Introduction

The LogiCORE™ IP Clocking Wizard core creates HDL source code for a clock circuit customized for your clocking requirements. The wizard automatically selects an appropriate clocking primitive and allows for the configuration of buffering, feedback, and timing parameters for the clocking network. In addition, it interactively aids the selection of correct attributes for the selected primitive and allows overriding of any wizard-calculated parameter.

As well as providing the clocking circuit as source HDL, the wizard delivers summary information about the timing parameters calculated for the clock circuit as reported by the Xilinx® timing tools.

Features

- Accepts up to two input clocks and up to seven output clocks per clock network

- Automatically chooses correct clocking primitive for a selected device

- Automatically configures clocking primitive based on user-selected clocking features

- Automatically calculates VCO frequency for primitives with an oscillator, and provides multiply and divide values based on input and output frequency requirements

- Automatically implements overall configuration that supports phase shift and duty cycle requirements

- Provides the ability to override the selected clock primitive and any calculated attribute

- Optionally buffers clock signals

- Provides timing estimates for the clock circuit and Xilinx Power Estimator (XPE) parameters

- Provides a synthesizeable example design including the clocking network and a simulation test bench

- Provides optional ports for the selected primitive

LogiCORE IP Facts				
Core Specifics				
Supported Device Family	Virtex®-6 Spartan®-6			
Resources Used	I/O	LUTs	FFs	Block RAMs
	1-2	0-1	0	0
Special Features	DCM, DCM_CLKGEN, MMCM, PLL			
Provided with Core				
Documentation	Product Specification Getting Started Guide Release Notes			
Design File Formats	Verilog and VHDL			
Constraints File	.ucf (user constraints file)			
Verification	Verilog and VHDL Test Bench			
Instantiation Template	Verilog and VHDL Wrapper			
Additional Items	Virtex-6 and Spartan-6 User Guides			
Design Tool Requirements				
Xilinx Implementation Tools	ISE® v11.4			
Verification and Simulation Tools	Mentor Graphics ModelSim 6.4b and above, Cadence IUS v8.1-s009 and above, Synopsys VCS and VCS MX 2008.09 and above, ISIM			
Synthesis	Synplify Pro C-2009.06, XST			

Functional Overview

The Clocking Wizard is an interactive Graphical User Interface (GUI) that creates a clocking network based on design-specific needs. The required clock network parameters are gathered in a linear manner to allow you to select only the desired parameters. Using the wizard, experienced users can explicitly configure their chosen clocking primitive; while less experienced users can let the wizard automatically determine the optimal primitive and configuration, based on the features required for their individual clocking networks.

For users who are already familiar with the Digital Clock Manager (DCM) and Phase-Locked Loop (PLL) wizards, see the Migration Guide Appendix in the Clocking Wizard Getting Started Guide for usage differences for the Clocking Wizard.

Clocking Features

Major functional clocking features can be used to determine an appropriate primitive. Incompatible features are then automatically displayed as dimmed to assist the designer with evaluating and selecting feature trade-offs.

Clocking features include:

- Frequency synthesis
- Phase alignment
- Minimization of output jitter
- Allowance of larger input jitter
- Minimization of power
- Dynamic phase shift
- Dynamic reconfiguration

Input Clocks

One input clock is the default behavior, but two input clocks can be chosen by selecting a secondary clock source. Only the timing parameters of the input clocks in their specified units is required; the wizard uses these parameters as needed to configure the output clocks.

Output Clocks

The number of required output clocks is user-configurable. The maximum number used depends upon the selected device and interactions with other major clocking features. You can input desired timing parameters such as frequency, phase, and duty cycle. The clocking wizard then configures the clocking primitive to comply exactly to the requested parameters. However, if this is not possible (based on the available input clocks) then best-attempt values are provided. The clocks are ordered such that CLK_OUT1 is the highest-priority clock and is most likely to comply to the requested timing parameters. Frequency parameters are chosen before phase and duty cycle parameters. Note that the port names may be different than the original primitive used.

Clock Buffering and Feedback

In addition to configuring the clocking primitive within the device, the wizard also assists with constructing the clocking network. Buffering options are provided for both input and output clocks. If a clock output requires special buffers like BUFPLL which the wizard does not generate in the design, alert messages are flagged to the user. Feedback for the primitive can be user-controlled or left to the wizard to automatically connect. If automatic feedback is selected, the feedback path is matched to timing for CLK_OUT1.

Optional Ports

All primitive ports are available for user-configuration. You can expose any of the ports on the clocking primitive, and these are provided as well in the source code.

Primitive Override

All configuration parameters are also user-configurable. In addition, should a provided value be undesirable, any of the calculated parameters can be overridden with the desired settings.

Summary

The Clocking Wizard provides a summary for the created network. Input and output clock settings are provided both visually and as constraint files. In addition, jitter numbers for the created network are provided along with a resource estimate. Lastly, the wizard provides the input setting for PLL and MMCM based designs for Xilinx Power Estimator (XPE) in an easy-to-use table.

Design Environment

Figure 1 shows the design environment provided by the wizard to assist in integrating the generated clocking network into a design. The wizard provides a synthesizable and downloadable example design to demonstrate how to use the network and allows you to place a very simple clocking network in your device. A simulation demonstration test bench is also provided, which simulates the example design and illustrates output clock waveforms with respect to input clock waveforms.
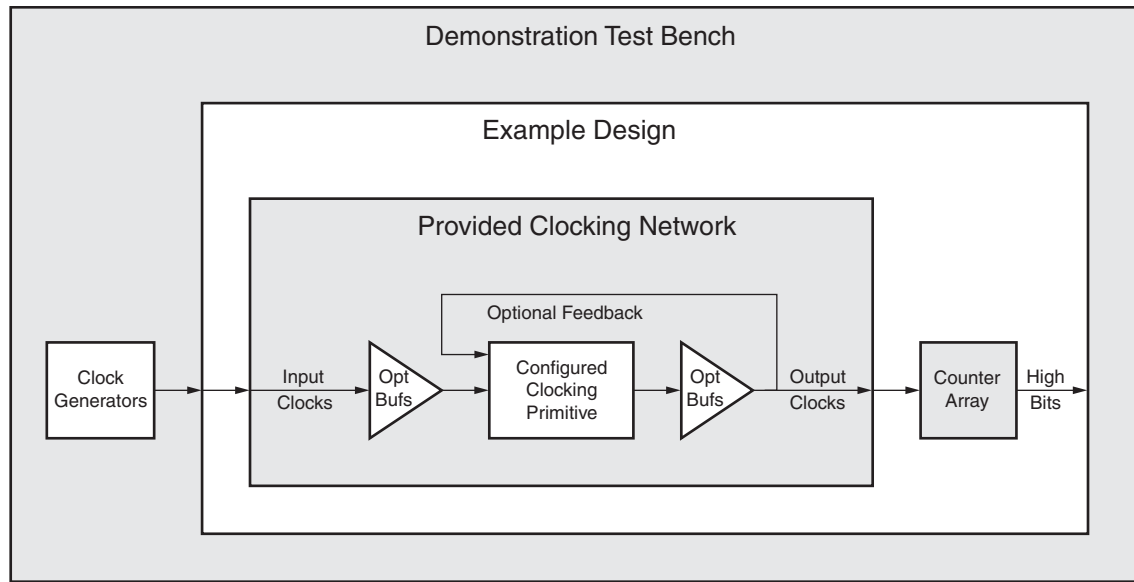

Figure 1: **Clocking Network and Support Modules Provided to the User**

I/O Signals

Table 1 describes the input and output ports provided from the clocking network. All ports are optional, with the exception of at least one input and one output clock are required. Availability of ports is controlled by user-selected parameters. For example, when Dynamic Reconfiguration is selected, these ports are exposed to the user. Any port that is not exposed is appropriately tied off or connected to a signal labeled *unused* in the delivered source code. Not all ports are available for all devices or primitives; for example, Dynamic Reconfiguration is a feature only available in Virtex-6 FPGA Mixed-Mode Clock Manager (MMCM) or Spartan-6 FPGA DCM_CLKGEN primitives. For more information about specific clocking primitive features, see the user guide for the associated FPGA device.

Table 1: **Clocking Wizard IO**

Port	I/O	Description
Input Clock Ports[1]		
CLK_IN1	Input	**Clock in 1**: Single-ended primary input clock port. Available when single-ended primary clock source is selected.
CLK_IN1_P CLK_IN1_N	Input	**Clock in 1 Positive and Negative**: Differential primary input clock port pair. Available when a differential primary clock source is selected.
CLK_IN2	Input	**Clock in 2:** Single-ended secondary input clock port. Available when a single-ended secondary clock source is selected.
CLK_IN2_P CLK_IN2_N	Input	**Clock in 2 Positive and Negative:** Differential secondary input clock port pair. Available when a differential secondary clock source is selected.
CLK_IN_SEL	Input	**Clock in Select:** When '1', selects the primary input clock; When '0', the secondary input clock is selected. Available when two input clocks are specified.
CLKFB_IN	Input	**Clock Feedback in**: Single-ended feedback in port of the clocking primitive. Available when user-controlled on-chip, user controller-off chip, or automatic control off-chip feedback option is selected.
CLKFB_IN_P	Input	**Clock Feedback in: Positive and Negative:** Differential feedback in port of the clocking primitive. Available when the automatic control off-chip feedback and differential feedback option is selected.
CLKFB_IN_N	Input	
Output Clock Ports		
CLK_OUT1	Output	**Clock Out 1:** Output clock of the clocking network. CLK_OUT1 is not optional.
CLK_OUT2-n	Output	**Clock Out 2 - n:** Optional output clocks of the clocking network that are user-specified. For an MMCM, up to seven are available. For a PLL or DCM, up to six are available. For a DCM_CLKGEN, up to three are available.
CLKFB_OUT	Output	**Clock Feedback Out:** Single ended feedback port of the clocking primitive. Available when the user-controlled feedback or automatic control off chip with single ended feedback option is selected.
CLKFB_OUT_P	Output	**Clock Feedback Out: Positive and Negative:** Differential feedback output port of the clocking primitive. Available when the user-controlled off-chip feedback and differential feedback option is selected.
CLKFB_OUT_N	Output	

Table 1: **Clocking Wizard IO** *(Cont'd)*

Port	I/O	Description
Dynamic Reconfiguration Ports for MMCM		
DADDR[6:0]	Input	**Dynamic Reconfiguration Address:** Address port for use in dynamic reconfiguration; active when DEN is asserted
DCLK	Input	**Dynamic Reconfiguration Clock:** Clock port for use in dynamic reconfiguration
DEN	Input	**Dynamic Reconfiguration Enable:** Starts a dynamic reconfiguration transaction
DI[15:0]	Input	**Dynamic Reconfiguration Data in:** Input data for a dynamic reconfiguration write transaction; active when DEN is asserted
DO[15:0]	Output	**Dynamic Reconfiguration Data Out:** Output data for a dynamic reconfiguration read transaction; active when DRDY is asserted
DRDY	Output	**Dynamic Reconfiguration Ready:** Completes a dynamic reconfiguration transaction
DWE	Input	**Dynamic Reconfiguration Write Enable:** When asserted, indicates that the dynamic reconfiguration transaction is a write; active when DEN is asserted
Dynamic Reconfiguration Ports for DCM_CLKGEN[2]		
PROGCLK	Input	**Program Clock**: Clock port for use in dynamic reconfiguration
PROGEN	Input	**Program Enable**: Starts a dynamic reconfiguration transaction
PROGDATA	Input	**Program Data**: Serial data stream to reprogram primitive settings
PROGDONE	Output	**Program Done**: When asserted, indicates that the reconfiguration transaction is complete
Dynamic Phase Shift Ports[3]		
PSCLK	Input	**Dynamic Phase Shift Clock:** Clock for use in dynamic phase shifting
PSEN	Input	**Dynamic Phase Shift Enable:** Starts a dynamic phase shift transaction
PSINCDEC	Input	**Dynamic Phase Shift increment/decrement:** When '1'; increments the phase shift of the output clock, when '0', decrements the phase shift
PSDONE	Output	**Dynamic Phase Shift Done:** Completes a dynamic phase shift transaction
Status and Control Ports[4]		
RESET	Input	**Reset:** When asserted, asynchronously clears the internal state of the primitive, and causes the primitive to re-initiate the locking sequence when released
POWER_DOWN [5]	Input	**Power Down:** When asserted, places the clocking primitive into low power state, which stops the output clocks
STATUS[2:0][6]	Output	**Status:** Contains information about the state of the primitive. Please see the user guide for specific bit descriptions
FREEZE[7]	Input	**Freeze**: When asserted, prevents tap adjustment in the event that the input clock is lost
INPUT_CLK_ STOPPED[8]	Output	**Input Clock Stopped:** When asserted, indicates that the selected input clock is no longer toggling

Table 1: **Clocking Wizard IO** *(Cont'd)*

Port	I/O	Description
LOCKED	Output	**Locked:** When asserted, indicates that the output clocks are stable and usable by downstream circuitry
CLK_VALID[9]	Output	**Clock Output Valid:** The CLK_VALID output is a logical combination of the DCM status ports which give the best indication that the input clock has been locked to, the DCM is functioning properly, and the output clocks are valid. When asserted, indicates the output clocks are valid.

1. At least one input clock is required; any design will have at least a CLK_IN1 or a CLK_IN1_P/CLK_IN1_N port. A secondary input clock is supported for Virtex-6 FPGAs only.
2. Dynamic reconfiguration ports are available only for the Spartan-6 FPGA DCM_CLKGEN primitive.
3. Dynamic phase shift ports are available for Virtex-6 FPGA MMCM or Spartan-6 FPGA DCM primitives.
4. Exposure of every status and control port is individually selectable-
5. The power-down port is available for the Virtex-6 FPGA MMCM primitive.
6. The status port is available for the Spartan-6 FPGA DCM and DCM_CLKGEN primitives.
7. The freeze port is available for the Spartan-6 FPGA DCM_CLKGEN primitive.
8. The input clock stopped port is available for the Virtex-6 FPGA MMCM and Spartan-6 FPGA DCM and DCM_CLKGEN primitives.
9. clk_valid port is available for DCM and DCM_CLKGEN.

Support

Xilinx provides technical support for this LogiCORE IP product when used as described in the product documentation. Xilinx cannot guarantee timing, functionality, or support of product if implemented in devices that are not defined in the documentation, if customized beyond that allowed in the product documentation, or if changes are made to any section of the design labeled *DO NOT MODIFY*.

Ordering Information

The Clocking Wizard is provided under the [SignOnce IP Site License](#).

Revision History

Date	Version	Revision
12/17/08	1.1.1	Xilinx Confidential DRAFT. Approved for external release under NDA only.
04/24/09	1.1	Initial major release. Supports core version 1.1 and Xilinx tools 11.1. Supports Spartan-6 and Virtex-6 devices.
06/24/09	1.2	Updated to support core version 1.2 and Xilinx tools 11.2.
09/16/09	1.3	Updated to support core version 1.3 and Xilinx tools 11.3.
12/02/09	1.4	Updated to support core version 1.4 and Xilinx tools 11.4.

Notice of Disclaimer