Understanding PLL Timing for Stratix II Devices

March 2006, ver. 1.0 Application Note 411

Introduction

Stratix® II devices have up to 12 phase-locked loops (PLLs) that provide robust clock management and synthesis for on-chip clock management, external system clock management, and high-speed I/O interfaces. The Stratix II PLL is highly versatile and can be used as a zero delay buffer, jitter attenuator, low skew fan-out buffer, and as a frequency synthesizer. To take advantage of the numerous features and capabilities provided by the Stratix II PLLs, a full understanding of all reports and analysis performed by the Quartus® II Timing Analyzer is necessary. This application note provides details, examples, and guidelines on how to read and understand the various Timing Analysis reports relating to PLLs, and how the analysis is performed by the Timing Analyzer.

This application note is applicable to designs that target Stratix II devices using the Quartus II software version 5.1 and earlier.

Stratix II PLL Overview

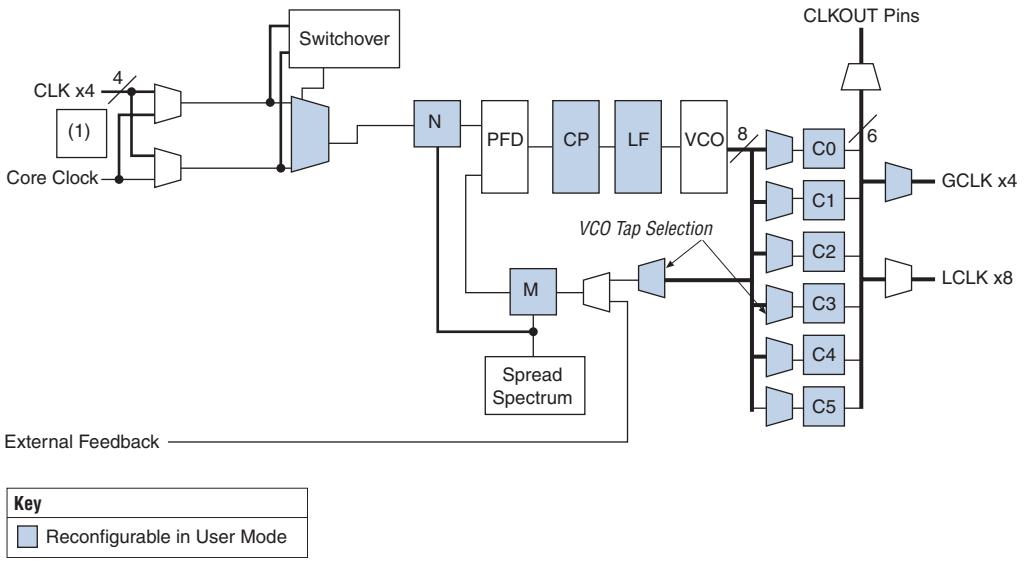
One of the primary objectives of a PLL is to synchronize the phase and frequency of an internal/external clock to an input reference clock. Numerous PLL components must work together to achieve this phase alignment.

Stratix II PLLs align the rising edge of the reference input clock to a feedback clock using a phase frequency detector (PFD) (Figure 1). The falling edges are determined by the duty cycle specifications. The PFD produces an up or down signal that determines whether the voltage-controlled oscillator (VCO) should operate at a higher or lower frequency. The PFD output is applied to the charge pump and loop filter, which produce a control voltage for setting the frequency of the VCO. If the PFD produces an up signal, the VCO frequency increases, while a down signal causes the VCO frequency to decrease. The PFD outputs these up and down signals to a charge pump. If the charge pump receives an up signal, current is driven into the loop filter. Conversely, if it receives a down signal, current is drawn from the loop filter. The loop filter converts these up and down signals to a voltage that is used to bias the VCO. The loop filter also removes glitches from the charge pump and prevents voltage over-shoot, which minimizes the jitter on the VCO.

The voltage from the loop filter determines how fast the VCO operates. The VCO is implemented as a four-stage differential ring oscillator. A divide counter *(M)* is inserted in the feedback loop to increase the VCO frequency above the input reference frequency, making the VCO frequency (f_{VCO}) equal to *M* times the input reference clock (f_{REF}). The f_{REF} to the PFD is equal to the input clock (f_{IN}) divided by the pre-scale counter *(N)*. Therefore, the feedback clock (f_{FB}) that is applied to one input of the PFD is locked to the f_{REF} that is applied to the other input of the PFD.

The VCO output feeds up to six post-scale counters (`C[5..0]`) in enhanced PLLs, or up to four post-scale counters (`C[3..0]`) in the case of fast PLLs. These post-scale counters allow a number of harmonically-related frequencies to be produced within the PLL. Figure 1 shows a simplified block diagram of the major components of the Stratix II enhanced PLL.

Figure 1. Stratix II Enhanced PLL



Both enhanced and fast PLLs can be enabled and users can specify the input and output clock frequencies and output clock phase shift using the `altpll` megafunction. The following section gives an overview of the `altpll` megafunction using the MegaWizard® Plug-In Manager.

For more information on the hardware features of the Stratix II PLLs, refer to volume 2 of the *Stratix II Device Handbook*.

Stratix II PLL Megafunction Overview

The `altpll` megafunction provides support for Altera® PLLs including the Stratix II PLLs. The `altpll` megafunction is used to implement different PLL configurations to generate and customize clock signals, distribute clock signals to different devices in a design, reduce clock skew between devices, and generate internal clock signals.

For more information on the `altpll` megafunction, refer to the *altpll Megafunction User Guide*.

Fitter Reports

The Fitter report section of the Compilation Report provides details that show the various configurations of all the PLLs implemented in the Stratix II design. There are two reports that provide information on PLL usage in the design; the PLL Summary report and the PLL Usage report.

Both reports are located in the Resource Section portion of the Compilation Report.

These reports are not generated if the design does not include PLLs.

PLL Summary

The PLL Summary report lists information about the specific kind of PLL chosen in the design. Table 1 describes the different PLL properties shown in the PLL Summary report.

Table 1. PLL Summary Report Values (Part 1 of 3)

PLL Property	Description	Value
PLL type	Shows the type of PLL.	Enhanced / Fast/Auto
PLL mode	Shows the PLL operating mode.	Normal / Zero delay buffer / No compensation / External feedback / Source Synchronous
Feedback source	Shows which output clock has a board level connection to the external feedback input pin to the PLL.	Clock0 / Clock1 / Clock2 / Clock3 / Clock4 / Clock5
Compensate clock	Shows the PLL compensate clock source.	Clock0 / Clock1 / Clock2 / Clock3 / Clock4 / Clock5 / Global clock / Regional clock / Diffio clock
Switchover type	Shows the clock switchover type selected by the user	Auto/Manual

Table 1. PLL Summary Report Values (Part 2 of 3)

PLL Property	Description	Value
Switchover on loss of clock	Shows whether or not the switchover on loss of clock was turned on for the PLL.	On / Off
Switchover counter	Shows the value of the PLL switchover counter.	*<counter value>*
Gate lock counter	Shows the value of the 20-bit counter (0-1048575) that gates the locked output from the PLL.	*<counter value>*
Input frequency 0	Shows the input frequency for input clock 0 in megahertz.	*<input frequency>* MHz
Input frequency 1	Shows the input frequency for input clock 1 in megahertz.	*<input frequency>* MHz
Nominal PFD frequency	Shows the value of the nominal PFD frequency for the PLL in megahertz.	*<frequency>*MHz
Nominal VCO frequency	Shows the value of the nominal VCO frequency for the PLL in megahertz.	*<frequency>*MHz
VCO post scale	Shows the value of the VCO post scale counter.	*<counter value>*
VCO multiply	Shows the multiplication factor for the VCO output clock.	*<value>*
VCO divide	Shows the division factor for the VCO output clock.	*<value>*
Freq min lock	Shows the minimum lock input frequency, in MHz, for the PLL.	*<lock input frequency>* MHz
Freq max lock	Shows the maximum lock input frequency, in MHz, for the PLL.	*<lock input frequency>* MHz
M Initial	Shows the number of initial VCO cycles before the M counter starts.	*<cycles>*
M value	Shows the value of the M counter.	*<counter value>*
N value	Shows the value of the N counter.	*<counter value>*
M2 value	Shows the spread spectrum modulus for the M counter.	*<modulus value>*
N2 value	Shows the spread spectrum modulus for the N counter.	*<modulus value>*
SS counter	Shows the value for the spread spectrum counter for the PLL.	*<counter value>*
Downspread	Shows the downspread percentage for the PLL.	*<downspread percentage>*
Spread frequency	Shows the spread frequency for the PLL in megahertz.	*<frequency>* MHz
Charge pump current	Shows the charge pump current value for the PLL in micro amperes.	*<current> uA*
Loop filter resistance	Shows the value for the loop filter R resistor in ohms.	*<value>* ohms
Loop filter capacitance	Shows the value for the loop filter C capacitor in picofarads.	*<value>* pF
Bandwidth	Shows the typical and min to max bandwidth value for the PLL in megahertz or kilohertz.	*<frequency>* MHz / KHz
Real time configurable	Shows whether the real time configuration option was turned on for the PLL.	On / Off

Table 1. PLL Summary Report Values (Part 3 of 3)		
PLL Property	**Description**	**Value**
Scan chain MIF file	Shows the Memory Initialization File (**.mif**) generated by the compiler to represent initial state of the scan chain. Use with `altpll_reconfig` megafunction to reconfigure the PLL.	*<file name>*
Preserve counter order	This logic option allows you to use specific counters with specific clock outputs.	On / Off
PLL Location	The location of the PLL.	*<PLL location>*
Inclk0 signal	Primary clock input to the PLL.	*<clock name>*
Inclk1 signal	Secondary clock input to the PLL.	*<clock name>*

Figure 2 shows a portion of the PLL Summary section generated for a sample design.

Figure 2. Sample PLL Summary Report

PLL Summary

	Name	lvds_pll:inst2|altpll:altpll_component|pll	EPLL:inst3|altpll:altpll_component|pll
1	PLL type	Fast	Enhanced
2	PLL mode	Normal	Zero Delay Buffer
3	Feedback source	--	--
4	Compensate clock	DIFFIOCLK	clock0
5	Switchover type	--	Auto
6	Switchover on loss of clock	--	Off
7	Switchover counter	--	2
8	Gate lock counter	--	1048575
9	Input frequency 0	210.04 MHz	100.0 MHz
10	Input frequency 1	--	100.0 MHz
11	Nominal PFD frequency	210.0 MHz	5.0 MHz
12	Nominal VCO frequency	840.3 MHz	1000.0 MHz
13	VCO post scale	--	--
14	VCO multiply	4	--
15	VCO divide	1	--
16	Freq min lock	95.44 MHz	97.61 MHz
17	Freq max lock	260.69 MHz	104.06 MHz
18	M VCO Tap	0	0
19	M Initial	1	1
20	M value	4	200
21	N value	1	20
22	M2 value	--	199
23	N2 value	--	20
24	SS counter	--	2000
25	Downspread	--	0.50 %
26	Spread frequency	--	50.0 kHz
27	Charge pump current	72 uA	36 uA
28	Loop filter resistance	1.000000 KOhm	5.000000 KOhm
29	Loop filter capacitance	2 pF	5 pF
30	Bandwidth	6.76 MHz (5.44 MHz to 11.93 MHz)	350 kHz (280 kHz to 610 kHz)
31	Real time reconfigurable	Off	On
32	Scan chain MIF file	--	pll_L2__clk0.mif
33	Preserve counter order	Off	On
34	PLL location	PLL_8	PLL_12
35	Inclk0 signal	rx_inclk	CLK0
36	Inclk1 signal	--	CLK1

PLL Usage

The PLL Usage section reports the values of the specific output clock(s) for the PLLs in the design.

Table 2 describes the PLL properties shown in the PLL Usage report.

Table 2. PLL Usage Report Values

Column Heading	Description	Value
Name	Shows the instance name of the PLL.	*<PLL instance name>*
Output Clock	Shows the output clock you specified for the PLL.	*<clock name>*
Mult	Shows the multiplication factor for the PLL output clock.	*<value>*
Div	Shows the division factor for the PLL output clock.	*<value>*
Output Frequency	Shows the clock frequency value for the PLL output port. This value is equal to the input frequency multiplied by the multiplication factor for the PLL output port, divided by the division factor for the PLL output port.	*<frequency>* MHz
Phase Shift	Shows the value you specified for the phase shift of the PLL output clock, measured in degrees and picoseconds (ps).	*<degrees>* (*<time>* ps)
Duty Cycle	Shows the duty cycle of the clock.	*<high percent / low percent>*
Counter	Shows the counter value you specified for the PLL output clock.	C0 / C1 / C2 / C3 / C4 / C5
Counter Value	Shows the counter value for the PLL. This value is equal to the sum of the high and low cycles specified for the PLL. Bypass is shown if there are no high or low cycles specified for the PLL.	*<counter value>* /Bypass
High/Low	Shows the number of high and low cycles for the PLL.	*<high cycles>*/*<low cycles>* Even/Odd
Cascade Input	Specifies the name of the preceding counter that is feeding the cascade input of the current counter.	*<counter>*
Initial	Shows the number of initial VCO cycles before the counter starts.	*<cycles>*
VCO Tap	Shows the VCO tap value for the counter.	*<tap number>*

Figure 3 shows a portion of the PLL Usage report generated for a sample design.

Figure 3. Sample PLL Usage Report

PLL Usage

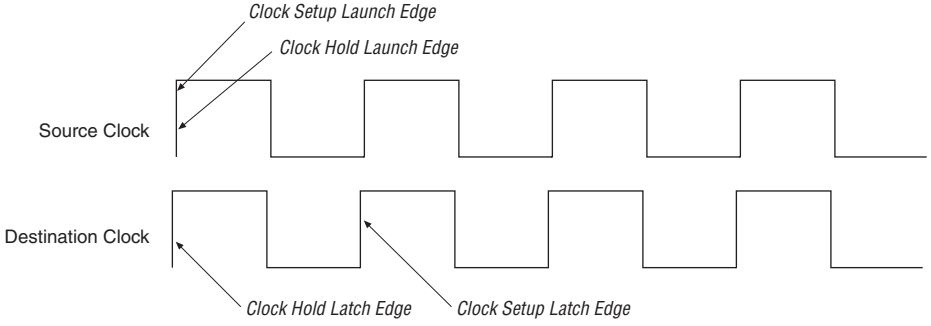
	Name	Output Clock	Mult	Div	Output Frequency	Phase Shift	Duty Cycle	Counter	Counter Value	High / Low	Cascade Input	Initial	VCO Tap
1	lvds_pll:inst2|altpll:altpll_component|_clk0	clock0	1	1	210.04 MHz	0 (0 ps)	50/50	C1	4	2/2 Even	--	1	0
2	lvds_pll:inst2|altpll:altpll_component|_clk1	clock1	1	2	105.02 MHz	0 (0 ps)	50/50	C0	8	4/4 Even	--	1	0
3	lvds_pll:inst2|altpll:altpll_component|_enable0	enable0	1	1	210.04 MHz	0 (0 ps)	--	C1	4	2/2 Even	--	1	0
4	lvds_pll:inst2|altpll:altpll_component|_enable1	enable1	1	2	105.02 MHz	0 (0 ps)	--	C0	8	4/4 Even	--	1	0
5	lvds_pll:inst2|altpll:altpll_component|_sclkout0	sclkout0	4	1	840.16 MHz	0 (0 ps)	50/50	C1	--	--	--	--	0
6	lvds_pll:inst2|altpll:altpll_component|_sclkout1	sclkout1	4	1	840.16 MHz	0 (0 ps)	50/50	C0	--	--	--	--	0
7	EPLL:inst3|altpll:altpll_component|_clk0	clock0	1	1	100.0 MHz	45 (1250 ps)	60/40	C0	10	6/4 Even	--	2	2
8	EPLL:inst3|altpll:altpll_component|_clk1	clock1	1	1	100.0 MHz	90 (2500 ps)	50/50	C1	10	5/5 Even	--	3	4
9	EPLL:inst3|altpll:altpll_component|_clk2	clock2	2	1	200.0 MHz	0 (0 ps)	50/50	C2	5	3/2 Odd	--	1	0
10	EPLL:inst3|altpll:altpll_component|_clk3	clock3	2	1	200.0 MHz	180 (2500 ps)	50/50	C3	5	3/2 Odd	--	3	4

Clock Offset & Clock Latency Differences

Designs that contain PLLs usually have them configured to compensate for any clock network delays driven by the output clocks of the PLL. This PLL compensation delay is modeled either as clock offset or clock latency in the Quartus II software. Both clock offset and clock latency have different effects on the static timing analysis results performed by the Timing Analyzer. This section explains the differences between clock offset and clock latency.

The Timing Analyzer calculates setup relationships and hold relationships based on the launch edge and latch edge of a clock signal. Figure 4 shows how the Timing Analyzer calculates setup relationship and hold relationship.

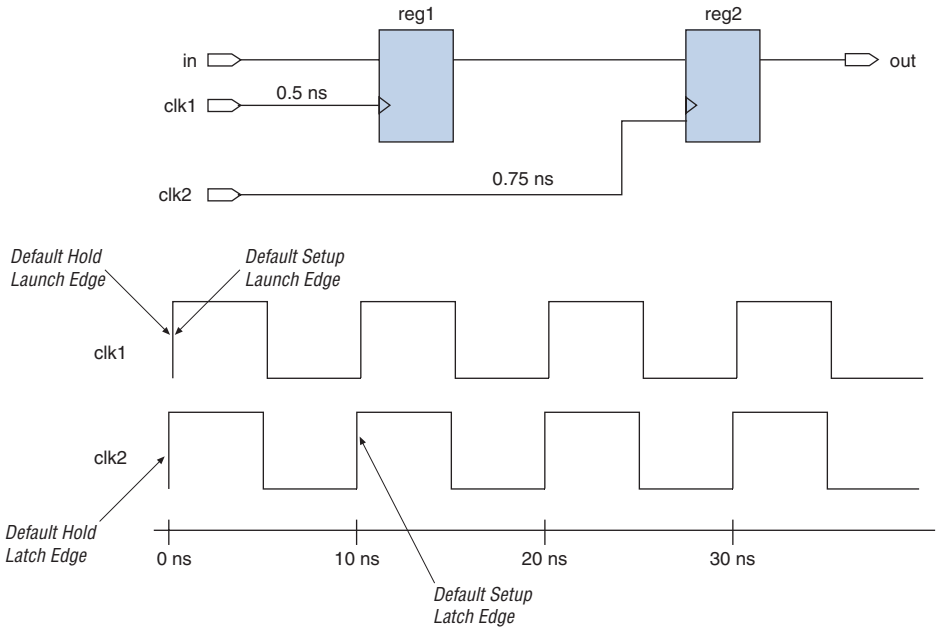
Figure 4. Clock Setup & Hold Check Launch & Latch Edges



For more information on setup and hold relationships, refer to the *Quartus II Timing Analysis* chapter in volume 3 of the *Quartus II Handbook*.

For a simple register-to-register path, as shown in Figure 5, the launch edge is defined as the active clock edge that starts the register-to-register transfer from the source register. The latch edge is defined as the active clock edge that captures the data at the destination register from the source register. By default, the Timing Analyzer selects the closest two consecutive edges from the source and destination registers clock source as the launch and latch edges, respectively. This is shown in Figure 5 with clock signals `clk1` and `clk2` having a 10 ns period.

Figure 5. Simple Register-to-Register Path



The launch and latch edges used for setup or hold check can be the same or different.

Equation 1 shows the calculations for the setup and hold relationships and clock skew for Figure 5.

(1)

Setup Relationship = (Setup Latch Edge + Destination Clock Offset) – (Setup Launch Edge + Source Clock Offset)
= (10.0 + 0.0) – (0.0 + 0.0)
= 10.0 ns

Hold Relationship = (Hold Latch Edge + Destination Clock Offset) – (Hold Launch Edge + Source Clock Offset)
= (0.0 + 0.0) - (0.0 + 0.0)
= 0.0 ns

Clock Skew (Setup Check) = Shortest Clock Path to Destination Register – Longest Clock Path to Source Register
= 0.75 – 0.5
= 0.25 ns

Clock Skew (Hold Check) = Longest Clock Path to Destination Register – Shortest Clock Path to Source Register
= 0.75 – 0.5
= 0.25 ns

Clock Offset

If an offset is associated with a clock signal, the latch and launch edges are affected. The offset moves the clock edges by the amount specified by the offset value. Depending on the polarity of the offset, the adjustment of the clock edges can be pushed forward in time or pulled back in time. Figure 6 shows the effect of a positive offset compared to the original clock signal.

Figure 6. Clock Offset Timing Diagram

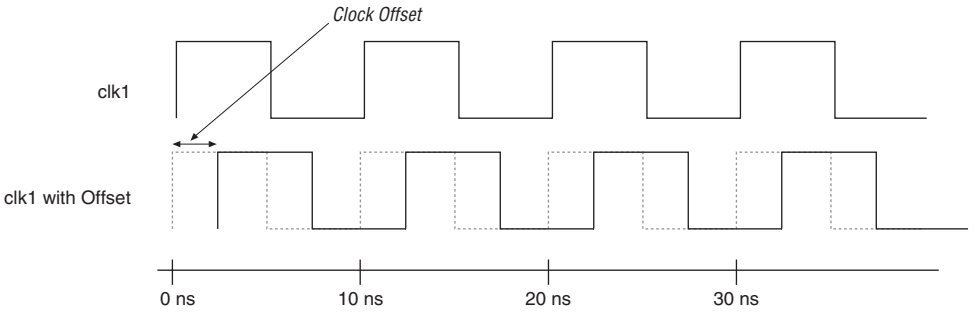


Figure 7 shows a register-to-register path similar to the one shown in Figure 5 except that in this case, the clock port of the source register is being fed by a PLL that has a compensation delay of -1 ns modeled as an offset.

Figure 7. Register-to-Register Path with a PLL

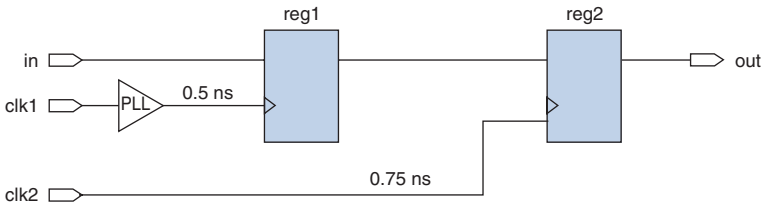
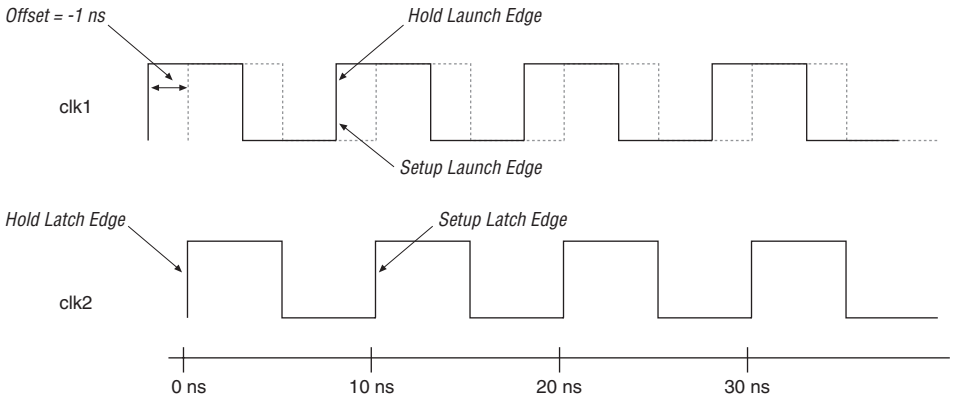


Figure 8 shows the timing diagram of the resulting launch and latch edges due to the PLL compensation delay of -1 ns modeled as clock offset.

Figure 8. Offset Timing Diagrams



From the timing diagram in Figure 8, the setup and hold relationships are changed due to the offset. Equation 2 calculates the setup and hold relationship for the register-to-register path given in Figure 7, based on a 10 ns clock period for both `clk1` and `clk2`.

(2)

Setup Relationship = (Setup Latch Edge + Destination Clock Offset*) – (Setup Launch Edge + Source Clock Offset*)
= (10.0 - 0.0) – (10 - 1)
= 1.0 ns

Hold Relationship = (Hold Latch Edge + Destination Clock Offset*) – (Hold Launch Edge + Source Clock Offset*)
= (0.0 + 0.0) – (10 - 1.0)
= -9.0 ns

* Clock Offset = PLL Compensation Delay + User Requested Phase

Equation 3 shows the clock skew calculation for the register-to-register path shown in Figure 8, with the PLL compensation delay modeled as clock offset.

(3)

$$\begin{aligned}\text{Clock Skew} &= \text{Shortest Clock Path to Destination Register} - \\ &\quad \text{Longest Clock Path to Source Register} \\ &= 0.75 - 0.5 \\ &= 0.25 \text{ ns}\end{aligned}$$

You can locate the offset value by performing the List Path command on any clock setup or clock hold path.

Figure 9 is an example of the clock setup path details with the offset value displayed (line 12.)

Figure 9. Clock Setup With Clock Offset

```
1   Info: Slack time is 3.043 ns for clock "pll_clk" between source register "src_reg" and destination register
2     "dest_reg"
3         Info: + Largest register to register requirement is 3.867 ns
4             Info: + Setup relationship between source and destination is 2.544 ns
5                 Info: + Latch edge is 10.000 ns
6                     Info: Clock period of Destination clock "pll_clk" is 10.000 ns with
7                     offset of 0.000 ns and duty cycle of 50
8                     Info: Multicycle Setup factor for Destination register is 1
9                 Info: - Launch edge is 7.456 ns
10                    Info: Clock period of Source clock
11                    "mypll_test:inst|altpll:altpll_component|_clk0" is 10.000 ns with
12                    offset of -2.544 ns and duty cycle of 50
13                    Info: Multicycle Setup factor for Source register is 1
14            Info: + Largest clock skew is 1.509 ns
15            Info: - Micro clock to output delay of source is 0.176 ns
16            Info: - Micro setup delay of destination is 0.010 ns
17        Info: - Longest register to register delay is 0.824 ns
```

Offset Value — line 12

Figure 10 is an example of the clock hold path details with the offset value displayed (line 14).

Figure 10. Clock Hold with Clock Offset

```
1   Info: Minimum slack time is 6.847 ns for clock "pll_clk" between source register "src_reg" and destination
2   register "dest_reg"
3       Info: + Shortest register to register delay is 0.824 ns
4       Info: - Smallest register to register requirement is -6.023 ns
5           Info: + Hold relationship between source and destination is -7.456 ns
6               Info: + Latch edge is 0.000 ns
7               Info: Clock period of Destination clock "pll_clk" is 10.000 ns with offset of
8                0.000 ns and duty cycle of 50
9                   Info: Multicycle Setup factor for Destination register is 1
10                  Info: Multicycle Hold factor for Destination register is 1
11              Info: - Launch edge is 7.456 ns
12              Info: Clock period of Source clock
13              mypll_test:inst|altpll:altpll_component|_clk0" is 10.000 ns with offset of
14              -2.544 ns and duty cycle of 50
15                  Info: Multicycle Setup factor for Source register is 1
16                  Info: Multicycle Hold factor for Source register is 1
17          Info: + Smallest clock skew is 1.509 ns
18          Info: - Micro clock to output delay of source is 0.176 ns
19          Info: + Micro hold delay of destination is 0.100 ns
```

Offset Value

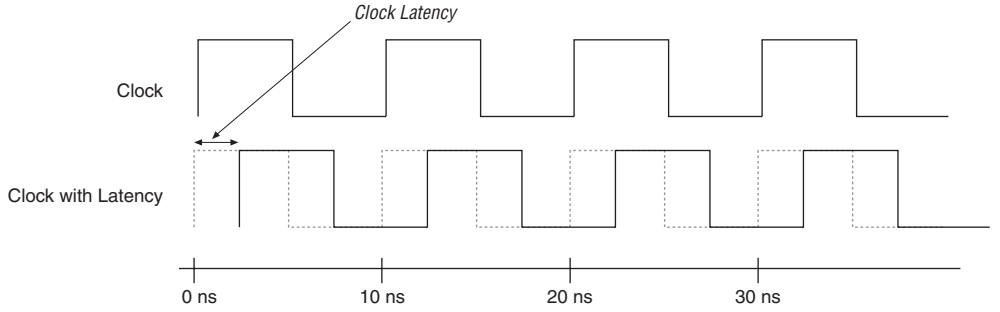
Clock Latency

In contrast to clock offset, clock latency does not affect the setup and hold relationships of register-to-register paths. Clock latency can be viewed as the delay on the clock path that affects only the clock skew calculation.

Any user-specified or requested phase on the output of the PLLs is modeled as an offset.

If clock latency is associated with a clock signal, the latch and launch edges are not affected, but the clock skew is affected. Essentially, clock latency delays the clock signal by the amount specified by the latency value. Depending on the polarity of the latency, the adjustment of the clock delay is pushed forward in time or pulled back in time. Figure 11 shows the affect of clock latency on a clock signal compared to the original clock signal.

Figure 11. Clock Latency



As a result of the latency delaying the clock signal, the clock skew value changes. Figure 12 shows a register-to-register path with the clock port of the source register being fed by a PLL that has a compensation delay of -1 ns modeled as clock latency.

Figure 12. Register-to-Register Path with a PLL

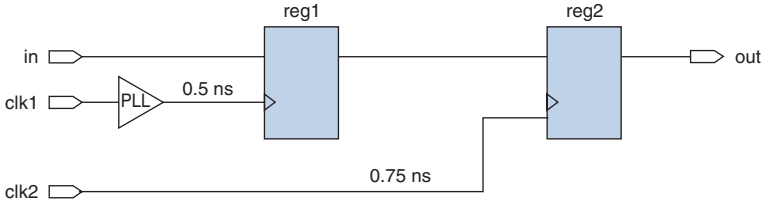
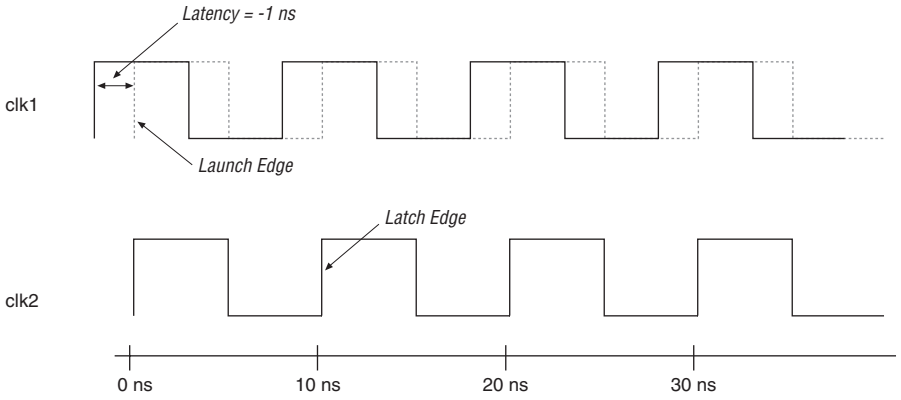


Figure 13 shows the timing diagram of the resulting clock signals due to the clock latency of -1 ns due to the PLL compensation delay.

Figure 13. Latency Timing Diagrams



The Timing Analyzer supports both the Early and Late Latency assignment. Depending on the type of check performed, the appropriate latency assignment is used.

For more information on the Early and Late Clock Latency assignments, refer to the *Quartus II Timing Analysis* chapter in volume 3 of the *Quartus II Handbook*.

From the timing diagrams in Figure 13, the clock signal `clk1` is delayed by 1 ns. Equation 4 calculates the clock skew for a setup check, and Equation 5 calculates the clock skew for a hold check for the register-to-register path given in Figure 12, as well as the clock skew for the path.

(4)
$$\begin{aligned}\text{Clock Skew} &= (\text{Shortest Clock Path to Destination Register} + \text{Early Clock Latency}) - \\ &\quad (\text{Longest Clock Path to Source Register} + \text{Late Clock Latency}) \\ &= (0.75 + 0.0) - (0.5 - 1.0) \\ &= 1.25 \text{ ns}\end{aligned}$$

(5)
$$\begin{aligned}\text{Clock Skew} &= (\text{Longest Clock Path to Destination Register} + \\ &\quad \text{Late Clock Latency}) - (\text{Shortest Clock Path to} \\ &\quad \text{Source Register} + \text{Early Clock Latency}) \\ &= (0.75 - 0.0) - (0.5 - 1.0) \\ &= 1.25 \text{ ns}\end{aligned}$$

Equation 6 shows the setup and hold relationships for the register-to-register path in Figure 12 with latency enabled.

(6)

Setup Relationship = (Latch Edge + Destination Clock Offset) – (Launch Edge + Source Clock Offset)
= (10.0 + 0.0) – (0.0 + 0.0)
= 10.0 ns

Hold Relationship = (Latch Edge + Destination Clock Offset) – (Launch Edge + Source Clock Offset)
= (0.0 + 0.0) – (00 + 0.0)
= 0.0 ns

* Clock Offset = User Requested Phase

The PLL compensation delay does not affect the setup relationship and hold relationship calculations.

You can determine the clock latency value by using the List Path command on any clock setup or clock hold path. Figure 14 is an example of the clock setup path with the clock latency values displayed on lines 6, 15, and 16.

Figure 14. Clock Setup with Clock Latency

```
1   Info: Slack time is 13.211 ns for clock "pll_clk" between source register "src_reg" and destination register "dst_reg"
2       Info: + Largest register to register requirement is 13.895 ns
3           Info: + Setup relationship between source and destination is 10.000 ns
4           Info: + Largest clock skew is 4.081 ns
5               Info: + Shortest clock path from clock "pll_clk" to destination register is 4.434 ns
6                   Info: + Early clock latency of clock "pll_clk" is 0.000 ns
7                   Info: 1: + IC(0.000 ns) + CELL(1.020 ns) = 1.020 ns; Loc. = PIN_AK15;
8                   Fanout = 5; CLK Node = 'pll_clk'
9                   Info: 2: + IC(2.854 ns) + CELL(0.560 ns) = 4.434 ns; Loc. =
10                  LC_X1_Y68_N4; Fanout = 1; REG Node = 'dst_reg'
11                  Info: Total cell delay = 1.580 ns ( 35.63 % )
12                  Info: Total interconnect delay = 2.854 ns ( 64.37 % )
13              Info: - Longest clock path from clock "mypll_test:inst|altpll:altpll_component|_clk0"
14              to source register is 0.353 ns
15                  Info: + Late clock latency of clock
16                  "mypll_test:inst|altpll:altpll_component|_clk0" is -2.544 ns
17                  Info: 1: + IC(0.000 ns) + CELL(0.000 ns) = -2.544 ns; Loc. = PLL_11;
18                   Fanout= 1; CLK Node = 'mypll_test:inst|altpll:altpll_component|_clk0'
19                  Info: 2: + IC(2.337 ns) + CELL(0.560 ns) = 0.353 ns; Loc. =
20                  LC_X2_Y68_N4; Fanout = 2; REG Node = 'src_reg'
21                  Info: Total cell delay = 0.560 ns ( 19.33 % )
22                  Info: Total interconnect delay = 2.337 ns ( 80.67 % )
23          Info: - Micro clock to output delay of source is 0.176 ns
24          Info: - Micro setup delay of destination is 0.010 ns
```

Clock Latency Values

Figure 15 is an example of the clock hold path with the clock latency values displayed on lines 10, 19, and 20.

Figure 15. Clock Hold with Clock Latency

```
1   Info: Minimum slack time is 932 ps for clock "pll_clk" between source register "src_reg" and
2   destination register "dst_reg"
3       Info: + Shortest register to register delay is 4.937 ns
4       Info: - Smallest register to register requirement is 4.005 ns
5           Info: + Hold relationship between source and destination is 0.000 ns
6           factor for Source register is 1
7           Info: + Smallest clock skew is 4.081 ns
8               Info: + Longest clock path from clock "pll_clk" to destination
9               register is 4.434 ns
10                  Info: + Late clock latency of clock "pll_clk" is 0.000 ns
11                  Info: 1: + IC(0.000 ns) + CELL(1.020 ns) = 1.020 ns; Loc. =
12                  PIN_AK15; Fanout = 5; CLK Node = 'pll_clk'
13                  Info: 2: + IC(2.854 ns) + CELL(0.560 ns) = 4.434 ns; Loc. =
14                  LC_X1_Y68_N4; Fanout = 1; REG Node = 'dst_reg'
15                  Info: Total cell delay = 1.580 ns ( 35.63 % )
16                  Info: Total interconnect delay = 2.854 ns ( 64.37 % )
17              Info: - Shortest clock path from clock
18              "mypll_test:inst|altpll:altpll_component|_clk0" to source register is 0.353 ns
19                  Info: + Early clock latency of clock
20                  "mypll_test:inst|altpll:altpll_component|_clk0" is -2.544 ns
21                  Info: 1: + IC(0.000 ns) + CELL(0.000 ns) = -2.544 ns; Loc. =
22                  PLL_11; Fanout = 1; CLK Node =
23                  'mypll_test:inst|altpll:altpll_component|_clk0'
24                  Info: 2: + IC(2.337 ns) + CELL(0.560 ns) = 0.353 ns; Loc. =
25                  LC_X2_Y68_N3; Fanout = 2; REG Node = 'src_reg'
26                  Info: Total cell delay = 0.560 ns ( 19.33 % )
27                  Info: Total interconnect delay = 2.337 ns ( 80.67 % )
28              Info: - Micro clock to output delay of source is 0.176 ns
29              Info: + Micro hold delay of destination is 0.100 ns
```

Clock Latency Values

Any user-specified phase to any of the output clocks of the PLL is modeled as an offset.

The enable clock latency option in the Timing Analyzer allows clock offset to be modeled as clock latency. When this option is turned on, the clock offset affects clock skew. It does not affect the setup and hold relationships of any register-to-register paths.

Table 3 shows a summary of the timing analysis results for the design shown in Figure 7, and compares the effects of the PLL compensation delay modeled as clock offset or clock latency.

Table 3. PLL Compensation Delay Relationships Modeled as Clock Offset Versus Clock Latency

	Clock Offset (Enable Clock Latency = Off)	Clock Latency (Enable Clock Latency = ON)
Source Clock Period	10.00 ns	
Destination Clock Period	10.00 ns	
Setup Relationship	1.00 ns	10.00 ns
Hold Relationship	-9.00 ns	0.00 ns
Clock Skew (Setup Check)	0.25 ns	1.25 ns
Clock Skew (Hold Check)	0.25 ns	1.25 ns

Enable the clock latency option to prevent the modification of the setup and hold relationships in designs that contain PLLs. This allows you to maintain the setup and hold relationships and have the PLL compensation delay modeled as clock skew for a static timing analysis. In addition, the use of clock latency prevents the use of multicycle assignments to define the correct setup and hold relationships.

Clock Switchover

Clock switchover is a feature that allows the PLL to switch between two input reference clocks. You can use clock switchover for clock redundancy, or for a dual clock domain application. This feature of the Stratix II PLL can be used to develop a highly reliable system in which you can use a redundant clock in case the primary clock fails. Clock switchover can be performed automatically, when the clock is no longer toggling, or manually through a user control signal.

For PLLs, the Timing Analyzer automatically creates base and derived clock settings based on the input clock frequency and parameterization of the PLL respectively. For example, if the input clock frequency to a PLL is 100 MHz, and the multiplication and division ratio chosen is 5:2, the Timing Analyzer creates a base clock setting of 100 MHz for the input clock to the PLL and computes the derived clock setting of 250 MHz for the output clock from the PLL.

For the Stratix II device family, you can override this default f_{MAX} frequency requirement of the PLL by applying a clock setting directly to the input clock pin of the PLL. For example, if the PLL input clock frequency is set to 100 MHz in the `altpll` megafunction, with a

multiplication and division ratio of 5:2, but an individual clock setting of 200 MHz has been applied to the input clock pin of the PLL, then the output clock of the PLL is timing analyzed with f_{MAX} requirement of 500 MHz, not 250 MHz.

The Timing Analyzer issues a message similar to the one shown in Figure 16 whenever an individual clock setting overrides the default clock setting on the input clock pin of the PLL.

Figure 16. PLL Warning Message

```
Warning: Clock "inclk" frequency requirement of 200.0 MHz overrides "Stratix II" PLL
"my_pll:inst|altpll:altpll_component|_clk0" input frequency requirement of 100.0 MHz
```

When using clock switchover, set individual clock settings on the input clock to the PLL and run timing analysis for all input clock frequencies and phase shifts. This ensures that the design meets timing requirements when running at the primary or secondary clock frequency. This is not required if all input clock frequencies and phase shifts are the same.

For more information on creating clock settings for individual clocks, refer to the *Quartus II Timing Analysis* chapter in volume 3 of the *Quartus II Handbook*.

Clock Uncertainty

Clock uncertainty is defined as the interval of confidence around the ideal clock value such that the measured value is always inside this stated interval.

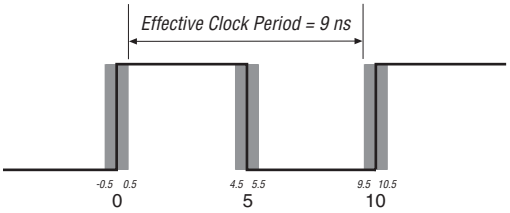
The common sources of clock uncertainty are:

- Clock jitter
- Clock skew
- Duty cycle distortion
- Phase shift error (when using non-zero PLL phase shifts)

Use Clock Setup Uncertainty and Clock Hold Uncertainty assignments to model jitter, skew, and a guard band associated with clock signals. When a clock uncertainty assignment exists for a clock signal, the Timing Analyzer performs the most conservative setup and hold checks.

An uncertainty value of 0.5 ns for `clk0` (with a clock period of 10 ns) means the first rising edge of `clk0` can arrive at any time between -0.5 and 0.5 ns, and the first falling edge of `clk0` can arrive at any time between 4.5 and 5.5 ns. This uncertainty interval effectively reduces the available launch and latch time period to 10 – 1 = 9 ns (Figure 17).

Figure 17. Clock Uncertainty



The clock uncertainty assignments do not change the micro t_{SU} and t_H values of the internal registers of the device, but decrease the clock setup and clock hold relationships for any register-to-register paths.

Clock Setup Uncertainty Assignment

Use the Clock Setup Uncertainty assignment to apply a setup uncertainty value to a clock signal, or as a point-to-point assignment between two clocks. After assigning this requirement, the Timing Analyzer subtracts the specified clock setup uncertainty from the data required time when calculating setup checks and reports this with other data in the Clock Setup timing analysis report.

Figure 18 shows the clock setup report with no uncertainty assignments. Since the clock frequency of the C0 output from the PLL is 300 MHz, the required setup relationship is 3.333 ns. After accounting for the micro t_{CO} and micro t_{SU} of the register, the required longest P2P (point-to-point) time is 3.099 ns. The slack time for this design is reported as 2.171 ns.

Figure 18. Clock Setup Report Before Clock Set-Up Uncertainty

Clock Setup: 'test_pll:inst|altpll:altpll_component|_clk0'

	Slack	Actual fmax (period)	From	To	From Clock	To Clock	Required Setup Relationship	Required Longest P2P Time	Actual Longest P2P Time
1	2.171 ns	Restricted to 816.99 MHz (period = 1.22 ns)	inst1	inst2	test_pll:inst|altpll:altpll_component|_clk0	test_pll:inst|altpll:altpll_component|_clk0	3.333 ns	3.099 ns	0.928 ns

The clock setup uncertainty assignment is set either in the Assignment Editor or with the `set_clock_uncertainty` Tcl command. The following command sets a clock setup uncertainty assignment of 500 ps on the C0 output clock of the PLL:

```
set_clock_uncertainty -to test_pll:inst|alt:altpll_component|_clk0 500ps -setup
```

This uncertainty can account for jitter on the PLL clock output.

By rerunning the Timing Analyzer, the required setup relationship is reduced by 500 ps to 2.833 ns, and the slack time is reduced to 1.671 ns (Figure 19).

Figure 19. Clock Setup Report After Clock Set-Up Uncertainty

Clock Setup: 'test_pll:inst | altpll:altpll_component | _clk0'

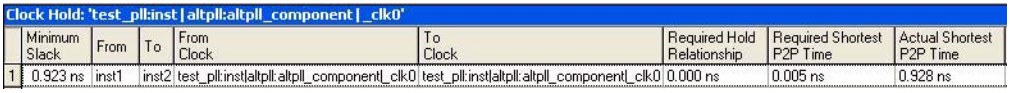
	Slack	Actual fmax (period)	From	To	From Clock	To Clock	Required Setup Relationship	Required Longest P2P Time	Actual Longest P2P Time
1	1.671 ns	Restricted to 816.99 MHz (period = 1.22 ns)	inst1	inst2	test_pll:inst|altpll:altpll_component|_clk0	test_pll:inst|altpll:altpll_component|_clk0	2.833 ns	2.599 ns	0.928 ns

Clock Hold Uncertainty Assignment

Use the Clock Hold Uncertainty assignment to apply a hold uncertainty value to a clock signal, or as a P2P assignment between two clocks. After this requirement is assigned, the specified hold uncertainty is added to the data required time in the Timing Analyzer when calculating hold checks and reports, in addition to other data in the Clock Hold timing analysis report.

Figure 20 shows the clock hold timing report with no uncertainty assignments. The required hold relationship is 0 ns, and after accounting for the micro t_{CO} and micro t_H of the register, the required shortest P2P time is 0.005 ns. The slack time for this design is reported as 0.923 ns.

Figure 20. Clock Hold Report Before Clock Hold Uncertainty



Clock Hold: 'test_pll:inst | altpll:altpll_component | _clk0'

	Minimum Slack	From	To	From Clock	To Clock	Required Hold Relationship	Required Shortest P2P Time	Actual Shortest P2P Time
1	0.923 ns	inst1	inst2	test_pll:inst|altpll:altpll_component|_clk0	test_pll:inst|altpll:altpll_component|_clk0	0.000 ns	0.005 ns	0.928 ns

The clock hold uncertainty assignment is set either in the Assignment Editor or with the `set_clock_uncertainty` Tcl command. The following command sets a clock hold uncertainty assignment of 500 ps on the C0 output clock of the PLL:

```
set_clock_uncertainty -to test_pll:inst|alt:altpll_component|_clk0 0.5ns -hold
```

This uncertainty can account for jitter on the PLL clock output.

By rerunning the Timing Analyzer, the required hold relationship has increased from 0.0 ns to 0.5 ns, reducing the slack time to 0.423 ns for this design (Figure 21).

Figure 21. Clock Hold Report After Clock Hold Uncertainty

Clock Hold: 'test_pll:inst | altpll:altpll_component | _clk0'

	Minimum Slack	From	To	From Clock	To Clock	Required Hold Relationship	Required Shortest P2P Time	Actual Shortest P2P Time
1	0.423 ns	inst1	inst2	test_pll:inst|altpll:altpll_component|_clk0	test_pll:inst|altpll:altpll_component|_clk0	0.500 ns	0.505 ns	0.928 ns

Application Examples

This section provides guidelines and recommendations on how to achieve your PLL design requirements using the Timing Analyzer. Four examples are presented and explained:

- Aligning the Clock Edges at a Dedicated Clock Input Pin, a Core Register & an External Clock Output Pin
- Aligning Clock Edges at a Core Register With a Non-Dedicated Clock Input Pin to the Corresponding PLL
- PLL Compensation Delay Modeled as Clock Offset Versus Clock Latency
- Clock Multiplexers & Timing Analysis Reports

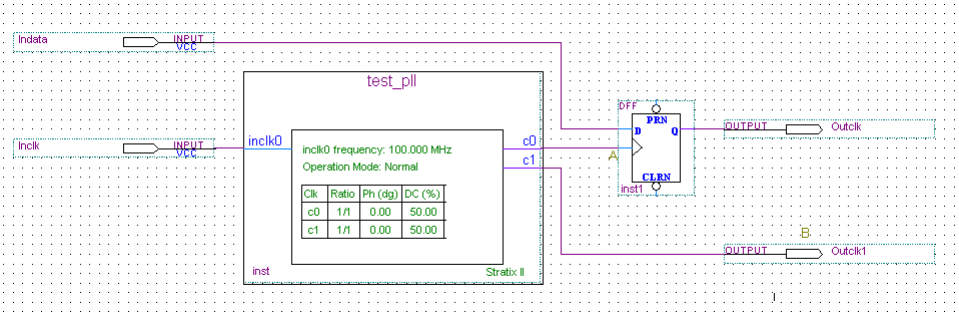
Aligning the Clock Edges at a Dedicated Clock Input Pin, a Core Register & an External Clock Output Pin

Objective: To align the clock edges at the D flipflop (DFF) and an output pin (outclk1) with respect to an input clock (inclk).

Description: You can set the PLL to either normal mode or zero delay buffer mode, and align the input clock with either the internal clock (A) or the external clock output (B), respectively. The goal of this example is to align both clock A and clock B with the input clock.

Figure 22 shows the sample design that is used in this example.

Figure 22. Simple PLL Register Design



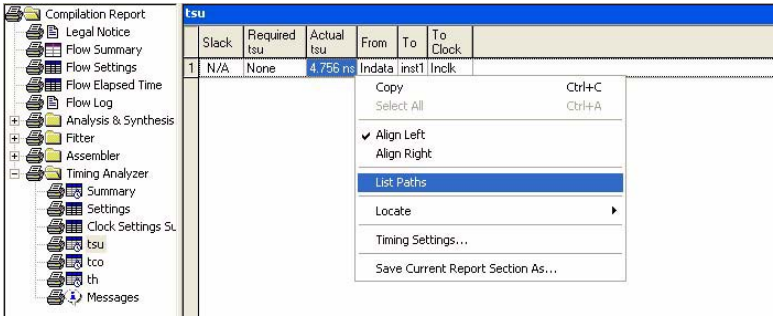
Use the following procedure to complete this design example:

1. Create the simple PLL register design shown in Figure 22, and set the PLL to operate in Normal mode. Normal mode aligns the input clock edge to the PLL (`inclk`) with the clock edge at the register's clock port (A).

2. Perform a full compilation on the design.

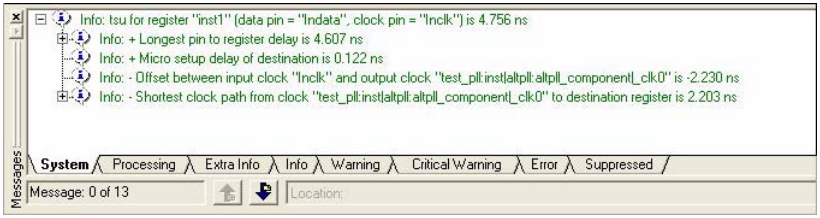
3. Expand the **tsu** section of the timing report, and select **List Paths** (Figure 23).

Figure 23. Timing Analyzer Report



4. Expand **List Paths** in the messages window to see a breakdown of the delays (Figure 24).

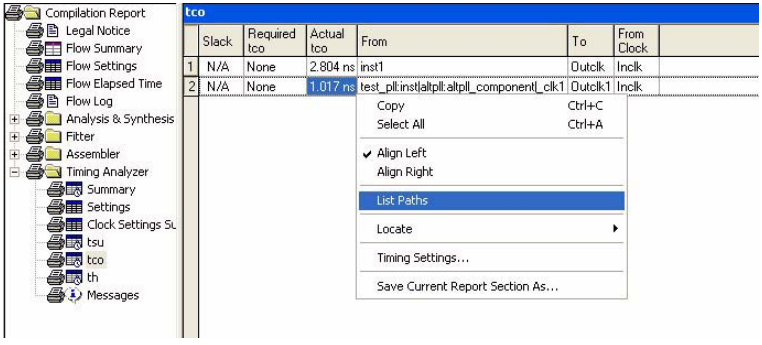
Figure 24. List Path Details



The Timing Analysis list path details show an offset of -2.23 ns between the input clock (`inclk`) and the C0 clock output of the PLL. In addition, the Timing Analyzer reports the shortest clock path from the C0 output of the PLL to point A is 2.203 ns. In effect, this means the input clock edge is aligned to the clock input at the DFF (A).

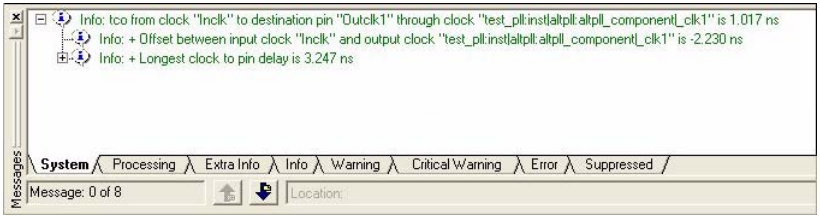
5. Expand the **tco** section of the Timing report and select **List Paths** for the PLL output to the output pin (`outclk1`) (Figure 25).

Figure 25. Timing Analyzer Report



6. Expand the **List Paths** in the messages window to see a breakdown of the delays (Figure 26).

Figure 26. List Path Details

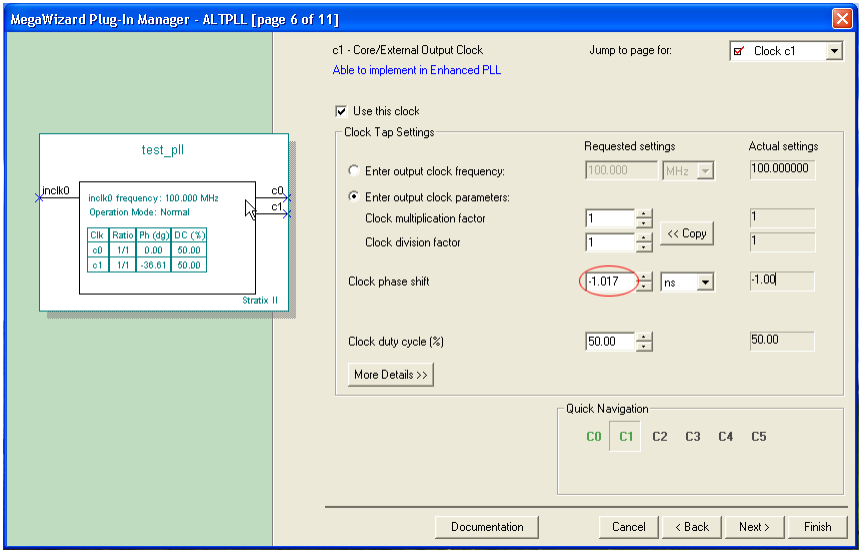


The longest clock to pin delay is 3.247 ns, and the offset between the input clock and the C1 clock output of the PLL is -2.23 ns. Therefore, the t_{CO} reported at point B is 3.247-2.23 = 1.017 ns.

Because the goal is to align the clock at B with both point A and the input clock, you must introduce a phase shift on the C1 clock output from the PLL.

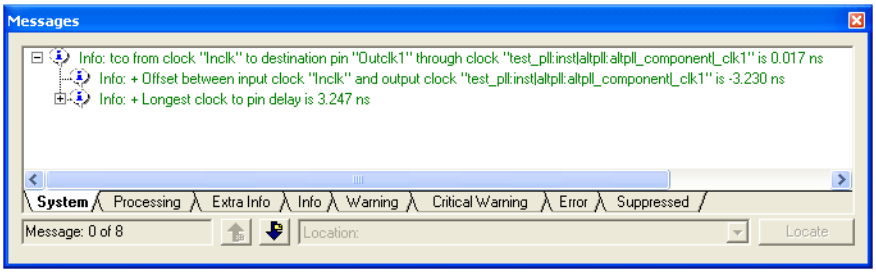
7. Modify the design, as shown in Figure 27, with a phase shift of -1.017 ns on the `C1` clock port of the PLL. The granularity of user phase shift depends on the VCO frequency chosen for the design, so the achievable phase shift is very close to, but not equal to, the required 1.017 ns.

Figure 27. Sample Design



8. Perform a full compilation on the design. Select the **tco** section of the timing report and expand the **List Paths** in the **Messages** window to see a breakdown of the delays (Figure 28).

Figure 28. List Path Details



9. The offset reported between the input clock and the C1 clock output port of the PLL is -3.230 ns because a phase shift of -1 ns was added to the default offset of -2.203 ns. The t_{CO} is now reported as 0.017 ns, so clock at B is leading the input clock by a small margin of 17 ps. Looking at the tsu report for the register, you see that the clock at A is leading the input clock edge by a small margin of 17 ps. In effect, you have successfully aligned the input clock with both A and B.

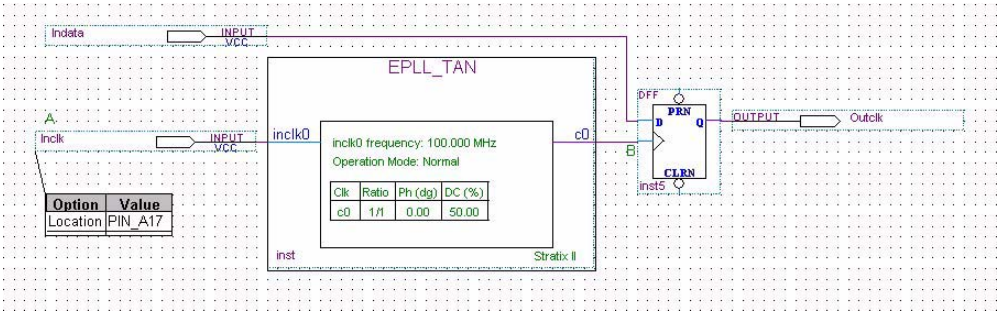
Aligning Clock Edges at a Core Register With a Non-Dedicated Clock Input Pin to the Corresponding PLL

Objective: The PLL is fed by a non-dedicated input pin. Align the clock edges at the input clock (A), and the flipflop (B).

Description: In this example, the PLL (EPLL 6) is fed by pin A17 (CLK14p), which is not the corresponding dedicated clock input pin for the PLL.

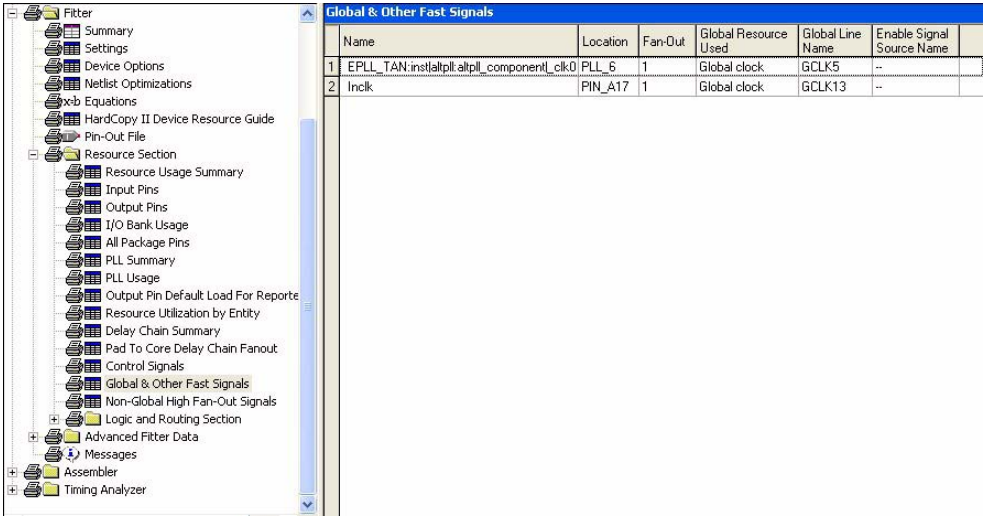
Figure 29 shows the sample design for this example.

Figure 29. Sample Design



Since pin A17 (CLK 14p) is not a dedicated clock input pin to EPLL6, the Quartus II software routes the clock to the `inclk0` port of the PLL through the global clock network (GCLK13). Expand the **Global & Other Fast Signals** section of the Compilation report (Figure 30) to view the exact global resources used to route the clock signals to and from the PLL.

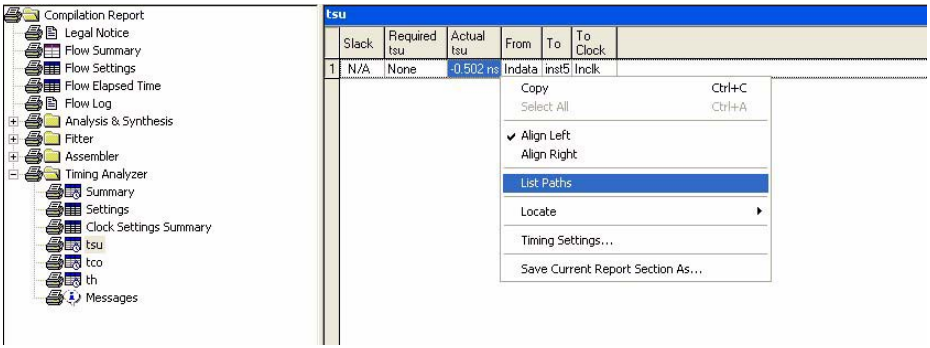
Figure 30. Global and Other Fast Signals Report



Use the following procedure to complete this design example:

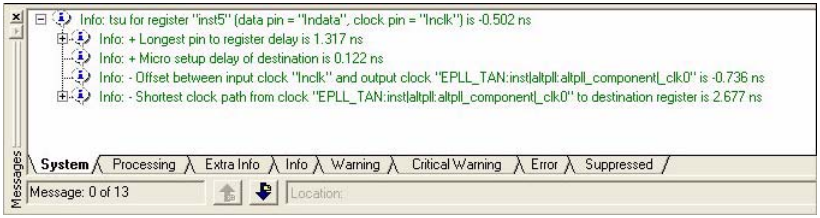
1. Create a test design as shown in Figure 29, and set the PLL to operate in Normal mode.
2. Perform a full compilation on the design.
3. Expand the **tsu** section of the Timing Analyzer report, and select **List Paths** (Figure 31).

Figure 31. Timing Analyzer Report



4. Expand **List Paths** in the messages window to see a breakdown of the delays (Figure 32).

Figure 32. List Path Details

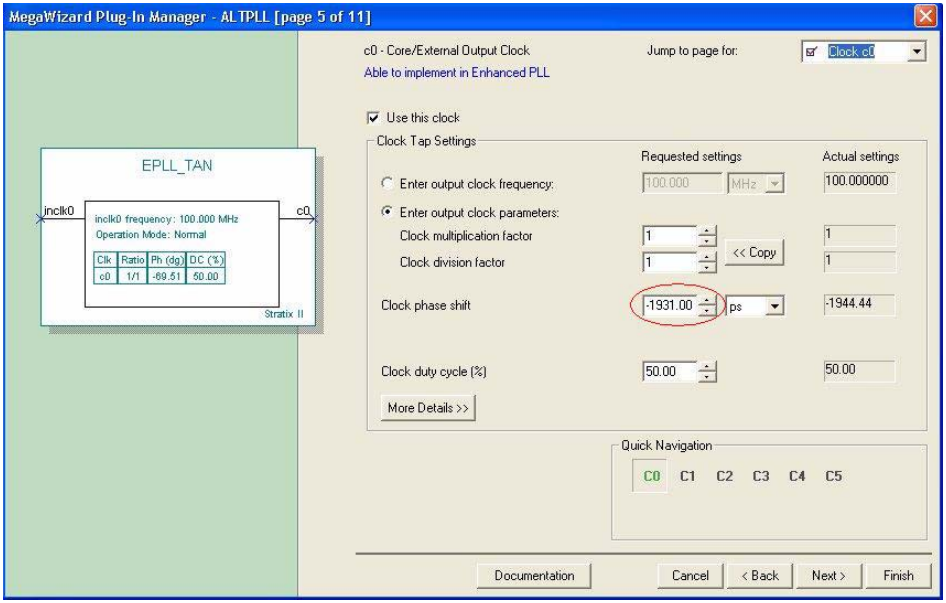


The PLL does not fully compensate for the entire clock path from point A to B. The timing analysis list detail reports an offset of -0.736 ns between the input clock and the C0 clock output of the PLL. The Timing Analyzer also reports the shortest clock path from the C0 output of the PLL to point B is 2.667 ns. This means the input clock edge is not aligned to the clock at point A (phase difference of 1.931 ns).

5. Modify the design in Figure 29 with a phase shift of -69.15 degrees (-1931 ps) on the C0 clock port of the PLL.

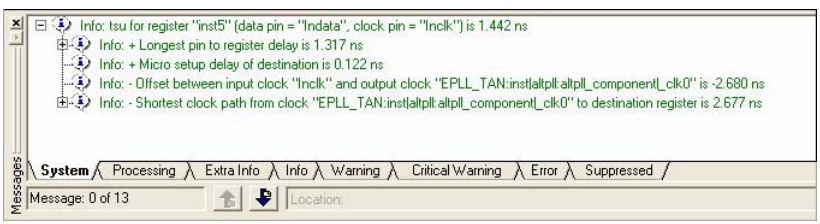
   Figure 33 shows the design in the `altpll` MegaWizard window.

Figure 33. ALTPLL MegaWizard



6. After recompiling the design, expand the **tsu** section of the Timing Analyzer report and select **List Paths**. Expand the **List Paths** in the Messages window to see a breakdown of the delays (Figure 34).

Figure 34. List Path Details



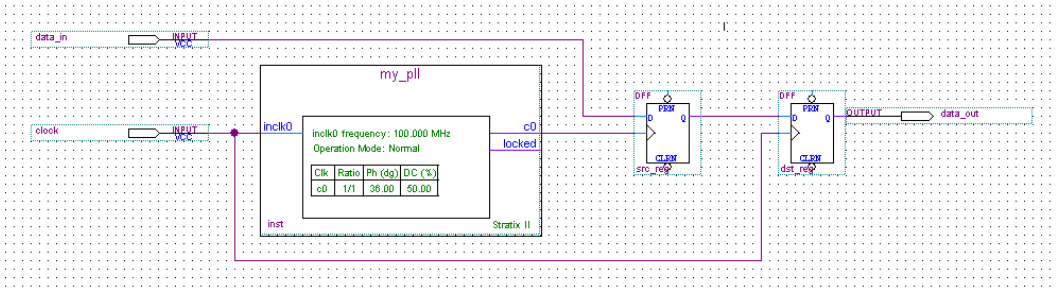
The shortest clock path from the PLL output port to point B matches the offset between the input clock and the C1 clock output of the PLL. This means that you have successfully aligned the clock at B with point A.

PLL Compensation Delay Modeled as Clock Offset Versus Clock Latency

This example illustrates the timing differences between clock offset and clock latency for a simple register-to-register path, and reporting for the Timing Analyzer.

Figure 35 shows a simple design that illustrates the differences between modeling the PLL compensation as a clock offset versus clock latency.

Figure 35. Register-To-Register Path

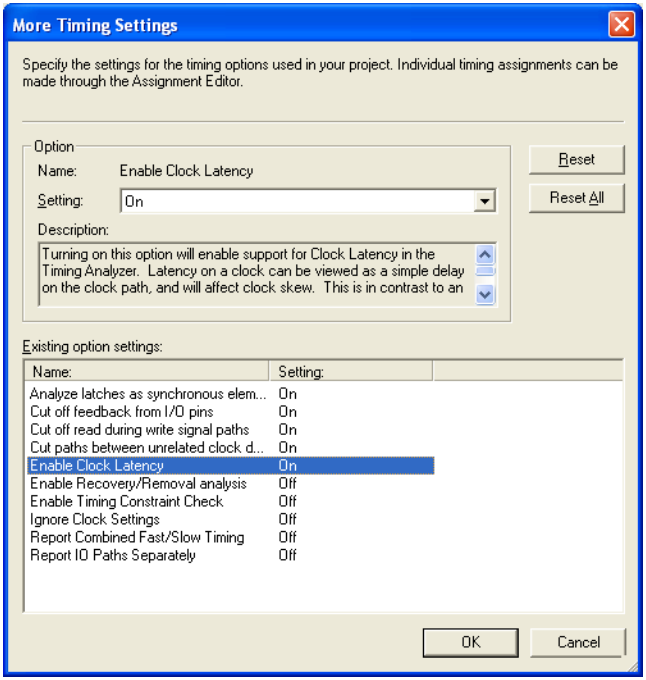


The design in Figure 35 contains a source register, `src_reg`, feeding a destination register, `dst_reg`. The destination register is fed directly by a 10 ns clock signal, `clock`. The source register, clocked by a PLL in normal mode, generates an output clock of 10 ns, and has a user-specified phase of 36 degrees (1.0 ns) with respect to the incoming clock signal, `clock`. To model the board clock trace delays to the `clock` signal, an Early Clock Latency assignment of 400 ps and a Late Clock Latency assignment of 600 ps are applied. The following two assignments apply the clock latency assignments required:

```
set_clock_latency -early -to clock 400ps
set_clock_latency -late -to clock 600ps
```

The enable clock latency option determines whether the Timing Analyzer uses clock offset or clock latency when determining setup relationship, hold relationship, and clock skew calculations. The enable clock latency option is in the **More Timing Settings** dialog box (Figure 36).

Figure 36. More Timing Settings Dialog Box



In this example, two scenarios are presented to model the PLL compensation delay. The first scenario shows an analysis with the enable clock latency set to off (clock offset is used). The second scenario shows an analysis with the option enable clock latency set to on (clock latency is used).

Figure 37 shows the PLL Usage report for the design shown in Figure 35. This report is located in the **Fitter** folder of the **Resource Section**.

Figure 37. PLL Usage Report

PLL Usage

	Name	Output Clock	Mult	Div	Output Frequency	Phase Shift	Duty Cycle	Counter	Counter Value	High / Low	Cascade Input	Initial	VCO Tap
1	my_pll:inst|altpll:altpll_component|_clk0	clock0	1	1	100.0 MHz	36 (1000 ps)	50/50	C0	10	5/5 Even	--	2	0

The PLL Usage report summarizes all specified settings of all PLLs in the design. In this example, the Phase Shift column shows the requested Phase Shift of 36 degrees (1000 ps) based upon the output frequency of the PLL.

Clock Offset (Enable Clock Latency = Off)

With the enable clock latency option set to off, any PLL compensation delay is modeled as a clock offset. Figure 38 shows the Clock Setup Summary report for Figure 35.

Figure 38. Clock Setup Summary Report

Clock Setup: 'clock'

	Slack	Actual fmax (period)	From	To	From Clock	To Clock	Required Setup Relationship	Required Longest P2P Time	Actual Longest P2P Time
1	-1.004 ns	None	src_reg	dst_reg	my_pll:inst|altpll:altpll_component|_clk0	clock	1.815 ns	0.369 ns	1.373 ns

From the Clock Setup report, the design has a required setup relationship of 1.815 ns, and a required longest P2P timing of 0.369 ns. To obtain more information on the register-to-register path, perform a List Path from the report to generate the report shown in Figure 39.

Figure 39. Path Details

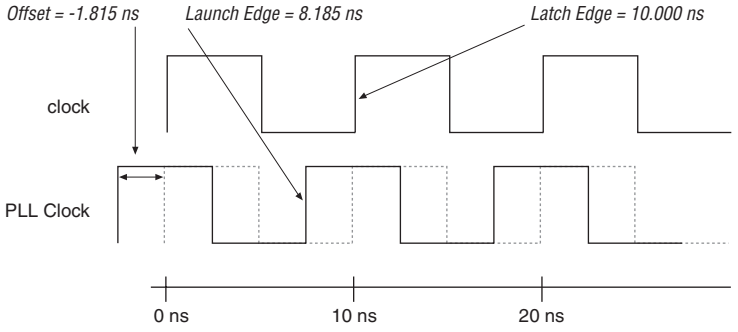
```
1   Info: Slack time is -1.004 ps for clock "clock" between source register "src_reg" and destination register "dst_reg"
2       Info: + Largest register to register requirement is 0.369 ns
3           Info: + Setup relationship between source and destination is 1.815 ns
4               Info: + Latch edge is 10.000 ns
5                   Info: Clock period of Destination clock "clock" is 10.000 ns with offset of
6                   0.000 ns and duty cycle of 50
7                   Info: Multicycle Setup factor for Destination register is 1
8               Info: - Launch edge is 8.185 ns
9                   Info: Clock period of Source clock
10                  "my_pll:inst4|altpll:altpll_component|_clk0" is 10.000 ns with offset of
11                  -1.815 ns and duty cycle of 50
12                  Info: Multicycle Setup factor for Source register is 1
13          Info: + Largest clock skew is -1.233 ns
14              Info: + Shortest clock path from clock "clock" to destination register is 2.023 ns
15              Info: - Longest clock path from clock "my_pll:inst4|altpll:altpll_component|_clk0" to
16              source register is 3.256 ns
17          Info: - Micro clock to output delay of source is 0.109 ns
18          Info: - Micro setup delay of destination is 0.104 ns
19      Info: - Longest register to register delay is 1.373 ns
```

Total PLL Offset

Because the source register is clocked by a PLL in normal mode, an offset is associated with this clock signal. Lines 9 through 11 in Figure 39 show the total PLL offset due to the PLL compensation as well as any user-specified phase shift. In this example, the user-specified phase shift is 36 degrees (1.0 ns).

Figure 40 shows the timing waveforms for the resulting clock signals.

Figure 40. Clock Timing Diagrams with Offset



Equation 7 shows the calculation used to determine the Setup Relationship shown in the **Clock Setup** report.

(7)

$$\begin{aligned}\text{Setup Relationship} &= (\text{Latch Edge} + \text{Offset}) - (\text{Launch Edge} + \text{Offset}) \\ &= (10.0 + 0.0) - (10 + [-1.815]) \\ &= 1.815 \text{ ns}\end{aligned}$$

Equation 8 shows the clock skew calculation.

(8)

$$\begin{aligned}\text{Clock Skew} &= \text{Shortest Clock Path to Destination Register} - \\ &\quad \text{Longest Clock Path to Source Register} \\ &= 2.023 - 3.256 \\ &= -1.233 \text{ ns}\end{aligned}$$

By default, the Timing Analyzer performs a conservative setup check for all register-to-register paths. As a result, the PLL clock edge at 8.185 ns is taken as the launching edge, and the clock edge at 10.0 ns is taken as the latching edge. The setup relationship is 1.815 ns. However, in the majority of cases, the latching edge occurs at 20 ns with respect to the launching edge at 8.185 ns. To specify the correct latch edge to the Timing Analyzer, a Multicycle assignment of 2 is required.

Figure 41 shows the resulting path details after the Multicycle assignment has been applied.

Figure 41. Clock Setup Check with Offset & Multicycle

```
1   Info: Slack time is 8.996 ns for clock "clock" between source register "src_reg" and destination register "dst_reg"
2       Info: + Largest register to register requirement is 10.369 ns
3           Info: + Setup relationship between source and destination is 11.815 ns
4               Info: + Latch edge is 20.000 ns
5                   Info: Clock period of Destination clock "clock" is 10.000 ns with offset of
6                   0.000 ns and duty cycle of 50
7               Info: Multicycle Setup factor for Destination register is 2
8           Info: - Launch edge is 8.185 ns
9               Info: Clock period of Source clock "my_pll:inst4|altpll:altpll_component|_clk0"
10              is 10.000 ns with offset of -1.815 ns and duty cycle of 50
11              Info: Multicycle Setup factor for Source register is 1
12          Info: + Largest clock skew is -1.233 ns
13              Info: + Shortest clock path from clock "clock" to destination register is 2.023 ns
14              Info: - Longest clock path from clock "my_pll:inst4|altpll:altpll_component|_clk0" to
15              source register is 3.256 ns
16          Info: - Micro clock to output delay of source is 0.109 ns
17          Info: - Micro setup delay of destination is 0.104 ns
18      Info: - Longest register to register delay is 0.501 ns
```

Multicycle of 2 Assigned

Latch Edge Occurs at 20.0 ns

Line 7 in Figure 41 indicates that a multicycle assignment of 2 has been applied. In addition, that latch edge now occurs at 20.0 ns, as indicated on line 4 of Figure 41.

Clock Latency (Enable Clock Latency = On)

With the enable clock latency option set to on, any PLL compensation delay is modeled as a Clock Latency. Figure 42 shows the Clock Setup report for Figure 35.

Figure 42. Clock Setup Summary Report

Clock Setup: 'clock'

	Slack	Actual fmax (period)	From	To	From Clock	To Clock	Required Setup Relationship	Required Longest P2P Time	Actual Longest P2P Time
1	8.796 ns	None	src_reg	dst_reg	my_pll:inst|altpll:altpll_component|_clk0	clock	9.000 ns	10.169 ns	1.373 ns

From the Clock Setup report, the design has a required setup relationship of 9.0 ns, and a required longest P2P timing of 10.169 ns. To obtain more information on the register-to-register path, perform a List Path from the report to generate the report shown in Figure 43.

Figure 43. Path Details

```
1   Info: Slack time is 8.796. ns for clock "clock" between source register "src_reg" and destination register "dst_reg"
2           Info: + Largest register to register requirement is 10.169 ns
3                   Info: + Setup relationship between source and destination is 9.000 ns
4                           Info: + Latch edge is 10.000 ns
5                                   Info: Clock period of Destination clock "clock" is 10.000 ns with offset
6                                   of 0.000 ns and duty cycle of 50
7                           Info: Multicycle Setup factor for Destination register is 1
8                           Info: - Launch edge is 1.000 ns
9                                   Info: Clock period of Source clock
10                                  "my_pll:inst4|altpll:altpll_component|_clk0" is 10.000 ns with offset
11                                  of 1.000 ns and duty cycle of 50
12                                  Info: Clock offset from Source is based on specified offset of 0.000 ns
13                                  and phase shift of 36.000 degrees of the derived clock
14                                  Info: Multicycle Setup factor for Source register is 1
15          Info: + Largest clock skew is 1.382 ns
16                  Info: + Shortest clock path from clock "clock" to destination register is 2.423 ns
17                          Info: + Early clock latency of clock "clock" is 0.400 ns
18                  Info: - Longest clock path from clock "my_pll:inst4|altpll:altpll_component|_clk0"
19                  to source register is 1.041 ns
20                          Info: + Late clock latency of clock "my_pll:inst4|altpll:altpll_component|_clk0"
21                          is -2.215 ns
22          Info: - Micro clock to output delay of source is 0.109 ns
23          Info: - Micro setup delay of destination is 0.104 ns
24  Info: - Longest register to register delay is 1.373 ns

Phase Shift Offset 1.000 ns
```

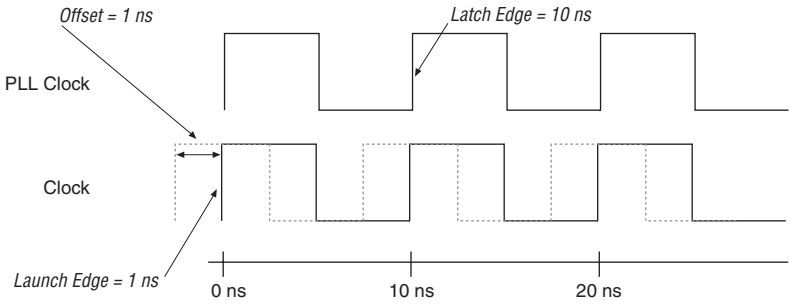
From the path details shown in Figure 43, you can determine the various offset and latency values used by the Timing Analyzer.

The source register, `src_reg`, is clocked by the output of the PLL with a user-specified phase shift of 36 degrees (1.0 ns.) Lines 10 and 11 of Figure 43 show this phase shift as an offset of 1.0 ns.

Any user-specified or requested phase on the output of the PLLs is modeled as an offset.

Figure 44 shows the timing waveforms for resulting clock signals.

Figure 44. Clock Timing Diagrams with Latency



Equation 9 shows the parameters that produce the setup relationship shown in the Clock Setup Summary report (Figure 42).

(9)

$$\begin{aligned}\text{Setup Relationship} &= (\text{Latch Edge} + \text{Offset}) - (\text{Launch Edge} + \text{Offset}) \\ &= (10.0 + 0.0) - (0 + 1.0) \\ &= 9.0 \text{ ns}\end{aligned}$$

With the enable clock latency option set to on, the total PLL compensation value can be located in the Clock Skew section of the detail report. Figure 45 shows just the clock skew calculation shown in Figure 35.

Figure 45. Clock Skew Calculation

```
1        Info: + Largest clock skew is 1.382 ns
2            Info: + Shortest clock path from clock "clock" to destination register is 2.423 ns
3                Info: + Early clock latency of clock "clock" is 0.400 ns
4            Info: - Longest clock path from clock "my_pll:inst4|altpll:altpll_component|_clk0"
5            to source register is 1.041 ns
6                Info: + Late clock latency of clock "my_pll:inst4|altpll:altpll_component|_clk0"
7                is -2.215 ns
```

Late Clock Latency Value
Early Clock Latency Value

For this design, the clock signal, `clock`, has an Early Clock Latency assignment of 400 ps, and a Late Clock Latency assignment of 600 ps applied to it. The early clock latency value is shown on line 3 of Figure 45, and the late clock latency value is shown on lines 6 and 7. The value of the late latency is -2.215. The late latency value is the total latency value associated with the output clock of the PLL, which includes any

user-specified latency and PLL compensation. To determine total PLL compensation based on the value given on lines 6 and 7 of Figure 45, refer to Equation 10.

(10)
$$\begin{aligned}\text{PLL Compensation} &= \text{Total Clock Latency} - \text{User Specified Late Latency}\\ &= -2.215 - (0.600)\\ &= -2.815\ \text{ns}\end{aligned}$$

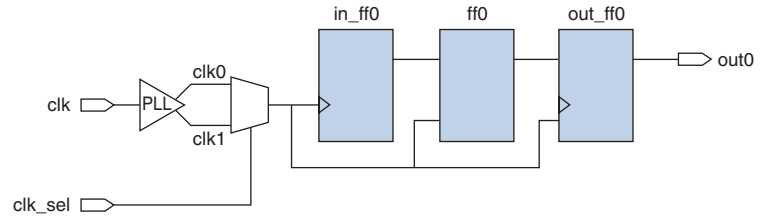
Equation 11 shows the clock skew calculation.

(11)
$$\begin{aligned}\text{Clock Skew} &= \text{Shortest Clock Path to Destination Register} -\\ &\quad\ \text{Longest Clock Path to Source Register}\\ &= 2.423 - 1.041\\ &= 1.382\ \text{ns}\end{aligned}$$

Clock Multiplexers & Timing Analysis Reports

The following example illustrates the timing analysis results for a clock multiplexer design. Figure 46 shows the example design that is used in this example.

Figure 46. Clock Multiplexer Design



The intention of the design in Figure 46 is to have either `clk0` or `clk1` of the PLL drive the internal registers. The clock multiplexing structure used is the `altclkctrl` megafunction and the clock selection is based upon the signal `clk_sel`.

Figure 47 shows the result of the Timing Analyzer report folder. The folder contains the timing analysis results for both `clk0` and `clk1`.

Figure 47. Timing Analysis Result

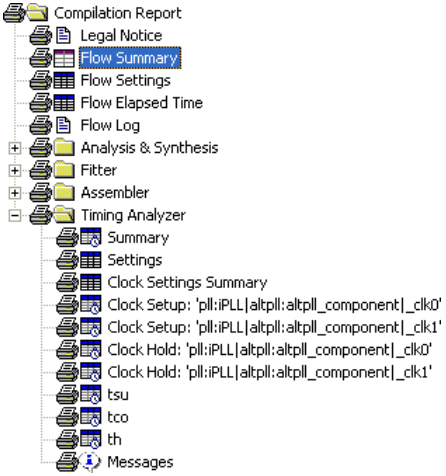


Figure 48 shows the expanded reports for Clock Setup: 'pll:PLL | altpll:altpll_component | _clk0' (`clk0`) and Clock Setup: 'pll:PLL | altpll:altpll_component | _clk1' (`clk1`).

The report selected shows the report for the destination clock. For example, the clock setup report for clk0 shows clock setup analysis where `clk0` is the destination clock.

Figure 48. Clock Setup Paths

Clock Setup: 'pll:iPLL | altpll:altpll_component | _clk1'

	Slack	Actual fmax (period)	From	To	From Clock	To Clock	Required Setup Relationship	Required Longest P2P Time	Actual Longest P2P Time
1	9.305 ns	None	ff0	out_ff0	pll:iPLL|altpll:altpll_component|_clk0	pll:iPLL|altpll:altpll_component|_clk1	10.000 ns	9.816 ns	0.511 ns
2	9.406 ns	None	in_ff0	ff0	pll:iPLL|altpll:altpll_component|_clk0	pll:iPLL|altpll:altpll_component|_clk1	10.000 ns	9.816 ns	0.410 ns

Clock Setup: 'pll:iPLL | altpll:altpll_component | _clk0'

	Slack	Actual fmax (period)	From	To	From Clock	To Clock	Required Setup Relationship	Required Longest P2P Time	Actual Longest P2P Time
1	9.305 ns	None	ff0	out_ff0	pll:iPLL|altpll:altpll_component|_clk1	pll:iPLL|altpll:altpll_component|_clk0	10.000 ns	9.816 ns	0.511 ns
2	9.406 ns	None	in_ff0	ff0	pll:iPLL|altpll:altpll_component|_clk1	pll:iPLL|altpll:altpll_component|_clk0	10.000 ns	9.816 ns	0.410 ns

In the clock setup report for clk0 and clk1, the register-to-register paths analyzed are from clk1 to clk0 and from clk0 to clk1, respectively. However, the paths from clk0 to clk0 and clk1 to clk1 are not analyzed. The Quartus II software default behavior is to analyze and report the worst case timing paths. For this design, these paths are from clk0 to clk1 and clk1 to clk0. However, for this design clock transfers between clk1 to clk1 and clk0 to clk0 are the only valid transfer for analysis.

For the proper analysis between clock transfers clk0 to clk0 and clk1 to clk1, two cut timing assignments are required: one between clk1 and clk0 and one between clk0 and clk1. The cut timing assignment for this design is as follows:

```
#cut clock transfer from clk0 to clk1
set_timing_cut_assignment -from pll:PLL|altpll:altpll_component|_clk0 -to
pll:PLL|altpll:altpll_component|_clk1

#cut clock transfer from clk1 to clk0
set_timing_cut_assignment -from pll:PLL|altpll:altpll_component|_clk1 -to
pll:PLL|altpll:altpll_component|_clk0
```

Figure 49 shows the clock setup reports for both clk0 and clk1 after the cut timing assignments have been applied.

Figure 49. Clock Setup Reports with Cut Timing Assignments

Clock Setup: 'pll:iPLL | altpll:altpll_component | _clk0'

	Slack	Actual fmax (period)	From	To	From Clock	To Clock	Required Setup Relationship	Required Longest P2P Time	Actual Longest P2P Time
1	19.305 ns	Restricted to 816.99 MHz (period = 1.22 ns)	ff0	out_ff0	pll:iPLL|altpll:altpll_component|_clk0	pll:iPLL|altpll:altpll_component|_clk0	20.000 ns	19.816 ns	0.511 ns
2	19.406 ns	Restricted to 816.99 MHz (period = 1.22 ns)	in_ff0	ff0	pll:iPLL|altpll:altpll_component|_clk0	pll:iPLL|altpll:altpll_component|_clk0	20.000 ns	19.816 ns	0.410 ns

Clock Setup: 'pll:iPLL | altpll:altpll_component | _clk1'

	Slack	Actual fmax (period)	From	To	From Clock	To Clock	Required Setup Relationship	Required Longest P2P Time	Actual Longest P2P Time
1	9.305 ns	Restricted to 816.99 MHz (period = 1.22 ns)	ff0	out_ff0	pll:iPLL|altpll:altpll_component|_clk1	pll:iPLL|altpll:altpll_component|_clk1	10.000 ns	9.816 ns	0.511 ns
2	9.406 ns	Restricted to 816.99 MHz (period = 1.22 ns)	in_ff0	ff0	pll:iPLL|altpll:altpll_component|_clk1	pll:iPLL|altpll:altpll_component|_clk1	10.000 ns	9.816 ns	0.410 ns

Each clock setup report now shows the proper analysis, which are the clock transfers from clk0 to clk0 and clk1 to clk1.

Summary

Enhanced Stratix II devices and fast PLLs provide you with complete control of device clocks and system timing. This application note enables you to take advantage of the numerous features and capabilities provided by the Stratix II PLLs by providing a full understanding of all reports and analysis performed by the Quartus II Timing Analyzer along with examples and guidelines on how to read and understand the various Timing Analysis reports relating to PLLs.

References

For more information, refer to the following documents on the Altera website:

- *Quartus II Timing Analysis* chapter in volume 3 of the *Quartus II Handbook*
- *altpll Megafunction User Guide*