6. Clock Networks and PLLs in Stratix III Devices

SIII51006-2.0

This chapter describes the hierarchical clock networks and multiple phase-locked loops (PLLs) with advanced features in Stratix® III devices. The large number of clocking resources, in combination with the clock synthesis precision provided by the PLLs, provide a complete clock management solution. The Altera® Quartus® II software compiler automatically turns off clock networks not used in the design, thereby reducing the overall power consumption of the device.

Stratix III devices deliver abundant PLL resources with up to 12 PLLs per device and up to 10 outputs per PLL. You can independently program every output, creating a unique, customizable clock frequency. Inherent jitter filtration and fine granularity control over multiply, divide ratios, and dynamic phase shift reconfiguration provide the high performance precision required in today's high-speed applications. Stratix III device PLLs are feature-rich, supporting advanced capabilities such as clock switchover, dynamic phase shifting, PLL reconfiguration, and reconfigurable bandwidth. Stratix III PLLs also support external feedback mode, spread-spectrum tracking, and post-scale counter cascading features.

The Quartus II software enables the PLLs and their features without requiring any external devices. The following sections describe the Stratix III clock networks and PLLs in detail.

Clock Networks in Stratix III Devices

The global clock networks (GCLKs), regional clock networks (RCLKs), and periphery clock networks (PCLKs) available in Stratix III devices are organized into hierarchical clock structures that provide up to 220 unique clock domains (16 GCLKs + 88 RCLKs + 116 PCLKs) within the Stratix III device and allow up to 67 unique GCLK, RCLK, and PCLK clock sources (16 GCLKs + 22 RCLKs + 29 PCLKs) per device quadrant.

Table 6–1 lists the clock resources available in Stratix III devices.

Table 6–1. Clock Resources in Stratix III Devices (Part 1 of 2)

Clock Resource	# 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
Global clock networks	16	`CLK[0..15]p/n` pins, PLL clock outputs, and logic array
Regional clock networks	64/88 *(1)*	`CLK[0..15]p/n` pins, PLL clock outputs, and logic array
Peripheral clock networks	116 (29 per device quadrant) *(2)*	DPA clock outputs, horizontal I/O pins, and logic array
GCLKs/RCLKs per quadrant	32/38 *(3)*	16 GCLKs + 16 RCLKs/ 16 GCLKs + 22 RCLKs

Table 6–1. Clock Resources in Stratix III Devices (Part 2 of 2)

Clock Resource	# 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 6–1:

(1) There are 64 RCLKs in EP3SL50, EP3SL70, EP3SL110, EP3SL150, EP3SE50, EP3SE80, and EP3SE110 devices. There are 88 RCLKs in EP3SL200, EP3SE260, and EP3SL340 devices.

(2) There are 56 PCLKs in EP3SL50, EP3SL70, and EP3SE50 devices. There are 88 PCLKs in EP3SL110, EP3SL150, EP3SL200, EP3SE80, and EP3SE110 devices. There are 112 PCLKs in EP3SE260 and 132 PCLKs in the EP3SL340 device.

(3) There are 32 GCLKs/RCLKs per quadrant in EP3SL50, EP3SL70, EP3SL110, EP3SL150, EP3SE50, EP3SE80, and EP3SE110 devices. There are 38 GCLKs/RCLKs per quadrant in EP3SL200, EP3SE260, and EP3SL340 devices.

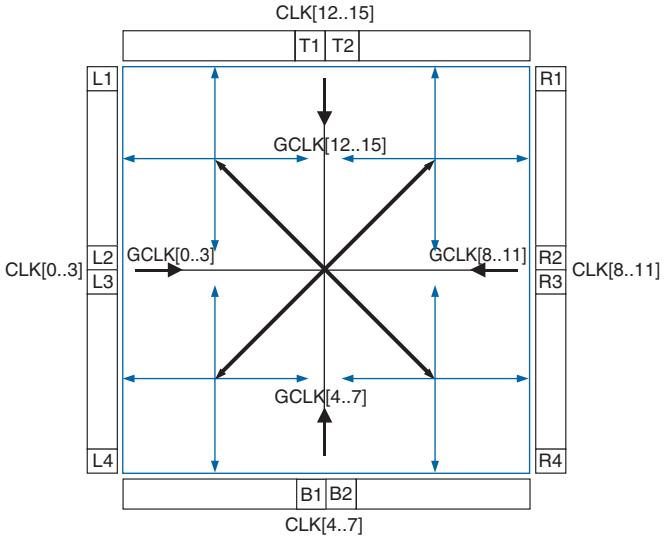
(4) There are 80 GCLKs/RCLKs per entire device in EP3SL50, EP3SL70, EP3SL110, EP3SL150, EP3SE50, EP3SE80, and EP3SE110 devices. There are 104 GCLKs/RCLKS per entire device in EP3SL200, EP3SE260, and EP3SL340 devices.

Stratix III 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 III device, as shown in Figure 6–1 to Figure 6–4.

Global Clock Networks

Stratix III devices provide up to 16 GCLKs that can drive throughout the entire 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 III 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 6–1 shows CLK pins and PLLs that can drive GCLK networks in Stratix III devices.

Figure 6–1. Global Clock Networks



Regional Clock Networks

The regional clock (RCLK) networks only pertain to the quadrant they drive into. The RCLK networks provide the lowest clock delay and skew for logic contained within a single device quadrant. Stratix III device I/O elements 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 6–2 to Figure 6–4 show CLK pins and PLLs that can drive RCLK networks in Stratix III devices. The EP3SL50, EP3SL70, EP3SL110, EP3SL150, EP3SE50, EP3SE80, and EP3SE110 devices contain 64 RCLKs; the EP3SL200, EP3SE260, and EP3SL340 devices contain 88 RCLKs.

Figure 6–2. Regional Clock Networks (EP3SL50, EP3SL70, and EP3SE50 Devices)

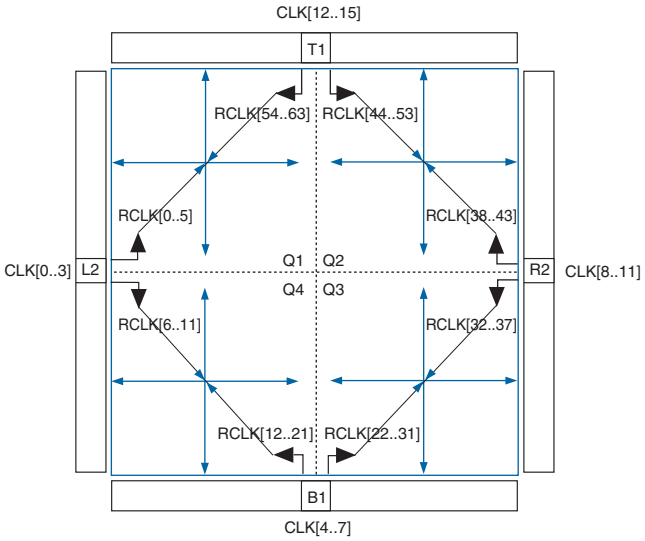


Figure 6–3. Regional Clock Networks (EP3SL110, EP3SL150, EP3SE80, and EP3SE110 Devices)

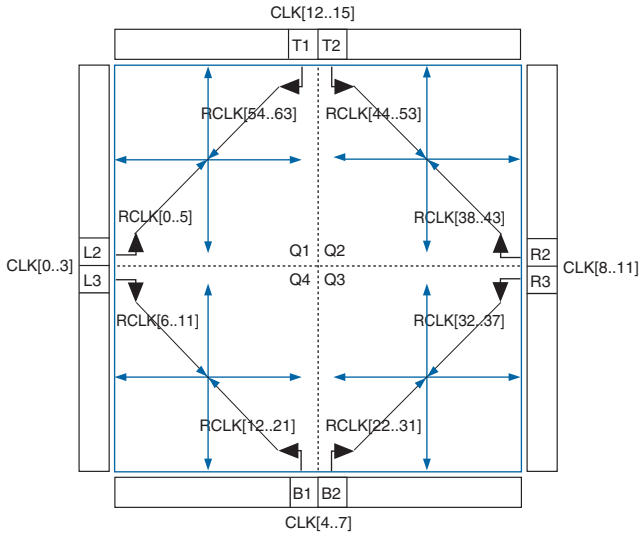
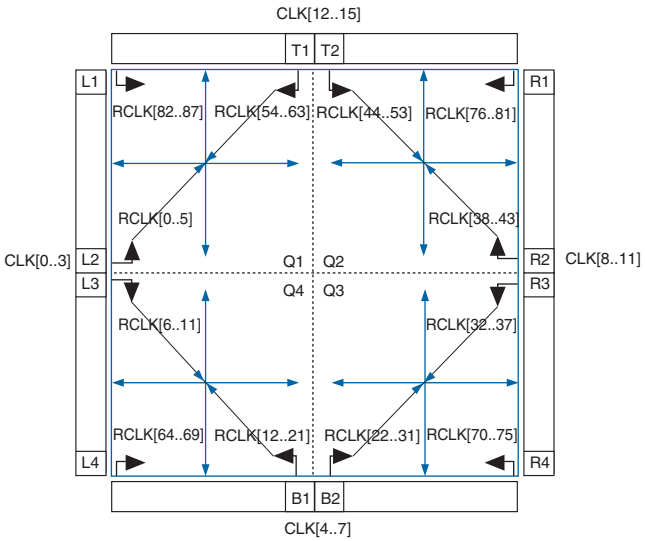


Figure 6–4. Regional Clock Networks (EP3SL200, EP3SE260, and EP3SL340 Devices) *(Note 1)*



Note to Figure 6–4:

(1) The corner RCLKs [64..87] can only be fed by their respective corner PLL outputs. Refer to Table 6–9 on page 6–13 for connectivity.

Periphery Clock Networks

Periphery clock (PCLK) networks shown in Figure 6–5 to Figure 6–9 are a collection of individual clock networks driven from the periphery of the Stratix III device. Clock outputs from the DPA block, horizontal I/O pins, and internal logic can drive the PCLK networks. The EP3SL50, EP3SL70, and EP3SE50 devices contain 56 PCLKs; the EP3SL110, EP3SL150, EP3SL200, EP3SE80, and EP3SE110 devices contain 88 PCLKs; the EP3SE260 device contains 112 PCLKs, and the EP3SL340 device contains 132 PCLKs. These PCLKs have higher skew compared to GCLK and RCLK networks and can be used instead of general purpose routing to drive signals into and out of the Stratix III device.

Figure 6–5. Periphery Clock Networks (EP3SL50, EP3SL70, and EP3SE50 Devices)

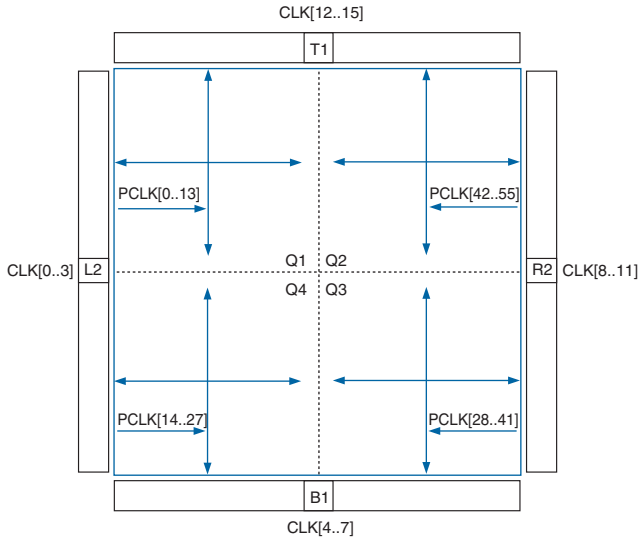


Figure 6–6. Periphery Clock Networks (EP3SL110, EP3SL150, EP3SE80, and EP3SE110 Devices)

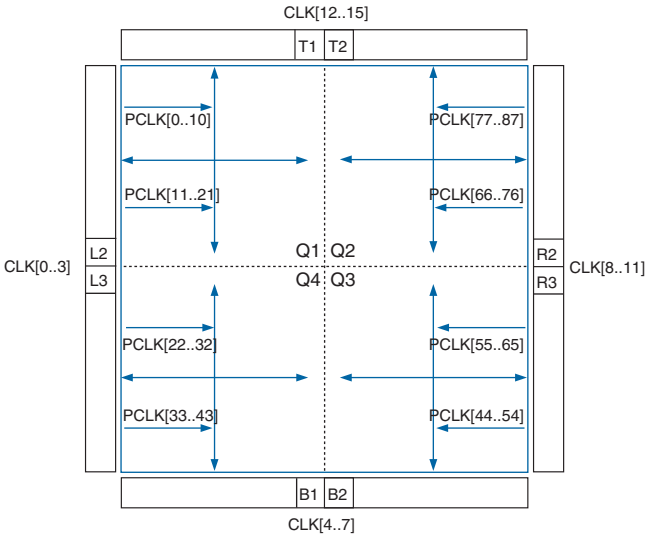


Figure 6–7. Periphery Clock Networks (EP3SL200 Devices)

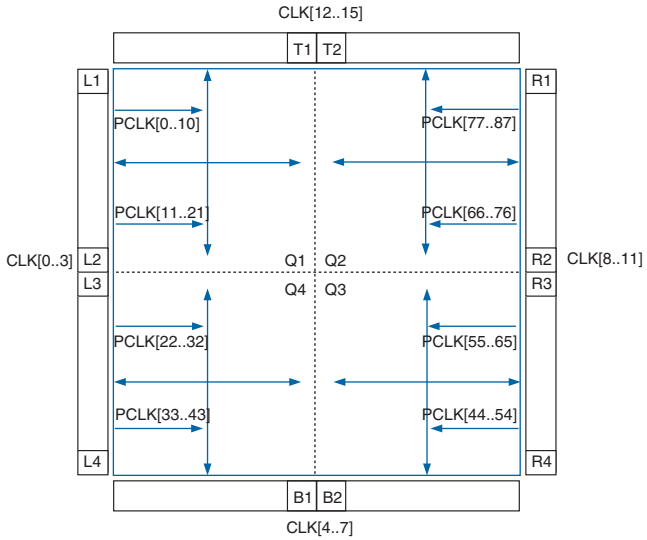


Figure 6–8. Periphery Clock Networks (EP3SE260 Devices)

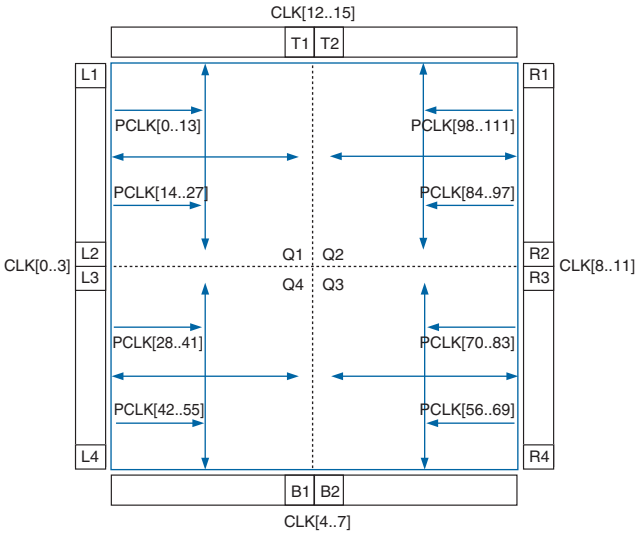
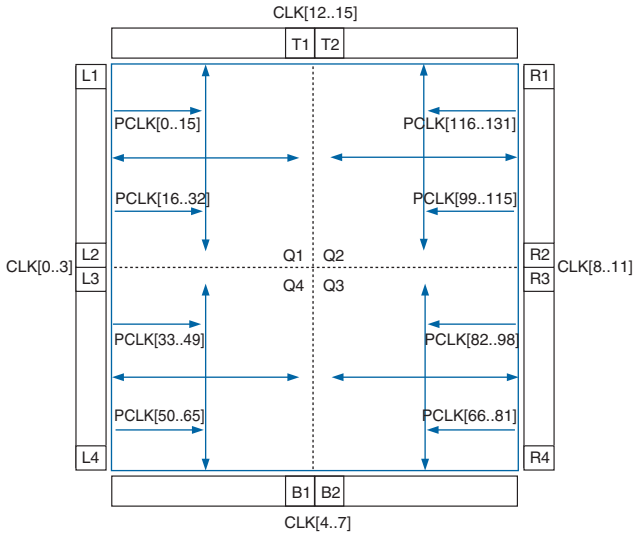


Figure 6–9. Periphery Clock Networks (EP3SL340 Devices)



Clocking Regions

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

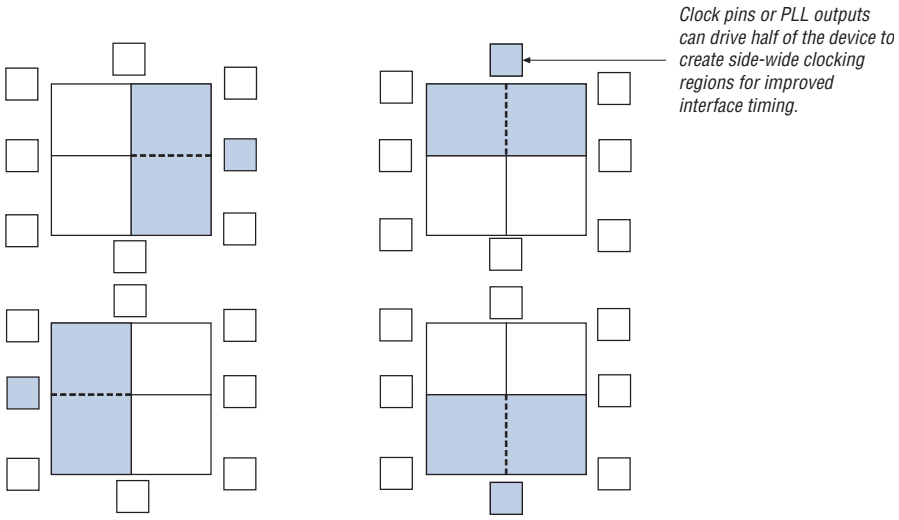
- Entire device clock region
- Regional clock region
- Dual-regional clock region

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

In order to form a regional clock 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 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 regional clock 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 in a regional clock region. Internal logic can also drive a dual-regional clock network. Corner PLL outputs only span one quadrant and hence cannot generate a dual-regional clock network. Figure 6–10 shows the dual-regional clock region.

Figure 6–10. Stratix III Dual-Regional Clock Region



Clock Network Sources

In Stratix III devices, clock input pins, PLL outputs, and internal logic can drive the global and regional clock networks. Refer to Table 6–2 to Table 6–6 for the connectivity between dedicated `CLK[0..15]` pins and the global and regional clock networks.

Dedicated Clock Inputs Pins

The `CLK` pins can either be differential clocks or single-ended clocks. Stratix III devices support 16 differential clock inputs or 32 single-ended clock inputs. You can also use the 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 PCI through global or regional clock networks.

Logic Array Blocks (LABs)

You can also drive each global and regional clock network via LAB-routing to enable internal logic to drive a high fan-out, low-skew signal.

Stratix III device PLLs cannot be driven by internally generated GCLKs or RCLKs. The input clock to the PLL must come from dedicated clock input pins/PLL-fed GCLKs or RCLKs only.

A spine clock is essentially another layer of routing below global/regional and periphery clocks before each clock is connected to the clock routing for each LAB row. The settings for a spine clock are transparent to all users. The Quartus II software takes care of the spine clock routing based on the global/regional and periphery clocks.

PLL Clock Outputs

Stratix III PLLs can drive both GCLK and RCLK networks, as detailed in Table 6–8 on page 6–12 and Table 6–9 on page 6–13.

Table 6–2 lists the connection between the dedicated clock input pins and GCLKs.

Table 6–2. Clock Input Pin Connectivity to Global Clock Networks

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

Table 6–3 lists the connectivity between the dedicated clock input pins and RCLKs in device Quadrant 1. A given clock input pin can drive two adjacent regional clock networks to create a dual-regional clock network.

Table 6–3. Clock Input Pin Connectivity to Regional Clock Networks (Quadrant 1)

Clock Resource	CLK (p/n Pins)															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
RCLK0	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RCLK1	—	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RCLK2	—	—	✓	—	—	—	—	—	—	—	—	—	—	—	—	—
RCLK3	—	—	—	✓	—	—	—	—	—	—	—	—	—	—	—	—
RCLK4	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RCLK5	—	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RCLK54	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
RCLK55	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	—
RCLK56	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	—	—
RCLK57	—	—	—	—	—	—	—	—	—	—	—	—	✓	—	—	—
RCLK58	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
RCLK59	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	—
RCLK60	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	—	—
RCLK61	—	—	—	—	—	—	—	—	—	—	—	—	✓	—	—	—
RCLK62	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
RCLK63	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	—

Table 6–4 lists the connectivity between the dedicated clock input pins and RCLKs in device Quadrant 2. A given clock input pin can drive two adjacent regional clock networks to create a dual-regional clock network.

Table 6–4. Clock Input Pin Connectivity to Regional Clock Networks (Quadrant 2) (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
RCLK38	—	—	—	—	—	—	—	—	—	—	—	✓	—	—	—	—
RCLK39	—	—	—	—	—	—	—	—	—	—	✓	—	—	—	—	—
RCLK40	—	—	—	—	—	—	—	—	—	✓	—	—	—	—	—	—
RCLK41	—	—	—	—	—	—	—	—	✓	—	—	—	—	—	—	—
RCLK42	—	—	—	—	—	—	—	—	—	—	—	✓	—	—	—	—
RCLK43	—	—	—	—	—	—	—	—	—	—	✓	—	—	—	—	—
RCLK44	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
RCLK45	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	—
RCLK46	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	—	—
RCLK47	—	—	—	—	—	—	—	—	—	—	—	—	✓	—	—	—
RCLK48	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
RCLK49	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	—

Table 6–4. Clock Input Pin Connectivity to Regional Clock Networks (Quadrant 2) (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
RCLK50	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	—	—
RCLK51	—	—	—	—	—	—	—	—	—	—	—	—	✓	—	—	—
RCLK52	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
RCLK53	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓	—

Table 6–5 lists the connectivity between the dedicated clock input pins and RCLKs in device Quadrant 3. A given clock input pin can drive two adjacent regional clock networks to create a dual-regional clock network.

Table 6–5. Clock Input Pin Connectivity to Regional Clock Networks (Quadrant 3)

Clock Resource	CLK (p/n Pins)															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
RCLK22	—	—	—	—	—	✓	—	—	—	—	—	—	—	—	—	—
RCLK23	—	—	—	—	✓	—	—	—	—	—	—	—	—	—	—	—
RCLK24	—	—	—	—	—	—	—	✓	—	—	—	—	—	—	—	—
RCLK25	—	—	—	—	—	—	✓	—	—	—	—	—	—	—	—	—
RCLK26	—	—	—	—	—	✓	—	—	—	—	—	—	—	—	—	—
RCLK27	—	—	—	—	✓	—	—	—	—	—	—	—	—	—	—	—
RCLK28	—	—	—	—	—	—	—	✓	—	—	—	—	—	—	—	—
RCLK29	—	—	—	—	—	—	✓	—	—	—	—	—	—	—	—	—
RCLK30	—	—	—	—	—	✓	—	—	—	—	—	—	—	—	—	—
RCLK31	—	—	—	—	✓	—	—	—	—	—	—	—	—	—	—	—
RCLK32	—	—	—	—	—	—	—	—	—	—	—	✓	—	—	—	—
RCLK33	—	—	—	—	—	—	—	—	—	—	✓	—	—	—	—	—
RCLK34	—	—	—	—	—	—	—	—	—	✓	—	—	—	—	—	—
RCLK35	—	—	—	—	—	—	—	—	✓	—	—	—	—	—	—	—
RCLK36	—	—	—	—	—	—	—	—	—	—	—	✓	—	—	—	—
RCLK37	—	—	—	—	—	—	—	—	—	—	✓	—	—	—	—	—

Table 6–6 lists the connectivity between the dedicated clock input pins and RCLKs in device Quadrant 4. A given clock input pin can drive two adjacent regional clock networks to create a dual-regional clock network.

Table 6–6. Clock Input Pin Connectivity to Regional Clock Networks (Quadrant 4)

Clock Resource	CLK (p/n Pins)															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
RCLK6	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RCLK7	—	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RCLK8	—	—	✓	—	—	—	—	—	—	—	—	—	—	—	—	—
RCLK9	—	—	—	✓	—	—	—	—	—	—	—	—	—	—	—	—
RCLK10	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RCLK11	—	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RCLK12	—	—	—	—	—	✓	—	—	—	—	—	—	—	—	—	—
RCLK13	—	—	—	—	✓	—	—	—	—	—	—	—	—	—	—	—
RCLK14	—	—	—	—	—	—	—	✓	—	—	—	—	—	—	—	—
RCLK15	—	—	—	—	—	—	✓	—	—	—	—	—	—	—	—	—
RCLK16	—	—	—	—	—	✓	—	—	—	—	—	—	—	—	—	—
RCLK17	—	—	—	—	✓	—	—	—	—	—	—	—	—	—	—	—
RCLK18	—	—	—	—	—	—	—	✓	—	—	—	—	—	—	—	—
RCLK19	—	—	—	—	—	—	✓	—	—	—	—	—	—	—	—	—
RCLK20	—	—	—	—	—	✓	—	—	—	—	—	—	—	—	—	—
RCLK21	—	—	—	—	✓	—	—	—	—	—	—	—	—	—	—	—

Table 6–7 lists the dedicated clock input pin connectivity to Stratix III device PLLs.

Table 6–7. Stratix III Device PLLs and PLL Clock Pin Drivers (Part 1 of 2) *(Note 1)*

Dedicated Clock Input Pin (CLKp/n pins)	PLL Number											
	L1	L2	L3	L4	B1	B2	R1	R2	R3	R4	T1	T2
CLK0	✓	✓	✓	✓	—	—	—	—	—	—	—	—
CLK1	✓	✓	✓	✓	—	—	—	—	—	—	—	—
CLK2	✓	✓	✓	✓	—	—	—	—	—	—	—	—
CLK3	✓	✓	✓	✓	—	—	—	—	—	—	—	—
CLK4	—	—	—	—	✓	✓	—	—	—	—	—	—
CLK5	—	—	—	—	✓	✓	—	—	—	—	—	—
CLK6	—	—	—	—	✓	✓	—	—	—	—	—	—
CLK7	—	—	—	—	✓	✓	—	—	—	—	—	—
CLK8	—	—	—	—	—	—	✓	✓	✓	✓	—	—
CLK9	—	—	—	—	—	—	✓	✓	✓	✓	—	—
CLK10	—	—	—	—	—	—	✓	✓	✓	✓	—	—
CLK11	—	—	—	—	—	—	✓	✓	✓	✓	—	—
CLK12	—	—	—	—	—	—	—	—	—	—	✓	✓

Table 6–7. Stratix III Device PLLs and PLL Clock Pin Drivers (Part 2 of 2) *(Note 1)*

Dedicated Clock Input Pin (CLKp/n pins)	PLL Number											
	L1	L2	L3	L4	B1	B2	R1	R2	R3	R4	T1	T2
CLK13	—	—	—	—	—	—	—	—	—	—	✓	✓
CLK14	—	—	—	—	—	—	—	—	—	—	✓	✓
CLK15	—	—	—	—	—	—	—	—	—	—	✓	✓
PLL_L1_CLKp *(2)*	✓	—	—	—	—	—	—	—	—	—	—	—
PLL_L1_CLKn *(2)*, *(3)*	✓	—	—	—	—	—	—	—	—	—	—	—
PLL_L4_CLKp *(2)*	—	—	—	✓	—	—	—	—	—	—	—	—
PLL_L4_CLKn *(2)*, *(3)*	—	—	—	✓	—	—	—	—	—	—	—	—
PLL_R1_CLKp *(2)*	—	—	—	—	—	—	✓	—	—	—	—	—
PLL_R1_CLKn *(2)*, *(3)*	—	—	—	—	—	—	✓	—	—	—	—	—
PLL_R4_CLKp *(2)*	—	—	—	—	—	—	—	—	—	✓	—	—
PLL_R4_CLKn *(2)*, *(3)*	—	—	—	—	—	—	—	—	—	✓	—	—

Notes to Table 6–7:

(1) For compensated PLLs input, only the dedicated CLK pins in the same I/O bank as the PLL used are compensated inputs.

(2) If both PLL_<L1/L4/R1/R4>_CLKp and PLL_<L1/L4/R1/R4>_CLKn pins are not used as a pair of differential clock pins, they can be used independently as single-ended clock input pins.

(3) For single-ended clock input, CLKn pins use the global network to drive the PLLs.

Clock Output Connections

PLLs in Stratix III devices can drive up to 20 regional clock networks and four global clock networks. Refer to Table 6–8 for Stratix III PLL connectivity to GCLK networks. The Quartus II software automatically assigns PLL clock outputs to regional or global clock networks.

Table 6–8 lists how the PLL clock outputs connect to GCLK networks.

Table 6–8. PLL Connectivity to GCLKs on Stratix III Devices (Part 1 of 2) *(Note 1)*

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

Table 6–8. PLL Connectivity to GCLKs on Stratix III Devices (Part 2 of 2) *(Note 1)*

Clock Network	PLL Number											
	L1	L2	L3	L4	B1	B2	R1	R2	R3	R4	T1	T2
GCLK12	—	—	—	—	—	—	—	—	—	—	✓	✓
GCLK13	—	—	—	—	—	—	—	—	—	—	✓	✓
GCLK14	—	—	—	—	—	—	—	—	—	—	✓	✓
GCLK15	—	—	—	—	—	—	—	—	—	—	✓	✓

Note to Table 6–8:

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

Table 6–9 lists how the PLL clock outputs connect to RCLK networks.

Table 6–9. Regional Clock Outputs From PLLs on Stratix III Devices *(Note 1)*

Clock Resource	PLL Number											
	L1	L2	L3	L4	B1	B2	R1	R2	R3	R4	T1	T2
RCLK[0..11]	—	✓	✓	—	—	—	—	—	—	—	—	—
RCLK[12..31]	—	—	—	—	✓	✓	—	—	—	—	—	—
RCLK[32..43]	—	—	—	—	—	—	—	✓	✓	—	—	—
RCLK[44..63]	—	—	—	—	—	—	—	—	—	—	✓	✓
RCLK[64..69]	—	—	—	✓	—	—	—	—	—	—	—	—
RCLK[70..75]	—	—	—	—	—	—	—	—	—	✓	—	—
RCLK[76..81]	—	—	—	—	—	—	✓	—	—	—	—	—
RCLK[82..87]	✓	—	—	—	—	—	—	—	—	—	—	—

Note to Table 6–9:

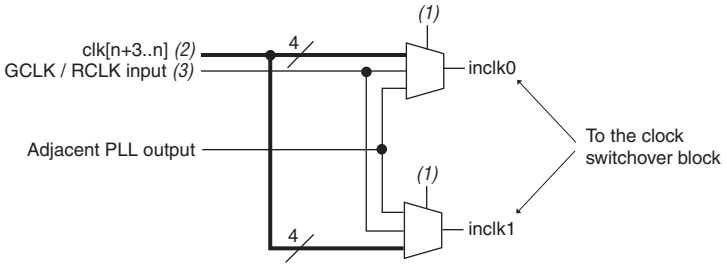
(1) All PLL counter outputs can drive RCLK networks.

Clock Source Control for PLLs

The clock input to Stratix III 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 6–11; the corresponding clock input sources to Left/Right PLLs (L1, L4, R1, and R4) are shown in Figure 6–12.

The multiplexer select lines are set in the configuration file (SRAM object file [**.sof**] or programmer object file [**.pof**]) only. Once 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 6–11. Clock Input Multiplexer Logic for L2, L3, T1, T2, B1, B2, R2, and R3 PLLs



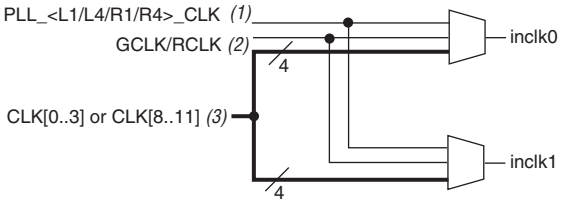
Notes to Figure 6–11:

(1) The input clock multiplexing is controlled through a configuration file (**.sof** or **.pof**) only and cannot be dynamically controlled in user mode operation.

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

Figure 6–12. Clock Input Multiplexer Logic for L1, L4, R1, and R4 PLLs



Notes to Figure 6–12:

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

(2) The global (GCLK) or regional (RCLK) clock input can be driven by an output from another PLL, a pin-driven global or regional clock, or through a clock control block provided the clock control block is fed by an output from another PLL or a pin driven dedicated global or regional clock. 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.

Clock Control Block

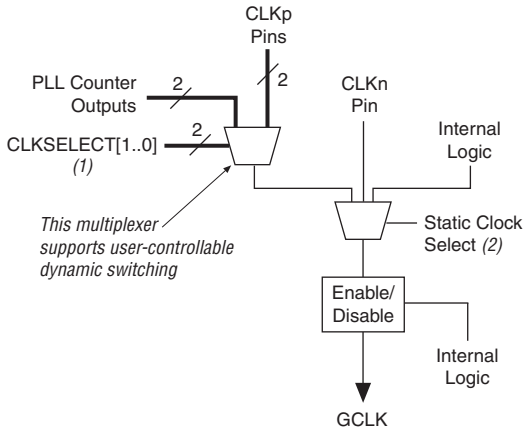
Every global and regional clock network has its own clock control block. The control block provides the following features:

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

You can select the clock source for the global clock select block either statically or dynamically. You can either 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 either select two PLL outputs (such as `CLK0` or `CLK1`), or a combination of clock pins or PLL outputs.

Figure 6–13 and Figure 6–14 show the global clock and regional clock select blocks, respectively.

Figure 6–13. Stratix III Global Clock Control Block

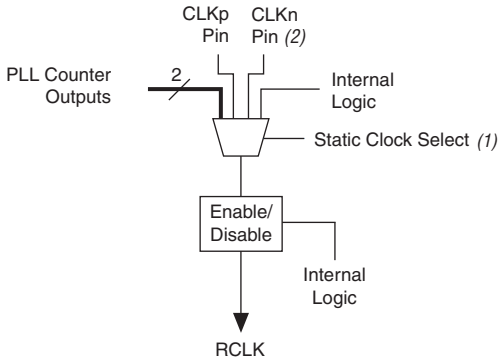


Notes to Figure 6–13:

(1) These clock select signals can only be dynamically controlled through internal logic when the device is operating in user mode.

(2) These clock select signals can only be set through a configuration file (**.sof** or **.pof**) and cannot be dynamically controlled during user mode operation.

Figure 6–14. Regional Clock Control Block



Notes to Figure 6–14:

(1) This clock select signal can only be statically controlled through a configuration file (**.sof** or **.pof**) and cannot be dynamically controlled during user mode operation.

(2) `CLKn` pin is not a dedicated clock input when it is use as a single-ended PLL clock input and it is not fully compensated.

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

The Stratix III clock networks can be powered down by both static and dynamic approaches. When a clock net is powered down, all the logic fed by the clock net is in an off-state, thereby reducing the overall power consumption of the device. The unused global and regional clock networks are automatically powered down through configuration bit settings in the configuration file (**.sof** or **.pof**) generated by the

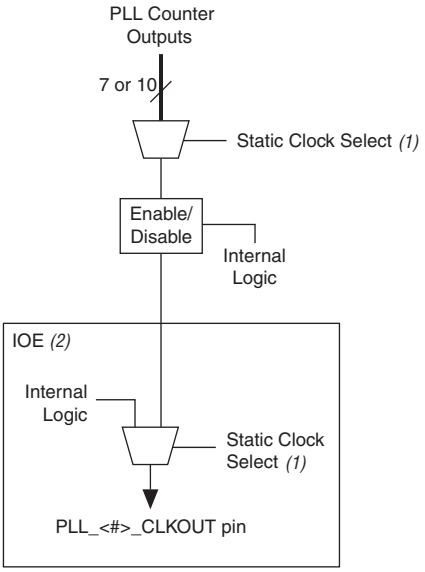
Quartus II software. The dynamic clock enable or disable feature allows the internal logic to control power-up or power-down synchronously on GCLK and RCLK networks, including dual-regional clock regions. This function is independent of the PLL and is applied directly on the clock network, as shown in Figure 6–13 and Figure 6–14.

You can set the input clock sources and the `clkena` signals for the global and regional clock 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.

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

Figure 6–15 shows the external PLL output clock control block.

Figure 6–15. Stratix III External PLL Output Clock Control Block



Notes to Figure 6–15:

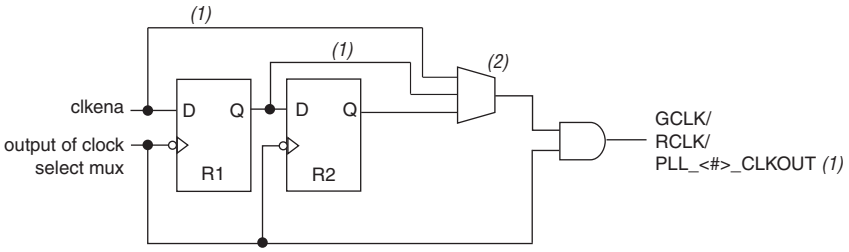
(1) This clock select signal can only be set through a configuration file (**.sof** or **.pof**) and cannot be dynamically controlled during user mode operation.

(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 6–16 shows how the clock enable/disable circuit of the clock control block is implemented in Stratix III devices.

Figure 6–16. clkena Implementation



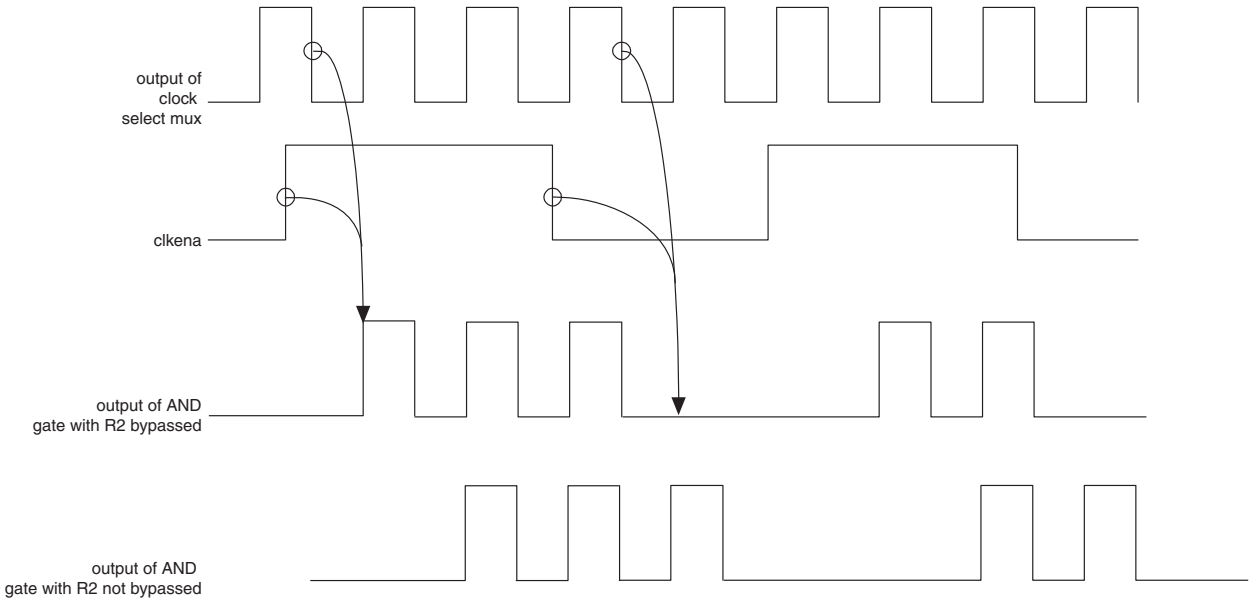
Notes to Figure 6–16:

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

In Stratix III 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 a PLL is not being used. You can also use the `clkena` signals to control the dedicated external clocks from the PLLs. Figure 6–17 shows the waveform example for a clock output enable. `clkena` is synchronous to the falling edge of the clock output.

Stratix III devices also have an additional metastability register that aids in asynchronous enable/disable of the GCLK and RCLK networks. This register can be optionally bypassed in the Quartus II software.

Figure 6–17. clkena Signals



Note to Figure 6–17:

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

The PLL can remain locked independently 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 overshoot during resynchronization.

PLLs in Stratix III Devices

Stratix III 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, respectively.

Table 6–10 lists the PLLs available in the Stratix III device family.

Table 6–10. Stratix III Device PLL Availability

Device	L1	L2	L3	L4	T1	T2	B1	B2	R1	R2	R3	R4
EP3SL50	—	✓	—	—	✓	—	✓	—	—	✓	—	—
EP3SL70	—	✓	—	—	✓	—	✓	—	—	✓	—	—
EP3SL110 *(1)*	—	✓	✓	—	✓	✓	✓	✓	—	✓	✓	—
EP3SL150 *(1)*	—	✓	✓	—	✓	✓	✓	✓	—	✓	✓	—
EP3SL200 *(1)*, *(2)*	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
EP3SL340 *(2)*	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
EP3SE50	—	✓	—	—	✓	—	✓	—	—	✓	—	—
EP3SE80 *(1)*	—	✓	✓	—	✓	✓	✓	✓	—	✓	✓	—
EP3SE110 *(1)*	—	✓	✓	—	✓	✓	✓	✓	—	✓	✓	—
EP3SE260 *(2)*	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Notes to Table 6–10:

(1) PLLs T2, B2, L3, and R3 are not available in the F780 package.

(2) PLLs L1, L4, R1, and R4 are not available in the H780, F1152, and H1152 packages.

All Stratix III PLLs have the same core analog structure with only minor differences in the features that are supported. Table 6–11 lists the features of the Top/Bottom and Left/Right PLLs in Stratix III devices.

Table 6–11. Stratix III PLL Features

Feature	Stratix III Top/Bottom PLLs	Stratix III 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
Clock input pins	4 single-ended or 2 differential pin pairs	4 single-ended or 2 differential pin pairs
External feedback input pin	Single-ended or differential	Single-ended only
Spread-spectrum input clock tracking	Yes *(1)*	Yes *(1)*
PLL cascading	Through GCLK and RCLK and dedicated path between adjacent PLLs	Through GCLK and RCLK and dedicated path between adjacent PLLs *(2)*
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 DPA clock	No	Yes
Phase shift resolution	Down to 96.125 ps *(3)*	Down to 96.125 ps *(3)*
Programmable duty cycle	Yes	Yes
Output counter cascading	Yes	Yes
Input clock switchover	Yes	Yes

Notes to Table 6–11:

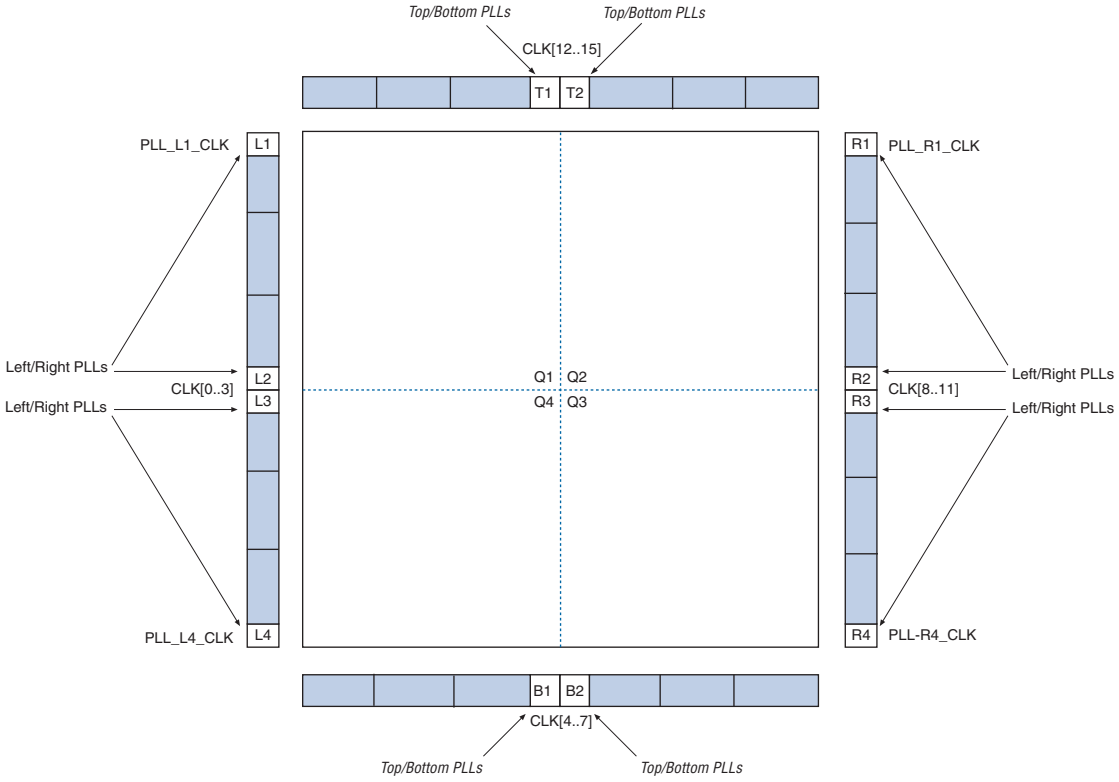
(1) Provided input clock jitter is within input jitter tolerance specifications.

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

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

Figure 6–18 shows the location of the PLLs in Stratix III devices.

Figure 6–18. Stratix III PLL Locations



Stratix III PLL Hardware Overview

Stratix III devices contain up to 12 PLLs with advanced clock management features. The main 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.

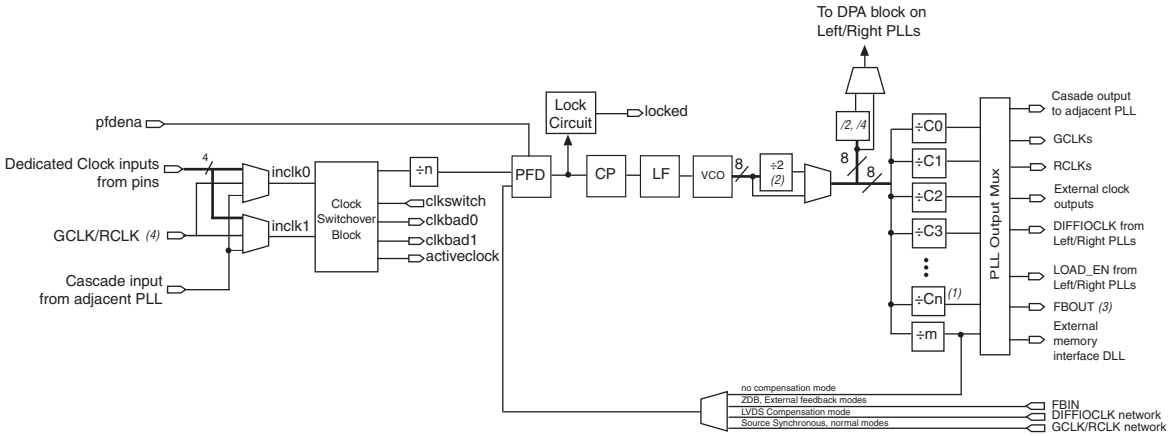
Stratix III PLLs align the rising edge of the input reference clock to a feedback clock using the phase-frequency detector (PFD). The falling edges are determined by the duty-cycle specifications. The PFD produces an up or down signal that determines whether the voltage-controlled oscillator (VCO) needs to 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, then 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 overshoot, 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 Left/Right PLLs can feed seven post-scale counters (C[0..6]), while the corresponding VCO output from Top/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 6–19 shows a simplified block diagram of the major components of the Stratix III PLL.

Figure 6–19. Stratix III PLL Block Diagram



Notes to Figure 6–19:

(1) The number of post-scale counters is 7 for Left/Right PLLs and 10 for Top/Bottom PLLs.

(2) This is the VCO post-scale counter K. If the design enables this ÷2 counter, the device can use a VCO frequency range of 300 to 650 MHz. The VCO frequency reported by the Quartus II software is divided by the post-scale counter K.

(3) The FBOUT port is fed by the M counter in Stratix III PLLs.

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

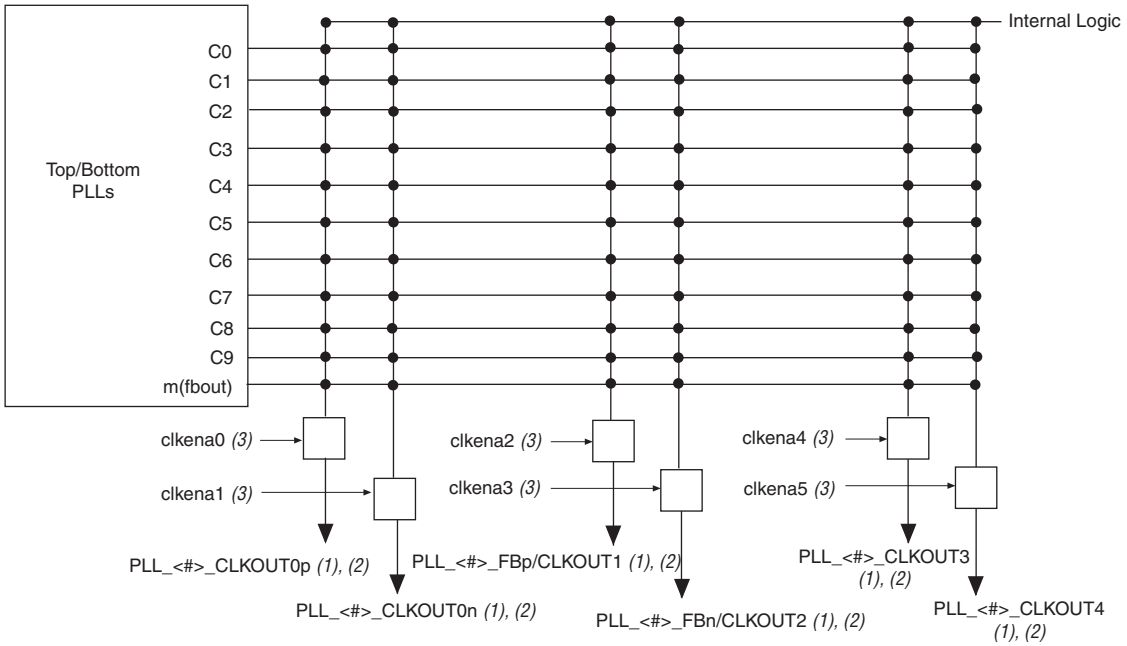
PLL Clock I/O Pins

Each Top/Bottom PLL supports six clock I/O pins, organized as three pairs of pins:

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

Figure 6–20 shows the clock I/O pins associated with Top/Bottom PLLs.

Figure 6–20. External Clock Outputs for Top/Bottom PLLs



Notes to Figure 6–20:

(1) These clock output pins can be fed by any one of the C[9..0], m counters.

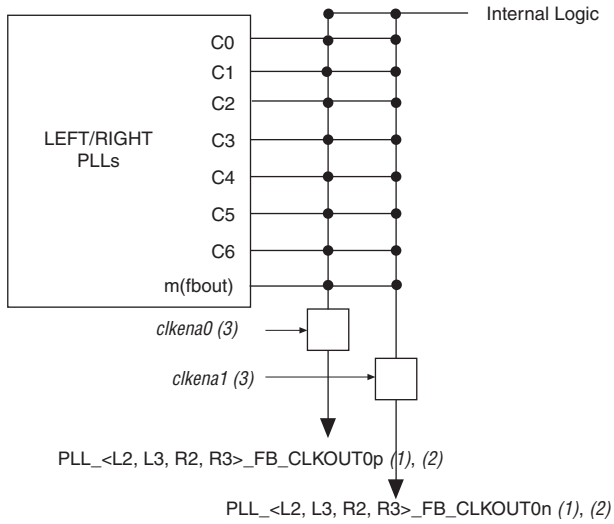
(2) The CLKOUT0p and CLKOUT0n pins can be either single-ended or differential clock outputs. CLKOUT1 and CLKOUT2 pins are dual-purpose I/O pins that can be used as two single-ended outputs, one differential external feedback input pin pair or one single-ended external feedback input pin (CLKOUT1 only). 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 Top/Bottom PLLs and C[6..0] on Left/Right PLLs) or the M counter can feed the dedicated external clock outputs, as shown in Figure 6–20 and Figure 6–21. Therefore, one counter or frequency can drive all output pins available from a given PLL.

Each Left/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. Hence, Left/Right PLLs only support external feedback mode for single-ended I/O standards.

Figure 6–21. External Clock Outputs for Left/Right PLLs



Notes to Figure 6–21:

(1) These clock output pins can be fed by any one of the `C[6..0]`, m counters.

(2) The `CLKOUT0p` and `CLKOUT0n` pins are dual-purpose I/O pins that can be used 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-degrees out-of-phase. The Quartus II software places the NOT gate in the design into the IOE to implement 180-degrees 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 HSTL, and differential SSTL.

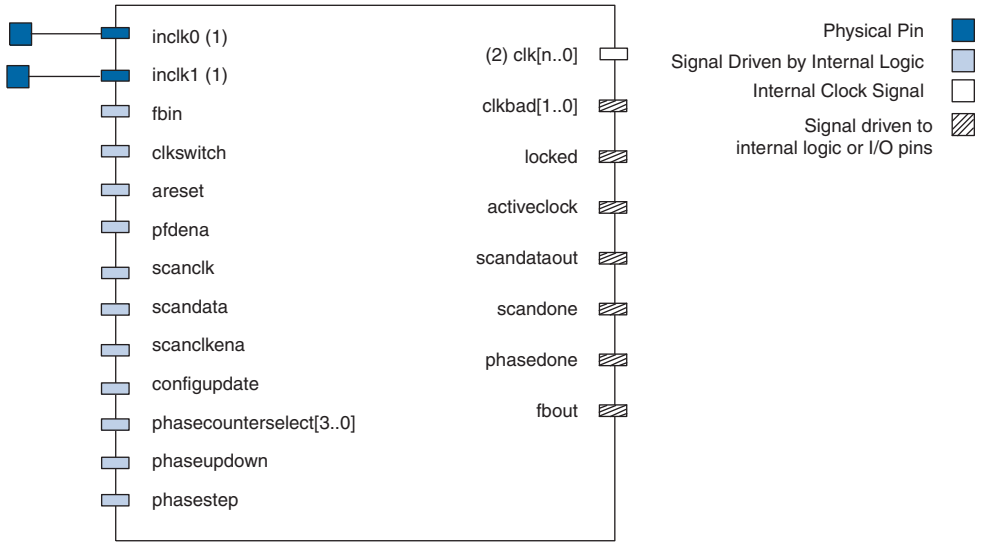
To determine which I/O standards are supported by the PLL clock input and output pins, refer to the *Stratix III Device I/O Features* chapter.

Stratix III PLLs can also drive out to any regular I/O pin through the global or regional clock network. You can use the external clock output pins as user I/O pins if external PLL clocking is not needed.

Stratix III PLL Software Overview

Stratix III PLLs are enabled in the Quartus II software by using the ALTPLL megafunction. Figure 6–22 shows the Stratix III PLL ports as they are named in the ALTPLL megafunction of the Quartus II software.

Figure 6–22. Stratix III PLL Ports



Notes to Figure 6–22:

(1) You can feed the `inclk0` or `inclk1` clock input from any one of four dedicated clock pins located on the same side of the device as the PLL.

(2) You can drive to global or regional clock networks or dedicated external clock output pins. n = 6 for Left/Right PLLs and n = 9 for Top/Bottom PLLs.

Table 6–12 lists the PLL input signals for Stratix III devices.

Table 6–12. PLL Input Signals (Part 1 of 2)

Port	Description	Source	Destination
`inclk0`	Input clock to the PLL	Dedicated pin, adjacent PLL, GCLK, or RCLK network	`N` counter
`inclk1`	Input clock to the PLL	Dedicated pin, adjacent PLL, GCLK, or RCLK network	`N` counter
`fbin`	Compensation feedback input to the PLL. Share the same clock spines used by GCLK/RCLKs.	Pin LVSDCLK	PFD
`clkswitch`	Switchover signal used to initiate clock switchover asynchronously. When used in manual switchover, `clkswitch` is used as a select signal between `inclk0` and `inclk1`. If `clkswitch = 0`, `inclk0` is selected. If `clkswitch = 1`, `inclk1` is selected. Both `inclk0` and `inclk1` must be switched in order for manual switchover to function.	Logic array or I/O pin	Clock switchover circuit

Table 6–12. PLL Input Signals (Part 2 of 2)

Port	Description	Source	Destination
areset	Signal used to reset the PLL which resynchronizes all the counter outputs. Active high	Logic array	General PLL control signal
pfdena	Enables the outputs from the phase frequency detector. Active high	Logic array	PFD
scanclk	Serial clock signal for the real-time PLL reconfiguration feature.	Logic array	Reconfiguration circuit
scandata	Serial input data stream for the real-time PLL reconfiguration feature.	Logic array	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 pins	PLL reconfiguration circuit
phasecounterselect[3:0]	Selects corresponding PLL counter for dynamic phase shift	Logic array or I/O pins	PLL reconfiguration circuit
phaseupdown	Selects dynamic phase shift direction; 1 = UP; 0 = DOWN	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 6–13 lists the PLL output signals for Stratix III devices.

Table 6–13. PLL Output Signals (Part 1 of 2)

Port	Description	Source	Destination
clk[9..0] for Top/Bottom PLLs clk[6..0] for Left/Right PLLs	PLL output counters driving regional, global, or external clocks.	PLL counter	Internal or external clock
clkbad[1..0]	Signals indicating which reference clock is no longer toggling. clkbad1 indicates inclk1 status, clkbad0 indicates inclk0 status. 0 = good; 1 = bad	PLL switchover circuit	Logic array
locked	Lock output from lock detect circuit. Active high	PLL lock detect	Logic array
activeclock	Signal to indicate which clock (0 = inclk0 or 1 = inclk1) is driving the PLL. If this signal is low, inclk0 drives the PLL. If this signal is high, inclk1 drives the PLL.	PLL clock multiplexer	Logic array
scandataout	Output of the last shift register in the scan chain.	PLL scan chain	Logic array

Table 6–13. PLL Output Signals (Part 2 of 2)

Port	Description	Source	Destination
scandone	Signal indicating when the PLL has completed reconfiguration. One-to-0 transition indicates that the PLL has been reconfigured.	PLL scan chain	Logic array
phasedone	When asserted it indicates that the phase reconfiguration is complete and the PLL is ready to act on a possible second reconfiguration. Asserts based on internal PLL timing. De-asserts on rising edge of SCANCLK.	PLL scan chain	Logic array
fbout	Output of m counter. Used for clock delay compensation.	M counter	Logic array

Clock Feedback Modes

Stratix III PLLs support up to six different clock feedback modes. Each mode allows clock multiplication and division, phase shifting, and programmable duty cycle. Table 6–14 lists the clock feedback modes supported by Stratix III PLLs.

Table 6–14. Clock Feedback Mode Availability

Clock Feedback Mode	Availability	
	Top/Bottom PLLs	Left/Right PLLs
Source-synchronous mode	Yes	Yes
No-compensation mode	Yes	Yes
Normal mode	Yes	Yes
Zero-delay buffer (ZDB) mode	Yes	Yes
External feedback mode *(2)*	Yes *(3)*	Yes *(1)*
LVDS compensation	No	Yes

Notes to Table 6–14:

(1) External feedback mode supported for single-ended inputs and outputs only on Left/Right PLLs.

(2) High-bandwidth PLL settings are not supported in external feedback mode. Select a "low" or "medium" PLL bandwidth in the ALTPLL MegaWizard™ Plug-in Manager when using PLLs in external feedback mode.

(3) Differential HSTL and SSTL I/O standards are not supported in Top/Bottom PLLs for external feedback mode.

The input and output delays are fully compensated by a PLL only when they are using the dedicated clock input pins associated with a given PLL as the clock source. Input and output delays are not compensated when cascading two adjacent top or bottom PLLs even if they are using dedicated routing for cascading. 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 four pins: CLK12, CLK13, CLK14, or CLK15. When an RCLK or GCLK network drives the PLL, the input and output delays may not be fully compensated in the Quartus II software.

Source Synchronous Mode

If the data and clock signals 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 6–23 shows an example waveform of the clock and data in this mode. This mode is recommended for source-synchronous data transfers. Data and clock signals at the IOE experience similar buffer delays when you use the same I/O standard.

Figure 6–23. Phase Relationship Between Clock and Data in Source-Synchronous Mode

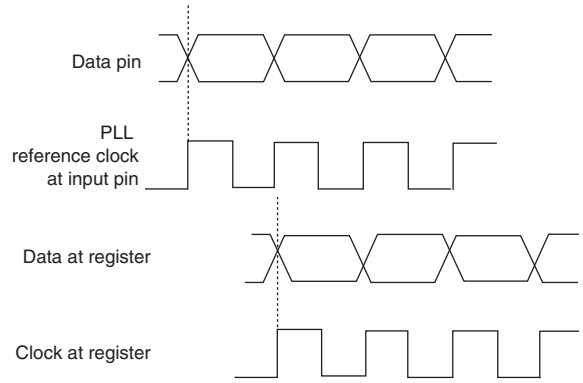
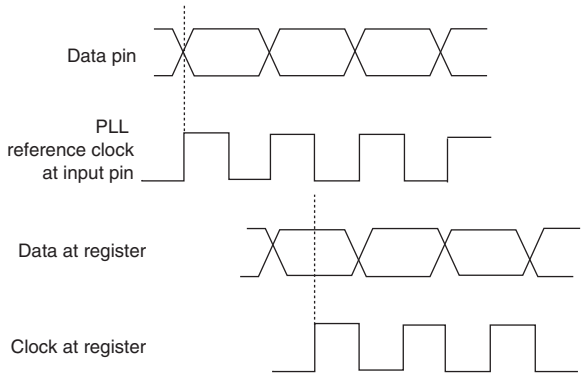


Figure 6–24 shows an example waveform of the clock and data in the LVDS mode.

Figure 6–24. Phase Relationship Between Clock and Data LVDS Modes



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

Set the input pin to register delay chain within the IOE to zero in the Quartus II software for all data pins clocked by a source-synchronous mode PLL. Also, all data pins must use the **PLL COMPENSATED** logic option in the Quartus II software.

Source-Synchronous Mode for LVDS Compensation

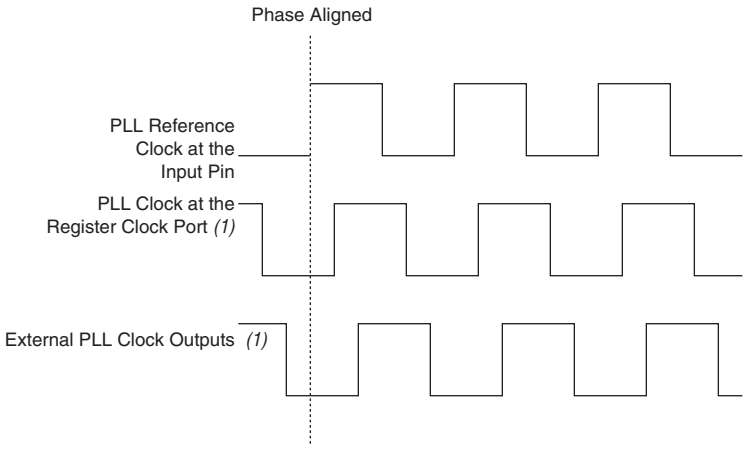
The goal of this mode is to maintain the same data and clock timing relationship seen at the pins at the internal SERDES capture register, except that the clock is inverted (180-degree phase shift). Thus, this mode ideally compensates for the delay of the LVDS clock network plus any difference in delay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

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

No-Compensation Mode

In the 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 6–25 shows an example waveform of the PLL clocks' phase relationship in this mode.

Figure 6–25. Phase Relationship Between PLL Clocks in No Compensation Mode



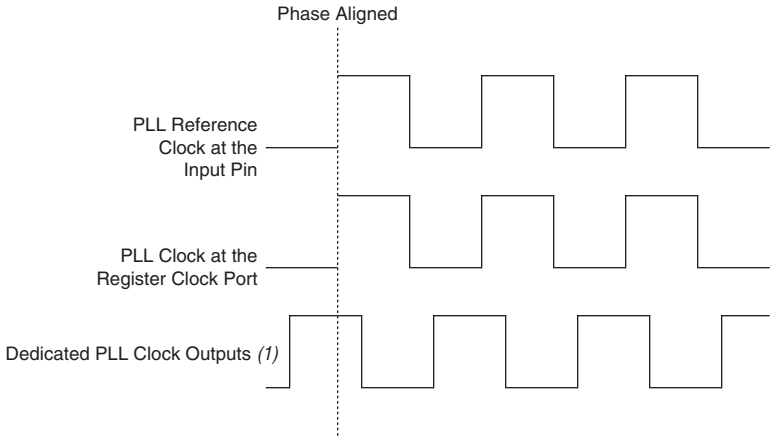
Note to Figure 6–25:

(1) The PLL clock outputs will lag the PLL input clocks, depending on routing 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 6–26 shows an example waveform of the PLL clocks' phase relationship in this mode.

Figure 6–26. Phase Relationship Between PLL Clocks in Normal Mode



Note to Figure 6–26:

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

Zero-Delay Buffer Mode

In zero-delay buffer (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 in order to guarantee clock alignment at the input and output pins. This mode is supported on all Stratix III PLLs.

When using Stratix III PLLs in ZDB mode, along with single-ended I/O standards, to ensure phase alignment between the clock input pin (CLK) 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 hence compensate for, the output delay from the clock output port of the PLL to the external clock output pin. Figure 6–27 shows ZDB mode implementation in Stratix III PLLs. You cannot use differential I/O standards on the PLL clock input or output pins when using ZDB mode.

To avoid reflection, do not place a board trace on the bi-directional I/O pins.

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

Figure 6–27. Zero-Delay Buffer Mode in Stratix III PLLs

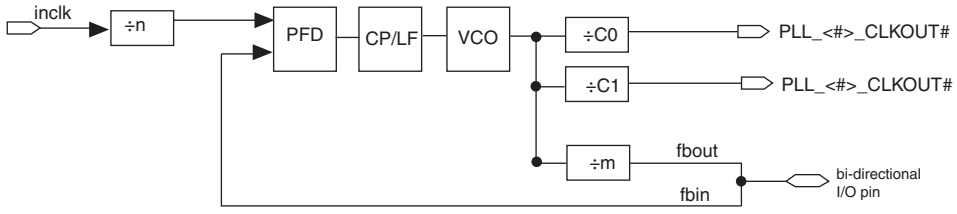
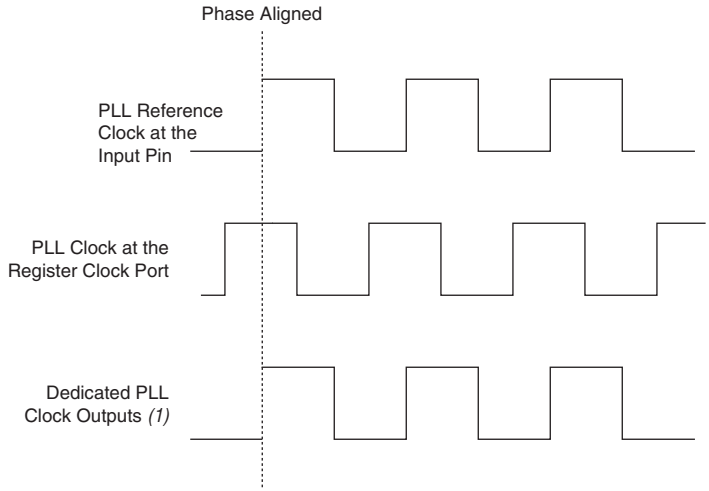


Figure 6–28 shows an example waveform of the PLL clocks' phase relationship in ZDB mode.

Figure 6–28. Phase Relationship Between PLL Clocks in Zero Delay Buffer Mode



Note to Figure 6–28:

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

External Feedback Mode

In external feedback (EFB) mode, the external feedback input pin (`fbin`) is phase-aligned with the clock input pin, as shown in Figure 6–30. Aligning these clocks allows you to remove clock delay and skew between devices. This mode is supported on all Stratix III PLLs.

In this 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, you can use one of the dual-purpose external clock outputs as the `fbin` input pin in EFB mode.

When using this mode, you must use the same I/O standard on the input clock, feedback input, and output clocks. Left/Right PLLs support EFB mode when using single-ended I/O standards only. Figure 6–29 shows the EFB mode implementation in Stratix III devices.

High-bandwidth PLL settings are not supported in external feedback mode. Select a "low" or "medium" PLL bandwidth in the ALTPLL MegaWizard Plug-In Manager when using PLLs in external feedback mode.

Figure 6–29. External Feedback Mode in Stratix III Devices

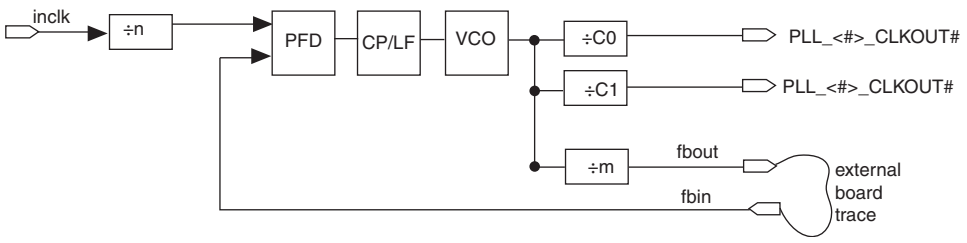
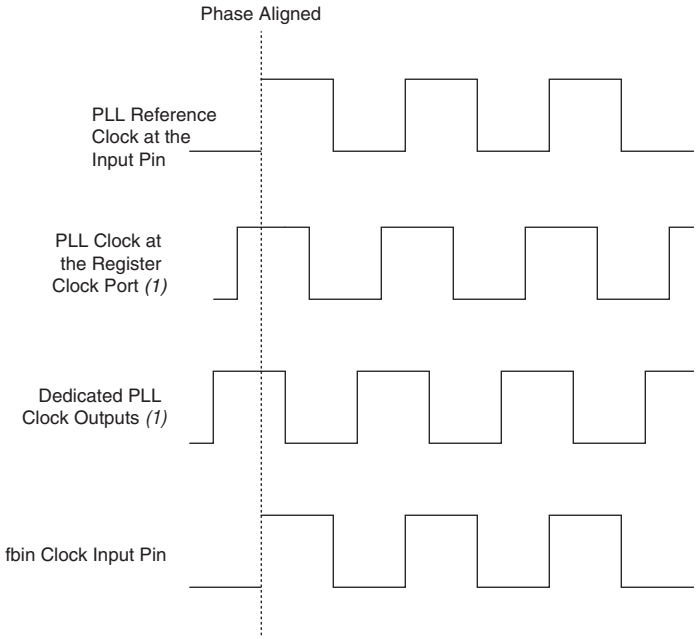


Figure 6–30 shows an example waveform of the phase relationship between PLL clocks in EFB mode.

Figure 6–30. Phase Relationship Between PLL Clocks in External-Feedback Mode



Note to Figure 6–30:

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

Clock Multiplication and Division

Each Stratix III 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 output frequencies required from one PLL are 33 and 66 MHz, then 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.

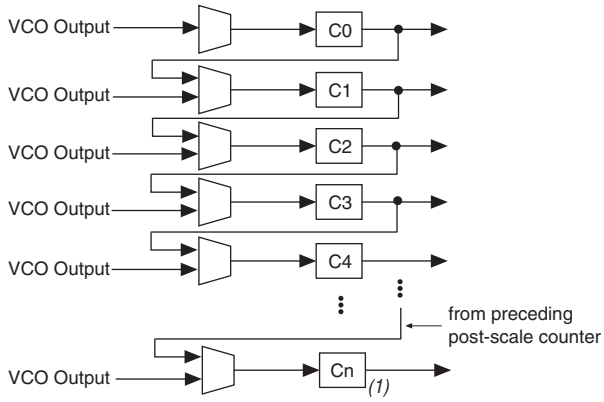
Each PLL has one pre-scale counter, `n`, and one multiply counter, `m`, with a range of 1 to 512 for both `m` and `n`. The `n` counter does not use duty-cycle control because the only purpose of this counter is to calculate frequency division. There are seven generic post-scale counters per Left/Right PLL and ten post-scale counters per Top/Bottom PLL that can feed 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

The Stratix III 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 6–31.

Figure 6–31. Counter Cascading



Note to Figure 6–31:

(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, then the cascaded value is C0*C1 = 800.

Post-scale counter cascading is set in the configuration file. It cannot be done 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. The Quartus II software uses the frequency input and the required multiply or divide rate to determine the duty cycle choices. The post-scale counter value determines the precision of the duty cycle. The precision is defined by 50% divided by the post-scale counter value. For example, if the C0 counter is 10, then steps of 5% are possible for duty-cycle choices between 5% to 90%.

If the PLL is in external feedback mode, you must 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.

PLL Control Signals

You can use the following three signals to observe and control the 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 the PFD, the VCO is free running and the PLL output drifts. The PLL output jitter may not meet the datasheet specifications. The lock signal cannot be used as an indicator when the PFD is disabled.

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 will resynchronize to its input as it re-locks.

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

- PLL reconfiguration or clock switchover is enabled in the design.

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

locked

The lock signal is an asynchronous output of the PLL. The locked 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 if the feedback clock has locked onto the reference clock both in phase and frequency.

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

Clock Switchover

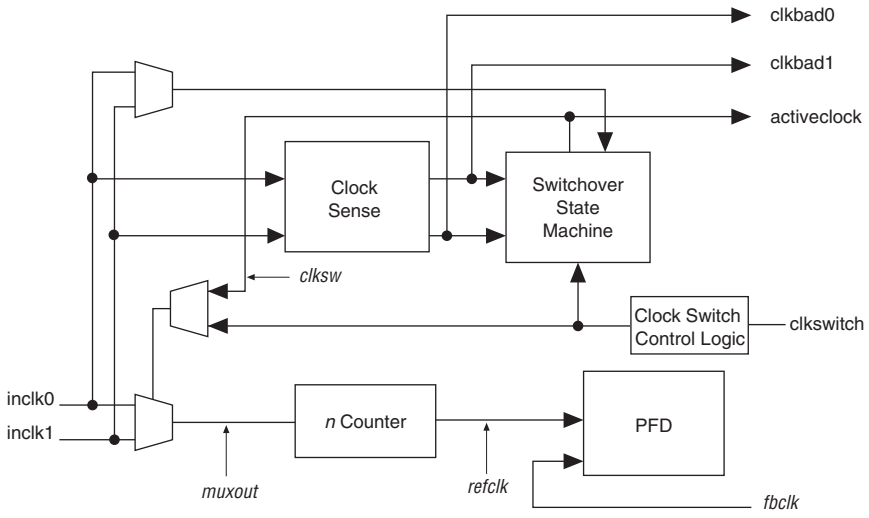
The clock switchover feature allows the PLL to switch between two reference input clocks. Use this feature for clock redundancy or for a dual-clock domain application such as in a system that turns on the redundant clock if the previous clock stops running. The design can perform clock switchover automatically,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

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

- Automatic switchover—The clock sense circuit monitors the current reference clock and if it stops toggling, automatically switches to the other clock `inclk0` or `inclk1`.
- Manual clock switchover—Clock switchover is controlled via the `clkswitch` signal in this mode. 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 Modes 1 and 2. When the `clkswitch` signal goes high, it overrides automatic clock switchover mode.

Stratix III device PLLs support a fully configurable clock switchover capability. Figure 6–32 shows the block diagram of the 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 6–32. 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 6–32. In this case, `inclk1` becomes the reference clock for the PLL. When using the automatic switchover mode, you can switch back and forth between `inclk0` and `inclk1` clocks any number of times, when one of the two clocks fails and the other clock is available.

When using the 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 will not be initiated and the `clkbad[0..1]` signals will not be valid. Also, if both clock inputs are not the same frequency, but their period difference is within 100%, the clock sense block will detect when a clock stops toggling, but the PLL may lose lock after the switchover is completed and need time to re-lock.

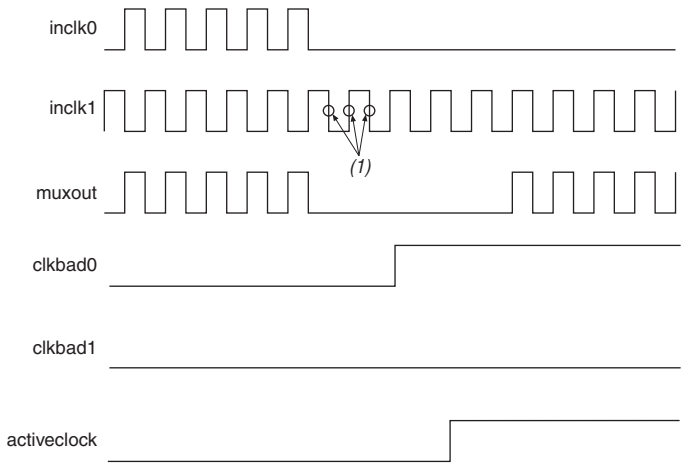
Altera recommends resetting the PLL using the `areset` signal to maintain the phase relationships between the PLL input and output clocks when using clock switchover.

When using 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 6–33 shows an example waveform of the switchover feature when using the automatic switchover mode. In this example, the `inclk0` signal remains low. After the `inclk0` signal remains 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 `clksw` signal to switch to the backup clock, `inclk1`.

Figure 6–33. Automatic Switchover Upon Loss of Clock Detection



Note to Figure 6–33:

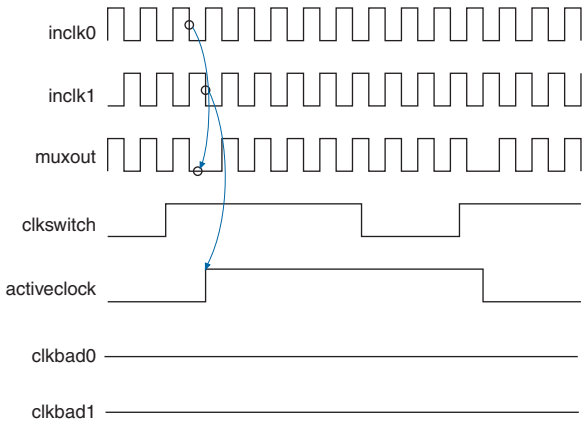
(1) Switchover is enabled on the falling edge of `inclk0` or `inclk1`, depending on which clock is available. In this figure, switchover is enabled on the falling edge of `inclk1`.

Manual Override

In the 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 the switchover using clkswitch because the automatic clock-sense circuitry cannot monitor clock input (inclk0, 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 should 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 1,300 MHz. The ALTPLL MegaWizard Plug-in Manager notifies users if a given combination of inclk0 and inclk1 frequencies cannot meet this requirement. In the Quartus II software, the VCO value reported is divided by the post scale counter (K).

Figure 6–34 shows an example of a waveform illustrating the switchover feature when 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 any 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 6–34. Clock Switchover Using the clkswitch (Manual) Control *(Note 1)*



Note to Figure 6–34:

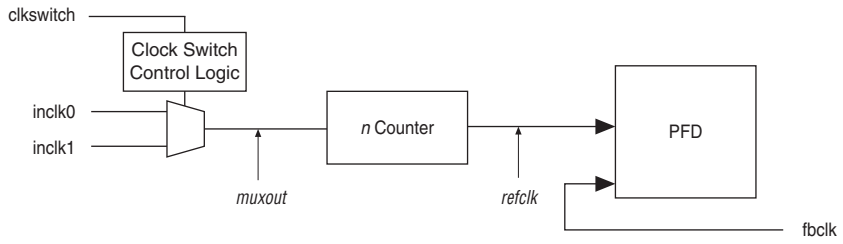
(1) Both inclk0 and inclk1 must be running when the clkswitch signal goes high to initiate a manual clock switchover event.

In this mode, the activeclock signal mirrors the clkswitch signal. As both clocks are still functional during the manual switch, neither clkbad signal goes high. Since 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/inclk1 cycles. Take note that manual switchover is only applicable when both clocks are switching. Figure 6–35 shows the block diagram of the manual switchover circuit.

Figure 6–35. Manual Clock Switchover Circuitry in Stratix III PLLs



For more information about PLL software support in the Quartus II software, refer to the *ALTPLL Megafunction User Guide*.

Guidelines

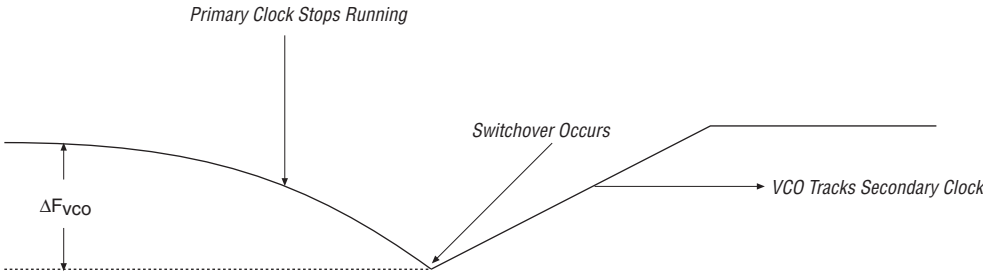
Use the following guidelines when implementing clock switchover in Stratix III PLLs.

- 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 and/or phase 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 input and output clocks.

  Both inclk0 and inclk1 must be running when the clkswitch signal goes high to instantiate 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 should use a low-bandwidth PLL. The low-bandwidth PLL reacts more slowly than a high-bandwidth PLL to reference input clock changes. When the 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 6–36 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 6–36. 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. Once the lock indication is stable, the system can re-enable the output clocks.

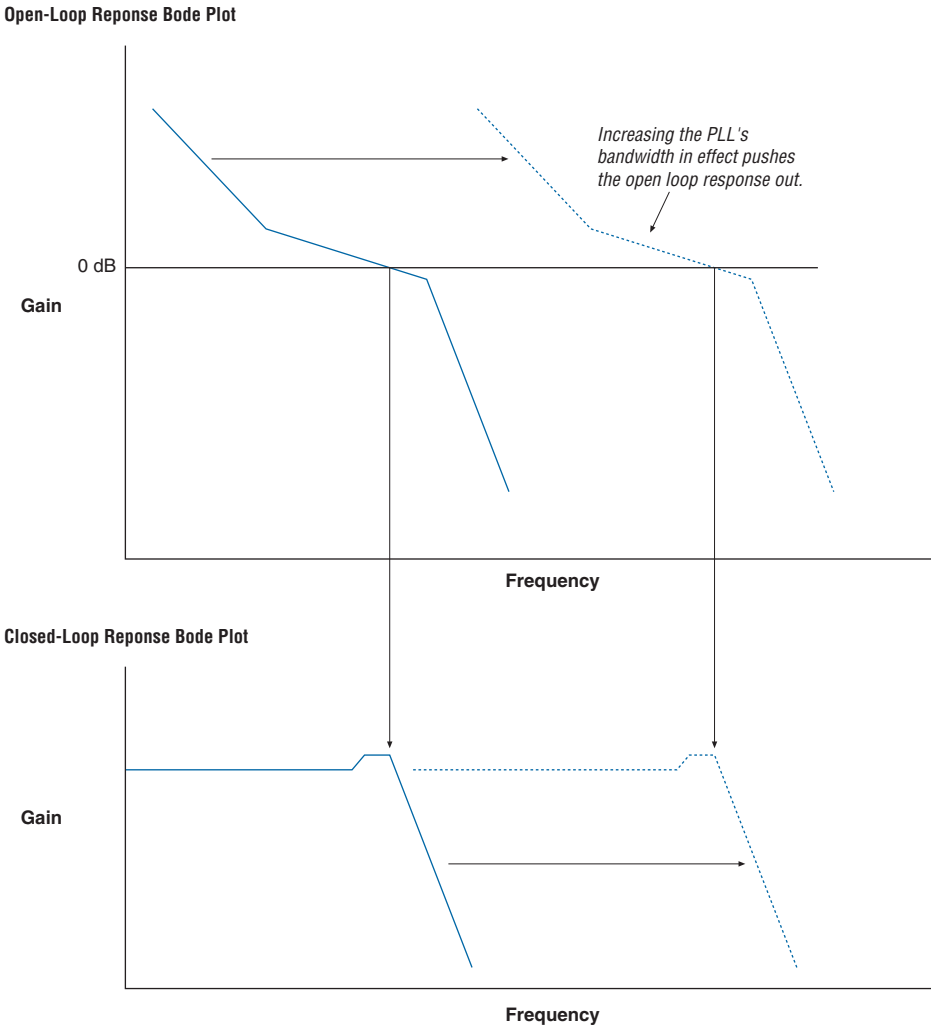
Programmable Bandwidth

Stratix III 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 the PLL bandwidth. The bandwidth is approximately the unity gain point for open loop PLL response. As Figure 6–37 shows, these points correspond to approximately the same frequency. Stratix III PLLs provide three bandwidth settings—low, medium (default), and high.

Figure 6–37. 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 III 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 III PLLs benefits applications requiring clock switchover.

A high-bandwidth PLL can benefit a system that needs to accept a spread-spectrum clock signal. Stratix III PLLs can track a spread-spectrum clock by using a high-bandwidth setting. Using a low-bandwidth 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 the clock switchover happens, 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 the input stopping) than a high-bandwidth PLL.

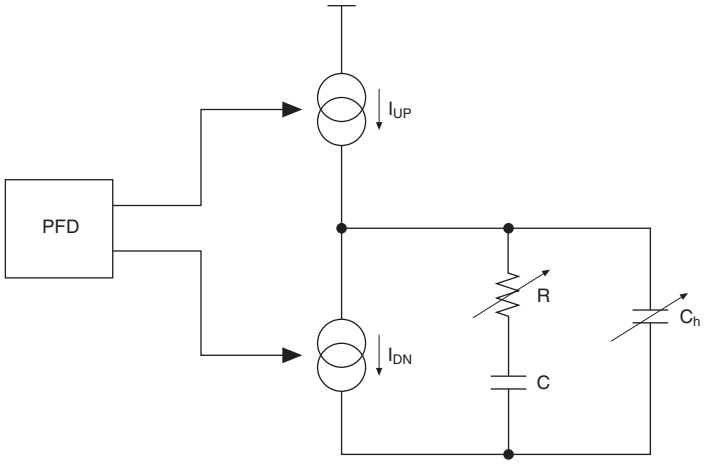
Implementation

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 III 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 (I_{cp}, R, C) values to achieve the desired bandwidth range.

Figure 6–38 shows the loop filter and the 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 6–38. Loop Filter Programmable Components



Phase-Shift Implementation

Phase shift is used to implement a robust solution for clock delays in Stratix III devices. Phase shift is implemented by using a combination of the VCO phase output and the counter starting time. The VCO phase output and counter starting time is the most accurate method of inserting delays, since it is based purely on counter settings, which are independent of process, voltage, and temperature.

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

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

Fine-resolution phase shifts are implemented 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. The minimum delay time that you can insert using this method is defined by Equation 6–1.

Equation 6–1.

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

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.

Coarse-resolution phase shifts are implemented by delaying the start of the counters for a predetermined number of counter clocks. You can express coarse phase shift as shown in Equation 6–2.

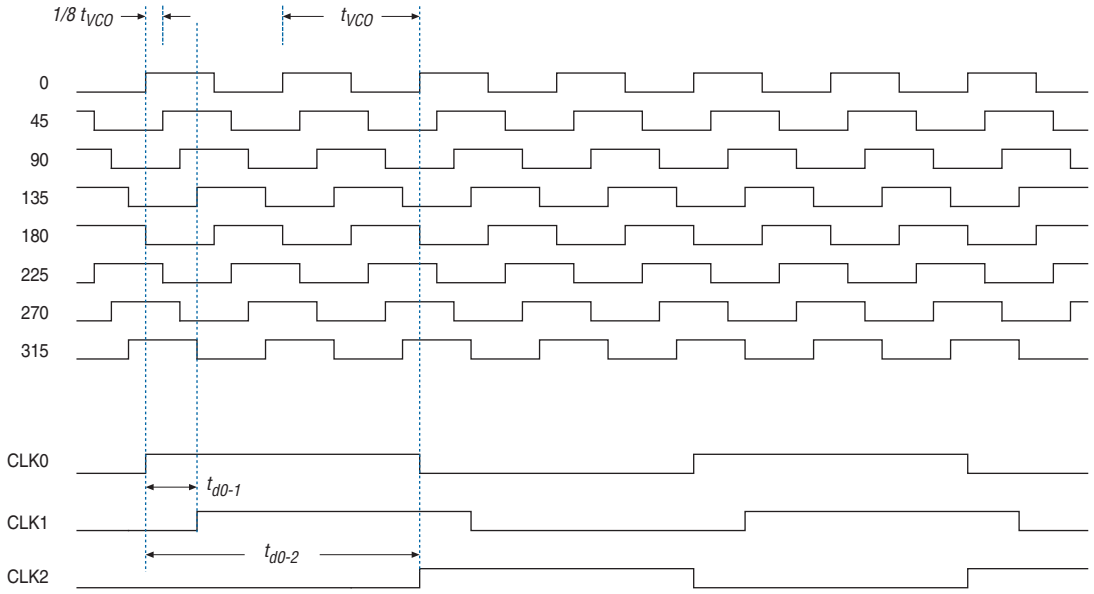
Equation 6–2.

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

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 6–39 shows an example of phase-shift insertion with the 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 off the `0phase` from the VCO and has the C value for the counter set to one. The `CLK0` signal is divided by four, two VCO clocks for high time and two VCO clocks for low time. `CLK1` is based off the 135° phase tap from the VCO and also has the C value for the counter set to one. The `CLK1` signal is also divided by 4. In this case, the two clocks are offset by $3\Phi_{fine}$. `CLK2` is based off the `0phase` from the VCO but has the C value for the counter set to three. This arrangement creates a delay of $2\Phi_{coarse}$ (two complete VCO periods).

Figure 6–39. Delay Insertion Using VCO Phase Output and Counter Delay Time



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

Stratix III devices support dynamic phase-shifting of VCO phase taps only. The phase shift is reconfigurable any number of times, and each phase shift takes about one SCANCLK cycle, allowing you to implement large phase shifts quickly.

PLL Reconfiguration

PLLs use several divide counters and different VCO phase taps to perform frequency synthesis and phase shifts. In Stratix III 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 the PLL bandwidth. You can use these PLL components to update the output-clock frequency and the PLL bandwidth and to phase-shift in real time, without reconfiguring the entire Stratix III 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 example, 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 ($\mathbf{t}_{CO}$) 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.

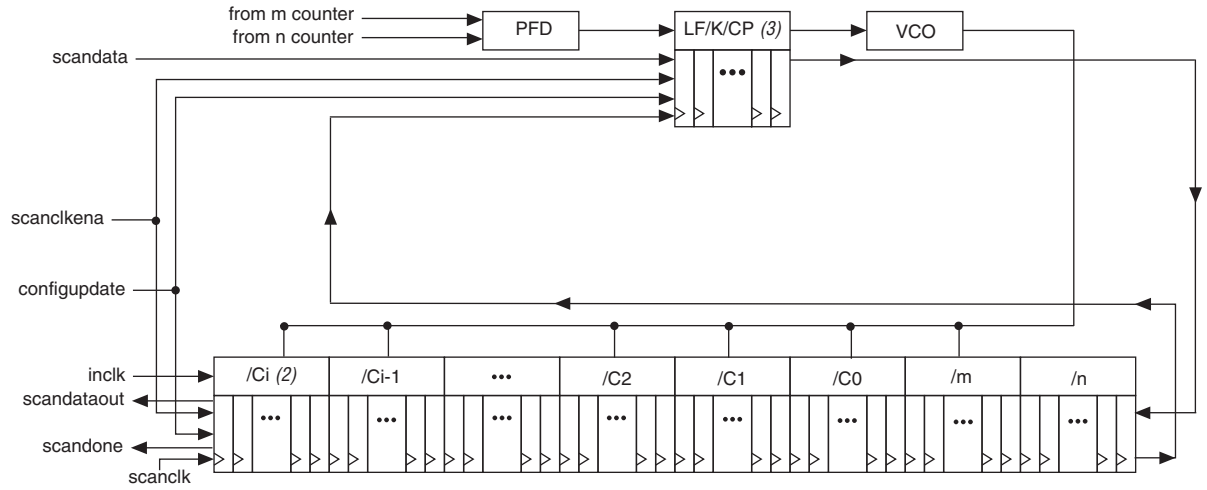
PLL Reconfiguration Hardware Implementation

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 (I_{cp}) and loop-filter components (R, C) to facilitate reconfiguration of the PLL bandwidth

Figure 6–40 shows how PLL counter settings can be dynamically adjusted by shifting their new settings into a serial shift-register chain or scan chain. Serial data is input to the scan chain via the scandataport and 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 6–40. PLL Reconfiguration Scan Chain



Notes to Figure 6–40:

(1) The Stratix III Left/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 6–15 lists how these signals can be driven by the programmable logic device (PLD) logic array or I/O pins.

Table 6–15. Real-Time PLL Reconfiguration Ports

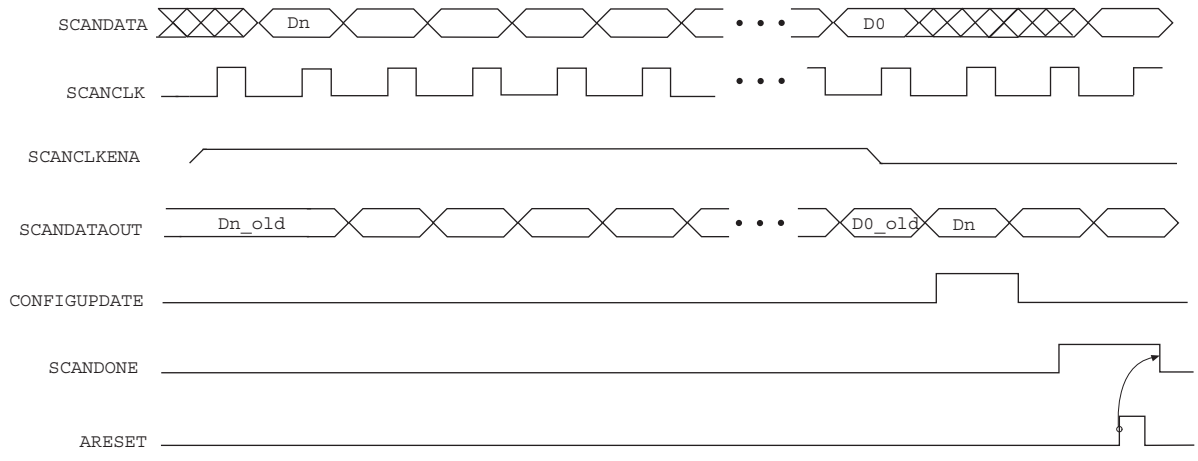
PLL Port Name	Description	Source	Destination
scandata	Serial input data stream to scan chain.	Logic array or I/O pin	PLL reconfiguration circuit
scanclk	Serial clock input signal. This clock can be free running.	GCLK/RCLK or I/O pins	PLL reconfiguration circuit
scanclkena	Enables scanclk and allows the 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 indicated 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

Use the following procedure to reconfigure the PLL counters:

1. The scanclkena signal is asserted at least one scanclk cycle prior to shifting in the first bit of scandata (Dn).
2. Serial data (scandata) is shifted into the scan chain on the second rising edge of scanclk.
3. After all 234 bits (Top/Bottom PLLs) or 180 bits (Left/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 C counters or the I_{cp}, R, or C settings.
7. Steps 1 through 5 can be repeated to reconfigure the PLL any number of times.

Figure 6–41 shows a functional simulation of the PLL reconfiguration feature.

Figure 6–41. 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 degrees) on the clock output, you must reconfigure the phase shift immediately after reconfiguring the counter clock frequency.

Post-Scale Counters (C0 to C9)

The multiply or divide values and duty cycle of post-scale counters can be reconfigured 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 this 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 could 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, would produce an output clock with 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 III PLLs. The Top/Bottom PLLs have 10 post-scale counters and a 234-bit scan chain, while the Left/Right PLLs have 7 post-scale counters and a 180-bit scan chain. Table 6–16 lists the number of bits for each component of a Stratix III PLL.

Table 6–16. Top/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
N	16	2	18
M	16	2	18
Charge Pump Current	0	3	3
VCO Post-Scale divider (K)	1	0	1
Loop Filter Capacitor *(4)*	0	2	2
Loop Filter Resistor	0	5	5
Unused CP/LF	0	7	7

Table 6–16. Top/Bottom PLL Reprogramming Bits (Part 2 of 2)

Block Name	Number of Bits		Total
	Counter	Other *(1)*	
Total number of bits	—	—	234

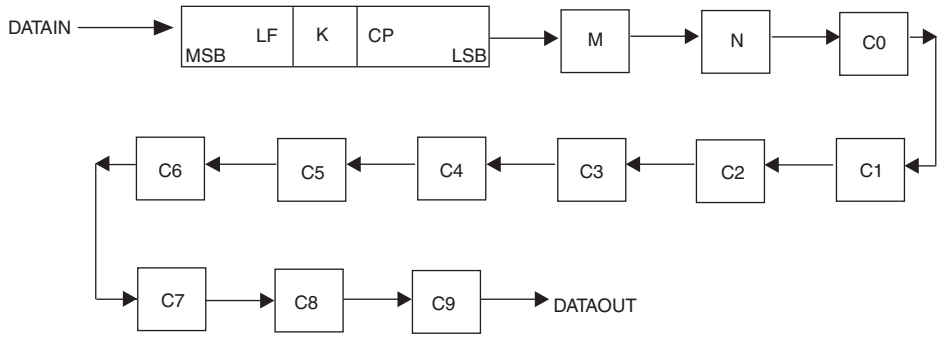
Notes to Table 6–16:

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

Table 6–16 lists the scan chain order of PLL components for Top/Bottom PLLs which have 10 post-scale counters. The order of bits is the same for the Left/Right PLLs, but the reconfiguration bits start with the `C6` post-scale counter.

Figure 6–42 shows the scan-chain order of PLL components for the Top/Bottom PLLs.

Figure 6–42. Scan-Chain Order of PLL Components for Top/Bottom PLLs *(Note 1)*

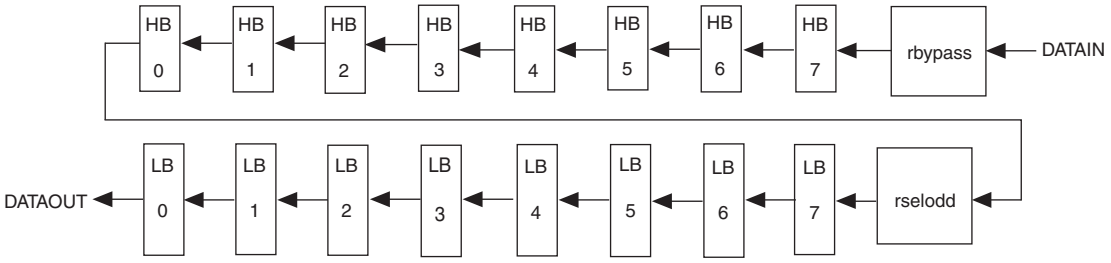


Note to Figure 6–42:

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

Figure 6–43 shows the scan-chain bit-order sequence for post-scale counters in all Stratix III PLLs.

Figure 6–43. Scan-Chain Bit-Order Sequence for Post-Scale Counters in Stratix III 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 6–17, Table 6–18, and Table 6–19 list the possible settings for charge pump current (I_{cp}), loop-filter resistor (R), and capacitor (C) values for Stratix III PLLs.

Table 6–17. 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 6–18. loop_filter_r 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 6–19. loop_filter_c Bit Settings

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

Bypassing PLL

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

Table 6–20 lists the settings for bypassing the counters in Stratix III PLLs.

Table 6–20. PLL Counter Settings

PLL Scan Chain Bits [0..10] Settings											
LSB *(2)*	**[1]**	**[2]**	**[3]**	**[4]**	**[5]**	**[6]**	**[7]**	**[8]**	**[9]**	**MSB** *(1)*	**Description**
0	X	X	X	X	X	X	X	X	X	1 *(3)*	PLL counter bypassed
X	X	X	X	X	X	X	X	X	X	0 *(3)*	PLL counter not bypassed because bit 10 (MSB) is set to 0

Notes to Table 6–20:

(1) Most significant bit (MSB).
(2) Least significant bit (LSB).
(3) 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 1.

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 the need to send serial data through the scan chain of the corresponding PLL. This feature simplifies the interface and allows you to quickly adjust 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 6–21 lists the control signals that are used for dynamic phase-shifting.

Table 6–21. 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 6–21. Dynamic Phase-Shifting Control Signals (Part 2 of 2)

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

Table 6–22 lists the PLL counter selection based on the corresponding PHASECOUNTERSELECT setting.

Table 6–22. Phase Counter Select Mapping

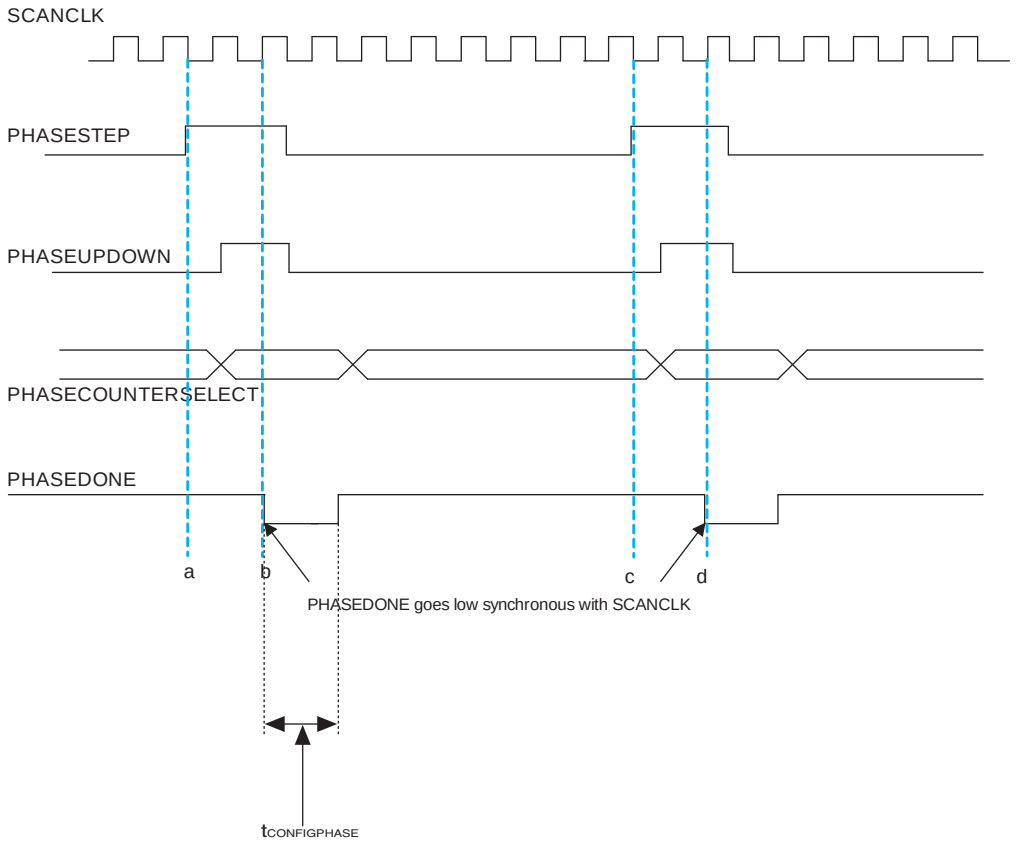
PHASECOUNTERSELECT[3]	[2]	[1]	[0]	Selects
0	0	0	0	All Output Counters
0	0	0	1	M Counter
0	0	1	0	C0 Counter
0	0	1	1	C1 Counter
0	1	0	0	C2 Counter
0	1	0	1	C3 Counter
0	1	1	0	C4 Counter
0	1	1	1	C5 Counter
1	0	0	0	C6 Counter
1	0	0	1	C7 Counter
1	0	1	0	C8 Counter
1	0	1	1	C9 Counter

The procedure to perform one dynamic phase-shift step is as follows:

1. Set phaseupdown and phasecounterselect as required.
2. Assert phasestep for at least two scanclk cycles. Each phasestep pulse enables one phase shift.
3. De-assert phasestep.
4. Wait for phasedone to go high.
5. Repeat steps 1 through 4 as many times as required to perform multiple phase-shifts.

All signals are synchronous to scanclk. They are latched on scanclk edges and must meet t_{su}/t_h requirements with respect to scanclk edges.

Figure 6–44. Dynamic Phase Shifting Waveform



Dynamic phase-shifting can be repeated indefinitely. All signals are synchronous to `scanclk` and must meet t_{su}/t_h requirements with respect to `scanclk` edges.

The `phasestep` signal is latched on the negative edge of `scanclk`. In Figure 6–44, this is shown by the second `scanclk` falling edge. `phasestep` must stay high for at least two `scanclk` cycles. On the second `scanclk` rising edge after `phasestep` is latched (the fourth `scanclk` rising edge in Figure 6–44), the values of `phaseupdown` and `phasecounterselect` are latched and the PLL starts dynamic phase-shifting for the specified counters and in the indicated direction. On the fourth `scanclk` rising edge, `phasedone` goes high to low and remains low until the PLL finishes dynamic phase-shifting. You can perform another dynamic phase-shift after the `phasedone` signal goes from low to high.

Depending on the VCO and `scanclk` frequencies, `phasedone` low time ($t_{CONFIGPHASE}$) 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 more information about the ALTPLL_RECONFIG MegaWizard Plug-In Manager, refer to the *ALTPLL_RECONFIG Megafunction User Guide*.

PLL Cascading and Clock Network Guidelines

When cascading PLLs in Stratix III devices, the source (upstream) PLL must have a low-bandwidth setting while the destination (downstream) PLL must have a high-bandwidth setting. There must be no overlap of the bandwidth ranges of the two PLLs.

To ensure that the memory interface's PLL is configured correctly in the external memory interface design, the following settings are required:

- The PLL used to generate the memory output clock signals and write data/clock signals must be set to **No compensation** mode to minimize output clock jitter.
- The reference input clock signal to the PLL must be driven by the dedicated clock input pin located adjacent to the PLL, or from the clock output signal from the adjacent PLL. To minimize output clock jitter, the reference input clock pin must not be routed through the core using global or regional clock networks. If reference clock is cascaded from another PLL, that upstream PLL must be configured in **No compensation** mode and **Low bandwidth** mode.

Spread-Spectrum Tracking

Stratix III 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 PLL. Stratix III PLLs can track a spread-spectrum input clock as long as it is within the input-jitter tolerance specifications and the modulation frequency of the input clock is below the PLL bandwidth which is specified in the fitter report. Stratix III devices cannot internally generate spread-spectrum clocks.

PLL Specifications

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

Chapter Revision History

Table 6–23 lists the revision history for this chapter.

Table 6–23. Chapter Revision History (Part 1 of 2)

Date	Version	Changes Made
July 2010	2.0	Updated Figure 6–44.
March 2010	1.9	Updated for the Quartus II software version 9.1 SP2 release: ■ Updated Table 6–10 and Table 6–11. ■ Updated Figure 6–42. ■ Updated the "Guidelines" and "PLL Cascading and Clock Network Guidelines" sections. ■ Removed "sub-regional clock networks" information. ■ Minor text edits.
July 2009	1.8	■ Updated "Clock Switchover" section. ■ Updated Figure 6–37.
May 2009	1.7	■ Added "PLL and Clock Network Guidelines for External Memory Interface" and "Zero-Delay Buffer Mode" sections. ■ Updated Figure 6–17.
February 2009	1.6	■ Updated Table 6–7 and Table 6–10. ■ Updated Figure 6–23. ■ Updated "PLL Clock I/O Pins", "Logic Array Blocks (LABs)", and "Clock Feedback Modes" sections. ■ Removed "Reference Documents" section.
October 2008	1.5	■ Updated Table 6–10, Table 6–13, and Table 6–14. ■ Updated "locked", "Manual Override", "Bypassing PLL", "PLL Clock I/O Pins", and "Dynamic Phase-Shifting" sections. ■ Updated Figure 6–22, Figure 6–24, and Figure 6–26. ■ Updated (Note 2) to Figure 6–22. ■ Added (Note 3) to Table 6–14. ■ Added Figure 6–27. ■ Updated New Document Format.
May 2008	1.4	■ Updated Table 6–3, Table 6–4, Table 6–5, Table 6–6, Table 6–7, and Table 6–14. ■ Added new Figure 6–5 through Figure 6–9 to "Periphery Clock Networks" section. ■ Updated "Logic Array Blocks (LABs)", "External Feedback Mode", "Phase-Shift Implementation", and "Spread-Spectrum Tracking" sections. ■ Updated notes to Figure 6–17. ■ Updated notes to Figure 6–22. ■ Updated notes to Figure 6–27. ■ Updated Figure 6–43.
November 2007	1.3	Updated "pfdena" on page 6–42.

Table 6–23. Chapter Revision History (Part 2 of 2)

Date	Version	Changes Made
October 2007	1.2	■ Updated Table 6–13 to remove a reference to gated locks. Updated Table 6–16 and added new rows to it. ■ Modified Figure 6–3 and Figure 6–40. ■ Edited notes for Figure 6–9, Figure 6–10, and Figure 6–17. ■ Replaced Figure 6–41. ■ Added section “Referenced Documents”. ■ Added live links for references.
May 2007	1.1	Changed frequency difference between inclk0 and inclk1 to more than 20% instead of 100% on page 42. Updated Table 6–16, note to Figure 6–17, and Figure 6–19.
November 2006	1.0	Initial Release.