DLPC3430, DLPC3432, DLPC3433, DLPC3435 and DLPC3438

Programmer's Guide

Literature Number: DLPU020D
July 2014–Revised April 2020

Contents

DLPC3430, DLPC3432, DLPC3433, DLPC3435, and DLPC3438 Software Programmer's Guide

This is the programmer's guide for the DLP3430 and DLPC3435 controllers (used with the DLP2010 DMD), the DLPC3432 controller (used with the DLP230GP DMD), and the DLP3433 and DLPC3438 controllers (used with the DLP3010 DMD). This guide primarily discusses the I²C interface of these controllers. For additional information please visit the desired device product folder on ti.com.

Trademarks

DLP® Pico, LightCrafter are trademarks of Texas Instruments.
DLP is a registered trademark of Texas Instruments.
Arm, Cortex are registered trademarks of Arm Limited.

Introduction

1.1 System Overview

A typical TI DLP® Pico™ chipset consists of the controller, the PMIC, and the DMD. The DMD and PMIC are controlled by the DLPC343x controller. An example system is shown in Figure 1-1. The controller communicates with the outside world with I²C commands.

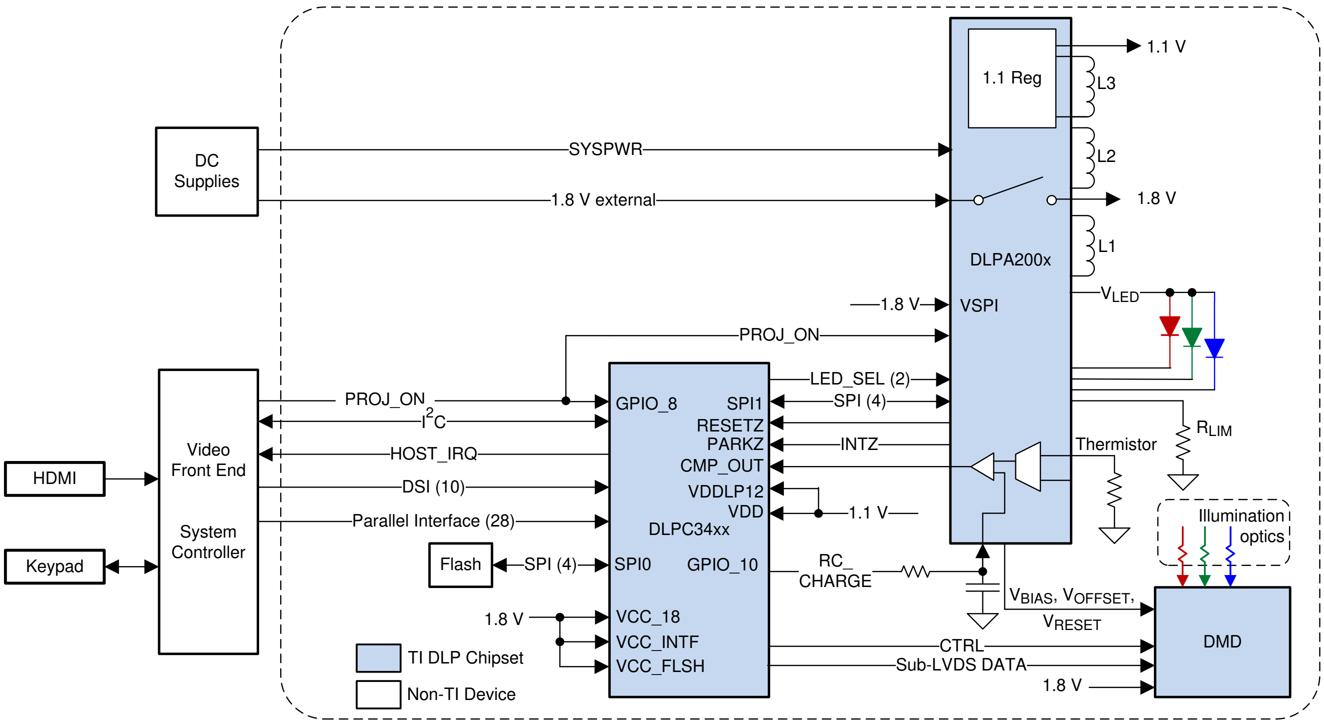


Figure 1-1. Typical Simplified Application Schematic

1.2 Software Overview

The DLPC343x controller contains an Arm® Cortex®-M3 processor with additional functional blocks to enable video processing and control. TI provides software as a firmware image. The firmware consists of the main application code (used by the Arm processor) along with other configuration and operational data required by the system for normal operation. The controller and its accompanying DLP chipset components require this proprietary software to operate.

The firmware must be programmed into the SPI flash memory. The DLPC343x controller loads the main application into the Arm processor which periodically accesses the operational data. The available controller functions depend on the firmware version installed. Different firmware is required for different chipset combinations (such as when using different PMIC devices). Visit the applicable controller's product folder on ti.com, visit the DLP Pico Firmware Selector, or contact TI for the latest firmware.

1.2.1 I²C Overview

The protocol used in communicating information to DLPC343x controller consist of a serial data bus conforming to the Philips I^2C specification. The controller can be configured at runtime by using these I^2C commands. The DLPC343x behaves as an I^2C slave operating at up to 100 kHz.

1.2.2 I²C Transactions

Since all I^2C commands are processed by software, only one type of I^2C transaction is supported. This transaction type is shown in Table 1-1 for both writes and reads. The I^2C interface supports variably-sized transactions (for example, a one byte transaction or a nine byte transaction) to match the commands discussed later in this document.

Table 1-1. I²C Write and Read Transactions

Transaction		Address(1)	Sub-Address(2)	Remaining Data Bytes(3)
Write	Size	8-bits	8-bits	8-bit parameter bytes (0 → N)
	Value	36h (or 3Ah)	Command value	Parameter values
Read Request	Size	8-bits	8-bits	8-bit parameter bytes (0 → N)
	Value	36h (or 3Ah)	Command value	Parameter values
Read Response	Size	8-bits		8-bit parameter bytes (0 → N)
	Value	37h (or 3Bh)		Parameter values

(1) The address corresponds to the chip address of the controller. The address is dependant upon the firmware image with the default value of 36h.
(2) The subaddress will correspond to a TI command.
(3) The data (if present) will correspond to any required command parameters.

1.2.3 Data Flow Control

While the I^2C interface inherently supports flow control by holding the clock, this is not sufficient for all transactions (for example, sequence and some other updates). In this case, the host software should use the *Read Short Status* to determine if the system is busy.

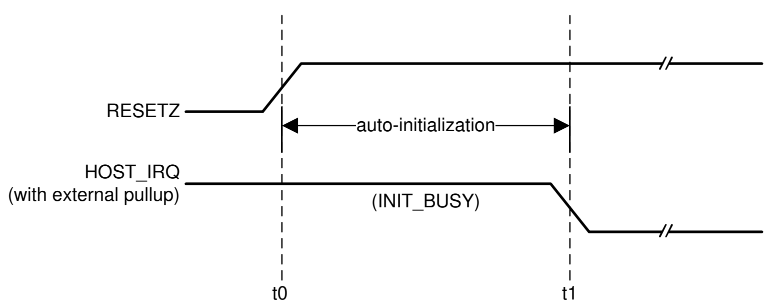
System Initialization

2.1 Boot ROM

The DLPC343x employs a boot ROM and associated boot software. This resident boot code consists of the minimum code necessary to complete the program loading from flash to internal RAM. Therefore, an external flash device is required to store the device firmware.

2.2 Device Startup

- The HOST_IRQ signal is provided to indicated when the system has completed auto-initialization.
- While reset is applied, HOST_IRQ is tri-stated (an external pullup resistor pulls the line high).
- HOST_IRQ remains tri-stated (pulled high externally) until the boot process completes. While the signal is pulled high, this indicates that the controller is performing boot-up and auto-initialization.
- As soon as possible after the controller boots-up, the controller drives HOST_IRQ to a logic high state to indicate that the controller is continuing to perform auto-initialization (no real state changes occur on the external signal).
- The software sets HOST_IRQ to a logic low state at the completion of the auto-initialization process. At the falling edge of the signal, the initialization is complete.
- The DLPC34xx controller is ready to receive commands through I^{2}C or accept video over the DSI or the parallel interface only after auto-initialization is complete.
- The controller initialization typically completes (HOST_IRQ goes low) within 500 ms of RESETZ being asserted. However, this time may vary depending on the software version and the contents of the user configurable auto initialization file.



t0: rising edge of RESETZ; auto-initialization begins

t1: falling edge of HOST_IRQ; auto-initialization is complete

Figure 2-1. HOST_IRQ Timing

I^2C Commands

3.1 General Operation

3.1.1 Write Single Buffer Mode (01h)

3.1.1.1 Write

This command specifies whether the controller operates in regular double buffer mode or in a single buffer mode.

3.1.1.2 Write Parameters

Figure 3-1 describes the command parameters.

Figure 3-1. Write Parameters

MSB	**Byte 1**						**LSB**
b7	b6	b5	b4	b3	b2	b1	b0

b(7:1) Reserved

b(0) Single Buffer Mode:
- 0: Disable
- 1: Enable

Normally the DLPC343x controller operates using a double buffered architecture. This command enables the controller to operate using a single buffer in order to reduce latency. A single buffer may introduce video artifacts and it is up to the user to determine if this mode is acceptable for their operation. Note: video artifacts are not the traditional full-frame tearing since the DLPC343x display uses a time sequential display approach.

To change (enable or disable) the single buffer mode, this command must be followed by the *Write Input Source Select* command. Single buffer mode is only active if enabled and when using external video input or test patterns. If a splash screen is loaded while single buffer bode is enabled, the controller will return to using a double buffer. When the input source returns to a test pattern or external video, the controller will return to using a single buffer (assuming the user didn't disable it).

Single buffer mode must not be enabled when using 3D video. Image freeze is not supported when Single Buffer Mode is enabled.

3.1.2 Read Single Buffer Mode (02h)

3.1.2.1 Read

This command reads the state of the single buffer mode for the display module.

3.1.2.2 Read Parameters

This command has no command parameters.

3.1.2.3 Return Parameters

Figure 3-2 describes the return parameters.

Figure 3-2. Return Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:1) Reserved

b(0) Single Buffer Mode
- 0: Disabled
- 1: Enabled

3.1.3 Write Idle Mode Select (03h)

3.1.3.1 Write

This command specifies if the DLPC343x controller enters a low-power mode when the input image isn't changing.

3.1.3.2 Write Parameters

Figure 3-3 describes the command parameters.

Figure 3-3. Write Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:2) Reserved

b(1) Manual Idle Mode:
- 0: Disable
- 1: Enable

b(0) Reserved

Manual Idle Mode is intended to provide power saving functionality by putting the controller into a low-power mode while the input image isn't changing. Therefore, when the Manual Idle Mode command is sent, the output image is frozen and new input images are ignored. To resume normal operation, the user must send the command to disable Manual Idle Mode.

When Manual Idle Mode is enabled, I^2C commands sent to the controller may not work as the controller is in a low power state. To enable full I^2C command functionality, Manual Idle Mode must be disabled

The Manual Idle Mode enable command should be sent in the Vertical Back Porch at least 7ms before the last VSYNC. The command will then take effect at the next VSYNC. The Manual Idle Mode disable command should be sent at least 7ms before the first VSYNC.

Manual idle mode is not supported when in curtain mode or when Single Buffer Mode is enabled.

Note: If using CAIC, six complete input frames are required to fully converge the algorithm. If fewer than six images are sent, the last image received will be displayed when Manual Idle Mode is enabled, even if the CAIC algorithm is not fully converged

3.1.4 Read Idle Mode Select (04h)

3.1.4.1 Read

This command reads the state of the idle mode for the display module.

3.1.4.2 Read Parameters

This command has no command parameters.

3.1.4.3 Return Parameters

Figure 3-4 describes the return parameters.

Figure 3-4. Return Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:2)	Reserved
b(1)	Idle Mode Select: • 0: Disabled • 1: Manual Idle Mode Enabled
b(0)	Reserved

3.1.5 Write Input Source Select (05h)

3.1.5.1 Write

This command selects the image input source for the display module.

3.1.5.2 Write Parameters

Figure 3-5 describes the command parameters.

Figure 3-5. Write Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:2)	Reserved
b(1:0)	Input Source: • 0h: External Video Port • 1h: Test Pattern Generator • 2h: Splash Screen • 3h: Reserved

---

NOTE: When selecting the external video port, there is a set of associated commands applicable only to this source selection. These associated commands are the *Write External Input image Size* and the *Write External Video Source Format Select*.

When selecting the test pattern generator (TPG), only one associated command is applicable to this source selection. This associated command is the *Write Test Pattern Select* command.

When selecting the splash screen, only two associated commands are applicable to this source selection. These associated commands are the *Write Splash Screen Select* and *Write splash Screen Execute* commands.

These associations are also shown in Table 3-1.

Table 3-1. Source Specific Associated Commands

Source Specific Associated Commands	Input Source Select Options		
	External Video Port	Test Pattern Generator	Splash Screen (1)
Write External Video Source Format Select	Only	N/A	N/A
Write External Input Image Size	Only	N/A	N/A
Write Test Pattern Select	N/A	Only	N/A
Write Splash Screen Select	N/A	N/A	Only
Write Splash Screen Execute	N/A	N/A	Special

(1) The *Write Splash Screen Execute* command is special in that there is no maintained state or history. Thus this command has no settings to be stored and reused by the system.

These commands (other than *Write Splash Screen Execute*) describe the characteristics of their associated source, and once these settings are defined the system stores them. Afterwards, each time an input source selection is made (using the *Write Input Source Select* command), the system remembers the settings described by the commands associated with the selected source, and automatically applies them. The user only needs to send these associated commands when the source is first defined, or when the source characteristics for that port must be changed. The appropriate associated commands must be updated when source characteristics change.

The user can send source-associated commands every time they make an input source selection. The source associated commands should be sent prior to sending the *Write Input Source Select* command. When source-associated commands are sent when that source is not active, the controller software saves the new settings, but does not execute these commands. When that source becomes active (via the *Write Input Source Select* command), the controller applies these new settings, as in the following example:

1. The user sends the following commands (active input source = test pattern generator):
   - *Write image Freeze* = freeze (to hide transition artifacts)
   - *Write External Video Source Format Select* (settings stored, command not executed)
   - *Write External Input Image Size* (settings stored, command not executed)
   - *Write Input Source Select* = external port (see step 2 below)
   - *Write Image Freeze* = unfreeze
2. When the *Write Input Source Select* command is received, the software applies the settings from these external video port-associated commands:
   - *External Video Source Format Select*
   - *External input Image Size*

If source-associated commands are sent for a source that is already active, the controller software executes these commands when received, as in the following example:

- The user sends the following commands (active input source = external video port):
  – *Write Image Freeze* = freeze (to hide transition artifacts)
  – *Write external Video Source Format Select* (command executed)

– *Write Image Freeze* = unfreeze

The rest of the commands that apply to image setup have settings applicable across all source selections, and typically remain the same across the three input source selections. A few examples are *Write Display Size* and *Write Display Image Orientation*. A representative list of these commands is shown in Table 3-2.

Table 3-2. Common Commands

Common Commands	Input Source Select Options		
	External Video Port	Test Pattern Generator	Splash Screen
Write Image Crop	Common	Common	Common
Write Display Image Size	Common	Common	Common
Write Keystone Correction Control	Common	Common	Common
Write Display Image Orientation	Common	Common	Common
Write Display Image Curtain	Common	Common	Common
Write Look Select	Common	Common	Common
Write Local Area Brightness Boost Control	Common	Common	Common
Write CAIC Image Processing Control	Common	Common	Common

NOTE: The user is required to specify the active data size for all external input sources, using the *Write Input Image Size* command.

NOTE: When a test pattern is selected, it is generated at the resolution of the DMD, modified by the settings specified by the *Write Image Crop* command, and displayed at the resolution specified by the *Write Display Size* command.

NOTE: The user should see the *Write Image Freeze* command for information on hiding on-screen artifacts when selecting an input source.

3.1.6 Read Input Source Select (06h)

3.1.6.1 Read

This command reads the state of the image input source for the display module.

3.1.6.2 Read Parameters

This command has no command parameters.

3.1.6.3 Return Parameters

Figure 3-6 describes the return parameters.

Figure 3-6. Return Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:2) Reserved

b(1:0) Input source

- 0h: External video port
- 1h: Test Pattern generator
- 2h: Splash screen
- 3h: Reserved

3.1.7 *Write External Video Source Format Select (07h)*

3.1.7.1 Write

This command specifies the active external video port and the source data type for the display module.

3.1.7.2 Write Parameters

Table 3-3 describes the command parameters.

Table 3-3. Write Parameters

CMD Parameter	Port	Bits/Pixel	Data Type	Bus Width	Clks/Pixel	Notes
00h	DSI	Variable	DSI	N/A	N/A	Data type is automatically detected; variable bus width configured using GPIO_01 and GPIO_02
40h	Parallel	16	RGB 565	16	1	Auto-select RGB CSC
41h	Parallel	18	RGB 666	18	1	Auto-select RGB CSC
42h	Parallel	24	RGB 888	8	3	Auto-select RGB CSC
43h	Parallel	24	RGB 888	24	1	Auto-select RGB CSC
50h	Parallel	18	YCbCr 666	18	1	Auto-select YCbCr CSC
51h	Parallel	24	YCbCr 888	24	1	Auto-select YCbCr CSC
60h	Parallel	16	YCbCr 4:2:2 88	8	2	Auto-select YCbCr CSC Auto-select 4:2:2 → 4:4:4
61h	Parallel	16	YCbCr 4:2:2 88	16	1	Auto-select YCbCr CSC Auto-select 4:2:2 → 4:4:4

This command is used in conjunction with the *Write Input Source Select* command. This command specifies which input port displays when the *Write Input Source Select* command selects external video port as the image source. The settings for this command are retained until changed using this command. These settings are automatically applied each time the external video port is selected.

When the external video port is selected as the input source, the software automatically selects and loads the proper CSC, based on the selected parameter of this command (appropriate matrix for RGB, selected matrix for YCbCr including offset).The appropriate data path is also automatically selected for 4:2:2 versus 4:4:4 processing.

The user should review the notes for the *Write Input Source Select* command to understand the concept of source-associated commands. This concept determines when source-associated commands are executed by the system. This command is a source-associated command.

3.1.8 Read External Video Source Format Select (08h)

3.1.8.1 Read

This command reads the state of the active external video port and the source data type for the display module.

3.1.8.2 Read Parameters

This command has no read parameters.

3.1.8.3 Return Parameters

Table 3-4 describes the return parameters.

Table 3-4. Return Parameters

Command Parameter	Port	Bits/Pixel	Data Type	Bus Width	Clks/Pixel	Notes
21h	DSI	16	RGB 565	N/A	N/A	Variable bus width configured using GPIO_01 and GPIO_02
22h	DSI	18	RGB 666	N/A	N/A	Variable bus width configured using GPIO_01 and GPIO_02
23h	DSI	18	RGB 666 (loosely packed)	N/A	N/A	Variable bus width configured using GPIO_01 and GPIO_02
24h	DSI	24	RGB 888	N/A	N/A	Variable bus width configured using GPIO_01 and GPIO_02
30h	DSI	8	YCbCr 4:2:2	N/A	N/A	Variable bus width configured using GPIO_01 and GPIO_02
40h	Parallel	16	RGB 565	16	1	Auto-select RGB CSC
41h	Parallel	18	RGB 666	18	1	Auto-select RGB CSC
42h	Parallel	24	RGB 888	8	3	Auto-select RGB CSC
43h	Parallel	24	RGB 888	24	1	Auto-select RGB CSC
50h	Parallel	18	YCbCr 666	18	1	Auto-select YCbCr CSC
51h	Parallel	24	YCbCr 888	24	1	Auto-select YCbCr CSC
60h	Parallel	16	YCbCr 4:2:2 88	8	2	Auto-select YCbCr CSC Auto-select 4:2:2 → 4:4:4
61h	Parallel	16	YCbCr 4:2:2 88	16	1	Auto-select YCbCr CSC Auto-select 4:2:2 → 4:4:4

3.1.9 Write Test Pattern Select (0Bh)

3.1.9.1 Write

This command specifies an internal test pattern for display on the display module.

3.1.9.2 Write Parameters

Table 3-5 describes the command parameters.

Table 3-5. Write Parameters

Parameter Bytes	Description
Byte 1	TPG pattern select
Byte 2	Foreground and background color (see Table 3-6)
Byte 3	Parameter 1 (see Table 3-7)
Byte 4	Parameter 2 (see Table 3-7)

Table 3-5. Write Parameters (continued)

Parameter Bytes	Description
Byte 5	Parameter 3 (see Table 3-7)
Byte 6	Parameter 4 (see Table 3-7)

Figure 3-7. Byte 1 Write Parameter

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7) Test pattern border:
- 00h: Disabled
- 01h: Enabled

b(6:4) Reserved

b(3:0) Left pattern select:
- 00h: Solid field
- 01h: Fixed step horizontal ramp
- 02h: Fixed step vertical ramp
- 03h: Horizontal lines
- 04h: Diagonal lines
- 05h: Vertical lines
- 06h: Horizontal and vertical grid
- 07h: Checkerboard
- 08h: Color bars
- 09h-0Fh: Reserved

Figure 3-8. Byte 2 Write Parameter

MSB	Byte 2						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7) Reserved

b(6:4) Foreground color:
- 0h: Black
- 1h: Red
- 2h: Green
- 3h: Blue
- 4h: Cyan
- 5h: Magenta
- 6h: Yellow
- 7h: White

b(3:0) Reserved

b(2:0) Background color:

- 0h: Black
- 1h: Red
- 2h: Green
- 3h: Blue
- 4h: Cyan
- 5h: Magenta
- 6h: Yellow
- 7h: White

Table 3-6. Foreground and Background Color Use

Pattern	Byte 2	
	Foreground Color	Background Color
Solid field	Yes	No
Fixed step horizontal ramp	Yes	No
Fixed step vertical ramp	Yes	No
Horizontal lines	Yes	Yes
Vertical lines	Yes	Yes
Diagonal lines	Yes	Yes
Grid lines	Yes	Yes
Checkerboard	Yes	Yes
Color bars	No	No

Table 3-7. Descriptions and Bit Assignments for Parameters 1-4

Pattern	Byte 6 (Parameter 4)		Byte 5 (Parameter 3)		Byte 4 (Parameter 2)		Byte 3 (Parameter 1)	
	Description	Bits	Description	Bits	Description	Bits	Description	Bits
Solid field	N/A		N/A		N/A		N/A	
Fixed step horizontal ramp	N/A		N/A		End value	8	Start value	8
Fixed step vertical ramp	N/A		N/A		End value	8	Start value	8
Horizontal lines	N/A		N/A		Background line width	8	Foreground line width	8
Vertical lines	N/A		N/A		Background line width	8	Foreground line width	8
Diagonal lines	N/A		N/A		Vertical spacing	8	Horizontal spacing	8
Grid lines	Vertical background line width	8	Vertical foreground line width	8	Horizontal background line width	8	Horizontal foreground line width	8
Checkerboard	Number of vertical checkers	3	Number of vertical checkers	8	Number of horizontal checkers	3	Number of horizontal checkers	8
Color bars	N/A		N/A		N/A		N/A	

This command is used in conjunction with the *Write Input Source Select* command. This command specifies which test pattern displays when the *Write Input Source Select* command selects test pattern generator as the image source. The settings for this command are retained until changed using this command. These settings automatically apply each time the test pattern generator is selected.

Batch files are created and stored in flash, and recall the settings for predefined test patterns.

Test patterns are created at the resolution of the display (DMD), are modified by the *Write Image Crop* command, and displayed at the resolution specified by the *Write Display Size* command.

Test patterns display at the default frame rate 60 Hz.

The *Test Pattern Border Selection* creates a white border, a single pixel wide and tall, around the specified test pattern.

The user must review the notes for the *Write Input Source Select* command to understand the concept of source-associated commands. This concept determines when source-associated commands are executed by the system. This command is a source-associated command.

When a foreground or background color is not used, the bit values are ignored (see Table 3-6). If both foreground and background color are not used, or when a parameter byte (bytes 3 thru 6) is not used, the byte should not be sent. Table 3-8 shows the number of bytes required, based on the specified pattern.

Table 3-8. Number of Bytes Required based on Pattern Selection

Specified Pattern	Number of Bytes Required
Solid field	2
Fixed step horizontal ramp	4
Fixed step vertical ramp	4
Horizontal lines	4
Vertical lines	4
Diagonal lines	4
Grid lines	6
Checkerboard	7
Color bars	1

As noted in Table 3-6, the color for the solid field pattern is specified using the foreground color. An example of a solid field pattern is shown in Figure 3-9.

Figure 3-9. Example of Solid Field Test Pattern (Red)

As noted in Table 3-6, the color for the fixed step horizontal ramp pattern is specified using the foreground color. As noted in Table 3-7, the user specifies the start value and the stop value for the ramp. For this pattern, the system automatically determines the step size based on the start and stop values and the size of the display (DMD). The minimum start value is 0, the maximum stop value is 255, and the start value must always be smaller than the stop value. For example, if the start value = 0, the stop value = 255, and the DMD resolution is 1280 wide, the step size would be 5 (1280 pixels / 256 values = 5). Thus every gray shade value from 0 to 255 would have a step size of 5 pixels (such that each step would have 5 columns of pixels with the same gray scale value). The gray scale value always increments by 1 for each step between the start and stop values. An example of a fixed step horizontal ramp pattern is shown in Figure 3-10.

Figure 3-10. Example of Fixed Step Horizontal Ramp Test Pattern

As noted in Table 3-6, the color for the fixed step vertical ramp pattern is specified using the foreground color. As noted in Table 3-7, the user specifies the start value and the stop value for the ramp. For this pattern, the system automatically determines the step size based on the start and stop values and the size of the display (DMD). The minimum start value = 0, the maximum stop value = 255, and the start value must always be smaller than the stop value. For example, if the start value = 0, the stop value = 255, and the DMD resolution is 768 tall, then the step size would be 3 (768 pixels / 256 values = 3). Thus every value from 0 to 255 would have a step size of 3 pixels (such that each step would have 3 rows of pixels with the same gray scale value). The gray scale value always increments by 1 for each step between the start and stop values. An example of a fixed step vertical ramp pattern is shown in Figure 3-11.

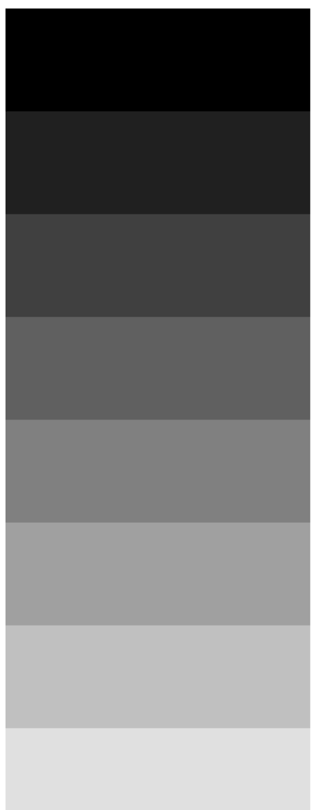


Figure 3-11. Example of Fixed Step Vertical Ramp Test Pattern

As noted in Table 3-6, the colors for the horizontal lines pattern are specified using both the foreground and background colors. The foreground color is used for the horizontal lines, and the background color is used for the space between the lines. As noted in Table 3-7, the user specifies the foreground line width, as well as the background line width. The user must determine the line spacing for each resolution display. For example, if the foreground line width = 1, and the background line width = 9, there would be a single pixel horizontal line on every tenth line. An example of a horizontal lines pattern is shown in Figure 3-12.

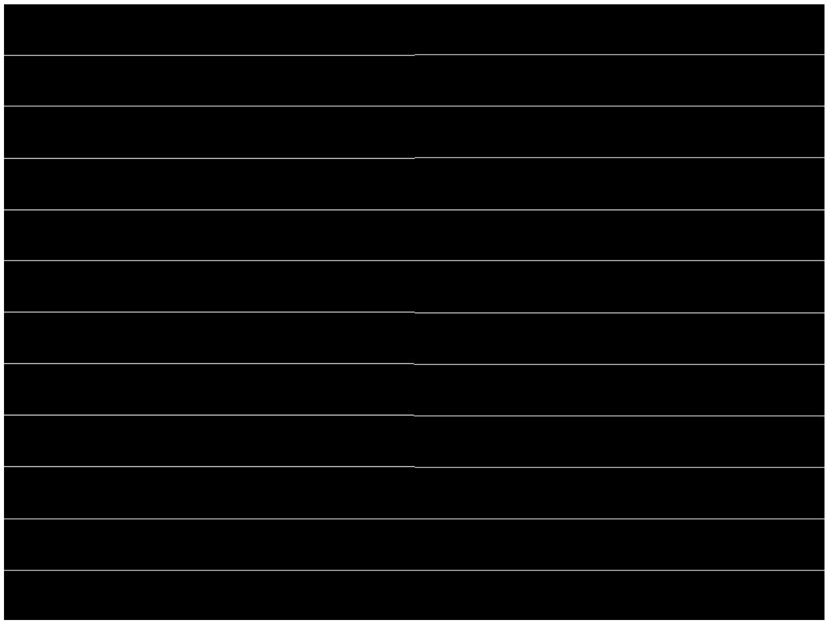


Figure 3-12. Example of Horizontal Lines Test Pattern

As noted in Table 3-6, the colors for the vertical lines pattern are specified using both the foreground and background colors. The foreground color is used for the vertical lines, and the background color is used for the space between the lines. As noted in Table 3-7, the user specifies the foreground line width, as well as the background line width. The user must determine the line spacing for each resolution display. For example, if the foreground line width = 1, and the background line width = 9, there would be a single pixel vertical line on every tenth line. An example of a vertical lines pattern is shown in Figure 3-13.

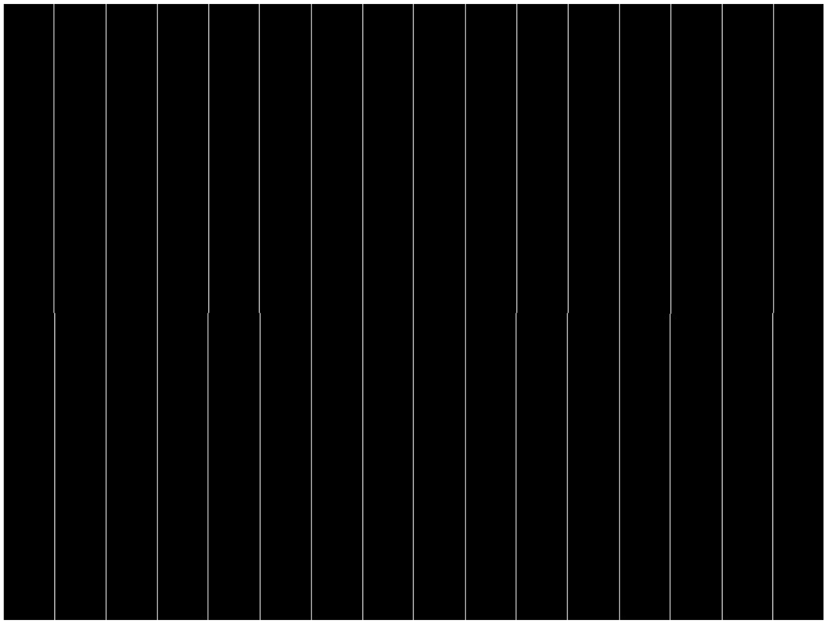


Figure 3-13. Example of Vertical Lines Test Pattern

As noted in Table 3-6, the colors for the diagonal lines pattern are specified using both the foreground and background colors. The foreground color is used for the diagonal lines, and the background color is used for the space between the lines. As noted in Table 3-7, the user specifies the horizontal and vertical line spacing. The line width is always one pixel. The user determines the line spacing for each resolution display. Both horizontal and vertical line spacing must use the same value, and are limited to values of 3, 7, 15, 31, 63, 127, and 255. Invalid values result in a communication error (invalid command parameter). An example of a diagonal lines pattern is shown in Figure 3-14.

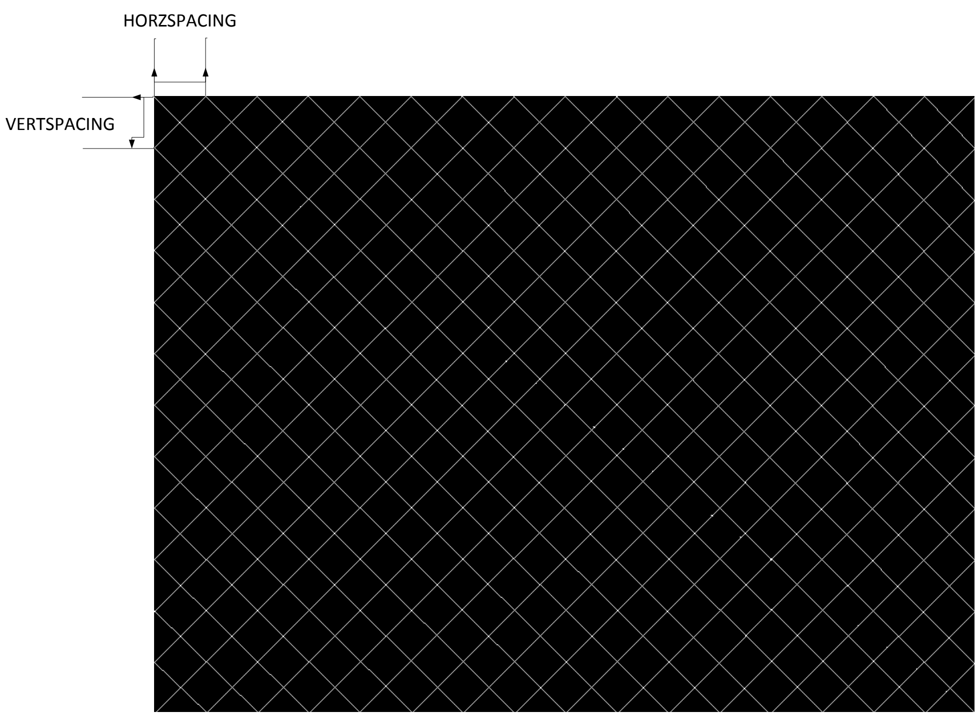


Figure 3-14. Example of Diagonal Lines Test Pattern

As noted in Table 3-6, the colors for the grid lines pattern are specified using both the foreground and background colors. The foreground color is used for the grid lines, and the background color is used for the space between the lines. As noted in Table 3-7, the user specifies the horizontal foreground and background line width, as well as the vertical foreground and background line width. The user determines the line spacing for each resolution display. For example, if the horizontal foreground line width = 1, and background line width = 9, there would be a single pixel horizontal line on every tenth line. If the vertical foreground line width = 1, and background line width = 9, there would be a single pixel vertical line on every tenth line. An example of a grid lines pattern is shown in Figure 3-15.

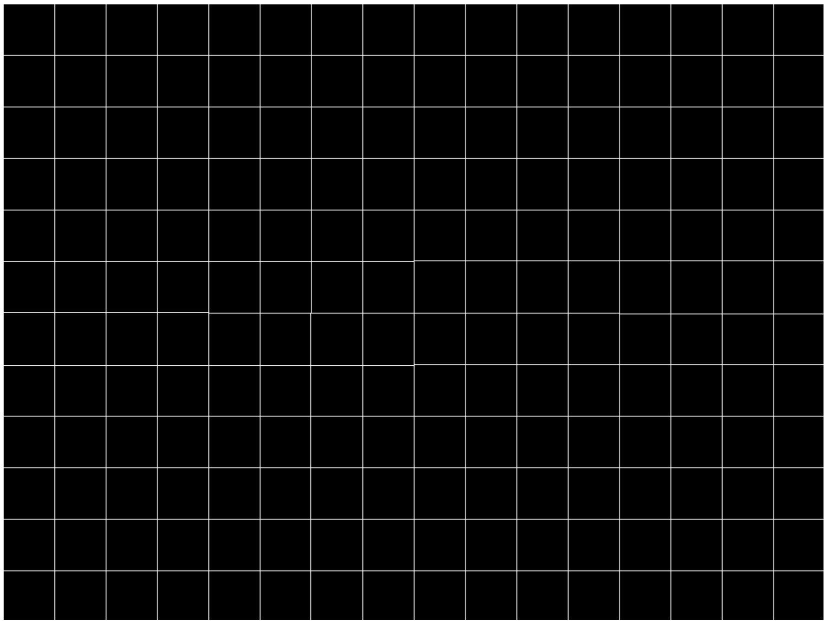


Figure 3-15. Example of Grid Lines Test Pattern

As noted in Table 3-6, the colors for the checkerboard pattern are specified using both the foreground and background colors. The foreground color is used for one of the checkers, and the background color is used for the alternating checker. As noted in Table 3-7, the user specifies the number of horizontal checkers and the number of vertical checkers. For this pattern, the system automatically determines the checker size in each direction based on the number of checkers and the size of the display (DMD). For example, if the number of horizontal checkers = 4, the number of vertical checkers = 4, and the DMD resolution is 1280x720, the size of the horizontal checkers is 320 pixels, and the size of the vertical checkers is 180 pixels (1280 pixels / 4 checkers = 320 pixels: 720 pixels / 4 checkers = 180 pixels). An example of a checkerboard pattern (16 checkers by 12 checkers) is shown in Figure 3-16.

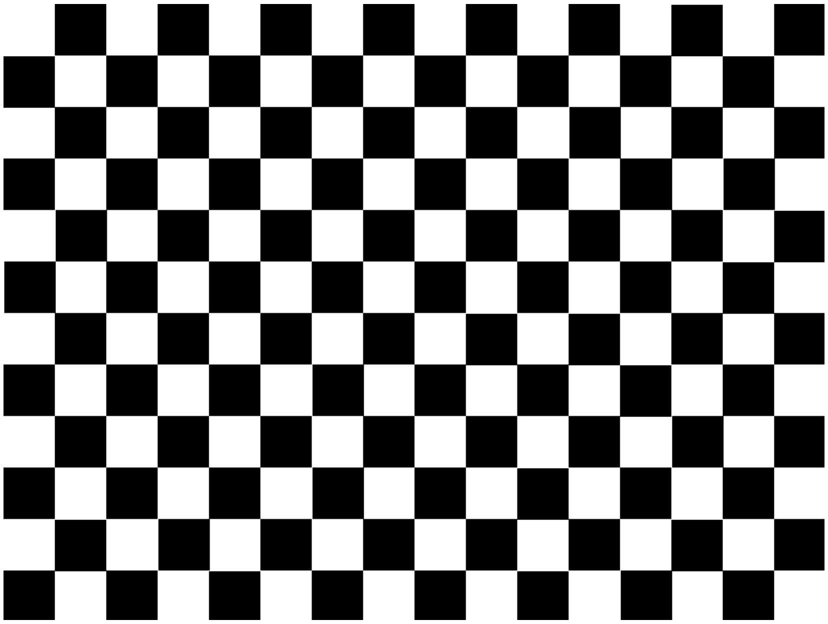


Figure 3-16. Example of Checkerboard Test Pattern

As noted in Table 3-6 and Table 3-7, there is no user programmability associated the color bars test pattern. This pattern is made up of eight vertical color bars: white, yellow, cyan, green, magenta, red, blue, and black. For this pattern, the system automatically determines the width for each color bar based on the size of the display (DMD). An example of the color bars pattern is shown in Figure 3-17.

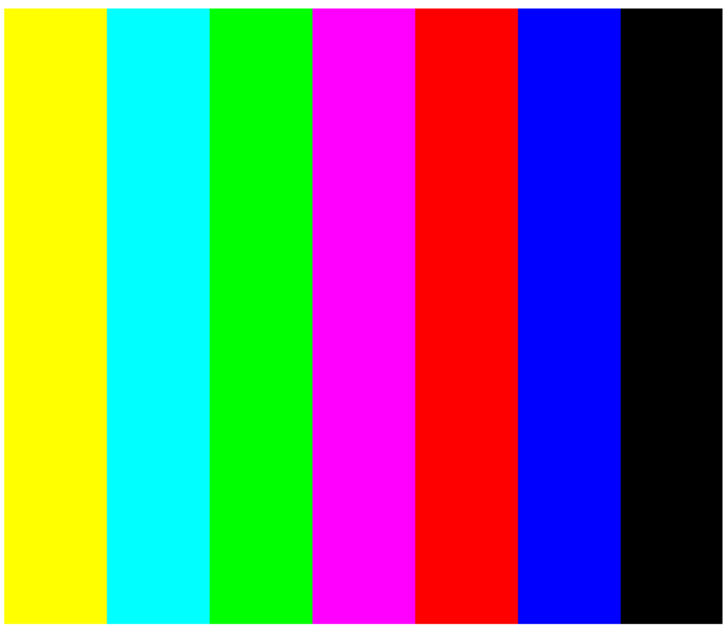


Figure 3-17. Example of Color Bars Test Pattern

3.1.10 Read Test Pattern Select (0Ch)

3.1.10.1 Read

This command reads the state of the test pattern select command for the display module.

3.1.10.2 Read Parameters

This command has no read parameters.

3.1.10.3 Return Parameters

Figure 3-18 describes the return parameters.

Figure 3-18. Return Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

Table 3-9. Parameter Bytes

Parameter Bytes	Description
Byte 1	TPG pattern select
Byte 2	Foreground and background color (see Table 3-6)
Byte 3	Parameter 1 (see Table 3-7)
Byte 4	Parameter 2 (see Table 3-7)
Byte 5	Parameter 3 (see Table 3-7)
Byte 6	Parameter 4 (see Table 3-7)

This command always returns six bytes, since the host does not know how many bytes are valid until the pattern is selected. All unnecessary bytes (see Table 3-8) are set to 0.

If a batch file is used to specify the parameters of the test pattern generator, those parameters are returned by this command.

3.1.11 Write Splash Screen Select (0Dh)

3.1.11.1 Write

This command selects a stored splash screen to be displayed on the display module.

3.1.11.2 Write Parameters

Table 3-10 describes the command parameters.

Table 3-10. Write Parameters

Parameter Bytes	Description
Byte 1	Splash screen reference number (integer)

This command is used in conjunction with the *Write Input Source Select* and the *Write Splash Screen Execute* commands, and specifies which splash screen is selected by the *Input Source Select* command. The settings for this command are retained until changed using this command.

The steps required to display a splash screen are:

1. Select the desired splash screen (using this command)
2. Change the input source to splash screen (using *Write Input Source Select*)
3. Start the splash screen retrieval process (using *Write Splash Screen Execute*).

The splash screen is read from flash and sent down the processing path of the controller once, to be stored in memory for display at the end of the processing path. As such, all image processing settings (such as image crop, image orientation, display size, splash screen select, splash screen as input source, and so forth) should be set by the user before executing the *Write Splash Screen Execute* command.

The user should review the notes for the *Write Input Source Select* command to understand the concept of source-associated commands. This concept determines when source-associated commands are executed by the system. This command is a source-associated command.

The availability of the splash screen is limited by the available space in flash memory. All splash screens must be landscape oriented.

The minimum splash image size allowed for flash storage is 427 x 240, with the maximum being the resolution of the utilized DMD. Typical splash image sizes for flash are 427 x 240 and 640 x 360. The full resolution size is typically used to support an optical test splash screen.

The user must specify how the splash image is displayed on the screen. Key commands for this are *Write Image Crop* and *Write Display Size*.

When this command is received while splash screen is the active source, other than storing the specified splash screen value, the only action taken by the controller software is to obtain the header information from the selected splash screen and store this in internal memory. When the *Write Splash Screen Execute* command is received, the controller software uses this stored information to set up the processing path prior to pulling the splash data from flash.

3.1.12 Read Splash Screen Select (0Eh)

3.1.12.1 Read

This command reads the state of the *Splash Screen Select* command of the display module.

3.1.12.2 Read Parameters

This command has no command parameters.

3.1.12.3 Return Parameters

Table 3-11 describes the return parameters.

Table 3-11. Return Parameters

Parameter Bytes	Description
Byte 1	Splash screen selected (integer)

3.1.13 Read Splash Screen Header (0Fh)

3.1.13.1 Read

This command reads the splash screen header information for the selected splash screen of the display module.

3.1.13.2 Read Parameters

The read parameter specifies the splash screen for which the header parameters are returned. If a splash screen value is provided for an unavailable splash screen, this is considered an error (invalid command parameter value – communication status).

Table 3-12. Read Parameters

Parameter Bytes	Description
Byte 1	Splash screen reference number (integer)

3.1.13.3 Return Parameters

Table 3-13 describes the return parameters.

Table 3-13. Return Parameters

Parameter Bytes	Description
Byte 1	Splash image width in pixels (LSByte)
Byte 2	Splash image width in pixels (MSByte)
Byte 3	Splash image height in pixels (LSByte)
Byte 4	Splash image height in pixels (MSByte)
Byte 5	Splash image size in bytes (LSByte)
Byte 6	Splash image size in bytes
Byte 7	Splash image size in bytes
Byte 8	Splash image size in bytes (MSByte)
Byte 9	Pixel format
Byte 10	Compression type
Byte 11	Color order
Byte 12	Chroma order
Byte 13	Byte order

Parameter definitions are referenced in Table 3-14.

Table 3-14. Splash Screen Header Definitions

Parameter	Values
Pixel format	0h = 24-bit RGB unpacked (not used) 1h = 24-bit RGB packed (not used) 2h = 16-bit RGB 5-6-5 3h = 16-bit YCbCr 4:2:2
Compression type	0h = Uncompressed 1h = RGB RLE compressed 2h = User-defined (not used) 3h = YUV RLE compressed
Color order	0h = 00RRGGBB 1h = 00GGRRBB
Chroma order	0h = Cr is first pixel 1h = Cb is first pixel
Byte order	0h = Little endian 1h = Big endian

3.1.14 Write Image Crop (10h)

3.1.14.1 Write

This command specifies which portion of the input image is captured and output from the cropping function of the display module.

3.1.14.2 Write Parameters

Table 3-15 describes the command parameters.

Table 3-15. Write Parameters

Parameter Bytes	Description	Details
Byte 1	Capture start pixel (LSByte)	Referenced to active data and 0-based (such that specifying the capture start pixel to be zero indicates the first active pixel of a line)
Byte 2	Capture start pixel (MSByte)	

Table 3-15. Write Parameters (continued)

Parameter Bytes	Description	Details
Byte 3	Capture start line (LSByte)	Referenced to active data and 0-based (such that specifying the capture start line to be zero indicates the first active line of a frame)
Byte 4	Capture start line (MSByte)	
Byte 5	Pixels per line (LSByte)	1-based (such that specifying a pixel per line value of 854 indicates 854 pixels to be captured)
Byte 6	Pixels per line (MSByte)	
Byte 7	Lines per frame (LSByte)	1-based (such that specifying a lines per frame value of 480 indicates 480 linesto be captured)
Byte 8	Lines per frame (MSByte)	

This command applies to all sources including test patterns, splash screens, and external sources. Making a change to the source or port does not impact the application of this command.

Cropping is done prior to the scaling function in the display module. As such, the size difference between the crop size and display size determines the amount of scaling needed in both dimensions. The scaling limits are listed in Table 3-16.

Table 3-16. Scaling Limits

Controller Configuration	Maximum Horizontal Interpolation Scale Factor	Maximum Horizontal Decimation Scale Factor	Maximum Vertical Interpolation Scale Factor	Maximum Vertical Decimation Scale Factor
Single controller (excluding interlaced NTSC/PAL)	3.0	3.0	3.0	3.0
Single controller (interlaced NTSC/PAL only)	3.0	3.0	3.0	6.0

The scaling limits noted in Table 3-16 may not be possible depending on other factors, such as keystone correction. In this case, the system does what is requested even if this results in a broken image. The user is responsible for providing the appropriate input settings to meet the display needs.

If a crop size parameter exceeds the size of the input image, the input image size minus the capture start pixel/line is be used (as shown in Figure 3-19). The crop size parameters returned by the read image crop command are always the values specified by the *Write Image Crop* command.

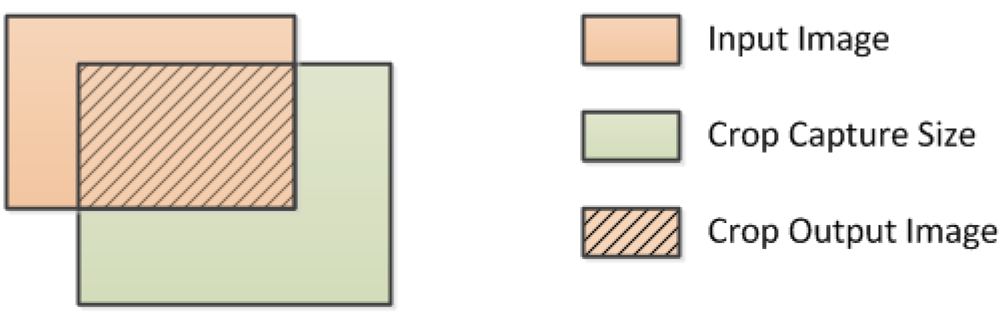


Figure 3-19. Cropping Rules when Crop Size exceeds Input Size

3.1.15 Read Image Crop (11h)

3.1.15.1 Read

This command reads the state of the image crop command for the display module.

3.1.15.2 Read Parameters

This command has no command parameters.

3.1.15.3 Return Parameters

Table 3-17 describes the return parameters.

Table 3-17. Return Parameters

Parameter Bytes	Description
Byte 1	Capture start pixel (LSByte)
Byte 2	Capture start pixel (MSByte)
Byte 3	Capture start line (LSByte)
Byte 4	Capture start line (MSByte)
Byte 5	Pixels per line (LSByte)
Byte 6	Pixels per line (MSByte)
Byte 7	Lines per frame (LSByte)
Byte 8	Lines per frame (MSByte)

All parameters for this command are referenced to active data, and are 1-based. (such that specifying the capture start pixel to be a value of one indicates the first active pixel of a line).

3.1.16 Write Display Size (12h)

3.1.16.1 Write

This command specifies the size of the active image to be displayed on the display module.

3.1.16.2 Write Parameters

Table 3-18 describes the command parameters.

Table 3-18. Write Parameters

Parameter Bytes	Description
Byte 1	Pixels per line (LSByte)
Byte 2	Pixels per line (MSByte)
Byte 3	Lines per frame (LSByte)
Byte 4	Lines per frame (MSByte)

This command specifies the size of the non-keystone corrected image to be output from the scaler function, which is the size of the active displayed image.

The parameter values are to be 1-based (such that a value of 1280 pixels displays 1280 pixels per line).

All sub-images (images smaller than the DMD display) are horizontally and vertically centered on the display (DMD).

If the display size exceeds the resolution of the DMD, this is considered an error (invalid command parameter value – communication status) and the command does not execute. The display size parameters are checked against the DMD resolution in both rotation image orientations (non-rotated and rotated), and if the DMD resolution is exceeded in both of these orientations, it is considered an error. The system does not check for proper image orientation setup.

For a DMD resolution of 854 × 480:

- Example 1: Display size parameter = 480 × 854 (not an error)
- Example 2: Display size parameter = 900 × 320 (error)
- Example 3: Display size parameter = 500 × 600 (error)

If the source, crop, and display parameter combinations exceed the capabilities of the scaler, the system implements the user request as best it can, and the displayed image may be broken. The user must provide updated parameters to fix the image.

3.1.17 *Read Display Size (13h)*

3.1.17.1 Read

This command reads the state of the display size command for the display module.

3.1.17.2 Read Parameters

This command has no read parameters.

3.1.17.3 Return Parameters

Table 3-19 describes the return parameters.

Table 3-19. Return Parameters

Parameter Bytes	Description
Byte 1	Pixels per line (LSByte)
Byte 2	Pixels per line (MSByte)
Byte 3	Lines per frame (LSByte)
Byte 4	Lines per frame (MSByte)

The parameter values are 1-based. (such that a value of 1280 pixels displays 1280 pixels per line).

3.1.18 *Write Display Image Orientation (14h)*

3.1.18.1 Write

This command specifies the image orientation of the displayed image for the display module.

3.1.18.2 Write Parameters

Figure 3-20 describes the command parameters.

Figure 3-20. Write Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:3) Reserved

b(2) Short axis image flip:
- 0: Image not flipped.
- 1: Image flipped.

b(1) Long axis image flip:
- 0: Image not flipped.
- 1: Image flipped.

b(0) Image rotation (for portrait source only):
- 0: No rotation
- 1: Minus 90° rotation

Figure 3-21 shows the result of non-rotation and rotation of a portrait source. If a portrait image is not rotated, it is centered and padded with black bars.

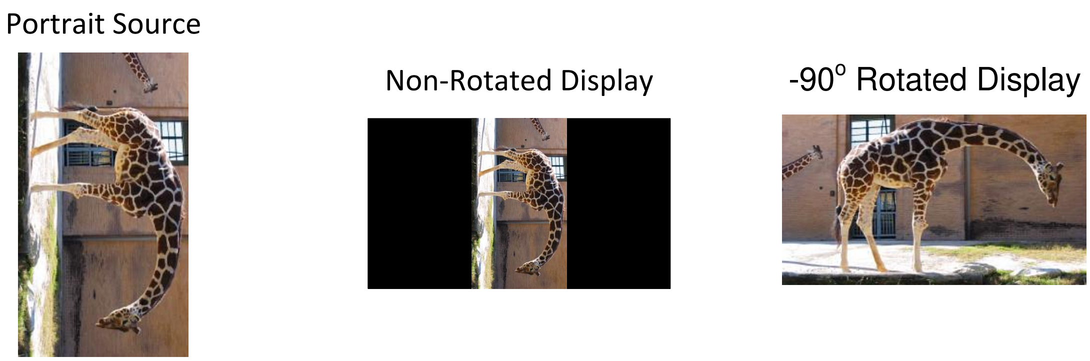


Figure 3-21. Rotation and Non-Rotation of Portrait Source

Landscape images typically should not be rotated, but the system allows this as it may be appropriate for some situations or configurations. The user is responsible for determining if the result is acceptable.

Image rotation is allowed while keystone correction is enabled, though it may not be appropriate for all situations or configurations. The user is responsible for determining if the result is acceptable.

Figure 3-22. Long-Axis Flip

Figure 3-23 shows the short-axis flip.

Figure 3-23. Short-Axis Flip

3.1.19 Read Display Image Orientation (15h)

3.1.19.1 Read

This command reads the state of the displayed image orientation function for the display module.

3.1.19.2 Read Parameters

This command has no read parameters.

3.1.19.3 Return Parameters

Figure 3-24 describes the return parameters.

Figure 3-24. Return Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:3) Reserved

b(2) Short-axis image flip:
- 0: Image not flipped.
- 1: Image flipped.

b(1) Long-axis image flip:
- 0: Image not flipped.
- 1: Image flipped.

b(0) Image rotation (for portrait source only):
- 0: No rotation
- 1: Minus 90° rotation

3.1.20 Write Display Image Curtain (16h)

3.1.20.1 Write

This command controls the display image curtain for the display module.

3.1.20.2 Write Parameters

Figure 3-25 shows the command parameters.

Figure 3-25. Write Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:4) Reserved

b(3:1) Select curtain color:
- 0h: Black
- 1h: Red
- 2h: Green
- 3h: Blue
- 4h: Cyan
- 5h: Magenta
- 6h: Yellow
- 7h: White

b(0) Curtain enable:
- 0: Curtain disabled
- 1: Curtain enabled

The image curtain fills the entire display with a user-specified color. The curtain color specified by this command is separate from the border color defined in the *Write Border Color* command, though both are displayed using the curtain capability.

3.1.21 Read Display Image Curtain (17h)

3.1.21.1 Read

This command reads the state of the image curtain control function for the display module.

3.1.21.2 Read Parameters

This command has no read parameters.

3.1.21.3 Return Parameters

Figure 3-26 describes the return parameters.

Figure 3-26. Return Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:4) Reserved

b(3:1) Select curtain color:

- 0h: Black
- 1h: Red
- 2h: Green
- 3h: Blue
- 4h: Cyan
- 5h: Magenta
- 6h: Yellow
- 7h: White

b(0) Curtain enable:

- 0: Curtain disabled
- 1: Curtain enabled

3.1.22 Write Image Freeze (1Ah)

3.1.22.1 Write

This command enables or disables the image freeze function for the display module.

3.1.22.2 Write Parameters

Figure 3-27 describes the command parameters.

Figure 3-27. Write Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:1) Reserved

b(0) Image freeze:
- 0: Image freeze disabled
- 1: Image freeze enabled

The image freeze capability has two main uses. The first use is to simply freeze the current image on the screen. The second use is to assist the user in reducing display artifacts during system changes. In this second case, the image is frozen, system changes are made, and the image is unfrozen when complete. In all cases, when the image is unfrozen, the display shows the most resent input image. Input data between the freeze point and the unfreeze point is lost.

The controller software does not freeze or unfreezes the image except when explicitly commanded by the *Write Image Freeze* command. This applies when software is making updates to the system on its own volition, and for any operation commanded via the I^2C interface.

The user must review the notes for the *Write Input Source Select* command to understand the concept of source-associated commands. This concept determines when source-associated commands are executed by the system.

If the OEM chooses not to make use of image freeze, they should change the source before changing the image parameters, to minimize transition artifacts.

3.1.22.3 Use of Image Freeze to Reduce On-Screen Artifacts

Commands that take a long time to process, require a lot a data to be loaded from flash, or change the frame timing of the system may create on-screen artifacts. The *Write Image Freeze* command can try and minimize, if not eliminate, these artifacts. The process is:

1. Send a *Write Image Freeze* command to enable freeze.
2. Send commands with the potential to create image artifacts.
3. Send a *Write Image Freeze* command to disable freeze.

Because commands to the controller process serially, no special timing or delay is required between these commands. The number of commands placed between the freeze and unfreeze should be small, as it is not desirable for the image to be frozen for a long period of time. A list of commands that may produce image artifacts is listed in Table 3-20. This is not an all-inclusive list, and the user is responsible for determining the correct use of the image freeze command.

Table 3-20. Partial List of Commands that May Benefit from the Use of Image Freeze

Command	Command OpCode	Notes
Write Input Source Select	05h	
Write External Video Source Format Select	07h	If changed while this source is the active source
Write Test Pattern Select	0Bh	If changed while this source is the active source
Write Splash Screen Select	0Dh	If changed while this source is the active source
Write Look Select	22h	

Table 3-21 and Table 3-22 show a few examples of how to use the image freeze command.

Table 3-21. Splash Screen Example Using Image Freeze

Command	Notes
Write Display Image Curtain = enable	May want to apply curtain if already displaying an unwanted image (such as a broken source)
Write Image Freeze = freeze	

Table 3-21. Splash Screen Example Using Image Freeze (continued)

Command	Notes
Write Image Crop, Write Display Size, Write Display Image Orientation	Potential data processing commands that may be required for proper display of splash image. These must be set prior to write splash screen execute command to affect the splash screen image.
Write Splash Screen Select Write Input Source Select = splash	These must be set prior to write splash screen execute
Write Splash Screen Execute	Retrieves the desired splash screen image for display
Write Image Freeze = unfreeze	

The new splash image displays when the *Write Splash Screen Execute* command executes, regardless of the state of the *Write Image Freeze* command (due to the one time nature of the splash image). *Write Image Freeze* = unfreeze must still be executed.

Table 3-22. Test Pattern Generator Example Using Image Freeze

Command	Notes
Write Image Freeze = freeze	
Write Image Crop, Write Display Size, Write Display Image Orientation, Write Test Pattern Select	Potential data processing commands that may be required for proper display of test pattern image. These should be set before the Write Input Source Select command.
Write Input Source Select = test pattern generator	
Write Image Freeze = unfreeze	

3.1.23 Read Image Freeze (1Bh)

3.1.23.1 Read

This command reads the state of the image freeze function for the display module.

3.1.23.2 Read Parameters

This command has no read parameters.

3.1.23.3 Return Parameters

Figure 3-28 describes the return parameters.

Figure 3-28. Return Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:1)	Reserved
b(0)	Image freeze: • 0: Image freeze disabled • 1: Image freeze enabled

3.1.24 Write Look Select (22h)

3.1.24.1 Write

This command specifies the Look for the image on the display module.

3.1.24.2 Write Parameters

Figure 3-29 describes the command parameters.

Figure 3-29. Write Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:0) Look number

In this product, a Look typically specifies a target white point. The number of Looks available may be limited by the available space in flash memory.

This command allows the host to select a Look (target white point) from a number of Looks stored in flash.

Looks are specified in this byte by an enumerated value (such as 0,1,2,3). There must always be at least one Look, with an enumerated value of 0.

There are two other items that the host should specify when determining a white point. These are:

- A desired degamma curve, achieved by selecting the appropriate degamma/CMT, which has the desired degamma curve and correct bit weights for the sequence selected.
- The desired color points, achieved by selecting the appropriate CCA parameters using the CCA select command.

3.1.25 Read Look Select (23h)

3.1.25.1 Read

This command reads the state of the Look select command for the display module.

3.1.25.2 Read Parameters

This command has no read parameters.

3.1.25.3 Return Parameters

Table 3-23 describes the return parameters.

Table 3-23. Return Parameters

Parameter Bytes	Description
Byte 1	See Figure 3-30
Byte 2	Current red duty cycle (LSByte)
Byte 3	Current red duty cycle (MSByte)
Byte 4	Current green duty cycle (LSByte)
Byte 5	Current green duty cycle (MSByte)
Byte 6	Current blue duty cycle (LSByte)
Byte 7	Current blue duty cycle (MSByte)
Byte 8	Current sequence frame rate (LSByte)
Byte 9	Current sequence frame rate
Byte 10	Current sequence frame rate
Byte 11	Current sequence frame rate (MSByte)

Figure 3-30. Byte 1 Return Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:0) Look number

Figure 3-31. Byte 2 Return Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:0) Sequence number

Looks are specified by an enumerated value (such as 0, 1, 2, 3).

The current sequence frame rate is returned as a count specified in units of 66.67 ns (based on the internal 15-MHz clock used to time between input frame syncs), and is valid regardless of whether the controller software made the sequence or duty cycle selection. The frame rate is specified in this way to enable fast and simple compares to the frame count by the software.

3.1.26 Read Sequence Header Attributes (26h)

3.1.26.1 Read

This command reads sequence header information for the active sequence of the display module.

3.1.26.2 Read Parameters

This command has no read parameters.

3.1.26.3 Return Parameters

Table 3-24 describes the return parameters.

Table 3-24. Return Parameters

Parameter Bytes	Description
Byte 1	Red duty cycle (LSByte) (Look structure)
Byte 2	Red duty cycle (MSByte) (Look structure)
Byte 3	Green duty cycle (LSByte)(Look structure)
Byte 4	Green duty cycle (MSByte) (Look structure)
Byte 5	Blue duty cycle (LSByte) (Look structure)
Byte 6	Blue duty cycle (MSByte) (Look structure)
Byte 7	Maximum frame count (LSByte) (Look structure)
Byte 8	Maximum frame count (Look structure)
Byte 9	Maximum frame count (Look structure)
Byte 10	Maximum frame count (MSByte) (Look structure)
Byte 11	Minimum frame count (LSByte) (Look structure)
Byte 12	Minimum frame count (Look structure)
Byte 13	Minimum frame count (Look structure)
Byte 14	Minimum frame count (MSByte) (Look structure)
Byte 15	Max number of sequence vectors (Look structure)
Byte 16	Red duty cycle (LSByte) (Sequence structure)
Byte 17	Red duty cycle (MSByte) (Sequence structure)
Byte 18	Green duty cycle (LSByte) (Sequence structure)
Byte 19	Green duty cycle (MSByte) (Sequence structure)
Byte 20	Blue duty cycle (LSByte) (Sequence structure)
Byte 21	Blue duty cycle (MSByte) (Sequence structure)

Table 3-24. Return Parameters (continued)

Parameter Bytes	Description
Byte 22	Maximum frame count (LSByte) (Sequence structure)
Byte 23	Maximum frame count (Sequence structure)
Byte 24	Maximum frame count (Sequence structure)
Byte 25	Maximum frame count (MSByte) (Sequence structure)
Byte 26	Minimum frame count (LSByte) (Sequence structure)
Byte 27	Minimum frame count (Sequence structure)
Byte 28	Minimum frame count (Sequence structure)
Byte 29	Minimum frame count (MSByte) (Sequence structure)
Byte 30	Max number of sequence vectors (Sequence structure)

The sequence header data is stored in two separate flash data structures (the Look structure and the sequence structure), and the values from each should match.

The bit weight and bit order for the duty cycle data is shown in Figure 3-32.

Figure 3-32. Bit Weight and Bit Order for Duty Cycle Data

MSB	Byte 2						LSB	*MSB*	Byte 1						LSB
b15 2^7	b14 2^6	b13 2^5	b12 2^4	b11 2^3	b10 2^2	b9 2^1	b8 2^0	b7 2^{-1}	b6 2^{-2}	b5 2^{-3}	b4 2^{-4}	b3 2^{-5}	b2 2^{-6}	b1 2^{-7}	b0 2^{-8}

The duty cycle data is specified as each colors percent of the frame time. The sum of the three duty cycles must add up to 100. An example possibility is, R = 30.5 = 1E80h, G = 50 = 3200h, and B = 19.5 = 1380h)

The sequence maximum and minimum frame counts are specified in units of 66.67 ns (based on the internal 15 MHz clock used to time between input frame syncs). These are specified in this way to enable fast and simple compares to the frame count by the software.

The maximum number of sequence vectors byte is defined in Figure 3-33.

Figure 3-33. Maximum Number of Sequence Vectors

MSB	Byte 15 and 30						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:4)	Reserved
b(3:0)	Maximum number of sequence vectors

3.1.27 Write CCA Select (29h)

3.1.27.1 Write

This command specifies which set of CCA (color coordinate adjustment) parameters to use.

3.1.27.2 Write Parameters

Figure 3-34 describes the command parameters.

Figure 3-34. Write Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:0)	CCA Parameter Set Index

One or more CCA parameter sets may be included in a firmware image. Each CCA parameter set is intended to specify a target color point in the system. This command enables selection from multiple CCA sets by an enumerated value (i.e 0, 1, 2, etc.) A CCA parameter set must exist for it to be selected. This command may be used in conjunction with *Write Color Coordinate Adjustment Control*.

3.1.28 Read CCA Select (2Ah)

3.1.28.1 Read

This command reads the state of the CCA Parameter Set Index.

3.1.28.2 Read Parameters

This command has no command parameters.

3.1.28.3 Return Parameters

Figure 3-35 describes the return parameters.

Figure 3-35. Return Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:0) CCA Parameter Set Index

3.1.29 Read DMD Sequencer Sync Mode (2Ch)

3.1.29.1 Read

This command reads the state of the DMD sequencer sync mode function of the display module.

3.1.29.2 Read Parameters

This command has no read parameters.

3.1.29.3 Return Parameters

Figure 3-36 describes the return parameters.

Figure 3-36. Return Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:2) Reserved

b(1) System auto-sync setting:
- 0h: Lock to external VSYNC (auto-sync)
- 1h: Lock to internal VSYNC (auto-sync)

b(0) DMD sequencer sync mode:
- 0h: Auto-sync
- 1h: Force lock to internal VSYNC

The DMD sequencer sync mode response indicates the setting specified by the *Write DMD Mode* command.

System auto-sync setting response is only valid when the DMD sequencer sync mode is set to auto-sync (otherwise set to 0). The lock to the external VSYNC (auto-sync) option indicates that the system is using the externally provided VSYNC to drive the display module. The lock to the internal VSYNC option indicates that the system is using the internal VSYNC generator to drive the display module.

3.1.30 Write Execute Flash Batch File (2Dh)

3.1.30.1 Write

This command executes a flash batch file for the display module.

3.1.30.2 Write Parameters

Table 3-25 describes the command parameters.

Table 3-25. Write Parameters

Parameter Bytes	Description
Byte 1	Batch file number

This command executes a batch file stored in the flash of the display module. Any system write command that can be sent by itself can be grouped together with other system commands or command parameters into a flash batch file, with the exception of all read commands.

The flash batch file numbers specified in this byte are enumerated values (such as 0,1,2,3). Flash batch file 0 is a special auto-init batch file that runs automatically by the DLPC343x software immediately after system initialization is complete. The flash batch file 0 is typically not called using the *Write Execute Batch File* command (although the system allows it). This special flash batch file specifies the source to be used (such as splash screen or data port) once the system initializes.

Embedding flash batch file calls within a flash batch file is not allowed (for example, calling another batch file from within a batch file is not allowed). To execute two batch files back to back, use back to back execute batch file commands.

The system allows adding an execution delay between commands within a flash batch file. This is done using the *Write Flash Batch File Delay* command.

The order of command execution for commands within a flash batch file is the same as if the commands are received over the I^2C port.

3.1.31 Write External Input Image Size (2Eh)

3.1.31.1 Write

This command specifies the active data size of the external input image to the display module.

3.1.31.2 Write Parameters

Table 3-26 describes the command parameters.

Table 3-26. Write Parameters

Parameter Bytes	Description
Byte 1	Pixels per line (LSByte)
Byte 2	Pixels per line (MSByte)
Byte 3	Lines per frame (LSByte)
Byte 4	Lines per frame (MSByte)

This command is used in conjunction with the *Write Input Source Select* command. This command specifies the active data size of the input image to the system for all external video interfaces, when the *Write Input Source Select* command selects external video port as the image source. The settings for this command are retained until changed using this command. These settings automatically apply each time the external video port is selected.

The parameter values are 1-based (for example, a value of 1280 pixels specifies 1280 pixels per line).

The user should review the notes for the *Write Input Source Select* command to understand the concept of source-associated commands. This concept determines when source-associated commands are executed by the system. This command is a source-associated command.

The maximum and minimum input values are shown in Table 3-27. Values outside of these ranges are flagged as an error (invalid command parameter), and the command does not execute.

Table 3-27. Input Source Limits for Active Data

Parameter	Minimum Value	Maximum Value
Input source active pixels per line	320	1280
Input source active lines per frame	200	800

3.1.32 Read External Input Image Size (2Fh)

3.1.32.1 Read

This command reads the specified data size of the external input image to the display module.

3.1.32.2 Read Parameters

This command has no read parameters.

3.1.32.3 Return Parameters

Table 3-28 describes the return parameters.

Table 3-28. Return Parameters

Parameter Bytes	Description
Byte 1	Pixels per line (LSByte)
Byte 2	Pixels per line (MSByte)
Byte 3	Lines per frame (LSByte)
Byte 4	Lines per frame (MSByte)

The parameter values are 1-based (for example, a value of 1280 pixels specifies 1280 pixels per line).

This command returns the value specified by the *Write External Input Image Size* command.

3.1.33 Write Splash Screen Execute (35h)

3.1.33.1 Write

This command starts the process of retrieving a splash screen from flash for display on the display module.

3.1.33.2 Write Parameters

This command has no write parameters.

This command is used in conjunction with the *Write Input Source Select* and the *Write Splash Screen Select* commands to start the process of retrieving a splash screen from flash for display.

The splash screen is read from flash and sent down the processing path of the controller once, to be stored in memory for display at the end of the processing path. All image processing settings (such as image crop, image orientation, display size, splash screen select, and splash screen as input source) should be set by the user before executing this command. Any data path processing changed after the splash screen has been executed requires this command to be re-executed before the result is seen on the display. Thus, the splash screen retrieval process repeats each time this command is received. See the *Write Image Freeze* command for more information on hiding on-screen artifacts when selecting and retrieving a splash image.

The user should review the notes for the *Write Input Source Select* command to understand the concept of source-associated commands. This concept determines when source-associated commands are executed by the system. This command is a source-associated command; however, this command has no maintained state or history and has no settings to be stored or reused by the system.

When this command is processed, the system automatically sets up the system color processing based on the splash header information, prior to sending the splash image down the data path.

3.1.34 Write Parallel Data Mask Control (37h)

3.1.34.1 Write

This command controls the masking function for the external parallel port interface of the DLPC343x controller.

3.1.34.2 Write Parameters

Figure 3-37 describes the command parameters.

Figure 3-37. Write Parameters

MSB	**Byte 1**						**LSB**
b7	b6	b5	b4	b3	b2	b1	b0

b(7:2) Reserved

b(1) Polarity Select for Data Mask Control:
- 0: Unmasked = 0, Masked = 1
- 1: Unmasked = 1, Masked = 0

b(0) Data Mask Enable:
- 0: Mask Disable
- 1: Mask Enable

When the parallel data mask is enabled, the DLPC343x input PDM_CVS_TE pin functions as a data mask control for the video data on the parallel port interface. Therefore, when this functionality is enabled and the mask control is active, input image frames will be ignored and the source image will not be propagated to the display. During image frames that are masked, the last unmasked image frame received will continue to be displayed. The mask control signal (PDM_CVS_TE) should only be updated during vertical blanking.

The Polarity Select specifies the active state for the mask control signal. The polarity should only be updated when the mask function is disabled (via this command).

3.1.35 Read Parallel Data Mask Control (38h)

3.1.35.1 Read

This command reads the state of the parallel data mask control.

3.1.35.2 Read Parameters

This command has no command parameters.

3.1.35.3 Return Parameters

Figure 3-38 describes the return parameters.

Figure 3-38. Return Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:2)	Reserved
b(1)	Polarity Select for Data Mask Control: • 0: Unmasked = 0, Masked = 1 • 1: Unmasked = 1, Masked = 0
b(0)	Data Mask Enable: • 0: Mask Disabled • 1: Mask Enabled

3.2 Illumination Control

3.2.1 Write LED Output Control Method (50h)

3.2.1.1 Write

This command specifies the method for controlling the LED outputs for the display module.

3.2.1.2 Write Parameters

Figure 3-39 describes the command parameters.

Figure 3-39. Write Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:2)	Reserved
b(1:0)	LED control method: • 00: Manual RGB LED currents (disables CAIC algorithm) • 01: CAIC (automatic) RGB LED power (enables CAIC algorithm) • 10: Reserved • 11: Reserved

This command selects the method used to control the output of the red, green, and blue LEDs. Based on the method chosen, a specific set of commands are available for controlling the LED outputs. These are shown in Table 3-29.

The manual RGB LED currents method provides for manual control of the LED currents, and disables the CAIC algorithm. The CAIC (automatic) RGB LED current control method provides automatic control of the LED currents using the CAIC algorithm.

Table 3-29. Available Commands Based on LED Control Method

LED Control Method	Available Commands
Manual RGB LED current control	Write RGB LED Enable Read RGB LED Enable Write Manual RGB LED Current Read Manual RGB LED Current Write Manual RGB LED Max Current (see *Section 3.2.8*) Read Manual RGB LED Max Current (see *Section 3.2.9*)
CAIC (automatic) RGB LED current control	Write RGB LED Enable Read RGB LED Enable Read CAIC LED Max Available Power Read CAIC LED RGB Current

3.2.2 Read LED Output Control Method (51h)

3.2.2.1 Read

This command reads the state of the LED output control method for the display module.

3.2.2.2 Read Parameters

This command has no read parameters.

3.2.2.3 Return Parameters

Figure 3-40 describes the return parameters.

Figure 3-40. Return Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:2) Reserved

b(1:0) LED control method:
- 00: Manual RGB LED currents (CAIC algorithm disabled)
- 01: CAIC (automatic) RGB LED current control (CAIC algorithm enabled)
- 10: Reserved
- 11: Reserved

3.2.3 Write RGB LED Enable (52h)

3.2.3.1 Write

This command enables the LEDs for the display module.

3.2.3.2 Write Parameters

Figure 3-41 describes the command parameters.

Figure 3-41. Write Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:3) Reserved

b(2) Blue LED enable:
- 0: Blue LED disabled
- 1: Blue LED enabled

b(1) Green LED enable:
- 0: Green LED disabled
- 1: Green LED enabled

b(0) Red LED enable:
- 0: Red LED disabled
- 1: Red LED enabled

3.2.4 Read RGB LED Enable (53h)

3.2.4.1 Read

This command reads the state of the LED enables for the display module.

3.2.4.2 Read Parameters

This command has no read parameters.

3.2.4.3 Return Parameters

Figure 3-42 describes the return parameters.

Figure 3-42. Return Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:3) Reserved

b(2) Blue LED enable:
- 0: Blue LED disabled
- 1: Blue LED enabled

b(1) Green LED enable:
- 0: Green LED disabled
- 1: Green LED enabled

b(0) Red LED enable:
- 0: Red LED disabled
- 1: Red LED enabled

3.2.5 Write RGB LED Current (54h)

3.2.5.1 Write

This command sets the current for the red, green, and blue LEDs of the display module.

3.2.5.2 Write Parameters

Table 3-30 describes the command parameters.

Table 3-30. Write Parameters

Parameter Bytes	Description
Byte 1	Red LED current parameter (LSByte)
Byte 2	Red LED current parameter (MSByte)
Byte 3	Green LED current parameter (LSByte)
Byte 4	Green LED current parameter (MSByte)
Byte 5	Blue LED current parameter (LSByte)
Byte 6	Blue LED current parameter (MSByte)

When an all-white image is displayed, this command allows the system white point to be adjusted while establishing the total LED power. This is true whether the CAIC algorithm is enabled or disabled.

The parameters specified by this command have a resolution of 10 bits, and are defined by the appropriate DLPA200x specification.

When the CAIC algorithm is disabled, this command directly sets the LED currents (the R, G, and B values provided are sent directly to the DLPA200x device) regardless of the image being displayed.

When the CAIC algorithm is enabled:

- This command directly sets the LED currents when an all-white image is displayed. If the image is changed from an all-white image, depending on the image the CAIC algorithm may alter one or more of the LED currents from those specified by this command, and the total LED power may drop. The *Read CAIC RGB LED Current* command can read the actual LED currents for the image currently displayed.
- In the case of an all-white image, the values read by the *Read CAIC RGB LED Current* command closely match, but may not exactly match, those requested using the *Write RGB LED Current* command. For an all-white image, the *Read CAIC RGB LED Current* command gives currents within +/-4 DLPA200x device current steps for each LED color relative to those requested by the *Write RGB LED Current* command.

- When the *Write RGB LED Current* command changes the LED currents, the LED current for any color should not be changed by more than +/-25% from the nominal current used for that color when the CAIC LUTs were created. No LED current should be set to a current value beyond the maximum value supported in the CAIC intensity-to-current LUT for the corresponding color.
- The maximum total LED power for any displayed image occurs for an all-white image, since the CAIC algorithm requests the CAIC LED maximum available power. The maximum available LED power for the CAIC is controlled by the *Write RGB LED Current* command, as this command controls currents for an all-white image. After the currents are adjusted, the *Read CAIC LED Max Available Power* command is used to see the maximum power in watts derived from the CAIC.

3.2.6 Read RGB LED Current (55h)

3.2.6.1 Read

This command reads the state of the current for the red, green, and blue LEDs of the display module.

3.2.6.2 Read Parameters

This command has no read parameters.

3.2.6.3 Return Parameters

Table 3-31 describes the return parameters.

Table 3-31. Return Parameters

Parameter Bytes	Description
Byte 1	Red LED current parameter (LSByte)
Byte 2	Red LED current parameter (MSByte)
Byte 3	Green LED current parameter (LSByte)
Byte 4	Green LED current parameter (MSByte)
Byte 5	Blue LED current parameter (LSByte)
Byte 6	Blue LED current parameter (MSByte)

See *Section 3.2.5* for a detailed description of the return parameters.

Unused most significant bits are set to 0.

3.2.7 Read CAIC LED Max Available Power (57h)

3.2.7.1 Read

This command reads the specified maximum LED power allowed for the display module.

3.2.7.2 Read Parameters

This command has no read parameters.

3.2.7.3 Return Parameters

Table 3-32 describes the return parameters.

Table 3-32. Return Parameters

Parameter Bytes	Description
Byte 1	Maximum LED power (LSByte)
Byte 2	Maximum LED power (MSByte)

The value is specified in watts × 100 (for example: 25.75 W = A0Fh). This command is only applicable when CAIC is enabled.

The CAIC maximum available LED power pertains if an all-white image is displayed where LED currents are set by the *Write RGB LED Current* command. The calculation is:

```
R duty cycle × R LED current × R LED voltage +
             G duty cycle × G LED current × G LED voltage +
             B duty cycle × B LED current × B LED voltage.
```

For example: (.30 × .49 A × 2.0 V) + (.50 × .39 A × 3.1 V) + (.20 × .39 A × 3.1 V) = (.30 × .980 W) + (.50 × 1.209 W) + (.20 × 1.209 W) = 1.140 W

3.2.8 Write RGB LED Max Current (5Ch)

3.2.8.1 Write

This command specifies the maximum LED current allowed for each LED in the display module.

3.2.8.2 Write Parameters

Table 3-33 describes the command parameters.

Table 3-33. Write Parameters

Parameter Bytes	Description
Byte 1	Maximum red LED current (LSByte)
Byte 2	Maximum red LED current (MSByte)
Byte 3	Maximum green LED current (LSByte)
Byte 4	Maximum green LED current (MSByte)
Byte 5	Maximum blue LED current (LSByte)
Byte 6	Maximum blue LED current (MSByte)

This command sets the maximum LED currents that can be used when CAIC is enabled or disabled. When CAIC is enabled, the maximum LED currents may be further limited by the CAIC LUTs stored in the flash.

3.2.9 Read RGB LED Max Current (5Dh)

3.2.9.1 Read

This command reads the specified maximum LED current allowed for each LED in the display module.

3.2.9.2 Read Parameters

This command has no read parameters.

3.2.9.3 Return Parameters

Table 3-34 describes the return parameters.

Table 3-34. Return Parameters

Parameter Bytes	Description
Byte 1	Maximum red LED current (LSByte)
Byte 2	Maximum red LED current (MSByte)
Byte 3	Maximum green LED current (LSByte)
Byte 4	Maximum green LED current (MSByte)
Byte 5	Maximum blue LED current (LSByte)

Table 3-34. Return Parameters (continued)

Parameter Bytes	Description
Byte 6	Maximum blue LED current (MSByte)

See the *Section 3.2.5* for a detailed description of the return parameters.

Unused most significant bits are set to 0.

3.2.10 Read CAIC RGB LED Current (5Fh)

3.2.10.1 Read

This command reads the state of the current for the red, green, and blue LEDs of the display module.

3.2.10.2 Read Parameters

This command has no read parameters.

3.2.10.3 Return Parameters

Table 3-35 describes the return parameters.

Table 3-35. Return Parameters

Parameter Bytes	Description
Byte 1	Red LED current parameter (LSByte)
Byte 2	Red LED current parameter (MSByte)
Byte 3	Green LED current parameter (LSByte)
Byte 4	Green LED current parameter (MSByte)
Byte 5	Blue LED current parameter (LSByte)
Byte 6	Blue LED current parameter (MSByte)

The parameters returned by this command have a resolution of 10 bits, and are defined by the appropriate DLPA200x specification.

When the CAIC algorithm is enabled using the *LED Output Control Method* command:

- The *Write RGB LED Current* command directly sets the LED currents when an all-white image is displayed. If the image changes from an all-white image, depending on the image, the CAIC algorithm may alter one or more of the LED currents from those specified by the *Write RGB LED Current* command, and the total LED power may drop. The actual LED currents for the image currently displayed are read using the *Read CAIC RGB LED Current* command.
- In the case of an all-white image, the values returned by this command closely match, but may not exactly match, those specified using the *Write RGB LED Current* command. For an all-white image, this command provides values within +/- 4 DLPA200x device current steps for each LED color relative to those specified with the *Write RGB LED Current* command.

Use of this command is only appropriate when the LED output control method is set to CAIC (automatic) RGB LED current control.

Unused most significant bits are set to 0.

3.3 Image Processing Control

3.3.1 Write Local Area Brightness Boost Control (80h)

3.3.1.1 Write

This command controls the local area brightness boost image processing functionality for the display module.

3.3.1.2 Write Parameters

Table 3-36 describes the command parameters.

Table 3-36. Write Parameters

Parameter Bytes	Description
Byte 1	See Figure 3-43
Byte 2	LABB strength setting

Figure 3-43. Byte 1 Return Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:4)	Sharpness strength
b(3:2)	Reserved
b(1:0)	LABB control: • 0h: Disabled • 1h: Enabled: Manual strength control • 2h: Reserved • 3h: Reserved

The key function of the LABB is to adaptively gain up darker parts of the image to achieve an overall brighter image.

For LABB strength, 0 indicates no boost applied and 255 indicates the maximum boost viable in a product. The strength is not a direct indication of the gain, since the gain varies depending on the image content.

Sharpness strength ranges from 0 to 15, with 0 indicating sharpness disabled and 15 indicating the maximum sharpness. The LABB function must be enabled to make use of sharpness.

LABB is supported in TPG, splash, and external input mode, but auto-disabled in curtain mode.

3.3.2 Read Local Area Brightness Boost Control (81h)

3.3.2.1 Read

This command reads the state of the local area brightness boost image processing functionality for the display module.

3.3.2.2 Read Parameters

This command has no read parameters.

3.3.2.3 Return Parameters

Figure 3-44 describes the return parameters.

Figure 3-44. Return Parameters

Parameter Bytes	Description
Byte 1	See Figure 3-45
Byte 2	LABB strength setting
Byte 3	LABB gain value

Figure 3-45. Byte 1 Return Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:4) Sharpness strength

b(3:2) Reserved

b(1:0) LABB control:

- 0h: Disabled
- 1h: Enabled: Manual strength control
- 2h: Reserved
- 3h: Reserved

Figure 3-46 shows the bit order and weighting for the LABB gain value, which ranges from 1 to 8 (the controller software should limit the lower value to 1).

Figure 3-46. Bit Weight Definition for LABB Gain Value

b7	b6	b5	b4	b3	b2	b1	b0
2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}

The software equation to calculate LABB Gain as a fixed point value is shown below:

```
LABB_gain = add_8lsb(APL) / pre_LABB_APL      (//add 8 LSBs (u8.0 / u8.0 = u8.8 / u8.0 = u8.8)
```

3.3.3 Write CAIC Image Processing Control (84h)

3.3.3.1 Write

This command controls the CAIC functionality for the display module.

3.3.3.2 Write Parameters

Table 3-37 describes the command parameters.

Table 3-37. Write Parameters

Parameter Bytes	Description
Byte 1	See Figure 3-47
Byte 2	CAIC maximum lumens gain
Byte 3	CAIC clipping threshold

Figure 3-47. Byte 1 Write Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7) CAIC gain display enable:

- 0h: Disabled
- 1h: Enabled

b(6) CAIC gain display scale:

- 0h: 100% = 1024 pixels
- 1h: 100% = 512 pixels

b(5:0) Reserved

The CAIC algorithm (Content Adaptive Illumination Control) provides adaptive control of the LED currents and the digital gain applied to the image.

The CAIC algorithm is enabled or disabled based on the method of LED current control selected by the OEM using the *Write LED Output Control Method* command. When enabled, the CAIC algorithm provides automatic control of the LED currents as specified by this command and the *Write LED Output Control Method* command.

The CAIC gain display provides a visual presentation of the instantaneous gain provided by the CAIC algorithm. This is typically used as a debug tool and to show the performance of the algorithm, and should never be used for normal operation. The display is composed of five bars, with the bottom three bars (green, red, and blue) showing the respective CAIC gain for each color. The top two bars are for TI debug use only. For the software, the CAIC gain display enable is controlled by CAIC_DEBUG_MODE (2:0), where disabled = 0h, and enabled = 3h. The display scale is set using CAIC_DEBUG_MODE(3).

Figure 3-48 shows the bit order and weighting for the CAIC maximum lumens gain value, which has a valid range from 1.0 to 4.0. Values outside of this range are considered an error (invalid command parameter value – communication status) and the command does not execute.

Figure 3-48. Bit Weight Definition for the CAIC Maximum Gain Value

b7	b6	b5	b4	b3	b2	b1	b0
2^{2}	2^{1}	2^{0}	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}

The CAIC maximum lumens gain parameter sets the maximum lumens gain for a pixel as a result of both digital gain and increasing LED currents. The CAIC maximum lumens gain parameter also serves to bias the CAIC algorithm towards either constant power (variable brightness) or constant lumens (variable power). Some examples are listed below:

- Maximum gain value = 1.0: This biases performance to constant lumens. In this case, LED power is reduced for those images where this is possible, but lumens do not increase or decrease.
- Maximum lumens gain value = 4.0: This biases performance to constant power. In this case, power is held constant for most images, while the lumens are gained up. For the small percent of images where the gain exceeds 4.0, lumens stop increasing and the power is reduced.

Figure 3-49 shows the bit order and weighting for the CAIC clipping threshold value, which has a valid range from 0.0% to 4.0%. Values outside of this range are considered an error (invalid command parameter value – communication status) and the command does not execute.

Figure 3-49. Bit Weight Definition for the CAIC Clipping Threshold Value

b7	b6	b5	b4	b3	b2	b1	b0
2^{1}	2^{0}	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}

The CAIC clipping threshold parameter sets the percentage of pixels clipped by the CAIC algorithm over the full frame of active data, due to the digital gain applied by the CAIC algorithm.

Figure 3-50 shows the bit order and weighting for the CAIC RGB intensity gain values, which have a valid range from 0.0 to almost 1.0. Values outside of this range are considered an error (invalid command parameter value – communication status) and the command does not execute.

Figure 3-50. Bit Weight Definition for the CAIC RGB Intensity Gain Values

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Res	Res	Res	Res	Res	Res	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}	2^{-8}	2^{-9}	2^{-10}

CAIC can be enabled in TPG and external input mode, but auto-disabled in splash and curtain mode.

Table 3-38. LABB and CAIC Modes

Feature	TPG	Splash	Curtain	External Input
LABB	Supported	Supported	Auto-disabled	Supported
CAIC	Supported	Auto-disabled	Auto-disabled	Supported
Manual idle mode	Supported	Supported	Auto-disabled	Supported

3.3.4 Read CAIC Image Processing Control (85h)

3.3.4.1 Read

This command reads the state of the CAIC functionality within the display module.

3.3.4.2 Read Parameters

This command has no read parameters.

3.3.4.3 Return Parameters

Table 3-39 describes the return parameters.

Table 3-39. Return Parameters

Parameter Bytes	Description
Byte 1	See Figure 3-51
Byte 2	CAIC maximum lumens gain
Byte 3	CAIC clipping threshold

Figure 3-51. Byte 1 Return Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7)	CAIC gain display enable: • 0h: Disabled • 1h: Enabled
b(6)	CAIC gain display scale: • 0h: 100% = 1024 pixels • 1h: 100% = 512 pixels
b(5:0)	Reserved

Information on these parameters can be found in *Section 3.3.3*.

3.3.5 Write Color Coordinate Adjustment Control (86h)

3.3.5.1 Write

This command controls the CCA image processing functionality for the display module.

3.3.5.2 Write Parameters

Figure 3-52 describes the command parameters.

Figure 3-52. Write Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:1)	Reserved
b(0)	CCA enable: • 0: Disabled • 1: Enabled

This command is for TI debug purposes only. This function should remain enabled during normal operation.

When CCA is disabled, an identity matrix is used.

3.3.6 Read Color Coordinate Adjustment Control (87h)

3.3.6.1 Read

This command reads the state of the CCA image processing within the display module.

3.3.6.2 Read Parameters

This command has no read parameters.

3.3.6.3 Return Parameters

Figure 3-53 describes the return parameters.

Figure 3-53. Return Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:1) Reserved

b(0) CCA enable:
- 0: Disabled
- 1: Enabled

3.3.7 Write Keystone Correction Control (88h)

3.3.7.1 Write

This command controls the keystone correction image processing functionality for the display module.

3.3.7.2 Write Parameters

Table 3-40 describes the command parameters.

Table 3-40. Write Parameters

Parameter Bytes	Description
Byte 1	See Figure 3-54
Byte 2	Optical throw ratio (LSByte)
Byte 3	Optical throw ratio (MSByte)
Byte 4	Optical DMD offset (LSByte)
Byte 5	Optical DMD offset (MSByte)

Figure 3-54. Byte 1 Write Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:1) Reserved

b(0) Keystone correction enable:
- 0: Disabled
- 1: Enabled

Keystone correction digitally compensates for distorted images when the projector is tilted up or down. Keystone correction is specified by the pitch angle (described in the *Write Keystone Projection Pitch Angle* command) and based on the throw ratio, vertical offset, and projector orientation. Each parameter is provided by this command. With this information, keystone correction corrects for both overall and local area aspect ratio distortion. For both full screen images and sub-images, the full active area of the DMD is keystone-corrected.

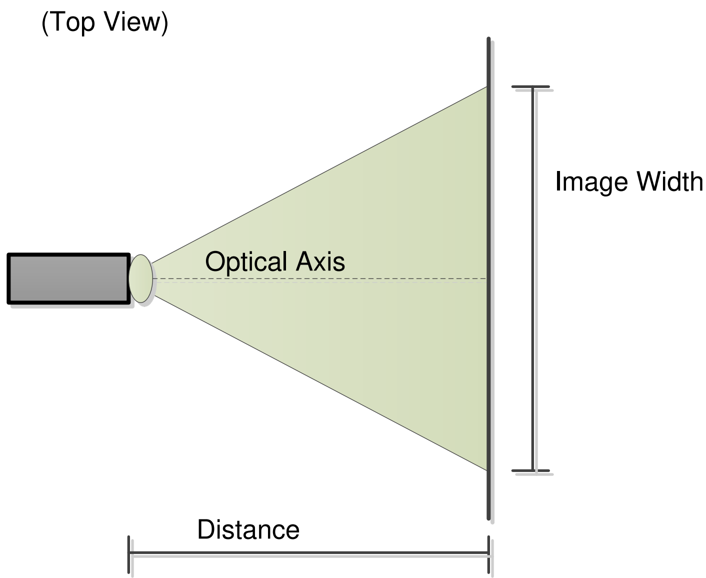
When keystone correction is enabled, the *Write Border Color* command sets the border color to black. Setting this parameter to any other color produces undesirable results.

Image rotation is allowed while keystone correction is enabled, but it may not be appropriate for all situations or configurations. The user is responsible for determining if the result is acceptable.

Figure 3-55 shows the bit order and weighting for the optical throw ratio data. Figure 3-56 defines how this data is determined.

Figure 3-55. Bit Weight Definition for the Optical Throw Ratio Data

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}	2^{-8}



Throw Ratio = Distance / Image Width
Throw Ratio Register Value = 256 × Throw Ratio

Figure 3-56. Visual Definition and Calculation for Optical Throw Ratio Data

Figure 3-57 shows the bit order and weighting for the 2's complement optical DMD offset data. Figure 3-58 shows how this data is calculated, while Figure 3-59 shows how the sign of the offset data is determined. The user must insure that both the value and the sign of the offset data are correctly determined.

Figure 3-57. Bit Weight Definition for the Optical DMD Offset Data

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}	2^{-8}

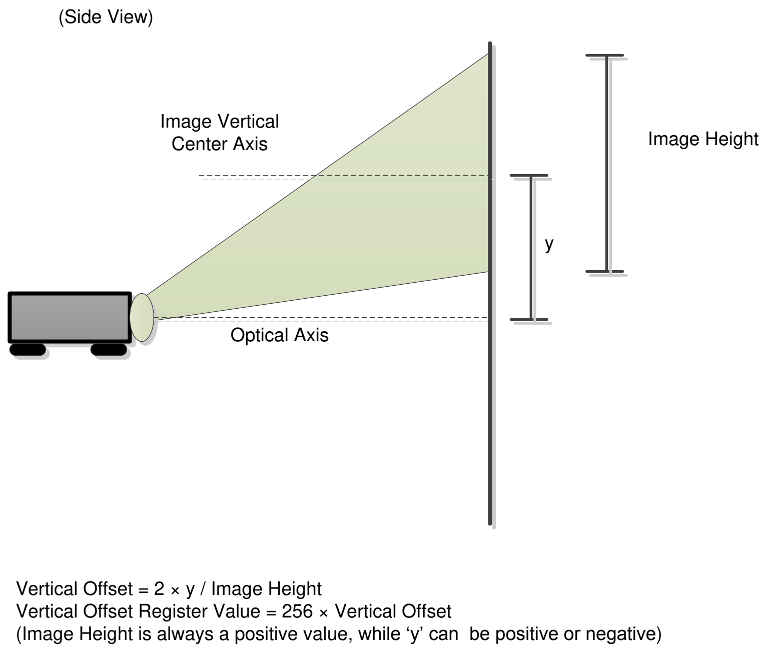


Figure 3-58. Method for Calculation for Optical DMD Offset Data

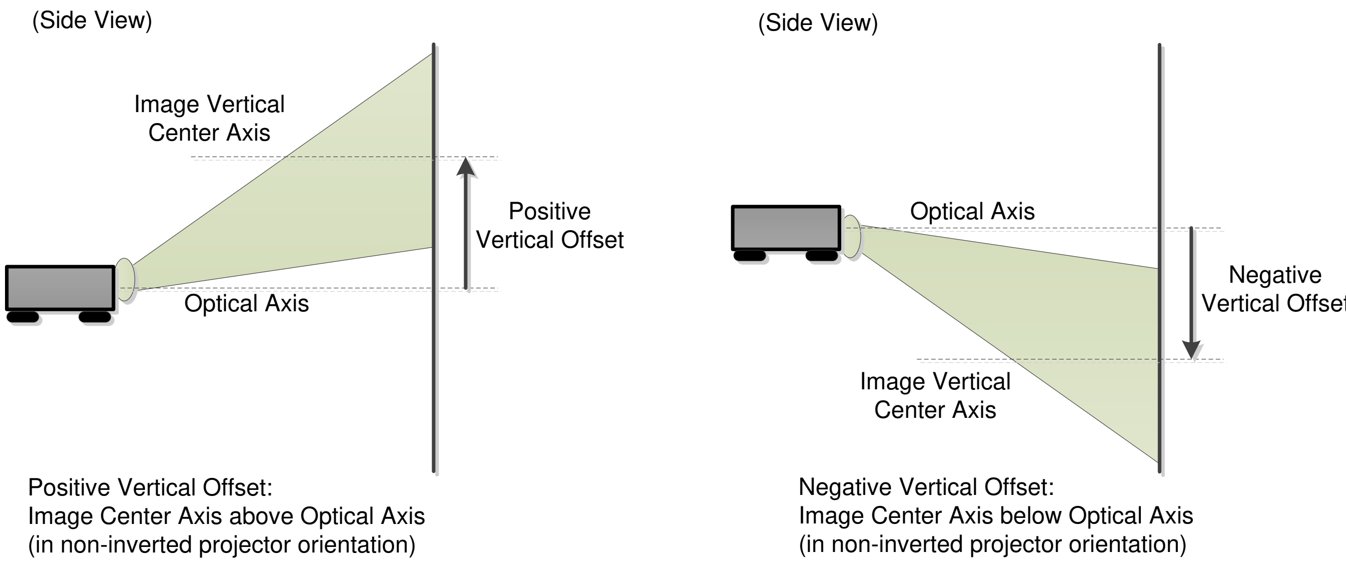


Figure 3-59. Sign Determination for Optical DMD Offset Data

Figure 3-60 shows examples of non-inverted and inverted projector orientation. This information is required for byte 1 of this command.

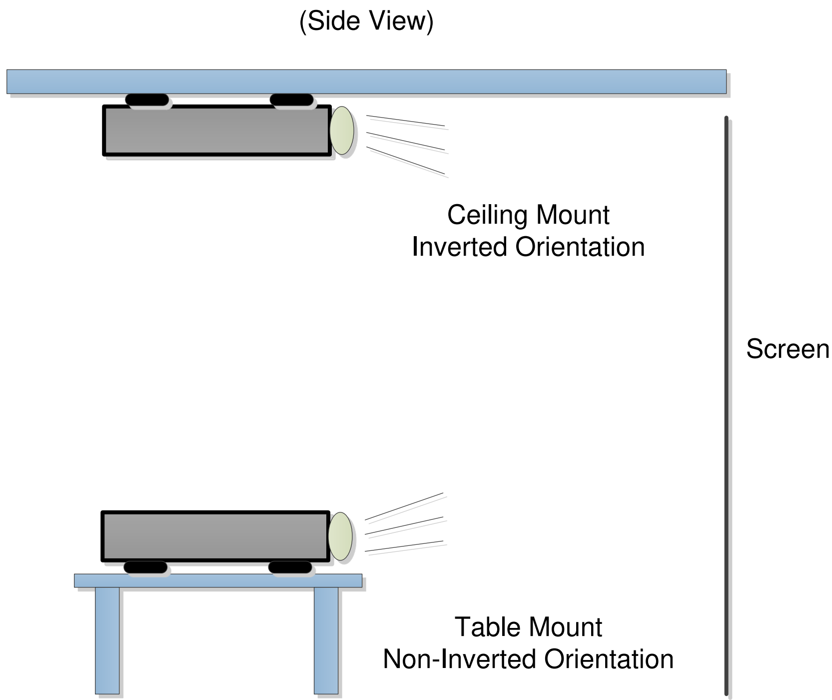


Figure 3-60. Examples of Non-Inverted and Inverted Projector Orientations

3.3.8 Read Keystone Correction Control (89h)

3.3.8.1 Read

This command reads the state of the keystone correction image processing within the display module.

3.3.8.2 Read Parameters

This command has no read parameters.

3.3.8.3 Return Parameters

Table 3-41 describes the return parameters.

Table 3-41. Return Parameters

Parameter Bytes	Description
Byte 1	See Figure 3-61
Byte 2	Optical throw ratio (LSByte)
Byte 3	Optical throw ratio (MSByte)
Byte 4	Optical DMD offset (LSByte)
Byte 5	Optical DMD offset (MSByte)

Figure 3-61. Byte 1 Return Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:1) Reserved

b(0) Keystone correction enable:
- 0: Disabled
- 1: Enabled

3.4 General Setup

3.4.1 Write Border Color (B2h)

3.4.1.1 Write

This command specifies the onscreen border color for the display module.

3.4.1.2 Write Parameters

Figure 3-62 describes the command parameters.

Figure 3-62. Write Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:3) Reserved

b(2:0) Display border color:

- 0h: Black
- 1h: Red
- 2h: Green
- 3h: Blue
- 4h: Cyan
- 5h: Magenta
- 6h: Yellow
- 7h: White

Whenever the display image size is smaller than the active area of the DMD, the border color is used for all non-image pixels. Some examples using a border include a window box, pillar box, or letterbox image.

To display a pillar box image (see Figure 3-63), the OEM can use the border color defined by this command.

The border color specified by this command is separate from the curtain color defined in the *Display Image Curtain* command, though both display using the curtain capability.

Whenever the keystone capability is used, the OEM should set the border color to black. Setting this parameter to any other color when keystone is enabled produces undesirable results.

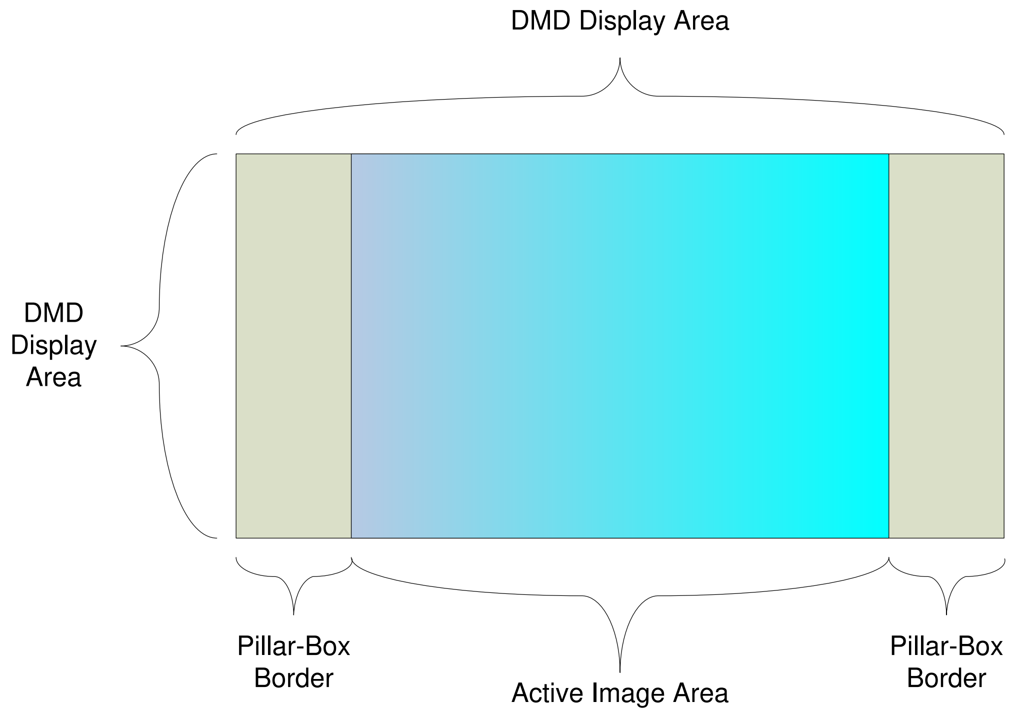


Figure 3-63. Pillar-Box Border Example

3.4.2 Read Border Color (B3h)

3.4.2.1 Read

This command reads the state of the onscreen border color for the display module.

3.4.2.2 Read Parameters

This command has no read parameters.

3.4.2.3 Return Parameters

Figure 3-64 describes the return parameters.

Figure 3-64. Return Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7) Pillar-box border color source:
- 0h: Defined by this command
- 1h: Flash defined 24-bit color

b(6:3) Reserved

b(2:0) Display border color:

- 0h: Black
- 1h: Red
- 2h: Green
- 3h: Blue
- 4h: Cyan
- 5h: Magenta
- 6h: Yellow
- 7h: White

For the special case of a pillar box image (see Figure 3-63), the OEM can use the border color defined by the *Write Border Color* command.

3.4.3 Write Parallel Interface Sync Polarity (B6h)

3.4.3.1 Write

This command specifies the VSYNC and HSYNC polarity for the DLPC343x parallel interface.

3.4.3.2 Write Parameters

Figure 3-65 describes the command parameters.

Figure 3-65. Write Parameters

MSB	Byte 1						**LSB**
b7	b6	b5	b4	b3	b2	b1	b0

b(7:3) Reserved

b(2) Manual Mode – Parallel Interface HSYNC Polarity:

- 0: Falling Edge Active (Negative Pulse)
- 1: Rising Edge Active (Positive Pulse)

b(1) Manual Mode – Parallel Interface VSYNC Polarity:

- 0: Falling Edge Active (Negative Pulse)
- 1: Rising Edge Active (Positive Pulse)

b(0) Parallel Interface Sync Polarity Mode:

- 0: Automatic Mode
- 1: Manual Mode

This command may be needed when using the parallel interface port. This command is not applicable for BT656 sources. In Automatic mode, the system can typically determine the appropriate polarity of the syncs. In Manual mode, the user is allowed to specify these polarities should the need arise.

3.4.4 Read Parallel Interface Sync Polarity (B7h)

3.4.4.1 Read

This command reads the state of the VSYNC and HSYNC polarity of the DLPC343x parallel interface.

3.4.4.2 Read Parameters

This command has no command parameters.

3.4.4.3 Return Parameters

Figure 3-66 describes the return parameters.

Figure 3-66. Return Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:2) Reserved

b(1) Manual Mode – Parallel Interface HSYNC Polarity:
- 0: Falling Edge Active (Negative Pulse)
- 1: Rising Edge Active (Positive Pulse)

b(0) Manual Mode – Parallel Interface VSYNC Polarity:
- 0: Falling Edge Active (Negative Pulse)
- 1: Rising Edge Active (Positive Pulse)

3.4.5 Write Keystone Projection Pitch Angle (BBh)

3.4.5.1 Write

This command specifies the projection pitch angle for the display module.

3.4.5.2 Write Parameters

Table 3-42 describes the command parameters.

Table 3-42. Write Parameters

Parameter Bytes	Description
Byte 1	Projection pitch angle (LSByte)
Byte 2	Projection pitch angle (MSByte)

Default: 0000h

Figure 3-67 shows the bit order and weighting for the 2's complement projection pitch angle data.

Figure 3-67. Bit Weight Definition for the Projection Pitch Angle Data

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}	2^{-8}

This command is used in conjunction with the *Write Keystone Correction Control* command.

The projection pitch angle is limited to the range of –40 to 40 degrees. Figure 3-68 shows examples of the projection pitch angle.

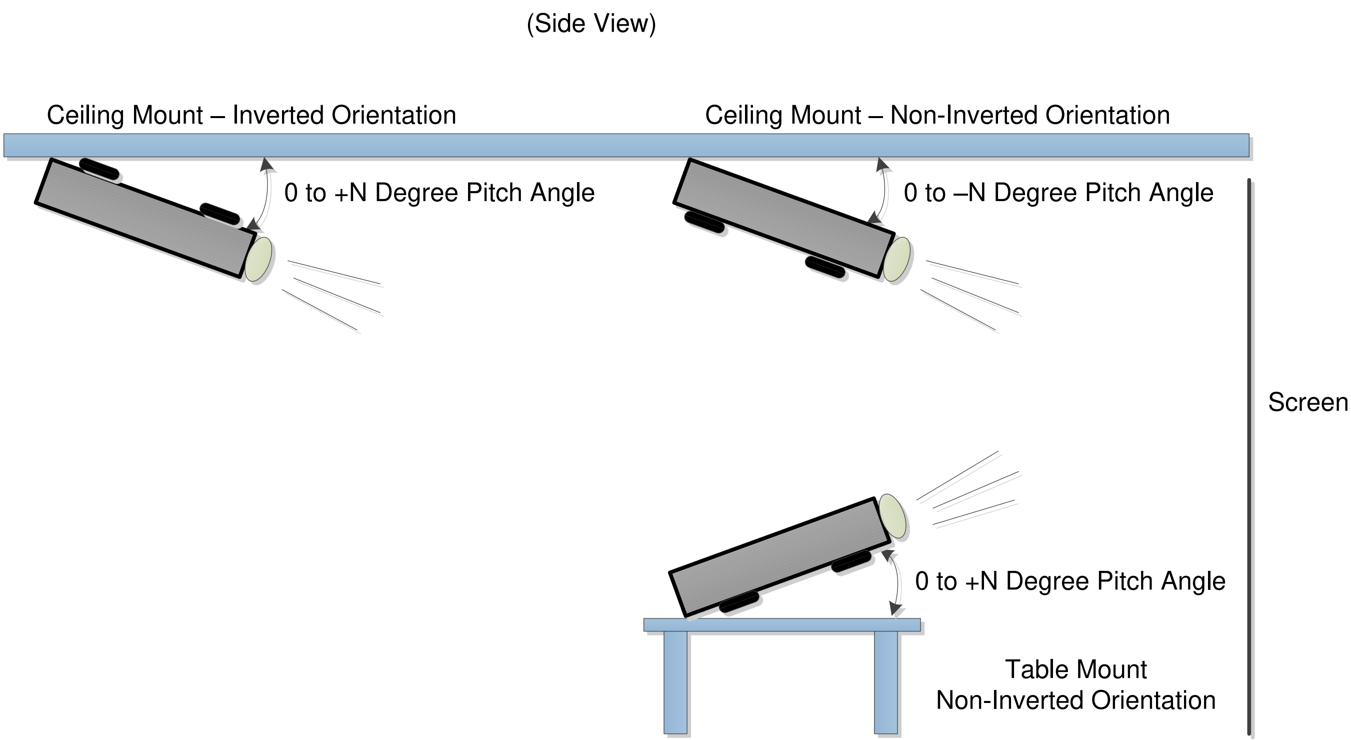


Figure 3-68. Examples of Projection Pitch Angle

3.4.6 Read Keystone Projection Pitch Angle (BCh)

3.4.6.1 Read

This command reads the specified projection pitch angle for the display module.

3.4.6.2 Read Parameters

This command has no read parameters.

3.4.6.3 Return Parameters

Table 3-43 describes the return parameters.

Table 3-43. Return Parameters

Parameter Bytes	Description
Byte 1	Projection pitch angle (LSByte)
Byte 2	Projection pitch angle (MSByte)

3.4.7 Write DSI HS Clock (BDh)

3.4.7.1 Write

This command specifies the DSI (display serial interface) host's HS (High Speed) clock rate for the DLPC343x controller.

3.4.7.2 Write Parameters

Table 3-44 describes the command parameters.

Table 3-44. Return Parameters

Parameter Bytes	Description
Byte 1	DSI HS clock rate (LSB)
Byte 2	DSI HS clock rate (MSB)

Before using DSI for video, the correct HS clock must first be set in the DLPC343x controller. The DLPC343x clock rate should be set to match the clock rate of the DSI host device. If the HS clock rate is changed during normal operation (i.e. on-the-fly), each time the clock rate is to be changed, the DSI HS transactions must first stop before sending this command.

The DSI HS clock rate is to be provided in units of MHz. For example, 376 MHz would result in 0178h and would be sent as *BD 78 01*. All values should be rounded up to the next megahertz value (e.g 75.1 MHz would be rounded to 76 MHz).

3.4.8 Read DSI HS Clock (BEh)

3.4.8.1 Read

This command reads the state of the DSI HS (High Speed) clock rate set in the DLPC343x controller.

3.4.8.2 Read Parameters

This command has no command parameters.

3.4.8.3 Return Parameters

describes the return parameters.

Table 3-45. Return Parameters

Parameter Bytes	Description
Byte 1	DSI HS clock rate (LSB)
Byte 2	DSI HS clock rate (MSB)

The DSI HS clock rate is returned in units of MHz. For example, 376 MHz would result in 0178h.

3.4.9 Write DSI Port Enable (D7h)

3.4.9.1 Write

This command is used to enable or disable the DSI port for the display module.

3.4.9.2 Write Parameters

Figure 3-69 describes the command parameters.

Figure 3-69. Write Parameters

MSB	Byte 1						**LSB**
b7	b6	b5	b4	b3	b2	b1	b0

b(7:1)	Reserved
b(0)	DSI Port Enable: • 0: Enable • 1: Disable

The DSI port must be enabled if using DSI video data. When the port is disabled, it will be held in a state of reset to conserve power. It should be noted that the enable process requires a reconfiguration of the DSI port (similar to that required at display module power-up), including sending the *Write DSI HS Clock* command. If the DLPC343x controller utilized does not support DSI, then this command will have no effect.

3.4.10 Read DSI Port Enable (D8h)

3.4.10.1 Read

This command reads the state of the DSI port enable of the display module.

3.4.10.2 Read Parameters

This command has no command parameters.

3.4.10.3 Return Parameters

Figure 3-70 describes the return parameters.

Figure 3-70. Return Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:1)	Reserved
b(0)	DSI Port Enable: • 0: Enabled • 1: Disabled

3.5 Administrative Commands

3.5.1 Read Short Status (D0h)

3.5.1.1 Read

This command provides a short system status for the display module.

3.5.1.2 Read Parameters

This command has no read parameters.

3.5.1.3 Return Parameters

Table 3-46 describes the return parameters.

Table 3-46. Return Parameters

Parameter Bytes	Description
Byte 1	Short System Status

Figure 3-71. Byte 1 Return Parameters

MSB	Byte 1 – General Status						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7) Boot/main application:
- 0: Boot
- 1: Main

b(6:4) Reserved

b(3) System error:
- 0: No error
- 1: Error

b(2) Reserved

b(1) Communication error:
- 0: No error
- 1: Error

b(0) System initialization:
- 0: Not complete
- 1: Complete

The communication error bit indicates any error on the I^2C command interfaces. Specific details about communication errors are available using the *Read Communication Status* command. Any errors other than a communication error are indicated by the system error bit. Specific details about system errors are available using the *Read System Status* command.

The communication error, and system error bits are cleared when the *Read Short Status* is read. The *Read Short Status* command should only be checked periodically, not continuously. Continuous access may severely impact system performance.

3.5.2 Read System Status (D1h)

3.5.2.1 Read

This command reads system status information for the display module.

3.5.2.2 Read Parameters

This command has no read parameters.

3.5.2.3 Return Parameters

Table 3-47 describes the return parameters.

Table 3-47. Return Parameters

Parameter Bytes	Description
Byte 1	DMD interface status
Byte 2	LED status
Byte 3	Internal interrupt status
Byte 4	Misc. status

All system status error bits are cleared when the read system status is read.

Figure 3-72. Byte 1 Return Parameters

MSB	Byte 1 – DMD Interface Status						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:3) Reserved

b(2) DMD training error:
- 0: No error
- 1: Error

b(1) DMD interface error:
- 0: No error
- 1: Error

b(0) DMD device error:
- 0: No error
- 1: Error

The system sets the DMD device error for the following conditions:
- The system cannot read the DMD device ID from the DMD.
- The system-specified DMD device ID does not match the actual DMD device ID.

The system sets the DMD interface error when there are power management setup conflicts on this interface. The system sets the DMD training error when the training algorithm can not find a data eye that meets the specified requirements.

Figure 3-73. Byte 2 Return Parameters

MSB	Byte 1 – LED Status						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:6) Reserved

b(5) Blue LED error:
- 0: No error
- 1: Error

b(4) Green LED error:
- 0: No error
- 1: Error

b(3) Red LED error:
- 0: No error
- 1: Error

b(2) Blue LED state:
- 0: Off
- 1: On

b(1) Green LED state:
- 0: Off
- 1: On

b(0) Red LED state:
- 0: Off
- 1: On

Figure 3-74. Byte 3 Return Parameters

MSB	Byte 1 – Internal Interrupt Status						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:2) Reserved

b(1) Sequence error:
- 0: No error
- 1: Error

b(0) Sequence abort error:
- 0: No error
- 1: Error

Figure 3-75. Byte 4 Return Parameters

MSB	Byte 1 – Misc. Status						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:6) Reserved

b(5) Watchdog timer timeout:
- 0: No timeout
- 1: Timeout

b(4) Product configuration error:
- 0: No error
- 1: Error

b(3) Master versus slave operation:
- 0: Master
- 1: Slave

b(2) Single versus dual controller configuration:
- 0: Single
- 1: Dual

b(1:0) Reserved

The system sets the product configuration error bit if it determines that some piece of the product configuration is not correct. Some examples are:

- Invalid controller or DMD combination
- Invalid controller or DLPA200x combination
- Invalid flash build for the current controller, DMD, or DLPA200x configuration

The system sets the watchdog timer timeout bit if the system has been reset due to a watchdog timer timeout.

3.5.3 Read System Software Version (D2h)

3.5.3.1 Read

This command reads the main application software version information for the display module.

3.5.3.2 Read Parameters

This command has no read parameters.

3.5.3.3 Return Parameters

Table 3-48 describes the return parameters.

Table 3-48. Return Parameters

Parameter Bytes	Description
Byte 1	Controller main application software version – patch LSByte
Byte 2	Controller main application software version – patch MSByte
Byte 3	Controller main application software version – Minor
Byte 4	Controller main application software version – Major

3.5.4 Read Communication Status (D3h)

3.5.4.1 Read

This command reads system status information for the display module.

3.5.4.2 Read Parameters

The read parameters are described in Table 3-49.

Table 3-49. Read Parameters

Parameter Bytes	Description
Byte 1	Command bus status selection

Figure 3-76. Byte 1 Read Parameters

MSB	Byte 1 – Command Bus Status Selection						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:2) Reserved

b(1:0) Command bus status selection:
- 00: Reserved
- 01: Reserved
- 10: I^2C only
- 11: Reserved

This command returns the communication status for the specified command bus. For I^2C only: This selection returns status bytes 5 though 6.

3.5.4.3 Return Parameters

Table 3-50 describes the return parameters.

Table 3-50. Return Parameters

Parameter Bytes	Description
Byte 5	Communication status
Byte 6	Aborted op-code

All communication status error bits are cleared when the *Read Communication Status* is read.

Figure 3-77. Byte 5 Return Parameters

MSB	Byte 5 – Communication Status						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7) Reserved

b(6) Bus timeout by display error:
- 0: No error
- 1: Error

b(5) Invalid number of command parameters:
- 0: No error
- 1: Error

b(4) Read command error:
- 0: No error
- 1: Error

b(3) Flash batch file error:
- 0: No error
- 1: Error

b(2) Command processing error:
- 0: No error
- 1: Error

b(1) Invalid command parameter value:
- 0: No error
- 1: Error

b(0) Invalid command error:
- 0: No error
- 1: Error

The system sets the invalid command error bit when it does not recognize the command op-code. The invalid command op-code is reported in the I^2C CMD error op-code byte of this status.

The system sets the invalid command parameter error bit when it detects that the value of a command parameter is not valid (for example, out of the allowed range).

The system sets the command processing error bit when a fault is detected when processing a command. In this case, the command aborts and the system moves on to the next command. The op-code for the aborted command is reported in the I^2C CMD error op-code byte of this status.

The system sets the flash batch file error bit when an error occurs during the processing of a flash batch file. When this bit is set, typically another bit is set to indicate what kind of error was detected (for example, an invalid command error).

The system sets the read command error bit when the host terminates the read operation before all of the requested data has been provided, or if the host continues to request read data after all of the requested data has been provided.

The system sets the invalid number of command parameters error bit when too many or too few command parameters are received. In this case, the command aborts and the system moves on to the next command. The op-code for the aborted command is reported in the I^2C CMD error op-code byte of this status.

The system sets the bus timeout by display error bit when the display releases control of the bus after the bus timeout value is exceeded.

Figure 3-78. Byte 6 Return Parameters

MSB	Byte 6 – CMD Error Op-Code						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:0)	I^2C CMD error op-code

The CMD error op-code is associated with various I^2C communication status bits, and reports the op-code for an I^2C command as noted.

3.5.5 Read Controller Device ID (D4h)

3.5.5.1 Read

This command reads the controller device ID for the display module.

3.5.5.2 Read Parameters

This command has no read parameters.

3.5.5.3 Return Parameters

Figure 3-79 describes the return parameters.

Figure 3-79. Return Parameters

MSB	Byte 1						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:4)	Reserved
b(3:0)	Controller device ID

The controller device ID can be decoded using Table 3-51.

Table 3-51. Controller Device ID Decode

Controller Device ID	Device Number	Application
00h	DLPC3430	Embedded (SD)
01h	DLPC3433	Embedded (SD)
02h	DLPC3432	Embedded (SD)
04h	DLPC3435	Standalone (SD)
05h	DLPC3438	Standalone (SD)

Unused controller device ID values are reserved.

3.5.6 Read DMD Device ID (D5h)

3.5.6.1 Read

This command reads the DMD device ID for the display module.

3.5.6.2 Read Parameters

The read parameters are described in Figure 3-80.

Figure 3-80. Read Parameters

MSB	Byte 1 – DMD Register Selection						LSB
b7	b6	b5	b4	b3	b2	b1	b0

b(7:3) Reserved

b(2:0) DMD data selection:
- 0h: DMD device ID
- 1h – 7h: Reserved

3.5.6.3 Return Parameters

Table 3-52 describes the return parameters.

Table 3-52. DMD Device ID Reference Table

DMD Device ID				Device Description
Byte 1 (Identifier)	Byte 2 (Byte Count)	Byte 3 (ID-msbyte)	Byte 4 (ID-lsbyte)	Resolution and Type
60h	0Dh	00h	64h	0.2 WVGA (854x480, Sub-LVDS)
60h	0Dh	00h	89h	0.23 qHD (960x540, Sub-LVDS)
60h	0Dh	00h	68h	0.3 720p (1280x720, Sub-LVDS)

3.5.7 Read Flash Build Version (D9h)

3.5.7.1 Write

This command reads the controller flash version for the display module.

3.5.7.2 Read Parameters

The command has no read parameters.

3.5.7.3 Return Parameters

Table 3-53 describes the return parameters.

Table 3-53. Return Parameters

Parameter Bytes	Description
Byte 1	Flash build version – patch LSByte
Byte 2	Flash build version – patch MSByte
Byte 3	Flash build version – Minor
Byte 4	Flash build version – Major

The OEM specifies a version number for the controller flash build in the format specified by this command. This command allows the OEM to read back this version information.

Revision History

Changes from C Revision (May 2018) to D Revision	Page
• General reformatting and rewriting along with removal of listing most default values (due to various firmware options)	4
• Deleted reference to DSI in the I^2C section	6
• Deleted reference to two I^2C ports	6
• Clarified boot ROM description and requirements	7
• Rewrote and corrected HOST_IRQ initialization Sequence section	7
• Added new Single Buffer Mode command (01h and 02h)	8
• Added support for the Idle Mode Select command (03h and 04h)	9
• Added DSI type to *External Video Source Format Select*	13
• Deleted references to dual-controller operation	24
• Deleted "and the command is be executed" from Read Splash Screen Header description	24
• Deleted information about automatically selecting looks using a light sensor	34
• Added CCA Select command (29h and 2Ah)	36
• Added Parallel Data Mask Control command (37h and 38h)	40
• Deleted automatic LABB adjustment using a light sensor	48
• Added Parallel Sync Polarity command (B6h and B7h)	60
• Added DSI HS Clock command (BDh and BEh)	62
• Added DSI Port Enable command (D7h and D8h)	63

Changes from B Revision (March 2018) to C Revision	Page
• Added DLPC3432 as a supported device	5
• Removed references to WPC enable bit from Section 3.3.3 (Write CAIC Image Processing Control (84h)) and Section 3.3.4 (Read CAIC Image Processing Control (85h)), command not supported	50
• Added controller device IDs for DLPC3433, DLPC3432, and DLPC3438 to Table 3-51	71
• Added device with 0.23 qHD (960x540, Sub-LVDS) to Table 3-52	71

Changes from A Revision (September 2014) to B Revision	Page
• Changed range of CAIC clipping threshold from "0.0% to 2.0%" : to "0.0% to 4.0%"	50

Changes from Original (July 2014) to A Revision	Page
• Added device types DLPC3433 and DLPC3438	5
• Replaced DisplayCrafter with LightCrafter™ Display	6
• Removed Software Command Philosophy section	6
• Removed Write Flash Batch File Delay (DBh) section	72

IMPORTANT NOTICE AND DISCLAIMER

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