5. Clock Networks and PLLs in Stratix IV Devices

SIV51005-3.4

This chapter describes the hierarchical clock networks and phase-locked loops (PLLs) which have advanced features in Stratix® IV devices. It includes details about the ability to reconfigure the PLL counter clock frequency and phase shift in real time, allowing you to sweep PLL output frequencies and dynamically adjust the output clock phase shift.

The Quartus® II software enables the PLLs and their features without external devices. The following sections describe the Stratix IV clock networks and PLLs in detail:

Clock Networks in Stratix IV Devices

The global clock networks (GCLKs), regional clock networks (RCLKs), and periphery clock networks (PCLKs) available in Stratix IV devices are organized into hierarchical clock structures that provide up to 236 unique clock domains (16 GCLKs + 88 RCLKs + 132 PCLKs) within the Stratix IV device and allow up to 71 unique GCLK, RCLK, and PCLK clock sources (16 GCLKs + 22 RCLKs + 33 PCLKs) per device quadrant. Table 5–1 lists the clock resources available in Stratix IV devices.

Table 5–1. Clock Resources in Stratix IV Devices (Part 1 of 2)

Clock Resource	Number of Resources Available	Source of Clock Resource
Clock input pins	32 Single-ended (16 Differential)	CLK[0..15]p and CLK[0..15]n pins
GCLK networks	16	CLK[0..15]p and CLK[0..15]n pins, PLL clock outputs, and logic array
RCLK networks	64/88 (1)	CLK[0..15]p and CLK[0..15]n pins, PLL clock outputs, and logic array
PCLK networks	56/88/112/132 (33 per device quadrant) (2)	DPA clock outputs, PLD-transceiver interface clocks, horizontal I/O pins, and logic array
GCLKs/RCLKs per quadrant	32/38 (3)	16 GCLKs + 16 RCLKs 16 GCLKs + 22 RCLKs

Table 5–1. Clock Resources in Stratix IV Devices (Part 2 of 2)

Clock Resource	Number of Resources Available	Source of Clock Resource
GCLKs/RCLKs per device	80/104 (4)	16 GCLKs + 64 RCLKs 16 GCLKs + 88 RCLKs

Notes to Table 5–1:

(1) There are 64 RCLKs in the EP4S40G2, EP4S100G2, EP4SE230, EP4SGX70, EP4SGX110, EP4SGX180, and EP4SGX230 devices. There are 88 RCLKs in the EP4S40G5, EP4S100G3, EP4S100G4, EP4S100G5, EP4SE360, EP4SE530, EP4SE820, EP4SGX290, EP4SGX360, and EP4SGX530 devices.

(2) There are 56 PCLKs in the EP4SGX70, and EP4SGX110 devices. There are 88 PCLKs in the EP4S40G2, EP4S100G2, EP4SE230, EP4SE360, EP4SGX180, EP4SGX230, EP4SGX290, and EP4SGX360 devices. There are 112 PCLKs in the EP4S40G5, EP4S100G3, EP4S100G4, EP4S100G5, EP4SE530 and EP4SGX530 devices. There are 132 PCLKs in the EP4SE820 device.

(3) There are 32 GCLKs/RCLKs per quadrant in the EP4S40G2, EP4S100G2, EP4SE230, EP4SGX70, EP4SGX110, EP4SGX180, and EP4SGX230 devices. There are 38 GCLKs/RCLKs per quadrant in the EP4S40G5, EP4S100G3, EP4S100G4, EP4S100G5, EP4SE360, EP4SE530, EP4SE820, EP4SGX290, EP4SGX360, and EP4SGX530 devices.

(4) There are 80 GCLKs/RCLKs per entire device in the EP4S40G2, EP4S100G2, EP4SE230, EP4SGX70, EP4SGX110, EP4SGX180, and EP4SGX230 devices. There are 104 GCLKs/RCLKS per entire device in the EP4S40G5, EP4S100G3, EP4S100G4, EP4S100G5, EP4SE360, EP4SE530, EP4SE820, EP4SGX290, EP4SGX360, and EP4SGX530 devices.

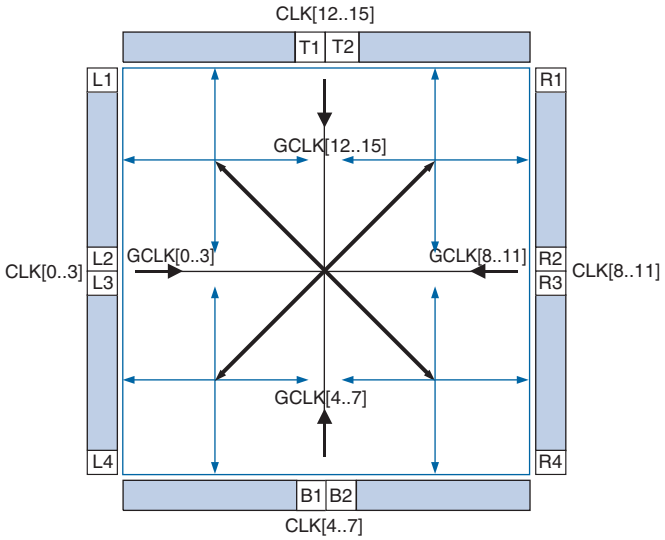
Stratix IV devices have up to 32 dedicated single-ended clock pins or 16 dedicated differential clock pins (`CLK[0..15]p` and `CLK[0..15]n`) that can drive either the GCLK or RCLK networks. These clock pins are arranged on the four sides of the Stratix IV device, as shown in Figure 5–1 through Figure 5–4 on page 5–5.

For more information about how to connect the clock input pins, refer to the *Stratix IV GX and Stratix IV E Device Family Pin Connection Guidelines*.

Global Clock Networks

Stratix IV devices provide up to 16 GCLKs that can drive throughout the device, serving as low-skew clock sources for functional blocks such as adaptive logic modules (ALMs), digital signal processing (DSP) blocks, TriMatrix memory blocks, and PLLs. Stratix IV device I/O elements (IOEs) and internal logic can also drive GCLKs to create internally generated global clocks and other high fan-out control signals; for example, synchronous or asynchronous clears and clock enables. Figure 5–1 shows the CLK pins and PLLs that can drive the GCLK networks in Stratix IV devices.

Figure 5–1. GCLK Networks

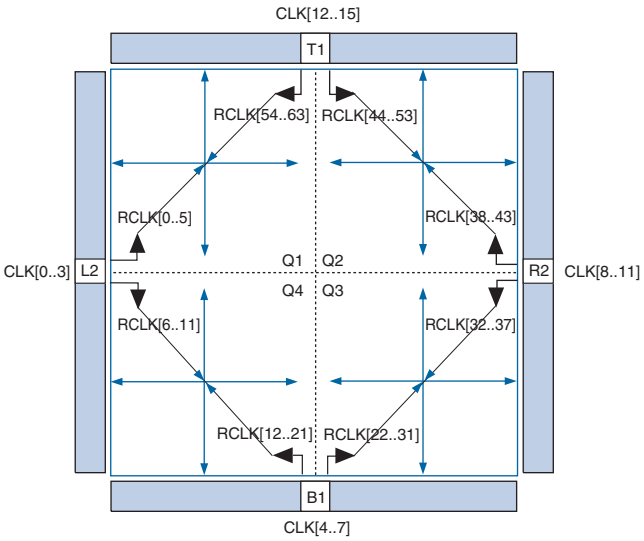


Regional Clock Networks

RCLK networks only pertain to the quadrant they drive into. RCLK networks provide the lowest clock delay and skew for logic contained within a single device quadrant. The Stratix IV device IOEs and internal logic within a given quadrant can also drive RCLKs to create internally generated regional clocks and other high fan-out control signals; for example, synchronous or asynchronous clears and clock enables.

Figure 5–2 through Figure 5–4 on page 5–5 show the CLK pins and PLLs that can drive the RCLK networks in Stratix IV devices.

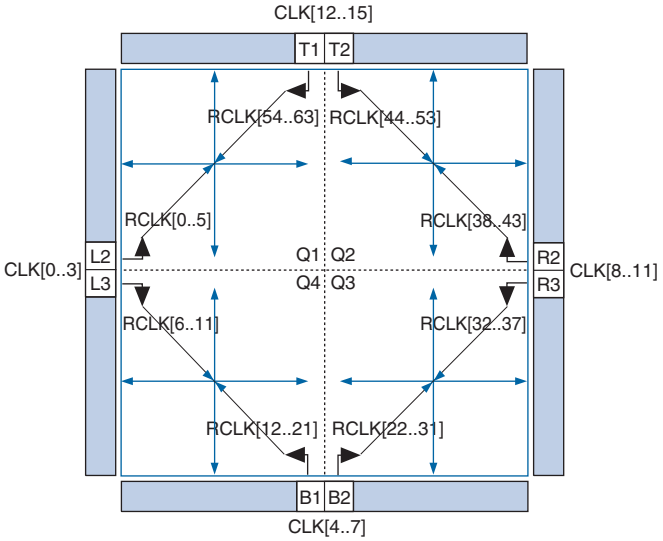
Figure 5–2. RCLK Networks (EP4SE230, EP4SGX70, and EP4SGX110 Devices) (1)



Note to Figure 5–2:

(1) A maximum of four signals from the core can drive into each group of RCLKs. For example, only four core signals can drive into `RCLK[0..5]` and another four core signals can drive into `RCLK[54..63]` at any one time.

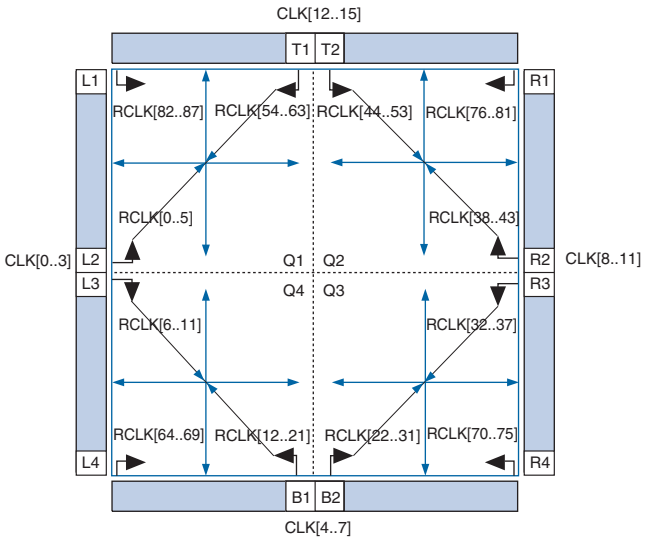
Figure 5–3. RCLK Networks (EP4S40G2, EP4S100G2, EP4SGX180, and EP4SGX230 Devices) (1)



Note to Figure 5–3:

(1) A maximum of four signals from the core can drive into each group of RCLKs. For example, only four core signals can drive into `RCLK[0..5]` and another four core signals can drive into `RCLK[54..63]` at any one time.

Figure 5–4. RCLK Networks (EP4S40G5, EP4S100G3, EP4S100G4, EP4S100G5, EP4SE360, EP4SE530, EP4SE820, EP4SGX290, EP4SGX360, and EP4SGX530 Devices) (1), (2), (3)



Notes to Figure 5–4:

(1) The corner `RCLK[64..87]` can only be fed by their respective corner PLL outputs. For more information about connectivity, refer to Table 5–6 on page 5–13.

(2) The EP4S40G5 and EP4SE360 devices have up to eight PLLs. For more information about PLL availability, refer to Table 5–7 on page 5–19.

(3) A maximum of four signals from the core can drive into each group of RCLKs. For example, only four core signals can drive into `RCLK[0..5]` and another four core signals can drive into `RCLK[54..63]` at any one time.

Periphery Clock Networks

PCLK networks shown in Figure 5–5 through Figure 5–8 on page 5–8 are collections of individual clock networks driven from the periphery of the Stratix IV device. Clock outputs from the dynamic phase aligner (DPA) block, programmable logic device (PLD)-transceiver interface clocks, I/O pins, and internal logic can drive the PCLK networks.

PCLKs have higher skew when compared with GCLK and RCLK networks. You can use PCLKs for general purpose routing to drive signals into and out of the Stratix IV device.

Figure 5–5. PCLK Networks (EP4SGX70 and EP4SGX110 Devices)

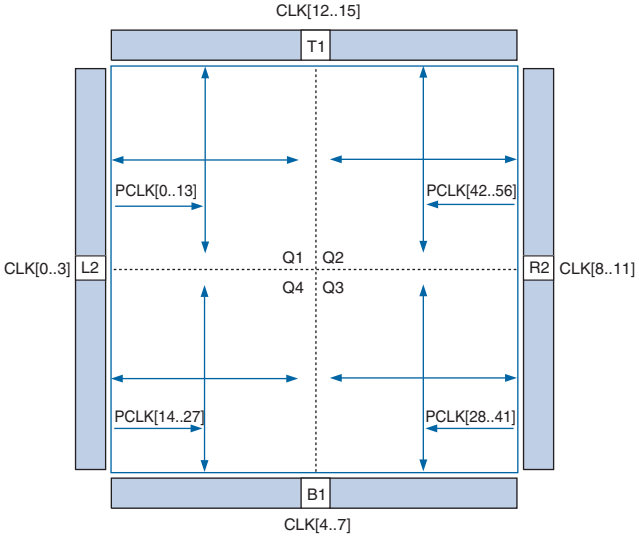
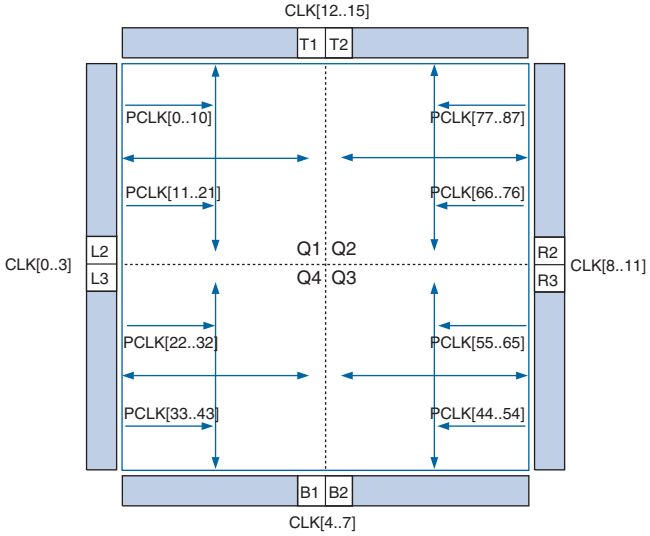


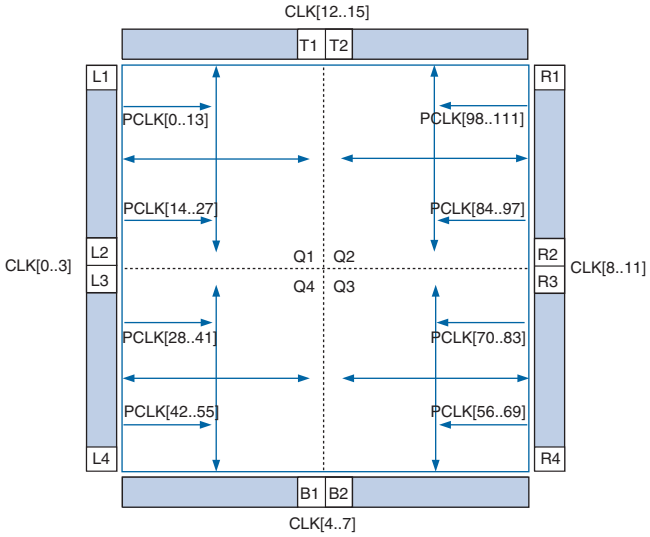
Figure 5–6. PCLK Networks (EP4S40G2, EP4S100G2, EP4SE230, EP4SE360, EP4SGX180, EP4SGX230, EP4SGX290, and EP4SGX360 Devices) (1)



Note to Figure 5–6:

(1) The EP4SE230 device has four PLLs. The EP4SGX290 and EP4SGX360 devices have up to 12 PLLs. For more information about PLL availability, refer to Table 5–7 on page 5–19.

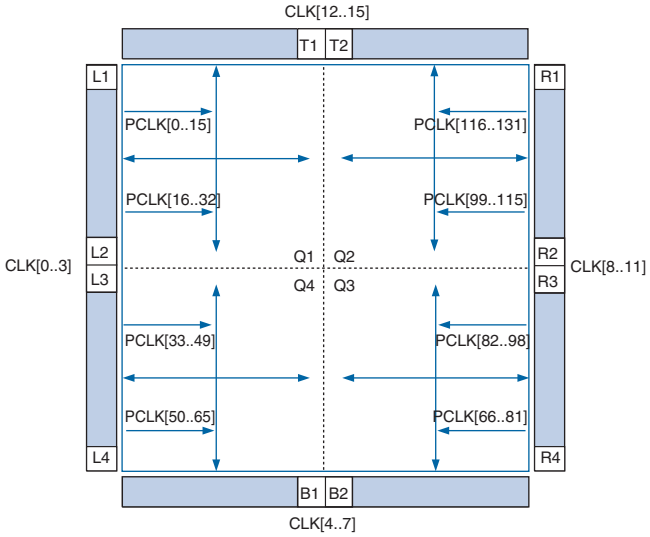
Figure 5–7. PCLK Networks (EP4S40G5, EP4S100G3, EP4S100G4, EP4S100G5, EP4SE530, and EP4SGX530 Devices) (1)



Note to Figure 5–7:

(1) The EP4S40G5 device has eight PLLs. For more information about PLL availability, refer to Table 5–7 on page 5–19.

Figure 5–8. PCLK Networks (EP4SE820 Device)

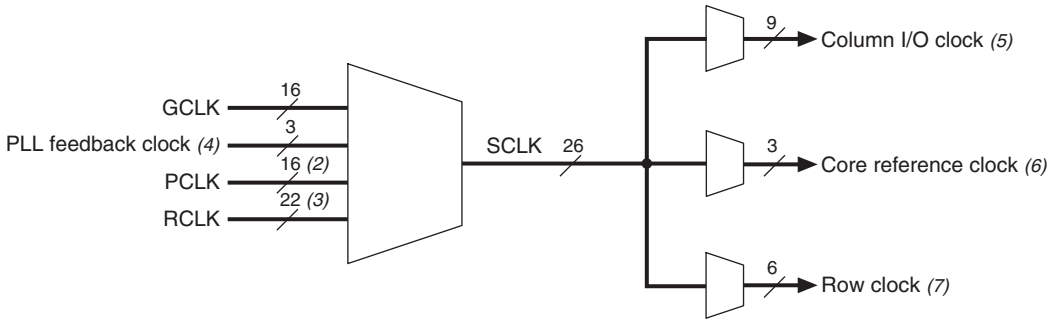


Clock Sources Per Quadrant

There are 26 section clock (SCLK) networks available in each spine clock that can drive six row clocks in each logic array block (LAB) row, nine column I/O clocks, and three core reference clocks. The SCLKs are the clock resources to the core functional blocks, PLLs, and I/O interfaces of the device. Figure 5–9 shows that the SCLKs can be driven by the GCLK, RCLK, PCLK, or the PLL feedback clock networks in each spine clock.

A spine clock is another layer of routing below the GCLKs, RCLKs, and PCLKs before each clock is connected to clock routing for each LAB row. The settings for spine clocks are transparent to all users. The Quartus II software automatically routes the spine clock based on the GCLK, RCLK, and PCLKs.

Figure 5–9. Hierarchical Clock Networks per Spine Clock (1)



Notes to Figure 5–9:

(1) The GCLK, RCLK, PCLK, and PLL feedback clocks share the same routing to the SCLKs. The total number of clock resources must not exceed the SCLK limits in each region to ensure successful design fitting in the Quartus II software.
(2) There are up to 16 PCLKs that can drive the SCLKs in each spine clock in the largest device.
(3) There are up to 22 RCLKs that can drive the SCLKs in each spine clock in the largest device.
(4) The PLL feedback clock is the clock from the PLL that drives into the SCLKs.
(5) The column I/O clock is the clock that drives the column I/O core registers and I/O interfaces.
(6) The core reference clock is the clock that feeds into the PLL as the PLL reference clock.
(7) The row clock is the clock source to the LAB, memory blocks, and row I/O interfaces in the core row.

Clock Regions

Stratix IV devices provide up to 104 distinct clock domains (16 GCLKs + 88 RCLKs) in the entire device. You can use these clock resources to form the following types of clock regions:

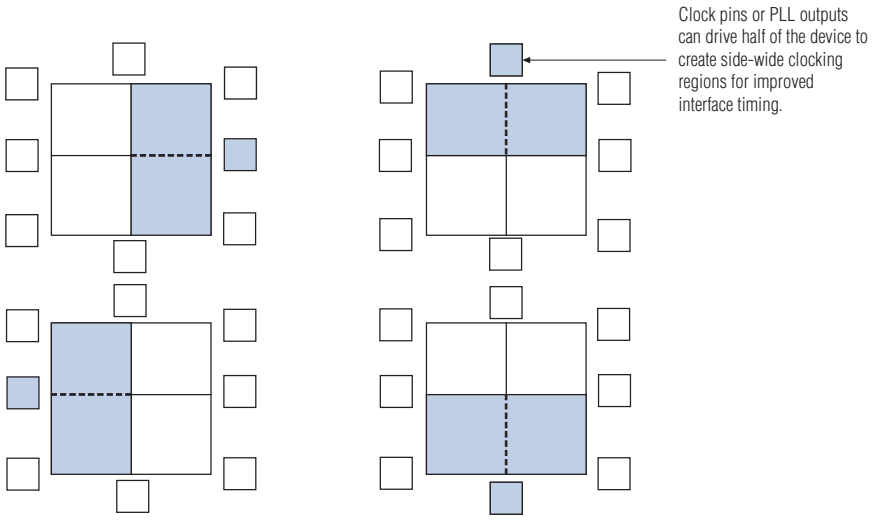
- Entire device
- Regional
- Dual-regional

To form the entire device clock region, a source (not necessarily a clock signal) drives a GCLK network that can be routed through the entire device. This clock region has the maximum delay when compared with other clock regions, but allows the signal to reach every destination within the device. This is a good option for routing global reset and clear signals or routing clocks throughout the device.

To form a RCLK region, a source drives a single quadrant of the device. This clock region provides the lowest skew within a quadrant and is a good option if all the destinations are within a single device quadrant.

To form a dual-regional clock region, a single source (a clock pin or PLL output) generates a dual-regional clock by driving two RCLK networks (one from each quadrant). This technique allows destinations across two device quadrants to use the same low-skew clock. The routing of this signal on an entire side has approximately the same delay as a RCLK region. Internal logic can also drive a dual-regional clock network. Corner PLL outputs only span one quadrant, they cannot generate a dual-regional clock network. Figure 5–10 shows the dual-regional clock region.

Figure 5–10. Stratix IV Dual-Regional Clock Region



Clock Network Sources

In Stratix IV devices, clock input pins, PLL outputs, and internal logic can drive the GCLK and RCLK networks. For connectivity between dedicated pins `CLK[0..15]` and the GCLK and RCLK networks, refer to Table 5–2 and Table 5–3 on page 5–11.

Dedicated Clock Input Pins

Clock pins can be either differential clocks or single-ended clocks. Stratix IV devices support 16 differential clock inputs or 32 single-ended clock inputs. You can also use dedicated clock input pins `CLK[15..0]` for high fan-out control signals such as asynchronous clears, presets, and clock enables for protocol signals such as `TRDY` and `IRDY` for PCIe through the GCLK or RCLK networks.

LABs

You can drive each GCLK and RCLK network using LAB-routing to enable internal logic to drive a high fan-out, low-skew signal.

☞ Stratix IV PLLs cannot be driven by internally generated GCLKs or RCLKs. The input clock to the PLL has to come from dedicated clock input pins or pin/PLL-fed GCLKs or RCLKs.

PLL Clock Outputs

Stratix IV PLLs can drive both GCLK and RCLK networks, as described in Table 5–5 on page 5–13 and Table 5–6 on page 5–13.

Table 5–2 lists the connection between the dedicated clock input pins and GCLKs.

Table 5–2. Clock Input Pin Connectivity to the GCLK Networks

Clock Resources	CLK (p/n Pins)															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
GCLK0	Y	Y	Y	Y	—	—	—	—	—	—	—	—	—	—	—	—
GCLK1	Y	Y	Y	Y	—	—	—	—	—	—	—	—	—	—	—	—
GCLK2	Y	Y	Y	Y	—	—	—	—	—	—	—	—	—	—	—	—
GCLK3	Y	Y	Y	Y	—	—	—	—	—	—	—	—	—	—	—	—
GCLK4	—	—	—	—	Y	Y	Y	Y	—	—	—	—	—	—	—	—
GCLK5	—	—	—	—	Y	Y	Y	Y	—	—	—	—	—	—	—	—
GCLK6	—	—	—	—	Y	Y	Y	Y	—	—	—	—	—	—	—	—
GCLK7	—	—	—	—	Y	Y	Y	Y	—	—	—	—	—	—	—	—
GCLK8	—	—	—	—	—	—	—	—	Y	Y	Y	Y	—	—	—	—
GCLK9	—	—	—	—	—	—	—	—	Y	Y	Y	Y	—	—	—	—
GCLK10	—	—	—	—	—	—	—	—	Y	Y	Y	Y	—	—	—	—
GCLK11	—	—	—	—	—	—	—	—	Y	Y	Y	Y	—	—	—	—
GCLK12	—	—	—	—	—	—	—	—	—	—	—	—	Y	Y	Y	Y
GCLK13	—	—	—	—	—	—	—	—	—	—	—	—	Y	Y	Y	Y
GCLK14	—	—	—	—	—	—	—	—	—	—	—	—	Y	Y	Y	Y
GCLK15	—	—	—	—	—	—	—	—	—	—	—	—	Y	Y	Y	Y

Table 5–3 lists the connectivity between the dedicated clock input pins and RCLKs in Stratix IV devices. A given clock input pin can drive two adjacent RCLK networks to create a dual-regional clock network.

Table 5–3. Clock Input Pin Connectivity to the RCLK Networks (Part 1 of 2)

Clock Resource	CLK (p/n Pins)															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
RCLK [0, 4, 6, 10]	Y	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RCLK [1, 5, 7, 11]	—	Y	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RCLK [2, 8]	—	—	Y	—	—	—	—	—	—	—	—	—	—	—	—	—
RCLK [3, 9]	—	—	—	Y	—	—	—	—	—	—	—	—	—	—	—	—
RCLK [13, 17, 21, 23, 27, 31]	—	—	—	—	Y	—	—	—	—	—	—	—	—	—	—	—
RCLK [12, 16, 20, 22, 26, 30]	—	—	—	—	—	Y	—	—	—	—	—	—	—	—	—	—
RCLK [15, 19, 25, 29]	—	—	—	—	—	—	Y	—	—	—	—	—	—	—	—	—
RCLK [14, 18, 24, 28]	—	—	—	—	—	—	—	Y	—	—	—	—	—	—	—	—
RCLK [35, 41]	—	—	—	—	—	—	—	—	Y	—	—	—	—	—	—	—

Table 5–3. Clock Input Pin Connectivity to the RCLK Networks (Part 2 of 2)

Clock Resource	CLK (p/n Pins)															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
RCLK [34, 40]	—	—	—	—	—	—	—	—	—	Y	—	—	—	—	—	—
RCLK [33, 37, 39, 43]	—	—	—	—	—	—	—	—	—	—	Y	—	—	—	—	—
RCLK [32, 36, 38, 42]	—	—	—	—	—	—	—	—	—	—	—	Y	—	—	—	—
RCLK [47, 51, 57, 61]	—	—	—	—	—	—	—	—	—	—	—	—	Y	—	—	—
RCLK [46, 50, 56, 60]	—	—	—	—	—	—	—	—	—	—	—	—	—	Y	—	—
RCLK [45, 49, 53, 55, 59, 63]	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Y	—
RCLK [44, 48, 52, 54, 58, 62]	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Y

Clock Input Connections to the PLLs

Table 5–4 lists the dedicated clock input pin connectivity to Stratix IV PLLs.

Table 5–4. Device PLLs and PLL Clock Pin Drivers (1), (2)

Dedicated Clock Input Pin CLK (p/n Pins)	PLL Number											
	L1 (3)	L2	L3	L4 (3)	B1	B2	R1 (3)	R2	R3	R4 (3)	T1	T2
CLK0	Y	Y	Y	Y	—	—	—	—	—	—	—	—
CLK1	Y	Y	Y	Y	—	—	—	—	—	—	—	—
CLK2	Y	Y	Y	Y	—	—	—	—	—	—	—	—
CLK3	Y	Y	Y	Y	—	—	—	—	—	—	—	—
CLK4	—	—	—	—	Y	Y	—	—	—	—	—	—
CLK5	—	—	—	—	Y	Y	—	—	—	—	—	—
CLK6	—	—	—	—	Y	Y	—	—	—	—	—	—
CLK7	—	—	—	—	Y	Y	—	—	—	—	—	—
CLK8	—	—	—	—	—	—	Y	Y	Y	Y	—	—
CLK9	—	—	—	—	—	—	Y	Y	Y	Y	—	—
CLK10	—	—	—	—	—	—	Y	Y	Y	Y	—	—
CLK11	—	—	—	—	—	—	Y	Y	Y	Y	—	—
CLK12	—	—	—	—	—	—	—	—	—	—	Y	Y
CLK13	—	—	—	—	—	—	—	—	—	—	Y	Y
CLK14	—	—	—	—	—	—	—	—	—	—	Y	Y
CLK15	—	—	—	—	—	—	—	—	—	—	Y	Y

Notes to Table 5–4:

(1) For single-ended clock inputs, only the CLK<#>p pin has a dedicated connection to the PLL. If you use the CLK<#>n pin, a global clock is used.

(2) For the availability of the clock input pins in each device density, refer to the "Stratix IV Device Pin-Out Files" section of the Pin-Out Files for Altera Devices site.

(3) These are non-compensated clock input paths. For the compensated input for these PLLs, use the corresponding PLL_[L, R][1,4]_CLK input pin.

Dedicated clock pins can drive PLLs over dedicated routing; they do not require the global or regional network. Compensated inputs, which are a subset of dedicated clock pins, drive PLLs that can only compensate the input delay when a dedicated clock pin is in the same I/O bank as the PLL used.

Clock Output Connections

PLLs in Stratix IV devices can drive up to 20 RCLK networks and four GCLK networks. For Stratix IV PLL connectivity to GCLK networks, refer to Table 5–5. The Quartus II software automatically assigns PLL clock outputs to RCLK and GCLK networks.

Table 5–5 lists how the PLL clock outputs connect to the GCLK networks.

Table 5–5. Stratix IV PLL Connectivity to the GCLK Networks (1)

Clock Network	PLL Number											
	L1	L2	L3	L4	B1	B2	R1	R2	R3	R4	T1	T2
GCLK0	Y	Y	Y	Y	—	—	—	—	—	—	—	—
GCLK1	Y	Y	Y	Y	—	—	—	—	—	—	—	—
GCLK2	Y	Y	Y	Y	—	—	—	—	—	—	—	—
GCLK3	Y	Y	Y	Y	—	—	—	—	—	—	—	—
GCLK4	—	—	—	—	Y	Y	—	—	—	—	—	—
GCLK5	—	—	—	—	Y	Y	—	—	—	—	—	—
GCLK6	—	—	—	—	Y	Y	—	—	—	—	—	—
GCLK7	—	—	—	—	Y	Y	—	—	—	—	—	—
GCLK8	—	—	—	—	—	—	Y	Y	Y	Y	—	—
GCLK9	—	—	—	—	—	—	Y	Y	Y	Y	—	—
GCLK10	—	—	—	—	—	—	Y	Y	Y	Y	—	—
GCLK11	—	—	—	—	—	—	Y	Y	Y	Y	—	—
GCLK12	—	—	—	—	—	—	—	—	—	—	Y	Y
GCLK13	—	—	—	—	—	—	—	—	—	—	Y	Y
GCLK14	—	—	—	—	—	—	—	—	—	—	Y	Y
GCLK15	—	—	—	—	—	—	—	—	—	—	Y	Y

Note to Table 5–5:

(1) Only PLL counter outputs C0 - C3 can drive the GCLK networks.

Table 5–6 lists how the PLL clock outputs connect to the RCLK networks.

Table 5–6. Stratix IV RCLK Outputs From the PLL Clock Outputs (1) (Part 1 of 2)

Clock Resource	PLL Number											
	L1	L2	L3	L4	B1	B2	R1	R2	R3	R4	T1	T2
RCLK[0..11]	—	Y	Y	—	—	—	—	—	—	—	—	—
RCLK[12..31]	—	—	—	—	Y	Y	—	—	—	—	—	—
RCLK[32..43]	—	—	—	—	—	—	—	Y	Y	—	—	—
RCLK[44..63]	—	—	—	—	—	—	—	—	—	—	Y	Y

Table 5–6. Stratix IV RCLK Outputs From the PLL Clock Outputs (1) (Part 2 of 2)

Clock Resource	PLL Number											
	L1	L2	L3	L4	B1	B2	R1	R2	R3	R4	T1	T2
RCLK[64..69]	—	—	—	Y	—	—	—	—	—	—	—	—
RCLK[70..75]	—	—	—	—	—	—	—	—	—	Y	—	—
RCLK[76..81]	—	—	—	—	—	—	Y	—	—	—	—	—
RCLK[82..87]	Y	—	—	—	—	—	—	—	—	—	—	—

Note to Table 5–6:

(1) All PLL counter outputs can drive the RCLK networks.

Clock Control Block

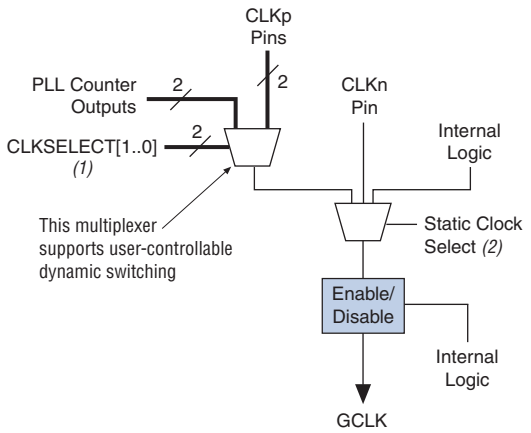
Every GCLK and RCLK network has its own clock control block. The control block provides the following features:

- Clock source selection (dynamic selection for GCLKs)
- Global clock multiplexing
- Clock power down (static or dynamic clock enable or disable)

Figure 5–11 and Figure 5–12 show the GCLK and RCLK select blocks, respectively.

You can select the clock source for the GCLK select block either statically or dynamically. You can statically select the clock source using a setting in the Quartus II software or you can dynamically select the clock source using internal logic to drive the multiplexer-select inputs. When selecting the clock source dynamically, you can select either PLL outputs (such as C0 or C1) or a combination of clock pins or PLL outputs.

Figure 5–11. Stratix IV GCLK Control Block



Notes to Figure 5–11:

(1) When the device is operating in user mode, you can dynamically control the clock select signals through internal logic.

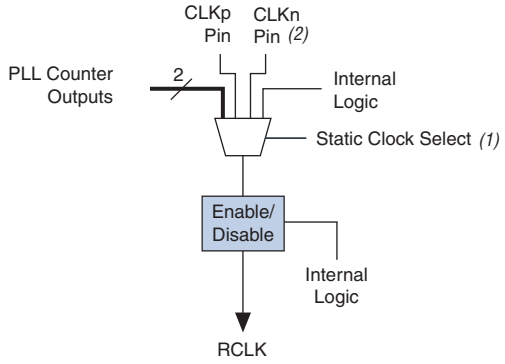
(2) When the device is operation in user mode, you can only set the clock select signals through a configuration file (SRAM object file [**.sof**] or programmer object file [**.pof**]) and cannot be dynamically controlled.

The mapping between the input clock pins, PLL counter outputs, and clock control block inputs is as follows:

- `inclk[0]` and `inclk[1]`—can be fed by any of the four dedicated clock pins on the same side of the Stratix IV device
- `inclk[2]`—can be fed by PLL counters C0 and C2 from the two center PLLs on the same side of the Stratix IV device
- `inclk[3]`—can be fed by PLL counters C1 and C3 from the two center PLLs on the same side of the Stratix IV device

The corner PLLs (L1, L4, R1, and R4) and the corresponding clock input pins (`PLL_L1_CLK` and so forth) do not support dynamic selection for the GCLK network. The clock source selection for the GCLK and RCLK networks from the corner PLLs (L1, L4, R1, and R4) and the corresponding clock input pins (`PLL_L1_CLK` and so forth) are controlled statically using configuration bit settings in the configuration file (**.sof** or **.pof**) generated by the Quartus II software.

Figure 5–12. RCLK Control Block



Notes to Figure 5–12:

(1) When the device is operation in user mode, you can only set the clock select signals through a configuration file (**.sof** or **.pof**) and cannot be dynamically controlled.

(2) The `CLKn` pin is not a dedicated clock input when used as a single-ended PLL clock input.

You can only control the clock source selection for the RCLK select block statically using configuration bit settings in the configuration file (**.sof** or **.pof**) generated by the Quartus II software.

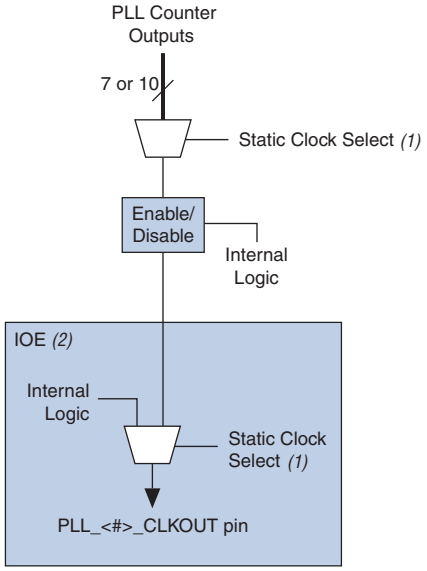
You can power down the Stratix IV clock networks using both static and dynamic approaches. When a clock network is powered down, all the logic fed by the clock network is in off-state, thereby reducing the overall power consumption of the device. The unused GCLK and RCLK networks are automatically powered down through configuration bit settings in the configuration file (**.sof** or **.pof**) generated by the Quartus II software. The dynamic clock enable or disable feature allows the internal logic to control power-up or power-down synchronously on the GCLK and RCLK networks, including dual-regional clock regions. This function is independent of the PLL and is applied directly on the clock network, as shown in Figure 5–11 and Figure 5–12.

You can set the input clock sources and the clkena signals for the GCLK and RCLK network multiplexers through the Quartus II software using the ALTCLKCTRL megafunction. You can also enable or disable the dedicated external clock output pins using the ALTCLKCTRL megafunction. Figure 5–13 shows the external PLL output clock control block.

When using the ALTCLKCTRL megafunction to implement dynamic clock source selection, the inputs from the clock pins feed the inclk[0..1] ports of the multiplexer, while the PLL outputs feed the inclk[2..3] ports. You can choose from among these inputs using the CLKSELECT[1..0] signal.

For more information, refer to the *Clock Control Block (ALTCLKCTRL) Megafunction User Guide*.

Figure 5–13. Stratix IV External PLL Output Clock Control Block



Notes to Figure 5–13:

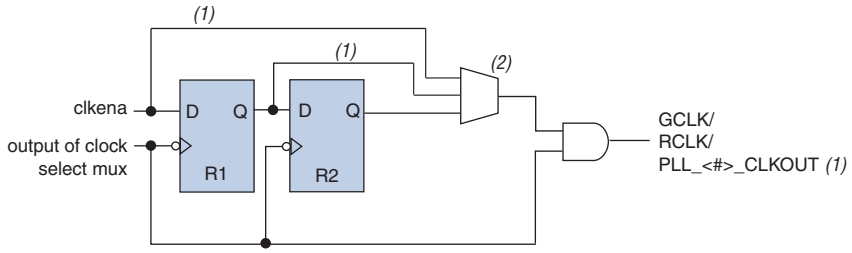
(1) When the device is operation in user mode, you can only set the clock select signals through a configuration file (**.sof** or **.pof**) and cannot be dynamically controlled.

(2) The clock control block feeds to a multiplexer within the PLL_<#>_CLKOUT pin's IOE. The PLL_<#>_CLKOUT pin is a dual-purpose pin. Therefore, this multiplexer selects either an internal signal or the output of the clock control block.

Clock Enable Signals

Figure 5–14 shows how the clock enable and disable circuit of the clock control block is implemented in Stratix IV devices.

Figure 5–14. clkena Implementation



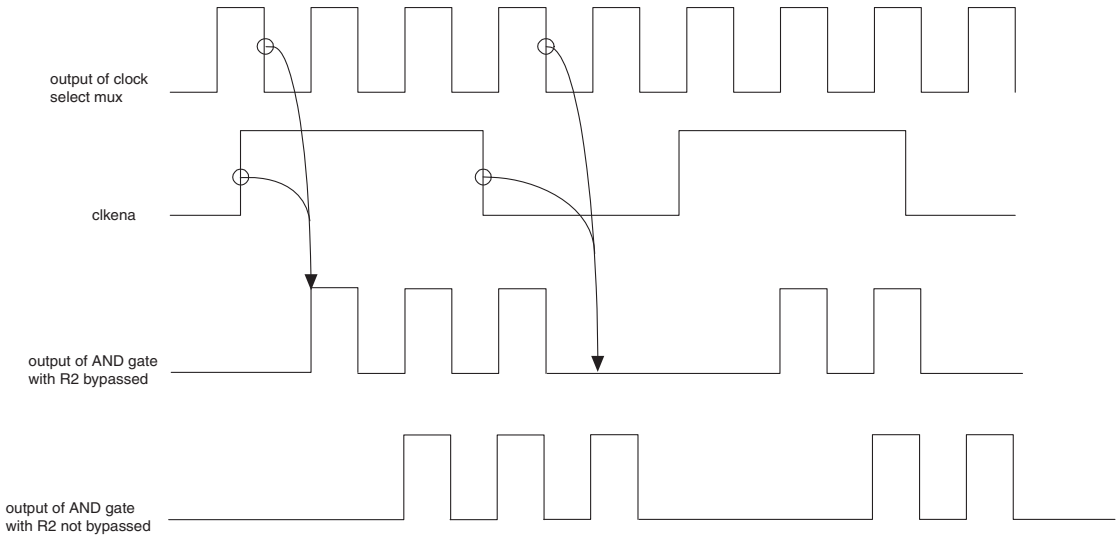
Notes to Figure 5–14:

(1) The R1 and R2 bypass paths are not available for the PLL external clock outputs.
(2) The select line is statically controlled by a bit setting in the configuration file (**.sof** or **.pof**).

In Stratix IV devices, the `clkena` signals are supported at the clock network level instead of at the PLL output counter level. This allows you to gate off the clock even when you are not using a PLL. You can also use the `clkena` signals to control the dedicated external clocks from the PLLs. Figure 5–15 shows a waveform example for a clock output enable. `clkena` is synchronous to the falling edge of the clock output.

Stratix IV devices also have an additional metastability register that aids in asynchronous enable and disable of the GCLK and RCLK networks. You can optionally bypass this register in the Quartus II software.

Figure 5–15. clkena Signals (1)



Note to Figure 5–15:

(1) You can use the `clkena` signals to enable or disable the GCLK and RCLK networks or the `PLL_<#>_CLKOUT` pins.

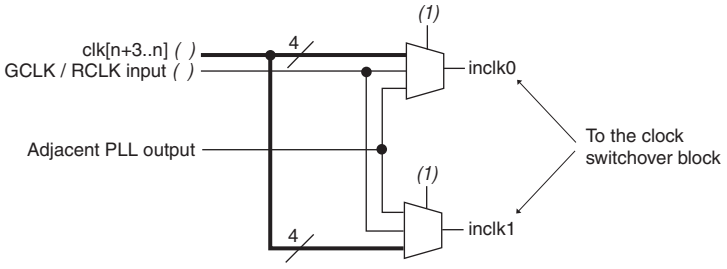
The PLL can remain locked independent of the clkena signals because the loop-related counters are not affected. This feature is useful for applications that require a low-power or sleep mode. The clkena signal can also disable clock outputs if the system is not tolerant of frequency over-shoot during resynchronization.

Clock Source Control for PLLs

The clock input to Stratix IV PLLs comes from clock input multiplexers. The clock multiplexer inputs come from dedicated clock input pins, PLLs through the GCLK and RCLK networks, or from dedicated connections between adjacent top/bottom and left/right PLLs. The clock input sources to top/bottom and left/right PLLs (L2, L3, T1, T2, B1, B2, R2, and R3) are shown in Figure 5–16; the corresponding clock input sources to left and right PLLs (L1, L4, R1, and R4) are shown in Figure 5–17.

The multiplexer select lines are only set in the configuration file (**.sof** or **.pof**). After programmed, this block cannot be changed without loading a new configuration file (**.sof** or **.pof**). The Quartus II software automatically sets the multiplexer select signals depending on the clock sources selected in the design.

Figure 5–16. Clock Input Multiplexer Logic for L2, L3, T1, T2, B1, B2, R2, and R3 PLLs



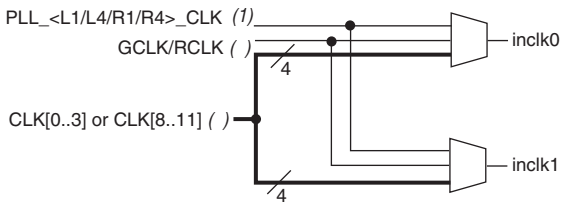
Notes to Figure 5–16:

(1) When the device is operating in user mode, input clock multiplexing is controlled through a configuration file (**.sof** or **.pof**) only and cannot be dynamically controlled.

(2) n=0 for L2 and L3 PLLs; n=4 for B1 and B2 PLLs; n=8 for R2 and R3 PLLs, and n=12 for T1 and T2 PLLs.

(3) You can drive the GCLK or RCLK input using an output from another PLL, a pin-driven GCLK or RCLK, or through a clock control block provided the clock control block is fed by an output from another PLL or a pin-driven dedicated GCLK or RCLK. An internally generated global signal or general purpose I/O pin cannot drive the PLL.

Figure 5–17. Clock Input Multiplexer Logic for L1, L4, R1, and R4 PLLs



Notes to Figure 5–17:

(1) Dedicated clock input pins to the PLLs are L1, L4, R1, and R4, respectively. For example, PLL_L1_CLK is the dedicated clock input for PLL_L1.

(2) You can drive the GCLK or RCLK input using an output from another PLL, a pin-driven GCLK or RCLK, or through a clock control block provided the clock control block is fed by an output from another PLL or a pin-driven dedicated GCLK or RCLK. An internally generated global signal or general purpose I/O pin cannot drive the PLL.

(3) The center clock pins can feed the corner PLLs on the same side directly through a dedicated path. However, these paths may not be fully compensated.

Cascading PLLs

You can cascade the left/right and top/bottom PLLs through the GCLK and RCLK networks. In addition, where two left/right or top/bottom PLLs exist next to each other, there is a direct connection between them that does not require the GCLK or RCLK network. Using this path reduces clock jitter when cascading PLLs.

Stratix IV GX devices allow cascading the left and right PLLs to transceiver PLLs (CMU PLLs and receiver CDRs).

For more information, refer to the "FPGA Fabric PLLs -Transceiver PLLs Cascading" section in the *Transceiver Clocking in Stratix IV Devices* chapter.

When cascading PLLs in Stratix IV devices, the source (upstream) PLL must have a low-bandwidth setting while the destination (downstream) PLL must have a high-bandwidth setting. Ensure that there is no overlap of the bandwidth ranges of the two PLLs.

For more information about PLL cascading in external memory interfaces designs, refer to the *External Memory PHY Interface (ALTMEMPHY) (nonAFI) Megafunction User Guide*.

PLLs in Stratix IV Devices

Stratix IV devices offer up to 12 PLLs that provide robust clock management and synthesis for device clock management, external system clock management, and high-speed I/O interfaces. The nomenclature for the PLLs follows their geographical location in the device floor plan. The PLLs that reside on the top and bottom sides of the device are named `PLL_T1`, `PLL_T2`, `PLL_B1` and `PLL_B2`; the PLLs that reside on the left and right sides of the device are named `PLL_L1`, `PLL_L2`, `PLL_L3`, `PLL_L4`, `PLL_R1`, `PLL_R2`, `PLL_R3`, and `PLL_R4`.

Table 5–7 lists the number of PLLs available in the Stratix IV device family.

Table 5–7. PLL Availability for Stratix IV Devices (Part 1 of 2)

Device	Package	L1	L2	L3	L4	T1	T2	B1	B2	R1	R2	R3	R4
EP4S40G2	F1517	—	Y	Y	—	Y	Y	Y	Y	—	Y	Y	—
EP4S40G5	H1517	—	Y	Y	—	Y	Y	Y	Y	—	Y	Y	—
EP4S100G2	F1517	—	Y	Y	—	Y	Y	Y	Y	—	Y	Y	—
EP4S100G3	F1932	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
EP4S100G4	F1932	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
EP4S100G5	H1517	—	Y	Y	—	Y	Y	Y	Y	—	Y	Y	—
	F1932	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
EP4SE230	F780	—	Y	—	—	Y	—	Y	—	—	Y	—	—
EP4SE360	H780	—	Y	—	—	Y	—	Y	—	—	Y	—	—
	F1152	—	Y	Y	—	Y	Y	Y	Y	—	Y	Y	—

Table 5–7. PLL Availability for Stratix IV Devices (Part 2 of 2)

Device	Package	L1	L2	L3	L4	T1	T2	B1	B2	R1	R2	R3	R4
EP4SE530	H1152	—	Y	Y	—	Y	Y	Y	Y	—	Y	Y	—
	H1517	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
	F1760	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
EP4SE820	H1152	—	Y	Y	—	Y	Y	Y	Y	—	Y	Y	—
	H1517	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
	F1760	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
EP4SGX70	F780	—	Y	—	—	Y	—	Y	—	—	—	—	—
	F1152	—	Y	—	—	Y	—	Y	—	—	Y	—	—
EP4SGX110	F780	—	Y	—	—	Y	—	Y	—	—	—	—	—
	F1152	—	Y	—	—	Y	—	Y	—	—	Y	—	—
EP4SGX180	F780	—	Y	—	—	Y	—	Y	—	—	—	—	—
	F1152	—	Y	—	—	Y	Y	Y	Y	—	Y	—	—
	F1517	—	Y	Y	—	Y	Y	Y	Y	—	Y	Y	—
EP4SGX230	F780	—	Y	—	—	Y	—	Y	—	—	—	—	—
	F1152	—	Y	—	—	Y	Y	Y	Y	—	Y	—	—
	F1517	—	Y	Y	—	Y	Y	Y	Y	—	Y	Y	—
EP4SGX290	H780	—	—	—	—	Y	Y	Y	Y	—	—	—	—
	F1152	—	Y	—	—	Y	Y	Y	Y	—	Y	—	—
	F1517	—	Y	Y	—	Y	Y	Y	Y	—	Y	Y	—
	F1760	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
	F1932	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
EP4SGX360	H780	—	—	—	—	Y	Y	Y	Y	—	—	—	—
	F1152	—	Y	—	—	Y	Y	Y	Y	—	Y	—	—
	F1517	—	Y	Y	—	Y	Y	Y	Y	—	Y	Y	—
	F1760	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
	F1932	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
EP4SGX530	H1152	—	Y	—	—	Y	Y	Y	Y	—	Y	—	—
	H1517	—	Y	Y	—	Y	Y	Y	Y	—	Y	Y	—
	F1760	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
	F1932	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

All Stratix IV PLLs have the same core analog structure with only minor differences in the features that are supported. Table 5–8 lists the features of top/bottom and left/right PLLs in Stratix IV devices.

Table 5–8. PLL Features in Stratix IV Devices (Part 1 of 2) (1)

Feature	Stratix IV Top/Bottom PLLs	Stratix IV Left/Right PLLs
C (output) counters	10	7
M, N, C counter sizes	1 to 512	1 to 512
Dedicated clock outputs	6 single-ended or 4 single-ended and 1 differential pair	2 single-ended or 1 differential pair

Table 5–8. PLL Features in Stratix IV Devices (Part 2 of 2) (1)

Feature	Stratix IV Top/Bottom PLLs	Stratix IV Left/Right PLLs
Clock input pins (2)	4 single-ended or 4 differential pin pairs	4 single-ended or 4 differential pin pairs
External feedback input pin	Single-ended or differential	Single-ended only
Spread-spectrum input clock tracking	Yes (3)	Yes (3)
PLL cascading	Through GCLK and RCLK and a dedicated path between adjacent PLLs	Through GCLK and RCLK and dedicated path between adjacent PLLs (4)
Compensation modes	All except LVDS clock network compensation	All except external feedback mode when using differential I/Os
PLL drives LVDSCLK and LOADEN	No	Yes
VCO output drives the DPA clock	No	Yes
Phase shift resolution	Down to 96.125 ps (5)	Down to 96.125 ps (5)
Programmable duty cycle	Yes	Yes
Output counter cascading	Yes	Yes
Input clock switchover	Yes	Yes

Notes to Table 5–8:

(1) While there is pin compatibility, there is no hard IP block placement compatibility.

(2) General purpose I/O pins cannot drive the PLL clock input pins.

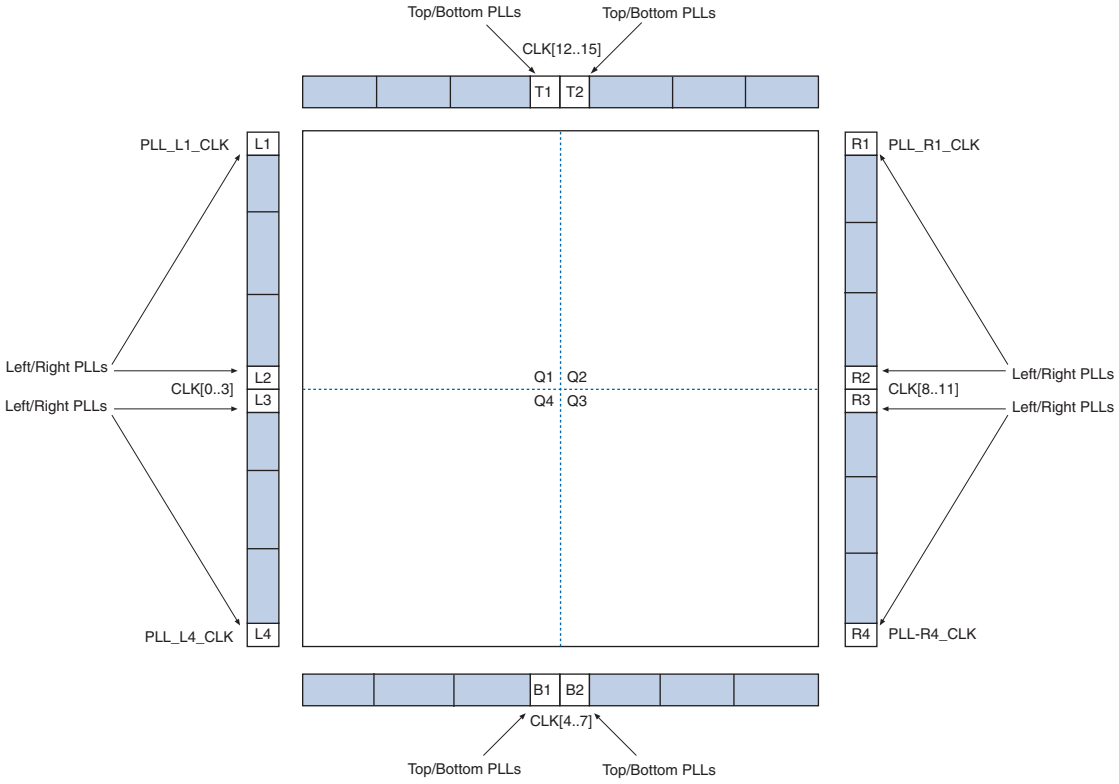
(3) Provided input clock jitter is within input jitter tolerance specifications.

(4) The dedicated path between adjacent PLLs is not available on L1, L4, R1, and R4 PLLs.

(5) The smallest phase shift is determined by the voltage-controlled oscillator (VCO) period divided by eight. For degree increments, the Stratix IV device can shift all output frequencies in increments of at least 45°. Smaller degree increments are possible depending on the frequency and divide parameters.

Figure 5–18 shows the location of PLLs in Stratix IV devices.

Figure 5–18. PLL Locations in Stratix IV Devices



Stratix IV PLL Hardware Overview

Stratix IV devices contain up to 12 PLLs with advanced clock management features. The goal of a PLL is to synchronize the phase and frequency of an internal or external clock to an input reference clock. There are a number of components that comprise a PLL to achieve this phase alignment.

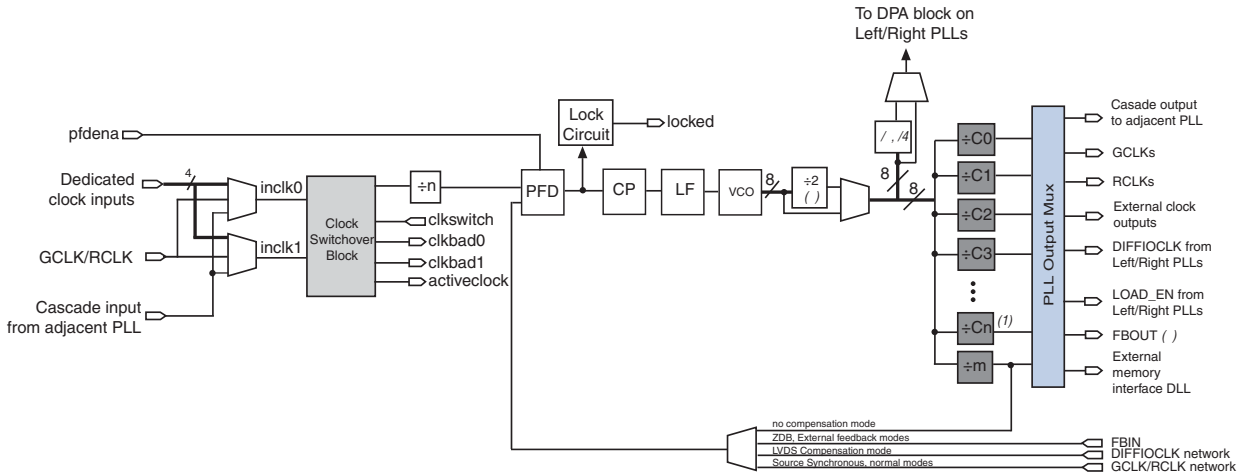
Stratix IV PLLs align the rising edge of the input reference clock to a feedback clock using the phase-frequency detector (PFD). The falling edges are determined by the duty-cycle specifications. The PFD produces an up or down signal that determines whether the VCO must operate at a higher or lower frequency. The output of the PFD feeds the charge pump and loop filter, which produces a control voltage for setting the VCO frequency. If the PFD produces an up signal, the VCO frequency increases. A down signal decreases the VCO frequency. 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 the charge pump 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 filters the jitter on the VCO. The voltage from the loop filter determines how fast the VCO operates. A divide counter (`m`) is inserted in the feedback loop to increase the VCO frequency above the input reference frequency. VCO frequency (f_{VCO}) is equal to (`m`) times the input reference clock (f_{REF}). The input reference clock (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}) 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 from the left and right PLLs can feed seven post-scale counters (`C[0..6]`), while the corresponding VCO output from the top and bottom PLLs can feed ten post-scale counters (`C[0..9]`). These post-scale counters allow a number of harmonically related frequencies to be produced by the PLL.

Figure 5–19 shows a simplified block diagram of the major components of the Stratix IV PLL.

Figure 5–19. Stratix IV PLL Block Diagram



Notes to Figure 5–19:

(1) The number of post-scale counters is seven for left and right PLLs and ten for top and bottom PLLs.
(2) This is the VCO post-scale counter K.
(3) The FBOUT port is fed by the M counter in Stratix IV PLLs.

You can drive the GCLK or RCLK inputs using an output from another PLL, a pin-driven GCLK or RCLK, or through a clock control block provided the clock control block is fed by an output from another PLL or a pin-driven dedicated GCLK or RCLK. An internally generated global signal or general purpose I/O pin cannot drive the PLL.

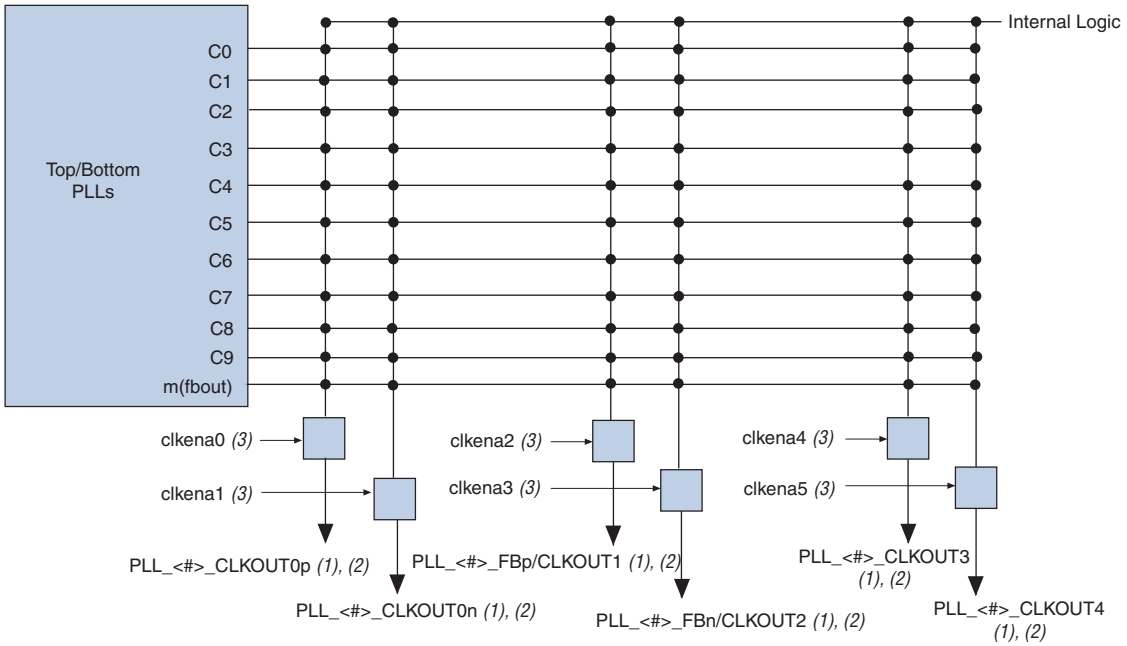
PLL Clock I/O Pins

Each top and bottom PLL supports six clock I/O pins, organized as three pairs of pins:

- 1st pair—two single-ended I/O or one differential I/O
- 2nd pair—two single-ended I/O or one differential external feedback input (FBp/FBn)
- 3rd pair—two single-ended I/O or one differential input

Figure 5–20 shows the clock I/O pins associated with the top and bottom PLLs.

Figure 5–20. External Clock Outputs for Top and Bottom PLLs



Notes to Figure 5–20:

(1) You can feed these clock output pins using any one of the `C[9..0]`, m counters.

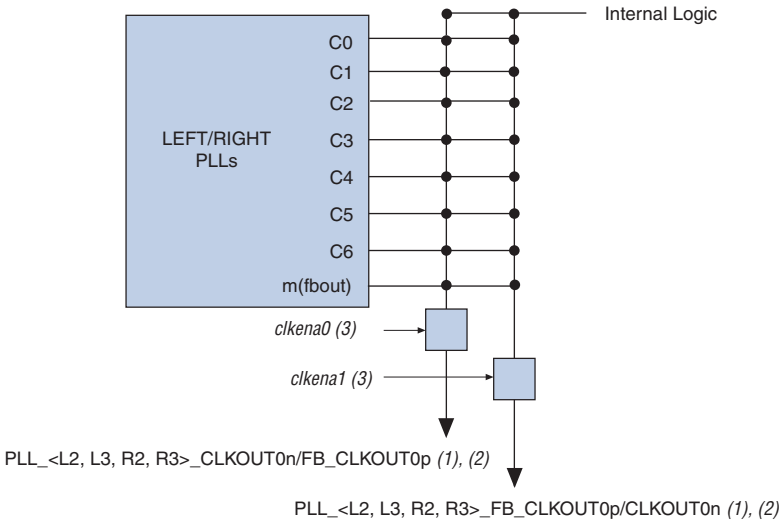
(2) The `CLKOUT0p` and `CLKOUT0n` pins can be either single-ended or differential clock outputs. The `CLKOUT1` and `CLKOUT2` pins are dual-purpose I/O pins that you can use as two single-ended outputs or one differential external feedback input pin. The `CLKOUT3` and `CLKOUT4` pins are two single-ended output pins.

(3) These external clock enable signals are available only when using the ALTCLKCTRL megafunction.

Any of the output counters (C[9..0] on the top and bottom PLLs and C[6..0] on the left and right PLLs) or the M counter can feed the dedicated external clock outputs, as shown in Figure 5–20 and Figure 5–21. Therefore, one counter or frequency can drive all output pins available from a given PLL.

Each left and right PLL supports two clock I/O pins, configured as either two single-ended I/Os or one differential I/O pair. When using both pins as single-ended I/Os, one of them can be the clock output while the other pin is the external feedback input (FB) pin. Therefore, for single-ended I/O standards, the left and right PLLs only support external feedback mode.

Figure 5–21. External Clock Outputs for Left and Right PLLs



Notes to Figure 5–21:

(1) You can feed these clock output pins using any one of the C[6..0], m counters.

(2) The CLKOUT0p and CLKOUT0n pins are dual-purpose I/O pins that you can use as two single-ended outputs or one single-ended output and one external feedback input pin.

(3) These external clock enable signals are available only when using the ALTCLKCTRL megafunction.

Each pin of a single-ended output pair can either be in-phase or 180° out-of-phase. The Quartus II software places the NOT gate in the design into the IOE to implement the 180° phase with respect to the other pin in the pair. The clock output pin pairs support the same I/O standards as standard output pins (in the top and bottom banks) as well as LVDS, LVPECL, differential High-Speed Transceiver Logic (HSTL), and differential SSTL.

To determine which I/O standards are supported by the PLL clock input and output pins, refer to the *I/O Features in Stratix IV Devices* chapter.

Stratix IV PLLs can also drive out to any regular I/O pin through the GCLK or RCLK network. You can also use the external clock output pins as user I/O pins if you do not need external PLL clocking.

PLL Control Signals

You can use the `pfdena`, `areset`, and `locked` signals to observe and control PLL operation and resynchronization.

pfdena

Use the `pfdena` signal to maintain the most recent locked frequency so your system has time to store its current settings before shutting down. The `pfdena` signal controls the PFD output with a programmable gate. If you disable PFD, the VCO operates at its most recent set value of control voltage and frequency, with some long-term drift to a lower frequency. The PLL continues running even if it goes out-of-lock or the input clock is disabled. You can use either your own control signal or the control signals available from the clock switchover circuit (`activeclock`, `clkbad[0]`, or `clkbad[1]`) to control `pfdena`.

areset

The `areset` signal is the reset or resynchronization input for each PLL. The device input pins or internal logic can drive these input signals. When `areset` is driven high, the PLL counters reset, clearing the PLL output and placing the PLL out-of-lock. The VCO is then set back to its nominal setting. When `areset` is driven low again, the PLL resynchronizes to its input as it re-locks.

You must assert the `areset` signal every time the PLL loses lock to guarantee the correct phase relationship between the PLL input and output clocks. You can set up the PLL to automatically reset (self reset) after a loss-of-lock condition using the Quartus II MegaWizard™ Plug-In Manager. You must include the `areset` signal in designs if either of the following conditions is true:

- PLL reconfiguration or clock switchover is enabled in the design
- Phase relationships between the PLL input and output clocks must be maintained after a loss-of-lock condition

If the input clock to the PLL is not toggling or is unstable after power up, assert the `areset` signal after the input clock is stable and within specifications.

locked

The `locked` signal output of the PLL indicates that the PLL has locked onto the reference clock and the PLL clock outputs are operating at the desired phase and frequency set in the Quartus II MegaWizard Plug-In Manager. The lock detection circuit provides a signal to the core logic that gives an indication when the feedback clock has locked onto the reference clock both in phase and frequency.

Altera recommends using the `areset` and `locked` signals in your designs to control and observe the status of your PLL.

Clock Feedback Modes

Stratix IV PLLs support up to six different clock feedback modes. Each mode allows clock multiplication and division, phase shifting, and programmable duty cycle. Table 5–9 lists the clock feedback modes supported by the Stratix IV device PLLs.

Table 5–9. Clock Feedback Mode Availability

Clock Feedback Mode	Availability	
	Top and Bottom PLLs	Left and Right PLLs
Source-synchronous	Yes	Yes
No-compensation	Yes	Yes
Normal	Yes	Yes
Zero-delay buffer (ZDB)	Yes	Yes
External feedback (1)	Yes	Yes (2)
LVDS compensation	No	Yes

Notes to Table 5–9:

(1) The high-bandwidth PLL setting is not supported in external feedback mode.

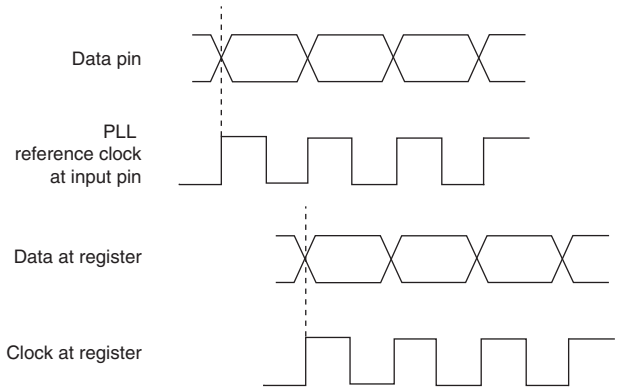
(2) External feedback mode is supported for single-ended inputs and outputs only on the left and right PLLs.

The input and output delays are fully compensated by a PLL only when using the dedicated clock input pins associated with a given PLL as the clock source. For example, when using PLL_T1 in normal mode, the clock delays from the input pin to the PLL clock output-to-destination register are fully compensated, provided the clock input pin is one of the following two pins: CLK14 and CLK15. Compensated pins are only in the same I/O bank as the PLL. When an RCLK or GCLK network drives the PLL, the input and output delays may not be fully compensated in the Quartus II software. Another example is when you configure PLL_T2 in zero-delay buffer mode and the PLL input is driven by a dedicated clock input pin, a fully compensated clock path results in zero-delay between the clock input and one of the output clocks from the PLL. If the PLL input is instead fed by a non-dedicated input (using the GCLK network), the output clock may not be perfectly aligned with the input clock.

Source Synchronous Mode

If data and clock arrive at the same time on the input pins, the same phase relationship is maintained at the clock and data ports of any IOE input register. Figure 5–22 shows an example waveform of the clock and data in this mode. Altera recommends source synchronous mode for source-synchronous data transfers. Data and clock signals at the IOE experience similar buffer delays as long as you use the same I/O standard.

Figure 5–22. Phase Relationship Between Clock and Data in Source-Synchronous Mode



Source-synchronous mode compensates for the delay of the clock network used plus any difference in the delay between these two paths:

- Data pin to the IOE register input
- Clock input pin to the PLL PFD input

The Stratix IV PLL can compensate multiple pad-to-input-register paths, such as a data bus when it is set to use source-synchronous compensation mode. You can use the "PLL Compensation" assignment in the Quartus II software Assignment Editor to select which input pins are used as the PLL compensation targets. You can include your entire data bus, provided the input registers are clocked by the same output of a source-synchronous-compensated PLL. In order for the clock delay to be properly compensated, all of the input pins must be on the same side of the device. The PLL compensates for the input pin with the longest pad-to-register delay among all input pins in the compensated bus.

If you do not make the "PLL Compensation" assignment, the Quartus II software automatically selects all of the pins driven by the compensated output of the PLL as the compensation target.

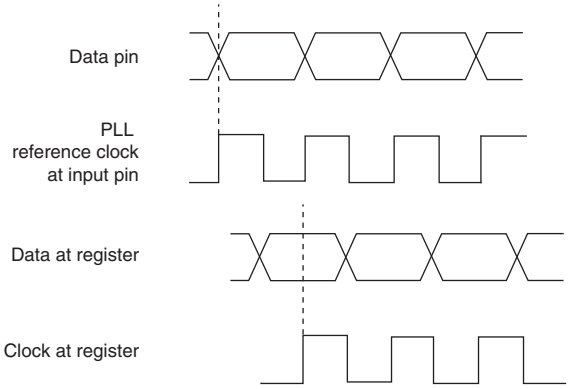
Source-Synchronous Mode for LVDS Compensation

The goal of source-synchronous mode is to maintain the same data and clock timing relationship seen at the pins of the internal serializer/deserializer (SERDES) capture register, except that the clock is inverted (180° phase shift). Thus, source-synchronous mode ideally compensates for the delay of the LVDS clock network plus any difference in delay between these two paths:

- Data pin-to-SERDES capture register
- Clock input pin-to-SERDES capture register. In addition, the output counter must provide the 180° phase shift

Figure 5–23 shows an example waveform of the clock and data in LVDS mode.

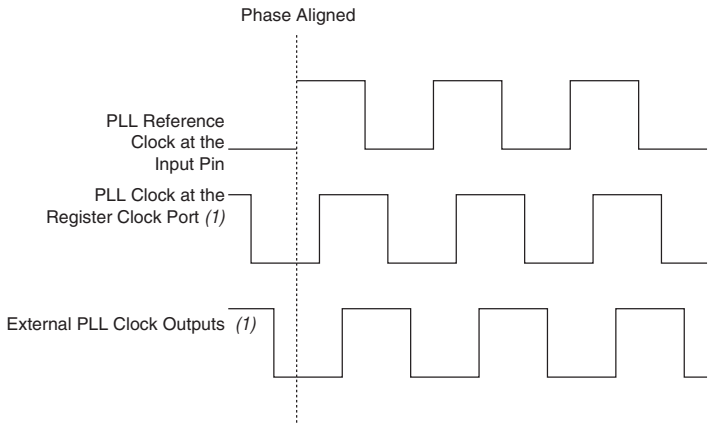
Figure 5–23. Phase Relationship Between the Clock and Data in LVDS Mode



No-Compensation Mode

In no-compensation mode, the PLL does not compensate for any clock networks. This mode provides better jitter performance because the clock feedback into the PFD passes through less circuitry. Both the PLL internal- and external-clock outputs are phase-shifted with respect to the PLL clock input. Figure 5–24 shows an example waveform of the PLL clocks' phase relationship in no-compensation mode.

Figure 5–24. Phase Relationship Between the PLL Clocks in No Compensation Mode



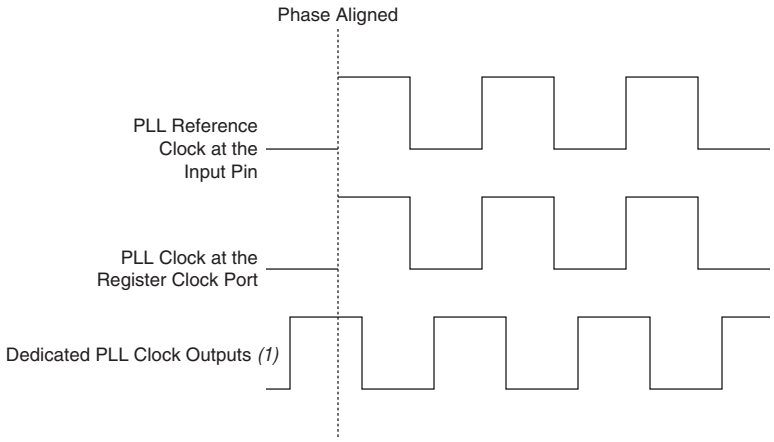
Note to Figure 5–24:

(1) The PLL clock outputs lag the PLL input clocks depending on routine delays.

Normal Mode

An internal clock in normal mode is phase-aligned to the input clock pin. The external clock-output pin has a phase delay relative to the clock input pin if connected in this mode. The Quartus II software timing analyzer reports any phase difference between the two. In normal mode, the delay introduced by the GCLK or RCLK network is fully compensated. Figure 5–25 shows an example waveform of the PLL clocks' phase relationship in normal mode.

Figure 5–25. Phase Relationship Between the PLL Clocks in Normal Mode



Note to Figure 5–25:

(1) The external clock output can lead or lag the PLL internal clock signals.

Zero-Delay Buffer (ZDB) Mode

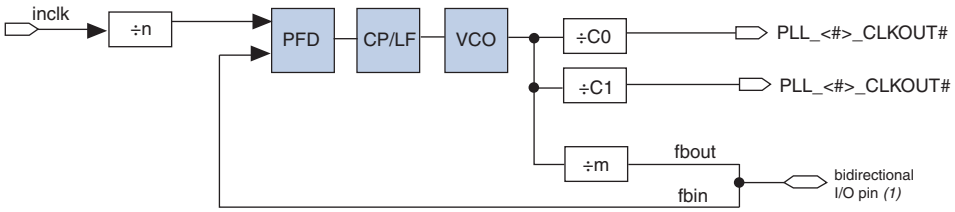
In ZDB mode, the external clock output pin is phase-aligned with the clock input pin for zero-delay through the device. When using this mode, you must use the same I/O standard on the input clocks and output clocks to guarantee clock alignment at the input and output pins. ZDB mode is supported on all Stratix IV PLLs.

When using Stratix IV PLLs in ZDB mode, along with single-ended I/O standards, to ensure phase alignment between the CLK pin and the external clock output (`CLKOUT`) pin, you must instantiate a bi-directional I/O pin in the design to serve as the feedback path connecting the `FBOUT` and `FBIN` ports of the PLL. The PLL uses this bi-directional I/O pin to mimic, and compensate for, the output delay from the clock output port of the PLL to the external clock output pin. Figure 5–26 shows ZDB mode in Stratix IV PLLs. When using ZDB mode, you cannot use differential I/O standards on the PLL clock input or output pins.

The bi-directional I/O pin that you instantiate in your design must always be assigned a single-ended I/O standard.

When using ZDB mode, to avoid signal reflection, do not place board traces on the bi-directional I/O pin.

Figure 5–26. ZDB Mode in Stratix IV PLLs

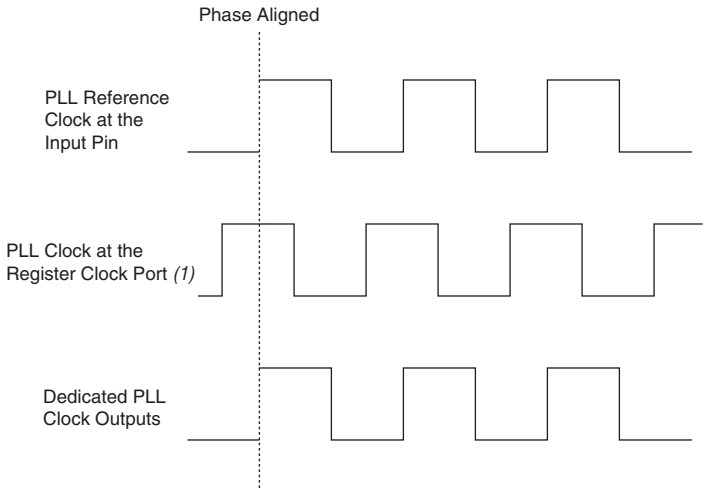


Note to Figure 5–26:

(1) The bidirectional I/O pin must be assigned to the `PLL_<#>_FB_CLKOUT0p` pin for left and right PLLs and to the `PLL_<#>_FBp_/CLKOUT1` pin for top and bottom PLLs.

Figure 5–27 shows an example waveform of the PLL clocks' phase relationship in ZDB mode.

Figure 5–27. Phase Relationship Between the PLL Clocks in ZDB Mode



Note to Figure 5–27:

(1) The internal PLL clock output can lead or lag the external PLL clock outputs.

External Feedback Mode

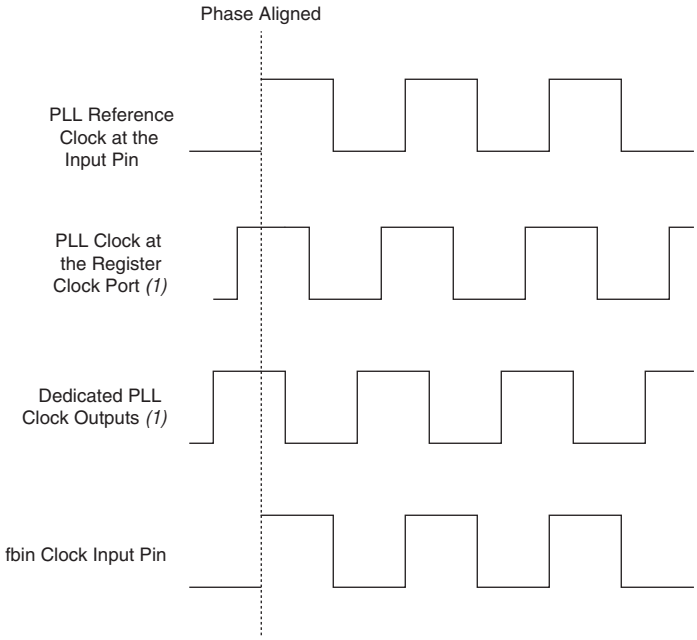
In external feedback mode, the external feedback input pin (`fbin`) is phase-aligned with the clock input pin, as shown in Figure 5–28. Aligning these clocks allows you to remove clock delay and skew between devices. This mode is supported on all Stratix IV PLLs.

In external feedback mode, the output of the `M` counter (`FBOUT`) feeds back to the PLL `fbin` input (using a trace on the board) becoming part of the feedback loop. Also, use one of the dual-purpose external clock outputs as the `fbin` input pin in this mode.

When using external feedback mode, you must use the same I/O standard on the input clock, feedback input, and output clocks. Left and right PLLs support this mode when using single-ended I/O standards only.

Figure 5–28 shows an example waveform of the phase relationship between the PLL clocks in external feedback mode.

Figure 5–28. Phase Relationship Between the PLL Clocks in External Feedback Mode

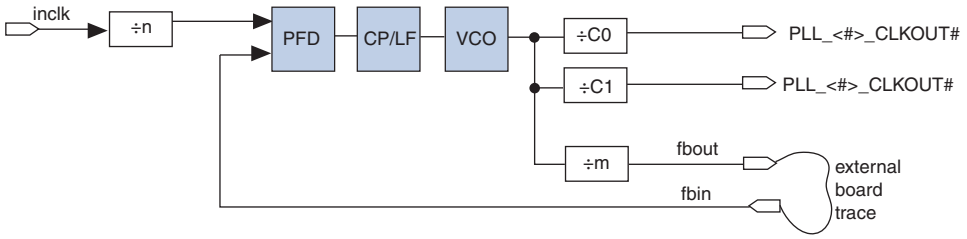


Note to Figure 5–28:

(1) The PLL clock outputs can lead or lag the `fbin` clock input.

Figure 5–29 shows external feedback mode implementation in Stratix IV devices.

Figure 5–29. External Feedback Mode in Stratix IV Devices



Clock Multiplication and Division

Each Stratix IV PLL provides clock synthesis for PLL output ports using *M*/(*N** post-scale counter) scaling factors. The input clock is divided by a pre-scale factor, `n`, and is then multiplied by the *m* feedback factor. The control loop drives the VCO to match f_{in} (*M*/*N*). Each output port has a unique post-scale counter that divides down the high-frequency VCO. For multiple PLL outputs with different frequencies, the VCO is set to the least common multiple of the output frequencies that meets its frequency specifications. For example, if the output frequencies required from one PLL are 33 and 66 MHz, the Quartus II software sets the VCO to 660 MHz (the least common multiple of 33 and 66 MHz within the VCO range). Then the post-scale counters scale down the VCO frequency for each output port.

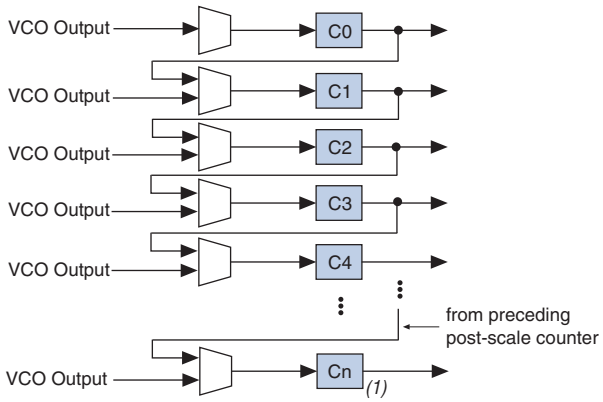
Each PLL has one pre-scale counter, n, and one multiply counter, m, with a range of 1 to 512 for both m and n. The n counter does not use duty-cycle control because the only purpose of this counter is to calculate frequency division. There are seven generic post-scale counters per left or right PLL and ten post-scale counters per top or bottom PLL that can feed the GCLKs, RCLKs, or external clock outputs. These post-scale counters range from 1 to 512 with a 50% duty cycle setting. The high- and low-count values for each counter range from 1 to 256. The sum of the high- and low-count values chosen for a design selects the divide value for a given counter.

The Quartus II software automatically chooses the appropriate scaling factors according to the input frequency, multiplication, and division values entered into the ALTPLL megafunction.

Post-Scale Counter Cascading

Stratix IV PLLs support post-scale counter cascading to create counters larger than 512. This is automatically implemented in the Quartus II software by feeding the output of one c counter into the input of the next c counter, as shown in Figure 5–30.

Figure 5–30. Counter Cascading



Note to Figure 5–30:

(1) N = 6 or N = 9

When cascading post-scale counters to implement a larger division of the high-frequency VCO clock, the cascaded counters behave as one counter with the product of the individual counter settings. For example, if C0 = 40 and C1 = 20, the cascaded value is C0 × C1 = 800.

Post-scale counter cascading is set in the configuration file. You cannot set this using PLL reconfiguration.

Programmable Duty Cycle

The programmable duty cycle allows PLLs to generate clock outputs with a variable duty cycle. This feature is supported on the PLL post-scale counters. The duty-cycle setting is achieved by a low and high time-count setting for the post-scale counters. To determine duty cycle choices, the Quartus II software uses the frequency input and the required multiply or divide rate. The post-scale counter value determines the precision of the duty cycle. Precision is defined as 50% divided by the post-scale counter value. For example, if the `C0` counter is 10, steps of 5% are possible for duty-cycle choices from 5% to 90%.

If the PLL is in external feedback mode, set the duty cycle for the counter driving the `fbin` pin to 50%. Combining the programmable duty cycle with programmable phase shift allows the generation of precise non-overlapping clocks.

Programmable Phase Shift

Use phase shift to implement a robust solution for clock delays in Stratix IV devices. Implement phase shift by using a combination of the VCO phase output and the counter starting time. A combination of VCO phase output and counter starting time is the most accurate method of inserting delays because it is only based on counter settings, which are independent of process, voltage, and temperature (PVT).

You can phase-shift the output clocks from the Stratix IV PLLs in either of these two resolutions:

- Fine resolution using VCO phase taps
- Coarse resolution using counter starting time

Implement fine-resolution phase shifts by allowing any of the output counters (`C[n..0]`) or the `m` counter to use any of the eight phases of the VCO as the reference clock. This allows you to adjust the delay time with a fine resolution. Equation 5–1 shows the minimum delay time that you can insert using this method.

Equation 5–1. Fine-Resolution Phase Shift

$$\Phi_{fine} = \frac{1}{8} T_{VCO} = \frac{1}{8 f_{VCO}} = \frac{N}{8 M f_{REF}}$$

where f_{REF} is the input reference clock frequency.

For example, if f_{REF} is 100 MHz, N is 1, and M is 8, then f_{VCO} is 800 MHz and Φ_{fine} equals 156.25 ps. This phase shift is defined by the PLL operating frequency, which is governed by the reference clock frequency and the counter settings.

Equation 5–2 shows the coarse-resolution phase shifts are implemented by delaying the start of the counters for a predetermined number of counter clocks.

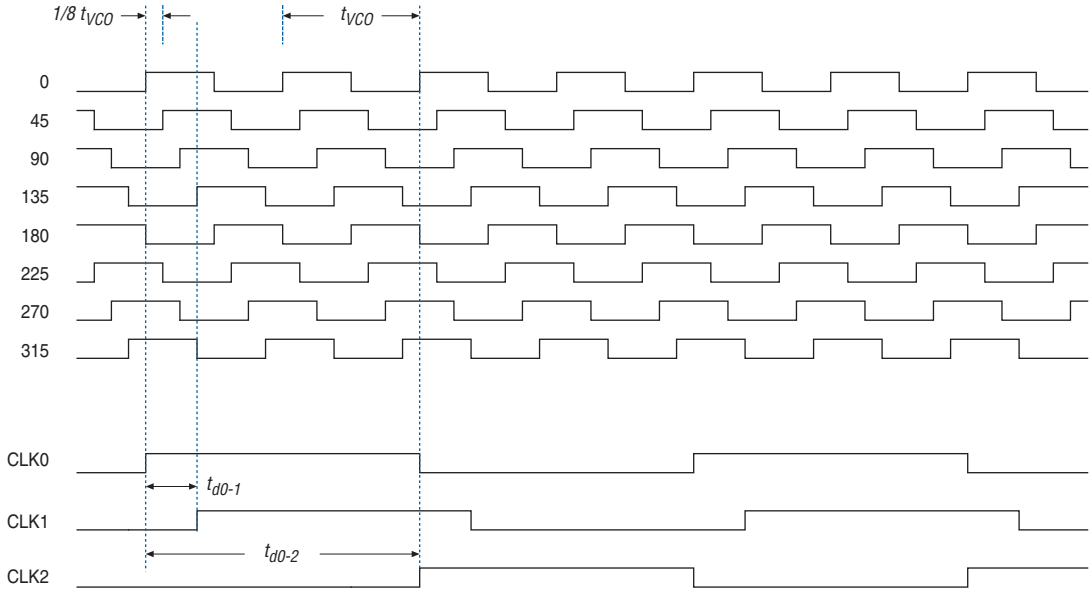
Equation 5–2. Coarse-Resolution Phase Shift

$$\Phi_{coarse} = \frac{C-1}{f_{VCO}} = \frac{(C-1)N}{Mf_{REF}}$$

where C is the count value set for the counter delay time (this is the initial setting in the "PLL usage" section of the compilation report in the Quartus II software). If the initial value is 1, C – 1 = 0° phase shift.

Figure 5–31 shows an example of phase-shift insertion with fine resolution using the VCO phase-taps method. The eight phases from the VCO are shown and labeled for reference. For this example, CLK0 is based on the 0phase from the VCO and has the C value for the counter set to one. The CLK1 signal is divided by four, two VCO clocks for high time and two VCO clocks for low time. CLK1 is based on the 135° phase tap from the VCO and also has the C value for the counter set to one. In this case, the two clocks are offset by 3 Φ_{FINE}. CLK2 is based on the 0phase from the VCO but has the C value for the counter set to **three**. This arrangement creates a delay of 2 Φ_{COARSE} (two complete VCO periods).

Figure 5–31. Delay Insertion Using VCO Phase Output and Counter Delay Time



You can use coarse- and fine-phase shifts to implement clock delays in Stratix IV devices.

Stratix IV devices support dynamic phase-shifting of VCO phase taps only. You can reconfigure the phase shift any number of times. Each phase shift takes about one SCANCLK cycle, allowing you to implement large phase shifts quickly.

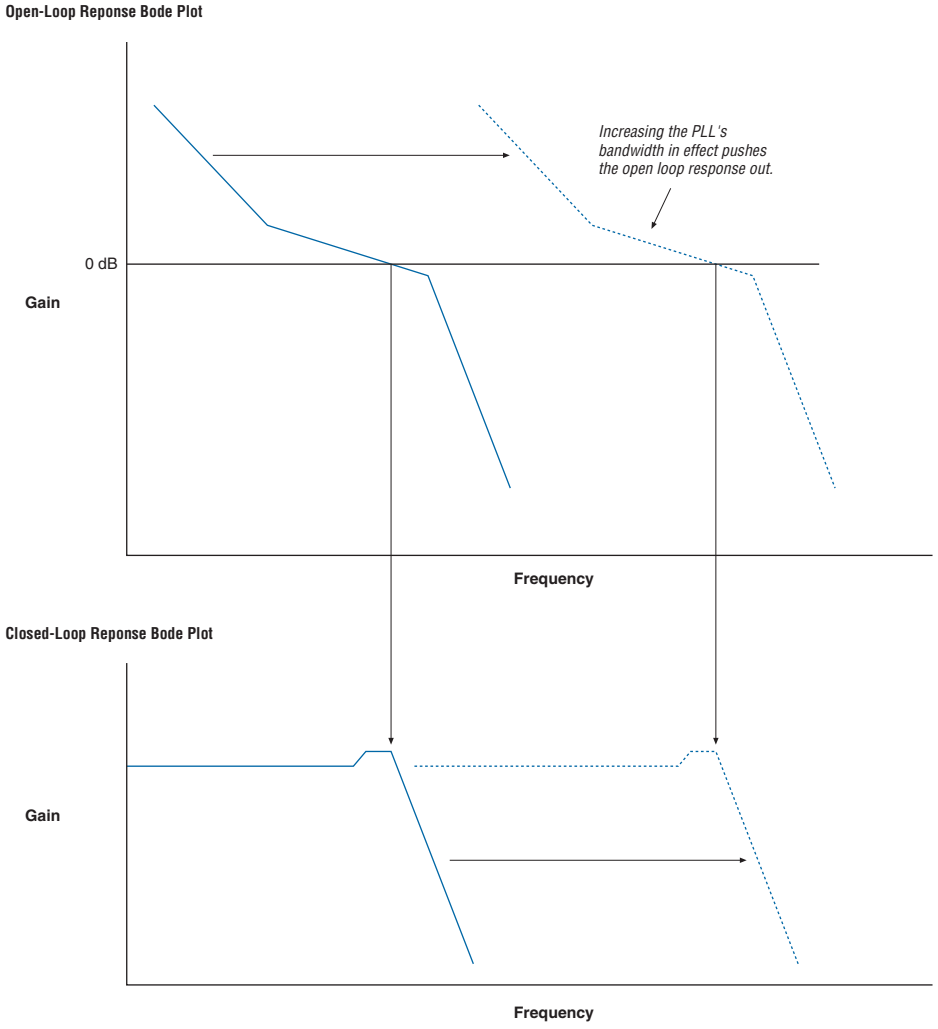
Programmable Bandwidth

Stratix IV PLLs provide advanced control of the PLL bandwidth using the PLL loop's programmable characteristics, including loop filter and charge pump.

Background

PLL bandwidth is the measure of the PLL's ability to track the input clock and its associated jitter. The closed-loop gain 3 dB frequency in the PLL determines PLL bandwidth. Bandwidth is approximately the unity gain point for open loop PLL response. As Figure 5–32 shows, these points correspond to approximately the same frequency. Stratix IV PLLs provide three bandwidth settings—low, medium (default), and high.

Figure 5–32. Open- and Closed-Loop Response Bode Plots



A high-bandwidth PLL provides a fast lock time and tracks jitter on the reference clock source, passing it through to the PLL output. A low-bandwidth PLL filters out reference clock jitter but increases lock time. Stratix IV PLLs allow you to control the bandwidth over a finite range to customize the PLL characteristics for a particular application. The programmable bandwidth feature in Stratix IV PLLs benefits applications requiring clock switchover.

A high-bandwidth PLL can benefit a system that must accept a spread-spectrum clock signal. Stratix IV PLLs can track a spread-spectrum clock by using a high-bandwidth setting. Using a low-bandwidth setting in this case could cause the PLL to filter out the jitter on the input clock.

A low-bandwidth PLL can benefit a system using clock switchover. When clock switchover occurs, the PLL input temporarily stops. A low-bandwidth PLL reacts more slowly to changes on its input clock and takes longer to drift to a lower frequency (caused by input stopping) than a high-bandwidth PLL.

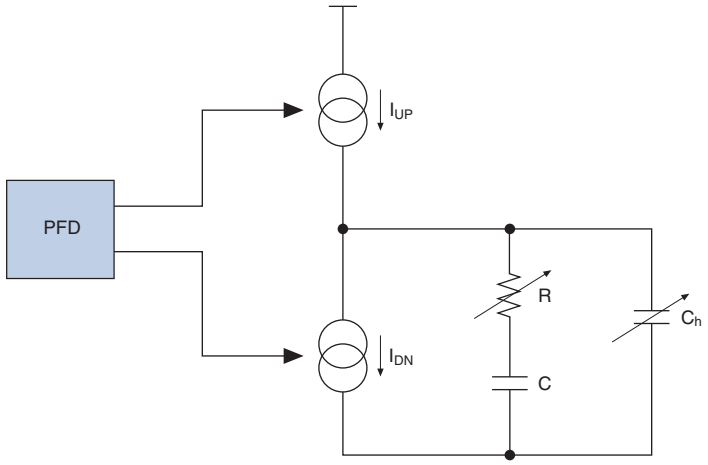
Implementation

Traditionally, external components such as the VCO or loop filter control a PLL's bandwidth. Most loop filters consist of passive components such as resistors and capacitors that take up unnecessary board space and increase cost. With Stratix IV PLLs, all the components are contained within the device to increase performance and decrease cost.

When you specify the bandwidth setting (low, medium, or high) in the ALTPLL MegaWizard Plug-in Manager, the Quartus II software automatically sets the corresponding charge pump and loop filter (`Icp`, `R`, `C`) values to achieve the desired bandwidth range.

Figure 5–33 shows the loop filter and components that you can set using the Quartus II software. The components are the loop filter resistor, R, the high frequency capacitor, C_h, and the charge pump current, I_{UP} or I_{DN}.

Figure 5–33. Loop Filter Programmable Components



Spread-Spectrum Tracking

Stratix IV devices can accept a spread-spectrum input with typical modulation frequencies. However, the device cannot automatically detect that the input is a spread-spectrum signal. Instead, the input signal looks like deterministic jitter at the input of the PLL. Stratix IV PLLs can track a spread-spectrum input clock as long as it is within input-jitter tolerance specifications. Stratix IV devices cannot internally generate spread-spectrum clocks.

Clock Switchover

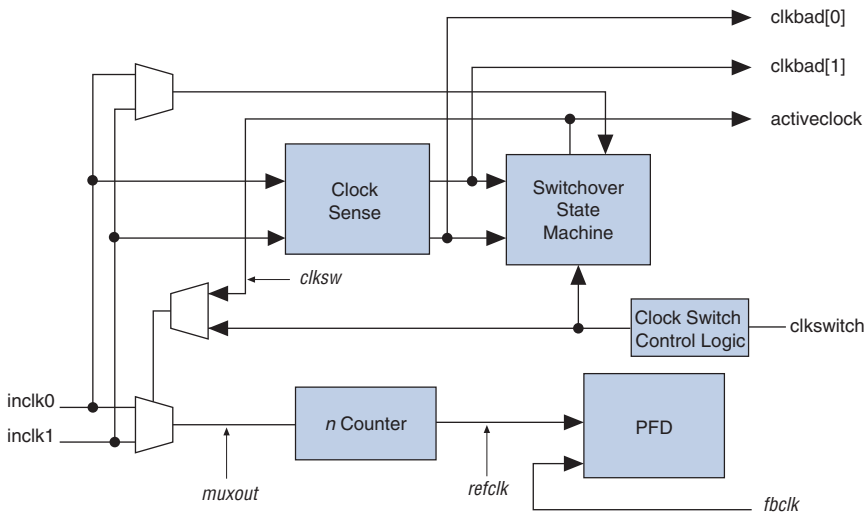
The clock switchover feature allows the PLL to switch between two reference input clocks. Use this feature for clock redundancy or for a dual-clock domain application such as in a system that turns on the redundant clock if the previous clock stops running. The design can perform clock switchover automatically when the clock is no longer toggling or based on a user control signal, `clkswitch`.

The following clock switchover modes are supported in Stratix IV PLLs:

- Automatic switchover—The clock sense circuit monitors the current reference clock and if it stops toggling, automatically switches to the other `inclk0` or `inclk1` clock.
- Manual clock switchover—Clock switchover is controlled using the `clkswitch` signal. When the `clkswitch` signal goes from logic low to logic high, and stays high for at least three clock cycles, the reference clock to the PLL is switched from `inclk0` to `inclk1`, or vice-versa.
- Automatic switchover with manual override—This mode combines automatic switchover and manual clock switchover. When the `clkswitch` signal goes high, it overrides the automatic clock switchover function. As long as the `clkswitch` signal is high, further switchover action is blocked.

Stratix IV PLLs support a fully configurable clock switchover capability. Figure 5–34 shows a block diagram of the automatic switchover circuit built into the PLL. When the current reference clock is not present, the clock sense block automatically switches to the backup clock for PLL reference. The clock switchover circuit also sends out three status signals—`clkbad[0]`, `clkbad[1]`, and `activeclock`—from the PLL to implement a custom switchover circuit in the logic array. You can select a clock source as the backup clock by connecting it to the `inclk1` port of the PLL in your design.

Figure 5–34. Automatic Clock Switchover Circuit Block Diagram



Automatic Clock Switchover

Use the switchover circuitry to automatically switch between `inclk0` and `inclk1` when the current reference clock to the PLL stops toggling. For example, in applications that require a redundant clock with the same frequency as the reference clock, the switchover state machine generates a signal (`clksw`) that controls the multiplexer select input, as shown in Figure 5–34. In this case, `inclk1` becomes the reference clock for the PLL. When using automatic switchover mode, you can switch back and forth between `inclk0` and `inclk1` any number of times when one of the two clocks fails and the other clock is available.

When using automatic clock switchover mode, the following requirements must be satisfied:

- Both clock inputs must be running
- The period of the two clock inputs can differ by no more than 100% (2×)

If the current clock input stops toggling while the other clock is also not toggling, switchover is not initiated and the `clkbad[0..1]` signals are not valid. Also, if both clock inputs are not the same frequency, but their period difference is within 100%, the clock sense block detects when a clock stops toggling, but the PLL may lose lock after the switchover is completed and needs time to re-lock.

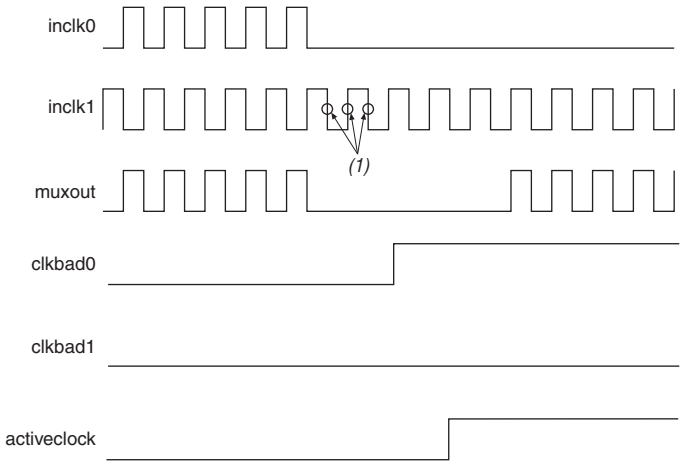
Altera recommends resetting the PLL using the `areset` signal to maintain the phase relationships between the PLL input and output clocks when using clock switchover.

In automatic switchover mode, the clkbad[0] and clkbad[1] signals indicate the status of the two clock inputs. When they are asserted, the clock sense block has detected that the corresponding clock input has stopped toggling. These two signals are not valid if the frequency difference between inclk0 and inclk1 is greater than 20%.

The activeclock signal indicates which of the two clock inputs (inclk0 or inclk1) is being selected as the reference clock to the PLL. When the frequency difference between the two clock inputs is more than 20%, the activeclock signal is the only valid status signal.

Figure 5–35 shows an example waveform of the switchover feature when using automatic switchover mode. In this example, the inclk0 signal is stuck low. After the inclk0 signal is stuck at low for approximately two clock cycles, the clock sense circuitry drives the clkbad[0] signal high. Also, because the reference clock signal is not toggling, the switchover state machine controls the multiplexer through the clkswitch signal to switch to the backup clock, inclk1.

Figure 5–35. Automatic Switchover After Loss of Clock Detection



Note to Figure 5–35:

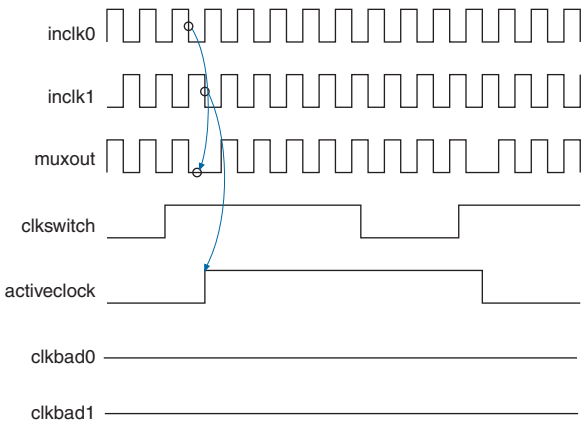
(1) Switchover is enabled on the falling edge of inclk0 or inclk1, depending on which clock is available. In this figure, switchover is enabled on the falling edge of inclk1.

Manual Override

In automatic switchover with manual override mode, you can use the clkswitch input for user- or system-controlled switch conditions. You can use this mode for same-frequency switchover, or to switch between inputs of different frequencies. For example, if inclk0 is 66 MHz and inclk1 is 200 MHz, you must control switchover using clkswitch because the automatic clock-sense circuitry cannot monitor clock input (inclk0 and inclk1) frequencies with a frequency difference of more than 100% (2×). This feature is useful when the clock sources originate from multiple cards on the backplane, requiring a system-controlled switchover between the frequencies of operation. You must choose the backup clock frequency and set the m, n, c, and k counters accordingly so the VCO operates within the recommended operating frequency range of 600 to 1,600 MHz. The ALTPLL MegaWizard Plug-in Manager notifies you if a given combination of inclk0 and inclk1 frequencies cannot meet this requirement.

Figure 5–36 shows a clock switchover waveform controlled by clkswitch. In this case, both clock sources are functional and inclk0 is selected as the reference clock; clkswitch goes high, which starts the switchover sequence. On the falling edge of inclk0, the counter's reference clock, muxout, is gated off to prevent clock glitching. On the falling edge of inclk1, the reference clock multiplexer switches from inclk0 to inclk1 as the PLL reference and the activeclock signal changes to indicate which clock is currently feeding the PLL.

Figure 5–36. Clock Switchover Using the clkswitch (Manual) Control (1)



Note to Figure 5–36:

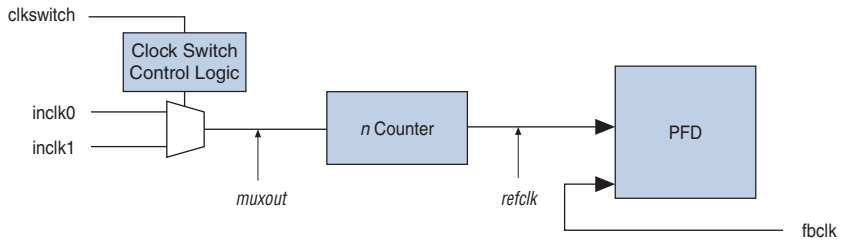
(1) To initiate a manual clock switchover event, both inclk0 and inclk1 must be running when the clkswitch signal goes high.

In automatic override with manual switchover mode, the activeclock signal mirrors the clkswitch signal. As both clocks are still functional during the manual switch, neither clkbad signal goes high. Because the switchover circuit is positive-edge sensitive, the falling edge of the clkswitch signal does not cause the circuit to switch back from inclk1 to inclk0. When the clkswitch signal goes high again, the process repeats. clkswitch and automatic switch only work if the clock being switched to is available. If the clock is not available, the state machine waits until the clock is available.

Manual Clock Switchover

In manual clock switchover mode, the `clkswitch` signal controls whether `inclk0` or `inclk1` is selected as the input clock to the PLL. By default, `inclk0` is selected. A low-to-high transition on `clkswitch` and `clkswitch` being held high for at least three `inclk` cycles initiates a clock switchover event. You must bring `clkswitch` back low again in order to perform another switchover event in the future. If you do not require another switchover event in the future, you can leave `clkswitch` in a logic high state after the initial switch. Pulsing `clkswitch` high for at least three `inclk` cycles performs another switchover event. If `inclk0` and `inclk1` are different frequencies and are always running, the `clkswitch` minimum high time must be greater than or equal to three of the slower frequency `inclk0` or `inclk1` cycles. Figure 5–37 shows a block diagram of the manual switchover circuit.

Figure 5–37. Manual Clock Switchover Circuitry in Stratix IV PLLs



For more information about PLL software support in the Quartus II software, refer to the *Phase-Locked Loop (ALTPLL) Megafunction User Guide*.

Guidelines

When implementing clock switchover in Stratix IV PLLs, use the following guidelines:

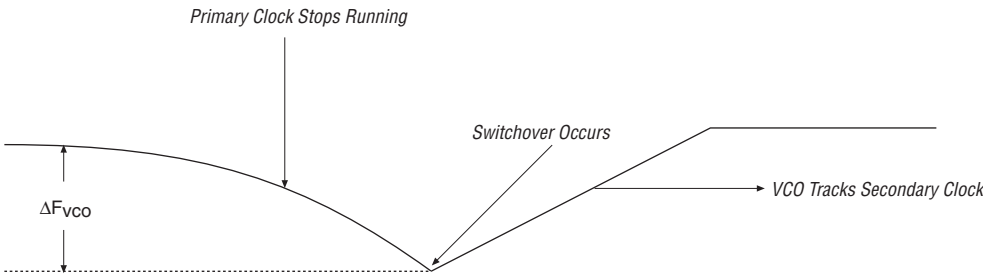
- Automatic clock switchover requires that the `inclk0` and `inclk1` frequencies be within 100% (2×) of each other. Failing to meet this requirement causes the `clkbad[0]` and `clkbad[1]` signals to not function properly.
- When using manual clock switchover, the difference between `inclk0` and `inclk1` can be more than 100% (2×). However, differences in frequency, phase, or both, of the two clock sources will likely cause the PLL to lose lock. Resetting the PLL ensures that the correct phase relationships are maintained between the input and output clocks.

  Both `inclk0` and `inclk1` must be running when the `clkswitch` signal goes high to initiate the manual clock switchover event. Failing to meet this requirement causes the clock switchover to not function properly.

- Applications that require a clock switchover feature and a small frequency drift must use a low-bandwidth PLL. The low-bandwidth PLL reacts more slowly than a high-bandwidth PLL to reference input clock changes. When switchover happens, a low-bandwidth PLL propagates the stopping of the clock to the output more slowly than a high-bandwidth PLL. However, be aware that the low-bandwidth PLL also increases lock time.

- After a switchover occurs, there may be a finite resynchronization period for the PLL to lock onto a new clock. The exact amount of time it takes for the PLL to re-lock depends on the PLL configuration.
- The phase relationship between the input clock to the PLL and the output clock from the PLL is important in your design. Assert `areset` for at least 10 ns after performing a clock switchover. Wait for the locked signal to go high and be stable before re-enabling the output clocks from the PLL.
- Figure 5–38 shows how the VCO frequency gradually decreases when the current clock is lost and then increases as the VCO locks on to the backup clock.

Figure 5–38. VCO Switchover Operating Frequency



- Disable the system during clock switchover if it is not tolerant of frequency variations during the PLL resynchronization period. You can use the `clkbad[0]` and `clkbad[1]` status signals to turn off the PFD (`PFDENA = 0`) so the VCO maintains its most recent frequency. You can also use the state machine to switch over to the secondary clock. When the PFD is re-enabled, output clock-enable signals (`clkena`) can disable clock outputs during the switchover and resynchronization period. When the lock indication is stable, the system can re-enable the output clocks.

PLL Reconfiguration

PLLs use several divide counters and different VCO phase taps to perform frequency synthesis and phase shifts. In Stratix IV PLLs, you can reconfigure both the counter settings and phase-shift the PLL output clock in real time. You can also change the charge pump and loop-filter components, which dynamically affects PLL bandwidth. You can use these PLL components to update the output-clock frequency and PLL bandwidth and to phase-shift in real time, without reconfiguring the entire Stratix IV device.

The ability to reconfigure the PLL in real time is useful in applications that operate at multiple frequencies. It is also useful in prototyping environments, allowing you to sweep PLL output frequencies and adjust the output-clock phase dynamically. For instance, a system generating test patterns is required to generate and transmit patterns at 75 or 150 MHz, depending on the requirements of the device under test. Reconfiguring the PLL components in real time allows you to switch between two such output frequencies within a few microseconds. You can also use this feature to adjust clock-to-out (`tCO`) delays in real time by changing the PLL output clock phase shift. This approach eliminates the need to regenerate a configuration file with the new PLL settings.

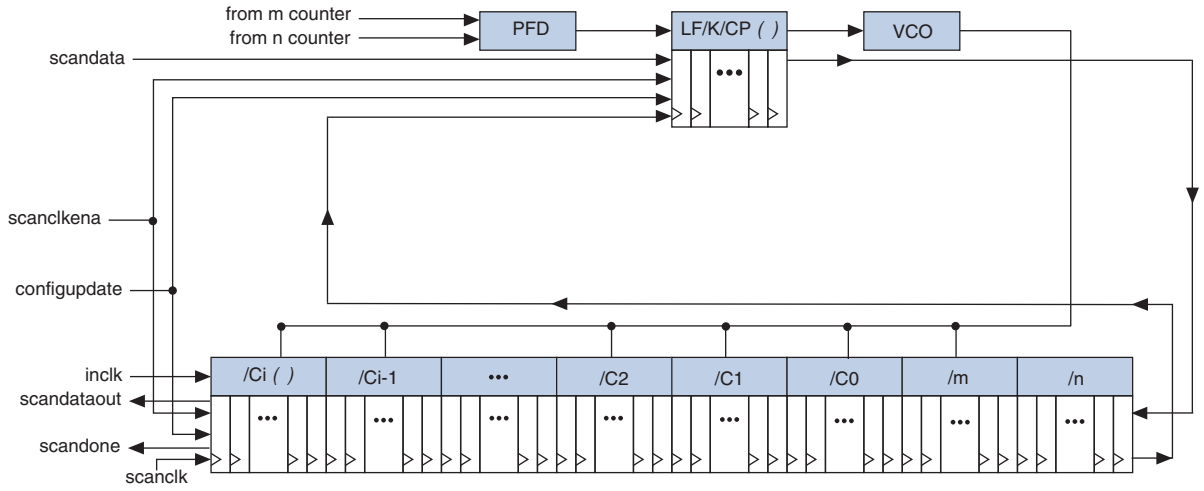
PLL Reconfiguration Hardware Implementation

The following PLL components are reconfigurable in real time:

- Pre-scale counter (n)
- Feedback counter (m)
- Post-scale output counters (C0 - C9)
- Post VCO Divider (K)
- Dynamically adjust the charge-pump current (Icp) and loop-filter components (R, C) to facilitate reconfiguration of the PLL bandwidth

Figure 5–39 shows how you can dynamically adjust the PLL counter settings by shifting their new settings into a serial shift-register chain or scan chain. Serial data is input to the scan chain using the scandata port. Shift registers are clocked by scanclk. The maximum scanclk frequency is 100 MHz. Serial data is shifted through the scan chain as long as the scanclkena signal stays asserted. After the last bit of data is clocked, asserting the configupdate signal for at least one scanclk clock cycle causes the PLL configuration bits to be synchronously updated with the data in the scan registers.

Figure 5–39. PLL Reconfiguration Scan Chain (1)



Notes to Figure 5–39:

(1) Stratix IV left and right PLLs support C0 - C6 counters.

(2) i = 6 or i = 9.

(3) This figure shows the corresponding scan register for the K counter in between the scan registers for the charge pump and loop filter. The K counter is physically located after the VCO.

The counter settings are updated synchronously to the clock frequency of the individual counters. Therefore, all counters are not updated simultaneously.

Table 5–10 lists how these signals can be driven by the PLD logic array or I/O pins.

Table 5–10. Real-Time PLL Reconfiguration Ports

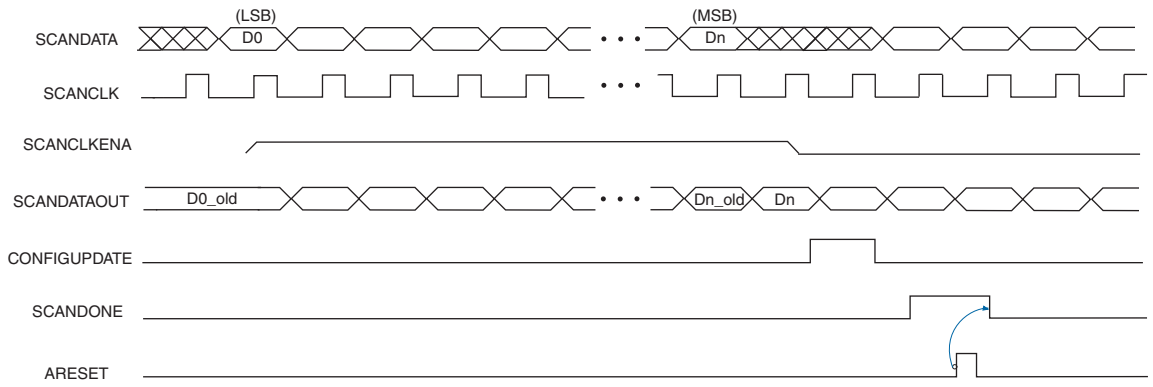
PLL Port Name	Description	Source	Destination
scandata	Serial input data stream to scan chain.	Logic array or I/O pin	PLL reconfiguration circuit
scanclk	Serial clock input signal. This clock can be free running.	GCLK, RCLK or I/O pins	PLL reconfiguration circuit
scanclkena	Enables scanclk and allows the scandata to be loaded in the scan chain. Active high.	Logic array or I/O pin	PLL reconfiguration circuit
configupdate	Writes the data in the scan chain to the PLL. Active high.	Logic array or I/O pin	PLL reconfiguration circuit
scandone	Indicates when the PLL has finished reprogramming. A rising edge indicates the PLL has begun reprogramming. A falling edge indicates the PLL has finished reprogramming.	PLL reconfiguration circuit	Logic array or I/O pins
scandataout	Used to output the contents of the scan chain.	PLL reconfiguration circuit	Logic array or I/O pins

To reconfigure the PLL counters, follow these steps:

1. The scanclkena signal is asserted at least one scanclk cycle prior to shifting in the first bit of scandata (D0).
2. Serial data (scandata) is shifted into the scan chain on the second rising edge of scanclk.
3. After all 234 bits (top and bottom PLLs) or 180 bits (left and right PLLs) have been scanned into the scan chain, the scanclkena signal is de-asserted to prevent inadvertent shifting of bits in the scan chain.
4. The configupdate signal is asserted for one scanclk cycle to update the PLL counters with the contents of the scan chain.
5. The scandone signal goes high, indicating the PLL is being reconfigured. A falling edge indicates the PLL counters have been updated with new settings.
6. Reset the PLL using the areset signal if you make any changes to the M, N, or post-scale output C counters or to the Icp, R, or C settings.
7. You can repeat steps 1-5 to reconfigure the PLL any number of times.

Figure 5–40 shows a functional simulation of the PLL reconfiguration feature.

Figure 5–40. PLL Reconfiguration Waveform



When you reconfigure the counter clock frequency, you cannot reconfigure the corresponding counter phase shift settings using the same interface. Instead, reconfigure the phase shifts in real time using the dynamic phase shift reconfiguration interface. If you reconfigure the counter frequency, but wish to keep the same non-zero phase shift setting (for example, 90°) on the clock output, you must reconfigure the phase shift immediately after reconfiguring the counter clock frequency.

Post-Scale Counters (C0 to C9)

You can reconfigure the multiply or divide values and duty cycle of post-scale counters in real time. Each counter has an 8-bit high-time setting and an 8-bit low-time setting. The duty cycle is the ratio of output high- or low-time to the total cycle time, which is the sum of the two. Additionally, these counters have two control bits, `rbypass`, for bypassing the counter, and `rselodd`, to select the output clock duty cycle.

When the `rbypass` bit is set to 1, it bypasses the counter, resulting in a divide by 1. When the `rbypass` bit is set to 0, the high- and low-time counters are added to compute the effective division of the VCO output frequency. For example, if the post-scale divide factor is 10, the high- and low-count values can be set to 5 and 5, respectively, to achieve a 50% - 50% duty cycle. The PLL implements this duty cycle by transitioning the output clock from high to low on the rising edge of the VCO output clock. However, a 4 and 6 setting for the high- and low-count values, respectively, produces an output clock with a 40% - 60% duty cycle.

The rselodd bit indicates an odd divide factor for the VCO output frequency along with a 50% duty cycle. For example, if the post-scale divide factor is 3, the high- and low-time count values could be set to 2 and 1, respectively, to achieve this division. This implies a 67% - 33% duty cycle. If you need a 50% - 50% duty cycle, you can set the rselodd control bit to 1 to achieve this duty cycle despite an odd division factor. The PLL implements this duty cycle by transitioning the output clock from high to low on a falling edge of the VCO output clock. When you set rselodd = 1, you subtract 0.5 cycles from the high time and you add 0.5 cycles to the low time. For example:

- High-time count = 2 cycles
- Low-time count = 1 cycle
- rselodd = 1 effectively equals:
  - High-time count = 1.5 cycles
  - Low-time count = 1.5 cycles
  - Duty cycle = (1.5/3) % high-time count and (1.5/3) % low-time count

Scan Chain Description

The length of the scan chain varies for different Stratix IV PLLs. The top and bottom PLLs have ten post-scale counters and a 234-bit scan chain, while the left and right PLLs have seven post-scale counters and a 180-bit scan chain. Table 5–11 lists the number of bits for each component of a Stratix IV PLL.

Table 5–11. Top and Bottom PLL Reprogramming Bits (Part 1 of 2)

Block Name	Number of Bits		Total
	Counter	Other (1)	
C9 (2)	16	2	18
C8	16	2	18
C7	16	2	18
C6 (3)	16	2	18
C5	16	2	18
C4	16	2	18
C3	16	2	18
C2	16	2	18
C1	16	2	18
C0	16	2	18
M	16	2	18
N	16	2	18
Charge Pump Current	0	3	3
VCO Post-Scale divider (K)	1	0	1

Table 5–11. Top and Bottom PLL Reprogramming Bits (Part 2 of 2)

Block Name	Number of Bits		Total
	Counter	Other (1)	
Loop Filter Capacitor (4)	0	2	2
Loop Filter Resistor	0	5	5
Unused CP/LF	0	7	7
Total number of bits	—	—	234

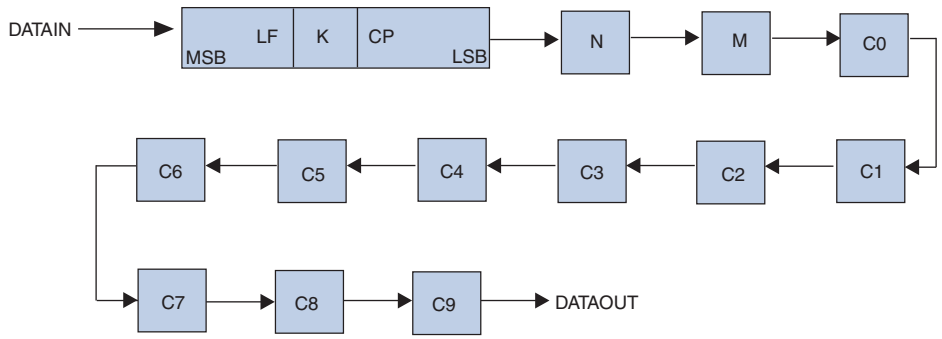
Notes to Table 5–11:

(1) Includes two control bits, `rbypass`, for bypassing the counter, and `rselodd`, to select the output clock duty cycle.
(2) The LSB for the C9 low-count value is the first bit shifted into the scan chain for the top and bottom PLLs.
(3) The LSB for the C6 low-count value is the first bit shifted into the scan chain for the left and right PLLs.
(4) The MSB for the loop filter is the last bit shifted into the scan chain.

Table 5–11 lists the scan chain order of PLL components for the top and bottom PLLs, which have 10 post-scale counters. The order of bits is the same for the left and right PLLs, but the reconfiguration bits start with the C6 post-scale counter.

Figure 5–41 shows the scan-chain order of PLL components for the top and bottom PLLs.

Figure 5–41. Scan-Chain Order of PLL Components for Top and Bottom PLLs (1)

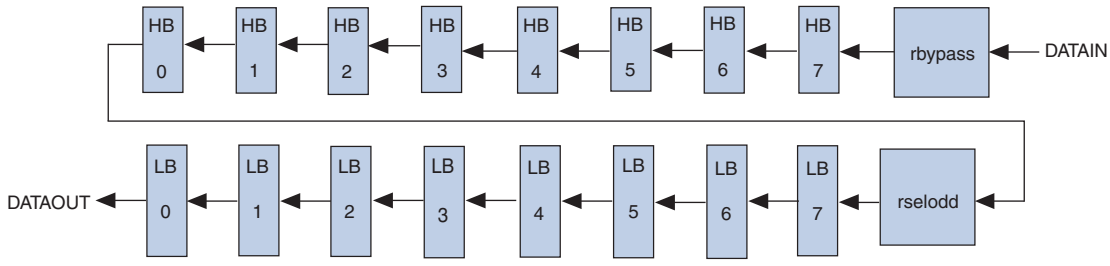


Note to Figure 5–41:

(1) Left and right PLLs have the same scan-chain order. The post-scale counters end at C6.

Figure 5–42 shows the scan-chain bit-order sequence for post-scale counters in all Stratix IV PLLs.

Figure 5–42. Scan-Chain Bit-Order Sequence for Post-Scale Counters in Stratix IV PLLs



Charge Pump and Loop Filter

You can reconfigure the charge-pump and loop-filter settings to update the PLL bandwidth in real time.

Table 5–12 lists the possible settings for charge pump current (Icp) values for Stratix IV PLLs.

Table 5–12. Charge Pump Current Bit Settings

CP[2]	CP[1]	CP[0]	Decimal Value for Setting
0	0	0	0
0	0	1	1
0	1	1	3
1	1	1	7

Table 5–13 lists the possible settings for loop-filter resistor (R) values for Stratix IV PLLs.

Table 5–13. Loop-Filter Resistor Bit Settings

LFR[4]	LFR[3]	LFR[2]	LFR[1]	LFR[0]	Decimal Value for Setting
0	0	0	0	0	0
0	0	0	1	1	3
0	0	1	0	0	4
0	1	0	0	0	8
1	0	0	0	0	16
1	0	0	1	1	19
1	0	1	0	0	20
1	1	0	0	0	24
1	1	0	1	1	27
1	1	1	0	0	28
1	1	1	1	0	30

Table 5–14 lists the possible settings for loop-filter capacitor (C) values for Stratix IV PLLs.

Table 5–14. Loop-Filter Capacitor Bit Settings

LFC[1]	LFC[0]	Decimal Value for Setting
0	0	0
0	1	1
1	1	3

Bypassing a PLL

Bypassing a PLL counter results in a multiply (m counter) or a divide (n and C0 to C9 counters) factor of one.

Table 5–15 lists the settings for bypassing the counters in Stratix IV PLLs.

Table 5–15. PLL Counter Settings

PLL Scan Chain Bits [0..8] Settings									
LSB								**MSB**	**Description**
X	X	X	X	X	X	X	X	1 (1)	PLL counter bypassed
X	X	X	X	X	X	X	X	0 (1)	PLL counter not bypassed because bit 8 (MSB) is set to 0

Note to Table 5–15:

(1) Counter-bypass bit.

To bypass any of the PLL counters, set the bypass bit to **1**. The values on the other bits are ignored. To bypass the VCO post-scale counter (K), set the corresponding bit to 0.

Dynamic Phase-Shifting

The dynamic phase-shifting feature allows the output phases of individual PLL outputs to be dynamically adjusted relative to each other and to the reference clock, without having to send serial data through the scan chain of the corresponding PLL. This feature simplifies the interface and allows you to quickly adjust the clock-to-out (t_{CO}) delays by changing the output clock phase-shift in real time. This adjustment is achieved by incrementing or decrementing the VCO phase-tap selection to a given C counter or to the M counter. The phase is shifted by 1/8 of the VCO frequency at a time. The output clocks are active during this phase-reconfiguration process.

Table 5–16 lists the control signals that are used for dynamic phase-shifting.

Table 5–16. Dynamic Phase-Shifting Control Signals (Part 1 of 2)

Signal Name	Description	Source	Destination
PHASECOUNTERSELECT [3..0]	Counter select. Four bits decoded to select either the M or one of the C counters for phase adjustment. One address maps to select all C counters. This signal is registered in the PLL on the rising edge of SCANCLK.	Logic array or I/O pins	PLL reconfiguration circuit
PHASEUPDOWN	Selects dynamic phase shift direction; 1 = UP; 0 = DOWN. Signal is registered in the PLL on the rising edge of SCANCLK.	Logic array or I/O pin	PLL reconfiguration circuit
PHASESTEP	Logic high enables dynamic phase shifting.	Logic array or I/O pin	PLL reconfiguration circuit

Table 5–16. Dynamic Phase-Shifting Control Signals (Part 2 of 2)

Signal Name	Description	Source	Destination
SCANCLK	Free running clock from the core used in combination with PHASESTEP to enable and disable dynamic phase shifting. Shared with SCANCLK for dynamic reconfiguration.	GCLK, RCLK or I/O pin	PLL reconfiguration circuit
PHASEDONE	When asserted, this indicates to core-logic that the phase adjustment is complete and the PLL is ready to act on a possible second adjustment pulse. Asserts based on internal PLL timing. De-asserts on the rising edge of SCANCLK.	PLL reconfiguration circuit	Logic array or I/O pins

Table 5–17 lists the PLL counter selection based on the corresponding PHASECOUNTERSELECT setting.

Table 5–17. Phase Counter Select Mapping

PHASECOUNTERSELECT[3]	[2]	[1]	[0]	Selects
0	0	0	0	All Output Counters
0	0	0	1	M Counter
0	0	1	0	C0 Counter
0	0	1	1	C1 Counter
0	1	0	0	C2 Counter
0	1	0	1	C3 Counter
0	1	1	0	C4 Counter
0	1	1	1	C5 Counter
1	0	0	0	C6 Counter
1	0	0	1	C7 Counter
1	0	1	0	C8 Counter
1	0	1	1	C9 Counter

To perform one dynamic phase-shift, follow these steps:

1. Set PHASEUPDOWN and PHASECOUNTERSELECT as required.
2. Assert PHASESTEP for at least two SCANCLK cycles. Each PHASESTEP pulse enables one phase shift.
3. Deassert PHASESTEP after PHASEDONE goes low.
4. Wait for PHASEDONE to go high.
5. Repeat steps 1-4 as many times as required to perform multiple phase-shifts.

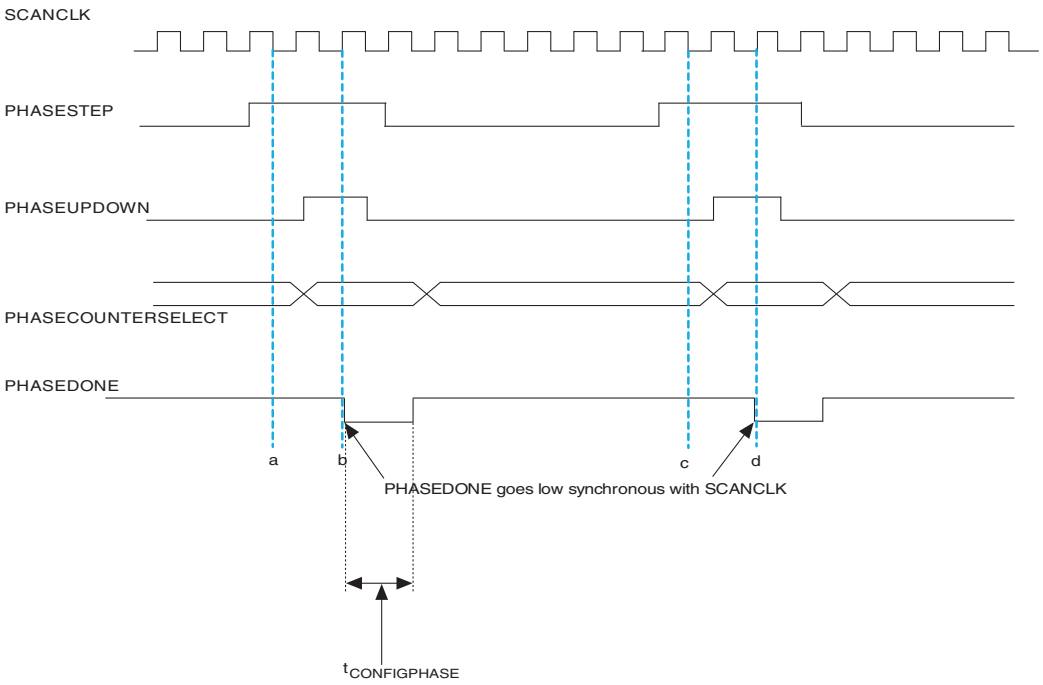
The PHASEUPDOWN and PHASECOUNTERSELECT signals are synchronous to SCANCLK and must meet tsu/th requirements with respect to SCANCLK edges.

You can repeat dynamic phase-shifting indefinitely. For example, in a design where the VCO frequency is set to 1000 MHz and the output clock frequency is 100 MHz, performing 40 dynamic phase shifts (each one yields 125 ps phase shift) results in shifting the output clock by 180°, which is a phase shift of 5 ns.

The PHASESTEP signal is latched on the negative edge of SCANCLK (a,c) and must remain asserted for at least two SCANCLK cycles. De-assert PHASESTEP after PHASEDONE goes low. On the second SCANCLK rising edge (b,d) after PHASESTEP is latched, the values of PHASEUPDOWN and PHASECOUNTERSELECT are latched and the PLL starts dynamic phase-shifting for the specified counters and in the indicated direction. PHASEDONE is de-asserted synchronous to SCANCLK at the second rising edge (b,d) and remains low until the PLL finishes dynamic phase-shifting. Depending on the VCO and SCANCLK frequencies, PHASEDONE low time may be greater than or less than one SCANCLK cycle.

You can perform another dynamic phase-shift after the PHASEDONE signal goes from low to high. Each PHASESTEP pulse enables one phase shift. PHASESTEP pulses must be at least one SCANCLK cycle apart.

Figure 5–43. Dynamic Phase Shifting Waveform



Depending on the VCO and SCANCLK frequencies, PHASEDONE low time may be greater than or less than one SCANCLK cycle.

After PHASEDONE goes from low to high, you can perform another dynamic phase shift. PHASESTEP pulses must be at least one SCANCLK cycle apart.

For information about the ALTPLL_RECONFIG MegaWizard Plug-In Manager, refer to the *Phase-Locked Loops Reconfiguration (ALTPLL_RECONFIG) Megafunction User Guide*.

PLL Specifications

For information about PLL timing specifications, refer to the *DC and Switching Characteristics for Stratix IV Devices* chapter.

Document Revision History

Table 5–18 lists the revision history for this chapter.

Table 5–18. Document Revision History (Part 1 of 2)

Date	Version	Changes
September 2012	3.4	Updated the "Periphery Clock Networks" section.
December 2011	3.3	■ Updated the "Dynamic Phase-Shifting" section. ■ Updated Figure 5–43.
February 2011	3.2	■ Updated the "Clock Input Connections to the PLLs," "PLL Clock I/O Pins," "Clock Feedback Modes," and "Clock Switchover" sections. ■ Updated Table 5–4 and Table 5–8. ■ Updated Figure 5–26, Figure 5–40, and Figure 5–43. ■ Applied new template. ■ Minor text edits.
March 2010	3.1	■ Updated Table 5–3. ■ Updated notes to Figure 5–2, Figure 5–3, Figure 5–4, and Figure 5–9. ■ Added a note to Table 5–5 and Table 5–6. ■ Added two notes to Table 5–4. ■ Updated Figure 5–43. ■ Updated the "Dynamic Phase-Shifting" section. ■ Minor text edits.
November 2009	3.0	■ Updated Table 5–1 and Table 5–7. ■ Updated "Clock Networks in Stratix IV Devices", "Periphery Clock Networks", and "Cascading PLLs" sections. ■ Added Figure 5–5, Figure 5–6, Figure 5–7, Figure 5–8, and Figure 5–9. ■ Added "Clock Sources Per Region" section. ■ Updated Figure 5–40. ■ Removed EP4SE110, EP4SE290, and EP4SE680 devices. ■ Added EP4S40G2, EP4S100G2, EP4S40G5, EP4S100G3, EP4S100G4, EP4S100G5, and EP4SE820 devices.
June 2009	2.3	■ Updated Table 5–7. ■ Updated the "PLL Reconfiguration Hardware Implementation" and "Zero-Delay Buffer Mode" sections. ■ Added introductory sentences to improve search ability. ■ Removed the Conclusion section. ■ Minor text edits.

Table 5–18. Document Revision History (Part 2 of 2)

Date	Version	Changes
April 2009	2.2	■ Updated Table 5–1 and Table 5–7. ■ Updated Figure 5–3 and Figure 5–4. ■ Updated the "Periphery Clock Networks" section.
March 2009	2.1	■ Updated Table 5–7. ■ Updated Figure 5–34. ■ Updated "Guidelines" section. ■ Removed "Referenced Documents" section.
November 2008	2.0	■ Updated Table 5–7. ■ Updated Note 1 of Figure 5–10. ■ Updated Figure 5–15. ■ Updated Figure 5–20. ■ Added Figure 5–21. ■ Made minor editorial changes.
May 2008	1.0	Initial release.