EPSON
EXCEED YOUR VISION

Epson / E Ink EPD Controller

S4E5B001B000A00 EPD Controller Module

SEIKO EPSON CORPORATION

Rev. 1.2

NOTICE

Table Of Contents

Chapter 1 Introduction

1.1 Scope

This is the Hardware Functional Specification for the S4E5B001B00A00 EPD Controller Module. Included in this document are timing diagrams, AC and DC characteristics, Command descriptions, and power management descriptions. This document is intended for initial system designer references only.

This document is updated as appropriate. Please check for the latest revision of this document before beginning any development. We appreciate your comments on our documentation. Please contact us via email at documentation@erd.epson.com.

1.2 Overview Description

EPSON's S4E5B001B000A00 EPD Controller Module is a complete, high performance, easy to use solution for E Ink EPDs (Electronic Paper Displays). The module includes all necessary digital and analog power circuitry for driving an E Ink panel. The EPD Controller Module consists of Epson's S1D13522 EPD controller, an on-board 26MHz crystal clock source, an 8 Mbit flash memory for command/waveform storage, and a Power Management IC (PMIC) with built in temperature sensor.

The S4E5B001B000A00 EPD Controller Module contains all the complex electronics needed to drive an EPD panel. With this module a customer can build a EPD system on a simple two layer PCB. This module removes the complexity of board design from the customer, making a variety of implementations possible.

The S4E5B001B000A00 EPD Controller Module is the ideal choice for new EPD designs and design upgrades.

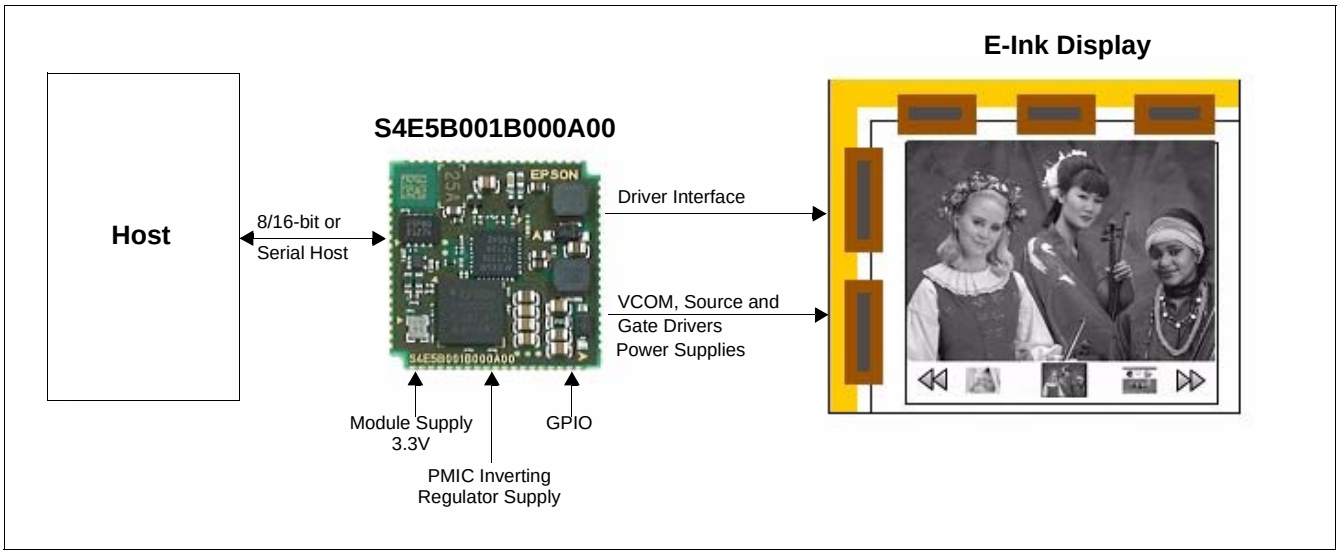


Figure 1-1: S4E5B001B00A00 Overview

Chapter 2 Features

2.1 Direct Source and Gate Driver for Electrophoretic Display

- Up to 4-bit grayscale waveforms (16 level grey-shades)
- Source Driver interfaces to the Micronix MX860 IC and compatible devices
- Gate Driver interfaces to the Sharp LH1692 and compatible devices
- 15-Region Pipeline Technology
  - Auto Pipeline Management for easy implementation
- Auto Waveform Management
  - Automatically determines the best/fastest waveforms based on the display contents
  - Supports Monochrome, 4 level grayscale, 8 level grayscale, 16 level grayscale and one default waveform

2.2 8/16-Bit and Serial Host Interface

- Indirect 8/16-bit Host Bus Interface (Intel 80)
- Serial Host Interface (SPI)
- Registers can be accessed using Command Mode
- DMA compatible memory bus style host interface
- Transparency write support

2.3 Advanced Sequencer Engine

- Sequencer logic executes pre-programmed series of commands request from host.

2.4 Memory

- 2M Bytes of stacked memory
- Sufficient Storage for:

	Panel Size			Memory Storage Configuration
	6” (800x600)	**8” (1024x768)**	**9” (1200x825)**	
Main Display	Up to 5 pages	Up to 2 pages	1 page	4bpp packed
Full PIP Size	Up to 4 pages	Up to 1.3 pages	0	1bpp, 2bpp and 4bpp packed (selectable)
Cursor	4KB available	4KB available	4KB available	1bpp, 2bpp and 4bpp packed (selectable)
Max Framerate	85Hz	85Hz	50Hz	—

2.5 Image Buffer Flexibility

- Host Writes can be rotated counter-clockwise by 90°, 180° or 270°
- New image data can be loaded to the image buffer while display updates are in progress
- Supported Data Formats: 1bpp, 2bpp, 4bpp and 1 byte-per-pixel

2.6 Picture-In-Picture

- Picture in Picture support on separate buffer
  - Independently Programmable Bit-Per-Pixel 1bpp, 2bpp and 4bpp
  - Programmable Look-up-table availability for 1bpp and 2bpp for display bit-per-pixel matching
  - Transparency key pixel value support.
  - Background pixel inversion support.

2.7 Cursor Buffer Support

- Cursor buffer storage in internal memory.
  - Dedicated 4K bytes of cursor buffer storage (in internal memory) - independently programmable bit-per-pixel (1bpp, 2bpp or 4bpp)
  - Extra memory for Multi-Buffered Cursor support.
  - Transparency key pixel value support.
  - Background pixel inversion support.

2.8 Thermal Sensor

- Power Management IC (PMIC) on-chip Thermal Sensor provides automatic temperature compensation.

2.9 Four Regulated Output Voltages for Source- and Gate-Driver Power Supplies

- Two source-driver power supplies can deliver up to 200mA.
- Positive source-driver power supply regulation voltage (VPOS) of +15V.
- Negative source-driver supply voltage (VNEG) tightly regulated to VPOS within ±50mV.
- Two gate-driver power supplies can deliver up to 20mA.
- +22V positive gate-driver power supply (GVDD) and -20V negative gate-driver power supply (GVEE).

2.10 VCOM Power Supply

- Negative output VCOM voltage programmable from -0.5V to -3.05V in 10mV steps.
- Non-volatile memory for storing programmed VCOM voltage, automatically loaded at power-up.

2.11 Controlled Inrush Current During Soft Start

- All outputs are brought up with soft-start control to limit the inrush current.

2.12 Clock Source

- No external clock source required.

Chapter 3 Typical System Implementation

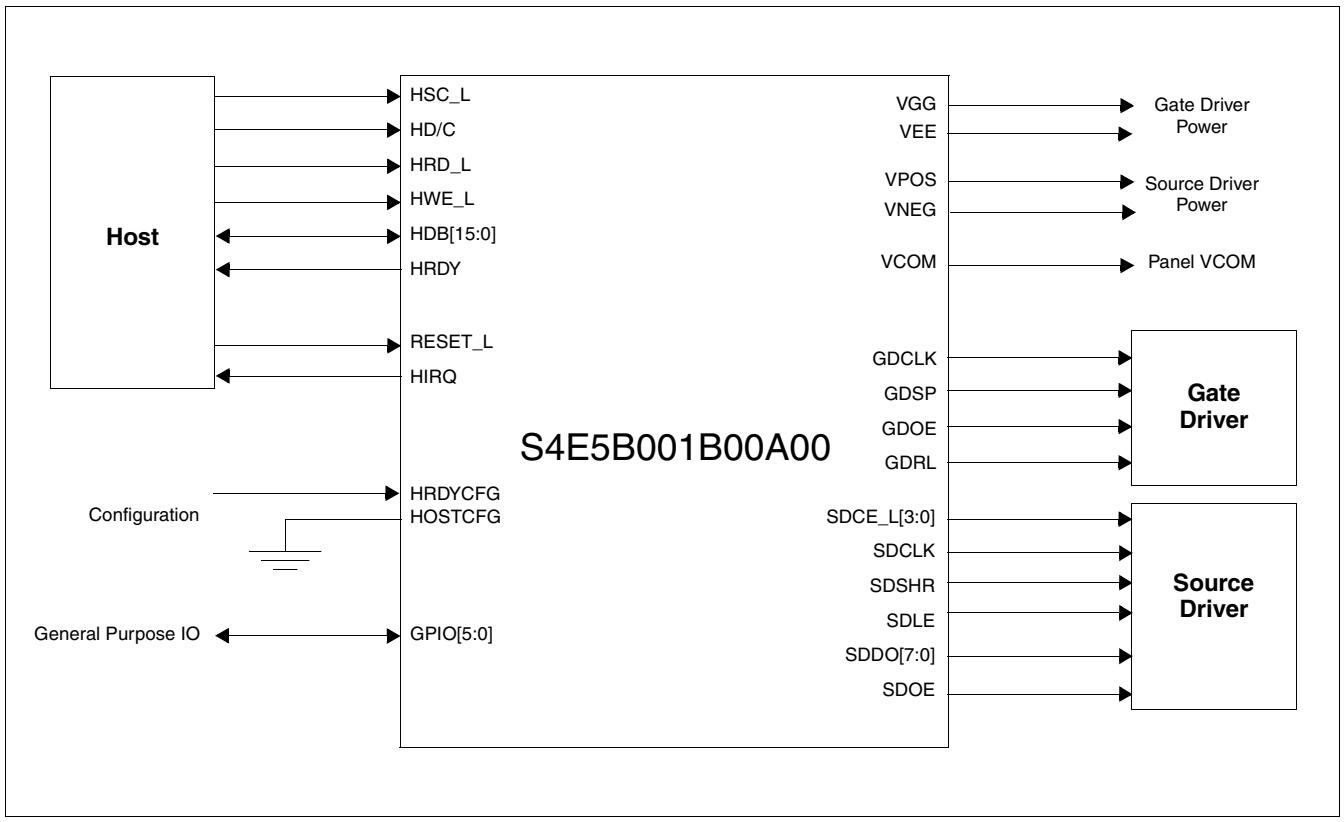


Figure 3-1: Typical 16-Bit Parallel Host Interface System Implementation

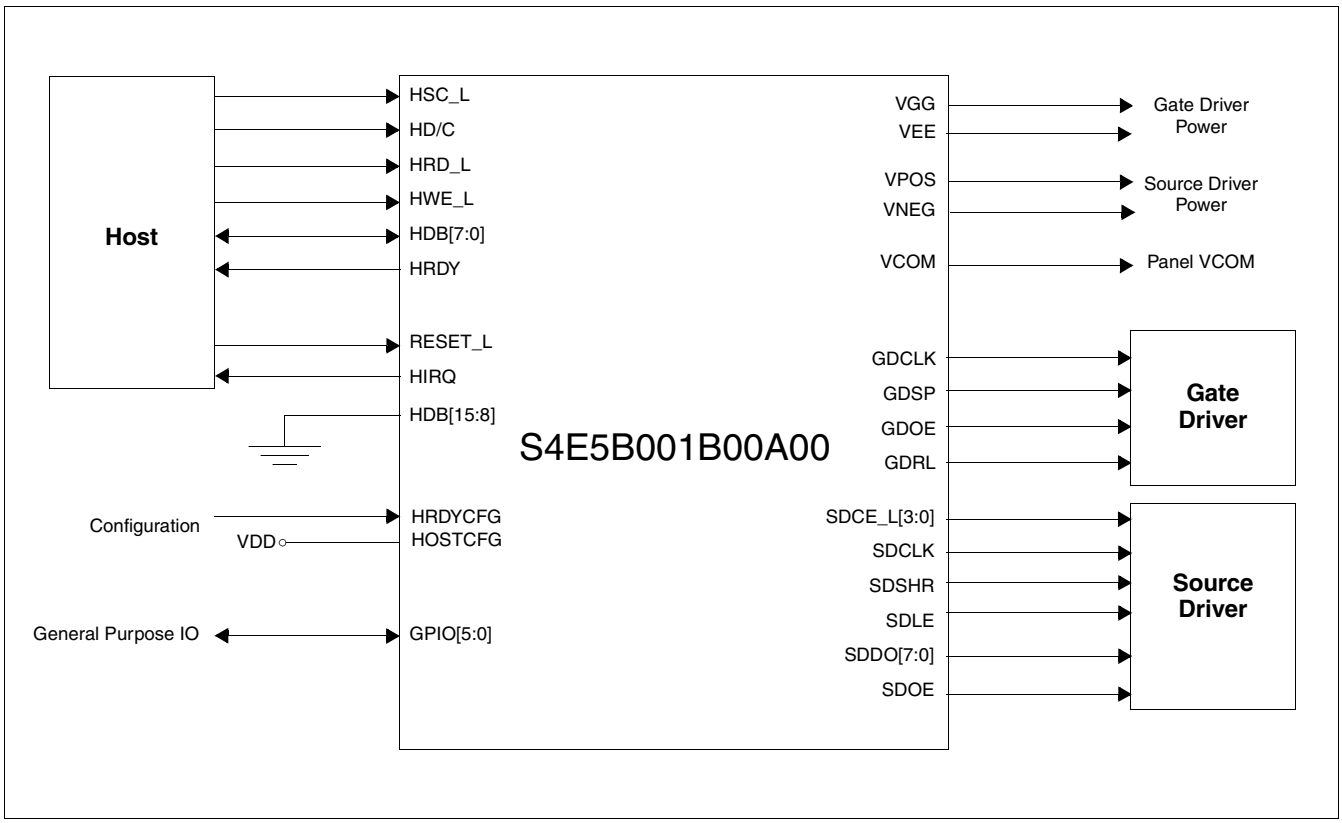


Figure 3-2: Typical 8-Bit Parallel Host Interface System Implementation

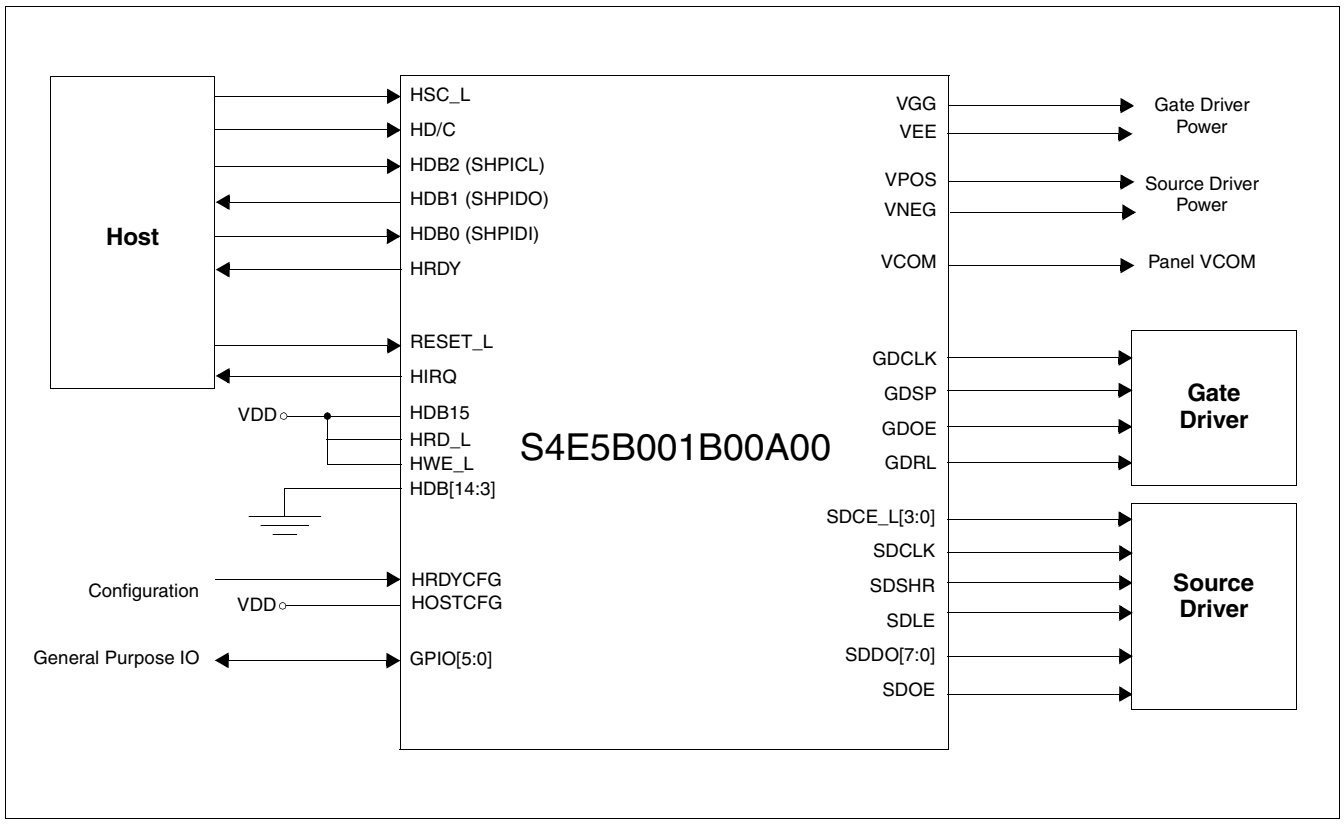


Figure 3-3: Typical Serial (SPI) Host Interface System Implementation

Chapter 4 Pins

4.1 Pinout Diagram

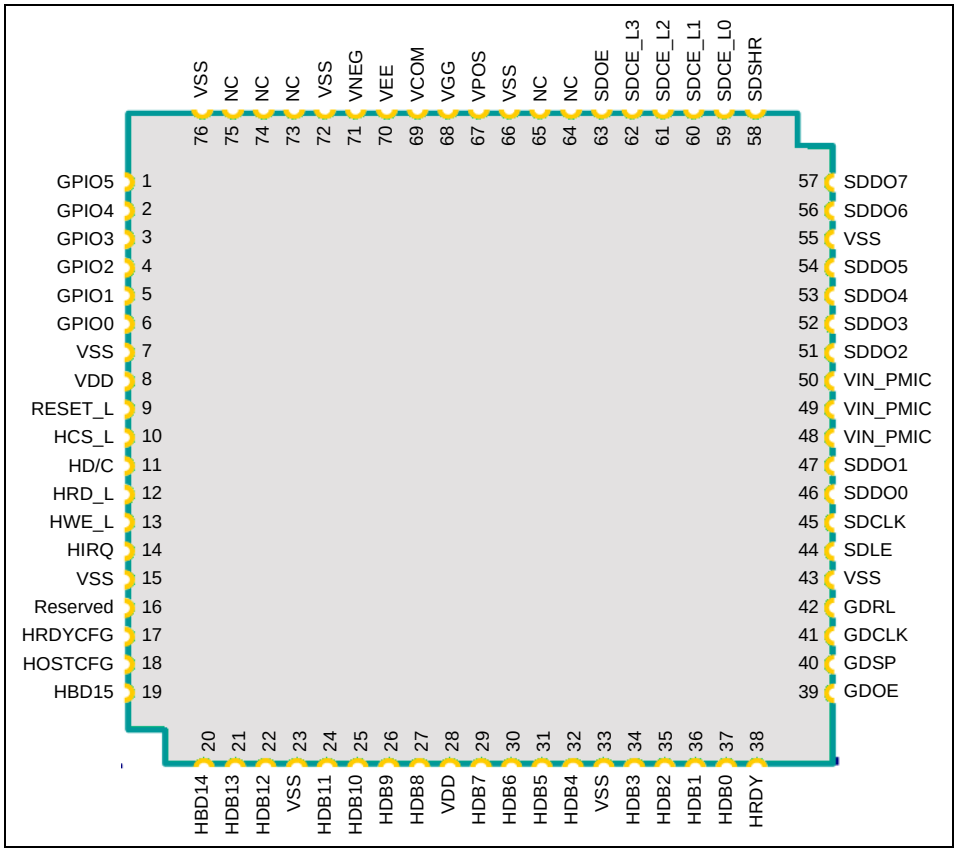


Figure 4-1: Pin Mapping (Top View)

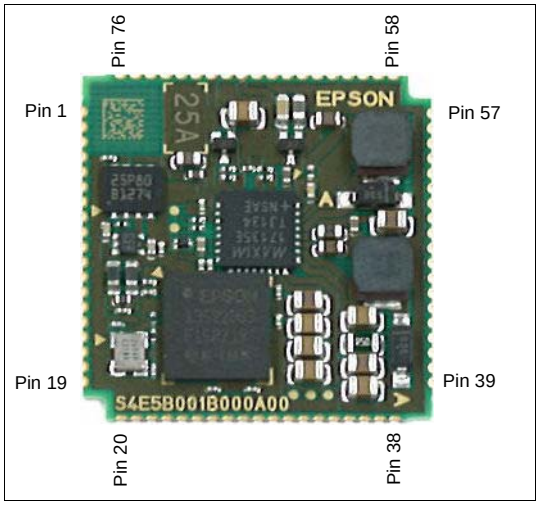


Figure 4-2: Pin Position Identification

4.2 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

Table 4-1: Cell Descriptions

Item	Description
IC	High voltage LVCMOS input buffer
ICS	High voltage LVCMOS Schmitt input buffer
ICD2	High voltage LVCMOS input buffer with pull-down (100kΩ@3.3V)
O	High voltage low noise output buffer (4mA@3.3V)
O4	High voltage low noise output buffer (12mA@3.3V)
OH	High voltage high speed output buffer (4mA@3.3V)
BC	High voltage LVCMOS low noise bi-directional buffer (4mA@3.3V)
BCS	High voltage LVCMOS schmitt low noise bi-directional buffer (4mA@3.3V)
BCD2	High voltage LVCMOS low noise bi-directional buffer (4mA@3.3V) with pull-down (100kΩ@3.3V)
ILTR	Low voltage transparent input buffer
OLTR	Low voltage transparent output buffer
ILTSD	Low voltage test mode control input buffer with pull-down

4.2.1 Shared Host Interface

Table 4-2: Shared Host Interface Pin Descriptions

Pin Name	Type	Pin #	Cell	RESET_L State	Description
RESET_L	I	9	ICS	—	This active low input sets all internal registers to their default states and forces all signals to their inactive states. When unused, this pin should be connected to VDD. For RESET_L timing details, see 6.2, “RESET_L Timing” on page 23.
HIRQ	O	14	O	L	This output pin is the Host IRQ.

4.2.2 Host Interface

These pins are used for the Host interface which can be selected from Intel 80 16-bit, Intel 80 8-bit, or Serial Host based on the setting of the HOSTCFG pin and HDB15.

Table 4-3: Host Interface Pin Descriptions

Pin Name	Type	Pin #	Cell	RESET_L State	Description
HDB[15:0]	IO	19, 20, 21, 22, 24, 25, 26, 27, 29, 30, 31, 32, 34, 35, 36, 37	BC	Z	These input/output pins have multiple functions. • For Intel 80 16-bit (HOSTCFG = 0), these pins are the shared command/parameter lines 15-0. • For Intel 80 8-bit (HOSTCFG = 1 and HDB15 = 0), HDB[7:0] are the shared address/data lines 7-0. HDB[14:8] must be connected to VSS. • For Serial Host (HOSTCFG = 1 and HDB15 = 1), HDB0 is the serial data input, SHPIDI. HDB1 is the serial data output, SHPIDO. HDB2 is the serial clock, SHPICK. HDB[14:3] must be connected to VSS.
HWE_L	I	13	ICS	—	This active low input pin has multiple functions. • For Intel 80 16-bit (HOSTCFG = 0), this pin is the Write Enable signal. • For Intel 80 8-bit (HOSTCFG = 1 and HDB15 = 0), this pin is the Write Enable signal. • For Serial Host (HOSTCFG = 1 and HDB15 = 1), this pin must be connected to VDD.
HRD_L	I	12	ICS	—	This active low input pin has multiple functions. • For Intel 80 16-bit (HOSTCFG = 0), this pin is the Read Enable signal. • For Intel 80 8-bit (HOSTCFG = 1 and HDB15 = 0), this pin is the Read Enable signal. • For Serial Host (HOSTCFG = 1 and HDB15 = 1), this pin must be connected to VDD.
HCS_L	I	10	IC	—	This active low input pin is the Chip Select signal.

Table 4-3: Host Interface Pin Descriptions (Continued)

Pin Name	Type	Pin #	Cell	RESET_L State	Description
HD/C	I	11	IC	—	This input pin selects between Command (0) and Parameter (1).
HRDY	IO	38	BC	H	This active high pin is the Host interface Ready (or WAIT) signal. When not used, this pin should be left unconnected

4.2.3 Gate Driver Interface

These pins are for the Gate Driver interface.

Table 4-4: Gate Driver Interface Pin Descriptions

Pin Name	Type	Pin #	Cell	RESET_L State	Description
GDOE	O	39	O	L	This output pin is the output enable for the Gate Driver.
GDCLK	O	41	O	L	This output pin is the clock for the Gate Driver.
GDSP	O	40	O	H	This output pin is the Gate Driver Start Pulse.
GDRL	O	42	O	L	This output pin is the Gate Driver Right or Left.

4.2.4 Source Driver Interface

These pins are for the Source Driver interface.

Table 4-5: Source Driver Interface Pin Descriptions

Pin Name	Type	Pin #	Cell	RESET_L State	Description
SDCLK	O	45	O4	L	This output pin is the clock for the Source Driver.
SDLE	O	44	O4	L	This output pin is the latch enable for the Source Driver.
SDDO[7:0]	O	57, 56, 54, 53, 52, 51, 47, 46	O4	L	These are the data output pins for the Source Driver.
SDCE_L[3:0]	O	62, 61, 60, 59	O4	H	These output pins are the Source Driver Chip Enables 3-0.
SDSHR	O	58	O4	L	This output pin is the Source Driver Shift Right Enable.
SDOE	O	63	O4	L	This output pin is the output enable for the Source Driver.

4.2.5 Miscellaneous

Table 4-6: Miscellaneous Pin Descriptions

Pin Name	Type	Pin #	Cell	RESET_L State	Description
HRDYCFG	I	17	IC	—	This input pin is used for configuring the S4E5B001B00A00 and must be connected to either VDD or VSS. For a summary of configuration options, see 4.3, “Configuration Pins” on page 20.
HOSTCFG	I	18	IC	—	This input pin is used for configuring the S4E5B001B00A00 and must be connected to either VDD or VSS. For a summary of configuration options, see 4.3, “Configuration Pins” on page 20.
GPIO[5:0]	IO	1, 2, 3, 4, 5, 6	BCD2	Z	These are the General Purpose Input/Output pins. When not used, these pins should be connected to VSS through a high value resistor (50k ~ 100k ohm).
Reserved	IO	16	—	—	This reserved pin must be connected to VSS.
NC	—	64, 65, 73, 74, 75	—	—	These pins must be left unconnected.

4.2.6 Power and Ground

Table 4-7: Power and Ground Pin Descriptions

Pin Name	Type	Pin#	Cell	Power	RESET_L State	Description
VDD	P	8, 28	P	+2.4V ~ +3.6V	—	Module main power supply
VIN_PMIC	P	48, 49, 50	P	+2.7V ~ +5.5V	—	PMIC inverting regulator power supply
VSS	P	7, 15, 23, 33, 43, 55, 66, 72, 76	P	0V	—	Ground

Table 4-8: Power Output Pin Descriptions

Pin Name	Type	Pin#	Cell	Power	RESET_L State	Description
VGG	P	68	P	+22V	—	Gate-driver positive power supply
VEE	P	70	P	-20V	—	Gate-driver negative power supply
VPOS	P	67	P	+15V	—	Source-driver positive power supply
VNEG	P	71	P	-15V	—	Source-driver negative power supply
VCOM	P	69	P	-0.5V ~ -3.05V	—	Panel VCOM power supply

4.3 Configuration Pins

The S4E5B001B00A00 has 3 configuration pins which should be pulled high or low based on the following table.

Table 4-9: Configuration Pin Summary

CNF Pin	1 (Connected to VDD)	0 (Connected to VSS)
HRDYCFG	HRDY only driven when HCS_L is asserted	HRDY is always driven
HOSTCFG	Intel 80 8-bit or Serial Host selected (see Note)	Intel 80 16-bit Host is selected

Note
When HOSTCFG = 1, the HDB15 pin is used to select between the Intel 80 8-bit or Serial Host interfaces. For pin mapping details, see 4.4.1, “Host Interface Modes” on page 20.

4.4 Pin Mapping

4.4.1 Host Interface Modes

The Host interface can be configured to either Intel 80 16-bit, Intel 80 8-bit, or Serial Host based on the setting of the HOSTCFG pin and HDB15. The following table summarizes the Host Interface configurations.

Table 4-10: Host Interface Selection

Pin Name	HOSTCFG = 0	HOSTCFG = 1, HDB15 = 0	HOSTCFG = 1, HDB15 = 1
	16-bit Host	8-Bit Host	Serial Host
HD/C	0 - Command 1 - Parameter	0 - Command 1 - Parameter	0 - Command 1 - Parameter
HDB[7:0]	Shared Command/Parameter[7:0]	Shared Command/Parameter[7:0]	HDB[0] - SHPIDI HDB[1] - SHPIDO HDB[2] - SHPICK HDB[7:3] - Connect to VSS
HDB[15:8]	Shared Command/Parameter[15:8]	Connect to VSS	HDB[15] -Connect to VDD HDB[14:8] - Connect to VSS
HCS_L	Cycle Enable	Cycle Enable	SHPICS_L
HRD_L	Read Enable	Read Enable	Connect to VDD
HWE_L	Write Enable	Write Enable	Connect to VDD
HRDY	Wait line	Wait line	Wait Line

The 8-bit parallel host interface can access commands and write to memory, but memory cannot be read back.

Chapter 5 D.C. Characteristics

5.1 Absolute Maximum Ratings

Table 5-1: Absolute Maximum Ratings

Symbol	Parameter	Rating	Units
VDD	Module Main Supply Voltage	VSS - 0.3 ~ 4.0	V
VIN_PMIC	PMIC Inverting Regulator Supply Voltage	VSS - 0.3 ~ 6.0	V
VIN	Input Signal Voltage	VSS - 0.3 ~ *VDD + 0.5	V
VOUT	Output Signal Voltage	VSS - 0.3 ~ *VDD + 0.5	V
IOUT	Output Signal Current	±10	mA
TSTG	Storage Temperature	-55 ~ 150	°C

Note
V_{SS} = 0 V

5.2 Recommended Operating Conditions

Table 5-2: Recommended Operating Conditions

Symbol	Parameter	Condition	Min	Typ	Max	Units
VDD	Module Main Supply Voltage	VSS = 0 V	2.4	3.3	3.6	V
VIN_PMIC	PMIC Inverting Regulator Supply Voltage	VSS = 0 V	2.7	3.3	5.5	V
VIN	Input Signal Voltage	—	VSS	—	VDD	V
TOPR	Operating Temperature	—	-20	25	85	°C

5.3 Electrical Characteristics

Table 5-3: Electrical Characteristics

Symbol	Parameter	Condition	Min	Typ	Max	Units
ISLP_VIN_PMIC	Sleep mode current for VIN_PMIC	Sleep mode, no floating input for VDD	—	0.011	—	mA
ISLP_VDD	Sleep mode current for VDD	Sleep mode, no floating input	—	0.205	—	mA
IIZ	Input Leakage Current	—	-5	—	5	µA
IOZ	Output Leakage Current	—	-5	—	5	µA
IOH	High Level Output Current	VDD = min VOH = VDD - 0.4 V	-4	—	—	mA
IOH4	High Level Output Current	VDD = min VOH = VDD - 0.4 V	-12	—	—	mA
IOL	Low Level Output Current	VDD = min VOL = 0.4 V	4	—	—	mA
IOL4	Low Level Output Current	VDD = min VOL = 0.4 V	12	—	—	mA
VIH	High Level Input Voltage	LVCMOS Level VDD = max	2.2	—	VDD + 0.3	V
VIL	Low Level Input Voltage	LVCMOS Level VDD = min	-0.3	—	0.8	V
VPOS	VPOS Output Voltage	—	—	15	—	V
VNEG	VNEG Output Voltage	—	—	-15	—	V
VGG	VGG Output Voltage	—	—	22	—	V
VEE	VEE Output Voltage	—	—	-20	—	V
VCOM	VCOM Output Voltage	—	-3.05	-1.77	-0.5	V
IVPOS	VPOS Output Current	—	—	—	200	mA
IVNEG	VNEG Output Current	—	—	—	200	mA
IVGG	VGG Output Current	—	—	—	20	mA
IVEE	VEE Output Current	—	—	—	20	mA
IVCOM	VCOM Output Current	—	70	—	—	mA
RPU	Pull-Up Resistance	VIN = 0 V	50	100	240	kΩ
RPD	Pull-Down Resistance	VIN = VDD	50	100	240	kΩ
CI	Input Pin Capacitance	f = 1MHz, VDD = 0V	—	—	6	pF
CO	Output Pin Capacitance	f = 1MHz, VDD = 0V	—	—	6	pF
CIO	Bi-Directional Pin Capacitance	f = 1MHz, VDD = 0V	—	—	6	pF

Chapter 6 A.C. Timings

6.1 Power-On Sequence

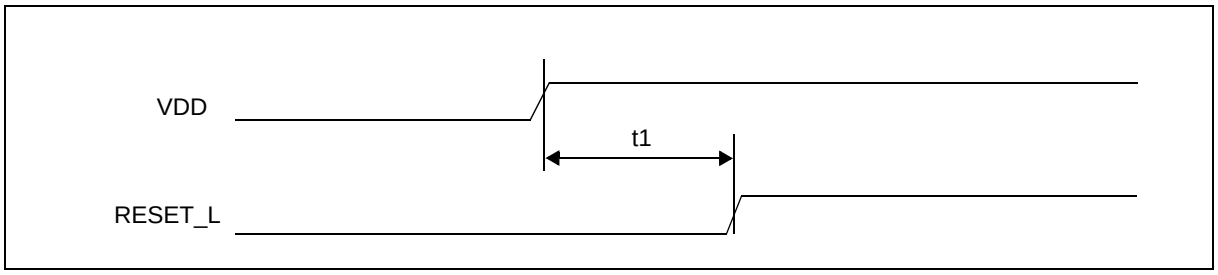


Figure 6-1: Power-On Sequence

Table 6-1: Power-On Sequence

Symbol	Parameter	Min	Max	Units
t1	RESET_L deasserted from VDD on	4	—	ms

6.2 RESET_L Timing

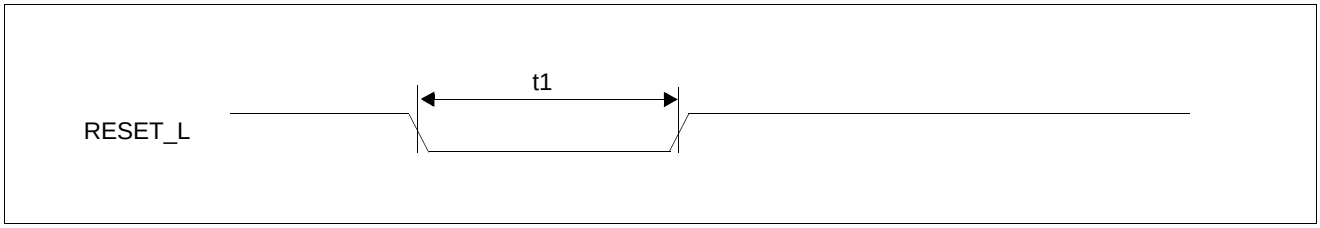


Figure 6-2: S4E5B001B00A00 *RESET_L Timing*

Table 6-2: S4E5B001B00A00 RESET_L Timing

Symbol	Parameter	Min	Max	Units
t1	Active Reset Pulse Width	4	—	ms

6.3 Host Interface Timing

6.3.1 Intel 80 Host Interface Timing

The following section provides AC timing information for the Intel 80 16-bit Host interface, which is selected when HOSTCFG = 0, and Intel 80 8-bit Host interface, which is selected when HOSTCFG = 1 and HDB15 = 0.

For a detailed description of the Host Interface pins, see Chapter 4.2.2, “Host Interface” on page 17.

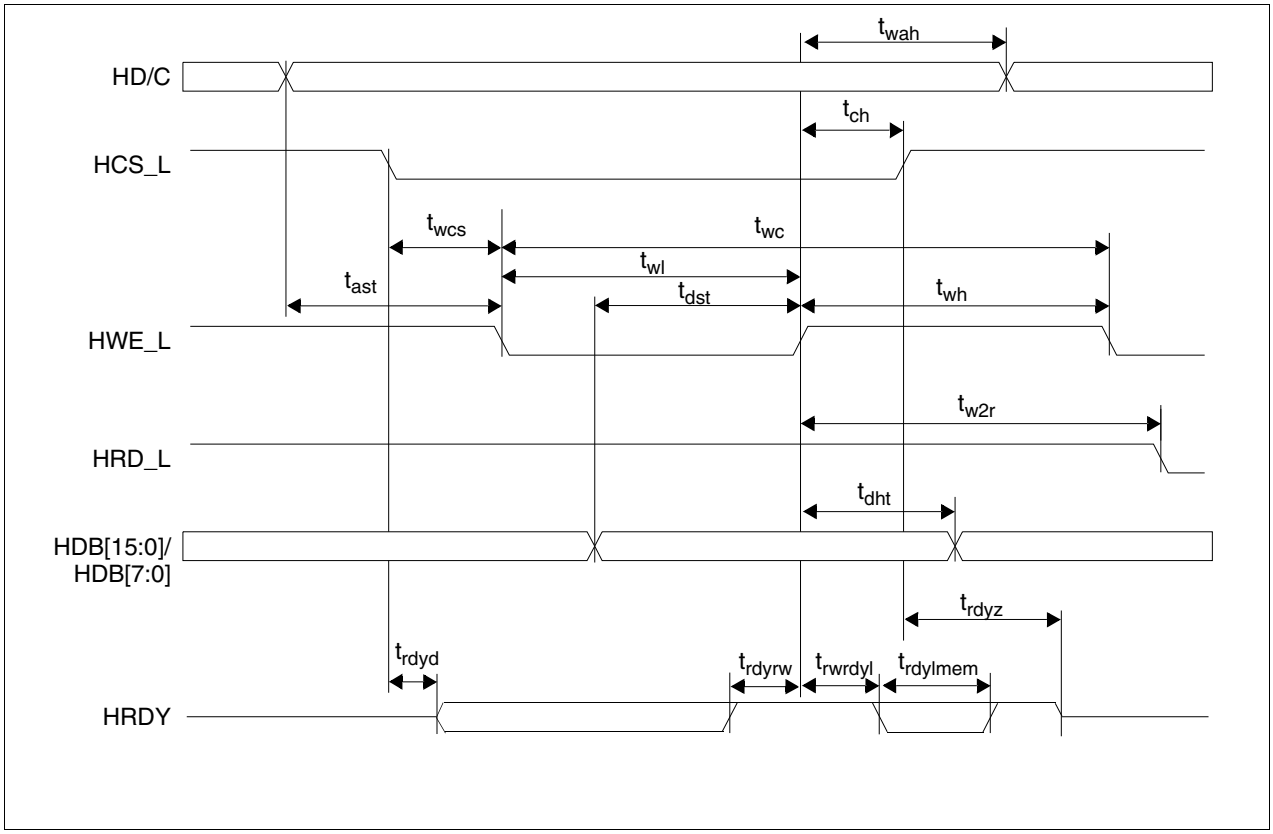


Figure 6-3: Intel 80 Host Interface Write Timing

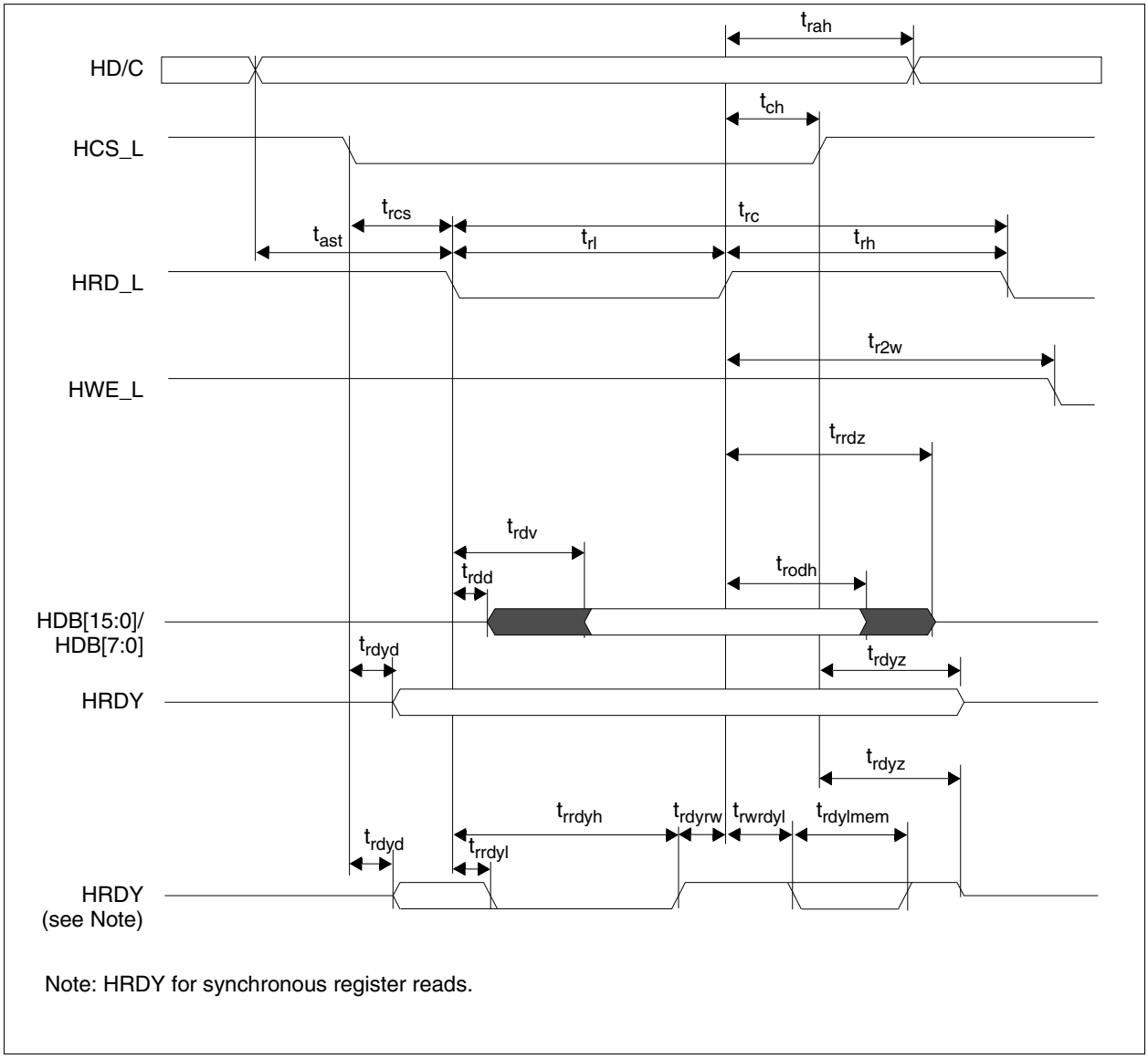


Figure 6-4: Intel 80 Host Interface Read Timing

Table 6-3: Intel 80 Host Interface AC Timing

Signal	Symbol	Parameter	Min	Max	Units	Description
HD/C	t_{ast}	Address setup time (read/write)	0	—	ns	
	t_{wah}	Address hold time (write)	3.7	—	ns	
	t_{rah}	Address hold time (read)	2.9	—	ns	
HCS_L	t_{wcs}	Chip Select setup time to HWE_L falling edge	1.8	—	ns	
	t_{rcs}	Chip Select setup time to HRD_L falling edge	2.2	—	ns	
	t_{ch}	Chip Select hold time (read/write)	3.4	—	ns	
HWE_L	t_{wl}	Pulse low duration	(Note 1)	t_{wc} - t_{wh}	ns	
	t_{wh}	Pulse high duration	(Note 1)	t_{wc} - t_{wl}	ns	
	t_{wc}	Write cycle for Registers	76.9	—	ns	
		Write cycle for Memory	76.9	—	ns	
	t_{w2r}	HWE_L rising edge to HRD_L falling edge	30.76	—	ns	
HRD_L	t_{r2w}	HRD_L rising edge to HWE_L falling edge	0	—	ns	
	t_{rc}	Read cycle for Registers	76.9	—	ns	
		Read cycle for Memory	76.9	—	ns	
	t_{rl}	Pulse low duration (for Registers)	61.52	t_{rc} - t_{rh}	ns	
		Pulse low duration (for Memory)	(Note 1)	t_{rc} - t_{rh}	ns	
	t_{rh}	Pulse high duration	(Note 1)	t_{rc} - t_{rl}	ns	
HDB[15:0]	t_{dst}	Write data setup time (16-bit bus)	7.1	—	ns	
	t_{dst}	Write data setup time (8-bit bus)	1.7	—	ns	
	t_{dht}	Write data hold time (16-bit bus)	4.8	—	ns	
	t_{dht}	Write data hold time (8-bit bus)	4.7	—	ns	
	t_{rodh}	Read data hold time from HRD_L rising edge	3.2	—	ns	
	t_{rrdz}	HRD_L rising edge to HDB[15:0] Hi-Z (16-bit bus)	3.6	9.5	ns	
	t_{rrdz}	HRD_L rising edge to HDB[7:0] Hi-Z (8-bit bus)	3.9	9.6	ns	
	t_{rdv}	HRD_L falling edge to HDB[15:0]/HDB[7:0] valid for Registers	—	83.52	ns	CL=30pF
		HRD_L falling edge to HDB[15:0]/HDB[7:0] valid for Memory (if t_{rc} not met)	—	83.42	ns	
	t_{rdd}	HRD_L falling edge to HDB[15:0] driven (16-bit bus)	4.4	17.0	ns	CL=30pF
	t_{rdd}	HRD_L falling edge to HDB[7:0] driven (8-bit bus)	3.5	7.1	ns	CL=30pF
HRDY	t_{rdyd}	HCS_L falling edge to HRDY driven (16-bit bus)	3.7	13.8	ns	CL=30pF
	t_{rdyd}	HCS_L falling edge to HRDY driven (8-bit bus)	3.8	11.5	ns	CL=30pF
	t_{rdyz}	HCS_L rising edge to HRDY Hi-Z (16-bit bus)	2.0	5.4	ns	CL=30pF
	t_{rdyz}	HCS_L rising edge to HRDY Hi-Z (8-bit bus)	2.4	5.5	ns	CL=30pF
	t_{rrdyl}	HRD_L falling edge to HRDY low	—	16.7	ns	CL=30pF
	t_{rrdyh}	HRD_L falling edge to HRDY high	—	85.12	ns	CL=30pF
	t_{rdyrw}	HRDY rising edge to HRD_L/HWE_L rising edge	146.4	—	ns	CL=30pF
	t_{rwrdyl}	HWE_L rising edge to HRDY falling edge (for command accesses)	—	16.4	ns	CL=30pF
		HRD_L/HWE_L rising edge to HRDY falling edge (for memory accesses)	—	15.8	ns	CL=30pF
	$t_{rrdylmem}$	HRDY low period for memory read/write	—	61.52	ns	CL=30pF

1. The pulse width high and pulse width low parameters have no required minimum duration. However, a pulse must occur for the cycle to take place and must adhere to the formula as defined by the maximum value.

6.3.2 Serial Host Interface Timing

The following section provides AC timing information for the Serial Host interface which is selected when HOSTCFG = 1 and HDB15 = 1.

For a detailed description of the Host Interface pins, see Chapter 4.2.2, “Host Interface” on page 17.

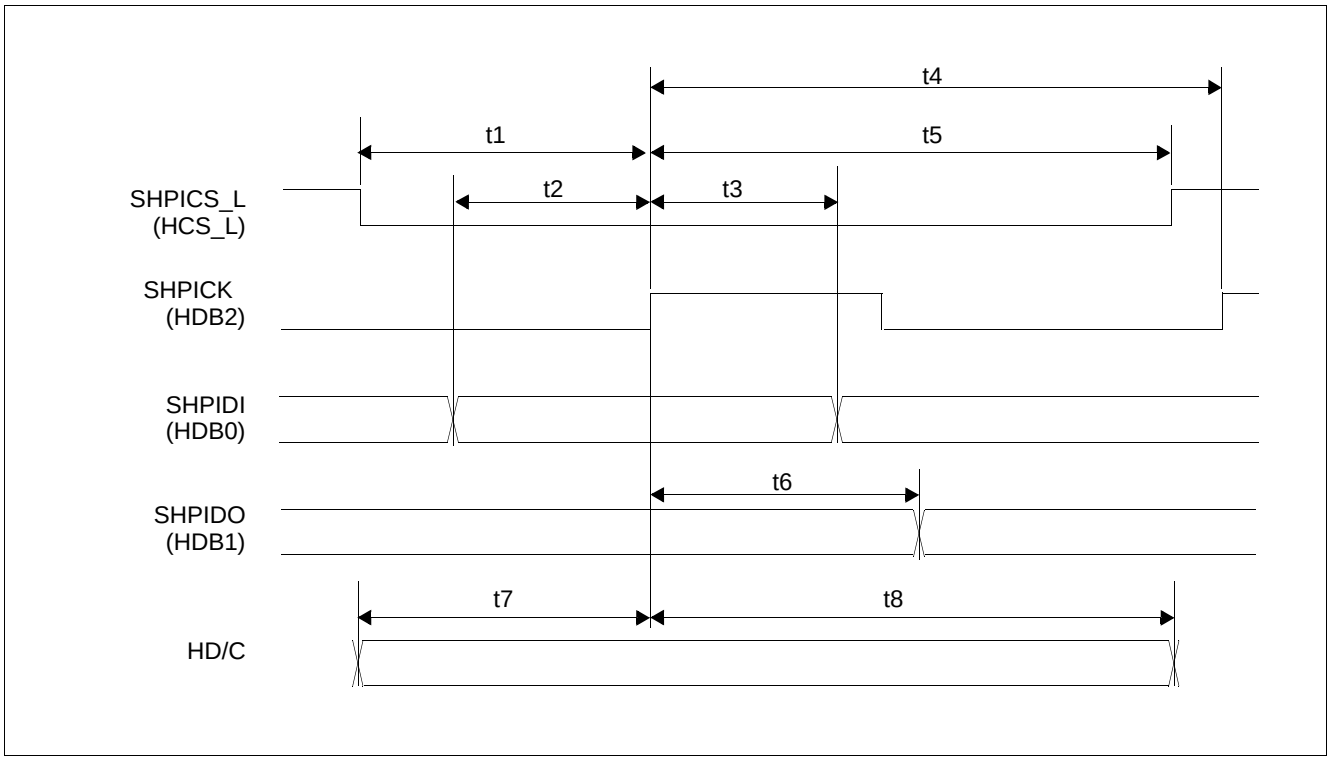


Figure 6-5: Serial Interface SHPICK Timing

Table 6-4: Serial Interface SHPICK Timing

Symbol	Parameter	Min	Max	Unit
t1	SHPICS_L active to positive edge of SHPICK	2	—	ns
t2	SHPIDI setup to positive edge of SHPICK	9	—	ns
t3	SHPIDI hold from positive edge of SHPICK	2	—	ns
t4	SHPICK period	18	—	ns
t5	Positive edge of SHPICK to SHPICS_L high	1	—	ns
t6	Positive edge of SHPICK to SHPIDO hold	3	16	ns
t7	HD/C setup to positive edge of SHPICK	3	—	ns
t8	HD/C hold from positive edge of SHPICK	1	—	ns

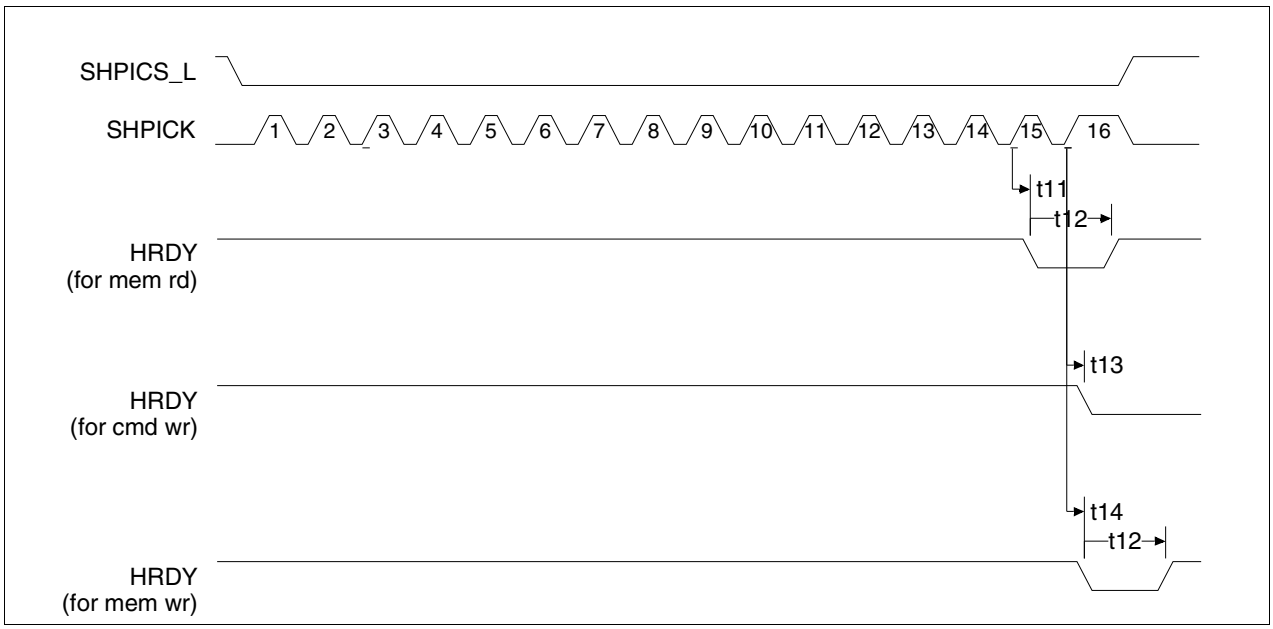


Figure 6-6: Serial Host Interface HRDY Timing

Table 6-5: Serial Interface HRDY Timing

Symbol	Parameter	Min	Max	Unit
t11	Positive edge of the 15th SHPICK to HRDY falling edge (for memory read)	—	16	ns
t12	HRDY low period for memory read/write	—	61.52	ns
t13	Positive edge of the 16th SHPICK to HRDY falling edge (for command access)	—	16	ns
t14	Positive edge of the 16th SHPICK to HRDY falling edge (for memory write)	—	16	ns

Chapter 7 Power Management

7.1 Power Management State Description

The S4E5B001B00A00 controller has the following power states:

- OFF State:
  Power for the S4E5B001B00A00 and internal Memory is off. Upon power-up, the Host must issue a hardware reset and an INIT_SYS_RUN command to initialize the S4E5B001B00A00.
- Run State:
  The S4E5B001B00A00 is ready for operation. In this state the display can be updated.
- Standby State:
  The S4E5B001B00A00 is in power save mode. The Host should not access memory while in this state.
- Sleep Mode:
  The S4E5B001B00A00 is in power save mode. Sleep Mode initiates the Power Pin power down cycles. The Host should not access memory while in this state.

The following table summarizes the S4E5B001B00A00 functionality for each power state.

Table 7-1: S4E5B001B00A00 Power State Functionality Summary

Power Mode	S4E5B001B00A00 Controller State	S4E5B001B00A00 PLL State	PMIC State	Memory Data Retained	Power Consumption
OFF	Unknown	Unknown	Unknown	No	NA
Run	Active All clocks active	Active	ON	Yes	Low or High (depending on host and display activity)
Standby	Power Save Mode All module clocks gated off PLL is running	Active	ON	Yes	Lower
Sleep	Power Save Mode Power Pin cycle off PLL off	Powered-Down	OFF	Yes	Lowest

The following figure shows how to transition between S4E5B001B00A00 power states.

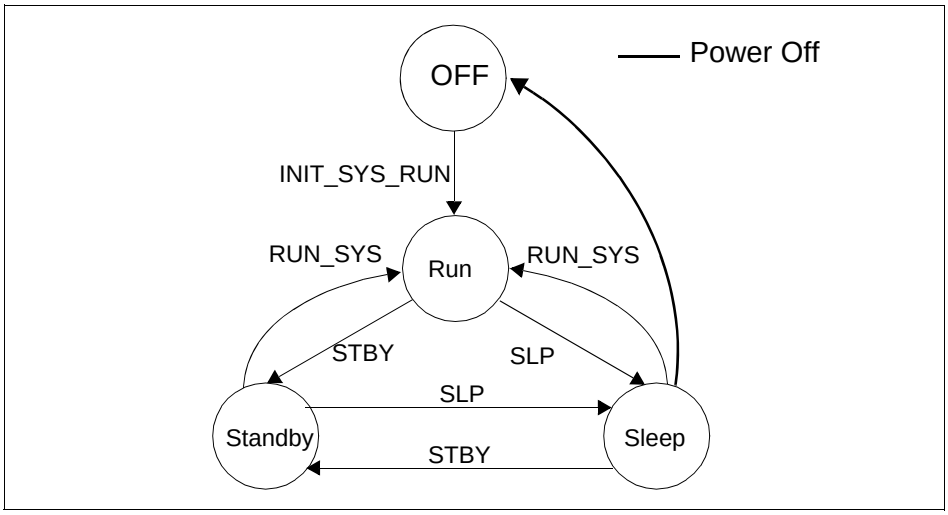


Figure 7-1: State Transition Diagram

To transition between S4E5B001B00A00 power states the following steps are required.

Table 7-2: State Transition Requirements

Current State	Next State Requirements			
	Off	**Run**	**Standby**	**Sleep**
OFF	NA	1. Power on Reset 2. Run Cmd INIT_SYS_RUN	Not Possible.	Not Possible.
Run	1. Host Save Memory Contents 2. Run Cmd SLP 3. Power off	NA	1. Run Cmd STBY	1. Run Cmd SLP
Standby	1. Run Cmd SLP 2. Power off	1. Run Cmd RUN_SYS	NA	1. Run Cmd RUN_SYS
Sleep	1. Power off	1. Run Cmd RUN_SYS	1. Run Cmd STBY	NA

The estimated time required to transition between S4E5B001B00A00 power states is shown in the following figure and summarized in the following table.

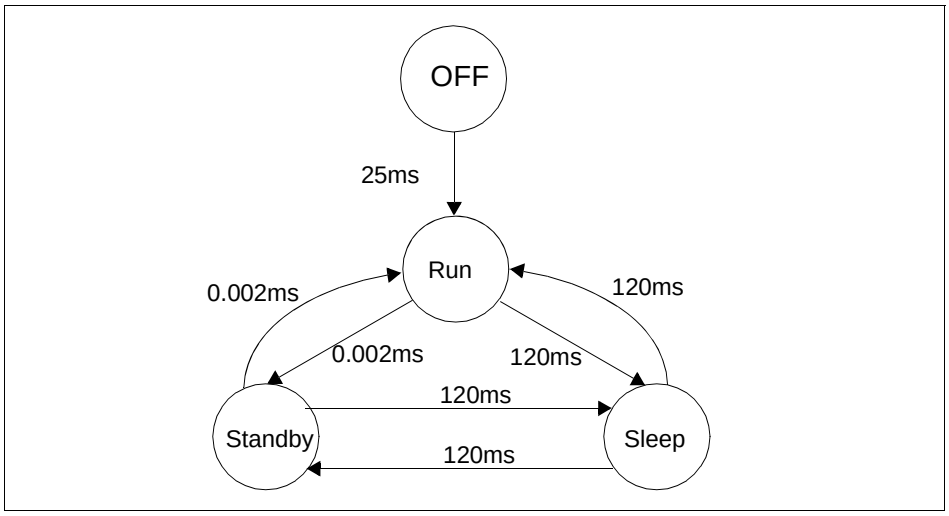


Figure 7-2: State Transition Estimated Time

Table 7-3: State Transition Estimated Time

Current State	Estimated Transition Time			
	Off	**Run**	**Standby**	**Sleep**
OFF	NA	25ms	NA	NA
Run	NA	NA	0.002ms	120ms
Standby	NA	0.002ms	NA	120ms
Sleep	NA	120ms	120ms	NA

Chapter 8 Host Interface

8.1 Host Cycle Sequences

8.1.1 Command and Parameter Cycle

A typical cycle consists of a Command and a variable number of parameters depending on the expected parameter count for each specific command. The following figures shows a typical cycle.

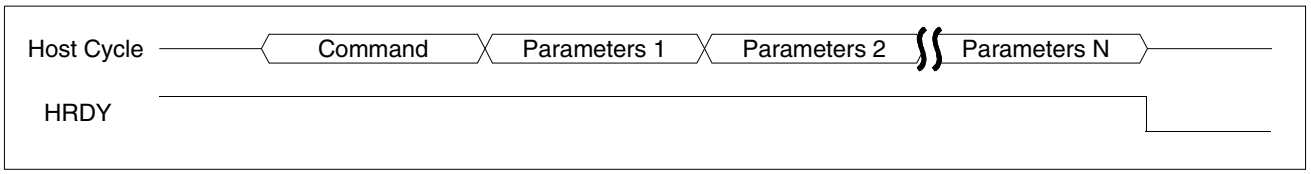


Figure 8-1: Command Mode Host Cycle Example

The HRDY line is deasserted (low) after the last parameter. When the sequencer is ready to accept new commands, HRDY is asserted (high) again.

8.1.2 Memory Access Combination Cycle

Commands that require Host memory access require a combination of Command/Parameter and Register Read/Write cycles. The HRDY line will deassert (low) during the memory read/write data cycles.

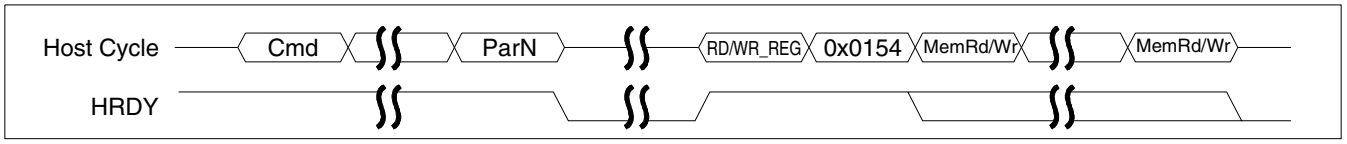


Figure 8-2: Memory Access Combination Cycle

8.2 HRDY (Wait Line) Usage

When a command is issued, the command sequencer must execute a set of instruction codes based on the specific command. The execution time required for each command depends on the number of instruction codes that must be completed.

Once a command is issued, the host interface must not issue any new commands until HRDY is asserted High. If new Commands are inserted during this time, it will be ignored.

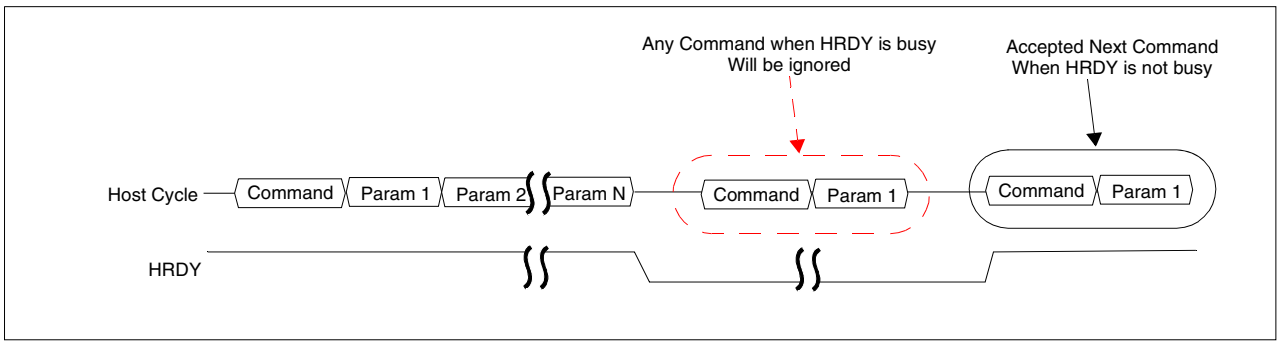


Figure 8-3: Command Mode Host Cycle Example for Host with HRDY

If the HRDY line is not used, the Host can poll the Host Interface Busy Status bit, REG[000Ah] bit 5, using the RD_REG command.

The following figure shows a typical command/parameter sequence using Host Interface Busy Status polling.

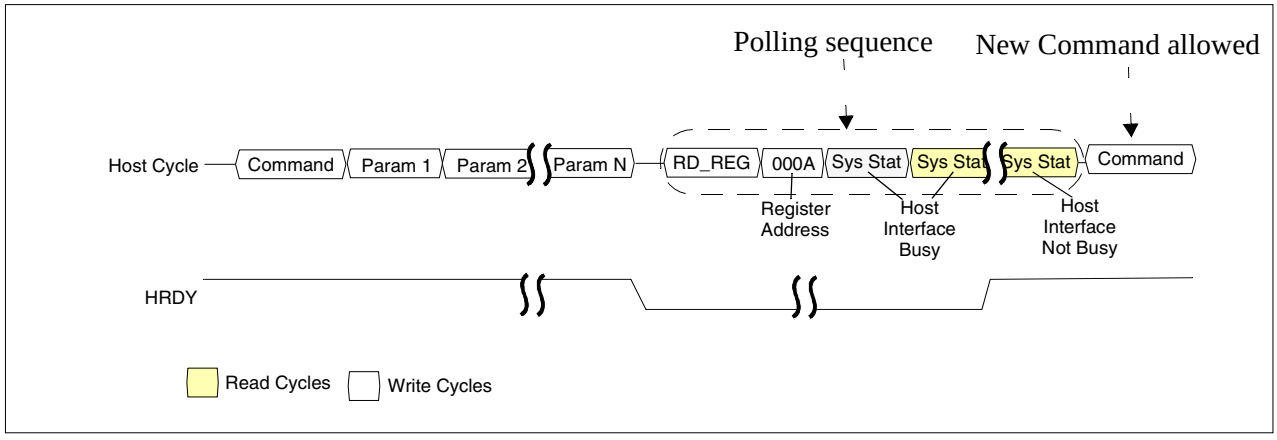


Figure 8-4: Command Mode Host Cycle Example for Host without HRDY

8.3 Intel 80 16-Bit Host Operations

The following figure shows a typical command mode operation with parameter/data writes only.

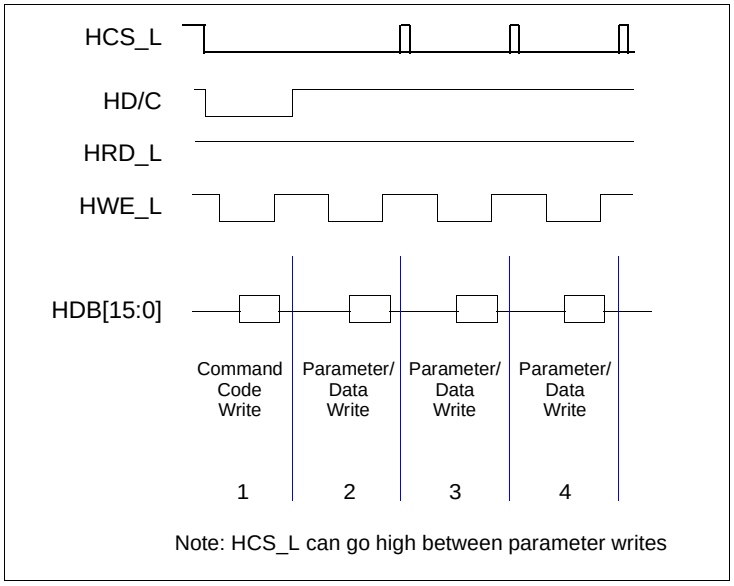


Figure 8-5: Command Mode Operation - Parameter/Data Writes Only

The following figure shows a typical command mode operation for reading register data.

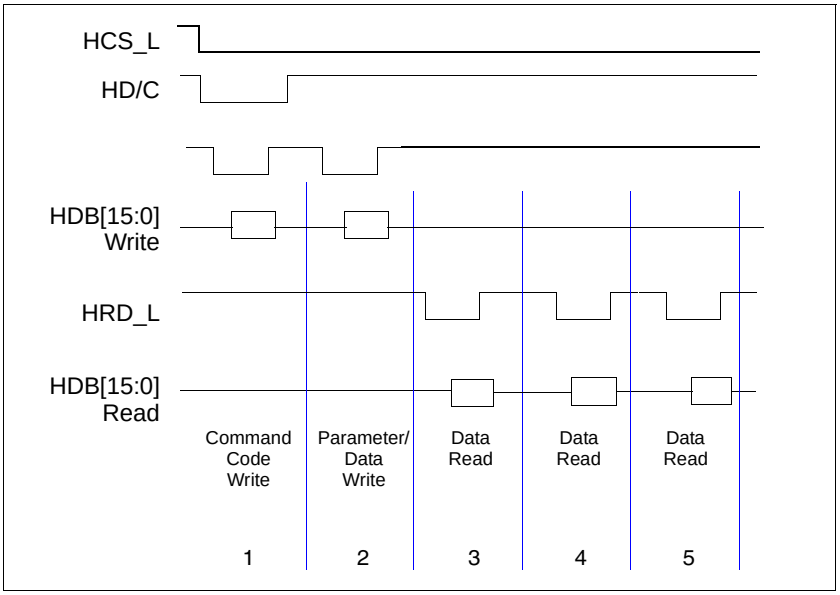


Figure 8-6: Command Mode Operation - Register Read Data

8.4 Intel 80 8-Bit Host Operations

The following figure shows a typical command mode operation with parameter/data writes only.

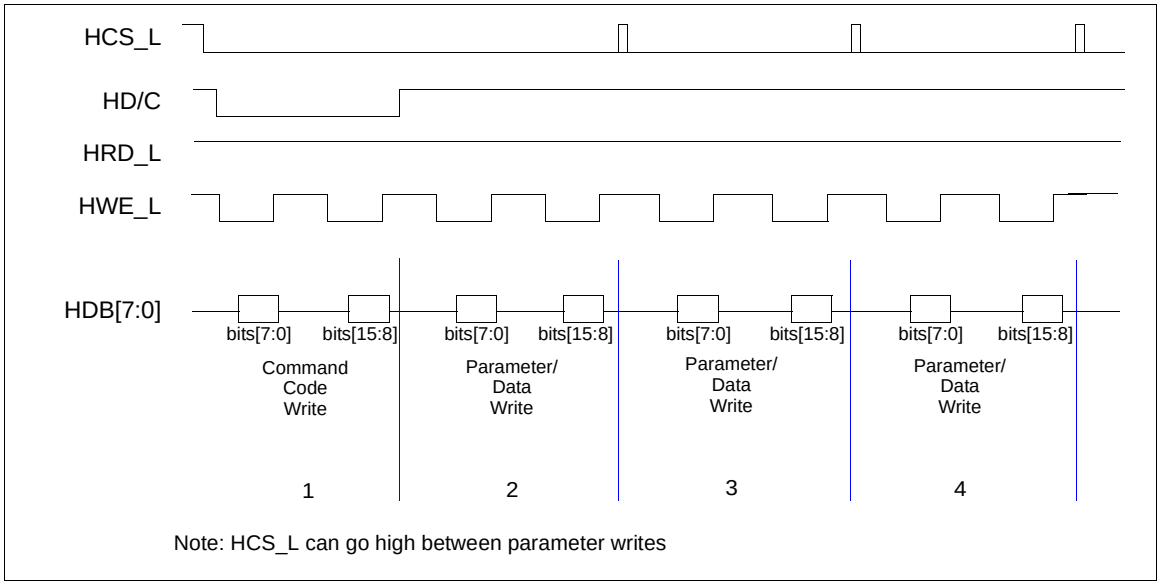


Figure 8-7: Command Mode Operation - Parameter/Data Writes Only

The following figure shows a typical command mode operation for reading register data.

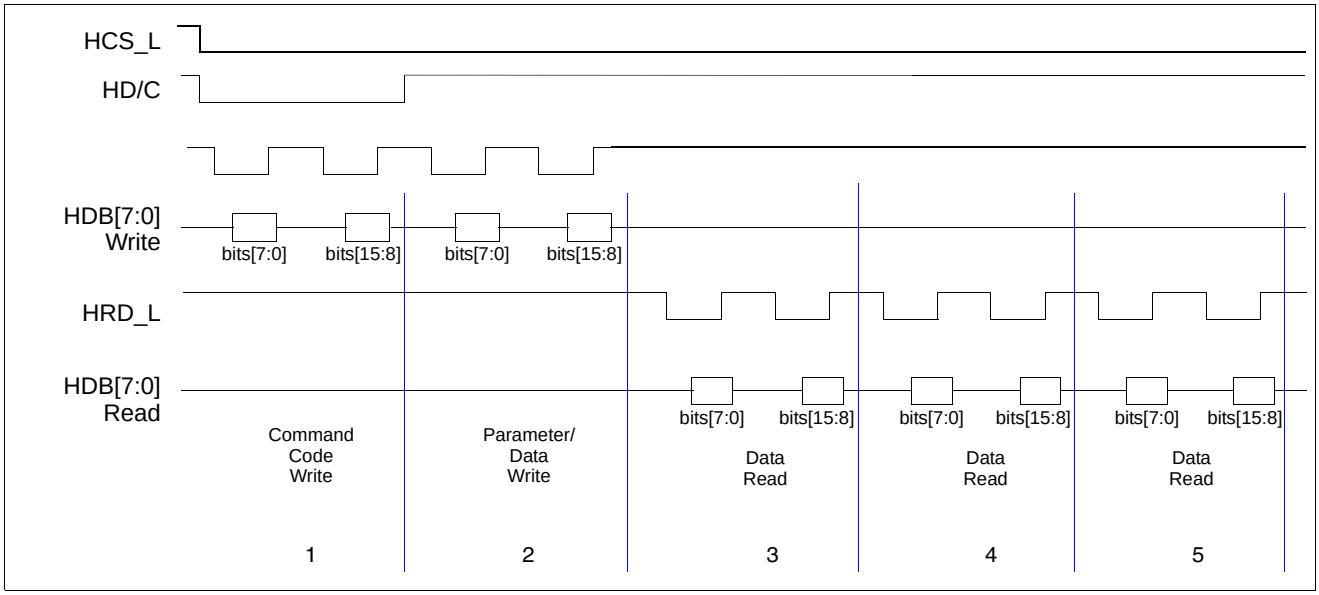


Figure 8-8: Command Mode Operation - Register Read Data

8.5 Serial Host Operations

8.5.1 Protocol Timing

Following diagram shows Serial Host Protocol timing. At the beginning of data transfer, SHPICS_L goes low and SHPICK begins toggling. The write data will be captured at the positive edge of SHPICK. The read data will begin to output with a small delay (see parameter t6 in Serial Host timing) from the rising edge of SHPICK which will be captured by the host at the positive edge of SHPICK.

There is a additional Command/Data pin (HD/C) to select command or data phase. This HD/C pin can be implemented by Host's GPIO.

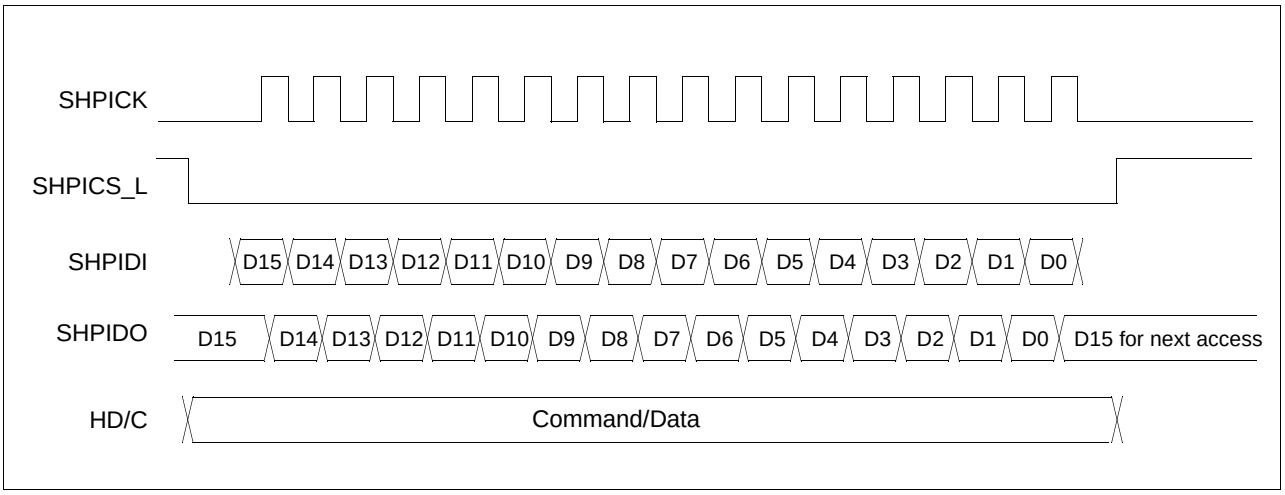


Figure 8-9: Serial Host Protocol timing

8.5.2 Command Transfers

The following figure shows the command transfer sequence for the S4E5B001B00A00 Serial Host. The command sequence is composed of two 16-bit phases.

1. Command phase
2. Data/Parameter Phase (Write or Read)

The command/data protocol is the same as for the 8/16-bit Host. Commands are sent from Host before the Data Phase. HD/C must be Low during the Command phase and must be High during the Data/Parameter phase. The Data/Parameter direction is determined by decoding the command during command phase. For a list of commands and their associated data/parameters, see Chapter 8.6, “Command List” on page 38.

All commands except for the RD_REG command (0x0010) use the command sequence shown below. Note that the data/parameter phase is not required for all commands.

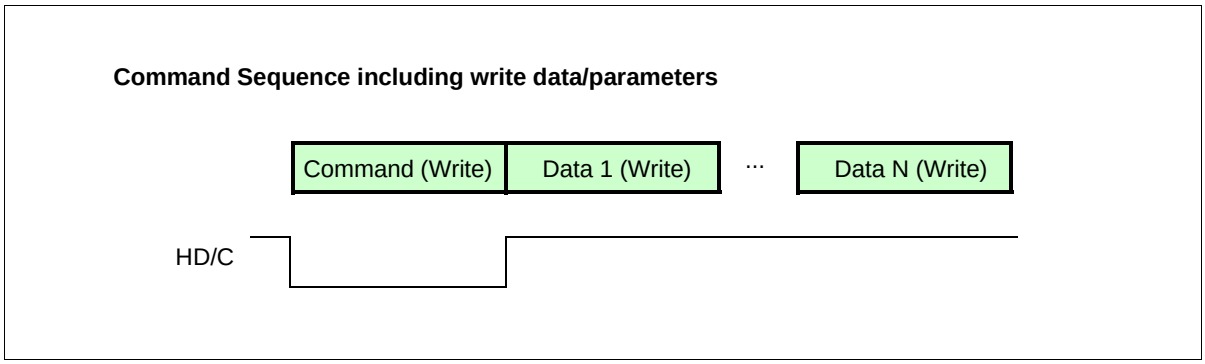


Figure 8-10: Serial Host Write Command Sequence

The RD_REG command (0x0010) uses the following command sequence which includes a Dummy Data Read cycle before the actual Read Data is available.

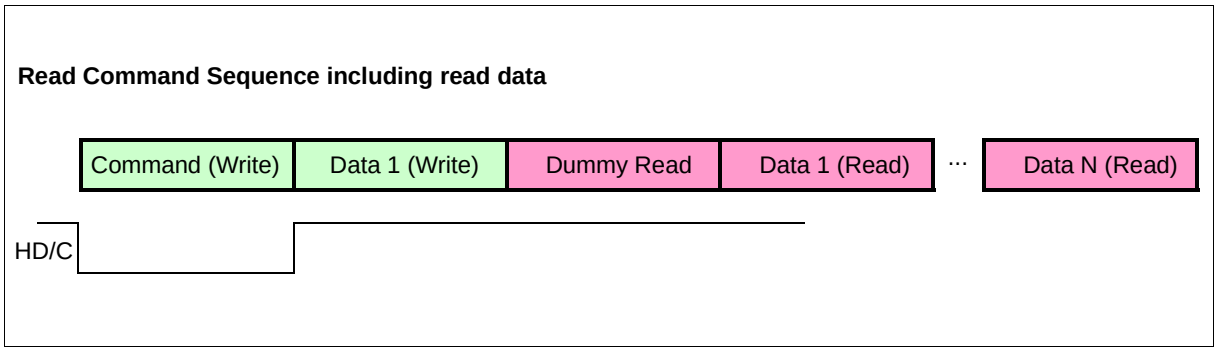


Figure 8-11: Serial Host Read Command Sequence

8.6 Command List

The following commands are applicable for the 16-bit Indirect Interface.

Table 8-1: Host Interface Command Summary

Code 16-bit Indirect - 2 bytes	Command	Parameters					Description
		1	2	3	4	5	
System Commands							
0x00	**Reserved**	—	—	—	—	—	Reserved
0x01	INIT_PLL	Fixed Value	Fixed Value	Fixed Value	—	—	and Go Into Standby Mode. This command should only be used when auto-load from flash is disabled, CNF4 = 1.
0x02	RUN_SYS	—	—	—	—	—	Got to Run Mode
0x04	STBY	—	—	—	—	—	Go to Standby Mode
0x05	SLP	—	—	—	—	—	Go to Sleep Mode
0x06	INIT_SYS_RUN	—	—	—	—	—	Initialize System and Go into Run State
0x07	—	—	—	—	—	—	
0x08	—	—	—	—	—	—	
0x09	**Reserved**	—	—	—	—	—	Reserved
0x0A	**Reserved**	—	—	—	—	—	Reserved
0x0B	INIT_ROTMODE	ROTMODE	—	—	—	—	Initialize Rotation Mode Timings
0x0C	**Reserved**	—	—	—	—	—	Reserved
0x0E	—	—	—	—	—	—	
0x0F	—	—	—	—	—	—	
Register and Memory Access Commands							
0x10	RD_REG	REGADDR[15:0]	RDATA[15:0]	—	—	—	Read Register
0x11	WR_REG	REGADDR[15:0]	WDATA[15:0]	—	—	—	Write Register
0x12	—	—	—	—	—	—	
0x13	—	—	—	—	—	—	
PIP and Cursor Init Commands							
0x14	PIP_DISABLE	—	—	—	—	—	Disable the PIP window
0x15	PIP_ENABLE	ARG[15:0]	HSIZE	VSIZE	PIP Pipeline[15:0]	—	Configure and enable the PIP window
0x16	PIP_ADRCFG	MA[15:0]	MA[25:16]	—	—	—	PIP Data Storage Location
0x17	PIP_XYSETUP	XSTART	YSTART	—	—	—	Setup Position on screen
0x18	CSR_MAINCFG	ARG[15:0]	HSIZE	VSIZE	CSR Pipeline[15:0]	—	Initialize Cursor
0x19	CSR_XYSETUP	XSTART	YSTART	—	—	—	Setup CSR and Position on screen
0x1A	CSR_ADRCFG	ARG[15:0]	—	—	—	—	Cursor Address Pointer Setup
Burst Access Commands							
0x1C	BST_RD_SDR	MA[15:0]	MA[25:16]	BC[15:0]	BC[25:16]	—	Start Burst Read Memory Memory
0x1D	BST_WR_SDR	MA[15:0]	MA[25:16]	BC[15:0]	BC[25:16]	—	Start Burst Write Memory Memory
0x1E	BST_END_SDR	—	—	—	—	—	Burst End
0x1F	—	—	—	—	—	—	
Image Loading Commands							
0x20	LD_IMG	ARG[15:0]	—	—	—	—	Load Image Full
0x22	LD_IMG_AREA	ARG[15:0]	XSTART[11:0]	YSTART[11:0]	WIDTH[12:0]	HEIGHT[12:0,	Load Image Area With parameters
0x23	LD_IMG_END	—	—	—	—	—	Load Image End
0x25	LD_IMG_SETADR	MA[15:0]	MA[31:16]	—	—	—	Set Load Image Manual Address
0x26	LD_IMG_DSPEADR	—	—	—	—	—	Set Load Image to use Display Engine's Address
Polling Commands							
0x28	WAIT_DSPE_TRG	—	—	—	—	—	Wait For Display Engine Trigger Done

Table 8-1: Host Interface Command Summary

Code 16-bit Indirect - 2 bytes	Command	Parameters					Description
		1	2	3	4	5	
0x29	WAIT_DSPE_FREND	—	—	—	—	—	Wait For Display Engine Frame End
0x2A	WAIT_DSPE_LUTFREE	—	—	—	—	—	Wait For Display Engine At least 1 Pipeline is Free
0x2B	**Reserved**	—	—	—	—	—	Reserved
Waveform Update Commands							
0x30	RD_WFM_INFO	MA[15:0]	MA[31:16]	—	—	—	Read Waveform Information
0x32	UPD_INIT	—	—	—	—	—	Panel Update Buffer Initialize
0x33	UPD_FULL	ARG[15:0]	—	—	—	—	Panel Update Buffer Full
0x34	UPD_FULL_AREA	ARG[15:0]	XSTART[11:0]	YSTART[11:0]	WIDTH[12:0]	HEIGHT[12:0]	Panel Update Buffer Full Area
0x35	UPD_PART	ARG[15:0]	—	—	—	—	Panel Update Buffer Partial
0x36	UPD_PART_AREA	ARG[15:0]	XSTART[11:0]	YSTART[11:0]	WIDTH[12:0]	HEIGHT[15:0]	Panel Update Buffer Partial Area
0x37	**Reserved**	—	—	—	—	—	Reserved
0x38	UPD_SET_IMGADR	ADR[15:0]	ADR[31:16]	—	—	—	Set Image Buffer Start Address
0x39	—	—	—	—	—	—	
0x3A-0x3F	**Reserved**	—	—	—	—	—	Reserved

8.7 Host Interface Command Descriptions

8.7.1 INIT_PLL (0x01 + 3 parameters)

This command initializes the PLL. PLL Configuration is programmed using the three input parameters. When PLL Lock is asserted, this command sequence is completed.

Parameter 1							
Value							
15	14	13	12	11	10	9	8
Value							
7	6	5	4	3	2	1	0

bits 15-0 Value
These bits must be set to 0003h.

Parameter 2							
Value							
15	14	13	12	11	10	9	8
Value							
7	6	5	4	3	2	1	0

bits 15-0 Value
These bits must be set to 5949h.

Parameter 3							
Value							
15	14	13	12	11	10	9	8
Value							
7	6	5	4	3	2	1	0

bits 15-0 Value
These bits must be set to 0040h.

8.7.2 RUN_SYS (0x02 + 0 parameters)

This command places the S4E5B001B00A00 into normal operation mode by removing it from Standby or Sleep mode.

8.7.3 STBY (0x04 + 0 parameters)

This command places the S4E5B001B00A00 into Standby mode. To wake up from Standby mode, use the RUN_SYS command.

8.7.4 SLP (0x05 + 0 parameters)

This command places the S4E5B001B00A00 into Sleep mode. Sleep mode differs from Standby mode in that the PLL is powered down. To wake up from Sleep mode, use the RUN_SYS command.

8.7.5 INIT_SYS_RUN (0x06 + 0 parameters)

This command initializes the S4E5B001B00A00 and must be run at power-on/reset.

8.7.6 INIT_ROTMODE (0x0B + 1 parameter)

This command initializes the S4E5B001B00A00 Rotation Mode

Parameter 1							
n/a						Area Coordinate Rotation Select bits 1-0	
15	14	13	12	11	10	9	8
n/a							
7	6	5	4	3	2	1	0

bits 9-8 Area Coordinate Rotation Select bits [1:0]
These bits select the rotation mode used to define the area update input coordinates

Table 8-2: Area Coordinate Rotation Selection

bits 9-8	Area Coordinate Rotation
00b	0°
01b	90°
10b	180°
11b	270°

8.7.7 RD_REG (0x10 + 2 parameters)

This command initiates a low level register read from the address specified by parameter 1. The returned data value is available in parameter 2.

Parameter 1							
Register Address bits 15-8							
15	14	13	12	11	10	9	8
Register Address bits 7-0							
7	6	5	4	3	2	1	0

Parameter 2							
Register Read Data bits 15-8							
15	14	13	12	11	10	9	8
Register Read Data bits 7-0							
7	6	5	4	3	2	1	0

Note
Parameter 2 may be repeated multiple times for burst operation, i.e. memory burst read operation. Refer to Firmware programming guides for memory read burst flow.

8.7.8 WR_REG (0x11 + 2 parameters)

This command initiates a low level register write of the data specified by parameter 2 to the address specified by parameter 1..

Parameter 1							
Register Address bits 15-8							
15	14	13	12	11	10	9	8
Register Address bits 7-0							
7	6	5	4	3	2	1	0

Parameter 2							
Register Write Data bits 15-8							
15	14	13	12	11	10	9	8
Register Write Data bits 7-0							
7	6	5	4	3	2	1	0

Note
Parameter 2 may be repeated multiple times for burst operation, i.e. memory burst write operation. Refer to Firmware programming guides for memory write burst flow.

8.7.9 PIP_DISABLE (0x14 + 0 parameters)

This command disables the PIP window.

8.7.10 PIP_ENABLE (0x15 + 4 parameters)

This command enables the PIP window.

Parameter 1							
PIP Transparency Value bits 3-0				n/a			PIP Transparency Enable
15	14	13	12	11	10	9	8
PIP Pixel Output Value Control bits 1-0		PIP Bpp Control bits 1-0		n/a			
7	6	5	4	3	2	1	0

bits 15-12 PIP Transparency Value bits [3:0]
These bits are used to specify a 4bpp transparency value for the PIP to determine the pixels that are transparent in the PIP when PIP Transparency is enabled.
If the PIP's pixel value equals the PIP Transparency Value, the corresponding pixel from the Main Image will be used instead.
For 1bpp mode, the PIP Transparency Value is compared against PIP Key Values 0 and 1 in parameter 4.
For 2bpp mode, the PIP Transparency Value is compared against PIP Key Values 0, 1, 2 and 3 in parameter 4.
For 4bpp mode, the PIP Transparency Value is compared against the PIP value in memory.

bit 8 PIP Transparency Enable
This bit enables transparency function.
When this bit = 0b, PIP Transparency is disabled
When this bit = 1b, PIP Transparency is enabled

bits 7-6 PIP Pixel Output Value Control bits [1:0]
These bits control the bit-wise logic operation performed on non-transparent PIP Pixels to produce resultant pixel output value.

Table 8-3: PIP Pixel Output Value Selection

bits 7-6	PIP Pixel Output Value
00b	No Change to PIP Pixel Value
01b	(Image Buffer Data) XOR (PIP Pixel Value)
10b	(Image Buffer Data) XOR ~(PIP Pixel Value)
11b	~(PIP Pixel Value)

bits 5-4 PIP Bpp Control bits [1:0]

These bits configure the bit-per-pixel format of the image data used for the PIP.

Table 8-4: PIP Bit-Per-Pixel Format Selection

Parameter 1 bits 5-4	PIP Bit-Per-Pixel Format
00b	the PIP is in 1 BPP mode
01b	the PIP is in 2 BPP mode
10b	the PIP is in 4 BPP mode
11b	Reserved

Parameter 2

n/a			PIP Width bits 12-8				
15	14	13	12	11	10	9	8
PIP Width bits 7-0							
7	6	5	4	3	2	1	0

bits 12-0 PIP Width bits [12:0]

These bits specify the width of the PIP, in pixels.

Note

The PIP width must be less than or equal to the width of the display size.

Parameter 3

n/a			PIP Height bits 12-8				
15	14	13	12	11	10	9	8
PIP Height bits 7-0							
7	6	5	4	3	2	1	0

bits 12-0 PIP Height bits [12:0]

These bits specify the height of the PIP, in pixels.

Note

The PIP height must be less than or equal to the height of the display size.

Parameter 4							
PIP Key Value 3 bits 3-0				PIP Key Value 2 bits 3-0			
15	14	13	12	11	10	9	8
PIP Key Value 1 bits 3-0				PIP Key Value 0 bits 3-0			
7	6	5	4	3	2	1	0

Bits 15-12 PIP Key Value 3 bits [3:0]
These bits maps a value of 11b in the PIP buffer to a user programmable 4bpp value that will be output to the display engine. These bits are used for 2bpp only.

Bits 11-8 PIP Key Value 2 bits [3:0]
These bits maps a value of 10b in the PIP buffer to a user programmable 4bpp value that will be output to the display engine. These bits are used for 2bpp only.

Bits 7-4 PIP Key Value 1 bits [3:0]
These bits maps a value of 01b or 1b in the PIP buffer to a user programmable 4bpp value that will be output to the display engine. These bits are used for 1 and 2bpp only.

Bits 11-8 PIP Key Value 0 bits [3:0]
These bits maps a value of 00b or 0b in the PIP buffer to a user programmable 4bpp value that will be output to the display engine. These bits are used for 1 and 2bpp only.

8.7.11 PIP_ADRCFG (0x16 + 2 parameters)

This command setup the PIP Start Address

Parameter 1							
PIP Buffer Start Address bits 15-8							
15	14	13	12	11	10	9	8
PIP Buffer Start Address bits 7-0 (bits 2-0 are always 000b)							
7	6	5	4	3	2	1	0

Parameter 2							
n/a						PIP Buffer Start Address bits 25-24	
15	14	13	12	11	10	9	8
PIP Buffer Start Address bits 23-16							
7	6	5	4	3	2	1	0

Parameter 2 bits 9-0
Parameter 1 bits 15-0 PIP Buffer Start Address bits [25:0]
These bits specify the PIP Buffer start address in internal memory address space. The PIP Buffer start address must be 64-bit aligned (writing to bits 2-0 has no effect).

8.7.12 PIP_XYSETUP (0x17 + 2 parameters)

This command setup the PIP to be displayed.

This command has the following limitations.

Rotation	Bpp	Limit
0 Degree	1	X start position must be a multiple of 64
	2	X start position must be a multiple of 32
	4	X start position must be a multiple of 16
90 Degree	1	Y start position must be a multiple of 64
	2	Y start position must be a multiple of 32
	4	Y start position must be a multiple of 16
180 Degree	1	(Panel Width - PIP Width - X Start): must be a multiple of 64
	2	(Panel Width - PIP Width - X Start): must be a multiple of 32
	4	(Panel Width - PIP Width - X Start): must be a multiple of 16
270 Degree	1	(Panel Width - PIP Height - Y Start): must be a multiple of 64
	2	(Panel Width - PIP Height - Y Start): must be a multiple of 32
	4	(Panel Width - PIP Height - Y Start): must be a multiple of 16

Parameter 1

n/a			PIP X-Start bits 12-8				
15	14	13	12	11	10	9	8
PIP X-Start bits 7-0							
7	6	5	4	3	2	1	0

bits 12-0 PIP X-Start bits [12:0]
These bits specify the X (or Y) start position of the PIP relative to the user reference point, in pixels. See Chapter 9.7, “PIP Relative to Panel Rotation” on page 81.

Note
The PIP cannot be placed entirely off screen. At least 1 pixel must remain on the visible display.

Parameter 2

n/a			PIP Y-Start bits 12-8				
15	14	13	12	11	10	9	8
PIP Y-Start bits 7-0							
7	6	5	4	3	2	1	0

bits 12-0 PIP Y-Start bits [12:0]
These bits specify the Y position start of the PIP window.

Note
The PIP cannot be placed entirely off screen. At least 1 pixel must remain on the visible display.

8.7.13 CSR_MAINCFG (0x18 + 4 parameter)

Initialize Cursor window setup.

Parameter 1							
Cursor Transparency Value bits 3-0				n/a			Cursor Transparency Enable
15	14	13	12	11	10	9	8
Cursor Pixel Output Value Control bits 1-0		Cursor Bpp Control bits 1-0		Cursor Pixel Write Swap Sel bits 1-0		n/a	Cursor Enable
7	6	5	4	3	2	1	0

bits 15-12 Cursor Transparency Value bits [3:0]
These bits are used to specify a 4bpp transparency value for the cursor to determine the pixels that are transparent in the cursor when cursor Transparency is enabled, Parameter 1 bit 8 = 1b.
If cursor's pixel value equals the cursor Transparency Value, the corresponding pixel from the Main or PIP Image will be used instead.
For 1bpp mode, the cursor Transparency Value is compared against cursor Key Values 0 and 1 in parameter 4.
For 2bpp mode, the cursor Transparency Value is compared against cursor Key Values 0, 1, 2 and 3 in parameter 4.
For 4bpp mode, the cursor Transparency Value is compared against the cursor value in memory.

bit 8 Cursor Transparency Enable
These bits enables the cursor Transparency function.
When this bit = 0b, cursor Transparency is disabled.
When this bit = 1b, cursor Transparency is enabled.

bits 7-6 Cursor Pixel Output Value Control bits [1:0]
These bits control the bit-wise logic operation performed on non-transparent Cursor Pixels and PIP/Image buffer resultant pixel data to produce resultant pixel output value.

Table 8-5: Cursor Pixel Output Value Selection

bits 7-6	Cursor Pixel Output Value
00b	No Change to Cursor Pixel Value
01b	(Image Buffer Data/PIP Pixel Value) XOR (Cursor Pixel Value)
10b	(Image Buffer Data/PIP Pixel Value) XOR ~(Cursor Pixel Value)
11b	~(Cursor Pixel Value)

bits 5-4 Cursor Bpp Control bits [1:0]
These bits configure the bit-per-pixel format of the image data used for the Cursor.

Table 8-6: Cursor Bit-Per-Pixel Format Selection

bits 5-4	Cursor Bit-Per-Pixel Format
00b	the Cursor is in 1 BPP mode
01b	the Cursor is in 2 BPP mode
10b	the Cursor is in 4 BPP mode
11b	Reserved

bits 3-2 Host Cursor Access Pixel Swap Selection bits [1:0]
These bits configure way the pixels are swapped when data in the cursor memory is accessed by the host.
Bit 0 - Pixel Order Reverse.
Bit 1 - Byte Order Reverse.
Refer to the table below for more information.

Table 8-7: Packed Pixel Mode Selection

	Host Pixel Data Expected Format			
Cursor BPP Mode	**Swap Select = 00**	**Swap Select = 01**	**Swap Select = 10**	**Swap Select = 11**
1 bpp	{P15,P14,P13,P12,P11,P10,P9,P8 P7,P6,P5,P4,P3,P2,P1,P0}	{P0,P1,P2,P3,P4,P5,P6,P7, P8,P9,P10,P11,P12,P13,P14,P15}	{P7,P6,P5,P4,P3,P2,P1,P0, P15,P14,P13,P12,P11,P10,P9,P8}	{P8,P9,P10,P11,P12,P13,P14,P15, P0,P1,P2,P3,P4,P5,P6,P7}
2 bpp	{P7,P6,P5,P4, P3,P2,P1,P0}	{P0,P1,P2,P3, P4,P5,P6,P7}	{P3,P2,P1,P0, P7,P6,P5,P4}	{P4,P5,P6,P7, P0,P1,P2,P3}
4 bpp	{P3,P2, P1,P0}	{P0,P1, P2,P3}	{P1,P0, P3,P2}	{P2,P3, P0,P1}

bit 0 Cursor Enable
This bit enables the Cursor function.

Parameter 2

n/a			Reserved	Cursor Width bits 11-8			
15	14	13	12	11	10	9	8
Cursor Width bits 7-0							
7	6	5	4	3	2	1	0

bit 12 Reserved
This bit must be set to 0b.

bits 11-0 Cursor Width bits [11:0]
These bits specify the width of the cursor, in pixels.

Note

1. For 90° and 270° rotation the Cursor Width becomes the Cursor Height.
2. The width for the cursor must NOT be programmed to 0 when the cursor is enabled
3. For cursor widths that are not a multiple of (16 bits/cursor bpp), the width of the image in memory must be padded up to the next multiple of (16 bits/cursor bpp).

Parameter 3							
n/a			Reserved	Cursor Height bits 11-8			
15	14	13	12	11	10	9	8
Cursor Height bits 7-0							
7	6	5	4	3	2	1	0

bit 12 Reserved
This bit must be set to 0b.

bits 11-0 Cursor Height bits [11:0]
These bits specify the height of the cursor, in pixels.

Note

1. The maximum value for Cursor Height is 2048 (800h).
2. The height for the cursor must NOT be programmed to 0 when the cursor is enabled
3. For cursor heights that are not a multiple of (16 bits/cursor bpp), the height of the image in memory must be padded up to the next multiple of (16 bits/cursor bpp).

Parameter 4							
Cursor Key Value 3 bits 3-0				Cursor Key Value 2 bits 3-0			
15	14	13	12	11	10	9	8
Cursor Key Value 1 bits 3-0				Cursor Key Value 0 bits 3-0			
7	6	5	4	3	2	1	0

Bits 15-12 Cursor Key Value 3 bits [3:0]
These bits maps a value of 11b in the Cursor buffer to a user programmable 4bpp value that will be output to the display engine. These bits are used for 2bpp only.

Bits 11-8 Cursor Key Value 2 bits [3:0]
These bits maps a value of 10b in the Cursor buffer to a user programmable 4bpp value that will be output to the display engine. These bits are used for 2bpp only.

Bits 7-4 Cursor Key Value 1 bits [3:0]
These bits maps a value of 01b or 1b in the Cursor buffer to a user programmable 4bpp value that will be output to the display engine. These bits are used for 1 and 2bpp only.

Bits 3-0 Cursor Key Value 0 bits [3:0]
These bits maps a value of 00b or 0b in the Cursor buffer to a user programmable 4bpp value that will be output to the display engine. These bits are used for 1 and 2bpp only.

8.7.14 CSR_XYSETUP (0x19 + 2 parameters)

This command setup the Cursor window position to be displayed.

Parameter 1							
n/a			Cursor X-Start bits 12-8				
15	14	13	12	11	10	9	8
Cursor X-Start bits 7-0							
7	6	5	4	3	2	1	0

bits 12-0 Cursor X-Start bits [12:0]£

These bits specify the X start position of the cursor relative to the user reference point, in pixels. Please see chapter 9.8, “Cursor Relative to Panel Rotation” on page 84.

Note

The cursor cannot be placed entirely off screen. At least 1 pixel must remain on the visible display.

Parameter 2							
n/a			Cursor Y-Start bits 12-8				
15	14	13	12	11	10	9	8
Cursor Y-Start bits 7-0							
7	6	5	4	3	2	1	0

bits 12-0 Cursor Y-Start bits [12:0]

These bits specify the Y start position of the cursor relative to the user reference point, in pixels. Please see chapter 9.8, “Cursor Relative to Panel Rotation” on page 84.

Note

The cursor cannot be placed entirely off screen. At least 1 pixel must remain on the visible display.

8.7.15 CSR_ADRCFG (0x1A + 1 parameters)

This command sets up the cursor address.

Parameter 1							
n/a					Cursor Memory Pointer bits 10-8		
15	14	13	12	11	10	9	8
Cursor Memory Pointer bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0 Cursor Memory Pointer bits [10:0]
This register configures the index for the Cursor Memory Access Port, as well as the address used by the display engine when accessing the cursor image.

This pointer address is automatically incremented when the Cursor Memory Access Port is written to or read from. For read accesses, one dummy read must be performed from the Cursor Memory Access Port before actual data can be retrieved from the cursor memory. During read accesses, the current word being read from the Cursor Memory Access Port is from the address pointed to by this parameter - 1.

Before the display engine can access the cursor memory, this parameter must be programmed to the correct address in memory where the cursor image starts.

8.7.16 BST_RD_SDR (0x1C + 4 parameters)

This command starts a memory burst read operation.

Parameter 1							
Host Memory Access Address bits 15-8							
15	14	13	12	11	10	9	8
Host Memory Access Address bits 7-0							
7	6	5	4	3	2	1	0

Parameter 2							
Reserved						Host Memory Access Address bits 25-24	
15	14	13	12	11	10	9	8
Host Memory Access Address bits 23-16							
7	6	5	4	3	2	1	0

Parameter 2 bits 9-0
Parameter 1 bits 15-0 Host Memory Access Address bits [25:0]
These bits specify the 26-bit memory address used for raw memory accesses by the Host. For raw memory reads and raw memory writes, the start address must be aligned on a 16-bit boundary (bit 0 = 0b).

Note

1. The Host Raw Memory Access Address must be set within the available memory range and the minimum read count is 2 words.
2. Memory read is not supported for Intel 80 8-bit Host Interface.

Parameter 3							
Host Raw Memory Access Count bits 15-8							
15	14	13	12	11	10	9	8
Host Raw Memory Access Count bits 7-0							
7	6	5	4	3	2	1	0

Parameter 4							
Reserved						Host Raw Memory Access Count bits 25-24	
15	14	13	12	11	10	9	8
Host Raw Memory Access Count bits 23-16							
7	6	5	4	3	2	1	0

Parameter 4 bits 9-0

Parameter 3 bits 15-0 Host Raw Memory Access Count bits [25:0]

These bits are only used for raw memory accesses. These bits are programmed before the start of a host memory operation to specify the number of 16-bit accesses that will be performed for raw memory accesses by the host memory interface during burst operations. The minimum number of raw memory accesses that can be performed is 1.

During a host memory access, these bits can be used with the following formulas to determine the number of 16-bit accesses remaining for the current memory access.
For write access, these bits = number of words left to be written.
For read access, these bits = number of words left to be read -1.

For raw memory access reads, the access count must be set to the required number of reads + 1. Once the read operation is complete, the host memory interface should be reset.

Note

1. These bits must be set such that the Host Memory Access operation is within the configured memory range.
2. These bits must not be set to 0000_0000h for a Host memory access.

8.7.17 BST_WR_SDR (0x1D + 4 parameters)

This command starts a memory burst write operation.

Parameter 1							
Host Memory Access Address bits 15-8							
15	14	13	12	11	10	9	8
Host Memory Access Address bits 7-0							
7	6	5	4	3	2	1	0

Parameter 2							
Reserved						Host Raw Memory Access Address bits 25-24	
15	14	13	12	11	10	9	8
Host Memory Access Address bits 23-16							
7	6	5	4	3	2	1	0

Parameter 2 bits 9-0
Parameter 1 bits 15-0 Host Memory Access Address bits [25:0]
These bits specify the 26-bit memory address used for raw memory accesses by the Host. For raw memory reads and raw memory writes, the start address must be aligned on a 16-bit boundary (bit 0 = 0b).

Note
The Host Raw Memory Access Address must be set within the available memory range and the minimum read count is 2 words.

Parameter 3							
Host Raw Memory Access Count bits 15-8							
15	14	13	12	11	10	9	8
Host Raw Memory Access Count bits 7-0							
7	6	5	4	3	2	1	0

Parameter 4							
Reserved						Host Raw Memory Access Count bits 25-24	
15	14	13	12	11	10	9	8
Host Raw Memory Access Count bits 23-16							
7	6	5	4	3	2	1	0

Parameter 4 bits 9-0

Parameter 3 bits 15-0 Host Raw Memory Access Count bits [25:0]

These bits are only used for raw memory accesses. These bits are programmed before the start of a host memory operation to specify the number of 16-bit accesses that will be performed for raw memory accesses by the host memory interface during burst operations. The minimum number of raw memory accesses that can be performed is 1.

During a host memory access, these bits can be used with the following formulas to determine the number of 16-bit accesses remaining for the current memory access.
For write access, these bits = number of words left to be written.
For read access, these bits = number of words left to be read -1.

For raw memory access reads, the access count must be set to the required number of reads + 1. Once the read operation is complete, the host memory interface should be reset.

Note

1. These bits must be set such that the Host Memory Access operation is within the configured memory range.
2. These bits must not be set to 0000_0000h for a Host memory access.

8.7.18 BST_END_SDR (0x1E + 0 parameters)

This command terminates a burst operation before it has completed.

Note

Any memory operation in progress should be terminated with this command before a new memory operation is initiated.

8.7.19 LD_IMG (0x20 + 1 parameter)

This command starts a Full Frame Memory Load operation according to the data packing settings in parameter 1.

Parameter 1							
n/a					Host Packed Write Display Location Select	n/a	
15	14	13	12	11	10	9	8
n/a	Host Packed Write Bit Expansion Disable	Host Packed Pixel Select bits 1-0		Host Packed Write Transparency Enable	n/a		
7	6	5	4	3	2	1	0

bit 10 Host Packed Memory Write Location Select
This bit selects the location for host packed memory writes.
When this bit = 0b, host packed memory writes are written to the Main Image Buffer.
When this bit = 1b, host packed memory writes are written to the PIP Buffer.

bit 6 Host Packed Write Bit Expansion Disable
This bit disables the expansion of packed pixel write data. Refer to bits 5-4 description for further info.
When this bit = 0b, expansion of packed pixel write data is enabled.
When this bit = 1b, expansion of packed pixel write data is disabled.

Note
This bit must be set to 0b for the Main Image Buffer (bit 10 = 0b).

bits 5-4 Host Packed Pixel Select bits [1:0]
These bits only have an effect when Packed Pixel Access is selected. These bits select the packed pixel mode used for host memory interface accesses.

Table 8-8: Packed Pixel Mode Selection

bits 5-4	**bit 6 = 0b- Bit Expanding Enable** (Main Image and PIP Buffers)		**bit 6 = 1b - Bit Expanding Disabled** (PIP Buffer Only)	
	Input Packed Pixel Mode	**Storage in Memory**	**Input Packed Pixel Mode**	**Storage in Memory**
00b	1 bpp	4 bpp	1 bpp	1 bpp
01b	2 bpp	4 bpp	2 bpp	2 bpp
10b	4 bpp	4 bpp	4 bpp	4 bpp
11b	1 Byte-per-pixel	4 bpp	1 Byte-per-pixel	4 bpp

bit 3 Host Packed Write Transparency Enable
Enable Transparency for host packed pixel write.
When this bit = 0b, Transparency is disabled.
When this bit = 1b, Transparency is enabled.

8.7.20 LD_IMG_AREA (0x22 + 5 parameters)

This command starts an Area Defined Frame Memory Load operation according to the data packing settings in the parameters.

Parameter 1							
n/a					Host Packed Write Display Location Select	n/a	
15	14	13	12	11	10	9	8
n/a	Host Packed Write Bit Expansion Disable	Host Packed Pixel Select bits 1-0		Host Packed Write Transparency Enable	n/a		
7	6	5	4	3	2	1	0

bit 10 Host Packed Memory Write Location Select
This bit selects the location for host packed memory writes.
When this bit = 0b, host packed memory writes are written to the Image Buffer.
When this bit = 1b, host packed memory writes are written to the PIP Buffer.

bit 6 Host Packed Write Bit Expansion Disable
This bit disables the expansion of packed pixel write data. Refer to bits 5-4 description for further info.
When this bit = 0b, expansion of packed pixel write data is enabled.
When this bit = 1b, expansion of packed pixel write data is disabled.

Note
This bit must be set to 0b for the Main Image Buffer (bit 10 = 0b).

bits 5-4 Host Packed Pixel Select bits [1:0]
These bits only have an effect when Packed Pixel Access is selected. These bits select the packed pixel mode used for host memory interface accesses.

Table 8-9: Packed Pixel Mode Selection

bits 5-4	**bit 6 = 0b- Bit Expanding Enable** (Main Image and PIP Buffers)		**bit 6 = 1b - Bit Expanding Disabled** (PIP Buffer Only)	
	Input Packed Pixel Mode	**Storage in Memory**	**Input Packed Pixel Mode**	**Storage in Memory**
00b	1 bpp	4 bpp	1 bpp	1 bpp
01b	2 bpp	4 bpp	2 bpp	2 bpp
10b	4 bpp	4 bpp	4 bpp	4 bpp
11b	1 Byte-per-pixel	4 bpp	1 Byte-per-pixel	4 bpp

bit 3 Host Packed Write Transparency Enable
Enable Transparency for host packed pixel write.
When this bit = 0b, Transparency is disabled.
When this bit = 1b, Transparency is enabled.

Parameter 2							
n/a			Reserved	Packed Pixel Rectangular X-Start Position bits 11-8			
15	14	13	12	11	10	9	8
Packed Pixel Rectangular X-Start Position bits 7-0							
7	6	5	4	3	2	1	0

bits 11-0 Packed Pixel Rectangular X-Start Position bits [11:0]

These bits are only used for packed pixel accesses. These bits specify the X-Start position of the image write, in pixels, relative to the top left corner of the display area. For further information, see 9.1, “Introduction” on page 68.

Note

For 0° and 180° rotation (, the X-Start position must be set smaller than the line data length.

For 90° and 270° rotation, the X-Start position must be set smaller than the frame data length.

Note

When 0° or 180° write rotation is selected and the packed pixel mode is:

2 bpp, the X-Start Position must be divisible by 8.

3 / 4 bpp, the X-Start Position must be divisible by 4.

1 Byte-per-pixel, the X-Start Position must be divisible by 2.

Parameter 3							
n/a			Reserved	Packed Pixel Rectangular Y-Start Position bits 11-8			
15	14	13	12	11	10	9	8
Packed Pixel Rectangular Y-Start Position bits 7-0							
7	6	5	4	3	2	1	0

bits 11-0 Packed Pixel Rectangular Y-Start Position bits [11:0]

These bits are only used for packed pixel accesses. These bits specify the Y-Start position of the image write, in pixels, relative to the top left corner of the display area. For further information, see 9.1, “Introduction” on page 68.

Note

For 0° and 180° rotation, the Y-Start position must be set smaller than the frame data length.

For 90° and 270° rotation, the Y-Start position must be set smaller than the line data length.

Parameter 4							
n/a			Packed Pixel Rectangular Width bits 12-8				
15	14	13	12	11	10	9	8
Packed Pixel Rectangular Width bits 7-0							
7	6	5	4	3	2	1	0

bits 12-0 Packed Pixel Rectangular Width bits [12:0]
These bits are only used for packed pixel accesses. These bits specify the width of the image write, in pixels. For further information, see 9.1, “Introduction” on page 68.

Parameter 5							
n/a			Packed Pixel Rectangular Height bits 12-8				
15	14	13	12	11	10	9	8
Packed Pixel Rectangular Height bits 7-0							
7	6	5	4	3	2	1	0

bits 12-0 Packed Pixel Rectangular Height bits [12:0]
These bits are only used for packed pixel accesses. These bits specify the height of the image write, in pixels. For further information, see 9.1, “Introduction” on page 68.

8.7.21 LD_IMG_END (0x23 + 0 parameters)

This command terminates a Load Image operation before it has completed and waits for the image load to finish before writing to the Memory.

Note
Any memory operation in progress should be terminated with this command before a new memory operation is initiated.

8.7.22 LD_IMG_SETADR (0x25 + 2 parameters)

This command sets the packed write memory location to the parameter value.

Parameter 1							
Host Memory Access Address bits 15-8							
15	14	13	12	11	10	9	8
Host Memory Access Address bits 7-0							
7	6	5	4	3	2	1	0

Parameter 2							
Reserved						Host Memory Access Address bits 25-24	
15	14	13	12	11	10	9	8
Host Memory Access Address bits 23-16							
7	6	5	4	3	2	1	0

Parameter 2 bits 9-0

Parameter 1 bits 15-0 Host Memory Access Address bits [25:0]
These bits specify the 26-bit memory address used for manually configuring the packed pixel write location. The start address must be aligned on a 64-bit boundary (bits 2-0 = 000b).

8.7.23 LD_IMG_DSPEADR (0x26 + 0 parameters)

This command sets the packed write memory location to the address specified by the Display Engine image buffer.

8.7.24 WAIT_DSPE_TRG (0x28 + 0 parameters)

This command waits for the Display Update operation to complete. For examples of how the WAIT_DSPE_TRG command is used, see Chapter 12, “Firmware Programming Guide” on page 105.

8.7.25 WAIT_DSPE_FREND (0x29 + 0 parameters)

This command waits for the Display Update to complete outputting display frames. For examples of how the WAIT_DSPE_FREND command is used, see Chapter 12, “Firmware Programming Guide” on page 105.

8.7.26 WAIT_DSPE_LUTFREE (0x2A + 0 parameters)

This command waits for the Display Pipeline to have room for another update.

8.7.27 RD_WFM_INFO (0x30 + 2 parameters)

This command prepares the Waveform for use.

Parameter 1							
Waveform Header Address bits 15-8							
15	14	13	12	11	10	9	8
Waveform Header Address bits 7-0							
7	6	5	4	3	2	1	0

Parameter 2							
n/a						Waveform Header Address bits 25-24	
15	14	13	12	11	10	9	8
Waveform Header Address bits 23-16							
7	6	5	4	3	2	1	0

Parameter 2 bits 9-0

Parameter 1 bits 15-0 Waveform Header Address bits [25:0]

These bits store the Waveform Header start address in Memory. These parameters should be programmed as follows.

Parameter 2 = 0000h

Parameter 1 = 108Ah

8.7.28 UPD_INIT (0x32 + 0 parameters)

This command issues a Panel Update Buffer Refresh with data from the Image Buffer. No display operation will occur.

8.7.29 UPD_FULL (0x33 + 1 parameter)

This command issues a Full Frame Full Update operation.

Parameter 1							
n/a	Reserved	n/a		Display Update Waveform Mode bits 3-0			
15	14	13	12	11	10	9	8
Display Update Pipeline Select bits 3-0				n/a			
7	6	5	4	3	2	1	0

bit 14 Reserved
The default value for this bit is 0b.

bits 11-8 Display Update Waveform Mode Select bits [3:0]
These bits are only used for Operation Modes 3 and 4. These bits select the Waveform Mode for the display update.

bits 7-4 Display Update Pipeline Select bits [3:0]
These bits select the Pipeline (from Pipeline 0 to Pipeline 14) that is used for the display update. If the Pipeline Auto Select Enable bit is set, these bits are ignored and the Pipeline is automatically selected.

Note
These bits must not be programmed with a value of 1111b.

8.7.30 UPD_FULL_AREA (0x34 + 5 parameters)

This command issues an Area Defined Full Update operation.

Parameter 1							
n/a	Reserved	n/a		Display Update Waveform Mode bits 3-0			
15	14	13	12	11	10	9	8
Display Update Pipeline Select bits 3-0				n/a			
7	6	5	4	3	2	1	0

bit 14 Reserved
The default value for this bit is 0b.

bits 11-8 Display Update Waveform Mode Select bits [3:0]
These bits are only used for Operation Modes 3 and 4. These bits select the Waveform Mode for the display update.

bits 7-4 Display Update Pipeline Select bits [3:0]
These bits select the Pipeline (from Pipeline 0 to Pipeline 14) that is used for the display update. If the Pipeline Auto Select Enable bit is set, these bits are ignored and the Pipeline is automatically selected.

Note
These bits must not be programmed with a value of 1111b.

Parameter 2

n/a			Reserved	Area Update Pixel Rectangular X-Start Position bits 11-8			
15	14	13	12	11	10	9	8
Area Update Pixel Rectangular X-Start Position bits 7-0							
7	6	5	4	3	2	1	0

bit 12 Reserved
The default value for this bit is 0b.

bits 11-0 Area Update Pixel Rectangular X-Start Position bits [11:0]
When the Update Rectangle Mode bits are set for user configured X/Y Start/End, these bits specify the X start position of the rectangular area to be updated, relative to the top left of the rotated image.

Parameter 3

n/a			Reserved	Area Update Pixel Rectangular Y-Start Position bits 11-8			
15	14	13	12	11	10	9	8
Area Update Pixel Rectangular Y-Start Position bits 7-0							
7	6	5	4	3	2	1	0

bit 12 Reserved
The default value for this bit is 0b.

bits 11-0 Area Update Pixel Rectangular Y-Start Position bits [11:0]
When the Update Rectangle Mode bits are set for user configured X/Y Start/End, these bits specify the Y start position of the rectangular area to be updated, relative to the top left of the rotated image.

Parameter 4

n/a			Area Update Pixel Rectangular X-End Position / Horizontal Size bits 12-8				
15	14	13	12	11	10	9	8
Area Update Pixel Rectangular X-End Position / Horizontal Size bits 7-0							
7	6	5	4	3	2	1	0

bits 12-0 Area Update Pixel Rectangular X-End Position / Horizontal Size bits [12:0]
When the Update Rectangle Mode bits are set for user configured X/Y Start/End, these bits specify either the X end position or the horizontal size of the rectangular area to be updated, relative to the top left of the rotated image.

Note
Update widths and heights must be greater than 0h.

Parameter 5							
n/a			Area Update Pixel Rectangular Y-End Position / Vertical Size bits 12-8				
15	14	13	12	11	10	9	8
Area Update Pixel Rectangular Y-End Position / Vertical Size bits 7-0							
7	6	5	4	3	2	1	0

bits 12-0 Area Update Pixel Rectangular Y-End Position / Vertical Size bits [12:0]
When the Update Rectangle Mode bits are set for user configured X/Y Start/End, these bits specify either the Y end position or the vertical size of the rectangular area to be updated, relative to the top left of the rotated image.

Note
Update widths and heights must be greater than 0h.

8.7.31 UPD_PART (0x35 + 1 parameter)

This command issues a Partial Screen Update operation. This operation affects changed pixels only.

Parameter 1							
n/a	Reserved	n/a		Display Update Waveform Mode bits 3-0			
15	14	13	12	11	10	9	8
Display Update Pipeline Select bits 3-0				n/a			
7	6	5	4	3	2	1	0

bit 14 Reserved
The default value for this bit is 0b.

bits 11-8 Display Update Waveform Mode Select bits [3:0]
These bits are only used for Operation Modes 3 and 4. These bits select the Waveform Mode for the display update.

bits 7-4 Display Update Pipeline Select bits [3:0]
These bits select the Pipeline (from Pipeline 0 to Pipeline 14) that is used for the display update. If the Pipeline Auto Select Enable bit is set, these bits are ignored and the Pipeline is automatically selected.

Note
These bits must not be programmed with a value of 1111b.

8.7.32 UPD_PART_AREA (0x36 + 5 parameters)

This command issues an area defined Partial Screen Update operation. This operation affects changed pixels only.

Parameter 1							
n/a	Reserved	n/a		Display Update Waveform Mode bits 3-0			
15	14	13	12	11	10	9	8
Display Update Pipeline Select bits 3-0				n/a			
7	6	5	4	3	2	1	0

bit 14 Reserved
The default value for this bit is 0b.

bits 11-8 Display Update Waveform Mode Select bits [3:0]
These bits are only used for Operation Modes 3 and 4. These bits select the Waveform Mode for the display update.

bits 7-4 Display Update Pipeline Select bits [3:0]
These bits select the Pipeline (from Pipeline 0 to Pipeline 14) that is used for the display update. If the Pipeline Auto Select Enable bit is set, these bits are ignored and the Pipeline is automatically selected.

Note
These bits must not be programmed with a value of 1111b.

Parameter 2							
n/a			Reserved	Area Update Pixel Rectangular X-Start Position bits 11-8			
15	14	13	12	11	10	9	8
Area Update Pixel Rectangular X-Start Position bits 7-0							
7	6	5	4	3	2	1	0

bit 12 Reserved
The default value for this bit is 0b.

bits 11-0 Area Update Pixel Rectangular X-Start Position bits [11:0]
When the Update Rectangle Mode bits are set for user configured X/Y Start/End, these bits specify the X start position of the rectangular area to be updated, relative to the top left of the rotated image.

Parameter 3

n/a			Reserved	Area Update Pixel Rectangular Y-Start Position bits 11-8			
15	14	13	12	11	10	9	8
Area Update Pixel Rectangular Y-Start Position bits 7-0							
7	6	5	4	3	2	1	0

bit 12 Reserved
The default value for this bit is 0b.

bits 11-0 Area Update Pixel Rectangular Y-Start Position bits [11:0]
When the Update Rectangle Mode bits are set for user configured X/Y Start/End, these bits specify the Y start position of the rectangular area to be updated, relative to the top left of the rotated image.

Parameter 4

n/a			Area Update Pixel Rectangular X-End Position / Horizontal Size bits 12-8				
15	14	13	12	11	10	9	8
Area Update Pixel Rectangular X-End Position / Horizontal Size bits 7-0							
7	6	5	4	3	2	1	0

bits 12-0 Area Update Pixel Rectangular X-End Position / Horizontal Size bits [12:0]
When the Update Rectangle Mode bits are set for user configured X/Y Start/End, these bits specify either the X end position or the horizontal size of the rectangular area to be updated, relative to the top left of the rotated image.

Note
Update widths and heights must be greater than 0h.

Parameter 5

n/a			Area Update Pixel Rectangular Y-End Position / Vertical Size bits 12-8				
15	14	13	12	11	10	9	8
Area Update Pixel Rectangular Y-End Position / Vertical Size bits 7-0							
7	6	5	4	3	2	1	0

bits 12-0 Area Update Pixel Rectangular Y-End Position / Vertical Size bits [12:0]
When the Update Rectangle Mode bits are set for user configured X/Y Start/End, these bits specify either the Y end position or the vertical size of the rectangular area to be updated, relative to the top left of the rotated image.

Note
Update widths and heights must be greater than 0h.

8.7.33 UPD_SET_IMGADR (0x38 + 2 parameters)

This command sets the Display Engine Image Buffer Start Address.

Parameter 1							
Image Buffer Start Address bits 15-8							
15	14	13	12	11	10	9	8
Image Buffer Start Address bits 7-0 (bits 2-0 are always 000b)							
7	6	5	4	3	2	1	0

Parameter 2							
n/a						Image Buffer Start Address bits 25-24	
15	14	13	12	11	10	9	8
Image Buffer Start Address bits 23-16							
7	6	5	4	3	2	1	0

Chapter 9 Display Memory Configurations

9.1 Introduction

The S4E5B001B00A00 controller manages its memory in 5 different areas.

- **Panel Update Buffer Area**
  - Panel Update Buffer is used for internal update operations only. It should be not accessed by the Host.
  - Panel Update Buffer data requires 1.5 bytes per pixel.
  - Contains the current display pixel value which should match the state of the panel.
  - Memory location starts at 0.
  - Memory size requirement for the Panel Update Buffer area is:
    Display Height x Roundup16(Display Width x 12 ÷ 8)
- **Main Image Buffer Area**
  - Host can consider the Main Image Buffer as the display buffer area.
  - Main Image Buffer data is always stored in packed 4-bit per pixel format.
  - Memory location is configurable, but should start immediately after the Panel Update Buffer Area (see Chapter 8.7.33, “UPD_SET_IMGADR (0x38 + 2 parameters)” on page 67). Do not overlap the buffer areas.
  - Memory size requirement for the Main Image Buffer area is:
    Roundup16(Display Width) x Display Height x 4 bits ÷ 8
- **Picture-in-Picture Image Buffer Area**
  - PIP image buffer is used as an overlay with transparency value select on top of the image buffer.
  - PIP can be configured to be 1bpp, 2bpp or 4bpp mode, packed.
  - Memory location is configurable (see Chapter 8.7.11, “PIP_ADRCFG (0x16 + 2 parameters)” on page 45).
  - Transparency setup is configurable (see Chapter 8.7.10, “PIP_ENABLE (0x15 + 4 parameters)” on page 43).
  - Memory size requirement for the PIP Image Buffer area is:
    PIP Height x Roundup16(PIP Width x bpp ÷ 8)
- **Cursor Buffer Storage Area**
  - Cursor Buffer Area is used as an overlay with transparency value selection on top of the Main and PIP image.
  - Cursor Buffer is stored in a dedicated 4K bytes of internal RAM.
  - Cursor can be configured to be 1bpp, 2bpp or 4bpp mode, packed.
  - Cursor location is configurable (see Chapter 8.7.15, “CSR_ADRCFG (0x1A + 1 parameters)” on page 51).
  - Transparency setup is configurable (see Chapter 8.7.13, “CSR_MAINCFG (0x18 + 4 parameter)” on page 47).
  - Memory size requirement for each cursor is:
    Cursor Height x [2 x roundup(Cursor Width x bpp ÷ 16])

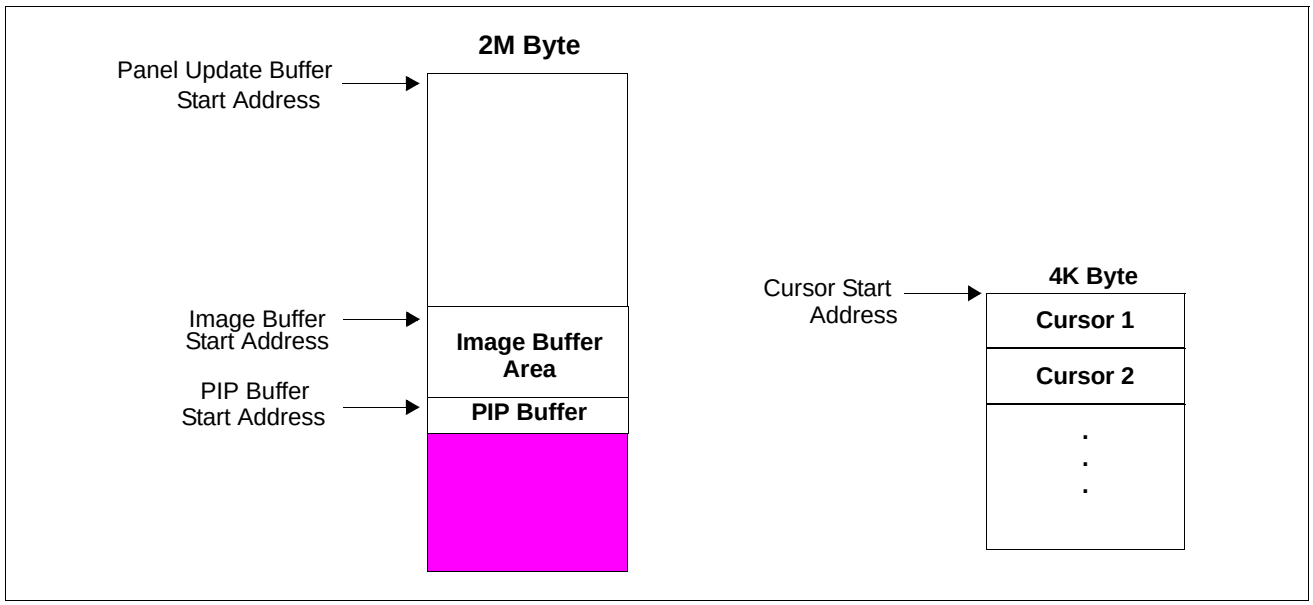


Figure 9-1: S4E5B001B00A00 Memory Area Configuration Example

9.2 Image Buffer Memory Area Setup

The S4E5B001B00A00 controller supports destructive partial region writes to the Image Buffer area. A predefined memory start region can be defined using the Host commands (see 8.6, “Command List” on page 38). The memory start region is referenced as the Image Buffer Start Address position in the memory.

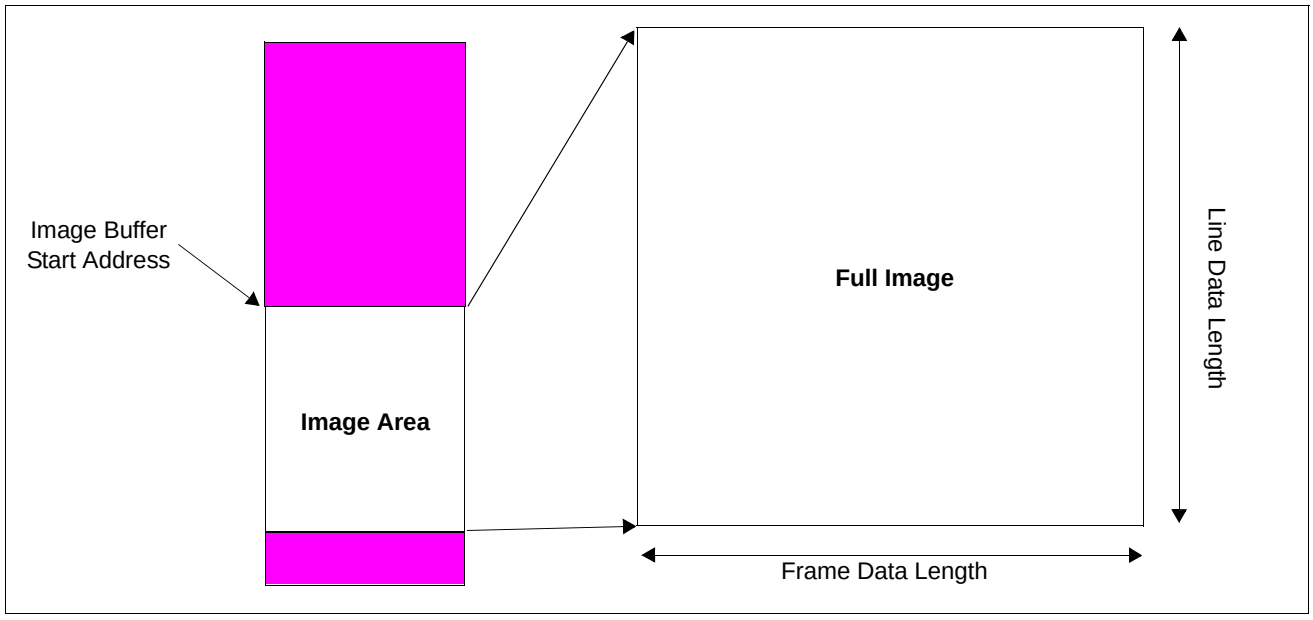


Figure 9-2: Full Image Area Setup

For any partial write using the predefined area (configured using the XStart, YStart, Width, and Height parameters), the S4E5B001B00A00 automatically calculates the start memory address position.

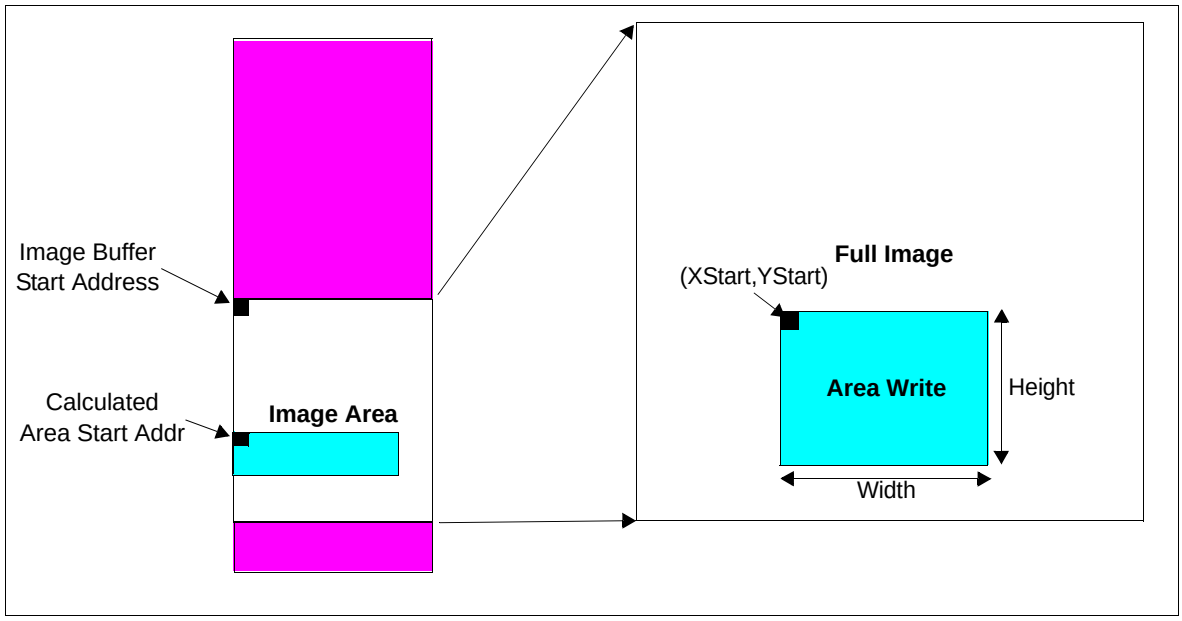


Figure 9-3: Host Memory Area Setup

9.2.1 Image Buffer Storage

Unprocessed pixel data is stored in the Image Buffer in raster sequence. The address pointer is defined by the command UPD_SET_IMGADR.

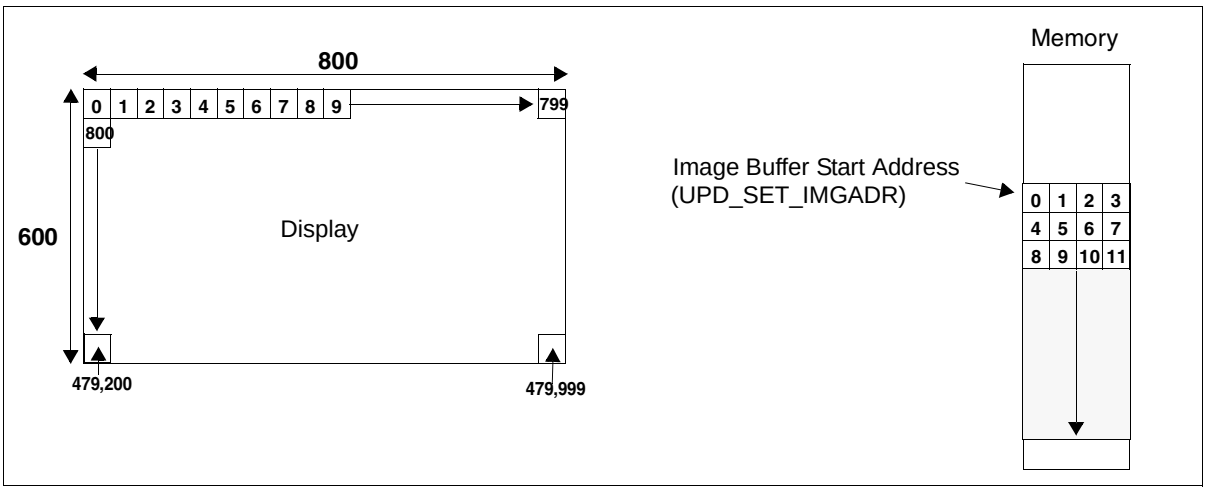


Figure 9-4: Image Buffer Storage Example - 800x600 Display Size

9.3 PIP Image Buffer Memory Area Setup

The S4E5B001B00A00 controller supports 1 PIP Image Buffer overlaid on top of Image Buffer area. Selectable transparency grey-scale value is available for 2bpp and 4bpp mode, while selectable grey-scale value is available for 1bpp mode.

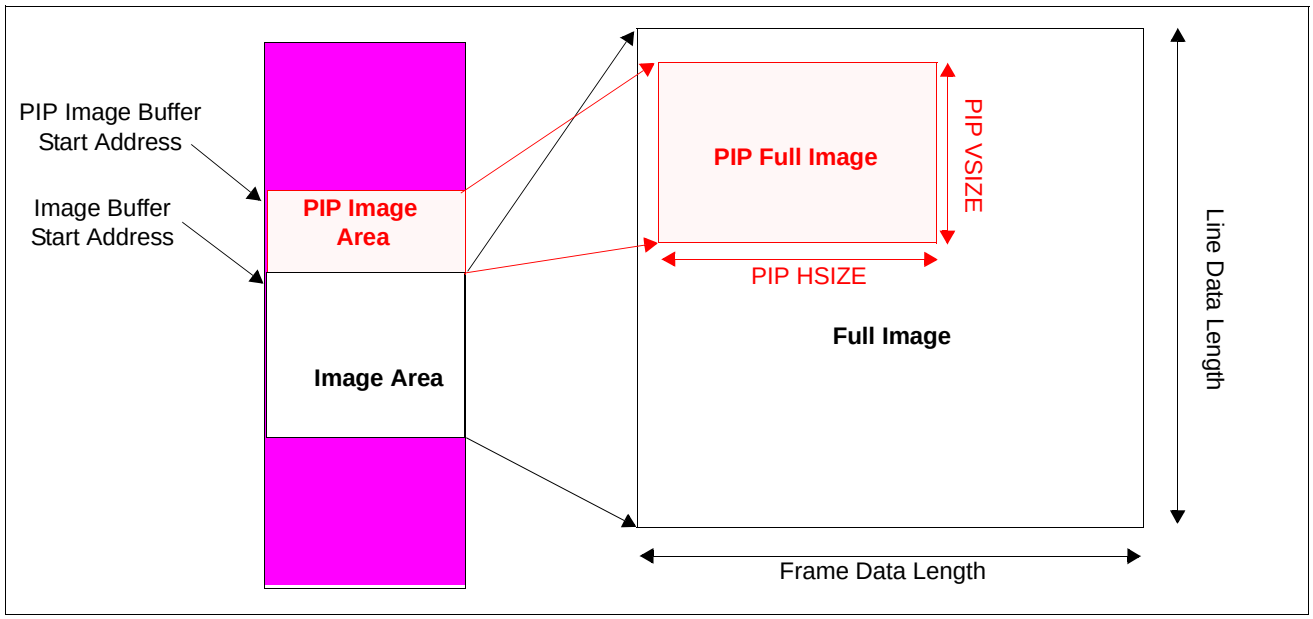


Figure 9-5: PIP Full Image Area Setup

For any partial write using the predefined area (configured using the XStart, YStart, Width, and Height parameters), the S4E5B001B00A00 automatically calculates the start memory address position.

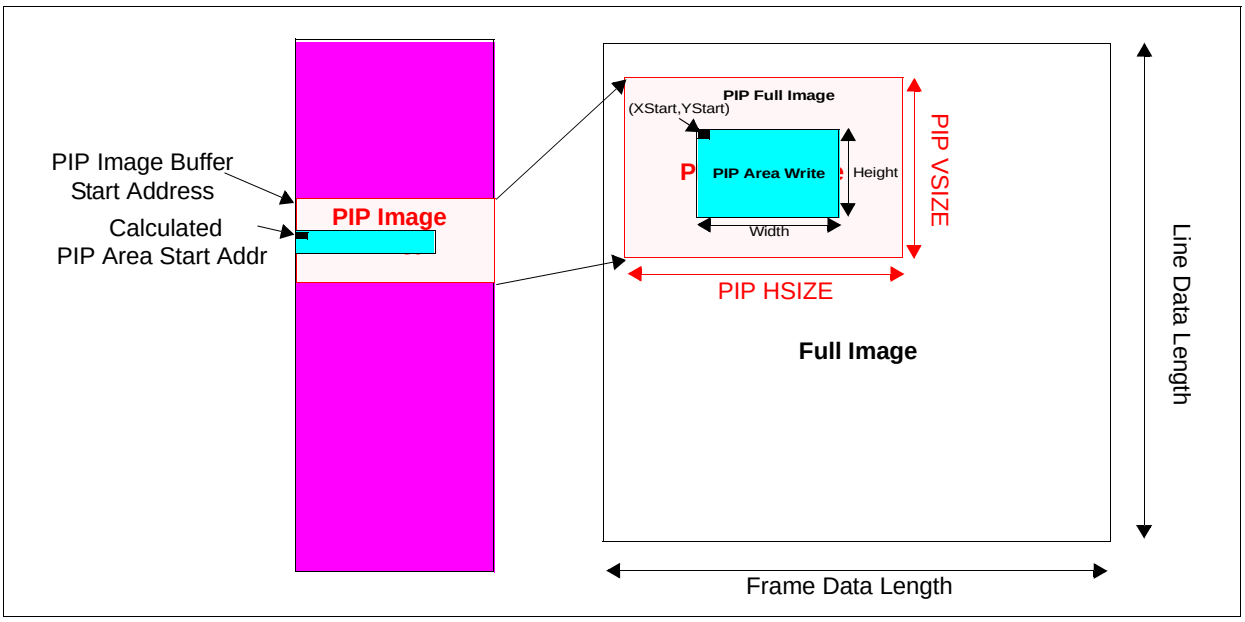


Figure 9-6: Host Memory PIP Area Setup

9.3.1 PIP Image Buffer Storage

Unprocessed pixel data is stored in the PIP Image Buffer in raster sequence. Depending on the PIP Memory Bpp setup, the storage can be 1Bpp, 2Bpp or 4Bpp.

9.4 Panel Update Buffer Memory Area Setup

The S4E5B001B00A00 requires that the Panel Update Buffer memory area is located such that it does not overlap with the Image Buffer memory area. The location for the Panel Update Buffer in memory is address 0x00000000.

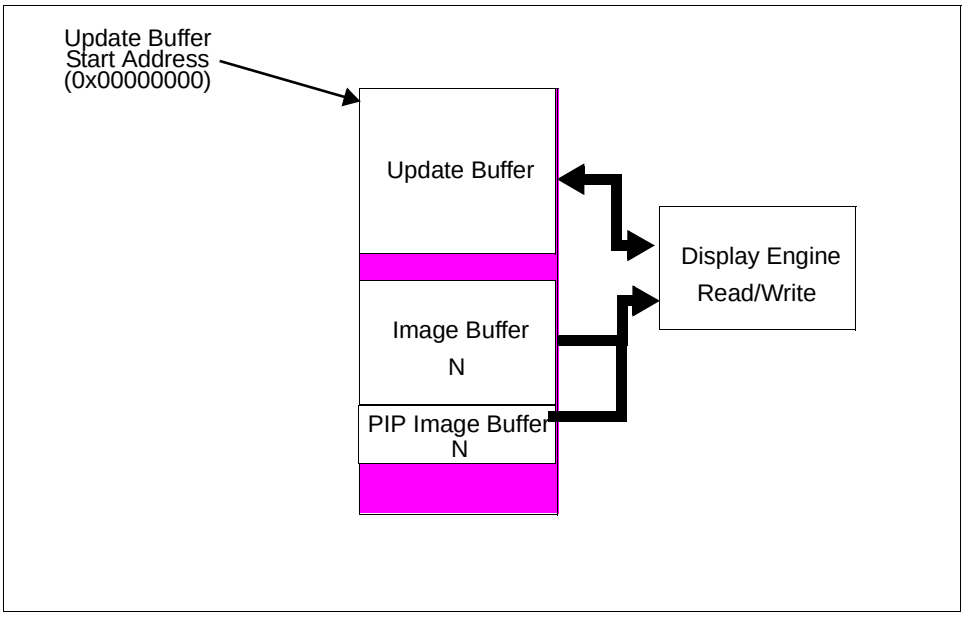


Figure 9-7: Panel Update Buffer Memory Start Address Setup

9.4.1 Panel Update Buffer Usage

The Panel Update Buffer contains the current pixel value/state of the EPD panel. It is recommended to perform initialization of the Panel Update Buffer using the following commands:

- **UPD_INIT -** A software application should copy the last known pixel value of the panel to the Image Buffer and perform an UPD_INIT command which will synchronize the contents of the Panel Update Buffer with the panel's pixel value.
- **UPD_FULL -** With waveform INIT (Waveform number 0), which will initialize the display panel to a known state. This should be used when the previous image cannot be retrieved for restoration.

The following programming flow shows how to initialize the Panel Update Buffer and synchronize it with the display contents.

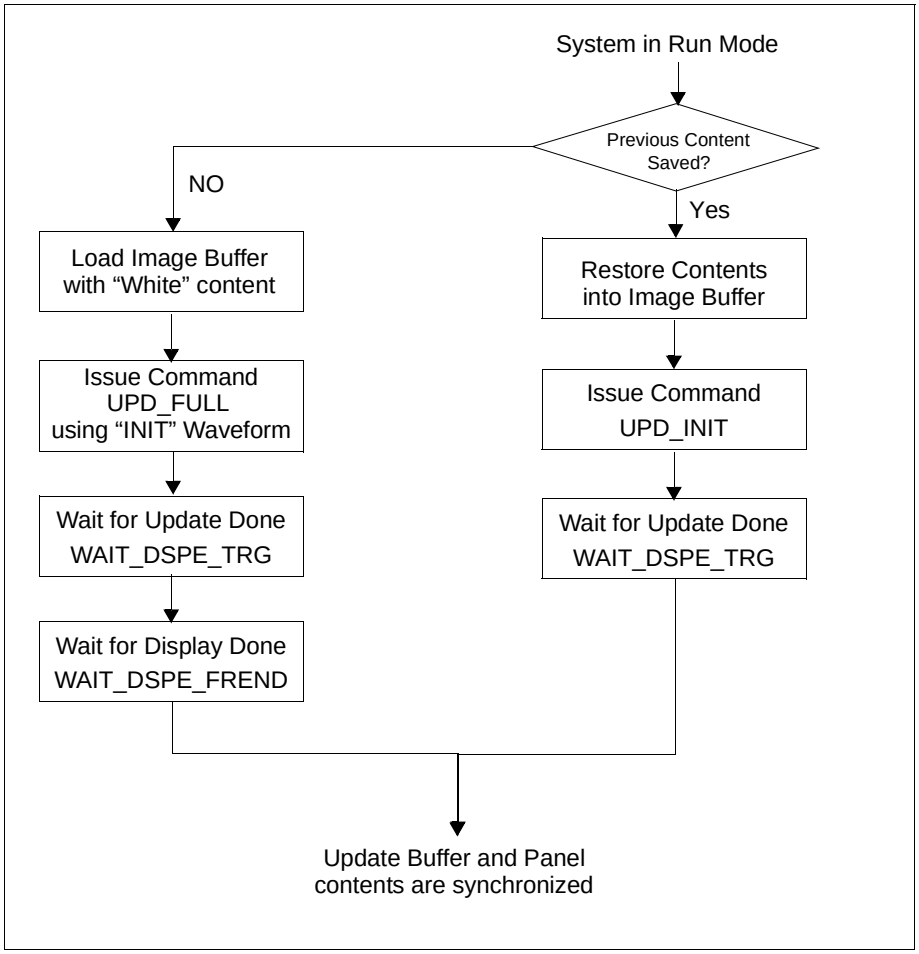


Figure 9-8: Panel Update Buffer Initialization Programming Flow

9.5 Rotation Support

- Rotation applies to the Main and PIP Image Buffer at the same time.
- Cursor Image is not rotated (see Chapter 9.8, “Cursor Relative to Panel Rotation” on page 84).

9.5.1 Rotation Introduction

Display panels are typically oriented in Landscape mode, where the Horizontal Size is larger than the Vertical Size. In this case, the display refresh occurs from Left to right and top to bottom.

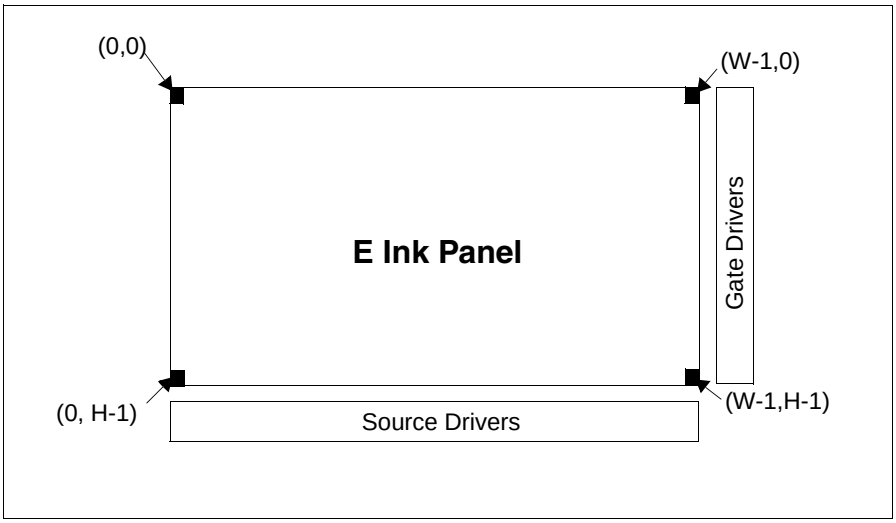


Figure 9-9: Landscape (0 Degrees) Display

The S4E5B001B00A00 supports rotation modes that allow 90°, 180°, and 270° rotation in a counter-clockwise direction. The rotation is done in hardware and is transparent to the user for all display buffer writes.

The actual address translation is performed during the Host Write and the Main and PIP image data are stored in memory in the rotated orientation.

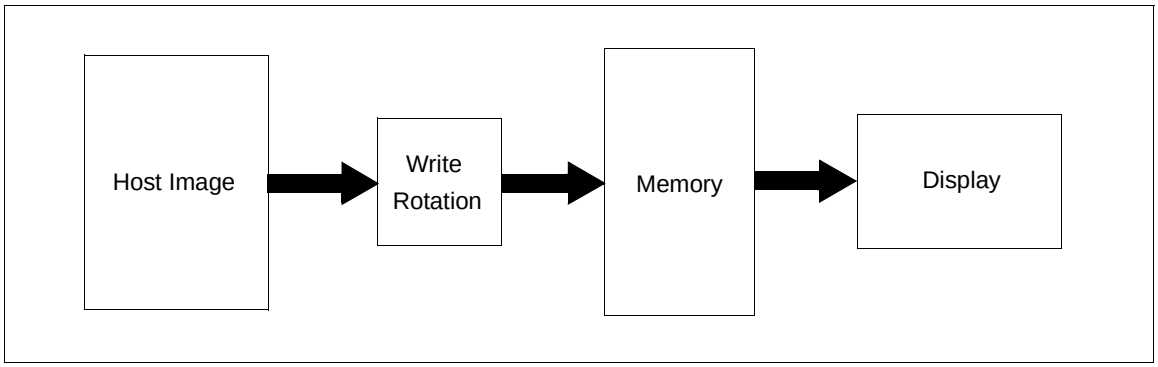


Figure 9-10: Write Rotation Flow in Hardware

Note
Memory Readback data does not support reverse-rotation.

9.5.2 90° Rotation

The following figure shows how the programmer sees a 600x800 portrait image and how the image is being displayed. The application image is written to the S4E5B001B00A00 in the following sense: A–B–C–D. The display is refreshed in the following sense: B-D-A-C.

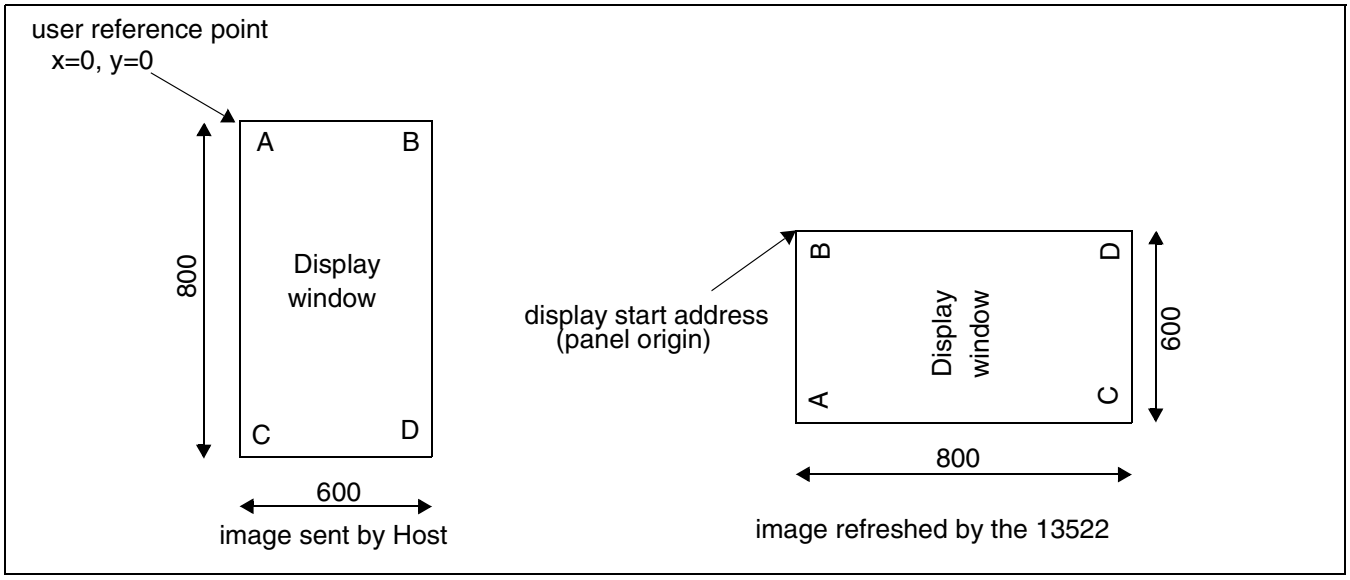


Figure 9-11: Relationship Between the Screen Image and the Image Refreshed in 90° Rotation

Programming

There are no special programming requirements other than enabling the rotation (see Chapter 8.7.6, “INIT_ROTMODE (0x0B + 1 parameter)” on page 41). The start address and line offset are automatically calculated by the hardware.

9.5.3 180° Rotation

The following figure shows how the programmer sees a 800x600 landscape image and how the image is being displayed. The application image is written to the S4E5B001B00A00 in the following sense: A–B–C–D. The display is refreshed in the following sense: D-C-B-A.

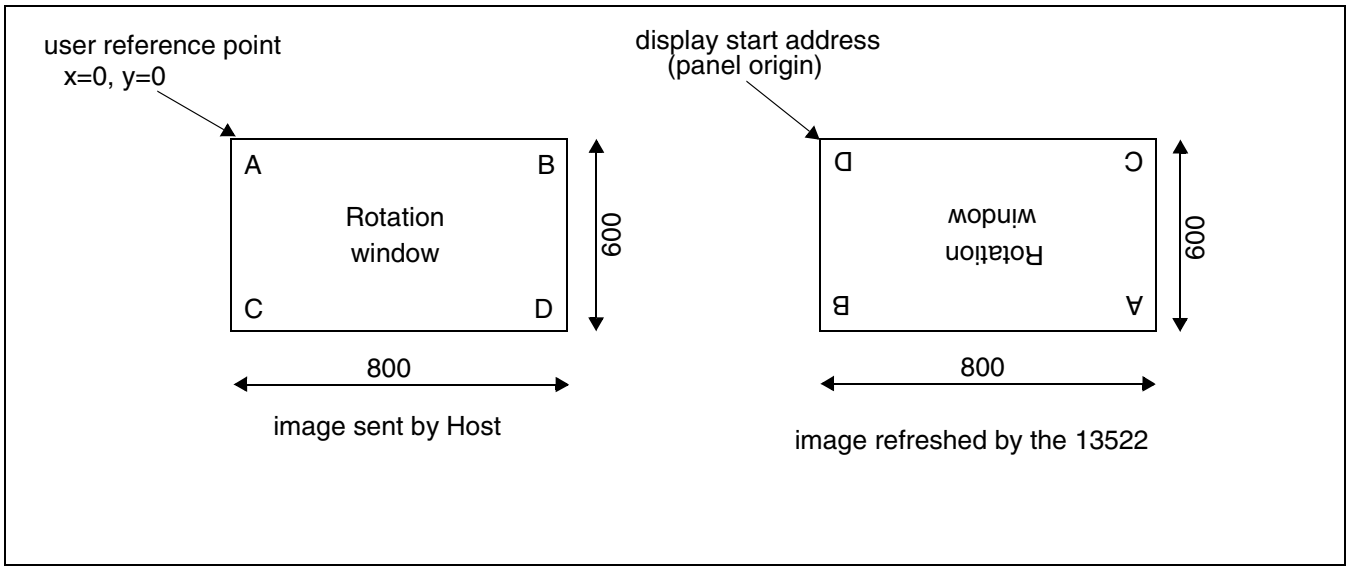


Figure 9-12: Relationship Between the Screen Image and the Image Refreshed in 180° Rotation

Programming

There are no special programming requirements other than enabling the rotation (see Chapter 8.7.6, “INIT_ROTMODE (0x0B + 1 parameter)” on page 41). The start address and line offset are automatically calculated by the hardware.

9.5.4 270° Rotation

The following figure shows how the programmer sees a 600x800 portrait image and how the image is being displayed. The application image is written to the S4E5B001B00A00 in the following sense: A–B–C–D. The display is refreshed in the following sense: C-A-D-B.

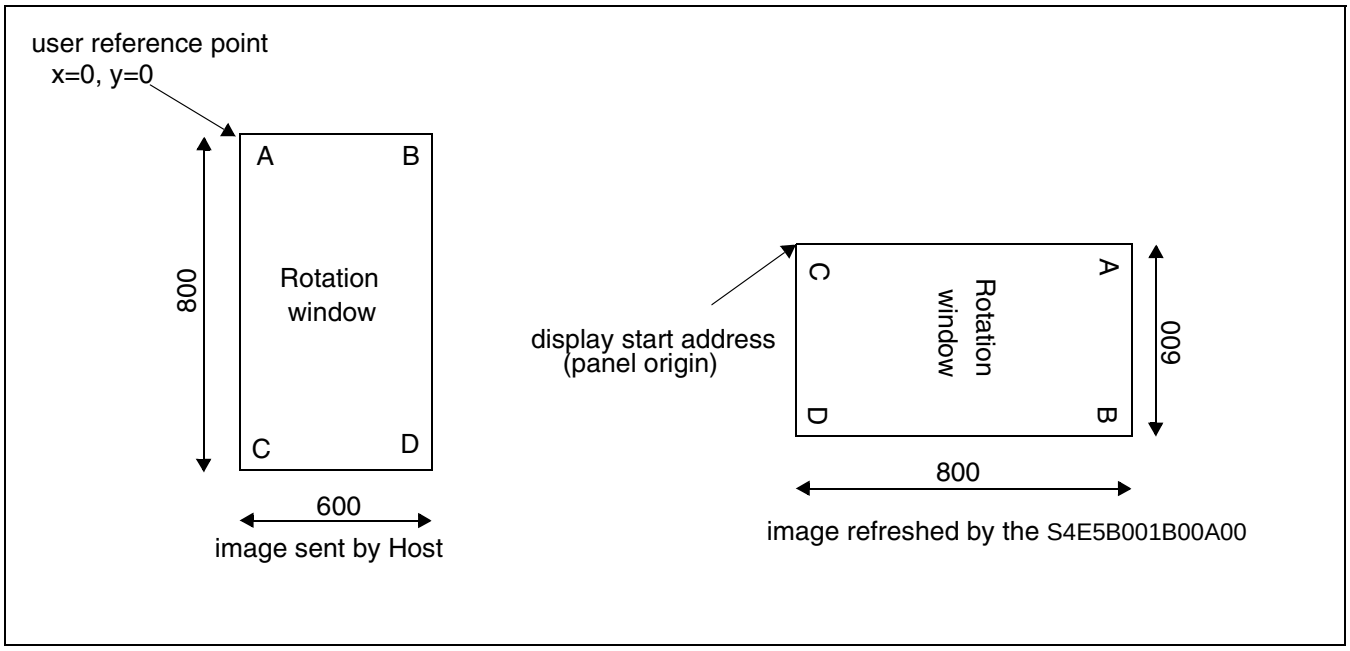


Figure 9-13: Relationship Between the Screen Image and the Image Refreshed in 270° Rotation

Programming

There are no special programming requirements other than enabling the rotation (see Chapter 8.7.6, “INIT_ROTMODE (0x0B + 1 parameter)” on page 41). The start address and line offset are automatically calculated by the hardware.

9.6 Window Area Position / Rotation

For a Windowed Area write operation when rotation is enabled, the X-Start, Y-Start, Width and Height settings must be specified relative to **the user's own reference point**.

9.6.1 90° Rotation

The following figure shows how the Windowed Area is rotated on the displayed image. The application Window Area is written to the S4E5B001B00A00 in the following sense: a–b–c–d.

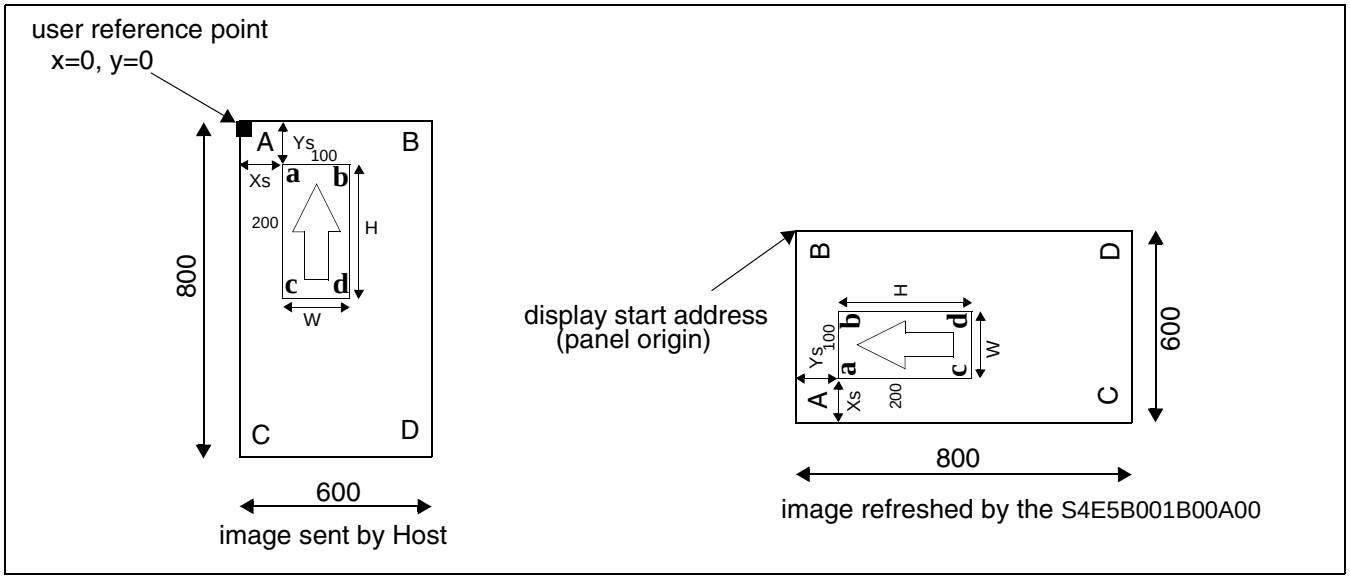


Figure 9-14: Relationship Between the Screen Window Area and the Image Refreshed in 90° Rotation

Programming

There are no special programming requirements other than enabling the rotation (see Chapter 8.7.6, "INIT_ROTMODE (0x0B + 1 parameter)" on page 41). The start address and line offset are automatically calculated by the hardware.

9.6.2 180° Rotation

The following figure shows how the Windowed Area is rotated on the displayed image. The application Window Area is written to the S4E5B001B00A00 in the following sense: a–b–c–d.

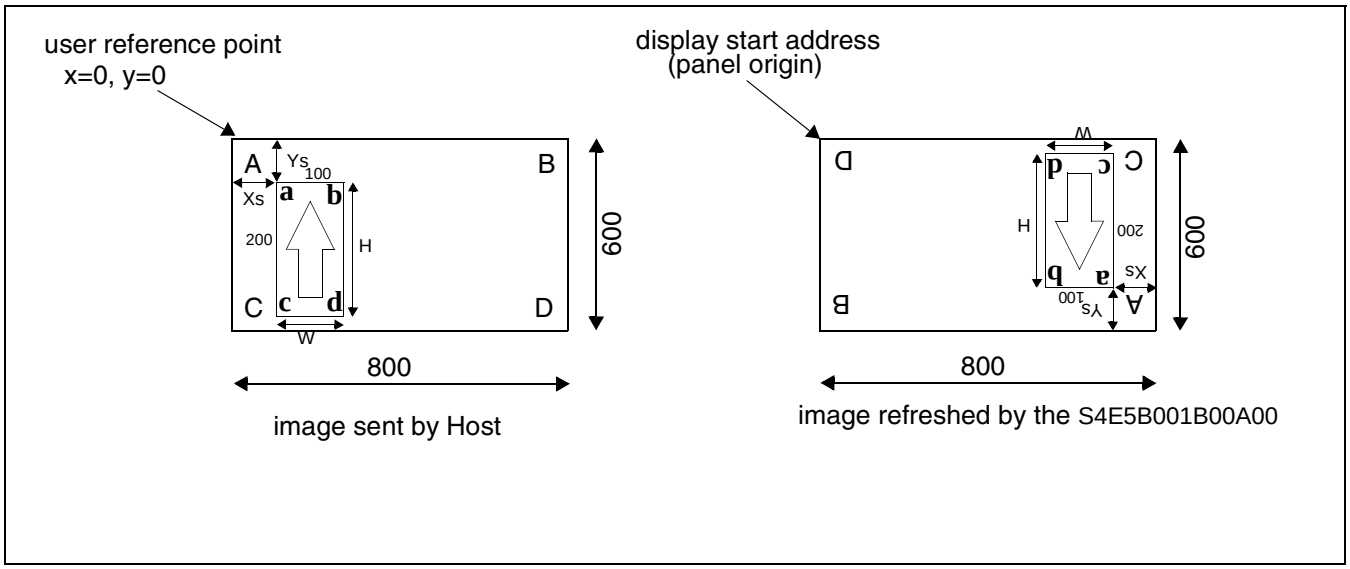


Figure 9-15: Relationship Between the Screen Window Area and the Image Refreshed in 180° Rotation

Programming

There are no special programming requirements other than enabling the rotation (see Chapter 8.7.6, “INIT_ROTMODE (0x0B + 1 parameter)” on page 41). The start address and line offset are automatically calculated by the hardware.

9.6.3 270° Rotation

The following figure shows how the Windowed Area is rotated on the displayed image. The application Window Area is written to the S4E5B001B00A00 in the following sense: a–b–c–d.

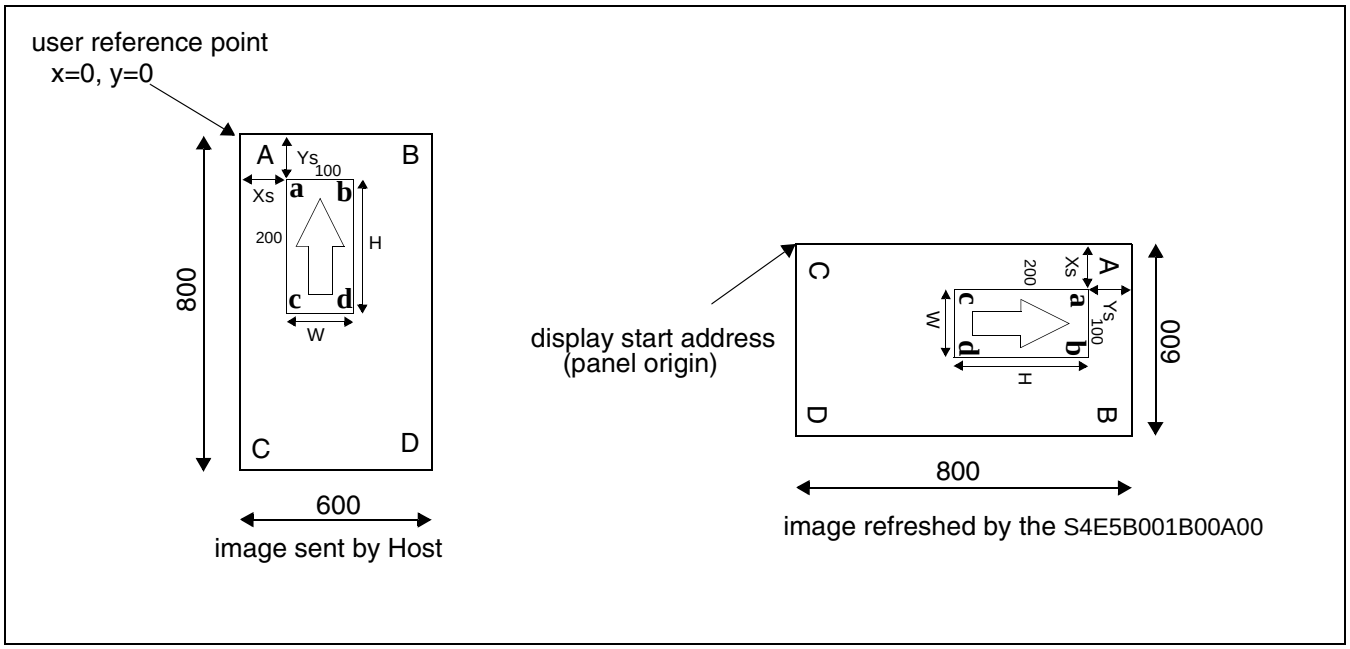


Figure 9-16: Relationship Between the Screen Window Area and the Image Refreshed in 270° Rotation

Programming

There are no special programming requirements other than enabling the rotation (see Chapter 8.7.6, “INIT_ROTMODE (0x0B + 1 parameter)” on page 41). The start address and line offset are automatically calculated by the hardware.

9.7 PIP Relative to Panel Rotation

For a PIP setup, X-Start, Y-Start, Width and Height settings must be specified relative to **the user's own reference point**.

9.7.1 90° Rotation

The following figure shows how the Windowed Area is rotated on the displayed image. The application Window Area is written to the S4E5B001B00A00 in the following sense: a–b–c–d.

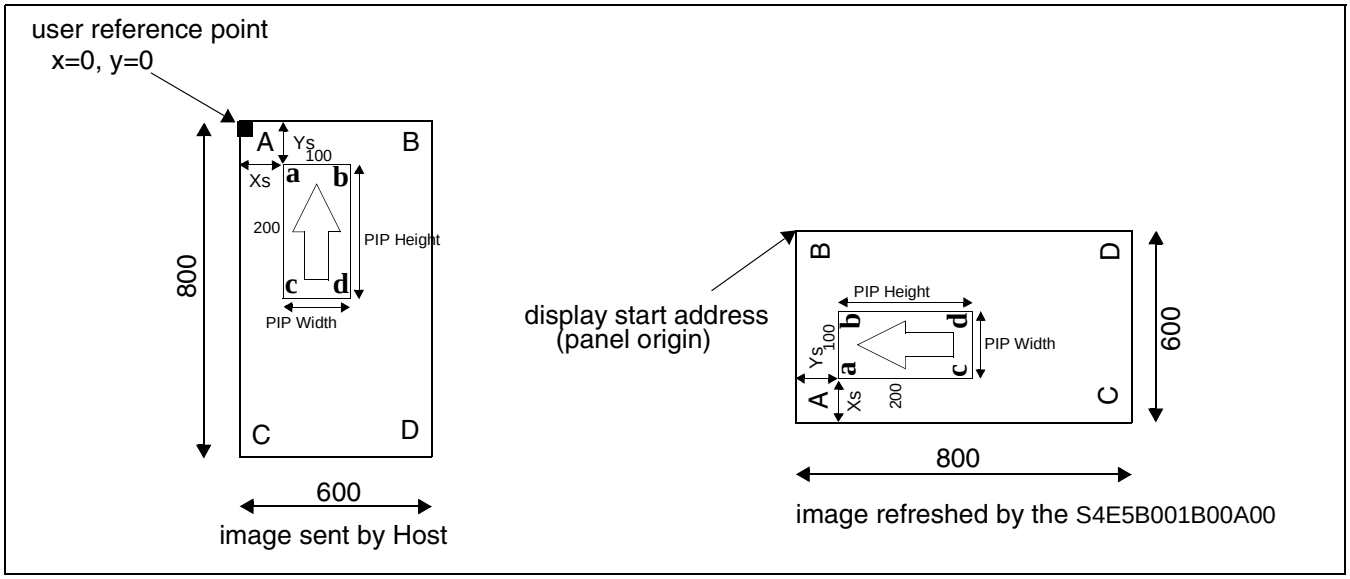


Figure 9-17: Relationship Between the Screen Window Area and the Image Refreshed in 90° Rotation

Programming

User must program the PIP and Cursor Width and Height according to their viewpoint
PIP Data will be rotated accordingly.

9.7.2 180° Rotation

The following figure shows how the Windowed Area is rotated on the displayed image. The application Window Area is written to the S4E5B001B00A00 in the following sense: a–b–c–d.

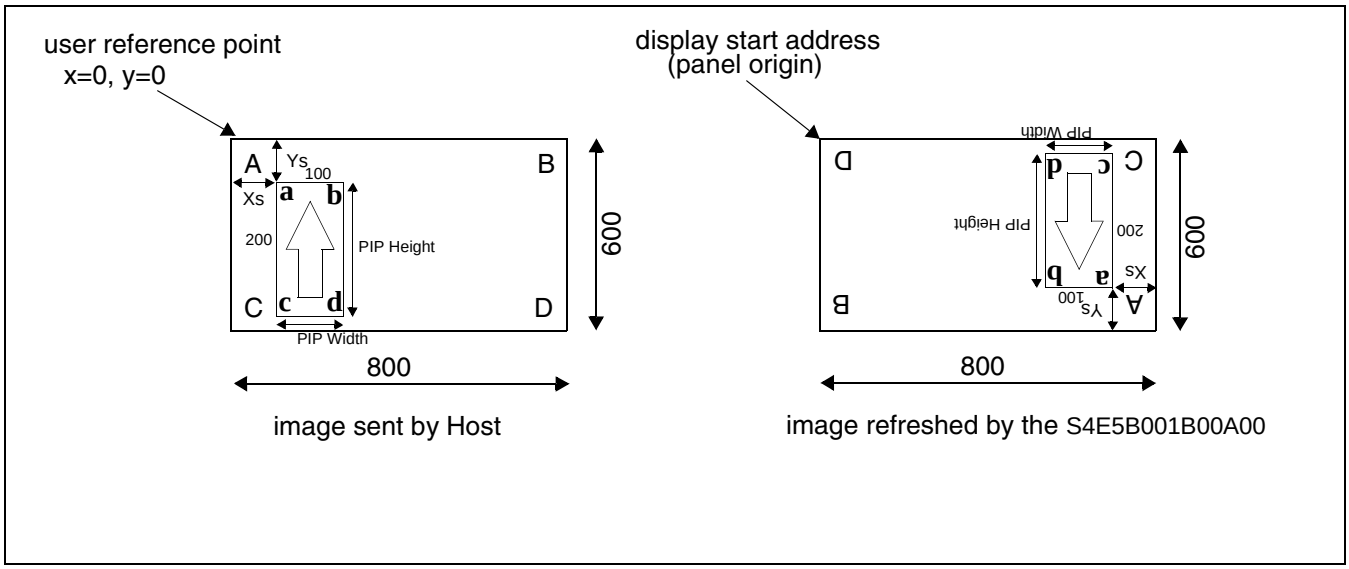


Figure 9-18: Relationship Between the Screen Window Area and the Image Refreshed in 180° Rotation

Programming

User must program the PIP Width and Height according to their viewpoint.
PIP Data will be rotated accordingly.

9.7.3 270° Rotation

The following figure shows how the Windowed Area is rotated on the displayed image. The application Window Area is written to the S4E5B001B00A00 in the following sense: a–b–c–d.

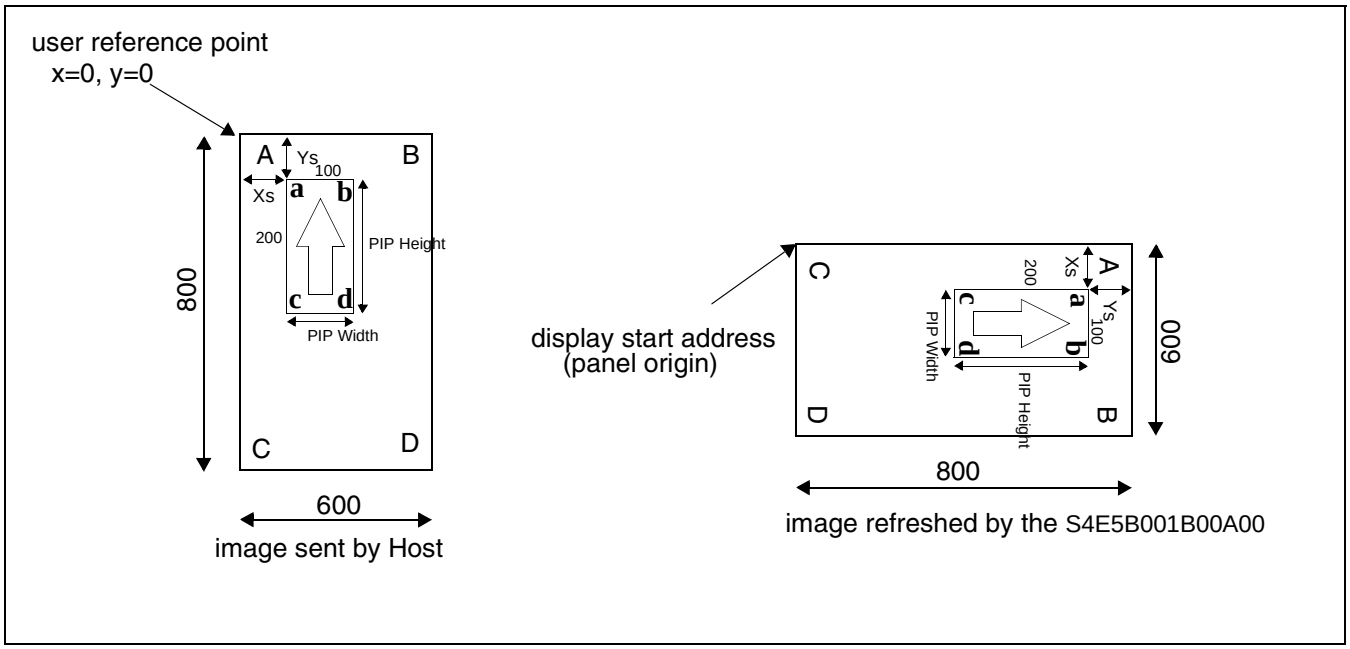


Figure 9-19: Relationship Between the Screen Window Area and the Image Refreshed in 270° Rotation

Programming

User must program the PIP Width and Height according to their viewpoint.
PIP Data will be rotated accordingly.

9.8 Cursor Relative to Panel Rotation

For a Cursor setup, X-Start and Y-Start settings must be specified relative to **the user's own reference point**. The width and height, as well as the image data, remain relative to the panel origin.

9.8.1 90° Rotation

The following figure shows how the Windowed Area is rotated on the display. The cursor position is referenced to A and its corner closest to A.

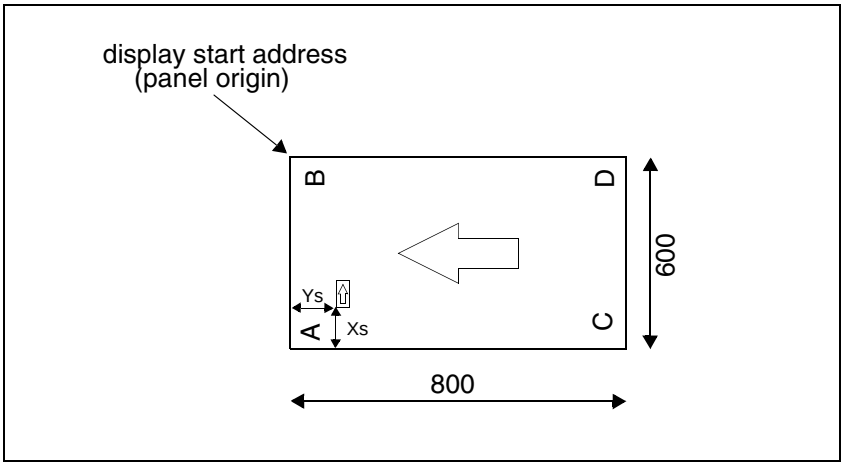


Figure 9-20: Relationship Between Cursor Position and the Image Refreshed in 90° Rotation

Programming

User must program the Cursor Width and Height according to their viewpoint.
Cursor data requires host assisted rotation support.

9.8.2 180° Rotation

The following figure shows how the Windowed Area is rotated on the display. The cursor position is referenced to A and its corner closest to A.

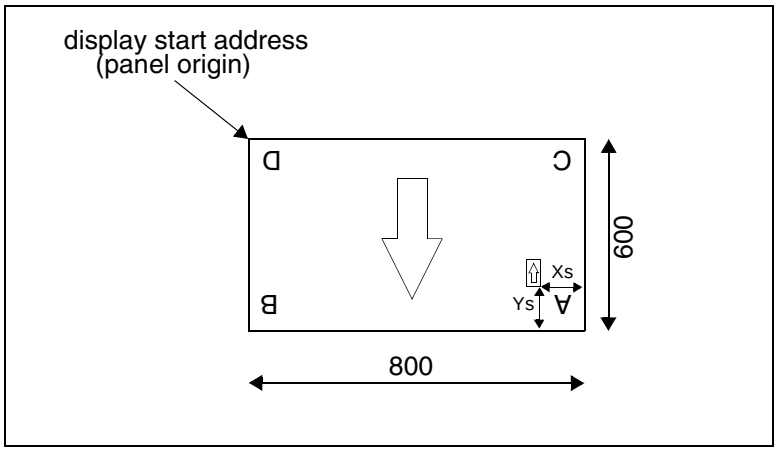


Figure 9-21: Relationship Between Cursor Position and the Image Refreshed in 180° Rotation

Programming

User must program the Cursor Width and Height according to their viewpoint.
Cursor data requires host assisted rotation support.

9.8.3 270° Rotation

The following figure shows how the Windowed Area is rotated on the display. The cursor position is referenced to A and its corner closest to A.

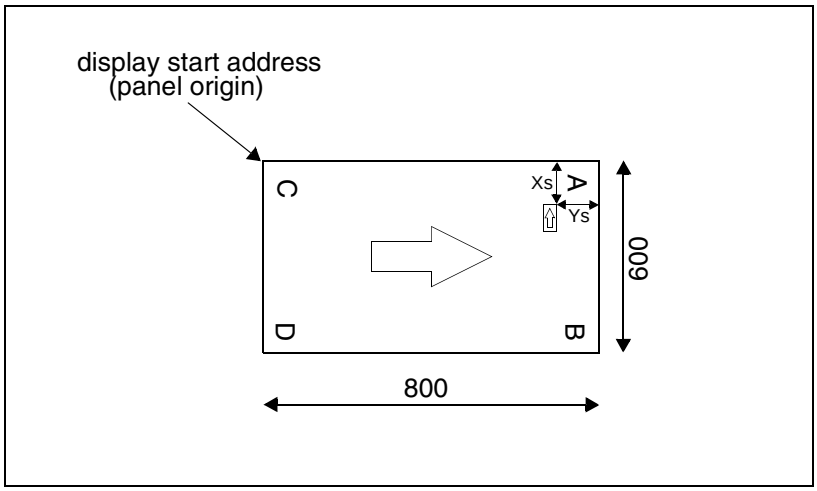


Figure 9-22: Relationship Between Cursor Position and the Image Refreshed in 270° Rotation

Programming

User must program the Cursor Width and Height according to their viewpoint.
Cursor data requires host assisted rotation support.

9.8.4 Cursor Rotation Example

The following figure shows the relationship between the rotated Cursor and the Main and PIP Image Area

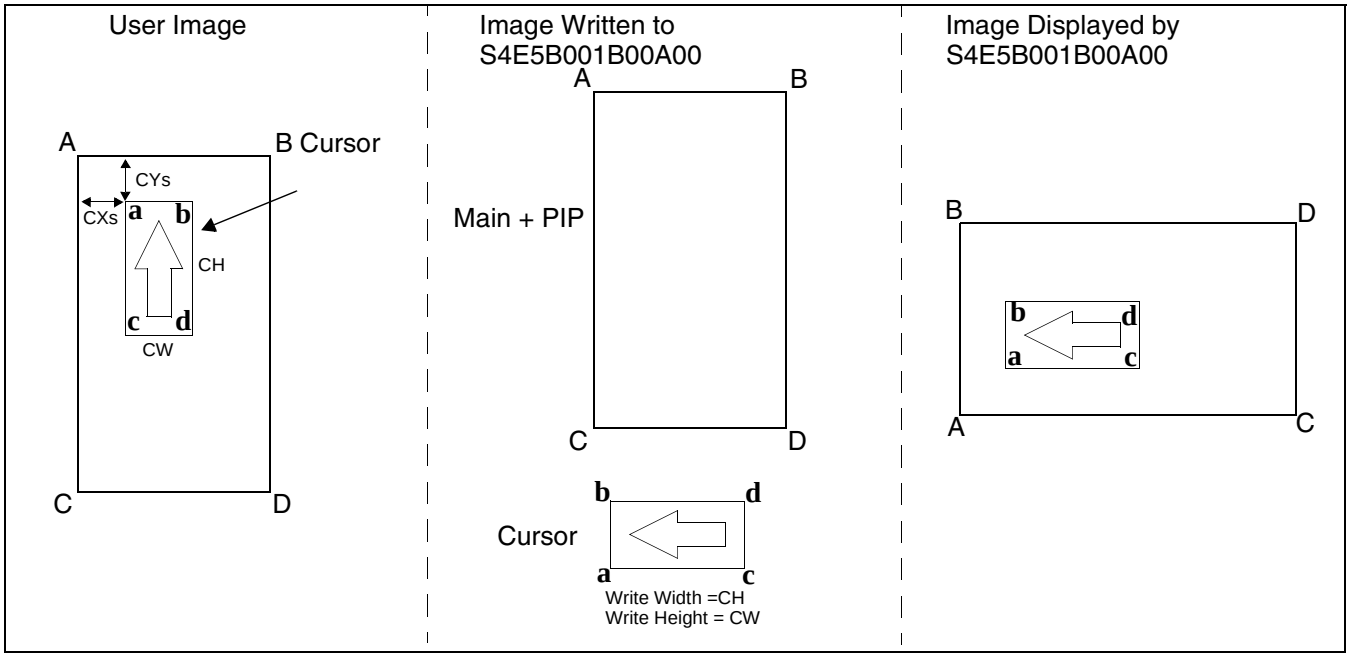


Figure 9-23: Relationship Between the User Image and the Image Displayed

Chapter 10 Host Memory Transfer Format

10.1 Host Interface Memory Transfer Mode

Data transfers from the Host interface to memory can be performed using either packed pixel mode or raw memory mode. It is recommended that image data is only programmed into the Image Buffer memory area. The Image Buffer start address is configured using the command UPD_SET_IMGADR. For further information on configuring the display memory, see Chapter 9, “Display Memory Configurations” on page 68.

Packed pixel mode “packs” multiple pixels into each 1-Nibble (4-bit) and provides a more efficient means of transferring pixel data to memory. Data transfers using raw memory mode always require 2pixels/byte for each pixel. Raw memory mode also allows memory reads. Note that pixel packing is done only for the data transfer and pixel data is stored in memory using unpacked 1-nibble format.

When the packed pixel data is unpacked, it is stored in the Image Buffer as shown below.

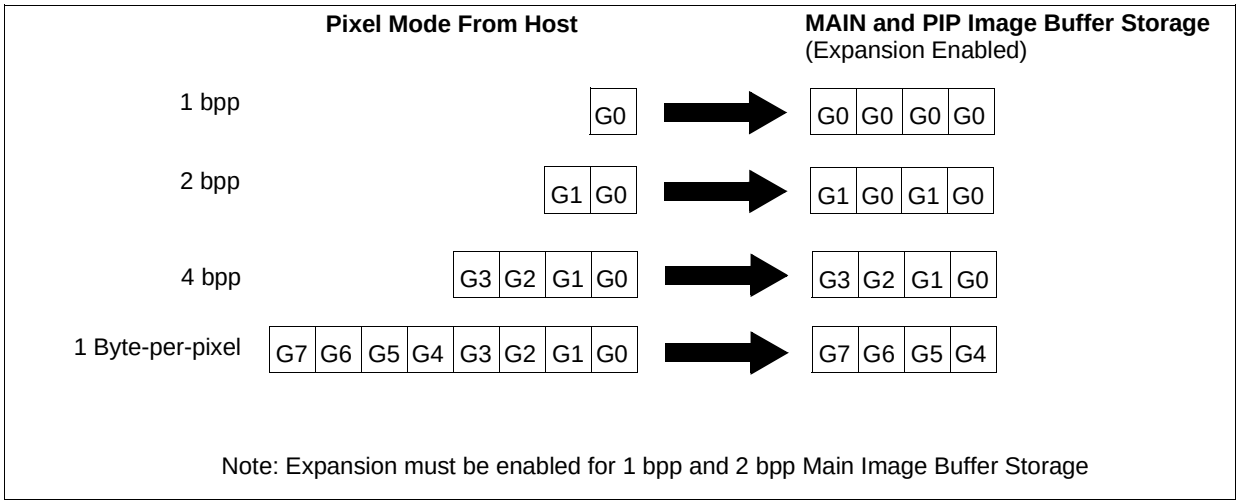


Figure 10-1: Packed Pixel Mode Data Unpacking Summary when Expansion is Enabled

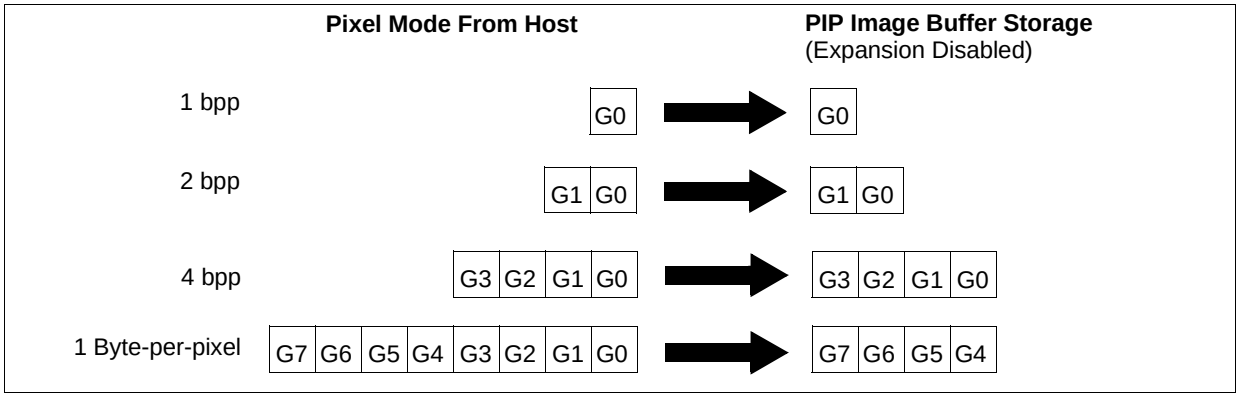


Figure 10-2: Packed Pixel Mode Data Unpacking Summary when Expansion is Disabled

10.2 Host Interface Packed Pixel Data Transfer Format Endian Formatting

Memory accesses support both Little Endian and Big Endian data transfers (see REG[015Eh] bits 1-0).

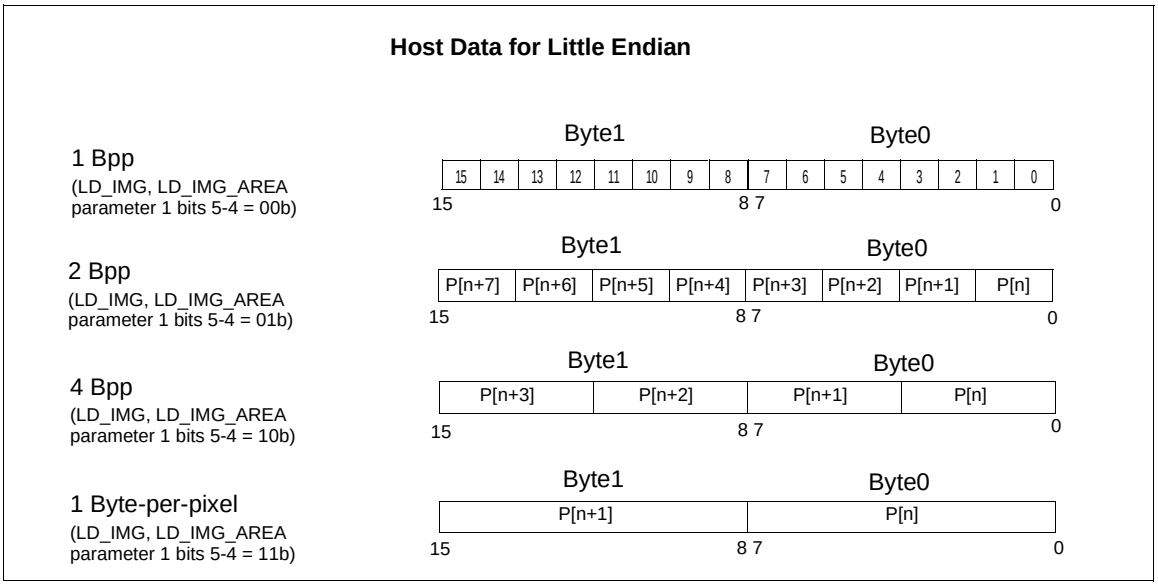


Figure 10-3: Host Interface Pixel Data Transfer Format for Little Endian

Note

1. For details on memory storage, see 9.2.1, “Image Buffer Storage” on page 70.
2. For details on Big Endian, see REG[015Eh] bits 1-0.

10.2.1 Packed Pixel Area Size Transfer Example

Note
Image data is always packed in 16-bit words with the last 16-bit word padded after Valid Data.

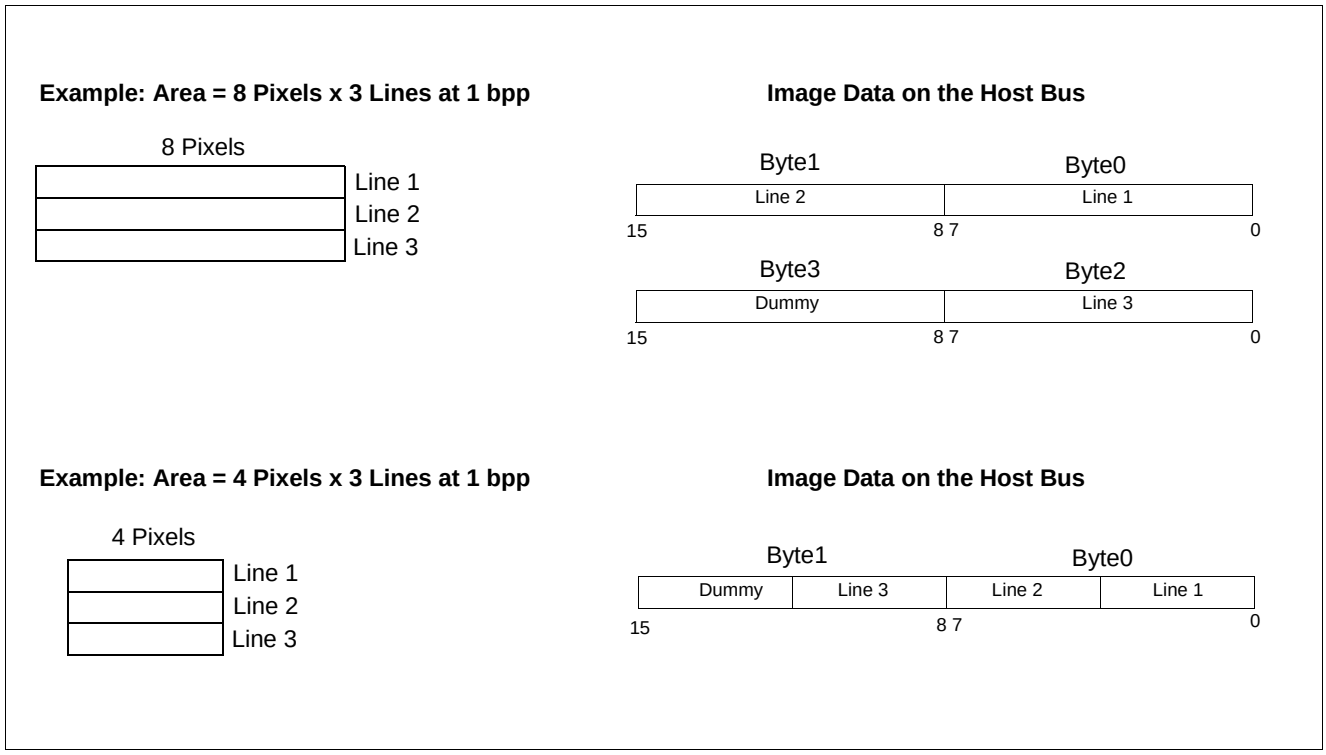


Figure 10-4: Packed Pixel Area Size Transfer Example

Chapter 11 Display Operations

11.1 Display Update: Multi-Region Pipeline Usage

The S4E5B001B00A00 Display Engine includes a high-performance 15-Region Pipeline. Each pipeline region may be assigned to:

- A group of pixels which do not overlap another pipeline region operation. The group of pixels doesn't need to be an enclosed rectangular region.
- A temperature compensated waveform Update Mode.

11.1.1 User Interface Applications with Multi-Region Pipeline

The 15-Region Pipeline is designed to provide fast response for User Interface applications while still maintaining the waveform update requirements and speed limitations. The 15-Region Pipeline is ideal for updating menu buttons, scroll bars, cursors, pen drawing, etc.

The following example shows a typical use for the 15-Region Pipeline.

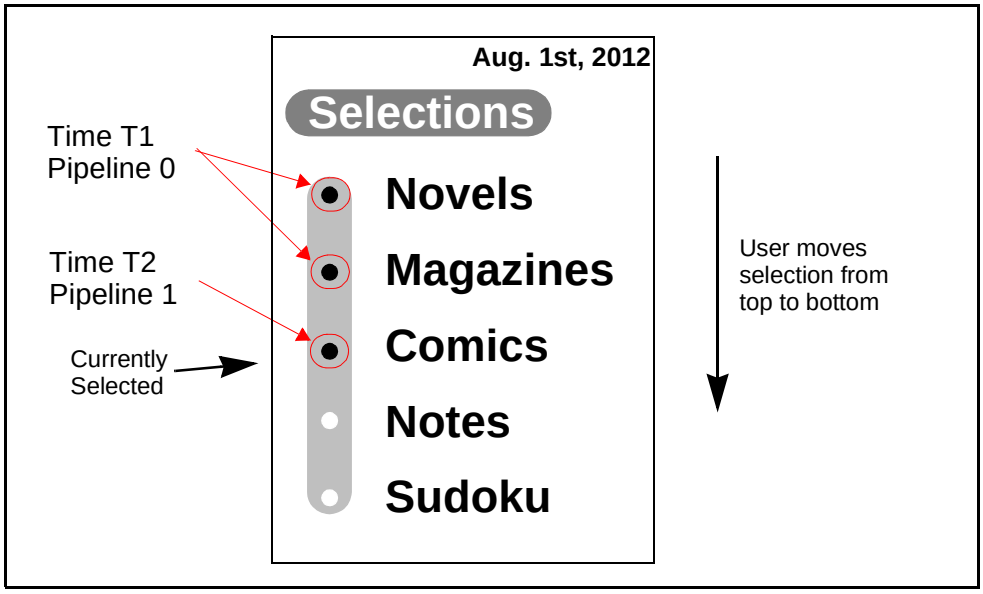


Figure 11-1: Example of a Simple Menu Button Movement

In the example, the starting position of the menu button is Novels. When the user moves the button from Novels to Magazines (T1), pipeline 0 may be initiated to update the 2 buttons to the desired state. However, this update may take up to 260ms to be completed. If the user further moves the button to Comics within the 260ms update time (T2), pipeline 1 may be assigned to the new position without waiting for the pipeline 0 to complete its assigned update operation.

The following figure shows the transition timing for the three buttons.

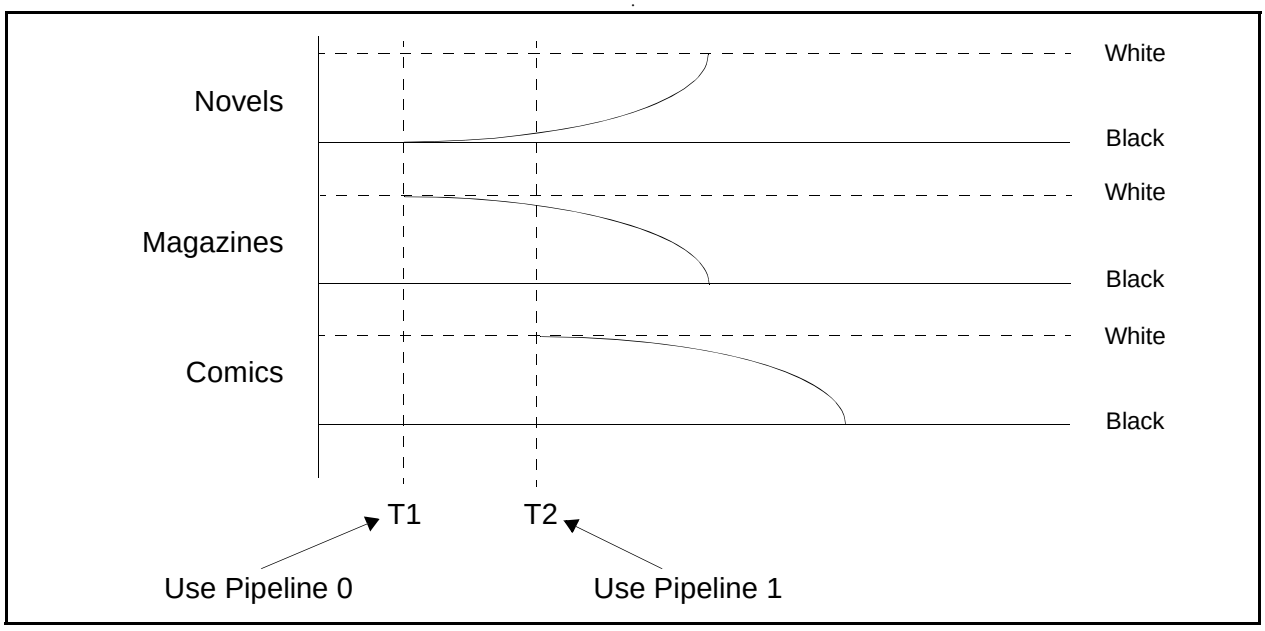


Figure 11-2: Transition Timing

11.1.2 Display Update Commands

There are four display update commands that may be used depending on the application requirement.

- UPD_FULL:
  This command forces an update on the entire display, even if the pixels have not changed levels.
- UPD_FULL_AREA
  This command forces an update on the defined rectangular area, even if the pixels have not changed levels.
- UPD_PART
  This command detects pixel changes between the image buffer and the current display for the entire display. Then, it updates the pixels with changed levels. If no pixel changes are detected, no display update takes place.
- UPD_PART_AREA
  This command detects pixel changes between the image buffer and the current display for the defined rectangular area. Then, it updates the pixels with changed levels. If no pixel changes are detected, no display update takes place.

11.1.3 Pipeline Usage

Pipeline Assignment

The S4E5B001B00A00 will automatically assign the highest numbered free pipeline on any update display operation. If no pipeline is available, the command will stop and report an error in the Pipeline Request Error Interrupt Raw Status bit, REG[033Ah] bit 9.

11.2 Guaranteed Display Update Operation Flow

Performing a display update operation involves 3 steps. In order to guarantee that the display update finishes completely, it is recommended that the Host software does not perform any image buffer memory write operations during Panel Update Buffer Synthesis (step 2). Otherwise, the display may be only partially updated resulting in display “tearing”.

1. **Host Memory Write Operation**
   The Host Memory Write Operation is dependent on the host software. For example, the Host may perform a multiple burst write before issuing an Update Trigger command.

2. **Panel Update Buffer Synthesis**
   Once an Update Trigger command is issued, Panel Update Buffer Synthesis is started. This step processes the data in the Image Buffer and places the appropriate data in the Panel Update Buffer for the next display operation. The Panel Update Buffer is not directly accessed by the Host software.

   Panel Update Buffer Synthesis processing time is dependent on whether Display Frame Output is already active. If Display Frame Output is not currently active, Panel Update Buffer Synthesis takes a maximum of 6ms for a 800x600 panel. If Display Frame Output is active, it takes a maximum of 2 ÷ FrameRate for a 800x600 panel.

3. **Display Frame Output**
   Display Frame Output starts automatically right after Panel Update Buffer Synthesis completes.

The following figure shows the Guaranteed Display Update Operation Flow steps.

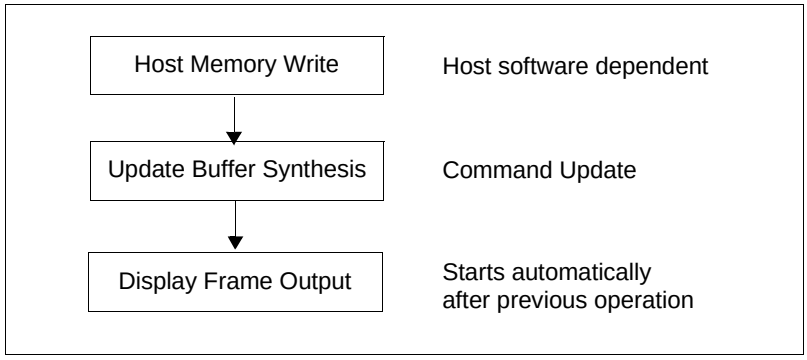


Figure 11-3: Guaranteed Display Update Operation Flow

11.3 Overlapping Display Update Operation Flow

11.3.1 Overlapping Display Updates

Under special conditions, an overlapping display update may be performed. An overlapping display update is defined as a partial display update that “overlaps” a display area which is already being updated. The following figure shows an example of this situation.

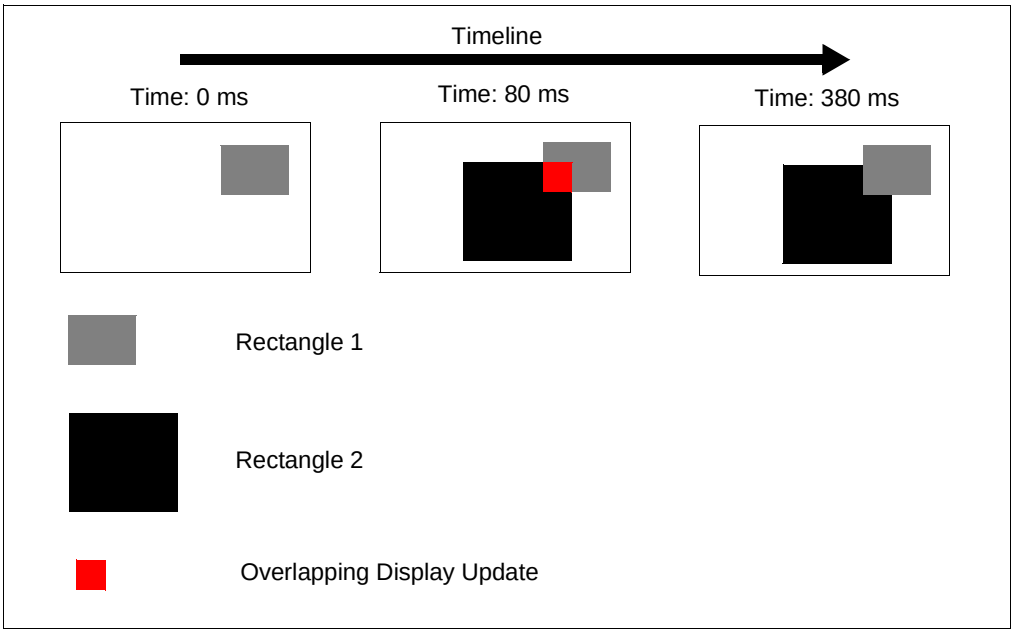


Figure 11-4: Overlapping Display Update Example

In this example, assume each display update will take 300 ms to complete. At time “0 ms”, the partial display update of Rectangle 1 is initiated. At time “80 ms”, a partial display update of Rectangle 2 is initiated. The area shown in “Red” is desired to have Rectangle 2 overlapping Rectangle 1. However, at time “380 ms”, once the partial display update of both rectangles are completed, rectangle 1 overlaps rectangle 2. To achieve the desired overlapping, the procedure described in 11.3.2, “Overlapping Display Update Recommended Programming Flow” on page 95 must be followed.

11.3.2 Overlapping Display Update Recommended Programming Flow

When overlapping display updates are performed, the Pipeline Busy Conflict Detected Interrupt Status and Raw Status bits (REG[033Ah] bit 7 and REG[033Ch] bit 7) can be used to detect Overlapping Display Update error conditions. This interrupt is triggered for every overlapping pixel, so it is recommended to check this bit only after the Image Buffer has been fully synthesized into the Panel Update Buffer (WAIT_DSPE_TRG).

The following figure shows the recommended procedure to correct overlapping display update errors.

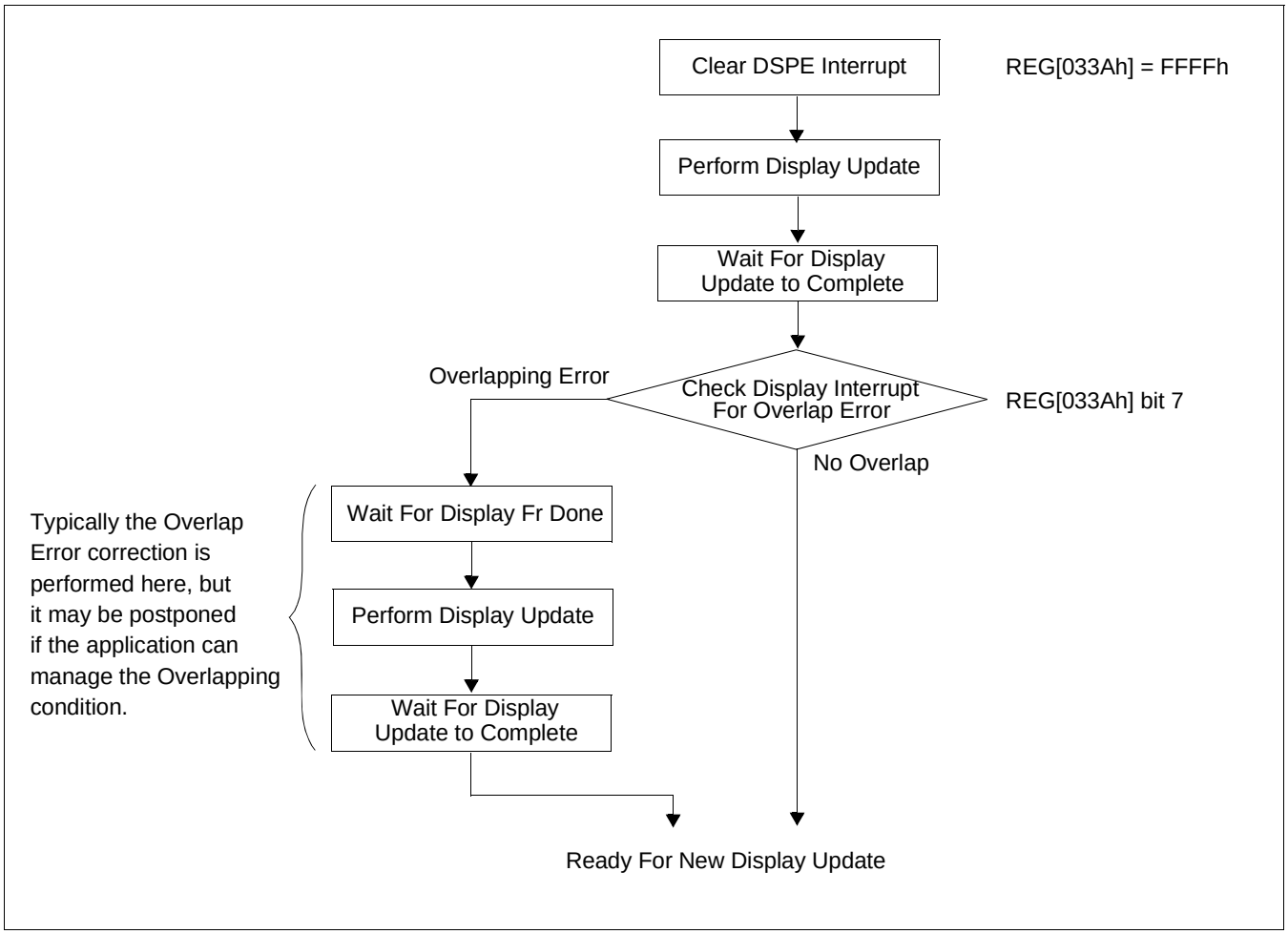


Figure 11-5: Overlapping Display Update Error Correction Programming Flow

11.4 Multi-layer Image Buffer Support

S4E5B001B00A00 controller supports 3 display layers. The following table shows an overview of each layer

Table 11-1: Multi-Layer Overview

Name	Layer Position	Layer Max Size	Layer Bpp Native Storage	Transparency Support	Additional Functions	Data Storage Location
Cursor Image	Top most	128x128x1bpp	Limited by Cursor Memory	1bpp, 2bpp and 4bpp transparent grey scale	0 - Replace 1 - XOR with Background 2 - ~XOR with background 3 - Invert Cursor Data	Internal Dedicated SRAM
Pip Image	Middle	Display Size	1bpp, 2bpp or 4bpp	1bpp, 2bpp and 4bpp transparent grey scale	0 - Replace 1 - XOR with Background 2 - ~XOR with background 3 - Invert PIP Data	Shared Memory
Main Image	Bottom layer	Display Size	4bpp	N.A.	NA	Shared Memory

The following diagram describe how each layer will be displayed.

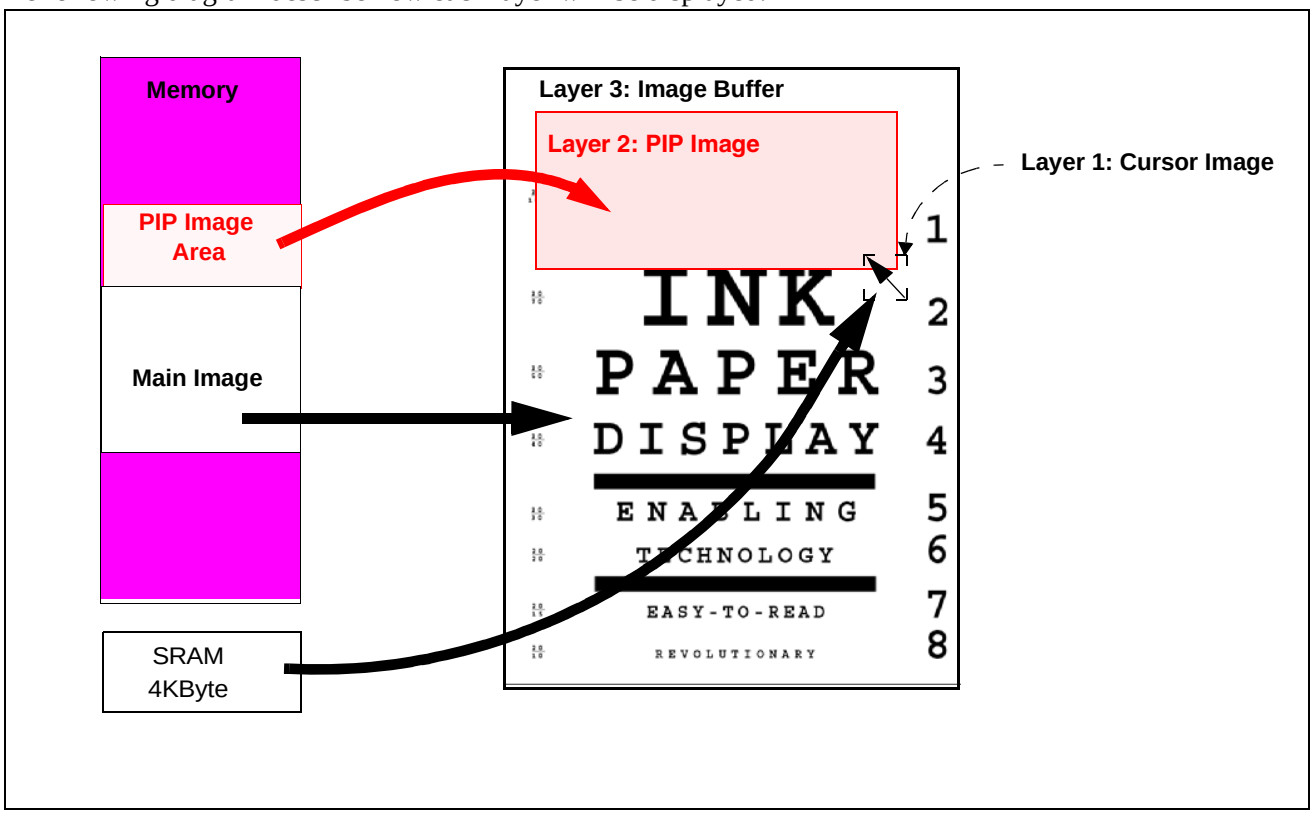


Figure 11-6: Multi-layer Image Buffer Support

11.5 PIP Window Support

A PIP Window overlays a new window (foreground image) on top of the currently displayed image (background image) without overwriting it. This function allows the background image to be restored without requiring the Host to rewrite the Image Buffer. PIP Windows are implemented using a separate PIP image buffer.

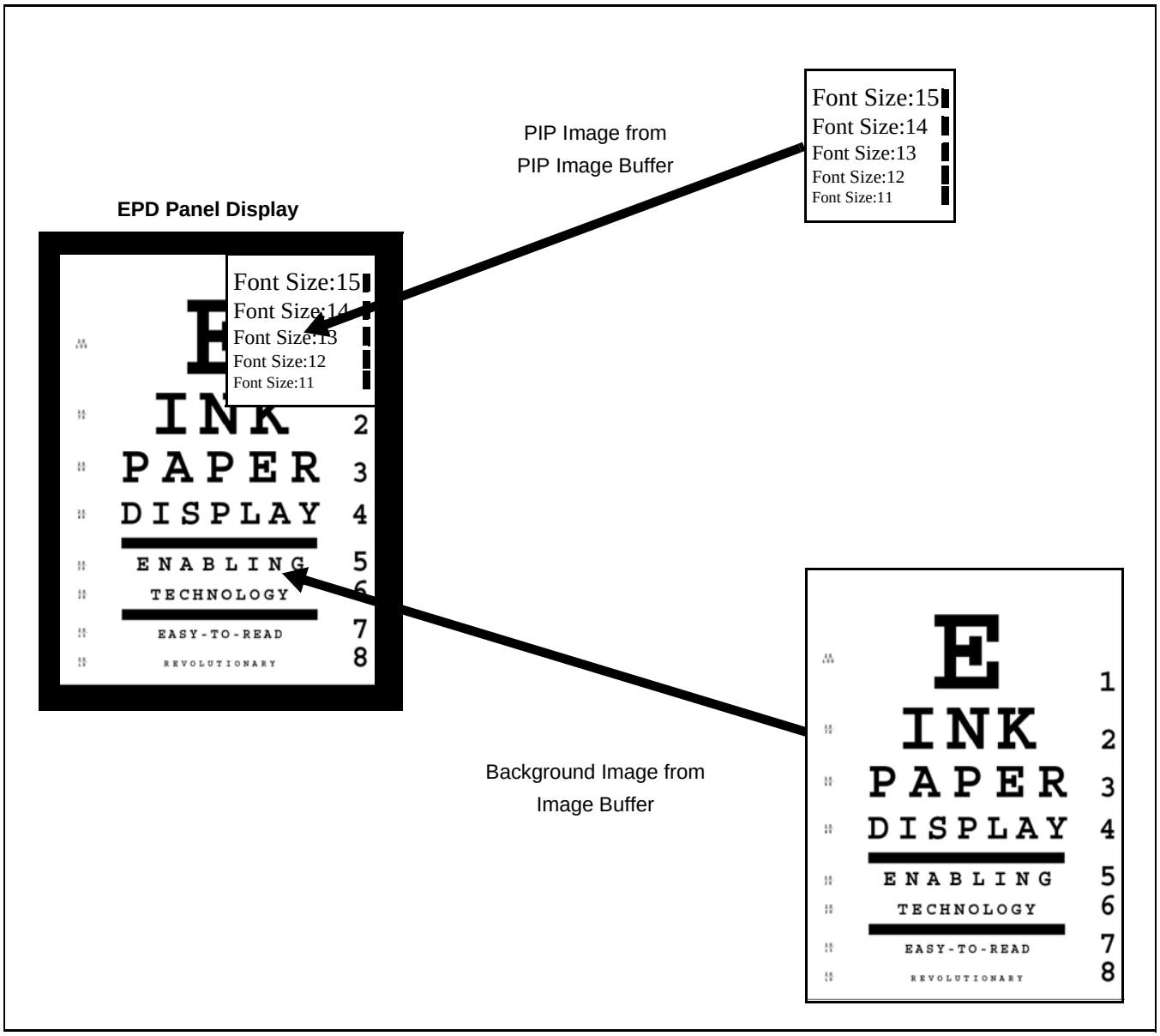


Figure 11-7: Pop-Up Window Example

11.5.1 PIP Window Implementation

The S4E5B001B00A00 can display one Main Image Buffer and one PIP Image Buffer overlaid on top of the Main Image Buffer itself.

To create a PIP Window, the following steps are recommended.

Step 1: Update the Main Image (full image update).

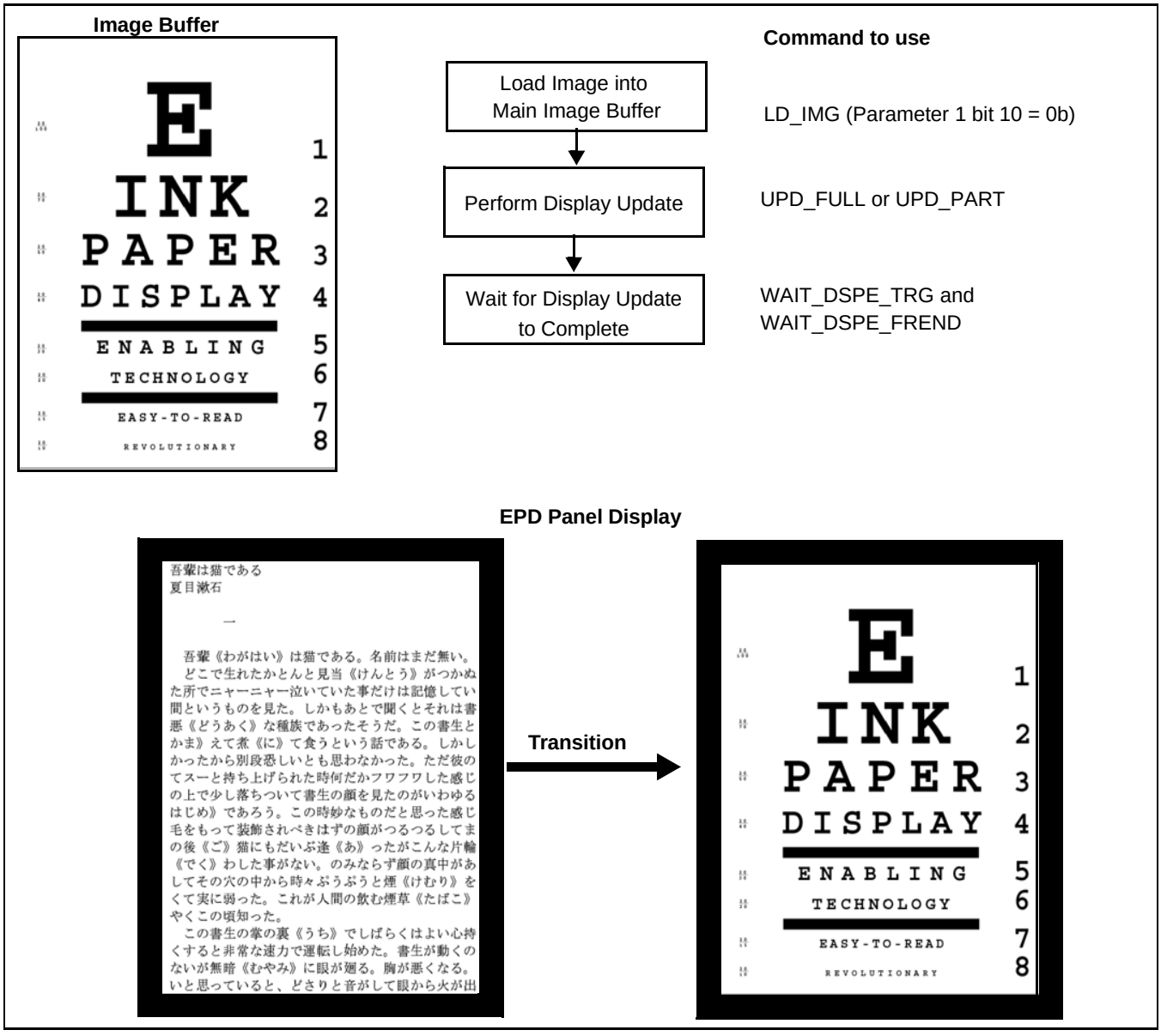


Figure 11-8: Creating a PIP Window - Step 1

Step 2: Displaying the PIP Image.

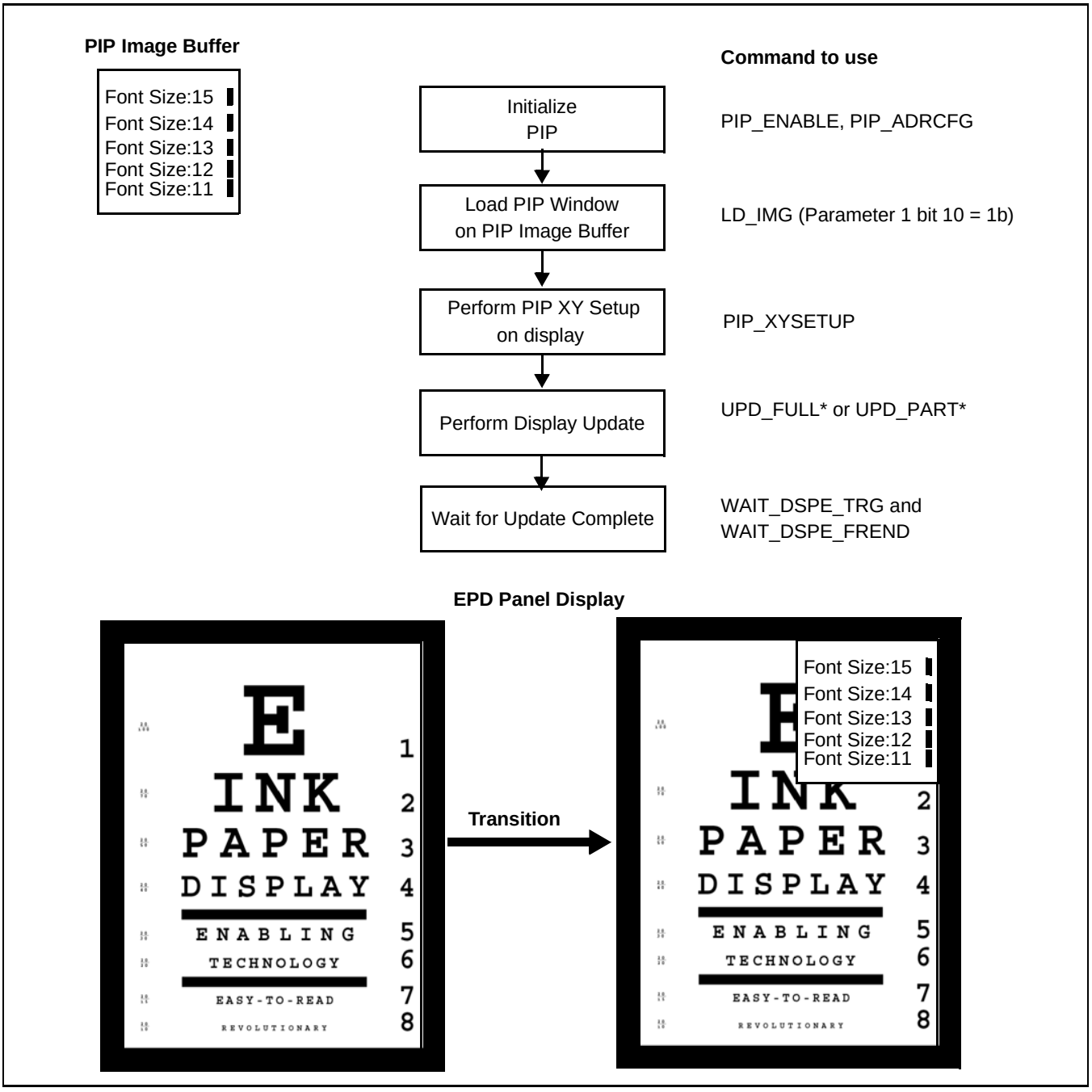


Figure 11-9: Creating a PIP Window - Step 2

Step 3: PIP Image disable.

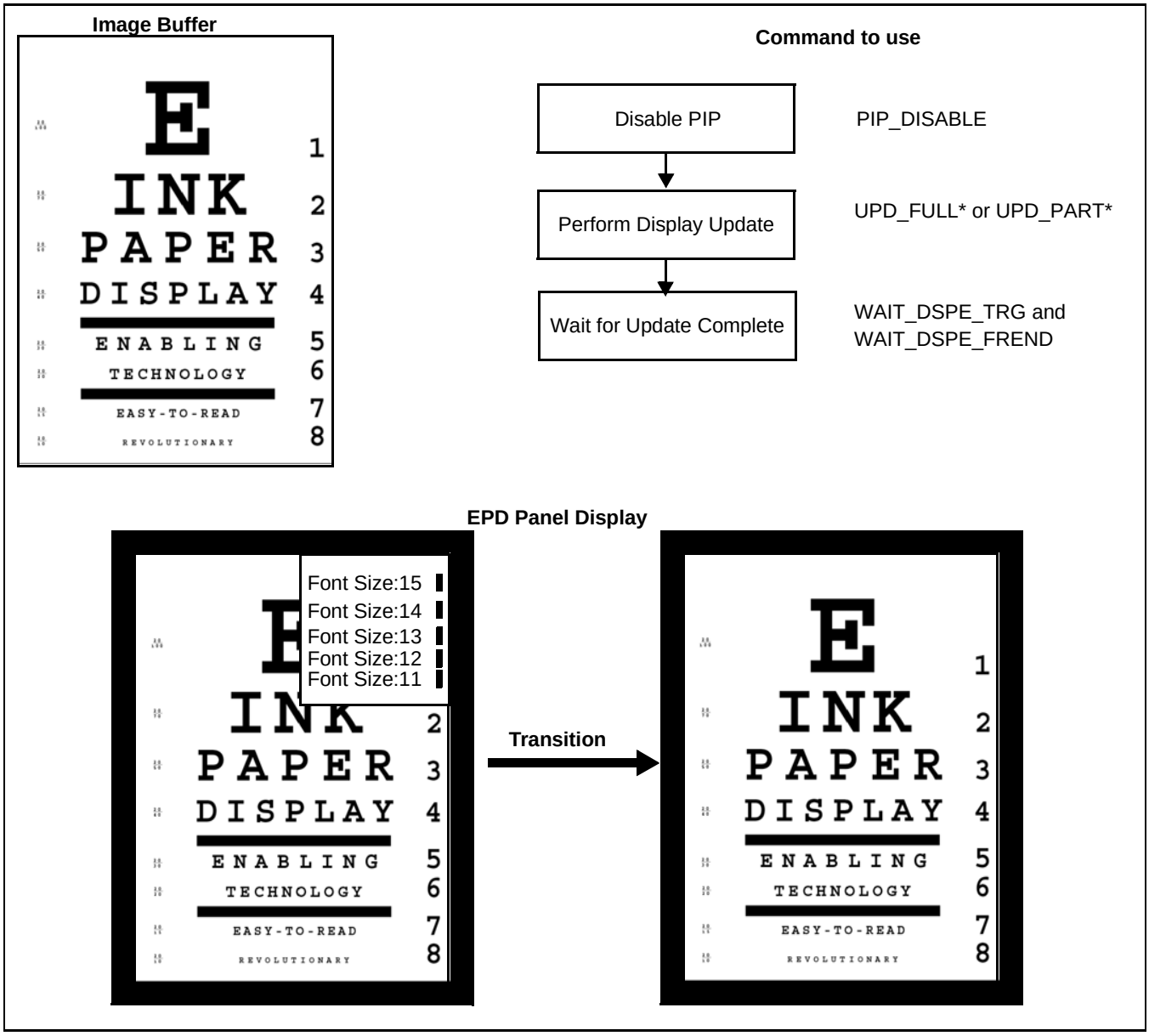


Figure 11-10: Creating a PIP Window - Step 3

11.6 Cursor Window Support

A Cursor Window overlays a new window (foreground image) on top of the currently displayed image (background image) without overwriting it. Background image includes PIP and Image Buffer display memory. This function allows the background image to be restored without requiring the Host to rewrite the background content. Cursor Windows are implemented using a separate Cursor image buffer.

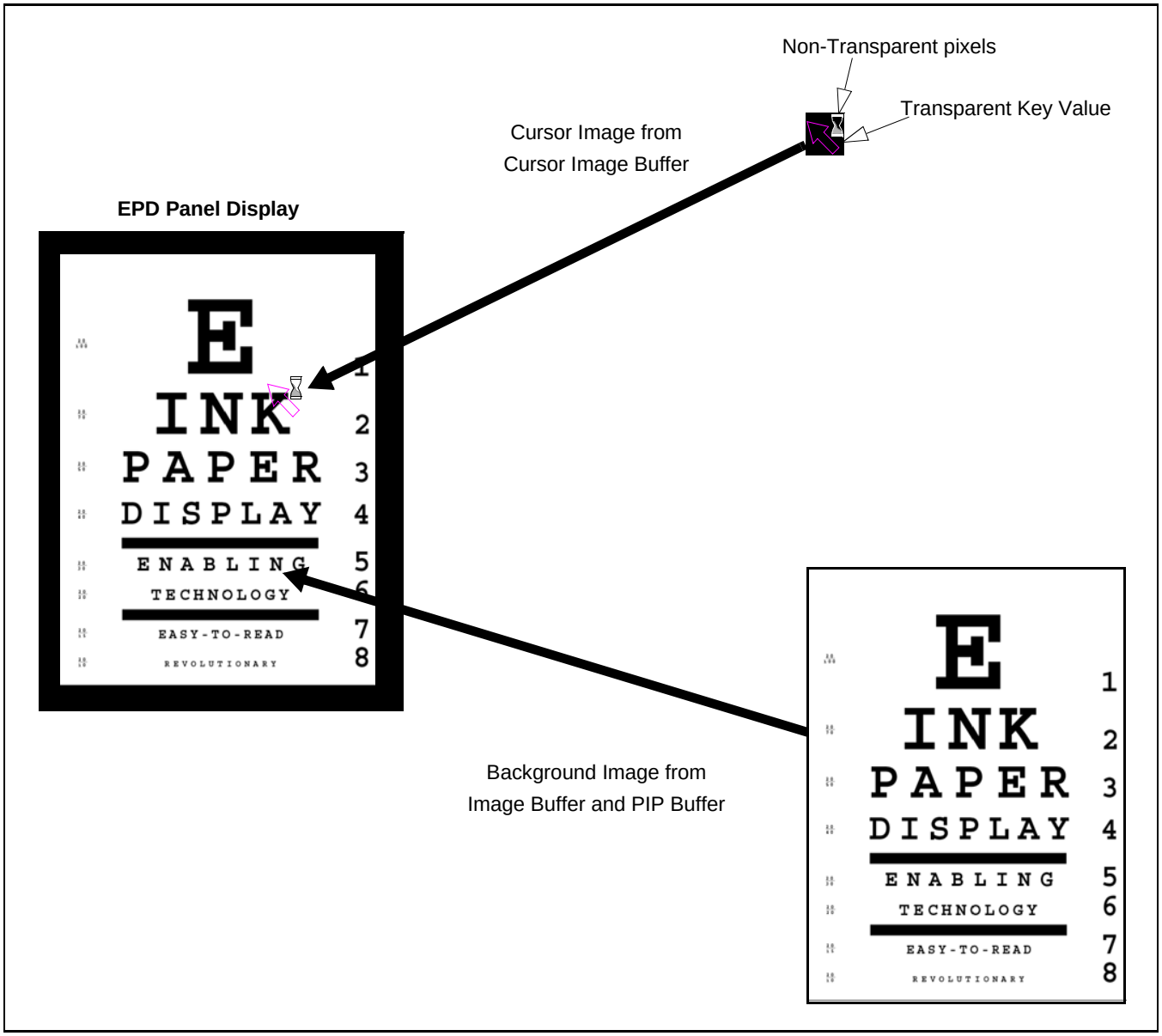


Figure 11-11: Cursor Example

11.6.1 Cursor Window Implementation

The S4E5B001B00A00 can have one Cursor Image Buffer overlaid on top of the Main and PIP Image Buffer itself.

To create a Cursor Window, the following steps are recommended.

Step 1: Update the Background Image (full image update).

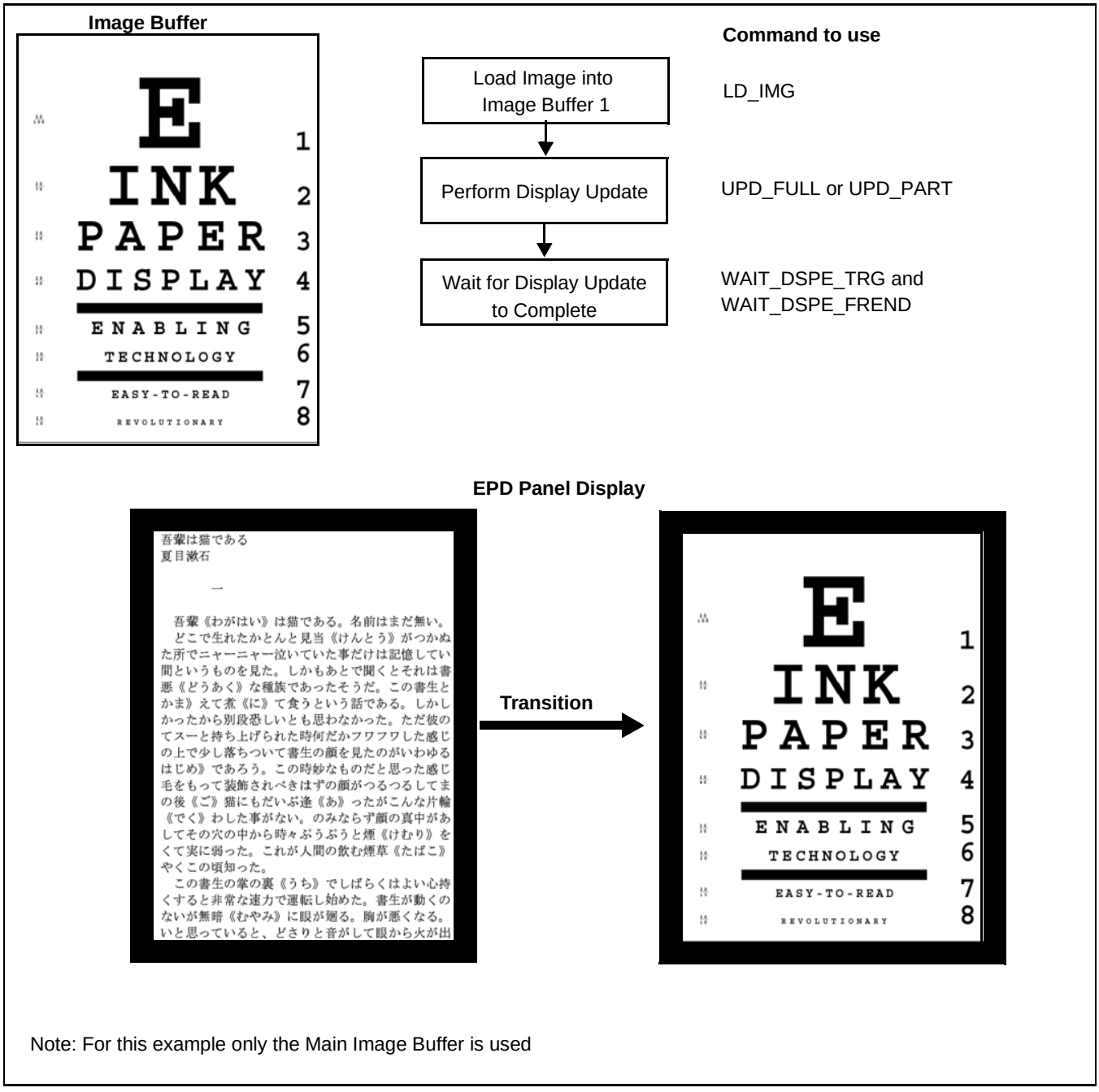


Figure 11-12: Creating a Cursor Window - Step 1

Step 2: Update the Cursor Image.

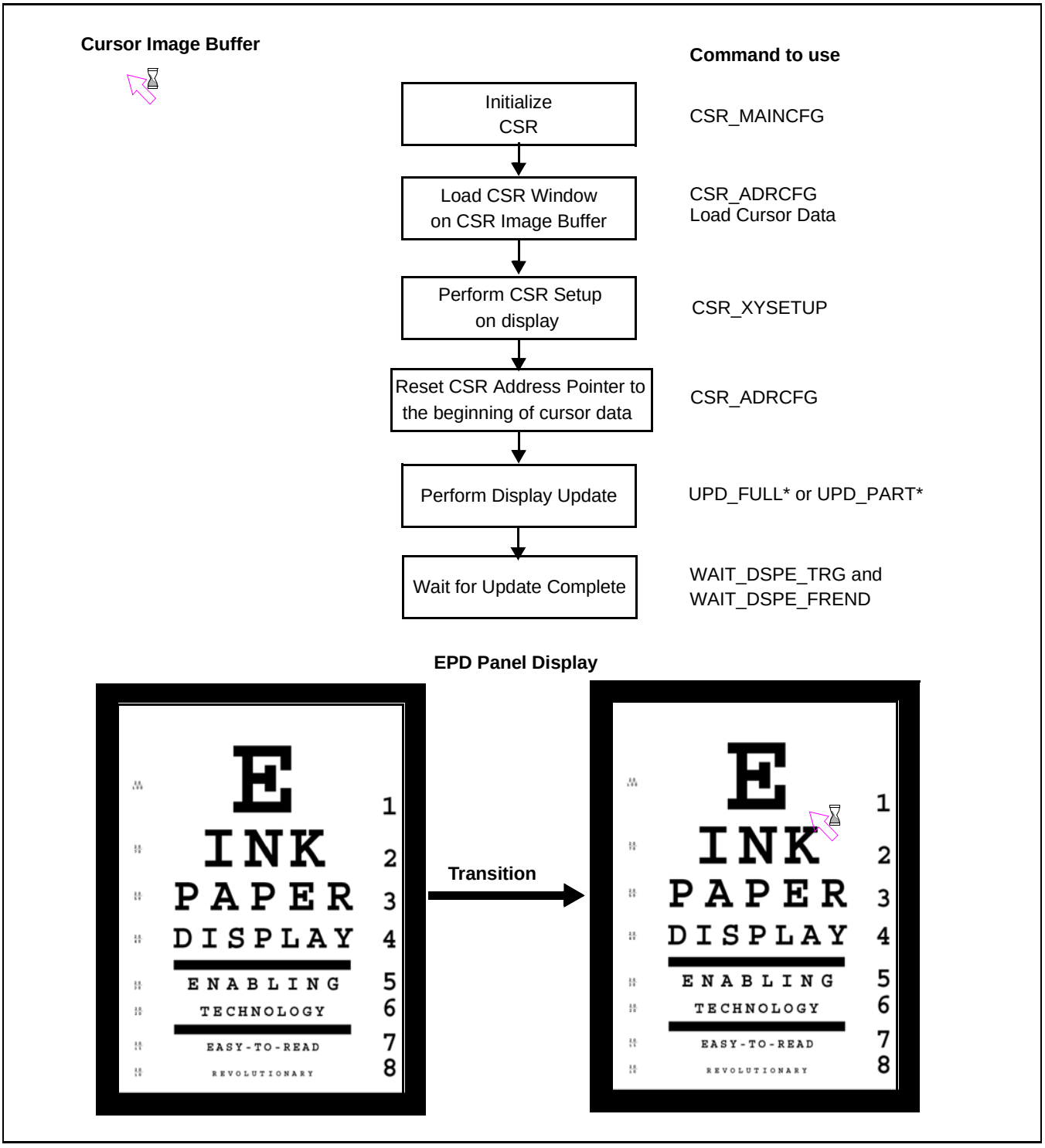


Figure 11-13: Creating a Cursor Window - Step 2

Step 3: Disabling the Cursor.

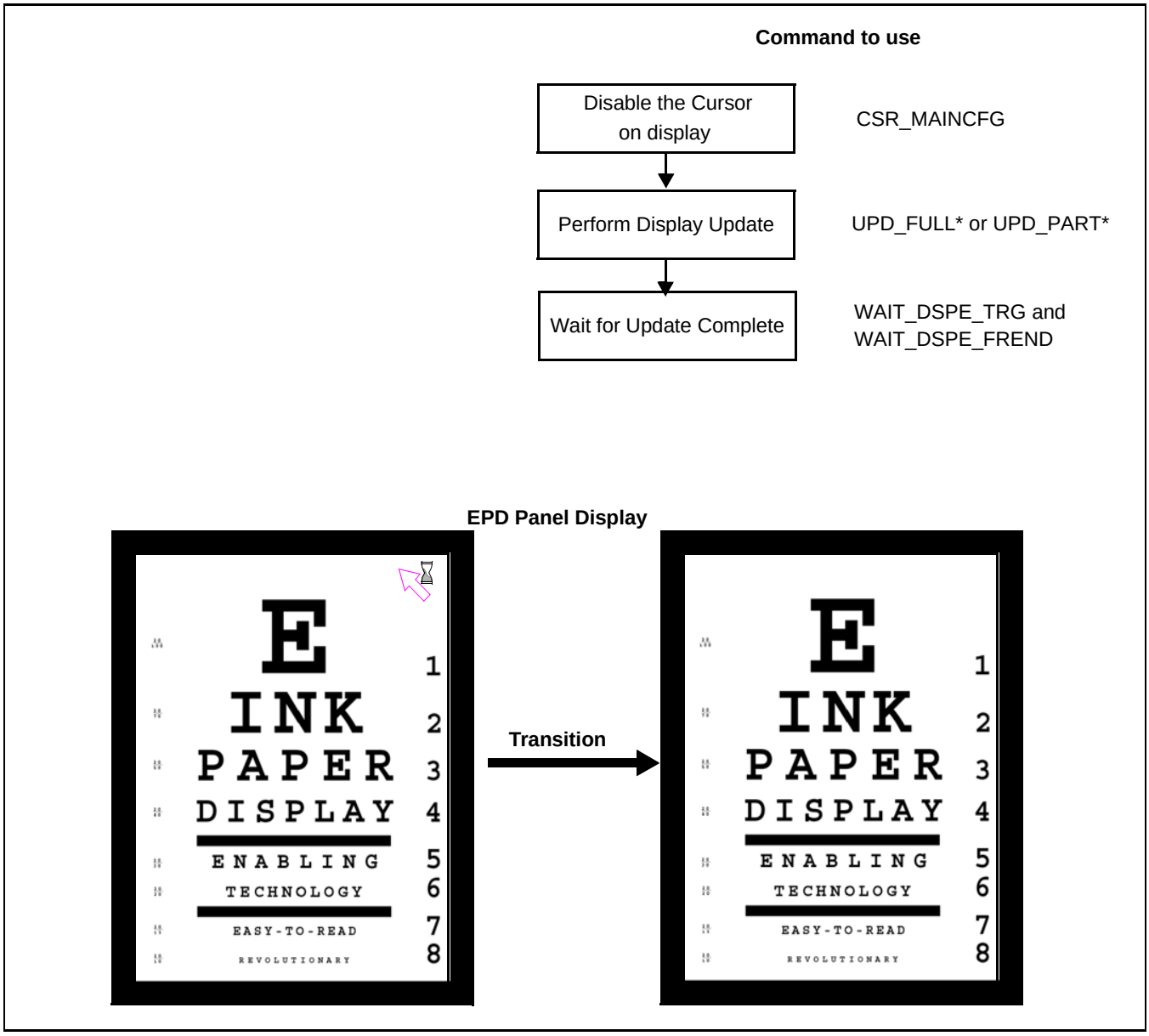


Figure 11-14: Creating a Cursor Window - Step 3

Chapter 12 Firmware Programming Guide

12.1 Command Based Programming

The following sections contain programming sequences for:

- Power On Initialize
- Standby Mode
- Sleep Mode
- Sleep Mode
- Run Mode
- Host Packed Pixel Write - to Main or PIP with Transparency
- Host Area Defined Packed Pixel Write - to Main or PIP
- Full Image Dump (16-bit Host Interface only)
- Manual Trigger to Thermal Sensor for Single Temperature Read
- Display Engine - Panel Update Buffer Initialize with Image Buffer
- Display Engine - Full Display Update - Full Image Size Sweep
- Display Engine - Partial Display Update - Full Image Size Sweep
- Display Engine - Partial Display Update - User Defined Area Sweep
- Display Engine - Picture-In-Picture Setup
- Display Engine - Picture-In-Picture Position Change
- Display Engine - Cursor Setup
- Display Engine - Cursor Position Change

12.1.1 Power On Initialize

The Serial Flash has been previously programmed with the Epson Instruction code with and E-Ink Waveforms.

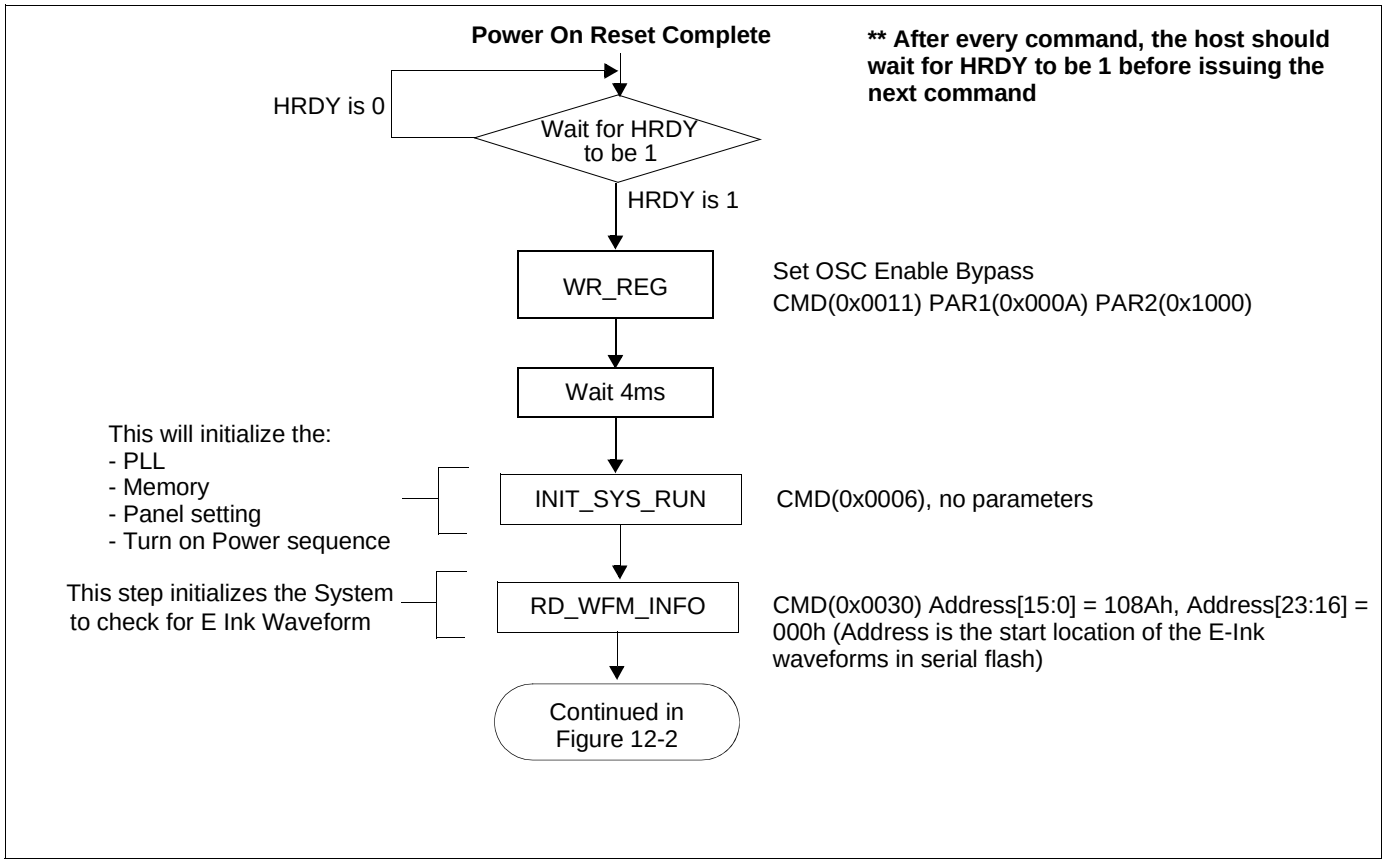


Figure 12-1: Power On Initialize Programming Flow (1 of 2)

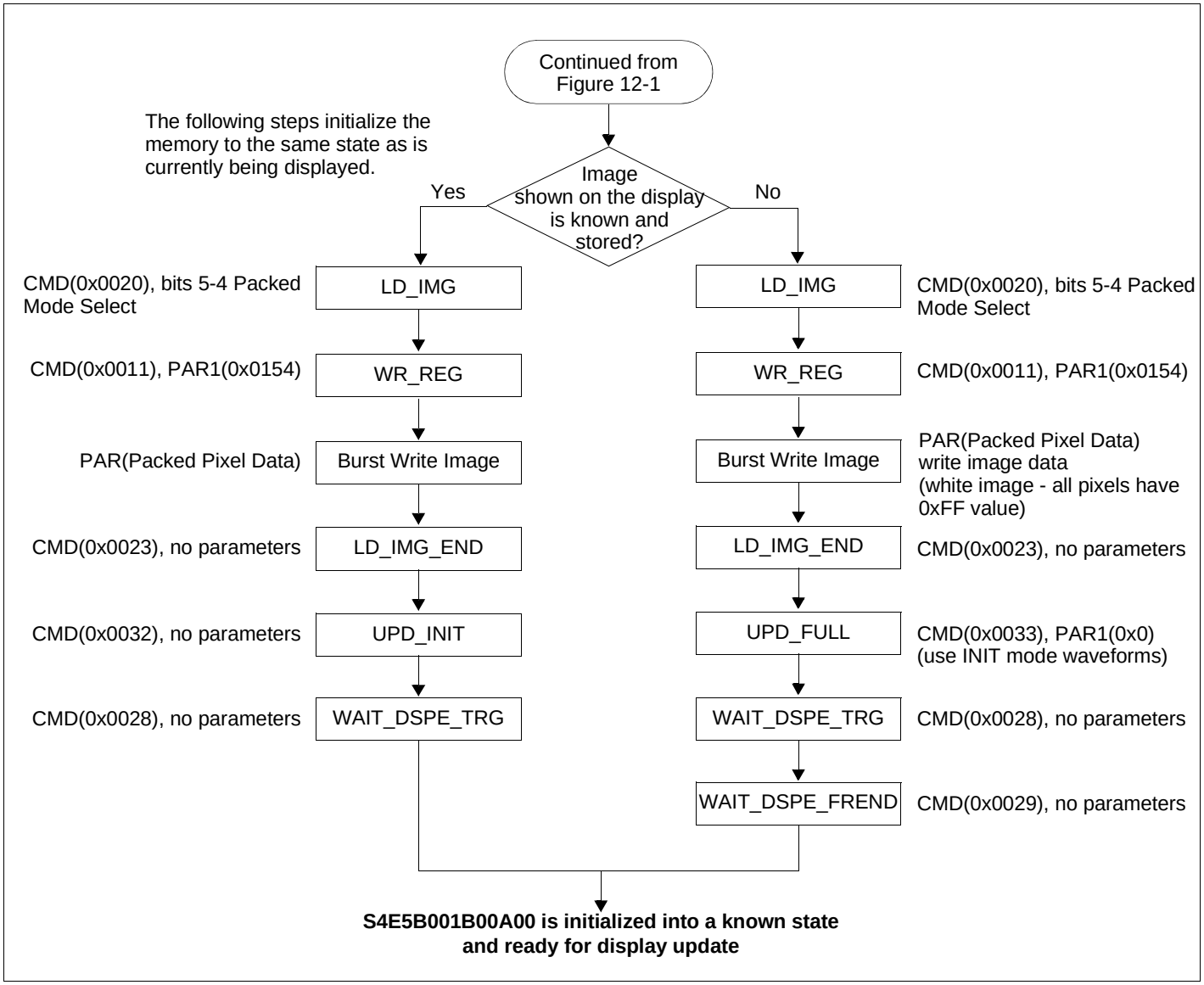


Figure 12-2: Power On Initialize Programming Flow (2 of 2)

12.1.2 Standby Mode

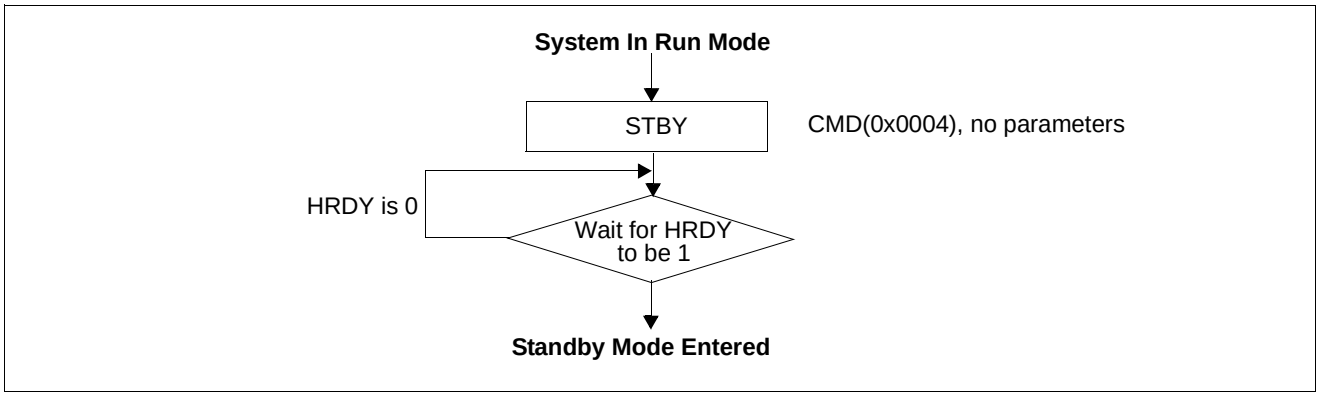


Figure 12-3: Standby Mode Programming Flow

12.1.3 Sleep Mode

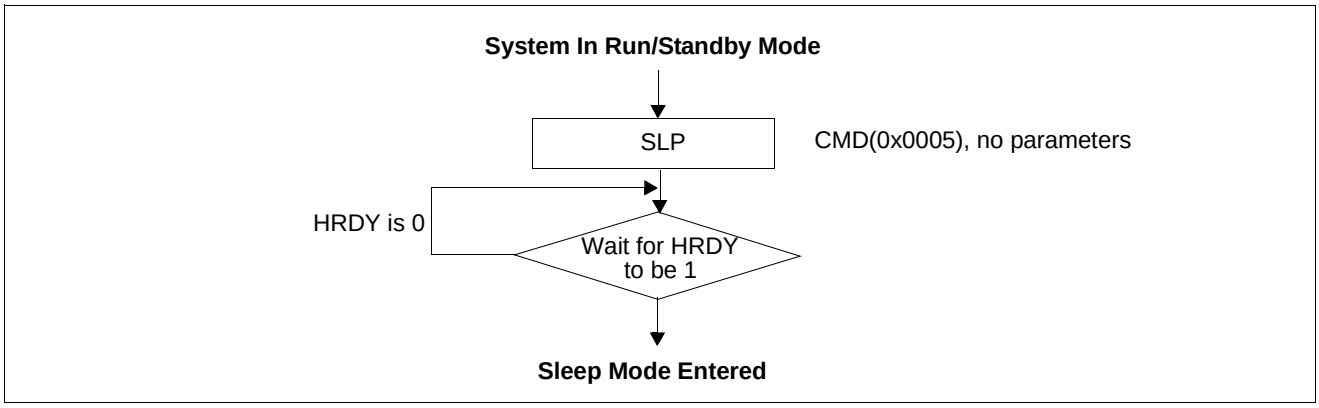


Figure 12-4: Sleep Mode Programming Flow

12.1.4 Run Mode

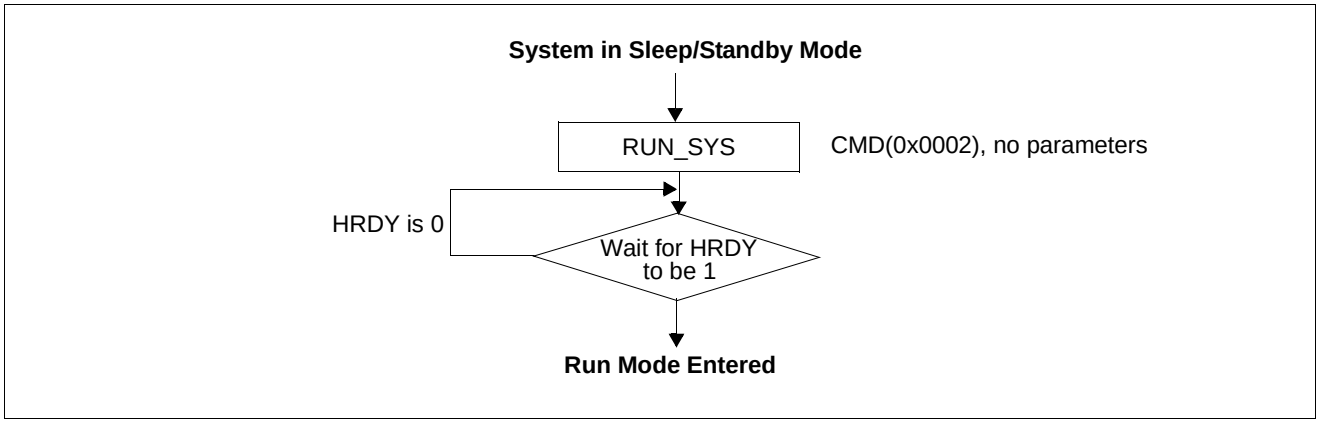


Figure 12-5: Run Mode Programming Flow

12.1.5 Host Packed Pixel Write - to Main or PIP with Transparency

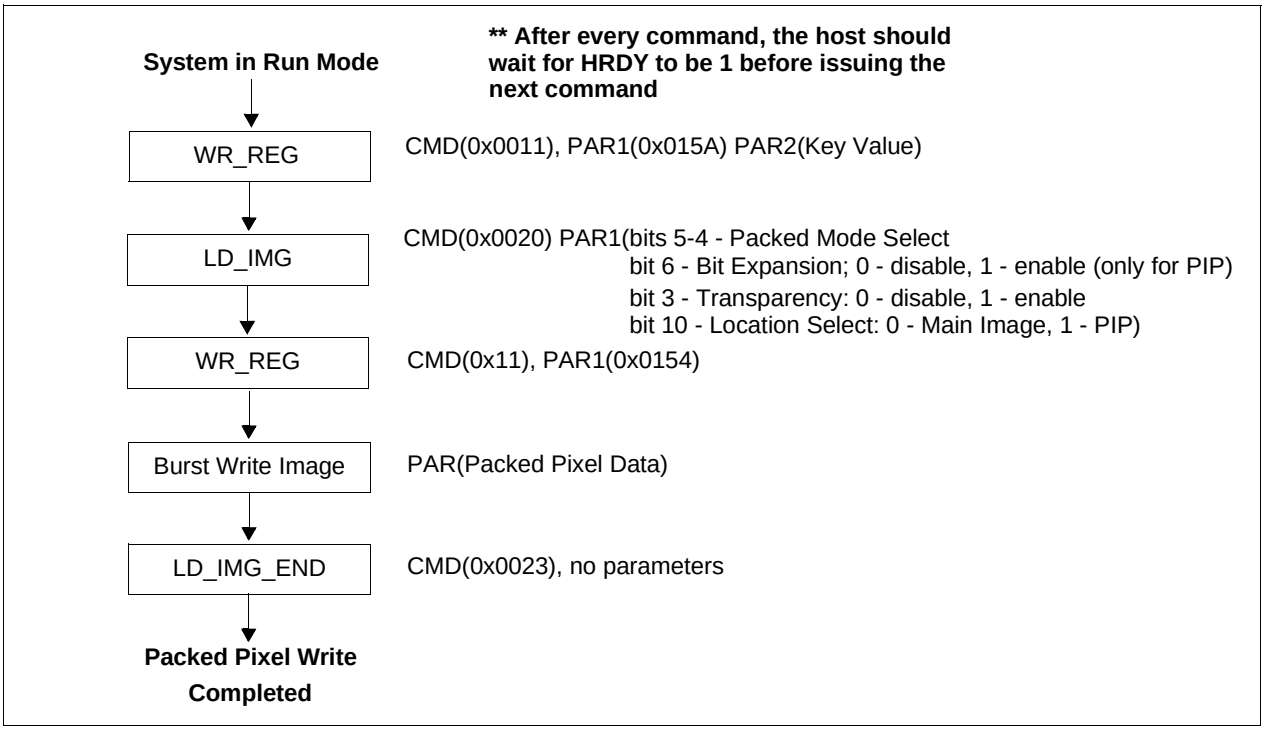


Figure 12-6: Host Packed Pixel Write Programming Flow

12.1.6 Host Area Defined Packed Pixel Write - to Main or PIP

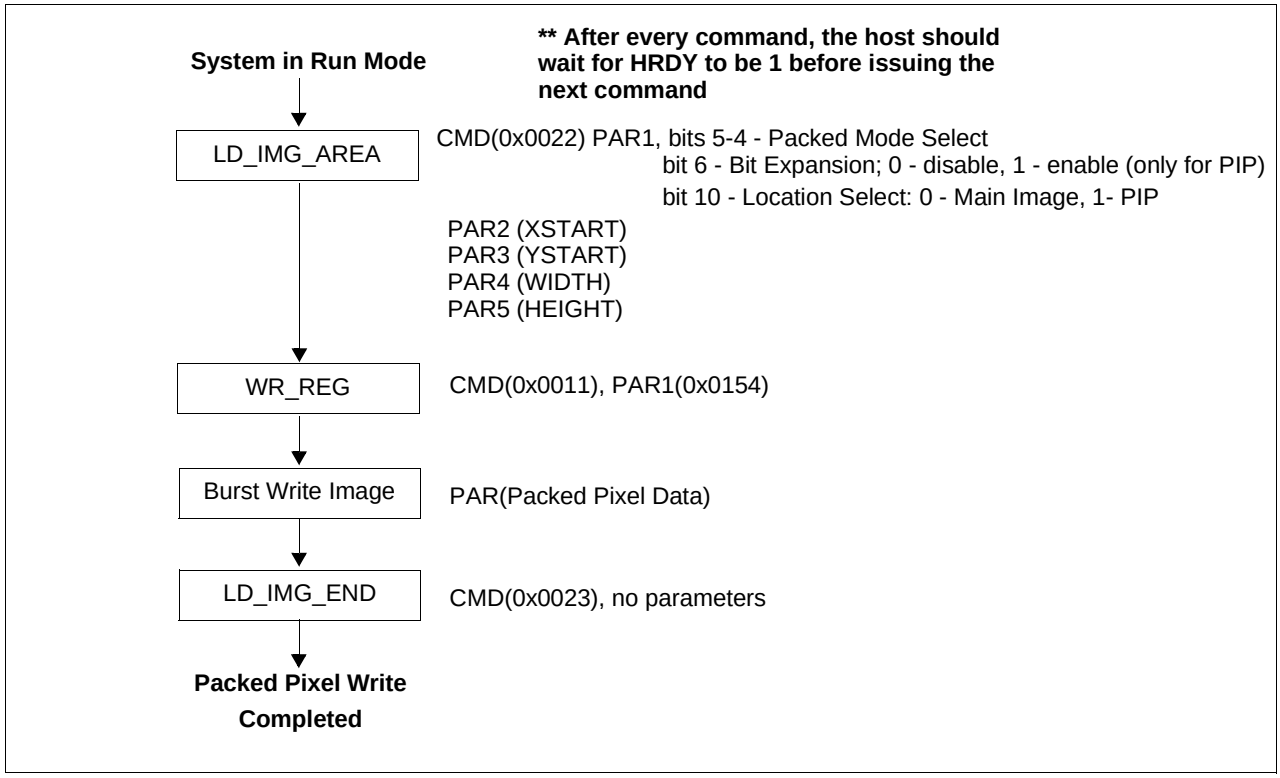


Figure 12-7: Host Area Defined Packed Pixel Write Programming Flow

12.1.7 Manual Trigger to Thermal Sensor for Single Temperature Read

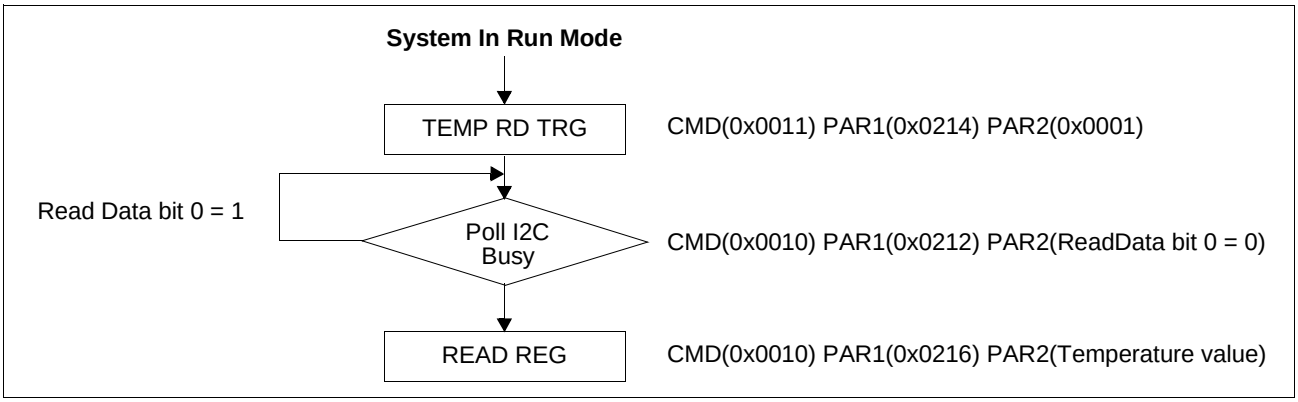


Figure 12-8: Manual Trigger Thermal Sensor for Single Temperature Read Programming Flow

12.1.8 Full Image Buffer Dump

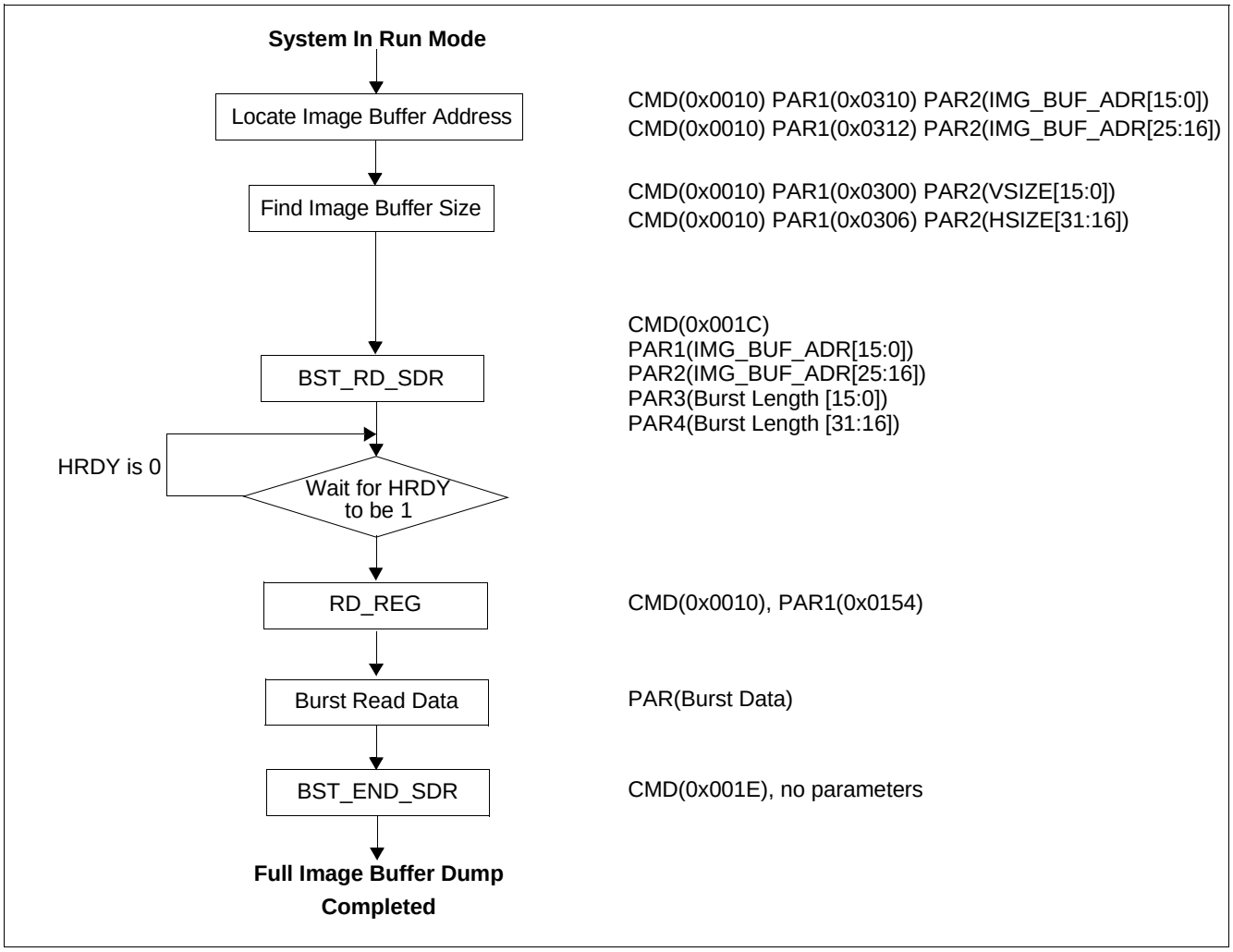


Figure 12-9: Full Image Buffer Dump Programming Flow

Note
Memory read is not supported for Intel 80 8-bit Host Interface.

12.1.9 Display Engine - Panel Update Buffer Initialize with Image Buffer

This sequence will initialize the panel update buffer without display output.

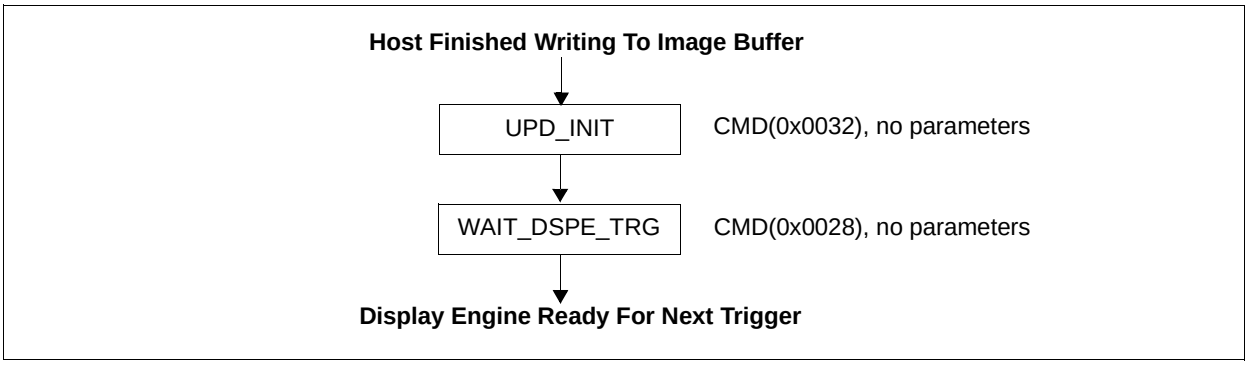


Figure 12-10: Display Engine - Panel Update Buffer Initialize with Image Buffer Programming Flow

12.1.10 Display Engine - Full Display Update - Full Image Size Sweep

This operation forces a display update for every pixel within the defined display size, regardless of whether the pixel data has changed.

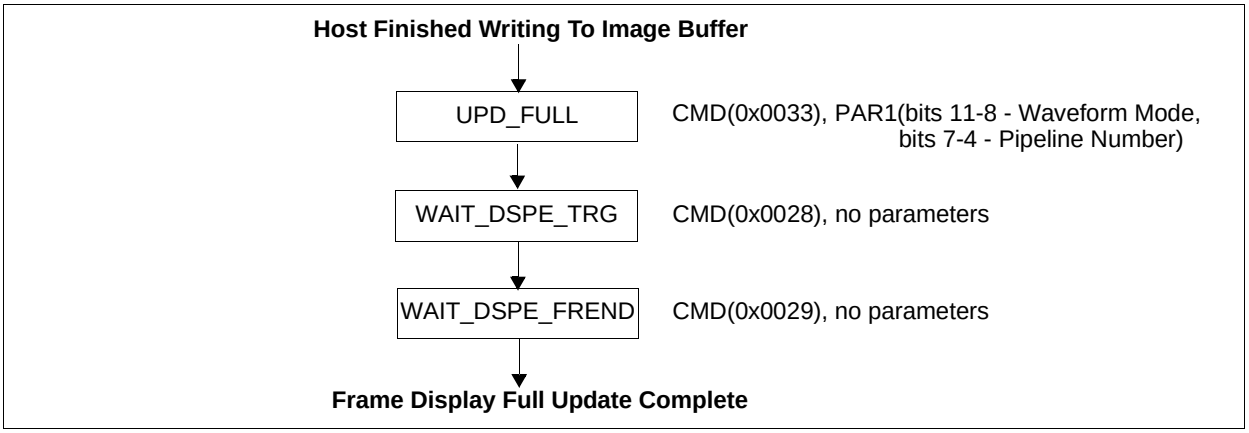


Figure 12-11: Display Engine - Full Display Update - Full Image Size Sweep Programming Flow

12.1.11 Display Engine - Full Display Update - User Defined Area Sweep

This operation forces a display update for every pixel within the user defined area, regardless of whether the pixel data has changed.

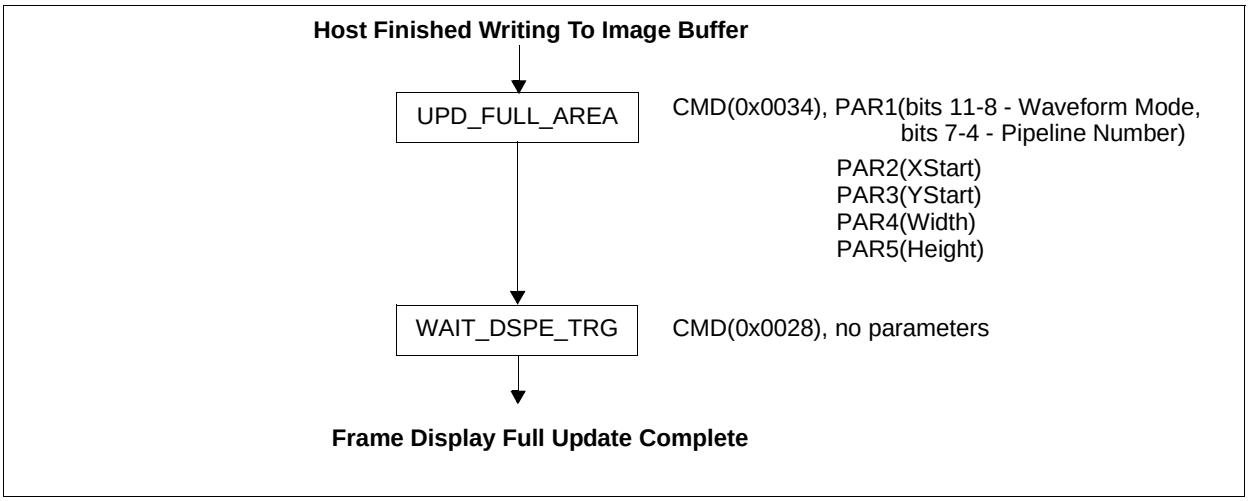


Figure 12-12: Display Engine - Full Display Update - User Defined Area Sweep Programming Flow

12.1.12 Display Engine - Partial Display Update - Full Image Size Sweep

This operation performs a display update for every pixel with changed pixel data within the defined display size.

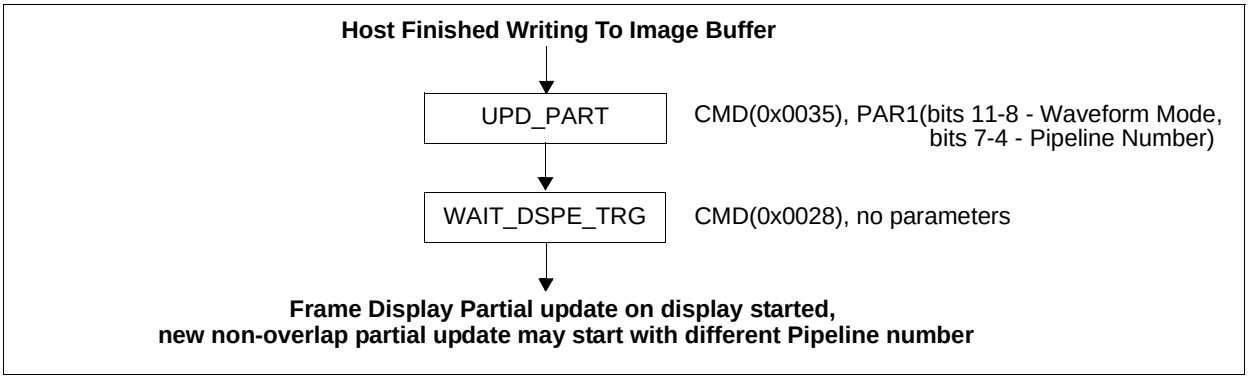


Figure 12-13: Display Engine - Partial Display Update - Full Image Size Sweep Programming Flow

12.1.13 Display Engine - Partial Display Update - User Defined Area Sweep

This operation performs a display update for every pixel with changed pixel data within the user defined area.

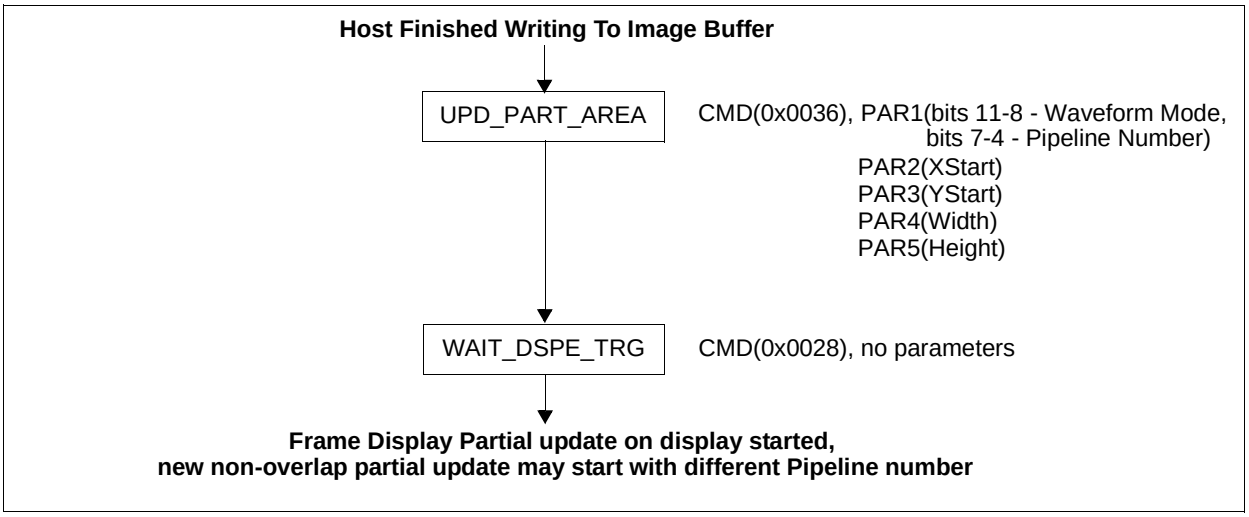


Figure 12-14: Display Engine - Partial Display Update - User Defined Area Sweep Programming Flow

12.1.14 Display Engine - Picture-In-Picture Setup

This Setups and enable/disable the Picture in Picture (PIP) feature.

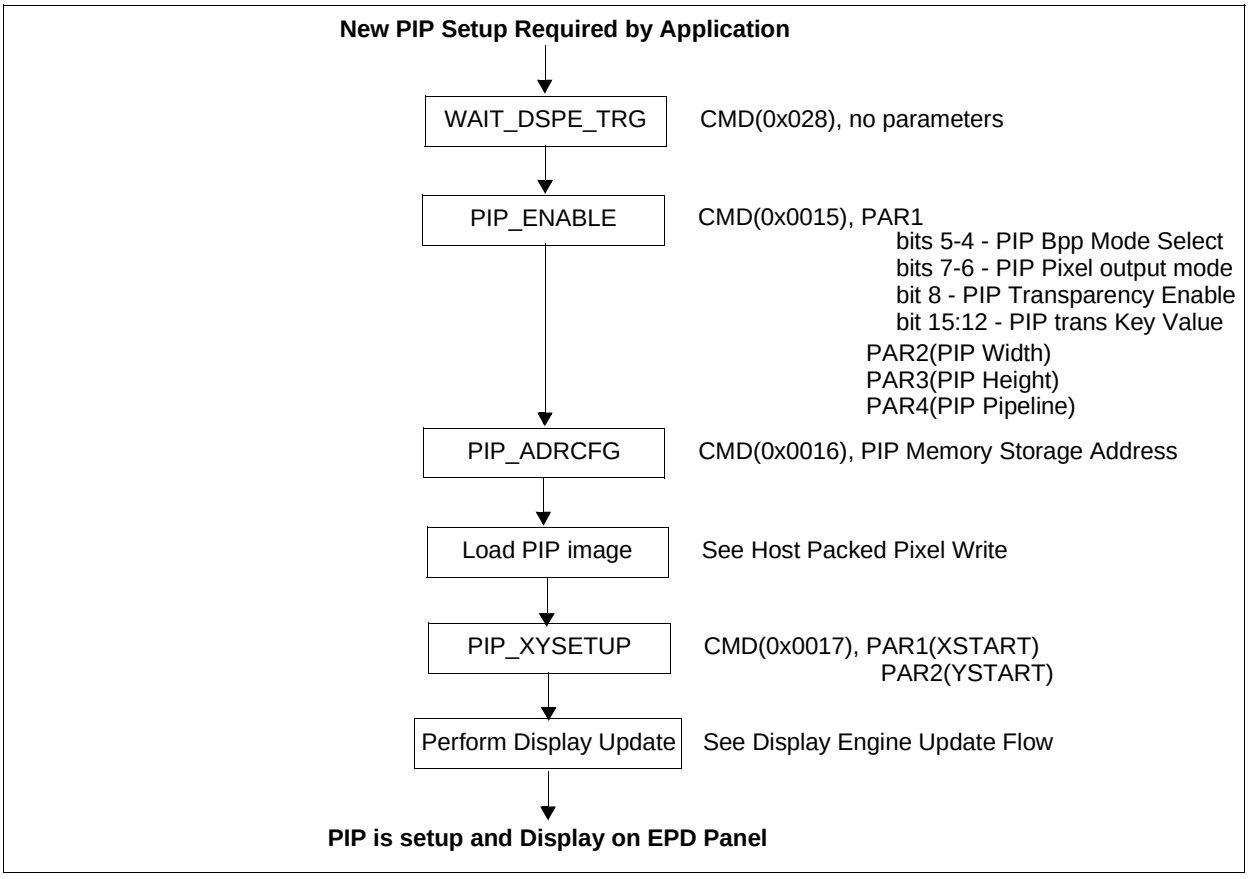


Figure 12-15: Display Engine - Picture-In-Picture Setup

12.1.15 Display Engine - Picture-In-Picture Position Change

This change the position of the Picture in Picture (PIP) on the EPD panel.

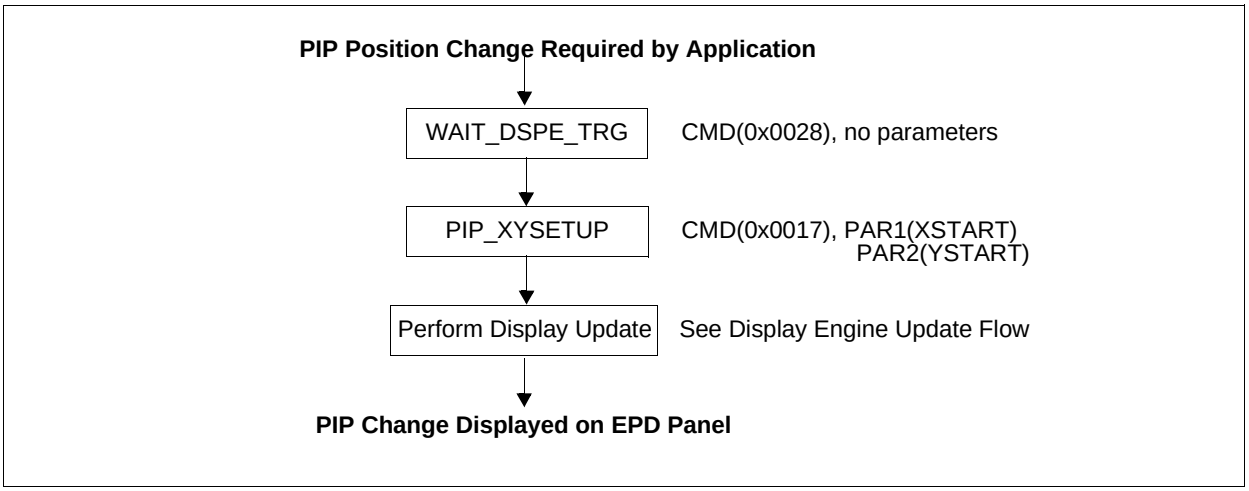


Figure 12-16: Display Engine - Picture-In-Picture Position Change

12.1.16 Display Engine - Cursor Setup

This Setups and enable/disable the Cursor feature.

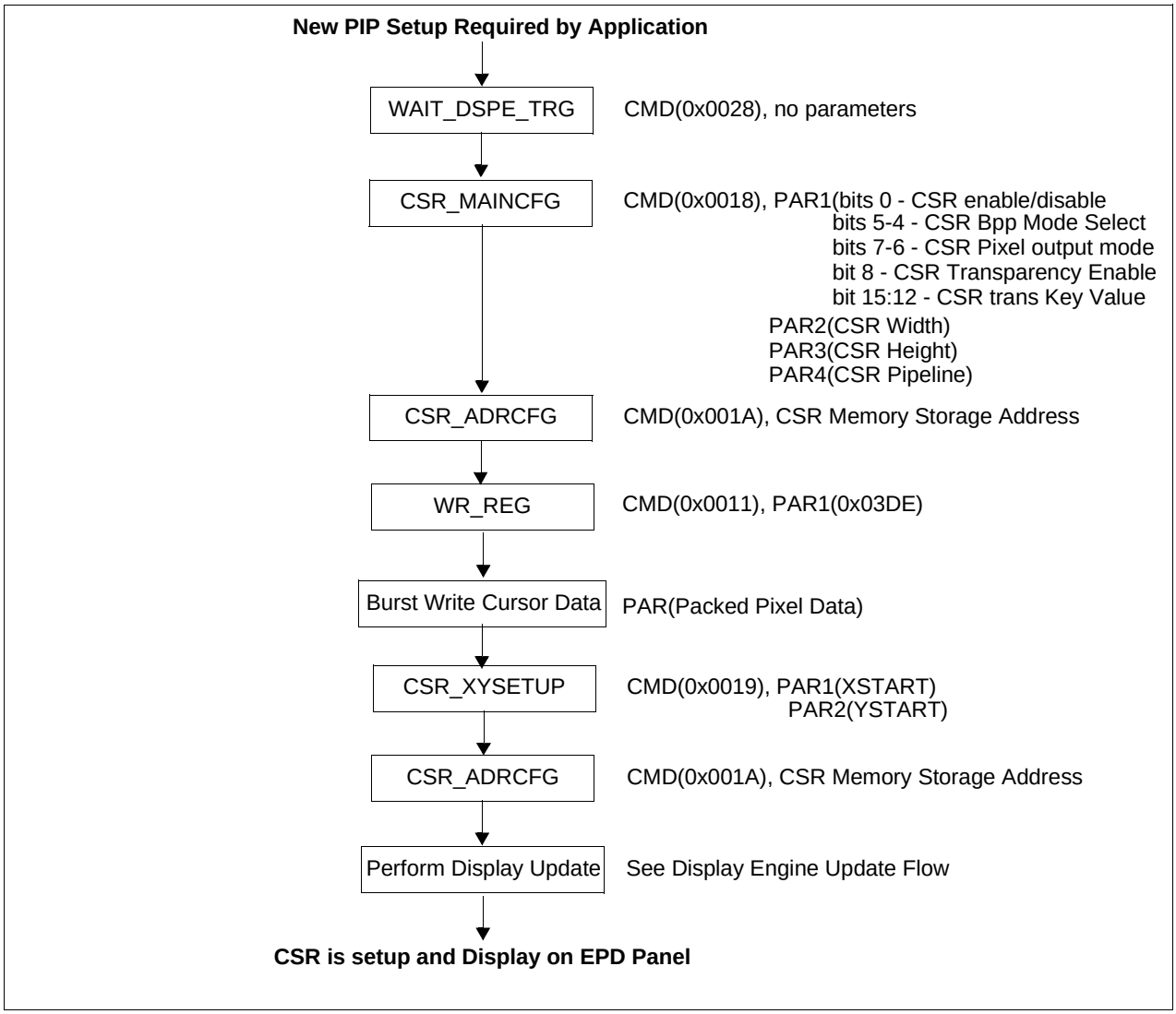


Figure 12-17: Display Engine - Cursor Setup

12.1.17 Display Engine - Cursor Position Change

This change the position of the Cursor On the EPD panel

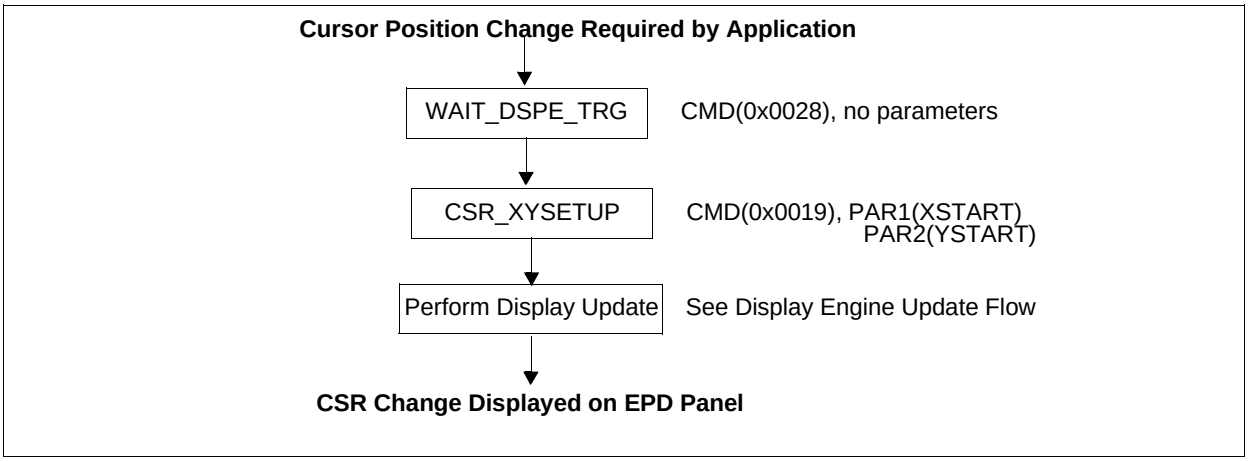


Figure 12-18: Display Engine - Cursor Position Change

Chapter 13 Auto Waveform

13.1 Introduction

E Ink has developed several waveforms which are optimized to efficiently display different grey-tones in the shortest possible time. Auto Waveform will automatically select the fastest waveform to use based on pixel transition required.

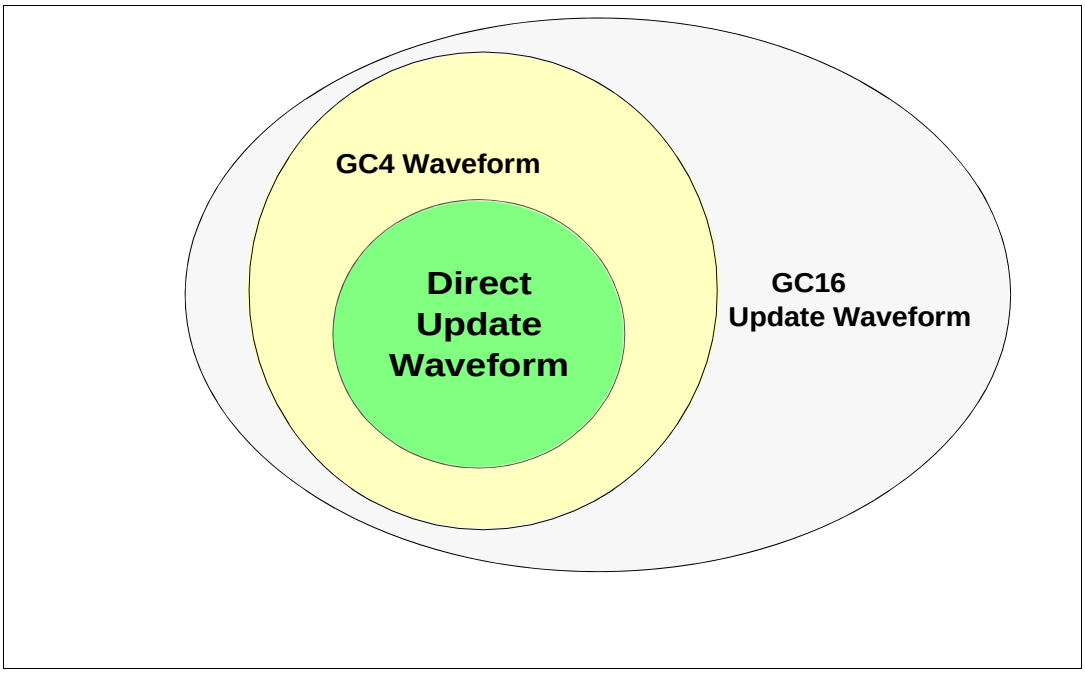


Figure 13-1: Waveform Example Sets

In the above example, the GC16 Waveforms can be used for different grey-toned pixel updates and require approximately 780 ms for the pixel update to complete. The GC4 Waveform is able to perform a pixel transition for 4 grey-toned pixel update at around 500ms. The Direct Update Waveform will convert any grey-toned pixel to a black or white level in approximately 260 ms.

13.2 Waveform Modes

E Ink waveforms are optimized for each specific panel model. For details on the specific waveforms for each panel, contact your E Ink representative.

The following table lists some standard waveform modes that are currently available from E Ink.

Note
The waveform specification is only available to end customers who have entered a non-disclosure agreement with E Ink. Contact E Ink for accurate information on Target Update Times.

Table 13-1: 3-Bit Typical Performance

Mode	Mode	Type	Gray Levels	Transition Appearance	Ghosting	Usage	Approximate Target Update Time
INIT	0	Global	White only	High Flash	n/a	Display Initialization	seconds
DU	1	Local	8 → Black and White	No Flash	Low	Monochrome menu, text input, touch screen/pen input	< 0.5 sec
GU	2	Local	8	Low Flash	Medium	High quality images and anti-aliased text	< 1 sec
GC	3	Global	8	High Flash	Low	High quality images and anti-aliased text	< 1 sec

Table 13-2: 4-Bit Typical Performance

Mode	Mode	Type	Gray Levels	Transition Appearance	Ghosting	Usage	Approximate Target Update Time
INIT	0	Global	White only	High Flash	n/a	Display Initialization	seconds
DU	1	Local	16 → Black and White	No Flash	Low	Monochrome menu, text input, touch screen/pen input	< 0.5 sec
GC16	2	Global	16	High Flash	Low	High quality images and anti-aliased text	< 1 sec
GC4	3	Global	4 → 4 16 → 4	High Flash	Low	4 gray-scaled images and anti-aliased text	< 1 sec

Chapter 14 Timing Requirements

14.1 Timing Requirements For Reset

The Command Interface automatically accesses the serial flash interface when Reset is deasserted. Therefore, the S4E5B001B00A00 must have a stable clock before Reset is deasserted.

The following Reset timing is required.

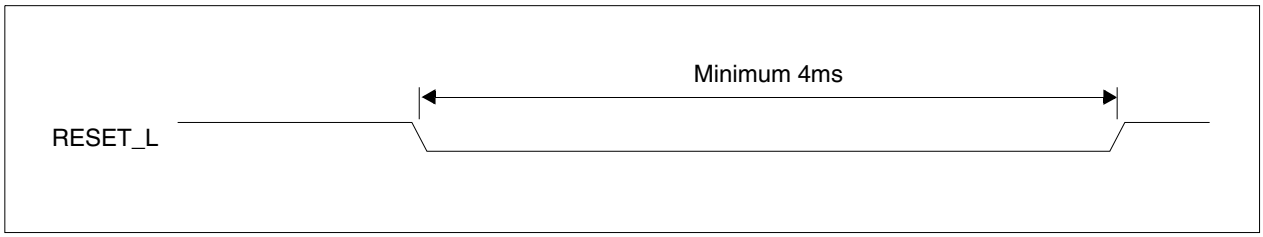


Figure 14-1: Reset Timing Requirement for Input Clock From Oscillator

The following programming flow is recommended.

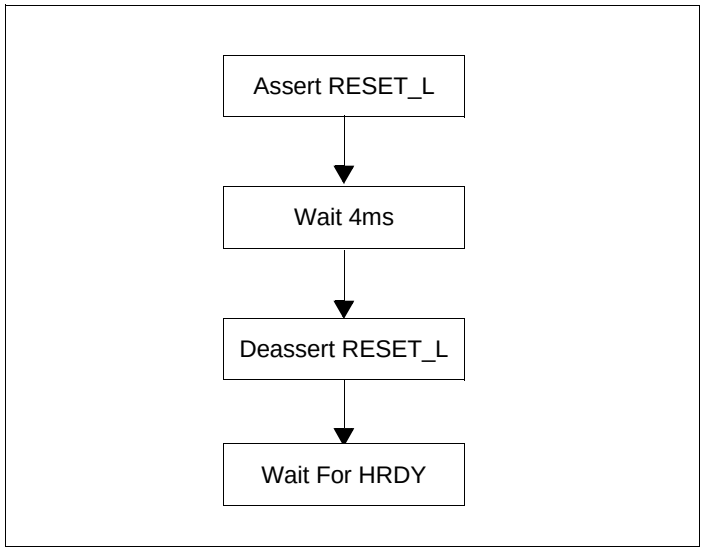


Figure 14-2: Reset Timing Requirement Programming Flow from Power-Up

14.2 Timing Requirements for Initialization or Exiting from Sleep Mode

When the S4E5B001B00A00 is initialized or exits sleep mode, it requires a stable clock before any new cycles can be executed.

The following timing is required.

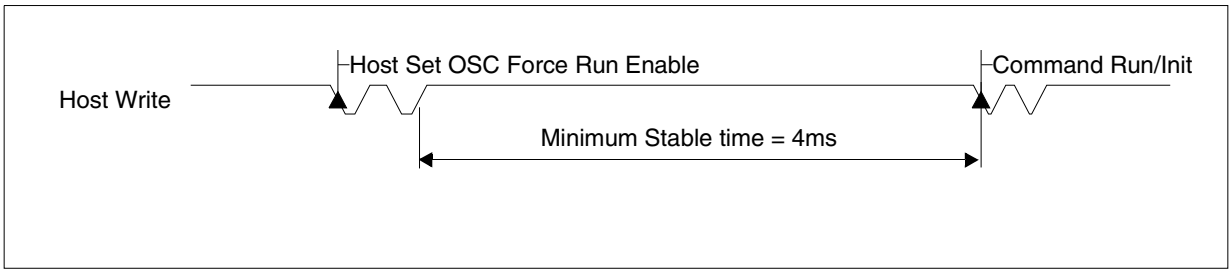


Figure 14-3: Initializing or Exiting Sleep Timing Requirement for Input Clock From Oscillator

The following programming flow is recommended.

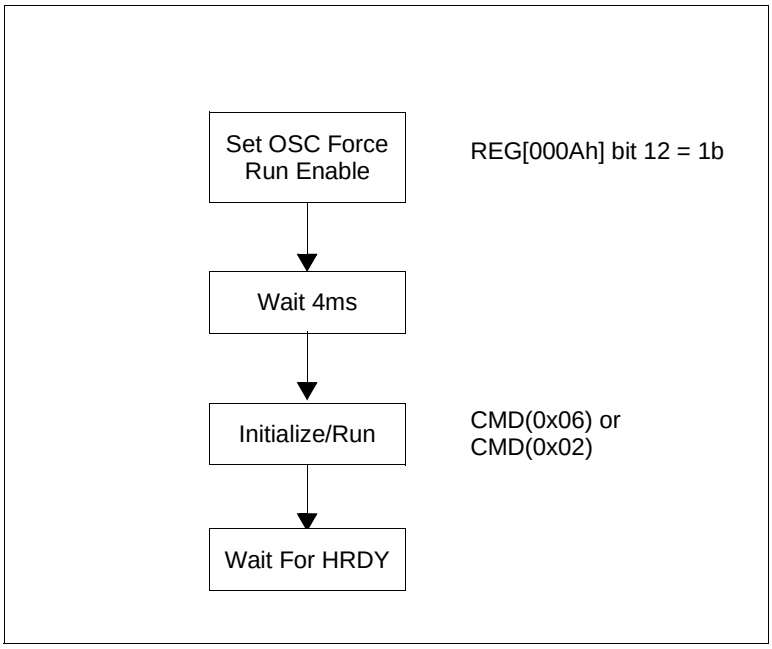


Figure 14-4: Initializing or Exiting Sleep Programming Flow for Input Clock From Oscillator

Chapter 15 Mechanical Data

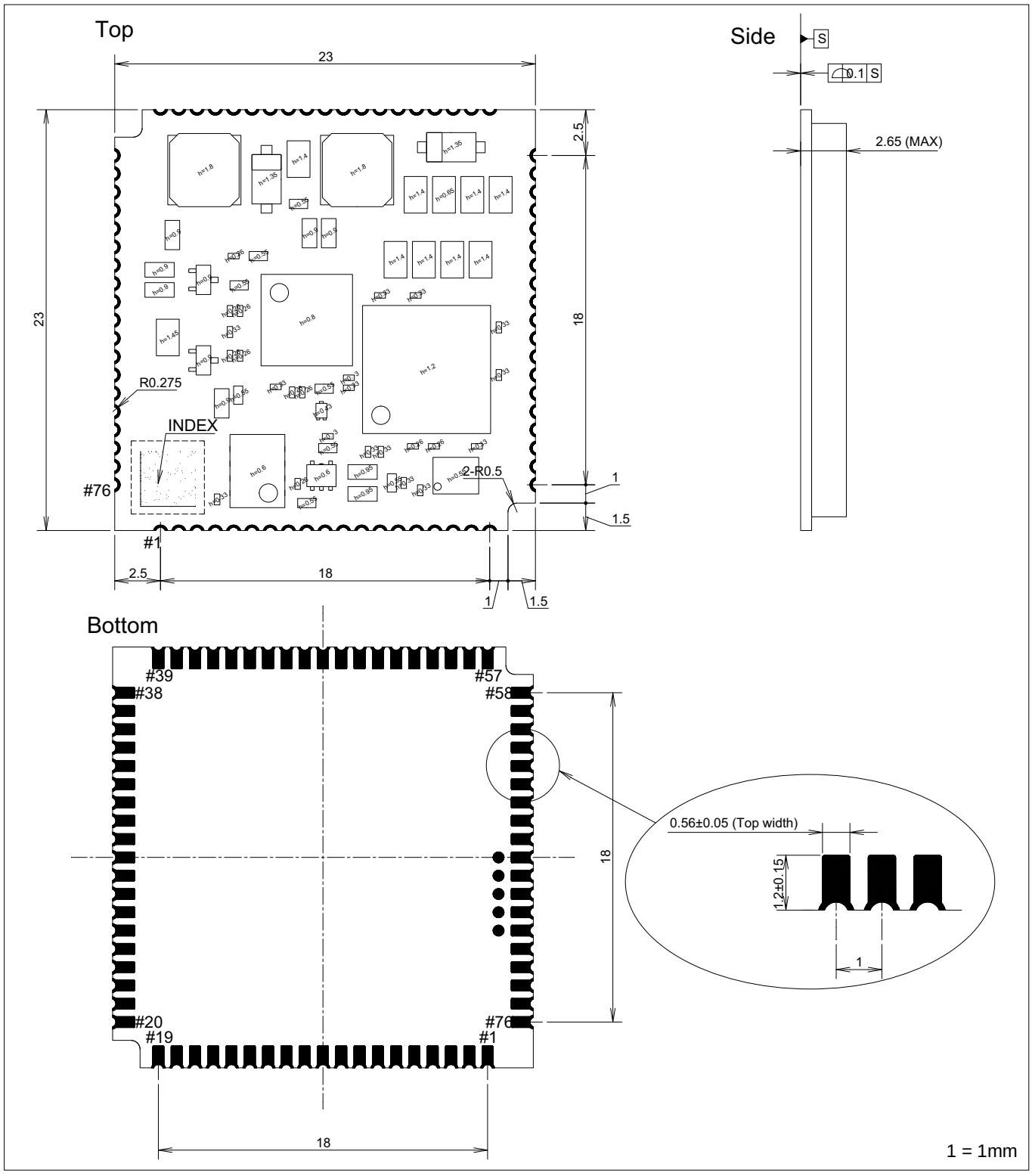


Figure 15-1: 76-pin Castellated-via Package

Chapter 16 Soldering, Handling and Storage Recommendations

The S4E5B001B00A00 EPD Controller Module is not intended for use in systems used for life support or similar devices where the failure or malfunction of the product can be reasonably expected to significantly affect the health or safety of users.

16.1 Recommended Reflow Profile

This module should be assembled with IR reflow, Full convection, or IR/Convection.

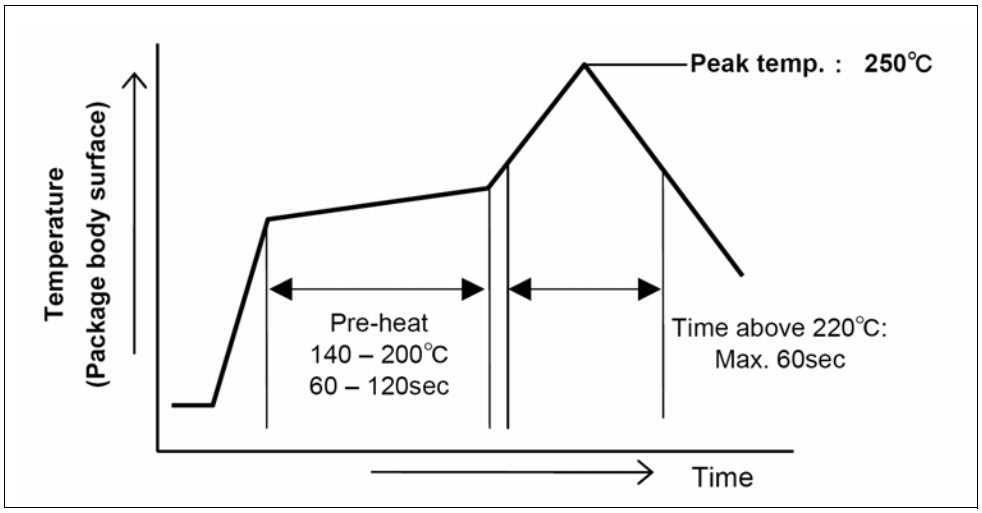


Figure 16-1: Recommended Reflow Profile

16.1.1 Repeat Reflow Soldering

The maximum allowable reflow soldering processes allowed for S4E5B001B00A00 EPD Controller Module is two times.

16.1.2 Baking Conditions

When packages exceed the storage conditions after opening the dry pack, please bake them according to the following conditions:

Table 16-1: Recommended Baking Conditions

Temperature	Time	Number of Times
125 ±5°C	20 ~ 36 hours	2

Storage conditions from the baking to the reflow soldering are the same as Chapter 16.2.2, “Storage” on page 128.

16.1.3 Hand Soldering

Hand soldering is possible.

Table 16-2: Recommended Hand Soldering Conditions

Temperature	Time	Number of Times
≤350°C	≤5 seconds	1 time / lead

16.1.4 Pb-Free Soldering Paste

Use of “No Clean” soldering paste is strongly recommended, as it does not require cleaning after the soldering process.

16.1.5 Cleaning

In general, cleaning the S4E5B001B00A00 EPD Controller Module is not recommended. Residuals under the module cannot be easily removed with any cleaning process.

- Cleaning with water can lead to capillary effects where water is absorbed into the space between the host board and the module. The combination of soldering flux residuals and encapsulated water could lead to short circuits between neighboring pads. Water could also damage any stickers or labels.
- Cleaning with alcohol or a similar organic solvent will likely flood soldering flux residuals under the module, which is not accessible for post-washing inspection. The solvent could also damage any stickers or labels.
- Ultrasonic cleaning could damage the module permanently.

The best approach is to consider using a “no clean” soldering paste and eliminate the post-soldering cleaning step.

16.1.6 Visual Inspection

The Visual Inspection Quality Level is ANSI/ASQC Z1.4 Normal Inspection Level AQL= 0.65%. After soldering the S4E5B001B00A00 EPD Controller Module to the host board, consider visual inspection to check the following:

- Proper alignment and centering of the module over the pads.
- Proper solder joints on all pads.
- Excessive solder or contacts to neighboring pads or vias.

Table 16-3: Visual Inspection

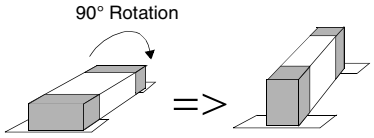
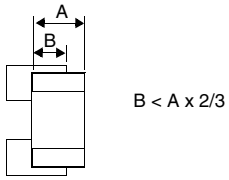
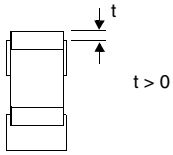
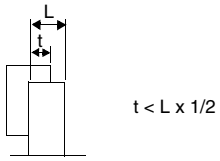
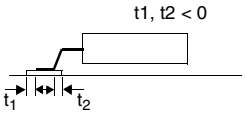
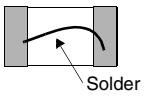
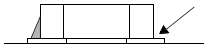
Defect	Description of Defect
Device	• Wrong part number
Height of Device	• The installed height of the device exceeds specification
Orientation of Component	• The device pin orientation is incorrect
Rotation of Component	• The device is rotated 90° on the solder pad 
Device Mounting	• X-Direction: ½ or more of the pin is not located on the land  • Y-Direction: any of the pin is not located on the land  • Leaded Parts (X-Direction): ½ or more of the pin width is not located on the land  • Leaded Parts (Y-Direction): any of the lead is not located on the land 
Open / Short	• Open or shorted pin

Table 16-3: Visual Inspection

Defect	Description of Defect
Solder Bridge	• Solder is bridging between pins  • Bonding of solder within the exclusion area • Solder flux in the land area is okay.
Solder Fillet	• Solder fillet is not formed 
Solder Ball	• Solder is bonded between pins • Solder balls of 0.1mm diameter or more are bonding • If the solder ball is bonded and less than 0.1mm it is okay (within spec)
Foreign Material on Solder Pad	• Foreign material is thicker than the solder mask thickness

16.1.7 Rework

The S4E5B001B00A00 EPD Controller Module can be unsoldered from the host board. Use of a hot air rework tool and hot plate for pre-heating from underneath is recommended. Avoid overheating.

Note

1. Do not re-solder a module which has previously been removed. Such action will terminate warranty coverage.
2. Never attempt a rework on the S4E5B001B00A00 EPD Controller Module itself, e.g. replacing individual components. Such actions will terminate warranty coverage.

16.2 Handling and Storage

16.2.1 Handling

The S4E5B001B00A00 EPD Controller Modules are designed and packaged to be processed in an automated assembly line. When mounting the S4E5B001B00A00 on a pcb, it is recommended to pick the part up by the semiconductor near the center of the module.

Note
The S4E5B001B00A00 EPD Controller Modules contain highly sensitive electronic circuitry. Handling without proper ESD protection may destroy or damage the module permanently.

16.2.2 Storage

Table 16-4: Recommended Storage Conditions

	Conditions	Time
Before opening pack	≤30°C 85% RH	1 year
After opening pack	125 ±5°C	20 ~ 36 hours
After baking	≤30°C 60% RH	7 days (168 hours)

Chapter 17 References

The following documents contain additional information related to the S4E5B001B00A00. Document numbers are listed in parenthesis after the document name. All documents can be found at the Epson Research and Development Website at **www.erd.epson.com**.

- S4E5B001B00A00 Product Brief (X93A-C-002-xx)

Appendix A Registers

The S4E5B001B00A00 has an internal register set which is used for configuration and status reporting. All necessary functionality can be controlled through the command interface discussed in Chapter 8, “Host Interface” on page 32.

This section discusses how and where to access the S4E5B001B00A00 registers. It also provides detailed information about the layout and usage of each register.

A.1 Register Access

The S4E5B001B00A00 registers are 16-bit wide. They can be accessed using the Command Interface

A.1.1 Register Access using Command Interface

Registers may be accessed using the command interface with the following sequences.

For single 16-bit register reads:

Table A-1: Register Read Sequence

Tx Count	16-bit Indirect Host
0	Write Code[15:0] <= 0x0010
1	Write Register Address[15:0]
2	Read

For single 16-bit register writes:

Table A-2: Register Write Sequence

Tx Count	16-bit Indirect Host
0	Write Code[15:0] <= 0x0011
1	Write Register Address[15:0]
2	Write

Note
Only single register accesses are supported (i.e. register address is not auto incremented to the next register address).

A.2 Register Set

The S4E5B001B00A00 registers are listed in the following table.

Table A-3: S4E5B001B00A00 Register Set

Register	Pg	Register	Pg
REG[0002h] Product Code Register	132	REG[000Ah] System Status Register	132
REG[0154h] Host Memory Access Port Register	134	REG[015Ah] Host Packed Write Transparency Register	134
REG[015Eh] Host Memory Access Pixel Swap Configuration Register	135	REG[0240h] Interrupt Raw Status Register	136
REG[0242h] Interrupt Masked Status Register	137	REG[0244h] Interrupt Control Register	138
REG[0250h] GPIO Configuration Register	139	REG[0252h] GPIO Status/Control Register	139
REG[0254h] GPIO Interrupt Enable Register	140	REG[0256h] GPIO Interrupt Status Register	141
REG[02AEh] Command Interrupt Control	141	REG[033Ah] Display Engine Interrupt Raw Status Register	142
REG[033Ch] Display Engine Interrupt Masked Status Register	143	REG[033Eh] Display Engine Interrupt Enable Register	144
REG[03DEh] Cursor Memory Access Port Register	144		

Note
All other register addresses should not be written.

A.3 Register Descriptions

REG[0002h] Product Code Register Default = 004Dh							Read Only
Product Code bits 15-8							
15	14	13	12	11	10	9	8
Product Code bits 7-0							
7	6	5	4	3	2	1	0

bits 15-0 Product Code bits [15:0] (Read Only)
These bits indicate the product code.
The product code for the S4E5B001B00A00 is 004Dh.

REG[000Ah] System Status Register Default = 0000h							Read/Write
Reserved	Power Management Busy Status (RO)	n/a	OSC Cell Force Run Enable	Power Save Status bits 1-0		Memory Self Refresh Mode Status (RO)	Power Pin Sequence Status (RO)
15	14	13	12	11	10	9	8
I2C Busy Status (RO)	SPI Busy Status (RO)	Host Interface Busy Status (RO)	Memory Controller Busy Status (RO)	Host Memory Access Busy Status (RO)	Display Engine Busy Status (RO)	Memory Initialized (RO)	PLL Lock (RO)
7	6	5	4	3	2	1	0

bit 15 Reserved
The default value for this bit is 0b.

bit 14 Power Management Busy Status (Read Only)
This bit indicates the status of the Power Management logic.
When this bit = 0b, the Power Management logic is not busy.
When this bit = 1b, the Power Management logic is busy.

bit 12 OSC Cell Force Run Enable
This bit configures the OSC Cell to always run even on power save mode with PLL off.

bits 11-10 Power Save Status bits [1:0]
These bits indicate the power save status of the S4E5B001B00A00.

Table A-4: Power Save Status

REG[000Ah] bits 11-10	Power Save Status
00b	Un-initialized System
01b	Run Mode
10b	Standby Mode
11b	Sleep Mode

bit 9 Memory Self Refresh Mode Status (Read Only)
This bit indicates whether the memory controller is in self refresh mode. Self refresh mode is controlled using the Memory Enter Self Refresh Trigger bit (REG[0104h] bit 0) and the Exit Self Refresh Trigger bit (REG[0104h] bit 1).
When this bit = 0b, the memory controller is not in self refresh mode.
When this bit = 1b, the memory controller is in self refresh mode.

Note
The memory must not be accessed while it is in self refresh mode. To read or write to memory, self refresh mode must be exited, REG[0104h] bit 1 = 1b.

bit 8 Power Pin Sequence Status (Read Only)
This bit provides the status of the pin sequence.
When this bit = 0b, it is currently in a power off state. (default)
When this bit = 1b, it is currently in a power on state.

bit 7 I2C Busy Status (Read Only)
This bit indicates whether the I2C interface is busy.
When this bit = 0b, the I2C interface is idle (not busy).
When this bit = 1b, the I2C interface is busy.

bit 6 SPI Busy Status (Read Only)
This bit indicates whether the SPI interface is busy.
When this bit = 0b, the SPI interface is idle (not busy).
When this bit = 1b, the SPI interface is busy.

bit 5 Host Command Interface Busy Status (Read Only)
This bit indicates whether the Host Command interface is busy.
When this bit = 0b, the Host Command interface is idle (not busy).
When this bit = 1b, the Host Command interface is busy.

bit 4 Memory Controller Busy Status (Read Only)
This bit indicates whether the memory controller is busy.
When this bit = 0b, the memory controller is idle (not busy).
When this bit = 1b, the memory controller is busy.

bit 3 Host Memory Access Busy Status (Read Only)
This bit indicates the status of the current Host Memory Access. For write accesses, this bit remains at 1b until the memory operation is complete. For read accesses, this bit remains at 1b until the read data has been transferred into the FIFO.
When this bit = 0b, a Host Memory Access is not taking place (not busy).
When this bit = 1b, a Host Memory Access is taking place (busy).

bit 2 Display Engine Busy Status (Read Only)
This bit indicates whether the Display Engine is busy.
When this bit = 0b, the Display Engine is idle (not busy).
When this bit = 1b, the Display Engine is busy.

bit 1 Memory Initialized (Read Only)
This bit indicates whether the memory has been initialized. The memory initialization sequence is triggered by the Memory Initialization Trigger bit, REG[0102h] bit 0.
When this bit = 0b, the memory has not been initialized.
When this bit = 1b, the memory has been initialized.

bit 0 PLL Lock (Read Only)
This bit indicates whether the PLL output is stable (locked). The S4E5B001B00A00 synchronous registers and the internal memory must not be accessed before the PLL output is stable.
When this bit = 0b, the PLL output is not stable.
When this bit = 1b, the PLL output is stable.

REG[0154h] Host Memory Access Port Register
Default = 0000h — Read/Write

Host Memory Access Port bits 15-8							
15	14	13	12	11	10	9	8
Host Memory Access Port bits 7-0							
7	6	5	4	3	2	1	0

bits 15-0 Host Memory Access Port bits [15:0]
These bits are the data port for host memory interface accesses.

REG[015Ah] Host Packed Write Transparency Register
Default = 0000h — Read/Write

n/a							
15	14	13	12	11	10	9	8
Transparency Key Value bits 7-4				n/a			
7	6	5	4	3	2	1	0

bits 7-4 Transparency Key Value bits [7:4]
These bits configure the transparency key value used for host packed write operations to memory. When the host writes a pixel value equivalent to these bits, the pixel will not be written to memory. For 1 and 2 bpp modes, only the most significant 1 or 2 bits are used.

REG[015Eh] Host Memory Access Pixel Swap Configuration Register

Default = 0000h — Read/Write

15	14	13	12	11	10	9	8
n/a							
7	6	5	4	3	2	1	0
n/a						Packed Pixel Mode Select bits 1-0	

bits 1-0 Packed Pixel Mode Select bits [1:0]

These bits select the pixel order/byte order of the host memory access operations.
When bit 1 = 0b, byte order in memory is not reversed.
When bit 1 = 1b, byte order is swapped when writing to or reading from memory.
When bit 0 = 0b, pixel order in memory is not reversed.
When bit 0 = 1b, the pixel order is reversed when writing to or reading from to memory.

Table A-5: Packed Pixel Mode Selection

LD_IMG / LD_IMG_AREA Parameter 1 bits 5-4	Memory Access Mode	Host Memory Data Format			
		REG[015Eh] bits 1-0 = 00b	REG[015Eh] bits 1-0 = 01b	REG[015Eh] bits 1-0 = 10b	REG[015Eh] bits 1-0 = 11b
00b	1 bpp packed	{P15,P14,P13,P12,P11,P10,P9,P8,P7,P6,P5,P4,P3,P2,P1,P0}	{P0,P1,P2,P3,P4,P5,P6,P7,P8,P9,P10,P11,P12,P13,P14,P15}	{P7,P6,P5,P4,P3,P2,P1,P0,P15,P14,P13,P12,P11,P10,P9,P8}	{P8,P9,P10,P11,P12,P13,P14,P15,P0,P1,P2,P3,P4,P5,P6,P7}
01b	2 bpp packed	{P7,P6,P5,P4,P3,P2,P1,P0}	{P0,P1,P2,P3,P4,P5,P6,P7}	{P3,P2,P1,P0,P7,P6,P5,P4}	{P4,P5,P6,P7,P0,P1,P2,P3}
10b	4 bpp packed (Host Raw Access)	{P3,P2,P1,P0}	{P0,P1,P2,P3}	{P1,P0,P3,P2}	{P2,P3,P0,P1}
11b	8 bpp packed	{P1,P0}	{P0,P1}	{P0,P1}	{P1,P0}

Note

Host Raw Access is treated the same as 4bpp.

REG[0240h] Interrupt Raw Status Register

Default = 0000h — Read/Write

15	14	13	12	11	10	9	8
n/a				Command Interrupt Raw Status	Reserved		
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved				GPIO Interrupt Raw Status	Reserved	Display Engine Interrupt Raw Status	Reserved

bit 11 — Command Interrupt Raw Status

This bit indicates the raw status of the Command Interrupt and is not masked by the Command Interrupt Enable bit, REG[0244h] bit 11. This interrupt is manually controlled by the Command Interrupt Control bit, REG[02AEh] bit 0.
When this bit = 0b, a Command Interrupt has not occurred.
When this bit = 1b, a Command Interrupt has occurred.

To clear this status bit, write a 0b to the Command Interrupt Control bit, REG[02AEh] bit 0.

bits 10 - 4 — Reserved

The default value of these bits is 0000000b

bit 3 — GPIO Interrupt Raw Status

This bit indicates the raw status of the GPIO Interrupt and is not masked by the GPIO Interrupt Enable bit, REG[0244h] bit 3. For configuration and status of each individual GPIO, see REG[0254h] and REG[0256h].
When this bit = 0b, a GPIO Interrupt has not occurred.
When this bit = 1b, a GPIO Interrupt has occurred.

To clear this status bit, write a 1b to the appropriate GPIO[5:0] Negative/Positive Interrupt Status bit in REG[0256h].

bit 2 — Reserved

The default value of this bit is 0b.

bit 1 — Display Engine Interrupt Raw Status

This bit indicates the raw status of the Display Engine Interrupt which occurs when one of the interrupts in REG[033Ah] or REG[033Ch] is triggered. This bit is not masked by the Display Engine Interrupt Enable bit, REG[0244h] bit 1.
When this bit = 0b, a Display Engine Interrupt has not occurred.
When this bit = 1b, a Display Engine Interrupt has occurred.

To clear this status bit, clear the triggering interrupt in REG[033Ah] or REG[033Ch].

bit 0 — Reserved

The default value of this bit is 0b.

REG[0242h] Interrupt Masked Status Register

Default = 0000h — Read/Write

15	14	13	12	11	10	9	8
n/a				Command Interrupt Masked Status	Reserved		
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved				GPIO Interrupt Masked Status	Reserved	Display Engine Interrupt Masked Status	Reserved

bit 11 — Command Masked Status

This bit indicates the masked status of the Command Interrupt (see REG[0244h] bit 11). This interrupt is manually controlled by the Command Interrupt Control bit, REG[02AEh] bit 0.
When this bit = 0b, a Command Interrupt has not occurred.
When this bit = 1b, a Command Interrupt has occurred.

To clear this status bit, write a 0b to the Command Interrupt Control bit, REG[02AEh] bit 0.

bits 10 - 4 — Reserved

The default value of these bits is 0000000b

bit 3 — GPIO Interrupt Masked Status

This bit indicates the masked status of the GPIO Interrupt (see REG[0244h] bit 3). For configuration and status of each individual GPIO, see REG[0254h] and REG[0256h].
When this bit = 0b, a GPIO Interrupt has not occurred.
When this bit = 1b, a GPIO Interrupt has occurred.

To clear this status bit, write a 1b to the appropriate GPIO[5:0] Negative/Positive Interrupt Status bit in REG[0256h].

bit 2 — Reserved

The default value of this bit is 0b.

bit 1 — Display Engine Interrupt Masked Status

This bit indicates the masked status of the Display Engine Interrupt (see REG[0244h] bit 1) which occurs when one of the interrupts in REG[033Ah] or REG[033Ch] is triggered.
When this bit = 0b, a Display Engine Interrupt has not occurred.
When this bit = 1b, a Display Engine Interrupt has occurred.

To clear this status bit, clear the triggering interrupt in REG[033Ah] or REG[033Ch].

bit 0 — Reserved

The default value of this bit is 0b.

REG[0244h] Interrupt Control Register

Default = 0000h — Read/Write

n/a				Command Interrupt Enable	Reserved		
15	14	13	12	11	10	9	8
Reserved				GPIO Interrupt Enable	Reserved	Display Engine Interrupt Enable	Reserved
7	6	5	4	3	2	1	0

bit 11 — Command Interrupt Enable
This bit controls whether the Command Interrupt causes an interrupt request on the HIRQ pin. The Command Interrupt is manually controlled using the Command Interrupt Control bit, REG[02AEh] bit 0.
When this bit = 0b, the Command Interrupt is disabled.
When this bit = 1b, the Command Interrupt is enabled.

bits 10 - 4 — Reserved
The default value of these bits is 0000000b

bit 3 — GPIO Interrupt Enable
This bit controls whether the GPIO Interrupt causes an interrupt request on the HIRQ pin. The status of this interrupt can be determined by reading REG[0240h] bit 3 (unmasked) or REG[0242h] bit 3 (masked).
When this bit = 0b, the interrupt is disabled.
When this bit = 1b, the interrupt is enabled.

bit 2 — Reserved
The default value of this bit is 0b.

bit 1 — Display Engine Interrupt Enable
This bit controls whether the Display Engine Interrupt causes an interrupt request on the HIRQ pin. The status of this interrupt can be determined by reading REG[0240h] bit 1 (unmasked) or REG[0242h] bit 1 (masked).
When this bit = 0b, the interrupt is disabled.
When this bit = 1b, the interrupt is enabled.

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

REG[0250h] GPIO Configuration Register

Default = 0000h — Read/Write

15	14	13	12	11	10	9	8
n/a		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
n/a		GPIO5 Configuration	GPIO4 Configuration	GPIO3 Configuration	GPIO2 Configuration	GPIO1 Configuration	GPIO0 Configuration

bits 13-8 GPIO[5:0] 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 the bit = 0b, the pull-down resistor for the corresponding GPIOx pin is inactive. (default)

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

bits 5-0 GPIO[5:0] 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 input pin. (default)

When this bit = 1b, the corresponding GPIO pin is configured as an output pin.

Note

The GPIOs, when used as inputs, are not debounced.

REG[0252h] GPIO Status/Control Register

Default = XX00h — Read/Write

15	14	13	12	11	10	9	8
n/a		GPIO5 Input Status (RO)	GPIO4 Input Status (RO)	GPIO3 Input Status (RO)	GPIO2 Input Status (RO)	GPIO1 Input Status (RO)	GPIO0 Input Status (RO)
7	6	5	4	3	2	1	0
n/a		GPIO5 Data Output Control	GPIO4 Data Output Control	GPIO3 Data Output Control	GPIO2 Data Output Control	GPIO1 Data Output Control	GPIO0 Data Output Control

bits 13-8 GPIO[5:0] Input Status (Read Only)

When GPIOx is configured as an input (see REG[0250h] bits 5-0), a read from this bit returns the state of the corresponding GPIOx pin.

When this bit = 0b, the GPIOx pin is 0 (low).

When this bit = 1b, the GPIOx pin is 1 (high).

Note

The GPIOs, when used as inputs, are not debounced.

bits 5-0 GPIO[5:0] Data Output Control

When GPIOx is configured as an output (see REG[0250h] bits 5-0), a write to this bit drives the output state of the corresponding GPIOx pin.

When this bit = 0b, the corresponding GPIOx pin is driven to 0 (low). (default)

When this bit = 1b, the corresponding GPIOx pin is driven to 1 (high).

REG[0254h] GPIO Interrupt Enable Register

Default = 0000h — Read/Write

15	14	13	12	11	10	9	8
n/a		GPIO5 Negative Edge Interrupt Enable	GPIO4 Negative Edge Interrupt Enable	GPIO3 Negative Edge Interrupt Enable	GPIO2 Negative Edge Interrupt Enable	GPIO1 Negative Edge Interrupt Enable	GPIO0 Negative Edge Interrupt Enable
7	6	5	4	3	2	1	0
n/a		GPIO5 Positive Edge Interrupt Enable	GPIO4 Positive Edge Interrupt Enable	GPIO3 Positive Edge Interrupt Enable	GPIO2 Positive Edge Interrupt Enable	GPIO1 Positive Edge Interrupt Enable	GPIO0 Positive Edge Interrupt Enable

Note

The GPIO interrupts should not be used with floating GPIO pins. Use the pull-downs in REG[0250h] bits [13:8] to avoid this.

bits 13-8 GPIO[5:0] Negative Edge Interrupt Enable

These bits control whether the corresponding GPIOx interrupt (see REG[0256h]) is triggered on the negative edge (when the GPIOx pin changes from 1to 0).

When this bit = 0b, the corresponding GPIOx interrupt is not triggered on the negative edge. (default)

When this bit = 1b, the corresponding GPIOx interrupt is triggered on the negative edge.

bits 5-0 GPIO[5:0] Positive Edge Interrupt Enable

These bits control whether the corresponding GPIOx interrupt (see REG[0256h]) is triggered on the positive edge (when the GPIOx pin changes from 0 to 1).

When this bit = 0b, the corresponding GPIOx interrupt is not triggered on the positive edge. (default)

When this bit = 1b, the corresponding GPIOx interrupt is triggered on the positive edge.

REG[0256h] GPIO Interrupt Status Register

Default = 0000h — Read/Write

15	14	13	12	11	10	9	8
n/a		GPIO5 Negative Edge Interrupt Status	GPIO4 Negative Edge Interrupt Status	GPIO3 Negative Edge Interrupt Status	GPIO2 Negative Edge Interrupt Status	GPIO1 Negative Edge Interrupt Status	GPIO0 Negative Edge Interrupt Status
7	6	5	4	3	2	1	0
n/a		GPIO5 Positive Edge Interrupt Status	GPIO4 Positive Edge Interrupt Status	GPIO3 Positive Edge Interrupt Status	GPIO2 Positive Edge Interrupt Status	GPIO1 Positive Edge Interrupt Status	GPIO0 Positive Edge Interrupt Status

Note

The GPIO interrupts should not be used with floating GPIO pins. Use the pull-downs in REG[0250h] bits [13:8] to avoid this.

bits 13-8 GPIO[5:0] Negative Edge Interrupt Status

These bits indicate the status of the corresponding GPIOx Negative Edge Interrupt.
When this bit = 0b, a Negative Edge Interrupt has not occurred. (default)
When this bit = 1b, a Negative Edge Interrupt has occurred.

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

bits 5-0 GPIO[5:0] Positive Edge Interrupt Status

These bits indicate the status of the corresponding GPIOx Positive Edge Interrupt.
When this bit = 0b, a Positive Edge Interrupt has not occurred. (default)
When this bit = 1b, a Positive Edge Interrupt has occurred.

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

REG[02AEh] Command Interrupt Control

Default = 0000h — Read/Write

15	14	13	12	11	10	9	8
n/a							
7	6	5	4	3	2	1	0
n/a							Command Interrupt Control

bit 0 Command Interrupt Control

This bit manually controls the Command Interrupt. The status the Command Interrupt is available in REG[0240h] bit 11 (un-masked) and REG[0242h] bit 11 (masked). The interrupt enable is set in REG[0244h] bit 11.
When this bit = 0b, the Command Interrupt is not triggered.
When this bit = 1b, the Command Interrupt is triggered.

REG[033Ah] Display Engine Interrupt Raw Status Register

Default = 0000h — Read/Write

n/a	Reserved						
15	14	13	12	11	10	9	8
Pipeline Busy Conflict Detected Interrupt Raw Status	Reserved						Operation Trigger Done Interrupt Raw Status
7	6	5	4	3	2	1	0

bits 14 - 8 — Reserved
The default value of these bits is 000_0000b

bit 7 — Pipeline Busy Conflict Detected Interrupt Raw Status
This bit indicates the raw status of the Pipeline Busy Conflict Detected Interrupt and is not masked by the Pipeline Busy Conflict Detected Interrupt Enable bit, REG[033Eh] bit 7.
This interrupt occurs during image buffer updates when a pixel update is required and the same pixel is currently being used for frame display (using another Pipeline).
When this bit = 0b, a Pipeline Busy Conflict Detected Interrupt has not occurred.
When this bit = 1b, a Pipeline Busy Conflict Detected Interrupt has occurred.

To clear this status bit, write a 1b to either this bit or REG[033Ch] bit 7.

bits 6 - 1 — Reserved
The default value of these bits is 00_0000b

bit 0 — Operation Trigger Done Interrupt Raw Status
This bit indicates the raw status of the Operation Trigger Done Interrupt and is not masked by the Operation Trigger Done Interrupt Enable bit, REG[033Eh] bit 0.
When this bit = 0b, an Operation Trigger Done Interrupt has not occurred.
When this bit = 1b, an Operation Trigger Done Interrupt has occurred.

To clear this status bit, write a 1b to either this bit or REG[033Ch] bit 0.

REG[033Ch] Display Engine Interrupt Masked Status Register
Default = 0000h — Read/Write

n/a	Reserved						
15	14	13	12	11	10	9	8
Pipeline Busy Conflict Detected Interrupt Masked Status	Reserved						Operation Trigger Done Interrupt Masked Status
7	6	5	4	3	2	1	0

bits 14 - 8 Reserved
The default value of these bits is 000_0000b

bit 7 Pipeline Busy Conflict Detected Interrupt Masked Status
This bit indicates the masked status of the Pipeline Busy Conflict Detected Interrupt (see REG[033Eh] bit 7). This interrupt occurs during image buffer updates when a pixel update is required and the same pixel is currently being used for frame display (using another Pipeline).
When this bit = 0b, a Pipeline Busy Conflict Detected Interrupt has not occurred.
When this bit = 1b, a Pipeline Busy Conflict Detected Interrupt has occurred.

To clear this status bit, write a 1b to either this bit or REG[033Ah] bit 7.

bits 6 - 1 Reserved
The default value of these bits is 00_0000b

bit 0 Operation Trigger Done Interrupt Masked Status
This bit indicates the masked status of the Operation Trigger Done Interrupt (see REG[033Eh] bit 0).
When this bit = 0b, an Operation Trigger Done Interrupt has not occurred.
When this bit = 1b, an Operation Trigger Done Interrupt has occurred.

To clear this status bit, write a 1b to either this bit or REG[033Ah] bit 0.

REG[033Eh] Display Engine Interrupt Enable Register

Default = 0000h — Read/Write

n/a	Reserved						
15	14	13	12	11	10	9	8
Pipeline Busy Conflict Detected Interrupt Enable	Reserved						Operation Trigger Done Interrupt Enable
7	6	5	4	3	2	1	0

bits 14 - 8 Reserved
The default value of these bits is 000_0000b

bit 7 Pipeline Busy Conflict Detected Interrupt Enable
This bit controls whether the Pipeline Busy Conflict Detected Interrupt triggers a Display Engine Interrupt (see REG[0240h] and REG[0242h]). The status of this interrupt can be determined by reading REG[033Ah] bit 7 (unmasked) or REG[033Ch] bit 7 (masked).
When this bit = 0b, the interrupt is disabled.
When this bit = 1b, the interrupt is enabled.

bits 6 - 1 Reserved
The default value of these bits is 00_0000b

bit 0 Operation Trigger Done Interrupt Enable
This bit controls whether the Operation Trigger Done Interrupt triggers a Display Engine Interrupt (see REG[0240h] and REG[0242h]). The status of this interrupt can be determined by reading REG[033Ah] bit 0 (unmasked) or REG[033Ch] bit 0 (masked).
When this bit = 0b, the interrupt is disabled.
When this bit = 1b, the interrupt is enabled.

REG[03DEh] Cursor Memory Access Port Register

Default = 0000h — Read/Write

Cursor Memory Access Port bits 15-8							
15	14	13	12	11	10	9	8
Cursor Memory Access Port bits 7-0							
7	6	5	4	3	2	1	0

bits 15-0 Cursor Memory Access Port bits [15:0]

Change Record

X93A-A-006-01 Revision 1.2 - Issued: February 22, 2013

- Official release of this document.

X93A-A-006-01 Revision 1.0 - Issued: September 25, 2012

- Preliminary release of this document.

EPSON

International Sales Operations

AMERICA

EPSON ELECTRONICS AMERICA, INC.
214 Devcon Drive
San Jose, CA 95112,USA
Phone: +1-800-228-3964 FAX: +1-408-922-0238

EUROPE

EPSON EUROPE ELECTRONICS GmbH
Riesstrasse 15, 80992 Munich,
GERMANY
Phone: +49-89-14005-0 FAX: +49-89-14005-110

ASIA

EPSON (CHINA) CO., LTD.
7F, Jinbao Bldg., No.89 Jinbao St.,
Dongcheng District,
Beijing 100005, CHINA
Phone: +86-10-8522-1199 FAX: +86-10-8522-1125

SHANGHAI BRANCH
7F, Block B, High-Tech Bldg., 900, Yishan Road,
Shanghai 200233, CHINA
Phone: +86-21-5423-5577 FAX: +86-21-5423-4677

SHENZHEN BRANCH
12F, Dawning Mansion, Keji South 12th Road,
Hi-Tech Park, Shenzhen 518057, CHINA
Phone: +86-755-2699-3828 FAX: +86-755-2699-3838

EPSON HONG KONG LTD.
Unit 715-723, 7/F Trade Square, 681 Cheung Sha Wan Road,
Kowloon, Hong Kong
Phone: +852-2585-4600 FAX: +852-2827-4346

EPSON TAIWAN TECHNOLOGY & TRADING LTD.
14F, No. 7, Song Ren Road,
Taipei 110, TAIWAN
Phone: +886-2-8786-6688 FAX: +886-2-8786-6660

EPSON SINGAPORE PTE., LTD.
1 HarbourFront Place,
#03-02 HarbourFront Tower One, Singapore 098633
Phone: +65-6586-5500 FAX: +65-6271-3182

SEIKO EPSON CORP.
KOREA OFFICE
5F, KLI 63 Bldg., 60 Yoido-dong
Youngdeungpo-Ku, Seoul, 150-763, KOREA
Phone: +82-2-784-6027 FAX: +82-2-767-3677

SEIKO EPSON CORP.
MICRODEVICES OPERATIONS DIVISION

IC Sales & Marketing Department
421-8, Hino, Hino-shi, Tokyo 191-8501, JAPAN
Phone: +81-42-587-5814 FAX: +81-42-587-5117

Document Code: X93A-A-006-01
First Issue: September 2012