MachXO2, MachXO3 and ECP5 7:1 LVDS Video Interface

Reference Design

FPGA-RD-02093-1.5

December 2019

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

Figures

Tables

1. Introduction

Source synchronous interfaces consisting of multiple data bits and clocks have become a common method for moving image data within electronic systems. A prevalent standard is the 7:1 LVDS video interface (employed in Channel Link, Flat Link, and Camera Link), which has become a common standard in many electronic products including consumer devices, industrial control, medical, and automotive telematics. In many of these applications, the practice of using low-cost PLDs for image processing has become quite common. The MachXO2™, MachXO3™ PLD and ECP5™ device families have been specifically engineered to support Display Interface (7:1 LVDS) video standard with built-in dedicated hardware interface blocks. This document describes the implementation methods and the advantages of using MachXO2, MachXO3 and ECP5 devices for implementing this interface. By extension, support for Display Interface in these devices proves the feasibility of hardware implementation for all other LVDS source synchronous requirements as well.

Two designs are included in the discussion of this document. The first design is a simple loopback test that illustrates the use of the Display Interface transmitter and Display Interface receiver. The second design is an example that brings video data into the PLD through the Display Interface receiver, processes it and transmits it out via the Display Interface transmitter. Both designs are verified using the MachXO2 Control Evaluation Board.

2. Display Interface Requirement

The Display Interface is a source synchronous LVDS interface. Seven data bits are serialized for each cycle of the low-speed clock as shown in Figure 2.1. Typically, the interface consists of four (three data, one clock) or five (four data, one clock) LVDS pairs. The four pairs translate to 21 parallel data bits and five pairs translate to 28 parallel data bits. Note that there is a 2-bit offset between the clock rising edge and the word boundary. Each word is 7 bits long.

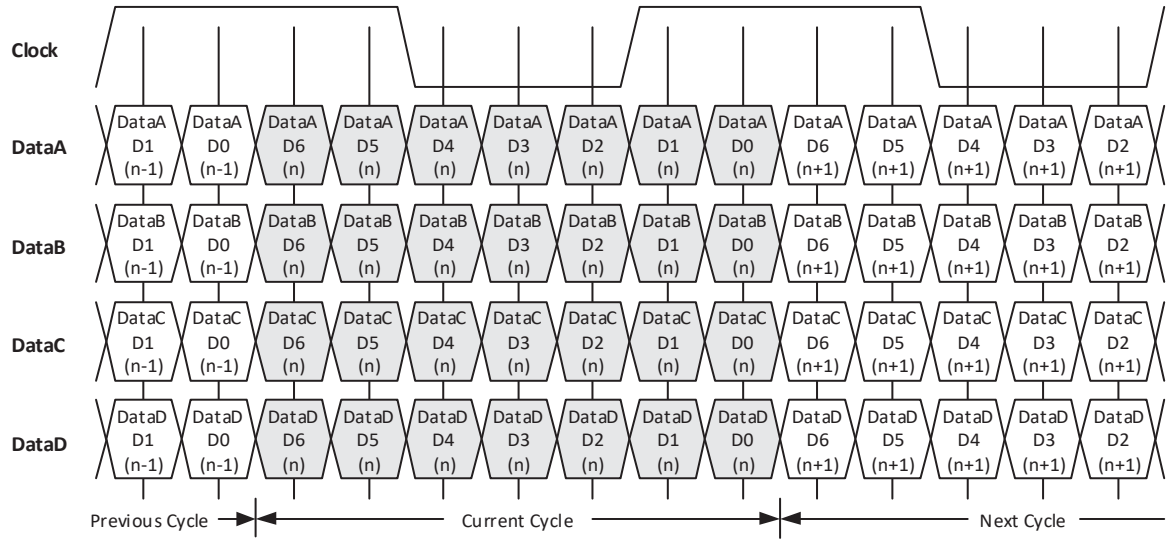

Figure 2.1. Basic Timing of the Display Interface

Each channel includes a serial LVDS data pair along with a source synchronous LVDS clock pair. The receiver receives this serial LVDS data, deserializes it and aligns it to the original word boundary to generate seven parallel data bits. The 7:1 transmitter serializes the seven parallel data bits to a single LVDS data bit and transmits this serial data channel along with a LVDS clock.

Figure 2.2. shows the Display Interface receiver receiving four LVDS data channels. When deserialized, it generates 28-bit wide parallel data. Similarly, the Display Interface transmitter serializes 28-bit parallel data to generate four LVDS data channels.

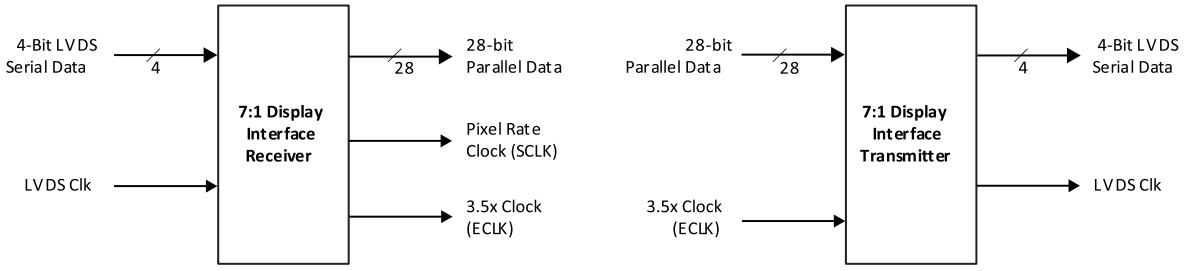

Figure 2.2. Display Interface Receiver and Transmitter Function

The requirements for an FPGA-based solution to the Channel Link and Flat Link style interfaces consist of four key components: high-speed LVDS buffers, a PLL for generating the de-serialization clock, input data capture and gearing, and data formatting. In previous devices, input data capture, gearing and formatting required user logic design and qualification. MachXO2, MachXO3 and ECP5 devices simplify this task with built-in hardware transmit and receive gearing to compliment the high-speed LVDS buffers and sysCLOCK™ PLLs to provide a full featured, pre-qualified Display Interface solution.

The data and clock are received or transmitted in LVDS format, with the data at relatively high speed. The exact speed depends on the resolution, frame rate and color depth used by the display. For example, 800x600 to 1024x768 displays require pixel data to be transmitted from 40 MHz to 78.5 MHz for 60 Hz to 75 Hz refresh rates. This translates to LVDS data rates of 280 Mbps to 549 Mbps. Higher resolution displays, such as the 1280x1024 60 Hz require pixel data to be transmitted at 108 MHz. For these systems, the LVDS data will transmit at 756 Mbps.

2.1. Clock Generation

In MachXO2, MachXO3 and ECP5 devices, the Display Interface input capture circuitry uses Double Data Rate (DDR) registers with data captured on both the rising and falling edges of the clock. When operating as a receiver, the low-speed clock that is provided with the data must be multiplied 3.5X in order to capture the data on both clock edges. The multiplied clock must have relatively low jitter since its jitter must be accounted for in the overall timing budget. Similarly, the skew of the clock distribution network used to provide this clock to input or output registers must be accounted for in any timing analysis.

In order to transmit high-speed data, a transmitter similarly must use or create the 3.5X high-speed DDR edge clock. Again, the jitter of the clock and the skew of its distribution are important as they impact the timing budget for the interface. Figure 2.3. shows the DDR transmit clock and how the R, G, B bits, Vsync, Hsync, and DE of a pixel on line 2 of a video frame get assigned to the four LVDS data pairs. The data bits are sampled on both rising and falling edges of the DDR transmit clock (ECLK).

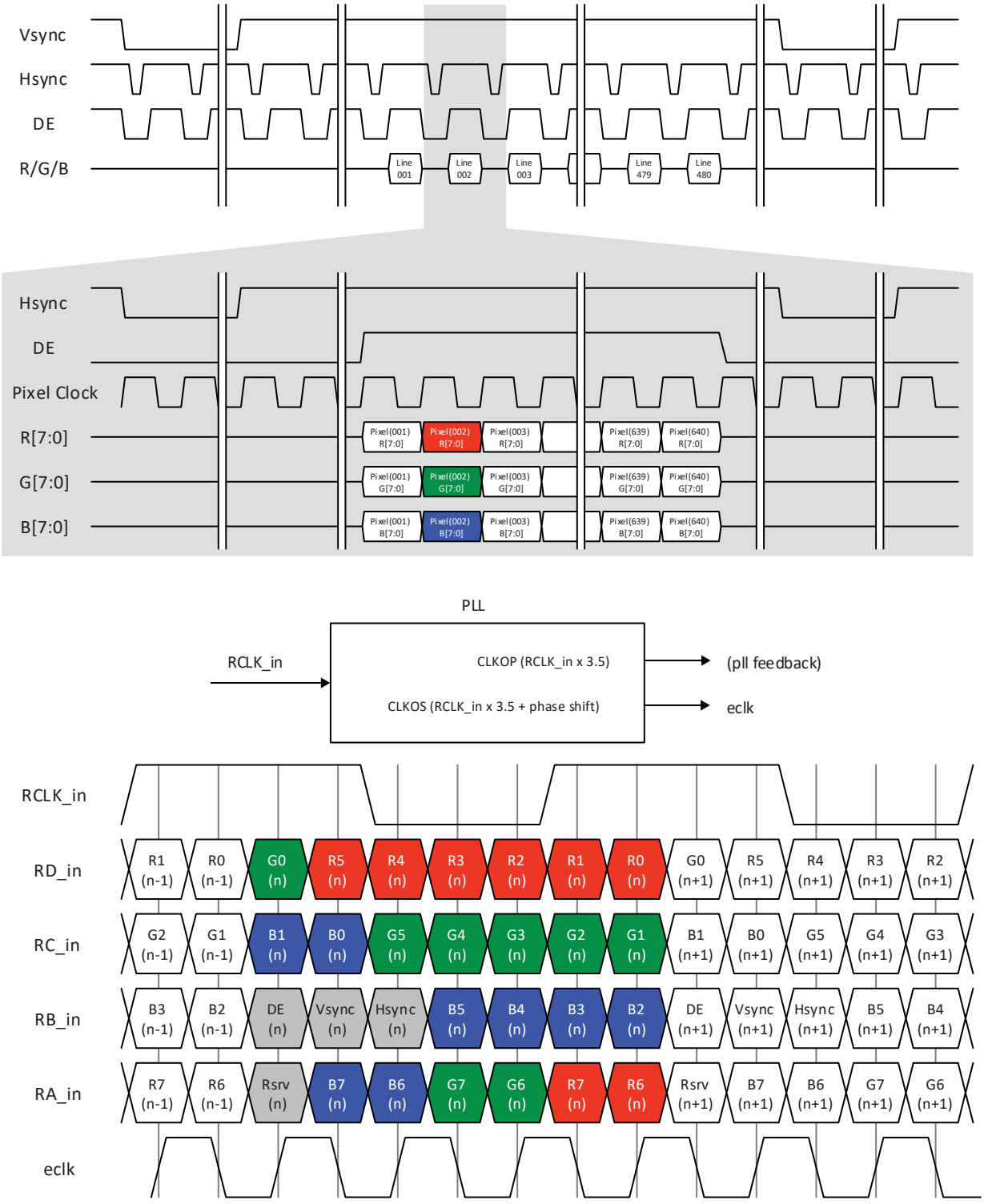

Figure 2.3. Timings of Video Signals and the Display Interface

3. Data Capture and Formatting

The registers that follow the LVDS input buffer must accurately capture the data. A tight control of the clock and data relationship is important to capture the incoming high-speed data stream. It is also necessary to gear, or reduce, the speed of the data before it is passed on to the FPGA fabric. MachXO2 devices specify the operation of individual circuit elements to around 350 MHz. A practical operating frequency with a reasonable amount of logic is 150-200 MHz. Therefore, the greater the gearing that can be done in the I/O structure, the lower the likelihood that the FPGA fabric will be the limit on overall performance. A similar discussion is applicable to the transmit path. The final step is to take the data from the I/O cells and format it into the original 7-bit width clocked by the low-speed clock.

In MachXO2 and ECP5 devices, the task of DDR data capture, gearing and 7:1 formatting has been greatly simplified. The architecture includes embedded Display Interface primitives for both receive and transmit data paths. These elements provide an ideal solution for this low-cost, high-confidence video interface solutions.

4. MachXO2 Display Interface

The MachXO2 architecture provides an ideal solution for this interface. This section describes implementation of the Display Interface receiver and Display Interface transmitter utilizing the specialized MachXO2 Display I/O structures. Similar architecture has been implemented in the ECP5 designs.

4.1. Display Interface Receiver

Figure 4.1. shows the block diagram of the receive side of an intra-system display interface within a MachXO2 device. The receiver receives four LVDS data channels and one LVDS clock.

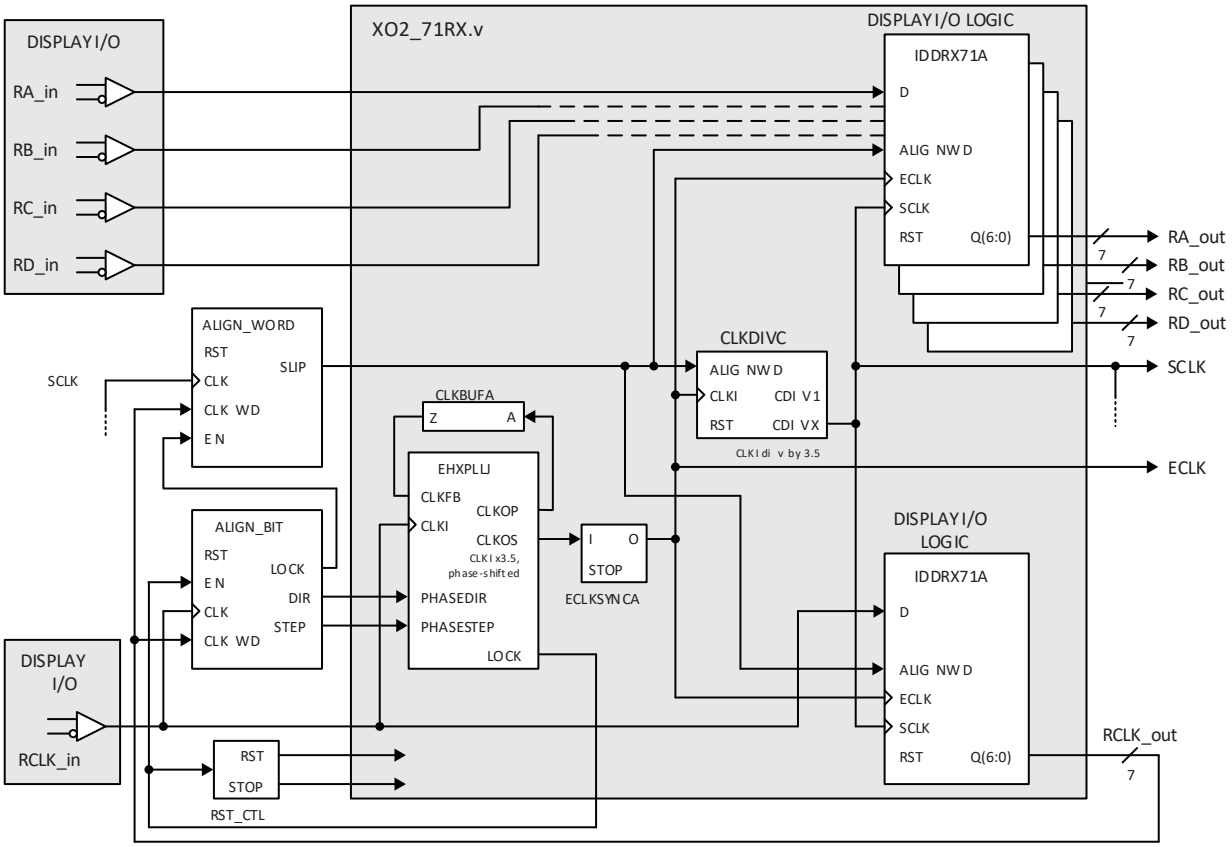

Figure 4.1. Display Interface Receiver

The data and clock enter the MachXO2 device through Display I/O buffers. These buffers operate at up to 303 MHz (606 Mbps), supporting high resolution and display refresh rates with up to a 85 MHz pixel rate (SXGA).

The source synchronous LVDS input clock is fed into a PLL. The PLL is used to multiply the clock by 3.5 and create a phase shift (nominally 90 degrees). This phase shift allows for placing the clock in the middle of the data valid window. This faster phase-shifted clock (ECLK) is then distributed via a low skew edge clock net to the DDR capture registers. An additional block (ECLKSYNCA) is used in conjunction with the RST_CTL block to ensure a controlled startup alignment of all subtended divider circuits driven by ECLK. The start-up timing is shown in Figure 4.2.

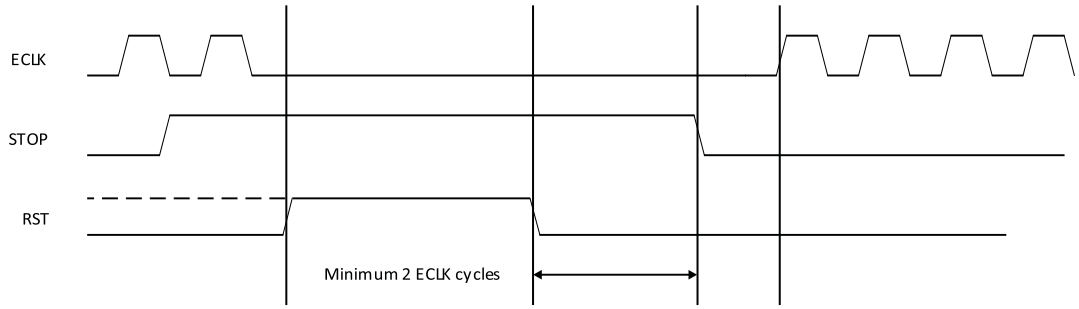

Figure 4.2. Synchronization Start-up Timing (ECLKSYNCA)

The output of the synchronizer block drives a dedicated div-by-3.5 clock divider circuit (CLKDIVC) to produce a pixel-rate clock (SCLK) phase aligned with ECLK. The pixel clock is used to clock parallel pixel data at a lower FPGA clock rate into the FPGA fabric.

The LVDS data is fed to the Display I/O Logic Cell's double data-rate (DDR) registers with 7:1 gearing function (IDDRX71A). The gearing allows demuxing of the I/O data clocked with the high-speed edge clock (ECLK) to the slower-speed FPGA clock rate (SCLK). As shown in Figure 4.3., the output data is driven by the rising edge of SCLK. The first serial bit received becomes bit '0' of the output word.

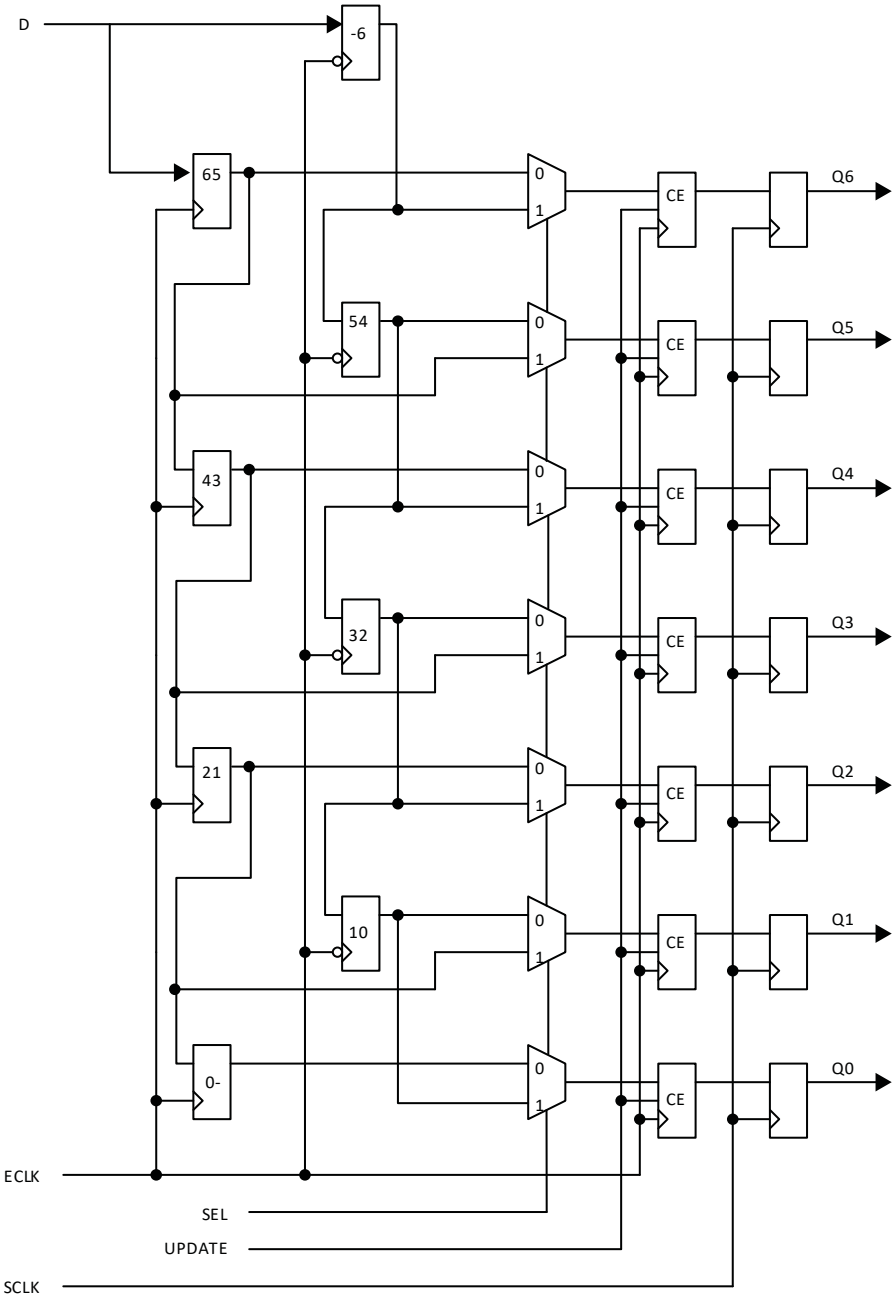

Figure 4.3. Receive 7:1 Gearing in Display I/O Logic Cell (IDDRX71A)

The output word is delineated using two control signals, SEL and UPDATE. These signals are generated internally to the I/O logic cell by dividing down ECLK. The timing of these signals is shown in Figure 4.4.

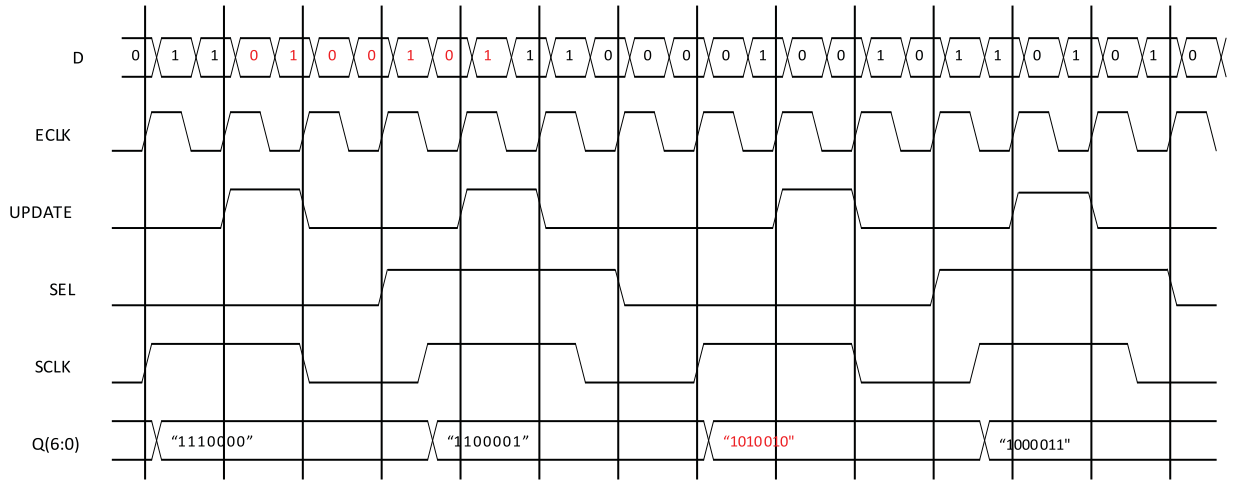

Figure 4.4. Deserializer Timing

The Display Interface includes logic for auto-aligning the PLL output clock to the optimum sample position for sampling the input LVDS data stream (bit_align.v), plus logic for auto-aligning the FPGA clock to the input data word (word_align.v). These 'soft' logic pieces work in concert with the 'hard' primitive resources to provide the full Display Interface solution.

After a global reset event, or the establishment of the input Clock link, the Bit Alignment module (bit_align.v) is activated to find the best sample point for the incoming data. The module slews the PLL clock output across 180° of phase in 8 steps, or 22.5° increments. (180° is sufficient because of the Dual Data Rate nature of the input data.) During this initialization, the circuit tests each phase for coherent data. After determining the extent of the good data, the module selects the phase in the middle of the good data window and indicates initialization complete by asserting 'DPHASE_LOCK'. This process is illustrated in Figure 4.5.

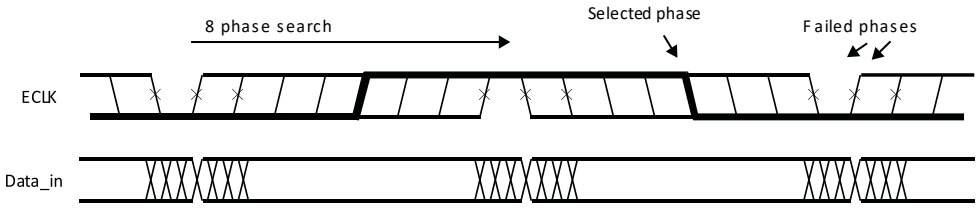

Figure 4.5. Bit-alignment Search

The Word Alignment module (word_align.v) simply compares RCLK_out against the expected output pattern of "1100011". If the expected alignment is not found, the module issues asserts ALIGNWD to CLKDIVC and IDDRX71A to accomplish a shift in UPDATE, SEL, SCLK and, ultimately, the received data. Each rising edge of ALIGNWD causes a 2-bit shift, or 'slip', relative to the input stream. (ALIGNWD may be asynchronous to the ECLK domain. Both high and low pulses must remain for a minimum of 4 ECLK cycles.) This action is repeated until the correct data word alignment is found. Figure 4.6. shows the effect of a 'slip' action.

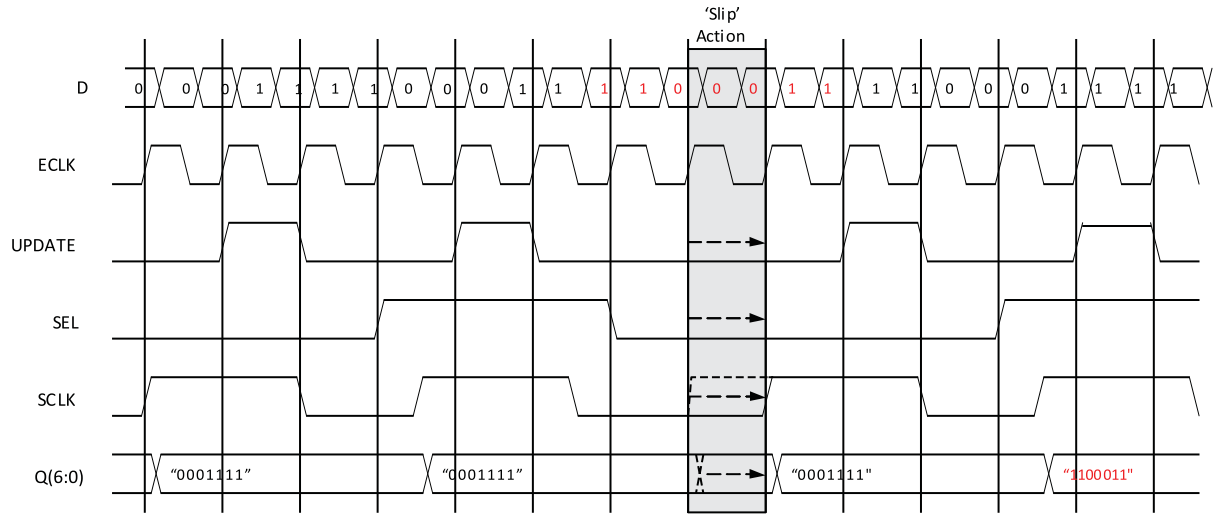

Figure 4.6. 'Slip' Action in Response to ALIGNWD Rising Edge

4.2. Display Interface Transmitter

Figure 4.7. shows the block diagram of the transmit side of an intra-system display interface within a MachXO2 device. The module receives four channels of 7-bit parallel data and the fast DDR clock (ECLK). The transmitter transmits four LVDS data channels and one LVDS clock.

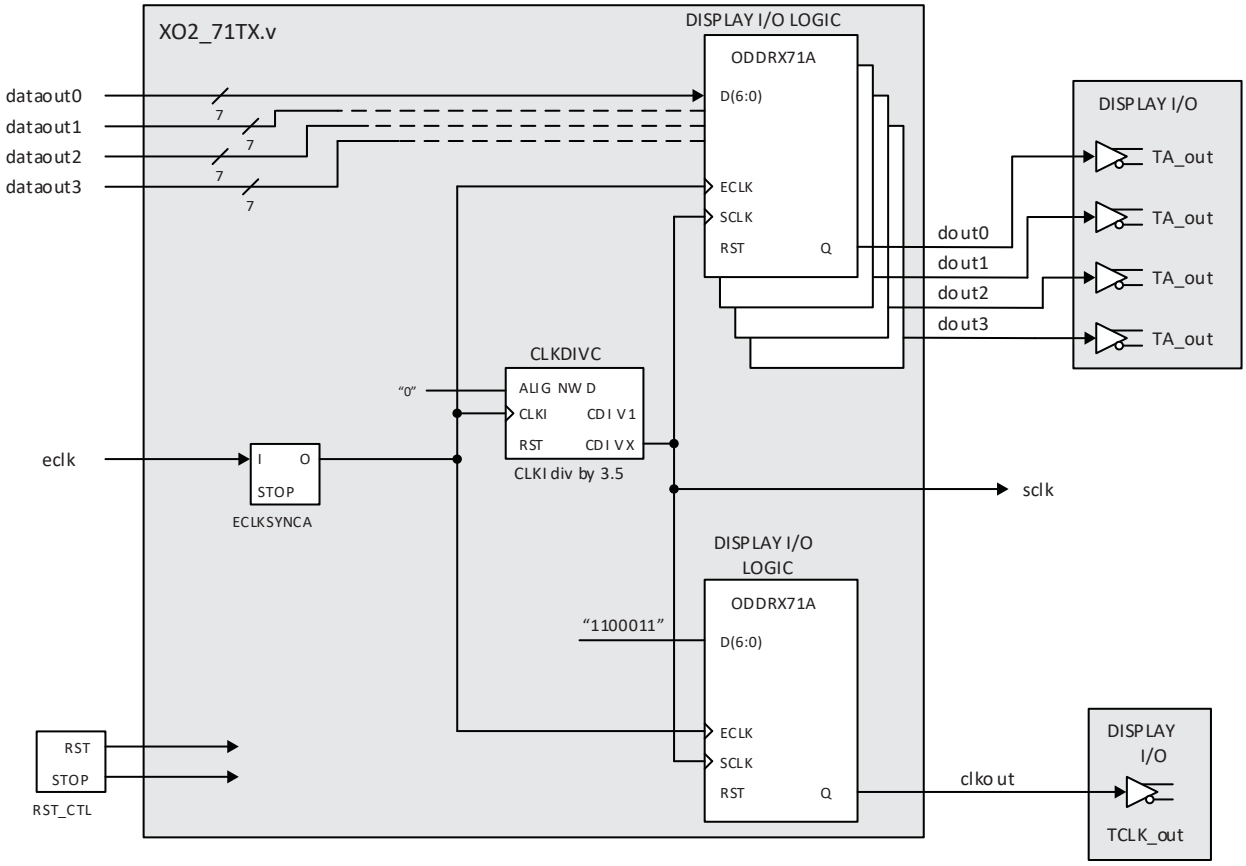

Figure 4.7. Display Interface Transmitter

All 28 bits of parallel data are registered at the transmitter inputs on the rising edge of the locally generated system clock (SCLK). Care must be taken as the parallel data is sourced from a separate clock domain. For example, data from the Display I/O receiver module, generated by a separate and distinct SCLK domain, must be phase aligned with the SCLK generated locally by CLKDIVC within the transmit module. This is possible by releasing the STOP port of each ECLKSYNCA (rcv and xmt) in unison. Timing analysis can then calculate the correct set-up times based upon data path delay and clock skews from the common ECLK.

The parallel data is fed to the Display I/O Logic Cell with 7:1 gearing function (ODDRX71A). The gearing allows the multiplexing of the input data clocked in with the low-speed system clock (SCLK) to the higher-speed DDR output edge clock rate (ECLK). As shown in Figure 4.8., the output data is driven by both edges of ECLK. Bit '0' of the input data word becomes the first bit of the serial output.

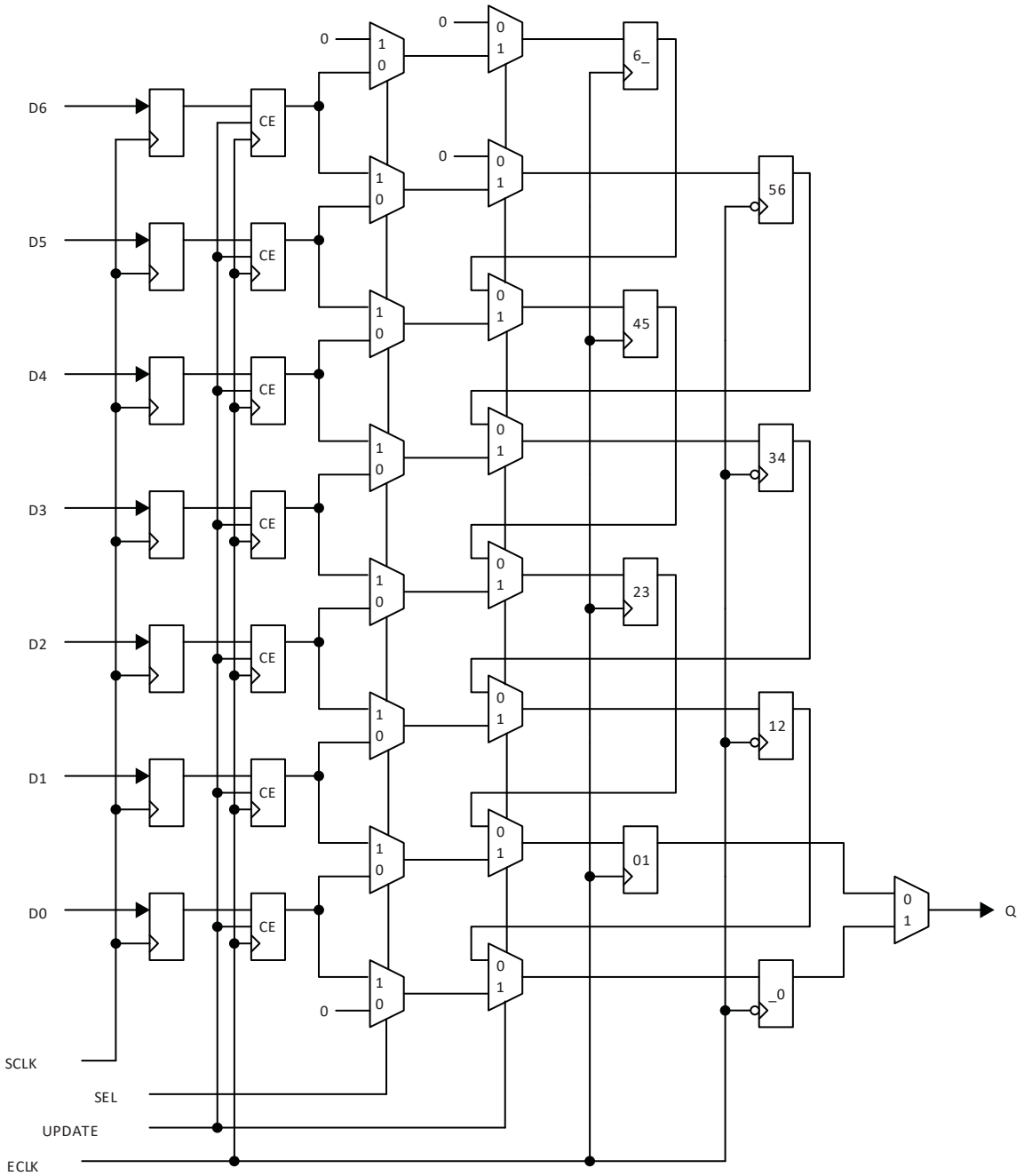

Figure 4.8. 7:1 Gearing in Tx Display I/O Logic Cell (ODDRX71A)

The output word is serialized using two control signals, SEL and UPDATE. These signals are generated internally to the I/O Logic Cell by dividing down ECLK. The timing of these signals is illustrated in Figure 4.9. SEL and UPDATE are synchronized with SCLK (generated by CLKDIVC) by the use of ECLKSYNCA.

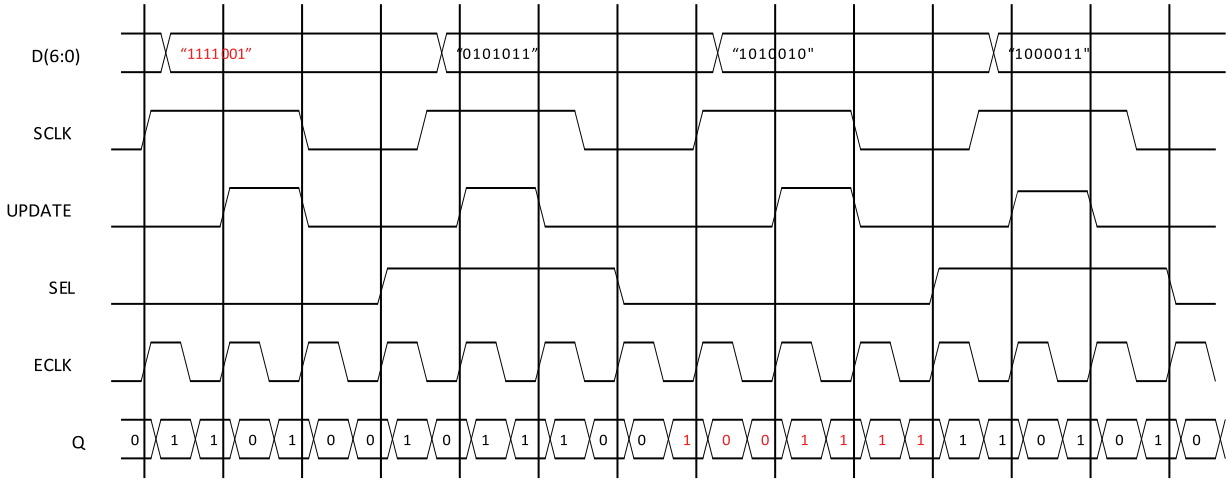

Figure 4.9. Serializer Timing

5. Design Example 1: Loopback Test

The loopback test design included with this document uses a MachXO2, MachXO3 or ECP5 device to implement both the 7:1 Display Interface transmitter and receiver. Figure 5.1. shows the design implementation. For more detailed information about the 7:1 Display Interface transmitter and receiver, refer to Figure 4.6. and Figure 4.7.

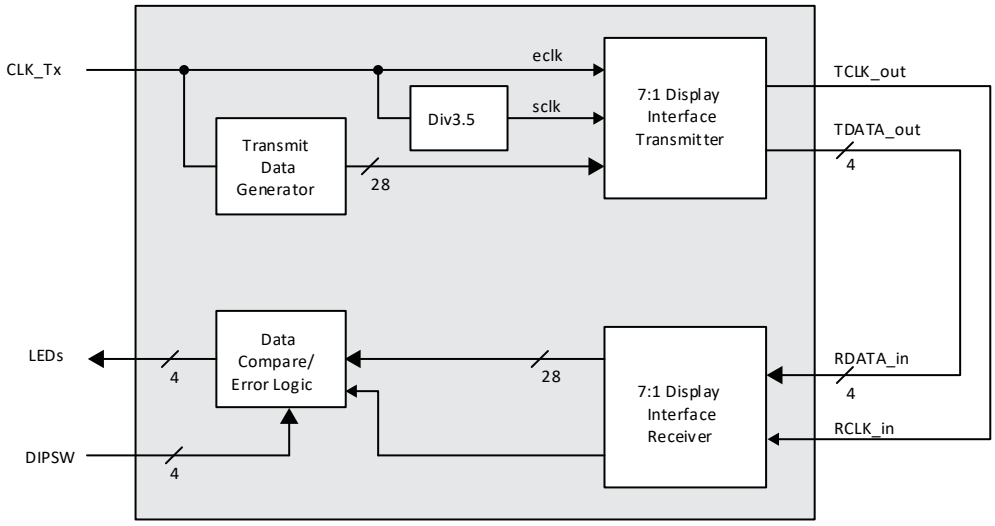

Figure 5.1. Loopback Test Block Diagram

28-bit transmit data is generated in the device logic using counter values. This data is then serialized and transmitted as four bits of LVDS data using the 7:1 Display Interface Transmitter. The 4-bit LVDS data is then looped back externally into the MachXO2 device receiver side and deserialized using the 7:1 Display Interface Receiver. This deserialized data is then fed to the data compare logic module which compares the deserialized receiver data to the original counter values transmitted. Errors are detected and latched for visual display through an LED port. The information displayed by the LEDs is selected by a DIPSW array input port. In addition, the DIPSW may be used to override the automated

ECLK phase detection (see Bit Alignment search, above). The relationship between the DIPSW setting and the LED display is given in Table 5.1.

Table 5.1. LED Display Options

DIPSW(3:0)	LED3	LED2	LED1	LED0
0000	0	Bit-alignment module phase determination (0-7)		
0001	Ch. A error detect	Ch. B error detect	Ch. C error detect	Ch. D error detect
0010	1	Rx PLL lock	Bit-align lock	Rx data lock
0011	Free-running “1010” – “0101” blink			
0100	Ch. A Tx data(6:5)	Ch. A Rx data(6:5)		
0101	Ch. B Tx data(6:5)	Ch. B Rx data(6:5)		
0110	Ch. C Tx data(6:5)	Ch. C Rx data(6:5)		
0111	Ch. D Tx data(6:5)	Ch. D Rx data(6:5)		
1xxx	Ch. A error detect	Ch. B error detect	Ch. C error detect	Ch. D error detect

Table 5.2. Input Sample Clock (ECLK) Phase Adjustment

DIPSW(3)	Action	
0	Automatic input clock sample phase adjustment	
1	Manual input clock sample phase adjustment:	
	DIPSW(2:0)	**Phase Adjustment from Nominal**
	000	+0°
	001	+22.5°
	010	+45°
	011	+67.5°
	100	+90°
	101	+112.5°
	110	+135°
	111	+157.5°

5.1. Implementation

Table 5.3. Performance and Resource Utilization[1]

Device Family	Language	Speed Grade	Utilization (LUTs)	fmax (MHz)	I/Os
ECP5 [3]	Verilog-LSE	–8	421	108	31
	Verilog-Syn	–8	426	108	31
	VHDL-LSE	–8	426	108	31
	VHDL-Syn	–8	421	108	31
MachXO3L [2]	Verilog-Syn	–6	352	85	31
	Verilog-LSE	–6	390	85	31
	VHDL-Syn	–6	354	85	31
	VHDL-LSE	–6	413	85	31
MachXO2 [1]	Verilog-LSE	–6	390	85	31
	Verilog-Syn	–6	352	85	31
	VHDL-LSE	–6	—	—	31
	VHDL-Syn	–6	—	—	31

Notes:

1. Performance and utilization characteristics are generated using LCMXO2-1200HC-6MG132C with Lattice Diamond® 3.4 design software. When using this design in a different device, density, speed, or grade, performance and utilization may vary.
2. Performance and utilization characteristics are generated using LCMXO3L-4300C-6BG256C with Lattice Diamond 3.4 design software. When using this design in a different device, density, speed, or grade, performance and utilization may vary.
3. Performance and utilization characteristics are generated using LFE5UM-85F-8BG765CES with Lattice Diamond 3.4 design software. When using this design in a different device, density, speed, or grade, performance and utilization may vary.

6. Design Example 2: PassThru Test

In order to verify the operation of Display Interfaces within the MachXO2 device, Lattice has developed the MachXO2 Control Evaluation Board. This system takes video data supplied in DVI format from a source such as a PC or a DVD player and converts it to the 7:1 LVDS source synchronous format using a Texas Instruments DVI Decoder and a National Semiconductor Channel Link Transmitter Device. This image data is fed to the MachXO2 where the Display Interface Receiver is used to deserialize the data. This data is then converted back into serial data using the Display Interface Transmitter within the MachXO2 device. It is then transmitted using a source synchronous 7:1 LVDS interface to a National Semiconductor Channel Link Receiver device and Texas Instruments DVI Encoder, then ultimately to a display. A simplified block diagram of the platform is shown in Figure 6.1.

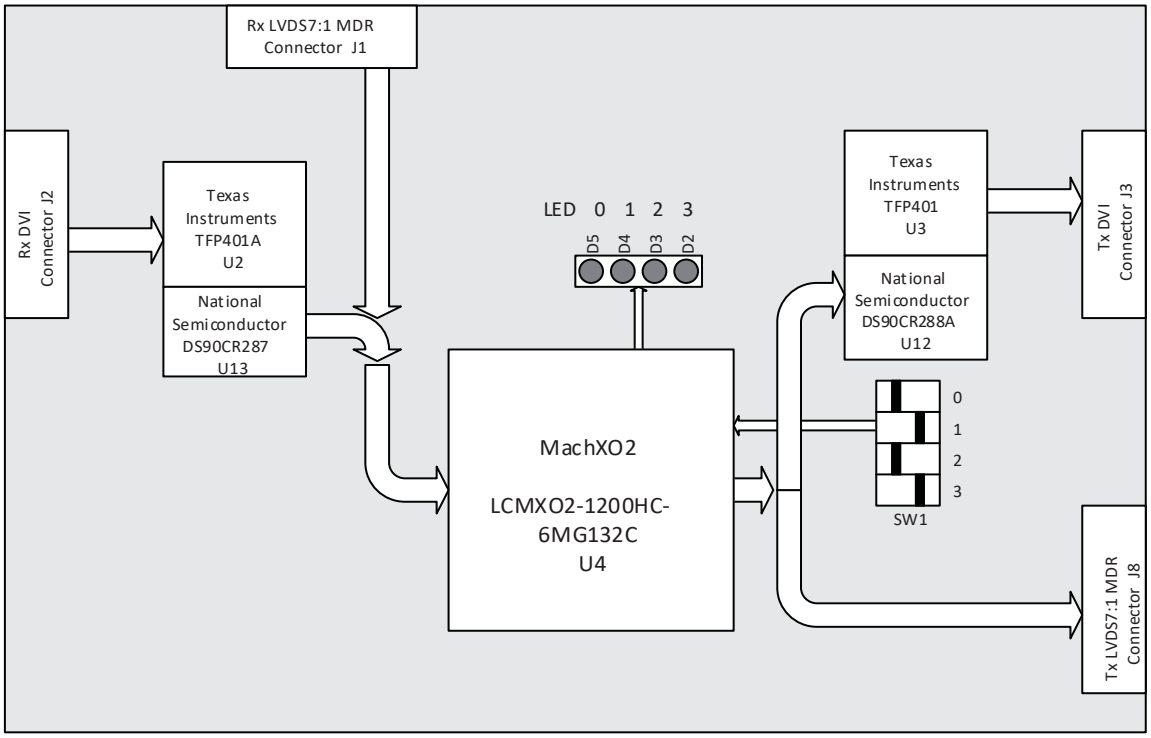

Figure 6.1. MachXO2 Control Board Simplified Block Diagram

Figure 6.2. shows a block diagram of the MachXO2 PassThru design. Other than the receiver and transmitter modules, the center logic block can be any customized video processing design. For demonstration purposes, the design shown in Figure 6.2. was created to include the following features:

- Independent control over red, green and blue channels
- Individual channel pass, suppress, invert, and 'rotate'
- Simple color bars insertion

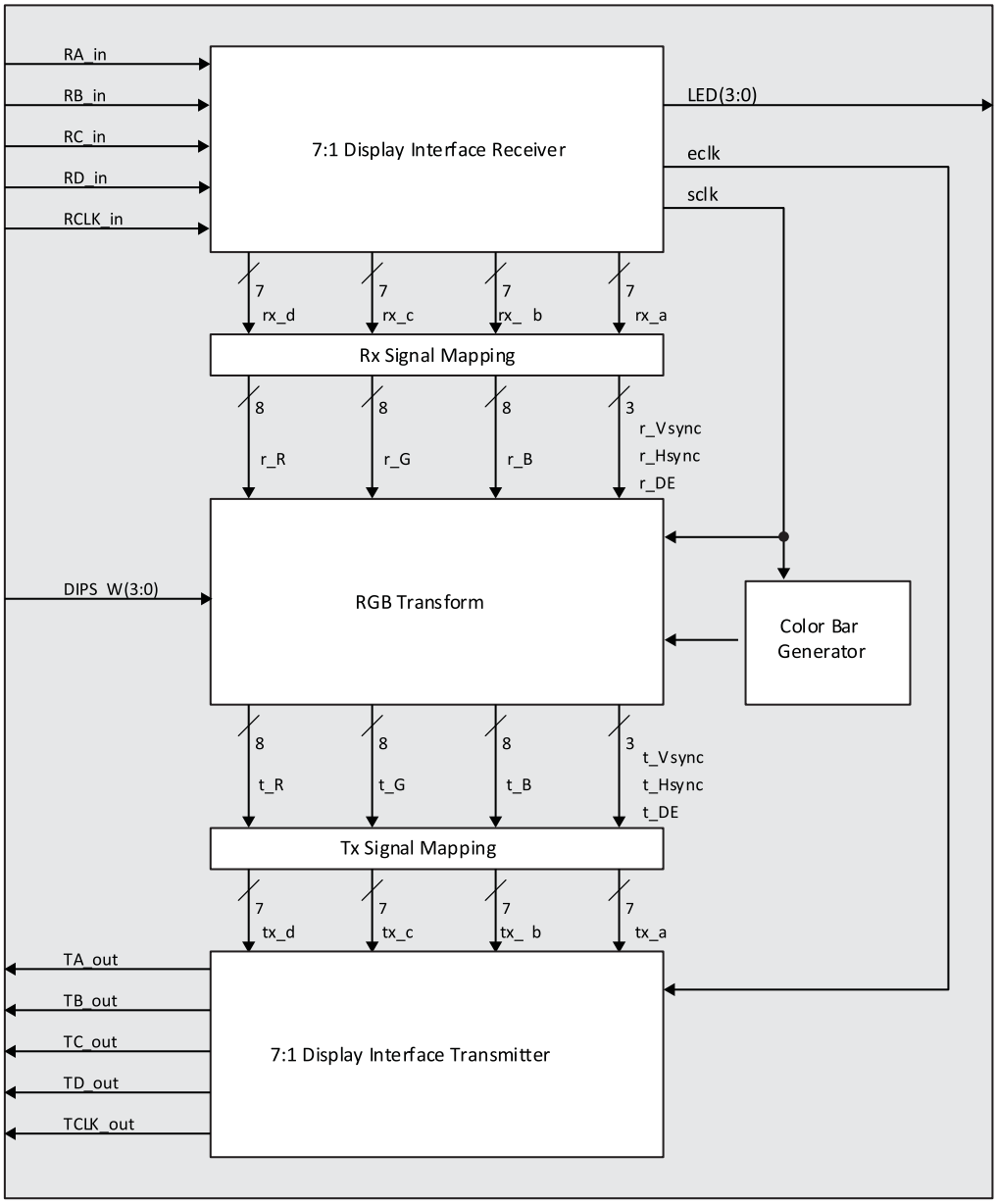

Figure 6.2. PassThru Test Block Diagram

The PassThru design example includes four sub-modules: Receiver, RGB Transform, Color Bars Generator, and Transmitter. On the MachXO2 Control Evaluation Board, the 4-position DIP-switch SW1 is used for manipulating the RGB channels. The functions of SW1 are listed in Table 6.1.

Table 6.1. RGB Transform Control

DIPSW(3:0) SW1[1]	Red Channel Output	Green Channel Output	Blue Channel Output
0000	Inverted Red	Inverted Green	Inverted Blue
0001	→Blue	→Red	→Green
0010	Green←	Blue←	Red←
0011	PassThru		
0100	Insert Colorbars		
0101	PassThru		
0110	PassThru		
0111	PassThru		
1000	Suppressed	Suppressed	Suppressed
1001	Suppressed	Suppressed	Blue
1010	Suppressed	Green	Suppressed
1011	Suppressed	Green	Blue
1100	Red	Suppressed	Suppressed
1101	Red	Suppressed	Blue
1110	Red	Green	Suppressed
1111	PassThru		

Note: '0' and '1' translate to 'off' and 'on', respectively, in the physical switch.

When selected, the Color Bar Generation module overwrites the active video frame with a color bar pattern. The blanking data is passed unchanged. The active video line is divided into eight columns. The color sequence is: White, Yellow, Cyan, Green, Magenta, Red, Blue, Black.

Table 6.2. describes the LED diplay function.

Table 6.2. LED Display

LED3	LED2	LED1	LED0
On: Rx Bit-align locked	Bit-alignment module phase determination (0-7)		

7. Implementation

Table 7.1. Performance and Resource Utilization[1]

Device Family	Language	Speed Grade	Utilization	f_{MAX} (MHz)	I/Os
ECP5 [3]	Verilog-LSE	–8	379	108	29
	Verilog-Syn	–8	399	108	29
	VHDL-LSE	–8	379	108	29
	VHDL-Syn	–8	419	108	29
MachXO3L [2]	Verilog-Syn	–6	327	85	29
	Verilog-LSE	–6	340	85	29
	VHDL-Syn	–6	327	85	29
	VHDL-LSE	–6	340	85	29
MachXO2 [1]	Verilog-LSE	–6	364	85	29
	Verilog-Syn	–6	327	85	29
	VHDL-LSE	–6	413	85	29
	VHDL-Syn	–6	354	85	29

Notes:

1. Performance and utilization characteristics are generated using LCMXO2-1200HC-6MG132C with Lattice Diamond 3.4 design software. When using this design in a different device, density, speed, or grade, performance and utilization may vary.
2. Performance and utilization characteristics are generated using LCMXO3L-4300C-6BG256C with Lattice Diamond 3.4 design software. When using this design in a different device, density, speed, or grade, performance and utilization may vary.
3. Performance and utilization characteristics are generated using LFE5UM-85F-8BG765CES with Lattice Diamond 3.4 design software. When using this design in a different device, density, speed, or grade, performance and utilization may vary.

Technical Support Assistance

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

Revision History

Revision 1.5, December 2019

Section	Change Summary
All	• Changed document number from RD1093 to FPGA-RD-02093. • Updated document template.
Disclaimers	Added this section.

Revision 1.4, September 2015

Section	Change Summary
All	Changed document title to MachXO2, MachXO3 and ECP5 7:1 LVDS Video Interface.
Technical Support Assistance	Updated this section.

Revision 1.3, January 2015

Section	Change Summary
Design Example 2: PassThru Test	Updated Table 7.1., Performance and Resource Utilization. • Updated to support Lattice Diamond 3.4 • Added LSE support for all the device families.

Revision 1.2, March 2014

Section	Change Summary
All	Updated Sapphire device to ECP5 device
Design Example 2: PassThru Test	Updated Table 7.1., Performance and Resource Utilization. • Added support for MachXO3L device family. • Updated to support ECP5 device family.

Revision 1.1, September 2013

Section	Change Summary
All	• Changed document title to Display Interface • Added support for Sapphire device family. • Updated corporate logo.
Technical Support Assistance	Updated Technical Support Assistance information.
Display Interface Requirement	Changed RD_in data from G7 to G0 in the Timings of Video Signals and the Display Interface figure.

Revision 1.0, November 2010

Section	Change Summary
All	Initial release.

www.latticesemi.com