LatticeECP3 HDMI/DVI Interface

Reference Design

FPGA-RD-02139-1.6

January 2020

Disclaimers

Lattice makes no warranty, representation, or guarantee regarding the accuracy of information contained in this document or the suitability of its products for any particular purpose. All information herein is provided AS IS and with all faults, and all risk associated with such information is entirely with Buyer. Buyer shall not rely on any data and performance specifications or parameters provided herein. Products sold by Lattice have been subject to limited testing and it is the Buyer's responsibility to independently determine the suitability of any products and to test and verify the same. No Lattice products should be used in conjunction with mission- or safety-critical or any other application in which the failure of Lattice's product could create a situation where personal injury, death, severe property or environmental damage may occur. The information provided in this document is proprietary to Lattice Semiconductor, and Lattice reserves the right to make any changes to the information in this document or to any products at any time without notice.

Contents

1. Introduction 5
2. Features 5
3. HDMI/DVI Overview 5
3.1. What is DVI? 5
3.2. What is HDMI? 5
4. What is TMDS? 6
4.1. Using the CML SERDES to Interface to TMDS 6
4.2. HDMI/DVI Link Topology 7
4.3. HDMI/DVI Coding Scheme 8
5. Functional Descriptions 11
5.1. HDMI/DVI Receiver 11
5.2. PCS Word Aligner 13
5.3. Link Controller 14
5.4. PCS Downsample FIFO 14
5.5. Multi-Channel Alignment 15
5.6. Decoder 16
5.7. HDMI/DVI Transmitter 18
5.8. Encoder 20
6. SERDES/PCS Configuration 20
6.1. Miscellaneous Functional Blocks 24
7. Hardware Validation 25
7.1. Evaluation Board 25
8. Reveal On-chip Logic Analysis 26
9. Hardware Test Results 26
10. Design Limitations and Disclaimers 27
11. Implementation 28
References 29
Technical Support Assistance 30
Revision History 31

Figures

Figure 4.1. TMDS Differential Link 6
Figure 4.2. AC Coupling Circuit 6
Figure 4.3. Single Link HDMI/DVI 8
Figure 4.4. DVI Periods 8
Figure 4.5. HDMI Periods 9
Figure 4.6. Video Data Coding Scheme 10
Figure 4.7. Control Period Coding Scheme 10
Figure 4.8. TERC4 Coding Scheme 11
Figure 5.1. HDMI/DVI Receiver and SERDES/PCS Block 12
Figure 5.2. Programmable Comma Character and Mask Registers 14
Figure 5.3. Clock Scheme Using tx_full_clk 15
Figure 5.4. Multi-Channel Alignment Algorithm 16
Figure 5.5. HDMI/DVI Decoder Block Diagram 16
Figure 5.6. DVI and HDMI Decoding Scheme 17
Figure 5.7. HDMI/DVI Transmitter and SERDES/PCS Block 18
Figure 5.8. HDMI/DVI Encoder 20
Figure 6.1. Channel_Enable and Mode_Selection 21
Figure 6.2. Rx/Tx Data_Rate and Related Reference_Clock 21

Figure 6.3. AC Coupling 22
Figure 6.4. Word_Alignment....................23
Figure 6.5. Control_Setup for ispLever 8.1 SP1 24
Figure 6.6. Control_Setup for Diamond 1.1 24
Figure 7.1. Hardware Demo Setup, Top View....................25
Figure 8.1. Reveal Waveform Showing Two Blanking Periods....................26
Figure 8.2. Enlarged Blanking Period with Data Island Packet 26

Tables

Table 4.1. Electrical Characteristics Comparison 7
Table 5.1. HDMI Receiver Interface 12
Table 5.2. DVI Receiver Interface....................13
Table 5.3. HDMI Transmitter Interface 19
Table 5.4. DVI Transmitter Interface 19
Table 9.1. HDMI (with Audio) Validation Results....................27
Table 9.2. DVI Validation Results 27
Table 11.1. Performance and Resource Utilization 28

1. Introduction

As the display industry has evolved rapidly in recent years, with the analog CRT being replaced by the digital flatpanel display, DVI and HDMI have emerged as the dominate standards for connecting digital display devices like a PC monitor and HDTV. DVI only carries uncompressed video data, whereas HDMI can transfer both uncompressed digital video and multi-channel audio over a single cable. Developed specifically to address the unique requirements of the consumer electronics market, HDMI offers additional enhancements including the support of the YCbCr color space format, and the universal remote control capability through Consumer Electronics Control (CEC) protocol. HDMI-enabled display devices are backward-compatible with DVI-based PCs so users can display the PC contents on their HDTV.

This document describes the implementation of HDMI and DVI Physical Layer interfaces using the low-power, highperformance SERDES channels featured on the LatticeECP3™ FPGA device. There are two separate reference designs implemented for demonstration, one features the HDMI interface, another features the DVI interface. Both reference designs have been validated using the LatticeECP3 Video Protocol Board and HDMI Mezzanine Card. The hardware configuration and demonstration procedures are not in the scope of this document. Refer to LatticeECP3 HDMI/DVI Loopback Demo User’s Guide (UG36) and LatticeECP3 DVI/7:1 LVDS Video Conversion Demo User’s Guide (UG37) for more details of demo procedures and instructions.

2. Features

- SERDES-based DVI and HDMI interface supports data rate up to 1.65 Gbps
(Note: The maximum data rate is limited by the operational range of the test equipment)
- Supports 48 kHz PCM linear mode audio samples built in HDMI data
- Shared TMDS Encoder and Decoder for both DVI and HDMI data
- Supports arbitrary HSYNC/VSYNC polarity
- PCS Embedded Word Alignment and FPGA FIFO-based Channel Alignment
- Multi-Channel Alignment exceeds the inter-channel skew requirements of DVI and HDMI specifications
- Emulated Extended Display Identification Data (EDID) support for DVI

3. HDMI/DVI Overview

3.1. What is DVI?

DVI stands for Digital Visual Interface, a video interface standard created by the Digital Display Working Group (DDWG) in 1999 to replace the legacy analog VGA connector standard. It is designed for carrying uncompressed digital video data to a display. It is partially compatible with the HDMI standard in digital mode (DVI-D) and VGA in analog mode (DVI-A). This document focuses on DVI-D mode. DVI handles a single-link bandwidth up to 165 MHz and thus supports UXGA and HDTV with resolutions up to 1600 x 1200 at 60Hz. Higher resolutions can be supported with reduced blanking periods or with a dual-link connection.

3.2. What is HDMI?

Short for High-Definition Multimedia Interface, HDMI is an industry-supported, all-digital audio/video interface for transmitting both uncompressed digital video and multi-channel audio over a single connector and cable that replaces the maze of cabling behind the home entertainment center. HDMI can carry eight channels of 192 kHz, 24-bit uncompressed audio, or any flavor of compressed audio format such as Dolby or DTS. HDMI has the capacity to support existing high-definition video formats such as 720p, 1080i, and 1080p. Because HDMI is electrically compatible with the DVI signal, no signal conversion is required, nor is there a loss of video quality when a DVI-to-HDMI adapter is used.

4. What is TMDS?

TMDS, or Transition Minimized Differential Signaling, is a technology for transmitting high-speed serial data and serves as the underlying protocol for both the HDMI and DVI standards.

A TMDS link consists of a single clock channel and three data channels. An advanced data-encoding algorithm, implemented on each of the three data channels, converts 8 bits of video or 4 bits of audio data into a 10-bit transition-minimized, DC-balanced sequence. This reduces electromagnetic interference (EMI) over copper cable and enables robust clock data recovery at the receiver with greater skew tolerance.

The TMDS electrical interface is current-driven, DC-coupled and terminated to 3.3 V. Figure 4.1 shows the conceptual schematic of a pair of TMDS differential links.

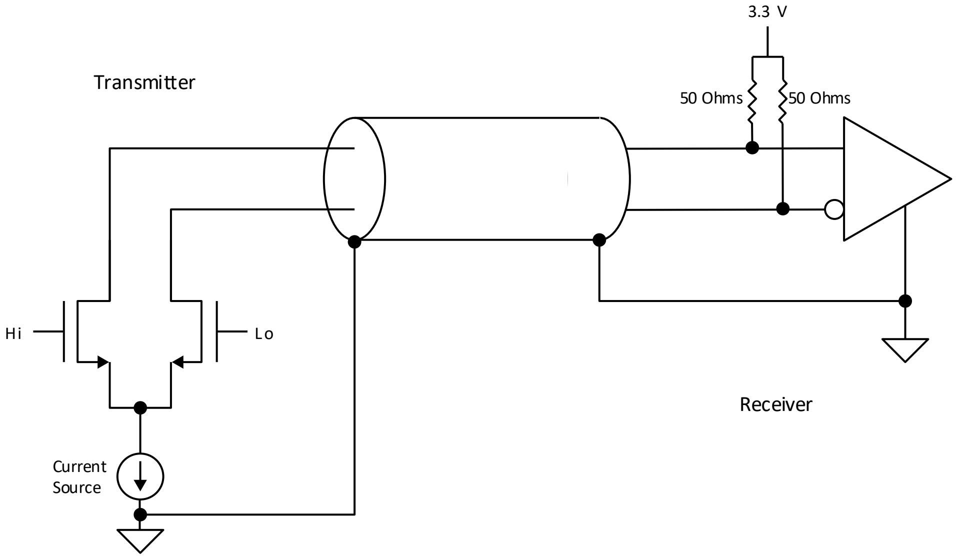


Figure 4.1. TMDS Differential Link

4.1. Using the CML SERDES to Interface to TMDS

This reference design takes advantage of the CML SERDES in the LatticeECP3 FPGA device to transmit and receive the TMDS signaling used by DVI and HDMI. This unique implementation leverages the CDR PLL built in the SERDES to reliably recover the source synchronous data. By implementing the DVI and HDMI interface in this way, it achieves up to a full 1.65 Gbps data rate with a low-cost FPGA device.

The TMDS electrical interface is similar to the differential CML SERDES interface, but it is DC-offset based on the specification. If the TMDS transmitter does not detect the DC offset at the receiver’s end, a transmitter may be turning off the current drive. Due to the different common mode voltage levels between these two interfaces, AC coupling is required in order for the SERDES CML interface to connect to the DC-offset TMDS interface (see Figure 4.2).

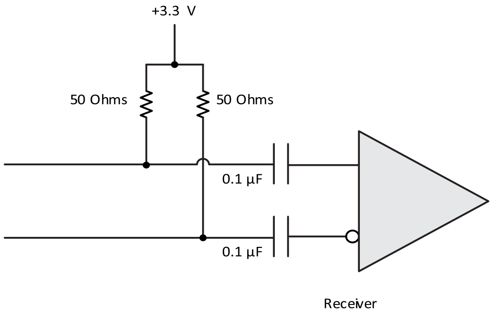


Figure 4.2. AC Coupling Circuit

Table 4.1 shows the difference in electrical characteristics between the LatticeECP3 SERDES CML interface and HDMI/DVI TMDS interface.

Table 4.1. Electrical Characteristics Comparison

HDMI Specification		**CML Characteristics**	
HDMI Source	75 ps ≤ Rise Time/Fall Time ≤ 0.4 Tbit	SERDES Tx	Rise Time/Fall Time (20% to 80%): 185 ps typ.
	Single-ended Swing: 400 mV ~ 600 mV		Differential Swing: 480 mV ~ 1730 mV p-p configurable
	Output Common Mode: (AVcc - 300 mV) ≤ Vicm1 ≤ (AVcc - 37.5 mV)		Output Common Mode: VCCOB - 0.60
	Intra Pair Skew: 0.15 Tbit		Intra Pair Skew: 75 ps max.
	Inter Pair Skew: 0.20 Tchar		Inter Pair Skew: 800 ps max.
	TMDS Differential Clock Jitter: 0.25 UI max.		TMDS Differential Clock Jitter: 0.25 UI max
HDMI Sink	Vidiff: 150 mV ~ 1560 mV	SERDES Rx	Differential Input Sensitivity: 150mV ~ 1760 mV
	Input Common Mode Voltage: (AVcc - 300 mV) ≤ Vicm1 ≤ (AVcc - 37.5 mV) Vicm2 (AC-coupled) = AVcc±10 mV		Input Common Mode Voltage: (50 Ohm, AC coupled): 50 mV
	TMDS Clock Jitter: 0.30 Tbit		Clock Jitter Tolerance: 0.25 UI max.
	Termination Resistance RT: 50 Ohm		Termination Resistance: 40 ~ 60 Ohm

As shown in Table 4.1, the common mode voltages are noticeably different between SERDES CML and TMDS electrical interfaces. In order for the SERDES transmitter to conform to HDMI revision 1.3 specifications, an external HDMI Level Shifter (e.g., STHDLS101T) is required to convert the AC-coupled CML differential signal to 3.3 V DCcoupled TMDS outputs. In certain applications where HDMI compliance is not required, the SERDES transmitter with AC coupling capacitors can drive the HDMI receiver directly.

The SEREDES receiver coupled with AC capacitors can interface the 3.3 V DC coupled TMDS inputs directly. An optional equalizer can be added to overcome the inter-symbol interference (ISI) jitter and regenerate the incoming attenuated TMDS signal.

HDMI Mezzanine Card Revision B User's Guide (EB55) contains schematics showing the connections and terminations between the LatticeECP3 SERDES CML interface and the HDMI-compliant TMDS interface.

4.2. HDMI/DVI Link Topology

HDMI or DVI consists of three data channels and one pixel clock channel. In DVI, three channels are designated for the red, green and blue colors. HDMI data channels also use the RGB color space by default, but they can also carry Luminance and Chrominance components (YCbCr) in each video pixel data. Figure 4.3 shows the topology of the HDMI link with encoders and decoders. A single link of DVI has a similar topology except for the Auxiliary Data (e.g. Audio Sample and InfoFrame) packets carried during the HDMI Data Island period exclusively.

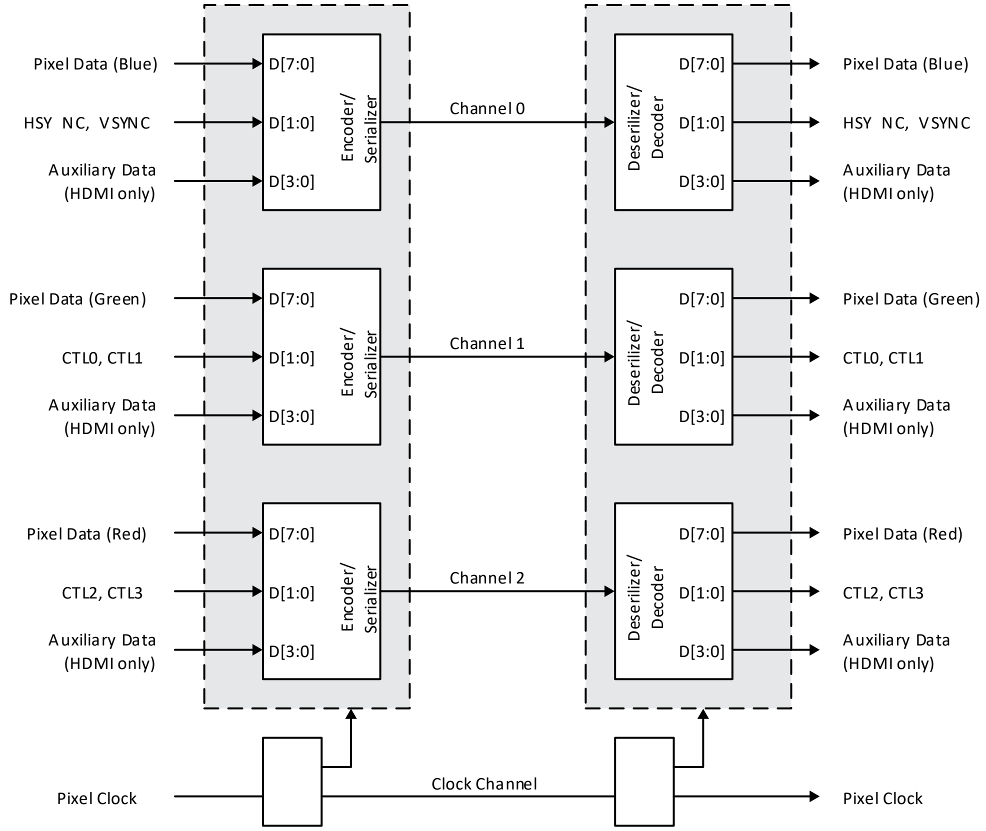


Figure 4.3. Single Link HDMI/DVI

4.3. HDMI/DVI Coding Scheme

The TMDS protocol incorporates an advanced coding algorithm which has reduced electromagnetic interference (EMI) over copper cables and enables robust clock recovery at the receiver to achieve high skew tolerance for driving longer cables as well as shorter low-cost cables. HDMI and DVI share many common aspects of the TMDS signaling and coding algorithm. This reference design implements a multi-protocol encoder and decoder, which allows a seamless switch between DVI and HDMI data streams.

The DVI data stream contains pixel and control data. The DVI transmitter encodes either pixel data or control data on any given input clock cycle, depending on the state of the Date Enable signal (DE). When DE is high, the 8-bit pixel data is encoded into 10-bit transition-minimized, DC-balanced TMDS sequence. During the blanking period when DE is low, the DVI transmitter encodes 2-bit control data into 10-bit sequence with high-transition. Figure 4.4 shows the relationship between the DVI periods and the DE signal.

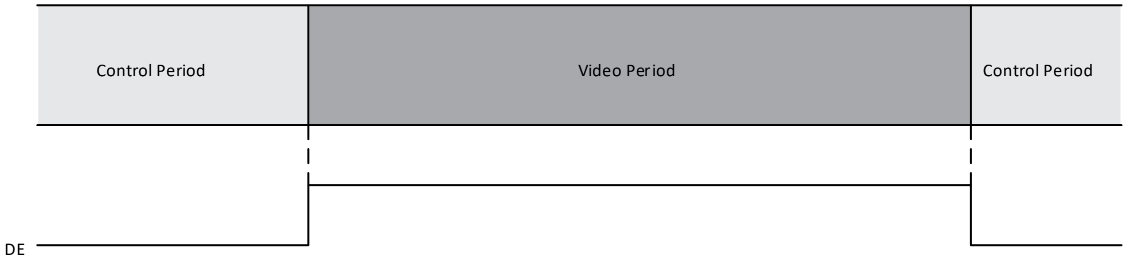


Figure 4.4. DVI Periods

The HDMI link operates in one of the three periods: Video Data Period, Data Island Period and Control Period. During the Video Data Period, the active video pixels are transmitted. During the Data Island Period, the audio and auxiliary data packets are transmitted. The Control Period is used when no video, audio or auxiliary data needs to be transmitted. A Control Period is required between any two periods that are not Control Periods.

Unlike the DVI link which only needs one DE signal to delineate the boundary between the Video Period and the Control Period, the HDMI link requires at least two indicators to delineate the boundaries between the Video Data Period, the Data Island Period and the Control Period.

This design has defined two indicators, Video Data Enable (VDE) and Audio/auxiliary Data Enable (ADE). The VDE is compatible with Data Enable (DE) used for the DVI link. When VDE is high, the HDMI link operates in the Video Period; when ADE is high, the HDMI link operates in the Data Island Period; when both VDE and ADE are low, the HDMI link operates in the Control Period. The relationship between the VDE/ADE indicators and the HDMI operation periods is shown in Figure 4.5.

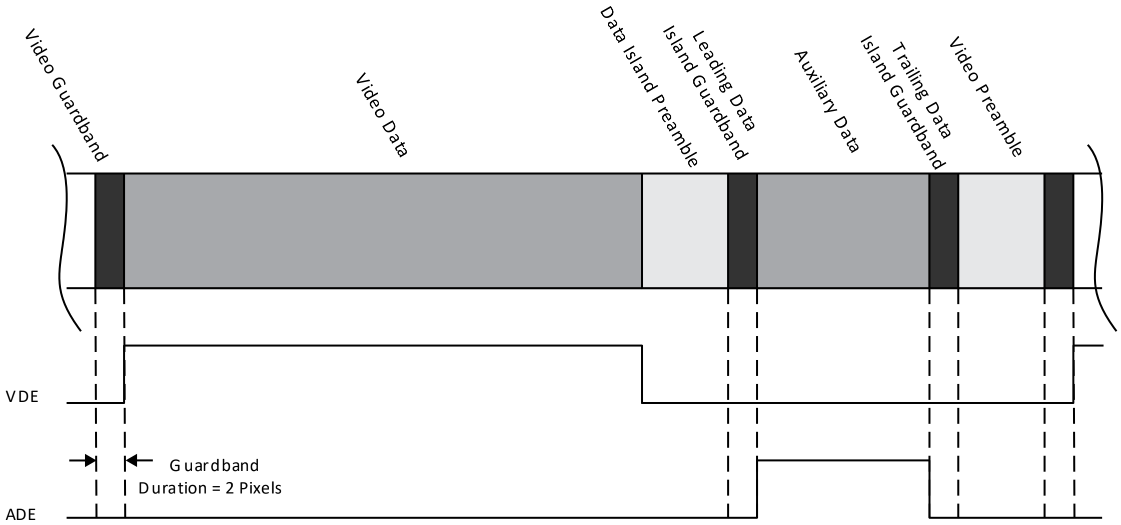

Video Guardband
Video Data
Data Island Preamble
Leading Data Island Guardband
Auxiliary Data
Trailing Data Island Guardband
Video Preamble
VDE
Guardband Duration = 2 Pixels
ADE

Figure 4.5. HDMI Periods

The HDMI and DVI protocols share the same algorithms to encode 8-bit video data and 2-bit control data into the 10-bit TMDS sequence. Additionally, the HDMI uses TMDS Error Reduction Coding (TERC4) to encode 4-bit audio and auxiliary data into the 10-bit TMDS sequence during the Data Island Period.

Video Data Coding

TMDS 8b/10b encoding for video data uses a code set that differs from the original IBM form. A two-stage process converts 8-bit code into 10-bit code with particularly desirable properties. Figure 4.6 shows an example of the TMDS 8b/10b coding scheme. In the first stage each bit is either XOR or XNOR transformed against the previous bit, while the first bit is not transformed at all. The encoder chooses between XOR and XNOR by determining which will result in the fewest transitions; the ninth bit is added to show which was used. In the second stage, the first eight bits are optionally inverted to even out the balance of ones and zeros in the serial stream. The tenth bit is added to indicate whether this inversion took place.

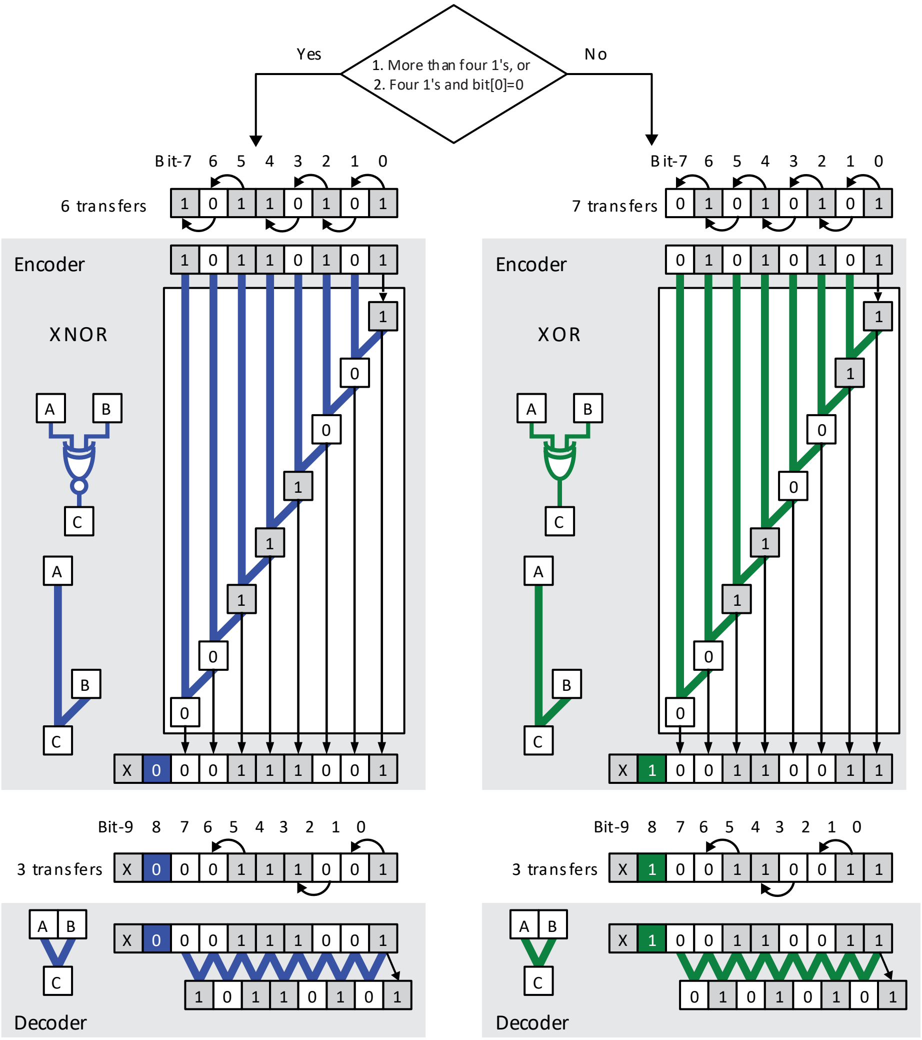


Figure 4.6. Video Data Coding Scheme

Control Period Coding

Each TMDS channel has two control signals, which are encoded as shown in Figure 4.7.

TMDS Channel	D0	D1
0	HSYNC	VSYNC
1	CTL0	CTL1
2	CTL2	CTL3

```
Case (D1, D0):
    0, 0 : q_out = 0b1101010100;
    0, 1 : q_out = 0b0010101011;
    1, 0 : q_out = 0b0101010100;
    1, 1 : q_out = 0b1010101011;
```

Figure 4.7. Control Period Coding Scheme

TERC4 Coding

TMDS Error Reduction Coding (TERC4) is used during the Data Island Period to encode 4 bits per channel into a 10-bit sequence. The TERC4 code group is shown in Figure 4.8.

```
case (D3, D2, D1, D0):
0000: q_out[9:0] = 0b1010011100;
0001: q_out[9:0] = 0b1001100011;
0010: q_out[9:0] = 0b1011100100;
0011: q_out[9:0] = 0b1011100010;
0100: q_out[9:0] = 0b0101110001;
0101: q_out[9:0] = 0b0100011110;
0110: q_out[9:0] = 0b0110001110;
0111: q_out[9:0] = 0b0100111100;
1000: q_out[9:0] = 0b1011001100;
1001: q_out[9:0] = 0b0100111001;
1010: q_out[9:0] = 0b0110011100;
1011: q_out[9:0] = 0b1011000110;
1100: q_out[9:0] = 0b1010001110;
1101: q_out[9:0] = 0b1001110001;
1110: q_out[9:0] = 0b0101100011;
1111: q_out[9:0] = 0b1011000011;
endcase;
```

Figure 4.8. TERC4 Coding Scheme

5. Functional Descriptions

This HDMI/DVI Encoder and Decoder reference design is logically divided into two portions: the Receiver and Transmitter. The functional block diagrams referenced in this document work for both HDMI and DVI standards if not otherwise clarified.

5.1. HDMI/DVI Receiver

The HDMI and DVI serial data streams are recovered and de-serialized over the three SERDES receive channels with the quad reference clock driven by the HDMI/DVI pixel clock. Without the proper reset sequence, the SERDES/PCS module may not work. This function is implemented in the Reset Sequence module (see Figure 5.1). For Lattice Diamond® 1.1, or later versions, this module has been integrated into the SERDES/PCS module (see Figure 6.6). The embedded PCS Word Aligner detects the four Control Character patterns in the serial data stream and aligns the 10-bit TMDS character boundary before transmitting the data to the FPGA fabric for decoding. The HDMI and DVI receivers share the common functional blocks as shown in Figure 5.1. The HDMI receiver contains three decoders for 10b8b, 10b4b and 10b2b decoding, while the DVI receiver only needs two of them for 10b8b and 10b2b decoding. The HDMI Receiver can decode both HDMI and DVI data stream by automatically detecting the data format and decoding the data with the proper decoders.

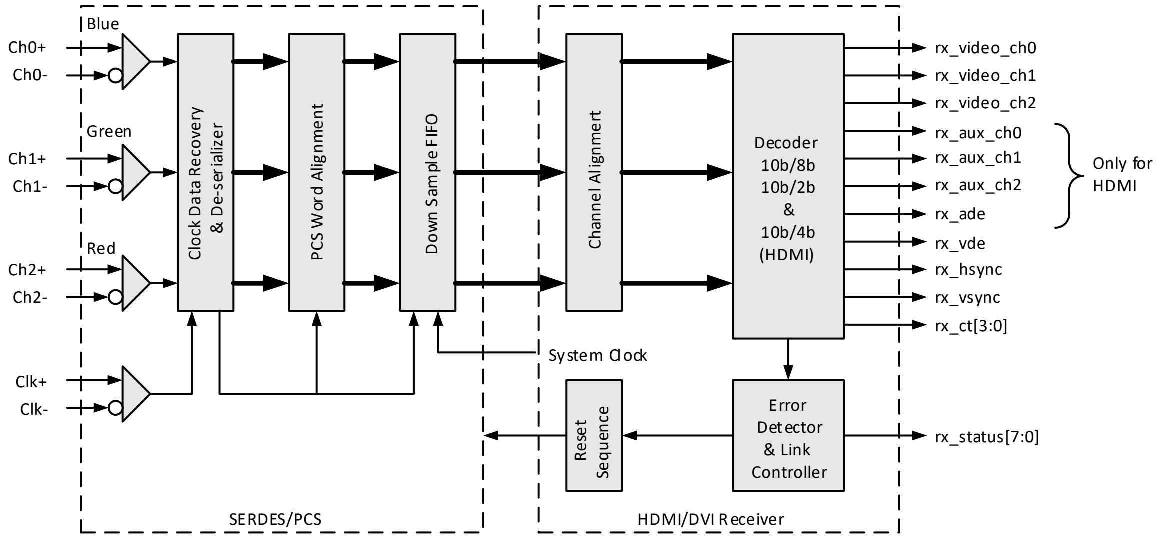


Figure 5.1. HDMI/DVI Receiver and SERDES/PCS Block

Table 5.1 and Table 5.2 depict the interface signals of the HDMI/DVI Receiver implemented in the FPGA fabric.

Table 5.1. HDMI Receiver Interface

Signal Name	Input/Output	Description
PCS Interface		
hdmi_rxd_ch0[9:0]	In	10-bit HDMI received raw data of Channel 0
hdmi_rxd_ch1[9:0]	In	10-bit HDMI received raw data of Channel 1
hdmi_rxd_ch2[9:0]	In	10-bit HDMI received raw data of Channel 2
User End’s Interface		
rx_video_ch0[7:0]	Out	8-bit TMDS decoded video data of Channel 0
rx_video_ch1[7:0]	Out	8-bit TMDS decoded video data of Channel 1
rx_video_ch2[7:0]	Out	8-bit TMDS decoded video data of Channel 2
rx_audio_ch0[3:0]	Out	4-bit TERC4 decoded audio data of Channel 0
rx_audio_ch1[3:0]	Out	4-bit TERC4 decoded audio data of Channel 1
rx_audio_ch2[3:0]	Out	4-bit TERC4 decoded audio data of Channel 2
rx_ctl[3:0]	Out	4-bit control bus decoded from Channels 1 & 2
rx_hsync	Out	HSYNC decoded from TMDS Channel 0
rx_vsync	Out	VSYNC decoded from TMDS Channel 0
rx_vde	Out	Video data enable
rx_ade	Out	HDMI audio/auxiliary data enable for receiver
rx_format	Out	Received data format. 1 = HDMI; 0 = DVI
System Interface		
rxclk	In	Receiver pixel clock
rstn	In	Asynchronous reset, active low
resync	In	Synchronous reset, active high
rx_status[7:0]	Out	8-bit receiver status indicator [0] Word alignment error [1] Multi-channel alignment error [2]Channel alignment FIFO overflow [3]Decoding error [4]Word synched [5] Channel synched [6.7] Reserved

Table 5.2. DVI Receiver Interface

Signal Name	Input/Output	Description
PCS Interface		
dvi_rx_ch0[9:0]	In	10-bit DVI received raw data of Channel 0
dvi_rx_ch1[9:0]	In	10-bit DVI received raw data of Channel 1
dvi_rx_ch2[9:0]	In	10-bit DVI received raw data of Channel 2
User End's Interface		
rx_video_ch0[7:0]	Out	8-bit TMDS decoded video data of Channel 0
rx_video_ch1[7:0]	Out	8-bit TMDS decoded video data of Channel 1
rx_video_ch2[7:0]	Out	8-bit TMDS decoded video data of Channel 2
rx_hsync	Out	TMDS decoded HSYNC signal
rx_vsync	Out	TMDS decoded VSYNC signal
rx_vde	Out	Video data enable signal
rx_ctl[3:0]	Out	4-bit decoded control bus
System Control Interface		
rxclk	In	Receiver pixel clock
rstn	In	Asynchronous reset, active-low
srst	In	Synchronous reset, active-high
resync	In	Resync command generated by link controller
rx_status[7:0]	Out	8-bit receiver status indicator [0] Word alignment error [1] Word alignment enable [2] Word aligned [3] Multi-channel alignment error [4] Channel alignment FIFO overflow [5] Channel aligned [6:7] [6.7] Reserved

5.2. PCS Word Aligner

The LatticeECP3 PCS Word Aligner searches for the 10-bit word boundary using the comma characters as specified in clause 36.2.4.9 of the IEEE 802.3 standard. A number of programmable options are also supported within the Word Aligner to allow users to customize the comma characters for word alignment. The COMMA_A and COMMA_B registers can store two 10-bit alignment characters; the COMMA_M mask register defines which word alignment bits to compare (a '1' in the mask register bit means check the corresponding bit of comma character; a '0' masks the corresponding bit).

The TMDS protocol does not use the comma characters as specified in IEEE 802.3 standard for word alignment. However, TMDS provides four control characters which are periodically transmitted during the Control Period. While these four control characters are not individually unique in the serial data stream, they are sufficiently alike with the signature of high-transition content distinguishing them from the rest of TMDS encoded data with reduced transitions transmitted during the Video Data Period or Data Island Period. The Word Aligner detects the presence of a succession of control characters with high-transition content and determines the location of the TMDS word boundary in the serial data stream.

The LatticeECP3 PCS Word Aligner only provides two programmable registers per channel, COMMA_A and COMMA_B, for users to define alignment characters. By masking the most significant bit, these two registers can represent all four control characters as shown in Figure 5.2.

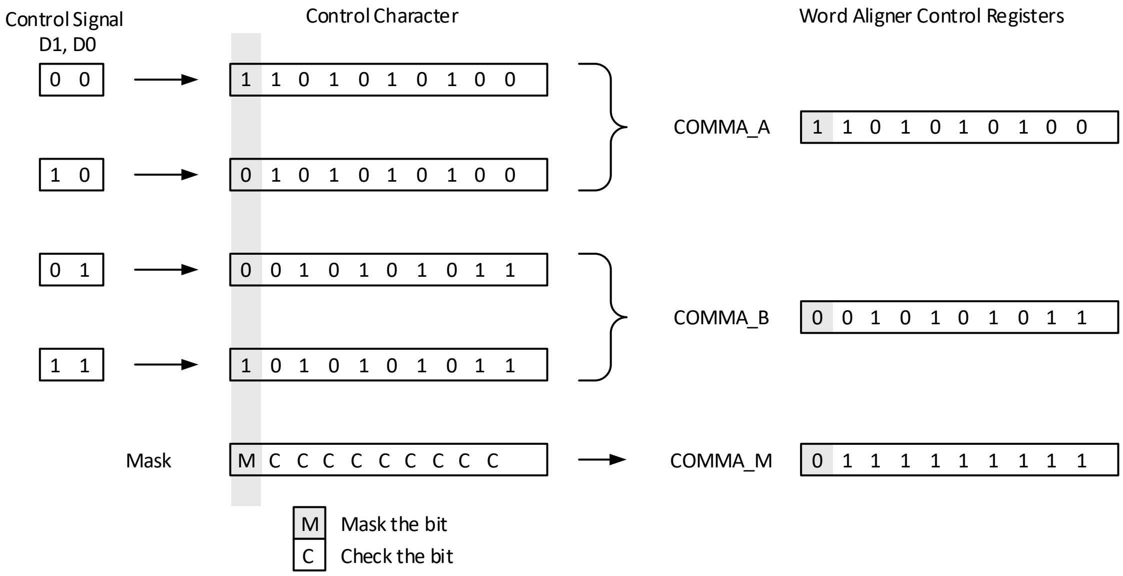


Figure 5.2. Programmable Comma Character and Mask Registers

5.3. Link Controller

Since the LatticeECP3 PCS Internal Link State Machine does not support the TMDS protocol, the CHx_ILSM (Internal Link State Machine) register is set to “DISABLED”, and the CHx_RXWA (Word Alignment) register is set to “ENABLED”. The PCS word aligner will lock the word boundary on the first match to the user-defined comma characters and stay locked. If re-alignment is required, pulse the word_align_en_ch(0-3) inputs from low to high.

A TMDS link controller is implemented in the FPGA fabric to monitor the word alignment status. It issues a command to re-align the 10-bit word boundary only under specific conditions. The link controller checks the control characters which are repeatedly coded during the Control Period across three HDMI/DVI data channels. If the control characters cannot be detected for a pre-defined interval, the TMDS link controller resets the PCS Word Aligner as well as other functional blocks in the FPGA.

5.4. PCS Downsample FIFO

The LatticeECP3 PCS provides Downsample FIFOs and Upsample FIFOs as the bridges between the PCS and the FPGA fabric. For each receive channel, the 10-bit words are written into the Downsample FIFO with the Receiver channel’s recovered clocks, and read out with the system clock. The Downsample FIFO, acting as phaseshift FIFO, transfers the 10-bit words from the per-channel recovered clock domain to the common system clock domain. There are two different clock schemes that can be used for the HDMI/DVI Encoder and Decoder design. One scheme chooses the recovered clock from one of three TMDS Data Receiver Channels, another uses the fullrate clock from the Transmitter PLL. The latter one, as shown in Figure 5.3, is preferable because the quad- level full rate clock from the TX PLL (tx_full_clk) has direct access to the FPGA center clock mux. This is a relatively higher performance path.

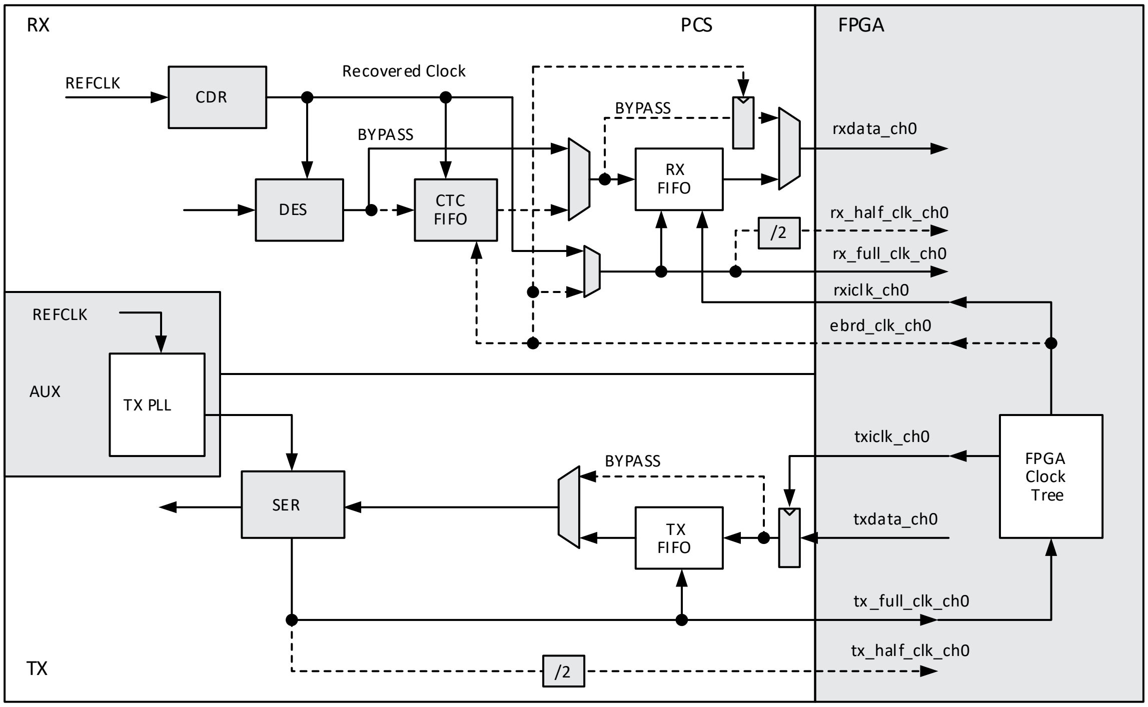


Figure 5.3. Clock Scheme Using tx_full_clk

5.5. Multi-Channel Alignment

The DVI and HDMI specifications define the Maximum Allowable Inter-Pair (channel-to-channel) Skew at Sink Connector. For DVI, it is up to 0.6 Tpixel; for HDMI, it is up to 0.2 Tcharacter + 1.78 ns. The Multi-Channel alignment function of the PCS allows data streams over three data channels with slight phase variances to be synchronized based on the frame structure defined in the HDMI/DVI specifications. It is important to ensure that the data arrives at the FPGA TMDS decoder end in perfect word and channel-to-channel synchronization.

The TMDS protocol used by HDMI and DVI does not define a special comma character for channel alignment. But the transition from the Control Period to the Video Data Period or Data Island Period can be used as a marker for channel alignment. The Multi-Channel Alignment is achieved by lining up these markers over all three data channels through the alignment FIFO.

The alignment FIFO per channel is provided with a 16-word x 10-bit distributed-RAM based DPRAM. Theoretically it can remove up to 16 pixel-clocks of skew, exceeding the HDMI and DVI specification requirements. The algorithm in control writes to the alignment FIFOs and reads from them, operating as follows:

1. Prior to detecting the alignment marker from any channel, each channel continuously writes to address 0 of its corresponding alignment FIFO.
2. When any channel detects the alignment marker, the write pointer for the corresponding FIFO increments one on each clock cycle; channels that have not detected a marker continue to write into address 0 of their respective FIFOs.
3. When all channels have detected the alignment markers, and are continuously incrementing their write pointers while writing into their corresponding FIFOs, the data is then simultaneously read out of each channel's FIFO one word at a time.

By lining up the markers at address 0, and commencing the Read operations only when all channels have received the marker, the Multi-Channel Alignment algorithm (as shown in Figure 5.4) effectively removes the relevant skews among three data channels.

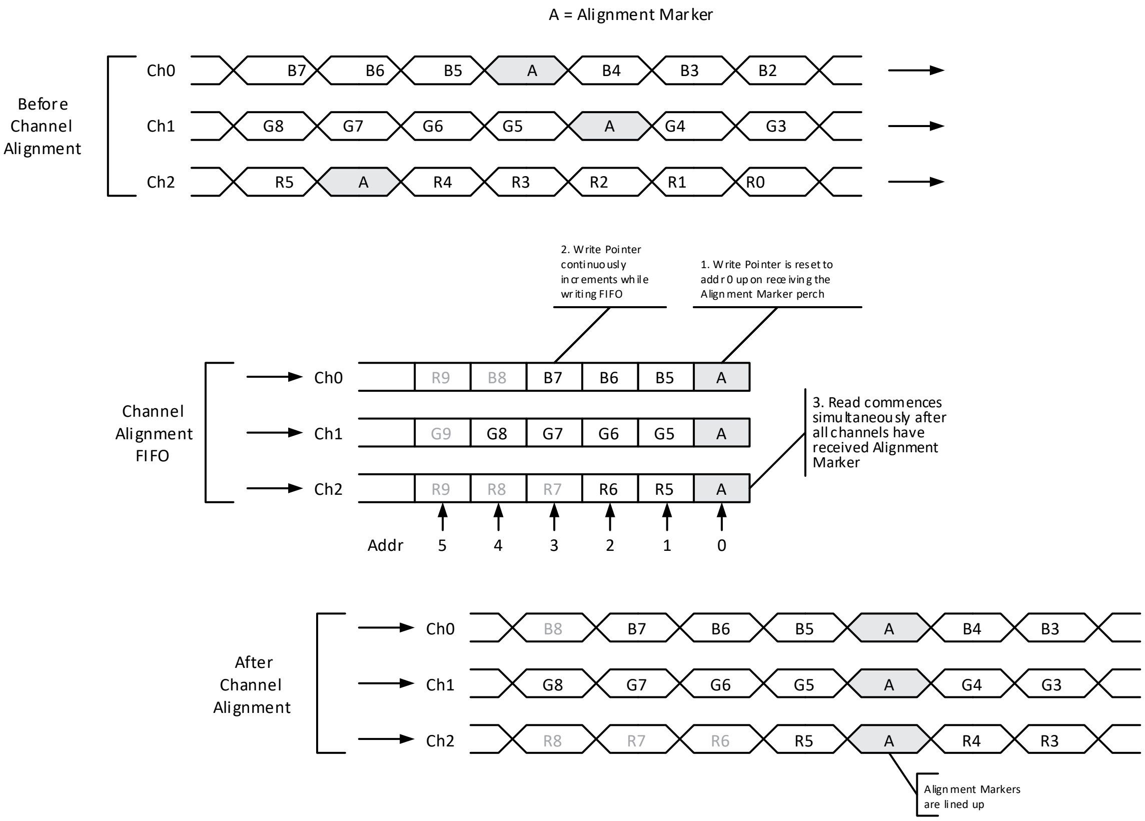

A = Alignment Marker
Before Channel Alignment
Ch0
B7
B6
B5
A
B4
B3
B2
Ch1
G8
G7
G6
G5
A
G4
G3
Ch2
R5
A
R4
R3
R2
R1
R0
2. Write Pointer continuously increments while writing FIFO
1. Write Pointer is reset to addr 0 upon receiving the Alignment Marker perch
Channel Alignment FIFO
Ch0
R9
B8
B7
B6
B5
A
Ch1
G9
G8
G7
G6
G5
A
Ch2
R9
R8
R7
R6
R5
A
3. Read commences simultaneously after all channels have received Alignment Marker
Addr
5
4
3
2
1
0
After Channel Alignment
Ch0
B8
B7
B6
B5
A
B4
B3
Ch1
G8
G7
G6
G5
A
G4
G3
Ch2
R8
R7
R6
R5
A
R4
R3
Alignment Markers are lined up

Figure 5.4. Multi-Channel Alignment Algorithm

5.6. Decoder

The Decoder for the HDMI link contains three sub-decoder modules: the 10b8b TMDS Decoder for the Video Data period, the 10b4b TERC4 Decoder for the Data Island Period, and the 10b2b Decoder for the Control Period. The same TMDS Decoder works seamlessly for both HDMI and DVI links. The TERC4 Decoder is only enabled when the HDMI data format is detected. Figure 5.5 shows the block diagram of the Decoder used for one TMDS data channel.

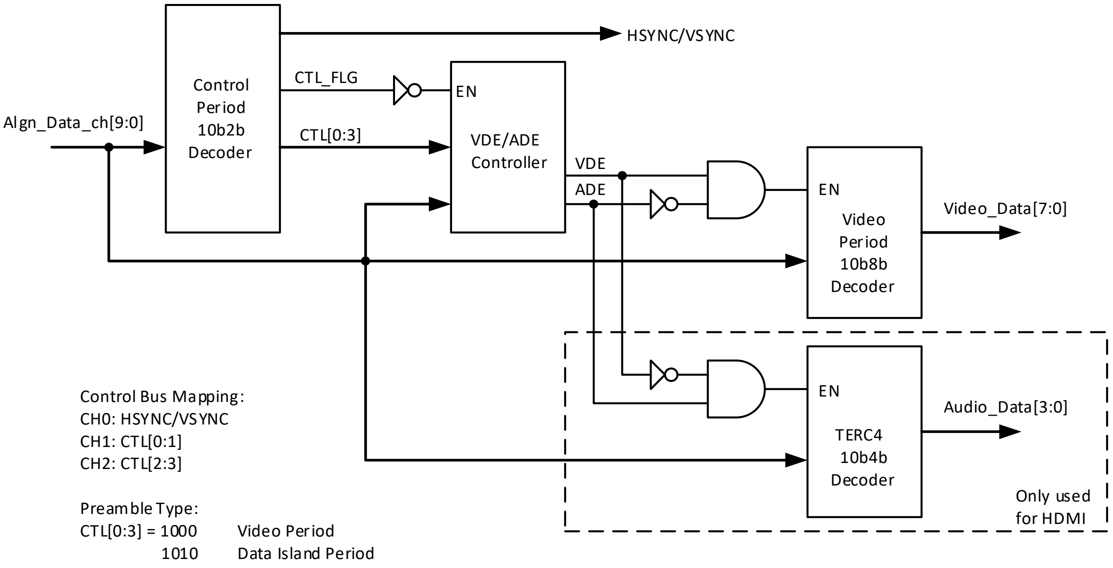


Figure 5.5. HDMI/DVI Decoder Block Diagram

For a DVI link, only VDE is used as the DE indicator to delineate the boundary between the Control Period and the Video Data Period. For an HDMI link, both VDE and ADE are required to delineate the boundaries between the Control Period, the Video Data Period and the Data Island Period.

Only in the HDMI sequence, the Video Data Period and Data Island Period are preceded by eight Preamble characters and two Leading Guard Band characters. The Data Island packet is also followed by two Trailing Guard Band characters. There is no Trailing Guard Band for the Video Data Period.

Though sharing some common decoding algorithms, the DVI and HDMI sequences go through different decoding flows as shown in Figure 5.6.

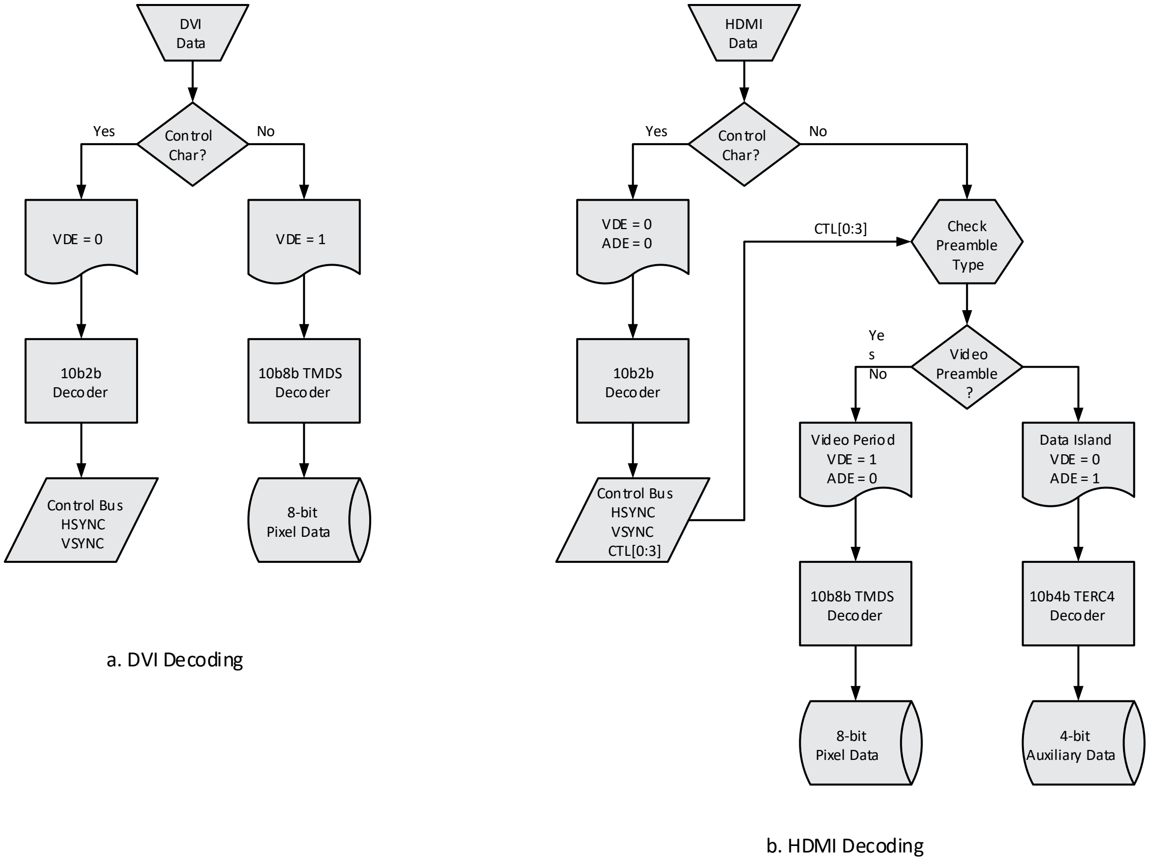


Figure 5.6. DVI and HDMI Decoding Scheme

DVI Decoding

- When Control Characters are detected, drive VDE low, convert the 10-bit control characters of Channel 0 (Blue) into HSYNC and VSYNC;
- When Control Characters are not found, drive VDE high, convert the 10-bit TMDS code into 8-bit video pixel data.

HDMI Decoding

- When Control Characters are detected, drive VDE and ADE low, decode control character into two-bit control bus per channel and remap them as:
  - HSYNC and VSYNC for Channel 0 (Blue)
  - CTL0 and CTL1 for Channel 1 (Green)
  - CTL2 and CTL3 for Channel 2 (Red)
- Decode CTL[0:3], if the Preamble type is Video Preamble:
  - Drive VDE high and ADE low
  - Convert 10-bit TMDS code into 8-bit video pixel data
- If the Preamble Type is Data Island Preamble:

- Drive ADE high and VDE low
- Convert 10-bit TERC4 code into 4-bit auxiliary data

5.7. HDMI/DVI Transmitter

The HDMI and DVI Transmitters encode the video pixel data, audio and auxiliary data, HSYNC/VSYNC and other control signals into three 10-bit TMDS data, and serializes them through the LatticeECP3 SERDES for high-speed transmission. An additional SERDES channel is used to transmit the pixel clock.

Because the video and audio sources are all aligned before encoding, there is no need to implement channel-tochannel alignment and character synchronization for the HDMI/DVI Transmitter. The Data Island placement and duration is based on the Audio/auxiliary Data Enable signal (ADE) derived from the Receiver Channel. Figure 5.7 shows the functional block diagram of the HDMI/DVI Transmitter.

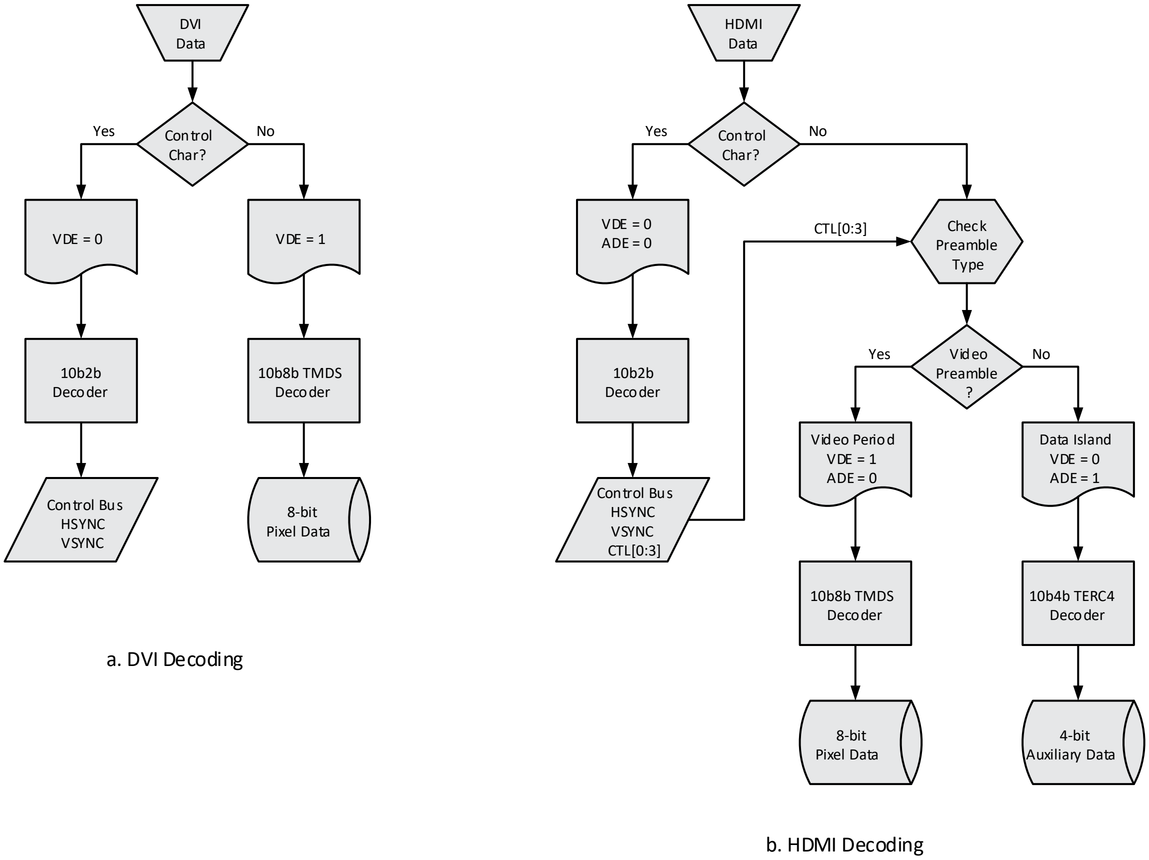


Figure 5.7. HDMI/DVI Transmitter and SERDES/PCS Block

Table 5.3 and Table 5.4 depict the top-level interface signals of the HDMI/DVI Transmitter implemented in the FPGA fabric.

Table 5.3. HDMI Transmitter Interface

Signal Name	Input/Output	Description
PCS Interface		
hdmi_txd_ch0[9:0]	Out	10-bit encoded data for HDMI Transmit Channel 0
hdmi_txd_ch1[9:0]	Out	10-bit encoded data for HDMI Transmit Channel 1
hdmi_txd_ch2[9:0]	Out	10-bit encoded data for HDMI Transmit Channel 2
User End's Interface		
tx_video_ch0[7:0]	In	8-bit video pixel data of Channel 0 for TMDS encoding
tx_video_ch1[7:0]	In	8-bit video pixel data of Channel 1 for TMDS encoding
tx_video_ch2[7:0]	In	8-bit video pixel data of Channel 2 for TMDS encoding
tx_audio_ch0[3:0]	In	4-bit audio/aux data of Channel 0 for TERC4 encoding
tx_audio_ch1[3:0]	In	4-bit audio/aux data of Channel 1 for TERC4 encoding
tx_audio_ch2[3:0]	In	4-bit audio/aux data of Channel 2 for TERC4 encoding
tx_hsync	In	HSYNC video control signal for transmit
tx_vsync	In	VSYNC video control signal for transmit
tx_ctl[3:0]	In	4-bit control bus for transmit
tx_vde	In	HDMI video data enable for transmit
tx_ade	In	HDMI audio/auxiliary data enable for transmit
tx_format	In	Transmit encoding format: 1 = HDMI, 0 = DVI
System Interface		
txclk	In	Transmit pixel clock
rstn	In	Asynchronous reset, active low
resync	In	Synchronous reset, active high

Table 5.4. DVI Transmitter Interface

Signal Name	Input/Output	Description
PCS Interface		
dvi_tx_ch0[9:0]	Output	10-bit encoded data for DVI Transmit Channel 0
dvi_tx_ch1[9:0]	Output	10-bit encoded data for DVI Transmit Channel 1
dvi_tx_ch2[9:0]	Output	10-bit encoded data for DVI Transmit Channel 2
User End's Interface		
tx_video_ch0[7:0]	Output	8-bit video pixel data of Channel 0 for TMDS encoding
tx_video_ch1[7:0]	Output	8-bit video pixel data of Channel 1 for TMDS encoding
tx_video_ch2[7:0]	Output	8-bit video pixel data of Channel 2 for TMDS encoding
tx_hsync	Output	HSYNC video control signal for transmit
tx_vsync	Output	VSYNC video control signal for transmit
tx_vde	Output	DVI video data enable signal for transmit
tx_ctl[3:0]	Output	4-bit control bus for transmit
System Interface		
txclk	Input	Transmit pixel clock
rstn	Input	Asynchronous reset, active-low
srst	Input	Synchronous reset, active-high

5.8. Encoder

The HDMI encoder converts the video pixel, audio and auxiliary data, and control signals through three sub-encoders: the 8b10b TMDS encoder for video pixel data, the 4b10b TERC4 encoder for audio and auxiliary data, and the 2b10b encoder for HSYNC/VSYNC and control bus. The DVI protocol does not transmit audio source, thus the 4b10b TERC4 encoder is bypassed when the DVI format is enabled. Figure 5.8 shows the functional block diagram of the HDMI/DVI encoder.

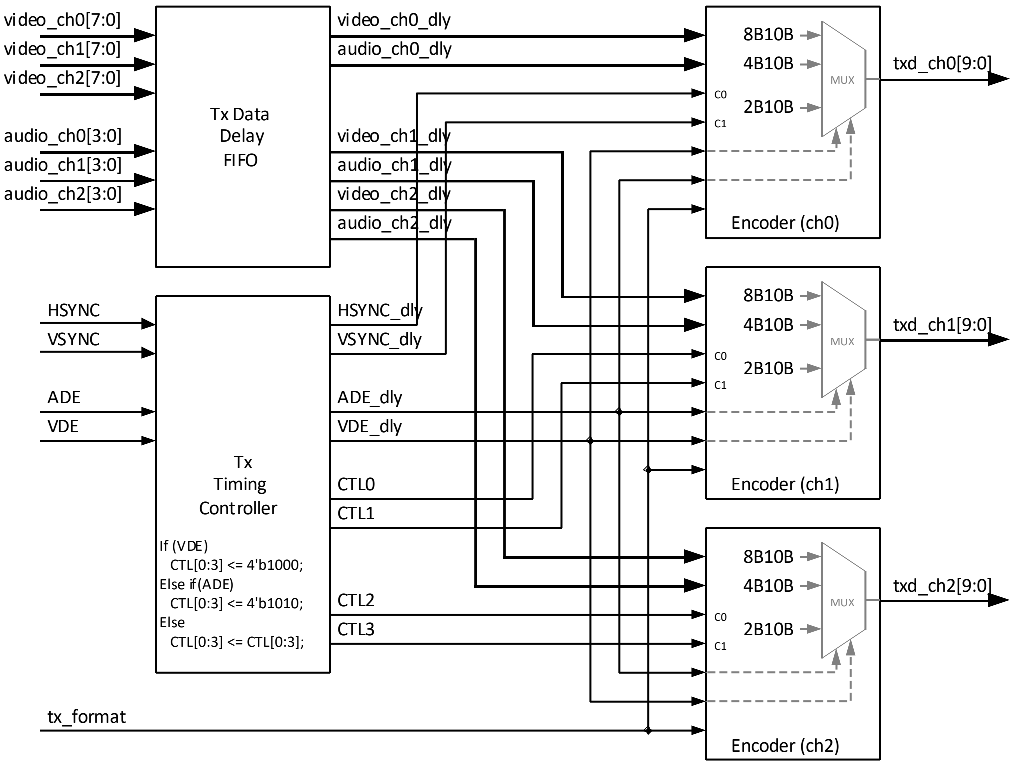


Figure 5.8. HDMI/DVI Encoder

The HDMI Transmitter does not provide the capability of automatic Data Island placement and requires the user interface to provide VDE, ADE, HSYNC and VSYNC input signals. For the HDMI/DVI Pass-through Demo, these control signals can be derived from the corresponding signals of the Receiver channels.

The Video Data period is preceded by at least eight Video Preambles and two video Leading Guard Bands. Similarly, the Data Island period is preceded by at least eight Data Island Preambles and two Data Island Guard Bands. At the end of Data Island Period, two Data Island Trailing Guard Bands are also transmitted. To accommodate the amount of time required to transmit preambles and guard bands, the video and audio data must be delayed for the same amount of time after the VDE the ADE signals are asserted.

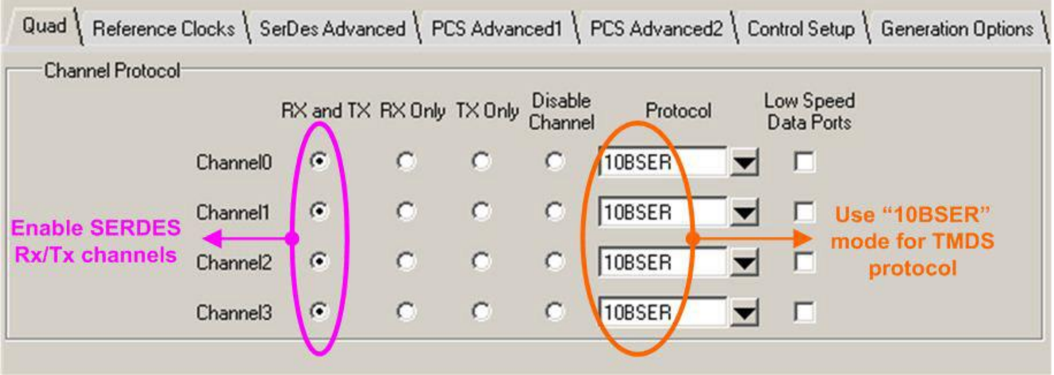
A tx_format input signal is provided to switch the encoder between the HDMI and DVI formats. The tx_format can be driven by the user interface or derived from the Rx_format detect signal of the Receiver channel. When tx_format is low, the TERC4 4B10B encoder is bypassed, only the video data is converted to TMDS data in DVI format. If an application does not require HDMI encoding, a simplified DVI encoder can be implemented without the Delay FIFOs.

6. SERDES/PCS Configuration

LatticeECP3 device SERDES/PCS supports multiple protocols with data rates from 150 Mbps to 3.2 Gbps. Using IPexpress, we can implement SERDES/PCS configuration. For further information, see LatticeECP3 SERDES/PCS Usage Guide (FPGA-TN-02190). Here, only the flow for LatticeECP3 SERDES/PCS configuration is shown.

The TMDS serial bit rate is 10 times the SERDES reference clock frequency which is equivalent to the pixel clock frequency defined for each video screen mode. The corresponding pixel clock frequencies for HDMI resolutions are defined in the CEA-861-D standard. Likewise, the pixel clock frequencies for DVI are defined in the Proposed VESA and Industry Standards and Guidelines for Computer Display Monitor Timing (DMTv1r12D3). By choosing 10BSER mode, the internal protocol-based 8B10B Decoder and the Link State Machine are bypassed.

As showing in Figure 6.1, the selection of “Rx and Tx”, “Rx Only” and “Tx Only” radio buttons is based on your actual design. You will need three channels for implementing HDMI/DVI receiver and four channels for implementing HDMI/DVI transmitter. Figure 6.1 shows an example where both the receiver and transmitter are implemented. To support HDMI/DVI TMDS protocol, a generic 10BSER mode is chosen.



Channel Protocol

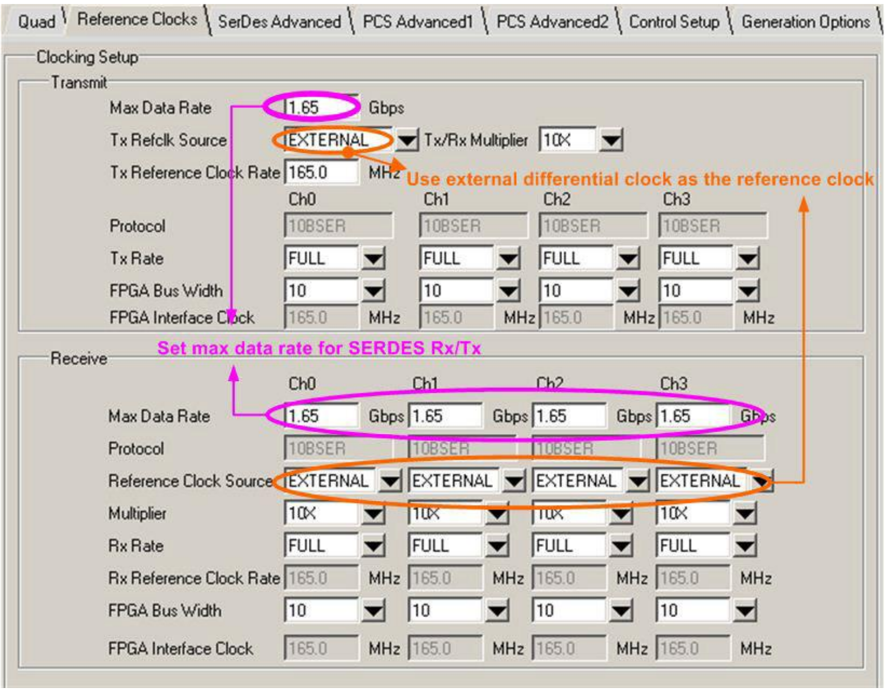
	RX and TX	RX Only	TX Only	Disable Channel	Protocol	Low Speed Data Ports
Channel0	●	○	○	○	10BSER	□
Channel1	●	○	○	○	10BSER	□
Channel2	●	○	○	○	10BSER	□
Channel3	●	○	○	○	10BSER	□

Enable SERDES Rx/Tx channels

Use “10BSER” mode for TMDS protocol

Figure 6.1. Channel_Enable and Mode_Selection

The serial data rate is set to 1.65 Gbps for both the Transmit and Receive channels. This serial data rate setting works for most of the HDMI/DVI resolutions. (Refer to Table 9.1 and Table 9.2 for the supported HDMI/DVI resolutions). Depending on your design, the Transmitter's Tx Refclk Source can be set to EXTERNAL or INTERNAL. We recommend setting this to EXTERNAL so that TMDS output with lower jitter can be achieved by a clean external reference clock. On the receiver side, as illustrated in Figure 5.1, the TMDS clock channel is connected to the SERDES external reference clock pins; therefore, the Reference Clock Source has to be set to EXTERNAL.



Clocking Setup

Transmit

Max Data Rate 1.65 Gbps

Tx Refclk Source EXTERNAL Tx/Rx Multiplier 10X

Tx Reference Clock Rate 165.0 MHz

Use external differential clock as the reference clock

	Ch0	Ch1	Ch2	Ch3
Protocol	10BSER	10BSER	10BSER	10BSER
Tx Rate	FULL	FULL	FULL	FULL
FPGA Bus Width	10	10	10	10
FPGA Interface Clock	165.0 MHz	165.0 MHz	165.0 MHz	165.0 MHz

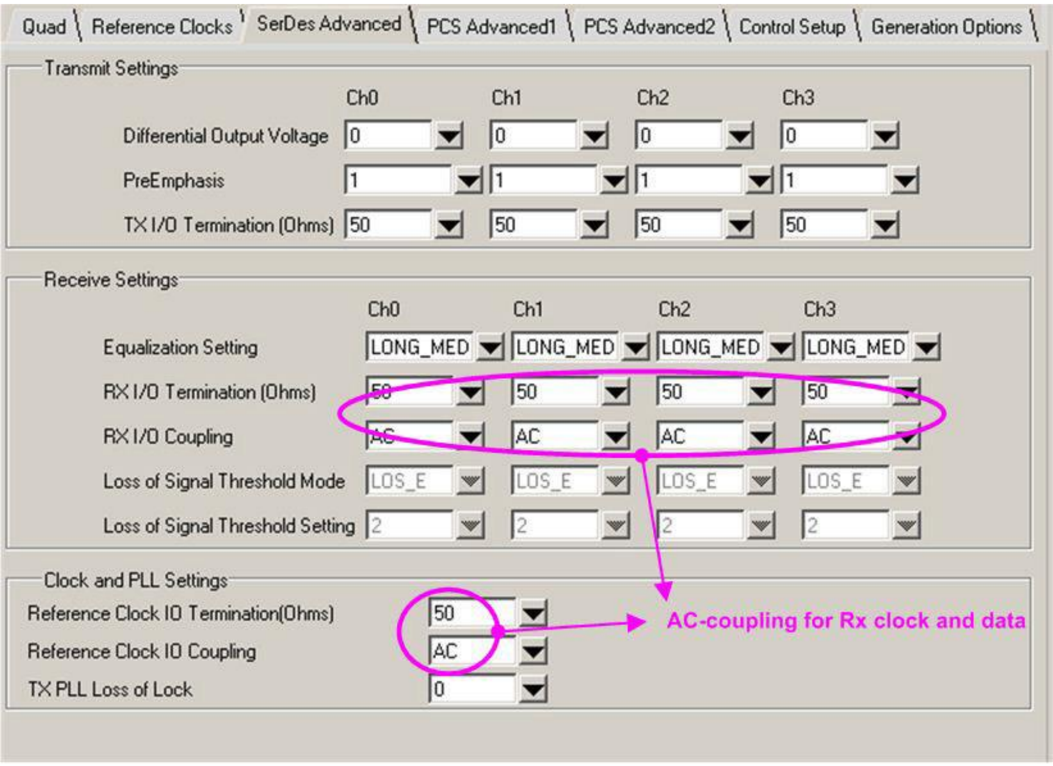
Set max data rate for SERDES Rx/Tx

Receive

	Ch0	Ch1	Ch2	Ch3
Max Data Rate	1.65 Gbps	1.65 Gbps	1.65 Gbps	1.65 Gbps
Protocol	10BSER	10BSER	10BSER	10BSER
Reference Clock Source	EXTERNAL	EXTERNAL	EXTERNAL	EXTERNAL
Multiplier	10X	10X	10X	10X
Rx Rate	FULL	FULL	FULL	FULL
Rx Reference Clock Rate	165.0 MHz	165.0 MHz	165.0 MHz	165.0 MHz
FPGA Bus Width	10	10	10	10
FPGA Interface Clock	165.0 MHz	165.0 MHz	165.0 MHz	165.0 MHz

Figure 6.2. Rx/Tx Data_Rate and Related Reference_Clock

AC-coupled configuration is required for SERDES.



Transmit Settings

	Ch0	Ch1	Ch2	Ch3
Differential Output Voltage	0	0	0	0
PreEmphasis	1	1	1	1
TX I/O Termination (Ohms)	50	50	50	50

Receive Settings

	Ch0	Ch1	Ch2	Ch3
Equalization Setting	LONG_MED	LONG_MED	LONG_MED	LONG_MED
RX I/O Termination (Ohms)	50	50	50	50
RX I/O Coupling	AC	AC	AC	AC
Loss of Signal Threshold Mode	LOS_E	LOS_E	LOS_E	LOS_E
Loss of Signal Threshold Setting	2	2	2	2

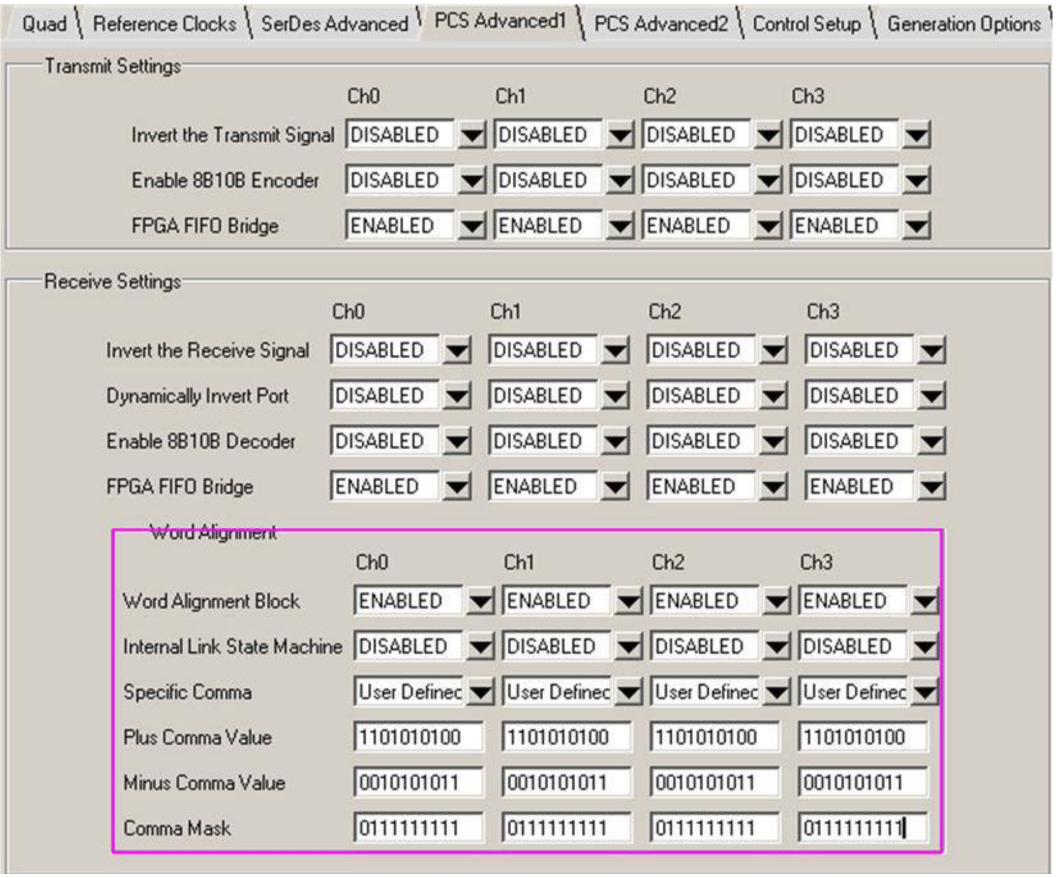
Clock and PLL Settings

Reference Clock IO Termination(Ohms)	50
Reference Clock IO Coupling	AC
TX PLL Loss of Lock	0

AC-coupling for Rx clock and data

Figure 6.3. AC Coupling

After receiving the serial data bits, the boundaries of the TMDS characters need to be identified so that the parallel 10-bit TMDS characters can be passed to the decoder. As illustrated in Figure 5.2, the PCS Word Alignment module is enabled to align the 10-bit raw data boundary with a set of user-defined alignment characters. Figure 6.4 highlights the settings for using the PCS Word Alignment feature.



Transmit Settings

	Ch0	Ch1	Ch2	Ch3
Invert the Transmit Signal	DISABLED	DISABLED	DISABLED	DISABLED
Enable 8B10B Encoder	DISABLED	DISABLED	DISABLED	DISABLED
FPGA FIFO Bridge	ENABLED	ENABLED	ENABLED	ENABLED

Receive Settings

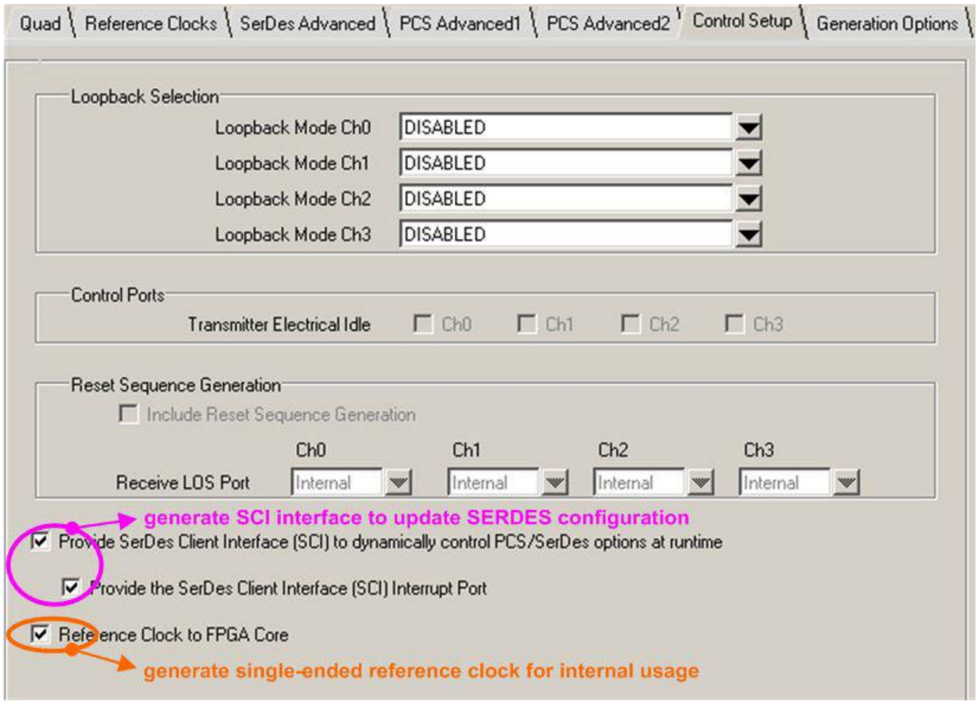
	Ch0	Ch1	Ch2	Ch3
Invert the Receive Signal	DISABLED	DISABLED	DISABLED	DISABLED
Dynamically Invert Port	DISABLED	DISABLED	DISABLED	DISABLED
Enable 8B10B Decoder	DISABLED	DISABLED	DISABLED	DISABLED
FPGA FIFO Bridge	ENABLED	ENABLED	ENABLED	ENABLED

Word Alignment

	Ch0	Ch1	Ch2	Ch3
Word Alignment Block	ENABLED	ENABLED	ENABLED	ENABLED
Internal Link State Machine	DISABLED	DISABLED	DISABLED	DISABLED
Specific Comma	User Definec	User Definec	User Definec	User Definec
Plus Comma Value	1101010100	1101010100	1101010100	1101010100
Minus Comma Value	0010101011	0010101011	0010101011	0010101011
Comma Mask	0111111111	0111111111	0111111111	0111111111

Figure 6.4. Word_Alignment

To support a wider range of HDMI/DVI resolutions, the SERDES PLL settings need to be adjusted by configuring the data rate registers (CH_10[2:0] for Rx, QD_0D[2:0] for Tx) along with the other registers through the SCI interface. To do this, the SCI interface must be enabled, as shown in Figure 6.5. This figure also shows that the Reference Clock is passed to the FPGA core. Some of the sequential logic in the FPGA fabric needs to run at this clock domain.



Loopback Selection

Loopback Mode Ch0: DISABLED
Loopback Mode Ch1: DISABLED
Loopback Mode Ch2: DISABLED
Loopback Mode Ch3: DISABLED

Control Ports

Transmitter Electrical Idle □ Ch0 □ Ch1 □ Ch2 □ Ch3

Reset Sequence Generation

□ Include Reset Sequence Generation

Receive LOS Port: Ch0 Internal, Ch1 Internal, Ch2 Internal, Ch3 Internal

generate SCI interface to update SERDES configuration

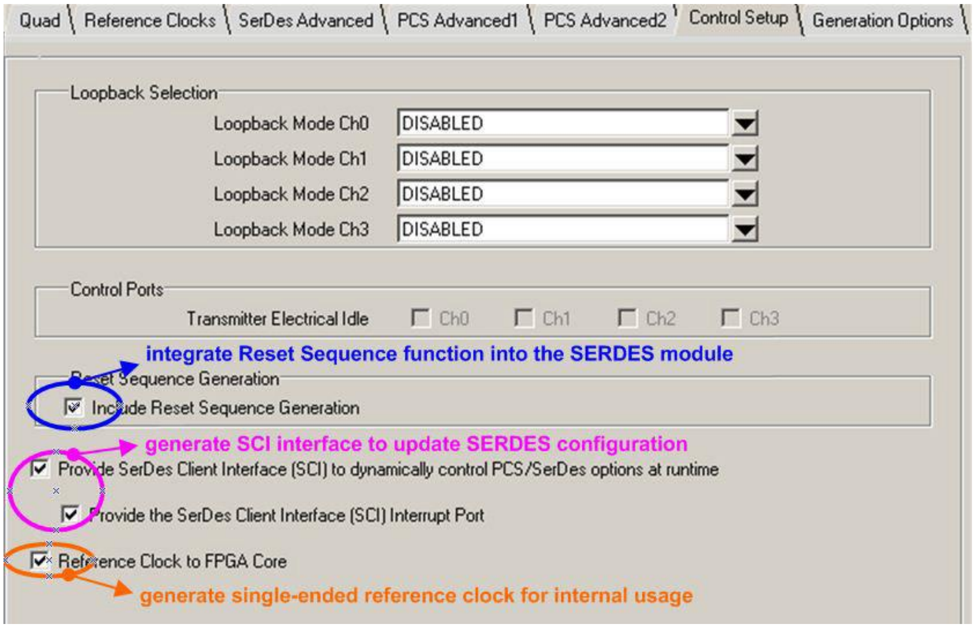
☑ Provide SerDes Client Interface (SCI) to dynamically control PCS/SerDes options at runtime

☑ Provide the SerDes Client Interface (SCI) Interrupt Port

☑ Reference Clock to FPGA Core

generate single-ended reference clock for internal usage

Figure 6.5. Control_Setup for ispLever 8.1 SP1



Loopback Selection

Loopback Mode Ch0: DISABLED
Loopback Mode Ch1: DISABLED
Loopback Mode Ch2: DISABLED
Loopback Mode Ch3: DISABLED

Control Ports

Transmitter Electrical Idle □ Ch0 □ Ch1 □ Ch2 □ Ch3

integrate Reset Sequence function into the SERDES module

Reset Sequence Generation

☑ Include Reset Sequence Generation

generate SCI interface to update SERDES configuration

☑ Provide SerDes Client Interface (SCI) to dynamically control PCS/SerDes options at runtime

☑ Provide the SerDes Client Interface (SCI) Interrupt Port

☑ Reference Clock to FPGA Core

generate single-ended reference clock for internal usage

Figure 6.6. Control_Setup for Diamond 1.1

6.1. Miscellaneous Functional Blocks

Reset Sequencing

After switching the DVI/HDMI source or changing the resolutions, the SERDES/PCS blocks as well as some functional blocks in the FPGA fabric must be reset following a recommended sequence as described in the corresponding section of LatticeECP3 SERDES/PCS Usage Guide (FPGA-TN-02190).

Loss_Of_Signal Detection

In order to support the HDMI/DVI cable's hot-plug capability, a special module is implemented to detect the Loss_Of_Signal condition over all three SERDES Receiver channels. If the TMDS data are presented on the SERDES Receiver channel again, this module will hold the LatticeECP3 device in a reset state for an extended period of time for the system to become stable and then release the reset signals following the sequence described above.

7. Hardware Validation

The HDMI/DVI Physical Layer Interface reference design has been implemented in a LatticeECP3 device and validated on hardware. This section discusses the evaluation board used for hardware validation and demonstration. It also shows the waveforms of internal signals captured by the Reveal™ on-chip logic analyzer as a proof of concept.

7.1. Evaluation Board

Figure 7.1 shows the LatticeECP3 Video Protocol Board with a HDMI Mezzanine Card snapped on top of it. The photo is overlaid with a block diagram of the HDMI/DVI Loopback Demo design.

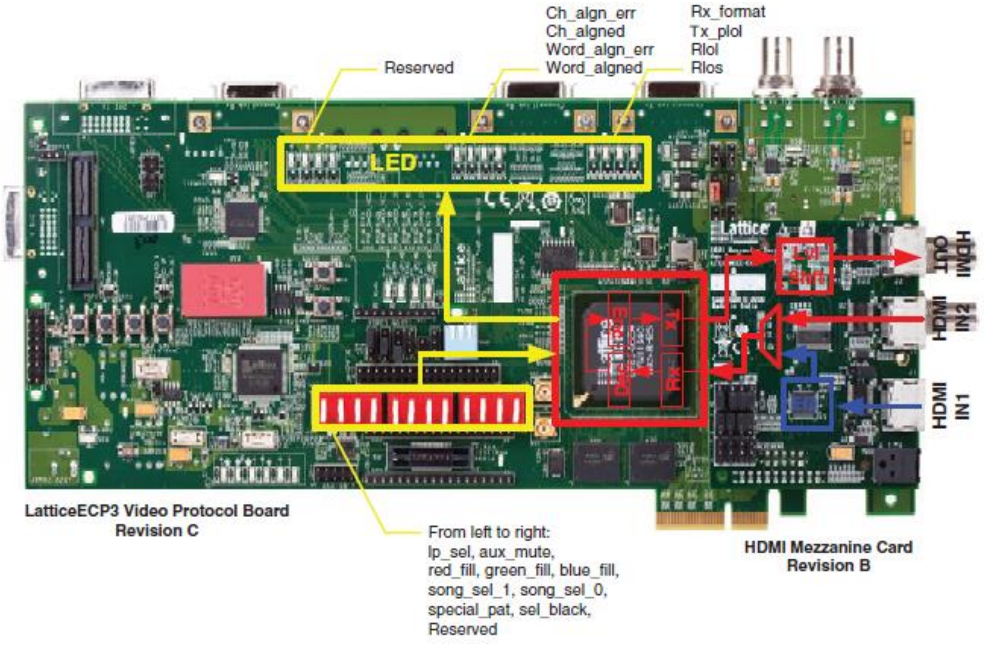


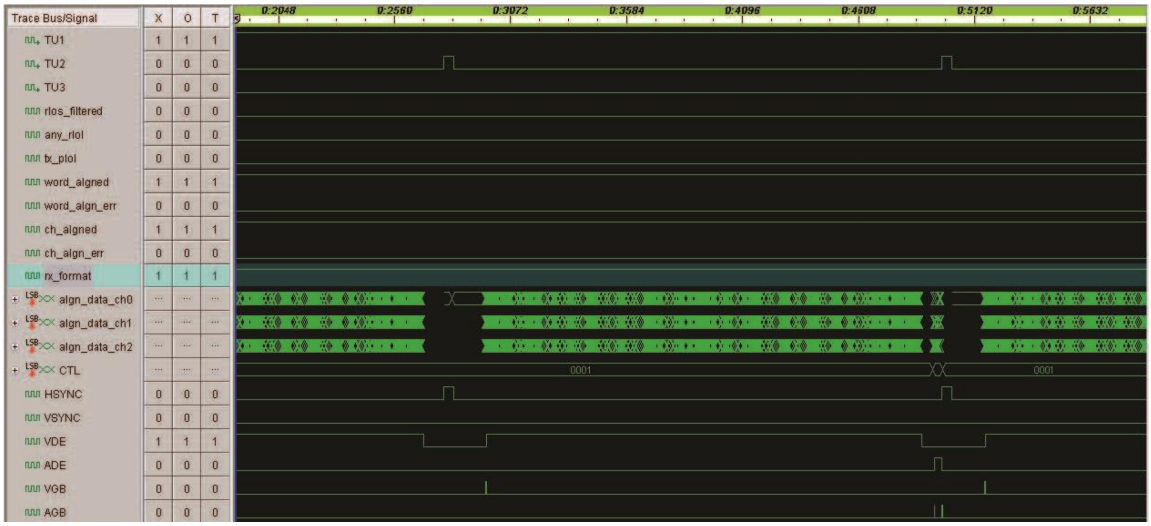
Figure 7.1. Hardware Demo Setup, Top View

The HDMI Mezzanine card is located at the lower right corner and attached to the LatticeECP3 Video Protocol Board through the Mezzanine Card Connector (Molex-75003-0005) and Control Signal Connector (Samtec-CES-110-01-S-D). The HDMI Mezzanine card performs as a level translator between the 3.3V DC-coupled TMDS and the AC-coupled SERDES CML interface. It also adds an extra level of ESD protection for both the HDMI Receiver and Transmitter. Two HDMI input connectors can be selected through a TMDS 2:1 Switch (Pericom PI3HDMI1210-A). The HDMI IN1 has a standalone equalizer (STDVE001); the HDMI IN2 is connected to a LatticeECP3 SERDES receiver through AC coupling capacitors. Refer to LatticeECP3 Video Protocol Board Revision C User's Guide (EB52) and HDMI Mezzanine Card Revision B User's Guide (EB55) for further information and schematics.

A HDMI/DVI Loopback Demo Design has been implemented using the HDMI/DVI Receiver and Transmitter modules discussed in this document. Refer to LatticeECP3 HDMI/DVI Loopback Demo User's Guide (UG36) for more details on the hardware setup and demo procedures.

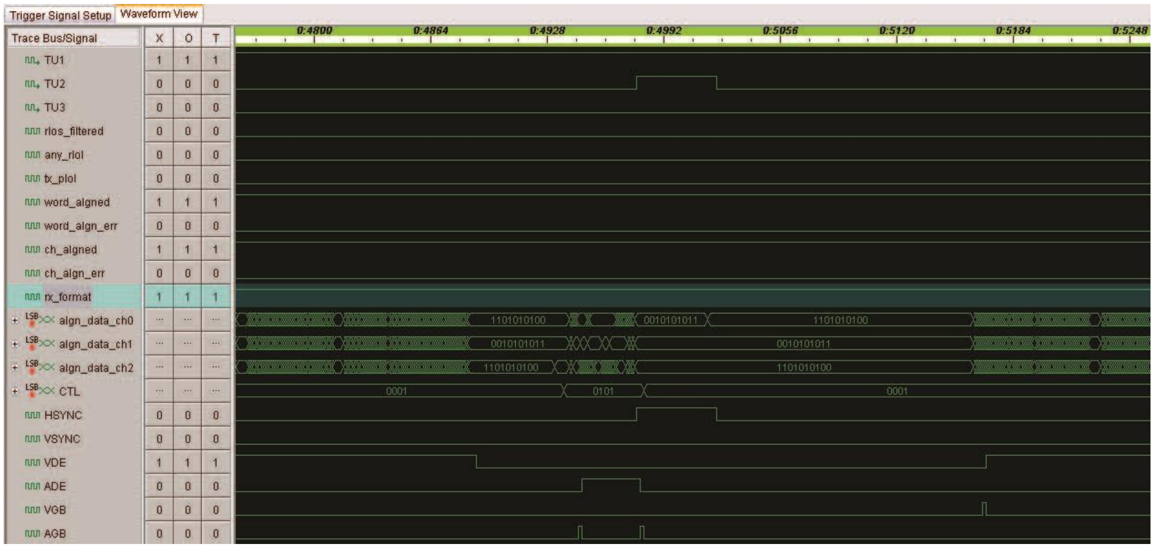
8. Reveal On-chip Logic Analysis

The Reveal software included in ispLEVER® is an on-chip debug tool which probes the design's internal signals and displays them in a Waveform Viewer graphical user interface. Figure 8.1 and Figure 8.2 show the waveforms of the aligned TMDS data as well as the decoded HSYNC, VSYNC, ADE and VDE signals.



Trace Bus/Signal	X	O	T
TU1	1	1	1
TU2	0	0	0
TU3	0	0	0
rlos_filtered	0	0	0
any_rlol	0	0	0
tx_plol	0	0	0
word_aligned	1	1	1
word_align_err	0	0	0
ch_aligned	1	1	1
ch_align_err	0	0	0
rx_format	1	1	1
align_data_ch0	...	...	...
align_data_ch1	...	...	...
align_data_ch2	...	...	...
CTL	...	...	...
HSYNC	0	0	0
VSYNC	0	0	0
VDE	1	1	1
ADE	0	0	0
VGB	0	0	0
AGB	0	0	0

Figure 8.1. Reveal Waveform Showing Two Blanking Periods



Trace Bus/Signal	X	O	T
TU1	1	1	1
TU2	0	0	0
TU3	0	0	0
rlos_filtered	0	0	0
any_rlol	0	0	0
tx_plol	0	0	0
word_aligned	1	1	1
word_align_err	0	0	0
ch_aligned	1	1	1
ch_align_err	0	0	0
rx_format	1	1	1
align_data_ch0	...	...	...
align_data_ch1	...	...	...
align_data_ch2	...	...	...
CTL	...	...	...
HSYNC	0	0	0
VSYNC	0	0	0
VDE	1	1	1
ADE	0	0	0
VGB	0	0	0
AGB	0	0	0

Figure 8.2. Enlarged Blanking Period with Data Island Packet

9. Hardware Test Results

Table 9.1 and Table 9.2 list the common the HDMI/DVI display modes that have been validated on the LatticeECP3 Video Protocol Board and HDMI Mezzanine Card with this reference design. The display modes marked with "TBD" require further testing to adjust the SERDES/PCS configuration settings for the best performance.

Table 9.1. HDMI (with Audio) Validation Results

Screen Display Mode	Pixel Frequency	Serial Data Rate	Validated
1920x1080p@59.94/60Hz	148.5 MHz	1.485 Gbps	Yes
1920x1080p@50Hz	148.5 MHz	1.485 Gbps	Yes
2280x480p@59.94/60Hz	108 MHz	1.08 Gbps	Yes
2280x576p@50Hz	108 MHz	1.08 Gbps	Yes
1920x1080i@59.94/60Hz	74.25 MHz	742.5 Mbps	Yes
1920x1080i@50Hz	74.25 MHz	742.5 Mbps	Yes
1280x720p@59.94/60Hz	74.25 MHz	742.5 Mbps	Yes
1280x720p@50Hz	74.25 MHz	742.5 Mbps	Yes
1920x1080p@23.98/24Hz	74.176/74.25	741.76/742.5	Yes
1440x480p@59.94/60Hz	54.054 MHz	540.54 Mbps	TBD
1440x576p@50Hz	54 MHz	540 Mbps	TBD
720x480p@59.94/60Hz	27.027 MHz	270.27 Mbps	Yes
720(1440)x480i@59.94/60Hz	27.027 MHz	270.27 Mbps	TBD
720x576p@50Hz	27 MHz	270 Mbps	TBD
720(1440)x576i@50Hz	27 MHz	270 Mbps	TBD
640x480p@59.94/60Hz	25.2 MHz	252 Mbps	Yes

Table 9.2. DVI Validation Results

DVI Display Mode	Pixel Frequency	Serial Data Rate	Validated
1920 x 1080@60 Hz	148.5 MHz	1.485 Gbps	Yes
1600 x 1200@60 Hz	162 MHz	1.62 Gbps	Yes
1680 x 1050@60 Hz	146.25 MHz	1.4625 Gbps	Yes
1400 x 1050@60 Hz	121.75 MHz	1.2175 Gbps	Yes
1400 x 900@60 Hz	106.5 MHz	1.065 Gbps	Yes
1280 x 1024@75 Hz	135 MHz	1.35 Gbps	Yes
1280 x 1024@60 Hz	108 MHz	1.08 Gbps	Yes
1024 x 768@75 Hz	78.5 MHz	785 Mbps	Yes
1024 x 768@60 Hz	65 MHz	650 Mbps	Yes
800 x 600@75 Hz	49.5 MHz	495 Mbps	TBD
640 x 480@60 Hz	25.2 MHz	252 Mbps	Yes

10. Design Limitations and Disclaimers

The goal of this design is to demonstrate a simple but effective solution to implement a HDMI/DVI physical layer interface using the LatticeECP3 device SERDES. It does not address all the required or optional aspects of the HDMI and DVI specifications. The TMDS transmitter and receiver are designed to support both HDMI and DVI links. If the application has only DVI links, they designer may choose to use a simpler DVI transmitter and receiver reference design implemented in the DVI/7:1 Video Conversion demo. Refer to LatticeECP3 DVI/7:1 LVDS Video Conversion Demo User's Guide (UG37), for the details of the design.

This reference design does not support HDCP encrypted contents, therefore any HDMI source from DVD or Bluray players will not be accepted.

This design does not support Data Island auto packaging and placement. For the purposes of the HDMI Loopback Demo, the ADE control signal for transmitting is derived from the receiver. Users can implement the control logic to generate the ADE signal based on the HDMI specification.

This design contains an emulated EDID ROM for DVI source to read the video formats that the DTV monitor is capable of receiving and rendering prior to establishing the DVI link. The HDMI Mezzanine Card offers the DDC/HPD/CEC bypass option to connect the HDMI source to the sink directly.

11. Implementation

Table 11.1. Performance and Resource Utilization

Device Family	Speed Grade	Module Name	Utilization (LUTs)	f_{MAX} (MHz)	Architecture Resources
LatticeECP3 [1]	–7	hdmi_receiver	1213	164	N/A
	–7	hdmi_transmitter	466	>165	N/A

Note:

1. Performance and utilization characteristics are generated using LFE3-95EA-7FN1156C with Lattice Diamond 3.3 design software.

References

- High-Definition Multimedia Interface Specification Version 1.3
- Digital Visual Interface DVI Revision 1.0
- LatticeECP3 HDMI/DVI Loopback Demo User's Guide (UG36)
- LatticeECP3 DVI/7:1 LVDS Video Conversion Demo User's Guide (UG37)
- LatticeECP3 Video Protocol Board Revision C User's Guide (EB52)
- HDMI Mezzanine Card Revision B User's Guide (EB55)

Technical Support Assistance

Submit a technical support case through www.latticesemi.com/techsupport.

Revision History

Revision 1.6, January 2020

Section	Change Summary
All	• Changed document number from RD1097 to FPGA-RD-02139. • Updated document template.
Disclaimers	Added this section.

Revision 1.5, April 2015

Section	Change Summary
Implementation	Updated Implementation section. Updated Table 11.1, Performance and Resource Utilization. • Updated LUTs values. • Added support for Lattice Diamond 3.3 design software. • Removed support for ECP5 device family.

Revision 1.4, March 2014

Section	Change Summary
Features	Updated Features section. Added support for 48 kHz PCM linear audio sample.
Hardware Validation	Updated Figure 7.1, Hardware Demo Setup, Top View. Revised callout.
Hardware Test Results	Updated Table 9.2, HDMI (with Audio) Validation Results.
Implementation	Updated Table 11.1, Performance and Resource Utilization. • Added support for ECP5 device family. • Added support for Lattice Diamond 3.1 design software. • Added support for ECP5 device family. • Updated Table 11.1, DVI Validation Results. Extended support for lower resolution of 640x480@60 Hz.
All	• Added TN1265, reference. • Updated corporate logo. • Removed “Receiver’s SERDES/PCS Configuration” text section. • Removed “Transmitter’s SERDES/PCS Configuration” text section. • Added “SERDES/PCS Configuration” text section. • Added built-in pattern generation with 1080P60 video and 48 kHz audio.
Technical Support Assistance	Updated Technical Support Assistance information.

Revision 1.3, April 2011

Section	Change Summary
Implementation	Added Implementation table.

Revision 1.2, January 2011

Section	Change Summary
All	Updated for Lattice Diamond 1.1 support.

Revision 1.1, December 2010

Section	Change Summary
What is TMDS?	Added AC Coupling Circuit figure.

Revision 1.0, September 2010

Section	Change Summary
All	Initial release.

www.latticesemi.com