TMS320C674x/OMAP-L1x Processor Liquid Crystal Display Controller (LCDC)

User's Guide

Literature Number: SPRUFM0B
August 2010

Preface **6**
1 Introduction **8**
1.1 Purpose of the Peripheral 8
1.2 Features 9
1.3 Terminology Used in this Document 9
2 Architecture **9**
2.1 Clocking 9
2.2 LCD External I/O Signals 11
2.3 DMA Engine 12
2.4 LIDD Controller 13
2.5 Raster Controller 15
3 Registers **25**
3.1 LCD Revision Identification Register (REVID) 25
3.2 LCD Control Register (LCD_CTRL) 26
3.3 LCD Status Register (LCD_STAT) 28
3.4 LCD LIDD Control Register (LIDD_CTRL) 30
3.5 LCD LIDD CS*n* Configuration Registers (LIDD_CS0_CONF and LIDD_CS1_CONF) 32
3.6 LCD LIDD CS*n* Address Read/Write Registers (LIDD_CS0_ADDR and LIDD_CS1_ADDR) 33
3.7 LCD LIDD CS*n* Data Read/Write Registers (LIDD_CS0_DATA and LIDD_CS1_DATA) 34
3.8 LCD Raster Control Register (RASTER_CTRL) 35
3.9 LCD Raster Timing Register 0 (RASTER_TIMING_0) 42
3.10 LCD Raster Timing Register 1 (RASTER_TIMING_1) 44
3.11 LCD Raster Timing Register 2 (RASTER_TIMING_2) 48
3.12 LCD Raster Subpanel Display Register (RASTER_SUBPANEL) 52
3.13 LCD DMA Control Register (LCDDMA_CTRL) 54
3.14 LCD DMA Frame Buffer *n* Base Address Registers (LCDDMA_FB0_BASE and LCDDMA_FB1_BASE) 55
3.15 LCD DMA Frame Buffer *n* Ceiling Address Registers (LCDDMA_FB0_CEILING and LCDDMA_FB1_CEILING) 55
Appendix A Revision History **56**

List of Figures

1 LCD Controller 8
2 Input and Output Clocks 9
3 Logical Data Path for Raster Controller 16
4 Frame Buffer Structure 17
5 16-Entry Palette/Buffer Format (1, 2, 4, 12, 16 BPP) 18
6 256-Entry Palette/Buffer Format (8 BPP) 19
7 16-BPP Data Memory Organization (TFT Mode Only)—Little Endian 19
8 12-BPP Data Memory Organization—Little Endian 20
9 8-BPP Data Memory Organization 20
10 4-BPP Data Memory Organization 20
11 2-BPP Data Memory Organization 21
12 1-BPP Data Memory Organization 21
13 Monochrome and Color Output 23
14 Raster Mode Display Format 24
15 LCD Revision Identification Register (REVID) 25
16 LCD Control Register (LCD_CTRL) 26
17 LCD Status Register (LCD_STAT) 28
18 LCD LIDD Control Register (LIDD_CTRL) 30
19 LCD LIDD CS*n* Configuration Register (LIDD_CS*n*_CONF) 32
20 LCD LIDD CS*n* Address Read/Write Register (LIDD_CS*n*_ADDR) 33
21 LCD LIDD CS*n* Data Read/Write Register (LIDD_CS*n*_DATA) 34
22 LCD Raster Control Register (RASTER_CTRL) 35
23 Monochrome Passive Mode Pixel Clock and Data Pin Timing 38
24 Color Passive Mode Pixel Clock and Data Pin Timing 38
25 Active Mode Pixel Clock and Data Pin Timing 39
26 TFT Alternate Signal Mapping Output 40
27 12-Bit STN Data in Frame Buffer 41
28 16-Bit STN Data in Frame Buffer 41
29 16-BPP STN Mode 41
30 LCD Raster Timing Register 0 (RASTER_TIMING_0) 42
31 LCD Raster Timing Register 1 (RASTER_TIMING_1) 44
32 Vertical Synchronization Pulse Width (VSW) - Active Mode 45
33 Vertical Front Porch (VFP) 46
34 Vertical Back Porch (VBP) 47
35 LCD Raster Timing Register 2 (RASTER_TIMING_2) 48
36 SYNC_CTRL = 0, IPC = 1 in TFT Mode 50
37 SYNC_CTRL = 1, SYNC_EDGE = 0, and IPC = 1 51
38 LCD Raster Subpanel Display Register (RASTER_SUBPANEL) 52
39 Subpanel Display: SPEN = 1, HOLS = 1 53
40 Subpanel Display: SPEN = 1, HOLS = 0 53
41 LCD DMA Control Register (LCDDMA_CTRL) 54
42 LCD DMA Frame Buffer *n* Base Address Register (LCDDMA_FB*n*_BASE) 55
43 LCD DMA Frame Buffer *n* Ceiling Address Register (LCDDMA_FB*n*_CEILING) 55

List of Tables

1 LCD External I/O Signals 11
2 Register Configuration for DMA Engine Programming 12
3 LIDD I/O Name Map 14
4 Operation Modes Supported by Raster Controller 15
5 Bits-Per-Pixel Encoding for Palette Entry 0 Buffer 17
6 Frame Buffer Size According to BPP 18
7 Color/Grayscale Intensities and Modulation Rates 22
8 Number of Colors/Shades of Gray Available on Screen 22
9 LCD Controller (LCDC) Registers 25
10 LCD Revision Identification Register (REVID) Field Descriptions 25
11 LCD Control Register (LCD_CTRL) Field Descriptions 26
12 Pixel Clock Frequency Programming Limitations 27
13 LCD Status Register (LCD_STAT) Field Descriptions 28
14 LCD LIDD Control Register (LIDD_CTRL) Field Descriptions 30
15 LCD LIDD CS*n* Configuration Register (LIDD_CS*n*_CONF) Field Descriptions 32
16 LCD LIDD CS*n* Address Read/Write Register (LIDD_CS*n*_ADDR) Field Descriptions 33
17 LCD LIDD CS*n* Data Read/Write Register (LIDD_CS*n*_DATA) Field Descriptions 34
18 LCD Raster Control Register (RASTER_CTRL) Field Descriptions 35
19 LCD Controller Data Pin Utilization for Mono/Color Passive/Active Panels 37
20 LCD Raster Timing Register 0 (RASTER_TIMING_0) Field Descriptions 42
21 LCD Raster Timing Register 1 (RASTER_TIMING_1) Field Descriptions 44
22 LCD Raster Timing Register 2 (RASTER_TIMING_2) Field Descriptions 48
23 LCD Raster Subpanel Display Register (RASTER_SUBPANEL) Field Descriptions 52
24 LCD DMA Control Register (LCDDMA_CTRL) Field Descriptions 54
25 LCD DMA Frame Buffer *n* Base Address Register (LCDDMA_FB*n*_BASE) Field Descriptions 55
26 LCD DMA Frame Buffer *n* Ceiling Address Register (LCDDMA_FB*n*_CEILING) Field Descriptions 55
27 Document Revision History 56

Read This First

About This Manual

This document describes the liquid crystal display controller (LCDC).

Notational Conventions

This document uses the following conventions.

- Hexadecimal numbers are shown with the suffix h. For example, the following number is 40 hexadecimal (decimal 64): 40h.
- Registers in this document are shown in figures and described in tables.
  - Each register figure shows a rectangle divided into fields that represent the fields of the register. Each field is labeled with its bit name, its beginning and ending bit numbers above, and its read/write properties below. A legend explains the notation used for the properties.
  - Reserved bits in a register figure designate a bit that is used for future device expansion.

Related Documentation From Texas Instruments

The following documents describe the TMS320C674x Digital Signal Processors (DSPs) and OMAP-L1x Applications Processors. Copies of these documents are available on the Internet at www.ti.com. *Tip:* Enter the literature number in the search box provided at www.ti.com.

The current documentation that describes the DSP, related peripherals, and other technical collateral, is available in the C6000 DSP product folder at: www.ti.com/c6000.

SPRUGM5 — ***TMS320C6742 DSP System Reference Guide.*** Describes the C6742 DSP subsystem, system memory, device clocking, phase-locked loop controller (PLLC), power and sleep controller (PSC), power management, and system configuration module.

SPRUGJ0 — ***TMS320C6743 DSP System Reference Guide.*** Describes the System-on-Chip (SoC) including the C6743 DSP subsystem, system memory, device clocking, phase-locked loop controller (PLLC), power and sleep controller (PSC), power management, and system configuration module.

SPRUFK4 — ***TMS320C6745/C6747 DSP System Reference Guide.*** Describes the System-on-Chip (SoC) including the C6745/C6747 DSP subsystem, system memory, device clocking, phase-locked loop controller (PLLC), power and sleep controller (PSC), power management, and system configuration module.

SPRUGM6 — ***TMS320C6746 DSP System Reference Guide.*** Describes the C6746 DSP subsystem, system memory, device clocking, phase-locked loop controller (PLLC), power and sleep controller (PSC), power management, and system configuration module.

SPRUGJ7 — ***TMS320C6748 DSP System Reference Guide.*** Describes the C6748 DSP subsystem, system memory, device clocking, phase-locked loop controller (PLLC), power and sleep controller (PSC), power management, and system configuration module.

SPRUG84 — ***OMAP-L137 Applications Processor System Reference Guide.*** Describes the System-on-Chip (SoC) including the ARM subsystem, DSP subsystem, system memory, device clocking, phase-locked loop controller (PLLC), power and sleep controller (PSC), power management, ARM interrupt controller (AINTC), and system configuration module.

SPRUGM7 — ***OMAP-L138 Applications Processor System Reference Guide.*** Describes the System-on-Chip (SoC) including the ARM subsystem, DSP subsystem, system memory, device clocking, phase-locked loop controller (PLLC), power and sleep controller (PSC), power management, ARM interrupt controller (AINTC), and system configuration module.

SPRUFK9 — ***TMS320C674x/OMAP-L1x Processor Peripherals Overview Reference Guide.*** Provides an overview and briefly describes the peripherals available on the TMS320C674x Digital Signal Processors (DSPs) and OMAP-L1x Applications Processors.

SPRUFK5 — ***TMS320C674x DSP Megamodule Reference Guide.*** Describes the TMS320C674x digital signal processor (DSP) megamodule. Included is a discussion on the internal direct memory access (IDMA) controller, the interrupt controller, the power-down controller, memory protection, bandwidth management, and the memory and cache.

SPRUFE8 — ***TMS320C674x DSP CPU and Instruction Set Reference Guide.*** Describes the CPU architecture, pipeline, instruction set, and interrupts for the TMS320C674x digital signal processors (DSPs). The C674x DSP is an enhancement of the C64x+ and C67x+ DSPs with added functionality and an expanded instruction set.

SPRUG82 — ***TMS320C674x DSP Cache User's Guide.*** Explains the fundamentals of memory caches and describes how the two-level cache-based internal memory architecture in the TMS320C674x digital signal processor (DSP) can be efficiently used in DSP applications. Shows how to maintain coherence with external memory, how to use DMA to reduce memory latencies, and how to optimize your code to improve cache efficiency. The internal memory architecture in the C674x DSP is organized in a two-level hierarchy consisting of a dedicated program cache (L1P) and a dedicated data cache (L1D) on the first level. Accesses by the CPU to the these first level caches can complete without CPU pipeline stalls. If the data requested by the CPU is not contained in cache, it is fetched from the next lower memory level, L2 or external memory.

Liquid Crystal Display Controller (LCDC)

1 Introduction

The liquid crystal display controller (LCDC) is capable of supporting an asynchronous (memory-mapped) LCD interface and a synchronous (raster-type) LCD interface.

1.1 Purpose of the Peripheral

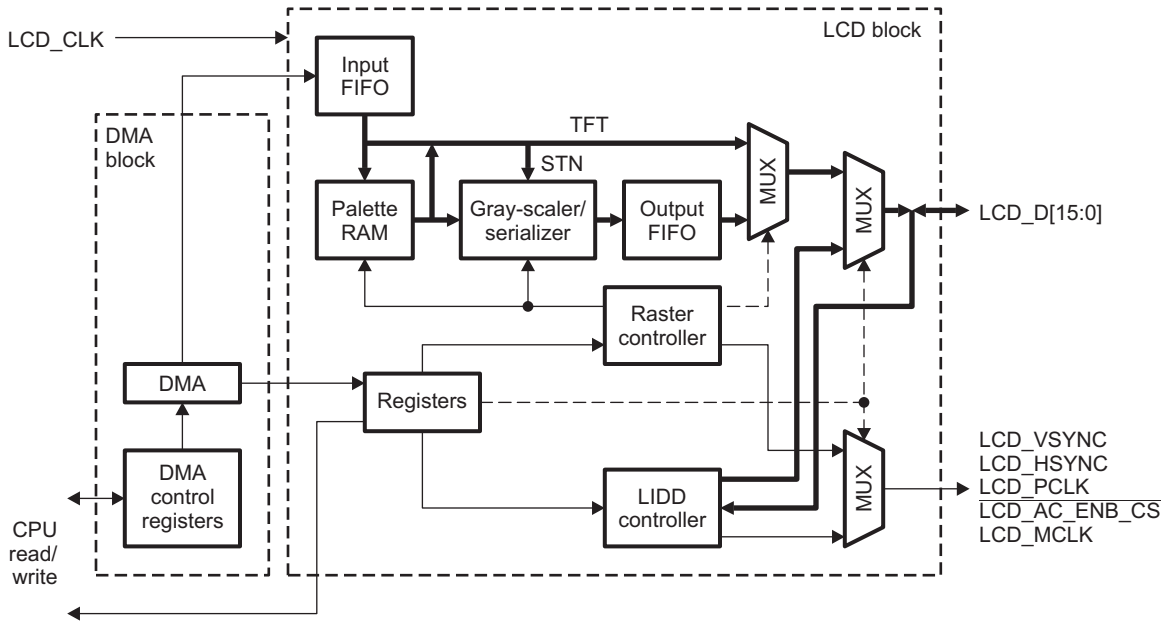
The LCD controller consists of two independent controllers, the Raster Controller and the LCD Interface Display Driver (LIDD) controller. Each controller operates independently from the other and only one of them is active at any given time.

- The Raster Controller handles the synchronous LCD interface. It provides timing and data for constant graphics refresh to a passive display. It supports a wide variety of monochrome and full-color display types and sizes by use of programmable timing controls, a built-in palette, and a gray-scale/serializer. Graphics data is processed and stored in frame buffers. A frame buffer is a contiguous memory block in the system. A built-in DMA engine supplies the graphics data to the Raster engine which, in turn, outputs to the external LCD device.
- The LIDD Controller supports the asynchronous LCD interface. It provides full-timing programmability of control signals (CS, WE, OE, ALE) and output data.

Figure 1 shows the LCD controller details. The raster and LIDD Controllers are responsible for generating the correct external timing. The DMA engine provides a constant flow of data from the frame buffer(s) to the external LCD panel via the Raster and LIDD Controllers. In addition, CPU access is provided to read and write registers.

The solid, thick lines in Figure 1 indicate the data path. The Raster Controller's data path is fairly complicated, for a thorough description of the Raster Controller data path, see Section 2.5.

Figure 1. LCD Controller



1.2 Features

See your device-specific data manual to check the features supported by the LCD controller.

1.3 Terminology Used in this Document

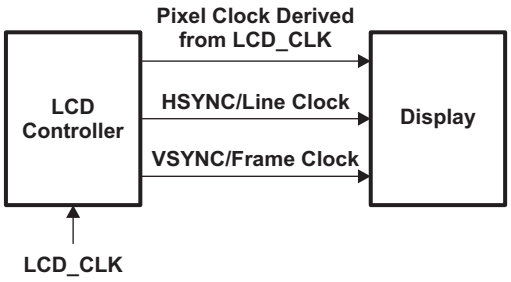
Term	Meaning
Passive (STN) device	Refers to the Super-Twisted Nematic (STN) display device.
Active (TFT) device	Refers to the Thin-Film Transistor (TFT) display device.
BPP	Bits per pixel; that is, the number of bits used for each pixel. In some documentation, this is also referred to as color depth.
RGB	Red, Green, Blue
ALE	Address latch enable
CS	Chip select
OE	Output enable
WE	Write enable

2 Architecture

2.1 Clocking

This section details the various clocks and signals. Figure 2 shows input and output LCD controller clocks.

Figure 2. Input and Output Clocks



2.1.1 Pixel Clock

The pixel clock (LCD_PCLK) frequency is derived from LCD_CLK, the reference clock to this LCD module (see Figure 2). The pixel clock is used by the LCD display to clock the pixel data into the line shift register.

$$\text{LCD\_PCLK} = \frac{\text{LCD\_CLK}}{\text{CLKDIV}}$$

where CLKDIV is a field in the LCD_CTRL register and should not be 0 or 1.

- **Passive (STN) mode.** LCD_PCLK only transitions when valid data is available for output. It does not transition when the horizontal clock is asserted or during wait state insertion.
- **Active (TFT) mode.** LCD_PCLK continuously toggles as long as the Raster Controller is enabled.

2.1.2 Horizontal Clock (LCD_HSYNC)

LCD_HSYNC toggles after all pixels in a horizontal line have been transmitted to the LCD and a programmable number of pixel clock wait states has elapsed both at the beginning and end of each line.

The RASTER_TIMING_0 register fully defines the behavior of this signal.

LCD_HSYNC can be programmed to be synchronized with the rising or falling edge of LCD_PCLK. The configuration field is bits 24 and 25 in the RASTER_TIMING_2 register.

- **Active (TFT) mode.** The horizontal clock or the line clock is also used by TFT displays as the horizontal synchronization signal (LCD_HSYNC).

  The timings of the horizontal clock(line clock) pins are programmable to support:
- Delay insertion both at the beginning and end of each line
- Line clock polarity.
- Line clock pulse width, driven on rising or falling edge of pixel clock.

2.1.3 Vertical Clock (LCD_VSYNC)

LCD_VSYNC toggles after all lines in a frame have been transmitted to the LCD and a programmable number of line clock cycles has elapsed both at the beginning and end of each frame.

The RASTER_TIMING_1 register fully defines the behavior of this signal.

LCD_VSYNC can be programmed to be synchronized with the rising or falling edge of LCD_PCLK. The configuration field is bits 24 and 25 in the RASTER_TIMING_2 register.

- **Passive (STN) mode.** The vertical clock; that is, the frame clock, toggles during the first line of the screen.
- **Active (TFT) mode.** The vertical clock, that is, the frame clock, is also used by TFT displays as the vertical synchronization signal (LCD_VSYNC).

  The timings of the vertical clock pins are programmable to support:
- Delay insertion both at the beginning and end of each frame
- Frame clock polarity

2.1.4 $\overline{\text{LCD\_AC\_ENB\_CS}}$

- **Passive (STN) mode.** To prevent a dc charge within the screen pixels, the power and ground supplies of the display are periodically switched. The Raster Controller signals the LCD to switch the polarity by toggling this pin ($\overline{\text{LCD\_AC\_ENB\_CS}}$).
- **Active (TFT) mode.** This signal acts as an output enable (OE) signal. It is used to signal the external LCD that the data is valid on the data bus (LCD_D[15:0]). (see)

2.2 LCD External I/O Signals

Table 1 shows the details of the LCD controller external signals.

Table 1. LCD External I/O Signals

Signal	Type	Description
LCD_VSYNC	OUT	Raster controller: Frame clock the LCD uses to signal the start of a new frame of pixels. Also used by TFT displays as the vertical synchronization signal. LIDD character: Register select (RS) or address latch enable (ALE) LIDD graphics: Address bit 0 (A0) or command/data select (C/D)
LCD_HSYNC	OUT	Raster controller: Line clock the LCD uses to signal the end of a line of pixels that transfers line data from the shift register to the screen and to increment the line pointer(s). Also used by TFT displays as the horizontal synchronization signal. LIDD character: not used. LIDD graphics: • 6800 mode = read or write enable • 8080 mode = not write strobe
LCD_PCLK	OUT	Raster controller: Pixel clock the LCD uses to clock the pixel data into the line shift register. In passive mode, the pixel clock transitions only when valid data is available on the data lines. In active mode, the pixel clock transitions continuously, and the ac-bias pin is used as an output enable to signal when data is available on the LCD pin. LIDD character: not used. LIDD graphics: • 6800 mode = enable strobe • 8080 mode = not read strobe
$\overline{\text{LCD\_AC\_ENB\_CS}}$	OUT	Raster controller: ac-bias used to signal the LCD to switch the polarity of the power supplies to the row and column axis of the screen to counteract DC offset. Used in TFT mode as the output enable to signal when data is latched from the data pins using the pixel clock. LIDD character: Primary enable strobe LIDD graphics: Chip select 0 (CS0)
LCD_MCLK	OUT	Raster controller: not used. LIDD character: Secondary enable strobe LIDD graphics: Chip select 1 (CS1)
LCD_D[15:0]	Raster: OUT	LCD data bus, providing a 4-, 8-, or 16-bit data path.
	LIDD: OUT/IN	Raster controller: For monochrome displays, each signal represents a pixel; for passive color displays, groupings of three signals represent one pixel (red, green, and blue). LCD_D[3:0] is used for monochrome displays of 2, 4, and 8 BPP; LCD_D[7:0] is used for color STN displays and LCD_D[15:0] is used for active (TFT) mode. LIDD character: Read and write the command and data registers. LIDD graphics: Read and write the command and data registers.

2.3 DMA Engine

The DMA engine provides the capability to output graphics data to constantly refresh LCDs, without burdening the CPU, via interrupts or a firmware timer. It operates on one or two frame buffers, which are set up during initialization. Using two frame buffers (ping-pong buffers) enables the simultaneous operation of outputting the current video frame to the external display and updating the next video frame. The ping-pong buffering approach is preferred in most applications.

When the Raster Controller is used, the DMA engine reads data from a frame buffer and writes it to the input FIFO (as shown in Figure 1). The Raster Controller requests data from the FIFO for frame refresh; as a result, the DMA's job is to ensure that the FIFO is always kept full.

When the LIDD Controller is used, the DMA engine accesses the LIDD Controller's address and/or data registers.

To program DMA engine, configure the following registers, as shown in Table 2.

Table 2. Register Configuration for DMA Engine Programming

Register	Configuration
LCDDMA_CTRL	Configure DMA data format
LCDDMA_FB0_BASE LCDDMA_FB0_CEILING	Configure frame buffer 0
LCDDMA_FB1_BASE LCDDMA_FB1_CEILING	Configure frame buffer 1. (If only one frame buffer is used, these two registers will not be used.)

In addition, the LIDD_CTRL register (for LIDD Controller) or the RASTER_CTRL register (for Raster Controller) should also be configured appropriately, along with all the timing registers.

To enable DMA transfers, the LIDD_DMA_EN bit (in the LIDD_CTRL register) or the LCDEN bit (in the RASTER_CTRL register) should be written with 1.

2.3.1 Interrupts

Interrupts in this LCD module are related to DMA engine operation. Three registers are closely related to this subject:

- The LIDD_CTRL and RASTER_CTRL registers enable or disable each individual interrupt sources.
- The LCD_STAT register collects all the interrupt status information.

2.3.1.1 LIDD Mode

When operating in LIDD mode, the DMA engine generates one interrupt signal every time the specified frame buffer has been transferred completely.

- The DONE_INT_EN bit in the LIDD_CTRL register specifies if the interrupt signal is delivered to the system interrupt controller, which in turn may or may not generate an interrupt to CPU.
- The EOF1, EOF0, and DONE bits in the LCD_STAT register reflect the interrupt signal, regardless of being delivered to the system interrupt controller or not.

2.3.1.2 Raster Mode

When operating in Raster mode, the DMA engine can generate the interrupts in the following scenarios:

1. **Output FIFO under-run**. This occurs when the DMA engine cannot keep up with the data rate consumed by the LCD (which is determined by the LCD_PCLK.) This is likely due to a system memory throughput issue or an incorrect LCD_PCLK setting. The FUF bit in LCD_STAT is set when this error occurs. This bit is cleared by disabling the Raster Controller (i.e., clearing the LCDEN bit in RASTER_CTRL).

2. **Frame synchronization lost**. This error happens when the DMA engine attempts to read what it believes to be the first word of the video buffer but it cannot be recognized as such. This could be caused by an invalid frame buffer address or an invalid BPP value (for more details, see Section 2.5.2). The SYNC bit in the LCD_STAT register is set when such an error is detected. This field is cleared by disabling the Raster Controller (clearing the LCDEN bit in the RASTER_CTRL register).
3. **Palette loaded**. This interrupt can be generated when the palette is loaded into the memory by the DMA engine. At the same time, the PL bit in the LCD_STAT register is set. In data-only (PLM = 2h) and palette-plus-data (PLM = 00) modes, writing 0 to this bit clears the interrupt. In the palette-only (PLM = 1) mode, this bit is cleared by disabling the Raster Controller (clearing the LCDEN bit in the RASTER_CTRL register).
4. **AC bias transition**. If the ACB_I bit in the RASTER_TIMING_2 register is programmed with a non-zero value, an internal counter will be loaded with this value and starts to decrement each time LCD_AC_ENB_CS (AC-bias signal) switches its state. When the counter reaches zero, the ABC bit in the LCD_STAT register is set, which will deliver an interrupt signal to the system interrupt controller (if the interrupt is enabled.) The counter reloads the value in field ACB_I, but does not start to decrement until the ABC bit is cleared by writing 0 to this bit.
5. **Frame transfer completed**. When one frame of data is transferred completely, the DONE bit in the LCD_STAT register is set. This bit is cleared by disabling the Raster Controller (i.e., clearing the LCDEN bit in the RASTER_CTRL register). Note that the EOF0 and EOF1 bits in the LCD_STAT register will be set accordingly.

Note that the interrupt enable bits are in the RASTER_CTRL register. The corresponding enable bit must be set in order to generate an interrupt to the CPU. However, the LCD_STAT register reflects the interrupt signal regardless of the interrupt enable bits settings.

2.3.1.3 Interrupt Handling

Refer the device-specific data manual for information about LCD interrupt number to CPU .The interrupt service routine needs to determine the interrupt source by examining the LCD_STAT register and clearing the interrupt properly.

2.4 LIDD Controller

The LIDD Controller is designed to support LCD panels with a memory-mapped interface. The types of displays range from low-end character monochrome LCD panels to high-end TFT smart LCD panels.

LIDD mode (and the use of this logic) is enabled by clearing the MODESEL bit in the LCD control register (LCD_CTRL).

LIDD Controller operation is summarized as follows:

- During initialization, the LCD LIDD CS0/CS1 configuration registers (LIDD_CS0_CONF and LIDD_CS1_CONF) are configured to match the requirements of the LCD panel being used.
- During normal operation, the CPU writes display data to the LCD data registers (LIDD_CS0_DATA and LIDD_CS1_DATA). The LIDD interface converts the CPU write into the proper signal transition sequence for the display, as programmed earlier. Note that the first CPU write should send the beginning address of the update to the LCD panel and the subsequent writes update data at display locations starting from the first address and continuing sequentially. Note that DMA may be used instead of CPU.
- The LIDD Controller is also capable of reading back status or data from the LCD panel, if the latter has this capability. This is set up and activated in a similar manner to the write function described above.

NOTE: If an LCD panel is not used, this interface can be used to control any MCU-like peripheral.

See your device-specific data manual to check the LIDD features supported by the LCD controller.

Table 3 describes how the signals are used to interface external LCD modules, which are configured by the LIDD_CTRL register.

Table 3. LIDD I/O Name Map

Display Type	Interface Type	Data Bits	LIDD_CTRL [2:0]	I/O Name	Display I/O Name	Comment
Character Display	HD44780 Type	4	100	LCD_D[7:4]	DATA[7:4]	Data Bus (length defined by Instruction)
				LCD_PCLK	E (or E0)	Enable Strobe (first display)
				LCD_HSYNC	R/$\overline{W}$	Read/$\overline{Write}$
				LCD_VSYNC	RS	Register Select (Data/not Instruction)
				LCD_MCLK	E1	Enable Strobe (second display optional)
Character Display	HD44780 Type	8	100	LCD_D[7:0]	DATA[7:0]	Data Bus (length defined by Instruction)
				LCD_PCLK	E (or E0)	Enable Strobe (first display)
				LCD_HSYNC	R/$\overline{W}$	Read/$\overline{Write}$
				LCD_VSYNC	RS	Register Select (Data/not Instruction)
				LCD_MCLK	E1	Enable Strobe (second display optional)
Micro Interface Graphic Display	6800 Family	Up to 16	001	LCD_D[15:0]	DATA[7:0]	Data Bus (16 bits always available)
				LCD_PCLK	E	Enable Clock
				LCD_HSYNC	R/$\overline{W}$	Read/$\overline{Write}$
				LCD_VSYNC	A0	Address/Data Select
				$\overline{LCD\_AC\_ENB\_CS}$	CS (or CS0)	Chip Select (first display)
				LCD_MCLK	CS1	Chip Select (second display optional)
			000	LCD_MCLK	None	Synchronous Clock (optional)
Micro Interface Graphic Display	8080 Family	Up to 16	011	LCD_D[15:0]	DATA[7:0]	Data Bus (16 bits always available)
				LCD_PCLK	RD	Read Strobe
				LCD_HSYNC	WR	Write Strobe
				LCD_VSYNC	A0	Address/Data Select
				$\overline{LCD\_AC\_ENB\_CS}$	CS (or CS0)	Chip Select (first display)
				LCD_MCLK	CS1	Chip Select (second display optional)
			010	LCD_MCLK	None	Synchronous Clock (optional)

The timing parameters are defined by the LIDD_CS0_CONF and LIDD_CS1_CONF registers, which are described in Section 3.5.

The timing configuration is based on an internal reference clock, MCLK. The MCLK is generated out of LCD_CLK, which is determined by the CLKDIV bit in the LCD_CTRL register.

$$MCLK = LCD\_CLK \text{ when } CLKDIV = 0.$$

$$MCLK = \frac{LCD\_CLK}{CLKDIV} \text{ when } CLKDIV \neq 0.$$

See your device-specific data manual for the timing configurations supported by the LCD controller.

2.5 Raster Controller

Raster mode (and the use of this logic) is enabled by setting the MODESEL bit in the LCD control register (LCD_CTRL). Table 4 shows the active external signals when this mode is active.

Table 4. Operation Modes Supported by Raster Controller

Interface	Data Bus Width	Register Bits RASTER_CTRL[9, 7, 1]	Signal Name	Description
Passive (STN) Mono 4-bit	4	001	LCD_D[3:0]	Data bus
			LCD_PCLK	Pixel clock
			LCD_HSYNC	Horizontal clock(Line Clock)
			LCD_VSYNC	Vertical clock (Frame Clock)
			$\overline{\text{LCD\_AC\_ENB\_CS}}$	AC Bias
			LCD_MCLK	Not used
Passive (STN) Mono 8-bit	8	101	LCD_D[7:0]	Data bus
			LCD_PCLK	Pixel clock
			LCD_HSYNC	Horizontal clock(Line Clock)
			LCD_VSYNC	Vertical clock (Frame Clock)
			$\overline{\text{LCD\_AC\_ENB\_CS}}$	AC Bias
			LCD_MCLK	Not used
Passive (STN) Color	8	100	LCD_D[7:0]	Data bus
			LCD_PCLK	Pixel clock
			LCD_HSYNC	Horizontal clock(Line Clock)
			LCD_VSYNC	Vertical clock (Frame Clock)
			$\overline{\text{LCD\_AC\_ENB\_CS}}$	AC Bias
			LCD_MCLK	Not used
Active (TFT) Color	16	x10	LCD_D[15:0]	Data bus
			LCD_PCLK	Pixel clock
			LCD_HSYNC	Horizontal clock(Line Clock)
			LCD_VSYNC	Vertical clock (Frame Clock)
			$\overline{\text{LCD\_AC\_ENB\_CS}}$	Output enable
			LCD_MCLK	Not used

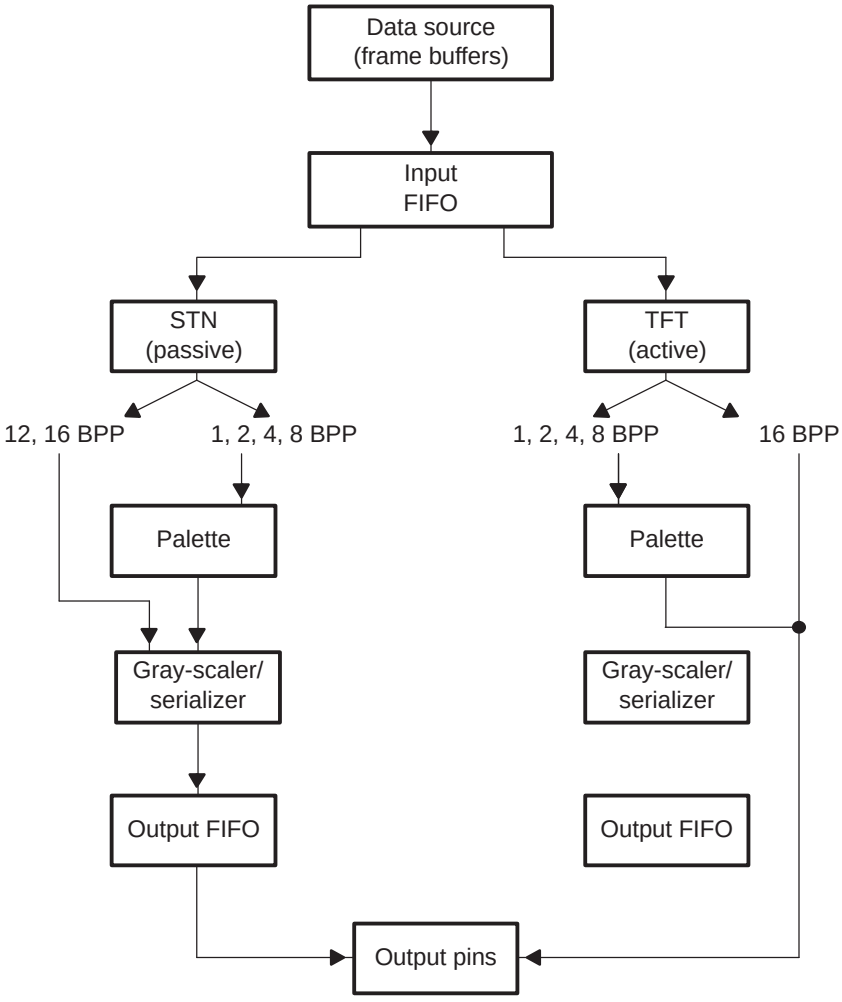
2.5.1 Logical Data Path

The block diagram of the Raster Controller is shown in Figure 1. Figure 3 illustrates its logical data path for various operation modes (passive (STN) versus active (TFT), various BPP size).

Figure 3 shows that:

- The gray-scaler/serializer and output FIFO blocks are bypassed in active (TFT) modes.
- The palette is bypassed in both 12- and 16-BPP modes.

Figure 3. Logical Data Path for Raster Controller



In summary:

- The display image is stored in frame buffers.
- The built-in DMA engine constantly transfers the data stored in the frame buffers to the Input FIFO.
- The Raster Controller relays data to the external pins according to the specified format.

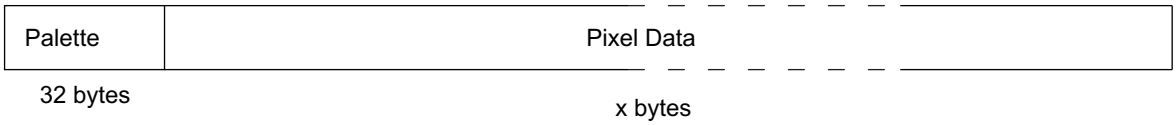
The remainder of this section describes the functioning blocks in Figure 3, including frame buffers, palette, and gray-scaler/serializer. Their operation and programming techniques are covered in detail. The output format is also described in Section 2.5.5.

2.5.2 Frame Buffer

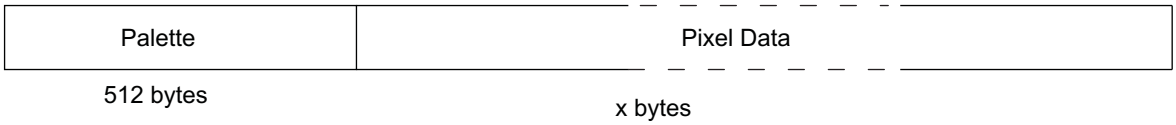
A frame buffer is a contiguous memory block, storing enough data to fill a full LCD screen. For this device, external memory needs to be used for the frame buffer. For specific details on which external memory interface (EMIF) controller can be accessed by the LCD controller, see your device-specific data manual. The data in the frame buffer consists of pixel values as well as a look-up palette. Figure 4 shows the frame buffer structure.

Figure 4. Frame Buffer Structure

1, 2, 4, 12, 16 BPP Modes



8 BPP Mode



NOTE:

- 8-BPP mode uses the first 512 bytes in the frame buffer as the palette while the other modes use 32 bytes.
- 12- and 16-BPP modes do not need a palette; i.e., the pixel data is the desired RGB value. However, the first 32 bytes are still considered a palette. The first entry should be 4000h (bit 14 is 1) while the remaining entries must be filled with 0. (For details, see Table 5.)
- Each entry in a palette occupies 2 bytes. As a result, 8-BPP mode palette has 256 color entries while the other palettes have up to 16 color entries.
- 4-BPP mode uses up the all the 16 entries in a palette.
- 1-BPP mode uses the first 2 entries in a palette while 2-BPP mode uses the first 4 entries. The remaining entries are not used and must be filled with 0.
- In 12- and 16-BPP modes, pixel data is RGB data. For all the other modes, pixel data is actually an index of the palette entry.

Table 5. Bits-Per-Pixel Encoding for Palette Entry 0 Buffer

Bit	Name	Value	Description [1] [2]
14-12	BPP		Bits-per-pixel.
		000	1 BPP
		001	2 BPP
		010	4 BPP
		011	8 BPP
		1xx	12 BPP in passive mode (TFT_STN = 0 and STN_565 = 0 in RASTER_CTRL)
			16 BPP in passive mode (TFT_STN = 0 and STN_565 = 1 in RASTER_CTRL)
			16 BPP in active mode (TFT_STN = 1 in RASTER_CTRL)

(1) Eight 1-bit pixels, four 2-bit pixels, and two 4-bit pixels are packed into each byte, and 12-bit pixels are right justified on (16-bit) word boundaries (in the same format as palette entry).
(2) For STN565, see the 16 BPP STN mode bit (Section 3.8.8).

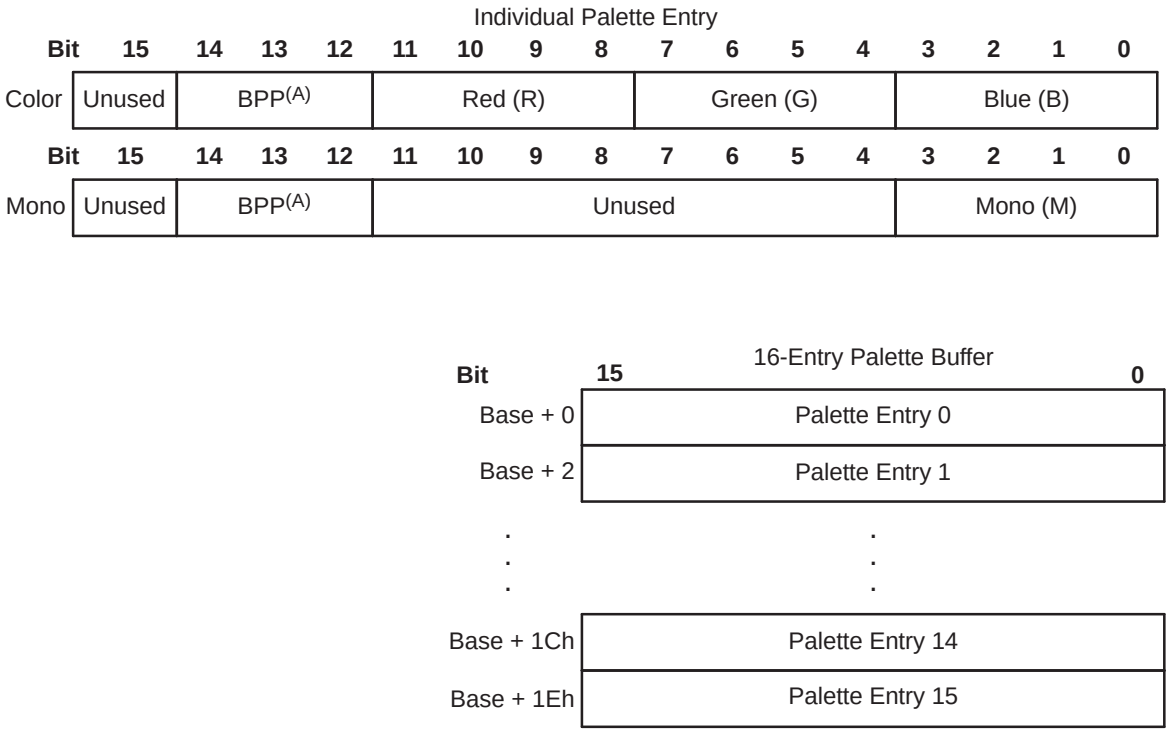
The equations shown in Table 6 are used to calculate the total frame buffer size (in bytes) based on varying pixel size encoding and screen sizes.

Figure 5 and Figure 6 show more detail of the palette entry organization.

Table 6. Frame Buffer Size According to BPP

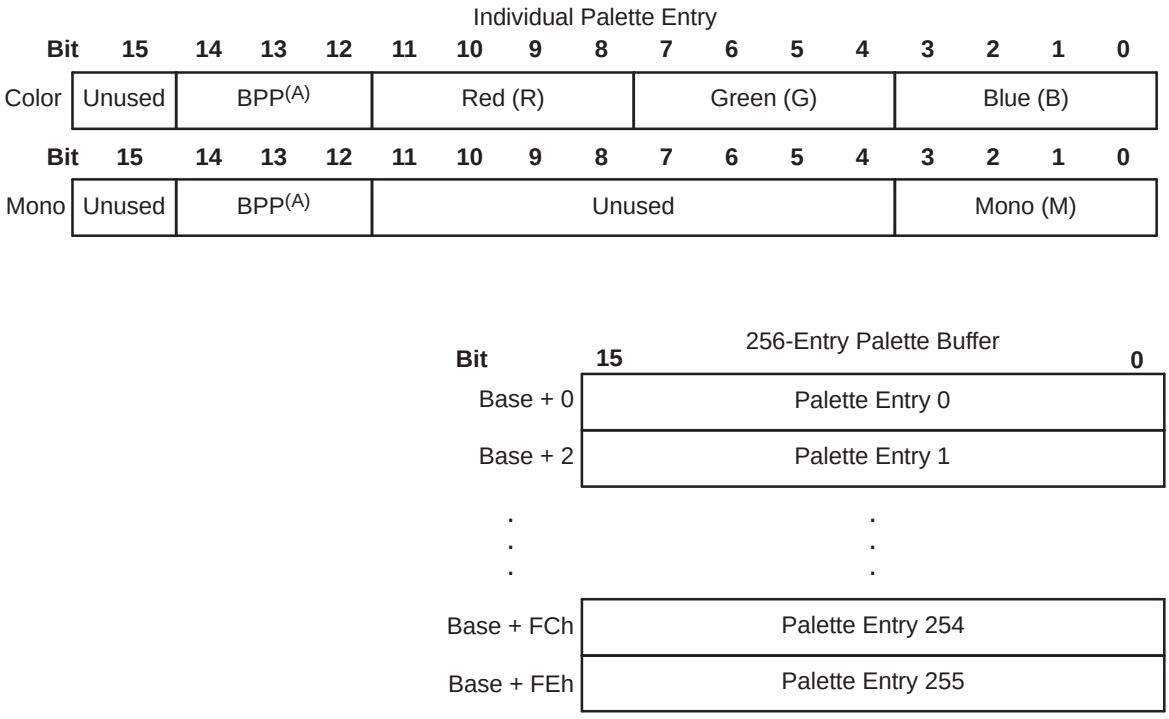
BPP	Frame Buffer Size
1	32 + (Lines × Columns)/8
2	32 + (Lines × Columns)/4
4	32 + (Lines × Columns)/2
8	512 + (Lines × Columns)
12/16	32 + 2 × (Lines × Columns)

Figure 5. 16-Entry Palette/Buffer Format (1, 2, 4, 12, 16 BPP)



A. Bits-per-pixels (BPP) is only contained within the first palette entry (palette entry 0).

Figure 6. 256-Entry Palette/Buffer Format (8 BPP)



A. Bits-per-pixels (BPP) is only contained within the first palette entry (palette entry 0).

Bits 12, 13, and 14 of the first palette entry select the number of bits-per-pixel to be used in the following frame and thus the number of palette RAM entries. The palette entry is used by the Raster Controller to correctly unpack pixel data.

The following figures show the memory organization within the frame buffer for each pixel encoding size.

Figure 7. 16-BPP Data Memory Organization (TFT Mode Only)—Little Endian

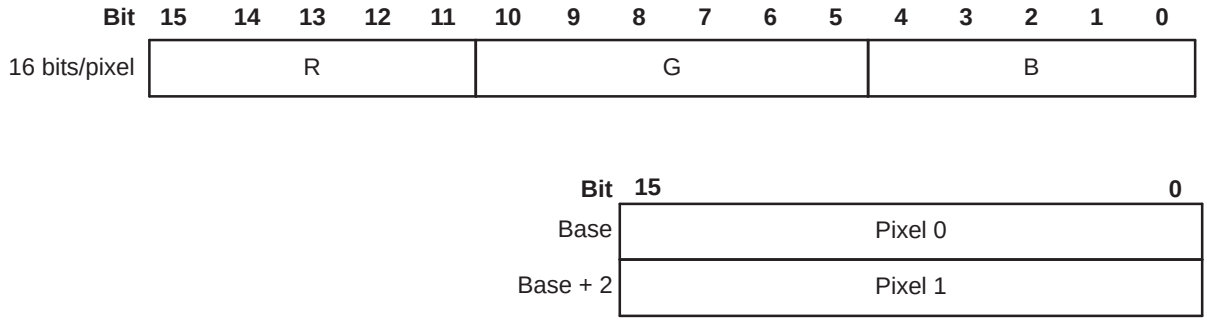
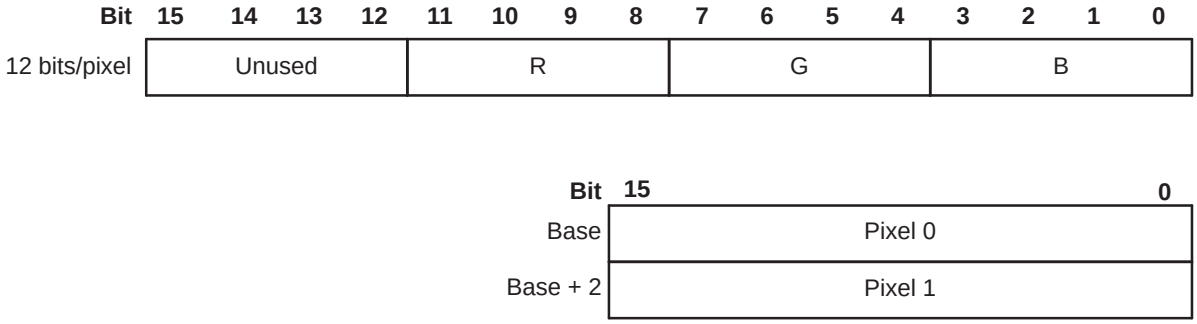


Figure 8. 12-BPP Data Memory Organization—Little Endian



Unused [15-12] bits are filled with zeroes in TFT mode.

Figure 9. 8-BPP Data Memory Organization

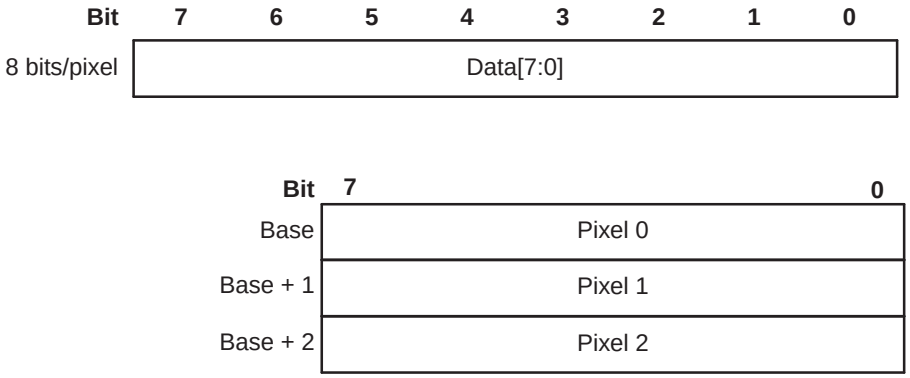


Figure 10. 4-BPP Data Memory Organization

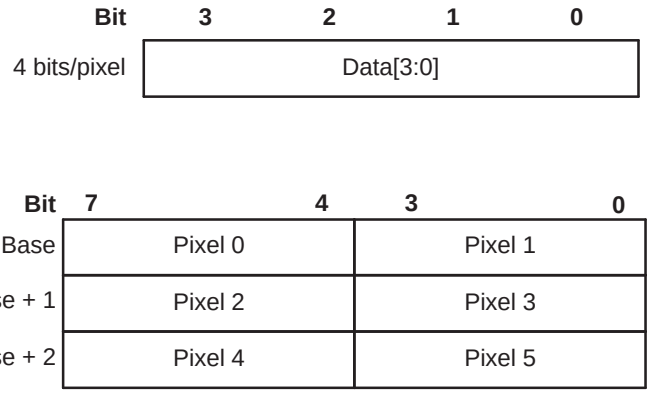


Figure 11. 2-BPP Data Memory Organization

Bit	7	6	5	4	3	2	1	0
Base	Pixel 0		Pixel 1		Pixel 2		Pixel 3	
Base + 1	Pixel 4		Pixel 5		Pixel 6		Pixel 7	
Base + 2	Pixel 8		Pixel 9		Pixel 10		Pixel 11	

Figure 12. 1-BPP Data Memory Organization

Bit	7	6	5	4	3	2	1	0
Base	P0	P1	P2	P3	P4	P5	P6	P7
Base + 1	P8	P9	P10	P11	P12	P13	P14	P15

2.5.3 Palette

As explained in the previous section, the pixel data is an index of palette entry (when palette is used). The number of colors supported is given by $2^{\text{number of BPP}}$. However, due to a limitation of the gray-scaler/serializer block, fewer grayscales or colors may be supported.

The PLM field (in RASTER_CTRL) affects the palette loading:

- If PLM is 00b (palette-plus-data mode) or 01b (palette-only mode), the palette is loaded by the DMA engine at the very beginning, which is followed by the loading of pixel data.
- If PLM is 10b (data-only mode), the palette is not loaded. Instead, the DMA engine loads the pixel data immediately.

2.5.4 Gray-Scaler/Serializer

2.5.4.1 Passive (STN) Mode

Once a palette entry is selected from the look-up palette by the pixel data, its content is sent to the gray-scaler/serializer. If it is monochrome data, it is encoded as 4 bits. If it is color data, it is encoded as 4 bits (Red), 4 bits (Green), and 4 bits (Blue).

These 4-bit values are used to select one of the 16 intensity levels, as shown in Table 7. A patented algorithm is used during this processing to provide an optimized intensity value that matches the eye's visual perception of color/gray gradations.

2.5.4.2 Active (TFT) Mode

The gray-scaler/serializer is bypassed.

Table 7. Color/Grayscale Intensities and Modulation Rates

Dither Value (4-Bit Value from Palette)	Intensity (0% is White)	Modulation Rate (Ratio of ON to ON+OFF Pixels)
0000	0.0%	0
0001	11.1%	1/9
0010	20.0%	1/5
0011	26.7%	4/15
0100	33.3%	3/9
0101	40.0%	2/5
0110	44.4%	4/9
0111	50.0%	1/2
1000	55.6%	5/9
1001	60.0%	3/5
1010	66.6%	6/9
1011	73.3%	11/15
1100	80.0%	4/5
1101	88.9%	8/9
1110	100.0%	1
1111	100.0%	1

2.5.4.3 Summary of Color Depth

Table 8. Number of Colors/Shades of Gray Available on Screen

Number of BPP	Passive Mode (TFT_STN = 0)		Active Mode (TFT_STN = 1)
	Monochrome (MONO_COLOR = 1)	Color (MONO_COLOR = 0)	Color Only (MONO_COLOR = 0)
1	2 palette entries to select within 15 grayscales	2 palette entries to select within 3375 possible colors	2 palette entries to select within 4096 possible colors
2	4 palette entries to select within 15 grayscales	4 palette entries to select within 3375 possible colors	4 palette entries to select within 4096 possible colors
4	16 palette entries to select within 15 grayscales	16 palette entries to select within 3375 possible colors	16 palette entries to select within 4096 possible colors
8	Not relevant since it would consist in 256 palette entries to select within 15 grayscales, but exists anyway	256 palette entries to select 3375 possible colors	256 palette entries to select within 4096 possible colors
12	x	3375 possible colors	4096 possible colors
16	x	3375 possible colors (STN_565 = 1)	Up to 65536 possible colors

2.5.5 Output Format

2.5.5.1 Passive (STN) Mode

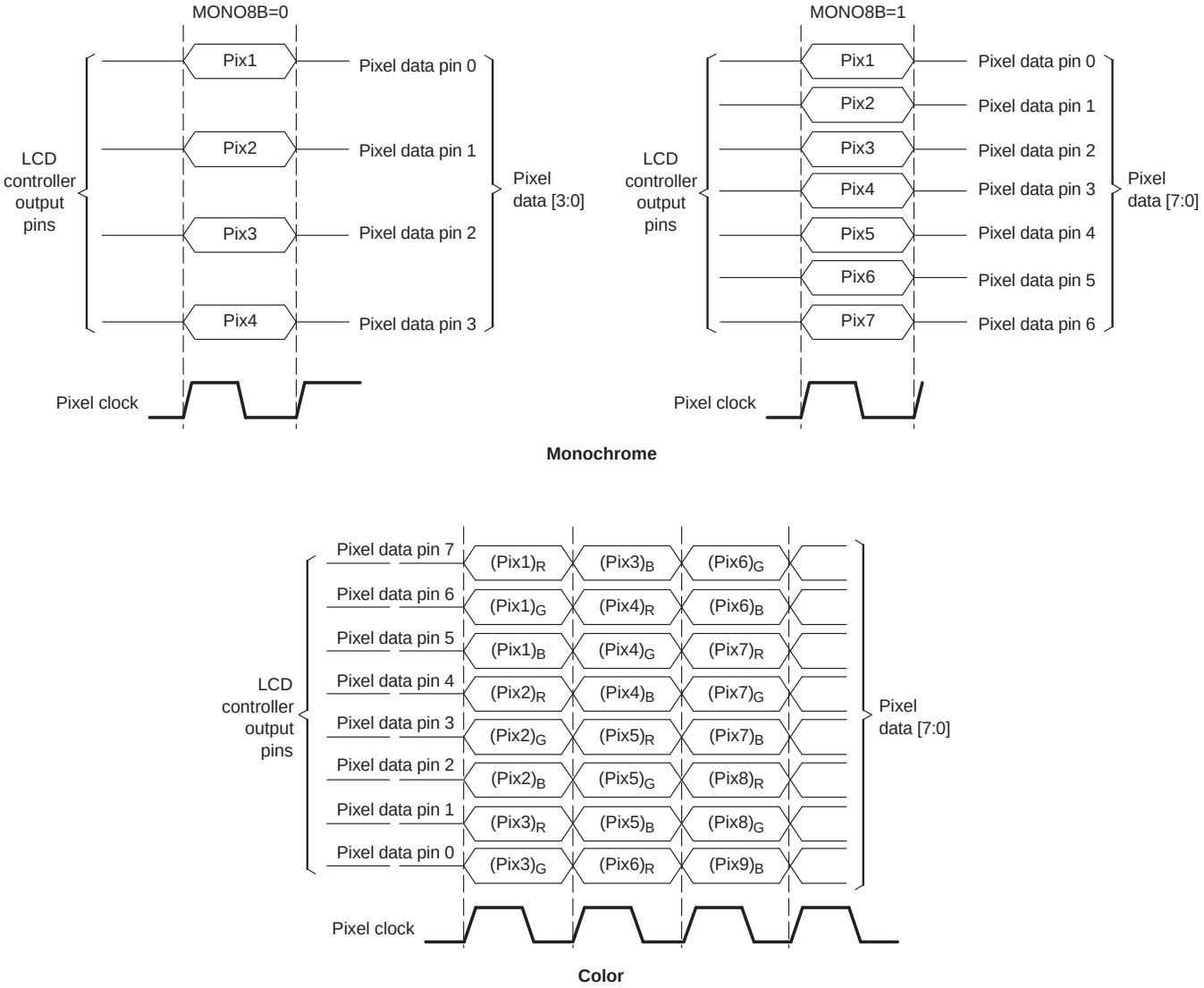
As shown in Figure 3, the pixel data stored in frame buffers go through palette (if applicable) and gray-scaler/serializer before reaching the Output FIFO. As a result, it is likely that the data fed to the Output FIFO is numerically different from the data in the frame buffers. (However, they represent the same color or grayscale.)

The output FIFO formats the received data according to display modes (see Table 4). Figure 13 shows the actual data output on the external pins.

2.5.5.2 Active (TFT) Mode

As shown in Figure 3, the gray-scaler/serializer and output FIFO are bypassed in active (TFT) mode. Namely, at each pixel clock, one pixel data (16 bits) is output to the external LCD.

Figure 13. Monochrome and Color Output



2.5.6 Subpanel Feature

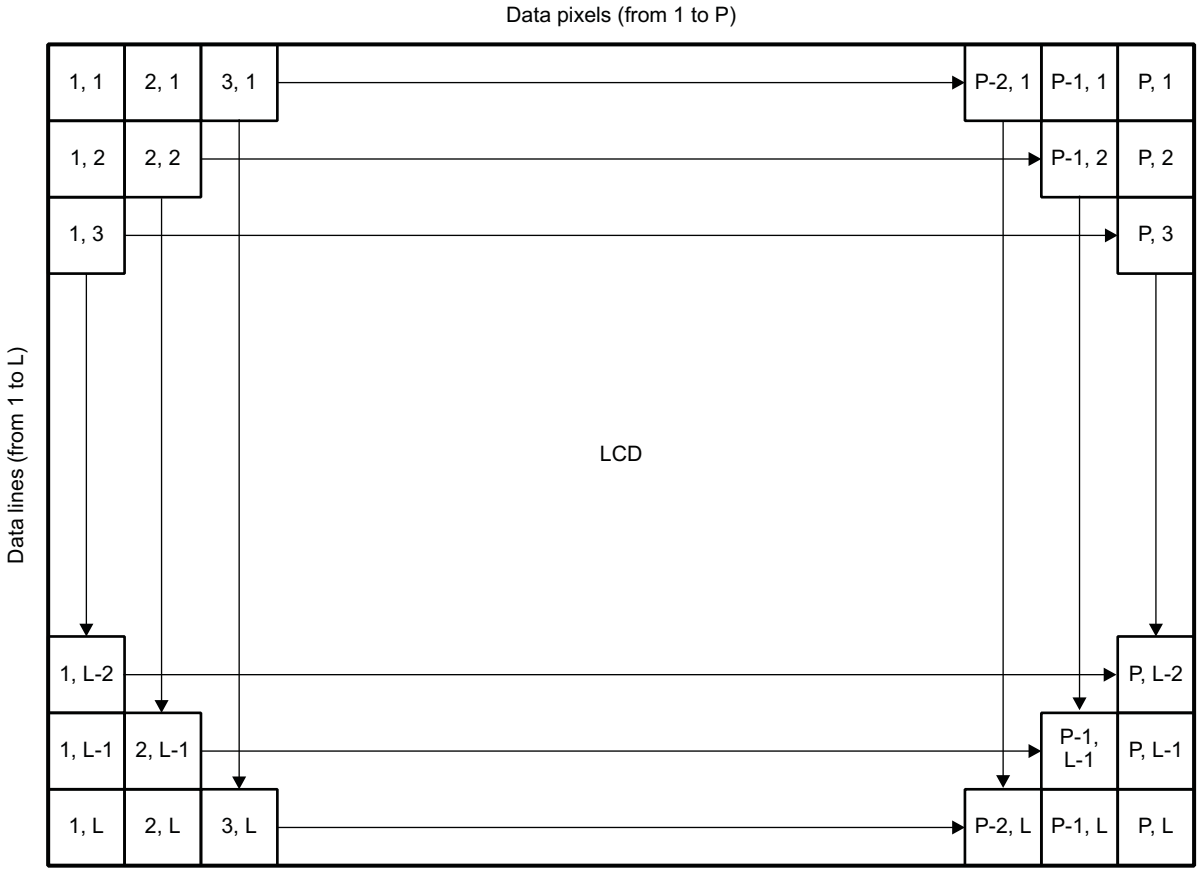
In some applications, it is desired to display only the first or last few lines of the LCD panel (see Figure 14). This is mainly used for power saving.

This is supported by the Raster Controller via its subpanel feature. The RASTER_SUBPANEL register fully defines its behavior, that is, the following parameters are defined:

- Whether the first or last few lines will be refreshed.
- A line number, which is the last (or first) line to be refreshed.
- The pixel data to be loaded to the refresh area.

Note that there is only one pixel value for all the pixels in the refresh area. As a result, frame buffers and DMA engine are not used in this case, which leads to power saving.

Figure 14. Raster Mode Display Format



3 Registers

Table 9 lists the memory-mapped registers for the LCD module.

Table 9. LCD Controller (LCDC) Registers

Address Offset	Acronym	Register Description	Section
0h	REVID	LCD Revision Identification Register	Section 3.1
4h	LCD_CTRL	LCD Control Register	Section 3.2
8h	LCD_STAT	LCD Status Register	Section 3.3
Ch	LIDD_CTRL	LCD LIDD Control Register	Section 3.4
10h	LIDD_CS0_CONF	LCD LIDD CS0 Configuration Register	Section 3.5
14h	LIDD_CS0_ADDR	LCD LIDD CS0 Address Read/Write Register	Section 3.6
18h	LIDD_CS0_DATA	LCD LIDD CS0 Data Read/Write Register	Section 3.7
1Ch	LIDD_CS1_CONF	LCD LIDD CS1 Configuration Register	Section 3.5
20h	LIDD_CS1_ADDR	LCD LIDD CS1 Address Read/Write Register	Section 3.6
24h	LIDD_CS1_DATA	LCD LIDD CS1 Data Read/Write Register	Section 3.7
28h	RASTER_CTRL	LCD Raster Control Register	Section 3.8
2Ch	RASTER_TIMING_0	LCD Raster Timing 0 Register	Section 3.9
30h	RASTER_TIMING_1	LCD Raster Timing 1 Register	Section 3.10
34h	RASTER_TIMING_2	LCD Raster Timing 2 Register	Section 3.11
38h	RASTER_SUBPANEL	LCD Raster Subpanel Display Register	Section 3.12
40h	LCDDMA_CTRL	LCD DMA Control Register	Section 3.13
44h	LCDDMA_FB0_BASE	LCD DMA Frame Buffer 0 Base Address Register	Section 3.14
48h	LCDDMA_FB0_CEILING	LCD DMA Frame Buffer 0 Ceiling Address Register	Section 3.15
4Ch	LCDDMA_FB1_BASE	LCD DMA Frame Buffer 1 Base Address Register	Section 3.14
50h	LCDDMA_FB1_CEILING	LCD DMA Frame Buffer 1 Ceiling Address Register	Section 3.15

3.1 LCD Revision Identification Register (REVID)

The LCD revision identification register (REVID) is shown in Figure 15 and described in Table 10.

Figure 15. LCD Revision Identification Register (REVID)

31	0
REV	
R-4C10 0100h	

LEGEND: R = Read only; -*n* = value after reset

Table 10. LCD Revision Identification Register (REVID) Field Descriptions

Bit	Field	Value	Description
31-0	REV	4C10 0100h	Peripheral Identification Number

3.2 LCD Control Register (LCD_CTRL)

The LCD control register (LCD_CTRL) contains the fundamental mode select bit for the LCD controller. The LCD_CTRL is shown in Figure 16 and described in Table 11.

Figure 16. LCD Control Register (LCD_CTRL)

31	16
Reserved	
R-0	

15	8	7	1	0
CLKDIV		Reserved		MODESEL
R/W-0		R-0		R/W-0

LEGEND: R/W = Read/Write; R = Read only; *-n* = value after reset

Table 11. LCD Control Register (LCD_CTRL) Field Descriptions

Bit	Field	Value	Description
31-16	Reserved	0	Reserved
15-8	CLKDIV	0-FFh	Clock Divisor. Value (from 0 to 255) is used to specify the frequency of the pixel clock (in Raster mode) or MCLK (in LIDD mode) based on the LCD_CLK frequency. Pixel clock frequency can range from LCD_CLK/2 to LCD_CLK/255 (CLKDIV = 0 or CLKDIV = 1 are not valid). MCLK can vary from LCD_CLK to LCD_CLK/255 (CLKDIV = 0 or CLKDIV = 1 sets MCLK = LCD_CLK).
7-1	Reserved	0	Reserved
0	MODESEL		LCD Mode Select
		0	LCD Controller in LIDD mode
		1	LCD Controller in Raster mode

The 8-bit clock divider (CLKDIV) field is used to select the frequency of the pixel clock. CLKDIV can generate a range of pixel clock frequencies from LCD_CLK/2 to LCD_CLK/255. The pixel clock frequency must be adjusted to meet the required screen refresh rate.

The refresh rate depends on:

- The number of pixels for the target display.
- Whether monochrome or color mode is selected.
- The number of pixel clock delays programmed at the beginning and end of each line.
- The number of line clocks inserted at the beginning and end of each frame.
- The width of the frame clock (LCD_VSYNC) signal in active mode or VSW line clocks inserted in passive mode.
- The width of the line clock (LCD_HSYNC) signal.

All of these factors alter the time duration from one frame transmission to the next. Different display manufacturers require different frame refresh rates, depending on the physical characteristics of the display. CLKDIV is used to alter the pixel clock frequency in order to meet these requirements. Pixel clock is used to synchronously signal the device to drive data to the LCD data pins, and to signal the output FIFO to latch the data from the pins. The frequency of the pixel clock for a set CLKDIV value or the required CLKDIV value to yield a target pixel clock frequency can be calculated using the following equation:

$$\text{LCD\_PCLK} = \frac{\text{LCD\_CLK}}{\text{CLKDIV}}$$

The pixel clock frequency is programmed taking into account the limitations shown in Table 12.

If CLKDIV equals 0 or 1, the effect is undefined. Dividing the pixel clock frequency by an odd number distorts the duty cycle.

Table 12. Pixel Clock Frequency Programming Limitations

Type of Screen	Output (In Bits)	Minimum Pixel Clock Divider
TFT 1,2,4,8 BPP	12 (1 pixel)	2
TFT 16 BPP	16 (1 pixel)	2
STN monochrome(4 output lines per panel)	4 (4 pixel)	4
STN monochrome(8 output lines per panel)	8 (8 pixel)	8
STN color	8 (2 2/3 pixel)	3

3.3 LCD Status Register (LCD_STAT)

The LCD status register (LCD_STAT) contains bits that signal status and error conditions to the processor. Each of the LCD status bits signals an interrupt request as long as the bit is set AND the interrupt enable for that bit is also set (see the LCD raster control and LCD DMA control registers for these enables). Writing a 1 to each bit clears it; once the bit is cleared, the interrupt is cleared. The LCD_STAT is shown in Figure 17 and described in Table 13.

Figure 17. LCD Status Register (LCD_STAT)

31											16
Reserved											
R-0											

15	10	9	8	7	6	5	4	3	2	1	0
Reserved		EOF1	EOF0	Rsvd	PL	FUF	Rsvd	ABC	SYNC	Rsvd	DONE
R-0		R/W-0	R/W-0	R-0	R/W-0	R/W-0	R-0	R/W-0	R/W-0	R-0	R/W-0

LEGEND: R = Read only; -*n* = value after reset

Table 13. LCD Status Register (LCD_STAT) Field Descriptions

Bit	Field	Value	Description
31-10	Reserved	0	Reserved
9	EOF1		End of Frame 1
		0	no end of frame 1 detected
		1	end of frame 1 detected
8	EOF0		End of Frame 0
		0	no end of frame 0 detected
		1	end of frame 0 detected
7	Reserved	0	Reserved
6	PL		Loaded Palette
		0	The palette is not loaded
		1	The palette is loaded
5	FUF		FIFO Underflow Status
		0	FIFO has not underrun
		1	LCD dither logic not supplying data to FIFO at a sufficient rate, FIFO has completely emptied and data pin driver logic has attempted to take added data from FIFO
4	Reserved	0	Reserved
3	ABC		AC Bias Count Status
		0	AC bias transition counter has not decremented to zero
		1	AC bias transition counter has decremented to zero, indicating that the LCD_AC_O line has transitioned the number of times that is specified by the ACB_I control bit field. Counter is reloaded with value in ACB_I but is disabled until the user clears ABC.
2	SYNC		Sync Lost
		0	normal
		1	Frame Synchronization Lost has occurred
1	Reserved	0	Reserved
0	DONE		Raster or LIDD Frame Done (shared; depends on whether Raster or LIDD mode enabled)
		0	Raster or DMA_to_LIDD engine is enabled
		1	Raster or DMA_to_LIDD disabled and the active frame has just completed

3.3.1 Frame Done (DONE)

When the LCD is disabled by clearing the LCD Raster Control enable bit (RASTER_EN = 0) in the LCD Raster Control Register, the LCD allows the current frame to complete before it is disabled. After the last set of pixels is clocked out onto the LCD data pins by the pixel clock, the LCD is disabled and DONE is set.

- DONE = 1 when the frame is complete.
- DONE = 0 as long as the frame is not complete.

The frame done (DONE) bit signals the frame is complete. It is cleared when the RASTER_EN bit is set to 1 (turned ON).

3.3.2 Frame Synchronization Lost (SYNC)

The frame synchronization lost (SYNC) bit is set if the LCD controller detects a frame synchronization error. A frame synchronization error happens when the LCD attempts to read what it believes to be the first word of the video buffer but cannot be recognized as such. This bit is cleared by disabling the LCD controller (RASTER_EN bit =0). This also resets the input FIFO in the DMA controller.

- SYNC = 1 when a frame synchronization lost occurred.
- SYNC = 0 as long as no frame synchronization error occurs.

3.3.3 AC-Bias Count Status (ABC)

The ac-bias count status (ABC) bit is set each time the ac-bias line transitions a particular number of times as specified by the ac-bias line transitions per interrupt (ACB_I) field in LCD Raster Timing Register 2. If ACB_I is programmed with a non-zero value, a counter is loaded with the value in ACB_I and is decremented each time the ac-bias line reverses state. When the counter reaches zero, the ABC bit is set that signals an interrupt request to the interrupt controller. The counter reloads using the value in ACB_I, but does not start to decrement again until you clear ABC by writing 0 to the LCD status register.

- ABC = 1 when the ac-bias transition counter ACB_I has decremented to 0
- ABC = 0 as long as ACB_I has not decremented to 0

3.3.4 FIFO Underflow Status (FUF)

The FIFO underflow status (FUF) bit is set when the input FIFO is completely empty and the LCD data pins driver logic attempts to fetch data from the FIFO. This bit is cleared by disabling the LCD controller (RASTER_EN = 0). This also resets the input FIFO in the DMA controller.

- FUF = 1 when the dithering logic is not supplying data to the FIFO at a sufficient rate.
- FUF = 0 as long as FIFO has not underrun.

3.3.5 Loaded Palette (PL)

The loaded palette (PL) bit is a read-only bit that is set after the LCD finished loading the palette into memory.

- PL = 1 when the palette is loaded.
- PL = 0 as long as the palette is not loaded.

In data-only (PL = 10) and palette-plus-data (PL = 00) modes, write 0 to clear the interrupt. However, in the palette only (PL = 01) mode, LCD must be turned off in order to reset/clear the interrupt. But in this particular mode, make sure not to turn off the LCD before getting the loading interrupt.

3.4 LCD LIDD Control Register (LIDD_CTRL)

The LCD LIDD control register (LIDD_CTRL) contains the polarity controls for LIDD output signals (to account for variety in the external LCD display/peripheral signal requirements), and the LIDD type select bits. These bits are not valid in Raster mode (when LCD control register bit 0 = 1). The LIDD_CTRL is shown in Figure 18 and described in Table 14.

NOTE: To activate DMA to drive LIDD interface, all other control bit-fields must be programmed before setting LIDD_DMA_EN = 1 and must also disable LIDD_DMA_EN bit when changing the state of any control bit within the LCD controller.

Figure 18. LCD LIDD Control Register (LIDD_CTRL)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R-0															

15	14	13	12	11	10	9	8
Reserved					DONE_INT_EN	DMA_CS0_CS1	LIDD_DMA_EN
R-0					R/W-0	R/W-0	R/W-0

7	6	5	4	3	2	1	0
CS1_E1_POL	CS0_E0_POL	WS_DIR_POL	RS_EN_POL	ALEPOL	LIDD_MODE_SEL		
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0		

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 14. LCD LIDD Control Register (LIDD_CTRL) Field Descriptions

Bit	Field	Value	Description
31-11	Reserved	0	Reserved
10	DONE_INT_EN		LIDD Frame Done Interrupt Enable
		0	Disable LIDD Frame Done interrupt
		1	Enable LIDD Frame Done interrupt (seen on LCD Status Reg bit 0)
9	DMA_CS0_CS1		CS0/CS1 Select for LIDD DMA writes
		0	DMA writes to LIDD CS0
		1	DMA writes to LIDD CS1
8	LIDD_DMA_EN		LIDD DMA Enable
		0	Deactivate DMA control of LIDD interface; DMA control is released upon completion of transfer of the current frame of data (LIDD Frame Done) after this bit is cleared. The MPU has direct read/write access to the panel in this mode
		1	Activate DMA to drive LIDD interface to support streaming data to "smart" panels. The MPU cannot access the panel directly in this mode
7	CS1_E1_POL		Chip Select 1/Enable 1 (Secondary) Polarity Control
		0	Do Not Invert Chip Select 1/Enable 1
		1	Invert Chip Select 1/Enable 1 Chip Select 1 is active low by default; Enable 1 is active high by default
6	CS0_E0_POL		Chip Select 0/Enable 0 (Primary) Polarity Control
		0	Do Not Invert Chip Select 0/Enable 0
		1	Invert Chip Select 0/Enable 0 Chip Select 0 is active low by default; Enable 0 is active high by default
5	WS_DIR_POL		Write Strobe/Direction Polarity Control
		0	Do Not Invert Write Strobe/Direction
		1	Invert Write Strobe/Direction Write Strobe/Direction is active low/write low by default
4	RS_EN_POL		Read Strobe/Enable Polarity Control
		0	Do Not Invert Read Strobe/Enable
		1	Invert Read Strobe/Enable Read Strobe is active low by default; Enable is active high by default

Table 14. LCD LIDD Control Register (LIDD_CTRL) Field Descriptions (continued)

Bit	Field	Value	Description
3	ALEPOL		Address Latch Enable (ALE) Polarity Control
		0	Do Not Invert ALE
		1	Invert ALE. ALE is active low by default
2-0	LIDD_MODE_SEL	0-7h	LIDD Mode Select. Selects type of LCD interface for the LIDD to drive. LIDD_MODE_SEL defines the function of LCD external pins as follows:

Pin	001b	011b	100b
LCD_PCLK	EN	RS	N/A
LCD_HSYNC	DIR	WS	DIR
LCD_VSYNC	ALE	ALE	ALE
$\overline{\text{LCD\_AC\_ENB\_CS}}$	CS0	CS0	E0
LCD_MCLK	CS1	CS1	E1

0	Sync MPU68
1h	Async MPU68
2h	Sync MPU80
3h	Async MPU80
4h	Hitachi (Async)
5h-7h	Reserved

3.5 LCD LIDD CS*n* Configuration Registers (LIDD_CS0_CONF and LIDD_CS1_CONF)

The LCD LIDD CS*n* configuration registers (LIDD_CS*n*_CONF) provides the capability to configure Write and Read Strobe timing parameters to meet a variety of interface timing requirements for the Chip Select 0 (Primary) device and Chip Select 1(Secondary) device, respectively. These values are in MCLK cycles; MCLK is divided down from LCD_CLK as defined by the CLKDIV field in the LCD control register. The LIDD_CS*n*_CONF is shown in Figure 19 and described in Table 15.

Figure 19. LCD LIDD CS*n* Configuration Register (LIDD_CS*n*_CONF)

31	27	26	21	20	17	16
W_SU		W_STROBE		W_HOLD		R_SU
R/W-0		R/W-1		R/W-1		R/W-0

15	12	11	6	5	2	1	0
R_SU		R_STROBE		R_HOLD		TA	
R/W-0		R/W-1		R/W-1		R/W-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 15. LCD LIDD CS*n* Configuration Register (LIDD_CS*n*_CONF) Field Descriptions

Bit	Field	Value	Description
31-27	W_SU	0-1Fh	Write Strobe Set-Up cycles. Field value defines number of MCLK cycles after Data Bus/Pad Output Enable, ALE, Direction bit and Chip Select 0 have been set up before the Write Strobe is asserted when performing a write access.
26-21	W_STROBE	0-3Fh	Write Strobe Duration cycles. Field value defines number of MCLK cycles that the Write Strobe is held active when performing a write access.
20-17	W_HOLD	0-Fh	Write Strobe Hold cycles. Field value defines number of MCLK cycles that the Data Bus/Pad Output Enable, ALE, Direction bit, and Chip Select 0 are held after the Write Strobe is deasserted when performing a write access.
16-12	R_SU	0-1Fh	Read Strobe Set-Up cycles. Field value defines number of MCLK cycles after Data Bus/Pad Output Enable, ALE, Direction bit and Chip Select 0 have been set up before the Read Strobe is asserted when performing a read access.
11-6	R_STROBE	0-3Fh	Read Strobe Duration cycles. Field value defines number of MCLK cycles that the Read Strobe is held active when performing a read access.
5-2	R_HOLD	0-Fh	Read Strobe Hold cycles. Field value defines number of MCLK cycles that the Data Bus/Pad Output Enable, ALE, Direction bit, and Chip Select 0 are held after the Read Strobe is deasserted when performing a read access.
1-0	TA	0-3h	Field value defines number of MCLK cycles between the end of one CS0 device access and the start of another CS0 device access unless the two accesses are both reads, in which case this delay is not incurred.

3.6 LCD LIDD CS*n* Address Read/Write Registers (LIDD_CS0_ADDR and LIDD_CS1_ADDR)

The LCD LIDD CS0 address read/write registers (LIDD_CS*n*_ADDR) are accessed by the processor to perform the address/index read or write operations on the CS0 and CS1 device respectively. Writing to LIDD_CS0_ADDR asserts CS0 and Address Latch Enable, which loads the ADR_INDX field of this register into the address generator of the peripheral device. Likewise, reading from LIDD_CS0_ADDR asserts CS0 and Address Latch Enable, which loads status information from the peripheral device into the ADR_INDX field of this register. Similarly writing to LIDD_CS1_ADDR asserts CS1 and Address Latch Enable, which loads the ADR_INDX field of this register into the address generator of the peripheral device. Likewise, reading from LIDD_CS1_ADDR asserts CS1 and Address Latch Enable, which loads status information from the peripheral device into the ADR_INDX field of this register. The LIDD_CS*n*_ADDR is shown in Figure 20 and described in Table 16.

Figure 20. LCD LIDD CS*n* Address Read/Write Register (LIDD_CS*n*_ADDR)

31	16
Reserved	
R-0	

15	0
ADR_INDX	
R/W-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 16. LCD LIDD CS*n* Address Read/Write Register (LIDD_CS*n*_ADDR) Field Descriptions

Bit	Field	Value	Description
31-16	Reserved	0	Reserved
15-0	ADR_INDX	0-FFFFh	Peripheral Device Address/Index value. On writes this field is loaded into the CS0 peripheral device's address generator On reads this field contains the CS0 peripheral device's status.

3.7 LCD LIDD CS*n* Data Read/Write Registers (LIDD_CS0_DATA and LIDD_CS1_DATA)

The LCD LIDD CS0 data read/write registers (LIDD_CS*n*_DATA) are accessed by the processor to perform the data read or write operations on the CS0 and CS1 device respectively . Writing to LIDD_CS0_DATA asserts CS0 and deasserts Address Latch Enable, which loads the DATA field of this register into the peripheral device. Likewise, reading from this register asserts CS0 and deasserts Address Latch Enable, which loads data from the peripheral device into the DATA field of this register. Similarly writing to LIDD_CS1_DATA asserts CS1 and deasserts Address Latch Enable, which loads the DATA field of this register into the peripheral device. Likewise, reading from LIDD_CS1_DATA asserts CS1 and deasserts Address Latch Enable, which loads data from the peripheral device into the DATA field of this register. The LIDD_CS*n*_DATA is shown in Figure 21 and described in Table 17.

Figure 21. LCD LIDD CS*n* Data Read/Write Register (LIDD_CS*n*_DATA)

31	16
Reserved	
R-0	
15	0
DATA	
R/W-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 17. LCD LIDD CS*n* Data Read/Write Register (LIDD_CS*n*_DATA) Field Descriptions

Bit	Field	Value	Description
31-16	Reserved	0	Reserved
15-0	DATA	0-FFFFh	Peripheral Device Data value. On writes this field is loaded into the CS0 peripheral device On reads this field contains the CS0 peripheral device's data.

3.8 LCD Raster Control Register (RASTER_CTRL)

The LCD raster control register (RASTER_CTRL) contains bit-fields that are used to control various functions within the Raster controller sub-module. The RASTER_CTRL is shown in Figure 22 and described in Table 18.

Figure 22. LCD Raster Control Register (RASTER_CTRL)

31							25	24
Reserved								STN_565
R-0								R/W-0
23	22	21	20	19				16
TFT_ALT_MAP	NIB_MODE	PLM		FIFO_DMA_DELAY				
R/W-0	R/W-0	R/W-0		R/W-0				
15			12	11	10	9		8
FIFO_DMA_DELAY				Reserved		MONO8B		RD_ORDER
R/W-0				R-0		R/W-0		R/W-0
7	6	5	4	3	2	1		0
TFT_STN	FUF_EN	SL_EN	PL_EN	DONE_EN	AC_EN	MONO_COLOR		RASTER_EN
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0		R/W-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 18. LCD Raster Control Register (RASTER_CTRL) Field Descriptions

Bit	Field	Value	Description
31-25	Reserved	0	Reserved
24	STN_565		12-Bit-Per-Pixel (5-6-5) Mode. This is only available in passive-color (STN) mode when 12 BPP is specified in the palette.
		0	Disabled: The lower 12 bits of pixel data is processed and output; i.e., “X X X X R3 R2 R1 R0 G3 G2 G1 G0 B3 B2 B1 B0” (where X is ignored by the Raster Controller).
		1	Enabled: Pixel data in the frame buffer is 16-bit but only 12 of the bits are processed and output; i.e. “R3 R2 R1 R0 X G3 G2 G1 G0 X X B3 B2 B1 B0 X” (where X is ignored by the Raster Controller).
			The data patterns above refer to frame buffer bits not output bits.
23	TFT_ALT_MAP		TFT Alternative Signal Mapping
		0	Output pixel data for 1, 2, 4, and 8 BPP will be right aligned on LCD_D[11:0]. For example, “R3 R2 R1 R0 G3 G2 G1 G0 B3 B2 B1 B0”.
		1	Output pixel data for 1, 2, 4, and 8 BPP will be converted to 5-6-5 format and transferred via LCD_D[15:0]. For example, “R3 R2 R1 R0 R3 G3 G2 G1 G0 G3 G2 B3 B2 B1 B0 B3”.
			The data patterns above refer to output bits not frame buffer bits.
22	NIB_MODE		Nibble Mode
		0	Nibble Mode disabled.
		1	Nibble Mode enabled.
			For 1, 2, and 4-BPP modes, this bit should be enabled. For 8, 12, and 16-BPP modes, this bit should be disabled.
21-20	PLM	0-3h	Palette Loading Mode
		0	Palette and data.
		1h	Palette only.
		2h	Data only.
		3h	Do not use.
19-12	FIFO_DMA_DELAY	0-FFh	FIFO DMA Request Delay. Encoded value used to specify the number of clocks the input FIFO DMA request should be disabled. The delay clock count starts after 16 words are loaded into the input FIFO. Delay Time = [(LCD Pixel Clock) × FIFO_DMA_DELAY]
11-10	Reserved	0	Reserved

Table 18. LCD Raster Control Register (RASTER_CTRL) Field Descriptions (continued)

Bit	Field	Value	Description
9	MONO8B		Mono 8-bit Mode
		0	LCD_D[3:0] is used to output four bits each LCD_PCLK.
		1	LCD_D[7:0] is used to output eight bits each LCD_PCLK.
			This bit is ignored in all other modes.
8	RD_ORDER		Raster Data Order Select
		0	Frame buffer data is ordered from least-to-most significant bit/nibble/byte/word/d-word.
		1	Frame buffer data is ordered from most-to-least significant bit/nibble/byte/word/d-word.
7	TFT_STN		TFT or STN Mode
		0	Passive (STN) display operation enabled.
		1	Active (TFT) display operation enabled.
6	FUF_EN		FIFO Underflow Interrupt Enable
		0	Disable the FIFO Underflow interrupt.
		1	Enable the FIFO Underflow interrupt.
5	SL_EN		Sync Lost Interrupt Enable
		0	Disable the Sync Lost interrupt.
		1	Enable the Sync Lost interrupt.
4	PL_EN		Palette Loaded Interrupt Enable
		0	Disable the Palette Loaded interrupt.
		1	Enable the Palette Loaded interrupt.
3	DONE_EN		Frame Done Interrupt Enable
		0	Disable the Frame Done interrupt.
		1	Enable the Frame Done interrupt.
2	AC_EN		AC Bias Count Interrupt Enable
		0	Disable the AC Bias Count interrupt.
		1	Enable AC Bias Count interrupt.
1	MONO_COLOR		LCD Monochrome or Color
		0	Enable Color display operation.
		1	Enable Monochrome display operation.
0	RASTER_EN		LCD Raster Controller Enable
		0	Disable the LCD Raster controller.
		1	Enable the LCD Raster controller.

3.8.1 LCD Raster Controller Enable (RASTER_EN)

NOTE: All other control bit-fields must be programmed before setting RASTER_EN = 1 and must also disable the LCD controller when changing the state of any control bit within the LCD controller.

The LCD Raster Controller enable (RASTER_EN) bit is used to enable and disable all LCD controller operation.

- When RASTER_EN = 0, the LCD controller is disabled.
- When RASTER_EN = 1, the LCD controller is enabled.

You can program the LCD control register (LcdControl) last, and configure all twenty-five bit fields at the same time via a word32 write to the register. If you clear RASTER_EN bit while the LCD controller is enabled, you can complete transmission of the current frame before being disabled. Completion of the current frame is signaled by the LCD controller to the DMA by setting the frame done (Done) bit within the LCD controller status register (see LCD Controller Status Register), which generates an interrupt request. If the LCD controller is disabled, the signals on pixel data [15:0] pins are set to 0 and the pixel clock, frame clock(vertical clock), line clock (horizontal clock), and ac-bias signals are set to their inactive state. This can be 0 or 1, depending on the inversions programmed in the LCD Raster Timing 2 register.

3.8.2 LCD Monochrome (MONO_COLOR)

The color/monochrome select (LcdBW) bit is used to determine whether the LCD controller operates in color or monochrome mode.

- When MONO_COLOR = 0:
  - Color mode is selected.
  - Palette entries are 12 bits wide, providing up to 4096 colors in active (TFT) mode and up to 3375 colors in passive (STN) color mode.
  - All three dither blocks are used (in passive mode only: TFT_STN = 0), one for each color component (R, G, B).
- When MONO_COLOR = 1:
  - Monochrome mode is selected.
  - Palette entries are 4 bits wide effective (15 levels of grayscale).
  - 4 or 8 data lines are enabled, according to the mono 8-bit mode (MONO8B).

Table 19. LCD Controller Data Pin Utilization for Mono/Color Passive/Active Panels

Color/Mono BPP	Passive/Active Panel	Pins
Mono 1,2,4	Passive	Pixel data[3:0]
Mono 8	Passive	Pixel data[7:0]
Color 1,2,4,8,12,16 (STN_565 = 1)	Passive	Pixel data[7:0]
Color 1,2,4,8,12	Active	Pixel data[11:0] or Pixel data[15:0] according to TFT_ALT_MAP bit in the LCD Raster Control Register (RASTER_CTRL)
Color 16	Active	Pixel data[15:0]

3.8.3 TFT_STN (TFT_STN)

The TFT_STN (TFT_STN) bit selects whether the LCD controller operates in passive (STN) or active (TFT) display control mode. When TFT_STN = 0, passive or STN mode is selected. LCD data flows from the frame buffer memory, via the LCD dedicated DMA channel, to the palette (the palette is bypassed for the 12 and 16 BPP modes), to the dithering logic and the output FIFO before being output on the LCD data pins. The clock and data pin behaviors is shown for the monochrome passive mode (Figure 23) and for the color passive mode (Figure 24).

Figure 23. Monochrome Passive Mode Pixel Clock and Data Pin Timing

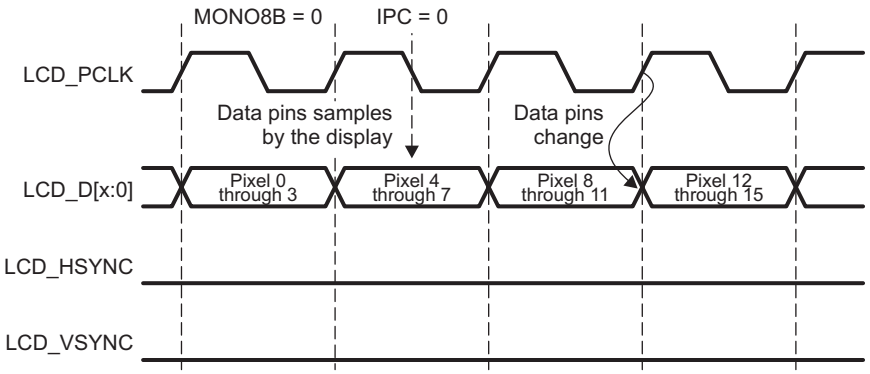
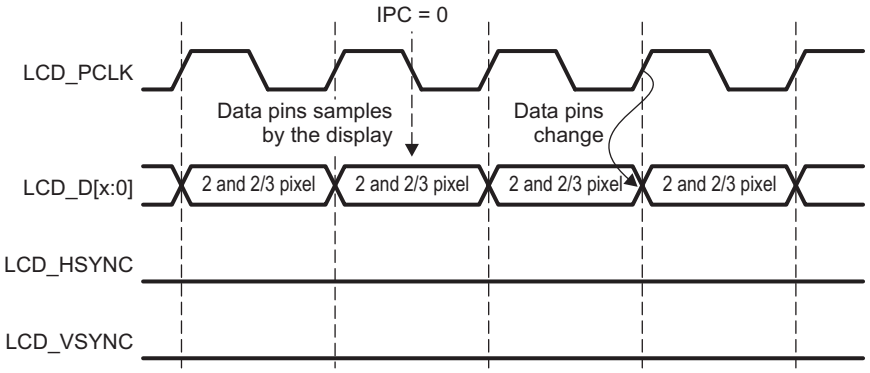
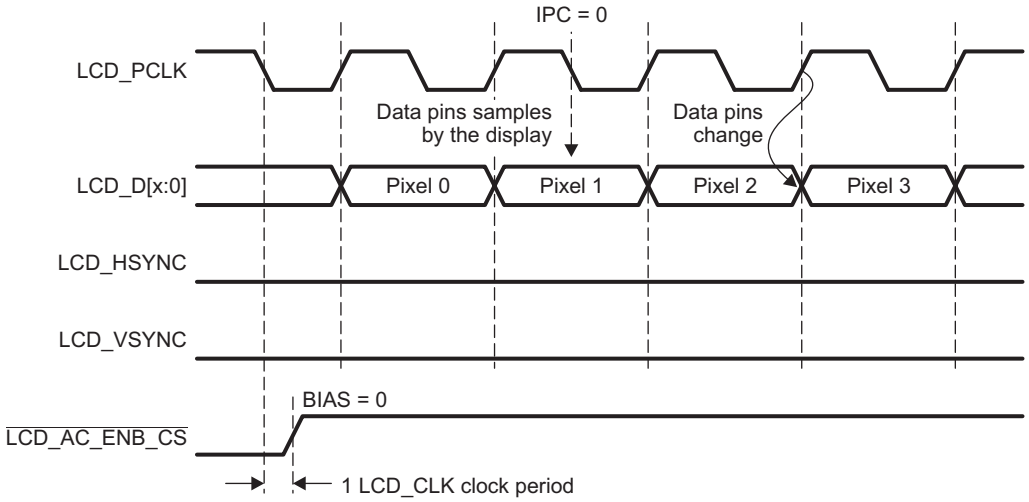


Figure 24. Color Passive Mode Pixel Clock and Data Pin Timing



When TFT_STN = 1, active or TFT mode is selected. Video data is transferred via the DMA from memory to the input FIFO, then is unpacked and used to select an entry from the palette (for 1, 2, 4, and 8 bits per pixel modes), just as in passive mode. The value read from the palette; however, bypasses both the LCD dither logic and the output FIFO to be output on the LCD data pins in TFT mode. The pixel size within the frame buffer is increased to 16 bits when 12- or 16-bit pixel encoding mode is enabled (BPP = 1xx). In TFT mode for 12 and 16 bits per pixel, palette entry is not selected. The clock and data pin behaviors is shown in Figure 25.

Figure 25. Active Mode Pixel Clock and Data Pin Timing



describes the clocks and data pin behaviors in active mode. The size of the pixel encoding is increased in TFT mode because the LCD dither logic is bypassed (the dither logic only supports 4 bits to encode each color component R, G, B that limits the pixel encoding size in passive mode). Increasing the size of the pixel representation allows a total of 64K colors to be addressed using an off-chip palette in conjunction with the LCD controller.

3.8.4 Mono 8 Bit Mode (MONO8B)

NOTE: MONO8B does not affect any of the color modes or TFT.

The mono 8-bit mode (MONO8B) bit selects whether four or eight data lines are used to output pixel data to the LCD screen.

- When MONO8B = 0, pixel data [3:0] is used to output four pixel values to the LCD panel at each pixel clock transition.
- When MONO8B = 1, pixel data [7:0] is used to output eight pixel values to the LCD panel at each pixel clock transition

3.8.5 FIFO DMA Request Delay (FIFO_DMA_DELAY)

The 8-bit FIFO DMA request delay (FIFO_DMA_DELAY) field is used to select the minimum number of LCD_CLK cycles to wait between the servicing of each DMA request issued by the LCD controller, sending an address to the input FIFO. The goal is to ensure enough bandwidth to other system accesses. A delay of FIFO_DMA_DELAY cycles is inserted every 16 words read from the input FIFO. This function is a concern only in 8 BPP mode, where the palette is 256 words. The FIFO_DMA_DELAY field needs to be set properly to avoid FIFO underflow during palette loading phase. When FDD = 00h, the FIFO DMA request delay function is disabled. This function is only used for palette loading.

3.8.6 Palette Loading (PLM)

The 2-bit palette loading field describes how the palette loading behaves when each new frame is loaded from memory.

- When PLM = 0, the data in the frame buffer represents the palette data and the picture data. Both palette and picture data are loaded.
- When PLM = 1 (palette-only mode), the data in the frame buffer just represents a new palette to be loaded. This data is loaded and placed into the palette. But be sure to turn off the LCD after getting the loading interrupt, or the LCD behavior would be unpredictable.
- When PLM = 2h (data loading mode), the data in the frame buffer only represents the picture data (data-only). This data is then used as an index (in the palette) or sent directly out. This mode assumes the palette was previously loaded. There is no need to keep loading the palette if it is not changing. As a matter of fact, in data-only mode, the BPP is fixed and can not change on the fly since the palette is not loaded at every frame.
- PLM = 3h is reserved.

3.8.7 TFT Alternate Signal Mapping (TFT_ALT_MAP)

This bit is relevant only if TFT_ALT_MAP = 1.

This bit field controls how the TFT pixel data is output. Via this feature, 12-BPP data can be output to all 16-bit LCD pins (this also applies to 1-, 2-, 4-, and 8- BPP). This feature allows you to switch BPP modes on the fly, duplicating the 12-bit output data across the 16 data lines if they are already hardwired to the 16 data lines.

Figure 26 shows how the four red, four green, and four blue bits are mapped to all pixel data [15-0] output pins when TFT_ALT_MAP = 1.

Figure 26. TFT Alternate Signal Mapping Output

Pins	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Data	R3	R2	R1	R0	R3	G3	G2	G1	G0	G3	G2	B3	B2	B1	B0	B3

When TFT_ALT_MAP = 0, the four red, four green, and four blue data are right-aligned on pixel data [11-0]. The upper pixel data [15-12] are cleared to 0. There is no duplication.

3.8.8 16 BPP STN Mode (STN_565)

The STN_565 bit is relevant only if TFT_STN = 0, but has no effect in 1-, 2-, 4- and 8-BPP modes. If STN_565 = 0, the frame buffer organization is in 12 BPP mode. In this mode, each color component is encoded in 4 bits, as shown in Figure 27.

Figure 27. 12-Bit STN Data in Frame Buffer

Pins	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Data	Data Ignored				R3	R2	R1	R0	G3	G2	G1	G0	B3	B2	B1	B0

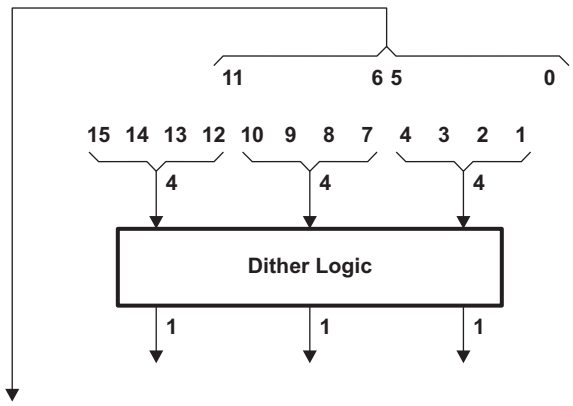
If STN_565 = 1 , the 16-bit STN mode is selected .The only difference from the 12 BPP mode is how the pixel data is organized in the frame buffer and which bits are sent to the dither logic. The 16-bit STN mode appears in frame buffer memory as shown in Figure 28.

Figure 28. 16-Bit STN Data in Frame Buffer

Pins	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Data	R4	R3	R2	R1	R0	G5	G4	G3	G2	G1	G0	B4	B3	B2	B1	B0

The 16-bit STN mode only sends 12 bits to the dithering logic as well as the 12-BPP STN mode. The LSB bit of the red component (bit 11), the two LSBs of green (bits 6 and 5), and the LSB of blue (bit 0) are not sent to the dithering logic, as shown in Figure 29.

Figure 29. 16-BPP STN Mode

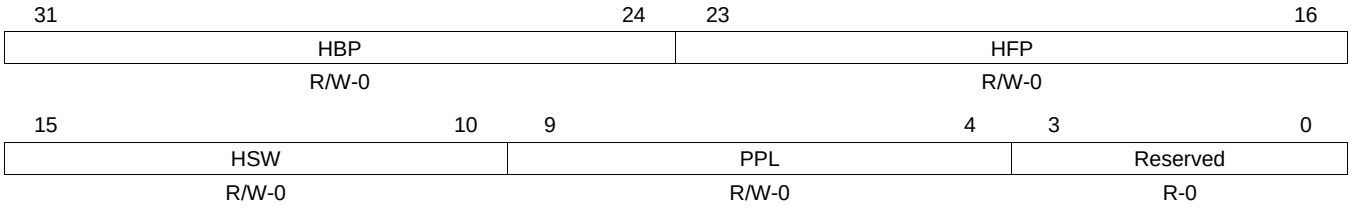


Note:
Red bit 11, green bits 6, 5 and blue bit 0 are not used as image data and are ignored. The 16-bit STN resolution is equal to the 12-bit STN resolution (3375 colors)

3.9 LCD Raster Timing Register 0 (RASTER_TIMING_0)

The LCD raster timing 0 register (RASTER_TIMING_0) contains four bit-fields that are used as modulus values for a collection of down counters, each of which performs a different function to control the timing of several of the LCD's pins. The RASTER_TIMING_0 is shown in Figure 30 and described in Table 20.

Figure 30. LCD Raster Timing Register 0 (RASTER_TIMING_0)



LEGEND: R = Read only; -*n* = value after reset

Table 20. LCD Raster Timing Register 0 (RASTER_TIMING_0) Field Descriptions

Bit	Field	Value	Description
31-24	HBP	0-FFh	Horizontal Back Porch. Encoded value (HBP + 1) is used to specify number of LCD_PCLKs to add to the beginning of a line transmission before the first set of pixels is output to the display (program to value minus one). Note that pixel clock is held in its inactive state during the beginning of line wait period in STN mode while it is active in TFT mode.
23-16	HFP	0-FFh	Horizontal Front Porch. Encoded value (HFP + 1) is used to specify number of LCD_PCLKs to add to the end of a line transmission before line clock is asserted (program to value minus one). Note that pixel clock is held in its inactive state during the end of line wait period in STN mode while it is active in TFT mode.
15-10	HSW	0-3Fh	Horizontal Sync Pulse Width. Encoded value (HSW + 1) is used to specify number of LCD_PCLKs to pulse the line clock at the end of each line (program to value minus one). Note that pixel clock is held in its inactive state during the generation of the line clock in STN mode while it is active in TFT mode.
9-4	PPL	0-3Fh	Pixels per Line. This value specifies the number of pixel transmissions per line . Number of pixels per Line = (PPL + 1) × 16
3-0	Reserved	0	Reserved

3.9.1 Pixels-Per-Line (PPL)

NOTE: PPL must be programmed to the value required minus 1 (for example, for a 640-pixel-per-line LCD panel, PPL = (640/16) - 1 = 40 - 1 = 39 = 27h).

The pixels-per-line (PPL) bit-field is used to specify the number of pixels in each line on the screen. The number of pixels per line = (PPL + 1) × 16, represents the screen width. PPL is a 6-bit value. Taking into account that the bottom 4 bits of this register are reserved, it is possible to support displays where the number of pixels-per-line ranges from 16-1024. PPL is used to count the correct number of pixel clocks that must occur before the line clock can be pulsed.

3.9.2 Horizontal Synchronization Pulse Width (HSW)

NOTE: The pixel clock does not transition during the line clock pulse in passive display mode, but it transitions in active display mode.

The 6-bit horizontal synchronization pulse width (HSW) field is used to specify the pulse width of the line clock in passive mode, or horizontal synchronization pulse in active mode. The line clock (or LCD_HSYNC) is asserted each time a line or row of pixels is output to the display and a programmable number of pixel clock delays have elapsed. When line clock is asserted, the value in HSW is transferred to a 6-bit down counter that uses the programmed pixel clock frequency to decrement. When the counter reaches zero, the line clock is negated. HSW can be programmed to generate a line clock pulse width ranging from 1–64 pixel clock periods (program to value required minus 1).

3.9.3 Horizontal Front Porch (HFP)

NOTE: The pixel clock does not transition during these dummy pixel clock cycles in passive display mode, but it transitions continuously in active display mode.

The 8-bit horizontal front porch (HFP) field is used to specify the number of dummy pixel clocks to insert at the end of each line or row of pixels before pulsing the line clock(or LCD_HSYNC) pin. Once a complete line of pixels is transmitted to the LCD driver, the value in HFP is used to count the number of pixel clocks to wait before pulsing the line clock. HFP generates a wait period ranging from 1–256 pixel clock cycles (program to value required minus 1).

3.9.4 Horizontal Back Porch (HBP)

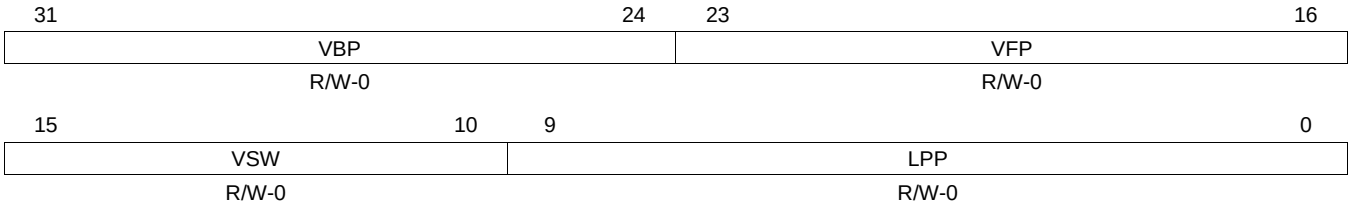
NOTE: The pixel clock does not transition during these dummy pixel clock cycles in passive display mode, but it transitions continuously in active display mode.

The 8-bit horizontal back porch (HBP) field is used to specify the number of dummy pixel clocks to insert at the beginning of each line or row of pixels. After the line clock(or LCD_HSYNC) for the previous line has been negated, the value in HBP is used to count the number of pixel clocks to wait before starting to output the first set of pixels in the next line. HBP generates a wait period ranging from 1–256 pixel clock cycles (program to value required minus 1).

3.10 LCD Raster Timing Register 1 (RASTER_TIMING_1)

The LCD raster timing 1 register (RASTER_TIMING_1) contains four bit-fields that are used as modulus values for a collection of down counters, each of which performs a different function to control the timing of several of the LCD's pins. The (RASTER_TIMING_1 is shown in Figure 31 and described in Table 21.

Figure 31. LCD Raster Timing Register 1 (RASTER_TIMING_1)



LEGEND: R = Read only; -*n* = value after reset

Table 21. LCD Raster Timing Register 1 (RASTER_TIMING_1) Field Descriptions

Bit	Field	Value	Description
31-24	VBP	0-FFh	Vertical Back Porch. Encoded value (VBP) used to specify number of line clock (LCD_HSYNC) periods to add to the beginning of a frame before the first set of pixels is output to the display. Note that line clock transitions during the insertion of the extra line clock periods.
23-16	VFP	0-FFh	Vertical Front Porch. Encoded value (VFP) used to specify number of line clock (LCD_HSYNC) periods to add to the end of each frame. Note that the line clock transitions during the insertion of the extra line clock periods.
15-10	VSW	0-3Fh	Vertical Synchronization Pulse Width. In TFT mode, encoded value (VSW + 1) defines the number of line clock (LCD_HSYNC) cycles to hold the frame clock (LCD_VSYNC) active. In STN mode, encoded value (VSW + 1) specifies the number of extra line clock (LCD_HSYNC) cycles to be inserted after the vertical front porch (VFP) period has elapsed.
9-0	LPP	0-3FFh	Lines per Panel. Encoded value (LPP + 1) used to specify number of lines per panel. It represents the total number of lines on the LCD.

3.10.1 Lines Per Panel (LPP)

NOTE: LPP must be programmed to the value required minus 1 (C7h for 200 lines per panel).

The lines per panel (LPP) bit-field is used to specify the number of lines or rows per LCD panel being controlled. It represents the total number of lines for the entire LCD display (the screen height). LPP is a 10-bit value, which represents between 1–1024 lines per panel. LPP is used to count the correct number of line clocks that must occur before the frame clock can be pulsed.

3.10.2 Vertical Synchronization Pulse Width (VSW)

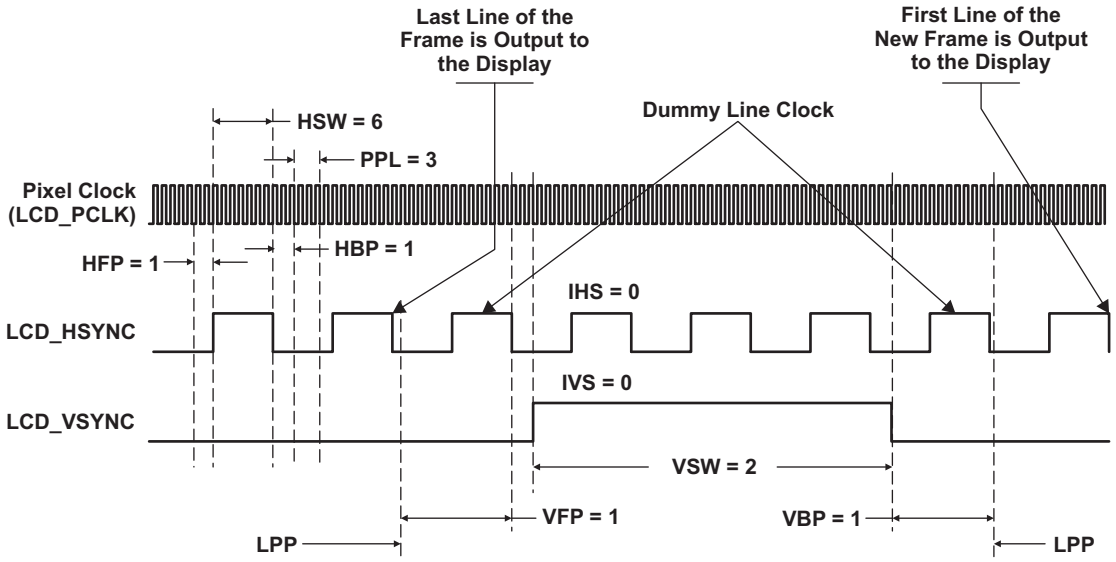
The 6-bit vertical synchronization pulse width (VSW) field is used to specify the pulse width of the vertical synchronization pulse in active mode or is used to add extra dummy line clock cycles between the vertical front porch and vertical back porch in passive mode.

3.10.2.1 Active Mode

NOTE: Remember that most of the parameters (HSW, HFP, PPL, HBP) must be programmed to value required minus 1.

In active mode (TFT_STN = 1), VSYNC is asserted each time the last line or row of pixels from the previous frame is output to the display and a programmable number of line clock delays (VFP) has elapsed. When the frame clock (LCD_VSYNC) is asserted, the value in VSW is transferred to a 6-bit down counter that uses the line clock frequency to decrement. When the counter reaches zero, the frame clock (LCD_VSYNC) is negated. VSW can be programmed to generate a vertical synchronization pulse width ranging from 1–64 line clock periods (program to value required minus 1—see Figure 32). The following frame starts after LCD_VSYNC is deasserted and a programmable number of line clock delays (VBP) has elapsed.

Figure 32. Vertical Synchronization Pulse Width (VSW) - Active Mode



3.10.2.2 Passive Mode

NOTE: The pixel clock does not transition during the whole dummy line clock periods that are inserted in passive mode before the frame pulse. The line clock does transition during the insertion of the dummy line clock cycles. VSW must be long enough to load the palette.

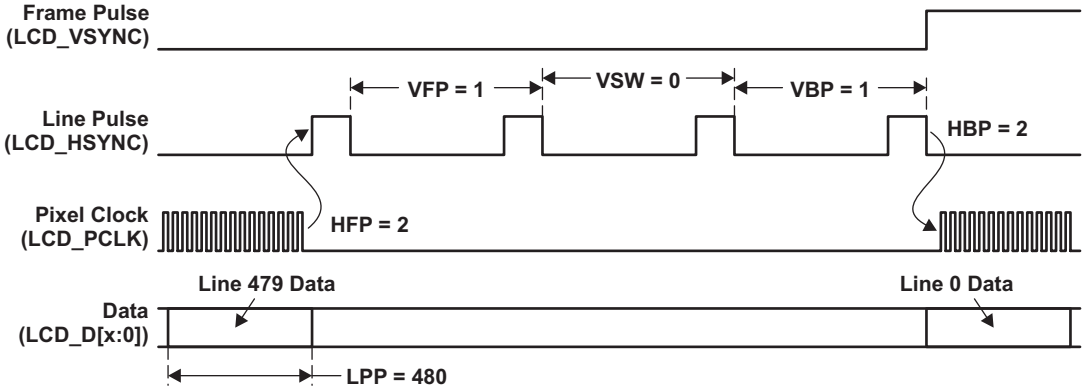
In passive mode (TFT_STN = 0), VSW does not affect the timing of the frame clock, but instead can be used to add extra line clock cycles between the end and beginning of frame line clock cycle counts. The total number of line clock cycles that are inserted between each frame is equal to the sum of the values in VFP, VSW and VBP. A counter is used to insert dummy line clock cycles between frames by first using the value in VFP, then VSW, then VBP. You must ensure that the sum of the values in the three fields is equal to the total number of line clock cycles that are needed between frames. The LCD controller frame clock pin is asserted on the rising-edge of the first pixel clock for each frame. The frame clock remains asserted for the remainder of the first line as pixels are output to the display, also during the assertion of the first line clock for the frame, and then negated on the rising-edge of the first pixel clock of the second line of each frame.

3.10.3 Vertical Front Porch (VFP)

NOTE: Remember that VSW must be programmed to value required minus 1.

The 8-bit vertical front porch (VFP) field is used to specify the number of line clocks to insert at the end of each frame. Once a complete frame of pixels is transmitted to the LCD display, the value in VFP is used to count the number of line clock periods to wait. After the count has elapsed the LCD_VSYNC signal is pulsed in active mode or extra line clocks are inserted as specified by the VSW bit-field in passive mode. VFP generates from 0–255 line clock cycles (see Figure 33).

Figure 33. Vertical Front Porch (VFP)

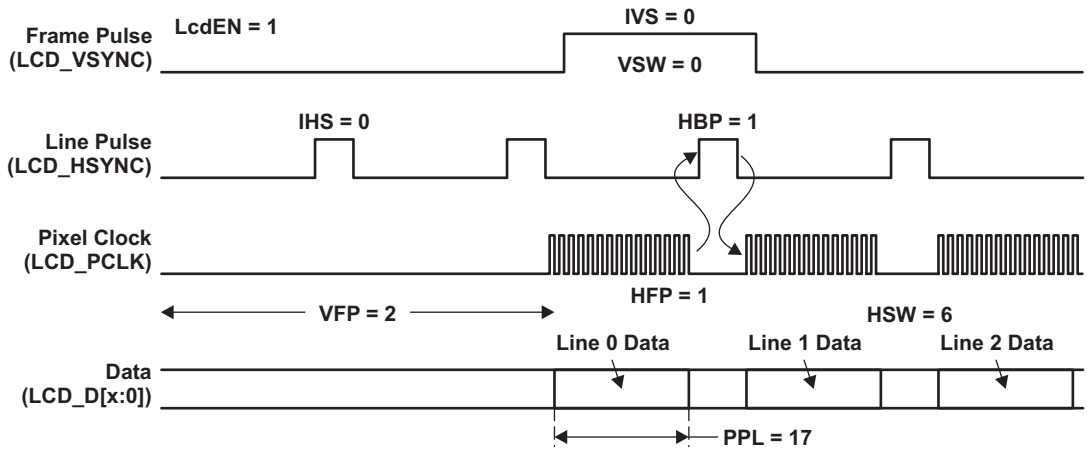


3.10.4 Vertical Back Porch (VBP)

NOTE: The line clock transitions during the generation of the VBP line clock wait periods. Note also that you must adjust the value of VBP appropriately such that enough line clock cycles are permitted to elapse; this allows the palette to be completely filled via the DMA, and allows a sufficient number of encoded pixel values to be input from the frame buffer, processed by the dither logic, then placed in the output FIFO, ready to be output to the LCD data lines.

The 8-bit vertical back porch (VBP) field is used to specify the number of line clocks (or LCD_HSYNC) to insert at the beginning of each frame. The VBP count starts just after the LCD_VSYNC signal for the previous frame has been negated for active mode, or the extra line clocks have been inserted as specified by the VSW bit-field in passive mode. After this has occurred, the value in VBP is used to count the number of line clock periods to insert before starting to output pixels in the next frame. VBP generates from 0–255 extra line clock cycles (see Figure 34).

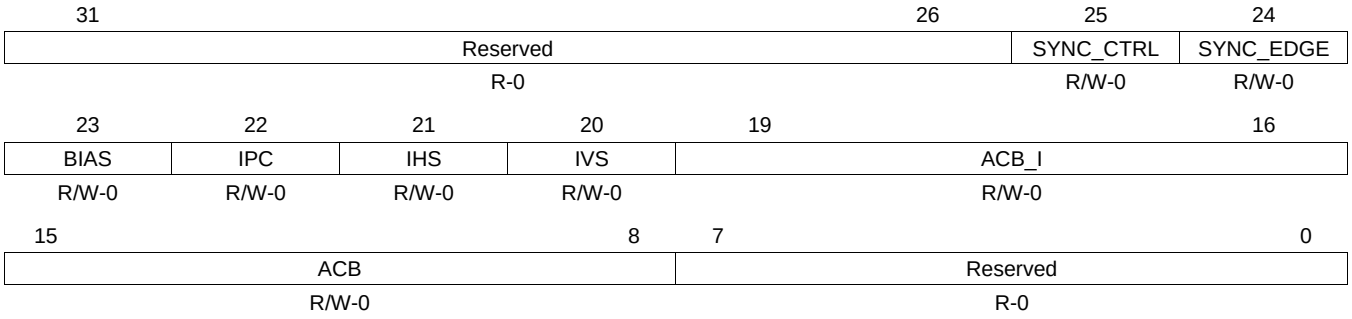
Figure 34. Vertical Back Porch (VBP)



3.11 LCD Raster Timing Register 2 (RASTER_TIMING_2)

LCD raster timing 2 register (RASTER_TIMING_2) contains bit-fields that are used to control various functions associated with the timing of the LCD controller. The RASTER_TIMING_2 is shown in Figure 35 and described in Table 22.

Figure 35. LCD Raster Timing Register 2 (RASTER_TIMING_2)



LEGEND: R/W = Read/Write; R = Read only; *-n* = value after reset

Table 22. LCD Raster Timing Register 2 (RASTER_TIMING_2) Field Descriptions

Bit	Field	Value	Description
31-26	Reserved	0	Reserved
25	SYNC_CTRL		Horizontal and Vertical Sync Control
		0	Inactive. SYNC_EDGE is ignored and the activation and deactivation of LCD_HSYNC and LCD_VSYNC will be defined by bit 22 below.
		1	Active. Allow SYNC_EDGE to define the LCD_PCLK edge (rising or falling) used to activate and deactivate LCD_HSYNC and LCD_VSYNC.
24	SYNC_EDGE		Horizontal and Vertical Sync Edge. The activation and deactivation of LCD_HSYNC and LCD_VSYNC will occur on the defined LCD_PCLK edge.
		0	Rising edge.
		1	Falling edge.
			SYNC_CTRL must be active in order to use this bit.
23	BIAS		Invert AC Bias
		0	$\overline{\text{LCD\_AC\_ENB\_CS}}$ is an active-high pulse.
		1	$\overline{\text{LCD\_AC\_ENB\_CS}}$ is an active-low pulse.
			In STN mode, the activation of this bit is ignored.
22	IPC		Invert Pixel Clock
		0	LCD Data (LCD_D[15:0]) is driven on the rising edge of LCD_PCLK, while LCD_VSYNC, LCD_HSYNC, and $\overline{\text{LCD\_AC\_ENB\_CS}}$ are driven on the falling edge.
		1	LCD Data (LCD_D[15:0]) is driven on the falling edge of LCD_PCLK, while LCD_VSYNC, LCD_HSYNC, and $\overline{\text{LCD\_AC\_ENB\_CS}}$ are driven on the rising edge.
			LCD_VSYNC and LCD_HSYNC may be altered as defined by bits 24 and 25 above.
21	IHS		Invert Line Clock
		0	LCD_HSYNC is an active high pulse.
		1	LCD_HSYNC is an active low pulse.
20	IVS		Invert Frame Clock
		0	LCD_VSYNC is an active high pulse.
		1	LCD_VSYNC is an active low pulse.
19-16	ACB_I	0-Fh	This value is used to specify the number of AC Bias ($\overline{\text{LCD\_AC\_ENB\_CS}}$) output transition counts before setting the AC bias interrupt bit in register LCD_STAT. This counter is stopped when the interrupt is set and remains stopped until the AC bias interrupt status is cleared. A value of zero will not produce an interrupt.

Table 22. LCD Raster Timing Register 2 (RASTER_TIMING_2) Field Descriptions (continued)

Bit	Field	Value	Description
15-8	ACB	0-FFh	AC Bias Pin Frequency. This value defines the number of Line Clock (LCD_HSYNC) cycles to count before transitioning signal LCD_AC_ENB_CS. This output may be used to periodically invert the polarity of the power supply in order to prevent a display DC charge build-up on the LCD panel. AC Bias Time Period = [2 × ((Line Clock) × (ABC))]
7-0	Reserved	0	Reserved

3.11.1 AC-Bias Pin Frequency (ACB)

NOTE: The 8-bit ac-bias frequency (ACB) field has no effect in active mode. This is due to the fact that the pixel clock transitions continuously in active mode ; the ac-bias line is used as an output enable signal. The ac-bias is asserted by the LCD controller in active mode; this occurs whenever pixel data is driven out to the data pins to signal to the display when it can latch pixels using the pixel clock.

The 8-bit ac-bias frequency (ACB) field is used to specify the number of line clock periods to count between each toggle of the ac-bias pin. After the LCD controller is enabled, the value in ACB is loaded to an 8-bit down counter, and the counter begins to decrement using the line clock. When the counter reaches zero it stops, the state of ac-bias pin is reversed, and the whole procedure starts again. The number of line clocks between each ac-bias pin transition ranges from 1–256 (program to value required minus 1). This line is used by the LCD display to periodically reverse the polarity of the power supplied to the screen to eliminate DC offset.

3.11.2 AC-Bias Line Transitions Per Interrupt (ACB_I)

The 4-bit ac-bias line transitions per interrupt (ACB_I) field is used to specify the number of line transitions to count before setting the ac-bias count status (ABC) bit in the LCD controller status register that signals an interrupt request. After the LCD controller is enabled, the value in ACB_I is loaded to a 4-bit down counter, and the counter decrements each time the ac-bias line state is inverted. When the counter reaches zero it stops, the ac-bias count (ABC) bit is set in the status register. Once ABC is set, the 4-bit down counter is reloaded with the value in ACB_I and is disabled until ABC is cleared. Once ABC is cleared by the CPU, the down counter is enabled, and again decrements each time the ac-bias line is flipped. The number of ac-bias line transitions between each interrupt request ranges from 0 to 15. Programming ACB_I = 0000 disables the ac-bias line transitions per interrupt function.

3.11.3 Invert VSYNC (IVS)

The invert VSYNC(IVS) bit is used to invert the polarity of the frame clock (VSYNC).

- When IVS = 1, the frame clock (VSYNC) is active low.
- When IVS = 0, it is active high.

3.11.4 Invert HSYNC (IHS)

The invert HSYNC(IHS) bit is used to invert the polarity of the line clock (HSYNC).

- When IHS = 1, the line clock (HSYNC) is active low.
- When IHS = 0, it is active high.

3.11.5 Invert Pixel Clock (IPC)

The invert pixel clock (IPC) bit is used to select the edge of the pixel clock that drives pixel data out onto the LCD data lines.

- When IPC = 1, data is driven onto the LCD data lines on the falling edge of the pixel clock.
- When IPC = 0, data is driven onto the LCD data lines on the rising edge of the pixel clock.

3.11.6 Invert Output Enable (BIAS)

NOTE: BIAS does not affect the ac-bias pin in passive display mode.

The invert output enable (BIAS) bit is used to select the active or inactive state of the output enable signal in active display mode. In this mode, the ac-bias pin is used as an enable that signals the device when data is actively being driven out using the pixel clock.

- When BIAS = 1, the ac-bias pin is active low. In active display mode, data is driven onto the LCD data lines on the programmed edge of the pixel clock when ac-bias pin is in its active state.
- When BIAS = 0, the ac-bias pin is active high.

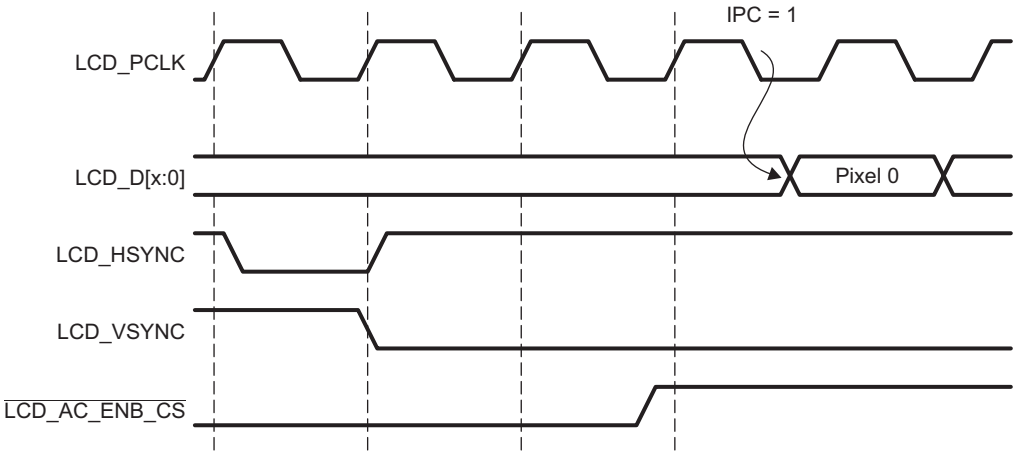
3.11.7 Horizontal and Vertical Sync Edge (SYNC_EDGE)

This bit determines whether the HSYNC/VSYNC is driven on the rising or falling edge of the pixel clock (see the SYNC_CTRL bit; SYNC_CTRL must be turned on first). By default, the LCD_HSYNC and LCD_VSYNC signals are driven on the falling edge of the pixel clock, and the pixel data is driven on the rising edge of pixel clock. However, if the invert pixel clock (IPC) bit is set to 1, then the LCD_HSYNC and LCD_VSYNC signals are driven on rising edge of pixel clock and pixel data is driven on falling edge. By setting the SYNC_EDGE bit and enabling it (SYNC_CTRL = 1), you can control on which edge the signals are driven.

In Figure 36:

- IPC = 1, pixel data is driven onto the LCD data lines on the falling edge of the pixel clock.
- SYNC_CTRL = 0, LCD_HSYNC and LCD_VSYNC signals are driven on opposite edges of the pixel clock from pixel data (=> rising edge). The rising edge or falling edge is determined by the IPC bit.

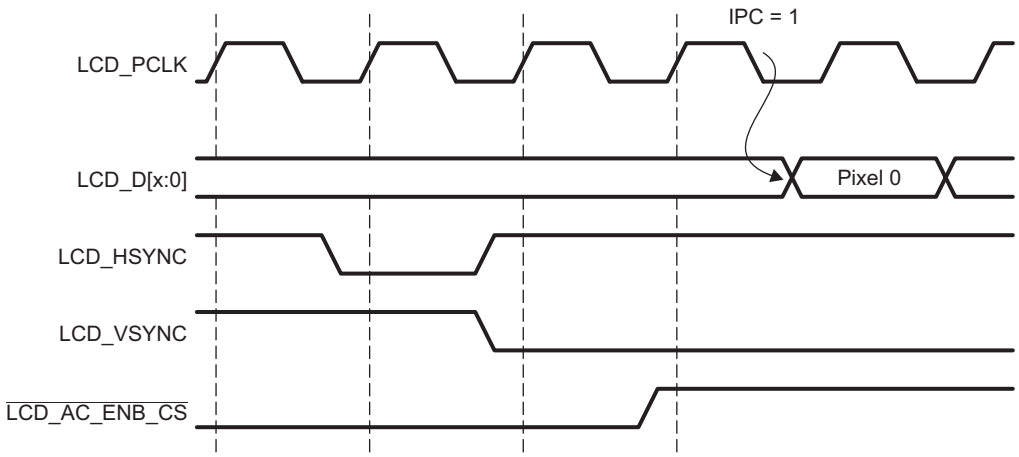
Figure 36. SYNC_CTRL = 0, IPC = 1 in TFT Mode



In Figure 37:

- IPC = 1, pixel data is driven on the falling edge of the pixel clock.
- SYNC_CTRL = 1, LCD_HSYNC and LCD_VSYNC signals are driven according to the SYNC_EDGE bit.
- SYNC_EDGE = 0, LCD_HSYNC and LCD_VSYNC signals are driven on the falling edge of the pixel clock.

Figure 37. SYNC_CTRL = 1, SYNC_EDGE = 0, and IPC = 1



3.11.8 Horizontal and Vertical Sync Control (SYNC_CTRL)

This bit enables/disables the possibility to make HSYNC and VSYNC programmable.

- When SYNC_CTRL = 1, HSYNC and VSYNC are driven according to the SYNC_EDGE bit.
- When SYNC_CTRL = 0, HSYNC and VSYNC are driven on opposite edges of the pixel clock from pixel data.

3.12 LCD Raster Subpanel Display Register (RASTER_SUBPANEL)

LCD raster subpanel display register (RASTER_SUBPANEL) displays only the first or last *n* lines of the panel and sends a fixed content for the others is supported with the LCD raster subpanel display register. For these others, there is no access to the frame buffer because the value stored in Default Pixel Data will be used. The RASTER_SUBPANEL is shown in Figure 38 and described in Table 23.

If LPPT is greater than the number of lines per panel then:

- If HOLS = 1: normal panel
- If HOLS = 0: panel with default data

LPPT has a minimum value = 2. DPD, LPPT, HOLS, and SPEN bit fields and bits are not considered if SPEN = 0.

Figure 38. LCD Raster Subpanel Display Register (RASTER_SUBPANEL)

31	30	29	28 26	25 16
SPEN	Rsvd	HOLS	Reserved	LPPT
R/W-0	R-0	R/W-0	R-0	R/W-0

15 4	3 0
DPD	Reserved
R/W-0	R-0

LEGEND: R = Read only; -*n* = value after reset

Table 23. LCD Raster Subpanel Display Register (RASTER_SUBPANEL) Field Descriptions

Bit	Field	Value	Description
31	SPEN		Subpanel Enable
		0	Disabled
		1	Enabled
30	Reserved	0	Reserved
29	HOLS		High or Low Signal. The field indicates the position of subpanel compared to the LPPT value.
		0	Low, below
		1	High, above
28-26	Reserved	0	Reserved
25-16	LPPT	0-3FFh	Line Per Panel Threshold. This field defines the number of lines to be refreshed (1 to 1024).
15-4	DPD	0-FFFh	Default Pixel Data. DPD defines the default value of the pixel data sent to the panel for the lines until LPPT is reached or after passing the LPPT.
3-0	Reserved	0	Reserved

3.12.1 Default Pixel Data (DPD)

The default pixel data (DPD) defines a default value, which is sent to the display in either the top or bottom region of the screen delimited by the LPPT threshold.

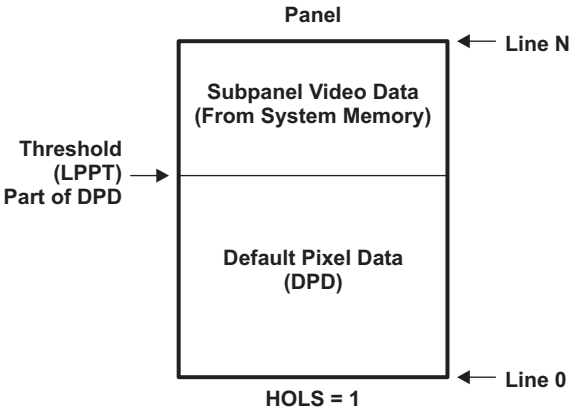
3.12.2 Line-per-Panel Threshold (LPPT)

The line-per-panel threshold bit-field delimits the screen portion filled with data fetched from the frame buffer (the subpanel) and the rest of the screen filled with default pixel data (DPD). Note that the LPPT line number points on a line filled with a DPD value when HOLS = 1, but on one filled with video data when HOLS = 0 (see Figure 39 and Figure 40).

3.12.3 High Or Low Signal (HOLS)

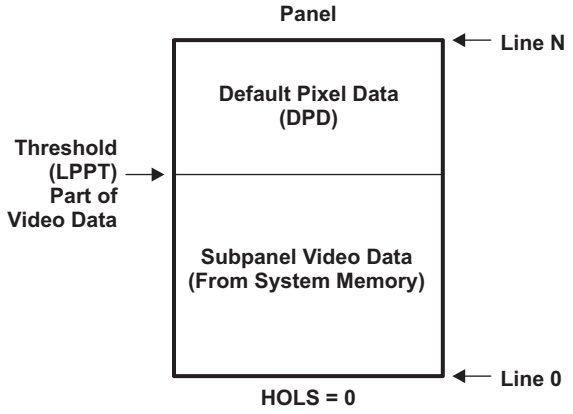
The HOLS bit indicates the position of the subpanel compared to the LPPT value. When HOLS = 1, the image from system memory is displayed above the threshold value. The threshold value is the line number where the DPD value begins to be displayed. The rest of the screen is filled with DPD value.

Figure 39. Subpanel Display: SPEN = 1, HOLS = 1



When HOLS = 0, the beginning of the screen is filled with DPD value until the LPPT excluded. From the LPPT line number, the rest of the screen (below LPPT) displays the image from system memory.

Figure 40. Subpanel Display: SPEN = 1, HOLS = 0



The bottom of the panel is line 0, and top line of the panel is Line N (where N is the number of lines-per-panel). For example, if you want to display four lines of video data at the bottom of the panel, the correct settings are HOLS = 0 and LPPT = 3. Here, the amount of video data to be transferred from the DMA_LCD channel is only four lines.

If the LPPT is above the number of LPP, then:

- When HOLS = 1: panel with default data (whole panel is filled with DPD value).
- When HOLS = 0: normal panel (whole panel is filled with video data from the frame buffer).

3.13 LCD DMA Control Register (LCDDMA_CTRL)

The LCD DMA control register (LCDDMA_CTRL) contains bits that control the LCD channel operation. The LCDDMA_CTRL is shown in Figure 41 and described in Table 24.

Figure 41. LCD DMA Control Register (LCDDMA_CTRL)

31	16
Reserved	
R-0	

15	11	10	8
Reserved		TH_FIFO_READY[1]	
R-0		R/W-0	

7	6	4	3	2	1	0
Reserved	BURST_SIZE		Reserved	EOF_INTEN	BIGENDIAN	FRAME_MODE
R-0	R/W-0		R-0	R/W-0	R/W-0	R/W-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

(1) This bit field is not supported and is Reserved on the C6743/C6745/C6747 DSPs and OMAP-L137 Applications Processor.

Table 24. LCD DMA Control Register (LCDDMA_CTRL) Field Descriptions

Bit	Field	Value	Description
31-11	Reserved	0	Reserved
10-8	TH_FIFO_READY	0-7h	DMA FIFO threshold. The input FIFO becomes ready so that the Raster controller can start reading its content only when the number of dwords (1 dword is 4 bytes) specified by TH_FIFO_READY have been loaded by the DMA from the frame buffer to the input FIFO. **This bit field is not supported and is Reserved on the C6743/C6745/C6747 DSPs and OMAP-L137 Applications Processor.**
		0	8 dwords
		1h	16 dwords
		2h	32 dwords
		3h	64 dwords
		4h	128 dwords
		5h	256 dwords
		6h	512 dwords
		7h	Reserved
7	Reserved	0	Reserved
6-4	BURST_SIZE	0-7h	Burst Size setting for DMA transfers (all DMA transfers are 32 bits wide)
		0	Burst size of 1
		1h	Burst size of 2
		2h	Burst size of 4
		3h	Burst size of 8
		4h	Burst size of 16
		5h-7h	Reserved
3	Reserved	0	Reserved
2	EOF_INTEN		End-of-Frame Interrupt Enable. Setting this bit allows the end-of-frame 0 or 1 status bits in the LCD status register to trigger an interrupt.
		0	End-of-frame 0/1 interrupt disabled
		1	End-of-frame 0/1 interrupt enabled
1	BIGENDIAN		Big Endian Enable. Use this bit when the processor is operating in Big Endian mode and writes to the frame buffer(s) are less than 32 bits wide; in this scenario only, change the byte alignment for data coming into the FIFO from the frame buffer(s).
		0	Big Endian data reordering disabled
		1	Big Endian data reordering enabled

Table 24. LCD DMA Control Register (LCDDMA_CTRL) Field Descriptions (continued)

Bit	Field	Value	Description
0	FRAME_MODE		Frame Mode
		0	One frame buffer (FB0 only) used.
		1	Two frame buffers used; DMA ping-pongs between FB0 and FB1 in this mode.

3.14 LCD DMA Frame Buffer *n* Base Address Registers (LCDDMA_FB0_BASE and LCDDMA_FB1_BASE)

The LCD DMA frame buffer *n* base address register (LCDDMA_FB*n*_BASE) contains the start address for frame buffer *n*, specified in 32-bit words. The LCDDMA_FB*n*_BASE is shown in Figure 42 and described in Table 25.

Figure 42. LCD DMA Frame Buffer *n* Base Address Register (LCDDMA_FB*n*_BASE)

31	0
FB*n*_BASE	
R/W-0	

LEGEND: R/W = Read/Write; -*n* = value after reset

Table 25. LCD DMA Frame Buffer *n* Base Address Register (LCDDMA_FB*n*_BASE) Field Descriptions

Bit	Field	Value	Description
31-0	FB*n*_BASE	0-FFFF FFFFh	Frame Buffer *n* Base Address pointer. Note: The 2 LSBs are hardwired to 00b.

3.15 LCD DMA Frame Buffer *n* Ceiling Address Registers (LCDDMA_FB0_CEILING and LCDDMA_FB1_CEILING)

The LCD DMA frame buffer *n* ceiling address register (LCDDMA_FB*n*_CEILING) contains the ending address for frame buffer *n*, specified in 32-bit words. The LCDDMA_FB*n*_CEILING is shown in Figure 43 and described in Table 26.

Figure 43. LCD DMA Frame Buffer *n* Ceiling Address Register (LCDDMA_FB*n*_CEILING)

31	0
FB*n*_CEIL	
R/W-0	

LEGEND: R/W = Read/Write; -*n* = value after reset

Table 26. LCD DMA Frame Buffer *n* Ceiling Address Register (LCDDMA_FB*n*_CEILING) Field Descriptions

Bit	Field	Value	Description
31-0	FB*n*_CEIL	0-FFFF FFFFh	Frame Buffer *n* Ceiling Address pointer. Note: The 2 LSBs are hardwired to 00b.

Appendix A Revision History

Table 27 lists the changes made since the previous version of this document.

Table 27. Document Revision History

Reference	Additions/Modifications/Deletions
Table 3	Changed LCD_D[x] bus width value for 6800 Family and 8080 Family.
Section 2.4	Changed second sentence in seventh paragraph.
	Added eighth paragraph.
	Deleted subsection 2.4.1 LIDD Controller Timing. Subsequent figures renumbered.
Table 4	Changed Description of $\overline{\text{LCD\_AC\_ENB\_CS}}$ for Active (TFT) Color.
Section 2.5.2	Changed first and second sentences in first paragraph.
Section 2.5.6	Deleted subsection 2.5.7 Raster Controller Timing. Subsequent figures renumbered.

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

TI products are not authorized for use in safety-critical applications (such as life support) where a failure of the TI product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Buyers represent that they have all necessary expertise in the safety and regulatory ramifications of their applications, and acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of TI products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by TI. Further, Buyers must fully indemnify TI and its representatives against any damages arising out of the use of TI products in such safety-critical applications.

TI products are neither designed nor intended for use in military/aerospace applications or environments unless the TI products are specifically designated by TI as military-grade or "enhanced plastic." Only products designated by TI as military-grade meet military specifications. Buyers acknowledge and agree that any such use of TI products which TI has not designated as military-grade is solely at the Buyer's risk, and that they are solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI products are neither designed nor intended for use in automotive applications or environments unless the specific TI products are designated by TI as compliant with ISO/TS 16949 requirements. Buyers acknowledge and agree that, if they use any non-designated products in automotive applications, TI will not be responsible for any failure to meet such requirements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		**Applications**	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DLP® Products	www.dlp.com	Communications and Telecom	www.ti.com/communications
DSP	dsp.ti.com	Computers and Peripherals	www.ti.com/computers
Clocks and Timers	www.ti.com/clocks	Consumer Electronics	www.ti.com/consumer-apps
Interface	interface.ti.com	Energy	www.ti.com/energy
Logic	logic.ti.com	Industrial	www.ti.com/industrial
Power Mgmt	power.ti.com	Medical	www.ti.com/medical
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
RFID	www.ti-rfid.com	Space, Avionics & Defense	www.ti.com/space-avionics-defense
RF/IF and ZigBee® Solutions	www.ti.com/lprf	Video and Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless-apps

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265