Read This First

About This Manual

This document specifies the command and control interface to the DLPC900 controller and defines all applicable commands, default settings, and control register bit definitions.

Related Documents from Texas Instruments

- DLPC900 Data Sheet. DLPS037
- DLP6500FLQ Data Sheet. DLPS040
- DLP6500FYE Data Sheet DLPS053
- DLP9000FLS Data Sheet. DLPS036

If You Need Assistance

See the DLP and MEMS TI E2E Community support forums.

Interface Protocol

This chapter describes the interface protocol between the DLPC900 and a host processor. The DLPC900 supports two host interface protocols: I²C and USB 1.1 slave interfaces.

1 I²C Interface

The DLPC900 controller uses the I²C protocol to exchange commands and data with a host processor. The I²C protocol is a two-wire serial data bus that conforms to the NXP I²C specification. One wire, SCL, serves as a serial clock, while the second wire, SDA, serves as serial data. Several different devices can be connected together in an I²C bus. Each device is software addressable by a unique address. Communication between devices occurs in a simple master-to-slave relationship.

1.1 I²C Transaction Structure

All I²C transactions are composed of a number of bytes, combined in the following order:

START Condition, Slave Address Byte + R/W Bit, Sub-Address Byte, N-Data Bytes, STOP Condition

where N in "N-Data Bytes" varies based on the sub-address.

1.1.1 I²C START Condition

All I²C transactions begin with a START condition. A START condition is defined by a high-to-low transition on the SDA line, followed by a high-to-low transition on the SCL line.

1.1.2 I²C STOP Condition

All I²C transactions end with a STOP condition. A STOP condition is defined by a low-to-high transition on the SDA line, followed by a low-to-high transition on the SCL line.

1.1.3 DLPC900 Slave Address

The DLPC900 offers a programmable slave address. Refer to the App Defaults Settings found in the DLP® LightCrafter™ 6500 & 9000 GUI Firmware tab to set a different slave address. The default I²C settings are shown in Table 1. The Write Slave Address must be an even 7-bit address, and the Read Slave Address must be the Write Slave Address plus 1.

Table 1. I²C Slave Settings

Addressing Mode	Default Write Address	Default Read Address	Maximum Clock Rate (kHz)
7-bit	0x34	0x35	400

1.1.4 DLPC900 Sub-Address and Data Bytes

The DLPC900 I²C sub-address corresponds to the byte address of the DLPC900 commands described in Appendix A. Most I²C sub-addresses have a Read and Write command pair where the Write command equals the Read command with the most significant bit set. For example, Table 2 and Table 4 shows the Input Data Channel Swap sub-address command pair is (0x04,0x84), where the Write sub-address command 0x84 is the Read sub-address command 0x04 with the most significant bit set. Each sub-address command requires a certain number of data bytes, and each command is followed by variable length data where the **least significant byte is first for each parameter**.

NOTE: The DLPC900 I²C command data is formatted with the least significant byte first for each parameter in the data. This maintains the same format with the USB protocol. However, it deviates from the DLPC350 I²C format where the most significant byte is first.

The DLPC900 internal command buffer has a maximum of 512 bytes and it is shared between the Read and Write commands; therefore, whenever a Read command is executed it must be followed by I²C operation with the Read Slave Address to retrieve the data otherwise the data will be overwritten by the next command executed. See Section 1.2 for a Read command example.

1.2 Example I²C Read Command Sequence

To execute a command to read the Input Data Channel Swap setting, the host builds a sequence of bytes containing the slave address, the sub-address, and the data (if any), and performing the following steps.

1. The host performs the required START condition followed by sending the sequence of bytes.
2. The DLPC900 will hold the SCL line low to indicate it is busy.
3. The host waits for the DLPC900 to release the SCL line.
4. Once the SCL line goes high, the host performs a STOP condition.
5. The host then performs a START condition followed by sending the Read Slave Address (0x35), and then reads the required number of bytes and concluding with a STOP condition.

An example of the above read command sequence is shown in Table 2, and a waveform diagram of a host executing this read sequence, is shown in Figure 1 and Figure 2.

Table 2. Read Command Sequence Example(1)

Slave Address	Sub-address	Data
34	04	
35		03

(1) All values shown are in HEX notation.

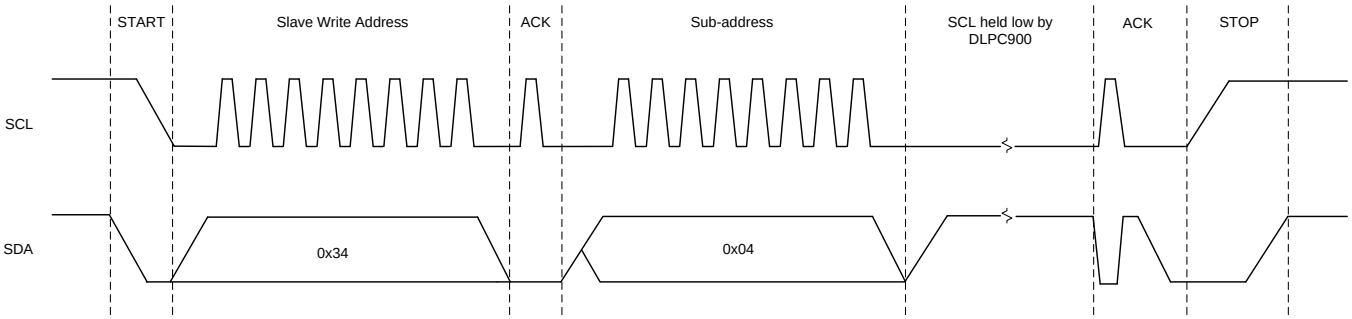

Figure 1. I²C Read Command Waveform Diagram

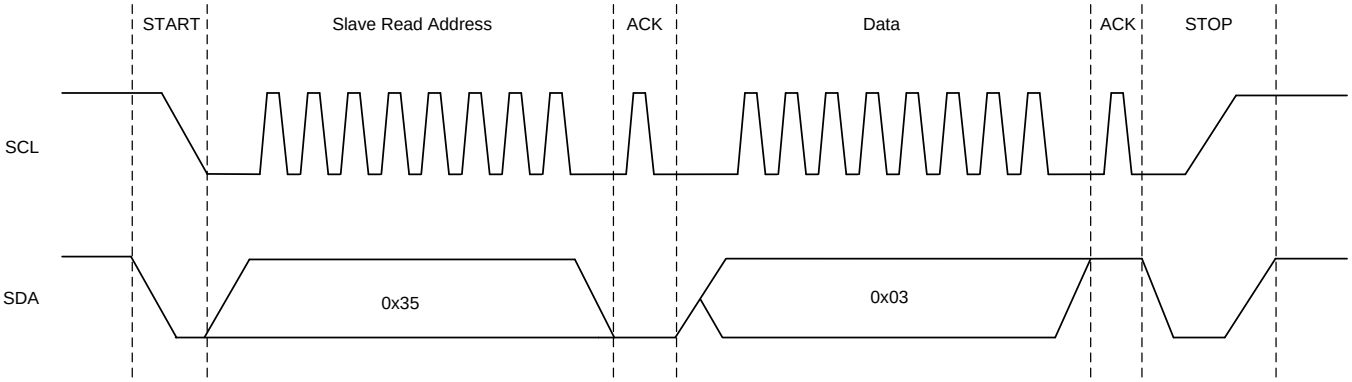

Figure 2. I²C Read Data Waveform Diagram

1.2.1 Read Command Example with Parameters

Some Read sub-address commands require a parameter(s) to be included in the sequence. For example, the command in Section 3.6.1 has multiple GPIO to choose from. Therefore, the GPIO selection parameter must be included in the Read byte sequence in order to retrieve the configuration for the GPIO chosen. Table 3 shows the two I²C operations, where the first row contains the parameter data 06 which indicates GPIO 6. The second row is the returned data of 06 03, where 06 was the chosen GPIO 6 and has a configuration of 03.

Table 3. Read Command with Parameter Sequence Example(1)

Slave Address	Sub-address	Data
34	44	06
35		06 03

(1) All values shown are in HEX notation.

1.3 Example I²C Write Command Sequence

To execute a command to set the Input Data Channel Swap value, the host builds a sequence of bytes containing the slave address, the sub-address, and the data, and performing the following steps.

1. The host performs the required START condition followed by sending the sequence of bytes.
2. The host performs a STOP condition

An example of the above write command sequence is shown in Table 4, and a waveform diagram of a host executing this write sequence, is shown in Figure 3.

Table 4. Write Command Sequence Example(1)

Slave Address	Sub-address	Data
34	84	02

(1) All values shown are in HEX notation.

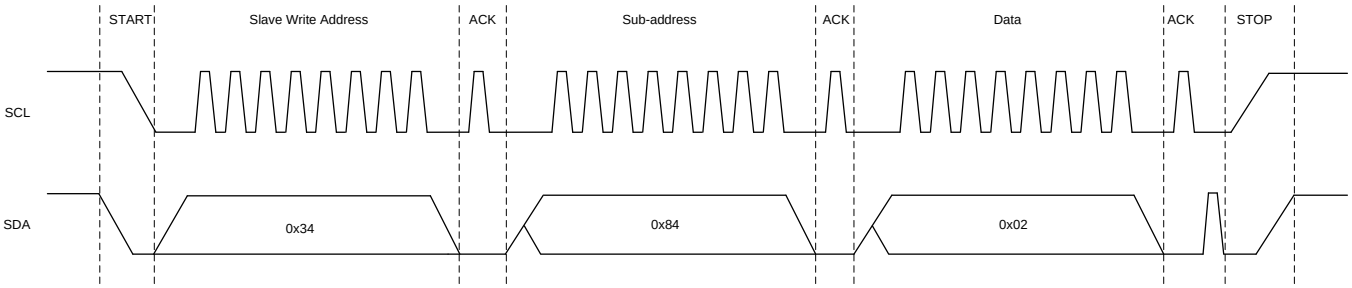

Figure 3. I²C Write Command Waveform Diagram

2 USB Interface

The DLPC900 controller also supports the USB 1.1 human interface device (HID) to exchange commands and data with a host processor. The USB commands are variable length data packets that are sent with the **least significant byte first for each parameter**.

2.1 USB Transaction Sequence

The USB 1.1 HID protocol has the structure shown in Figure 4. The host must build a stream of bytes that consist of the Report ID, Header, and the payload. The following is a description of these three parts.

Report ID: The Report ID is always set to 0 and always the leading byte of all transfers.

Header: The header consists of four bytes.

1) Flag Byte as shown in Figure 4 and described in the Read and Write examples. Section 2.2

2) Sequence Byte. The sequence byte can be a rolling counter. It is used primarily when the host wants a response from the DLPC900. The DLPC900 will respond with the same sequence byte that the host sent. The host can then match the sequence byte from the command it sent with the sequence byte from the DLPC900 response.

3) Length: Two byte length denotes the number of data bytes in the Payload only.

Payload Bytes: The payload bytes consists of the USB command followed by the data that is associated with the command.

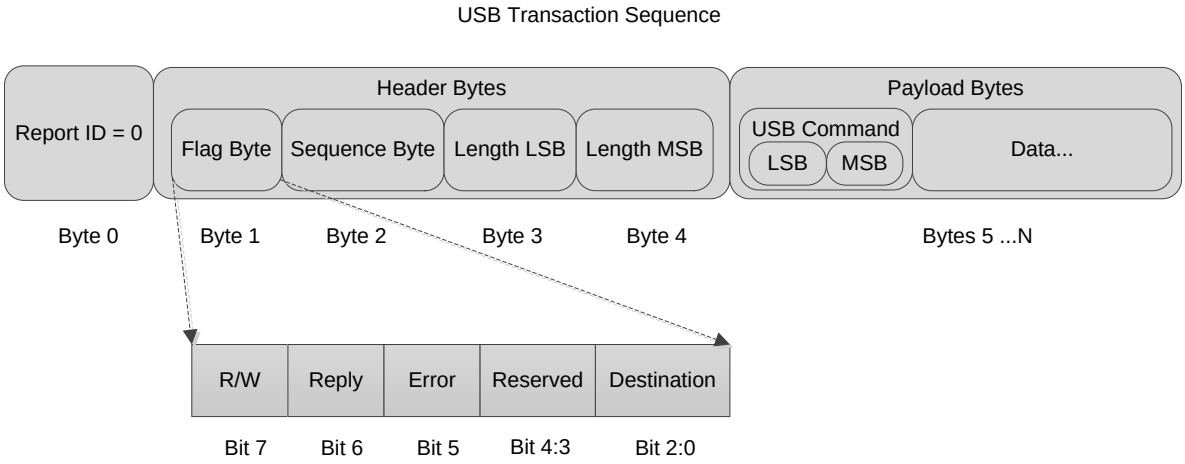

Figure 4. USB HID Protocol

During a Write operation, the host transmits the entire transaction sequence to the DLPC900, and the DLPC900 performs the operation associated with the Write command. During a Read operation, the host transmits the entire transaction sequence to the DLPC900, and the DLPC900 performs the operation associated with the Read command. Therefore, both Write and Read transactions are considered "writes" to the DLPC900 where the host performs an API level "Writefile" to the HID driver. The difference is when the DLPC900 executes a Read operation, where the DLPC900 places the response into its internal buffer and waits for the host to perform an API level "Readfile" to the HID driver and only then does the DLPC900 transmits the response data back to the host.

The DLPC900 internal command buffer has a maximum of 512 bytes and it is shared between both the Write and Read operations; therefore, whenever the host performs a Read operation, it must be followed by the "Readfile" to the HID driver to get the response otherwise the response data will be overwritten by the next Write or Read operation.

The HID protocol is limited to 64 byte transfers in both directions. Therefore, commands that are larger than 64 bytes require multiple transfers. Whenever such a command is used, only the very first transfer requires the Header and the USB Command. The Report ID is always the leading byte of all transfers. Figure 5 shows an example of a Write command that contains 76 bytes and requires two transfers. Notice that the first transfer contains 65 bytes, which is correct. The host hardware level HID driver will extract the Report ID before transmitting or receiving the data over the USB bus.

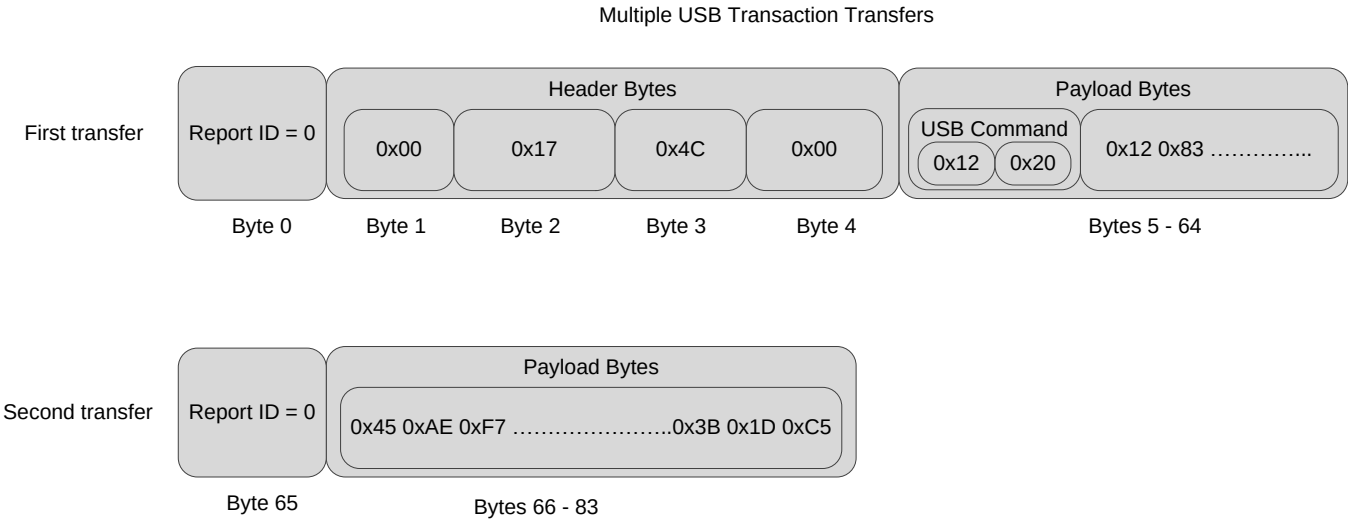

Figure 5. USB Multi-Transfer Transaction

2.2 USB Read Transaction Sequence Example

To perform a Read operation on the DLPC900, the host must assemble a sequence of bytes that corresponds to the command being used. The following Table 5 shows an example on how to read the curtain color intensity of each color.

Table 5. Read Operation Example(1)

Report ID Byte	Flag Byte	Sequence Byte	Length(2)	USB Command(2)
00	C0	11	02 00	00 11

(1) All values shown are in HEX notation.
(2) LSB precedes the MSB for each parameter.

1. Report ID byte. Always set to 0.
2. Flag byte. Where:
   - Bits 2:0 are set to 0x0 for regular DLPC900 operation.
   - Bit 6 is set to 0x1 to indicate the host wants a reply from the device

- Bit 7 is set to 0x1 to indicate a read transaction

3. Sequence byte: The sequence byte can be a rolling counter. It is used primarily when the host wants a response from the DLPC900. The DLPC900 will respond with the same sequence byte that the host sent. The host can then match the sequence byte from the command it sent with the sequence byte from the DLPC900 response.
4. Length: Two byte length denotes the number of data bytes in the sequence and excludes the number of bytes in steps 1 through 4. It denotes the total number of bytes sent in steps 5 (command bytes).
5. USB Command: Two byte USB command.
6. Once the host transmits the data over the USB interface, the DLPC900 will respond to the Read operation by placing the response data in its internal buffer. The host must then perform a HID driver read operation. Table 6 shows the response data sent back from the DLPC900.

   (a) Report ID: Always set to 0.

   (b) Flag byte: The same as was sent plus error bit. The host may check the error flag (bit 5) as follows.

   (i) 0 = No errors.

   (ii) 1 = Command not found or command failed.

   (c) Sequence byte: The same as was sent. The host may match the sent sequence byte with the response sequence byte.

   (d) Length: Number of data bytes. The host must assemble the data according to the definition of the command.

Table 6. Read Response Example(1)

Report ID Byte	Flag Byte	Sequence Byte	Length(2)	Data(2)
00	C0	11	06 00	FF 01 FF 01 FF 01

(1) All values shown are in HEX notation.
(2) LSB precedes the MSB for each parameter.

2.3 USB Write Transaction Sequence Example

To perform a Write operation on the DLPC900, the host must assemble a sequence of bytes that corresponds to the command being used. The following Table 7 shows an example on how to set the curtain color intensity of each color to 511.

Table 7. Write Operation Example(1)

Report ID Byte	Flag Byte	Sequence Byte	Length(2)	USB Command(2)	Data(2)
00	00	12	08 00	00 11	FF 01 FF 01 FF 01

(1) All values shown are in HEX notation.
(2) LSB precedes the MSB for each parameter.

1. Report ID byte. Always set to 0.
2. Flag byte. Where :
   - Bits 2:0 are set to 0x0 for regular DLPC900 operation.
   - Bit 6 is set to 0x0 to indicate the host does not want a reply from the device. This bit is set to 0x1 only if a reply is needed, which is usually not required.
   - Bit 7 is set to 0x0 to indicate a write transaction
3. Sequence byte: The sequence byte can be a rolling counter. It is used primarily when the host wants a response from the DLPC900. Normally during a write operation, the DLPC900 does not respond; however, the host can continue to increment the sequence byte for the next command operation.
4. Length: Two byte length denotes the number of data bytes in the sequence and excludes the number of bytes in steps 1 through 4. It denotes the total number of bytes sent in steps 5 (command bytes) and 6 (data bytes).

5. USB Command: Two byte USB command.
6. Data: The data appropriate to the command.

DLPC900 Control Commands

This chapter lists the DLPC900 control commands.

The following sections list the supported control commands of the DLPC900. In the *Type* column, 'wr' type is writeable field through I²C or USB write transactions. Data can also be read through I²C or USB read transactions for 'wr' type bits. Type *r* is read-only. Write transactions to read-only fields are ignored.

The Reset column in all of the following command tables is the default value after power up. These values may be overwritten after power up.

NOTE: **Reserved bits and registers.** When writing to valid command bit fields, all bits marked as unused or reserved should be set to 0, unless specified otherwise.

NOTE: **Momentary Image Corruption During Command Writes**. Certain commands may cause brief visual artifacts in the display image under some circumstances. Command data values may always be read without impacting displayed image. To avoid momentary image corruption due to a command, disable the LEDs prior to the command write, then reenable the LEDs after all commands have been issued.

NOTE: **Writing or reading from undocumented registers is NOT recommended**.

1 DLPC900 Status Commands

The DLPC900 has the following set of status commands:

- Hardware Status
- System Status
- Main Status
- Retrieve Firmware Version
- Read Error Codes

1.1 Hardware Status

The Hardware Status command provides status information on the DLPC900's sequencer, DMD controller, and initialization.

Command

I²C	USB
Read	0x1A0A
0x20	

Table 8. Hardware Status Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	0	Internal Initialization 0 = Error 1 = Successful	d1	r
	1	0 = No Error 1 = Incompatible Controller or DMD	d0	r
	2	DMD Reset Controller Error 0 = No error has occurred 1 = Multiple overlapping bias or reset operations are accessing the same DMD block.	d0	r
	3	Forced Swap Error 0 = No error has occurred. 1 = Forced Swap Error occurred.	d0	r
	4 (1)	0 = No Slave Controller Present 1 = Slave Controller Present and Ready	d0	r
	5	Reserved	d0	r
	6	Sequencer Abort Status Flag 0 = No error has occurred 1 = Sequencer has detected an error condition that caused an abort	d0	r
	7	Sequencer Error 0 = No error has occurred. 1 = Sequencer detected an error.	d0	r

(1) When the DLPC900 is combined with a DLP6500, this bit will be 0. When two DLPC900 controllers are combined with a DLP9000, this bit must be 1 for proper operation. If the bit is 0 and the DLPC900 is combined with a DLP9000, indicates a malfunction in one or both controllers.

NOTE: Any error condition indicates a fault condition and it must be corrected.

1.2 System Status

The System Status command provides DLPC900 status on internal memory tests.

Command

I²C	USB
Read	0x1A0B
0x21	

Table 9. System Status Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	0	Internal Memory Test 0 = Internal Memory Test failed 1 = Internal Memory Test passed	d1	r
	1:7	Reserved	d0	r

1.3 Main Status

The Main Status command provides the status of DMD park and DLPC900 sequencer, frame buffer, and gamma correction.

Command

I²C	USB
Read	0x1A0C
0x22	

Table 10. Main Status Command Definition

BITS	BITS	DESCRIPTION	RESET	TYPE
0	0	DMD Park Status 0 = DMD micromirrors are not parked 1 = DMD micromirrors are parked	d1	r
	1	Sequencer Run Flag 0 = Sequencer is stopped 1 = Sequencer is running normally	d0	r
	2	Video Frozen Flag 0 = Video is running (Normal frame change) 1 = Video is frozen (Displaying single frame)	d0	r
	7:3	Reserved	d0	r

1.4 Retrieve Firmware Version

This command reads the version information of the DLPC900 firmware.

Command

I²C	USB
Read	0x0205
0x11	

Table 11. Get Version Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
3:0		Application software revision:	Will match firmware version.	r
	15:0	Application software patch number		
	23:16	Application software minor revision		
	31:24	Application software major revision		
7:4		API software revision:	d0	r
	15:0	API patch number		
	23:16	API minor revision		
	31:24	API major revision		
11:8		Software configuration revision:	d0	r
	15:0	Software configuration patch number		
	23:16	Software configuration minor revision		
	31:24	Software configuration major revision		
15:12		Sequencer configuration revision:	d0	r
	15:0	Sequencer configuration patch number		
	23:16	Sequencer configuration minor revision		
	31:24	Sequencer configuration major revision		

1.5 Reading Error Codes and Description

This Commands retrieves the error codes that are generated by the DLPC900.

1.5.1 Read Error Code

This command retrieves the error code number from the DLPC900 of the last executed command.

Command

I²C	USB
Read	0x0100
0x32	

Table 12. Read Error Code Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	0	No error	0	r
	1	Batch file checksum error		
	2	Device failure		
	3	Invalid command number		
	4	In compatible controller / DMD		
	5	Command not allowed in current mode		
	6	Invalid command parameter		
	7	Item referred by the parameter is not present		
	8	Out of resource (RAM / Flash)		
	9	Invalid BMP compression type		
	10	Pattern bit number out of range		
	11	Pattern BMP not present in flash		
	12	Pattern dark time is out of range		
	13	Signal delay parameter is out of range		
	14	Pattern exposure time is out of range		
	15	Pattern number is out of range		
	16	Invalid pattern definition (errors other than 9-15)		
	17-254	Not defined		
	255	Internal Error		

1.5.2 Read Error Description

This command retrieves the error descriptive string from the DLPC900 of the last executed command. The string is composed of character bytes ending with a null termination character.

Command

I²C	USB
Read	0x0101
0x33	

Table 13. Read Error Description Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
127:0	All	Error description for the last executed command. 0 terminated string of character bytes.	0	r

2 DLPC900 Firmware Programming Commands

The Programming commands manage downloading a new firmware image into flash memory. This can be done over I²C or USB interfaces. The commands in the DLPC900 Programming Commands section are only valid in **program mode** except for Enter Program Mode (**I²C**: 0x30 or **USB** 0x3001), which exits normal mode and enters program mode. Once in program mode, the user must issue the proper Exit Program Mode (**I²C**: 0x30 or **USB** 0x0030) command to return to normal mode. While in program mode, commands outside of this section will not work.

2.1 Read Status

This command indicates if the flash is ready to be programmed and also if a flash operation is in progress.

Command

I²C	USB
Read	0x0000
0x23	

Table 14. Read Status Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	2:0	Reserved	d0	r
	3	Busy Bit 0 = No flash operation in progress 1 = Flash operation in progress		
	6:4	Reserved		
	7	Programming Mode Bit 0 = Does not allow flash programming operations 1 = Allows flash programming operations		

2.2 Enter Program Mode

This command tells the controller to enter its programming mode and jump to the boot loader. If the boot loader receives this command, then the command has no effect.

Command

I²C	USB
Write	0x3001
0x30	

Table 15. Enter Program Mode Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	0	Program Mode 0 = Enter Program Mode – Jump to boot loader	d0	w
	7:1	Reserved		

2.2.1 Exit Program Mode

This command tells the controller to exit its programming mode. If the application receives the exit command, the command has no effect.

Command

I²C	USB
Write	0x0030
0x30	

Table 16. Exit Program Mode Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	0	Program Mode 1 = Exit Program Mode – Reset controller and run application	d0	w
	7:1	Reserved		

2.3 Read Control

This command reads the Flash Manufacturer and Device IDs, as well as the Checksum, after the Calculate Checksum command is executed.

Command

I²C	USB
Read	0x0015
0x15	

Table 17. Query Flash IDs Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	3:0	ID 0 = Returns Checksum C = Requests Flash Manufacturer ID D = Requests Flash Device ID	d0	r
	7:4	Reserved		

2.4 Start Address

The Start Address command serves three purposes.

1) Specifies the start address of the flash download write operation. It is the responsibility of the user to ensure that the start address is on a sector boundary in the current flash device.

2) Specifies the start address where checksum operation begins.

3) Specifies the sector address to be erased. The address should be the start of a sector.

The Flash Data Size command should always follow 1 and 2 above, which defines how many bytes to be downloaded or how many bytes to include for the checksum operation.

The user must avoid erasing the first 64k bytes of the boot flash as this contains the boot image. The user must also avoid erasing other sectors that contain firmware that are required for proper controller operation.

Command

I²C	USB
Read	0x0032
0x32	

Table 18. Start Address Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
3:0	31:0	4 byte flash address. Valid Range : 0xF8000000 – 0xFAFFFFFF. Byte 0 is LSB, byte 4 is MSB.	x0	w

2.5 Erase Sector

This is a system write command to erase a sector of flash memory. This command should not be executed until valid data has been written to the Flash Start Address. Users are responsible for ensuring that a valid address has been written. The Busy bit will be set in the Boot Loader status byte while the sector erase is in progress. There is no data associated with this command.

Command

I²C	USB
Write	0x0028
0x28	

NOTE: TI cautions against erasing the boot sector of the device as this contains key initialization parameters and the flash programming functionality. Only the sector that contains the start address will be erased, not all sectors from the start address to the end of the device. Users must either pre-erase all sectors to be programmed, or erase and program each sector individually.

2.6 Download Flash Data Size

System write command to specify the size of the following flash download. The data size is sent to tell the Boot Loader how many bytes to expect to program into the flash device. It is also used for specifying the checksum range when requesting that operation.

Command

I²C	USB
Write	0x0033
0x33	

Table 19. Download Data Size Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
3:0	32:0	4 Byte flash size. Valid Range 4 - 0x2FFFFFF. Byte 0 is LSB, byte 3 is MSB.	x0	w

2.7 Download Data

This command contains the flash data to be programmed. The maximum data size which can be sent in each command is 512 bytes, which corresponds to a data length of 514. The number of bytes downloaded by consecutive download data commands must match the predefined Flash Data Size for the operation to be successful.

Command

I²C	USB
Write	0x0025
0x25	

Table 20. Download Data Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	7:0	Length LSB	x0	w
1	7:0	Length MSB		
513:2	4095:0	Up to 512 Data Bytes		
514	7:0	Checksum		

2.8 Calculate Checksum

This command calculates the checksum. Executing this command causes the Boot Loader to read the data in the flash memory and calculate a 4-byte 8-bit checksum. The Busy bit will be set in the Boot Loader status byte while the checksum computation is in progress. After completion, the 4-byte checksum can be read back through the Read Control command. The data range to be summed is specified by writing appropriate data with the Flash Start Address and Flash Data Size commands. There is no data associated with this command.

Command

I^2C	USB
Write	0x0026
0x26	

2.9 Dual Controller Control

This command stops the given controller from executing any further commands until enabled by the same command. This command is intended to be used when two DLPC900 controllers are combined with one DLP9000 DMD, where one controller is the master and the other is the slave.

Command

I^2C	USB
Write	0x0031
0x31	

Table 21. Dual Controller Control Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	0	1 – Disable Master Controller	x0	w
		0 – Enable Master Controller		
	1	1 – Disable Slave Controller	x0	w
		0 – Enable Slave Controller		

3 Chipset Control Commands

The DLPC900 I^2C and USB control commands are accepted in any order, except when special sequencing is required (for example, setting up the flash). Each control command is validated for sub-address and parameter errors as it is received. Commands failing validation are ignored. On power up, it is necessary to wait for DLPC900 to complete its initialization before sending any I^2C or USB commands.

3.1 Chipset Configuration Commands

The Chipset and Configuration commands manage software reset, power modes, and image curtain display.

3.1.1 Power Mode

The Power Control places the DLPC900 in a standby state and powers down the DMD interface. Standby mode should only be enabled after all data for the last frame to be displayed has been transferred to the DLPC900. Standby mode must be disabled prior to sending any new data. After executing this command, the host may poll the system status using I²C commands: 0x20, x21, and 0x22 or USB commands: 0x1A0A, 0x1A0B, and 0x1A0C to attain status.

Command

I²C		USB
Read	Write	0x0200
0x07	0x87	

Table 22. Power Mode Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	1:0	Power Mode 0 = Normal operation. The selected external source will be displayed 1 = Standby mode. Places DLPC900 in standby state and powers down the DMD interface 2 = Perform a software reset	d0	wr
	7:2	Reserved	d0	r

3.1.2 Curtain Color

This register provides image curtain control. When enabled and the input source is set to external video with no video source connected, a solid color field is displayed on the entire DMD display. The Display Curtain Control provides an alternate method of masking temporary source corruption from reaching the display due to on-the-fly reconfiguration. It is also useful for optical test and debug support.

Command

I²C		USB
Read	Write	0x1100
0x06	0x86	

Table 23. Display Curtain Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
1:0	9:0	Red color intensity in a scale from 0 to 1023	d0	wr
3:2	9:0	Green color intensity in a scale from 0 to 1023	d0	wr
5:4	9:0	Blue color intensity in a scale from 0 to 1023	d0	wr

3.2 Parallel Interface Configuration

The Parallel Interface Configuration manages the operation of the RGB parallel interface.

3.2.1 Input Data Channel Swap

The Input Data Channel Swap commands configure the specified input data ports and maps the data subchannels. The DLPC900 interprets Channel A as Green, Channel B as Red, and Channel C as Blue.

Command

I²C		USB
Read	Write	0x1A37
0x04	0x84	

Table 24. Input Data Channel Swap Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	0	Port Number 0 – Port 1 1 – Port 2	0	w
	3:1	Swap Parallel Interface Data Subchannel: 0 - ABC = ABC, No swapping of data subchannels 1 - ABC = CAB, Data subchannels are right shifted and circularly rotated 2 - ABC = BCA, Data subchannels are left shifted and circularly rotated 3 - ABC = ACB, Data subchannels B and C are swapped 4 - ABC = BAC, Data subchannels A and B are swapped 5 - ABC = CBA, Data subchannels A and C are swapped 6 - Reserved 7 - Reserved	d4	wr
	7:4	Reserved	d0	r

3.3 Input Source Commands

The Input Source Selection determines the input source for the DLPC900 data display.

3.3.1 Port and Clock Configuration

This command selects which port the RGB data is on and which pixel clock, data enable, and syncs to use. The user must select the correct port and clock configuration according to the PCB layout routing.

Command

I²C		USB
Read	Write	0x1A03
0x03	0x83	

Table 25. Port and Clock Configuration Command

BYTE	BITS	DESCRIPTION (1) (2)	RESET	TYPE
0	1:0	0 - Data Port 1, Single Pixel mode 1 - Data Port 2 , Single Pixel mode 2 - Data Port 1-2, Dual Pixel mode. Even pixel on port 1, Odd pixel on port 2 3 - Data Port 2-1 Dual Pixel mode. Even pixel on port 2, Odd pixel on port 1	d0	wr
	3:2	0 - Pixel Clock 1 1 - Pixel Clock 2 2 - Pixel Clock 3 3 - Reserved		
	4	0 - Data Enable 1 1 - Data Enable 2		
	5	0 - P1 VSync and P1 HSync 1 - P2 VSync and P2 HSync		
	7:6	Reserved		

(1) Single Pixel refers to the parallel data is connected to port 1 or port 2 and the input source pixel clock is less than 175MHz. Both ports cannot be used simultaneous in single pixel mode.
(2) Dual Pixel refers to the parallel data is connected to port 1 and port 2 and the input source pixel clock is less than 141MHz.

3.3.2 Input Source Configuration

The Input Source Configuration command selects the input source to be displayed by the DLPC900: 30-bit parallel port, Internal Test Pattern or flash memory. After executing this command, the host may poll the system status using I^2C commands: 0x20, 0x21, and 0x22, or the respective USB commands: 0x1A0A, 0x1A0b, and 0x1A0C.

Command

I^2C		USB
Read	Write	0x1A00
0x00	0x80	

Table 26. Input Source Configuration Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	2:0	Select the input source and interface mode: 0 = Primary parallel interface with 16-bit, 20-bit, 24-bit, or 30-bit RGB or YUV data formats. 1 = Internal test pattern generator. 2 = Flash. Images are 24-bit single-frame, still images stored in flash that are uploaded on command. 3 = Solid curtain.	d0	wr
	4:3	Parallel Interface bit depth 0 = 30 bits 1 = 24 bits 2 = 20 bits 3 = 16 bits	d1	wr
	7:5	Reserved	d0	r

3.3.3 Input Pixel Data Format

The Input Pixel Data Format command defines the pixel data format input into the DLPC900.

Command

I²C		USB
Read	Write	0x1A02
0x02	0x82	

Table 27. Input Pixel Data Format Command Definition

BYTE	BITS	DESCRIPTION				RESET	TYPE
0	3:0	Select the pixel data format:	Supported Pixel Formats vs Source Type			d0	wr
			Parallel	Test Pattern	Flash Image		
		0 - RGB 4:4:4 (30 bit)	Yes	Yes	Yes		
		1 - YCrCb 4:4:4 (30 bit)	Yes	No	No		
		2 - YCrCb 4:2:2	Yes	No	Yes		
	7:6	Reserved				d0	r

3.3.4 Internal Test Pattern Select

When the internal test pattern is the selected input, the Internal Test Pattern Select defines the test pattern displayed on the screen. These test patterns are internally generated; therefore, all image processing is performed on the test images. The resolution of the Test Pattern will be native to the DLP6500 or the DLP9000.

Command

I²C		USB
Read	Write	0x1203
0x0A	0x8A	

Table 28. Internal Test Patterns Select Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	3:0	Internal Test Patterns Select: 0 = Solid field 1 = Horizontal ramp 2 = Vertical ramp 3 = Horizontal lines 4 = Diagonal lines 5 = Vertical lines 6 = Grid 7 = Checkerboard 8 = RGB ramp 9 = Color bars 10 = Step bars	d8	wr
	7:4	Reserved		

3.3.5 Internal Test Patterns Color

When the internal test pattern is the selected input, the Internal Test Patterns Color Control defines the colors of the test pattern displayed on the screen. These test patterns are internally generated; therefore, all image processing is performed on the test images. All command registers should be set up as if the test images are input from an RGB 8:8:8 external source. The foreground color setting affects all test patterns. The background color setting affects those test patterns that have a foreground and background component, such as, Horizontal Lines, Diagonal Lines, Vertical Lines, Grid, and Checkerboard.

Command

I²C		USB
Read	Write	0x1204
0x1A	0x9A	

Table 29. Internal Test Patterns Color Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
1:0	9:0	Red Foreground Color intensity in a scale from 0 to 1023 0x0 = No Red Foreground color intensity ... 0x3FF = Full Red Foreground color intensity	x3FF	wr
3:2	9:0	Green Foreground Color intensity in a scale from 0 to 1023 0x0 = No Green Foreground color intensity ... 0x3FF = Full Green Foreground color intensity	x3FF	wr
5:4	9:0	Blue Foreground Color intensity in a scale from 0 to 1023 0x0 =No Blue Foreground color intensity ... 0x3FF = Full Blue Foreground color intensity	x3FF	wr
7:6	9:0	Red Background Color intensity in a scale from 0 to 1023 0x0 = No Red Background color intensity ... 0x3FF = Full Red Background color intensity	x0	wr
9:8	9:0	Green Background Color intensity in a scale from 0 to 1023 0x0 = No Green Background color intensity ... 0x3FF = Full Green Background color intensity	x0	wr
11:10	9:0	Blue Background Color intensity in a scale from 0 to 1023 0x0 = No Blue Background color intensity ... 0x3FF = Full Blue Background color intensity	x0	wr

3.3.6 Load Image

This command loads an image from flash memory and then displayed on the DMD. After executing this command, the host may poll the system status using I²C commands: 0x20, 0x21, and 0x22 or the respective USB commands: 0x1A0A, 0x1A0B, and 0x1A0C.

Command

I²C		USB
Read	Write	0x1A39
0x7F	0xFF	

Table 30. Load Image Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	7:0	Image Index. Loads the image at this index. Reading this back provides the index that was loaded most recently through this command.	d0	wr

3.4 Image Flip

The DLPC900 supports long- and short-axis image flips to support rear- and front-projection, as well as, table- and ceiling-mounted projection.

NOTE: If showing image from Flash, load image (I^2C: 0x7F, USB: 0x1A39), this must be called to update the image flip setting.

3.4.1 Long-Axis Image Flip:

The Long-Axis Image Flip defines whether the input image is flipped across the long axis of the DMD. If this parameter is changed while displaying a still image, the input still image should be re-sent. If the image is not re-sent, the output image might be slightly corrupted. Figure 6 shows an example of a long-axis image flip. In Structured Light mode, the image flip will take effect on the next bit-plane, image, or video frame load. **The DLPC900 does not support long axis image flip when combined with a DLP9000 DMD**.

Command

I^2C		USB
Read	Write	0x1008
0x08	0x88	

Table 31. Long Axis Image Flip Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	0	Flips image along the long side of the DMD: 0 = Disable flip 1 = Enable flip	d0	wr
	7:1	Reserved	d0	r

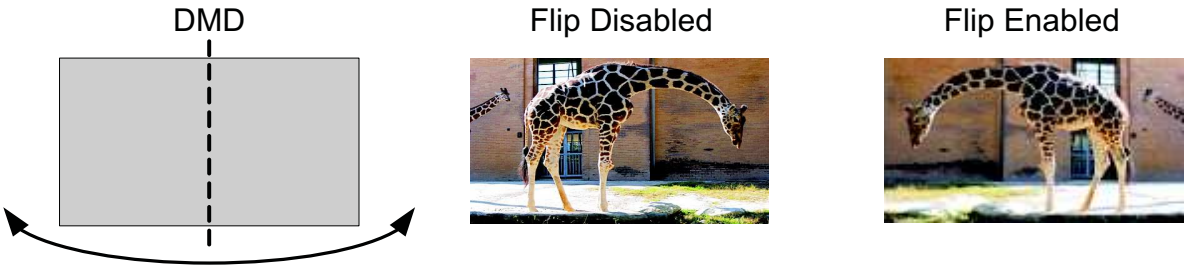

Figure 6. Image Long-Axis Flip Example

3.4.2 Short Axis Image Flip

The Short-Axis Image Flip defines whether the input image is flipped across the short axis of the DMD. If this parameter is changed while displaying a still image, the input still image should be resent. If the image is not re-sent, the output image might be slightly corrupted. Figure 7 shows an example of a short axis image flip. In Structured Light mode, the image flip will take effect on the next bit-plane, image, or video frame load.

Command

I^2C		USB
Read	Write	0x1009
0x09	0x89	

Table 32. Short-Axis Image Flip Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	0	Flips image along the short side of the DMD: 0 - Disable flip 1 - Enable flip	d0	wr
	7:1	Reserved	d0	r

Figure 7. Image Short-Axis Flip Example

3.4.3 IT6535 Power Mode

The IT6535 Power Mode command allows the user to power-down and tri-state the IT6535 digital receiver data and sync outputs. This command is ignored if the IT6535 is not present or has been disabled in the App Defaults Settings found in the DLP LightCrafter 6500 & 9000 GUI Firmware tab.

Command

I^2C		USB
Read	Write	0x1A01
0x0C	0x8C	

Table 33. IT6535 Power Mode Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	0	0 = Power-Down. (Outputs will be tri-stated) 1 = Power-Up for HDMI input. 2 = Power-Up for DisplayPort input.	d0	wr
	7:1	Reserved.	d0	r

3.5 LED Driver Commands

LED driver operation is a function of the individual red, green, and blue LED-enable software-control parameters. The recommended order for initializing LED drivers is to:

1. Program the individual red, green, and blue LED driver currents.
2. Program the LED PWM polarity.
3. Enable the individual LED enable outputs.
4. Turn ON the DLP display sequence (see Section 4.1).

The LED-current software-control parameters define PWM values that drive corresponding LED current. The LED enables indicate which LED will be activated.

CAUTION

Careful control of LED current is needed to prevent damage to LEDs. Follow all LED manufacturer recommendations and maintain LED current levels within recommended operating conditions. The setting of the LED current depends on many system and application parameters (including projector thermal design, LED specifications, selected display mode, and so forth). Therefore, the recommended and absolute-maximum settings vary greatly.

3.5.1 LED Enable Outputs

The DLPC900 offers three sets of pins to control the LED enables:

- LEDR_EN for the red LED
- LEDG_EN for the green LED
- LEDB_EN for the blue LED

After reset, all LED enables are placed in the inactive state until the board initializes.

Command

I^2C		USB
Read	Write	0x1A07
0x10	0x90	

Table 34. LED Enable Outputs Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	0	Red LED Enable 0 - Red LED is disabled 1 - Red LED is enabled	d0	wr
	1	Green LED Enable 0 - Green LED is disabled 1 - Green LED is enabled	d0	wr
	2	Blue LED Enable 0 - Blue LED is disabled 1 - Blue LED is enabled	d0	wr
	3	LED Enable Control 0 - All LED enables are controlled by bits 2:0 and ignore Sequencer control 1 - All LED enables are controlled by the Sequencer and ignore the settings in bits 2:0	d1	wr
	7:4	Reserved	d0	r

3.5.1.1 LED PWM Polarity

The LED PWM Polarity command sets the polarity of all PWM signals. This command must be issued before powering up the LED drivers.

Command

I^2C		USB
Read	Write	0x1A05
0x0B	0x8B	

Table 35. LED PWM Polarity Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	1:0	Polarity of PWM signals[1] 0 - Normal polarity, PWM 0 value corresponds to no current while PWM 255 value corresponds to maximum current. 1 - Inverted polarity. PWM 0 value corresponds to maximum current while PWM 255 value corresponds to no current.	d0	wr
	7:2	Reserved	d0	r

(1) Depending on the LED driver design, the polarity chosen may have an opposite affect.

3.5.2 LED Driver Current

This parameter controls the pulse duration of the specific LED PWM modulation output pin. The resolution is 8 bits and corresponds to a percentage of the LED current. The PWM value can be set from 0 to 100% in 256 steps. If the LED PWM polarity is set to normal polarity, a setting of 0xFF gives the maximum PWM current. The LED current is a function of the specific LED driver design.

Command

I²C		USB
Read	Write	0x0B01
0x4B	0xCB	

CAUTION

Care should be taken when using this command. Improper use of this command can lead to damage to the system. The setting of the LED current depends on many system and application parameters (including projector thermal design, LED specifications, selected display mode, and so forth). Therefore, recommended and absolute-maximum settings vary greatly.

Table 36. LED Driver Current Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	7:0	Red LED PWM current control Valid range, assuming normal polarity of PWM signals, is: 0x00 (0% duty cycle → Red LED driver generates no current) to 0xFF (100% duty cycle → Red LED driver generates maximum current)) The current level corresponding to the selected PWM duty cycle is a function of the specific LED driver design and thus varies by design.	x97	wr
1	7:0	Green LED PWM current control Valid range, assuming normal polarity of PWM signals, is: 0x00 (0% duty cycle → Green LED driver generates no current) to 0xFF (100% duty cycle → Green LED driver generates maximum current)) The current level corresponding to the selected PWM duty cycle is a function of the specific LED driver design and thus varies by design.	x78	wr

Table 36. LED Driver Current Command Definition (continued)

BYTE	BITS	DESCRIPTION	RESET	TYPE
2	7:0	Blue LED PWM current control Valid range, assuming normal polarity of PWM signals, is: 0x00 (0% duty cycle → Blue LED driver generates no current) to 0xFF (100% duty cycle → Blue LED driver generates maximum current)) The current level corresponding to the selected PWM duty cycle is a function of the specific LED driver design and thus varies by design.	x7D	wr

3.6 GPIO Commands

DLPC900 offers 9 general-purpose input/output pins (GPIO). Some of these pins can be configured for PWM output, PWM input, or clock output functionality. By default, all pins are configured as GPIO inputs.

3.6.1 GPIO Configuration

The GPIO Configuration command enables GPIO functionality on a specific set of DLPC900 pins. The command sets their direction, output buffer type, and output state.

Command

I²C		USB
Read	Write	0x1A38
0x44	0xC4	

Table 37. GPIO Configuration Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	7:0	GPIO selection. See Table 38 for description of available pins	d0	wr
1	0	Output state 0 = Low 1 = High	d0	wr
	1	1 – Configure pin as output 0 – Configure pin as input	d0	wr
	2	1 – Configure as open drain mode 0 – Configure as normal mode	d0	wr
	7:3	Reserved	d0	r

Table 38. GPIO Selection

GPIO Selection	DLPC900 GPIO Pin	Function	Alternate Function
0	GPIO_PWM_0	GPIO	PWM Output
1	GPIO_PWM_1	GPIO	PWM Output
2	GPIO_PWM_2	GPIO	PWM Output
3	GPIO_PWM_3	GPIO	PWM Output
4	GPIO_4	GPIO	None
5	GPIO_5	GPIO	None
6	GPIO_6	GPIO	None
7	GPIO_7	GPIO	None
8	GPIO_8	GPIO	None

3.6.2 GPIO Clock Configuration

DLPC900 supports one clock output capability. The OCLKA Clock Configuration command enables the clock output functionality and sets the clock frequency.

Command

I²C		USB
Read	Write	0x0807
0x48	0xC8	

Table 39. Clock Configuration Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	0	Clock Selection 0 = OCLKA 1 = Reserved	d0	wr
	7:1	Reserved	d0	r
1	0	Clock Functionality Disable 0 = Disable clock functionality on selected pin 1 = Enable clock functionality on selected pin	d0	wr
	7:1	Reserved	d0	r
2	7:0	Clock Divider. Allowed values in the range of 2 to 127. Output frequency = 100 MHz / (Clock Divider) 0x0 = Reserved 0x1 = Reserved 0x2 = 2 ... 0x7F = 127 0xFF:0x80 = Reserved	x7F	wr

3.7 Pulse Width Modulated (PWM) Control

DLPC900 provides four general-purpose PWM channels that can be used for a variety of control applications, such as fan speed. If the PWM functionality is not needed, these signals can be programmed as GPIO pins. To enable the PWM signals:

1. Program the PWM signal using the PWM Setup command.
2. Enable the PWM signal with the PWM Enable command.

3.7.1 PWM Setup

The PWM Setup command sets the clock period and duty cycle of the specified PWM channel. The PWM frequency and duty cycle is derived from an internal 18.67-MHz clock. To calculate the desired PWM period, divide the desired clock frequency from the internal 18.67Mhz clock. For example, a PWM frequency of 2 kHz, requires a 18666667 / 2000 = 9333 or 0x2475.

Command

I²C		USB
Read	Write	0x1A11
0x41	0xC1	

Table 40. PWM Setup Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	1:0	PWM Channel Output Select 0 - PWM channel 0 (GPIO_PWM_0) 1 - PWM channel 1 (GPIO_PWM_1) 2 - PWM channel 2 (GPIO_PWM_2) 3 - PWM channel 3 (GPIO_PWM_3)	d0	wr
	7:2	Reserved		
4:1	31:0	Clock Period in increments of 53.57 ns. Clock Period = (value + 1) × 53.5 ns	d0	wr
5	6:0	Duty Cycle = (value + 1)% Value range is 1% to 99%	d0	wr
	7	Reserved	d0	r

3.7.2 PWM Enable

After the PWM Setup command configures the clock period and duty cycle, the PWM Enable command activates the PWM signals.

Command

I^2C		USB
Read	Write	0x1A10
0x40	0xC0	

Table 41. PWM Enable Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	1:0	PWM Channel Output Select 0 - PWM channel 0 (GPIO_PWM_0) 1 - PWM channel 1 (GPIO_PWM_1) 2 - PWM channel 2 (GPIO_PWM_2) 3 - PWM channel 3 (GPIO_PWM_3)	d0	wr
	6:2	Reserved	d0	r
	7	PWM Channel Enable 0 -Disable selected PWM Channel 1 - Enable selected PWM Channel	d0	wr

3.8 Batch File Commands

During power-up and initialization or during normal operating, the DLPC900 can be commanded to execute a batch file containing a set of commands. The set of commands are created and saved in a text file. The text file then becomes an additional part of the firmware and updated into the flash memory. The user can also specify a default batch file that the DLPC900 will execute during its power-up sequence.

3.8.1 Batch File Name

This is a read command that returns the name of the given batch file index. This is useful for listing the set of batch files available for the user to execute. To list all the batch file names, the user should iterate through all the numbers from 0 to n until it returns an error, which identifies the end of the list.

Command

I²C	USB
Read	0x1A14
0x3A	

Table 42. Batch Name Command Definition

BYTES	BITS	DESCRIPTION	RESET	TYPE
0	7:0	Batch Command Index (Read parameter)	d0	w
15:1	All	Batch Command Name String (Read result)	d0	r

3.8.2 Batch File Execute

This command executes all the commands in the given batch file at the given index.

Command

I²C	USB
Write	0x1A15
0xBB	

Table 43. Batch File Execute Command Definition

BYTES	BITS	DESCRIPTION	RESET	TYPE
0	7:0	Batch command index to be executed.	d0	w

3.8.3 Batch File Delay

This command is useful for introducing the given amount of delay between batch commands within the same batch file. This command by itself does not perform any action.

Command

I²C	USB
Write	0x1A16
0xBC	

Table 44. Batch File Delay Command Definition

BYTES	BITS	DESCRIPTION	RESET	TYPE
3:0	31:0	Delay to be introduced in milliseconds	d0	w

3.8.4 Batch File Example

The following table shows an example of a batch file. Only Command Descriptors with parameters are allowed in the batch file.

Table 45. Batch File Example

Command Descriptor	Parameters	Description
VIDEO_CONT_SEL	0x01	Power on the IT6535 for HDMI input.
DELAY	0xC8	Delay 200 ms.
CHANNEL_SWAP	0x04	Select input data channel swap to ABC = BAC
FLIP_LONG	0x01	Flip the image on long axis.

When saving the batch file to a text file, only save the Command Descriptor and the Parameters as shown below with a colon after the Command Descriptor and space delimited. See Appendix B for a list of the supported Command Descriptors. Once the batch file has been created and saved as a text file, see the DLP LightCrafter 6500 & 9000 EVM Users Guide on how to add batch files to the firmware.

VIDEO_CONT_SEL:	0x01
DELAY:	0xC8
CHANNEL_SWAP:	0x04
FLIP_LONG:	0x01

4 Display Mode Commands

The DLPC900 display consists of several parameters which dictate the loading of the DMD and the control of PWM to the LEDs. The DLPC900 supports four main display modes:

- Normal Video mode
- Video Pattern mode
- Pre-Stored Pattern mode
- Pattern On-The-Fly mode

The Display Mode Selection command (Section 4.1) selects between these modes.

In Normal Video mode, the DLPC900 30-bit RGB interface supports up to 1080p @120Hz with a DLP6500 and WQXGA @60Hz with a DLP9000. The DLPC900 processes the digital input image and converts the data into the appropriate format for the DLP6500 or the DLP9000 DMDs. The DLPC900 offers scaling and cropping functions to appropriately display resolutions from SVGA to 1080p on a DLP6500. The DLPC900 combined with a DLP9000 does not support scaling or cropping functions.

In the latter three modes, the DLPC900 provides high-speed pattern rates. These modes support only 24-bit data input through the DLPC900 RGB interface, from flash memory, or dynamically loaded using Pattern On-The-Fly modes. These modes are well-suited for techniques such as structured light, additive manufacturing, or digital exposure. The DLPC900 also has the capability to display a set of patterns and signal a camera to capture when these patterns are displayed. Figure 8 shows the DLPC900/DLP6500 block diagram and the main functional blocks for the four display modes. Table 46 lists the allowed pattern combinations of bit-depth, number of patterns, and maximum pattern speed.

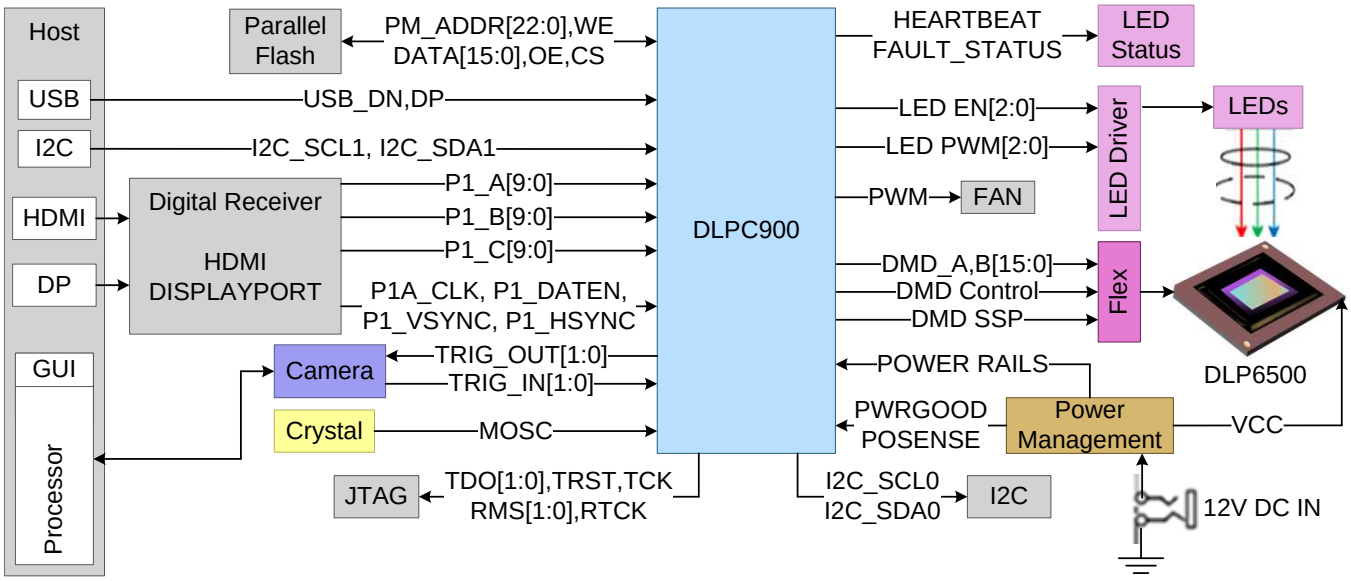

Figure 8. DLPC900 System Block Diagram

Table 46. Allowed Pattern Display Combinations

BIT-DEPTH	MAXIMUM EXTERNAL INPUT PATTERN RATE (Hz)	MAXIMUM PATTERN RATE FOR PRE-LOADED PATTERNS WITH INTERNAL TRIGGER (Hz)	MAXIMUM PATTERN RATE FOR PRE-LOADED PATTERNS WITH EXTERNAL TRIGGER (Hz)(1)	MAXIMUM NUMBER OF PATTERNS FOR PRE-LOADED PATTERNS (Hz)(2)
1	2880	9523	8333	400
2	1440	3289	3125	200
3	960	2544	2380	133
4	720	1215	1190	100
5	480	823	813	80
6	480	672	664	66
7	360	500	496	57
8	247	247	246	50

(1) The reduction in pattern rates are due to interrupt processing and sequence setup time from the time the external interrupt occurs.
(2) Numbers are based on uncompressed patterns.

In Normal Video mode, the DLPC900 operates on a per-frame basis where it takes the input data and appropriately allocates it in a frame. For example, a 24-bit RGB input image is allocated into a 60-Hz frame by dividing each color (red, green, and blue) into specific percentages of the frame. Therefore, for a 40% red, 45% green, and 15% blue ratio; the red, green, and blue colors would have a 6.67-, 7.5-, and 2.54-ms time slot allocated, respectively. Because each color has an 8-bit depth, each color time slot is further divided into bit-planes, as shown in Figure 9. A bit-plane is the two-dimensional arrangement of one bit extracted from all the pixels in the full color 2D image.

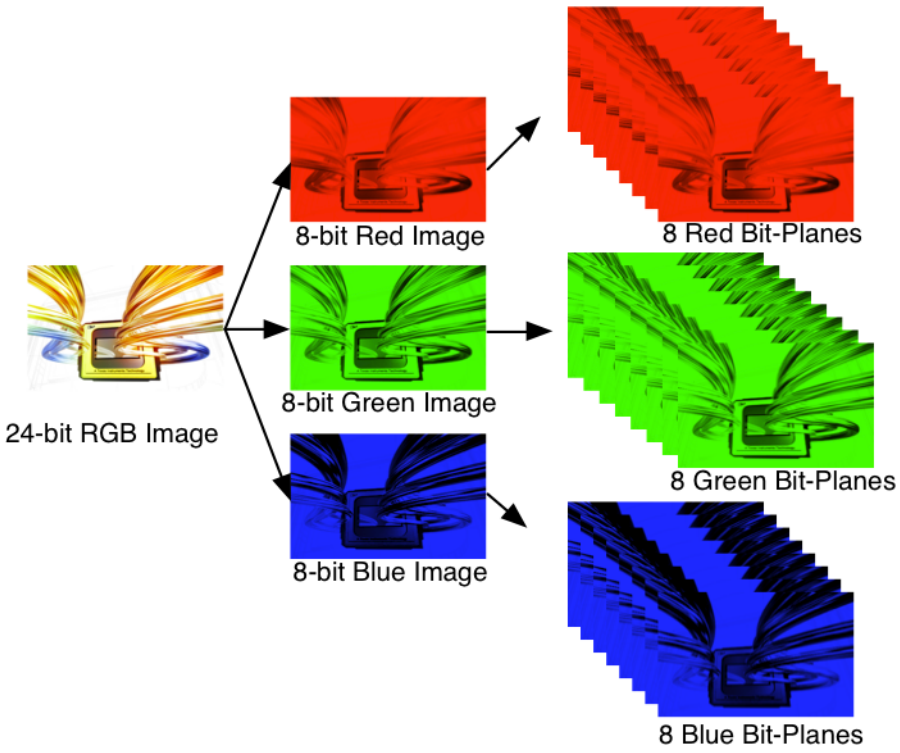

Figure 9. Bit-Planes of a 24-Bit RGB Image

The length of each bit-plane in the time slot is weighted by the corresponding power of two of its binary representation. This provides a binary pulse-width modulation of the image. For example, a 24-bit RGB input has three colors with 8-bit depth each. Each color time slot is divided into eight bit-planes, with the sum of the weight of all bit planes in the time slot equal to 256. See Figure 10 for an illustration of this partition of the bits in a frame.

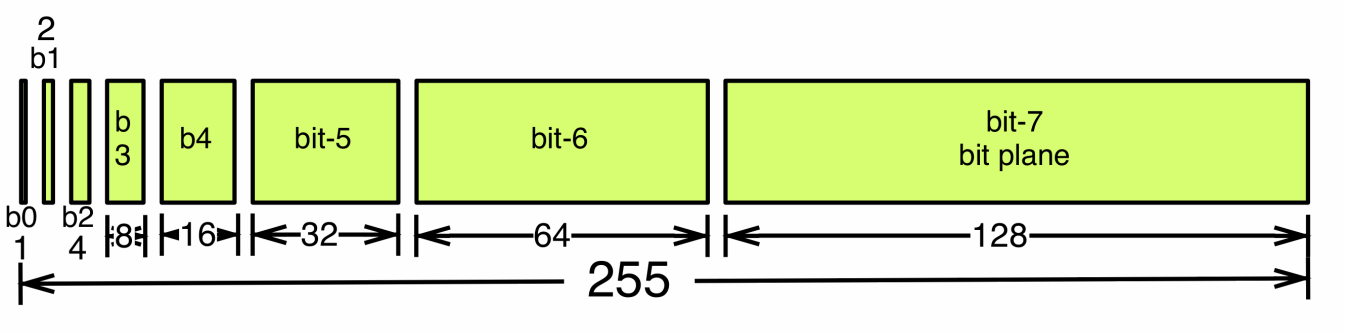

Figure 10. Bit Partition in a Frame for an 8-Bit Monochrome Image

Therefore, a single video frame is composed of a series of bit-planes. Because the DMD mirrors can be either on or off, an image is created by turning on the mirrors corresponding to the bit set in a bit-plane. With binary pulsewidth modulation, the intensity level of the color is reproduced by controlling the amount of time the mirror is on. For a 24-bit RGB frame image inputted to the DLPC900 controller, the DLPC900 controller creates 24 bit-planes, stores them in internal embedded DRAM, and sends them to the DMD, one bitplane at a time. The bit weight controls the illumination intensity of the bit-plane where smaller the bit weight is the less intense the bit-plane becomes. To improve image quality in video frames, these bit-planes, time slots, and color frames are shuffled and interleaved within the pixel processing functions of the DLPC900 controller.

For other applications where one-to-one pixel mapping to DMD micromirror is required, the scaling, cropping, and pixel processing functions are disabled and specific set of patterns are used. The bit-depth of the pattern is then allocated into the corresponding binary weighted time slots. Furthermore, output trigger signals are also synchronized with these time slots to indicate when the image is displayed. For structured light applications, this mechanism provides the capability to display a set of patterns and signal a camera to capture these patterns overlaid on an object.

4.1 Display Mode Selection

The Display Mode Selection command switches the internal image processing functions of the DLPC900 to operate in the mode selected. After executing this command, the host may poll the system status using I^2C commands: 0x20, 0x21, and 0x22 or the respective USB commands: 0x1A0A, 0x1A0B, and 0x1A0C.

Command

I^2C		USB
Read	Write	0x1A1B
0x69	0xE9	

Table 47. Display Mode Selection Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	1:0	0 = Normal video mode 1 = Pre-stored pattern mode (Images from flash) 2 = Video pattern mode 3 = Pattern On-The-Fly mode (Images loaded through USB/I^2C)	d1	wr

4.1.1 Video Mode Resolution

When Display Mode is set to Normal Video Mode, Table 48 shows the resolutions supported by the DLPC900 when combined with a DLP6500 or a DLP9000.

Table 48. Resolution Supported

DLP Lighcrafter	Resolution	Max Rate (Hz)	Notes
6500	Standard Video Sources from SVGA to 1080p	120	See DLPC900 data sheet for reduced blanking requirements for 1080p @120Hz
9000	WQXGA	120	See DLPC900 data sheet under Two Controller Considerations.

4.1.2 Input Display Resolution

The Input Display Resolution command defines the active input resolution and active output (displayed) resolution. This command provides the option to define a subset of active input frame data using pixel (column) and line (row) counts relative to the source-data enable signal (DATEN). In other words, this feature allows the source image to be cropped as the first step in the processing chain. After executing this command, the host may poll the system status using I^2C commands: 0x20, 0x21, and 0x22, or the respective USB commands: 0x1A0A, 0x1A0B, and 0x1A0C.**This command is not supported when DLPC900 is combined with a DLP9000 DMD**.

Command

I^2C		USB
Read	Write	0x1000
0x7E	0xFE	

Table 49. Input Display Resolution Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
1:0	15:0	Input image, first active pixel (column) of cropped area	d0	wr
3:2	15:0	Input image, first active line (row) of cropped area	d0	
5:4	15:0	Input image vertical resolution, pixels (columns) per line (row) of cropped area	d0	
7:6	15:0	Input image horizontal resolution, lines (rows) per frame of cropped area	d0	
9:8	15:0	Output image, first active pixel (column) of displayed image	d0	
11:10	15:0	Output image, first active line (row) of displayed image	d0	
13:12	15:0	Output image horizontal resolution, pixels (columns) per line (row)	d(1)	
15:14	15:0	Output image vertical resolution, lines (rows) per frame	d(2)	

(1) Maximum 1920 or 1280.
(2) Maximum 1080 or 1600.

4.2 Pattern Display Commands

In pattern display modes 0,2, and 3, the DLPC900 supports 1-, 2-, 3-, 4-, 5-, 6-, 7-, and 8-bit images with a 1080p or WQXGA pixel resolution streamed through the 24-bit RGB parallel interface, pre-stored patterns in the flash memory, or dynamically with Pattern On-The-Fly. The following commands are only supported in display modes 0, 2, and 3:

- Trigger Commands
- LED Enable Delay Commands
- Pattern Display Commands
- Pattern On-The-Fly Commands

NOTE: If the pattern display is already active, it must be stopped using I^2C command 0x65 or USB 0x1A24 before calling these commands.

4.2.1 Trigger Commands

To synchronize a camera with the displayed patterns, the DLPC900 supports three pattern modes:

- Video Pattern Mode (applicable when pattern data from RGB parallel port):
  - VSYNC used as trigger input.
  - TRIG_OUT1 frames the exposure time of the pattern.
  - TRIG_OUT2 : marks the beginning of each pattern start with 20us pulse. This can be selectively disabled for individual patterns.
- Pre-Stored Pattern Mode (applicable for pattern data from flash):
  - TRIG_IN1 advances to next pattern, while TRIG_IN2 starts and pauses the pattern sequence.
  - TRIG_OUT1 frames the exposure time of the pattern.
  - TRIG_OUT2 : marks the beginning of each pattern start with 20us pulse. This can be selectively disabled for individual patterns.
- Pattern On-The-Fly Mode (patterns downloaded over USB/I2C)
  - Triggers are the same as Pre-Stored Pattern Mode

Figure 11 shows an example of a Video Pattern Mode, where the VSYNC starts the pattern sequence display. Frame time indicates the time between VSYNC triggers, and display time indicates the length of pattern sequence. This display time should be less than the frame time. The pattern sequence consists of a series of three consecutive patterns. The first pattern sequence consists of P1, P2, and P3. Since P3 is an RGB pattern, it is shown with its time sequential representation of P3.1, P3.2, and P3.3. The second pattern sequence consists of three patterns: P4, P5, and P6. The third sequence consists of P7, P8, and P9. TRIG_OUT_1 frames each pattern exposed, while TRIG_OUT_2 indicates the start of each of the three pattern sequences.

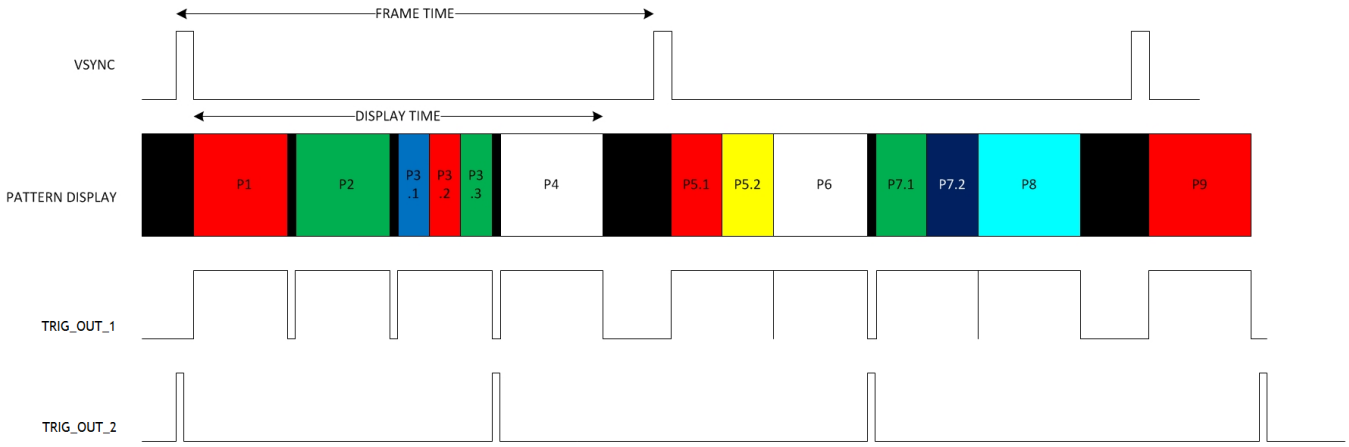

Figure 11. Video Pattern Mode Timing Diagram Example

Figure 12 shows an example of a Pre-Stored Pattern Mode with internal trigger to advance to next pattern. A set of three-pattern sequences are displayed. TRIG_OUT_1 frames each pattern exposed, while TRIG_OUT_2 indicates the start of each three-pattern sequence. TRIG_IN_2 serves as a start/stop signal. By raising TRIG_IN_2, the pattern sequence starts. By lowering TRIG_IN_2, the pattern sequence stops. If the pattern sequence had been previously started, raising TRIG_IN_2 continues the pattern sequence until this signal is lowered. If TRIG_IN_2 is lowered while a pattern is displayed (see P4 in Figure 12), when this pattern sequence is continued, this pattern is displayed again since its full exposure was not completed.

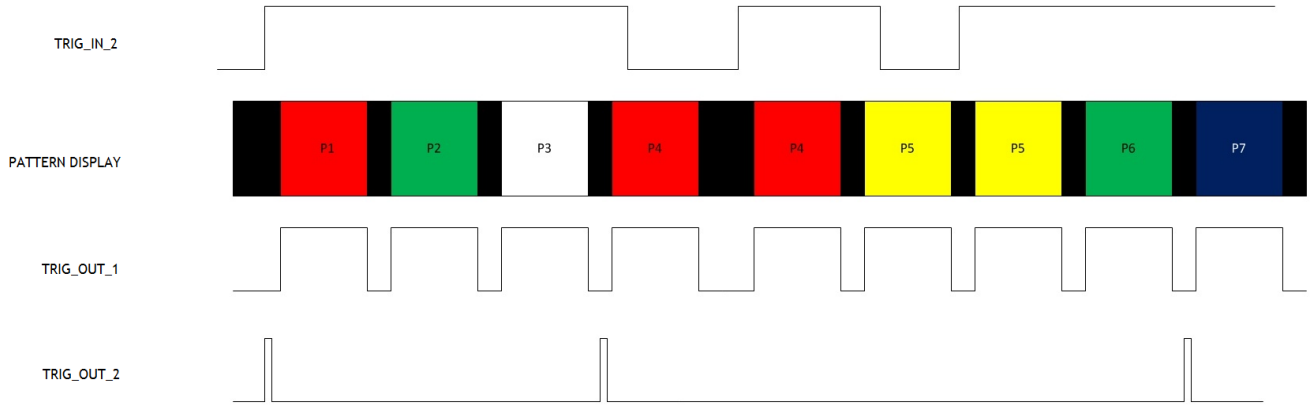

Figure 12. Pre-Stored Pattern Mode Timing Diagram Example

4.2.1.1 Trigger Out1

The Trigger Out1 command sets the polarity, rising edge delay, and falling edge delay of TRIG_OUT_1 signal. The delays are compared to when the pattern is displayed on the DMD. Before executing this command, stop the current pattern sequence.

Command

I²C		USB
Read	Write	0x1A1D
0x6A	0xEA	

Table 50. Trigger Out1 Command Definition

BYTE	BITS	DESCRIPTION(1)	RESET	TYPE
0	0	0 = Non inverted trigger output (2)	d0	wr
		1 = Inverted trigger output (3)		
	7:1	Reserved	d0	r
2:1	16:0	Trigger output Raising Edge delay in micro seconds (int16 number) Valid Range : -100 to 20000	d0	wr
4:3	16:0	Trigger output Falling Edge delay in micro seconds (int16 number) Valid Range : -100 to 20000	d0	wr

(1) Minimum pulse width is 20us.
(2) When non inverted output is selected, the rising edge must be less than or equal to the falling edge.
(3) When inverted output is selected, the rising edge must be greater than or equal to the falling edge.

4.2.1.2 Trigger Out2

The Trigger Out2 Control command sets the polarity and rising edge delay of TRIG_OUT_2 signal. The delay is compared to when the pattern is displayed on the DMD. Before executing this command, stop the current pattern sequence.

Command

I²C		USB
Read	Write	0x1A1E
0x6B	0xEB	

Table 51. Trigger Out2 Command Definition

BYTE	BITS	DESCRIPTION[1]	RESET	TYPE
0	0	0 = Non inverted trigger output [2]	d0	wr
		1 = Inverted trigger output [3]		
	7:1	Reserved	d0	r
2:1	16:0	Trigger output Raising Edge delay in micro seconds (int16 number) Valid Range : -100 to 20000	d0	wr
4:3	16:0	Trigger output Falling Edge delay in micro seconds (int16 number) Valid Range : -100 to 20000	d0	wr

(1) Minimum pulse width is 20us.
(2) When non inverted output is selected, the rising edge must be less than the falling edge.
(3) When inverted output is selected, the rising edge must be greater than the falling edge.

4.2.1.3 Trigger In1

The Trigger In1 command sets the rising edge delay of TRIG_IN_1 signal compared to when the pattern is displayed on the DMD. The polarity of TRIG_IN_1 is set in the lookup table of the pattern sequence. Before executing this command, stop the current pattern sequence.

Command

I²C		USB
Read	Write	0x1A35
0x79	0xF9	

Table 52. Trigger In1 Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
1:0	15:0	Trigger 1 delay in micro seconds. This is the time after which the pattern is displayed from trigger active edge. A minimum delay of 104 micro second is required by the HW	d105	wr
2	0	0 – Pattern advances on rising edge 1 – Pattern advances on falling edge	d0	wr
	7:1	Reserved	d0	r

4.2.1.4 Trigger In2

In Video Pattern and Pre-Stored Pattern modes, TRIG_IN_2 acts as a start or stop signal. If the sequence was not already started already by a software command, the rising edge on TRIG_IN_2 signal input will start or resume the pattern sequence. If the pattern sequence is active, the falling edge on TRIG_IN_2 signal input stops the pattern sequence. Before executing this command, stop the current pattern sequence.

Command

I²C		USB
Read	Write	0x1A36
0x7A	0xFA	

Table 53. Trigger In2 Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	0	0 – Pattern started on rising edge stopped on falling edge 1 – Pattern started on falling edge stopped on rising edge	d0	wr
	7:1	Reserved	d0	r

4.2.2 LED Enable Delay Commands

The LED Enable Delay commands set the rising and falling edge offsets of the LED enable signals compared to when the pattern is displayed on the DMD. This command is only for Pattern Display mode; When in a video mode, these delays should be set to 0x0.

4.2.2.1 Red LED Enable

The Red LED Enable Delay command sets the rising and falling edge delay of the Red LED enable signal.

Command

I²C		USB
Read	Write	0x1A1F
0x6C	0xEC	

Table 54. Red LED Enable Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
1:0	16:0	LED Enable Raising Edge delay in micro seconds (int16 number) Valid Range : -100 to 20000	d0	wr
3:2	16:0	LED Enable Falling Edge delay in micro seconds (int16 number) Valid Range : -100 to 20000	d0	wr

4.2.2.2 Green LED Enable

The Green LED Enable Delay command sets the rising and falling edge delay of the Green LED enable signal.

Command

I²C		USB
Read	Write	0x1A20
0x6D	0xED	

Table 55. Green LED Enable Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
1:0	16:0	LED Enable Raising Edge delay in micro seconds (int16 number) Valid Range : -100 to 20000	d0	wr
3:2	16:0	LED Enable Falling Edge delay in micro seconds (int16 number) Valid Range : -100 to 20000	d0	wr

4.2.2.3 Blue LED Enable

The Blue LED Enable Delay command sets the rising and falling edge delay of the Blue LED enable signal.

Command

I²C		USB
Read	Write	0x1A21
0x6E	0xEE	

Table 56. Blue LED Enable Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
1:0	16:0	LED Enable Raising Edge delay in micro seconds (int16 number) Valid Range : -100 to 20000	d0	wr
3:2	16:0	LED Enable Falling Edge delay in micro seconds (int16 number) Valid Range : -100 to 20000	d0	wr

4.2.3 Pattern Display Commands

4.2.3.1 Pattern Display Start/Stop

The Pattern Display Start/Stop command starts or stops the programmed patterns sequence. After executing this command, the host may poll the system status using I^2C commands: 0x20, 0x21, and 0x22 or the respective USB commands: 0x1A0A, 0x1A0B, and 0x1A0C.

Command

I^2C		USB
Read	Write	0x1A24
0x65	0xE5	

Table 57. Pattern Display Start/Stop Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	1:0	0 = Stop Pattern Display Sequence. The next "Start" command will restart the pattern sequence from the beginning. 1 = Pause Pattern Display Sequence. The next "Start" command will start the pattern sequence by re-displaying the current pattern in the sequence. 2 = Start Pattern Display Sequence	d0	wr
	7:2	Reserved	d0	r

4.2.3.2 Pattern Display Invert Data

The Pattern Display Invert Data command dictates how the DLPC900 interprets a value of 0 or 1 to control mirror position for displayed patterns. Before executing this command, stop the current pattern sequence. Once the command has been sent to the DLPC900, the Pattern Display LUT Definition for all the patterns must be re-sent to the DLPC900.

Command

I^2C		USB
Read	Write	0x1A30
0x74	0xF4	

Table 58. Pattern Display Invert Data Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	0	Pattern Display Invert Data 0 = Normal operation. A data value of 1 will flip the mirror to output light, while a data value of 0 will flip the mirror to block light 1 = Inverted operation. A data value of 0 will flip the mirror to output light, while a data value of 1 will flip the mirror to block light	d0	wr
	7:1	Reserved	d0	r

4.2.3.3 Pattern Display LUT Configuration

The Pattern Display LUT Configuration command controls the execution of patterns stored in the lookup table (LUT). Before executing this command, stop the current pattern sequence.

Command

I²C		USB
Read	Write	0x1A31
0x75	0xF5	

Table 59. Pattern Display LUT Configuration Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
1:0	10:0	Number of LUT entries (range 0 through 511) 0 = Zero entries 1 = One entries ... 512 = 512 entries	d0	wr
	15:11	Reserved		
5:2	31:0	Number of times to repeat the pattern sequence	d0	wr

4.2.3.4 Pattern Display LUT Definition

The Pattern Display LUT Definition contains the definition of each pattern to be displayed during the pattern sequence. **Display Mode and Pattern Display LUT Configuration must be set before sending any pattern LUT definition data. If the Pattern Display Data Input Source is set to streaming, the image indexes do not need to be set.** Regardless of the input source, the pattern definition must be set.

Command

I²C	USB
Write	0x1A34
0xF8	

Table 60. Pattern Display LUT Definition Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
1:0	15:0	Pattern Index (range 0 through 511)	d0	w
4:2	23:0	Pattern exposure in micro seconds		
5	0	Clear the pattern after exposure. This is only applicable for 1 bit pattern with external trigger. For other patterns, the clear is automatically handled.		w
	3:1	Select desired bit-depth b000 = 1 bit b001 = 2 bit b010 = 3 bit ... b111 = 8 bit		w
	6:4	b000 = All LEDs disabled b001 = Red b010 = Green b011 = Yellow (Green + Red) b100 = Blue b101 = Magenta (Blue + Red) b110 = Cyan (Blue + Green) b111 = White (Blue + Green + Red)		w
	7	1 = Wait for trigger before displaying the pattern 0 = Continue running after previous pattern		w
8:6	23:0	Dark display time following the exposure (in micro seconds)		w
9	0	1 = Disable trigger 2 output for this pattern 0 = Enable trigger 2 output for this pattern		w
	7:1	Reserved		w
11:10	10:0	Image pattern index (Not applicable in video pattern mode) Valid Range 0-255		w
	15:11	Bit position in the image pattern (Frame in video pattern mode) Valid range 0-23		w

4.2.4 Pattern On-The-Fly Commands

These commands allow the user to dynamically upload the pattern images over the I^2C or USB interface and store them directly into internal memory. The user can preview the pattern sequence to verify that the patterns and the pattern sequence are correct before actually writing the patterns to the flash. These commands should be used only in Pattern On-The-Fly mode and requires **Display Mode and Pattern Display LUT Configuration must be set prior to sending any pattern LUT data using the Pattern Display LUT Definition command**. Section Section 2 shows an example of Images On-The-Fly.

4.2.4.1 Initialize Pattern BMP Load

When Initialize Pattern BMP Load command is issued, the patterns in the flash are not used until the pattern mode is disabled by command. This command should be followed by the Pattern BMP Load command to load the images. The images should be loaded in the reverse order. Suppose there are 3 images 0,1 and 2 then the order for loading the image is 2, 1 and 0. **When the DLPC900 is combined with a DLP9000 DMD, the user must perform the same operation on both the master and slave controllers by choosing the appropriate command in the command table.**

Command

Controller	I^2C		USB
	Read	Write	
Master	0x2A	0xAA	0x1A2A
Slave	0x2C	0xAC	0x1A2C

Table 61. Initialize Pattern BMP Load Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
1:0	4:0	Image Index (0 – 17). In 24 bit format. This be referred in LUT command. Always load the images in reverse order.	d0	wr
5:2	31:0	Number of bytes in the compressed image	d0	wr

4.2.4.2 Pattern BMP Load

This command is used for updating the pattern images on-the-fly. This loads the full compressed 24bit BMP images into the internal memory of the DLPC900. This command is issued after the Init pattern BMP command and multiple times until all the bytes are sent. Images should be compressed using Run-Length Encoding (RLE) See Appendix C for a description of the compression formats. **When the DLPC900 is combined with a DLP9000 DMD, the user must load the images to both the master and slave controllers by choosing the appropriate command in the command table. The full WQXGA image must be divided in half where the master controller gets the left half and the slave controller gets the right half.**

NOTE: The images must be re-downloaded to the DLPC900 whenever changes are made to the number of entries in the Pattern Display LUT Configuration or changing the images, bit depth, image index, or bit position in the Pattern Display LUT Definition

Command

Controller	I²C		USB
	Read	Write	
Master	0x2B	0xAB	0x1A2B
Slave	0x2D	0xAD	0x1A2D

Table 62. Pattern BMP Load Command

BYTES	BITS	DESCRIPTION	RESET	TYPE
1:0	9:0	Number of bytes in this packet	d0	w
n:2	All	Compressed BMP Data	d0	w

4.2.5 I²C Pass Through Commands

The I²C Pass Through commands allows the user to use the controller's I²C port 1 or port 2 to control external devices.

4.2.5.1 I²C Pass Through Configuration

The I²C Pass Through Configuration command configures the I²C port to be used.

Command

I²C	USB
Write	0x1A4E
0xC5	

Table 63. I²C Pass Through Configuration Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
0	1:0	I²C Port number 1or 2 0 - Invalid Port 1 - Port 1 2 - Port 2	d0	w
	3:2	Reserved		
	4	Device addressing mode 0 – 7 bit addressing 1 – 10 bit addressing	d0	w
	7:5	Reserved		
4:1	31:0	I²C Clock rate 100000 – 400000 Hz (Actual rate may not be exactly as entered due to the dividers used in calculating the rate)	d0	w

4.2.5.2 I²C Pass Through Write

The I²C Pass Through Write command allows the user to send data to the specified I²C device on the port that was configured by the Pass Through Configuration command.

Command

I²C	USB
Write	0x1A4F
0xCF	

Table 64. I²C Pass Through Write Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
1:0	15:0	Number of bytes to write (1 – 512)	d0	w
2	1:0	I²C Port number 1or 2. (Port configuration of the port being used must have been done prior to using this command) 0 - Invalid Port 1 - Port 1 2 - Port 2	d0	w
	7:2	Reserved		
4:3	10:0	Slave Address	d0	w
	15:11	Reserved		
n:5	All	Bytes to be written	d0	w

4.2.5.3 I²C Pass Through Read

The I²C Pass Through Read command allows the user to read data from the specified I²C device on the port that was configure by the Pass Through Configuration command.

Command

I²C	USB
Read	0x1A4F
0x4F	

Table 65. I²C Pass Through Read Command Definition

BYTE	BITS	DESCRIPTION	RESET	TYPE
1:0	15:0	Number of bytes to write (1-512)	d0	w
3:2	15:0	Number of bytes to read (1-512)	d0	w

Table 65. I²C Pass Through Read Command Definition (continued)

BYTE	BITS	DESCRIPTION	RESET	TYPE
4	1:0	I²C Port number 1or 2 (Port configuration of the port being used must have been done prior to using this command) 0 - Invalid Port 1 - Port 1 2 - Port 2	d0	w
	7:2	Reserved		
6:5	10:0	Slave Address	d0	w
	15:11	Reserved		
n:7	All	Data to be written	d0	w
m:0	All	Data bytes read	d0	r

Power-Up and Power-Down and Initialization Considerations

This chapter describes the initial power-up and power-down considerations, as well as other initialization considerations.

1 Power Up

The DLPC900 is initialized and ready to process commands some time after the signal $\overline{\text{RESET}}$ is driven high. Detailed power-up timing is given in the DLPC900 data sheet, DLPS037.

2 Power Down

No commands are required at power down of the DLPC900. The DC power supplies must be turned off, and PWRGOOD must be set low, according to the timing in the DLPC900 data sheet, DLPS037.

3 Power-Up Auto-Initialization

Upon release of system reset, the DLPC900 executes an auto-initialization routine that is automatically uploaded from flash. This initialization process consists of setting specific configurations, uploading specific configuration tables (such as sequence), and displaying a defined splash screen. The goal of the auto-initialization process is to allow the DLPC900 to fully configure itself for default operation with no external I^2C control.

Command Examples

1 Video Pattern Mode Example

The following table lists the step for a Video Pattern Mode example with two exposures. Start with the projector powered on and displaying a video source.

Table 66. Video Pattern Mode Example

Step	I²C[1]	USB[1]	Data[1]	Description
1	E9	1A1B	02	Set video pattern mode
2	F8	1A34	00 00 FA 00 00 90 00 00 00 00 00 00	Define pattern 0 (200 us red 1 bit) and wait for trigger
3	F8	1A34	01 00 90 01 00 21 00 00 00 00 00 08	Define pattern 1 (400 us green 2 bit)
4	F5	1A31	02 00 00 00 00 00	Number of patterns 2 with indefinite repeat
5	E5	1A24	02	Start running the pattern[2]

(1) All bytes are in HEX notation.
(2) A video source must be connect before performing this step.

2 Pre-Stored Pattern Mode Example

The following table lists the steps for a Pre-Stored Pattern Mode example with two exposures. Start with the projector powered on and displaying a video source.

Table 67. Pre-Stored Pattern Mode Example

Step	I²C[1]	USB[1]	Data[1]	Description
1	E9	1A1B	01	Set pre-stored pattern mode
2	F8	1A34	00 00 FA 00 00 10 00 00 00 00 00 00	Define pattern 0 (200 us red 1 bit)
3	F8	1A34	01 00 90 01 00 21 00 00 00 00 00 08	Define pattern 1 (400 us green 2 bit)
4	F5	1A31	02 00 00 00 00 00	Number of patterns 2 and indefinite repeat
5	E5	1A24	02	Start running the pattern[2]

(1) All bytes are in HEX notation.
(2) There must be at least two pattern images in flash memory.

3 Pattern On-The-Fly Example

The following table lists the steps for Image On-The-Fly Pattern Mode example with two images. Start with the projector powered on and displaying a video source. Images should be compressed using Run-Length Encoding (RLE).

Table 68. Pattern On-The-Fly Example

Step	I²C[1]	USB[1]	Data[1]	Description
1	E9	1A1B	03	Set on-the-fly pattern mode
2	F5	1A31	02 00 00 00 00 00	Number of patterns 2 and indefinite repeat
3	F8	1A34	00 00 FA 00 00 10 00 00 00 00 00 00	Define pattern 0 from image 0 (200 us red 1 bit)

(1) All bytes are in HEX notation.

Table 68. Pattern On-The-Fly Example (continued)

Step	I²C[1]	USB[1]	Data[1]	Description
4	F8	1A34	01 00 90 01 00 21 00 00 00 00 01 08	Define pattern 1 from image 1 (400 us green 2 bit)
5	AA	1A2A	01 00 E8 03 00 00	Set BMP 1 Size to 1000 (0x03E8)
6	AB	1A2B	00 02 XX XX …… XX	Load 512 byte of compressed BMP 1 Data
7	AB	1A2B	E8 01 XX XX ….. XX	Load 488 bytes of compressed BMP 1 Data
8	AA	1A2A	00 00 D0 07 00 00	Set BMP 0 Size to 2000 (0x07D0)
9	AB	1A2B	00 02 XX XX …… XX	Load 512 byte of compressed BMP 0 Data
10	AB	1A2B	00 02 XX XX …… XX	Load 512 byte of compressed BMP 0 Data
11	AB	1A2B	00 02 XX XX …… XX	Load 512 byte of compressed BMP 0 Data
12	AB	1A2B	00 01 XX XX …… XX	Load 464 bytes of compressed BMP 0 Data
13	E5	1A24	02	Start running the pattern

4 I²C Pass Through Write Example

The following table lists the steps to communicate with an external device using one of the DLPC900 I²C ports. The example shows how to write 16 bytes to an EEPROM starting at address location 16.

Table 69. I²C Pass Through Write Example

Step	I²C[1]	USB[1]	Data[1]	Description
1	C5	1A4E	01 A0 86 01 00	Address mode = 7-bits, port = 1, and clock = 100kHz
2	CF	1A4F	11 00 01 A0 00 00 10 01 18 01 03 A5 00 00 00 DA 04 85 A0 57 4A 9B 26	Number of bytes = 17, port = 1, device address = A0, EEPROM address location = 16, and 16 bytes of data.

(1) All bytes are in HEX notation.

5 I²C Pass Through Read Example

The following table lists the steps to communicate with an external device using one of the DLPC900 I²C ports. The example shows how to read 16 bytes from an EEPROM starting at address location 16.

Table 70. I²C Pass Through Read Example

Step	I²C[1]	USB[1]	Data[1]	Description
1	C5	1A4E	01 A0 86 01 00	Address mode = 7-bits, port = 1, and clock = 100kHz
2	4F	1A4F	01 00 10 00 01 A0 00 10	Number of bytes to write = 1, number of bytes to read = 16, port = 1, device address = A0, EEPROM address location = 16
3			01 18 01 03 A5 00 00 00 DA 04 85 A0 57 4A 9B 26	The host performs an I²C read operation to retrieve the data.

(1) All bytes are in HEX notation.

Register Quick Reference

This appendix provides a quick reference summary of all available sub-address commands.

A.1 I²C Register Quick Reference

Table 71. Register Quick Reference

I²C Sub-address		USB	Description	Type	Reset Value	Default Action
Read	Write					
0x00	0x80	0x1A00	Input Source Select	WR	0x8	24-bit parallel interface
0x02	0x82	0X1A02	Pixel Format	WR	0x0	RGB 4:4:4
0x03	0x83	0x1A03	Port and Clock Configuration	WR	0x0	Single Pixel, Pixel Clock 1, Data enable 1
0x04	0x84	0x1A37	Channel Swap	WR	0x8	ABC = BAC
0x06	0x86	0x1100	Curtain Color	WR	0x0 0x0 0x0 0x0 0x0 0x0	Curtain is black
0x07	0x87	0x0200	Power Mode	WR	0x0	Normal operation
0x08	0x88	0x1008	Long Axis Flip	WR	0x0	Flip disabled
0x09	0x89	0x1009	Short Axis Flip	WR	0x0	Flip disabled
0x0A	0x8A	0x1203	Test Pattern Select	WR	0x0	Solid Field
0x0B	0x8B	0x1A05	LED PWM Polarity	WR	0x0	Normal polarity
0x0C	0x8C	0x1A01	IT6535 Power Mode	WR	0x0	Power down.
0x10	0x90	0x1A07	LED Enable	WR	0x8	LEDs controlled by Sequencer
0x11		0x0205	Get Version	R	Will match firmware version	Will match firmware version.
0x1A	0x9A	0x1204	Test Pattern Color	WR	0x3FF 0x3FF 0x3FF 0x0 0x0 0x0	White foreground, black background
0x20		0x1A0A	Hardware Status	R	0x1	No errors
0x21		0x1A0B	System Status	R	0x1	No errors
0x22		0x1A0C	Main Status	R	0x0	No errors
0x32		0x0100	Read Error Code	R	0x0	No errors
0x33		0x0101	Read Error Code Description	R	0x0	No description
	0xAA	0x1A2A	Initialize Pattern BMP Load	W	0x0	See Command Description
	0xAB	0x1A2B	Pattern BMP Load	W	0x0	See Command Description
	0xAC	0x1A2C	Initialize Pattern BMP Load	W	0x0	See Command Description
	0xAD	0x1A2D	Pattern BMP Load	W	0x0	See Command Description
0x3A		0x1A14	Batch File Name	WR	0x0	Index

Table 71. Register Quick Reference (continued)

I²C Sub-address		USB	Description	Type	Reset Value	Default Action
	0xBB	0x1A15	Batch File Execute	W	0x0	Index
	0xBC	0x1A16	Batch File Delay	W	0x0	Delay
0x40	0xC0	0x1A10	PWM Enable	WR	Channel dependent	Channel dependent
0x41	0xC1	0x1A11	PWM Setup	WR	Channel dependent	Channel dependent
0x43	0xC3	0x1A12	PWM Capture	WR	Channel dependent	Channel dependent
0x44	0xC4	0x1A38	GPIO Configuration	WR	Channel dependent	Channel dependent
	0xC5	0x1A4E	I²C Pass Through Configuration	W	0x0 0x0 0x0 0x0 0x0	See Command Description
0x48	0xC8	0x0807	Clock Configuration	WR	Channel dependent	Channel dependent
0x4B	0xCB	0x0B01	LED Current	WR	0x97 0x78 0x7D	LED PWMs
0x4F	0xCF	0x1A4F	I²C Pass Through Read or Write	WR	See Command Description	See Command Description
0x65	0xE5	0x1A24	Pattern Start/Stop	WR	0x0	Pattern stopped
0x69	0xE9	0x1A1B	Display Mode	WR	0x0	Video Mode
0x6A	0xEA	0x1A1D	Trigger Out 1	WR	0x0 0x0 0x0 0x0 0x0	Normal Polarity with no rising or falling delay
0x6B	0xEB	0x1A1E	Trigger Out 2	WR	0x0 0x0 0x0 0x0 0x0	Normal Polarity with no rising delay
0x6C	0xEC	0x1A1F	Red Enable Delay	WR	0x0 0x0 0x0 0x0	No rising or falling delay
0x6D	0xED	0x1A20	Green Enable Delay	WR	0x0 0x0 0x0 0x0	No rising or falling delay
0x6E	0xEE	0x1A21	Blue Enable Delay	WR	0x0 0x0 0x0 0x0	No rising or falling delay
0x74	0xF4	0x1A30	Invert Data	WR	0x0	Normal operation
0x75	0xF5	0x1A31	Pattern LUT Configuration	WR	See Command Description	See Command Description
	0xF8	0x1A34	Pattern LUT Definition	W	See Command Description	See Command Description
0x79	0xF9	0x1A35	Trigger In 1	WR	0x69	No delay
0x7A	0xFA	0x1A36	Trigger In 2	WR	0x0	Advance Pattern Pair on Rising Edge (for Trigger Mode 2)
0x7E	0xFE	0x1000	Manual Input Display Resolution	WR	0x0	Output Display Resolution is DMD Dependant
0x7F	0xFF	0x1A39	Image Load	WR	0x0	Image Index

A.2 Command Guide

This section shows which commands can be used in which modes. I²C control and USB commands are accepted in any order, except when special sequencing is required (for example, setting up the flash).

Table 72. Command Matrix

Command Name	I²C		USB	Normal Power Mode	Standby Power Mode	Normal Video Mode	Video Pattern Mode	Pre-Stored Pattern Mode	Pattern On-The-Fly Mode
	Read	Write							
Input Source Select	0x00	0x80	0x1A00	x		x	x		

Table 72. Command Matrix (continued)

Command Name	I²C		USB	Normal Power Mode	Standby Power Mode	Normal Video Mode	Video Pattern Mode	Pre-Stored Pattern Mode	Pattern On-The-Fly Mode
Pixel Format	0x02	0x82	0x1A02	x		x	x		
Port and Clock Configuration	0x03	0x83	0x1A03	x	x	x	x		
Channel Swap	0x04	0x84	0x1A37	x	x	x	x		
Curtain Color	0x06	0x86	0x1100	x	x	x			
Power Mode	0x07	0x87	0x0200	x	x	x	x	x	x
Long Axis Flip	0x08	0x88	0x1008	x		x	x	x	x
Short Axis Flip	0x09	0x89	0x1009	x		x	x	x	x
Test Pattern Select	0x0A	0x8A	0x1203	x					
LED PWM Polarity	0x0B	0x8B	0x1A05	x	x	x	x	x	x
IT6535 Power Mode	0x0C	0x8C	0x1A01	x		x	x		
LED Enable	0x10	0x90	0x1A07	x		x	x	x	x
Get Version	0x11		0x0205	x	x	x	x	x	x
Test Pattern Color	0x1A	0x9A	0x1204	x	x				
Hardware Status	0x20		0x1A0A	x	x	x	x	x	x
System Status	0x21		0x1A0B	x	x	x	x	x	x
Main Status	0x22		0x1A0C	x	x	x	x	x	x
Read Error Code	0x32		0x0100	x	x	x	x	x	x
Read Error Code Description	0x33		0x0101	x	x	x	x	x	x
Initialize Pattern BMP Load		0xAA	0x1A2A	x					x
Pattern BMP Load		0xAB	0x1A2B	x					x
Initialize Pattern BMP Load		0xAC	0x1A2C	x					x
Pattern BMP Load		0xAD	0x1A2D	x					x
Batch File Name	0x3A		0x1A14	x	x	x	x	x	x
Batch File Execute		0xBB	0x1A15	x	x	x	x	x	x
Batch File Delay		0xBC	0x1A16	x	x	x	x	x	x
PWM Enable	0x40	0xC0	0x1A10	x	x	x	x	x	x
PWM Setup	0x41	0xC1	0x1A11	x	x	x	x	x	x
GPIO Configuration	0x44	0xC4	0x1A38	x	x	x	x	x	x

Table 72. Command Matrix (continued)

Command Name	I²C		USB	Normal Power Mode	Standby Power Mode	Normal Video Mode	Video Pattern Mode	Pre-Stored Pattern Mode	Pattern On-The-Fly Mode
I²C Pass Through Configuration		0xC5	0x1A4E	x	x	x	x	x	x
Clock Configuration	0x48	0xC8	0x0807	x	x	x	x	x	x
LED Current	0x4B	0xCB	0x0B01	x	x	x	x	x	x
I²C Pass Through Read or Write	0x4F	0xCF	0x1A4F	x	x	x	x	x	x
Pattern Start/Stop	0x65	0xE5	0x1A24	x			x	x	x
Display Mode	0x69	0xE9	0x1A1B	x		x	x	x	x
Trigger Out 1	0x6A	0xEA	0x1A1D	x	x		x	x	x
Trigger Out 2	0x6B	0xEB	0x1A1E	x	x		x	x	x
Red Enable Delay	0x6C	0xEC	0x1A1F	x	x		x	x	x
Green Enable Delay	0x6D	0xED	0x1A20	x	x		x	x	x
Blue Enable Delay	0x6E	0xEE	0x1A21	x	x		x	x	x
Invert Data	0x74	0xF4	0x1A30	x			x	x	x
Pattern LUT Configuration	0x75	0xF5	0x1A31	x			x	x	x
Pattern LUT Definition	0x78	0xF8	0x1A34	x			x	x	x
Trigger In 1	0x79	0xF9	0x1A35	x	x		x	x	x
Trigger In 2	0x7A	0xFA	0x1A36	x	x		x	x	x
Manual Input Display Resolution	0x7E	0xFE	0x1000	x		x			
Image Load	0x7F	0xFF	0x1A39	x		x			

Batch File Command Descriptors

This appendix provides a quick reference to all supported batch file command descriptors..

B.1 Command Descriptors

Command descriptors are followed by a colon. Each line in the batch file is space delaminated and saved as a text file.

Table 73. Command Descriptors

Command Descriptor	Description
SOURCE_SEL	Input Source Select
PIXEL_FORMAT	Pixel Format
CLK_SEL	Port and Clock Configuration
CHANNEL_SWAP	Channel Swap
POWER_CONTROL	Power Mode
FLIP_LONG	Long Axis Flip
FLIP_SHORT	Short Axis Flip
TPG_SEL	Test Pattern Select
PWM_INVERT	LED PWM Invert
LED_ENABLE	LED Enable
PWM_ENABLE	PWM Enable
PWM_SETUP	PWM Setup
GPIO_CONFIG	GPIO Configuration
LED_CURRENT	LED Current
DISP_CONFIG	Display Configuration
DISP_MODE	Display Mode
TRIG_OUT1_CTL	Trigger 1 Output Control
TRIG_OUT2_CTL	Trigger 2 Output Contorl
RED_LED_ENABLE_DLY	Red LED Enable Delay
GREEN_LED_ENABLE_DLY	Green LED Enable Delay
BLUE_LED_ENABLE_DLY	Blue LED Enable Delay
PAT_START_STOP	Pattern Start, Pause, and Stop
TRIG_IN1_CTL	Trigger Input 1 Control
TRIG_IN2_CTL	Trigger Input 2 Control
INVERT_DATA	Invert Data
PAT_CONFIG	Pattern LUT Configuration
MBOX_DATA	Pattern LUT Definition
SPLASH_LOAD	Image Load
GPCLK_CONFIG	Clock Output Configuration
TPG_COLOR	Test Pattern Color
I2C_PASSTHRU	I^2C Pass Through
VIDEO_CONT_SEL	IT6535 Power Mode

Table 73. Command Descriptors (continued)

Command Descriptor	Description
PATMEM_LOAD_INIT_MASTER	Initialize BMP Pattern On-The-Fly Master
PATMEM_LOAD_DATA_MASTER(1)	Load BMP Pattern On-The-Fly Master
PATMEM_LOAD_INIT_SLAVE	Initialize BMP Pattern On-The-Fly Slave
PATMEM_LOAD_DATA_SLAVE(1)	Load BMP Pattern On-The-Fly Slave
DELAY	Batch File Delay
I2C_CONFIG	I^2C Pass Through Configuration
CURTAIN_COLOR	Curtain Color
BATCHFILE_EXECUTE	Batch File Execute

(1) These commands are not allowed to be included in a batch file that is added to the firmware.

Pattern Image Compression

In order to minimize Flash storage requirements, it is recommended (but not required) that pattern images be stored in a compressed format. The compression format supported by the DLPC900 is a subset of BMP Run-Length Encoding (RLE). The DLPC900 is able to perform the decompression of pattern images as they are loaded from external flash or when using Pattern On-The-Fly mode to its internal memory. The DLPC900 can also perform no decompression if the images are not compressed.

For most efficient storage and compression of images, stored images should be packed into groups of 24-bit RGB bitmap images.

NOTE: Compressed images must be stored right side up instead of upside down as in standard BMP format images.

NOTE: With RLE, there is always a question of whether or not the compressed image will be larger or smaller than the uncompressed image. The method to decide which to choose from is left up to the programmer.

C.1 Image Header

The image data should be preceded by the image header (48 Bytes) shown in Table 74

Table 74. Image Header

Number of Bytes	Description
4	Signature (Should be 53 70 6C 64)
2	Image Width
2	Image Height
4	Number of bytes in the encoded image data
8	Reserved (Should be FF FF FF FF FF FF FF FF)
4	Background color (BB GG RR 00)
1	Reserved (Should be 00)
1	Compression 0 – Uncompressed 1 – RLE Compression 2 – Enhanced RLE compression
1	Reserved (Should be 01)
21	Reserved (Should be 00…)

C.2 Run-Length Encoding

Table 75 defines the RLE Control Bytes recognized by the DLPC900. The DLPC900 firmware automatically decompresses the image when operating in Pre-Stored Pattern Mode or Pattern On-The-Fly Mode.

Table 75. RLE Control Bytes

Control Byte (n)	Color Byte (c)	Result
0	0	End-of-Line
0	1	End-of-Image (required)
0	>= 2	Uncompressed. The next c pixels are uncompressed
n > 0	n/a	Repeat; Repeat the next RGB pixel (or the next dual y/c pixel pair) n times

C.2.1 RLE Compression Example

Table 76 shows the hexadecimal values of a 2-line packed 24-bit compressed bitmap. The compressed data on the left is stored sequentially in Flash memory. The DLPC900 firmware automatically expands the data as shown on the right which is store in internal memory.

Table 76. RLE Compression Example

Compressed Data (HEX)	Expanded Data (HEX)
03 040506	040506 040506 040506
05 777777	777777 777777 777777 777777 777777
00 03 040506 070809 0A0B0C	040506 070809 0A0B0C
02 789ABC	789ABC 789ABC
00 00	(End-of-Line Command)
00 00 00	00 00 00 00 00 00 00 00 (End-of-Line Padding)
07 1D1E1F	1D1E1F 1D1E1F 1D1E1F 1D1E1F 1D1E1F 1D1E1F 1D1E1F
06 212223	212223 212223 212223 212223 212223 212223
00 01	(End-of-File command)
00 01 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 (End-of-Image Padding)

C.3 Enhanced Run-Length Encoding

To achieve higher compression ratios, this compression format takes advantage of the similarities from line-to-line and uses one or two bytes to encode the length. Table 77 defines the RLE Control Bytes recognized by the DLPC900. The DLPC900 firmware automatically decompresses the image when operating in Pre-Stored Pattern Mode or Pattern On-The-Fly Mode.

Table 77. Enhanced RLE Control Bytes

Control Byte 1	Control Byte 2	Control Byte 3	Result
0	0	n/a	End of Image
0	1	*n*	Copy *n* pixels from previous line
0	*n* > 1	n/a	*n* uncompressed sequence of pixels
n > 1	n/a	n/a	Repeat following pixel *n* times

If *n* is < 128 then encode it with 1 byte.

If *n* is >= 128 then encode it with 2byte. The MS bit of first byte is 1 followed by MS bytes of *n* (7 bits only). The Next byte is the LS bytes of *n*.

C.3.1 Enhanced RLE Compression Example

Table 78 shows an example of this RLE compression. The number 0x1234 is encoded as 0x92, 0x34 (0x12 | 0x80 => 0x92).

Table 78. Enhanced RLE Compression Example

Compressed Data (Hex)	Expanded Data (Hex)
03 040506	040506 040506 040506
05 777777	777777 777777 777777 777777 777777
00 03 040506 070809 0A0B0C	040506 070809 0A0B0C
82 01 789ABC	789ABC 789ABC … (513 times)
00 00	(End of line)(1)
01 010203	010203
00 01 09	040506 040506 777777 777777 777777 777777 777777 040506 070809
00 01 00	(End-of-Image Padding)

(1) End-of-Line Command and End-of-Line Padding is optional for this RLE compression.

NOTE: All padding should end on a 4 byte boundary.

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, enhancements, improvements and other changes to its semiconductor products and services per JESD46, latest issue, and to discontinue any product or service per JESD48, latest issue. Buyers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All semiconductor products (also referred to herein as “components”) are sold subject to TI’s terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its components to the specifications applicable at the time of sale, in accordance with the warranty in TI’s terms and conditions of sale of semiconductor products. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by applicable law, testing of all parameters of each component is not necessarily performed.

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

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

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

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

Buyer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products, and any use of TI components in its applications, notwithstanding any applications-related information or support that may be provided by TI. Buyer represents and agrees that it has all the necessary expertise to create and implement safeguards which anticipate dangerous consequences of failures, monitor failures and their consequences, lessen the likelihood of failures that might cause harm and take appropriate remedial actions. Buyer will fully indemnify TI and its representatives against any damages arising out of the use of any TI components in safety-critical applications.

In some cases, TI components may be promoted specifically to facilitate safety-related applications. With such components, TI’s goal is to help enable customers to design and create their own end-product solutions that meet applicable functional safety standards and requirements. Nonetheless, such components are subject to these terms.

No TI components are authorized for use in FDA Class III (or similar life-critical medical equipment) unless authorized officers of the parties have executed a special agreement specifically governing such use.

Only those TI components which TI has specifically designated as military grade or “enhanced plastic” are designed and intended for use in military/aerospace applications or environments. Buyer acknowledges and agrees that any military or aerospace use of TI components which have ***not*** been so designated is solely at the Buyer's risk, and that Buyer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI has specifically designated certain components as meeting ISO/TS16949 requirements, mainly for automotive use. In any case of use of non-designated products, TI will not be responsible for any failure to meet ISO/TS16949.

Products		**Applications**	
Audio	www.ti.com/audio	Automotive and Transportation	www.ti.com/automotive
Amplifiers	amplifier.ti.com	Communications and Telecom	www.ti.com/communications
Data Converters	dataconverter.ti.com	Computers and Peripherals	www.ti.com/computers
DLP® Products	www.dlp.com	Consumer Electronics	www.ti.com/consumer-apps
DSP	dsp.ti.com	Energy and Lighting	www.ti.com/energy
Clocks and Timers	www.ti.com/clocks	Industrial	www.ti.com/industrial
Interface	interface.ti.com	Medical	www.ti.com/medical
Logic	logic.ti.com	Security	www.ti.com/security
Power Mgmt	power.ti.com	Space, Avionics and Defense	www.ti.com/space-avionics-defense
Microcontrollers	microcontroller.ti.com	Video and Imaging	www.ti.com/video
RFID	www.ti-rfid.com		
OMAP Applications Processors	www.ti.com/omap	**TI E2E Community**	e2e.ti.com
Wireless Connectivity	www.ti.com/wirelessconnectivity		

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