PLL Module

User Guide

FPGA-IPUG-02063-2.0

October 2024

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

Acronyms in This Document

A list of acronyms used in this document.

Acronym	Definition
AMBA	Advanced Microcontroller Bus Architecture
APB	Advanced Peripheral Bus
FPGA	Field Programmable Gate Array
GPLL	Global Phase-Locked Loop
IP	Intellectual Property
LMMI	Lattice Memory Mapped Interface
PLL	Phase-Locked Loop

1. Introduction

1.1. Overview of the IP

The Phase-Locked Loop (PLL) Module is capable of frequency synthesis and clock phase management including clock injection delay cancellation. The module has the flexibility of input and feedback source selections, multiple output selections, VCO phase rotation, and independent phase shifting features. The multiple output clocks can be synchronized with the option to enable, or you can individually control dynamically.

1.2. Quick Facts

Table 1.1. Summary of the PLL Module

IP Requirements	Supported FPGA Family	CrossLink™-NX, Certus™-NX, CertusPro™-NX
	IP Version	Version 1.8.0
Resource Utilization	Targeted Devices	LIFCL-40, LIFCL-33, LIFCL-17, LFD2NX-40, LFD2NX-17, LFCPNX-100
	Supported User Interface	LMMI, APB
Design Tool Support	Lattice Implementation	Lattice Radiant® Software
	Synthesis	Lattice Synthesis Engine (LSE)
		Synopsys® Synplify Pro® for Lattice
	Simulation	For the list of supported simulators, see the Lattice Radiant Software User Guide.

1.3. Features

Key features of the PLL Module include:

- Multiple feedback inputs
- Multiple output clocks
- Output synchronization
- Lock detector for phase and frequency
- M (1 to 128), N (1 to 128) input and feedback dividers
- Fractional-N divider
- VCO phase shift – 8 VCO phase
- Divider phase shift
- Dynamic VCO and divider phase shift
- Output dividers (1 to 128)
- Output edge trim for clkop_o and clkos_o only
- Input bypass
- Power down mode
- Test mode
- Programmable bandwidth

1.4. Licensing and Ordering Information

The PLL Module is provided at no additional cost with the Lattice Radiant software.

1.5. IP Validation Summary

The IP is validated with the following synthesis tool:

- Synopsys Synplify Pro Q-2023.1LR, Build 151R, February 7, 2023
- Lattice Synthesis Engine (LSE)

The following table shows the validation status for the PLL Module core. The ✓ mark indicates whether the IP has been validated for Simulation, Timing, or with Hardware.

Table 1.2. IP Validation Level

Device Family	IP Version	Validation Level		
		Simulation	Timing	Hardware
Lattice Nexus™	1.8.0	✓	✓	✓

1.6. Minimum Device Requirements

All devices listed in Table 1.1 support PLL Module.

1.7. Naming Conventions

1.7.1. Nomenclature

The nomenclature used in this document is based on Verilog HDL.

1.7.2. Signal Names

- _*n* are active low (asserted when value is logic 0).
- _*i* are input signals.
- _*o* are output signals.

2. Functional Description

2.1. IP Architecture Overview

The PLL Module is capable of clock synthesis and clock phase management including clock tree delay cancellation.

The PLL Module performs frequency synthesis, multiplication, division, and clock injection delay removal. For clock synthesis application where removal of clock injection delay is not a requirement, the Global Phase-Locked Loop (GPLL) block has an internal feedback path that is used to complete the feedback loop. Applications that require the removal of the clock injection delay take the feedback clock form the output of the relevant clock tree.

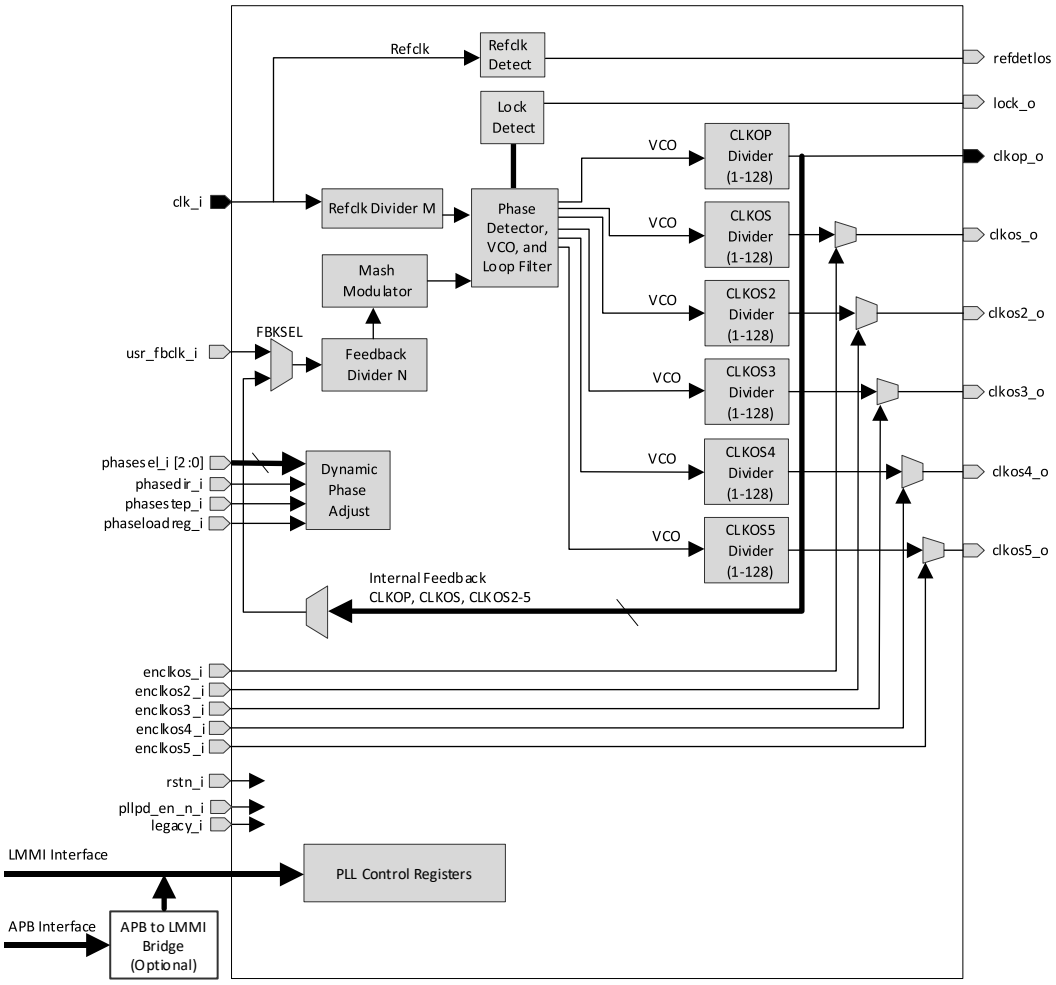

Figure 2.1. GPLL Module Top Level Block Diagram

This PLL Module has the flexibility of multiple feedback clock source selections. It has main logic features as six independent output clocks, output phase shifting through VCO phases, synchronous outputs, and divider phase shifts. The effective feedback divider value cannot exceed 80.

2.2. Clocking

The operation frequency range of the input clock, user feedback clock, and output clocks of the PLL Module are provided in the Lattice Nexus device family datasheet.

2.3. Reset

There is one asynchronous reset (rstn_i) from FPGA fabric to reset the PLL Module. The minimal pulse width for the rstn_i is specified in Lattice Nexus device family datasheet.

2.4. User Interfaces

The PLL Module provides optional LMMI interface or APB interface through a built-in APB to LMMI bridge to allow you to access PLL control registers for debugging.

Table 2.1. User Interfaces and Supported Protocols

User Interface	Supported Protocols	Description
Control	LMMI	Lattice Memory Mapped Interface
Control	APB	AMBA Advanced Peripheral Bus, through APB to LMMI bridge

3. IP Parameter Description

The configurable attributes of the PLL Module are shown in the following tables. You can configure the IP by setting the attributes accordingly in the IP Catalog Module/IP wizard of the Lattice Radiant software.

3.1. Attribute Summary

Wherever applicable, default values are in bold.

Table 3.1. PLL Module Attributes

Attribute	Selectable Values	Description
General		
Configuration Mode	**Frequency** Divider	Selects the configuration mode. Frequency – Sets the desired input and output frequency. Divider – Sets the desired input frequency and desired divider settings.
Set Parameter Optimization Target	Minimum Jitter (Higher VCO)	Display only.
Enable Fractional-N Divider	Checked **Unchecked**	Enables or disables the Fractional Feedback Clock divider.
Enable Spread Spectrum Clock Generation	Checked **Unchecked**	Enables the Spread Spectrum Clock Generation.
Enable User Feedback Clock	Checked **Unchecked**	When enabled, feedback clock is from user input. Not applicable when *Enable Fractional-N Divider* is checked.
Enable Internal Path Switching	Checked **Unchecked**	Enables or disables the internal path switching during POR, Sleep, and Standby modes.
VCO Frequency	Calculated	Display only.
Reference Clock		
CLKI: Frequency (MHz)	18 – 800 **100**	Sets the Reference Clock frequency. Applicable for *Frequency Configuration Mode* only.
CLKI: Divider Actual Value	Calculated	Display only. Applicable for *Frequency Configuration Mode*.
CLKI: Divider Desired Value	**1** to 128	Sets the Feedback Clock divider. Applicable for *Divider Configuration Mode* only.
Phase Detector Frequency (MHz)	Calculated	Display only.
Enable Reference Clock Monitor	Checked **Unchecked**	Enables or disables the Reference Clock Monitor.
Select Monitor Clock Frequency	**3.2 MHz** 1.0 MHz	Selects the frequency for reference clock monitoring logic. Applicable when *Enable Reference Clock Monitor* is checked.
Feedback		
CLKFB: Feedback Mode	**CLKOP** CLKOS CLKOS2 CLKOS3 CLKOS4 CLKOS5 INTCLKOP INTCLKOS INTCLKOS2 INTCLKOS3 INTCLKOS4 INTCLKOS5	Selects the feedback clock from the enabled PLL clock outputs (internal or external).

Attribute	Selectable Values	Description
CLKFB: FBK Divider Desired Value (Integer)	**1** to 16	Sets the Feedback Clock divider. Applicable for *Divider Configuration Mode* only. When the VCO is 800 MHz, the feedback divider must be greater than 1.
CLKFB: FBK Divider Actual Value (Integer)	1 to 128	Sets the Feedback Clock divider. Applicable for *Frequency Configuration Mode* only.
CLKFB: FBK Divider Desired Value (Fractional)	**0** to 4095	Sets the Feedback Clock fractional divider. Applicable if *Fractional-N Divider* is enabled.
CLKFB: FBK Divider Actual Value (Fractional)	Calculated	Display only.
CLKFB: FBK Divider Actual Value (Float)	Calculated	Display only (integer and fractional).
Primary Clock Output		
CLKOP: Bypass	Checked **Unchecked**	Bypasses the actual divider output and outputs the reference clock instead.
CLKO*: Frequency Desired Value (MHz)	6.25 – 800 **100**	Sets the Output Clock frequency. Applicable for Frequency mode only.
CLKOP: Tolerance (%)	**0**, 0.1, 0.2, 0.5, 1, 2, 5, 10	Sets the acceptable tolerance for the actual output frequency against the desired output frequency.
CLKOP: Divider Desired Value	1 to 128 **8**	Sets the Output Clock frequency. Applicable for Divider mode only.
CLKOP: Divider Actual Value	Calculated	Display Only.
CLKOP: Frequency Actual Value (MHz)	Calculated	Display Only.
CLKOP: ERROR (PPM)	Calculated	Display Only. Difference between desired and actual frequencies.
CLKOP: Enable Trim for CLKOP	Checked **Unchecked**	Enables or disables Trim for clock output.
CLKOP: Duty Trim Options Mode	Rising **Falling**	Selects Trim mode.
CLKOP: Duty Trim Options Delay Multiplier	**0**, 1, 2, 4	Selects Trim Delay Multiplier.
Secondary Clock Output		
CLKOS*: Enable	Checked **Unchecked**	Enables or disables PLL Clock Output.
CLKOS*: Bypass	Checked **Unchecked**	Bypasses the actual divider output and outputs the reference clock instead.
CLKOS*: Frequency Desired Value (MHz)	6.25 – 800 **100**	Sets the Output Clock frequency. Applicable for Frequency mode only.
CLKOS*: Tolerance (%)	**0**, 0.1, 0.2, 0.5, 1, 2, 5, 10	Sets the acceptable tolerance for the actual output frequency against the desired output frequency.
CLKOS*: Divider Desired Value	1 to 128 **8**	Sets the Output Clock frequency. Applicable for Divider mode only.
CLKOS*: Divider Actual Value	Calculated	Display Only.
CLKOS*: Frequency Actual Value (MHz)	Calculated	Display Only.
CLKOS*: Static Phase Shift (Degrees)	**0**, 45, 90, 135, 180, 225, 270, 315	Sets the desired clock output phase.
CLKOS*: ERROR (PPM)	Calculated	Display Only. Difference between desired and actual frequencies.
CLKOS*: Enable Trim for CLKOS*	Checked	Enables or disables Trim for clock output.

Attribute	Selectable Values	Description
	Unchecked	
CLKOS*: Duty Trim Options Mode	Rising **Falling**	Selects Trim mode.
CLKO*: Duty Trim Options Delay Multiplier	**0**, 1, 2, 4	Selects Trim Delay Multiplier.
Optional Ports		
Reference Clock I/O Pin		
Set I/O Pin for PLL Reference Clock	Checked **Unchecked**	Enables or disables I/O pin option for reference clock.
I/O Standard for Reference Clock	**LVDS**, SUBLVDS, SLVS, HSTL15_I, HSTL15D_I, LVTTL33, LVCMOS33, LVCMOS25, LVCMOS18, LVCMOS18H, HSTL15D_I, LVCMOS15, LVCMOS15H, LVCMOS12, LVCMOS12H, LVCMOS10, LVCMOS10H, LVCMOS10R	Selects the type of I/O pin.
Dynamic Phase Control Ports		
Enable Dynamic Phase Ports	Checked **Unchecked**	Enables or disables dynamic phase control ports.
Clock Enable Ports		
CLKOP/CLKOS[n] Enable Port	Checked **Unchecked**	Sets to provide clock enable port.
PLL Reset		
Provide PLL Reset	Checked **Unchecked**	Sets to provide PLL reset port.
PLL Lock		
Provide PLL Lock Signal	Checked **Unchecked**	Sets to provide PLL lock port.
PLL Lock is Sticky	Checked **Unchecked**	Sets the behavior of PLL lock signal.
Register Interface		
Select Register Interface	**None** APB LMMI	Selects the type of register interface.
Power Mode Settings		
Enable Legacy Mode	Checked **Unchecked**	Sets to provide legacy port.
Enable Power Down Mode	Checked **Unchecked**	Sets to provide power down port.

3.2. IP Parameter Settings for Example Use Cases

This section provides IP parameter settings for typical use cases, where you need to setup the reference clock frequency and desired output clock frequency in Frequency Configuration Mode, or setup the desired divider ration in Divider Configuration Mode. The reset of the internal setup is automatically calculated by the PLL Module.

3.2.1. Calculating the Clock Dividing Ratio

The following table shows the clock dividing ratio based on reference clock and output clock frequency. You must set the settings in **Bold**, and the reset of the settings are automatically evaluated by the PLL Module.

Table 3.2. PLL Module Clock Dividing Ratio Calculation

The VCO frequency is 1,200 MHz for all scenarios in this table.

Scenario	Configuration Mode	Reference Clock (MHz)	Output Clock Desired (MHz)	Output Clock Actual (MHz)	CLKI Divider Actual	CLKI Divider Desired	Phase Detector Frequency (MHz)	FBCLK Divider Actual	FBCLK Divider Desired	CLKO Divider Actual	CLKO Divider Desired
1	**Frequency**	**60**	**120**	—	5	—	20	6	—	10	—
2	**Frequency**	**60**	**100**	—	5	—	20	6	—	12	—
3	**Frequency**	**60**	**80**	—	5	—	20	6	—	15	—
4	**Divider**	**100**	—	120	—	**5**	**20**	—	**6**	—	**10**
5	**Divider**	**100**	—	100	—	**5**	**20**	—	**6**	—	**12**
6	**Divider**	**100**	—	80	—	**5**	**20**	—	**6**	—	**15**

4. Signal Description

This section provides the information about the PLL Module ports function and the supported settings.

The following figure illustrates only the enabled feature based on IP configuration. For more details about the ports and supported functions, refer to the subsequent subsections.

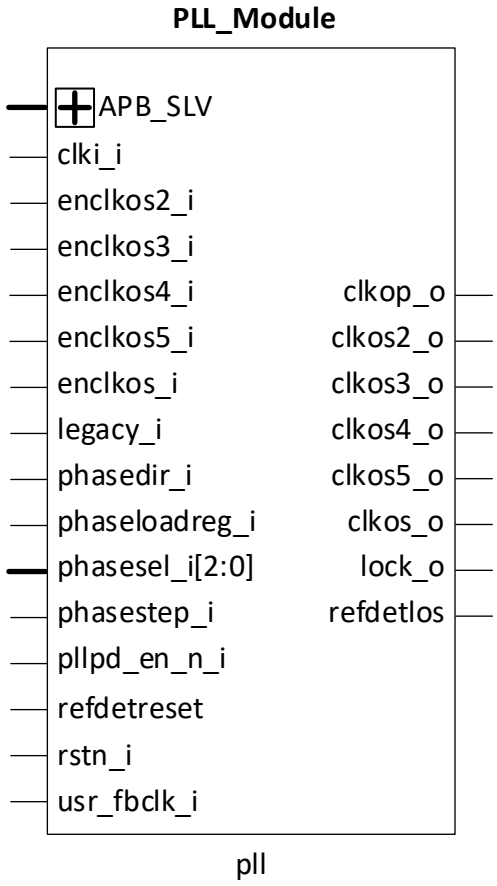

Figure 4.1. PLL Module Port

4.1. PLL Module Interface

The following table lists the top-level input and output signals for PLL Module. The port names in upper-case represent signals that connect to physical pins of the device and port names in lower-case represent internal signals.

Table 4.1. PLL Module Ports

Port	Type	Description
Clock and Reset		
clki_i	Input	System clock with supported frequency range 18 MHz – 500 MHz.
rstn_i	Input	PLL reset signal, can be enabled from the user interface via PLL Reset Options: *Provide PLL Reset*. Minimum reset pulse width is 1 ms.
clkop_o	Output	Primary output clock, always enabled.
clkos_o	Output	Secondary output clock, can be enabled from the user interface via *CLKOS: Enable*.
clkos2_o	Output	Secondary output clock, can be enabled from the user interface via *CLKOS2: Enable*.
clkos3_o	Output	Secondary output clock, can be enabled from the user interface via *CLKOS3: Enable*.
clkos4_o	Output	Secondary output clock, can be enabled from the user interface via *CLKOS4: Enable*.

Port	Type	Description
clkos5_o	Output	Secondary output clock, can be enabled from the user interface via *CLKOS5: Enable*.
usr_fbclk_i	Input	User Feedback Clock input. Available if selected from the user interface via *Enable User Feedback Clock*.
User Interface		
legacy_i	Input	PLL Legacy mode signal, can be enabled from the user interface via *Power Mode Settings: Enable Legacy Mode*.
pllpd_en_n_i	Input	Power Down mode signal, can be enabled from the user interface via *Power Mode Settings: Enable Power Down Mode*.
phasedir_i	input	Dynamic Phase direction signal, can be enabled from the user interface via *Optional Port Selections: Dynamic Phase Ports*.
phasesetp_i	input	Dynamic Phase step signal, can be enabled from the user interface via *Optional Port Selections: Dynamic Phase Ports*.
phaseloadreg_i	Input	Dynamic Phase load signal, can be enabled from the user interface via *Optional Port Selections: Dynamic Phase Ports*.
Phasesel_i [2:0]	Input	Dynamic Phase select signal, can be enabled from the user interface via *Optional Port Selections: Dynamic Phase Ports*.
enclkop_i	Input	CLKOP Enable signal, can be enabled from the user interface via *Optional Port Selections: Clock Enable Ports* to populate displayed ports for CLKOP.
enclkos_i	Input	CLKOS Enable signal, can be enabled from the user interface via *Optional Port Selections: Clock Enable Ports* to populate displayed ports for CLKOS.
enclkos2_i	Input	CLKOS2 Enable signal, can be enabled from the user interface via *Optional Port Selections: Clock Enable Ports* to populate displayed ports for CLKOS2.
enclkos3_i	Input	CLKOS3 Enable signal, can be enabled from the user interface via *Optional Port Selections: Clock Enable Ports* to populate displayed ports for CLKOS3.
enclkos4_i	Input	CLKOS4 Enable signal, can be enabled from the user interface via *Optional Port Selections: Clock Enable Ports* to populate displayed ports for CLKOS4.
enclkos5_i	Input	CLKOS5 Enable signal, can be enabled from the user interface via *Optional Port Selections: Clock Enable Ports* to populate displayed ports for CLKOS5.
lock_o	Output	PLL lock signal, can be enabled from the user interface via *Lock Settings: Provide PLL Lock Signal*.
refdetlos	Output	Reference clock detection lost signal, can be enabled from user interface via *Reference Clock Setting: Enable Reference Clock Monitor*.
LMMI Interface[1]		
lmmi_clk_i	Input	LMMI clock.
lmmi_resetn_i	Input	Reset (active low). Resets the LMMI interface and sets registers to default values. Does not reset the internals of the Hard IP block.
lmmi_request_i	Input	Start transaction.
lmmi_wr_rdn_i	Input	Write = HIGH, Read = LOW.
lmmi_offset_i [6:0]	Input	Offset (0–31 bits) – register offset within the target, starting at offset 0. Bit width is hard IP dependent.
lmmi_wdata_i [7:0]	Input	Write data (0–16 bits). Bit width is hard IP dependent.
lmmi_rdata_o [7:0]	Output	Read data (0–16 bits). Bit width is hard IP dependent.
lmmi_rdata_valid_o	Output	Read transaction is complete and lmmi_rdata_o[] contains valid data.

Port	Type	Description
lmmi_ready_o	Output	Target is ready to start a new transaction. Target can insert wait states by holding this signal low. This signal is optional in hardware and mandatory in RTL soft wrappers. Hardware targets may omit this signal if they do not need wait states. For zero-wait-state targets, the soft wrapper ties this signal high.
APB Interface[1]		
apb_pclk_i	Input	APB clock.
apb_preset_n_i	Input	APB reset (active low).
apb_paddr_i [6:0]	Input	APB Address signal.
apb_psel_i	Input	APB Select signal.
apb_penable_i	Input	APB Enable signal.
apb_pwrite_i	Input	APB Direction signal.
apb_pwdata_i [7:0]	Input	APB Write Data signal.
apb_pready_o	Output	APB Ready signal.
apb_prdata_o [7:0]	Output	APB Read Data signal.
apb_pslverr_o	Output	APB Responder Error signal.

Note:

1. Only one of the two interfaces is available as selected by Interface attribute. For more details, refer to *Appendix C. PLL LMMI Operation* in the sysCLOCK PLL Design and User Guide for Nexus Platform (FPGA-TN-02095).

5. Register Description

For PLL LMMI operation and register map, refer to *Appendix C. PLL LMMI Operation* in the sysCLOCK PLL Design and User Guide for Nexus Platform (FPGA-TN-02095).

6. Example Design

This example design shows the utilization of the PLL Module to generate the system clock which must be used for any sequential logic design.

6.1. Example Design Supported Configuration

Table 6.1. PLL Module Configuration Supported by the Example Design

Attribute	Example Design Setting	Description
General		
Configuration Mode	Frequency	Selects the *Frequency* configuration mode – sets the desired input or output frequency.
Enable Fractional-N Divider	—	Disables the Fractional Feedback Clock Divider.
Enable Spread Spectrum Clock Generation	—	Disables the Spread Spectrum Clock Generation.
Enable User Feedback Clock	—	Uses internal feedback clock.
Enable Internal Path Switching	—	Disables the internal path switching during POR, Sleep, and Standby modes.
Reference Clock		
CLKI: Frequency (MHz)	50	Sets the Reference Clock frequency to 50 MHz.
Enable Reference Clock Monitor	—	Disables the Reference Clock Monitor.
Feedback		
CLKFB: Feedback Mode	INTCLKOP	Selects the feedback clock from the enabled PLL clock outputs (internal or external).
Primary Clock Output		
CLKOP: Bypass	Unchecked	Bypasses the actual divider output and outputs the reference clock instead.
CLKOP: Frequency Desired Value (MHz)	100	Sets the Output Clock frequency to 100 MHz.
CLKOP: Tolerance (%)	0	Sets the acceptable tolerance for the actual output frequency against the desired output frequency.
CLKOP: Enable Trim for CLKOP	Unchecked	Disables Trim for clock output.
Secondary Clock Output		
CLKOS*: Enable	Unchecked	Disables Secondary PLL Clock Outputs.
Optional Ports		
Reference Clock I/O Pin		
Set I/O Pin for PLL Reference Clock	Unchecked	Disables I/O pin option for reference clock.
Dynamic Phase Control Ports		
Enable Dynamic Phase Ports	Unchecked	Disables dynamic phase control ports.
Clock Enable Ports		
CLKOP/CLKOS[n] Enable Port	Unchecked	Disables the clock enable port.
PLL Reset		
Provide PLL Reset	Checked	Sets to provide PLL reset port.
PLL Lock		
Provide PLL Lock Signal	Checked	Sets to provide PLL lock port.
PLL Lock is Sticky	Unchecked	Sets the PLL lock signal un-sticky.

Attribute	Example Design Setting	Description
Register Interface		
Select Register Interface	None	Selects no register interface.
Power Mode Settings		
Enable Legacy Mode	Unchecked	Sets to provide legacy port.
Enable Power Down Mode	Unchecked	Sets to provide power down port.

6.2. Overview of the Example Design and Features

This example design shows the following capabilities:

- Generate 100 MHz system clock for user logic or micro controller.
- Use on-die oscillator clock as reference clock for PLL Module.
- Synchronize the release of reset asserted before PLL is locked.

6.3. Example Design Components

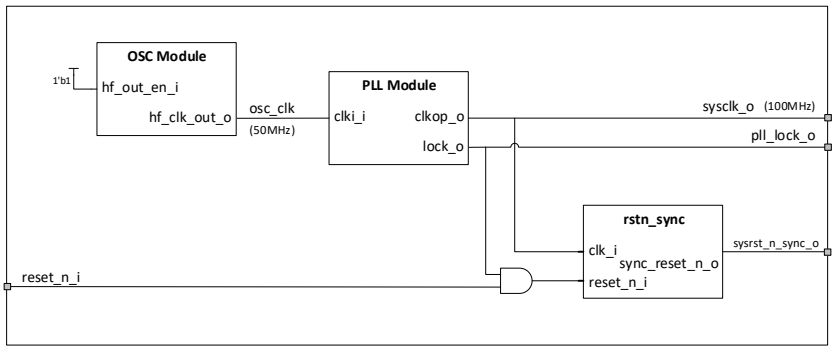

Figure 6.1. PLL Module Example Design Block Diagram

The PLL Module example design includes the following blocks:

- OSC Module
- PLL Module
- rstn_sync Module

6.4. OSC Module

The OSC Module generates the 50 MHz clock from internal oscillator.

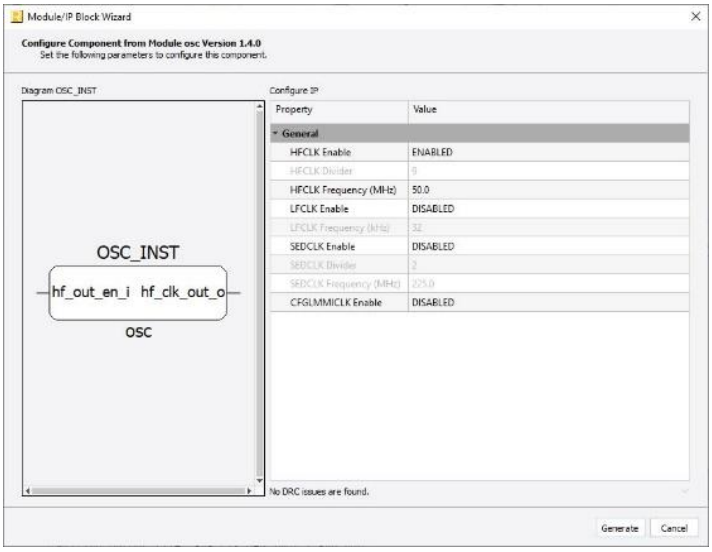

Figure 6.2. OSC Module Configuration in Example Design

6.5. PLL Module

The PLL Module generates the system clock for user logic.

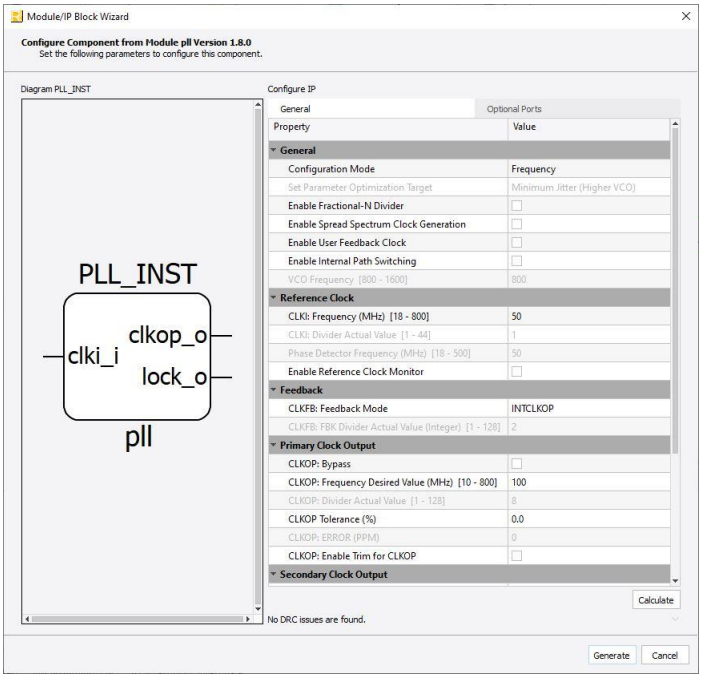

Figure 6.3. PLL Module Configuration in Example Design

6.6. rstn_sync Module

The rstn_sync module creates the synchronized asynchronous reset from asynchronous input and pll_lock signal. The sysrst_n_sync_o output from the rstn_sync module can be asynchronously asserted but is released synchronously. The sysrst_n_sync_o is asserted before the PLL Module is locked.

6.7. Simulating the Example Design

This example can be simulated with any user design. No specific test bench is needed for this example design.

6.8. Hardware Testing

The demo design was tested using the LIFCL-NX33U Evaluation Board.

7. Designing with the IP

This section provides information on how to generate the PLL Module using the Lattice Radiant software and how to run simulation and synthesis. For more details on the Lattice Radiant software, refer to the Lattice Radiant Software User Guide.

7.1. Generating and Instantiating the IP

You can use the Lattice Radiant software to generate IP modules and integrate them into the device architecture.

To generate the PLL Module in the Lattice Radiant software, follow these steps:

1. Create a new Lattice Radiant software project or open an existing project.
2. In the **IP Catalog** tab, double-click **PLL** under **Module, Architecture_Modules** category. The **Module/IP Block Wizard** opens as shown in the following figure. Enter values in the **Component name** and the **Create in** fields and click **Next**.

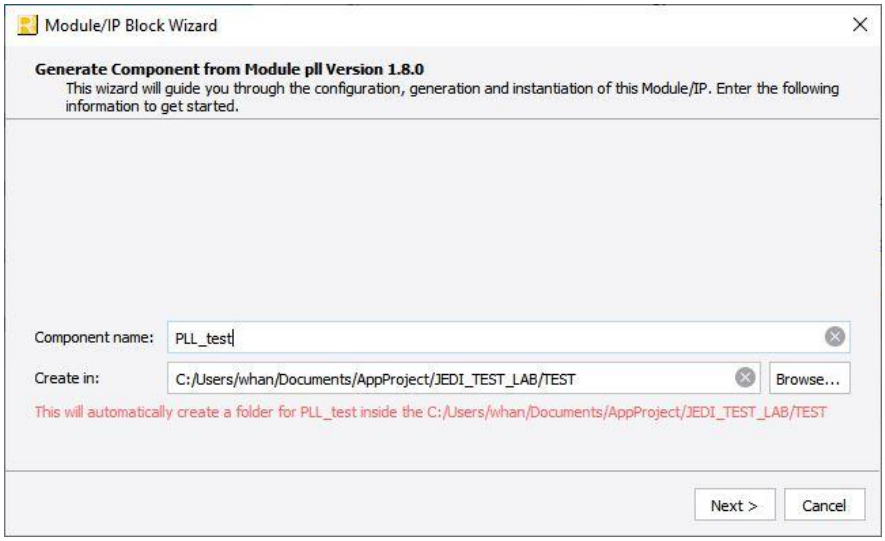

Figure 7.1. Configuring the Module/IP Block Wizard

3. In the next **Module/IP Block Wizard** window, customize the selected PLL Module using drop-down lists and check boxes. The following figures show the example configuration of the PLL Module. For details on the configuration options, refer to the IP Parameter Description section.

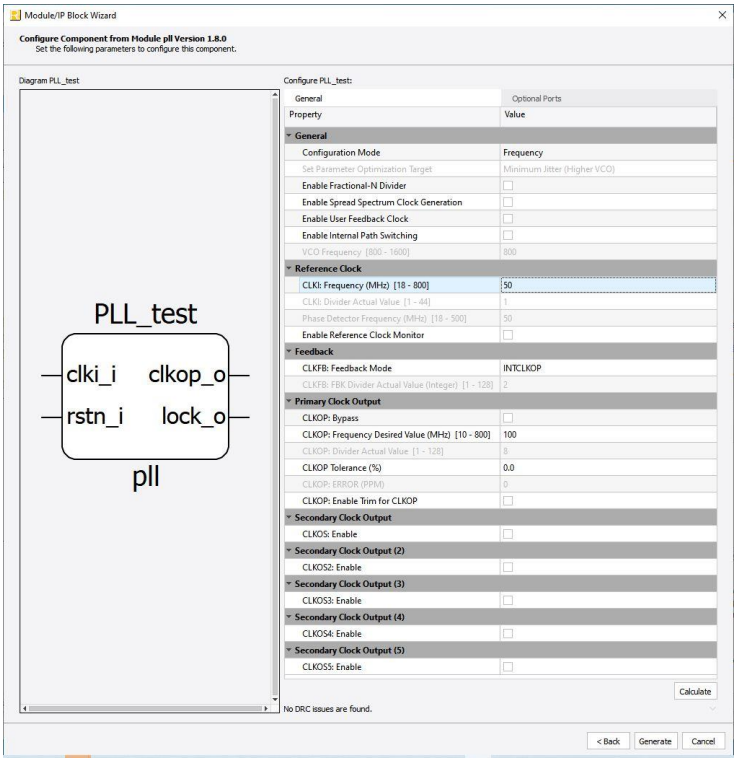

Figure 7.2. Configuring the User Interface of the PLL Module — General Tab

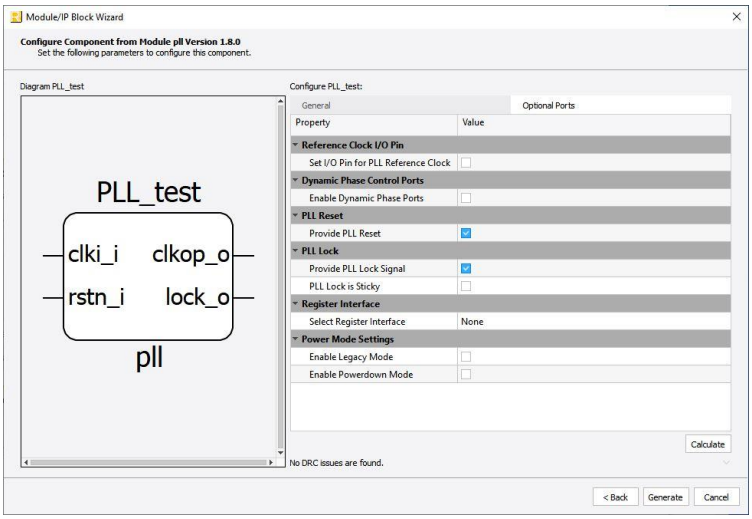

Figure 7.3. Configuring the User Interface of PLL Module — Optional Ports Tab

4. Click **Generate**. The **Check Generated Result** dialog box opens, showing design block messages and results as shown in the following figure.

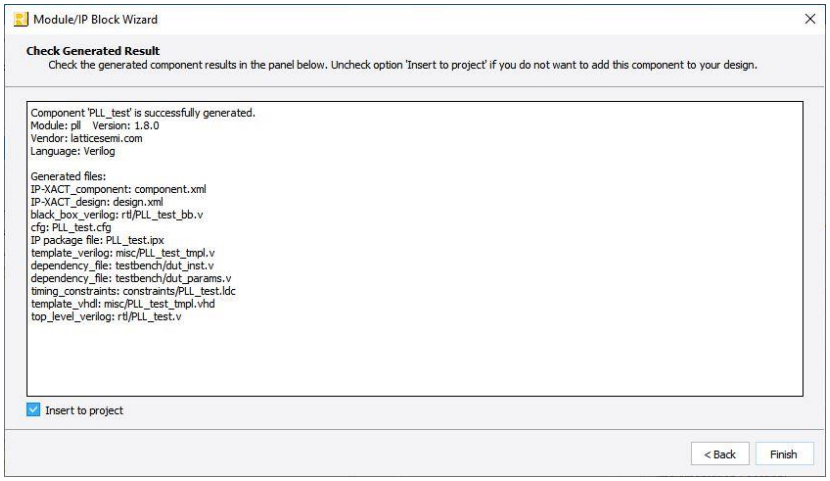

Figure 7.4. Check Generated Result

5. Click **Finish**. All the generated files are placed under the directory paths in the **Create in** and the **Component name** fields shown in Figure 7.1.

7.1.1. Generated Files and File Structure

The generated PLL Module package includes the closed-box (<Component name>_bb.v) and instance templates (<Component name>_tmpl.v/vhd) that can be used to instantiate the core in a top-level design. An example RTL top-level reference source file (<Component name>.v) that can be used as an instantiation template for the module is also provided. You may also use this top-level reference as the starting template for the top-level for their complete design. The generated files are listed in the following table.

Table 7.1. Generated File List

Attribute	Description
<Component name>.ipx	This file contains the information on the files associated to the generated IP.
<Component name>.cfg	This file contains the parameter values used in IP configuration.
component.xml	Contains the ipxact:component information of the IP.
design.xml	Documents the configuration parameters of the IP in IP-XACT 2014 format.
rtl/<Component name>.v	This file provides an example RTL top file that instantiates the module.
rtl/<Component name>_bb.v	This file provides the synthesis closed-box.
misc/<Component name>_tmpl.v misc /<Component name>_tmpl.vhd	These files provide instance templates for the module.

7.2. Design Implementation

Completing your design includes additional steps to specify analog properties, pin assignments, and timing and physical constraints. You can add and edit the constraints using the Device Constraint Editor or by manually creating a PDC File.

Post-Synthesis constraint files (.pdc) contain both timing and non-timing constraint.pdc source files for storing logical timing and physical constraints. Constraints that are added using the Device Constraint Editor are saved to the active .pdc file. The active post-synthesis design constraint file is then used as input for post-synthesis processes.

Refer to the relevant sections in the Lattice Radiant Software User Guide for more information on how to create or edit constraints and how to use the Device Constraint Editor.

7.3. Timing Constraints

The timing constraints are based on the clock frequency used. The timing constraints for the IP are defined in the relevant constraint files. The following example shows the IP timing constraints generated for the PLL Module.

```
set CLKI_FREQ 50.000000
set CLK_PERIOD [expr {double(round(1000000/$CLKI_FREQ))/1000}]
create_clock -name {clki_i} -period $CLK_PERIOD [get_ports clki_i]
```

Figure 7.5. Timing Constraint File (.pdc) for the PLL Module

For timing closure, add the following timing constraints in the .pdc file:

```
If {$LMMI_EN == 1} {
   create_clock -name {lmmi_clk_i} -period 8 -waveform {0 4} [get_ports lmmi_clk_i]
   set_multicycle_path -setup 2 -from [get_clocks clki_i] -to [get_clocks lmmi_clk_i]
} else {
   If {$APB_EN == 1} {
      create_clock -name {apb_pclk_i} -period 8 -waveform {0 4} [get_ports apb_pclk_i]
      set_multicycle_path -setup 2 -from [get_clocks clki_i] -to [get_clocks apb_pclk_i]
   }
}
```

Refer to Lattice Radiant Timing Constraints Methodology (FPGA-AN-02059) for details on how to constraint your design.

7.4. Physical Constraints

By default, the Lattice Radiant software automatically assigns the location of the PLL Module you specified. If you want to specify the PLL location to meet your needs, you can add the following physical constraints in the .pdc file:

```
ldc_set_location -site {PLL_LLC} [get_cells {pll_instance_name}]
```

Where the *PLL_LLC* refers to the PLL at the lower left corner. You can set the location to *PLL_ LRC* (PLL at the lower right corner), *PLL_ULC* (PLL at the upper left corner), or *PLL_URC* (PLL at the upper right corner) based on the requirement.

7.5. Specifying the Strategy

The Lattice Radiant software provides two predefined strategies: Area and Timing. The software also enables you to create customized strategies. For details on how to create a new strategy, refer to the *Strategies* section of the Lattice Radiant Software user guide.

7.6. Running Functional Simulation

You can run functional simulation after the IP is generated. For PLL Module, a customer testbench is provided.

To run functional simulation, follow these steps:

1. Go to **File List** tab, right click **Input Files**.
2. Click **Add** > **Existing File**.
3. A window appears. Open the instance folder **testbench**, then double-click **tb_top.v**.
4. Right-click the **tb_top.v** under **Input files** folder.
5. Choose **Include for** > **Simulation**.

To run the simulation, follow these steps:

1. Click the  button located on the **Toolbar** to initiate the **Simulation Wizard** shown in the following figure.

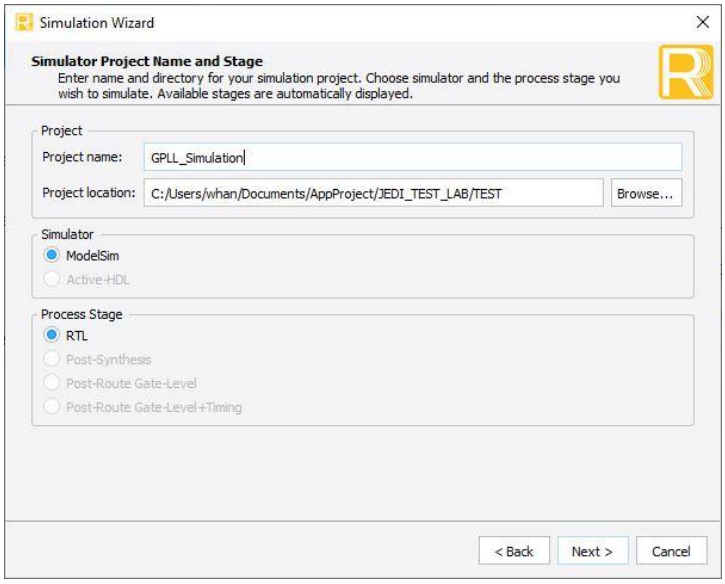

Figure 7.6. Simulation Wizard

2. Click **Next** to open the **Add and Reorder Source** window as shown in the following figure.

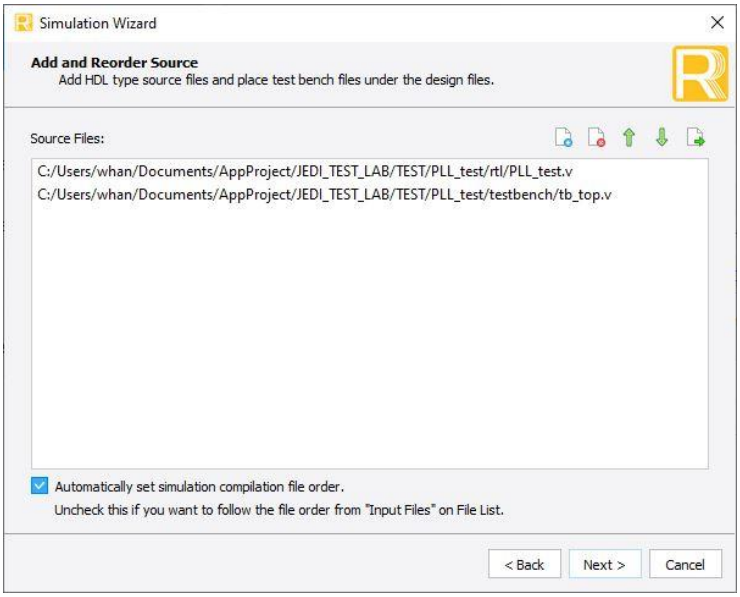

Figure 7.7. Add and Reorder Source

3. The parsing results are shown in the following figure. Click **Next**.

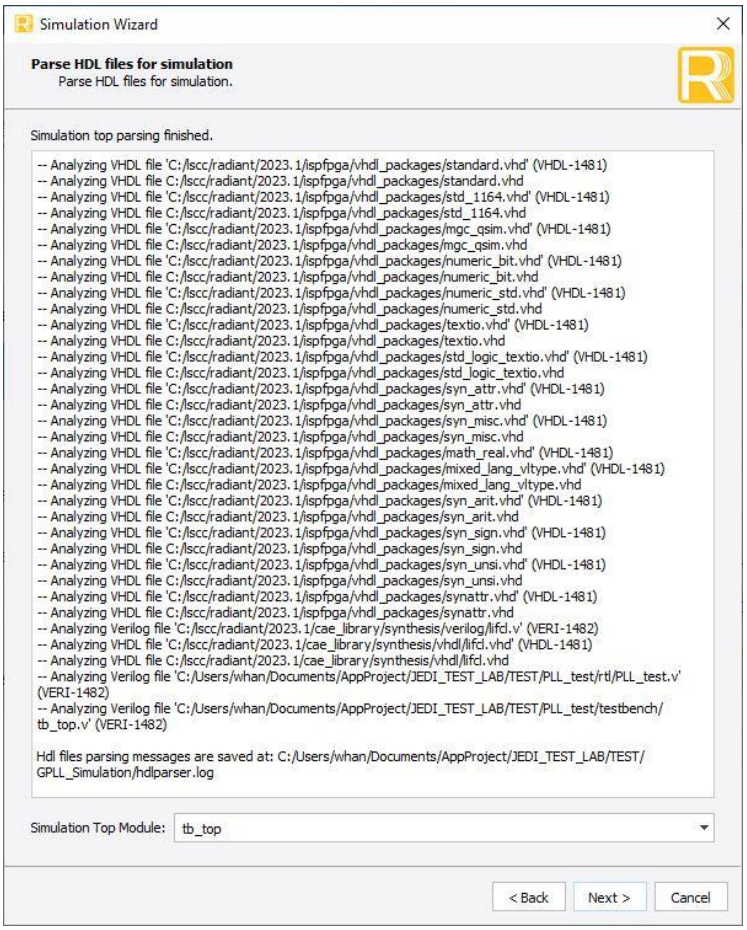

Figure 7.8. Parsing Summary for Simulation

4. In the Summary window shown in the following figure, click **Finish** to run the simulation.

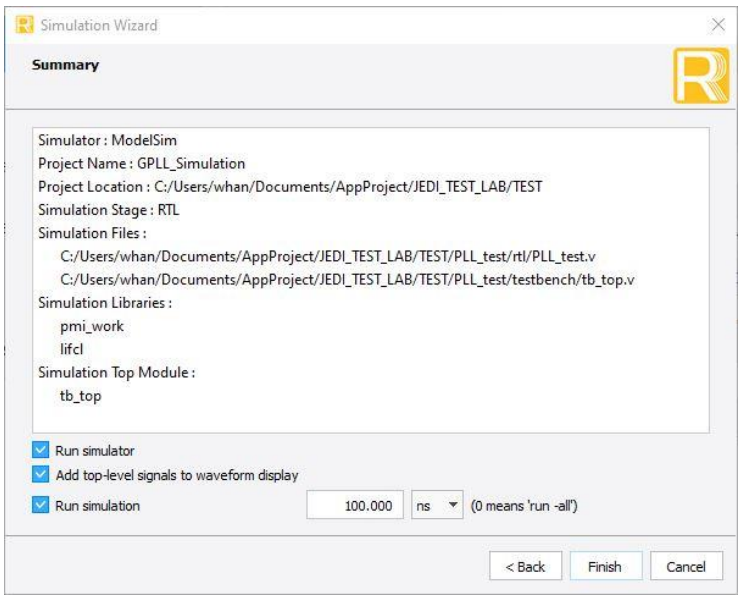

Figure 7.9. Summary Window for Simulation

It is necessary to follow the procedure above until it is fully automated in the Lattice Radiant Software Suite. The waveform in the following figure shows an example simulation result.

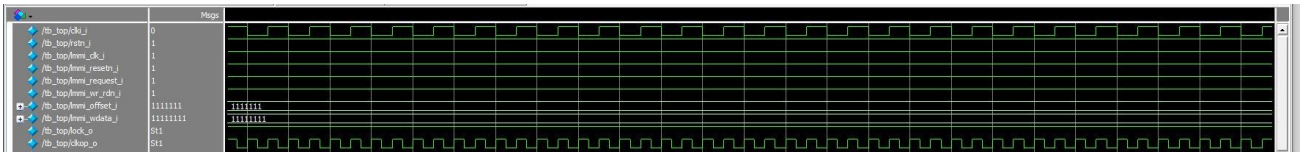

Figure 7.10. Simulation Waveform

7.6.1. Simulation Results

From the simulation results, the frequency of the reference clock input (clki_i) and clock output (clkop_o) are observed and compared to the expected value, where clki_i must be at 50 MHz and clkop_o must be at 100 MHz.

8. Debugging

This section lists the possible issues and the suggested troubleshooting steps that you can follow.

8.1. Debug Methods

8.1.1. Hardware Detection Failure

If the PLL outputs do not perform as expected, follow the steps shown in the following flow diagram.

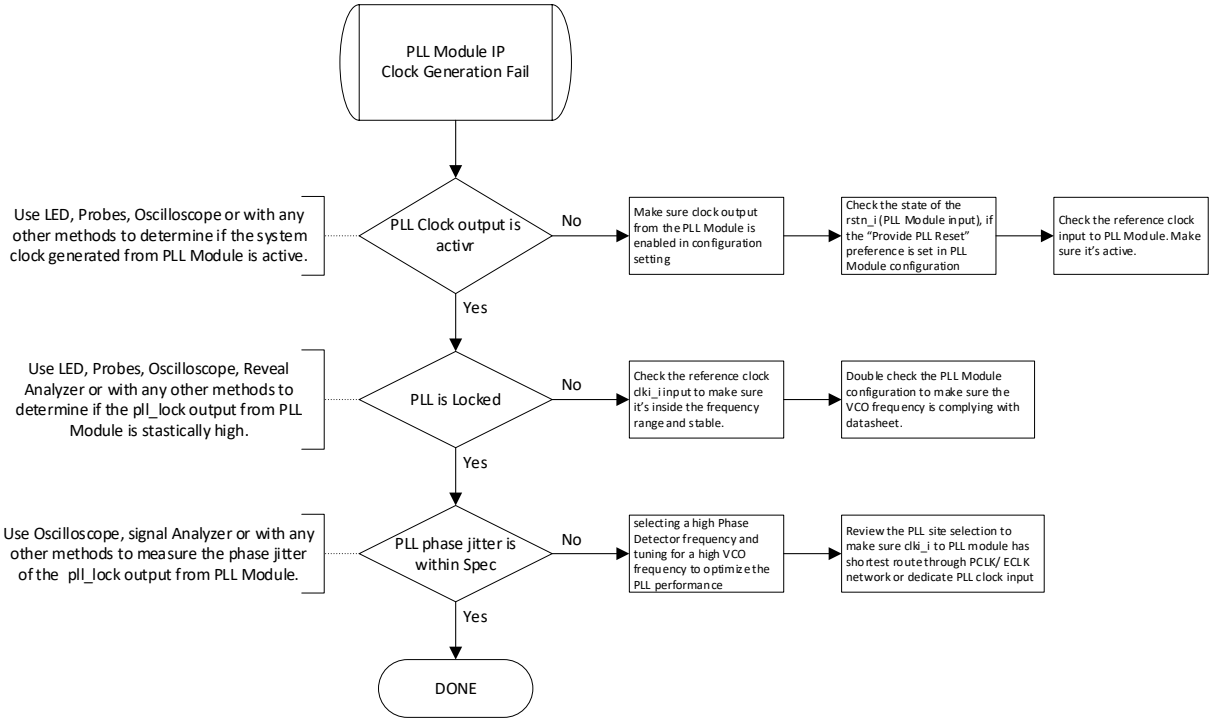

Figure 8.1. Hardware Detection Failure Debugging Flow

9. Design Considerations

When deploying the PLL Module in your design, consider the following guidelines:

- Ensure the reference clock input to the PLL Module is stable and less glittering.
- Select the PLL location to ensure clki_i to the PLL Module has the shortest route through PCLK network, ECLK network, or the dedicate PLL clock input.
- Reduce the noise on power supply on the board if possible.
- Selecting a high phase detector frequency and tuning for a high VCO frequency to optimize the PLL performance (reduce the phase jitter).

References

- sysCLOCK PLL Design and User Guide for Nexus Platform (FPGA-TN-02095)
- Lattice Radiant Software User Guide
- CrossLink-NX web page
- CertusPro-NX web page
- Certus-NX 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

Document Revision 2.0, Lattice Radiant SW version 2024.1, October 2024

Section	Change Summary
IP Parameter Description	Updated the description for the *CLKFB: FBK Divider Desired Value (Integer)* attribute in Table 3.1. PLL Module Attributes.

Document Revision 1.9, Lattice Radiant SW version 2024.1, September 2024

Section	Change Summary
Acronyms in This Document	Added definition for AMBA and GPLL.

Introduction	• Reworked section 1 *Introduction* and renamed to subsection 1.1 Overview of the IP.
	• Reworked subsection 1.1 *Quick Facts* and moved to subsection 1.2 Quick Facts.
	• Moved subsection 1.2 *Features* to subsection 1.3 Features.
	• Added the following subsections:
	  • 1.4 Licensing and Ordering Information
	  • 1.5 IP Validation Summary
	  • 1.6 Minimum Device Requirements
	• Reworked subsection 1.3 *Conventions* and renamed to subsection 1.7 Naming Conventions.

Functional Description	• Reworked subsection 2.1 *Overview* and renamed to subsection 2.1 IP Architecture Overview.
	• Added the following subsections:
	  • 2.2 Clocking
	  • 2.3 Reset
	  • 2.4 User Interfaces

IP Parameter Description	• Reworked subsection 2.3 *Attribute Summary* and renamed to subsection 3.1 Attribute Summary. • Added subsection 3.2 IP Parameter Settings for Example Use Cases.
Signal Description	Reworked subsection 2.2 *Signal Description* and moved to section 4 Signal Description.
Register Description	Added this section.
Example Design	Added this section.

Designing with the IP	• Reworked subsection 3.2 *Generating and Synthesizing the IP* and renamed to subsection 7.1 Generating and Instantiating the IP.
	• Added the following subsections:
	  • 7.2 Design Implementation
	  • 7.3 Timing Constraints
	  • 7.4 Physical Constraints
	  • 7.5 Specifying the Strategy
	• Reworked subsection 4.4 *Running the Functional Simulation* and renamed to subsection 7.6 Running Functional Simulation.

Debugging	Added this section.
Design Considerations	Added this section.
References	Updated references.

Document Revision 1.8, Lattice Radiant SW version 2.1, June 2024

Section	Change Summary
Functional Description	Updated the description for *rstn_i* port in Table 2.1. PLL Module Signal Description.

Document Revision 1.7, Lattice Radiant SW version 2.1, March 2024

Section	Change Summary
All	Renamed document from *PLL Module - Lattice Radiant Software* to *PLL Module*.
Disclaimers	Updated disclaimers.
Inclusive Language	Added inclusive language boilerplate.
Functional Description	• Updated the description for lmmi_offset_i, lmmi_wdata_i, and lmmi_rdata_o in Table 2.1. PLL Module Signal Description. • Updated the lower range for the *CLKI: Frequency (MHz)* parameter from 10 MHz to 18 MHz in Table 2.2. Attributes Table to match the data sheet specifications. • Changed *slave* to *target* for LMMI and changed *slave* to *responder* for APB in the Signal Description subsection.
IP Generation and Evaluation	Changed *black box* to *closed-box* in the Generating and Synthesizing the IP subsection.
References	Updated references.

Document Revision 1.6, Lattice Radiant SW version 2.1, August 2023

Section	Change Summary
Functional Description	Newly added *See FPGA-TN-02095, Appendix C. PLL LMMI Operation for more details* to Note 1 of Table 2.1. PLL Module Signal Description.
References	Newly added the links to sysCLOCK PLL Design and User Guide for Nexus Platform (FPGA-TN-02095), Lattice Radiant Software Web Page, CrossLink-NX Devices Web Page, CertusPro-NX Devices Web Page, Certus-NX Devices Web Page, and Lattice Insight for Training Series and Learning Plans.
Technical Support Assistance	Newly added the link to Lattice Answer Database.

Document Revision 1.5, Lattice Radiant SW version 2.1, September 2022

Section	Change Summary
Functional Description	Changed the Port Name rst_i to rstn_i in Table 2.1. PLL Module Signal Description.

Document Revision 1.4, Lattice Radiant SW version 2.1, July 2022

Section	Change Summary
Functional Description	Removed *Enable PMU Wait for Lock* from Table 2.2. Attributes Table.

Document Revision 1.3, Lattice Radiant SW version 2.1, June 2021

Section	Change Summary
Acronyms in This Document	Removed SSC.

Introduction	• Updated Table 1.1. Quick Facts.
	  • Revised Supported FPGA Families
	  • Revised Targeted Devices
	• Removed Spread Spectrum Clock Generation from the Features section.

Functional Description	• Updated description of *phasesel_i* port in Table 2.1. PLL Module Signal Description. • Removed Enable Spread Spectrum Clock Generation item and Spread Spectrum group from Table 2.2. Attributes Table.
All	Updated references to Lattice Radiant Software User Guide.

Document Revision 1.2, Lattice Radiant SW version 2.1, June 2020

Section	Change Summary
Introduction	• Updated Table 1.1 to add Certus-NX and LFD2NX-40 as targeted device. • Updated Lattice Implementation to Lattice Radiant 2.1. • Added items to Features section.
Functional Description	• Added ports to Table 2.1. PLL Module Signal Description. • General update to Table 2.2. Attributes Table.
All	• Updated references to Lattice Radiant Software User Guide. • Removed reference to Lattice Radiant Software 2.0 tutorials.

Document Revision 1.1, Lattice Radiant SW version 2.0, February 2020

Section	Change Summary
Introduction	Updated Table 1.1 to add LIFCL-17 as targeted device and minimal device needed.
Appendix	Removed this section.

Document Revision 1.0, Lattice Radiant SW version 2.0, December 2019

Section	Change Summary
All	• Updated document status from Preliminary to final. • Removed features not implemented in this release.
Appendix	Added this section.

Document Revision 0.80, Lattice Radiant SW version 2.0, October 2019

Section	Change Summary
All	Preliminary release

www.latticesemi.com