Lattice Semiconductor Corporation

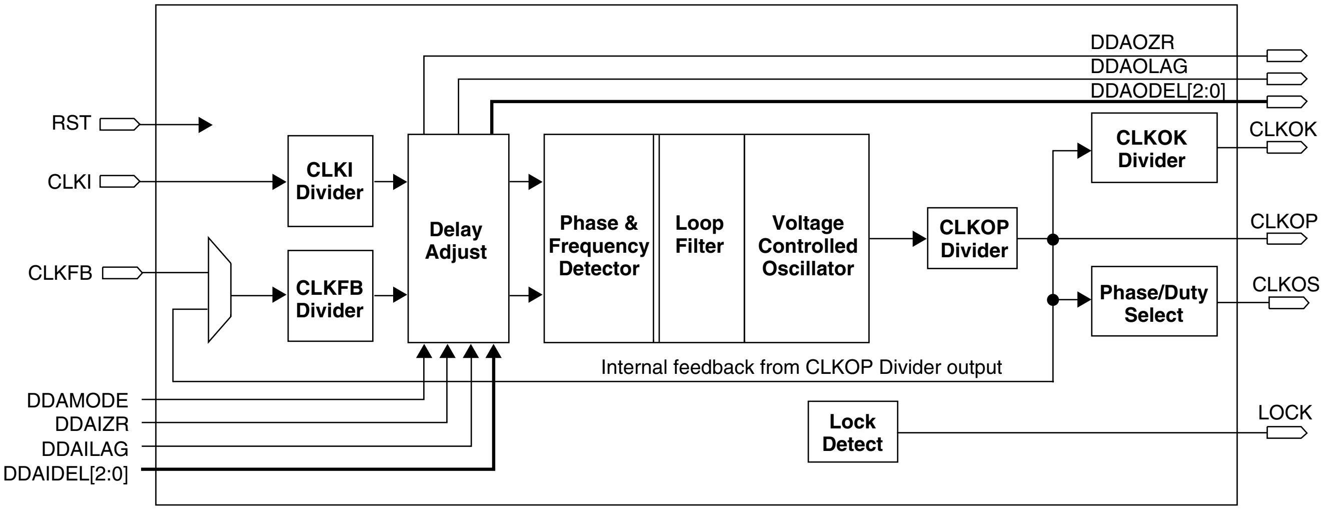
LatticeECP/EC and LatticeXP sysCLOCK PLL Design and Usage Guide

September 2006 Technical Note TN1049

Introduction

As clock distribution and clock skew management become critical factors in overall system performance, the Phase Locked Loop (PLL) is increasing in importance for digital designers. Lattice incorporates its sysCLOCK™ PLL technology in the LatticeECP™, LatticeEC™ and LatticeXP™ device families to help designers manage clocks within their designs. The PLL components in the LatticeECP/EC and LatticeXP device families share the same architecture. This technical note describes the features and functionalities of the PLLs and their configuration in the ispLEVER® design tool. Figure 11-1 shows the block diagram of the PLL.

Figure 11-1. LatticeECP/EC and LatticeXP sysCLOCK PLL Block Diagram



Features

- Clock synthesis
- Phase shift/duty cycle selection
- Internal and external feedback
- Dynamic delay adjustment
- No external components required
- Lock detect output

Functional Description

PLL Divider and Delay Blocks

Input Clock (CLKI) Divider

The CLKI divider is used to control the input clock frequency into the PLL block. It can be set to an integer value of 1 to 16. The divider setting directly corresponds to the divisor of the output clock. The input and output of the input divider must be within the input and output frequency ranges specified in the device data sheet.

Feedback Loop (CLKFB) Divider

The CLKFB divider is used to divide the feedback signal. Effectively, this multiplies the output clock, because the divided feedback must speed up to match the input frequency into the PLL block. The PLL block increases the output frequency until the divided feedback frequency equals the input frequency. Like the input divider, the feedback loop divider can be set to an integer value of 1 to 16. The input and output of the feedback divider must be within the input and output frequency ranges specified in the device data sheet.

Delay Adjustment

The delay adjust circuit provides programmable clock delay. The programmable clock delay allows for step delays in increments of 250ps (nominal) for a total of 2.00ns lagging or leading. The time delay setting has a tolerance. See device data sheet for details. Under this mode, CLKOP, CLKOS and CLKOK are identically affected. The delay adjustment has two modes of operation:

- **Static Delay Adjustment** – In this mode, the user-selected delay is configured at power-up.
- **Dynamic Delay Adjustment (DDA)** – In this mode, a simple bus is used to configure the delay. The bus signals are available to the general purpose FPGA.

Output Clock (CLKOP) Divider

The CLKOP divider serves the dual purposes of squaring the duty cycle of the VCO output and scaling up the VCO frequency into the 420MHz to 840MHz range to minimize jitter. Refer to Table 11-3 for CLKOP Divider value.

CLKOK Divider

The CLKOK divider feeds the global clock net. It divides the CLKOP signal of the PLL by the value of the divider. It can be set to values of 2, 4, 6,....126,128.

PLL Inputs and Outputs

CLKI Input

The CLKI signal is the reference clock for the PLL. It must conform to the specifications in the data sheet in order for the PLL to operate correctly. The CLKI can be derived from a dedicated dual-purpose pin or from routing.

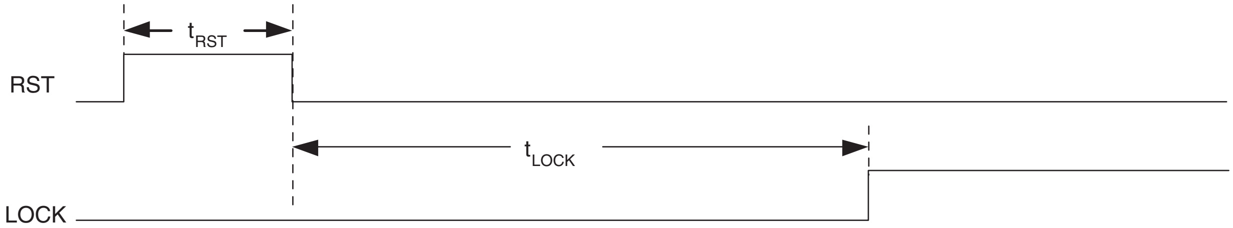
RST Input

The PLL reset occurs under two conditions. At power-up an internal power-up reset signal from the configuration block resets the PLL. The user controlled PLL reset signal RST is provided as part of the PLL module that can be driven by an internally generated reset function or a pin. This RST signal resets all internal PLL counters. When RST goes inactive, the PLL will start the lock-in process, and will take the t_{LOCK} time to complete the PLL lock.

Note: For LatticeECP/EC, RST must be asserted to re-start the locking process after losing lock. Refer to the LatticeECP/EC Family Data Sheet for the RST pulse width requirement. For LatticeXP, RST may be tied to GND.

Figure 11-2 shows the timing diagram of the RST Input.

Figure 11-2. RST Input Timing Diagram



CLKFBK Input

The feedback signal to the PLL, which is fed through the feedback divider can be derived from the global clock net, a dedicated dual-purpose pin, or directly from the CLKOP divider. Feedback must be supplied in order for the PLL to synchronize the input and output clocks. External feedback allows the designer to compensate for board-level clock alignment.

CLKOP Output

The sysCLOCK PLL main clock output, CLKOP, is a signal available for selection as a primary clock.

CLKOS Output with Phase and Duty Cycle Select

The sysCLOCK PLL auxiliary clock output, CLKOS, is a signal available for selection as a primary clock. The CLKOS is used when phase shift and/or duty cycle adjustment is desired. The programmable phase shift allows for different phase in increments of 45° to 315°. The duty select feature provides duty select in 1/8th of the clock period.

CLKOK Output with Lower Frequency

The CLKOK is used when a lower frequency is desired. It is a signal available for selection as a primary clock.

Dynamic Delay Control I/O Ports

Refer to Table 11-1 and Table 11-6 for detailed information.

LOCK Output

The LOCK output provides information about the status of the PLL. After the device is powered up and the input clock is valid, the PLL will achieve lock within the specified lock time. Once lock is achieved, the PLL lock signal will be asserted. If, during operation, the input clock or feedback signals to the PLL become invalid, the PLL will lose lock. PLL RST must be applied to re-synchronize the PLL to the reference clock. The LOCK signal is available to the FPGA routing to implement generation of RST.

PLL Attributes

The PLL utilizes several attributes that allow the configuration of the PLL through source constraints. The following section details these attributes and their usage.

FIN

The input frequency can be any value within the specified frequency range based on the divider settings.

CLKI_DIV, CLKFB_DIV, CLKOP_DIV, CLKOK_DIV

These dividers determine the output frequencies of each output clock. The user is not allowed to input an invalid combination; determined by the input frequency, the dividers, and the PLL specifications.

Frequency_Pin_CLKI, Frequency_Pin_CLKOP and Frequency_Pin_CLKOK

These output clock frequencies determine the divider values.

FDEL

The FDEL attribute is used to pass the Delay Adjustment step associated with the Output Clock of the PLL. This allows the user to advance or retard the Output Clock by the step value passed multiplied by 250ps(nominal). The step ranges from -8 to +8 resulting the total delay range to +/- 2ns.

PHASEADJ

The PHASEADJ attribute is used to select Coarse Phase Shift for CLKOS output. The phase adjustment is programmable in 45° increments.

DUTY

The DUTY attribute is used to select the Duty Cycle for CLKOS output. The Duty Cycle is programmable at 1/8 of the period increment.

FB_MODE

There are three sources of feedback signals that can drive the CLKFB Divider: internal, clocktree and external feedback. Clocktree feedback is used by default. Internal feedback takes the CLKOP output at CLKOP Divider output before the Clocktree to minimize the feedback path delay. The external feedback is driven from the pin.

DELAY_CNTL

This attribute is designed to select the Delay Adjustment mode. If the attribute is set to “DYNAMIC” the delay control switches between Dynamic and Static depending upon the input logic of DDAMODE pin. If the attribute is set to “STATIC”, Dynamic Delay inputs are ignored in this mode.

LatticeECP/EC and LatticeXP PLL Primitive Definitions

The PLL primitive name is EHXPLLB. Figure 11-3 shows the LatticeECP/EC and LatticeXP PLL primitive library symbol. Some features and I/Os are optional as described in Table 11-1 and Table 11-2.

Figure 11-3. LatticeECP/EC and LatticeXP PLL Primitive Symbol

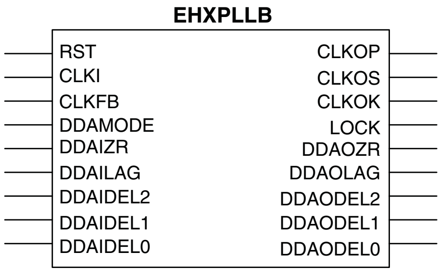


Table 11-1. LatticeECP/EC and LatticeXP PLL I/O Definitions

Signal	I/O	Description	Optional
CLKI	I	PLL reference clock input. From internal logic or dedicated clock pin.	No
CLKFB[1]	I	Feedback clock input. From internal node, CLKOP or dedicated pin.	No
RST	I	“1” to reset PLL	No
CLKOP	O	PLL output clock to clock tree	No
CLKOS	O	PLL output clock to clock tree with optional phase shift/duty cycle	Yes
CLKOK	O	PLL output clock to clock tree through K-divider for lower frequency	Yes
LOCK[2]	O	“1” indicates PLL locked to CLKI	Yes
DDAMODE	I	DDA Mode. “1”: Pin Control (dynamic), “0”: fuse control (static)	Yes
DDAIZR	I	DDA Delay Zero. “1” delay=0, “0”: delay=[DDILAG+DDAIDEL].	Yes
DDAILAG	I	DDA Lag/Lead. “1”: Lead, “0”: Lag.	Yes
DDAIDEL	I	DDA Delay	Yes
DDAOZR	O	DDA Delay Zero Output	Yes
DDAOLAG	O	DDA Lag/Lead Output	Yes
DDAODEL[2:0]	O	DDA Delay Output	Yes

1. When internal feedback or clocktree feedback is selected in the IPexpress™ GUI, software uses CLKOP as the source of CLKFB. CLKOS is not recommended as the source of CLKFB even in external feedback mode.
2. ModelSim® simulation models take two to four clock cycles from RST release to LOCK high.

PLL Attributes Definitions

The EHXPLLB can be configured through attributes in the source code. The following section details these attributes and their usage.

Table 11-2. LatticeECP/EC and LatticeXP PLL Attributes

User Accessible	IPexpress GUI Access	Attribute Name	Preference Language Support	Preference Editor Support	Value	Default Value	Units
CLKI Frequency	Y	FREQUENCY_PIN_CLKI	N	N	Note 5	100	MHz
CLKOP Frequency	Y	FREQUENCY_PIN_CLKOP	N	N	Note 5	100	MHz
CLKOK Frequency	Y	FREQUENCY_PIN_CLKOK	N	N	Note 5	50	MHz
CLKOP Frequency Tolerance	Y		N	N	0.0, 0.1, 0.2, 0.5, 1.0, 2.0, 5.0, 10.0	0.0	%
CLKOP Actual Frequency	Y		N	N			MHz
CLKOK Frequency Tolerance	Y		N	N	0.0, 0.1, 0.2, 0.5,1.0, 2.0, 5.0, 10.0	0.0	%

Table 11-2. LatticeECP/EC and LatticeXP PLL Attributes (Continued)

User Accessible	IPexpress GUI Access	Attribute Name	Preference Language Support	Preference Editor Support	Value	Default Value	Units
CLKOK Actual Frequency	Y		N	N			MHz
CLKI Divider Setting	Y	CLKI_DIV[4, 6]	Y	N	1 to 16 (1 to 15)	1	
CLKFB Divider Setting	Y	CLKFB_DIV[6]	Y	N	1 to 16 (1 to 15)	1	
CLKOP Divider Setting	Y	CLKOP_DIV[6]	Y	N	Note 3	8[2] (4 or 6)	
CLKOK Divider Setting	Y	CLKOK_DIV	Y	N	2, 4, 6,..,126, 128	2	
Fine Delay Adjust	N	FDEL	Y	Y	-8 to 8	0	ps
Coarse Phase Shift Selection (O)	Y	PHASEADJ	Y	N	0, 45, 90...315	0	Degrees
Duty Cycle Selection (1/8 increment)	Y	DUTY	Y	N	1 to 7	4	
Delay Control	Y	DELAY_CNTL1	Y	N	DYNAMIC/STATIC	STATIC	
Feedback Mode	Y	FB_MODE	N	N	INTERNAL/CLOCKTREE/EXTERNAL	CLOCKTREE	
CLKOS Select	Y		N	N			
CLKOK Select	Y		N	N			

1. DYNAMIC: This mode switches delay control between Dynamic and Static depending upon the input logic of the DDAMODE pin. STATIC: This is Static Control Only mode.
2. The CLKOP_DIV value is calculated to maximize the f_{VCO} within the specified range. For LatticeXP devices, if CLKOS is not used, the default value is 6. If CLKOS is used, the value is 4.
3. The CLKOP Divider values are 2, 4, 6, 8,..32 (2, 4, 6, 8..16 for LatticeXP devices) if CLKOS is not used. The CLKOP Divider values are 2, 4, 8, 16, 32 (2, 4, 8,16 for LatticeXP devices) if CLKOS is used.
4. All divider settings are user transparent in Frequency Mode. These are user attributes in Divider Mode.
5. Refer to data sheet for frequency limits.
6. Values in parentheses are for LatticeXP devices.
7. This attribute is not available in the IPexpress GUI. After reviewing the trace report file, users can determine the amount of delay that will best fit the clocking in their design. Further information on FDEL settings is described in the following section.

FDEL Settings

There are four ways the user can enter the desired FDEL value.

1. Although the FDEL entry is not available in the IPexpress GUI, the module generated by IPexpress includes the attribute with default value, “0”. Users can replace it with a desired value.

   Example of source code with default FDEL value:

   ```
   attribute FDEL of ehxpll_mod_0_0 : label is "0";
   generic map (…
      FDEL=>"0",
      …
      ")
   ```

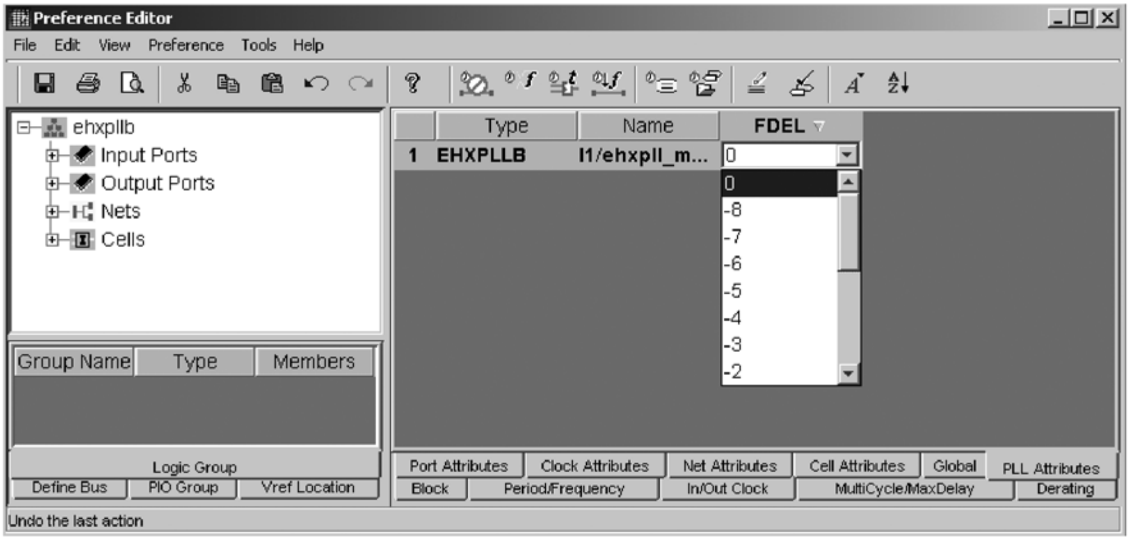
2. Preference File: User may specify the preference in the Preference file.

   Example:

   ```
   ASIC "FDEL_CODE_0_0" TYPE "EHXPLLB" FDEL="-2"  ;
   ```

3. Pre-Map Preference Editor: Users can enter the FDEL value in the Pre-Map Preference Editor as shown in Figure 11-4.

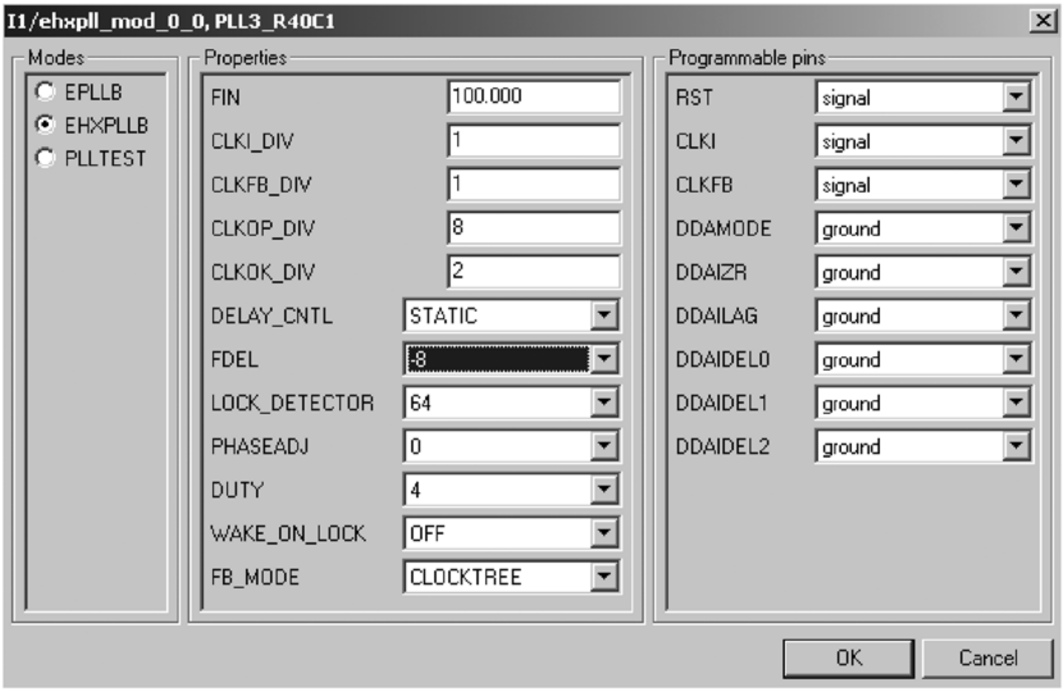
Figure 11-4. Pre-Map Preference Editor



	Type	Name	FDEL
1	EHXPLLB	I1/ehxpll_m...	0

4. EPIC Device Editor: Users can edit their preferences in the EPIC Device Editor as shown in Figure 11-5.

Figure 11-5. EPIC Preferences Edit Window


I1/ehxpll_mod_0_0, PLL3_R40C1

Modes

○ EPLLB
● EHXPLLB
○ PLLTEST

Properties

FIN	100.000
CLKI_DIV	1
CLKFB_DIV	1
CLKOP_DIV	8
CLKOK_DIV	2
DELAY_CNTL	STATIC
FDEL	-8
LOCK_DETECTOR	64
PHASEADJ	0
DUTY	4
WAKE_ON_LOCK	OFF
FB_MODE	CLOCKTREE

Programmable pins

RST	signal
CLKI	signal
CLKFB	signal
DDAMODE	ground
DDAIZR	ground
DDAILAG	ground
DDAIDEL0	ground
DDAIDEL1	ground
DDAIDEL2	ground

Dynamic Delay Adjustment

The Dynamic Delay Adjustment is controlled by the DDAMODE input. When the DDAMODE input is set to “1”, the delay control is handled through the inputs, DDAIZR, DDAILAG and DDAIDEL(2:0). For this mode, the attribute “DELAY_CNTL” must be set to “DYNAMIC”. Table 11-3 shows the delay adjustment values based on the attribute/input settings.

In this mode, the PLL may come out of lock due to the abrupt change of phase. RST must be asserted to re-lock the PLL. Upon de-assertion of RST, the PLL will start the lock-in process and will take the t_{LOCK} time to complete the PLL lock.

Table 11-3. Delay Adjustment

DDAMODE = 1: Dynamic Delay Adjustment			DELAY 1 t_{DLY} = 250ps (nominal)	DDAMODE = 0
DDAIZR	DDAILAG	DDAIDEL[2:0]		Equivalent FDEL Value
0	1	111	Lead 8 t_{DLY}	-8
0	1	110	Lead 7 t_{DLY}	-7
0	1	101	Lead 6 t_{DLY}	-6
0	1	100	Lead 5 t_{DLY}	-5
0	1	011	Lead 4 t_{DLY}	-4
0	1	010	Lead 3 t_{DLY}	-3
0	1	001	Lead 2 t_{DLY}	-2
0	1	000	Lead 1 t_{DLY}	-1
1	Don’t Care	Don’t Care	No delay	0
0	0	000	Lag 1 t_{DLY}	1
0	0	001	Lag 2 t_{DLY}	2
0	0	010	Lag 3 t_{DLY}	3
0	0	011	Lag 4 t_{DLY}	4
0	0	100	Lag 5 t_{DLY}	5
0	0	101	Lag 6 t_{DLY}	6
0	0	110	Lag 7 t_{DLY}	7
0	0	111	Lag 8 t_{DLY}	8

Note: t_{DLY} = Unit Delay Time = 250 ps (nominal). See the data sheet for the tolerance of this delay

PLL Usage in IPexpress

Including sysCLOCK PLLs in a Design

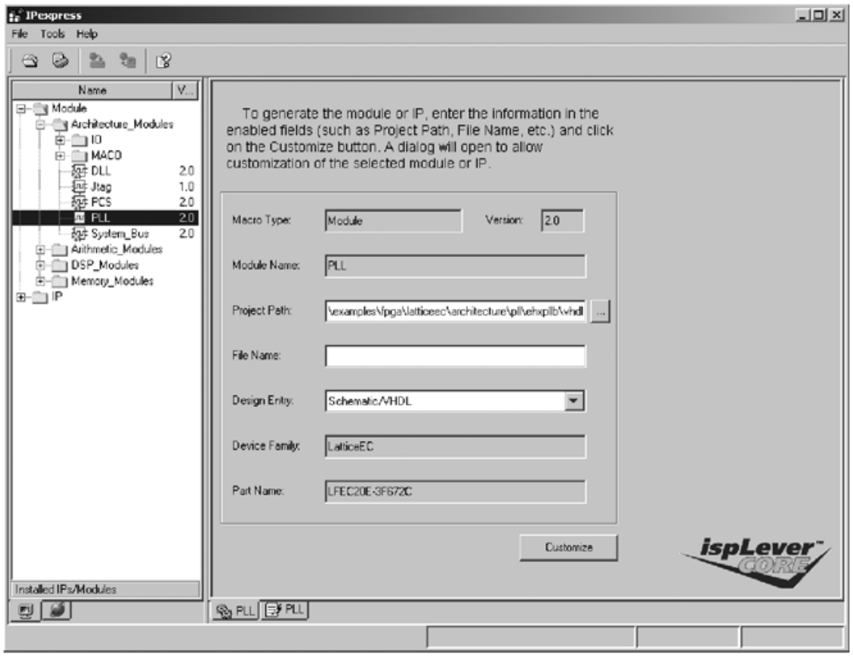
The sysCLOCK PLL capability can be accessed through the IPexpress GUI. The following section describes the usage of IPexpress.

IPexpress Usage

The LatticeECP/EC and LatticeXP PLL is fully supported in IPexpress in the ispLEVER software. IPexpress allows the user to define the desired PLL using a simple, easy-to-use GUI. Following definition, a VHDL or Verilog module that instantiates the desired PLL is created. This module can be included directly in the user’s design.

Figure 11-6 shows the main window when PLL is selected. The only entry required in this window is the module name. After entering the module name, clicking on “Customize” will open the “Configuration” window as shown in Figure 11-7.

Figure 11-6. IPexpress Main Window



To generate the module or IP, enter the information in the enabled fields (such as Project Path, File Name, etc.) and click on the Customize button. A dialog will open to allow customization of the selected module or IP.

Macro Type: Module Version: 2.0

Module Name: PLL

Project Path: \examples\fpga\latticeec\architecture\pll\ehxpllb\vhdl

File Name:

Design Entry: Schematic/VHDL

Device Family: LatticeEC

Part Name: LFEC20E-3F672C

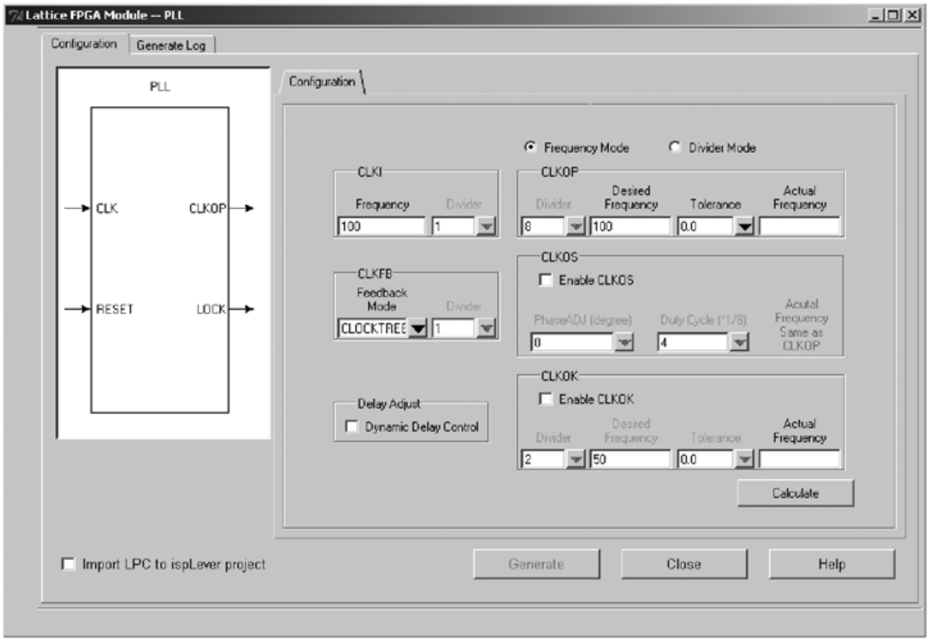
Configuration Tab

The Configuration Tab lists all user accessible attributes. Default values are set initially.

There are two modes in the Configuration Tab which can be used to configure the PLL, Frequency Mode and Divider Mode.

Frequency Mode: In this mode, the user enters input and output clock frequencies and the software calculates the divider settings for the user. If the output frequency the user entered is not achievable, the nearest frequency will be displayed in the 'Actual' text box. After input and output frequencies are entered, clicking the 'Calculate' button will display the divider values. If the desired output frequency is not achievable with the given frequency tolerance, the software generates an error. Users may increase the frequency tolerance or change the output frequencies. Figure 11-7 shows the Configuration Tab window.

Figure 11-7. Configuration Tab



● Frequency Mode ○ Divider Mode

CLKI: Frequency 100; Divider 1

CLKOP: Divider 8; Desired Frequency 100; Tolerance 0.0; Actual Frequency

CLKFB: Feedback Mode CLOCKTREE; Divider 1

CLKOS: □ Enable CLKOS; PhaseADJ (degree) 0; Duty Cycle (*1/8) 4; Acutal Frequency Same as CLKOP

CLKOK: □ Enable CLKOK; Divider 2; Desired Frequency 50; Tolerance 0.0; Actual Frequency

Delay Adjust: □ Dynamic Delay Control

□ Import LPC to ispLever project

Divider Mode: In this mode, the user sets the input frequency and divider settings. It is assumed the user is familiar with the PLL operation. The user must choose the CLKOP Divider value to maximize the f_{VCO} to achieve optimum PLL performance. After input frequency and divider settings are set, clicking the ‘Calculate’ button will display the output frequencies. If the divider settings are out of the PLL specification, the software will generate an error.

EHXPLLB Example Projects

ispLEVER provides example PLL projects for first time PLL users.

In the ispLEVER Project Navigator, go to the **File** menu and select **Open Examples...**.

Select the **FPGA** folder. The LatticeEC and LatticeXP folders include PLL example projects in both Verilog and VHDL.

Equations for Generating Input and Output Frequency Ranges

The values of $f_{IN,}$ f_{OUT} and f_{VCO} are the absolute frequency ranges for the PLL. The values of $f_{INMIN,}$ f_{INMAX}, f_{OUTMIN} and f_{OUTMAX} are the calculated frequency ranges based on the divider settings. These calculated frequency ranges become the limits for the specific divider settings used in the design.

Table 11-4. Frequency Limits

Parameter	LatticeECP/EC	LatticeXP
f_{IN}	Note 1	
f_{OUT}	Note 1	
f_{OUTK}	Note 1	
f_{VCO} (Hz)	Note 1	
CLKI Divider	1 to 16	1 to 15
CLKFB Divider	1 to 16	1 to 15
CLKOP Divider	See Table 11-2	
CLKOK Divider	2, 4, 6, 8,.. ,126, 128	
Maximum (N*V)	32	30
f_{PFD} (f_{IN}/M) (Hz)	Note 1	

Note: Refer to data sheet for the latest data.

The divider names are abbreviated with legacy names as:

- CLKI DIVIDER:M
- CLKFB DIVIDER:N
- CLKOP DIVIDER:V
- CLKOK DIVIDER:K

for use in the equations below.

f_{VCO} Constraint

From the loop:

$f_{OUT} = f_{IN} * (N/M)$ (1)

From the loop:

$f_{VCO} = f_{OUT} * V$ (2)

Substitute (1) in (2) yields:

$f_{VCO} = f_{IN} * (N/M) * V$ (3)

Arrange (3):

$f_{IN} = (f_{VCO} / (V*N))*M$ (4)

From equation (4):

$f_{INMIN} = ((f_{VCOMIN} /(V*N))*M$ (5)
$f_{INMAX} = (f_{VCOMAX}/(V*N))*M$ (6)

f_{PFD} Constraint

From the loop:

$f_{PFD} = f_{IN} / M$ (7)
$f_{IN} = f_{PFD} * M$
$f_{INMIN} = f_{PFDMIN} * M = 25 * M$ (assume $f_{PFDMIN} = 25$) (8)

Equation (5) becomes:

$$\mathbf{f_{INMIN} = ((f_{VCOMIN} / (V*N))*M, \text{ if below } 25 * M \text{ round up to } 25 * M} \quad (9)$$

From the loop:

$$f_{INMAX} = f_{PFDMAX} * M = 420 * M \quad (10)$$

Assume $f_{INMAX} = 420$

Equation (6) becomes:

$$\mathbf{f_{INMAX} = (f_{VCOMAX} / (V*N))*M, \text{ if above } 420 \text{ round down to } 420} \quad (11)$$

From equation (1):

$$\mathbf{f_{OUTMIN} = f_{INMIN} * (N/M), \text{ if below } 25 * N \text{ round up to } 25 * N} \quad (12)$$
$$\mathbf{f_{OUTMAX} = f_{INMAX} * (N/M), \text{ if above } 420 \text{ round down to } 420} \quad (13)$$

$$f_{OUTKMIN} = f_{OUTMIN} / K$$
$$f_{OUTKMAX} = f_{OUTMAX} / K$$

Clock Distribution in LatticeECP/EC and LatticeXP

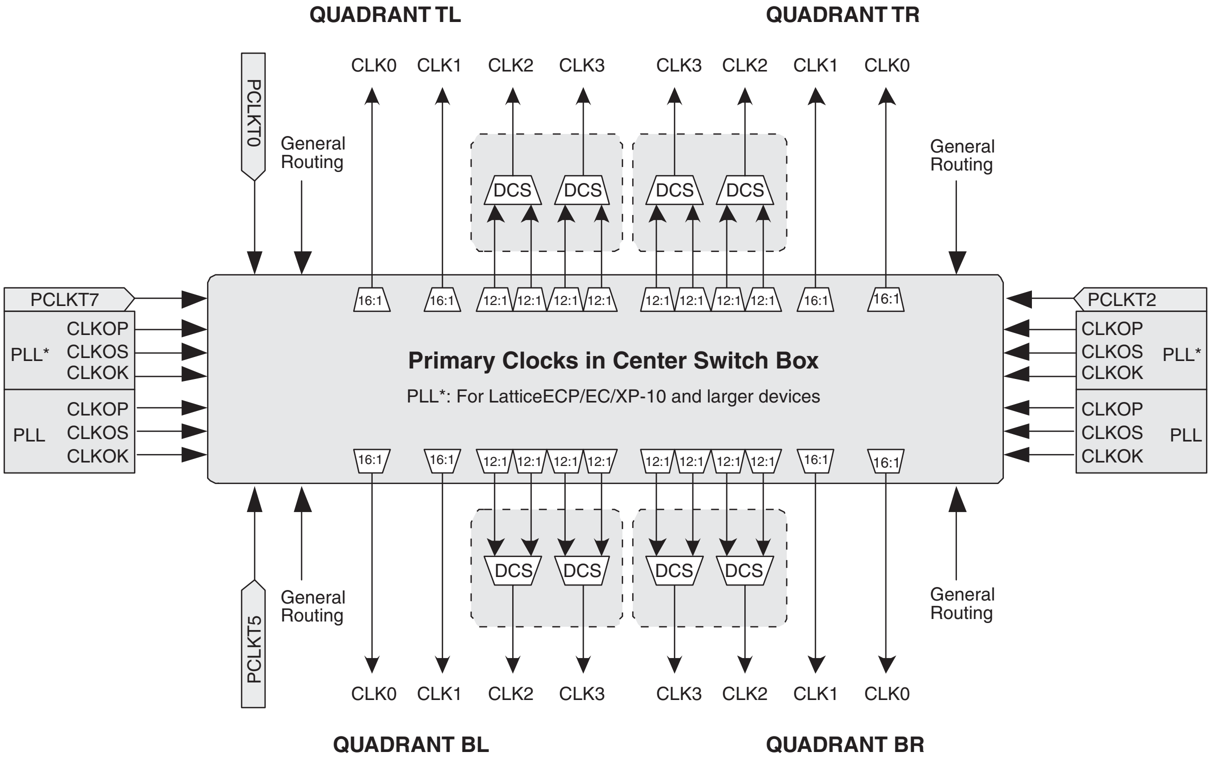
The clock inputs are selected from external I/Os, the sysCLOCK PLLs or general routing. These clock inputs are fed through the chip via a clock distribution system.

LatticeECP/EC and LatticeXP devices provide a quadrant-based primary and secondary clock structure.

Primary Clock Sources and Distribution

Each quadrant has four primary clock nets: CLK0, CLK1, CLK2 and CLK3. CLK2 and CLK3 provide dynamic clock selection (DCS) capability. Figure 11-8 illustrates the block diagram of the primary clock distribution.

Figure 11-8. Primary Clocks and Center Switch Boxes



Note: Two PLLs are available in LatticeECP/EC/XP-6 or smaller devices.

Note on the Primary Clock
The CLKOP must be used as the feedback source to optimize the PLL performance.

Most designers use the PLL for the clock tree injection removal mode and the CLKOP should be assigned as the primary clock. This is done automatically by the software unless the user specifies otherwise.

CLKOP can route to CLK0 and CLK1 only. CLKOS/CLKOK can route to all primary clocks (CLK0 to CLK3).

When CLK2 or CLK3 is used as a primary clock and there is only one clock input to the DCS, the DCS is assigned as a buffer mode by the software. Please see the DCS section of this document for further information.

Clock Net Preferences

There are two clock nets, primary clock and secondary clock.

As illustrated in Figure 11-9, users can set each clock to the desired clock net in the Pre-map Preference Editor or write in the Preference File as shown in the examples below.

Primary-Pure and Primary-DCS

Primary Clock Net can be assigned to either Primary-Pure (CLK0 and CLK1) or Primary-DCS (CLK2 and CLK3).

Syntax Example

```
USE PRIMARY DCS NET "bf_clk";
```

Global Primary Clock and Quadrant Primary Clock

Global Primary Clock
If a primary clock is not assigned as a quadrant clock, the software assumes it is a Global Clock.

There are two Global Primary/Pure Clocks and two Global Primary/DCS Clocks available.

Quadrant Primary Clock
Any Primary Clock may be assigned to a Quadrant Clock. The Clock may be assigned to a single quadrant or to two adjacent quadrants (not diagonally adjacent).

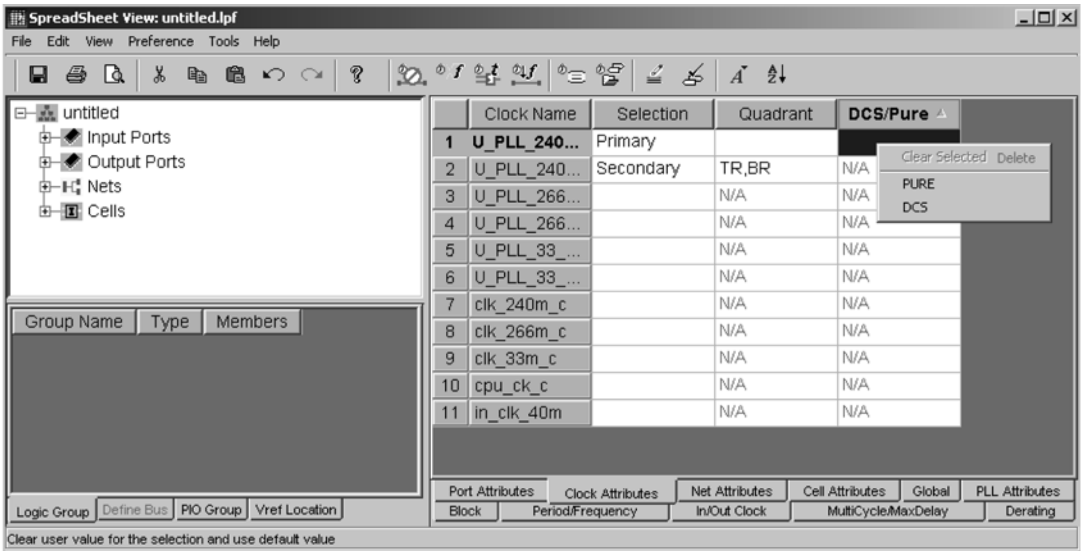
When the quadrant clock net is used, users must ensure that the registers each clock drives can be assigned in that quadrant without any routing issues.

With the Quadrant Primary Clocking scheme, the maximum number of Primary Clocks is 16 as long as all the Primary Clock Sources are avaialble.

Syntax Example

```
USE PRIMARY PURE NET "bf_clk" QUADRANT_TL;
```

Figure 11-9. Clock Preferences in the Pre-map Preference Editor



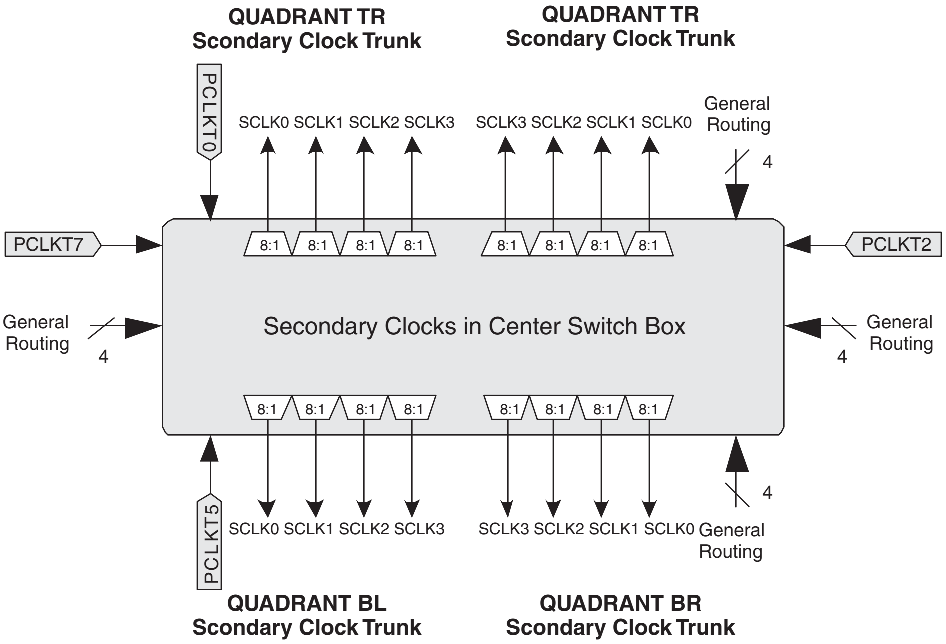
	Clock Name	Selection	Quadrant	DCS/Pure
1	U_PLL_240...	Primary		
2	U_PLL_240...	Secondary	TR,BR	N/A
3	U_PLL_266...		N/A	N/A
4	U_PLL_266...		N/A	N/A
5	U_PLL_33_...		N/A	N/A
6	U_PLL_33_...		N/A	N/A
7	clk_240m_c		N/A	N/A
8	clk_266m_c		N/A	N/A
9	clk_33m_c		N/A	N/A
10	cpu_ck_c		N/A	N/A
11	in_clk_40m		N/A	N/A

Clear user value for the selection and use default value

Secondary Clock Sources and Distribution

LatticeECP/EC and LatticeXP devices support quadrant base Secondary Clocks. Figure 11-10 describes the Secondary Clock arrangement.

Figure 11-10. Secondary Clock Center Switch Box



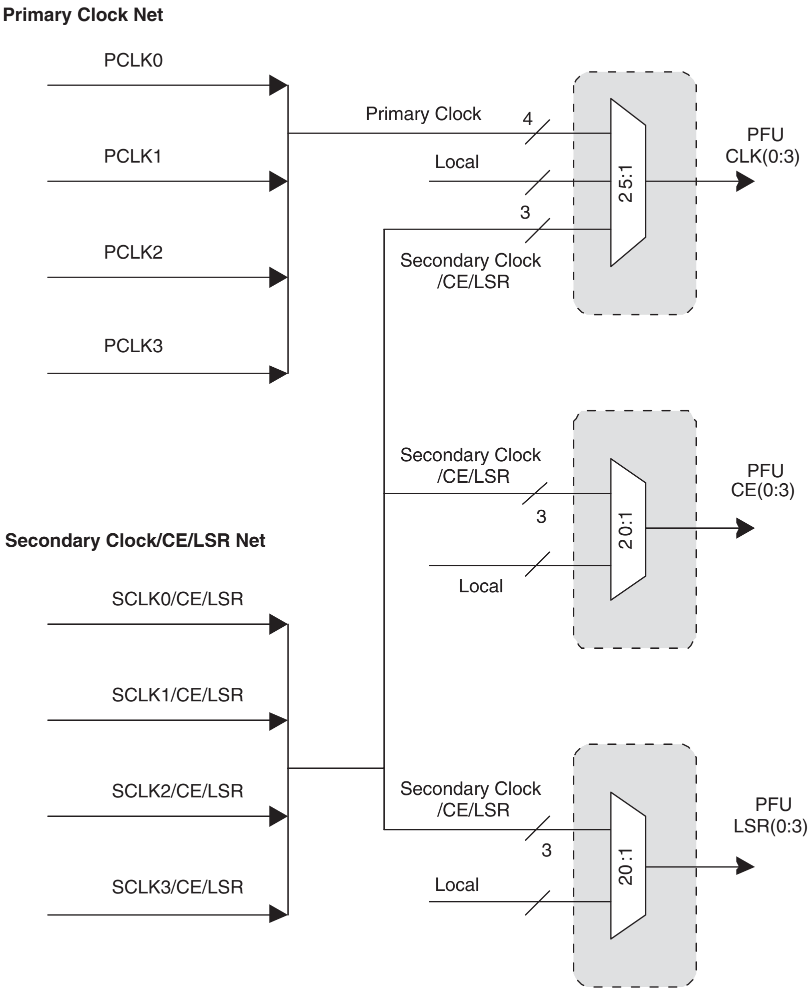
Limitations on Secondary Clock Availability

As illustrated in Figure 11-11, three secondary clocks are shared with CLK, CE and LSR.

This routing scheme limits the secondary clocks available per quadrant base to three, which results in a maximum of 12 available secondary clocks per device. Figure 11-11 illustrates the primary and secondary clock distribution structure of the PFUs.

LFEC6/LFXP6 and smaller devices have limited routing resources and can implement a maximum of nine secondary clocks per device.

Figure 11-11. Primary Clock and Secondary Clock/CE/LSR Distribution



Dynamic Clock Selection (DCS)

DCS is a global clock buffer incorporating a smart multiplexer function that takes two independent input clock sources and avoids glitches or runt pulses on the output clock, regardless of when the enable signal is toggled. The DCS blocks are located in pairs at the center of each side of the device. Thus, there are eight of them in every device.

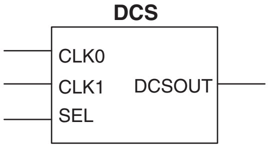
Table 11-5. DCS I/O

I/O	Name	Description
Input	SEL	Input Clock Select
	CLK0	Primary Clock Input 0
	CLK1	Primary Clock Input 1
Output	DCSOUT	To Primary Clock

Table 11-6. DCS Attributes

Attribute Name	Description	Output		Value
		SEL=0	SEL=1	
DCS MODE	Rising edge triggered, latched state is high	CLK0	CLK1	POS (Default)
	Falling edge triggered, latched state is low	CLK0	CLK1	NEG
	Sel is active high, Disabled output is low	0	CLK1	HIGH_LOW
	Sel is active high, Disabled output is high	1	CLK1	HIGH_HIGH
	Sel is active low, Disabled output is low	CLK0	0	LOW_LOW
	Sel is active low, Disabled output is high	CLK0	1	LOW_HIGH
	Buffer for CLK0	CLK0	CLK0	CLK0
	Buffer for CLK1	CLK1	CLK1	CLK1

Figure 11-12. DCS Primitive Symbol



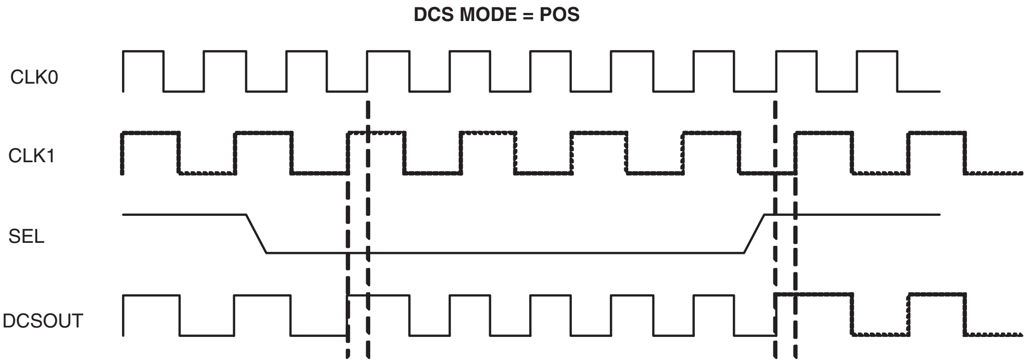
DCS Waveforms

The DCSOUT waveform timing is described in Figure 11-13 for each mode. The ‘POS’ and ‘NEG’ modes describe DCSOUT timing at both the falling and rising edges of SEL.

Figure 11-13. DCS Waveforms

DCS MODE = POS

At the rising edge (POS) of SEL, the DCSOUT changes from CLK0 to CLK1. This mode is the default mode.

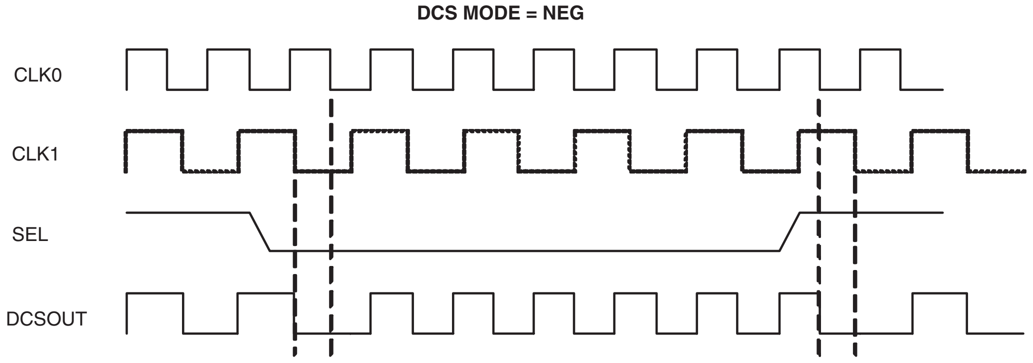


SEL Falling edge:
- Wait for CLK1 rising edge, latch output & remain high
- Switch output at CLK0 rising edge

SEL Rising edge:
- Wait for CLK0 rising edge, latch output & remain high
- Switch output at CLK1 rising edge

DCS MODE = NEG

At the falling edge (NEG) of SEL, the DCSOUT changes from CLK0 to CLK1.



SEL Falling edge:
- Wait for CLK1 falling edge, latch output & remain low
- Switch output at CLK0 falling edge

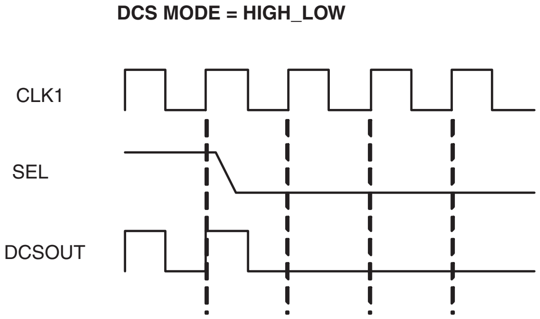
SEL Rising edge:
- Wait for CLK0 falling edge, latch output & remain low
- Switch output at CLK1 falling edge

DCS MODE = HIGH_LOW

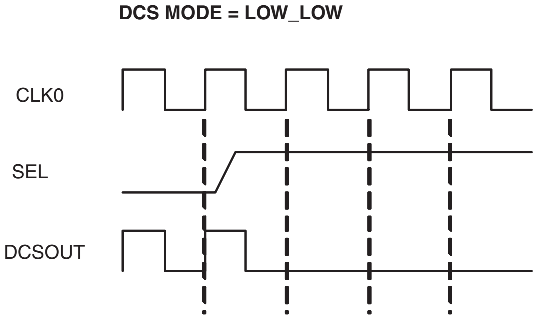
SEL is active high (HIGH) to select CLK1, and the disabled output is LOW.

DCS MODE = LOW_LOW

SEL is active low (LOW) to select CLK0, and the disabled output is LOW.



- Switch low at CLK1 falling edge.
- If SEL is low, output stays low at on CLK1 rising edge. SEL must not change during setup prior to rising clock.



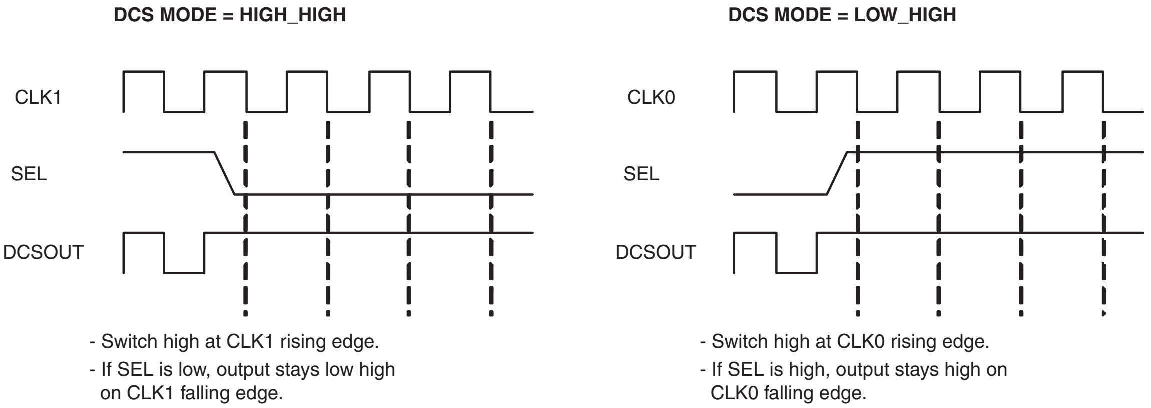
- Switch low at CLK0 falling edge.
- If SEL is high, output stays low at on CLK0 rising edge.

DCS MODE = HIGH_HIGH

SEL is active high (HIGH) to select CLK1, and the disabled output is HIGH.

DCS MODE = LOW_HIGH

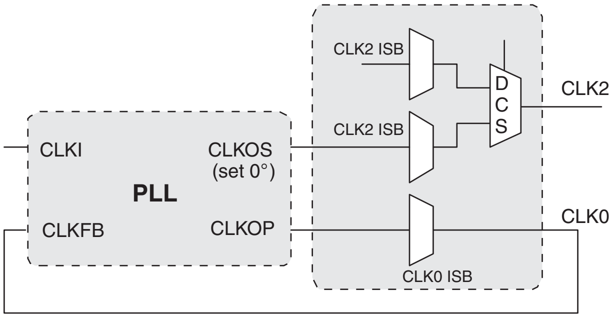
SEL is active low (LOW) to select CLK0, and the disabled output is HIGH.


DCS MODE = HIGH_HIGH
CLK1
SEL
DCSOUT
- Switch high at CLK1 rising edge.
- If SEL is low, output stays low high on CLK1 falling edge.
DCS MODE = LOW_HIGH
CLK0
SEL
DCSOUT
- Switch high at CLK0 rising edge.
- If SEL is high, output stays high on CLK0 falling edge.

Use of DCS with PLL

The four PLL CLKOP sources reach CLK0 and CLK1 of the quadrant clock. When using the DCS, the PLL needs a free-running feedback path to keep the PLL in lock. The user should use CLKOP as this feedback path, and CLKOS as the input into the DCS. CLKOP does not reach CLK2 or CLK3 to prevent the user from using the PLL improperly with DCS. See Figure 11-14.

Figure 11-14. Implementation of Dynamic Clock Select for a PLL Clock (Must Use Both CLKOP and CLKOS)



Other Design Considerations

Jitter Considerations

The Clock Output jitter specifications assume that the reference clock is free of jitter. Even if the clock source is clean, there are a number of sources that place noise in the PLL clock input. While intrinsic jitter is not avoidable, there are ways to minimize the input jitter and output jitter.

Signal inputs that share the same I/O bank with PLL clock inputs are preferably less noisy inputs and slower switching signals. Try to avoid placing any high speed and noisy signals in the same I/O bank with clock signals if possible. Use differential signaling if possible.

When external feedback is used, the PCB path must be well designed to avoid reflection as well as noise coupling from adjacent signal sources. A shorter PCB feedback path length does not necessarily reduce feedback input jitter.

Simulation Limitations

- Simulation does not compensate for external delays and dividers in the feedback loop.
- The LOCK signal is not simulated according to the t_{LOCK} specification. The LOCK signal will appear active shortly after the simulation begins, but will remain active throughout the simulation.
- The jitter specifications are not included.

PCB Layout Recommendations for VCCPLL and GNDPLL if Separate Pins are Available

It is best to connect VCCPLL to VCC at a single point using a filter and to create a separate GNDPLL plane directly under it (tied via a single point to GND).

Separate islands for both VCCPLL and GNDPLL are recommended if applicable.

DCS Usage with Verilog

```
module dcs(clk0,clk1,sel,dcsout);

input clk0, clk1, sel;
output dcsout;

DCS DCSInst0 (.SEL(sel),.CLK0(clk0),.CLK1(clk1),.DCSOUT(dcsout));
defparam DCSInst0.DCSMODE = "CLK0";

endmodule
```

DCS Usage with VHDL

```
COMPONENT DCS
-- synthesis translate_off
        GENERIC       (
                      DCSMODE : string :=  "POS"
                      );
-- synthesis translate_on

        PORT          (
                      CLK0           :IN    std_logic;
                      CLK1           :IN    std_logic;
                      SEL            :IN    std_logic;
                      DCSOUT         :OUT   std_logic
                      );
END COMPONENT;

    attribute DCSMODE : string;
    attribute DCSMODE of DCSinst0 : label is "POS";

begin

DCSInst0: DCS
-- synthesis translate_off

        GENERIC MAP(
                      DCSMODE  => "POS"
                      )
-- synthesis translate_on

        PORT MAP      (
                      SEL            =>  clksel,
                      CLK0           =>  dcsclk0,
                      CLK1           =>  sysclk1,
                      DCSOUT         =>  dcsclk
);
```

Technical Support Assistance

Hotline: 1-800-LATTICE (North America)
+1-503-268-8001 (Outside North America)
e-mail: techsupport@latticesemi.com
Internet: www.latticesemi.com

Revision History

Date	Version	Change Summary
June 2004	01.0	Initial release.
October 2004	01.1	RST description with timing diagram added.
		Primitive 'Wake_On_Lock' removed.
		Added RST input in EPLLB.
		Fin min 33 replaced with 25.
		FB_MODE default = CLOCKTREE
		CLKOP_DIV values for EHXPLLB 2, 4, 8,16, 32.
December 2004	02.0	Appendices C and D integrated to body of the document.
		DCS source code example moved to Appendix A.
January 2005	03.0	LatticeXP information added.
		Figures 6 and 7 updated.
		CLKOP_FREQ, CLKOK_FREQ user attributes added.
		FB_MODE added.
October 2005	04.0	Clock Distribution section added.
		Example code section removed and referred to help file.
		GUI screen shots updated.
		MM/IP Manager renamed as IPexpress.
		Epllb definition section removed.
		Clkos/clkok select attributes added.
September 2006	04.1	Detailed clock distribution information added.

Appendix A. Clock Preferences

A few key clock preferences are introduced below. Refer to the 'Help' file for other preferences and detailed information.

ASIC

The following preference command assigns a phase of 90° to the CIMDLLA CLKOP.

```
ASIC "my_pll" TYPE "EXHXPLLB" CLKOS_PHASE=90;
```

FREQUENCY

The following physical preference command assigns a frequency of 100 MHz to a net named clk1.

```
FREQUENCY NET "clk1" 100 MHz;
```

The following preference specifies a hold margin value for each clock domain.

```
FREQUENCY NET "RX_CLKA_CMOS_c" 100.000 MHz HOLD_MARGIN 1 ns;
```

MAXSKEW

The following command assigns a maximum skew of 5 ns to a net named NetB.

```
MAXSKEW NET "NetB" 5 NS;
```

MULTICYCLE

The following command will relax the period to 50 ns for the path starting at COMPA to COMPB (NET1).

```
MULTICYCLE "PATH1" START COMP "COMPA" END COMP "COMPB" NET "NET1" 50 NS ;
```

PERIOD

The following command assigns a clock period of 30 ns to the port named Clk1.

```
PERIOD PORT "Clk1" 30 NS;
```

PROHIBIT

This command prohibits the use of a primary clock to route a clock net named bf_clk.

```
PROHIBIT PRIMARY NET "bf_clk";
```

CLOCK_TO_OUT

Specifies a maximum allowable output delay relative to a clock.

Below are two preferences using both the CLKPORT and CLKNET keywords showing the corresponding scope of TRACE reporting.

The CLKNET will stop tracing the path before the PLL, so you will not get PLL compensation timing numbers.

```
CLOCK_TO_OUT PORT "RxAddr_0" 6.000000 ns CLKNET "pll_rxclk" ;
```

The above preference will yield the following clock path:

```
Physical Path Details:

     Clock path pll_inst/pll_utp_0_0 to PFU_33:

   Name    Fanout   Delay (ns)        Site             Resource

 ROUTE       49     2.892    ULPPLL.MCLK to     R3C14.CLK0 pll_rxclk

             --------

               2.892   (0.0% logic, 100.0% route), 0 logic levels.
```

If CLKPORT is used, the trace is complete back to the clock port resource and provides PLL compensation timing numbers.

```
CLOCK_TO_OUT PORT "RxAddr_0" 6.000000 ns CLKPORT "RxClk" ;
```

The above preference will yield the following clock path:

```
Clock path RxClk to PFU_33:

   Name    Fanout   Delay (ns)        Site             Resource

 IN_DEL      ---     1.431        D5.PAD to       D5.INCK RxClk

 ROUTE        1     0.843       D5.INCK to   ULPPLL.CLKIN RxClk_c

 MCLK_DEL    ---     3.605   ULPPLL.CLKIN to    ULPPLL.MCLK pll_inst/pll_utp_0_0

 ROUTE       49     2.892    ULPPLL.MCLK to     R3C14.CLK0 pll_rxclk

             --------

               8.771   (57.4% logic, 42.6% route), 2 logic levels.
```

INPUT_SETUP

Specifies an setup time requirement for input ports relative to a clock net.

```
INPUT_SETUP PORT "datain" 2.000000 ns HOLD 1.000000 ns CLKPORT "clk" PLL_PHASE_BACK ;
```

PLL_PHASE_BACK

This preference is used with INPUT_SETUP when the user wants a trace calculation based on the previous clock edge.

This preference is useful when setting the PLL output phase adjustment. Since there is no negative phase adjustment provided, the PLL_PHASE_BACK preference works as if negative phase adjustment is available.

For example:

If phase adjustment of -90° of CLKOS is desired, the user can set the phase to 270° and set the INPUT_SETUP preference with PLL_PHASE_BACK.