/LCD Fetcher/Warp/Sprite
- Interrupt
  - Maskable Non-Display (Vsync) Interrupt support
  - Delayed version of Vsync Interrupt support
  - All interrupts are sent to the internal CPU, but can also be redirected to the HOST

2.5 Embedded CPU

- Embedded CPU Speed: 50MHz (typical)
- 32-bit RISC CPU with the following routines:
  - Audio decode (supported codecs: MP3, AAC, WAV, ADPCM, Ogg Vorbis)
  - 2D BitBLT Acceleration with API
    Some functions will be embedded in mask ROM, others will be provided as optional.
  - OpenGL-ES Assist (OpenGL-ES v1.1 compliant)
  - OEM defined functions

2.6 Sprite Engine

- 2D Sprite Engine
  - Up to Eight Sprites
  - Image rotation and mirror functions
  - Alpha Blending
  - Typical usage: Instrument Cluster, Simple GUI composition, etc.

2.7 Video / Camera Input

- Video / Camera input port supporting one of the following configurations:
  - up to two 8-bit cameras
  - up to two RGB data streams
  - one 8-bit camera and one RGB data stream
  - Note: When the second camera input is used, only a single panel is available.
- Supports ITU-R BT.656 YUV format
- Supports Interlaced or Progressive input
- Supports down-scaling of the video input stream
- Captures YUV Data into SDRAM as RGB format

2.8 Clock Source

- Flexible Clock Structure:
  - Two embedded PLLs
  - Built-in crystal input
  - Digital clock input
- Clocks are dynamically turned off when modules are not needed

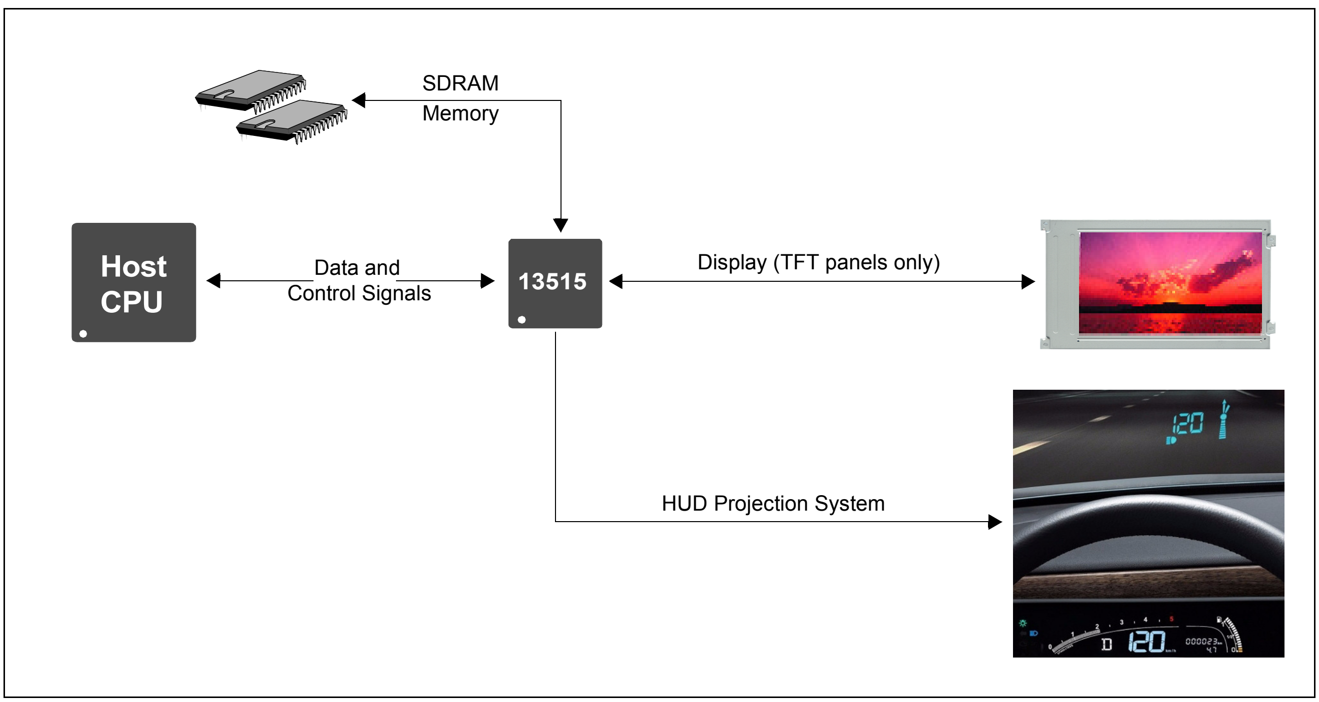
2.9 Miscellaneous

- Internal System Clock Speed: 50MHz (typical)
- IRQ output pin
  - Multiple interrupt sources (LCD1 / LCD2 / DMA / Timer / Keypad / etc.)
- I2C interface (typically used for camera)
- I2S interface (typically used for audio output)
- PWM: 2 channel for backlight control
- SPI Flash Memory interface
- Keypad Interface
  - 5 x 5 matrix support
- Software initiated power save mode
- General Purpose Input/Output pins are available
- IO operates at 3.3 volts ± 0.3v
- Core operates at 1.8 volts ± 0.15v
- Packages:
  - S1D13515B00B - PBGA1U 256-pin package (Body Size: 17 x 17 x 1.7 mm, Ball pitch: 1.0 mm)
  - S2D13515B00B - PBGA1U 256-pin package (Body Size: 17 x 17 x 1.7 mm, Ball pitch: 1.0 mm)
  - S1D13515F00A - QFP22 256-pin package (Body Size: 28 x 28 x 1.4 mm, Pin pitch: 0.4 mm)
  - S2D13515F00A - QFP22 256-pin package (Body Size: 28 x 28 x 1.4 mm, Pin pitch: 0.4 mm)
- Temperature Range:
  - S1D13515; -40° C to +85° C
  - S2D13515; -40° C to +105° C

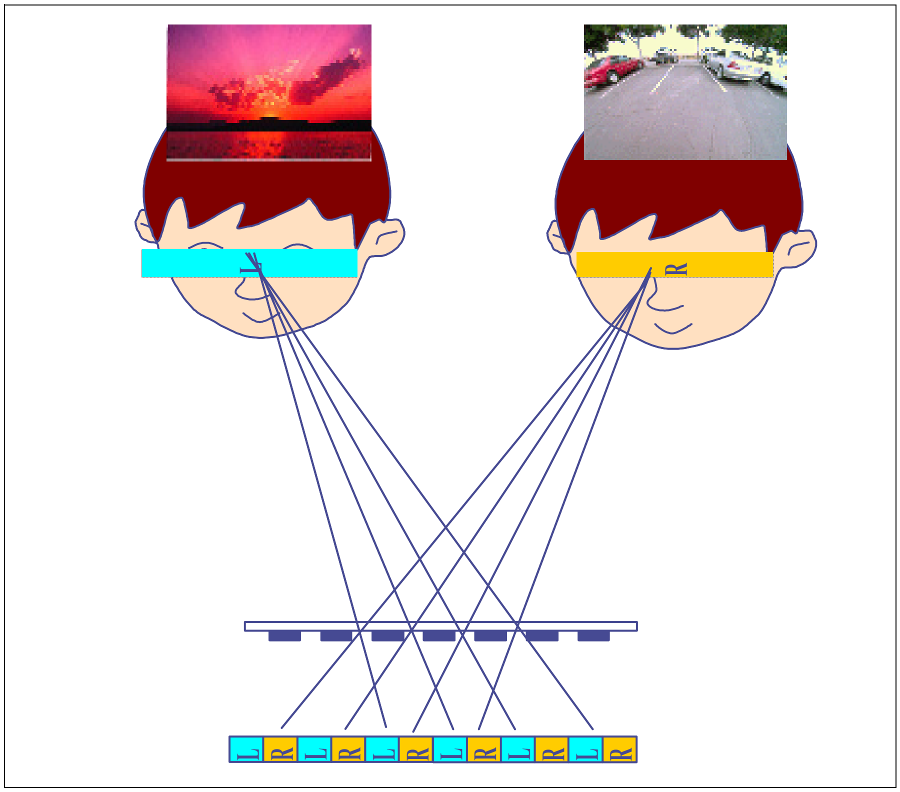
Chapter 3 Typical Implementation Use Cases

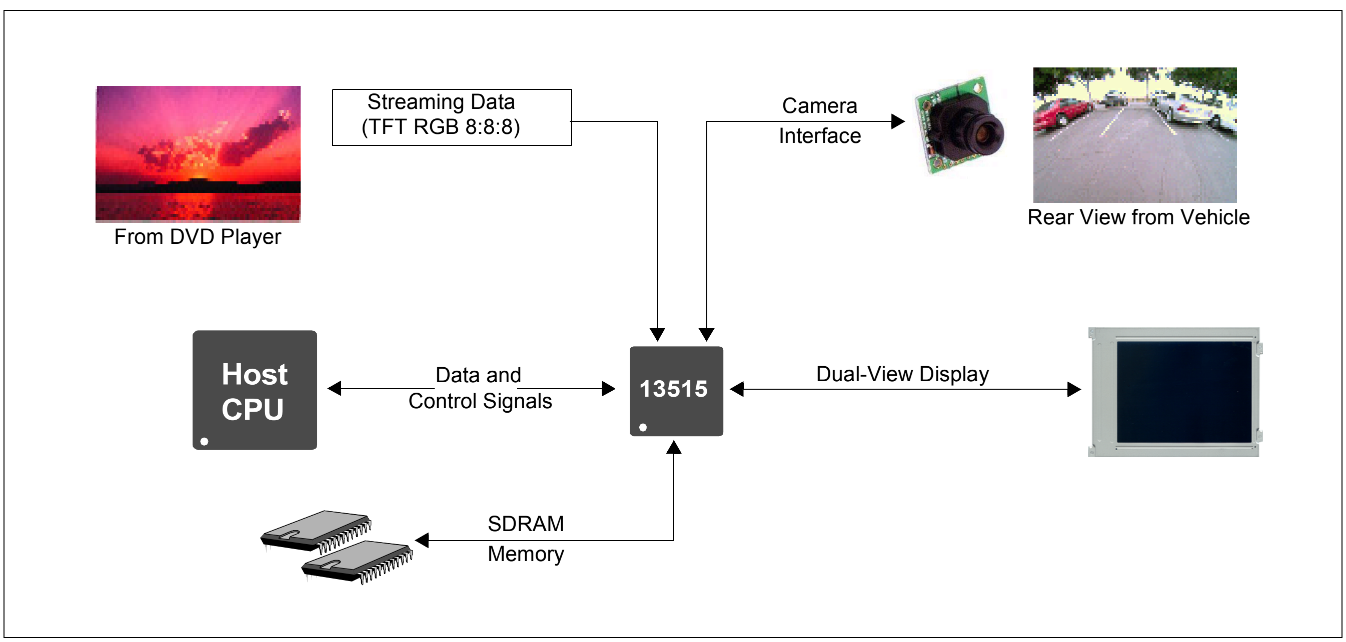
The following are generic Use Cases. For specific implementations of the S1D13515 and S2D13515, please see the Application Notes.

3.1 Use Case 1 - Heads-Up Display (HUD) with LCD Panel



3.2 Use Case 2 - Dual-View Panel with Streaming Data and Camera Input





Chapter 4 Block Diagram

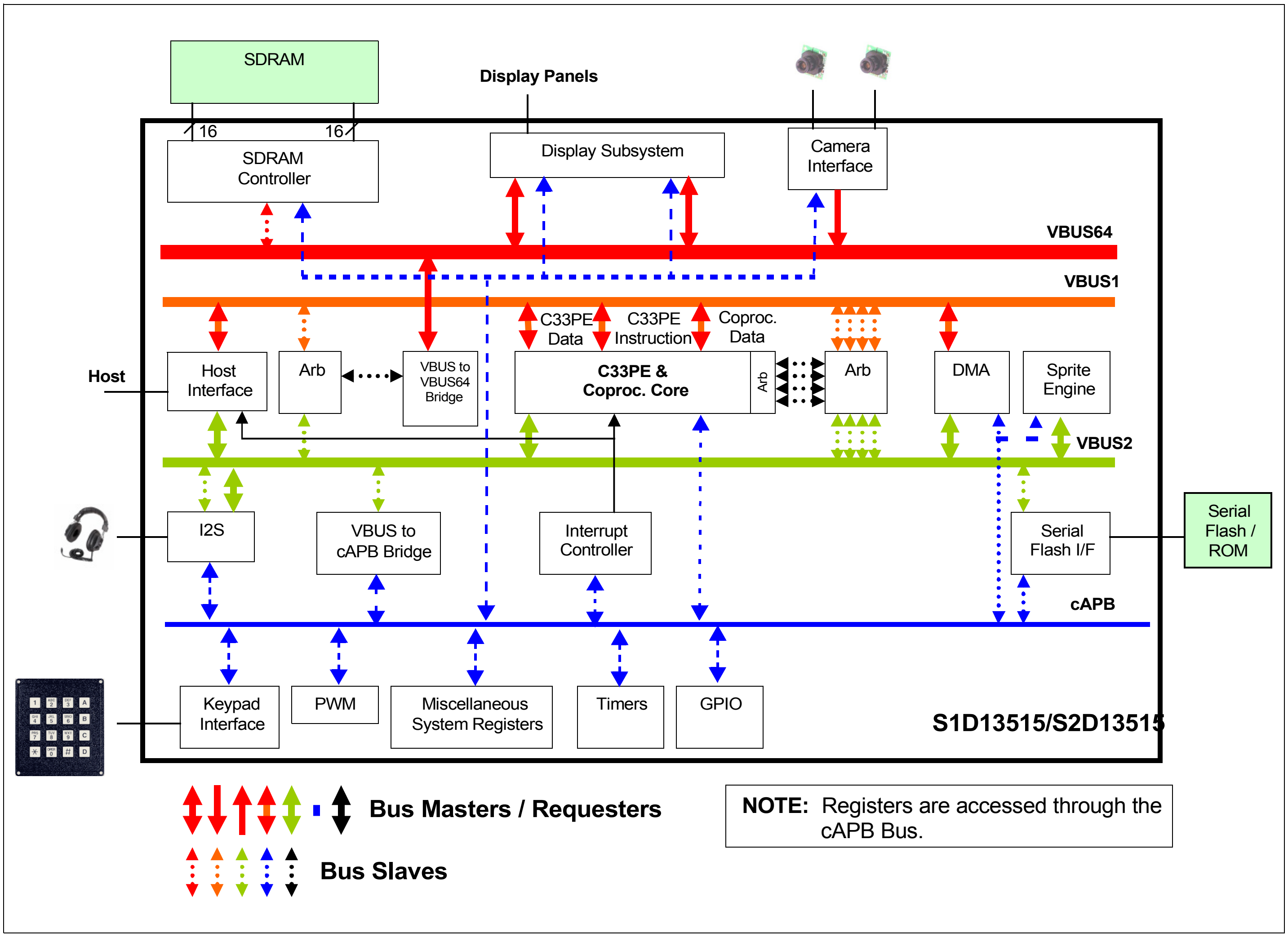


Figure 4-1: Block Diagram

Chapter 5 Pins

5.1 Pinout Diagram (QFP22 256-pin)

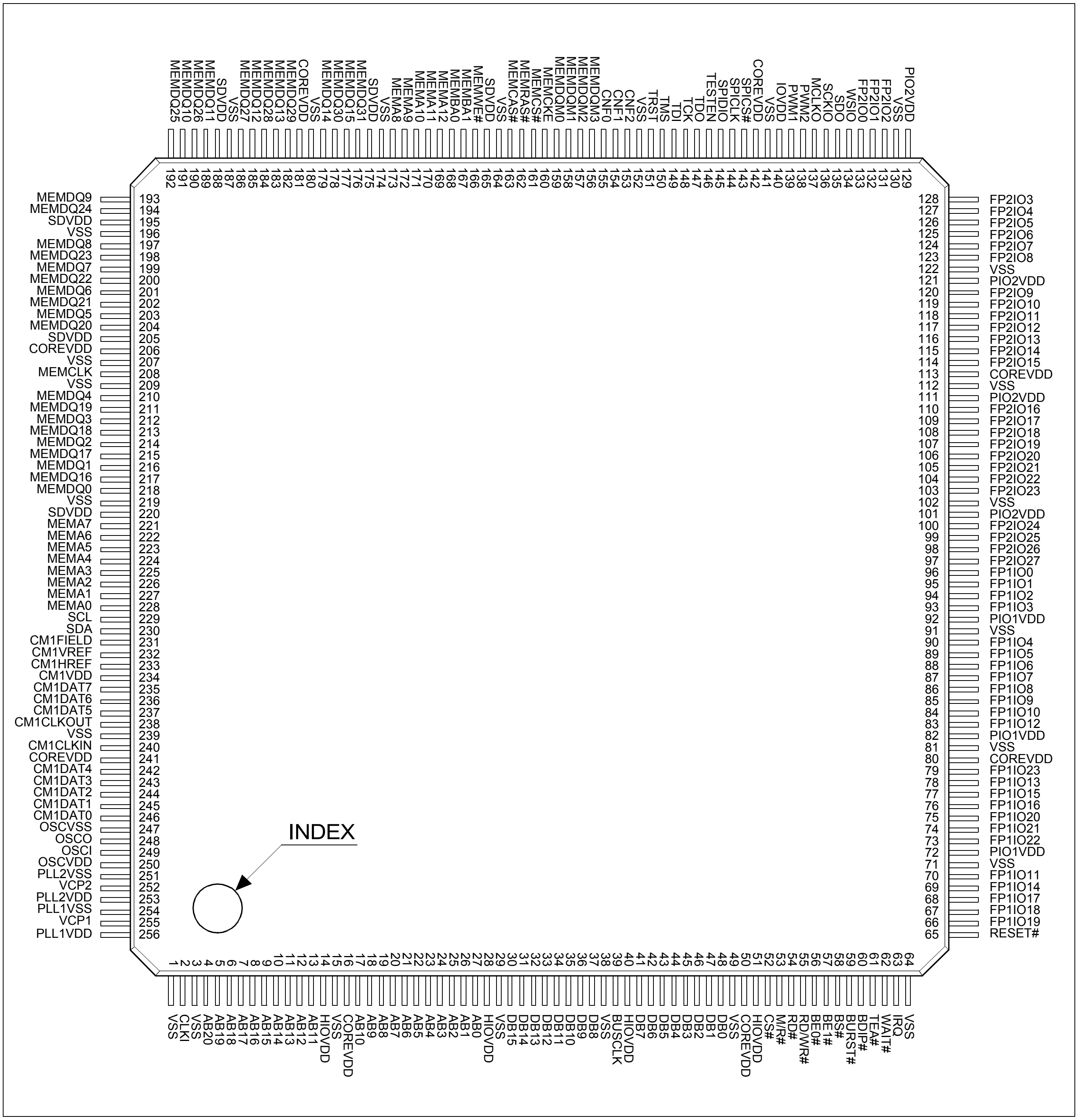


Figure 5-1: QFP22-256 Pin Mapping

5.2 Pinout Diagram (PBGA 256-pin)

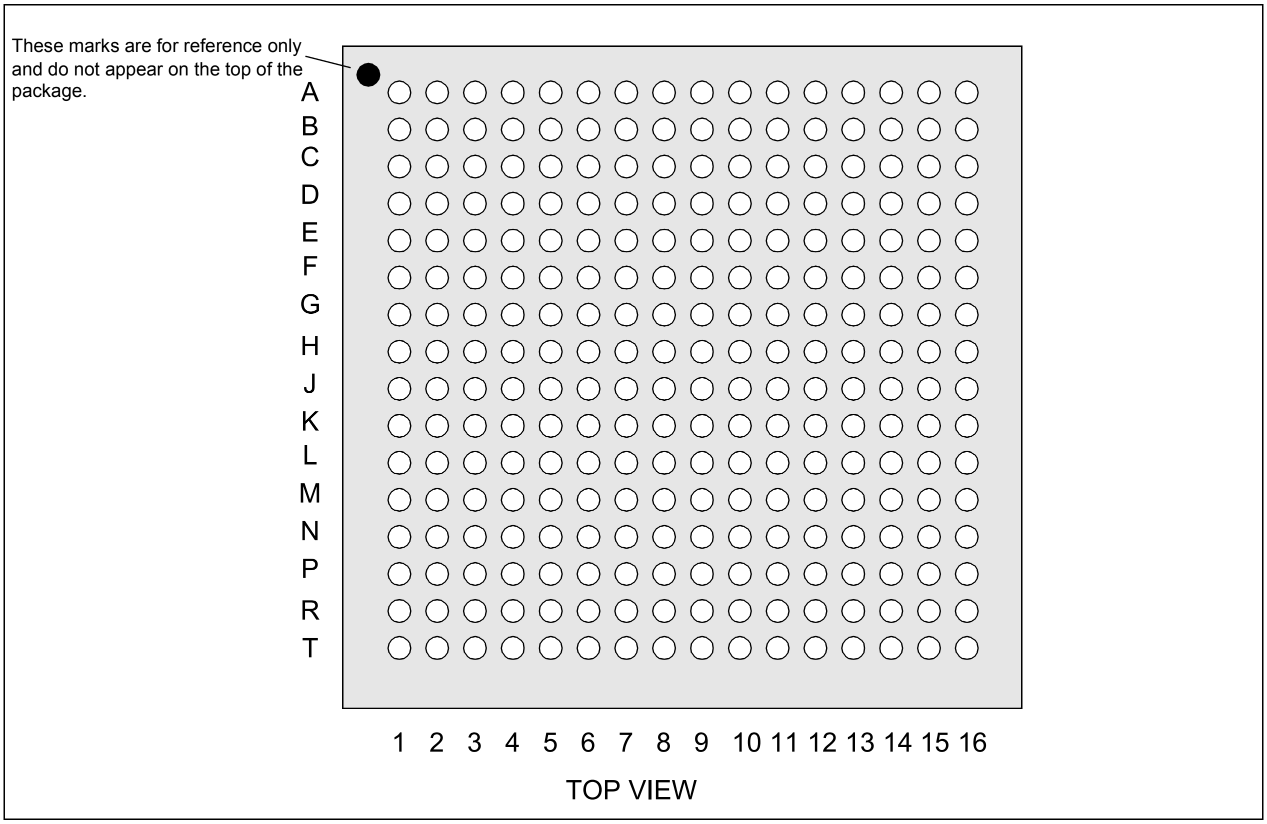


Figure 5-2: PBGA1U-256 Pin Mapping

Table 5-1: PBGA1U-256 Pin Mapping

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
A	VSS	VCP1	PLL2VDD	VCP2	OSCI	OSCO	CM1CLKOUT	CM1DAT5	MEMA0	MEMA6	MEMDQ1	MEMCLK	MEMDQ21	MEMDQ23	SDVDD	VSS	A
B	CLKI	PLL1VDD	PLL1VSS	PLL2VSS	OSCVDD	OSCVSS	CM1CLKIN	CM1VREF	MEMA1	SDVDD	MEMDQ18	VSS	MEMDQ6	MEMDQ8	MEMDQ25	MEMDQ10	B
C	AB20	VSS	CM1DAT0	CM1DAT1	CM1DAT2	CM1DAT3	VSS	CM1FIELD	MEMA3	MEMDQ0	MEMDQ4	COREVDD	MEMDQ22	MEMDQ9	MEMDQ26	MEMDQ11	C
D	AB15	AB16	AB18	AB19	CM1DAT4	COREVDD	CM1DAT6	SDA	MEMA5	MEMDQ16	VSS	MEMDQ20	MEMDQ24	SDVDD	MEMDQ27	MEMDQ12	D
E	COREVDD	HIOVDD	AB13	AB14	AB17	CM1DAT7	CM1VDD	SCL	MEMA7	MEMDQ2	SDVDD	MEMDQ7	VSS	MEMDQ28	VSS	COREVDD	E
F	AB6	AB7	AB10	VSS	AB11	AB12	CM1HREF	MEMA2	MEMDQ17	MEMDQ19	MEMDQ5	MEMDQ29	MEMDQ14	MEMDQ30	MEMA8	SDVDD	F
G	HIOVDD	AB2	AB3	AB4	AB5	AB8	AB9	MEMA4	MEMDQ3	MEMDQ13	MEMDQ15	MEMDQ31	VSS	MEMA9	MEMA10	MEMA12	G
H	DB12	DB15	DB13	DB14	VSS	AB0	AB1	VSS	VSS	MEMA11	MEMBA0	MEMBA1	MEMWE#	SDVDD	MEMRAS#	MEMCAS#	H
J	BUSCLK	DB8	DB9	HIOVDD	DB7	DB10	DB11	VSS	VSS	CNF0	MEMDQM3	MEMDQM2	MEMDQM1	MEMDQM0	MEMCS#	MEMCKE	J
K	DB3	DB2	DB4	DB5	DB6	DB1	FP1IO10	FP2IO26	FP2IO18	FP2IO10	TCK	TMS	TRST	VSS	CNF1	CNF2	K
L	DB0	COREVDD	CS#	VSS	HIOVDD	FP1IO16	FP1IO9	FP1IO0	FP2IO21	FP2IO13	SPIDIO	SPICLK	VSS	TESTEN	TDO	TDI	L
M	M/R#	RD#	RD/WR#	BE0#	BS#	FP1IO15	FP1IO8	FP1IO1	FP2IO22	VSS	FP2IO6	PWM2	PWM1	IOVDD	SPICS#	COREVDD	M
N	BE1#	BURST#	BDIP#	VSS	FP1IO21	COREVDD	FP1IO7	VSS	PIO2VDD	FP2IO17	FP2IO14	FP2IO8	WSIO	SDO	SCKIO	MCLKO	N
P	WAIT#	TEA#	FP1IO19	FP1IO14	FP1IO20	VSS	FP1IO4	FP1IO2	FP2IO24	FP2IO19	FP2IO15	FP2IO9	FP2IO7	FP2IO0	FP2IO2	FP2IO1	P
R	IRQ	RESET#	FP1IO17	FP1IO22	FP1IO13	FP1IO12	FP1IO5	FP1IO3	FP2IO25	FP2IO20	PIO2VDD	FP2IO12	VSS	FP2IO4	VP2IO3	PIO2VDD	R
T	VSS	FP1IO18	FP1IO11	PIO1VDD	FP1IO23	PIO1VDD	FP1IO6	PIO1VDD	FP2IO27	FP2IO23	FP2IO16	COREVDD	FP2IO11	PIO2VDD	FP2IO5	VSS	T
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	

5.3 Pin Descriptions

Key:

Pin Types

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

RESET# States

H	=	High level output
L	=	Low level output
Z	=	High Impedance (Hi-Z)
1	=	Pull-up resistor on input
0	=	Pull-down resistor on input
#	=	Active low level

Table 5-2: Cell Descriptions

Cell	Description
ILTR	Low voltage transparent input
OLTR	Low voltage transparent output
IC	LVCMOS input
ICS	LVCMOS schmitt input
ICD1T	LVCMOS input with pull-down resistor (50kΩ@3.3V) with Test Function
ICSU1T	LVCMOS schmitt input with pull-up resistor (50kΩ@3.3V) with Test Function
ICSU2T	LVCMOS schmitt input with pull-up resistor (100kΩ@3.3V) with Test Function
ICSD1T	LVCMOS schmitt input with pull-down resistor (50kΩ@3.3V) with Test Function
IOC2P1T	Low noise LVCMOS IO buffer (2mA/4mA@3.3V) with pull-up resistor (50kΩ@3.3V) with Test Function
IOC2P2T	Low noise LVCMOS IO buffer (2mA/4mA@3.3V) with pull-up resistor (100kΩ@3.3V) with Test Function
IOC2D1T	Low noise LVCMOS IO buffer (2mA/4mA@3.3V) with pull-down resistor (50kΩ@3.3V) with Test Function
IOC2D2T	Low noise LVCMOS IO buffer (2mA/4mA@3.3V) with pull-down resistor (100kΩ@3.3V) with Test Function
IOCS2D1T	Low noise LVCMOS schmitt IO buffer (2mA/4mA@3.3V) with pull-down resistor (50kΩ@3.3V) with Test Function
OLT2T	Low noise 3-state Output buffer (2mA/4mA@3.3V) with Test Function
OLT3	Low noise 3-state Output buffer (8mA@ 3.3V)
OLT3T	Low noise 3-state Output buffer (8mA@ 3.3V) with Test Function
P	Power

5.3.1 Host Interface

Many of the host interface pins have different functions depending on the host bus interface that is selected. For a summary of the possible host bus interface configurations and associated pin mapping details, see Section 5.4, “Configuration Pins” on page 33 and Section 5.5, “Host Interface Pin Mapping” on page 35. To determine the RESET# state for each pin, refer to Section 11.1, “Hard Reset State” on page 438.

Table 5-3: Host Interface Pin Descriptions

Pin Name	Type	QFP Pin#	PBGA Pin#	Cell	Power	Description
AB[20:19]	IO	4, 5	C1, D4	IOCS2D1T	HIOVDD	These input/output pins are the host address bus pins 20-19. For a summary of which pins are used for each host bus interface configuration, see Section 5.5, “Host Interface Pin Mapping” on page 35.
AB18	I	6	D3	ICSD1T	HIOVDD	This input pin is the host address pin 18. For a summary of which pins are used for each host bus interface configuration, see Section 5.5, “Host Interface Pin Mapping” on page 35.
AB[17:8]	IO	7, 8, 9, 10, 11, 12, 13, 17, 18, 19	E5, D2, D1, E4, E3, F6, F5, F3, G7, G6	IOCS2D1T	HIOVDD	These input/output pins are the host address bus pins 17-6. For a summary of which pins are used for each host bus interface configuration, see Section 5.5, “Host Interface Pin Mapping” on page 35.
AB7	IO	20	F2	IOCS2D1T	HIOVDD	This input/output pin is the host address bus pin 7. For a summary of which pins are used for each host bus interface configuration, see Section 5.5, “Host Interface Pin Mapping” on page 35.
AB6	IO	21	F1	IOCS2D1T	HIOVDD	This input/output pin is the host address bus pin 6. For a summary of which pins are used for each host bus interface configuration, see Section 5.5, “Host Interface Pin Mapping” on page 35.
AB[5:0]	I	22-27	G5, G4, G3, G2, H7, H6	ICSD1T	HIOVDD	These input pins are the host address bus pins 5-0. For a summary of which pins are used for each host bus interface configuration, see Section 5.5, “Host Interface Pin Mapping” on page 35.
DB[15:10]	IO	30, 31, 32, 33, 34, 35, 36	H2, H4, H3, H1, J7, J6	IOC2D1T	HIOVDD	These input/output pins are the host data bus pins 15-10. For a summary of which pins are used for each host bus interface configuration, see Section 5.5, “Host Interface Pin Mapping” on page 35.
DB9	IO	36	J3	IOC2D1T	HIOVDD	This input/output pin is the host data bus pin 9. For a summary of which pins are used for each host bus interface configuration, see Section 5.5, “Host Interface Pin Mapping” on page 35.
DB[8:0]	IO	37, 41, 42, 43, 44, 45, 46, 47, 48	J2, J5, K5, K4, K3, K1, K2, K6, L1	IOC2D1T	HIOVDD	These input/output pins are the host data bus pins 8-0. For a summary of which pins are used for each host bus interface configuration, see Section 5.5, “Host Interface Pin Mapping” on page 35.
CS#	I	52	L3	ICD1T	HIOVDD	This input pin is Chip Select.
M/R#	IO	53	M1	IOCS2D1T	HIOVDD	This input/output pin has multiple functions. For a summary of the pin functions for each host bus interface configuration, see Section 5.5, “Host Interface Pin Mapping” on page 35.

Table 5-3: Host Interface Pin Descriptions

Pin Name	Type	QFP Pin#	PBGA Pin#	Cell	Power	Description
RD#	I	54	M2	ICD1T	HIOVDD	This input pin has multiple functions. For a summary of the pin functions for each host bus interface configuration, see Section 5.5, “Host Interface Pin Mapping” on page 35.
RD/WR#	I	55	M3	ICD1T	HIOVDD	This input pin has multiple functions. For a summary of the pin functions for each host bus interface configuration, see Section 5.5, “Host Interface Pin Mapping” on page 35.
BE0#	I	56	M4	ICD1T	HIOVDD	This input pin has multiple functions. For a summary of the pin functions for each host bus interface configuration, see Section 5.5, “Host Interface Pin Mapping” on page 35.
BE1#	IO	57	N1	IOC2D1T	HIOVDD	This input/output pin has multiple functions. For the Intel 80 Type 2 Indirect 8-bit Host Interface, this pin must be connected to HIOVDD. For a summary of the pin functions for each host bus interface configuration, see Section 5.5, “Host Interface Pin Mapping” on page 35.
BS#	IO	58	M5	IOC2P2T	HIOVDD	This input/output pin has multiple functions. For a summary of the pin functions for each host bus interface configuration, see Section 5.5, “Host Interface Pin Mapping” on page 35.
BURST#	I	59	N2	IC	HIOVDD	This input pin is Burst Transfer for the MPC555 and TI TMS470 Host interfaces and is used for burst support. For all other host bus interfaces, it is used in combination with the CNF[2:1] pins for selecting the host bus interface. For a summary of all possible host bus interfaces, see Section 5.4, “Configuration Pins” on page 33.
BDIP#	I	60	N3	IC	HIOVDD	This input pin is used for the MPC555 and TI TMS470 Host interfaces and indicates a burst transfer is in progress. For all other host bus interfaces, it is used in combination with the CNF[2:1] pins for selecting the host bus interface. For a summary of all possible host bus interfaces, see Section 5.4, “Configuration Pins” on page 33.

Table 5-3: Host Interface Pin Descriptions

Pin Name	Type	QFP Pin#	PBGA Pin#	Cell	Power	Description
TEA#	IO	61	P2	IOC2D1T	HIOVDD	This input/output pin is Transfer Error Acknowledge and is used for burst support for the MPC555 and TI TMS470 Host interfaces. This signal indicates that a bus error occurred in the current transaction. The MCU asserts this signal when the bus monitor does not detect a bus cycle termination within a reasonable amount of time. The assertion of TEA# causes the termination of the current bus cycle, regardless of the state of TEA#. An external pull-up device is required to negate TEA# quickly, before a second error is detected. That is, the pin must be pulled up within one clock cycle of the time it was tri-stated by the MPC555 / TI TMS470. For all other host bus interfaces, it is used in combination with the CNF[2:1] pins for selecting the host bus interface. For a summary of all possible host bus interfaces, see Section 5.4, “Configuration Pins” on page 33.
WAIT#	IO	62	P1	IOC2P2T	HIOVDD	During a data transfer, this output pin is driven active to force the system to insert wait states. It is driven inactive to indicate the completion of a data transfer. WAIT# is released to a high impedance state after the data transfer is complete. For a summary of the pin functions for each host bus interface configuration, see Section 5.5, “Host Interface Pin Mapping” on page 35.
BUSCLK	I	39	J1	ICD1T	HIOVDD	This input clock is typically used for an external clock source for the Host CPU bus interface. For a summary of the pin functions for each host bus interface configuration, see Section 5.5, “Host Interface Pin Mapping” on page 35.
IRQ	O	63	R1	OLT2T	HIOVDD	This output pin is the IRQ output from the S1D13515/S2D13515.

5.3.2 LCD Interface

The LCD interface consists of LCD1 and LCD2. LCD1 uses the FP1IO[23:0] pins and LCD2 uses the FP2IO[27:0] pins. Alternately, LCD1 can be used as a Camera2 or RGB stream input. For detailed pin mapping, see Section 5.6, “LCD / Camera2 Pin Mapping” on page 40. To determine the RESET# state for each pin, refer to Section 11.1, “Hard Reset State” on page 438.

Table 5-4: LCD Interface Pin Descriptions

Pin Name	Type	QFP Pin#	PBGA Pin#	Cell	Power	Description
FP1IO[23:0]	IO	79, 73, 74, 75, 66, 67, 68, 76, 77, 69, 78, 83, 70, 84, 85, 86, 87, 88, 89, 90, 93, 94, 95, 96	T5, R4, N5, P5, P3, T2, R3, L6, M6, P4, R5, R6, T3, K7, L7, M7, N7, T7, R7, P7, R8, P8, M8, L8	IOCS2D1T	PIO1VDD	These input/output pins may be used for one of the following options. Note that if an EID Double Screen panel with TCON enabled is used on FP2, the available options may differ. • 18-bit TFT panel • 16-bit TFT panel w/ serial command interface • 15-bit TFT panel (when EID Double Screen with TCON enabled is on FP2) • 12-bit TFT panel w/ serial command interface (when EID Double Screen with TCON enabled on FP2) • 18-bit RGB input stream • 8-bit Camera2 input and 5x5 keypad/GPIOs • 15-bit RGB input stream (when EID Double Screen with TCON enabled is on FP2) • 8-bit Camera2 input and 3x3 keypad/GPIOs (when EID Double Screen with TCON enabled is on FP2) Note that for some options, unused pins may be available as GPIO pins. For detailed pin mapping for each option, see Section 5.6, “LCD / Camera2 Pin Mapping” on page 40.
FP2IO[27:24]	O	97, 98, 99, 100	T9, K8, R9, P9	OLT2T	PIO2VDD	These input/output pins may be used for one of the following options. • 24-bit TFT panel • 18-bit TFT panel w/ serial command interface • 18-bit TFT panel • EID 18-bit Double Screen panel with TCON disabled • EID 18-bit Double Screen panel with TCON enabled • Sharp 18-bit DualView panel Note that for some options, unused pins may be available as GPIO pins. For detailed pin mapping for each option, see Section 5.6, “LCD / Camera2 Pin Mapping” on page 40.
FP2IO[23:18]	IO	103, 104, 105, 106, 107, 108	T10, M9, L9, R10, P10, K9	IOCS2D1T	PIO2VDD	
FP2IO17	IO	109	N10	IOC2P1T	PIO2VDD	
FP2IO[16:0]	O	110, 114, 115, 116, 117, 118, 119, 120, 123, 124, 125, 126, 127, 128, 131, 132, 133	T11, P11, N11, L10, R12, T13, K10, P12, N12, P13, M11, T15, R14, R15, P15, P16, P14	OLT2T	PIO2VDD	

5.3.3 SDRAM Interface

To determine the RESET# state for each pin, refer to Section 11.1, “Hard Reset State” on page 438.

Table 5-5: SDRAM Interface Pin Descriptions

Pin Name	Type	QFP Pin#	PBGA Pin#	Cell	Power	Description
MEMA[12:0]	O	169, 170, 171, 172, 173, 221, 222, 223, 224, 225, 226, 227, 228	G16, H10, G15, G14, F15, E9, A10, D9, G8, C9, F8, B9, A9	OLT2T	SDVDD	These output pins are used for SDRAM bank row/column address mapping.
MEMBA[1:0]	O	167, 168	H12, H11	OLT2T	SDVDD	These output pins are used to select the SDRAM bank address.
MEMCS#	O	161	J15	OLT2T	SDVDD	This output pin is the chip select for the SDRAM.
MEMRAS#	O	162	H15	OLT2T	SDVDD	This output pin is the RAS# for the SDRAM.
MEMCAS#	O	163	H16	OLT2T	SDVDD	This output pin is the CAS# for the SDRAM.
MEMWE#	O	166	H13	OLT2T	SDVDD	This output pin is the write enable for the SDRAM.
MEMDQ[31:16]	IO	176, 178, 182, 184, 186, 190, 192, 194, 198, 200, 202, 204, 211, 213, 215, 217	G12, F14, F12, E14, D15, C15, B15, D13, A14, C13, A13, D12, F10, B11, F9, D10	IOC2D2T	SDVDD	These input/output pins are the upper data bus used for x32 SDRAM configurations. For x16 SDRAM configurations, these pins must be left unconnected since they have internal pull-down resistors.
MEMDQ[15:0]	IO	177, 179, 183, 185, 189, 191, 193, 197, 199, 201, 203, 210, 212, 214, 216, 218	G11, F13, G10, D16, C16, B16, C14, B14, E12, B13, F11, C11, G9, E10, A11, C10	IOC2D2T	SDVDD	These input/output pins are the data bus for the SDRAM. They are used for both x16 and x32 configurations. These pins have internal pull-down resistors.
MEMDQM[3:2]	O	156, 157	J11, J12	OLT2T	SDVDD	These output pins are the upper byte enables used for x32 SDRAM configurations. For x16 SDRAM configurations, they must be left unconnected.
MEMDQM[1:0]	O	158, 159	J13, J14	OLT2T	SDVDD	These output pins are the byte enables for the SDRAM. They are used for both x16 and x32 configurations.
MEMCLK	O	208	A12	OLT3T	SDVDD	This output pin is the clock for the SDRAM.
MEMCKE	O	160	J16	OLT2T	SDVDD	This output pin is the clock enable for the SDRAM.

5.3.4 Camera / I2C Interface

To determine the RESET# state for each pin, refer to Section 11.1, “Hard Reset State” on page 438.

Table 5-6: Camera / I2C Interface Pin Descriptions

Pin Name	Type	QFP Pin#	PBGA Pin#	Cell	Power	Description
CM1DAT[7:0]	I	235, 236, 237, 242, 243, 244, 245, 246	E6, D7, A8, D5, C6, C5, C4, C3	ICD1T	CM1VDD	These input pins are the Camera1 interface data pins.
CM1CLKIN	I	240	B7	ICD1T	CM1VDD	This pin is the camera clock input for the Camera1 interface.
CM1CLKOUT	O	238	A7	OLT2T	CM1VDD	This pin is the master clock output for the Camera1 interface.
CM1HREF	I	233	F7	ICD1T	CM1VDD	This input pin is the horizontal sync signal for the Camera1 interface.
CM1VREF	I	232	B8	ICD1T	CM1VDD	This input pin is the vertical sync signal for the Camera1 interface.
CM1FIELD	I	231	C8	ICD1T	CM1VDD	This input pin identifies the FIELD for interlaced input on the Camera1 interface.
SCL	IO	229	E8	IOC2P2T	CM1VDD	This input/output pin is the I2C bus serial clock. If the I2C interface is not used, this pin should be left unconnected.
SDA	IO	230	D8	IOC2P2T	CM1VDD	This input/output pin is the I2C bus serial data. If the I2C interface is not used, this pin should be left unconnected.

5.3.5 SPI Flash Interface

To determine the RESET# state for each pin, refer to Section 11.1, “Hard Reset State” on page 438.

Table 5-7: SPI Flash Interface Pin Descriptions

Pin Name	Type	QFP Pin#	PBGA Pin#	Cell	Power	Description
SPICS#	O	143	M15	OLT2T	IOVDD	This output pin is chip select for the SPI Flash Memory interface.
SPICLK	O	144	L12	OLT2T	IOVDD	This output pin is the clock for the SPI Flash Memory interface.
SPIDIO	IO	145	L11	IOC2D2T	IOVDD	This input/output is the data pin for the SPI Flash Memory interface. If the SPI Flash interface is not used, this pin should be left unconnected.

5.3.6 I2S Interface

To determine the RESET# state for each pin, refer to Section 11.1, "Hard Reset State" on page 438.

Table 5-8: I2S Interface Pin Descriptions

Pin Name	Type	QFP Pin#	PBGA Pin#	Cell	Power	Description
WSIO	IO	134	N13	IOC2P2T	IOVDD	This pin is the serial word clock input/output for the I2S interface. This pin is configured based on the setting of the I2S Data Clock Source bit, REG[0100h] bit 0. If the I2S interface is not used, this pin should be left unconnected.
SCKIO	IO	136	N15	IOC2P2T	IOVDD	This pin is the serial bit clock input/output for the I2S interface. This pin is configured based on the setting of the I2S Data Clock Source bit, REG[0100h] bit 0. If the I2S interface is not used, this pin should be left unconnected.
SDO	O	135	N14	OLT2T	IOVDD	This pin is the serial data output for the I2S interface.
MCLKO	O	137	N16	OLT2T	IOVDD	This pin is the bus output clock to the DAC for the I2S interface.

5.3.7 Miscellaneous

To determine the RESET# state for each pin, refer to Section 11.1, "Hard Reset State" on page 438.

Table 5-9: Miscellaneous Pin Descriptions

Pin Name	Type	QFP Pin#	PBGA Pin#	Cell	Power	Description
CNF[2:1]	I	153, 154	K16, K15	IC	IOVDD	These input pins are used in combination with other pins to select the host bus interface. For a summary of all possible host bus interfaces, see Section 5.4, "Configuration Pins" on page 33.
CNF0	I	155	J10	IC	IOVDD	This input pin is used to select the source for Input Clock 1 (see Chapter 9, "Clocks" on page 131). When CNF0=0, CLKI is the source for Input Clock 1. When CNF0=1, OSCI is the source for Input Clock 1.
OSCI	I	249	A5	ILTR	OSCVDD	Crystal input. If an external oscillator circuit is used, connect it to this pin. For details on the clock structure, see Chapter 9, "Clocks" on page 131.
OSCO	O	248	A6	OLTR	OSCVDD	Crystal output. If an external oscillator circuit is used, this pin must be left unconnected. For details on the clock structure, see Chapter 9, "Clocks" on page 131.
CLKI	I	2	B1	IC	HIOVDD	Clock input. For details on the clock structure, see Chapter 9, "Clocks" on page 131.
TESTEN	I	146	L14	ICSD1T	IOVDD	This input pin is for production test only and must be connected to VSS for normal operation.

Table 5-9: Miscellaneous Pin Descriptions

Pin Name	Type	QFP Pin#	PBGA Pin#	Cell	Power	Description
VCP1	O	255	A2	OLTR	PLL1VDD	This output pin is for production test only and must be left unconnected for normal operation.
VCP2	O	252	A4	OLTR	PLL2VDD	This output pin is for production test only and must be left unconnected for normal operation.
RESET#	I	65	R2	ICS	HIOVDD	This active low input sets all internal registers to their default states and forces all signals to their inactive states. For RESET# timing, see Section 7.3, “RESET# Timing” on page 51.
PWM2	O	138	M12	OLT2T	IOVDD	This output pin is for PWM output.
PWM1	O	139	M13	OLT2T	IOVDD	This output pin is for PWM output.
TCK	I	148	K11	ICSU1T	IOVDD	This input pin is a JTAG interface pin used for Boundary Scan tests. For normal operations, this pin must be left unconnected.
TMS	I	150	K12	ICSU1T	IOVDD	This input pin is a JTAG interface pin used for Boundary Scan tests. For normal operations, this pin must be left unconnected.
TDI	I	149	L16	ICSU1T	IOVDD	This input pin is a JTAG interface pin used for Boundary Scan tests. For normal operations, this pin must be left unconnected.
TDO	O	147	L15	OLT3	IOVDD	This output pin is a JTAG interface pin used for Boundary Scan tests. For normal operations, this pin must be left unconnected.
TRST	I	151	K13	ICSU2T	IOVDD	This input pin is a JTAG interface pin used for Boundary Scan tests. For normal operations, this pin must be left unconnected. For normal operations, this pin must be connected to RESET#.

5.3.8 Power And Ground

Table 5-10: Power And Ground Pin Descriptions

Pin Name	Type	QFP Pin#	PBGA Pin#	Cell	Description
COREVDD	P	16, 50, 80, 113, 142, 181, 206, 241	C12, D6, E1, E16, L2, M16, N6, T12	P	Core power supply
HIOVDD	P	14, 28, 40, 51	E2, G1, J4, L5	P	Power supply for the Host interface
PIO1VDD	P	72, 82, 92	T4, T6, T8	P	Power supply for the Panel 1 interface
PIO2VDD	P	101, 111, 121, 129	N9, R11, R16, T14	P	Power supply for the Panel 2 interface
SDVDD	P	165, 175, 188, 195, 205, 220	A15, B10, D14, E11, F16, H14	P	Power supply for the SDRAM interface
CM1VDD	P	234	E7	P	Power supply for the Camera1 interface
IOVDD	P	140	M14	P	Power supply for the SPI / I2S interfaces and some miscellaneous pins
VSS	P	1, 3, 15, 29, 38, 49, 64, 71, 81, 91, 102, 112, 122, 130, 141, 152, 164, 174, 180, 187, 196, 207, 209, 219, 239	A1, A16, B12, C2, C7, D11, E13, E15, F4, G13, H5, H8, H9, J8, J9, K14, L4, L13, M10, N4, N8, P6, R13, T1, T16	P	Common Ground
OSCVDD	P	250	B5	P	Power supply for OSC OSCVDD must be the same voltage as COREVDD.
OSCVSS	P	247	B6	P	Ground for OSC
PLL1VDD	P	256	B2	P	Power supply for PLL1
PLL1VSS	P	254	B3	P	Ground for PLL1
PLL2VDD	P	253	A3	P	Power supply for PLL2
PLL2VSS	P	251	B4	P	Ground for PLL2

5.4 Configuration Pins

The S1D13515/S2D13515 has three dedicated configuration pins, CNF[2:0], which should be pulled high or low based on the following table.

Table 5-11: Configuration Pin Summary

CNF[2:0]	1 (connected to VDD)	0 (connected to VSS)
CNF2	CNF[2:1] are used in combination with other host interface pins to select the host bus interface. For a summary of the possible host bus interfaces, see Section Table 5-12 :, “Host Interface Configuration Summary” on page 34.	
CNF1		
CNF0	OSCI is the source for Input Clock 1	CLKI is the source for Input Clock 1

The host bus interface is selected using a combination of the CNF[2:1] pins and host interface pins that are normally unused for the selected host bus interface.

Table 5-12 : Host Interface Configuration Summary

	MPC555/TI	8-bit/16-bit	Direct/Indirect	Parallel Type [2:0]			Serial Type	Host Interface
CNF1	**CNF2**		**CNF3**	**CNF4**	**CNF5**	**CNF6**	**CNF7**	
0	—	0	0 (TEA#)	0 (BDIP#)	0 (BURST#)	0 (AB3)	—	Indirect, 8-bit, Intel80 Type1
0	—	0	0 (TEA#)	0 (BDIP#)	0 (BURST#)	1 (AB3)	—	Indirect, 8-bit, Intel80 Type2
0	—	0	0 (TEA#)	0 (BDIP#)	1 (BURST#)	0 (AB3)	—	Reserved
0	—	0	0 (TEA#)	0 (BDIP#)	1 (BURST#)	1 (AB3)	0 (AB4)	Reserved
0	—	0	0 (TEA#)	0 (BDIP#)	1 (BURST#)	1 (AB3)	1 (AB4)	Reserved
0	—	0	0 (TEA#)	1 (BDIP#)	0 (BURST#)	0 (AB3)	—	Indirect, 8-bit, NEC V850 Type1
0	—	0	0 (TEA#)	1 (BDIP#)	0 (BURST#)	1 (AB3)	—	Indirect, 8-bit, NEC V850 Type2
0	—	0	0 (TEA#)	1 (BDIP#)	1 (BURST#)	0 (AB3)	—	Indirect, 8-bit, Renesas SH4
0	—	1	0 (TEA#)	0 (BDIP#)	0 (BURST#)	0 (AB3)	—	Indirect, 16-bit, Intel80 Type1
0	—	1	0 (TEA#)	0 (BDIP#)	0 (BURST#)	1 (AB3)	—	Indirect, 16-bit, Intel80 Type2
0	—	1	0 (TEA#)	0 (BDIP#)	1 (BURST#)	0 (AB3)	—	Reserved
0	—	1	0 (TEA#)	0 (BDIP#)	1 (BURST#)	1 (AB3)	0 (AB4)	SPI (2-stream)
0	—	1	0 (TEA#)	0 (BDIP#)	1 (BURST#)	1 (AB3)	1 (AB4)	Reserved
0	—	1	0 (TEA#)	1 (BDIP#)	0 (BURST#)	0 (AB3)	—	Indirect, 16-bit, NEC V850 Type1
0	—	1	0 (TEA#)	1 (BDIP#)	0 (BURST#)	1 (AB3)	—	Indirect, 16-bit, NEC V850 Type2
0	—	1	0 (TEA#)	1 (BDIP#)	1 (BURST#)	0 (AB3)	—	Indirect, 16-bit, Renesas SH4
0	—	0	1 (TEA#)	0 (BDIP#)	0 (BURST#)	0 (BE1#)	—	Direct, 8-bit, Intel80 Type1
0	—	0	1 (TEA#)	0 (BDIP#)	0 (BURST#)	1 (BE1#)	—	Direct, 8-bit, Intel80 Type2
0	—	0	1 (TEA#)	0 (BDIP#)	1 (BURST#)	0 (BE1#)	—	Reserved
0	—	0	1 (TEA#)	0 (BDIP#)	1 (BURST#)	1 (BE1#)	0 (AB4)	Reserved
0	—	0	1 (TEA#)	0 (BDIP#)	1 (BURST#)	1 (BE1#)	1 (AB4)	Reserved
0	—	0	1 (TEA#)	1 (BDIP#)	0 (BURST#)	0 (BE1#)	—	Direct, 8-bit, NEC V850 Type1
0	—	0	1 (TEA#)	1 (BDIP#)	0 (BURST#)	1 (BE1#)	—	Direct, 8-bit, NEC V850 Type2
0	—	0	1 (TEA#)	1 (BDIP#)	1 (BURST#)	0 (BE1#)	—	Direct, 8-bit, Renesas SH4
0	—	1	1 (TEA#)	0 (BDIP#)	0 (BURST#)	0 (AB0)	—	Direct, 16-bit, Intel80 Type1
0	—	1	1 (TEA#)	0 (BDIP#)	0 (BURST#)	1 (AB0)	—	Direct, 16-bit, Intel80 Type2
0	—	1	1 (TEA#)	0 (BDIP#)	1 (BURST#)	0 (AB0)	—	Direct, 16-bit, Marvell PXA3xx
0	—	1	1 (TEA#)	0 (BDIP#)	1 (BURST#)	1 (AB0)	0 (AB4)	SPI
0	—	1	1 (TEA#)	0 (BDIP#)	1 (BURST#)	1 (AB0)	1 (AB4)	I2C
0	—	1	1 (TEA#)	1 (BDIP#)	0 (BURST#)	0 (AB0)	—	Direct, 16-bit, NEC V850 Type1
0	—	1	1 (TEA#)	1 (BDIP#)	0 (BURST#)	1 (AB0)	—	Direct, 16-bit, NEC V850 Type2
0	—	1	1 (TEA#)	1 (BDIP#)	1 (BURST#)	0 (AB0)	—	Direct, 16-bit, Renesas SH4
1	0	—	0 (AB0)	—	—	—	—	Indirect, 16-bit, TI TMS470
1	0	—	1 (AB0)	—	—	—	—	Direct, 16-bit, TI TMS470
1	1	—	0 (BE1#)	—	—	—	—	Indirect, 16-bit, MPC555 (Little Endian only)
1	1	—	1 (BE1#)	—	—	—	—	Direct, 16-bit, MPC555 (Little Endian only)

5.5 Host Interface Pin Mapping

Table 5-13 : Host Interface Pin Mapping 1

S1D13515/ S2D13515 Pin	Intel80 Type1 8-bit Indirect	Intel80 Type2 8-bit Indirect	NEC V850 Type1 8-bit Indirect	NEC V850 Type2 8-bit Indirect	Renesas SH4 8-bit Indirect	Intel80 Type1 16-bit Indirect	Intel80 Type2 16-bit Indirect
DB15						D15	D15
DB14						D14	D14
DB13						D13	D13
DB12						D12	D12
DB11						D11	D11
DB10						D10	D10
DB9						D9	D9
DB8						D8	D8
DB7	D7	D7	D7	D7	D7	D7	D7
DB6	D6	D6	D6	D6	D6	D6	D6
DB5	D5	D5	D5	D5	D5	D5	D5
DB4	D4	D4	D4	D4	D4	D4	D4
DB3	D3	D3	D3	D3	D3	D3	D3
DB2	D2	D2	D2	D2	D2	D2	D2
DB1	D1	D1	D1	D1	D1	D1	D1
DB0	D0	D0	D0	D0	D0	D0	D0
M/R#	GPIO9/KPR0	GPIO9/KPR0	GPIO9/KPR0	GPIO9/KPR0	GPIO9/KPR0	GPIO9/KPR0	GPIO9/KPR0
AB20	GPIO10/KPR1	GPIO10/KPR1	GPIO10/KPR1	GPIO10/KPR1	GPIO10/KPR1	GPIO10/KPR1	GPIO10/KPR1
AB19	GPIO12/KPR2	GPIO12/KPR2	GPIO12/KPR2	GPIO12/KPR2	GPIO12/KPR2	GPIO12/KPR2	GPIO12/KPR2
AB18	KPR3	KPR3	KPR3	KPR3	KPR3	KPR3	KPR3
AB17	GPIO8/KPR4	GPIO8/KPR4	GPIO8/KPR4	GPIO8/KPR4	GPIO8/KPR4	GPIO8/KPR4	GPIO8/KPR4
AB16	GPIO13/KPC0	GPIO13/KPC0	GPIO13/KPC0	GPIO13/KPC0	GPIO13/KPC0	GPIO13/KPC0	GPIO13/KPC0
AB15	GPIO14/KPC1	GPIO14/KPC1	GPIO14/KPC1	GPIO14/KPC1	GPIO14/KPC1	GPIO14/KPC1	GPIO14/KPC1
AB14	GPIO15/KPC2	GPIO15/KPC2	GPIO15/KPC2	GPIO15/KPC2	GPIO15/KPC2	GPIO15/KPC2	GPIO15/KPC2
AB13	GPIO11/KPC3	GPIO11/KPC3	GPIO11/KPC3	GPIO11/KPC3	GPIO11/KPC3	GPIO11/KPC3	GPIO11/KPC3
AB12	KPC4	KPC4	KPC4	KPC4	KPC4	KPC4	KPC4
AB11	PEDST0	PEDST0	PEDST0	PEDST0	PEDST0	PEDST0	PEDST0
AB10	PEDST1	PEDST1	PEDST1	PEDST1	PEDST1	PEDST1	PEDST1
AB9	PEDST2	PEDST2	PEDST2	PEDST2	PEDST2	PEDST2	PEDST2
AB8	PEDCLK	PEDCLK	PEDCLK	PEDCLK	PEDCLK	PEDCLK	PEDCLK
AB7	PEDSIO	PEDSIO	PEDSIO	PEDSIO	PEDSIO	PEDSIO	PEDSIO
AB6	PEDCPCO	PEDCPCO	PEDCPCO	PEDCPCO	PEDCPCO	PEDCPCO	PEDCPCO
AB5							
AB4							
AB3	0 (as CNF6)	1 (as CNF6)	0 (as CNF6)	1 (as CNF6)	0 (as CNF6)	0 (as CNF6)	1 (as CNF6)
AB2						A2	A2
AB1	A1	A1	A1	A1	A1	A1	A1
AB0	A0	A0	A0	A0	A0		
BUSCLK			CLK	CLK	CLK		
BS#					BS#		
WAIT#	WAIT#	WAIT#	WAIT#	WAIT#	RDY#	WAIT#	WAIT#
RD#	RD#	RD#	DSTB#	RD#	RD#	RD#	RD#
RD/WR#	WE#		R/W#		WR#	WE#	
CS#	CS#	CS#	CS#	CS#	CS#	CS#	CS#
BE1#		1				0	WE#
BE0#		WE#		WR#		0	WE#
BURST#	0 (as CNF5)	0 (as CNF5)	0 (as CNF5)	0 (as CNF5)	1 (as CNF5)	0 (as CNF5)	0 (as CNF5)
BDIP#	0 (as CNF4)	0 (as CNF4)	1 (as CNF4)	1 (as CNF4)	1 (as CNF4)	0 (as CNF4)	0 (as CNF4)
TEA#	0 (as CNF3)	0 (as CNF3)	0 (as CNF3)	0 (as CNF3)	0 (as CNF3)	0 (as CNF3)	0 (as CNF3)
CNF2	0	0	0	0	0	1	1
CNF1	0	0	0	0	0	0	0
	= These pins select the interface.				= These pins are unused for the interface.		

Table 5-14 : Host Interface Pin Mapping 2

S1D13515/ S2D13515 Pin	SPI (2-stream)	NEC V850 Type1 16-bit Indirect	NEC V850 Type2 16-bit Indirect	Renesas SH4 16-bit Indirect	Intel80 Type1 8-bit Direct	Intel80 Type2 8-bit Direct	NEC V850 Type1 8-bit Direct
DB15	C1RIN5	D15	D15	D15			
DB14	C1GIN7	D14	D14	D14			
DB13	C1GIN6	D13	D13	D13	PEDST0	PEDST0	PEDST0
DB12	C1GIN5	D12	D12	D12	PEDST1	PEDST1	PEDST1
DB11	C1BIN7	D11	D11	D11	PEDST2	PEDST2	PEDST2
DB10	C1BIN6	D10	D10	D10	PEDCLK	PEDCLK	PEDCLK
DB9	C1BIN5	D9	D9	D9	PEDSIO	PEDSIO	PEDSIO
DB8	C1RIN4	D8	D8	D8	PEDCPCO	PEDCPCO	PEDCPCO
DB7	C1RIN3	D7	D7	D7	D7	D7	D7
DB6	C1RIN2	D6	D6	D6	D6	D6	D6
DB5	C1GIN4	D5	D5	D5	D5	D5	D5
DB4	C1GIN3	D4	D4	D4	D4	D4	D4
DB3	C1GIN2	D3	D3	D3	D3	D3	D3
DB2	C1BIN4	D2	D2	D2	D2	D2	D2
DB1	C1BIN3	D1	D1	D1	D1	D1	D1
DB0	C1BIN2	D0	D0	D0	D0	D0	D0
M/R#	GPIO9/KPR0	GPIO9/KPR0	GPIO9/KPR0	GPIO9/KPR0	M/R#	M/R#	M/R#
AB20	GPIO10/KPR1	GPIO10/KPR1	GPIO10/KPR1	GPIO10/KPR1	A20	A20	A20
AB19	GPIO12/KPR2	GPIO12/KPR2	GPIO12/KPR2	GPIO12/KPR2	A19	A19	A19
AB18	KPR3	KPR3	KPR3	KPR3	A18	A18	A18
AB17	GPIO8/KPR4	GPIO8/KPR4	GPIO8/KPR4	GPIO8/KPR4	A17	A17	A17
AB16	GPIO13/KPC0	GPIO13/KPC0	GPIO13/KPC0	GPIO13/KPC0	A16	A16	A16
AB15	GPIO14/KPC1	GPIO14/KPC1	GPIO14/KPC1	GPIO14/KPC1	A15	A15	A15
AB14	GPIO15/KPC2	GPIO15/KPC2	GPIO15/KPC2	GPIO15/KPC2	A14	A14	A14
AB13	GPIO11/KPC3	GPIO11/KPC3	GPIO11/KPC3	GPIO11/KPC3	A13	A13	A13
AB12	KPC4	KPC4	KPC4	KPC4	A12	A12	A12
AB11	PEDST0	PEDST0	PEDST0	PEDST0	A11	A11	A11
AB10	PEDST1	PEDST1	PEDST1	PEDST1	A10	A10	A10
AB9	PEDST2	PEDST2	PEDST2	PEDST2	A9	A9	A9
AB8	PEDCLK	PEDCLK	PEDCLK	PEDCLK	A8	A8	A8
AB7	PEDSIO	PEDSIO	PEDSIO	PEDSIO	A7	A7	A7
AB6	PEDCPCO	PEDCPCO	PEDCPCO	PEDCPCO	A6	A6	A6
AB5	SPICLKSEL				A5	A5	A5
AB4	0 (as CNF7)				A4	A4	A4
AB3	1 (as CNF6)	0 (as CNF6)	1 (as CNF6)	0 (as CNF6)	A3	A3	A3
AB2	C1HSIN	A2	A2	A2	A2	A2	A2
AB1	C1VSIN	A1	A1	A1	A1	A1	A1
AB0	C1DEIN				A0	A0	A0
BUSCLK		CLK	CLK	CLK			CLK
BS#	C1PCLKIN			BS#			
WAIT#	HSDO	WAIT#	WAIT#	RDY#	WAIT#	WAIT#	WAIT#
RD#	C1RIN7	DSTB#	RD#	RD#	RD#	RD#	DSTB#
RD/WR#	HSDI	R/W#			WE#		R/W#
CS#	HSCS#	CS#	CS#	CS#	CS#	CS#	CS#
BE1#	C1RIN6	0	WR#	WR#	0 (as CNF6)	1 (as CNF6)	0 (as CNF6)
BE0#	HSCK	0	WR#	WR#		WE#	
BURST#	1 (as CNF5)	0 (as CNF5)	0 (as CNF5)	1 (as CNF5)	0 (as CNF5)	0 (as CNF5)	0 (as CNF5)
BDIP#	0 (as CNF4)	1 (as CNF4)	1 (as CNF4)	1 (as CNF4)	0 (as CNF4)	0 (as CNF4)	1 (as CNF4)
TEA#	0 (as CNF3)	0 (as CNF3)	0 (as CNF3)	0 (as CNF3)	1 (as CNF3)	1 (as CNF3)	1 (as CNF3)
CNF2	1	1	1	1	0	0	0
CNF1	0	0	0	0	0	0	0
	= These pins select the interface.				= These pins are unused for the interface.		

Table 5-15 : Host Interface Pin Mapping 3

S1D13515/ S2D13515 Pin	NEC V850 Type2 8-bit Direct	Renesas SH4 8-bit Direct	Intel80 Type1 16-bit Direct	Intel80 Type2 16-bit Direct	Marvell PXA3xx 16-bit Direct	SPI	I2C
DB15			D15	D15	DF_IO15		
DB14			D14	D14	DF_IO14		
DB13	PEDST0	PEDST0	D13	D13	DF_IO13		
DB12	PEDST1	PEDST1	D12	D12	DF_IO12		
DB11	PEDST2	PEDST2	D11	D11	DF_IO11		
DB10	PEDCLK	PEDCLK	D10	D10	DF_IO10		
DB9	PEDSIO	PEDSIO	D9	D9	DF_IO9		
DB8	PEDCPCO	PEDCPCO	D8	D8	DF_IO8		
DB7	D7	D7	D7	D7	DF_IO7		
DB6	D6	D6	D6	D6	DF_IO6		SLADDR6
DB5	D5	D5	D5	D5	DF_IO5		SLADDR5
DB4	D4	D4	D4	D4	DF_IO4		SLADDR4
DB3	D3	D3	D3	D3	DF_IO3		SLADDR3
DB2	D2	D2	D2	D2	DF_IO2		SLADDR2
DB1	D1	D1	D1	D1	DF_IO1		SLADDR1
DB0	D0	D0	D0	D0	DF_IO0		SLADDR0
M/R#	M/R#	M/R#	M/R#	M/R#	GPIO9/KPR0	GPIO9/KPR0	GPIO9/KPR0
AB20	A20	A20	A20	A20	GPIO10/KPR1	GPIO10/KPR1	GPIO10/KPR1
AB19	A19	A19	A19	A19	GPIO12/KPR2	GPIO12/KPR2	GPIO12/KPR2
AB18	A18	A18	A18	A18	KPR3	KPR3	KPR3
AB17	A17	A17	A17	A17	GPIO8/KPR4	GPIO8/KPR4	GPIO8/KPR4
AB16	A16	A16	A16	A16	GPIO13/KPC0	GPIO13/KPC0	GPIO13/KPC0
AB15	A15	A15	A15	A15	GPIO14/KPC1	GPIO14/KPC1	GPIO14/KPC1
AB14	A14	A14	A14	A14	GPIO15/KPC2	GPIO15/KPC2	GPIO15/KPC2
AB13	A13	A13	A13	A13	GPIO11/KPC3	GPIO11/KPC3	GPIO11/KPC3
AB12	A12	A12	A12	A12	KPC4	KPC4	KPC4
AB11	A11	A11	A11	A11	PEDST0	PEDST0	PEDST0
AB10	A10	A10	A10	A10	PEDST1	PEDST1	PEDST1
AB9	A9	A9	A9	A9	PEDST2	PEDST2	PEDST2
AB8	A8	A8	A8	A8	PEDCLK	PEDCLK	PEDCLK
AB7	A7	A7	A7	A7	PEDSIO	PEDSIO	PEDSIO
AB6	A6	A6	A6	A6	nLUA	PEDCPCO	PEDCPCO
AB5	A5	A5	A5	A5	nLLA	SPICLKSEL	I2CCLKSEL
AB4	A4	A4	A4	A4	DF_ADDR3	0 (as CNF7)	1 (as CNF7)
AB3	A3	A3	A3	A3	DF_ADDR2		
AB2	A2	A2	A2	A2	DF_ADDR1		
AB1	A1	A1	A1	A1	DF_ADDR0		
AB0	A0	A0	0 (as CNF6)	1 (as CNF6)	0 (as CNF6)	1 (as CNF6)	1 (as CNF6)
BUSCLK	CLK	CLK					
BS#		BS#			PEDCPCO		HSDA
WAIT#	WAIT#	RDY#	WAIT#	WAIT#	RDY	HSDO	
RD#	RD#	RD#	RD#	RD#	DF_nOE		
RD/WR#		WR#	WE#		DF_nWE	HSDI	
CS#	CS#	CS#	CS#	CS#	CS#	HSCS#	
BE1#	1 (as CNF6)	0 (as CNF6)	UBE#	WEU#	nBE1		
BE0#	WR#		ULE#	WEL#	nBE0	HSCK	HSCL
BURST#	0 (as CNF5)	1 (as CNF5)	0 (as CNF5)	0 (as CNF5)	1 (as CNF5)	1 (as CNF5)	1 (as CNF5)
BDIP#	1 (as CNF4)	1 (as CNF4)	0 (as CNF4)	0 (as CNF4)	0 (as CNF4)	0 (as CNF4)	0 (as CNF4)
TEA#	1 (as CNF3)	1 (as CNF3)	1 (as CNF3)	1 (as CNF3)	1 (as CNF3)	1 (as CNF3)	1 (as CNF3)
CNF2	0	0	1	1	1	1	1
CNF1	0	0	0	0	0	0	0
	= These pins select the interface.				= These pins are unused for the interface.		

Note

The I2C slave address configuration from DB[6:0] is latched on RESET#. Reserved I2C slave addresses are not supported. See Section 21.11, “I2C Host Interface” on page 533 for information. Any changes to the I2C Slave Address requires a hardware RESET#.

Table 5-16: Host Interface Pin Mapping 4

S1D13515/ S2D13515 Pin	NEC V850 Type1 16-bit Direct	NEC V850 Type2 16-bit Direct	Renesas SH4 16-bit Direct	TI TMS470 16-bit Indirect	TI TMS470 16-bit Direct	MPC555 16-bit Indirect Little Endian	MPC555 16-bit Direct Little Endian
DB15	D15	D15	D15	D15	D15	D0	D0
DB14	D14	D14	D14	D14	D14	D1	D1
DB13	D13	D13	D13	D13	D13	D2	D2
DB12	D12	D12	D12	D12	D12	D3	D3
DB11	D11	D11	D11	D11	D11	D4	D4
DB10	D10	D10	D10	D10	D10	D5	D5
DB9	D9	D9	D9	D9	D9	D6	D6
DB8	D8	D8	D8	D8	D8	D7	D7
DB7	D7	D7	D7	D7	D7	D8	D8
DB6	D6	D6	D6	D6	D6	D9	D9
DB5	D5	D5	D5	D5	D5	D10	D10
DB4	D4	D4	D4	D4	D4	D11	D11
DB3	D3	D3	D3	D3	D3	D12	D12
DB2	D2	D2	D2	D2	D2	D13	D13
DB1	D1	D1	D1	D1	D1	D14	D14
DB0	D0	D0	D0	D0	D0	D15	D15
M/R#	M/R#	M/R#	M/R#	GPIO9/KPR0	M/R#	GPIO9/KPR0	M/R#
AB20	A20	A20	A20	GPIO10/KPR1		GPIO10/KPR1	A11
AB19	A19	A19	A19	GPIO12/KPR2	A19	GPIO12/KPR2	A12
AB18	A18	A18	A18	KPR3	A18	KPR3	A13
AB17	A17	A17	A17	GPIO8/KPR4	A17	GPIO8/KPR4	A14
AB16	A16	A16	A16	GPIO13/KPC0	A16	GPIO13/KPC0	A15
AB15	A15	A15	A15	GPIO14/KPC1	A15	GPIO14/KPC1	A16
AB14	A14	A14	A14	GPIO15/KPC2	A14	GPIO15/KPC2	A17
AB13	A13	A13	A13	GPIO11/KPC3	A13	GPIO11/KPC3	A18
AB12	A12	A12	A12	KPC4	A12	KPC4	A19
AB11	A11	A11	A11	PEDST0	A11	PEDST0	A20
AB10	A10	A10	A10	PEDST1	A10	PEDST1	A21
AB9	A9	A9	A9	PEDST2	A9	PEDST2	A22
AB8	A8	A8	A8	PEDCLK	A8	PEDCLK	A23
AB7	A7	A7	A7	PEDSIO	A7	PEDSIO	A24
AB6	A6	A6	A6	PEDCPCO	A6	PEDCPCO	A25
AB5	A5	A5	A5		A5		A26
AB4	A4	A4	A4		A4		A27
AB3	A3	A3	A3		A3		A28
AB2	A2	A2	A2	A2	A2	A29	A29
AB1	A1	A1	A1	A1	A1	A30	A30
AB0	0 (as CNF6)	1 (as CNF6)	0 (as CNF6)	0 (as CNF3)	1 (as CNF3)		A31
BUSCLK	CLK	CLK	CLK	CLK	CLK	CLK	CLK
WAIT#	WAIT#	WAIT#	RDY#	TA#	TA#	TA#	TA#
RD#	DSTB#	RD#	RD#	OE#	OE#	1	TSIZ0
RD/WR#	R/W#			RD/WR#	RD/WR#	RD/WR#	RD/WR#
CS#	CS#	CS#	CS#	CS#	CS#	CS#	CS#
BE1#	UBEN#	WRH#	WE1#	0	UB#	0 (as CNF3)	1 (as CNF3)
BE0#	LBEN#	WRL#	WE0#	0	LB#	0	TSIZ1
BS#			BS#	TS#	TS#	TS#	TS#
BURST#	0 (as CNF5)	0 (as CNF5)	1 (as CNF5)	BURST#	BURST#	BURST#	BURST#
BDIP#	1 (as CNF4)	1 (as CNF4)	1 (as CNF4)	BDIP#	BDIP#	BDIP#	BDIP#
TEA#	1 (as CNF3)	1 (as CNF3)	1 (as CNF3)	ERR_ACK#	ERR_ACK#	TEA#	TEA#
CNF2	1	1	1	0	0	1	1
CNF1	0	0	0	1	1	1	1
	= These pins select the interface.				= These pins are unused for the interface.		

5.6 LCD / Camera2 Pin Mapping

The primary use for the FP1IO[23:0] pins is for the LCD1 interface or Camera2 interface. However, these pins may also be used for an EID Double Screen panel on LCD2, Keypad interface, or GPIOs. In these cases, the pin mapping for each interface changes as shown in the following table.

Table 5-17: FP1IO Pin Mapping Summary (LCD1 / Camera2)

S1D13515/ S2D13515 Pin	Generic TFT/TFD on LCD1 (REG[4000h] bit 3 = 0b)				Camera2 Interface (REG[4000h] bit 3 = 1b)			
	LCD2 does not use any FP1 Pins (see Note 1)		LCD2 uses FP1 Pins (see Note 2)		LCD2 does not use any FP1 Pins (see Note 1)		LCD2 uses FP1 Pins (see Note 2)	
	RGB 6:6:6 (REG[4000h] bit 2 = 0b)	RGB 5:6:5 with SCI (REG[4000h] bit 2 = 1b)	RGB 5:5:5 (REG[4000h] bit 2 = 0b)	RGB 4:4:4 with SCI REG[4000h] bit 2 = 1b	8-bit Camera (REG[0D46h] bit 2 = 0b)	RGB Data Stream (REG[0D46h] bit 2 = 1b)	8-bit Camera (REG[0D46h] bit 2 = 0b)	RGB Data Stream (REG[0D46h] bit 2 = 1b)
FP1 Mode	0	1	2	3	5	4	7	6
FP1IO0	R7	R7	R7	R7	CM2DAT0	C2RIN7	CM2DAT0	C2RIN7
FP1IO1	R6	R6	R6	R6	CM2DAT1	C2RIN6	CM2DAT1	C2RIN6
FP1IO2	R5	R5	R5	R5	CM2DAT2	C2RIN5	CM2DAT2	C2RIN5
FP1IO3	G7	G7	G7	G7	CM2DAT3	C2GIN7	CM2DAT3	C2GIN7
FP1IO4	G6	G6	G6	G6	CM2DAT4	C2GIN6	CM2DAT4	C2GIN6
FP1IO5	G5	G5	G5	G5	CM2DAT5	C2GIN5	CM2DAT5	C2GIN5
FP1IO6	B7	B7	B7	B7	CM2DAT6	C2BIN7	CM2DAT6	C2BIN7
FP1IO7	B6	B6	B6	B6	CM2DAT7	C2BIN6	CM2DAT7	C2BIN6
FP1IO8	B5	B5	B5	B5	CM2CLKIN	C2BIN5	CM2CLKIN	C2BIN5
FP1IO9	R4	R4	R4	R4	CM2CLKOUT	C2RIN4	CM2CLKOUT	C2RIN4
FP1IO10	R3	R3	R3	SCS	CM2FIELD	C2RIN3	CM2FIELD	C2RIN3
FP1IO11	R2	SCS	POLGMA	POLGMA	KPR3	C2RIN2	POLGMA	POLGMA
FP1IO12	G4	G4	G4	G4	CM2VREF	C2GIN4	CM2VREF	C2GIN4
FP1IO13	G3	G3	G3	SCK	CM2HREF	C2GIN3	CM2HREF	C2GIN3
FP1IO14	G2	G2	DEXR	DEXR	KPR4/GPIO8	C2GIN2	DEXR	DEXR
FP1IO15	B4	B4	B4	B4	KPR0/GPIO9	C2BIN4	KPR0/GPIO9	C2BIN4
FP1IO16	B3	B3	B3	SDO	KPR1/GPIO10	C2BIN3	KPR1/GPIO10	C2BIN3
FP1IO17	B2	SCK	CPV	CPV	KPC3/GPIO11	C2BIN2	CPV	CPV
FP1IO18	GPIO6	SDA0	OE	OE	GPIO6	GPIO6	OE	OE
FP1IO19	GPIO7 (Note 3)	SDO	LED_DIM_OUT	LED_DIM_OUT	KPC4/GPIO7	GPIO7	LED_DIM_OUT	LED_DIM_OUT
FP1IO20	HSYNC	HSYNC	HSYNC	HSYNC	KPR2/GPIO12	C2HSIN	KPR2/GPIO12	C2HSIN
FP1IO21	VSYNC	VSYNC	VSYNC	VSYNC	KPC0/GPIO13	C2VSIN	KPC0/GPIO13	C2VSIN
FP1IO22	DE	DE	DE	DE	KPC1/GPIO14	C2DEIN	KPC1/GPIO14	C2DEIN
FP1IO23	PCLK	PCLK	PCLK	PCLK	KPC2/GPIO15	C2PCLKIN	KPC2/GPIO15	C2PCLKIN

1. This pin mapping applies when:
   - LCD2 is not an EID Double Screen panel (REG[4000h] bits 5-4 = 00b or 10b)
   - LCD2 is an EID Double Screen panel with TCON Disabled (REG[4000h] bits 5-4 = 01b and REG[4040h] bit 0 = 0b)
   - LCD2 is an EID Double Screen panel with TCON Enabled on the I2S pins ([REG[4000h] bits 5-4 = 01b and REG[4040h] bit 0 = 1b] and REG[4000h] bit 1 = 1b)
2. This pin mapping applies when:
   - LCD2 is an EID Double Screen panel with TCON Enabled on the FP1 pins ([REG[4000h] bits 5-4 = 01b and REG[4040h] bit 0 = 1b] and REG[4000h] bit 1 = 0b)
3. GPIO7 is not available when the Keypad Interface is configured to use the FP1IO pins, REG[0186h] bit 5 = 1b.

The FP2IO[27:0] pins are used for the LCD2 interface. When the LCD2 interface is configured for a generic TFT/TFD, EID Double Screen with TCON disabled (REG[4040h] bit 0 = 0b), or Sharp DualView panel, all LCD2 pins can be mapped to the FP2IO[27:0] pins. However, when LCD2 is configured for a EID Double Screen with TCON enabled (REG[4040h] bit 0 = 1b), additional pins are required and must be selected from either the FP1IO pins or the I2S/PWM1 pins. The following table summarizes the possible FP2IO pin mappings.

Table 5-18: FP2IO Pin Mapping Summary (LCD2)

S1D13515/ S2D13515 Pin	**Generic RGB or EID Double Screen with TCON Disabled[2] (REG[4000h] bits 5-4 = 00b) or (REG[4000h] bits 5-4 = 01b and REG[4040h] bit 0 = 0b)**			**EID Double Screen with TCON Enabled on FP1 (REG[4000h] bits 5-4 = 01b and REG[4040h] bit 0 = 1b) and REG[4000h] bit 1 = 0b**	**EID Double Screen with TCON Enabled on I2S (REG[4000h] bits 5-4 = 01b and REG[4040h] bit 0 = 1b) and REG[4000h] bit 1 = 1b**	**Sharp DualView (REG[4000h] bits 5-4 = 10b)**
	RGB 8:8:8 no SCI (REG[4000h] bits 7-6 = 00b)	RGB 6:6:6 with SCI (REG[4000h] bits 7-6 = 01b)	RGB 6:6:6 no SCI (REG[4000h] bits 7-6 = 10b)			
FP2 Mode	0	1	2	3	3	4
FP2IO0	R7	R7	R7	R7	R7	R7
FP2IO1	R6	R6	R6	R6	R6	R6
FP2IO2	R5	R5	R5	R5	R5	R5
FP2IO3	G7	G7	G7	G7	G7	G7
FP2IO4	G6	G6	G6	G6	G6	G6
FP2IO5	G5	G5	G5	G5	G5	G5
FP2IO6	B7	B7	B7	B7	B7	B7
FP2IO7	B6	B6	B6	B6	B6	B6
FP2IO8	B5	B5	B5	B5	B5	B5
FP2IO9	R4	R4	R4	R4	R4	R4
FP2IO10	R3 / PEDST0[1]	R3 / PEDST0[1]	R3 / PEDST0[1]	R3 / PEDST0[1]	R3 / PEDST0[1]	R3 / PEDST0[1]
FP2IO11	R2 / PEDST1[1]	R2 / PEDST1[1]	R2 / PEDST1[1]	R2 / PEDST1[1]	R2 / PEDST1[1]	R2 / PEDST1[1]
FP2IO12	G4	G4	G4	G4	G4	G4
FP2IO13	G3 / PEDST2[1]	G3 / PEDST2[1]	G3 / PEDST2[1]	G3 / PEDST2[1]	G3 / PEDST2[1]	G3 / PEDST2[1]
FP2IO14	G2 / PEDCLK[1]	G2 / PEDCLK[1]	G2 / PEDCLK[1]	G2 / PEDCLK[1]	G2 / PEDCLK[1]	G2 / PEDCLK[1]
FP2IO15	B4	B4	B4	B4	B4	B4
FP2IO16	B3 / PEDCPCO[1]	B3 / PEDCPCO[1]	B3 / PEDCPCO[1]	B3 / PEDCPCO[1]	B3 / PEDCPCO[1]	B3 / PEDCPCO[1]
FP2IO17	B2 / PEDSIO[1]	B2 / PEDSIO[1]	B2 / PEDSIO[1]	B2 / PEDSIO[1]	B2 / PEDSIO[1]	B2 / PEDSIO[1]
FP2IO18	R1	SCS	GPIO0	ONA	ONA	VCOM
FP2IO19	R0	SCK	GPIO1	ONB	ONB	VCOMB
FP2IO20	G1	SDA0	GPIO2	ONC	ONC	SPR
FP2IO21	G0	SDO	GPIO3	OND	OND	SPL
FP2IO22	B1	GPIO4	GPIO4	VREVOUT	VREVOUT	GPIO4
FP2IO23	B0	GPIO5	GPIO5	HREVOUT	HREVOUT	GPIO5
FP2IO24	HSYNC	HSYNC	OHSYNC	EISF	EISF	LS
FP2IO25	VSYNC	VSYNC	OVSYNC	FLMF	FLMF	SPS
FP2IO26	DE	DE	ODE	STRB	STRB	CLS
FP2IO27	PCLK	PCLK	ODCK	ODCK	ODCK	CK
FP1IO11	—			POLGMA	—	—
FP1IO14	—	—	—	DEXR	—	—
FP1IO17	—	—	—	CPV	—	—
FP1IO18	—	—	—	OE	—	—
FP1IO19	—	—	—	LED_DIM_OUT	—	—
WSIO	—	—	—	—	POLGMA	—
SCKIO	—	—	—	—	DEXR	—
SDO	—	—	—	—	CPV	—
MCLKO	—	—	—	—	OE	—
PWM1	—	—	—	—	LED_DIM_OUT	—

1. These pins are used for the C33PE debugger interface (PED*) if REG[008Ah] bit 1 is 1b, the Host Interface selected is Direct 16-bit, and the Host Interface selected is not Marvell PXA3xx Direct 16-bit.
2. When LCD2 is an EID Doublescreen with TCON disabled, FP2IO[23:18] is driven LOW.

Chapter 6 D.C. Characteristics

Note
1. When applying supply voltages to the S1D13515/S2D13515, Core V_{DD} must be applied to the chip before, or simultaneously with H V_{DD}, or damage to the chip may result.
2. Core V_{DD}, OSC V_{DD}, and PLL V_{DD} must be equal to or lower than H V_{DD}.

Table 6-1: Absolute Maximum Ratings

Symbol	Parameter	Rating	Units
Core V_{DD}	Supply Voltage	V_{SS} - 0.3 to 2.5	V
H V_{DD}	Supply Voltage	V_{SS} - 0.3 to 4.0	V
OSC V_{DD}	Supply Voltage	V_{SS} - 0.3 to 2.1	V
PLL V_{DD}	Supply Voltage	V_{SS} - 0.3 to 2.1	V
V_{IN}	Input Voltage	V_{SS} - 0.3 to H V_{DD} + 0.5	V
V_{OUT}	Output Voltage	V_{SS} - 0.3 to H V_{DD} + 0.5	V
T_{STG}	Storage Temperature	-65 to 150	°C

Table 6-2 : Recommended Operating Conditions 1

Symbol	Parameter	Condition	Min	Typ	Max	Units
Core V_{DD}	Supply Voltage	V_{SS} = 0 V	1.65	1.8	1.95	V
H V_{DD-HIO}	Supply Voltage	V_{SS} = 0 V	3.0	3.3	3.6	V
H $V_{DD-PIO1}$	Supply Voltage	V_{SS} = 0 V	3.0	3.3	3.6	V
H $V_{DD-PIO2}$	Supply Voltage	V_{SS} = 0 V	3.0	3.3	3.6	V
H V_{DD-SD}	Supply Voltage	V_{SS} = 0 V	3.0	3.3	3.6	V
H V_{DD-CM1}	Supply Voltage	V_{SS} = 0 V	3.0	3.3	3.6	V
H V_{DD-IO}	Supply Voltage	V_{SS} = 0 V	3.0	3.3	3.6	V
OSC V_{DD}	Supply Voltage (note)	V_{SS} = 0 V	1.65	1.8	1.95	V
PLL1 V_{DD}	Supply Voltage	V_{SS} = 0 V	1.65	1.8	1.95	V
PLL2 V_{DD}	Supply Voltage	V_{SS} = 0 V	1.65	1.8	1.95	V
V_{IN}	Input Voltage		V_{SS}	—	Core V_{DD}	V
			V_{SS}	—	IO V_{DD}	V
T_{OPR}	Operating Temperature	S1D13515	-40	25	85	° C
		S2D13515	-40	25	105	° C

Note
OSC V_{DD} must be the same voltage as CORE V_{DD}.

Table 6-3 : Recommended Operating Conditions 2

Symbol	Parameter	Condition	Min	Typ	Max	Units
Core V_{DD}	Supply Voltage	V_{SS} = 0 V	1.65	1.8	1.95	V
H V_{DD-HIO}	Supply Voltage	V_{SS} = 0 V	2.3	2.5	2.7	V
H $V_{DD-PIO1}$	Supply Voltage	V_{SS} = 0 V	2.3	2.5	2.7	V
H $V_{DD-PIO2}$	Supply Voltage	V_{SS} = 0 V	2.3	2.5	2.7	V
H V_{DD-SD}	Supply Voltage	V_{SS} = 0 V	3.0	3.3	3.6	V
H V_{DD-CM1}	Supply Voltage	V_{SS} = 0 V	2.3	2.5	2.7	V
H V_{DD-IO}	Supply Voltage	V_{SS} = 0 V	2.3	2.5	2.7	V
OSC V_{DD}	Supply Voltage	V_{SS} = 0 V	1.65	1.8	1.95	V
PLL1 V_{DD}	Supply Voltage	V_{SS} = 0 V	1.65	1.8	1.95	V
PLL2 V_{DD}	Supply Voltage	V_{SS} = 0 V	1.65	1.8	1.95	V
V_{IN}	Input Voltage		V_{SS}	—	Core V_{DD}	V
			V_{SS}	—	IO V_{DD}	V
T_{OPR}	Operating Temperature	S1D13515	-40	25	85	° C
		S2D13515	-40	25	105	° C

Table 6-4: Electrical Characteristics for VDD = 3.3V typical

Symbol	Parameter	Condition	Min	Typ	Max	Units
I_{DDS}	Quiescent Current	Quiescent Conditions		23		µA
I_{IZ}	Input Leakage Current	V_I = 0V or V_{DD}	-5	—	5	µA
I_{OZ}	Output Leakage Current	V_O = 0V or V_{DD}	-5	—	5	µA
I_{OH2}	High Level Output Current	V_{OH} = H V_{DD} - 0.4V	-4	—	—	mA
I_{OH3}	High Level Output Current	H V_{DD} = min	-8	—	—	mA
I_{OL2}	Low Level Output Current	V_{OL} = 0.4V	4	—	—	mA
I_{OL3}	Low Level Output Current	H V_{DD} = min	8	—	—	mA
V_{IH}	High Level Input Voltage	LVCMOS level, H V_{DD} = max	2.2	—	H V_{DD} + 0.3	V
V_{IL}	Low Level Input Voltage	LVCMOS level, H V_{DD} = min	-0.3	—	0.8	V
VT_+	Positive Trigger Voltage	LVCMOS Schmitt	1.4	—	2.7	V
VT_-	Negative Trigger Voltage	LVCMOS Schmitt	0.6	—	1.8	V
V_H	Hysteresis Voltage	LVCMOS Schmitt	0.3	—	—	V
R_{PU}	Pull-up Resistance	V_I = 0V, Type 1	25	50	120	kΩ
		V_I = 0V, Type 2	50	100	240	kΩ
R_{PD}	Pull-down Resistance	V_I = H V_{DD}, Type 1	25	50	120	kΩ
		V_I = H V_{DD}, Type 2	50	100	240	kΩ
C_I	Input Pin Capacitance	F = 1MHz, H V_{DD} = 0V	—	—	8	pF
C_O	Output Pin Capacitance	F = 1MHz, H V_{DD} = 0V	—	—	8	pF
C_{IO}	Bi-Directional Pin Capacitance	F = 1MHz, H V_{DD} = 0V	—	—	8	pF

Table 6-5: Electrical Characteristics for VDD = 2.5V typical

Symbol	Parameter	Condition	Min	Typ	Max	Units
I_{DDS}	Quiescent Current	Quiescent Conditions		23		μA
I_{IZ}	Input Leakage Current	V_I = 0V or V_{DD}	-5	—	5	μA
I_{OZ}	Output Leakage Current	V_O = 0V or V_{DD}	-5	—	5	μA
I_{OH2}	High Level Output Current	V_{OH} = H V_{DD} - 0.4V	-3	—	—	mA
I_{OH3}	High Level Output Current	H V_{DD} = min	-6	—	—	mA
I_{OL2}	Low Level Output Current	V_{OL} = 0.4V	3	—	—	mA
I_{OL3}	Low Level Output Current	H V_{DD} = min	6	—	—	mA
V_{IH}	High Level Input Voltage	LVCMOS level, H V_{DD} = max	1.7	—	H V_{DD} + 0.2	V
V_{IL}	Low Level Input Voltage	LVCMOS level, H V_{DD} = min	-0.2	—	0.7	V
VT_+	Positive Trigger Voltage	LVCMOS Schmitt	0.8	—	1.9	V
VT_-	Negative Trigger Voltage	LVCMOS Schmitt	0.5	—	1.3	V
V_H	Hysteresis Voltage	LVCMOS Schmitt	0.3	—	—	V
R_{PU}	Pull-up Resistance	V_I = 0V, Type 1	35	70	175	kΩ
		V_I = 0V, Type 2	70	140	350	kΩ
R_{PD}	Pull-down Resistance	V_I = H V_{DD}, Type 1	35	70	175	kΩ
		V_I = H V_{DD}, Type 2	70	140	350	kΩ
C_I	Input Pin Capacitance	F = 1MHz, H V_{DD} = 0V	—	—	8	pF
C_O	Output Pin Capacitance	F = 1MHz, H V_{DD} = 0V	—	—	8	pF
C_{IO}	Bi-Directional Pin Capacitance	F = 1MHz, H V_{DD} = 0V	—	—	8	pF

Chapter 7 A.C. Characteristics

Conditions: IO V_{DD} = 3.3V +/- 10%
Core V_{DD} = 1.8V +/- 10%
T_A = -40 to 85°C for the S1D13515
-40 to 105°C for the S2D13515
T_{rise} and T_{fall} for all inputs must be ≤ 5 ns (10% ~ 90%)
C_L = 30 pF, except for the Host Interface (50 pF) and the SDRAM Interface (15 pF)

7.1 Clock Timing

7.1.1 Input Clocks

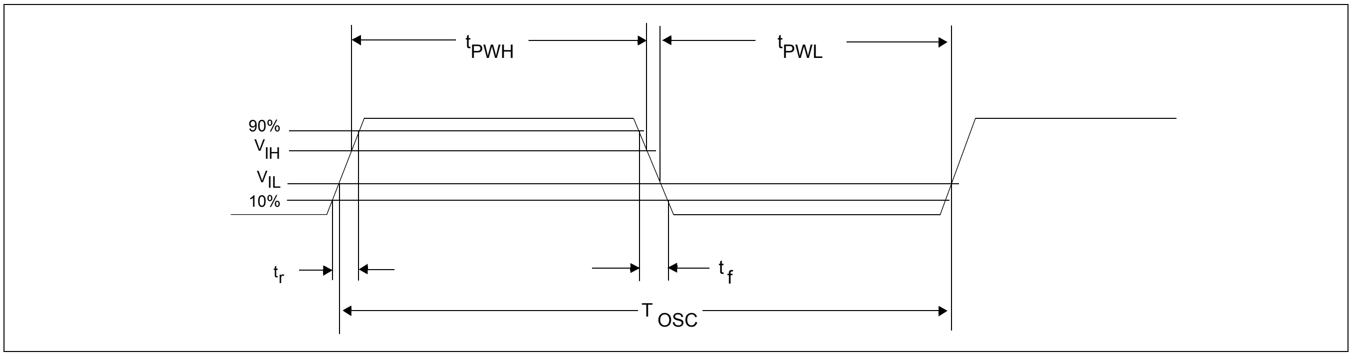


Figure 7-1: Clock Requirements for OSC/CLKI

Table 7-1: Clock Requirements for OSC/CLKI when used as Clock Input

Symbol	Parameter	Min	Typ	Max	Units
f_{OSC}	Input Clock Frequency for OSC	20	—	40	MHz
T_{OSC}	Input Clock Period for OSC	—	$1/f_{OSC}$	—	ns
f_{CLKI}	Input Clock Frequency for CLKI	5	—	100	MHz
T_{CLKI}	Input Clock Period for CLKI	—	$1/f_{CLKI}$	—	ns
t_{PWH}	Input Clock Pulse Width High	0.4	—	0.6	T_{OSC}
t_{PWL}	Input Clock Pulse Width Low	0.4	—	0.6	T_{OSC}
t_f	Input Clock Fall Time (10% - 90%)	—	—	0.2	T_{OSC}
t_r	Input Clock Rise Time (10% - 90%)	—	—	0.2	T_{OSC}
t_{jitter}	Input Clock Jitter	-150	—	150	ps

Table 7-2: Clock Requirements for OSC when used as Crystal Oscillator Input

Symbol	Parameter	Min	Typ	Max	Units
f_{OSC}	Input Clock Frequency for OSC	20	—	40	MHz
T_{OSC}	Input Clock Period for OSC	—	$1/f_{OSC}$	—	ns

7.1.2 Internal Clocks

Table 7-3: Internal Clock Requirements

Symbol	Parameter	Min	Max	Units
$f_{SDRAMCLK}$	SDRAM Clock Frequency	—	100	MHz
f_{SYSCLK}	System Clock Frequency	—	50	MHz

For further information on the internal clocks, refer to Section Chapter 9, “Clocks” on page 131.

Note
For XGA 1024x768 panel support, the DRAMCLK must be 100MHz. See Chapter 13, “Display Subsystem” on page 448 for further information.

7.1.3 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.

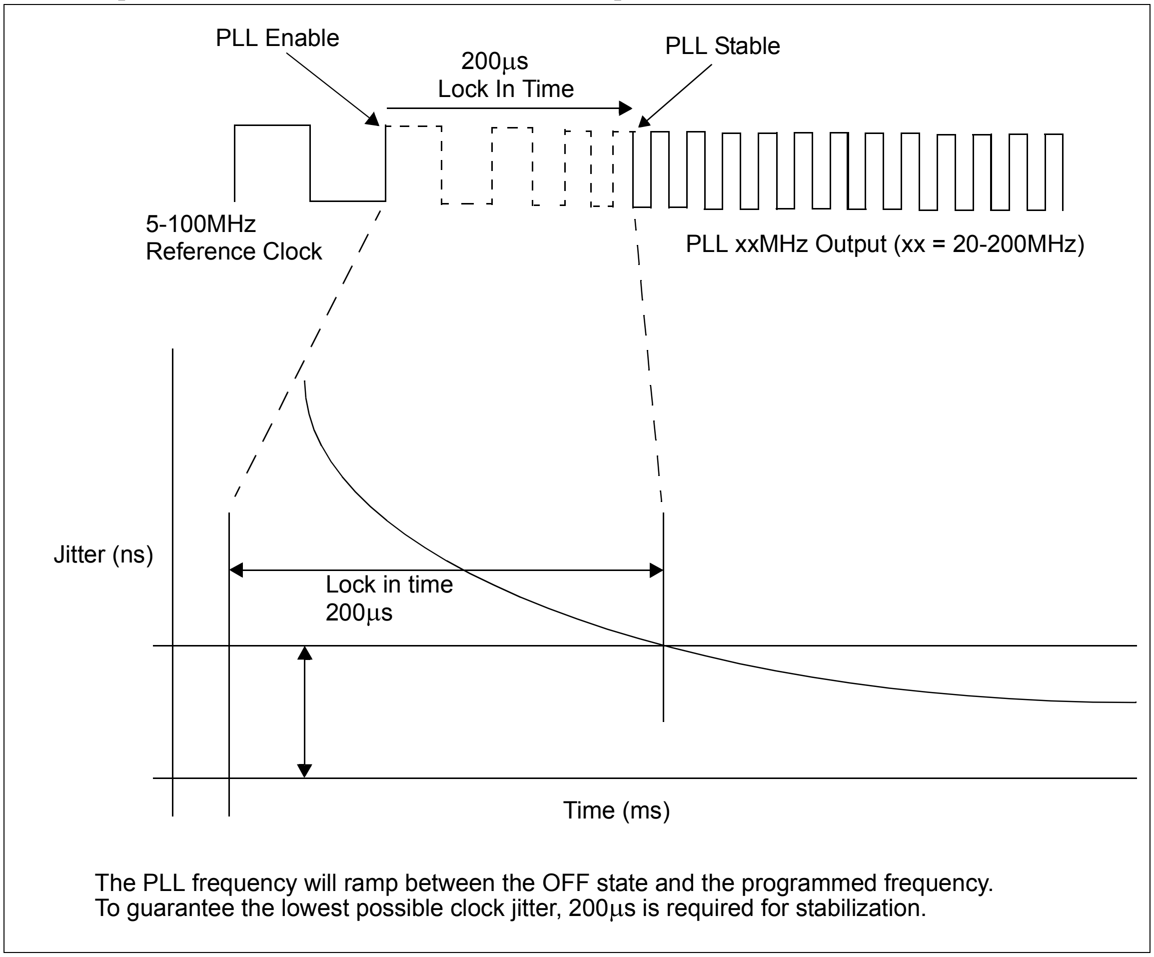


Figure 7-2: PLL Start-Up Time

Table 7-4: PLL Clock Requirements

Symbol	Parameter	Min	Max	Units
f_{PLL}	PLL output clock frequency	20	200	MHz
t_{PStal}	PLL output stable time	—	200	µs

7.2 Power Supply Sequence

7.2.1 Power Supply Structure

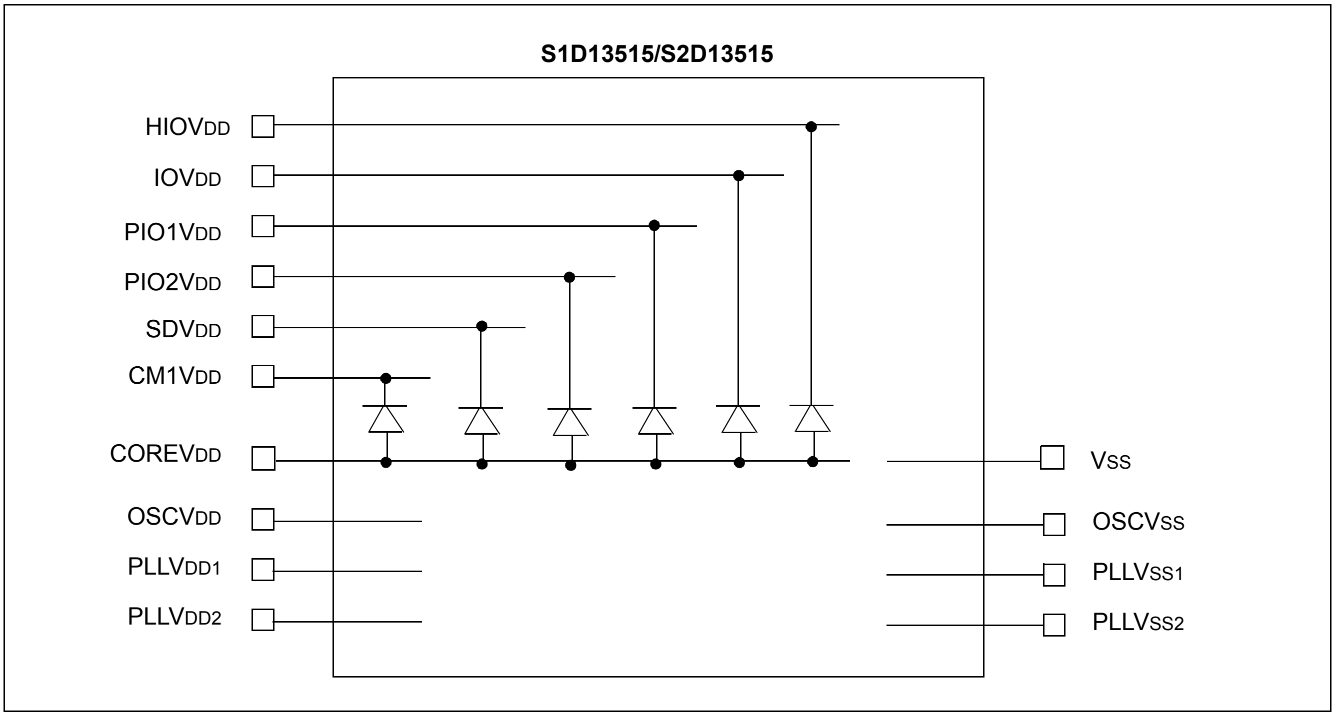


Figure 7-3: Internal Power Structure

7.2.2 Power-On Sequence

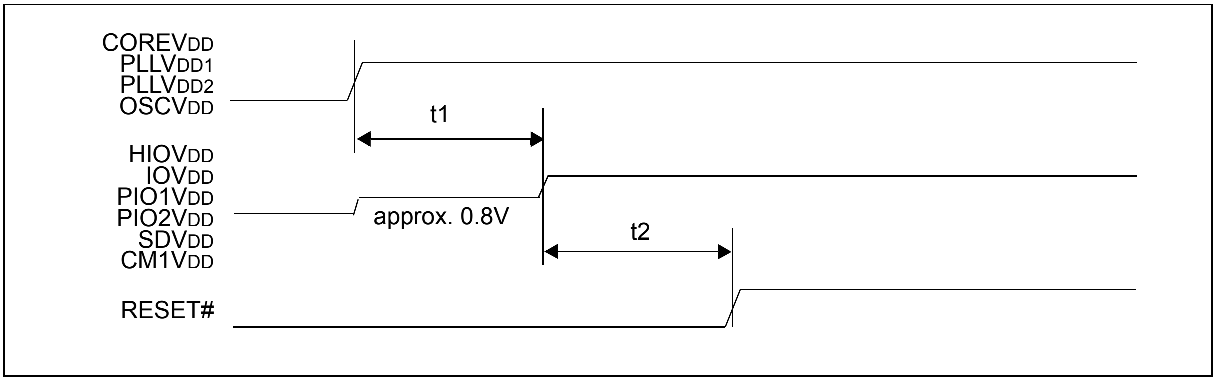


Figure 7-4: Power-On Sequence

Table 7-5: Power-On Sequence

Symbol	Parameter	Min	Max	Units
t1	$HIOV_{DD}$, IOV_{DD}, $PIO1V_{DD}$, $PIO2V_{DD}$, SDV_{DD}, $CM1V_{DD}$ on delay from $COREV_{DD}$, $OSCV_{DD}$, $PLLV_{DD1}$, $PLLV_{DD2}$ on	0	500	ms
t2	RESET# deasserted from $HIOV_{DD}$, IOV_{DD}, $PIO1V_{DD}$, $PIO2V_{DD}$, SDV_{DD}, $CM1V_{DD}$ on	55	—	ns

7.2.3 Power-Off Sequence

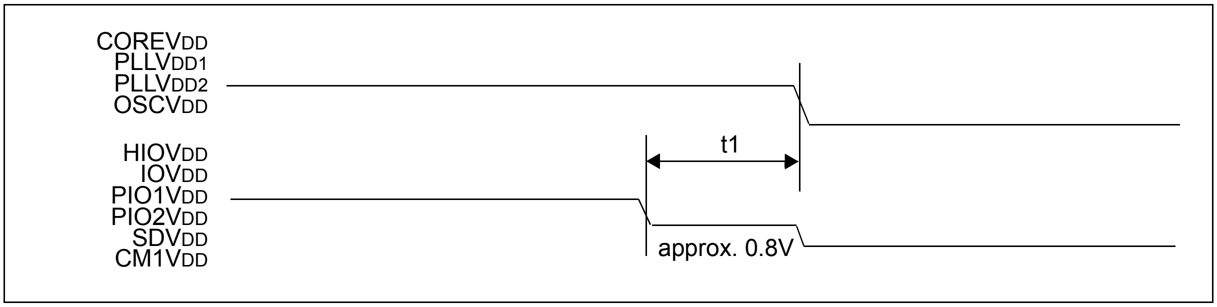


Figure 7-5: Power-Off Sequence

Table 7-6: Power-Off Sequence

Symbol	Parameter	Min	Max	Units
t1	$COREV_{DD}$, $OSCV_{DD}$, $PLLV_{DD1}$, $PLLV_{DD2}$ off delay from $HIOV_{DD}$, IOV_{DD}, $PIO1V_{DD}$, $PIO2V_{DD}$, SDV_{DD}, $CM1V_{DD}$ off	0	500	ms

7.3 RESET# Timing

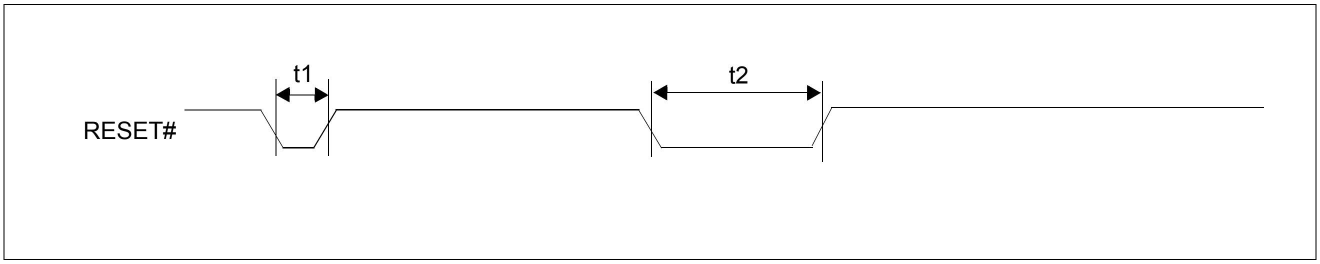


Figure 7-6 RESET# Timing

Table 7-7 RESET# Timing

Symbol	Parameter	Min	Max	Units
t1	Reset Pulse Width to be ignored	—	20	ns
t2	Active Reset Pulse Width	55	—	ns

1. If the reset pulse width is less than t1max, it is ignored. If the reset pulse width is between t1max and t2min, there is no guarantee that the reset will take effect. To ensure that reset takes effect, the reset pulse width must be greater than t2min.
2. When the OSC is used to supply clock source for system clock, CNF0 = 1b, then the RESET# should be asserted long enough for the crystal oscillator to stabilize its clock output before de-asserting. The crystal startup time varies based on crystal, and external crystal oscillator components used.

7.4 Parallel Host Bus Interface Timing

7.4.1 Direct/Indirect Intel 80 Type 1

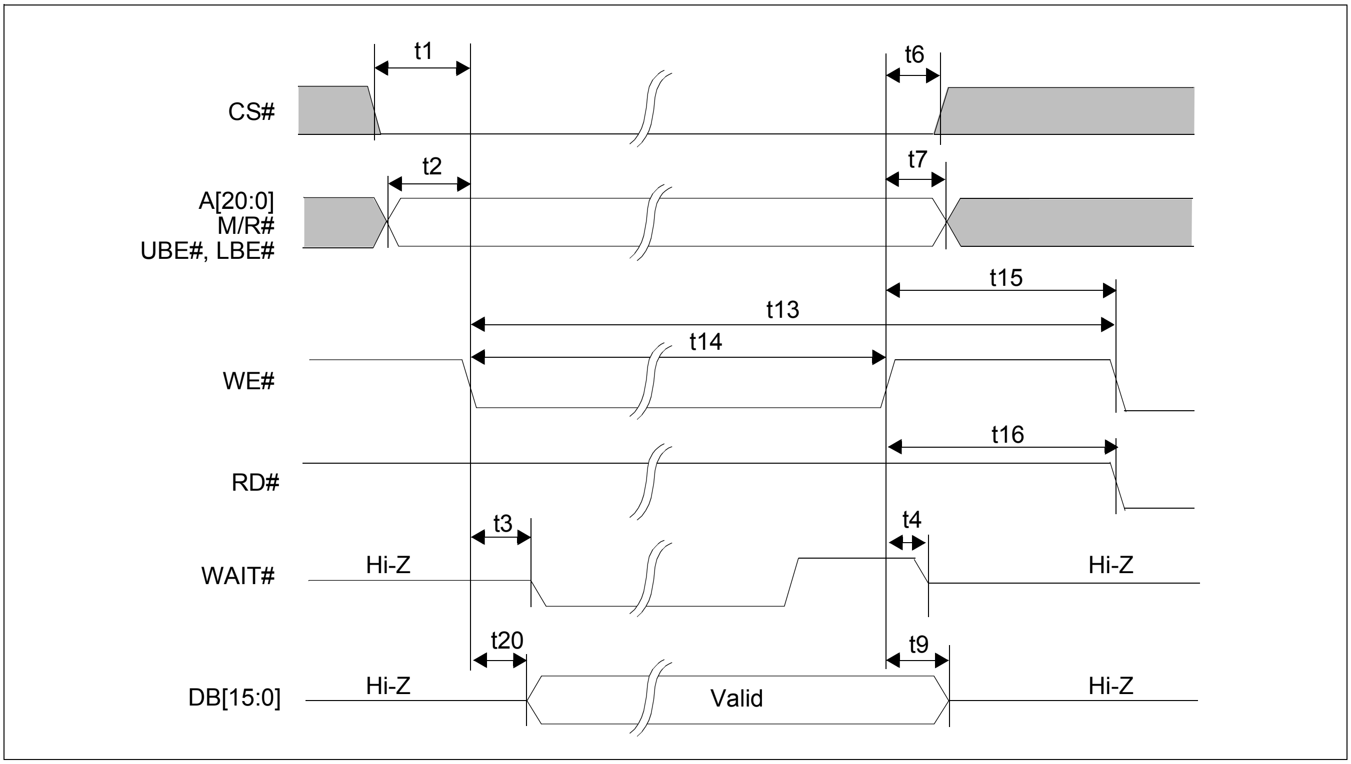


Figure 7-7: Direct/Indirect Intel 80 Type 1 Host Interface Write Timing

Note
For Indirect Intel 80 Type #1 8-bit, the BE1# and BE0# pins are not used.
For Indirect Intel 80 Type #1 16-bit, the BE1# and BE0# pins should be tied to logic 0. For byte access in this mode, refer to Section 21.2, "Intel80 Type1 Interface" on page 519, note 2.

Table 7-8: Direct/Indirect Intel 80 Type 1 Host Interface Write Timing

Symbol	Parameter	HIOVDD = 2.5V		HIOVDD = 3.3V		Units
		Min	Max	Min	Max	
t1	CS# setup time to WE# falling edge	7	—	7	—	ns
t2	AB[20:0], M/R#, UBE#, LBE# setup time to WE# falling edge	7	—	7	—	ns
t3	WE# falling edge to WAIT# driven for REG[003Dh] bit 0 = 0b	6	24	6	21	ns
	for REG[003Dh] bit 0 = 1b	5	23	5	20	ns
t4	WE# rising edge to WAIT# release for REG[003Dh] bit 0 = 0b	3	15	3	15	ns
	for REG[003Dh] bit 0 = 1b	3	15	3	15	ns
t6	WE# rising edge to CS# hold time	7	—	7	—	ns
t7	WE# rising edge to AB[20:0], M/R#, UBE#, LBE# hold time	7	—	7	—	ns
t9	DB[15:0] hold time from WE# rising edge	5	—	5	—	ns
t13	WE# cycle time - synchronous register access	3	—	3	—	Ts (Note 1)
	WE# cycle time - asynchronous register access	55	—	55	—	ns
t14	WE# pulse active time - synchronous register access	2	—	2	—	Ts
	WE# pulse active time - asynchronous register access	37	—	37	—	ns
t15	WE# pulse inactive time - synchronous register access	1	—	1	—	Ts
	WE# pulse inactive time - asynchronous register access	19	—	19	—	ns
t16	WE# rising edge to RD# falling edge - synchronous register access	1	—	1	—	Ts
	WE# rising edge to RD# falling edge - asynchronous register access	19	—	19	—	ns
t20	WE# falling edge to DB[15:0] valid write data - synchronous register access	—	Ts-10	—	Ts-10	ns
	WE# falling edge to DB[15:0] valid write data - asynchronous register access	—	8	—	8	ns

1. Ts = System clock period

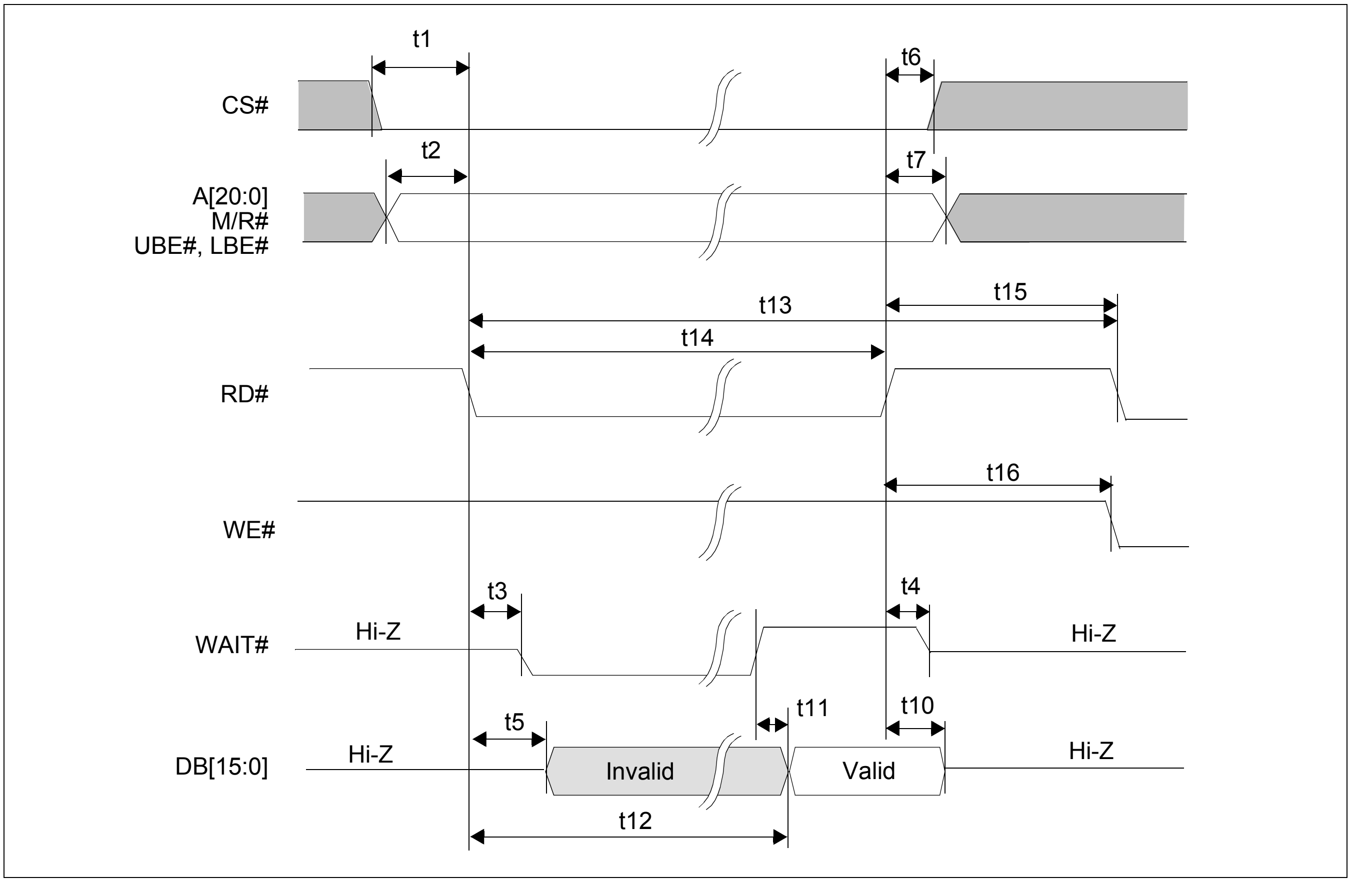


Figure 7-8: Direct/Indirect Intel 80 Type 1 Host Interface Read Timing

Note

For Indirect Intel 80 Type #1 8-bit, the BE1# and BE0# pins are not used.
For Indirect Intel 80 Type #1 16-bit, the BE1# and BE0# pins should be tied to logic 0. For byte access in this mode, refer to Section 21.2, "Intel80 Type1 Interface" on page 519, note 2.

Table 7-9: Direct/Indirect Intel 80 Type 1 Host Interface Read Timing

Symbol	Parameter	HIOVDD = 2.5V		HIOVDD = 3.3V		Units
		Min	Max	Min	Max	
t1	CS# setup time to RD# falling edge	7	—	7	—	ns
t2	AB[20:0], M/R#, UBE#, LBE# setup time to RD# falling edge	7	—	7	—	ns
t3	RD# falling edge to WAIT# driven for REG[003Dh] bit 0 = 0b	8	28	8	25	ns
	for REG[003Dh] bit 0 = 1b	7	27	7	25	ns
t4	RD# rising edge to WAIT# release for REG[003Dh] bit 0 = 0b	4	19	4	19	ns
	for REG[003Dh] bit 0 = 1b	4	19	4	18	ns
t5	RD# falling edge to DB[15:0] driven for REG[003Dh] bit 0 = 0b	7	28	7	25	ns
	for REG[003Dh] bit 0 = 1b	7	27	6	25	ns
t6	RD# rising edge to CS# hold time	10	—	10	—	ns
t7	RD# rising edge to AB[20:0], M/R#, UBE#, LBE# hold time	10	—	10	—	ns
t10	DB[15:0] hold time from RD# rising edge for REG[003Dh] bit 0 = 0b	4	20	4	20	ns
	for REG[003Dh] bit 0 = 1b	4	20	4	20	ns
t11	WAIT# rising edge to valid DATA if WAIT# asserted for REG[003Dh] bit 0 = 0b	—	6	—	4	ns
	for REG[003Dh] bit 0 = 1b	—	6	—	4	ns
t12	RD# falling edge to valid DATA if WAIT# is NOT asserted for REG[003Dh] bit 0 = 0b	—	28	—	26	ns
	for REG[003Dh] bit 0 = 1b	—	28	—	25	ns
t13	RD# cycle time - synchronous register access	3	—	3	—	Ts (Note 1)
	RD# cycle time - asynchronous register access	55	—	55	—	ns
t14	RD# pulse active time - synchronous register access	2	—	2	—	Ts
	RD# pulse active time - asynchronous register access	37	—	37	—	ns
t15	RD# pulse inactive time - synchronous register access	1	—	1	—	Ts
	RD# pulse inactive time - asynchronous register access	19	—	19	—	ns
t16	RD# rising edge to WE# falling edge - synchronous register access	1	—	1	—	Ts
	RD# rising edge to WE# falling edge - asynchronous register access	19	—	19	—	ns

1. Ts = System clock period

7.4.2 Direct/Indirect Intel 80 Type 2

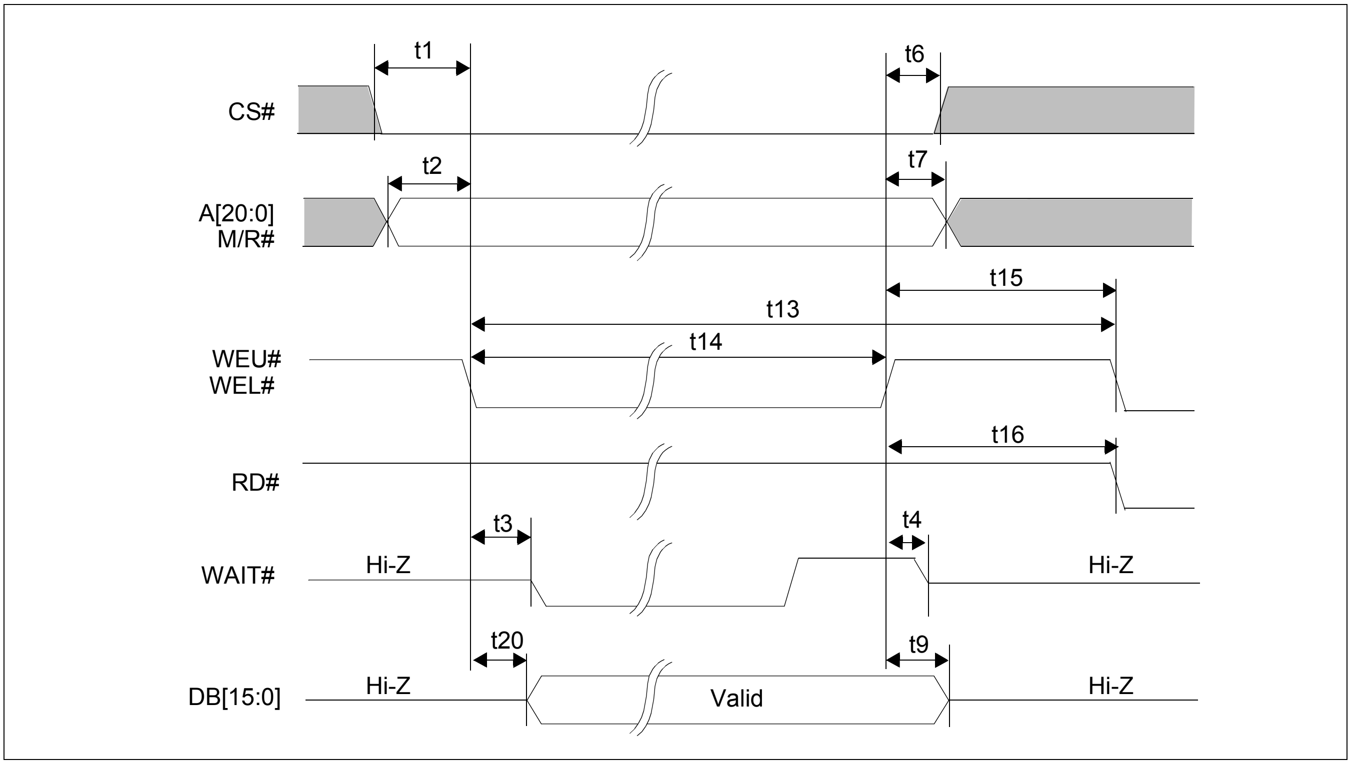


Figure 7-9: Direct/Indirect Intel 80 Type 2 Host Interface Write Timing

Note
For Indirect Intel 80 Type #2 8-bit, the WEU# is not used.
For Indirect Intel 80 Type #2 16-bit, the WEU# and WEL# pins should be driven in unison (16-bit host write access is mandatory). For byte access in this mode, refer to Section 21.3, "Intel80 Type2 Interface" on page 520, note 2.

Table 7-10: Direct/Indirect Intel 80 Type 2 Host Interface Write Timing

Symbol	Parameter	HIOVDD = 2.5V		HIOVDD = 3.3V		Units
		Min	Max	Min	Max	
t1	CS# setup time to WEU#, WEL# falling edge	7	—	7	—	ns
t2	AB[20:0], M/R# setup time to WEU#, WEL# falling edge	7	—	7	—	ns
t3	WEU#, WEL# falling edge to WAIT# driven for REG[003Dh] bit 0 = 0b	6	25	6	22	ns
	for REG[003Dh] bit 0 = 1b	5	24	5	21	ns
t4	WEU#, WEL# rising edge to WAIT# release for REG[003Dh] bit 0 = 0b	2	16	2	16	ns
	for REG[003Dh] bit 0 = 1b	2	16	2	16	ns
t6	WEU#, WEL# rising edge to CS# hold time	7	—	7	—	ns
t7	WEU#, WEL# rising edge to AB[20:0], M/R# hold time	7	—	7	—	ns
t9	DB[15:0] hold time from WEU#, WEL# rising edge	5	—	5	—	ns
t13	WEU#, WEL# cycle time - synchronous register access	3	—	3	—	Ts (Note 1)
	WEU#, WEL# cycle time - asynchronous register access	55	—	55	—	ns
t14	WEU#, WEL# pulse active time - synchronous register access	2	—	2	—	Ts
	WEU#, WEL# pulse active time - asynchronous register access	37	—	37	—	ns
t15	WEU#, WEL# pulse inactive time - synchronous register access	1	—	1	—	Ts
	WEU#, WEL# pulse inactive time - asynchronous register access	19	—	19	—	ns
t16	WEU#, WEL# rising edge to RD# falling edge - synchronous register access	1	—	1	—	Ts
	WEU#, WEL# rising edge to RD# falling edge - asynchronous register access	19	—	19	—	ns
t20	WEU#, WEL# falling edge to DB[15:0] valid write data - synchronous register access	—	Ts-10	—	Ts-10	ns
	WEU#, WEL# falling edge to DB[15:0] valid write data - asynchronous register access	—	8	—	8	ns

1. Ts = System clock period

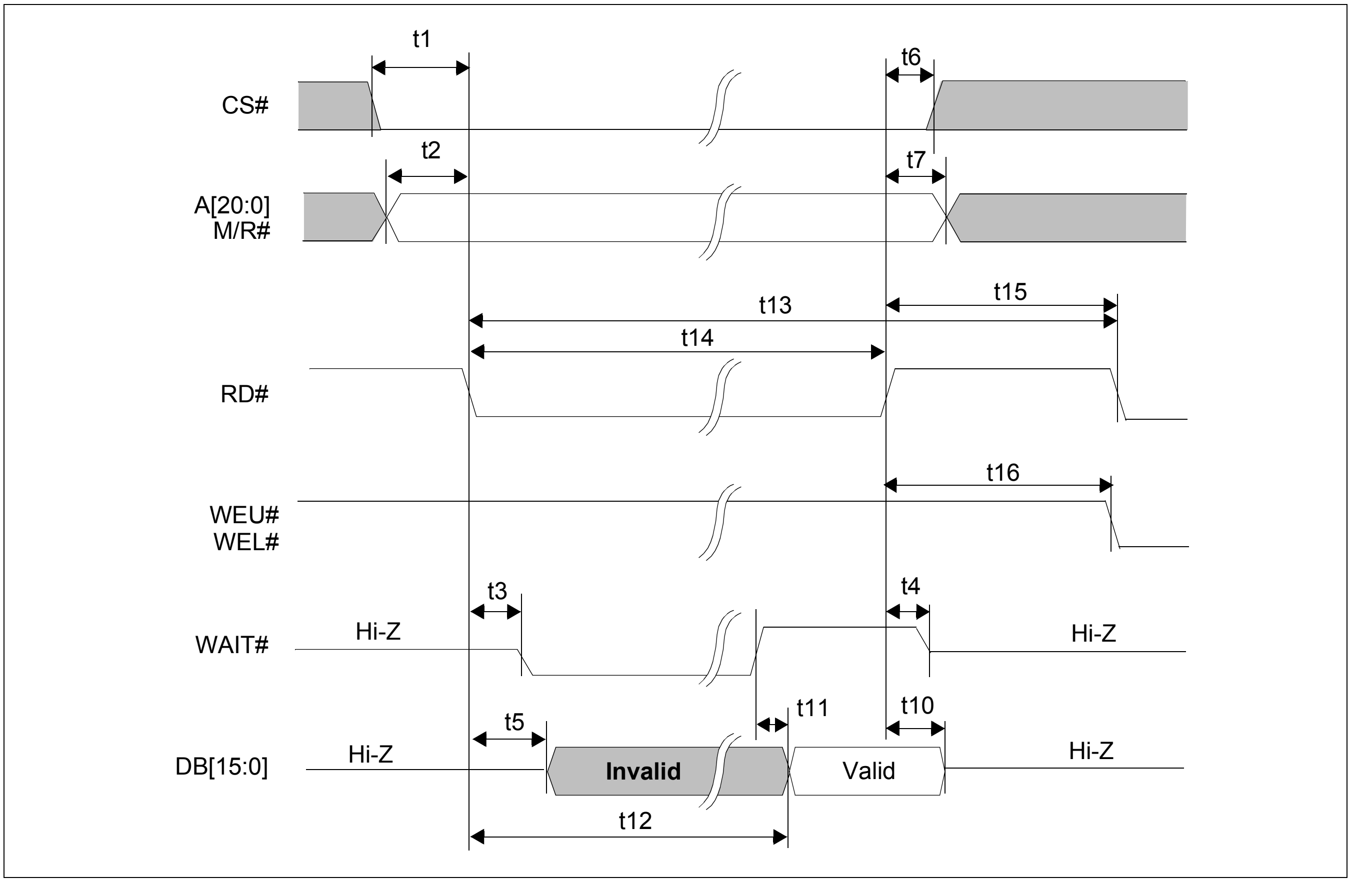


Figure 7-10: Direct/Indirect Intel 80 Type 2 Host Interface Read Timing

Note

For Indirect Intel 80 Type #2 8-bit, the WEU# is not used.
For Indirect Intel 80 Type #2 16-bit, the WEU# and WEL# pins should be driven in unison (16-bit host write access is mandatory). For byte access in this mode, refer to Section 21.3, "Intel80 Type2 Interface" on page 520, note 2.

Table 7-11: Direct/Indirect Intel 80 Type 2 Host Interface Read Timing

Symbol	Parameter	HIOVDD = 2.5V		HIOVDD = 3.3V		Units
		Min	Max	Min	Max	
t1	CS# setup time to RD# falling edge	7	—	7	—	ns
t2	AB[20:0], M/R# setup time to RD# falling edge	7	—	7	—	ns
t3	RD# falling edge to WAIT# driven for REG[003Dh] bit 0 = 0b	8	28	8	25	ns
	for REG[003Dh] bit 0 = 1b	7	27	7	25	ns
t4	RD# rising edge to WAIT# release for REG[003Dh] bit 0 = 0b	4	19	4	19	ns
	for REG[003Dh] bit 0 = 1b	4	19	4	19	ns
t5	RD# falling edge to DB[15:0] driven for REG[003Dh] bit 0 = 0b	7	28	7	25	ns
	for REG[003Dh] bit 0 = 1b	7	27	6	25	ns
t6	RD# rising edge to CS# hold time	9	—	9	—	ns
t7	RD# rising edge to AB[20:0], M/R# hold time	9	—	9	—	ns
t10	DB[15:0] hold time from RD# rising edge for REG[003Dh] bit 0 = 0b	4	20	4	20	ns
	for REG[003Dh] bit 0 = 1b	4	20	4	20	ns
t11	WAIT# rising edge to valid DATA if WAIT# asserted for REG[003Dh] bit 0 = 0b	—	5	—	4	ns
	for REG[003Dh] bit 0 = 1b	—	5	—	4	ns
t12	RD# falling edge to valid DATA if WAIT# is NOT asserted for REG[003Dh] bit 0 = 0b	—	29	—	27	ns
	for REG[003Dh] bit 0 = 1b	—	28	—	25	ns
t13	RD# cycle time - synchronous register access	3	—	3	—	Ts (Note 1)
	RD# cycle time - asynchronous register access	55	—	55	—	ns
t14	RD# pulse active time - synchronous register access	2	—	2	—	Ts
	RD# pulse active time - asynchronous register access	37	—	37	—	ns
t15	RD# pulse inactive time - synchronous register access	1	—	1	—	Ts
	RD# pulse inactive time - asynchronous register access	19	—	19	—	ns
t16	RD# rising edge to WEU#, WEL# falling edge - synchronous register access	1	—	1	—	Ts
	RD# rising edge to WEU#, WEL# falling edge - asynchronous register access	19	—	19	—	ns

1. Ts = System clock period

7.4.3 Direct Marvell PXA3xx VLIO

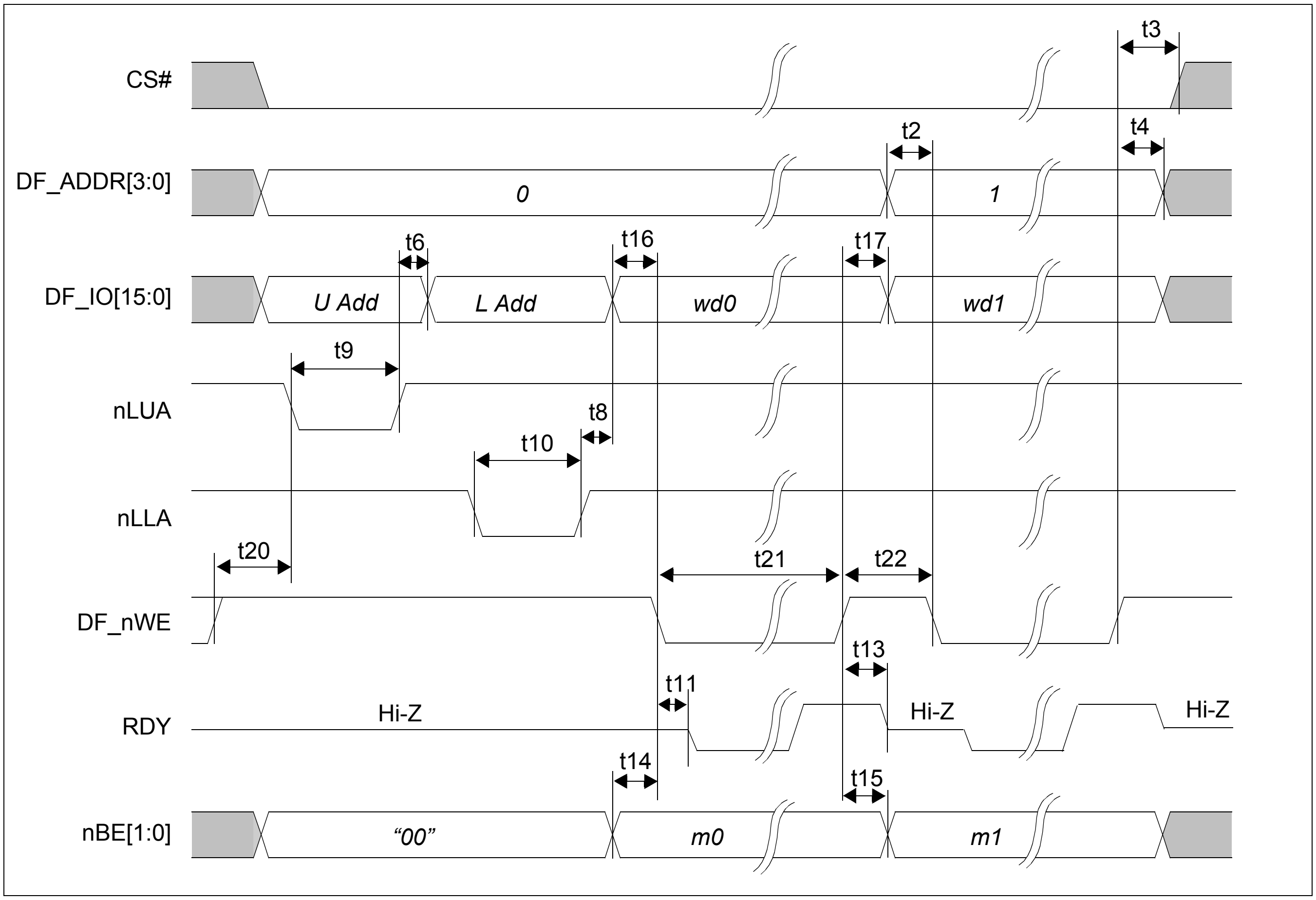


Figure 7-11: Direct Marvell PXA3xx VLIO Host Interface Write Timing

Table 7-12: Direct Marvell PXA3xx VLIO Host Interface Write Timing

Symbol	Parameter	HIOVDD = 2.5V		HIOVDD = 3.3V		Units
		Min	Max	Min	Max	
t2	DF_ADDR[3:0] setup time to DF_nWE falling edge	6	—	6	—	ns
t3	CS# hold time from DF_nWE rising edge	7	—	7	—	ns
t4	DF_ADDR[3:0] hold time from DF_nWE rising edge	7	—	7	—	ns
t6	DF_IO[15:0] hold time from nLUA rising edge	0	—	0	—	ns
t8	DF_IO[15:0] hold time from nLLA rising edge	0	—	0	—	ns
t9	nLUA pulse active time	25	—	25	—	ns
t10	nLLA pulse active time	25	—	25	—	ns
t11	DF_nWE falling edge to RDY driven for REG[003Dh] bit 0 = 0b	6	24	6	21	ns
	for REG[003Dh] bit 0 = 1b	5	23	5	20	ns
t13	DF_nWE rising edge to RDY tristate for REG[003Dh] bit 0 = 0b	3	15	3	15	ns
	for REG[003Dh] bit 0 = 1b	3	15	3	15	ns
t14	nBE[1:0] setup time to DF_nWE falling edge	6	—	6	—	ns
t15	nBE[1:0] hold time from DF_nWE rising edge	7	—	7	—	ns
t16	DF_IO[15:0] setup time to DF_nWE falling edge	0	—	0	—	ns
t17	DF_IO[15:0] hold time from DF_nWE rising edge	4	—	4	—	ns
t20	DF_nWE rising edge to nLUA falling edge - synchronous register access	1	—	1	—	Ts (Note 1)
	DF_nWE rising edge to nLUA falling edge - asynchronous register access	19	—	19	—	ns
t21	DF_nWE pulse active time - synchronous register access	2	—	2	—	Ts
	DF_nWE pulse active time - asynchronous register access	37	—	37	—	ns
t22	DF_nWE pulse inactive time - synchronous register access	1	—	1	—	Ts
	DF_nWE pulse inactive time - asynchronous register access	19	—	19	—	ns

1. Ts = System clock period

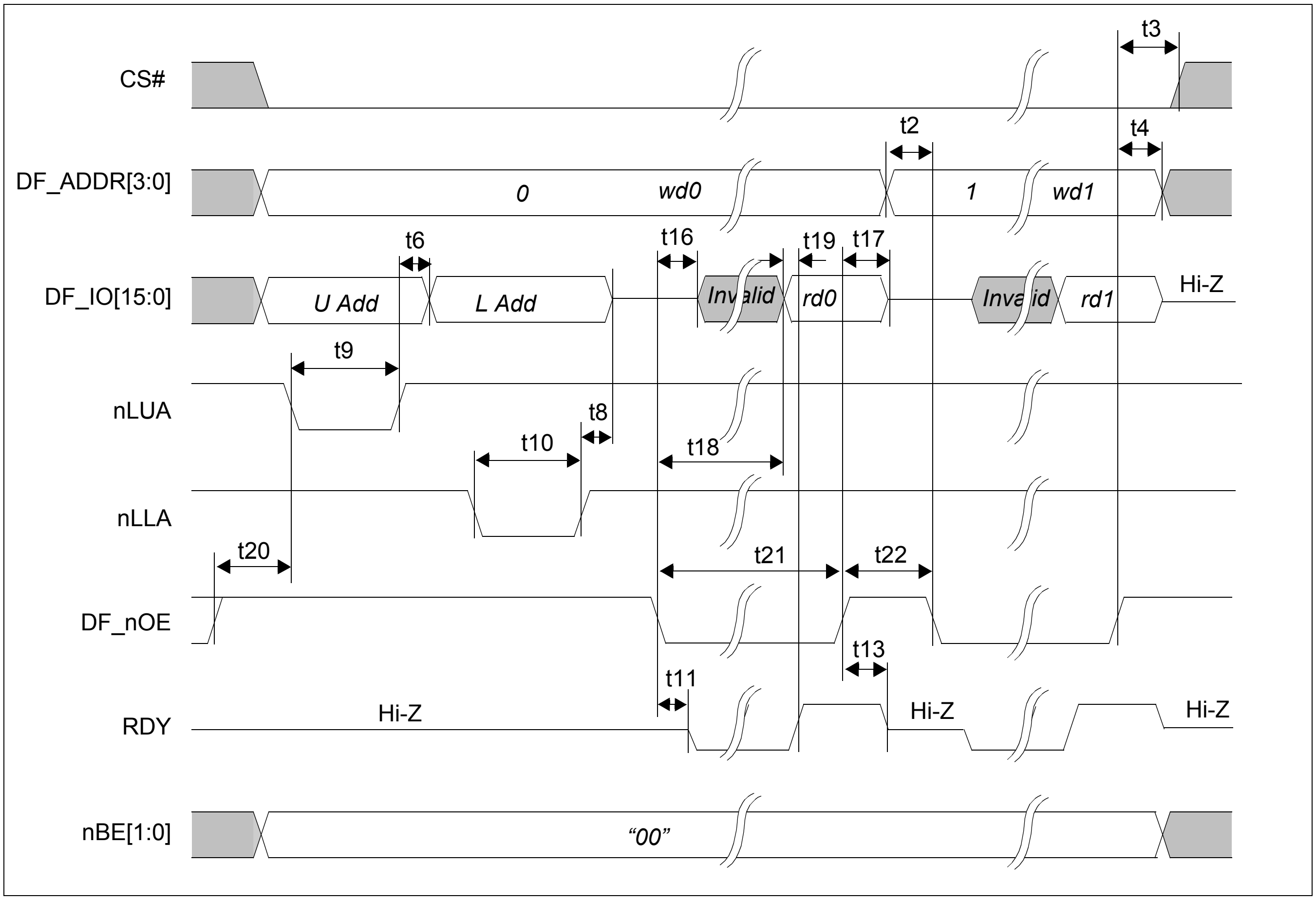


Figure 7-12: Direct Marvell PXA3xx VLIO Host Interface Read Timing

Table 7-13: Direct Marvell PXA3xx VLIO Host Interface Read Timing

Symbol	Parameter	HIOVDD = 2.5V		HIOVDD = 3.3V		Units
		Min	Max	Min	Max	
t2	DF_ADDR[3:0] setup time to DF_nOE falling edge	6	—	6	—	ns
t3	CS# hold time from DF_nOE rising edge	8	—	8	—	ns
t4	DF_ADDR[3:0] hold time from DF_nOE rising edge	8	—	8	—	Ns
t6	DF_IO[15:0] hold time from nLUA rising edge	0	—	0	—	Ns
t8	DF_IO[15:0] hold time from nLLA rising edge	0	—	0	—	Ns
t9	nLUA pulse active time	25	—	25	—	Ns
t10	nLLA pulse active time	25	—	25	—	Ns
t11	DF_nOE falling edge to RDY driven for REG[003Dh] bit 0 = 0b	8	28	8	25	ns
	for REG[003Dh] bit 0 = 1b	7	27	7	25	ns
t13	DF_nOE rising edge to RDY tristate for REG[003Dh] bit 0 = 0b	4	19	4	19	ns
	for REG[003Dh] bit 0 = 1b	4	19	4	19	ns
t16	DF_nOE falling edge to DF_IO[15:0] driven for REG[003Dh] bit 0 = 0b	7	28	7	25	ns
	for REG[003Dh] bit 0 = 1b	7	27	6	25	ns
t17	DF_IO[15:0] hold time from DF_nOE rising edge for REG[003Dh] bit 0 = 0b	4	20	4	20	ns
	for REG[003Dh] bit 0 = 1b	4	20	4	20	ns
t18	DF_nOE falling edge to valid data if RDY does not go to low - synchronous register access for REG[003Dh] bit 0 = 0b	—	28	—	25	ns
	for REG[003Dh] bit 0 = 1b	—	27	—	25	ns
t19	Valid data before RDY rising edge if RDY goes to low - asynchronous register access f or REG[003Dh] bit 0 = 0b	Note 2	—	Note 4	—	ns
	for REG[003Dh] bit 0 = 1b	Note 3	—	Note 5	—	ns
t20	DF_nOE rising edge to nLUA falling edge - synchronous register access	1	—	1	—	Ts (Note 1)
	DF_nOE rising edge to nLUA falling edge - asynchronous register access	19	—	19	—	ns
t21	DF_nOE pulse active time - synchronous register access	2	—	2	—	Ts
	DF_nOE pulse active time - asynchronous register access	37	—	37	—	ns
t22	DF_nOE pulse inactive time - synchronous register access	1	—	1	—	Ts
	DF_nOE pulse inactive time - asynchronous register access	19	—	19	—	ns

1. Ts = System clock period
2. When HIOVDD = 2.5V and REG[003Dh] bit 0 = 0b, t19min is calculated using the following formula.
   t19min = (REG[0085h] bits 2-0) x (System clock period) - 8.0ns
3. When HIOVDD = 2.5V and REG[003Dh] bit 0 = 1b, t19min is calculated using the following formula.
   t19min = (REG[0085h] bits 2-0) x (System clock period) - 8.0ns
4. When HIOVDD = 3.3V and REG[003Dh] bit 0 = 0b, t19min is calculated using the following formula.
   t19min = (REG[0085h] bits 2-0) x (System clock period) - 7.0ns
5. When HIOVDD = 3.3V and REG[003Dh] bit 0 = 1b, t19min is calculated using the following formula.
   t19min = (REG[0085h] bits 2-0) x (System clock period) - 7.0ns

7.4.4 Direct/Indirect Renesas SH4

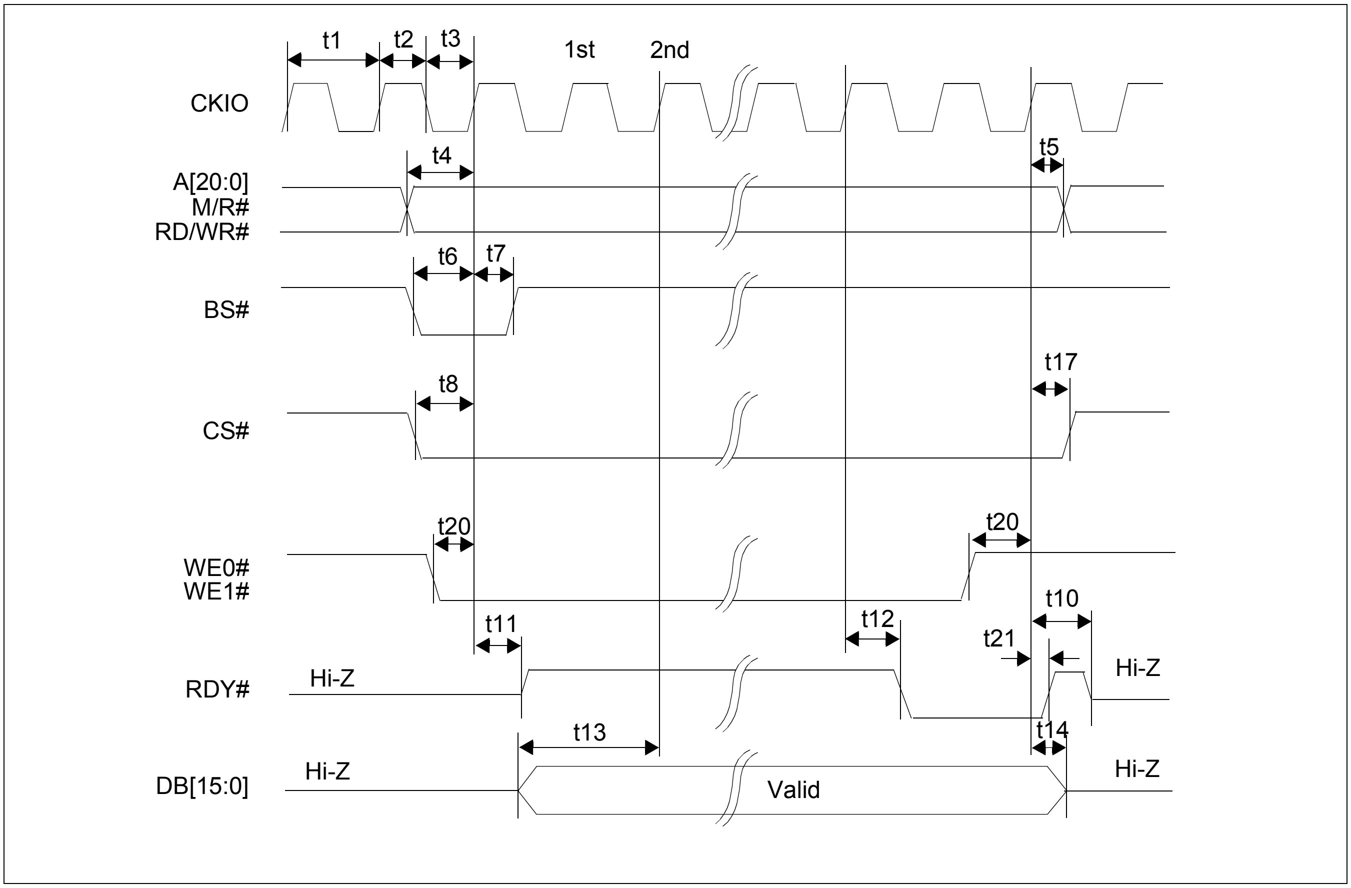


Figure 7-13: Direct/Indirect Renesas SH4 Host Interface Write Timing

Note
For Indirect SH4 8-bit, the WE1# and WE0# is not used.
For Indirect SH4 16-bit, the WE1# and WE0# pins should be driven in unison (16-bit host write access is mandatory). For byte access in this mode, refer to Section 21.6, “Renesas SH4 Interface” on page 523, note 2.

Table 7-14: Direct/Indirect Renesas SH4 Host Interface Write Timing

Symbol	Parameter	HIOVDD = 2.5V		HIOVDD = 3.3V		Units
		Min	Max	Min	Max	
fCKIO	Clock frequency	—	25	—	25	MHz
t1	Clock period	40	—	40	—	ns
t2	Clock pulse width high	20	—	20	—	ns
t3	Clock pulse width low	20	—	20	—	ns
t4	AB[20:0], M/R#, RD/WR# setup to CKIO	7	—	7	—	ns
t5	AB[20:0], M/R#, RD/WR# hold from CKIO	0	—	0	—	ns
t6	BS# setup	7	—	7	—	ns
t7	BS# hold	0	—	0	—	ns
t8	CS# setup	5	—	5	—	ns
t10	CKIO to RDY# tristate for REG[003Dh] bit 0 = 0b	4	24	4	21	ns
	for REG[003Dh] bit 0 = 1b	4	23	4	21	ns
t11	CKIO to RDY# driven for REG[003Dh] bit 0 = 0b	4	24	4	21	ns
	for REG[003Dh] bit 0 = 1b	4	23	4	21	ns
t12	CKIO to RDY# low for REG[003Dh] bit 0 = 0b	—	20	—	18	ns
	for REG[003Dh] bit 0 = 1b	—	19	—	17	ns
t13	DB[15:0] setup to 2nd CKIO after BS#	0	—	0	—	ns
t14	DB[15:0] hold from CKIO	0	—	0	—	ns
t17	CS# hold from CKIO	0	—	0	—	ns
t20	WE0#, WE1# setup to CKIO	8	—	8	—	ns
t21 (Note 2)	CKIO to RDY# high for REG[003Dh] bit 0 = 0b	0	—	0	—	ns
	for REG[003Dh] bit 0 = 1b	0	—	0	—	ns

Note

1. When the S1D13515/S2D13515 completes a write, RDY# is driven low and then asserted high 2 CKIO later. This means that RDY# is only low for a 2 CKIO period. To sample RDY# low correctly, the SH4 Wait Control Register 2 (WCR2) must be set appropriately. For details on SH4 registers, refer to the SH4 specification.
2. At the end of the write cycle, RDY# may not drive HIGH and may become tri-stated (high impedance) 1 bus clock after RDY# was asserted LOW.

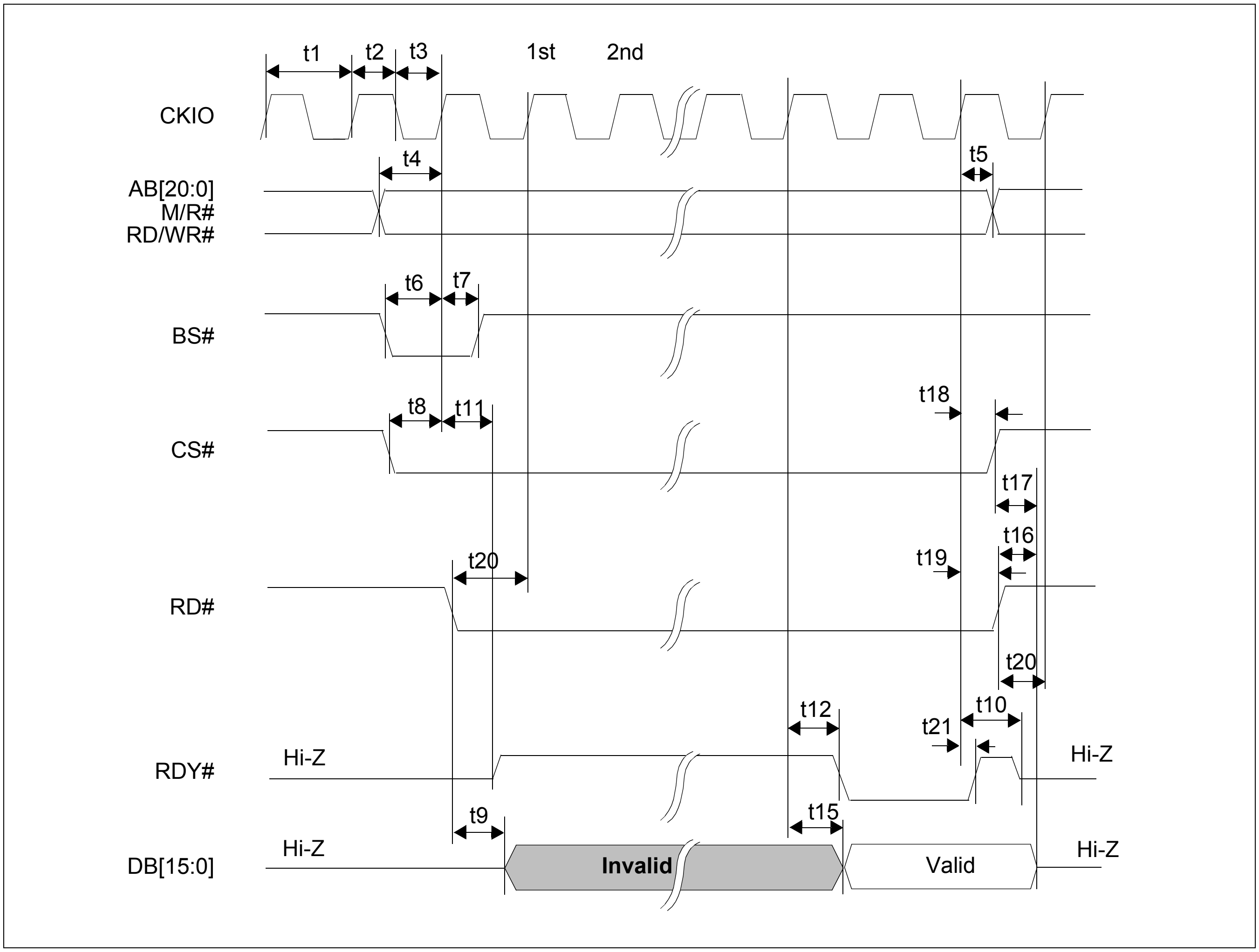


Figure 7-14: Direct/Indirect Renesas SH4 Host Interface Read Timing

Note

For Indirect SH4 8-bit, the WE1# and WE0# is not used.

For Indirect SH4 16-bit, the WE1# and WE0# pins should be driven in unison (16-bit host write access is mandatory). For byte access in this mode, refer to Section 21.6, "Renesas SH4 Interface" on page 523, note 2.

Table 7-15: Direct/Indirect Renesas SH4 Host Interface Read Timing

Symbol	Parameter	HIOVDD = 2.5V		HIOVDD = 3.3V		Units
		Min	Max	Min	Max	
fCKIO	Clock frequency	—	25	-	25	MHz
t1	Clock period	40	—	40	—	ns
t2	Clock pulse width high	20	—	20	—	ns
t3	Clock pulse width low	20	—	20	—	ns
t4	AB[20:0], M/R#, RD/WR# setup to CKIO	7	—	7	—	ns
t5	AB[20:0], M/R#, RD/WR# hold from CKIO	0	—	0	—	ns
t6	BS# setup	7	—	7	—	ns
t7	BS# hold	0	—	0	—	ns
t8	CS# setup	5	—	5	—	ns
t9	Falling edge of RD# to DB[15:0] driven for REG[003Dh] bit 0 = 0b	7	—	7	—	ns
	for REG[003Dh] bit 0 = 1b	7	—	7	—	ns
t10	CKIO to RDY# tristate for REG[003Dh] bit 0 = 0b	4	24	4	21	ns
	for REG[003Dh] bit 0 = 1b	4	23	4	21	ns
t11	CKIO to RDY# driven for REG[003Dh] bit 0 = 0b	4	24	4	21	ns
	for REG[003Dh] bit 0 = 1b	4	23	4	21	ns
t12	CKIO to RDY# low for REG[003Dh] bit 0 = 0b	—	20	—	18	ns
	for REG[003Dh] bit 0 = 1b	—	19	—	17	ns
t15	CKIO to DB[15:0] valid for REG[003Dh] bit 0 = 0b	—	20	—	17	ns
	for REG[003Dh] bit 0 = 1b	—	19	—	17	ns
t16	RD# rising edge to DB[15:0] tristate for REG[003Dh] bit 0 = 0b	4	22	4	22	ns
	for REG[003Dh] bit 0 = 1b	4	22	4	22	ns
t17	CS# rising edge to DB[15:0] tristate for REG[003Dh] bit 0 = 0b	3	13	3	13	ns
	for REG[003Dh] bit 0 = 1b	3	13	3	13	ns
t18	CS# hold from CKIO	0	—	0	—	ns
t19	RD# hold from CKIO	0	—	0	—	ns
t20	RD# setup to CKIO	10	—	10	—	ns
t21 (Note 2)	CKIO to RDY# high for REG[003Dh] bit 0 = 0b	0	—	0	—	ns
	for REG[003Dh] bit 0 = 1b	0	—	0	—	ns

Note

1. When read data is ready, RDY# is driven low and then asserted high 2 CKIO later. This means that RDY# is only low for a 2 CKIO period. To sample RDY# low correctly, the SH4 Wait Control Register 2 (WCR2) must be set appropriately. For details on SH4 registers, refer to the SH4 specification.
2. At the end of the read cycle, RDY# may not drive HIGH and may become tri-stated (high impedance) 1 bus clock after RDY# was asserted LOW.

7.4.5 Direct/Indirect Freescale MPC555 (Non-burst Mode)

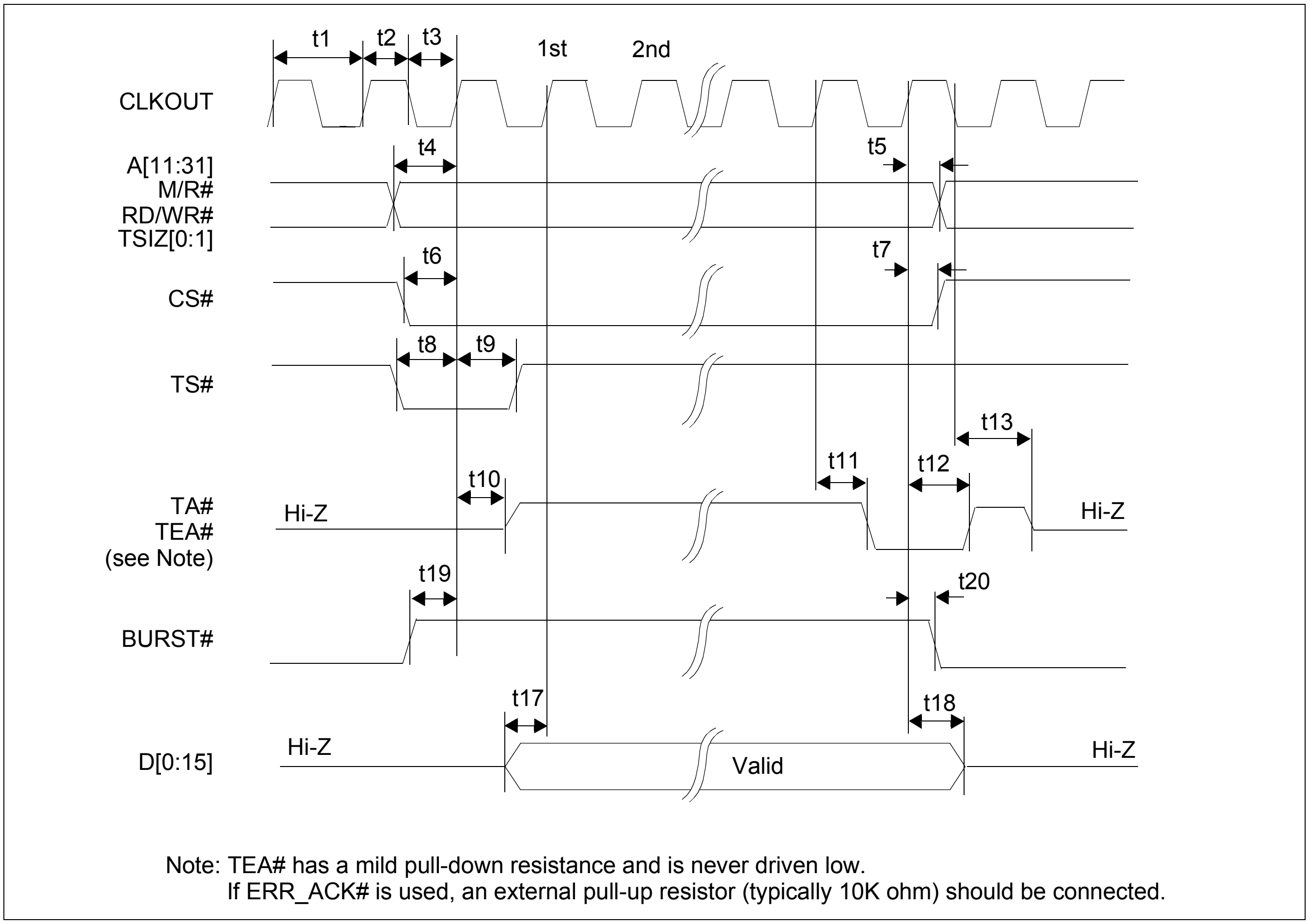


Figure 7-15: Direct/Indirect Freescale MPC555 Host Interface Write Timing (Non-burst Mode)

Note

1. For Indirect MPC555, the TSIZ0 pin should be tied to “1” and TSIZ1 should be tied to “0” (16-bit host access is mandatory). For byte access in this mode, refer to Section 21.9, “MPC555 Interface” on page 526 note 3.
2. The S1D13515/S2D13515 does not support Big Endian configuration. The host processor must be configured for Little Endian External Bus when connected to the S1D13515/S2D13515 using the MPC555 interface.

Table 7-16: Direct/Indirect Freescale MPC555 Host Interface Write Timing (Non-burst Mode)

Symbol	Parameter	HIOVDD = 2.5V		HIOVDD = 3.3V		Units
		Min	Max	Min	Max	
fCLKOUT	Clock frequency	—	25	—	25	MHz
t1	Clock period	40	—	40	—	ns
t2	Clock pulse width high	20	—	20	—	ns
t3	Clock pulse width low	20	—	20	—	ns
t4	A[11:31], M/R#, RD/WR#, TSIZ[0:1] setup	7	—	7	—	ns
t5	A[11:31], M/R#, RD/WR#, TSIZ[0:1] hold	0	—	0	—	ns
t6	CS# setup	5	—	5	—	ns
t7	CS# hold	0	—	0	—	ns
t8	TS# setup	8	—	8	—	ns
t9	TS# hold	0	—	0	—	ns
t10	CLKOUT to TA#, TEA# driven for REG[003Dh] bit0 = 0b	4	24	4	21	ns
	for REG[003Dh] bit0 = 1b	4	23	4	21	ns
t11	CLKOUT to TA#, TEA# low for REG[003Dh] bit0 = 0b	—	20	—	17	ns
	for REG[003Dh] bit0 = 1b	—	19	—	17	ns
t12	CLKOUT to TA#, TEA# high for REG[003Dh] bit0 = 0b	6	20	5	17	ns
	for REG[003Dh] bit0 = 1b	5	19	5	17	ns
t13	Negative edge CLKOUT to TA#, TEA# tristate for REG[003Dh] bit0 = 0b	4	24	4	21	ns
	for REG[003Dh] bit0 = 1b	4	23	4	20	ns
t17	D[0:15] setup to 1st CLKOUT after TS#=0	0	—	0	—	ns
t18	CLKOUT to D[0:15] hold	0	—	0	—	ns
t19	BURST# setup	7	—	7	—	ns
t20	BURST# hold	0	—	0	—	ns

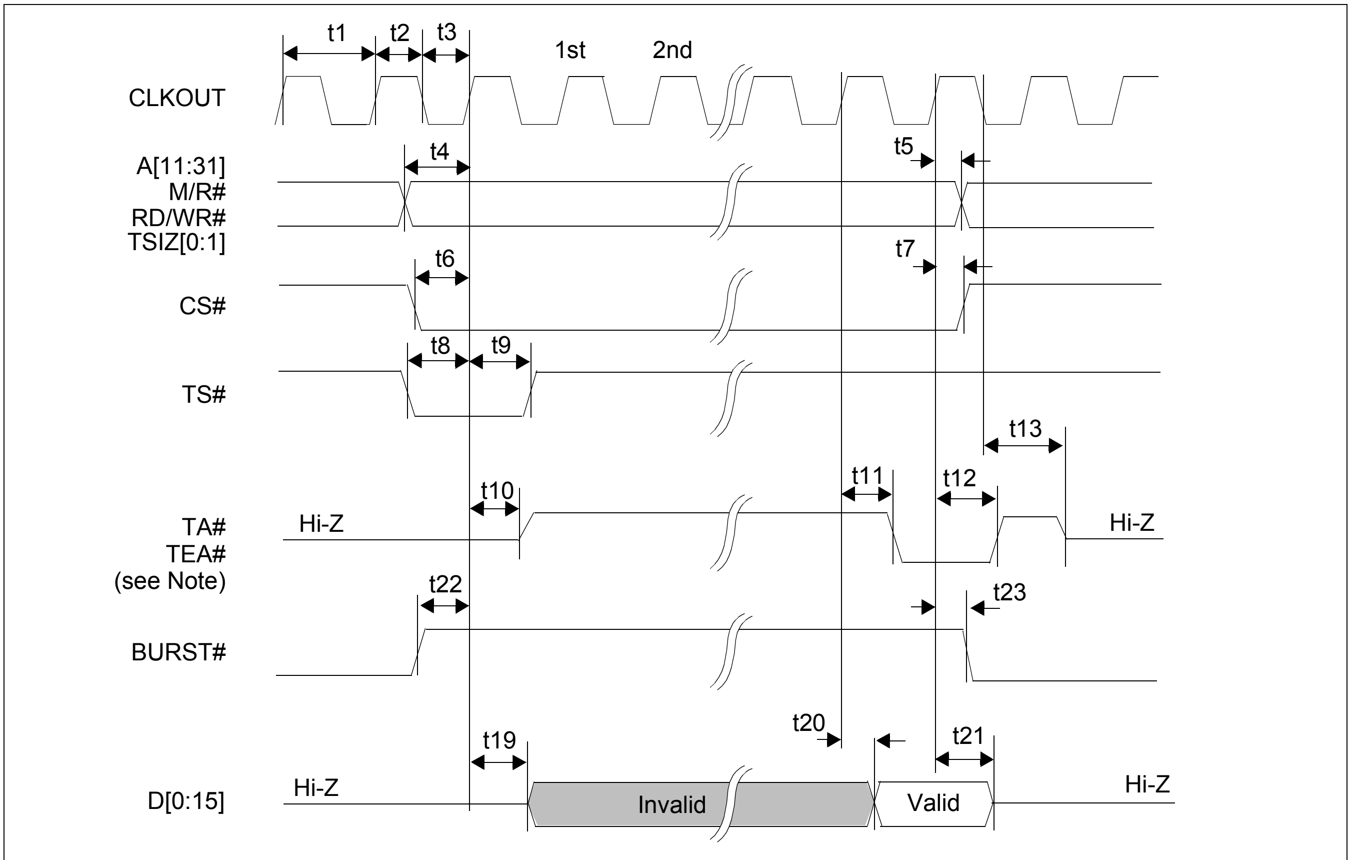


Figure 7-16: Direct/Indirect Freescale MPC555 Host Interface Read Timing (Non-burst Mode)

Note

1. For Indirect MPC555, the TSIZ0 pin should be tied to “1” and TSIZ1 should be tied to “0” (16-bit host access is mandatory). For byte access in this mode, refer to Section 21.9, “MPC555 Interface” on page 526 note 3.
2. The S1D13515/S2D13515 does not support Big Endian configuration. The host processor must be configured for Little Endian External Bus when connected to the S1D13515/S2D13515 using the MPC555 interface.

Table 7-17: Direct/Indirect Freescale MPC555 Host Interface Read Timing (Non-burst Mode)

Symbol	Parameter	HIOVDD = 2.5V		HIOVDD = 3.3V		Units
		Min	Max	Min	Max	
fCLKOUT	Clock frequency	—	25	—	25	MHz
t1	Clock period	40	—	40	—	ns
t2	Clock pulse width high	20	—	20	—	ns
t3	Clock pulse width low	20	—	20	—	ns
t4	A[11:31], M/R#, RD/WR#, TSIZ[0:1] setup	7	—	7	—	ns
t5	A[11:31], M/R#, RD/WR#, TSIZ[0:1] hold	0	—	0	—	ns
t6	CS# setup	5	—	5	—	ns
t7	CS# hold	0	—	0	—	ns
t8	TS# setup	8	—	8	—	ns
t9	TS# hold	0	—	0	—	ns
t10	CLKOUT to TA#, TEA# driven for REG[003Dh] bit 0 = 0b	4	24	4	21	ns
	for REG[003Dh] bit 0 = 1b	4	23	4	21	ns
t11	CLKOUT to TA#, TEA# low for REG[003Dh] bit 0 = 0b	—	20	—	17	ns
	for REG[003Dh] bit 0 = 1b	—	19	—	17	ns
t12	CLKOUT to TA#, TEA# high for REG[003Dh] bit 0 = 0b	6	20	5	17	ns
	for REG[003Dh] bit 0 = 1b	5	19	5	17	ns
t13	Negative edge CLKOUT to TA#, TEA# tristate for REG[003Dh] bit 0 = 0b	4	24	4	21	ns
	for REG[003Dh] bit 0 = 1b	4	23	4	20	ns
t19	CLKOUT to D[0:15] driven for REG[003Dh] bit 0 = 0b	5	—	5	—	ns
	for REG[003Dh] bit 0 = 1b	5	—	5	—	ns
t20	CLKOUT to D[0:15] valid for REG[003Dh] bit 0 = 0b	—	20	—	17	ns
	for REG[003Dh] bit 0 = 1b	—	19	—	16	ns
t21	CLKOUT to D[0:15] tristate for REG[003Dh] bit 0 = 0b	5	25	5	23	ns
	for REG[003Dh] bit 0 = 1b	5	24	5	22	ns
t22	BURST# setup	7	—	7	—	ns
t23	BURST# hold	0	—	0	—	ns

7.4.6 Direct/Indirect Freescale MPC555 (Burst Mode)

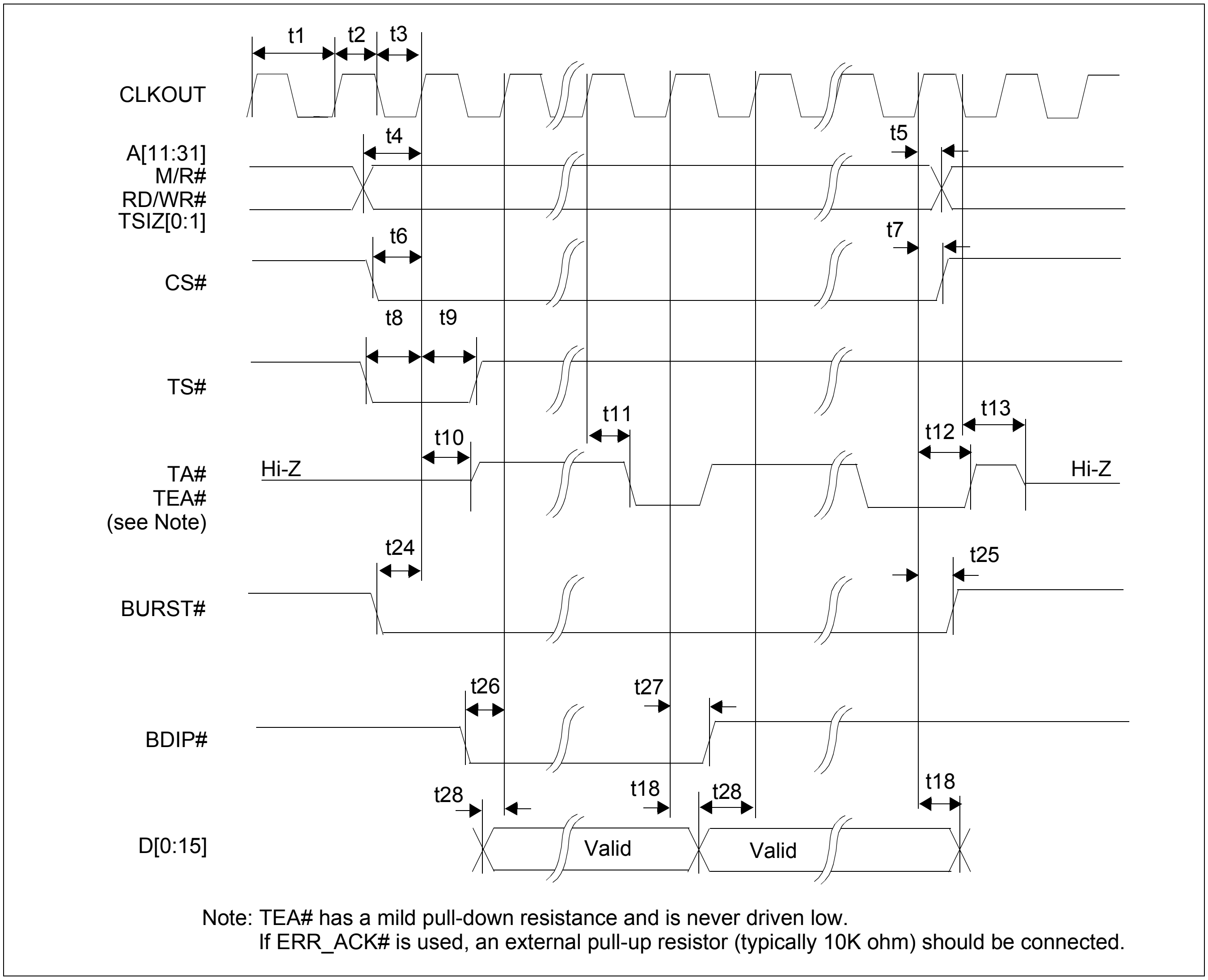


Figure 7-17: Direct/Indirect Freescale MPC555 Host Interface Write Timing (Burst Mode)

Note

1. For Indirect MPC555, the TSIZ0 pin should be tied to “1” and TSIZ1 should be tied to “0” (16-bit host access is mandatory). For byte access in this mode, refer to Section 21.9, “MPC555 Interface” on page 526 note 3.
2. The S1D13515/S2D13515 does not support Big Endian configuration. The host processor must be configured for Little Endian External Bus when connected to the S1D13515/S2D13515 using the MPC555 interface.

Table 7-18: Direct/Indirect Freescale MPC555 Host Interface Write Timing (Burst Mode)

Symbol	Parameter	HIOVDD = 2.5V		HIOVDD = 3.3V		Units
		Min	Max	Min	Max	
fCLKOUT	Clock frequency	—	25	—	25	MHz
t1	Clock period	40	—	40	—	ns
t2	Clock pulse width high	20	—	20	—	ns
t3	Clock pulse width low	20	—	20	—	ns
t4	A[11:31], M/R#, RD/WR#, TSIZ[0:1] setup	7	—	7	—	ns
t5	A[11:31], M/R#, RD/WR#, TSIZ[0:1] hold	0	—	0	—	ns
t6	CS# setup	5	—	5	—	ns
t7	CS# hold	0	—	0	—	ns
t8	TS# setup	8	—	8	—	ns
t9	TS# hold	0	—	0	—	ns
t10	CLKOUT to TA#, TEA# driven for REG[003Dh] bit 0 = 0b	4	24	4	21	ns
	for REG[003Dh] bit 0 = 1b	4	23	4	21	ns
t11	CLKOUT to TA#, TEA# low for REG[003Dh] bit 0 = 0b	—	20	—	17	ns
	for REG[003Dh] bit 0 = 1b	—	19	—	17	ns
t12	CLKOUT to TA#, TEA# high for REG[003Dh] bit 0 = 0b	6	20	5	17	ns
	for REG[003Dh] bit 0 = 1b	5	19	5	17	ns
t13	Negative edge CLKOUT to TA#, TEA# tristate for REG[003Dh] bit 0 = 0b	4	24	4	21	ns
	for REG[003Dh] bit 0 = 1b	4	23	4	20	ns
t18	CLKOUT to D[0:15] hold	0	—	0	—	ns
t24	BURST# setup	7	—	7	—	ns
t25	BURST# hold	0	—	0	—	ns
t26	BDIP# setup	8	—	8	—	ns
t27	BDIP# hold	0	—	0	—	ns
t28	D[0:15] setup to CLKOUT	0	—	0	—	ns

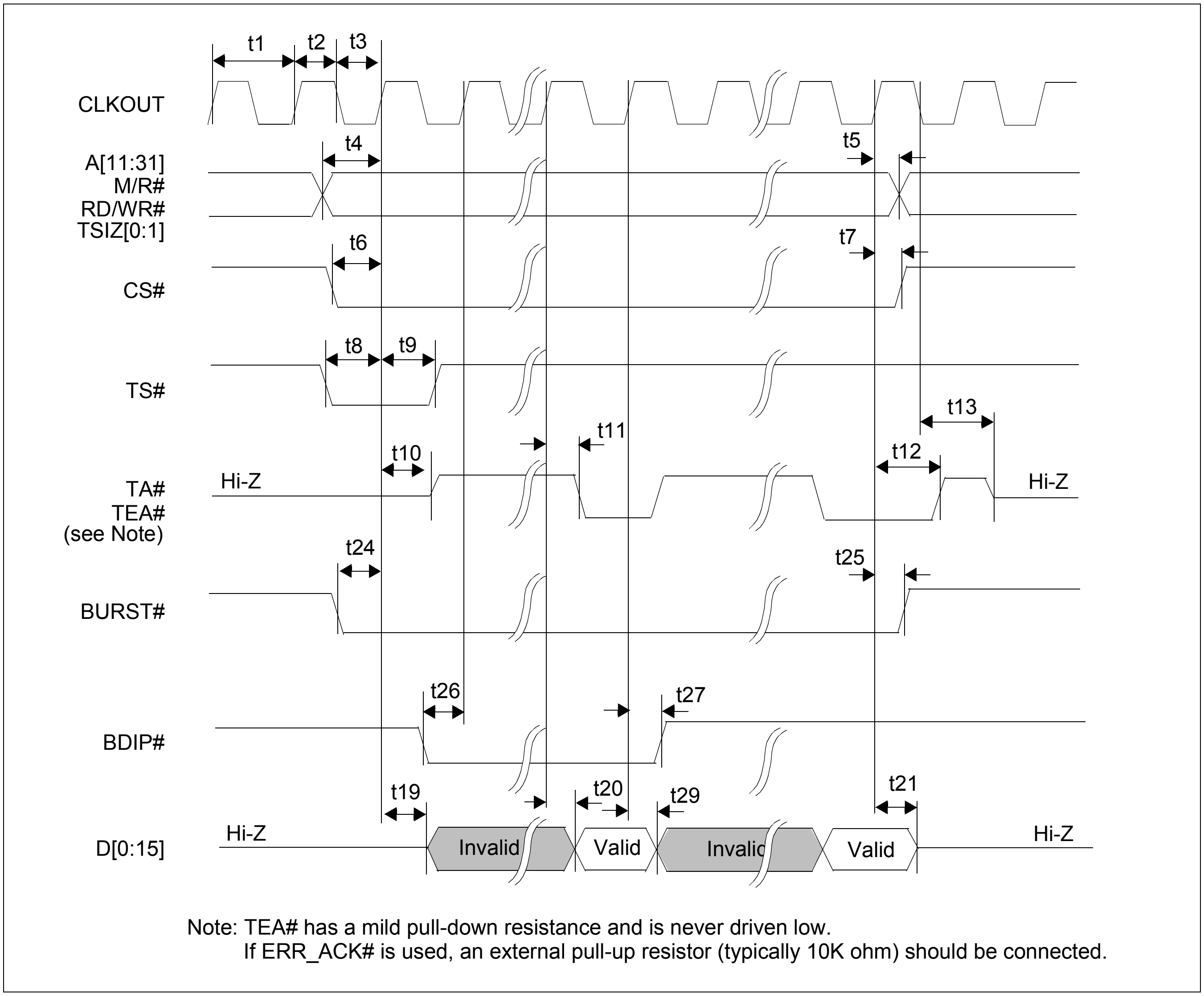


Figure 7-18: Direct/Indirect Freescale MPC555 Host Interface Read Timing (Burst Mode)

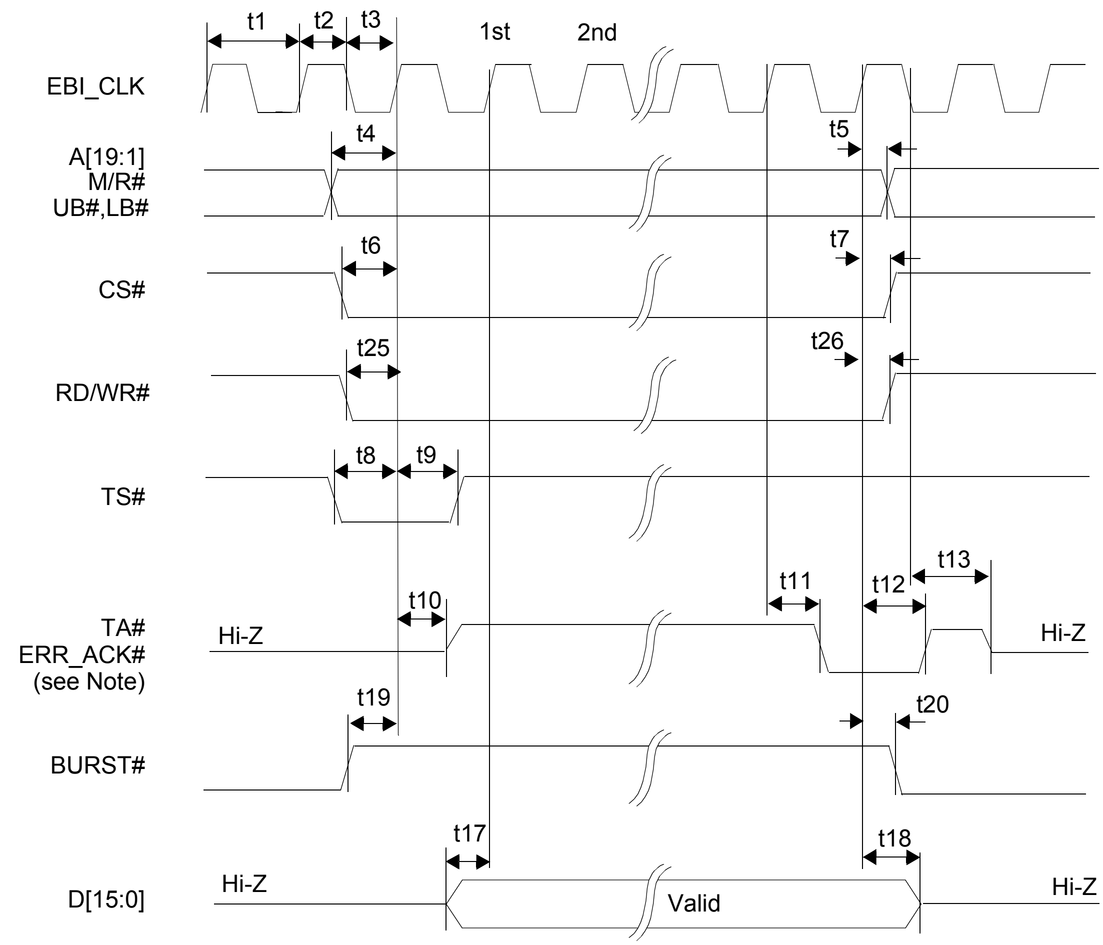
Note

1. For Indirect MPC555, the TSIZ0 pin should be tied to “1” and TSIZ1 should be tied to “0” (16-bit host access is mandatory). For byte access in this mode, refer to Section 21.9, “MPC555 Interface” on page 526 note 3.
2. The S1D13515/S2D13515 does not support Big Endian configuration. The host processor must be configured for Little Endian External Bus when connected to the S1D13515/S2D13515 using the MPC555 interface.

Table 7-19: Direct/Indirect Freescale MPC555 Host Interface Read Timing (Burst Mode)

Symbol	Parameter	HIOVDD = 2.5V		HIOVDD = 3.3V		Units
		Min	Max	Min	Max	
fCLKOUT	Clock frequency	—	25	—	25	MHz
t1	Clock period	40	—	40	—	ns
t2	Clock pulse width high	20	—	20	—	ns
t3	Clock pulse width low	20	—	20	—	ns
t4	A[11:31], M/R#, RD/WR#, TSIZ[0:1] setup	7	—	7	—	ns
t5	A[11:31], M/R#, RD/WR#, TSIZ[0:1] hold	0	—	0	—	ns
t6	CS# setup	5	—	5	—	ns
t7	CS# hold	0	—	0	—	ns
t8	TS# setup	8	—	8	—	ns
t9	TS# hold	0	—	0	—	ns
t10	CLKOUT to TA#, TEA# driven for REG[003Dh] bit 0 = 0b	4	24	4	21	ns
	for REG[003Dh] bit 0 = 1b	4	23	4	21	ns
t11	CLKOUT to TA#, TEA# low for REG[003Dh] bit 0 = 0b	—	20	—	17	ns
	for REG[003Dh] bit 0 = 1b	—	19	—	17	ns
t12	CLKOUT to TA#, TEA# high for REG[003Dh] bit 0 = 0b	6	20	5	17	ns
	for REG[003Dh] bit 0 = 1b	5	19	5	17	ns
t13	Negative edge CLKOUT to TA#, TEA# tristate for REG[003Dh] bit 0 = 0b	4	24	4	21	ns
	for REG[003Dh] bit 0 = 1b	4	23	4	20	ns
t19	CLKOUT to D[0:15] driven for REG[003Dh] bit 0 = 0b	5	—	5	—	ns
	for REG[003Dh] bit 0 = 1b	5	—	5	—	ns
t20	CLKOUT to D[0:15] valid for REG[003Dh] bit 0 = 0b	—	20	—	17	ns
	for REG[003Dh] bit 0 = 1b	—	19	—	16	ns
t21	CLKOUT to D[0:15] tristate for REG[003Dh] bit 0 = 0b	5	25	5	23	ns
	for REG[003Dh] bit 0 = 1b	5	24	5	22	ns
t24	BURST# setup	7	—	7	—	ns
t25	BURST# hold	0	—	0	—	ns
t26	BDIP# setup	8	—	8	—	ns
t27	BDIP# hold	0	—	0	—	ns
t29	CLKOUT to D[0:15] delay for REG[003Dh] bit 0 = 0b	5	—	5	—	ns
	for REG[003Dh] bit 0 = 1b	5	—	5	—	ns

7.4.7 Direct/Indirect TI TSM470 (Non-burst Mode)



Note: ERR_ACK# has a mild pull-down resistance and is never driven low.
If ERR_ACK# is used, an external pull-up resistor (typically 10K ohm) should be connected.

Figure 7-19: Direct/Indirect TI TSM470 Host Interface Write Timing (Non-burst Mode)

Note
For Indirect TI TMS470, the UB# and LB# pins should be tied to “0’ (16-bit host access is mandatory). For byte access in this mode, refer to Section 21.8, “TI TMS470 Interface” on page 525, note 3.

Table 7-20: Direct/Indirect TI TSM470 Host Interface Write Timing (Non-burst Mode)

Symbol	Parameter	HIOVDD = 2.5V		HIOVDD = 3.3V		Units
		Min	Max	Min	Max	
fEBI_CLK	Clock frequency	—	25	—	25	MHz
t1	Clock period	40	—	40	—	ns
t2	Clock pulse width high	20	—	20	—	ns
t3	Clock pulse width low	20	—	20	—	ns
t4	A[19:1], M/R#, UB#/LB# setup	7	—	7	—	ns
t5	A[19:1], M/R#, UB#/LB# hold	0	—	0	—	ns
t6	CS# setup	5	—	5	—	ns
t7	CS# hold	0	—	0	—	ns
t8	TS# setup	8	—	8	—	ns
t9	TS# hold	0	—	0	—	ns
t10	EBI_CLK to TA#, ERR_ACK# driven for REG[003Dh] bit 0 = 0b	4	24	4	22	ns
	for REG[003Dh] bit 0 = 1b	4	23	4	21	ns
t11	EBI_CLK to TA#, ERR_ACK# low for REG[003Dh] bit 0 = 0b	—	20	—	17	ns
	for REG[003Dh] bit 0 = 1b	—	19	—	17	ns
t12	EBI_CLK to TA#, ERR_ACK# high for REG[003Dh] bit 0 = 0b	6	20	5	17	ns
	for REG[003Dh] bit 0 = 1b	5	19	5	17	ns
t13	Negative edge EBI_CLK to TA#, ERR_ACK# tristate for REG[003Dh] bit 0 = 0b	4	24	4	21	ns
	for REG[003Dh] bit 0 = 1b	4	23	4	20	ns
t17	D[15:0] setup to 1st EBI_CLK after TS#=0	0	—	0	—	ns
t18	EBI_CLK to D[15:0] hold	0	—	0	—	ns
t19	BURST# setup	7	—	7	—	ns
t20	BURST# hold	0	—	0	—	ns
t25	RD/WR# setup	7	—	7	—	ns
t26	RD/WR# hold	0	—	0	—	ns

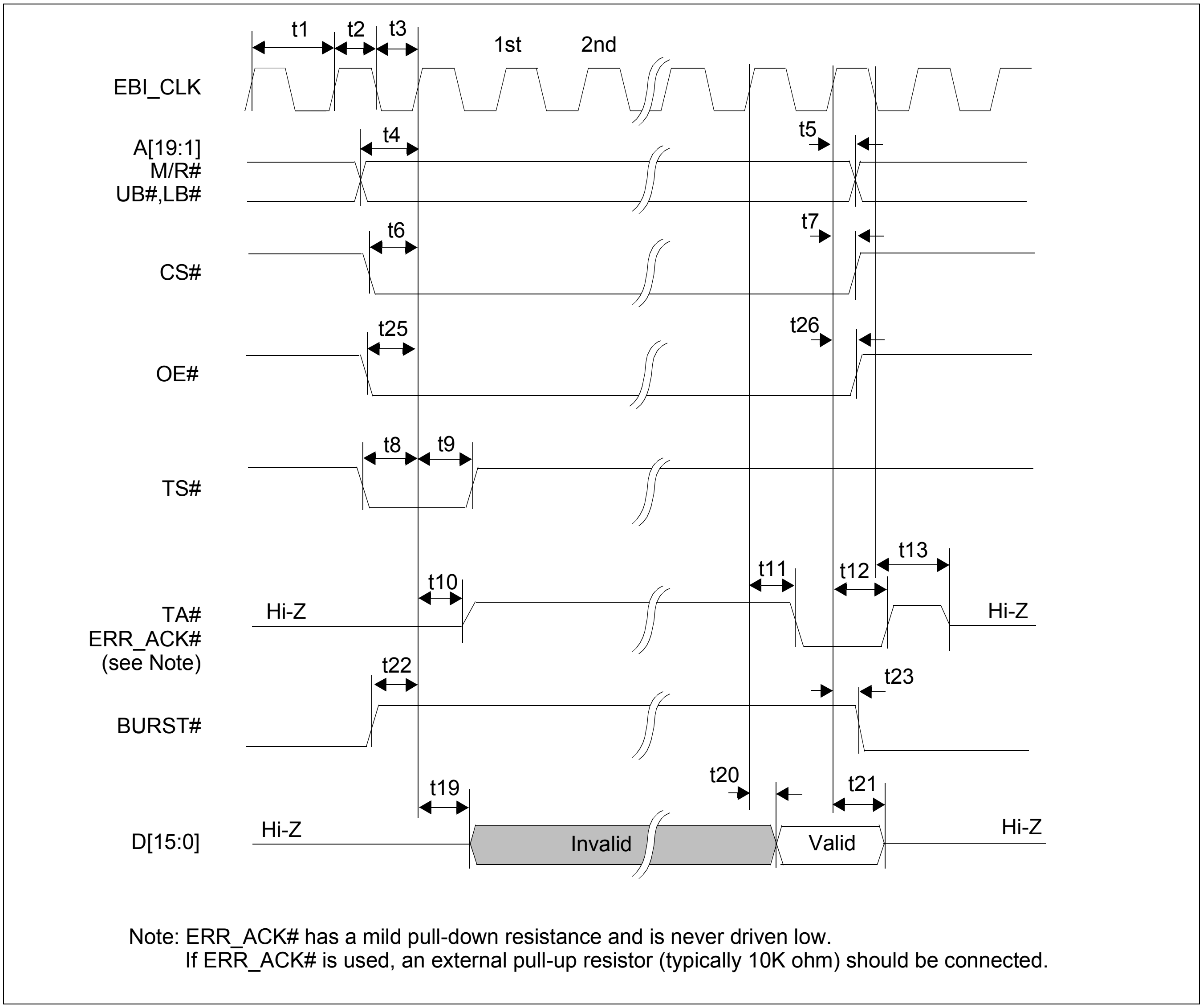


Figure 7-20: Direct/Indirect TI TSM470 Host Interface Read Timing (Non-burst Mode)

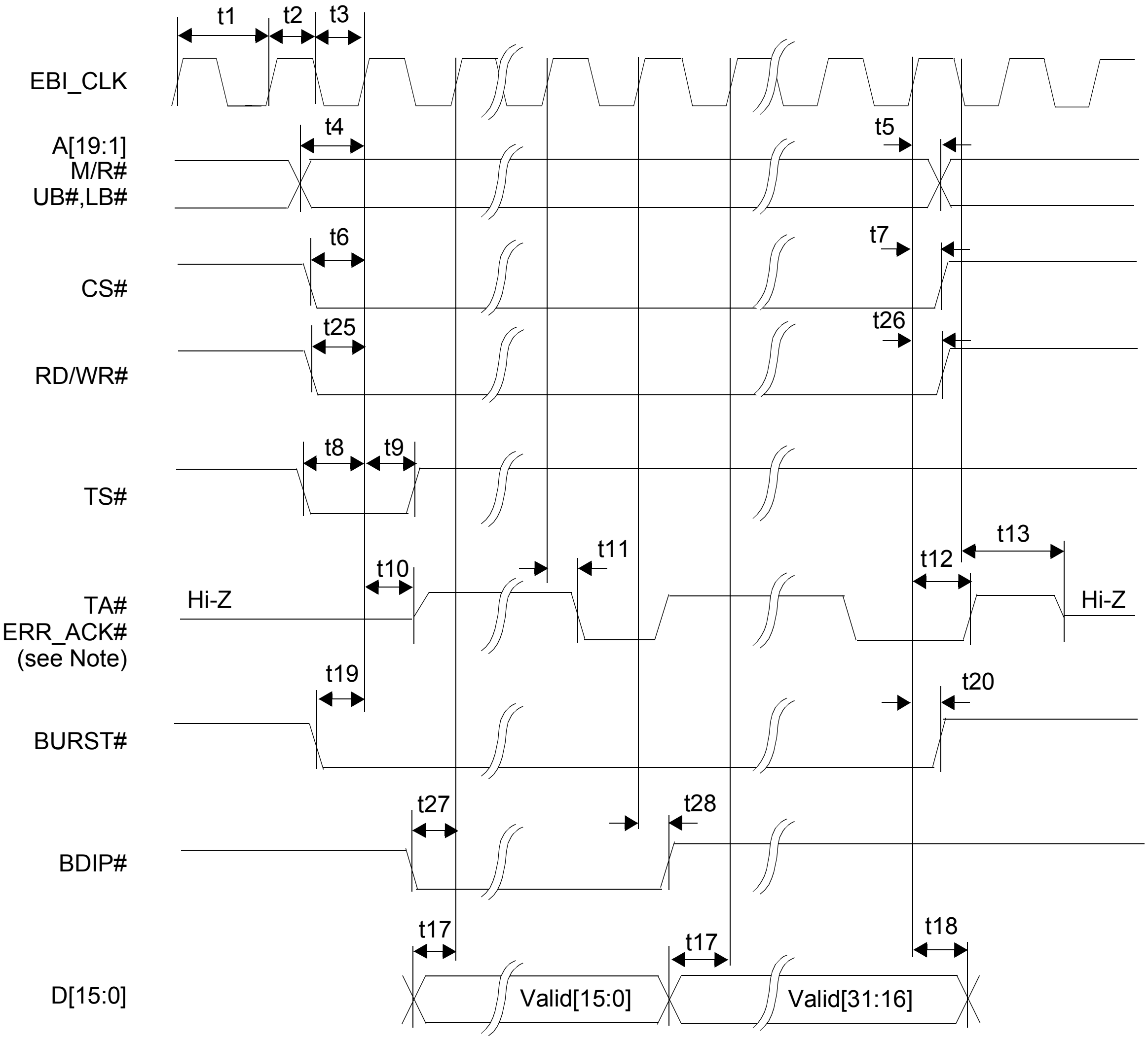
Note

For Indirect TI TMS470, the UB# and LB# pins should be tied to "0' (16-bit host access is mandatory). For byte access in this mode, refer to Section 21.8, "TI TMS470 Interface" on page 525, note 3.

Table 7-21: Direct/Indirect TI TSM470 Host Interface Read Timing (Non-burst Mode)

Symbol	Parameter	HIOVDD = 2.5V		HIOVDD = 3.3V		Units
		Min	Max	Min	Max	
fEBI_CLK	Clock frequency	—	25	—	25	MHz
t1	Clock period	40	—	40	—	ns
t2	Clock pulse width high	20	—	20	—	ns
t3	Clock pulse width low	20	—	20	—	ns
t4	A[19:1], M/R#, UB#/LB# setup	7	—	7	—	ns
t5	A[19:1], M/R#, UB#/LB# hold	0	—	0	—	ns
t6	CS# setup	5	—	5	—	ns
t7	CS# hold	0	—	0	—	ns
t8	TS# setup	8	—	8	—	ns
t9	TS# hold	0	—	0	—	ns
t10	EBI_CLK to TA#, ERR_ACK# driven for REG[003Dh] bit 0 = 0b	4	24	4	22	ns
	for REG[003Dh] bit 0 = 1b	4	23	4	21	ns
t11	EBI_CLK to TA#, ERR_ACK# low for REG[003Dh] bit 0 = 0b	—	20	—	17	ns
	for REG[003Dh] bit 0 = 1b	—	19	—	17	ns
t12	EBI_CLK to TA#, ERR_ACK# high for REG[003Dh] bit 0 = 0b	6	20	5	17	ns
	for REG[003Dh] bit 0 = 1b	5	19	5	17	ns
t13	Negative edge EBI_CLK to TA#, ERR_ACK# tristate for REG[003Dh] bit 0 = 0b	4	24	4	21	ns
	for REG[003Dh] bit 0 = 1b	4	23	4	20	ns
t19	EBI_CLK to D[15:0] driven for REG[003Dh] bit 0 = 0b	4	—	4	—	ns
	for REG[003Dh] bit 0 = 1b	4	—	4	—	ns
t20	EBI_CLK to D[15:0] valid for REG[003Dh] bit 0 = 0b	—	20	—	17	ns
	for REG[003Dh] bit 0 = 1b	—	19	—	17	ns
t21	EBI_CLK to D[15:0] tristate for REG[003Dh] bit 0 = 0b	4	25	4	23	ns
	for REG[003Dh] bit 0 = 1b	4	24	4	22	ns
t22	BURST# setup	7	—	7	—	ns
t23	BURST# hold	0	—	0	—	ns
t25	OE# setup	10	—	10	—	ns
t26	OE# hold	0	—	0	—	ns

7.4.8 Direct/Indirect TI TSM470 (Burst Mode)



Note: ERR_ACK# has a mild pull-down resistance and is never driven low.
If ERR_ACK# is used, an external pull-up resistor (typically 10K ohm) should be connected.

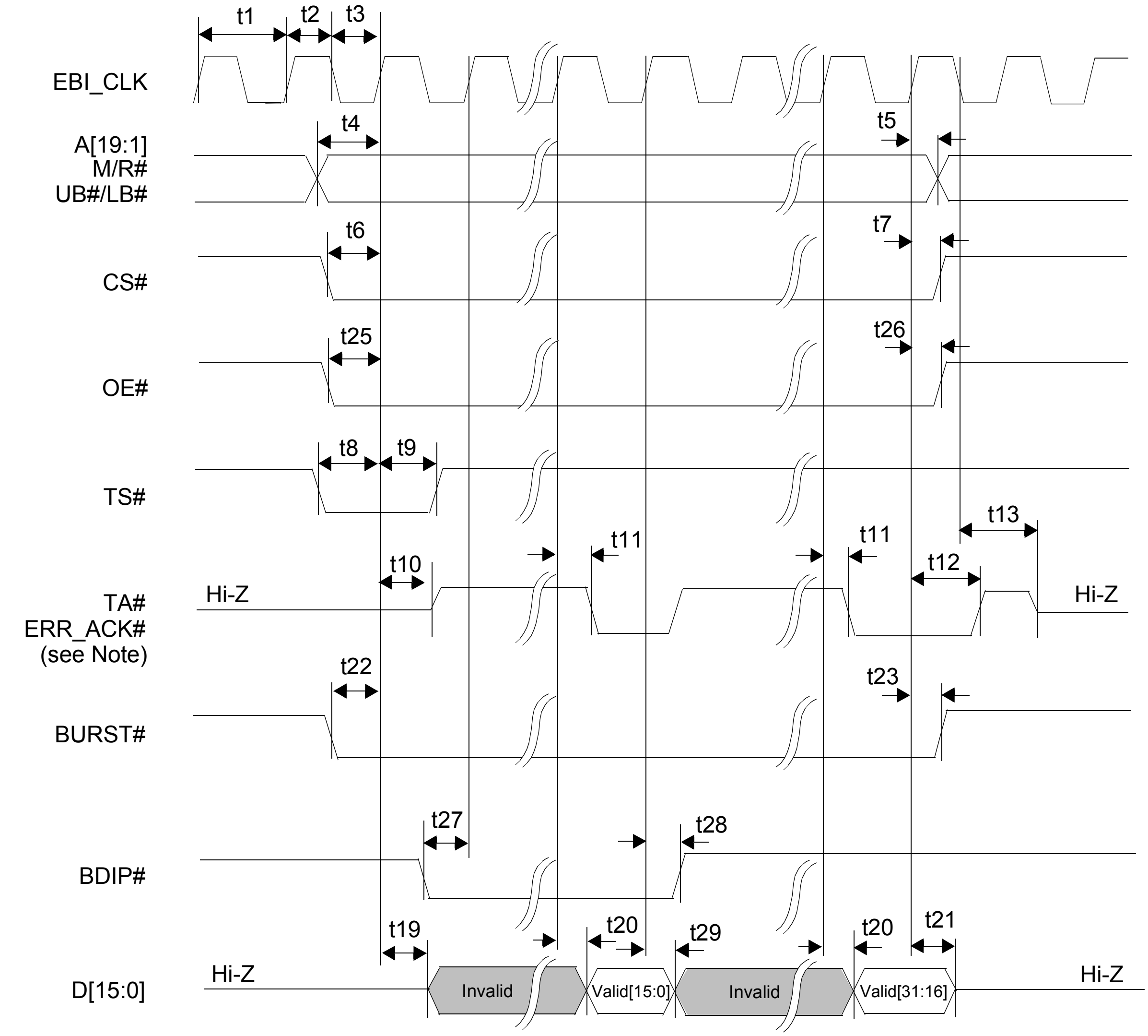
Figure 7-21: Direct/Indirect TI TSM470 Host Interface Write Timing (Burst Mode)

Note

For Indirect TI TMS470, the UB# and LB# pins should be tied to "0' (16-bit host access is mandatory). For byte access in this mode, refer to Section 21.8, "TI TMS470 Interface" on page 525, note 3.

Table 7-22: Direct/Indirect TI TSM470 Host Interface Write Timing (Burst Mode)

Symbol	Parameter	HIOVDD = 2.5V		HIOVDD = 3.3V		Units
		Min	Max	Min	Max	
fEBI_CLK	Clock frequency	—	25	—	25	MHz
t1	Clock period	40	—	40	—	ns
t2	Clock pulse width high	20	—	20	—	ns
t3	Clock pulse width low	20	—	20	—	ns
t4	A[19:1], M/R#, UB#/LB# setup	7	—	7	—	ns
t5	A[19:1], M/R#, UB#/LB# hold	0	—	0	—	ns
t6	CS# setup	5	—	5	—	ns
t7	CS# hold	0	—	0	—	ns
t8	TS# setup	8	—	8	—	ns
t9	TS# hold	0	—	0	—	ns
t10	EBI_CLK to TA#, ERR_ACK# driven for REG[003Dh] bit 0 = 0b	4	24	4	22	ns
	for REG[003Dh] bit 0 = 1b	4	23	4	21	ns
t11	EBI_CLK to TA#, ERR_ACK# low for REG[003Dh] bit 0 = 0b	—	20	—	17	ns
	for REG[003Dh] bit 0 = 1b	—	19	—	17	ns
t12	EBI_CLK to TA#, ERR_ACK# high for REG[003Dh] bit 0 = 0b	6	20	5	17	ns
	for REG[003Dh] bit 0 = 1b	5	19	5	17	ns
t13	Negative edge EBI_CLK to TA#, ERR_ACK# tristate for REG[003Dh] bit 0 = 0b	4	24	4	21	ns
	for REG[003Dh] bit 0 = 1b	4	23	4	20	ns
t17	D[15:0] setup to EBI_CLK	0	—	0	—	ns
t18	EBI_CLK to D[15:0] hold	0	—	0	—	ns
t19	BURST# setup	7	—	7	—	ns
t20	BURST# hold	0	—	0	—	ns
t25	RD/WR# setup	7	—	7	—	ns
t26	RD/WR# hold	0	—	0	—	ns
t27	BDIP# setup	8	—	8	—	ns
t28	BDIP# hold	0	—	0	—	ns



Note: ERR_ACK# has a mild pull-down resistance and is never driven low.
If ERR_ACK# is used, an external pull-up resistor (typically 10K ohm) should be connected.

Figure 7-22: Direct/Indirect TI TSM470 Host Interface Read Timing (Burst Mode)

Note

For Indirect TI TMS470, the UB# and LB# pins should be tied to “0’ (16-bit host access is mandatory). For byte access in this mode, refer to Section 21.8, “TI TMS470 Interface” on page 525, note 3.

Table 7-23: Direct/Indirect TI TSM470 Host Interface Read Timing (Burst Mode)

Symbol	Parameter	HIOVDD = 2.5V		HIOVDD = 3.3V		Units
		Min	Max	Min	Max	
fEBI_CLK	Clock frequency	—	25	—	25	MHz
t1	Clock period	40	—	40	—	ns
t2	Clock pulse width high	20	—	20	—	ns
t3	Clock pulse width low	20	—	20	—	ns
t4	A[19:1], M/R#, UB#/LB# setup	7	—	7	—	ns
t5	A[19:1], M/R#, UB#/LB# hold	0	—	0	—	ns
t6	CS# setup	5	—	5	—	ns
t7	CS# hold	0	—	0	—	ns
t8	TS# setup	8	—	8	—	ns
t9	TS# hold	0	—	0	—	ns
t10	EBI_CLK to TA#, ERR_ACK# driven for REG[003Dh] bit 0 = 0b	4	24	4	22	ns
	for REG[003Dh] bit 0 = 1b	4	23	4	21	ns
t11	EBI_CLK to TA#, ERR_ACK# low for REG[003Dh] bit 0 = 0b	—	20	—	17	ns
	for REG[003Dh] bit 0 = 1b	—	19	—	17	ns
t12	EBI_CLK to TA#, ERR_ACK# high for REG[003Dh] bit 0 = 0b	6	20	5	17	ns
	for REG[003Dh] bit 0 = 1b	5	19	5	17	ns
t13	Negative edge EBI_CLK to TA#, ERR_ACK# tristate for REG[003Dh] bit 0 = 0b	4	24	4	21	ns
	for REG[003Dh] bit 0 = 1b	4	23	4	20	ns
t19	EBI_CLK to D[15:0] driven for REG[003Dh] bit 0 = 0b	4	—	4	—	ns
	for REG[003Dh] bit 0 = 1b	4	—	4	—	ns
t20	EBI_CLK to D[15:0] valid for REG[003Dh] bit 0 = 0b	—	20	—	17	ns
	for REG[003Dh] bit 0 = 1b	—	19	—	17	ns
t21	EBI_CLK to D[15:0] tristate for REG[003Dh] bit 0 = 0b	4	25	4	23	ns
	for REG[003Dh] bit 0 = 1b	4	24	4	22	ns
t22	BURST# setup	7	—	7	—	ns
t23	BURST# hold	0	—	0	—	ns
t25	OE# setup	10	—	10	—	ns
t26	OE# hold	0	—	0	—	ns
t27	BDIP# setup	8	—	8	—	ns
t28	BDIP# hold	0	—	0	—	ns
t29	EBI_CLK to D[15:0] delay for REG[003Dh] bit 0 = 0b	4	—	4	—	ns
	for REG[003Dh] bit 0 = 1b	4	—	4	—	ns

7.4.9 Direct/Indirect NEC V850 Type 1

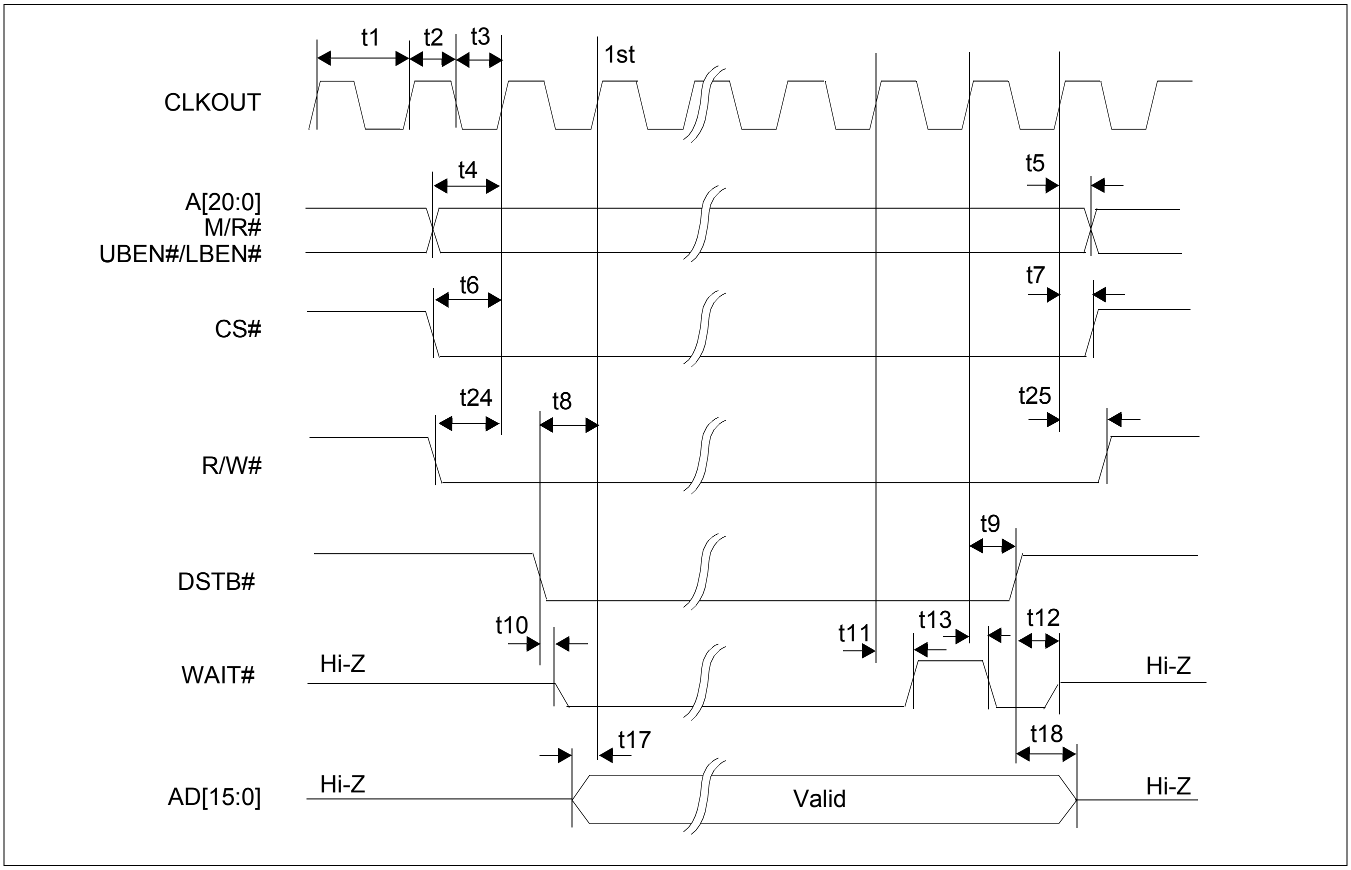


Figure 7-23: Direct/Indirect NEC V850 Type 1 Host Interface Write Timing

Note
For Indirect NECV850 Type #1 8-bit, the UBEN# and LBEN# pins are not used.
For Indirect NECV850 Type #1 16-bit, the UBEN# and LBEN# pins should be tied to logic 0. For byte access in this mode, refer to Section 21.4, "NEC V850 Type1 Interface" on page 521, note 2.

Table 7-24: Direct/Indirect NEC V850 Type 1 Host Interface Write Timing

Symbol	Parameter	HIOVDD = 2.5V		HIOVDD = 3.3V		Units
		Min	Max	Min	Max	
fCLKOUT	Clock frequency	—	20	—	20	MHz
t1	Clock period	50	—	50	—	ns
t2	Clock pulse width high	25	—	25	—	ns
t3	Clock pulse width low	25	—	25	—	ns
t4	A[20:0], M/R#, UBEN#/LBEN# setup	10	—	10	—	ns
t5	A[20:0], M/R#, UBEN#/LBEN# hold	0	—	0	—	ns
t6	CS# setup	10	—	10	—	ns
t7	CS# hold	0	—	0	—	ns
t8	DSTB# setup	11	—	11	—	ns
t9	DSTB# hold	-8	—	-8	—	ns
t10	DSTB# falling edge to WAIT# driven for REG[003Dh] bit 0 = 0b	6	28	6	25	ns
	for REG[003Dh] bit 0 = 1b	5	27	5	24	ns
t11	CLKOUT to WAIT# high for REG[003Dh] bit 0 = 0b	—	20	—	17	ns
	for REG[003Dh] bit 0 = 1b	—	19	—	17	ns
t12	DSTB# rising edge to WAIT# tristate for REG[003Dh] bit 0 = 0b	2	19	2	19	ns
	for REG[003Dh] bit 0 = 1b	2	19	2	19	ns
t13	CLKOUT to WAIT# low for REG[003Dh] bit 0 = 0b	4	—	4	—	ns
	for REG[003Dh] bit 0 = 1b	4	—	4	—	ns
t17	AD[15:0] write data setup to 1st CLKOUT	0	—	0	—	ns
t18	DSTB# rising edge to AD[15:0] hold	0	—	0	—	ns
t24	R/W# setup	10	—	10	—	ns
t25	R/W# hold	0	—	0	—	ns

Note

1. When the CLKOUT period (t1) is short, the V850 may sample an invalid WAIT# status because of the t10 timing. To allow the V850 to sample the WAIT# status correctly, a programmable wait must be inserted. The programmable wait is controlled by the V850 Data Wait Control Register (DWC). For details on V850 registers, refer to the V850 specification.
2. When the S1D13515/S2D13515 completes a write, WAIT# is driven high and then asserted low 1 CLKOUT later. This means that WAIT# is only high for a 1 CLKOUT period. To sample WAIT# high correctly, the V850 Data Wait Control Register (DWC) must be set appropriately. For details on V850 registers, refer to the V850 specification.

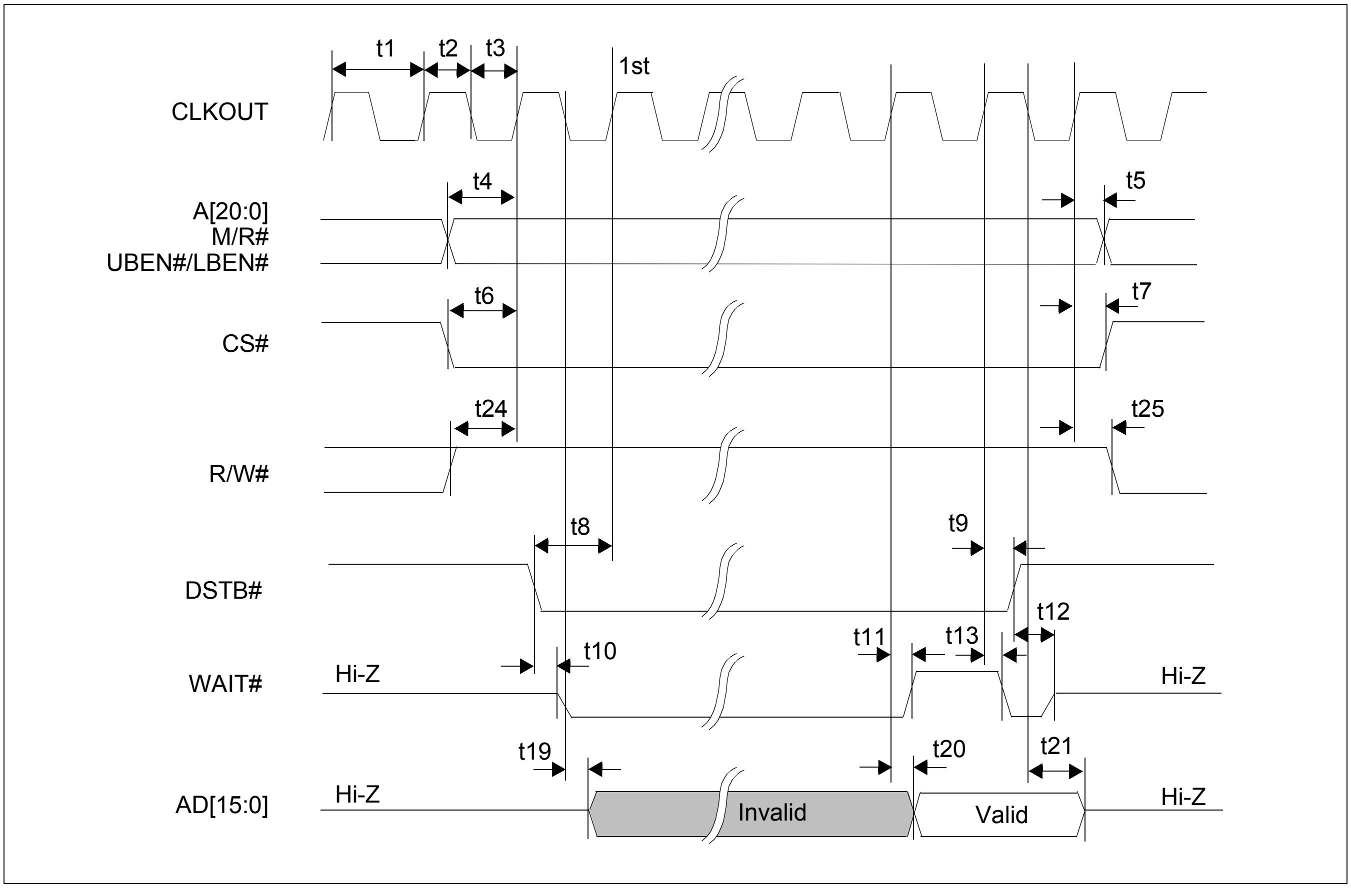


Figure 7-24: Direct/Indirect NEC V850 Type 1 Host Interface Read Timing

Note

For Indirect NECV850 Type #1 8-bit, the UBEN# and LBEN# pins are not used.
For Indirect NECV850 Type #1 16-bit, the UBEN# and LBEN# pins should be tied to logic 0. For byte access in this mode, refer to Section 21.4, “NEC V850 Type1 Interface” on page 521, note 2.

Table 7-25: Direct/Indirect NEC V850 Type 1 Host Interface Read Timing

Symbol	Parameter	HIOVDD = 2.5V		HIOVDD = 3.3V		Units
		Min	Max	Min	Max	
fCLKOUT	Clock frequency	—	20	—	20	MHz
t1	Clock period	50	—	50	—	ns
t2	Clock pulse width high	25	—	25	—	ns
t3	Clock pulse width low	25	—	25	—	ns
t4	A[20:0], M/R#, UBEN#/LBEN# setup	10	—	10	—	ns
t5	A[20:0], M/R#, UBEN#/LBEN# hold	0	—	0	—	ns
t6	CS# setup	10	—	10	—	ns
t7	CS# hold	0	—	0	—	ns
t8	DSTB# setup	11	—	11	—	ns
t9	DSTB# hold	-8	—	-8	—	ns
t10	DSTB# falling edge to WAIT# driven for REG[003Dh] bit 0 = 0b	6	28	6	25	ns
	for REG[003Dh] bit 0 = 1b	5	27	5	24	ns
t11	CLKOUT to WAIT# high for REG[003Dh] bit 0 = 0b	—	20	—	17	ns
	for REG[003Dh] bit 0 = 1b	—	19	—	17	ns
t12	DSTB# rising edge to WAIT# tristate for REG[003Dh] bit 0 = 0b	2	19	2	19	ns
	for REG[003Dh] bit 0 = 1b	2	19	2	19	ns
t13	CLKOUT to WAIT# low for REG[003Dh] bit 0 = 0b	4	—	4	—	ns
	for REG[003Dh] bit 0 = 1b	4	—	4	—	ns
t19	Negative edge CLKOUT to AD[15:0] driven for REG[003Dh] bit 0 = 0b	2	—	2	—	ns
	for REG[003Dh] bit 0 = 1b	2	—	2	—	ns
t20	CLKOUT to AD[15:0] valid for REG[003Dh] bit 0 = 0b	—	20	—	17	ns
	for REG[003Dh] bit 0 = 1b	—	19	—	16	ns
t21	Negative edge CLKOUT to AD[15:0] tristate for REG[003Dh] bit 0 = 0b	2	22	2	19	ns
	for REG[003Dh] bit 0 = 1b	2	21	2	18	ns
t24	R/W# setup	10	—	10	—	ns
t25	R/W# hold	0	—	0	—	ns

Note

1. When the CLKOUT period (t1) is short, the V850 may sample an invalid WAIT# status because of the t10 timing. To allow the V850 to sample the WAIT# status correctly, a programmable wait must be inserted. The programmable wait is controlled by the V850 Data Wait Control Register (DWC). For details on V850 registers, refer to the V850 specification.
2. When read data is ready, WAIT# is driven high and then asserted low 1 CLKOUT later. This means that WAIT# is only high for a 1 CLKOUT period. To sample WAIT# high correctly, the V850 Data Wait Control Register (DWC) must be set appropriately. For details on V850 registers, refer to the V850 specification.

7.4.10 Direct/Indirect NEC V850 Type 2

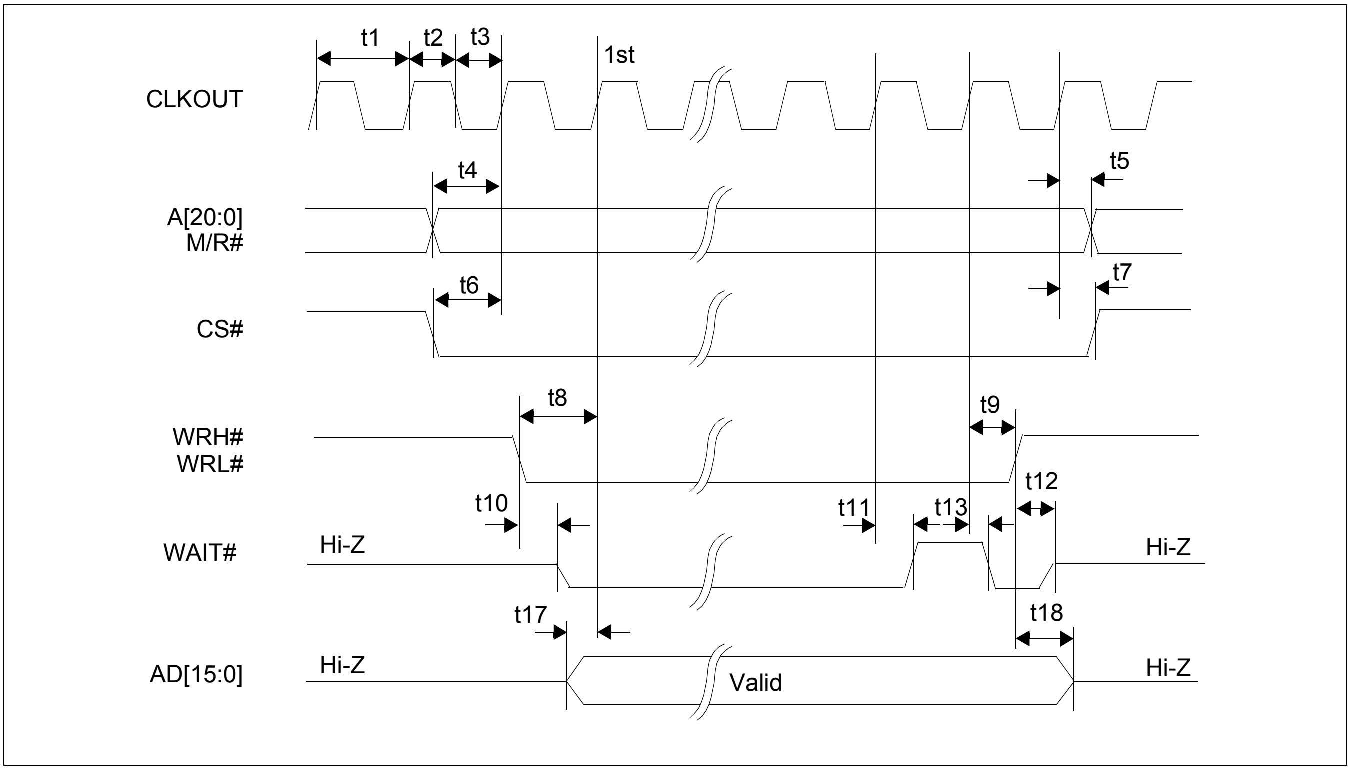


Figure 7-25: Direct/Indirect NEC V850 Type 2 Host Interface Write Timing

Note

For Indirect NEC V850 Type #2 8-bit, the WRH# is not used.
For Indirect NEC V850 Type #2 16-bit, the WRH# and WRL# pins should be driven in unison (16-bit host write access is mandatory). For byte access in this mode, refer to Section 21.5, “NEC V850 Type2 Interface” on page 522, note 2.

Table 7-26: Direct/Indirect NEC V850 Type 2 Host Interface Write Timing

Symbol	Parameter	HIOVDD = 2.5V		HIOVDD = 3.3V		Units
		Min	Max	Min	Max	
fCLKOUT	Clock frequency	—	20	—	20	MHz
t1	Clock period	50	—	50	—	ns
t2	Clock pulse width high	25	—	25	—	ns
t3	Clock pulse width low	25	—	25	—	ns
t4	A[20:0], M/R# setup	10	—	10	—	ns
t5	A[20:0], M/R# hold	0	—	0	—	ns
t6	CS# setup	10	—	10	—	ns
t7	CS# hold	0	—	0	—	ns
t8	WRL#/WRH# setup	8	—	8	—	ns
t9	WRL#/WRH# hold	-8	—	-8	—	ns
t10	WRL#/WRH# falling edge to WAIT# driven for REG[003Dh] bit 0 = 0b	6	24	6	21	ns
	for REG[003Dh] bit 0 = 1b	5	23	5	20	ns
t11	CLKOUT to WAIT# high for REG[003Dh] bit 0 = 0b	—	20	—	17	ns
	for REG[003Dh] bit 0 = 1b	—	19	—	17	ns
t12	WRL#/WRH# rising edge to WAIT# tristate for REG[003Dh] bit 0 = 0b	2	16	2	15	ns
	for REG[003Dh] bit 0 = 1b	2	16	2	15	ns
t13	CLKOUT to WAIT# low for REG[003Dh] bit 0 = 0b	4	—	4	—	ns
	for REG[003Dh] bit 0 = 1b	4	—	4	—	ns
t17	AD[15:0] write data setup to 1st CLKOUT	0	—	0	—	ns
t18	WRL#/WRH# rising edge to AD[15:0] hold	0	—	0	—	ns

Note

1. When the CLKOUT period (t1) is short, the V850 may sample an invalid WAIT# status because of the t10 timing. To allow the V850 to sample the WAIT# status correctly, a programmable wait must be inserted. The programmable wait is controlled by the V850 Data Wait Control Register (DWC). For details on V850 registers, refer to the V850 specification.
2. When the S1D13515/S2D13515 completes a write, WAIT# is driven high and then asserted low 1 CLKOUT later. This means that WAIT# is only high for a 1 CLKOUT period. To sample WAIT# high correctly, the V850 Data Wait Control Register (DWC) must be set appropriately. For details on V850 registers, refer to the V850 specification.

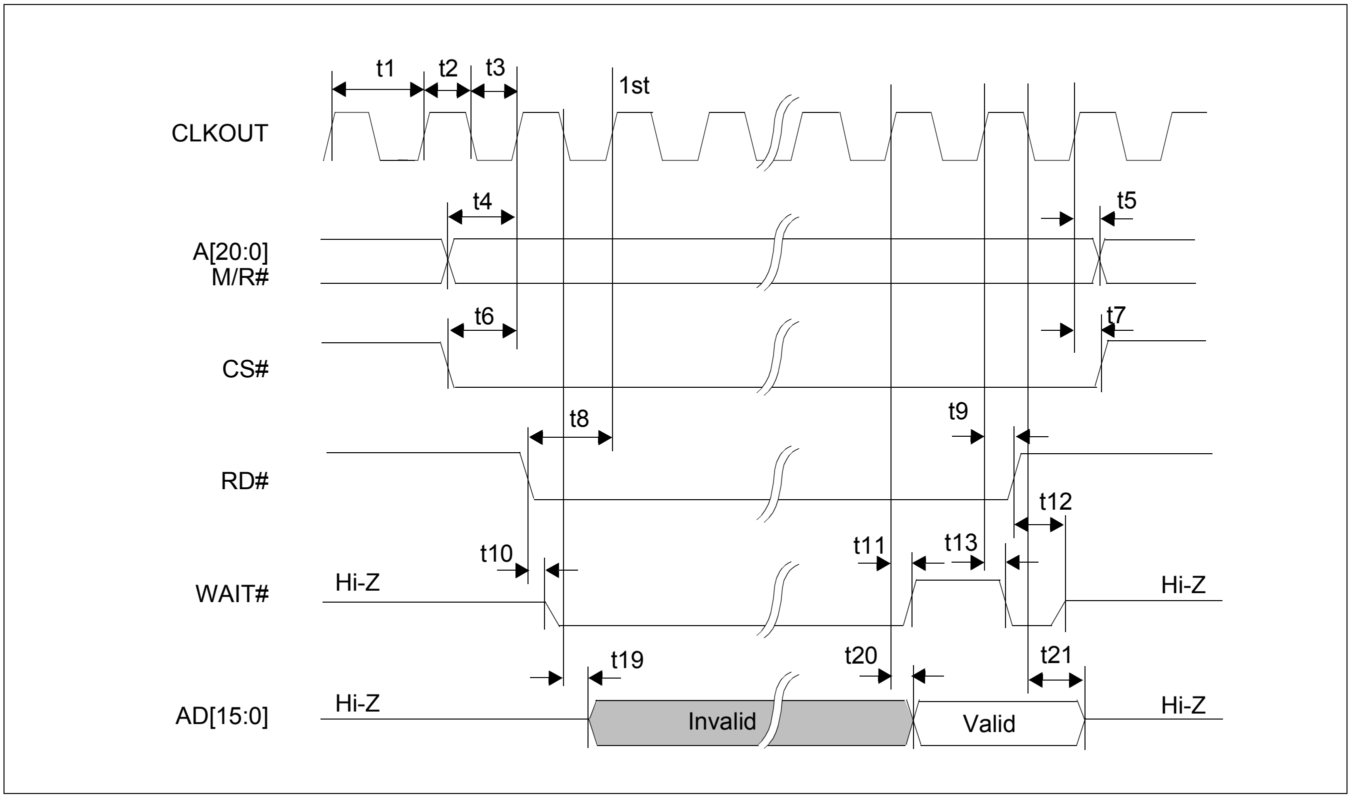


Figure 7-26: Direct/Indirect NEC V850 Type 2 Host Interface Read Timing

Note

For Indirect NEC V850 Type #2 8-bit, the WRH# is not used.

For Indirect NEC V850 Type #2 16-bit, the WRH# and WRL# pins should be driven in unison (16-bit host write access is mandatory). For byte access in this mode, refer to Section 21.5, “NEC V850 Type2 Interface” on page 522, note 2.

Table 7-27: Direct/Indirect NEC V850 Type 2 Host Interface Read Timing

Symbol	Parameter	HIOVDD = 2.5V		HIOVDD = 3.3V		Units
		Min	Max	Min	Max	
fCLKOUT	Clock frequency	—	20	—	20	MHz
t1	Clock period	50	—	50	—	ns
t2	Clock pulse width high	25	—	25	—	ns
t3	Clock pulse width low	25	—	25	—	ns
t4	A[20:0], M/R# setup	10	—	10	—	ns
t5	A[20:0], M/R# hold	0	—	0	—	ns
t6	CS# setup	10	—	10	—	ns
t7	CS# hold	0	—	0	—	ns
t8	RD# setup	11	—	11	—	ns
t9	RD# hold	-8	—	-8	—	ns
t10	RD# falling edge to WAIT# driven for REG[003Dh] bit 0 = 0b	8	28	7	25	ns
	for REG[003Dh] bit 0 = 1b	7	27	7	24	ns
t11	CLKOUT to WAIT# high for REG[003Dh] bit 0 = 0b	—	20	—	17	ns
	for REG[003Dh] bit 0 = 1b	—	19	—	17	ns
t12	RD# rising edge to WAIT# tristate for REG[003Dh] bit 0 = 0b	4	19	4	19	ns
	for REG[003Dh] bit 0 = 1b	4	19	4	19	ns
t13	CLKOUT to WAIT# low for REG[003Dh] bit 0 = 0b	4	—	4	—	ns
	for REG[003Dh] bit 0 = 1b	4	—	4	—	ns
t19	Negative edge CLKOUT to AD[15:0] driven for REG[003Dh] bit 0 = 0b	2	—	2	—	ns
	for REG[003Dh] bit 0 = 1b	2	—	2	—	ns
t20	CLKOUT to AD[15:0] valid for REG[003Dh] bit 0 = 0b	—	20	—	17	ns
	for REG[003Dh] bit 0 = 1b	—	19	—	16	ns
t21	Negative edge CLKOUT to AD[15:0] tristate for REG[003Dh] bit 0 = 0b	2	22	2	19	ns
	for REG[003Dh] bit 0 = 1b	2	21	2	18	ns

Note

1. When the CLKOUT period (t1) is short, the V850 may sample an invalid WAIT# status because of the t10 timing. To allow the V850 to sample the WAIT# status correctly, a programmable wait must be inserted. The programmable wait is controlled by the V850 Data Wait Control Register (DWC). For details on V850 registers, refer to the V850 specification.
2. When read data is ready, WAIT# is driven high and then asserted low 1 CLKOUT later. This means that WAIT# is only high for a 1 CLKOUT period. To sample WAIT# high correctly, the V850 Data Wait Control Register (DWC) must be set appropriately. For details on V850 registers, refer to the V850 specification.

7.5 Serial Host Bus Interface Timing

7.5.1 SPI

The SPI host module requires a valid clock selection before the interface can operate. The SPI host module clock selection is determined by a combination of SPICLKEN (AB5) pin and REG[0061h] bits 2 and 0.

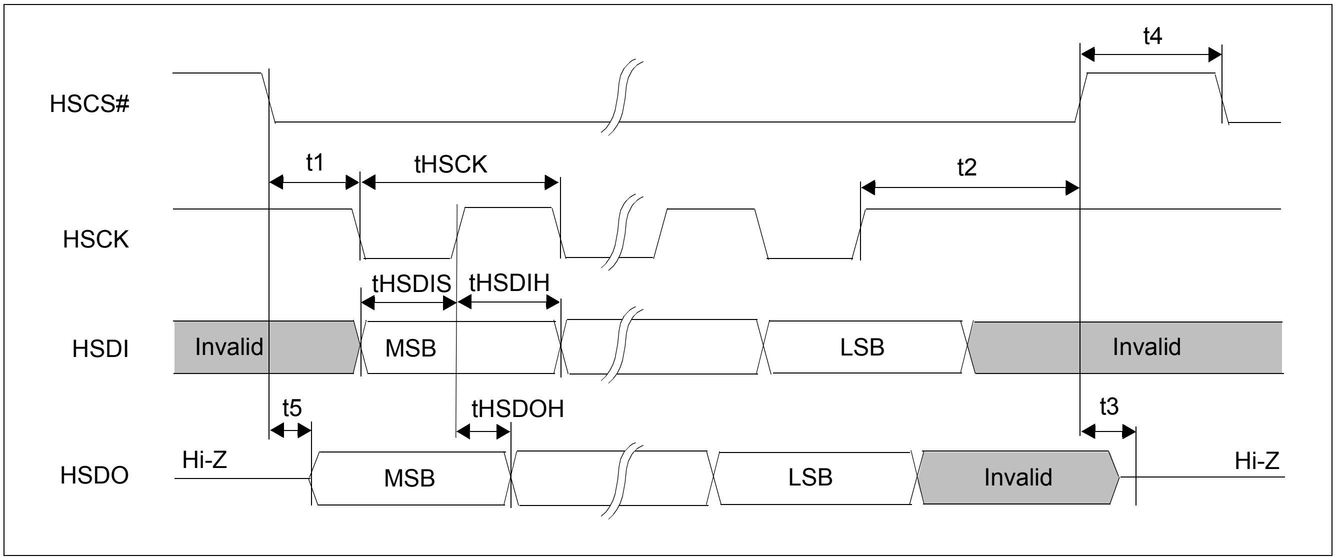


Figure 7-27: SPI Host Interface Timing

Table 7-28: SPI Host Interface Timing

Symbol	Parameter	HIOVDD = 2.5V		HIOVDD = 3.3V		Units
		Min	Max	Min	Max	
fHSCK	HSCK Clock frequency	—	10	—	10	MHz
tHSCK	HSCK Clock period (Note 2)	100	—	100	—	ns
tHSDIS	HSDI data setup time	3	—	3	—	ns
tHSDIH	HSDI data hold time	3	—	3	—	ns
tHSDOH	HSDO data hold time for REG[003Dh] bit 0 = 0b	5	—	5	—	ns
	for REG[003Dh] bit 0 = 1b	5	—	5	—	ns
t1	HSCS# falling edge to HSCK falling edge	5	—	5	—	ns
t2	HSCK rising edge to HSCS# rising edge	2	—	2	—	ClkSPI (Note1)
t3	HSCS# rising edge to HSDO tristate for REG[003Dh] bit 0 = 0b	3	11	3	10	ns
	for REG[003Dh] bit 0 = 1b	3	11	3	10	ns
t4	HSCS# rising edge to HSCS# falling edge	1	—	1	—	tHSCK
t5	HSCS# falling edge to HSDO driven for REG[003Dh] bit 0 = 0b	6	19	6	16	ns
	for REG[003Dh] bit 0 = 1b	5	18	5	16	ns

1. ClkSPI = SPI control module clock period
2. The user must select a HSCK (Serial Clock) frequency, ClkSPI (SPI control module clock) frequency and System Clock frequency that meet the following equation.

   For synchronous register access:

   8 HSCK cycles ≥ X + 7 ClkSPI cycles + 5 System Clock cycles

   where X is:

   0 if the DMA Controller is not running AND C33 processor is not running;

   16 system clocks if the DMA Controller is transferring data AND [the C33 processor is not running OR the C33 processing is running but the Instruction Cache is disabled];

   64 system clocks if the C33 processor is running with the Instruction Cache enabled.

   For asynchronous register access:

   8 HSCK cycles ≥ 7 ClkSPI cycles + 91ns

7.5.2 I2C

The I2C host module requires a valid clock selection before the interface can operate. The I2C host module clock selection is determined by a combination of I2CCLKEN (AB5) pin and REG[0063h] bits 2 and 0.

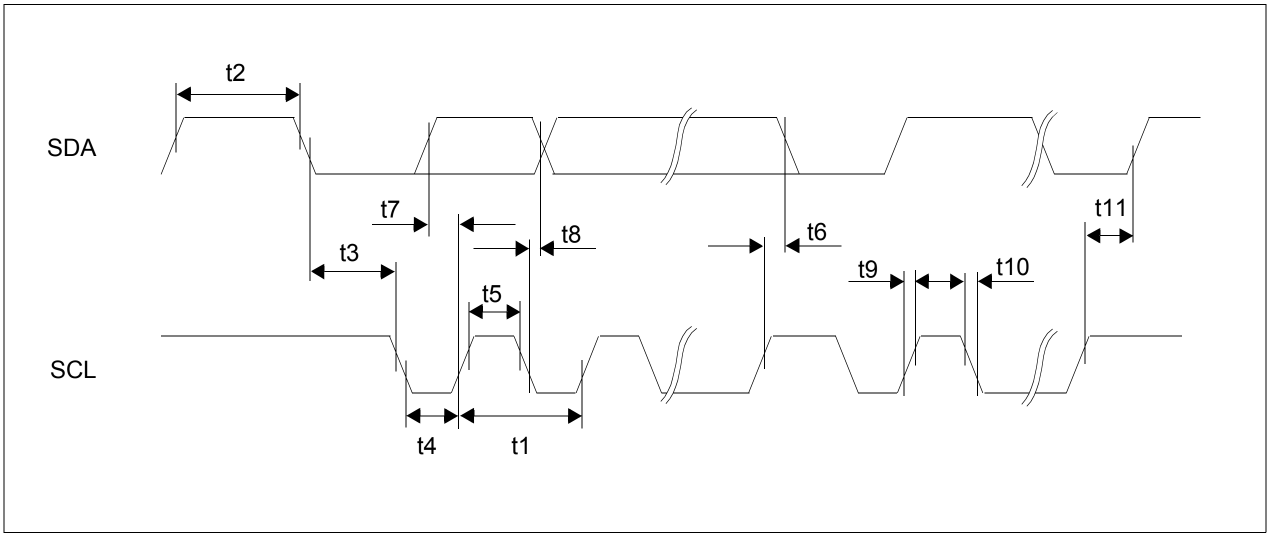


Figure 7-28: I2C Host Interface Timing

Table 7-29: I2C Host Interface Timing

Symbol	Parameter	HIOVDD = 2.5V		HIOVDD = 3.3V		Units
		Min	Max	Min	Max	
t1	SCL Frequency	—	400	—	400	KHz
t2	Bus Free time between a STOP and START condition	1.3	—	1.3	—	µs
t3	Hold time for a START Condition	0.6	—	0.6	—	µs
t4	SCL Low Width	1.3	—	1.3	—	µs
t5	SCL High Width	0.6	—	0.6	—	µs
t6	Setup time for a repeated START Condition	0.6	—	0.6	—	µs
t7	SDA setup time from SCL Rising	100	—	100	—	ns
t8	SDA hold time to SCL Falling	0	—	0	—	µs
t9	Rise Time of both SCL and SDA	—	300	—	300	ns
t10	Fall Time of both SCL and SDA	—	300	—	300	ns
t11	Setup time for a STOP Condition	0.6	—	0.6	—	µs

The user must select a ClkI2C (I2C control module clock) frequency and System Clock frequency that meet the following equation.

For synchronous register access:

8 SCL cycles ≥ X + 17 ClkI2C cycles + 5 System Clock cycles

where X is:

0 if the DMA Controller is not running AND C33 processor is not running;

16 system clocks if the DMA Controller is transferring data AND [the C33 processor is not running OR the C33 processing is running but the Instruction Cache is disabled];

64 system clocks if the C33 processor is running with the Instruction Cache enabled.

For asynchronous register access:

8 SCL cycles ≥ 17 ClkI2C cycles + 91ns

7.6 Panel Interface Timing

Note
For XGA 1024x768 panel support, only single panel, single window with no virtual width function is supported (i.e. Blend Mode 0 with MAIN window only (AUX and OSD windows disabled) and Main Virtual Width, REG[0954h] ~ REG[0955h] is same as the Main Width, REG[0950h] REG[0951h]).
Any additional accesses to DRAM could potentially result in internal bandwidth limitations and must be evaluated on a case-by-case situation to ensure bandwidth throughput availability. The following table contains recommended values for XGA panel support.

Table 7-30: Recommended Settings for XGA Support

DRAM CLK (MHz)	PCLK (MHz)	HT (REG[4020h] ~ REG[4021h])	VT (REG[402Ah] ~ REG[402Bh])	Frame Rate (Hz)
100	60	1280	774	60
100	50	1056	774	60
100	65	1402	774	60

7.6.1 Generic TFT Panel Timing

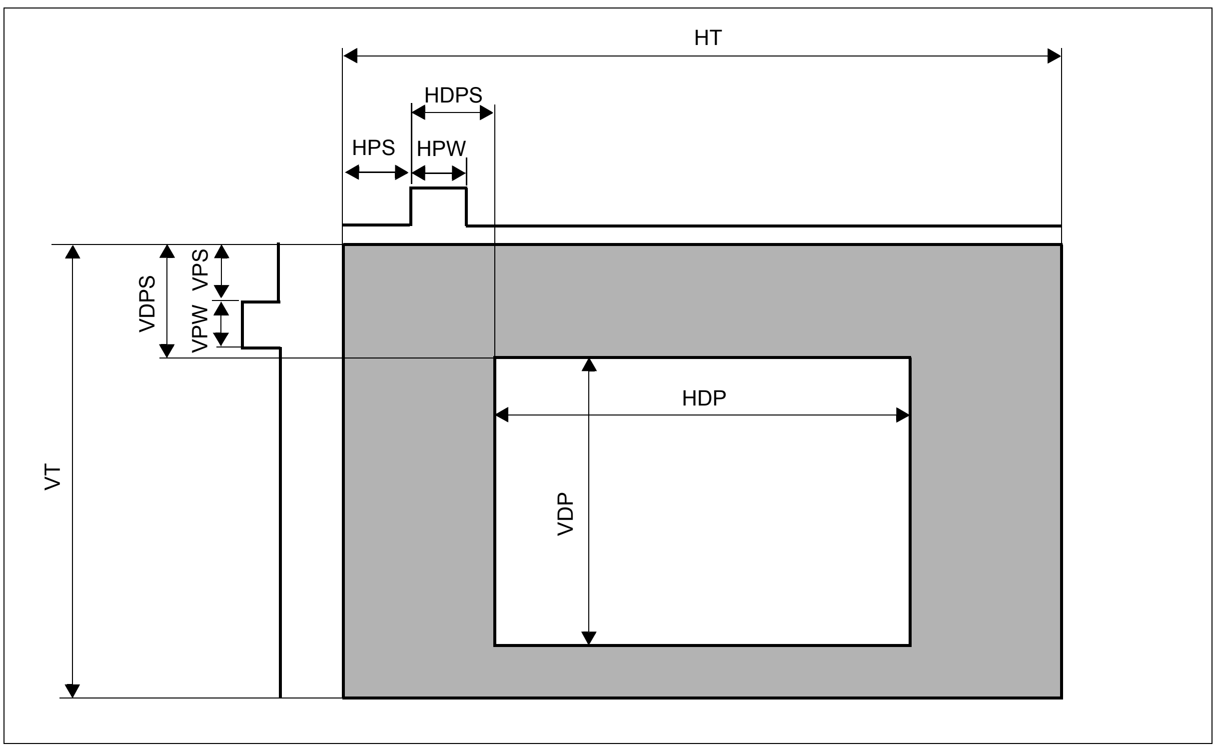


Figure 7-29: Generic TFT Panel Timing

Table 7-31: Generic TFT Panel Timing for LCD1

Symbol	Description	Derived From	Units
HT	Horizontal Total (HSYNC period)	(REG[4002h] bits 11-0) + 1	
HDP	Horizontal Display Period	((REG[4004h] bits 10-0) + 1) x 2	
HDPS	Horizontal Display Period Start Position	(REG[4006h] bits 11-0) + 1	Tp
HPW	Horizontal Pulse (HSYNC) Width	(REG[4008h] bits 8-0) + 1	
HPS	Horizontal Pulse (HSYNC) Start Position	REG[400Ah] bits 11-0	
VT	Vertical Total (VSYNC period)	(REG[400Ch] bits 11-0) + 1	
VDP	Vertical Display Period	(REG[400Eh] bits 11-0) + 1	
VDPS	Vertical Display Period Start Position	REG[4010h] bits 11-0	Lines
VPW	Vertical Pulse (VSYNC) Width	(REG[4012h] bits 4-0) + 1	
VPS	Vertical Pulse (VSYNC) Start Position	REG[4014h] bits 11-0	

1. Tp is the period of the pixel clock (1 / Fp) for LCD1. The frequency of the pixel clock (Fp) for LCD1 is determined by REG[003Ch] bit 2, REG[003Eh] bits 7-4, and REG[0030h].
2. The following formulas must be valid for all panel timings:
   HPS + HDPS + HDP < HT
   VDPS + VDP < VT

Table 7-32: Generic TFT Panel Timing for LCD2

Symbol	Description	Derived From	Units
HT	Horizontal Total (HSYNC period)	(REG[4020h] bits 11-0) + 1	
HDP	Horizontal Display Period	((REG[4022h] bits 10-0) + 1) x 2	
HDPS	Horizontal Display Period Start Position	(REG[4024h] bits 11-0) + 1	Tp
HPW	Horizontal Pulse (HSYNC) Width	(REG[4026h] bits 8-0) + 1	
HPS	Horizontal Pulse (HSYNC) Start Position	REG[4028h] bits 11-0	
VT	Vertical Total (VSYNC period)	(REG[402Ah] bits 11-0) + 1	
VDP	Vertical Display Period	(REG[402Ch] bits 11-0) + 1	
VDPS	Vertical Display Period Start Position	REG[402Eh] bits 11-0	Lines
VPW	Vertical Pulse (VSYNC) Width	(REG[4030h] bits 4-0) + 1	
VPS	Vertical Pulse (VSYNC) Start Position	REG[4032h] bits 11-0	

1. Tp is the period of the pixel clock (1 / Fp) for LCD2. The frequency of the pixel clock (Fp) for LCD2 is determined by REG[003Ch] bit 2, REG[003Eh] bits 7-4, and REG[0031h].
2. The following formulas must be valid for all panel timings:
   HPS + HDPS + HDP < HT
   VDPS + VDP < VT

Generic RGB Type Interface Panel Horizontal Timing

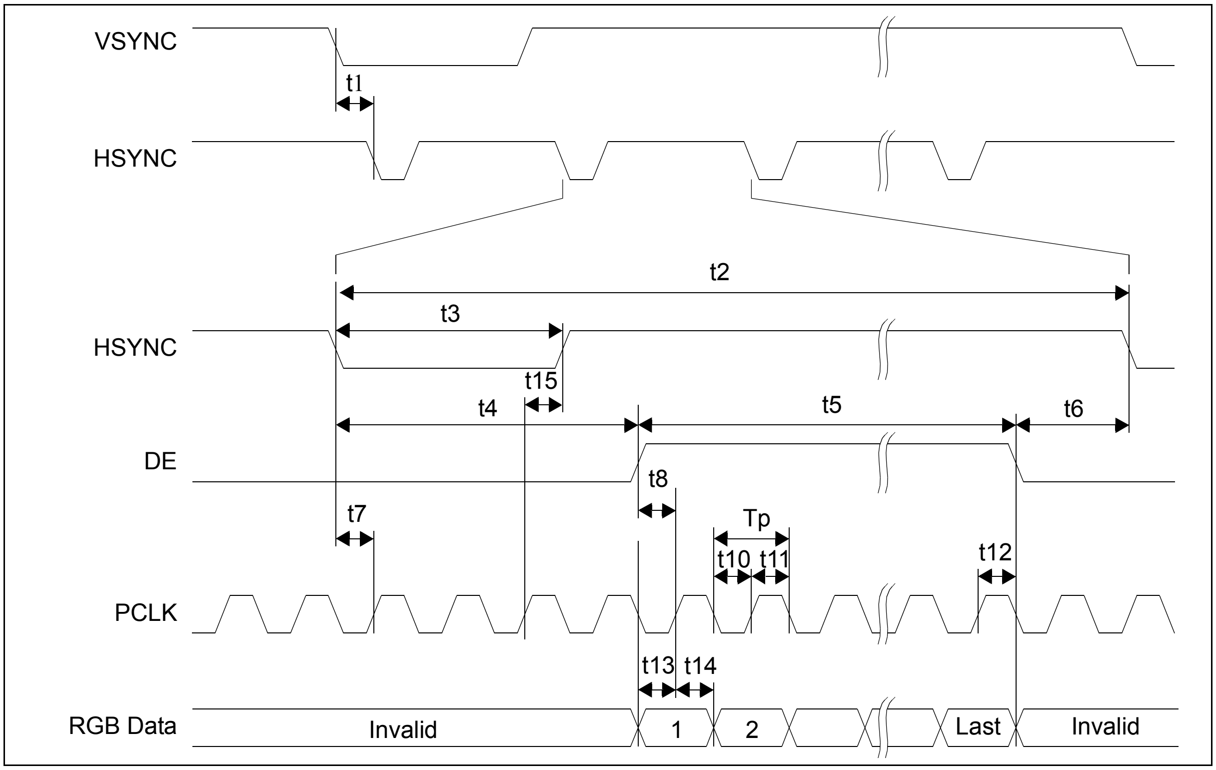


Figure 7-30: Generic RGB Type Interface Panel Horizontal Timing

Table 7-33: Generic RGB Type Interface Panel Horizontal Timing for LCD1 (FP1IO)*

Symbol	Parameter	Min	Typ	Max	Units
t1	VSYNC falling edge to HSYNC falling edge	—	HPS	—	Tp (Note 1)
t2	Horizontal total period	—	HT	—	Tp
t3	HSYNC pulse width	—	HPW	—	Tp
t4	HSYNC falling edge to DRDY active	—	HDPS	—	Tp
t5	Horizontal display period	—	HDP	—	Tp
t6	DE falling edge to HSYNC falling edge	—	Note 2	—	Tp
t7	HSYNC setup time to PCLK falling edge	0.5Tp	0.5	—	Tp
t8	DE setup to PCLK falling edge	0.5Tp	0.5	—	Tp
Tp	PCLK period	15.625	—	—	ns
t10	PCLK pulse width high	0.5Tp - 1.5ns	—	0.5Tp	Tp
t11	PCLK pulse width low	0.5Tp	—	0.5Tp+1.5ns	Tp
t12	DE hold from PCLK falling edge	0.5Tp - 5ns	0.5	—	Tp
t13	Data setup to PCLK falling edge	0.5Tp - 2ns	0.5	—	Tp
t14	Data hold from PCLK falling edge	0.5Tp - 5ns	0.5	—	Tp
t15	HSYNC hold time from PCLK falling edge	0.5Tp - 3ns	0.5	—	Tp

Table 7-34: Generic RGB Type Interface Panel Horizontal Timing for LCD2 (FP2IO)*

Symbol	Parameter	Min	Typ	Max	Units
t1	VSYNC falling edge to HSYNC falling edge	—	HPS	—	Tp (Note 1)
t2	Horizontal total period	—	HT	—	Tp
t3	HSYNC pulse width	—	HPW	—	Tp
t4	HSYNC falling edge to DRDY active	—	HDPS	—	Tp
t5	Horizontal display period	—	HDP	—	Tp
t6	DE falling edge to HSYNC falling edge	—	Note 2	—	Tp
t7	HSYNC setup time to PCLK falling edge	0.5Tp - 2ns	0.5	—	Tp
t8	DE setup to PCLK falling edge	0.5Tp - 2ns	0.5	—	Tp
Tp	PCLK period	13.89	—	—	ns
t10	PCLK pulse width high	0.5Tp - 0.5ns	—	0.5Tp	Tp
t11	PCLK pulse width low	0.5Tp	—	0.5Tp+0.5ns	Tp
t12	DE hold from PCLK falling edge	0.5Tp - 2ns	0.5	—	Tp
t13	Data setup to PCLK falling edge	0.5Tp - 2ns	0.5	—	Tp
t14	Data hold from PCLK falling edge	0.5Tp - 2ns	0.5	—	Tp
t15	HSYNC hold time from PCLK falling edge	0.5Tp - 1ns	0.5	—	Tp

1. Tp = pixel clock period
2. t6typ = t2 - t4 - t5
3. The Generic TFT timing diagrams assume the following polarity of signals:
   VSYNC Pulse Polarity bit is active low.
   HSYNC Pulse Polarity bit is active low.
   PCLK Polarity is programmed so that all panel interface signals change at the falling edge of PCLK.

Generic RGB Type Interface Panel Vertical Timing

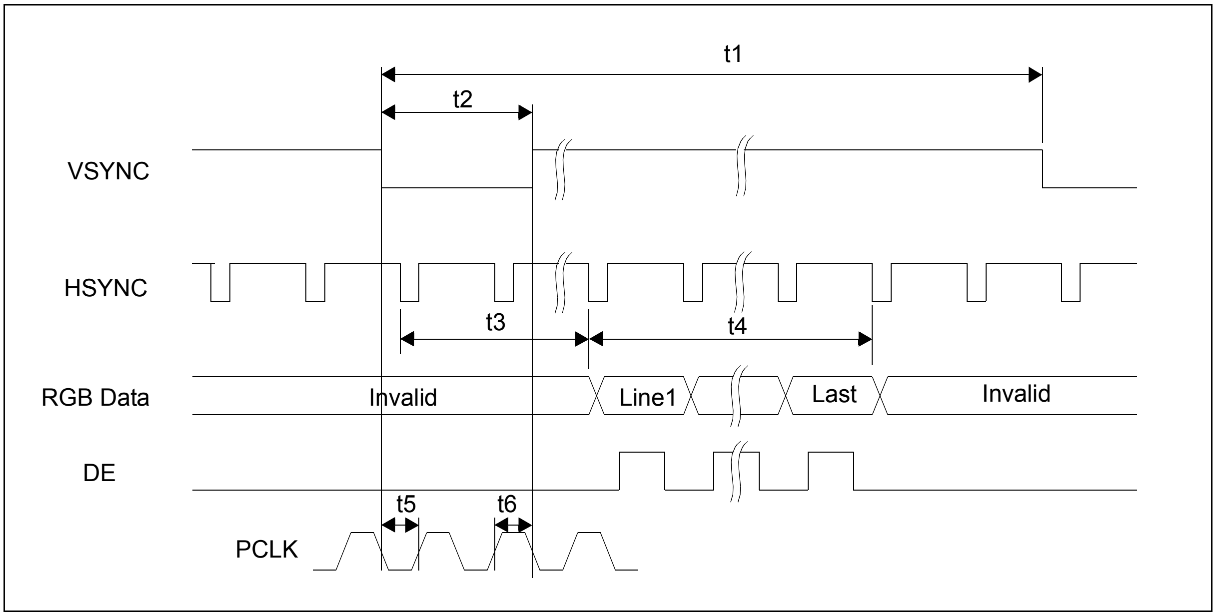


Figure 7-31: Generic RGB Type Interface Panel Vertical timing

Table 7-35: Generic RGB Type Interface Panel Vertical Timing for LCD1 (FP1IO)*

Symbol	Parameter	Min	Typ	Max	Units
t1	Vertical total period	—	VT	—	Lines
t2	VSYNC pulse width	—	VPW	—	Lines
t3	Vertical display start position (Note 1)	—	Note 2	—	Lines
t4	Vertical display period	—	VDP	—	Lines
t5	VSYNC setup to PCLK falling edge	0.5Tp - 1ns	0.5	—	Tp
t6	VSYNC hold from PCLK falling edge	0.5Tp - 3ns	0.5	—	Tp

Table 7-36: Generic RGB Type Interface Panel Vertical Timing for LCD2 (FP2IO)*

Symbol	Parameter	Min	Typ	Max	Units
t1	Vertical total period	—	VT	—	Lines
t2	VSYNC pulse width	—	VPW	—	Lines
t3	Vertical display start position (Note 1)	—	Note 2	—	Lines
t4	Vertical display period	—	VDP	—	Lines
t5	VSYNC setup to PCLK falling edge	0.5Tp - 2ns	0.5	—	Tp
t6	VSYNC hold from PCLK falling edge	0.5Tp - 1ns	0.5	—	Tp

1. t3 is measured from the first HSYNC pulse after the start of the frame to the first HSYNC pulse when RGB Data is valid.
2. t3typ = VDPS - VPS

7.6.2 ND-TFD 8-Bit Serial Interface Timing

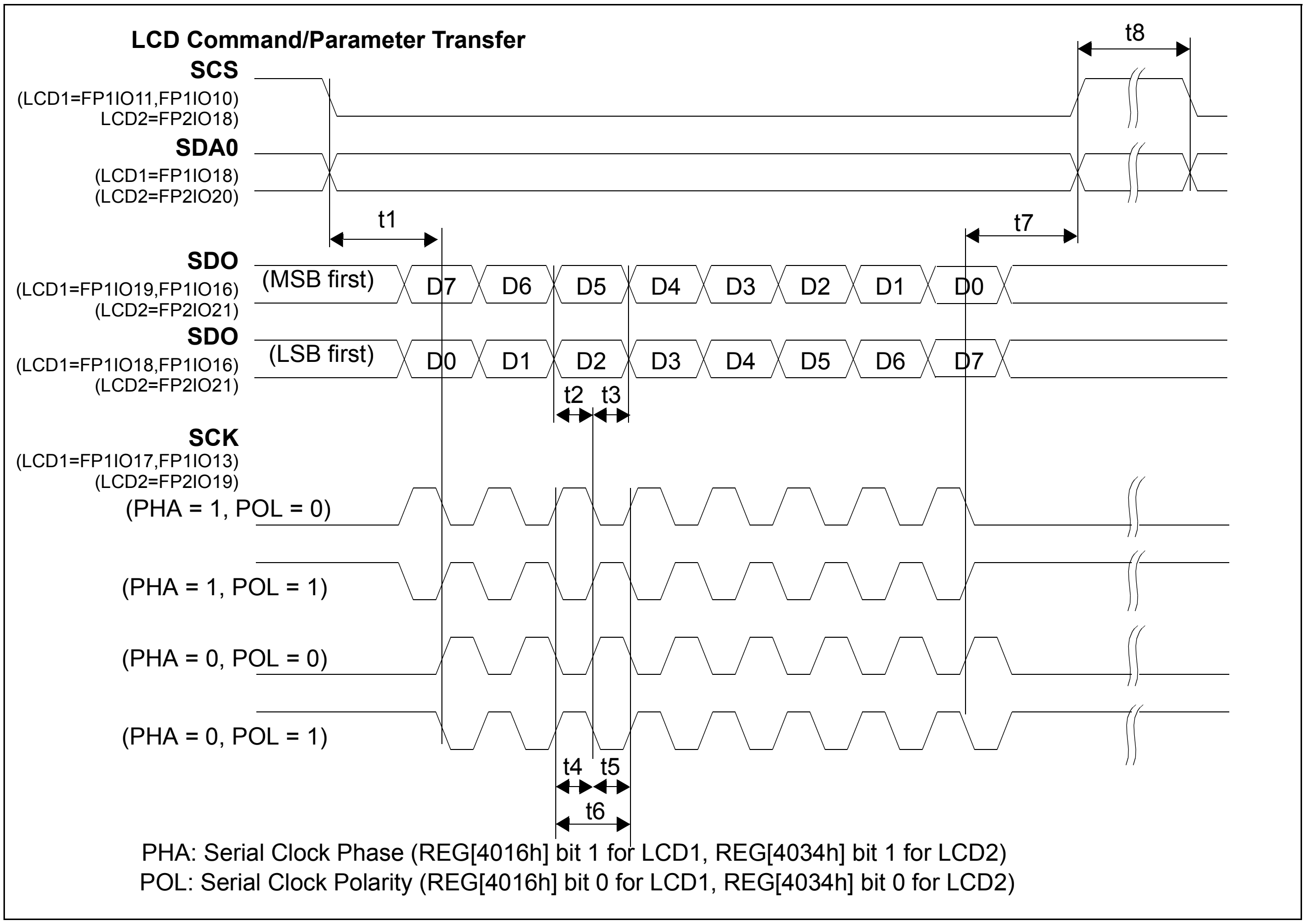


Figure 7-32: ND-TFD 8-Bit Serial Interface Timing

Table 7-37: ND-TFD 8-Bit Serial Interface Timing for LCD1(FP1IO)*

Symbol	Parameter	Min	Typ	Max	Units
t1	SCS/SDA0 setup time	1.5Ts - 3ns	1.5	—	Ts (Note 1)
t2	Data setup time	0.5Ts - 3ns	0.5	—	Ts
t3	Data hold time	0.5Ts - 2ns	0.5	—	Ts
t4	Serial clock pulse width low (high)	0.5Ts - 3ns	0.5	0.5Ts + 3ns	Ts
t5	Serial clock pulse width high (low)	0.5Ts - 3ns	0.5	0.5Ts + 3ns	Ts
t6	Serial clock period	—	1	—	Ts
t7	SCS/SDA0 hold time	1.5Ts -2ns	1.5	—	Ts
t8	Chip select de-assert to reassert	—	Note 2	—	Ts

Table 7-38: ND-TFD 8-Bit Serial Interface Timing for LCD2(FP2IO)*

Symbol	Parameter	Min	Typ	Max	Units
t1	SCS/SDA0 setup time	1.5Ts - 2ns	1.5	—	Ts (Note 1)
t2	Data setup time	0.5Ts - 1ns	0.5	—	Ts
t3	Data hold time	0.5Ts	0.5	—	Ts
t4	Serial clock pulse width low (high)	0.5Ts - 1ns	0.5	0.5Ts + 1ns	Ts
t5	Serial clock pulse width high (low)	0.5Ts - 1ns	0.5	0.5Ts + 1ns	Ts
t6	Serial clock period	—	1	—	Ts
t7	SCS/SDA0 hold time	1.5Ts	1.5	—	Ts
t8	Chip select de-assert to reassert	—	Note 2	—	Ts

1. Ts = Serial clock period
2. This result is software dependent, based on host register access latency.

7.6.3 ND-TFD 9-Bit Serial Interface Timing

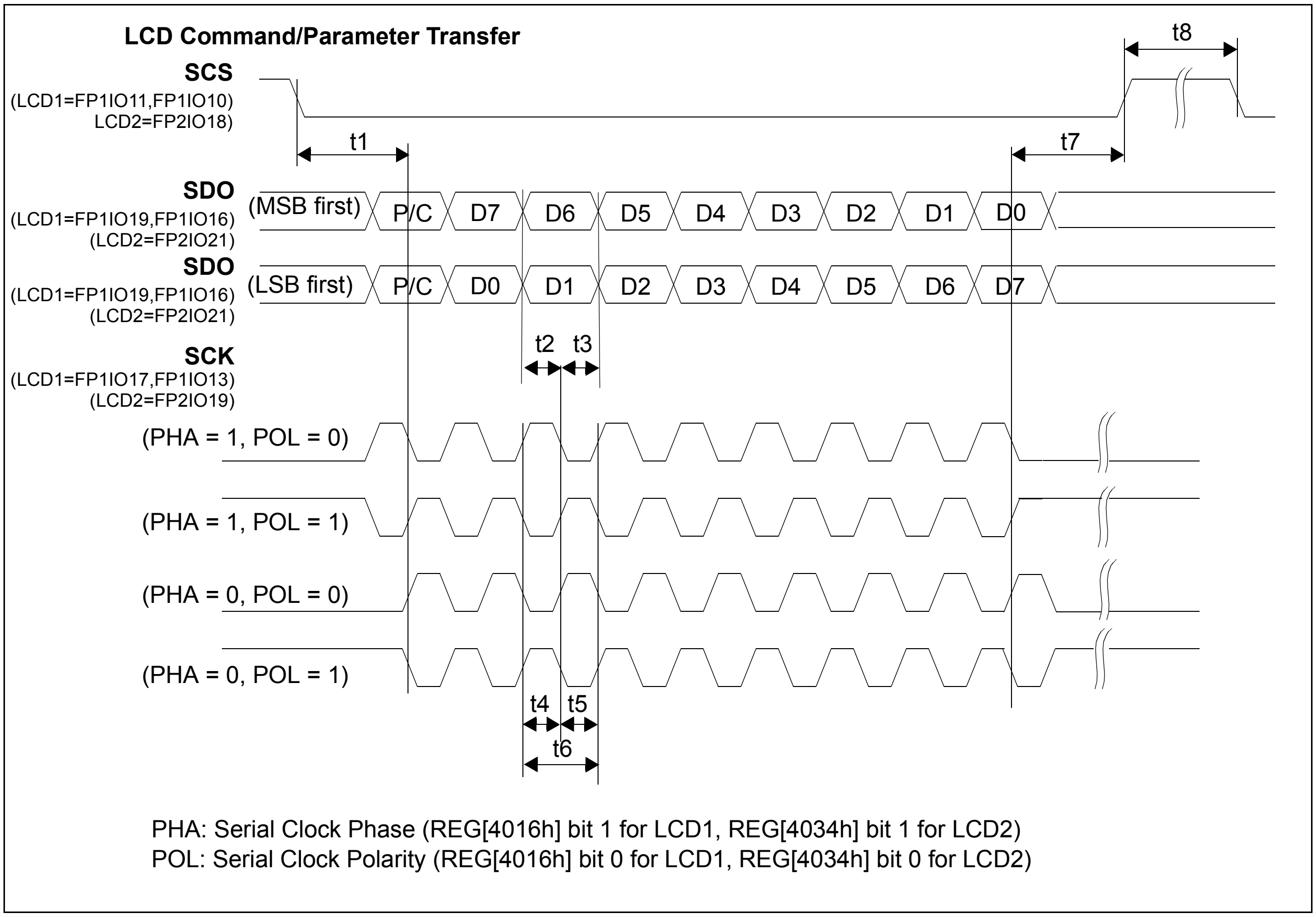


Figure 7-33: ND-TFD 9-Bit Serial Interface Timing

Table 7-39: ND-TFD 9-Bit Serial Interface Timing for LCD1 (FP1IO)*

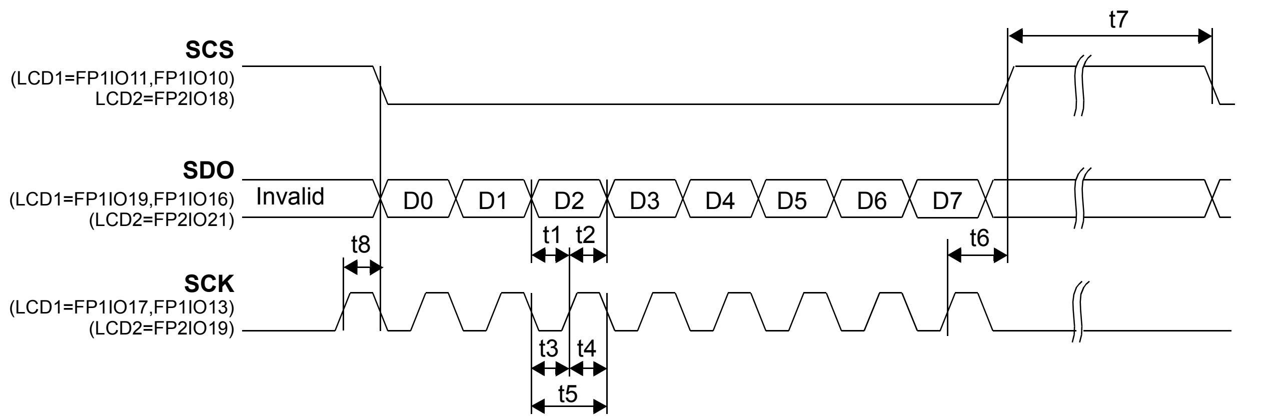
Symbol	Parameter	Min	Typ	Max	Units
t1	Chip select setup time	1.5Ts - 3ns	1.5	—	Ts (Note 1)
t2	Data setup time	0.5Ts - 3ns	0.5	—	Ts
t3	Data hold time	0.5Ts - 2ns	0.5	—	Ts
t4	Serial clock pulse width low (high)	0.5Ts - 3ns	0.5	0.5Ts + 3ns	Ts
t5	Serial clock pulse width high (low)	0.5Ts - 3ns	0.5	0.5Ts + 3ns	Ts
t6	Serial clock period	—	1	—	Ts
t7	Chip select hold time	1.5Ts - 2ns	1.5	—	Ts
t8	Chip select de-assert to reassert	—	Note 2	—	Ts

Table 7-40: ND-TFD 9-Bit Serial Interface Timing for LCD1 (FP2IO)*

Symbol	Parameter	Min	Typ	Max	Units
t1	Chip select setup time	1.5Ts - 2ns	1.5	—	Ts (Note 1)
t2	Data setup time	0.5Ts - 1ns	0.5	—	Ts
t3	Data hold time	0.5Ts	0.5	—	Ts
t4	Serial clock pulse width low (high)	0.5Ts - 1ns	0.5	0.5Ts + 1ns	Ts
t5	Serial clock pulse width high (low)	0.5Ts - 1ns	0.5	0.5Ts + 1ns	Ts
t6	Serial clock period	—	1	—	Ts
t7	Chip select hold time	1.5Ts	1.5	—	Ts
t8	Chip select de-assert to reassert	—	Note 2	—	Ts

1. Ts = Serial clock period
2. This result is software dependent, based on host register access latency.

7.6.4 a-Si TFT Serial Interface Timing



Note: The a-Si interface is affected by REG[4016h] / REG[4034h] bits 4 and 1-0.
For these AC timings, REG[4016h] / REG[4034h] bit 4 = 1b and REG[4016h] / REG[4034h] bits 1-0 = 00b.

Figure 7-34: a-Si TFT Serial Interface Timing

Table 7-41: a-Si TFT Serial Interface Timing for LCD1 (FP1IO)*

Symbol	Parameter	Min	Typ	Max	Units
t1	Data Setup Time	0.5Ts - 3ns	0.5	—	Ts (Note 1)
t2	Data Hold Time	0.5Ts - 2ns	0.5	—	Ts
t3	Serial clock plus low period	0.5Ts - 3ns	0.5	0.5Ts + 3ns	Ts
t4	Serial clock pulse high period	0.5Ts - 3ns	0.5	0.5Ts + 3ns	Ts
t5	Serial clock period	—	1	—	Ts
t6	Chip select hold time	1.5Ts - 2ns	1.5	—	Ts
t7	Chip select de-assert to reassert	—	Note 2	—	Ts
t8	SCK rising edge to SCS (strobe) falling edge	—	0.5	0.5Ts + 3ns	Ts

Table 7-42: a-Si TFT Serial Interface Timing for LCD1 (FP2IO)*

Symbol	Parameter	Min	Typ	Max	Units
t1	Data Setup Time	0.5Ts - 1ns	0.5	—	Ts (Note 1)
t2	Data Hold Time	0.5Ts	0.5	—	Ts
t3	Serial clock plus low period	0.5Ts - 1ns	0.5	0.5Ts + 1ns	Ts
t4	Serial clock pulse high period	0.5Ts - 1ns	0.5	0.5Ts + 1ns	Ts
t5	Serial clock period	—	1	—	Ts
t6	Chip select hold time	1.5Ts	1.5	—	Ts
t7	Chip select de-assert to reassert	—	Note 2	—	Ts
t8	SCK rising edge to SCS (strobe) falling edge	—	0.5	0.5Ts + 2ns	Ts

1. Ts = Serial clock period
2. This setting depends on software.

7.6.5 uWIRE Serial Interface Timing

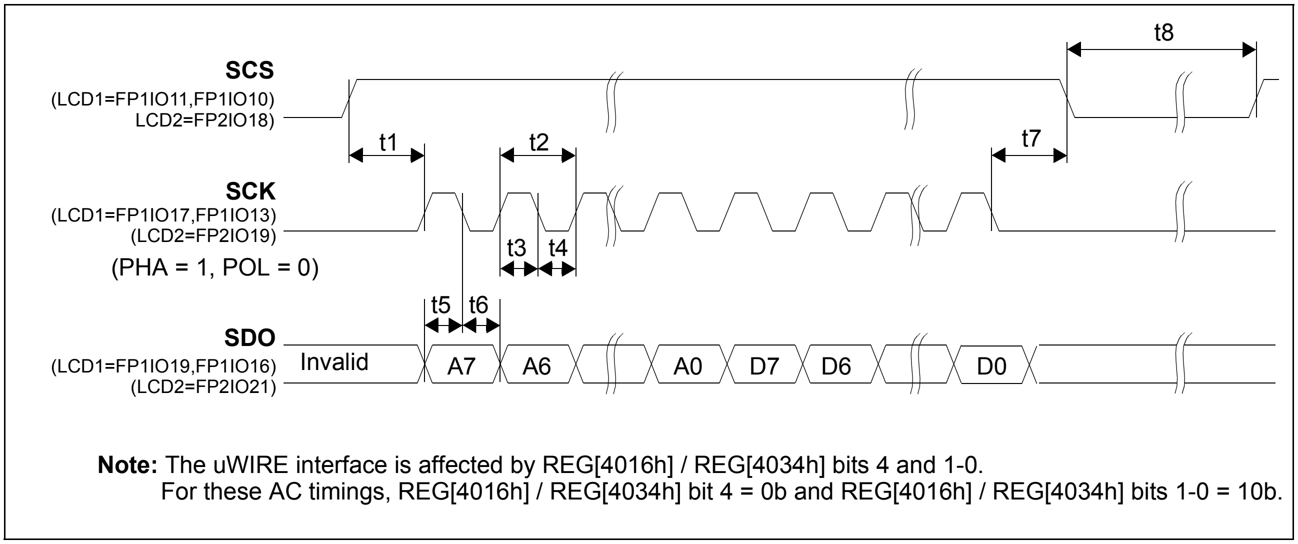


Figure 7-35: uWIRE Serial Interface Timing

Table 7-43: uWIRE Serial Interface Timing for LCD1 (FP1IO)*

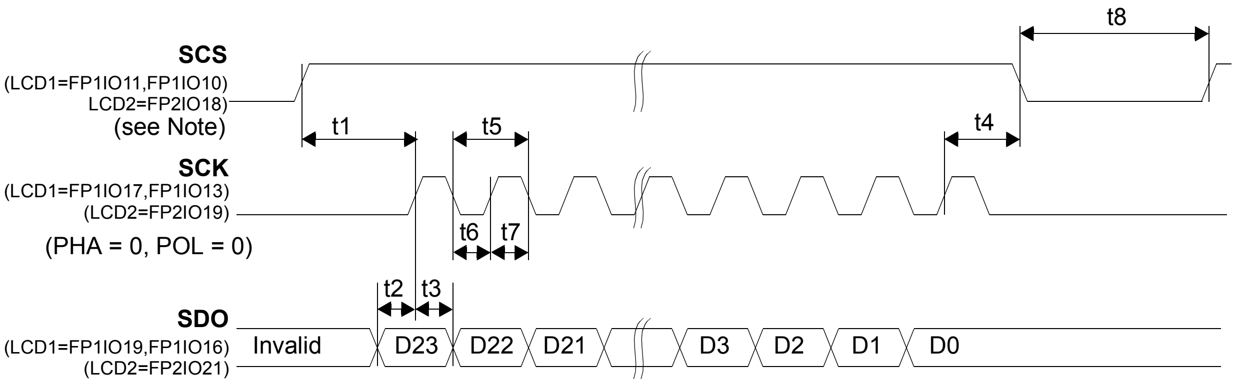
Symbol	Parameter	Min	Typ	Max	Units
t1	Chip select setup time	1.5Ts - 3ns	1.5	—	Ts (Note 1)
t2	Serial clock Period	—	1	—	Ts
t3	Serial clock pulse width low	0.5Ts - 3ns	0.5	0.5Ts + 3ns	Ts
t4	Serial clock pulse width high	0.5Ts - 3ns	0.5	0.5Ts + 3ns	Ts
t5	Data setup time	0.5Ts - 3ns	0.5	—	Ts
t6	Data hold time	0.5Ts -2ns	0.5	—	Ts
t7	Chip select hold time	1.5Ts - 2ns	1.5	—	Ts
t8	Chip select de-assert to reassert	—	Note 2	—	Ts

Table 7-44: uWIRE Serial Interface Timing for LCD2 (FP2IO)*

Symbol	Parameter	Min	Typ	Max	Units
t1	Chip select setup time	1.5Ts - 2ns	1.5	—	Ts (Note 1)
t2	Serial clock Period	—	1	—	Ts
t3	Serial clock pulse width low	0.5Ts - 1ns	0.5	0.5Ts + 1ns	Ts
t4	Serial clock pulse width high	0.5Ts - 1ns	0.5	0.5Ts + 1ns	Ts
t5	Data setup time	0.5Ts - 1ns	0.5	—	Ts
t6	Data hold time	0.5Ts	0.5	—	Ts
t7	Chip select hold time	1.5Ts	1.5	—	Ts
t8	Chip select de-assert to reassert	—	Note 2	—	Ts

1. Ts = Serial clock period
2. This setting depends on software

7.6.6 24-Bit Serial Interface Timing



Note: The 24-bit serial interface is affected by REG[4016h] / REG[4034h] bits 4 and 1-0.
For these AC timings, REG[4016h] / REG[4034h] bit 4 = 0b and REG[4016h] / REG[4034h] bits 1-0 = 00b.

Figure 7-36: 24-Bit Serial Interface Timing

Table 7-45: 24-bit Serial Interface Timing for LCD1 (FP1IO)*

Symbol	Parameter	Min	Typ	Max	Units
t1	Chip select setup time	1.5Ts - 3ns	1.5	—	Ts (Note 1)
t2	Data setup time	0.5Ts - 3ns	0.5	—	Ts
t3	Data hold time	0.5Ts - 2ns	0.5	—	Ts
t4	Chip select hold time	1.5Ts - 2ns	1.5	—	Ts
t5	Serial clock period	—	1	—	Ts
t6	Serial clock pulse low	0.5Ts - 3ns	0.5	0.5Ts + 3ns	Ts
t7	Serial clock pulse high	0.5Ts - 3ns	0.5	0.5Ts + 3ns	Ts
t8	Chip select de-assert to re-assert	—	Note 2	—	Ts

Table 7-46: 24-Bit Serial Interface Timing for LCD2 (FP2IO)*

Symbol	Parameter	Min	Typ	Max	Units
t1	Chip select setup time	1.5Ts - 2ns	1.5	—	Ts (Note 1)
t2	Data setup time	0.5Ts - 1ns	0.5	—	Ts
t3	Data hold time	0.5Ts	0.5	—	Ts
t4	Chip select hold time	1.5Ts	1.5	—	Ts
t5	Serial clock period	—	1	—	Ts
t6	Serial clock pulse low	0.5Ts - 1ns	0.5	0.5Ts + 1ns	Ts
t7	Serial clock pulse high	0.5Ts - 1ns	0.5	0.5Ts + 1ns	Ts
t8	Chip select de-assert to re-assert	—	Note 2	—	Ts

1. Ts = Serial clock period
2. This setting depends on software

7.6.7 Sharp DualView Panel Timing

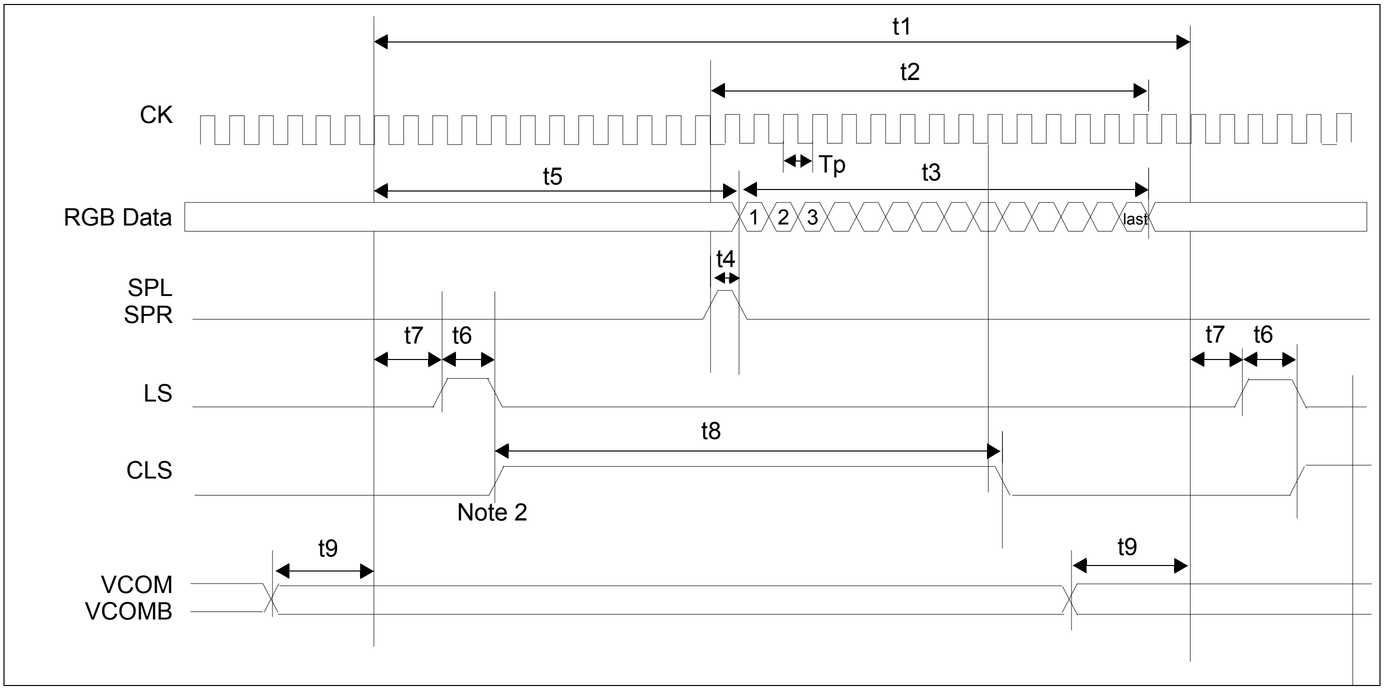


Figure 7-37: Sharp DualView Panel Horizontal Timing

Table 7-47: Sharp DualView Panel Programmable Horizontal Timing

Symbol	Description	Nominal	Units
t1	Horizontal Total (LS period)	(REG[4020h] bits 11-0) + 1	Tp
t2	CK Active Period	[((REG[4022h] bits 10-0) + 1) x 2] + 1	
t3	Horizontal Display Period	((REG[4022h] bits 10-0) + 1) x 2	
t4	SPL/SPR Pulse Width	1	
t5	Horizontal Display Period Start Position	(REG[4024h] bits 11-0) +1	
t6	Horizontal Pulse (LS) Width	(REG[4026h] bits 8-0) + 1	
t7	Horizontal Pulse (LS) Start Position	REG[4056h] bits 7-0	
t8	CLS Pulse Width	(REG[4052h] bits 10-0) > 0	
t9	VCOM/VCOMB Toggle Position	REG[4054h] bits 6-0	

1. Tp = pixel clock period
2. CLS rising edge occurs at the same time as the LS falling edge.
3. The Sharp DualView horizontal timings are based on the following:
   LS (HSYNC) Pulse Polarity bit is active high.
   CK Pulse Polarity is 0b (REG[4001h] bit 7 = 0b) so all panel interface signals change at the falling edge of CK.

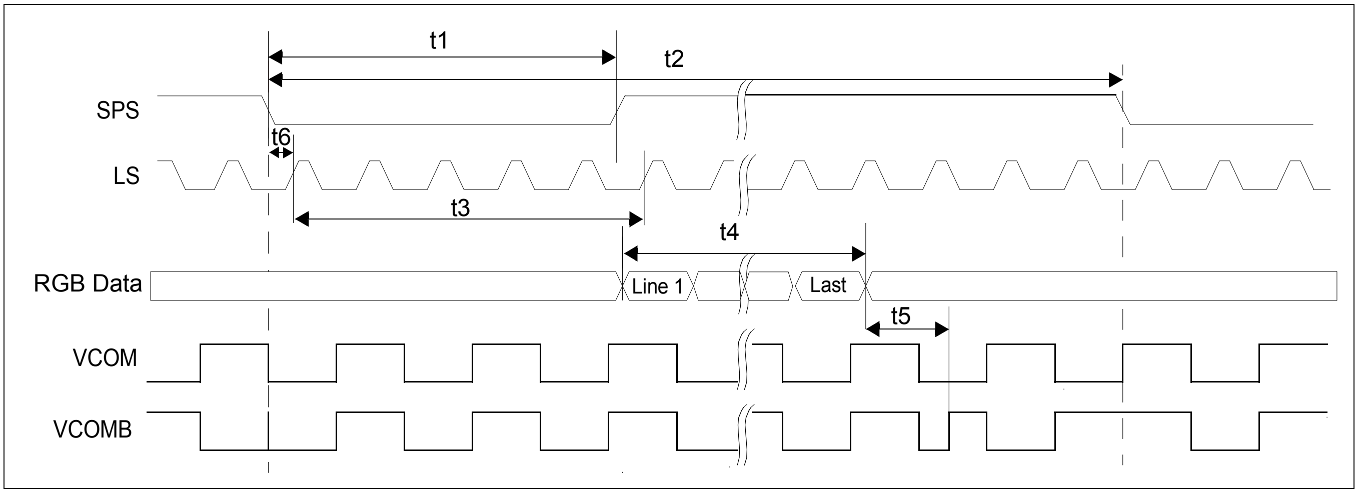


Figure 7-38: Sharp DualView Panel Vertical Timing

Table 7-48: Sharp DualView Panel Programmable Vertical Timing

Symbol	Description	Nominal	Units
t1	Vertical Pulse (SPS) Width (see Note 7)	(REG[4030h] bits 4-0) + 1	Lines
t2	Vertical Total (SPS period)	(REG[402Ah] bits 11-0) + 1	
t3	Vertical Display Period Start Position (see Note 3)	Note 4	
t4	Vertical Display Period	(REG[402Ch] bits 11-0) + 1	
t5	Last pixel data to VCOM/VCOMB inversion	(REG[4020h] bits 11-0) - (((REG[4022h] bits 10-0) + 1) x 2) - (REG[4024h] bits 11-0)	Tp
t6	SPS falling edge to LS rising edge	(REG[4028h] bits 11-0) + (REG[4056h] bits 7-0)	

1. Tp = pixel clock period
2. The Sharp DualView vertical timings are based on the following:
   SPS (VSYNC) Pulse Polarity bit is active low.
3. t3 is measured from the first LS pulse after the start of the frame to the first LS pulse when RGB Data is valid.
4. t3 = (REG[402Eh] bits 11-0) - (REG[4032h] bits 11-0)
5. VCOM toggles every line (including non-display period). The Vertical Total Period (REG[402Ah] + 1) should be programmed to be an odd number of lines so that the logic of VCOM at the beginning of the next frame is opposite of the logic of VCOM at the beginning of the current frame.
6. VCOM and VCOMB are in phase at the start of frame (SPS going low) until the end of display period, and they are out of phase (180 degrees) during the non-display period.
7. t1 $\geq$ t3 in order for VCOMB to be in phase with VCOM between SPS going low to the start of display period.

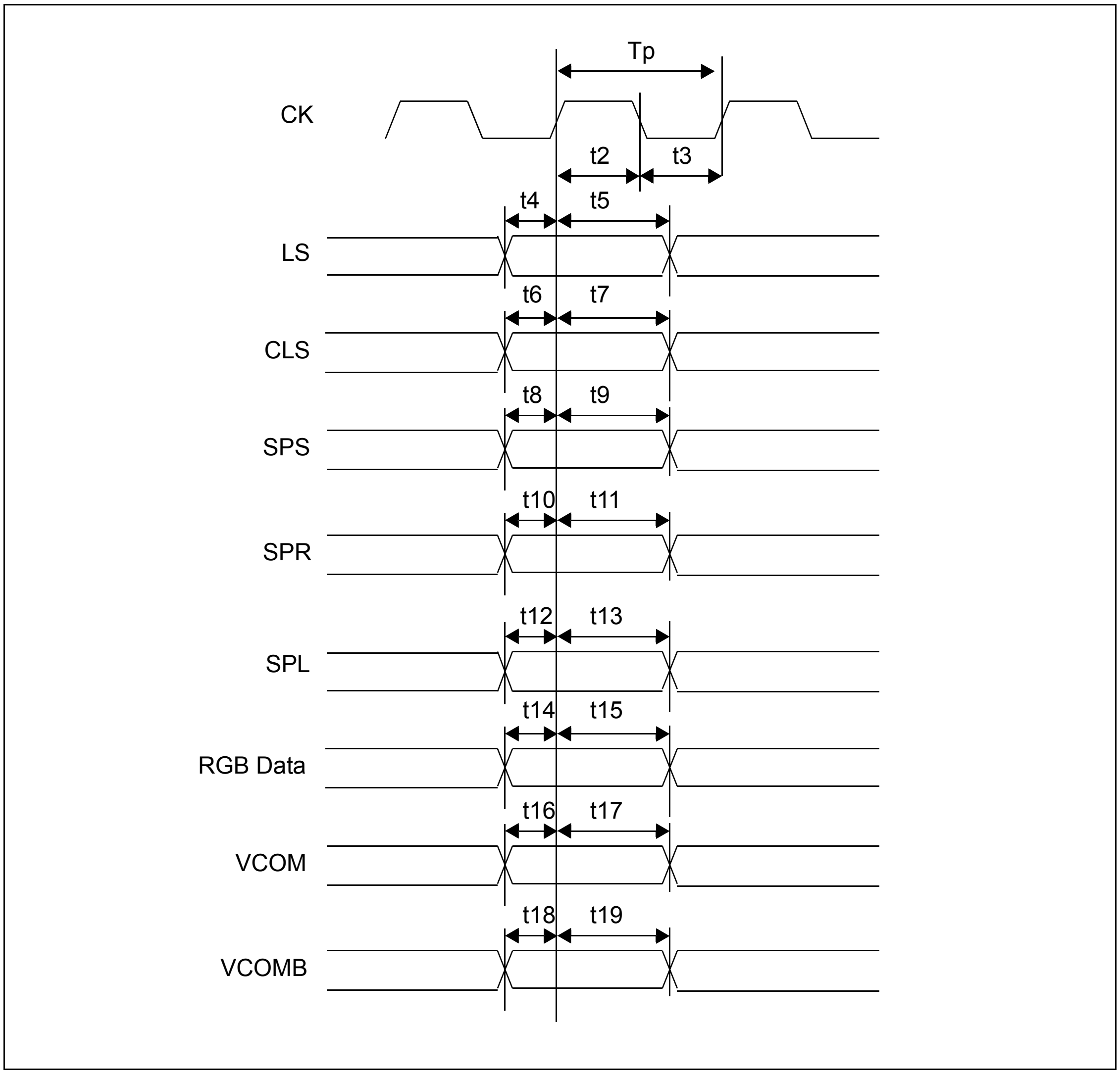


Figure 7-39: Sharp DualView Panel Timing

Table 7-49: Sharp DualView Panel Timing

Symbol	Parameter	Min	Typ	Max	Units
Tp	Pixel clock period	27.78	—	0.5Tp	ns
t2	Pixel clock pulse low	0.5Tp	—	0.5Tp+1.5ns	Tp
t3	Pixel clock pulse high	0.5Tp	—	0.5Tp+1.5ns	Tp
t4	LS setup before CK rising edge	0.5Tp - 2ns	0.5	—	Tp
t5	LS hold after CK rising edge	0.5Tp - 1ns	0.5	—	Tp
t6	CLS setup before CK rising edge	0.5Tp - 1ns	0.5	—	Tp
t7	CLS hold after CK rising edge	0.5Tp - 2ns	0.5	—	Tp
t8	SPS setup before CK rising edge	0.5Tp - 2ns	0.5	—	Tp
t9	SPS hold after CK rising edge	0.5Tp - 1ns	0.5	—	Tp
t10	SPR setup before CK rising edge	0.5Tp - 1ns	0.5	—	Tp
t11	SPR hold after CK rising edge	0.5Tp - 2ns	0.5	—	Tp
t12	SPL setup before CK rising edge	0.5Tp - 1ns	0.5	—	Tp
t13	SPL hold after CK rising edge	0.5Tp - 2ns	0.5	—	Tp
t14	Pixel Data setup before CK rising edge	0.5Tp - 2ns	0.5	—	Tp
t15	Pixel Data hold after CK rising edge	0.5Tp - 2ns	0.5	—	Tp
t16	VCOM setup before CK rising edge	0.5Tp - 2ns	0.5	—	Tp
t17	VCOM hold after CK rising edge	0.5Tp - 1ns	0.5	—	Tp
t18	VCOMB setup before CK rising edge	0.5Tp - 4ns	0.5	—	Tp
t19	VCOMB hold after CK rising edge	0.5Tp - 1ns	0.5	—	Tp

1. Tp = pixel clock period

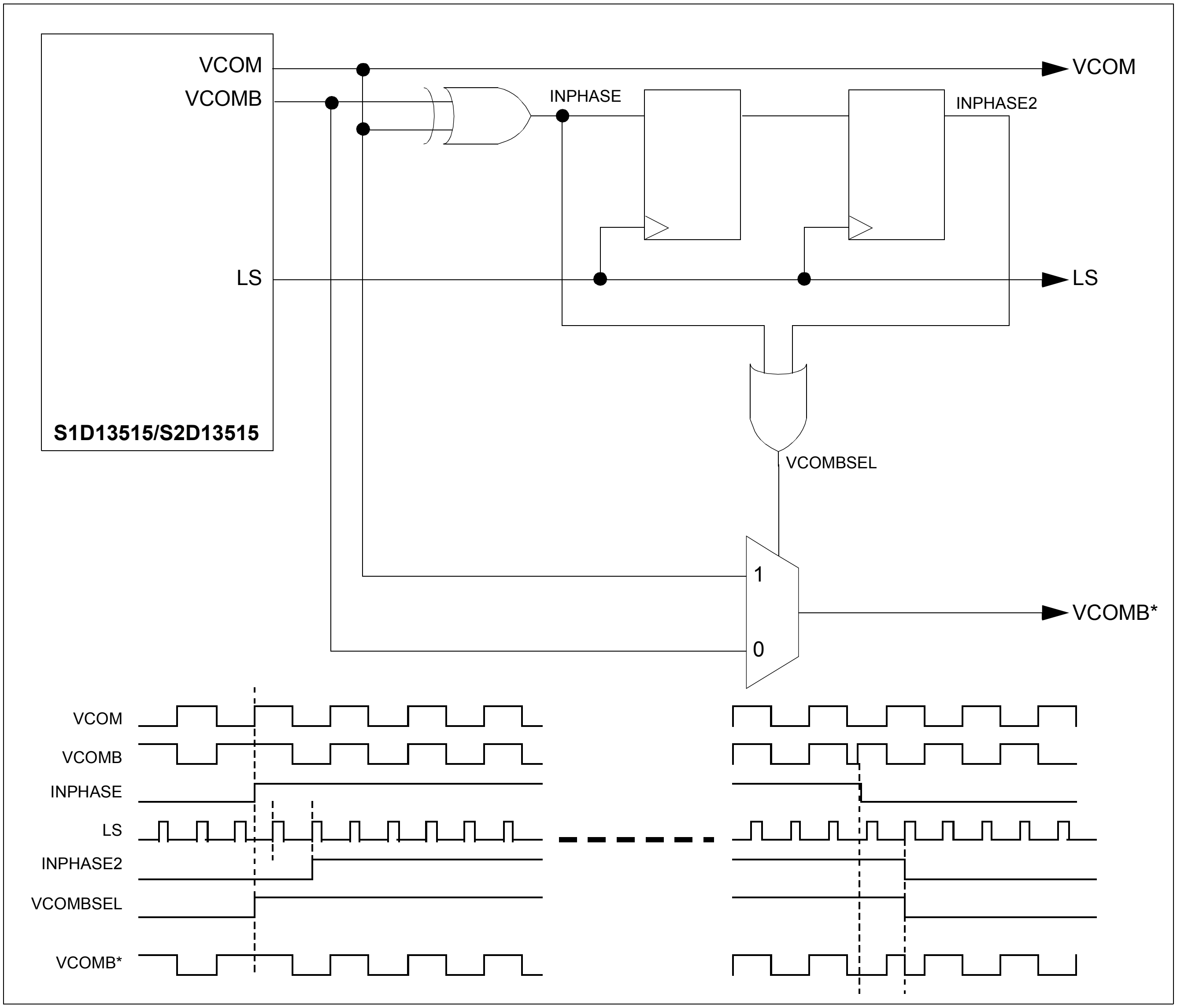


Figure 7-40: Required External VCOMB Logic

7.6.8 EID Double Screen Panel Timing (TCON Enabled)

Note
When using the EID Double Screen Panel with TCON enabled, the LCD2 Pixel Clock divide must be 1:1.

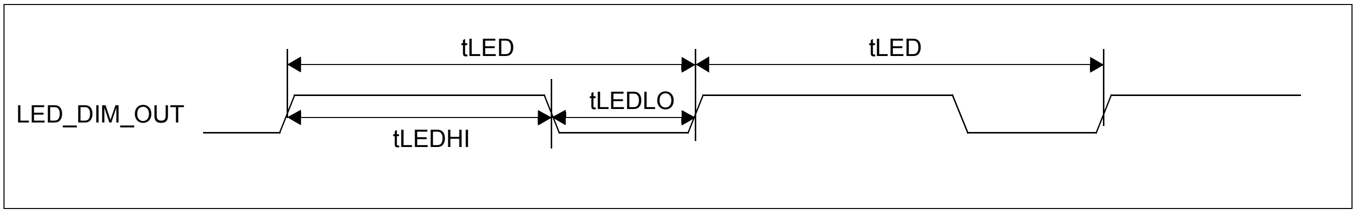


Figure 7-41: EID Double Screen Panel LED_DIM_OUT Timing

Table 7-50: EID Double Screen Panel LED_DIM_OUT Timing

Symbol	Description	Nominal	Units
tLED	LED clock period	400 x 16 x (100-(REG[404Fh] bits 7-0))	Tp
tLEDHI	LED HIGH time	[(REG[404Eh] bits 7-0) x 2] x 16 x (100-(REG[404Fh] bits 7-0))	Tp
tLEDLO	LED LOW time	tLED - tLEDHI	Tp

1. Tp = pixel clock period
2. REG[404Fh] bits 7-0 = 98 max. If REG[404Fh] bits 7-0 > 98, it will be clipped internally to 98.
3. REG[404Eh] bits 7-0 should be ≤ 200. If REG[404Eh] bits 7-0 > 200, it will be clipped internally to 200.

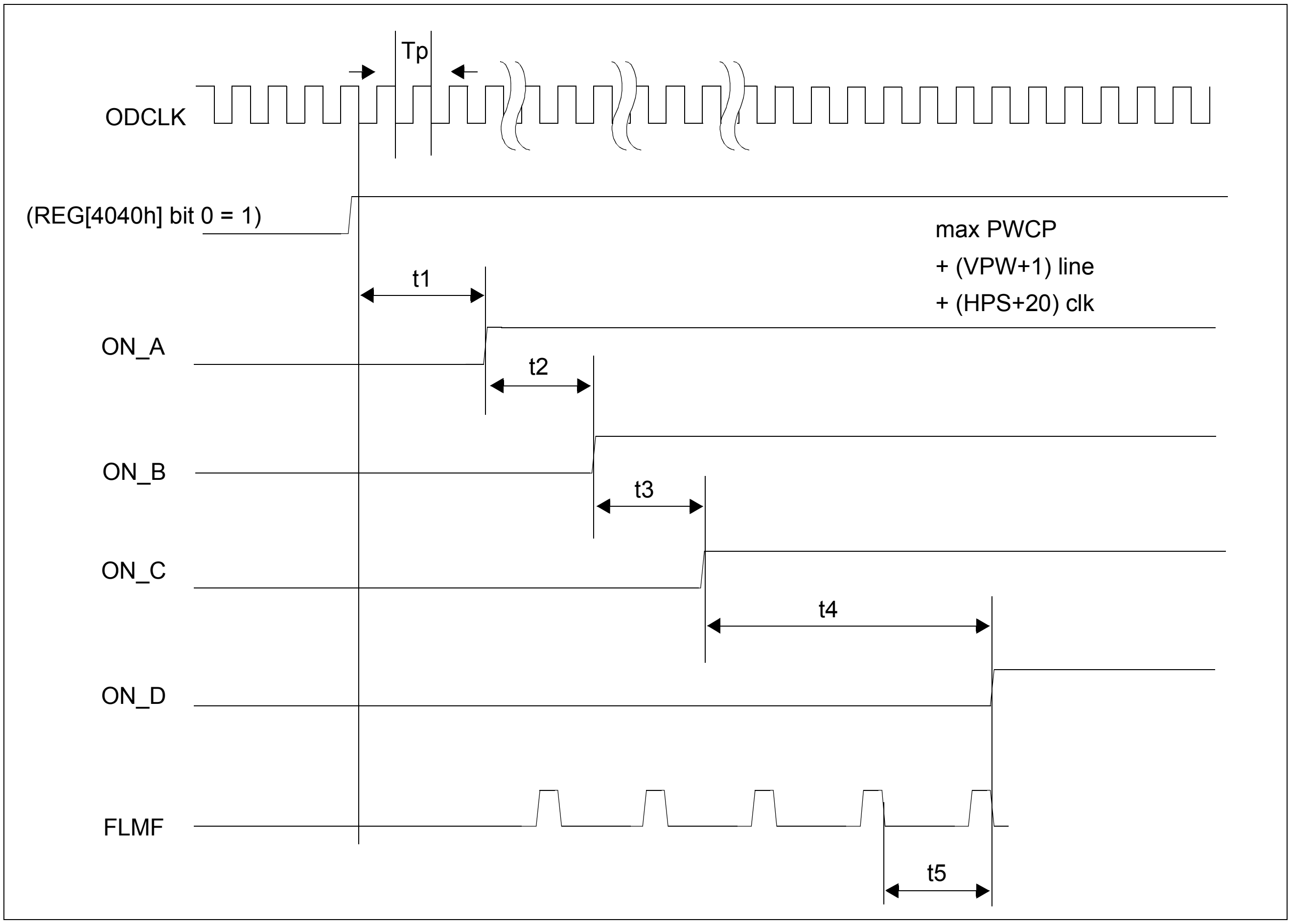


Figure 7-42: EID Double Screen Panel Start-Up Control Signals Timing

Table 7-51: EID Double Screen Panel Start-Up Control Signals Timing

Symbol	Description	Min	Typical	Max	Units
Tpwrclk	Period of internal PWR_CLK signal	—	5,242,880	—	Tp
t1	first ODCLK after Power On to ON_A	—	—	1	Tpwrclk
t2	ON_A high to ON_B high delay	—	1	—	Tpwrclk
t3	ON_B high to ON_C high delay	—	1	—	Tpwrclk
t4	ON_C to ON_D0 signal high delay	Tpwrclk + t5	—	Tpwrclk + 2(t5)	
t5	FLMF (Vertical Total (VSYNC Period))	—	VT	—	line

1. Tp = pixel clock period
2. VT = Vertical Total (VSYNC Period) = (REG[402Ah] bits 7-0, REG[402Bh] bits 3-0) +1

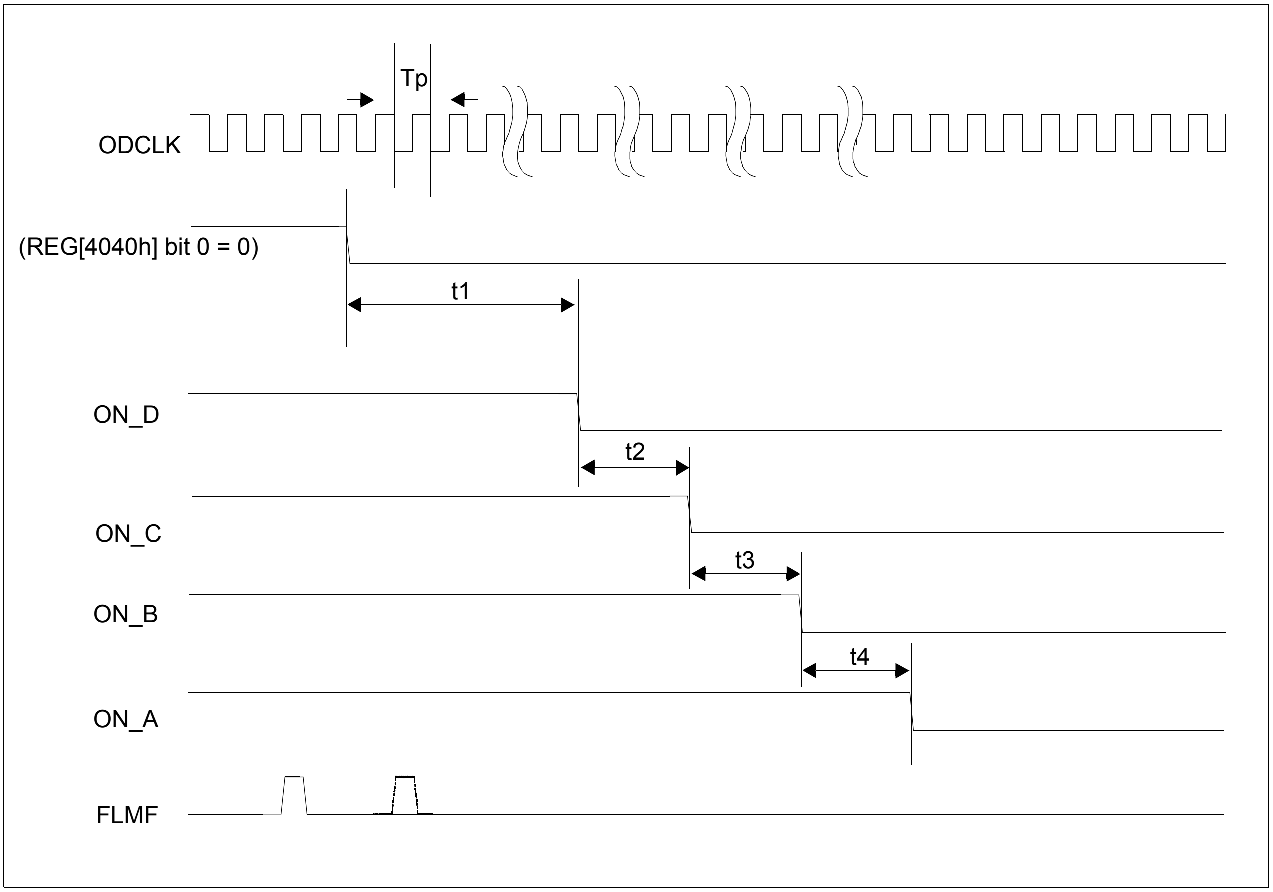


Figure 7-43: EID Double Screen Panel Shut-Down Control Signals Timing

Table 7-52: EID Double Screen Panel Shut-Down Control Signals Timing

Symbol	Parameter	Min	Typ	Max	Units
Tpwrclk	Period of internal PWR_CLK signal	—	5,242,880	—	Tp
t1	Power Off to ON_D delay	—	—	VT + 17Tpwrclk	
t2	ON_D low to ON_C low delay	—	1	—	Tpwrclk
t3	ON_C low to ON_B low delay	—	1	—	Tpwrclk
t4	ON_B low to ON_A low delay	—	1	—	Tpwrclk

1. Tp = pixel clock period
2. VT = Vertical Total (VSYNC Period) = (REG[402Ah] bits 7-0, REG[402Bh] bits 3-0) +1

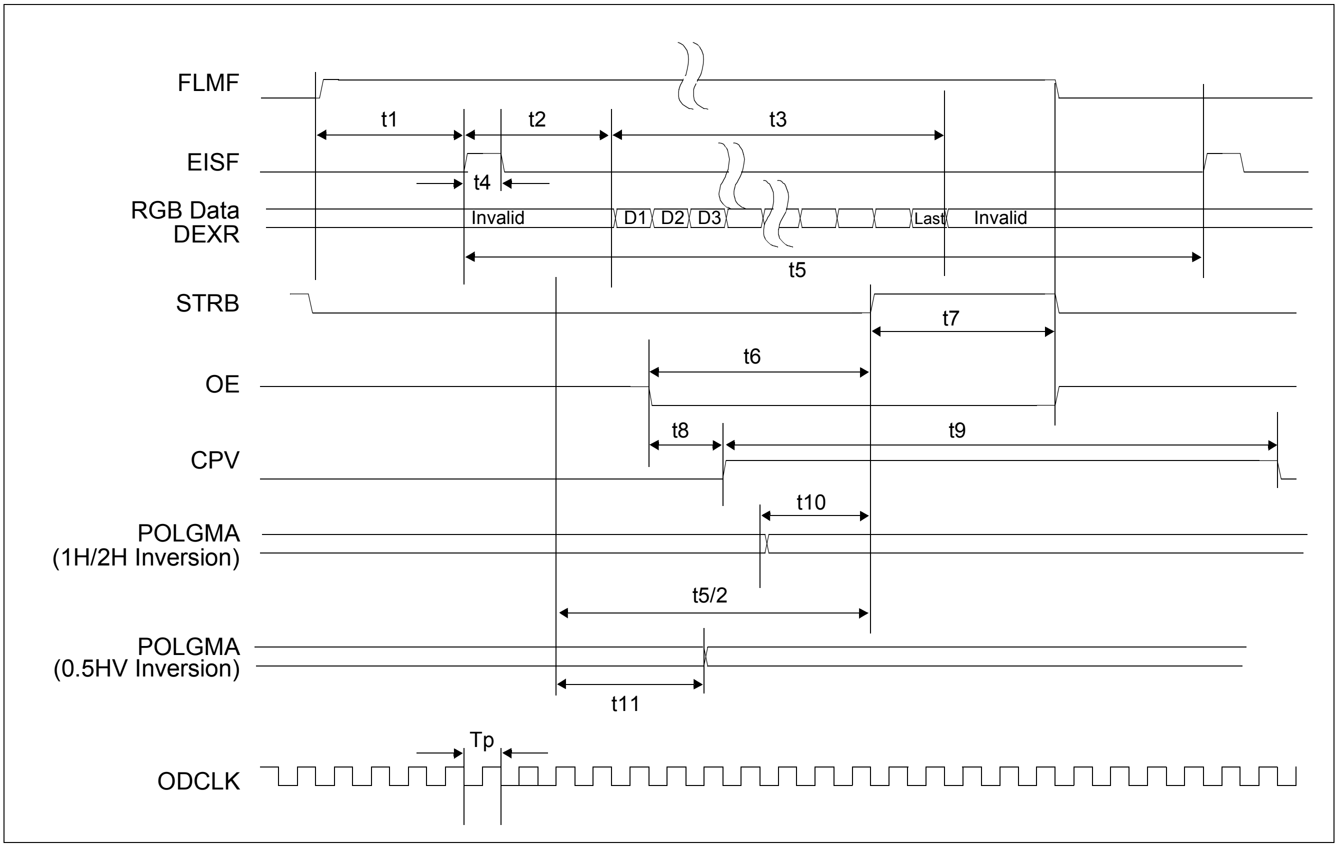


Figure 7-44: EID Double Screen Panel Horizontal Timing

Table 7-53: EID Double Screen Panel Horizontal Timing

Symbol	Description	Nominal	Units
Tp	ODCLK - Pixel Clock		
t1	FLMF rising edge to EISF rising edge	4	Tp
t2	Horizontal Display Period Start Position	(REG[4024h] bits 7-0, REG[4025h] bits 3-0) - 1	Tp
t3	Horizontal Display Period	(REG[4022h] bits 7-0, REG[4023h] bits 2-0) x 2	Tp
t4	EISF pulse width	1	Tp
t5	Horizontal Total (HSYNC period)	(REG[4020h] bits 7-0, REG[4021h] bits 3-0) + 1	Tp
t6	OE low width	REG[4046h] bits 7-0	Tp
t7	STRB rising to FLMF falling, OE rising	10	Tp
t8	OE falling to CPV rising	2	Tp
t9	CPV high width	50	Tp
t10	POLGMA 1H/2H Inversion to STRB rising	3	Tp
t11	POLGMA 0.5HV Inversion	REG[404Ah] bits 7-0	Tp

1. Tp = pixel clock period

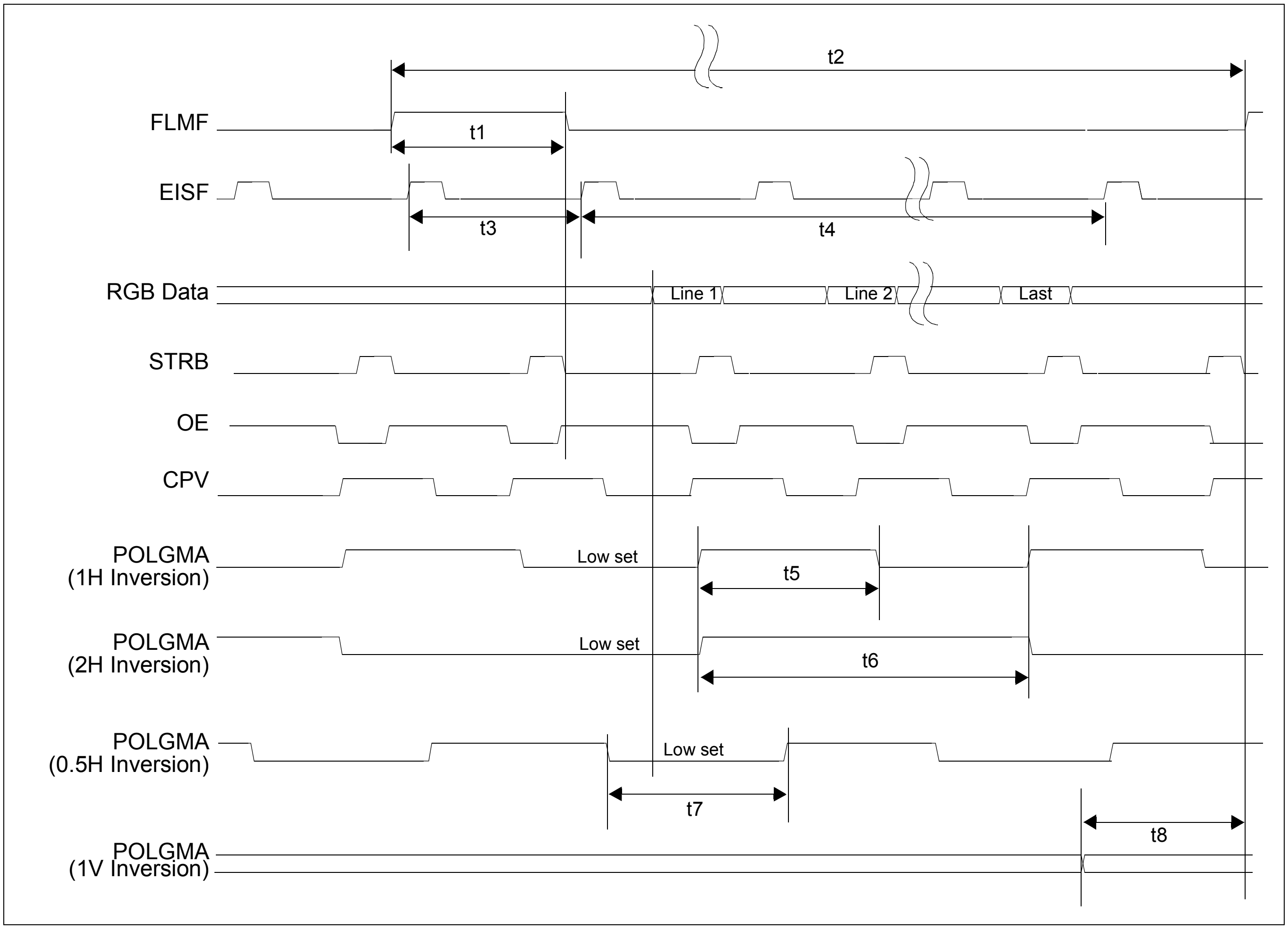


Figure 7-45: EID Double Screen Panel Vertical Timing

Table 7-54: EID Double Screen Panel Vertical Timing

Symbol	Description	Nominal	Units
t1	FLMF pulse width	1	line
t2	Vertical Total (VSYNC period)	REG[402Ah] bits 7-0, REG[402Bh] bits 3-0) +1	line
t3	Vertical Display Period Start Position	[(REG[402Eh] bits 7-0, REG[402Fh] bits 3-0) - 1	line
t4	Vertical Display Period	REG[402Ch] bits 7-0, REG[402Dh] bits 3-0) +1	line
t5	POLGMA 1H Inversion high width	1	line
t6	POLGMA 2H Inversion high width	2	line
t7	POLGMA 0.5H Inversion low width	1	line
t8	POLGMA 1V Inversion active to STRB falling	1	line

Note

1. EISF rising edge to Data/DEXR toggle timing

Table 7-55: EISF Rising Edge to Data/DEXR Toggle Timing

Hsync Polarity (REG[4027h] bit 7)	EID TCON Input Sync Polarity (REG[4041h] bit 4)	EISF Rise Edge to Data/Dexr Toggle Timing (H Back Porch)	Unit
0b	0b	HDPS	clk
	1b (Reserved)	—	
1b	0b (Reserved)	—	
	1b	HDPS	clk

Hsync Polarity (REG[4027h] bit 7) should be the same as EID TCON Input Sync Polarity (REG[4041h bit 4)
If Hsync Polarity is set to 0b (active-low), EID TCON Input Sync Polarity should be 0b (active-low).
If Hsync Polarity is set to 1b (active-high), EID TCON Input Sync Polarity should be 1b (active-high).

2. FLMF rising edge to Data/DEXR toggle timing

Table 7-56: FLMF Rising Edge to Data/DEXR Toggle Timing

Vsync Polarity (REG[4031h] bit 7)	EID TCON Input Sync Polarity (REG[4041h] bit 4)	FLMF Rise Edge to Data/Dexr Toggle Timing (V Back Porch)	Unit
0b	0b	VDPS - 1	line
	1b (Reserved)	—	
1b	0b (Reserved)	—	
	1b	VDPS - 1	line

Vsync Polarity (REG[4031h] bit 7) should be the same as EID TCON Input Sync Polarity (REG[4041h bit 4)
If Vsync Polarity is set to 0b (active-low), EID TCON Input Sync Polarity should be 0b (active-low).
If Vsync Polarity is set to 1b (active-high), EID TCON Input Sync Polarity should be 1b (active-high).

3. Horizontal Sync Pulse Width REG[4026h] should be greater than 1 (HPW minimum is 2clk width).
Vertical Sync Pulse Width REG[4030h] should be greater than 0 (VPW minimum is 1 line width).

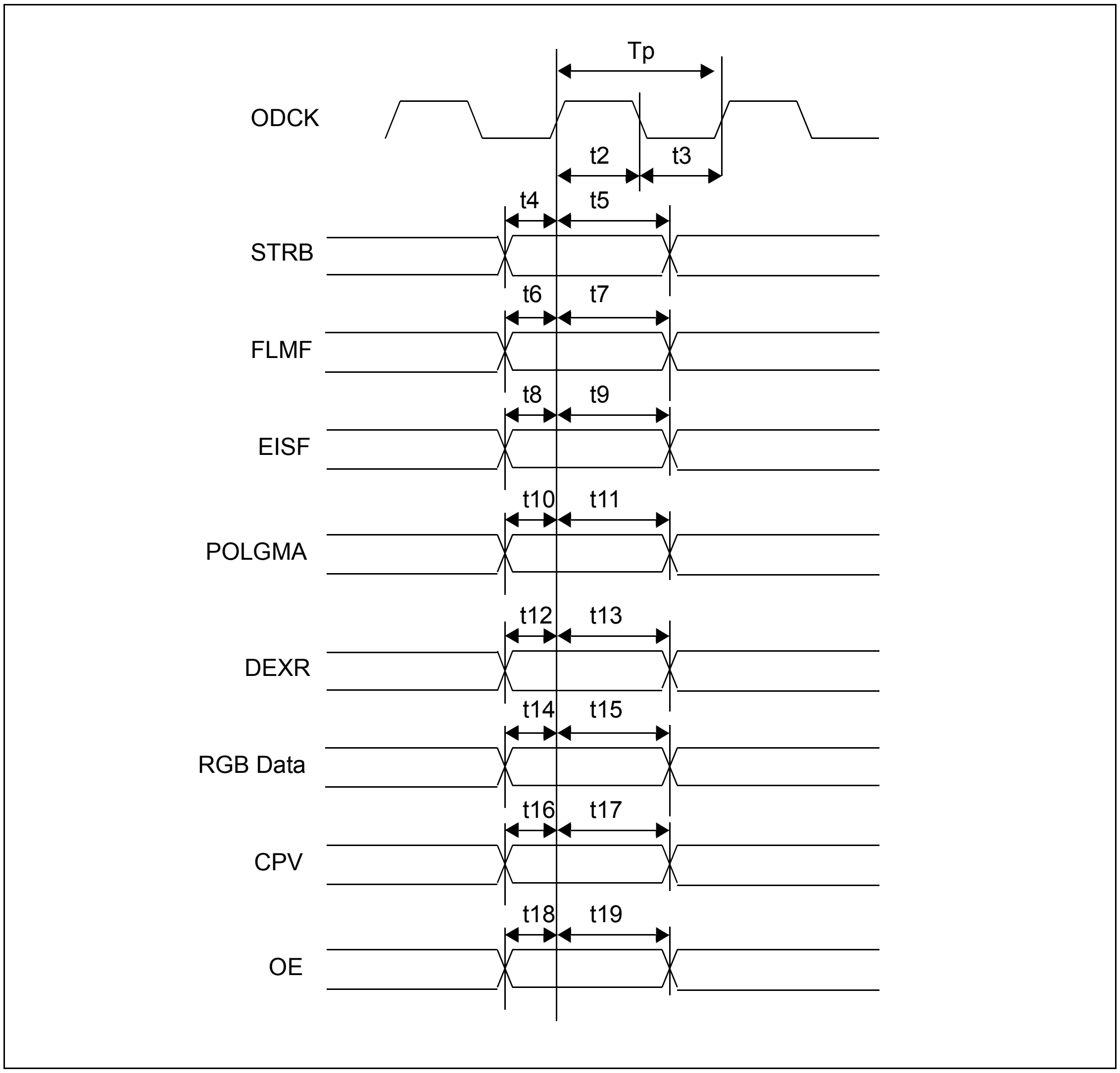


Figure 7-46: EID Double Screen Panel Timing

Table 7-57: EID Double Screen Panel Timing

Symbol	Parameter	Min	Typ	Max	Units
Tp	Pixel clock period	27.78	—	0.5Tp	ns
t2	Pixel clock pulse low	0.5Tp	—	0.5Tp+1.5ns	Tp
t3	Pixel clock pulse high	0.5Tp	—	0.5Tp+1.5ns	Tp
t4	STRB setup before CK rising edge	0.5Tp - 2ns	0.5	—	Tp
t5	STRB hold after CK rising edge	0.5Tp	0.5	—	Tp
t6	FLMF setup before CK rising edge	0.5Tp - 1ns	0.5	—	Tp
t7	FLMF hold after CK rising edge	0.5Tp - 1ns	0.5	—	Tp
t8	EISF setup before CK rising edge	0.5Tp - 1ns	0.5	—	Tp
t9	EISF hold after CK rising edge	0.5Tp - 1ns	0.5	—	Tp
t10	POLGMA setup before CK rising edge	0.5Tp - 3ns	0.5	—	Tp
t11	POLGMA hold after CK rising edge	0.5Tp	0.5	—	Tp
t12	DEXR setup before CK rising edge	0.5Tp - 4ns	0.5	—	Tp
t13	DEXR hold after CK rising edge	0.5Tp	0.5	—	Tp
t14	Pixel Data setup before CK rising edge	0.5Tp - 4ns	0.5	—	Tp
t15	Pixel Data hold after CK rising edge	0.5Tp	0.5	—	Tp
t16	CPV setup before CK rising edge	0.5Tp - 4ns	0.5	—	Tp
t17	CPV hold after CK rising edge	0.5Tp	0.5	—	Tp
t18	OE setup before CK rising edge	0.5Tp - 4ns	0.5	—	Tp
t19	OE hold after CK rising edge	0.5Tp	0.5	—	Tp

1. Tp = pixel clock period

7.7 Camera Interface Timing

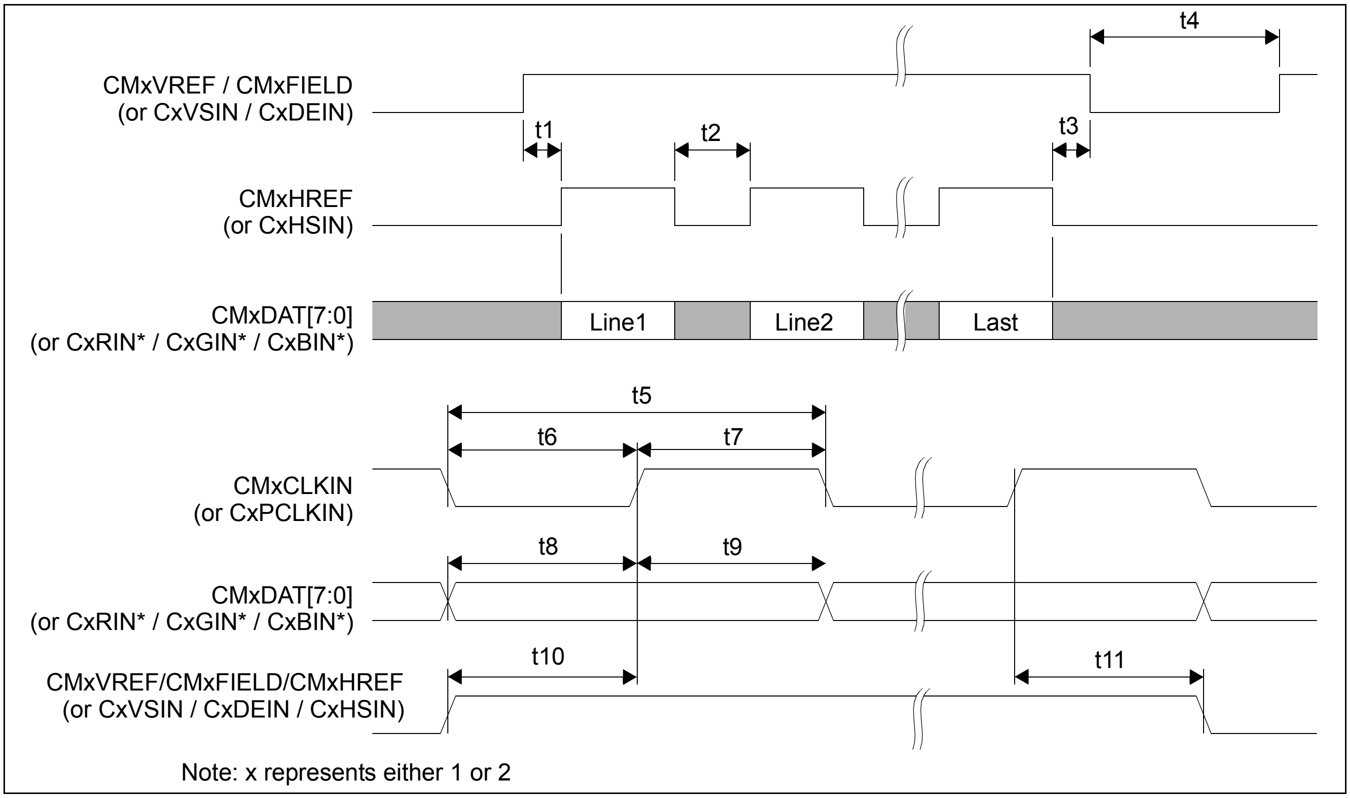


Figure 7-47: Camera Interface Timing

Table 7-58: Camera Interface Timing

Symbol	Parameter	Min	Max	Units
t1	CMxVREF/CMxFIELD rising edge to CMxHREF rising edge	0	—	Tc (Note 1)
t2	Horizontal blank period	1	—	Tc
t3	CMxHREF falling edge to CMxVREF falling edge	0	—	Tc
t4	Vertical blank period	1	—	Line
t5	Camera input clock period	1 (Note 3)	—	Ts (Note 2)
t6	Camera input clock pulse width low	4	—	ns
t7	Camera input clock pulse width high	4	—	ns
t8	Data setup time	2.4	—	ns
t9	Data hold time	3.8	—	ns
t10	CMxVREF, CMxFIELD, CMxHREF setup time	2.4	—	ns
t11	CMxVREF, CMxFIELD, CMxHREF hold time	3.8	—	ns

1. Tc = Camera block input clock period
2. Ts = System clock period
3. For RGB input streaming mode, REG[0D06h]/REG[0D46h] bits 2-1 = 10b, the minimum period is 2 Ts.

7.8 SDRAM Interface Timing

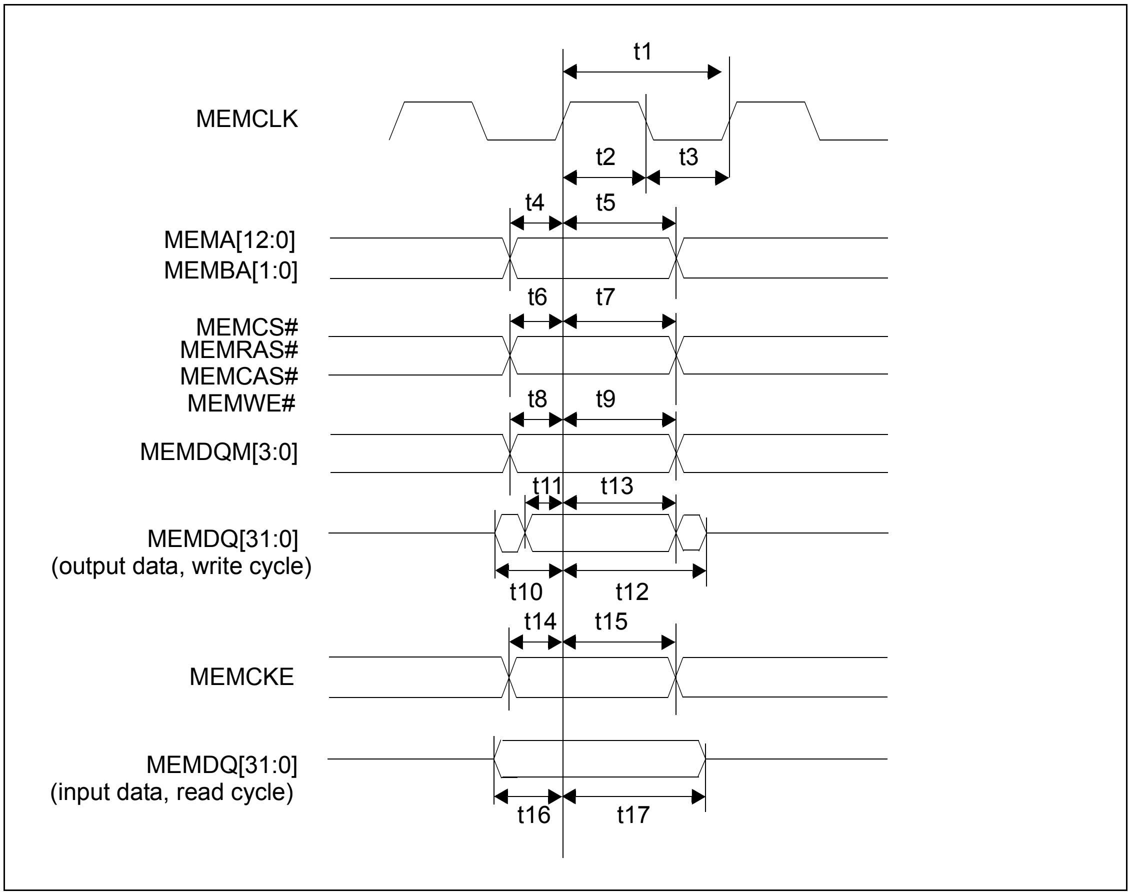


Figure 7-48: SDRAM Interface Timing

Table 7-59: SDRAM Interface Timing (Clock Source is PLL1)

Symbol	Parameter	Min	Max	Units
t1	MEMCLK cycle time	10.0	—	ns
t2	MEMCLK low pulse width	3.4	—	ns
t3	MEMCLK high pulse width	4.6	—	ns
t4	MEMA[12:0] and MEMBA[1:0] setup before MEMCLK rising	2.5	—	ns
t5	MEMA[12:0] and MEMBA[1:0] hold after MEMCLK rising	2.5	—	ns
t6	MEMCS#,MEMRAS#,MEMCAS#,MEMWE# setup before MEMCLK rising	2.5	—	ns
t7	MEMCS#,MEMRAS#,MEMCAS#,MEMWE# hold after MEMCLK rising	2.5	—	ns
t8	MEMDQM[3:0] setup before MEMCLK rising	2.5	—	ns
t9	MEMDQM[3:0] hold after MEMCLK rising	2.5	—	ns
t10	MEMCLK rising to MEMDQ[31:0] low-Z for write (see Note 1)	—	7.8	ns
t11	MEMDQ[31:0] output data setup before MEMCLK rising for write	2.9	—	ns
t12	MEMCLK rising to MEMDQ[31:0] high-Z for write (see Note 2)	2.4	6.1	ns
t13	MEMDQ[31:0] output data hold after MEMCLK rising for write	1.2	—	ns
t14	MEMCKE setup before MEMCLK rising	2.1	—	ns
t15	MEMCKE hold after MEMCLK rising	2.5	—	ns
t16	MEMDQ[31:0] input setup time for read	3.5	—	ns
t17	MEMDQ[31:0] input hold time for read	0	—	ns

1. MEMDQ[31:0] goes low-Z at the beginning of a write cycle, 2 clock periods before output data is available.
2. MEMDQ[31:0] does not go high-Z at the end of a write cycle and only goes high-Z at the start of the next read cycle.

7.9 I2S Interface Timing

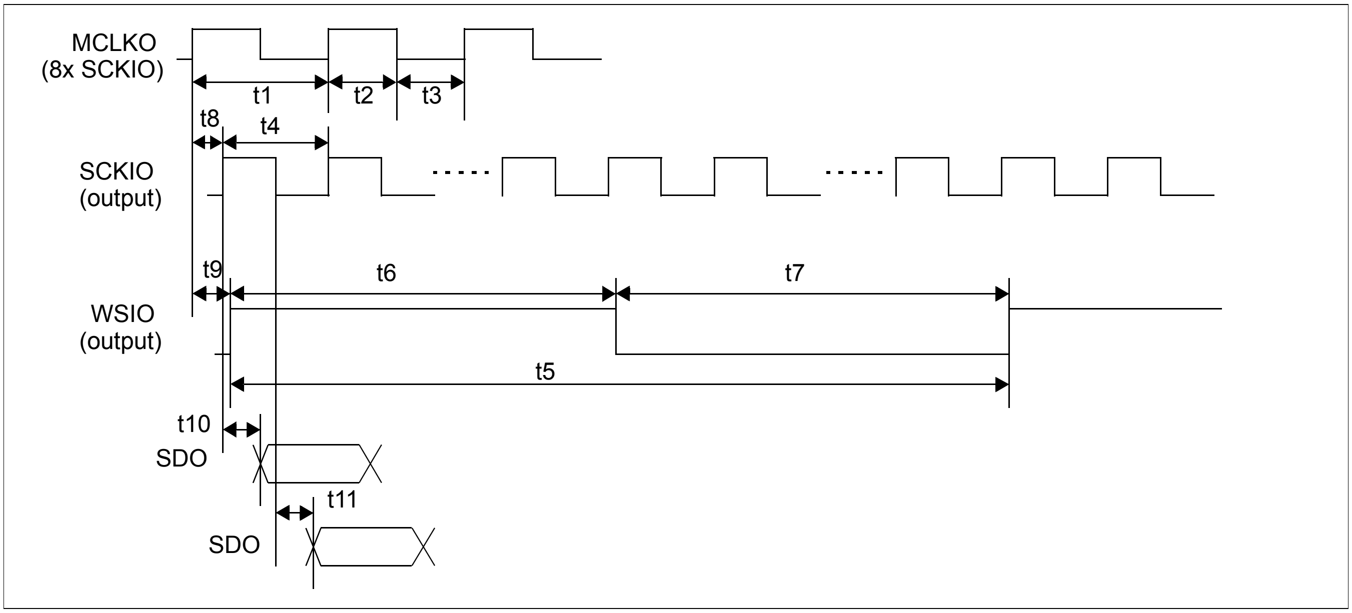


Figure 7-49 I2S Timing when SCKIO/WSIO are Outputs

Table 7-60 I2S Timing when SCKIO/WSIO are Outputs

Symbol	Description	Min / Nominal	Max	Units
t1	MCLKO period (see Note 1)	M	M + 1	Tsdram
t2	MCLKO high time (see Note 2)	N	N + 1	Tsdram
t3	MCLKO low time (see Note 2)	N	N + 1	Tsdram
t4	SCKIO output period	8	—	t1
t5	WSIO output period	32	—	t4
t6	WSIO output high time	16	—	t4
t7	WSIO output low time	16	—	t4
t8	MCLKO rising edge to SCKIO output rising/falling edge	—	2.7	ns
t9	MCLKO rising edge to WSIO output rising/falling edge	—	2.5	ns
t10	SCKIO output rising to SDO output valid (REG[0100h] bit 4 = 1b)	—	3.3	ns
t11	SCKIO output falling to SDO output valid (REG[0100h] bit 4 = 0b)	—	4.1	ns

1. Tsdram is one clock cycle period of the SDRAM clock which is Ts÷2, where Ts is the System Clock period. The MCLKO clock generator is a phase accumulator circuit which generates an average MCLKO output period of
   t1 = [65536 ÷ (REG[010Eh] bits 14-0)] Tsdram cycles

   The period of MCLKO will jitter between M and M+1 Tsdram cycles to generate the average period for t1, where M is the quotient of [65536 ÷ (REG[010Eh] bits 14-0)].
2. t2 and t3 will jitter between N and N+1 Tsdram clock cycles, where N is quotient of [32768 ÷ (REG[010Eh] bits 14-0)].

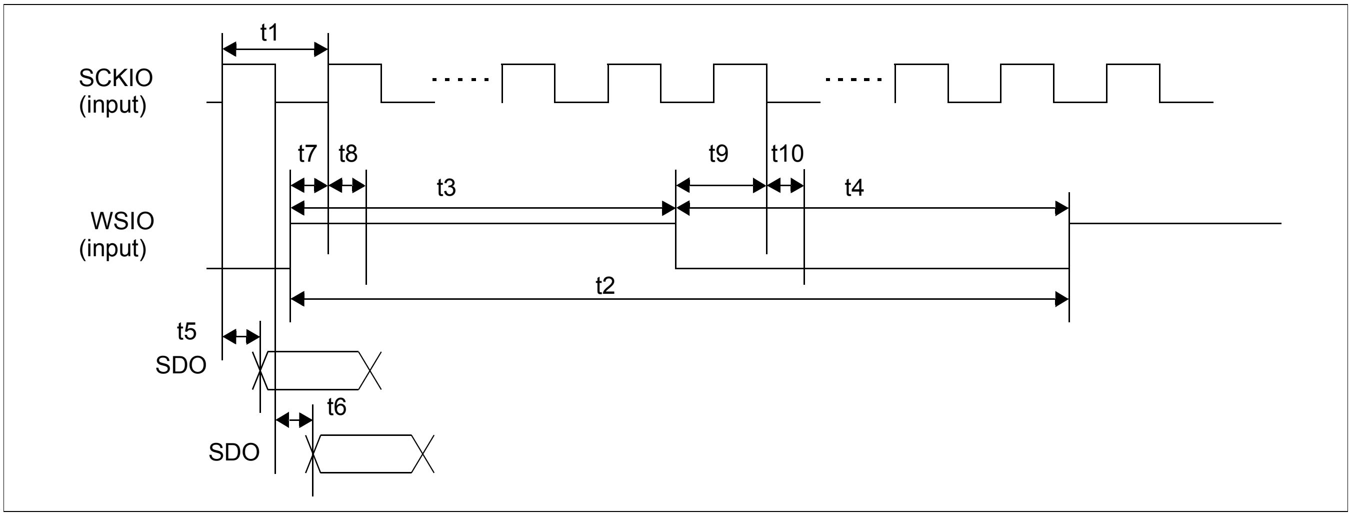


Figure 7-50 I2S Timing when SCKIO/WSIO are Inputs

Table 7-61 I2S Timing when SCKIO/WSIO are Inputs

Symbol	Description	Min / Nominal	Max	Units
t1	SCKIO period	—	—	—
t2	WSIO period	32	—	t1
t3	WSIO high time	16	—	t1
t4	WSIO low time	16	—	t1
t5	SCKIO rising to SDO output valid (REG[0100h] bit 4 = 1b)	—	15.7	ns
t6	SCKIO falling to SDO output valid (REG[0100h] bit 4 = 0b)	—	15.3	ns
t7	WSIO setup time before SCKIO rising (REG[0100h] bit 4 = 0b)	0	—	ns
t8	WSIO hold time after SCKIO rising (REG[0100h] bit 4 = 0b	1.4	—	ns
t9	WSIO setup time before SCKIO falling (REG[0100h] bit 4 = 1b)	0.4	—	ns
t10	WSIO hold time after SCKIO falling (REG[0100h] bit 4 = 1b)	1	—	ns

7.10 Keypad Interface Timing

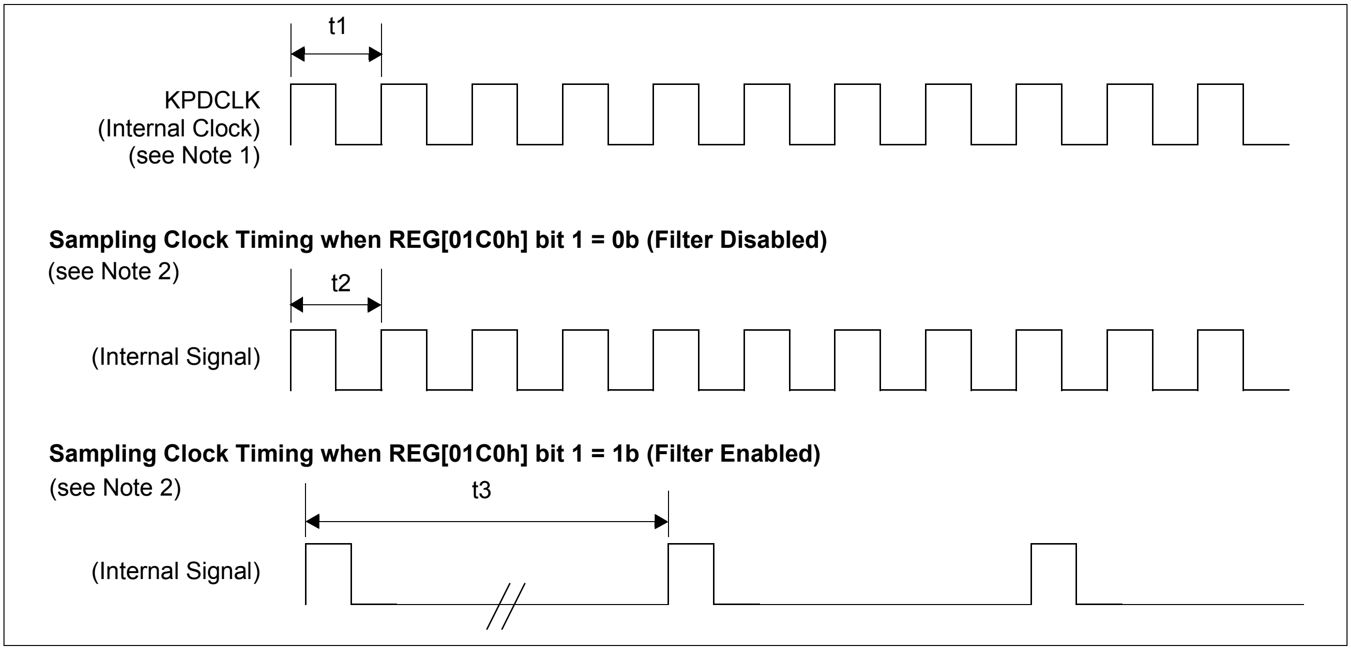


Figure 7-51: Keypad Interface Base Timing

Note

1. KPDCLK is an internal clock used for the Keypad interface. Users cannot see this clock.
2. Sampling Clock is the internal input sampling clock for the Keypad interface. Users cannot see this clock.

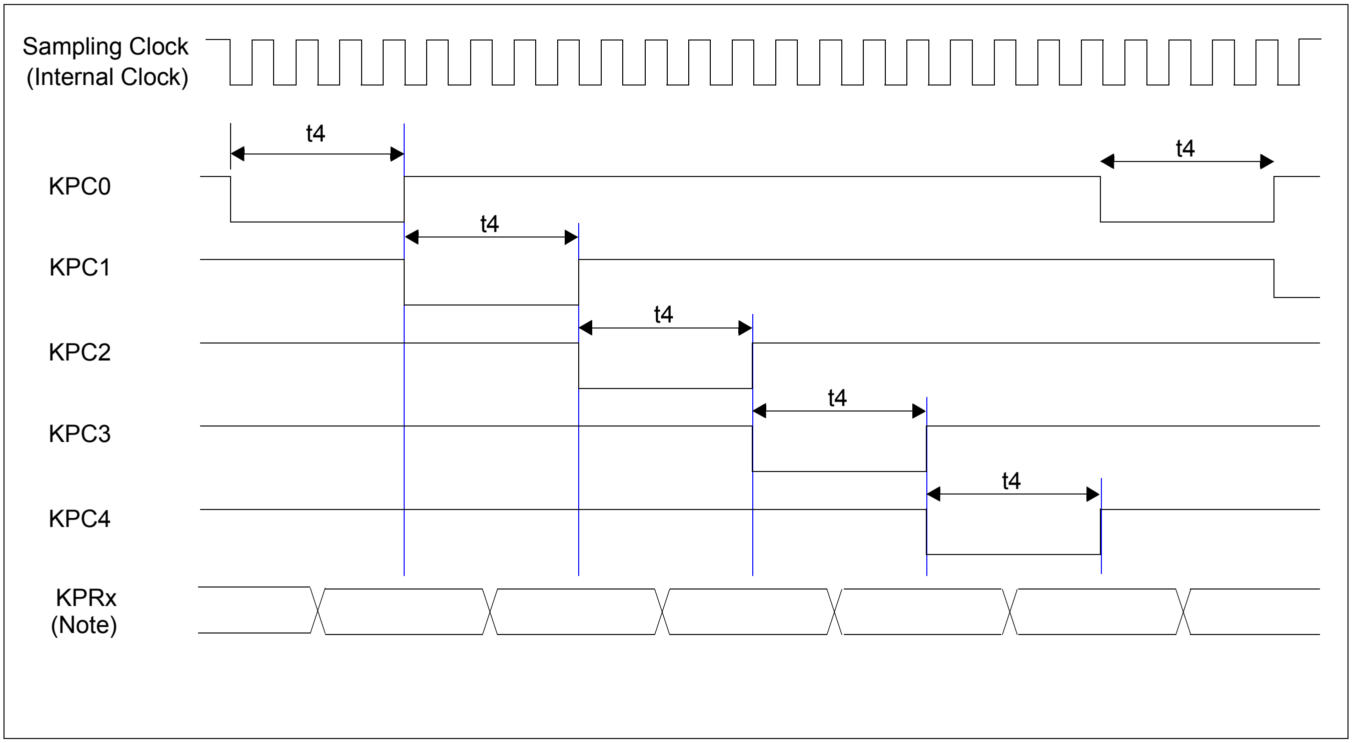


Figure 7-52: Keypad Interface Timing

Note
For Filter Disabled (REG[01C0h] bit 1 = 0b), KPRx are sampled/checked at the end of each KPCx pulse. For Filter Enabled (REG[01C0h] bit 1 = 1b), the filtered states of KPRx are sampled/checked at the end of each KPCx pulse. For details on filter input timing, see Figure 7-53: "Keypad Glitch Filter Input Timing," on page 128.

Table 7-62: Keypad Interface Timing

Symbol	Parameter	Min	Typ	Max	Units
t1	Keypad clock period (see Figure 7-51: on page 126)		Note 1		t_{INCLK1}
t2	Sampling Clock pulse width (same as t1) (see Figure 7-51: on page 126)		Note 1		t_{INCLK1}
t3	Sampling Clock pulse width (see Figure 7-51: on page 126)		Note 2		t1
t4	Key Driving Period		4 (Note 3)		t2 or t3

1) t1 is specified by REG[01D4h] ~ REG[01D5h].
2) t3 is specified by REG[01CCh] ~ REG[01CEh].
3) If REG[01C0h] bit 1 = 0b, t4 = (4 x t2). If REG[01C0h] bit 1 = 1b, t4 = (4 x t3)

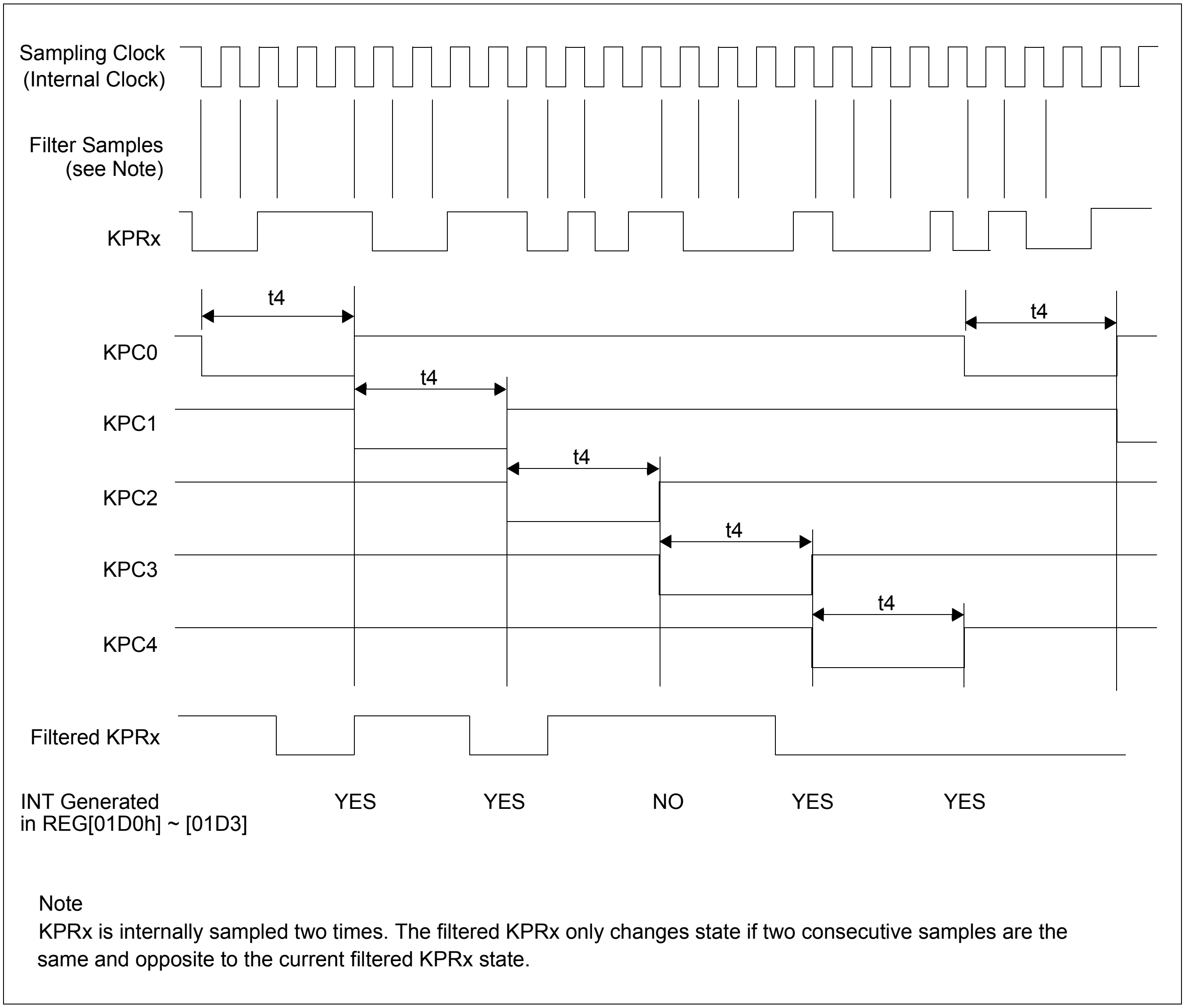


Figure 7-53: Keypad Glitch Filter Input Timing

Note

KPRx is internally sampled two times. The filtered KPRx only changes state if two consecutive samples are the same and opposite to the current filtered KPRx state.

7.11 Serial Flash (SPI) Interface Timing

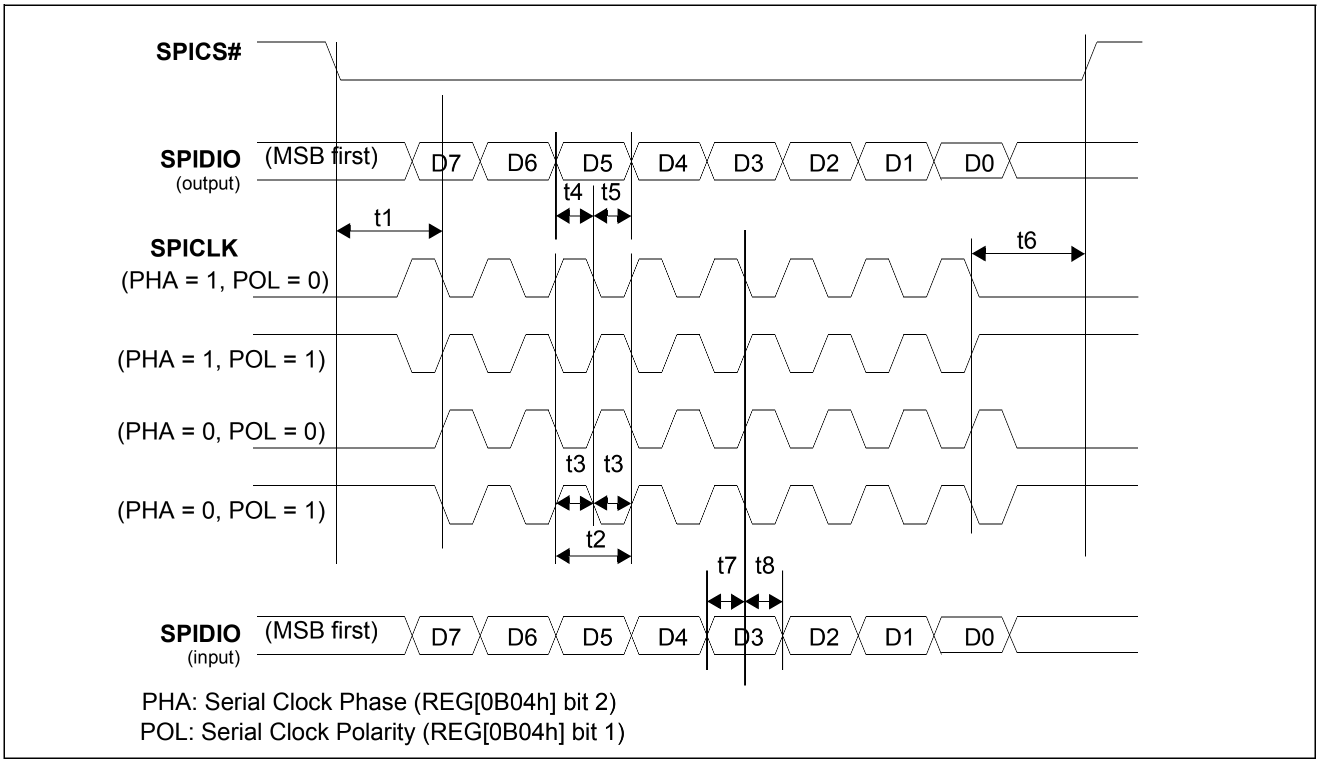


Figure 7-54: Serial Flash (SPI) Interface Timing

Table 7-63: Serial Flash (SPI) Interface Timing

Symbol	Parameter	Min	Typ	Max	Units
t1	Chip select low setup time (see Note 2)	Tmincsl - 0.7ns	—	—	Tsdram (Note 1)
t2	Serial clock period (see Note 3)	—	Tsck	—	Tsdram
t3	Serial clock pulse width low/high (see Note 4)	Thsckmin - 0.7ns	—	Thsckmax + 0.7ns	Tsdram
t4	Data output setup time (see Note 4)	Thsckmin - 1.6ns	—	—	Tsdram
t5	Data output hold time (see Note 4)	Thsckmin - 0.6ns	—	—	Tsdram
t6	Chip select high hold time (see Note 5)	Tmincsh + 0.3ns	—	—	Tsdram
t7	Data input setup time	13	—	—	ns
t8	Data input hold time	0	—	—	ns

1. Tsdram = SDRAM clock period in ns.
2. Tmincsl = ROUNDUP[(REG[0B04h] bits 5-3) ÷ 2] + (1 - (REG[0B04h] bit 3)) + 3
3. Tsck = [(REG[0B04h] bits 5-3) + 2]
4. Thsckmin + Thsckmax = Tsck
   Thsckmin = ROUNDDOWN[Tsck ÷ 2]
   Thsckmax = ROUNDUP[Tsck ÷ 2]
5. Tmincsh = Thsckmin + 1
6. Tmincshb = Thsckmin + 1

Chapter 8 Memory Map

The memory, devices, and slaves on all S1D13515/S2D13515 busses are treated as a single 32-bit memory-mapped address space.

Table 8-1: Memory Map

Current Address Range	Description
0400_0000h to 0400_7FFFh	Internal SRAM1 (32K bytes)
0400_8000h to 0400_FFFFh	Internal SRAM2 (32K bytes)
0401_0000h to 0401_7FFFh	Internal SRAM3 (32K bytes)
0430_0000h to 0430_FFFFh	Internal ROM (64K bytes)
1000_0000h to 1FFF_FFFFh	External SDRAM (up to 256M bytes)
2000_0000h to 2FFF_FFFFh	Serial Flash Read (up to 256M bytes) (see Note 1)
3800_0000h to 3800_FFFFh	Registers / APB Bus (including Keypad Interface, PWM)
3801_0000h to 3801_FFFFh	Reserved
4000_0000h to 4FFF_FFFFh	Bit Per Pixel Converter (BPPC) Port 0 (see Note 2)
5000_0000h to 5FFF_FFFFh	Bit Per Pixel Converter (BPPC) Port 1 (see Note 2)
6000_0000h to 6FFF_FFFFh	Bit Per Pixel Converter (BPPC) Port 2 (see Note 2)
7000_0000h to 7FFF_FFFFh	Bit Per Pixel Converter (BPPC) Port 3 (see Note 2)

Note

1. When SPI is disabled (REG[0B04h] bit 4 = 0b), the Serial Flash read area must not be accessed.
2. The Bit Per Pixel Converter (BPPC) Ports cannot be accessed through the Host interface. Accesses to and from the BPPC ports must be in 32-bit units.
3. DMAC may not burst access across more than 1 SRAM bank.
4. Enable “non-burst” mode in the DMAC in REG[3C0C] bit 6 or REG[3C1C] bit 6, if the DMAC transfer will cross SRAM banks.
5. The Sprite Engine is not allowed to access SRAM.

Chapter 9 Clocks

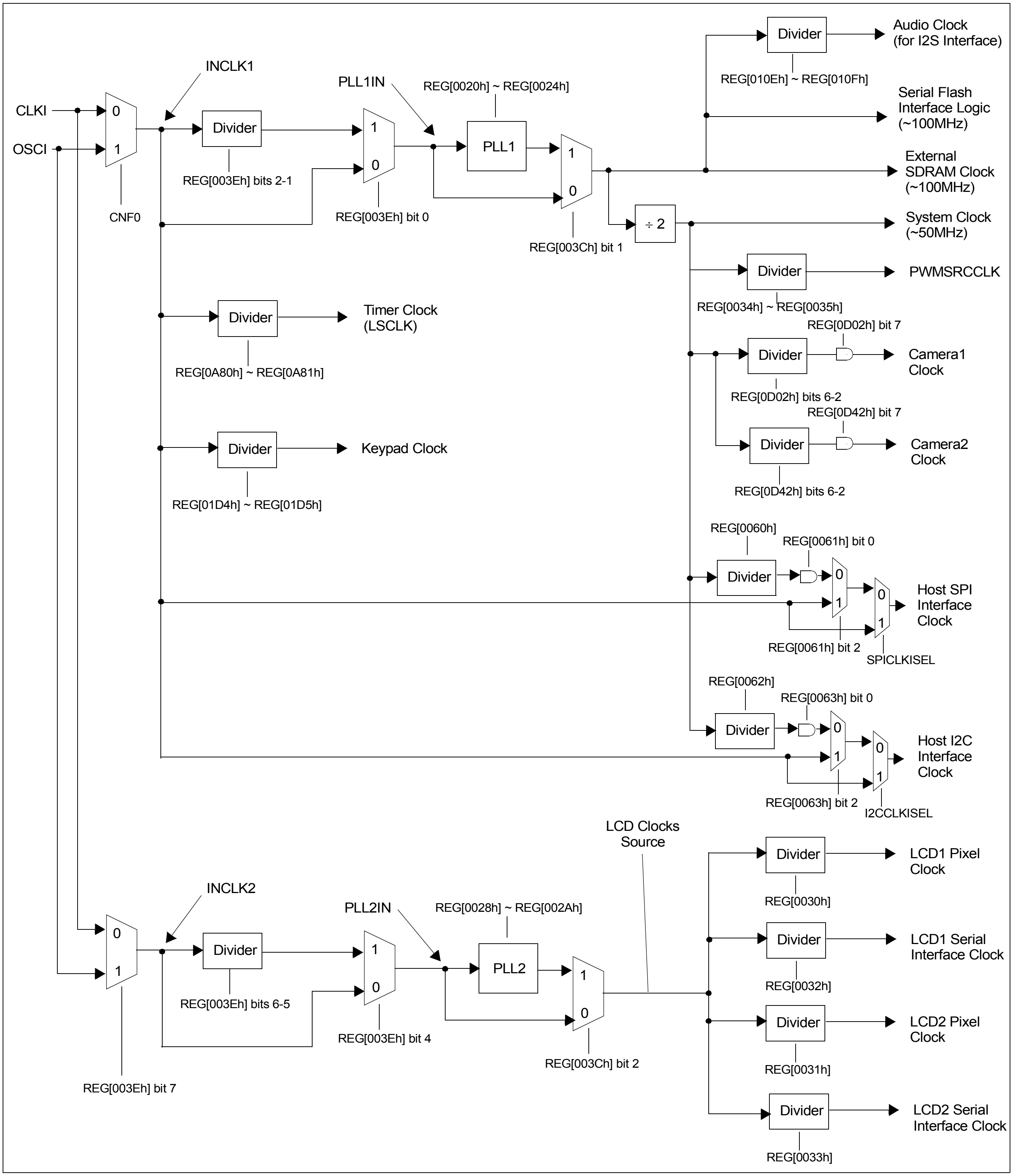


Figure 9-1: Clock Overview

Chapter 10 Registers

This section discusses how and where to access the S1D13515/S2D13515 registers. It also provides detailed information about the layout and usage of each register.

10.1 Register Mapping

The registers are memory-mapped. When the system decodes the input pins as CS# = 0 and M/R# = 0, the registers may be accessed.

Table 10-1: Memory/Register Selection

M/R#	Address	Size	Function
1	000000h to 1FFFFFh	2M bytes	Memory space
0	0000h to FFFFh	64Kbytes	Register space

The register space is decoded by AB[15:0] and is mapped as follows.

Table 10-2: Register Mapping

Address	Type	Function
System Control Registers		
0000h to 001Eh	Synchronous	System Control Registers (same as 3800_xxxxh of Internal Space, accessible by both Host and internal C33 processor)
0020h to 004Fh	Asynchronous	
0050h to 007Fh	Synchronous	
Host Interface Registers (accessible by Host only)		
0080h to 0081h	Asynchronous	MUADDR[31:16] - Internal Memory Space Upper Address Register
0082h	Asynchronous	MUMASK[20:16] - Internal Memory Space Upper Address Mask Register
0084h	Asynchronous	HOSTCTL[7:0] - Host Control Register
00A8h to 00ABh	Synchronous	MRWADDR[31:0] - Internal Memory Space Read/Write Address
00ACh to 00ADh	Synchronous	MRWDATA[15:0] - Internal Memory Space Read/Write Data Port
Internal Registers		
00B0h to FFFFh	Synchronous	Internal Registers (same as 3800_xxxxh of Internal Space, accessible by both Host and internal C33 processor)

Note
When Power Save Mode is enabled (REG[003Ch] bit 0 = 1b), only asynchronous registers may be accessed. Synchronous registers must not be accessed.

10.2 Register Set

The registers are listed in the following table.

Table 10-3: Register Set

Register	Page	Register	Page
System Control Registers			
REG[0000h] Product ID Register 0	143	REG[0001h] Product ID Register 1	143
REG[0002h] Product ID Register 2	143	REG[0003h] Product ID Register 3	143
REG[000Ch] through REG[000Fh] are Reserved	143		
REG[0010h] C33 TTBR Remap Address Register 0	144	REG[0011h] C33 TTBR Remap Address Register 1	144
REG[0012h] C33 TTBR Remap Address Register 2	144	REG[0013h] C33 TTBR Remap Address Register 3	144
REG[001Ch] C33 Control Register	144	REG[001Dh] C33 Software Reset Register	145
REG[001Eh] C33 Status Register	145		
REG[0020h] PLL1 Configuration Register 0	146	REG[0021h] PLL1 Configuration Register 1	147
REG[0022h] PLL1 Configuration Register 2	147	REG[0024h] PLL1 Control Register	148
REG[0028h] PLL2 Configuration Register 0	148	REG[0029h] PLL2 Configuration Register 1	149
REG[002Ah] PLL2 Configuration Register 2	150	REG[002Ch] PLL2 Control Register	150
REG[0030h] LCD1PCLK Configuration Register	150	REG[0031h] LCD2PCLK Configuration Register	151
REG[0032h] LCD1SCLK Configuration Register	152	REG[0033h] LCD2SCLK Configuration Register	153
REG[0034h] PWMSRCCLK Configuration Register 0	153	REG[0035h] PWMSRCCLK Configuration Register 1	153
REG[003Ch] Power Save Configuration Register	154	REG[003Dh] IO Drive Select Register	155
REG[003Eh] Input Clock Control Register	156	REG[0060h] Host SPI Clock Configuration Register	157
REG[0061h] Host SPI Enable Register	158	REG[0062h] Host I2C Clock Configuration Register	159
REG[0063h] Host I2C Enable Register	160		
Host Interface Registers			
REG[0080h] Internal Memory Space Upper Address Register 0	161	REG[0081h] Internal Memory Space Upper Address Register 1	161
REG[0082h] Internal Memory Space Upper Address Mask Register	161		
REG[0084h] Host Control Register 0	162	REG[0085h] Host Control Register 1	162
REG[008Ah] Host Control Register 2	163	REG[00A6h] Internal Memory Space Read/Write Control Register	163
REG[00A8h] Internal Memory Space Read/Write Address Register 0	164	REG[00A9h] Internal Memory Space Read/Write Address Register 1	164
REG[00AAh] Internal Memory Space Read/Write Address Register 2	164	REG[00ABh] Internal Memory Space Read/Write Address Register 3	164
REG[00ACh] Internal Memory Space Read/Write Data Port Register 0	165	REG[00ADh] Internal Memory Space Read/Write Data Port Register 1	165

Table 10-3: Register Set (Continued)

Register	Page	Register	Page
Bit Per Pixel Converter Configuration Registers			
REG[00B0h] BPPC Port 0 Mode Configuration Register 0	167	REG[00B1h] BPPC Port 0 Mode Configuration Register 1	167
REG[00B4h] BPPC Port 0 Base Register 0	168	REG[00B5h] BPPC Port 0 Base Register 1	168
REG[00B6h] BPPC Port 0 Base Register 2	168	REG[00B7h] BPPC Port 0 Base Register 3	168
REG[00B8h] BPPC Port 0 Mask Register 0	169	REG[00B9h] BPPC Port 0 Mask Register 1	169
REG[00BAh] BPPC Port 0 Mask Register 2	169	REG[00BBh] BPPC Port 0 Mask Register 3	169
REG[00BCh] BPPC Port 0 Target Base Register 0	170	REG[00BDh] BPPC Port 0 Target Base Register 1	170
REG[00BEh] BPPC Port 0 Target Base Register 2	170	REG[00BFh] BPPC Port 0 Target Base Register 3	170
REG[00C0h] BPPC Port 1 Mode Configuration Register 0	171	REG[00C1h] BPPC Port 1 Mode Configuration Register 1	171
REG[00C4h] BPPC Port 1 Base Register 0	172	REG[00C5h] BPPC Port 1 Base Register 1	172
REG[00C6h] BPPC Port 1 Base Register 2	172	REG[00C7h] BPPC Port 1 Base Register 3	172
REG[00C8h] BPPC Port 1 Mask Register 0	173	REG[00C9h] BPPC Port 1 Mask Register 1	173
REG[00CAh] BPPC Port 1 Mask Register 2	173	REG[00CBh] BPPC Port 1 Mask Register 3	173
REG[00CCh] BPPC Port 1 Target Base Register 0	174	REG[00CDh] BPPC Port 1 Target Base Register 1	174
REG[00CEh] BPPC Port 1 Target Base Register 2	174	REG[00CFh] BPPC Port 1 Target Base Register 3	174
REG[00D0h] BPPC Port 2 Mode Configuration Register 0	175	REG[00D1h] BPPC Port 2 Mode Configuration Register 1	175
REG[00D4h] BPPC Port 2 Base Register 0	176	REG[00D5h] BPPC Port 2 Base Register 1	176
REG[00D6h] BPPC Port 2 Base Register 2	176	REG[00D7h] BPPC Port 2 Base Register 3	176
REG[00D8h] BPPC Port 2 Mask Register 0	177	REG[00D9h] BPPC Port 2 Mask Register 1	177
REG[00DAh] BPPC Port 2 Mask Register 2	177	REG[00DBh] BPPC Port 2 Mask Register 3	177
REG[00DCh] BPPC Port 2 Target Base Register 0	178	REG[00DDh] BPPC Port 2 Target Base Register 1	178
REG[00DEh] BPPC Port 2 Target Base Register 2	178	REG[00DFh] BPPC Port 2 Target Base Register 3	178
REG[00E0h] BPPC Port 3 Mode Configuration Register 0	179	REG[00E1h] BPPC Port 3 Mode Configuration Register 1	179
REG[00E4h] BPPC Port 3 Base Register 0	180	REG[00E5h] BPPC Port 3 Base Register 1	180
REG[00E6h] BPPC Port 3 Base Register 2	180	REG[00E7h] BPPC Port 3 Base Register 3	180
REG[00E8h] BPPC Port 3 Mask Register 0	181	REG[00E9h] BPPC Port 3 Mask Register 1	181
REG[00EAh] BPPC Port 3 Mask Register 2	181	REG[00EBh] BPPC Port 3 Mask Register 3	181
REG[00ECh] BPPC Port 3 Target Base Register 0	182	REG[00EDh] BPPC Port 3 Target Base Register 1	182
REG[00EEh] BPPC Port 3 Target Base Register 2	182	REG[00EFh] BPPC Port 3 Target Base Register 3	182
I2S Control Registers			
REG[0100h] I2S Interface Control Register 0	183	REG[0101h] I2S Interface Control Register 1	184
REG[0104h] I2S FIFO Register 0	185	REG[0105h] I2S FIFO Register 1	186
REG[010Ah] I2S FIFO Status Register 0	187	REG[010Ch] I2S FIFO Status Register 1	187
REG[010Eh] I2S Audio Clock Control Register 0	188	REG[010Fh] I2S Audio Clock Control Register 1	188

Table 10-3: Register Set (Continued)

Register	Page	Register	Page
I2S DMA Registers			
REG[0148h] I2S DMA Buffer 0 Address Register 0	189	REG[0149h] I2S DMA Buffer 0 Address Register 1	189
REG[014Ah] I2S DMA Buffer 0 Address Register 2	189	REG[014Bh] I2S DMA Buffer 0 Address Register 3	189
REG[014Ch] I2S DMA Buffer 1 Address Register 0	190	REG[014Dh] I2S DMA Buffer 1 Address Register 1	190
REG[014Eh] I2S DMA Buffer 1 Address Register 2	190	REG[014Fh] I2S DMA Buffer 1 Address Register 3	190
REG[0152h] I2S DMA Buffers Size Register 0	190	REG[0153h] I2S DMA Buffers Size Register 1	190
REG[0154h] I2S DMA Status Register	191		
GPIO Registers			
REG[0180h] GPIO Configuration Register 0	192	REG[0181h] GPIO Configuration Register 1	192
REG[0182h] GPIO Status Register 0	192	REG[0183h] GPIO Status Register 1	192
REG[0184h] GPIO Pull-down Control Register 0	193	REG[0185h] GPIO Pull-down Control Register 1	193
REG[0186h] GPIO[15:8] / Keypad Configuration Register	193	REG[0188h] Miscellaneous Pull-up/Pull-down Register 0	194
REG[0189h] Miscellaneous Pull-up/Pull-down Register 1	195		
Keypad Registers			
REG[01C0h] Keypad Control Register	198	REG[01C4h] Keypad Interrupt Enable Register 0	199
REG[01C5h] Keypad Interrupt Enable Register 1	199	REG[01C6h] Keypad Interrupt Enable Register 2	199
REG[01C7h] Keypad Interrupt Enable Register 3	199	REG[01C8h] Keypad Input Polarity Register 0	200
REG[01C9h] Keypad Input Polarity Register 1	200	REG[01CAh] Keypad Input Polarity Register 2	200
REG[01CBh] Keypad Input Polarity Register 3	200	REG[01CCh] Keypad Filter Sampling Period Register 0	201
REG[01CDh] Keypad Filter Sampling Period Register 1	201	REG[01CEh] Keypad Filter Sampling Period Register 2	201
REG[01D0h] Keypad Interrupt Raw Status/Clear Register 0	202	REG[01D1h] Keypad Interrupt Raw Status/Clear Register 1	202
REG[01D2h] Keypad Interrupt Raw Status/Clear Register 2	202	REG[01D3h] Keypad Interrupt Raw Status/Clear Register 3	202
REG[01D4h] Keypad Clock Configuration Register 0	203	REG[01D5h] Keypad Clock Configuration Register 1	203
REG[01D6h] Keypad GPI Function Enable Register	203		
PWM Registers			
REG[0200h] PWM Control Register	204	REG[0201h] PWM1 Enable/On Register	206
REG[0202h] PWM1 Off Register	206	REG[0203h] PWM1 Control Register	206
REG[0204h] PWM2 Enable/On Register	207	REG[0205h] PWM2 Off Register	207
REG[0206h] PWM2 Control Register	208		
SDRAM Read/Write Buffer Registers			
REG[0240h] SDRAM Buffer 0 Configuration Register	209	REG[0242h] SDRAM Buffer 0 Control Register	210
REG[0244h] SDRAM Buffer 0 Read Bytes Register	211		
REG[0248h] SDRAM Buffer 0 Target Address Register 0	211	REG[0249h] SDRAM Buffer 0 Target Address Register 1	211
REG[024Ah] SDRAM Buffer 0 Target Address Register 2	211	REG[024Bh] SDRAM Buffer 0 Target Address Register 3	211
REG[024Ch] SDRAM Buffer 0 Data Port Register 0	212	REG[024Dh] SDRAM Buffer 0 Data Port Register 1	212
REG[0250h] SDRAM Buffer 1 Configuration Register	213	REG[0252h] SDRAM Buffer 1 Control Register	214
REG[0254h] SDRAM Buffer 1 Read Bytes Register	215		
REG[0258h] SDRAM Buffer 1 Target Address Register 0	215	REG[0259h] SDRAM Buffer 1 Target Address Register 1	215
REG[025Ah] SDRAM Buffer 1 Target Address Register 2	215	REG[025Bh] SDRAM Buffer 1 Target Address Register 3	215
REG[025Ch] SDRAM Buffer 1 Data Port Register 0	216	REG[025Dh] SDRAM Buffer 1 Data Port Register 1	216
REG[0260h] SDRAM Buffer 0 Rectangular Increment Register 0	216	REG[0261h] SDRAM Buffer 0 Rectangular Increment Register 1	216
REG[0262h] SDRAM Buffer 1 Rectangular Increment Register 0	217	REG[0263h] SDRAM Buffer 1 Rectangular Increment Register 1	217
REG[0264h] SDRAM Read/Write Buffer Internal Address Register 0	217	REG[0265h] SDRAM Read/Write Buffer Internal Address Register 1	217
REG[0266h] SDRAM Read/Write Buffer Internal Address Register 2	217	REG[0267h] SDRAM Read/Write Buffer Internal Address Register 3	217
REG[0300h] ~ REG[037Eh] (Even Addresses) Aliased SDRAM Buffer 0 Data Port Register 0	218	REG[0301h] ~ REG[037Fh] (Odd Addresses) Aliased SDRAM Buffer 0 Data Port Register 1	218
REG[0380h] ~ REG[03FEh] (Even Addresses) Aliased SDRAM Buffer 1 Data Port Register 0	219	REG[0381h] ~ REG[03FFh] (Odd Addresses) Aliased SDRAM Buffer 1 Data Port Register 1	219

Table 10-3: Register Set (Continued)

Register	Page	Register	Page
Warp Logic Configuration Registers			
REG[0400h] Warp Logic Configuration Register	220	REG[0402h] Warp Logic Event Flag Register	221
REG[0404h] Warp Logic Event Enable Register	221	REG[0406h] Warp Logic Event Clear Register	223
REG[0408h] Warp Logic Frame Status Register	223	REG[040Ah] Warp Logic Frame Ready Set Register	224
REG[0410h] Warp Logic Input Width Register 0	225	REG[0411h] Warp Logic Input Width Register 1	225
REG[0412h] Warp Logic Input Height Register 0	225	REG[0413h] Warp Logic Input Height Register 1	225
REG[0414h] Warp Logic Output Width Register 0	226	REG[0415h] Warp Logic Output Width Register 1	226
REG[0416h] Warp Logic Output Height Register 0	226	REG[0417h] Warp Logic Output Height Register 1	226
REG[0420h] Warp Logic Frame Buffer 0 Start Address Register 0	227	REG[0421h] Warp Logic Frame Buffer 0 Start Address Register 1	227
REG[0422h] Warp Logic Frame Buffer 0 Start Address Register 2	227	REG[0423h] Warp Logic Frame Buffer 0 Start Address Register 3	227
REG[0424h] Warp Logic Frame Buffer 1 Start Address Register 0	228	REG[0425h] Warp Logic Frame Buffer 1 Start Address Register 1	228
REG[0426h] Warp Logic Frame Buffer 1 Start Address Register 2	228	REG[0427h] Warp Logic Frame Buffer 1 Start Address Register 3	228
REG[0430h] Warp Logic Background Color Blue Register	228	REG[0431h] Warp Logic Background Color Green Register	229
REG[0432h] Warp Logic Background Color Red Register	229	REG[0434h] Warp Logic Input X Offset Register 0	230
REG[0435h] Warp Logic Input X Offset Register 1	230	REG[0436h] Warp Logic Input Y Offset Register 0	230
REG[0437h] Warp Logic Input Y Offset Register 1	230	REG[0440h] Warp Logic Offset Table Configuration Register	231
REG[0444h] Warp Logic Offset Table SDRAM Start Address Register 0	232	REG[0445h] Warp Logic Offset Table SDRAM Start Address Register 1	232
REG[0446h] Warp Logic Offset Table SDRAM Start Address Register 2	232	REG[0447h] Warp Logic Offset Table SDRAM Start Address Register 3	232
REG[0450h] Warp Logic Luminance Table Configuration Register 0	233	REG[0452h] Warp Logic Luminance Table Configuration Register 1	234
REG[0454h] Warp Logic Luminance Table SDRAM Start Address Register 0	235	REG[0455h] Warp Logic Luminance Table SDRAM Start Address Register 1	235
REG[0456h] Warp Logic Luminance Table SDRAM Start Address Register 2	235	REG[0457h] Warp Logic Luminance Table SDRAM Start Address Register 3	235
Blending Engine Configuration Registers			
REG[0900h] CH1OUT Control Register	236	REG[0904h] CH1OUT Writeback Frame Buffer 0 Address Register 0	237
REG[0905h] CH1OUT Writeback Frame Buffer 0 Address Register 1	237	REG[0906h] CH1OUT Writeback Frame Buffer 0 Address Register 2	237
REG[0907h] CH1OUT Writeback Frame Buffer 0 Address Register 3	237	REG[0908h] CH1OUT Writeback Frame Buffer 1 Address Register 0	238
REG[0909h] CH1OUT Writeback Frame Buffer 1 Address Register 1	238	REG[090Ah] CH1OUT Writeback Frame Buffer 1 Address Register 2	238
REG[090Bh] CH1OUT Writeback Frame Buffer 1 Address Register 3	238		
REG[090Ch] Scratchpad Register 0	238	REG[090Dh] Scratchpad Register 1	238
REG[090Eh] Scratchpad Register 2	238	REG[090Fh] Scratchpad Register 3	238
REG[0920h] CH2OUT Control Register	239		
REG[0930h] OSDOUT Control Register	239	REG[0940h] MAIN Window Control Register	240
REG[0942h] MAIN Window Frame Control/Status Register	241	REG[0944h] MAIN Blank Color Blue Register	243
REG[0945h] MAIN Blank Color Green Register	243	REG[0946h] MAIN Blank Color Red Register	243
REG[0948h] MAIN Window Frame Buffer 0 Address Register 0	244	REG[0949h] MAIN Window Frame Buffer 0 Address Register 1	244
REG[094Ah] MAIN Window Frame Buffer 0 Address Register 2	244	REG[094Bh] MAIN Window Frame Buffer 0 Address Register 3	244
REG[094Ch] MAIN Window Frame Buffer 1 Address Register 0	245	REG[094Dh] MAIN Window Frame Buffer 1 Address Register 1	245
REG[094Eh] MAIN Window Frame Buffer 1 Address Register 2	245	REG[094Fh] MAIN Window Frame Buffer 1 Address Register 3	245
REG[0950h] MAIN Window Width Register 0	246	REG[0951h] MAIN Window Width Register 1	246
REG[0952h] MAIN Window Height Register 0	246	REG[0953h] MAIN Window Height Register 1	246

Table 10-3: Register Set (Continued)

Table 10-3: Register Set (Continued)

Register	Page	Register	Page
Image Fetcher Configuration Registers			
REG[09AAh] Image Fetcher Input X Offset Register 0	272	REG[09ABh] Image Fetcher Input X Offset Register 1	272
REG[09ACh] Image Fetcher Input Y Offset Register 0	272	REG[09ADh] Image Fetcher Input Y Offset Register 1	272
REG[09B0h] Image Fetcher Control Register	272	REG[09B2h] Image Fetcher Frame Control/Status Register	274
REG[09B4h] Image Fetcher Blank Color Blue Register	275	REG[09B5h] Image Fetcher Blank Color Green Register	275
REG[09B6h] Image Fetcher Blank Color Red Register	275		
REG[09B8h] Image Fetcher Frame Buffer 0 Address Register 0	276	REG[09B9h] Image Fetcher Frame Buffer 0 Address Register 1	276
REG[09BAh] Image Fetcher Frame Buffer 0 Address Register 2	276	REG[09BBh] Image Fetcher Frame Buffer 0 Address Register 3	276
REG[09BCh] Image Fetcher Frame Buffer 1 Address Register 0	277	REG[09BDh] Image Fetcher Frame Buffer 1 Address Register 1	277
REG[09BEh] Image Fetcher Frame Buffer 1 Address Register 2	277	REG[09BFh] Image Fetcher Frame Buffer 1 Address Register 3	277
REG[09C0h] Image Fetcher Width Register 0	278	REG[09C1h] Image Fetcher Width Register 1	278
REG[09C2h] Image Fetcher Height Register 0	278	REG[09C3h] Image Fetcher Height Register 1	278
REG[09C4h] Image Fetcher Virtual Width Register 0	278	REG[09C5h] Image Fetcher Virtual Width Register 1	278
LCD Configuration Registers			
REG[09C8h] LCD Control A Register	279	REG[09CAh] LCD Control B Register	281
REG[09D0h] Warp Writeback Frame Buffer 0 Address Register 0	282	REG[09D1h] Warp Writeback Frame Buffer 0 Address Register 1	282
REG[09D2h] Warp Writeback Frame Buffer 0 Address Register 2	282	REG[09D3h] Warp Writeback Frame Buffer 0 Address Register 3	282
REG[09D4h] Warp Writeback Frame Buffer 1 Address Register 0	283	REG[09D5h] Warp Writeback Frame Buffer 1 Address Register 1	283
REG[09D6h] Warp Writeback Frame Buffer 1 Address Register 2	283	REG[09D7h] Warp Writeback Frame Buffer 1 Address Register 3	283
REG[09D8h] LCD Frame Control A Register 0	284	REG[09D9h] LCD Frame Control A Register 1	285
REG[09DAh] LCD Frame Control B Register 0	286	REG[09DBh] LCD Frame Control B Register 1	287
REG[09DCh] LCD Frame Control C Register 0	288	REG[09DDh] LCD Frame Control C Register 1	289
REG[09DEh] LCD Frame Control D Register 0	290	REG[09DFh] LCD Frame Control D Register 1	291
REG[09E0h] Camera1 Frame Buffer 0 Address Register 0	292	REG[09E1h] Camera1 Frame Buffer 0 Address Register 1	292
REG[09E2h] Camera1 Frame Buffer 0 Address Register 2	292	REG[09E3h] Camera1 Frame Buffer 0 Address Register 3	292
REG[09E4h] Camera1 Frame Buffer 1 Address Register 0	293	REG[09E5h] Camera1 Frame Buffer 1 Address Register 1	293
REG[09E6h] Camera1 Frame Buffer 1 Address Register 2	293	REG[09E7h] Camera1 Frame Buffer 1 Address Register 3	293
REG[09E8h] Camera2 Frame Buffer 0 Address Register 0	294	REG[09E9h] Camera2 Frame Buffer 0 Address Register 1	294
REG[09EAh] Camera2 Frame Buffer 0 Address Register 2	294	REG[09EBh] Camera2 Frame Buffer 0 Address Register 3	294
REG[09ECh] Camera2 Frame Buffer 1 Address Register 0	295	REG[09EDh] Camera2 Frame Buffer 1 Address Register 1	295
REG[09EEh] Camera2 Frame Buffer 1 Address Register 2	295	REG[09EFh] Camera2 Frame Buffer 1 Address Register 3	295
REG[09F0h] Camera1 Frame Buffer Width Register 0	296	REG[09F1h] Camera1 Frame Buffer Width Register 1	296
REG[09F2h] Camera1 Frame Buffer Height Register 0	296	REG[09F3h] Camera1 Frame Buffer Height Register 1	296
REG[09F4h] Camera1 Frame Buffer Virtual Width Register 0	296	REG[09F5h] Camera1 Frame Buffer Virtual Width Register 1	296
REG[09F6h] Camera1 Write Control Register	297		
REG[09F8h] Camera2 Frame Buffer Width Register 0	298	REG[09F9h] Camera2 Frame Buffer Width Register 1	298
REG[09FAh] Camera2 Frame Buffer Height Register 0	298	REG[09FBh] Camera2 Frame Buffer Height Register 1	298
REG[09FCh] Camera2 Frame Buffer Virtual Width Register 0	298	REG[09FDh] Camera2 Frame Buffer Virtual Width Register 1	298
REG[09FEh] Camera2 Write Control Register	299		

Table 10-3: Register Set (Continued)

Register	Page	Register	Page
Interrupt Configuration Registers			
REG[0A00h] Interrupt Status Register 0	300	REG[0A02h] Interrupt Status Register 1	302
REG[0A04h] Interrupt Status Register 2	303	REG[0A06h] Host Interrupt Enable Register 0	307
REG[0A08h] Host Interrupt Enable Register 1	308	REG[0A0Ah] Host Interrupt Enable Register 2	310
REG[0A0Ch] Host Interrupt Control Register	311	REG[0A0Eh] C33PE Device Interrupt Enable Register 0	312
REG[0A10h] C33PE Device Interrupt Enable Register 1	313	REG[0A12h] C33PE Device Interrupt Enable Register 2	315
REG[0A20h] C33PE Interrupt 0 Control Register 0	316	REG[0A21h] C33PE Interrupt 0 Control Register 1	316
REG[0A22h] C33PE Interrupt 1 Control Register 0	316	REG[0A23h] C33PE Interrupt 1 Control Register 1	317
REG[0A24h] C33PE Interrupt 2 Control Register 0	317	REG[0A25h] C33PE Interrupt 2 Control Register 1	317
REG[0A26h] C33PE Interrupt 3 Control Register 0	317	REG[0A27h] C33PE Interrupt 3 Control Register 1	317
REG[0A28h] C33PE Interrupt 4 Control Register 0	317	REG[0A29h] C33PE Interrupt 4 Control Register 1	318
REG[0A2Ah] C33PE Interrupt 5 Control Register 0	318	REG[0A2Bh] C33PE Interrupt 5 Control Register 1	318
REG[0A2Ch] C33PE Interrupt 6 Control Register 0	318	REG[0A2Dh] C33PE Interrupt 6 Control Register 1	318
REG[0A2Eh] C33PE Interrupt 7 Control Register 0	318	REG[0A2Fh] C33PE Interrupt 7 Control Register 1	319
REG[0A40h] C33PE Manual Interrupt Trigger Register	319	REG[0A42h] C33PE Interrupt Enable Register	319
REG[0A43h] C33PE NMI Interrupt Enable Register	320	REG[0A44h] C33PE Interrupt Status Register	320
REG[0A46h] C33 to Host Interrupt Trigger Register	320		
Timer Configuration Registers			
REG[0A80h] Timer Clock Configuration Register 0	321	REG[0A81h] Timer Clock Configuration Register 1	321
REG[0A84h] Timer Control Register	321		
REG[0A86h] Watchdog Timer Period Register 0	322	REG[0A87h] Watchdog Timer Period Register 1	322
REG[0A88h] Timer 0 Period Register 0	323	REG[0A89h] Timer 0 Period Register 1	323
REG[0A8Ah] Timer 1 Period Register	323		
REG[0A8Ch] Watchdog Timer Clear Register 0	324	REG[0A8Dh] Watchdog Timer Clear Register 1	324
SPI Flash Memory Interface Registers			
REG[0B00h] SPI Flash Read Data Register	325	REG[0B02h] SPI Flash Write Data Register	325
REG[0B03h] SPI Flash Data Control Register	325	REG[0B04h] SPI Flash Control Register	326
REG[0B06h] SPI Flash Status Register	327	REG[0B0Ah] SPI Flash Chip Select Control Register	328
Cache Control Register			
REG[0C00h] C33 Instruction Cache Control Register	329		
Camera Interface Registers			
REG[0D00h] Camera1 Enable Register	330	REG[0D02h] Camera1 Clock Configuration Register	330
REG[0D04h] Camera1 Signal Polarity Register	331	REG[0D06h] Camera1 Configuration Register 0	331
REG[0D07h] Camera1 Configuration Register 1	332	REG[0D08h] Camera1 Input Frame Control Register	333
REG[0D0Ah] Camera1 Input Horizontal Size Register 0	334	REG[0D0Bh] Camera1 Input Horizontal Size Register 1	334
REG[0D0Ch] Camera1 Input Vertical Size Register 0	335	REG[0D0Dh] Camera1 Input Vertical Size Register 1	335
REG[0D0Eh] Camera1 Status Register	335		
REG[0D10h] Camera1 Resizer X Start Position Register 0	337	REG[0D11h] Camera1 Resizer X Start Position Register 1	337
REG[0D12h] Camera1 Resizer Y Start Position Register 0	337	REG[0D13h] Camera1 Resizer Y Start Position Register 1	337
REG[0D14h] Camera1 Resizer X End Position Register 0	337	REG[0D15h] Camera1 Resizer X End Position Register 1	337
REG[0D16h] Camera1 Resizer Y End Position Register 0	338	REG[0D17h] Camera1 Resizer Y End Position Register 1	338
REG[0D18h] Camera1 Resizer Horizontal Scaling Rate Register	338	REG[0D19h] Camera1 Resizer Vertical Scaling Rate Register	338
REG[0D1Ah] Camera1 Resizer Scaling Control Register	339	REG[0D1Ch] is Reserved	339
REG[0D1Eh] Camera1 YRC Control Register 0	339	REG[0D1Fh] Camera1 YRC Control Register 1	340
REG[0D20h] Camera1 YRC U Fixed Data Register	341	REG[0D21h] Camera1 YRC V Fixed Data Register	341

Table 10-3: Register Set (Continued)

Register	Page	Register	Page
REG[0D22h] is Reserved	341		
REG[0D24h] Camera1 YRC X Size Register 0	342	REG[0D25h] Camera1 YRC X Size Register 1	342
REG[0D26h] Camera1 YRC Y Size Register 0	342	REG[0D27h] Camera1 YRC Y Size Register 1	342
REG[0D28h] is Reserved	343		
REG[0D40h] Camera2 Enable Register	345	REG[0D42h] Camera2 Clock Configuration Register	345
REG[0D44h] Camera2 Signal Polarity Register	346	REG[0D46h] Camera2 Configuration Register 0	346
REG[0D47h] Camera2 Configuration Register 1	347	REG[0D48h] Camera2 Input Frame Control Register	348
REG[0D4Ah] Camera2 Input Horizontal Size Register 0	349	REG[0D4Bh] Camera2 Input Horizontal Size Register 1	349
REG[0D4Ch] Camera2 Input Vertical Size Register 0	350	REG[0D4Dh] Camera2 Input Vertical Size Register 1	350
REG[0D4Eh] Camera2 Status Register 0	350		
REG[0D50h] Camera2 Resizer X Start Position Register 0	351	REG[0D51h] Camera2 Resizer X Start Position Register 1	351
REG[0D52h] Camera2 Resizer Y Start Position Register 0	352	REG[0D53h] Camera2 Resizer Y Start Position Register 1	352
REG[0D54h] Camera2 Resizer X End Position Register 0	352	REG[0D55h] Camera2 Resizer X End Position Register 1	352
REG[0D56h] Camera2 Resizer Y End Position Register 0	352	REG[0D57h] Camera2 Resizer Y End Position Register 1	352
REG[0D58h] Camera2 Resizer Horizontal Scaling Rate Register	353	REG[0D59h] Camera2 Resizer Vertical Scaling Rate Register	353
REG[0D5Ah] Camera2 Resizer Scaling Control Register	353		
REG[0D5Eh] Camera2 YRC Control Register 0	354	REG[0D5Fh] Camera2 YRC Control Register 1	355
REG[0D60h] Camera2 YRC U Fixed Data Register	356	REG[0D61h] Camera2 YRC V Fixed Data Register	356
REG[0D62h] is Reserved	356		
REG[0D64h] Camera2 YRC X Size Register 0	357	REG[0D65h] Camera2 YRC X Size Register 1	357
REG[0D66h] Camera2 YRC Y Size Register 0	357	REG[0D67h] Camera2 YRC Y Size Register 1	357
REG[0D68h] is Reserved	358		
DMA Controller Registers			
REG[3C00h] DMA Channel 0 Source Address Register 0	360	REG[3C01h] DMA Channel 0 Source Address Register 1	360
REG[3C02h] DMA Channel 0 Source Address Register 2	360	REG[3C03h] DMA Channel 0 Source Address Register 3	360
REG[3C04h] DMA Channel 0 Destination Address Register 0	361	REG[3C05h] DMA Channel 0 Destination Address Register 1	361
REG[3C06h] DMA Channel 0 Destination Address Register 2	361	REG[3C07h] DMA Channel 0 Destination Address Register 3	361
REG[3C08h] DMA Channel 0 Transfer Count Register 0	362	REG[3C09h] DMA Channel 0 Transfer Count Register 1	362
REG[3C0Ah] DMA Channel 0 Transfer Count Register 2	362		
REG[3C0Ch] DMA Channel 0 Control Register 0	362	REG[3C0Dh] DMA Channel 0 Control Register 1	364
REG[3C10h] DMA Channel 1 Source Address Register 0	365	REG[3C11h] DMA Channel 1 Source Address Register 1	365
REG[3C12h] DMA Channel 1 Source Address Register 2	365	REG[3C13h] DMA Channel 1 Source Address Register 3	365
REG[3C14h] DMA Channel 1 Destination Address Register 0	366	REG[3C15h] DMA Channel 1 Destination Address Register 1	366
REG[3C16h] DMA Channel 1 Destination Address Register 2	366	REG[3C17h] DMA Channel 1 Destination Address Register 3	366
REG[3C18h] DMA Channel 1 Transfer Count Register 0	367	REG[3C19h] DMA Channel 1 Transfer Count Register 1	367
REG[3C1Ah] DMA Channel 1 Transfer Count Register 2	367		
REG[3C1Ch] DMA Channel 1 Control Register 0	367	REG[3C1Dh] DMA Channel 1 Control Register 1	369
REG[3C20h] DMA Status Register	370	REG[3C22h] DMA Start Register	370
SDRAM Controller Configuration Registers			
REG[3C40h] SDRAM Control Register	372	REG[3C42h] SDRAM Refresh Period Register 0	373
REG[3C43h] SDRAM Refresh Period Register 1	373	REG[3C44h] SDRAM Clock Control Register	374

Table 10-3: Register Set (Continued)

Table 10-3: Register Set (Continued)

Register	Page	Register	Page
REG[4065h] CH1IN FIFO Empty Status Register	407	REG[4070h] LCD2 Display Mode Register 0	407
REG[4072h] LCD2 Display Mode Register 1	409	REG[4073h] LCD2 Display Mode Register 2	410
REG[4074h] CH2IN FIFO Threshold Register	410	REG[4075h] CH2IN FIFO Empty Status Register	411
REG[4076h] OSDIN FIFO Threshold Register	411	REG[4077h] OSDIN FIFO Empty Status Register	411
REG[4078h] through REG[407Fh] are Reserved	411		
REG[4080h] LCD1 Bias/Gain Control Register	411	REG[4082h] LCD1 Bias Red Register 0	412
REG[4083h] LCD1 Bias Red Register 1	412	REG[4084h] LCD1 Bias Green Register 0	412
REG[4085h] LCD1 Bias Green Register 1	412	REG[4086h] LCD1 Bias Blue Register 0	412
REG[4087h] LCD1 Bias Blue Register 1	413	REG[4088h] LCD1 Gain Red Register	414
REG[408Ah] LCD1 Gain Green Register	414	REG[408Ch] LCD1 Gain Blue Register	414
REG[4090h] LCD2 Bias/Gain Control Register	414	REG[4092h] LCD2 Bias Red Register 0	415
REG[4093h] LCD2 Bias RED Register 1	415	REG[4094h] LCD2 Bias Green Register 0	415
REG[4095h] LCD2 Bias Green Register 1	415	REG[4096h] LCD2 Bias Blue Register 0	415
REG[4097h] LCD2 Bias Blue Register 1	415	REG[4098h] LCD2 Gain Red Register	416
REG[409Ah] LCD2 Gain Green Register	416	REG[409Ch] LCD2 Gain Blue Register	416
REG[40A0h] LCD2 Gamma LUT Data Port	416		
REG[40A2h] LCD2 Gamma LUT Configuration Register 0	416	REG[40A3h] LCD2 Gamma LUT Configuration Register 1	418
REG[40B0h] LCD1 Power Save Register	418	REG[40B1h] LCD2 Power Save Register	418
Sprite Registers			
REG[5000h] Sprite Control Register	419	REG[5001h] Sprite Software Reset Register	420
REG[5002h] Sprite SDRAM Registers Busy Register	421	REG[5003h] Sprite Engine Status Register	421
REG[5004h] Sprite Frame Trigger Control Register	422	REG[5006h] Sprite Interrupt Control Register	422
REG[5008h] Sprite Interrupt Status Register	422		
REG[5020h] Sprite Frame Buffer 0 Start Address Register 0	423	REG[5021h] Sprite Frame Buffer 0 Start Address Register 1	423
REG[5022h] Sprite Frame Buffer 0 Start Address Register 2	423	REG[5023h] Sprite Frame Buffer 0 Start Address Register 3	423
REG[5024h] Sprite Frame Buffer 1 Start Address Register 0	424	REG[5025h] Sprite Frame Buffer 1 Start Address Register 1	424
REG[5026h] Sprite Frame Buffer 1 Start Address Register 2	424	REG[5027h] Sprite Frame Buffer 1 Start Address Register 3	424
REG[5028h] Sprite SDRAM Based Registers Start Address Register 0	425	REG[5029h] Sprite SDRAM Based Registers Start Address Register 1	425
REG[502Ah] Sprite SDRAM Based Registers Start Address Register 2	425	REG[502Bh] Sprite SDRAM Based Registers Start Address Register 3	425
Sprite Memory Based Registers			
SDRAM[**000h] Sprite #n General Control Register 0	427	SDRAM[**001h] Sprite #n General Control Register 1	427
SDRAM[**004h] Sprite #n Image Start Address Register 0	428	SDRAM[**005h] Sprite #n Image Start Address Register 1	428
SDRAM[**006h] Sprite #n Image Start Address Register 2	428	SDRAM[**007h] Sprite #n Image Start Address Register 3	428
SDRAM[**008h] Sprite #n Rotated Image Start Address Register 0	429	SDRAM[**009h] Sprite #n Rotated Image Start Address Register 1	429
SDRAM[**00Ah] Sprite #n Rotated Image Start Address Register 2	429	SDRAM[**00Bh] Sprite #n Rotated Image Start Address Register 3	429
SDRAM[**00Ch] Sprite #n X Position Register 0	430	SDRAM[**00Dh] Sprite #n X Position Register 1	430
SDRAM[**00Eh] Sprite #n Y Position Register 0	431	SDRAM[**00Fh] Sprite #n Y Position Register 1	431
SDRAM[**010h] Sprite #n Frame Width Register 0	432	SDRAM[**011h] Sprite #n Frame Width Register 1	432
SDRAM[**012h] Sprite #n Frame Height Register 0	432	SDRAM[**013h] Sprite #n Frame Height Register 1	432
SDRAM[**014h] Sprite #n Reference Point X Offset Register 0	433	SDRAM[**015h] Sprite #n Reference Point X Offset Register 1	433
SDRAM[**016h] Sprite #n Reference Point Y Offset Register 0	434	SDRAM[**017h] Sprite #n Reference Point Y Offset Register 1	434
SDRAM[**018h] Sprite #n Transparency Color / Texture Alpha Register 0	435	SDRAM[**019h] Sprite #n Transparency Color / Texture Alpha Register 1	435
SDRAM[**01Ah] Sprite #n Color Format Register	436		

10.3 Register Restrictions

All reserved bits must be set to 0b unless otherwise specified. Writing a value to a reserved bit may produce undefined results. Bits marked as n/a have no hardware effect.

10.4 Register Descriptions

10.4.1 System Control Registers

REG[0000h] Product ID Register 0 Default = 00h							Read Only
Reserved							
7	6	5	4	3	2	1	0

bits 7-0 Reserved
These bits always return 0000_0000b (00h).

REG[0001h] Product ID Register 1 Default = 00h							Read Only
Revision Code bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 Revision Code bits [7:0]
These bits indicate the revision code.
The revision code for the S2D13515 is 00h.

REG[0002h] Product ID Register 2 Default = 45h							Read Only
Product Code bits 7-0							
7	6	5	4	3	2	1	0

REG[0003h] Product ID Register 3 Default = 00h							Read Only
Product Code bits 15-8							
7	6	5	4	3	2	1	0

REG[0003h] bits 7-0
REG[0002h] bits 7-0 Product Code bits [15:0]
These bits indicate the product code.
The product code is 0045h.

REG[000Ch] through REG[000Fh] are Reserved

These registers are Reserved and should not be written.

REG[0010h] C33 TTBR Remap Address Register 0 Default = 00h							Read/Write
n/a							
7	6	5	4	3	2	1	0

REG[0011h] C33 TTBR Remap Address Register 1 Default = 00h							Read/Write
C33 TTBR Remap Address bits 15-8						n/a	
7	6	5	4	3	2	1	0

REG[0012h] C33 TTBR Remap Address Register 2 Default = 00h							Read/Write
C33 TTBR Remap Address bits 23-16							
7	6	5	4	3	2	1	0

REG[0013h] C33 TTBR Remap Address Register 3 Default = 00h							Read/Write
C33 TTBR Remap Address bits 31-24							
7	6	5	4	3	2	1	0

REG[0013h] bits 7-0
REG[0012h] bits 7-0
REG[0011h] bits 7-0
REG[0010h] bits 7-0 C33 TTBR Remap Address bits [31:10]

These bits specify the address (on a 1K boundary) where the C33 TTBR (exception vector table) will be remapped. REG[0011h] bits 1-0 and REG[0010h] bits 7-0 are always 0. These registers are read by the boot monitor code at boot-up. If the value of these registers is not 0, the boot monitor reprograms the TTBR address according to the address specified.

Note

SRAM region 0400_0200h ~ 0400_0D28h is cleared by the ROM monitor and must not be used by the TTBR function (00A0_0200h ~ 00A0_0D28h from the C33 memory map).

REG[001Ch] C33 Control Register Default =C0h							Read/Write
Reserved	C33 Enable	n/a					
7	6	5	4	3	2	1	0

bit 7 Reserved

This bit **MUST** be set to 0b.

bit 6 C33 Enable

This bit controls the C33. The C33 cannot be enabled when power save mode is enabled (REG[003Ch] bit 0 = 1b).
When this bit = 0b, the C33 is disabled.
When this bit = 1b, the C33 is enabled. (default)

Note

1. The C33 should be reset before entering power save mode (REG[003Ch] bit 0) and reset disabled if necessary after exiting power save mode. 2.For minimum current consumption of the C33 when not used, REG[001Dh] bit 0 and REG[001Ch] bits 7 and 6 should be set to 0b.

REG[001Dh] C33 Software Reset Register Default =00h							Read/Write
n/a							C33 Software Reset
7	6	5	4	3	2	1	0

bit 0 C33 Software Reset
This bit is used to perform a software reset of the C33. This is done by writing a 1b then a 0b to this bit.
When this bit = 0b, the C33 is released from reset. (default)
When this bit = 1b, the C33 is held in reset.

Note

For minimum current consumption of the C33 when not used, REG[001Dh] bit 0 and REG[001Ch] bits 7 and 6 should be set to 0b.

REG[001Eh] C33 Status Register Default = 00h							Read Only
n/a						C33 Sleep Status	C33 Halt Status
7	6	5	4	3	2	1	0

bit 1 C33 Sleep Status (Read Only)
This bit indicates the status of the C33 internal sleep bit.
When this bit = 0b, the C33 is not in a sleep state.
When this bit = 1b, the C33 is in a sleep state.

bit 0 C33 Halt Status (Read Only)
This bit indicates the status of the C33 internal halt bit.
When this bit = 0b, the C33 is not in a halt state.
When this bit = 1b, the C33 is in a halt state.

REG[0020h] PLL1 Configuration Register 0 Default = 11h							Read/Write
n/a		PLL1V[1:0]		PLL1N[3:0]			
7	6	5	4	3	2	1	0

bits 5-4 PLL1V[1:0]
These bits are used to configure the VCO frequency which must be set between 100MHz and 400MHz. These bits should be set using the following formula.

fVCO = fPLL1OUT x VV

Where:
fVCO is the frequency of VCO, in MHz
fPLL1OUT is the desired PLL1 output frequency, in MHz (see N Multiplier bits)
VV is the value based on the V Divider bits as follows.

Table 10-4: VV Value

REG[0020h] bits 5-4	VV Value
00b	Reserved
01b	2
10b	4
11b	8

Note
Normally VV is set to 2. When fPLL1OUT is lower than 50MHz, stabilize VCO by setting VV = 4 or 8. Also, the PLL1 VC bits (REG[0021h] bits 3-0) must be set according to the resulting fVCO. The frequency of VCO (fVCO) must always be within 100MHz ~ 400MHz.

bits 3-0 PLL1N[3:0]
These bits are used to determine the output frequency of PLL1 according to the following formula.

fPLL1OUT = fPLL1REFCLK x NN

Where:
fPLL1OUT is the desired PLL1 output frequency, in MHz
fPLL1REFCLK is the PLL1 reference clock input frequency, in MHz
NN is the N Multiplier value + 1

REG[0021h] PLL1 Configuration Register 1 Default = 83h							Read/Write
PLL1RS[3:0]				PLL1VC[3:0]			
7	6	5	4	3	2	1	0

bits 7-4 PLL1RS[3:0]
These bits are used to configure the Low Pass Filter (LPF) resistance and should be set based on the frequency of the PLL1 reference clock.

Table 10-5: PLL1 RS Configuration

REG[0021h] bits 7-4	PLL1 Reference Clock Frequency
0000b ~ 0111b	Reserved
1000b	20MHz ≤ fPLL1REFCLK ≤ 150MHz
1001b	Reserved
1010b	5MHz ≤ fPLL1REFCLK ≤ 20MHz
1011b ~ 1111b	Reserved

bits 3-0 PLL1VC[3:0]
These bits set the analog adjustment pins for PLL1 and should be set according to the VCO frequency.

Table 10-6: PLL1 VC Configuration

REG[0021h] bits 3-0	PLL1 VCO Frequency
0000b	Reserved
0001b	100MHz ≤ fVCO ≤ 120MHz
0010b	120MHz < fVCO ≤ 160MHz
0011b	160MHz < fVCO ≤ 200MHz
0100b	200MHz < fVCO ≤ 240MHz
0101b	240MHz < fVCO ≤ 280MHz
0110b	280MHz < fVCO ≤ 320MHz
0111b	320MHz < fVCO ≤ 360MHz
1000b	360MHz < fVCO ≤ 400MHz
1001b ~ 1111b	Reserved

REG[0022h] PLL1 Configuration Register 2 Default = 40h							Read/Write
PLL1 Configuration 2 bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 PLL1 Configuration 2 bits [7:0]
These bits are used to configure PLL1 and should be set to the recommended value of 40h.

REG[0024h] PLL1 Control Register Default = 00h							Read/Write
n/a							PLL1 Enable
7	6	5	4	3	2	1	0

bit 0 PLL1 Enable
This bit controls PLL1. PLL1 must be disabled before changing the PLL1 Configuration registers, REG[0020h] ~ REG[0022h].
When this bit = 0b, PLL1 is disabled. (default)
When this bit = 1b, PLL1 is enabled.

REG[0028h] PLL2 Configuration Register 0 Default = 11h							Read/Write
n/a		PLL2V[1:0]		PLL2N[3:0]			
7	6	5	4	3	2	1	0

bits 5-4 PLL2V[1:0]
These bits are used to configure the VCO frequency which must be set between 100MHz and 400MHz. These bits should be set using the following formula.

fVCO = fPLL1OUT x VV

Where:
fVCO is the frequency of VCO, in MHz
fPLL2OUT is the desired PLL2 output frequency, in MHz (see N Multiplier bits)
VV is the value based on the V Divider bits as follows.

Table 10-7: VV Value

REG[0028h] bits 5-4	VV Value
00b	Reserved
01b	2
10b	4
11b	8

Note
Normally VV is set to 2. When fPLL2OUT is lower than 50MHz, stabilize VCO by setting VV = 4 or 8. Also, the PLL2 VC bits (REG[0029h] bits 3-0) must be set according to the resulting fVCO. The frequency of VCO (fVCO) must always be within 100MHz ~ 400MHz.

bits 3-0 PLL2N[3:0]
These bits are used to determine the output frequency of PLL2 according to the following formula.

fPLL2OUT = fPLL2REFCLK x NN

Where:
fPLL2OUT is the desired PLL2 output frequency, in MHz
fPLL2REFCLK is the PLL2 reference clock input frequency, in MHz
NN is the N Multiplier value + 1

REG[0029h] PLL2 Configuration Register 1 Default = 83h							Read/Write
PLL2RS[3:0]				PLL2VC[3:0]			
7	6	5	4	3	2	1	0

bits 7-4 PLL2RS[3:0]
These bits are used to configure the Low Pass Filter (LPF) resistance and should be set based on the frequency of the PLL2 reference clock.

Table 10-8: PLL2 RS Configuration

REG[0029h] bits 7-4	PLL2 Reference Clock Frequency
0000b ~ 0111b	Reserved
1000b	20MHz ≤ fPLL2REFCLK ≤ 150MHz
1001b	Reserved
1010b	5MHz ≤ fPLL2REFCLK ≤ 20MHz
1011b ~ 1111b	Reserved

bits 3-0 PLL2VC[3:0]
These bits set the analog adjustment pins for PLL2 and should be set according to the VCO frequency.

Table 10-9: PLL2 VC Configuration

REG[0029h] bits 3-0	PLL2 VCO Frequency
0000b	Reserved
0001b	100MHz ≤ fVCO ≤ 120MHz
0010b	120MHz < fVCO ≤ 160MHz
0011b	160MHz < fVCO ≤ 200MHz
0100b	200MHz < fVCO ≤ 240MHz
0101b	240MHz < fVCO ≤ 280MHz
0110b	280MHz < fVCO ≤ 320MHz
0111b	320MHz < fVCO ≤ 360MHz
1000b	360MHz < fVCO ≤ 400MHz
1001b ~ 1111b	Reserved

REG[002Ah] PLL2 Configuration Register 2 Default = 40h							Read/Write
PLL2 Configuration 2 bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 PLL2 Configuration 2 bits [7:0]
These bits are used to configure PLL2 and should be set to the recommended value of 40h.

REG[002Ch] PLL2 Control Register Default = 00h							Read/Write
n/a							PLL2 Enable
7	6	5	4	3	2	1	0

bit 0 PLL2 Enable
This bit controls PLL2. PLL2 must be disabled before changing the PLL2 Configuration registers, REG[0028h] ~ REG[002Ah].
When this bit = 0b, PLL2 is disabled. (default)
When this bit = 1b, PLL2 is enabled.

REG[0030h] LCD1PCLK Configuration Register Default = 05h							Read/Write
n/a			LCD1PCLK Divide Select bits 4-0				
7	6	5	4	3	2	1	0

bits 4-0 LCD1PCLK Divide Select bits [4:0]
These bits specify the divide ratio for the LCD1 pixel clock (LCD1PCLK). LCD1PCLK is derived from LCDCLK.

Table 10-10 LCD1PCLK Divide Ratio Selection

REG[0030h] bits 4-0	LCD1PCLK Divide Ratio	REG[0030h] bits 4-0	LCD1PCLK Divide Ratio
00000b	1:1	10000b	1:32
00001b	1:2	10001b	1:34
00010b	1:4	10010b	1:36
00011b	1:6	10011b	1:38
00100b	1:8	10100b	1:40
00101b	1:10	10101b	1:42
00110b	1:12	10110b	1:44
00111b	1:14	10111b	1:46
01000b	1:16	11000b	1:48
01001b	1:18	11001b	1:50
01010b	1:20	11010b	1:52
01011b	1:22	11011b	1:54
01100b	1:24	11100b	1:56
01101b	1:26	11101b	1:58
01110b	1:28	11110b	1:60
01111b	1:30	11111b	1:62

REG[0031h] LCD2PCLK Configuration Register Default = 02h							Read/Write
n/a			LCD2PCLK Divide Select bits 4-0				
7	6	5	4	3	2	1	0

bits 4-0 LCD2PCLK Divide Select bits [4:0]
These bits specify the divide ratio for the LCD2 pixel clock (LCD2PCLK). LCD2PCLK is derived from LCDCLK.

Table 10-11 LCD2PCLK Divide Ratio Selection

REG[0031h] bits 4-0	LCD2PCLK Divide Ratio	REG[0031h] bits 4-0	LCD2PCLK Divide Ratio
00000b	1:1	10000b	1:32
00001b	1:2	10001b	1:34
00010b	1:4	10010b	1:36
00011b	1:6	10011b	1:38
00100b	1:8	10100b	1:40
00101b	1:10	10101b	1:42
00110b	1:12	10110b	1:44
00111b	1:14	10111b	1:46
01000b	1:16	11000b	1:48
01001b	1:18	11001b	1:50
01010b	1:20	11010b	1:52
01011b	1:22	11011b	1:54
01100b	1:24	11100b	1:56
01101b	1:26	11101b	1:58
01110b	1:28	11110b	1:60
01111b	1:30	11111b	1:62

REG[0032h] LCD1SCLK Configuration Register Default = 05h							Read/Write
n/a			LCD1SCLK Divide Select bits 4-0				
7	6	5	4	3	2	1	0

bits 4-0 LCD1SCLK Divide Select bits [4:0]
These bits specify the divide ratio for the LCD1 serial clock (LCD1SCLK). LCD1SCLK is derived from LCDCLK.

Table 10-12 LCD1SCLK Divide Ratio Selection

REG[0032h] bits 4-0	LCD1SCLK Divide Ratio	REG[0032h] bits 4-0	LCD1SCLK Divide Ratio
00000b	Reserved	10000b	1:32
00001b	1:2	10001b	1:34
00010b	1:4	10010b	1:36
00011b	1:6	10011b	1:38
00100b	1:8	10100b	1:40
00101b	1:10	10101b	1:42
00110b	1:12	10110b	1:44
00111b	1:14	10111b	1:46
01000b	1:16	11000b	1:48
01001b	1:18	11001b	1:50
01010b	1:20	11010b	1:52
01011b	1:22	11011b	1:54
01100b	1:24	11100b	1:56
01101b	1:26	11101b	1:58
01110b	1:28	11110b	1:60
01111b	1:30	11111b	1:62

REG[0033h] LCD2SCLK Configuration Register							
Default = 05h							Read/Write
n/a			LCD2SCLK Divide Select bits 4-0				
7	6	5	4	3	2	1	0

bits 4-0 LCD2SCLK Divide Select bits [4:0]
These bits specify the divide ratio for the LCD2 serial clock (LCD2SCLK). LCD2SCLK is derived from LCDCLK.

Table 10-13 LCD2SCLK Divide Ratio Selection

REG[0033h] bits 4-0	LCD2SCLK Divide Ratio	REG[0033h] bits 4-0	LCD2SCLK Divide Ratio
00000b	Reserved	10000b	1:32
00001b	1:2	10001b	1:34
00010b	1:4	10010b	1:36
00011b	1:6	10011b	1:38
00100b	1:8	10100b	1:40
00101b	1:10	10101b	1:42
00110b	1:12	10110b	1:44
00111b	1:14	10111b	1:46
01000b	1:16	11000b	1:48
01001b	1:18	11001b	1:50
01010b	1:20	11010b	1:52
01011b	1:22	11011b	1:54
01100b	1:24	11100b	1:56
01101b	1:26	11101b	1:58
01110b	1:28	11110b	1:60
01111b	1:30	11111b	1:62

REG[0034h] PWMSRCCLK Configuration Register 0							
Default = 00h							Read/Write
PWMSRCCLK Divide Select bits 7-0							
7	6	5	4	3	2	1	0

REG[0035h] PWMSRCCLK Configuration Register 1							
Default = 00h							Read/Write
n/a				PWMSRCCLK Divide Select bits 11-8			
7	6	5	4	3	2	1	0

REG[0035h] bits 3-0
REG[0034h] bits 7-0 PWMSRCCLK Divide Select bits [11:0]
These bits specify the divide ratio for the PWM source clock (PWMSRCCLK). PWMSRCCLK is derived from the system clock. The divide ratio is calculated using the following formula.

PWMSRCCLK Divide Ratio = 1:(REG[0035h] bits 3-0, REG[0034h] bits 7-0 + 1)

REG[003Ch] Power Save Configuration Register Default = 00h							Read/Write
n/a					LCD Clock Source Select	SDRAM Clock Source Select	Power Save Mode Enable
7	6	5	4	3	2	1	0

bit 2 LCD Clock Source Select
This bit selects whether PLL2 is the source for the LCD clocks. For details on the clock structure, see Section Chapter 9, "Clocks" on page 131.
When this bit = 0b, the LCD clocks source is PLL2IN which can be derived from either CLKI or OSCI as determined by the Input Clock 2 Source Select bit, REG[003Eh] bit 7.
When this bit = 1b, the LCD clocks source is PLL2.

bit 1 SDRAM Clock Source Select
This bit selects whether PLL1 is the source for the SDRAM Clock. For details on the clock structure, see Section Chapter 9, "Clocks" on page 131.
When this bit = 0b, the SDRAM clock source is PLL1IN which can be derived from either CLKI or OSCI as determined by the CNF0 pin (see Section 5.4, "Configuration Pins" on page 33).
When this bit = 1b, the SDRAM clock source is PLL1.

bit 0 Power Save Mode Enable
The S1D13515/S2D13515 features dynamic internal clocking which enables internal clocks only when required. If all internal clocks must be stopped, this bit may be used to override dynamic clocking and stop all internal clocks.
When this bit = 0b, all internal clocks are dynamically controlled as required.
When this bit = 1b, all internal clocks are stopped.

Note

1. When Power Save Mode is enabled (REG[003Ch] bit 0 = 1b), only asynchronous registers may be accessed. Synchronous registers must not be accessed. To confirm whether registers are asynchronous or synchronous, refer to Section 10.1, "Register Mapping" on page 132.
2. To achieve the lowest power consumption, PLL1 and PLL2 should be disabled in REG[0024h] bit 0 and REG[002Ch] bit 0, respectively, and REG[003Ch] bits 2-0 should be set to 111b.
3. Before entering power save mode, the I2S Audio Interface must be disabled in REG[0104h] bit 0 and REG[010Fh] bit 7.
4. Before entering power save mode, the C33 must be placed in HALT or SLEEP mode (through instruction code), or placed in reset (REG[001Dh] bit 0).
   To maintain DRAM contents while in powersave mode, place the DRAM controller in self-refresh mode in REG[3C44h] bit 6 before entering power save mode.
5. After exiting powersave mode, if self refresh mode is enabled, exit self refresh mode in REG[3C44h] bit 6 before enabling any accesses to DRAM.
6. After exiting power save mode, the DRAM controller must be re-initialized by writing a 1b to REG[3C40h] bit 0 and waiting for the bit to return a 0b before enabling any accesses to DRAM.
7. After exiting power save mode, Note 5 or 6 must be met before the C33 can safely exit HALT or SLEEP mode, or be released from reset (REG[001Dh] bit 0).

REG[003Dh] IO Drive Select Register Default = 1Fh							Read/Write
Reserved	Reserved	Miscellaneous IO Drive Select	SDRAM IO Drive Select	Camera IO Drive Select	Panel2 IO Drive Select	Panel1 IO Drive Select	Host IO Drive Select
7	6	5	4	3	2	1	0

bit 7 Reserved
This bit must be set to 0b.

bit 6 Reserved
This bit must be set to 0b.

bit 5 Miscellaneous IO Drive Select
This bit determines the drive level, in mA, for the Miscellaneous IO interface output pins.
When this bit = 0b, the Miscellaneous IO drive level is set to 2mA.
When this bit = 1b, the Miscellaneous IO drive level is set to 4mA (default).

bit 4 SDRAM IO Drive Select
This bit determines the drive level, in mA, for the Miscellaneous IO interface output pins.
When this bit = 0b, the SDRAM IO drive level is set to 2mA.
When this bit = 1b, the SDRAM IO drive level is set to 4mA (default).

bit 3 Camera IO Drive Select
This bit determines the drive level, in mA, for the Camera IO interface output pins (CM1CLKOUT, SCL, and SDA).
When this bit = 0b, the Camera IO drive level is set to 2mA.
When this bit = 1b, the Camera IO drive level is set to 4mA (default).

bit 2 Panel2 IO Drive Select
This bit determines the drive level, in mA, for the Panel2 IO interface output pins.
When this bit = 0b, the Panel2 IO drive level is set to 2mA.
When this bit = 1b, the Panel2 IO drive level is set to 4mA (default).

bit 1 Panel1 IO Drive Select
This bit determines the drive level, in mA, for the Panel1 IO interface output pins.
When this bit = 0b, the Panel1 IO drive level is set to 2mA.
When this bit = 1b, the Panel1 IO drive level is set to 4mA (default).

bit 0 Host IO Drive Select
This bit determines the drive level, in mA, for the Host IO interface output pins.
When this bit = 0b, the Host IO drive level is set to 2mA.
When this bit = 1b, the Host IO drive level is set to 4mA (default).

REG[003Eh] Input Clock Control Register Default = 0Xh							Read/Write
Input Clock 2 Source Select	PLL2 Input Divide Select bits 1-0		PLL2 Input Divide Enable	Input Clock 1 Source (RO)	PLL1 Input Divide Select bits 1-0		PLL1 Input Divide Enable
7	6	5	4	3	2	1	0

bit 7 Input Clock 2 Source Select
This bit selects whether CLKI or OSCI is the source for Input Clock 2 (INCLK2). For details on the clock structure, see Section Chapter 9, “Clocks” on page 131.
When this bit = 0b, the Input Clock 2 source is CLKI.
When this bit = 1b, the Input Clock 2 source is OSCI.

bits 6-5 PLL2 Input Divide Select bits [1:0]
If the PLL2 Input Divide Enable bit is set to 1b (REG[003Eh] bit 4 = 1b), these bits select the divide ratio applied to Input Clock 2 (INCLK2) before it goes to PLL2. If the PLL2 Input Divide Enable bit is set to 0b (REG[003Eh] bit 4 = 0b), Input Clock 2 is not divided (1:1).

Table 10-14: PLL2 Input Divide Ratio Selection

REG[003Eh] bits 6-5	PLL2 Input Divide Ratio
00b	2:1
01b	4:1
10b	6:1
11b	8:1

Note
If the system is already operating with a divided clock and the divide ratio needs to be switched to a different ratio, the following sequence must be used.
1. Disable the PLL2 input divider (REG[003Eh] bit 4 = 0b)
2. Change the PLL2 input divide ratio (REG[003Eh] bits 6-5)
3. Enable the PLL2 input divider (REG[003Eh] bit 4 = 1b)

bit 4 PLL2 Input Divide Enable
This bit determines whether Input Clock 2 (INCLK2) which is used to derive the PLL2 input clock (PLL2IN) is divided or not. For details on the clock structure, see Section Chapter 9, “Clocks” on page 131.
When this bit = 0b, Input Clock 2 is not divided (1:1).
When this bit = 1b, Input Clock 2 is divided according to the setting of the PLL2 Input Divide Select bits, REG[003Eh] bits 6-5.

bit 3 Input Clock 1 Source (Read Only)
This bit indicates the Input Clock 1 (INCLK1) source which is controlled by the state of the CNF0 pin.
When this bit = 0b, the Input Clock 1 source is CLKI.
When this bit = 1b, the Input Clock 1 source is OSCI.

bits 2-1 PLL1 Input Divide Select bits [1:0]
If the PLL1 Input Divide Enable bit is set to 1b (REG[003Eh] bit 0 = 1b), these bits select the divide ratio applied to Input Clock 1 (INCLK1) before it goes to PLL1. If the PLL1 Input Divide Enable bit is set to 0b (REG[003Eh] bit 0 = 0b), Input Clock 1 is not divided (1:1).

Table 10-15: PLL1 Input Divide Ratio Selection

REG[003Eh] bits 2-1	PLL1 Input Divide Ratio
00b	2:1
01b	4:1
10b	6:1
11b	8:1

Note
If the system is already operating with a divided clock and the divide ratio needs to be switched to a different ratio, the following sequence must be used.
1. Disable the PLL1 input divider (REG[003Eh] bit 0 = 0b)
2. Change the PLL1 input divide ratio (REG[003Eh] bits 2-1)
3. Enable the PLL1 input divider (REG[003Eh] bit 0 = 1b)

bit 0 PLL1 Input Divide Enable
This bit determines whether Input Clock 1 (INCLK1) which is used to derive the PLL1 input clock (PLL1IN) is divided or not. For details on the clock structure, see Section Chapter 9, “Clocks” on page 131.
When this bit = 0b, Input Clock 1 is not divided (1:1).
When this bit = 1b, Input Clock 1 is divided according to the setting of the PLL1 Input Divide Select bits, REG[003Eh] bits 2-1.

REG[0040h] through REG[0041h] are Reserved

These registers are Reserved and should not be written.

REG[0060h] Host SPI Clock Configuration Register Default = 00h							Read/Write
n/a				SPI Clock Divide Select bits 3-0			
7	6	5	4	3	2	1	0

bits 3-0 SPI Clock Divide Select bits [3:0]
These bits specify the divide ratio for the clock used for the Host SPI interface. The clock source for this divider is the system clock. This setting is used only when the SPI clock is generated from the system clock (REG[0061h] bit 0 = 1b).

Table 10-16: SPI Clock Divide Ratio Selection

REG[0060h] bits 3-0	SPI Clock Divide Ratio	REG[0060h] bits 3-0	SPI Clock Divide Ratio
0000b	1:1	1000b	9:1
0001b	2:1	1001b	10:1
0010b	3:1	1010b	11:1
0011b	4:1	1011b	12:1
0100b	5:1	1100b	13:1
0101b	6:1	1101b	14:1
0110b	7:1	1110b	15:1
0111b	8:1	1111b	16:1

Note
SPI Clock = System Clock frequency / Divide Ratio > HSCK frequency.

REG[0061h] Host SPI Enable Register Default = 00h or 10h if SPI Enabled							Read/Write
n/a			SPICLKEN Pin Status (RO)	n/a	SPI Clock Source Select	n/a	SPI Clock Enable
7	6	5	4	3	2	1	0

bit 4 SPICLKEN Pin Status (Read Only)
This bit indicates the status of the SPICLKEN (AB5) pin.
When this bit = 0b, the SPICLKEN (AB5) pin is low.
When this bit = 1b, the SPICLKEN (AB5) pin is high.

bit 2 SPI Clock Source Select
When the host is configured for SPI (see Section 5.4, “Configuration Pins” on page 33), the SPICLKSEL input pin (pin AB5) determines how the source for the Host SPI clock is selected. For details on the clock structure, see Section Chapter 9, “Clocks” on page 131.

When SPICLKSEL is 0, this bit is used to select the source for the Host SPI clock between Input Clock 1 (INCLK1) and the system clock (SYSCLK) as follows.
When this bit = 0b, the source for the Host SPI clock is the system clock. It can be further divided using the SPI Clock Divide Select bits (REG[0060h] bits 3-0) and can be controlled by the SPI Clock Enable bit (REG[0061h] bit 0).
When this bit = 1b, the source for the Host SPI clock is Input Clock 1 (INCLK1). It cannot be further divided and is not controlled by the SPI Clock Enable bit.

When SPICLKSEL is 1, this bit is ignored and the source for the Host SPI clock is Input Clock 1 (INCLK1).

bit 0 SPI Clock Enable
This bit enables/disables the Host SPI clock when the clock source is the divided down system clock.
When this bit = 0b, the Host SPI clock is disabled.
When this bit = 1b, the Host SPI clock is enabled.

REG[0062h] Host I2C Clock Configuration Register							
Default = 00h							Read/Write
n/a				I2C Clock Divide Select bits 4-0			
7	6	5	4	3	2	1	0

bits 3-0 I2C Clock Divide Select bits [3:0]
These bits specify the divide ratio for the clock used for the Host I2C interface. The clock source for this divider is the system clock. This setting is used only when I2C clock is generated from the system clock (REG[0063h] bit 0 = 1b).

Table 10-17: I2C Clock Divide Ratio Selection

REG[0062h] bits 3-0	I2C Clock Divide Ratio	REG[0062h] bits 3-0	I2C Clock Divide Ratio
0000b	1:1	1000b	9:1
0001b	2:1	1001b	10:1
0010b	3:1	1010b	11:1
0011b	4:1	1011b	12:1
0100b	5:1	1100b	13:1
0101b	6:1	1101b	14:1
0110b	7:1	1110b	15:1
0111b	8:1	1111b	16:1

Note
For fast mode (400kbps)
I2C Clock = System Clock frequency / Divide Ratio > 24MHz frequency.
For standard mode (100kbps)
I2C Clock = System Clock frequency / Divide Ratio > 5.4MHz frequency.

REG[0063h] Host I2C Enable Register Default = 00h							Read/Write
n/a			I2CCLKEN Pin Status (RO)	n/a	I2C Clock Source Select	n/a	I2C Clock Enable
7	6	5	4	3	2	1	0

bit 4 I2CCLKEN Pin Status (Read Only)
This bit indicates the status of the I2CCLKEN (AB5) pin.
When this bit = 0b, the I2CCLKEN (AB5) pin is low.
When this bit = 1b, the I2CCLKEN (AB5) pin is high.

bit 2 I2C Clock Source Select
When the host is configured for I2C (see Section 5.4, “Configuration Pins” on page 33), the I2CCLKSEL input pin (pin AB5) determines how the source for the Host I2C clock is selected. For details on the clock structure, see Section Chapter 9, “Clocks” on page 131.

When I2CCLKSEL is 0, this bit is used to select the source for the Host I2C clock between Input Clock 1 (INCLK1) and the system clock (SYSCLK) as follows.
When this bit = 0b, the source for the Host I2C clock is the system clock. It can be further divided using the I2C Clock Divide Select bits (REG[0062h] bits 3-0) and can be controlled by the I2C Clock Enable bit (REG[0063h] bit 0).
When this bit = 1b, the source for the Host I2C clock is Input Clock 1 (INCLK1). It cannot be further divided and is not controlled by the I2C Clock Enable bit.

When I2CCLKSEL is 1, this bit is ignored and the source for the Host I2C clock is Input Clock 1 (INCLK1).

bit 0 I2C Clock Enable
This bit enables/disables the Host I2C clock when the clock source is the divided down system clock.
When this bit = 0b, the Host I2C clock is disabled.
When this bit = 1b, the Host I2C clock is enabled.

10.4.2 Host Interface Registers

REG[0080h] Internal Memory Space Upper Address Register 0 Default = 00h							Read/Write
Internal Memory Space Upper Address bits 23-16							
7	6	5	4	3	2	1	0

REG[0081h] Internal Memory Space Upper Address Register 1 Default = 00h							Read/Write
Internal Memory Space Upper Address bits 31-24							
7	6	5	4	3	2	1	0

REG[0081h] bits 7-0
REG[0080h] bits 7-0 Internal Memory Space Upper Address bits [31:16]

These bits contain the upper 16 bits of the address to concatenate to the Host address for Parallel Direct mode access to Memory Space. The internal Memory Space has a 32-bit address and the Host interface only has up to 21 bits of address lines. This register serves as a "page" register to the internal 32-bit Memory Space.

Bits 31-21 of the internal address use MUADDR[31:21]. Each of bits 20-16 of the internal address can use either MUADDR[x] or the corresponding Host input address line. The selection between MUADDR[x] and input address line is determined by the MUMASK[x] register bits. Bits 15-0 of the internal address use the lower 16 bits of the Host input address lines.

REG[0082h] Internal Memory Space Upper Address Mask Register Default = 00h							Read/Write
n/a			Internal Memory Space Upper Address Mask bits 20-16				
7	6	5	4	3	2	1	0

bits 4-0 Internal Memory Space Upper Address Mask bits [20:16]

These bits select the source of the internal address bits 20-16 for Parallel Direct access to Memory Space.
When MUMASK[x]=0b, the corresponding Host input address line is used.
When MUMASK[x]=1b, MUADDR[x] is used.

REG[0084h] Host Control Register 0 Default = 00h							Read/Write
n/a							Asynchronous System Control Registers Host Access
7	6	5	4	3	2	1	0

bit 0 Asynchronous System Control Registers Host Access
This bit controls write accesses to the asynchronous registers REG[0020h] ~ REG[003Fh]. This bit has no effect on read accesses from REG[0020h] ~ REG[003Fh] or read/write accesses for all other registers.
When this bit = 0b, REG[0020h] ~ REG[003Fh] are accessed synchronously by the internal VBUS and cannot be directly written by the Host. In this mode, the Host can still indirectly write to REG[0020h] ~ REG[003Fh] using the Internal Memory Space Data Port (REG[00ACh] ~ REG[00ADh]) at the internal memory space 3800_xxxxh (see REG[00A8h] ~ REG[00ABh]).
When this bit = 1b, REG[0020h] ~ REG[003Fh] are accessed asynchronously by the Host and cannot be written by the internal VBUS.

REG[0085h] Host Control Register 1 Default = 04h							Read/Write
n/a			Reserved	n/a	Read Data Setup Cycles bits 2-0		
7	6	5	4	3	2	1	0

bit 4 Reserved
The default value of this bit is 0b.

bits 2-0 Read Data Setup Cycles bits [2:0]
When the Marvell PXA3xx host interface is used, a read data setup time of 30ns is required. These bits specify the read data setup cycles before rising edge of RDY (WAIT#). Read data setup cycles must be set based on system clock cycle as follows.
read data setup = (value of these bits) x (system clock period).

REG[008Ah] Host Control Register 2 Default = 00h							Write Only
n/a						FP2IO C33PE Debugger Pins Enable	S1D13515/ S2D13515 Software Reset
7	6	5	4	3	2	1	0

bit 4 C33PE Debugger Pins Enable
This bit controls the function of the FP2IO10, FP2IO11, FP2IO13, FP2IO14, FP2IO16, and FP2IO17 pins when the host select is Direct 16-bit and not Marvell PXA3xx.
When this bit = 0b, the FP2IOx pins are used for pixel data.
When this bit = 1b, the FP2IOx pins are used for the C33PE Debugger interface.

bit 0 S1D13515/S2D13515 Software Reset
This bit controls the S1D13515/S2D13515 software reset.
When this bit = 0b, the S1D13515/S2D13515 systems are released from reset.
When this bit = 1b, all S1D13515/S2D13515 systems except the Host Interface are held in reset.

The following sequence must be used to correctly execute a software reset.

1. Set software reset, REG[008Ah] bit 0 = 1b.
2. Disable software reset, REG[008Ah] bit 0 = 0b.
3. Set Async Register Write Access to Host, REG[0084h] bit 0 = 1b.
4. Enable PLL1, set REG[0024h] bit 0 = 1b.
5. Set Async Register Write Access to internal VBUS, REG[0084h] bit 0 = 0b.

REG[00A6h] Internal Memory Space Read/Write Control Register Default = 00h							Read/Write
n/a							Internal Memory Space Auto-Increment Enable
7	6	5	4	3	2	1	0

bit 0 Internal Memory Space Auto-Increment Enable
This bit controls auto-increment of the Internal Memory Space Read/Write Address registers (REG[00A8h] ~ REG[00ABh]) for host accesses to the internal memory space through the Internal Memory Space Read/Write Data Port (REG[00ACh] ~ REG[00ADh]).
When this bit = 0b, the internal memory space address is not auto-incremented.
When this bit = 1b, the internal memory space address is auto-incremented.

REG[00A8h] Internal Memory Space Read/Write Address Register 0 Default = 00h							Read/Write
Internal Memory Space Read/Write Address bits 7-0							
7	6	5	4	3	2	1	0

REG[00A9h] Internal Memory Space Read/Write Address Register 1 Default = 00h							Read/Write
Internal Memory Space Read/Write Address bits 15-8							
7	6	5	4	3	2	1	0

REG[00AAh] Internal Memory Space Read/Write Address Register 2 Default = 00h							Read/Write
Internal Memory Space Read/Write Address bits 23-16							
7	6	5	4	3	2	1	0

REG[00ABh] Internal Memory Space Read/Write Address Register 3 Default = 00h							Read/Write
Internal Memory Space Read/Write Address bits 31-24							
7	6	5	4	3	2	1	0

REG[00ABh] bits 7-0
REG[00AAh] bits 7-0
REG[00A9h] bits 7-0
REG[00A8h] bits 7-0 Internal Memory Space Read/Write Address bits [31:0]

These bits specify the internal memory space address to read/write when the Host accesses the Internal Memory Space Read/Write Data Port (REG[00ACh] ~ REG[00ADh]). These bits are auto-incremented when REG[00A6h] bit 0 = 1b. See Chapter 8, “Memory Map” on page 130 for address information.

Note

When using SPI, I2C, or any interface without WAIT, SDRAM must be accessed using the SDRAM Read/Write Buffers (see Section 10.4.9, “SDRAM Read/Write Buffer Registers” on page 209).

REG[00ACh] Internal Memory Space Read/Write Data Port Register 0 Default = 00h Read/Write							
Internal Memory Space Read/Write Data Port bits 7-0							
7	6	5	4	3	2	1	0

REG[00ADh] Internal Memory Space Read/Write Data Port Register 1 Default = 00h Read/Write							
Internal Memory Space Read/Write Data Port bits 15-8							
7	6	5	4	3	2	1	0

REG[00ADh] bits 7-0
REG[00ACh] bits 7-0 Internal Memory Space Read/Write Data Port bits [15:0]
These bits are the data port where the Host can access the internal memory space. The address that will be written to or read from is specified in REG[00A8h] ~ REG[00ABh].

Note

1. When using SPI, I2C, or any interface without WAIT, SDRAM must be accessed using the SDRAM Read/Write Buffers (see Section 10.4.9, “SDRAM Read/Write Buffer Registers” on page 209).
2. When using SPI for non-SDRAM read accesses, the Internal Memory Space Read/Write Address bits (REG[00A8h] ~ REG[00ABh]) must be set before read accesses from this port.

10.4.3 Bit Per Pixel Converter Configuration Registers

The Bit-Per-Pixel Converter (BPPC) can be used to up-convert or down-convert image data between 32 bpp unpacked and 8/16 bpp as shown below. See Chapter 12, “Bit-Per-Pixel Converter Functional Description” on page 445 for further information.

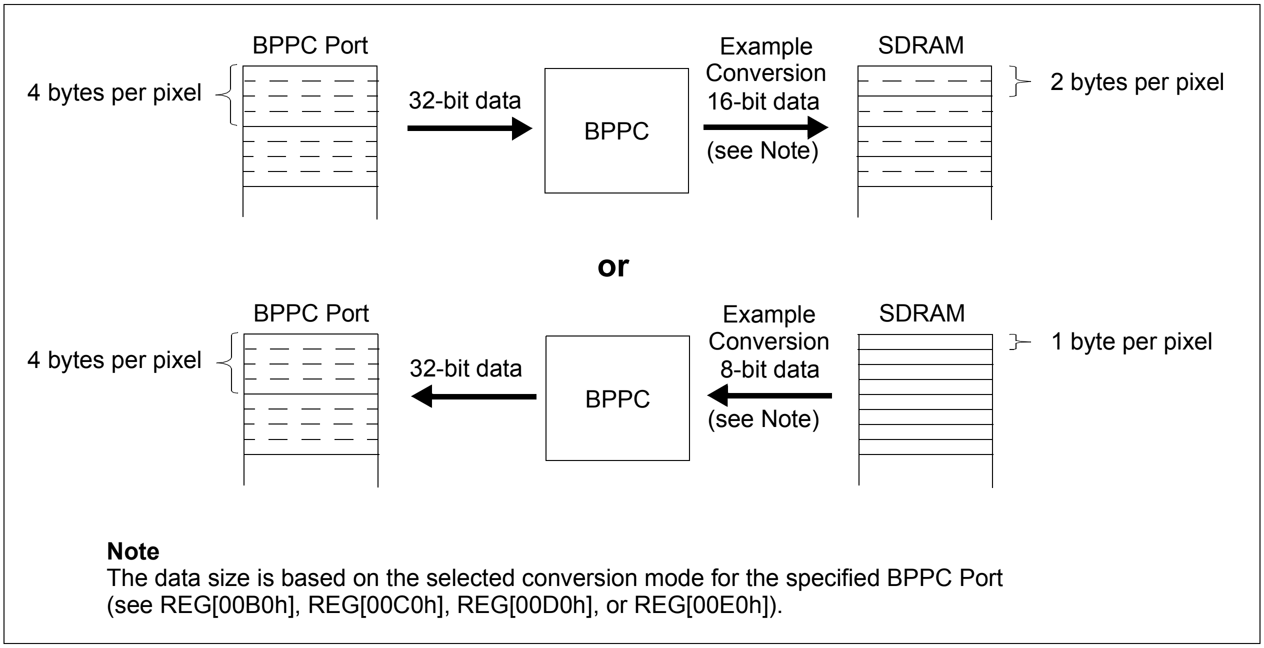


Figure 10-1: BPPC Conversion Example

Note
The Bit Per Pixel Converter (BPPC) Ports cannot be accessed through the Host interface. Accesses to and from the BPPC ports must be in 32-bit units.

REG[00B0h] BPPC Port 0 Mode Configuration Register 0 Default = 00h							Read/Write
n/a				BPPC Port 0 Conversion Mode bits 3-0			
7	6	5	4	3	2	1	0

bits 3-0 BPPC Port 0 Conversion Mode bits [3:0]
These bits determine the address and data conversion mode as shown in the following table.

Table 10-18: BPPC Port 0 Conversion Modes

REG[00B0h] bits 3-0	Data Conversion Mode
0000b	No change
0001b	8 bpp conversion, [A8,R8,G8,B8] → R3G3B2, [FFh,R8,G8,B8] ← R3G3B2
0010b	16 bpp conversion, [A8,R8,G8,B8] → R5G6B5, [FFh,R8,G8,B8] ← R5G6B5
0011b	8 bpp conversion for reads only, [FFh, Lum8, Lum8, Lum8] ← Lum8
0100b	Reserved
0101b	8 bpp conversion for reads only, [Alpha8, 00h, 00h, 00h] ← Alpha8
0110b	16 bpp conversion for reads only, [{A4,A4}, {R4,R4}, {G4,G4}, {B4,B4}] ← R4G4B4A4
0111b	8 bpp conversion for reads only, [{Alpha4,Alpha4}, {Lum4,Lum4}, {Lum4,Lum4}, {Lum4,Lum4}] ← [Lum4, Alpha4]
1000b ~ 1001b	Reserved
1010b	16 bpp conversion for reads only, [Alpha8, Lum8, Lum8, Lum8] ← [Lum8, Alpha8]
1011b ~ 1111b	Reserved

REG[00B1h] BPPC Port 0 Mode Configuration Register 1 Default = 00h							Read/Write
n/a						BPPC Port 0 ARGB Byte Arrangement bits 1-0	
7	6	5	4	3	2	1	0

bits 1-0 BPPC Port 0 ARGB Byte Arrangement bits [1:0]
These bits configure the expected ARGB data arrangement in 32-bit WORD.

Table 10-19: Expected BPPC Port 0 ARGB Data Arrangement

REG[00B1h] bits 1-0	32-bit WORD			
	Bits[31:24]	Bits[23:16]	Bits[15:8]	Bits[7:0]
00b	Alpha	Red	Green	Blue
01b	Red	Green	Blue	Alpha
10b	Alpha	Blue	Green	Red
11b	Blue	Green	Red	Alpha

REG[00B4h] BPPC Port 0 Base Register 0 Default = 00h							Read Only
BPPC Port 0 Base bits 7-0							
7	6	5	4	3	2	1	0

REG[00B5h] BPPC Port 0 Base Register 1 Default = 00h							Read Only
BPPC Port 0 Base bits 15-8							
7	6	5	4	3	2	1	0

REG[00B6h] BPPC Port 0 Base Register 2 Default = 00h							Read Only
BPPC Port 0 Base bits 23-16							
7	6	5	4	3	2	1	0

REG[00B7h] BPPC Port 0 Base Register 3 Default = 40h							Read Only
BPPC Port 0 Base bits 31-24							
7	6	5	4	3	2	1	0

REG[00B7h] bits 7-0
REG[00B6h] bits 7-0
REG[00B5h] bits 7-0
REG[00B4h] bits 7-0 BPPC Port 0 Base bits [31:0] (Read Only)
These bits indicate the base address for Port 0 of the BPPC. These bits are read only and have a value of 4000_0000h.

REG[00B8h] BPPC Port 0 Mask Register 0 Default = 00h							Read/Write
BPPC Port 0 Mask bits 7-0							
7	6	5	4	3	2	1	0

REG[00B9h] BPPC Port 0 Mask Register 1 Default = 00h							Read/Write
BPPC Port 0 Mask bits 15-8							
7	6	5	4	3	2	1	0

REG[00BAh] BPPC Port 0 Mask Register 2 Default = 00h							Read/Write
BPPC Port 0 Mask bits 23-16							
7	6	5	4	3	2	1	0

REG[00BBh] BPPC Port 0 Mask Register 3 Default = 00h							Read/Write
n/a				BPPC Port 0 Mask bits 27-24			
7	6	5	4	3	2	1	0

REG[00BBh] bits 3-0
REG[00BAh] bits 7-0
REG[00B9h] bits 7-0
REG[00B8h] bits 7-0 BPPC Port 0 Mask bits [27:0]

These bits are used in combination with the BPPC Port 0 Target Base bits (see REG[00BCh] ~ REG[00BFh]) and specify the mask to validate the port address to a specific range for Port 0 of the BPPC. These bits must be byte, 2 byte, or 4 byte aligned based on the selected BPPC Port 0 Conversion Mode (see REG[00B0h] bits 3-0). For example, if 8 bpp conversion is selected, the bits must be byte aligned. For 16 bpp conversion, the bits must be 2 byte aligned.

The lower 28 bits of the port address is ANDed with the compliment of the Mask Register and the result is then added to the Target Base Register. Refer to the BPPC Port 0 Target Base register description (REG[00BCh] ~ REG[00BFh]) for the required equations.

REG[00BCh] BPPC Port 0 Target Base Register 0							
Default = 00h							Read/Write
BPPC Port 0 Target Base bits 7-0							
7	6	5	4	3	2	1	0

REG[00BDh] BPPC Port 0 Target Base Register 1							
Default = 00h							Read/Write
BPPC Port 0 Target Base bits 15-8							
7	6	5	4	3	2	1	0

REG[00BEh] BPPC Port 0 Target Base Register 2							
Default = 00h							Read/Write
BPPC Port 0 Target Base bits 23-16							
7	6	5	4	3	2	1	0

REG[00BFh] BPPC Port 0 Target Base Register 3							
Default = 00h							Read/Write
BPPC Port 0 Target Base bits 31-24							
7	6	5	4	3	2	1	0

REG[00BFh] bits 7-0
REG[00BEh] bits 7-0
REG[00BDh] bits 7-0
REG[00BCh] bits 7-0 BPPC Port 0 Target Base bits [31:0]

These bits are used in combination with the BPPC Port 0 Mask bits (see REG[00B8h] ~ REG[00BBh]) and specify the target base address which determines the memory target address for Port 0 of the BPPC. These bits must be byte, 2 byte, or 4 byte aligned based on the selected BPPC Port 0 Conversion Mode (see REG[00B0h] bits 3-0). For example, if 8 bpp conversion is selected, the bits must be byte aligned. For 16 bpp conversion, the bits must be 2 byte aligned.

The target address is generated according to the following equations:

```
MaskedAddr[27:0] = PortAddr[27:0] & ~Mask[27:0]

if (8 bpp format)
    ConvertedAddr[27:0] = {00, MaskedAddr[27:2]}
else if (16 bpp format)
    ConvertedAddr[27:0] = {0, MaskedAddr[27:1]}
else ConvertedAddr[27:0] = MaskedAddr[27:0]

TargetAddr[31:0] = TargetBase[31:0] + {0000, ConvertedAddr[27:0]}
```

REG[00C0h] BPPC Port 1 Mode Configuration Register 0 Default = 00h							Read/Write
n/a				BPPC Port 1 Conversion Mode bits 3-0			
7	6	5	4	3	2	1	0

bits 3-0 BPPC Port 1 Conversion Mode bits [3:0]
These bits determine the address and data conversion mode as follows.

Table 10-20: BPPC Port 1 Conversion Modes

REG[00C0h] bits 3-0	Data Conversion Mode
0000b	No change
0001b	8 bpp conversion, [A8,R8,G8,B8] → R3G3B2, [FFh,R8,G8,B8] ← R3G3B2
0010b	16 bpp conversion, [A8,R8,G8,B8] → R5G6B5, [FFh,R8,G8,B8] ← R5G6B5
0011b	8 bpp conversion for reads only, [FFh, Lum8, Lum8, Lum8] ← Lum8
0100b	Reserved
0101b	8 bpp conversion for reads only, [Alpha8, 00h, 00h, 00h] ← Alpha8
0110b	16 bpp conversion for reads only, [{A4,A4}, {R4,R4}, {G4,G4}, {B4,B4}] ← R4G4B4A4
0111b	8 bpp conversion for reads only, [{Alpha4,Alpha4}, {Lum4,Lum4}, {Lum4,Lum4}, {Lum4,Lum4}] ← [Lum4, Alpha4]
1000b ~ 1001b	Reserved
1010b	16 bpp conversion for reads only, [Alpha8, Lum8, Lum8, Lum8] ← [Lum8, Alpha8]
1011b ~ 1111b	Reserved

REG[00C1h] BPPC Port 1 Mode Configuration Register 1 Default = 00h							Read/Write
n/a						BPPC Port 1 ARGB Byte Arrangement bits 1-0	
7	6	5	4	3	2	1	0

bits 1-0 BPPC Port 1 ARGB Byte Arrangement bits [1:0]
These bits configure the expected ARGB data arrangement in 32-bit WORD.

Table 10-21: Expected BPPC Port 1 ARGB Data Arrangement

REG[00C1h] bits 1-0	32-bit WORD			
	Bits[31:24]	Bits[23:16]	Bits[15:8]	Bits[7:0]
00b	Alpha	Red	Green	Blue
01b	Red	Green	Blue	Alpha
10b	Alpha	Blue	Green	Red
11b	Blue	Green	Red	Alpha

REG[00C4h] BPPC Port 1 Base Register 0 Default = 00h							Read Only
BPPC Port 1 Base bits 7-0							
7	6	5	4	3	2	1	0

REG[00C5h] BPPC Port 1 Base Register 1 Default = 00h							Read Only
BPPC Port 1 Base bits 15-8							
7	6	5	4	3	2	1	0

REG[00C6h] BPPC Port 1 Base Register 2 Default = 00h							Read Only
BPPC Port 1 Base bits 23-16							
7	6	5	4	3	2	1	0

REG[00C7h] BPPC Port 1 Base Register 3 Default = 50h							Read Only
BPPC Port 1 Base bits 31-24							
7	6	5	4	3	2	1	0

REG[00C7h] bits 7-0
REG[00C6h] bits 7-0
REG[00C5h] bits 7-0
REG[00C4h] bits 7-0 BPPC Port 1 Base bits [31:0] (Read Only)
These bits indicate the base address for Port 1 of the BPPC. These bits are read only and have a value of 5000_0000h.

REG[00C8h] BPPC Port 1 Mask Register 0 Default = 00h							Read/Write
BPPC Port 1 Mask bits 7-0							
7	6	5	4	3	2	1	0

REG[00C9h] BPPC Port 1 Mask Register 1 Default = 00h							Read/Write
BPPC Port 1 Mask bits 15-8							
7	6	5	4	3	2	1	0

REG[00CAh] BPPC Port 1 Mask Register 2 Default = 00h							Read/Write
BPPC Port 1 Mask bits 23-16							
7	6	5	4	3	2	1	0

REG[00CBh] BPPC Port 1 Mask Register 3 Default = 00h							Read/Write
n/a				BPPC Port 1 Mask bits 27-24			
7	6	5	4	3	2	1	0

REG[00CBh] bits 3-0
REG[00CAh] bits 7-0
REG[00C9h] bits 7-0
REG[00C8h] bits 7-0 BPPC Port 1 Mask bits [27:0]

These bits are used in combination with the BPPC Port 1 Target Base bits (see REG[00CCh] ~ REG[00CFh]) and specify the mask to validate the port address to a specific range for Port 1 of the BPPC. These bits must be byte, 2 byte, or 4 byte aligned based on the selected BPPC Port 1 Conversion Mode (see REG[00C0h] bits 3-0). For example, if 8 bpp conversion is selected, the bits must be byte aligned. For 16 bpp conversion, the bits must be 2 byte aligned.

The lower 28 bits of the port address is ANDed with the compliment of the Mask Register and the result is then added to the Target Base Register. Refer to the BPPC Port 1 Target Base register description (REG[00CCh] ~ REG[00CFh]) for the required equations.

REG[00CCh] BPPC Port 1 Target Base Register 0 Default = 00h							Read/Write
BPPC Port 1 Target Base bits 7-0							
7	6	5	4	3	2	1	0

REG[00CDh] BPPC Port 1 Target Base Register 1 Default = 00h							Read/Write
BPPC Port 1 Target Base bits 15-8							
7	6	5	4	3	2	1	0

REG[00CEh] BPPC Port 1 Target Base Register 2 Default = 00h							Read/Write
BPPC Port 1 Target Base bits 23-16							
7	6	5	4	3	2	1	0

REG[00CFh] BPPC Port 1 Target Base Register 3 Default = 00h							Read/Write
BPPC Port 1 Target Base bits 31-24							
7	6	5	4	3	2	1	0

REG[00CFh] bits 7-0
REG[00CEh] bits 7-0
REG[00CDh] bits 7-0
REG[00CCh] bits 7-0 BPPC Port 1 Target Base bits [31:0]

These bits are used in combination with the BPPC Port 1 Mask bits (see REG[00C8h] ~ REG[00CBh]) and specify the target base address which determines the memory target address for Port 1 of the BPPC. These bits must be byte, 2 byte, or 4 byte aligned based on the selected BPPC Port 1 Conversion Mode (see REG[00C0h] bits 3-0). For example, if 8 bpp conversion is selected, the bits must be byte aligned. For 16 bpp conversion, the bits must be 2 byte aligned.

The target address is generated according to the following equations.

```
MaskedAddr[27:0] = PortAddr[27:0] & ~Mask[27:0]

if (8 bpp format)
    ConvertedAddr[27:0] = {00, MaskedAddr[27:2]}
else if (16 bpp format)
    ConvertedAddr[27:0] = {0, MaskedAddr[27:1]}
else ConvertedAddr[27:0] = MaskedAddr[27:0]

TargetAddr[31:0] = TargetBase[31:0] + {0000, ConvertedAddr[27:0]}
```

REG[00D0h] BPPC Port 2 Mode Configuration Register 0 Default = 00h							Read/Write
n/a				BPPC Port 2 Conversion Mode bits 3-0			
7	6	5	4	3	2	1	0

bits 3-0 BPPC Port 2 Conversion Mode bits [3:0]
These bits determine the address and data conversion mode as follows.

Table 10-22: BPPC Port 2 Conversion Modes

REG[00D0h] bits 3-0	Data Conversion Mode
0000b	No change
0001b	8 bpp conversion, [A8,R8,G8,B8] → R3G3B2, [FFh,R8,G8,B8] ← R3G3B2
0010b	16 bpp conversion, [A8,R8,G8,B8] → R5G6B5, [FFh,R8,G8,B8] ← R5G6B5
0011b	8 bpp conversion for reads only, [FFh, Lum8, Lum8, Lum8] ← Lum8
0100b	Reserved
0101b	8 bpp conversion for reads only, [Alpha8, 00h, 00h, 00h] ← Alpha8
0110b	16 bpp conversion for reads only, [{A4,A4}, {R4,R4}, {G4,G4}, {B4,B4}] ← R4G4B4A4
0111b	8 bpp conversion for reads only, [{Alpha4,Alpha4}, {Lum4,Lum4}, {Lum4,Lum4}, {Lum4,Lum4}] ← [Lum4, Alpha4]
1000b ~ 1001b	Reserved
1010b	16 bpp conversion for reads only, [Alpha8, Lum8, Lum8, Lum8] ← [Lum8, Alpha8]
1011b ~ 1111b	Reserved

REG[00D1h] BPPC Port 2 Mode Configuration Register 1 Default = 00h							Read/Write
n/a						BPPC Port 2 ARGB Byte Arrangement bits 1-0	
7	6	5	4	3	2	1	0

bits 1-0 BPPC Port 2 ARGB Byte Arrangement bits [1:0]
These bits configure the expected ARGB data arrangement in 32-bit WORD.

Table 10-23: Expected BPPC Port 2 ARGB Data Arrangement

REG[00D1h] bits 1-0	32-bit WORD			
	Bits[31:24]	Bits[23:16]	Bits[15:8]	Bits[7:0]
00b	Alpha	Red	Green	Blue
01b	Red	Green	Blue	Alpha
10b	Alpha	Blue	Green	Red
11b	Blue	Green	Red	Alpha

REG[00D4h] BPPC Port 2 Base Register 0 Default = 00h							Read Only
BPPC Port 2 Base bits 7-0							
7	6	5	4	3	2	1	0

REG[00D5h] BPPC Port 2 Base Register 1 Default = 00h							Read Only
BPPC Port 2 Base bits 15-8							
7	6	5	4	3	2	1	0

REG[00D6h] BPPC Port 2 Base Register 2 Default = 00h							Read Only
BPPC Port 2 Base bits 23-16							
7	6	5	4	3	2	1	0

REG[00D7h] BPPC Port 2 Base Register 3 Default = 60h							Read Only
BPPC Port 2 Base bits 31-24							
7	6	5	4	3	2	1	0

REG[00D7h] bits 7-0
REG[00D6h] bits 7-0
REG[00D5h] bits 7-0
REG[00D4h] bits 7-0 BPPC Port 2 Base bits [31:0] (Read Only)

These bits indicate the base address for Port 2 of the BPPC. These bits are read only and have a value of 6000_0000h.

REG[00D8h] BPPC Port 2 Mask Register 0 Default = 00h							Read/Write
BPPC Port 2 Mask bits 7-0							
7	6	5	4	3	2	1	0

REG[00D9h] BPPC Port 2 Mask Register 1 Default = 00h							Read/Write
BPPC Port 2 Mask bits 15-8							
7	6	5	4	3	2	1	0

REG[00DAh] BPPC Port 2 Mask Register 2 Default = 00h							Read/Write
BPPC Port 2 Mask bits 23-16							
7	6	5	4	3	2	1	0

REG[00DBh] BPPC Port 2 Mask Register 3 Default = 00h							Read/Write
n/a				BPPC Port 2 Mask bits 27-24			
7	6	5	4	3	2	1	0

REG[00DBh] bits 3-0
REG[00DAh] bits 7-0
REG[00D9h] bits 7-0
REG[00D8h] bits 7-0 BPPC Port 2 Mask bits [27:0]

These bits are used in combination with the BPPC Port 2 Target Base bits (see REG[00DCh] ~ REG[00DFh]) and specify the mask to validate the port address to a specific range for Port 2 of the BPPC. These bits must be byte, 2 byte, or 4 byte aligned based on the selected BPPC Port 2 Conversion Mode (see REG[00D0h] bits 3-0). For example, if 8 bpp conversion is selected, the bits must be byte aligned. For 16 bpp conversion, the bits must be 2 byte aligned.

The lower 28 bits of the port address is ANDed with the compliment of the Mask Register and the result is then added to the Target Base Register. Refer to the BPPC Port 2 Target Base register description (REG[00DCh] ~ REG[00DFh]) for the required equations.

REG[00DCh] BPPC Port 2 Target Base Register 0 Default = 00h							Read/Write
BPPC Port 2 Target Base bits 7-0							
7	6	5	4	3	2	1	0

REG[00DDh] BPPC Port 2 Target Base Register 1 Default = 00h							Read/Write
BPPC Port 2 Target Base bits 15-8							
7	6	5	4	3	2	1	0

REG[00DEh] BPPC Port 2 Target Base Register 2 Default = 00h							Read/Write
BPPC Port 2 Target Base bits 23-16							
7	6	5	4	3	2	1	0

REG[00DFh] BPPC Port 2 Target Base Register 3 Default = 00h							Read/Write
BPPC Port 2 Target Base bits 31-24							
7	6	5	4	3	2	1	0

REG[00DFh] bits 7-0
REG[00DEh] bits 7-0
REG[00DDh] bits 7-0
REG[00DCh] bits 7-0 BPPC Port 2 Target Base bits [31:0]

These bits are used in combination with the BPPC Port 2 Mask bits (see REG[00D8h] ~ REG[00DBh]) and specify the target base address which determines the memory target address for Port 2 of the BPPC. These bits must be byte, 2 byte, or 4 byte aligned based on the selected BPPC Port 2 Conversion Mode (see REG[00D0h] bits 3-0). For example, if 8 bpp conversion is selected, the bits must be byte aligned. For 16 bpp conversion, the bits must be 2 byte aligned.

The target address is generated according to the following equations.

```
MaskedAddr[27:0] = PortAddr[27:0] & ~Mask[27:0]

if (8 bpp format)
    ConvertedAddr[27:0] = {00, MaskedAddr[27:2]}
else if (16 bpp format)
    ConvertedAddr[27:0] = {0, MaskedAddr[27:1]}
else ConvertedAddr[27:0] = MaskedAddr[27:0]

TargetAddr[31:0] = TargetBase[31:0] + {0000, ConvertedAddr[27:0]}
```

REG[00E0h] BPPC Port 3 Mode Configuration Register 0 Default = 00h							Read/Write
n/a				BPP Port 3 Conversion Mode bits 3-0			
7	6	5	4	3	2	1	0

bits 3-0 BPP Port 3 Conversion Mode bits [3:0]
These bits determine the address and data conversion mode as follows.

Table 10-24: BPPC Port 3 Conversion Modes

REG[00E0h] bits 3-0	Data Conversion Mode
0000b	No change
0001b	8 bpp conversion, [A8,R8,G8,B8] → R3G3B2, [FFh,R8,G8,B8] ← R3G3B2
0010b	16 bpp conversion, [A8,R8,G8,B8] → R5G6B5, [FFh,R8,G8,B8] ← R5G6B5
0011b	8 bpp conversion for reads only, [FFh, Lum8, Lum8, Lum8] ← Lum8
0100b	Reserved
0101b	8 bpp conversion for reads only, [Alpha8, 00h, 00h, 00h] ← Alpha8
0110b	16 bpp conversion for reads only, [{A4,A4}, {R4,R4}, {G4,G4}, {B4,B4}] ← R4G4B4A4
0111b	8 bpp conversion for reads only, [{Alpha4,Alpha4}, {Lum4,Lum4}, {Lum4,Lum4}, {Lum4,Lum4}] ← [Lum4, Alpha4]
1000b ~ 1001b	Reserved
1010b	16 bpp conversion for reads only, [Alpha8, Lum8, Lum8, Lum8] ← [Lum8, Alpha8]
1011b ~ 1111b	Reserved

REG[00E1h] BPPC Port 3 Mode Configuration Register 1 Default = 00h							Read/Write
n/a						BPPC Port 3 ARGB Byte Arrangement bits 1-0	
7	6	5	4	3	2	1	0

bits 1-0 BPPC Port 3 ARGB Byte Arrangement bits [1:0]
These bits configure the expected ARGB data arrangement in 32-bit WORD.

Table 10-25: Expected BPPC Port 3 ARGB Data Arrangement

REG[00E1h] bits 1-0	32-bit WORD			
	Bits[31:24]	Bits[23:16]	Bits[15:8]	Bits[7:0]
00b	Alpha	Red	Green	Blue
01b	Red	Green	Blue	Alpha
10b	Alpha	Blue	Green	Red
11b	Blue	Green	Red	Alpha

REG[00E4h] BPPC Port 3 Base Register 0 Default = 00h							Read Only
BPPC Port 3 Base bits 7-0							
7	6	5	4	3	2	1	0

REG[00E5h] BPPC Port 3 Base Register 1 Default = 00h							Read Only
BPPC Port 3 Base bits 15-8							
7	6	5	4	3	2	1	0

REG[00E6h] BPPC Port 3 Base Register 2 Default = 00h							Read Only
BPPC Port 3 Base bits 23-16							
7	6	5	4	3	2	1	0

REG[00E7h] BPPC Port 3 Base Register 3 Default = 70h							Read Only
BPPC Port 3 Base bits 31-24							
7	6	5	4	3	2	1	0

REG[00E7h] bits 7-0
REG[00E6h] bits 7-0
REG[00E5h] bits 7-0
REG[00E4h] bits 7-0 BPPC Port 3 Base bits [31:0] (Read Only)
These bits indicate the base address for Port 3 of the BPPC. These bits are read only and have a value of 7000_0000h.

REG[00E8h] BPPC Port 3 Mask Register 0 Default = 00h							Read/Write
BPPC Port 3 Mask bits 7-0							
7	6	5	4	3	2	1	0

REG[00E9h] BPPC Port 3 Mask Register 1 Default = 00h							Read/Write
BPPC Port 3 Mask bits 15-8							
7	6	5	4	3	2	1	0

REG[00EAh] BPPC Port 3 Mask Register 2 Default = 00h							Read/Write
BPPC Port 3 Mask bits 23-16							
7	6	5	4	3	2	1	0

REG[00EBh] BPPC Port 3 Mask Register 3 Default = 00h							Read/Write
n/a				BPPC Port 3 Mask bits 27-24			
7	6	5	4	3	2	1	0

REG[00EBh] bits 3-0
REG[00EAh] bits 7-0
REG[00E9h] bits 7-0
REG[00E8h] bits 7-0 BPPC Port 3 Mask bits [27:0]

These bits are used in combination with the BPPC Port 3 Target Base bits (see REG[00ECh] ~ REG[00EFh]) and specify the mask to validate the port address to a specific range for Port 3 of the BPPC. These bits must be byte, 2 byte, or 4 byte aligned based on the selected BPPC Port 3 Conversion Mode (see REG[00E0h] bits 3-0). For example, if 8 bpp conversion is selected, the bits must be byte aligned. For 16 bpp conversion, the bits must be 2 byte aligned.

The lower 28 bits of the port address is ANDed with the compliment of the Mask Register and the result is then added to the Target Base Register. Refer to the BPPC Port 3 Target Base register description (REG[00ECh] ~ REG[00EFh]) for the required equations.

REG[00ECh] BPPC Port 3 Target Base Register 0 Default = 00h							Read/Write
BPPC Port 3 Target Base bits 7-0							
7	6	5	4	3	2	1	0

REG[00EDh] BPPC Port 3 Target Base Register 1 Default = 00h							Read/Write
BPPC Port 3 Target Base bits 15-8							
7	6	5	4	3	2	1	0

REG[00EEh] BPPC Port 3 Target Base Register 2 Default = 00h							Read/Write
BPPC Port 3 Target Base bits 23-16							
7	6	5	4	3	2	1	0

REG[00EFh] BPPC Port 3 Target Base Register 3 Default = 00h							Read/Write
BPPC Port 3 Target Base bits 31-24							
7	6	5	4	3	2	1	0

REG[00EFh] bits 7-0
REG[00EEh] bits 7-0
REG[00EDh] bits 7-0
REG[00ECh] bits 7-0 BPPC Port 3 Target Base bits [31:0]

These bits are used in combination with the BPPC Port 3 Mask bits (see REG[00E8h] ~ REG[00EBh]) and specify the target base address which determines the memory target address for Port 3 of the BPPC. These bits must be byte, 2 byte, or 4 byte aligned based on the selected BPPC Port 3 Conversion Mode (see REG[00E0h] bits 3-0). For example, if 8 bpp conversion is selected, the bits must be byte aligned. For 16 bpp conversion, the bits must be 2 byte aligned.

The target address is generated according to the following equations.

```
MaskedAddr[27:0] = PortAddr[27:0] & ~Mask[27:0]

if (8 bpp format)
    ConvertedAddr[27:0] = {00, MaskedAddr[27:2]}
else if (16 bpp format)
    ConvertedAddr[27:0] = {0, MaskedAddr[27:1]}
else ConvertedAddr[27:0] = MaskedAddr[27:0]

TargetAddr[31:0] = TargetBase[31:0] + {0000, ConvertedAddr[27:0]}
```

10.4.4 I2S Control Registers

The S1D13515/S2D13515 includes an I2S interface which is typically used for audio output. For information concerning this interface, see Chapter 14, “I2S Audio Output Interface” on page 481. For information on configuring the I2S DMA buffers, refer to Section 10.4.5, “I2S DMA Registers” on page 189.

REG[0100h] I2S Interface Control Register 0 Default = 21h							Read/Write
I2S Blank Left Channel	I2S Blank Right Channel	I2S Left/Right Channel Data Order	I2S Data Transition Clock Edge	I2S WSIO Data Timing	I2S Data Bit Ordering	n/a	IS2 Output Data Clock Source
7	6	5	4	3	2	1	0

bit 7 I2S Blank Left Channel
This bit is used to blank left channel data for the I2S interface.
When this bit = 0b, the left channel data is normal.
When this bit = 1b, the left channel data is blanked.

bit 6 I2S Blank Right Channel
This bit is used to blank right channel data for the I2S interface.
When this bit = 0b, the right channel data is normal.
When this bit = 1b, the right channel data is blanked.

bit 5 I2S Left/Right Channel Data Order
This bit determines the left/right channel data order relative to the state of the WSIO pin.
When this bit = 0b, the left/right channel data order is left channel when WSIO = 1, right channel when WSIO = 0.
When this bit = 1b, the left/right channel data order is right channel when WSIO = 1, left channel when WSIO = 0.

Note
If the channel data order must be changed while the I2S interface is running, the I2S FIFO must be cleared using the following sequence.
1. Disable the I2S DAC Controller, REG[0104h] bit 0 = 0b
2. Reset the I2S FIFO, REG[010Ch] bit 8 = 1b
3. Change the I2S Left/Right Channel Data Order, REG[0100h] bit 5
4. Enable the I2S DAC Controller, REG[0104h] bit 0 = 1b

bit 4 I2S Data Transition Clock Edge
This bit determines when the serial output data on the SDO pin changes.
When this bit = 0b, serial output data changes on the falling edge of the serial output source clock.
When this bit = 1b, serial output data changes on the rising edge of the serial output source clock.

bit 3 I2S WSIO Data Timing
This bit determines when serial data output on the SDO pin occurs relative to the word sync signal edge (WSIO).
When this bit = 0b, serial output data starts one clock after the WSIO edge.
When this bit = 1b, serial output data starts on the same clock edge as WSIO.

bit 2 — I2S Data Bit Ordering
This bit determines the bit order for serial data output on the SDO pin.
When this bit = 0b, the most significant bit (msb) is sent first.
When this bit = 1b, the least significant bit (lsb) is sent first.

bit 0 — I2S Data Clock Source
This bit selects the source of the data clock used for serial data output on the SDO pin. This bit must be set in combination with the WSIO and SCKIO Output Enable bit (REG[0101h] bit 0) as shown in the following table.

Table 10-26 : I2S Data Clock (WSIO/SCKIO) Settings

REG[0101h] bit 0	REG[0100h] bit 0	Description
0b (default)	0b	Reserved
	1b (default)	I2S data clock source is the internal clock. WSIO/SCKIO are outputs driven by the internal clocks.
1b	0b	I2S data clock source is an external clock and WSIO/SCKIO are inputs (high-impedance).
	1b	Reserved

REG[0101h] I2S Interface Control Register 1
Default = 40h — Read/Write

7	6	5	4	3	2	1	0
n/a	Reserved	n/a					WSIO and SCKIO Output Enable

bit 6 — Reserved
This bit must be set to 1b.

bit 0 — WSIO and SCKIO Output Enable
This bit controls whether the serial word clock (WSIO) and the serial bit clock (SCKIO) are outputs for the I2S interface. This bit must be set in combination with the I2S Data Clock Source bit (REG[0100h] bit 0) as shown in Table 10-26 “I2S Data Clock (WSIO/SCKIO) Settings” above.

REG[0104h] I2S FIFO Register 0 Default = 00h							Read/Write
I2S FIFO Mode	n/a	I2S FIFO Threshold Level bits 3-0				Reserved	I2S DAC Controller Enable
7	6	5	4	3	2	1	0

bit 7 I2S FIFO Mode
This bit determines whether the data stored in the 16 byte I2S FIFO is stereo (16-bit left channel, 16-bit right channel) or mono (16-bit single data).
When this bit = 0b, the data stored in the I2S FIFO is stereo.
When this bit = 1b, the data stored in the I2S FIFO is mono.

Note
When stereo mode is selected, the I2S FIFO can hold up to 4 audio data samples.
When mono mode is selected, the I2S FIFO can hold up to 8 audio data samples.

bits 5-2 I2S FIFO Threshold Level bits [3:0]
The I2S FIFO size is 16 bytes. These bits specify the I2S FIFO Threshold Level which determines the minimum number of bytes that should be in the I2S FIFO. If the number of bytes becomes less than or equal to the threshold level, an I2S FIFO Threshold Interrupt occurs (see REG[010Ch] bit 2) and a DMA transfer is initiated to increase the number of bytes in the I2S FIFO to the specified level. The recommended setting for these bits is 8h (1000b).

bit 1 Reserved
This bit must be set to 1b.

bit 0 I2S DAC Controller Enable
This bit controls the I2S DAC Controller.
When this bit = 0b, the I2S DAC Controller is disabled and the I2S output stream is stopped.
When this bit = 1b, the I2S DAC Controller is enabled and the I2S output stream is started.

Note

1. When the I2S DAC Controller is enabled and stereo mode is selected (REG[0104h] bit 7 = 0b), the first serial output data is always the Left Channel data.If the I2S Left/Right Channel Data Order bit (REG[0100h] bit 5) is 0b, then the Left Channel data occurs when WSIO = 1 and the Right Channel data occurs when ‘WSIO = 0. If REG[0100h] bit 5 is 1b, then the Left Channel data occurs with WSIO = 0 and the Right Channel data occurs with WSIO = 1.
2. The I2S Audio Interface must be disabled in REG[0104h] bit 0 and REG[010Fh] bit 7 before enabling power save mode in REG[003Ch] bit 0.

REG[0105h] I2S FIFO Register 1 Default = 00h							Read/Write
n/a					I2S FIFO Threshold Interrupt Enable	I2S FIFO Overrun Interrupt Enable	I2S FIFO Underrun Interrupt Enable
7	6	5	4	3	2	1	0

bit 2 I2S FIFO Threshold Interrupt Enable
This bit determines whether the I2S FIFO Threshold Interrupt is indicated at the I2S DAC Interrupt Status bit, REG[0A00h] bit 6, where it can be redirected to the Host.
When this bit = 0b, the I2S FIFO Threshold Interrupt is disabled.
When this bit = 1b, the I2S FIFO Threshold Interrupt is enabled.

bit 1 I2S FIFO Overrun Interrupt Enable
This bit determines whether the I2S FIFO Overrun Interrupt is indicated at the I2S DAC Interrupt Status bit, REG[0A00h] bit 6, where it can be redirected to the Host.
When this bit = 0b, the I2S FIFO Overrun Interrupt is disabled.
When this bit = 1b, the I2S FIFO Overrun Interrupt is enabled.

bit 0 I2S FIFO Underrun Interrupt Enable
This bit determines whether the I2S FIFO Underrun Interrupt is indicated at the I2S DAC Interrupt Status bit, REG[0A00h] bit 6, where it can be redirected to the Host.
When this bit = 0b, the I2S FIFO Underrun Interrupt is disabled.
When this bit = 1b, the I2S FIFO Underrun Interrupt is enabled.

REG[010Ah] I2S FIFO Status Register 0 Default = 00h							Read Only
n/a			I2S FIFO Level bits 4-0				
7	6	5	4	3	2	1	0

bits 4-0 I2S FIFO Level bits [4:0] (Read Only)
These bits indicate the number of bytes of data in the I2S FIFO. The FIFO size is 16 bytes.

REG[010Ch] I2S FIFO Status Register 1 Default = 04h							Read/Write
n/a				I2S FIFO Software Reset (WO)	I2S FIFO Threshold Interrupt Status (RO)	I2S FIFO Overrun Interrupt Status	I2S FIFO Underrun Interrupt Status
7	6	5	4	3	2	1	0

bit 3 I2S FIFO Software Reset (Write Only)
This bit resets the I2S FIFO.
Writing a 0b to this bit has no effect.
Writing a 1b to this bit resets the I2S FIFO.

bit 2 I2S FIFO Threshold Interrupt Status (Read Only)
This read only bit indicates the status of the I2S FIFO Threshold Interrupt which occurs when the number of bytes in the I2S FIFO becomes less than the I2S FIFO Threshold Level, REG[0104h] bits 5-2.
When this bit = 0b, an I2S FIFO Threshold Interrupt has not occurred.
When this bit = 1b, an I2S FIFO Threshold Interrupt has occurred.

This status bit is cleared when data is written to the FIFO to make the number of bytes in the FIFO greater than the threshold value (REG[0104h] bits 5-2).

bit 1 I2S FIFO Overrun Interrupt Status
This bit indicates the status of the I2S FIFO Overrun Interrupt which occurs when the IS2 DMA Controller tries to write to the I2S FIFO when it is already full. If the I2S FIFO Overrun Interrupt Enable bit is set (REG[0105h] bit 1 = 1b), this interrupt is also indicated at the I2S DAC Interrupt Status bit, REG[0A00h] bit 6, where it can be redirected to the Host.
When this bit = 0b, an I2S FIFO Overrun Interrupt has not occurred.
When this bit = 1b, an I2S FIFO Overrun Interrupt has occurred.

To clear this status bit, write a 1b to this bit.

bit 0 — I2S FIFO Underrun Interrupt Status
This bit indicates the status of the I2S FIFO Underrun Interrupt which occurs when the I2S DAC Controller has attempted to read the I2S FIFO while it is empty. If the I2S FIFO Underrun Interrupt Enable bit is set (REG[0105h] bit 0 = 1b), this interrupt is also indicated at the I2S DAC Interrupt Status bit, REG[0A00h] bit 6, where it can be redirected to the Host.
When this bit = 0b, an I2S FIFO Underrun Interrupt has not occurred.
When this bit = 1b, an I2S FIFO Underrun Interrupt has occurred.

To clear this status bit, write a 1b to this bit.

REG[010Eh] I2S Audio Clock Control Register 0 Default = 00h							Read/Write
Audio Clock Phase Increment bits 7-0							
7	6	5	4	3	2	1	0

REG[010Fh] I2S Audio Clock Control Register 1 Default = 00h							Read/Write
Audio Clock Enable	Audio Clock Phase Increment bits 14-8						
7	6	5	4	3	2	1	0

REG[010Fh] bits 6-0
REG[010Eh] bits 7-0 — Audio Clock Phase Increment bits [14:0]
The audio clock (MCLKO) is 256 times the audio sample rate and is derived from the SDRAM clock (see Section Chapter 9, “Clocks” on page 131). The frequency is calculated using the following formula.

MCLKO frequency = (REG[010Fh] bits 6-0, REG[010Eh] bits 7-0) ÷ 65536 x SDRAM clock

Note
The audio clock frequency must be less than one half of 2 x System Clock in order for the phase accumulator logic to work.

REG[010Fh] bit 7 — Audio Clock Enable
This bit controls the Audio Clock (MCLKO).
When this bit = 0b, the audio clock is disabled.
When this bit = 1b, the audio clock is enabled.

Note
The I2S Audio Interface must be disabled in REG[0104h] bit 0 and REG[010Fh] bit 7 before enabling power save mode in REG[003Ch] bit 0.

10.4.5 I2S DMA Registers

When I2S DMA is enabled for the I2S interface (REG[0104h] bit 1 = 1b), data for the I2S FIFO can be written to the I2S DMA buffers (Buffer 0 and Buffer 1). The memory address for each buffer is configurable using the following registers. The I2S DMA controller toggles between reading from these two these buffers when sending data to the I2S FIFO.

REG[0148h] I2S DMA Buffer 0 Address Register 0 Default = 00h							Read/Write
I2S DMA Buffer 0 Address bits 7-0							
7	6	5	4	3	2	1	0

REG[0149h] I2S DMA Buffer 0 Address Register 1 Default = 00h							Read/Write
I2S DMA Buffer 0 Address bits 15-8							
7	6	5	4	3	2	1	0

REG[014Ah] I2S DMA Buffer 0 Address Register 2 Default = 00h							Read/Write
I2S DMA Buffer 0 Address bits 23-16							
7	6	5	4	3	2	1	0

REG[014Bh] I2S DMA Buffer 0 Address Register 3 Default = 00h							Read/Write
I2S DMA Buffer 0 Address bits 31-24							
7	6	5	4	3	2	1	0

REG[014Bh] bits 7-0
REG[014Ah] bits 7-0
REG[0149h] bits 7-0
REG[0148h] bits 7-0 I2S DMA Buffer 0 Address bits [31:0]
These bits specify the memory start address for DMA Buffer 0. The address must be 32-bit aligned (i.e. 0, 4, 8, C,..., etc.).

Note

When the I2S Audio DMA Buffers are configured for DRAM, the performance of the I2S audio function will vary based on the other internal modules concurrently accesses DRAM. The I2S audio function can only be guaranteed if the I2S Audio DMA buffers are located in SRAM.

REG[014Ch] I2S DMA Buffer 1 Address Register 0 Default = 00h							Read/Write
I2S DMA Buffer 1 Address bits 7-0							
7	6	5	4	3	2	1	0

REG[014Dh] I2S DMA Buffer 1 Address Register 1 Default = 00h							Read/Write
I2S DMA Buffer 1 Address bits 15-8							
7	6	5	4	3	2	1	0

REG[014Eh] I2S DMA Buffer 1 Address Register 2 Default = 00h							Read/Write
I2S DMA Buffer 1 Address bits 23-16							
7	6	5	4	3	2	1	0

REG[014Fh] I2S DMA Buffer 1 Address Register 3 Default = 00h							Read/Write
I2S DMA Buffer 1 Address bits 31-24							
7	6	5	4	3	2	1	0

REG[014Fh] bits 7-0
REG[014Eh] bits 7-0
REG[014Dh] bits 7-0
REG[014Ch] bits 7-0 I2S DMA Buffer 1 Address bits [31:0]

These bits specify the memory start address for DMA Buffer 1. The address must be 32-bit aligned (i.e. 0, 4, 8, C,..., etc.).

REG[0152h] I2S DMA Buffers Size Register 0 Default = 00h							Read/Write
I2S DMA Buffers Size bits 7-0							
7	6	5	4	3	2	1	0

REG[0153h] I2S DMA Buffers Size Register 1 Default = 00h							Read/Write
I2S DMA Buffers Size bits 15-8							
7	6	5	4	3	2	1	0

REG[0153h] bits 7-0
REG[0152h] bits 7-0 I2S DMA Buffers Size bits [15:0]

These bits specify the size, in bytes, of the I2S DMA buffers (Buffer 0 and Buffer 1). The maximum size for the I2S DMA buffers is 65536 bytes and the minimum size is 4 bytes.

I2S DMA Buffer Size = (REG[0153h], REG[0152h]) + 4

Bits 1-0 of REG[0152h] should always be programmed to 00b.

REG[0154h] I2S DMA Status Register Default = 00h							Read/Write
n/a				I2S DMA Interrupt Status	n/a	I2S DMA Buffer Selection Status	n/a
7	6	5	4	3	2	1	0

bit 3 I2S DMA Interrupt Status
This bit indicates when the I2S DMA Controller has finished reading from an I2S DMA buffer and switches to reading from the other buffer. This status of this interrupt can also be read at REG[0A00h] bit 3. To enable this interrupt to the Host, set the I2S DMA Interrupt Enable bit (REG[0A06h] bit 3 = 1b).
When this bit = 0b, the I2S DMA Controller has not finished reading from an I2S DMA buffer.
When this bit = 1b, the I2S DMA Controller has finished reading from an I2S DMA buffer.

To clear this status bit, write a 1b to this bit.

bit 1 I2S DMA Buffer Selection Status
If I2S DMA is enabled (REG[0104h] bit 0 = 1b), this bit is read only and indicates which I2S DMA buffer is currently being read from.
When this bit = 0b, I2S DMA Buffer 0 is being read from.
When this bit = 1b, I2S DMA Buffer 1 is being read from.

If I2S DMA is disabled (REG[0104h] bit 0 = 0b), this bit is read/write and can be used as a general-purpose “flag” bit.

10.4.6 GPIO Registers

REG[0180h] GPIO Configuration Register 0 Default = FFh							Read/Write
GPIO7 Config	GPIO6 Config	GPIO5 Config	GPIO4 Config	GPIO3 Config	GPIO2 Config	GPIO1 Config	GPIO0 Config
7	6	5	4	3	2	1	0

REG[0181h] GPIO Configuration Register 1 Default = FFh							Read/Write
GPIO15 Config	GPIO14 Config	GPIO13 Config	GPIO12 Config	GPIO11 Config	GPIO10 Config	GPIO9 Config	GPIO8 Config
7	6	5	4	3	2	1	0

REG[0181h] bits 7-0
REG[0180h] bits 7-0 GPIO[15:0] Configuration
These bits configure each individual GPIO pin between an input or an output.
When this bit = 0b, the corresponding GPIO pin is configured as an output pin.
When this bit = 1b, the corresponding GPIO pin is configured as an input pin. (default)

REG[0182h] GPIO Status Register 0 Default = XXh							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

REG[0183h] GPIO Status Register 1 Default = XXh							Read/Write
GPIO15 Status	GPIO14 Status	GPIO13 Status	GPIO12 Status	GPIO11 Status	GPIO10 Status	GPIO9 Status	GPIO8 Status
7	6	5	4	3	2	1	0

REG[0183h] bits 7-0
REG[0182h] bits 7-0 GPIO[15:0] Status
When GPIOx is configured as an input (see REG[0180h] ~ REG[0181h]), a read from this bit returns the status of the corresponding GPIOx pin.
When GPIOx is configured as an output (see (REG[0180h] ~ REG[0181h]), writing a 1b to the bit drives the corresponding GPIOx pin high and writing a 0b to the bit drives the corresponding GPIOx pin low.

REG[0184h] GPIO Pull-down Control Register 0 Default = 00h							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

REG[0185h] GPIO Pull-down Control Register 1 Default = 00h							Read/Write
GPIO15 Pull-down Control	GPIO14 Pull-down Control	GPIO13 Pull-down Control	GPIO12 Pull-down Control	GPIO11 Pull-down Control	GPIO10 Pull-down Control	GPIO9 Pull-down Control	GPIO8 Pull-down Control
7	6	5	4	3	2	1	0

REG[0185h] bits 7-0
REG[0184h] bits 7-0 GPIO[15:0] Pull-down Control
All GPIO pins have internal pull-down resistors. These bits control the state of the pull-down resistor for each GPIOx pin.
When this bit = 0b, the pull-down resistor for the corresponding GPIOx pin is active. (default)
When this bit = 1b, the pull-down resistor for the corresponding GPIOx pin is inactive.

REG[0186h] GPIO[15:8] / Keypad Configuration Register Default = 00h							Read/Write
n/a		GPIO[15:8] / Keypad Pin Mapping Select	n/a				
7	6	5	4	3	2	1	0

bit 5 GPIO[15:8] / Keypad Pin Mapping Select
The GPIO[15:8] / Keypad interface pins can be multiplexed/mapped on either unused Host interface pins, or unused FP1 (LCD1 interface) pins. This bit selects which interface the pins are mapped to.
When this bit = 0b, the Keypad interface signals are mapped on the Host Interface pins and the GPIO[15:8] signals are mapped on the FP1 pins (see Section 5.5, “Host Interface Pin Mapping” on page 35).
When this bit = 1b, the Keypad interface signals are mapped on the FP1 pins and the GPIO[15:8] signals are mapped on the Host Interface pins (see Section 5.6, “LCD / Camera2 Pin Mapping” on page 40).

Note

GPIO7 is not available when the Keypad Interface is configured to use the FP1IO pins, REG[0186h] bit 5 = 1b.

REG[0188h] Miscellaneous Pull-up/Pull-down Register 0 Default = 00h Read/Write							
SPIDIO Pull-down Control	SDA Pull-up Control	SCL Pull-up Control	CM1CLKIN Pull-down Control	CM1FIELD Pull-down Control	CM1HREF Pull-down Control	CM1VREF Pull-down Control	CM1DAT[7:0] Pull-down Control
7	6	5	4	3	2	1	0

bit 7 SPIDIO Pull-down Control
This bit controls the state of the pull-down resistor on the Serial Flash interface data pin, SPIDIO.
When this bit = 0b, the pull-down resistor on the SPIDIO pin is active. (default)
When this bit = 1b, the pull-down resistor on the SPIDIO pin is inactive.

bit 6 SDA Pull-up Control
This bit controls the state of the pull-up resistor on the I2C interface data pin, SDA. The I2C interface is typically used for programming the cameras.
When this bit = 0b, the pull-up resistor on the SDA pin is active. (default)
When this bit = 1b, the pull-up resistor on the SDA pin is inactive.

bit 5 SCL Pull-up Control
This bit controls the state of the pull-up resistor on the I2C interface clock pin, SCL. The I2C interface is typically used for programming the cameras.
When this bit = 0b, the pull-up resistor on the SCL pin is active. (default)
When this bit = 1b, the pull-up resistor on the SCL pin is inactive.

bit 4 CM1CLKIN Pull-down Control
This bit controls the state of the pull-down resistor on the Camera1 interface clock input pin, CM1CLKIN. When the Camera1 interface is configured for 24-bit RGB 8:8:8 streaming input (REG[0D06h] bits 2-1 = 10b), the Host Interface pins (SPI 2-stream mode) are used and this bit also controls the pull-up resistor on the C1PCLKIN input pin (BS#).
When this bit = 0b, the pull-down/pull-up resistor is active. (default)
When this bit = 1b, the pull-down/pull-up resistor is inactive.

bit 3 CM1FIELD Pull-down Control
This bit controls the state of the pull-down resistor on the Camera1 interface field input pin, CM1FIELD. When the Camera1 interface is configured for 24-bit RGB 8:8:8 streaming input (REG[0D06h] bits 2-1 = 10b), the Host Interface pins (SPI 2-stream mode) are used and this bit also controls the pull-down resistor on the C1DEIN input pin (AB0).
When this bit = 0b, the pull-down resistor is active. (default)
When this bit = 1b, the pull-down resistor is inactive.

bit 2 CM1HREF Pull-down Control
This bit controls the state of the pull-down resistor on the Camera1 interface horizontal sync input pin, CM1HREF. When the Camera1 interface is configured for 24-bit RGB 8:8:8 streaming input (REG[0D06h] bits 2-1 = 10b), the Host Interface pins (SPI 2-stream mode) are used and this bit also controls the pull-down resistor on the C1HSIN input pin (AB2).
When this bit = 0b, the pull-down resistor is active. (default)
When this bit = 1b, the pull-down resistor is inactive.

bit 1 CM1VREF Pull-down Control
This bit controls the state of the pull-down resistor on the Camera1 interface vertical sync input pin, CM1VREF. When the Camera1 interface is configured for 24-bit RGB 8:8:8 streaming input (REG[0D06h] bits 2-1 = 10b), the Host Interface pins (SPI 2-stream mode) are used and this bit also controls the pull-down resistor on the C1VSIN input pin (AB1).
When this bit = 0b, the pull-down resistor is active. (default)
When this bit = 1b, the pull-down resistor is inactive.

bit 0 CM1DAT[7:0] Pull-down Control
This bit controls the state of the pull-down resistors on the Camera1 interface bi-directional data pins (CM1DAT[7:0]). When the Camera1 interface is configured for 24-bit RGB 8:8:8 streaming input (REG[0D06h] bits 2-1 = 10b), the Host Interface pins (SPI 2-stream mode) are used and this bit also controls the pull-down/pull-up resistors on the C1RINx, C1GINx, and C1BINx input pins (RD#, BE1#, DB[15:0]).
When this bit = 0b, the pull-down resistors are active. (default)
When this bit = 1b, the pull-down resistors are inactive.

REG[0189h] Miscellaneous Pull-up/Pull-down Register 1 Default = 00h							Read/Write
n/a		MEMDQ[31:0] Pull-down Control	CM2CLKIN Pull-down Control	CM2FIELD Pull-down Control	CM2HREF Pull-down Control	CM2VREF Pull-down Control	CM2DAT[7:0] Pull-down Control
7	6	5	4	3	2	1	0

bit 5 MEMDQ[31:0] Pull-down Control
This bit controls the state of the pull-down resistors on the SDRAM interface bidirectional data pins, MEMDQ[31:0].
When this bit = 0b, the pull-down resistors on the MEMDQ[31:0] pins are active. (default)
When this bit = 1b, the pull-down resistors on the MEMDQ[31:0] pins are inactive.

bit 4 CM2CLKIN Pull-down Control
This bit controls the state of the pull-down resistor on the Camera2 interface clock input pin, CM2CLKIN (FP1IO8). When the Camera2 interface is configured for 24-bit RGB 8:8:8 streaming input (REG[0D46h] bits 2-1 = 10b), this bit also controls the pull-down resistor on the C2PCLKIN input pin (FP1IO23).
When this bit = 0b, the pull-down resistor is active. (default)
When this bit = 1b, the pull-down resistor is inactive.

bit 3 CM2FIELD Pull-down Control
This bit controls the state of the pull-down resistor on the Camera2 interface field input pin, CM2FIELD (FP1IO10). When the Camera2 interface is configured for 24-bit RGB 8:8:8 streaming input (REG[0D46h] bits 2-1 = 10b), this bit also controls the pull-down resistor on the C2DEIN input pin (FP1IO22).
When this bit = 0b, the pull-down resistor is active. (default)
When this bit = 1b, the pull-down resistor is inactive.

bit 2 CM2HREF Pull-down Control
This bit controls the state of the pull-down resistor on the Camera2 interface horizontal sync input pin, CM2HREF (FP1IO13). When the Camera2 interface is configured for 24-bit RGB 8:8:8 streaming input (REG[0D46h] bits 2-1 = 10b), this bit also controls the pull-down resistor on the C2HSIN input pin (FP1IO20).
When this bit = 0b, the pull-down resistor is active. (default)
When this bit = 1b, the pull-down resistor is inactive.

bit 1 CM2VREF Pull-down Control
This bit controls the state of the pull-down resistor on the Camera2 interface vertical sync input pin, CM2VREF (FP1IO12). When the Camera2 interface is configured for 24-bit RGB 8:8:8 streaming input (REG[0D46h] bits 2-1 = 10b), this bit also controls the pull-down resistor on the C2VSIN input pin (FP1IO21).
When this bit = 0b, the pull-down resistor is active. (default)
When this bit = 1b, the pull-down resistor is inactive.

bit 0 CM2DAT[7:0] Pull-down Control
This bit controls the state of the pull-down resistors on the Camera2 interface bidirectional data pins, CM2DAT[7:0] (FP1IO[7:0]). When the Camera2 interface is configured for 24-bit RGB 8:8:8 streaming input (REG[0D46h] bits 2-1 = 10b), this bit also controls the pull-down resistors on the C2RINx, C2GINx, and C2BINx input pins (FP1IO[17:0]).
When this bit = 0b, the pull-down resistors are active. (default)
When this bit = 1b, the pull-down resistors are inactive.

10.4.7 Keypad Registers

The Keypad Interface scans for key presses using up to a 5x5 matrix. Each row, column input coordinate is associated with an interrupt which has independent enable, input polarity select, and status/clear controls. If a keypad smaller than 5x5 is used, the interrupt number associated with the coordinate does not change.

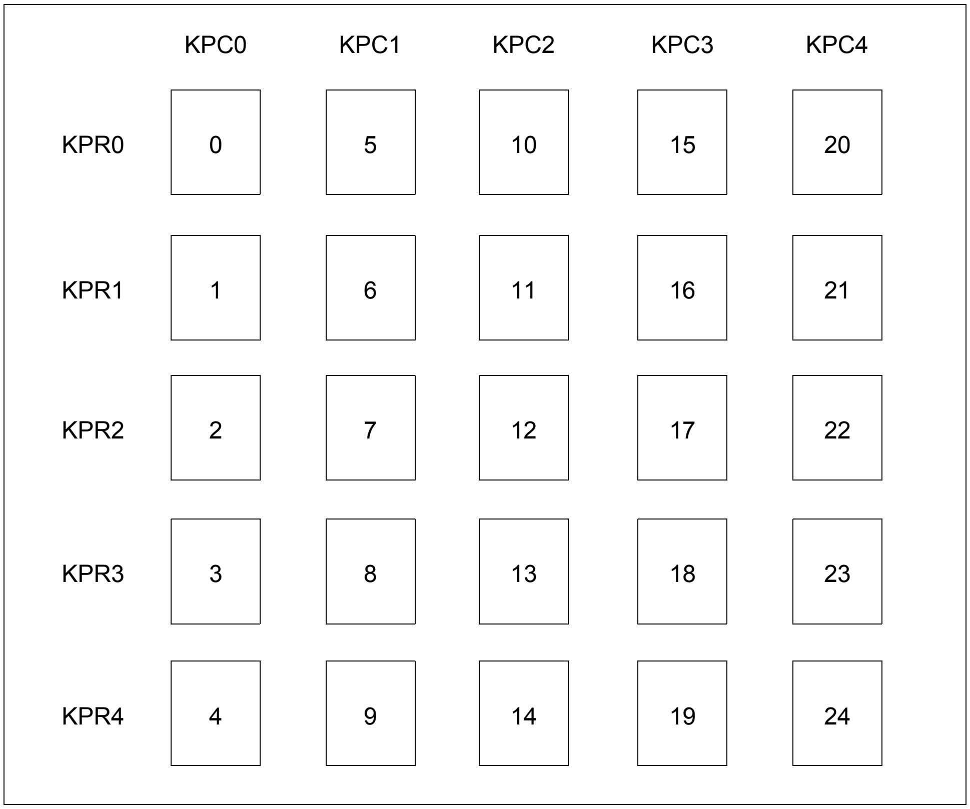


Figure 10-2: Keypad Interface Example

REG[01C0h] Keypad Control Register Default = 00h							Read/Write
n/a						Keypad Filter Enable	Keypad Enable
7	6	5	4	3	2	1	0

bit 1 Keypad Filter Enable
This bit controls glitch filtering for the keypad interface input pins (KPR[4:0] and KPC[4:0]). The sampling period for the filter is controlled using REG[01CCh] ~ REG[01CEh].
When this bit = 0b, the keypad filter is disabled.
When this bit = 1b, the keypad filter is enabled.

bit 0 Keypad Enable
This bit controls glitch filtering for the keypad interface input pins (KPR[4:0] and KPC[4:0]). The sampling clock period for the filter is controlled using REG[01CCh] ~ REG[01CEh]. For detailed timing information, see Section 7.10, “Keypad Interface Timing” on page 126.
When this bit = 0b, the keypad filter is disabled.
When this bit = 1b, the keypad filter is enabled.

Note
After enabling the keypad, all interrupts in REG[01C4h] ~ REG[01C7h] should be cleared before enabling the Keypad Host Interrupt (REG[A08] bit 4 = 1b)

REG[01C4h] Keypad Interrupt Enable Register 0 Default = 00h							Read/Write
Keypad Interrupt 7 Enable	Keypad Interrupt 6 Enable	Keypad Interrupt 5 Enable	Keypad Interrupt 4 Enable	Keypad Interrupt 3 Enable	Keypad Interrupt 2 Enable	Keypad Interrupt 1 Enable	Keypad Interrupt 0 Enable
7	6	5	4	3	2	1	0

REG[01C5h] Keypad Interrupt Enable Register 1 Default = 00h							Read/Write
Keypad Interrupt 15 Enable	Keypad Interrupt 14 Enable	Keypad Interrupt 13 Enable	Keypad Interrupt 12 Enable	Keypad Interrupt 11 Enable	Keypad Interrupt 10 Enable	Keypad Interrupt 9 Enable	Keypad Interrupt 8 Enable
7	6	5	4	3	2	1	0

REG[01C6h] Keypad Interrupt Enable Register 2 Default = 00h							Read/Write
Keypad Interrupt 23 Enable	Keypad Interrupt 22 Enable	Keypad Interrupt 21 Enable	Keypad Interrupt 20 Enable	Keypad Interrupt 19 Enable	Keypad Interrupt 18 Enable	Keypad Interrupt 17 Enable	Keypad Interrupt 16 Enable
7	6	5	4	3	2	1	0

REG[01C7h] Keypad Interrupt Enable Register 3 Default = 00h							Read/Write
n/a							Keypad Interrupt 24 Enable
7	6	5	4	3	2	1	0

REG[01C7h] bit 0
REG[01C6h] bits 7-0
REG[01C5h] bits 7-0
REG[01C4h] bits 7-0 Keypad Interrupt [24:0] Enable

These bits control Keypad Interrupts 24-0 and determine if a Keypad Interrupt occurs in REG[0A02h] bit 4. Each keypad interrupt is associated with a specific row, column coordinate as shown in Figure 10-2: "Keypad Interface Example" on page 197. The status of each interrupt is indicated in REG[01D0h] ~ REG[01D3h] and the polarity of each interrupt can be changed using REG[01C8h] ~ REG[01CBh].
When this bit = 0b, Keypad Interrupt X is disabled.
When this bit = 1b, Keypad Interrupt X is enabled.

REG[01C8h] Keypad Input Polarity Register 0 Default = 00h							Read/Write
Keypad Input 7 Polarity Select	Keypad Input 6 Polarity Select	Keypad Input 5 Polarity Select	Keypad Input 4 Polarity Select	Keypad Input 3 Polarity Select	Keypad Input 2 Polarity Select	Keypad Input 1 Polarity Select	Keypad Input 0 Polarity Select
7	6	5	4	3	2	1	0

REG[01C9h] Keypad Input Polarity Register 1 Default = 00h							Read/Write
Keypad Input 15 Polarity Select	Keypad Input 14 Polarity Select	Keypad Input 13 Polarity Select	Keypad Input 12 Polarity Select	Keypad Input 11 Polarity Select	Keypad Input 10 Polarity Select	Keypad Input 9 Polarity Select	Keypad Input 8 Polarity Select
7	6	5	4	3	2	1	0

REG[01CAh] Keypad Input Polarity Register 2 Default = 00h							Read/Write
Keypad Input 23 Polarity Select	Keypad Input 22 Polarity Select	Keypad Input 21 Polarity Select	Keypad Input 20 Polarity Select	Keypad Input 19 Polarity Select	Keypad Input 18 Polarity Select	Keypad Input 17 Polarity Select	Keypad Input 16 Polarity Select
7	6	5	4	3	2	1	0

REG[01CBh] Keypad Input Polarity Register 3 Default = 00h							Read/Write
n/a							Keypad Input 24 Polarity Select
7	6	5	4	3	2	1	0

REG[01CBh] bit 0
REG[01CAh] bits 7-0
REG[01C9h] bits 7-0
REG[01C8h] bits 7-0

Keypad Input [24:0] Polarity Select
These bits specify the polarity for Keypad inputs 24-0. Each keypad input is associated with a specific row, column coordinate as shown in Figure 10-2: “Keypad Interface Example” on page 197.
When this bit = 0b, the polarity of Keypad Input X is inverted and will cause the corresponding Keypad Interrupt to occur, if enabled, when the key is released.
When this bit = 1b, the polarity of Keypad Input X is normal and will cause the corresponding Keypad Interrupt to occur, if enabled, when the key is pressed.

Note

1. These bits should only be changed when the keypad is disabled (REG[01C0h] bit 0 = 0b).
2. When a Keypad Input Polarity bit is changed from 1b to 0b, 2 keypad sampling clocks must take place before clearing the corresponding Interrupt Status bit in REG[01D0h] ~ REG[01D3h].

REG[01CCh] Keypad Filter Sampling Period Register 0 Default = 00h							Read/Write
Keypad Filter Sampling Period bits 7-0							
7	6	5	4	3	2	1	0

REG[01CDh] Keypad Filter Sampling Period Register 1 Default = 00h							Read/Write
Keypad Filter Sampling Period bits 15-8							
7	6	5	4	3	2	1	0

REG[01CEh] Keypad Filter Sampling Period Register 2 Default = 00h							Read/Write
n/a				Keypad Filter Sampling Period bits 19-16			
7	6	5	4	3	2	1	0

REG[01CEh] bits 3-0
REG[01CDh] bits 7-0
REG[01CCh] bits 7-0 Keypad Filter Sampling Period [19:0]

When the Keypad Filter is enabled (REG[01C0h] bit 1 = 1b), these bits specify the sampling clock period for the keypad input glitch filter. The value in these bits specifies the number of keypad clocks between each sample of the keypad input pins. For detailed timing information, see Keypad Interface Timing section.

The keypad clock is derived from the input clock INCLK1 which is sourced from either CLKI or OSCI. The keypad clock can be further divided using the Keypad Clock Divide Select bits, REG[01D4h] ~ REG[01D5h].

When the Keypad Filter is enabled, these bits should be set according to the following formula.

$$\text{Filter sampling period} = \frac{\text{Minimum Key Press Time}}{\text{Keypad Clock Period} \times \text{number of clocks per column} \times \text{number of columns}}$$

Where:

Filter sampling period is defined by REG[01CCh] ~ REG[01CEh]
Minimum Key Press Time is the shortest key press that will be detected
Keypad Clock Period is defined by REG[01D4h] ~ REG[01D5h]
number of clocks per column is 4
number of columns is 5

For example, use the following calculation to detect a minimum keypress of 10ms for a keypad clock period of 30us (32.68KHz).

Filter Sampling Period = 10000us / (30us x 4 x 5)
= 10000 / 600
= 16.667
= 16

REG[01D0h] Keypad Interrupt Raw Status/Clear Register 0 Default = 00h Read/Write							
Keypad Interrupt 7 Raw Status/ Clear	Keypad Interrupt 6 Raw Status/ Clear	Keypad Interrupt 5 Raw Status/ Clear	Keypad Interrupt 4 Raw Status/ Clear	Keypad Interrupt 3 Raw Status/ Clear	Keypad Interrupt 2 Raw Status/ Clear	Keypad Interrupt 1 Raw Status/ Clear	Keypad Interrupt 0 Raw Status/ Clear
7	6	5	4	3	2	1	0

REG[01D1h] Keypad Interrupt Raw Status/Clear Register 1 Default = 00h Read/Write							
Keypad Interrupt 15 Raw Status/ Clear	Keypad Interrupt 14 Raw Status/ Clear	Keypad Interrupt 13 Raw Status/ Clear	Keypad Interrupt 12 Raw Status/ Clear	Keypad Interrupt 11 Raw Status/ Clear	Keypad Interrupt 10 Raw Status/ Clear	Keypad Interrupt 9 Raw Status/ Clear	Keypad Interrupt 8 Raw Status/ Clear
7	6	5	4	3	2	1	0

REG[01D2h] Keypad Interrupt Raw Status/Clear Register 2 Default = 00h Read/Write							
Keypad Interrupt 23 Raw Status/ Clear	Keypad Interrupt 22 Raw Status/ Clear	Keypad Interrupt 21 Raw Status/ Clear	Keypad Interrupt 20 Raw Status/ Clear	Keypad Interrupt 19 Raw Status/ Clear	Keypad Interrupt 18 Raw Status/ Clear	Keypad Interrupt 17 Raw Status/ Clear	Keypad Interrupt 16 Raw Status/ Clear
7	6	5	4	3	2	1	0

REG[01D3h] Keypad Interrupt Raw Status/Clear Register 3 Default = 00h Read/Write							
n/a							Keypad Interrupt 24 Raw Status/ Clear
7	6	5	4	3	2	1	0

REG[01D3h] bits 7-0
REG[01D2h] bits 7-0
REG[01D1h] bits 7-0
REG[01D0h] bit 0 Keypad Interrupt [24:0] Raw Status/Clear

For Reads:
These bits indicate the raw status of the corresponding Keypad Interrupt, regardless of whether or not the corresponding Keypad Interrupt is enabled (see REG[01C4h] ~ REG[01C7h]). These bits indicate the status of the keypad interrupt associated with a specific row, column coordinate as shown in Figure 10-2: “Keypad Interface Example” on page 183. These bits are not masked by the Keypad Interrupt [24:0] Enable bits in REG[01C4h] ~ REG[01C7h].
When this bit = 0b, Keypad Interrupt X has not occurred.
When this bit = 1b, Keypad Interrupt X has occurred which indicates that the corresponding key has been pressed/released according to the setting of the Keypad Input [24:0] Polarity Select bits (REG[01C8h] ~ REG[01CBh]).
For Writes:
Writing a 0b to this bit has no effect.
Writing a 1b then 0b to this bit clears the interrupt status.

Note
After enabling the keypad, all interrupts in REG[01C4h] ~ REG[01C7h] should be cleared before enabling the Keypad Host Interrupt (REG[A08] bit 4 = 1b)

REG[01D4h] Keypad Clock Configuration Register 0 Default = 00h Read/Write							
Keypad Clock Divide Select bits 7-0							
7	6	5	4	3	2	1	0

REG[01D5h] Keypad Clock Configuration Register 1 Default = 00h Read/Write							
n/a				Keypad Clock Divide Select bits 11-8			
7	6	5	4	3	2	1	0

REG[01D5h] bits 3-0
REG[01D4h] bits 7-0 Keypad Clock Divide Select [11:0]

These bits specify the clock divide ratio for the keypad clock. The keypad clock is derived from the input clock INCLK1 which is sourced from either CLKI or OSCI. For details, see Chapter 9, "Clocks" on page 131. The keypad clock divide ratio is calculated using the following formula.

Keypad Clock Divide Ratio = 1: (REG[01D5h] bits 3-0, REG[01D4h] bits 7-0) + 1

REG[01D6h] Keypad GPI Function Enable Register Default = 00h Read/Write							
n/a			Keypad GPI Function Enable bits 4-0				
7	6	5	4	3	2	1	0

bits 4-0 Keypad GPI Function Enable bits [4:0]

The keypad interface row pins (KPR[4:0]) can be configured as general purpose input pins which can generate edge-trigger interrupts. These bits control the GPI function for each corresponding KPR[4:0] pin. When configured as GPI pins, the status of each associated interrupt is indicated by REG[01D0h] bits 4-0 and the polarity of each interrupt can be controlled using REG[01C8h] bits 4-0. If the filter function is enabled (REG[01C0h] bit 1 = 1b), an interrupt is generated only when two consecutive samples (as controlled by the Keypad Filter Sampling Period bits in REG[01CCh] ~ REG[01CEh]) are the same.

When this bit = 0b, the corresponding KPR[4:0] pin functions as a scan input pin for the keypad interface.

When this bit = 1b, the corresponding KPR[4:0] pin functions as a general purpose input which can generate edge-trigger interrupts.

Note

If bit 0 = 1b, Keypad Interrupts 5, 10, 15, 20 are disabled.
If bit 1 = 1b, Keypad Interrupts 6, 11, 16, 21 are disabled.
If bit 2 = 1b, Keypad Interrupts 7, 12, 17, 22 are disabled.
If bit 3 = 1b, Keypad Interrupts 8, 13, 18, 23 are disabled.
If bit 4 = 1b, Keypad Interrupts 9, 14, 19, 24 are disabled.

10.4.8 PWM Registers

REG[0200h] PWM Control Register Default = 00h							Read/Write
PWM Rate bits 2-0			PWM Output Polarity	PWM Logic Clock Divide Select bits 3-0			
7	6	5	4	3	2	1	0

bits 7-5 PWM Rate bits [2:0]

These bits determine the M value used for slope calculations. These bits determine the rate (M value) at which the duty cycle of the Pulse Cycles is increased/decreased during duty cycle ramp-up/ramp-down. During ramp-up/ramp-down of the duty cycle of Pulse Cycles, the duty cycle is increased/decreased by a value (1/16 x N), where N is determined by the corresponding PWM1/PWM2 Slope bits (see REG[0203h] bits 7-4 or REG[0206h] bits 7-4), every M Pulse Clock cycles. These bits have no effect when the Slope bits are set to 0. These bits have no effect when the Slope bits are set to 0.

REG[0200h] bits 7-5 = M - 1

bit 4 PWM Output Polarity

This bit specifies the polarity of the PWM1/PWM2 outputs pin relative to the digital value output by the PWM circuit.

When this bit = 0b, the PWM outputs are normal which means that the PWM1/PWM2 pin voltage is driven low when a logic 1 is driven from the PWM circuit and driven high when a logic 0 is driven from the PWM circuit.

When this bit = 1b, the PWM outputs are inverted which means that the PWM1/PWM2 pin voltage is driven high when a logic 1 is driven from the PWM circuit and driven low when a logic 0 is driven from the PWM circuit.

bits 3-0 PWM Logic Clock Divide Select bits [3:0]
These bits specify the divide ratio used to generate the PWM Logic Clock which is used to drive the PWM circuits. The PWM Logic Clock is derived from the internal PWM Source Clock (PWMSRCCLK) which is sourced from SYSCLK and is configured using the PWMSRCCLK Divide Select bits (REG[0034h] ~ REG[0035h]). For further details on PWMSRCCLK, see Section Chapter 9, “Clocks” on page 131.

Table 10-27: PWM Logic Clock Divide Selection

REG[0200h] bits 3-0	PWM Logic Clock Divide Ratio
0000b	1:1
0001b	2:1
0010b	4:1
0011b	6:1
0100b	8:1
0101b	10:1
0110b	12:1
0111b	14:1
1000b	16:1
1001b ~ 1111b	Reserved (PWM Logic Clock is stopped)

Note

BOTH PWM1 and PWM2 must be disabled when bits [3:0] are changed, then re-enabled.

REG[0201h] PWM1 Enable/On Register Default = 00h							Read/Write
PWM1 Enable	PWM1 On Time bits 6-0						
7	6	5	4	3	2	1	0

bit 7 PWM1 Enable
This bit controls PWM1 output.
When this bit = 0b, PWM1 output is disabled (becomes logic 0 before the PWM Output Polarity specified by REG[0200h] bit 4 is applied).
When this bit = 1b, PWM1 output is enabled.

bits 6-0 PWM1 On Time bits [6:0]
These bits specify the point at which the PWM1 LED turns “on” relative to the start of the 128 clock pulse cycle. A value of 0 means the LED starts the turn on sequence immediately at the start of the 128 clock cycle. For further information on using PWM, see Section Chapter 19, “Pulse Width Modulation (PWM)” on page 512.

REG[0202h] PWM1 Off Register Default = 00h							Read/Write
n/a	PWM1 Off Time bits 6-0						
7	6	5	4	3	2	1	0

bits 6-0 PWM1 Off Time bits [6:0]
These bits specify the point at which the PWM1 LED turns “off” relative to the start of the 128 clock pulse cycle. This value must be greater than the PWM1 On Duration specified in REG[0201h] bits 6-0. For further information on using PWM, see Section Chapter 19, “Pulse Width Modulation (PWM)” on page 512.

REG[0202h] bits 6-0 = PWM1 Off Duration - 1

Note

If a value of 7Fh is specified, the LED is on for the entire duration of the PWM1 duty cycle, REG[0203h] bits 3-0.

REG[0203h] PWM1 Control Register Default = 00h							Read/Write
PWM1 Slope bits 3-0				PWM1 Maximum Duty Cycle bits 3-0			
7	6	5	4	3	2	1	0

bits 7-4 PWM1 Slope bits [3:0]
Within each Repeat Cycle consisting of 128 Pulse Clocks, the PWM1 output can start from completely off (0/16 duty cycle), ramp up to the maximum duty cycle specified REG[0203h] bits 3-0, and then ramp down back to completely off. The rate in which the duty cycle is incremented/decremented during ramp-up/ramp-down is determined by REG[0200h] bits 7-5 and for each increment/decrement step the duty cycle is increased/decreased by (1/16 x N) where N is the decimal value represented by these bits. If these bits are set to 0h, the duty cycle immediately changes from completely off, to the maximum duty cycle as specified by the PWM1 Maximum Duty Cycle bits.

bits 3-0 PWM1 Maximum Duty Cycle bits [3:0]
These bits specify the "full on" duty cycle for PWM1 which determines the maximum brightness that the LED reaches at the peak of the pulse. A value of Fh indicates full brightness (i.e. continuously on). A value of 0h means the LED is on for 1/16th of the time.

Note
When the PWM1 Slope (REG[0203h] bits 7-4) is non-zero, the PWM1 Duty Cycle must not be set to 1111b (Fh).

REG[0204h] PWM2 Enable/On Register Default = 00h							Read/Write
PWM2 Enable	PWM2 On Time bits 6-0						
7	6	5	4	3	2	1	0

bit 7 PWM2 Enable
This bit controls PWM2 output.
When this bit = 0b, PWM2 output is disabled (becomes logic 0 before the PWM Output Polarity specified by REG[0200h] bit 4 is applied).
When this bit = 1b, PWM2 output is enabled.

bits 6-0 PWM2 On Time bits [6:0]
These bits specify the point at which the PWM2 LED turns "on" relative to the start of the 128 clock pulse cycle. A value of 0 means the LED starts the turn on sequence immediately at the start of the 128 clock cycle. For further information on using PWM, see Section Chapter 19, "Pulse Width Modulation (PWM)" on page 512.

REG[0205h] PWM2 Off Register Default = 00h							Read/Write
n/a	PWM2 Off Time bits 6-0						
7	6	5	4	3	2	1	0

bits 6-0 PWM2 Off Time bits [6:0]
These bits specify the point at which the PWM2 LED turns "off" relative to the start of the 128 clock pulse cycle. This value must be greater than the PWM2 On Duration specified in REG[0204h] bits 6-0. For further information on using PWM, see Section Chapter 19, "Pulse Width Modulation (PWM)" on page 512.
REG[0205h] bits 6-0 = PWM2 Off Duration - 1

Note
If a value of 7Fh is specified, the LED is on for the entire duration of the PWM2 duty cycle, REG[0206h] bits 3-0.

REG[0206h] PWM2 Control Register Default = 00h							Read/Write
PWM2 Slope bits 3-0				PWM2 Maximum Duty Cycle bits 3-0			
7	6	5	4	3	2	1	0

bits 7-4 PWM2 Slope bits [3:0]
Within each Repeat Cycle consisting of 128 Pulse Clocks, the PWM2 output can start from completely off (0/16 duty cycle), ramp up to the maximum duty cycle specified REG[0206h] bits 3-0, and then ramp down back to completely off. The rate in which the duty cycle is incremented/decremented during ramp-up/ramp-down is determined by REG[0200h] bits 7-5 and for each increment/decrement step the duty cycle is increased/decreased by (1/16 x N) where N is the decimal value represented by these bits. If these bits are set to 0h, the duty cycle immediately changes from completely off, to the maximum duty cycle as specified by the PWM2 Maximum Duty Cycle bits.

bits 3-0 PWM2 Maximum Duty Cycle bits [3:0]
These bits specify the “full on” duty cycle for PWM2 which determines the maximum brightness that the LED reaches at the peak of the pulse. A value of Fh indicates full brightness (i.e. continuously on). A value of 0h means the LED is on for 1/16th of the time.

Note
When the PWM2 Slope (REG[0206h] bits 7-4) is non-zero, the PWM2 Duty Cycle must not be set to 1111b (Fh).

10.4.9 SDRAM Read/Write Buffer Registers

SDRAM Buffer 0 and SDRAM Buffer 1 are designed to work together so that while one buffer is busy (the SDRAM Buffer 0/1 Start bit = 1b), the other buffer can also be started. The second buffer will wait for the first buffer to complete the transfer, and then will start the next transfer. The SDRAM Buffers are 128 bytes in size. See Section Chapter 18, "SDRAM Read/Write Buffer" on page 507 for further information.

Note
When using SPI, I2C, or any interface without WAIT, SDRAM must be accessed using the SDRAM Read/Write Buffers.

REG[0240h] SDRAM Buffer 0 Configuration Register Default = 00h							Read/Write
n/a						SDRAM Buffer 0 Done Interrupt Enable	SDRAM Buffer 0 Mode
7	6	5	4	3	2	1	0

bit 1 SDRAM Buffer 0 Done Interrupt Enable
This bit controls whether the SDRAM Buffer 0 Done Interrupt can generate an interrupt request (see also REG[0A06h] bit 5). The status of this interrupt is indicated by the SDRAM Buffer 0 Done Status/Clear bit, REG[0242h] bit 3.
When this bit = 0b, the interrupt is disabled.
When this bit = 1b, the interrupt is enabled.

bit 0 SDRAM Buffer 0 Mode
This bit selects whether SDRAM Buffer 0 is used for reading from or writing to the SDRAM.
When this bit = 0b, SDRAM Buffer 0 is used for writing data from the Host to SDRAM.
When this bit = 1b, SDRAM Buffer 0 is used for reading data from SDRAM to the Host.

Note
When the SDRAM Buffer 0 mode is changed between read and write mode, the Read/Write Buffer FIFO is reset.

REG[0242h] SDRAM Buffer 0 Control Register Default = 00h							Read/Write
n/a				SDRAM Buffer 0 Done Interrupt Status/Clear	SDRAM Buffer 0 Rectangular Increment (WO)	SDRAM Buffer 0 Load Address (WO)	SDRAM Buffer 0 Start
7	6	5	4	3	2	1	0

bit 3 SDRAM Buffer 0 Done Interrupt Status/Clear
This bit indicates the status of the SDRAM Buffer 0 Done Interrupt which occurs when a transfer between SDRAM Buffer 0 and the SDRAM has finished. This interrupt can generate an interrupt request when the SDRAM Buffer 0 Done Interrupt Enable bit (REG[0240h] bit 1) and the SDRAM Read/Write Buffers Interrupt Enable bit (REG[0A06h] bit 5) are set to 1b.
When this bit = 0b, a SDRAM Buffer 0 Done Interrupt has not occurred.
When this bit = 1b, a SDRAM Buffer 0 Done Interrupt has occurred.

To clear this status bit, write a 1b to this bit.

bit 2 SDRAM Buffer 0 Rectangular Increment (Write Only)
This bit determines the type of address increment done to the SDRAM Buffer 0 Target Address (REG[0248h] ~ REG[024Bh]) at the completion of a SDRAM Buffer 0 transfer. This bit should be set at the same time as the SDRAM Buffer 0 Start bit (REG[0242h] bit 0) is set.
Writing a 0b to this bit selects linear address incrementing.
Writing a 1b to this bit selects rectangular address incrementing. The rectangular increment value is specified by the SDRAM Buffer 0 Rectangular Increment Value bits (see REG[0260h] ~ REG[0261h]).

bit 1 SDRAM Buffer 0 Load Address (Write Only)
This bit determines whether the SDRAM Buffer 0 Target Address (REG[0248h] ~ REG[024Bh]) is loaded before starting a SDRAM Buffer 0 transfer. This bit should be set at the same time as the SDRAM Buffer 0 Start bit (REG[0242h] bit 0) is set.
Writing a 0b to this bit causes the SDRAM Buffer 0 Target Address to be ignored and the SDRAM Buffer 0 transfer uses the current value of the internal target address register (resulting from the end of the previous transfer).
Writing a 1b to this bit causes the SDRAM Buffer 0 Target Address (REG[0248h] ~ REG[024Bh]) to be loaded before starting the SDRAM Buffer 0 transfer.

bit 0 SDRAM Buffer 0 Start
This bit starts a transfer between SDRAM Buffer 0 and the SDRAM. The type of transfer (SDRAM Buffer 0 to SDRAM or SDRAM to SDRAM Buffer 0) is determined by the SDRAM Buffer 0 Mode bit, REG[0240h] bit 0. If necessary, the SDRAM Buffer 0 Rectangular Increment and SDRAM Buffer 0 Load Address bits should be set at the same time as this bit.
Writing a 0b to this bit has no effect.
Writing a 1b to this bit starts a transfer between SDRAM Buffer 0 and the SDRAM. This bit remains at 1b during the transfer, and returns to 0b when the transfer completes.

REG[0244h] SDRAM Buffer 0 Read Bytes Register Default = 00h							Read/Write
SDRAM Buffer 0 Read Bytes bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 — SDRAM Buffer 0 Read Bytes bits [7:0]
These bits specify the number of bytes to read when the SDRAM Buffer 0 Mode is set for read mode, REG[0240h] bit 0 = 1b. The minimum value is 0 and the maximum value is 128.

REG[0248h] SDRAM Buffer 0 Target Address Register 0 Default = 00h							Read/Write
SDRAM Buffer 0 Target Address bits 7-0							
7	6	5	4	3	2	1	0

REG[0249h] SDRAM Buffer 0 Target Address Register 1 Default = 00h							Read/Write
SDRAM Buffer 0 Target Address bits 15-8							
7	6	5	4	3	2	1	0

REG[024Ah] SDRAM Buffer 0 Target Address Register 2 Default = 00h							Read/Write
SDRAM Buffer 0 Target Address bits 23-16							
7	6	5	4	3	2	1	0

REG[024Bh] SDRAM Buffer 0 Target Address Register 3 Default = 00h							Read/Write
SDRAM Buffer 0 Target Address bits 31-24							
7	6	5	4	3	2	1	0

REG[024Bh] bits 7-0
REG[024Ah] bits 7-0
REG[0249h] bits 7-0
REG[0248h] bits 7-0 — SDRAM Buffer 0 Target Address bits [31:0]
These bits specify the target address in SDRAM for transfers between SDRAM Buffer 0 and the SDRAM. These bits are automatically incremented at the end of a read/write operation (when REG[0242h] bit 0 returns to 0b) according to the setting of the SDRAM Buffer 0 Rectangular Increment bit, REG[0242h] bit 2.

REG[024Ch] SDRAM Buffer 0 Data Port Register 0 Default = 00h							Read/Write
SDRAM Buffer 0 Data Port bits 7-0							
7	6	5	4	3	2	1	0

REG[024Dh] SDRAM Buffer 0 Data Port Register 1 Default = 00h							Read/Write
SDRAM Buffer 0 Data Port bits 15-8							
7	6	5	4	3	2	1	0

REG[024Dh] bits 7-0
REG[024Ch] bits 7-0 SDRAM Buffer 0 Data Port bits [15:0]

These bits are the data port where the Host reads from or writes to SDRAM Buffer 0. These registers are also "aliased" in the range REG[0300h] ~ REG[037Fh]. For example, writing to REG[0318h] is the same as writing to REG[024Ch]. The purpose of this "aliased" address range is for Direct host interfaces with "burst" mode which have incrementing addresses.

When the host interface is 16-bit and both byte and 16-bit word accesses of the SDRAM Buffer port is desired, an even number of byte accesses are required before a 16-bit word access is possible.

Note

When using SPI for SDRAM read accesses, the number of bytes specified by the SDRAM Buffer 0 Read Bytes bits (REG[0244h] bits 7-0) must be read from this port without interruption.

REG[0250h] SDRAM Buffer 1 Configuration Register Default = 00h						Read/Write	
n/a						SDRAM Buffer 1 Done Interrupt Enable	SDRAM Buffer 1 Mode
7	6	5	4	3	2	1	0

bit 1 SDRAM Buffer 1 Done Interrupt Enable
This bit controls whether the SDRAM Buffer 1 Done Interrupt can generate an interrupt request (see also REG[0A06h] bit 5). The status of this interrupt is indicated by the SDRAM Buffer 1 Done Status/Clear bit, REG[0252h] bit 3.
When this bit = 0b, the interrupt is disabled.
When this bit = 1b, the interrupt is enabled.

bit 0 SDRAM Buffer 1 Mode
This bit selects whether SDRAM Buffer 1 is used for reading from or writing to the SDRAM.
When this bit = 0b, SDRAM Buffer 1 is used for writing data from the Host to SDRAM.
When this bit = 1b, SDRAM Buffer 1 is used for reading data from SDRAM to the Host.

Note
When the SDRAM Buffer 1 mode is changed between read and write mode, the Read/Write Buffer FIFO is reset.

REG[0252h] SDRAM Buffer 1 Control Register Default = 00h							Read/Write
n/a				SDRAM Buffer 1 Done Interrupt Status/Clear	SDRAM Buffer 1 Rectangular Increment (WO)	SDRAM Buffer 1 Load Address (WO)	SDRAM Buffer 1 Start
7	6	5	4	3	2	1	0

bit 3 SDRAM Buffer 1 Done Interrupt Status/Clear
This bit indicates the status of the SDRAM Buffer 1 Done Interrupt which occurs when a transfer between SDRAM Buffer 1 and the SDRAM has finished. This interrupt can generate an interrupt request when the SDRAM Buffer 1 Done Interrupt Enable bit (REG[0250h] bit 1) and the SDRAM Read/Write Buffers Interrupt Enable bit (REG[0A06h] bit 5) are set to 1b.
When this bit = 0b, a SDRAM Buffer 1 Done Interrupt has not occurred.
When this bit = 1b, a SDRAM Buffer 1 Done Interrupt has occurred.

To clear this status bit, write a 1b to this bit.

bit 2 SDRAM Buffer 1 Rectangular Increment (Write Only)
This bit determines the type of address increment done to the SDRAM Buffer 1 Target Address (REG[0258h] ~ REG[025Bh]) at the completion of a SDRAM Buffer 1 transfer. This bit should be set at the same time as the SDRAM Buffer 1 Start bit (REG[0252h] bit 0) is set.
Writing a 0b to this bit selects linear address incrementing.
Writing a 1b to this bit selects rectangular address incrementing. The rectangular increment value is specified by the SDRAM Buffer 1 Rectangular Increment Value bits (see REG[0262h] ~ REG[0263h]).

bit 1 SDRAM Buffer 1 Load Address (Write Only)
This bit determines whether the SDRAM Buffer 1 Target Address (REG[0258h] ~ REG[025Bh]) is loaded before starting a SDRAM Buffer 1 transfer. This bit should be set at the same time as the SDRAM Buffer 1 Start bit (REG[0252h] bit 0) is set.
Writing a 0b to this bit causes the SDRAM Buffer 1 Target Address to be ignored and the SDRAM Buffer 1 transfer uses the current value of the internal target address register (resulting from the end of the previous transfer).
Writing a 1b to this bit causes the SDRAM Buffer 1 Target Address (REG[0258h] ~ REG[025Bh]) to be loaded before starting the SDRAM Buffer 1 transfer.

bit 0 SDRAM Buffer 1 Start
This bit starts a transfer between SDRAM Buffer 1 and the SDRAM. The type of transfer (SDRAM Buffer 1 to SDRAM or SDRAM to SDRAM Buffer 1) is determined by the SDRAM Buffer 1 Mode bit, REG[0250h] bit 0. If necessary, the SDRAM Buffer 1 Rectangular Increment and SDRAM Buffer 1 Load Address bits should be set at the same time as this bit.
Writing a 0b to this bit has no effect.
Writing a 1b to this bit starts a transfer between SDRAM Buffer 1 and the SDRAM. This bit remains at 1b during the transfer, and returns to 0b when the transfer completes.

REG[0254h] SDRAM Buffer 1 Read Bytes Register Default = 00h							Read/Write
SDRAM Buffer 1 Read Bytes bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 SDRAM Buffer 1 Read Bytes bits [7:0]
These bits specify the number of bytes to read when the SDRAM Buffer 1 Mode is set for read mode, REG[0250h] bit 0 = 1b. The minimum value is 0 and the maximum value is 128.

REG[0258h] SDRAM Buffer 1 Target Address Register 0 Default = 00h							Read/Write
SDRAM Buffer 1 Target Address bits 7-0							
7	6	5	4	3	2	1	0

REG[0259h] SDRAM Buffer 1 Target Address Register 1 Default = 00h							Read/Write
SDRAM Buffer 1 Target Address bits 15-8							
7	6	5	4	3	2	1	0

REG[025Ah] SDRAM Buffer 1 Target Address Register 2 Default = 00h							Read/Write
SDRAM Buffer 1 Target Address bits 23-16							
7	6	5	4	3	2	1	0

REG[025Bh] SDRAM Buffer 1 Target Address Register 3 Default = 00h							Read/Write
SDRAM Buffer 1 Target Address bits 31-24							
7	6	5	4	3	2	1	0

REG[025Bh] bits 7-0
REG[025Ah] bits 7-0
REG[0259h] bits 7-0
REG[0258h] bits 7-0 SDRAM Buffer 1 Target Address bits [31:0]
These bits specify the target address in SDRAM for transfers between SDRAM Buffer 1 and the SDRAM. These bits are automatically incremented at the end of a read/write operation (when REG[0252h] bit 0 returns to 0b) according to the setting of the SDRAM Buffer 1 Rectangular Increment bit, REG[0252h] bit 2.

REG[025Ch] SDRAM Buffer 1 Data Port Register 0 Default = 00h							Read/Write
SDRAM Buffer 1 Data Port bits 7-0							
7	6	5	4	3	2	1	0

REG[025Dh] SDRAM Buffer 1 Data Port Register 1 Default = 00h							Read/Write
SDRAM Buffer 1 Data Port bits 15-8							
7	6	5	4	3	2	1	0

REG[025Dh] bits 7-0
REG[025Ch] bits 7-0 SDRAM Buffer 1 Data Port bits [15:0]

These bits are the data port where the Host reads from or writes to SDRAM Buffer 1. These registers are also "aliased" in the range REG[0380h] ~ REG[03FFh]. For example, writing to REG[0398h] is the same as writing to REG[025Ch]. The purpose of this "aliased" address range is for Direct host interfaces with "burst" mode which have incrementing addresses.

When the host interface is 16-bit and both byte and 16-bit word accesses of the SDRAM Buffer port is desired, an even number of byte accesses are required before a 16-bit word access is possible.

Note

When using SPI for SDRAM read accesses, the number of bytes specified by the SDRAM Buffer 1 Read Bytes bits (REG[0254h] bits 7-0) must be read from this port without interruption.

REG[0260h] SDRAM Buffer 0 Rectangular Increment Register 0 Default = 00h							Read/Write
SDRAM Buffer 0 Rectangular Increment Value bits 7-0							
7	6	5	4	3	2	1	0

REG[0261h] SDRAM Buffer 0 Rectangular Increment Register 1 Default = 00h							Read/Write
n/a			SDRAM Buffer 0 Rectangular Increment Value bits 12-8				
7	6	5	4	3	2	1	0

REG[0261h] bits 4-0
REG[0260h] bits 7-0 SDRAM Buffer 0 Rectangular Increment Value bits [12:0]

When the SDRAM Buffer 0 Rectangular Increment bit is set to 1b (REG[0242h] bit 2 = 1b), these bits specify the value that is added to the SDRAM Buffer 0 Target Address (REG[0248h] ~ REG[024Bh]) when the SDRAM Buffer 0 transfer completes. This method is used to perform rectangular image reads/writes between the Host and SDRAM.

REG[0262h] SDRAM Buffer 1 Rectangular Increment Register 0 Default = 00h							Read/Write
SDRAM Buffer 1 Rectangular Increment Value bits 7-0							
7	6	5	4	3	2	1	0

REG[0263h] SDRAM Buffer 1 Rectangular Increment Register 1 Default = 00h							Read/Write
n/a			SDRAM Buffer 1 Rectangular Increment Value bits 12-8				
7	6	5	4	3	2	1	0

REG[0263h] bits 4-0
REG[0262h] bits 7-0 SDRAM Buffer 1 Rectangular Increment Value bits [12:0]

When the SDRAM Buffer 1 Rectangular Increment bit is set to 1b (REG[0252h] bit 2 = 1b), these bits specify the value that is added to the SDRAM Buffer 1 Target Address (REG[0258h] ~ REG[025Bh]) when the SDRAM Buffer 1 transfer completes. This method is used to perform rectangular image reads/writes between the Host and SDRAM.

REG[0264h] SDRAM Read/Write Buffer Internal Address Register 0 Default = 00h							Read Only
SDRAM Read/Write Buffer Internal Address bits 7-0							
7	6	5	4	3	2	1	0

REG[0265h] SDRAM Read/Write Buffer Internal Address Register 1 Default = 00h							Read Only
SDRAM Read/Write Buffer Internal Address bits 15-8							
7	6	5	4	3	2	1	0

REG[0266h] SDRAM Read/Write Buffer Internal Address Register 2 Default = 00h							Read Only
SDRAM Read/Write Buffer Internal Address bits 23-16							
7	6	5	4	3	2	1	0

REG[0267h] SDRAM Read/Write Buffer Internal Address Register 3 Default = 00h							Read Only
SDRAM Read/Write Buffer Internal Address 31-24							
7	6	5	4	3	2	1	0

REG[0267h] bits 7-0
REG[0266h] bits 7-0
REG[0265h] bits 7-0
REG[0264h] bits 7-0 SDRAM Read/Write Buffer Internal Address bits [31:0] (Read Only)

These bits specify the internal memory pointer of the SDRAM Read/Write Buffer.

Note

These bits are updated at the end of each SDRAM Buffer transfer.

REG[0300h] ~ REG[037Eh] (Even Addresses) Aliased SDRAM Buffer 0 Data Port Register 0 Default = 00h Read/Write							
Aliased SDRAM Buffer 0 Data Port bits 7-0							
7	6	5	4	3	2	1	0

REG[0301h] ~ REG[037Fh] (Odd Addresses) Aliased SDRAM Buffer 0 Data Port Register 1 Default = 00h Read/Write							
Aliased SDRAM Buffer 0 Data Port bits 15-8							
7	6	5	4	3	2	1	0

REG[0300h] bits 7-0
through
REG[037Fh] bits 7-0

Aliased SDRAM Buffer 0 Data Port bits [15:0]
These are the “aliased” registers of the SDRAM Buffer 0 Data Port REG[024Ch] ~ REG[024Dh]. Writing to REG[0300h], REG[0302h], REG0304h], REG[0306h], and so on, is the same as writing to REG[024Ch]. Writing to REG[0301h], REG[0303h], REG0304h], REG[0305h], and so on, is the same as writing to REG[024Dh].The purpose of this “aliased” address range is for Direct host interfaces with “burst” mode which have incrementing addresses.

When the host interface is 16-bit and both byte and 16-bit word accesses of the SDRAM Buffer port is desired, an even number of byte accesses are required before a 16-bit word access is possible.

Note

These registers should not be used when the SPI host interface is selected (see Section Table 5-12 :, “Host Interface Configuration Summary” on page 34). For SDRAM Buffer 0 accesses, use the SDRAM Buffer 0 Data Port at REG[024Ch] ~ REG[024Dh].

REG[0380h] ~ REG[03FEh] (Even Addresses) Aliased SDRAM Buffer 1 Data Port Register 0 Default = 00h Read/Write							
Aliased SDRAM Buffer 1 Data Port bits 7-0							
7	6	5	4	3	2	1	0

REG[0381h] ~ REG[03FFh] (Odd Addresses) Aliased SDRAM Buffer 1 Data Port Register 1 Default = 00h Read/Write							
Aliased SDRAM Buffer 1 Data Port bits 15-8							
7	6	5	4	3	2	1	0

REG[0380h] bits 7-0
through
REG[03FFh] bits 7-0 Aliased SDRAM Buffer 1 Data Port bits [15:0]

These are the “aliased” registers of the SDRAM Buffer 1 Data Port REG[025Ch] ~ REG[025Dh]. Writing to REG[0380h], REG[0382h], REG0384h], REG[0386h], and so on, is the same as writing to REG[025Ch]. Writing to REG[0381h], REG[0383h], REG0384h], REG[0385h], and so on, is the same as writing to REG[025Dh].The purpose of this “aliased” address range is for Direct host interfaces with “burst” mode which have incrementing addresses.

When the host interface is 16-bit and both byte and 16-bit word accesses of the SDRAM Buffer port is desired, an even number of byte accesses are required before a 16-bit word access is possible.

Note

These registers should not be used when the SPI host interface is selected (see Table 5-12: Host Interface Configuration Summary). For SDRAM Buffer 1 accesses, use the SDRAM Buffer 1 Data Port at REG[025Ch] ~ REG[025Dh].

10.4.10 Warp Logic Configuration Registers

For a detailed discussion on the Display Subsystem, including the Warp Module, see Section Chapter 13, “Display Subsystem” on page 448.

REG[0400h] Warp Logic Configuration Register Default = 00h							Read/Write
Warp Logic Software Reset (WO)	Warp Logic Frame Double-Buffering Control Source	Luminance Bilinear Enable	Warp Logic Bilinear Enable	Warp Logic Input/Output Pixel Data Format	Reserved	Luminance Effect Enable	Warp Logic Effect Enable
7	6	5	4	3	2	1	0

bit 7 Warp Logic Software Reset (Write Only)
This bit performs a software reset of the Warp logic.
Writing a 0b to this bit has no effect.
Writing a 1b to this bit initiates a software reset of the Warp logic.

bit 6 Warp Logic Frame Double-Buffering Control Source
This bit determines how frame double-buffering is controlled for the Warp Logic.
When this bit = 0b, frame double-buffering is manually controlled through software (see REG[0408h] and REG[040Ah]).
When this bit = 1b, frame double-buffering is controlled is through hardware and is selectable using REG[09DCh] bits 6-4.

bit 5 Luminance Bilinear Enable
This bit determines whether bilinear blending is used when the Luminance effect is enabled, REG[0400h] bit 1 = 1b.
When this bit = 0b, the Luminance effect is non-bilinear.
When this bit = 1b, the Luminance effect uses bilinear blending of adjacent pixels.

bit 4 Warp Logic Bilinear Enable
This bit determines whether bilinear blending is used when the Warp Logic effect is enabled, REG[0400h] bit 0= 1b.
When this bit = 0b, the Warp Logic effect is non-bilinear.
When this bit = 1b, the Warp Logic effect uses bilinear blending of adjacent pixels.

bit 3 Warp Logic Input/Output Pixel Data Format
This bit selects the RGB data pixel format of image data input to and output from the Warp Logic.
When this bit = 0b, the data pixel format is 16 bpp (RGB 5:6:5).
When this bit = 1b, the data pixel format is 8 bpp (RGB 3:3:2).

bit 2 Reserved
This bit must be set to 0b.

bit 1 Luminance Effect Enable
When the Warp Logic Effect is enabled (REG[0400h] bit 0 = 1b), this bit controls the Luminance effect.
When this bit = 0b, the Luminance effect is disabled.
When this bit = 1b, the Luminance effect is enabled.

bit 0 Warp Logic Effect Enable
This bit controls the Warp Logic effect. For details on the Warp Logic, see Section 13.2.3, "Warp Engine" on page 465.
When this bit = 0b, the Warp Logic effect is disabled.
When this bit = 1b, the Warp Logic effect is enabled.

REG[0402h] Warp Logic Event Flag Register Default = 00h							Read Only
n/a		Read Luminance Table End Event Flag	Read Offset Table End Event Flag	Reserved	Warp Logic Frame End Event Flag	n/a	
7	6	5	4	3	2	1	0

bit 5 Read Luminance Table End Event Flag (Read Only)
This bit indicates whether the end of the Warp Logic Luminance Table (see REG[0454h] ~ REG[0457h]) has been read. This flag is masked by the Read Luminance Table End Event Enable bit, REG[0404h] bit 5.
When this bit = 0b, the end of the Luminance table has not been read yet.
When this bit = 1b, the end of the Luminance table has been read.

To clear this flag, write a 1b to REG[0406h] bit 5.

bit 4 Read Offset Table End Event Flag (Read Only)
This bit indicates whether the end of the Warp Logic Offset Table (see REG[0444h] ~ REG[0447h]) has been read. This flag is masked by the Read Offset Table End Event Enable bit, REG[0404h] bit 4.
When this bit = 0b, the end of the Luminance table has not been read yet.
When this bit = 1b, the end of the Luminance table has been read.

To clear this flag, write a 1b to REG[0406h] bit 4.

bit 3 Reserved
The default value of this bit is 0b.

bit 2 Warp Logic Frame End Event Flag (Read Only)
This bit indicates whether the Warp Logic has processed the frame (not when the frame is completely written). This flag is masked by the Warp Logic Frame End Event Enable bit, REG[0404h] bit 2.
When this bit = 0b, the end of the Warp Logic frame has not occurred.
When this bit = 1b, the end of the Warp Logic frame has occurred.

To clear this flag, write a 1b to REG[0406h] bit 2.

REG[0404h] Warp Logic Event Enable Register Default = 00h							Read/Write
n/a		Read Luminance Table End Event Enable	Read Offset Table End Event Enable	Reserved	Warp Logic Frame End Event Enable	n/a	
7	6	5	4	3	2	1	0

bit 5	Read Luminance Table End Event Enable This bit controls the Read Luminance Table End Event. The status of the event is indicated by the Read Luminance Table End Event Flag, REG[0402h] bit 5. When this bit = 0b, the Read Luminance Table End Event is disabled. When this bit = 1b, the Read Luminance Table End Event is enabled.
bit 4	Read Offset Table End Event Enable This bit controls the Read Offset Table End Event. The status of the event is indicated by the Read Offset Table End Event Flag, REG[0402h] bit 4. When this bit = 0b, the Read Offset Table End Event is disabled. When this bit = 1b, the Read Offset Table End Event is enabled.
bit 3	Reserved The default value of this bit is 0b.
bit 2	Warp Logic Frame End Event Enable This bit controls the Warp Logic Frame End Event. The status of the event is indicated by the Warp Logic Frame End Event Flag, REG[0402h] bit 2. When this bit = 0b, the Warp Logic Frame End Event is disabled. When this bit = 1b, the Warp Logic Frame End Event is enabled.

REG[0406h] Warp Logic Event Clear Register Default = 00h							Write Only
n/a		Read Luminance Table End Event Clear	Read Offset Table End Event Clear	Reserved	Warp Logic Frame End Event Clear	n/a	
7	6	5	4	3	2	1	0

bit 5 Read Luminance Table End Event Clear (Write Only)
This bit clears the Read Luminance Table End Event Flag, REG[0402h] bit 5.
Writing a 0b to this bit has no effect.
Writing a 1b to this bit clears the Read Luminance Table End Event Flag.

bit 4 Read Offset Table End Event Clear (Write Only)
This bit clears the Read Offset Table End Event Flag, REG[0402h] bit 4.
Writing a 0b to this bit has no effect.
Writing a 1b to this bit clears the Read Offset Table End Event Flag.

bit 3 Reserved
The default value of this bit is 0b.

bit 2 Warp Logic Frame End Event Clear (Write Only)
This bit clears the Warp Logic Frame End Event Flag, REG[0402h] bit 2.
Writing a 0b to this bit has no effect.
Writing a 1b to this bit clears the Warp Logic Frame End Event Flag.

REG[0408h] Warp Logic Frame Status Register Default = 00h							Read Only
n/a				Warp Logic Busy	Warp Logic Current Frame Buffer	Warp Logic Frame Buffer 1 Ready Status	Warp Logic Frame Buffer 0 Ready Status
7	6	5	4	3	2	1	0

bit 3 Warp Logic Busy (Read Only)
This bit indicates whether the Warp Logic is busy processing a frame.
When this bit = 0b, the Warp Logic is idle (not busy).
When this bit = 1b, the Warp Logic is busy processing a frame.

bit 2 Warp Logic Current Frame Buffer (Read Only)
This bit indicates which frame buffer (0 or 1) that the Warp Logic is currently reading from (or processing).
When this bit = 0b, the current buffer is Warp Logic Frame Buffer 0.
When this bit = 1b, the current buffer is Warp Logic Frame Buffer 1.

bit 1 Warp Logic Frame Buffer 1 Ready Status (Read Only)
This bit indicates the ready status of Warp Logic Frame Buffer 1. The frame buffer is ready when it contains valid frame image data.
When this bit = 0b, Warp Logic Frame Buffer 1 is not ready.
When this bit = 1b, Warp Logic Frame Buffer 1 is ready.

bit 0 Warp Logic Frame Buffer 0 Ready Status (Read Only)
This bit indicates the ready status of Warp Logic Frame Buffer 0. The frame buffer is ready when it contains valid frame image data.
When this bit = 0b, Warp Logic Frame Buffer 0 is not ready.
When this bit = 1b, Warp Logic Frame Buffer 0 is ready.

REG[040Ah] Warp Logic Frame Ready Set Register Default = 00h							Write Only
n/a						Set Warp Logic Frame Buffer 1 Ready	Set Warp Logic Frame Buffer 0 Ready
7	6	5	4	3	2	1	0

bit 1 Set Warp Logic Frame Buffer 1 Ready (Write Only)
This bit only has an effect when Warp Logic double-buffering is configured for software control, REG[0400h] bit 6 = 0b.
Writing a 0b to this bit has no effect.
Writing a 1b to this bit sets this bit to 1b and indicates that the Warp Logic Frame Buffer 1 input image data is ready for reading by the Warp Logic. Once this bit is set to 1b, it remains at 1b until it is reset by the Warp Logic hardware.

bit 0 Set Warp Logic Frame Buffer 0 Ready (Write Only)
This bit only has an effect when Warp Logic double-buffering is configured for software control, REG[0400h] bit 6 = 0b.
Writing a 0b to this bit has no effect.
Writing a 1b to this bit sets this bit to 1b and indicates that the Warp Logic Frame Buffer 0 input image data is ready for reading by the Warp Logic. Once this bit is set to 1b, it remains at 1b until it is reset by the Warp Logic hardware.

REG[0410h] Warp Logic Input Width Register 0 Default = 00h							Read/Write
Warp Logic Input Width bits 7-0							
7	6	5	4	3	2	1	0

REG[0411h] Warp Logic Input Width Register 1 Default = 00h							Read/Write
Warp Logic Input Width bits 15-8							
7	6	5	4	3	2	1	0

REG[0411h] bits 7-0
REG[0410h] bits 7-0 Warp Logic Input Width bits [15:0]
These bits specify the width of the image data input to the Warp Logic, in pixels.

REG[0412h] Warp Logic Input Height Register 0 Default = 00h							Read/Write
Warp Logic Image Height bits 7-0							
7	6	5	4	3	2	1	0

REG[0413h] Warp Logic Input Height Register 1 Default = 00h							Read/Write
Warp Logic Input Height bits 15-8							
7	6	5	4	3	2	1	0

REG[0413h] bits 7-0
REG[0412h] bits 7-0 Warp Logic Input Height bits [15:0]
These bits specify the height of the image data input to the Warp Logic, in pixels.

REG[0414h] Warp Logic Output Width Register 0 Default = 00h							Read/Write
Warp Logic Output Width bits 7-0 (bit 0 is read only = 0b)							
7	6	5	4	3	2	1	0

REG[0415h] Warp Logic Output Width Register 1 Default = 00h							Read/Write
n/a					Warp Logic Output Width bits 10-8		
7	6	5	4	3	2	1	0

REG[0415h] bits 2-0
REG[0414h] bits 7-0 Warp Logic Output Width bits [10:0]

These bits specify the width of the image data output by the Warp Logic, in pixels. REG[0414h] bit 0 is read only and always returns 0b (writes to this bit have no effect).

Note

These bits must be set such that the Warp Logic output width is a multiple of the offset horizontal block size (see REG[0440h] bits 2-0) and luminance horizontal block size (see REG[0450h] bits 2-0).

REG[0416h] Warp Logic Output Height Register 0 Default = 00h							Read/Write
Warp Logic Output Height bits 7-0 (bit 0 is read only = 0b)							
7	6	5	4	3	2	1	0

REG[0417h] Warp Logic Output Height Register 1 Default = 00h							Read/Write
n/a					Warp Logic Output Height bits 10-8		
7	6	5	4	3	2	1	0

REG[0417h] bits 2-0
REG[0416h] bits 7-0 Warp Logic Output Height bits [10:0]

These bits specify the height of the image data output by the Warp Logic, in pixels. REG[0416h] bit 0 is read only and always returns 0b (writes to this bit have no effect).

Note

These bits must be set such that the Warp Logic output height is a multiple of the offset vertical block size (see REG[0440h] bits 6-4) and luminance horizontal block size (see REG[0450h] bits 6-4).

REG[0420h] Warp Logic Frame Buffer 0 Start Address Register 0 Default = 00h							Read/Write
Warp Logic Frame Buffer 0 Start Address bits 7-0 (bits 2-0 are read only = 000b)							
7	6	5	4	3	2	1	0

REG[0421h] Warp Logic Frame Buffer 0 Start Address Register 1 Default = 00h							Read/Write
Warp Logic Frame Buffer 0 Start Address bits 15-8							
7	6	5	4	3	2	1	0

REG[0422h] Warp Logic Frame Buffer 0 Start Address Register 2 Default = 00h							Read/Write
Warp Logic Frame Buffer 0 Start Address bits 23-16							
7	6	5	4	3	2	1	0

REG[0423h] Warp Logic Frame Buffer 0 Start Address Register 3 Default = 00h							Read/Write
Warp Logic Frame Buffer 0 Start Address bits 31-24							
7	6	5	4	3	2	1	0

REG[0423h] bits 7-0
REG[0422h] bits 7-0
REG[0421h] bits 7-0
REG[0420h] bits 7-0 Warp Logic Frame Buffer 0 Start Address bits [31:0]

These bits specify the memory start address for Warp Logic Frame Buffer 0 which is used for input image data to the Warp Logic. These bits must be set such that the start address is 8 byte (64-bit) aligned. REG[0420h] bits 2-0 are read only and always return 000b (writes to these bits have no effect).

REG[0424h] Warp Logic Frame Buffer 1 Start Address Register 0 Default = 00h Read/Write							
Warp Logic Frame Buffer 1 Start Address bits 7-0 (bits 2-0 are read only = 000b)							
7	6	5	4	3	2	1	0

REG[0425h] Warp Logic Frame Buffer 1 Start Address Register 1 Default = 00h Read/Write							
Warp Logic Frame Buffer 1 Start Address bits 15-8							
7	6	5	4	3	2	1	0

REG[0426h] Warp Logic Frame Buffer 1 Start Address Register 2 Default = 00h Read/Write							
Warp Logic Frame Buffer 1 Start Address bits 23-16							
7	6	5	4	3	2	1	0

REG[0427h] Warp Logic Frame Buffer 1 Start Address Register 3 Default = 00h Read/Write							
Warp Logic Frame Buffer 1 Start Address bits 31-24							
7	6	5	4	3	2	1	0

REG[0427h] bits 7-0
REG[0426h] bits 7-0
REG[0425h] bits 7-0
REG[0424h] bits 7-0 Warp Logic Frame Buffer 1 Start Address bits [31:0]

These bits specify the memory start address for Warp Logic Frame Buffer 1 which is used for input image data to the Warp Logic. These bits must be set such that the start address is 8 byte (64-bit) aligned. REG[0424h] bits 2-0 are read only and always return 000b (writes to these bits have no effect).

REG[0430h] Warp Logic Background Color Blue Register Default = 00h Read/Write							
Warp Logic Background Color Blue bits 7-0 (bits 2-0 RO)							
7	6	5	4	3	2	1	0

bits 7-0 Warp Logic Background Color Blue bits [7:0]

These bits specify the blue component of the Warp Logic background color. Bits 2-0 of this register are read only and always return 000b. The background color registers (REG[0430h] ~ REG[0432h]) specify the background color as RGB 8:8:8, but only the most significant bits of each color byte are actually used.

If the Warp Logic Input/Output Data Pixel Format is RGB 5:6:5 (REG[0400h] bit 3 = 0b).
REG[0432h] bits 7-3 = RED
REG[0431h] bits 7-2 = GREEN
REG[0430h] bits 7-3 = BLUE

If the Warp Logic Input/Output Data Pixel Format is RGB 3:3:2 (REG[0400h] bit 3 = 1b).
REG[0432h] bits 7-5 = RED
REG[0431h] bits 7-5 = GREEN
REG[0430h] bits 7-6 = BLUE

REG[0431h] Warp Logic Background Color Green Register							
Default = 00h							Read/Write
Warp Logic Background Color Green bits 7-0 (bits 1-0 RO)							
7	6	5	4	3	2	1	0

bits 7-0 Warp Logic Background Color Green bits [7:0]
These bits specify the green component of the Warp Logic background color. Bits 1-0 of this register are read only and always return 00b. The background color registers (REG[0430h] ~ REG[0432h) specify the background color as RGB 8:8:8, but only the most significant bits of each color byte are actually used.

If the Warp Logic Input/Output Data Pixel Format is RGB 5:6:5 (REG[0400h] bit 3 = 0b).
REG[0432h] bits 7-3 = RED
REG[0431h] bits 7-2 = GREEN
REG[0430h] bits 7-3 = BLUE
If the Warp Logic Input/Output Data Pixel Format is RGB 3:3:2 (REG[0400h] bit 3 = 1b).
REG[0432h] bits 7-5 = RED
REG[0431h] bits 7-5 = GREEN
REG[0430h] bits 7-6 = BLUE

REG[0432h] Warp Logic Background Color Red Register							
Default = 00h							Read/Write
Warp Logic Background Color Red bits 7-0 (bits 2-0 RO)							
7	6	5	4	3	2	1	0

bits 7-0 Warp Logic Background Color Red bits [7:0]
These bits specify the red component of the Warp Logic background color. Bits 2-0 of this register are read only and always return 000b. The background color registers (REG[0430h] ~ REG[0432h]) specify the background color as RGB 8:8:8, but only the most significant bits of each color byte are actually used.

If the Warp Logic Input/Output Data Pixel Format is RGB 5:6:5 (REG[0400h] bit 3 = 0b).
REG[0432h] bits 7-3 = RED
REG[0431h] bits 7-2 = GREEN
REG[0430h] bits 7-3 = BLUE
If the Warp Logic Input/Output Data Pixel Format is RGB 3:3:2 (REG[0400h] bit 3 = 1b).
REG[0432h] bits 7-5 = RED
REG[0431h] bits 7-5 = GREEN
REG[0430h] bits 7-6 = BLUE

REG[0434h] Warp Logic Input X Offset Register 0 Default = 00h							Read/Write
Warp Logic Input X Offset bits 7-0							
7	6	5	4	3	2	1	0

REG[0435h] Warp Logic Input X Offset Register 1 Default = 00h							Read/Write
Warp Logic Input X Offset bits 15-8							
7	6	5	4	3	2	1	0

REG[0435h] bits 7-0
REG[0434h] bits 7-0 Warp Logic Input X Offset bits [15:0]

These bits specify the Warp Logic Input X Offset, in pixels. When the Warp Logic output size is smaller than the input size, the input X,Y offset values (see also REG[0436h] ~ REG[0437h]) specify the top left corner of the output window which can "pan" the larger input image. The input X offset value is specified relative to the top left corner of the input image. The X offset supports both positive and negative values using 2's complement.

REG[0436h] Warp Logic Input Y Offset Register 0 Default = 00h							Read/Write
Warp Logic Input Y Offset bits 7-0							
7	6	5	4	3	2	1	0

REG[0437h] Warp Logic Input Y Offset Register 1 Default = 00h							Read/Write
Warp Logic Input Y Offset bits 15-8							
7	6	5	4	3	2	1	0

REG[0437h] bits 7-0
REG[0436h] bits 7-0 Warp Logic Input Y Offset bits [15:0]

These bits specify the Warp Logic Input Y Offset, in pixels. When the Warp Logic output size is smaller than the input size, the input X,Y offset values (see also REG[0434h] ~ REG[0435h]) specify the top left corner of the output window which can "pan" the larger input image. The input Y offset value is specified relative to the top left corner of the input image. The Y offset supports both positive and negative values using 2's complement.

REG[0440h] Warp Logic Offset Table Configuration Register Default = 33h							Read/Write
n/a	Offset Vertical Block Power bits 2-0			n/a	Offset Horizontal Block Power bits 2-0		
7	6	5	4	3	2	1	0

bits 6-4 Offset Vertical Block Power bits [2:0]
The Warp Logic divides the output image into NxM pixel blocks. These bits specify the vertical size (M) of the pixel block.

Table 10-28: Offset Vertical Block Power Selection

REG[0440h] bits 6-4	Vertical Block Power
000b	Reserved
001b	Reserved
010b	4 (2^2)
011b (default)	8 (2^3)
100b	16 (2^4)
101b	32 (2^5)
110b	64 (2^6)
111b	Reserved

bits 2-0 Offset Horizontal Block Power bits [2:0]
The Warp Logic divides the output image into NxM pixel blocks. These bits specify the horizontal size (N) of the pixel block.

Table 10-29: Offset Horizontal Block Power Selection

REG[0440h] bits 2-0	Horizontal Block Power
000b	Reserved
001b	Reserved
010b	4 (2^2)
011b (default)	8 (2^3)
100b	16 (2^4)
101b	32 (2^5)
110b	64 (2^6)
111b	Reserved

REG[0444h] Warp Logic Offset Table SDRAM Start Address Register 0 Default = 00h Read/Write							
Warp Logic Offset Table SDRAM Start Address bits 7-0 (bits 2-0 are read only = 000b)							
7	6	5	4	3	2	1	0

REG[0445h] Warp Logic Offset Table SDRAM Start Address Register 1 Default = 00h Read/Write							
Warp Logic Offset Table SDRAM Start Address bits 15-8							
7	6	5	4	3	2	1	0

REG[0446h] Warp Logic Offset Table SDRAM Start Address Register 2 Default = 00h Read/Write							
Warp Logic Offset Table SDRAM Start Address bits 23-16							
7	6	5	4	3	2	1	0

REG[0447h] Warp Logic Offset Table SDRAM Start Address Register 3 Default = 00h Read/Write							
Warp Logic Offset Table SDRAM Start Address bits 31-24							
7	6	5	4	3	2	1	0

REG[0447h] bits 7-0
REG[0446h] bits 7-0
REG[0445h] bits 7-0
REG[0444h] bits 7-0 Warp Logic Offset Table SDRAM Start Address bits [31:0]

These bits specify the location in SDRAM of the Warp Logic Offset Table. These bits must be set such that the start address is 8 byte (64-bit) aligned. REG[0444h] bits 2-0 are read only and always return 000b (writes to these bits have no effect).

Additionally, the Warp Logic Offset Table layout also requires 8 byte alignment for each row. The byte arrangement for each row must be set as described below.

Table 10-30: Warp Logic Offset Table Layout

Warp Table = each value is 16-bit (2's complement)									
X(0,0)	Y(0,0)	X(1,0)	Y(1,0)	X(2,0)	Y(2,0)	•••	X(outputwidth÷N,0)	Y(outputwidth÷N,0)	See Note
X(0,1)	Y(0,1)	X(1,1)	Y(1,1)	X(2,1)	Y(2,1)	•••	X(outputwidth÷N,1)	Y(outputwidth÷N,1)	See Note
• • •	• • •	• • •	• • •	• • •	• • •	• • •	• • •	• • •	• • •
X(0,outputheight÷M)	Y(0,outputheight÷M	•••	•••	•••	•••	•••	X(outputwidth÷N, outputheight÷M)	Y(outputwidth÷N, outputheight÷M	See Note

N is the horizontal size of the pixel block as specified by REG[0440h] bits 2-0
M is the vertical size of the pixel block as specified by REG[0440h] bits 6-4

Note

Each row must be padded if it does not end on an 8 byte boundary.

REG[0450h] Warp Logic Luminance Table Configuration Register 0 Default = 33h Read/Write							
n/a	Luminance Vertical Block Power bits 2-0			n/a	Luminance Horizontal Block Power bits 2-0		
7	6	5	4	3	2	1	0

bits 6-4 Luminance Vertical Block Power bits [2:0]
The Luminance function divides the output image into NxM pixel blocks. These bits specify the vertical size (M) of the pixel block.

Table 10-31: Luminance Vertical Block Power

REG[0450h] bits 6-4	Vertical Block Power
000b	Reserved
001b	Reserved
010b	4 (2^2)
011b (default)	8 (2^3)
100b	16 (2^4)
101b	32 (2^5)
110b	64 (2^6)
111b	Reserved

bits 2-0 Luminance Horizontal Block Power bits [2:0]
The Luminance function divides the output image into NxM pixel blocks. These bits specify the horizontal size (N) of the pixel block.

Table 10-32: Luminance Horizontal Block Power

REG[0450h] bits 2-0	Horizontal Block Power
000b	Reserved
001b	Reserved
010b	4 (2^2)
011b (default)	8 (2^3)
100b	16 (2^4)
101b	32 (2^5)
110b	64 (2^6)
111b	Reserved

REG[0452h] Warp Logic Luminance Table Configuration Register 1 Default = 01h							Read/Write
n/a						Warp Logic Background Color Luminance Disable	Warp Logic Black Color Luminance Disable
7	6	5	4	3	2	1	0

bit 1 Warp Logic Background Color Luminance Disable
When the Luminance effect is enabled (REG[0400h] bit 1 = 1b), this bit determines whether the luminance effect is applied to the background color (REG[0430h] ~ REG[0432h]).
When this bit = 0b, the luminance effect is applied to the background color.
When this bit = 1b, the luminance effect is not applied to background color.

bit 0 Warp Logic Black Color Luminance Disable
When the Luminance effect is enabled (REG[0400h] bit 1 = 1b), this bit determines whether the luminance effect is applied to black pixels.
When this bit = 0b, the luminance effect is applied to black pixels.
When this bit = 1b, the luminance effect is not applied to black pixels.

REG[0454h] Warp Logic Luminance Table SDRAM Start Address Register 0 Default = 00h Read/Write							
Warp Logic Luminance Table SDRAM Start Address bits 7-0 (bits 2-0 are read only = 000b)							
7	6	5	4	3	2	1	0

REG[0455h] Warp Logic Luminance Table SDRAM Start Address Register 1 Default = 00h Read/Write							
Warp Logic Luminance Table SDRAM Start Address bits 15-8							
7	6	5	4	3	2	1	0

REG[0456h] Warp Logic Luminance Table SDRAM Start Address Register 2 Default = 00h Read/Write							
Warp Logic Luminance Table SDRAM Start Address bits 23-16							
7	6	5	4	3	2	1	0

REG[0457h] Warp Logic Luminance Table SDRAM Start Address Register 3 Default = 00h Read/Write							
Warp Logic Luminance Table SDRAM Start Address bits 31-24							
7	6	5	4	3	2	1	0

REG[0457h] bits 7-0
REG[0456h] bits 7-0
REG[0455h] bits 7-0
REG[0454h] bits 7-0 Warp Logic Luminance Table SDRAM Start Address bits [31:0]

These bits specify the location in SDRAM of the Warp Logic Luminance Table. These bits must be set such that the start address is 8 byte (64-bit) aligned. REG[0454h] bits 2-0 are read only and always return 000b (writes to these bits have no effect).

Additionally, the Luminance Table layout also requires 8 byte alignment for each row. The byte arrangement for each row must be set as described below.

Table 10-33: Luminance Table Layout

Luminance Table = each value is 8-bit (2's complement)						
Luminance(0,0)	Luminance(1,0)	Luminance(2,0)	Luminance(3,0)	•••	Luminance (outputwidth÷N,0)	See Note
Luminance(0,1)	Luminance(1,1)	Luminance(2,1)	Luminance(3,1)	•••	Luminance (outputwidth÷N,1)	See Note
⋮	⋮	⋮	⋮	⋮	⋮	⋮
Luminance (0,outputheight÷M)	•••	•••	•••	•••	Luminance (outputwidth÷N, outputheight÷M)	See Note

N is the horizontal size of the pixel block as specified by REG[0450h] bits 2-0
M is the vertical size of the pixel block as specified by REG[0450h] bits 6-4

Note

Each row must be padded if it does not end on an 8 byte boundary.

10.4.11 Blending Engine Configuration Registers

For a detailed discussion on the Display Subsystem, including the Blending Engine, see Section Chapter 13, “Display Subsystem” on page 448.

REG[0900h] CH1OUT Control Register Default = 00h Read/Write							
n/a		CH1OUT Writeback Pixel Format bits 1-0		CH1OUT Vertical Flip Enable	CH1OUT Writeback Memory Mode	CH1OUT Mode	CH1OUT Enable
7	6	5	4	3	2	1	0

bits 5-4 CH1OUT Writeback Pixel Format bits [1:0]
When CH1OUT writeback mode is selected (REG[0900h] bit 1 = 1b), these bits specify the RGB pixel format for the image data written to the SDRAM.

Table 10-34: CH1OUT Writeback Pixel Format Select

REG[0900h] bits 5-4	CH1OUT Writeback Pixel Format
00b	RGB 3:3:2
01b	RGB 5:6:5
10b	RGB 8:8:8
11b	Reserved

bit 3 CH1OUT Vertical Flip Enable
This bit determines whether the image data output on CH1OUT is flipped around the X axis (vertical). This bit must set to 0b for tiled frame mode, REG[0900h] bit 2 = 1b.
When this bit = 0b, CH1OUT image data is not vertically flipped (disabled).
When this bit = 1b, CH1OUT image data is vertically flipped (enabled).

bit 2 CH1OUT Writeback Memory Mode
When CH1OUT writeback mode is selected (REG[0900h] bit 1 = 1b), this bit determines how the image data is stored in memory. For details on the memory organization methods, see Section 13.3, “Memory Organization of Frames” on page 470.
When this bit = 0b, CH1OUT writeback uses “line-by-line” mode to write to SDRAM.
When this bit = 1b, CH1OUT writeback uses “tiled frame” mode to write to SDRAM.

Note

For tiled frame mode, the image width must be a multiple of 8 pixels and CH1OUT Vertical Flip must be disabled, REG[0900h] bit 3 = 0b.

bit 1 CH1OUT Mode
This bit selects the CH1OUT mode which determines whether the image data output on CH1OUT goes to the LCD interface, or is written back to the SDRAM. For further details on CH1OUT Writeback mode, see Section 13.2.4, “CH1OUT Writeback” on page 467.
When this bit = 0b, CH1OUT image data goes to LCD interface.
When this bit = 1b, CH1OUT image data is written back to SDRAM.

bit 0 CH1OUT Enable
This bit controls the blending engine output CH1OUT. For an overview of the Blending Engine, see Section 13.1, “Block Diagram” on page 448.
When this bit = 0b, CH1OUT is disabled.
When this bit = 1b, CH1OUT is enabled.

Note

If hardware frame control is selected for the MAIN window (REG[09D8h] bit 0 = 1b), it must be disabled before CH1OUT can be disabled. The following sequence is recommended.

1. Disable hardware frame control, REG[09D8h] bit 0 = 0b.
2. Wait 1 frame.
3. Disable CH1OUT, REG[0900h] bit 0 = 0b.

REG[0904h] CH1OUT Writeback Frame Buffer 0 Address Register 0 Default = 00h							Read/Write
CH1OUT Writeback Frame Buffer 0 Address bits 7-0							
7	6	5	4	3	2	1	0

REG[0905h] CH1OUT Writeback Frame Buffer 0 Address Register 1 Default = 00h							Read/Write
CH1OUT Writeback Frame Buffer 0 Address bits 15-8							
7	6	5	4	3	2	1	0

REG[0906h] CH1OUT Writeback Frame Buffer 0 Address Register 2 Default = 00h							Read/Write
CH1OUT Writeback Frame Buffer 0 Address bits 23-16							
7	6	5	4	3	2	1	0

REG[0907h] CH1OUT Writeback Frame Buffer 0 Address Register 3 Default = 10h							Read/Write
CH1OUT Writeback Frame Buffer 0 Address bits 31-24							
7	6	5	4	3	2	1	0

REG[0907h] bits 7-0
REG[0906h] bits 7-0
REG[0905h] bits 7-0
REG[0904h] bits 7-0 CH1OUT Writeback Frame Buffer 0 Address bits [31:0]
These bits specify the start address in SDRAM of CH1OUT Writeback Frame Buffer 0. These bits must be set such that the start address is 8 byte (64-bit) aligned.

REG[0908h] CH1OUT Writeback Frame Buffer 1 Address Register 0
Default = 00h — Read/Write

CH1OUT Writeback Frame Buffer 1 Address bits 7-0							
7	6	5	4	3	2	1	0

REG[0909h] CH1OUT Writeback Frame Buffer 1 Address Register 1
Default = 00h — Read/Write

CH1OUT Writeback Frame Buffer 1 Address bits 15-8							
7	6	5	4	3	2	1	0

REG[090Ah] CH1OUT Writeback Frame Buffer 1 Address Register 2
Default = 00h — Read/Write

CH1OUT Writeback Frame Buffer 1 Address bits 23-16							
7	6	5	4	3	2	1	0

REG[090Bh] CH1OUT Writeback Frame Buffer 1 Address Register 3
Default = 10h — Read/Write

CH1OUT Writeback Frame Buffer 1 Address bits 31-24							
7	6	5	4	3	2	1	0

REG[090Bh] bits 7-0
REG[090Ah] bits 7-0
REG[0909h] bits 7-0
REG[0908h] bits 7-0 CH1OUT Writeback Frame Buffer 1 Address bits [31:0]
These bits specify the start address in SDRAM of CH1OUT Writeback Frame Buffer 1. These bits must be set such that the start address is 8 byte (64-bit) aligned.

REG[090Ch] Scratchpad Register 0
Default = 40h — Read/Write

Scratchpad Register bits 7-0							
7	6	5	4	3	2	1	0

REG[090Dh] Scratchpad Register 1
Default = 00h — Read/Write

Scratchpad Register bits 15-8							
7	6	5	4	3	2	1	0

REG[090Eh] Scratchpad Register 2
Default = 00h — Read/Write

Scratchpad Register bits 23-16							
7	6	5	4	3	2	1	0

REG[090Fh] Scratchpad Register 3
Default = 00h — Read/Write

Scratchpad Register bits 31-24							
7	6	5	4	3	2	1	0

REG[090Fh] bits 7-0
REG[090Eh] bits 7-0
REG[090Dh] bits 7-0
REG[090Ch] bits 7-0 Scratchpad Register bits [31:0]
These bits have no hardware effect and are available for scratchpad use.

REG[0920h] CH2OUT Control Register Default = 00h							Read/Write
n/a							CH2OUT Enable
7	6	5	4	3	2	1	0

bit 0 CH2OUT Enable
This bit controls the blending engine output CH2OUT. For an overview of the Blending Engine, see Section 13.1, “Block Diagram” on page 448.
When this bit = 0b, CH2OUT is disabled.
When this bit = 1b, CH2OUT is enabled.

Note
If hardware frame control is selected for the AUX window (REG[09D9h] bit 0 = 1b) or the OSD window (REG[09DAh] bit 0 = 1b) and either window is the source for CH2, hardware frame control for the windows must be disabled before CH2OUT can be disabled. The following sequence is recommended.
1. Disable hardware frame control, REG[09D9h] and/or REG[09DAh] bit 0 = 0b.
2. Wait 1 frame.
3. Disable CH2OUT, REG[0920h] bit 0 = 0b.

REG[0930h] OSDOUT Control Register Default = 00h							Read/Write
n/a							OSDOUT Enable
7	6	5	4	3	2	1	0

bit 0 OSDOUT Enable
This bit controls the blending engine output OSDOUT. For an overview of the Blending Engine, see Section 13.1, “Block Diagram” on page 448.
When this bit = 0b, OSDOUT is disabled.
When this bit = 1b, OSDOUT is enabled.

Note
If hardware frame control is selected for the OSD window (REG[09DAh] bit 0 = 1b), it must be disabled before OSDOUT can be disabled. The following sequence is recommended.
1. Disable hardware frame control, REG[09DAh] bit 0 = 0b.
2. Wait 1 frame.
3. Disable OSDOUT, REG[0930h] bit 0 = 0b.

REG[0940h] MAIN Window Control Register Default = 00h							Read/Write
MAIN Window Line Double Enable	MAIN Horizontal Flip Enable	MAIN Vertical Flip Enable	n/a	MAIN Window Pixel Format bits 1-0		MAIN Window Fetch Mode	MAIN Window Blank
7	6	5	4	3	2	1	0

bit 7 MAIN Window Line Double Enable
This bit controls "line double" mode which is typically used for displaying interlaced images from the camera interface. When line doubling is enabled, each line of the input image stored in the SDRAM is read twice.
When this bit = 0b, MAIN window line doubling is disabled.
When this bit = 1b, MAIN window line doubling is enabled.

bit 6 MAIN Horizontal Flip Enable
This bit determines whether the image data input from the MAIN window is flipped around the Y axis (horizontal). This bit must be set to 0b when the MAIN window fetch uses "tiled-frame" mode, REG[0940h] bit 1 = 1b.
When this bit = 0b, the MAIN image data is not horizontally flipped (disabled).
When this bit = 1b, the MAIN image data is horizontally flipped (enabled).

Note
If the AUX and/or OSD window overlays the MAIN window (REG[09A0h] bits 1-0 = 00b or 01b) when MAIN Horizontal Flip is enabled, the relative position of the overlaid window(s) is flipped. However, the image in the AUX and/or OSD window is NOT flipped and is still controlled by the individual flip enable bits (see REG[0960h] bits 6-5 for AUX, REG[0980h] bits 6-5 for OSD).

bit 5 MAIN Vertical Flip Enable
This bit determines whether the image data input from the MAIN window is flipped around the X axis (vertical). This bit must be set to 0b when the MAIN window fetch uses "tiled-frame" mode, REG[0940h] bit 1 = 1b.
When this bit = 0b, the MAIN image data is not vertically flipped (disabled).
When this bit = 1b, the MAIN image data is vertically flipped (enabled).

Note
If the AUX and/or OSD window overlays the MAIN window (REG[09A0h] bits 1-0 = 00b or 01b) when MAIN Vertical Flip is enabled, the relative position of the overlaid window(s) is flipped. However, the image in the AUX and/or OSD window is NOT flipped and is still controlled by the individual flip enable bits (see REG[0960h] bits 6-5 for AUX, REG[0980h] bits 6-5 for OSD).

bits 3-2 MAIN Window Pixel Format bits [1:0]
These bits determine the RGB pixel format of the MAIN window image data that is input to the Blending Engine.

Table 10-35: MAIN Window Pixel Format Selection

REG[0940h] bits 3-2	Pixel Format
00b	8 bpp (RGB 3:3:2)
01b	16 bpp (RGB 5:6:5)
10b	24 bpp (RGB 8:8:8)
11b	Reserved

bit 1 MAIN Window Fetch Mode
This bit specifies how the MAIN window image data is stored in memory. For details on the memory organization methods, see Section 13.3, “Memory Organization of Frames” on page 470.
When this bit = 0b, MAIN window fetch uses “line-by-line” mode to read from SDRAM.
When this bit = 1b, MAIN window fetch uses “tiled-frame” mode to read from SDRAM.

Note
For tiled frame mode, the image width and virtual width must be a multiple of 8 pixels and the MAIN window image data must not be flipped (REG[0940h] bit 6 = 0b and bit 5 = 0b).

bit 0 MAIN Window Blank
This bit controls the MAIN window blank function. The blank function replaces the image data input to the Blending Engine from the MAIN window with the color specified by the MAIN Blank Color registers, REG[0944h] ~ REG[0946h].
When this bit = 0b, the MAIN window image data is read normally (not blanked).
When this bit = 1b, the MAIN window image data is “blanked” with the specified color.

REG[0942h] MAIN Window Frame Control/Status Register
Default = 00h — Read/Write

n/a		MAIN Frame Buffer 1 Ready Clear (WO)	MAIN Frame Buffer 0 Ready Clear (WO)	n/a	Main Window Current Frame Status (RO)	MAIN Frame Buffer 1 Ready	MAIN Frame Buffer 0 Ready
7	6	5	4	3	2	1	0

bit 5 MAIN Frame Buffer 1 Ready Clear (Write Only)
This bit is used to manually clear the MAIN Frame Buffer 1 Ready bit, REG[0942h] bit 1.
Writing a 0b to this bit has no effect.
Writing a 1b to this bit clears the MAIN Frame Buffer 1 Ready bit.

bit 4 MAIN Frame Buffer 0 Ready Clear (Write Only)
This bit is used to manually clear the MAIN Frame Buffer 0 Ready bit, REG[0942h] bit 0.
Writing a 0b to this bit has no effect.
Writing a 1b to this bit clears the MAIN Frame Buffer 0 Ready bit.

bit 2 — Main Window Current Frame Status (Read Only)
This bit indicates which MAIN frame buffer is currently being read by the Blending Engine.
When this bit = 0b, MAIN Frame Buffer 0 is being read by the Blending Engine.
When this bit = 1b, MAIN Frame Buffer 1 is being read by the Blending Engine.

Note

When the MAIN window is disabled and then re-enabled using the CH1OUT Enable bit (REG[0900h] bit 0), the hardware always sets the Current Frame status to 0b and checks the MAIN Frame Buffer 0 Ready bit first. Therefore before re-enabling the MAIN window, the MAIN window image stream must be reset to start with Buffer 0, the MAIN Frame Buffer 0/1 Ready bits must be cleared (see REG[0942h] bits 5-4), and the MAIN Frame Buffer 0 Ready bit must be set to 1b (REG[0942h] bit 0 = 1b).

bit 1 — MAIN Frame Buffer 1 Ready
This bit only has an effect when MAIN window double-buffering is configured for software control, REG[09D8h] bit 0 = 0b.
For Writes:
Writing a 0b to this bit has no effect.
Writing a 1b to this bit sets this bit to 1b and indicates that the MAIN Frame Buffer 1 image data is ready for reading by the Blending Engine. Once this bit is set to 1b, it remains at 1b until it is reset by the Blending Engine.
For Reads:
When this bit = 0b, MAIN Frame Buffer 1 does not contain valid image data.
When this bit = 1b, MAIN Frame Buffer 1 contains valid image data.

bit 0 — MAIN Frame Buffer 0 Ready
This bit only has an effect when MAIN window double-buffering is configured for software control, REG[09D8h] bit 0 = 0b.
For Writes:
Writing a 0b to this bit has no effect.
Writing a 1b to this bit sets this bit to 1b and indicates that the MAIN Frame Buffer 0 image data is ready for reading by the Blending Engine. Once this bit is set to 1b, it remains at 1b until it is reset by the Blending Engine.
For Reads:
When this bit = 0b, MAIN Frame Buffer 0 does not contain valid image data.
When this bit = 1b, MAIN Frame Buffer 0 contains valid image data.

REG[0944h] MAIN Blank Color Blue Register Default = 00h							Read/Write
MAIN Blank Color Blue bits 7-0							
7	6	5	4	3	2	1	0

REG[0945h] MAIN Blank Color Green Register Default = 00h							Read/Write
MAIN Blank Color Green bits 7-0							
7	6	5	4	3	2	1	0

REG[0946h] MAIN Blank Color Red Register Default = 00h							Read/Write
MAIN Blank Color Red bits 7-0							
7	6	5	4	3	2	1	0

REG[0946h] bits 7-0 MAIN Blank Color Red bits [7:0]
REG[0945h] bits 7-0 MAIN Blank Color Green bits [7:0]
REG[0944h] bits 7-0 MAIN Blank Color Blue bits [7:0]

When the MAIN Window Blank bit is set (REG[0940h] bit 0 = 1b), these bits specify the RGB components of the color that the Blending Engine replaces MAIN window image data with.

If the MAIN Window Pixel Format is RGB 8:8:8 (REG[940h] bits 3-2 = 10b).
REG[0946h] bits 7-0 = RED
REG[0945h] bits 7-0 = GREEN
REG[0944h] bits 7-0 = BLUE

If the MAIN Window Pixel Format is RGB 5:6:5 (REG[0940h] bits 3-2 = 01b).
REG[0946h] bits 7-3 = RED
REG[0945h] bits 7-2 = GREEN
REG[0944h] bits 7-3 = BLUE

If the MAIN Window Pixel Format is RGB 3:3:2 (REG[0940h] bits 3-2 = 00b).
REG[0946h] bits 7-5 = RED
REG[0945h] bits 7-5 = GREEN
REG[0944h] bits 7-6 = BLUE

REG[0948h] MAIN Window Frame Buffer 0 Address Register 0 Default = 00h							Read/Write
MAIN Window Frame Buffer 0 Address bits 7-0							
7	6	5	4	3	2	1	0

REG[0949h] MAIN Window Frame Buffer 0 Address Register 1 Default = 00h							Read/Write
MAIN Window Frame Buffer 0 Address bits 15-8							
7	6	5	4	3	2	1	0

REG[094Ah] MAIN Window Frame Buffer 0 Address Register 2 Default = 00h							Read/Write
MAIN Window Frame Buffer 0 Address bits 23-16							
7	6	5	4	3	2	1	0

REG[094Bh] MAIN Window Frame Buffer 0 Address Register 3 Default = 10h							Read/Write
MAIN Window Frame Buffer 0 Address bits 31-24							
7	6	5	4	3	2	1	0

REG[094Bh] bits 7-0
REG[094Ah] bits 7-0
REG[0949h] bits 7-0
REG[0948h] bits 7-0 MAIN Window Frame Buffer 0 Address bits [31:0]

These bits specify the memory start address for MAIN Window Frame Buffer 0 which is used for input image data to the Blending Engine. These bits must be set such that the start address is 8 byte (64-bit) aligned.

REG[094Ch] MAIN Window Frame Buffer 1 Address Register 0 Default = 00h Read/Write							
MAIN Window Frame Buffer 1 Address bits 7-0							
7	6	5	4	3	2	1	0

REG[094Dh] MAIN Window Frame Buffer 1 Address Register 1 Default = 00h Read/Write							
MAIN Window Frame Buffer 1 Address bits 15-8							
7	6	5	4	3	2	1	0

REG[094Eh] MAIN Window Frame Buffer 1 Address Register 2 Default = 00h Read/Write							
MAIN Window Frame Buffer 1 Address bits 23-16							
7	6	5	4	3	2	1	0

REG[094Fh] MAIN Window Frame Buffer 1 Address Register 3 Default = 10h Read/Write							
MAIN Window Frame Buffer 1 Address bits 31-24							
7	6	5	4	3	2	1	0

REG[094Fh] bits 7-0
REG[094Eh] bits 7-0
REG[094Dh] bits 7-0
REG[094Ch] bits 7-0 MAIN Window Frame Buffer 1 Address bits [31:0]

These bits specify the memory start address for MAIN Window Frame Buffer 1which is used for input image data to the Blending Engine. These bits must be set such that the start address is 8 byte (64-bit) aligned.

REG[0950h] MAIN Window Width Register 0 Default = 40h							Read/Write
MAIN Window Width bits 7-0							
7	6	5	4	3	2	1	0

REG[0951h] MAIN Window Width Register 1 Default = 01h							Read/Write
n/a					MAIN Window Width bits 10-8		
7	6	5	4	3	2	1	0

REG[0951h] bits 2-0
REG[0950h] bits 7-0 MAIN Window Width bits [10:0]
These bits specify the width of the MAIN window, in pixels.

Note
For tiled frame mode, the image width must be a multiple of 8 pixels.

REG[0952h] MAIN Window Height Register 0 Default = F0h							Read/Write
MAIN Window Height bits 7-0							
7	6	5	4	3	2	1	0

REG[0953h] MAIN Window Height Register 1 Default = 00h							Read/Write
n/a					MAIN Window Height bits 10-8		
7	6	5	4	3	2	1	0

REG[0953h] bits 2-0
REG[0952h] bits 7-0 MAIN Window Height bits [10:0]
These bits specify the height of the MAIN window, in pixels.

REG[0954h] MAIN Window Virtual Width Register 0 Default = 40h							Read/Write
MAIN Window Virtual Width bits 7-0							
7	6	5	4	3	2	1	0

REG[0955h] MAIN Window Virtual Width Register 1 Default = 01h							Read/Write
n/a			MAIN Window Virtual Width bits 12-8				
7	6	5	4	3	2	1	0

REG[0955h] bits 4-0
REG[0954h] bits 7-0 MAIN Window Virtual Width bits [12:0]
These bits specify the width of the MAIN window virtual image, in pixels. For an example showing a virtual source window, see Figure 13-9: "Virtual Source Window Example," on page 464.

Note
1. The Main window virtual width must be set such that the virtual width multiplied by the pixel format (in bpp, see REG[0940h] bits 3-2) is divisible by 64.
2. For tiled frame mode, the image virtual width must be a multiple of 8 pixels.

REG[095Ah] MAIN Input X Offset Register 0 Default = 00h							Read/Write
MAIN Input X Offset bits 7-0							
7	6	5	4	3	2	1	0

REG[095Bh] MAIN Input X Offset Register 1 Default = 00h							Read/Write
n/a			MAIN Input X Offset bits 12-8				
7	6	5	4	3	2	1	0

REG[095Bh] bits 4-0
REG[095Ah] bits 7-0 MAIN Input X Offset bits [12:0]
These bits specify the X offset of the top left corner of the MAIN window relative to the top left corner of the MAIN window virtual image, in pixels. For an example showing a virtual source window, see Figure 13-9: "Virtual Source Window Example," on page 464.

REG[095Ch] MAIN Input Y Offset Register 0 Default = 00h Read/Write							
MAIN Input Y Offset bits 7-0							
7	6	5	4	3	2	1	0

REG[095Dh] MAIN Input Y Offset Register 1 Default = 00h Read/Write							
n/a			MAIN Input Y Offset bits 12-8				
7	6	5	4	3	2	1	0

REG[095Dh] bits 4-0
REG[095Ch] bits 7-0 MAIN Input Y Offset bits [12:0]
These bits specify the Y offset of the top left corner of the MAIN window relative to the top left corner of the MAIN window virtual image, in pixels. For an example showing a virtual source window, see Figure 13-9: "Virtual Source Window Example," on page 464.

REG[0960h] AUX Window Control Register Default = 00h Read/Write							
AUX Window Line Double Enable	AUX Horizontal Flip Enable	AUX Vertical Flip Enable	AUX Enable	AUX Window Pixel Format bits 1-0		AUX Window Fetch Mode	AUX Window Blank
7	6	5	4	3	2	1	0

bit 7 AUX Window Line Double Enable
This bit controls "line double" mode which is typically used for displaying interlaced images from the camera interface. When line doubling is enabled, each line of the input image stored in the SDRAM is read twice.
When this bit = 0b, AUX window line doubling is disabled.
When this bit = 1b, AUX window line doubling is enabled.

bit 6 AUX Horizontal Flip Enable
This bit determines whether the image data input from the AUX window is flipped around the Y axis (horizontal). This bit must be set to 0b when the AUX window fetch uses "tiled-frame" mode, REG[0960h] bit 1 = 1b.
When this bit = 0b, the AUX image data is not horizontally flipped (disabled).
When this bit = 1b, the AUX image data is horizontally flipped (enabled).

Note

If the OSD overlays the AUX window (REG[09A0h] bits 1-0 = 10b) when AUX Horizontal Flip is enabled, the relative position of the OSD window is flipped. However, the image in the OSD window is NOT flipped and is still controlled by the OSD flip enable bits, REG[0980h] bits 6-5.

bit 5 AUX Vertical Flip Enable
This bit determines whether the image data input from the AUX window is flipped around the X axis (vertical). This bit must be set to 0b when the AUX window fetch uses “tiled-frame” mode, REG[0960h] bit 1 = 1b.
When this bit = 0b, the AUX image data is not vertically flipped (disabled).
When this bit = 1b, the AUX image data is vertically flipped (enabled).

Note
If the OSD overlays the AUX window (REG[09A0h] bits 1-0 = 10b) when AUX Vertical Flip is enabled, the relative position of the OSD window is flipped. However, the image in the OSD window is NOT flipped and is still controlled by the OSD flip enable bits, REG[0980h] bits 6-5.

bit 4 AUX Enable
This bit only has an effect when Blend Mode 0 is selected, REG[09A0h] bits 1-0 = 00b.
This bit controls whether the AUX window is displayed (enabled) or not (disabled).
When this bit = 0b, the AUX window is disabled.
When this bit = 1b, the AUX window is enabled.

Note
If hardware frame control is selected for the AUX window (REG[09D9h] bit 0 = 1b), it must be disabled before the AUX window can be disabled. The following sequence is recommended.

1. Disable hardware frame control, REG[09D9h] bit 0 = 0b.
2. Wait 1 frame.
3. Disable the AUX window, REG[0960h] bit 4 = 0b.

bits 3-2 AUX Window Pixel Format bits [1:0]
These bits determine the RGB pixel format of the AUX window image data that is input to the Blending Engine.

Table 10-36: AUX Window Pixel Format Selection

REG[0960h] bits 3-2	Pixel Format
00b	8 bpp (RGB 3:3:2)
01b	16 bpp (RGB 5:6:5)
10b	24 bpp (RGB 8:8:8)
11b	Reserved

bit 1 AUX Window Fetch Mode
This bit specifies how the AUX window image data is stored in memory. For details on the memory organization methods, see Section 13.3, “Memory Organization of Frames” on page 470.
When this bit = 0b, AUX window fetch uses “line-by-line” mode to read from SDRAM.
When this bit = 1b, AUX window fetch uses “tiled-frame” mode to read from SDRAM.

Note
For tiled frame mode, the image width and virtual width must be a multiple of 8 pixels and the AUX window image data must not be flipped (REG[0960h] bit 6 = 0b and bit 5 = 0b).

bit 0 AUX Window Blank
This bit controls the AUX window blank function. The blank function replaces the image data input to the Blending Engine from the AUX window with the color specified by the AUX Blank Color registers, REG[0964h] ~ REG[0966h].
When this bit = 0b, the AUX window image data is read normally (not blanked).
When this bit = 1b, the AUX window image data is “blanked” with the specified color.

REG[0962h] AUX Window Frame Control/Status Register Default = 00h							Read/Write
n/a		AUX Frame Buffer 1 Ready Clear (WO)	AUX Frame Buffer 0 Ready Clear (WO)	n/a	AUX Window Current Frame Status (RO)	AUX Frame Buffer 1 Ready	AUX Frame Buffer 0 Ready
7	6	5	4	3	2	1	0

bit 5 AUX Frame Buffer 1 Ready Clear (Write Only)
This bit is used to manually clear the AUX Frame Buffer 1 Ready bit, REG[0962h] bit 1.
Writing a 0b to this bit has no effect.
Writing a 1b to this bit clears the AUX Frame Buffer 1 Ready bit.

bit 4 AUX Frame Buffer 0 Ready Clear (Write Only)
This bit is used to manually clear the AUX Frame Buffer 0 Ready bit, REG[0962h] bit 0.
Writing a 0b to this bit has no effect.
Writing a 1b to this bit clears the AUX Frame Buffer 0 Ready bit.

bit 2 AUX Window Current Frame Status (Read Only)
This bit indicates which AUX frame buffer is currently being read by the Blending Engine.
When this bit = 0b, AUX Frame Buffer 0 is being read by the Blending Engine.
When this bit = 1b, AUX Frame Buffer 1 is being read by the Blending Engine.

Note
When the AUX window is disabled and then re-enabled using the AUX Enable bit (REG[0960h] bit 4), the hardware always sets the Current Frame status to 0b and checks the AUX Frame Buffer 0 Ready bit first. Therefore before re-enabling the AUX window, the AUX window image stream must be reset to start with Buffer 0, the AUX Frame Buffer 0/1 Ready bits must be cleared (see REG[0962h] bits 5-4), and the AUX Frame Buffer 0 Ready bit must be set to 1b (REG[0962h] bit 0 = 1b).

bit 1 AUX Frame Buffer 1 Ready
This bit only has an effect when AUX window double-buffering is configured for software control, REG[09D9h] bit 0 = 0b.
For Writes:
Writing a 0b to this bit has no effect.
Writing a 1b to this bit sets this bit to 1b and indicates that the AUX Frame Buffer 1 image data is ready for reading by the Blending Engine. Once this bit is set to 1b, it remains at 1b until it is reset by the Blending Engine.
For Reads:
When this bit = 0b, AUX Frame Buffer 1 does not contain valid image data.
When this bit = 1b, AUX Frame Buffer 1 contains valid image data.

bit 0	AUX Frame Buffer 0 Ready This bit only has an effect when AUX window double-buffering is configured for software control, REG[09D9h] bit 0 = 0b. For Writes: Writing a 0b to this bit has no effect. Writing a 1b to this bit sets this bit to 1b and indicates that the AUX Frame Buffer 0 image data is ready for reading by the Blending Engine. Once this bit is set to 1b, it remains at 1b until it is reset by the Blending Engine. For Reads: When this bit = 0b, AUX Frame Buffer 0 does not contain valid image data. When this bit = 1b, AUX Frame Buffer 0 contains valid image data.

REG[0964h] AUX Blank Color Blue Register
Default = 00h — Read/Write

7	6	5	4	3	2	1	0
AUX Blank Color Blue bits 7-0							

REG[0965h] AUX Blank Color Green Register
Default = 00h — Read/Write

7	6	5	4	3	2	1	0
AUX Blank Color Green bits 7-0							

REG[0966h] AUX Blank Color Red Register
Default = 00h — Read/Write

7	6	5	4	3	2	1	0
AUX Blank Color Red bits 7-0							

REG[0966h] bits 7-0 AUX Blank Color Red bits [7:0]
REG[0965h] bits 7-0 AUX Blank Color Green bits [7:0]
REG[0964h] bits 7-0 AUX Blank Color Blue bits [7:0]

When the AUX Window Blank bit is set (REG[0960h] bit 0 = 1b), these bits specify the RGB components of the color that the Blending Engine replaces AUX window image data with.

If the AUX Window Pixel Format is RGB 8:8:8 (REG[960h] bits 3-2 = 10b).
REG[0966h] bits 7-0 = RED
REG[0965h] bits 7-0 = GREEN
REG[0964h] bits 7-0 = BLUE

If the AUX Window Pixel Format is RGB 5:6:5 (REG[0960h] bits 3-2 = 01b).
REG[0966h] bits 7-3 = RED
REG[0965h] bits 7-2 = GREEN
REG[0964h] bits 7-3 = BLUE

If the AUX Window Pixel Format is RGB 3:3:2 (REG[0960h] bits 3-2 = 00b).
REG[0966h] bits 7-5 = RED
REG[0965h] bits 7-5 = GREEN
REG[0964h] bits 7-6 = BLUE

REG[0968h] AUX Window Frame Buffer 0 Address Register 0 Default = 00h							Read/Write
AUX Window Frame Buffer 0 Address bits 7-0							
7	6	5	4	3	2	1	0

REG[0969h] AUX Window Frame Buffer 0 Address Register 1 Default = 00h							Read/Write
AUX Window Frame Buffer 0 Address bits 15-8							
7	6	5	4	3	2	1	0

REG[096Ah] AUX Window Frame Buffer 0 Address Register 2 Default = 00h							Read/Write
AUX Window Frame Buffer 0 Address bits 23-16							
7	6	5	4	3	2	1	0

REG[096Bh] AUX Window Frame Buffer 0 Address Register 3 Default = 10h							Read/Write
AUX Window Frame Buffer 0 Address bits 31-24							
7	6	5	4	3	2	1	0

REG[096Bh] bits 7-0
REG[096Ah] bits 7-0
REG[0969h] bits 7-0
REG[0968h] bits 7-0 AUX Window Frame Buffer 0 Address bits [31:0]

These bits specify the memory start address for AUX Window Frame Buffer 0 which is used for input image data to the Blending Engine. These bits must be set such that the start address is 8 byte (64-bit) aligned.

REG[096Ch] AUX Window Frame Buffer 1 Address Register 0							
Default = 00h							Read/Write
AUX Window Frame Buffer 1 Address bits 7-0							
7	6	5	4	3	2	1	0

REG[096Dh] AUX Window Frame Buffer 1 Address Register 1							
Default = 00h							Read/Write
AUX Window Frame Buffer 1 Address bits 15-8							
7	6	5	4	3	2	1	0

REG[096Eh] AUX Window Frame Buffer 1 Address Register 2							
Default = 00h							Read/Write
AUX Window Frame Buffer 1 Address bits 23-16							
7	6	5	4	3	2	1	0

REG[096Fh] AUX Window Frame Buffer 1 Address Register 3							
Default = 10h							Read/Write
AUX Window Frame Buffer 1 Address bits 31-24							
7	6	5	4	3	2	1	0

REG[096Fh] bits 7-0
REG[096Eh] bits 7-0
REG[096Dh] bits 7-0
REG[096Ch] bits 7-0 AUX Window Frame Buffer 1 Address bits [31:0]

These bits specify the memory start address for AUX Window Frame Buffer 1 which is used for input image data to the Blending Engine. These bits must be set such that the start address is 8 byte (64-bit) aligned.

REG[0970h] AUX Window Width Register 0 Default = 40h							Read/Write
AUX Window Width bits 7-0							
7	6	5	4	3	2	1	0

REG[0971h] AUX Window Width Register 1 Default = 01h							Read/Write
n/a					AUX Window Width bits 10-8		
7	6	5	4	3	2	1	0

REG[0971h] bits 2-0
REG[0970h] bits 7-0 AUX Window Width bits [10:0]
These bits specify the width of the AUX window, in pixels.

Note
For tiled frame mode, the image width must be a multiple of 8 pixels.

REG[0972h] AUX Window Height Register 0 Default = F0h							Read/Write
AUX Window Height bits 7-0							
7	6	5	4	3	2	1	0

REG[0973h] AUX Window Height Register 1 Default = 00h							Read/Write
n/a					AUX Window Height bits 10-8		
7	6	5	4	3	2	1	0

REG[0973h] bits 2-0
REG[0972h] bits 7-0 AUX Window Height bits [10:0]
These bits specify the height of the AUX window, in pixels.

REG[0974h] AUX Window Virtual Width Register 0 Default = 40h							Read/Write
AUX Window Virtual Width bits 7-0							
7	6	5	4	3	2	1	0

REG[0975h] AUX Window Virtual Width Register 1 Default = 01h							Read/Write
n/a			AUX Window Virtual Width bits 12-8				
7	6	5	4	3	2	1	0

REG[0975h] bits 4-0
REG[0974h] bits 7-0 AUX Window Virtual Width bits [12:0]

These bits specify the width of the AUX window virtual image, in pixels. For an example showing a virtual source window, see Figure 13-9: "Virtual Source Window Example," on page 464.

Note

1. The AUX window virtual width must be set such that the virtual width multiplied by the pixel format (in bpp, see REG[0960h] bits 3-2) is divisible by 64.
2. For tiled frame mode, the image virtual width must be a multiple of 8 pixels.

REG[0976h] AUX Window X Offset Register 0 Default = 00h							Read/Write
AUX Window X Offset bits 7-0							
7	6	5	4	3	2	1	0

REG[0977h] AUX Window X Offset Register 1 Default = 00h							Read/Write
n/a					AUX Window X Offset bits 10-8		
7	6	5	4	3	2	1	0

REG[0977h] bits 2-0
REG[0976h] bits 7-0 AUX Window X Offset bits [10:0]

These bits only have an effect when Blend Mode 0 is selected, REG[09A0h] bits 1-0 = 00b. These bits specify the X offset of the top left corner of the AUX window relative to the top left corner of the LCD display, in pixels.

REG[0978h] AUX Window Y Offset Register 0 Default = 00h Read/Write							
AUX Window Y Offset bits 7-0							
7	6	5	4	3	2	1	0

REG[0979h] AUX Window Y Offset Register 1 Default = 00h Read/Write							
n/a					AUX Window Y Offset bits 10-8		
7	6	5	4	3	2	1	0

REG[0979h] bits 2-0
REG[0978h] bits 7-0 AUX Window Y Offset bits [10:0]
These bits only have an effect when Blend Mode 0 is selected, REG[09A0h] bits 1-0 = 00b. These bits specify the Y offset of the top left corner of the AUX window relative to the top left corner of the LCD display, in pixels.

REG[097Ah] AUX Input X Offset Register 0 Default = 00h Read/Write							
AUX Input X Offset bits 7-0							
7	6	5	4	3	2	1	0

REG[097Bh] AUX Input X Offset Register 1 Default = 00h Read/Write							
n/a			AUX Input X Offset bits 12-8				
7	6	5	4	3	2	1	0

REG[097Bh] bits 4-0
REG[097Ah] bits 7-0 AUX Input X Offset bits [12:0]
These bits specify the X offset of the top left corner of the AUX window relative to the top left corner of the AUX window virtual image, in pixels. For an example showing a virtual source window, see Figure 13-9: “Virtual Source Window Example,” on page 464.

REG[097Ch] AUX Input Y Offset Register 0 Default = 00h Read/Write							
AUX Input Y Offset bits 7-0							
7	6	5	4	3	2	1	0

REG[097Dh] AUX Input Y Offset Register 1 Default = 00h Read/Write							
n/a			AUX Input Y Offset bits 12-8				
7	6	5	4	3	2	1	0

REG[097Dh] bits 4-0
REG[097Ch] bits 7-0 AUX Input Y Offset bits [12:0]
These bits specify the Y offset of the top left corner of the AUX window relative to the top left corner of the AUX window virtual image, in pixels, For an example showing a virtual source window, see Figure 13-9: “Virtual Source Window Example,” on page 464.

REG[0980h] OSD Window Control Register Default = 00h							Read/Write
OSD Window Line Double Enable	OSD Horizontal Flip Enable	OSD Vertical Flip Enable	OSD Enable	OSD Window Pixel Format bits 1-0		OSD Window Fetch Mode	OSD Window Blank
7	6	5	4	3	2	1	0

bit 7 OSD Window Line Double Enable
This bit controls "line double" mode which is typically used for displaying interlaced images from the camera interface. When line doubling is enabled, each line of the input image stored in the SDRAM is read twice.
When this bit = 0b, OSD window line doubling is disabled.
When this bit = 1b, OSD window line doubling is enabled.

bit 6 OSD Horizontal Flip Enable
This bit determines whether the image data input from the OSD window is flipped around the Y axis (horizontal). This bit must be set to 0b when the OSD window fetch uses "tiled-frame" mode, REG[0980h] bit 1 = 1b.
When this bit = 0b, the OSD image data is not horizontally flipped (disabled).
When this bit = 1b, the OSD image data is horizontally flipped (enabled).

bit 5 OSD Vertical Flip Enable
This bit determines whether the image data input from the OSD window is flipped around the X axis (vertical). This bit must be set to 0b when the OSD window fetch uses "tiled-frame" mode, REG[0980h] bit 1 = 1b.
When this bit = 0b, the OSD image data is not vertically flipped (disabled).
When this bit = 1b, the OSD image data is vertically flipped (enabled).

bit 4 OSD Enable
This bit only has an effect when Blend Mode 0, 1, or 2 is selected, REG[09A0h] bits 1-0 = 00b or 01b or 10b. This bit controls whether the OSD window is displayed (enabled) or not (disabled).
When this bit = 0b, the OSD window is disabled.
When this bit = 1b, the OSD window is enabled.

Note

If hardware frame control is selected for the OSD window (REG[09DAh] bit 0 = 1b), it must be disabled before the OSD window can be disabled. The following sequence is recommended.

1. Disable hardware frame control, REG[09DAh] bit 0 = 0b.
2. Wait 1 frame.
3. Disable the OSD window, REG[0980h] bit 4 = 0b.

bits 3-2 OSD Window Pixel Format bits [1:0]
These bits determine the RGB or ARGB pixel format of the OSD window image data that is input to the Blending Engine.

Table 10-37: OSD Window Pixel Format Selection

REG[09A0h] bit 3 (Alpha Format)	REG[0980h] bits 3-2	Pixel Format
0b	00b	8 bpp (RGB 3:3:2)
	01b	16 bpp (RGB 5:6:5)
	10b	24 bpp (RGB 8:8:8)
	11b	Reserved
1b	00b	16 bpp (ARGB 4:4:4:4)
	01b	16 bpp (ARGB 1:5:5:5)
	10b	24 bpp (ARGB 8:5:6:5)
	11b	Reserved

Note
When Blend Mode 3 is selected (REG[09A0h] bits 1-0 = 11b), ARGB pixel formats are not supported for the OSD window.

bit 1 OSD Window Fetch Mode
This bit specifies how the OSD window image data is stored in memory. For details on the memory organization methods, see Section 13.3, “Memory Organization of Frames” on page 470.
When this bit = 0b, OSD window fetch uses “line-by-line” mode to read from SDRAM.
When this bit = 1b, OSD window fetch uses “tiled-frame” mode to read from SDRAM.

Note
For tiled frame mode, the image width and virtual width must be a multiple of 8 pixels and the OSD window image data must not be flipped (REG[0980h] bit 6 = 0b and bit 5 = 0b).

bit 0 OSD Window Blank
This bit controls the OSD window blank function. The blank function replaces the image data input to the Blending Engine from the OSD window with the color specified by the OSD Blank Color registers, REG[0984h] ~ REG[0986h].
When this bit = 0b, the OSD window image data is read normally (not blanked).
When this bit = 1b, the OSD window image data is “blanked” with the specified color.

Note
If the OSD window is blanked while OSD Alpha Format is enabled (REG[09A0h] bit 3 = 1b), the RGB blank color is specified by the OSD Blank Color registers (REG[0984h] ~ REG[0986h]) and the alpha ratio is specified by the OSD Alpha Blend Ratio register (REG[09A1h]).

REG[0982h] OSD Window Frame Control/Status Register Default = 00h							Read/Write
n/a		OSD Frame Buffer 1 Ready Clear (WO)	OSD Frame Buffer 0 Ready Clear (WO)	n/a	OSD Window Current Frame Status (RO)	OSD Frame Buffer 1 Ready	OSD Frame Buffer 0 Ready
7	6	5	4	3	2	1	0

bit 5 OSD Frame Buffer 1 Ready Clear (Write Only)
This bit is used to manually clear the OSD Frame Buffer 1 Ready bit, REG[0982h] bit 1.
Writing a 0b to this bit has no effect.
Writing a 1b to this bit clears the OSD Frame Buffer 1 Ready bit.

bit 4 OSD Frame Buffer 0 Ready Clear (Write Only)
This bit is used to manually clear the OSD Frame Buffer 0 Ready bit, REG[0982h] bit 0.
Writing a 0b to this bit has no effect.
Writing a 1b to this bit clears the OSD Frame Buffer 0 Ready bit.

bit 2 OSD Window Current Frame Status (Read Only)
This bit indicates which OSD frame buffer is currently being read by the Blending Engine.
When this bit = 0b, OSD Frame Buffer 0 is being read by the Blending Engine.
When this bit = 1b, OSD Frame Buffer 1 is being read by the Blending Engine.

Note

When the OSD window is disabled and then re-enabled using the OSD Enable bit (REG[0980h] bit 4), the hardware always sets the Current Frame status to 0b and checks the OSD Frame Buffer 0 Ready bit first. Therefore before re-enabling the OSD window, the OSD window image stream must be reset to start with Buffer 0, the OSD Frame Buffer 0/1 Ready bits must be cleared (see REG[0982h] bits 5-4), and the OSD Frame Buffer 0 Ready bit must be set to 1b (REG[0982h] bit 0 = 1b).

bit 1	OSD Frame Buffer 1 Ready This bit only has an effect when OSD window double-buffering is configured for software control, REG[09DAh] bit 0 = 0b. For Writes: Writing a 0b to this bit has no effect. Writing a 1b to this bit sets this bit to 1b and indicates that the OSD Frame Buffer 1 image data is ready for reading by the Blending Engine. Once this bit is set to 1b, it remains at 1b until it is reset by the Blending Engine. For Reads: When this bit = 0b, OSD Frame Buffer 1 does not contain valid image data. When this bit = 1b, OSD Frame Buffer 1 contains valid image data.
bit 0	OSD Frame Buffer 0 Ready This bit only has an effect when OSD window double-buffering is configured for software control, REG[09DAh] bit 0 = 0b. For Writes: Writing a 0b to this bit has no effect. Writing a 1b to this bit sets this bit to 1b and indicates that the OSD Frame Buffer 0 image data is ready for reading by the Blending Engine. Once this bit is set to 1b, it remains at 1b until it is reset by the Blending Engine. For Reads: When this bit = 0b, OSD Frame Buffer 0 does not contain valid image data. When this bit = 1b, OSD Frame Buffer 0 contains valid image data.

REG[0984h] OSD Blank Color Blue Register Default = 00h							Read/Write
OSD Blank Color Blue bits 7-0							
7	6	5	4	3	2	1	0

REG[0985h] OSD Blank Color Green Register Default = 00h							Read/Write
OSD Blank Color Green bits 7-0							
7	6	5	4	3	2	1	0

REG[0986h] OSD Blank Color Red Register Default = 00h							Read/Write
OSD Blank Color Red bits 7-0							
7	6	5	4	3	2	1	0

REG[0986h] bits 7-0 OSD Blank Color Red bits [7:0]
REG[0985h] bits 7-0 OSD Blank Color Green bits [7:0]
REG[0984h] bits 7-0 OSD Blank Color Blue bits [7:0]

When the OSD Window Blank bit is set (REG[0980h] bit 0 = 1b), these bits specify the RGB components of the color that the Blending Engine replaces OSD window image data with.

If the OSD Window Pixel Format is RGB 8:8:8 (REG[0980h] bits 3-2 = 10b).

REG[0986h] bits 7-0 = RED
REG[0985h] bits 7-0 = GREEN
REG[0984h] bits 7-0 = BLUE

If the OSD Window Pixel Format is RGB 5:6:5 (REG[0980h] bits 3-2 = 01b).

REG[0986h] bits 7-3 = RED
REG[0985h] bits 7-2 = GREEN
REG[0984h] bits 7-3 = BLUE

If the OSD Window Pixel Format is RGB 3:3:2 (REG[0980h] bits 3-2 = 00b).

REG[0986h] bits 7-5 = RED
REG[0985h] bits 7-5 = GREEN
REG[0984h] bits 7-6 = BLUE

Note

If the OSD window is blanked (REG[0980h] bit 0 = 1b) while OSD Alpha Format is enabled (REG[09A0h] bit 3 = 1b), the RGB blank color is specified by the OSD Blank Color registers (REG[0984h] ~ REG[0986h]) and the alpha ratio is specified by the OSD Alpha Blend Ratio register (REG[09A1h]).

REG[0988h] OSD Window Frame Buffer 0 Address Register 0 Default = 00h							Read/Write
OSD Window Frame Buffer 0 Address bits 7-0							
7	6	5	4	3	2	1	0

REG[0989h] OSD Window Frame Buffer 0 Address Register 1 Default = 00h							Read/Write
OSD Window Frame Buffer 0 Address bits 15-8							
7	6	5	4	3	2	1	0

REG[098Ah] OSD Window Frame Buffer 0 Address Register 2 Default = 00h							Read/Write
OSD Window Frame Buffer 0 Address bits 23-16							
7	6	5	4	3	2	1	0

REG[098Bh] OSD Window Frame Buffer 0 Address Register 3 Default = 10h							Read/Write
OSD Window Frame Buffer 0 Address bits 31-24							
7	6	5	4	3	2	1	0

REG[098Bh] bits 7-0
REG[098Ah] bits 7-0
REG[0989h] bits 7-0
REG[0988h] bits 7-0 OSD Window Frame Buffer 0 Address bits [31:0]

These bits specify the memory start address for OSD Window Frame Buffer 0 which is used for input image data to the Blending Engine. These bits must be set such that the start address is 8 byte (64-bit) aligned.

REG[098Ch] OSD Window Frame Buffer 1 Address Register 0 Default = 00h Read/Write							
OSD Window Frame Buffer 1 Address bits 7-0							
7	6	5	4	3	2	1	0

REG[098Dh] OSD Window Frame Buffer 1 Address Register 1 Default = 00h Read/Write							
OSD Window Frame Buffer 1 Address bits 15-8							
7	6	5	4	3	2	1	0

REG[098Eh] OSD Window Frame Buffer 1 Address Register 2 Default = 00h Read/Write							
OSD Window Frame Buffer 1 Address bits 23-16							
7	6	5	4	3	2	1	0

REG[098Fh] OSD Window Frame Buffer 1 Address Register 3 Default = 10h Read/Write							
OSD Window Frame Buffer 1 Address bits 31-24							
7	6	5	4	3	2	1	0

REG[098Fh] bits 7-0
REG[098Eh] bits 7-0
REG[098Dh] bits 7-0
REG[098Ch] bits 7-0 OSD Window Frame Buffer 1 Address bits [31:0]

These bits specify the memory start address for OSD Window Frame Buffer 1 which is used for input image data to the Blending Engine. These bits must be set such that the start address is 8 byte (64-bit) aligned.

REG[0990h] OSD Window Width Register 0 Default = 40h							Read/Write
OSD Window Width bits 7-0							
7	6	5	4	3	2	1	0

REG[0991h] OSD Window Width Register 1 Default = 01h							Read/Write
n/a					OSD Window Width bits 10-8		
7	6	5	4	3	2	1	0

REG[0991h] bits 2-0
REG[0990h] bits 7-0 OSD Window Width bits [10:0]
These bits specify the width of the OSD window, in pixels.

Note
For tiled frame mode, the image width must be a multiple of 8 pixels.

REG[0992h] OSD Window Height Register 0 Default = F0h							Read/Write
OSD Window Height bits 7-0							
7	6	5	4	3	2	1	0

REG[0993h] OSD Window Height Register 1 Default = 00h							Read/Write
n/a					OSD Window Height bits 10-8		
7	6	5	4	3	2	1	0

REG[0993h] bits 2-0
REG[0992h] bits 7-0 OSD Window Height bits [10:0]
These bits specify the height of the OSD window, in pixels.

REG[0994h] OSD Window Virtual Width Register 0 Default = 40h							Read/Write
OSD Window Virtual Width bits 7-0							
7	6	5	4	3	2	1	0

REG[0995h] OSD Window Virtual Width Register 1 Default = 01h							Read/Write
n/a			OSD Window Virtual Width bits 12-8				
7	6	5	4	3	2	1	0

REG[0995h] bits 4-0
REG[0994h] bits 7-0 OSD Window Virtual Width bits [12:0]

These bits specify the width of the OSD window virtual image, in pixels. For an example showing a virtual source window, see Figure 13-9: “Virtual Source Window Example,” on page 464.

Note

1. The OSD window virtual width must be set such that the virtual width multiplied by the pixel format (in bpp, see REG[0980h] bits 3-2) is divisible by 64.
2. For tiled frame mode, the image virtual width must be a multiple of 8 pixels.

REG[0996h] OSD Window X Offset Register 0 Default = 00h							Read/Write
OSD Window X Offset bits 7-0							
7	6	5	4	3	2	1	0

REG[0997h] OSD Window X Offset Register 1 Default = 00h							Read/Write
n/a					OSD Window X Offset bits 10-8		
7	6	5	4	3	2	1	0

REG[0997h] bits 2-0
REG[0996h] bits 7-0 OSD Window X Offset bits [10:0]

These bits only have an effect when Blend Mode 0, 1, or 2 is selected, REG[09A0h] bits 1-0 = 00b or 01b or 10b. These bits specify the X offset of the top left corner of the OSD window relative to the top left corner of the LCD display, in pixels.

REG[0998h] OSD Window Y Offset Register 0 Default = 00h							Read/Write
OSD Window Y Offset bits 7-0							
7	6	5	4	3	2	1	0

REG[0999h] OSD Window Y Offset Register 1 Default = 00h							Read/Write
n/a					OSD Window Y Offset bits 10-8		
7	6	5	4	3	2	1	0

REG[0999h] bits 2-0
REG[0998h] bits 7-0 OSD Window Y Offset bits [10:0]
These bits only have an effect when Blend Mode 0, 1, or 2 is selected, REG[09A0h] bits 1-0 = 00b or 01b or 10b. These bits specify the Y offset of the top left corner of the OSD window relative to the top left corner of the LCD display, in pixels.

REG[099Ah] OSD Input X Offset Register 0 Default = 00h							Read/Write
OSD Input X Offset bits 7-0							
7	6	5	4	3	2	1	0

REG[099Bh] OSD Input X Offset Register 1 Default = 00h							Read/Write
n/a			OSD Input X Offset bits 12-8				
7	6	5	4	3	2	1	0

REG[099Bh] bits 4-0
REG[099Ah] bits 7-0 OSD Input X Offset bits [12:0]
These bits specify the X offset of the top left corner of the OSD window relative to the top left corner of the OSD window virtual image, in pixels. For an example showing a virtual source window, see Figure 13-9: “Virtual Source Window Example,” on page 464.

REG[099Ch] OSD Input Y Offset Register 0 Default = 00h							Read/Write
OSD Input Y Offset bits 7-0							
7	6	5	4	3	2	1	0

REG[099Dh] OSD Input Y Offset Register 1 Default = 00h							Read/Write
n/a			OSD Input Y Offset bits 12-8				
7	6	5	4	3	2	1	0

REG[099Dh] bits 4-0
REG[099Ch] bits 7-0 OSD Input Y Offset bits [12:0]
These bits specify the Y offset of the top left corner of the OSD window relative to the top left corner of the OSD window virtual image, in pixels, For an example showing a virtual source window, see Figure 13-9: “Virtual Source Window Example,” on page 464.

REG[09A0h] Blending Engine Control Register Default = 00h — Read/Write							
Reserved	n/a		ARGB 1:5:5:5 Alpha Ratio Select	OSD Alpha Format Enable	AUX on Top	Blend Mode Select bits 1-0	
7	6	5	4	3	2	1	0

bit 7 — Reserved
This bit must be set to 0b.

bit 4 — ARGB 1:5:5:5 Alpha Ratio Select
When the OSD window is configured for ARGB 1:5:5:5 (REG[09A0h] bit 3 = 1b and REG[0980h] bits 3-2 = 01b), this bit selects the ratio used to alpha-blend the OSD window when the 1-bit alpha value is 1b. When the 1-bit alpha value is 0b, the ratio is 00% (00h).
When this bit = 0b, the 8-bit alpha blend ratio for ARGB 1:5:5:5 is 50% (80h).
When this bit = 1b, the 8-bit alpha blend ratio for ARGB 1:5:5:5 is 75% (C0h).

bit 3 — OSD Alpha Format Enable
This bit determines the method used for alpha-blending the OSD window.
When this bit = 0b, the OSD window pixel format is non-alpha (RGB 3:3:2, RGB 5:6:5, or RGB 8:8:8, see REG[0980h] bits 3-2). In this mode, the OSD window is alpha-blended using a common alpha ratio as specified by the OSD Alpha Blend Ratio register, REG[09A1h].
When this bit = 1b, the OSD window pixel format is alpha (ARGB 4:4:4:4, ARGB 1:5:5:5, or ARGB 8:5:6:5, see REG[0980h] bits 3-2). In this mode, the OSD window is alpha-blended using the alpha ratio for each pixel.

Note

1. If the OSD window is blanked (REG[0980h] bit 0 = 1b) while OSD Alpha Format is enabled, the RGB blank color is specified by the OSD Blank Color registers (REG[0984h] ~ REG[0986h]) and the alpha ratio is specified by the OSD Alpha Blend Ratio register (REG[09A1h]).
2. If OSD Alpha Format is enabled and OSD Transparency is enabled (REG[09A7h] bit 7 = 1b), only the RGB components of the pixel value are compared.

bit 2 — AUX on Top
This bit only has an effect when Blend Mode 0 is selected, REG[09A0h] bits 1-0 = 00b.
This bit determines whether the AUX or OSD window is on top.
When this bit = 0b, the OSD window is on top of the AUX window.
When this bit = 1b, the AUX window is on top of the OSD window.

Note

When the AUX window is on top, the OSD window is alpha-blended with the MAIN window only.

bits 1-0 Blend Mode Select bits [1:0]
These bits select the Blending Engine mode of operation. For details on each mode, see Section 13.2.2, “Blending Engine” on page 456.

Table 10-38: Blend Mode Selection

REG[09A0h] bits 1-0	Blend Mode	CH1OUT	CH2OUT	OSDOUT
00b	0	MAIN+AUX+OSD	Off	Off
01b	1	MAIN+OSD	AUX	Off
10b	2	MAIN	AUX+OSD	Off
11b	3	MAIN	AUX	OSD

REG[09A1h] OSD Alpha Blend Ratio Register
Default = FFh Read/Write

7	6	5	4	3	2	1	0
OSD Alpha Blend Ratio bits 7-0							

bits 7-0 OSD Alpha Blend Ratio bits [7:0]
When OSD Alpha Format is disabled (REG[09A0h] bit 3 = 0b), the OSD window is alpha-blended using the common alpha ratio specified by these bits. When the Alpha value is FFh, the OSD window is fully displayed. When the Alpha value is 00h, the OSD window is turned off. If the Alpha value changes from zero to non-zero, it turns on the OSD window and care must be taken by software to ensure that the frame double-buffering between the OSD window and its source image stream restarts at Buffer 0 (see note for REG[0982h] bit 2).

When the OSD window is blanked (REG[0980h] bit 0 is 1b), these bits specify the alpha blend ratio for all OSD window pixel formats. For RGB 3:3:2, RGB 5:6:5, and RGB 8:8:8, and ARGB 8:5:6:5 formats (see REG[09A0h] bit 3 and REG[0980h] bits 3-2), bits 7-0 of this register are used as the alpha blend ratio. For ARGB 1:5:5:5, bit 7 of this register is used as the 1-bit alpha blend ratio. For ARGB 4:4:4:4, bits 7-4 of this register are used as the 4-bit alpha blend ratio.

REG[09A2h] Camera I2C Data Register Default = 0Xh							Read Only
n/a						I2C SDA Pin Status	I2C SCL Pin Status
7	6	5	4	3	2	1	0

bit 1 I2C SDA Pin Status (Read Only)
This bit indicates the input status of the SDA pin used for the I2C interface.
When this bit = 0b, the SDA pin is 0 (low).
When this bit = 1b, the SDA pin is 1 (high).

bit 0 I2C SCL Pin Status (Read Only)
This bit indicates the input status of the SCL pin used for the I2C interface.
When this bit = 0b, the SCL pin is 0 (low).
When this bit = 1b, the SCL pin is 1 (high).

REG[09A3h] Camera I2C Output Enable Register Default = 03h							Read/Write
n/a						I2C SDA Output Enable	I2C SCL Output Enable
7	6	5	4	3	2	1	0

bit 1 I2C SDA Output Enable
This bit controls SDA pin output for the I2C interface.
When this bit = 0b, the I2C SDA pin is enabled and driven low.
When this bit = 1b, the I2C SDA pin is disabled, tri-stated (high-impedance), and pulled high.

bit 0 I2C SCL Output Enable
This bit controls SCL pin output for the I2C interface.
When this bit = 0b, the I2C SCL pin is enabled and driven low.
When this bit = 1b, the I2C SCL pin is disabled, tri-stated (high-impedance), and pulled high.

REG[09A4h] OSD Transparency Color Blue Register Default = 00h							Read/Write
OSD Transparency Color Blue bits 7-0							
7	6	5	4	3	2	1	0

REG[09A5h] OSD Transparency Color Green Register Default = 00h							Read/Write
OSD Transparency Color Green bits 7-0							
7	6	5	4	3	2	1	0

REG[09A6h] OSD Transparency Color Red Register Default = 00h							Read/Write
OSD Transparency Color Red bits 7-0							
7	6	5	4	3	2	1	0

REG[09A6h] bits 7-0 OSD Transparency Color Red bits [7:0]
REG[09A5h] bits 7-0 OSD Transparency Color Green bits [7:0]
REG[09A4h] bits 7-0 OSD Transparency Color Blue bits [7:0]

These bits only have an effect when OSD Transparency is enabled, REG[09A7h] bit 7 = 1b. These bits specify the RGB components of the transparency color for the OSD window which are compared with the OSD window pixels to determine whether the OSD window pixel or the “background” pixel is displayed.

If the pixel format is RGB 8:8:8 (see REG[09A0h] bit 3 and REG[0980h] bits 3-2).
REG[09A6h] bits 7-0 = RED
REG[09A5h] bits 7-0 = GREEN
REG[09A4h] bits 7-0 = BLUE
If the pixel format is RGB 5:6:5 (see REG[09A0h] bit 3 and REG[0980h] bits 3-2).
REG[09A6h] bits 7-3 = RED
REG[09A5h] bits 7-2 = GREEN
REG[09A4h] bits 7-3 = BLUE
If the pixel format is RGB 3:3:2 (see REG[09A0h] bit 3 and REG[0980h] bits 3-2).
REG[09A6h] bits 7-5 = RED
REG[09A5h] bits 7-5 = GREEN
REG[09A4h] bits 7-6 = BLUE
If the pixel format is ARGB 8:5:6:5 (see REG[09A0h] bit 3 and REG[0980h] bits 3-2).
REG[09A6h] bits 7-3 = RED
REG[09A5h] bits 7-2 = GREEN
REG[09A4h] bits 7-3 = BLUE
If the pixel format is ARGB 1:5:5:5 (see REG[09A0h] bit 3 and REG[0980h] bits 3-2).
REG[09A6h] bits 7-3 = RED
REG[09A5h] bits 7-3 = GREEN
REG[09A4h] bits 7-3 = BLUE
If the pixel format is ARGB 4:4:4:4 (see REG[09A0h] bit 3 and REG[0980h] bits 3-2).
REG[09A6h] bits 7-4 = RED
REG[09A5h] bits 7-4 = GREEN
REG[09A4h] bits 7-4 = BLUE

Note

If OSD Alpha Format is enabled (REG[09A0h] bit 3 = 1b) and OSD Transparency is enabled (REG[09A7h] bit 7 = 1b, only the RGB components of the pixel value are compared.

REG[09A7h] OSD Transparency Enable Register Default = 00h Read/Write							
OSD Transparency Enable	n/a						
7	6	5	4	3	2	1	0

bit 7 OSD Transparency Enable
This bit controls the transparency function for the OSD window. The transparency color is specified by the OSD Transparency Color registers, REG[09A4h] ~ REG[09A6h].
When this bit = 0b, OSD transparency is disabled.
When this bit = 1b, OSD transparency is enabled.

Note

If OSD Alpha Format is enabled (REG[09A0h] bit 3 = 1b) and OSD Transparency is enabled, only the RGB components of the pixel value are compared.

10.4.12 Image Fetcher Configuration Registers

REG[09AAh] Image Fetcher Input X Offset Register 0 Default = 00h							Read/Write
Image Fetcher Input X Offset bits 7-0							
7	6	5	4	3	2	1	0

REG[09ABh] Image Fetcher Input X Offset Register 1 Default = 00h							Read/Write
n/a			Image Fetcher Input X Offset bits 12-8				
7	6	5	4	3	2	1	0

REG[09ABh] bits 4-0
REG[09AAh] bits 7-0 Image Fetcher Input X Offset bits [12:0]
These bits specify the X offset of the top left corner of the Image Fetcher window relative to the top left corner of the Image Fetcher window virtual image, in pixels. For an example showing a virtual source window, see Figure 13-9: "Virtual Source Window Example," on page 464.

REG[09ACh] Image Fetcher Input Y Offset Register 0 Default = 00h							Read/Write
Image Fetcher Input Y Offset bits 7-0							
7	6	5	4	3	2	1	0

REG[09ADh] Image Fetcher Input Y Offset Register 1 Default = 00h							Read/Write
n/a			Image Fetcher Input Y Offset bits 12-8				
7	6	5	4	3	2	1	0

REG[09ADh] bits 4-0
REG[09ACh] bits 7-0 Image Fetcher Input Y Offset bits [12:0]
These bits specify the Y offset of the top left corner of the Image Fetcher window relative to the top left corner of the Image Fetcher window virtual image, in pixels. For an example showing a virtual source window, see Figure 13-9: "Virtual Source Window Example," on page 464.

REG[09B0h] Image Fetcher Control Register Default = 00h							Read/Write
Image Fetcher Line Double Enable	Image Fetcher Horizontal Flip	Image Fetcher Vertical Flip	Image Fetcher Enable	n/a		Image Fetcher Mode	Image Fetcher Blank
7	6	5	4	3	2	1	0

bit 7 Image Fetcher Line Double Enable
This bit controls "line double" mode which is typically used for displaying interlaced images from the camera interface. When line doubling is enabled, each line of the input image stored in the SDRAM is read twice.
When this bit = 0b, Image Fetcher line doubling is disabled.
When this bit = 1b, Image Fetcher line doubling is enabled.

bit 6 Image Fetcher Horizontal Flip
This bit determines whether the image data input from the Image Fetcher is flipped around the Y axis (horizontal). This bit must be set to 0b when the Image Fetcher uses “tiled-frame” mode, REG[09B0h] bit 1 = 1b.
When this bit = 0b, the Image Fetcher image data is not horizontally flipped (disabled).
When this bit = 1b, the Image Fetcher image data is horizontally flipped (enabled).

bit 5 Image Fetcher Vertical Flip
This bit determines whether the image data input from the Image Fetcher is flipped around the X axis (vertical). This bit must be set to 0b when the Image Fetcher uses “tiled-frame” mode, REG[09B0h] bit 1 = 1b.
When this bit = 0b, the Image Fetcher image data is not vertically flipped (disabled).
When this bit = 1b, the Image Fetcher image data is vertically flipped (enabled).

bit 4 Image Fetcher Enable
This bit controls whether the Image Fetcher image data is displayed (enabled) or not (disabled).
When this bit = 0b, the Image Fetcher is disabled.
When this bit = 1b, the Image Fetcher is enabled.

Note

If hardware frame control is selected for the Image Fetcher (REG[09DBh] bit 0 = 1b), it must be disabled before the Image Fetcher can be disabled. The following sequence is recommended.

1. Disable hardware frame control, REG[09DBh] bit 0 = 0b.
2. Wait 1 frame.
3. Disable the Image Fetcher, REG[09B0h] bit 4 = 0b.

bit 1 Image Fetcher Mode
This bit specifies how the Image Fetcher image data is stored in memory. For details on the memory organization methods, see Section 13.3, “Memory Organization of Frames” on page 470.
When this bit = 0b, the Image Fetcher uses “line-by-line” mode to read from SDRAM.
When this bit = 1b, the Image Fetcher uses “tiled-frame” mode to read from SDRAM.

Note

For tiled frame mode, the image width must be a multiple of 8 pixels and the Image Fetcher image data must not be flipped (REG[09B0h] bit 6 = 0b and bit 5 = 0b).

bit 0 Image Fetcher Blank
This bit controls the Image Fetcher blank function. The blank function replaces the image data from the Image Fetcher with the color specified by the Image Fetcher Blank Color registers, REG[09B4h] ~ REG[09B6h].
When this bit = 0b, the Image Fetcher image data is read normally (not blanked).
When this bit = 1b, the Image Fetcher image data is “blanked” with the specified color.

REG[09B2h] Image Fetcher Frame Control/Status Register Default = 00h							Read/Write
n/a		Image Fetcher Frame Buffer 1 Ready Clear (WO)	Image Fetcher Frame Buffer 0 Ready Clear (WO)	n/a	Image Fetcher Current Frame Status (RO)	Image Fetcher Frame Buffer 1 Ready	Image Fetcher Frame Buffer 0 Ready
7	6	5	4	3	2	1	0

bit 5 Image Fetcher Frame Buffer 1 Ready Clear (Write Only)
This bit is used to manually clear the Image Fetcher Frame Buffer 1 Ready bit, REG[09B2h] bit 1.
Writing a 0b to this bit has no effect.
Writing a 1b to this bit clears the Image Fetcher Frame Buffer 1 Ready bit.

bit 4 Image Fetcher Frame Buffer 0 Ready Clear (Write Only)
This bit is used to manually clear the Image Fetcher Frame Buffer 0 Ready bit, REG[09B2h] bit 0.
Writing a 0b to this bit has no effect.
Writing a 1b to this bit clears the Image Fetcher Frame Buffer 0 Ready bit.

bit 2 Image Fetcher Current Frame Status (Read Only)
This bit indicates which Image Fetcher frame buffer is currently being read.
When this bit = 0b, Image Fetcher Frame Buffer 0 is being read.
When this bit = 1b, Image Fetcher Frame Buffer 1 is being read.

Note

When the Image Fetcher is disabled and then re-enabled using the Image Fetcher Enable bit (REG[09B0h] bit 4), the hardware always sets the Current Frame status to 0b and checks the Image Fetcher Frame Buffer 0 Ready bit first. Therefore before re-enabling the Image Fetcher, the Image Fetcher image stream must be reset to start with Buffer 0, the Image Fetcher Frame Buffer 0/1 Ready bits must be cleared (see REG[09B2h] bits 5-4), and the Image Fetcher Frame Buffer 0 Ready bit must be set to 1b (REG[09B2h] bit 0 = 1b).

bit 1 Image Fetcher Frame Buffer 1 Ready
This bit only has an effect when Image Fetcher double-buffering is configured for software control, REG[09DBh] bit 0 = 0b.
For Writes:
Writing a 0b to this bit has no effect.
Writing a 1b to this bit sets this bit to 1b and indicates that the Image Frame Buffer 1 image data is ready for reading. Once this bit is set to 1b, it remains at 1b until it is reset by the Image Fetcher when it switches reading from frame buffer 1 to frame buffer 0.
For Reads:
When this bit = 0b, Image Fetcher Frame Buffer 1 does not contain valid image data.
When this bit = 1b, Image Fetcher Frame Buffer 1 contains valid image data.

bit 0 Image Fetcher Frame Buffer 0 Ready
This bit only has an effect when Image Fetcher double-buffering is configured for software control, REG[09DBh] bit 0 = 0b.
For Writes:
Writing a 0b to this bit has no effect.
Writing a 1b to this bit sets this bit to 1b and indicates that the Image Frame Buffer 0 image data is ready for reading. Once this bit is set to 1b, it remains at 1b until it is reset by the Image Fetcher when it switches reading from frame buffer 0 to frame buffer 1.
For Reads:
When this bit = 0b, Image Fetcher Frame Buffer 0 does not contain valid image data.
When this bit = 1b, Image Fetcher Frame Buffer 0 contains valid image data.

REG[09B4h] Image Fetcher Blank Color Blue Register Default = 00h							Read/Write
Image Fetcher Blank Color Blue bits 7-0							
7	6	5	4	3	2	1	0

REG[09B5h] Image Fetcher Blank Color Green Register Default = 00h							Read/Write
Image Fetcher Blank Color Green bits 7-0							
7	6	5	4	3	2	1	0

REG[09B6h] Image Fetcher Blank Color Red Register Default = 00h							Read/Write
Image Fetcher Blank Color Red bits 7-0							
7	6	5	4	3	2	1	0

REG[09B6h] bits 7-0 Image Fetcher Blank Color Red bits [7:0]
REG[09B5h] bits 7-0 Image Fetcher Blank Color Green bits [7:0]
REG[09B4h] bits 7-0 Image Fetcher Blank Color Blue bits [7:0]
When the Image Fetcher Blank bit is set (REG[09B0h] bit 0 = 1b), these bits specify the RGB components of the color that the Image Fetcher replaces image data with. Note that the Image Fetcher pixel format is determined by the CH1IN pixel format, REG[4062h] bits 2-0.

If the Image Fetcher Pixel Format is RGB 8:8:8 (REG[4062h] bits 2-0 = 010b).
REG[09B6h] bits 7-0 = RED
REG[09B5h] bits 7-0 = GREEN
REG[09B4h] bits 7-0 = BLUE

If the Image Fetcher Pixel Format is RGB 5:6:5 (REG[4062h] bits 2-0 = 001b).
REG[09B6h] bits 7-3 = RED
REG[09B5h] bits 7-2 = GREEN
REG[09B4h] bits 7-3 = BLUE

If the Image Fetcher Pixel Format is RGB 3:3:2 (REG[4062h] bits 2-0 = 000b).
REG[09B6h] bits 7-5 = RED
REG[09B5h] bits 7-5 = GREEN
REG[09B4h] bits 7-6 = BLUE

REG[09B8h] Image Fetcher Frame Buffer 0 Address Register 0 Default = 00h Read/Write							
Image Fetcher Frame Buffer 0 Address bits 7-0							
7	6	5	4	3	2	1	0

REG[09B9h] Image Fetcher Frame Buffer 0 Address Register 1 Default = 00h Read/Write							
Image Fetcher Frame Buffer 0 Address bits 15-8							
7	6	5	4	3	2	1	0

REG[09BAh] Image Fetcher Frame Buffer 0 Address Register 2 Default = 00h Read/Write							
Image Fetcher Frame Buffer 0 Address bits 23-16							
7	6	5	4	3	2	1	0

REG[09BBh] Image Fetcher Frame Buffer 0 Address Register 3 Default = 10h Read/Write							
Image Fetcher Frame Buffer 0 Address bits 31-24							
7	6	5	4	3	2	1	0

REG[09BBh] bits 7-0
REG[09BAh] bits 7-0
REG[09B9h] bits 7-0
REG[09B8h] bits 7-0 Image Fetcher Frame Buffer 0 Address bits [31:0]
These bits specify the memory start address for Image Fetcher Frame Buffer 0. These bits must be set such that the start address is 8 byte (64-bit) aligned.

REG[09BCh] Image Fetcher Frame Buffer 1 Address Register 0 Default = 00h							Read/Write
Image Fetcher Frame Buffer 1 Address bits 7-0							
7	6	5	4	3	2	1	0

REG[09BDh] Image Fetcher Frame Buffer 1 Address Register 1 Default = 00h							Read/Write
Image Fetcher Frame Buffer 1 Address bits 15-8							
7	6	5	4	3	2	1	0

REG[09BEh] Image Fetcher Frame Buffer 1 Address Register 2 Default = 00h							Read/Write
Image Fetcher Frame Buffer 1 Address bits 23-16							
7	6	5	4	3	2	1	0

REG[09BFh] Image Fetcher Frame Buffer 1 Address Register 3 Default = 10h							Read/Write
Image Fetcher Frame Buffer 1 Address bits 31-24							
7	6	5	4	3	2	1	0

REG[09BFh] bits 7-0
REG[09BEh] bits 7-0
REG[09BDh] bits 7-0
REG[09BCh] bits 7-0 Image Fetcher Frame Buffer 1 Address bits [31:0]

These bits specify the memory start address for Image Fetcher Frame Buffer 1. These bits must be set such that the start address is 8 byte (64-bit) aligned.

REG[09C0h] Image Fetcher Width Register 0 Default = 40h							Read/Write
Image Fetcher Width bits 7-0							
7	6	5	4	3	2	1	0

REG[09C1h] Image Fetcher Width Register 1 Default = 01h							Read/Write
n/a					Image Fetcher Width bits 10-8		
7	6	5	4	3	2	1	0

REG[09C1h] bits 2-0
REG[09C0h] bits 7-0 Image Fetcher Width bits [10:0]
These bits specify the width of the Image Fetcher image, in pixels.

REG[09C2h] Image Fetcher Height Register 0 Default = F0h							Read/Write
Image Fetcher Height bits 7-0							
7	6	5	4	3	2	1	0

REG[09C3h] Image Fetcher Height Register 1 Default = 00h							Read/Write
n/a					Image Fetcher Height bits 10-8		
7	6	5	4	3	2	1	0

REG[09C3h] bits 2-0
REG[09C2h] bits 7-0 Image Fetcher Height bits [10:0]
These bits specify the height of the Image Fetcher image, in pixels.

REG[09C4h] Image Fetcher Virtual Width Register 0 Default = 40h							Read/Write
Image Fetcher Virtual Width bits 7-0							
7	6	5	4	3	2	1	0

REG[09C5h] Image Fetcher Virtual Width Register 1 Default = 01h							Read/Write
n/a			Image Fetcher Virtual Width bits 12-8				
7	6	5	4	3	2	1	0

REG[09C5h] bits 4-0
REG[09C4h] bits 7-0 Image Fetcher Virtual Width bits [12:0]
These bits specify the width of the Image Fetcher virtual image, in pixels. For an example showing a virtual source window, see Figure 13-9: “Virtual Source Window Example,” on page 464.

Note
The Image Fetcher virtual width must be set such that the virtual width multiplied by the pixel format (in bpp, see REG[4062h] bits 2-0) is divisible by 64.

10.4.13 LCD Configuration Registers

REG[09C8h] LCD Control A Register Default = F0h							Read/Write
Camera2 Frame Write Idle (RO)	Camera1 Frame Write Idle (RO)	CH1OUT Writeback Frame Write Idle (RO)	Warp Writeback Frame Write Idle (RO)	OSDIN Source Select	CH2IN Source Select	CH1IN Source Select bits 1-0	
7	6	5	4	3	2	1	0

bit 7 Camera2 Frame Write Idle (Read Only)
This bit indicates whether the Camera2 Writer is writing a frame to SDRAM.
When this bit = 0b, the Camera2 Writer is busy writing a frame to SDRAM.
When this bit = 1b, the Camera2 Writer is idle. (default)

bit 6 Camera1 Frame Write Idle (Read Only)
This bit indicates whether the Camera1 Writer is writing a frame to SDRAM.
When this bit = 0b, the Camera1 Writer is busy writing a frame to SDRAM.
When this bit = 1b, the Camera1 Writer is idle. (default)

bit 5 CH1OUT Writeback Frame Write Idle (Read Only)
This bit indicates whether CH1OUT Writeback is writing a frame to SDRAM. For further information on CH1OUT Writeback, see Section 13.2.4, “CH1OUT Writeback” on page 467.
When this bit = 0b, CH1OUT Writeback is busy writing a frame to SDRAM.
When this bit = 1b, CH1OUT Writeback is idle. (default)

bit 4 Warp Writeback Frame Write Idle (Read Only)
This bit indicates whether Warp Writeback is writing a frame to SDRAM. For further information on Warp Writeback, see Section 13.2.5, “Warp Writeback” on page 468.
When this bit = 0b, Warp Writeback is busy writing a frame to SDRAM.
When this bit = 1b, Warp Writeback is idle. (default)

bit 3 OSDIN Source Select
This bit selects the Blending Engine output source used for the LCD controller input OSDIN.
When this bit = 0b, OSDOUT is the OSDIN source.
When this bit = 1b, CH1OUT is the OSDIN source (see Note).

Note

Only one of the LCD controller input channels can have CH1OUT as the source. For a summary of the possible settings, see Table 10-39: “CH1/CH2/OSD Input Source Selection,” on page 280.

bit 2 — CH2IN Source Select
This bit selects the Blending Engine output source used for the LCD controller input CH2IN.
When this bit = 0b, CH2OUT is the CH2IN source.
When this bit = 1b, CH1OUT is the CH2IN source (see Note).

Note
Only one of the LCD controller input channels can have CH1OUT as the source. For a summary of the possible settings, see Table 10-39: “CH1/CH2/OSD Input Source Selection,” on page 280.

bits 1-0 — CH1IN Source Select bits [1:0]
These bits select the output source used for the LCD controller input CH1IN.

Note
Only one of the LCD controller input channels can have CH1OUT as the source. For a summary of the possible settings, see Table 10-39: “CH1/CH2/OSD Input Source Selection,” on page 280.

Table 10-39: CH1/CH2/OSD Input Source Selection

<table>
<tr><th>REG[09C8h] bits 1-0</th><th>REG[09C8h] bit 2</th><th>REG[09C8h] bit 3</th><th>CH1IN Source</th><th>CH2IN Source</th><th>OSDIN Source</th></tr>
<tr><td rowspan="4">00b</td><td rowspan="2">0b</td><td>0b</td><td>CH1OUT</td><td>CH2OUT</td><td>OSDOUT</td></tr>
<tr><td>1b</td><td colspan="3">Reserved</td></tr>
<tr><td rowspan="2">1b</td><td>0b</td><td colspan="3">Reserved</td></tr>
<tr><td>1b</td><td colspan="3">Reserved</td></tr>
<tr><td rowspan="4">01b</td><td rowspan="2">0b</td><td>0b</td><td rowspan="3">Warp</td><td rowspan="2">CH2OUT</td><td>OSDOUT</td></tr>
<tr><td>1b</td><td>CH1OUT</td></tr>
<tr><td rowspan="2">1b</td><td>0b</td><td>CH1OUT</td><td>OSDOUT</td></tr>
<tr><td>1b</td><td colspan="3">Reserved</td></tr>
<tr><td rowspan="4">10b</td><td rowspan="2">0b</td><td>0b</td><td rowspan="3">Image Fetcher</td><td rowspan="2">CH2OUT</td><td>OSDOUT</td></tr>
<tr><td>1b</td><td>CH1OUT</td></tr>
<tr><td rowspan="2">1b</td><td>0b</td><td>CH1OUT</td><td>OSDOUT</td></tr>
<tr><td>1b</td><td colspan="3">Reserved</td></tr>
<tr><td>11b</td><td>Xb</td><td>Xb</td><td colspan="3">Reserved</td></tr>
</table>

REG[09CAh] LCD Control B Register Default = 00h Read/Write							
Warp Writeback Mode	Reserved	Warp Writeback Vertical Flip	Warp Writeback Manual Trigger (WO)	CH1OUT Writeback Manual Trigger (WO)	n/a		
7	6	5	4	3	2	1	0

bit 7 Warp Writeback Mode
This bit only has an effect when Warp output is written back to the SDRAM, REG[09CAh] bit 6 = 1b. The bit specifies how Warp Writeback image data is stored in memory. For details on the memory organization methods, see Section 13.3, "Memory Organization of Frames" on page 470.
When this bit = 0b, Warp Writeback uses "line-by-line" mode to write to SDRAM.
When this bit = 1b, Warp Writeback uses "tiled-frame" mode to write to SDRAM.

Note
For tiled frame mode, the image width must be a multiple of 8 pixels and the Warp Writeback image data must not be flipped (REG[09CA0h] bit 5 = 0b).

bit 6 Reserved
This bit MUST be set to 1b when the HUD/Warp engine is used.

bit 5 Warp Writeback Vertical Flip
This bit only has an effect when Warp output is written back to the SDRAM, REG[09CAh] bit 6 = 1b. This bit determines whether image data output from the Warp Logic is flipped around the X axis (vertical).This bit must be set to 0b when Warp Writeback uses "tiled-frame" mode, REG[09CAh] bit 7 = 1b.
When this bit = 0b, the Warp image data is not vertically flipped (disabled).
When this bit = 1b, the Warp image data is vertically flipped (enabled).

bit 4 Warp Writeback Manual Trigger (Write Only)
This bit is a manual trigger which forces the Warp Writeback logic to process another frame and store it in the SDRAM.
Writing a 0b to this bit has no effect.
Writing a 1b to this bit forces the Warp Writeback logic to process another frame.

bit 3 CH1OUT Writeback Manual Trigger (Write Only)
This bit is a manual trigger which forces the CH1OUT Writeback logic to process another frame and store it in the SDRAM.
Writing a 0b to this bit has no effect.
Writing a 1b to this bit forces the CH1OUT Writeback logic to process another frame.

Note
Manually triggering CH1OUT Writeback to process another frame does not cause the MAIN frame buffer to switch.

REG[09D0h] Warp Writeback Frame Buffer 0 Address Register 0 Default = 00h							Read/Write
Warp Writeback Frame Buffer 0 Address bits 7-0							
7	6	5	4	3	2	1	0

REG[09D1h] Warp Writeback Frame Buffer 0 Address Register 1 Default = 00h							Read/Write
Warp Writeback Frame Buffer 0 Address bits 15-8							
7	6	5	4	3	2	1	0

REG[09D2h] Warp Writeback Frame Buffer 0 Address Register 2 Default = 00h							Read/Write
Warp Writeback Frame Buffer 0 Address bits 23-16							
7	6	5	4	3	2	1	0

REG[09D3h] Warp Writeback Frame Buffer 0 Address Register 3 Default = 10h							Read/Write
Warp Writeback Frame Buffer 0 Address bits 31-24							
7	6	5	4	3	2	1	0

REG[09D3h] bits 7-0
REG[09D2h] bits 7-0
REG[09D1h] bits 7-0
REG[09D0h] bits 7-0 Warp Writeback Frame Buffer 0 Address bits [31:0]
These bits specify the memory start address for Warp Writeback Frame Buffer 0 which is used for writing image data processed by the Warp Logic back to the SDRAM. These bits must be set such that the start address is 8 byte (64-bit) aligned.

REG[09D4h] Warp Writeback Frame Buffer 1 Address Register 0 Default = 00h Read/Write							
Warp Writeback Frame Buffer 1 Address bits 7-0							
7	6	5	4	3	2	1	0

REG[09D5h] Warp Writeback Frame Buffer 1 Address Register 1 Default = 00h Read/Write							
Warp Writeback Frame Buffer 1 Address bits 15-8							
7	6	5	4	3	2	1	0

REG[09D6h] Warp Writeback Frame Buffer 1 Address Register 2 Default = 00h Read/Write							
Warp Writeback Frame Buffer 1 Address bits 23-16							
7	6	5	4	3	2	1	0

REG[09D7h] Warp Writeback Frame Buffer 1 Address Register 3 Default = 10h Read/Write							
Warp Writeback Frame Buffer 1 Address bits 31-24							
7	6	5	4	3	2	1	0

REG[09D7h] bits 7-0
REG[09D6h] bits 7-0
REG[09D5h] bits 7-0
REG[09D4h] bits 7-0 Warp Writeback Frame Buffer 1 Address bits [31:0]

These bits specify the memory start address for Warp Writeback Frame Buffer 1 which is used for writing image data processed by the Warp Logic back to the SDRAM. These bits must be set such that the start address is 8 byte (64-bit) aligned.

REG[09D8h] LCD Frame Control A Register 0 Default = 00h							Read/Write
n/a	MAIN Window Hardware Frame Control Source bits 2-0			n/a			MAIN Window HW/SW Frame Control
7	6	5	4	3	2	1	0

bits 6-4 MAIN Window Hardware Frame Control Source bits [2:0]
When hardware frame control is selected for the MAIN window (REG[09D8h] bit 0 = 1b), these bits determine the control source (or producer) that will set the MAIN Window Frame Control status bits in REG[0942h].

Table 10-40: MAIN Window Hardware Frame Source Selection

REG[09D8h] bits 6-4	Frame Source
000b	Camera1
001b	Camera2
010b	Reserved
011b	Warp Writeback
100b ~ 111b	Sprite Engine

bit 0 MAIN Window HW/SW Frame Control
This bit determines whether MAIN window double-buffering frame control is done by hardware or software. When hardware frame control is selected, the control source (or producer), as selected by REG[09D8h] bits 6-4, directly sets the Frame Control status bits in REG[0942h]. When software frame control is selected, software must set the Frame Control status bits. For further information on frame control and double buffering, see Section 13.4, “Frame Double-Buffering Scheme” on page 472.
When this bit = 0b, software frame control is selected.
When this bit = 1b, hardware frame control is selected.

Note

1. When Camera1 or Camera2 uses double buffer method 1 (REG[09F6h]/[09FEh] bit 7 = 1b) and Camera1 or Camera2 is selected as the MAIN window frame source, the setting of this bit is ignored and hardware frame control is used.
2. If the frame source for the MAIN window is double buffered (see REG[09DCh]), the frame source double buffering must be disabled before the MAIN window frame control setting is changed. The frame source double buffering may be re-enabled once the setting is changed.

REG[09D9h] LCD Frame Control A Register 1 Default = 00h							Read/Write
n/a	AUX Window Hardware Frame Control Source bits 2-0			n/a			AUX Window HW/SW Frame Control
7	6	5	4	3	2	1	0

bits 6-4 AUX Window Hardware Frame Control Source bits [2:0]
When hardware frame control is selected for the AUX window (REG[09D9h] bit 0 = 1b), these bits determine the control source (or producer) that will set the AUX Window Frame Control status bits in REG[0962h].

Table 10-41: AUX Window Hardware Frame Source Selection

REG[09D9h] bits 6-4	Frame Source
000b	Camera1
001b	Camera2
010b	CH1OUT writeback
011b	Warp writeback
100b - 111b	Sprite Engine

bit 0 AUX Window HW/SW Frame Control
This bit determines whether AUX window double-buffering frame control is done by hardware or software. When hardware frame control is selected, the control source (or producer), as selected by REG[09D9h] bits 6-4, directly sets the Frame Control status bits in REG[0962h]. When software frame control is selected, software must set the Frame Control status bits. For further information on frame control and double buffering, see Section 13.4, "Frame Double-Buffering Scheme" on page 472.
When this bit = 0b, software frame control is selected.
When this bit = 1b, hardware frame control is selected.

Note

1. When Camera1 or Camera2 uses double buffer method 1 (REG[09F6h]/[09FEh] bit 7 = 1b) and Camera1 or Camera2 is selected as the AUX window frame source, the setting of this bit is ignored and hardware frame control is used.
2. If the frame source for the AUX window is double buffered (see REG[09DCh]), the frame source double buffering must be disabled before the AUX window frame control setting is changed. The frame source double buffering may be re-enabled once the setting is changed.
3. Hardware Frame Control is only supported for Blend Modes 1, 2, and 3 (see REG[09A0h] bits 1-0) and CH1OUT writeback (REG[09D9h] bits 6-4 = 010b).

REG[09DAh] LCD Frame Control B Register 0 Default = 00h							Read/Write
n/a	OSD Window Hardware Frame Control Source bits 2-0			n/a			OSD Window HW/SW Frame Control
7	6	5	4	3	2	1	0

bits 6-4 OSD Window Hardware Frame Control Source bits [2:0]
When hardware frame control is selected for the OSD window (REG[09DAh] bit 0 = 1b), these bits determine the control source (or producer) that will set the OSD Window Frame Control status bits in REG[0982h].

Table 10-42: OSD Window Hardware Frame Source Selection

REG[09DAh] bits 6-4	Frame Source
000b	Camera1
001b	Camera2
010b	CH1OUT writeback
011b	Warp writeback
100b	Sprite Engine

bit 0 OSD Window HW/SW Frame Control
This bit determines whether OSD window double-buffering frame control is done by hardware or software. When hardware frame control is selected, the control source (or producer), as selected by REG[09DAh] bits 6-4, directly sets the Frame Control status bits in REG[0982h]. When software frame control is selected, software must set the Frame Control status bits. For further information on frame control and double buffering, see Section 13.4, “Frame Double-Buffering Scheme” on page 472.
When this bit = 0b, software frame control is selected.
When this bit = 1b, hardware frame control is selected.

Note

1. When Camera1 or Camera2 uses double buffer method 1 (REG[09F6h]/[09FEh] bit 7 = 1b) and Camera1 or Camera2 is selected as the OSD window frame source, the setting of this bit is ignored and hardware frame control is used.
2. If the frame source for the OSD window is double buffered (see REG[09DCh]), the frame source double buffering must be disabled before the OSD window frame control setting is changed. The frame source double buffering may be re-enabled once the setting is changed.
3. Hardware Frame Control is only supported for Blend Modes 2 and 3 (see REG[09A0h] bits 1-0) and CH1OUT writeback (REG[09DAh] bits 6-4 = 010b).

REG[09DBh] LCD Frame Control B Register 1 Default = 00h							Read/Write
n/a	Image Fetcher Hardware Frame Control Source bits 2-0			n/a			Image Fetcher HW/SW Frame Control
7	6	5	4	3	2	1	0

bits 6-4 Image Fetcher Hardware Frame Control Source bits [2:0]
When hardware frame control is selected for the Image Fetcher (REG[09DBh] bit 0 = 1b), these bits determine the control source (or producer) that will set the Image Fetcher Frame Control status bits in REG[09B2h].

Table 10-43: Image Fetcher Hardware Frame Source Selection

REG[09DBh] bits 6-4	Frame Source
000b	Camera1
001b	Reserved
010b	CH1OUT writeback
011b	Warp writeback
100b	Sprite Engine

bit 0 Image Fetcher HW/SW Frame Control
This bit determines whether Image Fetcher double-buffering frame control is done by hardware or software. When hardware frame control is selected, the control source (or producer), as selected by REG[09DBh] bits 6-4, directly sets the Frame Control status bits in REG[09B2h]. When software frame control is selected, software must set the Frame Control status bits. For further information on frame control and double buffering, see Section 13.4, “Frame Double-Buffering Scheme” on page 472.
When this bit = 0b, software frame control is selected.
When this bit = 1b, hardware frame control is selected.

Note

1. When Camera1 or Camera2 uses double buffer method 1 (REG[09F6h]/[09FEh] bit 7 = 1b) and Camera1 or Camera2 is selected as the Image Fetcher frame source, the setting of this bit is ignored and hardware frame control is used.
2. If the frame source for the Image Fetcher is double buffered (see REG[09DCh]), the frame source double buffering must be disabled before the Image Fetcher frame control setting is changed. The frame source double buffering may be re-enabled once the setting is changed.

REG[09DCh] LCD Frame Control C Register 0 Default = 00h Read/Write							
n/a	Warp Logic Hardware Frame Control Source bits 2-0			Camera2 Frame Double-Buffer Disable	Camera1 Frame Double-Buffer Disable	CH1OUT Writeback Frame Double-Buffer Disable	Warp Writeback Frame Double-Buffer Disable
7	6	5	4	3	2	1	0

bits 6-4 Warp Logic Hardware Frame Control Source bits [2:0]
When hardware frame control is selected for the Warp Logic (REG[0400h] bit 6 = 1b), these bits determine the control source (or producer) that will set the Warp Frame Control status bits in REG[0408h] ~ REG[040Ah].

Table 10-44: Warp Logic Hardware Frame Source Selection

REG[09DCh] bits 6-4	Frame Source
000b	Camera1
001b	Camera2
010b	CH1OUT writeback
011b	Reserved
100b ~ 111b	Sprite Engine

bit 3 Camera2 Writeback Frame Double-Buffer Disable
This bit is used to disable hardware controlled frame double-buffering for Camera2 Writeback.
When this bit = 0b, hardware controlled frame double-buffering is enabled. (default)
When this bit = 1b, hardware controlled frame double-buffering is disabled and Camera2 Writeback only writes to buffer 0.

bit 2 Camera1 Writeback Frame Double-Buffer Disable
This bit used to disable hardware controlled frame double-buffering for Camera1 Writeback.
When this bit = 0b, hardware controlled frame double-buffering is enabled. (default)
When this bit = 1b, hardware controlled frame double-buffering is disabled and Camera1 Writeback only writes to buffer 0.

bit 1 CH1OUT Writeback Frame Double-Buffer Disable
This bit used to disable hardware controlled frame double-buffering for CH1OUT Writeback.
When this bit = 0b, hardware controlled frame double-buffering is enabled. (default)
When this bit = 1b, hardware controlled frame double-buffering is disabled and CH1OUT Writeback only writes to buffer 0.

bit 0 Warp Writeback Frame Double-Buffer Disable
This bit used to disable the hardware controlled frame double-buffering for Warp Writeback.
When this bit = 0b, hardware controlled frame double-buffering is enabled. (default)
When this bit = 1b, hardware controlled frame double-buffering is disabled and Warp Writeback only writes to buffer 0.

REG[09DDh] LCD Frame Control C Register 1 Default = 00h							Read/Write
n/a				Sprite Engine Hardware Frame Control Destination bits 3-0			
7	6	5	4	3	2	1	0

bits 3-0 Sprite Engine Hardware Frame Control Destination bits [3:0]
These bits select the destination (or consumer) for image data from the Sprite Engine. This allows the Sprite Engine (or producer) to receive frame control status information from the selected destination when Hardware Frame Control is enabled (see REG[09D8h] ~ REG[09DBh] bit 0 or REG[0400h] bit 6). For further information on frame control and double buffering, see Section 13.4, “Frame Double-Buffering Scheme” on page 472.

Table 10-45: Sprite Engine Hardware Frame Control Destination Selection

REG[09DDh] bits 3-0	Frame Control Destination
0000b	MAIN Window Hardware Frame Control
0010b	AUX Window Hardware Frame Control
0100b	OSD Window Hardware Frame Control
0110b	Image Fetcher Hardware Frame Control
Other values	Warp Hardware Frame Control

REG[09DEh] LCD Frame Control D Register 0 Default = 00h Read/Write							
Camera2 Hardware Frame Control Destination bits 3-0				Camera1 Hardware Frame Control Destination bits 3-0			
7	6	5	4	3	2	1	0

bits 7-4 Camera2 Hardware Frame Control Destination bits [3:0]
These bits select the destination (or consumer) for image data from the Camera2 interface. This allows the Camera2 interface (or producer) to receive frame control status information from the selected destination when Hardware Frame Control is enabled (see REG[09D8h] ~ REG[09DBh] bit 0 or REG[0400h] bit 6). For further information on frame control and double buffering, see Section 13.4, "Frame Double-Buffering Scheme" on page 472.

Table 10-46: Camera2 Hardware Frame Control Destination Selection

REG[09DEh] bits 7-4	Frame Control Destination
0000b	MAIN Window Hardware Frame Control
0010b	AUX Window Hardware Frame Control
0100b	OSD Window Hardware Frame Control
0110b	Image Fetcher Hardware Frame Control
Other values	Warp Hardware Frame Control

bits 3-0 Camera1 Hardware Frame Control Destination bits [3:0]
These bits select the destination (or consumer) for image data from the Camera1 interface. This allows the Camera1 interface (or producer) to receive frame control status information from the selected destination when Hardware Frame Control is enabled (see REG[09D8h] ~ REG[09DBh] bit 0 or REG[0400h] bit 6). For further information on frame control and double buffering, see Section 13.4, "Frame Double-Buffering Scheme" on page 472.

Table 10-47: Camera1 Hardware Frame Control Destination Selection

REG[09DEh] bits 3-0	Frame Control Destination
0000b	MAIN Window Hardware Frame Control
0010b	AUX Window Hardware Frame Control
0100b	OSD Window Hardware Frame Control
0110b	Image Fetcher Hardware Frame Control
Other values	Warp Hardware Frame Control

REG[09DFh] LCD Frame Control D Register 1 Default = 00h							Read/Write
Warp Writeback Hardware Frame Control Destination bits 3-0				CH1OUT Writeback Hardware Frame Control Destination bits 3-0			
7	6	5	4	3	2	1	0

bits 7-4 Warp Writeback Hardware Frame Control Destination bits [3:0]
These bits select the destination (or consumer) for image data from Warp Writeback. This allows the Warp Writeback (or producer) to receive frame control status information from the selected destination when Hardware Frame Control is enabled (see REG[09D8h] ~ REG[09DBh] bit 0 or REG[0400h] bit 6). For further information on frame control and double buffering, see Section 13.4, "Frame Double-Buffering Scheme" on page 472.

Table 10-48: Warp Writeback Hardware Frame Control Destination Selection

REG[09DFh] bits 7-4	Frame Control Destination
0000b	MAIN Window Hardware Frame Control
0010b	AUX Window Hardware Frame Control
0100b	OSD Window Hardware Frame Control
0110b	Image Fetcher Hardware Frame Control
Other values	Reserved

bits 3-0 CH1OUT Writeback Hardware Frame Control Destination bits [3:0]
These bits select the destination (or consumer) for image data from CH1OUT Writeback. This allows the CH1OUT Writeback (or producer) to receive frame control status information from the selected destination when Hardware Frame Control is enabled (see REG[09D8h] ~ REG[09DBh] bit 0 or REG[0400h] bit 6). For further information on frame control and double buffering, see Section 13.4, "Frame Double-Buffering Scheme" on page 472.

Table 10-49: CH1OUT Writeback Hardware Frame Control Destination Selection

REG[09DFh] bits 3-0	Frame Control Destination
0000b	MAIN Window Hardware Frame Control
0010b	AUX Window Hardware Frame Control
0100b	OSD Window Hardware Frame Control
0110b	Image Fetcher Hardware Frame Control
Other values	Warp Hardware Frame Control

REG[09E0h] Camera1 Frame Buffer 0 Address Register 0 Default = 00h							Read/Write
Camera1 Frame Buffer 0 Address bits 7-0							
7	6	5	4	3	2	1	0

REG[09E1h] Camera1 Frame Buffer 0 Address Register 1 Default = 00h							Read/Write
Camera1 Frame Buffer 0 Address bits 15-8							
7	6	5	4	3	2	1	0

REG[09E2h] Camera1 Frame Buffer 0 Address Register 2 Default = 00h							Read/Write
Camera1 Frame Buffer 0 Address bits 23-16							
7	6	5	4	3	2	1	0

REG[09E3h] Camera1 Frame Buffer 0 Address Register 3 Default = 10h							Read/Write
Camera1 Frame Buffer 0 Address bits 31-24							
7	6	5	4	3	2	1	0

REG[09E3h] bits 7-0
REG[09E2h] bits 7-0
REG[09E1h] bits 7-0
REG[09E0h] bits 7-0 Camera1 Frame Buffer 0 Address bits [31:0]

These bits specify the memory start address for Camera1 Frame Buffer 0 which is used for input image data from Camera1. These bits must be set such that the start address is 8 byte (64-bit) aligned.

REG[09E4h] Camera1 Frame Buffer 1 Address Register 0 Default = 00h Read/Write							
Camera1 Frame Buffer 1 Address bits 7-0							
7	6	5	4	3	2	1	0

REG[09E5h] Camera1 Frame Buffer 1 Address Register 1 Default = 00h Read/Write							
Camera1 Frame Buffer 1 Address bits 15-8							
7	6	5	4	3	2	1	0

REG[09E6h] Camera1 Frame Buffer 1 Address Register 2 Default = 00h Read/Write							
Camera1 Frame Buffer 1 Address bits 23-16							
7	6	5	4	3	2	1	0

REG[09E7h] Camera1 Frame Buffer 1 Address Register 3 Default = 10h Read/Write							
Camera1 Frame Buffer 1 Address bits 31-24							
7	6	5	4	3	2	1	0

REG[09E7h] bits 7-0
REG[09E6h] bits 7-0
REG[09E5h] bits 7-0
REG[09E4h] bits 7-0 Camera1 Frame Buffer 1 Address bits [31:0]

These bits specify the memory start address for Camera1 Frame Buffer 1 which is used for input image data from Camera1. These bits must be set such that the start address is 8 byte (64-bit) aligned.

REG[09E8h] Camera2 Frame Buffer 0 Address Register 0							
Default = 00h							Read/Write
Camera2 Frame Buffer 0 Address bits 7-0							
7	6	5	4	3	2	1	0

REG[09E9h] Camera2 Frame Buffer 0 Address Register 1							
Default = 00h							Read/Write
Camera2 Frame Buffer 0 Address bits 15-8							
7	6	5	4	3	2	1	0

REG[09EAh] Camera2 Frame Buffer 0 Address Register 2							
Default = 00h							Read/Write
Camera2 Frame Buffer 0 Address bits 23-16							
7	6	5	4	3	2	1	0

REG[09EBh] Camera2 Frame Buffer 0 Address Register 3							
Default = 10h							Read/Write
Camera2 Frame Buffer 0 Address bits 31-24							
7	6	5	4	3	2	1	0

REG[09EBh] bits 7-0
REG[09EAh] bits 7-0
REG[09E9h] bits 7-0
REG[09E8h] bits 7-0 Camera2 Frame Buffer 0 Address bits [31:0]

These bits specify the memory start address for Camera2 Frame Buffer 0 which is used for input image data from Camera2. These bits must be set such that the start address is 8 byte (64-bit) aligned.

REG[09ECh] Camera2 Frame Buffer 1 Address Register 0 Default = 00h Read/Write							
Camera2 Frame Buffer 1 Address bits 7-0							
7	6	5	4	3	2	1	0

REG[09EDh] Camera2 Frame Buffer 1 Address Register 1 Default = 00h Read/Write							
Camera2 Frame Buffer 1 Address bits 15-8							
7	6	5	4	3	2	1	0

REG[09EEh] Camera2 Frame Buffer 1 Address Register 2 Default = 00h Read/Write							
Camera2 Frame Buffer 1 Address bits 23-16							
7	6	5	4	3	2	1	0

REG[09EFh] Camera2 Frame Buffer 1 Address Register 3 Default = 10h Read/Write							
Camera2 Frame Buffer 1 Address bits 31-24							
7	6	5	4	3	2	1	0

REG[09EFh] bits 7-0
REG[09EEh] bits 7-0
REG[09EDh] bits 7-0
REG[09ECh] bits 7-0 Camera2 Frame Buffer 1 Address bits [31:0]

These bits specify the memory start address for Camera2 Frame Buffer 1 which is used for input image data from Camera2. These bits must be set such that the start address is 8 byte (64-bit) aligned.

REG[09F0h] Camera1 Frame Buffer Width Register 0							
Default = 40h							Read/Write
Camera1 Frame Buffer Width bits 7-0							
7	6	5	4	3	2	1	0

REG[09F1h] Camera1 Frame Buffer Width Register 1							
Default = 01h							Read/Write
n/a					Camera1 Frame Buffer Width bits 10-8		
7	6	5	4	3	2	1	0

REG[09F1h] bits 2-0
REG[09F0h] bits 7-0 Camera1 Frame Buffer Width bits [10:0]
These bits specify the width of the Camera1 frame buffer, in pixels.

Note
The Camera1 frame buffer width must be set such that the width multiplied by the pixel format (in bpp, see REG[09F6h] bits 3-2) is divisible by 64.

REG[09F2h] Camera1 Frame Buffer Height Register 0							
Default = F0h							Read/Write
Camera1 Frame Buffer Height bits 7-0							
7	6	5	4	3	2	1	0

REG[09F3h] Camera1 Frame Buffer Height Register 1							
Default = 00h							Read/Write
n/a					Camera1 Frame Buffer Height bits 10-8		
7	6	5	4	3	2	1	0

REG[09F3h] bits 2-0
REG[09F2h] bits 7-0 Camera1 Frame Buffer Height bits [10:0]
These bits specify the height of the Camera1 frame buffer, in pixels.

REG[09F4h] Camera1 Frame Buffer Virtual Width Register 0							
Default = 40h							Read/Write
Camera1 Frame Buffer Virtual Width bits 7-0							
7	6	5	4	3	2	1	0

REG[09F5h] Camera1 Frame Buffer Virtual Width Register 1							
Default = 01h							Read/Write
n/a			Camera1 Frame Buffer Virtual Width bits 12-8				
7	6	5	4	3	2	1	0

REG[09F5h] bits 4-0
REG[09F4h] bits 7-0 Camera1 Frame Buffer Virtual Width bits [12:0]
These bits specify the virtual width of the Camera1 frame buffer, in pixels.

Note
The Camera1 frame buffer virtual width must be set such that the virtual width multiplied by the pixel format (in bpp, see REG[09F6h] bits 3-2) is divisible by 64.

REG[09F6h] Camera1 Write Control Register Default = 00h							Read/Write
Camera1 Double-Buffer Method Select	Reserved	n/a		Camera1 Pixel Format bits 1-0		n/a	Camera1 Vertical Flip Enable
7	6	5	4	3	2	1	0

bit 7 Camera1 Double-Buffer Method Select
This bit selects the double-buffering method used for Camera1 input image data.
When this bit = 0b, method 0 is used.
When this bit = 1b, method 1 is used. This method can be used when the LCD refresh rate is faster than the Camera1 input stream rate.

Note

1. When this bit = 1b, Vertical Flip must be disabled.
   Vertical Flip for the destination (or consumer) must also be disabled (REG[0940h]/[0960h]/[0980h]/[09B0h] bit 5 = 0b).
2. When this bit = 1b, software frame control is not supported for double buffering to the destination window (REG[09D8h]/[09D9h]/[09DAh]/[09DBh] bit 0 must be set to 1b). Also, when this bit = 1b, the frame buffer ready bits for the destination windows are invalid (see REG[0942h]/[0962h]/[0982h]/[09B2h] bits 1-0).
3. When this bit = 1b, Camera1 cannot be the source for Warp Hardware Frame Control (see REG[09DCh] and REG[09DEh] bits 3-0).

bit 6 Reserved
This bit must be set to 0b.

bits 3-2 Camera1 Pixel Format bits [1:0]
These bits determine the RGB pixel format of the Camera1 image data stored in SDRAM.

Table 10-50: Camera1 Pixel Format Selection

REG[09F6h] bits 3-2	Pixel Format
00b	8 bpp (RGB 3:3:2)
01b	16 bpp (RGB 5:6:5)
10b	24 bpp (RGB 8:8:8)
11b	Reserved

bit 0 Camera1 Vertical Flip Enable
This bit determines whether the image data input from the Camera1 interface is flipped around the X axis (vertical).
When this bit = 0b, the Camera1 image data is not vertically flipped (disabled).
When this bit = 1b, the Camera1 image data is vertically flipped (enabled).

REG[09F8h] Camera2 Frame Buffer Width Register 0 Default = 40h							Read/Write
Camera2 Frame Buffer Width bits 7-0							
7	6	5	4	3	2	1	0

REG[09F9h] Camera2 Frame Buffer Width Register 1 Default = 01h							Read/Write
n/a					Camera2 Frame Buffer Width bits 10-8		
7	6	5	4	3	2	1	0

REG[09F9h] bits 2-0
REG[09F8h] bits 7-0 Camera2 Frame Buffer Width bits [10:0]
These bits specify the width of the Camera2 frame buffer, in pixels.

Note
The Camera2 frame buffer width must be set such that the width multiplied by the pixel format (in bpp, see REG[09FEh] bits 3-2) is divisible by 64.

REG[09FAh] Camera2 Frame Buffer Height Register 0 Default = F0h							Read/Write
Camera2 Frame Buffer Height bits 7-0							
7	6	5	4	3	2	1	0

REG[09FBh] Camera2 Frame Buffer Height Register 1 Default = 00h							Read/Write
n/a					Camera2 Frame Buffer Height bits 10-8		
7	6	5	4	3	2	1	0

REG[09FBh] bits 2-0
REG[09FAh] bits 7-0 Camera2 Frame Buffer Height bits [10:0]
These bits specify height of the Camera2 frame buffer, in pixels.

REG[09FCh] Camera2 Frame Buffer Virtual Width Register 0 Default = 40h							Read/Write
Camera1 Frame Buffer Virtual Width bits 7-0							
7	6	5	4	3	2	1	0

REG[09FDh] Camera2 Frame Buffer Virtual Width Register 1 Default = 01h							Read/Write
n/a			Camera2 Frame Buffer Virtual Width bits 12-8				
7	6	5	4	3	2	1	0

REG[09FDh] bits 4-0
REG[09FCh] bits 7-0 Camera2 Frame Buffer Virtual Width bits [12:0]
These bits specify the virtual width of the Camera2 frame buffer, in pixels.

Note
The Camera2 frame buffer virtual width must be set such that the virtual width multiplied by the pixel format (in bpp, see REG[09FEh] bits 3-2) is divisible by 64.

<table>
<tr><td colspan="8">REG[09FEh] Camera2 Write Control Register
Default = 00h   Read/Write</td></tr>
<tr><td>Camera2 Double Buffer Method Select</td><td>Reserved</td><td colspan="2">n/a</td><td colspan="2">Camera2 Pixel Format bits 1-0</td><td>n/a</td><td>Camera2 Vertical Flip Enable</td></tr>
<tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
</table>

bit 7 — Camera2 Double Buffer Method Select
This bit selects the double-buffering method used for Camera2 input image data.
When this bit = 0b, method 0 is used.
When this bit = 1b, method 1 is used. This method can be used when the LCD refresh rate is faster than the Camera2 input stream rate.

Note

1. When this bit = 1b, vertical mirroring of the streaming Camera2 image is not supported. Therefore, Camera2 vertical mirroring must be disabled (REG[09FEh] bit 0 = 0b) and the destination window vertical mirroring must be disabled (REG[0940h]/[0960h]/[0980h]/[09B0h] bit 5 = 0b).
2. When this bit = 1b, software frame control is not supported for double buffering to the destination window (REG[09D8h]/[09D9h]/[09DAh]/[09DBh] bit 0 is ignored). Also, when this bit = 1b, the frame buffer ready bits for the destination windows are invalid (see REG[0942h]/[0962h]/[0982h]/[09B2h] bits 1-0).
3. When this bit = 1b, Camera2 cannot be the source for Warp Hardware Frame Control (see REG[09DCh] and REG[09DEh] bits 7-4).

bit 6 — Reserved
This bit must be set to 0b.

bits 3-2 — Camera2 Pixel Format bits [1:0]
These bits determine the RGB pixel format of the Camera2 image data stored in SDRAM.

Table 10-51: Camera2 Pixel Format Selection

REG[09FEh] bits 3-2	Pixel Format
00b	8 bpp (RGB 3:3:2)
01b	16 bpp (RGB 5:6:5)
10b	24 bpp (RGB 8:8:8)
11b	Reserved

bit 0 — Camera2 Vertical Flip Enable
This bit determines whether the image data input from the Camera2 interface is flipped around the X axis (vertical).
When this bit = 0b, the Camera2 image data is not vertically flipped (disabled).
When this bit = 1b, the Camera2 image data is vertically flipped (enabled).

10.4.14 Interrupt Configuration Registers

REG[0A00h] Interrupt Status Register 0 Default = 00h							Read/Write
Sprite Interrupt Status (RO)	I2S DAC Interrupt (RO)	SDRAM Read/Write Buffer Interrupt Status (RO)	n/a	I2S DAC DMA Interrupt Status (RO)	Watchdog Timer Interrupt Status	LCD2 Interrupt Status (RO)	LCD1 Interrupt Status (RO)
7	6	5	4	3	2	1	0

bit 7 Sprite Interrupt Status (Read Only)
This bit indicates the status of the Sprite Interrupt which occurs when a sprite operation completes (REG[5008h] bit 1 = 1b) and the Sprite Operation Complete Interrupt Enable bit is set (REG[5006h] bit 1 = 1b). This interrupt can be configured to cause a Host interrupt signal (see REG[0A06h] bit 7) or a C33PE interrupt signal (see REG[0A0Eh] bit 7).
When this bit = 0b, a Sprite Interrupt has not occurred.
When this bit = 1b, a Sprite Interrupt has occurred.

To clear this status bit, write a 1b to REG[5008h] bit 1.

bit 6 I2S DAC Interrupt Status (Read Only)
This bit indicates the status of the I2S DAC Interrupt which occurs when one of three I2S FIFO interrupts occurs. This bit is the combination (logical OR) of the I2S FIFO Threshold Interrupt Status, I2S FIFO Overrun Interrupt Status, and I2S FIFO Underrun Interrupt Status bits (REG[010Ch] bits 2-0). Each I2S FIFO interrupt status bit can be masked from causing an I2S DAC Interrupt using the corresponding interrupt enable bits in REG[0105h] bit 2-0. This interrupt can be configured to cause a Host interrupt signal (see REG[0A06h] bit 6) or a C33PE interrupt signal (see REG[0A0Eh] bit 6).
When this bit = 0b, an I2S DAC Interrupt has not occurred.
When this bit = 1b, an I2S DAC Interrupt has occurred.

To clear this status bit, write a 1b to the corresponding interrupt status bit in REG[010Ch].

bit 5 SDRAM Read/Write Buffer Interrupt Status (Read Only)
This bit indicates the status of the SDRAM Read/Write Buffer Interrupt which occurs when a transfer between one of the SDRAM buffers and SDRAM completes. This bit is the combination (logical OR) of the SDRAM Buffer 0 Done Interrupt Status/Clear and the SDRAM Buffer 1 Done Interrupt Status/Clear bits (REG[0242h]/[0252h] bit 3). Each SDRAM buffer done interrupt status bit can be masked from causing a SDRAM Read/Write Buffer Interrupt using the corresponding interrupt enable bits in REG[0240h]/[0250h] bit 3. This interrupt can be configured to cause a Host interrupt signal (see REG[0A06h] bit 5) or a C33PE interrupt signal (see REG[0A0Eh] bit 5).
When this bit = 0b, a SDRAM Read/Write Buffer Interrupt has not occurred.
When this bit = 1b, a SDRAM Read/Write Buffer Interrupt has occurred.

To clear this status bit, clear both SDRAM Buffer Done Interrupt Status bits (REG[0242h] bit 3 = 0b and REG[0252h] bit 3 = 0b).

bit 3 I2S DAC DMA Interrupt Status (Read Only)
This bit indicates the status of the I2S DAC DMA Interrupt which occurs when the I2S DMA logic finishes reading from a DAC DMA buffer and switches to reading from the other buffer. This bit mirrors the I2S DMA Interrupt Status bit in REG[0154h] bit 3. This interrupt can be configured to cause a Host interrupt signal (see REG[0A06h] bit 3). This interrupt bit goes to IRQ3 of the C33PE Interrupt Controller (see REG[0A42h] and REG[0A44h].
When this bit = 0b, an I2S DAC DMA Interrupt has not occurred.
When this bit = 1b, an I2S DAC DMA Interrupt has occurred.

To clear this status bit, write a 1b to REG[0154h] bit 3.

bit 2 Watchdog Timer Interrupt Status
This bit indicates the status of the Watchdog Timer Interrupt which occurs when the Watchdog Timer logic finishes counting. This interrupt can be configured to cause a Host interrupt signal (see REG[0A06h] bit 2). This interrupt bit goes to IRQ2 of the C33PE Interrupt Controller (see REG[0A42h] and REG[0A44h].
When this bit = 0b, an Watchdog Timer Interrupt has not occurred.
When this bit = 1b, an Watchdog Timer Interrupt has occurred.

To clear this status bit, write a 1b to this bit.

bit 1 LCD2 Interrupt Status (Read Only)
This bit indicates the status of the LCD2 Interrupt which occurs when a LCD2 VSYNC Interrupt occurs (REG[4037h] bit 3 = 1b) and the LCD2 VSYNC Interrupt is enabled (REG[4019h] bit 7 = 1b).
When this bit = 0b, a LCD2 Interrupt has not occurred.
When this bit = 1b, a LCD2 Interrupt has occurred.

To clear this status bit, write a 1b to REG[4037h] bit 3.

bit 0 LCD1 Interrupt Status (Read Only)
This bit indicates the status of the LCD1 Interrupt which occurs when a LCD1 VSYNC Interrupt occurs (REG[4019h] bit 3 = 1b) and the LCD1 VSYNC Interrupt is enabled (REG[4019h] bit 7 = 1b).
When this bit = 0b, a LCD1 Interrupt has not occurred.
When this bit = 1b, a LCD1 Interrupt has occurred.

To clear this status bit, write a 1b to REG[4019h] bit 3.

REG[0A02h] Interrupt Status Register 1 Default = 00h							Read/Write
Manual C33PE to Host Interrupt Status	Reserved	Reserved	Keypad Interrupt Status (RO)	Timer 1 Interrupt Status	Timer 0 Interrupt Status	DMA Channel 1 Transfer Done Interrupt Status	DMA Channel 0 Transfer Done Interrupt Status
7	6	5	4	3	2	1	0

bit 7 Manual C33PE to Host Interrupt Status
This bit indicates the status of the Manual C33PE to Host Interrupt which can be triggered using the Manual C33PE to Host Interrupt Trigger bit, REG[0A46h] bit 0. The C33PE or the Host itself can trigger this interrupt. This interrupt will only cause a Host interrupt signal if REG[0A08h] bit 7 is set to 1b.
When this bit = 0b, a Manual C33PE to Host Interrupt has not occurred.
When this bit = 1b, a Manual C33PE to Host Interrupt has occurred.

To clear this status bit, write a 1b to this bit.

bit 6 Reserved
The default value for this bit is 0b.

bit 5 Reserved
The default value for this bit is 0b.

bit 4 Keypad Interrupt Status (Read Only)
This bit indicates the status of the Keypad Interrupt which occurs when one of the 25 Keypad Interrupt Status/Clear bits are set in REG[01D0h] ~ REG[01D3h]. Each status bit can be masked from causing a Keypad Interrupt using the corresponding interrupt enable bits in REG[01C4h] ~ REG[01C7h]. This interrupt can be configured to cause a Host interrupt signal (see REG[0A08h] bit 4) or a C33PE interrupt signal (see REG[0A10h] bit 4).
When this bit = 0b, a Keypad Interrupt has not occurred.
When this bit = 1b, a Keypad Interrupt has occurred.

To clear this status bit, clear all the status bits in REG[01D0h] ~ REG[01D3h].

bit 3 Timer 1 Interrupt Status
This bit indicates the status of the Timer 1 Interrupt which occurs when Timer 1 is enabled (REG[0A84h] bit 1 = 1b) and the Timer 1 Period (REG[0A8Ah]) has passed. This bit is not masked by the Timer 1 Interrupt Enable bit, REG[0A08h] bit 3. This interrupt can be configured to cause a Host interrupt signal (see REG[0A08h] bit 3) or a C33PE interrupt signal (see REG[0A10h] bit 3).
When this bit = 0b, a Timer 1 Interrupt has not occurred.
When this bit = 1b, a Timer 1 Interrupt has occurred.

To clear this status bit, write a 1b to this bit.

bit 2 Timer 0 Interrupt Status
This bit indicates the status of the Timer 0 Interrupt which occurs when Timer 0 is enabled (REG[0A84h] bit 0 = 1b) and the Timer 0 Period (REG[0A88h] ~ REG[0A89h]) has passed. This bit is not masked by the Timer 0 Interrupt Enable bit, REG[0A08h] bit 2. This interrupt can be configured to cause a Host interrupt signal (see REG[0A08h] bit 2) or a C33PE interrupt signal (see REG[0A10h] bit 2).
When this bit = 0b, a Timer 0 Interrupt has not occurred.
When this bit = 1b, a Timer 0 Interrupt has occurred.

To clear this status bit, write a 1b to this bit.

bit 1 DMA Channel 1 Transfer Done Interrupt Status
This bit indicates the status of the DMA Channel 1 Transfer Done Interrupt which occurs when a transfer on DMA Channel 1 completes. This bit is not masked by the DMA Channel 1 Transfer Done Interrupt Enable bit, REG[0A08h] bit 1. This interrupt can be configured to cause a Host interrupt signal (see REG[0A08h] bit 1) or a C33PE interrupt signal (see REG[0A10h] bit 1).
When this bit = 0b, a DMA Channel 1 Transfer Done Interrupt has not occurred.
When this bit = 1b, a DMA Channel 1 Transfer Done Interrupt has occurred.

To clear this status bit, write a 1b to this bit.

bit 0 DMA Channel 0 Transfer Done Interrupt Status
This bit indicates the status of the DMA Channel 0 Transfer Done Interrupt which occurs when a transfer on DMA Channel 0 completes. This bit is not masked by the DMA Channel 1 Transfer Done Interrupt Enable bit, REG[0A08h] bit 1. This interrupt can be configured to cause a Host interrupt signal (see REG[0A08h] bit 0) or a C33PE interrupt signal (see REG[0A10h] bit 0).
When this bit = 0b, a DMA Channel 0 Transfer Done Interrupt has not occurred.
When this bit = 1b, a DMA Channel 0 Transfer Done Interrupt has occurred.

To clear this status bit, write a 1b to this bit.

REG[0A04h] Interrupt Status Register 2 Default = 00h							Read/Write
n/a	Image Fetcher Frame Start Interrupt Status	OSD Window Frame Start Interrupt Status	AUX Window Frame Start Interrupt Status	MAIN Window Frame Start Interrupt Status	Warp Logic Frame Buffer Switch Interrupt Status	Warp Logic Luminance Table Interrupt Status	Warp Logic Offset Table Interrupt Status
7	6	5	4	3	2	1	0

bit 6 Image Fetcher Frame Start Interrupt Status
This bit indicates the status of the Image Fetcher Frame Start Interrupt which occurs when the Image Fetcher has started processing a new frame and has latched the width and virtual width registers. This interrupt can be used to prevent the “tearing effect” when programming new width/virtual width values. It can also be used by software to control frame double-buffering. This interrupt can be configured to cause a Host interrupt signal (see REG[0A0Ah] bit 6) or a C33PE interrupt signal (see REG[0A12h] bit 6).
When this bit = 0b, an Image Fetcher Frame Start Interrupt has not occurred.
When this bit = 1b, an Image Fetcher Frame Start Interrupt has occurred.

To clear this status bit, write a 1b to this bit.

bit 5 OSD Window Frame Start Interrupt Status
This bit indicates the status of the OSD Window Frame Start Interrupt which occurs when the Blending Engine has started processing a new OSD window frame and has latched the width and virtual width registers. This interrupt can be used to prevent the “tearing effect” when programming new width/virtual width values. It can also be used by software to control frame double-buffering. This interrupt can be configured to cause a Host interrupt signal (see REG[0A0Ah] bit 5) or a C33PE interrupt signal (see REG[0A12h] bit 5).
When this bit = 0b, an OSD Window Frame Start Interrupt has not occurred.
When this bit = 1b, an OSD Window Frame Start Interrupt has occurred.

To clear this status bit, write a 1b to this bit.

Note

If this interrupt is enabled (REG[0A0Ah] bit 5 = 1b) before the OSD window is enabled (REG[0980h] bit 4 = 1b), the first occurrence of the OSD Window Frame Start Interrupt Status should be ignored and cleared (REG[0A04h] bit 5 = 1b). Any subsequent OSD Window Frame Start Interrupt Status is valid.

bit 4 AUX Window Frame Start Interrupt Status
This bit indicates the status of the AUX Window Frame Start Interrupt which occurs when the Blending Engine has started processing a new AUX window frame and has latched the width and virtual width registers. This interrupt can be used to prevent the “tearing effect” when programming new width/virtual width values. It can also be used by software to control frame double-buffering. This interrupt can be configured to cause a Host interrupt signal (see REG[0A0Ah] bit 4) or a C33PE interrupt signal (see REG[0A12h] bit 4).
When this bit = 0b, an AUX Window Frame Start Interrupt has not occurred.
When this bit = 1b, an AUX Window Frame Start Interrupt has occurred.

To clear this status bit, write a 1b to this bit.

Note

If this interrupt is enabled (REG[0A0Ah] bit 4 = 1b) before the AUX window is enabled (REG[0960h] bit 4 = 1b), the first occurrence of the AUX Window Frame Start Interrupt Status should be ignored and cleared (REG[0A04h] bit 4 = 1b). Any subsequent AUX Window Frame Start Interrupt Status is valid.

bit 3 MAIN Window Frame Start Interrupt Status
This bit indicates the status of the MAIN Window Frame Start Interrupt which occurs when the Blending Engine has started processing a new MAIN window frame and has latched the width and virtual width registers. This interrupt can be used to prevent the “tearing effect” when programming new width/virtual width values. It can also be used by software to control frame double-buffering. This interrupt can be configured to cause a Host interrupt signal (see REG[0A0Ah] bit 3) or a C33PE interrupt signal (see REG[0A12h] bit 3).
When this bit = 0b, a MAIN Window Frame Start Interrupt has not occurred.
When this bit = 1b, a MAIN Window Frame Start Interrupt has occurred.

To clear this status bit, write a 1b to this bit.

bit 2 Warp Logic Frame Buffer Switch Interrupt Status
This bit indicates the status of the Warp Logic Frame Buffer Switch Interrupt which occurs when the Warp Logic switches from reading one frame buffer to the other frame buffer. This interrupt can be configured to cause a Host interrupt signal (see REG[0A0Ah] bit 2) or a C33PE interrupt signal (see REG[0A12h] bit 2).
When this bit = 0b, a Warp Frame Buffer Switch Interrupt has not occurred.
When this bit = 1b, a Warp Frame Buffer Switch Interrupt has occurred.

To clear this status bit, write a 1b to this bit.

bit 1 Warp Logic Luminance Table Interrupt Status
This bit indicates the status of the Warp Logic Luminance Table Interrupt which occurs when the Warp Logic starts using a new luminance table address. It is used by software when updating the Warp Logic Luminance Table SDRAM Start Address registers (REG[0454h] ~ REG[0457h]). When this interrupt occurs, it means software can write the next luminance table start address value to the register. Each time software writes to the start address registers, an internal “start address written” bit inside the Warp Logic is set to indicate to the hardware that a new value has been written. Whenever the Warp Logic finishes processing a frame and starts a new frame, it latches the start address and sets this interrupt bit if its internal “start address written” bit is set. If the “start address written” bit is not set, no interrupt is generated. The “start address written” bit is automatically cleared whenever the start address is latched. This interrupt can be configured to cause a Host interrupt signal (see REG[0A0Ah] bit 1) or a C33PE interrupt signal (see REG[0A12h] bit 1).
When this bit = 0b, a Warp Logic Luminance Table Interrupt has not occurred.
When this bit = 1b, a Warp Logic Luminance Table Interrupt has occurred.

To clear this status bit, write a 1b to this bit.

bit 0 Warp Logic Offset Table Interrupt Status
This bit indicates the status of the Warp Logic Offset Table Interrupt which occurs when the Warp Logic starts using a new offset table address. It is used by software when updated the Warp Logic Offset Table SDRAM Start Address registers (REG[0444h] ~ REG[0447h]). When this interrupt occurs, it means software can write the next offset table start address value to the register. Each time software writes to the start address registers, an internal "start address written" bit inside the Warp Logic is set to indicate to the hardware that a new value has been written. Whenever the Warp finishes processing a frame and starts a new frame, it latches the start address and sets this interrupt bit if its internal "start address written" bit is set. If the "start address written" bit is not set, no interrupt is generated. The "start address written" bit is automatically cleared whenever the start address is latched. This interrupt can be configured to cause a Host interrupt signal (see REG[0A0Ah] bit 0) or a C33PE interrupt signal (see REG[0A12h] bit 0).
When this bit = 0b, a Warp Logic Offset Table Interrupt has not occurred.
When this bit = 1b, a Warp Logic Offset Table Interrupt has occurred.

To clear this status bit, write a 1b to this bit.

REG[0A06h] Host Interrupt Enable Register 0 Default = 00h							Read/Write
Sprite Interrupt Enable	I2S DAC Interrupt Enable	SDRAM Read/Write Buffer Interrupt Enable	n/a	I2S DAC DMA Interrupt Enable	Watchdog Timer Interrupt Enable	Host LCD2 Interrupt Enable	Host LCD1 Interrupt Enable
7	6	5	4	3	2	1	0

Note
The Host Interrupt Enable bit (see REG[0A0Ch] bit 2) is the master Host interrupt control. If REG[0A0Ch] bit 2 = 0b, an interrupt will not be sent to the Host regardless of the individual interrupt settings in this register.

bit 7 Sprite Interrupt Enable
This bit controls whether a Sprite Interrupt can cause a Host interrupt signal. The status of the Sprite Interrupt is indicated by the Sprite Interrupt Status bit, REG[0A00h] bit 7.
When this bit = 0b, a Sprite Interrupt cannot cause a Host interrupt signal.
When this bit = 1b, a Sprite Interrupt can cause a Host interrupt signal.

bit 6 I2S DAC Interrupt Enable
This bit controls whether an I2S DAC Interrupt can cause a Host interrupt signal. The status of the I2S DAC Interrupt is indicated by the I2S DAC Interrupt Status bit, REG[0A00h] bit 6.
When this bit = 0b, an I2S DAC Interrupt cannot cause a Host interrupt signal.
When this bit = 1b, an I2S DAC Interrupt can cause a Host interrupt signal.

bit 5 SDRAM Read/Write Buffer Interrupt Enable
This bit controls whether a SDRAM Read/Write Buffer Interrupt can cause a Host interrupt signal. The status of the SDRAM Read/Write Buffer Interrupt is indicated by the SDRAM Read/Write Buffer Interrupt Status bit, REG[0A00h] bit 5.
When this bit = 0b, a SDRAM Read/Write Buffer Interrupt cannot cause a Host interrupt signal.
When this bit = 1b, a SDRAM Read/Write Buffer Interrupt can cause a Host interrupt signal.

bit 3 I2S DAC DMA Interrupt Enable
This bit controls whether an I2S DAC DMA Interrupt can cause a Host interrupt signal. The status of the I2S DAC DMA Interrupt is indicated by the I2S DAC DMA Interrupt Status bit, REG[0A00h] bit 3.
When this bit = 0b, an I2S DAC DMA Interrupt cannot cause a Host interrupt signal.
When this bit = 1b, an I2S DAC DMA Interrupt can cause a Host interrupt signal.

bit 2 Watchdog Timer Interrupt Enable
This bit controls whether a Watchdog Timer Interrupt can cause a Host interrupt signal. The status of the Watchdog Timer Interrupt is indicated by the Watchdog Timer Interrupt Status bit, REG[0A00h] bit 2.
When this bit = 0b, a Watchdog Timer Interrupt cannot cause a Host interrupt signal.
When this bit = 1b, a Watchdog Timer Interrupt can cause a Host interrupt signal.

bit 1 Host LCD2 Interrupt Enable
This bit controls whether a LCD2 Interrupt can cause a Host interrupt signal. The status of the LCD2 Interrupt is indicated by the LCD2 Interrupt Status bit, REG[0A00h] bit 1.
When this bit = 0b, a LCD2 Interrupt cannot cause a Host interrupt signal.
When this bit = 1b, a LCD2 Interrupt can cause a Host interrupt signal.

bit 0 Host LCD1 Interrupt Enable
This bit controls whether a LCD1 Interrupt can cause a Host interrupt signal. The status of the LCD1 Interrupt is indicated by the LCD1 Interrupt Status bit, REG[0A00h] bit 0.
When this bit = 0b, a LCD1 Interrupt cannot cause a Host interrupt signal.
When this bit = 1b, a LCD1 Interrupt can cause a Host interrupt signal.

REG[0A08h] Host Interrupt Enable Register 1 Default = 80h							Read/Write
Manual C33PE to Host Interrupt Enable	Reserved	Reserved	Keypad Interrupt Enable	Host Timer 1 Interrupt Enable	Host Timer 0 Interrupt Enable	Host DMA Channel 1 Transfer Done Interrupt Enable	Host DMA Channel 0 Transfer Done Interrupt Enable
7	6	5	4	3	2	1	0

Note
The Host Interrupt Enable bit (see REG[0A0Ch] bit 2) is the master Host interrupt control. If REG[0A0Ch] bit 2 = 0b, an interrupt will not be sent to the Host regardless of the individual interrupt settings in this register.

bit 7 Manual C33PE to Host Interrupt Enable
This bit controls whether a Manual C33PE to Host Interrupt can cause a Host interrupt signal. The status of the Manual C33PE to Host Interrupt is indicated by the Manual C33PE to Host Interrupt Status bit, REG[0A02h] bit 7.
When this bit = 0b, a Manual C33PE to Host Interrupt cannot cause a Host interrupt signal.
When this bit = 1b, a Manual C33PE to Host Interrupt can cause a Host interrupt signal.

bit 6 Reserved
The default value for this bit is 0b.

bit 5 Reserved
The default value for this bit is 0b.

bit 4 Keypad Interrupt Enable
This bit controls whether a Keypad Interrupt can cause a Host interrupt signal. The status of the Keypad Interrupt is indicated by the Keypad Interrupt Status bit, REG[0A02h] bit 4.
When this bit = 0b, a Keypad Interrupt cannot cause a Host interrupt signal.
When this bit = 1b, a Keypad Interrupt can cause a Host interrupt signal.

Note

After enabling the keypad (REG[01C0h] bit 0 = 1b), all interrupts in REG[01C4h] ~ REG[01C7h] should be cleared before enabling the Keypad Host Interrupt.

bit 3 Host Timer 1 Interrupt Enable
This bit controls whether a Host Timer 1 Interrupt can cause a Host interrupt signal. The status of the Host Timer 1 Interrupt is indicated by the Host Timer 1 Interrupt Status bit, REG[0A02h] bit 3.
When this bit = 0b, a Host Timer 1 Interrupt cannot cause a Host interrupt signal.
When this bit = 1b, a Host Timer 1 Interrupt can cause a Host interrupt signal.

bit 2 Host Timer 0 Interrupt Enable
This bit controls whether a Host Timer 0 Interrupt can cause a Host interrupt signal. The status of the Host Timer 0 Interrupt is indicated by the Host Timer 0 Interrupt Status bit, REG[0A02h] bit 2.
When this bit = 0b, a Host Timer 0 Interrupt cannot cause a Host interrupt signal.
When this bit = 1b, a Host Timer 0 Interrupt can cause a Host interrupt signal.

bit 1 Host DMA Channel 1 Transfer Done Interrupt Enable
This bit controls whether a Host DMA Channel 1 Transfer Done Interrupt can cause a Host interrupt signal. The status of the Host DMA Channel 1 Transfer Done Interrupt is indicated by the Host DMA Channel 1 Transfer Done Interrupt Status bit, REG[0A02h] bit 1.
When this bit = 0b, a Host DMA Channel 1 Transfer Done Interrupt cannot cause a Host interrupt signal.
When this bit = 1b, a Host DMA Channel 1 Transfer Done Interrupt can cause a Host interrupt signal.

bit 0 Host DMA Channel 0 Transfer Done Interrupt Enable
This bit controls whether a Host DMA Channel 0 Transfer Done Interrupt can cause a Host interrupt signal. The status of the Host DMA Channel 0 Transfer Done Interrupt is indicated by the Host DMA Channel 0 Transfer Done Interrupt Status bit, REG[0A02h] bit 0.
When this bit = 0b, a Host DMA Channel 0 Transfer Done Interrupt cannot cause a Host interrupt signal.
When this bit = 1b, a Host DMA Channel 0 Transfer Done Interrupt can cause a Host interrupt signal.

REG[0A0Ah] Host Interrupt Enable Register 2 Default = 00h							Read/Write
n/a	Image Fetcher Frame Start Interrupt Enable	OSD Window Frame Start Interrupt Enable	AUX Window Frame Start Interrupt Enable	MAIN Window Frame Start Interrupt Enable	Warp Logic Frame Buffer Switch Interrupt Enable	Warp Logic Luminance Table Interrupt Enable	Warp Logic Offset Table Interrupt Enable
7	6	5	4	3	2	1	0

Note
The Host Interrupt Enable bit (see REG[0A0Ch] bit 2) is the master Host interrupt control. If REG[0A0Ch] bit 2 = 0b, an interrupt will not be sent to the Host regardless of the individual interrupt settings in this register.

bit 6 Image Fetcher Frame Start Interrupt Enable
This bit controls whether an Image Fetcher Frame Start Interrupt can cause a Host interrupt signal. The status of the Image Fetcher Frame Start Interrupt is indicated by the Image Fetcher Frame Start Interrupt Status bit, REG[0A04h] bit 6.
When this bit = 0b, an Image Fetcher Frame Start Interrupt cannot cause a Host interrupt signal.
When this bit = 1b, an Image Fetcher Frame Start Interrupt can cause a Host interrupt signal.

bit 5 OSD Window Frame Start Interrupt Enable
This bit controls whether an OSD Window Frame Start Interrupt can cause a Host interrupt signal. The status of the OSD Window Frame Start Interrupt is indicated by the OSD Window Frame Start Interrupt Status bit, REG[0A04h] bit 5.
When this bit = 0b, an OSD Window Frame Start Interrupt cannot cause a Host interrupt signal.
When this bit = 1b, an OSD Window Frame Start Interrupt can cause a Host interrupt signal.

bit 4 AUX Window Frame Start Interrupt Enable
This bit controls whether an AUX Window Frame Start Interrupt can cause a Host interrupt signal. The status of the AUX Window Frame Start Interrupt is indicated by the AUX Window Frame Start Interrupt Status bit, REG[0A04h] bit 4.
When this bit = 0b, an AUX Window Frame Start Interrupt cannot cause a Host interrupt signal.
When this bit = 1b, an AUX Window Frame Start Interrupt can cause a Host interrupt signal.

bit 3 MAIN Window Frame Start Interrupt Enable
This bit controls whether a MAIN Window Frame Start Interrupt can cause a Host interrupt signal. The status of the MAIN Window Frame Start Interrupt is indicated by the MAIN Window Frame Start Interrupt Status bit, REG[0A04h] bit 3.
When this bit = 0b, a MAIN Window Frame Start Interrupt cannot cause a Host interrupt signal.
When this bit = 1b, a MAIN Window Frame Start Interrupt can cause a Host interrupt signal.

bit 2 Warp Logic Frame Buffer Switch Interrupt Enable
This bit controls whether a Warp Logic Frame Buffer Switch Interrupt can cause a Host interrupt signal. The status of the Warp Logic Frame Buffer Switch Interrupt is indicated by the Warp Logic Frame Buffer Switch Interrupt Status bit, REG[0A04h] bit 2.
When this bit = 0b, a Warp Logic Frame Buffer Switch Interrupt cannot cause a Host interrupt signal.
When this bit = 1b, a Warp Logic Frame Buffer Switch Interrupt can cause a Host interrupt signal.

bit 1 Warp Logic Luminance Table Interrupt Enable
This bit controls whether a Warp Logic Luminance Table Interrupt can cause a Host interrupt signal. The status of the Warp Logic Luminance Table Interrupt is indicated by the Warp Logic Luminance Table Interrupt Status bit, REG[0A04h] bit 1.
When this bit = 0b, a Warp Logic Luminance Table Interrupt cannot cause a Host interrupt signal.
When this bit = 1b, a Warp Logic Luminance Table Interrupt can cause a Host interrupt signal.

bit 0 Warp Logic Offset Table Interrupt Enable
This bit controls whether a Warp Logic Offset Table Interrupt can cause a Host interrupt signal. The status of the Warp Logic Offset Table Interrupt is indicated by the Warp Logic Offset Table Interrupt Status bit, REG[0A04h] bit 0.
When this bit = 0b, a Warp Logic Offset Table Interrupt cannot cause a Host interrupt signal.
When this bit = 1b, a Warp Logic Offset Table Interrupt can cause a Host interrupt signal.

REG[0A0Ch] Host Interrupt Control Register
Default = 04h Read/Write

n/a	Host Interrupt Pin Tri-state Enable	n/a	Host Interrupt Pin Polarity	n/a	Host Interrupt Enable	n/a	
7	6	5	4	3	2	1	0

bit 6 Host Interrupt Pin Tri-state Enable
When this bit = 0b, the INT pin is driven based on the configuration of the Host Interrupt Pin Polarity bit, REG[0A0Ch] bit 4.
When this bit = 1b, the INT pin is active low and is high impedance (Hi-Z) when no interrupt has occurred.

bit 4 Host Interrupt Pin Polarity
When REG[0A0Ch] bit 6 = 0b, this bit controls the polarity of the Host interrupt pin, INT.
When this bit = 0b, the INT pin is active high when a Host interrupt is triggered.
When this bit = 1b, the INT pin is active low when a Host interrupt is triggered.

bit 2 Host Interrupt Enable
This bit is the Host Interrupt master control.
When this bit = 0b, the interrupt status bits in REG[0A00h] ~ REG[0A04h] cannot cause a Host interrupt (INT pin is disabled).
When this bit = 1b, the interrupt status bits in REG[0A00h] ~ REG[0A04h] can cause a Host interrupt (INT pin is enabled) when the corresponding interrupt enable bit is set (see REG[0A06h] ~ REG[0A0Ah].

REG[0A0Eh] through REG[0A46h]

REG[0A0Eh] through REG[0A46h] are typically used by the C33PE and are not accessed by the Host.

REG[0A0Eh] C33PE Device Interrupt Enable Register 0 Default = 00h							Read/Write
C33PE Sprite Interrupt Enable	C33PE I2S DAC Interrupt Enable	C33PE SDRAM Read/Write Buffer Interrupt Enable	n/a			C33PE LCD2 Interrupt Enable	C33PE LCD1 Interrupt Enable
7	6	5	4	3	2	1	0

Note
C33PE Interrupt Enable bit 0 must be set (REG[0A42h] bit 0 = 1b) or an interrupt will not be sent to the C33PE regardless of the individual interrupt settings in this register.

bit 7 C33PE Sprite Interrupt Enable
This bit controls whether a Sprite Interrupt can cause a C33PE interrupt signal. The status of the Sprite Interrupt is indicated by the Sprite Interrupt Status bit, REG[0A00h] bit 7.
When this bit = 0b, a Sprite Interrupt cannot cause a C33PE interrupt signal.
When this bit = 1b, a Sprite Interrupt can cause a C33PE interrupt signal.

bit 6 C33PE I2S DAC Interrupt Enable
This bit controls whether an I2S DAC Interrupt can cause a C33PE interrupt signal. The status of the I2S DAC Interrupt is indicated by the I2S DAC Interrupt Status bit, REG[0A00h] bit 6.
When this bit = 0b, an I2S DAC Interrupt cannot cause a C33PE interrupt signal.
When this bit = 1b, an I2S DAC Interrupt can cause a C33PE interrupt signal.

bit 5 C33PE SDRAM Read/Write Buffer Interrupt Enable
This bit controls whether a SDRAM Read/Write Buffer Interrupt can cause a C33PE interrupt signal. The status of the SDRAM Read/Write Buffer Interrupt is indicated by the SDRAM Read/Write Buffer Interrupt Status bit, REG[0A00h] bit 5.
When this bit = 0b, a SDRAM Read/Write Buffer Interrupt cannot cause a C33PE interrupt signal.
When this bit = 1b, a SDRAM Read/Write Buffer Interrupt can cause a C33PE interrupt signal.

bit 1 C33PE LCD2 Interrupt Enable
This bit controls whether a LCD2 Interrupt can cause a C33PE interrupt signal. The status of the LCD2 Interrupt is indicated by the LCD2 Interrupt Status bit, REG[0A00h] bit 1.
When this bit = 0b, the LCD2 Interrupt cannot cause a C33PE interrupt signal.
When this bit = 1b, the LCD2 Interrupt can cause a C33PE interrupt signal.

bit 0 C33PE LCD1 Interrupt Enable
This bit controls whether a LCD1 Interrupt can cause a C33PE interrupt signal. The status of the LCD1 Interrupt is indicated by the LCD1 Interrupt Status bit, REG[0A00h] bit 0.
When this bit = 0b, the LCD1 Interrupt cannot cause a C33PE interrupt signal.
When this bit = 1b, the LCD1 Interrupt can cause a C33PE interrupt signal.

REG[0A10h] C33PE Device Interrupt Enable Register 1 Default = 00h							Read/Write
n/a	Reserved	Reserved	C33PE Keypad Interrupt Enable	C33PE Timer 1 Interrupt Enable	C33PE Timer 0 Interrupt Enable	C33PE DMA Channel 1 Transfer Done Interrupt Enable	C33PE DMA Channel 0 Transfer Done Interrupt Enable
7	6	5	4	3	2	1	0

Note

C33PE Interrupt Enable bit 0 must be set (REG[0A42h] bit 0 = 1b) or an interrupt will not be sent to the C33PE regardless of the individual interrupt settings in this register.

bit 6 Reserved
The default value for this bit is 0b.

bit 5 Reserved
The default value for this bit is 0b.

bit 4 C33PE Keypad Interrupt Enable
This bit controls whether a Keypad Interrupt can cause a C33PE interrupt signal. The status of the Keypad Interrupt is indicated by the Keypad Interrupt Status bit, REG[0A02h] bit 4.
When this bit = 0b, a Keypad Interrupt cannot cause a C33PE interrupt signal.
When this bit = 1b, a Keypad Interrupt can cause a C33PE interrupt signal.

bit 3 C33PE Timer 1 Interrupt Enable
This bit controls whether a Timer 1 Interrupt can cause a C33PE interrupt signal. The status of the Timer 1 Interrupt is indicated by the Timer 1 Interrupt Status bit, REG[0A02h] bit 3.
When this bit = 0b, a Timer 1 Interrupt cannot cause a C33PE interrupt signal.
When this bit = 1b, a Timer 1 Interrupt can cause a C33PE interrupt signal.

bit 2 C33PE Timer 0 Interrupt Enable
This bit controls whether a Timer 0 Interrupt can cause a C33PE interrupt signal. The status of the Timer 0 Interrupt is indicated by the Timer 0 Interrupt Status bit, REG[0A02h] bit 2.
When this bit = 0b, a Timer 0 Interrupt cannot cause a C33PE interrupt signal.
When this bit = 1b, a Timer 0 Interrupt can cause a C33PE interrupt signal.

bit 1 C33PE DMA Channel 1 Transfer Done Interrupt Enable
This bit controls whether a DMA Channel 1 Transfer Done Interrupt can cause a C33PE interrupt signal. The status of the DMA Channel 1 Transfer Done Interrupt is indicated by the DMA Channel 1 Transfer Done Interrupt Status bit, REG[0A02h] bit 1.
When this bit = 0b, a DMA Channel 1 Transfer Done Interrupt cannot cause a C33PE interrupt signal.
When this bit = 1b, a DMA Channel 1 Transfer Done Interrupt can cause a C33PE interrupt signal.

bit 0 C33PE DMA Channel 0 Transfer Done Interrupt Enable
This bit controls whether a DMA Channel 0 Transfer Done Interrupt can cause a C33PE interrupt signal. The status of the DMA Channel 0 Transfer Done Interrupt is indicated by the DMA Channel 0 Transfer Done Interrupt Status bit, REG[0A02h] bit 0.
When this bit = 0b, a DMA Channel 0 Transfer Done Interrupt cannot cause a C33PE interrupt signal.
When this bit = 1b, a DMA Channel 0 Transfer Done Interrupt can cause a C33PE interrupt signal.

REG[0A12h] C33PE Device Interrupt Enable Register 2 Default = 00h Read/Write							
n/a	C33PE Image Fetcher Frame Start Interrupt Enable	C33PE OSD Window Frame Start Interrupt Enable	C33PE AUX Window Frame Start Interrupt Enable	C33PE MAIN Window Frame Start Interrupt Enable	C33PE Warp Logic Frame Buffer Switch Interrupt Enable	C33PE Warp Logic Luminance Table Interrupt Enable	C33PE Warp Logic Offset Table Interrupt Enable
7	6	5	4	3	2	1	0

Note

C33PE Interrupt Enable bit 0 must be set (REG[0A42h] bit 0 = 1b) or an interrupt will not be sent to the C33PE regardless of the individual interrupt settings in this register.

bit 6 C33PE Image Fetcher Frame Start Interrupt Enable
This bit controls whether an Image Fetcher Frame Start Interrupt can cause a C33PE interrupt signal. The status of the Image Fetcher Frame Start Interrupt is indicated by the Image Fetcher Frame Start Interrupt Status bit, REG[0A04h] bit 6.
When this bit = 0b, an Image Fetcher Frame Start Interrupt cannot cause a C33PE interrupt signal.
When this bit = 1b, an Image Fetcher Frame Start Interrupt can cause a C33PE interrupt signal.

bit 5 C33PE OSD Window Frame Start Interrupt Enable
This bit controls whether an OSD Window Frame Start Interrupt can cause a C33PE interrupt signal. The status of the OSD Window Frame Start Interrupt is indicated by the OSD Window Frame Start Interrupt Status bit, REG[0A04h] bit 5.
When this bit = 0b, an OSD Window Frame Start Interrupt cannot cause a C33PE interrupt signal.
When this bit = 1b, an OSD Window Frame Start Interrupt can cause a C33PE interrupt signal.

bit 4 C33PE AUX Window Frame Start Interrupt Enable
This bit controls whether an AUX Window Frame Start Interrupt can cause a C33PE interrupt signal. The status of the AUX Window Frame Start Interrupt is indicated by the AUX Window Frame Start Interrupt Status bit, REG[0A04h] bit 4.
When this bit = 0b, an AUX Window Frame Start Interrupt cannot cause a C33PE interrupt signal.
When this bit = 1b, an AUX Window Frame Start Interrupt can cause a C33PE interrupt signal.

bit 3 C33PE MAIN Window Frame Start Interrupt Enable
This bit controls whether a MAIN Window Frame Start Interrupt can cause a C33PE interrupt signal. The status of the MAIN Window Frame Start Interrupt is indicated by the MAIN Window Frame Start Interrupt Status bit, REG[0A04h] bit 3.
When this bit = 0b, a MAIN Window Frame Start Interrupt cannot cause a C33PE interrupt signal.
When this bit = 1b, a MAIN Window Frame Start Interrupt can cause a C33PE interrupt signal.

bit 2 — C33PE Warp Logic Frame Buffer Switch Interrupt Enable
This bit controls whether a Warp Logic Frame Buffer Switch Interrupt can cause a C33PE interrupt signal. The status of the Warp Logic Frame Buffer Switch Interrupt is indicated by the Warp Logic Frame Buffer Switch Interrupt Status bit, REG[0A04h] bit 2.
When this bit = 0b, a Warp Logic Frame Buffer Switch Interrupt cannot cause a C33PE interrupt signal.
When this bit = 1b, a Warp Logic Frame Buffer Switch Interrupt can cause a C33PE interrupt signal.

bit 1 — C33PE Warp Logic Luminance Table Interrupt Enable
This bit controls whether a Warp Logic Luminance Table Interrupt can cause a C33PE interrupt signal. The status of the Warp Logic Luminance Table Interrupt is indicated by the Warp Logic Luminance Table Interrupt Status bit, REG[0A04h] bit 1.
When this bit = 0b, a Warp Logic Luminance Table Interrupt cannot cause a C33PE interrupt signal.
When this bit = 1b, a Warp Logic Luminance Table Interrupt can cause a C33PE interrupt signal.

bit 0 — C33PE Warp Logic Offset Table Interrupt Enable
This bit controls whether a Warp Logic Offset Table Interrupt can cause a C33PE interrupt signal. The status of the Warp Logic Offset Table Interrupt is indicated by the Warp Logic Offset Table Interrupt Status bit, REG[0A04h] bit 0.
When this bit = 0b, a Warp Logic Offset Table Interrupt cannot cause a C33PE interrupt signal.
When this bit = 1b, a Warp Logic Offset Table Interrupt can cause a C33PE interrupt signal.

REG[0A20h] C33PE Interrupt 0 Control Register 0
Default = 10h — Read/Write

7	6	5	4	3	2	1	0
C33PE Interrupt 0 Vector Number bits 7-0							

bits 7-0 — C33PE Interrupt 0 Vector Number bits [7:0]
These bits specify the vector number for C33PE interrupt 0.

REG[0A21h] C33PE Interrupt 0 Control Register 1
Default = 0Fh — Read/Write

7	6	5	4	3	2	1	0
n/a				C33PE Interrupt 0 Priority Level bits 3-0			

bits 3-0 — C33PE Interrupt 0 Priority Level bits [3:0]
These bits specify the priority level for C33PE interrupt 0.

REG[0A22h] C33PE Interrupt 1 Control Register 0
Default = 11h — Read/Write

7	6	5	4	3	2	1	0
C33PE Interrupt 1 Vector Number bits 7-0							

bits 7-0 — C33PE Interrupt 1 Vector Number bits [7:0]
These bits specify the vector number for C33PE interrupt 1.

REG[0A23h] C33PE Interrupt 1 Control Register 1 Default = 01h							Read/Write
n/a				C33PE Interrupt 1 Priority Level bits 3-0			
7	6	5	4	3	2	1	0

bits 3-0 C33PE Interrupt 1 Priority Level bits [3:0]
These bits specify the priority level for C33PE interrupt 1.

REG[0A24h] C33PE Interrupt 2 Control Register 0 Default = 12h							Read/Write
C33PE Interrupt 2 Vector Number bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 C33PE Interrupt 2 Vector Number bits [7:0]
These bits specify the vector number for C33PE interrupt 2.

REG[0A25h] C33PE Interrupt 2 Control Register 1 Default = 01h							Read/Write
n/a				C33PE Interrupt 2 Priority Level bits 3-0			
7	6	5	4	3	2	1	0

bits 3-0 C33PE Interrupt 2 Priority Level bits [3:0]
These bits specify the priority level for C33PE interrupt 2.

REG[0A26h] C33PE Interrupt 3 Control Register 0 Default = 13h							Read/Write
C33PE Interrupt 3 Vector Number bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 C33PE Interrupt 3 Vector Number bits [7:0]
These bits specify the vector number for C33PE interrupt 3.

REG[0A27h] C33PE Interrupt 3 Control Register 1 Default = 01h							Read/Write
n/a				C33PE Interrupt 3 Priority Level bits 3-0			
7	6	5	4	3	2	1	0

bits 3-0 C33PE Interrupt 3 Priority Level bits [3:0]
These bits specify the priority level for C33PE interrupt 3.

REG[0A28h] C33PE Interrupt 4 Control Register 0 Default = 14h							Read/Write
C33PE Interrupt 4 Vector Number bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 C33PE Interrupt 4 Vector Number bits [7:0]
These bits specify the vector number for C33PE interrupt 4.

REG[0A29h] C33PE Interrupt 4 Control Register 1 Default = 01h							Read/Write
n/a				C33PE Interrupt 4 Priority Level bits 3-0			
7	6	5	4	3	2	1	0

bits 3-0 C33PE Interrupt 4 Priority Level bits [3:0]
These bits specify the priority level for C33PE interrupt 4.

REG[0A2Ah] C33PE Interrupt 5 Control Register 0 Default = 15h							Read/Write
C33PE Interrupt 5 Vector Number bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 C33PE Interrupt 5 Vector Number bits [7:0]
These bits specify the vector number for C33PE interrupt 5.

REG[0A2Bh] C33PE Interrupt 5 Control Register 1 Default = 0Fh							Read/Write
n/a				C33PE Interrupt 5 Priority Level bits 3-0			
7	6	5	4	3	2	1	0

bits 3-0 C33PE Interrupt 5 Priority Level bits [3:0]
These bits specify the priority level for C33PE interrupt 5.

REG[0A2Ch] C33PE Interrupt 6 Control Register 0 Default = 16h							Read/Write
C33PE Interrupt 6 Vector Number bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 C33PE Interrupt 6 Vector Number bits [7:0]
These bits specify the vector number for C33PE interrupt 6.

REG[0A2Dh] C33PE Interrupt 6 Control Register 1 Default = 0Dh							Read/Write
n/a				C33PE Interrupt 6 Priority Level bits 3-0			
7	6	5	4	3	2	1	0

bits 3-0 C33PE Interrupt 6 Priority Level bits [3:0]
These bits specify the priority level for C33PE interrupt 6.

REG[0A2Eh] C33PE Interrupt 7 Control Register 0 Default = 17h							Read/Write
C33PE Interrupt 7 Vector Number bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 C33PE Interrupt 7 Vector Number bits [7:0]
These bits specify the vector number for C33PE interrupt 7.

REG[0A2Fh] C33PE Interrupt 7 Control Register 1 Default = 0Ch							Read/Write
n/a				C33PE Interrupt 7 Priority Level bits 3-0			
7	6	5	4	3	2	1	0

bits 3-0 C33PE Interrupt 7 Priority Level bits [3:0]
These bits specify the priority level for C33PE interrupt 7.

REG[0A40h] C33PE Manual Interrupt Trigger Register Default = 00h							Write Only
C33PE Manual Interrupt Trigger bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 C33PE Manual Interrupt Trigger bits [7:0] (Write Only)
These bits allow manual triggering of the corresponding C33PE interrupts. When each interrupt is triggered, the corresponding bit in REG[0A44h] will indicate a 1b showing the interrupt status until the interrupt is cleared. Only the interrupts enabled using REG[0A42h] will cause an interrupt request to the C33PE.
Writing a 0b to this bit has no hardware effect.
Writing a 1b to this bit manually triggers the corresponding interrupt.

Note

1. C33PE Interrupt 0 is triggered by devices, which includes any of the enabled interrupts from REG[0A00h] ~ REG[0A04h]. These interrupts should be enabled specifically for the C33PE using REG[0A0Eh] ~ REG[0A12h].
2. Interrupt 0 corresponds to the C33PE devices interrupt and cannot be controlled from this register.

REG[0A42h] C33PE Interrupt Enable Register Default = E1h							Read/Write
C33PE Interrupt Enable bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 C33PE Interrupt Enable bits [7:0]
These bits control the corresponding C33PE interrupts. The raw status of these interrupts is available in the C33PE Interrupt Status register, REG[0A44h].
When this bit = 0b, the corresponding interrupt is disabled.
When this bit = 1b, the corresponding interrupt is enabled.

REG[0A43h] C33PE NMI Interrupt Enable Register Default = 80h							Read/Write
C33PE NMI Interrupt Enable	n/a						
7	6	5	4	3	2	1	0

bit 7 C33PE NMI Interrupt Enable
This bit controls the C33PE NMI interrupt.
When this bit = 0b, the NMI interrupt is not triggered.
When this bit = 1b, the NMI interrupt is triggered when the Timer 0 Period (REG[0A88h] ~ REG[0A89h]) is exceeded.

REG[0A44h] C33PE Interrupt Status Register Default = 00h							Read/Write
C33PE Interrupt Status bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 C33PE Interrupt Status bits [7:0]
These bits indicate the raw status of the corresponding C33PE interrupt. These bits are not masked by the corresponding bit in the C33PE Interrupt Enable register, REG[0A42h].
When this bit = 0b, the corresponding interrupt has not occurred.
When this bit = 1b, the corresponding interrupt has occurred.

To clear these interrupts (except interrupt 0, 2, and 3 which are read only), write a 1b to the corresponding status bit.

Note

Interrupt 0 corresponds to the C33PE devices interrupt and cannot be controlled from this register.
Interrupt 2 corresponds to the Watchdog Interrupt which can be read and cleared in Interrupt Status Register 0 (REG[0A00h]) bit 2.
Interrupt 3 corresponds to the I2S DAC DMA interrupt which can be read and cleared in the I2S DMA Status Register (REG[0154h] bit 3).

REG[0A46h] C33 to Host Interrupt Trigger Register Default = 00h							Write Only
n/a							Manual C33PE to Host Interrupt Trigger
7	6	5	4	3	2	1	0

bit 0 Manual C33PE to Host Interrupt Trigger (Write Only)
This bit is the trigger for the Manual C33PE to Host Interrupt. This interrupt is used by the C33PE to signal the Host. The status of the Manual C33PE to Host Interrupt is indicated by the Manual C33PE to Host Interrupt Status bit, REG[0A02h] bit 7. This interrupt will only cause a Host interrupt signal if REG[0A08h] bit 7 is set to 1b.
Writing a 0b to this bit has no hardware effect.
Writing a 1b to this bit triggers a Manual C33PE to Host Interrupt.

10.4.15 Timer Configuration Registers

REG[0A80h] Timer Clock Configuration Register 0 Default = 24h							Read/Write
Timer Clock Divide Select bits 7-0							
7	6	5	4	3	2	1	0

REG[0A81h] Timer Clock Configuration Register 1 Default = 00h							Read/Write
n/a				Timer Clock Divide Select bits 11-8			
7	6	5	4	3	2	1	0

REG[0A81h] bits 3-0
REG[0A80h] bits 7-0 Timer Clock Divide Select bits [11:0]

These bits determine the divide ratio for the Timer Clock (LSCLK) which is used for Timer 0, Timer 1, and the Watchdog Timer. The Timer Clock is derived from the input clock INCLK1 which is sourced from either CLKI or OSCI depending on the setting of CNF0. For further details on the clock structure, see Section Chapter 9, “Clocks” on page 131.

The divide ratio should be set appropriately for use by the timers according to the following formula.

Time Clock Divide Ratio = 1: (REG[0A81h] bits 3-0, REG[0A80h] bits 7-0) + 1

REG[0A84h] Timer Control Register Default = 01h							Read/Write
n/a				Watchdog Time-out Action	Watchdog Timer Enable	Timer 1 Enable	Timer 0 Enable
7	6	5	4	3	2	1	0

bit 3 Watchdog Time-out Action

These bit only has an effect when the Watchdog Timer is enabled, REG[0A84h] bit 2 = 1b. This bit determines what happens when a Watchdog Timer time-out occurs. A time-out occurs when the Watchdog Timer Period is reached (see REG[0A86h] ~ REG[0A87h]). To reset the counter and prevent a time-out from occurring, periodically write the value of 2371h to the Watchdog Timer Clear registers, REG[0A8Ch] ~ REG[0A8Dh].
When this bit = 0b, a watchdog timer time-out generates an IRQ2 interrupt.
When this bit = 1b, the watchdog timer time-out generates a system reset.

bit 2 Watchdog Timer Enable

This bit controls the Watchdog Timer. The Watchdog Timer Period bits (REG[0A86h] ~ REG[0A87h]) must be set before the timer is enabled.
When this bit = 0b, the Watchdog Timer is disabled. (default)
When this bit = 1b, the Watchdog Timer is enabled.

bit 1 Timer 1 Enable

This bit controls Timer 1. The Timer 1 Period bits (REG[0A8Ah] bits 7-0) must be set before the timer is enabled. The status of the timer is indicated by the Timer 1 Interrupt Status bit, REG[0A02h] bit 3.
When this bit = 0b, Timer 1 is disabled. (default)
When this bit = 1b, Timer 1 is enabled.

bit 0 Timer 0 Enable
This bit controls Timer 0 which can be used to generate a C33PE NMI interrupt (see REG[0A43h]). The Timer 0 Period bits (REG[0A88h] ~ REG[0A89h]) must be set before the timer is enabled. The status of the timer is indicated by the Timer 0 Interrupt Status bit, REG[0A02h] bit 2.
When this bit = 0b, the Timer 0 is disabled. (default)
When this bit = 1b, the Timer 0 is enabled.

REG[0A86h] Watchdog Timer Period Register 0 Default = 00h							Read/Write
Watchdog Timer Period bits 7-0							
7	6	5	4	3	2	1	0

REG[0A87h] Watchdog Timer Period Register 1 Default = 00h							Read/Write
Watchdog Timer Period bits 15-8							
7	6	5	4	3	2	1	0

REG[0A87h] bits 7-0
REG[0A86h] bits 7-0 Watchdog Timer Period bits [15:0]
These bits only have an effect when the Watchdog Timer is enabled, REG[0A84h] bit 2 = 1b. These bits determine the period, in number of Timer Clocks (LSCLK), that the timer counts before triggering the Watchdog Time-out Action (see REG[0A84h] bit 3). To reset the counter and prevent a time-out from occurring, periodically write the value of 2371h to the Watchdog Timer Clear registers, REG[0A8Ch] ~ REG[0A8Dh]. The Watchdog Timer period is defined by the following formulas.

Initial Timer Period max. = ((REG[0A87h] bits 7-0, REG[0A86h] bits 7-0) - 1) x LSCLKs
Initial Timer Period min. = ((REG[0A87h] bits 7-0, REG[0A86h] bits 7-0) - 2) x LSCLKs

Subsequent Timer Period = ((REG[0A87h] bits 7-0, REG[0A86h] bits 7-0) - 1) x LSCLKs

Note
The Watchdog Timer Period bits must not be set to 0000h as this value causes a delay of 65536 LSCLKs.

REG[0A88h] Timer 0 Period Register 0 Default = E8h							Read/Write
Timer 0 Period bits 7-0							
7	6	5	4	3	2	1	0

REG[0A89h] Timer 0 Period Register 1 Default = 03h							Read/Write
n/a				Timer 0 Period bits 11-8			
7	6	5	4	3	2	1	0

REG[0A89h] bits 3-0
REG[0A88h] bits 7-0 Timer 0 Period bits [11:0]

These bits only have an effect when the Timer 0 is enabled, REG[0A84h] bit 0 = 1b. These bits determine the period, in number of Timer Clocks (LSCLK), that the timer counts before triggering the Timer 0 Interrupt Status bit, REG[0A02h] bit 2. The Timer 0 period is defined by the following formulas.

Initial Timer Period max. = ((REG[0A89h] bits 3-0, REG[0A88h] bits 7-0) - 1) x LSCLKs
Initial Timer Period min. = ((REG[0A89h] bits 3-0, REG[0A88h] bits 7-0) - 2) x LSCLKs

Subsequent Timer Period = ((REG[0A89h] bits 3-0, REG[0A88h] bits 7-0) - 1) x LSCLKs

Note

The Timer 0 Period bits must not be set to 000h as this value causes a delay of 8192 LSCLKs.

REG[0A8Ah] Timer 1 Period Register Default = 00h							Read/Write
Timer 1 Period bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 Timer 1 Period bits [7:0]

These bits only have an effect when Timer 1 is enabled, REG[0A84h] bit 1 = 1b. These bits determine the period, in number of Timer Clocks (LSCLK), that the timer counts before triggering the Timer 1 Interrupt Status bit, REG[0A02h] bit 3. The Timer 1 period is defined by the following formulas.

Initial Timer Period max. = (REG[0A8Ah] bits 7-0 - 1) x LSCLKs
Initial Timer Period min. = (REG[0A8Ah] bits 7-0 - 2) x LSCLKs

Subsequent Timer Period = (REG[0A8Ah] bits 7-0 - 1) x LSCLKs

Note

The Timer 1 Period bits must not be set to 00h as this value causes a delay of 8192 LSCLKs.

REG[0A8Ch] Watchdog Timer Clear Register 0 Default = 00h							Write Only
Watchdog Timer Clear bits 7-0							
7	6	5	4	3	2	1	0

REG[0A8Dh] Watchdog Timer Clear Register 1 Default = 00h							Write Only
Watchdog Timer Clear bits 15-8							
7	6	5	4	3	2	1	0

REG[0A8Dh] bits 7-0
REG[0A8Ch] bits 7-0 Watchdog Timer Clear bits [15:0] (Write Only)

When the watchdog timer is enabled (REG[0A84h] bit 2 = 1b), software should periodically write these bits with the 16-bit value of 2371h which will restart the watchdog timer and prevent a time-out from occurring.

10.4.16 SPI Flash Memory Interface Registers

REG[0B00h] SPI Flash Read Data Register Default =FFh							Read Only
SPI Flash Read Data bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 SPI Flash Read Data bits [7:0] (Read Only)
These bits contain the 8-bit value read when a “dummy” write is written to the SPI Flash Memory interface.

REG[0B02h] SPI Flash Write Data Register Default = 00h							Write Only
SPI Flash Write Data bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 SPI Flash Write Data bits [7:0] (Write Only)
These bits are the write data register for the SPI Flash Memory interface. When a value is written to this register, a serial output transfer of the specified value is initiated on the SPI Flash Memory interface.

REG[0B03h] SPI Flash Data Control Register Default = 00h							Read/Write
n/a							SPI Flash Data Output Enable
7	6	5	4	3	2	1	0

bit 0 SPI Flash Data Output Enable
This bit controls data output for the SPI Flash Memory interface data line (SPIDIO pin).
When this bit = 0b, the SPIDIO pin is high impedance allowing SPI Flash Memory reads when the SPI Flash Read Mode is set to 0b, REG[0B04h] bit 7 = 0b.
When this bit = 1b, the SPIDIO pin is driven allowing SPI Flash Memory writes when the SPI Flash Read Mode is set to 0b, REG[0B04h] bit 7 = 0b.

REG[0B04h] SPI Flash Control Register Default = 11h							Read/Write
SPI Flash Read Mode	Reserved	SPI Flash Clock Divide Select bits 2-0			SPI Flash Clock Phase Select	SPI Flash Clock Polarity Select	SPI Flash Enable
7	6	5	4	3	2	1	0

bit 7 SPI Flash Read Mode
This bit selects the mode for reading the SPI Flash Memory.
When this bit = 0b, the SPI Flash Memory is read by firmware through the registers at REG[0B00h].
When this bit = 1b, the SPI Flash Memory is read by firmware at base address 2000_0000h. In this mode, the contents of the flash memory is read by the Serial Flash Read logic which handles serial reads and deserialization of the read data. This mode makes the serial flash memory device accessible like a memory-mapped parallel flash device.

Note
When this bit = 1b, writes to the SPI Flash Memory are not possible.

bit 6 Reserved
This bit is reserved and MUST be set to 1b.

bits 5-3 SPI Flash Clock Divide Select bits [2:0]
These bits select the divide ratio for the SPI Flash clock. The source for the SPI Flash clock is the external SDRAM clock.

Table 10-52: SPI Flash Clock Divide Ratio Selection

REG[0B04h] bits 5-3	SPI Flash Clock Divide Ratio	REG[0B04h] bits 5-3	SPI Flash Clock Divide Ratio
000b	1:2	100b	1:6
001b	1:3	101b	1:7
010b	1:4	110b	1:8
011b	1:5	111b	1:9

Note
For odd SPI clock divides the SPICLK output does not maintain 50/50 duty cycle.

bit 2 SPI Flash Clock Phase Select
This bit selects the SPI Flash clock phase. For a summary of the SPI Flash Memory clock phase and polarity settings, see Table 10-53 “SPI Flash Clock Phase and Polarity,” on page 327.

bit 1 SPI Flash Clock Polarity Select (CPOL)
This bit selects the SPI Flash clock polarity. The following table summarizes the SPI Flash clock polarity and phase settings.

Table 10-53 : SPI Flash Clock Phase and Polarity

REG[0B04h] bit 2	REG[0B04h] bit 1	Valid Data	Clock Idling Status
0b	0b	Rising edge of SPI Flash Clock	Low
	1b	Falling edge of SPI Flash Clock	High
1b	0b	Falling edge of SPI Flash Clock	Low
	1b	Rising edge of SPI Flash Clock	High

bit 0 SPI Flash Enable
This bit controls the SPI Flash Memory interface logic.
When this bit = 0b, the SPI Flash Memory interface is disabled and the SPI Flash Read Port at 2000_0000h must not be accessed.
When this bit = 1b, the SPI Flash Memory interface is enabled.

REG[0B06h] SPI Flash Status Register Default = 04h							Read Only
n/a				SPI Flash Busy Flag	SPI Flash Write Data Register Empty Flag	SPI Flash Read Data Overrun Flag	SPI Flash Read Data Ready Flag
7	6	5	4	3	2	1	0

bit 3 SPI Flash Busy Flag (Read Only)
This bit indicates the state of the SPI Flash Memory interface.
When this bit = 0b, the SPI Flash Memory interface is not busy.
When this bit = 1b, the SPI Flash Memory interface is busy.

bit 2 SPI Flash Write Data Register Empty Flag (Read Only)
This bit indicates when the SPI Flash Write Data register is empty which occurs when data written to the register is latched for serialization/transmission.
When this bit = 0b, the SPI Flash Write Data register is not empty.
When this bit = 1b, the SPI Flash Write Data register is empty. (default)

To clear this flag, write data to the SPI Flash Write Data register, REG[0B02h].

bit 1 SPI Flash Read Data Overrun Flag (Read Only)
This bit indicates when new data is loaded into the SPI Flash Read Data register before the existing data has been read (REG[0B06h] bit 0 = 1b while the new data is loaded). In this this case, the old data is no longer available and must be re-read.
When this bit = 0b, a SPI Flash Read Data overrun has not occurred.
When this bit = 1b, an SPI Flash Read Data overrun has occurred.

To clear this flag, read the SPI Flash Read Data register, REG[0B00h].

bit 0 — SPI Flash Read Data Ready Flag (Read Only)
This bit indicates when read data from the SPI Flash Memory is available (or ready) in the SPI Flash Read Data register, REG[0B00h].
When this bit = 0b, SPI Flash Memory read data is not ready.
When this bit = 1b, SPI Flash Memory read data is ready.

To clear this flag, read the SPI Flash Read Data register, REG[0B00h].

REG[0B0Ah] SPI Flash Chip Select Control Register Default = 00h							Read/Write
n/a							SPI Flash Chip Select Enable
7	6	5	4	3	2	1	0

bit 0 — SPI Flash Chip Select Enable
This bit only has an effect when the SPI Flash Read Mode bit is set to 0b, REG[0B04h] bit 7 = 0b. This bit controls chip select (SPICS pin) for the SPI Flash Memory interface.
When this bit = 0b, chip select is disabled.
When this bit = 1b, chip select is enabled.

Note

The chip select output pin for the Serial Flash Memory interface is active low. Therefore, SPICS is high when this bit = 0b, and SPICS is low when this bit = 1b.

10.4.17 Cache Control Register

REG[0C00h] C33 Instruction Cache Control Register Default = 00h							Read/Write
n/a						Reserved	C33 Instruction Cache Enable
7	6	5	4	3	2	1	0

bit 1 Reserved
This bit must be set to 0b.

bit 0 C33 Instruction Cache Enable
This bit controls the C33 Instruction Cache. The C33 Instruction Cache can be enabled only when the C33 is enabled (REG[001Ch] bit 6 = 1b) and not in a reset state (REG[001Dh] bit 0 ≠ 1b). This bit can be read to determine whether the cache has been enabled or disabled after writing to this bit, however, enable/disable sequencing logic delays when the actual state of the cache is reflected in this bit. If this bit is written while the C33 is not running, the read back value of this bit will not change.
When this bit = 0b, the C33 Instruction Cache is disabled.
When this bit = 1b, the C33 Instruction Cache is enabled.

10.4.18 Camera Interface Registers

REG[0D00h] Camera1 Enable Register Default = 00h							Read/Write
Camera1 Software Reset (WO)	n/a				Reserved	Reserved	Camera1 Interface Enable
7	6	5	4	3	2	1	0

bit 7 Camera1 Software Reset (Write Only)
This bit performs a software reset of the Camera1 logic and resets the Camera1 registers (REG[0D00h] ~ REG[0D35h]) to their default values.
Writing a 0b to this bit has no effect.
Writing a 1b to this bit initiates a Camera1 software reset.

bit 2 Reserved
This bit must be set to 0b.

bit 1 Reserved
This bit must be set to 0b.

bit 0 Camera1 Interface Enable
This bit enables the Camera1 interface logic.
When this bit = 0b, the Camera1 interface is disabled.
When this bit = 1b, the Camera1 interface is enabled.

REG[0D02h] Camera1 Clock Configuration Register Default = 00h							Read/Write
Camera1 Clock Output Disable	Camera1 Clock Divide Select bits 4-0					Reserved	Camera1 Clock Polarity
7	6	5	4	3	2	1	0

bit 7 Camera1 Clock Output Disable
This bit controls the Camera1 clock (CM1CLKOUT).
When this bit = 0b, the Camera1 clock is enabled.
When this bit = 1b, the Camera1 clock is disabled.

Note
For SPI 2 Stream Mode, (see Section 5.4, "Configuration Pins" on page 33) when the Camera1 Interface is configured for RGB stream input mode, REG[0D02h] bit 7 should be set to 1b.

bits 6-2 Camera1 Clock Divide Select bits [4:0]
These bits specify the divide ratio used to generate the Camera1 Clock Output (CM1CLKOUT). The source of the clock is the system clock and the divide ratio is programmable using the following formula.
Camera1 Clock Divide Ratio = (REG[0D02h] bits 6-2) + 1

bit 1 Reserved
This bit must be set to 0b.

bit 0 Camera1 Clock Polarity
This bit selects the Camera1 input clock (CM1CLKIN) polarity.
When this bit = 0b, the Camera1 input signals are latched on the rising edge of the CM1CLKIN signal. (default)
When this bit = 1b, the Camera1 input signals are latched on the falling edge of the CM1CLKIN signal.

REG[0D04h] Camera1 Signal Polarity Register Default = 00h							Read/Write
n/a				Reserved	CM1VREF Polarity	CM1HREF Polarity	CM1DATEN Polarity
7	6	5	4	3	2	1	0

bit 3 Reserved
This bit must be set to 0b.

bit 2 CM1VREF Polarity
This bit selects the VSYNC signal polarity for Camera1.
When this bit = 0b, the CM1VREF signal is active low. (default)
When this bit = 1b, the CM1VREF signal is active high.

bit 1 CM1HREF Polarity
This bit selects the HSYNC signal polarity for Camera1.
When this bit = 0b, the CM1HREF signal is active low. (default)
When this bit = 1b, the CM1HREF signal is active high.

bit 0 CM1DATEN Polarity
This bit only has an effect when the Camera1 Use Data Enable bit is set to 1b, REG[0D06h] bit 7 = 1b. This bit selects the data enable signal polarity for Camera1.
When this bit = 0b, the Camera1 data enable is active high. (default)
When this bit = 1b, the Camera1 data enable is active low.

REG[0D06h] Camera1 Configuration Register 0 Default = 00h							Read/Write
Camera1 ITU-R BT.656 Enable	n/a	Camera1 YUV Offset Enable	Camera1 YUV Data Format bits 1-0		Camera1 Interface Mode bits 1-0		n/a
7	6	5	4	3	2	1	0

bit 7 Camera1 ITU-R BT.656 Enable
This bit controls the camera interface type for Camera1.
When this bit = 0b, ITU-R BT.656 mode is disabled (normal camera). In this mode the hsync, vsync, clock, and data signals are independent input signals. (default)
When this bit = 1b, ITU-R BT.656 mode is enabled. In this mode the hsync and vsync signal information is embedded in the data signals and the CM1VREF and CM1HREF input pins are ignored.

Note
When ITU-R BT656 mode is enabled (REG[0D06h] bit 7 = 1b), REG[0D32h] ~ REG[0D35h] have no effect and are ignored.

bit 5 Camera1 YUV Offset Enable
This bit controls whether a UV offset is applied to the incoming Camera1 data and must be configured based on the YUV data type of the camera (see also REG[0D1Eh] bit 4).

Table 10-54 : Camera1 YUV Offset Selection

REG[0D06h] bit 5	YUV Data Type	Data Range 1 (REG[0D1Eh] bit 4 = 0b)	Data Range 2 (REG[0D1Eh] bit 4 = 1b)
0b	Straight Binary	$0 \le U \le 255$ $0 \le V \le 255$	$16 \le Cb \le 240$ $16 \le Cr \le 240$
1b	Offset Binary	$-128 \le U \le 127$ $-128 \le V \le 127$	$-112 \le Cb \le 112$ $-112 \le Cr \le 112$

bits 4-3 Camera1 YUV Data Format bits [1:0]
When the Camera1 interface mode is set for 8-bit YUV 4:2:2 (REG[0D06h] bits 2-1 = 00b), these bits select the YUV data sequence order format for Camera1.

Table 10-55: Camera1 YUV Data Format Selection

REG[0D06h] bits 4-3	8-bit YUV Data Format
00b (default)	(1st) UYVY (last)
01b	(1st) VYUY (last)
10b	(1st) YUYV (last)
11b	(1st) YVYU (last)

bits 2-1 Camera1 Interface Mode bits [1:0]
These bits select the interface mode for Camera1.

Table 10-56: Camera1 Interface Mode Selection

REG[0D06h] bits 2-1	Camera Interface Mode
00b (default)	8-bit YUV 4:2:2
01b	Reserved
10b	24-bit RGB 8:8:8
11b	Reserved

Note

For SPI 2 Stream Mode, (see Section 5.4, “Configuration Pins” on page 33) when the Camera1 Interface is configured for RGB stream input mode, REG[0D02h] bit 7 should be set to 1b.

REG[0D07h] Camera1 Configuration Register 1
Default = 00h — Read/Write

n/a							Camera1 Use Data Enable
7	6	5	4	3	2	1	0

bit 0 Camera1 Use Data Enable
This bit controls Camera1 Data Enable which is typically used when 24-bit RGB streaming is selected, REG[0D06h] bits 2-1 = 10b. If Camera1 Data Enable is enabled, the polarity of the signal can be configured using the CM1DATEN Polarity bit, REG[0D04h] bit 0. The Camera1 signals are available on the Host Interface Pins (SPI 2-stream mode, see Section 5.4, “Configuration Pins” on page 33) when 24-bit RGB streaming is selected. For pin mapping details, see Section 5.5, “Host Interface Pin Mapping” on page 35.
When this bit = 0b, Camera1 Data Enable is not used.
When this bit = 0b, Camera1 Data Enable is used.

REG[0D08h] Camera1 Input Frame Control Register Default = 00h							Read/Write
n/a	Camera1 Frame Capture Start/Stop	Camera1 Frame Event Select	Camera1 Frame Event Enable	Camera1 Frame Event Control	Reserved		
7	6	5	4	3	2	1	0

bit 6 Camera1 Frame Capture Start/Stop
This bit is used to start or stop frame capturing for Camera1.
When this bit = 0b, Camera1 frame capturing is stopped after the current frame.
When this bit = 1b, Camera1 frame capturing is started on the next frame.

bit 5 Camera1 Frame Event Select
This bit selects which edge of the frame causes the frame event. The frame event status is not indicated by the Camera1 Frame Event Status bit (REG[0D0Eh] bit 5) until it is triggered by the condition specified by the Camera1 Frame Event Control bit (REG[0D08h] bit 3).
When this bit = 0b, the frame event is caused by the start of a frame.
When this bit = 1b, the frame event is caused by the end of a frame.

bit 4 Camera1 Frame Event Enable
This bit controls whether the frame event can occur. If enabled, the status of the frame event is indicated by the Camera1 Frame Event Status bit (REG[0D0Eh] bit 5).
When this bit = 0b, the frame event is disabled.
When this bit = 1b, the frame event is enabled.

bit 3 Camera1 Frame Event Control
This bit determines what triggers the frame event. The frame event will occur at the next frame start/end after the trigger takes place.
When this bit = 0b, the frame event is triggered by Camera1 VSYNC.
When this bit = 1b, the frame event is triggered by a Camera1 Frame Capture Stop, REG[0D08h] bit 6 = 0b.

bits 2-0 Reserved
The default value for these bits is 000b.

REG[0D09h] Camera1 Flag Clear Register Default = 00h							Write Only
n/a					Reserved	Reserved	Camera1 Frame Event Clear
7	6	5	4	3	2	1	0

bit 2 Reserved
The default value of this bit is 0b.

bit 1 Reserved
The default value of this bit is 0b.

bit 0 Camera1 Frame Event Clear (Write Only)
This bit is used to clear the Camera1 Frame Event Status bit, REG[0D0Eh] bit 5.
Writing a 0b to this bit has no effect.
Writing a 1b to this bit clears the Camera1 Frame Event Status bit.

REG[0D0Ah] Camera1 Input Horizontal Size Register 0 Default = 00h							Read/Write
Camera1 Input Horizontal Size bits 7-0							
7	6	5	4	3	2	1	0

REG[0D0Bh] Camera1 Input Horizontal Size Register 1 Default = 00h							Read/Write
n/a					Camera1 Input Horizontal Size bits 10-8		
7	6	5	4	3	2	1	0

REG[0D0Bh] bits 2-0
REG[0D0Ah] bits 7-0 Camera1 Input Horizontal Size bits [10:0]
These bits specify the horizontal size of the Camera1 input image, in pixels. The input horizontal size is calculated as follows.

For interlaced modes (see REG[0D30h] bits 1-0) when ITU-R BT.656 mode is enabled (REG[0D06h] bit 7 = 1b):
Input horizontal size = HDP

For interlaced modes when ITU-R BT.656 mode is disabled (REG[0D06h] bit 7 = 0b):
Input horizontal size = HDP + HNDP

For progressive mode (REG[0D30h] bits 1-0 = 00b):
Input horizontal size = HDP

REG[0D0Ch] Camera1 Input Vertical Size Register 0 Default = 00h Read/Write							
Camera1 Input Vertical Size bits 7-0							
7	6	5	4	3	2	1	0

REG[0D0Dh] Camera1 Input Vertical Size Register 1 Default = 00h Read/Write							
n/a					Camera1 Input Vertical Size bits 10-8		
7	6	5	4	3	2	1	0

REG[0D0Dh] bits 2-0
REG[0D0Ch] bits 7-0 Camera1 Input Vertical Size bits [10:0]

These bits specify the vertical size of the Camera1 input image, in pixels. The input vertical size is calculated as follows.

For interlaced modes (see REG[0D30h] bits 1-0) when ITU-R BT.656 mode is enabled (REG[0D06h] bit 7 = 1b):
Input vertical size = VDP

For interlaced modes when ITU-R BT.656 mode is disabled (REG[0D06h] bit 7 = 0b):
Input vertical size = VDP + VNDP

For progressive mode (REG[0D30h] bits 1-0 = 00b):
Input vertical size = VDP

REG[0D0Eh] Camera1 Status Register Default = 0Xh Read Only							
n/a		Camera1 Frame Event Status	Camera1 Effective Capture Status	Camera1 Effective Frame Status	Camera1 Raw VSYNC Status	Reserved	Reserved
7	6	5	4	3	2	1	0

bit 5 Camera1 Frame Event Status (Read Only)

This bit indicates the status of the Camera1 Frame Event. The frame event is configured using the Camera1 Frame Event Select/Enable/Control bits (REG[0D08h] bits 5-3).
When this bit = 0b, a frame event has not occurred.
When this bit = 1b, a frame event has occurred.

To clear this bit, write a 1b to REG[0D09h] bit 0.

bit 4 Camera1 Effective Capture Status (Read Only)

The camera input interface has a programmable frame sampling rate. Frame capture occurs at the effective rate which is selected by the Camera1 Frame Sampling Select bits, REG[0D08h] bits 2-0. This bit indicates if the Camera1 input interface is capturing a frame whether the frame is valid or not.
When this bit = 0b, a frame is not being captured.
When this bit = 1b, a frame is being captured.

bit 3 Camera1 Effective Frame Status (Read Only)
This bit indicates if the Camera1 input interface is capturing a valid frame.
When this bit = 0b, a frame is not being captured.
When this bit = 1b, a valid frame has been captured.

bit 2 Camera1 Raw VSYNC Status (Read Only)
This bit indicates the current state of the CM1VREF input pin. The polarity of this pin is controlled by the CM1VREF Polarity bit, REG[0D04h] bit 2.

When REG[0D04h] bit 2 = 0b:
When this bit = 0b, the CM1VREF input is low.
When this bit = 1b, the CM1VREF input is high.

When REG[0D04h] bit 2 = 1b:
When this bit = 0b, the CM1VREF input is high.
When this bit = 1b, the CM1VREF input is low.

bit 1 Reserved
The default value of this bit is 0b.

bit 0 Reserved
The default value of this bit is 0b.

REG[0D0Fh] is Reserved

This register is Reserved and should not be written.

REG[0D10h] Camera1 Resizer X Start Position Register 0 Default = 00h							Read/Write
Camera1 Resizer X Start Position bits 7-0							
7	6	5	4	3	2	1	0

REG[0D11h] Camera1 Resizer X Start Position Register 1 Default = 00h							Read/Write
n/a					Camera1 Resizer X Start Position bits 10-8		
7	6	5	4	3	2	1	0

REG[0D11h] bits 2-0
REG[0D10h] bits 7-0 Camera1 Resizer X Start Position bits [10:0]
These bits specify the Camera1 resizer horizontal (X) start position, relative to the top left corner of the camera input image, in pixels. The resizer is used for cropping and/or defining the area of the camera image that will be down-scaled (see REG[0D18h] ~ REG[0D1Ah]).

REG[0D12h] Camera1 Resizer Y Start Position Register 0 Default = 00h							Read/Write
Camera1 Resizer Y Start Position bits 7-0							
7	6	5	4	3	2	1	0

REG[0D13h] Camera1 Resizer Y Start Position Register 1 Default = 00h							Read/Write
n/a					Camera1 Resizer Y Start Position bits 10-8		
7	6	5	4	3	2	1	0

REG[0D13h] bits 2-0
REG[0D12h] bits 7-0 Camera1 Resizer Y Start Position bits [10:0]
These bits specify the Camera1 resizer vertical (Y) start position, relative to the top left corner of the camera input image, in pixels. The resizer is used for cropping the camera input image and/or defining the area of the camera image that will be down-scaled (see REG[0D18h] ~ REG[0D1Ah]).

REG[0D14h] Camera1 Resizer X End Position Register 0 Default = 00h							Read/Write
Camera1 Resizer X End Position bits 7-0							
7	6	5	4	3	2	1	0

REG[0D15h] Camera1 Resizer X End Position Register 1 Default = 00h							Read/Write
n/a					Camera1 Resizer X End Position bits 10-8		
7	6	5	4	3	2	1	0

REG[0D15h] bits 2-0
REG[0D14h] bits 7-0 Camera1 Resizer X End Position bits [10:0]
These bits specify the Camera1 resizer horizontal (X) end position, relative to the top left corner of the camera input image, in pixels. The resizer is used for cropping the camera input image and/or defining the area of the camera image that will be down-scaled (see REG[0D18h] ~ REG[0D1Ah]).

REG[0D16h] Camera1 Resizer Y End Position Register 0 Default = 00h Read/Write							
Camera1 Resizer Y End Position bits 7-0							
7	6	5	4	3	2	1	0

REG[0D17h] Camera1 Resizer Y End Position Register 1 Default = 00h Read/Write							
n/a					Camera1 Resizer Y End Position bits 10-8		
7	6	5	4	3	2	1	0

REG[0D17h] bits 2-0
REG[0D16h] bits 7-0 Camera1 Resizer Y End Position bits [10:0]
These bits specify the Camera1 resizer vertical (Y) end position, relative to the top left corner of the camera input image, in pixels. The resizer is used for cropping the camera input image and/or defining the area of the camera image that will be down-scaled (see REG[0D18h] ~ REG[0D1Ah]).

REG[0D18h] Camera1 Resizer Horizontal Scaling Rate Register Default = 00h Read/Write							
Camera1 Resizer Horizontal Scaling Rate bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 Camera1 Resizer Horizontal Scaling Rate bits [7:0]
The Camera1 resizer supports down-scaling (reduction) of the camera input image. These bits specify the horizontal scaling rate for the Camera1 resizer according to the following formula.

Camera1 horizontal scaling rate = REG[0D18h] bits 7-0 ÷ 128

REG[0D19h] Camera1 Resizer Vertical Scaling Rate Register Default = 00h Read/Write							
Camera1 Resizer Vertical Scaling Rate bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 Camera1 Resizer Vertical Scaling Rate bits [7:0]
The Camera1 resizer supports down-scaling (reduction) of the camera input image. These bits specify the vertical scaling rate for the Camera1 resizer according to the following formula.

Camera1 vertical scaling rate = REG[0D19h] bits 7-0 ÷ 128

REG[0D1Ah] Camera1 Resizer Scaling Control Register Default = 00h							Read/Write
n/a						Camera1 Resizer Scaling Mode bits 1-0	
7	6	5	4	3	2	1	0

bits 1-0 — Camera1 Resizer Scaling Mode bits [1:0]
These bits determine the Camera1 resizer scaling mode. Before selecting a scaling mode, set the horizontal (REG[0D18h]) and/or vertical (REG[0D19h]) scaling rates.

Table 10-57: Camera1 Resizer Scaling Mode Selection

REG[0D1Ah] bits 1-0	Resizer Scaling Mode
00b	no scaling
01b	V/H reduction
10b	V: Reduction, H: Average
11b	Reserved

REG[0D1Ch] is Reserved

This register is Reserved and should not be written.

REG[0D1Eh] Camera1 YRC Control Register 0 Default = 00h							Read/Write
n/a	Camera1 YRC RGB Pixel Output Format bits 1-0		Camera1 YRC YUV Input Data Type	Camera1 YRC YUV Transfer Mode bits 2-0			Camera1 YRC Bypass Enable
7	6	5	4	3	2	1	0

bits 6-5 — Camera1 YRC RGB Pixel Output Format bits [1:0]
These bits specify the RGB pixel format output by the Camera1 YRC (YUV to RGB Converter). The output from the Camera1 YRC goes to the Camera1 Writer which writes the image data to external SDRAM. For further information on the Camera1 Writer, see Section 22.6, “Camera Writer” on page 547.

Table 10-58: RGB Pixel Format Selection

REG[0D1Eh] bits 6-5	RGB Pixel Format
00b	RGB 3:3:2
01b	RGB 5:6:5
10b	RGB 8:8:8
11b	Reserved

bit 4 — Camera1 YRC YUV Input Data Type
This bit selects the input data type for the Camera1 YRC (YUV to RGB Converter).
When this bit = 0b, the input data type is YUV
(0 <= Y <= 255, 0 <= U <= 255, 0<= V <= 255).
When this bit = 1b, the input data type is YCbCr
(16 <= Y <= 235, 16 <= U <= 240, 16 <= V <= 240).

bits 3-1 Camera1 YRC YUV Transfer Mode bits [2:0]
These bits specify the transfer mode used by the Camera1 YRC (YUV to RGB Converter). Recommended settings are provided for various specifications.

Table 10-59: YUV Transfer Mode Selection

REG[0D1Eh] bits 3-1	YUV Transfer Mode
000b	Reserved
001b	Recommended for ITU-R BT.709
010b	Reserved
011b	Reserved
100b	Recommended for ITU-R BT.470-6 System M
101b	Recommend for ITU-R BT.470-6 System B, G
110b	SMPTE 170M
111b	SMPTE 240M (1987)

bit 0 Camera1 YRC Bypass Enable
This bit determines whether YUV to RGB conversion for Camera1 takes place. Typically, the Camera1 YRC is bypassed when using 24-bit RGB input, REG[0D06h] bits 2-1 = 10b. When this bit = 0b, the Camera1 YRC is enabled (YUV to RGB conversion takes place). When this bit = 1b, the Camera1 YRC is bypassed (YUV to RGB conversion does not take place).

REG[0D1Fh] Camera1 YRC Control Register 1 Default = 00h							Read/Write
n/a						Camera1 YRC UV Fixed Data Select bits 1-0	
7	6	5	4	3	2	1	0

bits 1-0 Camera1 YRC UV Fixed Data Select bits [1:0]
These bits control the UV input to the Camera1 YRC (YUV to RGB Converter) by allowing the U data, V data, or both, to be “fixed” to the value specified by the Camera1 YRC U Fixed Data (REG[0D20h]) and Camera1 YRC V Fixed Data (REG[0D21h]) registers. These bits have an effect on the UV data even when the Camera1 YRC is bypassed, REG[0D1Eh] bit 0 = 1b.

Table 10-60: Camera1 YRC UV Fixed Data Selection

REG[0D1Fh] bits 1-0	UV Data Input to the YRC
00b	Original U data, Original V data
01b	U data = REG[0D20h] bits 7-0, Original V data
10b	Original U data, V data = REG[0D21h] bits 7-0
11b	U data = REG[0D20h] bits 7-0, V data = REG[0D21h] bits 7-0

REG[0D20h] Camera1 YRC U Fixed Data Register Default = 00h							Read/Write
Camera1 YRC U Fixed Data bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 Camera1 YRC U Fixed Data bits [7:0]
These bits only have an effect when the Camera1 YRC UV Fixed Data Select bits are set to 01b or 11b (REG[0D1Fh] bits 1-0 = 01b or 11b). The U data input to the Camera1 YRC (YUV to RGB Converter) is fixed to the value of these bits.

REG[0D21h] Camera1 YRC V Fixed Data Register Default = 00h							Read/Write
Camera1 YRC V Fixed Data bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 Camera1 YRC V Fixed Data bits [7:0]
These bits only have an effect when the Camera1 YRC UV Fixed Data Select bits are set to 10b or 11b (REG[0D1Fh] bits 1-0 = 10b or 11b). The V data input to the Camera1 YRC (YUV to RGB Converter) is fixed to the value of these bits.

REG[0D22h] is Reserved

This register is Reserved and should not be written.

REG[0D24h] Camera1 YRC X Size Register 0 Default = 00h							Read/Write
Camera1 YRC X Size bits 7-0							
7	6	5	4	3	2	1	0

REG[0D25h] Camera1 YRC X Size Register 1 Default = 00h							Read/Write
n/a					Camera1 YRC X Size bits 10-8		
7	6	5	4	3	2	1	0

REG[0D25h] bits 2-0
REG[0D24h] bits 7-0 Camera1 YRC X Size bits [10:0]

These bits specify the horizontal (X) size of the Camera1 YRC, in pixels.

X Size = INT((Resizer X End - Resizer X Start + 1) x Resizer X Scaling Rate ÷ 128)
= INT (((REG[0D15h],REG[0D14h]) - (REG[0D11h],REG[0D10h]) + 1) x REG[0D18h] ÷ 128)

Note

The Camera1 YRC X Size must be set such that the X size multiplied by the pixel format (in bpp, see REG[0D1Eh] bits 6-5) is divisible by 64.

REG[0D26h] Camera1 YRC Y Size Register 0 Default = 00h							Read/Write
Camera1 YRC Y Size bits 7-0							
7	6	5	4	3	2	1	0

REG[0D27h] Camera1 YRC Y Size Register 1 Default = 00h							Read/Write
n/a					Camera1 YRC Y Size bits 10-8		
7	6	5	4	3	2	1	0

REG[0D27h] bits 2-0
REG[0D26h] bits 7-0 Camera1 YRC Y Size bits [10:0]

These bits specify the vertical (Y) size of the Camera1 YRC, in pixels.

Y Size = INT((Resizer Y End - Resizer Y Start + 1) x Resizer Y Scaling Rate ÷ 128)
= INT (((REG[0D17h],REG[0D16h]) - (REG[0D13h],REG[0D12h]) + 1) x REG[0D19h] ÷ 128)

REG[0D28h] is Reserved

This register is Reserved and should not be written.

REG[0D30h] Camera1 Video Mode Register							
Default = 00h							Read/Write
n/a				Camera1 Write Field Select bits 1-0		Camera1 Video Mode Select bits 1-0	
7	6	5	4	3	2	1	0

bits 3-2 Camera1 Write Field Select bits [1:0]
These bits select which video write fields are written to memory.

Table 10-61: Camera1 Write Field Selection

REG[0D30h] bits 3-2	Write Field Selection
00b	Both Odd and Even Fields are written
01b	Only Odd Field is written
10b	Only Even Field is written
11b	Reserved

bits 1-0 Camera1 Video Mode Select bits [1:0]
These bits select the video mode for the Camera1 interface.

Table 10-62: Camera1 Video Mode Selection

REG[0D30h] bits 1-0	Video Mode Selection
00b	Progressive (Field is not used)
01b	Reserved
10b	Interlaced (HSYNC and Field are used)
11b	Interlaced (HSYNC and VSYNC are used)

REG[0D32h] Camera1 Odd Field Offset Register 0							
Default = 00h							Read/Write
Camera1 Odd Field Offset bits 7-0							
7	6	5	4	3	2	1	0

REG[0D33h] Camera1 Odd Field Offset Register 1							
Default = 00h							Read/Write
n/a					Camera1 Odd Field Offset bits 10-8		
7	6	5	4	3	2	1	0

REG[0D33h] bits 2-0
REG[0D32h] bits 7-0 Camera1 Odd Field Offset bits [10:0]
When REG[0D30h] bits 1-0 = 10b or 11b, these bits specify the odd field offset.

REG[0D34h] Camera1 Even Field Offset Register 0 Default = 00h							Read/Write
Camera1 Even Field Offset bits 7-0							
7	6	5	4	3	2	1	0

REG[0D35h] Camera1 Even Field Offset Register 1 Default = 00h							Read/Write
n/a					Camera1 Even Field Offset bits 10-8		
7	6	5	4	3	2	1	0

REG[0D35h] bits 2-0
REG[0D34h] bits 7-0 Camera1 Even Field Offset bits [10:0]
When REG[0D30h] bits 1-0 = 10b or 11b, these bits specify the even field offset.

REG[0D40h] Camera2 Enable Register Default = 00h							Read/Write
Camera2 Software Reset (WO)	n/a				Reserved	Reserved	Camera2 Interface Enable
7	6	5	4	3	2	1	0

bit 7 Camera2 Software Reset (Write Only)
This bit performs a software reset of the Camera2 logic and resets the Camera2 registers (REG[0D40h] ~ REG[0D75h]) to their default values.
Writing a 0b to this bit has no effect.
Writing a 1b to this bit initiates a Camera2 software reset.

bit 2 Reserved
This bit must be set to 0b.

bit 1 Reserved
This bit must be set to 0b.

bit 0 Camera2 Interface Enable
This bit enables the Camera2 interface logic.
When this bit = 0b, the Camera2 interface is disabled.
When this bit = 1b, the Camera2 interface is enabled.

REG[0D42h] Camera2 Clock Configuration Register Default = 00h							Read/Write
Camera2 Clock Output Disable	Camera2 Clock Divide Select bits 4-0					Reserved	Camera2 Clock Polarity
7	6	5	4	3	2	1	0

bit 7 Camera2 Clock Output Disable
This bit controls the Camera2 clock (CM2CLKOUT).
When this bit = 0b, the Camera2 clock is enabled.
When this bit = 1b, the Camera2 clock is disabled.

bits 6-2 Camera2 Clock Divide Select bits [4:0]
These bits specify the divide ratio used to generate the Camera2 Clock Output (CM2CLKOUT). The source of the clock is the system clock and the divide ratio is programmable using the following formula.

Camera2 Clock Divide Ratio = (REG[0D42h] bits 6-2) + 1

bit 1 Reserved
This bit must be set to 0b.

bit 0 Camera1 Clock Polarity
This bit selects the Camera2 input clock (CM2CLKIN) polarity.
When this bit = 0b, the Camera2 input signals are latched on the rising edge of the CM2CLKIN signal. (default)
When this bit = 1b, the Camera2 input signals are latched on the falling edge of the CM2CLKIN signal.

REG[0D44h] Camera2 Signal Polarity Register Default = 00h							Read/Write
n/a				Reserved	CM2VREF Polarity	CM2HREF Polarity	CM2DATEN Polarity
7	6	5	4	3	2	1	0

bit 3 Reserved
This bit must be set to 0b.

bit 2 CM2VREF Polarity
This bit selects the VSYNC signal polarity for Camera2.
When this bit = 0b, the CM2VREF signal is active low. (default)
When this bit = 1b, the CM2VREF signal is active high.

bit 1 CM2HREF Polarity
This bit selects the HSYNC signal polarity for Camera2.
When this bit = 0b, the CM2HREF signal is active low. (default)
When this bit = 1b, the CM2HREF signal is active high.

bit 0 CM2DATEN Polarity
This bit only has an effect when the Camera2 Use Data Enable bit is set to 1b, REG[0D46h] bit 7 = 1b.This bit selects the data enable signal polarity for Camera2.
When this bit = 0b, the Camera2 data enable is active high. (default)
When this bit = 1b, the Camera2 data enable is active low.

REG[0D46h] Camera2 Configuration Register 0 Default = 04h							Read/Write
Camera2 ITU-R BT.656 Enable	n/a	Camera2 YUV Offset Enable	Camera2 YUV Data Format bits 1-0		Camera2 Interface Mode bits 1-0		n/a
7	6	5	4	3	2	1	0

bit 7 Camera2 ITU-R BT.656 Enable
This bit controls the camera interface type for Camera2.
When this bit = 0b, ITU-R BT.656 mode is disabled (normal camera). In this mode the hsync, vsync, clock, and data signals are independent input signals. (default)
When this bit = 1b, ITU-R BT.656 mode is enabled. In this mode the hsync and vsync signal information is embedded in the data signals and the CM2VREF and CM2HREF input pins are ignored.

Note
When ITU-R BT656 mode is enabled (REG[0D46h] bit 7 = 1b), REG[0D72h] ~ REG[0D75h] have no effect and are ignored.

bit 5 Camera2 YUV Offset Enable
This bit controls whether a UV offset is applied to the incoming Camera2 data and must be configured based on the YUV data type of the camera (see also REG[0D5Eh] bit 4).

Table 10-63 : Camera2 YUV Offset Selection

REG[0D46h] bit 5	YUV Data Type	Data Range 1 (REG[0D5Eh] bit 4 = 0b)	Data Range 2 (REG[0D5Eh] bit 4 = 1b)
0b	Straight Binary	$0 \le U \le 255$ $0 \le V \le 255$	$16 \le Cb \le 240$ $16 \le Cr \le 240$
1b	Offset Binary	$-128 \le U \le 127$ $-128 \le V \le 127$	$-112 \le Cb \le 112$ $-112 \le Cr \le 112$

bits 4-3 Camera2 YUV Data Format bits [1:0]
When the Camera2 interface mode is set for 8-bit YUV 4:2:2 (REG[0D46h] bits 2-1 = 00b), these bits select the YUV data sequence order format for Camera2.

Table 10-64: Camera2 YUV Data Format Selection

REG[0D46h] bits 4-3	8-bit YUV Data Format
00b (default)	(1st) UYVY (last)
01b	(1st) VYUY (last)
10b	(1st) YUYV (last)
11b	(1st) YVYU (last)

bits 2-1 Camera2 Interface Mode bits [1:0]
These bits select the interface mode for Camera2.

Table 10-65: Camera2 Interface Mode Selection

REG[0D46h] bits 2-1	Camera Data Format
00b	8-bit YUV 4:2:2
01b	Reserved
10b (default)	24-bit RGB 8:8:8
11b	Reserved

REG[0D47h] Camera2 Configuration Register 1 Default = 00h							Read/Write
n/a							Camera2 Use Data Enable
7	6	5	4	3	2	1	0

bit 0 Camera2 Use Data Enable
This bit controls Camera2 Data Enable which is typically used when 24-bit RGB streaming is selected, REG[0D46h] bits 2-1 = 10b. If Camera2 Data Enable is enabled, the polarity of the signal can be configured using the CM2DATEN Polarity bit, REG[0D44h] bit 0. For pin mapping details, see Section 5.6, “LCD / Camera2 Pin Mapping” on page 40.
When this bit = 0b, Camera2 Data Enable is not used.
When this bit = 0b, Camera2 Data Enable is used.

REG[0D48h] Camera2 Input Frame Control Register Default = 00h							Read/Write
n/a	Camera2 Frame Capture Start/Stop	Camera2 Frame Event Select	Camera2 Frame Event Enable	Camera2 Frame Event Control	Reserved		
7	6	5	4	3	2	1	0

bit 6 Camera2 Frame Capture Start/Stop
This bit is used to start or stop frame capturing for Camera2.
When this bit = 0b, Camera2 frame capturing is stopped after the current frame.
When this bit = 1b, Camera2 frame capturing is started on the next frame.

bit 5 Camera2 Frame Event Select
This bit selects which edge of the frame causes the frame event. The frame event status is not indicated by the Camera2 Frame Event Status bit (REG[0D4Eh] bit 5) until it is triggered by the condition specified by the Camera2 Frame Event Control bit (REG[0D48h] bit 3).
When this bit = 0b, the frame event is caused by the start of a frame.
When this bit = 1b, the frame event is caused by the end of a frame.

bit 4 Camera2 Frame Event Enable
This bit controls whether the frame event can occur. If enabled, the status of the frame event is indicated by the Camera2 Frame Event Status bit (REG[0D4Eh] bit 5).
When this bit = 0b, the frame event is disabled.
When this bit = 1b, the frame event is enabled.

bit 3 Camera2 Frame Event Control
This bit determines what triggers the frame event. The frame event will occur at the next frame start/end after the trigger takes place.
When this bit = 0b, the frame event is triggered by Camera2 VSYNC.
When this bit = 1b, the frame event is triggered by a Camera2 Frame Capture Stop, REG[0D48h] bit 6 = 0b.

bits 2-0 Reserved
The default value for these bits is 000b.

REG[0D49h] Camera2 Flag Clear Register Default = 00h							Write Only
n/a					Reserved	Reserved	Camera2 Frame Event Clear
7	6	5	4	3	2	1	0

bit 2 Reserved
The default value of this bit is 0b.

bit 1 Reserved
The default value of this bit is 0b.

bit 0 Camera2 Frame Event Clear (Write Only)
This bit is used to clear the Camera2 Frame Event Status bit, REG[0D4Eh] bit 5.
Writing a 0b to this bit has no effect.
Writing a 1b to this bit clears the Camera2 Frame Event Status bit.

REG[0D4Ah] Camera2 Input Horizontal Size Register 0 Default = 00h							Read/Write
Camera2 Input Horizontal Size bits 7-0							
7	6	5	4	3	2	1	0

REG[0D4Bh] Camera2 Input Horizontal Size Register 1 Default = 00h							Read/Write
n/a					Camera2 Input Horizontal Size bits 10-8		
7	6	5	4	3	2	1	0

REG[0D4Bh] bits 2-0
REG[0D4Ah] bits 7-0 Camera2 Input Horizontal Size bits [10:0]

These bits specify the horizontal size of the Camera2 input image, in pixels. The input horizontal size is calculated as follows.

For interlaced modes (see REG[0D70h] bits 1-0) when ITU-R BT.656 mode is enabled (REG[0D46h] bit 7 = 1b):
Input horizontal size = HDP

For interlaced modes when ITU-R BT.656 mode is disabled (REG[0D46h] bit 7 = 0b):
Input horizontal size = HDP + HNDP

For progressive mode (REG[0D70h] bits 1-0 = 00b):
Input horizontal size = HDP

REG[0D4Ch] Camera2 Input Vertical Size Register 0 Default = 00h							Read/Write
Camera2 Input Vertical Size bits 7-0							
7	6	5	4	3	2	1	0

REG[0D4Dh] Camera2 Input Vertical Size Register 1 Default = 00h							Read/Write
n/a					Camera2 Input Vertical Size bits 10-8		
7	6	5	4	3	2	1	0

REG[0D4Dh] bits 2-0
REG[0D4Ch] bits 7-0 Camera2 Input Vertical Size bits [10:0]

These bits specify the vertical size of the Camera2 input image, in pixels. The input vertical size is calculated as follows.

For interlaced modes (see REG[0D70h] bits 1-0) when ITU-R BT.656 mode is enabled (REG[0D46h] bit 7 = 1b):
Input vertical size = VDP

For interlaced modes when ITU-R BT.656 mode is disabled (REG[0D46h] bit 7 = 0b):
Input vertical size = VDP + VNDP

For progressive mode (REG[0D70h] bits 1-0 = 00b):
Input vertical size = VDP

REG[0D4Eh] Camera2 Status Register 0 Default = 0Xh							Read Only
n/a		Camera2 Frame Event Status	Camera2 Effective Capture Status	Camera2 Effective Frame Status	Camera2 Camera Raw VSYNC Status	Reserved	Reserved
7	6	5	4	3	2	1	0

bit 5 Camera2 Frame Event Status (Read Only)

This bit indicates the status of the Camera2 Frame Event. The frame event is configured using the Camera2 Frame Event Select/Enable/Control bits (REG[0D48h] bits 5-3).
When this bit = 0b, a frame event has not occurred.
When this bit = 1b, a frame event has occurred.

To clear this bit, write a 1b to REG[0D49h] bit 0.

bit 4 Camera2 Effective Capture Status (Read Only)

The camera input interface has a programmable frame sampling rate. Frame capture occurs at the effective rate which is selected by the Camera2 Frame Sampling Select bits, REG[0D48h] bits 2-0. This bit indicates if the Camera2 input interface is capturing a frame whether the frame is valid or not.
When this bit = 0b, a frame is not being captured.
When this bit = 1b, a frame is being captured.

bit 3 Camera2 Effective Frame Status (Read Only)
This bit indicates if the Camera2 input interface is capturing a valid frame.
When this bit = 0b, a frame is not being captured.
When this bit = 1b, a valid frame has been captured.

bit 2 Camera2 Camera Raw VSYNC Status
This bit indicates the current state of the CM2VREF input pin. The polarity of this pin is controlled by the CM2VREF Polarity bit, REG[0D44h] bit 2.

When REG[0D44h] bit 2 = 0b:
When this bit = 0b, the CM2VREF input is low.
When this bit = 1b, the CM2VREF input is high.

When REG[0D44h] bit 2 = 1b:
When this bit = 0b, the CM2VREF input is high.
When this bit = 1b, the CM2VREF input is low.

bit 1 Reserved
The default value of this bit is 0b.

bit 0 Reserved
The default value of this bit is 0b.

REG[0D4Fh] is Reserved

This register is Reserved and should not be written.

REG[0D50h] Camera2 Resizer X Start Position Register 0 Default = 00h							Read/Write
Camera2 Resizer X Start Position bits 7-0							
7	6	5	4	3	2	1	0

REG[0D51h] Camera2 Resizer X Start Position Register 1 Default = 00h							Read/Write
n/a					Camera2 Resizer X Start Position bits 10-8		
7	6	5	4	3	2	1	0

REG[0D51h] bits 2-0
REG[0D50h] bits 7-0 Camera2 Resizer X Start Position bits [10:0]
These bits specify the Camera2 resizer horizontal (X) start position, relative to the top left corner of the camera input image, in pixels. The resizer is used for cropping and/or defining the area of the camera image that will be down-scaled (see REG[0D58h] ~ REG[0D5Ah]).

REG[0D52h] Camera2 Resizer Y Start Position Register 0 Default = 00h							Read/Write
Camera2 Resizer Y Start Position bits 7-0							
7	6	5	4	3	2	1	0

REG[0D53h] Camera2 Resizer Y Start Position Register 1 Default = 00h							Read/Write
n/a					Camera2 Resizer Y Start Position bits 10-8		
7	6	5	4	3	2	1	0

REG[0D53h] bits 2-0
REG[0D52h] bits 7-0 Camera2 Resizer Y Start Position bits [10:0]
These bits specify the Camera2 resizer vertical (Y) start position, relative to the top left corner of the camera input image, in pixels. The resizer is used for cropping the camera input image and/or defining the area of the camera image that will be down-scaled (see REG[0D58h] ~ REG[0D5Ah]).

REG[0D54h] Camera2 Resizer X End Position Register 0 Default = 00h							Read/Write
Camera2 Resizer X End Position bits 7-0							
7	6	5	4	3	2	1	0

REG[0D55h] Camera2 Resizer X End Position Register 1 Default = 00h							Read/Write
n/a					Camera2 Resizer X End Position bits 10-8		
7	6	5	4	3	2	1	0

REG[0D55h] bits 2-0
REG[0D54h] bits 7-0 Camera2 Resizer X End Position bits [10:0]
These bits specify the Camera2 resizer horizontal (X) end position, relative to the top left corner of the camera input image, in pixels. The resizer is used for cropping the camera input image and/or defining the area of the camera image that will be down-scaled (see REG[0D58h] ~ REG[0D5Ah]).

REG[0D56h] Camera2 Resizer Y End Position Register 0 Default = 00h							Read/Write
Camera2 Resizer Y End Position bits 7-0							
7	6	5	4	3	2	1	0

REG[0D57h] Camera2 Resizer Y End Position Register 1 Default = 00h							Read/Write
n/a					Camera2 Resizer Y End Position bits 10-8		
7	6	5	4	3	2	1	0

REG[0D57h] bits 2-0
REG[0D56h] bits 7-0 Camera2 Resizer Y End Position bits [10:0]
These bits specify the Camera2 resizer vertical (Y) end position, relative to the top left corner of the camera input image, in pixels. The resizer is used for cropping the camera input image and/or defining the area of the camera image that will be down-scaled (see REG[0D58h] ~ REG[0D5Ah]).

REG[0D58h] Camera2 Resizer Horizontal Scaling Rate Register Default = 00h							Read/Write
Camera2 Resizer Horizontal Scaling Rate bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 Camera2 Resizer Horizontal Scaling Rate bits [7:0]
The Camera2 resizer supports down-scaling (reduction) of the camera input image. These bits specify the horizontal scaling rate for the Camera2 resizer according to the following formula.

Camera2 horizontal scaling rate = REG[0D58h] bits 7-0 ÷ 128

REG[0D59h] Camera2 Resizer Vertical Scaling Rate Register Default = 00h							Read/Write
Camera2 Resizer Vertical Scaling Rate bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 Camera2 Resizer Vertical Scaling Rate bits [7:0]
The Camera2 resizer supports down-scaling (reduction) of the camera input image. These bits specify the vertical scaling rate for the Camera2 resizer according to the following formula.

Camera2 vertical scaling rate = REG[0D59h] bits 7-0 ÷ 128

REG[0D5Ah] Camera2 Resizer Scaling Control Register Default = 00h							Read/Write
n/a						Camera2 Resizer Scaling Mode bits 1-0	
7	6	5	4	3	2	1	0

bits 1-0 Camera2 Resizer Scaling Mode bits [1:0]
These bits determine the Camera2 resizer scaling mode. Before selecting a scaling mode, set the horizontal (REG[0D58h]) and/or vertical (REG[0D59h]) scaling rates.

Table 10-66: Camera2 Resizer Scaling Mode Selection

REG[0D5Ah] bits 1-0	Resizer Scaling Mode
00b	no scaling
01b	V/H reduction
10b	V: Reduction, H: Average
11b	Reserved

REG[0D5Ch] is Reserved

This register is Reserved and should not be written.

REG[0D5Eh] Camera2 YRC Control Register 0
Default = 00h — Read/Write

n/a	Camera2 YRC RGB Pixel Output Format bits 1-0		Camera2 YRC YUV Input Data Type	Camera2 YRC YUV Transfer Mode bits 2-0			Camera2 YRC Bypass Enable
7	6	5	4	3	2	1	0

bits 6-5 — Camera2 YRC RGB Pixel Output Format [1:0]
These bits specify the RGB pixel format output by the Camera2 YRC (YUV to RGB Converter). The output from the Camera2 YRC goes to the Camera2 Writer which writes the image data to external SDRAM. For further information on the Camera2 Writer, see Section 22.6, “Camera Writer” on page 547.

Table 10-67: RGB Pixel Format Selection

REG[0D5Eh] bits 6-5	RGB Pixel Format
00b	RGB 3:3:2
01b	RGB 5:6:5
10b	RGB 8:8:8
11b	Reserved

bit 4 — Camera2 YRC YUV Input Data Type
This bit selects the input data type for the Camera2 YRC (YUV to RGB Converter).
When this bit = 0b, the input data type is YUV
(0 <= Y <= 255, 0 <= U <= 255, 0<= V <= 255).
When this bit = 1b, the input data type is YCbCr
(16 <= Y <= 235, 16 <= U <= 240, 16 <= V <= 240).

bits 3-1 — Camera2 YRC YUV Transfer Mode bits [2:0]
These bits specify the transfer mode used by the Camera2 YRC (YUV to RGB Converter). Recommended settings are provided for various specifications.

Table 10-68: YUV Transfer Mode Selection

REG[0D5Eh] bits 3-1	YUV Transfer Mode
000b	Reserved
001b	Recommended for ITU-R BT.709
010b	Reserved
011b	Reserved
100b	Recommended for ITU-R BT.470-6 System M
101b	Recommend for ITU-R BT.470-6 System B, G
110b	SMPTE 170M
111b	SMPTE 240M (1987)

bit 0 Camera2 YRC Bypass Enable
This bit determines whether YUV to RGB conversion for Camera2 takes place. Typically, the Camera2 YRC is bypassed when using 24-bit RGB input, REG[0D46h] bits 2-1 = 10b.
When this bit = 0b, the Camera2 YRC is enabled (YUV to RGB conversion takes place).
When this bit = 1b, the Camera2 YRC is bypassed (YUV to RGB conversion does not take place).

REG[0D5Fh] Camera2 YRC Control Register 1 Default = 00h							Read/Write
n/a						Camera2 YRC UV Fixed Data Select bits 1-0	
7	6	5	4	3	2	1	0

bits 1-0 Camera2 YRC UV Fixed Data Select bits [1:0]
These bits control the UV input to the Camera2 YRC (YUV to RGB Converter) by allowing the U data, V data, or both, to be “fixed” to the value specified by the Camera2 YRC U Fixed Data (REG[0D60h]) and Camera2 YRC V Fixed Data (REG[0D61h]) registers.
These bits have an effect on the UV data even when the Camera2 YRC is bypassed, REG[0D5Eh] bit 0 = 1b.

Table 10-69: Camera2 YRC UV Fixed Data Selection

REG[0D5Fh] bits 1-0	UV Data Input to the YRC
00b	Original U data, Original V data
01b	U data = REG[0D60h] bits 7-0, Original V data
10b	Original U data, V data = REG[0D61h] bits 7-0
11b	U data = REG[0D60h] bits 7-0, V data = REG[0D61h] bits 7-0

REG[0D60h] Camera2 YRC U Fixed Data Register Default = 00h							Read/Write
Camera2 YRC U Fixed Data bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 Camera2 YRC U Fixed Data bits [7:0]
These bits only have an effect when the Camera2 YRC UV Fixed Data Select bits are set to 01b or 11b (REG[0D5Fh] bits 1-0 = 01b or 11b). The U data input to the Camera2 YRC (YUV to RGB Converter) is fixed to the value of these bits.

REG[0D61h] Camera2 YRC V Fixed Data Register Default = 00h							Read/Write
Camera2 YRC V Fixed Data bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 Camera2 YRC V Fixed Data bits [7:0]
These bits only have an effect when the Camera2 YRC UV Fixed Data Select bits are set to 10b or 11b (REG[0D51Fh] bits 1-0 = 10b or 11b). The V data input to the Camera2 YRC (YUV to RGB Converter) is fixed to the value of these bits.

REG[0D62h] is Reserved

This register is Reserved and should not be written.

REG[0D64h] Camera2 YRC X Size Register 0 Default = 00h							Read/Write
Camera2 YRC X Size bits 7-0							
7	6	5	4	3	2	1	0

REG[0D65h] Camera2 YRC X Size Register 1 Default = 00h							Read/Write
n/a					Camera2 YRC X Size bits 10-8		
7	6	5	4	3	2	1	0

REG[0D65h] bits 2-0
REG[0D64h] bits 7-0 Camera2 YRC X Size bits [10:0]

These bits specify the horizontal (X) size of the Camera2 YRC, in pixels.

X Size = INT((Resizer X End - Resizer X Start + 1) x Resizer X Scaling Rate ÷ 128)
= INT (((REG[0D55h],REG[0D54h]) - (REG[0D51h],REG[0D50h]) + 1) x REG[0D58h] ÷ 128)

Note

The Camera2 YRC X Size must be set such that the X size multiplied by the pixel format (in bpp, see REG[0D5Eh] bits 6-5) is divisible by 64.

REG[0D66h] Camera2 YRC Y Size Register 0 Default = 00h							Read/Write
Camera2 YRC Y Size bits 7-0							
7	6	5	4	3	2	1	0

REG[0D67h] Camera2 YRC Y Size Register 1 Default = 00h							Read/Write
n/a					Camera2 YRC Y Size bits 10-8		
7	6	5	4	3	2	1	0

REG[0D67h] bits 2-0
REG[0D66h] bits 7-0 Camera2 YRC Y Size bits [10:0]

These bits specify the vertical (Y) size of the Camera2 YRC, in pixels.

Y Size = INT((Resizer Y End - Resizer Y Start + 1) x Resizer Y Scaling Rate ÷ 128)
= INT (((REG[0D57h],REG[0D56h]) - (REG[0D53h],REG[0D52h]) + 1) x REG[0D59h] ÷ 128)

REG[0D68h] is Reserved

This register is Reserved and should not be written.

REG[0D70h] Camera2 Video Mode Register							
Default = 00h							Read/Write
n/a				Camera2 Write Field Select bits 1-0		Camera2 Video Mode Select bits 1-0	
7	6	5	4	3	2	1	0

bits 3-2 Camera2 Write Field Select bits [1:0]
These bits select which video write fields are written to memory.

Table 10-70: Camera2 Write Field Selection

REG[0D70h] bits 3-2	Write Field Selection
00b	Both Odd and Even Fields are written
01b	Only Odd Field is written
10b	Only Even Field is written
11b	Reserved

bits 1-0 Camera2 Video Mode Select bits [1:0]
These bits select the video mode for the Camera2 interface.

Table 10-71: Camera2 Video Mode Selection

REG[0D70h] bits 1-0	Video Mode Selection
00b	Progressive (Field is not used)
01b	Interlaced (HSYNC and VSYNC and Field are used)
10b	Interlaced (HSYNC and Field are used)
11b	Interlaced (HSYNC, VSYNC are used)

REG[0D72h] Camera2 Odd Field Offset Register 0							
Default = 00h							Read/Write
Camera2 Odd Field Offset bits 7-0							
7	6	5	4	3	2	1	0

REG[0D73h] Camera2 Odd Field Offset Register 1							
Default = 00h							Read/Write
n/a					Camera2 Odd Field Offset bits 10-8		
7	6	5	4	3	2	1	0

REG[0D73h] bits 2-0
REG[0D72h] bits 7-0 Camera2 Odd Field Offset bits [10:0]
When REG[0D70h] bits 1-0 = 10b or 11b, these bits specify the odd field offset.

REG[0D74h] Camera2 Even Field Offset Register 0 Default = 00h							Read/Write
Camera2 Even Field Offset bits 7-0							
7	6	5	4	3	2	1	0

REG[0D75h] Camera2 Even Field Offset Register 1 Default = 00h							Read/Write
n/a					Camera2 Even Field Offset bits 10-8		
7	6	5	4	3	2	1	0

REG[0D75h] bits 2-0
REG[0D74h] bits 7-0 Camera2 Even Field Offset bits [10:0]
When REG[0D70h] bits 1-0 = 10b or 11b, these bits specify the even field offset.

10.4.19 DMA Controller Registers

Note
The DMAC controller must not be programmed for burst accesses that cross SRAM banks. See Chapter 8, "Memory Map" on page 130 for further information.

REG[3C00h] DMA Channel 0 Source Address Register 0 Default = 00h							Read/Write
DMA Channel 0 Source Address bits 7-0							
7	6	5	4	3	2	1	0

REG[3C01h] DMA Channel 0 Source Address Register 1 Default = 00h							Read/Write
DMA Channel 0 Source Address bits 15-8							
7	6	5	4	3	2	1	0

REG[3C02h] DMA Channel 0 Source Address Register 2 Default = 00h							Read/Write
DMA Channel 0 Source Address bits 23-16							
7	6	5	4	3	2	1	0

REG[3C03h] DMA Channel 0 Source Address Register 3 Default = 00h							Read/Write
DMA Channel 0 Source Address bits 31-24							
7	6	5	4	3	2	1	0

REG[3C03h] bits 7-0
REG[3C02h] bits 7-0
REG[3C01h] bits 7-0
REG[3C00h] bits 7-0 DMA Channel 0 Source Address bits [31:0]

These bits specify the source start address for DMA Channel 0. The source address is incremented/decremented according to the settings specified in the DMA Channel 0 Control registers (REG[3C0Ch] and REG[3C0Dh]). The source start address must be aligned based on the transfer size specified by the DMA Channel 0 Transfer Size bits, REG[3C0Ch] bits 5-4. For 8-bit transfers, any alignment is allowed. For 16-bit transfers, the address must 2-byte aligned. For 32-bit transfers, the address must be 4-byte aligned.

These bits also specify the "fill" data source when Fill Mode is selected as the DMA Channel 0 Source Mode, REG[3C0Ch] bits 1-0 = 11b.

REG[3C04h] DMA Channel 0 Destination Address Register 0 Default = 00h Read/Write							
DMA Channel 0 Destination Address bits 7-0							
7	6	5	4	3	2	1	0

REG[3C05h] DMA Channel 0 Destination Address Register 1 Default = 00h Read/Write							
DMA Channel 0 Destination Address bits 15-8							
7	6	5	4	3	2	1	0

REG[3C06h] DMA Channel 0 Destination Address Register 2 Default = 00h Read/Write							
DMA Channel 0 Destination Address bits 23-16							
7	6	5	4	3	2	1	0

REG[3C07h] DMA Channel 0 Destination Address Register 3 Default = 00h Read/Write							
DMA Channel 0 Destination Address bits 31-24							
7	6	5	4	3	2	1	0

REG[3C07h] bits 7-0
REG[3C06h] bits 7-0
REG[3C05h] bits 7-0
REG[3C04h] bits 7-0 DMA Channel 0 Destination Address bits [31:0]

These bits specify the destination start address for DMA Channel 0. The destination address is incremented/decremented according to the settings specified in the DMA Channel 0 Control registers (REG[3C0Ch] and REG[3C0Dh]). The destination start address must be aligned based on the transfer size specified by the DMA Channel 0 Transfer Size bits, REG[3C0Ch] bits 5-4. For 8-bit transfers, any alignment is allowed. For 16-bit transfers, the address must 2-byte aligned. For 32-bit transfers, the address must be 4-byte aligned.

REG[3C08h] DMA Channel 0 Transfer Count Register 0 Default = 00h							Read/Write
DMA Channel 0 Transfer Count bits 7-0							
7	6	5	4	3	2	1	0

REG[3C09h] DMA Channel 0 Transfer Count Register 1 Default = 00h							Read/Write
DMA Channel 0 Transfer Count bits 15-8							
7	6	5	4	3	2	1	0

REG[3C0Ah] DMA Channel 0 Transfer Count Register 2 Default = 00h							Read/Write
DMA Channel 0 Transfer Count bits 23-16							
7	6	5	4	3	2	1	0

REG[3C0Ah] bits 7-0
REG[3C09h] bits 7-0
REG[3C08h] bits 7-0 DMA Channel 0 Transfer Count bits [23:0]
These bits specify the amount of data units (8, 16, or 32-bit words) to transfer for DMA Channel 0. For example, if the transfer size (REG[3C03h] bits 5-4) is 16-bit data and the value of this register is 20, 20 x 16-bit of data will be transferred. These registers are decremented for each word transferred and will be 0000_0000h at the end of a transfer

REG[3C0Ch] DMA Channel 0 Control Register 0 Default = 00h							Read/Write
n/a	DMA Channel 0 Non-Burst Mode Enable	DMA Channel 0 Transfer Size bits 1-0		DMA Channel 0 Destination Address Mode bits 1-0		DMA Channel 0 Source Address Mode bits 1-0	
7	6	5	4	3	2	1	0

bit 6 DMA Channel 0 Non-Burst Mode Enable
This bit determines whether the transfer on DMA Channel 0 uses non-burst mode or burst mode.
When this bit = 0b, DMA Channel 0 uses burst mode for transfers. (default)
When this bit = 1b, DMA Channel 0 uses non-burst mode for transfers.

Note
If the DMA operation will span across SRAM banks (see Chapter 8, "Memory Map" on page 130) this bit must be set to 1b.

bits 5-4 DMA Channel 0 Transfer Size bits [1:0]
These bits select the transfer size for DMA Channel 0.

Table 10-72 : DMA Channel 0 Transfer Size Selection

REG[3C0Ch] bits 5-4	DMA Channel 0 Transfer Size
00b	8-bit
01b	16-bit
10b	32-bit
11b	Reserved

bits 3-2 DMA Channel 0 Destination Address Mode bits [1:0]
These bits select the method used to update the DMA Channel 0 Destination Address registers (REG[3C04h] ~ REG[3C07h]) after a successful DMA transfer.

Table 10-73 : DMA Channel 0 Destination Address Mode Selection

REG[3C0Dh] bit 3 (Destination Stride Enable)	REG[3C0Ch] bits 3-2	DMA Channel 0 Destination Address Mode
0b	00b	Destination address is not changed.
	01b	Destination address is incremented according to the transfer size, REG[3C0Ch] bits 5-4. (8-bit: +1, 16-bit: +2, 32-bit: +4)
	10b	Destination address is decremented according to the transfer size, REG[3C0Ch] bits 5-4. (8-bit: -1, 16-bit: -2, 32-bit: -4)
	11b	Reserved
1b	00b	Destination address is not changed.
	01b	Destination address is incremented according to the specified stride, REG[3C0Dh] bits 5-4. (8-bit: +1, 16-bit: +2, 32-bit: +4, 64-bit: +8)
	10b	Destination address is decremented according to the specified stride, REG[3C0Dh] bits 5-4. (8-bit: -1, 16-bit: -2, 32-bit: -4, 64-bit: -8)
	11b	Reserved

bits 1-0 DMA Channel 0 Source Address Mode bits [1:0]
These bits select the method used to update the DMA Channel 0 Source Address registers (REG[3C00h] ~ REG[3C03h]) after a successful DMA transfer.

Table 10-74 : DMA Channel 0 Source Address Mode Selection

REG[3C0Dh] bit 0 (Source Stride Enable)	REG[3C0Ch] bits 1-0	DMA Channel 0 Source Address Mode
0b	00b	Source address is not changed.
	01b	Source address is incremented according to the transfer size, REG[3C0Ch] bits 5-4. (8-bit: +1, 16-bit: +2, 32-bit: +4)
	10b	Source address is decremented according to the transfer size, REG[3C0Ch] bits 5-4. (8-bit: -1, 16-bit: -2, 32-bit: -4)
	11b	Fill Mode - the Source Address registers are used as the fill data and are not incremented or decremented.
1b	00b	Source address is not changed.
	01b	Source address is incremented according to the specified stride, REG[3C0Dh] bits 2-1. (8-bit: +1, 16-bit: +2, 32-bit: +4, 64-bit: +8)
	10b	Source address is decremented according to the specified stride, REG[3C0Dh] bits 2-1. (8-bit: -1, 16-bit: -2, 32-bit: -4, 64-bit: -8)
	11b	Reserved

<table>
<tr><td colspan="8">REG[3C0Dh] DMA Channel 0 Control Register 1
Default = 00h — Read/Write</td></tr>
<tr><td colspan="2">n/a</td><td colspan="2">DMA Channel 0 Destination Stride bits 1-0</td><td>DMA Channel 0 Destination Stride Enable</td><td colspan="2">DMA Channel 0 Source Stride bits 1-0</td><td>DMA Channel 0 Source Stride Enable</td></tr>
<tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
</table>

bits 5-4 DMA Channel 0 Destination Stride bits [1:0]
When the DMA Channel 0 Destination Stride Enable bit is set (REG[3C0Dh] bit 3 = 1b), these bits determine the size (or stride) used to increment/decrement the DMA Channel 0 destination address. For further information, refer to the DMA Channel 0 Destination Address Mode bit description (see REG[3C0Ch] bits 3-2).

Table 10-75 : DMA Channel 0 Destination Stride Selection

REG[3C0Dh] bits 5-4	DMA Channel 0 Destination Stride
00b	8-bit, destination address is incremented/decremented by 1
01b	16-bit, destination address is incremented/decremented by 2
10b	32-bit, destination address is incremented/decremented by 4
11b	64-bit, destination address is incremented/decremented by 8

bit 3 DMA Channel 0 Destination Stride Enable
This bit selects whether the transfer size (REG[3C0Ch] bits 5-4) or the destination stride (REG[3C0Dh] bits 5-4) determines the increment/decrement size applied to the DMA Channel 0 Destination Address registers (REG[3C04h] ~ REG[3C07h]) after a successful DMA transfer.
When this bit = 0b, the destination stride is disabled and the DMA Channel 0 Transfer Size bits (REG[3C0Ch] bits 5-4) determine the increment/decrement size.
When this bit = 1b, the destination stride is enabled and the DMA Channel 0 Destination Stride bits (REG[3C0Dh] bits 5-4) determine the increment/decrement size.

bits 2-1 DMA Channel 0 Source Stride bits [1:0]
When the DMA Channel 0 Source Stride Enable bit is set (REG[3C0Dh] bit 0 = 1b), these bits determine the size (or stride) used to increment/decrement the DMA Channel 0 source address. For further information, refer to the DMA Channel 0 Source Address Mode bit description (see REG[3C0Ch] bits 1-0).

Table 10-76 : DMA Channel 0 Source Stride Selection

REG[3C0Dh] bits 2-1	DMA Channel 0 Source Stride
00b	8-bit, source address is incremented/decremented by 1
01b	16-bit, source address is incremented/decremented by 2
10b	32-bit, source address is incremented/decremented by 4
11b	64-bit, source address is incremented/decremented by 8

bit 0 DMA Channel 0 Source Stride Enable
This bit selects whether the transfer size (REG[3C0Ch] bits 5-4) or the source stride (REG[3C0Dh] bits 2-1) determines the increment/decrement size applied to the DMA Channel 0 Source Address registers (REG[3C00h] ~ REG[3C03h]) after a successful DMA transfer.
When this bit = 0b, the source stride is disabled and the DMA Channel 0 Transfer Size bits (REG[3C0Ch] bits 5-4) determine the increment/decrement size.
When this bit = 1b, the source stride is enabled and the DMA Channel 0 Source Stride bits (REG[3C0Dh] bits 2-1) determine the increment/decrement size.

REG[3C10h] DMA Channel 1 Source Address Register 0 Default = 00h							Read/Write
DMA Channel 1 Source Address bits 7-0							
7	6	5	4	3	2	1	0

REG[3C11h] DMA Channel 1 Source Address Register 1 Default = 00h							Read/Write
DMA Channel 1 Source Address bits 15-8							
7	6	5	4	3	2	1	0

REG[3C12h] DMA Channel 1 Source Address Register 2 Default = 00h							Read/Write
DMA Channel 1 Source Address bits 23-16							
7	6	5	4	3	2	1	0

REG[3C13h] DMA Channel 1 Source Address Register 3 Default = 00h							Read/Write
DMA Channel 1 Source Address bits 31-24							
7	6	5	4	3	2	1	0

REG[3C13h] bits 7-0
REG[3C12h] bits 7-0
REG[3C11h] bits 7-0
REG[3C10h] bits 7-0 DMA Channel 1 Source Address bits [31:0]
These bits specify the source start address for DMA Channel 1. The source address is incremented/decremented according to the settings specified in the DMA Channel 1 Control registers (REG[3C1Ch] and REG[3C1Dh]). The source start address must be aligned based on the transfer size specified by the DMA Channel 1 Transfer Size bits, REG[3C1Ch] bits 5-4. For 8-bit transfers, any alignment is allowed. For 16-bit transfers, the address must 2-byte aligned. For 32-bit transfers, the address must be 4-byte aligned.

These bits also specify the “fill” data source when Fill Mode is selected as the DMA Channel 1 Source Mode, REG[3C1Ch] bits 1-0 = 11b.

REG[3C14h] DMA Channel 1 Destination Address Register 0 Default = 00h							Read/Write
DMA Channel 1 Destination Address bits 7-0							
7	6	5	4	3	2	1	0

REG[3C15h] DMA Channel 1 Destination Address Register 1 Default = 00h							Read/Write
DMA Channel 1 Destination Address bits 15-8							
7	6	5	4	3	2	1	0

REG[3C16h] DMA Channel 1 Destination Address Register 2 Default = 00h							Read/Write
DMA Channel 1 Destination Address bits 23-16							
7	6	5	4	3	2	1	0

REG[3C17h] DMA Channel 1 Destination Address Register 3 Default = 00h							Read/Write
DMA Channel 1 Destination Address bits 31-24							
7	6	5	4	3	2	1	0

REG[3C17h] bits 7-0
REG[3C16h] bits 7-0
REG[3C15h] bits 7-0
REG[3C14h] bits 7-0 DMA Channel 1 Destination Address bits [31:0]

These bits specify the destination start address for DMA Channel 1. The destination address is incremented/decremented according to the settings specified in the DMA Channel 1 Control registers (REG[3C1Ch] and REG[3C1Dh]). The destination start address must be aligned based on the transfer size specified by the DMA Channel 1 Transfer Size bits, REG[3C1Ch] bits 5-4. For 8-bit transfers, any alignment is allowed. For 16-bit transfers, the address must 2-byte aligned. For 32-bit transfers, the address must be 4-byte aligned.

REG[3C18h] DMA Channel 1 Transfer Count Register 0 Default = 00h							Read/Write
DMA Channel 1 Transfer Count bits 7-0							
7	6	5	4	3	2	1	0

REG[3C19h] DMA Channel 1 Transfer Count Register 1 Default = 00h							Read/Write
DMA Channel 1 Transfer Count bits 15-8							
7	6	5	4	3	2	1	0

REG[3C1Ah] DMA Channel 1 Transfer Count Register 2 Default = 00h							Read/Write
DMA Channel 1 Transfer Count bits 23-16							
7	6	5	4	3	2	1	0

REG[3C1Ah] bits 7-0
REG[3C19h] bits 7-0
REG[3C18h] bits 7-0 DMA Channel 1 Transfer Count bits [23:0]
These bits specify the amount of data units (8, 16, or 32-bit words) to transfer for DMA Channel 1. For example, if the control register specifies transfer size of 16-bit data and the value of this register is 20, 20 x 16-bit of data will be transferred. These registers are decremented for each word transferred and will be 0000_0000h at the end of a transfer.

REG[3C1Ch] DMA Channel 1 Control Register 0 Default = 00h							Read/Write
n/a	DMA Channel 1 Non-Burst Mode Enable	DMA Channel 1 Transfer Size bits 1-0		DMA Channel 1 Destination Address Mode bits 1-0		DMA Channel 1 Source Address Mode bits 1-0	
7	6	5	4	3	2	1	0

bit 6 DMA Channel 1 Non-Burst Mode Enable
This bit determines whether the transfer on DMA Channel 1 uses non-burst mode or burst mode.
When this bit = 0b, DMA Channel 1 uses burst mode for transfers. (default)
When this bit = 1b, DMA Channel 1 uses non-burst mode for transfers.

Note
If the DMA operation will span across SRAM banks (see Chapter 8, “Memory Map” on page 130) this bit must be set to 1b.

bits 5-4 DMA Channel 1 Transfer Size bits [1:0]
These bits select the transfer size for DMA Channel 1.

Table 10-77 : DMA Channel 1 Transfer Size Selection

REG[3C1Ch] bits 5-4	DMA Channel 1 Transfer Size
00b	8-bit
01b	16-bit
10b	32-bit
11b	Reserved

bits 3-2 DMA Channel 1 Destination Address Mode bits [1:0]
These bits select the method used to update the DMA Channel 1 Destination Address registers (REG[3C14h] ~ REG[3C17h]) after a successful DMA transfer.

Table 10-78 : DMA Channel 1 Destination Address Mode Selection

REG[3C1Dh] bit 3 (Destination Stride Enable)	REG[3C1Ch] bits 3-2	DMA Channel 1 Destination Address Mode
0b	00b	Destination address is not changed.
	01b	Destination address is incremented according to the transfer size, REG[3C1Ch] bits 5-4. (8-bit: +1, 16-bit: +2, 32-bit: +4)
	10b	Destination address is decremented according to the transfer size, REG[3C1Ch] bits 5-4. (8-bit: -1, 16-bit: -2, 32-bit: -4)
	11b	Reserved
1b	00b	Destination address is not changed.
	01b	Destination address is incremented according to the specified stride, REG[3C1Dh] bits 5-4. (8-bit: +1, 16-bit: +2, 32-bit: +4, 64-bit: +8)
	10b	Destination address is decremented according to the specified stride, REG[3C1Dh] bits 5-4. (8-bit: -1, 16-bit: -2, 32-bit: -4, 64-bit: -8)
	11b	Reserved

bits 1-0 DMA Channel 1 Source Address Mode bits [1:0]
These bits select the method used to update the DMA Channel 1 Source Address registers (REG[3C10h] ~ REG[3C13h]) after a successful DMA transfer.

Table 10-79 : DMA Channel 1 Source Address Mode Selection

REG[3C1Dh] bit 0 (Source Stride Enable)	REG[3C1Ch] bits 1-0	DMA Channel 1 Source Address Mode
0b	00b	Source address is not changed.
	01b	Source address is incremented according to the transfer size, REG[3C1Ch] bits 5-4. (8-bit: +1, 16-bit: +2, 32-bit: +4)
	10b	Source address is decremented according to the transfer size, REG[3C1Ch] bits 5-4. (8-bit: -1, 16-bit: -2, 32-bit: -4)
	11b	Fill Mode - the Source Address registers are used as the fill data and are not incremented or decremented.
1b	00b	Source address is not changed.
	01b	Source address is incremented according to the specified stride, REG[3C1Dh] bits 2-1. (8-bit: +1, 16-bit: +2, 32-bit: +4, 64-bit: +8)
	10b	Source address is decremented according to the specified stride, REG[3C1Dh] bits 2-1. (8-bit: -1, 16-bit: -2, 32-bit: -4, 64-bit: -8)
	11b	Reserved

<table>
<tr><td colspan="8">REG[3C1Dh] DMA Channel 1 Control Register 1
Default = 00h Read/Write</td></tr>
<tr><td colspan="2">n/a</td><td colspan="2">DMA Channel 1 Destination Stride bits 1-0</td><td>DMA Channel 1 Destination Stride Enable</td><td colspan="2">DMA Channel 1 Source Stride bits 1-0</td><td>DMA Channel 1 Source Stride Enable</td></tr>
<tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
</table>

bits 5-4 DMA Channel 1 Destination Stride bits [1:0]
When the DMA Channel 1 Destination Stride Enable bit is set (REG[3C1Dh] bit 3 = 1b), these bits determine the size (or stride) used to increment/decrement the DMA Channel 1 destination address. For further information, refer to the DMA Channel 1 Destination Address Mode bit description (see REG[3C1Ch] bits 3-2).

Table 10-80 : DMA Channel 1 Destination Stride Selection

REG[3C1Dh] bits 5-4	DMA Channel 1 Destination Stride
00b	8-bit, destination address is incremented/decremented by 1
01b	16-bit, destination address is incremented/decremented by 2
10b	32-bit, destination address is incremented/decremented by 4
11b	64-bit, destination address is incremented/decremented by 8

bit 3 DMA Channel 1 Destination Stride Enable
This bit selects whether the transfer size (REG[3C1Ch] bits 5-4) or the destination stride (REG[3C1Dh] bits 5-4) determines the increment/decrement size applied to the DMA Channel 1 Destination Address registers (REG[3C14h] ~ REG[3C17h]) after a successful DMA transfer.
When this bit = 0b, the destination stride is disabled and the DMA Channel 1 Transfer Size bits (REG[3C1Ch] bits 5-4) determine the increment/decrement size.
When this bit = 1b, the destination stride is enabled and the DMA Channel 1 Destination Stride bits (REG[3C1Dh] bits 5-4) determine the increment/decrement size.

bits 2-1 DMA Channel 1 Source Stride bits [1:0]
When the DMA Channel 1 Source Stride Enable bit is set (REG[3C1Dh] bit 0 = 1b), these bits determine the size (or stride) used to increment/decrement the DMA Channel 1 source address. For further information, refer to the DMA Channel 1 Source Address Mode bit description (see REG[3C1Ch] bits 1-0).

Table 10-81 : DMA Channel 1 Source Stride Selection

REG[3C1Dh] bits 2-1	DMA Channel 1 Source Stride
00b	8-bit, source address is incremented/decremented by 1
01b	16-bit, source address is incremented/decremented by 2
10b	32-bit, source address is incremented/decremented by 4
11b	64-bit, source address is incremented/decremented by 8

bit 0 DMA Channel 1 Source Stride Enable
This bit selects whether the transfer size (REG[3C1Ch] bits 5-4) or the source stride (REG[3C1Dh] bits 2-1) determines the increment/decrement size applied to the DMA Channel 1 Source Address registers (REG[3C10h] ~ REG[3C13h]) after a successful DMA transfer.
When this bit = 0b, the source stride is disabled and the DMA Channel 1 Transfer Size bits (REG[3C1Ch] bits 5-4) determine the increment/decrement size.
When this bit = 1b, the source stride is enabled and the DMA Channel 1 Source Stride bits (REG[3C1Dh] bits 2-1) determine the increment/decrement size.

REG[3C20h] DMA Status Register Default = 00h							Read Only
n/a						DMA Channel Status	DMA Controller Busy
7	6	5	4	3	2	1	0

bit 1 DMA Channel Status (Read Only)
When the DMA is busy (REG[3C20h] bit 0 = 1b), this bit indicates which DMA channel is being serviced. If there is no pending transfer on the other channel after the current DMA transfer finishes, this bit remains at the current state.
When this bit = 0b, DMA Channel 0 is being serviced.
When this bit = 1b, DMA Channel 1 is being serviced.

bit 0 DMA Controller Busy (Read Only)
This bit indicates when the DMA Controller is busy doing a transfer.
When this bit = 0b, the DMA controller is idle.
When this bit = 1b, the DMA controller is busy.

REG[3C22h] DMA Start Register Default = 00h							Read/Write
n/a						DMA Channel 1 Start	DMA Channel 0 Start
7	6	5	4	3	2	1	0

bit 1 DMA Channel 1 Start
This bit initiates a DMA transfer for DMA Channel 1.
Writes:
Writing a 0b to this bit has no effect.
Writing a 1b to this bit starts the DMA transfer for Channel 1. Once a transfer is started, this bit remains at 1b until the end of the transfer, even if a 0b is written to it. After the transfer completes, this bit is automatically cleared to 0b.
Reads:
When this bit = 0b, there are no active DMA transfers on DMA Channel 1.
When this bit = 1b, a DMA transfer is active or queued for DMA Channel 1.

Note
A DMA transfer for DMA Channel 0 may be started before the DMA transfer on channel 1 has completed. The new DMA transfer will start once the current DMA transfer completes.

bit 0 DMA Channel 0 Start
This bit initiates a DMA transfer for DMA Channel 0.
Writes:
Writing a 0b to this bit has no effect.
Writing a 1b to this bit starts the DMA transfer for Channel 0. Once a transfer is started, this bit remains at 1b until the end of the transfer, even if a 0b is written to it. After the transfer completes, this bit is automatically cleared to 0b.
Reads:
When this bit = 0b, there are no active DMA transfers on DMA Channel 0.
When this bit = 1b, a DMA transfer is active or queued for DMA Channel 0.

Note

A DMA transfer for DMA Channel 1 may be started before the DMA transfer on channel 0 has completed. The new DMA transfer will start once the current DMA transfer completes.

10.4.20 SDRAM Controller Configuration Registers

REG[3C40h] SDRAM Control Register Default = 02h							Read/Write
SDRAM tRCD Timing	SDRAM tRAS Timing	SDRAM tRP Timing	SDRAM CAS Latency	SDRAM Type bits 1-0		16 or 32 Bit SDRAM Interface	SDRAM Initialize
7	6	5	4	3	2	1	0

bit 7 SDRAM tRCD Timing
This bit selects the tRCD timing for the SDRAM (active to read/write command).
When this bit = 0b, tRCD is a minimum of 2 clock cycles.
When this bit = 1b, tRCD is a minimum of 4 clock cycles.

bit 6 SDRAM tRAS Timing
This bit selects the minimum tRAS timing for the SDRAM burst READ timing (active to precharge).
When this bit = 0b, tRAS is a minimum of 4 clock cycles.
When this bit = 1b, tRAS is a minimum of 6 clock cycles.

Note that the actual number of clock cycles for tRAS is determined by this bit and bit 7 (tRCD) according to the following formula.
tRAS = 4 + (2 x REG[3C40h] bit 6) + (2 x REG[3C40h] bit 7)

bit 5 SDRAM tRP Timing
This bit selects the tRP timing for the SDRAM (precharge to active).
When this bit = 0b, tRP is a minimum of 2 clock cycles.
When this bit = 1b, tRP is a minimum of 4 clock cycles.

bit 4 SDRAM CAS Latency
This bit selects the CAS Latency for reads from the SDRAM.
When this bit = 0b, the CAS latency is 2 clocks.
When this bit = 1b, the CAS latency is 3 clocks.

bits 3-2 SDRAM Type bits [1:0]
These bits selects the type of 16-bit SDRAM used.

Table 10-82 : SDRAM Type Selection

REG[3C40h] bits 3-2	SDRAM Type
00b	64Mbit, 1M x 16 x 4 banks, row A11-A0, column A7-A0
01b	128Mbit, 2M x 16 x 4 banks, row A11-A0, column A8-A0
10b	256Mbit, 4M x 16 x 4 banks, row A12-A0, column A8-A0
11b	512Mbit, 8M x 16 x 4 banks, row A12-A0, column A9-A0

bit 1 16 or 32 Bit SDRAM Interface
This bit specifies whether one or two 16-bit SDRAM devices are used.
When this bit = 0b, one 16-bit SDRAM device is used.
When this bit = 1b, two 16-bit SDRAM devices are used to form a 32-bit device.

bit 0 SDRAM Initialize
This bit is used to initialize the SDRAM after power-up. The SDRAM must be initialized before it can be used as memory space. The SDRAM is programmed using the settings in REG[3C40h] bits 7-4, and full page mode access.
For Writes:
Writing a 0b to this bit has no effect.
Writing a 1b to this bit causes the SDRAM controller to initiate the initialization sequence for the SDRAM. This bit remains at 1b while the SDRAM is being initialized and is automatically reset to 0b once the initialization is complete.
For Reads:
When this bit = 0b, the SDRAM is not being initialized.
When this bit = 1b, the SDRAM is being initialized.

Note

1. Before entering power save mode, the C33 must be placed in HALT or SLEEP mode (through instruction code), or placed in reset (REG[001Dh] bit 0).
   To maintain DRAM contents while in powersave mode, place the DRAM controller in self-refresh mode in REG[3C44h] bit 6 before entering power save mode.
2. After exiting powersave mode, if self refresh mode is enabled, exit self refresh mode in REG[3C44h] bit 6 before enabling any accesses to DRAM.
3. After exiting power save mode, the DRAM controller must be re-initialized by writing a 1b to REG[3C40h] bit 0 and waiting for the bit to return a 0b before enabling any accesses to DRAM.
4. After exiting power save mode, Note 5 or 6 must be met before the C33 can safely exit HALT or SLEEP mode, or be released from reset (REG[001Dh] bit 0).

REG[3C42h] SDRAM Refresh Period Register 0
Default = 00h Read/Write

SDRAM Refresh Period bits 7-0							
7	6	5	4	3	2	1	0

REG[3C43h] SDRAM Refresh Period Register 1
Default = 01h Read/Write

n/a				SDRAM Refresh Period bits 11-8			
7	6	5	4	3	2	1	0

REG[3C43h] bits 3-0
REG[3C42h] bits 7-0 SDRAM Refresh Period bits [11:0]
These bits specify the time period between Auto-Refresh commands used for refreshing the SDRAM. The refresh period is defined by the following formula.

Refresh period = ((REG[3C43h], REG[3C42h]) + 1) x System Clock Period

REG[3C44h] SDRAM Clock Control Register Default = 05h							Read/Write
n/a	SDRAM Self Refresh Enable	n/a	Reserved				
7	6	5	4	3	2	1	0

bit 6 SDRAM Self Refresh Enable
This bit controls whether the SDRAM is in self refresh mode.
When this bit = 0b, the SDRAM is in normal mode.
When this bit = 1b, the SDRAM is in self refresh mode.

bits 4-0 Reserved
These bits must be set to 14h (1_0100b).

10.4.21 LCD Panel Configuration Registers

REG[4000h] LCD Panel Type Select Register 0 Default = 88h							Read/Write
LCD2 Panel Mode bits 1-0		LCD2 Panel Type Select bits 1-0		LCD1 / Camera2 Select	LCD1 Panel Mode	Use I2S/PWM Pins For EID	Reserved
7	6	5	4	3	2	1	0

bits 7-6 LCD2 Panel Mode Select bits [1:0]
These bits select the panel mode for LCD2 which uses the FP2IO pins. For pin mapping, see Section 5.6, "LCD / Camera2 Pin Mapping" on page 40.

Table 10-83: LCD2 Panel Mode Select

REG[4000h] bits 7-6	LCD2 Panel Mode Select
00b	RGB 8:8:8 only
01b	RGB 6:6:6 with serial interface
10b	RGB 6:6:6 without serial interface (default)
11b	Reserved

bits 5-4 LCD2 Panel Type Select bits [1:0]
These bits select the type of panel connected to the LCD2 panel interface. For pin mapping, see Section 5.6, "LCD / Camera2 Pin Mapping" on page 40.

Table 10-84: LCD2 Panel Type Select

REG[4000h] bits 5-4	LCD2 Panel Type Select
00b	Generic RGB
01b	EID Double Screen
10b	Sharp DualView
11b	Reserved

bit 3 LCD1 / Camera2 Select
This bit selects whether the FP1IO pins are used for LCD1 or Camera2 support. For pin mapping, see Section 5.6, "LCD / Camera2 Pin Mapping" on page 40.
When this bit = 0b, the FP1IO pins are used for LCD1.
When this bit = 1b, the FP1IO pins are used for Camera2. (default)

bit 2 LCD1 Panel Mode Select
This bit selects the panel mode for LCD1 which uses the FP1IO pins.
When this bit = 0b, a RGB interface without serial interface is selected. When this mode is selected and LCD2 does not use any LCD1 pins, the format is RGB 6:6:6. If LCD2 uses LCD1 pins for EID Double Screen support (see Section 5.6, “LCD / Camera2 Pin Mapping” on page 40), the format is RGB 5:5:5. (default)
When this bit = 1b, a RGB interface with serial interface is selected. When this mode is selected and LCD2 does not use any LCD1 pins, the format is RGB 5:6:5. If LCD2 uses LCD1 pins for EID Double Screen support, the format is RGB 4:4:4.

Note

If an EID Double Screen panel is used on LCD2 and a RGB 5:6:5 interface with serial interface is required on LCD1, the I2S/PWM pins can be used for the extra pins required by the EID Double Screen panel (see REG[4000h] bit 1).

bit 1 Use I2S/PWM Pins For EID
This bit determines whether the I2S/PWM pins are used for the I2S interface or for EID Double Screen panel support. For pin mapping, see Section 5.6, “LCD / Camera2 Pin Mapping” on page 40.
When this bit = 0b, the I2S/PWM pins are not used for outputting EID Double Screen panel signals. In this case, the LCD1 RGB interface panel supports either RGB 5:5:5 or RGB 4:4:4 depending on whether the serial interface is enabled (see REG[4000h] bit 2). (default)
When this bit = 1b, the I2S/PWM pins are used for outputting EID Double Screen panel signals. In this case, the LCD1 RGB interface panel supports either RGB 6:6:6 or RGB 5:6:5 depending on whether the serial interface is enabled (see REG[4000h] bit 2).

bit 0 Reserved
This bit must be set to 0b.

REG[4001h] LCD Panel Type Select Register 1 Default = 00h							Read/Write
LCD2 PCLK Polarity Select	LCD2 Panel Data Bus Width bits 2-0			LCD1 PCLK Polarity Select	Reserved	LCD1 Panel Data Bus Width bits 1-0	
7	6	5	4	3	2	1	0

bit 7 LCD2 PCLK Polarity Select
This bit selects the polarity of the PCLK signal for the LCD2 interface.
When this bit = 0b, the LCD2 PCLK signal polarity is normal (display data is latched on rising edge).
When this bit = 1b, the LCD2 PCLK signal polarity is inverted (display data is latched on falling edge).

Note
PCLK Polarity Select does not affect the polarity of ODCK when REG[4040h] bit 0 Double Screen Panel TCON Enable = 1b and EID display is selected (REG[4000h] bits 5-4 = 01b), but does affect the polarity when REG[4040h] bit 0 = 0b.

bits 6-4 LCD2 Panel Data Bus Width bits [2:0]
These bits select the data bus width for the LCD2 panel.

Table 10-85: LCD2 Panel Data Bus Width Selection

REG[4001h] bits 6-4	LCD2 Panel Data Bus Width
000b	12-bit
001b	16-bit
010b	18-bit
011b	24-bit
100b ~ 111b	Reserved

bit 3 LCD1 PCLK Polarity Select
This bit selects the polarity of the PCLK signal for the LCD1 interface.
When this bit = 0b, the LCD1 PCLK signal polarity is normal (display data is latched on rising edge).
When this bit = 1b, the LCD1 PCLK signal polarity is inverted (display data is latched on falling edge).

bit 2 Reserved
The default value for this bit is 0b.

bits 1-0 LCD1 Panel Data Bus Width bits [1:0]
These bits select the data bus width for the LCD1 panel.

Table 10-86: LCD1 Panel Data Bus Width Selection

REG[4001h] bits 1-0	LCD1 Panel Data Bus Width
00b	12-bit
01b	16-bit
10b	18-bit
11b	Reserved

REG[4002h] LCD1 Horizontal Total Register 0 Default = 00h							Read/Write
LCD1 Horizontal Total bits 7-0							
7	6	5	4	3	2	1	0

REG[4003h] LCD1 Horizontal Total Register 1 Default = 00h							Read/Write
n/a				LCD1 Horizontal Total bits 11-8			
7	6	5	4	3	2	1	0

REG[4003h] bits 3-0
REG[4002h] bits 7-0 LCD1 Horizontal Total bits [11:0]

These bits specify the Horizontal Total for LCD1, in pixel clock periods. The Horizontal Total is the sum of the Horizontal Display Period and the Horizontal Non-Display Period. For detailed timing information, see Section 7.6, "Panel Interface Timing" on page 95.

(REG[4003h] bits 3-0, REG[4002h] bits 7-0) = Horizontal Total Period - 1

REG[4004h] LCD1 Horizontal Display Period Register 0 Default = 00h							Read/Write
LCD1 Horizontal Display Period bits 7-0							
7	6	5	4	3	2	1	0

REG[4005h] LCD1 Horizontal Display Period Register 1 Default = 00h							Read/Write
n/a					LCD1 Horizontal Display Period bits 10-8		
7	6	5	4	3	2	1	0

REG[4005h] bits 2-0
REG[4004h] bits 7-0 LCD1 Horizontal Display Period bits [10:0]

These bits specify the Horizontal Display Period for LCD1, in 2 pixel resolution. The Horizontal Display Period must be less than the Horizontal Total to allow for sufficient Horizontal Non-Display Period. For detailed timing information, see Section 7.6, "Panel Interface Timing" on page 95.

(REG[4005h] bits 2-0, REG[4004h] bits 7-0) = (Horizontal Display Period ÷ 2) - 1

REG[4006h] LCD1 Horizontal Display Period Start Position Register 0 Default = 00h Read/Write							
LCD1 Horizontal Display Period Start Position bits 7-0							
7	6	5	4	3	2	1	0

REG[4007h] LCD1 Horizontal Display Period Start Position Register 1 Default = 00h Read/Write							
n/a				LCD1 Horizontal Display Period Start Position bits 11-8			
7	6	5	4	3	2	1	0

REG[4007h] bits 3-0
REG[4006h] bits 7-0 LCD1 Horizontal Display Period Start Position bits [11:0]

These bits specify the Horizontal Display Period Start Position for LCD1, in pixel clock periods. For detailed timing information, see Section 7.6, "Panel Interface Timing" on page 95.

(REG[4007h] bits 3-0, REG[4006h] bits 7-0) = Horizontal Display Period Start Position - 1

REG[4008h] LCD1 Horizontal Pulse Width Register 0 Default = 00h Read/Write							
LCD1 Horizontal Pulse Width bits 7-0							
7	6	5	4	3	2	1	0

REG[4009h] LCD1 Horizontal Pulse Width Register 1 Default = 00h Read/Write							
LCD1 Horizontal Pulse Polarity Select	n/a						LCD1 Horizontal Pulse Width bit 8
7	6	5	4	3	2	1	0

REG[4009h] bit 0
REG[4008h] bits 7-0 LCD1 Horizontal Pulse Width bits [8:0]

These bits specify the pulse width of the LCD1 horizontal sync signal (HSYNC), in pixel clock periods. For detailed timing information, see Section 7.6, "Panel Interface Timing" on page 95.

(REG[4009h] bit 0, REG[4008h] bits 7-0) = Horizontal Pulse Width - 1

REG[4009h] bit 7 LCD1 Horizontal Pulse Polarity Select

This bit selects the polarity of the LCD1 horizontal sync signal (HSYNC).
When this bit = 0b, the horizontal sync signal (HSYNC) is active low. (default)
When this bit = 1b, the horizontal sync signal (HSYNC) is active high.

REG[400Ah] LCD1 Horizontal Pulse Start Position Register 0 Default = 00h							Read/Write
LCD1 Horizontal Pulse Start Position bits 7-0							
7	6	5	4	3	2	1	0

REG[400Bh] LCD1 Horizontal Pulse Start Position Register 1 Default = 00h							Read/Write
n/a				LCD1 Horizontal Pulse Start Position bits 11-8			
7	6	5	4	3	2	1	0

REG[400Bh] bits 3-0
REG[400Ah] bits 7-0 LCD1 Horizontal Pulse Start Position bits [11:0]
These bits specify the start position of the LCD1 horizontal sync pulse (HSYNC), in pixel clock periods. For detailed timing information, see Section 7.6, “Panel Interface Timing” on page 95.
(REG[400Bh] bits 3-0, REG[400Ah] bits 7-0) = Horizontal Pulse Start Position

REG[400Ch] LCD1 Vertical Total Register 0 Default = 00h							Read/Write
LCD1 Vertical Total bits 7-0							
7	6	5	4	3	2	1	0

REG[400Dh] LCD1 Vertical Total Register 1 Default = 00h							Read/Write
n/a				LCD1 Vertical Total bits 11-8			
7	6	5	4	3	2	1	0

REG[400Dh] bits 3-0
REG[400Ch] bits 7-0 LCD1 Vertical Total bits [11:0]
These bits specify the Vertical Total for LCD1, in lines. The Vertical Total is the sum of the Vertical Display Period and the Vertical Non-Display Period. For detailed timing information, see Section 7.6, “Panel Interface Timing” on page 95.
(REG[400Dh] bits 3-0, REG[400Ch] bits 7-0) = Vertical Total Period in lines - 1

REG[400Eh] LCD1 Vertical Display Period Register 0 Default = 00h							Read/Write
LCD1 Vertical Display Period bits 7-0							
7	6	5	4	3	2	1	0

REG[400Fh] LCD1 Vertical Display Period Register 1 Default = 00h							Read/Write
n/a				LCD1 Vertical Display Period bits 11-8			
7	6	5	4	3	2	1	0

REG[400Fh] bits 3-0
REG[400Eh] bits 7-0 LCD1 Vertical Display Period bits [11:0]
These bits specify the Vertical Display Period for LCD1, in lines. The Vertical Display Period must be less than the Vertical Total to allow for sufficient Vertical Non-Display Period. For detailed timing information, see Section 7.6, "Panel Interface Timing" on page 95.

(REG[400Fh] bits 3-0, REG[400Eh] bits 7-0) = Vertical Display Period in lines - 1

REG[4010h] LCD1 Vertical Display Period Start Position Register 0 Default = 00h							Read/Write
LCD1 Vertical Display Period Start Position bits 7-0							
7	6	5	4	3	2	1	0

REG[4011h] LCD1 Vertical Display Period Start Position Register 1 Default = 00h							Read/Write
n/a				LCD1 Vertical Display Period Start Position bits 11-8			
7	6	5	4	3	2	1	0

REG[4011h] bits 3-0
REG[4010h] bits 7-0 LCD1 Vertical Display Period Start Position bits [11:0]
These bits specify the Vertical Display Period Start Position for LCD1, in lines. For detailed timing information, see Section 7.6, "Panel Interface Timing" on page 95.

REG[4011h] bits 3-0, REG[4010h] bits 7-0) = Vertical Display Period Start Position in lines

REG[4012h] LCD1 Vertical Pulse Width Register 1 Default = 00h							Read/Write
n/a			LCD1 Vertical Pulse Width bits 4-0				
7	6	5	4	3	2	1	0

bits 4-0 LCD1 Vertical Pulse Width bits [4:0]
These bits specify the pulse width of the LCD1 vertical sync signal (VSYNC), in lines. For detailed timing information, see Section 7.6, "Panel Interface Timing" on page 95.

REG[4012h] bits 4-0 = Vertical Pulse Width in lines - 1

REG[4013h] LCD1 Vertical Pulse Polarity Register Default = 00h							Read/Write
LCD1 Vertical Pulse Polarity Select	n/a						
7	6	5	4	3	2	1	0

bit 7 LCD1 Vertical Pulse Polarity Select
This bit selects the polarity of the LCD1 vertical sync signal (VSYNC).
When this bit = 0b, the vertical sync signal (VSYNC) is active low. (default)
When this bit = 1b, the vertical sync signal (VSYNC) is active high.

REG[4014h] LCD1 Vertical Pulse Start Position Register 0 Default = 00h							Read/Write
LCD1 Vertical Pulse Start Position bits 7-0							
7	6	5	4	3	2	1	0

REG[4015h] LCD1 Vertical Pulse Start Position Register 1 Default = 00h							Read/Write
n/a				LCD1 Vertical Pulse Start Position bits 11-8			
7	6	5	4	3	2	1	0

REG[4015h] bits 3-0
REG[4014h] bits 7-0 LCD1 Vertical Pulse Start Position bits [11:0]
These bits specify the start position of the LCD1 vertical sync pulse (VSYNC), in lines.
For detailed timing information, see Section 7.6, “Panel Interface Timing” on page 95.

(REG[4015h] bits 3-0, REG[4014h] bits 7-0) = Vertical Pulse Start Position in lines

REG[4016h] LCD1 Serial Interface Configuration Register Default = 00h							Read/Write
LCD1 Serial Command Type bits 2-0			LCD1 Serial Command Direction	n/a		LCD1 Serial Clock Phase	LCD1 Serial Clock Polarity
7	6	5	4	3	2	1	0

bits 7-5 LCD1 Serial Command Type bits [2:0]
These bits determine the serial command type for LCD1. For AC timing information, see Section 7.6.2, “ND-TFD 8-Bit Serial Interface Timing” on page 101, Section 7.6.3, “ND-TFD 9-Bit Serial Interface Timing” on page 103, Section 7.6.4, “a-Si TFT Serial Interface Timing” on page 105, and Section 7.6.5, “uWIRE Serial Interface Timing” on page 106.

Table 10-87: LCD1 Serial Command Type Selection

REG[4016h] bits 7-5	Serial Command Type
000b	ND-TFD 4 pin Serial (8-bit serial data)
001b	ND-TFD 3 pin Serial (9-bit serial data)
010b	a-Si TFT Serial (8-bit serial data)
011b	Reserved
100b	μWire serial (16-bit serial data)
101b	24-bit serial data
110b - 111b	Reserved

bit 4 LCD1 Serial Command Direction
This bit determines the serial command bit direction for LCD1.
When this bit = 0b, the most significant bit is first. (default)
When this bit = 1b, the least significant bit is first.

Note
For details on timing, see Section 7.6, “Panel Interface Timing” on page 95 and refer to the appropriate serial interface.

bit 1 LCD1 Serial Clock Phase
This bit specifies the serial clock phase for LCD1. For a summary of the serial clock phase and polarity settings, see Table 10-88 “LCD1 Serial Clock Phase and Polarity Selection,” on page 384.

Note
For details on timing, see Section 7.6, “Panel Interface Timing” on page 95 and refer to the appropriate serial interface.

bit 0 LCD1 Serial Clock Polarity
This bit specifies the serial clock polarity for LCD1. The following table is a summary of the serial clock phase and polarity settings.

Table 10-88 : LCD1 Serial Clock Phase and Polarity Selection

REG[4016h] bit 1	REG[4016h] bit 0	Valid Data	Clock Idling Status
0b	0b	Rising edge of Serial Clock	Low
	1b	Falling edge of Serial Clock	High
1b	0b	Falling edge of Serial Clock	Low
	1b	Rising edge of Serial Clock	High

Note
For details on timing, see Section 7.6, "Panel Interface Timing" on page 95 and refer to the appropriate serial interface.

REG[4017h] LCD1 Serial Interface Status Register
Default = 00h Read Only

LCD1 Serial Busy Status	n/a						
7	6	5	4	3	2	1	0

bit 7 LCD1 Serial Busy Status (Read Only)
This bit indicates the busy status of the LCD1 serial interface. While serial command/parameter data is being issued, this bit will return a 1b. After the data transfer is completed, it is cleared automatically. When this bit = 1b, the host interface cannot write to the LCD Serial Command/Parameter registers, REG[401Ch] ~ REG[401Fh].
When this bit = 0b, the LCD1 serial interface is ready (not busy).
When this bit = 1b, the LCD1 serial interface is busy.

REG[4018h] LCD1 Interface Status Register
Default = 00h Read Only

n/a							LCD1 VNDP Status
7	6	5	4	3	2	1	0

bit 0 LCD1 VNDP Status (Read Only)
This bit indicates whether the LCD1 panel is in a Vertical Display Period or a Vertical Non-Display Period. To use this bit, the configured VNDP for LCD1 must be greater than 1 line.
When this bit = 0b, the LCD1 panel output is in a Vertical Display Period.
When this bit = 1b, the LCD1 panel output is in a Vertical Non-Display Period.

Note
This bit is not set when the LCD1 output is disabled (REG[4060h] bit 0 = 0b) or LCD1 power save mode is enabled (REG[40B0h] bit 0 = 1b). When the LCD interface is disabled, or power save is enabled, this bit should be ignored.

REG[4019h] LCD1 VSYNC Register Default = 00h Read/Write							
LCD1 VSYNC Interrupt Enable	n/a			LCD1 VSYNC Interrupt Status	Reserved	n/a	
7	6	5	4	3	2	1	0

bit 7 LCD1 VSYNC Interrupt Enable
This bit controls whether a LCD1 VSYNC Interrupt will cause a LCD1 Interrupt to occur in REG[0A00h] bit 0.
When this bit = 0b, the LCD1 VSYNC Interrupt status is not output to the Interrupt Controller.
When this bit = 1b, the LCD1 VSYNC Interrupt status is output to the Interrupt Controller.

bit 3 LCD1 VSYNC Interrupt Status
This bit indicates the status of the VSYNC Interrupt for LCD1. This bit is masked by the LCD1 VSYNC Interrupt Mask Disable bit and is not available when REG[4019h] bit 2 = 0b. When this bit indicates that a LCD1 VSYNC Interrupt has occurred and the LCD1 VSYNC Interrupt Enable bit is set (REG[4019h] bit 7 = 1b), a LCD1 Interrupt occurs in REG[0A00h] bit 0.
When this bit = 0b, a LCD1 VSYNC Interrupt has not occurred.
When this bit = 1b, a LCD1 VSYNC Interrupt has occurred.

To clear this status bit, write a 1b to this bit.

bit 2 Reserved
This bit MUST be set to 1b for normal operation.

REG[401Ah] LCD1 VSYNC Interrupt Delay Register 0 Default = 00h							Read/Write
LCD1 VSYNC Interrupt Delay bits 7-0							
7	6	5	4	3	2	1	0

REG[401Bh] LCD1 VSYNC Interrupt Delay Register 1 Default = 00h							Read/Write
n/a				LCD1 VSYNC Interrupt Delay bits 11-8			
7	6	5	4	3	2	1	0

REG[401Bh] bits 3-0
REG[401Ah] bits 7-0 LCD1 VSYNC Interrupt Delay bits [11:0]
These bits specify the VSYNC interrupt assertion timing delay from the start of the frame, in lines. If the interrupt delay is greater than the Vertical Total for LCD1 (see REG[400Ch] ~ REG[400Dh]), the interrupt will not occur.

REG[401Ch] LCD1 Serial Data Register 0 Default = 00h							Read/Write
LCD1 Serial Data bits 7-0							
7	6	5	4	3	2	1	0

REG[401Dh] LCD1 Serial Data Register 1 Default = 00h							Read/Write
LCD1 Serial Data bits 15-8							
7	6	5	4	3	2	1	0

REG[401Fh] LCD1 Serial Data Register 2 Default = 00h							Read/Write
LCD1 Serial Data bits 23-16							
7	6	5	4	3	2	1	0

REG[401Fh] bits 7-0
REG[401Dh] bits 7-0
REG[401Ch] bits 7-0 LCD1 Serial Data bits [23:0]
These bits specify the data for the LCD1 serial interface. The serial data is issued to the panel module once REG[401Dh] is written.

Table 10-89: LCD1 Serial Data

Register	24-bit Serial Interface	uWire Serial Interface	ND-TFD 8-bit Serial Interface, ND-TFD 9-bit Serial Interface, a-Si TFT Serial Interface
REG[401Ch]	Data bits 7-0	Data bits 7-0	Data bits 7-0
REG[401Dh]	Data bits 15-8	Data bits 15-8	Bit 0 is output as the signal A0 and is only used to determine whether the LCD Serial data bits 7-0 (REG[401Ch]) contain a command or parameter.
REG[401Fh]	Data bits 23-16	n/a	n/a

REG[4020h] LCD2 Horizontal Total Register 0 Default = 00h							Read/Write
LCD2 Horizontal Total bits 7-0							
7	6	5	4	3	2	1	0

REG[4021h] LCD2 Horizontal Total Register 1 Default = 00h							Read/Write
n/a				LCD2 Horizontal Total bits 11-8			
7	6	5	4	3	2	1	0

REG[4021h] bits 3-0
REG[4020h] bits 7-0 LCD2 Horizontal Total bits [11:0]

These bits specify the Horizontal Total for LCD2, in pixel clock periods. The Horizontal Total is the sum of the Horizontal Display Period and the Horizontal Non-Display Period. For detailed timing information, see Section 7.6, “Panel Interface Timing” on page 95.

(REG[4021h] bits 3-0, REG[4020h] bits 7-0) = Horizontal Total Period - 1

REG[4022h] LCD2 Horizontal Display Period Register 0 Default = 00h							Read/Write
LCD2 Horizontal Display Period bits 7-0							
7	6	5	4	3	2	1	0

REG[4023h] LCD2 Horizontal Display Period Register 1 Default = 00h							Read/Write
n/a					LCD2 Horizontal Display Period bits 10-8		
7	6	5	4	3	2	1	0

REG[4023h] bits 2-0
REG[4022h] bits 7-0 LCD2 Horizontal Display Period bits [10:0]

These bits specify the Horizontal Display Period for LCD2, in 2 pixel resolution. The Horizontal Display Period must be less than the Horizontal Total to allow for sufficient Horizontal Non-Display Period. For detailed timing information, see Section 7.6, “Panel Interface Timing” on page 95.

(REG[4023h] bits 2-0, REG[4022h] bits 7-0) = (Horizontal Display Period ÷ 2) - 1

REG[4024h] LCD2 Horizontal Display Period Start Position Register 0 Default = 00h							Read/Write
LCD2 Horizontal Display Period Start Position bits 7-0							
7	6	5	4	3	2	1	0

REG[4025h] LCD2 Horizontal Display Period Start Position Register 1 Default = 00h							Read/Write
n/a				LCD2 Horizontal Display Period Start Position bits 11-8			
7	6	5	4	3	2	1	0

REG[4025h] bits 3-0
REG[4024h] bits 7-0 LCD2 Horizontal Display Period Start Position bits [11:0]
These bits specify the Horizontal Display Period Start Position for LCD2, in pixel clock periods. For detailed timing information, see Section 7.6, "Panel Interface Timing" on page 95.

(REG[4025h] bits 3-0, REG[4024h] bits 7-0) = Horizontal Display Period Start Position - 1

REG[4026h] LCD2 Horizontal Pulse Width Register 0 Default = 00h							Read/Write
LCD2 Horizontal Pulse Width bits 7-0							
7	6	5	4	3	2	1	0

REG[4027h] LCD2 Horizontal Pulse Width Register 1 Default = 00h							Read/Write
LCD2 Horizontal Pulse Polarity Select	n/a						LCD2 Horizontal Pulse Width bit 8
7	6	5	4	3	2	1	0

REG[4027h] bit 0
REG[4026h] bits 7-0 LCD2 Horizontal Pulse Width bits [8:0]
These bits specify the pulse width of the LCD2 horizontal sync signal (HSYNC), in pixel clock periods. For detailed timing information, see Section 7.6, "Panel Interface Timing" on page 95.

(REG[4027h] bit 0, REG[4026h] bits 7-0) = Horizontal Pulse Width - 1

bit 7 LCD2 Horizontal Pulse Polarity Select
This bit selects the polarity of the LCD2 horizontal sync signal (HSYNC).
When this bit = 0b, the horizontal sync signal (HSYNC) is active low. (default)
When this bit = 1b, the horizontal sync signal (HSYNC) is active high.

REG[4028h] LCD2 Horizontal Pulse Start Position Register 0 Default = 00h							Read/Write
LCD2 Horizontal Pulse Start Position bits 7-0							
7	6	5	4	3	2	1	0

REG[4029h] LCD2 Horizontal Pulse Start Position Register 1 Default = 00h							Read/Write
n/a				LCD2 Horizontal Pulse Start Position bits 11-8			
7	6	5	4	3	2	1	0

REG[4029h] bits 3-0
REG[4028h] bits 7-0 LCD2 Horizontal Pulse Start Position bits [11:0]
These bits specify the start position of the LCD2 horizontal sync pulse (HSYNC), in pixel clock periods. For detailed timing information, see Section 7.6, “Panel Interface Timing” on page 95.

(REG[4029h] bits 3-0, REG[4028h] bits 7-0) = Horizontal Pulse Start Position

REG[402Ah] LCD2 Vertical Total Register 0 Default = 00h							Read/Write
LCD2 Vertical Total bits 7-0							
7	6	5	4	3	2	1	0

REG[402Bh] LCD2 Vertical Total Register 1 Default = 00h							Read/Write
n/a				LCD2 Vertical Total bits 11-8			
7	6	5	4	3	2	1	0

REG[402Bh] bits 3-0
REG[402Ah] bits 7-0 LCD2 Vertical Total bits [11:0]
These bits specify the Vertical Total for LCD2, in lines. The Vertical Total is the sum of the Vertical Display Period and the Vertical Non-Display Period. For detailed timing information, see Section 7.6, “Panel Interface Timing” on page 95.

(REG[402Bh] bits 3-0, REG[402Ah] bits 7-0) = Vertical Total Period in lines - 1

REG[402Ch] LCD2 Vertical Display Period Register 0 Default = 00h Read/Write							
LCD2 Vertical Display Period bits 7-0							
7	6	5	4	3	2	1	0

REG[402Dh] LCD2 Vertical Display Period Register 1 Default = 00h Read/Write							
n/a				LCD2 Vertical Display Period bits 11-8			
7	6	5	4	3	2	1	0

REG[402Dh] bits 3-0
REG[402Ch] bits 7-0 LCD2 Vertical Display Period bits [11:0]

These bits specify the Vertical Display Period for LCD2, in lines. The Vertical Display Period must be less than the Vertical Total to allow for sufficient Vertical Non-Display Period. For detailed timing information, see Section 7.6, "Panel Interface Timing" on page 95.

(REG[402Dh] bits 3-0, REG[402Ch] bits 7-0) = Vertical Display Period in lines - 1

REG[402Eh] LCD2 Vertical Display Period Start Position Register 0 Default = 00h Read/Write							
LCD2 Vertical Display Period Start Position bits 7-0							
7	6	5	4	3	2	1	0

REG[402Fh] LCD2 Vertical Display Period Start Position Register 1 Default = 00h Read/Write							
n/a				LCD2 Vertical Display Period Start Position bits 11-8			
7	6	5	4	3	2	1	0

REG[402Fh] bits 3-0
REG[402Eh] bits 7-0 LCD2 Vertical Display Period Start Position bits [11:0]

These bits specify the Vertical Display Period Start Position for LCD2, in lines. For detailed timing information, see Section 7.6, "Panel Interface Timing" on page 95.

REG[402Fh] bits 3-0, REG[402Eh] bits 7-0) = Vertical Display Period Start Position in lines

REG[4030h] LCD2 Vertical Pulse Width Register Default = 00h Read/Write							
LCD2 Vertical Pulse Width bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 LCD2 Vertical Pulse Width bits [7:0]

These bits specify the pulse width of the LCD2 vertical sync signal (VSYNC), in lines. For detailed timing information, see Section 7.6, "Panel Interface Timing" on page 95.

REG[4030h] bits 7-0 = Vertical Pulse Width in lines - 1

Note

For EID Double Screen panels with TCON enabled (REG[4000h] bits 5-4 = 01b and REG[4040h] bit 0 = 1b), these bits should be set to 01h.

REG[4031h] LCD2 Vertical Pulse Polarity Register							
Default = 00h							Read/Write
LCD2 Vertical Pulse Polarity Select	n/a						
7	6	5	4	3	2	1	0

bit 7 LCD2 Vertical Pulse Polarity Select
This bit selects the polarity of the LCD2 vertical sync signal (VSYNC).
When this bit = 0b, the vertical sync signal (VSYNC) is active low. (default)
When this bit = 1b, the vertical sync signal (VSYNC) is active high.

REG[4032h] LCD2 Vertical Pulse Start Position Register 0							
Default = 00h							Read/Write
LCD2 Vertical Pulse Start Position bits 7-0							
7	6	5	4	3	2	1	0

REG[4033h] LCD2 Vertical Pulse Start Position Register 1							
Default = 00h							Read/Write
n/a				LCD2 Vertical Pulse Start Position bits 11-8			
7	6	5	4	3	2	1	0

REG[4033h] bits 3-0
REG[4032h] bits 7-0 LCD2 Vertical Pulse Start Position bits [11:0]
These bits specify the start position of the LCD2 vertical sync pulse (VSYNC), in lines.
For detailed timing information, see Section 7.6, “Panel Interface Timing” on page 95.
(REG[4033h] bits 3-0, REG[4032h] bits 7-0) = Vertical Pulse Start Position in lines

REG[4034h] LCD2 Serial Interface Configuration Register Default = 00h							Read/Write
LCD2 Serial Command Type bits 2-0			LCD2 Serial Command Direction	n/a		LCD2 Serial Clock Phase	LCD2 Serial Clock Polarity
7	6	5	4	3	2	1	0

bits 7-5 LCD2 Serial Command Type bits [2:0]
These bits determine the serial command type for LCD2. For AC timing information, see Section 7.6.2, “ND-TFD 8-Bit Serial Interface Timing” on page 101, Section 7.6.3, “ND-TFD 9-Bit Serial Interface Timing” on page 103, Section 7.6.4, “a-Si TFT Serial Interface Timing” on page 105, and Section 7.6.5, “uWIRE Serial Interface Timing” on page 106.

Table 10-90: LCD2 Serial Command Type Select

REG[4034h] bits 7-5	Serial Command Type Selected
000b	ND-TFT 4 pin Serial (8-bit serial data)
001b	ND-TFD 3 pin Serial (9-bit serial data)
010b	a-Si TFT Serial (8-bit serial data)
011b	Reserved
100b	µWire serial (16-bit serial data)
101b	24-bit serial data
110b - 111b	Reserved

bit 4 LCD2 Serial Command Direction
This bit determines the serial command bit direction for LCD2.
When this bit = 0b, the most significant bit is first. (default)
When this bit = 1b, the least significant bit is first.

Note
For details on timing, see Section 7.6, “Panel Interface Timing” on page 95 and refer to the appropriate serial interface.

bit 1 LCD2 Serial Clock Phase
This bit specifies the serial clock phase for LCD2. For a summary of the serial clock phase and polarity settings, see Table 10-91 “LCD2 Serial Clock Phase and Polarity,” on page 393.

Note
For details on timing, see Section 7.6, “Panel Interface Timing” on page 95 and refer to the appropriate serial interface.

bit 0 LCD2 Serial Clock Polarity
This bit specifies the serial clock polarity for LCD2. For a summary of the serial clock phase and polarity settings, see Table 10-91 "LCD2 Serial Clock Phase and Polarity," on page 393.

Table 10-91 : LCD2 Serial Clock Phase and Polarity

REG[4034h] bit 1	REG[4034h] bit 0	Valid Data	Clock Idling Status
0b	0b	Rising edge of Serial Clock	Low
	1b	Falling edge of Serial Clock	High
1b	0b	Falling edge of Serial Clock	Low
	1b	Rising edge of Serial Clock	High

REG[4035h] LCD2 Serial Interface Status Register Default = 00h							Read Only
LCD2 Serial Busy Status	n/a						
7	6	5	4	3	2	1	0

bit 7 LCD2 Serial Busy Status
This bit indicates the busy status of the LCD2 serial interface. While serial command/parameter data is being issued, this bit will return a 1b. After the data transfer is completed, it is cleared automatically. When this bit = 1b, the host interface cannot write to the LCD Serial Command/Parameter registers, REG[403Ah] ~ REG[403Dh].
When this bit = 0b, the LCD2 serial interface is ready (not busy).
When this bit = 1b, the LCD2 serial interface is busy.

REG[4036h] LCD2 Interface Status Register Default = 00h							Read Only
n/a							LCD2 VNDP Status
7	6	5	4	3	2	1	0

bit 0 LCD2 VNDP Status (Read Only)
This bit indicates whether the LCD2 panel is in a Vertical Display Period or a Vertical Non-Display Period. To use this bit, the configured VNDP for LCD2 must be greater than 1 line.
When this bit = 0b, the LCD2 panel output is in a Vertical Display Period.
When this bit = 1b, the LCD2 panel output is in a Vertical Non-Display Period.

Note
This bit is not set when the LCD2 output is disabled (REG[4070h] bit 0 = 0b) or LCD2 power save mode is enabled (REG[40B1h] bit 0 = 1b). When the LCD interface is disabled, or power save is enabled, this bit should be ignored.

REG[4037h] LCD2 VSYNC Register Default = 00h							Read/Write
LCD2 VSYNC Interrupt Enable	n/a			LCD2 VSYNC Interrupt Status	Reserved	n/a	

REG[4037h] LCD2 VSYNC Register Default = 00h							Read/Write
7	6	5	4	3	2	1	0

bit 7 LCD2 VSYNC Interrupt Enable
This bit controls whether a LCD2 VSYNC Interrupt will cause a LCD2 Interrupt to occur in REG[0A00h] bit 1.
When this bit = 0b, the LCD2 VSYNC Interrupt status is not output to the Interrupt Controller.
When this bit = 1b, the LCD2 VSYNC Interrupt status is output to the Interrupt Controller.

bit 3 LCD2 VSYNC Interrupt Status
This bit indicates the status of the VSYNC Interrupt for LCD2. This bit is masked by the LCD2 VSYNC Interrupt Mask Disable bit and is not available when REG[4037h] bit 2 = 0b. When this bit indicates that a LCD2 VSYNC Interrupt has occurred and the LCD2 VSYNC Interrupt Enable bit is set (REG[4037h] bit 7 = 1b), a LCD2 Interrupt occurs in REG[0A00h] bit 1.
When this bit = 0b, a LCD2 VSYNC Interrupt has not occurred.
When this bit = 1b, a LCD2 VSYNC Interrupt has occurred.

To clear this status bit, write a 1b to this bit.

bit 2 Reserved
This bit MUST be set to 1b for normal operation.

REG[4038h] LCD2 VSYNC Interrupt Delay Register 0 Default = 00h							Read/Write
LCD2 VSYNC Interrupt Delay bits 7-0							
7	6	5	4	3	2	1	0

REG[4039h] LCD2 VSYNC Interrupt Delay Register 1 Default = 00h							Read/Write
n/a				LCD2 VSYNC Interrupt Delay bits 11-8			
7	6	5	4	3	2	1	0

REG[4039h] bits 3-0
REG[4038h] bits 7-0 LCD2 VSYNC Interrupt Delay bits [11:0]

These bits specify the VSYNC interrupt assertion timing delay from the start of the frame, in lines. If the interrupt delay is greater than the Vertical Total for LCD2 (see REG[402Ah] ~ REG[402Bh]), the interrupt will not occur.

REG[403Ah] LCD2 Serial Data Register 0 Default = 00h							Read/Write
LCD2 Serial Data bits 7-0							
7	6	5	4	3	2	1	0

REG[403Bh] LCD2 Serial Data Register 1 Default = 00h							Read/Write
LCD2 Serial Data bits 15-8							
7	6	5	4	3	2	1	0

REG[403Dh] LCD2 Serial Data Register 2 Default = 00h							Read/Write
LCD2 Serial Data bits 23-16							
7	6	5	4	3	2	1	0

REG[403Dh] bits 7-0
REG[403Bh] bits 7-0
REG[403Ah] bits 7-0 LCD2 Serial Data bits [23:0]

These bits specify the data for the LCD2 serial interface. The serial data is issued to the panel module once REG[403Bh] is written.

Table 10-92: LCD2 Serial Data

Register	24-bit Serial Interface	uWire Serial Interface	ND-TFD 8-bit Serial Interface, ND-TFD 9-bit Serial Interface, a-Si TFT Serial Interface
REG[403Ah]	Data bits 7-0	Data bits 7-0	Data bits 7-0
REG[403Bh]	Data bits 15-8	Data bits 15-8	Bit 0 is output as the signal A0 and is only used to determine whether the LCD Serial data bits 7-0 (REG[401Ch]) contain a command or parameter.
REG[403Dh]	Data bits 23-16	n/a	n/a

REG[4040h] EID Double Screen Panel Configuration Register 0 Default = 00h							Read/Write
n/a			Reserved	n/a			Double Screen Panel Timing Controller Enable
7	6	5	4	3	2	1	0

bit 4 Reserved
This bit must be set to 0b.

bit 0 Double Screen Panel Timing Controller Enable
This bit controls the S1D13515/S2D13515 internal timing controller for EID Double Screen panels. When the EID Double Screen panel has the timing controlled built-in, this bit should be set to 0b (disabled).
When this bit = 0b, the timing controller is disabled.
When this bit = 1b, the timing controller is enabled.

Note
When LCD2 is an EID Doublescreen with TCON disabled, FP2IO[23:18] is driven LOW.

REG[4041h] EID Double Screen Panel Configuration Register 1 Default = 10h							Read/Write
n/a			Double Screen Panel Input Polarity	n/a			Reserved
7	6	5	4	3	2	1	0

bit 4 Double Screen Panel Input Polarity
This bit controls the active state of the HSYNC and VSYNC signals to the EID Double Screen panel timing controller (TCON).
When this bit = 0b, the input polarity of HSYNC and VSYNC is active-low.
When this bit = 1b, the input polarity of HSYNC and VSYNC is active-high. (default)

bit 0 Reserved
This bit must be set to 0b.

REG[4042h] EID Double Screen Panel REV Signal Register 0 Default = 11h							Read/Write
VREVOUT Configuration	n/a		VREVOUT Polarity	HREVOUT Configuration	n/a		HREVOUT Polarity
7	6	5	4	3	2	1	0

bit 7 VREVOUT Configuration
This bit determines whether the EID Double Screen panel VREVOUT signal (FP2IO22 pin) is driven High or Low. It also determines whether the FLMF or FLMB signal is output on the FP2IO25 pin.
When this bit = 0b, the VREVOUT signal is driven High and the FLMF signal is output on the FP2IO25 pin.
When this bit = 1b, the VREVOUT signal is driven Low and the FLMB signal is output on FP2IO25 pin.

The output state of VREVOUT is affected by the VREVOUT Polarity bit, REG[4042h] bit 4. The following table summarizes the possible configurations.

Table 10-93 : VREVOUT and FLMF/FLMB Configuration Summary

REG[4042h] bit 7 (VREVOUT Configuration)	REG[4042h] bit 4 (VREVOUT Polarity)	VREVOUT (FP2IO22)	FLMF/FLMB (FP2IO25)
0b	0b	High	FLMF (active High)
	1b	Low	FLMF (active High)
1b	0b	Low	FLMB (active High)
	1b	High	FLMB (active High)

bit 4 VREVOUT Polarity
This bit selects the polarity of the EID Double Screen panel VREVOUT signal output on the FP2IO22 pin. For a summary of the possible configurations, see Table 10-93 “VREVOUT and FLMF/FLMB Configuration Summary,” on page 397.
When this bit = 0b, the VREVOUT signal is normal.
When this bit = 1b, the VREVOUT signal is inverted. (default)

bit 3 HREVOUT Configuration
This bit determines whether the EID Double Screen panel HREVOUT signal (FP2IO23 pin) is driven High or Low. It also determines whether the EISF or EISB signal is output on the FP2IO24 pin.
When this bit = 0b, the HREVOUT signal is driven High and the EISF signal is output on the FP2IO24 pin.
When this bit = 1b, the HREVOUT signal is driven Low and the EISB signal is output on the FP2IO24 pin.

The output state of HREVOUT is affected by the HREVOUT Polarity bit, REG[4042h] bit 0. The following table summarizes the possible configurations.

Table 10-94 : HREVOUT and EISF/EISB Configuration Summary

REG[4042h] bit 3 (HREVOUT Configuration)	REG[4042h] bit 0 (HREVOUT Polarity)	HREVOUT (FP2IO23)	EISF/EISB (FP2IO24)
0b	0b	High	EISF (active High)
	1b	Low	EISF (active High)
1b	0b	Low	EISB (active High)
	1b	High	EISB (active High)

bit 0 HREVOUT Polarity
This bit selects the polarity of the EID Double Screen panel HREVOUT signal output on the FP2IO23 pin. For a summary of the possible configurations, see Table 10-94 “HREVOUT and EISF/EISB Configuration Summary,” on page 397.
When this bit = 0b, the HREVOUT signal is normal.
When this bit = 1b, the HREVOUT signal is inverted. (default)

REG[4043h] EID Double Screen Panel REV Signal Register 1 Default = 00h							Read/Write
n/a			Data Toggle Reduction	n/a			HREVOUT Data Select
7	6	5	4	3	2	1	0

bit 4 Data Toggle Reduction
This bit selects low EMI mode for the EID Double Screen panel. Low EMI mode is achieved by reducing RGB data toggle using the DEXR signal, pin FP1IO14.
When this bit = 0b, normal RGB data toggling is selected.
When this bit = 1b, reduced RGB data toggling is selected (Low EMI Mode).

bit 0 HREVOUT Data Select
This bit works in conjunction with the HREVOUT Configuration bit (REG[4041h] bit 3) to determine the RGB data direction for the EID Double Screen panel.

Table 10-95 : HREVOUT Data Selection

REG[4043h] bit 0	REG[4042h] bit 3	RGB Data Direction
0b	0b	RGB data is output normally
	1b	RGB data is output normally
1b	0b	R and B components are swapped
	1b	RGB data is output normally

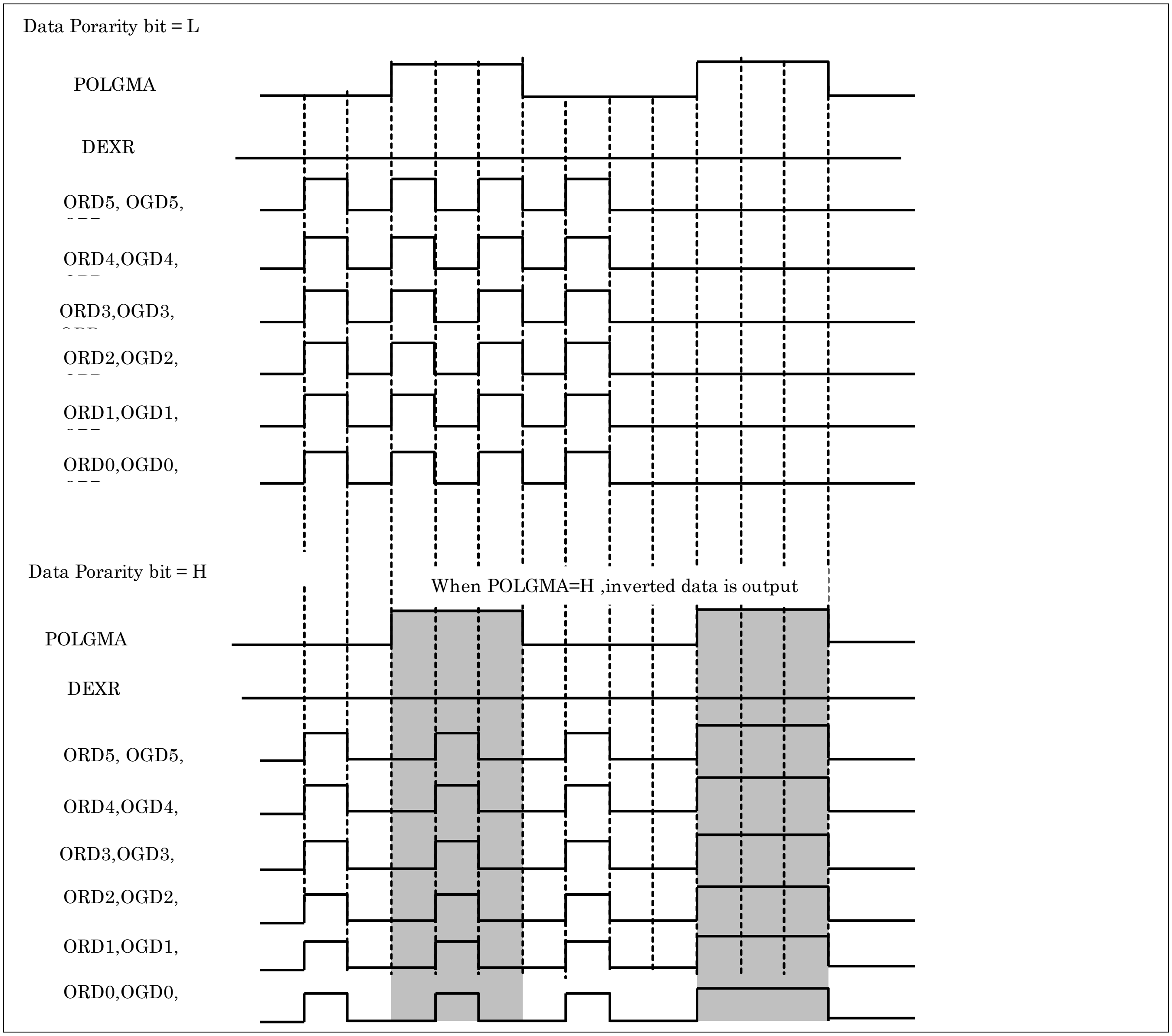
REG[4044h] EID Double Screen Panel Data Out Mode Register Default = 00h							Read/Write
n/a		Blank Mask bits 1-0		n/a			Data Polarity
7	6	5	4	3	2	1	0

bits 5-4 — Blank Mask bits [1:0]
These bits select the mask data during the blanking period (non display period) of the EID Double Screen panel.

Table 10-96: Blank Mask Select

REG[4044h] bits 5-4	Blank Mask Selected
00b ~ 01b	Black (00h)
10b	White (3Fh)
11b	Gray (1Fh)

bit 0 — Data Polarity
This bit selects the data polarity for the EID Double Screen panel.
When this bit = 0b, the data polarity is normal.
When this bit = 1b, dot inversion is enabled.



REG[4046h] EID Double Screen Panel OE Signal Register 0 Default = 00h							Read/Write
OE Signal Low Width bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 OE Signal Low Width bits [7:0]
These bits function differently based on the setting of the Special Drive Mode bit, REG[4049h] bit 0.
When Special Drive Mode is disabled (REG[4049h] bit 0 = 0b), these bits set the OE Low Width for EID Double Screen panels (FP1IO18 pin) which is defined from the OE rising edge to the STRB rising edge (0-255 clocks).
When Special Drive Mode is enabled (REG[4049h] bit 0 = 1b), these bits set the CPV Low Period width which is defined from the rising edge to the STRB rising edge (1-256 clocks).

REG[4047h] EID Double Screen Panel OE Signal Register 1 Default = 00h							Read/Write
OE Signal Invert	n/a						
7	6	5	4	3	2	1	0

bit 7 OE Signal Invert
These bits work in conjunction with the Special Drive Mode bit (REG[4049h] bit 0) to control the active polarity of the OE signal for the EID Double Screen panel, pin FP1IO18.

Table 10-97 : OE Signal Configuration

REG[4047h] bit 7	REG[4049h] bit 0	OE Signal Output
0b	0b	Active Low
	1b	Low Fixed
1b	0b	Active High
	1b	High Fixed

REG[4048h] EID Double Screen Panel Drive Mode Register 0 Default = 00h							Read/Write
n/a						Panel Drive Polarity Mode bits 1-0	
7	6	5	4	3	2	1	0

bits 1-0 Panel Drive Polarity Mode bits [1:0]
These bits select the EID Double Screen panel drive (voltage) polarity.

Table 10-98: Drive Polarity Mode Selection

REG[4048h] bits 1-0	Drive Polarity Selected
00b	1H Inversion
01b	0.5H Inversion
10b	1V Inversion
11b	2H Inversion

REG[4049h] EID Double Screen Panel Drive Mode Register 1 Default = 00h							Read/Write
n/a							Special Drive Mode
7	6	5	4	3	2	1	0

bit 0 Special Drive Mode
This bit selects the drive mode for the EID Double Screen panel.
When this bit = 0b, normal drive mode is selected.
When this bit = 1b, special drive mode is selected.

REG[404Ah] EID Double Screen Panel POLGMA Timing Register Default = 00h Read/Write							
0.5H Drive POLGMA Signal Toggle Position bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 0.5H Drive POLGMA Signal Toggle Position bits [7:0]
These bits only have an effect for 0.5H Inversion, REG[4048h] bits 1-0 = 01b.
These bits are used to change the toggle position of the POLGMA signal by ± 0 ~ 127 clocks. Bit 7 is the sign (direction) bit and bits 6-0 set the number of clocks for positioning.

REG[404Ch] is Reserved

This register is Reserved and should not be written.

REG[404Eh] EID Double Screen Panel Backlight LED Control Register 0 Default = 00h Read/Write							
Duty Control bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 Duty Control bits [7:0]
These bits control the duty cycle of the backlight LED for the EID Double Screen panel.

REG[404Fh] EID Double Screen Panel Backlight LED Control Register 1 Default = 00h Read/Write							
Frequency Control bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 Frequency Control bits [7:0]
These bits control the frequency of the backlight LED for the EID Double Screen panel.

REG[4050h] Sharp DualView Panel Mirror Mode Register Default = 00h Read/Write							
n/a							Scanning Direction
7	6	5	4	3	2	1	0

bit 0 Scanning Direction
This bit controls the scanning direction for Sharp DualView panels.
When this bit = 0b, SPL is output on FP2IO21.
When this bit = 1b, SPR is output on FP2IO20.

REG[4052h] Sharp DualView Panel CLS Pulse Width Register 0 Default = 00h Read/Write							
CLS Pulse Width bits 7-0							
7	6	5	4	3	2	1	0

REG[4053h] Sharp DualView Panel CLS Pulse Width Register 1 Default = 00h					Read/Write		
n/a					CLS Pulse Width bits 10-8		
7	6	5	4	3	2	1	0

REG[4053h] bits 2-0
REG[4052h] bits 7-0 Sharp DualView Panel CLS Pulse Width bits [10:0]
These bits specify the CLS pulse width for Sharp DualView panels, in pixel clock periods.
CLS Pulse Width = REG[4053h] bits 2-0, REG[4052h] bits 7-0

REG[4054h] Sharp DualView Panel VCOM Toggle Point Register Default = 00h							Read/Write
n/a	VCOM Toggle Point Control bits 6-0						
7	6	5	4	3	2	1	0

bits 6-0 Sharp DualView Panel VCOM Toggle Point Control bits [6:0]
These bits specify the VCOM/VCOMB toggle position for Sharp DualView panels, in pixel clock periods.
VCOM/VCOMB Toggle Position = REG[4054h] bits 6-0

REG[4056h] Sharp DualView Panel LS Delay Register Default = 00h							Read/Write
LS Delay bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 Sharp DualView Panel LS Delay bits [7:0]
These bits specify the LS (horizontal pulse) start position for Sharp DualView panels, in pixel clock periods.
LS Delay = REG[4056h] bits 7-0

REG[4060h] LCD1 Display Mode Register 0 Default = 00h							Read/Write
LCD1 Software Reset	LCD1 Display Blank	LCD1 Video Invert	n/a		LCD1 Output Status	n/a	LCD1 Output Enable
7	6	5	4	3	2	1	0

bit 7 LCD1 Software Reset
This bit initiates a software reset of the LCD1 module which resets all registers associated with LCD1 and CH1IN to their default values, resets the CH1IN display pipe, and sets all LCD1 pins to their reset states.
Writing a 0b to this bit has no hardware effect.
Writing a 1b to this bit performs a software reset of the LCD1 module.

bit 6 LCD1 Display Blank
This bit blanks the LCD1 display by forcing all display data outputs low (or high). All display control signals remain unchanged.
When this bit = 0b, the LCD1 display is active.
When this bit = 1b, the LCD1 display is blanked and all data outputs are forced low or high depending on the setting of the LCD1 Video Invert bit (REG[4060h] bit 5).

Table 10-99 : LCD1 Data Output Selection

REG[4060h] bit 6	REG[4060h] bit 5	LCD1 Data Output
0b	0b	Normal
	1b	Inverted
1b	0b	Forced Low
	1b	Forced High

bit 5 LCD1 Video Invert
This bit determines whether the LCD1 display data output is inverted or left unchanged (normal). This bit has an effect when the display is active and when the display is blanked (see REG[4060h] bit 6).
When this bit = 0b, the LCD1 display data is unchanged (normal).
When this bit = 1b, the LCD1 display data is inverted.

bit 2 LCD1 Output Status
This bit indicates whether the S1D13515/S2D13515 is outputting to the LCD1 interface.
When this bit = 0b, LCD1 output is not active.
When this bit = 1b, LCD1 output is active.

Note
When LCD1 power save mode is enabled, REG[40B0h] bit 0 = 1b, REG[4060h] bit 2 is invalid and should be ignored.

bit 0 LCD1 Output Enable
This bit controls whether the LCD1 control signals and display data are output on the LCD1 interface.
When this bit = 0b, LCD1 output is disabled.
When this bit = 1b, LCD1 output is enabled.

REG[4062h] LCD1 Display Mode Register 1 Default = 00h							Read/Write
n/a				Reserved	CH1IN Pixel Format bits 2-0		
7	6	5	4	3	2	1	0

bit 3 Reserved
This bit must be set to 0b.

bits 2-0 CH1IN Pixel Format bits [2:0]
These bits select the color depth, in bits-per-pixel, for CH1IN. The CH1IN color depth must be set according to the configuration of the input source (i.e. CH1OUT, Image Fetcher, or Warp Logic).

Table 10-100: CH1IN Pixel Format Selection

REG[4062h] bits 2-0	CH1IN Pixel Format
000b	RGB 3:3:2
001b	RGB 5:6:5
010b	RGB 8:8:8
011b ~ 111b	Reserved

REG[4064h] CH1IN FIFO Threshold Register Default = 7Fh							Read/Write
n/a	CH1IN FIFO Threshold bits 6-0						
7	6	5	4	3	2	1	0

bits 6-0 CH1IN FIFO Threshold bits [6:0]
These bits specify the CH1IN FIFO Threshold. When the difference between the CH1IN FIFO read and write pointer is less than the value specified by these bits, a memory read request is generated.

REG[4065h] CH1IN FIFO Empty Status Register Default = 00h							Read/Write
CH1IN FIFO Empty Status	n/a						
7	6	5	4	3	2	1	0

bit 7 CH1IN FIFO Empty Status
This bit indicates the empty status of the CH1IN FIFO. The CH1IN FIFO becomes empty when a CH1IN FIFO underflow occurs.
When this bit = 0b, the CH1IN FIFO is not empty.
When this bit =1b, the CH1IN FIFO is empty.

To clear this status bit, write a 1b to this bit.

REG[4070h] LCD2 Display Mode Register 0 Default = 00h							Read/Write
LCD2 Software Reset	LCD2 Display Blank	LCD2 Video Invert	n/a		LCD2 Output Status	n/a	LCD2 Output Enable
7	6	5	4	3	2	1	0

bit 7 LCD2 Software Reset
This bit initiates a software reset of the LCD2 module which resets all registers associated with LCD2 and CH2IN/OSDIN to their default values, resets the CH2IN/OSDIN display pipes, and sets all LCD2 pins to their reset states.
Writing a 0b to this bit has no hardware effect.
Writing a 1b to this bit performs a software reset of the LCD2 module.

bit 6 LCD2 Display Blank
This bit blanks the LCD2 display by forcing all display data outputs low (or high). All display control signals remain unchanged.
When this bit = 0b, the LCD2 display is active.
When this bit = 1b, the LCD2 display is blanked and all data outputs are forced low or high depending on the setting of the LCD2 Video Invert bit (REG[4070h] bit 5).

Table 10-101 : LCD2 Data Output Selection

REG[4070h] bit 6	REG[4070h] bit 5	LCD2 Data Output
0b	0b	Normal
	1b	Inverted
1b	0b	Forced Low
	1b	Forced High

bit 5 LCD2 Video Invert
This bit determines whether the LCD2 display data output is inverted or left unchanged (normal). This bit has an effect when the display is active and when the display is blanked (see REG[4070h] bit 6).
When this bit = 0b, the LCD2 display data is unchanged (normal).
When this bit = 1b, the LCD2 display data is inverted.

bit 5 LCD2 Video Invert
This bit inverts the display by inverting all display data outputs. All display control signals remain unchanged. This bit has no effect if display blank is enabled, REG[4070h] bit 6 = 1b.
When this bit = 0b, the display data is unchanged (normal).
When this bit = 1b, the display data is inverted.

bit 2 LCD2 Output Status
This bit indicates whether the S1D13515/S2D13515 is outputting to the LCD2 interface.
When this bit = 0b, LCD1 output is not active.
When this bit = 1b, LCD1 output is active.

Note
When LCD2 power save mode is enabled, REG[40B1h] bit 0 = 1b, REG[4070h] bit 2 is invalid and should be ignored.

bit 0 LCD2 Output Enable
This bit controls whether the LCD2 control signals and display data are output on the LCD2 interface.
When this bit = 0b, LCD1 output is disabled.
When this bit = 1b, LCD1 output is enabled.

REG[4072h] LCD2 Display Mode Register 1 Default = 00h Read/Write							
n/a				Reserved	CH2IN Pixel Format bits 2-0		
7	6	5	4	3	2	1	0

bit 3 — Reserved
This bit must be set to 0b.

bits 2-0 — CH2IN Pixel Format bits [2:0]
These bits select the color depth, in bits-per-pixel, for CH2IN. The CH2IN color depth must be set according to the configuration of the input source (i.e. CH2OUT or CH1OUT).

Table 10-102: CH2IN Pixel Format Selection

REG[4072h] bits 2-0	CH2IN Pixel Format
000b	RGB 3:3:2
001b	RGB 5:6:5
010b	RGB 8:8:8
011b ~ 111b	Reserved

REG[4073h] LCD2 Display Mode Register 2 Default = 00h							Read/Write
Reserved		EID Double Screen Mode bits 1-0		Reserved	OSD Pixel Format bits 2-0		
7	6	5	4	3	2	1	0

bits 7-6 Reserved
The default value for these bits is 00b.bits 5-4EID Double Screen Mode bits [1:0]
These bits select the double screen mode for the EID Double Screen panel.

Table 10-103: EID Double Screen Mode Selection

REG[4073h] bits 5-4	EID Double Screen Mode	
	Left	Right
00b	CH2IN	CH2IN
01b	OSDIN	OSDIN
10b	OSDIN	CH2IN
11b	CH2IN	OSDIN

bit 3 Reserved
This bit must be set to 0b.

bits 2-0 OSDIN Pixel Format bits [2:0]
These bits select the color depth, in bits-per-pixel, for OSDIN. The OSDIN color depth must be set according to the configuration of the input source (i.e. OSDOUT or CH1OUT).

Table 10-104: OSDIN Pixel Format Selection

REG[4073h] bits 2-0	OSDIN Pixel Format
000b	RGB 3:3:2
001b	RGB 5:6:5
010b	RGB 8:8:8
011b ~ 111b	Reserved

REG[4074h] CH2IN FIFO Threshold Register Default = 7Fh							Read/Write
n/a	CH2IN FIFO Threshold bits 6-0						
7	6	5	4	3	2	1	0

bits 6-0 CH2IN FIFO Threshold bits [6:0]
These bits specify the CH2IN FIFO Threshold. When the difference between the CH2IN FIFO read and write pointer is less than the value specified by these bits, a memory read request is generated.

REG[4075h] CH2IN FIFO Empty Status Register Default = 00h							Read/Write
CH2IN FIFO Empty Status	n/a						
7	6	5	4	3	2	1	0

bit 7 CH2IN FIFO Empty Status
This bit indicates the empty status of the CH2IN FIFO. The CH2IN FIFO becomes empty when a CH2IN FIFO underflow occurs.
When this bit = 0b, the CH2IN FIFO is not empty.
When this bit =1b, the CH2IN FIFO is empty.

To clear this status bit, write a 1b to this bit.

REG[4076h] OSDIN FIFO Threshold Register Default = 7Fh							Read/Write
n/a	OSDIN FIFO Threshold bits 6-0						
7	6	5	4	3	2	1	0

bits 6-0 OSDIN FIFO Threshold bits [6:0]
These bits specify the OSDIN FIFO Threshold. When the difference between the OSDIN FIFO read and write pointer is less than the value specified by these bits, a memory read request is generated.

REG[4077h] OSDIN FIFO Empty Status Register Default = 00h							Read/Write
OSDIN FIFO Empty Status	n/a						
7	6	5	4	3	2	1	0

bit 7 OSDIN FIFO Empty Status
This bit indicates the empty status of the OSDIN FIFO. The OSDIN FIFO becomes empty when a OSDIN FIFO underflow occurs.
When this bit = 0b, the OSDIN FIFO is not empty.
When this bit =1b, the OSDIN FIFO is empty.

To clear this status bit, write a 1b to this bit.

REG[4078h] through REG[407Fh] are Reserved

These registers are Reserved and should not be written.

REG[4080h] LCD1 Bias/Gain Control Register Default = 00h							Read/Write
n/a							LCD1 Bias/Gain Enable
7	6	5	4	3	2	1	0

bit 0 LCD1 Bias/Gain Enable
This bit controls the luminance and contrast (dynamic range) of each RGB component. This function can be used for any panel type. The bias settings for each RGB component are set in REG[4082h] ~ REG[4087h] and the gain settings are set in REG[4088h] ~ REG[408Ch].
When this bit = 0b = LCD1 bias/gain is disabled.
When this bit = 1b = LCD1 bias/gain is enabled.

For each color channel, the formula is:
output = (original image + bias) × gain factor

The bias value ranges from -256 to +255. After the bias, before the gain, the data is clipped to 0 ~ 255.

REG[4082h] LCD1 Bias Red Register 0 Default = 00h							Read/Write
LCD1 Bias Red bits 7-0							
7	6	5	4	3	2	1	0

REG[4083h] LCD1 Bias Red Register 1 Default = 00h							Read/Write
n/a							LCD1 Bias Red bit 8
7	6	5	4	3	2	1	0

REG[4083h] bit 0
REG[4082h] bits 7-0 LCD1 Bias Red bits [8:0]
When REG[4080h] bit 0 = 1b, these bits set the red luminance rate for LCD1. These bits form a signed, 2’s complement value ranging from -256 to +255.

REG[4084h] LCD1 Bias Green Register 0 Default = 00h							Read/Write
LCD1 Bias Green bits 7-0							
7	6	5	4	3	2	1	0

REG[4085h] LCD1 Bias Green Register 1 Default = 00h							Read/Write
n/a							LCD1 Bias Green bit 8
7	6	5	4	3	2	1	0

REG[4085h] bit 0
REG[4084h] bits 7-0 LCD1 Bias Green bits [8:0]
When REG[4080h] bit 0 = 1b, these bits set the green luminance rate for LCD1. These bits form a signed, 2’s complement value ranging from -256 to +255.

REG[4086h] LCD1 Bias Blue Register 0 Default = 00h							Read/Write
LCD1 Bias Blue bits 7-0							
7	6	5	4	3	2	1	0

REG[4087h] LCD1 Bias Blue Register 1 Default = 00h							Read/Write
n/a							LCD1 Bias Blue bit 8
7	6	5	4	3	2	1	0

REG[4087h] bit 0
REG[4086h] bits 7-0 LCD1 Bias Blue bits [8:0]
When REG[4080h] bit 0 = 1b, these bits set the blue luminance rate for LCD1. These bits form a signed, 2's complement value ranging from -256 to +255.

REG[4088h] LCD1 Gain Red Register Default = 40h							Read/Write
LCD1 Gain Red bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 LCD1 Gain Red bits [7:0]
When REG[4080h] bit 0 = 1b, these bits set the red contrast rate for LCD1. Bits 7-6 are the integer part and bits 5-0 are the fractional part of the red gain.

REG[408Ah] LCD1 Gain Green Register Default = 40h							Read/Write
LCD1 Gain Green bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 LCD1 Gain Green bits [7:0]
When REG[4080h] bit 0 = 1b, these bits set the green contrast rate for LCD1. Bits 7-6 are the integer part and bits 5-0 are the fractional part of the green gain.

REG[408Ch] LCD1 Gain Blue Register Default = 40h							Read/Write
LCD1 Gain Blue bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 LCD1 Gain Blue bits [7:0]
When REG[4080h] bit 0 = 1b, these bits set the blue contrast rate for LCD1. Bits 7-6 are the integer part and bits 5-0 are the fractional part of the blue gain.

REG[4090h] LCD2 Bias/Gain Control Register Default = 00h							Read/Write
n/a							LCD2 Bias/Gain Enable
7	6	5	4	3	2	1	0

bit 0 LCD2 Bias/Gain Enable
This bit controls the luminance and contrast (dynamic range) of each RGB component. This function can be used for any panel type. The bias settings for each RGB component are set in REG[4092h] ~ REG[4097h] and the gain settings are set in REG[4098h] ~ REG[409Ch].
When this bit = 0b = LCD2 bias/gain is disabled.
When this bit = 1b = LCD2 bias/gain is enabled.

For each color channel, the formula is:
output = (original image + bias) × gain factor

The bias value ranges from -256 to +255. After the bias, before the gain, the data is clipped to 0 ~ 255.

REG[4092h] LCD2 Bias Red Register 0 Default = 00h							Read/Write
LCD2 Bias Red bits 7-0							
7	6	5	4	3	2	1	0

REG[4093h] LCD2 Bias RED Register 1 Default = 00h							Read/Write
n/a							LCD2 Bias Red bit 8
7	6	5	4	3	2	1	0

REG[4093h] bit 0
REG[4092h] bits 7-0 LCD2 Bias Red bits [8:0]
When REG[4080h] bit 0 = 1b, these bits set the red luminance rate for LCD2. These bits form a signed, 2's complement value ranging from -256 to +255.

REG[4094h] LCD2 Bias Green Register 0 Default = 00h							Read/Write
LCD2 Bias Green bits 7-0							
7	6	5	4	3	2	1	0

REG[4095h] LCD2 Bias Green Register 1 Default = 00h							Read/Write
n/a							LCD2 Bias Green bit 8
7	6	5	4	3	2	1	0

REG[4095h] bit 0
REG[4094h] bits 7-0 LCD2 Bias Green bits [8:0]
When REG[4080h] bit 0 = 1b, these bits set the green luminance rate for LCD2. These bits form a signed, 2's complement value ranging from -256 to +255.

REG[4096h] LCD2 Bias Blue Register 0 Default = 00h							Read/Write
LCD2 Bias Blue bits 7-0							
7	6	5	4	3	2	1	0

REG[4097h] LCD2 Bias Blue Register 1 Default = 00h							Read/Write
n/a							LCD2 Bias Blue bit 8
7	6	5	4	3	2	1	0

REG[4097h] bit 0
REG[4096h] bits 7-0 LCD2 Bias Blue bits [8:0]
When REG[4080h] bit 0 = 1b, these bits set the blue luminance rate for LCD2. These bits form a signed, 2's complement value ranging from -256 to +255.

REG[4098h] LCD2 Gain Red Register Default = 40h							Read/Write
LCD2 Gain Red bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 LCD2 Gain Red bits [7:0]
When REG[4080h] bit 0 = 1b, these bits set the red contrast rate for LCD2. Bits 7-6 are the integer part and bits 5-0 are the fractional part of the red gain.

REG[409Ah] LCD2 Gain Green Register Default = 40h							Read/Write
LCD2 Gain Green bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 LCD2 Gain Green bits [7:0]
When REG[4080h] bit 0 = 1b, these bits set the green contrast rate for LCD2. Bits 7-6 are the integer part and bits 5-0 are the fractional part of the green gain.

REG[409Ch] LCD2 Gain Blue Register Default = 40h							Read/Write
LCD2 Gain Blue bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 LCD2 Gain Blue bits [7:0]
When REG[4080h] bit 0 = 1b, these bits set the blue contrast rate for LCD2. Bits 7-6 are the integer part and bits 5-0 are the fractional part of the blue gain.

REG[40A0h] LCD2 Gamma LUT Data Port Default = 00h							Write Only
LCD2 Gamma LUT Data Port bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 LCD2 Gamma LUT Data Port bits [7:0] (Write Only)
These bits are the data port for writing to the LCD2 Gamma look-up table (LUT). The Gamma LUT Write Access Enable bit must be set (REG[40A2h] bit 4 = 1b) before writing to the LUT.

REG[40A2h] LCD2 Gamma LUT Configuration Register 0 Default = 00h							Write Only
Gamma LUT Write Color Select bits 1-0		n/a	Gamma LUT Write Access Enable	n/a	Gamma LUT Display Bank Select bits 1-0		Gamma LUT Correction Display Enable
7	6	5	4	3	2	1	0

bits 7-6 — Gamma LUT Write Color Select bits [1:0]
These bits select which RGB color component of the Gamma LUT will be written to using the data port in REG[40A0h]. Each color component can be selected individually or all color components can be programmed simultaneously with the same value. Before writing to the Gamma LUT, the Gamma LUT Write Access Enable bit must be set (REG[40A2h] bit 4 = 1b).

Table 10-105: Gamma LUT Write Color Selection

REG[40A2h] bits 7-6	Color Component
00b	Red color component will be written
01b	Green color component will be written
10b	Blue color component will be written
11b	All color components (RGB) will be written

bit 4 — Gamma LUT Write Access Enable
This bit controls whether write accesses to the LCD2 Gamma LUT using REG[40A0h] are allowed.
When this bit = 0b, write access to the LCD2 Gamma LUT is disabled.
When this bit = 1b, write access to the LCD2

bits 2-1 — Gamma LUT Display Bank Select bits [1:0]
These bits determines which Gamma LUT Bank and Segment are used when Gamma LUT Correction is enabled, REG[40A2h] bit 0 = 1b. These bits also select which Gamma LUT Bank can be written to by the Host. When either Bank A1 or A2 is selected (bits = 00b or 01b), Banks B1 and B2 can be programmed. When either Bank B1 or B2 is selected (bits = 10b or 11b), Banks A1 and B2 can be programmed.

Table 10-106: Gamma LUT Display Bank Selection

REG[40A2h] bits 2-1	Active Gamma LUT Bank/Segment
00b	Bank A1 is used for Gamma LUT Correction
01b	Bank A2 is used for Gamma LUT Correction
10b	Bank B1 is used for Gamma LUT Correction
11b	Bank B2 is used for Gamma LUT Correction

bit 0 — Gamma LUT Correction Display Enable
This bit controls whether the Gamma LUT Correction function has an effect on LCD2. The actual Gamma LUT Bank and Segment used for gamma correction is determined by the Gamma LUT Display Bank Select bits, REG[40A2h] bits 2-1.
When this bit = 0b, Gamma LUT Correction is disabled.
When this bit = 1b, Gamma LUT Correction is enabled.

REG[40A3h] LCD2 Gamma LUT Configuration Register 1 Default = 00h							Write Only
n/a							Gamma LUT Index Reset
7	6	5	4	3	2	1	0

bit 0 Gamma LUT Index Reset (Write Only)
When the Gamma LUT is being programmed, the internal LUT index is automatically incremented for each write to the data port (REG[40A0h]). This bit manually resets the index into the LCD2 Gamma LUT to 000h.
Writing a 0b to this bit has no effect.
Writing a 1b to this bit resets the Gamma LUT index.

Note
The Gamma LUT index is also reset when Gamma LUT Correction is enabled, REG[40A2h] bit 0 = 1b.

REG[40B0h] LCD1 Power Save Register Default = 00h							Read/Write
n/a							LCD1 Power Save Mode Enable
7	6	5	4	3	2	1	0

bit 0 LCD1 Power Save Mode Enable
This bit controls power save mode for LCD1.
When this bit = 0b, LCD1 is in normal mode (running).
When this bit = 1b, LCD1 is in power save mode (turned off).

REG[40B1h] LCD2 Power Save Register Default = 00h							Read/Write
n/a							LCD2 Power Save Mode Enable
7	6	5	4	3	2	1	0

bit 0 LCD2 Power Save Mode Enable
This bit controls power save mode for LCD2.
When this bit = 0b, LCD2 is in normal mode (running).
When this bit = 1b, LCD2 is in power save mode (turned off).

10.4.22 Sprite Registers

The S1D13515/S2D13515 Sprite Engine has two types of registers.

1. General control registers for the Sprite Engine which control the Sprite Engine itself and settings common for all sprites. These registers are from REG[5000h] ~ REG[502Bh] and are discussed in the following section.

2. Sprite specific registers which specify settings for each individual sprite (Sprite #0-#7). These registers are SDRAM based registers that are stored in a portion of SDRAM as defined by the Sprite SDRAM Based Registers Start Address bits, REG[5028h] ~ REG[502Bh]. For detailed descriptions of these registers, see Section 10.4.23, “Sprite Memory Based Registers” on page 426.

Note

1. The Sprite registers (REG[5000h] ~ REG[502Bh]) and Sprite SDRAM Based registers (SDRAM[**000h] ~ SDRAM[**01Ah]) must be updated only while the Sprite Module is idle, REG[5003h] bit 7 = 1b.
2. The Sprite Engine must use SDRAM memory space and may not use SRAM memory space in REG[5020h] ~ REG[5027h] and REG[5028h] ~ REG[502Bh].

REG[5000h] Sprite Control Register Default = 02h							Read/Write
n/a	Sprite Individual Color Format Enable	Sprite Color Format bits 1-0		n/a		Sprite Frame Double Buffer Enable	Sprite Engine Enable
7	6	5	4	3	2	1	0

bit 6 — Sprite Individual Color Format Enable
This bit determines whether all sprites share the same color format as specified by the Sprite Color Format bits (REG[5000h] bits 5-4), or each sprite has a specific color format associated with it.
When this bit = 0b, all sprites share the same color format as specified by the Sprite Color Format bits (see REG[5000h] bits 5-4).
When this bit = 1b, each sprite has a color format specified for it in the SDRAM based sprite registers (see SDRAM[**01Ah]).

bits 5-4 — Sprite Color Format bits [1:0]
When the Sprite Individual Color Format bit is disabled (REG[5000h] bit 6 = 0b), these bits specify the color format for all sprite data.

Table 10-107: Sprite Color Format

REG[5000h] bits 5-4	RGB Format
00b	RGB 5:6:5
01b	ARGB 1:5:5:5:
10b	ARGB 4:4:4:4
11b	Reserved

bit 1 Sprite Frame Double Buffer Enable
This bit determines whether sprite frame data is written to SDRAM using single or double buffer mode.
When this bit = 0b, the Sprite Engine uses a single buffer for rendered sprite frames. Buffer 0 is specified by REG[5020h] ~ REG[5022h].
When this bit = 1b, the Sprite Engine uses double buffers for rendered sprite frames. Buffer 0 is specified by REG[5020h] ~ REG[5022h] and Buffer 1 is specified by REG[5024h] ~ REG[5026h]. (default)

bit 0 Sprite Engine Enable
This bit controls the Sprite Engine. Sprite operations cannot be triggered while the Sprite Engine is disabled.
When this bit = 0b, the Sprite Engine is disabled.
When this bit = 1b, the Sprite Engine is enabled.

Note

1. The Sprite Engine must be enabled before writing to REG[5004h] ~ REG[501Eh].
2. If the Sprite Engine Enable bit is set to 0b while the Sprite Engine is busy (REG[5003h] bit 7 = 0b), the Sprite Engine is not disabled until it becomes idle (REG[5003h] bit 7 = 1b).

REG[5001h] Sprite Software Reset Register Default = 00h							Read/Write
Sprite Software Reset (WO)	n/a						
7	6	5	4	3	2	1	0

bit 7 Sprite Software Reset (Write Only)
This bit only has an effect when the Sprite Engine is enabled, REG[5000h] bit 0 = 1b.
This bit performs a software reset of the Sprite Engine. This bit does not clear the Sprite registers.
Writing a 0b to this bit has no hardware effect.
Writing a 1b to this bit initiates a software reset of the Sprite Engine.

Note

The Sprite Engine must be idle (REG[5003h] bit 7 = 1b) before initiating a Sprite Software Reset using this bit.

REG[5002h] Sprite SDRAM Registers Busy Register Default = 00h							Read/Write
Sprite SDRAM Registers Busy (RO)	n/a						
7	6	5	4	3	2	1	0

bit 7 Sprite SDRAM Registers Busy (Read Only)
This bit indicates when the Sprite Engine is sampling the SDRAM based sprite registers. The Sprite SDRAM Based Registers Start Address (REG[5028h] ~ REG[502Bh]) and the Sprite SDRAM Based registers (see Section 10.4.23, “Sprite Memory Based Registers” on page 426) must not be written to when this bit is 1b.
When this bit = 0b, the Sprite Engine is not sampling the SDRAM based sprite registers.
When this bit = 1b, the Sprite Engine is sampling the SDRAM based sprite registers.

Note
The busy time is typically less than 0.2% of the frame time.

REG[5003h] Sprite Engine Status Register Default = 80h							Read/Write
Sprite Engine Status (RO)	n/a				Reserved		
7	6	5	4	3	2	1	0

bit 7 Sprite Engine Status (Read Only)
This bit indicates the status of the Sprite Engine. The Sprite Engine becomes busy when a new sprite operation is triggered (REG[5004h] bit 0 = 1b) and returns to an idle state once the current sprite operation is complete. The time required to complete a sprite operation varies based on the number of enabled sprites, the size of the sprites, refresh rate, etc.
When this bit = 0b, the Sprite Engine is busy.
When this bit = 1b, the Sprite Engine is idle (or ready). (default)

Note
The Sprite registers (REG[5000h] ~ REG[502Bh]) and Sprite SDRAM Based registers (SDRAM[**000h] ~ SDRAM[**01Ah]) must be updated only while the Sprite Module is idle, REG[5003h] bit 7 = 1b.

bits 2-0 Reserved
The default value of these bits is 000b.

REG[5004h] Sprite Frame Trigger Control Register Default = 00h							Write Only
n/a							Sprite Manual Trigger
7	6	5	4	3	2	1	0

bit 0 Sprite Manual Trigger (Write Only)
This bit triggers a new sprite operation. If no sprites are enabled (see SDRAM[**000h]), a new sprite operation (paint) must not be triggered.
Writing a 0b to this bit has no effect.
Writing a 1b to this bit triggers a new sprite operation.

REG[5006h] Sprite Interrupt Control Register Default = 00h							Read/Write
n/a						Sprite Operation Complete Interrupt Enable	n/a
7	6	5	4	3	2	1	0

bit 1 Sprite Operation Complete Interrupt Enable
This bit determines whether the state of the Sprite Operation Complete Interrupt Status bit (REG[5008h] bit 1) is reflected in the Sprite Interrupt Status bit in REG[0A00h] bit 7.
When this bit = 0b, the Sprite Operation Complete Interrupt is disabled and the state is not reflected in REG[0A00h] bit 7.
When this bit = 1b, the Sprite Operation Complete Interrupt is enabled and the state is reflected in REG[0A00h] bit 7.

REG[5008h] Sprite Interrupt Status Register Default = 00h							Read/Write
n/a						Sprite Operation Complete Interrupt Status	n/a
7	6	5	4	3	2	1	0

bit 1 Sprite Operation Complete Interrupt Status
This bit indicates the status of the current sprite operation. This bit is not masked by the Sprite Operation Complete Interrupt Enable bit, REG[5006h] bit 1.
When this bit = 0b, a Sprite Operation Complete Interrupt has not occurred (the Sprite operation has not completed yet).
When this bit = 1b, a Sprite Operation Complete Interrupt has occurred (the Sprite operation has completed).

To clear this status bit, write a 1b to this bit.

REG[5020h] Sprite Frame Buffer 0 Start Address Register 0 Default = 00h							Read/Write
Sprite Frame Buffer 0 Start Address bits 7-0 (bits 1-0 always return 00b)							
7	6	5	4	3	2	1	0

REG[5021h] Sprite Frame Buffer 0 Start Address Register 1 Default = 00h							Read/Write
Sprite Frame Buffer 0 Start Address bits 15-8							
7	6	5	4	3	2	1	0

REG[5022h] Sprite Frame Buffer 0 Start Address Register 2 Default = 00h							Read/Write
Sprite Frame Buffer 0 Start Address bits 23-16							
7	6	5	4	3	2	1	0

REG[5023h] Sprite Frame Buffer 0 Start Address Register 3 Default = 00h							Read/Write
Sprite Frame Buffer 0 Start Address bits 31-24							
7	6	5	4	3	2	1	0

REG[5023h] bits 7-0
REG[5022h] bits 7-0
REG[5021h] bits 7-0
REG[5020h] bits 7-0 Sprite Frame Buffer 0 Start Address bits [31:0]

These bits specify the memory start address for Sprite Frame Buffer 0 which is used for both single buffer and double buffer modes (see REG[5000h] bit 1). These bits must be set such that the start address is 8 byte (64-bit) aligned.

REG[5024h] Sprite Frame Buffer 1 Start Address Register 0 Default = 00h							Read/Write
Sprite Frame Buffer 1 Start Address bits 7-0 (bits 1-0 always return 00b)							
7	6	5	4	3	2	1	0

REG[5025h] Sprite Frame Buffer 1 Start Address Register 1 Default = 80h							Read/Write
Sprite Frame Buffer 1 Start Address bits 15-8							
7	6	5	4	3	2	1	0

REG[5026h] Sprite Frame Buffer 1 Start Address Register 2 Default = 02h							Read/Write
Sprite Frame Buffer 1 Start Address bits 23-16							
7	6	5	4	3	2	1	0

REG[5027h] Sprite Frame Buffer 1 Start Address Register 3 Default = 00h							Read/Write
Sprite Frame Buffer 1 Start Address bits 31-24							
7	6	5	4	3	2	1	0

REG[5027h] bits 7-0
REG[5026h] bits 7-0
REG[5025h] bits 7-0
REG[5024h] bits 7-0 Sprite Frame Buffer 1 Start Address bits [31:0]

These bits specify the memory start address for Sprite Frame Buffer 1 which is used for double buffer mode, REG[5000h] bit 1 = 1b. These bits must be set such that the start address is 8 byte (64-bit) aligned.

REG[5028h] Sprite SDRAM Based Registers Start Address Register 0 Default = 00h							Read Only
Sprite SDRAM Based Registers Start Address bits 7-0 (bits 7-0 always return 0000_0000b)							
7	6	5	4	3	2	1	0

REG[5029h] Sprite SDRAM Based Registers Start Address Register 1 Default = F0h							Read/Write
Sprite SDRAM Based Registers Start Address bits 15-8 (bits 3-0 always return 0000b)							
7	6	5	4	3	2	1	0

REG[502Ah] Sprite SDRAM Based Registers Start Address Register 2 Default = 04h							Read/Write
Sprite SDRAM Based Registers Start Address bits 23-16							
7	6	5	4	3	2	1	0

REG[502Bh] Sprite SDRAM Based Registers Start Address Register 3 Default = 00h							Read/Write
Sprite SDRAM Based Registers Start Address bits 31-24							
7	6	5	4	3	2	1	0

REG[502Bh] bits 7-0
REG[502Ah] bits 7-0
REG[5029h] bits 7-0
REG[5028h] bits 7-0 Sprite SDRAM Based Registers Start Address bits [31:0]

These bits specify the memory start address for the Sprite SDRAM based registers. Bits 11-0 of this address always return 000h and writing to them has no effect (REG[5028h] bits 7-0 always return 0000_0000b and REG[5029h] bits 3-0 always return 0000b).

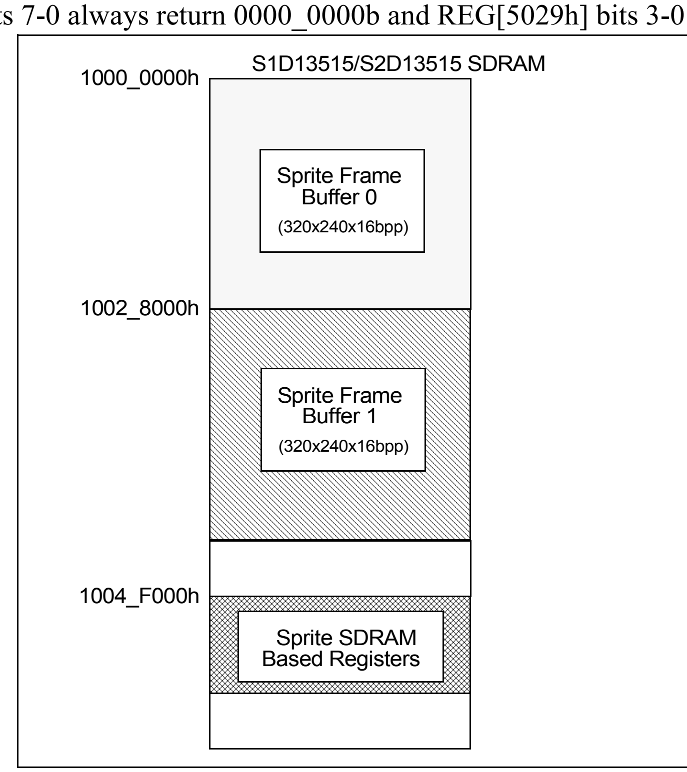


Figure 10-3: Sprite Memory Map Example

10.4.23 Sprite Memory Based Registers

The Sprite SDRAM Based registers specify settings for each individual sprite (Sprite #0-#7). These registers occupy 4K bytes of SDRAM starting from the offset specified by the Sprite SDRAM Based Registers Start Address bits (REG[5028h] ~ REG[502Bh]. Note that the programmed offset must be on a 4K byte boundary. The registers are always 16-bits wide and can be accessed by specifying the specified SDRAM location. Access timing is the same as regular SDRAM Read/Write Accesses. The following figure shows an example of the Sprite SDRAM Based registers located at 1004_F000h in SDRAM memory.

Note
The Sprite registers (REG[5000h] ~ REG[502Bh]) and Sprite SDRAM Based registers (SDRAM[**000h] ~ SDRAM[**01Ah]) must be updated only while the Sprite Module is idle, REG[5003h] bit 7 = 1b.

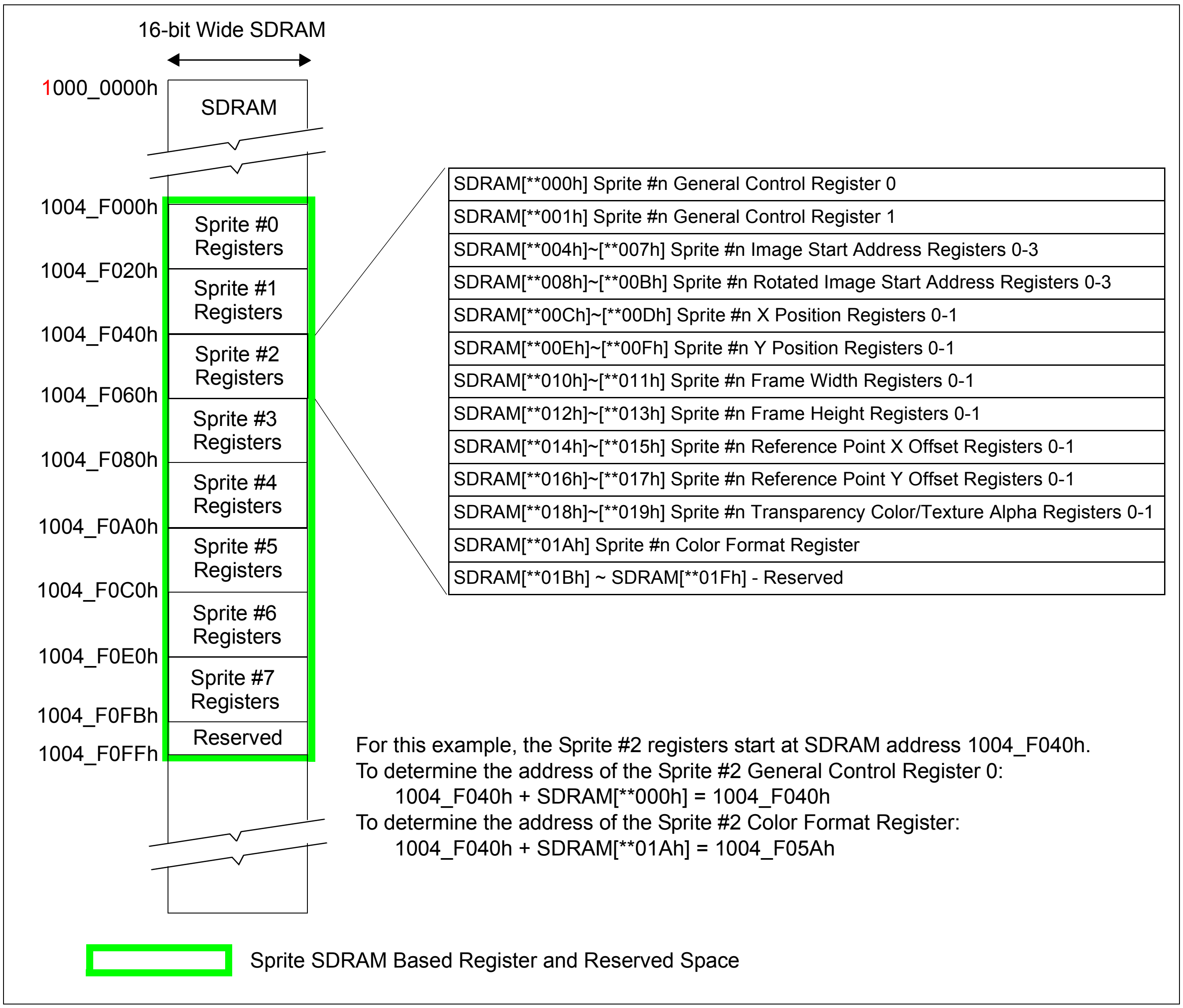


Figure 10-4: Sprite SDRAM Based Register Mapping Example

SDRAM[**000h] Sprite #n General Control Register 0
Default = XXh
							Read/Write

n/a							Sprite #n Enable
7	6	5	4	3	2	1	0

bit 0 Sprite #n Enable
This bit controls the associated sprite (i.e. Sprite #n). At least one sprite must be enabled before a sprite operation is triggered using the Sprite Manual Trigger bit, REG[5004h] bit 0.
When this bit = 0b, sprite #n is disabled.
When this bit = 1b, sprite #n is enabled.

Note
Sprite #0 is used as the background sprite and must always be enabled and the lowest Z-order of all sprites when a Sprite paint is triggered using REG[5004h] bit 0.

SDRAM[**001h] Sprite #n General Control Register 1
Default = XXh
							Read/Write

n/a	Sprite #n Z-order bits 2-0			Sprite #n Rotation bits 1-0		Sprite #n Mirror Enable	Sprite #n Transparency Enable
7	6	5	4	3	2	1	0

bits 6-4 Sprite #n Z-Order bits [2:0]
These bits specify the Z-order associated with sprite #n which determines the priority of the sprite for alpha blending and transparency functions. The Z-order value ranges from 7h which signifies the top (foreground) to 0h which signifies the bottom (background). If more than one sprite is assigned the same Z-order, the higher numbered sprite takes priority over the lower numbered sprite.

Note
Sprite #0 is used as the background and must be set to the lowest Z-order.

bits 3-2 Sprite #n Rotation bits [1:0]
These bits specify the clockwise rotation applied to the Sprite #n image.

Table 10-108 : Sprite #n Rotation

SDRAM[**001h] bits 3-2	Sprite #n Rotation
00b	0° rotation
01b	90° rotation
10b	180° rotation
11b	270° rotation

bit 1 Sprite #n Mirror Enable
This bit controls the horizontal mirror function for Sprite #n.
When this bit = 0b, Sprite #n is not mirrored (normal).
When this bit = 1b, Sprite #n is mirrored horizontally.

bit 0 Sprite #n Transparency Enable
This bit controls the transparency function for Sprite #n. For RGB 5:6:5 format, when a pixel is transparent as specified by SDRAM[***19h] ~ SDRAM[***18h], the next visible pixel below it, according to the Z-order, is visible. For ARGB 4:4:4:4 format, the specified transparent pixel is only used to determine pixel collisions and has no effect on image rendering or visibility.
When this bit = 0b, Sprite #n transparency is disabled.
When this bit = 1b, Sprite #n transparency is enabled.

SDRAM[004h] Sprite #n Image Start Address Register 0**
Default = XXh
							Read/Write

Sprite #n Image Start Address bits 7-0							
7	6	5	4	3	2	1	0

SDRAM[005h] Sprite #n Image Start Address Register 1**
Default = XXh
							Read/Write

Sprite #n Image Start Address bits 15-8							
7	6	5	4	3	2	1	0

SDRAM[006h] Sprite #n Image Start Address Register 2**
Default = XXh
							Read/Write

Sprite #n Image Start Address bits 23-16							
7	6	5	4	3	2	1	0

SDRAM[007h] Sprite #n Image Start Address Register 3**
Default = XXh
							Read/Write

Sprite #n Image Start Address bits 31-24							
7	6	5	4	3	2	1	0

SDRAM[**007h] bits 7-0
SDRAM[**006h] bits 7-0
SDRAM[**005h] bits 7-0
SDRAM[**004h] bits 7-0
Sprite #n Image Start Address bits [31:0]
These bits specify the memory start address for the 0° or 180° rotated Sprite #n image stored in SDRAM. These bits must be set such that the start address is 16-bit aligned.

SDRAM[**008h] Sprite #n Rotated Image Start Address Register 0
Default = XXh
							Read/Write

Sprite #n Rotated Image Start Address bits 7-0							
7	6	5	4	3	2	1	0

SDRAM[**009h] Sprite #n Rotated Image Start Address Register 1
Default = XXh
							Read/Write

Sprite #n Rotated Image Start Address bits 15-8							
7	6	5	4	3	2	1	0

SDRAM[**00Ah] Sprite #n Rotated Image Start Address Register 2
Default = XXh
							Read/Write

Sprite #n Rotated Image Start Address bits 23-16							
7	6	5	4	3	2	1	0

SDRAM[**00Bh] Sprite #n Rotated Image Start Address Register 3
Default = XXh
							Read/Write

Sprite #n Rotated Image Start Address bits 31-24							
7	6	5	4	3	2	1	0

SDRAM[**00Bh] bits 7-0
SDRAM[**00Ah] bits 7-0
SDRAM[**009h] bits 7-0
SDRAM[**008h] bits 7-0

Sprite #n Rotated Image Start Address bits [31:0]
These bits specify the memory start address for the 90° or 270° rotated Sprite #n image stored in SDRAM. These bits must be set such that the start address is 16-bit aligned.

SDRAM[**00Ch] Sprite #n X Position Register 0
Default = XXh
							Read/Write

Sprite #n X Position bits 7-0							
7	6	5	4	3	2	1	0

SDRAM[**00Dh] Sprite #n X Position Register 1
Default = XXh
							Read/Write

Sprite #n X Position Sign bits 5-0						Sprite #n X Position bits 9-8	
7	6	5	4	3	2	1	0

SDRAM[**00Dh] bits 7-2

Sprite #n X Position Sign bits [5:0]
These bits are the extended sign bits which determine if the X position is negative with respect to the top left corner.

Note
Sprite #0 must not be set to a negative X position as it must remain on screen.

SDRAM[**00Dh] bits 1-0
SDRAM[**00Ch] bits 7-0

Sprite #n X Position bits [9:0]
These bits specify the X position of the sprite reference point with respect to the top left corner of the display. A negative position value allows the sprite to move off the display in any direction. The X position must be programmed such that the following formulas are valid.

$-1007 < \text{X position} < 1007$
$\text{X position} + (\text{sprite width} - \text{sprite reference point X offset}) \leq 1024.$

Note
SDRAM[**00Dh] bits 7-2 and SDRAM[**00Dh] bits 1-0, SDRAM[**00Ch] bits 7-0 together form an 11-bit 2’s complement number. The 16-bit register value is a 2’s complement number and that the range of the values should be within -1024 (1111_1100_0000_0000b) to 1023 (0000_0011_1111_1111b).

SDRAM[**00Eh] Sprite #n Y Position Register 0
Default = XXh
							Read/Write

Sprite #n Y Position bits 7-0							
7	6	5	4	3	2	1	0

SDRAM[**00Fh] Sprite #n Y Position Register 1
Default = XXh
							Read/Write

Sprite #n Y Position Sign bits 5-0						Sprite #n Y Position bits 9-8	
7	6	5	4	3	2	1	0

SDRAM[**00Fh] bits 7-2 Sprite #n Y Position Sign bits [5:0]

These bits are the extended sign bits which determine if the Y position is negative with respect to the top left corner.

Note

Sprite #0 must not be set to a negative Y position as it must remain on screen.

SDRAM[**00Fh] bits 1-0
SDRAM[**00Eh] bits 7-0 Sprite #n Y Position bits [9:0]

These bits specify the Y position of the sprite reference point with respect to the top left corner of the display. A negative position value allows the sprite to move off the display in any direction. The Y position must be programmed such that the following formula is valid.

$-1007 < \text{Y position} < 1007$

$\text{Y position} + (\text{sprite height} - \text{sprite reference point Y offset}) \leq 1024.$

SDRAM[010h] Sprite #n Frame Width Register 0**							
Default = XXh							Read/Write
Sprite #n Frame Width bits 7-0							
7	6	5	4	3	2	1	0

SDRAM[011h] Sprite #n Frame Width Register 1**							
Default = XXh							Read/Write
Reserved						Sprite #n Frame Width bits 9-8	
7	6	5	4	3	2	1	0

SDRAM[**011h] bits 7-2Reserved

These bits must be set to 00_0000b.

SDRAM[**011h] bits 1-0
SDRAM[**010h] bits 7-0Sprite #n Frame Width bits [9:0]

These bits specify the width of the sprite frame, in pixels. All sprites, except for Sprite #0, must conform to this size when written to memory. These bits must be programmed such that the following formula is valid.

Frame Width < 1007

Note

For Sprite #0, when SDRAM[**00h] bit 10 is 0 (0°/180° rotation), this register also defines the frame buffer width, it must be divisible by 2 and must be greater than 8. When SDRAM[**00h] bit 10 is 1 (90°/270° rotation), this register also defines the frame buffer height.

SDRAM[012h] Sprite #n Frame Height Register 0**							
Default = XXh							Read/Write
Sprite #n Frame Height bits 7-0							
7	6	5	4	3	2	1	0

SDRAM[013h] Sprite #n Frame Height Register 1**							
Default = XXh							Read/Write
Reserved						Sprite #n Frame Height bits 9-8	
7	6	5	4	3	2	1	0

SDRAM[**013h] bits 7-2Reserved

These bits must be set to 00_0000b.

SDRAM[**013h] bits 1-0
SDRAM[**012h] bits 9-0Sprite #n Frame Height bits [9:0]

These bits specify the height of the sprite frame, in lines. All sprites, except for Sprite #0, must conform to this size when written to memory. These bits must be programmed such that the following formula is valid.

Frame Height < 1007

Note

For sprite #0, when SDRAM[**00h] bit 10 is 0 (0°/180° rotation), this register also defines the frame buffer height. When SDRAM[**00h] bit 10 is 1 (90°/270° rotation), this register also defines the frame buffer width, it must be divisible by 2 and must be greater than 8.

SDRAM[**014h] Sprite #n Reference Point X Offset Register 0
Default = XXh Read/Write							

Sprite #n Reference Point X Offset bits 7-0							
7	6	5	4	3	2	1	0

SDRAM[**015h] Sprite #n Reference Point X Offset Register 1
Default = XXh Read/Write							

Sprite #n Reference Point X Offset Sign bits 5-0						Sprite #n Reference Point X Offset bits 9-8	
7	6	5	4	3	2	1	0

SDRAM[**015h] bits 7-2Sprite #0 Reference Point X Offset Sign bits [5:0]

These are the extended sign bits to determine if the X offset is negative with respect to the top left corner of the sprite.

SDRAM[**015h] bits 1-0
SDRAM[**014h] bits 7-0Sprite #n Reference Point X Offset bits [9:0]

These bits specify the X direction offset of the sprite reference point with respect to the top left corner of the sprite.

Note

Once set, the reference point can be used to set and query the location of the sprite. The reference point also serves as the “center” for all transforms (rotation and mirror). If desired, the reference point may be defined outside of the sprite’s bounds.

SDRAM[**016h] Sprite #n Reference Point Y Offset Register 0
Default = XXh
							Read/Write

Sprite #n Reference Point Y Offset bits 7-0							
7	6	5	4	3	2	1	0

SDRAM[**017h] Sprite #n Reference Point Y Offset Register 1
Default = XXh
							Read/Write

Sprite #n Reference Point Y Offset Sign bits 5-0						Sprite #n Reference Point Y Offset bits 9-8	
7	6	5	4	3	2	1	0

SDRAM[**017h] bits 7-2Sprite #n Reference Point Y Offset Sign bits [5:0]

These are the extended sign bits to determine if the Y offset is negative with respect to the top left corner of the sprite.

SDRAM[**017h] bits 1-0
SDRAM[**016h] bits 7-0Sprite #0 Reference Point Y Offset bits [9:0]

These bits specify the Y direction offset of the sprite reference point with respect to the top left corner of the sprite.

Note

Once set, the reference point can be used to set and query the location of the sprite. The reference point also serves as the ‘center’ for all transforms (rotation and mirror). If desired, the reference point may be defined outside of the sprite’s bounds.

SDRAM[**018h] Sprite #n Transparency Color / Texture Alpha Register 0
Default = XXh
							Read/Write

Sprite #n Transparency Color / Texture Alpha bits 7-0							
7	6	5	4	3	2	1	0

SDRAM[**019h] Sprite #n Transparency Color / Texture Alpha Register 1
Default = XXh
							Read/Write

Sprite #n Transparency Color / Texture Alpha bits 15-8							
7	6	5	4	3	2	1	0

SDRAM[**019h] bits 7-0
SDRAM[**018h] bits 7-0

Sprite #n Transparency Color / Texture Alpha bits [15:0]
If the sprite data format is RGB 5:6:5, bits 15-0 define the 16 bpp transparency color, in RGB 5:6:5 format. When a pixel of the Sprite #n transparency color is found in Sprite #n, the pixel is replaced with the color of the pixel "under" it from the next lowest Z-order sprite. If all pixels "under" the sprite are also transparent, including Sprite #0, the pixel color is replaced with the OSD transparency color (REG[09A4h] ~ REG[09A6h]). If the Sprite Individual Color Format Enable bit is enabled (REG[5000h] bit 6 = 1b), the Sprite #0 pixel color is not be replaced with the OSD transparency color.

Note
Sprite #0 must not have transparency enabled.

SDRAM[**019h] bits 3-0

If the sprite data is ARGB 1:5:5:5, these bits give the 4-bit alpha value when the alpha index value is 1.

SDRAM[**018h] bits 3-0

If the sprite data is ARGB 1:5:5:5, these bits give the 4-bit alpha value when the alpha index value is 0.

SDRAM[**01Ah] Sprite #n Color Format Register
Default = XXh
							Read/Write

n/a						Sprite #n Color Format bits 1-0	
7	6	5	4	3	2	1	0

bits 1-0 Sprite #n Color Format bits [1:0]
If the Sprite Individual Color Format Enable bit is set to 1b (REG[5000h] bit 6 = 1b), these bits determine the color format for Sprite #n.

Table 10-109: Sprite #n Color Format

SDRAM[**01Ah] bits 1-0	Color Format	Sprite Transparency Color
00b	RGB 5:6:5	Transparency Color is defined by SDRAM[**018h] ~ SDRAM[**019h]
01b	ARGB 1:5:5:5	Texture Alpha is defined by SDRAM[**018h] ~ SDRAM[**019h] When the 1-bit Alpha value is 0b, SDRAM[**018h] bits 3-0 are used When the 1-bit Alpha value is 1b, SDRAM[**019h] bits 3-0 are used
10b	ARGB 4:4:4:4	SDRAM[**018h] ~ SDRAM[**019h] are not used.
11b	Reserved	

SDRAM[**01Bh] through SDRAM[**01Fh] are Reserved

These registers are Reserved and should not be written.

Chapter 11 Operating Configurations and States

The S1D13515/S2D13515 has two general operating configurations: Stand-Alone and Host-Controlled.

After hardware reset is released, the S1D13515/S2D13515 enables the system clock to be running from either CLKI or OSCI (selectable by the CNF0 pin) and boots up the internal C33PE processor to run from the internal boot ROM. If there is no Host interface connected, the S1D13515/S2D13515 operates in the Stand-Alone configuration. If a Host interface is connected, the Host software can hold the C33PE processor in reset and perform software reset on the S1D13515/S2D13515, in which case the S1D13515/S2D13515 is operating in the Host-Controlled configuration.

The following diagram shows the operating configurations and states of the S1D13515/S2D13515.

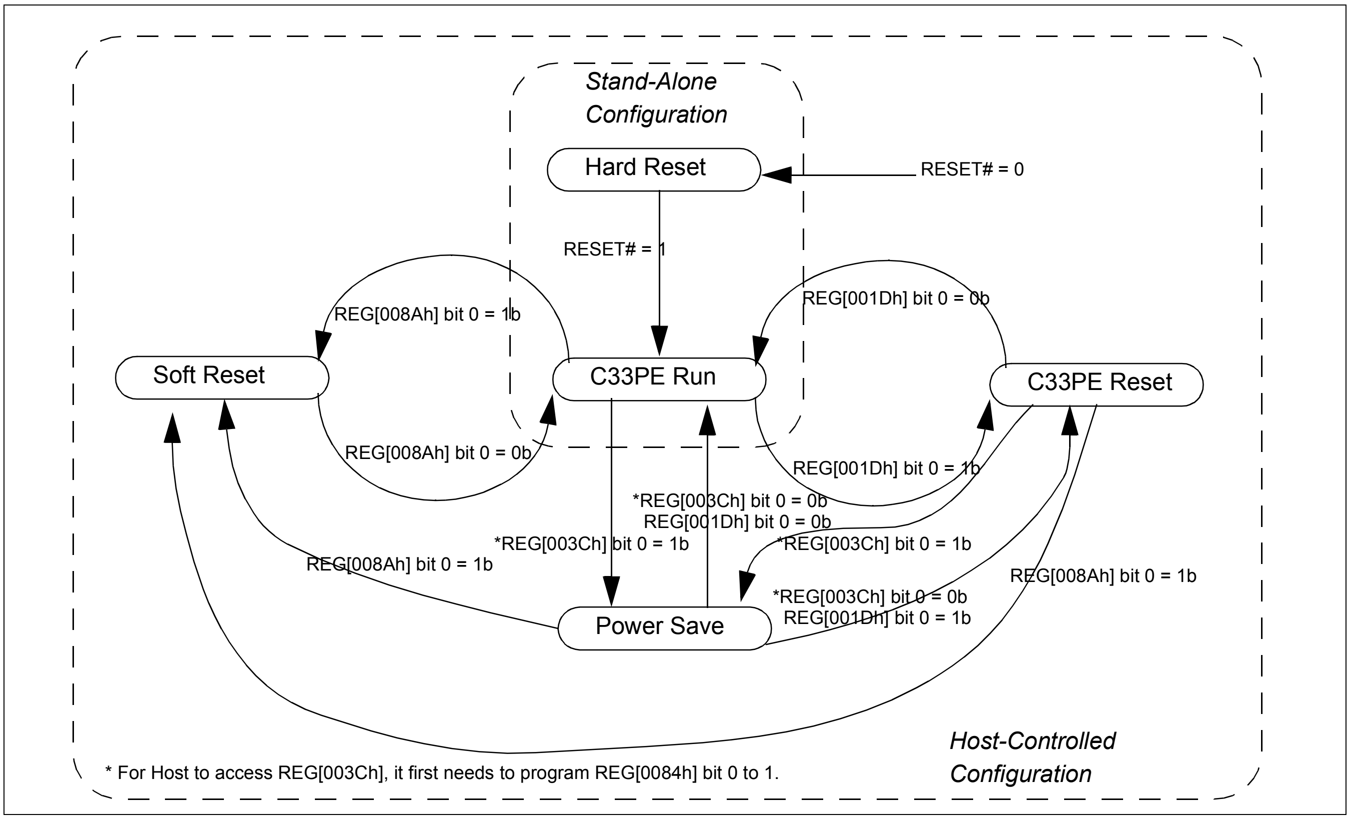


Figure 11-1: Operating Configurations and States

The Stand-Alone configuration is actually a subset of the Host-Controlled configuration. Even in Host-Controlled configuration, the S1D13515/S2D13515 will always start to operate in Stand-Alone configuration with the C33PE processor booting up, and it is up to the Host software to stop the C33PE processor or perform Soft Reset.

11.1 Hard Reset State

The Hard Reset state is entered whenever the RESET# input pin is asserted to 0. When RESET# is deasserted to 1, the S1D13515/S2D13515 goes from the Hard Reset state to the C33PE Run state.

In Hard Reset state, the System Clock source is selected by the CNF0 pin between CLKI/OSCI and PLL1 is disabled. The LCD Clock source is CLKI. The states of the IO pins of the S1D13515/S2D13515 are shown in Table 11-1 “Hard Reset Pin States for Signals Which Are Not Part of Host Interface,” on page 438.

In the following Tables, PD = pull-down, PU = pull-up, Z = high impedance, X = unknown, CLK = clock signal

Table 11-1 : Hard Reset Pin States for Signals Which Are Not Part of Host Interface

S1D13515/S2D13515 Pin	Type	DIR	State
LCD1/CAMERA2 INTERFACE PINS			
FP1IO23	IO	I	PD
FP1IO22	IO	I	PD
FP1IO21	IO	I	PD
FP1IO20	IO	I	PD
FP1IO19	IO	I	PD
FP1IO18	IO	I	PD
FP1IO17	IO	I	PD
FP1IO16	IO	I	PD
FP1IO15	IO	I	PD
FP1IO14	IO	I	PD
FP1IO13	IO	I	PD
FP1IO12	IO	I	PD
FP1IO11	IO	I	PD
FP1IO10	IO	I	PD
FP1IO9	IO	I	PD
FP1IO8	IO	I	PD
FP1IO7	IO	I	PD
FP1IO6	IO	I	PD
FP1IO5	IO	I	PD
FP1IO4	IO	I	PD
FP1IO3	IO	I	PD
FP1IO2	IO	I	PD
FP1IO1	IO	I	PD
FP1IO0	IO	I	PD
SDRAM INTERFACE PINS			
MEMA[12:0]	O	O	0
MEMBA[1:0]	O	O	0
MEMCS#	O	O	1
MEMRAS#	O	O	1
MEMCAS#	O	O	1
MEMWE#	O	O	1
MEMDQM[3:0]	O	O	1
MEMCLK	O	O	CLK
MEMCKE	O	O	1
MEMDQ[31:0]	IO	I	PD
SERIAL FLASH / SPI INTERFACE PINS			
SPICS#	O	O	1
SPICLK	O	O	0
SPIDIO	IO	I	PD

S1D13515/S2D13515 Pin	Type	DIR	State
CAMERA1 INTERFACE PINS			
CM1DAT[7:0]	I	I	PD
CM1CLKIN	I	I	PD
CM1CLKOUT	O	O	0
CM1HREF	I	I	PD
CM1VREF	I	I	PD
CM1FIELD	I	I	PD
SCL	IO	I	PU
SDA	IO	I	PU
LCD2 INTERFACE PINS			
FP2IO[27:24]	O	O	0
FP2IO23	IO	I	PD
FP2IO22	IO	I	PD
FP2IO21	IO	I	PD
FP2IO20	IO	I	PD
FP2IO19	IO	I	PD
FP2IO18	IO	I	PD
FP2IO[17:0]	O	O	0
MISCELLANEOUS PINS			
CNF0	I	I	Z
OSCI	I	I	Z
OSCO	O	O	X
CLKI	I	I	Z
TESTEN	I	I	see note
RESET#	I	I	Z
IRQ	O	O	0
PWM2	O	O	1
PWM1	O	O	1
TCK	I	I	PU
TMS	I	I	PU
TDI	I	I	PU
TDO	O	O	0
TRST	I	I	PU
I2S AUDIO OUTPUT INTERFACE PINS			
WSIO	IO	I	PD
SCKIO	IO	I	PD
SDO	O	O	0
MCLKO	O	O	0

Note
The TESTEN pin must be connected to VSS for normal operation.

Table 11-2 : Hard Reset Pin States for Host Interface 1

S1D13515/ S2D13515 Pin	Type	Intel80 Type1 8-bit Indirect		Intel80 Type2 8-bit Indirect		NEC V850 Type1 8-bit Indirect		NEC V850 Type2 8-bit Indirect		Renesas SH4 8-bit Indirect		Intel80 Type1 16-bit Indirect		Intel80 Type2 16-bit Indirect		NEC V850 Type1 16-bit Indirect		NEC V850 Type2 16-bit Indirect		Renesas SH4 16-bit Indirect	
		DIR	PU/D	DIR	PU/D	DIR	PU/D	DIR	PU/D	DIR	PU/D	DIR	PU/D	DIR	PU/D	DIR	PU/D	DIR	PU/D	DIR	PU/D
DB15	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB14	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB13	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB12	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB11	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB10	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB9	IO	I	PU	I	PU	I	PU	I	PU	I	PU	I	-	I	-	I	-	I	-	I	-
DB8	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB7	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB6	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB5	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB4	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB3	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB2	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB1	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB0	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
M/R#	IO	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z
AB20	IO	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z
AB19	IO	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z
AB18	I	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z
AB17	IO	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z
AB16	IO	O	1	O	1	O	1	O	1	O	1	O	1	O	1	O	1	O	1	O	1
AB15	IO	O	1	O	1	O	1	O	1	O	1	O	1	O	1	O	1	O	1	O	1
AB14	IO	O	1	O	1	O	1	O	1	O	1	O	1	O	1	O	1	O	1	O	1
AB13	IO	O	1	O	1	O	1	O	1	O	1	O	1	O	1	O	1	O	1	O	1
AB12	IO	O	1	O	1	O	1	O	1	O	1	O	1	O	1	O	1	O	1	O	1
AB11	IO	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD
AB10	IO	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD
AB9	IO	O	0/PD	O	0/PD	O	0/PD	O	0/PD	O	0/PD	O	0/PD	O	0/PD	O	0/PD	O	0/PD	O	0/PD
AB8	IO	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD
AB7	IO	I	PU	I	PU	I	PU	I	PU	I	PU	I	PU	I	PU	I	PU	I	PU	I	PU
AB6	IO	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD
AB5	I	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
AB4	I	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
AB3	I	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z
AB2	I	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
AB1	I	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
AB0	I	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
BUSCLK	I	I	PD	I	PD	I	Z	I	Z	I	Z	I	PD	I	PD	I	Z	I	Z	I	Z
BS#	IO	I	PU	I	PU	I	PU	I	PU	I	Z	I	PU	I	PU	I	PU	I	PU	I	Z
WAIT#	IO	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z
RD#	I	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z
RD/WR#	I	I	Z	I	PD	I	Z	I	PD	I	Z	I	Z	I	PD	I	Z	I	PD	I	Z
CS#	I	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z
BE1#	IO	I	PD	I	note1	I	PD	I	PD	I	PD	I	note2	I	Z	I	note2	I	Z	I	Z
BE0#	I	I	PD	I	Z	I	PD	I	Z	I	PD	I	note2	I	Z	I	note2	I	Z	I	Z
BURST#	I	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z
BDIP#	I	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z
TEA#	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
CNF1	I	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z
CNF2	I	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z

Note

1. For the Intel 80 Type 2 Indirect 8-bit Host Interface, the BE1# pin must be connected to HIOVDD.
2. For the Intel 80 and NEC V850 Type 1 Indirect 16-bit to Host Interfaces, BE1# and BE0# are “Z”. Both BE1# and BE0# should be tied to VSS for this host interface (byte access using the byte enables is not supported).

Table 11-3 : Hard Reset Pin States for Host Interface 2

S1D13515/ S2D13515 Pin	Type	Intel80 Type1 8-bit Direct		Intel80 Type2 8-bit Direct		NEC V850 Type1 8-bit Direct		NEC V850 Type2 8-bit Direct		Renesas SH4 8-bit Direct		Intel80 Type1 16-bit Direct		Intel80 Type2 16-bit Direct		NEC V850 Type1 16-bit Direct		NEC V850 Type2 16-bit Direct		Renesas SH4 16-bit Direct	
		DIR	PU/D	DIR	PU/D	DIR	PU/D	DIR	PU/D	DIR	PU/D	DIR	PU/D	DIR	PU/D	DIR	PU/D	DIR	PU/D	DIR	PU/D
DB15	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB14	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB13	IO	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB12	IO	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB11	IO	O	0/PD	O	0/PD	O	0/PD	O	0/PD	O	0/PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB10	IO	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB9	IO	I	PU	I	PU	I	PU	I	PU	I	PU	I	Z	I	Z	I	Z	I	Z	I	Z
DB8	IO	O	1/PD	O	1/PD	O	1/PD	O	1/PD	O	1/PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB7	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB6	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB5	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB4	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB3	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB2	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB1	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB0	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
M/R#	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
AB20	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
AB19	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
AB18	I	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
AB17	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
AB16	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
AB15	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
AB14	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
AB13	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
AB12	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
AB11	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
AB10	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
AB9	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
AB8	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
AB7	IO	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z
AB6	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
AB5	I	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
AB4	I	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
AB3	I	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
AB2	I	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
AB1	I	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
AB0	I	I	PD	I	PD	I	PD	I	PD	I	PD	I	Z	I	Z	I	Z	I	Z	I	Z
BUSCLK	I	I	PD	I	PD	I	Z	I	Z	I	Z	I	PD	I	PD	I	Z	I	Z	I	Z
BS#	IO	I	PU	I	PU	I	PU	I	PU	I	Z	I	PU	I	PU	I	PU	I	PU	I	Z
WAIT#	IO	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z
RD#	I	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z
RD/WR#	I	I	Z	I	PD	I	Z	I	PD	I	Z	I	Z	I	PD	I	Z	I	PD	I	Z
CS#	I	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z
BE1#	IO	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z
BE0#	I	I	PD	I	Z	I	PD	I	Z	I	PD	I	Z	I	Z	I	Z	I	Z	I	Z
BURST#	I	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z
BDIP#	I	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z
TEA#	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
CNF1	I	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z
CNF2	I	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z

Table 11-4 : Hard Reset Pin States for Host Interface 3

S1D13515/ S2D13515 Pin	Type	Marvell PXA3xx 16-bit Direct		TI TMS470 16-bit Indirect		TI TMS470 16-bit Direct		MPC555 16-bit Indirect		MPC555 16-bit Direct		I2C		SPI1		SPI2 (Camera1 streaming)	
		DIR	PU/D	DIR	PU/D	DIR	PU/D	DIR	PU/D	DIR	PU/D	DIR	PU/D	DIR	PU/D	DIR	PU/D
DB15	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB14	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB13	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB12	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB11	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB10	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB9	IO	I	Z	I	Z	I	Z	I	Z	I	Z	I	PU	I	PU	I	PU
DB8	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB7	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB6	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB5	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB4	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB3	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB2	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB1	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
DB0	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
M/R#	IO	I	Z	I	Z	I	PD	I	Z	I	PD	I	Z	I	Z	I	Z
AB20	IO	I	Z	I	Z	I	PD	I	Z	I	PD	I	Z	I	Z	I	Z
AB19	IO	I	Z	I	Z	I	PD	I	Z	I	PD	I	Z	I	Z	I	Z
AB18	I	I	Z	I	Z	I	PD	I	Z	I	PD	I	Z	I	Z	I	Z
AB17	IO	I	Z	I	Z	I	PD	I	Z	I	PD	I	Z	I	Z	I	Z
AB16	IO	O	1	O	1	I	PD	O	1	I	PD	O	1	O	1	O	1
AB15	IO	O	1	O	1	I	PD	O	1	I	PD	O	1	O	1	O	1
AB14	IO	O	1	O	1	I	PD	O	1	I	PD	O	1	O	1	O	1
AB13	IO	O	1	O	1	I	PD	O	1	I	PD	O	1	O	1	O	1
AB12	IO	O	1	O	1	I	PD	O	1	I	PD	O	1	O	1	O	1
AB11	IO	O	1/PD	O	1/PD	I	PD	O	1/PD	I	PD	O	1/PD	O	1/PD	O	1/PD
AB10	IO	O	1/PD	O	1/PD	I	PD	O	1/PD	I	PD	O	1/PD	O	1/PD	O	1/PD
AB9	IO	O	0/PD	O	0/PD	I	PD	O	0/PD	I	PD	O	0/PD	O	0/PD	O	0/PD
AB8	IO	O	1/PD	O	1/PD	I	PD	O	1/PD	I	PD	O	1/PD	O	1/PD	O	1/PD
AB7	IO	I	PU	I	PU	I	Z	I	PU	I	Z	I	PU	I	PU	I	PU
AB6	IO	I	Z	O	1/PD	I	PD	O	1/PD	I	PD	O	1/PD	O	1/PD	O	1/PD
AB5	I	I	Z	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
AB4	I	I	PD	I	PD	I	PD	I	PD	I	PD	I	Z	I	Z	I	Z
AB3	I	I	PD	I	PD	I	PD	I	PD	I	PD	I	-/PD	I	-/PD	I	Z
AB2	I	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
AB1	I	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
AB0	I	I	Z	I	Z	I	Z	I	PD	I	PD	I	-/PD	I	-/PD	I	PD
BUSCLK	I	I	PD	I	Z	I	Z	I	Z	I	Z	I	PD	I	PD	I	PD
BS#	IO	O	1/PU	I	Z	I	Z	I	Z	I	Z	I	PU	I	PU	I	PU
WAIT#	IO	I	Z	I	Z	I	Z	I	Z	I	Z	I	PU	I	Z	I	Z
RD#	I	I	Z	I	Z	I	Z	I	Z	I	Z	I	PD	I	PD	I	PD
RD/WR#	I	I	Z	I	Z	I	Z	I	Z	I	Z	I	PD	I	Z	I	Z
CS#	I	I	Z	I	Z	I	Z	I	Z	I	Z	I	PD	I	Z	I	Z
BE1#	IO	I	Z	I	Z	I	Z	I	Z	I	Z	I	-/PD	I	-/PD	I	PD
BE0#	I	I	Z	I	Z	I	Z	I	Z	I	Z	I	PD	I	Z	I	Z
BURST#	I	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z
BDIP#	I	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z
TEA#	IO	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD	I	PD
CNF1	I	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z
CNF2	I	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z	I	Z

11.2 C33PE Run State

When the C33PE is released from reset, it fetches the reset vector from address 00D00000h (boot ROM, which is also aliased/mapped at address 04300000h). The code in the boot ROM performs the following sequence of operations:

- Execute register initializations to prepare for reading of the external Serial Flash.
- Read the file system from the Serial Flash.
- If the data read from the Serial Flash is not valid (checksum error), the C33PE will execute a task which waits for commands from the Host.
- If there is valid data in the Serial Flash, the boot ROM reads two files which contain initialization values for programming the internal registers (including values for programming the clocks and PLL). It then goes ahead and programs all the registers.
- After successful programming of the two register files, the boot ROM will look for a startup batch file to run. If no batch file exists, the C33PE will go to the task which waits for command from the Host. If there is a batch, the boot ROM will load the program(s) from Serial Flash to SDRAM and starts executing the program(s).

The Host can reset the C33PE by writing a 1b to REG[001Dh] bit 0, and this puts the S1D13515/S2D13515 in the C33PE Reset state. See Section 11.3, “C33PE Reset State” on page 444 for more details.

The Host can put the S1D13515/S2D13515 into the Power Save state by writing 1b to REG[0084h] bit 0 (to enable asynchronous access by the Host to the clock/PLL control registers) and then writing 1b to REG[003Ch] bit 0. See Section 11.4, “Power Save State” on page 444 for more details.

The Host can put the S1D13515/S2D13515 into the Soft Reset state by writing 1 to REG[008Ah] bit 0. See the Section 11.5, “Soft Reset State” on page 444 for more details.

11.3 C33PE Reset State

In the C33PE Reset state, the C33PE processor is held in reset. This state can be used in the Host-Controlled Configuration for the Host to take full control of the S1D13515/S2D13515's resources and not have the C33PE processor's code interfere with the Host's operations.

To release the C33PE from reset, the Host writes 0b to REG[001Dh] bit 0. This will reboot the C33PE processor to read the reset vector and execute the code in the boot ROM.

11.4 Power Save State

This state is entered whenever REG[003Ch] bit 0 is set to 1b. In Power Save state, all clocks in the S1D13515/S2D13515 are stopped (gated off). the PLLs are still running, register values are retained, and the state of all IO pins are retained. Clocks are re-enabled when REG[003Ch] bit 0 is set to 0b and the S1D13515/S2D13515 exits the Power Save state.

If the contents of the external SDRAM need to be retained while in Power Save state, the Host should enable Self-Refresh mode of the SDRAM first before writing 1b to REG[003Ch] bit 0 to put the S1D13515/S2D13515 into Power Save state.

11.5 Soft Reset State

The Soft Reset state is entered by writing 1b to REG[008Ah] bit 0. Most of the S1D13515/S2D13515 is held in reset (similar to Hard Reset) except for some Host Interface logic which are not affected. The state of the IO pins will be the same as those in the Hard Reset state and all programmable registers (except those needed for the Host Interface) will be reset.

To exit the Soft Reset state, the Host writes 0 to REG[008Ah] bit 0. The S1D13515/S2D13515 will go to the C33PE Run State, similar to exiting from Hard Reset State, and start executing code from the boot ROM.

Chapter 12 Bit-Per-Pixel Converter Functional Description

The Bit-Per-Pixel (BPP) Converter assists the internal C33PE with up conversion or down conversion of graphics color depth.

Note
The Bit Per Pixel Converter (BPPC) Ports cannot be accessed through the Host interface. Accesses to and from the BPPC ports must be in 32-bit units.

For example, the case of the internal C33 operating in 32bpp unpacked mode, where a single 32-bits will be written to the BPP Converter, to be converted to 16bpp (or 8bpp) and written to a specific memory location.

Address and data Conversion is done as follows:

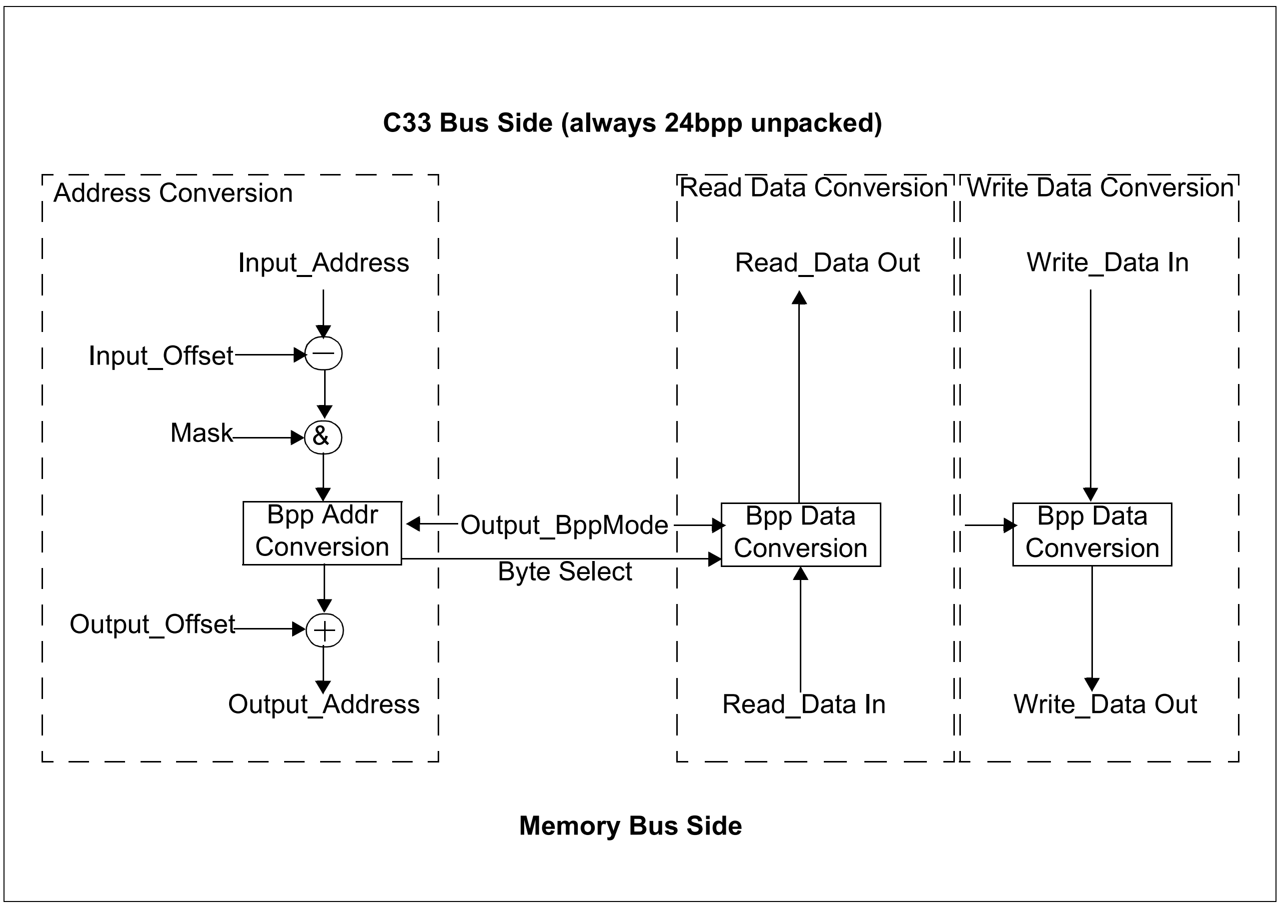


Figure 12-1: Functional Operation of Bit-Per-Pixel Conversion

Address conversion converts to bpp aligned address with byte enable.

Table 12-1: Address Conversion

Select	Conversion Mode	Input Address	Output Address	Output Byte Enable
0	No Conversion	Addr[31:0]	Addr[31:0]	1111
1	To 8bpp	Addr[31:0]	Addr[31:2]	Addr[1:0] = 00, 0001 Addr[1:0] = 01, 0010 Addr[1:0] = 10, 0100 Addr[1:0] = 11, 1000
2	To 16bpp	Addr[31:0]	Addr[31:1]	Addr[0] = 0, 0011 Addr[0] = 1, 1100

Write Data from C33 will always be 24bpp unpacked data, where ARGB data is stored in a 32-bit word. Depending on the conversion mode, the 32-bit data will be packed into 8bpp or 16bpp data by truncating the LSB of the full 24bpp data.

Table 12-2: Write Data Conversion

Select	Conversion Mode	Input Data	Output Data
0	No Conversion	A[7:0],R[7:0],G[7:0],B[7:0]	A[7:0],R[7:0],G[7:0],B[7:0]
1	To 8bpp	A[7:0],R[7:0],G[7:0],B[7:0]	R[7:5].G[7:5],B[7:6]
2	To 16bpp	A[7:0],R[7:0],G[7:0],B[7:0]	R[7:3].G[7:2],B[7:3]

Read data will require bit expansion from the configured data bpp mode to 32-bit ARGB data. Alpha byte may be just garbage data. RGB additional data bits are generated from the LSB of the compacted color channel bits. This method should spread out the missing gradients in between the color ramp.

12.1 System Level Connections

The Bit-Per-Pixel Converter is connected to the internal C33PE using a Memory Mapped Interface. When C33 needs a bpp conversion, it will write to a bpp converter register port which has been pre-setup by the C33 where it will map to a specific memory region.

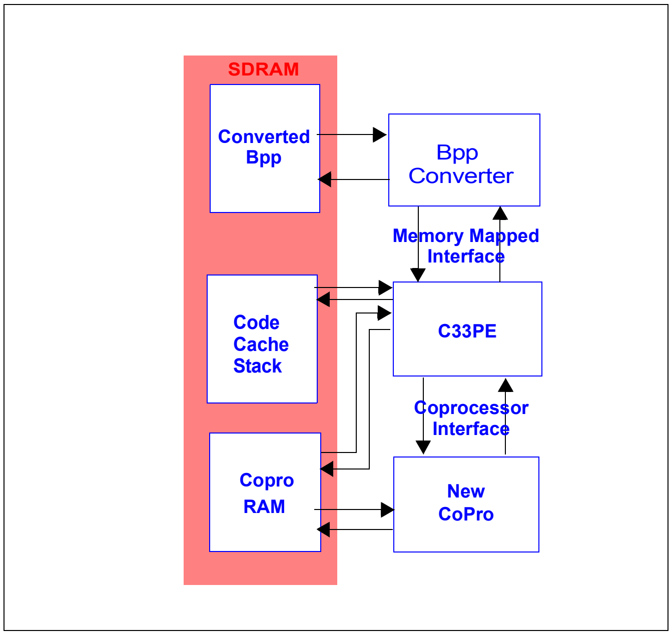


Figure 12-2: System Level Connection Block Diagram

Chapter 13 Display Subsystem

This section provides a high level description of the S1D13515/S2D13515's display subsystem.

Note
For XGA 1024x768 panel support, only single panel, single window with no virtual width function is supported (i.e. Blend Mode 0 with MAIN window only (AUX and OSD windows disabled) and Main Virtual Width, REG[0954h] ~ REG[0955h] is same as the Main Width, REG[0950h] REG[0951h]).
Any additional accesses to DRAM could potentially result in internal bandwidth limitations and must be evaluated on a case-by-case situation to ensure bandwidth throughput availability. The following table contains recommended values for XGA panel support.

Table 13-1: Recommended Settings for XGA Support

DRAM CLK (MHz)	PCLK (MHz)	HTOTAL (REG[4020h] ~ REG[4021h])	VTOTAL (REG[402Ah] ~ REG[402Bh])	Frame Rate (Hz)
100	60	1280	774	60
100	50	1056	774	60
100	65	1402	774	60

13.1 Block Diagram

The display subsystem consists of the following main subblocks:

- LCD Panel Interface
- Blending Engine
- Warp
- Image Fetcher
- Blending Engine CH1OUT Writeback
- Warp Writeback

The block diagram of the Display Subsystem is shown in Figure 13-1: "Display Subsystem Block Diagram," on page 450.

There are two panel interface outputs: LCD1 and LCD2. LCD1 supports generic TFT panels. LCD2 supports the same panels as LCD1, plus it also supports dual-image panel interfaces where frames from two image / stream sources are multiplexed into one frame image / stream (such as Epson's Double Screen panels, Sharp's Dual-View panels, or panels which display two views of the same image to create a 3D effect).

The LCD Panel Interface is the subblock which generates the proper timings for the panels. It has three input channels (streams of images): CH1IN, CH2IN, and OSDIN. The CH1 input stream is for LCD1. The source for the CH1 input is selectable between the Blending Engine CH1OUT output, Image Fetcher, and Warp. The CH2 and OSD input streams are for LCD2. The source for the CH2 input of the LCD Panel Interface is selectable between the CH2OUT and CH1OUT outputs of the Blending Engine. The source for the OSD input of the LCD Panel Interface is selectable between the OSDOUT and CH1OUT outputs of the Blending Engine. See Section 13.2.1, “LCD Panel Interface” on page 451 for more details.

The Blending Engine has three output streams which feed the LCD Panel Interface: CH1OUT, CH2OUT, and OSDOUT. It has three input window sources (images stored in SDRAM): MAIN, AUX, and OSD. The Blending Engine has four modes of operation which provide four different combinations of “blending” of the input windows onto the output streams. See Section 13.2.2, “Blending Engine” on page 456 for more details.

The Warp submodule reads frames from SDRAM and generates “warped” image frames which can be written back into SDRAM (through the Warp Writeblock submodule). See Section 13.2.3, “Warp Engine” on page 465 for more details.

The warped frames are written back into SDRAM at a slower rate and the Image Fetcher is used to fetch the warped frames from SDRAM at the rate of the panel. (Frames are repeated if there is no new frame available yet.) The Image Fetcher output goes to the CH1 input of the LCD Panel Interface. It can also be used in the case where there is only one window to be displayed (on LCD1) and the Blending Engine is not needed. See Section 13.2.6, “Image Fetcher” on page 468 for more details.

The CH1OUT Writeback submodule is mainly used to write “blended” frames (from the CH1OUT output of the Blending Engine) back into SDRAM to be processed by the Warp in order to generate “blended”, “warped” image / streams. See Section 13.2.4, “CH1OUT Writeback” on page 467 for more details.

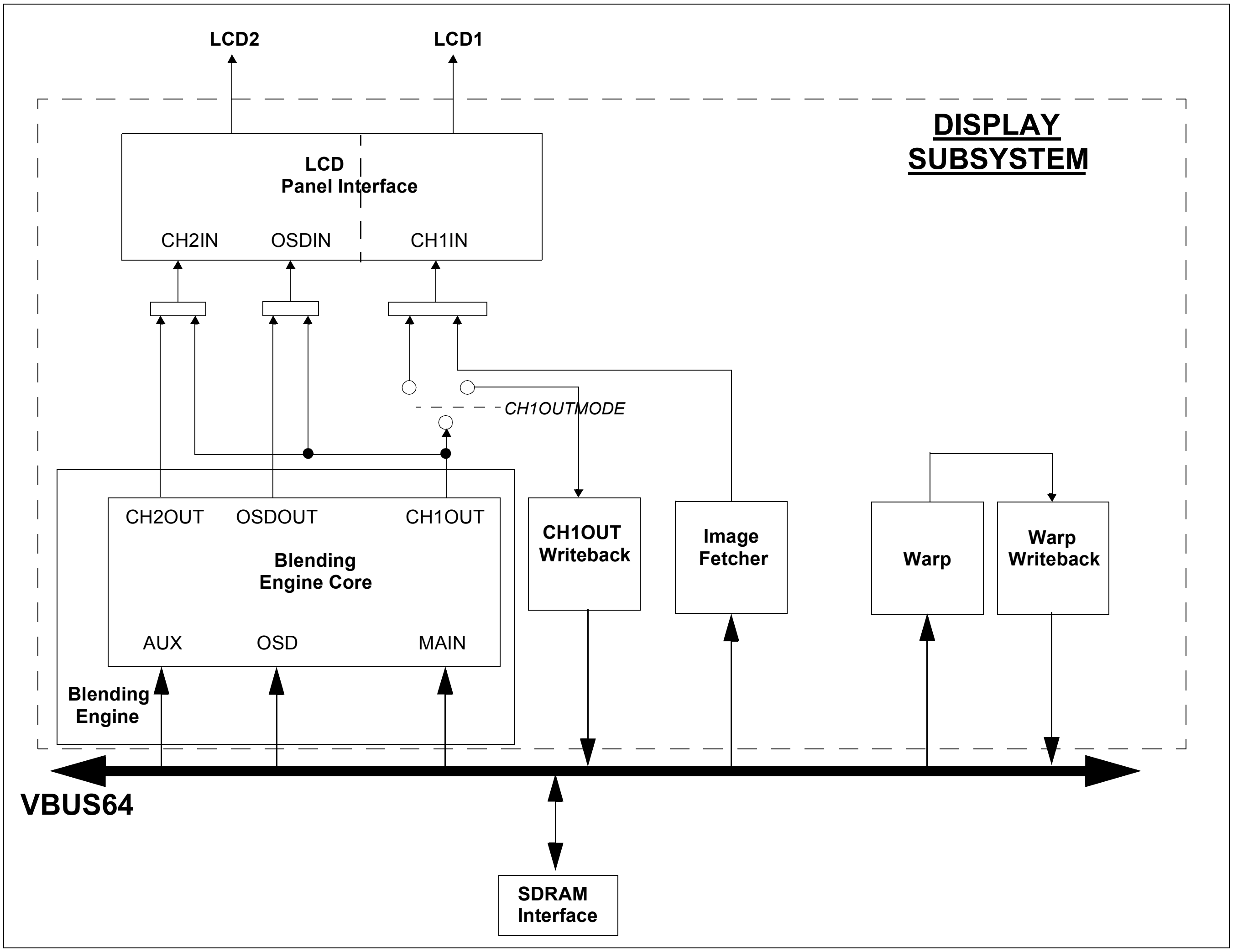


Figure 13-1: Display Subsystem Block Diagram

13.2 Hardware Blocks

13.2.1 LCD Panel Interface

The LCD Panel Interface has three input streams (CH1IN, CH2IN, and OSDIN) and two output panel interfaces (LCD1 and LCD2). The following shows a block diagram of the LCD Panel Interface subblock:

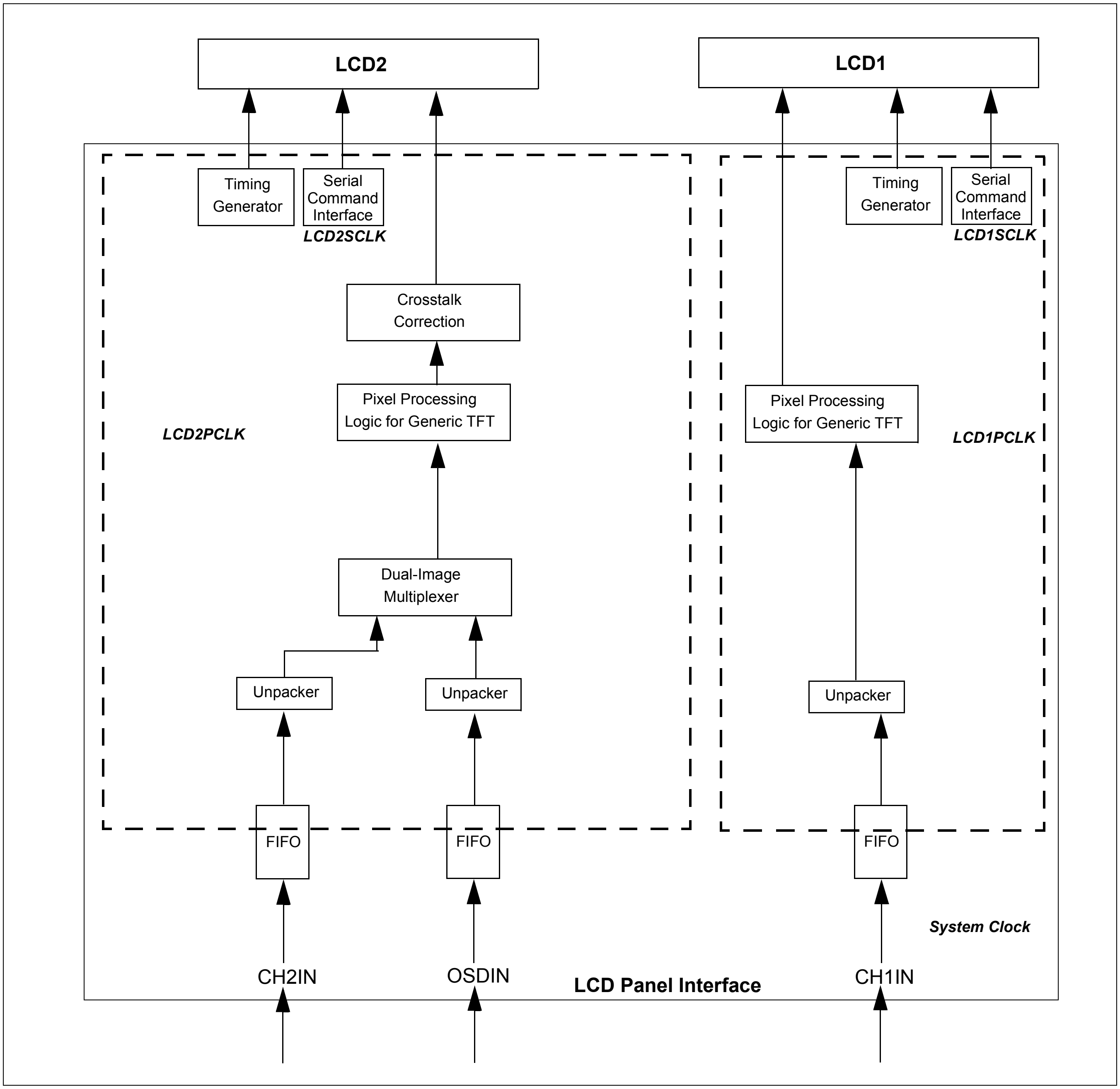


Figure 13-2: LCD Panel Interface Block Diagram

LCD1

The signals for the LCD1 panel interface are mapped to the FP1IO pins and are shared / multiplexed with the Camera2 interface. LCD1 supports the following types of panel interfaces:

- RGB Color TFT Panel
  - Generic TFT / TFD interface
  - 12 / 15 / 16 / 18-bit pixel data output modes
- Serial Command Interfaces
  - a-Si TFT interface (8-bit)
  - TFT w/u-Wire interface (16-bit)
  - EPSON ND-TFD 4 pin interface (8-bit)
  - EPSON ND-TFD 3 pin interface (9-bit)

To select LCD1 output function for the FP1IO pins, REG[4000h] bit 3 should be programmed to 0b.

There are four configurations of the FP1IO pins for LCD1 which depend on two factors:

1. Whether or not the LCD2 panel interface uses some of the FP1IO pins. The LCD2 panel interface uses FP1IO pins if all of the following settings are true:
   - REG[4000h] bits 5-4 = 01b (EID Double Screen panel interface is selected)
   - REG[4040h] bit 0 = 1b (EID Double Screen panel uses TCON signals)
   - REG[4000h] bit 1 = 0b (I2S / PWM pins are not used for TCON signals of EID Double Screen)
2. Whether or not the Serial Command interface is enabled for LCD1 (determined by REG[4000h] bit 2 - LCD1 Panel Mode Select).

If item 1 is false (LCD2 panel interface does not use FP1IO pins), the pixel data width of the LCD1 output is either 16-bit (REG[4000h] bit 2 = 1b, Serial Command interface is enabled for LCD1) or 18-bit (REG[4000h] bit 2 = 0b, Serial Command interface is disabled for LCD1).

Note
If the LCD1 interface pins are configured for 16-bit pixel data width, REG[4001h] bits 1-0 must be 01b or 10b. If the LCD1 interface pins are configured for 18-bit pixel data width, REG[4001h] bits 1-0 must be 10b.

If item 2 is true (LCD2 panel interface uses FP1IO pins), the pixel data width of the LCD1 output is either 12-bit (REG[4000h] bit 2 = 1b, Serial Command interface is enabled for LCD1) or 15-bit (REG[4000h] bit 2 = 0b, Serial Command interface is disabled for LCD1).

Note
If the LCD1 interface pins are configured for 12-bit pixel data width, REG[4001h] bits 1-0 must be 00b, 01b, or 10b. If the LCD1 interface pins are configured for 15-bit pixel data width, REG[4001h] bits 1-0 must be 01b or 10b.

The type of Serial Command interface for LCD1 (if enabled) is determined by REG[4016h] bits 7-5. Other control bits for the Serial Command interface for LCD1 are also programmed in REG[4016h]. Serial data for LCD1 Serial Command interface are written through REG[401Ch] ~ REG[401Fh].

Programmable parameters for the LCD1 panel interface output and timing are in REG[4002h] ~ REG[4015h], REG[4060h], REG[4080h] ~ REG[408Ch], and REG[40B0h].

Programmable parameters for the CH1IN input of the LCD Panel Interface are in REG[4062h] ~ REG[4065h].

LCD2

The signals for the LCD2 panel interface are mapped mainly to the FP2IO pins, but some of the FP1IO or I2S+PWM pins are also used if EID Double Screen panel interface with TCON signals is enabled. LCD2 supports the following types of panel interfaces:

- RGB Color TFT Panel
  - Generic TFT / TFD interface
  - 16 / 18-bit pixel data output
  - Single-image (regular) or dual-image (multiplexed, for Sharp Dual-View or Epson EID Double Screen panels) pixel data stream
- Serial Command Interfaces
  - a-Si TFT interface (8-bit)
  - TFT w/u-Wire interface (16-bit)
  - EPSON ND-TFD 4 pin interface (8-bit)
  - EPSON ND-TFD 3 pin interface (9-bit)

The LCD2 pins can be configured for 5 modes as follows:

Table 13-2: LCD2 Mode Configuration

	LCD2 Pin Mode					
	0	**1**	**2**	**3**		**4**
Pixel Data Width	24-bit	18-bit	18-bit	18-bit		18-bit
GPIOs Available	—	GPIO4 GPIO5	GPIO0 GPIO1 GPIO2 GPIO3 GPIO4 GPIO5	—		GPIO4 GPIO5
Serial Command Interface	No	Yes	No	No		No
Panel Interface Type	Generic RGB or EID Double Screen with TCON Disabled			EID Double Screen with TCON Enabled		Sharp Dual-View
REG[4000h] bits 5-4 REG[4040h] bit 0	REG[4000h] bits 5-4 = 00b or REG[4000h] bits 5-4 = 01b & REG[4040h] bit 0 = 0b			REG[4000h] bits 5-4 = 01b and REG[4040h] bit 0 = 1b		REG[4000h] bits 5-4 = 10b
REG[4000h] bits 7-6 (LCD2 Panel Mode)	00b (RGB 8:8:8 without Serial Command interface)	01b (RGB 6:6:6 with Serial Command interface)	10b (RGB 6:6:6 without Serial Command interface)	00b or 10b		00b or 10b
REG[4001h] bits 6-4 (LCD2 Panel Data Width)	011b (24-bit)	010b (18-bit) or 011b (24-bit)	010b (18-bit) or 011b (24-bit)	010b (18-bit) or 011b (24-bit)		010b (18-bit) or 011b (24-bit)
REG[4000h] bit 1	—			0b	1b	—
Pins Used for extra EID TCON signals	—			FP1IO	I2S / PWM	—

The type of Serial Command interface for LCD2 (if enabled) is determined by REG[4034h] bits 7-5. Other control bits for the Serial Command interface for LCD2 are also programmed in REG[4034h]. Serial data for LCD2 Serial Command interface are written through REG[403Ah] ~ REG[403Dh].

Programmable parameters for the LCD2 panel interface output and timing are in REG[4020h] ~ REG[4033h], REG[4070h], REG[4090h] ~ REG[409Ch], and REG[40B1h].

Programmable parameters for the CH2IN and OSDIN inputs of the LCD Panel Interface are in REG[4072h] ~ REG[4077h].

Programmable parameters for EID Double Screen panel mode are in REG[4040h] ~ REG[404Fh].

Programmable parameters for Sharp Dual-View panel mode are in REG[4050h] ~ REG[4056h].

In non-dual-image mode (REG[4000h] bits 5-4 = 00b), the image stream sent to LCD2 is from the CH2IN input of the LCD Panel Interface block.

In dual-image mode (REG[4000h] bits 5-4 = 01b or 10b), the pixel data of the image stream sent to the LCD2 output is interpreted by the panel as a multiplexed pixel data format as shown in Figure 13-3:

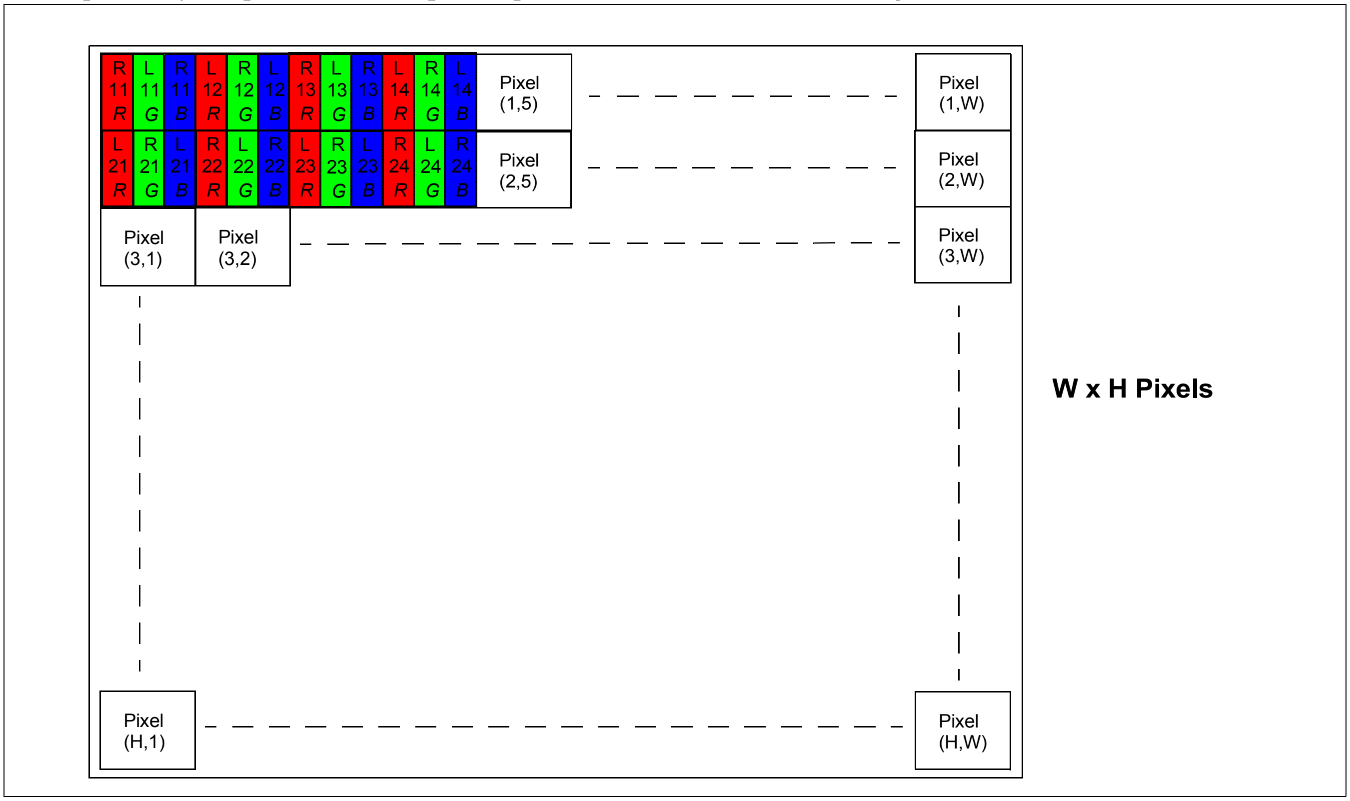


Figure 13-3: Dual-Image Multiplexed Pixel Data Format

A Left image and a Right image is defined for the display. In the first pixel, the Red and Blue data is for pixel (1,1) of the Right image and the Green data is for pixel (1,1) of the Left image. In the second pixel, the Red and Blue data is for pixel (1,2) of the Left image and the Green data is for pixel (1,2) of the Right image. The Red and Blue data for pixel (1,2) of the Left image is copied / used as the Red and Blue data for pixel (1,1) of the Left image, and the Green data for pixel (1,1) of the Left image is copied / used as the Green data for pixel(1,2) of the Left image. Similarly, Red, Green, and Blue data for pixels (1,1) and (1,2) of the Right image are also shared. The Left / Right pixel data multiplexing continues for the rest of the frame. (Half of the pixel data from the image sources are thrown away by the LCD Panel Interface block.)

When LCD2 is programmed for dual-image output, the Left and Right image source can be configured with the following four selections by programming REG[4073h] bits 5-4:

Table 13-3: Dual-Image Source Selection

REG[4073h] bits 5-4	LEFT Image	RIGHT Image
00b	CH2IN	CH2IN
01b	OSDIN	OSDIN
10b	OSDIN	CH2IN
11b	CH2IN	OSDIN

Clocks

The inputs sources (CH1IN, CH2IN, OSDIN) of the LCD Panel Interface run on the System Clock. Each of the two panel interface outputs (LCD1 and LCD2) run on their own independently programmable pixel clock. The FIFOs at the input of the LCD Panel Interface are used to buffer pixel data between the System Clock domain and the LCD1 / LCD2 clock domains. The Serial Command interfaces for LCD1 and LCD2 also have independently programmable clock frequencies.

The pixel and serial clocks for the panel interfaces are derived from the LCD clock path (PLL2 output or CLKI / OSC input). The divide ratio for the pixel clock of LCD1 (LCD1PCLK) is programmable through REG[0030h] and the divide ratio for the pixel clock of LCD2 (LCD2PCLK) is programmable through REG[0031h]. The divide ratio for the serial clock of LCD1 (LCD1SCLK) is programmable through REG[0032h] and the divide ratio for the serial clock of LCD2 (LCD2SCLK) is programmable through REG[0033h].

13.2.2 Blending Engine

The Blending Engine has three image stream output pipes / channels: CH1OUT, CH2OUT, and OSDOUT.

The following figure shows a block diagram of the Blending Engine.

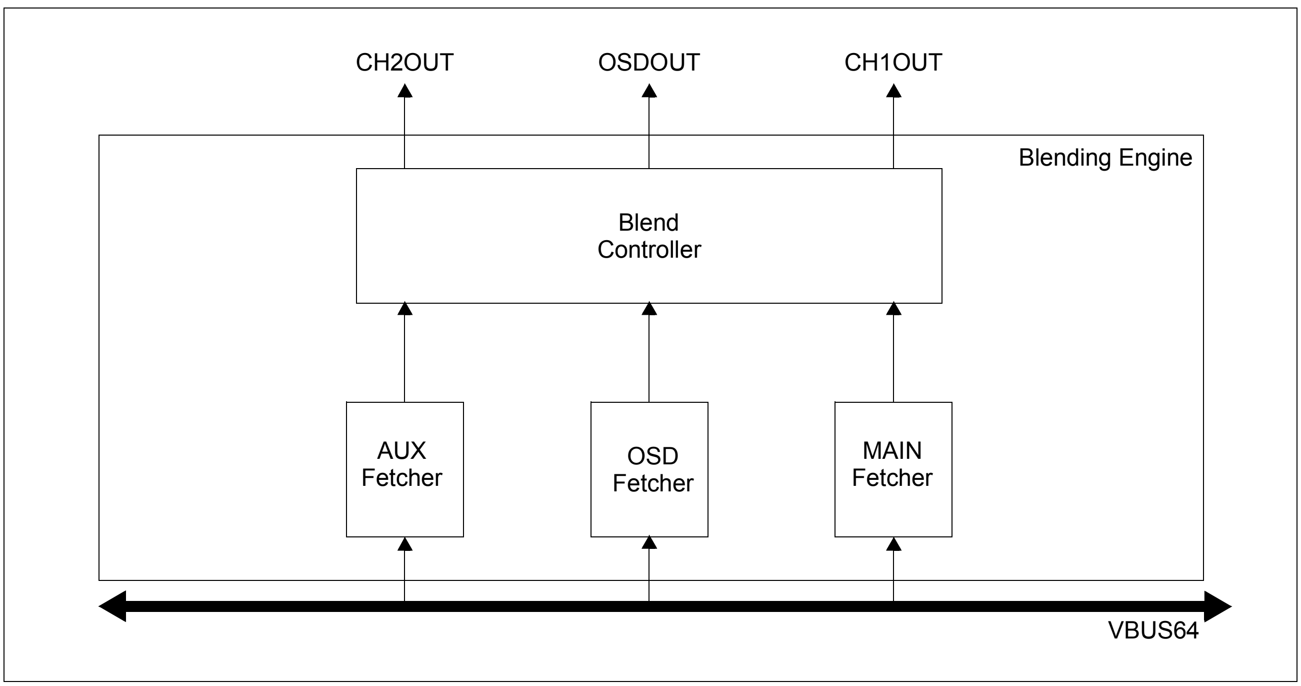


Figure 13-4: Blending Engine Block Diagram

There are three separate source windows (frames / images in SDRAM) defined for the Blending Engine: MAIN, AUX, and OSD. The Blending Engine has three separate input fetching buffers / pipes for these three windows and it combines the windows to generate three separate output streams, CH1OUT, CH2OUT, and OSDOUT, respectively. There are four modes of operation for the Blending Engine as follows.

Table 13-4: Modes of Operation for the Blending Engine

MODE	CH1OUT	CH2OUT	OSDOUT	NOTES
0	MAIN + AUX + OSD	—	—	MAIN is always at the bottom. OSD can be on top of AUX or vice versa (register programmable). There are register bits to turn on / off the OSD and AUX windows.
1	MAIN + OSD	AUX	—	OSD cannot be in both MAIN and AUX. There is a register bit to turn on / off the OSD window.
2	MAIN	AUX + OSD	—	OSD cannot be in both MAIN and AUX. There is register bit to turn on / off the OSD window.
3	MAIN	AUX	OSD	There is no “blending”. The 3 input streams are fed to the 3 output streams.

The Blend Mode is programmable through REG[09A0h] bits 1-0.

Note
The size of the OSD or AUX window when it is a sub-window, must be smaller than the background window.

MODE 0

In MODE 0 (REG[09A0h] bits 1-0 = 00b), only the CH1OUT output is on and the MAIN, AUX, and OSD windows are overlaid on top of each other with the MAIN window (background) at the bottom. The AUX and OSD windows should be less than or equal to the size of the MAIN window.

- The width and height of the MAIN window is programmed through REG[0950h] ~ REG[0951h] and REG[0952h]-REG[0953h], respectively.
- The width and height of the AUX window is programmed through REG[0970h] ~ REG[0971h] and REG[0972h] ~ REG[0973h], respectively.
- The width and height of the OSD window is programmed through REG[0990h] ~ REG[0991h] and REG[0992h] ~ REG[0993h], respectively.
- The AUX window is programmable to be above or below the OSD window through REG[09A0h] bit 2.
- The CH1OUT output is turned on / off through REG[0900h] bit 0.
- The AUX window can be turned on / off through REG[0960h] bit 4 and the OSD window can be turned on / off through REG[0980h] bit 4.
- The position (X and Y offsets, in pixels) of the AUX and OSD windows within the MAIN window are independently programmable. For the AUX window, the X offset is programmable through REG[0976h] ~ REG[0977h] and the Y offset is programmable through REG[0978h] ~ REG[0979h]. For the OSD window, the X offset is programmable through REG[0996h] ~ REG[0997h] and the Y offset is programmable through REG[0998h] ~ REG[0999h].

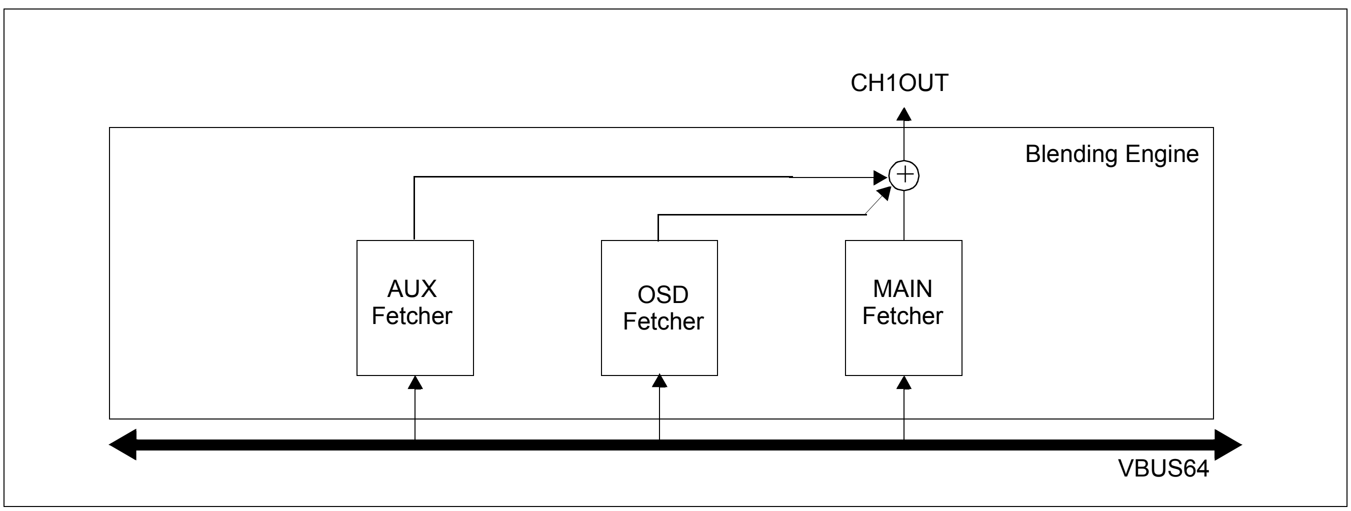


Figure 13-5: Blend Mode 0 Display Path

MODE 1

In MODE1 (REG[09A0h] bits 1-0 = 01b), the CH1OUT and CH2OUT outputs are on and the OSDOUT output is off. The CH1OUT output is the OSD window overlaid on top of the MAIN window, and the CH2OUT output is the AUX window. The OSD window can be less than or equal to the size of the MAIN window.

- The width and height of the MAIN window is programmed through REG[0950h] ~ REG[0951h] and REG[0952h] ~ REG[0953h], respectively.
- The width and height of the AUX window is programmed through REG[0970h] ~ REG[0971h] and REG[0972h] ~ REG[0973h], respectively.
- The width and height of the OSD window is programmed through REG[0990h] ~ REG[0991h] and REG[0992h] ~ REG[0993h], respectively.
- The CH1OUT output is turned on / off through REG[0900h] bit 0. The OSD window can be turned on / off through REG[0980h] bit 4.
- The CH2OUT output is turned on / off through REG[0920h] bit 0.
- The X and Y offsets of the OSD window within the MAIN window is programmable through REG[0996h] ~ REG[0997h] and REG[0998h] ~ REG[0999h], respectively.

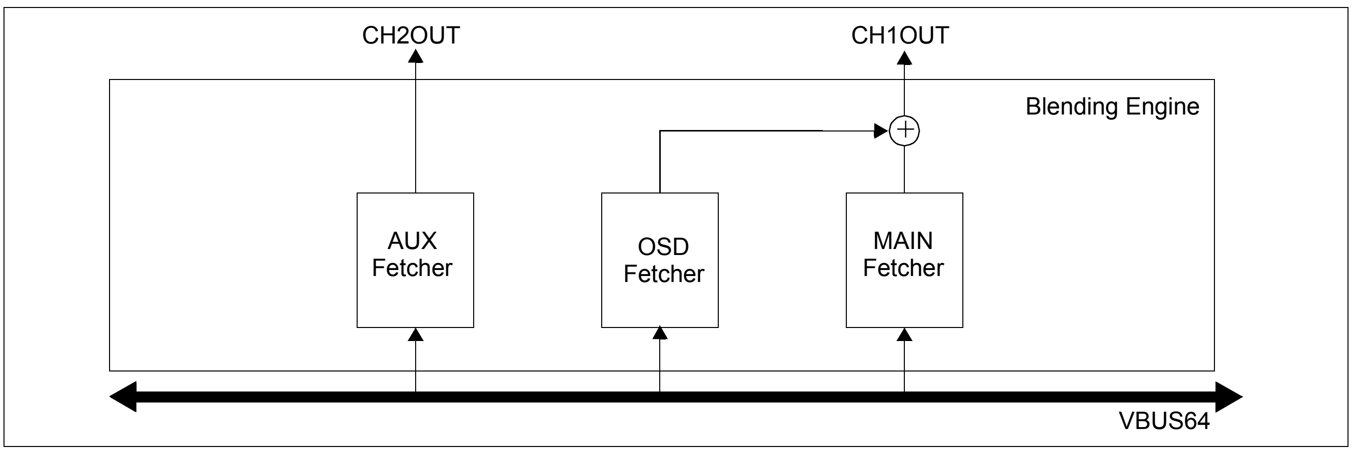


Figure 13-6: Blend Mode 1 Display Path

MODE 2

In MODE2 (REG[09A0h] bits 1-0 = 10b), the CH1OUT and CH2OUT outputs are on and the OSDOUT output is off. The CH1OUT output is the MAIN window, and the CH2OUT output is the OSD window overlaid on top of the AUX window. The OSD window can be less than or equal to the size of the AUX window.

- The width and height of the MAIN window is programmed through REG[0950h] ~ REG[0951h] and REG[0952h] ~ REG[0953h], respectively.
- The width and height of the AUX window is programmed through REG[0970h] ~ REG[0971h] and REG[0972h] ~ REG[0973h], respectively.
- The width and height of the OSD window is programmed through REG[0990h] ~ REG[0991h] and REG[0992h] ~ REG[0993h], respectively.
- The CH1OUT output is turned on / off through REG[0900h] bit 0.
- The CH2OUT output is turned on / off through REG[0920h] bit 0. The OSD window can be turned on / off through REG[0980h] bit 4.
- The X and Y offsets of the OSD window within the AUX window is programmable through REG[0996h] ~ REG[0997h] and REG[0998h] ~ REG[0999h], respectively.

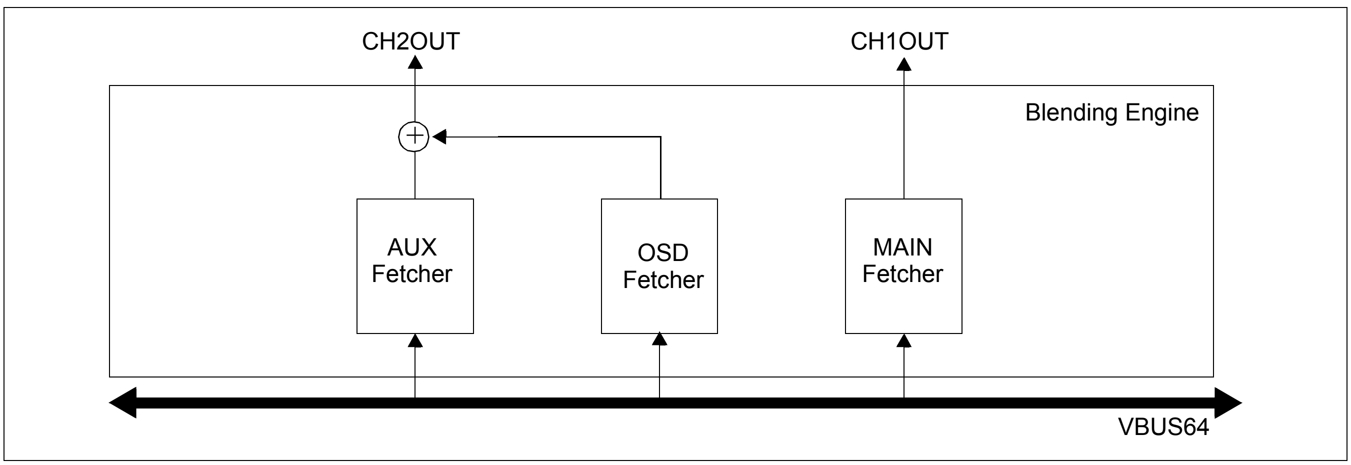


Figure 13-7: Blend Mode 2 Display Path

MODE 3

In MODE3 (REG[09A0h] bits 1-0 = 11b), the CH1OUT, CH2OUT, and OSDOUT outputs are all on. The CH1OUT output is the MAIN window, the CH2OUT output is the AUX window, and the OSDOUT output is the OSD window. (Each output only has a single window and no overlays.)

- The width and height of the MAIN window is programmed through REG[0950h] ~ REG[0951h] and REG[0952h] ~ REG[0953h], respectively.
- The width and height of the AUX window is programmed through REG[0970h] ~ REG[0971h] and REG[0972h] ~ REG[0973h], respectively.
- The width and height of the OSD window is programmed through REG[0990h] ~ REG[0991h] and REG[0992h] ~ REG[0993h], respectively.
- The CH1OUT output is turned on / off through REG[0900h] bit 0.
- The CH2OUT output is turned on / off through REG[0920h] bit 0.
- The OSDOUT output is t turned on / off through REG[0930h] bit 0.

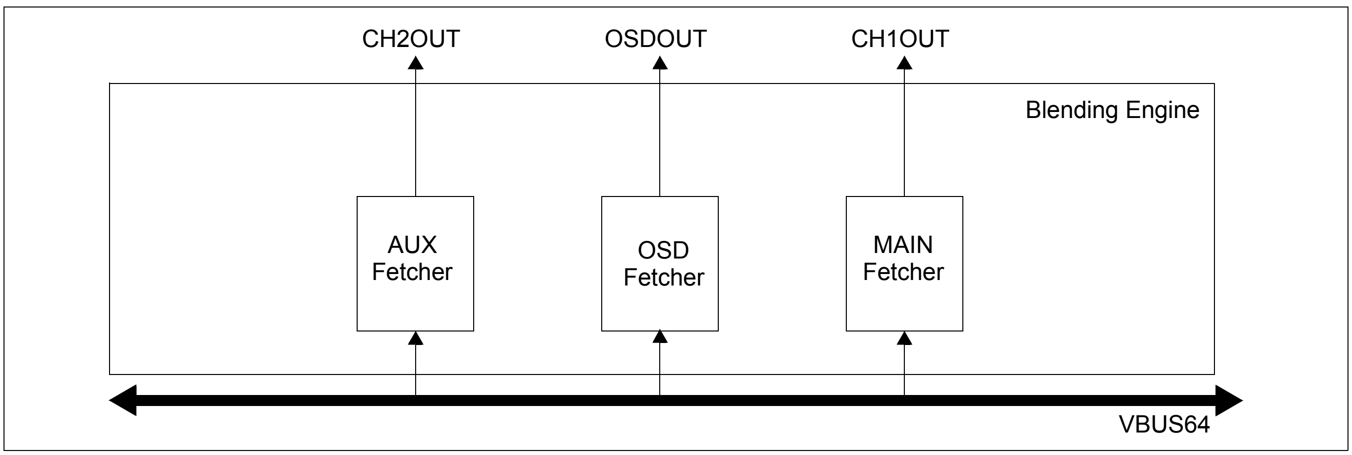


Figure 13-8: Blend Mode 3 Display Path

CH1OUT / CH2OUT / OSDOUT Pixel Formats

The pixel format of the CH2OUT output of the Blending Engine is determined by the CH2IN pixel format of the LCD Panel Interface block (REG[4072h] bits 2-0). The pixel format of the OSDOUT output of the Blending Engine is determined by the OSDIN pixel format of the LCD Panel Interface block (REG[4073h] bits 2-0).

The pixel format of the CH1OUT output of the Blending Engine is determined according to the destination of its image stream as follows:

- If the CH1OUT image stream is routed to the CH1OUT Writeback block (REG[0900h] bit 1 = 1b), the pixel format is determined by the CH1OUT Writeback pixel format bits (REG[0900h] bits 5-4).
- If the CH1OUT image stream is routed to the CH1IN input of the LCD Panel Interface (REG[0900h] bit 1 = 0b, REG[09C8h] bits 1-0 = 00b), the pixel format is determined by the CH1IN pixel format of the LCD Panel Interface (REG[4062h] bits 2-0).
- If the CH1OUT image stream is routed to the CH2IN input of the LCD Panel Interface (REG[0900h] bit 1 = 0b, REG[09C8h] bits 1-0 is not 00b, REG[09C8h] bit 2 = 1b), the pixel format is determined by the CH2IN pixel format of the LCD Panel Interface (REG[4072h] bits 2-0).
- If the CH1OUT image stream is routed to the OSDIN input of the LCD Panel Interface (REG[0900h] bit 1 = 0b, REG[09C8h] bits 1-0 is not 00b, REG[09C8h] bit 2 = 0b, REG[09C8h] bit 3 = 1b), the pixel format is determined by the OSDIN pixel format of the LCD Panel Interface (REG[4073h] bits 2-0).

MAIN / AUX / OSD Programmable Parameters

Each of the three source windows (MAIN, AUX, OSD) of the Blending Engine has a dedicated pixel fetcher. For each pixel fetcher, a set of two frame buffers are defined: BUFFER0 and BUFFER1. These two buffers are used in a frame double-buffering scheme (described in a later section) to ensure “tear-free” transition when displaying one frame to the next.

The frame buffer addresses for the MAIN, AUX, and OSD windows are specified in the following registers:

- REG[0948h] ~ REG[094Bh] = MAIN Buffer0 Start address
- REG[094Ch] ~ REG[094Fh] = MAIN Buffer1 Start address
- REG[0968h] ~ REG[096Bh] = AUX Buffer0 Start address
- REG[096Ch] ~ REG[096Fh] = AUX Buffer1 Start address
- REG[0988h] ~ REG[098Bh] = OSD Buffer0 Start address
- REG[098Ch] ~ REG[098Fh] = OSD Buffer1 Start address

Generally, MAIN registers are located in REG[094xh] ~ REG[095xh], AUX registers are located in REG[096xh] ~ REG[097xh], and OSD registers are located in REG[098xh] ~ REG[099xh].

The following are other programmable parameters for the MAIN / AUX / OSD windows:

- The pixel format for the MAIN / AUX / OSD window is specified by bits 3-2 in REG[0940h] / REG[0960h] / REG[0980h]. The formats are RGB 3:3:2, RGB 5:6:5, and RGB 8:8:8. In addition, the OSD window also supports the following alpha-blending formats: ARGB 4:4:4:4, ARGB 1:5:5:5, and ARGB 8:5:6:5. The alpha-blending formats for the OSD window / layer is enabled by REG[09A0h] bit 3 and ARGB format is selected by bits 3-2 of REG[0980h]. See Section , “Alpha-Blending for OSD Layer” on page 465 for more details on alpha-blending for the OSD layer.
- The MAIN / AUX / OSD window image can be “blanked” (filled with a constant pixel color) setting bit 0 of REG[0940h] / REG[0960h] / REG[0980h] to 1b. The “blank” color is specified in REG[0944h] ~ REG[0946h] / REG[0964h] ~ REG[0966h] / REG[0984h] ~ REG[0986h]. Note that the pixel fetchers will continue to fetch pixels from the frame buffer but the pixel data is not forwarded (“absorbed”) and is replaced by the “blank” color.
- The MAIN / AUX / OSD window can be flipped vertically (around the x-axis) by setting bit 5 of REG[0940h] / REG[0960h] / REG[0980h] to 1b.
- The MAIN / AUX / OSD window can be flipped horizontally (around the y-axis) by setting bit 6 of REG[0940h] / REG[0960h] / REG[0980h] to 1b.
- The MAIN / AUX / OSD window can be set to “line-double” mode by setting bit 7 of REG[0940h] / REG[0960h] / REG[0980h] to 1b. In this mode, the source image stored in SDRAM only has half the number of lines displayed for the window and each line of the source image is repeated twice. This is mainly for displaying interlaced images which are written by the Camera Interface to SDRAM.
- The source image for the MAIN / AUX / OSD window can be a larger virtual image than the displayed image. This allows the displayed MAIN / AUX / OSD window to “pan” within a larger source image. The Virtual Width Register (REG[0954h] ~ REG[0955h] / REG[0974h] ~ REG[0975h] / REG[0994h] ~ REG[0995h]) is used to determine the address jump to go to the next line of the source image. The Input X Offset register (REG[095Ah]

~ REG[095Bh] / REG[097Ah] ~ REG[097Bh] / REG[099Ah] ~ REG[099Bh]) specifies the X offset and the Input Y Offset register (REG[095Ch] ~ REG[095Dh] / REG[097Ch] ~ REG[097Dh] / REG[099Ch] ~ REG[099Dh]) specifies the Y offset relative to the top left corner of the virtual / larger source image for the top left corner of the displayed image. Figure 13-9: “Virtual Source Window Example” shows an example.

- The image for the MAIN / AUX / OSD window can be stored either as “line-by-line” organization or as “tiled frame” organization. Bit 1 of REG[0940h] / REG[0960h] / REG[0980h] specifies the memory organization of pixels for the frame. See Section 13.3, “Memory Organization of Frames” on page 470 for more details on these two types of organization of pixels in memory.

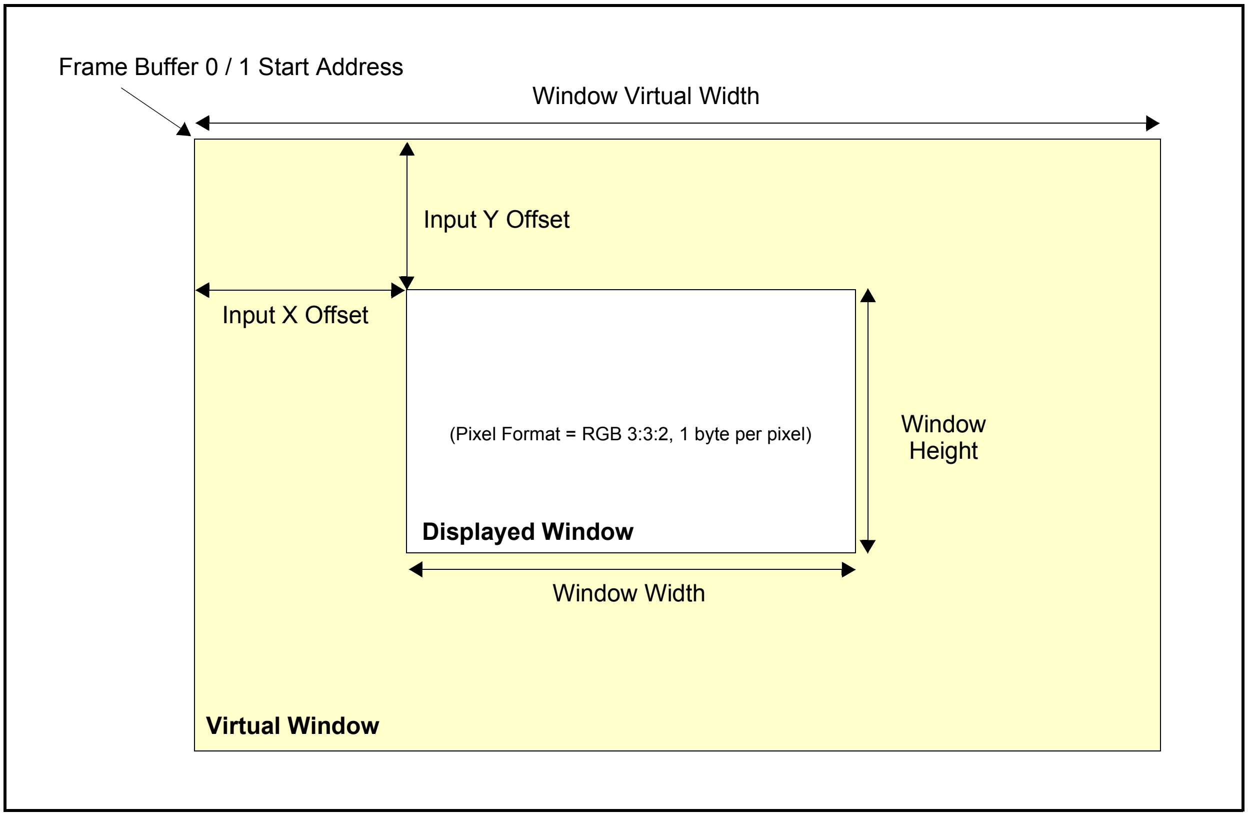


Figure 13-9: Virtual Source Window Example

Note
The size of the OSD or AUX window when it is a sub-window, must be smaller than the background window.

Alpha-Blending for OSD Layer

The OSD window / layer supports alpha-blending with the MAIN and AUX layers for mode 0, 1 or 2 of the Blending Engine.

In Mode0 of the Blending Engine, the OSD layer is alpha-blended with the layers below it. If the “AUX on Top” bit (REG[09A0h] bit 2) is set to 0, the OSD layer is alpha-blended with the AUX and MAIN layers below it. If the “AUX on Top” bit is 1, the OSD layer is only alpha-blended with the MAIN layer.

In Mode1 of the Blending Engine, the OSD layer is alpha-blended with the MAIN layer, and in Mode2 of the Blending Engine, the OSD layer is alpha-blended with the AUX layer.

There are two modes for alpha-blending of the OSD layer which is determined by the OSD Alpha Format Enable bit (REG[09A0h] bit 3). If this bit is 0b, the pixel format for the OSD source image is RGB 3:3:2 / RGB 5:6:5 / RGB 8:8:8 and does not have alpha value. A common alpha is applied to all the pixels for the OSD layer by programming REG[09A1h] (8-bit alpha value). If this bit is 1b, the pixel format for the OSD source image is ARGB 4:4:4:4 / ARGB 1:5:5:5 / ARGB 8:5:6:5 and has alpha value for each individual pixel.

The alpha-blending logic works with 8-bit alpha values. An alpha value of FFh means that the OSD pixel is fully turned on (on top). An alpha value of 0 means the OSD pixel is fully turned off. For the ARGB 4:4:4:4 format, the alpha value is only 4 bits, and the 4 bits are duplicated / concatenated to produce the 8-bit alpha value (lower and upper 4 bits are the same). For the ARGB 1:5:5:5 format, the alpha value is only 1-bit. If the bit is 0, an 8-bit alpha value of 00h is generated. If the bit is 1, the alpha value generated is selectable between 50% (80h) and 75% (C0h) by programming the ARGB 1:5:5:5 Alpha Ratio Select bit (REG[09A0h] bit 4).

13.2.3 Warp Engine

The Warp engine generates a warped version of a source frame / image from SDRAM. The Warp engine can also generate luminance effects on the output image to brighten selective areas / blocks. The warped frames can be written back to another location in SDRAM (through the Warp Writeback block).

In the case where the Warp block writes frames back to SDRAM, the Image Fetcher is used to fetch the warped frames at the LCD panel refresh rate (typically 60Hz) to feed to the LCD Panel Interface. The Blending Engine and Warp processing can run at a lower frame rate in order to conserve bandwidth demand while the Image Fetcher runs at the higher frame rate to keep up with the panel’s refresh rate requirement.

Warping Operation

The Warp engine divides the output image into NxM pixel blocks. Each NxM block of pixels is assigned a (X,Y) offset value which is used to determine where in the input image source to fetch the pixel. (The pixels in each NxM block share a common (X,Y) offset value.) An Offset Table is used to specify the (X,Y) offset values for all the blocks in the output image. If the calculated coordinate of the input pixel to fetch is outside the input source image’s boundaries, a programmable background (“filler”) pixel color is used instead. The values in the Offset Table determine the warping characteristics of the output image.

The input source image size can be greater than the output image size and there are programmable Input X Offset and Y Offset registers which let the output image “pan” the larger input image.

Luminance Operation

The luminance operation is applied on the output image after warping. The output image is also divided into NxM blocks with the pixels in each block assigned a common luminance (pixel brightness) value. A Luminance Table is used to specify the luminance values for all the blocks in the output image. The Luminance Table can be used to provide brighten / darken effects on each NxM block.

Warp Programming

The following are programmable registers for the warping operation:

- REG[0400h] bit 0 enables / disables the warp operation.
- REG[0400h] bit 4 specifies whether or not bilinear smoothing (averaging with neighboring pixels) is enabled for the warp operation.
- REG[0444h] ~ REG[0447h] specifies the address location of the Offset table in SDRAM.
- REG[0440h] specifies the block size for warp operation.
- REG[0420h] ~ REG[0423h] = Input Image Buffer0 Start address
- REG[0424h] ~ REG[0427h] = Input Image Buffer1 Start address
- REG[0414h] ~ REG[0415h] specifies the width of the Warp engine’s output image.
- REG[0416h] ~ REG[0417h] specifies the height of the Warp engine’s output image.
- REG[0410h] ~ REG[0411h] specifies the width of the Warp engine’s input image.
- REG[0412h] ~ REG[0413h] specifies the height of the Warp engine’s input image.
- REG[0434h] ~ REG[0435h] specifies the X Offset (in pixels), relative to the top left corner of the input source image, where the top left corner of the panning window for the output image is located.
- REG[0436h] ~ REG[0437h] specifies the Y Offset (in pixels), relative to the top left corner of the input source image, where the top left corner of the panning window for the output image is located.
- The pixel format for the Warp engine’s input / output images is specified by REG[0400h] bit 3 (RGB 3:3:2 or RGB 5:6:5).
- REG[0430h] ~ REG[0432h] specify the Background color for the Warp engine.

Luminance Programming

The following are programmable registers for the luminance operation:

- REG[0400h] bit 1 enables / disables the luminance effect.
- REG[0400h] bit 5 specifies whether or not bilinear smoothing (averaging with neighboring pixels) is enabled for the luminance operation.
- REG[0454h] ~ REG[0457h] specifies the address location of the Luminance table in SDRAM.
- REG[0450h] specifies the block size for the luminance operation.
- REG[0452h] bit 0 specifies whether or not luminance effect is applied to black pixels.

- REG[0452h] bit 1 specifies whether or not luminance effect is applied to the Background color. (The Background color is used if the calculated input pixel location is outside the boundaries of the input source image.)

13.2.4 CH1OUT Writeback

The CH1OUT output image stream of the Blending Engine can be written back to SDRAM through the CH1OUT Writeback block. This path can be used in cases where a “blended” image stream (for example, Blend Mode = 0) will be post-processed (such as warping) before displaying on the LCD panel.

The CH1OUT Writeback has the following programmable features:

- REG[0904h] ~ REG[0907h] = CH1OUT Writeback Buffer0 Start address (for writing)
- REG[0904h] ~ REG[0907h] = CH1OUT Writeback Buffer1 Start address (for writing)
- The CH1OUT Writeback block is turned on when the CH1OUT output of the Blending Engine is turned on (REG[0900h] bit 0 = 1b) and the “CH1OUT Mode” bit is set to 1 (REG[0900h] bit 1 = 1b).
- The pixel format for the CH1OUT Writeback is specified by REG[0900h] bits 5-4.
- The output image stream of the CH1OUT Writeback can be flipped vertically (around the x-axis) by setting bit 3 of REG[0900h] to 1b.
- The output image stream of the CH1OUT Writeback can be stored either as “line-by-line” organization or as “tiled frame” organization. Bit 2 of REG[0900h] specifies the memory organization of pixels for the frame. See Section 13.3, “Memory Organization of Frames” on page 470 for more details on these two types of organization of pixels in memory.

13.2.5 Warp Writeback

The output image stream of the Warp block is written back to SDRAM through the Warp Writeback block. This path can be used to avoid limitations in the VBUS64 bandwidth which may not allow the Warp block to keep up with the frame rate of the LCD Panel Interface. The warped image stream can be written back to SDRAM at a slower frame rate and the MAIN, AUX, OSD or Image Fetcher block can be used to display the warped image stream at the higher panel frame refresh rate.

The Warp Writeback has the following programmable features:

- REG[09D0h] ~ REG[09D3h] = Warp Writeback Buffer0 Start address (for writing)
- REG[09D4h] ~ REG[09D7h] = Warp Writeback Buffer1 Start address (for writing)
- The pixel format for the Warp Writeback is determined by the Warp block's pixel format bit (REG[0400h] bit 3).
- The output image stream of the Warp Writeback can be flipped vertically (around the x-axis) by setting bit 5 of REG[09CAh] to 1b.
- The output image stream of the Warp Writeback can be stored either as "line-by-line" organization or as "tiled frame" organization. Bit 7 of REG[09CAh] specifies the memory organization of pixels for the frame. See Section 13.3, "Memory Organization of Frames" on page 470 for more details on these two types of organization of pixels in memory.

13.2.6 Image Fetcher

The Image Fetcher is mainly used in the case where the Warp is enabled and the bandwidth limitation on VBUS64 does not allow the Warp to process frames at the panel's refresh rate. In this case, the Warp Writeback is used to write the output frames of the Warp back to SDRAM at a slower frame rate and the warped image is displayed by the Image Fetcher at the rate of the panel. The Image Fetcher can also just be used as a general-purpose single-window (no overlay windows) pixel fetcher.

The Image Fetcher has the following programmable features:

- REG[09C0h] ~ REG[09C1h] specifies the width of the Image Fetcher's output image.
- REG[09C2h] ~ REG[09C3h] specifies the height of the Image Fetcher's output image.
- REG[09B8h] ~ REG[09BBh] = Image Fetcher Buffer0 Start address
- REG[09BCh] ~ REG[09BFh] = Image Fetcher Buffer1 Star address
- The Image Fetcher output is turned on / off through REG[09B0h] bit 4.
- The pixel format for the Image Fetcher image is determined by the CH1IN input format register (REG[4062h] bits 2-0) of the LCD Panel Interface.
- The Image Fetcher image can be "blanked" (filled with a constant pixel color) by setting bit 0 of REG[09B0h] to 1b. The "blank" color is specified in REG[09B4h] ~ REG[09B6h]. Note that the Image Fetcher will continue to fetch pixels from the frame buffer but the pixel data is not forwarded ("absorbed") and is replaced by the "blank" color.
- The Image Fetcher image can be flipped vertically (around the x-axis) by setting bit 5 of REG[09B0h] to 1b.
- The Image Fetcher image can be flipped horizontally (around the y-axis) by setting bit 6 of REG[09B0h] to 1b.

- The Image Fetcher image can be set to “line-double” mode by setting bit 7 of REG[09B0h] to 1b. In this mode, the source image stored in SDRAM only has half the number of lines displayed and each line of the source image is repeated twice. This is mainly for displaying interlaced images which are written by the Camera Interface to SDRAM.
- The source image for the Image Fetcher can be a larger virtual image than the displayed image. This allows the displayed Image Fetcher image to “pan” within a larger source image. The Virtual Width Register (REG[09C4h] ~ REG[09C5h]) is used to determine the address jump to go to the next line of the source image. The Input X Offset register (REG[09AAh] ~ REG[09ABh]) specifies the X offset and the Input Y Offset register (REG[09ACh] ~ REG[09ADh]) specifies the Y offset relative to the top left corner of the virtual / larger source image for the top left corner of the displayed image.
- The image for the Image Fetcher can be stored either as “line-by-line” organization or as “tiled frame” organization. Bit 1 of REG[09B0h] specifies the memory organization of pixels for the frame. See Section 13.3, “Memory Organization of Frames” on page 470 for more details on these two types of organization of pixels in memory.

13.2.7 Input Selectors for LCD Panel Interface

The S1D13515/S2D13515 has programmable registers to select the image streams which feed into the CH1IN, CH2IN, and OSDIN inputs of the LCD Panel Interface block.

The image stream for the CH1IN input of the LCD Panel Interface is selectable, by programming REG[09C8h] bits 1-0, between the following three sources:

- CH1OUT output of Blending Engine (REG[09C8h] bits 1-0 = 00b)
- Output of Warp block (REG[09C8h] bits 1-0 = 01b)
- Output of Image Fetcher (REG[09C8h] bits 1-0 = 10b)

The image stream for the CH2IN input of the LCD Panel Interface is selectable, by programming REG[09C8h] bit 2, between the following two sources:

- CH2OUT output of Blending Engine (REG[09C8h] bit 2 = 0b)
- CH1OUT output of Blending Engine (REG[09C8h] bit 2 = 1b)

The image stream for the OSDIN input of the LCD Panel Interface is selectable, by programming REG[09C8h] bit 3, between the following two sources:

- OSDOUT output of Blending Engine (REG[09C8h] bit 3 = 0b)
- CH1OUT output of Blending Engine (REG[09C8h] bit 3 = 1b)

13.3 Memory Organization of Frames

Frames / images are stored in display memory (SDRAM) either in the traditional “line-by-line” addressing or the “tiled-frame” addressing.

13.3.1 “Line-by-Line” Image Storage

In this method of storing images, pixels are stored on a line-by-line basis. The top left pixel of the frame image is stored at address offset 0 and address increases as we go from left to right on the first line of the frame. When we reach the end of the first line, the next pixel stored in memory would be the left-most pixel of the second line, and so on and so on. This is the traditional method of storage.

The following diagram shows an example of how a 64x32 frame with **8 bits per pixel** is stored in memory using the line-by-line method

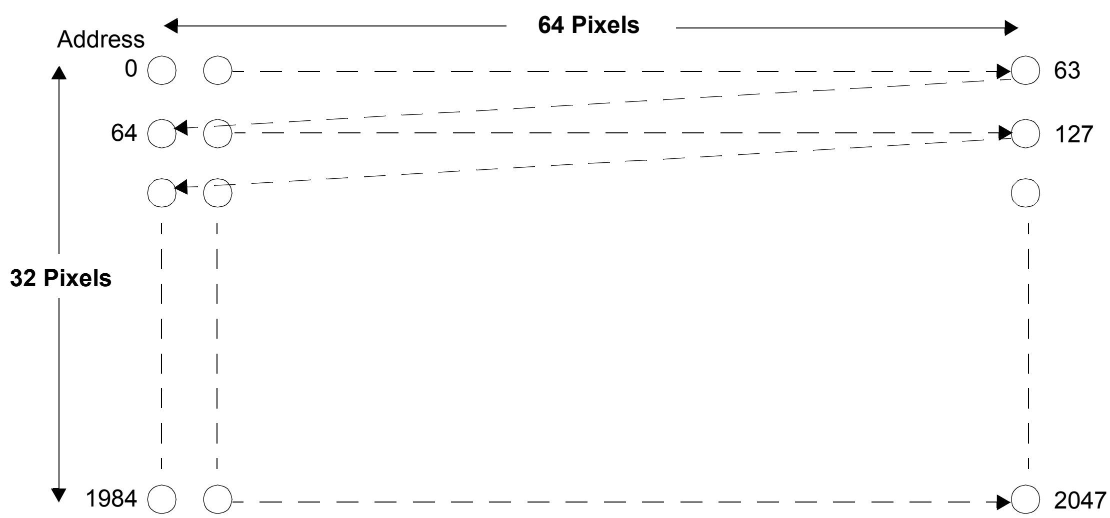


Figure 13-10: Example of “Line-by-Line” Storage of a 64x32 Frame

13.3.2 “Tiled Frame” Image Storage

In this method of storing images, a frame is divided into 8x8 pixel blocks with the top left block residing at address offset 0, followed by the block to the right of it. The right-most block of the first row of blocks is followed by the left-most block of the second row of blocks, and so on and so on. It is “row-by-row” storage of 8x8 pixel blocks with “line-by-line” storage of pixels within a block.

Note
For tiled frame image storage the frame width and height must be multiples of eight.

The following Figure shows an example of how a 64x32 frame with **8 bits per pixel** is stored in memory using the tiled frame method.

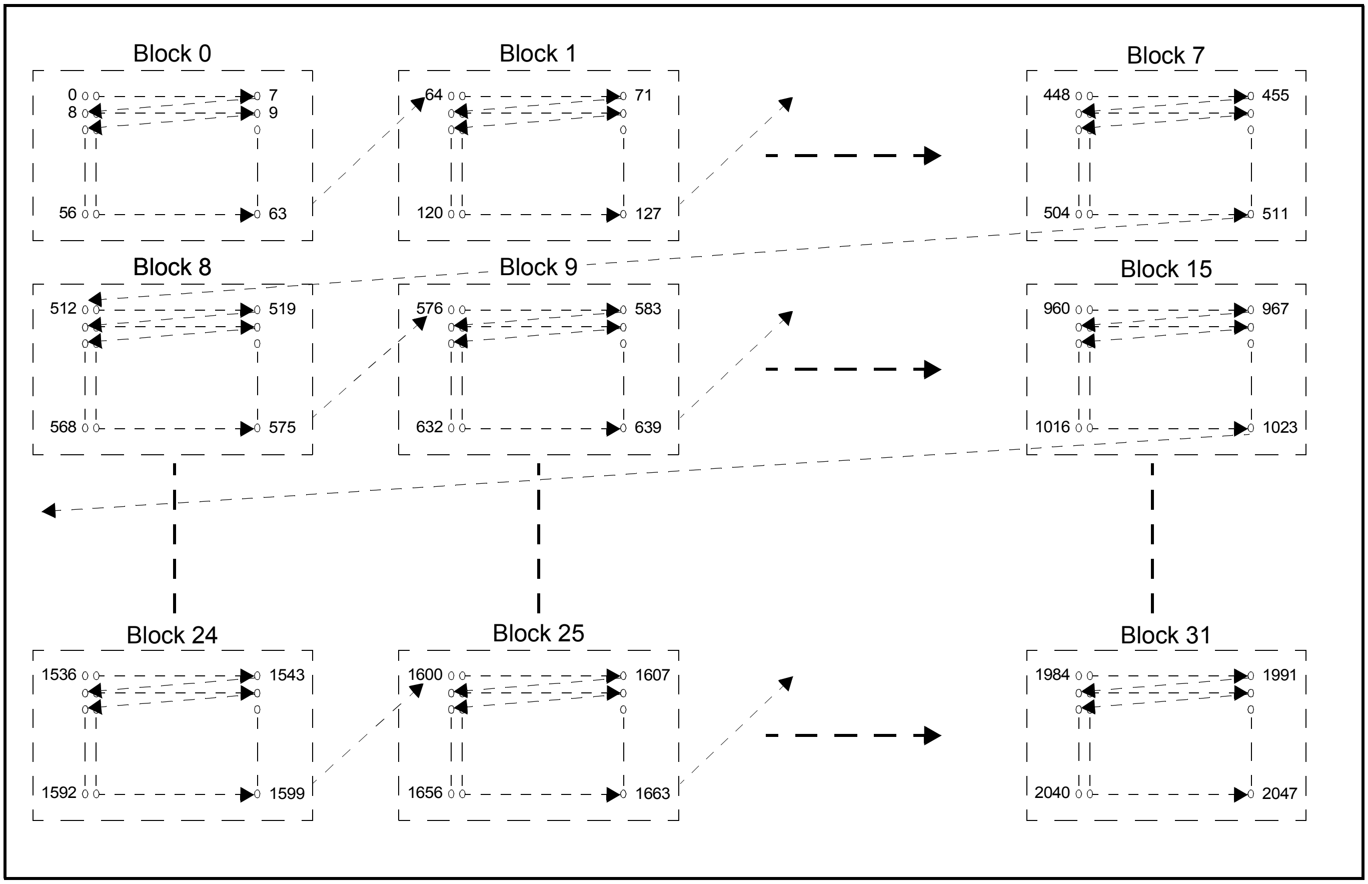


Figure 13-11: Example of “Tiled Frame” Storage of a 64x32 Frame

The tiled frame memory storage is advantageous for OpenGL-ES / OpenVG image rendering because they work on 8x8 pixel blocks. Organizing the image in tiled frame format will allow for efficient bursting of 8x8 pixel blocks in and out of SDRAM.

13.4 Frame Double-Buffering Scheme

13.4.1 Overview

In the S1D13515/S2D13515, there are programmable paths for the flow of image streams (frames) in the system. All frames are stored / buffered in the external SDRAM and the proper sequencing of the writing / reading of the frames to / from memory is required. A frame double-buffering scheme, described in this section, is implemented in the S1D13515/S2D13515 for proper sequencing of the writing / reading of frames. In the system, frame Producers and frame Consumers are defined for the purpose of describing the frame double-buffering scheme.

A Producer writes sequences / streams of images into memory. It can be a hardware block or software / firmware which loads images into memory. There are 5 hardware blocks in the S1D13515/S2D13515 which are Producers: Camera1 Image Writer, Camera2 Image Writer, CH1OUT Writeback, Warp Writeback, and the Sprite Engine.

A Consumer is a hardware block that reads sequences / streams of images from memory for display or further processing. There are 5 Consumers in the S1D13515/S2D13515: MAIN Fetcher, AUX Fetcher, OSD Fetcher, Image Fetcher, Warp Engine.

The frame rate between a Producer and Consumer can differ and can be asynchronous. Therefore, a frame double-buffering scheme is needed to prevent the “tearing effect”. If there is only a single buffer between the Producer and Consumer, the “tearing effect” will occur when the Consumer is still not finished reading a frame but the Producer has started writing a new frame to the buffer.

Each Consumer hardware block in the system can be programmed to connect to only one of the five hardware Producer blocks, or the Producer can be software / firmware which writes images / frames to memory. For each Producer and Consumer pair, a “connection” is defined. Each “connection” has a set of two frame buffers defined: Buffer0 and Buffer1. The Producer can only write to a frame buffer which is free and the Consumer can only read from a frame buffer which has valid contents (a complete frame of data). A set of control signal connections is also defined between the Producer and Consumer as shown in the following diagram:

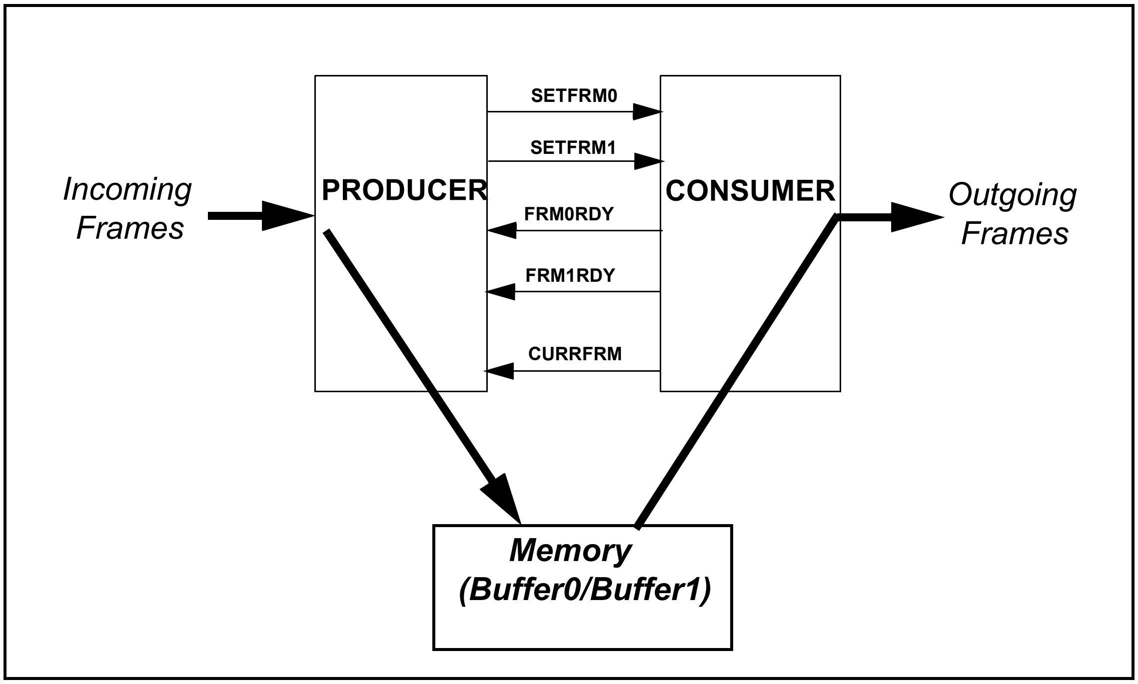


Figure 13-12: Control Signals for Frame Double-Buffering

FRM0RDY and FRM1RDY are status bits which reside in the Consumer. FRM0RDY indicates whether or not Buffer0 has a valid image ready and FRM1RDY indicates whether or not Buffer1 has a valid image ready for the Consumer to read. CURRFRM indicates which frame buffer the Consumer is currently reading / processing.

FRM0RDY / FRM1RDY can only be set by the Producer (SETFRM0 / SETFRM1 signals from the Producer) and can only be cleared by the Consumer.

Initially, FRM0RDY, FRM1RDY, and CURRFRM are assumed to be all equal to 0b. This means that both frame buffers are not ready and the Producer will first write to Buffer0.

The flowchart for the Producer and Consumer behavior are described in the next sections.

13.4.2 Frame Producer Flowchart

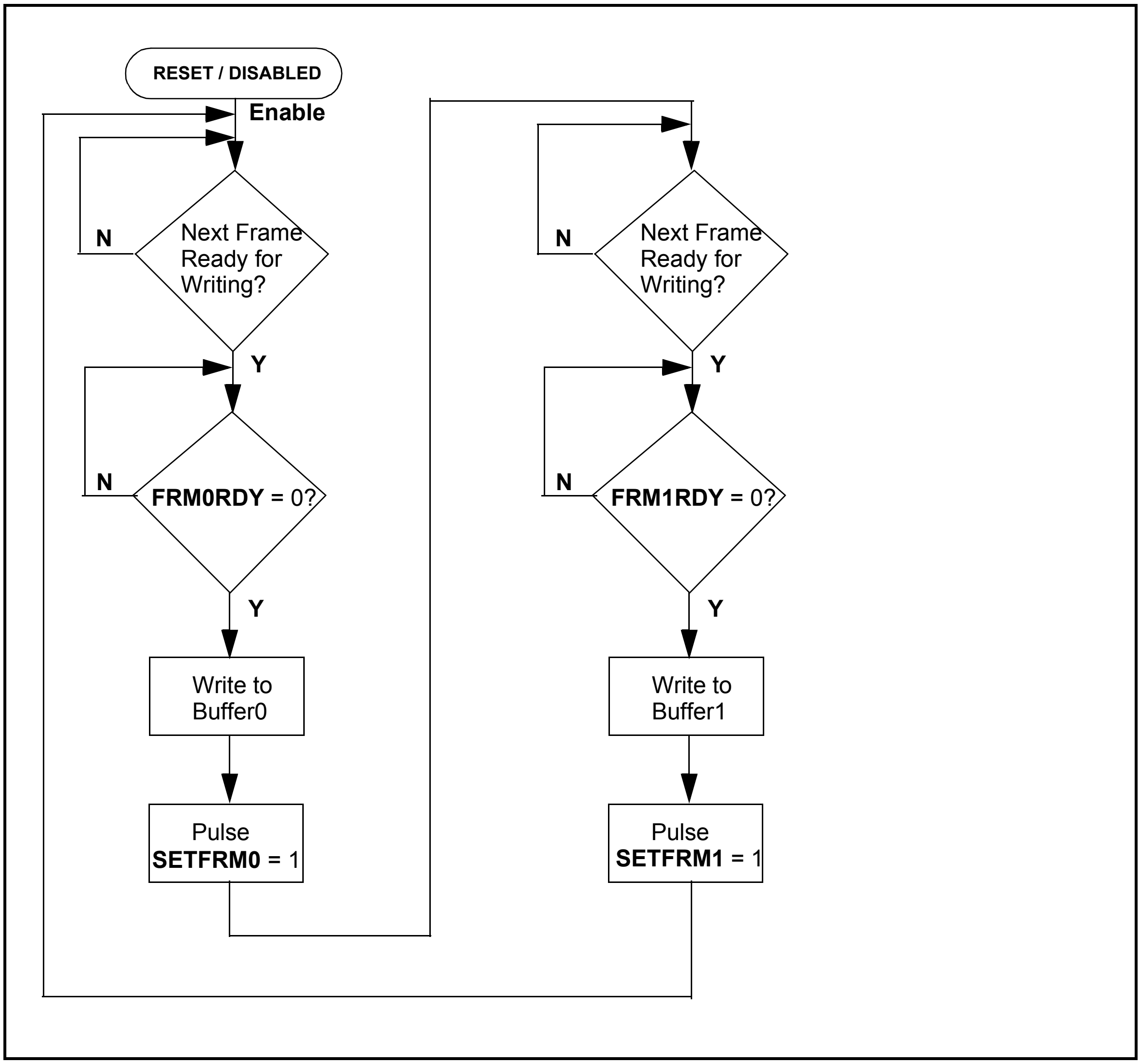


Figure 13-13: Flowchart for Producer of Frames

The primary guideline for the Producer of the frame double-buffering scheme is that it will only write to a frame buffer if the corresponding bit is 0 (indicating that the frame buffer is free). If both status bits are 1 and there is a new frame incoming into the Producer, the Producer will “absorb” the incoming frame (throw it away). If the Consumer has a slower frame (consumption) rate than the Producer, there will be periodic occasions when both status bits are 1 and the incoming frame into the Producer is discarded.

13.4.3 Frame Consumer Flowchart

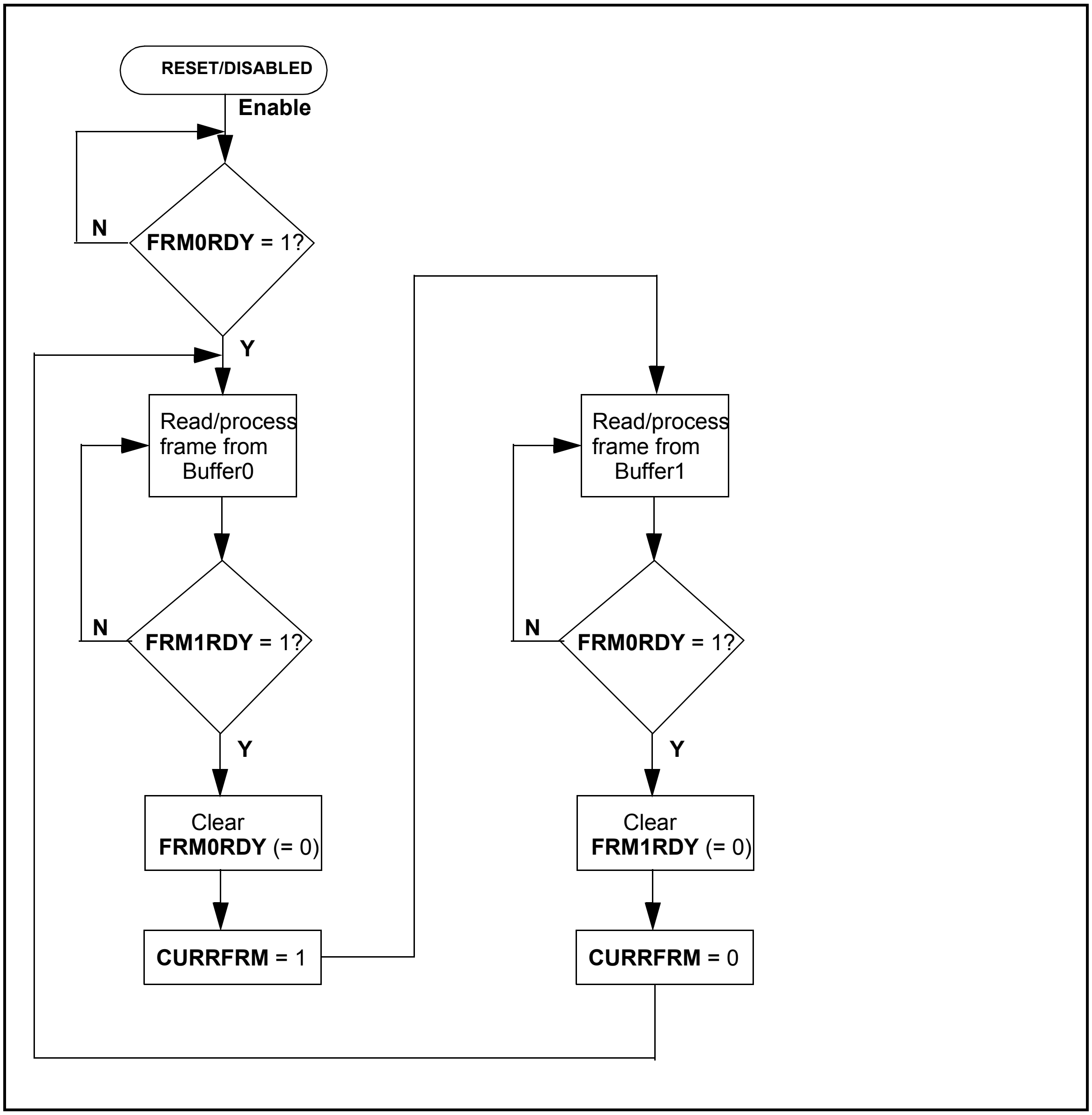


Figure 13-14: Flowchart for Consumer of Frames

The Consumer will only read / process from a frame buffer if the corresponding status bit is 1 (indicating that the frame buffer is ready). Initially, both status bits are 0 and CURRFRM is 0. The Consumer waits for FRM0RDY to go high and then processes Buffer0. After finishing with Buffer0, it checks the FRM1RDY bit to see if it is 1 (Buffer1 is ready). If FRM1RDY is 0 (Buffer1 is not ready), the Consumer repeats reading / processing from Buffer0 for the next outgoing frame. Reading / processing from Buffer0 will repeat until FRM1RDY becomes 1. When FRM1RDY becomes 1, the Consumer clears the FRM0RDY bit to 0, sets CURRFRM to 1, and starts reading / processing data from Buffer1 for the next outgoing frame.

While CURRFRM is 1, after the Consumer has finished reading / processing a frame from Buffer1, it checks the FRM0RDY to see if it is 1 (Buffer0 is ready). If FRM0RDY is 0 (Buffer0 is not ready), the Consumer repeats reading / processing from Buffer1 for the next outgoing frame. Reading / processing from Buffer1 will repeat until FRM0RDY becomes 1. When FRM0RDY becomes 1, the Consumer clears the FRM1RDY bit to 0, clears CURRFRM to 0, and starts reading / processing data from Buffer0 for the next outgoing frame. This process of toggling between the two buffers continues in this manner to achieve the “tear free” streaming of frames between the Producer and Consumer.

The primary guideline for the Consumer of the frame double-buffering scheme is that it repeats reading / processing from the same frame buffer if the other buffer is not ready.

13.4.4 Registers for Frame Double-Buffering Control

Hardware or Software / Firmware Frame Control

Each of the five Consumer hardware blocks has a HW / SW Frame Control bit to select whether the Producer is a hardware block or software / firmware. If the bit is 0, software / firmware sets the FRM0RDY and FRM1RDY bits and interacts with the Consumer hardware to implement the frame double-buffering scheme. If the bit is 1, a hardware block (Producer) sets the FRM0RDY and FRM1RDY bits. The HW / SW Frame Control bit for each of the five Consumers are accessed in the following registers:

- REG[09D8h] bit 0 = MAIN Fetcher
- REG[09D9h] bit 0 = AUX Fetcher
- REG[09DAh] bit 0 = OSD Fetcher
- REG[09DBh] bit 0 = Image Fetcher
- REG[0400h] bit 6 = Warp Engine

Frame Control / Status Register

Each of the five Consumers have a Frame Control / Status Register. Bit 2 of the register is the CURRFRM status (read-only). Bit 1 is the FRM1RDY bit and bit 0 is the FRM0RDY bit. The Frame Control / Status registers for the five Consumers are accessed in the following registers:

- REG[0942h] bit 2-0 = MAIN Fetcher (bit 2 is read-only, bits 1-0 are read / write)
- REG[0962h] bit 2-0 = AUX Fetcher (bit 2 is read-only, bits 1-0 are read / write)
- REG[0982h] bit 2-0 = OSD Fetcher (bit 2 is read-only, bits 1-0 are read / write)
- REG[09B2h] bit 2-0 = Image Fetcher (bit 2 is read-only, bits 1-0 are read / write)
- REG[0408h] bit 2-0 = Read-Only Status for Warp Engine
- REG[040Ah] bits 1-0 = Write-Only for setting FRM0RDY and FRM1RDY of Warp Engine

If software / firmware frame control is selected, the software / firmware writes a 1 to the FRM0RDY / FRM1RDY bit to set it to 1. The Consumer hardware is the one which clears the FRM0RDY / FRM1RDY bit and software / firmware writing a 0 to the FRM0RDY / FRMRDY1 bit has no effect.

Frame Control Signals Selection

If hardware frame control is selected for the Consumer, a Producer hardware block needs to be selected to “connect” to the Consumer. The following diagram shows a “connection” of the frame control signals between a Producer and a Consumer:

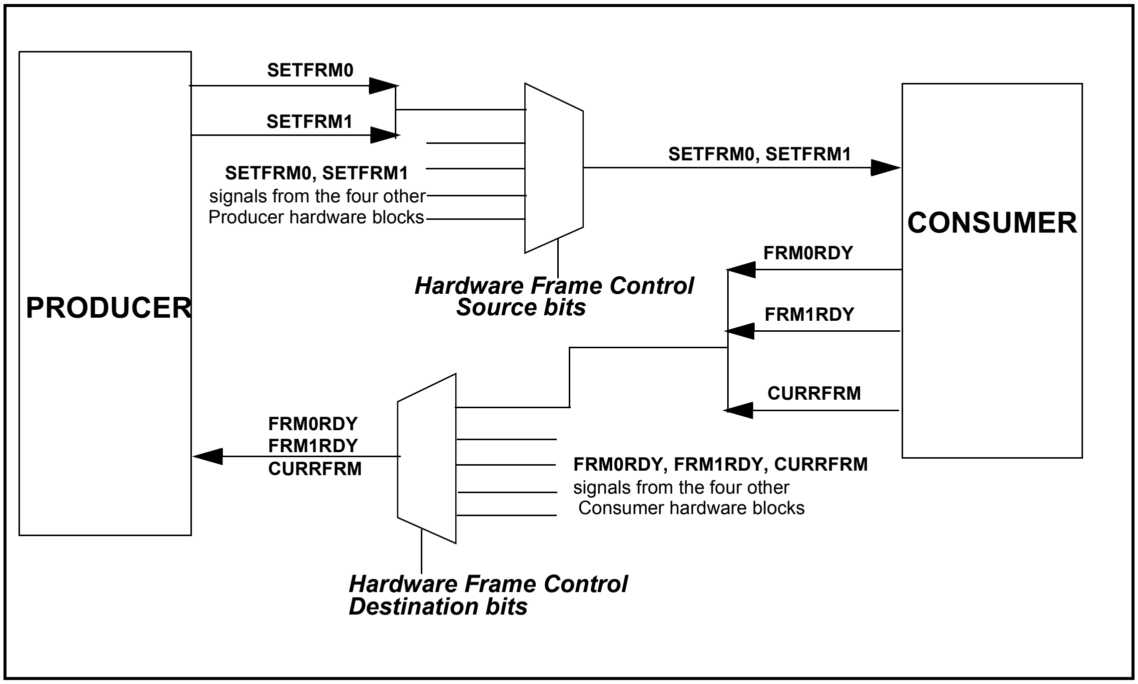


Figure 13-15: Frame Control Signals Selection

Each Producer has a set of SETFRM0 / SETFRM1 outputs and a set of FRM0RDY / FRM1RDY / CURRFRM inputs. Each Consumer has a set of SETFRM0 / SETFRM1 inputs and a set of FRM0RDY / FRM1RDY / CURRFRM outputs. The set of FRM0RDY / FRM1RDY / CURRFRM inputs to each Producer is selectable to come from one of the five Consumers (Frame Control Source bits), and the set of SETFRM0 / SETFRM1 inputs to the each Consumer is selectable to come from one of the five Producers (Frame Control Destination bit).

The following are registers for selecting the SETFRM0 / SETFRM1 inputs to each Consumer:

- REG[09D8h] bits 6-4 = MAIN Fetcher Hardware Frame Control Source
- REG[09D9h] bits 6-4 = AUX Fetcher Hardware Frame Control Source
- REG[09DAh] bits 6-4 = OSD Fetcher Hardware Frame Control Source
- REG[09DBh] bits 6-4 = Image Fetcher Hardware Frame Control Source
- REG[09DCh] bits 6-4 = Warp Engine Hardware Frame Control Source

The following are registers for selecting the FRM0RDY / FRM1RDY / CURRFRM inputs to each Producer:

- REG[09DEh] bits 3-0 = Camera1 Writer Hardware Frame Control Destination
- REG[09DEh] bits 7-4 = Camera2 Writer Hardware Frame Control Destination

- REG[09DFh] bits 3-0 = CH1OUT Writeback Hardware Frame Control Destination
- REG[09DFh] bits 7-4 = Warp Writeback Hardware Frame Control Destination
- REG[09DDh] bits 3-0 = Sprite Engine Hardware Frame Control Destination

For example, if we want to make a “connection” between the Camera2 Writer (Producer) and the AUX Fetcher (Consumer), REG[09D9h] bits 6-4 should be programmed to 001b (to select the SETFRM0 / SETFRM1 signals for the AUX Fetcher to come from the Camera2 Writer) and REG[09DEh] bits 7-4 should be programmed to 0010b (to select the FRM0RDY / FRM1RDY / CURRFRM signals for the Camera2 Writer to come from the AUX Fetcher).

Disabling Frame Double-Buffering

Each Producer hardware can be programmed to disable double-buffering through its Frame Double-Buffer Disable bit. If frame double-buffer is disabled and the Producer only writes to Buffer0 all the time. The following are the register bits to enable / disable frame double-buffering for each of the five Producers:

- REG[09DCh] bit 0 = Warp Writeback
- REG[09DCh] bit 1 = CH1OUT Writeback
- REG[09DCh] bit 2= Camera1 Writer
- REG[09DCh] bit 3= Camera2 Writer
- REG[5000h] bit 1 = Sprite Engine

13.5 Gamma LUT

The S1D13515/S2D13515 includes a Look-up Table architecture that can be used for Gamma Correction of LCD2. When Gamma Correction is enabled, the color correction affects the entire display including all active windows on LCD2.

Note
The LUT can also be used for Optical Crosstalk Correction when an EID Double Screen or Sharp DualView panel is used. For detailed information on Optical Crosstalk Correction, please contact your EID representative.

The LCD2 Gamma LUT is arranged as Bank A and Bank B. This allows one bank to be programmed while the other bank is used for gamma correction. Each bank consists of an independent table for each 8-bit RGB color component. There are 512 indexes in each color component table divided into two segments. Segment 1 includes indexes 0-255 and Segment 2 includes indexes 256-511.

For details on programming and using the LCD2 Gamma LUT, refer to the register descriptions "REG[40A0h] LCD2 Gamma LUT Data Port" ~ "REG[40A3h] LCD2 Gamma LUT Configuration Register 1" starting on page 416.

The following figure shows the architecture of the Gamma LUT.

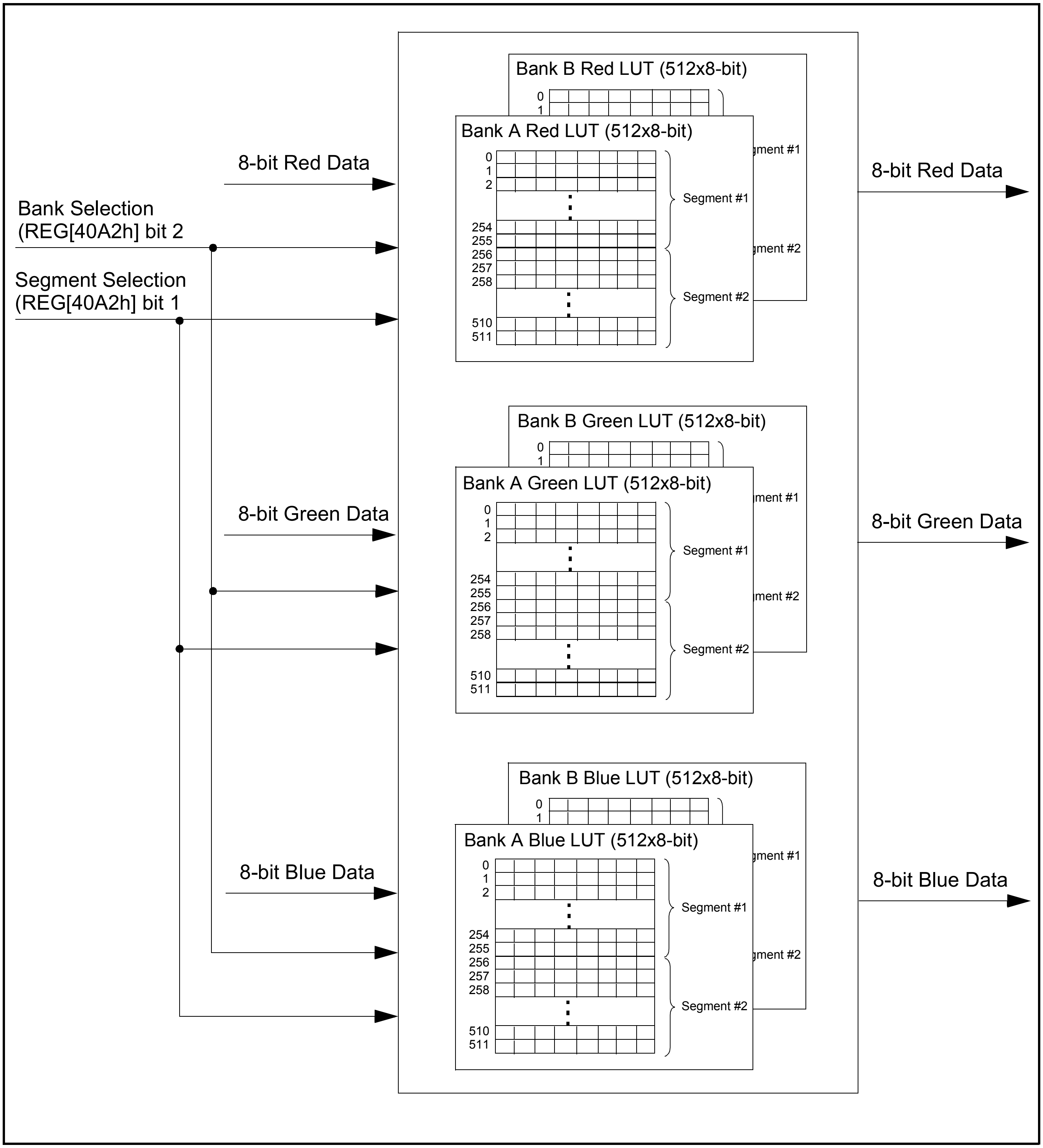


Figure 13-16: LUT Architecture

Chapter 14 I2S Audio Output Interface

14.1 Overview of Operation

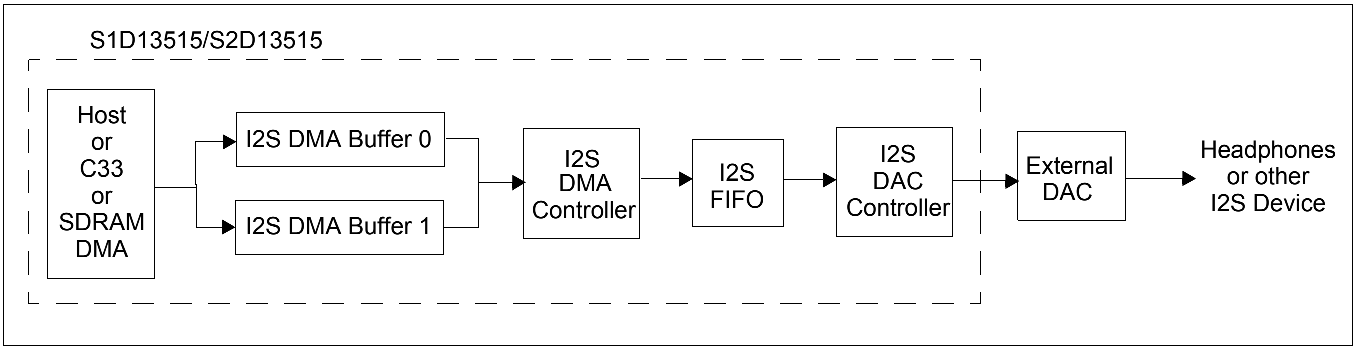


Figure 14-1: I2S Interface Overview Diagram

The I2S Audio Output logic consists of a 16-byte FIFO which feeds 16-bit PCM audio data for the I2S synchronous serial output stream. The I2S DMA Controller block reads PCM audio data from memory and writes to the FIFO. The FIFO has a programmable threshold level (REG[104h] bits 5-2) which is used to request the I2S DMA Controller to write more data. If the number of bytes in the FIFO is less than or equal to the threshold level, a request to the I2S DMA Controller is triggered to write the next 32-bit (4 bytes) value to the FIFO.

There are two buffers in memory defined for the I2S DMA Controller: I2S DMA Buffer 0 and I2S DMA Buffer 1. The location in memory of the buffers are programmable through REG[0148h] ~ REG[014Bh] and REG[014Ch] ~ REG[014Fh]. The size of the two buffers are also programmable through REG[0152h] ~ REG[0153h]. When the I2S Interface is disabled (REG[0104h] bit 0 is 0b), the I2S DMA Controller's internal address point resets to Buffer0's start address. Before enabling the I2S Interface, the audio data generator should fill Buffer0 with the first block of audio data.

When the I2S Interface is enabled and a request is made by the FIFO, the I2S DMA Controller reads a 32-bit value from the first address in Buffer0, writes to the FIFO, and increments its internal address pointer by 4.

When the I2S DMA Controller finishes reading the last data of Buffer0 (whose size is defined by REG[0152h] ~ REG[0153h]), the I2S DMA Interrupt Status bit (REG[0154h] bit 3) is set to 1b and the I2S DMA Controller switches to using Buffer1. The audio data generator can then start writing new data to Buffer0 while the I2S DMA Controller is reading from Buffer1.

After the I2S DMA Controller is finished with Buffer1, it sets the I2S DMA Interrupt Status bit to 1 and switches back to reading from Buffer0. As long as the I2S Interface is enabled, the I2S DMA Controller toggles between the two buffers. The I2S DMA Buffer Selection Status bit (REG[0154h] bit 1) indicates which buffer the I2S DMA Controller is currently reading data from.

Note
It is strongly recommended for performance reasons to locate the I2S DMA buffers in the internal RAM area 0400_0000h ~ 0401_7FFFh, unless a C33 operation is needed which uses this internal RAM area, for proper operation.

14.2 Audio Data Formats and Organization in Memory

The I2S Interface is programmable to be either Mono or Stereo through REG[0104h] bit 7. The WS signal is high for 16 SCK cycles and low for 16 SCK cycles. In Mono mode, a single 16-bit PCM value is read from the FIFO for each WS period. The 16-bit mono value is shifted out when WS is high and the same value repeated for WS low.

In Stereo mode, two 16-bit PCM values from read from the FIFO for each WS period. When the I2S Interface is first enabled, the first 16-bit data shifted out is always defined as the Left channel data and the second 16-bit data shifted out is defined as the Right channel data. The rest of the 16-bit data that follow toggles between Left and Right channel data. In Mono mode, one 16-bit data is read from the FIFO and shifted out twice (i.e. repeat the same data for Left and Right channels). For Stereo mode, one 16-bit data is read from the FIFO for each 16-bit shifted out of the I2S Interface.

For Mono mode, the 16-bit PCM audio data stored in memory are single mono audio data samples starting from the base address of the buffer.

For Stereo mode, the first 16-bit data stored at the base address of the buffer is always the Left channel data followed by the Right channel data at offset 2.

14.3 WS Polarity

REG[0100h] bit 5 defines the polarity of the WS signal with respect to Left and Right channel data. If REG[0100h] bit 5 is 0b, WS=1 for Left channel data and WS=0 for Right channel data. If REG[0100h] bit 5 is 1b, WS=0 for Left channel data and WS=1 for Right channel data.

14.4 Channel Data Blanking

REG[0100h] bit 6 can be used to “blank” (16-bit data shifted out is 0) the Right Channel data. If REG[0100h] bit 6 is 0b, Right Channel data is not “blanked”. If REG[0100h] bit 6 is 1b, Right Channel data is “blanked”. For Stereo mode, the value read from the FIFO for the Right Channel data is “absorbed” (lost) and not shifted out. The audio generator needs to generate “dummy” data in memory/buffer for the Right channel.

REG[0100h] bit 7 can be used to “blank” (16-bit data shifted out is 0) the Left Channel data. If REG[0100h] bit 7 is 0b, Left Channel data is not “blanked”. If REG[0100h] bit 7 is 1b, Left Channel data is “blanked”. For Stereo mode, the value read from the FIFO for the Left Channel data is “absorbed” (lost) and not shifted out. The audio generator needs to generate “dummy” data in memory/buffer for the Left channel.

14.5 WS Timing in Relation to SDO

The timing between the rising/falling edge of WS in relation to the 16-bit data shifted out on SDO/SCK is selectable by REG[0100h] bit 3. If REG[0100h] bit 3 is 0b, the first bit of the 16-bit PCM data is shifted out on SDO one SCK clock cycle after the rising/falling edge of WS. If REG[0100h] bit 3 is 1b, the first bit of the 16-bit PCM data is shifted out on the same edge as the WS rising/falling edge.

14.6 PCM Data Bit Order

REG[0100h] bit 2 determines the order for bits shifted out on SDO. If REG[0100h] bit 2 is 0b, the most significant bit of the 16-bit PCM data is shifted out first. If REG[0100h] bit 2 is 1b, the bit order is reversed and the least significant bit of the 16-bit PCM data is shifted out first.

14.7 WS/SCK Signal Direction

The WS and SCK signals for the I2S Interface can either be generated by the S1D13515/S2D13515 or by an external source. REG[0100h] bit 0 and REG[0101h] bit 0 are used to select the clocking source for the I2S Interface. To select the S1D13515/S2D13515 as the clocking source for the I2S Interface's WS and SCK signals, REG[0100h] bit 0 should be set to 0b and REG[0101h] bit 0 should be set to 1b. To select an external source for WS and SCK signals, REG[0100h] bit 0 should be set to 1b and REG[0101h] bit 0 should be set to 0b.

14.8 Interrupts

14.8.1 I2S FIFO Interrupts

REG[010Ch] contains 3 interrupt status bits which can be used to indicate three types of error condition for the I2S FIFO and REG[0105h] contains interrupt enable bits for corresponding interrupt status bits in REG[010Ch]. The three interrupt sources for the I2S FIFO are OR'ed together to produce the read-only I2S DAC Interrupt status bit in the Interrupt Status Register 0 (REG[0A00h] bit 6). The enable/disable for the I2S DAC Interrupt to the Host is programmed in REG[0A06h] bit 6. The enable/disable for the I2S DAC Interrupt to the C33 is programmed in REG[0A0Eh] bit 6.

14.8.2 I2S DMA Interrupt

The I2S DMA Interrupt status bit (REG[0154h] bit 3) is the source for the IRQ3 interrupt of the Interrupt Controller for the C33. To enable the IRQ3 interrupt, program bit 3 of REG[0A42h] to 1b.

14.9 I2S Typical Operation Flow

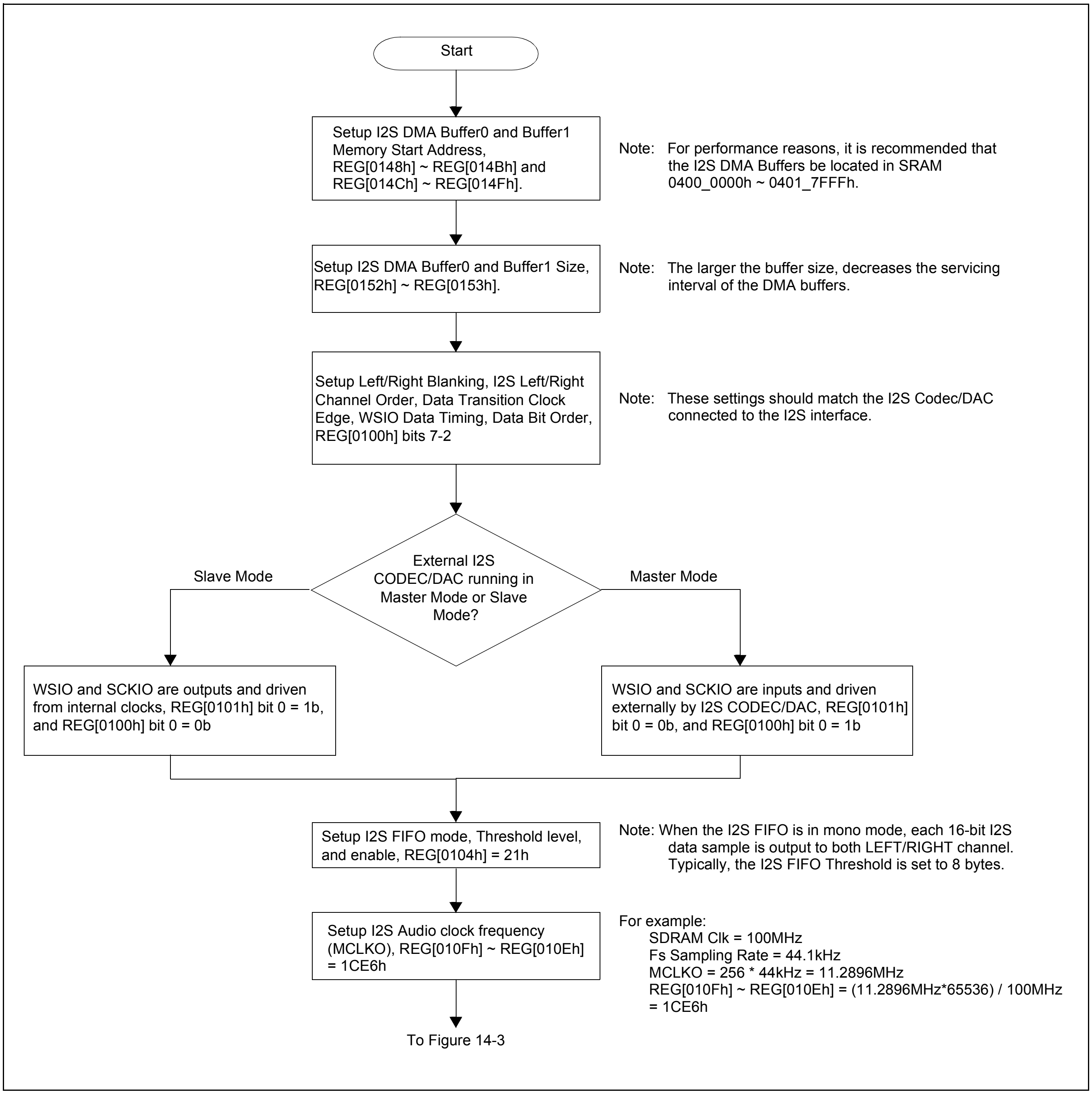


Figure 14-2: I2S Typical Operation Flow

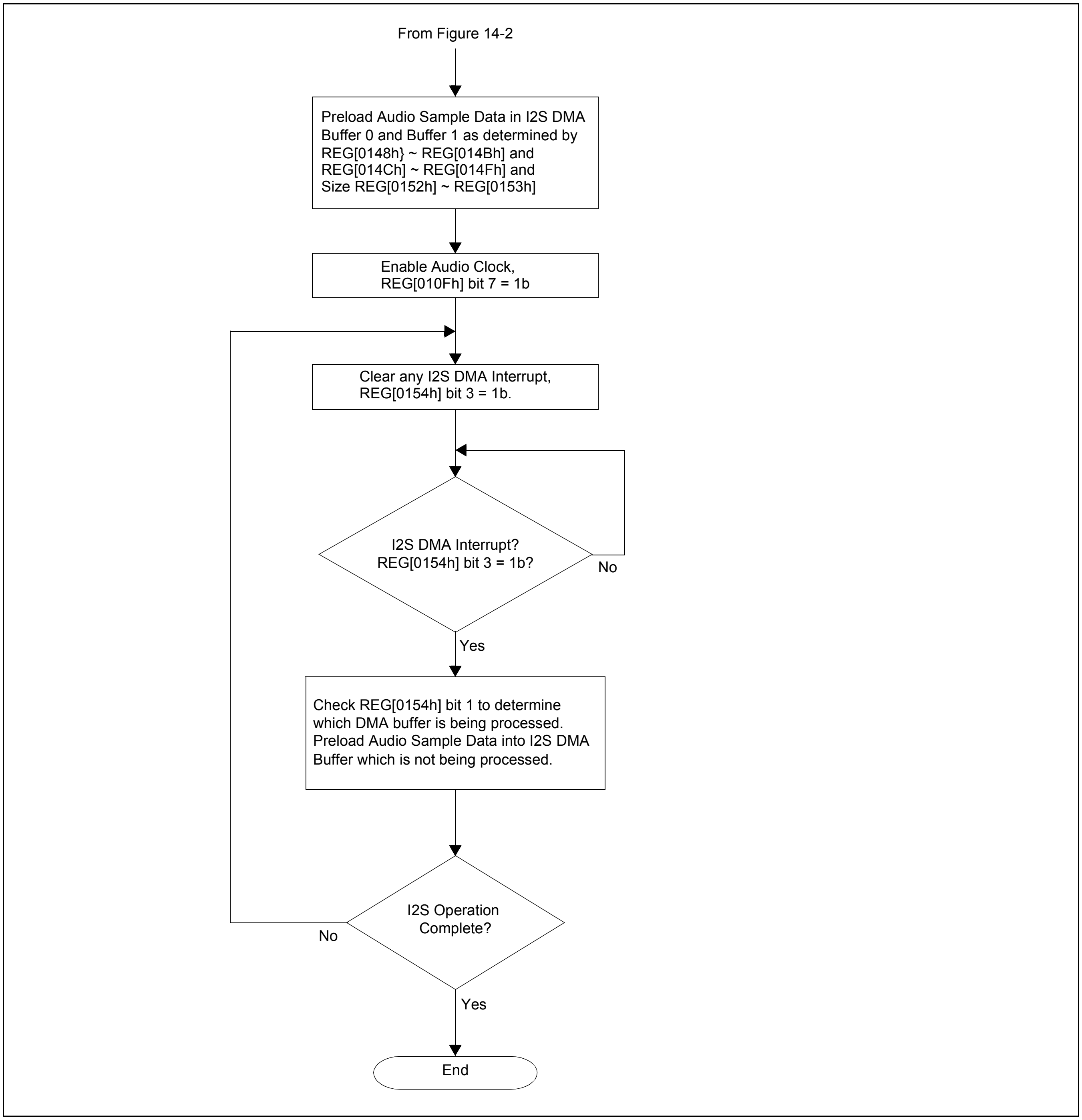


Figure 14-3: I2S Typical Operation Flow (Continued)

Chapter 15 2D BitBLT

The S1D13515/S2D13515 has no specific hardware BitBLT functions. However, some BitBLT functionality is provided through the API using the on-chip C33PE RISC processor. The available BitBLT functions come in 2 forms.

1. ROM Monitor BLT Functions (built-in)
2. Loadable BLT Functions (optional)

15.1 ROM Monitor BitBLT Functions

The ROM Monitor implements basic BLT functions in firmware. Basic BLT functions allow fast block transfers in memory using a solid FILL color as the source data. One of 16 raster operations can be applied to source and destination data providing various blending effects.

To use this function, the Host must setup the BLT command parameters and then issue an interrupt to the C33PE which triggers command execution. The following raster operations are supported for basic BLT functions.

Table 15-1: BitBLT Raster Operations

ROP Code	Operation		ROP Code	Operation
0	D = 0		8	D = S.D
1	D = _(S+D)		9	D = _(S^D)
2	D = _S.D		A	D = D
3	D = _S		B	D = _S+D
4	D = S._D		C	D = S
5	D = _D		D	D = S+_D
6	D = S^D		E	D = S+D
7	D = _(S.D)		F	D = 1

For further information on using BitBLT Raster Operations, refer to the *S1D13515/S2D13515 API documentation*.

15.2 Loadable BitBLT Functions

The S1D13515/S2D13515 supports optional 2D graphics functions that can be loaded from the flash memory or directly from the Host. These functions are available as two optional libraries: a small library and a larger library.

15.2.1 Small Library

The intent of the small library is to provide basic BLT functions that are commonly used for window support by most graphics processors. This library will consist of functions such as:

- MoveBLT
- ColorExpand
- StretchBLT

Note
Write and Read BLT functions are not provided as there is a limited performance gain over direct writes and reads from the Host.

15.2.2 Large Library

The larger library includes all functions of the small library and is based on the LIBART open source graphics library. It supports a much larger set of functions that include (but are not limited to):

- LineDraw
- DrawCircle/Arc
- DrawRetangle

15.2.3 Other Libraries

In addition to the graphics libraries developed by Epson, several 3rd party graphics library vendors develop for Epson display controllers. For a complete list of these vendors and their products, please visit the Epson website at vdc.epson.com.

Note
The individual library functions are documented in separate API library documents and are not considered part of the S1D13515/S2D13515 Hardware Specification.

Chapter 16 Sprite Engine

The S1D13515/S2D13515 is designed with a Sprite Engine to enhance the performance of applications requiring independent object based graphics. The Sprite Engine allows these objects to be defined as “sprites” which can be easily moved over another image without modifying the background image.

The Sprite Engine features the following.

- Support for up to 8 individual sprites to be simultaneously displayed. Sprite #0 is defined as the background sprite image.
- Programmable Sprite Size Register - each sprite can vary in size and is only limited by the amount of available memory in SDRAM).
- Individual Sprite X,Y location register (location can be negative on all edges of display to allow the sprite to gradually go off one side of the display).
- Individual Sprite Z-Order (each sprite has an associated z-order which determines which sprite is visible over another sprite when their locations overlap).
- Alpha blending support for all ARGB format sprites
- Sprite image data can be RGB 5:6:5, ARGB 1:5:5:5 or ARGB 4:4:4:4 data formats.
- Sprite Rotation / Mirror functions
  - sprite rotation is independent of the main display orientation
  - programmable rotation reference point (by X/Y offsets from the upper left corner of the sprite, both can be positive or negative)
- Sprite images are stored in SDRAM.
- Any combination of rotation and mirroring can be generated from only 0 degree and 90 degree versions of stored sprite images.

16.1 Sprite Data Path

All individual sprites are stored in SDRAM. When required, sprite data is read from the SDRAM and synthesized to SDRAM for display on the panel. Optionally, double buffering can be enabled to reduce tearing and allow faster frame rates.

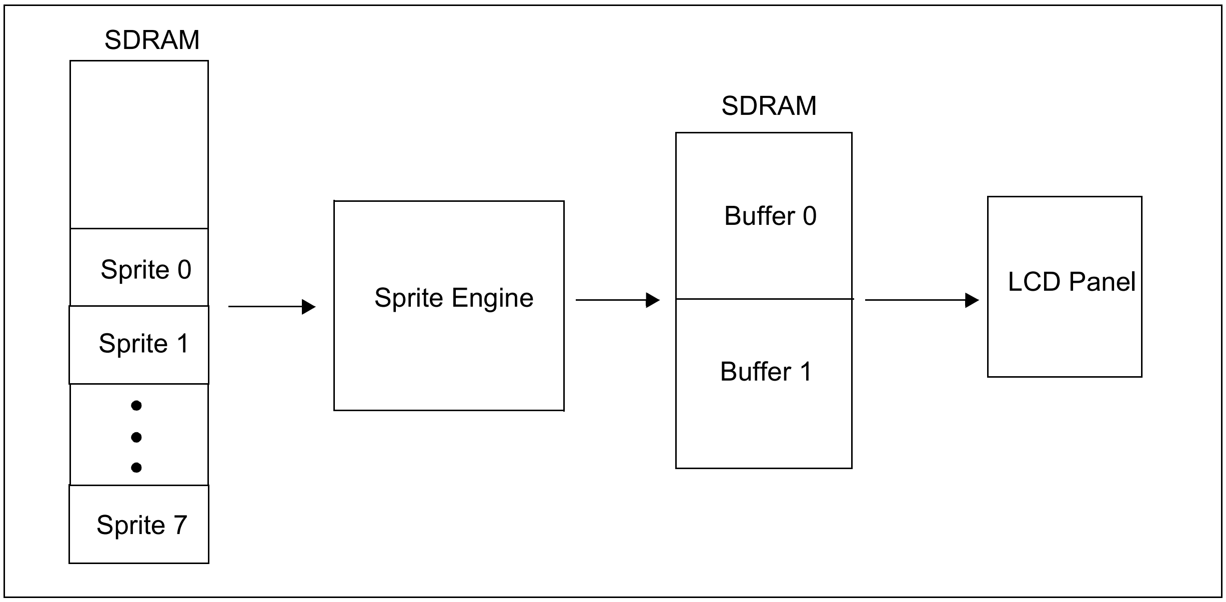


Figure 16-1: Sprite Data Path

16.2 8 Sprite Support with Z-ordering Transparency

Each sprite has an associated z-order which is used to determine which part of the sprite is displayed when the sprite overlaps the main image or other sprites.

Note
When configuring the Z-order and transparency settings, Sprite #0 must always be set to the lowest Zorder and must have transparency disabled.

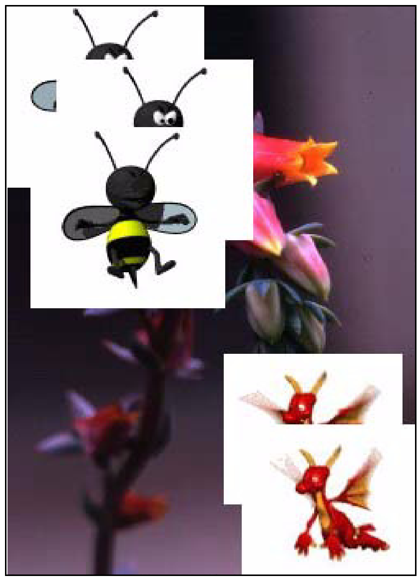


Figure 16-2: Z-order Example

When RGB 5:6:5 format is selected, one programmable transparency color may be associated with it. Transparency allows an irregular shaped image to be displayed over the background.

Figure 16-3: Z-order with Transparency Example

16.3 8 Sprite Support with Z-ordering Alpha-Blending

The Sprite Engine supports Alpha-Blending which provides further visual enhancement for games and similar applications. Alpha-blending is used in computer graphics to create the effect of transparency. This technique is useful for graphics that feature glass or liquid objects and is done by combining a translucent foreground with a background color to create a blend. It can also be used for animation, where one image gradually fades into another image.

Note

When configuring the Z-order and alpha blending settings, Sprite #0 must always be set to the lowest Z-order and must not have an alpha value of 0 (transparent).

The Sprite Engine supports alpha-blending for 2 alpha formats.

- ARGB 1:5:5:5 - one Alpha bit points to 2 programmable indexed 4-bit alpha values
- ARGB 4:4:4:4 - the four bits represent the actual alpha value

The following equation describes the alpha blending technique used.

$$[r, g, b]_{blended} = \alpha[r, g, b]_{foreground} + (1 - \alpha)[r, g, b]_{background}$$

Where:

[r,g,b] are the red, green, and blue color channels
α is the weighting factor

The weighting factor value can be from 0 to 1 (represented as 0 to 15 for Sprite Engine). When set to 0, the foreground is completely transparent. When it is set to 1, the background is completely transparent. All values between specify a mixture of the foreground and the background.

Figure 16-4: Alpha Blending with Alpha Value of 0, 0.5 and 1

The Sprite Engine allows up to 8 sprites to be alpha blended together. Z-ordering determines which sprites are displayed in the foreground and background for each alpha-blending operation.

Figure 16-5: Z-order with Alpha-Blending

16.4 Reference Point Based 90°, 180° and 270° Rotation + Mirror

Each sprite can be independently rotated (90°, 180°, 270°) and/or mirrored. The resulting orientation of the sprite is independent of the main display orientation.

Each sprite has a programmable rotation reference point. Unlike other designs where the rotation is always based on the center of the image, this design allows the user to program any point on the display as the rotation axis. This reference point can even be outside of the sprite area.

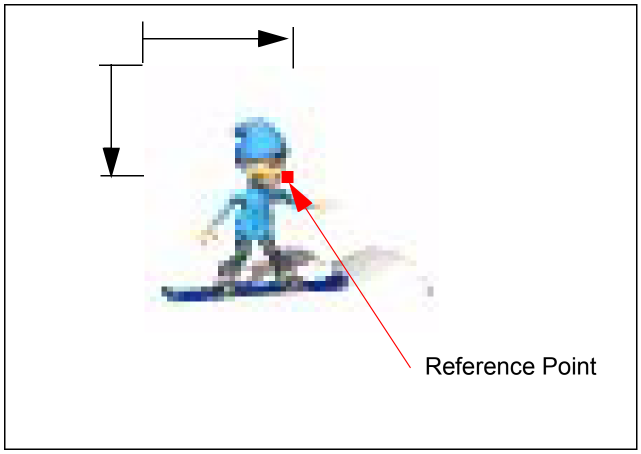


Figure 16-6: Sprite Reference Point

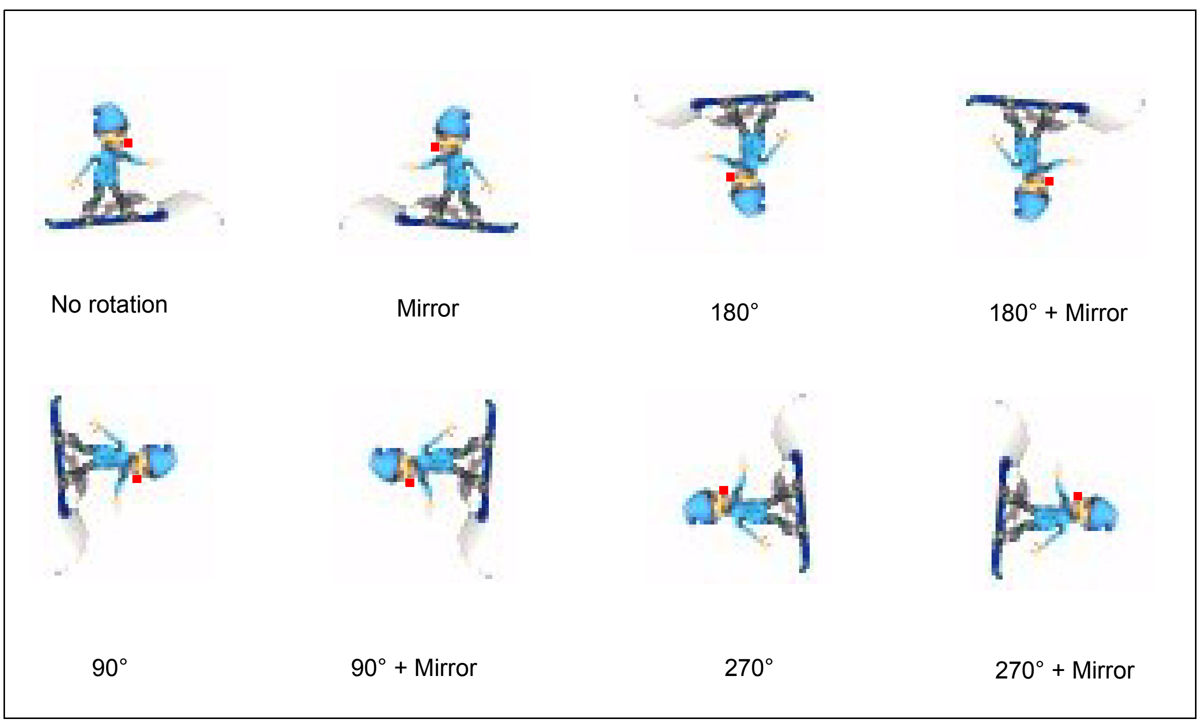


Figure 16-7: Sprite Rotation and Mirror Examples

16.5 Sprite Display Orientation and Positioning

The sprite frame rendered to the SDRAM frame buffer is determined by the dimensions of Sprite #0. Therefore, Sprite #0 defines the resulting SDRAM memory size, and sprite #1 - #7 position is rendered with reference to the rectangle defined by the Sprite #0 frame width and height parameters.

Note
Rotation is not supported for Sprite #0.

The Main / AUX / OSD window dimensions and memory start address should match the sprite #0 dimensions and the sprite frame buffer start address, in order to display the rendered sprite frame to the Main / AUX / OSD Window of the display.

The following figures demonstrate how to size and position a sprite for rendering to the frame buffer. Examples are shown for several combinations of rotation and mirroring. Sprite collision rectangle orientation and positioning is done in a similar manner. The figures assume the following values:

A = X offset of the reference point relative to the upper left corner of the sprite
B = Y offset of the reference point relative to the upper left corner of the sprite
C = X offset of the sprite position (reference point) relative to the upper left corner of the display
D = Y offset of the sprite position (reference point) relative to the upper left corner of the display
E = New effective X-Start of the sprite on the display after rotation/mirroring
F = New effective Y-Start of the sprite on the display after rotation/mirroring
G = Width of the sprite - A
H = Height of the sprite - B
I = New effective X-End of the sprite on the display after rotation/mirroring
J = New effective Y-End of the sprite on the display after rotation/mirroring

0° Rotation with Mirror Disabled

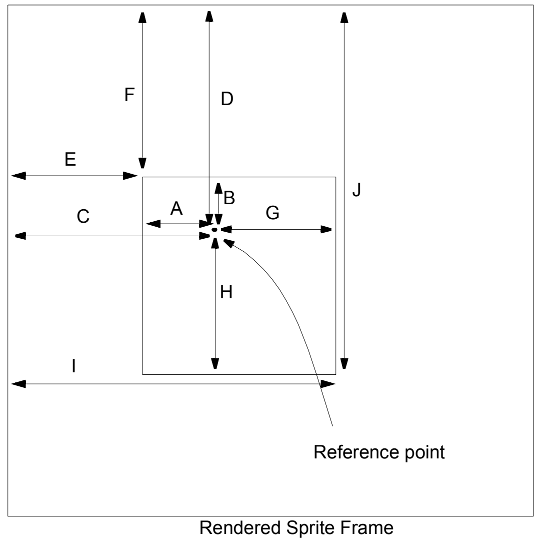


Figure 16-8: Sprite Display for Rotation 0° with Mirror Disabled

E = C - A F = D - B
I = C + G J = D + H

90° Rotation with Mirror Disabled

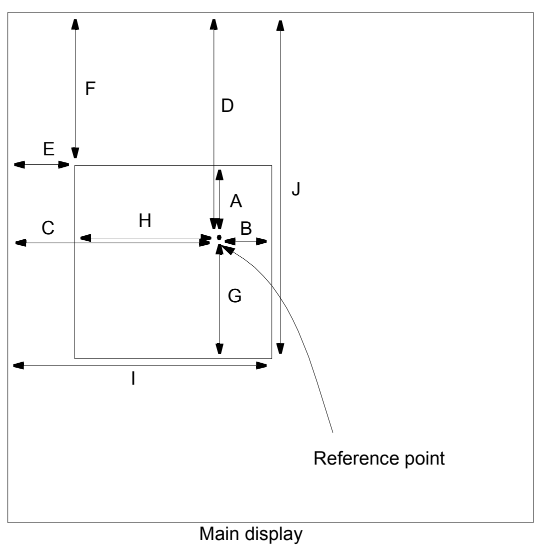


Figure 16-9: Sprite Display for Rotation 90° with Mirror Disabled

E = C - H F = D - A
I = C + B J = D + G

180° Rotation with Mirror Disabled

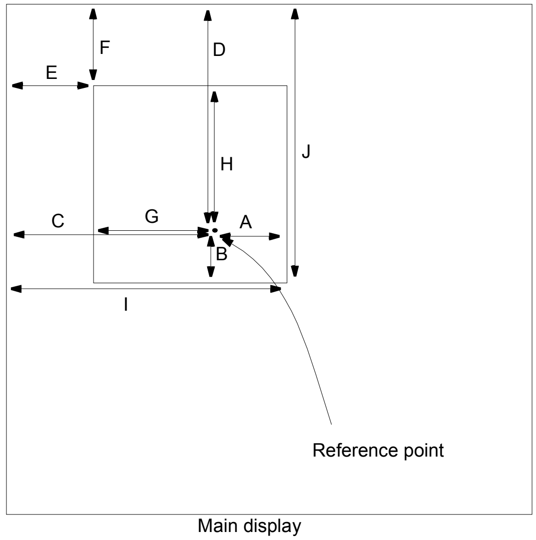


Figure 16-10: Sprite Display for Rotation 180° with Mirror Disabled

E = C - G	F = D - H
I = C + A	J = D + B

270° Rotation with Mirror Disabled

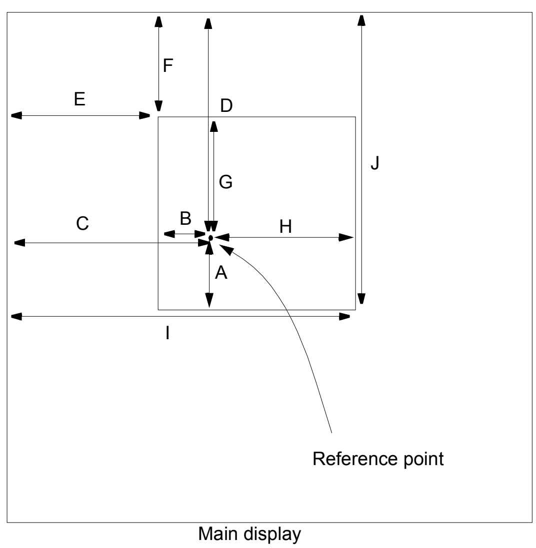


Figure 16-11: Sprite Display for Rotation 270° with Mirror Disabled

E = C - B	F = D - G
I = C + H	J = D + A

0° Rotation with Mirror Enabled (Left <-> Right)

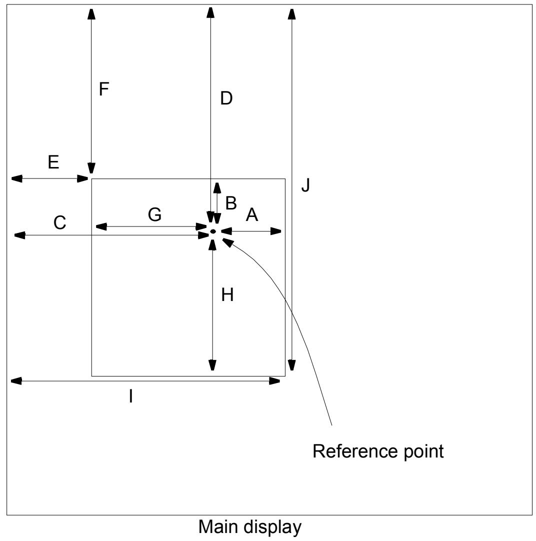


Figure 16-12: Sprite Display for Rotation 0° with Mirror Enabled

E = C - G	F = D - B
I = C + A	J = D + H

90° Rotation with Mirror Enabled

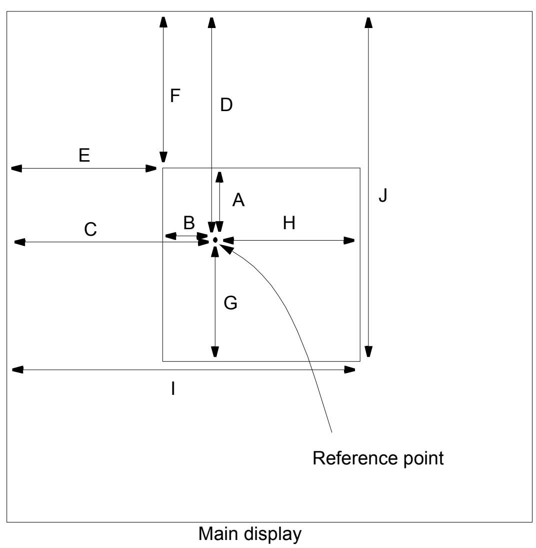


Figure 16-13: Sprite Display for Rotation 90° with Mirror Enabled

E = C - B	F = D - A
I = C + H	J = D + G

180° Rotation with Mirror Enabled

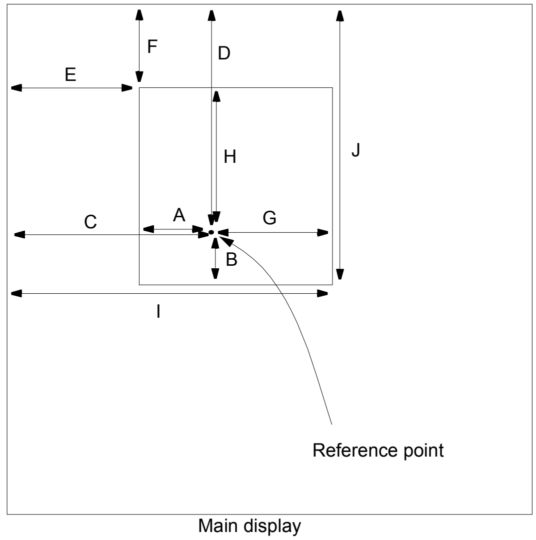


Figure 16-14: Sprite Display for Rotation 180° with Mirror Enabled

E = C - A	F = D - H
I = C + G	J = D + B

270° Rotation with Mirror Enabled

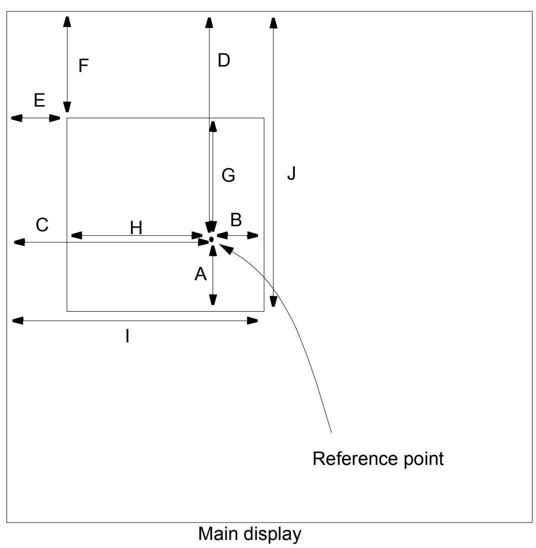


Figure 16-15: Sprite Display for Rotation 270° with Mirror Enabled

E = C - H	F = D - G
I = C + B	J = D + A

16.6 Sprite Programming Flow

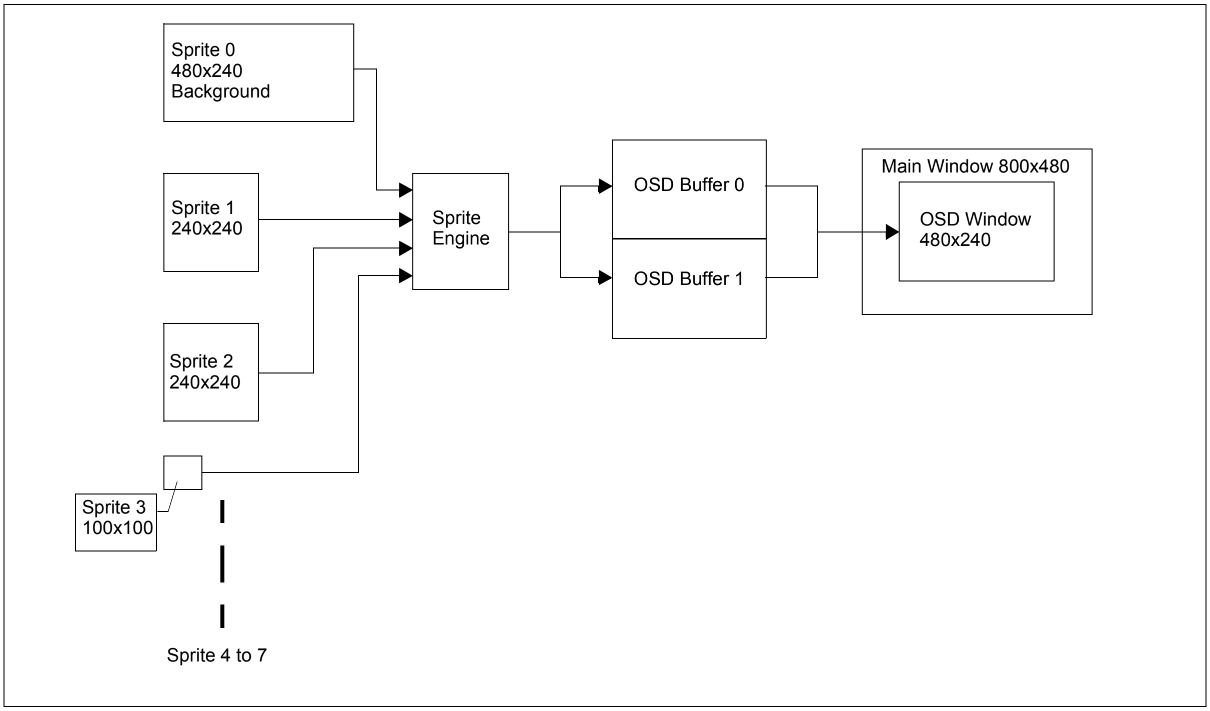


Figure 16-16: Typical Sprite Programming Block Diagram

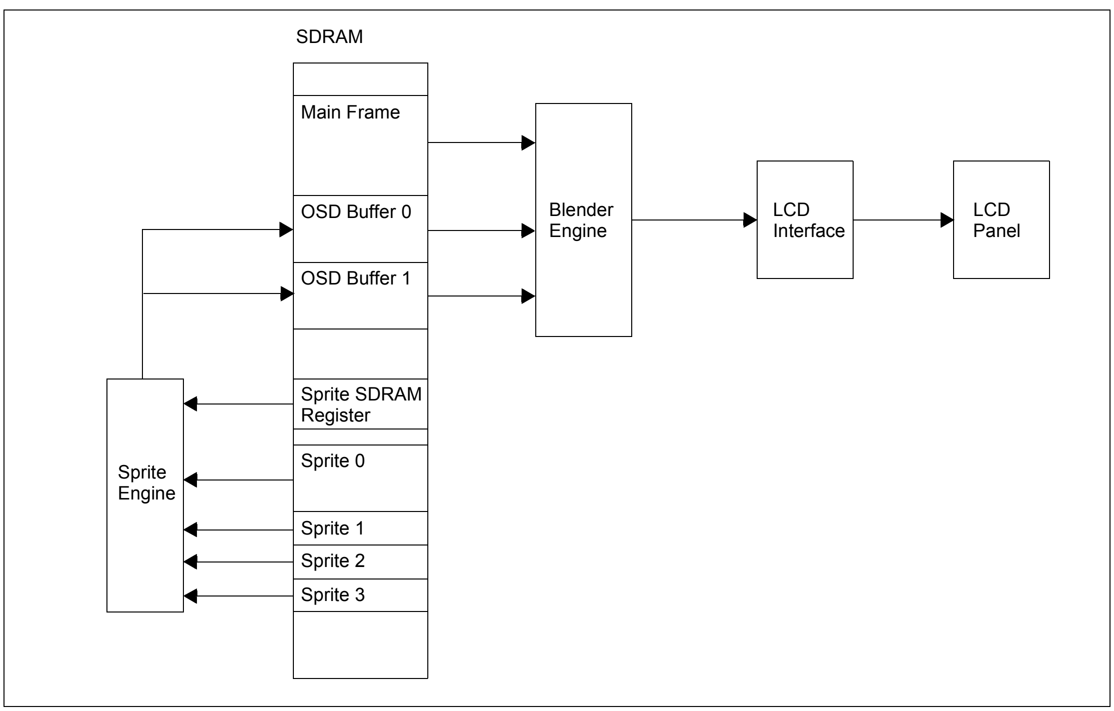


Figure 16-17: Typical Sprite Programming Memory Map

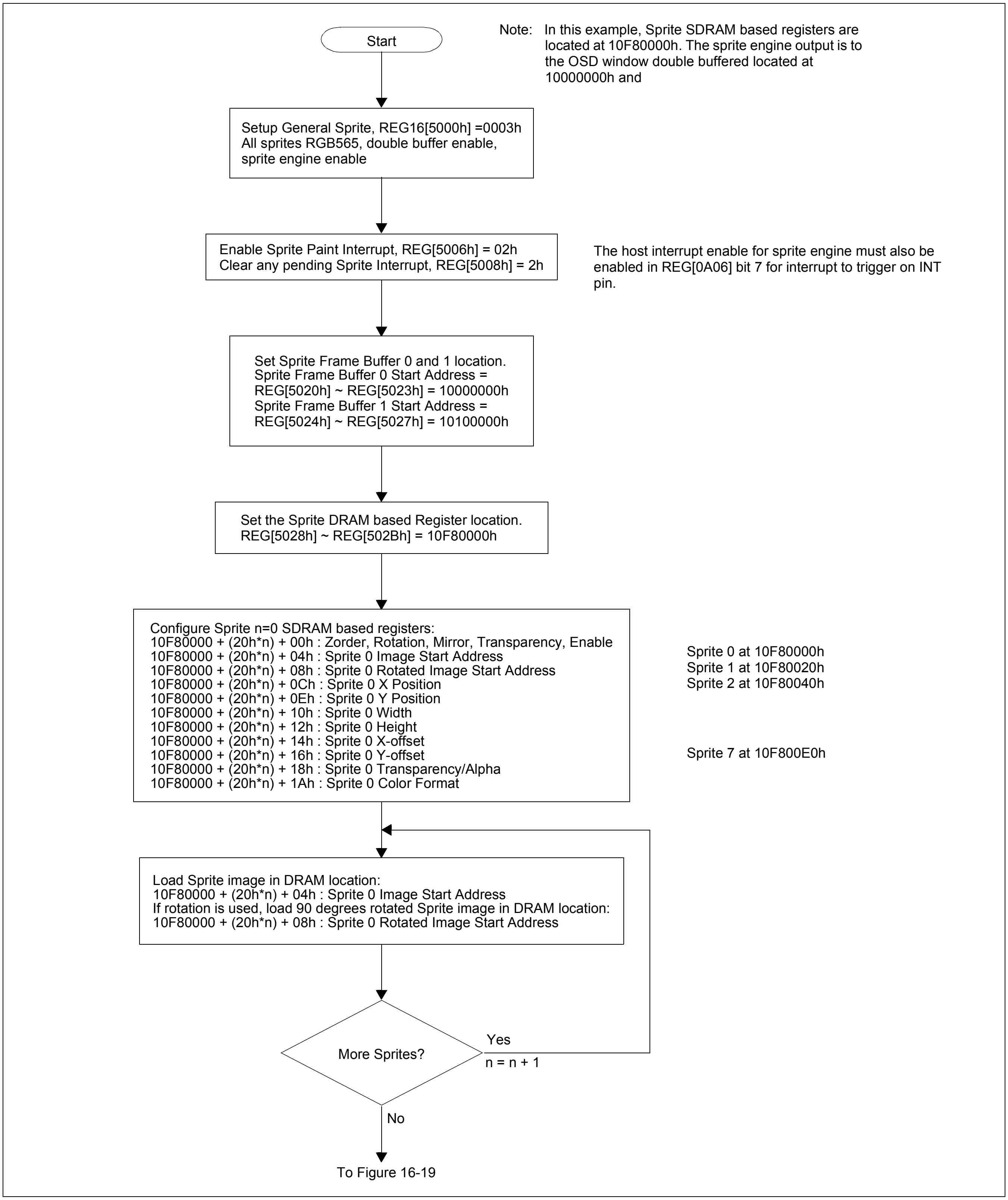


Figure 16-18: Typical Sprite Programming Flow

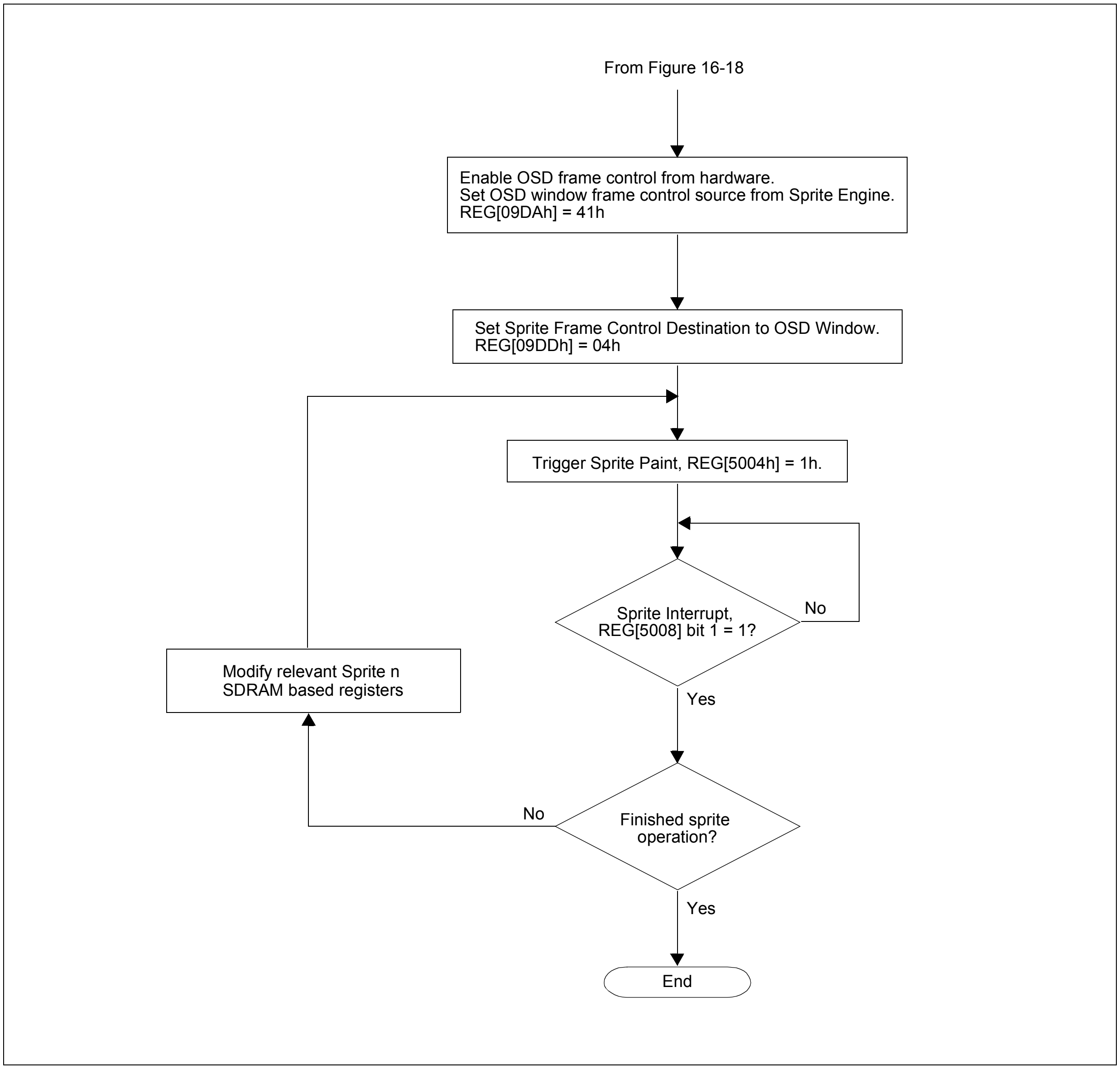


Figure 16-19: Typical Sprite Programming Flow (Continued)

Chapter 17 SDRAM Interface

In the S1D13515/S2D13515 Memory Map, the address range 1000_0000h ~ 1FFF_FFFFh is allocated for the external SDRAM. The external SDRAM interface is clocked at twice the frequency of the internal System Clock.

The SDRAM controller interface has the following features:

- Burst Length is Full Page only, not 1, 2, 4, or 8.
- Burst Type is Sequential, not Interleave.
- Standard Operation mode.
- Write Burst Mode is Burst, not Single.
- Auto-Precharge is not used.
- Power Down Mode is not supported.
- Clock Suspend Mode is not supported.
- Auto-Refresh Mode is supported with programmable refresh rate.
- Self-Refresh Mode is supported.
- CAS Latency of 2 or 3 is supported.
- Selectable timing options for tRCD, tRAS, and tRP.

Page burst accesses are always Full Page and terminated by PRECHARGE at the end of the every burst cycle. Random accesses within a page is supported.

17.1 SDRAM Device Types

There are two sets of programmable parameters which specify the type of SDRAM devices and interface to use:

- REG[3C40h] bit 1 selects 16-bit or 32-bit data bus interface.
- REG[3C40h] bits 3-2 select the SDRAM type/size.

For 16-bit data bus interface (REG[3C40h] bit 1 is 0), a single 16-bit SDRAM device can be connected to the S1D13515/S2D13515. For 32-bit data bus interface (REG[3C40h] bit 1 is 1), a single 32-bit or two 16-bit SDRAM devices can be connected to the S1D13515/S2D13515.

Note
32-bit data bus is highly recommended to avoid SDRAM bandwidth/performance limitations

REG[3C40h] bits 3-2 select the SDRAM type/size as follows:

- 00b = 4 banks x 4096 rows x 256 words/row
- 01b = 4 banks x 4096 rows x 512 words/row
- 10b = 4 banks x 8192 rows x 512 words/row

- 11b = 4 banks x 8192 rows x 1024 words/row

The following table shows different types and configurations of SDRAM devices which can be connected to the S1D13515/S2D13515:

Table 17-1 : SDRAM Configurations

REG[3C40h] bit 1	REG[3C40h] bits 3-2	SDRAM Type	SDRAM Device Connections
0b	00b	4B x 4096R x 256C	1 x 64Mbit (x16)
	01b	4B x 4096R x 512C	1 x 128Mbit (x16)
	10b	4B x 8192R x 512C	1 x 256Mbit (x16)
	11b	4B x 8192R x 1024C	1 x 512Mbit (x16)
1b (Note)	00b	4B x 4096R x 256C	1 x 128Mbit (x32) or 2 x 64Mbit (x16)
	01b	4B x 4096R x 512C	1 x 256Mbit (x32) or 2 x 128Mbit (x16)
	10b	4B x 8192R x 512C	1 x 512Mbit (x32) or 2 x 256Mbit (x16)
	11b	4B x 8192R x 1024C	1 x 1Gbit (x32) or 2 x 512Mbit (x16)

Note
32-bit data bus is highly recommended to avoid SDRAM bandwidth/performance limitations

17.2 SDRAM Timing Options

The SDRAM Controller has programmable timing options for the SDRAM interface as follows:

- REG[3C40h] bit 4 specifies the CAS Latency (2 or 3) for reads.
- REG[3C40h] bit 5 specifies the tRP timing (2 or 4 clocks).
- REG[3C40h] bit 6 specifies the tRAS timing (4 or 6 clocks).
- REG[3C40h] bit 7 specifies the tRCD timing (2 or 4 clocks).

17.2.1 tRP Timing Parameter

REG[3C40h] bit 5 is used to specify the minimum tRP (PRECHARGE-to-ACTIVE) timing between the end of a burst cycle (when PRECHARGE is issued) and the beginning of the next burst cycle (when ACTIVE is issued). If the register bit is 0, the tRP is 2 clock cycles minimum. If the register bit is 1, the tRP is 4 clock cycles minimum.

To determine the value to program into bit 5 of REG[3C40h], the following steps are required:

- From the data sheet of the SDRAM used, determine the tRP (ns) value.
- Divide the tRP value (ns) from the data sheet by the SDRAM clock period (ns). If the ratio is less than or equal to 2, REG[3C40h] bit 5 should be 0. If the ratio is greater than 2, REG[3C40h] bit 5 should be 1.

17.2.2 tRCD Timing Parameter

REG[3C40h] bit 7 is used to specify the tRCD (ACTIVE-to-READ/WRITE) timing between the beginning of a burst cycle (when ACTIVE is issued) and the READ/WRITE command. If the register bit is 0, the tRCD is 2 clock cycles. If the register bit is 1, the tRCD is 4 clock cycles.

To determine the value to program into bit 7of REG[3C40h], the following steps are required:

- From the data sheet of the SDRAM used, determine the tRCD (ns) value.
- Divide the tRCD value (ns) from the data sheet by the SDRAM clock period (ns). If the ratio is less than or equal to 2, REG[3C40h] bit 7 should be 0. If the ratio is greater than 2, REG[3C40h] bit 7 should be 1.

17.2.3 tRAS Timing Parameter

REG[3C40h] bit 6 is actually used in conjunction with the tRCD parameter (REG[3C40h] bit 7) to specify the minimum tRAS (ACTIVE-to-PRECHARGE) timing between the beginning of a burst cycle (when ACTIVE is issued) and the PRECHARGE command to terminate the burst cycle. The minimum tRAS (in number of clock cycles) is determined by the following equation:

tRAS = 4 + (2 * REG[3C40h] bit 6) + (2 * REG[3C40h] bit 7)

To determine the value to program into bit 6 of REG[3C40h], the following steps are required:

- From the data sheet of the SDRAM used, determine the tRAS (ns) value.
- Divide the tRAS value (ns) from the data sheet by the SDRAM clock period (ns). Round up the result to get the number of clock cycles required for tRAS.
- From the number of clock cycles required for tRAS, subtract 4 and then subtract the number of clock cycles programmed for tRCD. If the result is less than or equal to 0, REG[3C40h] bit 6 should be programmed to 0. If the result is greater than 0, REG[3C40h] bit 6 should be programmed to 1.

17.3 SDRAM Initialization

Before the SDRAM can be used, it has to be initialized first. The following programming sequence should be followed to initialize SDRAM:

1. Program the SDRAM Refresh Period Register (REG[3C42h] ~ REG[3C43h]) to the appropriate value according the system clock frequency and the type of SDRAM used.

2. Write the SDRAM Control Register (REG[3C40h]) with bit 0 set to 1b to start the SDRAM initialization. The value of bits 7-4 should select the appropriate timing parameters, the value of bit 2 should select the appropriate SDRAM device type used, and the value of bit 1 should select the appropriate data width (16-bit or 32-bit).

3. Keep reading bit 0 of the SDRAM Control Register (REG[3C40h]) and wait until it becomes 0b which indicates that the SDRAM initialization has finished.

Note
The following SDRAM command sequence is performed when REG[3C40h] bit 0 is set to 1:
- PRECHARGE all banks
- NOP
- Auto-Refresh
- 7 NOPs
- Auto-Refresh
- 7 NOPs
- Load Mode Register
- NOP

Some SDRAM devices may require more than two Auto-Refresh cycles before memory can be used. For these cases, additional iterations of Step 2 and 3 should be executed to provide the appropriate number of Auto-Refresh cycles required by the SDRAM device.

4. The SDRAM is now ready for use.

17.4 Self-Refresh Mode

The SDRAM Controller supports Self-Refresh mode for putting the SDRAM into a low power state while retaining its contains. Self-Refresh mode for the SDRAM is initiated by writing a 1 to REG[3C44h] bit 6. In Self-Refresh mode, the clock to the SDRAM is stopped and the SDRAM is not accessible. To exit Self-Refresh mode, write a 0 to REG[3C44h] bit 6.

Note
Before the SDRAM is placed in self-refresh mode all accesses by modules to SDRAM should be stopped and DRAM accesses must not occur. While the SDRAM is in refresh mode DRAM accesses must not occur. To access DRAM, the SDRAM must first be taken out of self-refresh mode.

Chapter 18 SDRAM Read/Write Buffer

18.1 Introduction

The external SDRAM memory of the S1D13515/S2D13515 is a shared hardware resource and is used mainly to store display and camera images, and the display and camera logic are given highest priority access to the SDRAM. The Host can access the SDRAM, but it does not have guaranteed access time to SDRAM because of bus contention/arbitration. Host bus cycles to SDRAM may take a long time.

Note
The SDRAM read/write buffer can also be used by hosts that do not have a WAIT/RDY pin to access the SDRAM

The purpose of the SDRAM Read/Write Buffer block is to alleviate this problem by buffering accesses to the SDRAM and guarantee host bus access time. The Host can do other tasks while the data is being transferred between the buffer(s) and the SDRAM.

18.2 Operation

The following is a block diagram of the SDRAM Read/Write Buffer block and where it resides in the internal logic of the S1D13515/S2D13515:

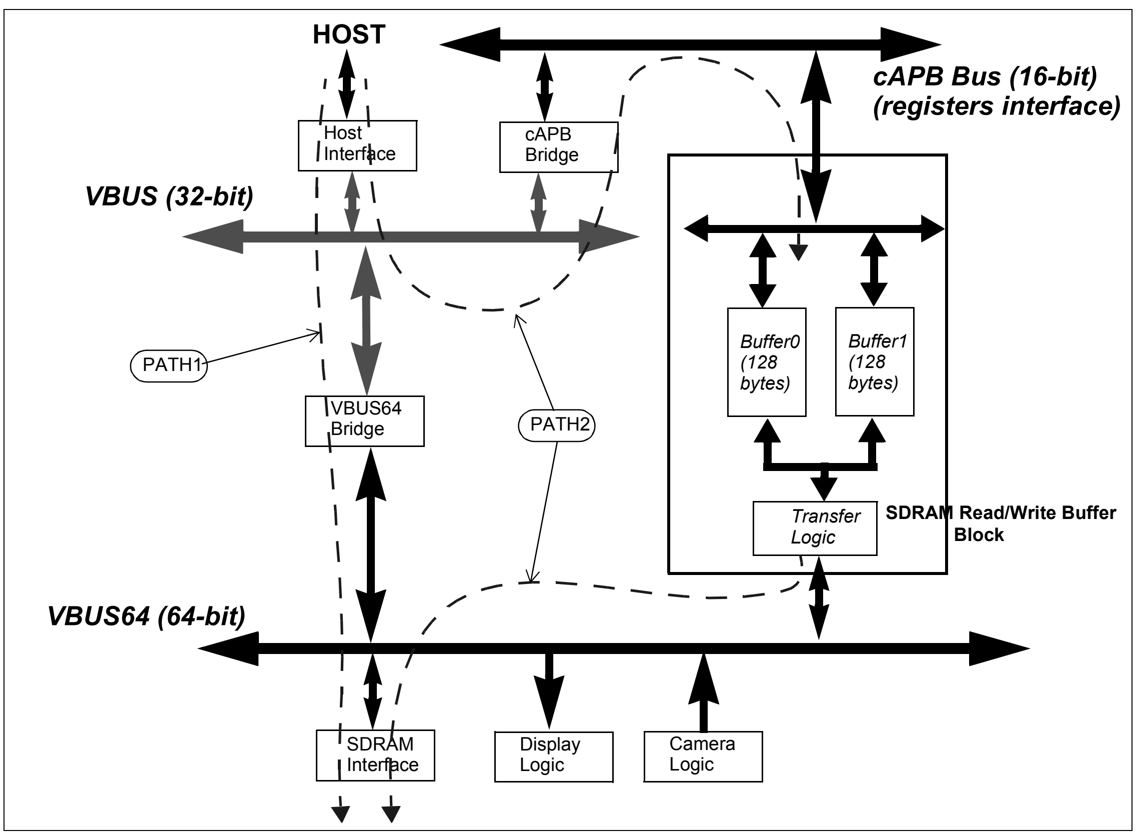


Figure 18-1: SDRAM Read/Write Buffer Block Diagram

PATH1 is the direct access to SDRAM from the Host which does not have guaranteed bus access time. PATH2 uses the SDRAM Read/Write Buffer block to access the SDRAM. There are two independent 128-byte FIFO buffers (Buffer0 and Buffer1) in the SDRAM Read/Write Buffer block which are accessed through the register space (cAPB bus). Buffer0 data is accessible at REG[024Ch] ~ REG[024Dh] or at aliased addresses in the range of REG[0300h] to REG[037Fh]. Buffer1 data is accessible at REG[025Ch] ~ REG[025Dh] or at aliased addresses in the range of REG[0380h] to REG[03FFh].

Associated with each buffer is a target address register (REG[0248h] ~ REG[024Bh] / REG[0258h] ~ REG[025Bh]) and a Mode bit (REG[0240h] / REG[0250h] bit 0). The Mode bit determines if the buffer is used for reading data from SDRAM or writing data to SDRAM.

Each buffer has an associated SDRAM Buffer Control Register (REG[0242h] / REG[0252h]) which contains 4 control/status bits (Done Interrupt Status/Clear, Rectangular Increment, Load Address, and Start). On reset, both buffers are empty.

There is an SDRAM Read/Write Buffer Internal Address Register (read-only at REG[0264h] ~ REG[0267h]) inside the SDRAM Read/Write Buffer block which holds the target SDRAM address for the data transfers between the buffers and SDRAM.

Although there are two buffers, there is only one interface to the SDRAM and the hardware will process only one buffer at a time.

18.2.1 Write Operation

The following is a description of performing writes to SDRAM through the SDRAM Read/Write Buffer block:

- If a buffer is empty (Start bit = REG[0242h] / REG[0252h] bit 0 = 0b), the Host can configure the buffer for write operation by programming the Mode bit (REG[0240h] / REG[0250h] bit 0) to 0. The Host can then write data to the buffer through the SDRAM Buffer Data Port Register 0/1 (REG[024Ch] ~ REG[024Dh] / REG[025Ch] ~ REG[025Dh]). The amount of data can be 0 to 128 bytes.
- The Host can program a target address to the SDRAM Buffer Target Address Register 0/1/2/3 (REG[0248h] ~ REG[024Bh] / REG[0258h] ~ REG[025Bh]) whenever the buffer is not busy (Start = 0).
- When the Host wants the data in the buffer to be transferred to SDRAM, it sets the Start bit to 1b to start the transfer. The Start bit stays at 1b until transfer is finished and is cleared to 0b by the hardware when transfer is finished.
- While the Start bit is 1b, writing the SDRAM Buffer Target Address Register or the SDRAM Buffer Data Port Register has no effect.
- When the Host sets the Start bit to 1b to start the transfer, it can also specify two control bits for the transfer: Load Address bit (REG[0242h] / REG[0252h] bit 1) and Rectangular Increment bit (REG[0242h] / REG[0252h] bit 2).
- If the Load Address bit is 1b on the same register write cycle when the SDRAM Buffer Control Register (REG[0242h] / REG[0252h]) is written to set the Start bit to 1b, this tells the hardware to load the SDRAM Buffer Target Address Register (REG[0248h] ~ REG[024Bh] / REG[0258h] ~ REG[024Bh]) value into its SDRAM Read/Write Buffer Internal Address Register (REG[0264h] ~ REG[0267h]) before starting the transfer. If the Load Address bit is 0b, the internal SDRAM Read/Write Buffer Internal Address Register carries on from its current value.
- If the Rectangular Increment bit is 1b on the same register write cycle when the SDRAM Buffer Control Register (REG[0242h] / REG[0252h]) is written to set the Start bit to 1b, this tells the hardware to add the value in the SDRAM Buffer Rectangular Increment Register 0/1 (REG[0260h] ~ REG[0261h] / REG[0262h] ~ REG[0263h]) to the SDRAM Read/Write Buffer Internal Address Register (REG[0264h] ~ REG[0267h]) at the end of the transfer (before the Start bit is cleared to 0b). This is used for “jumping” to the next line when writing an image/frame to SDRAM. If the Rectangular Increment bit is 0, the SDRAM Read/Write Buffer Internal Address Register just increments to the next logical address after the last byte written.
- At the end of a transfer, after the SDRAM Read/Write Buffer Internal Address Register is incremented appropriately, the hardware updates the SDRAM Buffer Target Address Register (REG[0248h] ~ REG[024Bh] / REG[0258h] ~ REG[025Bh]) with the value of the SDRAM Read/Write Buffer Internal Address Register (REG[0264h] ~ REG[0267h]) before clearing the Start bit to 0 to indicate end of transfer.
- Each buffer has an associated Done Interrupt Status bit (REG[0242h] / REG[0252h] bit 3) which gets set to 1b on the 1-to-0 transition of the Start bit at end of a transfer. The Host can clear the Done Interrupt Status bit by writing a 1b to it.

18.2.2 Read Operation

The following is a description of performing reads from SDRAM through the SDRAM Read/Write Buffer block:

- If a buffer is empty (Start bit = REG[0242h] / REG[0252h] bit 0 = 0b), the Host can configure the buffer for read operation by programming the Mode bit (REG[0240h] / REG[0250h] bit 0) to 1b.
- The Host can program a target address to the SDRAM Buffer Target Address Register 0/1/2/3 (REG[0248h] ~ REG[024Bh] / REG[0258h] ~ REG[025Bh]) whenever the buffer is not busy (Start = 0b).
- The Host can program the number of bytes to read for each transfer request by programming the SDRAM Buffer Read Bytes Register (REG[0244h] / REG[0254h]).
- When the Host wants to trigger a read from SDRAM to the buffer, it sets the Start bit to 1b to start the transfer. The hardware will start transferring the programmed number of bytes from SDRAM to the buffer. The Start bit stays at 1b until transfer is finished and is cleared to 0b by the hardware when transfer is finished.
- While the Start bit is 1b, writing the SDRAM Buffer Target Address Register or the SDRAM Buffer Data Port Register has no effect.
- When the Host sets the Start bit to 1b to start the transfer, it can also specify two control bits for the transfer: Load Address bit (REG[0242h] / REG[0252h] bit 1) and Rectangular Increment bit (REG[0242h] / REG[0252h] bit 2).
- If the Load Address bit is 1b on the same register write cycle when the SDRAM Buffer Control Register (REG[0242h] / REG[0252h]) is written to set the Start bit to 1b, this tells the hardware to load the SDRAM Buffer Target Address Register (REG[0248h] ~ REG[024Bh] / REG[0258h] ~ REG[024Bh]) value into its SDRAM Read/Write Buffer Internal Address Register (REG[0264h] ~ REG[0267h]) before starting the transfer. If the Load Address bit is 0, the internal SDRAM Read/Write Buffer Internal Address Register carries on from its current value.
- If the Rectangular Increment bit is 1b on the same register write cycle when the SDRAM Buffer Control Register (REG[0242h] / REG[0252h]) is written to set the Start bit to 1b, this tells the hardware to add the value in the SDRAM Buffer Rectangular Increment Register 0/1 (REG[0260h] ~ REG[0261h] / REG[0262h] ~ REG[0263h]) to the SDRAM Read/Write Buffer Internal Address Register (REG[0264h] ~ REG[0267h]) at the end of the transfer (before the Start bit is cleared to 0b). This is used for “jumping” to the next line when writing an image/frame to SDRAM. If the Rectangular Increment bit is 0b, the SDRAM Read/Write Buffer Internal Address Register just increments to the next logical address after the last byte written.
- At the end of a transfer, after the SDRAM Read/Write Buffer Internal Address Register is incremented appropriately, the hardware updates the SDRAM Buffer Target Address Register (REG[0248h] ~ REG[024Bh] / REG[0258h] ~ REG[025Bh]) with the value of the SDRAM Read/Write Buffer Internal Address Register (REG[0264h] ~ REG[0267h]) before clearing the Start bit to 0 to indicate end of transfer.
- Each buffer has an associated Done Interrupt Status bit (REG[0242h] / REG[0252h] bit 3) which gets set to 1b on the 1-to-0 transition of the Start bit at end of a transfer. The Host can clear the Done Interrupt Status bit by writing a 1b to it.
- After a transfer is done, the Host can read data from the buffer by reading the SDRAM Buffer Data Port Register (REG[024Ch] ~ REG[024Dh] / REG[025Ch] ~ REG[025Dh]).

18.2.3 Interrupts

Each buffer's Done Interrupt Status bit (REG[0242h] / REG[0252h] bit 3) has an associated interrupt enable bit (REG[0240h] / REG[0250h] bit 1). Each buffer's status and interrupt enable bits are ANDed together and then the two outputs are OR'ed together to go to the Interrupt Controller as a read-only SDRAM Read/Write Buffer Interrupt Status bit (REG[0A00h] bit 5).

The SDRAM Read/Write Buffer Interrupt Status can generate an interrupt to the Host if bit 5 of the Host Interrupt Enable Register 0 (REG[0A06h]) is set to 1b.

The SDRAM Read/Write Buffer Interrupt Status can generate an interrupt to the C33PE processor if bit 5 of the C33PE Device Interrupt Enable Register 0 (REG[0A0Eh]) is set to 1b.

Chapter 19 Pulse Width Modulation (PWM)

The PWM block provides two pulse-width modulation outputs (PWM1 and PWM2) which have programmable pulse modulation characteristics. Each PWM output has the following defined parameters of operation (see Figure 19-1: “PWM Timing Example” on page 513).

- A Repeat Cycle consisting of 128 Pulse Clock cycles.
- Each Pulse Clock cycle has 16 PWM Clock cycles.
- The PWM Clock cycle is derived from the PWM Source Clock and the divide ratio is programmable through the PWM Control Register (REG[0200h]) bits 3-0.
- The PWM Source Clock is derived from the System Clock and the divide ratio is programmable through the PWMSRCCLK Divide Select bits (REG[0034h] and REG[0035h]).
- At the beginning of each Repeat Cycle, the PWM output starts at 0 (OFF). The PWM On Time register (REG[0201h] / REG[0204h] bits 6-0) determines how many Pulse Clock cycles from the beginning of the Repeat Cycle the PWM output turns “on” and starts ramping up with pulses. If the PWM On Time register is 0, the PWM output will start ramping up immediately at the first Pulse Clock cycle of the Repeat Cycle.
- Each pulse cycle has 16 PWM Clock cycles and the number of PWM Clock cycles that the PWM output is high within the pulse cycle will ramp up starting from 0 PWM Clock cycles (0% duty cycle) to a maximum of number of PWM Clock cycles (max. duty cycle) as determined by the PWM Maximum Duty Cycle register (REG[0203h] / REG[0206h] bits 3-0). The increase (and decrease on ramp-down) in steps (number of PWM Clock cycles) of the duty cycle is programmable by the PWM Slope register (REG[0203h] / REG[0206h] bits 7-4), and the rate of increase/decrease of steps (time between steps) of the duty cycle for both PWM outputs is programmable by the PWM Rate register (REG[0200h] bits 7-5).
- When the duty cycle reaches the maximum within the Repeat Cycle, it stays the maximum duty cycle until ramp-down starts. The PWM Off Time register (REG[0202h] / REG[0205h]) determines how many Pulse Clock cycles from the beginning of the Repeat Cycle the PWM output turns “off” and starts ramping down. The steps of decrease of the duty cycle is the same as that of the ramp-up (PWM Slope register). When the duty cycle reaches 0% (PWM output is 0), it stays at this level until the end of the Repeat Cycle.

Note
The PWM1/2 should only be programmed when the respective PWM is disabled.

The following register settings are recommended for generating “errant-free” square waves of duty cycles 1/16 to 15/16.

REG[0034h] ~ REG[0035h] and REG[0200h] bits 3-0 determine the frequency of the square wave
REG[0200h] bits 7-5 (Rate) = 010b
REG[0201h]/REG[0204h] bits 6-0 (ON time) = 00h
REG[0202h]/REG[0205h] bits 6-0 (OFF time) = 7Fh

1/16 duty cycle: REG[0203h] = C0h, REG[0200h] bit 4 = 1b
2/16 duty cycle: REG[0203h] = C1h, REG[0200h] bit 4 = 1b
3/16 duty cycle: REG[0203h] = C2h, REG[0200h] bit 4 = 1b
4/16 duty cycle: REG[0203h] = C3h, REG[0200h] bit 4 = 1b
5/16 duty cycle: REG[0203h] = C4h, REG[0200h] bit 4 = 1b

6/16 duty cycle: REG[0203h] = C5h, REG[0200h] bit 4 = 1b
7/16 duty cycle: REG[0203h] = C6h, REG[0200h] bit 4 = 1b
8/16 duty cycle: REG[0203h] = C7h, REG[0200h] bit 4 = 1b

9/16 duty cycle: REG[0203h] = C6h, REG[0200h] bit 4 = 0b
10/16 duty cycle: REG[0203h] = C5h, REG[0200h] bit 4 = 0b
11/16 duty cycle: REG[0203h] = C4h, REG[0200h] bit 4 = 0b
12/16 duty cycle: REG[0203h] = C3h, REG[0200h] bit 4 = 0b
13/16 duty cycle: REG[0203h] = C2h, REG[0200h] bit 4 = 0b
14/16 duty cycle: REG[0203h] = C1h, REG[0200h] bit 4 = 0b
15/16 duty cycle: REG[0203h] = C0h, REG[0200h] bit 4 = 0b

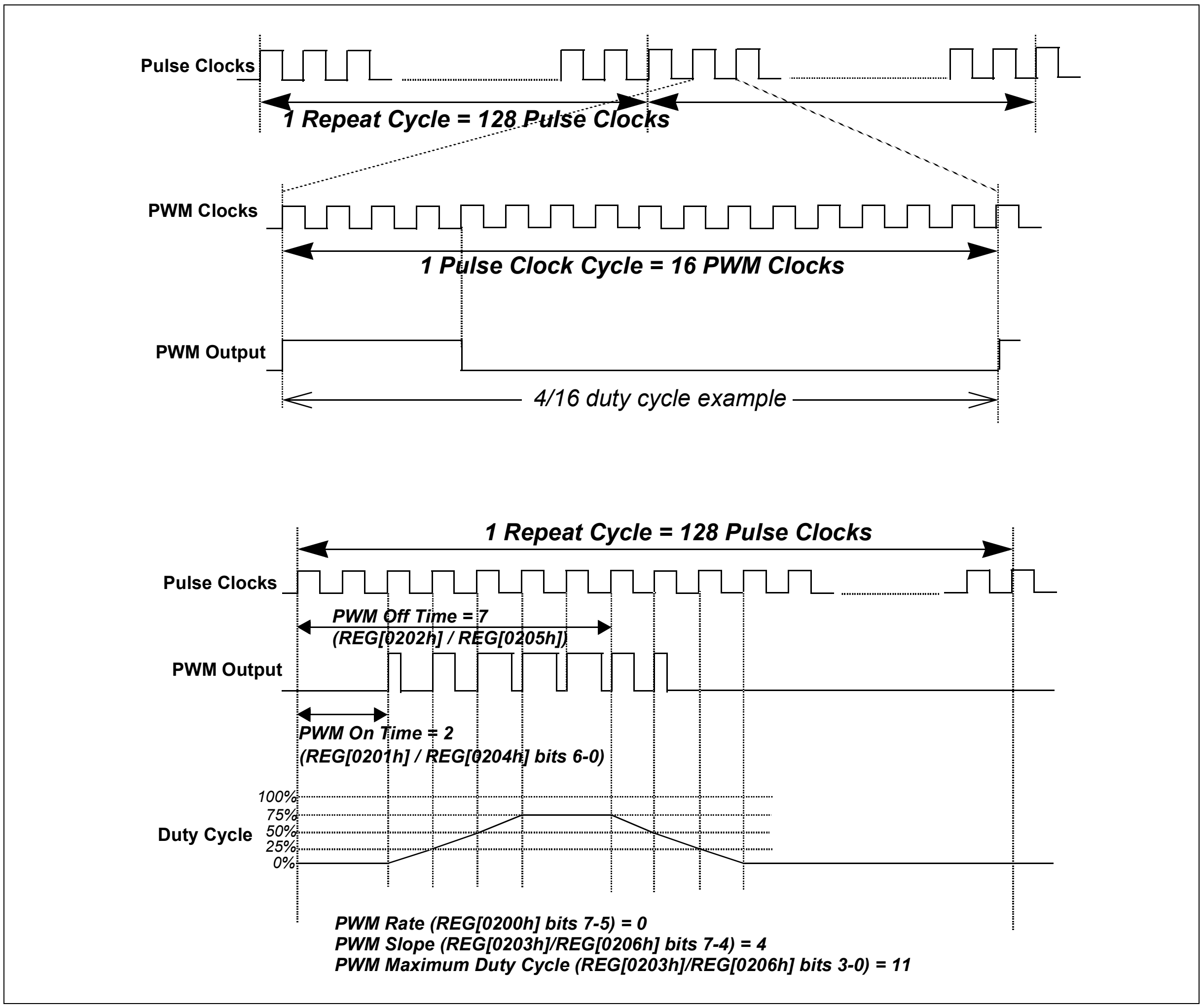


Figure 19-1: PWM Timing Example

PWM1 is enabled/disabled by programming bit 7 of REG[0201h].
PWM2 is enabled/disabled by programming bit 7 of REG[0204h].
The output polarity of both PWM1 and PWM2 outputs is programmable through REG[0200h] bit 4.

Chapter 20 General-Purpose IO Pins

Depending on the programming of the pin mapping register bits, there are up to 16 general-purpose IO (GPIO) pins available. Each GPIO pin (when available) has a programmable direction bit (REG[0180h] ~ REG[0181h]) and a pull-down enable/disable bit (REG[0184h] ~ REG[0185h]). The status and output data of the GPIO pin is read and programmed through REG[0182h] ~ REG[0183h].

GPIO[3:0]

The FP2IO18 ~ FP2IO21 pins function as GPIO0 ~ GPIO3 when the FP2IO LCD panel pins are programmed as Generic RGB output (REG[4000h] bits 5-4 = 00b) and RGB 6:6:6 color format with no serial interface (REG[4000h] bits 7-6 = 10b). See Table 5-18: “FP2IO Pin Mapping Summary (LCD2),” on page 41.

GPIO[5:4]

The FP2IO22 ~ FP2IO23 pins function as GPIO4 ~ GPIO5 when the FP2IO LCD panel pins are programmed as Generic RGB output (REG[4000h] bits 5-4 = 00b) and RGB 6:6:6 color format is used (REG[4000h] bits 7-6 = 01b or 10b). See Table 5-18: “FP2IO Pin Mapping Summary (LCD2),” on page 41.

GPIO6

The FP1IO18 pin functions as GPIO6 when LCD2 does not use FP1IOx pins (see Note 2 for Table 5-17: “FP1IO Pin Mapping Summary (LCD1 / Camera2),” on page 40) and any of the following conditions are true:

1. Camera2 Interface is enabled (REG[4000h] bit 3 = 1b)
2. FP1IOx pins are used as LCD1 output (REG[4000h] bit 3 = 0b) and the panel signals don’t have a serial interface (REG[4000h] bit 2 = 0b).

GPIO7

The FP1IO19 pin functions as GPIO7 when LCD2 does not use FP1IOx pins (see Note 2 for Table 5-17: “FP1IO Pin Mapping Summary (LCD1 / Camera2),” on page 40) and any of the following conditions are true:

1. FP1IOx pins are used as LCD1 output (REG[4000h] bit 3 = 0b) and the panel signals don’t have a serial interface (REG[4000h] bit 2 = 0b).
2. Camera2 interface is enabled (REG[4000h] bit 3 = 1b) and RGB Data Stream interface is selected (REG[0D46h] bit 2 = 1b).
3. Camera2 interface is enabled (REG[4000h] bit 3 = 1b), 8-bit Camera interface is selected (REG[0D46h] bit 2 = 0b), and the Keypad interface signals are mapped to the Host Interface pins (REG[0186h] bit 5 = 0b).

Note
GPIO7 is not available when the Keypad Interface is configured to use the FP1IO pins, REG[0186h] bit 5 = 1b

GPIO8

GPIO8 is mapped to either FP1IO14 or AB17 as determined by REG[0186h] bit 5.

When REG[0186h] bit 5 = 0b, GPIO8 is mapped to FP1IO14 when LCD2 does not use FP1IOx pins (see Note 2 in Table 5-17), Camera2 interface is enabled (REG[4000h] bit 3 = 1b), and 8-bit Camera interface is selected (REG[0D46h] bit 2 = 0b).

When REG[0186h] bit 5 = 1b, GPIO8 is mapped to AB17 if the host interface selected does not use AB17 (see Table 5-15 “Host Interface Pin Mapping 3,” on page 37.

GPIO9

GPIO9 is mapped to either FP1IO15 or M/R# as determined by REG[0186h] bit 5.

REG[0186h] bit 5 = 0b
GPIO9 is mapped to FP1IO15 when Camera2 interface is enabled (REG[4000h] bit 3 = 1b) and 8-bit Camera interface is selected (REG[0D46h] bit 2 = 0b).

REG[0186h] bit 5 = 1b
GPIO9 is mapped to M/R# if the host interface selected does not use M/R# (see Table 5-15 “Host Interface Pin Mapping 3,” on page 37).

GPIO10

GPIO10 is mapped to either FP1IO16 or AB20 as determined by REG[0186h] bit 5.

REG[0186h] bit 5 = 0b
GPIO10 is mapped to FP1IO16 when Camera2 interface is enabled (REG[4000h] bit 3 = 1b) and 8-bit Camera interface is selected (REG[0D46h] bit 2 = 0b).

REG[0186h] bit 5 = 1b
GPIO10 is mapped to AB20 if the host interface selected does not use AB20 (see Table 5-15 “Host Interface Pin Mapping 3,” on page 37).

GPIO11

GPIO11 is mapped to either FP1IO17 or AB13 as determined by REG[0186h] bit 5.

REG[0186h] bit 5 = 0b
GPIO11 is mapped to FP1IO17 when LCD2 does not use FP1IOx pins (see Note 2 in Table 5-17: “FP1IO Pin Mapping Summary (LCD1 / Camera2),” on page 40), Camera2 interface is enabled (REG[4000h] bit 3 = 1b), and 8-bit Camera interface is selected (REG[0D46h] bit 2 = 0b).

REG[0186h] bit 5 = 1b
GPIO11 is mapped to AB13 if the host interface selected does not use AB13 (see Table 5-15 “Host Interface Pin Mapping 3,” on page 37).

GPIO12

GPIO12 is mapped to either FP1IO20 or AB19 as determined by REG[0186h] bit 5.

REG[0186h] bit 5 = 0b
GPIO12 is mapped to FP1IO20 when Camera2 interface is enabled (REG[4000h] bit 3 = 1b) and 8-bit Camera interface is selected (REG[0D46h] bit 2 = 0b).

REG[0186h] bit 5 = 1b
GPIO12 is mapped to AB19 if the host interface selected does not use AB19 (see Table 5-15 “Host Interface Pin Mapping 3,” on page 37).

GPIO[15:13]

GPIO[15:13] are mapped to either FP1IO[23:21] or AB[14:16] as determined by REG[0186h] bit 5.

REG[0186h] bit 5 = 0b
GPIO[15:13] are mapped to FP1IO[23:21] when Camera2 interface is enabled (REG[4000h] bit 3 = 1b) and 8-bit Camera interface is selected (REG[0D46h] bit 2 = 0b).

REG[0186h] bit 5 = 1b
GPIO[15:13] are mapped to AB[14:16] if the host interface selected does not use AB[14:16] (see Table 5-15 “Host Interface Pin Mapping 3,” on page 37).

Chapter 21 Host Interface

21.1 Overview

Note
The S1D/S2D13515 supports Little Endian interface only.

The S1D13515/S2D13515 supports multiple types of host interfaces. The host interfaces can be 8-bit or 16-bit data and categorized as follows:

- Parallel Direct 8-bit
    - Intel80 Type1, Intel80 Type2
    - NEC V850 Type1, NEC V850 Type2
    - Renesas SH4
- Parallel Direct 16-bit
    - Intel80 Type1, Intel80 Type2
    - NEC V850 Type1, NEC V850 Type2
    - Renesas SH4
    - Marvell PXA3xx
    - TI TMS470
    - MPC555
- Parallel Indirect 8-bit
    - Intel80 Type1, Intel80 Type2
    - NEC V850 Type1, NEC V850 Type2
    - Renesas SH4
- Parallel Indirect 16-bit
    - Intel80 Type1, Intel80 Type2
    - NEC V850 Type1, NEC V850 Type2
    - Renesas SH4
    - TI TMS470
    - MPC555
- Serial Indirect
    - SPI
    - I2C

There are two dedicated input configuration pins, CNF1 and CNF2, which are used for selecting the host interface type. Due to pin limitations, depending on the state of the CNF1 and CNF2 pins, some of the host interface pins are also used as configuration pins for selecting the host interface type.

The TI TMS470 and MPC555 interfaces require the most number of pins and they are selected by CNF1=1. If CNF1=1, the CNF2 is used to select between the TI TMS470 and MPC555. If CNF2=0, TI TMS470 is selected. If CNF2=1, MPC555 is selected.

For TI TMS470, only 16-bit Direct and 16-bit Indirect is supported and the Indirect/Direct mode is selected by the AB0 pin.

For the MPC555, only 16-bit Direct and 16-bit Indirect is supported. The BE1# pin is used to select between Indirect (=0) and Direct (=1).

If CNF1=0, the TEA#, BDIP#, and BURST# input pins are not used for the host interface (because they are only needed for TI TMS470 and MPC555) and they are used as host configuration pins (CNF3, CNF4, and CNF5, respectively) to select the other host interface types. The CNF2 pin is used to select between 8-bit (CNF2=0) or 16-bit (CNF2=1) interface and the CNF3 pin (TEA#) is used to select between Direct (CNF3=1) or Indirect (CNF3=0) mode.

For Indirect 8-bit and 16-bit modes (CNF3=0), the upper address lines are not used for the host interface and the AB3 pin is used as the CNF6 configuration pin. The CNF4 (BDIP#), CNF5 (BURST#), and CNF6 (AB3) pins are used to select the Indirect host interface type.

For Direct 8-bit modes (CNF3=1, CNF2=0), the BE1# pin is not used for the host interface and it is used as the CNF6 configuration pin. The CNF4 (BDIP#), CNF5 (BURST#), and CNF6 (BE1#) pins are used to select the Direct 8-bit host interface type.

For Direct 16-bit modes (CNF3=1, CNF2=1), the AB0 in is not used for the host interface and it is used as the CNF6 configuration pin. The CNF4 (BDIP#), CNF5 (BURST#), and CNF6 (AB0) pins are used to select the Direct 16-bit host interface type.

For the serial interfaces, [CNF4, CNF5, CNF6] = 011b, and the AB4 pin is used as the CNF7 pin to select between SPI and I2C. The selection of the serial host interfaces is repeated/mirrored in all 4 combinations of [CNF2, CNF3]. For one of the SPI serial host interface selections, [CNF2, CNF3]=10b, [CNF4, CNF5, CNF6]=011b, and CNF7=0b, the unused host interface pins are used as the RGB streaming input interface for Camera1.

For the Indirect host interfaces and the Marvell PXA3xx host interface, some of the unused host interface pins are used as GPIO or Keypad function.

21.2 Intel80 Type1 Interface

The following table shows the pins used for the Intel80 Type1 interface:

Table 21-1: Intel80 Type1 Host Interface Signals

S1D13515/S2D13515 Pin	Intel80 Type1 8-bit Indirect	Intel80 Type1 16-bit Indirect	Intel80 Type1 8-bit Direct	Intel80 Type1 16-bit Direct
CS#	CS#	CS#	CS#	CS#
RD#	RD#	RD#	RD#	RD#
RD/WR#	WE#	WE#	WE#	WE#
WAIT#	WAIT#	WAIT#	WAIT#	WAIT#
BE0#	-	0[2]	-	LBE#[1]
BE1#	-	0[2]	-	UBE#[1]
DB15 - DB8	-	D15 - D8	-	D15 - D8
DB7 - DB0	D7 - D0	D7 - D0	D7 - D0	D7 - D0
M/R#	-	-	Address Line[3] or Output from Host	Address Line[3] or Output from Host
AB20 - AB3	-	-	A20 - A3	A20 - A3
AB2	-	A2	-	A2
AB1	A1	A1	A1	A1
AB0	A0	-	A0	-

Note

1. In 16-bit Direct mode, the LBE# and UBE# pins are used as byte enables for reads and writes. For reads, both D15-D8 and D7-D0 bytes are driven, but the data on D15-D8 / D7-D0 is only valid if UBE# / LBE# is 0.
2. In 16-bit Indirect mode, the BE0# and BE1# input pins should be tied or pulled down to 0 and only 16-bit accesses should be performed on the host interface bus. The host can still perform 8-bit accesses using bit 15 of the INDEX[15:0] register. If INDEX[0]=0, the byte data is in D7-D0. If INDEX[0]=1, the byte data is in D15-D8.
3. For Direct modes, the connection to the M/R# input of the S1D13515/S2D13515 can be an upper address line or an output pin from the Host to select Memory or Register space.

21.3 Intel80 Type2 Interface

The following table shows the pins used for the Intel80 Type2 interface:

Table 21-2: Intel80 Type2 Host Interface Signals

S1D13515/S2D13515 Pin	Intel80 Type2 8-bit Indirect	Intel80 Type2 16-bit Indirect	Intel80 Type2 8-bit Direct	Intel80 Type2 16-bit Direct
CS#	CS#	CS#	CS#	CS#
RD#	RD#	RD#	RD#	RD#[1]
BE0#	WE#	WE#[2]	WE#	WEL#[1]
BE1#	-	WE#[2]	-	WEU#[1]
WAIT#	WAIT#	WAIT#	WAIT#	WAIT#
DB15 - DB8	-	D15 - D8	-	D15 - D8
DB7 - DB0	D7 - D0	D7 - D0	D7 - D0	D7 - D0
M/R#	-	-	Address Line[3] or Output from Host	Address Line[3] or Output from Host
AB20 - AB3	-	-	A20 - A3	A20 - A3
AB2	-	A2	-	A2
AB1	A1	A1	A1	A1
AB0	A0	-	A0	-

Note

1. In 16-bit Direct mode, the WEL# and WEU# pins are used as byte write enables. Reads are always 16-bit.
2. In 16-bit Indirect mode, only 16-bit writes should be performed on the host interface bus and the BE0# and BE1# inputs should be connected to a write enable signal (WEL# or WEU#). The host can still perform 8-bit writes using bit 15 of the INDEX[15:0] register. If INDEX[0]=0, the byte data is in D7-D0. If INDEX[0]=1, the byte data is in D15-D8.
3. For Direct modes, the connection to the M/R# input of the S1D13515/S2D13515 can be an upper address line or an output pin from the Host to select Memory or Register space.

21.4 NEC V850 Type1 Interface

The following table shows the pins used for the NEC V850 Type1 interface:

Table 21-3: NEC V850 Type1 Host Interface Signals

S1D13515/S2D13515 Pin	NEC V850 Type1 8-bit Indirect	NEC V850 Type1 16-bit Indirect	NEC V850 Type1 8-bit Direct	NEC V850 Type1 16-bit Direct
CS#	CS#	CS#	CS#	CS#
RD#	DSTB#	DSTB#	DSTB#	DSTB#
RD/WR#	R/W#	R/W#	R/W#	R/W#
WAIT#	WAIT#	WAIT#	WAIT#	WAIT#
BE0#	-	0[2]	-	LBEN#[1]
BE1#	-	0[2]	-	UBEN#[1]
BUSCLK	CLK	CLK	CLK	CLK
DB15 - DB8	-	D15 - D8	-	D15 - D8
DB7 - DB0	D7 - D0	D7 - D0	D7 - D0	D7 - D0
M/R#	-	-	Address Line[3] or Output from Host	Address Line[3] or Output from Host
AB20 - AB3	-	-	A20 - A3	A20 - A3
AB2	-	A2	-	A2
AB1	A1	A1	A1	A1
AB0	A0	-	A0	-

Note

1. In 16-bit Direct mode, the LBEN# and UBEN# pins are used as byte enables for reads and writes. For reads, both D15-D8 and D7-D0 bytes are driven, but the data on D15-D8 / D7-D0 is only valid if UBEN# / LBEN# is 0.
2. In 16-bit Indirect mode, the BE0# and BE1# input pins should be tied or pulled down to 0 and only 16-bit accesses should be performed on the host interface bus. The host can still perform 8-bit accesses using bit 15 of the INDEX[15:0] register. If INDEX[0]=0, the byte data is in D7-D0. If INDEX[0]=1, the byte data is in D15-D8.
3. For Direct modes, the connection to the M/R# input of the S1D13515/S2D13515 can be an upper address line or an output pin from the Host to select Memory or Register space.

21.5 NEC V850 Type2 Interface

The following table shows the pins used for the NEC V850 Type2 interface:

Table 21-4: NEC V850 Type2 Host Interface Signals

S1D13515/S2D13515 Pin	NEC V850 Type2 8-bit Indirect	NEC V850 Type2 16-bit Indirect	NEC V850 Type2 8-bit Direct	NEC V850 Type2 16-bit Direct
CS#	CS#	CS#	CS#	CS#
RD#	RD#	RD#	RD#	RD#
BE0#	WR#	WR#[2]	WR#	WRL#[1]
BE1#	-	WR#[2]	-	WRH#[1]
BUSCLK	CLK	CLK	CLK	CLK
WAIT#	WAIT#	WAIT#	WAIT#	WAIT#
DB15 - DB8	-	D15 - D8	-	D15 - D8
DB7 - DB0	D7 - D0	D7 - D0	D7 - D0	D7 - D0
M/R#	-	-	Address Line[3] or Output from Host	Address Line[3] or Output from Host
AB20 - AB3	-	-	A20 - A3	A20 - A3
AB2	-	A2	-	A2
AB1	A1	A1	A1	A1
AB0	A0	-	A0	-

Note

1. In 16-bit Direct mode, the WRL# and WRH# pins are used as byte write enables. Reads are always 16-bit.
2. In 16-bit Indirect mode, only 16-bit writes should be performed on the host interface bus and the BE0# and BE1# inputs should be connected to a write enable signal (WRL# or WRH#). The host can still perform 8-bit writes using bit 15 of the INDEX[15:0] register. If INDEX[0]=0, the byte data is in D7-D0. If INDEX[0]=1, the byte data is in D15-D8.
3. For Direct modes, the connection to the M/R# input of the S1D13515/S2D13515 can be an upper address line or an output pin from the Host to select Memory or Register space.

21.6 Renesas SH4 Interface

The following table shows the pins used for the Renesas SH4 interface:

Table 21-5: Renesas SH4 Host Interface Signals

S1D13515/S2D13515 Pin	Renesas SH4 8-bit Indirect	Renesas SH4 16-bit Indirect	Renesas SH4 8-bit Direct	Renesas SH4 16-bit Direct
CS#	CS#	CS#	CS#	CS#
RD#	RD#	RD#	RD#	RD#
RD/WR#	WR#	-	WR#	-
BE0#	-	WE#[2]	-	WE0#[1]
BE1#	-	WE#[2]	-	WE1#[1]
WAIT#	RDY#	RDY#	RDY#	RDY#
BS#	BS#	BS#	BS#	BS#
BUSCLK	CLK	CLK	CLK	CLK
DB15 - DB8	-	D15 - D8	-	D15 - D8
DB7 - DB0	D7 - D0	D7 - D0	D7 - D0	D7 - D0
M/R#	-	-	Address Line[3] or Output from Host	Address Line[3] or Output from Host
AB20 - AB3	-	-	A20 - A3	A20 - A3
AB2	-	A2	-	A2
AB1	A1	A1	A1	A1
AB0	A0	-	A0	-

Note

1. In 16-bit Direct mode, the WE0# and WE1# pins are used as byte write enables. Reads are always 16-bit.
2. In 16-bit Indirect mode, only 16-bit writes should be performed on the host interface bus and the BE0# and BE1# inputs should be connected to a write enable signal (WE0# or WE1#). The host can still perform 8-bit accesses using bit 15 of the INDEX[15:0] register. If INDEX[0]=0, the byte data is in D7-D0. If INDEX[0]=1, the byte data is in D15-D8.
3. For Direct modes, the connection to the M/R# input of the S1D13515/S2D13515 can be an upper address line or an output pin from the Host to select Memory or Register space.

21.7 Marvell PXA3xx Interface

The following table shows the pins used for the Marvell PXA3xx VLIO interface:

Table 21-6: Marvell PXA3xx VLIO Interface Signals

S1D13515/S2D13515 Pin	Marvell PXA3xx VLIO 16-bit Direct
CS#	CS#
RD#	DF_nOE
RD/WR#	DF_nWE
WAIT#	RDY
BE0#	nBE0[4]
BE1#	nBE1[4]
DB15 - DB8	DF_IO15 - DF_IO8
DB7 - DB0	DFIO7 - DFIO0
AB6	nLUA[1,2]
AB5	nLLA[1,2]
AB4 - AB1	DF_ADDR3 - DF_ADDR0[1,3]

Note

1. The Marvell PXA3xx VLIO interface is assumed to be 16-bit and addresses are 16-bit word addresses, not byte addresses. The word address is latched by nLUA and nLLA and DF_ADDR3-DF_ADDR0 is assumed to be word address.
2. Bit 21 of the latched word address (bit 22 of the internal byte address) is used as M/R# internally to select Memory or Register space. Bits 19-0 of the latched word address are used as byte address bits 20-1 internally.
3. Bits 4-1 of the internal byte address will always use bits 3-0 of the latched word address immediately after the word address is latched by nLUA and/or nLLA. At the end of the first read or write following address latching, bits 4-1 of the internal byte address will switch to using DF_ADDR3 - DF_ADDR0 for subsequent reads or writes in a burst unless another address latching (nLUA and/or nLLA pulsing low) occurs, in which case bits 4-1 of the internal byte address switches back to using bits 3-0 of the latched word address.
4. The nBE0 and nBE1 pins are used as byte enables for reads and writes. For reads, both DFIO15-DF_IO8 and DF_IO7-DF_IO0 bytes are driven, but the data on DF_IO15-DF_IO8 / DF_IO7-DF_IO0 is only valid if nBE1 / nBE0 is 0.

21.8 TI TMS470 Interface

The following table shows the pins used for the TI TMS470 interface:

Table 21-7: TI TMS470 Host Interface Signals

S1D13515/S2D13515 Pin	TI TMS470 16-bit Indirect	TI TMS470 16-bit Direct
CS#	CS#	CS#
RD#	OE#	OE#
RD/WR#	RD/WR#	RD/WR#
WAIT#	TA#	TA#
BE0#	0[3]	LB#[2]
BE1#	0[3]	UB#[2]
BS#	TS#	TS#
BURST#	BURST#	BURST#
BDIP#	BDIP#	BDIP#
TEA#	ERR_ACK#	ERR_ACK#
BUSCLK	CLK	CLK
DB15 - DB8	D15 - D8	D15 - D8
DB7 - DB0	D7 - D0	D7 - D0
M/R#	-	Address Line[4] or Output from Host
AB20 - AB3	-	A20 - A3
AB2	A2	A2
AB1	A1	A1

Note

1. For burst access, the burst length is 2 because the data bus width is 16-bit and largest word to transfer is 32-bit.
2. In 16-bit Direct mode, the LB# and UB# pins are used as byte enables for reads and writes. For reads, both D15-D8 and D7-D0 bytes are driven, but the data on D15-D8 / D7-D0 is only valid if UB# / LB# is 0.
3. In 16-bit Indirect mode, the BE0# and BE1# input pins should be tied or pulled down to 0 and only 16-bit accesses should be performed on the host interface bus. The host can still perform 8-bit accesses using bit 15 of the INDEX[15:0] register. If INDEX[0]=0, the byte data is in D7-D0. If INDEX[0]=1, the byte data is in D15-D8.
4. For Direct modes, the connection to the M/R# input of the S1D13515/S2D13515 can be an upper address line or an output pin from the Host to select Memory or Register space.

21.9 MPC555 Interface

The S1D13515/S2D13515 does not support Big Endian configuration. The host processor must be configured for Little Endian External Bus when connected to the S1D13515/S2D13515 using the MPC555 interface

The following table shows the pins used for the MPC555 interface:

Table 21-8: MPC555 Host Interface Signals

S1D13515/S2D13515 Pin	MPC555 16-bit Indirect	MPC555 16-bit Direct
CS#	CS#	CS#
RD#	1[3,4]	TSIZ0[1,2]
RD/WR#	RD/WR#	RD/WR#
WAIT#	BI#	BI#
BE0#	0[3,4]	TSIZ1[1,2]
BE1#	0[6]	1[6]
BS#	TS#	TS#
BURST#	BURST#	BURST#
BDIP#	BDIP#	BDIP#
TEA#	TEA#	TEA#
BUSCLK	CLK	CLK
DB15 - DB8	D0 - D7	D0 - D7
DB7 - DB0	D8 - D15	D8 - D15
M/R#	-	Address Line[5] or Output from Host
AB20 - AB3	-	A11 - A28
AB2	A29	A29
AB1	A30	A30
AB0	-[3,4]	A31[1,2]

Note

1. In 16-bit Direct mode, non-burst access, the TSIZ1, TSIZ0, A31 pins are decoded to generate byte enables internally for reads and writes. For reads, both D0-D7 and D8-D15 bytes are driven, but the data on D0-D7 / D8-D15 is only valid if the byte lane is enabled.
2. In 16-bit Direct mode, burst access, each word transferred (TA# low) is assumed to be 16-bit. An internal address counter is initially loaded with the address in A11-A30 and increments by 2 for each word transferred, and it wraps around modulo 16. For example, if the starting address is 8, the internal address increment sequence will be 8->A->C->E->0->2->.....
3. In 16-bit Indirect modes, the TSIZ0 pin should be tied or pulled up to 1, the TSIZ1 pin should be tied or pulled down to 0, and only 16-bit accesses should be performed on the host interface bus. The host can still perform 8-bit accesses using bit 15 of the INDEX[15:0] register. If INDEX[0]=0, the byte data is in D8-D15. If INDEX[0]=1, the byte data is in D0-D7.
4. There are only 3 registers/ports available in Indirect mode: INDEX ([A29,A30] = 00b), DATA ([A29,A30] = 01b), and CONTROL ([A29,A30] = 10b). The following are the types of burst accesses which can occur in Indirect 16-bit mode:

- Burst Length = 1:
  - Read or Write INDEX
  - Read or Write DATA
  - Read or Write CONTROL

5. For Direct mode, the connection to the M/R# input of the S1D13515/S2D13515 can be an address line (A0 - A10) or an output pin from the Host to select Memory or Register space.
6. For the MPC555 interface, the BE1# pin is used to select whether the interface uses Indirect or Direct addressing.

21.10 SPI Host Interface

The following table shows the pins used for the SPI Host interface:

Table 21-9: SPI Host Interface Signals

S1D13515/S2D13515 Pin	SPI Host
CS#	HSCS#
RD/WR#	HSDI (Host-->S1D13515/S2D13515)
WAIT#	HSDO (S1D13515/S2D13515--> Host)
BE0#	HSCK
AB5	SPICLKSEL

The SPI host module requires a valid clock selection before the interface can operate. The SPI host module clock selection is determined by a combination of SPICLKEN (AB5) pin and REG[0061h] bits 2 and 0.

The SPI serial host interface is a byte-based interface and operates in a similar manner as the Parallel Indirect 8-bit host interfaces (see Section 21.12, “Host Interface Access Methods” on page 536). Each SPI transfer cycle always starts with a Command byte followed by subsequent bytes which are determined by the Command byte. The Commands have similar functions as the read/write cycles of Parallel Indirect host interfaces. The following are the Command bytes defined for the SPI serial host interface:

Table 21-10: SPI Host Interface Commands

Command[7:0]	Name	Description
00000000b (00h)	Write INDEX[15:0]	This is the “Write INDEX[15:0]” command. It writes the INDEX[15:0] register. The Command bytes is followed by two byte writes. The first byte after the Command byte is the INDEX[7:0] value and the second byte is the INDEX[15:8] value.
10000000b (80h)	Read INDEX[15:0]	This is the “Read INDEX[15:0]” command. It reads the INDEX[15:0] register. The Command byte (write) is followed by two read bytes. The first byte after the Command byte is the INDEX[7:0] value and the second byte is the INDEX[15:8] value.
00000001b (01h)	Write DATA Port	This is the “Write DATA Port” command. It is used to write a sequence of bytes to the DATA port. The Command byte is followed by a sequence of data bytes to write. If the Auto-Increment bit (CONTROL[0]) is set to 1, the INDEX is incremented by 1 for each byte of data until the chip-select pin goes high. If the Auto-Increment bit is 0, INDEX does not increment.
10000001b (81h)	Read DATA Port	This is the “Read DATA Port” command. It is used to read a sequence of bytes from the DATA port. The Command byte is followed by a sequence of data bytes to read. If the Auto-Increment bit (CONTROL[0]) is set to 1, the INDEX is incremented by 1 for each byte of data until the chip-select pin goes high. If the Auto-Increment bit is 0, INDEX does not increment.
00000010b (02h)	Write CONTROL[7:0]	This is the “Write CONTROL[7:0]” command. It writes the CONTROL[7:0] register. (Currently only CONTROL[0] is defined and CONTROL[7:1] are reserved for future use.) The Command byte is followed by a byte write which contains the CONTROL[7:0] value.
10000010b (82h)	Read CONTROL[7:0]	This is the “Read CONTROL[7:0]” command. It reads the CONTROL[7:0] register. The Command byte is followed by a byte read which contains the CONTROL[7:0] value.

The following figures show the transfer cycles for the SPI host interface Commands:

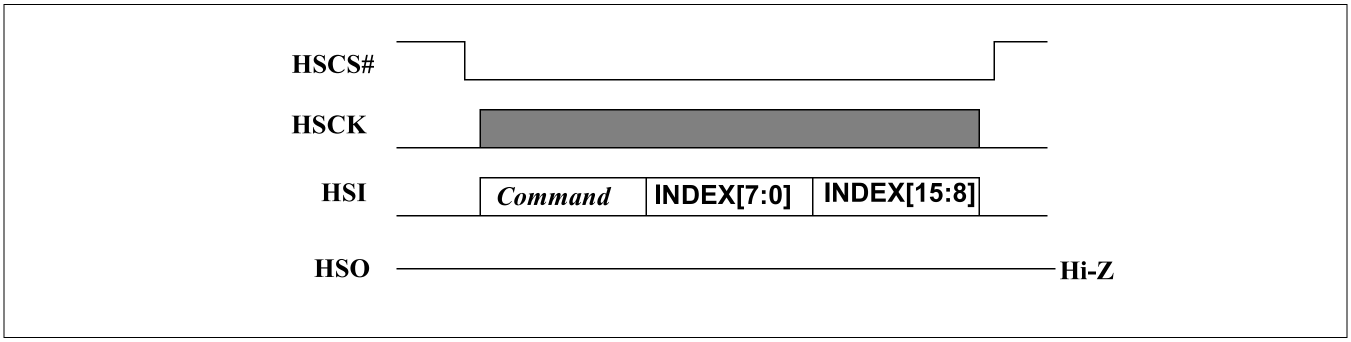


Figure 21-1: SPI Host "Write INDEX" Command Transfer Cycle

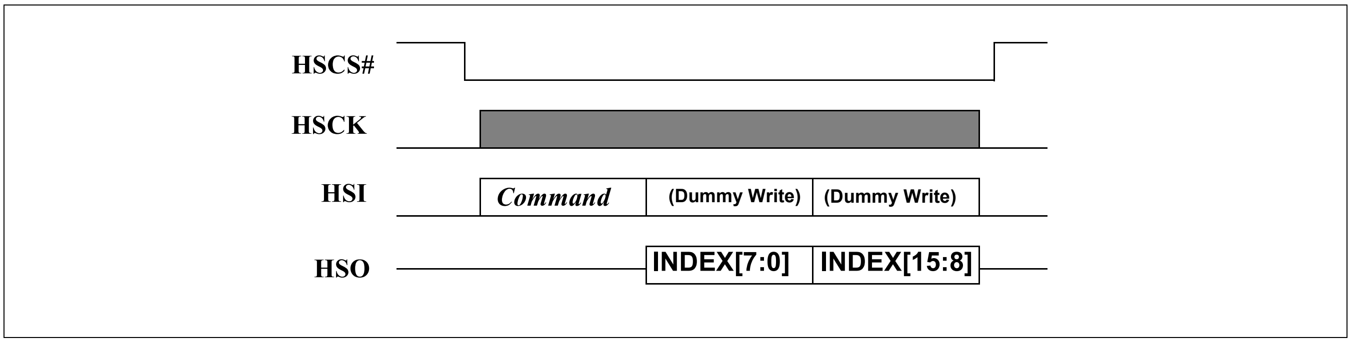


Figure 21-2: SPI Host "Read INDEX" Command Transfer Cycle

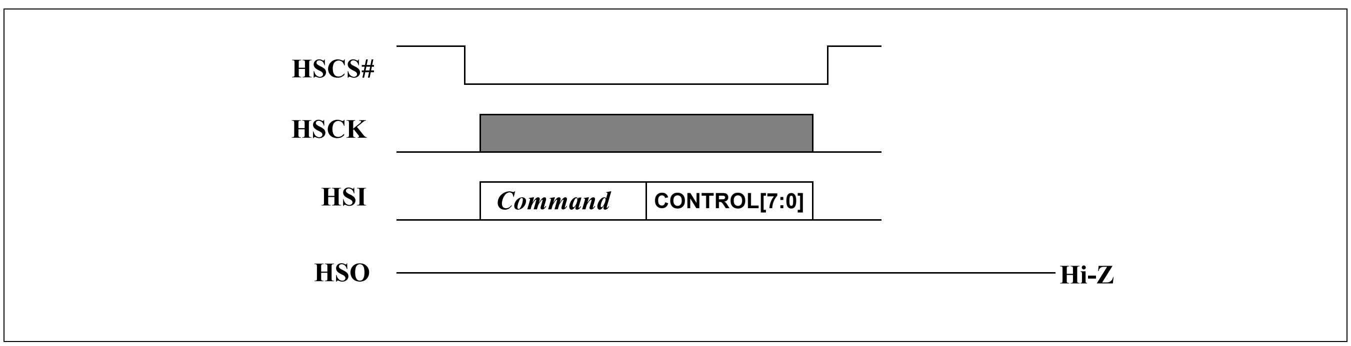


Figure 21-3: SPI Host "Write CONTROL" Command Transfer Cycle

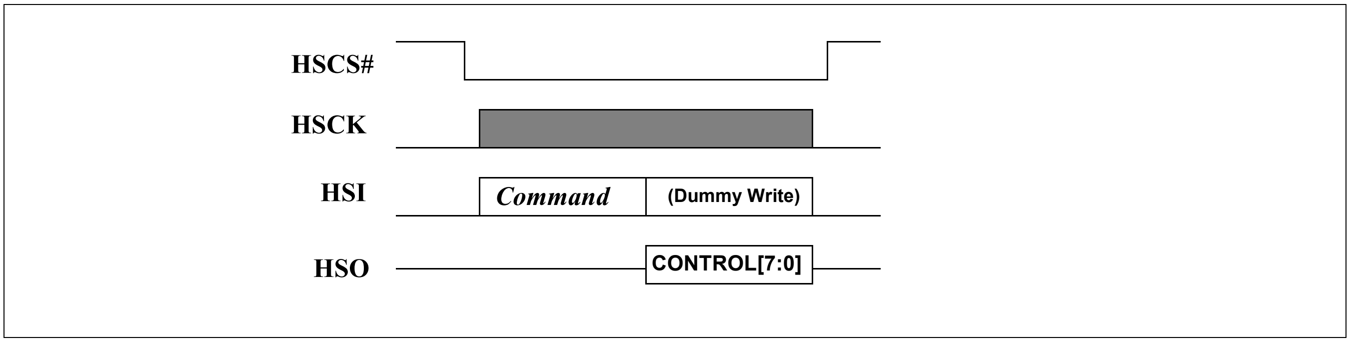


Figure 21-4: SPI Host "Read CONTROL" Command Transfer Cycle

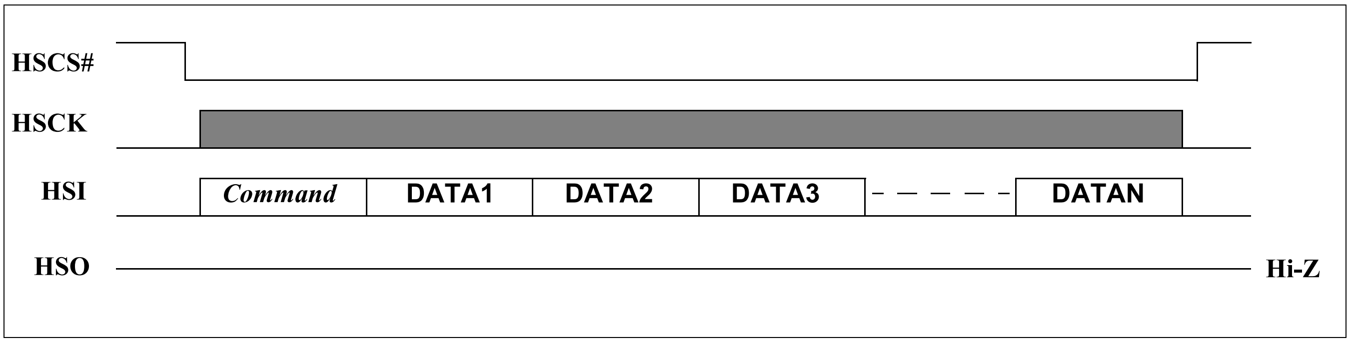


Figure 21-5: SPI Host "Write DATA" Command Transfer Cycle

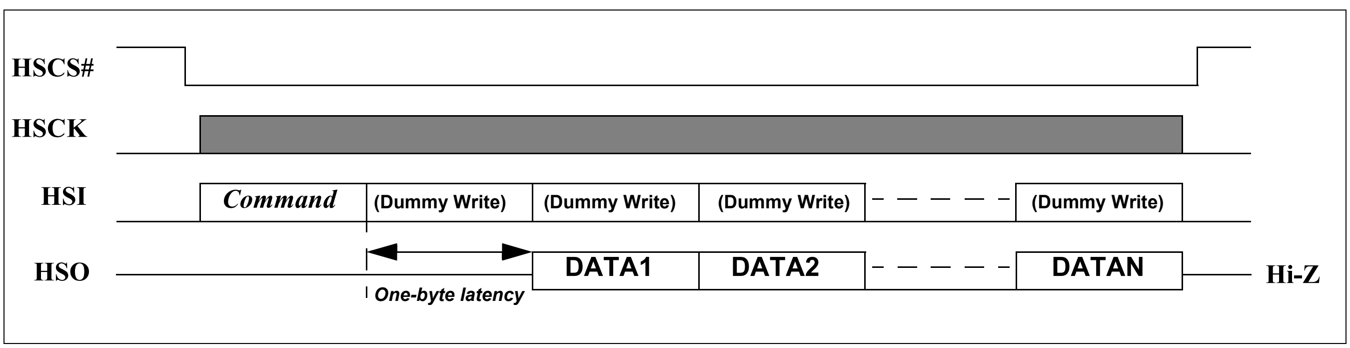


Figure 21-6: SPI Host "Read DATA" Command Transfer Cycle

Note

1. For the "Read DATA" transfer cycle, the S1D13515/S2D13515 requires time to internally read the first byte of data and
   does not output the first byte until a one-byte latency after the Command byte.
2. The maximum frequency / minimum period of HSCK is determined by the following equation:

   (8 HSCK cycles) >= [((W + 5) SysClk cycles) + (7 ClkSpi cycles)]

   W = worst VBUS access period
   SysClk = System Clock

ClkSpi = Host SPI Interface Clock

For access to internal registers, W = 4.
For access to internal RAMs, and DMA Controller is not performing burst access to the internal RAM block being accessed, W = 3.
For access to internal RAMs, and DMA Controller is performing burst access to the internal RAM block being accessed, W = 17.

21.11 I2C Host Interface

The following table shows the pins used for the I2C Host interface:

Table 21-11: I2C Host Interface Signals

S1D13515/S2D13515 Pin	I2C Host
WAIT#	HSDA
BS#	HSCL
AB5	SPICLKSEL

The I2C host module requires a valid clock selection before the interface can operate. The I2C host module clock selection is determined by a combination of I2CCLKEN (AB5) pin and REG[0063h] bits 2 and 0.

The host access method for the I2C is similar to that of the SPI host. The main difference between the SPI and I2C is the presence of the HSCS# (chip-select) signal in the SPI. I2C does not have a chip-select and the slave device (S1D13515/S2D13515) is selected by the Slave Address in the I2C packet.

The 7-bit Slave Address for the S1D13515/S2D13515 in I2C host interface mode is defined by the DB6-DB0 pins. The DB6-DB0 pins should be tied to the desired 7-bit Slave Address value. The following table shows the slave addresses.

Table 21-12: I2C Slave Addresses

Slave Address	Note
0000_000b	reserved
0000_001b	reserved
0000_010b	reserved
0000_011b	reserved
0000_1xxb	reserved
0001_000b ~ 1110_111b	allowed
1111_0xxb	reserved
1111_1xxb	reserved

Note

1. Any change to the I2C Slave Address requires a hardware RESET#.
2. Reserved I2C slave addresses are not supported. Refer to the latest *I2C-bus specification and user manual, UM10204,* for details.

I2C host transfer cycles are similar to SPI host transfer cycles in that the Command bytes are defined. The following figures show the transfer cycles for the I2C host interface Commands:

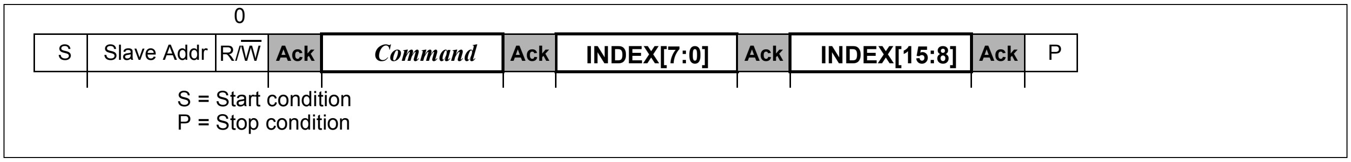


Figure 21-7: I2C Host "Write INDEX" Command Transfer Cycle

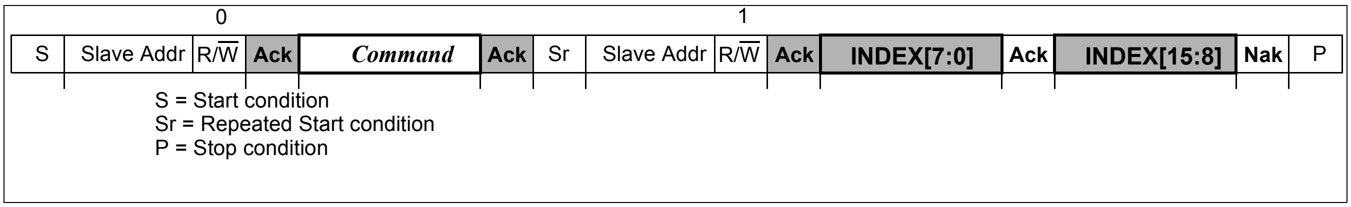


Figure 21-8: I2C Host "Read INDEX" Command Transfer Cycle

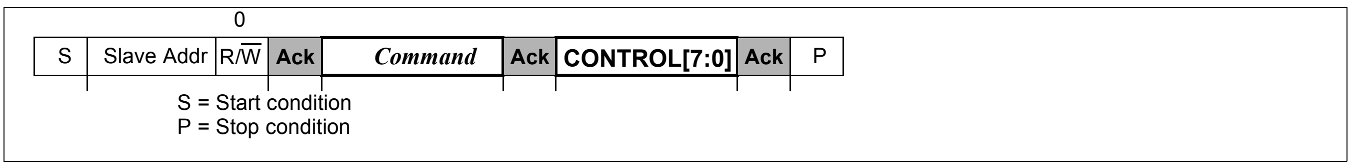


Figure 21-9: I2C Host "Write CONTROL" Command Transfer Cycle

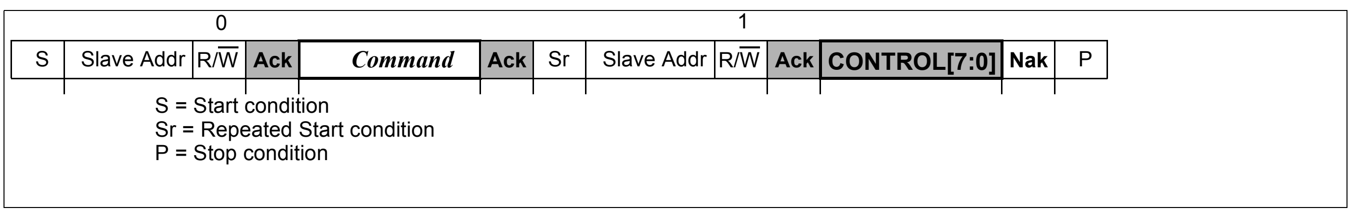


Figure 21-10: I2C Host "Read CONTROL" Command Transfer Cycle

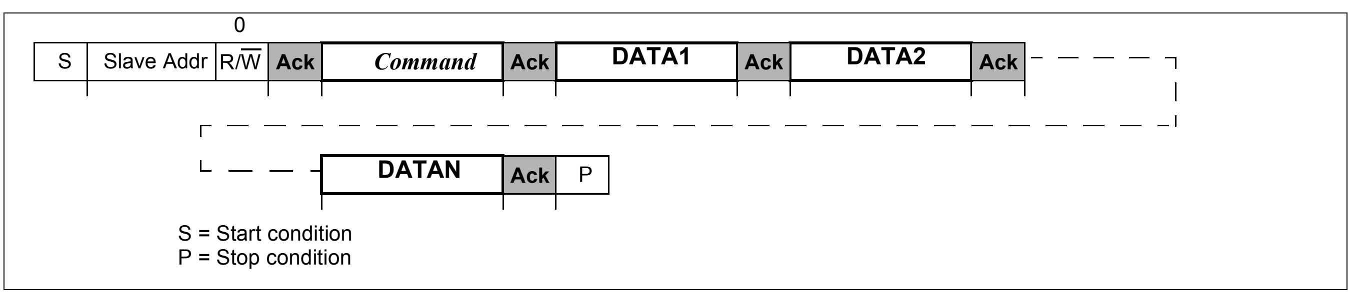


Figure 21-11: I2C Host "Write DATA" Command Transfer Cycle

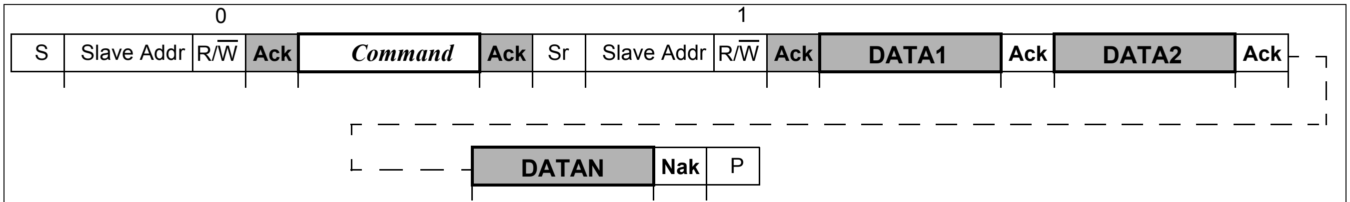


Figure 21-12: I2C Host "Read DATA" Command Transfer Cycle

Note

1. In the "Read DATA" transfer cycle, for each DATA byte, the S1D13515/S2D13515 will hold SCL low until it has internally read the requested data byte.
2. For the "Write DATA" transfer cycle, the maximum frequency / minimum period of SCL is determined by the following equation:

   (8 SCL cycles) >= [((W + 5) SysClk cycles) + (17 ClkI2c cycles)]

   W = worst VBUS access period
   SysClk = System Clock
   ClkI2c = Host I2C Interface Clock

   For access to internal registers, W = 4.
   For access to internal RAMs, and DMA Controller is not performing burst access to the internal RAM block being accessed, W = 3.
   For access to internal RAMs, and DMA Controller is performing burst access to the internal RAM block being accessed, W = 17.

21.12 Host Interface Access Methods

The S1D13515/S2D13515 has an internal 32-bit address space. The external SDRAM, internal registers, internal RAMs, and Serial Flash are all mapped into this single internal 32-bit address space. The internal C33PE processor and the Host have full access to this 32-bit address space. The methods for the Host to access this internal 32-bit address space is described in this section.

The Host interfaces supported by the S1D13515/S2D13515 can be divided into two modes: Direct or Indirect. Direct mode is only for Parallel (not SPI or I2C) Hosts and assumes that the AB20-AB0 address lines are used to interface the Host to the S1D13515/S2D13515. Indirect mode is for both Parallel and Serial Hosts. The Indirect mode for Parallel Hosts is used if there are limitations in the system which restrict the connection of more than 2 address lines between the Host and the S1D13515/S2D13515.

21.12.1 Direct Mode

For Direct mode Parallel Hosts, there are two address spaces defined for the Host Interface which is selected by the M/R# input pin: Memory (M/R# = 1) and Register (M/R# = 0) space.

When the Host accesses Memory space, it is directly accessing a window/page of the 32-bit internal address space. The Host Interface only has AB20-AB0 pins available for address lines and they form the lower address bits for the direct access into the internal 32-bit address space. The upper address bits are provided by the Internal Memory Space Upper Address Register which is accessed in the Register space.

The Register space is a 64Kbyte address space and only uses address lines AB15-AB0. There are 3 groups of registers in the Register space: (see Figure 21-13:)

- Group1 - Registers which are only accessible and used by the Host Interface.
- Group2 - Internal Core Registers which are accessible by both the Host and the internal C33PE processor and arbitrated for simultaneous access by both.
- Group3 - Internal Core Registers which are accessible by both the Host and the internal C33PE processor but only one can access these registers at a time.

There are also two types of registers: Synchronous and Asynchronous. Synchronous registers require the System Clock in the S1D13515/S2D13515 to be running. Asynchronous registers do not need the System Clock to be running in order to access them.

The Group3 registers are mainly clock control registers which can be asynchronously accessed from the Host. REG[0084h] bit 0 (asynchronously accessible only by the Host) is used to select the control of the Group3 registers between the Host and C33PE. When this bit is 0, the internal C33PE processor has access. When this bit is 1, the Host has asynchronous access.

The following diagram shows the Register space of the Host Interface and its relation to the internal 32-bit address space of the S1D13515/S2D13515:

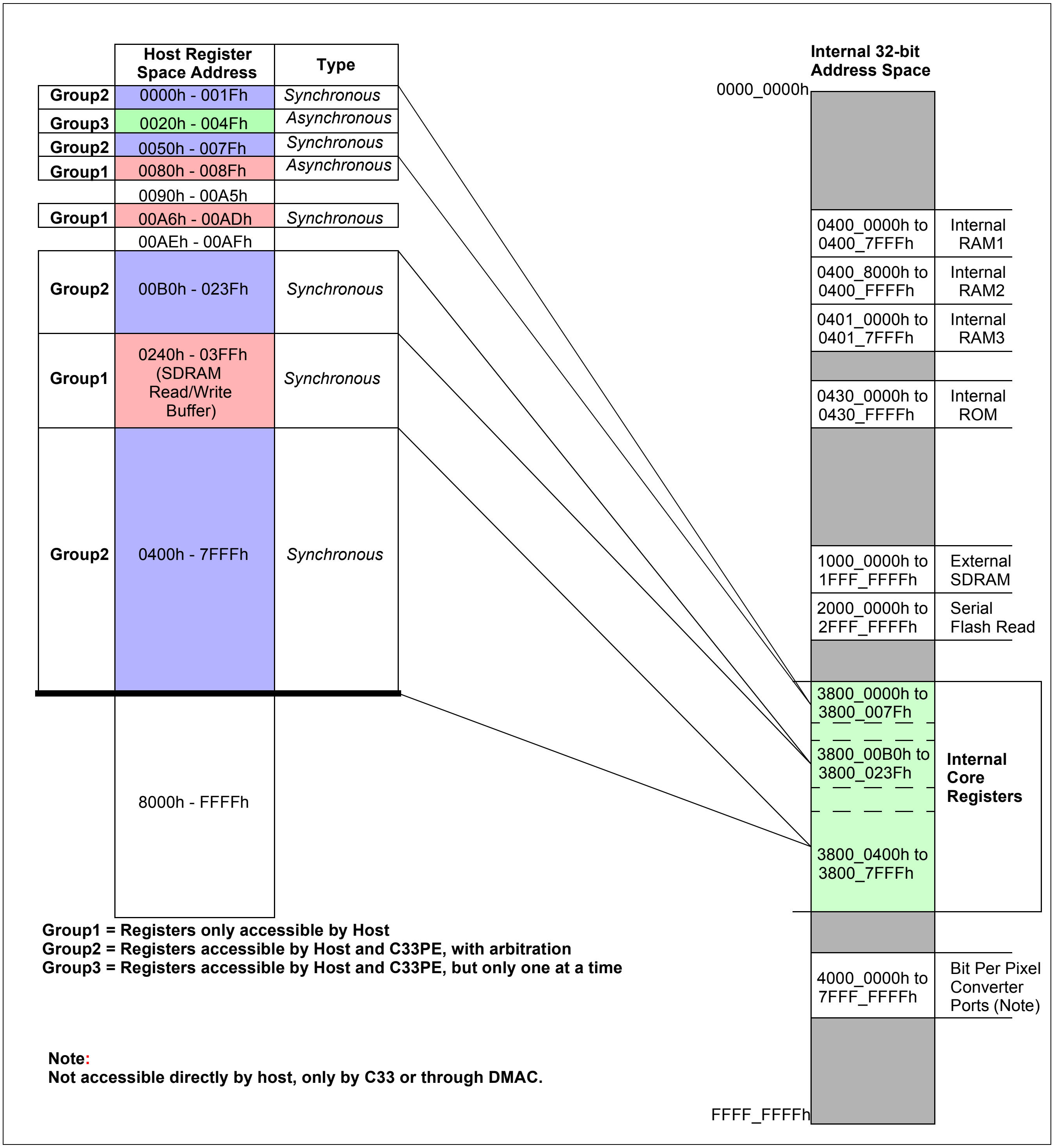


Figure 21-13: Host Interface Register Space

The Register space of the Host Interface allows the Host to direct access the Internal Core Registers of the S1D13515/S2D13515. There are also registers in the Register space to allow the Host to indirectly access the 32-bit internal address space as opposed to access through the Memory space.

The Internal Memory Space Upper Address Register (MUADDR[31:16]) in REG[0080h]-REG[0081h] is for programming the upper address bits for Memory space access. During a Memory space access, the internal 32-bit address is formed by concatenating AB20-AB0 with the MUADDR register bits. There is also an Internal Memory Space Upper Address Mask Register (MUMASK[20:16]) in REG[0082h] which is used to specify how the internal 32-bit address for Memory space access is formed. Bits 31-21 of the 32-bit address uses MUADDR[31:21] and bits 15-0 uses AB15-AB0. Bits 20-16 of the 32-bit address is determined by MUMASK[20:16]. If MUMASK[x] is 0, bit x of the 32-bit address uses ABxx. If MUMASK[x] is 1, bit x of the 32-bit address uses MUADDR[x].

The Internal Memory Space Read/Write Address Register in REG[00A8h]-REG[00ABh] and Internal Memory Space Read/Write Data port Register in REG[00ACh]-REG[00ADh] are used to indirectly access the 32-bit internal address space. To access a location in the internal 32-bit address space, the Host writes the desired 32-bit address location into the REG[00A8h]-REG[00ABh] and then performs the data access by reading/writing REG[00ACh]-REG[00ADh]. Therefore, for Direct mode parallel hosts, the internal 32-bit address space can be accessed either through the Memory space or the Register space (by way of REG[00A8h]-REG[00ADh]). The Internal Memory Space Read/Write Address Register (REG[00A8h]-REG[00ABh]) can be programmed to not increment or to increment whenever the Internal Memory Space Read/Write Data Port Register (REG[00ACh]-REG[00ADh]) is accessed by programming bit 0 of the Internal Memory Space Read/Write Control Register (REG[00A6h]).

21.12.2 Indirect Mode

For Indirect mode hosts (Parallel or Serial), there is a limited number of pins for the host interface connection. For Parallel Indirect hosts, there are only two address lines available. For Serial hosts, there are no address lines. In Indirect mode, only Register space is available to the Host and Memory space is not directly available because there is no M/R# signal. The Host accesses the internal 32-bit address space through the Internal Memory Space Read/Write Address and Data Port Registers. Additionally, SDRAM can be accessed using the SDRAM Read/Write Buffer.

For Indirect mode hosts, there are 3 registers defined for the indirect access: INDEX[15:0], DATA, and CONTROL[7:0]. The INDEX[15:0] register is the 16-bit Register space address the Host wants to access. To access a Register space location, the Host first writes the 16-bit Register space address into the INDEX register and then reads/write the DATA register to do the actually access of the Register space location. The CONTROL[7:0] is a control register which only has bit 0 defined. Bits 7-1 are reserved. Bit 0 specifies whether or not the INDEX[15:0] register is incremented on each access to the DATA register.

Indirect Parallel 8-bit Hosts

For Indirect Parallel 8-bit hosts, the AB1-AB0 address lines are used as follows:

Table 21-13: Indirect Parallel 8-bit Host Interface

AB[1:0]	Name	Description
00b	INDEX[7:0]	Lower 8 bits of INDEX[15:0] register.
01b	INDEX[15:8]	Upper 8 bits of INDEX[15:0] register.
10b	DATA[7:0]	Port to access Register space.
11b	CONTROL[7:0]	CONTROL register. Bit 0 = INDEX auto-increment bit. 0 = no increment 1 = increment Bit 7-1 = Reserved.

All accesses are 8-bit accesses and INDEX[15:0] increments by 1 (if CONTROL[0] = 1) whenever DATA[7:0] is accessed.

Indirect Parallel 16-bit Hosts

For Indirect Parallel 16-bit hosts, the AB2-AB1 address lines are used as follows:

Table 21-14: Indirect Parallel 16-bit Host Interface

AB[2:1]	Name	Description
00b	INDEX[15:0]	INDEX[15:0] register.
01b	DATA[15:0]	Port to access Register space.
10b	CONTROL[7:0]	CONTROL register. Bit 0 = INDEX auto-increment bit. 0 = no increment 1 = increment Bit 7-1 = Reserved.
11b	Reserved	Reserved.

Although physically on the Host Interface bus all accesses are 16-bit accesses, there is a method of specifying 8-bit DATA port access using INDEX[15]. If INDEX[15] is 0, the access to DATA[15:0] is assumed to be 16-bit. If INDEX[15] is 1, it specifies that the access to DATA[15:0] is 8-bit and INDEX[0] specifies odd or even byte. If INDEX[0] = 0, the 8-bit data is in DATA[7:0]. If INDEX[0] = 1, the 8-bit data is in DATA[15:8].

Indirect Serial Hosts

For Indirect Serial hosts, there are no address lines and a Command byte is used to access the INDEX, DATA, and CONTROL registers. The access method for Indirect Serial Hosts is similar to that of the Indirect Parallel 8-bit Hosts. See Section 21.10 for more details.

21.13 Initialization Examples

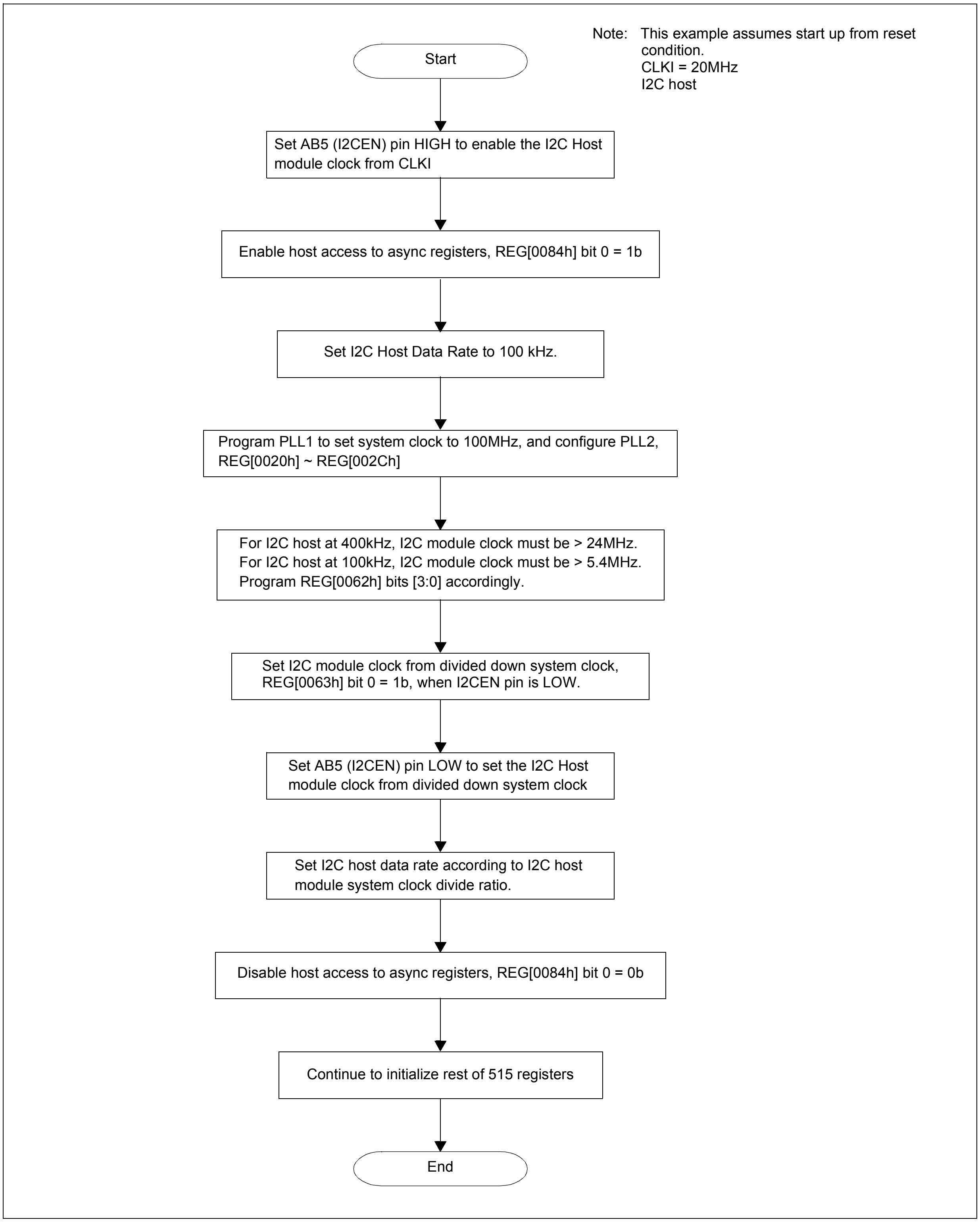


Figure 21-14: I2C Initialization Example

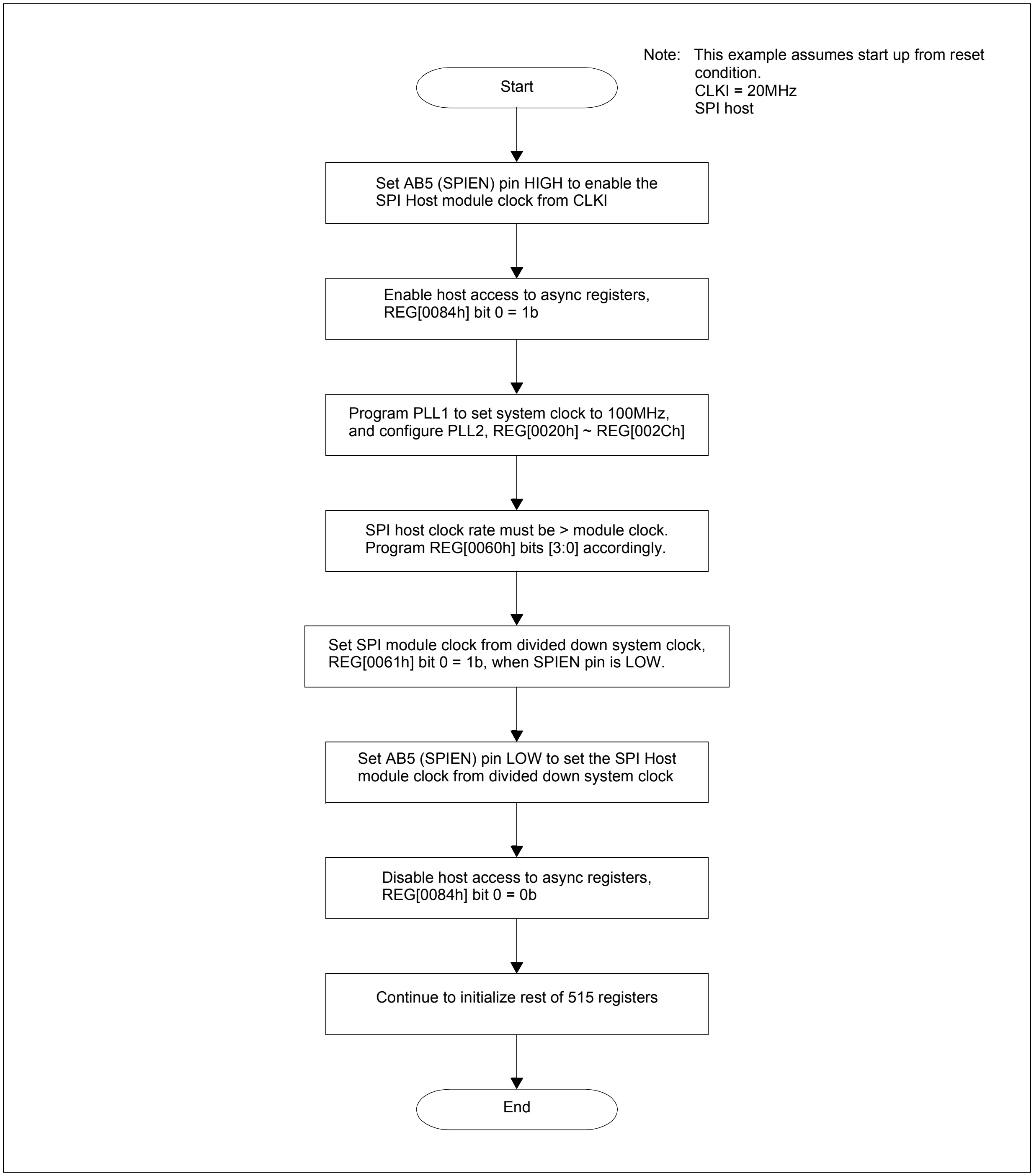


Figure 21-15: SPI Initialization Example

Chapter 22 Camera Interface Subsystem

22.1 Overview

The S1D13515/S2D13515 can support up to two camera input interfaces. It has two instances (Camera1 and Camera2) of a Camera Interface Core, each connected to its own block for writing RGB pixel data to SDRAM (Camera1 Writer and Camera2 Writer). The following shows a block diagram of the Camera Interface Subsystem:

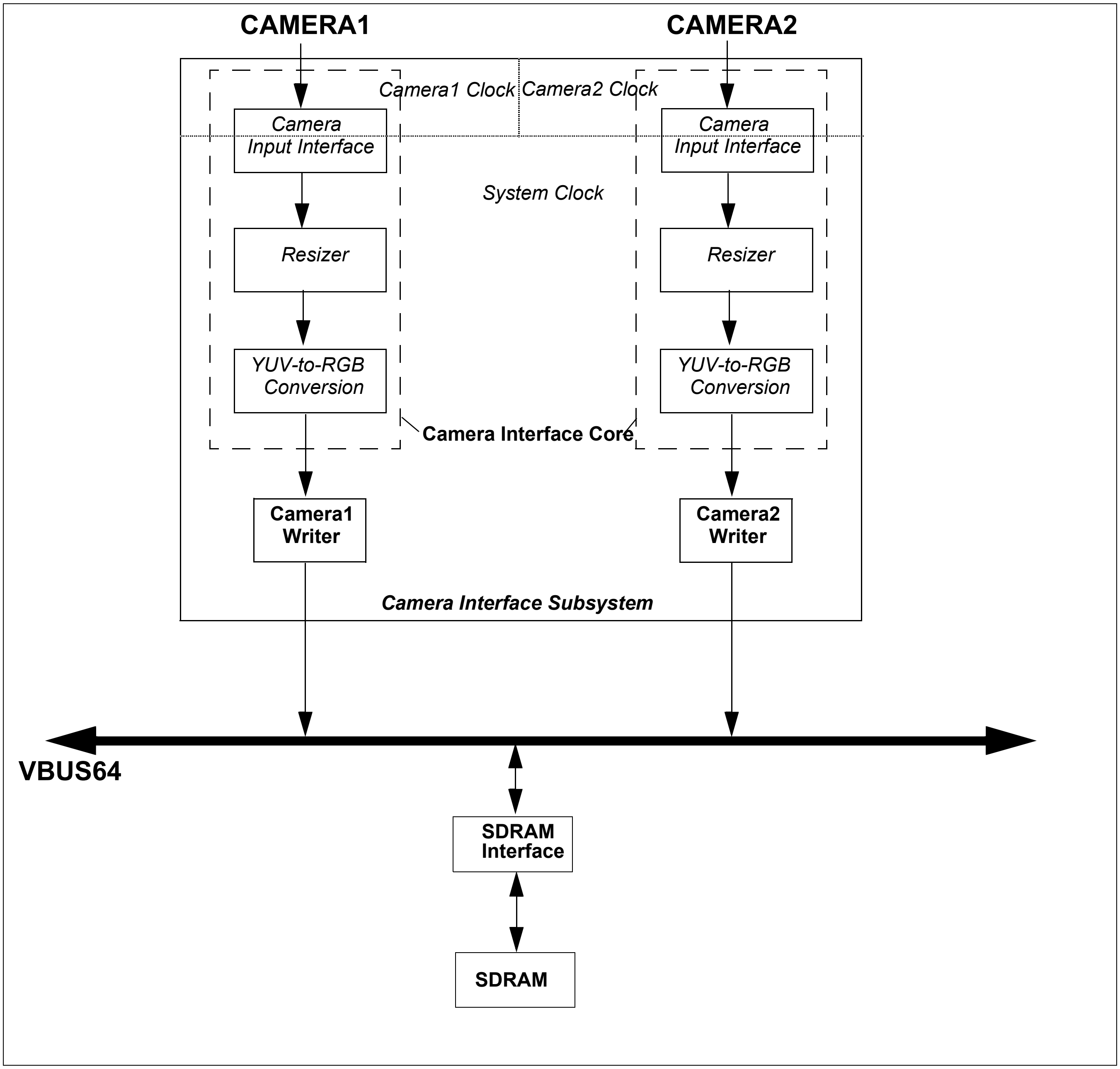


Figure 22-1: Camera Interface Subsystem Block Diagram

The Camera Interface Subsystem supports two types of interfaces: 8-bit camera input interfaces with YUV data and RGB (up to 24-bit) streaming input interface. REG[0D06h]/REG[0D46h] bits 2-1 selects 8-bit YUV (= 00b)or 24-bit RGB streaming mode (= 10b) for Camera1/Camera2. REG[0D00h]/REG[0D40h] bit 0 enables/disables the Camera Interface Core for Camera1/Camera2.

The Camera Input Interface subblock handles the raw camera input timing and also synchronizes data from the camera clock to the system clock. The Resizer subblock has logic to crop and downscale the input image. The YUV-to-RGB Converter subblock performs the task of converting YUV input data to RGB format for writing to SDRAM. For RGB streaming input, there is a bypass bit in the YUV-to-RGB Converter to turn off the YUV-to-RGB conversion. The Camera Writer block writes the camera input pixel data to SDRAM.

22.2 IO Pins for Camera Interfaces

22.2.1 8-bit Camera Interface

For 8-bit camera interface, Camera1 has dedicated IO pins (CM1*) and Camera2 is shared with LCD1 panel interface pins (FP1*). REG[4000h] bit 3 selects Camera2 (= 1b) or LCD1 (= 0b) for the FP1* pins.

22.2.2 RGB Streaming Input Interface

Camera1 RGB streaming input interface is only available if the Host interface is configured to be SPI (2 Stream) serial interface (CNF1=0, CNF2=1, TEA#=0, BDIP#=0, BURST#=1, AB3=1, AB4=0). The signals for the RGB streaming interface are mapped to the unused pins of the Host Interface when the SPI serial interface is selected. Internally, the pixel data input to the Camera Interface Subsystem is RGB888 (24-bit), however, due to pin limitations, only 18 bits are connected externally (RGB666). For padding the lower bits of each color of the RGB888 internal pixel data, the MSB of the input pixel data of each color (RGB666) is used.

For Camera2, the RGB streaming input interface is shared with LCD1 panel interface pins (FP1*) and REG[4000h] bit 3 selects Camera2 (= 1b) or LCD1 (= 0b). Internally, the pixel data input to the Camera Interface Subsystem is RGB888 (24-bit), however, due to pin limitations, only 15 or 18 bits are connected externally (RGB555 or RGB666) depending on the configuration of the interface pins for the LCD2 interface. For padding the lower bits of each color of the RGB888 internal pixel data, the MSB of the input pixel data of each color (RGB555 or RGB666) is used.

22.3 Camera Input Interface

The Camera Input Interface subblock handles the interface with the external pins and synchronizes the signals between the asynchronous camera clock and internal system clock which runs the rest of the Camera Interface Core subblocks. The following are programmable registers for the Camera Input Interface subblock:

- REG[0D00h]/REG[0D40h] enables/disables Camera1/Camera2.
- REG[0D02h]/REG[0D42h] configures the camera clock Camera1/Camera2
- REG[0D04h]/REG[0D44h] configures the polarity of the input interface signals for Camera1/Camera2
- REG[0D06h]/REG[0D46h] configures the input pixel data format for Camera1/Camera2:
  - Bit 7 enables/disables the ITU-R BT.656 mode for 8-bit YUV interface.
  - Bit 5 enables/disables UV offset of -128 for 8-bit YUV interface.
  - Bits 4-3 select the arrangement of the Y, U, and V components of the YUV input data.
  - Bits 2-1 selects 8-bit YUV 4:2:2 or 24-bit RGB 8:8:8 input.
- REG[0D07h]/REG[0D47h] bit 0 is enable/disable the use of the DE (data enable) pin of 24-bit RGB interface for Camera1/Camera2.
- REG[0D08h]/REG[0D48h] is the Input Frame Control Register for Camera1/Camera2:
  - Bit 6 enables/disables the capturing of frames. It should be set to 1 to start the camera capturing.
  - Bit 5 selects the type of event to capture for the Frame Event flag in REG[0D0Eh]/REG[0D4Eh] bit 5: Frame Start of Frame End.
  - Bit 4 enables/disables the Frame Event flag.
  - Bit 3 selects the trigger signal for capturing the Frame Event flag: VSYNC or camera stop.
- REG[0D09h]/REG[0D49h] is the Input Flag Clear Register (write-only) for clearing the Frame Event status bits in REG[0D0Eh]/REG[0D4Eh].
- REG[0D0Ah]-REG[0D0Bh] / REG[0D4Ah]-REG[0D4Bh] specifies the Horizontal Size of the input image for Camera1/Camera2.
- REG[0D0Ch]-REG[0D0Dh] / REG[0D4Ch]-REG[0D4Dh] specifies the Vertical Size of the input image for Camera1/Camera2.
- REG[0D0Eh]/REG[0D4Eh] is the Status Register for Camera1/Camera2 (read-only):
  - Bit 5 is the Frame Event flag and is configured by REG[0D08h]/REG[0D48h] bits 5-3.
  - Bit 4 is the Effective Capture status bit and indicates the effective frame capture status according to the Frame Sampling Select bits in REG[0D08h]/REG[0D48h].
  - Bit 3 is the Effective Frame status bit. It is the same as bit 4 but is 1 only when there is a valid frame. Bit 4 is 1 even on invalid frames.
  - Bit 2 is the raw VSYNC status.
- REG[0D30h]/REG[0D70h] is the Video Mode Register for Camera1/Camera2. The following are the three modes:

    - Progressive
    - Interlaced2: HSYNC and Field inputs are used.
    - Interlaced3: HSYNC and VSYNC are used.
- REG[0D32h]-REG[0D33h] / REG[0D72h]-REG[0D73h] specify the Odd field signal timing offset for Camera1/Camera2 when Interlaced2 or Interlaced3 mode is selected.
- REG[0D34h]-REG[0D35h] / REG[0D74h]-REG[0D75h] specify the Even field signal timing offset for Camera1/Camera2 when Interlaced2 or Interlaced3 mode is selected.

22.4 Resizer

The Resizer subblock handles cropping and down-scaling of the input camera input before it goes to the YUV-to-RGB converter. The following are programmable registers for the Resizer subblock:

- REG[0D10h]-REG[0D11h] / REG[0D50h]-REG[0D51h] specify the X Start Position for cropping.
- REG[0D12h]-REG[0D13h] / REG[0D52h]-REG[0D53h] specify the Y Start Position for cropping.
- REG[0D14h]-REG[0D15h] / REG[0D54h]-REG[0D55h] specify the X End Position for cropping.
- REG[0D16h]-REG[0D17h] / REG[0D56h]-REG[0D57h] specify the Y End Position for cropping.
- REG[0D18h]/REG[0D58h] specifies the Horizontal Scaling Rate.
- REG[0D19h]/REG[0D59h] specifies the Vertical Scaling Rate.
- REG[0D1Ah]/REG[0D5Ah] specifies the Resizer Scaling Type.

22.5 YUV-to-RGB Converter

The YUV-to-RGB Converter subblock handles the conversion of YUV input pixel data to RGB. The following are programmable registers for the YUV-to-RGB Converter subblock:

- REG[0D1Eh]/REG[0D5Eh] is the YUV-to-RGB Conversion (YRC) Control Register 0 for Camera1/Camera2:
  - Bits 6-5 specify the output RGB format: RGB332, RGB565, or RGB888.
  - Bit 4 selects the YUV Data Type: YUV or YCbCr.
  - Bits 3-1 select the YUV Transfer Mode.
  - Bit 0 is the YRC Bypass Enable to bypass the YUV-to-RGB conversion for the 24-bit RGB input streaming data.
- REG[0D1Fh]/REG[0D5Fh] is the YUV-to-RGB Conversion (YRC) Control Register 1 for Camera1/Camera2:
  - Bits 1-0 specify whether or not the U and V components of the YUV data is fixed before conversion to RGB. The fixed values are programmed in REG[0D20h]-REG[0D21h] / REG[0D60h]-REG[0D61h].
- REG[0D20h]/REG[0D60h] specifies the U fixed data.
- REG[0D21h]/REG[0D61h] specifies the V fixed data.
- REG[0D24h]-REG[0D25h] / REG[0D64h]-REG[0D65h] specify the X Size (width) of the input image to the YRC subblock.
- REG[0D26h]-REG[0D27h] / REG[0D66h]-REG[0D67h] specify the Y Size (height) of the input image to the YRC subblock.

22.6 Camera Writer

Each of the Camera Interface Core for Camera1 and Camera2 and connected to a corresponding Camera Writer block. The Camera Writer block receives and buffers RGB pixel data from the Camera Interface Core and performs burst writes frame buffers in the external SDRAM through the VBUS64 bus.

The following are programmable registers for the Camera Writer subblock of Camera1:

- REG[09E0h]-REG[09E3h] specifies the destination base address for Frame Buffer 0.
- REG[09E4h]-REG[09E7h] specifies the destination base address for Frame Buffer 1.
- REG[09F0h]-REG[09F1h] specifies the width (in pixels) of Camera1's frame buffer.
- REG[09F2h]-REG[09F3h] specifies the height (in pixels) of Camera1's frame buffer.
- REG[09F4h]-REG[09F5h] specifies the virtual width (in pixels) of Camera1's frame buffer.
- REG[09F6h] is the Camera1 Writer Control Register:
    - Bit 7 is the Camera1 Double Buffer Method Select bit.
    - Bits 3-2 specify the RGB format of the pixel data: RGB332, RGB565, or RGB888.
    - Bit 0 is Camera1 Flip Around X Axis (vertical flip) bit.

The following are programmable registers for the Camera Writer subblock of Camera2:

- REG[09E8h]-REG[09EBh] specifies the destination base address for Frame Buffer 0.
- REG[09ECh]-REG[09EFh] specifies the destination base address for Frame Buffer 1.
- REG[09F8h]-REG[09F9h] specifies the width (in pixels) of Camera2's frame buffer.
- REG[09FAh]-REG[09FBh] specifies the height (in pixels) of Camera2's frame buffer.
- REG[09FCh]-REG[09FDh] specifies the virtual width (in pixels) of Camera2's frame buffer.
- REG[09E6h] is the Camera2 Writer Control Register:
    - Bit 7 is the Camera2 Double Buffer Method Select bit.
    - Bits 3-2 specify the RGB format of the pixel data: RGB332, RGB565, or RGB888.
    - Bit 0 is Camera2 Flip Around X Axis (vertical flip) bit.

Chapter 23 Keypad Interface

Depending on the configuration of IO pins, the S1D13515/S2D13515 has keypad drive/detect logic which can support up to a 5x5 matrix. The keypad drive (column) / detect (row) pins can be mapped to either the FP1IOx pins or the Host Interface pins. The key scanning clock frequency is programmable. Each of the five detect (row) inputs can be programmed to be filtered or unfiltered.

23.1 Keypad Pin Mapping

The keypad interface can be mapped to either the FP1IOx pins or the Host Interface pins through the GPIO[15:8]/Keypad Pin Mapping Select bit (REG[0186h] bit 5).

Note
GPIO7 is not available when the Keypad Interface is configured to use the FP1IO pins, REG[0186h] bit 5 = 1b.

When REG[0186h] bit 5 is 0b, the keypad interface is mapped to the Host Interface pins. The keypad interface (5x5 matrix) is available in the Host Interface pins for Parallel Indirect, Serial (SPI, I2C), and Marvell PXA3xx 16-bit Direct host interfaces only. The keypad interface is not available for all other Parallel Direct interfaces.

See Table 5-13 “Host Interface Pin Mapping 1,” on page 35 through Table 5-16: “Host Interface Pin Mapping 4,” on page 39 for more details.

When REG[0186h] bit 5 is 1b, the keypad interface is mapped to the FP1IOx pins. When mapped to the FP1IOx pins, the keypad interface is only available when the FP1IOx pins are programmed for Camera2 interface (REG[4000h] bit 3 = 1b) and the camera interface type is 8-bit (REG[0D46h] bit 2 = 0b). See Table 5-17: “FP1IO Pin Mapping Summary (LCD1 / Camera2),” on page 40 for more details on the actual pin mappings. If LCD2 does not use FP1IOx pins (see Note 2 in Table 5-17:, “FP1IO Pin Mapping Summary (LCD1 / Camera2)”), a 5x5 keypad matrix is available. If LCD2 uses FP1IOx pins, only a 3x3 keypad matrix is available.

23.2 Scanning Operation

The keypad scanning logic works with five drive and five detect active-low signals. The logic is clocked by the Keypad Clock which is a divide-down from the input clock (OSCI or CLKI). The frequency of the Keypad Clock is programmed through REG[01D4h] ~ REG[01D5h].

The scanning logic works by sequentially driving each of the 5 drive lines (KPCx pins) low at a time and reading the five detect inputs (KPRx pins). If input filter is not enabled (REG[01C0h] bit 1 = 0b), each drive output is driven low for four Keypad Clock cycles each and the five detect inputs are checked at the end of the 4th Keypad Clock cycle when a drive output is low. When there is no keys pressed, the detect inputs are normally high. If a key is pressed, the detect input (row) will go low when a drive output (column) is driven low.

There are 25 flip-flops used to detect each key in the 5x5 matrix (Keypad Interrupt Status bits in REG[01D0h] through REG[01D3h]). Each of the 25 flip-flops is set to 1b on the rising edge of its input clock signal and is cleared by writing a 1b to the corresponding Keypad Interrupt Status bit. The input clock signal is the XOR of the corresponding polarity bit (REG[01C8h] ~ REG[01CBh]) and the input detect signal. If the polarity bit is 0b, key release (rising edge) is detected. If the polarity bit is 1b, key press (falling edge) is detected.

23.3 Input Glitch Filter

If input glitch filter is enabled (REG[01C0h] bit 1 = 1b), each drive output is driven low for four Sample Clocks instead of four Keypad Clocks. The Sample Clock is a divided-down of the Keypad Clock and programmed through REG[01CCh] ~ REG[01CEh]. The five detect inputs are checked at the end of the fourth Sample Clock.

When input filter is enabled, the clock input to each of the 25 Keypad Interrupt Status flip-flops is a filtered version of the detect input. The output of the filtered signal will only change state if two consecutive samples of the input signal are the same level and opposite to the current filtered output logic level.

23.4 General-Purpose Input Function

Each of the five detect input (KPRx row) pins can be programmed to be used as a general-purpose input by programming the corresponding bit in REG[01D6h] to 1b. If the bit is 1b, the corresponding KPRx input is disassociated with the drive output logic and functions strictly as a general-purpose input pin. This provides extra general-purpose input functionality if not all of the 5x5 keypad matrix is used and also provides glitch-filtered general-purpose input functionality.

23.5 Interrupts

Each of the 25 Keypad Interrupt Status bits has a corresponding Keypad Interrupt Enable bit in REG[01C4h] ~ REG[01C7h]. Each status and interrupt enable bit is ANDed and the 25 ANDed signals are OR'ed together to generate the Keypad Interrupt Status bit which goes the Interrupt Controller and can be read from REG[0A02h] bit 4. To enable keypad interrupts to the Host, REG[0A08h] bit 4 should be set to 1b. To enable keypad interrupts to the C33, REG[0A10h] bit 4 should be set to 1b.

23.6 Keypad Operation Flow

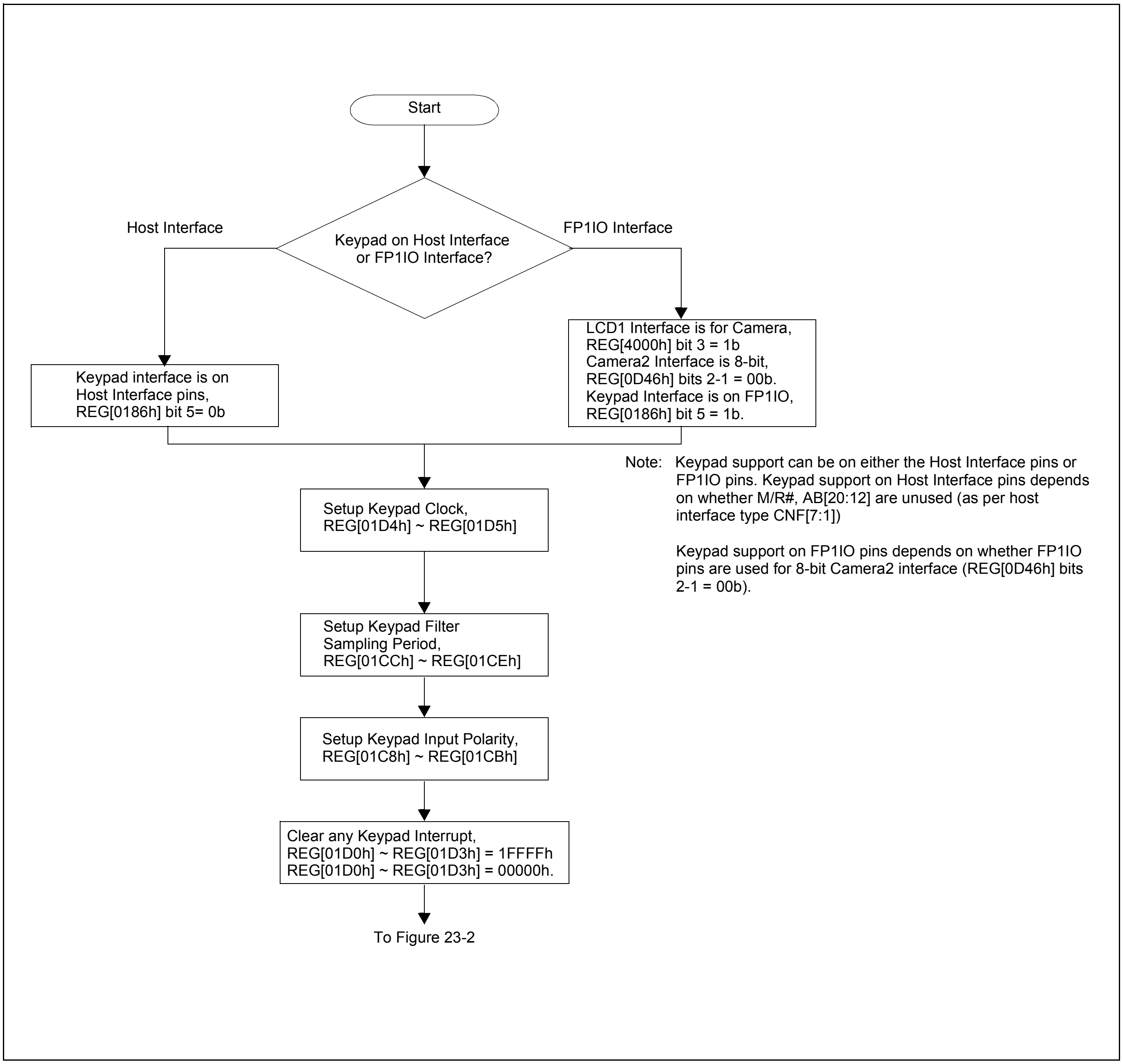


Figure 23-1: Typical Keypad Operation Flow

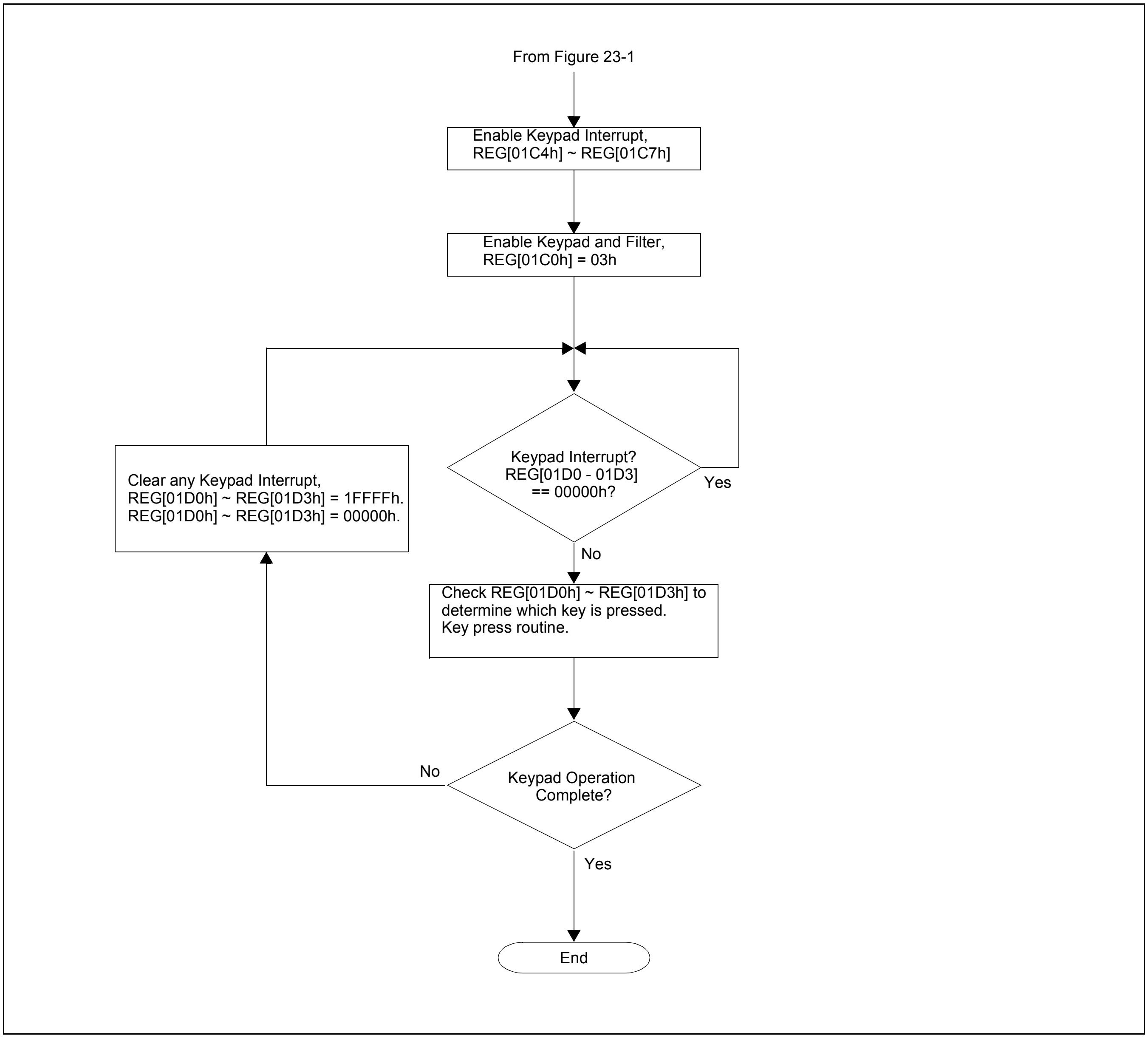


Figure 23-2: Typical Keypad Operation Flow (Continued)

Chapter 24 Timers

24.1 Watchdog Timer

The S1D13515/S2D13515 has watchdog timer logic which can be used to reset the chip in case there is stray code which hangs the software. The watchdog timer logic is disabled by default and should be enabled by software during power-up initialization. The watchdog timer is enabled by writing a 1b to bit 2 of REG[0A84h]. When the watchdog timer is enabled, an up-counter is started. If the counter reaches a programmed threshold value (as programmed in REG[0A86/7h]), a watchdog interrupt or chip reset is generated (as specified by the Watchdog Time-out Action, bit 3 of REG[0A84h]). The up-counter can periodically be reset to 0b by the software (to prevent watchdog time-out) by writing 2371h to the Watchdog Timer Clear Register (REG[0A8Ch] ~ (REG[0A8Dh]).

If the Watchdog Time-out Action bit is set to 0b, the Watchdog Interrupt Status bit (REG[0A00h] bit 2) is set to 1b when a watchdog time-out occurs. To enable the watchdog interrupt to the Host, set REG[0A06h] bit 2 to 1b. On the C33 side, the watchdog interrupt can generate the IRQ2 interrupt by setting REG[0A42h] bit 2 to 1b. To clear the watchdog interrupt, write a 1b to the Watchdog Interrupt Status bit (REG[0A00h] bit 2).

If the Watchdog Time-out bit is set to 1b, a chip reset will occur when there is a watchdog time-out.

24.2 Timer 0

Timer 0 is a general purpose timer which is used by the C33 ROM Monitor and is not be available as a general purpose timer when the C33 is used.

Timer 0 is enabled by writing a 1b to bit 0 of REG[0A84h]. When the timer is enabled, an up-counter is started. If the counter reaches a programmed threshold value (as programmed in REG[0A88/9h]), an interrupt is generated. The Timer 0 Interrupt Status bit (REG[0A02h] bit 2) is set to 1b when a time-out occurs.To clear the timer interrupt, write a 1b to the Timer 0 Interrupt Status bit (REG[0A02h] bit 2)

24.3 Timer 1

Timer 1 is a general purpose timer which is used by the C33 ROM Monitor. This timer may be used as a general purpose timer after the C33 ROM boot process has completed.

Timer 1 is enabled by writing a 1b to bit 1 of REG[0A84h]. When the timer is enabled, an up-counter is started. If the counter reaches a programmed threshold value (as programmed in REG[0A8Ah]), an interrupt is generated. The Timer 1 Interrupt Status bit (REG[0A02h] bit 3) is set to 1b when a time-out occurs.To clear the timer interrupt, write a 1b to the Timer 0 Interrupt Status bit (REG[0A02h] bit 3)

24.4 Timer Operation Flow

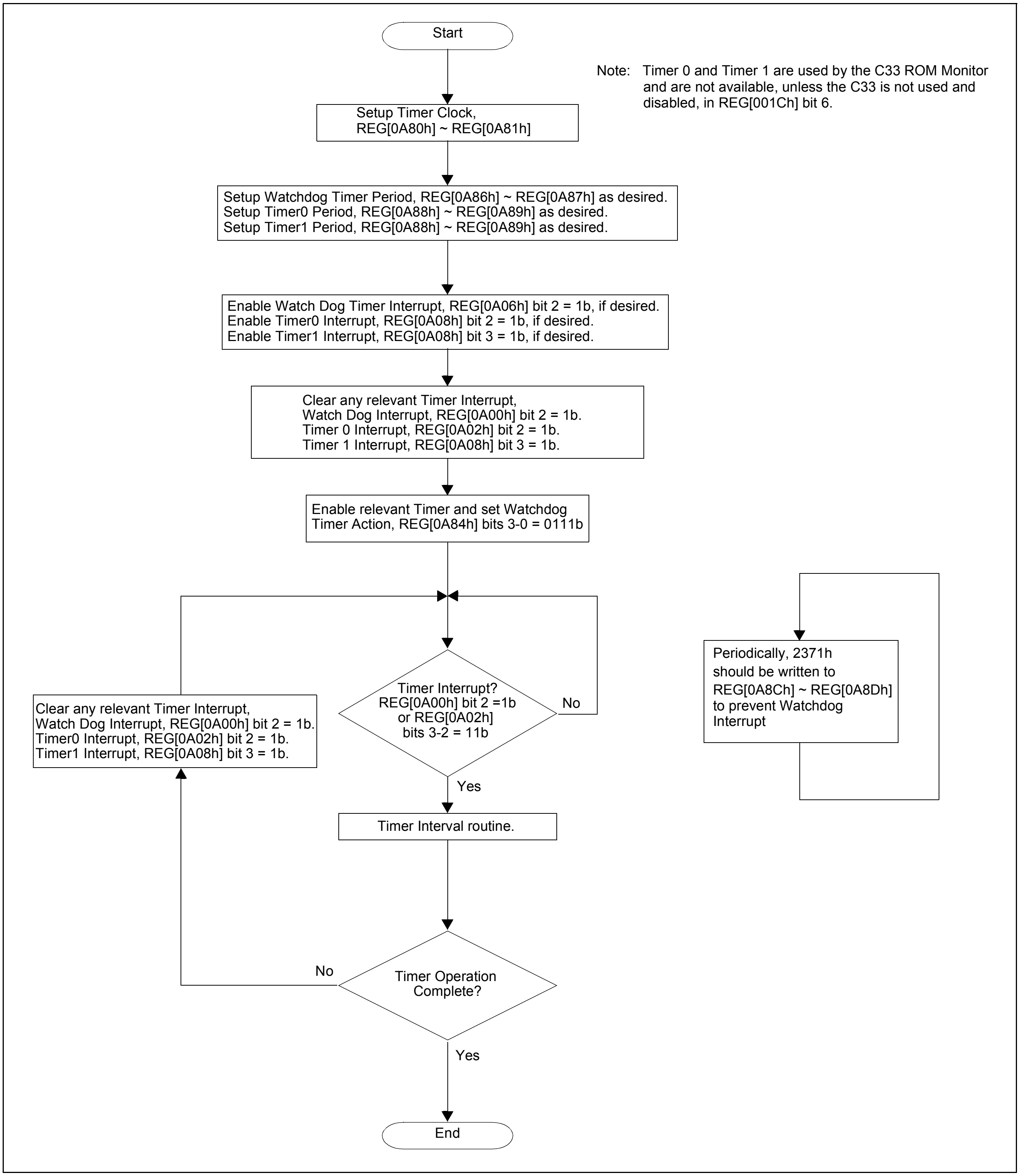


Figure 24-1: Typical Timer Operation Flow

Chapter 25 SPI Flash Memory Interface

25.1 Overview

The S1D13515/S2D13515 has dedicated SPI serial interface pins which can be used to access an external SPI device such as Serial Flash memory. (Although the main intent of the SPI pins is for a Serial Flash memory interface, it may also be used to interface to other external SPI devices.) The Serial Flash memory can be accessed (read/write data) by a sequence of operations on the SPI Interface Registers on a byte-by-byte basis. The S1D13515/S2D13515 also has dedicated logic (Serial Flash Read Logic) for allowing direct read data access of the Serial Flash memory through the internal VBUS bus interface. This logic allows for faster and more efficient reads and removes the need for software/firmware to program SPI registers to perform the reads.

The following diagram shows the SPI / Serial Flash Interface block in the S1D13515/S2D13515:

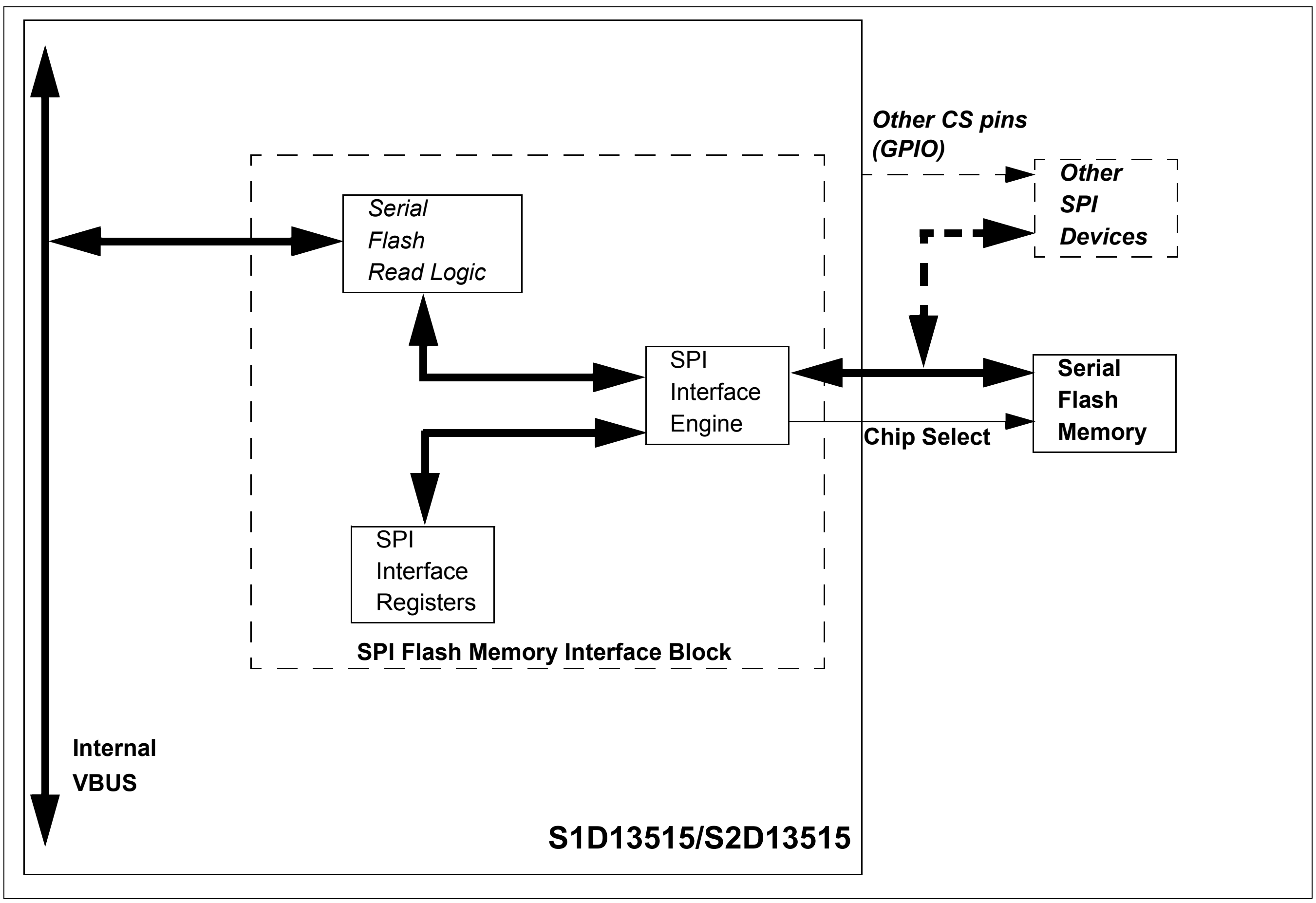


Figure 25-1: SPI Flash Memory Interface Block Diagram

25.2 IO Pins for SPI Interface

The S1D13515/S2D13515 has 3 dedicated pins for the SPI Interface: SPICS#, SPICLK, and SPIDIO. SPIDIO is a bidirectional data pin for reading/writing serial data. SPICS# is a dedicated chip-select pin intended for the Serial Flash Memory interface. Other external SPI devices can also be connected to SPICLK and SPIDIO by using another output pin of the S1D13515/S2D13515 such as a GPIO pin (assuming that the GPIO pin is not used for some other function).

Note
SPICS# is asserted automatically when using SPI Flash read logic (see REG[0B04h] bit 7, on page 326 for more information).

25.3 SPI Interface Registers

25.3.1 SPI Flash Chip Select Control Register

Bit 0 of the SPI Flash Chip Select Control Register (REG[0B0Ah]) is used to assert/deassert the SPICS# pin. The value programmed to this bit is the inverse of the SPICS# output. The default value of this bit is 0 (SPICS# = 1, chip-select is disabled).

25.3.2 SPI Flash Control Register

The SPI Flash Control Register configures the operation of the SPI Flash Memory Interface. It has the following control bits:

- Bit 0 is the SPI Flash Enable bit and should be set to 1 to enable the SPI Interface.
- Bits 2-1 are the SPI Flash Clock Phase Select and SPI Flash Clock Polarity Select bits for selecting the SPICLK phase and polarity.
- Bits 5-3 are the SPI Flash Clock Divide Select bits for programming the SPICLK frequency.
- Bit 7 is the SPI Flash Read Mode bit. When it is 0b, access to the external Serial Flash is through firmware programming of the SPI Registers. Control of the SPI Interface Engine is given to the SPI Interface Registers. When it is 1b, the Serial Flash Read Logic has control of the SPI Interface Engine and reads from the external Serial Flash memory can be performed through VBUS in the memory mapped region with starting base address of 2000_0000h / A000_0000h.

Note
1. The Serial Flash Read Logic feature requires serial flash that supports “Fast Read”.
2. The Serial Flash Read Logic feature is not available for host interfaces which do not support a WAIT/RDY pin.

25.3.3 SPI Flash Data Control Register

Bit 0 of the SPI Flash Data Control Register (REG[0B03h]) is used to control direction of the SPIDIO pin. When this bit is 0b, the SPIDIO pin is an input. When this bit is 1b, the SPIDIO pin is an output.

Note
SPDIO is automatically controlled when using SPI Flash read logic (see REG[0B04h] bit 7, on page 326 for more information).

25.3.4 SPI Flash Write Data Register

This write-only register (REG[0B02h]) is for triggering a byte serial transfer on the SPICLK/SPIDIO pins. Write a byte value to this register will cause the byte value to be serial shifted out on SPICLK/SPIDIO (assuming REG[0B03h] bit 0 is 1b).

25.3.5 SPI Flash Read Data Register

This read-only register (REG[0B00h]) is for reading byte data received from the SPI interface. In order to read a byte of data into the this register, a "dummy" write to REG[0B02h] should be performed with REG[0B03h] bit 0 set to 0b (SPIDIO is input).

25.3.6 SPI Flash Status Register

This read-only register (REG[0B06h]) provides status bits indicating the state of the SPI Interface Engine. The following are the status bits in this register:

- Bit 0 is the SPI Flash Read Data Ready Flag. It is set to 1b whenever a new byte data has been loaded into the SPI Flash Read Data Register (REG[0B00h]). It is cleared when REG[0B00h] is read.
- Bit 1 is the SPI Flash Read Data Overrun Flag. It is set to 1b whenever a new byte data is loaded into the SPI Flash Read Data Register (REG[0B00h]) and the SPI Flash Read Data Ready Flag (bit 0) is still 1 (indicating that the previous byte has not yet been read out). This bit is cleared by reading REG[0B00h].
- Bit 2 is the SPI Flash Write Data Register Empty Flag. It is 1b whenever the SPI Flash Write Data Register (REG[0B02h]) is empty. Writing a byte value to REG[0B02h] will initially cause this bit to become 0b. When the byte value is transferred to the serial shift register, this bit becomes 1b again.
- Bit 3 is the SPI Flash Busy Flag. It is 1b when the SPI Interface Engine is busy shifting a byte of data in/out on the SPI interface.

25.4 SPI Interface Operation Flow

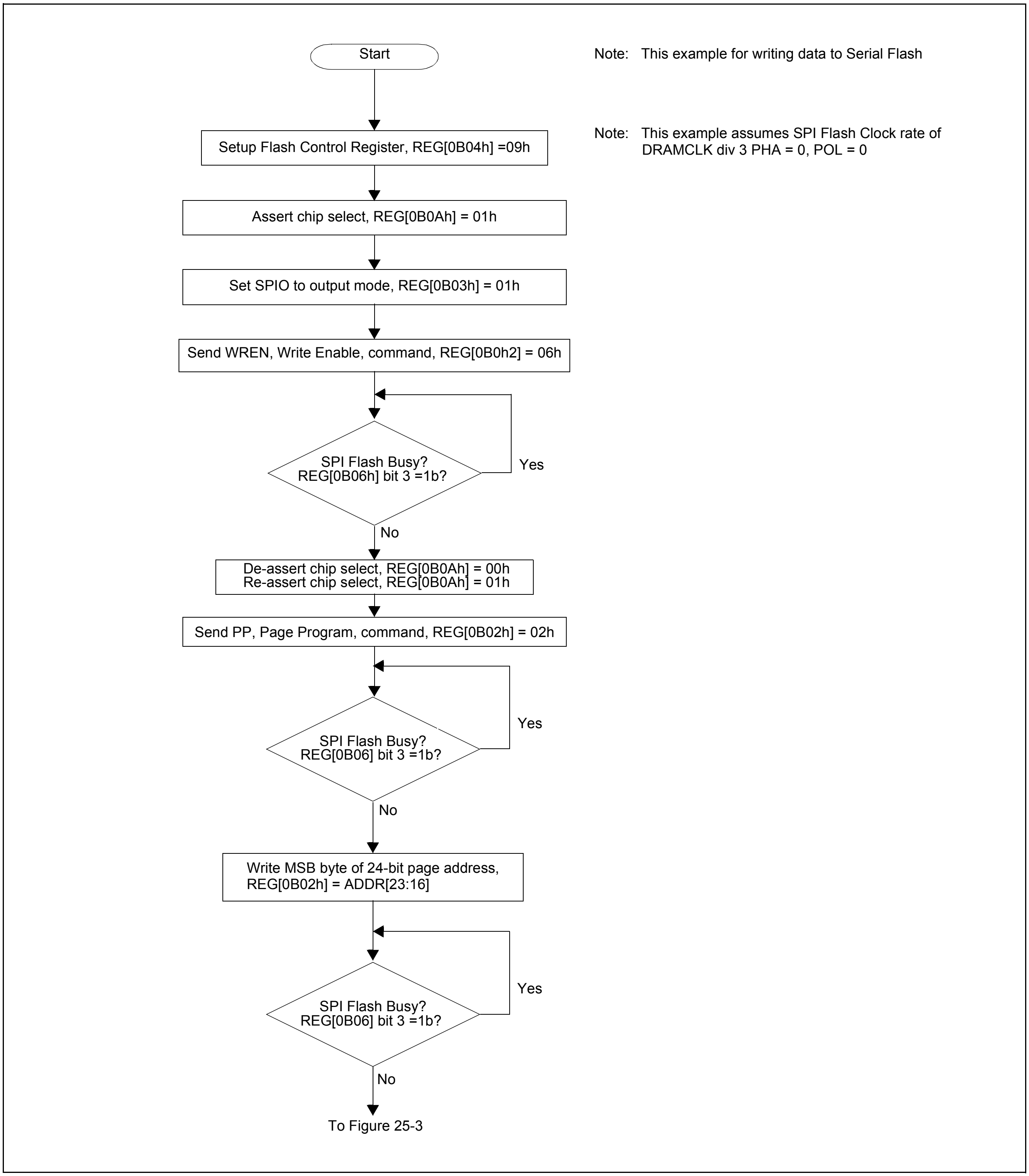


Figure 25-2: Typical SPI Interface Write Operation Flow

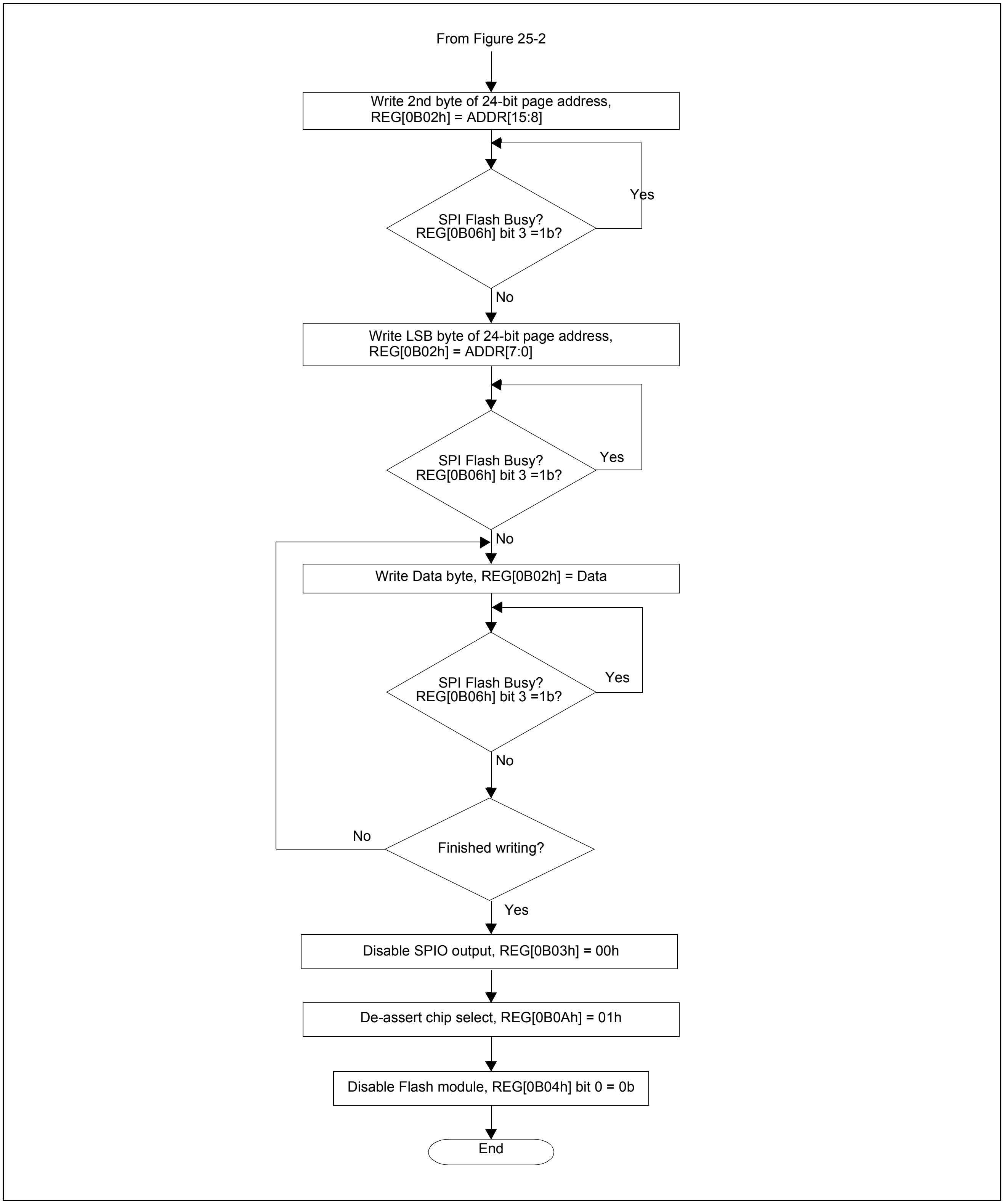


Figure 25-3: Typical SPI Interface Write Operation Flow (Continued)

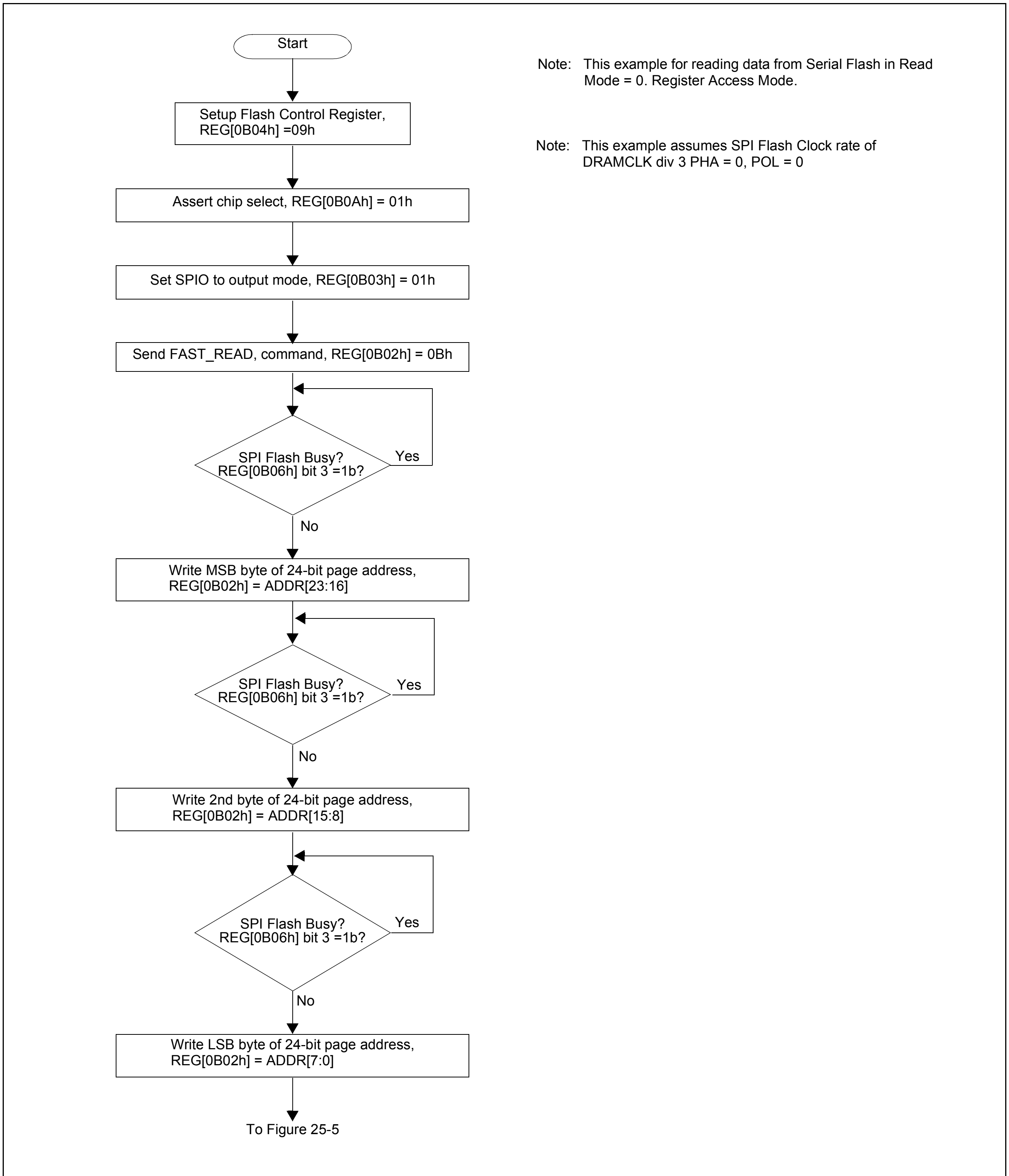


Figure 25-4: Typical SPI Interface Serial Flash Mode 0 Read Operation Flow

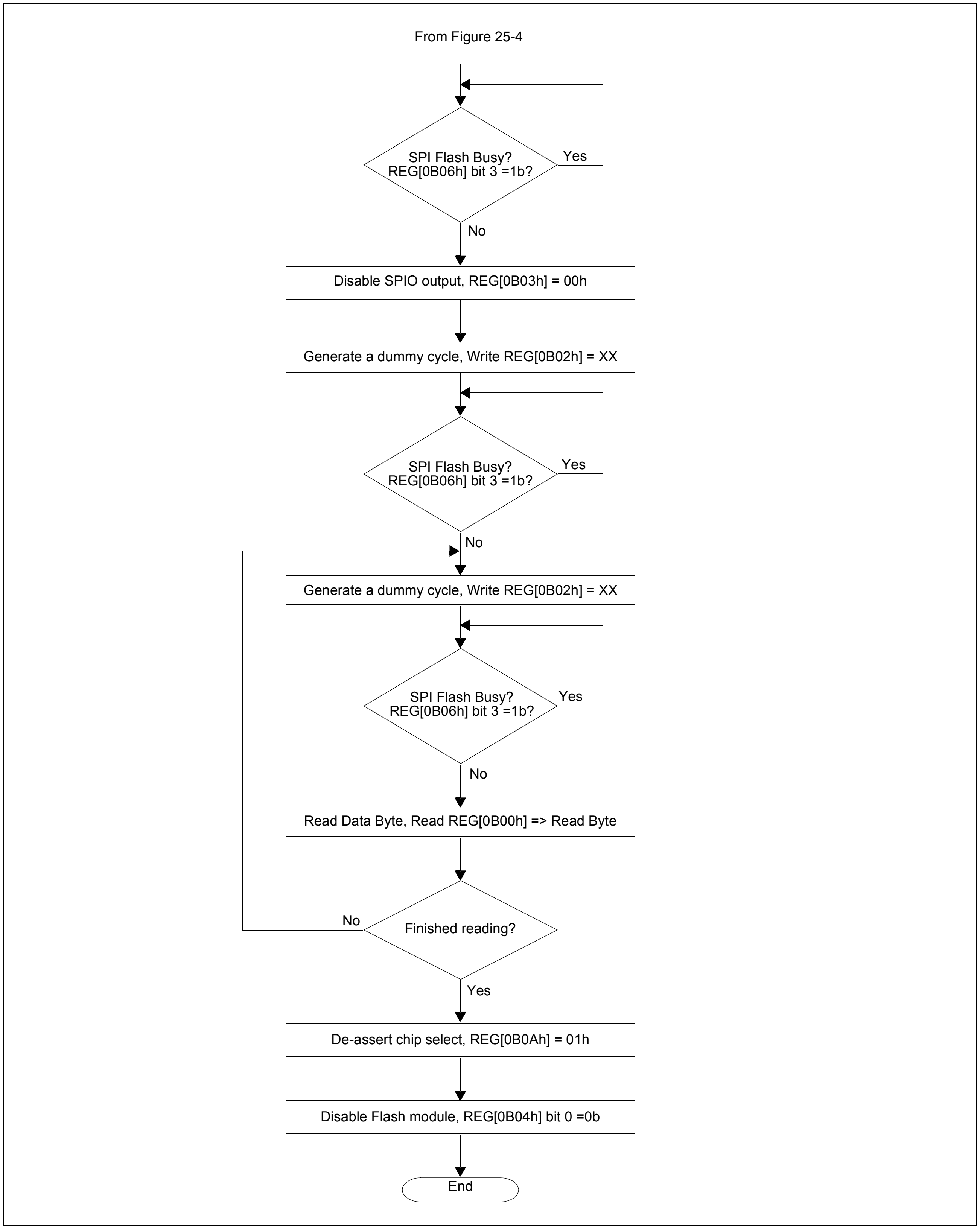


Figure 25-5: Typical SPI Interface Serial Flash Mode 0 Read Operation Flow (Continued

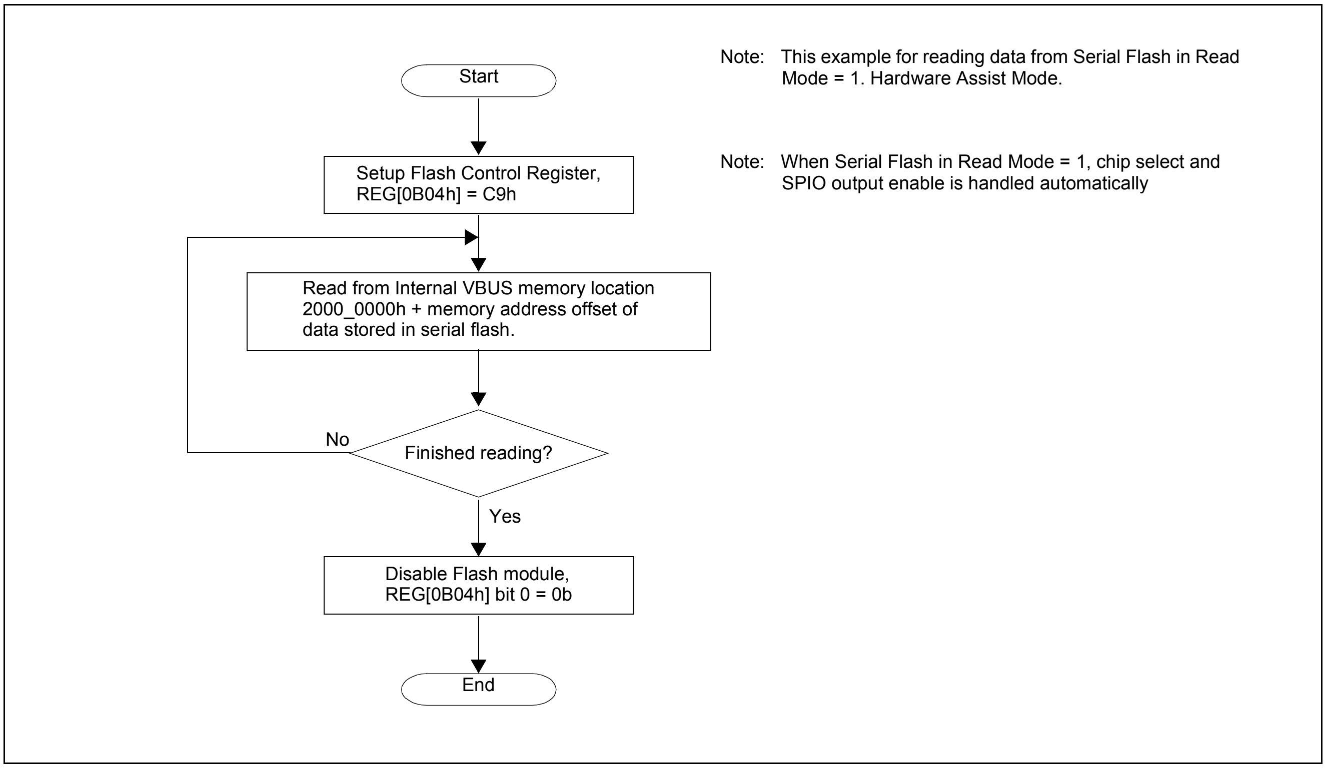
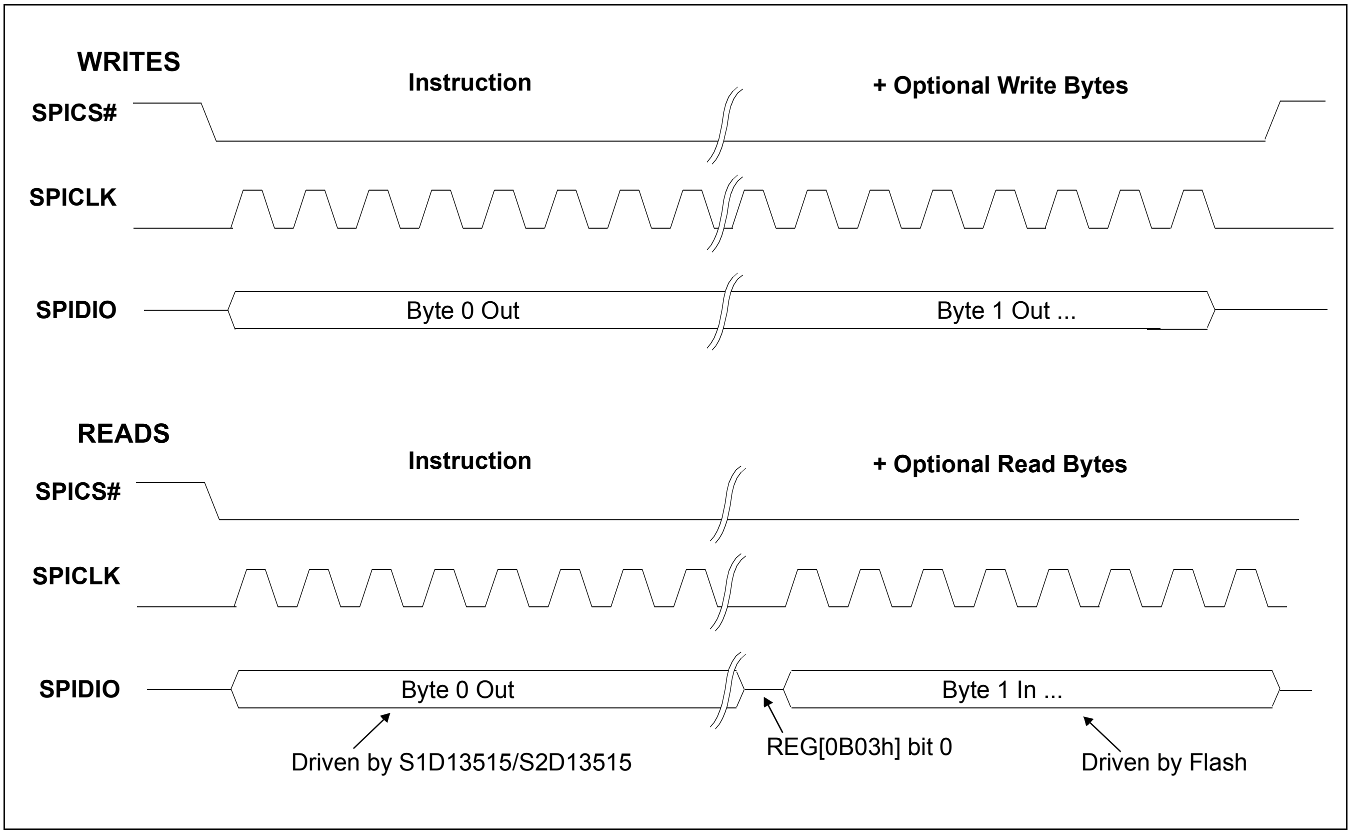


Figure 25-6: Typical SPI Interface Serial Flash Mode 1 Read Operation Flow

25.5 SPI Flash Interface Timing



Chapter 26 JTAG Interface

The S1D13515/S2D13515 is designed with a JTAG interface which can be used for Boundary-Scan testing. The S1D13515/S2D13515 JTAG interface is compliant with the IEEE 1149.1 standard. For details on the JTAG test access port, refer to the *IEEE Std 1149.1a-1993*.

26.1 JTAG Pins

The S1D13515/S2D13515 JTAG interface uses 5 pins. For further details on the JTAG pins, see Section 5.3.7, “Miscellaneous” on page 30.

- TCK is the test clock input which controls the timing of the test interface. This clock is supplied to the test logic and is independent of the system clock.
- TMS is the test mode input and controls the state changes during test operations (see Figure 26-2: “TAP Controller State Machine,” on page 565). TMS is sampled on the rising edge of TCK.
- TDI is the test data input and inputs test instruction codes and test data serially to the test logic. TDI is sampled on the rising edge of TCK.
- TDO is the test data output and outputs data serially from the test logic. TDO is Tacda of TCK. Is changed, and it is output only at Shift-IR and the state of Shift-DR. Other cases become Hi-Z.
- TRST is the test reset input. It is an active low signal which asynchronously initializes the test logic. When TRST is high, TMS must be kept high. If TMS remains high for five or more TCK rising edges, the test logic is initialized. For normal operations, TRST must be tied to VSS or connected to RESET#.

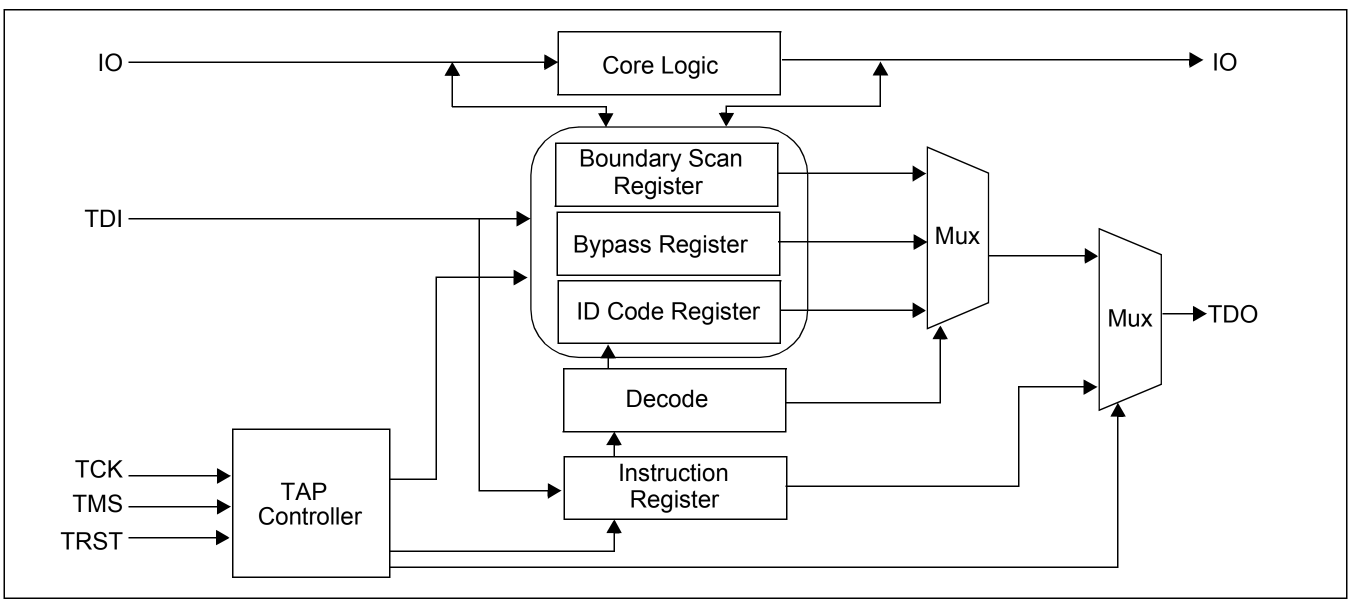


Figure 26-1: Overview of JTAG Circuit

26.2 TAP Controller

JTAG operation is controlled by a state machine called the TAP controller. Figure 26-2: shows the states of the TAP controller. Transitions between each of the 16 states are controlled by the value of TMS at the rising edge of TCK. The TAP controller has three main states and two possible paths. All other states are temporary states or allow changes in flow.

26.2.1 TAP Controller Paths

The TAP controller has two main paths.

- DR Path (Data Register)
  The DR path is used to write new data into the data register (Boundary Scan, Bypass, or ID Code) as specified by the Instruction Register. It can also provide the previous value. The new value is shifted into the currently selected Data Register through the TDI pin one bit at a time when the Shift-DR state is entered. The previous value is shifted out to the TDO pin one bit at a time when the Shift-DR state is exited.
- IR Path (Instruction Register)
  The IR path is used to write a new instruction code into the instruction register. It can also provide the previous value. The new value is shifted into the IR through the TDI pin one bit at a time when the Shift-IR state is entered. The previous value is shifted out to the TDO pin one bit at a time when the Shift-IR state is exited.

26.2.2 TAP Controller Main State

The TAP controller has three main states.

- The Capture state prepares the Instruction Register (Capture-IR) or Data Register (Capture-DR) for data to be shifted in/out for the boundary scanning test.
- The Shift state allows new data to be input through TDI, or existing data to be output through TDO. Shift-IR allows data access for the Instruction Register and Shift-DR allows data access for the Data Register.
- The Update state applies the new data that has been shifted in/out. For Update-IR, the new instruction takes effect. For Update-DR, the new data appears for output from the Boundary Scan Register (BSR).

26.2.3 TAP Controller State Machine

The following figure shows the S1D13515/S2D13515 TAP Controller state machine.

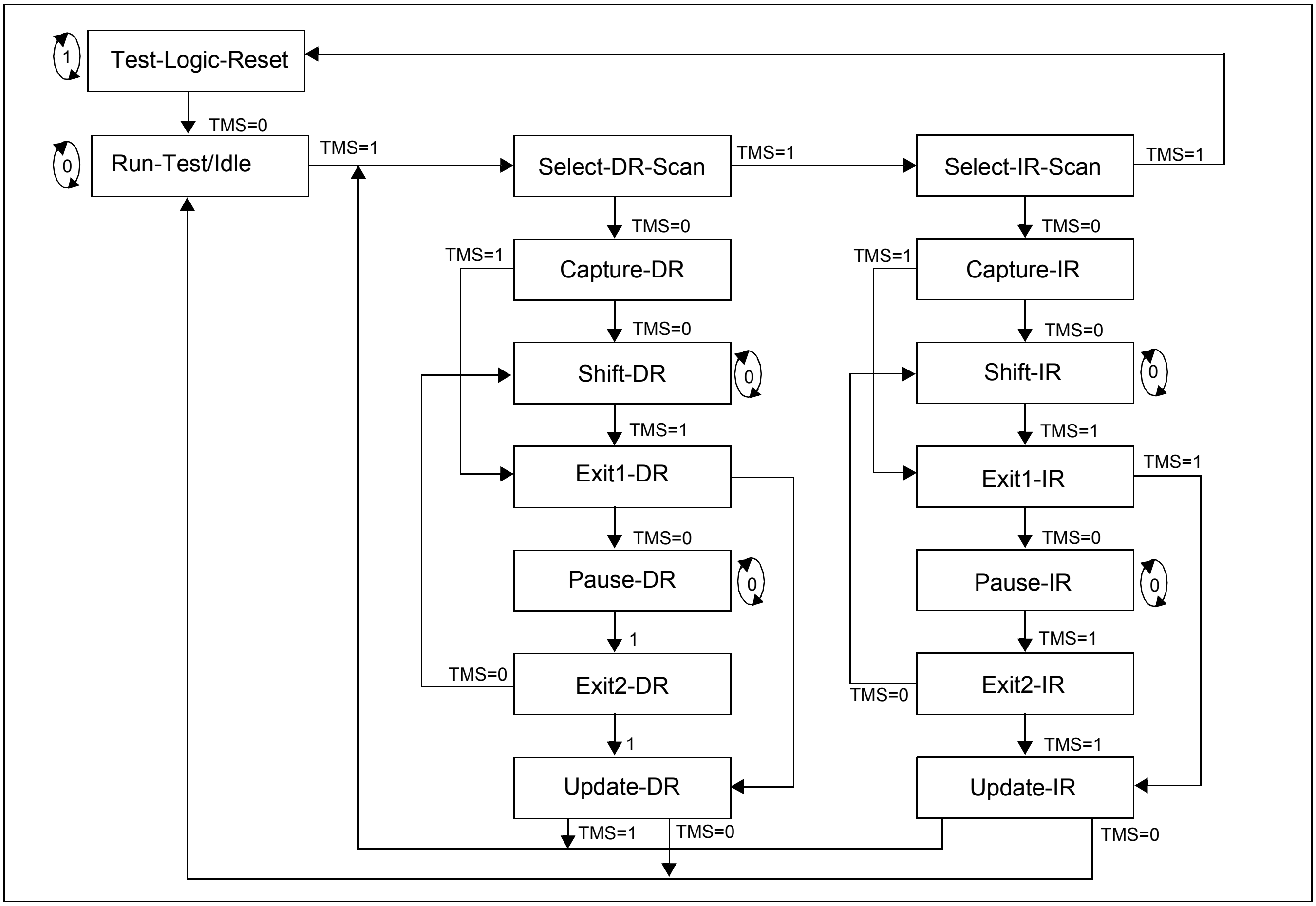


Figure 26-2: TAP Controller State Machine

26.3 JTAG Instruction Codes

The S1D13515/S2D13515 supports the instructions EXTEST, CLAMP, SAMPLE/PRELOAD, and BYPASS as detailed in IEEE1149.1. Device recognition instructions (IDCODE) are also supported. Each instruction code and its function are shown in the following table.

Table 26-1: JTAG Instruction Codes

Instruction	Instruction code	Function
EXTEST	000b	This instruction samples the S1D13515/S2D13515 pin states and captures them to the BSR when the Capture-DR state is entered. When the Shift-DR state is entered, the contents of the BSR are shifted out through the TDO line. At the same time new data is shifted in. This new data will be applied to the S1D13515/S2D13515 pins when the Update-DR state is entered.
CLAMP	001b	This instruction sets the outputs of the S1D13515/S2D13515 to logic levels specified in the boundary-scan register, while the bypass register is connected from TDI to TDO.

Table 26-1: JTAG Instruction Codes

IDCODE	011b	This instruction outputs the identification code for the device and manufacturer on TDO. For a description of the S1D13515/S2D13515 identification code, see Table 26-2:, “Identification Code”.
SAMPLE/ PRELOAD	100b	This instruction samples the S1D13515/S2D13515 internal core logic signals and captures them to the BSR when the Capture-DR state is entered. When the Shift-DR state is entered, the contents of the BSR are shifted out through the TDO line and at the same time new data may also be shifted in. This new data in the BSR will be applied to the S1D13515/S2D13515 core logic when the Update-DR state is entered.
BYPASS	111b	This instruction bypasses boundary scanning when the S1D13515/S2D13515 is not targeted. For this instruction the TDI and TDO lines are connected and data pass through the S1D13515/S2D13515.

Table 26-2: Identification Code

Identification Code Description	Value
Version Number (4 bits)	0001b
Part Number (16 bits)	0000000000011011b
Identity of the manufacturer (11 bits)	00010111110b
LSB (1 bit)	1b

26.3.1 Boundary Scan Cell Definitions

The following list specifies the characteristics of each cell in the boundary scan register from TDI to TDO. The following is a description of the label fields:

- num The cell number.
- cell The cell type as defined by the standard.
- port The design port name. Control cells do not have a port name.
- function The function of the cell as defined by the standard (input, output2, output3, bidir, control or controlr).
- safe Specifies the value that the BSR cell should be loaded with for safe operation when the software might otherwise choose a random value.
- ccell The control cell number. Specifies the control cell that drives the output enable for this port. Writing a 1 to a control cell disables the output enable for the corresponding port.
- disval Specifies the value that is loaded into the control cell to disable the output enable for the corresponding port.
- rslt Resulting state. Shows the state of the driver when it is disabled.

26.3.2 Example BSDL File for the S2D13515

```
-- ******************************************************************************
--   BSDL file for design s2d13515
-- ******************************************************************************

 entity s2d13515 is

-- This section identifies the default device package selected.

   generic (PHYSICAL_PIN_MAP: string:= "S2D13515_QFP22_256");

-- This section declares all the ports in the design.

   port (
          AB0       : in       bit;
          AB1       : in       bit;
          AB18      : in       bit;
          AB2       : in       bit;
          AB3       : in       bit;
          AB4       : in       bit;
          AB5       : in       bit;
          BDIPX     : in       bit;
          BE0X      : in       bit;
          BURSTX    : in       bit;
          BUSCLK    : in       bit;
          CLKI2     : in       bit;
          CM1CLKI   : in       bit;
          CM1D0     : in       bit;
          CM1D1     : in       bit;
          CM1D2     : in       bit;
          CM1D3     : in       bit;
          CM1D4     : in       bit;
          CM1D5     : in       bit;
          CM1D6     : in       bit;
          CM1D7     : in       bit;
          CM1FIELD  : in       bit;
```

```
CM1HREF   : in      bit;
CM1VREF   : in      bit;
CNF0      : in      bit;
CNF1      : in      bit;
CNF2      : in      bit;
CSX       : in      bit;
RDX       : in      bit;
RESETX    : in      bit;
TCK       : in      bit;
TDI       : in      bit;
TMS       : in      bit;
TRST      : in      bit;
WRX       : in      bit;
AB10      : inout   bit;
AB11      : inout   bit;
AB12      : inout   bit;
AB13      : inout   bit;
AB14      : inout   bit;
AB15      : inout   bit;
AB16      : inout   bit;
AB17      : inout   bit;
AB19      : inout   bit;
AB20      : inout   bit;
AB6       : inout   bit;
AB7       : inout   bit;
AB8       : inout   bit;
AB9       : inout   bit;
BE1X      : inout   bit;
BSX       : inout   bit;
DB0       : inout   bit;
DB1       : inout   bit;
DB10      : inout   bit;
DB11      : inout   bit;
DB12      : inout   bit;
DB13      : inout   bit;
DB14      : inout   bit;
DB15      : inout   bit;
DB2       : inout   bit;
DB3       : inout   bit;
DB4       : inout   bit;
DB5       : inout   bit;
DB6       : inout   bit;
DB7       : inout   bit;
DB8       : inout   bit;
DB9       : inout   bit;
FP1IO0    : inout   bit;
FP1IO1    : inout   bit;
FP1IO10   : inout   bit;
FP1IO11   : inout   bit;
FP1IO12   : inout   bit;
FP1IO13   : inout   bit;
FP1IO14   : inout   bit;
FP1IO15   : inout   bit;
FP1IO16   : inout   bit;
FP1IO17   : inout   bit;
FP1IO18   : inout   bit;
FP1IO19   : inout   bit;
FP1IO2    : inout   bit;
FP1IO20   : inout   bit;
FP1IO21   : inout   bit;
FP1IO22   : inout   bit;
FP1IO23   : inout   bit;
FP1IO3    : inout   bit;
FP1IO4    : inout   bit;
FP1IO5    : inout   bit;
FP1IO6    : inout   bit;
FP1IO7    : inout   bit;
FP1IO8    : inout   bit;
FP1IO9    : inout   bit;
FP2IO17   : inout   bit;
FP2IO18   : inout   bit;
FP2IO19   : inout   bit;
FP2IO20   : inout   bit;
FP2IO21   : inout   bit;
FP2IO22   : inout   bit;
FP2IO23   : inout   bit;
I2CSCL    : inout   bit;
I2CSDA    : inout   bit;
I2SCKO    : inout   bit;
I2SWSO    : inout   bit;
```

```
MEMDQ0     : inout    bit;
MEMDQ1     : inout    bit;
MEMDQ10    : inout    bit;
MEMDQ11    : inout    bit;
MEMDQ12    : inout    bit;
MEMDQ13    : inout    bit;
MEMDQ14    : inout    bit;
MEMDQ15    : inout    bit;
MEMDQ16    : inout    bit;
MEMDQ17    : inout    bit;
MEMDQ18    : inout    bit;
MEMDQ19    : inout    bit;
MEMDQ2     : inout    bit;
MEMDQ20    : inout    bit;
MEMDQ21    : inout    bit;
MEMDQ22    : inout    bit;
MEMDQ23    : inout    bit;
MEMDQ24    : inout    bit;
MEMDQ25    : inout    bit;
MEMDQ26    : inout    bit;
MEMDQ27    : inout    bit;
MEMDQ28    : inout    bit;
MEMDQ29    : inout    bit;
MEMDQ3     : inout    bit;
MEMDQ30    : inout    bit;
MEMDQ31    : inout    bit;
MEMDQ4     : inout    bit;
MEMDQ5     : inout    bit;
MEMDQ6     : inout    bit;
MEMDQ7     : inout    bit;
MEMDQ8     : inout    bit;
MEMDQ9     : inout    bit;
MNRX       : inout    bit;
SPIDIO     : inout    bit;
TEAX       : inout    bit;
WAITX      : inout    bit;
IRQ        : out      bit;
TDO        : out      bit;
CM1CLKO    : buffer   bit;
FP2IO0     : buffer   bit;
FP2IO1     : buffer   bit;
FP2IO10    : buffer   bit;
FP2IO11    : buffer   bit;
FP2IO12    : buffer   bit;
FP2IO13    : buffer   bit;
FP2IO14    : buffer   bit;
FP2IO15    : buffer   bit;
FP2IO16    : buffer   bit;
FP2IO2     : buffer   bit;
FP2IO24    : buffer   bit;
FP2IO25    : buffer   bit;
FP2IO26    : buffer   bit;
FP2IO27    : buffer   bit;
FP2IO3     : buffer   bit;
FP2IO4     : buffer   bit;
FP2IO5     : buffer   bit;
FP2IO6     : buffer   bit;
FP2IO7     : buffer   bit;
FP2IO8     : buffer   bit;
FP2IO9     : buffer   bit;
I2SMCLKO   : buffer   bit;
I2SSDO     : buffer   bit;
MEMA0      : buffer   bit;
MEMA1      : buffer   bit;
MEMA10     : buffer   bit;
MEMA11     : buffer   bit;
MEMA12     : buffer   bit;
MEMA2      : buffer   bit;
MEMA3      : buffer   bit;
MEMA4      : buffer   bit;
MEMA5      : buffer   bit;
MEMA6      : buffer   bit;
MEMA7      : buffer   bit;
MEMA8      : buffer   bit;
MEMA9      : buffer   bit;
MEMBA0     : buffer   bit;
MEMBA1     : buffer   bit;
MEMCASX    : buffer   bit;
MEMCKE     : buffer   bit;
MEMCLK     : buffer   bit;
```

```
        MEMCSX    : buffer   bit;
        MEMDQM0   : buffer   bit;
        MEMDQM1   : buffer   bit;
        MEMDQM2   : buffer   bit;
        MEMDQM3   : buffer   bit;
        MEMRASX   : buffer   bit;
        MEMWEX    : buffer   bit;
        PWM0      : buffer   bit;
        PWM1      : buffer   bit;
        SPICK     : buffer   bit;
        SPICS     : buffer   bit;
        OSCI1     : linkage  bit;
        OSCO1     : linkage  bit;
        TESTEN    : linkage  bit;
        VCP1      : linkage  bit;
        VCP2      : linkage  bit;
        HVDDY     : linkage  bit_vector (1 to 19);
        LVDDY     : linkage  bit_vector (1 to 11);
        VSSY      : linkage  bit_vector (1 to 27)
   );

   use STD_1149_1_2001.all;

   attribute COMPONENT_CONFORMANCE of s2d13515: entity is "STD_1149_1_2001";

   attribute PIN_MAP of s2d13515: entity is PHYSICAL_PIN_MAP;

-- This section specifies the pin map for each port. This information is
-- extracted from the port-to-pin map file that was read in using the
-- "read_pin_map" command.

     constant S2D13515_QFP22_256: PIN_MAP_STRING :=
        "AB0       : 27," &
        "AB1       : 26," &
        "AB18      : 6," &
        "AB2       : 25," &
        "AB3       : 24," &
        "AB4       : 23," &
        "AB5       : 22," &
        "BDIPX     : 60," &
        "BE0X      : 56," &
        "BURSTX    : 59," &
        "BUSCLK    : 39," &
        "CLKI2     : 2," &
        "CM1CLKI   : 240," &
        "CM1D0     : 246," &
        "CM1D1     : 245," &
        "CM1D2     : 244," &
        "CM1D3     : 243," &
        "CM1D4     : 242," &
        "CM1D5     : 237," &
        "CM1D6     : 236," &
        "CM1D7     : 235," &
        "CM1FIELD  : 231," &
        "CM1HREF   : 233," &
        "CM1VREF   : 232," &
        "CNF0      : 155," &
        "CNF1      : 154," &
        "CNF2      : 153," &
        "CSX       : 52," &
        "RDX       : 54," &
        "RESETX    : 65," &
        "TCK       : 148," &
        "TDI       : 149," &
        "TMS       : 150," &
        "TRST      : 151," &
        "WRX       : 55," &
        "AB10      : 17," &
        "AB11      : 13," &
        "AB12      : 12," &
        "AB13      : 11," &
        "AB14      : 10," &
        "AB15      : 9," &
        "AB16      : 8," &
        "AB17      : 7," &
        "AB19      : 5," &
        "AB20      : 4," &
        "AB6       : 21," &
        "AB7       : 20," &
        "AB8       : 19," &
```

```
"AB9       : 18," &
"BE1X      : 57," &
"BSX       : 58," &
"DB0       : 48," &
"DB1       : 47," &
"DB10      : 35," &
"DB11      : 34," &
"DB12      : 33," &
"DB13      : 32," &
"DB14      : 31," &
"DB15      : 30," &
"DB2       : 46," &
"DB3       : 45," &
"DB4       : 44," &
"DB5       : 43," &
"DB6       : 42," &
"DB7       : 41," &
"DB8       : 37," &
"DB9       : 36," &
"FP1IO0    : 96," &
"FP1IO1    : 95," &
"FP1IO10   : 84," &
"FP1IO11   : 70," &
"FP1IO12   : 83," &
"FP1IO13   : 78," &
"FP1IO14   : 69," &
"FP1IO15   : 77," &
"FP1IO16   : 76," &
"FP1IO17   : 68," &
"FP1IO18   : 67," &
"FP1IO19   : 66," &
"FP1IO2    : 94," &
"FP1IO20   : 75," &
"FP1IO21   : 74," &
"FP1IO22   : 73," &
"FP1IO23   : 79," &
"FP1IO3    : 93," &
"FP1IO4    : 90," &
"FP1IO5    : 89," &
"FP1IO6    : 88," &
"FP1IO7    : 87," &
"FP1IO8    : 86," &
"FP1IO9    : 85," &
"FP2IO17   : 109," &
"FP2IO18   : 108," &
"FP2IO19   : 107," &
"FP2IO20   : 106," &
"FP2IO21   : 105," &
"FP2IO22   : 104," &
"FP2IO23   : 103," &
"I2CSCL    : 229," &
"I2CSDA    : 230," &
"I2SCKO    : 136," &
"I2SWSO    : 134," &
"MEMDQ0    : 218," &
"MEMDQ1    : 216," &
"MEMDQ10   : 191," &
"MEMDQ11   : 189," &
"MEMDQ12   : 185," &
"MEMDQ13   : 183," &
"MEMDQ14   : 179," &
"MEMDQ15   : 177," &
"MEMDQ16   : 217," &
"MEMDQ17   : 215," &
"MEMDQ18   : 213," &
"MEMDQ19   : 211," &
"MEMDQ2    : 214," &
"MEMDQ20   : 204," &
"MEMDQ21   : 202," &
"MEMDQ22   : 200," &
"MEMDQ23   : 198," &
"MEMDQ24   : 194," &
"MEMDQ25   : 192," &
"MEMDQ26   : 190," &
"MEMDQ27   : 186," &
"MEMDQ28   : 184," &
"MEMDQ29   : 182," &
"MEMDQ3    : 212," &
"MEMDQ30   : 178," &
"MEMDQ31   : 176," &
```

```
        "MEMDQ4    : 210," &
        "MEMDQ5    : 203," &
        "MEMDQ6    : 201," &
        "MEMDQ7    : 199," &
        "MEMDQ8    : 197," &
        "MEMDQ9    : 193," &
        "MNRX      : 53," &
        "SPIDIO    : 145," &
        "TEAX      : 61," &
        "WAITX     : 62," &
        "IRQ       : 63," &
        "TDO       : 147," &
        "CM1CLKO   : 238," &
        "FP2IO0    : 133," &
        "FP2IO1    : 132," &
        "FP2IO10   : 119," &
        "FP2IO11   : 118," &
        "FP2IO12   : 117," &
        "FP2IO13   : 116," &
        "FP2IO14   : 115," &
        "FP2IO15   : 114," &
        "FP2IO16   : 110," &
        "FP2IO2    : 131," &
        "FP2IO24   : 100," &
        "FP2IO25   : 99," &
        "FP2IO26   : 98," &
        "FP2IO27   : 97," &
        "FP2IO3    : 128," &
        "FP2IO4    : 127," &
        "FP2IO5    : 126," &
        "FP2IO6    : 125," &
        "FP2IO7    : 124," &
        "FP2IO8    : 123," &
        "FP2IO9    : 120," &
        "I2SMCLKO  : 137," &
        "I2SSDO    : 135," &
        "MEMA0     : 228," &
        "MEMA1     : 227," &
        "MEMA10    : 171," &
        "MEMA11    : 170," &
        "MEMA12    : 169," &
        "MEMA2     : 226," &
        "MEMA3     : 225," &
        "MEMA4     : 224," &
        "MEMA5     : 223," &
        "MEMA6     : 222," &
        "MEMA7     : 221," &
        "MEMA8     : 173," &
        "MEMA9     : 172," &
        "MEMBA0    : 168," &
        "MEMBA1    : 167," &
        "MEMCASX   : 163," &
        "MEMCKE    : 160," &
        "MEMCLK    : 208," &
        "MEMCSX    : 161," &
        "MEMDQM0   : 159," &
        "MEMDQM1   : 158," &
        "MEMDQM2   : 157," &
        "MEMDQM3   : 156," &
        "MEMRASX   : 162," &
        "MEMWEX    : 166," &
        "PWM0      : 139," &
        "PWM1      : 138," &
        "SPICK     : 144," &
        "SPICS     : 143," &
        "OSCI1     : 249," &
        "OSCO1     : 248," &
        "TESTEN    : 146," &
        "VCP1      : 255," &
        "VCP2      : 252," &
        "HVDDY     : (14, 28, 40, 51, 72, 82, 92, 101, 111, 121, 129, 140, " &
        "165, 175, 188, 195, 205, 220, 234)," &
        "LVDDY     : (16, 50, 80, 113, 142, 181, 206, 241, 250, 253, 256)," &
        "VSSY      : (1, 3, 15, 29, 38, 49, 64, 71, 81, 91, 102, 112, 122, " &
        "130, 141, 152, 164, 174, 180, 187, 196, 207, 209, 219, 239, 251, " &
        "254)";

-- This section specifies the TAP ports. For the TAP TCK port, the parameters in
-- the brackets are:
--        First Field : Maximum  TCK frequency.
```

```
--       Second Field: Allowable states TCK may be stopped in.

   attribute TAP_SCAN_CLOCK of TCK : signal is (5.000000e+06, BOTH);
   attribute TAP_SCAN_IN    of TDI : signal is true;
   attribute TAP_SCAN_MODE  of TMS : signal is true;
   attribute TAP_SCAN_OUT   of TDO : signal is true;
   attribute TAP_SCAN_RESET of TRST: signal is true;

-- Specifies the compliance enable patterns for the design. It lists a set of
-- design ports and the values that they should be set to, in order to enable
-- compliance to IEEE Std 1149.1

   attribute COMPLIANCE_PATTERNS of s2d13515: entity is
        "(TESTEN) (0)";

-- Specifies the number of bits in the instruction register.

   attribute INSTRUCTION_LENGTH of s2d13515: entity is 3;

-- Specifies the boundary-scan instructions implemented in the design and their
-- opcodes.

   attribute INSTRUCTION_OPCODE of s2d13515: entity is
     "BYPASS  (111)," &
     "EXTEST  (000)," &
     "SAMPLE  (100)," &
     "PRELOAD (100)," &
     "CLAMP   (001)," &
     "IDCODE  (011)";

-- Specifies the bit pattern that is loaded into the instruction register when
-- the TAP controller passes through the Capture-IR state. The standard mandates
-- that the two LSBs must be "01". The remaining bits are design specific.

   attribute INSTRUCTION_CAPTURE of s2d13515: entity is "001";

-- Specifies the bit pattern that is loaded into the DEVICE_ID register during
-- the IDCODE instruction when the TAP controller passes through the Capture-DR
-- state.

   attribute IDCODE_REGISTER of s2d13515: entity is
     "0001" &
 -- 4-bit version number
     "0000000000011011" &
 -- 16-bit part number
     "00010111110" &
 -- 11-bit identity of the manufacturer
     "1";
 -- Required by IEEE Std 1149.1

-- This section specifies the test data register placed between TDI and TDO for
-- each implemented instruction.

   attribute REGISTER_ACCESS of s2d13515: entity is
        "BYPASS    (BYPASS, CLAMP)," &
        "BOUNDARY  (EXTEST, SAMPLE, PRELOAD)," &
        "DEVICE_ID (IDCODE)";

-- Specifies the length of the boundary scan register.

   attribute BOUNDARY_LENGTH of s2d13515: entity is 347;

-- The following list specifies the characteristics of each cell in the boundary
-- scan register from TDI to TDO. The following is a description of the label
-- fields:
--      num     : Is the cell number.
--      cell    : Is the cell type as defined by the standard.
--      port    : Is the design port name. Control cells do not have a port
--                name.
--      function: Is the function of the cell as defined by the standard. Is one
--                of input, output2, output3, bidir, control or controlr.
--      safe    : Specifies the value that the BSR cell should be loaded with
--                for safe operation when the software might otherwise choose a
--                random value.
--      ccell   : The control cell number. Specifies the control cell that
--                drives the output enable for this port.
--      disval  : Specifies the value that is loaded into the control cell to
--                disable the output enable for the corresponding port.
--      rslt    : Resulting state. Shows the state of the driver when it is
--                disabled.
```

```
   attribute BOUNDARY_REGISTER of s2d13515: entity is
--
--    num   cell   port      function      safe  [ccell  disval  rslt]
--
     "346  (BC_4,  CLKI2,    observe_only, X),                        " &
     "345  (BC_4,  AB20,     observe_only, X),                        " &
     "344  (BC_2,  *,        control,      1),                        " &
     "343  (BC_1,  AB20,     output3,      X,    344,    1,      Z),  " &
     "342  (BC_4,  AB19,     observe_only, X),                        " &
     "341  (BC_2,  *,        control,      1),                        " &
     "340  (BC_1,  AB19,     output3,      X,    341,    1,      Z),  " &
     "339  (BC_4,  AB18,     observe_only, X),                        " &
     "338  (BC_4,  AB17,     observe_only, X),                        " &
     "337  (BC_2,  *,        control,      1),                        " &
     "336  (BC_1,  AB17,     output3,      X,    337,    1,      Z),  " &
     "335  (BC_4,  AB16,     observe_only, X),                        " &
     "334  (BC_2,  *,        control,      1),                        " &
     "333  (BC_1,  AB16,     output3,      X,    334,    1,      Z),  " &
     "332  (BC_4,  AB15,     observe_only, X),                        " &
     "331  (BC_2,  *,        control,      1),                        " &
     "330  (BC_1,  AB15,     output3,      X,    331,    1,      Z),  " &
     "329  (BC_4,  AB14,     observe_only, X),                        " &
     "328  (BC_2,  *,        control,      1),                        " &
     "327  (BC_1,  AB14,     output3,      X,    328,    1,      Z),  " &
     "326  (BC_4,  AB13,     observe_only, X),                        " &
     "325  (BC_2,  *,        control,      1),                        " &
     "324  (BC_1,  AB13,     output3,      X,    325,    1,      Z),  " &
     "323  (BC_4,  AB12,     observe_only, X),                        " &
     "322  (BC_2,  *,        control,      1),                        " &
     "321  (BC_1,  AB12,     output3,      X,    322,    1,      Z),  " &
     "320  (BC_4,  AB11,     observe_only, X),                        " &
     "319  (BC_2,  *,        control,      1),                        " &
     "318  (BC_1,  AB11,     output3,      X,    319,    1,      Z),  " &
     "317  (BC_4,  AB10,     observe_only, X),                        " &
     "316  (BC_1,  AB10,     output3,      X,    319,    1,      Z),  " &
     "315  (BC_4,  AB9,      observe_only, X),                        " &
     "314  (BC_1,  AB9,      output3,      X,    319,    1,      Z),  " &
     "313  (BC_4,  AB8,      observe_only, X),                        " &
     "312  (BC_1,  AB8,      output3,      X,    319,    1,      Z),  " &
     "311  (BC_4,  AB7,      observe_only, X),                        " &
     "310  (BC_2,  *,        control,      1),                        " &
     "309  (BC_1,  AB7,      output3,      X,    310,    1,      Z),  " &
     "308  (BC_4,  AB6,      observe_only, X),                        " &
     "307  (BC_2,  *,        control,      1),                        " &
     "306  (BC_1,  AB6,      output3,      X,    307,    1,      Z),  " &
     "305  (BC_4,  AB5,      observe_only, X),                        " &
     "304  (BC_4,  AB4,      observe_only, X),                        " &
     "303  (BC_4,  AB3,      observe_only, X),                        " &
     "302  (BC_4,  AB2,      observe_only, X),                        " &
     "301  (BC_4,  AB1,      observe_only, X),                        " &
     "300  (BC_4,  AB0,      observe_only, X),                        " &
     "299  (BC_4,  DB15,     observe_only, X),                        " &
     "298  (BC_2,  *,        control,      1),                        " &
     "297  (BC_1,  DB15,     output3,      X,    298,    1,      Z),  " &
     "296  (BC_4,  DB14,     observe_only, X),                        " &
     "295  (BC_1,  DB14,     output3,      X,    298,    1,      Z),  " &
     "294  (BC_4,  DB13,     observe_only, X),                        " &
     "293  (BC_2,  *,        control,      1),                        " &
     "292  (BC_1,  DB13,     output3,      X,    293,    1,      Z),  " &
     "291  (BC_4,  DB12,     observe_only, X),                        " &
     "290  (BC_1,  DB12,     output3,      X,    293,    1,      Z),  " &
     "289  (BC_4,  DB11,     observe_only, X),                        " &
     "288  (BC_1,  DB11,     output3,      X,    293,    1,      Z),  " &
     "287  (BC_4,  DB10,     observe_only, X),                        " &
     "286  (BC_1,  DB10,     output3,      X,    293,    1,      Z),  " &
     "285  (BC_4,  DB9,      observe_only, X),                        " &
     "284  (BC_2,  *,        control,      1),                        " &
     "283  (BC_1,  DB9,      output3,      X,    284,    1,      Z),  " &
     "282  (BC_4,  DB8,      observe_only, X),                        " &
     "281  (BC_1,  DB8,      output3,      X,    293,    1,      Z),  " &
     "280  (BC_4,  BUSCLK,   observe_only, X),                        " &
     "279  (BC_4,  DB7,      observe_only, X),                        " &
     "278  (BC_1,  DB7,      output3,      X,    298,    1,      Z),  " &
     "277  (BC_4,  DB6,      observe_only, X),                        " &
     "276  (BC_2,  *,        control,      1),                        " &
     "275  (BC_1,  DB6,      output3,      X,    276,    1,      Z),  " &
     "274  (BC_4,  DB5,      observe_only, X),                        " &
     "273  (BC_1,  DB5,      output3,      X,    276,    1,      Z),  " &
     "272  (BC_4,  DB4,      observe_only, X),                        " &
```

```
"271  (BC_1,  DB4,      output3,      X,    276,    1,      Z),  " &
"270  (BC_4,  DB3,      observe_only, X),                        " &
"269  (BC_1,  DB3,      output3,      X,    276,    1,      Z),  " &
"268  (BC_4,  DB2,      observe_only, X),                        " &
"267  (BC_1,  DB2,      output3,      X,    276,    1,      Z),  " &
"266  (BC_4,  DB1,      observe_only, X),                        " &
"265  (BC_2,  *,        control,      1),                        " &
"264  (BC_1,  DB1,      output3,      X,    265,    1,      Z),  " &
"263  (BC_4,  DB0,      observe_only, X),                        " &
"262  (BC_1,  DB0,      output3,      X,    265,    1,      Z),  " &
"261  (BC_4,  CSX,      observe_only, X),                        " &
"260  (BC_4,  MNRX,     observe_only, X),                        " &
"259  (BC_2,  *,        control,      1),                        " &
"258  (BC_1,  MNRX,     output3,      X,    259,    1,      Z),  " &
"257  (BC_4,  RDX,      observe_only, X),                        " &
"256  (BC_4,  WRX,      observe_only, X),                        " &
"255  (BC_4,  BE0X,     observe_only, X),                        " &
"254  (BC_4,  BE1X,     observe_only, X),                        " &
"253  (BC_2,  *,        control,      1),                        " &
"252  (BC_1,  BE1X,     output3,      X,    253,    1,      Z),  " &
"251  (BC_4,  BSX,      observe_only, X),                        " &
"250  (BC_2,  *,        control,      1),                        " &
"249  (BC_1,  BSX,      output3,      X,    250,    1,      Z),  " &
"248  (BC_4,  BURSTX,   observe_only, X),                        " &
"247  (BC_4,  BDIPX,    observe_only, X),                        " &
"246  (BC_4,  TEAX,     observe_only, X),                        " &
"245  (BC_2,  *,        control,      1),                        " &
"244  (BC_1,  TEAX,     output3,      X,    245,    1,      Z),  " &
"243  (BC_4,  WAITX,    observe_only, X),                        " &
"242  (BC_2,  *,        control,      1),                        " &
"241  (BC_1,  WAITX,    output3,      X,    242,    1,      Z),  " &
"240  (BC_2,  *,        control,      1),                        " &
"239  (BC_1,  IRQ,      output3,      X,    240,    1,      Z),  " &
"238  (BC_4,  RESETX,   observe_only, X),                        " &
"237  (BC_4,  FP1IO19,  observe_only, X),                        " &
"236  (BC_2,  *,        control,      1),                        " &
"235  (BC_1,  FP1IO19,  output3,      X,    236,    1,      Z),  " &
"234  (BC_4,  FP1IO18,  observe_only, X),                        " &
"233  (BC_2,  *,        control,      1),                        " &
"232  (BC_1,  FP1IO18,  output3,      X,    233,    1,      Z),  " &
"231  (BC_4,  FP1IO17,  observe_only, X),                        " &
"230  (BC_2,  *,        control,      1),                        " &
"229  (BC_1,  FP1IO17,  output3,      X,    230,    1,      Z),  " &
"228  (BC_4,  FP1IO14,  observe_only, X),                        " &
"227  (BC_2,  *,        control,      1),                        " &
"226  (BC_1,  FP1IO14,  output3,      X,    227,    1,      Z),  " &
"225  (BC_4,  FP1IO11,  observe_only, X),                        " &
"224  (BC_2,  *,        control,      1),                        " &
"223  (BC_1,  FP1IO11,  output3,      X,    224,    1,      Z),  " &
"222  (BC_4,  FP1IO22,  observe_only, X),                        " &
"221  (BC_2,  *,        control,      1),                        " &
"220  (BC_1,  FP1IO22,  output3,      X,    221,    1,      Z),  " &
"219  (BC_4,  FP1IO21,  observe_only, X),                        " &
"218  (BC_2,  *,        control,      1),                        " &
"217  (BC_1,  FP1IO21,  output3,      X,    218,    1,      Z),  " &
"216  (BC_4,  FP1IO20,  observe_only, X),                        " &
"215  (BC_2,  *,        control,      1),                        " &
"214  (BC_1,  FP1IO20,  output3,      X,    215,    1,      Z),  " &
"213  (BC_4,  FP1IO16,  observe_only, X),                        " &
"212  (BC_2,  *,        control,      1),                        " &
"211  (BC_1,  FP1IO16,  output3,      X,    212,    1,      Z),  " &
"210  (BC_4,  FP1IO15,  observe_only, X),                        " &
"209  (BC_2,  *,        control,      1),                        " &
"208  (BC_1,  FP1IO15,  output3,      X,    209,    1,      Z),  " &
"207  (BC_4,  FP1IO13,  observe_only, X),                        " &
"206  (BC_2,  *,        control,      1),                        " &
"205  (BC_1,  FP1IO13,  output3,      X,    206,    1,      Z),  " &
"204  (BC_4,  FP1IO23,  observe_only, X),                        " &
"203  (BC_2,  *,        control,      1),                        " &
"202  (BC_1,  FP1IO23,  output3,      X,    203,    1,      Z),  " &
"201  (BC_4,  FP1IO12,  observe_only, X),                        " &
"200  (BC_1,  FP1IO12,  output3,      X,    206,    1,      Z),  " &
"199  (BC_4,  FP1IO10,  observe_only, X),                        " &
"198  (BC_1,  FP1IO10,  output3,      X,    206,    1,      Z),  " &
"197  (BC_4,  FP1IO9,   observe_only, X),                        " &
"196  (BC_2,  *,        control,      1),                        " &
"195  (BC_1,  FP1IO9,   output3,      X,    196,    1,      Z),  " &
"194  (BC_4,  FP1IO8,   observe_only, X),                        " &
"193  (BC_1,  FP1IO8,   output3,      X,    206,    1,      Z),  " &
"192  (BC_4,  FP1IO7,   observe_only, X),                        " &
```

```
"191  (BC_1,  FP1IO7,   output3,      X,    206,    1,      Z),  " &
"190  (BC_4,  FP1IO6,   observe_only, X),                        " &
"189  (BC_2,  *,        control,      1),                        " &
"188  (BC_1,  FP1IO6,   output3,      X,    189,    1,      Z),  " &
"187  (BC_4,  FP1IO5,   observe_only, X),                        " &
"186  (BC_1,  FP1IO5,   output3,      X,    189,    1,      Z),  " &
"185  (BC_4,  FP1IO4,   observe_only, X),                        " &
"184  (BC_1,  FP1IO4,   output3,      X,    189,    1,      Z),  " &
"183  (BC_4,  FP1IO3,   observe_only, X),                        " &
"182  (BC_1,  FP1IO3,   output3,      X,    189,    1,      Z),  " &
"181  (BC_4,  FP1IO2,   observe_only, X),                        " &
"180  (BC_1,  FP1IO2,   output3,      X,    189,    1,      Z),  " &
"179  (BC_4,  FP1IO1,   observe_only, X),                        " &
"178  (BC_2,  *,        control,      1),                        " &
"177  (BC_1,  FP1IO1,   output3,      X,    178,    1,      Z),  " &
"176  (BC_4,  FP1IO0,   observe_only, X),                        " &
"175  (BC_1,  FP1IO0,   output3,      X,    178,    1,      Z),  " &
"174  (BC_1,  FP2IO27,  output2,      X),                        " &
"173  (BC_1,  FP2IO26,  output2,      X),                        " &
"172  (BC_1,  FP2IO25,  output2,      X),                        " &
"171  (BC_1,  FP2IO24,  output2,      X),                        " &
"170  (BC_4,  FP2IO23,  observe_only, X),                        " &
"169  (BC_2,  *,        control,      1),                        " &
"168  (BC_1,  FP2IO23,  output3,      X,    169,    1,      Z),  " &
"167  (BC_4,  FP2IO22,  observe_only, X),                        " &
"166  (BC_2,  *,        control,      1),                        " &
"165  (BC_1,  FP2IO22,  output3,      X,    166,    1,      Z),  " &
"164  (BC_4,  FP2IO21,  observe_only, X),                        " &
"163  (BC_2,  *,        control,      1),                        " &
"162  (BC_1,  FP2IO21,  output3,      X,    163,    1,      Z),  " &
"161  (BC_4,  FP2IO20,  observe_only, X),                        " &
"160  (BC_2,  *,        control,      1),                        " &
"159  (BC_1,  FP2IO20,  output3,      X,    160,    1,      Z),  " &
"158  (BC_4,  FP2IO19,  observe_only, X),                        " &
"157  (BC_2,  *,        control,      1),                        " &
"156  (BC_1,  FP2IO19,  output3,      X,    157,    1,      Z),  " &
"155  (BC_4,  FP2IO18,  observe_only, X),                        " &
"154  (BC_2,  *,        control,      1),                        " &
"153  (BC_1,  FP2IO18,  output3,      X,    154,    1,      Z),  " &
"152  (BC_4,  FP2IO17,  observe_only, X),                        " &
"151  (BC_2,  *,        control,      1),                        " &
"150  (BC_1,  FP2IO17,  output3,      X,    151,    1,      Z),  " &
"149  (BC_1,  FP2IO16,  output2,      X),                        " &
"148  (BC_1,  FP2IO15,  output2,      X),                        " &
"147  (BC_1,  FP2IO14,  output2,      X),                        " &
"146  (BC_1,  FP2IO13,  output2,      X),                        " &
"145  (BC_1,  FP2IO12,  output2,      X),                        " &
"144  (BC_1,  FP2IO11,  output2,      X),                        " &
"143  (BC_1,  FP2IO10,  output2,      X),                        " &
"142  (BC_1,  FP2IO9,   output2,      X),                        " &
"141  (BC_1,  FP2IO8,   output2,      X),                        " &
"140  (BC_1,  FP2IO7,   output2,      X),                        " &
"139  (BC_1,  FP2IO6,   output2,      X),                        " &
"138  (BC_1,  FP2IO5,   output2,      X),                        " &
"137  (BC_1,  FP2IO4,   output2,      X),                        " &
"136  (BC_1,  FP2IO3,   output2,      X),                        " &
"135  (BC_1,  FP2IO2,   output2,      X),                        " &
"134  (BC_1,  FP2IO1,   output2,      X),                        " &
"133  (BC_1,  FP2IO0,   output2,      X),                        " &
"132  (BC_4,  I2SWSO,   observe_only, X),                        " &
"131  (BC_2,  *,        control,      1),                        " &
"130  (BC_1,  I2SWSO,   output3,      X,    131,    1,      Z),  " &
"129  (BC_1,  I2SSDO,   output2,      X),                        " &
"128  (BC_4,  I2SCKO,   observe_only, X),                        " &
"127  (BC_1,  I2SCKO,   output3,      X,    131,    1,      Z),  " &
"126  (BC_1,  I2SMCLKO, output2,      X),                        " &
"125  (BC_1,  PWM1,     output2,      X),                        " &
"124  (BC_1,  PWM0,     output2,      X),                        " &
"123  (BC_1,  SPICS,    output2,      X),                        " &
"122  (BC_1,  SPICK,    output2,      X),                        " &
"121  (BC_4,  SPIDIO,   observe_only, X),                        " &
"120  (BC_2,  *,        control,      1),                        " &
"119  (BC_1,  SPIDIO,   output3,      X,    120,    1,      Z),  " &
"118  (BC_4,  CNF2,     observe_only, X),                        " &
"117  (BC_4,  CNF1,     observe_only, X),                        " &
"116  (BC_4,  CNF0,     observe_only, X),                        " &
"115  (BC_1,  MEMDQM3,  output2,      X),                        " &
"114  (BC_1,  MEMDQM2,  output2,      X),                        " &
"113  (BC_1,  MEMDQM1,  output2,      X),                        " &
"112  (BC_1,  MEMDQM0,  output2,      X),                        " &
```

```
"111  (BC_1,  MEMCKE,   output2,      X),                        " &
"110  (BC_1,  MEMCSX,   output2,      X),                        " &
"109  (BC_1,  MEMRASX,  output2,      X),                        " &
"108  (BC_1,  MEMCASX,  output2,      X),                        " &
"107  (BC_1,  MEMWEX,   output2,      X),                        " &
"106  (BC_1,  MEMBA1,   output2,      X),                        " &
"105  (BC_1,  MEMBA0,   output2,      X),                        " &
"104  (BC_1,  MEMA12,   output2,      X),                        " &
"103  (BC_1,  MEMA11,   output2,      X),                        " &
"102  (BC_1,  MEMA10,   output2,      X),                        " &
"101  (BC_1,  MEMA9,    output2,      X),                        " &
"100  (BC_1,  MEMA8,    output2,      X),                        " &
"99   (BC_4,  MEMDQ31,  observe_only, X),                        " &
"98   (BC_2,  *,        control,      1),                        " &
"97   (BC_1,  MEMDQ31,  output3,      X,    98,     1,      Z),  " &
"96   (BC_4,  MEMDQ15,  observe_only, X),                        " &
"95   (BC_1,  MEMDQ15,  output3,      X,    98,     1,      Z),  " &
"94   (BC_4,  MEMDQ30,  observe_only, X),                        " &
"93   (BC_2,  *,        control,      1),                        " &
"92   (BC_1,  MEMDQ30,  output3,      X,    93,     1,      Z),  " &
"91   (BC_4,  MEMDQ14,  observe_only, X),                        " &
"90   (BC_1,  MEMDQ14,  output3,      X,    93,     1,      Z),  " &
"89   (BC_4,  MEMDQ29,  observe_only, X),                        " &
"88   (BC_1,  MEMDQ29,  output3,      X,    93,     1,      Z),  " &
"87   (BC_4,  MEMDQ13,  observe_only, X),                        " &
"86   (BC_1,  MEMDQ13,  output3,      X,    93,     1,      Z),  " &
"85   (BC_4,  MEMDQ28,  observe_only, X),                        " &
"84   (BC_1,  MEMDQ28,  output3,      X,    98,     1,      Z),  " &
"83   (BC_4,  MEMDQ12,  observe_only, X),                        " &
"82   (BC_1,  MEMDQ12,  output3,      X,    93,     1,      Z),  " &
"81   (BC_4,  MEMDQ27,  observe_only, X),                        " &
"80   (BC_2,  *,        control,      1),                        " &
"79   (BC_1,  MEMDQ27,  output3,      X,    80,     1,      Z),  " &
"78   (BC_4,  MEMDQ11,  observe_only, X),                        " &
"77   (BC_1,  MEMDQ11,  output3,      X,    80,     1,      Z),  " &
"76   (BC_4,  MEMDQ26,  observe_only, X),                        " &
"75   (BC_1,  MEMDQ26,  output3,      X,    80,     1,      Z),  " &
"74   (BC_4,  MEMDQ10,  observe_only, X),                        " &
"73   (BC_1,  MEMDQ10,  output3,      X,    98,     1,      Z),  " &
"72   (BC_4,  MEMDQ25,  observe_only, X),                        " &
"71   (BC_1,  MEMDQ25,  output3,      X,    98,     1,      Z),  " &
"70   (BC_4,  MEMDQ9,   observe_only, X),                        " &
"69   (BC_2,  *,        control,      1),                        " &
"68   (BC_1,  MEMDQ9,   output3,      X,    69,     1,      Z),  " &
"67   (BC_4,  MEMDQ24,  observe_only, X),                        " &
"66   (BC_1,  MEMDQ24,  output3,      X,    80,     1,      Z),  " &
"65   (BC_4,  MEMDQ8,   observe_only, X),                        " &
"64   (BC_1,  MEMDQ8,   output3,      X,    69,     1,      Z),  " &
"63   (BC_4,  MEMDQ23,  observe_only, X),                        " &
"62   (BC_1,  MEMDQ23,  output3,      X,    80,     1,      Z),  " &
"61   (BC_4,  MEMDQ7,   observe_only, X),                        " &
"60   (BC_2,  *,        control,      1),                        " &
"59   (BC_1,  MEMDQ7,   output3,      X,    60,     1,      Z),  " &
"58   (BC_4,  MEMDQ22,  observe_only, X),                        " &
"57   (BC_1,  MEMDQ22,  output3,      X,    60,     1,      Z),  " &
"56   (BC_4,  MEMDQ6,   observe_only, X),                        " &
"55   (BC_1,  MEMDQ6,   output3,      X,    69,     1,      Z),  " &
"54   (BC_4,  MEMDQ21,  observe_only, X),                        " &
"53   (BC_1,  MEMDQ21,  output3,      X,    69,     1,      Z),  " &
"52   (BC_4,  MEMDQ5,   observe_only, X),                        " &
"51   (BC_1,  MEMDQ5,   output3,      X,    60,     1,      Z),  " &
"50   (BC_4,  MEMDQ20,  observe_only, X),                        " &
"49   (BC_1,  MEMDQ20,  output3,      X,    69,     1,      Z),  " &
"48   (BC_1,  MEMCLK,   output2,      X),                        " &
"47   (BC_4,  MEMDQ4,   observe_only, X),                        " &
"46   (BC_1,  MEMDQ4,   output3,      X,    60,     1,      Z),  " &
"45   (BC_4,  MEMDQ19,  observe_only, X),                        " &
"44   (BC_1,  MEMDQ19,  output3,      X,    60,     1,      Z),  " &
"43   (BC_4,  MEMDQ3,   observe_only, X),                        " &
"42   (BC_2,  *,        control,      1),                        " &
"41   (BC_1,  MEMDQ3,   output3,      X,    42,     1,      Z),  " &
"40   (BC_4,  MEMDQ18,  observe_only, X),                        " &
"39   (BC_1,  MEMDQ18,  output3,      X,    42,     1,      Z),  " &
"38   (BC_4,  MEMDQ2,   observe_only, X),                        " &
"37   (BC_2,  *,        control,      1),                        " &
"36   (BC_1,  MEMDQ2,   output3,      X,    37,     1,      Z),  " &
"35   (BC_4,  MEMDQ17,  observe_only, X),                        " &
"34   (BC_1,  MEMDQ17,  output3,      X,    42,     1,      Z),  " &
"33   (BC_4,  MEMDQ1,   observe_only, X),                        " &
"32   (BC_1,  MEMDQ1,   output3,      X,    42,     1,      Z),  " &
```

```
    "31   (BC_4,  MEMDQ16,  observe_only, X),                         " &
    "30   (BC_1,  MEMDQ16,  output3,      X,    42,     1,      Z),  " &
    "29   (BC_4,  MEMDQ0,   observe_only, X),                         " &
    "28   (BC_2,  *,        control,      1),                         " &
    "27   (BC_1,  MEMDQ0,   output3,      X,    28,     1,      Z),  " &
    "26   (BC_1,  MEMA7,    output2,      X),                         " &
    "25   (BC_1,  MEMA6,    output2,      X),                         " &
    "24   (BC_1,  MEMA5,    output2,      X),                         " &
    "23   (BC_1,  MEMA4,    output2,      X),                         " &
    "22   (BC_1,  MEMA3,    output2,      X),                         " &
    "21   (BC_1,  MEMA2,    output2,      X),                         " &
    "20   (BC_1,  MEMA1,    output2,      X),                         " &
    "19   (BC_1,  MEMA0,    output2,      X),                         " &
    "18   (BC_4,  I2CSCL,   observe_only, X),                         " &
    "17   (BC_2,  *,        control,      1),                         " &
    "16   (BC_1,  I2CSCL,   output3,      X,    17,     1,      Z),  " &
    "15   (BC_4,  I2CSDA,   observe_only, X),                         " &
    "14   (BC_2,  *,        control,      1),                         " &
    "13   (BC_1,  I2CSDA,   output3,      X,    14,     1,      Z),  " &
    "12   (BC_4,  CM1FIELD, observe_only, X),                         " &
    "11   (BC_4,  CM1VREF,  observe_only, X),                         " &
    "10   (BC_4,  CM1HREF,  observe_only, X),                         " &
    "9    (BC_4,  CM1D7,    observe_only, X),                         " &
    "8    (BC_4,  CM1D6,    observe_only, X),                         " &
    "7    (BC_4,  CM1D5,    observe_only, X),                         " &
    "6    (BC_1,  CM1CLKO,  output2,      X),                         " &
    "5    (BC_4,  CM1CLKI,  observe_only, X),                         " &
    "4    (BC_4,  CM1D4,    observe_only, X),                         " &
    "3    (BC_4,  CM1D3,    observe_only, X),                         " &
    "2    (BC_4,  CM1D2,    observe_only, X),                         " &
    "1    (BC_4,  CM1D1,    observe_only, X),                         " &
    "0    (BC_4,  CM1D0,    observe_only, X)                          ";

end s2d13515;
```

Chapter 27 Design Considerations

27.1 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.

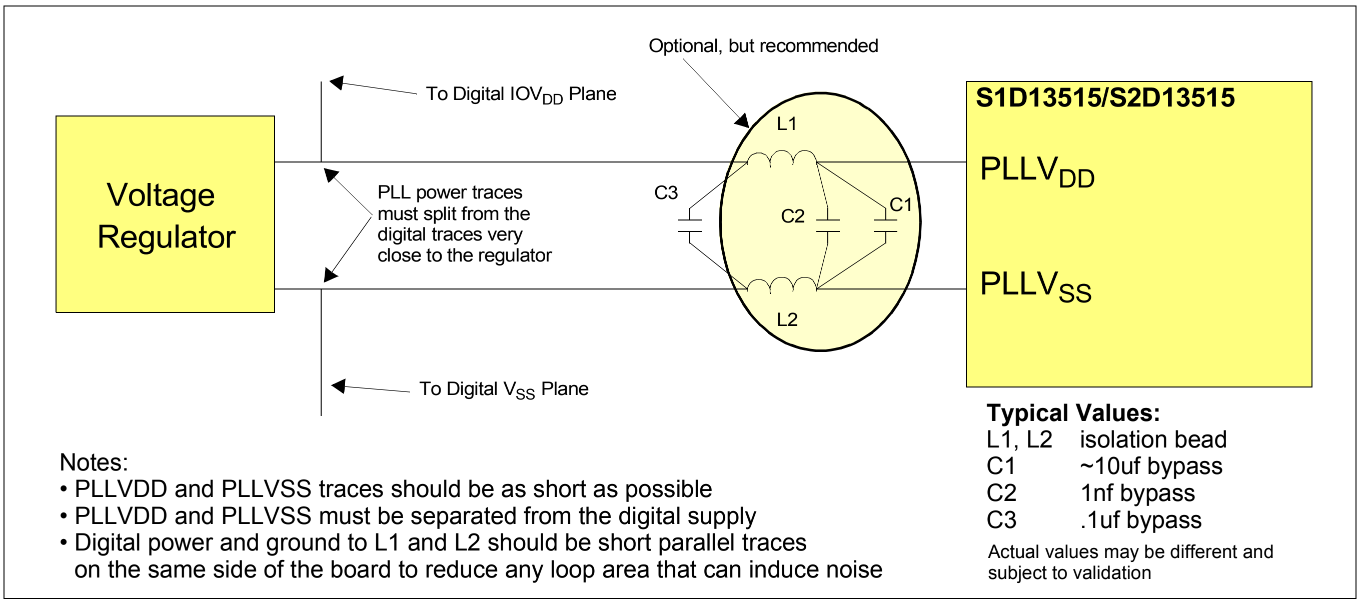


Figure 27-1: PLL Power Layout

- 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 (L1) becomes the analog central point for a ground star topology. None of the components connect directly to the analog ground pin of the S1D13515/S2D13515 ($PLLV_{SS}$) except for a single short trace from C2 to the $PLLV_{SS}$ 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 reflew 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.

Chapter 28 Mechanical Data

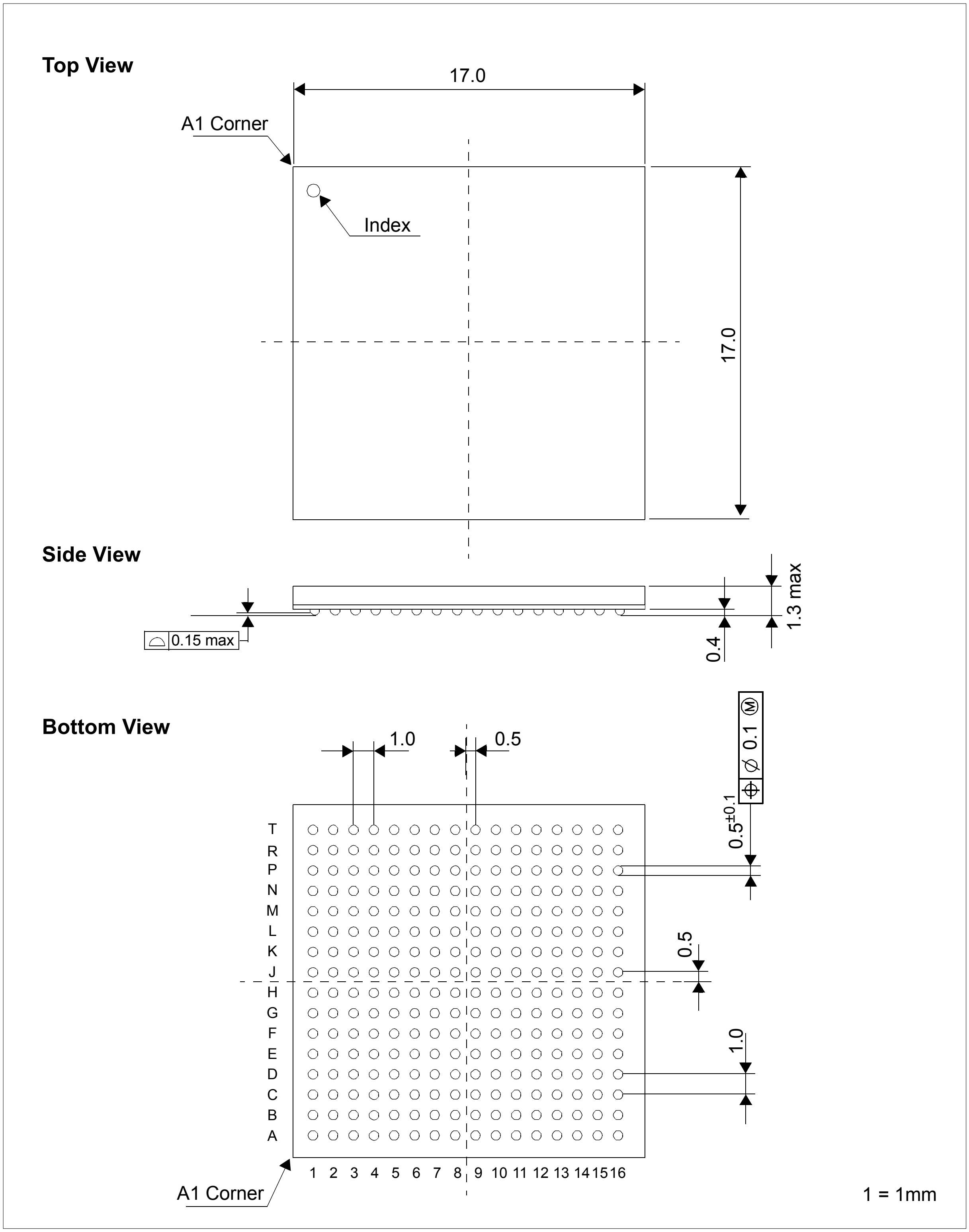
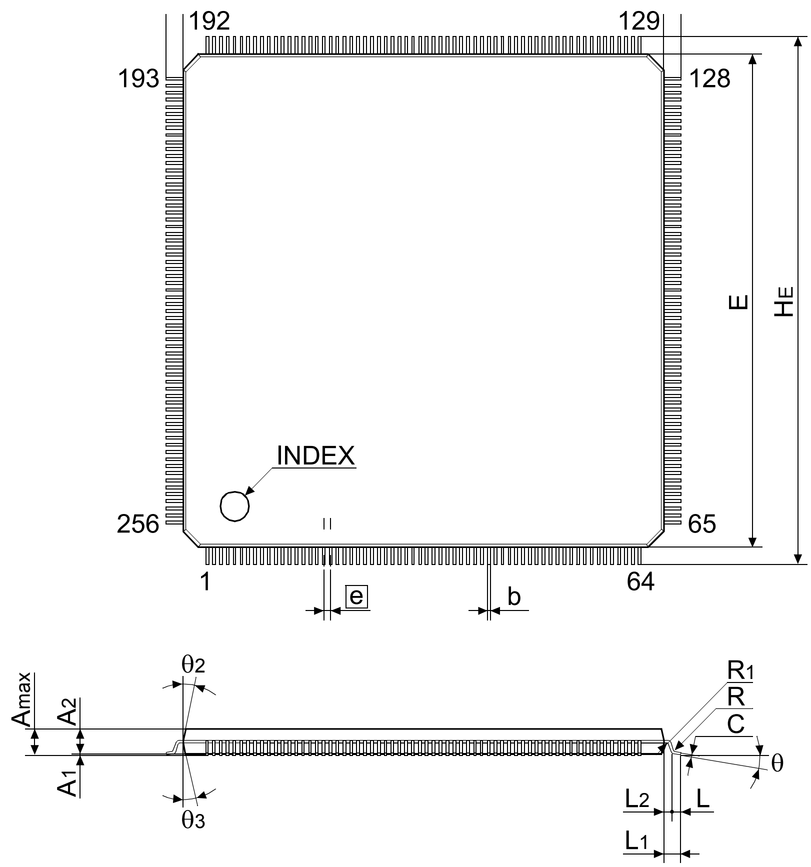
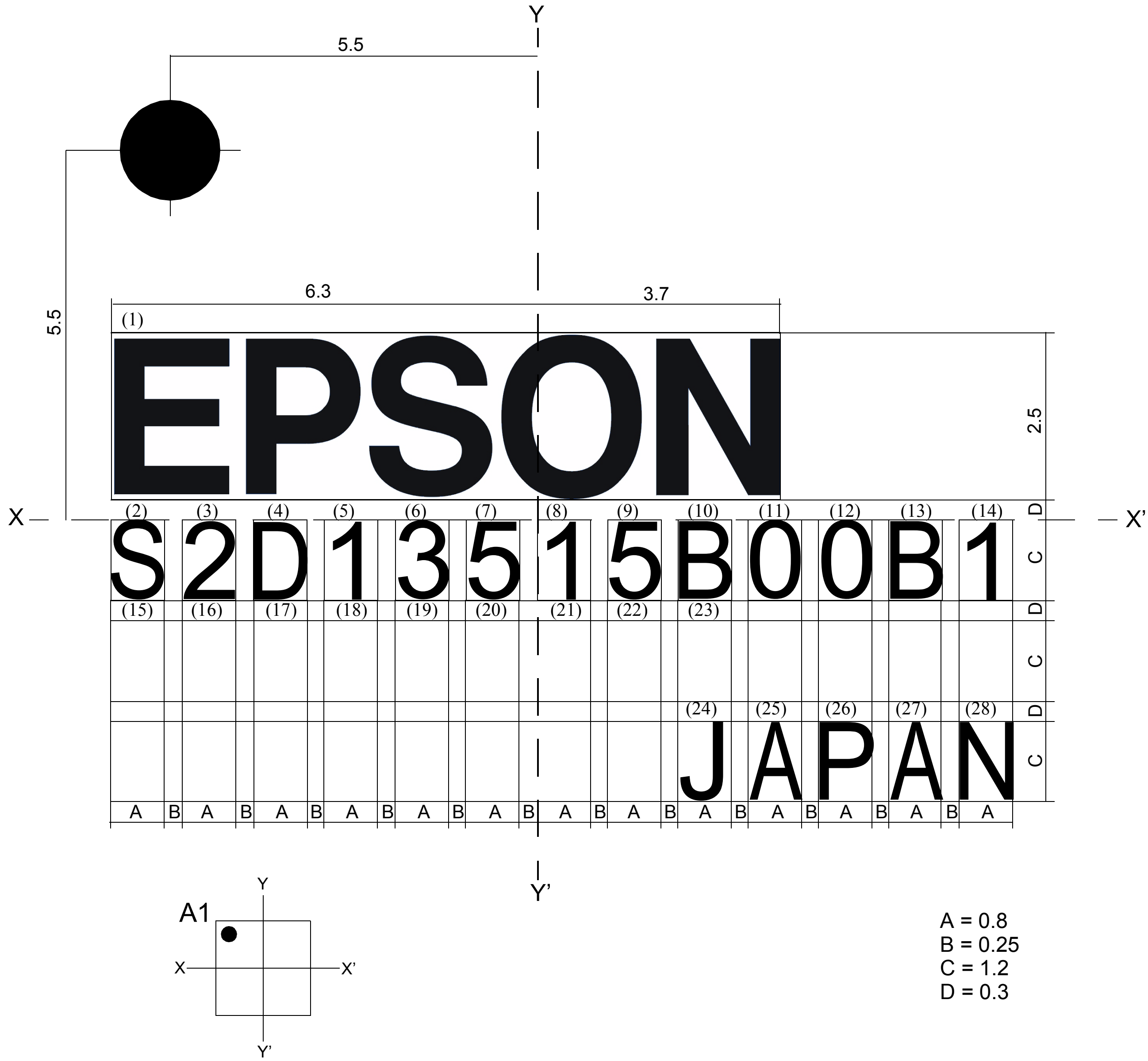


Figure 28-1: PBGA1U 256-pin Package



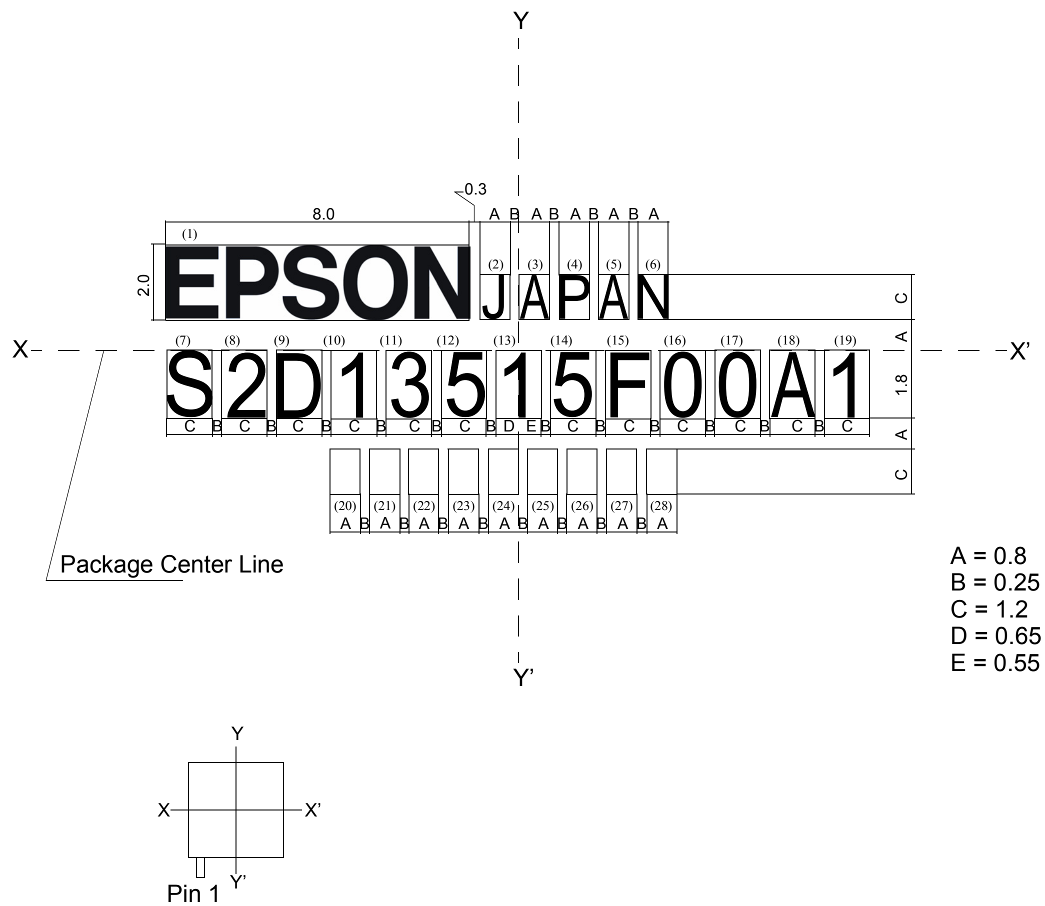
Symbol	Dimension in Millimeters		
	Min.	Nom.	Max.
E	27.9	28	28.1
D	27.9	28	28.1
A			1.7
A_1		0.1	
A_2	1.3	1.4	1.5
e		0.4	
b	0.13	0.16	0.21
C	0.1	0.125	0.175
θ	0°		10°
L	0.3	0.5	0.7
L_1		1	
L_2		0.5	
H_E	29.6	30	30.4
H_D	29.6	30	30.4
θ_2		12°	
θ_3		12°	
R		0.2	

Figure 28-2: QFP22 256-pin Package



Item	No.	Notes
Logo Specified	(1)	
Name of Goods	(2) ~ (14)	S2D13515B00B1 or S1D13515B00B1
Control Code	(15)	
Year of Manufacture	(16) ~ (17)	Last two numbers of A.D.
Week of Manufacture	(18) ~ (19)	Calendar week of the year
Lot No.	(20) ~ (23)	
[JAPAN]	(24) ~ (28)	

Figure 28-3: PBGA1U 256-pin Package Marking



Item	No.	Notes
Logo Specified	(1)	
[JAPAN]	(2) ~ (6)	
Name of Goods	(7) ~ (19)	S2D13515F00A1 or S1D13515F00A1
Control Code	(20)	
Year of Manufacture	(21) ~ (22)	Last two numbers of A.D.
Week of Manufacture	(23) ~ (24)	Calendar week of the year
Lot No.	(25) ~ (28)	

Figure 28-4: QFP22 256-pin Package Marking

Chapter 29 Change Record

X83A-A-001-01 Revision 1.8 - Issued: March 15, 2018

- updated Sales and Technical Support Section
- updated some formatting

X83A-A-001-01 Revision 1.7 - Issued: January 19, 2011

- chapter 19 Pulse Width Modulation (PWM) - add information for generating “errant-free” square waves

X83A-A-001-01 Revision 1.6 - Issued: September 07, 2010

- chapter 2.2 CPU Interfaces - add note “The S1D/S2D13515 supports Little Endian interface only” to section and “Little Endian configuration” to FreeScale MPC555 bullet
- chapter 5.4 Configuration Pins - add “Little Endian only” to MPC555 Host Interface in table 5-12, *Host Interface Configuration Summary*
- chapter 5.5 Host Interface Pin Mapping - add “Little Endian” to MPC555 heading in table 5-16, *Host Interface Pin Mapping 4*
- chapter 7.4.5 Direct/Indirect Freescale MPC555 (Non-burst Mode) - add note 2 “The S1D13515/S2D13515 does not support Big Endian...” after figure 7-15, *Direct/Indirect Freescale MPC555 Host Interface Write Timing (Non-burst Mode),* and figure 7-16, *Direct/Indirect Freescale MPC555 Host Interface Read Timing (Non-burst Mode)*
- chapter 7.4.6 Direct/Indirect Freescale MPC555 (Burst Mode) - add note 2 “The S1D13515/S2D13515 does not support Big Endian...” after figure 7-17, *Direct/Indirect Freescale MPC555 Host Interface Write Timing (Burst Mode),* and figure 7-18, *Direct/Indirect Freescale MPC555 Host Interface Read Timing (Burst Mode)*
- chapter 21.1 Overview - add note “The S1D/S2D13515 supports Little Endian interface only”
- chapter 21.9 MPC555 Interface - add “The S1D13515/S2D13515 does not support Big Endian...”

X83A-A-001-01 Revision 1.5 - Issued: August 26, 2009

- chapter 8 Memory Map - add notes 3, 4 and 5 below table 8-1
- chapter 10.4.19 DMA Controller Registers - add note “The DMAC controller must not be programmed for burst...”
- REG[3C0Ch] bit 6 - add note “If the DMA operation will span across SRAM banks...” to bit description
- REG[3C1Ch] bit 6 - add note “If the DMA operation will span across SRAM banks...” to bit description
- chapter 10.4.22 Sprite Registers - add note 2 “The Sprite Engine must use SDRAM memory space and may not use SRAM...”
- REG[5028h] ~ REG[502Bh] - correct typo, change memory addresses to 1xxx_xxxh in figure10-3 *Sprite Memory Map Example*

chapter 16.1 Sprite Data Path - correct typos, change DRAM to SDRAM in text and figure 16-1 *Sprite Data Path*

Change Record

X83A-A-001-01 Revision 1.4 - Issued: May 22, 2009

- chapter 5.3.7 Miscellaneous - in table 5-9 Miscellaneous Pin Descriptions, add “For normal operations, this pin must be connected to RESET#” to the TRST pin description
- chapter 6 D.C. Characteristics - in table 6-4, Electrical Characteristics for VDD = 3.3V typical, and table 6-5, Electrical Characteristics for VDD = 2.5V typical, change I_{DDS} Typ to “23”
- chapter 7.1.1 Input Clocks - in table 7-1, Clock Requirements for OSC/CLKI when used as Clock Input, change t_f and t_r Max to “0.2 TOSC”
- chapter 26.1 JTAG Pins - add “For normal operations, TRST must be tied to VSS or connected to RESET#” to the TRST description

X83A-A-001-01 Revision 1.3 - Issued: April 28, 2009

- changes from the previous revision are highlighted in Red
- chapter 6 D.C. Characteristics - in table 6-3 Recommended Operating Conditions 2, change H VDD-SD Min to 3.0, Typ to 3.3, and Max to 3.6

X83A-A-001-01 Revision 1.2 - Issued: March 30, 2009

- changes from the previous revision are highlighted in Red
- globally add S1D13515 information
- section 2.4 Display Features - in first indented bullet under “• Four input window sources can be stored in SDRAM...” change “/32 bpp” to “/24 bpp”
- section 7.1.2 Internal Clocks - add note “For XGA 1024x768 panel support, the DRAMCLK must be 100MHz”
- section 7.6 Panel Interface Timing - add note “For XGA 1024x768 panel support, only single panel, single window...”
- section 13 Display Subsystem - add note “For XGA 1024x768 panel support, only single panel, single window...”
- changes to International Sales Operations page - office changes and address changes

X83A-A-001-01 Revision 1.1 - Issued: November 05, 2008

- changes from the previous revision are highlighted in Red
- section 14.9 I2S Typical Operation Flow - add this section
- section 16.6 Sprite Programming Flow - add this section
- section 21.13 Initialization Examples - add this section
- section 23.6 Keypad Operation Flow - add this section
- section 24.4 Timer Operation Flow - add this section
- section 25.4 SPI Interface Operation Flow - add this section

X83A-A-001-01 Revision 1.0 - Issued: September 30, 2008

- changes from the previous revision are highlighted in Red

- section 7.6.8 EID Double Screen Panel Timing (TCON Enabled) - in figures 7-44, EID Double Screen Panel Horizontal Timing, and 7-45, EID Double Screen Panel Vertical Timing, and tables 7-52, EID Double Screen Panel Horizontal Timing, and 7-53, EID Double Screen Panel Vertical Timing, change “Conversion” to “Inversion”
- REG[001Ch] bit 6 - rewrite note 1
- REG[003Ch] bit 0 - rewrite note 4, add note 5, rewrite note 7
- REG[0085h] bit 4 - reserve this bit
- REG[008Ah] bit 0 - delete “To reset the S2D13515, write 1b, then...” from bit description
- REG[0200h] bits 3-0 - rewrite note in bit description
- REG[090Ch] - change default register value to 40h
- REG[0D09h] - reserve bits 1 and 2
- REG[0D0Eh] - reserve bits 0 and 1
- REG[0D49h] - reserve bits 1 and 2
- REG[0D4Eh] - reserve bits 0 and 1
- REG[3C40h] bit 0 - rewrite notes in bit description
- REG[3C44h] bits 4-0 - change value which these bits must be set to 14h
- chapter 11 Operating Configurations and States - in second paragraph, remove “(the reset vector address is...)” and change “Host software can enable/disable the C33PE processor” to “Host software can hold the C33PE processor in reset”
- section 13.2.2 Blending Engine - add figures 13-6 Blend Mode 0 Display Path, 13-7 Blend Mode 1 Display Path, figure 13-8 Blend Mode 2 Display Path, and figure 13-9 Blend Mode 3 Display Path
- section 13.2.4 Image Fetcher - move this section to after 13.2.6 Warp Writeback (now section 13.2.5)
- section 13.2.5 Warp Writeback - rewrite first paragraph
- section 13.3 Alpha-Blending for OSD Layer - move section to be part of section 13.2.2.
- section 14.1 Overview of Operation - add note “It is strongly recommended for performance reasons to locate the I2S DMA buffers...”
- section 16.5 Sprite Display Orientation and Positioning - add note “Rotation is not supported for Sprite #0.”
- section 17.1 SDRAM Device Types - add note “32-bit data bus is highly recommended to avoid...” in two places (in body text and after table)
- section 17.4 Self-Refresh Mode - add note “Before the SDRAM is placed in self-refresh mode all accesses...”
- chapter 18 SDRAM Read/Write Buffer - delete sentence “This leads to inefficient access by Host...” from first paragraph, add note “The SDRAM read/write buffer can also be used by...”
- chapter 19 Pulse Width Modulation (PWM) - add note “The PWM1/2 should only be programmed when...”
- chapter 21.9 MPC555 Interface - from note 4, delete “In 16-bit Indirect mode, burst access, the maximum number of 16-bit words transferred (burst length) is 3 because” and remove bulleted text “Burst Length = 2” and “Burst Length = 3”

- section 21.10 SPI Host Interface - add AB5 to table 21-9, SPI Host Interface Signals, and add paragraph “The SPI host module requires a valid clock...”
- section 21.11 I2C Host Interface - add AB5 to table 21-11, I2C Host Interface Signals, and add paragraph “The I2C host module requires a valid clock...”
- section 21.12.1 Direct Mode - add note to figure 21-13 Host Interface Register Space
- section 21.12.2 Indirect Mode - add “directly” and “Additionally, SDRAM can be accessed using...” to first paragraph
- section 22.2.2 RGB Streaming Input Interface - in the first line of paragraph one add “2 stream”
- section 22.3 Camera Input Interface - add “UV” to REG[0D06h]/REG[0D46h] bit 5 bullet, remove references to ITU-R BT.565, and remove “Interlaced1...” REG[0D30h]/REG[0D70h] bullet
- section 22.5 YUV-to-RGB Converter - remove bullets “REG[0D22h]/REG[0D62h] is the YRC...” and “REG[0D28h]/REG[0D68h] is the...”
- Chapter 24 Timers - add this chapter including Watchdog timer, Timer0 and Timer 1
- section 25.2 IO Pins for SPI Interface - add note “SPICS# is asserted automatically when using SPI Flash read...”
- section 25.3.2 SPI Flash Control Register - add notes 1 and 2
- section 25.3.3 SPI Flash Data Control Register - add note “SPDIO is asserted automatically when using SPI Flash read...”
- chapter 26 JTAG - add this section

X83A-A-001-00 Revision 0.09 - Issued: August 28, 2008

- changes from the previous revision are highlighted in Red
- section 2.2 CPU Interfaces - remove “Host Bus Clock: TBD”
- section 5.3 Pin Descriptions - changes to cell descriptions table and update all cell types in tables
- section 5.3.1 Host Interface - change BE1# pin Description, change to read “...For the Intel 80 Type 2 Indirect 8-bit Host Interface...” and change IRQ power to “HIOVDD”
- section 5.3.7 Miscellaneous - change TESTEN pin Description, change to read “...must be connected to VSS...”
- section 5.3.7 Miscellaneous - change RESET# pin Power to “HIOVDD” from “IOVDD”
- section 5.3.8 Power and Ground - for OSCVDD add “OSCVDD must be the same...”
- section 5.4 Configuration Pins - in table 5-12, Host Interface Configuration Summary, reserve the following settings; CNF[6:1]= 010000b, CNF[7:1]= 0110000b, CNF[7:1]= 1110000b, CNF[6:1]= 010010b, CNF[7:1]= 1110010b, CNF[6:1]= 010100b, CNF[7:1]= 0110100b, and CNF[7:1]= 1110100b
- section 5.5 Host Interface Pin Mapping - tables 5-13 through 5-16, remove reserved pin mappings and re-arrange table contents
- section 5.5 Host Interface Pin Mapping - in table 5-13, Host Interface Pin Mapping 1, change Intel80 Type2 8-bit Indirect BE1# pin to “1”
- section 5.5 Host Interface Pin Mapping - add note “The I2C slave address configuration from DB[6:0] is latched...” after table 5-15, Host Interface Pin Mapping 3

- section 5.6 LCD/Camera2 Pin Mapping -table 5-17, add a note 3 for FPIO19 (8-bit Camera REG[0D46h] bit 2 = 0b column) “GPIO7 is not available...”
- section 5.6 LCD/Camera2 Pin Mapping - table 5-18, add a note 2 “When LCD2 is an EID Doublescreen...”
- chapter 6 D.C. Characteristics - for table 6-2, OSC VDD add note
- section 7.1.1 Input Clocks - remove f_{BUSCLK} and T_{BUSCLK} from table 7-1 Clock Requirements for OSC/CLKI when used as Clock Input
- section 7.1.2 Internal Clocks - change the max value of $f_{SDRAMCLK}$ and f_{SYSCLK} to 100 and 50 respectively
- section 7.2.2 Power-On Sequence - in table 7-5 change t2 min to “55”
- section 7.3 RESET# Timing - add note 2 “When the OSC is used to supply clock source...”
- section 7.4.1 Direct/Indirect Intel 80 Type 1 - add note “For Indirect Intel 80 Type #1 8-bit...” after figures 7-7 and 7-8, and add asynchronous register access timings to tables 7-8 and 7-9
- section 7.4.2 Direct/Indirect Intel 80 Type 2 - add note “For Indirect Intel 80 Type #1 8-bit...” after figures 7-9 and 7-10, and add asynchronous register access timings to tables 7-12 and 7-13
- section 7.4.3 Direct Marvell PXA3xx VLIO - add asynchronous register access timings to tables 7-10 and 7-11
- section 7.4.4 Direct/Indirect Renesas SH4 - add note “For Indirect SH4 8-bit, the WE1# and...” after figures 7-13 and 7-14, change table 7-14 and 7-15 t21 min to “0” and add note 2 “At the end of the read cycle...” after tables
- section 7.4.5 Direct/Indirect Freescale MPC555 (Non-burst Mode) - changes to note in figures 7-15 and 7-16, and add note “For Indirect MPC555, the TSIZ0 pin...” after figures
- section 7.4.6 Direct/Indirect Freescale MPC555 (Burst Mode) - changes to note in figures 7-17 and 7-18, and add note “For Indirect MPC555, the TSIZ0 pin...” after figures
- section 7.4.7 Direct/Indirect TI TSM470 (Non-burst Mode) - changes to note in figures 7-19 and 7-20, and add note “For Indirect TI TMS470, the UB#...” after figures
- section 7.4.8 Direct/Indirect TI TSM470 (Burst Mode) - changes to note in figures 7-21 and 7-22, and add note “For Indirect TI TMS470, the UB#...” after figures
- section 7.4.9 Direct/Indirect NEC V850 Type 1 - add note “For Indirect NECV850 Type #1 8-bit...” after figures 7-23 and 7-24
- section 7.4.10 Direct/Indirect NEC V850 Type 2 - add note “For Indirect NECV850 Type #2 8-bit...” after figures 7-25 and 7-26
- section 7.5.1 SPI - add text “The SPI host module requires a valid clock selection...”, change t2 units to ClkSPI and change note 1, and add note 2 “The user must use a HSCK...”
- section 7.5.2 I2C - add text “The I2C host module requires a valid clock selection...” and “The user must select a ClkI2C...”
- section 7.6.2 ND-TFD 8-Bit Serial Interface Timing - change t8 typical value to “Note 2” and add note 2 “This result is software dependent...” to table 7-36, ND-TFD 8-Bit Serial Interface Timing for LCD1(FP1IO*), and table 7-37, ND-TFD 8-Bit Serial Interface Timing for LCD2(FP2IO*)
- section 7.6.3 ND-TFD 9-Bit Serial Interface Timing - change t8 typical value to “Note 2” and add note 2 “This result is software dependent...” to table 7-38, ND-TFD 9-Bit Serial Interface Timing for LCD1(FP1IO*), and table 7-39, ND-TFD 9-Bit Serial Interface Timing for LCD2(FP2IO*)

- section 7.6.5 uWIRE Serial Interface Timing - change t1 typical value to “1.5” in both tables
- section 7.6.6 24-Bit Serial Interface Timing - in figure 7-36, 24-Bit Serial Interface Timing, change PHA to “0”, POL to “0”, and in the note change REG[4016h] / REG[4034h] bits 1-0 value to “00b”
- section 7.6.7 Sharp DualView Panel Timing - add figure 7-40 Required External VCOMB Logic
- section 7.6.8 EID Double Screen Panel Timing (TCON Enabled) - add note “When using the EID Double Screen Panel with TCON enabled...”
- section 7.6.8 EID Double Screen Panel Timing (TCON Enabled) -remove TLEDON from figure 7-41 and table 7-49 “EID Double Screen Panel LED_DIM_OUT Timing”, update figure 7-42 and table 7-50 “EID Double Screen Panel Start-Up Control Signals Timing” with new figure and table, update figure 7-43 and table 7-51 “EID Double Screen Panel Shut-Down Control Signals Timing” with new figure and table, update figure 7-44 and table 7-52 “EID Double Screen Panel Horizontal Timing” with new figure and table, update figure 7-45 and table 7-53 “EID Double Screen Panel Vertical Timing” with new figure and table, and add notes after table 7-53
- section 7.7 Camera Interface Timing - add note 3 “For RGB input streaming mode...”
- section 7.8 SDRAM Interface Timing - changes to the figure and table
- section 7.10 Keypad Interface Timing - add “Filter Disabled” and “Filter Enabled” to figure 7-51 Keypad Interface Base Timing and remove “case” from note 1, replace note after figure 7-52 Keypad Interface Timing, and replace figure 7-53 Keypad Glitch Filter Input Timing
- REG[000Ch] ~ REG[000Fh] - reserve these registers
- REG[0010h] ~ REG[0013h] - change the bits to 31:10, add “on a 1K boundary” and “REG[0011h] bits 1-0 and REG[0010h] bits 7-0 are always 0” to register text, add note “SRAM region 0400_0200h ~ 0400_0D28h is cleared...”
- REG[001Ch] bit 7 - reserve this bit
- REG[001Ch] bit 6 - add note 1 “The C33 should be disabled before...”
- REG[001Ch] bit 6 - add note 2 “For minimum current consumption of the C33...”
- REG[003Ch] bit 0 - add note 2 “To achieve the lowest power consumption...”, add note 3 “Before entering powersave mode, the I2S Audio...” and note 4 “The C33 must be placed in HALT or SLEEP mode (through instruction code), or disabled...”, note 5 “After exiting powersave mode, the DRAM controller must be re-initialized by...” and note 6 “After exiting powersave mode, Note #4 must be met before the C33...”
- REG[0061h] bit 4 - rename bit and rewrite description
- REG[0063h] bit 4 - rename bit and rewrite description
- REG[0084h] bit 0 - rewrite bit description
- REG[008Ah] bit 0 - add “...” to bit description
- REG[00A8h] ~ REG[00ABh] - add “See Chapter 8...” to bit description
- section 10.4.3 Bit Per Pixel Converter Configuration Registers - add “See Chapter 12, “Bit-Per-Pixel Converter Functional Description” on page...” to first paragraph
- REG[0104h] bit 0 - add note 2 “The I2S Audio Interface must be disabled...”
- REG[010Fh] bit 7 - add note “The I2S Audio Interface must be disabled...”

- REG[0148h] ~ REG[014Bh] - add note “When the I2S Audio DMA Buffers are configured for...”
- REG[0152h] ~ REG[0153h] - change equation and add “Bits 1-0 of REG[0152h] should always be programmed to 00b” to bit description
- REG[0182h] ~ REG[0183h] - change default register values to XXh
- REG[0186h] bit 5 - add a note “GPIO7 is not available...”
- REG[01C0h] bit 1 - rewrite bit description
- REG[01C4h] ~ REG[01C7h] - add “and determine if a Keypad Interrupt occurs in REG[0A02h] bit 4”
- REG[01C8h] ~ REG[01CBh] - add note 2 “When a Keypad Input Polarity bit is changed from 1b to 0b...”
- REG[01CCh] ~ REG[01CEh] - rewrite bit description
- REG[01D0h] ~ REG[01D3h] - rename to “REG[01Dxh] Keypad Interrupt Raw Status/Clear” registers and changed the description to “These bits indicate the raw status of the corresponding Keypad Interrupt, regardless of whether or not the corresponding Keypad Interrupt is enabled (see REG[01C4h] ~ REG[01C7h])...”
- REG[01D0h] ~ REG[01D3h] - replace the “For Reads” portion of the bit description, add “then 0b” to writing a 1b, and add note
- REG[01D6h] - rewrite bit description
- REG[0200h] bits 3-0 - add note “When bits [3:0] are changed while PWM1 or PWM2 is active...”
- section 10.4.9 SDRAM Read/Write Buffer Registers - add “The SDRAM Buffers are 128 bytes...” to description
- REG[024Ch] ~ REG[024Dh] - add “When the host interface is 16-bit and both byte and...” to the bit description
- REG[025Ch] ~ REG[025Dh] - add “When the host interface is 16-bit and both byte and...” to the bit description
- REG0264h] ~ REG[0267h] - add note “These bits are updated at the end of each...”
- REG[0300h] ~ REG[037Fh] - add “When the host interface is 16-bit and both byte and...” to the bit description and replace note with “These registers should not be used when the SPI host interface is...”
- REG[0380h] ~ REG[03FFh] - add “When the host interface is 16-bit and both byte and...” to the bit description and replace note with “These registers should not be used when the SPI host interface is...”
- REG[0400h] bit 2 - reserve this bit
- REG[0402h] bit 3 - reserve this bit
- REG[0404h] bit 3 - reserve this bit
- REG[0406h] bit 3 - reserve this bit
- REG[0434h] ~ REG[0435h] - add “The X offset supports both positive and negative...” to bit description
- REG[0436h] ~ REG[0437h] - add “The Y offset supports both positive and negative...” to bit description
- REG[0940h] bit 1 - add “the image width and virtual image width” to note
- REG[0950h] ~ REG[0951h] - add note “For tiled frame mode, the image width...”
- REG[0954h] ~ REG[0955h] - add note 2 “For tiled frame mode, the image virtual width...”
- REG[0960h] bit 1 - add “the image width and virtual image width” to note

- REG[0970h] ~ REG[0971h] - add note “For tiled frame mode, the image width...”
- REG[0974h] ~ REG[0975h] - add note 2 “For tiled frame mode, the image virtual width...”
- REG[0980h] bit 1 - add “the image width and virtual image width” to note
- REG[0990h] ~ REG[0991h] - add note “For tiled frame mode, the image width...”
- REG[0994h] ~ REG[0995h] - add note 2 “For tiled frame mode, the image virtual width...”
- REG[0B04h] bit 6 - reserved this bit
- REG[09CAh] bit 6 - reserve this bit
- REG[09DBh] bits 6-4 - in table reserve setting of 001b
- REG[0A04h] bit 5 - add note “If this interrupt is enabled (REG[0A0Ah] bit 5 = 1b) before the OSD window...”
- REG[0A04h] bit 4 - add note “If this interrupt is enabled (REG[0A0Ah] bit 4 = 1b) before the AUX window...”
- REG[0A08h] - change default register value to 80h
- REG[0A0Ch] bit 6 - rewrite bit description
- REG[0A20h] - change default register value to 10h
- REG[0A21h] - change default register value to 0Fh
- REG[0A22h] - change default register value to 11h
- REG[0A23h] - change default register value to 01h
- REG[0A24h] - change default register value to 12h
- REG[0A25h] - change default register value to 01h
- REG[0A26h] - change default register value to 13h
- REG[0A27h] - change default register value to 01h
- REG[0A28h] - change default register value to 14h
- REG[0A29h] - change default register value to 01h
- REG[0A2Ah] - change default register value to 15h
- REG[0A2Bh] - change default register value to 0Fh
- REG[0A2Ch] - change default register value to 16h
- REG[0A2Dh] - change default register value to 0Dh
- REG[0A2Eh] - change default register value to 17h
- REG[0A2Fh] - change default register value to 0Ch
- REG[0A42h] - change default register value to E1h
- REG[0A43h] - change default register value to 80h
- REG[0A80h] - change default register value to 24h
- REG[0A84h] - change default register value to 01h

- REG[0A88h] - change default register value to E8h
- REG[0A89h] - change default register value to 03h
- REG[0A8Ah] - in note change “512” to 8192”
- REG[0B00h] - change default register value to FFh
- REG[0B04h] - change default register value to 11h
- REG[0B04h] bits 5-3 - add note “For odd SPI clock divides the SPICLK output...”
- REG[0D02h] bit 7 - add note “For SPI 2 Stream Mode...”
- REG[0D06h] bit 7 - add note “When ITU-R BT656 mode is enabled...”
- REG[0D06h] bits 2-1 - add note “For SPI 2 Stream Mode...”
- REG[0D08h] bits 2-0 - reserve these bits
- REG[0D30h] bits 1-0 - in table, reserve 01b setting
- REG[0D46h] bit 7 - add note “When ITU-R BT656 mode is enabled...”
- REG[0D48h] bits 2-0 - reserve these bits
- REG[3C40h] bit 6 - add “burst READ” to bit description
- REG[3C40h] bit 0 - add notes 1 through 3
- REG[4001h] bit 7 - add note “PCLK Polarity Select does not affect the polarity of...”
- REG[4018h] bit 0 - add “When the LCD interface is disabled...” to note
- REG[4019h] bit 2 - reserve this bit
- REG[4036h] bit 0 - add “When the LCD interface is disabled...” to note
- REG[4037h] bit 2 - reserve this bit
- REG[4040h] bit 0 - add note “When LCD2 is an EID Doublescreen with...”
- REG[4060h] bit 2 - add note “When LCD1 powersave mode is enabled...”
- REG[4070h] bit 2 - add note “When LCD2 powersave mode is enabled...”
- REG[4073h] bits 7-6 - reserve these bits
- REG[4078h] ~ REG[407Fh] - reserve these registers
- REG[40A0h] - change register name and rewrite bit description
- REG[40A2h] - add this register
- REG[40A3h] - add this register
- REG[5001h] bit 7 - add note “The Sprite Engine must be idle...”
- REG[5002h] bit 7 - rename bit
- REG[5003h] bits 2-0 - reserve these bits
- REG[5020h] ~ REG[5023h] - rewrite bit description

- REG[5024h] ~ REG[5027h] - rewrite bit description
- section 10.4.23 Sprite Memory Based Registers - in figure change address of SDRAM and SDRAM Based registers to “1xxx_xxxxh” from “0xxx_xxxxh” and reserve SDRAM[**01Bh] ~ SDRAM[**01Fh]
- SDRAM[**000h] bit 0 - add note “Sprite #0 is used as the background sprite and must...”
- SDRAM[**004h] ~ SDRAM[**007h] - add “These bits must be set such that...” to the bit description
- SDRAM[**008h] ~ SDRAM[**00Bh] - add “These bits must be set such that...” to the bit description
- SDRAM[**00Ch] ~ SDRAM[**00Dh] - add note “SDRAM[**00Dh] bits 7-2 and SDRAM[**00Dh] bits 1-0, SDRAM[**00Ch] bits 7-0 together form...”
- SDRAM[**018h] ~ SDRAM[**019h] - add note “Sprite #0 must not have...” a
- SDRAM[**01Ah] bits 1-0 - update descriptions in table
- section 11.1 Hard Reset - add note “The TESTEN pin must be connected to VSS for normal operation” after table 11-1 S2D13515 Hard Reset Pin States for Signals Which Are Not Part of Host Interface
- section 11.1 Hard Reset - add note “1. For the Intel 80 Type 2 Indirect 8-bit interface...” after table 11-2 S2D13515 Hard Reset Pin States for Host Interface 2
- section 11.1 Hard Reset - in table 11-2. 11-3, and 11-4, change TEA# PU/D to “PD” for all Host Interfaces
- section 11.1 Hard Reset - in table 11-2 S2D13515 Hard Reset Pin States for Host Interface 1, change NEC V850 Type #2 8-bit Indirect BUSCLK PU/D to “Z”
- section 11.1 Hard Reset - add note “2. For the Intel 80 and VEC V850 Type 1 Indirect 16-bit interfaces...” after table 11-2 S2D13515 Hard Reset Pin States for Host Interface 2
- section 11.1 Hard Reset - in table 11-4 S2D13515 Hard Reset Pin States for Host Interface 3, change SPI1 and SPI2 AB6 PU/D to “1/PD”
- section 11.1 Hard Reset - in table 11-4 S2D13515 Hard Reset Pin States for Host Interface 3, change MPC555 16-bit Indirect, TI TMS470 16-bit Indirect and I2C AB6 PU/D to “1/PD”
- section 13.1 Black Diagram - remove “can either go to the LCD Panel Interface (to LCD1) or” from paragraph starting “The Warp submodule reads frames from SDRAM...”, remove “Image Fetcher is mainly used in the case where the Warp cannot keep with the frame / refresh rate of the panel if it is connected to the LCD Panel Interface. In this case, the” from the next paragraph after, and remove “Warp OUTMODE” from figure 13-1
- section 13.2.3 Warp Engine - rewrite first paragraph by removing text “fed to the LCD Panel Interface directly or” and “In an application where there is no need to combine different...”, under Warp Programming, remove bulleted text “The Warp engine’s input image can be set to...”
- section 13.2.6 Warp Writeback - remove bulleted text “The Warp Writeback block is turned on...”
- section 13.6 Gamma LUT - add this section
- chapter 16 Sprite Engine - change last sentence in the first bullet to “Sprite #0 is defined as the background sprite image” and change references to “DRAM” to “SDRAM”
- section 16.2 8 Sprite Support with Z-ordering Transparency - rewrite note
- section 16.3 8 Sprite Support with Z-ordering Alpha-Blending - rewrite note

- section 16.4 Reference Point Based 90°, 180° and 270° Rotation + Mirror - correct the orientation of “180°” and “180° + Mirror” in figure 16-7
- section 16.5 Sprite Display Orientation and Positioning - in figure 16-8, replace “Y position” with “F”
- chapter 20 General-Purpose IO Pins - add note for GPIO7 “GPIO7 is not available...”
- section 21.11 I2C Host Interface - add table 21-12 I2C Slave Addresses and notes 1 and 2 following the table
- section 23.1 Keypad Pin Mapping - add note “GPIO7 is not available...”
- section 25.3.2 SPIFlash Control Register - remove bulleted text “Bit 6 is the SPI Flash Read Command Select bit...”

X83A-A-001-00 Revision 0.08 - Issued: February 25, 2008

- all changes from the last revision are highlighted in Red
- section 5.2, added PBGA pin mapping diagram
- section 5.3, added PBGA pin #’s for all pin descriptions
- section 5.3.4, added comment that SCL and SDA pins should be left unconnected if I2C is not used
- section 5.3.5, added comment that SPIDIO pin should be left unconnected if the SPI Flash is not used
- section 5.3.6, added comment that WSIIO and SCKIO pins should be left unconnected if I2S is not used
- section 7, added AC Timing Conditions
- section 7.1.1 Input Clocks - in table 7-2, change fOSC min and max from TBD to 20 and 40 respectively
- section 7.2, updated the Power Supply Sequence timing information
- section 7.4.4, updated the SH4 Write and Read Timing figures and tables with new timing “t21” to clarify RDY# state after writes are completed and read data is ready
- section 7.4.9 and 7.4.10, updated the NEC V850 Type 1 and Type 2 Read/Write Timing tables with new min/max values for fCLKOUT, t1, t2, and t3
- section 7.4.9 and 7.4.10, updated the NEC V850 Type 1 and Type 2 Read/Write Timing tables with a note regarding programmable wait states
- section 7.4.9 and 7.4.10, updated the NEC V850 Type 1 and Type 2 Write and Read Timing figures and tables with new timing “t13” to clarify WAIT# state after writes are completed and read data is ready
- section 7.5.2, removed note 1 from the I2C Host Interface Timing table
- section 7.6, updated the Panel Interface Timing figures and tables
- section 7.7, added min/max values for the Camera Interface Timing table
- section 11.1, corrected typos for the PU/D conditions for NEC V850 Type 2 8-bit Direct and Renesas SH4 8-bit Direct in the Hard Reset Pin States table
- section 16.5, changed references from “PIP+” to “AUX / OSD”
- section 17.3, added note to step 3 regarding the SDRAM command sequence

X83A-A-001-00 Revision 0.07 - Issued: December 06, 2007

- all changes from the last revision are highlighted in Red
- changed all references from “Intel Monahans” to “Marvell PXA3xx”
- changed all references from “TI EBI” to “TI TMS470”
- removed all references to LCD1 being the “Primary” interface and LCD2 being the “Secondary” interface
- Globally - change Keypad references from “KB...” to “KP...”
- section 3.2, revised Use Case 2 figure to clarify that “Streaming Data” is TFT RGB 8:8:8 input
- section 5.3, removed the RESET# State column from the pin description tables (this information is now included in section 11)
- section 5.3, moved IRQ pin description from the Miscellaneous pins section to the Host Interface pins section
- section 5.3.1, changed:
  AB6 Cell Type from “BHSC4D2” to “BHSC4P2”
  DB9 Cell Type from “BHSC4D2” to “BHSC4P2”
  CS# Cell Type from “ICU1” to “ICD1”
  RD# Cell Type from “ICU1” to “ICD1”
  RD/WR# Cell Type from “ICU1” to “ICD1”
  BE0# Cell Type from “ICU1” to “ICD1”
  BE1# Cell Type from “BHSC4P2” to “BHSC4D2”
  BS# Cell Type from “BHSC4D2” to “BHSC4P2”
  BURST# Cell Type from “ICU1” to “IC”
  BDIP# Cell Type from “ICU1” to “IC”
  BUSCLK Cell Type from “ICU1” to “ICD1”
  CNF[2:1] Cell Type from “ICD2” to “IC”
- section 5.3.2, changed the FP2IO17 Cell Type to “BHSC4P2”
- section 5.4, changed MPC555 to use BE1# to determine Indirect/Direct
- section 5.5, changed MPC555 host pin mapping to show that BE1# is used to determine Indirect/Direct
- tables 5-12 ~ 5-18 - changes to table formats and some signal names
- section 6 D.C. Characteristics - for tables 6-2 and 6-3 change T_{OPR} max to “105”
- section 7.1.1, added min/max values for Clock Requirements tables
- section 7.1.2, added max values for Internal Clock Requirements table
- section 7.1.3, added max values for the PLL Clock Requirements table
- section 7.3, added RESET# Timing min/max values and note 1
- section 7.4, updated all Parallel Host Interface Timing figures/tables
- section 7.5, updated all Serial Host Timing timing figures/tables
- section 7.6, added Panel Interface Timing min/max values
- section 7.7, changed all Camera Interface Timing min values to TBD
- section 7.8, updated the SDRAM Interface Timing section

- removed I2C Interface Timing section
- section 7.9, updated I2S Interface Timing section with new figures and tables
- section 7-12 Keypad Interface Timing - changes to table 7-40 Keypad Interface Timing, replace figure 7-43 “Keypad Interface Input Timing” with “Keypad Glitch Filter Input Timing”
- section 7.11, added Serial Flash (SPI) Interface Timing section
- section 9, added the Camera1/Camera2 Clock Output Disable bits
- section 10.1, added a note about accessing synchronous/asynchronous registers when power save mode is enabled
- REG[000Ch] ~ REG[000Fh], updated the C33 Debugger Start Address registers with information on calculating the memory range used by the debugger
- REG[003Ch] bit 0, added a note about accessing synchronous/asynchronous registers when power save mode is enabled
- REG[003Dh] - add this register
- REG[03Dh], clarified the bit descriptions and added information for bit 3 that the Camera IO Drive Select bit affects CM1CLKOUT, SCL, and SDA
- REG[0084h] bit 2, removed this bit and bit description
- REG[008Ah] - rename register and change register bits and descriptions
- REG[0100h], changed default register value from “20h” to “21h”
- REG[0100h] bit 0 and REG[0101h] bit 0, clarified the I2S Data Clock Source and WSIO and SCKIO Output Enable bit description by summarizing the settings in a table
- REG[0101h] bit 6 - reserved this bit
- REG[0104h] bit 0, for the I2S DAC Controller Enable bit description removed “The data written...by Right Channel Data” from the Note
- REG[0186h] bit 5 - rewrite “When this bit = 0b,...” and “When this bit = 1b,...”
- REG[0188h], changed default register value from “80h” to “00h”
- REG[0188h] bits 4-0, added information about the pull-up/pull-down resistor controls when the Camera1 interface is configured for 24-bit RGB 8:8:8 streaming input
- REG[0189h], changed default register value from “20h” to “00h”
- REG[0189h] bits 4-0, added information about the pull-up/pull-down resistor controls when the Camera2 interface is configured for 24-bit RGB 8:8:8 streaming input
- REG[0200h] bits 7-4 - correct typo in register table for bit name, change “counter” to “rate”
- REG[0400h] ~ REG[0457h], clarified the Warp Logic bit names and bit descriptions
- REG[0444h] ~ REG[0447h], corrected the Warp Logic Offset Table definitions, should be “outputwidth/N” or “outputwidth/M” instead of “inputwidth+1”
- REG[0454h] ~ REG[0457h], corrected the Luminance Table definitions, should be “outputwidth/N” or “outputwidth/M” instead of “inputwidth+1”

- REG[0900h] ~ REG[09A7h], clarified the Blending Engine Configuration Register bit names and bit descriptions
- REG[0900h] bit 3 - change the bit name to CH1 Output Vertical Flip Enable and rewrite the bit description to match
- REG[0942h] bit 5, added the MAIN Frame Buffer 1 Ready Clear bit and bit description
- REG[0942h] bit 4, added the MAIN Frame Buffer 0 Ready Clear bit and bit description
- REG[0942h] bit 2, added note to the MAIN Window Current Frame Status bit description describing the procedure when the MAIN window is disabled and re-enabled
- REG[0962h] bit 5, added the AUX Frame Buffer 1 Ready Clear bit and bit description
- REG[0962h] bit 4, added the AUX Frame Buffer 0 Ready Clear bit and bit description
- REG[0962h] bit 2, added note to the AUX Window Current Frame Status bit description describing the procedure when the AUX window is disabled and re-enabled
- REG[0982h] bit 5, added the OSD Frame Buffer 1 Ready Clear bit and bit description
- REG[0982h] bit 4, added the OSD Frame Buffer 0 Ready Clear bit and bit description
- REG[0982h] bit 2, added note to the OSD Window Current Frame Status bit description describing the procedure when the OSD window is disabled and re-enabled
- REG[09A0h] bit 4 - change the bit name and rewrite bit description
- REG[09A1h], added information about disabling and re-enabling OSD layer
- REG[09AAh] ~ REG[09C5h], clarified the Image Fetcher Configuration Register bit names and bit descriptions
- REG[09B2h] bit 5, added the Image Fetcher Frame Buffer 1 Ready Clear bit and bit description
- REG[09B2h] bit 4, added the Image Fetcher Frame Buffer 0 Ready Clear bit and bit description
- REG[09B2h] bit 2, added note to the Image Fetcher Current Frame Status bit description describing the procedure when the Image Fetcher is disabled and re-enabled
- REG[09C8h] ~ REG[09FEh], clarified the LCD Configuration Register bit names and bit descriptions
- REG[09C8h] bits 7-4 - rename these bits to “... Idle” and mark them as read only
- REG[09CAh] bit 5 - correct typo in bit description, change “horizontal” to “vertical”
- REG[09F0h] ~ REG[09F5h], changed the bit description of these registers to define the Width/Height/Virtual Width of the Camera1 Frame Buffer instead of the Camera1 image
- REG[09F6h], changed register name from “Camera1 Control Register” to “Camera1 Write Control Register”
- REG[09F8h] ~ REG[09FDh], changed the bit description of these registers to define the Width/Height/Virtual Width of the Camera2 Frame Buffer instead of the Camera2 image
- REG[09FEh], changed register name from “Camera2 Control Register” to “Camera2 Write Control Register”
- REG[0A00h] ~ REG[0A46h], clarified the Interrupt Configuration bit names and bit descriptions
- REG[0A02h]/REG[0A08h]/REG[0A10h] bits 6-5, reserved the VBUS1/2 Address Error Interrupt Status and Enable bits

- REG[0D00h] ~ REG[0D35h], clarified the Camera1 bit names and bit descriptions
- REG[0D00h] bit 7, changed Camera1 Software Reset to a Write Only bit
- REG[0D02h] bit 7, added the Camera1 Clock Output Disable bit and bit description
- REG[0D09h], changed the register name from “Camera1 Input Frame Control Register” to “Camera1 Flag Clear Register”, changed the register from Read/Write to Write Only and updated the bit descriptions accordingly
- REG[0D0Eh] bit 5, added a bit description for the Frame Event Status bit
- REG[0D0Eh] bit 4, added a bit description for the Effective Capture Status bit
- REG[0D0Eh] bit 3, added a bit description for the Effective Frame Status bit
- REG[0D0Eh] bit 1, clarified how to clear the ITU-R BT.656 Error Flag 1 Status bit
- REG[0D0Eh] bit 0, clarified how to clear the ITU-R BT.656 Error Flag 0 Status bit
- REG[0D22h], changed “Camera1 VRAM Buffer Overflow Clear Register” to “Camera1 YRC Buffer Overflow Clear Register”
- REG[0D22h], reserved the Camera1 YRC Buffer Overflow Clear register
- REG[0D28h], changed “Camera1 VRAM Buffer Overflow Status Register” to “Camera1 YRC Buffer Overflow Status Register”
- REG[0D28h], reserved the Camera1 YRC Buffer Overflow Status register
- REG[0D30h] bits 3-2, added the Camera1 Write Field Select bits and bit description
- REG[0D40h] ~ REG[0D75h], clarified the Camera2 bit names and bit descriptions
- REG[0D40h] bit 7, changed Camera2 Software Reset to Write Only bit
- REG[0D42h] bit 7, added the Camera2 Clock Output Disable bit and bit description
- REG[0D46h], changed default register value from “00h” to 04h”
- REG[0D49h], added the Camera2 Flag Clear Register
- REG[0D4Eh] bit 5, added a bit description for the Frame Event Status bit
- REG[0D4Eh] bit 4, added a bit description for the Effective Capture Status bit
- REG[0D4Eh] bit 3, added a bit description for the Effective Frame Status bit
- REG[0D4Eh] bit 1, clarified how to clear the ITU-R BT.656 Error Flag 1 Status bit
- REG[0D4Eh] bit 0, clarified how to clear the ITU-R BT.656 Error Flag 0 Status bit
- REG[0D62h], changed “Camera2 VRAM Buffer Overflow Clear Register” to “Camera2 YRC Buffer Overflow Clear Register”
- REG[0D62h], reserved the Camera2 YRC Buffer Overflow Clear register
- REG[0D68h], changed “Camera2 VRAM Buffer Overflow Status Register” to “Camera2 YRC Buffer Overflow Status Register”
- REG[0D68h], reserved the Camera2 YRC Buffer Overflow Status register
- REG[0D70h] bits 3-2, added the Camera2 Write Field Select bits and bit description

- REG[3C00h] ~ REG[3C22h], clarified the DMA Controller bit names and bit descriptions
- REG[3C0Ch] bits 1-0, removed restriction for Fill Mode where destination is the external SDRAM
- REG[3C1Ch] bits 1-0, removed restriction for Fill Mode where destination is the external SDRAM
- REG[3C40h] ~ REG[3C44h], clarified the SDRAM Controller Configuration bit descriptions
- REG[3C40h] bit 7, added the SDRAM tRCD Timing bit and bit description
- REG[3C40h] bit 6, added the SDRAM tRAS Timing bit and bit description
- REG[3C40h] bit 5, added the SDRAM tRP Timing bit and bit description
- REG[3C40h] bit 4, added the SDRAM CAS Latency bit and bit description
- REG[3C40h] bit 0, updated the SDRAM Initialize bit description to replace the sentence "The SDRAM is programmed..." with "The SDRAM is programmed using the settings in REG[3C40h] bits 7-4, and full page mode access."
- REG[3C44h] bit 6, added the SDRAM Self Refresh Enable bit and bit description
- REG[3C44h] bits 4-0, changed the value these bits must be set to from "05h" to "13h"
- REG[4000h] ~ REG[40B1h], clarified the LCD Panel Configuration registers
- REG[4001h] bit 7, changed bit name from "FPSHIFT2 Polarity Select" to "LCD2 PCLK Polarity Select"
- REG[4001h] bit 3, changed bit name from "FPSHIFT1 Polarity Select" to "LCD1 PCLK Polarity Select"
- REG[4001h] bit 2, reserved the Panel Signals Swap bit and bit description
- REG[4044h] bits 5-4, corrected reference in the table from "middle" to "gray"
- REG[5000h] bit 0 - change bit name in register table to match bit name in description
- SDRAM[**004h] ~ SDRAM[**007h], removed note that the Sprite #n Image Start Address must not be set within the range 1000_0000h through 1000_000Fh
- SDRAM[**008h] ~ SDRAM[**00Bh], removed note that the Sprite #n Rotated Image Start Address must not be set within the range 1000_0000h through 1000_000Fh
- section 11, replaced the power save section with new Operating Configurations and States section
- section 12 Display modes - remove this section
- section 13 Display Subsystem - rewrite entire section
- section 17 SDRAM Interface - rewrite section
- section 17, replaced the SDRAM Interface section
- section 20 General-Purpose IO Pins - add this section
- section 21 Host Interface - add this section
- section 22 LCD Panel Interface - remove this section
- section 22, replaced the Camera Interface section
- section 25, added the SPI Flash Memory Interface section

X83A-A-001-00 Revision 0.06 - Issued: September 14, 2007

- all changes from the last revision are highlighted in Red
- Globally - change LCDC Fetcher to Image Fetcher
- Globally - remove register/block references to HUD
- section 3 - rename to “Typical Implementation Use Cases”
- section 5.1, added pinout diagram for QFP package
- section 5.3, added QFP Pin #s to the pin descriptions
- section 5.3.2, changed the FP2IO17 pin from Output to Input/Output
- section 5.3.5, for the SPI Flash Interface pin description section corrected the SPICS# and SPICLK pins to be outputs and updated the pin descriptions accordingly
- section 5.6, updated the names of the Camera2 interface pins from “CAM2...” to “CM1...”
- section 6, added preliminary DC Characteristics
- section 7.12 Keypad Interface Timing - changes to figure 7-41 Keypad Interface Base Timing and figure 7-42 Keypad Interface Timing, change “CLK32K” to “KPDCLK” and add note “KBRx are sampled/checked at the end of each KBCx pulse”
- section 8, added note about not accessing the SPI port when SPI is disabled
- section 8, added note about accessing the BPPC
- section 9.1, added LSCLK reference for the Timer Clock
- REG[0033h] - in table, mark 00000b as reserved
- REG[003Ch] bit 2, renamed the “PLL2 Select” bit to the “LCD Clock Source Select” bit and updated the bit description accordingly
- REG[003Ch] bit 1, renamed the “PLL1 Select” bit to the “SDRAM Clock Source Select” bit and updated the bit description accordingly
- REG[003Eh] bit 7, renamed the “LCD Clocks Source Select” bit to the “Input Clock 2 Source Select” bit and updated the bit description accordingly
- REG[003Eh] bits 6-5, renamed the “LCD Clocks Divide Select” bits to “PLL2 Input Divide Select” bits and updated the description accordingly
- REG[003Eh] bit 4, renamed the “LCD Clocks Divide Enable” bit to “PLL2 Input Divide Enable” and updated the description accordingly
- REG[003Eh] bit 3, renamed the “SYSCLK Source” bit to “Input Clock 1 Source” bit and updated the description accordingly
- REG[003Eh] bits 2-1, renamed the “SYSCLK Divide Select” bits to “PLL1 Input Divide Select” bits and updated the description accordingly
- REG[003Eh] bit 0, renamed the “SYSCLK Divide Enable” bit to “PLL1 Input Divide Enable” and updated the description accordingly
- REG[0040h] ~ REG[0041h], added these registers as Reserved

- REG[0061h] bit 2, reworded the bit description for the SPI Clock Source Select bit
- REG[0063h] bit 2, reworded the bit description for the I2C Clock Source Select bit
- REG[00ACh] ~ REG[00ADh], added note about using SDRAM Buffers for interfaces without wait
- REG[00ACh] ~ REG[00ADh], added note about Internal Memory Space R/W Port reads when using SPI
- section 10.4.3, added note about accessing the BPPC
- REG[0104h] bits 5-2 - add "...or equal to..." to the third line of the bit description
- REG[0188h], changed the default register value from "FFh" to "80h"
- REG[0189h], changed the default register value from "3Fh" to "20h"
- REG[0189h], updated the names of the Camera2 interface pins in the pull-down bit descriptions from "CAM2..." to "CM1..."
- REG[01C8h] ~ REG[01CBh], added note about the Keypad Interrupt when the polarity is changed from 1 to 0
- REG[01C8h] ~ REG[01CBh] - swap the "When this bit = 0b" and "When this bit = 1b" descriptions
- REG[01D6h], added comment about corresponding Keypad Interrupt going high
- REG[0200h] bits 7-5 - change bit name and re-write description
- REG[0201h] bits 6-0 - rename register bits
- REG[0202h] bits 6-0 - rename register bits
- REG[0203h] bits 7-4 - re-write description
- REG[0203h] bits 3-0 - rename register bits
- REG[0204h] bits 6-0 - rename register bits
- REG[0205h] bits 6-0 - rename register bits
- REG[0206h] bits 7-4 - re-write description
- REG[0206h] bits 3-0 - rename register bits
- REG[0240h] ~ REG[03FFh], clarified the bit descriptions for the SDRAM Read/Write Buffer Registers
- REG[024Ch] ~ REG[024Dh], added note about SDRAM Buffer 0 Port reads when using SPI
- REG[025Ch] ~ REG[025Dh], added note about SDRAM Buffer 1 Port reads when using SPI
- REG[0264h] ~ REG[0267h], changed the SDRAM Read/Write Buffer Internal Address registers from "Read/Write" to "Read Only"
- REG[0300h] ~ REG[037Fh], added note about Aliased SDRAM Buffer 0 Port reads when using SPI
- REG[0380h] ~ REG[03FFh], added note about Aliased SDRAM Buffer 1 Port reads when using SPI
- REG[0408h] bits 1-0, added descriptions for each bit state for the Frame Buffer 0 and Frame Buffer 1 Ready Status bits
- REG[0900h], changed the register from "Write Only" to "Read/Write"
- REG[0900h] bit 0, added note about disabling hardware frame control before disabling CH1 Output

- REG[0920h] bit 0, added note about disabling hardware frame control before disabling CH2 Output
- REG[0930h] bit 0, added note about disabling hardware frame control before disabling OSD Output
- REG[0960h] bit 4, updated bit description for the AUX Window Enable bit and added note about disabling hardware frame control before disabling the AUX window
- REG[0980h] bit 4, updated bit description for the OSD Window Enable bit and added note about disabling hardware frame control before disabling the OSD window
- REG[0980h] bits 3-2, added note that ARGB formats for the OSD window are not supported when Blend Mode 3 is selected
- REG[09A0h] bit 7, added this bit as a reserved bit
- REG[09B0h] bit 4, renamed the “LCDC Fetcher Fetch Mode” bit to “LCDC Fetcher Mode” bit
- REG[09B0h] bit 4, updated bit description for the LCDC Fetcher Enable bit and added note about disabling hardware frame control before disabling the LCDC Fetcher
- REG[09C8h] bits 7-4 - rewrite bit descriptions “When this bit = 1b...”
- REG[09CAh] bit 3, added note that manual trigger will not cause a MAIN buffer switch
- REG[09D8h] bit 0, added notes about double buffering to the MAIN Window HW/SW Frame Control bit description
- REG[09D9h] bit 0, added notes about double buffering to the AUX Window HW/SW Frame Control bit description
- REG[09DAh] bit 0, added notes about double buffering to the OSD Window HW/SW Frame Control bit description
- REG[09DBh] bit 0, added notes about double buffering to the LCDC Fetcher HW/SW Frame Control bit description
- REG[09F6h] bit 7, added the Camera1 Double Buffer Method Select bit and bit description
- REG[09F6h] bit 7, added notes 1, 2, and 3 about restrictions for Camera1 Double Buffer Method 1
- REG[09F6h] bit 6, added this bit as a reserved bit
- REG[09FEh] bit 7, added the Camera2 Double Buffer Method Select bit and bit description
- REG[09FEh] bit 7, added notes 1, 2, and 3 about restrictions for Camera2 Double Buffer Method 1
- REG[09FEh] bit 6, added this bit as a reserved bit
- REG[0A00] bit 2 - remove “Read-Only” from bit
- REG[0A44h] - change note to “Interrupt 2 corresponds to the Watchdog Interrupt which can be read and cleared in Interrupt Status Register 0 (REG[0A00h]) bit 2.”
- REG[0A80h] ~ REG[0A8Dh], clarified the bit descriptions for the Timer Configuration registers
- REG[0B00h] ~ REG[0B0Ah], clarified the bit descriptions for the SPI Flash Memory interface
- REG[0C00h], clarified the bit description for the C33 Instruction Cache Enable bit
- REG[0D04h], clarified the bit names for the CM1VREF Polarity, CM1HREF Polarity bits

- REG[0D08h], clarified the bit names for the Camera1 Frame Event bits
- REG[0D0Ah] ~ REG[0D0Bh], clarified the Camera1 Input Horizontal Size bit description for different modes
- REG[0D0Ch] ~ REG[0D0Dh], clarified the Camera1 Input Vertical Size bit description for different modes
- REG[0D18h] ~ REG[0D19h], added formulas for Camera1 Horizontal and Vertical scaling registers
- REG[0D44h], clarified the bit names for the CM2VREF Polarity, CM2HREF Polarity bits
- REG[0D48h], clarified the bit names for the Camera2 Frame Event bits
- REG[0D4Ah] ~ REG[0D4Bh], clarified the Camera2 Input Horizontal Size bit description for different modes
- REG[0D4Ch] ~ REG[0D4Dh], clarified the Camera2 Input Vertical Size bit description for different modes
- REG[0D58h] ~ REG[0D59h], added formulas for Camera2 Horizontal and Vertical scaling registers
- REG[3C0Ch] bits 1-0 - add note “When performing a memory fill using the DMA Controller...”
- REG[3C1Ch] bits 1-0 - add note “When performing a memory fill using the DMA Controller...”
- REG[3C44h] bit 4, added this bit as a Reserved bit
- REG[4018h] bit 0, added note with conditions when the LCD1 VNDP Status will not be set
- REG[4019h] bit 3, changed the reference from the “VSYNC Interrupt Mask Disable bit, REG[0818h] bit 10” to “LCD1 Interrupt Enable bit, REG[0A06h] bit 0”
- REG[401Ah] ~ REG[401Bh], added comment about case where LCD1 VSYNC Interrupt Delay is greater than VT
- REG[4030h], removed the reference to DualView panels and added a note with the recommended setting for EID Double Screen panels with TCON enabled
- REG[4036h] bit 0, added note with conditions when the LCD2 VNDP Status will not be set
- REG[4037h] bit 3, changed the reference from the “VSYNC Interrupt Mask Disable bit, REG[0818h] bit 10” to “LCD2 Interrupt Enable bit, REG[0A06h] bit 1”
- REG[4038h] ~ REG[4039h], added comment about case where LCD2 VSYNC Interrupt Delay is greater than VT
- REG[4041h] bit 0, reserved this bit
- REG[4042h] bit 7, changed the polarity of the VREVOUT Configuration bit and added table summarizing the possible configurations
- REG[4042h] bit 3, changed the polarity of the HREVOUT Configuration bit and added table summarizing the possible configurations
- REG[4046h], updated the OE Signal Low Width bit description as to the Special Drive Mode bit
- REG[4060h] bit 7, clarified what is reset when a LCD1 Software Reset is performed
- REG[4060h] bit 6, clarified the bit description for the LCD1 Display Blank bit and added a summary table
- REG[4060h] bit 5, changed the LCD1 Video Invert bit description to state that the bit has an effect when the display is blanked
- REG[4070h] bit 7, clarified what is reset when a LCD2 Software Reset is performed

- REG[4070h] bit 6, clarified the bit description for the LCD2 Display Blank bit and added a summary table
- REG[4070h] bit 5, changed the LCD2 Video Invert bit description to state that the bit has an effect when the display is blanked
- REG[5000h] ~ REG[502Bh], clarified the bit descriptions for the Sprite Engine Registers
- SDRAM[**001h] bits 3-2, added the Sprite #n Rotation bits and bit description
- SDRAM[**004h] ~ SDRAM[**007h], added note about address range restriction
- SDRAM[**018h] ~ SDRAM[**019h], added information about how the Sprite transparency works
- section 10.4.9, added note about using SDRAM Buffers for interfaces without wait
- section 13, added note about accessing the BPPC
- section 15 - rename this chapter to “I2S Audio Output Interface” and add body text
- section 16, added 2D BitBLT section
- section 19 SDRAM Read/Write Buffer - add this section
- section 20 Pulse Width Modulation (PWM) - add body text to this chapter
- section 24 Keypad Interface - add body text to this section
- section 25 - add this chapter “Watchdog Timer Interface”
- section 28, added Product Brief to the References section
- section 29, remove this section and place sales office info on last page

X83A-A-001-00 Revision 0.05 - Issued: July 27, 2007

- all changes from the last revision are highlighted in Red
- section 1 Introduction - rewrite section
- section 2 Features - rewrite section
- section 5.3 ~ 5.6 - rewrite these sections
- Section 7.5 Parallel Host Bus Interface Timing - updates to timing diagrams and numbers throughout section
- Section 7.6 Serial Host Bus Interface Timing - add this section
- section 8 Memory Map - correct typos, clean up and re-arrange table 8-1
- section 9-1 Clock Overview - in figure 9-1, rename SPIEN to SPICLKISEL and I2CEN to I2CCLKISEL
- REG[0032h] - in table, mark 00000b as reserved
- REG[0061h] - add “or 10h if SPI Enabled” to the register default value
- REG[0061h] bit 2 - rename this bit and add to description
- REG[0061h] bit 0 - rename this bit and rewrite bit description
- REG[0063h] bit 2 - rename this bit and add to description
- REG[0063h] bit 0 - rename this bit and rewrite bit description

- REG[0084h] bit 0 - rewrite bit description
- REG[0085h] bits 2-0 - rewrite bit description
- REG[00A8h] ~ REG[00ABh] - add note “The user must access DRAM using the SDRAM Read/Write Buffer...” to bit description
- section 10.4.4 I2S Control Registers - rewrite bit descriptions throughout section
- section 10.4.5 I2S DMA Registers - rewrite bit descriptions throughout section
- section 10.4.6 GPIO Registers - rewrite bit descriptions throughout section
- section 10.4.7 Keypad Registers - rewrite bit descriptions throughout section
- section 10.4.8 PWM Registers - rewrite bit descriptions throughout section
- REG[0400h] bit 2 - add this bit
- REG[0430h] ~ REG[0432h] - rewrite bit descriptions
- REG[0440h] bits 6-4 - changes to Block Power in table, add “Where n = the...” after table
- REG[0440h] bits 2-0 - changes to Block Power in table, add “Where n = the...” after table
- REG[0450h] bits 6-4 - changes to Block Power in table, add “Where n = the...” after table
- REG[0450h] bits 2-0 - changes to Block Power in table, add “Where n = the...” after table
- REG[090Fh] - change default value to 00h
- REG[0940h] bit 7 - add this bit
- REG[0954h] ~ REG[0955h] - rewrite note
- REG[0960h] bit 7 - add this bit
- REG[0974h] ~ REG[0975h] - rewrite note
- REG[0980h] bit 7 - add this bit
- REG[0994h] ~ REG[0995h] - rewrite note
- REG[09A2h] - change default value to 0Xh
- REG[09B0h] bit 7 - add this bit
- REG[09C4h] ~ REG[09C5h] - rewrite note
- REG[09C8h] bits 7, 6, 5 and 4 - add these bits
- REG[09F0h] ~ REG[09F1h] - add note “The Camera1 width must be set such that the width multiplied by...”
- REG[09F4h] ~ REG[09F5h] - add note “The Camera1 virtual width must be set such that the virtual width multiplied by...”
- REG[09F8h] ~ REG[09F9h] - add note “The Camera2 width must be set such that the width multiplied by...”
- REG[09FCh] ~ REG[09FDh] - add note “The Camera2 virtual width must be set such that the virtual width multiplied by...”
- REG[09F6h] bit 4 - delete this bit and mark as n/a

- REG[09FEh] bit 4 - delete this bit and mark as n/a
- REG[0A00h] bit 6 - mark this bit as read only and clean up description and bit references
- REG[0A00h] bit 1 and 0 - mark these bits as read only and change “To clear this status bit...” description
- REG[0A08h] bit 4 - add note “After enabling the keypad (REG[01C0h] bit 0 = 1b), all interrupts...”
- REG[0A8Bh] - remove this register and correct REG[0A8Ah] bit description
- REG[0B03h] - make this register Read/Write
- REG[0B04h] - add “and access to the VBUS I2S port is restricted” to bit description
- REG[0D02h] bits 6-2 - add formula for divide ratio to bit description
- REG[0D02h] bit 1 - reserve this bit
- REG[0D04h] bit 3 - reserve this bit
- REG[0D08h] bit 5, 4 and 3 - rename bits to “event” from “interrupt”
- REG[0D08h] bit 0 - rename bit to “event” from “interrupt”
- REG[0D0Eh] - change default value to 0Xh
- REG[0D0Eh] bit 5 - rename bit to “event” from “interrupt”
- REG[0D0Eh] bit 2 - add bit description
- REG[0D0Eh] bit 1 and 0 - correct typo in register reference
- REG[0D0Fh] - reserve this register
- REG[0D1Ch] - reserve this register
- REG[0D1Eh] bit 7 - delete this bit and mark it as n/a
- REG[0D30h] ~ REG[0D35h] - add these registers
- REG[0D40h] bit 2 and 1 - reserve these bits
- REG[0D42h] bits 6-2 - add formula for divide ratio to bit description
- REG[0D42h] bit 1 - reserve this bit
- REG[0D42h] bit 0 -rewrite bit description
- REG[0D44h] ~ REG[0D48h] - rewrite bit descriptions
- REG[0D44h] bit 3 - reserve this bit
- REG[0DeEh] - change default value to 0Xh
- REG[0D4Eh] bit 5 - rename bit to “event” from “interrupt”
- REG[0D4Eh] bit 2 - add bit description
- REG[0D4Eh] bit 1 and 0 - change bit name and rewrite bit description
- REG[0D4Fh] - reserve this register
- REG[0D5Ch] - reserve this register

- REG[0D70h] ~ REG[0D75h] - add these registers
- REG[3C43h] - change default register value to 01h
- REG[4000h] bit 0 - reserve this register
- REG[4016h] bits 7-5 - correct typo in table, for 000b change ND-TFT to ND-TFD
- REG[401Ch] ~ REG[401Fh] - rename registers and rewrite bit description
- REG[403Ah] ~ REG[403Dh] - rename registers and rewrite bit description
- REG[4042h] change default register value to 11h
- REG[4040h] ~ REG[404Fh] - changes to register bit descriptions
- REG[4073h] - changes to register bit descriptions
- REG[4078h] ~ REG[409Ch] - changes to register bit descriptions
- REG[4078h] bit 7 - remove this bit and mark it n/a
- REG[5002h] bit 1 and 0 - delete bits and mark them as n/a
- SDRAM[**001h] bits 6-4 - add "The sprite 0 should always be the background in sprite..."
- SDRAM[**001h] bits 3-2 - delete these bits
- SDRAM[**00Ch] ~ SDRAM[**00Dh] - add note "The X position + sprite width must not be greater than 1024" to formula in bit description

X83A-A-001-00 Revision 0.04 - Issued: July 05, 2007

- all changes from the last revision are highlighted in Red
- section 10.1 Register Mapping - in table 10-2, for System Control Registers, change Asynchronous to "0020h to 003Fh" and the second Synchronous to "0050h to 007Fh"
- section 10.3 Register Restrictions - remove "All register accesses must be 16-bit accesses"
- REG[0002h], changed default register value from "0045h" to "45h"
- REG[0003h], changed default register value from "0000h" to "00h"
- REG[0008h] - move to REG[008A and rewrite bit description
- REG[0010h] ~ REG[0013h], minor rewording
- REG[001Ch] bit 7, added C33 Wakeup bit description
- REG[001Ch] bit 6, added C33 Enable bit description
- REG[001Dh] bit 0, added C33 Software Reset bit description
- REG[0024h] bit 0, removed comment about the System Clock in the PLL1 Enable bit description
- REG[002Ch] bit 0, removed comment about the LCD Clock in the PLL2 Enable bit description
- REG[0034h] ~ REG[0035h], added PWMCLK Divide Ratio information
- REG[0036h] ~ REG[0039h] - move these registers to REG[0060h] ~ REG[0063h]

- REG[003Ah], removed the Host I2C Slave Address register
- REG[003Ch], added bit description for Power Save Mode Enable bit
- REG[003Eh] bit 3 - change the name and function of this bit
- REG[003Fh] - move this register to REG[003Eh] and change register default value to 08h
- REG[0084h] bit 0, updated the Asynchronous System Control Registers Host Access bit name and description
- REG[0085h] bit 4, added Host Data Byte Swap Enable bit description
- REG[00A6h] bit 0, renamed the bit to “Internal Memory Space Auto-Increment Enable”
- REG[0104h] bit 0 - add note “When the DAC is enabled, and the DAC is in stereo mode...”
- REG[0182h] - change default register value to FFh
- REG[0183h] - change default register value to FFh
- REG[024Ch] ~ REG[024Dh] - rewrite bit description
- REG[025Ch] ~ REG[025Dh] - rewrite bit description
- REG[0300h] ~ REG[037Fh] - add these registers
- REG[0380h] ~ REG[03FFh] - add these registers
- REG[0402h] - change the name of bit 2, add Output to name
- REG[0402h] - change the name of bit 2, remove bits 1 and 0 and mark as n/a
- REG[0404h] - change the name of bit 2, remove bits 1 and 0 and mark as n/a
- REG[0406h] - change the name of bit 2, remove bits 1 and 0 and mark as n/a
- REG[0411h] - add 4 bits to HUD/Warp Input Width and remove note “These bits must be set such that the HUD input width is...”
- REG[0414h] ~ REG[0415h] - add note “These bits must be set such that the HUD output width is...”
- REG[0416h] ~ REG[0417h] - add note “These bits must be set such that the HUD output height is...”
- REG[0454h] ~ REG[0457h] - in table remove Y component and rename X component to Luminance
- REG[0900h] bits 5-4 - add these new bits
- REG[0900h] bit 3 - add “This bit should be set to 0b for tiled frame mode (REG[0900h] bit 2 = 1b)” to bit description
- REG[0900h] bit 2 - add note “For tiled frame mode, the image width must be a multiple of 8 pixels” to bit description
- REG[0904h] ~ REG[0907h] - change register name to “CH1OUT Writeback Frame Buffer 0 Address...”
- REG[0940h] bits 6 and 5 - add note “If Blend Mode is 2, where OSD is an overlay on top of AUX...” to bit description
- REG[0940h] bit 1 - add note “For tiled frame mode, the image width must be a multiple of 8 pixels” to bit description

- REG[0960h] bits 6 and 5 - add note “If Blend Mode is 2, where OSD is an overlay on top of AUX...” to bit description
- REG[0960h] bit 1 - add note “For tiled frame mode, the image width must be a multiple of 8 pixels” to bit description
- REG[0980h] bit 1 - add note “For tiled frame mode, the image width must be a multiple of 8 pixels” to bit description
- REG[09A0h] bit 4 - add this bit and description to register
- REG[09A0h] bit 2 - rewrite bit description
- REG[09A0h] bits 1-0 - changes to table layout in but description
- REG[09A1h] - rewrite register description
- REG[09A2h] - rename pins to “... Pin Status” and rewrite bit description
- REG[09A3h] - change register default value to 03h and rewrite bit descriptions
- REG[09B0h] bits 3-2 - remove these bits
- REG[09B0h] bit 1 - add note “For tiled frame mode, the image width must be a multiple of 8 pixels” to bit description
- REG[09CAh] bit 7 - add note “For tiled frame mode, the image width must be a multiple of 8 pixels” to bit description
- REG[09CAh] bit 5 - add “This bit should be set to 0b for tiled frame mode (REG[09CAh] bit 7 = 1b)” to bit description
- REG[09DCh] bits 3, 2, 1 and 0 - add these bits
- REG[0A00h] bit 7 - add this bit
- REG[0A02h] bits 5 and 6 - correct internal RAM addresses in bit descriptions and correct upper I2S Port address (change 3801_FFFFh to B801_FFFFh)
- REG[0A06h] bit 7 - add this bit
- REG[0A0Ch] - change default register value to 04h
- REG[0A0Eh] bit 7 - add this bit
- REG[0A86h] ~ REG[0A87h] - rewrite bit description
- REG[0A88h] ~ REG[0A89h] - rewrite formulas in bit description
- REG[0A8Ah] ~ REG[0A8Bh] - rewrite formulas in bit description
- REG[0B03h] - change register to Write Only and rewrite bit description
- REG[0C00h] - fix bit descriptions to match register layout
- REG[0D00h] bits 1 and 2 - Reserve theses bits
- REG[0D06h] bit 0 - remove this bit
- REG[0D09h] - add this register

- REG[0D40h] - add bits 2 and 1 to register
- REG[0D46h] bit 0 - remove this bit
- REG[3C20h] bit 1 - rewrite bit description
- section 10.4.21 LCD Panel Configuration Registers - expand bit descriptions throughout entire section
- REG[4000h] - change default register value to 88h
- REG[4000h] bit 1 - rename this bit and change bit description
- REG[4001h] bits 1-0 - reserve setting of 11b
- REG[4005h] bit 3 - mark this bit as n/a and make LCD1 Horizontal Display Period 11 bits long (bits 10-0)
- REG[4009h] bit 0 - add this bit to LCD1 Horizontal Pulse Width (REG[4008h]) as bit 8 and rename register
- REG[4023h] bit 3 - mark this bit as n/a and make LCD2 Horizontal Display Period 11 bits long (bits 10-0)
- REG[4027h] bit 0 - add this bit to LCD2 Horizontal Pulse Width (REG[408h]) as bit 8 and rename register
- REG[4040h] bit 4 - reserve this bit
- REG[4041h] - change default register value to 01h
- REG[4042h] - change default register value to 98h
- REG[404Ch] - reserve this register
- REG[4062h] bit 3 - add this reserved bit
- REG[4064h] ~ REG[4065h] - make CH1 FIFO Threshold 7 bits, mark REG[4064] bit 7 and REG[4065h] bit 0 as n/a, rename REG[4065h], change REG[4064h] default register value to 7F
- REG[4072h] bit 3 - add this reserved bit
- REG[4073h] bits 7-6 - add these bits to the register
- REG[4073h] bit 3 - add this reserved bit
- REG[4074h] ~ REG[4075h] - make CH2 FIFO Threshold 7 bits, mark REG[4074] bit 7 and REG[4075h] bit 0 as n/a, rename REG[4075h], change REG[4074h] default register value to 7F
- REG[4076h] ~ REG[4077h] - make OSD FIFO Threshold 7 bits, mark REG[4076] bit 7 and REG[4077h] bit 0 as n/a, rename REG[4077h], change REG[4076h] default register value to 7F
- REG[4078h] - rewrite entire register by removing all bits then adding all new bits
- REG[4079h] - add this register
- REG[407Ah] ~ REG[407Fh] - add these registers
- REG[4088h] - change default register value to 40h
- REG[408Ah] - change default register value to 40h
- REG[408Ch] - change default register value to 40h
- REG[4098h] - change default register value to 40h
- REG[409Ah] - change default register value to 40h

- REG[409Ch] - change default register value to 40h
- REG[5000h] - change default register value to 02h
- REG[5000h] bits 5-4 - changes to table in bit description
- REG[5000h] bit 2 - remove this bit and mark it as n/a
- REG[5001h] bit 7 - make this bit write only
- REG[5003h] - change register default value to 80h
- REG[5028h] ~ REG[502Bh] - changes to the address locations in figure 10-2
- section 10.4.23 Sprite Memory Based Registers - changes to addresses in body text and figure 10-3
- SDRAM[**01Ah] - change note to "...set to 0b (REG[5000h] bit 6 = 0b)
- section 14.2 "Tiled Frame" Storage - add note "For tiled frame storage the frame width..."

X83A-A-001-00 Revision 0.03 - Issued: June 06, 2007

- all changes from the last revision are highlighted in Red
- section 10.1 Register Mapping - in table 10-2 change System Control Registers Asynchronous from "001Fh' to "0020h"
- REG[001Fh] - move this register to REG[003Fh] and add bit 3
- REG[0100h] bit 5 - change bit description from "When this bit = 1b, the... when WS = 1, left channel when WS = 1" to "When this bit = 1b, the... when WS = 1, left channel when WS = 0"
- REG[010Ch] - change default register value to 04h
- REG[0154h] bit 1 - change this bit to read/write from read only
- REG[0242h] bits 2 and 1 - change these two bits to write only and add "This bit always reads 0b" to the bit descriptions
- REG[024Bh] - add four bits to this register
- REG[0252h] bits 2 and 1 - change these two bits to write only and add "This bit always reads 0b" to the bit descriptions
- REG[025Bh] - add four bits to this register
- REG[0400h] - swap positions of bit 1 and bit 4
- REG[0414h] - make bit 0 read only
- REG[0416h] - make bit 0 read only
- REG[0420h] bits 2-0 - mark these bits as read only
- REG[0424h] bits 2-0 - mark these bits as read only
- REG[0440h] - change default register value to 33h
- REG[0444h] bits 2-0 - mark these bits as read only
- REG[0450h - re-arrange register bits (delete memory table select) and change default value to 33h

- REG[0452h] - change default register value to 01h
- REG[0454h] bits 2-0 - mark these bits as read only
- REG[0900h] bit 3, added the CH1 Output Horizontal Flip Enable bit and bit description
- REG[0979h], expanded the AUX Window Y Offset bits from bits [9:0] to bits [10:0]
- REG[0999h], expanded the OSD Window Y Offset bits from bits [9:0] to bits [10:0]
- REG[09A1h], changed default register value from "00h" to "FFh"
- REG[09C8h] - add register description "Only one of the LCD Controller Core inputs can be "connected" to the CH1OUT output..."
- REG[0A00h] bit 2 - change the function of this bit
- REG[0A06h] bit 2 - change the function of this bit
- REG[0A84h] - add bits 3 and 2 to register
- REG[0A86h] ~ REG[0A87h] - add these registers
- REG[0A8Ch] ~ REG[0A8Dh] - add these registers
- REG[0B04h] - change default register value to 81h
- REG[0C00h] - add reserved bit 1 and rewrite bit 0 description
- REG[0D04h] bit 0 - add this bit to register
- REG[2025h] - change the default register value to 80h
- REG[2026h] - change the default register value to 02h
- REG[2027h] - change the default register value to 00h
- REG[202Ah] - change the default register value to 04h
- REG[202Bh] - change the default register value to 00h
- REG[3C42h] ~ REG[3C43h] - add these registers
- REG[3C44h] - add this register
- REG[5020h] bits 1-0 - mark these bits as read only
- REG[5024h] bits 1-0 - mark these bits as read only
- REG[5028h] - mark this register as read only
- REG[5029h] - change the default register value to F0h and mark bits 3-0 as read only

X83A-A-001-00 Revision 0.02 - Issued: May 18, 2007

- includes all requested updates

X83A-A-001-00 Revision 0.01 - Issued: April 16, 2007

- created from the S1D13513 Spec Rev 0.09

Chapter 30 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