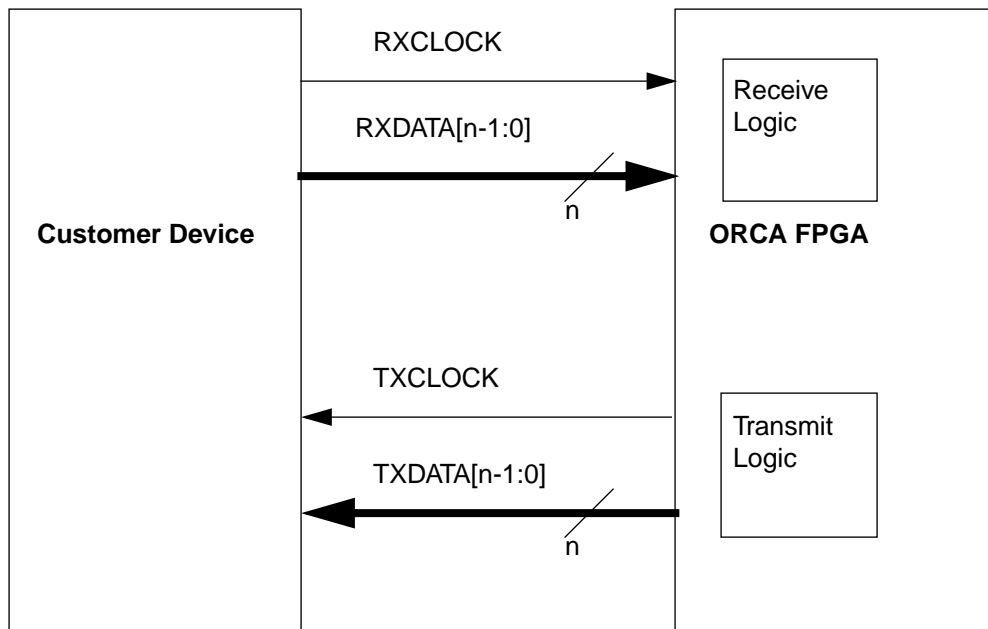


Introduction

This document will specify the capabilities of the ORCA[®] series 4 I/O logic. Specifically, Double Data Rate (DDR) interface schemes with clock forwarding. Single-ended I/O standards, such as XGMII, and speed capabilities up to 311MHz for single-ended I/Os and 350MHz for differential I/Os will be addressed. This document will discuss the ORCA Series 4 FPGA receiving interface and transmitting interface separately. A reference design is included showing the design practices required to meet XGMII specifications with DDR mode output registers using both clock edges. Similar to the XGMII reference design, precise design practices must be followed to achieve system level performance using 311MHz single-ended I/Os and 350MHz differential I/Os. The Lattice web site (www.latticesemi.com) should be consulted periodically to obtain the latest reference designs and information. All timing in this document should be used instead of the results from ispLEVER[™], due to the full range of simulations used to obtain the results described in this document.

A typical interface between an ORCA FPGA and another device, using DDR with clock forwarding, is shown in Figure 1. The Data bits are always transmitted and received with a transmit clock and a receive clock, respectively. Note that for the purposes of this technical note, DATA and CONTROL are both referred to as DATA.

Figure 1. ORCA Chip to Chip Receive Interface



Receive Interface

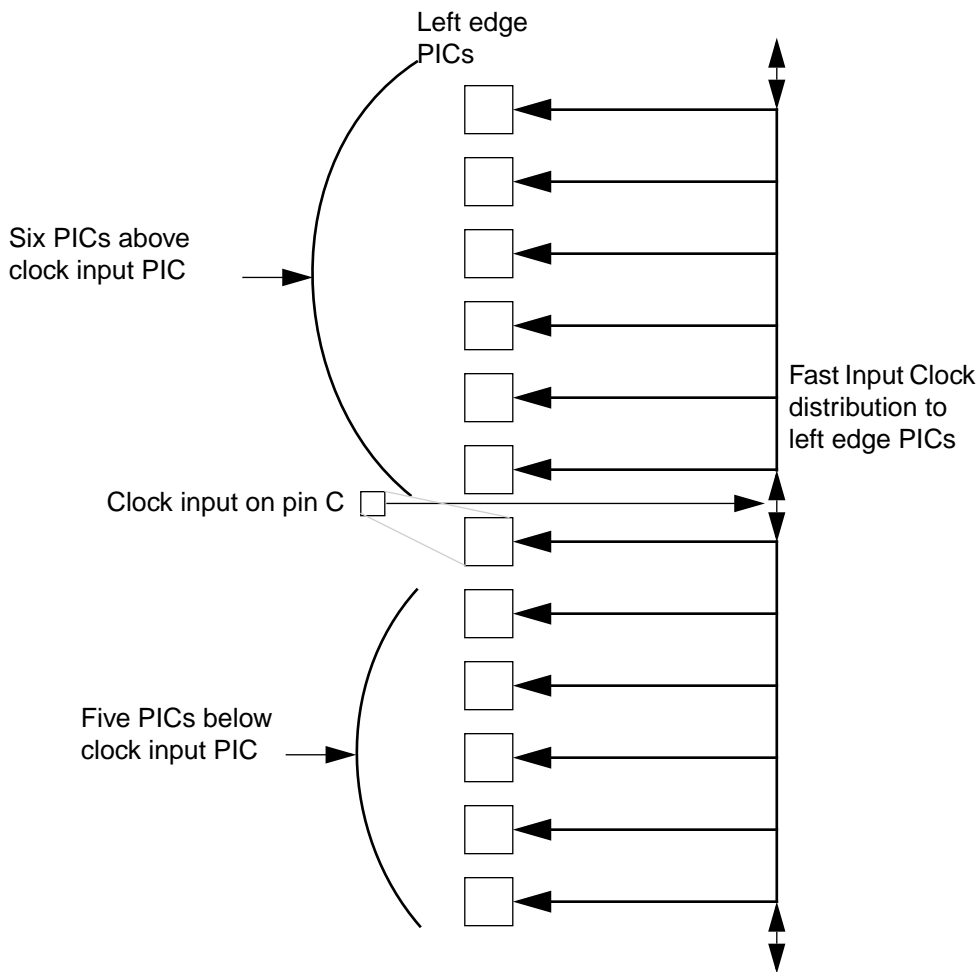
Each receive interface is limited to a 12 Programmable Interface Cell (PIC) region, due to the interface clock distribution scheme. Each receive interface can have a maximum of 47 pins as single-ended data inputs, or 22 LVDS data inputs and two pins for the clock input. The number of data input pins per clock input pin is limited due to the bonding constraints of different packages. All of the potential pins are not bonded in some package types. Some signal interface standards require reference voltage inputs. The number of single-ended data input pins is further

limited by the number of reference voltage input, Vref pins, pins required. Each PIC has up to four bonded out pads (A,B,C and D).

Only C pads can be used as a clock input for the Fast Input DDR interface, and the pad C clock input must be programmed in Fast Input mode. The reference design, mentioned in the appendix of this document, shows the attributes to be put on the hardware description language (HDL) buffer instantiations to guarantee that these, and other, hardware restrictions are adhered to when the FPGA/FPSC is configured. Data signals can be any pin (A, B, C and D). The top and bottom edges of the FPGA have 12 PIC interface regions that include the PIC with the clock input pin, five PICS to the right and six PICS to the left. The left and right edges of the FPGA have 12 PIC interface regions that include the PIC with the clock input pin and, five PICS below and six PICS above.

No PLL should be used with this Fast Input interface. The input data has an instantaneous relationship with the input clock, rather than being smoothed by the PLL. A diagram of the receive interface on the left edge of the device is shown in Figure 2.

Figure 2. ORCA FPGA Receive Interface



Each PIC is composed of four pins (A,B,C and D) and four DDR logic blocks. Each DDR logic block can be configured as an input DDR logic block or an output DDR logic block. When pins or PIOs are configured in Fast Input DDR mode, they use a special clock distribution on the edge of the device. The Fast Input clock distribution, Edge Clock, spans six PICS at a time as shown in Figure 2. Each PIO programmed in Fast Input DDR mode uses two flip-flops. One flip-flop is a negative edge triggered flip-flop in the shift register and the other flip-flop is a positive edge triggered flip-flop in the PIO logic. Diagrams of this are shown in Figure 3 and Figure 4.

Figure 3. Fast Input DDR Flip-Flop Distribution

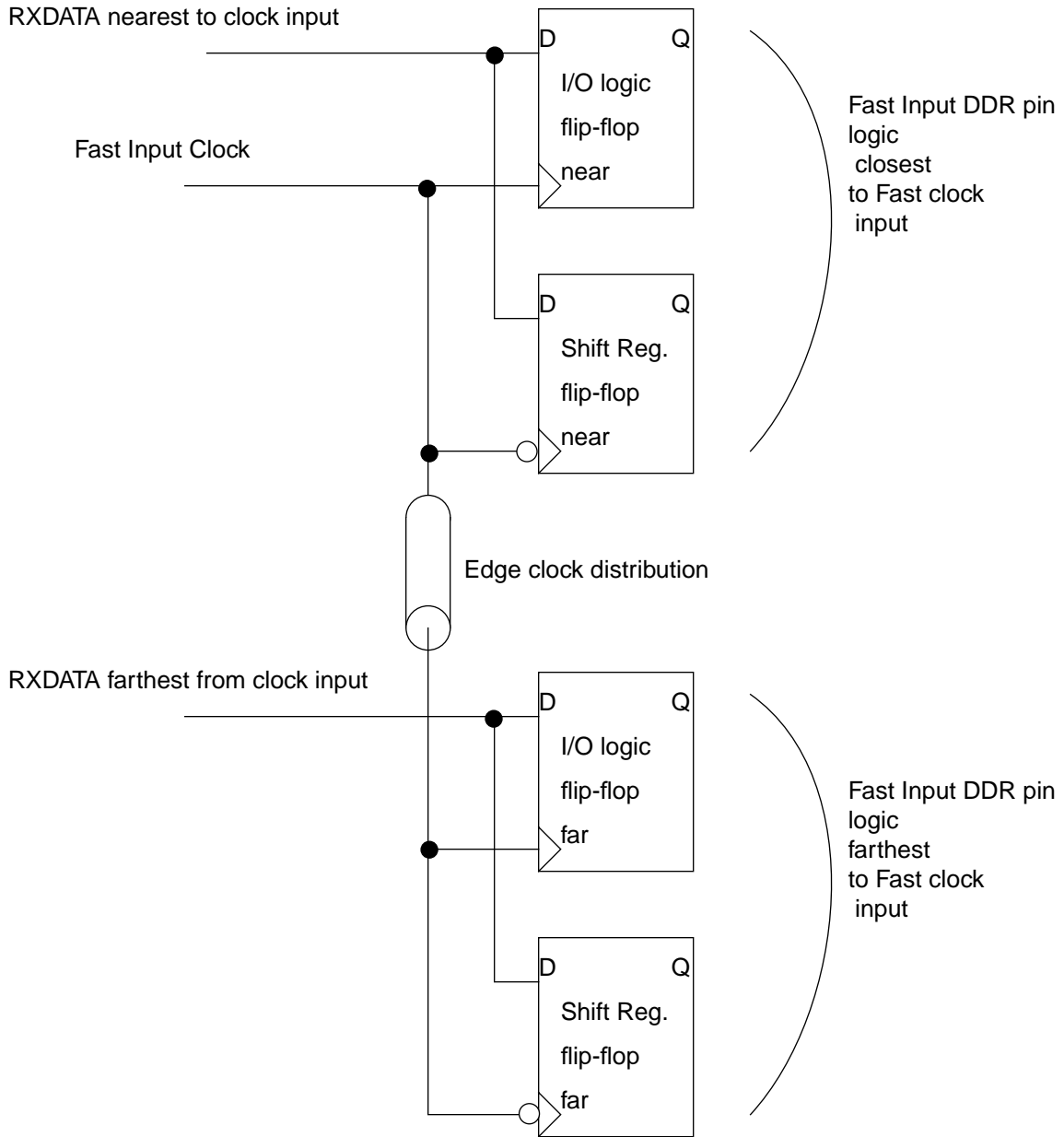
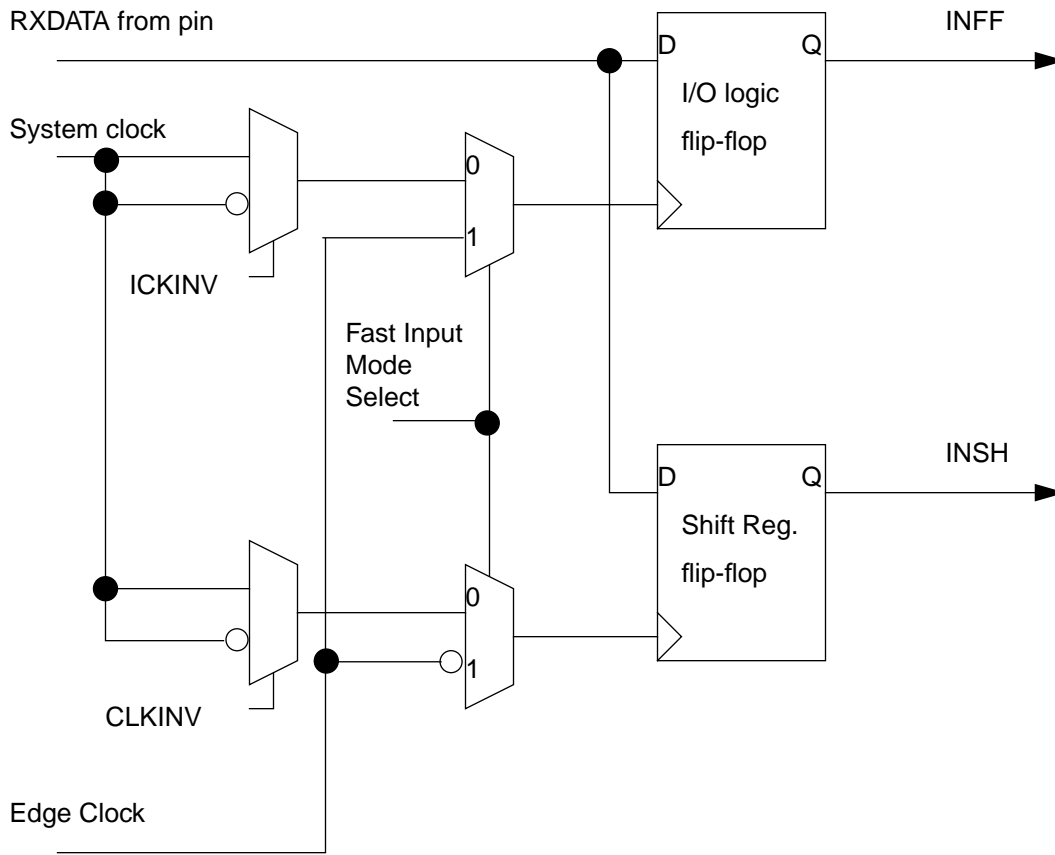
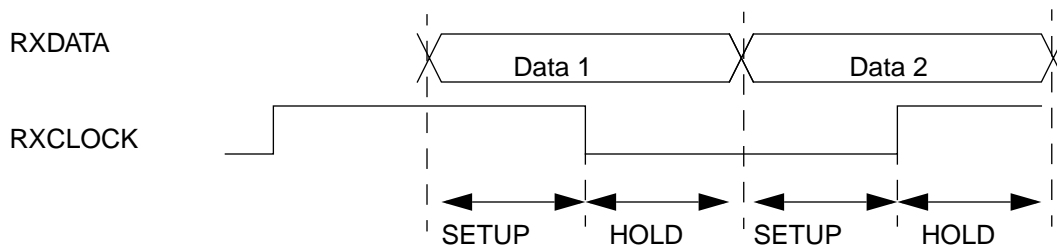


Figure 4. PIC Input DDR Logic Scheme



The clock and data bits have a required relationship at the input pins. The input relationship is specified as a setup time and hold time of the data with respect to the clock edge, either rising or falling as shown in Figure 5.

Figure 5. Fast Input DDR Timing Diagram



The Input interface setup and hold times are a function of the input buffer type and the signal characteristics.

1.5V HSTL Inputs

A common Fast Input DDR interface buffer type is HSTL class 1, as used in XGMII interface schemes. HSTL class 1 signal input characteristic specifications are listed in Table 1. The signal levels are a direct function of the buffer power supply, V_{DDIO} . V_{REF} is a function of the V_{DDIO} and the input signal levels are required to be at least 200mV around the reference voltage. The V_{REF} input signal pins will be required for this interface and will reduce the number of data input signal pins available within a 12 PIC region.

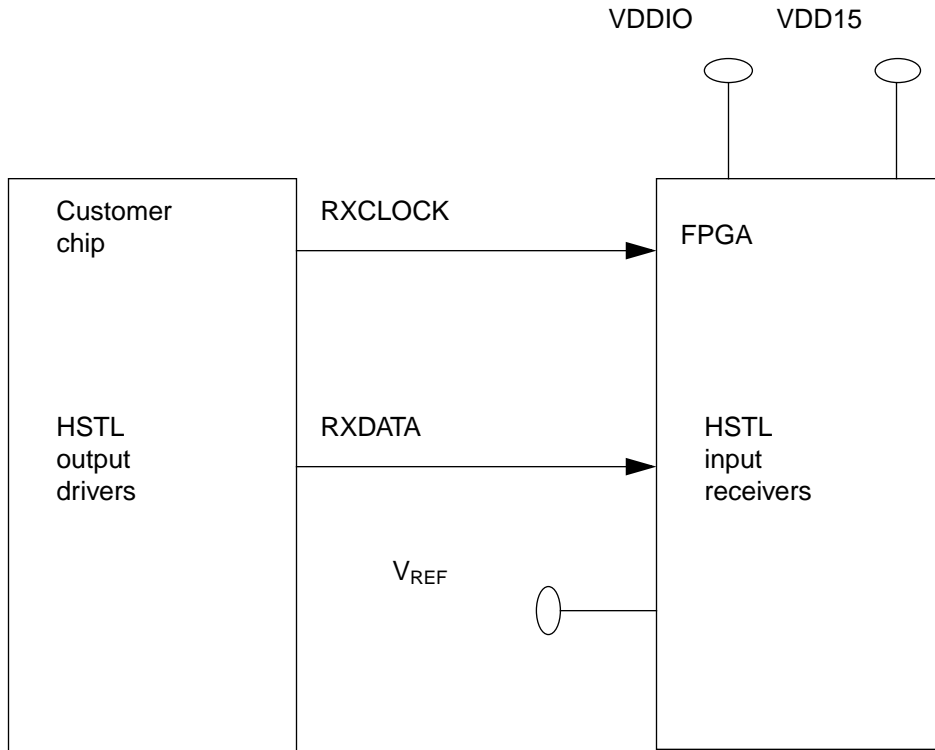
Please refer to both the XGMII and HSTL standard specifications IEEE 802.3ae and EIA/JESD8-6 respectively for more information.

Table 1. XGMII HSTL Receive Signal Specifications

Parameter	Low	Nominal	High
V_{DDIO}	1.40 V	1.50 V	1.60 V
V_{DD15}	1.425 V	1.50 V	1.575 V
V_{IH} (min.)	0.88 V	0.95 V	1.10 V
V_{REF}	0.68 V	0.75 V	0.90 V
V_{IL} (max.)	0.48 V	0.55 V	0.70 V
Input Slew Rate	1 V/ns	1 V/ns	1 V/ns

A Fast Input DDR interface scheme using an HSTL receiver and 1.5V signaling is shown in Figure 6.

Figure 6. XGMII Fast Input DDR HSTL Receive Interface Diagram



Simulation details and results using a Series 4 HSTL receiver at 156MHz and 1.5V power supply are listed in Table 2.

Table 2. XGMII Fast Input DDR HSTL Receive Interface Results

Setup (ps)	Hold (ps)	Speed Grade	Minimum Temperature (°C)	Maximum Temperature (°C)
580	550	-1	-40	125
480	480	-2	-40	125
480	480	-3	-40	125

The setup and hold simulation results presented throughout this document are the worst case delta between the clock and data measured at seven different process corners. The fact that different speed grade setup and hold numbers are the same is a result of the these measurements.

1.8V HSTL Inputs - Non-Standard

Simulation signal characteristics and results using a Series 4 ORCA HSTL receiver at 156MHz and a 1.8V power supply with higher amplitude signal levels are shown in Table 3 and Table 4.

Table 3. 1.8V Fast Input DDR HSTL Receive Interface Signal Specifications

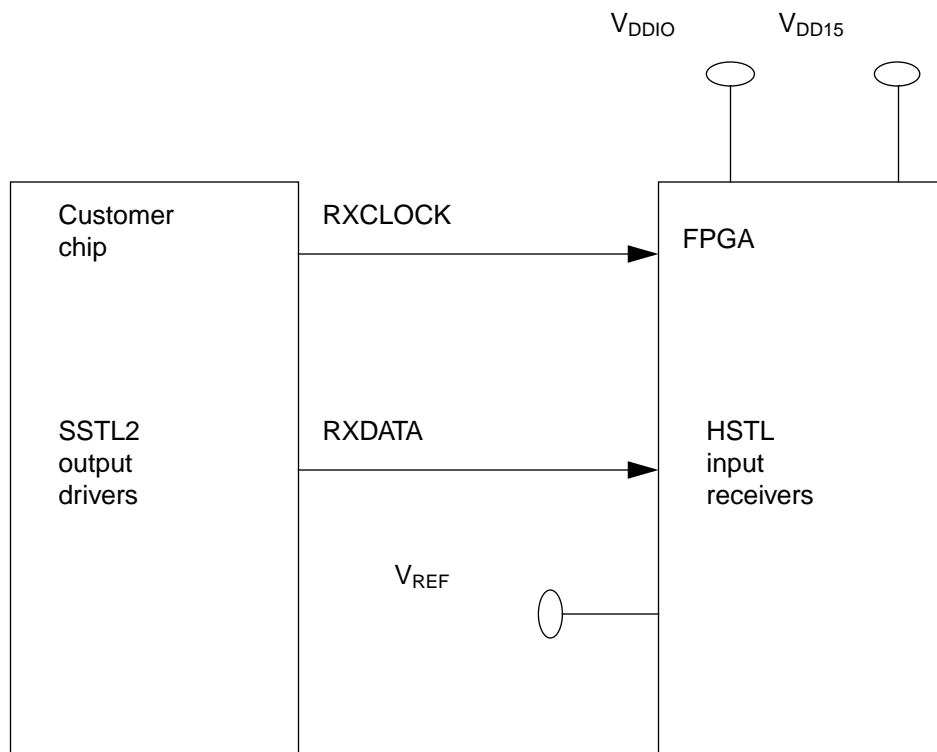
Parameter	Low	Nominal	High
V _{DDIO}	1.7 V	1.8 V	1.9 V
V _{DD15}	1.425 V	1.5 V	1.575 V
V _{IH} (minimum level)	1.4 V	1.5 V	1.8 V
V _{REF}	0.85 V	0.9 V	0.95 V
V _{IL} (maximum level)	0 V	0 V	0.10 V
Input Slew Rate	1 V/ns	1 V/ns	1 V/ns

Table 4. 1.8V Fast Input DDR HSTL Receive Interface Results

Setup (ps)	Hold (ps)	Speed Grade	Minimum Temperature (°C)	Maximum Temperature (°C)
580	550	-1	-40	125
580	480	-2	-40	125
580	480	-3	-40	125

Fast Input DDR interface scheme using an HSTL receiver being driven by an SSTL2 output driver (2.5V signaling) is shown in Figure 7.

Figure 7. 2.5V Fast Input DDR HSTL Receive Interface Diagram



2.5V HSTL Inputs with High Non-Centered Reference Voltage Scheme - Non-Standard

Simulation signal characteristics, simulation details and results using an HSTL receiver at 156MHz and a 1.5V power supply with higher amplitude signal levels are shown in Table 5 and Table 6.

Table 5. 2.5V Fast Input DDR HSTL Receive Interface Signal Specifications

Parameter	Low	Nominal	High
VDDIO	1.4 V	1.5 V	1.6 V
VDD15	1.425 V	1.50 V	1.575 V
Vih (minimum level)	1.45 V	1.55 V	1.65 V
Vref (not standard STTL2 level)	1.30 V	1.40 V	1.5 V
Vil (maximum level)	0.85 V	0.95 V	1.05 V
Input Slew Rate	1 V/ns	1 V/ns	1 V/ns

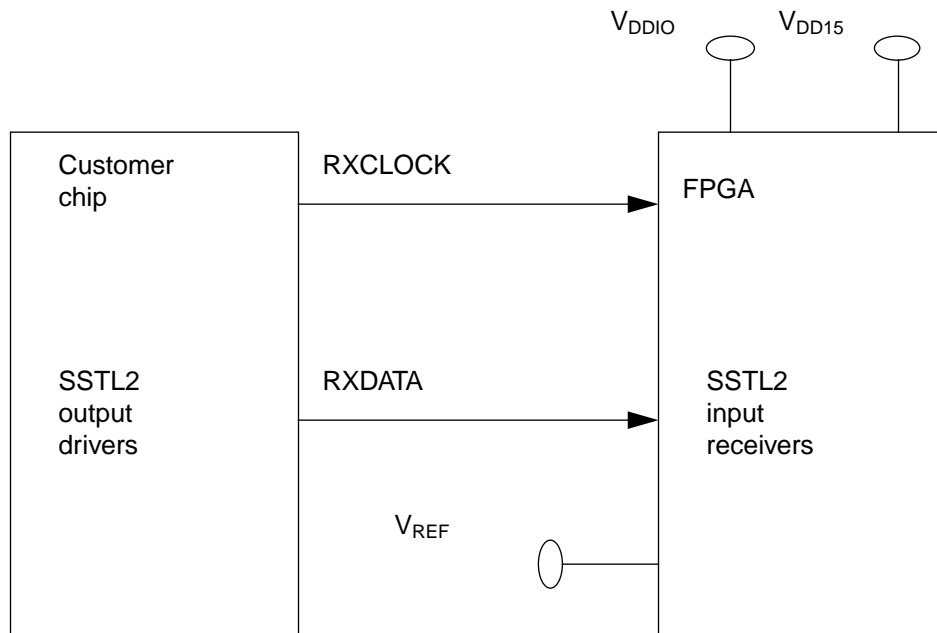
Table 6. 2.5 V Fast Input DDR HSTL Receive Interface Results

Setup (ps)	Hold (ps)	Speed Grade	Minimum Temperature(°C)	Maximum Temperature (°C)
820	625	-1	-40	125
820	625	-2	-40	125
820	625	-3	-40	125

2.5V SSTL Inputs

Fast Input DDR interface scheme using an SSTL2 receiver being driven by an SSTL2 output driver (2.5V signaling) is shown in Figure 8.

Figure 8. 2.5V Fast Input DDR SSTL2 Receive Interface Diagram



Signal specifications and simulation results using an SSTL2 receiver at 156 MHz and a 2.5V power supply are shown in Table 7 and Table 8.

Table 7. 2.5 V Fast Input DDR SSTL2 Receive Interface Signal Specifications

Parameter	Low	Nominal	High
VDDIO	2.30 V	2.50 V	2.70 V
VDD15	1.425 V	1.50 V	1.575 V
Vih (minimum level)	1.50 V	1.60 V	1.70 V
Vref	1.15 V	1.25 V	1.35 V
Vil (maximum level)	0.80 V	0.90 V	1.00 V
Input Slew Rate	1 V/ns	1 V/ns	1 V/ns

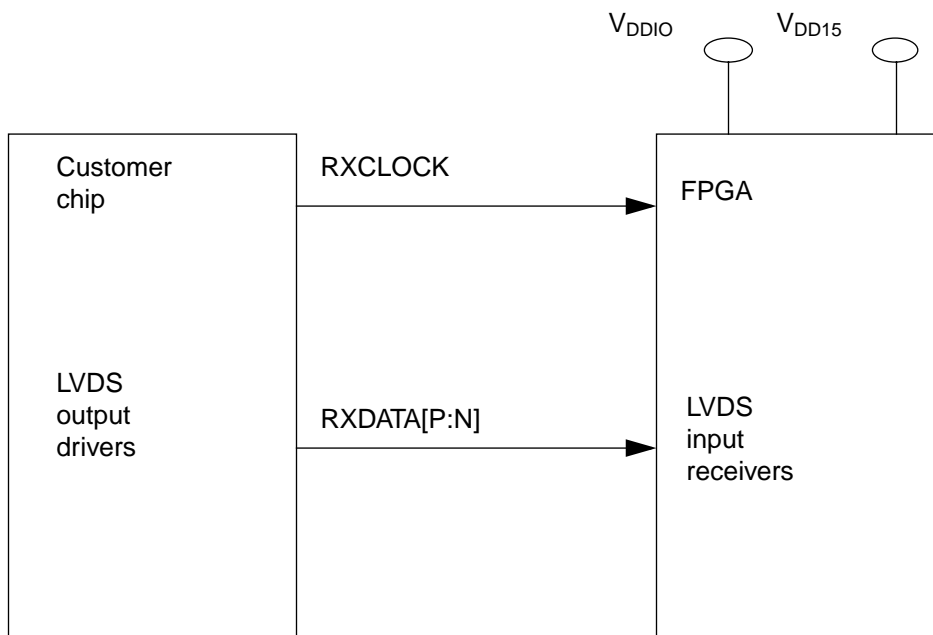
Table 8. 2.5 V Fast Input DDR SSTL2 Receive Interface Results

Setup (ps)	Hold (ps)	Speed Grade	Minimum Temperature(°C)	Maximum Temperature (°C)
700	700	-1	-40	125
500	500	-2	-40	125
500	500	-3	-40	125

2.5V LVDS Inputs

Fast Input DDR interface scheme using an LVDS receiver being driven by an LVDS output driver (2.5V signaling) is shown in Figure 9.

Figure 9. 2.5 V Fast Input DDR LVDS Receive Interface Diagram



Signal specifications and simulation results using an LVDS receiver and a 2.5V power supply are shown in Table 9 and Table 10.

Table 9. 2.5V Fast Input DDR LVDS Receive Interface Signal Specifications

Parameter	Low	Nominal	High
VDDIO	2.3 V	2.5 V	2.7 V
Differential Input Voltage	0 V	-	2.4 V
Differential Input Threshold	-100 mV	-	100 mV
Output Common-mode (Offset) Voltage	1.125 V	1.25 V	1.357 V
Input Common-mode Voltage	0.2 V	1.25 V	2.2 V
Integrated Receiver Termination Resistor	95 Ω	100 Ω	105 Ω

Table 10. 2.5V Fast Input DDR LVDS Receive Interface Results

Setup (ps)	Hold (ps)	Speed Grade	Minimum Temperature(°C)	Maximum Temperature (°C)
TBD	TBD	-1	-40	125
TBD	TBD	-2	-40	125
TBD	TBD	-3	-40	125

Additional Hold Considerations

Fast Input DDR interface schemes using I/O standards that require reference voltage inputs become difficult to fit on the Series 4 ORCA device as the bus width requirement increases and the package size decreases. There are two possible solutions for these scenarios. The first solution is to use more than one clock input per group of data bits. The second solution is to extend the Fast Input clock (Edge Clock) in either direction from the clock pin, by units of six PICs. The first solution can be achieved with manipulation of the board level clock and data traces, essentially adding more RC delay to the clock and then matching data traces to the clock. The second solution changes the data to clock setup and hold time specifications for all pins in the extended region of the data bits. For the pins used in the extended region, an additional amount of hold time must be added to the pin timing specifications. The amount to be added is listed in Table 11.

Table 11. Additional Hold Time Specifications

Additional Sections of Clock Network	Setup Time	Hold Time (-1)	Hold Time (-2)	Hold Time (-3)
1 Group of 6 PICs in each direction	0 ps	+345 ps	+380 ps	+340 ps
2 Groups of 6 PICs (12) in each direction	0 ps	+690 ps	+760 ps	+680 ps

The (-1), (-2), and (-3) in the above table are the speed grades for the Series 4 ORCA devices.

Transmit Interface

Output DDR interfaces transmit data (and control) signals on both the positive and negative edges of a clock. Clock forwarding transmits a clock signal along with the data and control signals with a specified relationship between the data (and control) bits and the clock signal.

All forms of signal skew directly reduce the margin of the specified relationship. Each transmit interface should use Series 4 ORCA Primary Clock routing resources to distribute clocks with as little skew as possible. The skew of the Primary Clock distribution is a function of the size of the array, process, voltage and temperature corners and the relative relationship of the rise and fall edges of the clock as seen at the end points of the distribution.

Unlike the Fast Input DDR interface, there are no constraints limiting the relative distance between Output DDR pins. The architecture of the Primary Clock resources are such that horizontal circuits have less skew than vertical circuits. This is due to the fact that the primary distribution has a very low skew spine network and branches contribute most of the skew. The branches run vertically, so any circuits that run horizontally across the FPGA, tap off

the branches at the same point. PICs on the top and bottom edges of the FPGA have less skew between them than the PICs on the left or right edges of the FPGA.

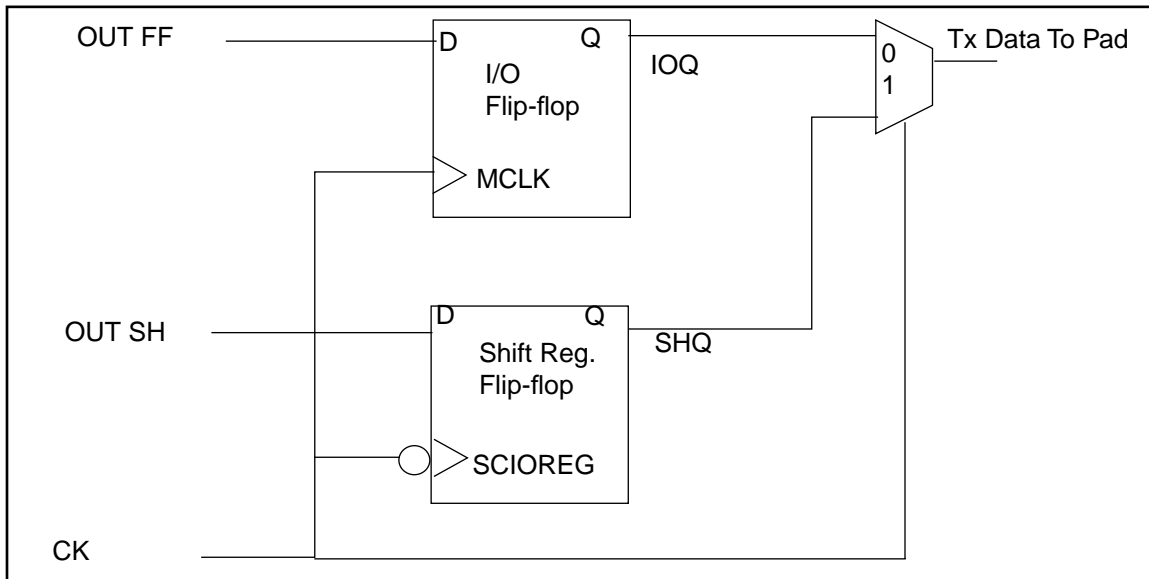
To implement an output DDR circuit, program the output pins as output DDR elements. Each and all of the four pins (A,B,C and D) in a PIC can be in Output DDR mode, and they must share the same clock. This implementation will be discussed in the following sections and a reference design described in the appendix is available. The following sections will also discuss the idea of clock forwarding which is an essential aspect of high speed Output DDR interface implementation.

Output DDR Interface

Implementing output DDR circuits in the PICs requires using both the PIO logic and Shift Register flip-flops together. A diagram showing the Output DDR logic is shown in Figure 10.

For clock forwarding, make sure that the clock is using the primary routing distribution for lowest skew. The Output DDR logic can be driven from either the System Clock (SC) or the Edge Clock(EC).

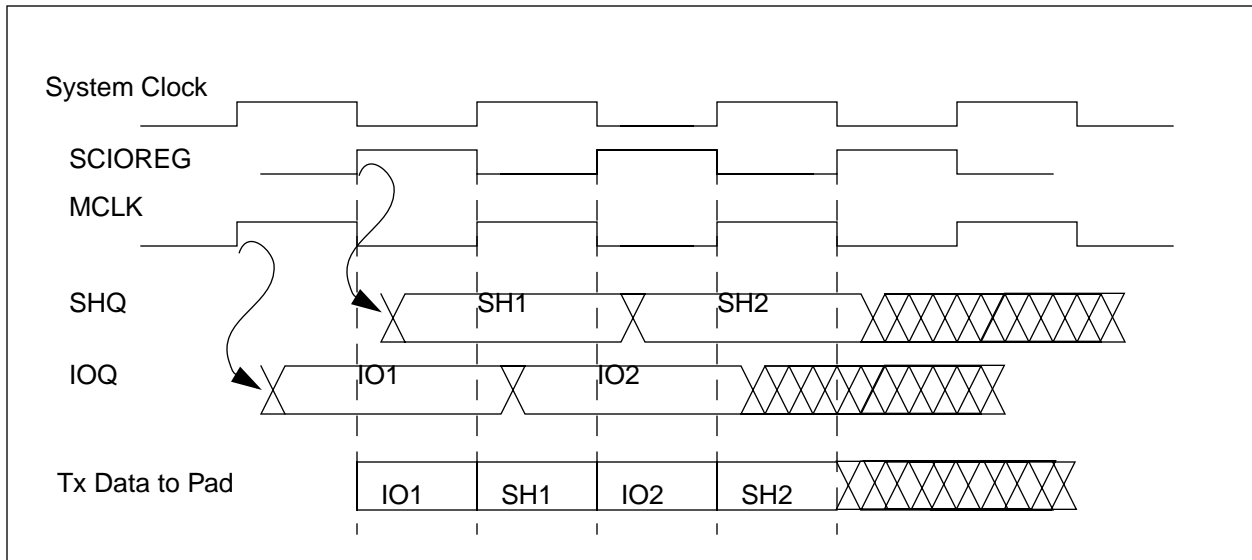
Figure 10. Different Edge Flip-Flops for Output DDR Interface



Note: Each flip-flop changes value on the opposite edge from which it's output is selected by the MUX, reducing skew.

A diagram of the timing for output DDR with different edge flip-flops is shown in Figure 11.

Figure 11. Output DDR Mode Using Flip-Flops Triggered on Different Edge

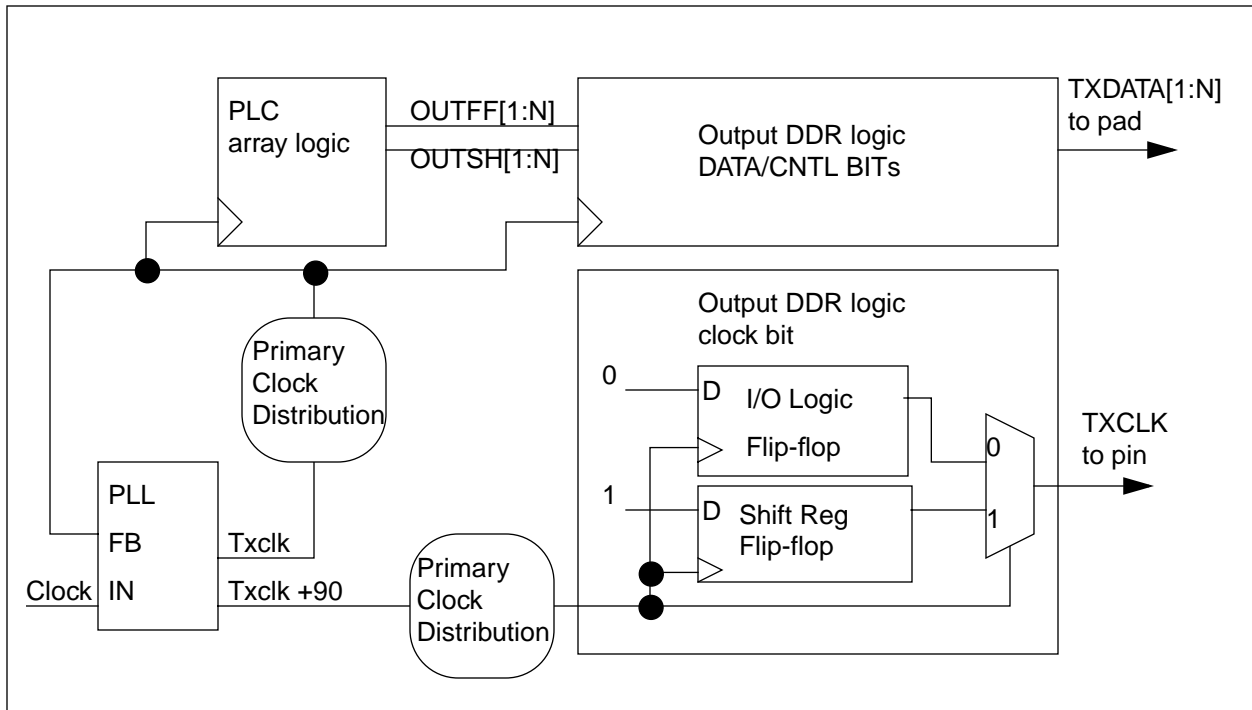


One consideration of using flip-flops with different edge triggers in the output DDR mode PICs is the requirement to have a half cycle transfer somewhere in the logic path prior to the output PICs. A half cycle transfer logic path is a path of logic where the data is transmitted from one flip-flop on one edge of the clock cycle and captured in a flip-flop on the opposite edge of the clock cycle. A half cycle transfer essentially doubles the speed requirements of the logic.

It is further suggested that no half cycle transfers are performed across the PLC array to PIC boundary because the PLC array to PIC boundary has a tighter timing constraint than PLC to PLC timing constraints. The PLC logic and PIC logic use the same clock distribution for Output DDR registers and they are running at the same frequency and clock phase. In Output DDR mode operation the output pins transmit data on both relative edges of the clock. Both the rising and falling input edges of the clock distribution are used and data is transmitted out on both edges of the clock, inverted and non-inverted.

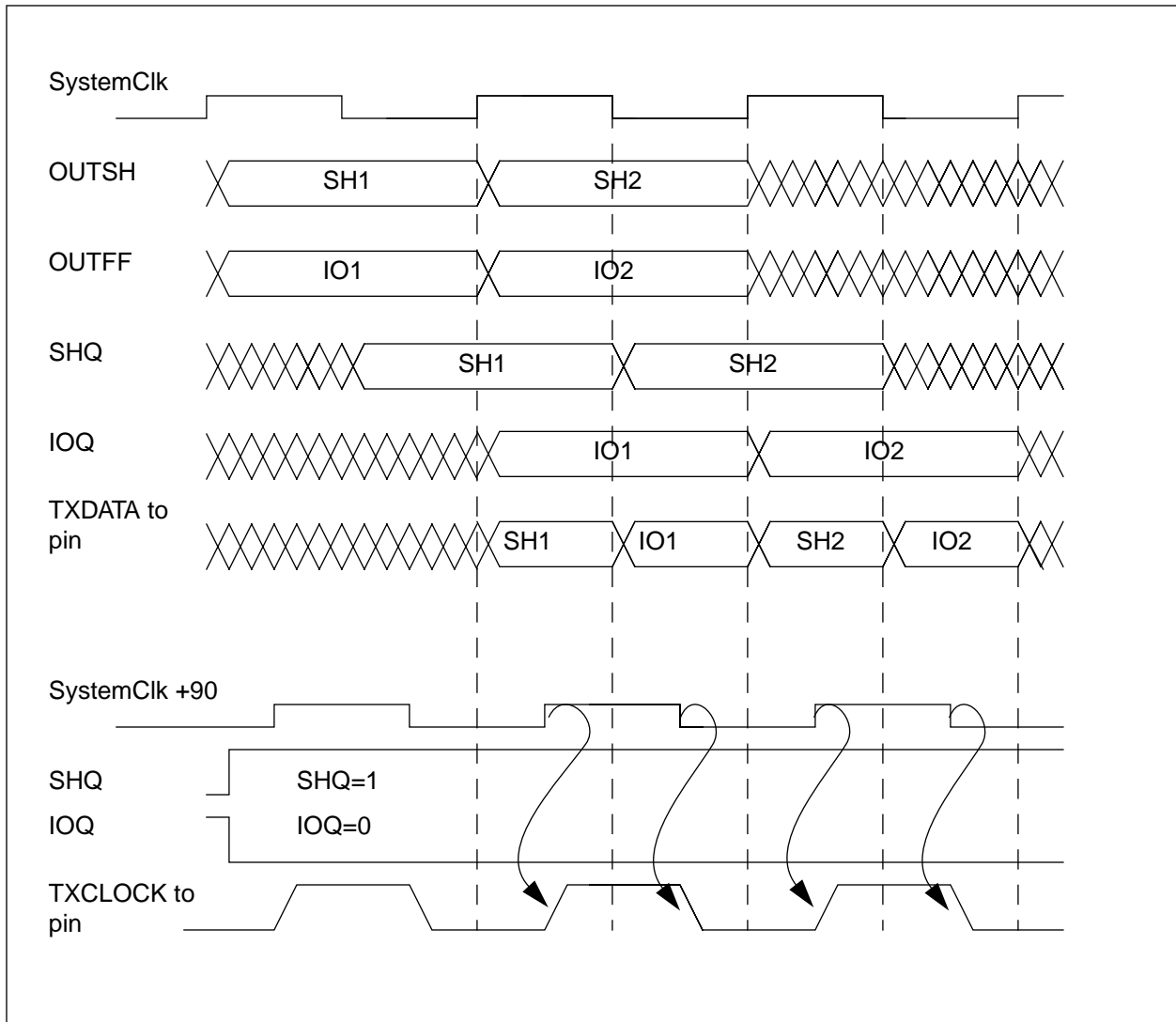
Clock forwarding in Output DDR mode is accomplished by transmitting the clock out in the same mode as the data, with the shift register flip-flop input and the I/O logic flip-flop input signals tied to opposite levels (high and low). A +90 Degree phase shifted clock is used from a PLL to drive the output clock signal, so that it is in the center of the data transitions. A diagram of the Output DDR mode with clock forwarding scheme is shown in Figure 12.

Figure 12. Output DDR with Clock Forwarding



A timing diagram of the Output DDR circuit using flip-flops triggered on different edges and clock forwarding is shown in Figure 13.

Figure 13. Output DDR with Clock Forwarding Timing Diagram



Clock Skew Considerations for the Transmit Interface

Output DDR in Series 4 ORCA devices should use the Primary Clock distributions. The skew of the Primary Clock distribution for different edge clocks for different process and temperature conditions are listed in Figure 12 and for different speed grades in Figure 13.

Table 12. Primary Clock Skews

Condition	OR4E02	OR4E04	OR4E06
Slow @ 125 °C	262 ps	283 ps	297 ps
Nominal @ 125 °C	188 ps	207 ps	219 ps
Nominal @ 85 °C	168 ps	184 ps	195 ps
Nominal @ 25 °C	78 ps	92 ps	101 ps
Fast @ -40 °C	65 ps	73 ps	79 ps

Table 13. Primary Clock Skew by Speed Grade

Speed Grade	OR4E02	OR4E04	OR4E06
-1	262 ps	283 ps	297 ps
-2	258 ps	279 ps	293 ps
-3	217 ps	233 ps	243 ps

DDR Outputs, including forwarded clocks, from Series 4 ORCA devices have skew due to the non-symmetry of the “Clock to pin” rising output delay versus falling output delay. This delay and skew is a function of the output buffer type, the capacitive load at the pin and the value of the transmit data. Output DDR modes are listed in Table 14 for HSTL and SSTL2 buffer types at a loading of 30pf. HSTL18 is a non-standard transmit buffer with a 1.8V power supply.

Table 14. Output Circuit Skew by Speed Grade

Speed Grade	DDR Mode with 30 pF Load			
	HSTL	HSTL18	SSTL2	LVDS
-1	542 ps	396 ps	186 ps	TBD
-2	463 ps	344 ps	186 ps	TBD
-3	172 ps	184 ps	186 ps	TBD

Capacitive Loading Effects on Transmit Interface

Output slew changes due to capacitive loading will affect the skew of the “Clock to pin” delay. A listing of the slew rate versus capacitance is listed below for HSTL and SSTL2 in Table 15 and Table 16.

Table 15. SSTL2 Class 1 Output Slew Rates

Active Load	Speed Grade -1 Rise Slew	Speed Grade -1 Fall Slew	Speed Grade -2 Rise Slew	Speed Grade -2 Fall Slew	Speed Grade -3 Rise Slew	Speed Grade -3 Fall Slew	Worst Case Fast Rise Slew	Worst Case Fast Fall Slew
5pf	0.300044	0.379045	0.226525	0.263872	TBD	TBD	0.116569	0.113527
10pf	0.379675	0.461125	0.30894	0.340709	TBD	TBD	0.181648	0.177309
15pf	0.472546	0.554026	0.400628	0.425461	TBD	TBD	0.258786	0.249184
25pf	0.693088	0.728555	0.61478	0.599365	TBD	TBD	0.415799	0.399804
30pf	0.809378	0.815816	0.72652	0.692307	TBD	TBD	0.497285	0.474028
35pf	0.926506	0.903643	0.836652	0.780091	TBD	TBD	0.580173	0.551003
50pf	1.29856	1.18164	1.18738	1.06974	TBD	TBD	0.827322	0.781925
75pf	1.92047	1.64757	1.77291	1.54603	TBD	TBD	1.23570	1.16498
100pf	2.55386	2.13613	2.35006	2.03201	TBD	TBD	1.65066	1.55485

Table 16. HSTL Class 1 Output Slew Rates

Active Load	Speed Grade -1 Rise Slew	Speed Grade -1 Fall Slew	Speed Grade -2 Rise Slew	Speed Grade -2 Fall Slew	Speed Grade -3 Rise Slew	Speed Grade -3 Fall Slew	Worst Case Fast Rise Slew	Worst Case Fast Fall Slew
5pf	0.700712	2.62658	0.465135	1.32322	TBD	TBD	0.202517	0.304563
10pf	0.931923	2.61930	0.627783	1.31717	TBD	TBD	0.266703	0.323708
15pf	1.16967	2.62711	0.79414	1.32718	TBD	TBD	0.334558	0.348327
25pf	1.67313	2.65482	1.14563	1.36522	TBD	TBD	0.479858	0.390572
30pf	1.93303	2.66708	1.32808	1.39210	TBD	TBD	0.553889	0.410266
35pf	2.20361	2.68380	1.51216	1.42150	TBD	TBD	0.630547	0.431769
50pf	3.03033	2.75449	2.07544	1.51824	TBD	TBD	0.86584	0.490902
75pf	4.42320	2.91914	3.04104	1.70250	TBD	TBD	1.24917	0.582507
100pf	5.87570	3.10590	3.99798	1.87271	TBD	TBD	1.66138	0.671156

Series 4 DDR Outputs with Clock Forwarding Results

Output DDR modes with clock forwarding use PLLs to align the output data bits with the output clock signal. PLL's have jitter which add skew to the output signals. This added skew reduces the margin of the output signals. The three types of jitter that affect the Output DDR modes are duty cycle jitter, period jitter and output jitter. A definition and specification of each of these is listed in Table 17. More information on PLL operation and PLL jitter can be found in Lattice technical note TN1011, *ORCA Series 4 I/O Tuning via PLL*.

Table 17. PLL Jitter Definitions and Specifications

Jitter Type	Definition	Maximum
Duty Cycle	The min/max measurement of the time between successive rising to falling edges, or falling to rising edges. This value does not include the input jitter, a percentage of which must be added to this value to obtain the total output jitter. Values are 6 sigma results based on the UI at the PFD inputs.	0.025 Ulp-p
Period	The min/max measurement of the time when all rising edges occur versus the ideal edge locations. This value does not include the input jitter a percentage of which must be added to this value to obtain the total output jitter. Values are 6 sigma results based on the UI at the PFD inputs.	0.015Ulp-p
M Output vs. N Output	The worst case relationship between these two outputs from the PLL when they are programmed to have the same phase.	20ps

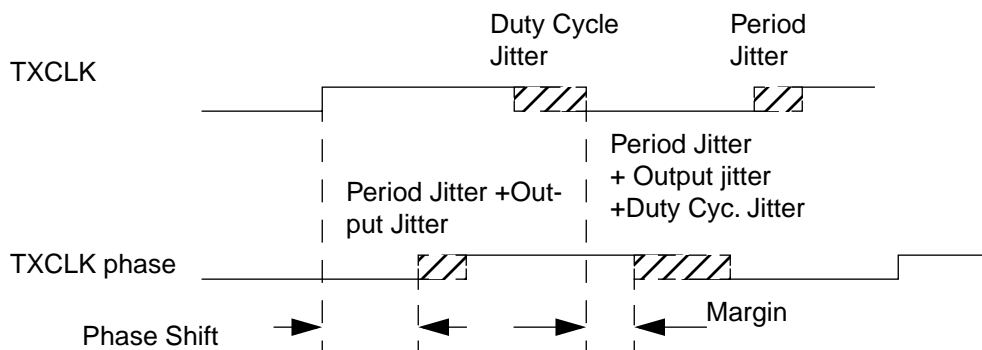
Typically, two outputs from a single Series 4 PLL are used to create two clock signals which can be used to drive any of the output DDR circuits discussed previously. The first clock signal is used to drive all data output pins. The second clock signal also drives an identical output DDR circuit, but it is shifted in phase by 90 degrees in order to create a centered, forwarded clock.

In order to create a clock signal, the DDR output mode logic selects between a logic "0" and logic "1".

Output DDR Timing in Series 4

The timing relationship, due to PLL jitter, between the forwarded clock and any data bit is shown in Figure 14.

Figure 14. Timing Diagram (with Phase Lock Loop Jitter)



From Figure 14 it is shown how the Output DDR mode data margin can be calculated by use of the following formula:

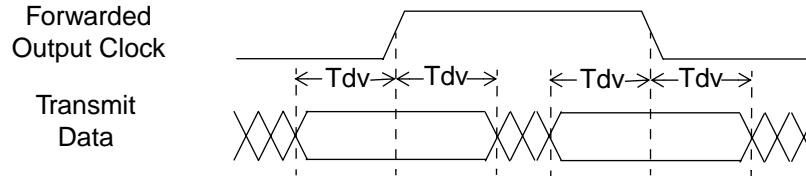
$$\text{MARGIN} = \text{PERIOD}/4 - \text{PLL jitter} - \text{Clock Skew} - \text{“Clock to pin” skew} \tag{1}$$

The following example shows how the margin for a -3 speed grade OR4E04 array.

- Margin = 1600ps Maximum ideal setup or hold time (bit period = 3200ps)
- 230ps Jitter of PLL
- 206ps Clock skew @ 125 degrees Celsius
- 235ps DDR out circuitry R/F skew HSTL @ 1.8 V
- = 929 ps Total margin

A timing diagram which references the data valid timing parameters is shown in Figure 15.

Figure 15. Data Valid Timing Diagram



The specifications for XGMII data valid time, relative to the forwarded output clock, is listed in Table 18 through Table 20 for different sized Series 4 arrays. Since the clock is centered on the data, the minimum guaranteed valid time before and after the forwarded clock are the same.

Table 18. 156MHz XGMII Output DDR Timing Specification for Array OR4E02

Symbol	Speed Grade	Data Valid Window			
		1.8 V HSTL	1.5 V HSTL	2.5 V SSTL2	2.5 V LVDS
Tdv	XGMII	0.9600 ns	0.9600 ns	0.9600 ns	N/A
	-1	0.6882 ns	0.5422 ns	0.8982 ns	TBD
	-2	0.7442 ns	0.6252 ns	0.9022 ns	TBD
	-3	0.9452 ns	0.9572 ns	0.9432 ns	TBD

Table 19. 156MHz XGMII Output DDR Timing Specification for Array OR4E04

Symbol	Speed Grade	Data Valid Window			
		1.8 V HSTL	1.5 V HSTL	2.5 V SSTL2	2.5 V LVDS
Tdv	XGMII	0.9600 ns	0.9600 ns	0.9600 ns	N/A
	-1	0.6672 ns	0.5212 ns	0.8772 ns	TBD
	-2	0.7232 ns	0.6042 ns	0.8812 ns	TBD
	-3	0.9292 ns	0.9412 ns	0.9272ns	TBD

Table 20. 156MHz XGMII Output DDR Timing Specification for Array OR4E06

Symbol	Speed Grade	Data Valid Window			
		1.8 V HSTL	1.5 V HSTL	2.5 V SSTL2	2.5 V LVDS
Tdv	XGMII	0.9600 ns	0.9600 ns	0.9600 ns	N/A
	-1	0.6532 ns	0.5072 ns	0.8632 ns	TBD
	-2	0.7092 ns	0.5092 ns	0.8672 ns	TBD
	-3	0.9192 ns	0.9312 ns	0.9172ns	TBD

Conclusion

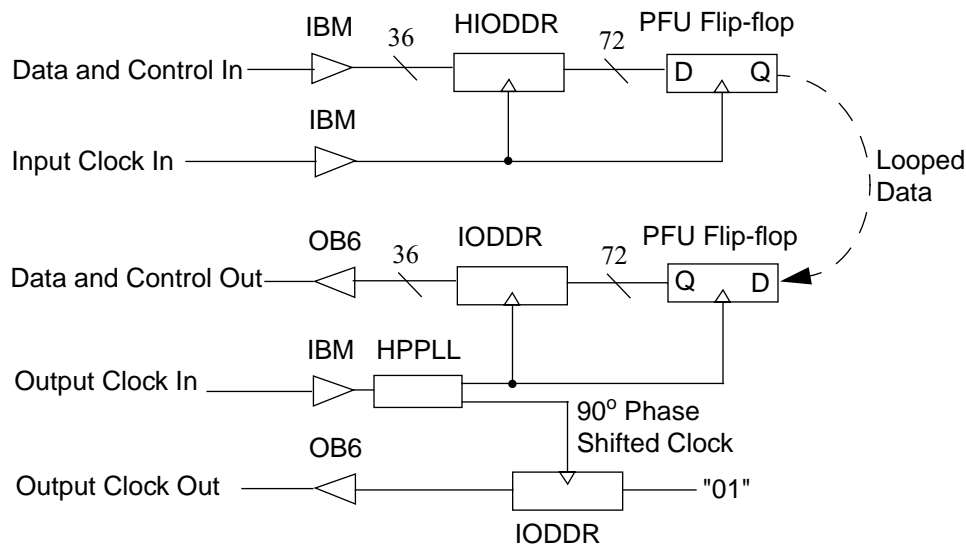
This document shows the specification on how to high-speed, clock-forwarded, double-data rate interfaces are accomplished with ORCA Series 4 FPGA and FPSC devices. Their potential use in applications such as the standard XGMII interface has been shown for the various options, with the results showing operation is possible at very high speed rates.

Appendix A. XGMII Example Code

A structural design example available for download from the Lattice website shows how XGMII specifications are met with DDR Mode input and output registers using both edges of the clock. Clock forwarding is also shown in this example design. The design is in Verilog HDL with attributes set for synthesis with the Synplicity Synplify synthesis tool. A preference file for ispLever is also included with the design example. This preference file, also known as a constraints file, has the necessary `use primary` clock directives for the input and output clocks.

At the top of the design there are module instantiations for ORCA library elements with attributes that keep the synthesis software from removing these black box modules. These elements include the input buffer (IBM), output buffer (OB6), the high-speed, programmable, phase-locked loop (HPPLL), and the DDR modules (HIODDR and IODDR). A block diagram showing the dataflow and clocking structure is shown in Figure 16.

Figure 16. XGMII Example Design Block Diagram



In this interface design example, the IODDR and HIODDR instantiated elements envelope the shift register and the I/O logic registers discussed in this document. The locate attributes are needed so that the I/O buffer and shift register elements fit the ORL110G-680 hardware package.

This example design loops the input data back to the output data. Insertion of the data processing (i.e. an encoder/decoder) code can be connected to this design example in place of the data loop shown in Figure 15.

The HIODDR and IODDR library elements must be port mapped to match the pin bond-out for a given package. Therefore, care must be taken when porting this design example to any other ORCA device, such as the ORT82G5, which is not pin for pin compatible with the ORL110G-680 device.

For example, the first HIODDR in the design example is instantiated as follows:

```
HIODDR DDR_B17_inst (
    .IND0(bxgmii_data_in[0]), .IND2(bxgmii_data_in[1]),
    .IND3(bxgmii_data_in[2]), .CK(bxgmii_inclk),
    .IN1Q1(xgmii_indata_64b[32]), .IN1Q0(xgmii_indata_64b[0]),
    .IN3Q1(xgmii_indata_64b[33]), .IN3Q0(xgmii_indata_64b[1]),
    .IN4Q1(xgmii_indata_64b[34]), .IN4Q0(xgmii_indata_64b[2])
    /* synthesis LOC="SHIFT_B17" */;
```

With this HIODDR element, the first three external DDR bits of data, `bxgmii_data_in[0:2]`, are converted into 6 bits of single data rate (SDR) bits: negative-edged `xgmii_indata_64b[32:34]` and positive-edged `xgmii_indata_64b[0:3]`. With all HIODDR and IODDR elements, `IND[0:3]` maps to `IN[1:4]Q0` and `IN[1:4]Q1`.

As can be seen from the Verilog attribute above, this HIODDR element is located to the I/O shift-register `SHIFT_B17`, which is the 17th I/O shift register on the bottom side of the device (counted left to right). There is one PIC, with four PIOs, for each I/O shift register. For `SHIFT_B17`, the corresponding PIC that it can connect to is named `PB17` in pinout tables. The four individual I/Os for this PIC are `PB17[A:D]`, but not all of these four I/Os are bonded out.

The `IND0`, `IND2`, and `IND3` pins are used in this port-mapping as they correspond to the bonded out A, C, and D pads of the PIC `PB17`. `IND1` is not used because pad B is not bonded out for the `ORLI10G` in the 680 pin package. The pinout for all PICs are given in the device data sheet pinout table, where the pins are listed in PIC name order, under the Pin Description column.

Appendix B. LVDS 8:1 MUX/DEMUX Example Code

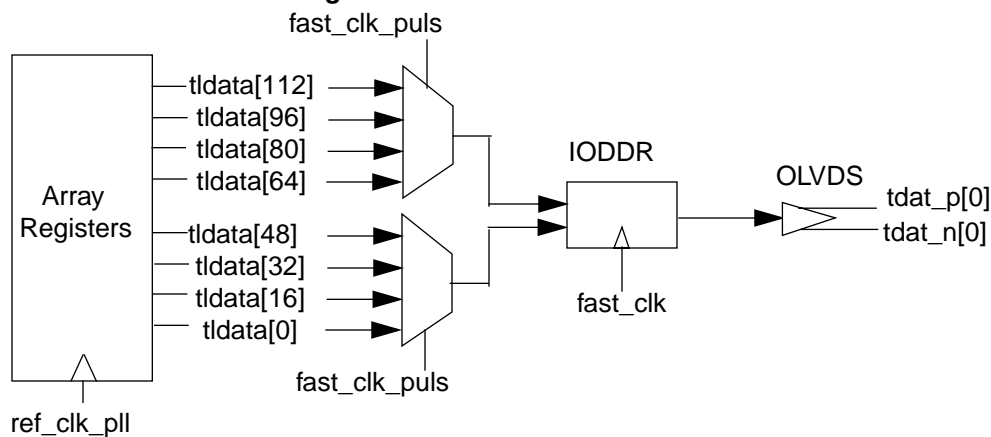
A structural design example available for download from the Lattice website shows how 350 MHz LVDS DDR interfaces can be used in ORCA Series 4 devices. The 350 MHz data rates are achieved with the -3 speed grade ORCA devices. This example design is 17-bits wide, 16 bits of data and one bit of control.

There are two designs in Verilog HDL with attributes set for synthesis with the Synplicity Synplify synthesis tool. One design, `lvds_dds_in.v`, is for a receive interface and the another design, `lvds_dds_out.v`, is for a transmit interface. A preference file for ispLever is also included for each example design. This preference file, also known as a constraints file, has the necessary `use primary` clock directives for the input and output clocks along with the frequency constraints for easily making 311 MHz with -2 speed grade ORCA devices. An `orca4_synplify.v` source file is also included which may need to be included when running a stand-alone synthesis run with Synplify.

In order to perform data processing at lower speeds inside the array logic, the data needs to be parallelized and synchronized to a lower speed clock. This is done in two stages, a 2:1 multiplexer/demultiplexer with the DDR elements coupled with a dual 4:1 multiplexer/demultiplexer with PFU based registers.

A block diagram showing the transmit dataflow and clocking structure for 1 of the 17 multiplexers is shown in Figure 17.

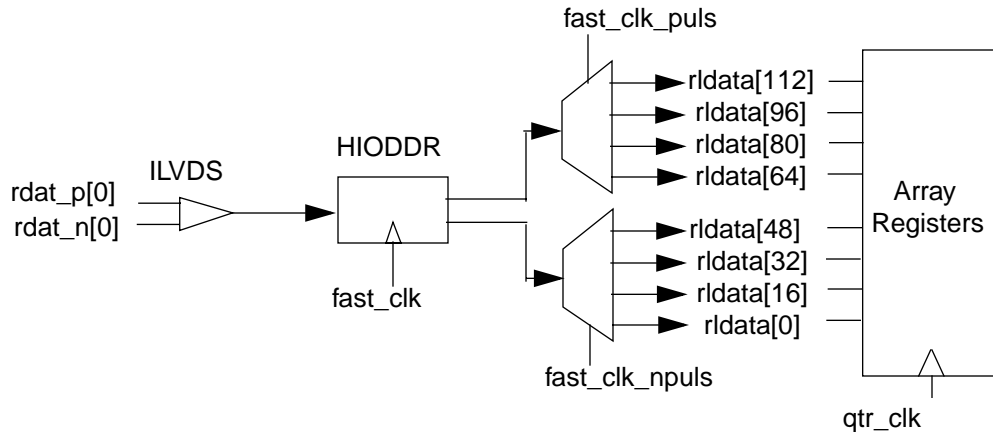
Figure 17. LVDS Transmit Data Flow Diagram



The 4:1 multiplexers in the above figure are implemented in the design example as PFU based 4-bit shift registers (FD1S3AX elements) and 2-input multiplexer registers (FL1S3AY elements). These elements are instantiated in the code so that they can be easily located in the array via the LOC attribute. The manual placement of these components is key to meeting the high-speed performance of the interface. The `fast_clk_puls` select signal is created from a 2-bit sample and decode of the low-speed clock, to manage the transfer from the low-speed reference clock domain into the high-speed clock domain. For the transmit interface, the reference clock is phase-locked with a high-speed programmable PLL element (HPPLL) which also multiplies the reference clock by 4 to generate the fast clock. This fast clock is forwarded off-chip with the high-speed data through an IODDR element to match the data skew off-chip.

A block diagram showing the receive dataflow and clocking structure for 1 of the 17 demultiplexers is shown in Figure 18.

Figure 18. LVDS Receive Data Flow Diagram



The 4:1 demultiplexers in the above figure are implemented in the design example as PFU based 4-bit shift registers (FD1S3AX elements) and 2-input multiplexer registers (FL1S3AY elements). These elements are instantiated in the code so that they can be easily located in the array via the LOC attribute. The manual placement of these components is key to meeting the high-speed performance of the interface. The `fast_clk_puls` and `fast_clk_npuls` select signals are created from a 2-bit sample and decode of the low-speed clock to manage the transfer from the fast clock domain into the quarter clock domain. Two different pulses are used: one for the rising edge that interfaces to the DDR element and 1 for the falling edge. For the receive interface the fast clock is an input and is divided by four by counters with inverted feedback to generate the quarter-speed clock (`qtr_clk`).

In this interface design example, the `IODDR` and `HIODDR` instantiated elements envelope the shift register and the I/O logic registers discussed in this document. The locate attributes are needed so that the I/O buffer and shift register elements fit the OR4E04-680 hardware package.

This example design registers the multiplexed/demultiplexed data in the array registers.. Insertion of an application-dependent design can be connected to these array registers.

The `HIODDR` and `IODDR` library elements must be port mapped to match the pin bond-out for a given package. Therefore, care must be taken when porting this design example to any other ORCA device, such as the ORT82G5, which is not pin for pin compatible with the OR4E04-680 device. For examples of the `IODDR` and `HIODDR` instantiations, please refer to Appendix A.

Technical Support Assistance

Hotline: 1-800-LATTICE (Domestic)
 1-408-826-6002 (International)
 e-mail: techsupport@latticesemi.com