SiI9022A and SiI9024A Transmitter Programming Interface (TPI)

Programmer's Reference

Document # SiI-PR-1064-A

September 2011

Patents

The subject matter described herein may contain one or more inventions claimed in patents or patents pending owned by Silicon Image, Inc. or its affiliates.

Trademark Acknowledgment

Silicon Image®, the Silicon Image logo, SteelVine®, Simplay®, Simplay HD®, the Simplay HD logo, Satalink®, InstaPort®, the InstaPort Logo, and TMDS® are trademarks or registered trademarks of Silicon Image, Inc. in the United States or other countries. HDMI® and the HDMI logo with High-Definition Multimedia Interface are trademarks or registered trademarks of, and are used under license from, HDMI Licensing, LLC. in the United States or other countries. MHL® and the MHL Logo are trademarks or registered trademarks of, and are used under license from, MHL, LLC. in the United States or other countries. All other trademarks and registered trademarks are the property of their respective owners in the United States or other countries. The absence of a trademark symbol does not constitute a waiver of Silicon Image's trademark or other intellectual property rights with regard to a product name, logo or slogan.

Export Controlled Document

This document contains information subject to the Export Administration Regulations (EAR) and has a classification of EAR99 or is controlled for Anti-Terrorism (AT) purposes. Transfer of this information by any means to an EAR Country Group E:1 or foreign national thereof (whether in the U.S. or abroad) may require an export license or other approval from the U.S. Department of Commerce. For more information, contact the Silicon Image Director of Global Trade Compliance.

Further Information

To request other materials, documentation, and information, contact your local Silicon Image, Inc. sales office or visit the Silicon Image, Inc. web site at www.siliconimage.com.

Revision History

Revision	Date	Comment
A	9/28/2011	First release; content extracted from PR-1032.

Contents

List of Figures

List of Tables

Overview

Silicon Image HDMI transmitter solutions merge independent video and audio streams for transmission over HDMI. To do this, the transmitter employs a unique mechanism that allows host graphics and audio software to interact with the HDMI transmitter.

All new devices implement a simplified interface and special logic that automates most HDMI functions. This Transmitter Programming Interface (TPI) maps a concise set of registers into I^2C address space that the host can readily access.

TPI offers a significantly simplified operating scheme, using built-in hardware to handle tasks such as the following.

- Secure operation is fully automatic on devices equipped with HDCP. With only a single-bit write, the device establishes and maintains link security, interrupting the host only if the secure link is lost.
- DDC arbitration is handled cleanly, allowing the host to simply request the bus and then fetch EDID information directly – no need to program transfers into or out of a FIFO. Arbitration is handled even when link security is enabled.
- Complex audio setup is nearly eliminated. Hardware calculates N/CTS values and automatically sends out the appropriate packet information with no setup needed.
- All frequency-dependent internal settings, such as for PLLs, are derived automatically in hardware from the video mode information that has been programmed.

In addition:

- Devices still implement the standard Silicon Image register set used by legacy transmitters, for backwards compatibility with special functions in existing code.
- All devices implement an automated CEC controller, using the Silicon Image standard CEC Programming Interface (CPI).
- Many versions additionally support HDCP, implemented as part of the automated solution and requiring no host intervention for normal operation.

This document describes the TPI programming solution for the following devices.

HDCP-enabled Version	Non HDCP-enabled Version	Upgrade to this existing product	Application
SiI9024A Tx	SiI9022A Tx	SiI9022/24 Tx	HDMI Mobile

Figure 1 illustrates the major functional blocks for a typical TPI-based transmitter solution.

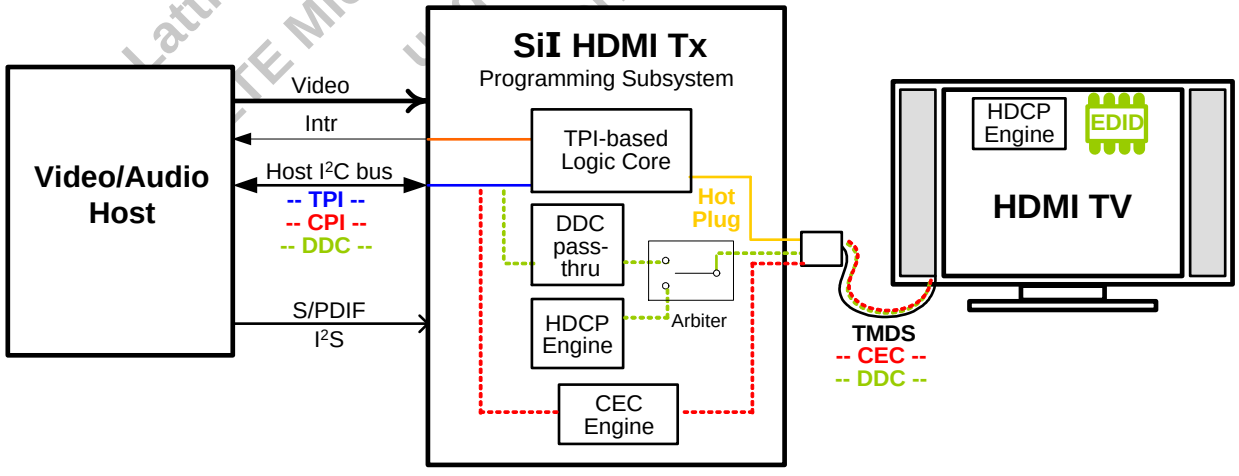

Figure 1. Transmitter Data Flow Paths

Register Group Summary

Transmitter Programming Interface (TPI)

TPI provides a programming interface that operates at a higher hardware level than traditional register file interfaces. The TPI register groups below handle all normal chip operations in a concise format.

Table 1. Register Group Summary

Group Function	Register Addresses	Page	Register Name	What Firmware Does with these Registers
Identification	0x1B–0x1D	9	Identification	Identifies the chip and version of TPI implemented.
Input Configuration – one-time configuration of the input bus for its application environment.	0x08	11	Input Bus and Pixel Repetition	Selects input bus characteristics like pixel size, clock edge.
	0x0B	15	YC Input Mode Select	Selects YC Mux modes, signal timing features, and chooses sync method registers to access.
	0x60–0x61	13	Sync Configuration and Monitoring	
	0x62–0x6D with 0x60[7] = 0	16	Explicit Sync DE Generation	Defines parameters for explicit sync method.
	0x62-6D with 0x60[7] = 1	18	Embedded Sync Extraction	Defines parameters for embedded sync method.
Audio Configuration	0x1F–0x28	23	Audio Configuration	Configures audio input channels and rates.
Video Mode Select – resolution, color space, InfoFrame headers	0x00–0x07	10	Video Mode	Defines the incoming resolution.
	0x09–0x0A	12	Input and Output Format	Defines color space, color depth.
	0x0C–0x19	19	AVI InfoFrame	Programs header information as defined by HDMI specification.
	0xBF–0xDE	20	Other InfoFrame	
	0x60	15	YC Mux Mode	
System Control – single-byte control for most-used functions	0x1A	22	System Control	Requests DDC bus access, selects between DVI/HDMI, controls TMDS output and AV Mute.
Interrupt – single-byte status for monitoring significant events	0x3C–0x3D	29	Interrupt	Polls for and clears events, selects the interrupt events that should cause hardware INT activation.
Power Control	1E	33	Power Control	Selects full-power operational mode or low-power standby mode.
HDCP – automatic security	0x29–0x3A	36	HDCP	Sets up and monitors HDCP link security.

Internal (Indexed) Registers

Underlying the TPI register set is a broader and more complex internal register set that is normally hidden from direct access. This register set includes the legacy registers that were available in older programming methodologies, but also some additional registers that are used primarily during chip development and testing.

From time to time, situations arise that require customer software to manually access these internal registers. Therefore, a mechanism is provided for accessing single bytes. Where defined, Internal Registers are accessed as noted below.

1. Set Page	2. Select Indexed Offset within Page	3. Obtain Read/Write Register Access
0xBC	0xBD	0xBE

Register Set	0x72:0xC7[7] = 1	0x72:0xC7[7] = 0
Page 0	I2C Slave Address 0x72	Indexed. TPI:0xBC = 0x01
Page 1	I2C Slave Address 0x7A	Indexed. TPI:0xBC = 0x02

Device-Specific Information

Several documents are required for a full understanding of the system level operation of the chip. Information in the Data Sheet and starter kit User's Guide is specific to the device in question; references are provided in the appropriate section below. CEC information is common to all devices and is provided in the following document.

- SiI-PR-0041: *CEC Programming Interface (CPI) Programmer's Reference*

SiI9022A and SiI9024A Transmitters

The SiI9022A transmitter and SiI9024A transmitter are pin- and feature-compatible replacements for the SiI9022 transmitter and SiI9024 transmitter, respectively. Refer to the *SiI9022A/9024A HDMI Transmitter Data Sheet* (SiI-DS-1053) for additional information.

Differences between SiI9022/SiI9024 and SiI9022A/SiI9024A Transmitters

Several new features have been added to the *A* version TPI register set, improving the ease of use.

- xvYCC is supported by sending gamut descriptor packets, described in the Other InfoFrame Data section on page 20.
- Top level TPI support is provided for TClkSel (TPI 0x08) and Audio Packet Layout Control (TPI 0x26).
- I^2S Audio Input Word Length selection is now written automatically (TPI 0x25).
- CEC no longer requires calibration (refer to the *CEC Programming Interface (CPI) Programmer's Reference* (SiI-PR-0041).

Firmware that runs on the SiI9022 or SiI9024 transmitter will run unchanged on the *A* version of these parts. However, certain changes should be considered for most efficient operation.

- The D3 state now has an additional requirement to program the Cold mode to take advantage of the new lower-power TMDS core.
- Wakeup from Receiver Sense (RSEN) is no longer available in D3 Cold mode but, is available in D3 Hot mode.
- The devices come up in operational state after a hardware reset, with the TMDS output enabled. To enter an intermediate power mode similar to the reset state of a non -*A* device, it is necessary to enter TPI mode (writing 00 to 0xC7).

Detection of HDCP capability on the *A* devices works slightly differently than on previous devices.

- On non -*A* devices, software could read the HDCP revision ID register TPI 0x30 to distinguish SiI9024 Tx (HDCP-equipped) from SiI9022 Tx (not HDCP-equipped). A zero value indicated no HDCP support. This method is no longer supported on *A* devices.
- New method: To distinguish SiI9024A transmitter (HDCP-equipped) from SiI9022 transmitter or SiI9022A transmitter (not HDCP-equipped), software must read the AKSV value from registers starting at TPI 0x36, and check for a valid value (20 *1*s and 20 *0*s). This method is a more secure means of verifying host HDCP capability.

I²C Requirements

The host must have hardware reset control over the transmitter, typically through a GPIO pin. The transmitter notifies the host of service needs through an interrupt line. All other communication occurs by way of the local I²C interface.

I²C Access Speeds. The internal hardware I²C interface of the transmitter runs at speeds up to 400 kHz. This throughput can also be achieved on the DDC interface, if the connected device supports it, when using the pass-through I²C feature of the TPI register set.

The host software is responsible for ensuring its DDC access speed does not exceed the limitations of any device present on the bus – all devices must be determined capable of supporting 400 kHz operation before the host attempts to do any accesses at that speed. Speeds in excess of 100 kHz may not be allowed by the interface specification in use.

→ **Important Note:** There is a lower speed limit of 40 kHz on DDC pass-through operation.

I²C Access Addresses. Use I²C address 0x72 to access the TPI registers. The transmitter can optionally respond at address 0x76 depending on the CI2CA hardware strap setting described in the transmitter data sheet. Certain legacy registers, not supported on all devices, respond at 0x7A (or 0x7E depending on CI2CA pin strapping).

The I²C address used to access the CPI registers is noted in the *CEC Programming Interface (CPI) Programmer's Reference* (SiI-PR-0041). In a TPI transmitter environment, the host masters one I²C interface, and the transmitter masters another, as illustrated for a typical transmitter in Figure 1. For transmitters without HDCP capability, the I²C paths are the same, but the DDC bus is not used for HDCP in this case.

- **Host TPI/CPI/DDC I²C Interface.** The transmitter implements this slave interface, mastered by the host. The host serially transmits and receives data over this bus, which implements a register interface structure where the host writes video mode and HDCP control data and reads back connection status information.
- **DDC I²C Interface.** When using the native TPI operating interface, the transmitter implements this interface as a pass-through connection from the host to the DDC bus of the HDMI TV. The transmitter has the ability to block the graphics host from accessing this bus when using it for HDCP, and also monitors the bus for activity to prevent breaking an active connection.

Summary of Host Software Tasks

Managing TPI operations is straightforward, starting from when the host resets the subsystem after power-up.

Initialization. After reset, the subsystem comes up in a pre-determined state:

- Standby (D2) state.SiI9022A/SiI9024A transmitter

Host software initializes it by:

- Configuring the registers according to known video/audio input format information
- Enabling any interrupt events to be signaled on the INT pin

From this point, there are various service conditions that will be called for.

Hot Plug Service. Until a device is attached, the host just monitors the connection status either through interrupts or by polling. The host software waits for an attached Sink hot plug event, and then:

- Reads the downstream Sink EDID information
- Powers up the transmitter (D0 state)
- Sends active video corresponding to the Sink capabilities, and selects DVI or HDMI mode, depending on the EDID information found

Operational State. The subsystem manages most routine HDMI activities in Operational state. The host software usually only needs to:

- Change video modes
- Enable HDCP operation and check the KSV list against known bad keys in the SRM
- Monitor ongoing HDCP operation, through interrupt service or by polling

Unplug Service. Once the subsystem detects a hot unplug event, it:

- Notifies the host through INT (if the interrupt is enabled)
- Disables HDCP and restores certain other logic states
- On the next *plug-in* event, drives the INT pin active (if the interrupt was previously enabled)

  Within the host software, Unplug Service events are typically handled by the Hot Plug Service routine.

Reserved Bits. Bits marked *RSVD* or *Reserved* should be written as 0 when other bits in the register are accessed, unless otherwise noted.

Detailed Sequences

In the following descriptions:

- *host* refers to activities of the software for the graphics host processor.
- *TPI* refers to use of the I^2C interface to access the TPI registers implemented in the transmitter itself.
- *Tx* refers to the respective transmitter chip variant. *Rx* refers to the HDCP registers on the remote HDMI receiver.

Initialization

The following steps are required to prepare for operation.

Step	Function	Registers Involved	Page	Details
1	Reset and Initialize	0xC7	—	host → TPI: Hardware Reset to Tx subsystem The host must reset the chip, then write **0x00** to **0x72:0xC7** to enable TPI mode.
2	Detect Revision	0x1B–0x1D	9	host ← TPI: Detect Tx type and TPI revision. When TPI 0x1B can be read correctly, the TPI subsystem is ready.
3	Power up transmitter	0x1E	33	host → TPI: Enable active mode. Write TPI 0x1E[1:0] = 00
4	Configure Input Bus and Pixel Repetition	0x08	11	host → TPI: Select input bus characteristics like pixel size, clock edge
5	Select YC Input Mode	0x0B	15	host → TPI: Select YC input mode and signal timing features
6	Configure Sync Methods	0x60–0x61	13	host → TPI: Choose the sync method register group to access, enable YC Mux mode
7	Configure Explicit Sync DE Generation	0x62–0x6D with 0x60[7] = 0	16	host → TPI: Define parameters for explicit sync method
8	Configure Embedded Sync Extraction	0x62–0x6D with 0x60[7] = 1	18	host → TPI: Define parameters for embedded sync method Note that the TPI 0x63 values must be rewritten after TPI 0x19 is written.
9	Set up Interrupt Service	0x3C	29	host → TPI: Enable hardware interrupts to be serviced (TPI 0x3C).

→ **Important Note:** For TPI operation, **always** write device address 0x72, register offset 0xC7 = 0x00 as the first step after hardware reset.

Step 1 above is mandatory to allow operation of the TPI register set on HDMI transmitters. If the write is not done as the first register write, the transmitter will revert to Compatible Mode register set operation; the TPI registers will not be accessible. Refer to Appendix A for details.

After powering up the transmitter, the host should write the following sequence to enable source termination.

```
a) Write 0xBC=0x01              // Internal page 0
b) Write 0xBD=0x82              // Indexed register 82
c) Read 0xBE                    // Read current value
d) Modify bit[0] = 1            // Enable source termination
e) Write 0xBE                   // Write back modified value
```

Servicing a Hot Plug Event

The host must recognize and handle Plug and Unplug events either by polling or interrupts as described in the Hot Plug Management section. Once a Plug event has been confirmed, the host follows the sequence below.

Step	Function	Registers Involved	Page	Details
1	Host reads EDID and sets interface type	0x1A	22	host → Tx → Monitor: Use DDC to access EDID. Host uses TPI 0x1A to request access to the DDC bus; scans EDID and reads video information, audio descriptors, and HDMI signature; then writes to TPI 0x1A to set interface type detected (HDMI or DVI) and to release the DDC bus.
2	Host sends video and audio	—	—	host → Tx: Begin transmitting video and audio streams.
3	Host sets video format and resolution	0x00–0x0A	10	host → TPI: Set the VMode and Format registers. Writing VMode optimizes the operation of the internal logic for the particular speed selected. Expansion and compression are set in the Input and Output Format registers.
4	Host sets AVI InfoFrame	0x0C–0x19	19	host → TPI: Set AVI InfoFrame. For an HDMI sink, write the AVI InfoFrame registers (TPI 0x0C-0x19). For a DVI sink, these registers must be cleared.
5	Host optionally selects YC Mux mode	0x60	15	host → TPI: Set YC Mux Mode if needed. Any time TPI 0x19 is written, TPI 0x60[5] will be reset to its default state and must be explicitly restored to enable YC Mux (one- to two-data-channel de-mux) mode.
6	Host optionally configures Embedded Sync Extraction	0x63 with 0x60[7] = 1	18	host → TPI: Define parameters for embedded sync method Any time TPI 0x19 is written, TPI 0x63 will be reset to its default state and must be explicitly restored.
7	Host sets audio mode	0x1F–0x28	23	host → TPI: Host parses audio descriptors from EDID, then sets audio configuration using capabilities information found.
8	Host enables video output	0x1A	22	host → TPI: Enable TMDS Output. Write TPI 0x1A[4] = 0.
9	Host chooses pixel repetition rate	0x08	—	host → TPI: Select pixel repetition rate. This setting must always be made *after* changing TPI 0x1A[4] from 1 to 0.
10	TPI looks for sink HDCP capability	—	—	TPI → Monitor: HDCP read of Bksv, Bcaps from Rx. This process happens without host intervention, triggered by the enabling of the TMDS output. This step occurs for HDCP-capable transmitters only, and can take up to 2 seconds to complete depending on sink readiness.

After configuration, the chip must be explicitly commanded to enter active (D0) mode as noted above. Video, audio, and interrupt settings can also be changed after entering D0 state, but changing the audio input selection may cause disturbances in output video and audio streams. Refer to the Transmitter Power State Register section on page 33 for information on entering various power states.

Changing Video Modes

Switching input video or audio modes requires a resolution change sequence. Note that if not using AV Mute, the host must **disable** HDCP before changing video modes. Otherwise, the disruption in video could break authentication at the receiver.

Step	Details
1	host → TPI: Blank the Display (optional). Write TPI 0x1A[3] = 1 to send an AV Mute control packet to blank the sink device. Wait at least 128ms to allow control InfoFrames to pass through to the sink device.
2	host → TPI: Prepare for Resolution Change. Write TPI 0x1A[4] = 1 to start the resolution change process.
3–10	Same as steps 3–10 in the Servicing a Hot Plug Event section.
11	host → TPI: Enable Display (optional). Write TPI 0x1A[3] = 0 to un-blank the sink device.

HDCP Authentication and Encryption

The hardware takes control of the DDC bus during HDCP operation. Refer to the DDC Bus Request/Grant section on page 22 for limitations on using the DDC bus while HDCP is enabled.

Step	HDCP-Capable Transmitter Details
1	host ← TPI: HDCP available? (TPI 0x30 for chip availability, TPI 0x29[1] for Sink availability).
2	host ← TPI: Read Aksv (TPI 0x36 group) and Bksv (TPI 0x2B group) for verification purposes (check each for 20 *1*s and 20 *0*s).
3	host → TPI: Request link security (TPI 0x2A).
4	Tx → Rx: Authenticate local link, authenticate repeater link, start link integrity check.
5	host ← TPI: Await *link secure* status (TPI 0x29).
6	host → Tx: Send secure content.
7	host ← TPI: Periodically verify *link secure* status (TPI 0x29).

HDCP Revocation Check

Step	HDCP-Capable Transmitter Details
1	host ← TPI: Repeater attached? (TPI 0x29).
2	host ← TPI: Extended link (through repeater) secure? (TPI 0x29).
3	host → TPI: Prepare to read KSV list (use Req/Gnt, TPI 0x1A).
4	host → TPI: Select V* value to read (TPI 0x31) host ← TPI: Read V* value (TPI group 0x32–35) Repeat until finished.
5	host: Calculate V value from repeater information and KSV lists read, and compare with V* for a match.
6	host: Check for revoked keys.
7	host: Abort if V != V* or a KSV key is on the revoked keys list.

General Programming Registers

The TPI register set in transmitter applications provides access to data structures through the following register groups.

- Identification
- Video Mode
- Audio Configuration
- Power State Management
- HDCP
- Interrupt Service

Figure 2 illustrates how the TPI subsystem presents a register interface of several local configuration and operational register groups to the host. The example shows an HDCP-capable transmitter; the logic for a non HDCP-capable transmitter is similar, but omits the HDCP blocks.

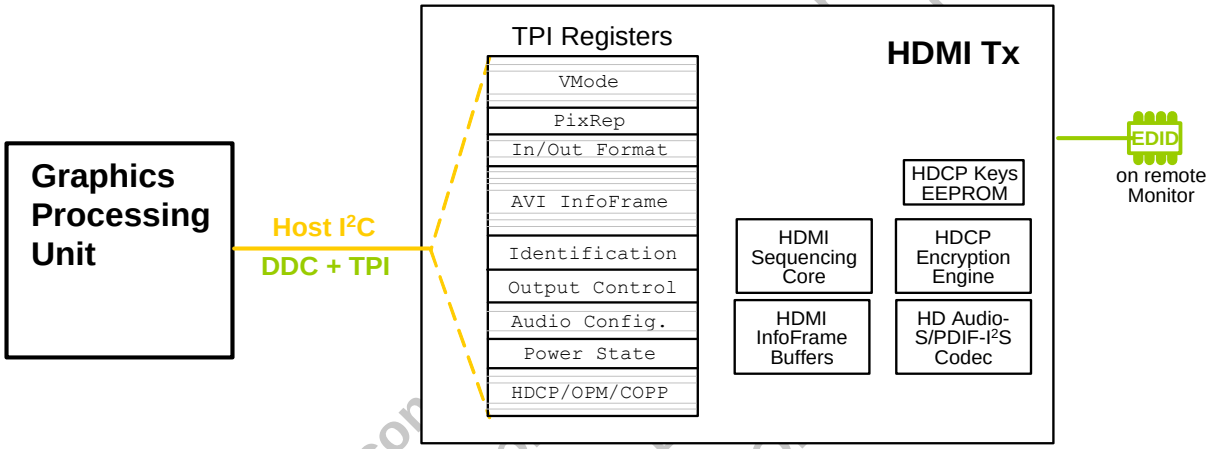

Figure 2. Transmitter TPI Register Interface to Host

The sections below describe the interaction required by the host to set up and operate the HDMI platform.

Identification

The ID registers return the device ID and TPI revision ID. The ID registers are listed in Table 3 on the next page. HDCP-capable and non HDCP-capable transmitters are distinguishable only by reading the HDCP revision register (TPI 0x30).

Access. These registers are accessed as single bytes.

Table 2. TPI Identification Registers (RO)

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x1B	Device ID Refer to Table 3							
0x1C	Device Production Revision ID revision level (major.minor) Refer to Table 3							
0x1D	TPI Scheme 0 – Hardware (always) 1–Software	TPI revision level (major.minor) Refer to Table 3						

Table 3. Device ID Information

Device	Device ID TPI 0x1B	Device Production Revision ID TPI 0x1C	TPI Revision ID TPI 0x1D	HDCP Revision TPI 0x30
SiI9022A Tx	0xB0	0x02	0x03	0x00
SiI9024A Tx	0xB0	0x02	0x03	0x12

Video Configuration

Setup for the incoming video resolution at minimum requires video mode and AVI InfoFrame data. The registers below describe the interface.

Video Mode

The host notifies TPI of its intended resolution through the following read/write registers.

Note: This information must be written for both HDMI mode and DVI mode operation.

Basic Video Mode Data

The video host provides the video mode data listed below. TPI requires that these values be set for proper operation.

Access. These registers can be read or written individually or by bursts as desired. The actual write to the HDMI transmitter registers takes place only once the final byte of the burst is written.

Table 4. TPI Video Mode Data (R/W)

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x00 [00]	PixelClock Pixel Clock / 10000 – LSB							
0x01 [00]	MSB							
0x02 [00]	VFreq Vertical Frequency in Hz – LSB							
0x03 [00]	MSB							
0x04 [00]	Pixels Total Pixels per line – LSB							
0x05 [00]	MSB							
0x06 [00]	Lines Total Lines – LSB							
0x07 [00]	MSB							

PixelClock. The logic uses this value to configure the internal PLL circuitry for optimal operation at the incoming frequency.

Input Video Mode Data

The input bus clocking format, along with clocking rate and edge, are specified in this register. The video host also indicates the pixel repetition factor here. Refer to the Format Matching section below for an example.

Access. This register is accessed as a single byte which must be written **after** TPI 0x1A[4] is set to 0, also any time TPI 0x1A[4] goes from 1 to 0, TPI 0x08[3:0] will be reset to their default state.

Table 5. TPI Input Bus and Pixel Repetition Data (R/W)

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x08 [60]	InputBusFmt							
	TClkSel Ratio of output TMDS clock to input video clock 00 – x 0.5 01 – x 1 (default) 10 – x 2 11 – x 4		Input Bus Select 0 – half-pixel wide 1 – full pixel wide (default)	Edge Select 0 – Falling edge 1 – Rising edge	PR3:0 Pixel Repetition Factor* 0000 – Pixel data is not replicated 0001 – Pixels are sent two times each 0011 – Pixels are sent four times each All others – RSVD			

***Note:** PR3:0 are cleared whenever TPI 0x1A[4] goes from 1 to 0.

Input Bus Select. The input data bus can be either one pixel wide or one-half pixel wide. The bit defaults to 1 to select full pixel mode. In ½ pixel mode, the full pixel is brought in on two successive clock edges (one rising, one falling). Support is provided for 24-bit full-pixel and 12-bit half-pixel input modes.

Edge Select. Input data is latched on the selected rising or falling clock edge. For ½ pixel mode, this bit indicates when = 0 that data present at the falling edge is latched first, and when = 1, data present at the rising edge is latched first. The high-order bits are latched first.

TClkSel. If the video host drives in data using anything other than a 1:1 ratio of input clock speed to TMDS clock speed, the host must program the clock multiplier logic. Refer to the Format Matching section below for an example.

Format Matching

For certain combinations of video input clock frequency and audio sampling rate, the HDMI transmitter must use a higher multiple of the input pixel clock when sampling the S/PDIF input.

Set PR3:0 to reflect the pixel replication factor of the input data stream so that it is properly decoded. `TClkSel` indicates the factor by which the input clock (IDCK) must be multiplied to yield the output clock frequency. These settings ensure that the output clock can provide sufficient bandwidth for multi-channel audio data on an HDMI link. The pixel replication count bits in the AVI InfoFrame Packet must be accurate. Refer to Table 6 for an example based on the 480p instance cited in section 7.3 of the *HDMI Specification version 1.3a*.

Table 6. 480p Mode Format Matching Example

Input Clock	Write to TPI 0x01:00 (decimal)	Audio Mode & max fs	TPI 0x08 PR3:0 Pixel rep.	YC Mux Mode (TPI 0x60[5])	TClkSel (refer to previous page)	Output Pixel Rep.	Resulting TMDS Link Clock	AVI InfoFrame Packet Byte 5 bits PR3:PR0
54 MHz	5400	8 ch, 96 kHz	0b01 (2x)	0	01 (1.0)	2x[7]	54 MHz[6]	0001
54 MHz	5400	2 ch, 192 kHz	0b01 (2x)	0	00 (0.5)	1x	27 MHz	0000
54 MHz	5400	8 ch, 96 kHz	0b00 (1x)	1[5]	01	2x[7]	54 MHz[6]	0001
54 MHz	5400	2 ch, 192 kHz	0b00 (1x)	1[5]	00	1x	27 MHz	0000
27 MHz	2700	8 ch, 96 kHz	0b00 (1x)	0	0b10 (2.0)	2x[7]	54 MHz[6]	0001
27 MHz	2700	2 ch, 192 kHz	0b00 (1x)	0	0b11 (4.0)	4x[8]	108 MHz	0011
27 MHz	2700	8 ch, 48 kHz	0b00 (1x)	0	0b01 (1.0)	1x[9]	27 MHz	0000

Notes:

1. Input Clock (IDCK) and the TMDS Link Clock must be within the min/max range for the HDMI transmitter.
2. For proper decoding, set PR3:0 to reflect the pixel replication factor of the input data stream.
3. `TClkSel` selects the factor by which the input clock must be multiplied to give output clock frequency.
4. There is only one pixel per 27 MHz clock cycle, so each must be replicated.
5. When YCbCr 4:2:2 data is multiplexed onto a single channel, the input clock must be doubled.
6. 54 MHz is necessary so that the blanking intervals have sufficient bandwidth to carry the 8-channel audio data sampled at frequencies up to 96 kHz.
7. Because the output clock has been doubled, pixels must be replicated.
8. Illustrates 4x pixel replication on output.
9. 27 MHz input clock provides sufficient bandwidth for 8-channel audio data sampled at frequencies 48 kHz and below. Refer to the *HDMI Specification*.
10. Bits PR0:PR3 of Byte 5 of the AVI InfoFrame packet indicate to the HDMI sink how many repetitions of each unique pixel are transmitted. Refer to the *CEA-861-E Specification*.

These settings ensure that the output clock provides sufficient bandwidth for multi-channel audio data on an HDMI link (25 MHz minimum).

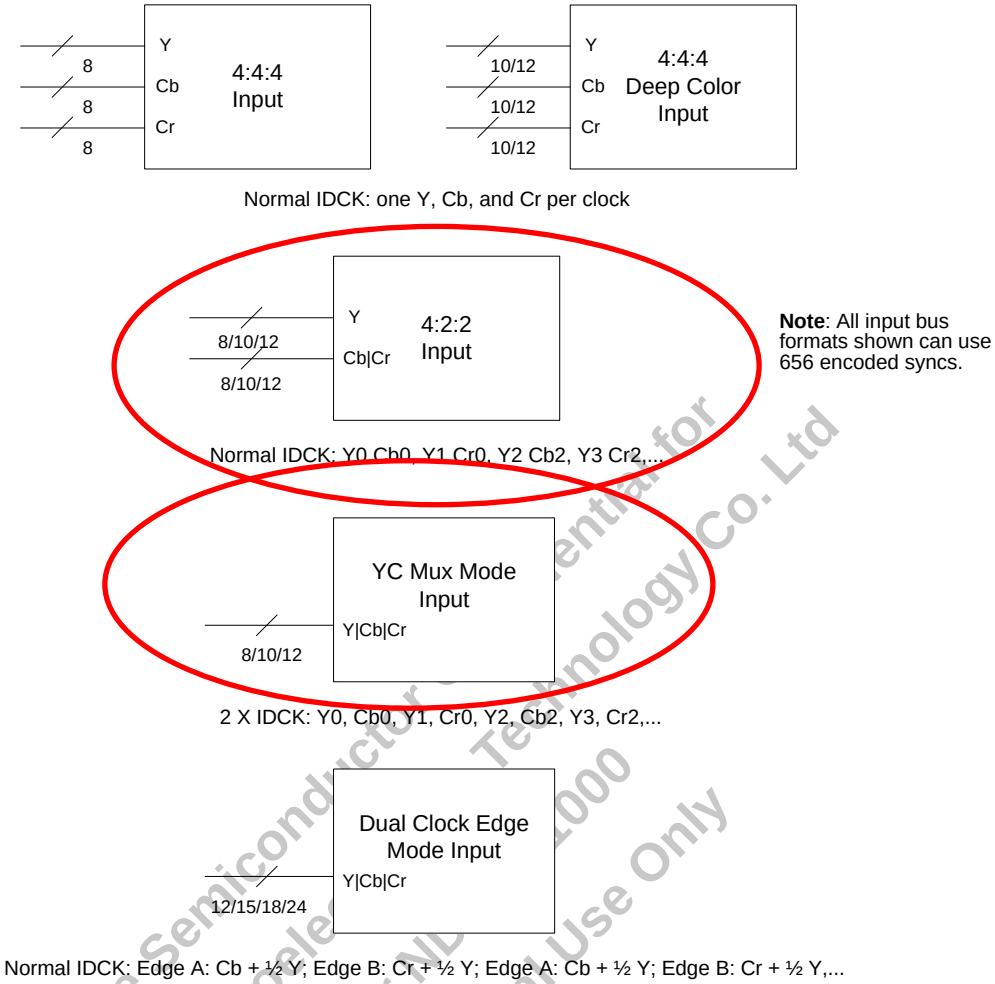

Figure 3. Input Bus Diagram for Different Formats

Input and Output Format

`InputFormat` and `OutputFormat` are used by the host to specify the data format and range. At a reset event, the chip defaults set input and output to be full-range RGB.

To set these registers, the host should read the EDID to determine whether the sink is DVI or HDMI, and what its preferred output format is. Once written, the selection remains until overwritten or until the next reset event.

Input Color Space / Output Format. These bits should be set as needed. Note that the settings made to these bits do not take effect until the AVI InfoFrame registers are programmed, so that any color space change can be synchronized to the color settings specified in the InfoFrame (avoiding temporarily green or pink images).

Range Compression. Range compression is enabled when RGB input and YCbCr output conversion is selected.

Range Expansion. Range expansion is enabled when YCbCr input and RGB output conversion is selected

Range Override. Range Expansion is associated with the Input Format; the Range bits either force expansion (01) or block it (10). Range Compression is associated with the Output Format; the Range bits either block compression (01) or force it (10).

Input Color Depth. This setting selects the incoming bus width to allow for proper handling on output. Note that only two YCbCr input bus widths, 8-bit and 12-bit, are possible. For 4:2:2 inputs 9, 10, or 11 bits wide, the host should set TPI 0x09[7:6] = 11 (12 bits), and drive unused bits of the video stream to 0.

Black Mode. This setting disables the video data input bus, forcing the video output to black (as long as valid video clock and control inputs are still available). This setting should not be used when HDCP is enabled, as the screen will show snow.

Access. These registers can be accessed individually or by bursts as desired. For writes, the actual write to the HDMI transmitter logic takes place only once the final byte of the burst write to TPI 0x0C-19 occurs (refer to the Input Color Space / Output Format note above).

Table 7. TPI AVI Input and Output Format Data (R/W)

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x09	InputFormat							
[00]	Input Color Depth 00 – 8-bit 01 – Rsvd 10 – 10/12-bit un-dithered for 4:2:2 mode 11 – 10/12-bit dithered to 8, for 4:2:2 mode		RSVD		Video Range Expansion 00 – Auto-selected by [1:0] 01 – On 10 – Off 11 – Rsvd		Input Color Space 00 – RGB 01 – YCbCr 4:4:4 10 – YCbCr 4:2:2 11 – Black Mode	
0x0A	OutputFormat							
[00]	RSVD			Color Space Standard 0 – BT.601 conversion 1 – BT.709 conversion	Video Range Compression 00 – Auto-selected by [1:0] 01 – Off 10 – On 11 – Rsvd		Output Format 00 – RGB 01 – YCbCr 4:4:4 10 – YCbCr 4:2:2 11 – RGB (same as 00)	

Input Setup Operations

Programming setup for the device is straightforward. A typical startup configuration sequence is shown below.

Write:

```
1. 0x00=0x00   // Set pixel rate for 74.24MHz (7424 = 0x1D00)
2. 0x01=0x1D   //
3. 0x1A=0x11   // Disable TMDS output
4. 0x1E=0x00   // Enter full-operation D0 state
5. 0x26=0x50   // Set to S/PDIF, Mute
6. 0x25=0x03   //
7. 0x27=0x00   //
8. 0x26=0x40   // Unmute S/PDIF
9. 0x1A=0x01   // Enable TMDS output
```

Sync Generation Options

For input video modes that do not provide explicit HSYNC, VSYNC, and/or DE signals, the transmitter logic offers two methods for sync signal generation:

- DE Generation (when explicit HSYNC and VSYNC signals are provided)
- Sync Extraction (when incoming video uses the ITU 656 method for embedding sync information).

The register sets for both modes overlap, so only one can be accessed at a time. However, by toggling between the two groups, features from both can be intermixed.

The dual data paths are shown in Figure 4, with the path for embedded sync extraction highlighted. When decoding syncs from the embedded sync stream, the DE Generator block should be disabled through TPI 0x63[6].

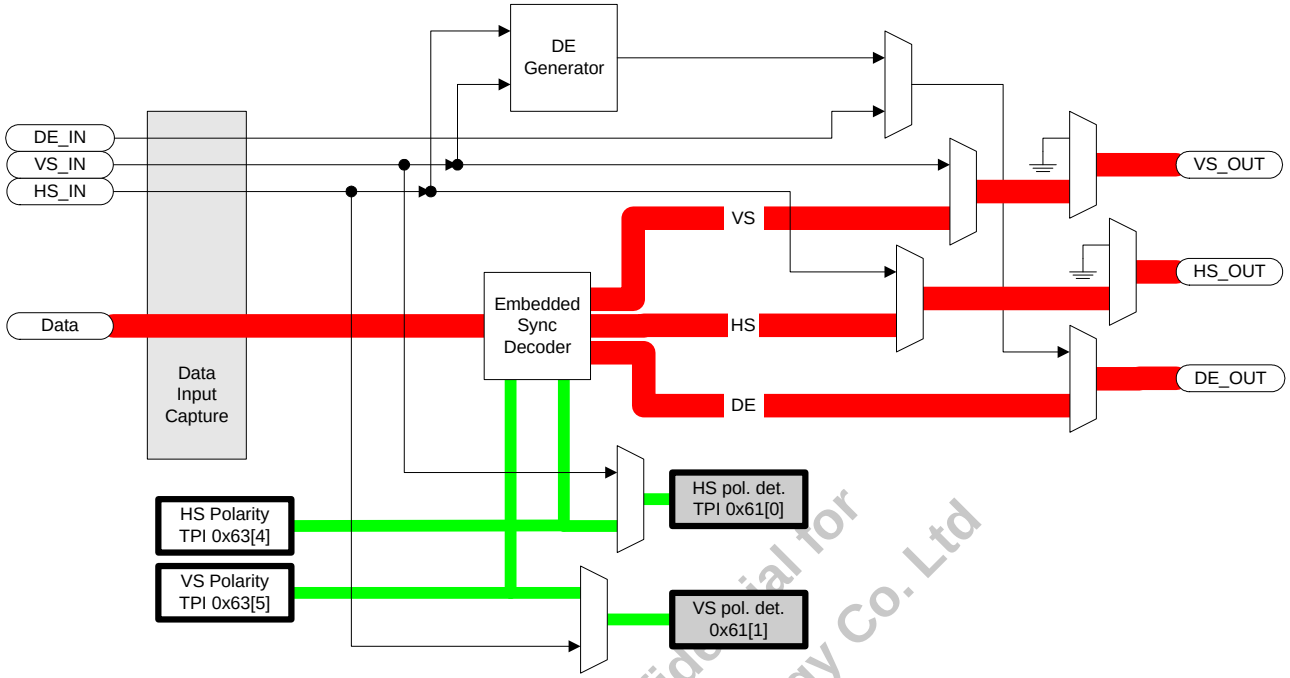

Figure 4. Transmitter Video Data with Sync Decoding

The registers available at TPI 0x62–0x6D depend on the setting of TPI 0x60[7].

- Setting TPI 0x60[7] = 0 selects external sync and access to the DE Generator registers. The DE signal can be generated internally and sent over TMDS if TPI 0x62–0x6D are set and then TPI 0x63[6] is set to 1.
- Setting TPI 0x60[7] = 1 allows access to the Embedded Sync Extraction registers. The DE, HSYNC, and VSYNC signals can then be extracted and sent over TMDS if TPI 0x62–0x69 are set and then TPI 0x63[6] is set to 1.

Features from both groups can be enabled together, by enabling the features of each group with TPI 0x60[7] set appropriately.

Access. These registers are accessed as single bytes or as part of a burst.

→ **Important Note:** TPI 0x60 must be written **after** the AVI InfoFrame is set. Any time TPI 0x19 is written, TPI 0x60[5] will be reset to its default state.

Table 8. Sync Register Configuration and Sync Monitoring Registers

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x60 [04]	Sync Generation Control Register							
	Sync Method 0 – External 1 – Embedded	Rsvd	YC Mux Mode One- to two-data-channel de-mux 0 – Disable 1 – Enable	Invert Field Polarity 0 – Leave bit as is 1 – Invert field bit	RSVD	DE_ADJ# 0 – Enable (recom-mended) 1 – Disable (default)	F2VADJ Adjust VBIT to VSYNC per bit [0] 0 – Disable (default) 1 – Enable	F2VOFST Adjust VBIT to VSYNC if bit [1] = 1 0 – Decrement by 1 1 – Increment by 1
0x61 [00]	Video Sync Polarity Detection Register (RO)							
	RSVD					Interlace Mode detected 0 – non interlaced 1 – interlaced	Input VSYNC polarity detected 0 – active high (leading edge rises) 1 – active low (leading edge falls)	Input HSYNC polarity detected 0 – active high (leading edge rises) 1 – active low (leading edge falls)

DE_ADJ# enables detection circuits to locate the position of VSYNC relative to HSYNC and only include HSYNC edges that are greater than ¾ lines from VSYNC in the line count for DE_TOP. Clearing this bit enables the function and is recommended for normal operation. Setting it high disables VSYNC adjustments and is not a recommended setting.

F2VADJ adjusts the VBIT_TO_VSYNC value during field 2 of an interlace frame; **F2VOFST** sets the direction of adjustment (increment or decrement by 1).

Invert Field Polarity. The Invert Field Polarity bit is used when the 656 Flag Bit is opposite the standard polarity for Field1 and Field2. Inverting polarity causes the sync extraction to format HSYNC and VSYNC properly based on the 'F' bit. In embedded sync mode, the transmitter does not detect 'even' from 'odd' field, except based on the setting of the 'F' bit. With explicit syncs, the transmitter simply encodes HSYNC and VSYNC across the HDMI/TMDS link without regard for field sequence.

YC Mux Mode

When YCbCr 4:2:2 data is multiplexed onto a single channel (YC Mux mode), the input clock rate from the host must be doubled. The one- to two-data-channel de-mux feature is used to decode the incoming data accordingly. When using YC Mux mode (TPI 0x60[5]=1), the input data can be additionally manipulated using the register controls provided below.

Table 9. TPI YC Input Mode Select (R/W)

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x0B [00]	Swap MSB and LSB within each data channel. 0 – Disable 1 – Enable	Video Input DDR Select 0 – Lower 12 bits 1 – Upper 12 bits	RSVD		Non Gap Mode 0 – Disable 1 – Enable	YC Input Mode Select. 000 – YC normal 001 – YC channel data swap 010 – YC 8-bit DDR input 011 – YC 8-bit DDR input, channel data swap 100 – YC 24-bit input in consecutive order All other values = Rsvd		

Swap MSB and LSB swaps the most significant and least significant bytes of the incoming data. This feature is not supported in YC Mux Mode.

Video Input DDR Select chooses the pins on which 12-bit DDR video will arrive.

- Lower 12 bits D11:0 are the default input path.
- Upper 12 bits D23:12 are available as an optional input path, for the SiI9022A and SiI9024A devices only.

YC Input Mode Select adds new input bus formats when using YC Input modes. Refer to the *SiI9022A/SiI9024A HDMI Transmitter Data Sheet* (SiI-DS-1053) for details.

- *YC channel data swap* reverses the order of data on the 1st and 2nd clocks, normally C-then-Y, to Y-then-C.
- *YC 8-bit DDR input* enables double-data-rate clocking, so that the C data comes on the rising (or falling) clock edge, and Y data on the falling (or rising) clock edge.
- *YC 8-bit DDR input, channel data swap* enables both DDR and swap, so that the Y data comes on the rising (or falling) clock edge, and C data on the falling (or rising) clock edge.
- *YC 24-bit input in consecutive order* enables full data input of all 24 bits on a single clock edge.

DE Generator Register Set

The DE Generator registers are accessible only when TPI 0x60[7] = 0.

The transmitter provides an explicit sync DE generator that allows the Data Enable (DE) signal, required to be encoded in the TMDS output, to be derived from the incoming HSYNC and VSYNC signals. When programmed through TPI registers 0x62–0x6D, the output DE is generated based on the values in the DE generator registers and the arrival times of the HSYNC and VSYNC pulses from the video source. These registers are used only for DE generation and do not affect HSYNC or VSYNC.

The registers are shown diagrammatically in Figure 5. The vertical sync pulse (VSYNC) occurs in the top area of the frame, before the active video area. The active (leading) edge is shown with an arrow. The horizontal sync pulse (HSYNC) occurs every line before the active video area during the DE_DLY time. The active (leading) edge of HSYNC is shown with an arrow. (Note that VSYNC and HSYNC widths are not shown to scale.)

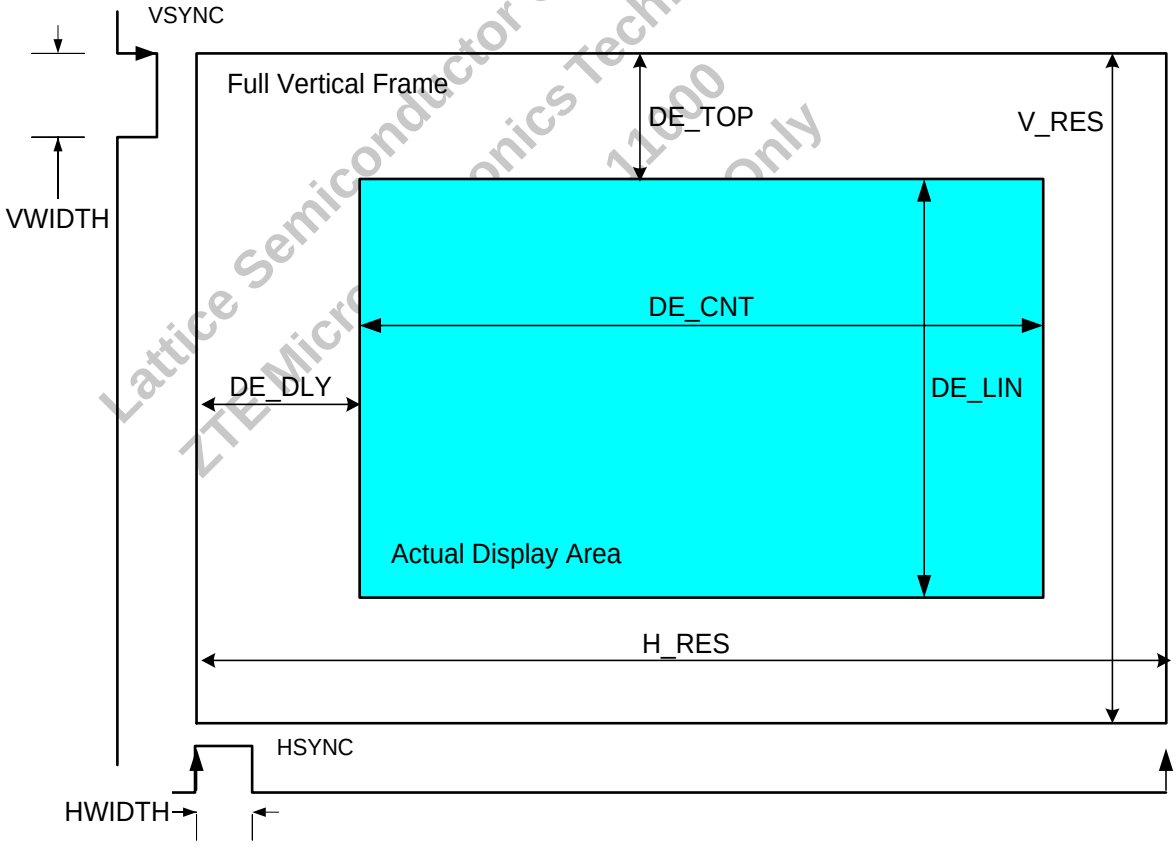

Figure 5. DE Generator Measurements

In the following definitions, *pixels* is used to mean *unique pixels*. The counts in the DE Generator registers, if expressed in pixels, are counted according to the original clock, even if that original input clock is multiplied internal to the chip.

For example, a 480i field will contain 720 unique pixels per line in the active video area, even when the clock is multiplied to 1440 clock cycles per active video time.

Polarity, To change Vsync and Hsync Polarity use the following sequence:

1. Switch to Explicit Sync DE Generator configuration (TPI 0x60[7] = 0).
2. Modify Vsync and Hsync polarity (TPI 0x63[5:4]).
3. Stay in Explicit Sync DE Generation configuration (TPI 0x60[7] = 0) or switch to Embedded Sync Extraction configuration (TPI 0x60[7] = 1).

Default. The DE Generator defaults to 'disabled' after reset.

Access. These registers can be read or written individually or by bursts (the registers available at TPI 0x62–0x6D depend on the setting of TPI 0x60[7]) as desired. All are read/write except as noted.

Table 10. Explicit Sync DE Generator Registers (TPI 0x60[7] = 0)

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x62 [00]	DE_DLY[7:0]							
0x63 [00]	RSVD	DE Generator 0 – Disable 1 – Enable	VSYNC Polarity: 0 – Positive (leading edge rises) 1 – Negative (leading edge falls)	HSYNC Polarity: 0 – Positive (leading edge rises) 1 – Negative (leading edge falls)	RSVD		DE_DLY[9:8]	
0x64 [00]	RSVD	DE_TOP[6:0]						
0x65 [00]	RSVD							
0x66 [00]	DE_CNT[7:0]							
0x67 [00]	RSVD				DE_CNT[11:8]			
0x68 [00]	DE_LIN[7:0]							
0x69 [00]	RSVD					DE_LIN[10:8]		
0x6A [00]	H_RES[7:0] (RO) – Measures the time between two HSYNC active edges. Unit of measure is pixels.							
0x6B [00]	RSVD				H_RES[11:8] (RO)			
0x6C [00]	V_RES[7:0] (RO) – Measures the time between two VSYNC active edges. Unit of measure is lines.							
0x6D [00]	RSVD				V_RES[11:8] (RO)			

DE_DLY[9:0] set the width of the area to the left of the active display. Unit of measure is pixels. This register should be set to the sum of (HSYNC width) + (horizontal back porch) + (horizontal left border), and is used only for DE generation. Valid range is 1–1023. 0 is invalid.

HSYNC and VSYNC Polarity bits 0x63[5:4] should be set to the polarity of the source providing the sync signals. Setting these bits does not change the polarity of HSYNC or VSYNC in the encoded TMDS output stream.

DE_TOP[6:0] define the height of the area above the active display. The unit of measure is lines (HSYNC pulses). This register should be set to the sum of (VSYNC width) + (vertical back porch) + (vertical top border). Valid 1–127. 0 is invalid.

DE_CNT[12:0] define the width of the active display. Unit of measure is pixels. This register should be set to the desired horizontal resolution. The DE_CNT is 12 bits (valid values are 1 through 2047, 0 is invalid).

DE_LIN[11:0] define the height of the active display. The unit of measure is lines (HSYNC pulses). This register should be set to the desired vertical resolution. The DE_LIN is 11 bits (valid values are 1 through 2047, 0 is invalid).

H_RES [13:0] and V_RES [11:0] (Read Only) measure the time between two sync active edges and may vary slightly from one reading to the next. The values in these registers are accurate only when there are active HSYNC and VSYNC controls arriving on the video input. The H_RES is 12 bits (valid values are 1 through 4095, 0 is invalid).

Embedded Sync Register Set

The Embedded Sync Extraction registers are available only when TPI 0x60[7] = 1.

The transmitter provides logic to extract ITU 656 encoding from the video stream.

Table 11. Embedded Sync Extraction Registers (TPI 0x60[7] = 1)

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x62 [00]	HBIT_TO_HSYNC[7:0]							
0x63 [00]	RSVD	Embedded Sync Extraction[1] 0 – Disable 1 – Enable	RSVD				HBIT_TO_HSYNC[9:8] 1	
0x64 [00]	FIELD2_OFST[7:0]							
0x65 [00]	RSVD			FIELD2_OFST[12:8]				
0x66 [00]	HWIDTH[7:0]							
0x67 [00]	RSVD						HWIDTH[9:8]	
0x68 [00]	RSVD		VBIT_TO_VSYNC					
0x69 [00]	RSVD		VWIDTH					

Notes:

1. The Embedded Sync Mode information at TPI 0x63 is always reset to its default disabled state whenever AVI InfoFrame register TPI 0x19 is written. Therefore, if embedded sync video input is being used, TPI 0x63 must be re-written after any write to TPI 0x19.
2. Ri check must be enabled while in embedded mode, see the section in HDCP implementation for the procedure.

The settings described below are useful only when the input video uses 656 encoded syncs. 0 is an invalid setting for all these parameters.

HBIT_TO_HSYNC[9:0] create HSYNC pulses. Set register to the delay from the detection of an EAV sequence (H bit change from 1 to 0) to the active edge of HSYNC. Unit of measure is pixels. Valid 1–1023.

FIELD2_OFST[11:0] determine VSYNC pixel offset for the odd field of an interlaced source. Set this to half the number of pixels/line. Valid 1–4095.

HWIDTH[9:0] set the width of the HSYNC pulses. Set the register to the desired HSYNC pulse width. Unit of measure is pixels. Valid 1–1023.

VBIT_TO_VSYNC sets the delay from the detection of V bit changing from 1 to 0 in an EAV sequence, to the asserting edge of VSYNC. Unit of measure is lines. Valid 1–63.

VWIDTH sets the width of the VSYNC pulse. Unit of measure is lines. Valid 1–63.

HSYNC and VSYNC Polarity TPI 0x63[5:4] should be set to the polarity of the source providing the sync signals. Setting these bits does not change the polarity of HSYNC or VSYNC in the encoded TMDS output stream.

InfoFrame Data

The transmitter register map provides simplified direct access to the commonly used AVI InfoFrame registers, and a separate overlay register space for all the other InfoFrame registers.

AVI InfoFrame Data

A dedicated AVI InfoFrame buffer is provided to meet HDMI requirements for transmission every 2 frames. The host fills in the AVI InfoFrame according to the HDMI 1.x specification. This data is used only for transmission, and is not used for any internal register control purposes.

Access. These registers can be read or written individually or by bursts as desired. The data sent out on the link gets updated only when TPI 0x19 is written, either by itself or as the last byte of a burst. Writing TPI 0x19 also causes the values last written to TPI 0x09-0A to be loaded into the chip logic.

> → **Important Note:** After switching to a DVI mode by using TPI 0x0A[1:0], at least the one byte at TPI 0x19 **must** be written in order for the mode switch to take place.

Table 12. TPI AVI InfoFrame Data (R/W)

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x0C [00]	AVI_DBYTE0 AVI InfoFrame Checksum							
0x0D [00]	AVI_DBYTE1							
	RSVD (0)	Y1:0 RGB/YCbCr Indicator		A0 Active info	B1:0 Bar Info Data Valid		S1:0 Scan Information	
0x0E [00]	AVI_DBYTE2							
	C1:0 Colorimetry Info		M1:0 Picture Aspect Ratio		R3:0 Active Format Aspect Ratio			
0x0F [00]	AVI_DBYTE3							
	ITC IT Content	EC2:0 Extended Colorimetry			Q1:0 RGB Quantization Range		SC1:0 Non Uniform Scaling	
0x10 [00]	AVI_DBYTE4							
	RSVD (0)	VIC6:0 Video Format Identification Code						
0x11 [00]	AVI_DBYTE5							
	YQ1:0 YCC Quantization Range		CN1:0 Content Type		PR3:0 Pixel Repetition Factor			
0x12 [00]	EndTopBar Line Number of End of Top Bar – LSB							
0x13 [00]	MSB							
0x14 [00]	StartBottomBar Line Number of start of Bottom Bar – LSB							
0x15 [00]	MSB							
0x16 [00]	EndLeftBar Pixel Number of End of Left Bar – LSB							
0x17 [00]	MSB							
0x18 [00]	EndRightBar Pixel Number of End of Right Bar – LSB							
0x19 [00]	MSB							

Calculating the Checksum. The host must calculate the checksum to include the InfoFrame identifier headers and the expected length byte, per the HDMI specification. The routine for generating the checksum is provided below.

```
byte CRC = 0x82 + 0x02 + 13; // Identifier code for AVI InfoFrame, length
byte TXAddr = AVI_IF_ADDR + 4;
byte i;
for(i = 1; i < 14; i++ )
      CRC += ReadByteHDMITXP1( TXAddr++ );
CRC = 0x100 - CRC;
regAVIInfoFrm[0]=CRC;
return CRC;
```

Using xvYCC

All transmitters support sending video data in xvYCC format across HDMI. Color space conversion is not available; input video data must be provided in the correct xvYCC format. The programming is the same as the YCbCr 4:4:4 mode of operation.

The Gamut Boundary Descriptor packet is written to the Miscellaneous InfoFrame buffer selected by TPI 0xBF [2:0] = 011. This buffer is also designated for use with MPEG packets, but these are rarely required. Contact Silicon Image technical support for recommendations on sending both GBD and MPEG packets in the same system.

Other InfoFrame Data

The host writes miscellaneous InfoFrame information as needed to the packet buffers, according to the HDMI 1.x specification. Seven buffers are provided, selected according to TPI 0xBF[2:0]. This data is used only for transmission, and not for any internal control purposes.

Access. These registers can be read individually or by bursts as desired. For writes, they should be accessed as a complete group in a burst. The actual data is transferred to the HDMI transmitter logic only once the final byte of the InfoFrame is written, as determined by the length of the InfoFrame set in TPI 0xBF[2:0]. Alternatively, individual bytes can be modified and then the updated data can be transferred by writing just to the final byte.

Table 13. TPI Miscellaneous InfoFrame Data (R/W)

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0xBF [00]	I-F_SELECT							
	Enable selected InfoFrame transmission 0 – not enabled (default) 1 – enable	Repeat selected InfoFrame transmission 0 – don't repeat (default) 1 – repeat each frame	RSVD			Selects the InfoFrame to load (*n* = 31 bytes except where noted) 000 - Reserved 001 – SPD or ACP 010 – Audio (*n* = 14 bytes) 011 – MPEG or GBD 100 – Generic 1 / ISRC1 101 – Generic 2 / ISRC2 110 – HDMI VSIF 111 – Reserved		
0xC0 [00]	I-F_TYPE							
	Type 0 – Packet 1 – InfoFrame	InfoFrame Type per CEA-861-E spec						
0xC1 [00]	I-F_VER							
	InfoFrame version per CEA-861-E spec							
0xC2 [00]	I-F_LENGTH							
	RSVD = 0			InfoFrame length per CEA-861-E spec				
0xC3 [00]	DBYTE0 InfoFrame Checksum							
0xC4 [00]	DBYTE1							
	InfoFrame data byte 1							
0xC5 [00] . . 0xDD [00]	DBYTE2.. DBYTEn-1							
	InfoFrame data bytes (0xCD is the last location for Audio InfoFrame)							
0xDE [00]	DBYTEn							
	Last InfoFrame data byte (for all except Audio InfoFrames)							

Select. The host chooses the InfoFrame to which the data corresponds, and sets the Enable and Repeat bits, in the `I-F_SELECT` byte. When the last byte (n) of that selected InfoFrame group is written, the Enable and Repeat bit settings originally written will take effect. This sequence allows the command, checksum, and data to be written as a continuous burst over the TPI I^2C interface, preventing a partially written InfoFrame from being sent.

Because there is a limited number of packet buffers, it may be necessary to alternate the buffers where appropriate. For example, ACP packets can be sent every 300ms normally, but interrupted on occasion to use the same buffer to send an SPD packet for identification of the source device.

Enable bit. Enables selected InfoFrame for transmission over HDMI. To enable transmission, write "1" into this bit. Once transmission is done, the hardware will reset it back to 0 unless the Repeat bit was also set.

Repeat bit. The Repeat bit causes the logic to repeatedly send the selected InfoFrame data each frame. When this bit is set along with Enable (bit [7]), hardware will try to send the InfoFrame once every vertical blanking period. Software has to disable this bit first to force clear of the Enable bit by the hardware.

Length. The length is calculated without including the type, version, length, or checksum bytes.

Calculating the Checksum. Refer to the previous section for an example of calculating the checksum, noting that the Identifier Code and Length values will change according to the specific InfoFrame being sent.

System Control

TPI has the ability to determine certain system functional needs and act automatically to control operation. The System Control register configures these functions and also allows the host to override them when needed.

Table 14. TPI System Control (R/W)

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x1A [10]	SystemControl							
	RSVD – write as 0 always.	Link Integrity Mode 0 – Static 1 – Dynamic	RSVD – write as 0 always.	TMDS Output Control 0 – Active 1 – Power down (default)	AV Mute 0 – Normal audio/video 1 – Mute HDMI audio/video at receiver	DDC Bus Request by host 0 – Host does not need to use DDC 1 – Host requests to use DDC	DDC Bus Grant by TPI Read: Status 0 – Bus not available 1 – Bus can be used by host Write: Force bus ownership 0 – TPI 1 – Host	Output Mode Select Write in Hot Plug Service Loop 0 – Set DVI 1 – Set HDMI

DDC Bus Request/Grant. TPI logic automatically utilizes the DDC bus as needed. If the host needs to use the DDC bus, it must notify TPI logic and then wait until it is safe to take the bus over. The host graphics software should use the DDC bus Request control and Grant status bits as follows.

1. Before performing a DDC bus access, the host writes TPI 0x1A[2] = 1 to request the bus.
2. It then loops on a read of TPI 0x1A[1] until it returns 1, indicating that it is safe to read the bus.
3. Once it has been granted the bus, the host **must** write TPI 0x1A = 0x06. This action closes the switch to allow subsequent I^2C accesses to flow out onto the DDC bus.
4. Once the host has finished reading EDID, it must write TPI 0x1A = 00 to clear the request. The device will **not** ACK this write.
5. Finally, the host must read TPI 0x1A[2:1] and verify that they are 00 (repeat step 4 if not). Reading back from this register opens the switch to direct subsequent I^2C accesses back to the TPI bus.

If link security has been requested any time since the last hardware reset or hot plug event, HDCP link integrity checking will be taking place every 128 frames. Therefore, in order to receive full access, the host must **disable** link security before requesting use of the DDC bus.

Output Mode. During the Hot Plug Service Loop, before TPI register 0x1E is written to 00 to select full-power operational mode, the DVI/HDMI control bit at TPI 0x1A[0] must be written to force the mode of operation.

0 = DVI

1 = HDMI

TPI register 0x1A can be written and read multiple times while in the Hot Plug Service Loop, but for bit 0, only the last write made is important. After writing TPI 0x1E = 00, the most recent setting of TPI 0x1A[0] is used to establish the operating mode.

The transmitter subsystem will operate in the selected mode (DVI or HDMI) until the next hardware reset, or until the next time TPI 0x1E is written to 0.

AV Mute. HDMI allows the display and audio output of the sink device to be muted by remote command. The host can send this command by using the AV Mute bit. This feature is useful when switching video resolutions, for example, to prevent the screen from flickering.

Important Notes

→ **1:** Once the system has been set to DVI output mode, the AV mute and un-mute packets can no longer be sent. Therefore, sending an AV Mute packet and then switching from HDMI to DVI output mode could result in a permanently muted screen until the next hardware reset.

→ **2:** Setting TPI 0x1A[3] sends an AV Mute command and temporarily disables HDCP encryption, as required for HDMI. However, as there is no concept of an AV Mute command for DVI, setting this bit when in DVI mode results only in disabling HDCP encryption. Therefore, in general, this bit should not be used in DVI mode.

TMDS Output Control. The transmitter TMDS outputs can be powered down, leaving them floating. TMDS output must be explicitly enabled with this bit after entering operational (D0) mode.

Link Integrity Mode. Not used for devices without HDCP. For details, refer to the HDCP section for devices with HDCP capability.

Resolution Change. Setting TPI 0x1A[4] = 1 is a signal from the host that a resolution change is about to happen. The host should disable HDCP security using TPI 0x2A[0], wait for at least 64ms or poll TPI 0x29[7:4] to verify that it is disabled, then follow this procedure.

1. Disable TMDS Output (TPI 0x1A[4] = 1).
2. Wait at least 128ms to allow control InfoFrames to pass through to the sink device.
3. Change video resolution, and wait for the new resolution to stabilize.
4. Set the VMode registers.
5. Write the AVI InfoFrame registers (TPI 0x0C).
6. Enable TMDS Output (0x1A[4] = 0).

Upon completion, re-enable HDCP security if needed using TPI 0x2A[0].

Audio Configuration

The audio subsystem can operate in S/PDIF or I^2S mode, S/PDIF or I^2S, as selected by the host through TPI. In either case, the host determines which configurations are valid by reading the audio descriptors directly from the EDID.

Configuring Audio using S/PDIF

The S/PDIF audio subsystem selection is made by TPI control. Most operations thereafter are automatic.

S/PDIF Initialization and Operation

The transmitter requires only the following initialization for S/PDIF applications.

1. Ensure that a valid S/PDIF audio stream is coming into the transmitter.
2. Select S/PDIF input mode using register TPI 0x26[7:6].

The transmitter supports automatic recognition of new S/PDIF audio rates when the correct header information is supplied in the S/PDIF stream. It continually checks the f_s value in the S/PDIF stream, and calculates N values accordingly whenever f_s changes.

For audio sources that do not supply correct S/PDIF stream headers, the information must be programmed manually as described in the Configuring Audio using I2S section.

Additional Audio Features

The following registers are provided for the host to select an audio mode. The default is Basic Audio, since HDMI hosts are always allowed to send Basic Audio even before reading the EDID descriptors. The Basic Audio configuration can always be selected.

Audio Muting. The Mute bit is used when the host has a specific need to temporarily turn off audio. It stops the audio stream to the audio FIFO, muting the audio. However, the audio module is still enabled, so that un-muting requires no additional re-initialization.

Audio Handling. Three modes are available.

- **Pass Basic Audio Only.** Incoming audio will be transmitted over HDMI only if is uncompressed (PCM) and is 32 kHz, 44.1 kHz, or 48 kHz. All other streams are blocked and the output is muted.
- **Pass All Audio Modes.** The hardware transmits the audio stream over HDMI exactly as it is received.
- **Down-sample Incoming Audio as Needed.** The hardware uses the transmitter down-sampler to automatically detect and convert incoming high audio rates to their Basic Audio correspondent. For example, 192 kHz audio would always be down-sampled to 48 kHz before transmission over HDMI.

For proper operation, the down-sampling feature requires that the incoming S/PDIF header information be correct.

Audio Override. By default, the incoming S/PDIF stream header is used by chip logic to automatically configure audio operation. Override bits are provided in registers TPI 0x26 and 0x27 to manually set this information. Note that these bits must be used to set the correct output stream header when the Audio Handling bits in TPI 0x25[1:0] = 10. Setting TPI 0x27[5:3], in particular, will overwrite the incoming audio stream rate data on the outgoing audio packets. Setting the override bits does not affect the Audio InfoFrame information.

Access. These are byte-accessed registers. Refer to the following section for sequence information.

Table 15. TPI Audio Configuration Write Data (RW)

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x24* [D0] read-only	S/PDIF Header bits 35:32, Sample Length refer to IEC60958 specification				S/PDIF Header bits 27:24, Sample Frequency refer to IEC60958 specification			
0x25* [03]	RSVD						Audio Handling 00 – Pass Basic Audio only 01 – Pass all audio modes 10 – Down-sample incoming audio as needed 11 – Do not check audio stream (default)	
0x26 [00]	Audio Interface 00 – Disabled 01 – S/PDIF 10 – I^2S 11– Reserved		Layout Bit Audio packet header layout indicator 0 – Layout 0 (2-channel) 1 – Layout 1 (up to 8 channels)	Mute 0 – Normal 1 – Mute	Audio Coding Type CT[3:0] (not used by chip logic) 0000 – Refer to Stream Header 0001 – PCM 0010 – AC-3 0011 – MPEG1 0100 – MP3 0101 – MPEG2 0110 – AAC 0111 – DTS 1000 – ATRAC all others are reserved			
0x27 [00]	Audio Sample Size SS[1:0] 00 – Refer to Stream Header 01 – 16-bit 10 – 20-bit 11 – 24-bit		Audio Sample Frequency SF[2:0] 000 – Refer to Stream Header 001 – 32kHz 010 – 44.1kHz 011 – 48kHz 100 – 88.2kHz 101 – 96kHz 110 – 176.4kHz (MCLK = 128 f_s) 111 – 192kHz (MCLK = 128 f_s)			RSVD		
0x28 [00]	RSVD							

***Note:** TPI 0x24 and 0x25 are specific to S/PDIF operating mode, and are available only when TPI 0x26[7:6] = 01.

Audio Configuration Overview. S/PDIF audio handling is fully automated with TPI. The audio logic automatically detects the incoming S/PDIF sampling frequency and makes the register settings to pass the audio through correctly. I^2S audio handling is nearly as automated; only the sampling frequency needs to be written since it is not available from header information.

The incoming audio data is raw data, which is not parsed in any way by the audio logic. Therefore, information such as audio sample size is not known. Therefore, the data written to TPI 0x26 and 0x27 serves to inform the transmitter of the source choices and preferences.

A typical configuration sequence takes place after a Hot Plug event as follows.

1. The host requests and is granted the DDC bus, reads video and audio capabilities from the EDID ROM, and starts sending video at a supported resolution. It also passes sink audio information to the source audio handler.
2. The host audio subsystem sends 2-channel uncompressed audio at 48 kHz and a 16-bit sample size, the minimum configuration required to be supported by all HDMI sinks. The audio logic passes this audio data through by default.

   Therefore, audio will be heard immediately at the sink device once the audio input type is selected.
3. The host audio application makes an appropriate audio mode choice. It writes registers:
   - TPI 0x26 with the Mute bit set
   - TPI 0x27 as required
   - TPI 0x26 again, this time with the Mute bit cleared and either S/PDIF or I^2S selected.
4. When it detects an unmute write to TPI 0x26 which will send Audio InfoFrames as required by HDMI.

Pixel Rate. Slower video pixel rates may not allow enough bandwidth to send compressed modes with high audio sample frequencies and high channel number counts. The only real example of this for CE applications is for 480p resolution, where 8-channel compressed audio is limited to a 48 kHz sample frequency.

Therefore, when programming the TPI Audio Configuration registers, the host audio handler may need to eliminate any configurations that are not compatible with the current video mode. Note that high-definition CE modes of 720p, 1080i, and 1080p have pixel rates high enough to leave no restrictions on the audio selection.

The S/PDIF input logic automatically compensates for slow pixel rates with fast S/PDIF stream clocks by increasing the sampling rate.

HDMI Audio InfoFrame Requirements for S/PDIF Operation

Audio InfoFrame requirements in the HDMI specification are handled automatically for S/PDIF. As long as the S/PDIF stream carries all header information, the HDMI specification allows most Audio InfoFrame information to be set to 0 (the *Refer to Stream Header* selection). The only required information not carried in the stream header is the Speaker Configuration byte of the InfoFrame, which must always be set according to the CEA-861-E standard when more than two speakers are used. TPI assumes that S/PDIF headers are being provided correctly by the audio host.

Configuring Audio using I^2S

The I^2S audio subsystem selection is made by TPI control.

I^2S Initialization and Operation

The transmitter requires the following initialization for I^2S applications.

1. Ensure that a valid I^2S audio stream is coming into the transmitter.
2. Select I^2S input mode using TPI 0x26[7:6], with Mute enabled (bit [4] = 1).
3. Write register TPI 0x20 to select the general incoming SD format.
4. Write register TPI 0x1F up to four times, to program each of the SD inputs.
5. Program register TPI 0x27[5:3] with the correct audio rate.
6. Program registers TPI 0x21–0x25 with the correct header information for the stream that will be sent over HDMI.
7. Write registers TPI 0xBF–0xCD with the appropriate Audio InfoFrame information.
8. Set the audio packet header layout indicator to 2-channel or multi-channel mode as needed using the sequence described below. Note that Audio InfoFrame byte 1 must also have this same setting.
9. Again write register TPI 0x26 with I^2S selected, this time with Mute disabled (bit [4] = 0).

I²S Input Configuration Register. Software configures the incoming SD format as shown in Table 16.

Table 16. Configuration of I²S Interface (RW)

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x20 [95]	I²S Input Configuration Register							
	SCK Sample Edge 0 – Falling (change data on rising edge) 1 – Rising	MCLK Multiplier – the Tx uses these bits to divide the MCLK input to produce CTS values according to the 128* f_s formula. The MCLK-to- f_s ratio is for input f_s, not down-sampled output f_s. 000 – 128 100 – 768 001 – 256 101 – 1024 010 – 384 110 – 1152 011 – 512 111 – 192			WS Polarity – Left when: 0 – WS is LOW 1 – WS is HIGH	SD Justify Data is justified: 0 – Left 1 – Right	SD Direction Byte shifted first: 0 – MSB 1 – LSB	WS to SD First Bit Shift? 0 – Yes (per spec) 1 – No

I²S Enable and Mapping. Software typically writes TPI 0x1F multiple times, with a separate FIFO selected each time, to assign SD pins to FIFOs. A single SD pin may be connected to multiple FIFOs. For example, the same SD0 pin could be assigned to FIFO#0, FIFO#1, FIFO#2, and FIFO#3 in order to provide 8 audio output channels. Unused FIFOs can be assigned to disabled SD inputs.

Note that no gaps are allowed when mapping channels to FIFOs – SD pins must be mapped to FIFO#0 and FIFO#1 before mapping a channel to FIFO#2, and so on.

Automatic Down-Sample. This feature provides sample rate conversion of stereo input to the Basic Audio rate that all HDMI sinks are required to accept; the specification allows this level of audio to be sent even before determining the device capabilities from EDID. The logic is only capable of converting a stereo signal, and cannot down-sample more than two channels. The down-sample logic output operates only on FIFO#0.

Table 17. Mapping of I²S Interface (RW)

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x1F [00]	I²S Enable and Mapping Register*							
	SD pin selected by [5:4] 0 – Disable 1 – Enable	RSVD	Select SD pin to be connected to a FIFO 00 – SD0 01 – SD1 10 – SD2 11 – SD3		Automatic down-sample to Basic Audio mode (FIFO#0 only) 0 – Disable 1 – Enable	Swap Left / Right I²S channels on this channel 0 – No swap 1 – Swap	This FIFO will take its input from the SD pin selected in bits [5:4] 00 – FIFO#0 01 – FIFO#1 10 – FIFO#2 11 – FIFO#3	

***Note:** Reads of this register return the last value written, so the value read back has meaning only for the channel most recently configured.

Header Layout Setting Sequence

Use TPI 0x26[5] to select the header layout. Once set, this bit setting will remain in effect and does not need to be rewritten on f_s change events.

Audio Word Length

When using the I²S interface, audio word length must be defined independently for the incoming I²S stream and outgoing HDMI stream.

Output Word Length. Whether for S/PDIF or I²S, the HDMI stream always accommodates a 24-bit audio sample word, regardless of whether all bits of the word are used. An indicator is provided in the stream header to identify the actual length of data to be used. For S/PDIF, this information is transferred automatically to HDMI. However, for I²S, it is necessary to program the length explicitly if it is less than the default value of 24.

Use TPI 0x25[3:0] to select the word length to be indicated to the receiver as shown in Table 18 on the next page.

Table 18. Stream Header Settings for I²S (RW)

Note that these registers are available **only** when TPI 0x26[7:6] = 10 to select I²S input.

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x21 [00]	I²S Channel Status Byte 0							
	cbit7	cbit6	cbit5	cbit4	cbit3	cbit2	cbit1	cbit0
0x22 [00]	I²S Channel Status Byte 1: Category Code							
	cbit15	cbit14	cbit13	cbit12	cbit11	cbit10	cbit9	cbit8
0x23 [00]	I²S Channel Status Byte 2: Source, Channel Number							
	I²S Channel Number cbit23:20				I²S Source Number cbit19:16			
0x24 [0F]	I²S Channel Status Byte 3: Accuracy, Sampling f_s							
	Clock Accuracy cbit31:28				Sampling Frequency cbit27:24 0000 – 44.1 kHz 1000 – 88.2 kHz 1100 – 176.4 kHz 0010 – 48 kHz 1010 – 96 kHz 1110 – 192 kHz 0011 – 32 kHz 0001 – not indicated			
0x25 [0B]	I²S Channel Status Byte 4: Original f_s, Sample Length							
	Original f_s, Channel Status bits 39:36 Refer to IEC60958 specification				Sample Length, Channel Status bits 35:32 Word Length 1011 – 24 bits (default) 1001 – 23 bits 0101 – 22 bits 1101 – 21 bits 1010 – 20 bits 1000 – 19 bits 0100 – 18 bits 1100 – 17 bits 0010 – 16 bits			

Input Word Length. Unlike S/PDIF, the incoming I²S audio stream has a variable word length. The output word length should be set to this same value or less, but never more.

The I²S sampling circuit defaults to 24-bit word length. The Input Word Length bits follow the same definition as the output word length bits: they are automatically set whenever the Output Word Length (TPI 0x25[3:0]) is written. To configure the circuit for sample word sizes less than 24 bits, follow the procedure listed below; it is necessary only if the Input and Output word length values are different.

HDMI Audio InfoFrame Requirements for I²S Operation

Audio InfoFrame requirements in the HDMI specification must be handled manually for I²S. All stream header information must be programmed by the host. The host must also program the Audio InfoFrame information.

HDMI allows for up to eight channels of audio content. Two channels pass through each of the four FIFOs. HDMI does not restrict the source to use any specific subset of the FIFOs. For example, 4-channel content can use many combinations of two FIFOs and two fields in the Audio InfoFrame packets (indicated by each packet's B.X and SP.X bits; see the *HDMI Specification*) limited only by the channel assignment choices in EIA/CEA-861-E Annex K.

The HDMI transmitter logic sets the B and PR bits automatically in each Audio InfoFrame packet, using both the I²S channel enables and the channel FIFO mapping. Some HDMI receiver chips do not make the arriving B and PR bit information accessible to the sink firmware. Therefore, the only indicator of *used audio channels* is in Data Byte 4 of the Audio InfoFrame packet.

1) `back modified value`

Configuring Audio for DSD Format

The Direct Stream Digital (DSD) Audio format, referred to as One-Bit Audio format in the HDMI specification, is supported using the I^2S clock and data inputs, S/PDIF input, and two additional pins for the 7^{th} and 8^{th} channels. Not all transmitters support DSD – for details, refer to the device data sheet.

Setting TPI 0x26[7:6] = 11 selects the DSD input format. When selected, the appropriate I^2S configuration register bits become control for the DSD logic. According to the number of channels, TPI 0x1F needs to be set up the same way as for I^2S input. Note that automatic down sampling of One-Bit Audio is not possible. Therefore, TPI 0x1F[3] needs to be set to zero.

The following is an example of how to set up DSD Audio for 5.1 Channel SACD transfer.

```
a) Write TPI 0x26=0xF0 // DSD, Layout 1, Mute
b) Write TPI 0x1F=0x80
c) Write TPI 0x1F=0x91
d) Write TPI 0x1F=0xA2
```

In addition to these settings, for standard 5.1-channel SACD, the following Audio InfoFrame fields need to be set:

- Coding Type (CT) → 0x00 (Refer to Stream Header)
- Channel Count (CC) → 0x05 (6 Channels)
- Sample Frequency (SF) → 0x02 (44.1 kHz).

Other settings are also possible depending on usage case.

As a final step, un-mute the audio inputs.

```
e) Write TPI 0x26=0xE0 // DSD, Layout 1
```

Interrupt Service

The TPI can be configured to generate an interrupt to the host to notify it of various events. The host can either poll for activity or use an interrupt handler routine. The TPI generates only a single interrupt signal (INT) to the host.

Interrupt Enable Register

Enables TPI and transmitter to generate interrupts to the host. Hot plug interrupts to the host will be generated even in D3 (low-power) state. Writing any bit to 1 enables the interrupt source, and also clears any pending interrupts. Note that writing 0 to disable the interrupt does not clear any previously pending interrupt.

Access. This register is accessed as a single byte.

Table 19. TPI Interrupt Enable (R/W)

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x3C [00]	HDCP Authentica-tion Status Change 0 – Disable 1 – Enable	HDCP V* Value ready 0 – Disable 1 – Enable	Security Status Change 0 – Disable 1 – Enable	Audio Error Event 0 – Disable 1 – Enable	CPI Event in place of Rx Sense 0 – Disable 1 – Enable	RSVD	Receiver Sense Event 0 – Disable 1 – Enable	Hot Plug / Connection (cable plugged/ unplugged) Event 0 – Disable 1 – Enable

Interrupt Status Register

Shows current status of interrupt events, even if the event has been disabled. This register can be polled for activity if the associated interrupt has been disabled. Write 1 to interrupt bits to clear the 'pending' status. Bits 3 and 2 serve only to show the current state and cannot be cleared.

Access. This register is accessed as a single byte.

Table 20. TPI Interrupt Status (R/W)

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x3D [00]	HDCP Authentica-tion status change Event pending 0 – No 1 – Yes	RSVD	Security Status Change Event pending 0 – No 1 – Yes	Audio Error Event pending 0 – No 1 – Yes	RxSense current state or CPI Event pending state (RO) 0 – No Rx sensed/CPI 1 – Powered Rx attached/ CPI event pending	Hot Plug pin current state or CTRL Bus Transaction Event pending state (RO) 0 – HP Low or no event pending 1 – HP High or CTRL event pending	Receiver Sense Event pending or CTRL Bus Error indicated 0 – No 1 – Yes	Hot Plug / Connection Event pending 0 – No 1 – Yes

Hot Plug / Connection. The Hot Plug *state* (HTPLG signal from the DVI or HDMI sink) indicates whether a display is attached (the EDID is readable, but the display is not necessarily powered up). The Hot Plug *event* indicates whether the state has changed.

Receiver Sense. The Receiver Sense *state* (RxSense signal from HDCP) indicates whether a powered-up receiver is sensed (whether the TMDS lines are being pulled externally to 3.3 V). The Receiver Sense *event* indicates whether the state has changed.

CPI Event. Even though the CPI register set is accessed at a completely separate I^2C slave address, CPI event pending status can optionally be reflected in the TPI Interrupt Status register in place of the Receiver Sense status bit. In this way, a single register read is adequate to check all interrupt sources at once. Setting TPI 0x3C[3] = 1 allows TPI 0x3D[3] to indicate a CPI Event Pending status. Note that the event itself must be cleared from the CPI registers (writing 1 to TPI 0x3D[3] has no effect).

To further clarify: Setting TPI 0x3C[3] = 1 simply allows the CPI interrupt pending status registers to reflect into TPI 0x3D[3]. The interrupt is caused by enabling its source in the CPI registers, not by setting TPI 0x3C[3] = 1; the interrupt would happen just the same if this bit were 0. It is only a convenient way to be able to see the pending CPI interrupt status from the main TPI interrupt register.

Audio Error. The Audio Error event indicates that an event related to the incoming S/PDIF audio stream has been detected and handled automatically. The most common event is a change in audio header f_s information. The hardware handles the event without a need for intervention, but the host can use this interrupt to read back the updated status information. No audio events related to I^2S input are reported.

Security Status Change. Any change in the Link Status value (TPI 0x29[5:4]) generates a Security Status Change event so that the host can take appropriate action to re-establish the link.

HDCP Authentication Status Change. An authentication status change event reflects changes in TPI 0x29[7:6], indicating that:

- The previous authentication request (from a write to the Protection Level bit) has completed successfully.
- The extended authentication process failed to complete within about 5 seconds.
- An Ri mismatch has caused authentication to fail.

Refer to the Extended Link Protection Level discussion in the HDCP Data Structure section on page 38 for details.

Interrupt Operation

In TPI register mode, this line is pre-set to operate as open-drain. To change the INT to either push-pull or open drain, the following logic can be applied during initialization while in TPI mode:

```
1. Write 0xBC=0x01                  // Internal page 0
2. Write 0xBD=0x79                  // Indexed register 0x79
3. Read   0xBE                      // Read current value
4. Modify bit[2] = INT_Output       // Change only the identified bit
                                    // bit 2 = 0, push-pull
                                    // bit 2 = 1, open-drain (default)
5. Write 0xBE                       // Write back modified value
```

In TPI register mode, this line is pre-set to operate as active low, which allows the external signal to be shared with other active low interrupt capable devices. To change the INT to active HIGH or active LOW, the following logic can be applied during initialization while in TPI mode:

```
1. Write 0xBC=0x01                  // Internal page 0
2. Write 0xBD=0x79                  // Indexed register 0x79
3. Read   0xBE                      // Read current value
4. Modify bit[1] = INT_Polarity     // Change only the identified bit
                                    // bit 1 = 0, active high
                                    // bit 1 = 1, active low (default)
5. Write 0xBE                       // Write back modified value
```

A number of interrupt functions can be optionally added to INT after startup. If the host enables interrupt sources by setting bits in TPI 0x3C, one or more events from an enabled source cause a pulse on the INT line. The host will use the signal to recognize and service the interrupt. This sequence is illustrated in Figure 6 on the next page.

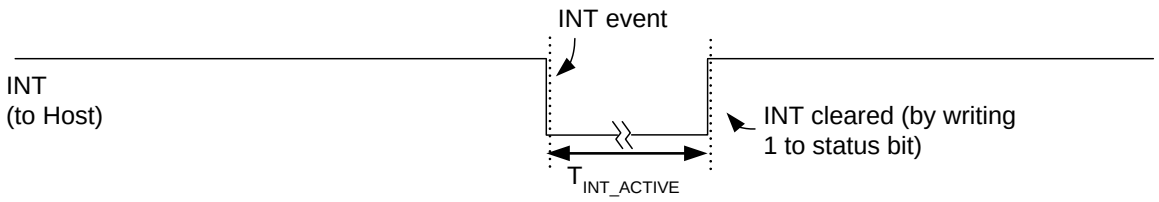

Figure 6. INT Pulse on Event other than HTPLG

When the host clears the interrupt event (by writing 1 to the triggering Interrupt Status Register bit), the INT line will go HIGH, as long as there are no more active events. T_{INT_ACTIVE} duration depends on the response time of the host and firmware.

Before leaving the service routine, the host should poll for additional events that could have occurred after the host completed the initial service.

To use the interrupt in D3 power state, it is required to perform a hardware reset before entering the power down mode and HPD must be low or Cable must not be connected prior to entering D3 Hot or D3 Cold mode for wake-up to be valid.

TPI System

The transmitter subsystem has additional control necessary from the firmware for the following:

There may be times it is necessary to know whether the TMDS clock is stable and the transmitter can send reliable data on the TMDS link before the HDCP link integrity check is completed. This can be accomplished by monitoring the TCLK STABLE interrupt, bit 1 of indexed register 0x72. If TCLK STABLE has changed logic level, indicating a change (from stable to non-stable or vice-versa) in the stability of the clock going to TMDS, bit 1 of indexed register 0x72 will be asserted. Write *1* to clear this interrupt.

Monitor TCLK STABLE interrupt bit 1 of indexed register 0x72.

```
1. Write 0xBC=0x01          // Internal page 0
2. Write 0xBD=0x72          // Indexed register 0x72
3. Read 0xBE                // Read current value
4. Check bit[1] = 1         // Indicates a change in the clock
5. Write 0xBE bit[1] = 1    // Clear the clock change indicator
```

Power State Control and Hot Plug Management

The transmitter subsystem requires control of power state after a hardware reset, following by ongoing management of plug events to determine when the subsystem should be put back into a low-power mode. The Tx defaults to device power state D2 after reset.

Transmitter Power State Register

TPI supports ACPI power states D0, D2, and D3. Transition from D0 to any other state requires only a write to the power state register TPI 0x1E[1:0] bits.

When the host writes TPI 0x1E[1:0] after reset, the subsystem operates as follows.

→ 03 (D3):

1. An RSEN or Hot Plug interrupt event bit in TPI 0x3D[1:0] must be set prior to entry to D3 if a wakeup interrupt is to be enabled.
    a. A full hardware reset is required to wake the system up from D3 Hot or D3 Cold mode.
    b. HPD must be LOW or a cable must not be connected prior to entering D3
    c. D3 Cold: Analog core is disabled, only HPD is working to generate INT.
    D3 Hot: Analog core is not fully disabled, HPD and RSEN is working.
    Both have no access to I^2C to clear INT until a RESET is asserted.

RSEN	Hot Plug
D3 Hot	D3 Hot
	D3 Cold

    For D3 Cold:
        i. Disconnect Cable
        ii. Clear any pending interrupts via TPI 0x3D
        iii. Reset Tx via HW
        iv. Write device address 0x72, register offset 0xC7 = 0x00 to enter TPI mode
        v. Set INT# source to Hotplug via TPI 0x3C[0] = 1b
        vi. Clear any pending interrupts via TPI 0x3D
        vii. Enter D3 Cold mode via TPI 0x1E[1:0] = 11b
    d. For D3 Hot
        i. Cable can be disconnected or left connected.
        ii. Set INT# source to either RSEN or HPD
        iii. Clear any pending interrupts via TPI 0x3D
        iv. Enable D3 Hot mode via register 0x1E[1:0] = 11b
        v. Go back to D3 Hot or enter D3 Cold if cable is unconnected or wake up if cable is connected based on cable setting before entering D3 Hot.
2. The transmitter is brought to its lowest-power state.
3. The host can no longer read any registers.
4. If a wakeup interrupt bit was set before entering D3 state, a plug-in event will cause INT to be asserted and stay there (it cannot be cleared since register access is disabled).
5. The host must assert RESET# to the chip to leave D3 state. Reset is **required** for proper return from D3.

→ 02 (D2):

1. The transmitter is brought to a low-power state, but not quite as low as D3 state.
2. Host can read registers.
3. A plug-in event will cause INT to be asserted if enabled.

4. The host must then program the system back to D0 state in order to restore full operation, and must clear the host plug interrupt in TPI 0x3D to allow INT to be deasserted.

→ 00 (D0):

1. The transmitter is brought to full operation. TMDS output will be enabled according to TPI 0x1A settings.
2. Writing D1 has the same effect as writing D0.

D0→D2 Transition

1. The D2 state powers down the transmitter (TMDS output is disabled), but leaves the registers operational. All register settings remain unchanged.

D2→D0 Transition

1. All register settings remain unchanged except for TPI 0x1A, which disables TMDS output.

Hot vs Cold Wakeup Modes

Enabling best power savings requires disabling one of the wakeup features.

- Hot Plug Detect (HDP from the HDMI connector) is always available as a wakeup source in D2 and D3 states.
- Receiver Sense (RSEN, monitoring the TMDS lines for active pullups at the receiver side) is only available as a wakeup source when TPI 0x1E[2] = 0.

TPI 0x1E[2] is effective only for D3 state. It defaults to a setting of 0, for which the D3 description above is valid, except that the power consumption is not yet at its absolute lowest. This is known as D3 hot mode, in which HPD and RSEN wakeup events can be monitored.

To enter the lowest power mode, the host must write TPI 0x1E[2] = 1 to select D3 cold mode *prior* to writing TPI 0x1E[1:0] = 11. That is, the mode must already be set before selecting D3 mode. In D3cold state, the chip will go to its absolute lowest power state. However, it will ignore HPD and RSEN wakeup events regardless of other programming.

To prevent HPD and RSEN from clearing when writing a new value to TPI 0x1A[4], change at least one other bit at 0x1A [2], 0x1A[3], or 0x1A[6].

Access. This register is accessed as a single byte.

Table 21. TPI Device Power State Control Data (R/W)

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x1E [02]	RSVD					Wakeup state 0 – Hot 1 – Cold	Tx Power_ state[1]	Tx Power_ state[0]

Hot Plug Management

To properly handle plug and unplug events, the host must implement the Hot Plug Service Loop.

Hot Plug Service Loop

After the host releases Reset to the HDMI transmitter subsystem and initializes the part, the host program typically enters a Hot Plug Service Loop. The transmitter subsystem logic monitors for Hot Plug activity, but will not attempt detection of the attached device until the host program takes specific action.

During this time, the following registers are of importance.

- Request/Grant register TPI 0x1A. Any time the host wants to read EDID, it must follow the correct procedure for requesting access to the DDC bus.
- Interrupt Enable TPI 0x3C. If desired, interrupt-driven operation can be enabled through TPI 0x3C. Otherwise polling is used. The bits in TPI 0x3C serve only to enable corresponding active interrupt event bits in TPI 0x3D to cause the INT line to go low.
- Interrupt Status TPI 0x3D. Whether polling or interrupt driven mode is used, the host uses TPI 0x3D to monitor and service all hot plug activity.

The host can poll the Interrupt Status register to watch for EDID availability (bit 2 = 1) and monitor powered-on status (bit 3 = 1), or can enable interrupts for hot plug events.

Suggested Service Flow. When the host either receives an interrupt or polls TPI 0x3D and detects Hot Plug activity with TPI 0x3D[0] = 1, it should follow this procedure.

1. Clear the service event. Write TPI 0x3D[1:0] = 11 to clear the event.
2. Read and record the active pin states. TPI 0x3D[3:2] provide active state information for the HPD and RSEN signal levels.
3. Determine the event that occurred.
   - If the prior state was plugged-in, an Unplug event has occurred. An interrupt is registered **immediately** in TPI 0x3D[1:0] for an Unplug event. TPI 0x3D[3:2] = 00 confirms an Unplug event; any other state indicates instability in the connection (which should be monitored until it becomes stable before proceeding).
   - If TPI 0x3D[2] = 1, a Plug event has occurred and has likely stabilized. Interrupts due to Plug events are **delayed** until after the HPD signal has stabilized at a High level for ~500ms. Host software can choose to provide additional de-bouncing if desired (See Hot Plug Delay De-bounce).
4. Take action as needed.
   - For a confirmed Plug event, the host should use the Request / Grant protocol to take control of the DDC bus and read EDID as described in the Detailed Sequences section.
   - For an Unplug event, the host should stop sending all video. This step also ensures that no secure video content is being sent.

Operating Sequences

Some of the major events involved in setting up and running the chip are illustrated below.

Reset. The host is required to reset the transmitter subsystem at power-up time. The reset pulse width T_{RST} should meet the minimum requirement of each device. Please refer to the respective Transmitter Data Sheet of the device used to determine the minimum value. **DDC Access by HDCP Subsystem.** When the host selects Operational (D0) state, the chip reserves the DDC bus for immediate host use. As always, the host must use the request/grant mechanism to get the bus. However, only after it has finished and releases the request will the HDCP subsystem look for an HDCP-capable sink on the DDC bus.

In all future uses of the bus, the host must rely on the Request/Grant register bits to use the DDC bus, preventing conflicts with the HDCP logic.

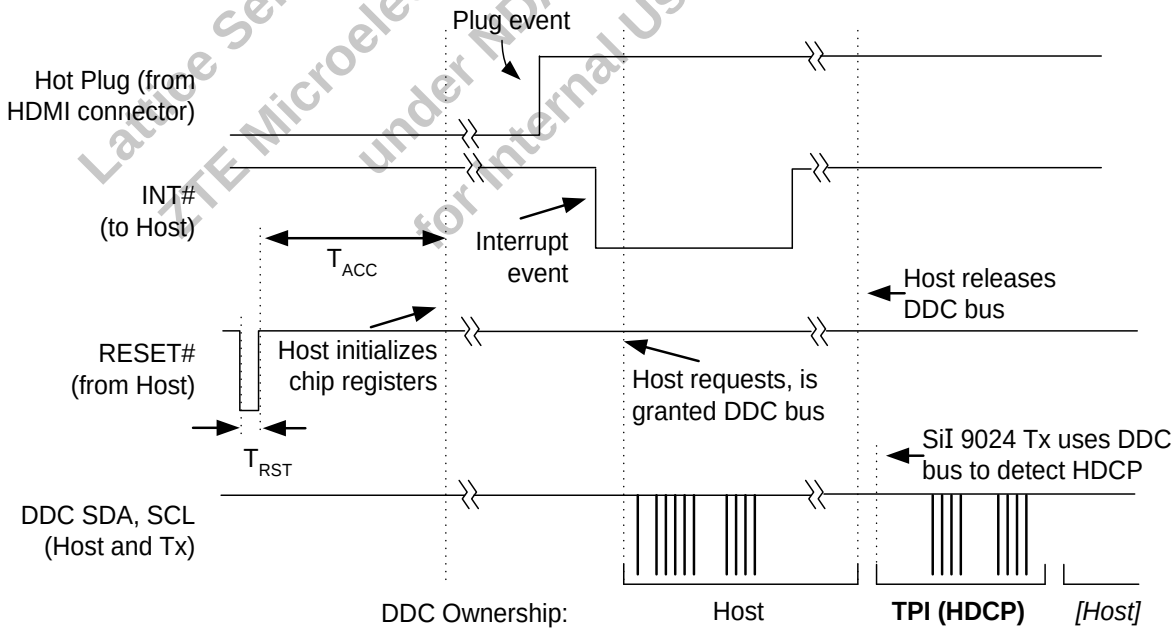

Figure 7. Setup Events

Table 22. Hot Plug Timing Requirements

Parameter	Description	Min	Max	Unit
T_{RST}	Reset pulse width	50	—	µs
T_{ACC}	Time from reset until host can access chip registers	—	1	ms

Hot Unplug and Plug-in. When the sink device is unplugged, or if the sink device momentarily toggles its HTPLG line low, the hardware immediately returns a Link Lost message when queried by way of TPI. If the interrupt is enabled, the INT line is also asserted until the host clears the interrupt.

The host should check register TPI 0x3D[2] to determine whether a hot plug or hot unplug event is being signaled. If hot plug, the host should continue checking for a minimum of T_{RST_DLY} before recognizing the attachment state, to ensure that the HDMI plug attachment is stable.

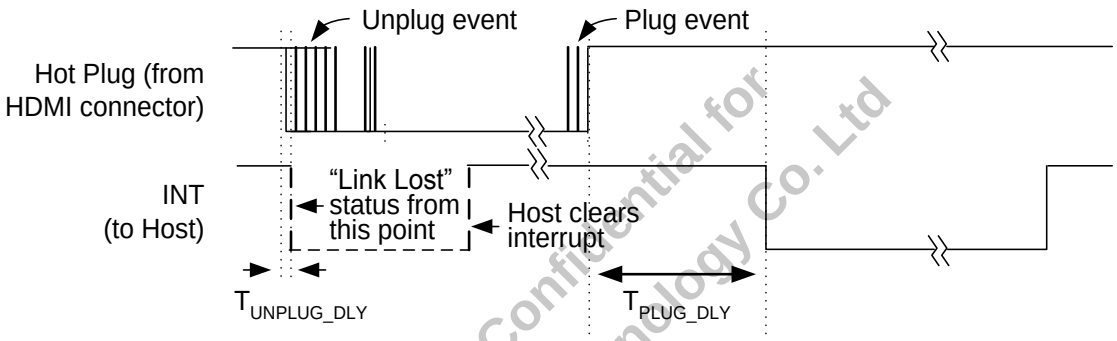

Figure 8. Hot Plug and Interrupt Timing

Table 23. HPD to INT Delays

Parameter	Description	Min	Typ	Max	Unit
T_{PLUG_DLY}	Delay from Hot Plug (HPD) HIGH and stable to INT active (programmable, see the De-bounce: section below)	400	480	600	ms
T_{UNPLUG_DLY}	Delay from Hot Plug (HPD) LOW to INT active (not programmable)	—	—	1	ms

Hot Plug Delay De-bounce

The hot plug delay debounce is programmable in steps of 3.4 ms. The logic needs to be manually configured as follows:

```
1. Write 0xBC=0x01          // Internal page 0Write 0xBD=0x7A          //
   Indexed register 0x7A
2. Write 0xBE=0x13          // Program a delay of approximately 64ms
```

HDCP Implementation

HDCP link security logic is implemented in certain transmitters; unique keys are embedded in each chip as part of the solution. The security scheme is fully automatic and is handled completely by hardware.

DEVICE COUNT = 0 condition in BSTATUS of the HDCP port must be checked by enabling bit 3 of indexed register 0x0A.

```
1) Write 0xBC=0x01          // Internal page 0
2) Write 0xBD=0x0A          // Indexed register 0x0A
3) Read 0xBE                // Read current value
4) Modify bit[3] = 1        // Change only the identified bit
5) Write 0xBE               // Write back modified value
```

Ri check must be enabled while in embedded mode via bit 0 of indexed register 0x0A.

```
1) Write 0xBC=0x01          // Internal page 0
2) Write 0xBD=0x0A          // Indexed register 0x0A
3) Read 0xBE                // Read current value
4) Modify bit[0] = 1        // Change only the identified bit
5) Write 0xBE               // Write back modified value
```

Control for HDCP

The host uses standard data structure elements to select the correct level of high definition content protection.

Table 24. Mapping of HDCP Data to TPI Register Bits

Name	R/W	How to Derive from TPI Registers
ProtLevel Min (0) Max (1)	W	The host writes this value to request that link security (HDCP) be enabled (1) or disabled (0).
ProtType None (0) HDCP (1)	R	The host reads this value to determine whether HDCP is available on the attached monitor. The transmitter logic automatically detects whether the sink is HDCP-capable after a hot-plug event. The host must have requested and released the DDC bus at least once to trigger the automatic detection.
Flags Status Normal (0x00) Link Lost (0x01) Renegotiation Required (0x02)	R	The host polls this value, or can be interrupted when this value changes. Status Normal: The data link is operating according to the requested Protection Level. Link Lost: The physical link has been interrupted (through cable detachment or error). Renegotiation Required: During HDCP operation, an Ri mismatch has occurred requiring the host to re-request security.
ConnectorType DVI (0) HDMI (1)	R	The host reads this value to determine whether the attached sink device is DVI or HDMI. It reflects the value written to TPI 0x1A[0].
ProtectionLevel (Local/Extended) No Link Protection (0) Link Secure (1)	R	The host polls this value, or can be interrupted when the value changes. Local Link: The link from the transmitter to the attached sink, whether display or repeater. Extended Link: The link downstream from the attached repeater. If no repeater is attached, this value will always be 0.
HDCPFlags HDCPRepeater (0x01)	R	The host reads this value to determine whether the device is an HDCP repeater.
BKey Bksv from attached device.	R	The host requests BKey to determine whether the HDCP keys are valid.

HDCP Operation

Once a connection is made, video is being transmitted, and link video is stable, the host can request link security through TPI 0x2A[0]. All subsequent authentication is performed automatically, and the resulting security level is reflected in TPI 0x29[7:6]. Typically the process takes under 1 second.

Once the link is authenticated, periodic link integrity checks take place every 128 frames. The time interval between checks is therefore roughly 2 seconds, depending on whether 60 Hz or 50 Hz video is being supplied. Note that some resolutions run at lower frame rates (such as 24 Hz) will create a corresponding slow-down in the frequency of the periodic checking. It is possible to enable intermediate Ri reading TPI 0x2A[3] to satisfy the 2 s requirement for Ri checking.

Because any lengthy host use of the bus could conflict with HDCP link checks that occurs every 2 s, Silicon Image recommends that the host clear its link security request (TPI 0x2A) whenever it wants to access the DDC bus.

HDCP Data Structure

The registers below reflect the information needed for the security interface. Use of these bits is detailed on the previous page. Additional information is provided below.

Link Status. These bits operate as follows; a status change can cause an interrupt if enabled.

> **Normal:** The link is operating correctly; valid whether HDCP is running or not.
>
> **Link Lost:** A hot plug or other event has caused the link to go down. A reset may be necessary to restore operation, if the logic is unable to return the connected link to Normal condition.
>
> **Renegotiation required:** The link is still active, but some event (usually Ri mismatch) has caused HDCP to fail. The host must disable and then re-enabled the security request (TPI 0x2A[0]).
>
> **Link Encryption Suspended:** The link is still active, but encryption has been turned off by clearing TPI 0x2A[0]. This condition will return to Normal once TPI 0x2A[0] is set to 1 again.

The HDCP subsystem continues to keep the link authenticated (doing Ri checking) even after TPI 0x2A[0] is switched from 1 to 0, so that subsequent security requests can return a secure status more quickly.

Extended Link Protection Level. TPI 0x29[7] indicates that the repeater authentication is complete. The bit should be ignored if the HDCP Repeater bit is not = 1.

> → **Link Integrity Mode.** If a downstream HDCP repeater is discovered, Link Integrity Mode bit TPI 0x1A[6] should be set to 1 (Dynamic). This setting forces the logic to re-authenticate any time the incoming clock resolution changes. This same setting will work even when the downstream sink is not a repeater; however, authentication may take slightly longer.

Repeater authentication takes place automatically after local link protection is established. The possible outcomes are:

- **Success:** Within 5 s, TPI 0x3D[7] → 1 to indicate the change in security status. Firmware then checks that TPI 0x29[7] = 1, indicating that the extended link is now secure.
- **Failure:** After approximately 5 s, if the extended link cannot be authenticated (typically due to lack of Ready status by the downstream HDCP repeater), TPI 0x3D[7] → 1 to indicate that authentication could not take place. TPI 0x29[7] remains at 0.

Using this mechanism, firmware can avoid implementing a timer – the logic design guarantees that TPI 0x3D[7] will be triggered as soon as the outcome is known.

> → **Important Note:** AV Mute must not be set before requesting authentication. If it is set, there will be no indication in TPI 0x3D[7] of the status outcome.

Protection Type. This bit indicates whether the automatic logic has detected an attached HDCP-capable device. The TPI subsystem will not attempt to detect an HDCP receiver until the host has requested and then released the DDC bus once after hot plug, and then turned on TMDS output to send valid clocks. Therefore, even if the host does not read the EDID immediately, it should still request and then release the DDC bus to trigger detection.

Access. All registers can be read either individually or in a burst.

The correct sequence for enabling or disabling HDCP is:

- Poll TPI 0x29[1] until it indicates that HDCP is available; the Bksv values are only valid when TPI 0x29[1] = 1.
- Write TPI 0x2A[0] with the desired setting.
- Poll TPI 0x2A[0] until it matches the setting just made.
- Poll TPI 0x29[7:6] until they indicate the desired security level.

Note that TPI 0x29 may not be updated immediately after a cable connect event, due to the delay (max 1s) introduced by the automatic de-bounce logic. If desired, Interrupt Status register TPI 0x3D bits 2 and 0 can be read to verify the connection before reading TPI 0x29.

Table 25. TPI Security Registers

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x29	QueryData (RO)							
	Extended Link Protection Level (across repeaters) 0 – No link protection 1 – Link secure	Local Link Protection Level 0 – No link protection 1 – Link secure	Link Status 00 – Normal 01 – Link Lost 10 – Renegotiation required 11 – Link Encryption Suspended		HDCP Repeater 0 – No 1 – Yes	Connector Type Bit 2 + bit 0 work together: 0+0 – DVI 1+0 – HDMI 0+1 – Rsvd 1+1 – Future	Protection Type 0 – None 1 – HDCP	Connector Type Extension See bit 2
0x2A [00]	ControlData (RW)							
	RSVD		encryption disable 0 – encryption enabled if authentication succeeded 1 – no HDCP encryption (even after authentication)	RSVD				Protection Level 0 – Min (no protection needed) 1 – Max (HDCP required)
0x2B	BKSV_1(RO)							
0x2C	BKSV_2(RO)							
0x2D	BKSV_3(RO)							
0x2E	BKSV_4(RO)							
0x2F	BKSV_5(RO)							

Important Notes

The automatic HDCP module expects firmware to follow a certain logical order in order to correctly sequence the HDCP events. Important considerations for this sequencing are noted below.

- **Initial Request/Grant Requirement.** On a hot plug event, host firmware requests and is granted the DDC bus for reading EDID. The bus must be requested, granted, and released at least once after any hot plug event (through TPI 0x1A). This action releases the bus from an initial hold state, allowing internal logic to access it as needed.
- **Triggering the Bksv/Bcaps Read.** After the host has read the EDID and determined an acceptable resolution, it starts sending video to the transmitter input. Once this video is stable, the host enables the TMDS output from the transmitter (again through TPI 0x1A). Turning on TMDS output triggers the logic to attempt to discover downstream HDCP devices, which require a TMDS clock before they become accessible. Only then can the Tx read Bcaps and Bksv.

- **Causes of Renegotiation Required.** Once the host requests link security through TPI 0x2A[0]=1, the DDC bus is tied up by the 1st, 2nd, and 3rd parts of the authentication process described in the HDCP specification. After the 1st part is complete, TPI 0x29[6] = 1. After the 2nd part is complete, TPI 0x29[7] = 1. The 3rd part is a recurring event, repeating every 128 frames. Interruption of any of these phases will result in TPI 0x29[5:4] = 11, indicating Renegotiation Required. This would likely happen if an intentional external hack were taking place on the DDC bus to circumvent HDCP.
- **Causes of Link Encryption Suspended.** Even when HDCP encryption is turned off through TPI 0x2A[0] = 0, the logic will attempt to keep the link authenticated through periodic Ri checks. If no other DDC activity has interfered with an Ri check event, turning HDCP encryption back on will take effect immediately and the link will still be considered secure (no new authentication needed). But if some device takes over the DDC bus and prevents a periodic Ri check, the result is TPI 0x29[6:5] = 11, Link Encryption Suspended.
- **KSV List Acquisition in Source Applications.** When the transmitter is used in an HDMI source system, the host uses a normal request/grant procedure to obtain control of the DDC bus, and then reads the downstream KSV lists for comparison against keys in the SRM. HDCP encryption and link integrity checking can be left enabled; a bus grant will arrive within 2 s of the request, and the host can then retrieve and compare the KSV list.
- **KSV List Acquisition in Repeater Applications.** When the transmitter is used in an HDMI repeater system, the HDMI CTS restricts the amount of time allowed for loading the downstream KSV list into the upstream receiver FIFO and setting the Ready bit. TPI development firmware supplied by Silicon Image includes special shortcut routines for quickly and safely forcing a grant of the DDC bus, allowing the KSV list to be quickly acquired and loaded to the upstream receiver FIFO. Refer to the supplied firmware for details on this and other repeater-related topics.

Auxiliary HDCP Registers

TPI supports HDCP repeater authentication and revocation through the simple interface described below.

HDCP Info Register

Identifies the support for HDCP revision 1.2. If this byte reads 00, HDCP is not supported in this device. However, if the byte is not zero, HDCP may still not be supported. Software must read and verify Aksv as described on the next page (TPI 0x36-3A) to verify HDCP capability.

Access. This register is accessed as a single byte.

Table 26. TPI HDCP Revision Data (RO)

Offset	Bits	Bit Field Description	Return value
0x30	[3:0]	HDCP Minor revision	0x02
	[7:4]	HDCP Major revision	0x01

V* Value Select Register

V value* is the Silicon Image designation for the KSV values it has hashed together with the transmitter M0 value. The host reads the V* value and compares it with its own calculated value to verify a match.

Write: Allows the host to select V*0 to V*4 for readback at TPI 0x32–0x35.

Read: The microcontroller needs some time to load the selected V* values for reading. After the host writes this register to select the set of V* values to be read, it must poll the 'ready' bit until it reads = 1 indicating that the V* values are ready for reading.

Authentication State. This value returns the status of an internal state machine. It should **not** be used by the host for normal activities, as its timings depend on the system configuration. However, it can be convenient for use in HDCP debugging in the case of repeater authentication, where it will stay in the '11' state until all downstream devices have reported back as "ready".

Downstream HDCP Device Count. The transmitter supports a maximum of 16 downstream devices. This bit indicates whether this limit has been exceeded. It is only valid when the repeater authentication process has completed.

Access. This register is accessed as a single byte.

Table 27. TPI KSV and V* Value Data (R/W)

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x31 [00]	Authentication State (RO) 00 – Auth. required 01 – Re-auth. required 10 – Authenticated 11 – Repeater auth. required		RSVD for internal use	Downstream HDCP Device Count (RO) 0 – Within Tx limit 1 – Exceeds Tx limit	Selected V* Values Ready? (RO) 0 – No 1 – Yes	V* Value Select 000 – H0 001 – H1 010 – H2 011 – H3 100 – H4 101 – Rsvd 11x – Rsvd		

V* Value Readback Registers

The host reads back the 32-bit value for the selected V* value from these registers.

Access. Registers 0x32–0x35 must be accessed as a complete group using a burst read.

Table 28. TPI V* Value Readback Data (RO)

Offset	Bits	Bit Field Description
0x32	[7:0]	V* Hx bits [7:0]
0x33	[7:0]	V* Hx bits [15:8]
0x34	[7:0]	V* Hx bits [23:16]
0x35	[7:0]	V* Hx bits [31:24]

Aksv Value Readback Registers

The host reads back the 40-bit value for the transmitter Aksv from these registers. It must check the Aksv for 20 1s and 20 0s to verify that HDCP capability is available. A 9022A Tx will typically return the values 0x09, 0x00, 0x02, 0x02, 0x0A from these registers, indicating that it is not HDCP-capable.

Access. Registers 0x36–0x3A must be accessed as a complete group using a burst read.

Table 29. TPI Aksv Readback Data (RO)

Offset	Bits	Bit Field Description
0x36	[7:0]	AKSV_1
0x37	[7:0]	AKSV_2
0x38	[7:0]	AKSV_3
0x39	[7:0]	AKSV_4
0x3A	[7:0]	AKSV_5

Disclaimers

These materials are provided on an “AS IS” basis. Silicon Image, Inc. and its affiliates disclaim all representations and warranties (express, implied, statutory or otherwise), including but not limited to: (i) all implied warranties of merchantability, fitness for a particular purpose, and/or non-infringement of third party rights; (ii) all warranties arising out of course-of-dealing, usage, and/or trade; and (iii) all warranties that the information or results provided in, or that may be obtained from use of, the materials are accurate, reliable, complete, up-to-date, or produce specific outcomes. Silicon Image, Inc. and its affiliates assume no liability or responsibility for any errors or omissions in these materials, makes no commitment or warranty to correct any such errors or omissions or update or keep current the information contained in these materials, and expressly disclaims all direct, indirect, special, incidental, consequential, reliance and punitive damages, including WITHOUT LIMITATION any loss of profits arising out of your access to, use or interpretation of, or actions taken or not taken based on the content of these materials.

Silicon Image, Inc. and its affiliates reserve the right, without notice, to periodically modify the information in these materials, and to add to, delete, and/or change any of this information.

Notwithstanding the foregoing, these materials shall not, in the absence of authorization under U.S. and local law and regulations, as required, be used by or exported or re-exported to (i) any U.S. sanctioned or embargoed country, or to nationals or residents of such countries; or (ii) any person, entity, organization or other party identified on the U.S. Department of Commerce's Denied Persons or Entity List, the U.S. Department of Treasury's Specially Designated Nationals or Blocked Persons List, or the Department of State's Debarred Parties List, as published and revised from time to time; (iii) any party engaged in nuclear, chemical/biological weapons or missile proliferation activities; or (iv) any party for use in the design, development, or production of rocket systems or unmanned air vehicles.

Products and Services

The products and services described in these materials, and any other information, services, designs, know-how and/or products provided by Silicon Image, Inc. and/or its affiliates are provided on as “AS IS” basis, except to the extent that Silicon Image, Inc. and/or its affiliates provides an applicable written limited warranty in its standard form license agreements, standard Terms and Conditions of Sale and Service or its other applicable standard form agreements, in which case such limited warranty shall apply and shall govern in lieu of all other warranties (express, statutory, or implied). EXCEPT FOR SUCH LIMITED WARRANTY, SILICON IMAGE, INC. AND ITS AFFILIATES DISCLAIM ALL REPRESENTATIONS AND WARRANTIES (EXPRESS, IMPLIED, STATUTORY OR OTHERWISE), REGARDING THE INFORMATION, SERVICES, DESIGNS, KNOW-HOW AND PRODUCTS PROVIDED BY SILICON IMAGE, INC. AND/OR ITS AFFILIATES, INCLUDING BUT NOT LIMITED TO, ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND/OR NON-INFRINGEMENT OF THIRD PARTY RIGHTS. YOU ACKNOWLEDGE AND AGREE THAT SUCH INFORMATION, SERVICES, DESIGNS, KNOW-HOW AND PRODUCTS HAVE NOT BEEN DESIGNED, TESTED, OR MANUFACTURED FOR USE OR RESALE IN SYSTEMS WHERE THE FAILURE, MALFUNCTION, OR ANY INACCURACY OF THESE ITEMS CARRIES A RISK OF DEATH OR SERIOUS BODILY INJURY, INCLUDING, BUT NOT LIMITED TO, USE IN NUCLEAR FACILITIES, AIRCRAFT NAVIGATION OR COMMUNICATION, EMERGENCY SYSTEMS, OR OTHER SYSTEMS WITH A SIMILAR DEGREE OF POTENTIAL HAZARD. NO PERSON IS AUTHORIZED TO MAKE ANY OTHER WARRANTY OR REPRESENTATION CONCERNING THE PERFORMANCE OF THE INFORMATION, PRODUCTS, KNOW-HOW, DESIGNS OR SERVICES OTHER THAN AS PROVIDED IN THESE TERMS AND CONDITIONS.

1140 E. Arques Avenue
Sunnyvale, CA 94085
T 408.616.4000 F 408.830.9530
www.siliconimage.com

1140 E. Arques Avenue
Sunnyvale, CA 94085
T 408.616.4000 F 408.830.9530
www.siliconimage.com