MachXO4 sysCLOCK PLL Design User Guide

Technical Note

FPGA-TN-02391-1.0

December 2025

Disclaimers

Lattice makes no warranty, representation, or guarantee regarding the accuracy of information contained in this document or the suitability of its products for any particular purpose. All information herein is provided AS IS, with all faults, and all associated risk is the responsibility entirely of the Buyer. The information provided herein is for informational purposes only and may contain technical inaccuracies or omissions, and may be otherwise rendered inaccurate for many reasons, and Lattice assumes no obligation to update or otherwise correct or revise this information. Products sold by Lattice have been subject to limited testing and it is the Buyer's responsibility to independently determine the suitability of any products and to test and verify the same. LATTICE PRODUCTS AND SERVICES ARE NOT DESIGNED, MANUFACTURED, OR TESTED FOR USE IN LIFE OR SAFETY CRITICAL SYSTEMS, HAZARDOUS ENVIRONMENTS, OR ANY OTHER ENVIRONMENTS REQUIRING FAIL-SAFE PERFORMANCE, INCLUDING ANY APPLICATION IN WHICH THE FAILURE OF THE PRODUCT OR SERVICE COULD LEAD TO DEATH, PERSONAL INJURY, SEVERE PROPERTY DAMAGE OR ENVIRONMENTAL HARM (COLLECTIVELY, "HIGH-RISK USES"). FURTHER, BUYER MUST TAKE PRUDENT STEPS TO PROTECT AGAINST PRODUCT AND SERVICE FAILURES, INCLUDING PROVIDING APPROPRIATE REDUNDANCIES, FAIL-SAFE FEATURES, AND/OR SHUT-DOWN MECHANISMS. LATTICE EXPRESSLY DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY OF FITNESS OF THE PRODUCTS OR SERVICES FOR HIGH-RISK USES. The information provided in this document is proprietary to Lattice Semiconductor, and Lattice reserves the right to make any changes to the information in this document or to any products at any time without notice.

Inclusive Language

This document was created consistent with Lattice Semiconductor's inclusive language policy. In some cases, the language in underlying tools and other items may not yet have been updated. Please refer to Lattice's inclusive language FAQ 6878 for a cross reference of terms. Note in some cases such as register names and state names it has been necessary to continue to utilize older terminology for compatibility.

Contents

Figures

Tables

Abbreviations in This Document

A list of abbreviations used in this document.

Abbreviation	Definition
CLKDIVC	Clock Dividers
DCCA	Dynamic Clock Control
DCMA	Dynamic Clock Multiplexer
DDR	Double Data Rate
ECLKSYNCA	Edge Clock Synchronization
EFB	Embedded Function Block
FPGA	Field-Programmable Gate Array
GPLL	General Purpose PLL
IP	Intellectual Property
JTAG	Joint Test Action Group
LVDS	Low-Voltage Differential Signaling
NVCM	Non-Volatile Configuration Memory
OSCH	Internal Oscillator
PLL	Phase-Locked Loop
PLLREFCS	PLL Reference Clock Switch
SPI	Serial Peripheral Interface
VCO	Voltage-Controlled Oscillator

1. Introduction

MachXO4™ devices support a variety of I/O interfaces such as display interfaces (7:1 LVDS) and high-speed DDR interfaces with gearing. To support applications that use these interfaces, the MachXO4 device architecture is designed to include advanced clocking features that are typically found in higher density FPGAs. These features provide you the ability to synthesize clocks, minimize clock skew, improve performance, and manage power consumption.

This technical note describes the clock resources available in MachXO4 devices. Details are provided for primary clocks, edge clocks, clock dividers, sysCLOCK™ PLLs, DCC elements, the secondary high fan-out nets, and the internal oscillator available in MachXO4 devices.

Table 1.1. Number of PLLs, Edge Clocks, and Clock Dividers

Parameter	Description	LFMXO4-010	LFMXO4-015	LFMXO4-025	LFMXO4-050	LFMXO4-080	LFMXO4-110
Number of PLLs	General purpose PLLs	1	1	1	2	2	2
Number of edge clocks	Edge clocks for high-speed applications (top and bottom sides)	4	4	4	4	4	4
Number of clock dividers	Clock dividers for DDR applications	4	4	4	4	4	4

2. Clock/Control Distribution Network

MachXO4 devices provide global clock distribution in the form of eight global primary clocks and eight secondary high fan-out nets. Two edge clocks are provided on the top and bottom sides. Other clock sources include clock input pins, internal nodes, PLLs, clock dividers, and the internal oscillator.

3. MachXO4 Top Level View

A top-level view of the major clocking resources for a MachXO4 device is shown in the figure below.

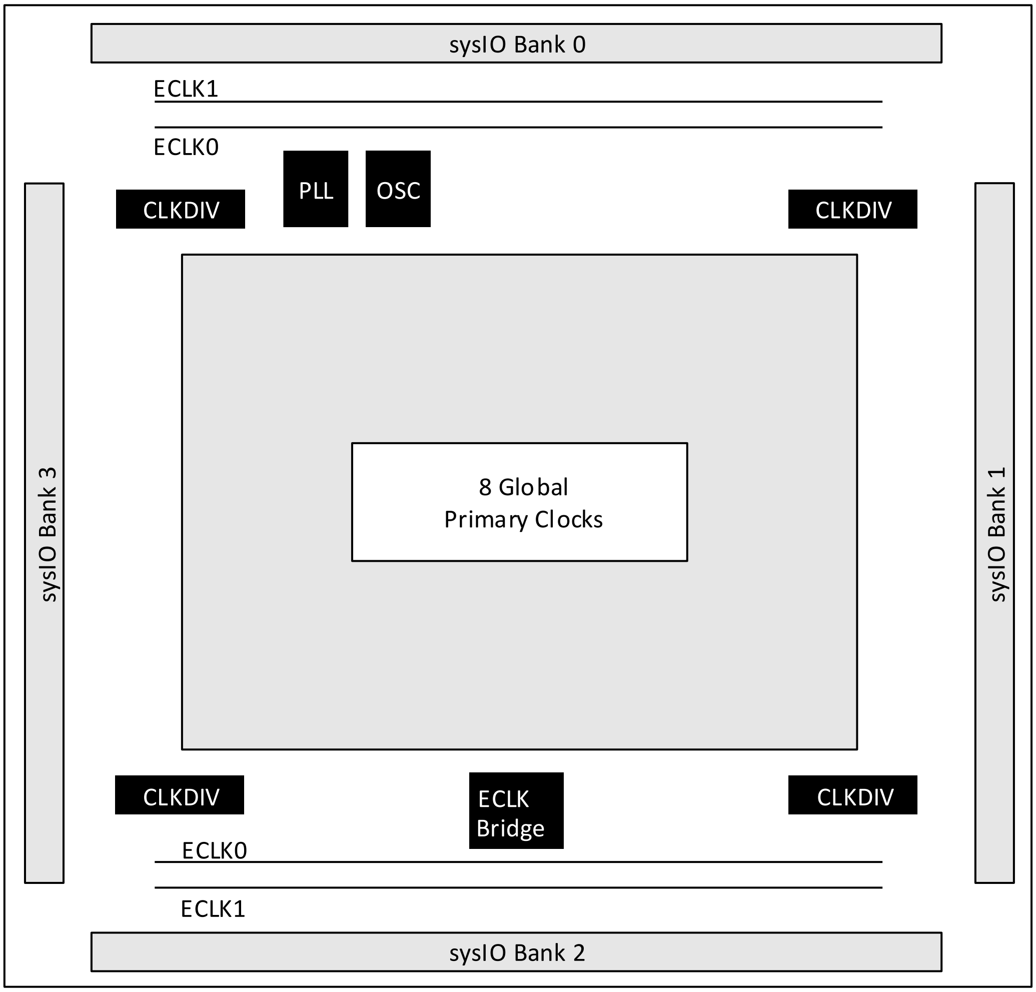


Figure 3.1. MachXO4 Clocking Structure

4. Primary Clocks

A MachXO4 device has eight global primary clocks. The primary clock networks provide a low skew clock distribution path across the chip for high fan-out signals. Two of the primary clocks are equipped with a dynamic clock multiplexer (DCMA) feature that provides the ability to switch between two different clock sources.

The sources of the primary clocks are as follows:

- Dedicated clock pins
- PLL outputs
- CLKDIV outputs
- Internal nodes

5. Dynamic Clock Multiplexer (DCMA)

MachXO4 devices have two dynamic clock multiplexers (DCMA) that allow a design to dynamically switch between two independent primary clock signals. The output of the DCMA is to the primary clock distribution network. The inputs to the DCMA can be any of the clock sources available to the primary clock network.

The DCMA is a simple clock buffer with a multiplexer function. There is no synchronization of the clock signals when switching occurs, hence a glitch may occur.

5.1. DCMA Primitive Definition

The DCMA primitive can be instantiated in the source code of a design as defined in this section.

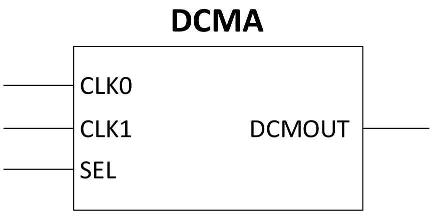


Figure 5.1. DCMA Primitive Symbol

Table 5.1. DCMA Primitive Port Definition

Port Name	I/O	Description
CLK0	I	Clock input port zero – default
CLK1	I	Clock input port one
SEL	I	Select port • SEL=0 for CLK0 • SEL=1 for CLK1
DCMOUT	O	Clock output port

5.2. DCMA Declaration in VHDL Source Code

Library Instantiation

```
library lfmxo4;
use lfmxo4.all;
```

Component Declaration

```
component DCMA
port (CLK0: in std_logic;
      CLK1: in std_logic;
      SEL: in std_logic;
      DCMOUT:out std_logic);
end component;
```

DCMA Instantiation

```
I1: DCMA
port map (CLK0 => CLK0,
CLK1 => CLK1,
SEL => SEL,
DCMOUT => DCMOUT);
```

5.3. DCMA Usage with Verilog Source Code

Component Declaration

```
module DCMA (CLK0, CLK1, SEL, DCMOUT);
input CLK0;
input CLK1;
input SEL;
output DCMOUT;
endmodule
```

DCMA Instantiation

```
DCMA I1 (.CLK0 (CLK0),
        .CLK1 (CLK1),
        .SEL (SEL),
        .DCMOUT (DCMOUT));
```

6. Dynamic Clock Control (DCCA)

MachXO4 devices have a dynamic clock control feature that is available for each of the primary clock networks. The dynamic clock control (DCCA) allows each primary clock to be disabled from the associated core logic in the design when clock is not needed and thus saves power.

6.1. DCCA Primitive Definition

The DCCA primitive can be instantiated in the source code of a design as defined in this section.

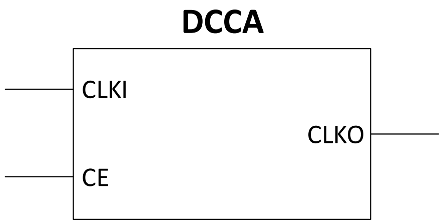


Figure 6.1. DCCA Primitive Symbol

Table 6.1. DCCA Primitive Port Definition

Port Name	I/O	Description
CLKI	I	Clock input
CE	I	Clock enable port • CE = 0 – disabled • CE = 1 – enabled
CLKO	O	Clock output port

6.2. DCCA Declaration in VHDL Source Code

Library Instantiation

```
library lfmxo4;
use lfmxo4.all;
```

Component Declaration

```
component DCCA
port(CLKI: in std_logic;
        CE:in std_logic;
        CLKO: out std_logic);
end component;
```

DCCA Instantiation

```
I1: DCCA
 port map(CLKI=> CLKI,
        CE => CE,
        CLKO=>CLKO);
end component;
```

6.3. DCCA Usage with Verilog Source Code

Component Declaration

```
module DCCA (CLKI, CE, CLKO);
        input CLKI;
        input CE;
        output CLKO;
endmodule
```

DCCA Instantiation

```
DCCA I1(.CLKI (CLKI),
        .CE (CE),
        .CLKO (CLKO));
```

7. Edge Clocks

There are two edge clock resources on the top and bottom sides of the device. These clocks, which have low injection time and skew, are used to clock I/O registers. Edge clock resources are designed for high-speed I/O interfaces with high fan-out capability. Refer to Appendix B. Edge Clock Sources and Connectivity for detailed information on the ECLK locations and connectivity.

The sources of edge clocks are as follows:

- Dedicated clock pins
- PLL outputs
- Internal nodes

7.1. Edge Clock Bridge

MachXO4 devices have an edge clock bridge to enhance communication of ECLKs across the device. The bridge allows an input on the bottom of the device to drive the edge clock on the top edge of the device with minimal skew. Edge clock sources can either go through the edge clock bridge to connect to the edge clock or can be directly connected using the shortest path.

The edge clock bridge is primarily intended for use with high-speed data interfaces such as DDR or 7:1 LVDS Video. For more information on the use of the edge clock bridge, see the MachXO4 Implementing High-Speed I/O Interface User Guide (FPGA-TN-02410).

In the edge clock bridge, there is a clock select multiplexer that allows a design to switch between two different clock sources for each edge clock. This clock select multiplexer is modeled using the ECLKBRIDGECS primitive. A block diagram of the edge clock bridge is shown in Appendix B. Edge Clock Sources and Connectivity.

8. ECLKBRIDGECS Primitive Definition

The ECLKBRIDGECS primitive can be instantiated in the source code of a design as defined in this section. A design can have up to two instantiations of ECLKBRIDGECS primitives if desired.

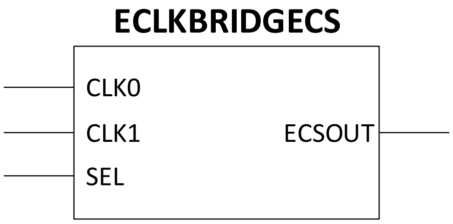


Figure 8.1. ECLKBRIDGECS Primitive Symbol

Table 8.1. ECLKBRIDGECS Primitive Port Definition

Port Name	I/O	Description
CLK0	I	Clock input port zero – default
CLK1	I	Clock input port one
SEL	I	Select port • SEL=0 for CLK0 • SEL=1 for CLK1
ECSOUT	O	Clock output port

8.1. ECLKBRIDGECS Declaration in VHDL Source Code

Library Instantiation

```
library lfmxo4;
use lfmxo4.all;
```

Component Declaration

```
component ECLKBRIDGECS
 port (CLK0:in std_logic;
  CLK1: in std_logic;
  SEL: in std_logic;
  ECSOUT: out std_logic);
end component;
```

ECLKBRIDGECS Instantiation

```
I1: ECLKBRIDGECS
 port map (CLK0=>CLK0,
  CLK1 => CLK1,
  SEL => SEL,
  ECSOUT => ECSOUT);
```

8.2. ECLKBRIDGECS Usage with Verilog Source Code

Component Declaration

```
module ECLKBRIDGECS (CLK0, CLK1, SEL, ECSOUT);

  input CLK0;
  input CLK1;
  input SEL;
  output ECSOUT;

endmodule
```

ECLKBRIDGECS Instantiation

```
ECLKBRIDGECS I1 (.CLK0 (CLK0),
        .CLK1 (CLK1),
        .SEL (SEL),
    .ECSOUT (ECSOUT));
```

9. Edge Clock Synchronization (ECLKSYNCA)

MachXO4 devices have a dynamic edge clock synchronization control (ECLKSYNCA). This feature allows each edge clock to be disabled from core logic if desired. You can use this feature to synchronize the edge clock to an event or external signal if desired. You can also use this feature to design applications in which a clock and the associated logic can be dynamically disabled to save power.

9.1. ECLKSYNCA Primitive Definition

The ECLKSYNCA primitive can be instantiated in the source code of a design as defined in this section.

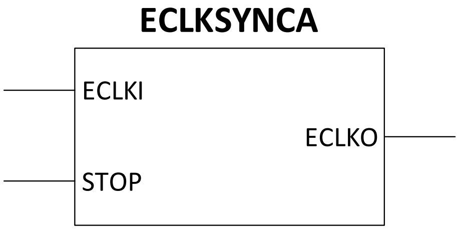


Figure 9.1. ECLKSYNCA Primitive Symbol

Table 9.1. ECLKSYNCA Primitive Port Definition

Port Name	I/O	Description
ECLK	I	Clock input port
STOP	I	Control signal to stop edge clock • STOP=0 – Clock is active • STOP=1 – Clock is off
ECLKO	O	Clock output port

9.2. ECLKSYNCA Declaration in VHDL Source Code

Library Instantiation

```
library lfmxo4;
use lfmxo4.all;
```

Component Declaration

```
component ECLKSYNCA
port (ECLKI : in std_logic;
       STOP : in std_logic;
       ECLKO : out std_logic);
end component;
```

ECLKSYNCA Instantiation

```
I1: ECLKSYNCA
port map(ECLKI => ECLKI,
       STOP => STOP,
       ECLKO => ECLKO);
```

9.3. ECLKSYNCA Usage with Verilog Source Code

Component Declaration

```
module ECLKSYNCA (ECLKI, STOP, ECLKO);
input ECLKI;
input STOP;
output ECLKO;
endmodule
```

ECLKSYNCA Instantiation

```
ECLKSYNCA I1 (.ECLKI (ECLKI),
       .STOP (STOP),
      .ECLKO (ECLKO));
```

10. Clock Dividers (CLKDIVC)

MachXO4 devices have four clock dividers. Each clock divider provides two outputs. One is the same frequency as the input clock and the other is the input clock divided by either 2, 3.5, or 4. Both outputs have matched input-to-output delay. The input to the clock divider is the output from the edge clock multiplexer. The outputs of the clock divider drive the primary clock network and are available for general purpose routing or secondary clocks.

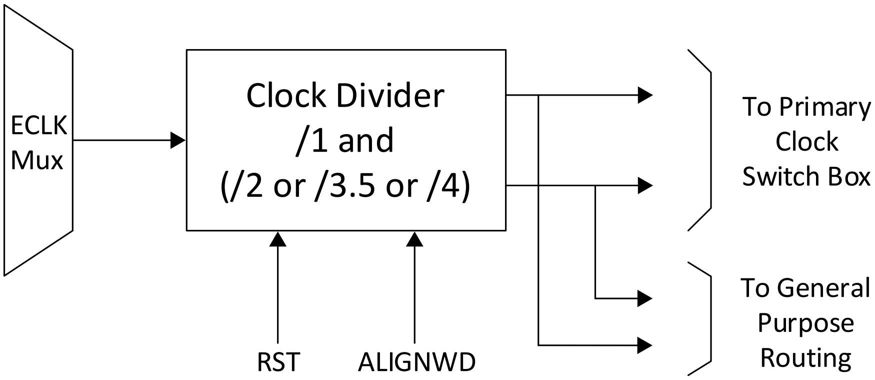

ECLK Mux
Clock Divider /1 and (/2 or /3.5 or /4)
RST
ALIGNWD
To Primary Clock Switch Box
To General Purpose Routing

Figure 10.1. MachXO4 Clock Divider

10.1. CLKDIVC Primitive Definition

The CLKDIVC primitive can be instantiated in the source code of a design as defined in this section.

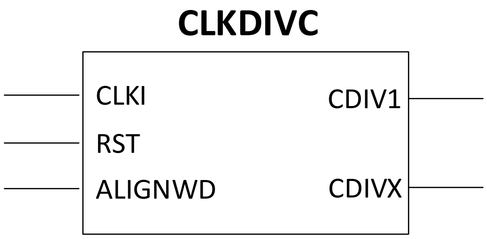


Figure 10.2. CLKDIVC Primitive Symbol

Table 10.1. CLKDIVC Primitive Port Definition

Port Name	I/O	Description
CLKI	I	Clock input
RST	I	Reset input – asynchronously forces all outputs low • RST = 0 – Clock output outputs are active • RST = 1 – Clock output outputs are off
ALIGNWD	I	Signal is used for word alignment ALIGNWD = 0 when not used See the MachXO4 Implementing High-Speed I/O Interface User Guide (FPGA-TN-02410) for more information
CDIV1	O	Divides by 1 output port. When RST = 1 CDIV1 output does not toggle and stays either L or H
CDIVX	O	Divides by 2, 3.5, or 4 output port

Table 10.2. CLKDIVC Primitive Attribute Definition

Name	Description	Value	Default
GSR	GSR Enable	ENABLED, DISABLED	DISABLED
DIV	CLK Divider	2.0, 3.5, or 4.0	2.0

The ALIGNWD input is intended for use with high-speed data interfaces such as DDR or 7:1 LVDS video. For more information on the use of ALIGNWD, see the MachXO4 Implementing High-Speed I/O Interface User Guide (FPGA-TN-02410).

10.2. CLKDIVC Declaration in VHDL Source Code

Library Instantiation

```
library lfmxo4;
use lfmxo4.all;
```

Component and Attribute Declaration

```
component CLKDIVC
generic (DIV : string;
        GSR : string);
port (RST: in std_logic;
        CLKI: in std_logic;
        ALIGNWD: in std_logic;
        CDIV1: out std_logic;
        CDIVX : out std_logic);
end component;
```

CLKDIVC Instantiation

```
I1: CLKDIVC
generic map (DIV => "2.0",
        GSR => "DISABLED")
port map (RST => RST,
        CLKI => CLKI,
        ALIGNWD => ALIGNWD,
        CDIV1 => CDIV1,
        CDIVX = > CDIVX);
```

10.3. CLKDIVC Usage with Verilog Source Code

Component and Attribute Declaration

```
module CLKDIVC (RST, CLKI, ALIGNWD, CDIV1, CDIVX);
parameter DIV = "2.0";// "2.0", "3.5", "4.0"
parameter GSR = "DISABLED";// "ENABLED", "DISABLED"
input RST;
input CLKI;
input ALIGNWD;
output CDIV1;
output CDIVX;
endmodule
```

CLKDIVC Instantiation

```
defparam I1.DIV = "2.0";
defparam I1.GSR = "DISABLED";

CLKDIVC I1 (.RST (RST),
        .CLKI(CLKI),
        .ALIGNWD (ALIGNWD),
        .CDIV1 (CDIV1),
    .CDIVX (CDIVX));
```

11. sysCLOCK PLL

The MachXO4 PLL provides features such as clock injection delay removal, frequency synthesis, and phase adjustment.

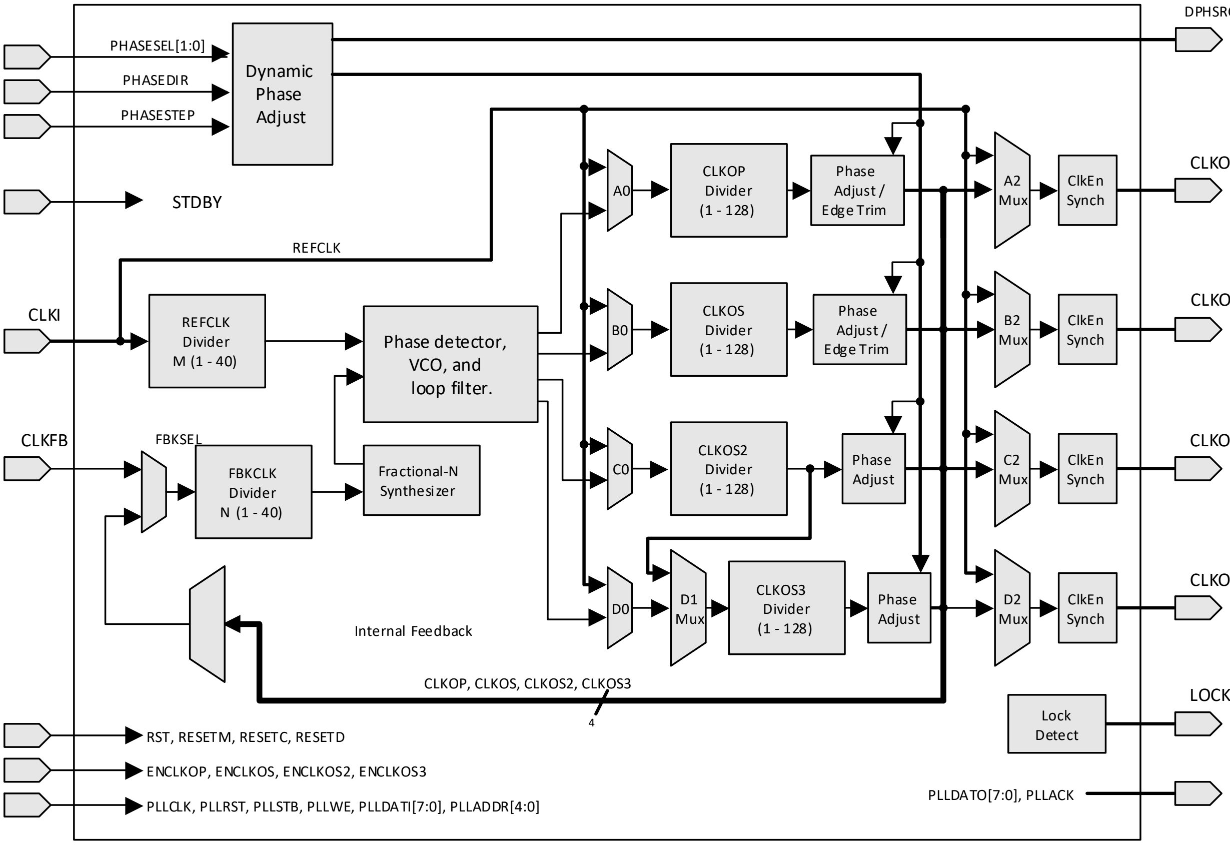


Figure 11.1. MachXO4 PLL Block Diagram

11.1. Functional Description

11.1.1. PLL Divider Blocks

Input Clock (CLKI) Divider

The CLKI divider is used to control the input clock frequency into the PLL block. The divider setting directly corresponds to the divisor of the output clock. The input must be within the input frequency range specified in the MachXO4 Family Data Sheet (FPGA-DS-02125). The output of the input divider must also be within the phase detector frequency range specified in the 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.

The output of the feedback divider must be within the phase detector frequency range specified in the MachXO4 Family Data Sheet (FPGA-DS-02125).

Output Clock Dividers (CLKOP, CLKOS, CLKOS2, CLKOS3)

The output clock dividers allow the VCO frequency to be scaled up to the 400–800 MHz range, which minimizes jitter. Each of the output dividers is independent of the other dividers and each uses the VCO as the source by default. Each of the output dividers can be set to a value of 1 to 128. The CLKOS2 and CLKOS3 dividers can be cascaded together to produce a lower frequency output if desired.

Phase Adjustment (Static Mode)

The CLKOP, CLKOS, CLKOS2, and CLKOS3 outputs can be phase adjusted relative to the input clock. The phase adjustments can be done in 45° steps. The clock output selected as the feedback cannot use the static phase adjustment feature.

Phase Adjustment (Dynamic Mode)

The phase adjustments can be controlled in a dynamic mode using the PHASESEL, PHASEDIR, and PHASESTEP ports. The clock output selected as the feedback cannot use the dynamic phase adjustment feature. See the Dynamic Phase Adjustment section for more details.

Phase Alignment

After the device reaches the steady state operation after power-up, and after releasing RST and RESETM, the CLKOP and CLKOS outputs are edge aligned (for related frequencies) when phase adjustment is set to 0°. Under the same conditions, CLKOS2 and CLKOS3 are aligned to CLKOP and CLKOS to within one VCO clock period.

Edge Trim Adjustment (Static Mode)

The CLKOP and CLKOS ports can be finely tuned with an edge trim adjustment feature.

11.2. PLL Features

11.2.1. Standby Mode

The MachXO4 PLL contains a standby mode that allows the PLL to be placed into a standby state to save power when not needed in the design. The PLL can be powered down completely or partially depending on the needs of the design.

11.2.2. Fractional-N synthesis

The MachXO4 PLL contains a fractional-N synthesis feature that allows you to generate an output clock that is a non-integer multiple of the input frequency. You are allowed to enter a value between 0 and 65535 for the fractional-N divider. This value is then divided by 65536 and the result is added to the feedback divider. A MASH Delta-Sigma modulation technique is used such that the average effective feedback divide value is equal to this value. Fractional-N synthesis can be used to create a closer PPM match to the target frequency.

11.2.3. WISHBONE Ports

The MachXO4 PLL contains a WISHBONE port feature that allows the PLL settings to be dynamically changed from the user logic. When using this feature, the EFB block must also be instantiated in the design to allow access to the WISHBONE ports. The WISHBONE ports of the PLL must be connected to the WISHBONE ports of the EFB block for proper simulation and operation. The use of the WISHBONE ports is described in detail in Appendix D. PLL WISHBONE Bus Operation.

11.3. PLL Inputs and Outputs

11.3.1. CLKI Input

The CLKI signal is the reference clock for the PLL. The signal must conform to the specifications in the data sheet for the PLL to operate correctly. The CLKI signal can come from a dedicated dual-purpose I/O pin, from any I/O pin, or from routing. The dedicated dual-purpose I/O pin provides a low skew input path and is the recommended source for the PLL. The reference clock is divided by the input (M) divider to create one input to the phase detector of the PLL. The dedicated GPLL pins and PCLK pins located on the top and bottom sides provide direct connection to the PLL input. The PCLK pins located on the left and right sides use primary clock routing to connect to the PLL input pin.

11.3.2. CLKFB Input

The CLKFB signal is the feedback signal to the PLL. The feedback signal is used by the PLL to determine if the output clock needs adjustment to maintain the correct frequency, phase, or other characteristic. The CLKFB signal can come from the primary clock net, from a dedicated dual-purpose I/O pin, directly from an output clock divider, or from routing. By using external feedback, you can compensate for board-level clock alignment. The feedback clock signal is divided by the feedback (N) divider to create an input to the phase detector of the PLL. A bypassed PLL output cannot be used as the feedback signal.

11.3.3. 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 can be provided as a part of the PLL module. The RST signal can be driven by an internally-generated reset function or by an I/O pin. This RST signal resets the PLL core (VCO, phase detector, and charge pump) and the output dividers, which causes the outputs to be grounded, even in bypass mode.

After the RST signal is de-asserted, the PLL starts the lock-in process and takes t_{LOCK} time to complete the PLL LOCK. Figure 11.2 shows the timing diagram of the RST input. The RST signal is active high and is optional.

The RST input does not reset the input divider (M-divider) because there may be a clock used externally that is a synchronized to the reference clock. In this case, there is a state relationship between the external clock and the M-divided clock (which the PLL is synchronized to). You need to preserve this relationship when resetting the PLL. In this condition, RST is used to reset the PLL without resetting the M-divider.

11.3.4. RESETM Input

The user-controlled PLL reset signal RESETM can be provided as a part of the PLL module. The RESETM signal can be driven by an internally-generated reset function or by an I/O pin. The RESETM signal resets the PLL core (similar to RST) and all the dividers, including the M-divider. This causes the outputs to be grounded, including when the PLL is in bypass mode.

After the RESETM signal is de-asserted, the PLL starts the lock-in process and takes t_{LOCK} time to complete the PLL LOCK. Figure 11.2 shows the timing diagram of the RESETM input. The RESETM signal is active high and is optional.

To synchronize the PLL output to an external clock source, use the RESET signal to reset the PLL.

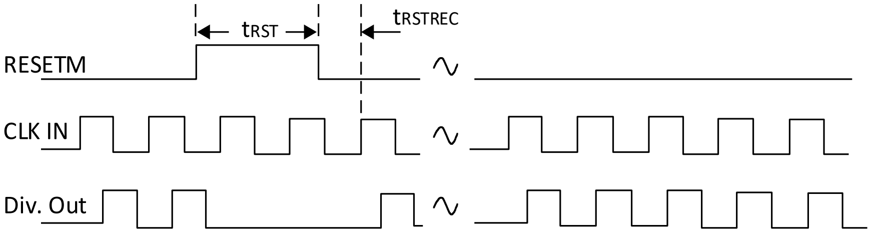


Figure 11.2. RST and RESETM Timing Diagram

11.3.5. RESETC Input

The user-controlled PLL reset signal RESETC can be provided as a part of the PLL module. The RESETC signal can be driven by an internally-generated reset function or by an I/O pin. This RESETC signal resets only the CLKOS2 output divider, which causes the CLKOS2 output to be grounded unless the output is in the bypass mode. If this output is in bypass mode as a clock divider, this output is reset by the RESETC signal. The RESETC signal can be used to synchronize the CLKOS2 output to an external clock signal.

After the RESETC signal is de-asserted, there is a time delay of t_{RSTREC_DIV} before the next clock edge toggles the CLKOS2 output divider. Figure 11.3 shows the timing diagram of the RESETC input. The RESETC signal does not affect the PLL loop unless the CLKOS2 output is used in the feedback path. If the CLKOS2 output is used in the feedback path, it is recommended to use the RST or RESETM signal to reset the PLL instead of RESETC. The RESETC signal is active high and is optional.

11.3.6. RESETD Input

The user-controlled PLL reset signal RESETD can be provided as a part of the PLL module. The RESETD signal can be driven by an internally-generated reset function or by an I/O pin. This RESETD signal resets only the CLKOS3 output divider, which causes the CLKOS3 output to be grounded unless the output is in the bypass mode. If this output is in bypass mode as a clock divider, this output is reset by the RESETD signal. The RESETD signal can be used to synchronize the CLKOS3 output to an external clock signal.

After the RESETD signal is de-asserted, there is a time delay of t_{RSTREC_DIV} before the next clock edge toggles the CLKOS3 output divider. Figure 11.3 shows the timing diagram of the RESETD input. The RESETD signal does not affect the PLL loop unless the CLKOS3 output is used in the feedback path. If the CLKOS3 output is used in the feedback path, it is recommended to use the RST or RESETM signal to reset the PLL instead of RESETD. The RESETD signal is active high and is optional.

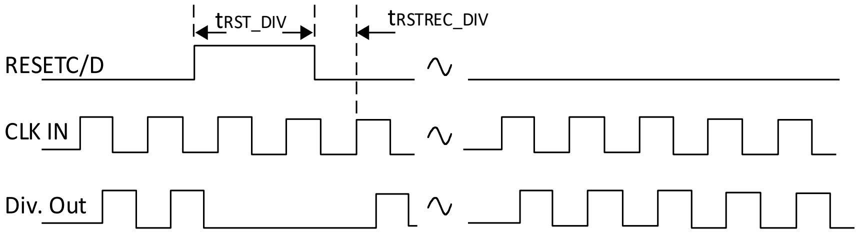


Figure 11.3. RESETC and RESETD Timing Diagram

11.3.7. ENCLKOP Input

The ENCLKOP signal is used to enable and disable the CLKOP output from a user signal. This enables you to save power by stopping the CLKOP output when the output is not used. Additionally, this signal also allows you to synchronize CLKOP with another signal in the design. The ENCLKOP signal is optional and is available only if you select the clock enable ports option in IP Catalog™. If the ENCLKOP signal is not requested, the CLKOP output is always active (when the PLL is instantiated) unless the PLL is in standby mode. The ENCLKOP signal is active high.

11.3.8. ENCLKOS Input

The ENCLKOS signal is used to enable and disable the CLKOS output from a user signal. This enables you to save power by stopping the CLKOS output when the output is not used. Additionally, this signal also allows you to synchronize CLKOS with another signal in the design. The ENCLKOS signal is optional and is available only when the PLL is configured with the CLKOS output and the Clock Enable ports options in IP Catalog. If the PLL is configured with the CLKOS output enabled and the ENCLKOS signal is not requested, the CLKOS output is always active unless the PLL is in standby mode. The ENCLKOS signal is active high.

11.3.9. ENCLKOS2 Input

The ENCLKOS2 signal is used to enable and disable the CLKOS2 output from a user signal. This enables you to save power by stopping the CLKOS2 output when the output is not used. Additionally, this signal allows you to synchronize

CLKOS2 with another signal in the design. The ENCLKOS2 signal is optional and is available only when the PLL is configured with the CLKOS2 output and the Clock Enable ports options in IP Catalog. If the PLL is configured with the CLKOS2 output enabled and the ENCLKOS2 signal is not requested, the CLKOS2 output is always active unless the PLL is in standby mode. The ENCLKOS2 signal is active high.

11.3.10. ENCLKOS3 Input

The ENCLKOS3 signal is used to enable and disable the CLKOS3 output from a user signal. This enables you to save power by stopping the CLKOS3 output when the output is not used. Additionally, this signal allows you to synchronize CLKOS3 with another signal in the design. The ENCLKOS3 signal is optional and is available only when the PLL is configured with the CLKOS3 output and the Clock Enable ports options in IP Catalog. If the ENCLKOS3 signal is not requested, the CLKOS3 output is always active unless the PLL is in standby mode. The ENCLKOS3 signal is active high.

11.3.11. STDBY Input

The STDBY signal is used to put the PLL into a low power standby mode when PLL is not required. The STDBY port can be connected to the power controller so that the PLL enters the low power state when device is driven to the standby mode. Alternatively, the STDBY port can be driven by user logic independent of the standby mode. The STDBY signal is optional and is available only if you select the Standby ports option in IP Catalog. The STDBY signal is active high.

11.3.12. PHASESEL Input

The PHASESEL[1:0] input is used to specify which PLL output port is affected by the dynamic phase adjustment ports. The settings available are shown in the Dynamic Phase Adjustment section. The PHASESEL signal must be stable before the PHASESTEP signal is toggled. The PHASESEL signal is optional and is available only if you select the Dynamic Phase ports option in IP Catalog.

11.3.13. PHASEDIR Input

The PHASEDIR input is used to specify which direction the dynamic phase shift occurs, advanced (leading) or delayed (lagging). When PHASEDIR = 0, the phase shift is delayed from the current clock by one step. When PHASEDIR = 1, the phase shift is advanced from the current clock by one step. The PHASEDIR signal must be stable before the PHASESTEP signal is toggled. The PHASEDIR signal is optional and is available only if you select the Dynamic Phase ports option in IP Catalog.

11.3.14. PHASESTEP Input

The PHASESTEP signal is used to initiate the dynamic phase adjustment for the clock output port and in the direction specified by the PHASESEL and PHASEDIR inputs respectively. The PHASESTEP signal is optional and is available only if you select the Dynamic Phase ports option in IP Catalog.

11.3.15. CLKOP Output

CLKOP is the main clock output of the sysCLOCK PLL. This signal is always available by default and can be routed to the primary clock network of the chip. The CLKOP output can also be routed to top and bottom edge clocks. The CLKOP output can be phase-shifted either statically or dynamically and can also be used with the duty trim adjustment feature. The CLKOP signal output can either come from the CLKOP output divider or can bypass the PLL. When CLKOP is in the bypass mode, the output divider can either be bypassed or used in the circuit.

11.3.16. CLKOS Output

The secondary clock output of the sysCLOCK PLL is the CLKOS signal. This signal is available when you select the signal and can be routed to the primary clock network of the device. The CLKOS output can also be routed to top and bottom edge clocks. The CLKOS output can be phase-shifted either statically or dynamically and can also be used with the duty trim adjustment feature. The CLKOS signal output can either come from the CLKOS output divider or can bypass the PLL. When CLKOS is in the bypass mode, the output divider can either be bypassed or used in the circuit. The CLKOS signal is optional.

11.3.17. CLKOS2 Output

The CLKOS2 signal is another secondary clock output that is available in the sysCLOCK PLL. This signal is available when you select the signal and can be routed to the primary clock network of the chip. The CLKOS2 output cannot be routed to top and bottom edge clocks. The CLKOS2 output can be phase-shifted either statically or dynamically but does not have the duty trim adjustment feature. The CLKOS2 signal output can either come from the CLKOS2 output divider or can bypass the PLL. When CLKOS2 is in the bypass mode, the output divider can either be bypassed or used in the circuit. The CLKOS2 signal is optional.

11.3.18. CLKOS3 Output

The CLKOS3 signal is another secondary clock output that is available in the sysCLOCK PLL. This signal is available when you select the signal and can be routed to the primary clock network of the chip. The CLKOS3 output cannot be routed to top/bottom edge clocks. The CLKOS3 output can be phase-shifted either statically or dynamically but does not support the duty trim adjustment feature. The CLKOS3 signal output can either come from the CLKOS3 output divider or can bypass the PLL. When CLKOS3 is in the bypass mode, the output divider can either be bypassed or used in the circuit. The CLKOS3 signal is optional.

The CLKOS3 output also supports lower frequency outputs that require an output divider value larger than 128. This is accomplished by cascading the CLKOS2 and CLKOS3 output dividers. When used in this application, the CLKOS2 output cannot be used as an independent clock output. A cascaded clock output cannot be used for the feedback signal of the PLL.

11.3.19. DPHSRC Output

The DPHSRC output is used to indicate whether the dynamic phase ports or the WISHBONE registers are used to control the dynamic phase adjustment feature. The dynamic phase ports are the PHASESEL, PHASEDIR, and PHASESTEP ports. The DPHSRC signal is optional and is available if you select the Dynamic Phase ports option in IP Catalog. If you do not select the Dynamic Phase ports option, the WISHBONE registers are used to set the dynamic phase adjustment feature by default.

11.3.20. 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 achieves lock within the specified lock time. When lock is achieved, the PLL LOCK signal is asserted.

The LOCK can either be in the normal lock mode or the sticky lock mode. In the normal lock mode, the LOCK signal is asserted when the PLL has achieved lock and de-asserted if a loss of lock is detected. In sticky lock mode, when the LOCK signal is asserted, the signal stays asserted until the PLL reset is asserted or until the PLL is powered down. It is recommended to assert PLL RST to re-synchronize the PLL to the reference clock when the PLL loses lock. The LOCK signal is available to the FPGA routing to implement the generation of the RST signal if you request. The LOCK signal is optional and is available if you select the Provide PLL Lock signal option in IP Catalog.

11.3.21. WISHBONE Ports

The WISHBONE parts are listed in Appendix D. PLL WISHBONE Bus Operation. The WISHBONE ports are optional.

11.4. PLL Attributes

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

11.4.1. FIN

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

11.4.2. CLKI_DIV, CLKFB_DIV, CLKOP_DIV, CLKOS_DIV, CLKOS2_DIV, CLKOS3_DIV

These dividers determine the output frequencies of each of the output clocks. You are not allowed to input an invalid combination when using IP Catalog. Valid combinations are determined by the input frequency, the dividers, and the PLL specifications.

The CLKOP_DIV value is calculated to maximize the FVCO within the specified range based on FIN and CLKOP_FREQ in conjunction with the CLKI_DIV and CLKFB_DIV values. This applies when the CLKOP output is used for the feedback signal. If another output is used for the feedback signal, the corresponding output divider must be calculated.

The output signals that are not used for the feedback signal use an output divider value based on the VCO frequency and the desired output frequency. The divider values for all these dividers ranges from 1 to 128, however, the full range is not allowed in certain conditions as the values do not meet the PLL specifications.

11.4.3. FREQUENCY_PIN_CLKI, FREQUENCY_PIN_CLKOP, FREQUENCY_PIN_CLKOS, FREQUENCY_PIN_CLOS2, FREQUENCY_PIN_CLKOS3

These input and output clock frequencies determine the divider values.

11.4.4. Frequency Tolerance – CLKOP, CLKOS, CLKOS2, CLKOS3

When the desired output frequency is not achievable, you may enter the frequency tolerance of the clock output.

12. MachXO4 PLL Primitive Definition

The PLL primitive can be instantiated in the source code of a design as defined in this section.

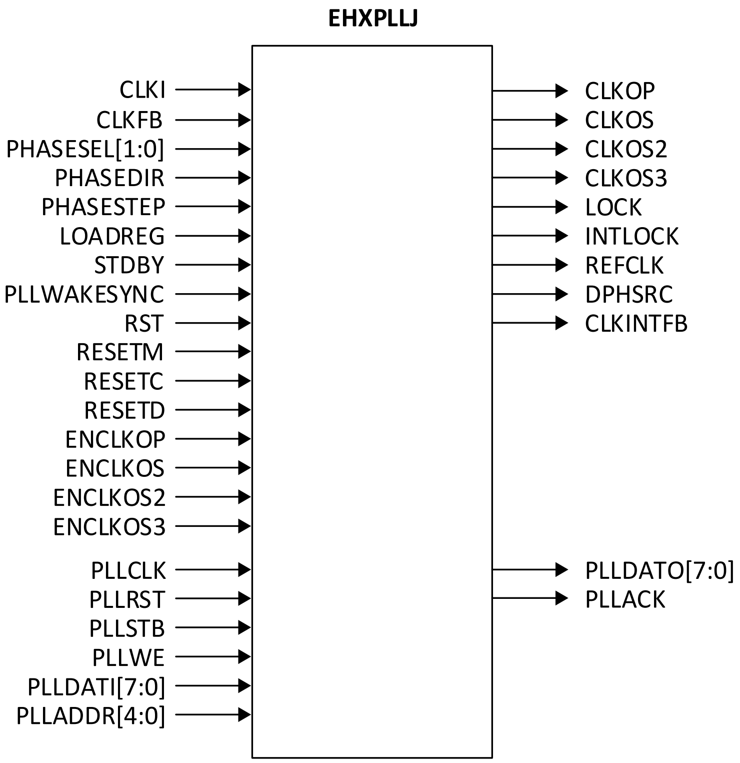


Figure 12.1. PLL Primitive Symbol

Table 12.1. PLL Primitive Port Definition

Port Name	I/O	Description
CLKI	I	Input clock to PLL
CLKFB	I	Feedback clock
PHASESEL[1:0]	I	Selects which output is affected by dynamic phase adjustment ports
PHASEDIR	I	Dynamic phase adjustment direction
PHASESTEP	I	Dynamic phase step – toggles shifts VCO phase adjust by one step
LOADREG	I	Dynamic phase load – toggles loads divider phase adjustment values into PLL
CLKOP	O	Primary PLL output clock (with phase shift adjustment)
CLKOS	O	Secondary PLL output clock (with phase shift adjust)
CLKOS2	O	Secondary PLL output clock2 (with phase shift adjust)
CLKOS3	O	Secondary PLL output clock3 (with phase shift adjust)
LOCK	O	PLL LOCK, asynchronous signal. Active high indicates PLL is locked to input and feedback signals.
INTLOCK	O	PLL internal LOCK, asynchronous signal. Active high indicates PLL lock using internal feedback.[1]
REFCLK	O	Output of reference clock multiplexer
DPHSRC	O	Dynamic phase source – ports or WISHBONE is active
STDBY	I	Standby signal to power down the PLL
PLLWAKESYNC	I	PLL wake-up sync. Enable PLL to switch from internal to user feedback path when the PLL wakes up.[1]
RST	I	PLL reset without resetting the M-divider. Active high reset.
RESETM	I	PLL reset - includes resetting the M-divider. Active high reset.

Port Name	I/O	Description
RESETC	I	Reset for CLKOS2 output divider only. Active high reset.
RESETD	I	Reset for CLKOS3 output divider only. Active high reset.
ENCLKOP	I	Clock enable for CLKOP output
ENCLKOS	I	Clock enable for CLKOS output – only available if CLKOS port is active
ENCLKOS2	I	Clock enable for CLKOS2 output – only available if CLKOS2 port is active
ENCLKOS3	I	Clock enable for CLKOS3 output – only available if CLKOS3 port is active
PLLCLK	I	PLL data bus clock input signal
PLLRST	I	PLL data bus reset. This resets only the data bus not any register values.
PLLSTB	I	PLL data bus strobe signal
PLLWE	I	PLL data bus write enable signal
PLLADDR [4:0]	I	PLL data bus address
PLLDATI [7:0]	I	PLL data bus data input
PLLDATO [7:0]	O	PLL data bus data output
PLLACK	O	PLL data bus acknowledge signal

Note:

1. The PLLWAKWSYNC and INTLOCK primitive ports are not shown in the module level when IP Catalog is used to generate the PLL. The ports are tied off in the module.

13. Dynamic Phase Adjustment

The MachXO4 PLL supports dynamic phase adjustments through either the dynamic phase adjust ports or the WISHBONE interface. For more details about the WISHBONE interface, refer to Appendix D. PLL WISHBONE Bus Operation.

To use the dynamic phase adjustment feature, the PHASESEL[1:0], PHASEDIR, and PHASESTEP ports and signals are used. The DPHSRC port is also available and can be used to confirm that the correct signal source, the primitive ports, or WISHBONE signals, and is selected prior to implementing the phase adjustment. The default setting when the dynamic phase ports are selected is to use the primitive ports for dynamic phase adjustments. The source for the dynamic phase adjustments can also be changed from the WISHBONE interface if desired using the MC1_DYN_SOURCE WISHBONE register. If you do not select the dynamic phase ports from the interface, the WISHBONE signals are used for dynamic phase adjustments.

All four output clocks, CLKOP, CLKOS, CLKOS2, and CLKOS3, have the dynamic phase adjustment feature but only one output clock can be adjusted at a time. The table below shows the output clock selection settings available using the PHASESEL[1:0] signal. The PHASESEL signal must be stable before the PHASESTEP signal is toggled.

Table 13.1. PHASESEL Signal Settings Definitions

PHASESEL[1:0]	PLL Output Shifted
00	CLKOS
01	CLKOS2
10	CLKOS3
11	CLKOP

The selected output clock phase is either advanced or delayed depending on the value of the PHASEDIR port or signal. The table below shows the PHASEDIR settings available. The PHASEDIR signal must be stable before the PHASESTEP signal is toggled.

Table 13.2. PHASEDIR Signal Settings Definitions

PHASEDIR	Direction
00	Delayed (lagging)
01	Advanced (leading)

When the PHASESEL and PHASEDIR are set, the phase adjustment is made by toggling the PHASESTEP signal. Each pulse of the PHASESTEP signal generates a phase shift of one step. The PHASESTEP signal pulse must be initiated from a logic zero value and the phase shift is initiated on the negative edge of the PHASESTEP signal. The step size is specified in the equation below.

```
Step size = 45° / Output Divider
```

If the phase shift desired is larger than one step, the PHASESTEP signal can be pulsed several times to generate the desired phase shift. One step size is the smallest phase shift that can be generated by the PLL. The dynamic phase adjustment results in a glitch-free adjustment when delaying the output clock but glitches may result when advancing the output clock.

The following timing diagrams show the setup and hold timing requirements for PHASESEL[1:0] and PHASEDIR with respect to PHASESTEP, when dynamically changing the phase controls signals.

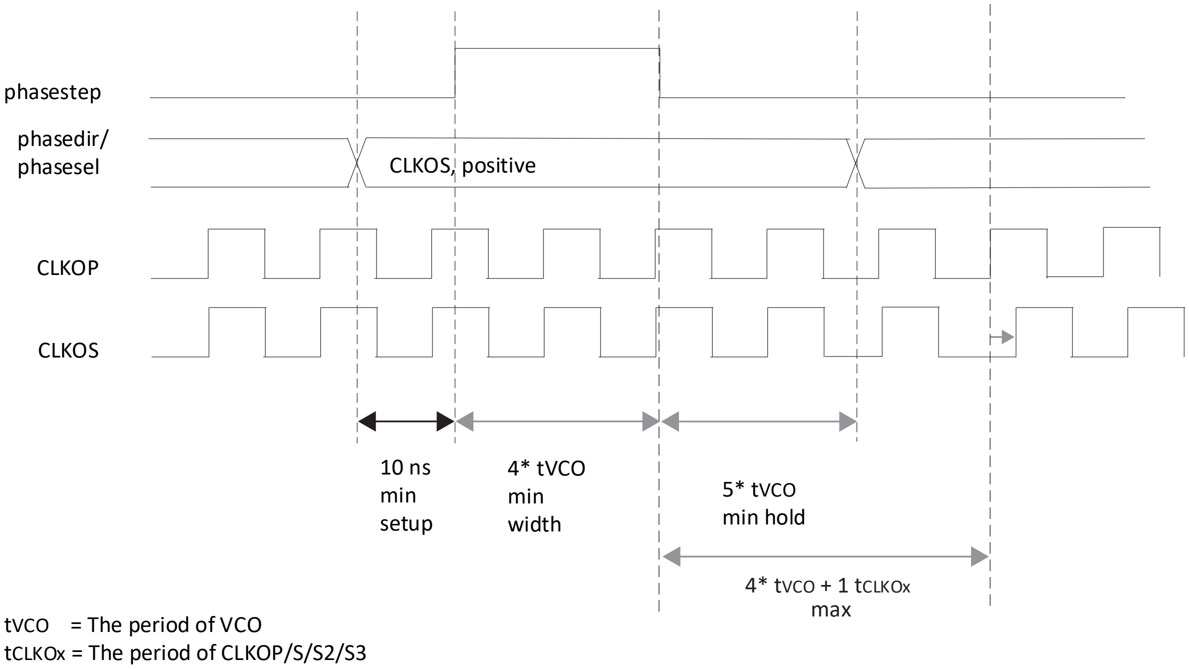


Figure 13.1. GPLL VCO Phase Rotation Timing Diagram

14. Frequency Calculation

The PLL can be used to synthesize a clock frequency that is needed in a design when your board does not have the necessary frequency source. The synthesized frequency can be calculated using the equations below.

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

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

$$f_{PFD} = f_{IN} / M = f_{FB} / N \quad (3)$$

Where:

f_{OUT} is the output frequency.

f_{IN} is the input frequency.

f_{VCO} is the VCO frequency.

f_{PFD} is the PFD (phase detector) frequency.

f_{FB} is the feedback signal frequency.

N is the feedback divider (integer value shown in the IP Catalog user interface).

M is the input divider (integer value shown in the IP Catalog user interface).

V is the output divider (integer value shown in the IP Catalog user interface).

These equations are applicable for the clock output signal that is used for the feedback source to the PLL. When the VCO frequency is calculated from these equations, the frequency can be used to calculate the remaining output clock signals using equation (2).

The equations listed are valid provided that the divider value used for the output and feedback paths are equivalent. If the divider values are not equivalent, equation (1) becomes more complex because the two dividers must be included.

15. Fractional-N Synthesis Operation

The MachXO4 sysCLOCK PLLs support high resolution (16-bit) fractional-N synthesis. Fractional-N frequency synthesis allows you to generate an output clock that is a non-integer multiple of the input frequency. The fractional-N synthesis option is enabled in the IP Catalog interface by selecting **Enable** under **Fractional-N Divider** and entering a number between 0 and 65535. The value that is entered is divided by 65536 to form the fractional part of the feedback divider (also called the N divider) value. The effective feedback divider value is given by the equation:

$$\mathrm{Neff} = N + (F/65536) \qquad (4)$$

Where:

N is the integer feedback divider (shown in the IP Catalog user interface).

F is the value entered in the fractional-N synthesis.

The output frequency is given by the equation:

$$f_{OUT} = (f_{IN}/M) * \mathrm{Neff} \qquad (5)$$

Where:

f_{OUT} is the output frequency.

f_{IN} is the input frequency.

M is the input divider (shown in the IP Catalog user interface).

The fractional-N synthesis works by using a delta-sigma technique to approximate the fractional value that you entered. Therefore, using the fractional-N synthesis option results in higher jitter of the PLL VCO and output clocks compared to using an integer value for the feedback divider. It is recommended that fractional-N synthesis is used only if the N/M divider ratio is 4 or larger to minimize the impact to the PLL jitter performance. Fractional N jitter numbers are found in the sysCLOCK PLL Timing section of the MachXO4 Family Data Sheet (FPGA-DS-02125).

16. Low Power Features

The MachXO4 PLL contains features that enable you to minimize the power consumption of a design. These features include dynamic clock enable and support for the standby mode.

16.1. Dynamic Clock Enable

The dynamic clock enable feature allows you to turn off selected output clocks during periods when the clocks are not used in the design. To support this feature, each output clock has an independent output enable signal that can be selected. The output enable signals are ENCLKOP, ENCLKOS, ENCLKOS2, and ENCLKOS3. When the Clock Enable Ports option is selected in the IP Catalog interface, the output enable signal is shown in the top-level ports of the PLL module for the CLKOP port and any other ports that are enabled in the IP Catalog interface.

If an output is not enabled in the IP Catalog interface, the ports for that selected output signal are not present in the module and that output is inactive.

16.2. Standby Mode

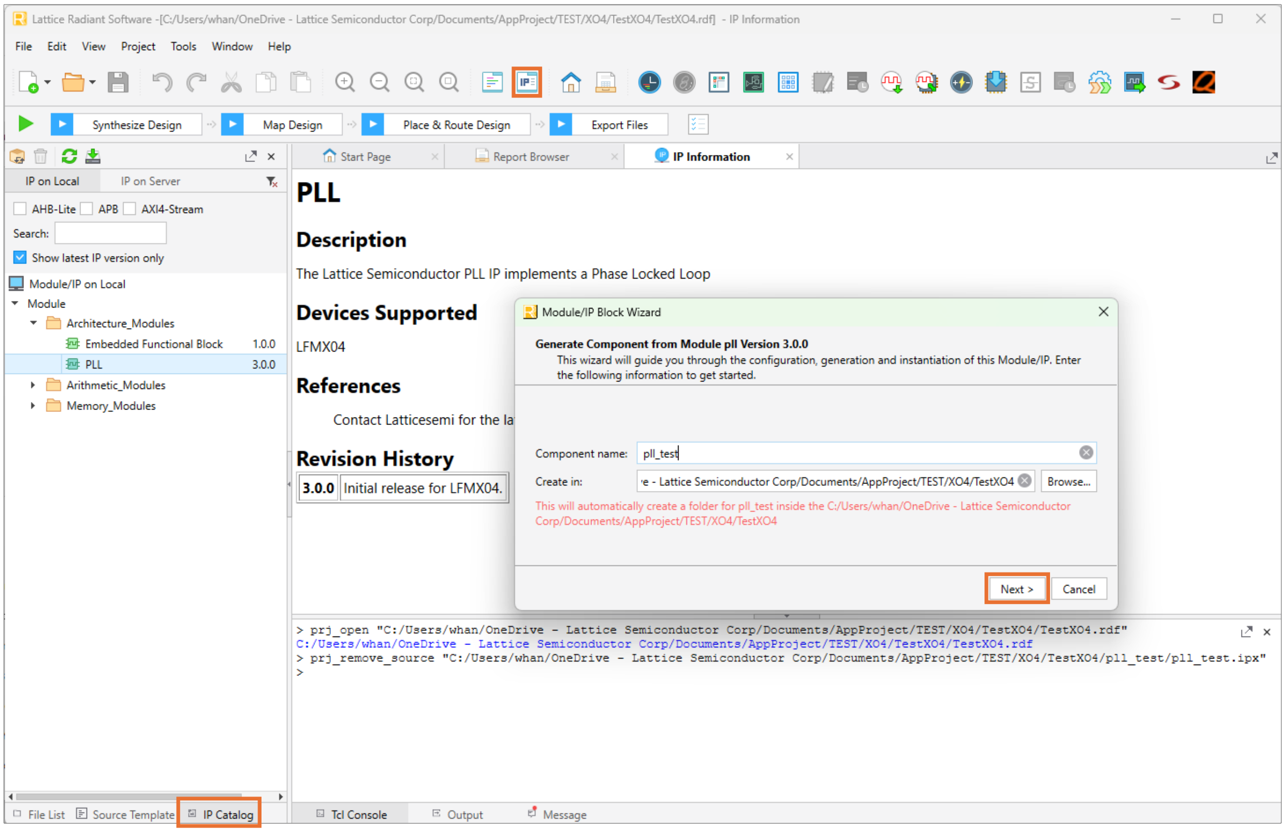
To minimize power consumption, the PLL can be shut down when PLL is not required by the application. The PLL can be restarted when PLL is needed again and, after a short delay to allow the PLL to lock to the feedback signal, the output clocks are reactivated. To support this mode, the Standby Ports option is selected in the IP Catalog interface. This causes the STDBY signal to be shown in the top level of the PLL module. Placing the PLL into the standby mode powers down the PLL and causes all the outputs to be disabled.

The PLL enters the standby mode when the STDBY signal is driven high and the outputs are driven low. The STDBY port can be connected to the power controller so that the PLL enters the low power state when device is driven to the standby mode. Alternatively, the STDBY port can be driven by user logic independent of the standby mode.

The PLL wakes up from the standby mode when the STDBY signal is driven low. When waking up from standby mode, the PLL automatically locks to the external feedback signal that is originally selected prior to entering standby mode. The PLL locks to the external feedback signal after a maximum time delay of t_{LOCK}. When the PLL achieves lock to the external feedback signal, the LOCK signal is asserted high to indicate that it is locked.

17. Configuring the PLL Using IP Catalog

IP Catalog is used to create and configure a PLL. You can select the parameters for the PLL using the graphical user interface. This process results in an HDL model that is used in the simulation and synthesis flow. The figure below shows the main window when the PLL is selected in the IP Catalog in the Lattice Radiant® software. After entering the **Component name**, click **Next** to open the Module/IP Block Wizard window as shown in the following subsections.



PLL

Description

The Lattice Semiconductor PLL IP implements a Phase Locked Loop

Devices Supported

LFMX04

References

Contact Latticesemi for the la

Revision History

3.0.0	Initial release for LFMX04.

Module/IP Block Wizard

Generate Component from Module pll Version 3.0.0

This wizard will guide you through the configuration, generation and instantiation of this Module/IP. Enter the following information to get started.

Component name: pll_test

Create in: 'e - Lattice Semiconductor Corp/Documents/AppProject/TEST/XO4/TestXO4 Browse...

This will automatically create a folder for pll_test inside the C:/Users/whan/OneDrive - Lattice Semiconductor Corp/Documents/AppProject/TEST/XO4/TestXO4

Next > Cancel

```
> prj_open "C:/Users/whan/OneDrive - Lattice Semiconductor Corp/Documents/AppProject/TEST/XO4/TestXO4/TestXO4.rdf"
C:/Users/whan/OneDrive - Lattice Semiconductor Corp/Documents/AppProject/TEST/XO4/TestXO4/TestXO4.rdf
> prj_remove_source "C:/Users/whan/OneDrive - Lattice Semiconductor Corp/Documents/AppProject/TEST/XO4/TestXO4/pll_test/pll_test.ipx"
>
```

Figure 17.1. IP Catalog Main Window for PLL Module

17.1. General Tab

The **General** tab lists all user-accessible attributes with default values set. After complete entering the settings, click **Generate** to generate the source.

17.2. Configuration Modes

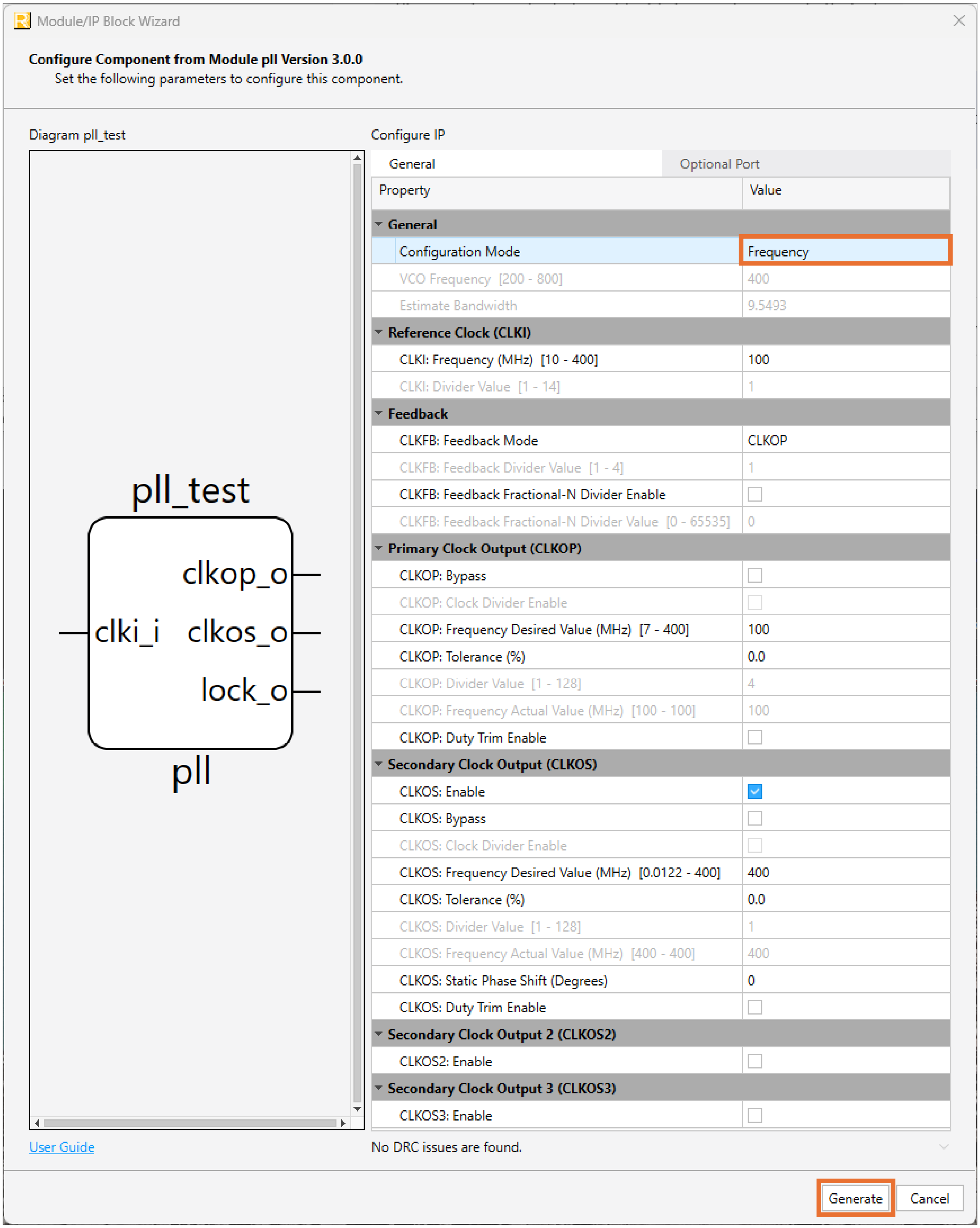
There are two configuration modes that can be used to configure the PLL in the **General** tab: **Frequency Mode** and **Divider Mode**.

17.2.1. Frequency Mode

In this mode, you enter the input and output clock frequencies and the IP Catalog calculates the divider settings. After input and output frequencies are entered, the divider values and actual frequencies are automatically calculated.

If the output frequency entered is not achievable, the nearest frequency is displayed in **Actual Frequency** and an error message is displayed. You can also enter a tolerance value in percent.

If an entered value is out of range, the value is displayed in red and an error message is displayed.



Configure Component from Module pll Version 3.0.0

Set the following parameters to configure this component.

Property	Value
General	
Configuration Mode	Frequency
VCO Frequency [200 - 800]	400
Estimate Bandwidth	9.5493
Reference Clock (CLKI)	
CLKI: Frequency (MHz) [10 - 400]	100
CLKI: Divider Value [1 - 14]	1
Feedback	
CLKFB: Feedback Mode	CLKOP
CLKFB: Feedback Divider Value [1 - 4]	1
CLKFB: Feedback Fractional-N Divider Enable	☐
CLKFB: Feedback Fractional-N Divider Value [0 - 65535]	0
Primary Clock Output (CLKOP)	
CLKOP: Bypass	☐
CLKOP: Clock Divider Enable	☐
CLKOP: Frequency Desired Value (MHz) [7 - 400]	100
CLKOP: Tolerance (%)	0.0
CLKOP: Divider Value [1 - 128]	4
CLKOP: Frequency Actual Value (MHz) [100 - 100]	100
CLKOP: Duty Trim Enable	☐
Secondary Clock Output (CLKOS)	
CLKOS: Enable	☑
CLKOS: Bypass	☐
CLKOS: Clock Divider Enable	☐
CLKOS: Frequency Desired Value (MHz) [0.0122 - 400]	400
CLKOS: Tolerance (%)	0.0
CLKOS: Divider Value [1 - 128]	1
CLKOS: Frequency Actual Value (MHz) [400 - 400]	400
CLKOS: Static Phase Shift (Degrees)	0
CLKOS: Duty Trim Enable	☐
Secondary Clock Output 2 (CLKOS2)	
CLKOS2: Enable	☐
Secondary Clock Output 3 (CLKOS3)	
CLKOS3: Enable	☐

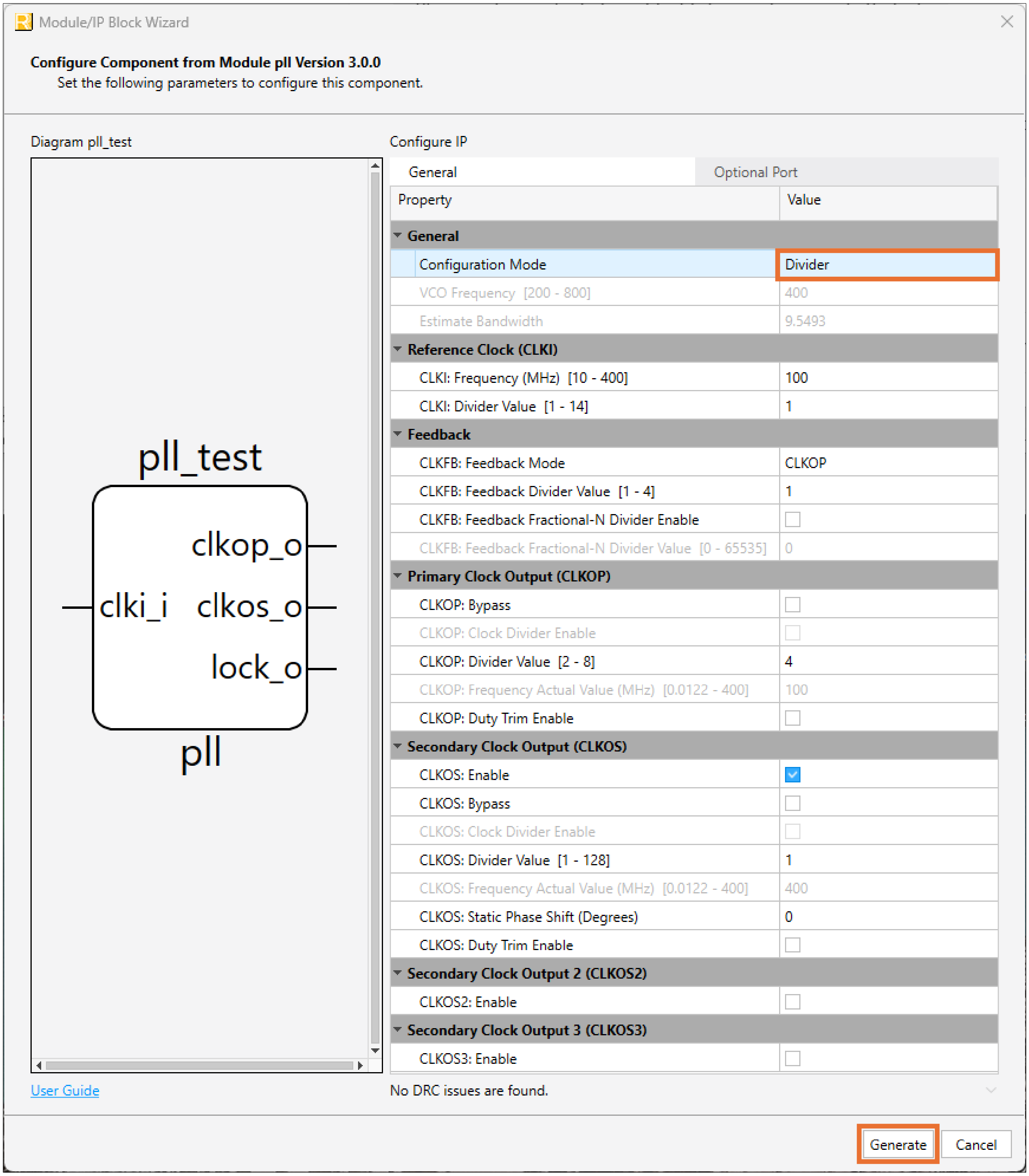
No DRC issues are found.

Figure 17.2. MachXO4 PLL General Tab in Frequency Mode

17.2.2. Divider Mode

In this mode, you set the input frequency and the divider settings. Choose the CLKOP divider value to maximize the frequency of the VCO within the acceptable range as specified in the MachXO4 Family Data Sheet (FPGA-DS-02125).

If the combination of entered values results in an invalid PLL configuration, you are prompted by a text box to change the value with a suggestion for the value that is out of range.



Configure Component from Module pll Version 3.0.0
Set the following parameters to configure this component.

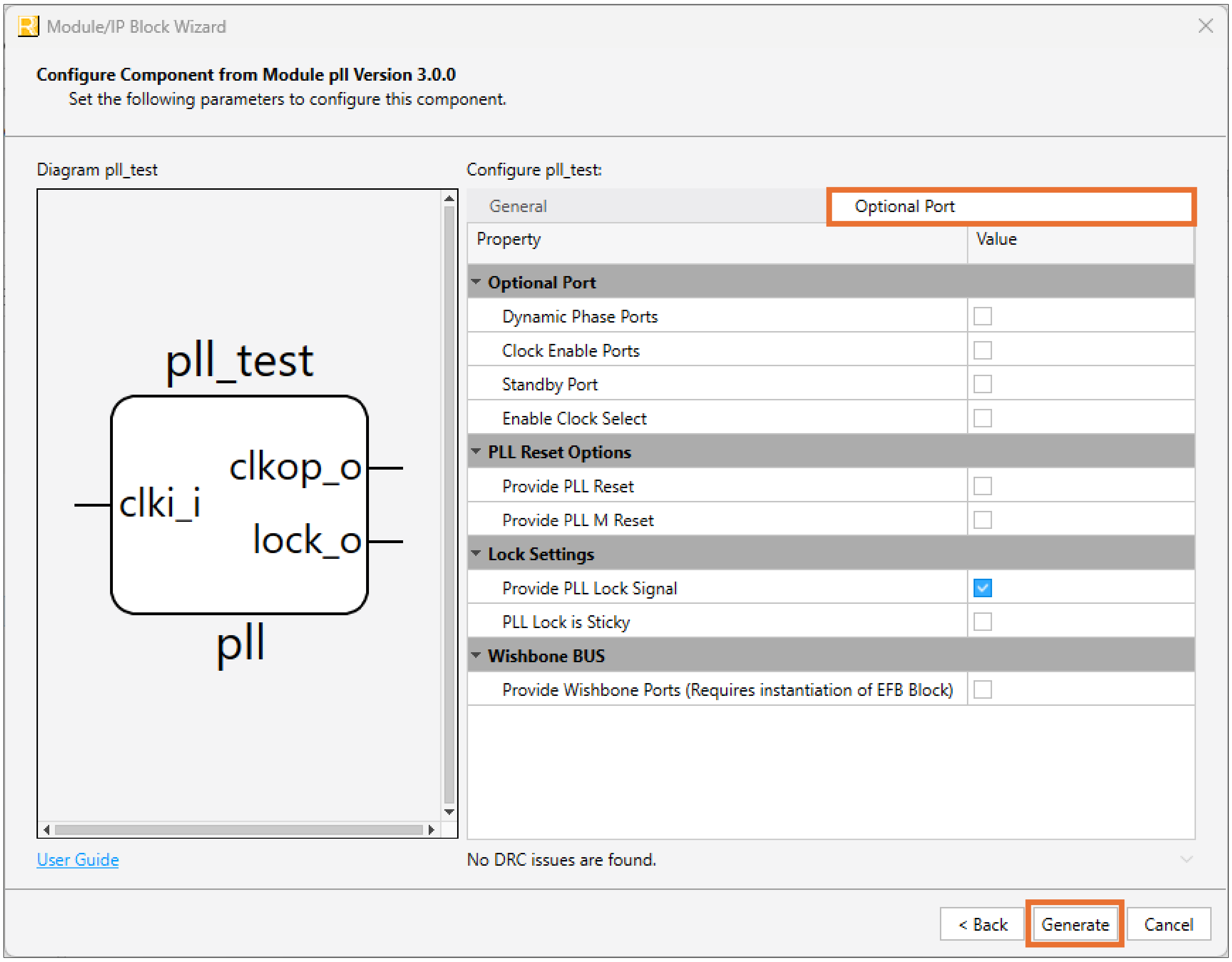
Property	Value
General	
Configuration Mode	Divider
VCO Frequency [200 - 800]	400
Estimate Bandwidth	9.5493
Reference Clock (CLKI)	
CLKI: Frequency (MHz) [10 - 400]	100
CLKI: Divider Value [1 - 14]	1
Feedback	
CLKFB: Feedback Mode	CLKOP
CLKFB: Feedback Divider Value [1 - 4]	1
CLKFB: Feedback Fractional-N Divider Enable	☐
CLKFB: Feedback Fractional-N Divider Value [0 - 65535]	0
Primary Clock Output (CLKOP)	
CLKOP: Bypass	☐
CLKOP: Clock Divider Enable	☐
CLKOP: Divider Value [2 - 8]	4
CLKOP: Frequency Actual Value (MHz) [0.0122 - 400]	100
CLKOP: Duty Trim Enable	☐
Secondary Clock Output (CLKOS)	
CLKOS: Enable	☑
CLKOS: Bypass	☐
CLKOS: Clock Divider Enable	☐
CLKOS: Divider Value [1 - 128]	1
CLKOS: Frequency Actual Value (MHz) [0.0122 - 400]	400
CLKOS: Static Phase Shift (Degrees)	0
CLKOS: Duty Trim Enable	☐
Secondary Clock Output 2 (CLKOS2)	
CLKOS2: Enable	☐
Secondary Clock Output 3 (CLKOS3)	
CLKOS3: Enable	☐

No DRC issues are found.

Figure 17.3. MachXO4 PLL General Tab in Divider Mode

17.3. Optional Port Tab

The **Optional Port** tab lists user-accessible attributes to enable optional ports for auxiliary PLL functionalities.



Configure Component from Module pll Version 3.0.0

Set the following parameters to configure this component.

Diagram pll_test

Configure pll_test:

Property	Value
Optional Port	
Dynamic Phase Ports	□
Clock Enable Ports	□
Standby Port	□
Enable Clock Select	□
PLL Reset Options	
Provide PLL Reset	□
Provide PLL M Reset	□
Lock Settings	
Provide PLL Lock Signal	☑
PLL Lock is Sticky	□
Wishbone BUS	
Provide Wishbone Ports (Requires instantiation of EFB Block)	□

No DRC issues are found.

Figure 17.4. MachXO4 PLL Optional Port Tab

17.4. User-Accessible Attributes and Parameters for the PLL IP

The user-accessible attributes and parameters for the PLL IP are listed in the table below.

Table 17.1. User Parameters in the IP Catalog Interface

User Parameter	Description	Range	Default
Frequency Mode	Enters the desired CLKI and CLKOP frequency	ON/OFF	ON
Divider Mode	Enters the desired CLKI frequency and divider settings	ON/OFF	OFF
CLKI	Frequency	7 to 400 MHz	100 MHz
	Divider	1 to 40	1
CLKFB	Feedback mode	CLKOP, CLKOS, CLKOS2, CLKOS3, INT_OP, INT_OS, INT_OS2, INT_OS3, UserClock	CLKOP
	Fractional-N divider enable	ON/OFF	OFF
	Fractional-N divider	0 to 65535	0
Output Port Selections	Dynamic phase ports	ON/OFF	OFF
	Clock enable ports	ON/OFF	OFF
	Standby ports	ON/OFF	OFF
PLL Reset Options	Provide PLL reset	ON/OFF	OFF
	Provide PLLM reset	ON/OFF	OFF
	Provide CLKOS2 reset	ON/OFF	OFF
	Provide CLKOS3 reset	ON/OFF	OFF
Lock Settings	Provide PLL LOCK signal	ON/OFF	OFF
	PLL LOCK is Sticky	ON/OFF	OFF
WISHBONE Bus	Provide WISHBONE ports	ON/OFF	OFF
CLKOP	Bypass	ON/OFF	OFF
	Clock divider (in Bypass mode only)	ON/OFF	OFF
	Desired frequency	3.125 to 400 MHz	100 MHz
	Tolerance (%)	0.0, 0.1, 0.2, 0.5, 1.0, 2.0, 5.0, 10.0	0.0
	Divider	1–128	8
	Actual frequency (read only)	—	—
	Static phase shift (degrees)	0°, 45°, 90°, 135°, 180°, 225°, 270°, 315°	0°
	Duty trim options mode	Rising/Falling	Rising
	Duty trim options delay multiplier	0, 1, 2, 4	0
CLKOS	Enable	ON/OFF	OFF
	Bypass	ON/OFF	OFF
	Clock divider (in Bypass mode only)	ON/OFF	OFF
	Desired frequency	0.024 – 400 MHz	100 MHz
	Tolerance (%)	0.0, 0.1, 0.2, 0.5, 1.0, 2.0, 5.0, 10.0	0.0
	Divider	1–128	8
	Actual frequency (read only)	—	—
	Static phase shift (degrees)	0°, 45°, 90°, 135°, 180°, 225°, 270°, 315°	0°
	Duty trim options mode	Rising/Falling	Rising
	Duty trim options delay multiplier	0, 1, 2, 4	0
CLKOS2	Enable	ON/OFF	OFF
	Bypass	ON/OFF	OFF
	Clock divider (in Bypass mode only)	ON/OFF	OFF
	Desired frequency	0.024 to 400 MHz	100 MHz
	Tolerance (%)	0.0, 0.1, 0.2, 0.5, 1.0, 2.0, 5.0, 10.0	0.0
	Divider	1–128	8
	Actual frequency (read only)	—	—
	Static phase shift (degrees)	0°, 45°, 90°, 135°, 180°, 225°, 270°, 315°	0°

User Parameter	Description	Range	Default
CLKOS3	Enable	ON/OFF	OFF
	Bypass	ON/OFF	OFF
	Clock divider (in Bypass mode only)	ON/OFF	OFF
	Desired frequency	0.024 – 400 MHz	100 MHz
	Tolerance (%)	0.0, 0.1, 0.2, 0.5, 1.0, 2.0, 5.0, 10.0	0.0
	Divider	1–128	8
	Actual frequency (read only)	—	—
	Static phase shift (degrees)	0°, 45°, 90°, 135°, 180°, 225°, 270°, 315°	0°

18. PLL Reference Clock Switch (PLLREFCS)

The MachXO4 PLL reference clock can be optionally switched between two different clock sources if desired.

To use this feature, the PLLREFCS primitive must be instantiated in the design. The PLLREFCS can only be used with the PLL.

When the reference clock is switched, the PLL may lose lock for a period of time up to t_{LOCK}, as specified in the MachXO4 Family Data Sheet (FPGA-DS-02125), to re-acquire lock. It is recommended that the PLL be reset when switching between reference clock signals, which are at different frequencies.

The PLLREFCS primitive can be instantiated in the source code of a design as defined in this section.

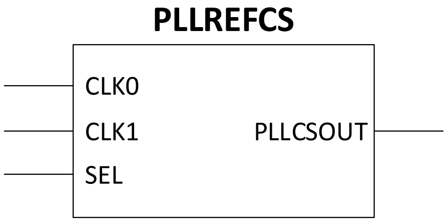


Figure 18.1. PLLREFCS Primitive Symbol

Table 18.1. PLLREFCS Primitive Port Definition

Port Name	I/O	Description
CLK0	NO	CLK0
CLK1	NO	CLK1
SEL	NO	SEL • SEL = 0 – CLK0 input is selected • SEL = 1 – CLK1 input is selected
PLLCSOUT	NO	PLLCSOUT

18.1. PLLREFCS Declaration in VHDL Source Code

Library Instantiation

```
library lfmxo4;
use lfmxo4.all;
```

Component Declaration

```
component PLLREFCS
port(CLK0: in std_logic;
       CLK1: in std_logic;
       SEL:in std_logic;
       PLLCSOUT: out std_logic);
end component;
```

PLLREFCS Instantiation

```
I1: PLLREFCS
port map(CLK0=> CLK0,
       CLK1=> CLK1,
       SEL => SEL,
       PLLCSOUT=>PLLCSOUT);
end component;
```

18.2. PLLREFCS Usage with Verilog Source Code

Component Declaration

```
module PLLREFCS (CLK0, CLK1, SEL, PLLCSOUT);
input CLK0;
input CLK1;
input SEL;
output PLLCSOUT;
endmodule
```

PLLREFCS Instantiation

```
PLLREFCS I1(.CLK0 (CLK0),
        .CLK1 (CLK1),
        .SEL (SEL),
        .PLLCSOUT (PLLCSOUT));
```

19. Internal Oscillator

MachXO4 devices have an internal oscillator that can be used as a clock source in a design. The internal oscillator accuracy is ± 5% (nominal). This oscillator can be used as a clock source for applications that do not require a higher degree of accuracy in the clock.

The internal oscillator remains active to the user logic during transparent configuration. The clock provided by the internal oscillator to the fabric does not stop nor is it influenced while the oscillator is also used internally for background configuration. Although only one internal oscillator is within the device, the user and configuration clocks are sourced from independent clock dividers and resources.

As one general clock source, the oscillator output *OSC* is routed through a divider to provide a flexible clock frequency source. The available output frequencies are shown in Table 19.3.

19.1. Internal Oscillator Primitive Definition

The internal oscillator primitive can be instantiated in the source code of a design as defined in this section.

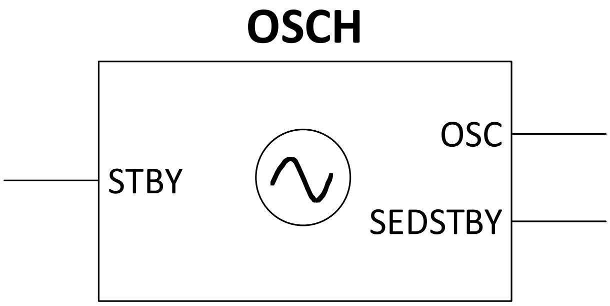


Figure 19.1. OSCH Primitive Symbol

Table 19.1. Internal Oscillator Primitive Port Definition

Port Name	I/O	Description
STDBY	I	Standby – powers down the oscillator in standby mode • STDBY = 0 OSC output is active • STDBY = 1 OSC output is OFF
OSC	O	Clock output port
SEDSTDBY	O	Standby – powers down SED clock[1]

Note:

1. This output is used to notify the SED block that the oscillator shuts down when the device goes into standby. Only required for simulation purposes.

Table 19.2. Internal Oscillator Primitive Attribute Definition

Name	Description	Value	Default
Nominal Frequency (MHz)	NOM_FREQ	2.08, 2.15, 2.22, … 66.5, 88.67, 133.0 (See Table 19.3 for a complete listing)	2.08 MHz

Table 19.3. Internal Oscillator Supported Frequency Settings

2.08	4.16	8.31	16.63
2.15	4.29	8.58	17.73
2.22	4.43	8.87	19.00
2.29	4.59	9.17	20.46
2.38	4.75	9.50	22.17
2.46	4.93	9.85	24.18

2.56	5.12	10.23	26.60
2.66	5.32	10.64	29.56
2.77	5.54	11.08	33.25
2.89	5.78	11.57	38.00
3.02	6.05	12.09	44.33
3.17	6.33	12.67	53.20
3.33	6.65	13.30	66.50
3.50	7.00	14.00	88.67
3.69	7.39	14.78	133.00
3.91	7.82	15.65	

The NOM_FREQ attribute setting must match the value in the table. If the setting does not match, the software issues a warning message and ignores the attribute value.

The STDBY port can be used to power down the oscillator when the oscillator is not used. This port can be connected to a user signal or an I/O pin. You must ensure that the oscillator is not turned off when the oscillator is needed for operations such as WISHBONE bus operations, SPI or I2C configuration, SPI or I2C user mode operations, background flash/NVCM updates, or SED.

19.2. OSCH Declaration in VHDL Source Code

Library Instantiation

```
library lfmXO4;
use lfmXO4.all;
```

Component and Attribute Declaration

```
COMPONENT OSCH
-- synthesis translate_off
       GENERIC (NOM_FREQ: string := "2.56");
-- synthesis translate_on
       PORT (STDBY:INstd_logic;
               OSC:OUTstd_logic;
               SEDSTDBY:OUTstd_logic);
END COMPONENT;

        attribute NOM_FREQ : string;
    attribute NOM_FREQ of OSCinst0 : label is "2.56";
```

OSCH Instantiation

```
begin
OSCInst0: OSCH
-- synthesis translate_off
GENERIC MAP( NOM_FREQ => "2.56" )
-- synthesis translate_on
PORT MAP (STDBY=> stdby,
OSC => osc_int,
SEDSTDBY => stdby_sed
);
```

19.3. OSCH Instantiation in Verilog Source Code

```
// Internal Oscillator
// defparam OSCH_inst.NOM_FREQ = "2.08";// This is the default frequency defparam OSCH_inst.NOM_FREQ =
"24.18";

OSCH OSCH_inst( .STDBY(1'b0), // 0=Enabled, 1=Disabled
// also Disabled with Bandgap=OFF
.OSC(osc_clk),
.SEDSTDBY()); // this signal is not required if not
// using SED
```

Appendix A. Primary Clock Sources and Distribution

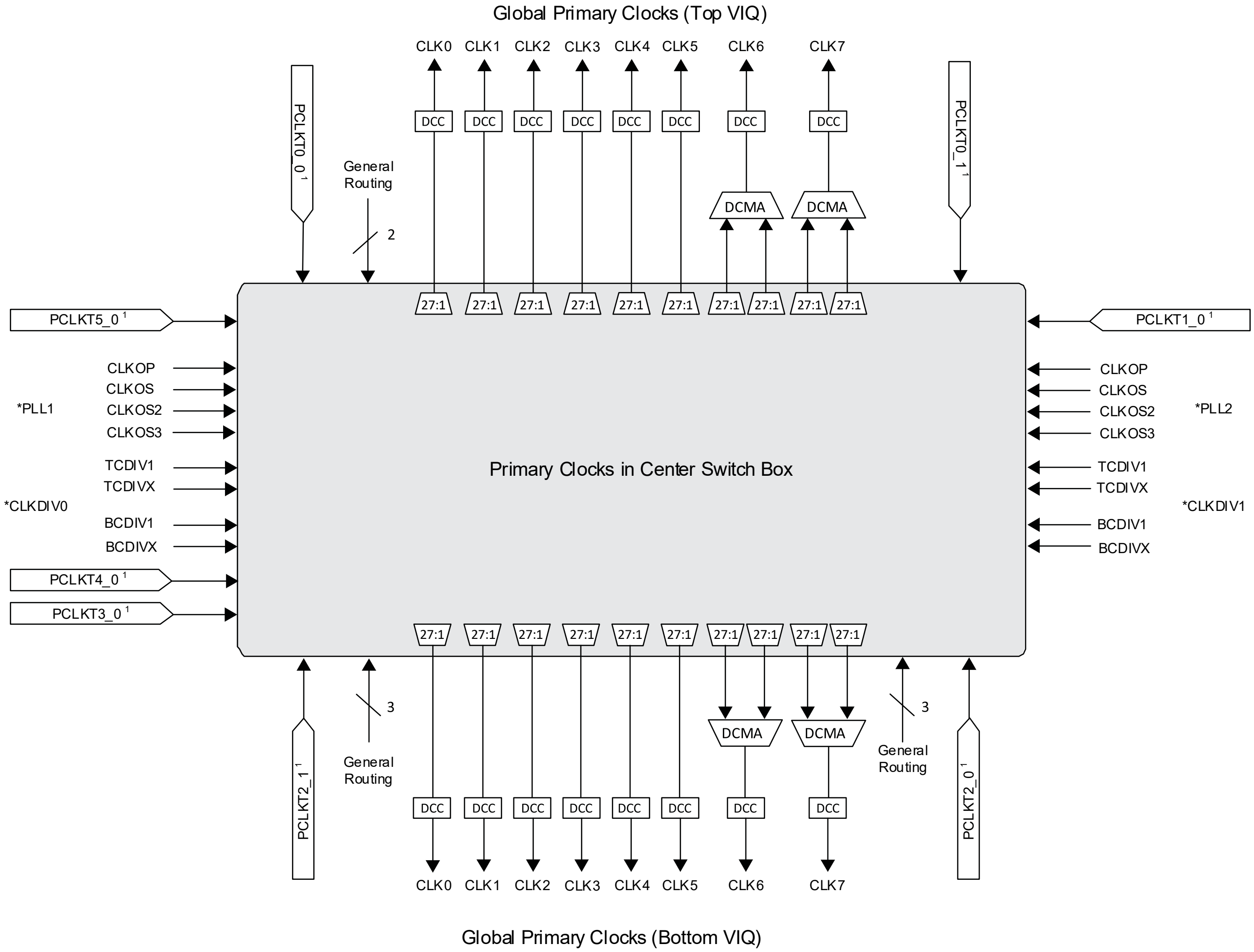


Figure A.1. MachXO4 Primary Clock Sources and Distribution

Note:

1. For differential clock inputs, use only the corresponding PCLKCx_x pin for each PCLKTx_x. The software tools detect and assign the pin automatically for differential IO_TYPEs. Note that PCLKCx_x pins cannot drive the primary clock trees directly and must not be used as independent, single ended clock inputs.

MachXO4 devices have eight global primary clocks. Each primary clock is driven out of the top and bottom of the Primary Clock Center Switch Box. The top and bottom drivers must use the same clock source for each primary clock.

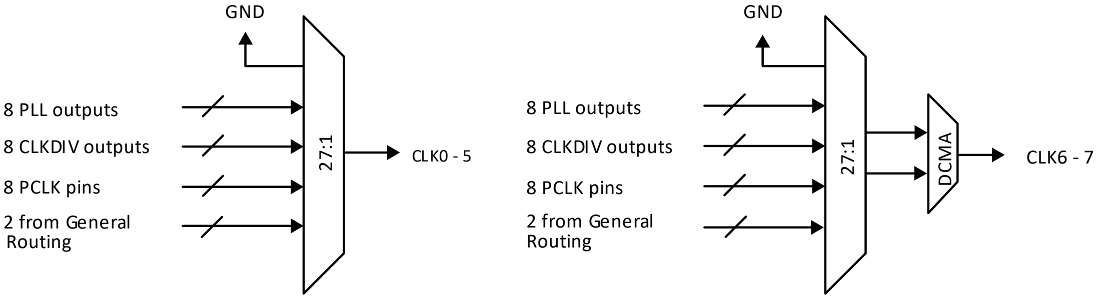


Figure A.2. MachXO4 Primary Clock Multiplexers

Appendix B. Edge Clock Sources and Connectivity

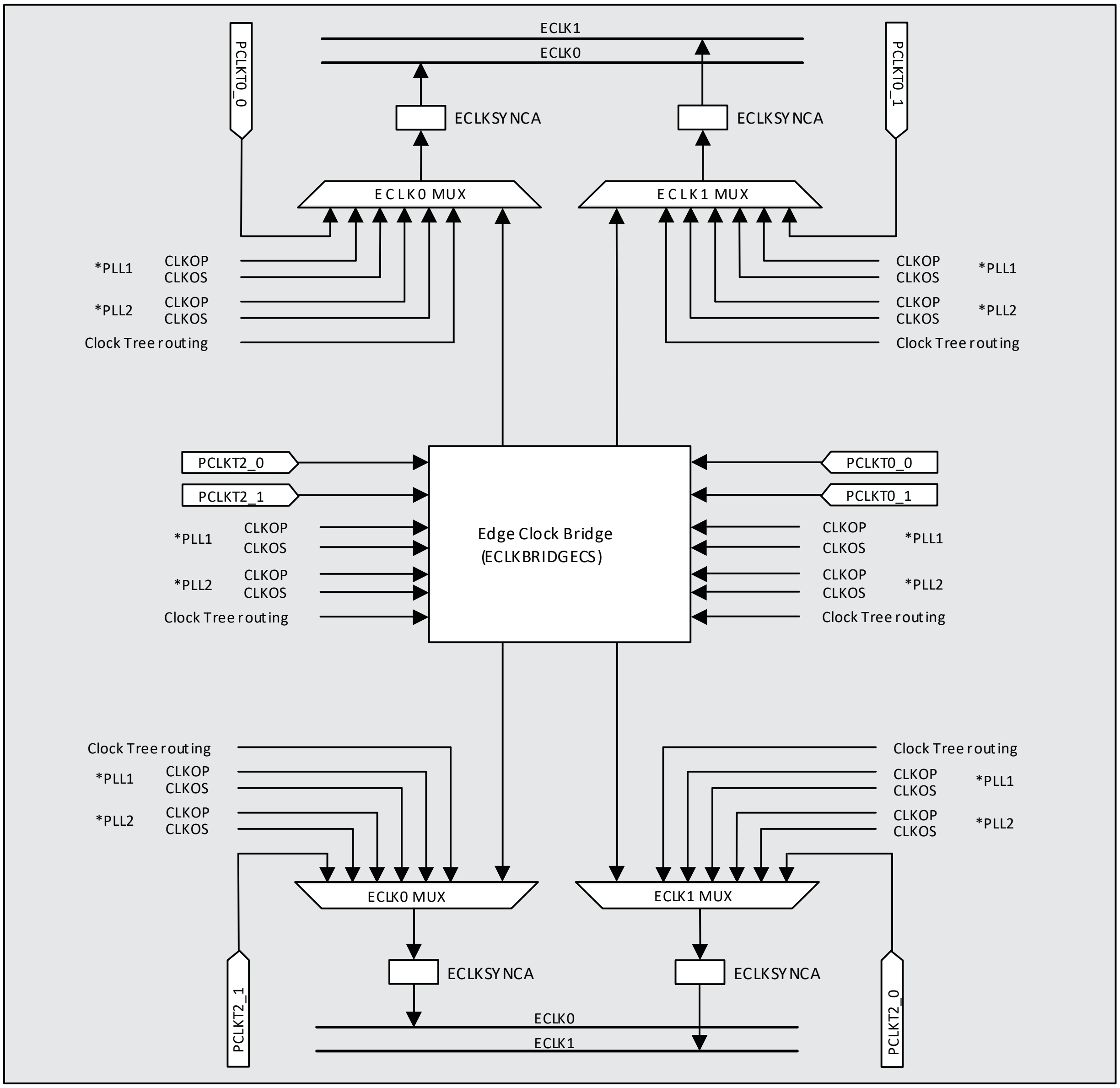


Figure B.1. MachXO4 Edge Clock Sources and Connectivity

Note: The edge clock multiplexers, ECLK0 multiplexer and ECLK1 multiplexer, are routing resources available to the software. There is no dynamic switching between inputs on these multiplexers. To dynamically switch between edge clock drivers, the ECLKBRIDGECS element must be instantiated in the design.

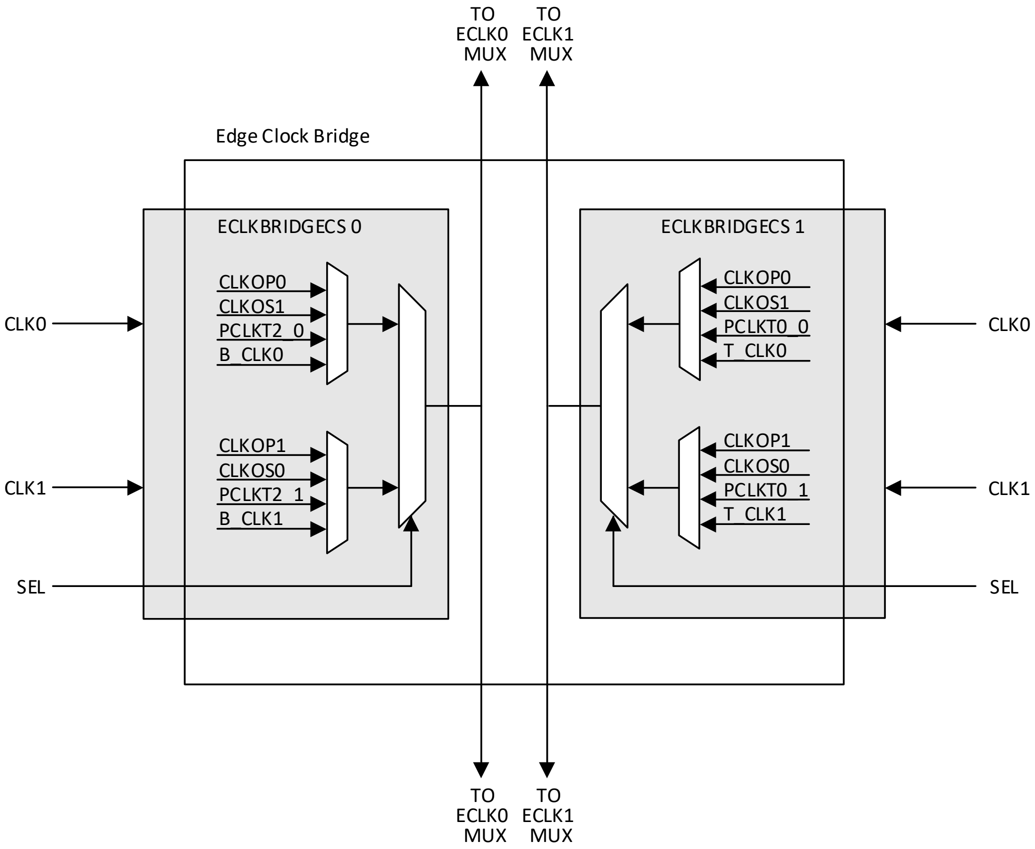


Figure B.2. MachXO4 Edge Clock Bridge Sources and Connectivity

Notes:

1. The edge clock bridge allows a single clock signal to drive both the top and bottom edge clocks with minimal skew. The bridge can also be used where switching between the clock sources is desired.
2. To use the edge clock bridge, the ECLKBRIDGECS primitive must be instantiated in the design. There are two ECLKBRIDGECS resources available in devices that have an edge clock bridge.

Appendix C. Clock Preferences

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

FREQUENCY

The following physical preference 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 preference assigns a maximum skew of 5 ns to a net named NetB:

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

MULTICYCLE

The following preference relaxes 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 preference assigns a clock period of 30 ns to the port named Clk1:

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

PROHIBIT

The following preference prohibits the use of a primary clock to route a clock net named bf_clk:

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

The following preference prohibits the use of a secondary high fan-out net to route a clock net named bf_clk:

```
        PROHIBIT SECONDARY NET "bf_clk";
```

PROHIBIT_BOTH

When this setting is selected it causes Design Planner to generate both the PROHIBIT PRIMARY NET net_name and PROHIBIT SECONDARY NET net_name.

USE PRIMARY

Use a primary clock resource to route the specified net:

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

USE SECONDARY

Use a secondary high fan-out net resource to route the specified net:

```
        USE SECONDARY NET "clk_lessfast" QUADRANT_TL;
```

USE EDGE

Use an edge clock resource to route the specified net. The net must be eligible for routing using the edge clock resources.

```
        USE EDGE NET "clk_fast";
```

EDGE2EDGE

Use the ECLK bridge resource to route the specified net. The net must be eligible for routing using the edge clock resources.

```
        USE EDGE2EDGE NET "clk_fast";
```

CLOCK_TO_OUT

This preference specifies a maximum allowable output delay relative to a clock.

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

The CLKNET stops tracing the path before the PLL, hence you do not get PLL compensation timing numbers.

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

The above preference yields the following clock path:

```
        Clock path pll_inst/pll_utp_0_0 to PFU_33:
NameFanoutDelay (ns)Site Resource
ROUTE 49 2.892ULPPLL.MCLK toR3C14.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 yields the following clock path:

```
        Clock path RxClk to PFU_33:
NameFanoutDelay (ns)Site Resource
IN_DEL--- 1.431 D5.PAD toD5.INCK RxClk
ROUTE 1 0.843 D5.INCK toULPPLL.CLKIN RxClk_c
MCLK_DEL--- 3.605 ULPPLL.CLKIN to ULPPLL.MCLK
                pll_inst/pll_utp_0_0
ROUTE492.892 ULPPLL.MCLK toR3C14.CLK0 pll_rxclk
        --------
        8.771 (57.4% logic, 42.6% route), 2 logic levels.
```

INPUT_SETUP

This preference specifies a 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 you need a trace calculation based on the previous clock edge.

This preference is useful when setting the PLL output phase adjustment. As 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, you can set the phase to 270° and set the INPUT_SETUP preference with PLL_PHASE_BACK.

Appendix D. PLL WISHBONE Bus Operation

The MachXO4 PLL operating parameters can be changed dynamically via the Embedded Function Block (EFB) WISHBONE bus. You must instantiate the EFB block in your design to use this feature. The user logic WISHBONE bus is then connected to the EFB block. A hard-wired PLL data bus is used to communicate between the EFB and the PLL. See MachXO4 Hardened Control Functions User Guide (FPGA-TN-02403) for more information about the using of the EFB block in a design.

The PLL data bus on the PLL module provides support for functional simulation of this operation. You must connect the PLL data bus to the EFB in your HDL design for simulation to work properly. The PLL data bus ports and the corresponding EFB PLL bus port connections are listed in the table below.

Table D.1. PLL Data Bus Port Definitions

PLL Port Name	I/O	Description	EFB Port Name
PLLCLK	I	PLL data bus clock input signal	pll_bus_o[16]
PLLRST	I	PLL data bus reset. This resets only the data bus, not any register values.	pll_bus_o[15]
PLLSTB	I	PLL data bus strobe signal.	pll_bus_o[14]
PLLWE	I	PLL data bus write enable signal	pll_bus_o[13]
PLLADDR [4:0]	I	PLL data bus address	pll_bus_o[12:8]
PLLDATI [7:0]	I	PLL data bus data input	pll_bus_o[7:0]
PLLDATO [7:0]	O	PLL data bus data output	pll_bus_i[8:1]
PLLACK	O	PLL data bus acknowledge signal	pll_bus_i[0]

D.1. PLL Architecture

The MachXO4 PLL has four output sections with flexible configuration settings to support a variety of different applications. IP Catalog is able to support most of the common PLL configurations, but for more complex needs, the WISHBONE bus can be used to change the PLL configuration, which allows for more advanced support options.

Each of the four PLL output sections have similar configuration options. Each output section is assigned a letter designator; A for the CLKOP output, B for the CLKOS output, C for the CLKOS2 output, and D for the CLKOS3 output section. Within each of the four output sections, there are three signal selection multiplexers, which are used to control the PLL configuration. A diagram of the A output section is shown in Figure D.1. The B output section is the same as the A section except the multiplexers are labeled B0, B1, and B2. The C and D sections are similar with multiplexers labeled C0, C1, C2, D0, D1, and D2. The C and D sections have the phase adjust block without the edge trim feature.

Note: A1 multiplexer does not exist in the CLKOP path, but is shown to illustrate the corresponding B1, C1, and D1 multiplexers location. See Figure 11.1.

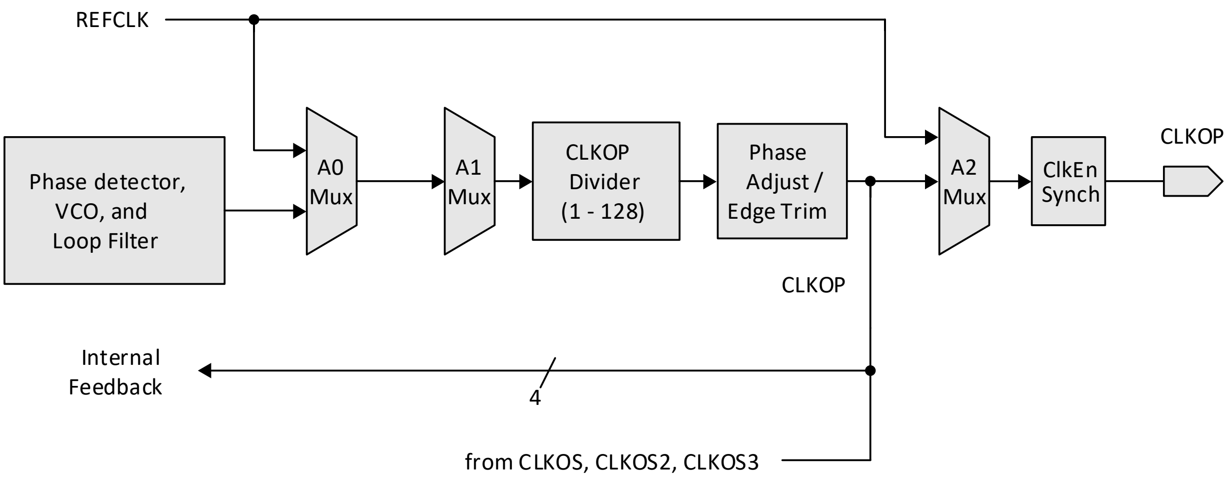


Figure D.1. PLL CLKOP Output Section

The EFB WISHBONE register map for the PLL registers is shown in Table D.2. Add 0x20 for the corresponding locations to access an optional second MachXO4 PLL.

Table D.2. EFB WISHBONE Locations for PLL Registers

Reg.	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	MC1_DIVFBK_FRAC[7:0]							
1	MC1_DIVFBK_FRAC[15:8]							
2	MC1_LOADREG	MC1_DELA[6:0]						
3	MC1_PLLPDN	MC1_DELB[6:0]						
4	MC1_WBRESET	MC1_DELC[6:0]						
5	MC1_USE_DESI	MC1_DELD[6:0]						
6	MC1_REFIN_RESET	MC1_DIVA[6:0]						
7	MC1_PLLRST_ENA	MC1_DIVB[6:0]						
8	MC1_MRST_ENA	MC1_DIVC[6:0]						
9	MC1_STDBY	MC1_DIVD[6:0]						
A	MC1_ENABLE_SYNC	MC1_PHIB[2:0]			MC1_INT_LOCK_STICKY	MC1_PHIA[2:0]		
B	MC1_DCRST_ENA	MC1_PHID[2:0]			MC1_RESERVED2	MC1_PHIC[2:0]		
C	MC1_DDRST_ENA	MC1_SEL_OUTB[2:0]			MC1_INTFB	MC1_SEL_OUTA[2:0]		
D	MC1_LOCK[1:0]		MC1_SEL_OUTC[2:0]			MC1_SEL_OUTD[2:0]		
E	MC1_SEL_DIVA[1:0]		MC1_SEL_DIVB[1:0]		MC1_SEL_DIVC[1:0]		MC1_SEL_DIVD[1:0]	
F	MC1_CLKOP_TRIM[3:0]				MC1_CLKOS_TRIM[3:0]			
10	MC1_DYN_SOURCE	MC1_LOCK_SEL[2:0]			MC1_ENABLE_CLK[3:0]			
11	MC1_TRIMOS3_BYPASS_N	MC1_TRIMOS2_BYPASS_N	MC1_TRIMOS_BYPASS_N	MC1_TRIMOP_BYPASS_N	MC1_DYN_SEL[1:0]		MC1_DIRECTION	MC1_ROTATE
12	MC1_LF_RESGRND	MC1_SEL_REF1[2:0]			MC1_EN_UP	MC1_SEL_REF2[2:0]		
13	MC1_DIVFBK_ORDER[1:0]		MC1_CLKMUX_FB[1:0]		MC1_SEL_FBK[3:0]			
14	MC1_GMC_RESET	MC1_DIVREF[6:0]						
15	MC1_FORCE_VFILTER	MC1_DIVFBK[6:0]						
16	MC1_LF_PRESET	MC1_LF_RESET	MC1_TEST_ICP	MC1_EN_FILTER_OPAMP	MC1_FLOAT_ICP	MC1_GPROG[2:0]		

Reg.	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
17	MC1_KPROG[2:0]			MC1_IPROG[4:0]				
18	MC1_GMC_PRESET	MC1_RPROG[6:0]						
19	MC1_GMCREF_SEL[1:0]		MC1_MFGOUT2_SEL[2:0]			MC1_MFGOUT1_SEL[2:0]		
1A	MC1_GMCSEL[3:0]				MC1_VCO_BYPASS_D0	MC1_VCO_BYPASS_C0	MC1_VCO_BYPASS_B0	MC1_VCO_BYPASS_A0
1B	MC1_RESERVED[4:0]					MC1_EN_PHI	MC1_DPROG[1:0]	
1C	RESERVED							LOCK_STS

Note: Registers 0 through 11 are user accessible registers. The remaining registers are read-only.

Table D.3. PLL Register Descriptions

Register Name	Register Addr (Hex)	Size (Bits)	Description	Default Value	User Access	User Interface Access
MC1_DIVFBK_FRAC[15:0]	0[7:0] 1[7:0]	16	Fractional-N divider value. Fractional-N divider is equal to this value / 65535.	0	Yes	Yes
MC1_LOADREG	2[7]	1	Only valid if MC1_DYN_PHASE=0. Command to start a divider output phase shift on negative edge of MC1_LOADREG bit. The divider output phase shift for CLKOP occurs if the MC1_DIVA and MC1_DELA values are not the same. A CLKOS divider output phase shift occurs if the MC1_DIVB and MC1_DELB values are not the same. A CLKOS2 divider output phase shift occurs if the MC1_DIVC and MC1_DELC values are not the same. A CLKOS3 divider output phase shift occurs if the MC1_DIVD and MC1_DELD values are not the same.	0	Yes	—
MC1_PLLPDN	3[7]	1	Powers down the PLL when not used. Software automatically sets this to 1 when the PLL is used in a design and to 0 if the PLL is not used. 0 = Powers down PLL. 1 = PLL is powered up.	1	Yes	Yes automatic
MC1_WBRESET	4[7]	1	PLL reset from Wishbone – Equivalent to the RESETM port operation. 0 = PLL normal operation. 1 = PLL reset active.	0	Yes	No
MC1_USE_DESI	5[7]	1	Controls whether the Fractional-N divider is used. 0 = PLL normal operation. 1 = Use Fractional-N divider.	0	Yes	Yes
MC1_REFIN_RESET	6[7]	1	Controls whether the PLL is automatically reset when the input clock reference is switched using the PLLREFCS primitive 0 = Do not reset PLL. 1 = Automatically reset PLL if input switches.	0	Yes	No
MC1_PLLRST_ENA	7[7]	1	Enables the PLLRESET port. 0 = PLLRESET port not active. 1 = PLLRESET port is enabled.	0	Yes	Yes
MC1_MRST_ENA	8[7]	1	Enables the RESETM port. 0 = RESETM port not active. 1 = RESETM port is enabled.	0	Yes	Yes
MC1_STDBY	9[7]	1	Enables the STDBY port on PLL 0 = STDBY port not active. 1 = STDBY port is enabled.	0	Yes	Yes
MC1_ENABLE_SYNC	A[7]	1	Enables synchronous disable or enable of secondary clocks CLKOS, CLKOS2, CLKOS3 with respect to CLKOP. 0 = Synchronous disable or enable not active. 1 = Synchronous disable or enable is active.	0	Yes	No
MC1_DCRST_ENA	B[7]	1	Enables the RESETDC port – CLKOS2 reset. 0 = RESETDC port not active. 1 = RESETDC port is enabled.	0	Yes	Yes

Register Name	Register Addr (Hex)	Size (Bits)	Description	Default Value	User Access	User Interface Access
MC1_DDRST_ENA	C[7]	1	Enables the RESETDD port – CLKOS3 reset. 0 = RESETDD port not active. 1 = RESETDD port is enabled.	0	Yes	Yes
MC1_DELA[6:0]	2[6:0]	7	CLKOP section delay value for coarse phase adjustments. For zero delay, this value equals to the value of MC1_DIVA[6:0].	7	Yes	Yes
MC1_DELB[6:0]	3[6:0]	7	CLKOS section delay value for coarse phase adjustments. For zero delay, this value equals to the value of MC1_DIVB[6:0].	7	Yes	Yes
MC1_DELC[6:0]	4[6:0]	7	CLKOS2 section delay value for coarse phase adjustments. For zero delay, this value equals to the value of MC1_DIVC[6:0].	7	Yes	Yes
MC1_DELD[6:0]	5[6:0]	7	CLKOS3 section delay value for coarse phase adjustments. For zero delay, this value equals to the value of MC1_DIVD[6:0].	7	Yes	Yes
MC1_DIVA[6:0]	6[6:0]	7	CLKOP section output divider setting equals to the Divide value – 1.	7	Yes	Yes
MC1_DIVB[6:0]	7[6:0]	7	CLKOS section output divider setting equals to the Divide value – 1.	7	Yes	Yes
MC1_DIVC[6:0]	8[6:0]	7	CLKOS2 section output divider setting equals to the Divide value – 1.	7	Yes	Yes
MC1_DIVD[6:0]	9[6:0]	7	CLKOS3 section output divider setting equals to the Divide value – 1.	7	Yes	Yes
MC1_PHIA[2:0]	A[2:0]	3	Selects the VCO phase shift (0–7) for CLKOP. Each tap represents 45° shift of the VCO.	0	Yes	Yes
MC1_PHIB[2:0]	A[6:4]	3	Selects the VCO phase shift (0–7) for CLKOS. Each tap represents 45° shift of the VCO.	0	Yes	Yes
MC1_PHIC[2:0]	B[2:0]	3	Selects the VCO phase shift (0–7) for CLKOS2. Each tap represents 45° shift of the VCO.	0	Yes	Yes
MC1_PHID[2:0]	B[6:4]	3	Selects the VCO phase shift (0–7) for CLKOS3. Each tap represents 45° shift of the VCO.	0	Yes	Yes
MC1_INT_LOCK_STICKY	A[3]	1	Sets internal lock to be sticky or not. Sticky lock stays high when lock is achieved until the PLL is reset or powered down. Internal lock is not used in the PLL. 0 = Internal lock normal operation. 1 = Internal lock sticky operation.	1	Yes	Not used
MC1_RESERVED2	B[3]	1	Not used.	—	—	—
MC1_SEL_OUTA[2:0]	C[2:0]	3	Multiplexer A2 selects value for CLKOP output. Can be used to cascade dividers if desired. 000 = DIVA output to CLKOP. 001 = DIVB output to CLKOP. 010 = DIVC output to CLKOP. 011 = DIVD output to CLKOP. 100 = REFCLK output to CLKOP (same as bypass mode without using any clock divider).	000	Yes	No
MC1_SEL_OUTB[2:0]	C[6:4]	3	Multiplexer B2 selects value for CLKOS output. Can be used to cascade dividers if desired. 000 = DIVB output to CLKOS. 001 = DIVC output to CLKOS.	000	Yes	No

Register Name	Register Addr (Hex)	Size (Bits)	Description	Default Value	User Access	User Interface Access
			010 = DIVD output to CLKOS. 011 = DIVA output to CLKOS. 100 = REFCLK output to CLKOS (same as bypass mode without using any clock divider).			
MC1_SEL_OUTC[2:0]	D[5:3]	3	Multiplexer C2 selects value for CLKOS2 output. Can be used to cascade dividers if desired. 000 = DIVC output to CLKOS2. 001 = DIVD output to CLKOS2. 010 = DIVA output to CLKOS2. 011 = DIVB output to CLKOS2. 100 = REFCLK output to CLKOS2 (same as bypass mode without using any clock divider).	000	Yes	No
MC1_SEL_OUTD[2:0]	D[2:0]	3	Multiplexer D2 selects value for CLKOS3 output. Can be used to cascade dividers if desired. 000 = DIVD output to CLKOS3. 001 = DIVA output to CLKOS3. 010 = DIVB output to CLKOS3. 011 = DIVC output to CLKOS3. 100 = REFCLK output to CLKOS3 (same as bypass mode without using any clock divider).	000	Yes	No
MC1_INTFB	C[3]	1	Use the PLL internal feedback for initial PLL lock. Used with INTLOCK and PLLWAKESYNC ports. Not recommended to change this setting. 0 = PLL internal feedback is not used. 1 = Uses PLL internal feedback.	0	Yes	No
MC1_LOCK[1:0]	D[7:6]	2	Frequency lock-detector resolution or sensitivity. 00 = ±250 ppm 01 = ±1000 ppm 10 = ±4000 ppm 11 = ±16000 ppm	00	Yes	No
MC1_SEL_DIVA[1:0]	E[7:6]	2	Multiplexer A1 selects value for input to DIVA (CLKOP). Can be used to cascade dividers if desired. 00 = Multiplexer A0 output. 01 = DIVD (CLKOS3) output. 10 = DIVB (CLKOS) output. 11 = DIVC (CLKOS2) output.	00	Yes	No
MC1_SEL_DIVB[1:0]	E[5:4]	2	Multiplexer B1 selects value for input to DIVB (CLKOS). Can be used to cascade dividers if desired. 00 = Multiplexer B0 output. 01 = DIVA (CLKOP) output. 10 = DIVD (CLKOS3) output. 11 = DIVC (CLKOS2) output.	00	Yes	No
MC1_SEL_DIVC[1:0]	E[3:2]	2	Multiplexer C1 selects value for input to DIVC (CLKOS2). Can be used to cascade dividers if desired. 00 = Multiplexer C0 output.	00	Yes	No

Register Name	Register Addr (Hex)	Size (Bits)	Description	Default Value	User Access	User Interface Access
			01 = DIVA (CLKOP) output. 10 = DIVB (CLKOS) output. 11 = DIVD (CLKOS3) output.			
MC1_SEL_DIVD[1:0]	E[1:0]	2	Multiplexer D1 selects value for input to DIVD (CLKOS3). Can be used to cascade dividers if desired. 00 = Multiplexer D0 output. 01 = DIVA (CLKOP) output. 10 = DIVB (CLKOS) output. 11 = DIVC (CLKOS2) output.	00	Yes	No
MC1_CLKOP_TRIM[3:0]	F[7:4]	4	CLKOP output trimming control. Bit 3 of TRIM[3:0] sets the edge to be affected. TRIM[3] = 0 sets falling edge trim active. TRIM[3] = 1 sets rising edge trim active. TRIM[2:0] is a one hot signal. TRIM[2:0] = 001 sets 70 ps trim. TRIM[2:0] = 010 sets 140 ps trim. TRIM[2:0] = 100 sets 280 ps trim.	0000	Yes	Yes
MC1_CLKOS_TRIM[3:0]	F[3:0]	4	CLKOS output trimming control. Bit 3 of TRIM[3:0] sets the edge to be affected. TRIM[3] = 0 sets falling edge trim active. TRIM[3] = 1 sets rising edge trim active. TRIM[2:0] is a one hot signal. TRIM[2:0] = 001 sets 70 ps trim. TRIM[2:0] = 010 sets 140 ps trim. TRIM[2:0] = 100 sets 280 ps trim.	0000	Yes	Yes
MC1_ENABLE_CLK[3:0]	10[3:0]	4	Clock output enable for each PLL output port. This fuse setting is ORed with the corresponding Enable port signal to set the clock output enable control. Software sets this value automatically based on the settings in the user interface. Not recommended to change this setting. xxx1 = Enables CLKOP. xx1x = Enables CLKOS. x1xx = Enables CLKOS2. 1xxx = Enables CLKOS3.	0001	Yes	Yes
MC1_LOCK_SEL[2:0]	10[6:4]	3	Lock-detector operation mode – normal or sticky. Sticky lock stays high when lock is achieved until the PLL is reset or powered down. 000 = PLL Lock normal operation. 001 = PLL Lock sticky operation. 100 = Alternate PLL Lock normal operation. Other values are not supported modes.	000	Yes	Yes
MC1_DYN_SOURCE	10[7]	1	Specifies whether the Wishbone or external ports control the dynamic phase settings. 0 = Wishbone registers are in control. 1 = External ports are in control.	1	Yes	Indirect
MC1_DIRECTION	11[1]	1	Only valid if MC1_DYN_PHASE=0. Specifies the direction of the dynamic phase change for MC1_ROTATE command.	0	Yes	—

Register Name	Register Addr (Hex)	Size (Bits)	Description	Default Value	User Access	User Interface Access
			0 = Phase rotates to a later phase. 1 = Phase rotates to an earlier phase.			
MC1_ROTATE	11[0]	1	Only valid if MC1_DYN_PHASE=0. Command to start a change from current VCO phase to later or earlier phase. Phase changes on negative edge of the MC1_ROTATE bit. Each step change represents a 45° change of VCO. (MC1_ROTATE is equivalent to the PHASESTEP signal.)	0	Yes	—
MC1_ DYN_SEL[1:0]	11[3:2]	2	Only valid if MC1_DYN_PHASE=0. Specifies which port is being controlled by dynamic phase controls. 00 = Enables CLKOS 01 = Enables CLKOS2 10 = Enables CLKOS3 11 = Enables CLKOP	00	Yes	—
MC1_TRIMOP_BYPASS_N	11[4]	1	Bypasses the CLKOP output trim circuit. This setting selects whether to bypass the trim circuit. 0 = Bypasses the trim circuit. 1 = Does not bypass the trim circuit.	0	Yes	Indirect
MC1_TRIMOS_BYPASS_N	11[5]	1	Bypasses the CLKOS output trim circuit. This setting selects whether to bypass the trim circuit. 0 = Bypasses the trim circuit. 1 = Does not bypass the trim circuit.	0	Yes	Indirect
MC1_TRIMOS2_BYPASS_N	11[6]	1	Bypasses the CLKOS2 output trim bits. There is not a trim control on CLKOS2. There is a dummy trim circuit used to equalize the delays between CLKOP, CLKOS, CLKOS2, and CLKOS3 outputs when trim is active on the CLKOP or CLKOS outputs. This setting selects whether to bypass the trim circuit. 0 = Bypasses the trim circuit. 1 = Does not bypass the trim circuit.	0	Yes	Indirect
MC1_TRIMOS3_BYPASS_N	11[7]	1	Bypasses the CLKOS3 output trim bits. There is not a trim control on CLKOS3. There is a dummy trim circuit used to equalize the delays between CLKOP, CLKOS, CLKOS2, anda CLKOS3 outputs when trim is active on the CLKOP or CLKOS outputs. This setting selects whether to bypass the trim circuit. 0 = Bypasses the trim circuit. 1 = Does not bypass the trim circuit.	0	Yes	Indirect

References

- MachXO4 Implementing High-Speed I/O Interface User Guide (FPGA-TN-02410)
- MachXO4 Hardened Control Functions User Guide (FPGA-TN-02403)
- MachXO4 Family Data Sheet (FPGA-DS-02125)
- Lattice Radiant Timing Constraints Methodology (FPGA-AN-02059)
- MachXO4 web page
- Lattice Radiant Software web page
- Lattice Insights for Lattice Semiconductor training courses and learning plans

Technical Support Assistance

Submit a technical support case through www.latticesemi.com/techsupport.

For frequently asked questions, refer to the Lattice Answer Database at www.latticesemi.com/Support/AnswerDatabase.

Revision History

Revision 1.0, December 2025

Section	Change Summary
Abbreviations in This Document	Updated the abbreviations.
Introduction	Updated Table 1.1. Number of PLLs, Edge Clocks, and Clock Dividers.
Dynamic Clcock Multiplier (DCMA)	Updated the Library Instantiation in the DCMA Declaration in VHDL Source Code section.
Dynamic Clock Control (DCCA)	Updated the Library Instantiation in the DCCA Declaration in VHDL Source Code section.
Edge Clocks	Updated the description on the use of the edge clock bridge in the Edge Clock Bridge section.
ECLKBRIDGECS Primitive Definition	Updated the Library Instantiation in the ECLKBRIDGECS Declaration in VHDL Source Code section.
Edge Clock Synchronization (ECLKSYNCA)	Updated the Library Instantiation in the ECLKSYNCA Declaration in VHDL Source Code section.
Clock Dividers (CLKDIVC)	• Updated the description on ALIGNWD in the CLKDIVC Primitive Definition section. • Updated the Library Instantiation in the CLKDIVC Declaration in VHDL Source Code section.
sysCLOCK PLL	• Updated the description on Output Clock Dividers (CLKOP, CLKOS, CLKOS2, CLKOS3) in the PLL Divider Blocks section. • Updated the LOCK Output section.
Dynamic Phase Adjustment	• Updated Table 13.2. PHASEDIR Signal Settings Definitions. • Updated the GPLL VCO Phase Rotation Timing Diagram.
Configuring the PLL Using IP Catalog	Added this section.
PLL Reference Clock Switch (PLLREFCS)	Updated the Library Instantiation in the PLLREFCS Declaration in VHDL Source Code section.
Internal Oscillator	Made editorial changes and modifications to cover MachXO4 devices.
Appendix A. Primary Clock Sources and Distribution	Updated the Primary Clock Sources and Distribution diagram.
Appendix C. Clock Preferences	• Updated the MULTICYCLE clock preference. • Removed the spreadsheet view feature.
Appendix D. PLL WISHBONE Bus Operation	Added description on the using of the EFB block in a design.
References	Updated references.

Revision 0.80, March 2025

Section	Change Summary
All	Initial preliminary release.

www.latticesemi.com