XAPP880 (v1.0) February 10, 2010

SFI-4.1 16-Channel SDR Interface with Bus Alignment Using Virtex-6 FPGAs

Author: Vasu Devunuri

Summary

This application note describes an SFI-4.1 reference design that implements the OIF-SFI4-01.01 interface [Ref 1], a 16-channel, source-synchronous LVDS interface operating at single data rate (SDR). The transmitter (TX) requires 16 LVDS pairs for data and one LVDS pair for the forwarded clock. The transmitter operates at 4:1 serialization on each of the 16 data channels. The receiver (RX) also requires 16 LVDS pairs for data and one LVDS pair for the source-synchronous clock input. The receiver operates at 1:4 deserialization on each of the 16 data channels. The timing of the receiver is described in depth and is characterized in hardware.

Introduction

The design described in this application note targets a Virtex®-6 FPGA, taking advantage of ChipSync™ technology, available for every I/O in all Virtex-6 devices. The features of this design include the ability to dynamically adjust the delay of the clock path in the receiver with 78 ps resolution. Using this dynamic delay feature, the receiver escapes the limitations of static setup/hold timing by creating its own dynamic setup/hold timing. The interface calibrates out process variations by finding the optimal setup/hold timing for each individual device.

Figure 1 shows a Virtex-6 FPGA SFI-4.1 interface talking to an SFI-4.1 interface in another device that can be either an ASIC or an FPGA that supports a 16-channel SDR interface. Because the interface is a source-synchronous link, the receivers of both devices get their clock from the TX side of the other device. The transmitter clocks originate in the backend systems and can be provided from a variety of sources, for example, an oscillator on the PCB. Because each of the 16 data channels on the serial side of the interface runs at 4:1 serialization, the data width on the parallel side of the SDR interface is 64 bits.

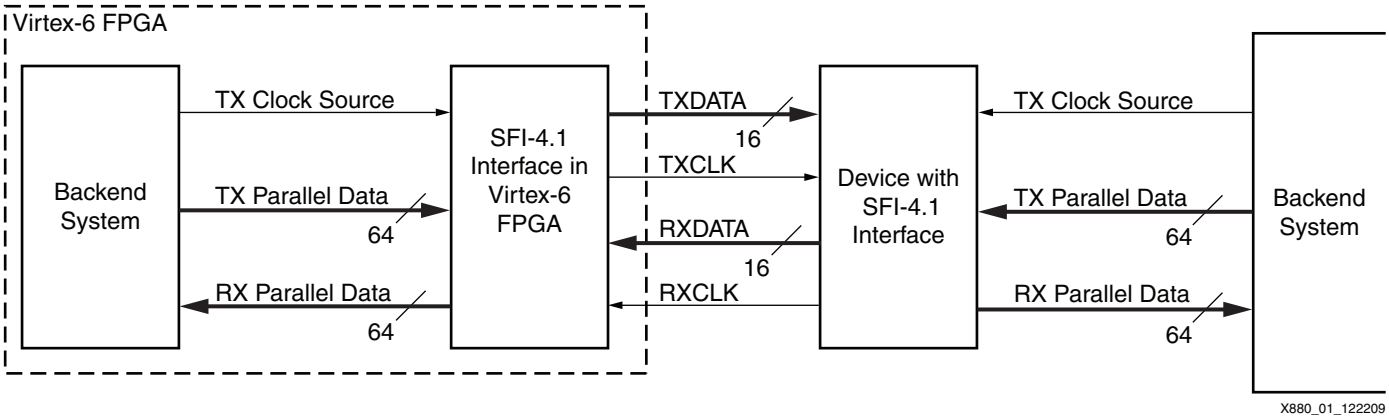


Figure 1: **Full-Duplex, 16-Channel SFI-4.1 Link between a Virtex-6 FPGA and Another Device**

A separate hardware testbench (Figure 2) is available with the ML605 development board to evaluate the performance of this interface in hardware.

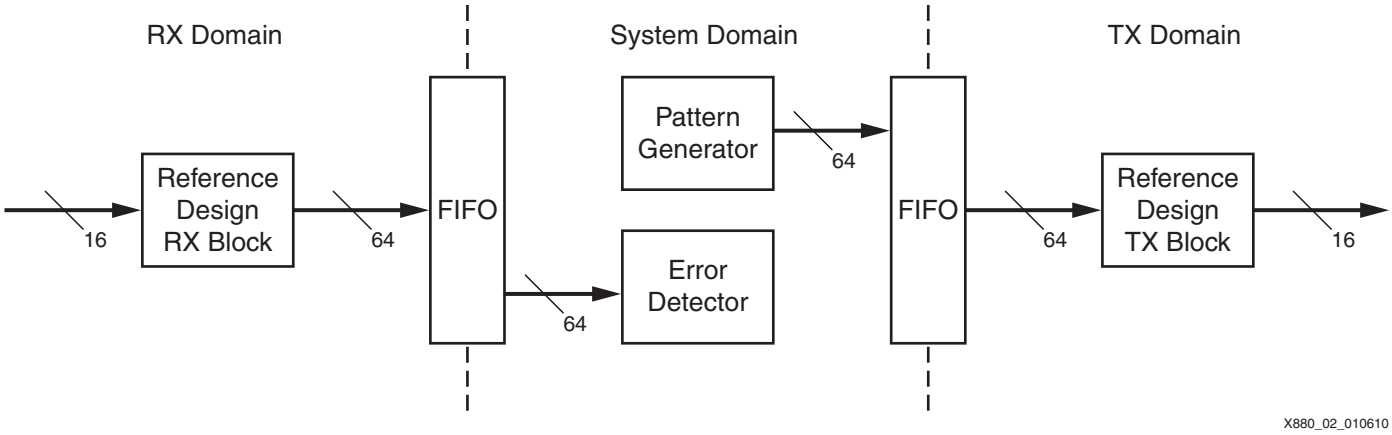


Figure 2: **Testbench for Hardware Verification of the SFI-4.1 Interface**

The hardware testbench includes the latest version of the SFI-4.1 interface. The Virtex-6 FPGA SFI-4.1 transmitter is looped back to the receiver in the same Virtex-6 FPGA on an ML605 Evaluation Board. The hardware testbench communicates statistics like IODELAY tap count of the clock channel and link status etc. to the VIO interface of the ChipScope™ Pro analyzer running on a PC. The hardware testbench is described more in Hardware Testbench Design Details, page 29.

Design Hierarchy of the Interface

The design hierarchy of the stand-alone SFI-4.1 interface is shown in Figure 3. The transmitter is simple and contains almost no logic. The receiver contains logic that performs dynamic alignment on the clock channel.

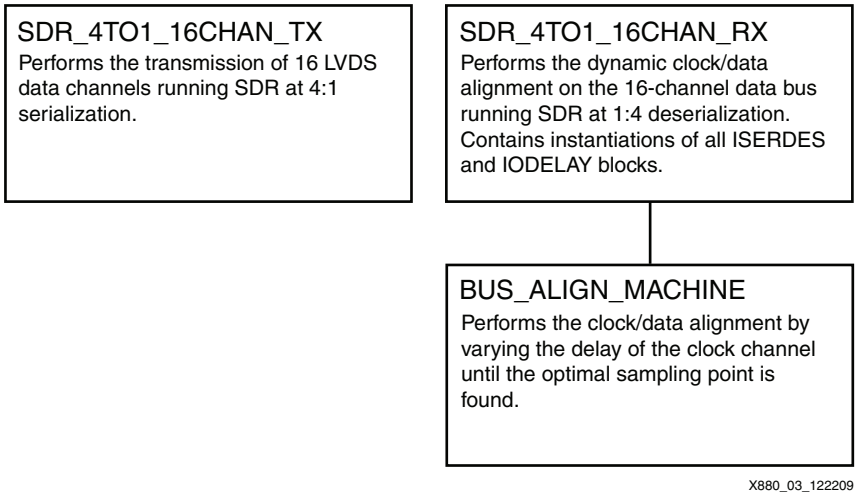


Figure 3: **Design Hierarchy of the Interface (TX and RX)**

SFI-4.1 Transmitter

The SFI-4.1 SDR transmitter consists of only the SDR_4TO1_16CHAN_TX module. The module takes 64 bits of data on the parallel side, performs a 4:1 serialization, and transmits 16 channels of LVDS data on the serial side. The module port list is provided in Table 1.

Table 1: **SDR_4TO1_16CHAN_TX Module Port Definitions**

Port Name	I/O	Definition
DATA_TX_P[15:0]	Output	16 data channels (P)
DATA_TX_N[15:0]	Output	16 data channels (N)
CLOCK_TX_P	Output	Forwarded clock (P)
CLOCK_TX_N	Output	Forwarded clock (N)
TXCLK	Input	TX clock source
TXCLKDIV	Input	TX clock source divided by 4
DATA_TO_OSERDES[63:0]	Input	Parallel side data from the backend system
RESET	Input	Reset synchronous to the TX domain

No logic is used in the transmitter because all required functionality is contained within the OSERDES. The OSERDES is part of the ChipSync technology and is available for every I/O in all Virtex-6 devices. The OSERDES can be programmed to perform any serialization up to 10:1 and do SDR or double data rate (DDR) transmission. Figure 4 shows a single channel of the SFI-4.1 TX.

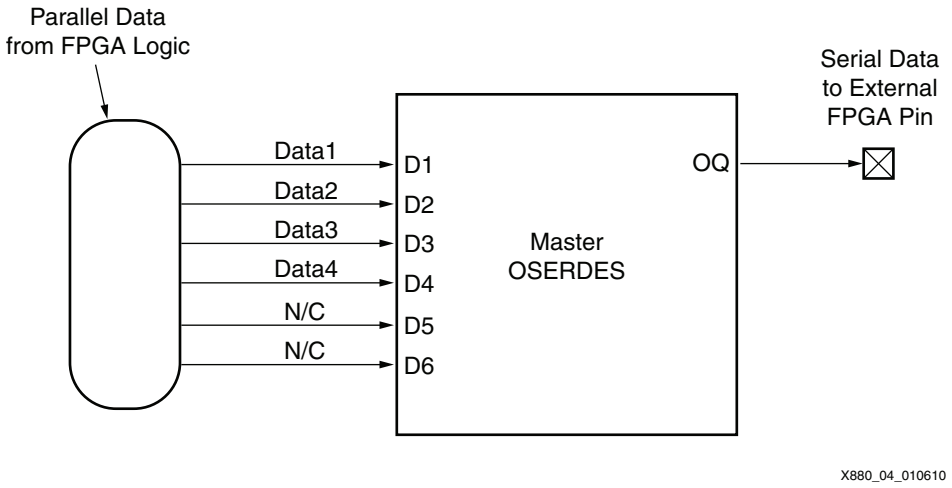


Figure 4: **OSERDES for a Single Data Channel with 4:1 Serialization**

The transmitter interface (Figure 5) consists of 16 OSERDES (the same as Figure 4). Because this interface is source synchronous, the clock must also be forwarded using the ODDR module.

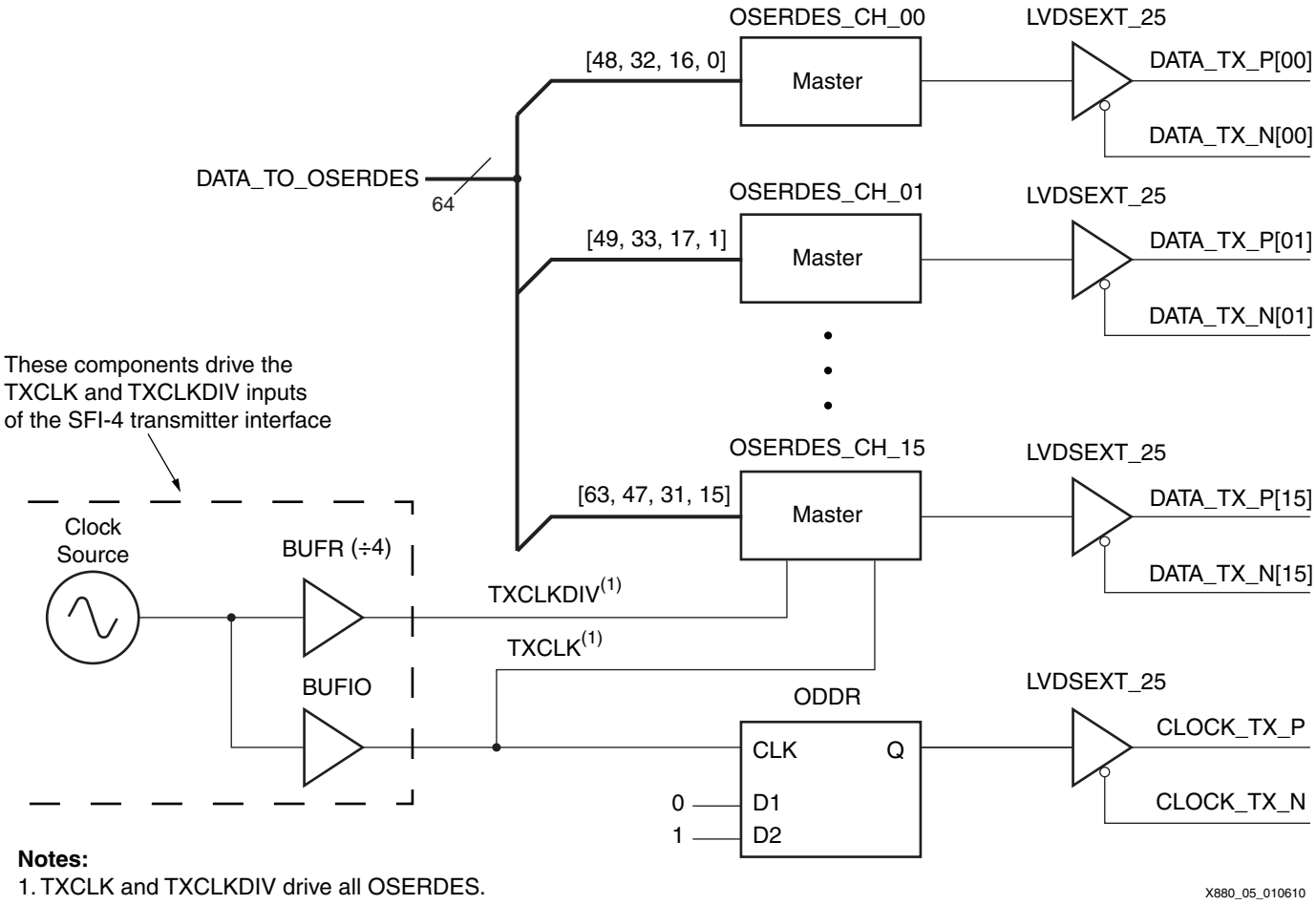


Figure 5: **Block Diagram of the TX Interface**

A training pattern is not allowed in the SFI-4.1 specification, so the transmitter begins transmitting user data immediately following reset. The receiver design must accommodate this limitation. The clock channel is transmitted 180° offset from the data channels. To transmit the clock in phase with the data channels, the D1 and D2 inputs of the ODDR module in Figure 5 can be swapped.

Table 2 provides the device utilization statistics for the TX interface implemented in a Virtex-6 device. Because there is no logic in the transmitter, no look-up tables (LUTs) or slices are used. Table 2 shows one BUFIO and one BUFR is used. Different clocking schemes for the transmitter are discussed in Interface Clocking, page 5.

Table 2: **SFI-4.1 TX Interface Utilization Statistics**

Component	Quantity	Usage Description
Slice Flip-Flop	0	Registered MUX output of DATA_TO_OSERDES
Slice	0	–
LUT	0	–
IOB	34	17 LVDS output pairs (16 data, 1 clock)
OSERDES	16	One OSERDES for every LVDS data pair
ODDR	1	Phase shifts the clock by 180°

Table 2: **SFI-4.1 TX Interface Utilization Statistics *(Cont'd)***

Component	Quantity	Usage Description
BUFIO	1	TXCLK
BUFR	1	TXCLKDIV

Interface Clocking

Two schemes for implementing the SDR transmitter are shown in Figure 6. The regional ChipSync Technology scheme uses BUFIO and BUFR to drive the clock networks. The global scheme drives the clock networks using global clocks divided by four with an MMCM. Using either scheme ensures the phase relationship between the two clocks meets the input requirements of the OSERDES (and ISERDES).

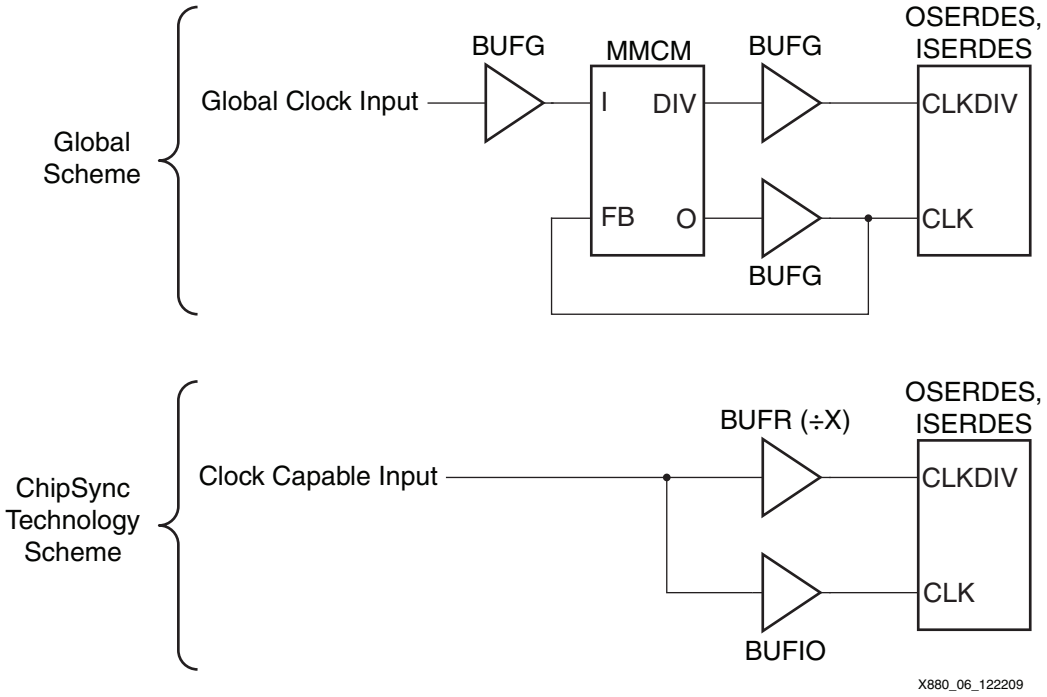


Figure 6: **Global and Regional Clocking Schemes in the Virtex-6 FPGA**

Which clocking scheme is the best to use? The BUFIO/BUFR networks (ChipSync Technology scheme) maximize performance at the expense of some convenience. BUFIO and BUFR are regional clocks that cannot span the entire device like a global clock. This means that the design must include a means to transfer data between the regional and global domains, for example, adding a FIFO between the clock domains.

Two out of four BUFIO in every clock region of the Virtex-6 FPGA can span one clock region above and below it (equivalent to three banks). To implement multiple SFI-4.1 transmitters using regional clocks, a clock source with a rate equal to the data rate must be provided to every transmitter separately. It is possible to provide only one clock source for every three transmitters. However, this is not recommended because it results in more BUFIO clock skew (almost half of the BUFG clock skew).

Global clocks (global clock scheme) have lower performance than BUFIO but are more convenient to use. If a clock is placed on a global network, it can reach any component in the entire device. To show the performance advantage of BUFIO, Table 3 shows snapshots of several timing parameters, some of which are included in the timing budget calculations for source synchronous interfaces.

Table 3: **Comparison of Global and I/O Clock Performance by Speed Grade**

	-3	-2	-1	Units
Maximum Frequency				
BUFG	800	750	700	MHz
BUFIO	800	800	710	MHz
Duty Cycle Distortion				
BUFG $T_{DCD\ CLK}$	120	120	120	ps
BUFIO $T_{DCD\ BUFIO}$	80	80	80	ps
Sampling Error				
BUFG T_{SAMP}	510	560	610	ps
BUFIO T_{SAMP}	300	350	400	ps
Clock Tree Skew				
BUFG skew for XC6VLX240T	290	290	310	ps
BUFIO skew for all devices	30	30	30	ps

Notes:

Values might change based on the further characterization. See *Virtex-6 FPGA Data Sheet: DC and Switching Characteristics* [Ref 2].

In each case, BUFIO has an advantage over BUFG, and collectively those advantages are significant. The values provided in Table 3 are used in the timing budget calculations (Interface Timing Budget, page 15). For the latest specifications, see *Virtex-6 FPGA Data Sheet: DC and Switching Characteristics* [Ref 2].

Although the maximum frequency of both BUFG and BUFIO appear to be within the range of clock speeds required by the SFI-4.1 specification (600–700 MHz), only a BUFIO network is capable of driving an SFI-4.1 design. Because sampling error and clock tree skew affect the size of the data valid window, only the lower values available from BUFIO meet requirements. Even in the highest speed grade, BUFG falls short of these requirements.

Because both the transmitter and receiver of this design are required to be implemented using BUFIO clock networks, all I/Os in the transmitter must be constrained to a single bank, and all I/Os in the receiver must be constrained to a single bank.

The interface does not include the FIFO logic necessary to transfer the parallel data to or from the backend system domain. However, the hardware testbench design discussed in Hardware Testbench Design Details, page 29 shows how to transfer data to and from the RX and TX clock domains using FIFO18E1 primitives.

SFI-4.1 Receiver

The SFI-4.1 receiver consists of two core modules: SDR_4TO1_16CHAN_RX and BUS_ALIGN_MACHINE. The receiver is more complex than the transmitter because it includes a dynamic alignment algorithm implemented in the FPGA logic. The SDR_4TO1_16CHAN_RX module takes 16 channels of data on the serial side, optimizes the timing relationship of the data bus with the clock, performs 1:4 deserializations, and presents 64 bits of data on the parallel side. The module port list is provided in Table 4.

Table 4: **SDR_4TO1_16CHAN_RX Module Port Definitions**

Port Name	I/O	Definition
DATA_RX_P[15:0]	Input	16 data channels (P).
DATA_RX_N[15:0]	Input	16 data channels (N).

Table 4: **SDR_4TO1_16CHAN_RX Module Port Definitions *(Cont'd)***

Port Name	I/O	Definition
CLOCK_RX_P	Input	Forwarded clock (P).
CLOCK_RX_N	Input	Forwarded clock (N).
INC_PAD	Input	Pulsing this pin causes the IDELAY tap setting of the clock channel to increment by 1.
DEC_PAD	Input	Pulsing this pin causes the IDELAY tap setting of the clock channel to decrement by 1.
DATA_FROM_ISERDES[63:0]	Output	Parallel-side data to backend system.
RESET	Input	Reset synchronous to the RX domain.
IDLY_RESET	Input	Reset synchronous to the RX domain that only resets the IDELAY tap settings of the clock channel.
IDELAYCTRL_RESET	Input	Reset synchronous to the TX domain that only resets the IDELAYCTRL module.
CLK200	Input	Reference clock to IDELAYCTRL module.
TAP_00, TAP_01, TAP_03, TAP_04, TAP_05, TAP_06, TAP_07, TAP_08, TAP_09, TAP_10, TAP_11, TAP_12, TAP_13, TAP_14, TAP_15	Output	Each of these 5-bit signals contains the current IDELAY tap value of the 16 data channels. Equal to zero for all data channels because the clock channel is the only dynamically delayed path.
TAP_CLK[4:0]	Output	The 5-bit signal containing the current IDELAY tap value of the clock channel.
TRAINING_DONE	Output	Flag indicating alignment is complete.
RXCLK	Output	RX Source Sync Clock.
RXCLKDIV	Output	RX Source Sync Clock divided by 4.
IDELAY_READY	Output	Flag indicating IDELAYCTRL is calibrated.

Most of the logic used in the receiver is contained within the ISERDES. The ISERDES, which is part of the ChipSync technology, is found in I/Os of all Virtex-6 devices. The ISERDES can be programmed to do any deserialization up to 1:10 and do SDR or DDR reception. Figure 7 shows a single channel of the RX.

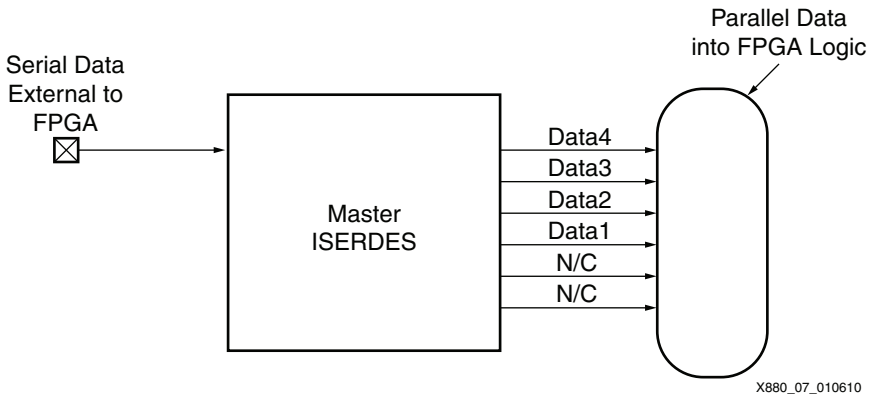
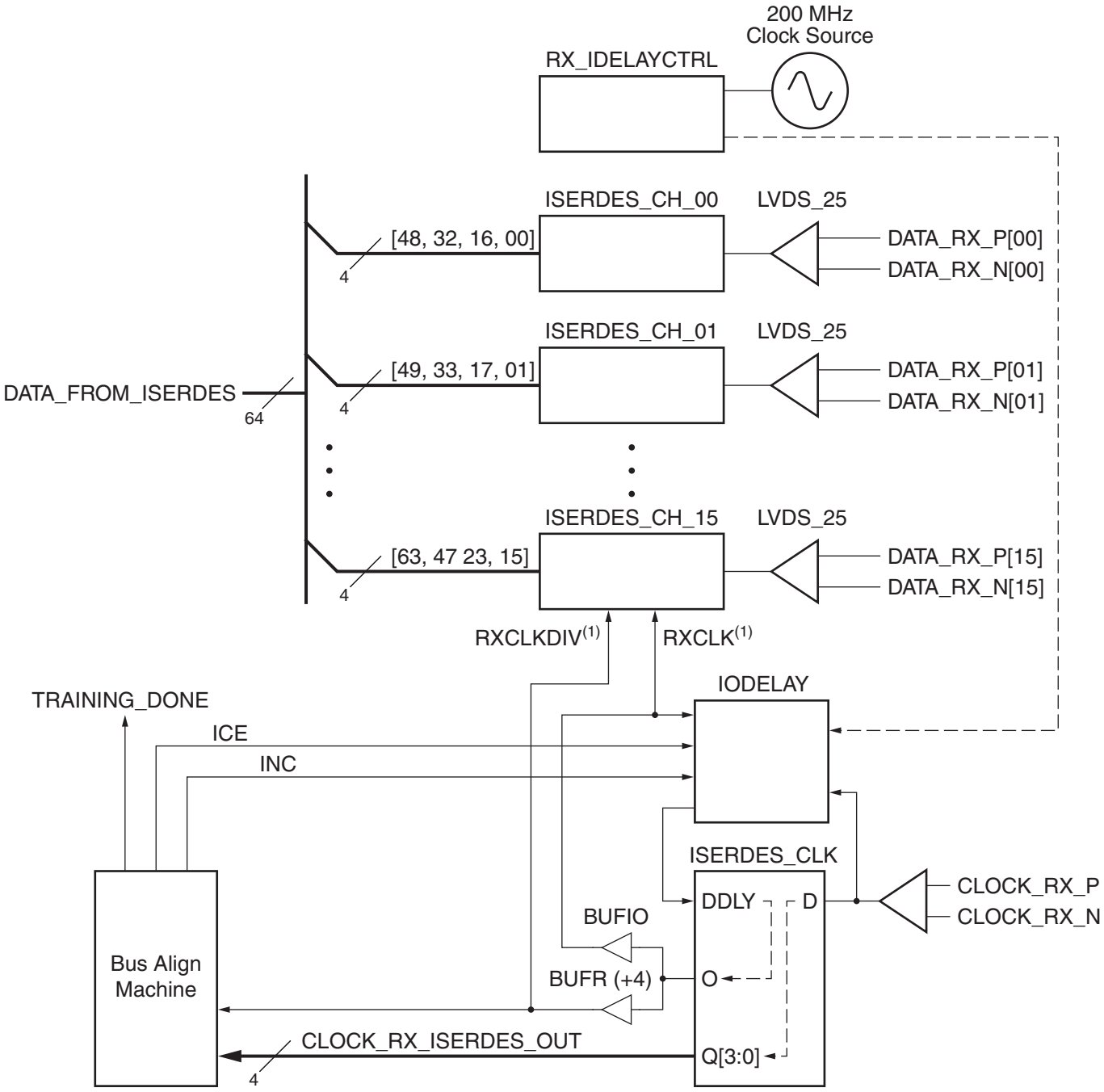


Figure 7: **ISERDES for a Single Data Channel with 1:4 Deserialization**

The receiver interface shown in Figure 8 consists of 16 ISERDES pairs. Because this interface is source synchronous, the clock source comes from the transmitter of the partner device. The

clock is buffered in the receiver by a BUFIO component and divided to the parallel clock rate by a BUFR component.



Notes:
1. RXCLKDIV and RXCLK go to all ISERDES.

X880_08_010710

Figure 8: **RX Interface Block Diagram**

Each ISERDES in Figure 8 contributes 4 bits to the 64-bit parallel data bus. The data outputs of each ISERDES are not contiguous within the 64-bit bus. The 64-bit parallel data bus is divided into four 16-bit words that are striped across the 16 serializer/deserializer channels. Each ISERDES has a single bit from each of the four 16-bit words (for example, Channel 0 has bits 0, 16, 32, and 48). By striping 16-bit words across the 16 channels, the continuity of those

words are guaranteed by design without the need for bit rotation of the parallel data outputs. This is required in an SFI-4.1 design because a training pattern is not allowed at initialization. Without a training pattern, the BITSLIP feature of the ISERDES is not an option. Word alignment must be guaranteed by design. Both the OSERDES and ISERDES are designed to maintain synchronization across channels, so that words striped across multiple ISERDES are continuous.

Because there is no training pattern, the clock channel must play a key role in the initial training process. The data channels continuously stream user data that an algorithm cannot reliably use to perform dynamic alignment, as is described in *16-Channel, DDR LVDS Interface with Per-Channel Alignment* [Ref 3] and *16-Channel, DDR LVDS Interface with Real-Time Window Monitoring* [Ref 4]. The clock channel, however, does contain known content. By treating the clock channel as another data channel and de-serializing it at a ratio of 1:4, the information in the oversampled clock bits can be used to perform dynamic alignment. This is why the forwarded clock in Figure 8 takes two separate paths: one to the BUFIO and BUFR clock networks and one to an ISERDES. This procedure is discussed in detail in Dynamic Timing and BUS_ALIGN_MACHINE, page 13.

The RX_IDELAYCTRL module in Figure 8 is shown as having a dotted line connection to IODELAY (this connection is not visible to the user because the ISE® software takes care of this routing). The RX_IDELAYCTRL is required to calibrate the IODELAY block in the path of the clock channel. The delayed output of the IODELAY block goes to the input of the ISERDES_CLK block (DDLY) and is output to the FPGA logic from the ISERDES_CLK block ouput (O), also shown in Figure 8.

The BUS_ALIGN_MACHINE generates two control signals that adjust the timing of the clock channel. INC and ICE cause the IODELAY to increment or decrement the delay in the clock path by a fixed amount of approximately 78 ps, based on a 200 MHz reference clock.

Table 5 shows the device utilization statistics for the RX interface implemented in a Virtex-6 device. Most of the LUTs utilized are consumed by the BUS_ALIGN_MACHINE, which is the core of the receiver alignment algorithm. As in the TX interface, the RX interface consumes one BUFIO and one BUFR for clocking.

Table 5: **SFI-4.1 RX Interface Utilization Statistics**

Component	Quantity	Usage Description
Slice Flip-Flop	30	Multiple uses
Slice	27	Multiple uses
LUT	64	Multiple uses
IOB	34	17 LVDS input pairs
ISERDES	17	One ISERDES for every LVDS data pair plus one for the clock
IODELAY	1	One for the clock channel
BUFIO	1	RXCLK
BUFR	1	RXCLKDIV

Receiver Interface Dynamic Timing

The overall timing of the receiver interface is influenced by two factors:

- The suboptimal timing inherent in the data and clock paths.
- The optimal timing created by the receiver when adjusting dynamic delays on the clock channels at initialization.

Inherent Timing

All propagation delays in the data and clock paths must be taken into consideration to derive the inherent setup/hold times. The data and clock paths for a single channel of this interface and the timing components that comprise the paths are shown in Figure 9. Referring back to Figure 8, page 8, the clock channel followed two paths: one through the IODELAY/ISERDES_CLK combination, and one through the ISERDES_CLK. The path through the IODELAY/ISERDES_CLK combination is represented explicitly in Figure 9 because it is required to pass the clock to BUFIO. Ideally, the data and clock channels are perfectly phase-aligned at the input pins (there is, of course, skew, which is considered in Interface Timing Budget, page 15). All components in the path have both a minimum and maximum value to account for process variations.

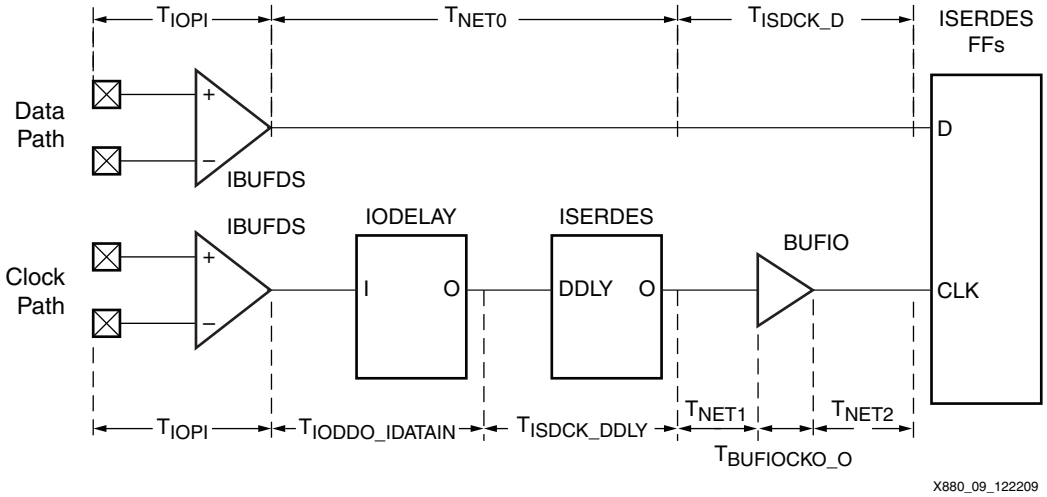


Figure 9: **Timing Components of RX Data and Clock Paths**

The minimum and maximum values are generated using the ISE design suite timing analyzer. Figure 10 shows the timing analyzer settings used to generate the information about each of the parameters. The timing parameters are described in Table 6 and Table 7. Example minimum and maximum values generated for the data and clock paths are shown in Table 8 and Table 9

Note: All raw timing numbers referenced in this document are subject to minor changes in subsequent revisions of the ISE design suite.

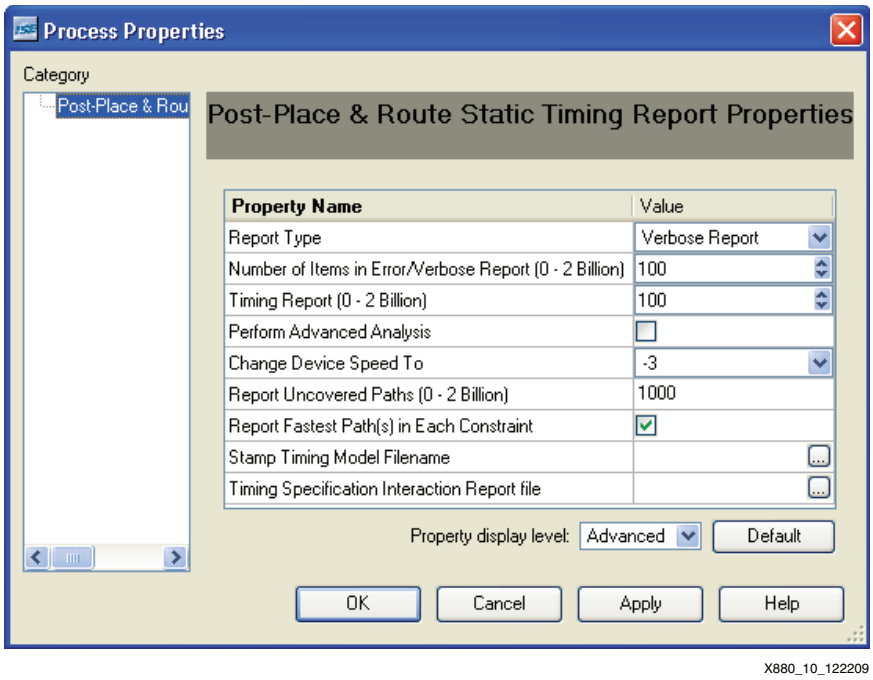


Figure 10: **ISE Design Suite Timing Analyzer Properties**

Table 6: **Datapath Timing Definitions**

Timing Parameter	Description
T_{IOPI}	Delay from the IOB pad through the LVDS input buffer to the I pin of the IOB pad
T_{NET0}	Delay from the I pin of IOB pad to the D input of the ISERDES
T_{ISDCK_D}	Delay from the D input of the ISERDES to the sampling registers in the ISERDES (setup and hold times of ISERDES)

Table 7: **Clock Path Timing Definitions**

Timing Parameter	Description
T_{IOPI}	Delay from the IOB pad through the LVDS input buffer to the I pin of the IOB Pad
$T_{IODDO\_IDATAIN}$	Delay from I pin of IOB pad to the O output of the IODELAY block
T_{ISDCK_DDLY}	Delay from O output of the IODELAY block to O output of ISERDES via DDLY input of ISERDES
T_{NET1}	Delay from the O output of the IODELAY block to the I pin of BUFIO
$T_{BUFIOCKO\_O}$	Clock to out delay of BUFIO
T_{NET2}	Clock distribution delay from BUFIO output to the clock input of data ISERDES

Table 8: **Datapath Delay Inventory for a -2 Speed Grade XC6VLX240T Device**

Timing Parameter	Maximum Datapath Delay	Minimum Datapath Delay
T_{IOPI}	0.875 ns	0.599 ns
T_{NET0}	0.001 ns	0.001 ns
T_{ISDCK_D}	–0.110 ns	0.008 ns
TOTAL	0.766 ns	0.608 ns

Table 9: **Clock Path Delay Inventory for a -2 Speed Grade XC6VLX240T Device**

Timing Parameter	Maximum Clock Path Delay	Minimum Clock Path Delay
T_{IOPI}	0.885 ns	0.611 ns
$T_{IODDO\_IDATAIN}$	0.532 ns	0.316 ns
T_{ISDCK_DDLY}	–0.074 ns	0.023 ns
T_{NET1}	0.130 ns	0.067 ns
$T_{BUFIOCKO\_O}$	0.162 ns	0.065 ns
T_{NET2}	1.278 ns	0.564 ns
TOTAL	2.915 ns	1.646 ns

According to the inherent timing prediction, the data is going to arrive before the clock by an amount determined by Equation 1 and Equation 2. Equation 1 calculates that data arrives before the clock by at least 0.88 ns in a -2 device. Equation 2 calculates that data arrives before the clock by at most 2.307 ns. Equation 3 shows a timing window that covers all -2 devices under all conditions.

$$\text{Setup Time} = \text{Max Data Delay} - \text{Min Clock Delay} = 0.766 - 1.646 = -0.88 \text{ ns} \quad \textit{Equation 1}$$

$$\text{Hold Time} = \text{Max Clock Delay} - \text{Min Data Delay} = 2.915 - 0.608 = 2.307 \text{ ns} \quad \textit{Equation 2}$$

$$\text{Timing Window} = 2.307 - 0.88 = 1.427 \text{ ns} \quad \textit{Equation 3}$$

This timing window describes the inherent timing of the data and clock paths. Assuming skew-matched PCB traces for all 16 data channels and the clock channel, it can be assumed that all 16 channels in the receiver have roughly the same timing window.

Figure 11 shows the timing windows for all speed grades of an XC6VLX240T device and the locations of hardware measurements of inherent timing relationships (to validate the calculation). The hardware measurements show the SDR interface data arriving before the clock within the ranges of the respective speed grades.

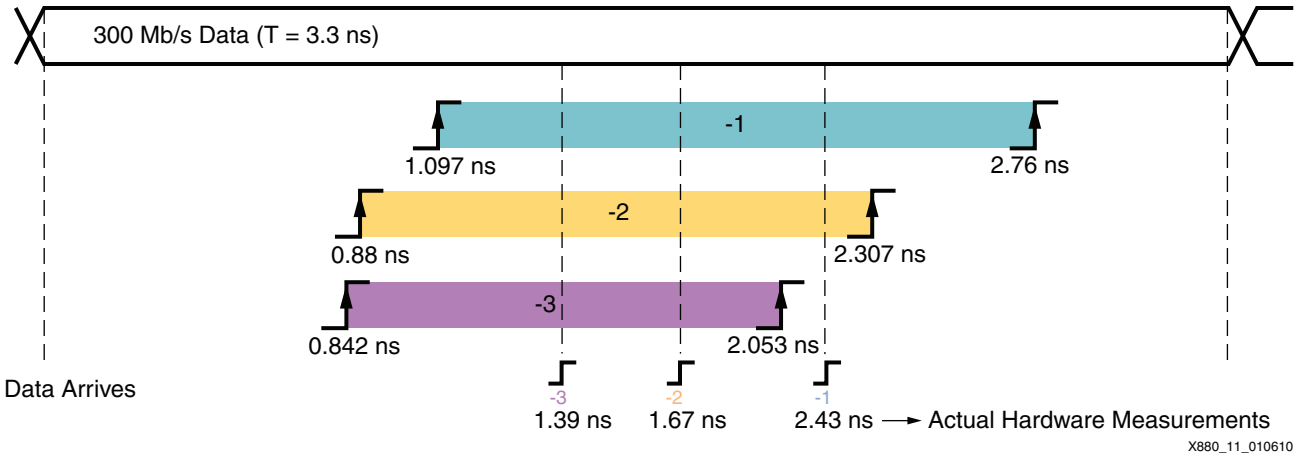


Figure 11: **Timing Windows Show Uncertainty in Clock/Data Relationship as Calculated by Timing Analyzer**

Dynamic Timing and BUS_ALIGN_MACHINE

In Inherent Timing, page 10, the inherent timing of the interface is calculated for all speed grades and validated with a hardware measurement. In this case, Figure 11 shows the inherent timing is sufficient because the data rate is only 300 Mb/s. Figure 12 illustrates what happens when the data rate is increased, causing the hold edge of the data window to shrink.

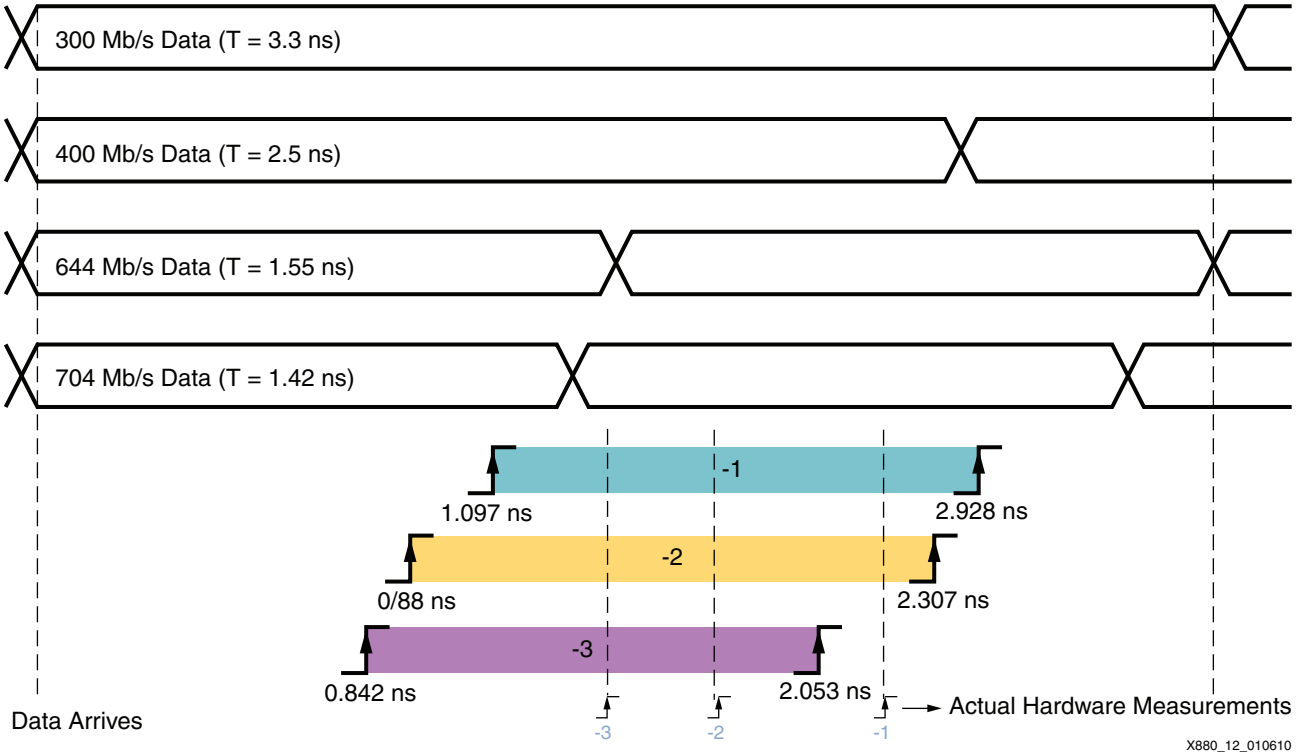


Figure 12: **Inherent Timing Windows Shown in Relation to Multiple Data Rates**

At 400 Mb/s, the inherent timing relationship is no longer adequate to meet timing because there are cases when the sampling clock edge is in the middle of the data transition. The

solution to replace the suboptimal, inherent timing with dynamic timing. When using dynamic timing, the BUS_ALIGN_MACHINE module in the receiver positions the sampling edge of the clock at the center of the data eye by adding delay to the clock path.

The principle of bus alignment is to treat the entire data bus as a single entity and adjust the clock channel to the most optimal sampling position for the bus. No delay adjustments are made to individual data channels to compensate for skew because any skew between data channels directly counts against the interface timing budget

The difficulty when implementing dynamic timing is finding the center of the data eye without a training pattern at initialization. To find the center, two assumptions are made:

- In an SFI-4.1 interface where the data and clock bus are skew-matched as closely as possible, the center of the data eye coincides with the rising edge of the clock.
- The clock has 50% duty cycle distortion.

Under these assumptions, the clock can be treated as a pseudo data channel and can be deserialized by sampling itself. The sampled clock bits can be monitored while increasing the clock delay until the rising edge is found, indicating that the clock is sampling at the center of the data bus eye.

The steps executed by the bus-alignment procedure are shown in Figure 13. The algorithm's task is to add delay to the clock channel until a rising edge is observed. Given the initial timing shown in Figure 13, the clock must first pass a falling edge before it detects the rising edge. The initial timing could also be oriented so that the rising edge is found first. If all possible initial timing relationships are considered, the maximum delay that is needed on the clock channel is one period of the clock, and the minimum delay that is needed is 78 ps, or 1 IDELAY tap.

There are several advantages to applying delay to the clock channel rather than the data channels. First, the need for a training pattern is eliminated. Second, interrogating the clock channel is easier than interrogating a data channel because the algorithm has no jitter to anticipate. Finally, a clock pattern does not suffer degradation when passing through the IDELAY chain, whereas a data pattern collects a finite amount of pattern jitter in each IDELAY tap.

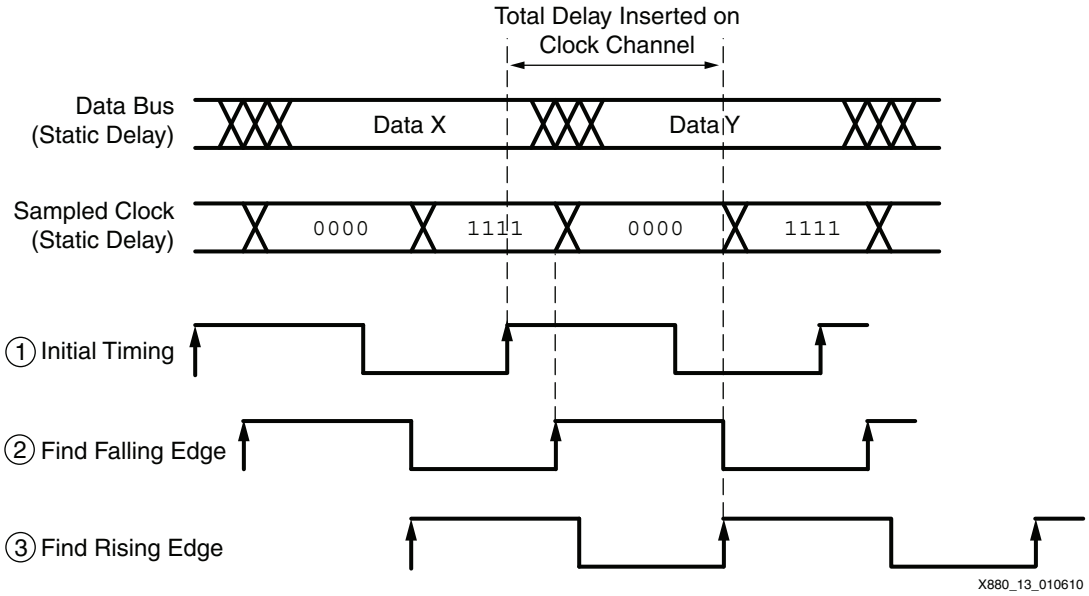


Figure 13: **Dynamic Bus Alignment Procedure in the BUS_ALIGN_MACHINE Module**

The FSM in the BUS_ALIGN_MACHINE module used to implement bus alignment is shown in Figure 14. The FSM considers three possible scenarios:

- When the first sampled clock equals 1
- When the first sampled clock equals 0
- When the first sampled clock is unstable

All scenarios end by finding the rising edge of the clock, which is the center of the data eye.

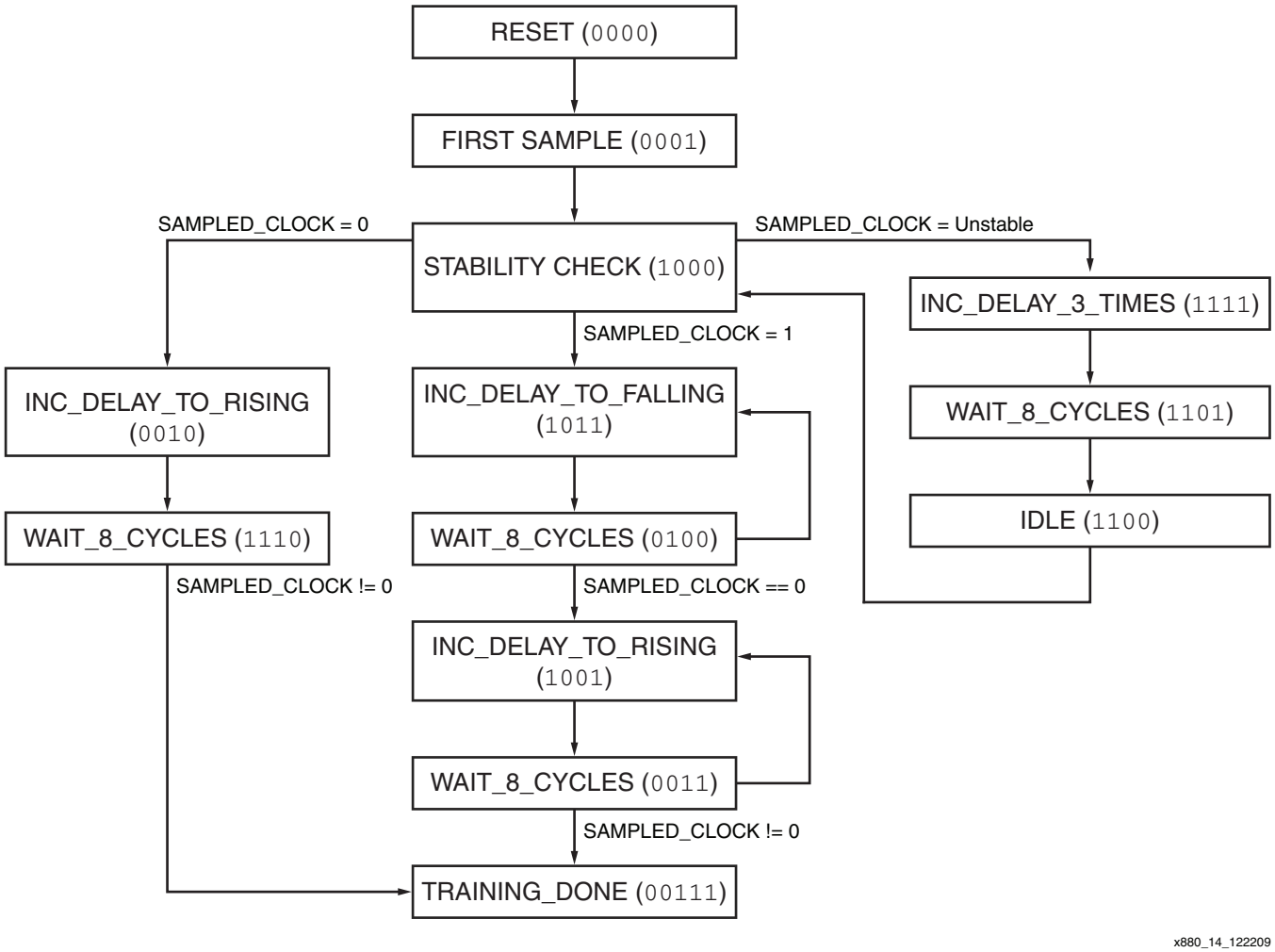


Figure 14: **Dynamic Bus Alignment Procedure in the BUS_ALIGN_MACHINE Module**

Interface Timing Budget

In addition to sampling the data as closely as possible to the center of the data eye, the actual size of the data eye is important. By accurately predicting the size of the data eye in the receiver, the overall performance ceiling of the interface can be inferred. The ideal data eye width is the period of the data rate (for example, T_{Bit_Period} = 1.43 ns for 700 Mb/s). Both the transmitter and receiver circuits have sources of error that subtract from the ideal data eye. The equation for the eye width at the output of the transmitter is shown in Equation 4.

$$DATA\_EYE\_WIDTH\_TX = T_{PERIOD} - T_{JITTER} - T_{BUFIOSKEW} - T_{PKGSKEW} \quad \textit{Equation 4}$$

T_{JITTER} is the peak-to-peak jitter of the clock source driving the transmitter data bus and forwarded clock. Skew must be subtracted from the timing budget because the dynamic alignment algorithm aligns the clock to the *entire* data bus, not each individual channel. In the

transmitter, skew on the clock network that drives the data bus and forwarded clock ($T_{BUFIOSKEW}$) and skew in the device package ($T_{PKGSKEW}$) count against the transmitter timing budget. Duty cycle distortion of the clock network does not cause data eye closure and need not be included as a source of error.

Equation 5 calculates the width of the data eye at the receiver flip-flops. Ideally, the eye width at the receiver is the same as the eye width at the output of the transmitter, but there are sources of error in the transmission medium and receiver circuitry. T_{SAMP_BUFIO} is the receiver sampling error due to drift caused by voltage and temperature variations. The receiver implementation eliminates the need to consider pattern jitter in the IDELAY chain because IDELAY is only used on the clock channel. For a clock pattern, $T_{IDELAYPAT\_JIT}$ is 0 ps/tap peak to peak. For a data pattern with long and short run lengths of zeroes and ones (simulated by PRBS23), $T_{IDELAYPAT\_JIT}$ is a finite value that must be considered in the timing budget. Skew must be counted again in the receiver.

$$DATA\_EYE\_WIDTH\_RX = DATA\_EYE\_WIDTH\_TX - T_{BOARD\_JITTER} - T_{PCBTRACE\_SKEW} - T_{BUFIOSKEW} - T_{PKGSKEW} - T_{SAMP\_BUFIO} - T_{QUANTIZATION\_ERR} \quad \textit{Equation 5}$$

T_{BOARD_JITTER} includes jitter induced by receiver input capacitance and the parasitics of the physical datapath on the PCB. T_{BOARD_JITTER} must be determined by simulation because it varies as a function of frequency and for receivers of different types and vendors. On the ML605 board, with a -1 speed-grade Virtex-6 device, the T_{BOARD_JITTER} is measured at roughly 200 ps at 700 Mb/s (assuming data content equivalent to PRBS23). $T_{QUANTIZATION\_ERR}$, or quantization error, is caused by the granularity of the IDELAY tap chain. Working with 78 ps increments, it is not possible to utilize the entire data eye because inevitably some valid eye width is lost between two taps. The value for this loss is 78 ps. As long as the receiver eye width calculated using Figure 5 is greater than or equal to 0, the interface operates correctly over process, voltage, and temperature.

For example, Equation 6 and Equation 7 show the timing budget calculation for the SFI-4.1 interface running at 700 Mb/s in a -1 speed grade Virtex-6 FPGA on an ML605 evaluation board. Refer to *Virtex-6 FPGA Data Sheet: DC and Switching Characteristics* [Ref 2] for the latest values. The TX interface is looped back to the RX interface on the same Virtex-6 FPGA, as shown in Figure 2.

$$DATA\_EYE\_WIDTH\_TX = 1430\ ps - 25\ ps - 30\ ps - 107\ ps = 1268\ ps \quad \textit{Equation 6}$$

$$DATA\_EYE\_WIDTH\_RX = 1193\ ps - 200\ ps - 25\ ps - 30\ ps - 107\ ps - 400\ ps - 78\ ps = 354\ ps \quad \textit{Equation 7}$$

Changes in voltage and temperature cause the data channels of the interface to drift relative to the clock. If the SFI-4.1 interface were to transfer user data for a long period of time over all specified conditions of temperature and voltage, Equation 7 states that there would be at least 354 ps (approximately four IDELAY taps) of data window remaining valid during all fluctuations of temperature and voltage. This calculation must be compared to the measurements in the 700 Mb/s case in Table 10, page 18. There are twelve taps that are error free under all conditions.

At a given temperature and voltage setting, the data valid window is much larger because there is no drift when temperature and voltage are constant. The size of the data valid window under constant conditions is 354 ps + T_{SAMP_BUFIO}, which in this case is at least 754 ps (nine or ten IDELAY taps). This calculation must be compared to the measurements of any single condition for the 700 Mb/s case in Table 10, page 18. In the worst case (–5% supply), there are 14 taps that are error free.

The skew specifications used in these calculations represent the worst possible skew for any set of I/Os in a single bank used to implement an interface. When a designer reaches the implementation phase of the SFI-4.1 Interface design and selects I/O location constraints, the ISE Design Suite timing analyzer can be used to determine the exact skew of the interface. The

actual skew is smaller or equal to the worst-case specification calculated in Equation 6 and Equation 7.

Interface Characterization

By following the procedure illustrated in Figure 13, the bus-alignment algorithm should never insert more than approximately one bit time of delay on the clock channel. The largest number of IDELAY taps are inserted when the clock initially samples just after the center of the data eye, as in the 300 Mb/s case in Figure 12. The algorithm must move all the way through the rest of the first eye to find the first transition (~0.5 bit time) and then move halfway into the next eye to center the sampling point (0.5 bit time). This process produces a maximum total delay insertion of 1.0 bit times, which is shown in Figure 15.

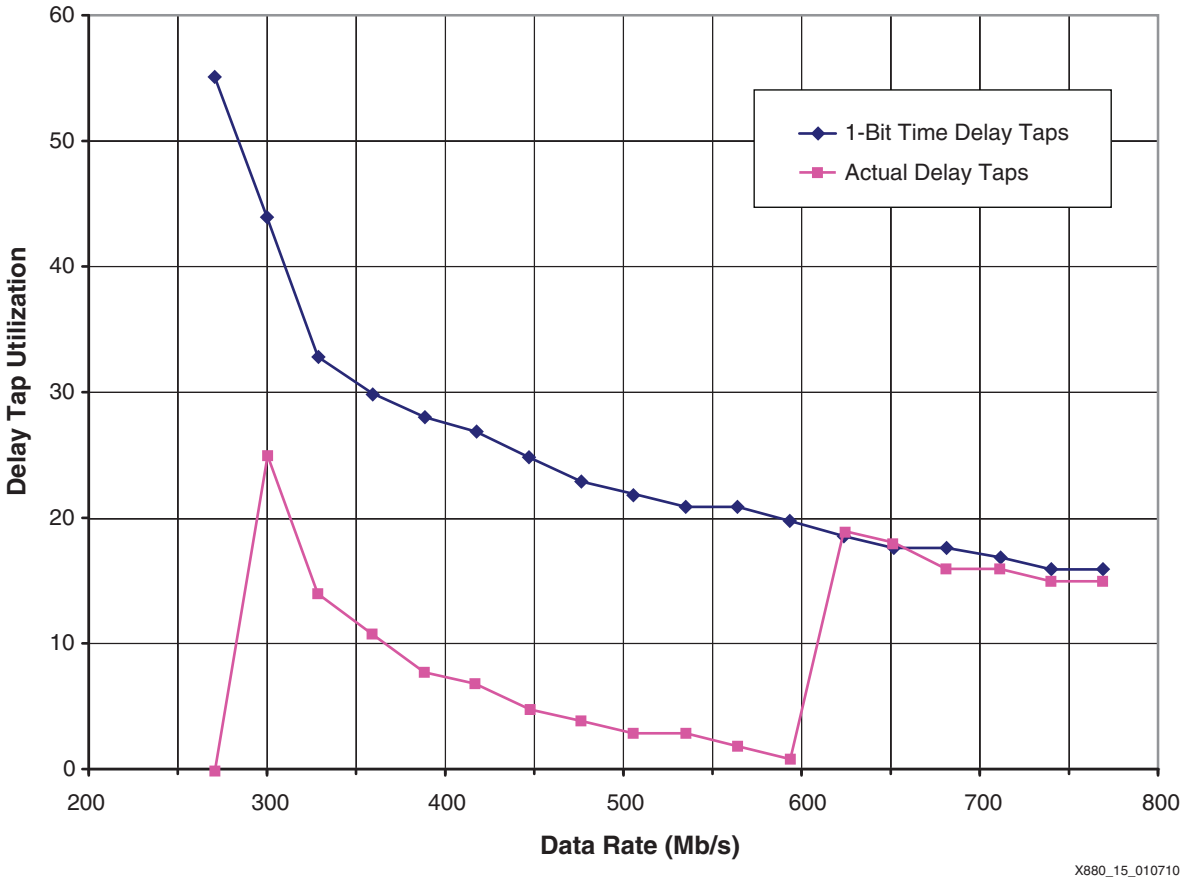


Figure 15: **Measured IODELAY Tap Settings vs. Data Rate on Clock Channel**

The actual performance of the bus-alignment algorithm is characterized as a function of data rate in Figure 15. For all data rates, the expectation is that the tap setting of the clock channel in the SFI-4.1 receiver does not exceed approximately 1-bit times of delay. Figure 15 only shows the data for a single device. Comparing the 1-bit time delay tap to the actual delay tap settings, it is clear that the algorithm remains within the 1-bit time limit.

The tap settings in Figure 15 generally decrease linearly with shrinking bit periods, except in the case of 240 Mb/s and 680 Mb/s data rates. These data rates are “seams” in the inherent timing of this particular device. On one side of the seam, the tap setting is the minimal value of one, and on the other side of the seam, the tap setting is the maximum value of taps (totaling one bit period). The reason for this non-linear change in tap setting is the inherent timing that the bus-alignment algorithm starts with. At 240 Mb/s, the clock is sampling just slightly before the center of the data eye (indicated by the falling edge of the clock), and the algorithm does not need to insert more than a single tap of delay to get to the center of the data eye. At 300 Mb/s, the slightly shorter bit period has moved the initial sampling point of the clock past the center of

the data eye and the bus-alignment algorithm must move through more than half bit period to find the next data eye (and the falling edge of the clock).

These seams might cause a word misalignment because from one reset to another, the algorithm could choose two different data eyes (for example, Tap 1 or Tap 19 at 680 Mb/s). However, these seams are not an issue with this implementation because data is divided into 16-bit words and striped across the 16 channels. Regardless of which data eye is chosen by the algorithm, the continuity of the 16-bit words is always preserved. As long as the algorithm gets to the center of any data eye, the receiver sees a stream of uncorrupted 16-bit words.

Because the receiver alignment logic runs only one time immediately after reset, the resulting dynamic timing must be adequate to ensure error-free operation over the specified voltage and temperature variations that can occur over time. The data-valid window at the receiver input must be wide enough to withstand these variations (receiver drift). Table 10 shows the data valid window at the ISERDES registers in the SDR receiver. The eyes shown are collective eyes of all 16 channels. The testbench configuration loops back the SFI-4.1 transmitter to the SFI-4.1 receiver. Using pattern generators and error detectors programmed to send and receive a pseudorandom bit sequence (PRBS23), the width of the data-valid window is evaluated using pseudorandom data that closely resembles real user data.

Table 10: **Measurement of Receiver Drift under Extreme Conditions for Multiple Data Rates**

Condition	IODELAY Taps(1)																						
	–11	–10	–9	–8	–7	–6	–5	–4	–3	–2	–1	0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+10	+11
Case: 620 Mb/s																							
25°C, Nom Supplies	P	P	P	P	P	P	P	P	P	P	P	C	P	P	P	P	P	P	P	P	P	F	F
0°C, Nom Supplies	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F
85°C, Nom Supplies	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F
25°C, +5% Supplies	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
0°C, +5% Supplies	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
85°C, +5% Supplies	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F
25°C, –5% Supplies	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F
0°C, –5% Supplies	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F	F
85°C, –5% Supplies	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F	F	F
Center Tap Setting	Tap 2																						
Conclusion	Error free under all conditions when calibrated under nominal conditions to position C																						
Case: 700 Mb/s																							
25°C, Nom Supplies	F	F	F	F	F	F	P	P	P	P	P	C	P	P	P	P	P	P	P	P	P	P	F
0°C, Nom Supplies	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F
85°C, Nom Supplies	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F
25°C, +5% Supplies	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
0°C, +5% Supplies	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
85°C, +5% Supplies	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F
25°C, –5% Supplies	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F	F
0°C, –5% Supplies	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F	F
85°C, –5% Supplies	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F	F	F
Center Tap Setting	Tap 19																						
Conclusion	Error free under all conditions when calibrated under nominal conditions to position C																						

Table 10: **Measurement of Receiver Drift under Extreme Conditions for Multiple Data Rates *(Cont'd)***

Condition	IODELAY Taps[1]																					
	–11	–10	–9	–8	–7	–6	–5	–4	–3	–2	–1	0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+10	+11
Case: 730 Mb/s																							
25°C, Nom Supplies	F	F	F	F	F	F	F	F	F	P	P	C	P	P	P	P	P	P	P	P	P	P	P
0°C, Nom Supplies	F	F	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P
85°C, Nom Supplies	F	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
25°C, +5% Supplies	F	F	F	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P
0°C, +5% Supplies	F	F	F	F	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P
85°C, +5% Supplies	F	F	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	F	F
25°C, –5% Supplies	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F	F
0°C, –5% Supplies	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F	F
85°C, –5% Supplies	F	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F
Center Tap Setting	Tap 16																						
Conclusion	Error free under all conditions when calibrated under nominal conditions to position C																						

Notes:

1. Where 0 is the reference tap determined by the bus-align machine under nominal conditions at 25°C.
2. P = Error-free transmission; F = Error in transmission; C = IODELAY tap position selected by the bus-align machine under nominal conditions at 25°C.
3. Device under test is S/N 5, -1 speed grade.

In all test cases shown in Table 10, the bus-align machine accurately places the sampling point in the center of the data eye (indicated by the "C" in each test case). The + or – values beside the 0 (center) in Table 10 indicate the number of TAP delays moved away from center on either side.

After the dynamic timing is established for a given test case under nominal conditions, the device is subjected to extremes of temperature and voltage to stress the receiver timing. For all data rates up to 730 Mb/s, the dynamic timing established after reset under nominal conditions is sufficient to meet timing over all specified extreme conditions (the column with the "C" passes traffic error free under all conditions).

The full datapath of the testbench consists of:

TX → 5" FR-4 → SAMTEC QSE Conn → 12" Ribbon Cable → SAMTEC QSE Conn → 5" FR-4 → RX

Performance is achieved in accordance with the section Interface Timing Budget, page 15. Interface performance is not solely a question of the LVDS driver performance specification, but rather a question of adequate timing margin to withstand production and environmental variations. From Table 10, the total amount of variation experienced over temperature and voltage is roughly 450 ps (6 taps at 78 ps per tap). This number should be comparable to the specification for receiver sampling error (T_{SAMP_BUFIO}) shown in *Virtex-6 FPGA Data Sheet: DC and Switching Characteristics* [Ref 2].

The ML605 board supports digitally programmable FPGA power supplies (V_{CCINT}, V_{CCO} and V_{CCAUX}) using PMBus programming cable. A Texas Instruments UCD9240, 4-channel PMBUS controller manages the embedded FPGA power supplies. The controller contains nonvolatile flash memory for storage of control constants for each supply. The programming cable is not needed for normal operation. Default control constants have been programmed into the UCD9240 so that the DUT rails power-up to the nominal values as required. Nonvolatile memory can be programmed to generate other power-up set-points.

The graphical user interface (GUI) is shown in Figure 16. The GUI is used to select rails and make changes.

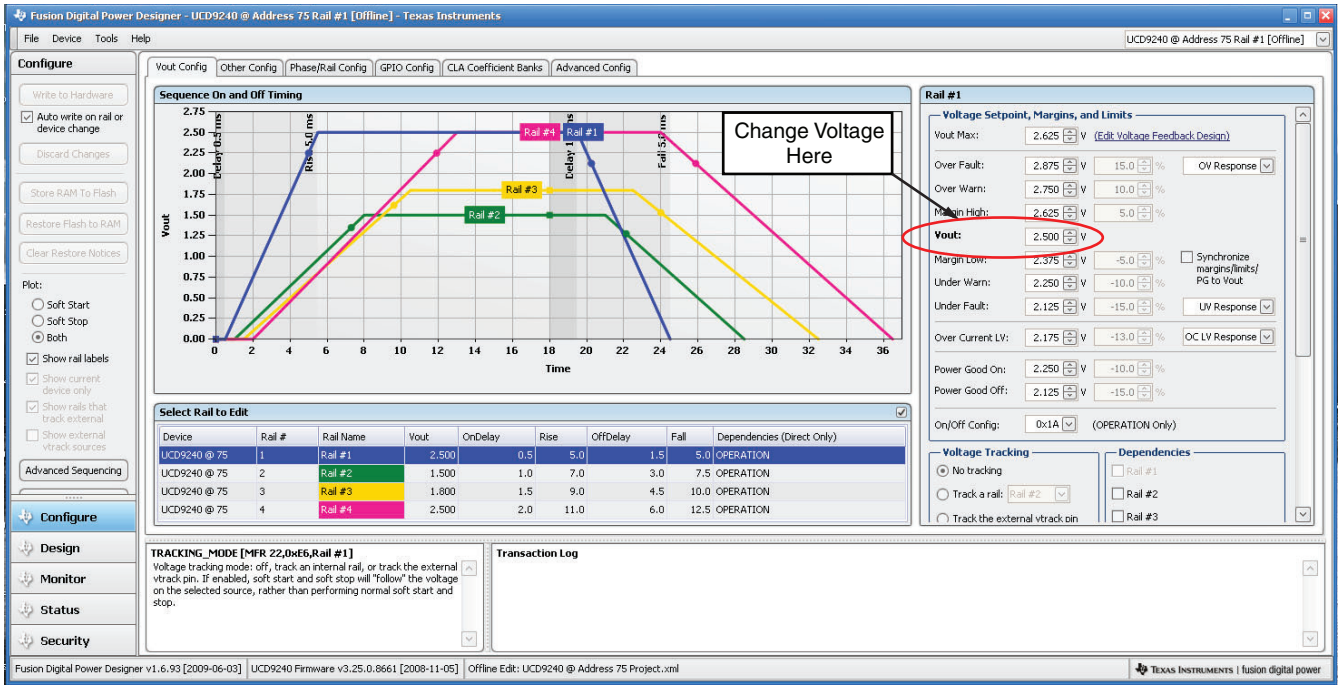


Figure 16: **GUI to Change the Voltage**

Resetting the Interface

Several dependencies determine the order in which circuits are reset in the interface. The receiver cannot begin the alignment algorithm if the transmitter is not sending the clock. The transmitter must be reset before the receiver, and the clock must have adequate time to propagate across the link before the receiver comes out of reset. The receiver also cannot begin the alignment algorithm until IDELAYCTRL recovers from reset, which can take hundreds of cycles. The recommended reset timing for the interface is shown in Figure 17. This reset timing is implemented in the testbench for this interface.

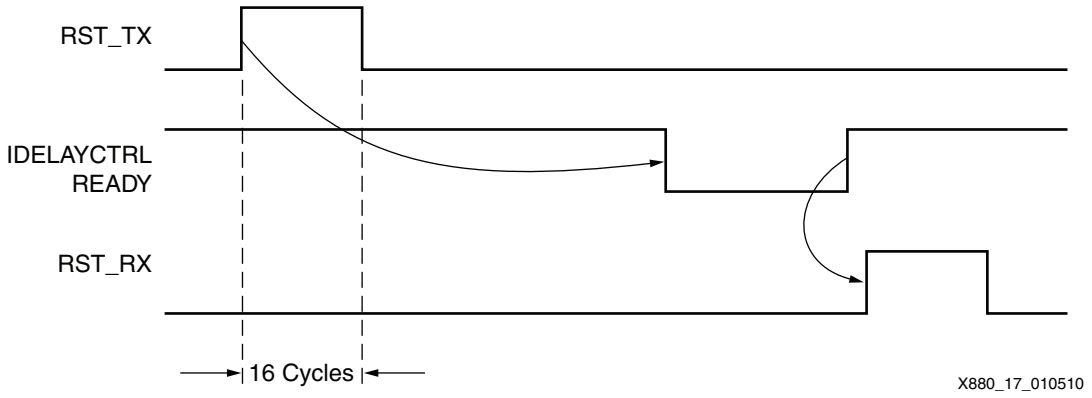


Figure 17: **Recommended Interface Reset Timing**

In the testbench, the TX reset (RST_TX) resets all TX circuitry and IDELAYCTRL. The IDELAYCTRL reset is asynchronous, so driving it with RST_TX is acceptable. The minimum pulse width of IDELAYCTRL reset is 50 ns ($T_{IDELAYCTRL\_RPW}$), so resets are held High for 16 clock cycles. IDELAYCTRL reset does not need to be driven specifically by RST_TX, so it can be driven by an entirely independent reset as long as the reset signal adheres to the timing in Figure 17 with respect to RST_RX. The RX reset (RST_RX) is used to reset all RX circuitry and

is asserted when the IDELAYCTRL READY flag goes High, indicating that the IODELAY blocks are calibrated.

The timing of the READY flag in Figure 17 is a potential hazard. It is likely that READY remains High while IDELAYCTRL is being reset because it takes many cycles for IDELAYCTRL to deassert the READY flag. For this reason, it is easy to make the mistake of resetting the receiver too early. To avoid this hazard, the RX reset should be asserted only after the READY flag is deasserted and reasserted.

Conclusion

Based on the design and characterization described in this document, the SFI-4.1 16-Channel SDR reference design meets and exceeds the performance targets in Table 11 under all conditions of process, voltage and temperature. However, to meet these targets over voltage and temperature, the timing budget must be guaranteed according to the analysis in Interface Timing Budget, page 15. Any deviation from the timing budget causes further degradation in the maximum performance. See Appendix, page 23 for further justification of the maximum performance recommendations in Table 11.

Table 11: **Performance of SFI-4.1 Interface**

Speed Grade	Maximum Performance
-1	655 Mb/s(1)
-2	704 Mb/s(1)
-3	740 Mb/s(1)

Notes:

1. These performance targets are dictated by the OLOGIC component switching limitation specification in *Virtex-6 FPGA Data Sheet: DC and Switching Characteristics* [Ref 2]. Although this reference design shows significant performance margin, the ODDR (OLOGIC) are only guaranteed to the rates in this table.

Reference Design

The reference design files can be downloaded at:

https://secure.xilinx.com/webreg/clickthrough.do?cid=142978

The reference design checklist is shown in Table 12.

Table 12: **Reference Design Checklist**

Parameter	Description
Developer Name	Xilinx
Target Devices (stepping level, ES, production, speed grades)	Virtex-6 devices
Source Code Provided	Yes
Source Code Format	Verilog
Design Uses Code/IP from an Existing Reference Design/Application Note, Third Party, or CORE Generator™ software	No
Simulation	
Functional Simulation Performed	Yes
Timing Simulation Performed	No
Testbench Used for Functional Simulations Provided	Yes
Testbench Format	Verilog
Simulator Software Used/Version (e.g., ISE software, Mentor, Cadence, other)	Mentor

Table 12: **Reference Design Checklist *(Cont'd)***

Parameter	Description
SPICE/IBIS Simulations	No
Implementation	
Synthesis Software Tools Used/Version	XST
Implementation Software Tools Used/Versions	ISE design suite, version 11.4
Static Timing Analysis Performed	Yes
Hardware Verification	
Hardware Verified	Yes
Hardware Used for Verification	ML605 development board

Getting Started with the Demonstration

These steps are performed to run the demo:

1. Connect the USB cables from the PC to the ML605 board.
2. Connect the LVDS ribbon cable to the SAMTEC connectors on the FMC mezzanine card (from QSE1_LA[17:34] to QSE0_LA[0:16])
3. Connect the 5V power supply to the board. This supply is included in the ML605 kit.
4. Program the clock module with a rate equal to the serial rate to connect to the CLK1 slot on the FMC card (shown in Figure 18). Also connect a 200 MHz clean clock (within the 200 ±10 MHz range) to the ML605 board differential USER CLOCK input, as shown in Figure 18.
5. Power up the ML605 board.
6. Configure the device with the ChipScope Pro analyzer using the ChipScope Pro analyzer project file (`SFI4.1_EA_DESIGN.cpj`) provided in the ChipScope Pro analyzer folder. This brings up the predefined VIO GUI and ILA signals to monitor/control.
7. Select the pattern to transmit from the list provided in Table 16, page 32 using the PATTERN_SEL tab.
8. Select the channel on which errors are monitored using the ERROR_COUNT_CH_SEL tab.
9. Set TAP_COUNT_CHAN_SEL to `11111` to view the TAP count of the clock channel. Also change CAPTURE_TAP_COUNT to `1` to continuously monitor TAP count on the TAP_COUNT tab.
10. Start the test by pressing the RESET pushbutton followed by the RESET_DETECTOR pushbutton on the VIO GUI window.

Appendix

Refer to Interface Characterization, page 17 for explanations about interpreting the data in this section. Table 13 through Table 15 show *collective* data eyes for the entire interface over process, voltage, and temperature (PVT). This section contains the characterization data for one device from the -3 and -2 speed grades.

The calibrated center position of the eye (marked by the “C” in Table 13 through Table 15) refers to the calibrated delay setting of the clock channel. Analysis of these three tables show that the center tap setting for slow and fast parts differ significantly at the same data rate (because the BUS_ALIGN_MACHINE algorithm adjusts for differences in speed).

All data in this appendix is collected using the hardware testbench shown in Figure 2, page 2. The data pattern is PRBS23 and the device is an XC6VLX240T in an FF1156 package. The transmission path for the cases shown in Table 13 through Table 15 is repeated here for convenience:

TX → 5" FR-4 → SAMTEC QSE Conn → 12" Ribbon Cable → SAMTEC QSE Conn → 5" FR-4 → RX

Each part was tested at 620, 700, and 730 Mb/s.

Table 13: **Measurement of Receiver Drift under Extreme Conditions at 620 Mb/s**

Condition	IODELAY Taps[1]																						
	–11	–10	–9	–8	–7	–6	–5	–4	–3	–2	–1	0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+10	+11
S/N 7, Speed Grade: -3																							
25°C, Nom Supplies	P	P	P	P	P	P	P	P	P	P	P	C	P	P	P	P	P	P	P	P	P	P	F
0°C, Nom Supplies	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F
85°C, Nom Supplies	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F
25°C, +5% Supplies	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
0°C, +5% Supplies	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
85°C, +5% Supplies	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
25°C, –5% Supplies	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F
0°C, –5% Supplies	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F	F
85°C, –5% Supplies	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F
Center Tap Setting	Tap 2																						
Conclusion	Error free under all conditions when calibrated under nominal conditions to position C																						
S/N 3, Speed Grade: -2																							
25°C, Nom Supplies	F	F	F	F	P	P	P	P	P	P	P	C	P	P	P	P	P	P	P	P	P	F	F
0°C, Nom Supplies	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F
85°C, Nom Supplies	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F
25°C, +5% Supplies	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
0°C, +5% Supplies	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
85°C, +5% Supplies	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F
25°C, –5% Supplies	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F	F	F
0°C, –5% Supplies	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F	F	F
85°C, –5% Supplies	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F	F	F
Center Tap Setting	Tap 19																						

Table 13: **Measurement of Receiver Drift under Extreme Conditions at 620 Mb/s *(Cont'd)***

Condition	IODELAY Taps[1]																				
	–11	–10	–9	–8	–7	–6	–5	–4	–3	–2	–1	0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+10	+11
Conclusion	Error free under all conditions when calibrated under nominal conditions to position C																						

Notes:

1. IODELAY taps, where "0" is the reference tap determined by the bus-align machine at nominal voltage and 25°C.
2. P = Error-free transmission; F = Error in transmission; C = IODELAY tap position selected by the bus-align machine under nominal conditions at 25°C.
3. If the center tap setting is too low to measure the full RX data eye, the width of an eye further along in the delay chain is measured and used to fill in part of the table.

Table 14: **Measurement of Receiver Drift under Extreme Conditions at 700 Mb/s**

Condition	IODELAY Taps[1]																						
	–11	–10	–9	–8	–7	–6	–5	–4	–3	–2	–1	0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+10	+11
S/N 7, Speed Grade: -3																							
25°C, Nom Supplies	F	F	F	F	F	F	F	F	P	P	P	C	P	P	P	P	P	P	P	P	P	P	P
0°C, Nom Supplies	F	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
85°C, Nom Supplies	F	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
25°C, +5% Supplies	F	F	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P
0°C, +5% Supplies	F	F	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P
85°C, +5% Supplies	F	F	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P
25°C, –5% Supplies	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F
0°C, –5% Supplies	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F
85°C, –5% Supplies	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F
Center Tap Setting	Tap 2																						
Conclusion	Error free under all conditions when calibrated under nominal conditions to position C																						
S/N 3, Speed Grade: -2																							
25°C, Nom Supplies	F	F	F	F	F	F	F	P	P	P	P	C	P	P	P	P	P	P	P	P	P	F	F
0°C, Nom Supplies	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F
85°C, Nom Supplies	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F	F
25°C, +5% Supplies	F	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F
0°C, +5% Supplies	F	F	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	F
85°C, +5% Supplies	F	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F
25°C, –5% Supplies	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F	F	F
0°C, –5% Supplies	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F	F	F
85°C, –5% Supplies	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F	F	F
Center Tap Setting	Tap 19																						

Table 14: **Measurement of Receiver Drift under Extreme Conditions at 700 Mb/s *(Cont'd)***

Condition	IODELAY Taps[1]																				
	–11	–10	–9	–8	–7	–6	–5	–4	–3	–2	–1	0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+10	+11
Conclusion	Error free under all conditions when calibrated under nominal conditions to position C																						

Notes:

1. IODELAY taps, where "0" is the reference tap determined by the bus-align machine at nominal voltage and 25°C.
2. P = Error-free transmission; F = Error in transmission; C = IODELAY tap position selected by the bus-align machine under nominal conditions at 25°C.
3. If the center tap setting is too low to measure the full RX data eye, the width of an eye further along in the delay chain is measured and used to fill in part of the table.

Table 15: **Measurement of Receiver Drift under Extreme Conditions at 730 Mb/s**

Condition	IODELAY Taps[1]																						
	–11	–10	–9	–8	–7	–6	–5	–4	–3	–2	–1	0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+10	+11
S/N 7, Speed Grade: -3																							
25°C, Nom Supplies	F	F	F	F	F	F	P	P	P	P	P	C	P	P	P	P	P	P	P	P	P	P	P
0°C, Nom Supplies	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
85°C, Nom Supplies	F	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
25°C, +5% Supplies	F	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
0°C, +5% Supplies	F	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
85°C, +5% Supplies	F	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F
25°C, –5% Supplies	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F	F
0°C, –5% Supplies	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F	F
85°C, –5% Supplies	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F
Center Tap Setting	Tap 2																						
Conclusion	Error free under all conditions when calibrated under nominal conditions to position C																						
S/N 3, Speed Grade: -2																							
25°C, Nom Supplies	F	F	F	F	F	F	F	P	P	P	P	C	P	P	P	P	P	P	P	P	P	F	F
0°C, Nom Supplies	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F
85°C, Nom Supplies	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F	F
25°C, +5% Supplies	F	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	P	F
0°C, +5% Supplies	F	F	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	F	F
85°C, +5% Supplies	F	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F
25°C, –5% Supplies	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	P	F	F	F	F
0°C, –5% Supplies	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	F	F	F	F
85°C, –5% Supplies	F	F	F	F	F	F	F	P	P	P	P	P	P	P	P	P	P	P	P	F	F	F	F
Center Tap Setting	Tap 19																						

Table 15: **Measurement of Receiver Drift under Extreme Conditions at 730 Mb/s *(Cont'd)***

Condition	IODELAY Taps[1]																				
	–11	–10	–9	–8	–7	–6	–5	–4	–3	–2	–1	0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+10	+11
Conclusion	Error free under all conditions when calibrated under nominal conditions to position C																						

Notes:

1. IODELAY taps, where "0" is the reference tap determined by the bus-align machine at nominal voltage and 25°C.
2. P = Error-free transmission; F = Error in transmission; C = IODELAY tap position selected by the bus-align machine under nominal conditions at 25°C.
3. If the center tap setting is too low to measure the full RX data eye, the width of an eye further along in the delay chain is measured and used to fill in part of the table.

Hardware Testbench Design Details

Information covering the clocking, the design hierarchy, and getting started with the demonstration are provided in this section.

Clocking

The reference design requires two clock sources. One clock source is the calibration clock to IDELAY of the receiver design. This source is a clean clock with a frequency of 200 ±10 MHz going to IDELAYCTRL, which controls the precision of the tap delay of the IDELAY. The other clock is a serial clock with a rate equal to the transmit rate.

The differential serial clock comes from the FMC mezzanine card via an SMA connector, as shown in Figure 18. The differential calibration clock is connected to a SMA connector on the ML605 board.

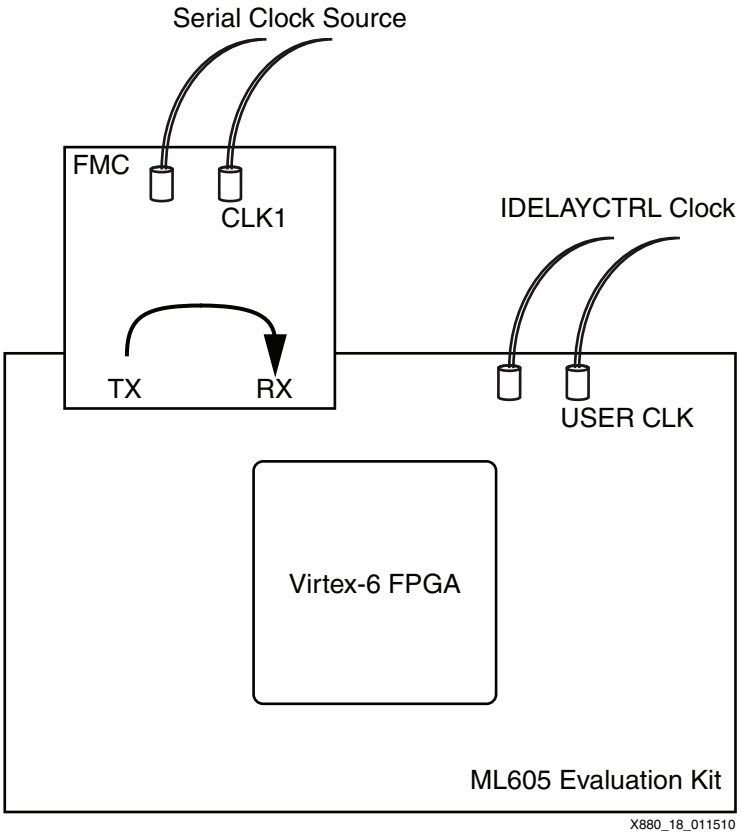


Figure 18: **ML605 Clock Connection Locations**

Design Hierarchy

Figure 19 shows the HDL source file hierarchy of the hardware testbench.

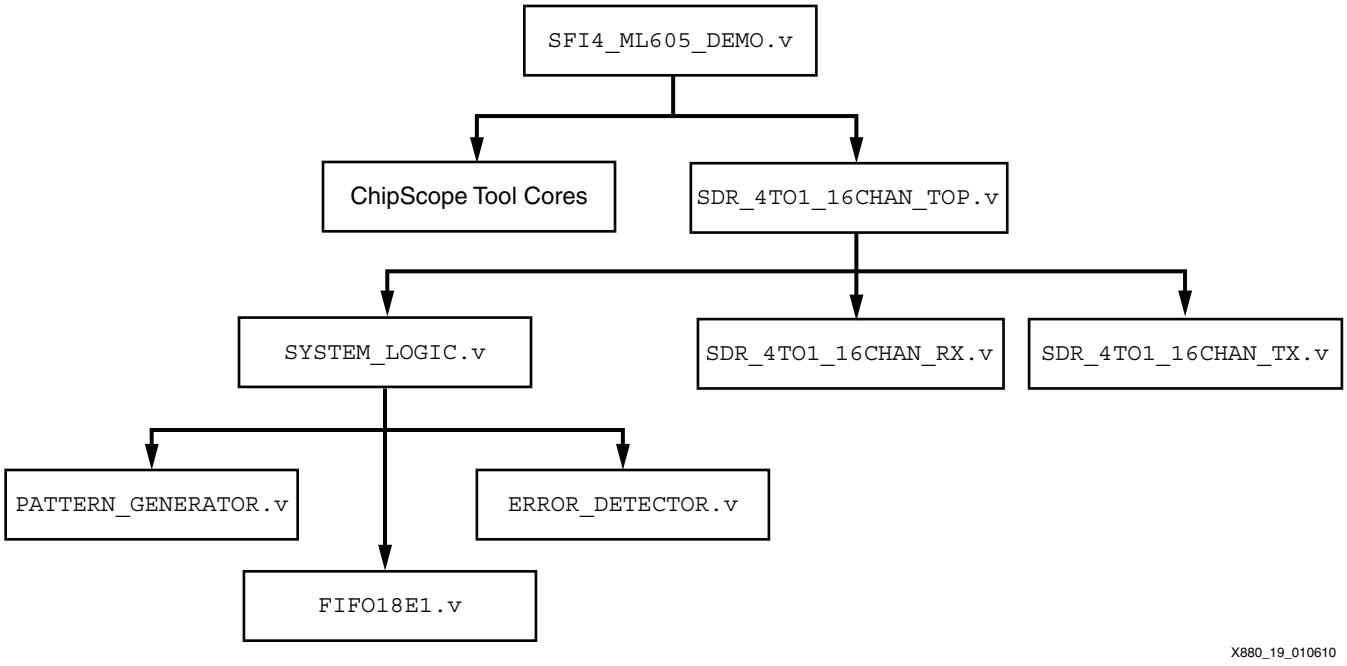


Figure 19: **HDL Source File Hierarchy of Hardware Testbench**

Figure 20 shows the complete hardware testbench, including clocking.

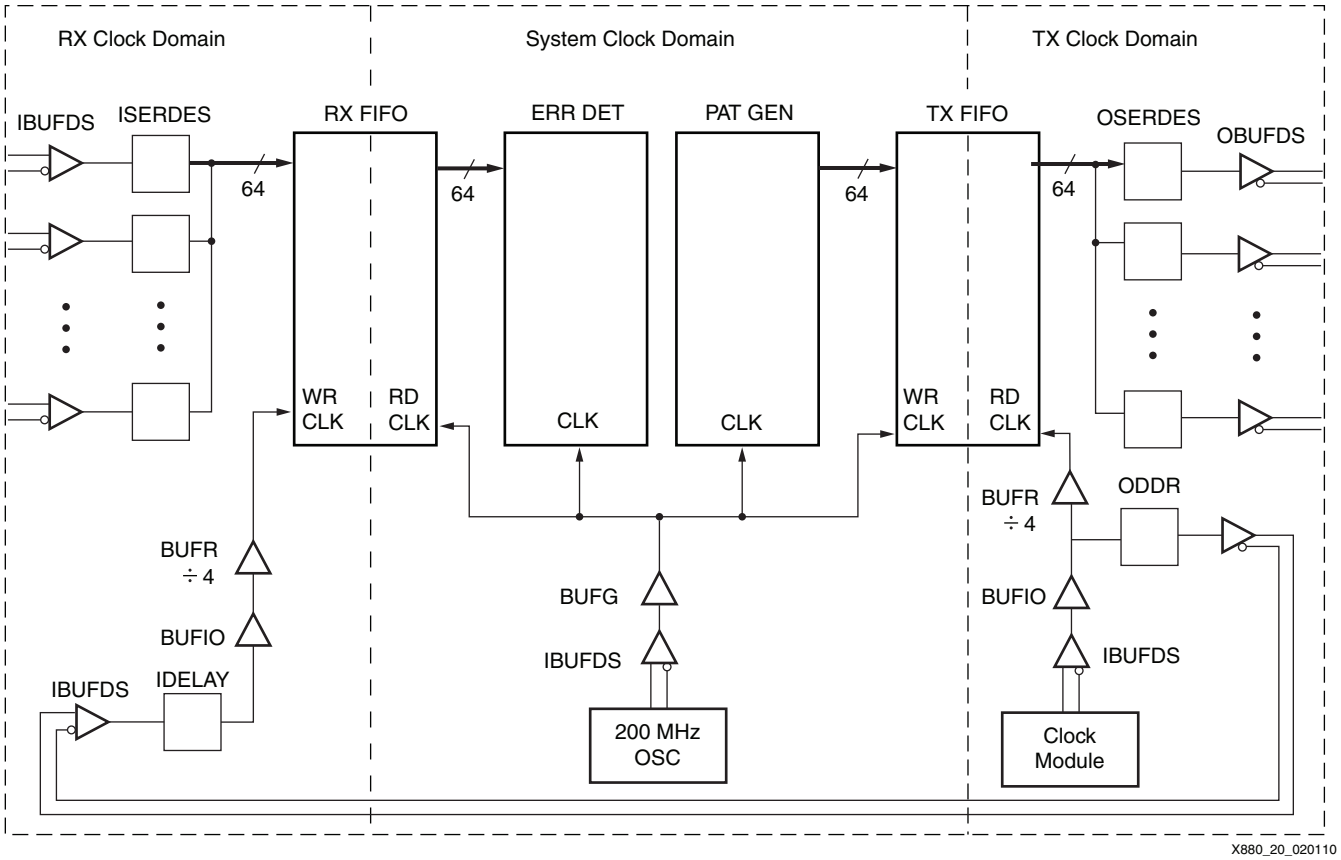


Figure 20: **Clock Domains Driving an SFI-4.1 Interface**

Communication between the system clock domain and the interface clock domains is achieved using FIFO18E1 IP. As shown in Figure 20, the system clock writes data into the TX FIFO at 200 MHz, and the TX clock reads data from the FIFO at the TXCLKDIV rate. TXCLKDIV is determined by the serialization ratio of the interface. For example, with a 4:1 serialization and 700 MHz SDR interface, TXCLKDIV is 175 MHz. Similarly, on the receive side, the system clock reads data from the RX FIFO at 200 MHz, and the RX clock writes data into the FIFO at the RXCLKDIV rate. RXCLKDIV is also determined by the serialization ratio of the interface. For example, with a 4:1 serialization and 700 MHz SDR interface, RXCLKDIV is 175 MHz.

The system clock must be faster than the divided interface clock. If the divided clock of the interface runs at a rate faster than 200 MHz, the design suffers a hard failure. The interfaces transmit and receive data at a constant rate and do not have the flexibility to accommodate almost-full or almost-empty FIFOs. Instead, the system clock domain provides this flexibility, ceasing to read from the RX FIFO if the FIFO raises an almost-empty flag. Similarly, on the transmit side, the system clock ceases to write to the TX FIFO if the FIFO raises an almost full flag. This same 200 MHz clock drives the IDELAYCTRL (not shown in Figure 20), which in turn determines the precision of the TAP delay value of IDELAY.

The ChipScope Pro analyzer monitors and controls the data flow via the system clock domain using the VIO. The VIO console GUI is shown in Figure 21.

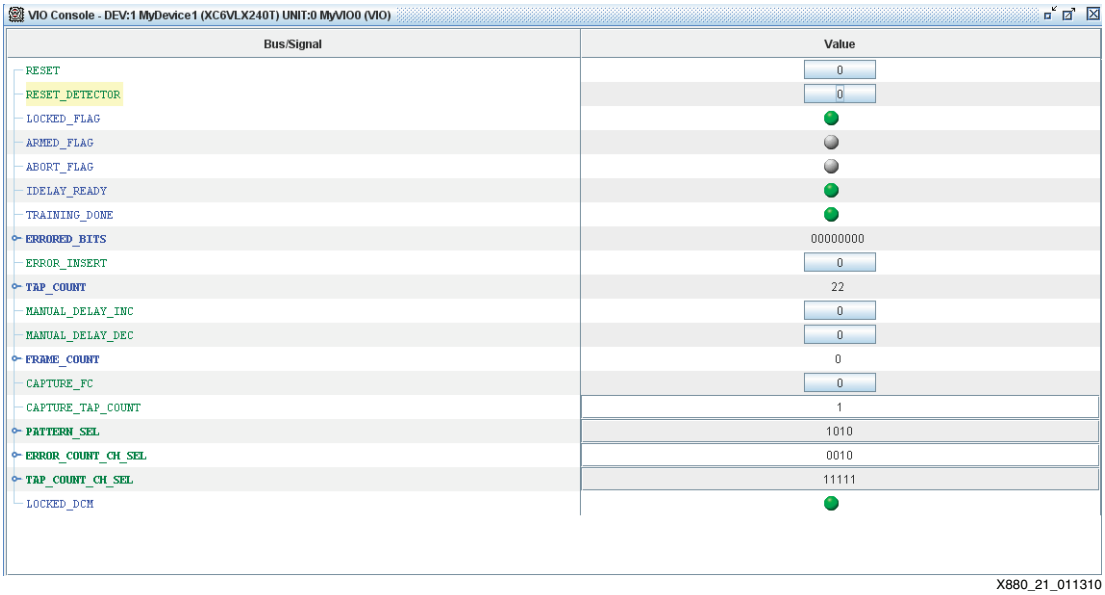


Figure 21: **ChipScope Pro Analyzer VIO GUI**

The user can start the test by pressing RESET pushbutton followed by RESET_DETECTOR pushbutton. This reset sequence resets the entire system, including the error detectors.

A variety of data patterns for simulating data across the link can be generated using the PATTERN_GENERATOR block. The data pattern generated is determined by the PATTERN_SEL tab, as shown in Table 16. Clock patterns can be used to test the link with data of limited frequency content. Pseudorandom patterns provide much more diverse spectral content.

Table 16: **Data Pattern Selection for the PATTERN_GENERATOR Block**

PATTERN_SEL	Data Pattern Selected
0000	Clock Pattern (10101010)
0001	Clock Pattern (11001100)
0010	PRBS3 (00101100)
0011	PRBS7
0100	PRBS15
0101	PRBS23
0110	PRBS29
0111	User Defined (11111110)

The ability to insert errors is provided as a way to check the ability of the receiver to detect errors. When the ERROR_INSERT button is pressed, all data on all channels is inverted for one clock cycle, resulting in an entire frame of errors. These errors appear numerically in the VIO console under the ERRORED_BITS tab. The ERROR_COUNT_CH_SEL tab selects the channel on which errors are monitored.

Errors might not be inserted every time the button is pressed. This is because the system clock domain runs faster than the interface domain clock and does not write data to the FIFO on every clock cycle. If ERROR_INSERT is asserted during a blank cycle, no errors reach the interface.

The VIO console shows these parameters:

- Frames Received (FRAME_COUNT): the number of data frames received while the receiver is locked to the incoming data. VIO latches this value based on the latch enable pressed i.e., CAPTURE_FC tab.
- Bit Error Count (ERRORED_BITS): the sum of all bit errors received on all channels.

There are six VIO flags in total. Three are related to the Link Diagnostics. Their states are indicated by a red or green LED. A green LED indicates the state required for proper operation of the link. After pressing Reset buttons, all three LEDs should be green, indicating these states for the flags:

- Delay Control Ready (IDELAY_READY)
  - The reference clock to the IDLYCTRL module is present and stable.
  - DLY tap values are nominal (78 ps).
  - Bus alignment in the receiver is ready to begin.
- DCM Locked (LOCKED_DCM)
  - System clock is present and stable.
  - The locked signal has initiated resets in the entire system.
- Training Done (TRAINING_DONE)
  - Bus alignment is complete.
  - Data is being transmitted, and error detectors are processing the received data.

The remaining three flags are related to the Error Detector. Their states are also indicated by a red or green LED.

- LOCKED_FLAG

  This flag indicates when the detector is synchronized to data from the transmitter. The detector compares a single frame of data with each incoming data frame from the transmitter. When it sees the matching frame, the receiver pattern synchronizes to the incoming pattern, making each subsequent frame also match. This state is indicated by the LOCKED flag. If the detector has locked to the incoming signal, the LED is green. Otherwise, it is off.
- ARMED_FLAG

  This flag indicates the ARMED state. When one or more errors are detected in a frame, the link temporarily enters the ARMED state. This state is indicated by the ARMED flag, indicated by a red LED. Otherwise, the LED is off.
- ABORT_FLAG

  This flag indicates the ABORT state. If two consecutive frames contain errors, the detector considers the link to have too many errors to remain locked to the incoming data. This state is indicated by the ABORT flag (LED is red), and the link does not recover from this state until the detector is reset. If not in the ABORT state, the LED is off.

References

This document uses these references:

1. SFI-4 (OC-192 Serdes-Framer Interface) OIF-PLL-02.0 - *Proposal for a common electrical interface between SONET framer and serializer/deserializer parts for OC-192 interfaces),* http://www.oiforum.com/public/documents/OIF-SFI4-01.0.pdf.
2. DS152, *Virtex-6 FPGA Data Sheet: DC and Switching Characteristics*.
3. XAPP855, *16-Channel, DDR LVDS Interface with Per-Channel Alignment*.
4. XAPP860, *16-Channel, DDR LVDS Interface with Real-Time Window Monitoring*.

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
02/10/10	1.0	Initial Xilinx release.

Notice of Disclaimer