Timing Constraints Methodology for Source-Synchronous Interfaces

Application Note

FPGA-AN-02078-1.0

January 2024

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, with all faults, and all associated risk is the responsibility entirely of the Buyer. The information provided herein is for informational purposes only and may contain technical inaccuracies or omissions, and may be otherwise rendered inaccurate for many reasons, and Lattice assumes no obligation to update or otherwise correct or revise this information. 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. LATTICE PRODUCTS AND SERVICES ARE NOT DESIGNED, MANUFACTURED, OR TESTED FOR USE IN LIFE OR SAFETY CRITICAL SYSTEMS, HAZARDOUS ENVIRONMENTS, OR ANY OTHER ENVIRONMENTS REQUIRING FAIL-SAFE PERFORMANCE, INCLUDING ANY APPLICATION IN WHICH THE FAILURE OF THE PRODUCT OR SERVICE COULD LEAD TO DEATH, PERSONAL INJURY, SEVERE PROPERTY DAMAGE OR ENVIRONMENTAL HARM (COLLECTIVELY, "HIGH-RISK USES"). FURTHER, BUYER MUST TAKE PRUDENT STEPS TO PROTECT AGAINST PRODUCT AND SERVICE FAILURES, INCLUDING PROVIDING APPROPRIATE REDUDANCIES, FAIL-SAFE FEATURES, AND/OR SHUT-DOWN MECHANISMS. LATTICE EXPRESSLY DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY OF FITNESS OF THE PRODUCTS OR SERVICES FOR HIGH-RISK USES. 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

Acronyms in This Document

A list of acronyms used in this document.

Acronym	Definition
DDR	Double Data Rate
MAX	Maximum
MIN	Minimum
SDC	Synopsys Design Constraints
SDR	Single Data Rate
PLL	Phase-Locked Loop
UI	Unit Interface

1. Introduction

This application note describes the timing constraints methodology for source-synchronous interfaces. In source-synchronous interfaces, the clock signal transmits or receives in parallel with the data signal, as shown in Figure 1.1.

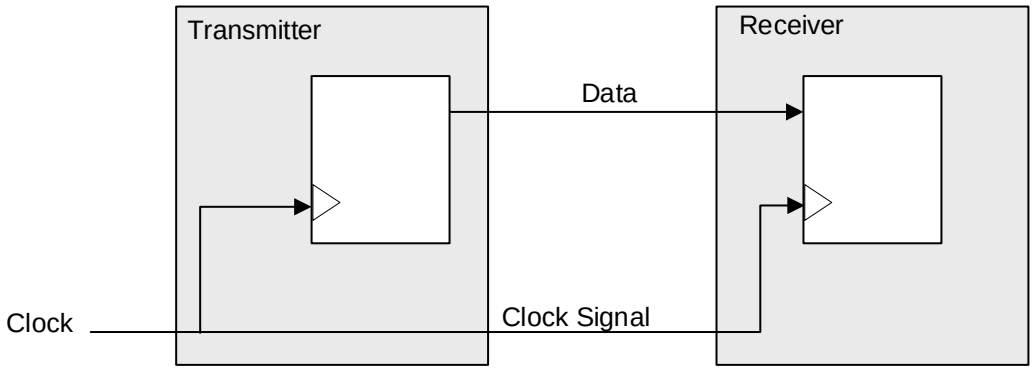

Figure 1.1. Basic Source-Synchronous Interface

A source-synchronous interface has low skew between the clock and data path. Source-synchronous interfaces are widely used for high-speed data transfer. Timing constraints for source-synchronous interfaces can be quite complex as these interfaces can run at single data rate (SDR) and double data rate (DDR).

Note: This application note only describes the timing constraints methodology for SDR source-synchronous interfaces. The timing constraints details for DDR source-synchronous interfaces will be provided in a later revision of this application note.

2. Source-Synchronous Interfaces

This section describes the high-level concept of source-synchronous interfaces. It covers the clock and data relationship, SDR vs. DDR, details of the timing constraints for SDR source-synchronous input and output interfaces, and the Synopsys Design Constraints (SDC) and timing exceptions for source-synchronous interfaces.

2.1. Clock and Data Relationship

The clock-to-data relationship for source-synchronous interfaces can be edge-aligned or center-aligned with the data. An edge-aligned source-synchronous interface means the clock and data signals are aligned at the edge of each other. Figure 2.1 shows the waveform diagram of the edge-aligned source-synchronous interface. Meanwhile, the center-aligned source-synchronous interface has its clock shifted by 90 degree whereby the clock edge is centered to the data signal, as shown in Figure 2.2. Note that for SDR, the clock is shifted by 180 degree to center-aligned the clock-to-data relationship.

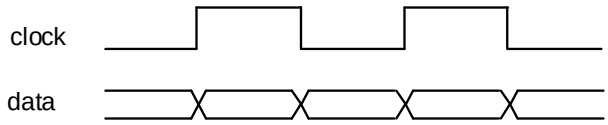

Figure 2.1. Edge-Aligned Clock and Data

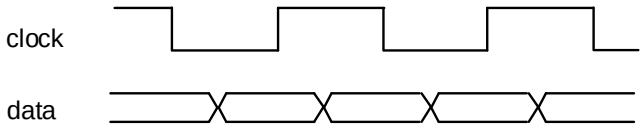

Figure 2.2. Center-Aligned Clock and Data

2.2. Single Data Rate and Double Data Rate

In SDR source-synchronous interfaces, only one edge of the clock is active (either the rising or falling edge). Figure 2.3 shows the waveform diagram of the SDR source-synchronous interface. The time required to transmit one bit, known as the unit interval (UI), is equal to the period of the clock.

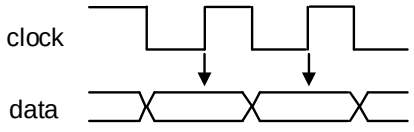

Figure 2.3. SDR with the Rising Edge of an Active Clock

In a DDR source-synchronous interface, data is transferred on both edges of the clock, as shown in Figure 2.4. The UI for DDR is equal to half the period of the clock, assuming a 50/50 duty cycle.

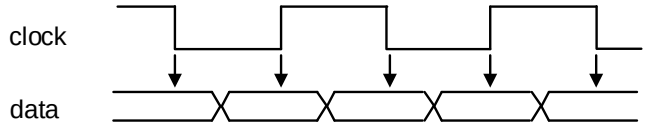

Figure 2.4. DDR with Both Rising and Falling Clock Edges

2.3. Interface Constraints

Source-synchronous interfaces require the following three types of SDC constraints:

- **Clock constraints** (`create_clock` and `create_generated_clock`) to define the clock referenced to the interface.
- **Input and output delay constraints** (`set_input_delay` and `set_output_delay`) to specify the required times for data to be valid at the interface.
- **Timing exceptions** (`set_false_path` and `set_multicycle_path`) to ensure the invalid paths are not analyzed.

3. Constraints for SDR Source-Synchronous Interface

This section describes the following SDC constraints for SDR source-synchronous interface:

- SDR source-synchronous output constraints
- SDR source-synchronous input constraints

3.1. Source-Synchronous Output Constraints

Figure 3.1 shows the circuit diagram of an SDR source-synchronous output interface. For source-synchronous interface, the clock and data are fed in parallel to the external device, as shown in Figure 3.1.

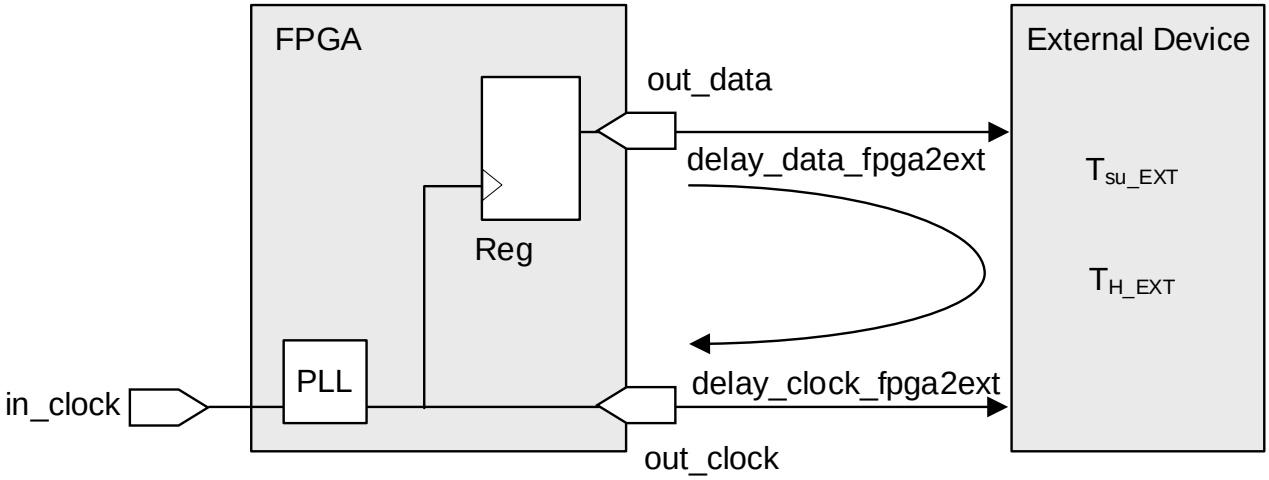

Figure 3.1. SDR Source-Synchronous Output Interface with PLL

3.1.1. Output Clock Constraints

For source-synchronous output interface, it is required to create a generated clock at the FPGA output clock port. The generated clock is used as the clock reference for output delay constraints. Figure 3.2 shows the clock constraints for the SDR source-synchronous output interface.

```
create_clock -name in_clock -period <clock_period> [get_ports in_clock]
create_generated_clock -name sdr_clock -source [get_pins pll|refclk] [get_pins pll|outclk_0]
create_generated_clock -name sdr_clock_out -source [get_pins pll|outclk_0] [get_ports
out_clock]
```

Figure 3.2. Clock Constraints for SDC Source-Synchronous Output Interface with PLL

3.1.2. Output Delay Constraints

Constraining output delay requires external timing parameters. The trace delay on board for clock and data as well as the setup and hold time requirements of an external device are needed, as shown in Figure 3.1. Figure 3.3 shows the SDC constraints for SDR source-synchronous output interface.

```
set_output_delay -max <max output delay> -clock [get_clocks sdr_clock_out] [get_ports out_data]
set_output_delay -min <min output delay> -clock [get_clocks sdr_clock_out] [get_ports out_data]
```

Figure 3.3. MAX and MIN Output Delay Constraints for a SDR Output

The formula to compute the maximum and minimum value of the output delay is shown in Figure 3.4. Figure 3.4 shows that the input setup and hold timing parameters for the external device are needed to compute the value of the MAX and MIN output delay.

```
output delay (max) = delay_data_fpga2ext(max) + t_SU_EXT – delay_clock_fpga2ext(min)
output delay (min) = delay_data_fpga2ext(min) - t_H_EXT + delay_clock_fpga2ext(max)
```

Figure 3.4. Formula to Compute Maximum and Minimum Value of the Output Delay

When using the data-to-clock skew ($Tdata_{skew}$) timing parameter instead, follow the equation in Figure 3.5 to calculate the MAX and MIN value of the output delay.

```
output delay (max) = setup_latch_clock – setup_launch_clock - Tdata_skew
output delay (min) = -(hold_launch_clock – hold_latch_clock) + Tdata_skew

# For center-aligned interface
setup_latch_clock – setup_launch_clock = clock_period/2
hold_launch_clock – hold_latch_clock   = clock_period/2

output delay (max) = clock_period/2 - Tdata_skew
output delay (min) = -clock_period/2 + Tdata_skew

# For edge-aligned interface
setup_latch_clock – setup_launch_clock = 0
hold_launch_clock – hold_latch_clock   = clock_period

output delay (max) = - Tdata_skew
output delay (min) = - clock_period + Tdata_skew
```

Figure 3.5. Formula for the Maximum and Minimum Value of Output Delay Using $Tdata_{skew}$

3.1.3. Case Study

This section describes the following SDC constraints examples or case studies for different situations of SDR source-synchronous output interface:

- Case 1: Edge-aligned interface
- Case 2: Center-aligned interface with external device clock phase shift
- Case 3: Center-aligned interface with FPGA phase-locked loop (PLL) clock phase shift
- Case 4: Using $Tdata_{skew}$

The following lists the characteristics of the external device for Case 1 to Case 3:

- Receiving SDR data at 100 MHz
- External device has T_{su_EXT} of 1.5 ns and T_{H_EXT} of 0 ns

3.1.3.1. Case 1 - Edge-Aligned Interface

There is no clock phase shift anywhere in the system to center-aligned the data and clock. Therefore, the setup relationship will be a full clock period and hold relationship will be 0 ns. Based on the formula in Figure 3.4, the MAX and MIN value of the output delay is shown in the following equation. Trace delay on board is assumed to be 0 ns since it wasn't given.

Setup relationship = 10 ns
Hold relationship = 0 ns

output_delay (max)
= delay_data_fpga2ext(max) + T_{SU_EXT} –delay_clock_fpga2ext(min)
= 0 + 1.5 – 0
= 1.5 ns
output_delay (min)
= delay_data_fpga2ext(min) - T_{H_EXT} –delay_clock_fpga2ext(max)
= 0 + 0 – 0
= 0 ns

Figure 3.6 shows the complete SDC constraints for Case 1: Edge-aligned output interface.

```
create_clock -name in_clock -period 10.000 [get_ports in_clock]

# We do no need to create PLL generated clock. It will be added by Radiant automatically.
create_generated_clock -name sdr_clock -source [get_pins pll|refclk] [get_pins pll|outclk_0]

create_generated_clock -name sdr_clock_out -source [get_pins pll|outclk_0] [get_ports
out_clock]
set_output_delay -max 1.5 -clock [get_clocks sdr_clock_out] [get_ports out_data]
set_output_delay -min 0 -clock [get_clocks sdr_clock_out] [get_ports out_data]
```

Figure 3.6. SDC Constraints for Case 1: Edge-Aligned Output Interface

The setup and hold timing results are shown in Figure 3.7. As shown in Figure 3.7, the source-synchronous output interface has difficulty to close timing for hold time. The FPGA clock path routing delay to the I/O interface is generally much larger compared to I/O registers to I/O interface.

	Start Point	End Point	Setup/Hold Constraint	Slack	Delay	Source Clock	Destination Clock	Analysis Type
1	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[0].u_data_OFD1P3DX.PIC_inst/Q	c_out[0]	10000	10550	-550	clkop	clkos_o	setup
2	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[1].u_data_OFD1P3DX.PIC_inst/Q	c_out[1]	10000	10593	-593	clkop	clkos_o	setup
3	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[2].u_data_OFD1P3DX.PIC_inst/Q	c_out[2]	10000	10593	-593	clkop	clkos_o	setup
4	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[3].u_data_OFD1P3DX.PIC_inst/Q	c_out[3]	10000	10593	-593	clkop	clkos_o	setup

	Start Point	End Point	Setup/Hold Constraint	Slack	Delay	Source Clock	Destination Clock	Analysis Type
1	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[0].u_data_OFD1P3DX.PIC_inst/Q	c_out[0]	0	-1429	1429	clkop	clkos_o	hold
2	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[1].u_data_OFD1P3DX.PIC_inst/Q	c_out[1]	0	-1455	1455	clkop	clkos_o	hold
3	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[2].u_data_OFD1P3DX.PIC_inst/Q	c_out[2]	0	-1455	1455	clkop	clkos_o	hold
4	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[3].u_data_OFD1P3DX.PIC_inst/Q	c_out[3]	0	-1455	1455	clkop	clkos_o	hold

Figure 3.7. Setup and Hold Timing Results for Case 1 SDR Source-Synchronous Output Interface

3.1.3.2. Case 2 - Center-Aligned Interface with External Device Clock Phase Shift

The external device is performing the clock phase shift of 180 degree to center-align the latching data. Therefore, the setup relationship will become 5 ns and hold become -5 ns. Based on the formula in Figure 3.4, the MAX and MIN value of output delay is shown in the following equation. Trace delay on board is assumed to be 0 ns since it wasn't given.

Setup relationship = 5 ns
Hold relationship = -5 ns

output_delay(max)
= delay_data_fpga2ext(max) + T_{SU_EXT} –delay_clock_fpga2ext(min)
= 0 + 1.5 – 0
= 1.5 ns
output_delay(min)
= delay_data_fpga2ext(min) - T_{H_EXT} –delay_clock_fpga2ext(max)
= 0 + 0 – 0
= 0 ns

Figure 3.8. shows the complete SDC constraints for Case 2: Center-aligned interface with external device clock phase shift. The `-invert` is used to inform the Timing Analyzer that the external device is shifting the clock by 180 degree.

```
create_clock -name in_clock -period 10.000 [get_ports in_clock]

# We do no need to create PLL generated clock. It will be added by Radiant automatically.
create_generated_clock -name sdr_clock -source [get_pins pll|refclk] [get_pins pll|outclk_0]

create_generated_clock -name sdr_clock_out -source [get_pins pll|outclk_0] -invert [get_ports
out_clock]
set_output_delay -max 1.5 -clock [get_clocks sdr_clock_out] [get_ports out_data]
set_output_delay -min 0 -clock [get_clocks sdr_clock_out] [get_ports out_data]
```

Figure 3.8. SDC Constraints for Case 2: Center-Aligned Interface with External Device Clock Phase Shift

The setup and hold timing results are shown in Figure 3.9. As shown in Figure 3.9, there is no hold time violation when the data is center-aligned to its clock.

	Start Point	End Point	Setup/Hold Constraint	Slack	Delay	Source Clock	Destination Clock	Analysis Type
1	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[0].u_data_OFD1P3DX.PIC_inst/Q	c_out[0]	5000	5550	-550	clkop	clkos_o	setup
2	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[1].u_data_OFD1P3DX.PIC_inst/Q	c_out[1]	5000	5593	-593	clkop	clkos_o	setup
3	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[2].u_data_OFD1P3DX.PIC_inst/Q	c_out[2]	5000	5593	-593	clkop	clkos_o	setup
4	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[3].u_data_OFD1P3DX.PIC_inst/Q	c_out[3]	5000	5593	-593	clkop	clkos_o	setup

	Start Point	End Point	Setup/Hold Constraint	Slack	Delay	Source Clock	Destination Clock	Analysis Type
1	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[0].u_data_OFD1P3DX.PIC_inst/Q	c_out[0]	-5000	3570	-8570	clkop	clkos_o	hold
2	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[1].u_data_OFD1P3DX.PIC_inst/Q	c_out[1]	-5000	3544	-8544	clkop	clkos_o	hold
3	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[2].u_data_OFD1P3DX.PIC_inst/Q	c_out[2]	-5000	3544	-8544	clkop	clkos_o	hold
4	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[3].u_data_OFD1P3DX.PIC_inst/Q	c_out[3]	-5000	3544	-8544	clkop	clkos_o	hold

Figure 3.9. Setup and Hold Timing Results for Case 2 SDR Source-Synchronous Output Interface

3.1.3.3. Case 3 - Center-Aligned Interface with the FPGA PLL Clock Phase Shift

The FPGA PLL performs the clock phase shift of 180 degree to center-align the transfer data. Therefore, the setup relationship will become 5 ns and hold become -5 ns. Based on the formula in Figure 3.4, the MAX and MIN value of the output delay is shown in the following equation. Trace delay on board is assumed to be 0 ns since it wasn't given.

Setup relationship = 5 ns

Hold relationship = -5 ns

output_delay(max)

= delay_data_fpga2ext(max) + T_{SU_EXT} –delay_clock_fpga2ext(min)

= 0 + 1.5 – 0

= 1.5 ns

output_delay(min)

= delay_data_fpga2ext(min) - T_{H_EXT} –delay_clock_fpga2ext(max)

= 0 + 0 – 0

= 0 ns

Figure 3.10 shows the complete SDC constraints for Case 3: Center-aligned interface with the FPGA PLL clock phase shift.

```
create_clock -name in_clock -period 10.000 [get_ports in_clock]

# We do no need to create PLL generated clock. It will be added by Radiant automatically.
create_generated_clock -name sdr_clock -source [get_pins pll|refclk] [get_pins pll|outclk_0] -
phase 180

create_generated_clock -name sdr_clock_out -source [get_pins pll|outclk_0] [get_ports
out_clock]
set_output_delay -max 1.5 -clock [get_clocks sdr_clock_out] [get_ports out_data]
set_output_delay -min 0 -clock [get_clocks sdr_clock_out] [get_ports out_data]
```

Figure 3.10. SDC Constraints for Case 3: Center-Aligned Interface with FPGA PLL Clock Phase Shift

The setup and hold timing results are shown in Figure 3.11. As shown in Figure 3.11, there is no hold time violation when the data is center-aligned to its clock.

	Start Point	End Point	Setup/Hold Constraint	Slack	Delay	Source Clock	Destination Clock	Analysis Type
1	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[0].u_data_OFD1P3DX.PIC_inst/Q	c_out[0]	5000	5550	-550	clkop	clkos_o	setup
2	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[1].u_data_OFD1P3DX.PIC_inst/Q	c_out[1]	5000	5593	-593	clkop	clkos_o	setup
3	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[2].u_data_OFD1P3DX.PIC_inst/Q	c_out[2]	5000	5593	-593	clkop	clkos_o	setup
4	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[3].u_data_OFD1P3DX.PIC_inst/Q	c_out[3]	5000	5593	-593	clkop	clkos_o	setup

	Start Point	End Point	Setup/Hold Constraint	Slack	Delay	Source Clock	Destination Clock	Analysis Type
1	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[0].u_data_OFD1P3DX.PIC_inst/Q	c_out[0]	-5000	3570	-8570	clkop	clkos_o	hold
2	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[1].u_data_OFD1P3DX.PIC_inst/Q	c_out[1]	-5000	3544	-8544	clkop	clkos_o	hold
3	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[2].u_data_OFD1P3DX.PIC_inst/Q	c_out[2]	-5000	3544	-8544	clkop	clkos_o	hold
4	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[3].u_data_OFD1P3DX.PIC_inst/Q	c_out[3]	-5000	3544	-8544	clkop	clkos_o	hold

Figure 3.11. Setup and Hold Timing Results for Case 3 SDR Source-Synchronous Output Interface

When you compare the setup slack and hold slack for Case 1 to Case 3 as shown in Table 3.1, you will notice that Case 1 has hold time violation and it is resolved after the clock and data is center-aligned. The FPGA clock path routing delay to the I/O interface is generally much larger compared to I/O registers. Therefore, it is always better to center-aligned the clock and data somewhere in the system whether by external device or through the FPGA PLL.

Table 3.1. Setup and Hold Slack Comparison for SDR Source-Synchronous Output Interface

Case	Setup Slack (ns)	Hold Slack (ns)
Case 1 - Edge-aligned interface	10.550	-1.429
Case 2 - Center-aligned interface with external device clock phase shift	5.550	3.570
Case 3 - Center-aligned interface with FPGA PLL clock phase shift	5.550	3.570

3.1.3.4. Case 4 – Using $Tdata_{skew}$

For Case 4, reuse the Case 3 situation with the following external device characteristics:

- Receiving SDR data at 100 MHz
- External device has $Tdata_{skew}$ of 3.5 ns/-5 ns

For this case, use the formula in Figure 3.5.

Setup relationship = 5 ns
Hold relationship = -5 ns

output_delay(max)
= clock_period/2 - Tdataskew
= 10/2 – 3.5
= 1.5 ns
output_delay(min)
= - clock_period/2 + Tdataskew
= -10/2 + (5)
= 0 ns

Figure 3.12 shows the complete SDC constraints for Case 4: Using $Tdata_{skew}$.

```
create_clock -name in_clock -period 10.000 [get_ports in_clock]

# We do no need to create PLL generated clock. It will be added by Radiant automatically.
create_generated_clock -name sdr_clock -source [get_pins pll|refclk] [get_pins pll|outclk_0] -
phase 180

create_generated_clock -name sdr_clock_out -source [get_pins pll|outclk_0] [get_ports
out_clock]
set_output_delay -max 1.5 -clock [get_clocks sdr_clock_out] [get_ports out_data]
set_output_delay -min 0 -clock [get_clocks sdr_clock_out] [get_ports out_data]
```

Figure 3.12. SDC Constraints for Case 4: Using $Tdata_{skew}$

Figure 3.13 shows the setup and hold timing results.

	Start Point	End Point	Setup/Hold Constraint	Slack	Delay	Source Clock	Destination Clock	Analysis Type
1	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[0].u_data_OFD1P3DX.PIC_inst/Q	c_out[0]	5000	5550	-550	clkop	clkos_o	setup
2	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[1].u_data_OFD1P3DX.PIC_inst/Q	c_out[1]	5000	5593	-593	clkop	clkos_o	setup
3	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[2].u_data_OFD1P3DX.PIC_inst/Q	c_out[2]	5000	5593	-593	clkop	clkos_o	setup
4	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[3].u_data_OFD1P3DX.PIC_inst/Q	c_out[3]	5000	5593	-593	clkop	clkos_o	setup

	Start Point	End Point	Setup/Hold Constraint	Slack	Delay	Source Clock	Destination Clock	Analysis Type
1	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[0].u_data_OFD1P3DX.PIC_inst/Q	c_out[0]	-5000	3570	-8570	clkop	clkos_o	hold
2	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[1].u_data_OFD1P3DX.PIC_inst/Q	c_out[1]	-5000	3544	-8544	clkop	clkos_o	hold
3	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[2].u_data_OFD1P3DX.PIC_inst/Q	c_out[2]	-5000	3544	-8544	clkop	clkos_o	hold
4	osdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_tx_static_bypass/Data[3].u_data_OFD1P3DX.PIC_inst/Q	c_out[3]	-5000	3544	-8544	clkop	clkos_o	hold

Figure 3.13. Setup and Hold Timing Results for Case 4 SDR Source-Synchronous Output Interface

3.2. Source-Synchronous Input Constraints

Figure 3.14 shows the circuit diagram of an SDR source-synchronous input interface. As shown in Figure 3.14, the clock and data are fed in parallel from an external device to the FPGA.

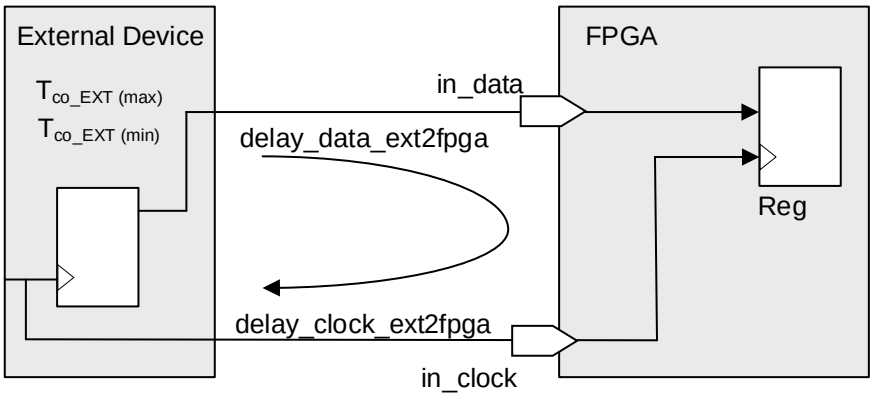

Figure 3.14. SDR Source-Synchronous Input interface

3.2.1. Input Clocks Constraints

For source-synchronous input interface, the clock source is from an external device. Therefore, a virtual clock is used as the clock reference for input delay constraints. Note that it is not necessary for you to use a virtual clock as the clock reference for input delay constraints. You can use the real input clock instead. However, using a virtual clock for input delay constraints makes the constraint easier, especially when dealing with clock shifting.

As described in the Single Data Rate section, the clock and data relationship at the FPGA interface for source-synchronous interface can be edge-aligned or center-aligned. Figure 3.15 and Figure 3.16 show the clock constraints example for edge-aligned and center-aligned SDR source-synchronous input interface.

```
create_clock -name in_clock -period 10.000 [get_ports in_clock]
create_clock -name ext_clock_virtual -period 10.000
```

Figure 3.15. Clock Constraints for Edge-Aligned SDR Source-Synchronous Input Interface

```
create_clock -name in_clock -period 10.000 [get_ports in_clock] -waveform {5 10}
create_clock -name ext_clock_virtual -period 10.000
```

Figure 3.16. Clock Constraints for Center-Aligned SDR Source-Synchronous Input Interface

3.2.2. Input Delay Constraints

Constraining input delay requires external timing parameters. Trace delay on board for clock and data as well as the maximum and minimum Tco of the external device information are required. Figure 3.17 shows the SDC constraints for SDR source-synchronous input interface.

```
set_input_delay -max <max input delay value> -clock [get_clocks ext_clock_virtual] [get_ports
in_data]
set_input_delay -min <min input delay value> -clock [get_clocks ext_clock_virtual] [get_ports
in_data]
```

Figure 3.17. MAX and MIN Input Delay Constraints for SDR Input

The formula to compute the maximum and minimum value of the input delay is shown in Figure 3.18. It uses the longest data path and shortest clock path for maximum input delay calculation and shortest data path and longest clock path for minimum input delay calculation.

$$\text{input delay (max)} = \text{delay\_data\_ext2fpga(max)} + T_{co\_EXT}\text{(max)} - \text{delay\_clock\_ext2fpga(min)}$$
$$\text{input delay (min)} = \text{delay\_data\_ext2fpga(min)} + T_{co\_EXT}\text{(min)} - \text{delay\_clock\_ext2fpga(max)}$$

Figure 3.18. MAX and MIN Input Delay Calculation Using T_{co_EXT} (max) and T_{co_EXT} (min) of an External Device

Sometimes, the setup time (T_{su_EXT}) and hold time (T_{H_EXT}) for the output data of the external device is given instead of T_{co}. In this case, use the equation in Figure 3.19 to compute the maximum and minimum value of the input delay.

$$\text{input delay (max)} = \text{delay\_data\_ext2fpga(max)} + \text{clock\_period} - T_{su\_EXT} - \text{delay\_clock\_ext2fpga(min)}$$
$$\text{input delay (min)} = \text{delay\_data\_ext2fpga(min)} + T_{H\_EXT} - \text{delay\_clock\_ext2fpga(min)}$$

Figure 3.19. MAX and MIN Input Delay Calculation Using T_{su_EXT} and T_{H_EXT} of an External Device

If the output data skew is given instead, use the equation in Figure 3.20 to compute the maximum and minimum value of the input delay.

$$\text{input delay (max)} = Tdata_{skew}$$
$$\text{input delay (min)} = - Tdata_{skew}$$

Figure 3.20. MAX and MIN Input Delay Calculation Using $Tdata_{skew}$ of an External Device

3.2.3. Case Study

This section describes the SDC constraints examples for the following situations of SDR source-synchronous input interface:

- Case 1: Edge-aligned interface
- Case 2: Center-aligned interface with FPGA clock phase shift
- Case 3: Center-aligned interface with an external device performing clock phase shift

Assuming the external device has the following characteristics:

- Transmitting SDR data at 100 MHz
- Data is skewed by +/- 0.3 ns around the clock edge

3.2.3.1. Case 1 - Edge-Aligned Interface

There is no clock phase shift anywhere in the system to center-aligned the input data and clock. Therefore, the setup relationship will be a full clock period and Hold relationship will be 0 ns. Based on the equation in Figure 3.20, the MAX and MIN value of the input delay is as follows:

Setup relationship = 10 ns	Input_delay(max) = 0.3 ns
Hold relationship = 0 ns	Input_delay(min) = -0.3 ns

- Figure 3.21 shows the complete SDC constraints for Case 1: Edge-aligned input interface.

```
create_clock -name in_clock -period 10.000 [get_ports in_clock]
create_clock -name ext_clock_virtual -period 10.000

# We do no need to create PLL generated clock. It will be added by Radiant automatically.
create_generated_clock -name sdr_clock -source [get_pins pll|refclk] [get_pins pll|outclk_0]

set_input_delay -max 0.3 -clock [get_clocks ext_clock_virtual] [get_ports in_data]
set_input_delay -min -0.3 -clock [get_clocks ext_clock_virtual] [get_ports in_data]
```

Figure 3.21. SDC Constraints for Case 1: Edge-Aligned Input Interface

Figure 3.22 shows the setup and hold timing results. The setup slack has a huge margin compared to the hold slack.

	Start Point	End Point	Setup/Hold Constraint	Slack	Delay	Source Clock	Destination Clock	Analysis Type
1	in_data[0]	isdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_rx_static_bypass/Data[0].u_IFD1P3DX.PIC_inst/D	10000	8570	1429	in_clock_vtr	clkos_o_c	setup
2	in_data[1]	isdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_rx_static_bypass/Data[1].u_IFD1P3DX.PIC_inst/D	10000	8579	1420	in_clock_vtr	clkos_o_c	setup
3	in_data[2]	isdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_rx_static_bypass/Data[2].u_IFD1P3DX.PIC_inst/D	10000	8579	1420	in_clock_vtr	clkos_o_c	setup
4	in_data[3]	isdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_rx_static_bypass/Data[3].u_IFD1P3DX.PIC_inst/D	10000	8579	1420	in_clock_vtr	clkos_o_c	setup

	Start Point	End Point	Setup/Hold Constraint	Slack	Delay	Source Clock	Destination Clock	Analysis Type
1	in_data[0]	isdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_rx_static_bypass/Data[0].u_IFD1P3DX.PIC_inst/D	0	355	-355	in_clock_vtr	clkos_o_c	hold
2	in_data[1]	isdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_rx_static_bypass/Data[1].u_IFD1P3DX.PIC_inst/D	0	350	-350	in_clock_vtr	clkos_o_c	hold
3	in_data[2]	isdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_rx_static_bypass/Data[2].u_IFD1P3DX.PIC_inst/D	0	350	-350	in_clock_vtr	clkos_o_c	hold
4	in_data[3]	isdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_rx_static_bypass/Data[3].u_IFD1P3DX.PIC_inst/D	0	350	-350	in_clock_vtr	clkos_o_c	hold

Figure 3.22. Setup and Hold Timing Results for Case 1 SDR Source-Synchronous Input Interface.

3.2.3.2. Case 2 - Edge-Aligned Interface with FPGA Clock Phase Shift

The FPGA PLL performs the clock phase shift of 180 degree to center-align the latching data. Therefore, the setup relationship will become 5 ns and hold become -5 ns. Based on the equation in Figure 3.20, the MAX and MIN value of input delay is as follows:

Setup relationship = 5 ns
Hold relationship = -5 ns

Input_delay(max) = 0.3 ns
Input_delay(min) = -0.3 ns

Figure 3.23 shows the complete SDC constraints for Case 2: Edge-aligned input interface with FPGA clock phase shift.

```
create_clock -name in_clock -period 10.000 [get_ports in_clock]
create_clock -name ext_clock_virtual -period 10.000

# We do no need to create PLL generated clock. It will be added by Radiant automatically
create_generated_clock -name sdr_clock -source [get_pins pll|refclk] [get_pins pll|outclk_0] -
phase 180

set_input_delay -max 0.3 -clock [get_clocks ext_clock_virtual] [get_ports in_data]
set_input_delay -min -0.3 -clock [get_clocks ext_clock_virtual] [get_ports in_data]
```

Figure 3.23. SDC Constraints for Case 2: Edge-Aligned Input Interface with FPGA Clock Phase Shift

Figure 3.24 shows the setup and hold timing results. The setup slack and hold slack is more balanced compared to Case 1.

	Start Point	End Point	Setup/Hold Constraint	Slack	Delay	Source Clock	Destination Cloc	Analysis Type
1	in_data[0]	isdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_rx_static_bypass/Data[0].u_IFD1P3DX.PIC_inst/D	5000	3570	1429	in_clock_vtr	clkos_o_c	setup
2	in_data[1]	isdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_rx_static_bypass/Data[1].u_IFD1P3DX.PIC_inst/D	5000	3579	1420	in_clock_vtr	clkos_o_c	setup
3	in_data[2]	isdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_rx_static_bypass/Data[2].u_IFD1P3DX.PIC_inst/D	5000	3579	1420	in_clock_vtr	clkos_o_c	setup
4	in_data[3]	isdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_rx_static_bypass/Data[3].u_IFD1P3DX.PIC_inst/D	5000	3579	1420	in_clock_vtr	clkos_o_c	setup

	Start Point	End Point	Setup/Hold Constraint	Slack	Delay	Source Clock	Destination Clock	Analysis Type
1	in_data[0]	isdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_rx_static_bypass/Data[0].u_IFD1P3DX.PIC_inst/D	-5000	5355	-10355	in_clock_vtr	clkos_o_c	hold
2	in_data[1]	isdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_rx_static_bypass/Data[1].u_IFD1P3DX.PIC_inst/D	-5000	5350	-10350	in_clock_vtr	clkos_o_c	hold
3	in_data[2]	isdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_rx_static_bypass/Data[2].u_IFD1P3DX.PIC_inst/D	-5000	5350	-10350	in_clock_vtr	clkos_o_c	hold
4	in_data[3]	isdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_rx_static_bypass/Data[3].u_IFD1P3DX.PIC_inst/D	-5000	5350	-10350	in_clock_vtr	clkos_o_c	hold

Figure 3.24. Setup and Hold Timing Results for Case 2 SDR Source-Synchronous Input Interface

3.2.3.3. Case 3 - Center-Aligned Interface with External Device Performing Clock Phase Shift

The external device is transferring the data and clock in center-aligned. Therefore, the setup relationship will become 5 ns and hold become -5 ns. Based on the equation in Figure 3.20., the MAX and MIN value of the input delay is as follows:

Setup relationship = 5 ns
Hold relationship = -5 ns

Input_delay(max) = 0.3 ns
Input_delay(min) = -0.3 ns

Figure 3.25 shows the complete SDC constraints for Case 3: Center-aligned input interface with external device performing a clock phase shift. For this case, the create clock on input clock performs a phase shift by 180 degree. This is to inform the Timing Analyzer that the input clock has performed a phase shift.

```
create_clock -name in_clock -period 10.000 [get_ports in_clock] -waveform {5 10}
create_clock -name ext_clock_virtual -period 10.000

# We do no need to create PLL generated clock. It will be added by Radiant automatically.
create_generated_clock -name sdr_clock -source [get_pins pll|refclk] [get_pins pll|outclk_0]

set_input_delay -max 0.3 -clock [get_clocks ext_clock_virtual] [get_ports in_data]
set_input_delay -min -0.3 -clock [get_clocks ext_clock_virtual] [get_ports in_data]
```

Figure 3.25. SDC Constraints for Case 3: Center-aligned input interface with External Device Performing Clock Phase Shift

Figure 3.26 shows the setup and hold timing results.

	Start Point	End Point	Setup/Hold Constraint	Slack	Delay	Source Clock	estination Clo	Analysis Type
1	in_data[0]	isdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_rx_static_bypass/Data[0].u_IFD1P3DX.PIC_inst/D	5000	3570	1429	in_clock_vtr	clkos_o_c	setup
2	in_data[1]	isdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_rx_static_bypass/Data[1].u_IFD1P3DX.PIC_inst/D	5000	3579	1420	in_clock_vtr	clkos_o_c	setup
3	in_data[2]	isdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_rx_static_bypass/Data[2].u_IFD1P3DX.PIC_inst/D	5000	3579	1420	in_clock_vtr	clkos_o_c	setup
4	in_data[3]	isdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_rx_static_bypass/Data[3].u_IFD1P3DX.PIC_inst/D	5000	3579	1420	in_clock_vtr	clkos_o_c	setup

	Start Point	End Point	Setup/Hold Constraint	Slack	Delay	Source Clock	Destination Clock	Analysis Type
1	in_data[0]	isdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_rx_static_bypass/Data[0].u_IFD1P3DX.PIC_inst/D	-5000	5355	-10355	in_clock_vtr	clkos_o_c	hold
2	in_data[1]	isdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_rx_static_bypass/Data[1].u_IFD1P3DX.PIC_inst/D	-5000	5350	-10350	in_clock_vtr	clkos_o_c	hold
3	in_data[2]	isdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_rx_static_bypass/Data[2].u_IFD1P3DX.PIC_inst/D	-5000	5350	-10350	in_clock_vtr	clkos_o_c	hold
4	in_data[3]	isdr_inst/lscc_sdr_inst/genblk2.u_lscc_sdr_rx_static_bypass/Data[3].u_IFD1P3DX.PIC_inst/D	-5000	5350	-10350	in_clock_vtr	clkos_o_c	hold

Figure 3.26. Setup and Hold Timing Results for Case 3 SDR Source-Synchronous Input Interface

Table 3.2. lists the setup slack and hold slack comparison for all three cases of SDR source-synchronous input interface. The setup and hold slack margin is more balanced when the clock and data relationship is center-aligned whether it is done by external device or the FPGA's PLL. For these cases, the PLL is configured as feedback compensated mode. Therefore, the routing delay of the clock path is compensated. If without the PLL, the FPGA clock path routing delay is generally much larger compared to the FPGA data path delay from the I/O interface. As a result, the hold time is more difficult to close. Therefore, it is always better to center-aligned the clock and data somewhere in the system whether by an external device or through the FPGA PLL.

Table 3.2. Setup and Hold Time Comparison for SDR Source-Synchronous Input Interface

Case	Setup Slack (ns)	Hold Slack (ns)
Case 1 - Edge-aligned interface	8.570	0.355
Case 2 - Center-aligned interface with FPGA clock phase shift	3.570	5.355
Case 3 - Center-aligned interface with external device clock phase shift	3.570	5.355

References

For more information, refer to the following resources:

- Lattice Radiant Timing Constraints Methodology Application Note (FPGA-AN-02059)
- Lattice Insights web page for training courses and learning plans

Technical Support Assistance

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

For frequently asked questions, refer to the Lattice Answer Database at www.latticesemi.com/Support/AnswerDatabase.

Revision History

Revision 1.0, January 2024

Section	Change Summary
All	Initial release.

www.latticesemi.com