Virtex-6 FPGA Clocking Resources

User Guide

UG362 (v2.1) May 7, 2012

XILINX®

Xilinx is disclosing this user guide, manual, release note, and/or specification (the "Documentation") to you solely for use in the development of designs to operate with Xilinx hardware devices. You may not reproduce, distribute, republish, download, display, post, or transmit the Documentation in any form or by any means including, but not limited to, electronic, mechanical, photocopying, recording, or otherwise, without the prior written consent of Xilinx. Xilinx expressly disclaims any liability arising out of your use of the Documentation. Xilinx reserves the right, at its sole discretion, to change the Documentation without notice at any time. Xilinx assumes no obligation to correct any errors contained in the Documentation, or to advise you of any corrections or updates. Xilinx expressly disclaims any liability in connection with technical support or assistance that may be provided to you in connection with the Information.

THE DOCUMENTATION IS DISCLOSED TO YOU “AS-IS” WITH NO WARRANTY OF ANY KIND. XILINX MAKES NO OTHER WARRANTIES, WHETHER EXPRESS, IMPLIED, OR STATUTORY, REGARDING THE DOCUMENTATION, INCLUDING ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT OF THIRD-PARTY RIGHTS. IN NO EVENT WILL XILINX BE LIABLE FOR ANY CONSEQUENTIAL, INDIRECT, EXEMPLARY, SPECIAL, OR INCIDENTAL DAMAGES, INCLUDING ANY LOSS OF DATA OR LOST PROFITS, ARISING FROM YOUR USE OF THE DOCUMENTATION.

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
06/24/09	1.0	Initial Xilinx release.
09/16/09	1.1	Updated the About This Guide section in the Preface. Made clarifying edits in various discussion in Chapter 2 including updating the description of CLKOUT4_CASCADE. Added Virtex-6 HXT devices to Table 1-5, page 25. Added note 1 to Table 2-3, page 47. Revised the COMPENSATION attribute description and added Note 1 to Table 2-4.
01/19/10	1.2	Updated I/O Clock Buffer - BUFIO. Changed Regional Clock Buffer - BUFR. Clarified Horizontal Clock Buffer - BUFH. Removed CLKFBOUT fractional divide (fractional M counter) capability from Chapter 2, Mixed-Mode Clock Manager. These changes include updating Figure 2-2, revising the allowed values for CLKFBOUT_MULT_F in Table 2-2, and revising the description for CLKFBOUT_USE_FINE_PS. Clarified Equation 2-6 and Equation 2-7. Updated CLKINSEL – Clock Input Select. Updated Zero Delay Buffer, page 58 description.
03/15/10	1.3	Updated the Global Clock Buffers section. Updated the setup/hold requirements for S0 and S1 on page 21. A third paragraph about calibration circuits was added to the Introduction of Chapter 2. Updated Clock Network Deskew. Changed the VCO example in Interpolated Fine Phase Shift in Fixed or Dynamic Mode. In Table 2-4, updated the allowed values for CLKFBOUT_MULT_F and changed any type listed as String to Boolean to match the software models. Updated Dynamic Reconfiguration Port.
04/07/10	1.4	Updated the STARTUP_WAIT attribute allowed value on page 37 and Table 2-4.
08/16/10	1.5	Updated CE descriptions in Table 1-7 and Table 1-9. Clarified adjacent bank connections in Figure 1-22.
01/17/11	1.6	Updated Global Clock Buffers section with information on cascading BUFGs. Updated waveform in Figure 1-12. In Table 2-4, added Note 1 to DIVCLK_DIVIDE(1). Also, in Table 2-4, corrected allowed values for CLKIN1_PERIOD and CLKIN2_PERIOD. The Reference Clock Switching section now includes the need to force a RESET after clock switchover. Updated the Maximum Phase Shift equation on page 43. Updated the GTX/GTH transceiver MMCM discussion on page 53.
07/11/11	2.0	Revised the Frequency Synthesis Only Using Integer Divide section including Figure 2-4. Updated discussion in Reference Clock Switching, page 56. Updated the examples after Equation 2-9. Added Appendix A, Summary of Clocking Connectivity.
05/07/12	2.1	Updated I/O Clock Buffer - BUFIO. Updated Dynamic Phase Shift Interface. Added Note on page page 46. Updated allowed values for CLKOUT[0]_DIVIDE_F in Table 2-4. Revised the MRCC description in Table A-1.

Table of Contents

Appendix A: Summary of Clocking Connectivity

About This Guide

This guide serves as a technical reference describing the Virtex®-6 FPGA clocking resources.

Guide Contents

This manual contains the following chapters:

- Chapter 1, Clocking Resources
- Chapter 2, Mixed-Mode Clock Manager

Additional Documentation

The following documents are also available for download at http://www.xilinx.com/support/documentation/virtex-6.htm.

- Virtex-6 Family Overview

  The features and product selection of the Virtex-6 family are outlined in this overview.
- Virtex-6 FPGA Data Sheet: DC and Switching Characteristics

  This data sheet contains the DC and Switching Characteristic specifications for the Virtex-6 family.
- Virtex-6 FPGA Packaging and Pinout Specifications

  This specification includes the tables for device/package combinations and maximum I/Os, pin definitions, pinout tables, pinout diagrams, mechanical drawings, and thermal specifications.
- Virtex-6 FPGA Configuration Guide

  This all-encompassing configuration guide includes chapters on configuration interfaces (serial and SelectMAP), bitstream encryption, boundary-scan and JTAG configuration, reconfiguration techniques, and readback through the SelectMAP and JTAG interfaces.
- Virtex-6 FPGA SelectIO Resources User Guide

  This guide describes the SelectIO™ resources available in all Virtex-6 devices.
- Virtex-6 FPGA Configurable Logic Blocks User Guide

  This guide describes the capabilities of the configurable logic blocks (CLBs) available in all Virtex-6 devices.
- Virtex-6 FPGA Memory Resources User Guide

  The functionality of the block RAM and FIFO are described in this user guide.

- Virtex-6 FPGA GTH Transceivers User Guide

  This guide describes the GTH transceivers available in all Virtex-6 HXT FPGAs except the XC6VHX250T and the XC6VHX380T in the FF1154 package.

- Virtex-6 FPGA GTX Transceivers User Guide

  This guide describes the GTX transceivers available in all Virtex-6 FPGAs except the XC6VLX760.

- Virtex-6 FPGA Embedded Tri-Mode Ethernet MAC User Guide

  This guide describes the dedicated Tri-Mode Ethernet Media Access Controller available in all Virtex-6 FPGAs except the XC6VLX760.

- Virtex-6 FPGA DSP48E1 Slice User Guide

  This guide describes the architecture of the DSP48E1 slice in Virtex-6 FPGAs and provides configuration examples.

- Virtex-6 FPGA System Monitor User Guide

  The System Monitor functionality available in all Virtex-6 devices is outlined in this guide.

- Virtex-6 FPGA PCB Design Guide

  This guide provides information on PCB design for Virtex-6 devices, with a focus on strategies for making design decisions at the PCB and interface level.

Additional Support Resources

To search the database of silicon and software questions and answers or to create a technical support case in WebCase, see the Xilinx website at:

> http://www.xilinx.com/support.

Chapter 1

Clocking Resources

Global, Regional and I/O Clocks

For clocking purposes, each Virtex-6 device is divided into regions. The number of regions varies with device size, six regions in the smallest device to 18 regions in the largest one. A region is 40 CLBs high with a horizontal clock row in its center (HROW). Global I/O and regional clocking resources manage complex and simple clocking requirements. Non-clock resources, such as local routing, are not recommended when performing clock functions.

Global Clocks

Each Virtex-6 device has 32 global clock lines that can clock all sequential resources on the whole device (CLB, block RAM, DSPs, and I/O). Any 12 of these 32 global clock lines can be used in any region. Global clock lines are only driven by a global clock buffer, which can also be used as a clock enable circuit, or a glitch-free multiplexer. It can select between two clock sources, and can also switch away from a failed clock source.

A global clock buffer is often driven by a Clock Management Tile (CMT) to eliminate the clock distribution delay, or to adjust its delay relative to another clock. There are more global clocks than CMTs, but a CMT often drives more than one global clock.

Regional Clocks and I/O Clocks

Each region has up to eight differential regional clock buffers and six regional clock trees. A Virtex-6 FPGA I/O bank spans exactly one region. Each bank contains four clock-capable clock inputs. Each of these inputs can differentially or single-endedly drive four I/O clocks, four regional clocks, and one CMT in the same bank or region. Two of the four I/O clocks can drive into the bank above and below while the remaining two can only drive the local bank. In addition, regional clocks can drive regional clock trees in the adjacent regions. When the clock-capable I/Os are driven by single-ended clocks, then the clock must be connected to the positive (P) side of the differential “clock capable” pin pair. The negative (N) side can be used as a general purpose I/O or left unconnected.

The regional clock buffer can be programmed to divide the incoming clock rate by any integer number from 1 to 8. This feature, in conjunction with the programmable serializer/deserializer in the IOB, (see Chapter 3 in the *Virtex-6 FPGA SelectIO Resources User Guide*), allows source-synchronous systems to cross clock domains without using additional logic resources.

Clocking Architecture

Each Virtex-6 device has a center column containing the dedicated configuration pins. Free regions above and below are filled with CLBs (logic only). There is a CMT column adjacent to the right of the center column with one CMT per region. A CMT has two Mixed-Mode Clock Managers (MMCMs). See Chapter 2, Mixed-Mode Clock Manager. The CMT column also contains the 32 vertical spines of the global clock trees. In the horizontal direction, Virtex-6 FPGAs are organized by regions each 40 CLBs and one bank high. There is a horizontal clock row (HROW) in the center of each region containing the horizontal clock spines (12), six regional clock tracks (BUFR) and the horizontal clocks (up to 12 BUFH). BUFHs use the same resources as the horizontal clock spines. A new type of horizontal clock tree, the high-performance clock is introduced in this architecture providing a low jitter clock path from the MMCMs to the I/O. See the *Virtex-6 FPGA SelectIO User Guide* for more detail.

Every Virtex-6 FPGA has two I/O columns to the left and right of the center column labeled I/O center left (IOCL) and I/O center right (IOCR) with CLBs in between. Every LX, LXT, and SXT device has an I/O outer column at the left edge of the device (IOOL) and some devices have an outer edge I/O column to the right. Other devices have a Gigabit Transceiver (GT) column to the right instead. There is a horizontal clock row (HROW) running in the center of each region/bank. The HROW contains the vertical global clock spines of the global clock buffers (BUFG) and the BUFHs if the vertical global clock spines are used as such. The inner I/O columns contain eight global clock pin pairs (GCs) spread over four banks for maximum flexibility in I/O standards. All I/O columns contain four clock-capable pin pairs (CCs) which can connect to BUFIO and BUFR. Two of the four CCs per bank can connect to BUFIOs spanning the adjacent regions. Additionally, the BUFRs and CC pins in the center columns can directly drive MMCMs in the same region and indirectly BUFGs through the vertical global clock spins that drive the BUFGs. Figure 1-1 shows an example of the high-level banking and global-clocking architecture. Figure 1-2 shows a more detailed view of the clocking in a single region with two inner column I/O banks.

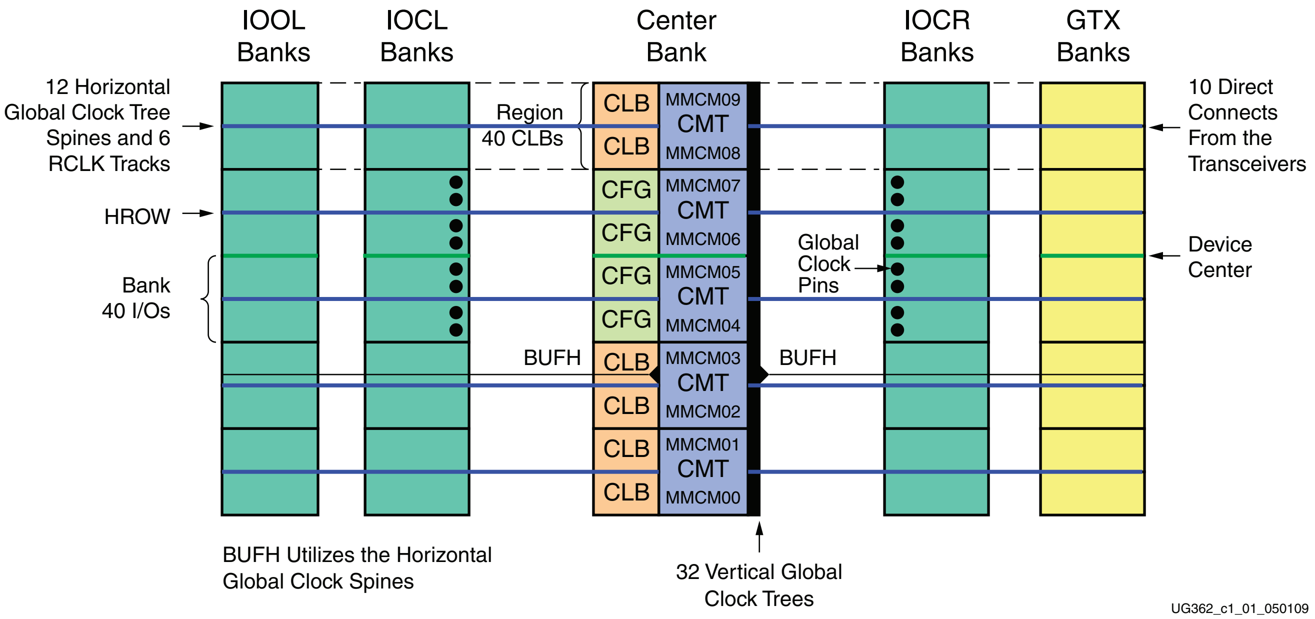


Figure 1-1: **Example of Block Level Banking and Global Clocking Architecture**

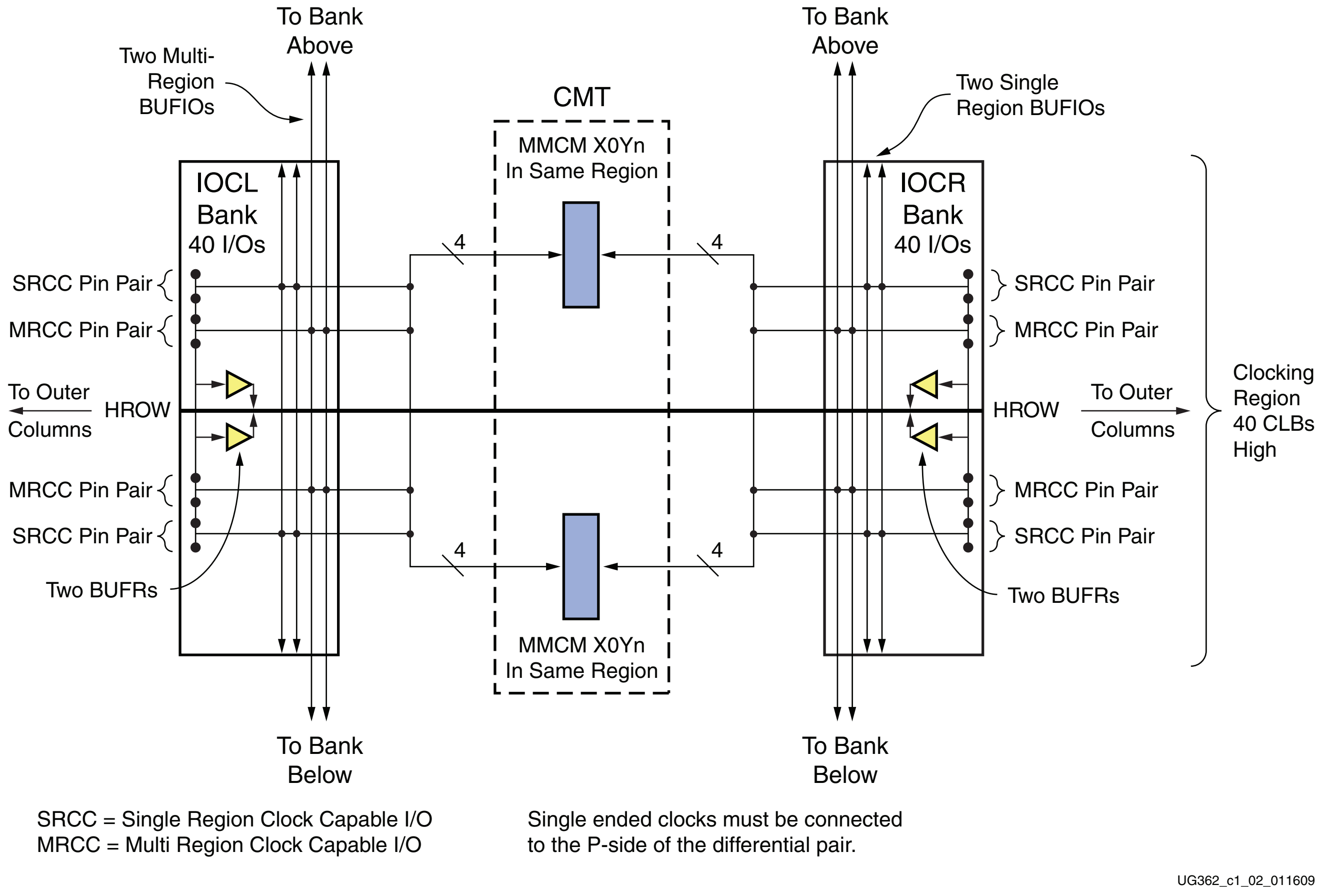


Figure 1-2: **Inner I/O Column Single Region Clocking Structure**

For more information on clock input pins, consult the *Die Level Bank Numbering and Clock Pins Overview* section in UG365: *Virtex-6 FPGA Packaging and Pinout Specification*.

Global Clocking Resources

Global clocks are a dedicated network of interconnect specifically designed to reach all clock inputs to the various resources in an FPGA. These networks are designed to have low skew and low duty cycle distortion, low power, and improved jitter tolerance. They are also designed to support very high frequency signals.

Understanding the signal path for a global clock expands the understanding of the various global clock resources. The global clocking resources and network consist of the following paths and components:

- Global Clock Inputs
- Global Clock Buffers
- Clock Tree and Nets - GCLK
- Clock Regions

Global Clock Inputs

Virtex-6 FPGAs contain specialized global clock input locations for use as regular user I/Os if not used as clock inputs. There are eight global clock inputs per device. Clock inputs can be configured for any I/O standard, including differential I/O standards. Each clock input can be either single-ended or differential. All eight clock inputs can be differential if desired. When used as outputs, global clock input pins can be configured for any output standard. Each global clock input pin supports any single-ended output standard or any output differential standard. The global clock inputs are distributed across four banks in the inner I/O columns for the most flexible selection of I/O standards.

Global Clock Input Buffer Primitives

The primitives in Table 1-1 are different configurations of the input clock I/O input buffer.

Table 1-1: **Clock Buffer Primitives**

Primitive	Input	Output	Description
IBUFG	I	O	Input clock buffer for single-ended I/O
IBUFGDS	I, IB	O	Input clock buffer for differential I/O

These two primitives work in conjunction with the Virtex-6 FPGA SelectIO™ resource by setting the IOSTANDARD attribute to the desired standard. Refer to Chapter 1 of the *Virtex-6 FPGA SelectIO Resources User Guide* for a complete list of possible I/O standards.

Clock Gating for Power Savings

The Virtex-6 FPGA clock architecture provides a straightforward means of implementing clock gating for the purposes of powering down portions of a design. Most designs contain several unused BUFGCE resources. A clock can drive a BUFGCE input, and a BUFGCE output can drive distinct regions of logic. For example, if all the logic that is required to always be operating is constrained to a few clocking regions, then the BUFGCE output can drive those regions. Toggling the enable of the BUFGCE provides a simple means of stopping all dynamic power consumption in the logic regions available for power savings.

The Xilinx Power Estimator (XPE) or the Xilinx Power Analyzer (XPower) tools are used to estimate power savings. The difference is calculated by setting the frequency on the corresponding clock net to 0 MHz or providing the appropriate stimulus data to the tool.

Global Clock Buffers

There are 32 global clock buffers in every Virtex-6 device. A global clock input can directly connect from the P-side of the differential input pin pair to any global clock buffer input. There are eight global clock pin inputs. The top/bottom half rules from previous Virtex architectures no longer apply. Each differential global clock pin pair can connect to either a differential or single-ended clock on the PCB. If using a single-ended clock, then the P-side of the pin pair must be used because a direct connection only exists on this pin. For pin naming conventions please refer to the *Virtex-6 FPGA Packaging and Pinout Specification*. If a single-ended clock is connected to the P-side of a differential pin pair, then the N-side can not be used as another single-ended clock pin. However, it can be used as a user I/O.

MMCMs in the top half of the device can only drive the BUFGs in the top half of the device and MMCMs in bottom half can only drive BUFGs in the bottom half. Similarly, only BUFGs in the same half of the device can be used as feedback to the MMCMs in the same half of the device.

Global clock buffers allow various clock/signal sources to access the global clock trees and nets. The possible sources for input to the global clock buffers include:

- Global clock inputs
- Clock-capable inputs in the same region of the inner I/O columns.
- Clock Management Tile (CMT) consisting of mixed-mode clock managers (two MMCMs per CMT) driving BUFGs in the same half of the device.
- Other global clock buffer outputs (BUFGs)
- General interconnect
- Regional clock buffers (BUFRs)
- Gigabit transceivers

The Virtex-6 FPGA clock-capable inputs can drive global clock buffers indirectly through the vertical clock network that exists in the MMCM column. The 32 BUFGs are organized into two groups of 16 BUFGs in the top and bottom of the device. Any resources (e.g., GTX transceivers) connecting to the BUFGs directly have a top/bottom limitation. For example, each MMCM in the top can only drive the 16 BUFGs residing in that top of the device. Similarly, the MMCMs in the bottom drive the 16 BUFGs in the bottom.

All global clock buffers can drive all clock regions in Virtex-6 devices. However, only 12 different clocks can be driven in a single clock region. A clock region (40 CLBs) is a branch of the clock tree consisting of 20 CLB rows up and 20 CLB rows down. A clock region only spans halfway across the device.

The clock buffers are designed to be configured as a synchronous or asynchronous glitch-free 2:1 multiplexer with two clock inputs. Virtex-6 device control pins provide a wide range of functionality and robust input switching. In the Virtex-6 clocking architecture, BUFGCNTRL multiplexers and all derivatives can be cascaded to adjacent clock buffers within the group of 16 in the upper/lower half of the device, effectively creating a ring of 16 BUFGMUXes (BUFGCNTRL multiplexers) in the upper half and another ring of 16 in the lower half. Figure 1-3 shows a simplified diagram of cascading BUFGs.

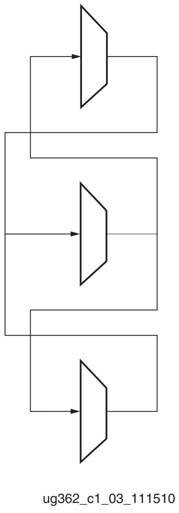


Figure 1-3: **Cascading BUFGs**

The following subsections detail the various configurations, primitives, and use models of the Virtex-6 FPGA clock buffers.

Global Clock Buffer Primitives

The primitives in Table 1-2 are different configurations of the global clock buffers.

Table 1-2: **Global Clock Buffer Primitives**

Primitive	Input	Output	Control
BUFGCTRL	I0, I1	O	CE0, CE1, IGNORE0, IGNORE1, S0, S1
BUFG	I	O	–
BUFGCE	I	O	CE
BUFGCE_1	I	O	CE
BUFGMUX	I0, I1	O	S
BUFGMUX_1	I0, I1	O	S
BUFGMUX_CTRL	I0, I1	O	S

Notes:

1. All primitives are derived from a software preset of BUFGCTRL.

BUFGCTRL

The BUFGCTRL primitive shown in Figure 1-4, can switch between two asynchronous clocks. All other global clock buffer primitives are derived from certain configurations of BUFGCTRL. The ISE software tools manage the configuration of all these primitives.

BUFGCTRL has four select lines, S0, S1, CE0, and CE1. It also has two additional control lines, IGNORE0 and IGNORE1. These six control lines are used to control the input I0 and I1.

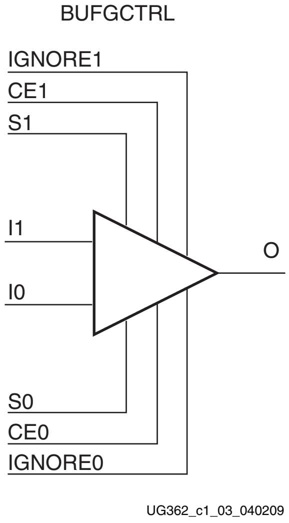


Figure 1-4: **BUFGCTRL Primitive**

BUFGCTRL is designed to switch between two clock inputs without the possibility of a glitch. When the presently selected clock transitions from High to Low after S0 and S1 change, the output is kept Low until the other (to-be-selected) clock has transitioned from High to Low. Then the new clock starts driving the output.The default configuration for BUFGCTRL is falling edge sensitive and held at Low prior to the input switching. BUFGCTRL can also be rising edge sensitive and held at High prior to the input switching by using the INIT_OUT attribute.

In some applications the conditions previously described are not desirable. Asserting the IGNORE pins will bypass the BUFGCTRL from detecting the conditions for switching between two clock inputs. In other words, asserting IGNORE causes the MUX to switch the inputs at the instant the select pin changes. IGNORE0 causes the output to switch away from the I0 input immediately when the select pin changes, while IGNORE1 causes the output to switch away from the I1 input immediately when the select pin changes.

Selection of an input clock requires a “select” pair (S0 and CE0, or S1 and CE1) to be asserted High. If either S or CE is not asserted High, the desired input will not be selected. In normal operation, both S and CE pairs (all four select lines) are not expected to be asserted High simultaneously. Typically only one pin of a “select” pair is used as a select line, while the other pin is tied High. The truth table is shown in Table 1-3.

Table 1-3: **Truth Table for Clock Resources**

CE0	S0	CE1	S1	O
1	1	0	X	I0
1	1	X	0	I0
0	X	1	1	I1
X	0	1	1	I1
1	1	1	1	Old Input [(1)]

Notes:

1. Old input refers to the valid input clock before this state is achieved.
2. For all other states, the output becomes the value of INIT_OUT and does not toggle.

Although both S and CE are used to select a desired output, each one of these pins behaves slightly different. When using CE to switch clocks, the change in clock selection can be faster than when using S. Violation in Setup/Hold time of the CE pins causes a glitch at the clock output. On the other hand, using the S pins allows the user to switch between the two clock inputs without regard to Setup/Hold times. It will not result in a glitch. See BUFGMUX_CTRL. The CE pin is designed to allow backward compatibility from previous Virtex architectures.

The timing diagram in Figure 1-5 illustrates various clock switching conditions using the BUFGCTRL primitives. Exact timing numbers are best found using the speed specification.

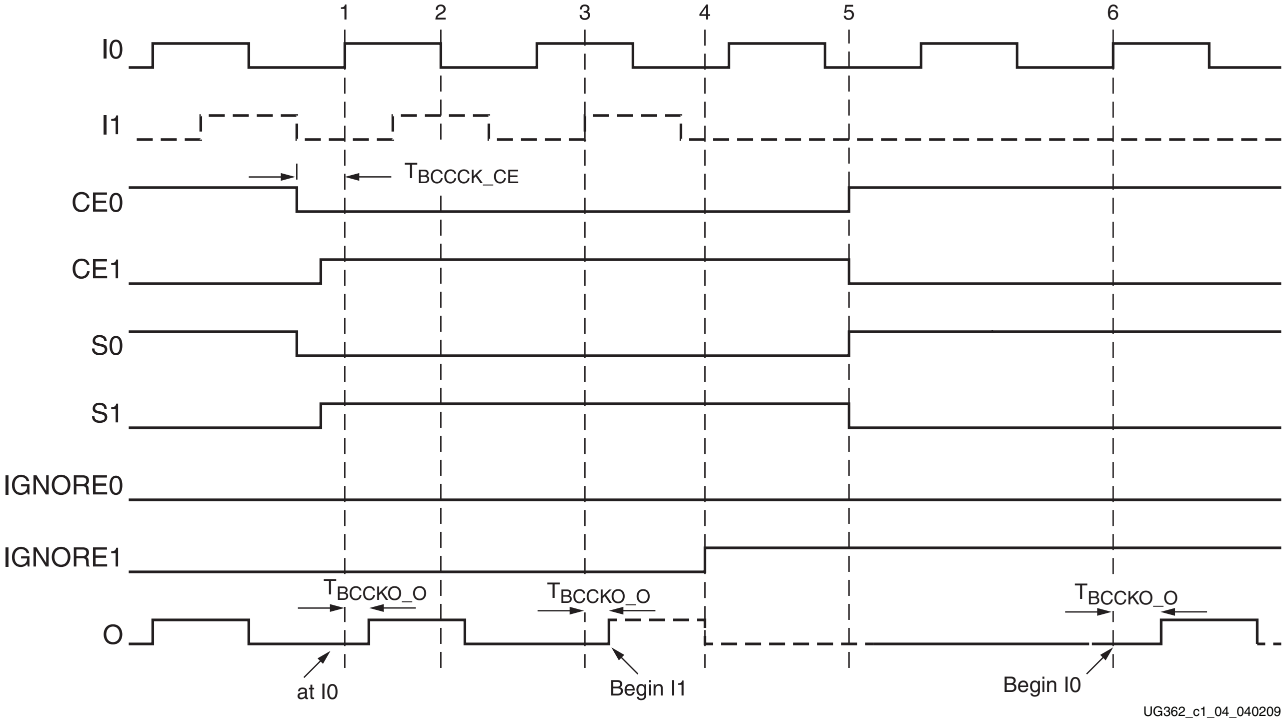


Figure 1-5: **BUFGCTRL Timing Diagram**

- Before time event 1, output O uses input I0.
- At time T_{BCCCK_CE}, before the rising edge at time event 1, both CE0 and S0 are deasserted Low. At about the same time, both CE1 and S1 are asserted High.
- At time T_{BCCKO_O}, after time event 3, output O uses input I1. This occurs after a High to Low transition of I0 (event 2) followed by a High to Low transition of I1.
- At time event 4, IGNORE1 is asserted.
- At time event 5, CE0 and S0 are asserted High while CE1 and S1 are deasserted Low. At T_{BCCKO_O}, after time event 6, output O has switched from I1 to I0 without requiring a High to Low transition of I1.

Other capabilities of BUFGCTRL are:

- Pre-selection of the I0 and I1 inputs are made after configuration but before device operation.
- The initial output after configuration can be selected as either High or Low.
- Clock selection using CE0 and CE1 only (S0 and S1 tied High) can change the clock selection without waiting for a High to Low transition on the previously selected clock.

Table 1-4 summarizes the attributes for the BUFGCTRL primitive.

Table 1-4: **BUFGCTRL Attributes**

Attribute Name	Description	Possible Values
INIT_OUT	Initializes the BUFGCTRL output to the specified value after configuration. Sets the positive or negative edge behavior. Sets the output level when changing clock selection.	0 (default), 1
PRESELECT_I0	If TRUE, BUFGCTRL output will use the I0 input after configuration[1]	FALSE (default), TRUE
PRESELECT_I1	If TRUE, BUFGCTRL output will use the I1 input after configuration[1]	FALSE (default), TRUE

Notes:

1. Both PRESELECT attributes cannot be TRUE at the same time.
2. The LOC constraint is available.

BUFG

BUFG is simply a clock buffer with one clock input and one clock output. This primitive is based on BUFGCTRL with some pins connected to logic High or Low. Figure 1-6 illustrates the relationship of BUFG and BUFGCTRL. A LOC constraint is available for BUFG.

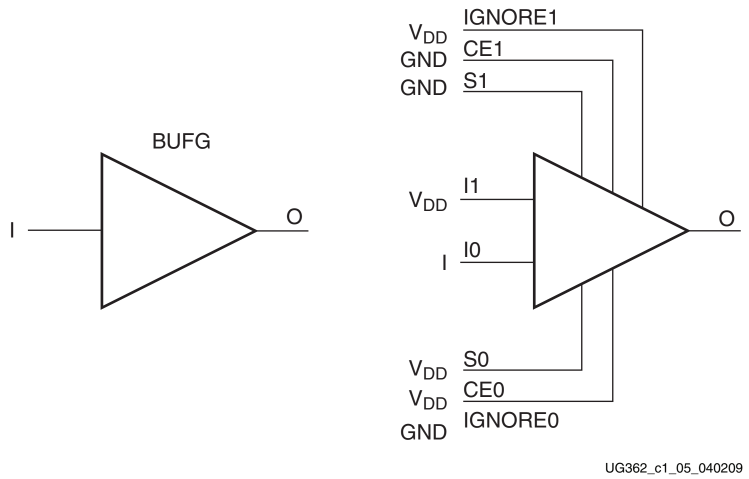


Figure 1-6: **BUFG as BUFGCTRL**

The output follows the input as shown in the timing diagram in Figure 1-7.

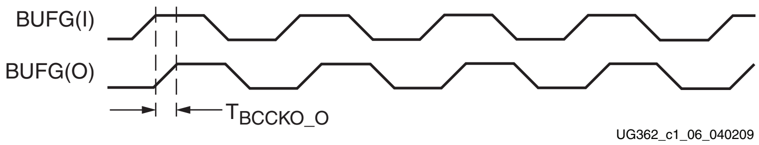


Figure 1-7: **BUFG Timing Diagram**

BUFGCE and BUFGCE_1

Unlike BUFG, BUFGCE is a clock buffer with one clock input, one clock output and a clock enable line. This primitive is based on BUFGCTRL with some pins connected to logic High or Low. Figure 1-8 illustrates the relationship of BUFGCE and BUFGCTRL. A LOC constraint is available for BUFGCE and BUFGCE_1.

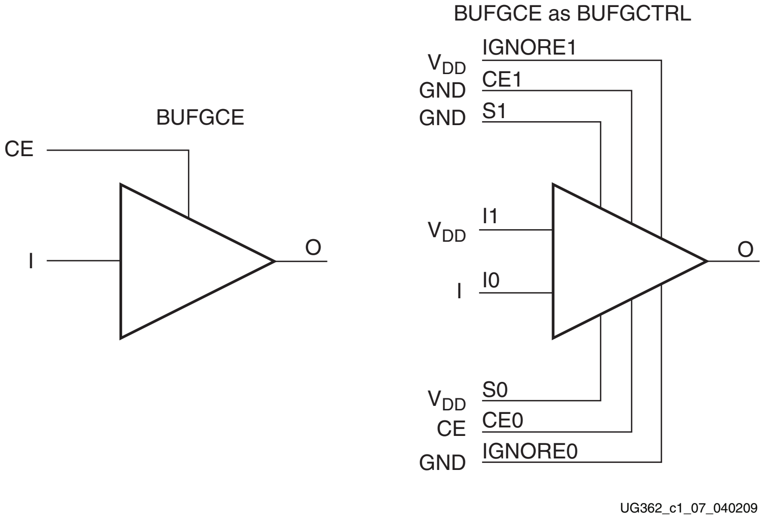


Figure 1-8: **BUFGCE as BUFGCTRL**

The switching condition for BUFGCE is similar to BUFGCTRL. If the CE input is Low prior to the incoming rising clock edge, the following clock pulse does not pass through the clock buffer, and the output stays Low. Any level change of CE during the incoming clock High pulse has no effect until the clock transitions Low. The output stays Low when the clock is disabled. However, when the clock is being disabled it completes the clock High pulse.

Since the clock enable line uses the CE pin of the BUFGCTRL, the select signal must meet the setup time requirement. Violating this setup time may result in a glitch. Figure 1-9 illustrates the timing diagram for BUFGCE.

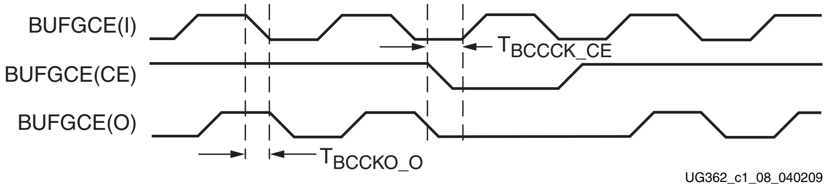


Figure 1-9: **BUFGCE Timing Diagram**

BUFGCE_1 is similar to BUFGCE, with the exception of its switching condition. If the CE input is Low prior to the incoming falling clock edge, the following clock pulse does not pass through the clock buffer, and the output stays High. Any level change of CE during the incoming clock Low pulse has no effect until the clock transitions High. The output stays High when the clock is disabled. However, when the clock is being disabled it completes the clock Low pulse.

Figure 1-10 illustrates the timing diagram for BUFGCE_1.

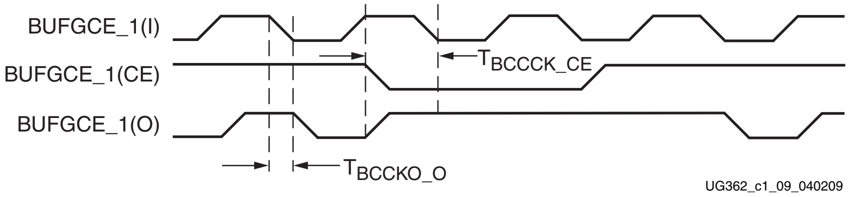


Figure 1-10: **BUFGCE_1 Timing Diagram**

BUFGMUX and BUFGMUX_1

BUFGMUX is a clock buffer with two clock inputs, one clock output, and a select line. This primitive is based on BUFGCTRL with some pins connected to logic High or Low. Figure 1-11 illustrates the relationship of BUFGMUX and BUFGCTRL. A LOC constraint is available for BUFGMUX and BUFGCTRL.

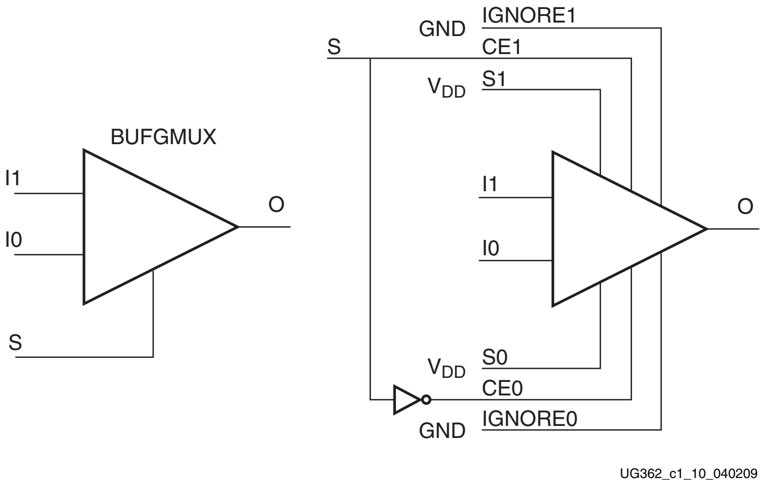


Figure 1-11: **BUFGMUX as BUFGCTRL**

Since the BUFGMUX uses the CE pins as select pins, when using the select, the setup time requirement must be met. Violating this setup time may result in a glitch.

Switching conditions for BUFGMUX are the same as the CE pins on BUFGCTRL. Figure 1-12 illustrates the timing diagram for BUFGMUX.

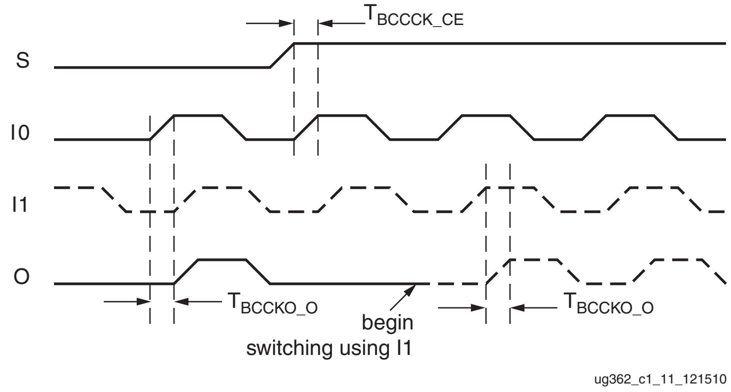


Figure 1-12: **BUFGMUX Timing Diagram**

In Figure 1-12:

- The current clock is I0.
- S is activated High.
- If I0 is currently High, the multiplexer waits for I0 to deassert Low.
- Once I0 is Low, the multiplexer output stays Low until I1 transitions High to Low.
- When I1 transitions from High to Low, the output switches to I1.
- If Setup/Hold are met, no glitches or short pulses can appear on the output.

BUFGMUX_1 is rising edge sensitive and held at High prior to input switch. Figure 1-13 illustrates the timing diagram for BUFGMUX_1. A LOC constraint is available for BUFGMUX and BUFGMUX_1.

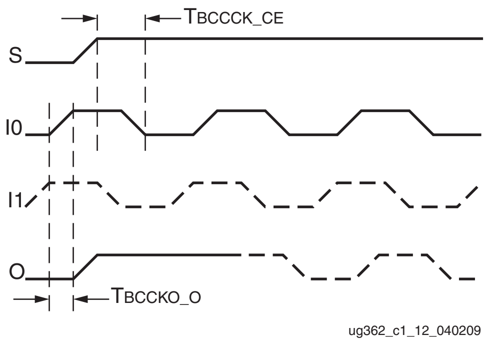


Figure 1-13: **BUFGMUX_1 Timing Diagram**

In Figure 1-13:

- The current clock is I0.
- S is activated High.
- If I0 is currently Low, the multiplexer waits for I0 to be asserted High.
- Once I0 is High, the multiplexer output stays High until I1 transitions Low to High.
- When I1 transitions from Low to High, the output switches to I1.
- If Setup/Hold are met, no glitches or short pulses can appear on the output.

BUFGMUX_CTRL

The BUFGMUX_CTRL replaces the BUFGMUX_VIRTEX4 legacy primitive. BUFGMUX_CTRL is a clock buffer with two clock inputs, one clock output, and a select line. This primitive is based on BUFGCTRL with some pins connected to logic High or Low. Figure 1-14 illustrates the relationship of BUFGMUX_CTRL and BUFGCTRL.

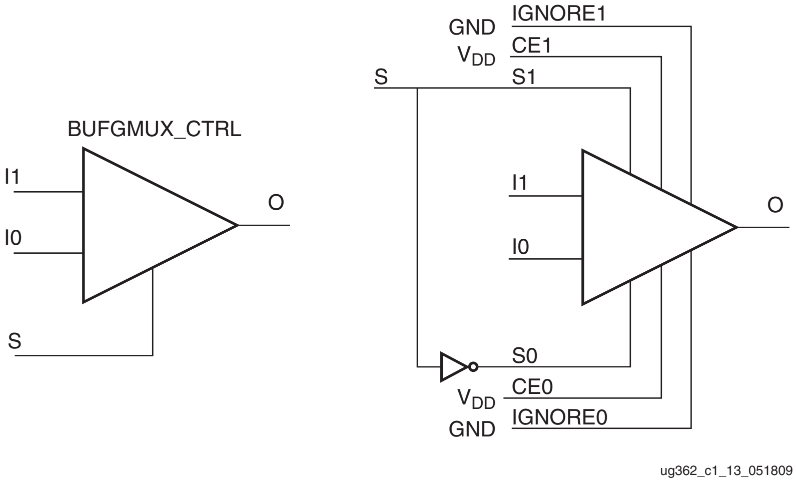


Figure 1-14: **BUFGMUX_CTRL as BUFGCTRL**

BUFGMUX_CTRL uses the S pins as select pins. S can switch anytime without causing a glitch. The Setup/Hold time on S is for determining whether the output will pass an extra pulse of the previously selected clock before switching to the new clock. If S changes as shown in Figure 1-15, prior to the setup time T_{BCCCK_S} and before I0 transitions from High to Low, then the output will not pass an extra pulse of I0. If S changes following the hold time for S, then the output will pass an extra pulse. If S violates the Setup/Hold requirements, the output might pass the extra pulse, but it will not glitch. In any case, the output will change to the new clock within three clock cycles of the slower clock.

The Setup/Hold requirements for S0 and S1 are with respect to the falling clock edge, not the rising edge as for CE0 and CE1.

Switching conditions for BUFGMUX_CTRL are the same as the S pin of BUFGCTRL. Figure 1-15 illustrates the timing diagram for BUFGMUX_CTRL.

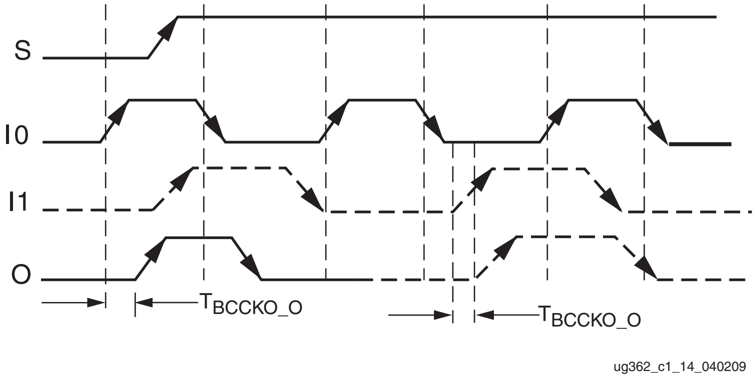


Figure 1-15: **BUFGMUX_CTRL Timing Diagram**

Other capabilities of the BUFGMUX_CTRL primitive are:

- Pre-selection of I0 and I1 input after configuration.
- Initial output can be selected as High or Low after configuration.

Additional Use Models

Asynchronous MUX Using BUFGCTRL

In some cases an application requires immediate switching between clock inputs or bypassing the edge sensitivity of BUFGCTRL. An example is when one of the clock inputs is no longer switching. If this happens, the clock output would not have the proper switching conditions because the BUFGCTRL never detected a clock edge. This case uses the asynchronous MUX. Figure 1-16 illustrates an asynchronous MUX with BUFGCTRL design example. Figure 1-17 shows the asynchronous MUX timing diagram.

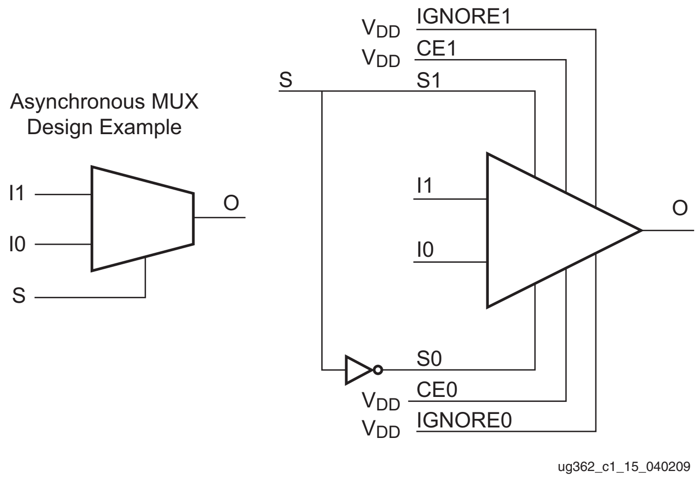


Figure 1-16: **Asynchronous MUX with BUFGCTRL Design Example**

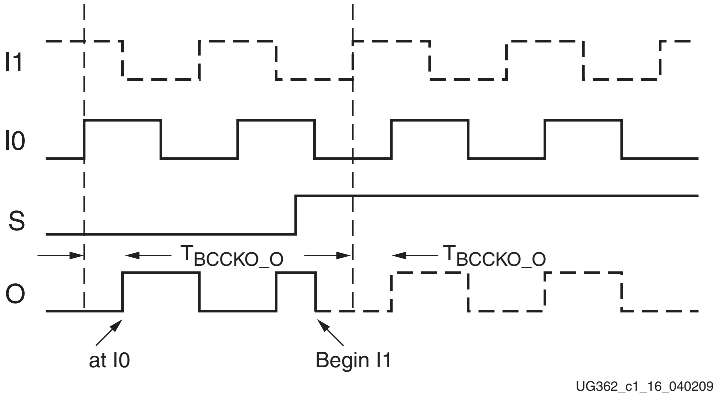


Figure 1-17: **Asynchronous MUX Timing Diagram**

In Figure 1-17:

- The current clock is from I0.
- S is activated High.
- The Clock output immediately switches to I1.
- When Ignore signals are asserted High, glitch protection is disabled.

BUFGMUX_CTRL with a Clock Enable

A BUFGMUX_CTRL with a clock enable BUFGCTRL configuration allows the user to choose between the incoming clock inputs. If needed, the clock enable is used to disable the output. Figure 1-18 illustrates the BUFGCTRL usage design example and Figure 1-19 shows the timing diagram.

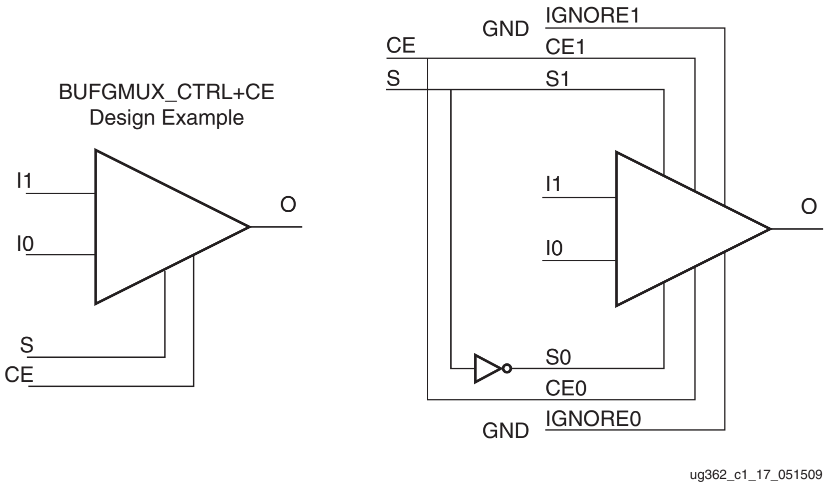


Figure 1-18: **BUFGMUX_CTRL with a CE and BUFGCTRL**

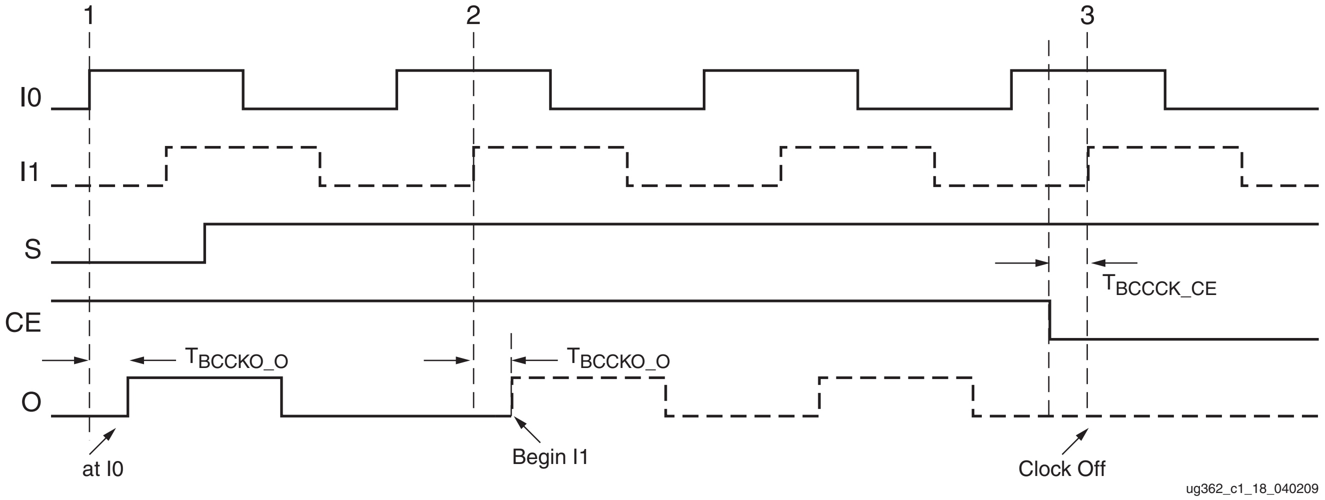


Figure 1-19: **BUFGMUX_CTRL with a CE Timing Diagram**

In Figure 1-19:

- At time event 1, output O uses input I0.
- Before time event 2, S is asserted High.
- At time T_{BCCKO_O}, after time event 2, output O uses input I1. This occurs after a High to Low transition of I0 followed by a High to Low transition of I1 is completed.
- At time T_{BCCCK_CE}, before time event 3, CE is asserted Low. The clock output is switched Low and kept at Low after a High to Low transition of I1 is completed.

Clock Tree and Nets - GCLK

Virtex-6 FPGA clock trees are designed for low-skew and low-power operation. Any unused branch is disconnected. The clock trees also manage the load/fanout when all the logic resources are used.

All global clock lines and buffers are implemented differentially. This facilitates much better duty cycles and common-mode noise rejection.

In the Virtex-6 architecture, the pin access of the global clock lines are not limited to the logic resources clock pins. The global clock lines can drive pins in the CLB other than CLK pins (for example: the control pins SR and CE). Applications requiring a very fast signal connection and large load/fanout benefit from this architecture.

Clock Regions

Virtex-6 devices improve the clocking distribution by the use of clock regions. Each clock region can have up to 12 global clock domains. These 12 global clocks can be driven by any combination of the 32 global clock buffers. The dimensions of a clock region are fixed to 40 CLBs tall (40 IOBs) and spanning half of the die (Figure 1-20). By fixing the dimensions of the clock region, larger Virtex-6 devices can have more clock regions. As a result, Virtex-6 devices can support many more multiple clock domains than previous FPGA architectures. Table 1-5 shows the number of clock regions in each Virtex-6 device. The CMT and global clocking resources are located to the right of the center column containing the configuration pins.

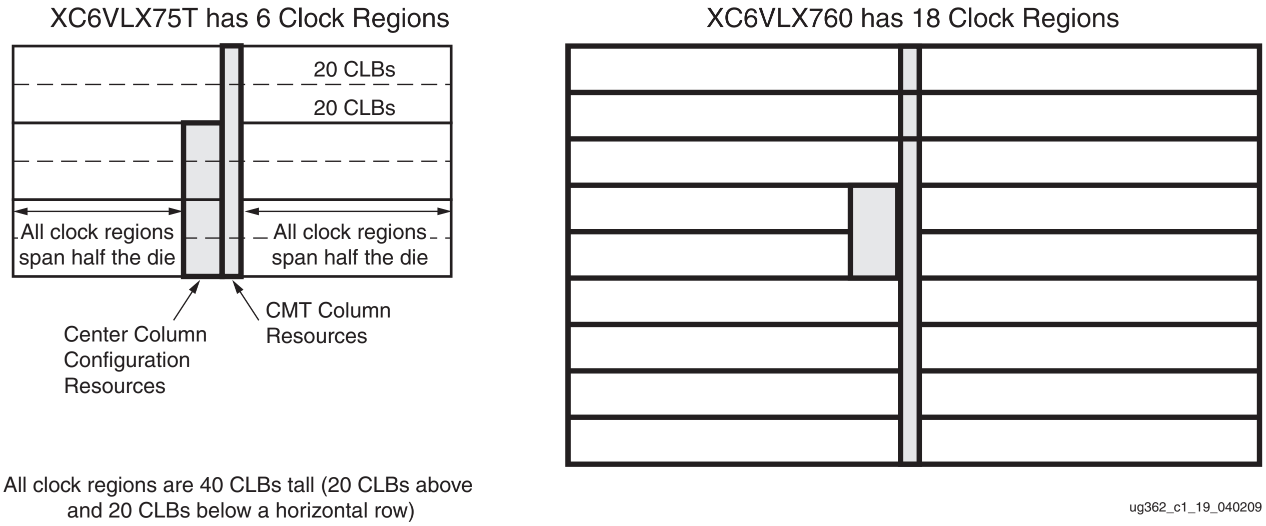

XC6VLX75T has 6 Clock Regions
20 CLBs
20 CLBs
All clock regions span half the die
All clock regions span half the die
Center Column Configuration Resources
CMT Column Resources
All clock regions are 40 CLBs tall (20 CLBs above and 20 CLBs below a horizontal row)
XC6VLX760 has 18 Clock Regions
ug362_c1_19_040209

Figure 1-20: **Clock Regions**

Table 1-5: **Virtex-6 FPGA Clock Regions**

Device	Number of Clock Regions
XC6VLX75T	6
XC6VLX130T	10
XC6VLX195T	10
XC6VLX240T	12
XC6VLX365T	12
XC6VLX550T	18
XC6VLX760	18
XC6VSX315T	12
XC6VSX475T	18
XC6VHX250T	12
XC6VHX255T	12
XC6VHX380T	18
XC6VHX565T	18

Regional Clocking Resources

Regional clock networks are a set of differential clock networks independent of the global clock network. Unlike global clocks, the span of a regional clock signal (BUFR) is limited to three clock regions, while two I/O clock signals drive a single region and an additional two I/O clocks can drive the regions/banks above and below. These networks are especially useful for source-synchronous interface designs. The I/O banks in Virtex-6 devices are the same size as a clock region.

To understand how regional clocking works, it is important to understand the signal path of a regional clock signal. The regional clocking resources and network in Virtex-6 devices consist of the following paths and components:

- Clock-Capable I/O
- I/O Clock Buffer - BUFIO
- Regional Clock Buffer - BUFR
- Regional Clock Nets
- Horizontal Clock Buffer - BUFH
- High-Performance Clocks

Clock-Capable I/O

Each clock region has four clock-capable I/O pin pairs per I/O bank in every I/O column. Clock-capable I/O pairs are regular I/O pairs in select locations with special hardware connections to nearby regional clock resources and other clock resources. There are four dedicated clock-capable I/O sites in every bank. When used as clock inputs, clock-capable pins can drive BUFIO and BUFR. Each I/O column supports regional clock buffers (BUFR). There are up to four I/O columns in each device. Two inner I/O columns are available in each device and support four BUFRs in each region. Depending on the device used, up to two outer I/O columns are available.

When used as single-ended clock pins, then as described in Global Clock Buffers the P-side of the pin pair must be used because a direct connection only exists on this pin.

In Virtex-6 devices, the inner I/O column clock-capable pins can also drive MMCM and BUFG clock inputs. This method of driving MMCM input clocks produces a higher performance path than connecting clocks to the MMCMs using the global clock pins. The clock-capable pins must be in the same region/bank and to either the left or right of where the MMCM is located.

I/O Clock Buffer - BUFIO

The I/O clock buffer (BUFIO) is a clock buffer available in Virtex-6 devices. The BUFIO drives a dedicated clock net within the I/O columns, independent of the global clock resources. Thus, BUFIOs are ideally suited for source-synchronous data capture (forwarded/receiver clock distribution). If directly driven by an I/O, then BUFIOs can only be driven by clock-capable I/Os located in the same bank. In a clock region, there are four BUFIOs per bank. Some devices have two banks per region (inner and outer columns). Each BUFIO can drive a single I/O clock network in the same region/bank and two of the four BUFIOs can drive the regions above and below. BUFIOs cannot drive logic resources (CLB, block RAM, DSP, etc.) because the I/O clock network only reaches the I/O column in the same bank/clock region or bank/clock region above and below.

BUFIO Primitive

BUFIO is simply a clock in, clock out buffer. There is a phase delay between input and output. Figure 1-21 shows the BUFIO. Table 1-6 lists the BUFIO ports. A location constraint is available for BUFIO.

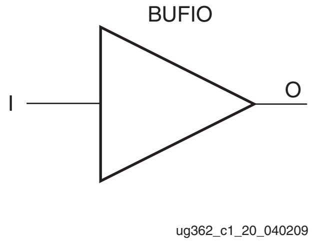


Figure 1-21: **BUFIO Primitive**

Table 1-6: **BUFIO Port List and Definitions**

Port Name	Type	Width	Definition
O	Output	1	Clock output port
I	Input	1	Clock input port

BUFIO Use Models

In Figure 1-22, a BUFIO is used to drive the I/O logic using the clock-capable I/O. This implementation is ideal in source-synchronous applications where a forwarded clock is used to capture incoming data.

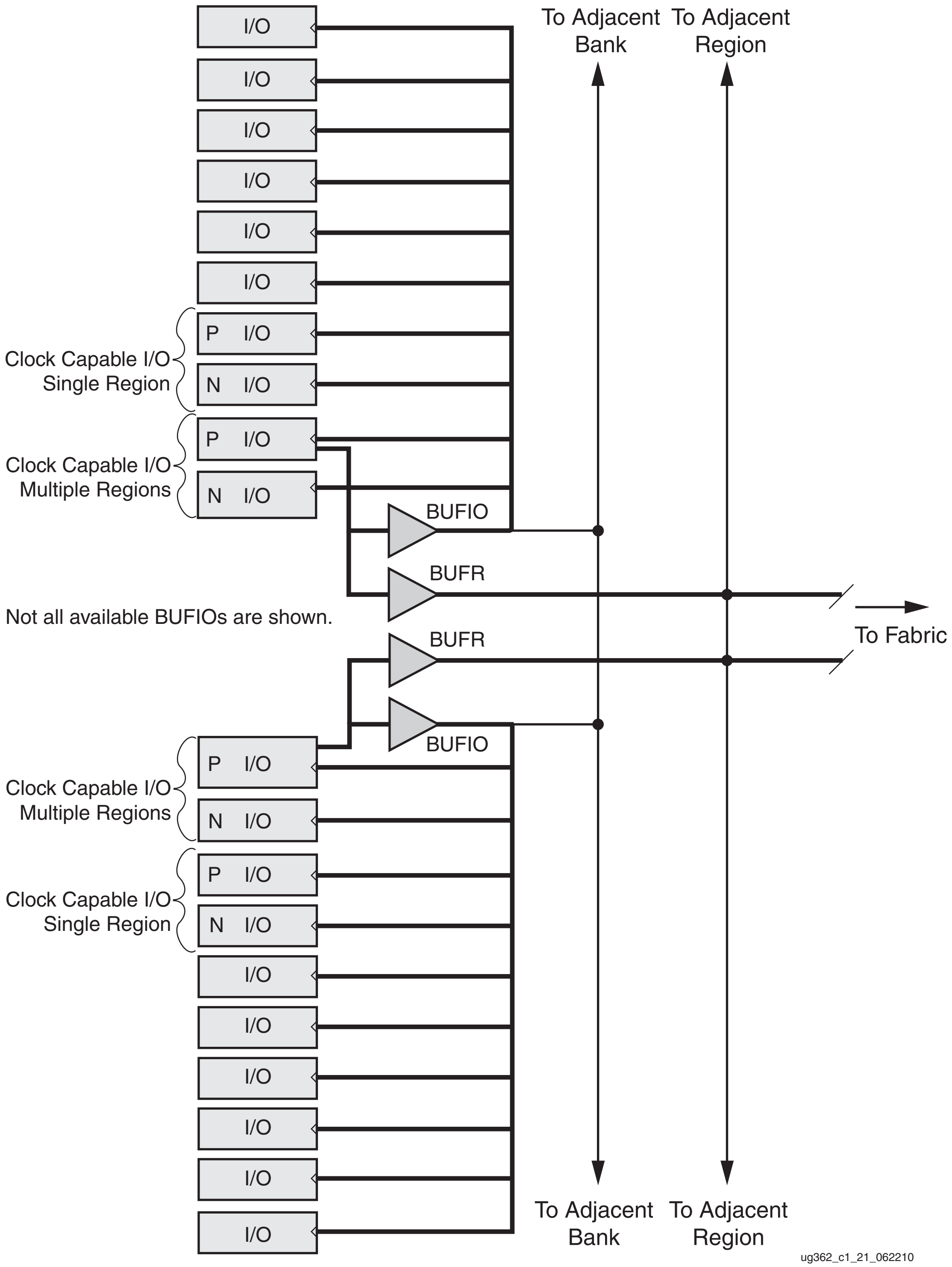

I/O
To Adjacent Bank
To Adjacent Region
P
N
Clock Capable I/O Single Region
Clock Capable I/O Multiple Regions
BUFIO
BUFR
Not all available BUFIOs are shown.
To Fabric
ug362_c1_21_062210

Figure 1-22: **BUFIO Driving I/O Logic**

Regional Clock Buffer - BUFR

The regional clock buffer (BUFR) is another clock buffer available in Virtex-6 devices. BUFRs drive clock signals to a dedicated clock net within a clock region, independent from the global clock tree. Each BUFR can drive the six regional clock nets in the region it is located, and the six clock nets in the adjacent clock regions (up to three clock regions). Unlike BUFIOs, BUFRs can drive the I/O logic *and* logic resources (CLB, block RAM, etc.) in the existing and adjacent clock regions. BUFRs can be driven by clock-capable pins, local interconnect, GTs, and the MMCMs high-performance clocks. In addition, BUFR is capable of generating divided clock outputs with respect to the clock input. The divide values are an integer between one and eight. BUFRs are ideal for source-synchronous applications requiring clock domain crossing or serial-to-parallel conversion.

Each I/O column supports regional clock buffers. There are up to four I/O columns in a device with two inner columns (center left and right) and up to two outer left and right columns. The availability of the outer columns are device dependant while the inner columns are always present. The Virtex-6 architecture therefore can have up to four BUFRs per region with two driving from the inner columns out (always present), and two BUFRs per region driving from the outer I/O columns in (when present). In Virtex-6 devices, BUFRs can also directly drive MMCM clock inputs and BUFGs.

BUFR Primitive

BUFR (Figure 1-23 and Table 1-7) is a clock-in/clock-out buffer with the capability to divide the input clock frequency. The Virtex-6 FPGA BUFRs can also directly drive MMCM clock inputs and BUFGs.

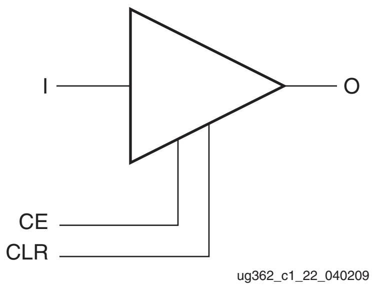


Figure 1-23: **BUFR Primitive**

Table 1-7: **BUFR Port List and Definitions**

Port Name	Type	Width	Definition
O	Output	1	Clock output port
CE	Input	1	Output clock enable port. Cannot be used in BYPASS mode.
CLR	Input	1	Asynchronous clear for the divide logic, and sets the output Low. Cannot be used in BYPASS mode.
I	Input	1	Clock input port

Additional Notes on the CE Pin

When CE is asserted/deasserted, the output clock signal turns on/off. When global set/reset (GSR) signal is High, BUFR does not toggle, even if CE is held High. The BUFR output toggles after the GSR signal is deasserted when a clock is on the BUFR input port.

BUFR Attributes and Modes

Clock division in the BUFR is controlled in software through the BUFR_DIVIDE attribute. Table 1-8 lists the possible values when using the BUFR_DIVIDE attribute.

Table 1-8: **BUFR_DIVIDE Attribute**

Attribute Name	Description	Possible Values
BUFR_DIVIDE	Defines whether the output clock is a divided version of the input clock.	1, 2, 3, 4, 5, 6, 7, 8 BYPASS (default)

Notes:

1. Location constraint is available for BUFR.

The propagation delay through BUFR is different for BUFR_DIVIDE = 1 and BUFR_DIVIDE = BYPASS. When set to 1, the delay is slightly more than BYPASS. All other divisors have the same delay BUFR_DIVIDE = 1. The phase relationship between the input clock and the output clock is the same for all possible divisions except BYPASS.

The timing relationship between the inputs and output of BUFR when using the BUFR_DIVIDE attribute is illustrated in Figure 1-24. In this example, the BUFR_DIVIDE attribute is set to three. Sometime before this diagram CLR was asserted.

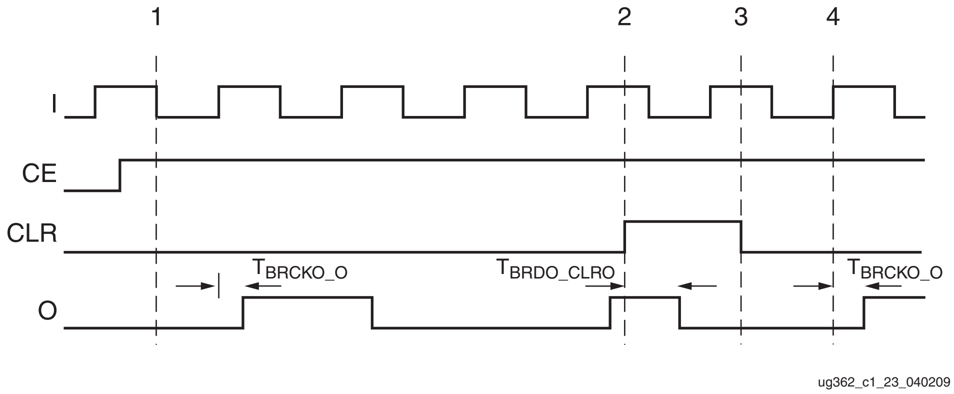


Figure 1-24: **BUFR Timing Diagrams with BUFR_DIVIDE Values**

In Figure 1-24:

- Before clock event 1, CE is asserted High.
- After CE is asserted and time T_{BRCKO_O}, the output O begins toggling at the divide by three rate of the input I. T_{BRCKO_O} and other timing numbers are best found in the speed specification.

  Note: The duty cycle is not 50/50 for odd division. The Low pulse is one cycle of I longer.
- At time event 2, CLR is asserted. After T_{BRDO_CLRO} from time event 2, O stops toggling.
- At time event 3, CLR is deasserted.
- At time T_{BRCKO_O} after clock event 4, O begins toggling again at the divided by three rate of I.

BUFR Use Models

BUFRs are ideal for source-synchronous applications requiring clock domain crossing or serial-to-parallel conversion. Unlike BUFIOs, BUFRs are capable of clocking logic resources in the FPGAs other than the IOBs. Figure 1-25 is a BUFR design example.

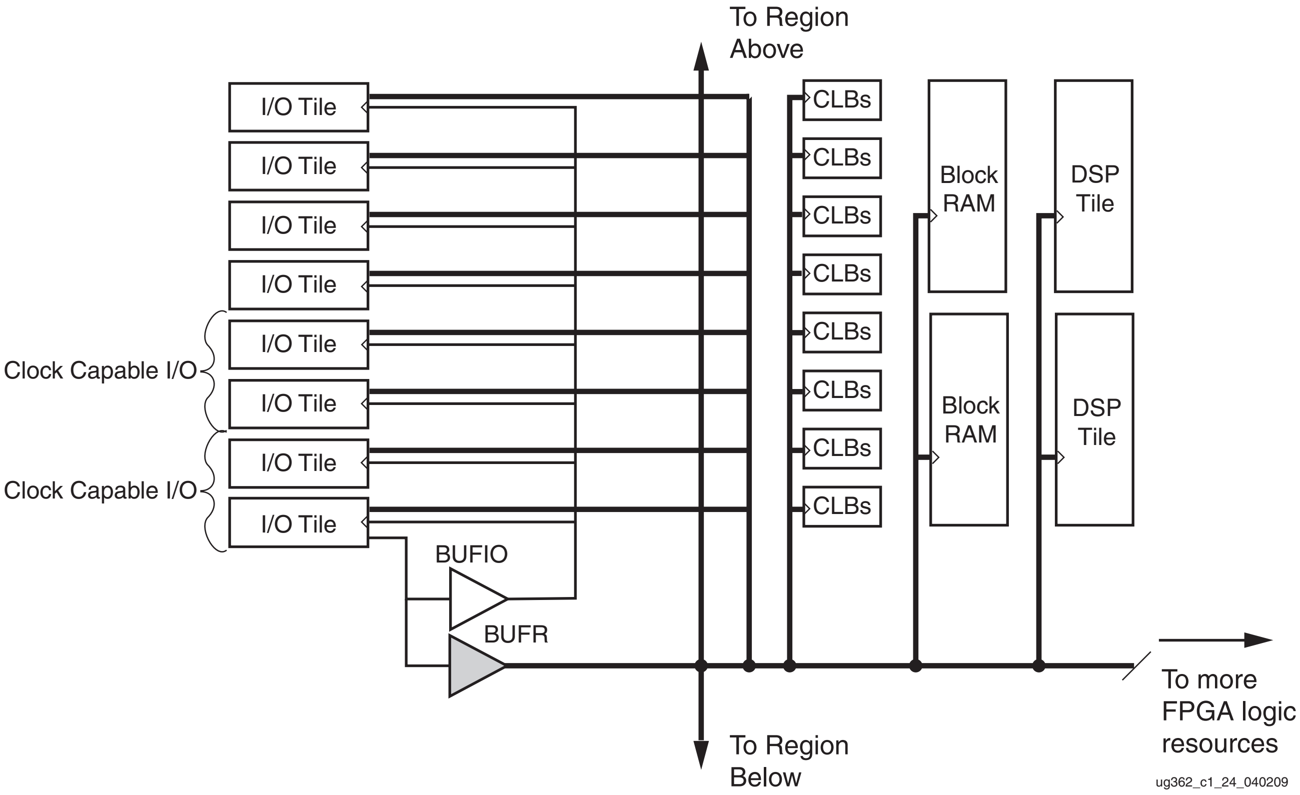


Figure 1-25: **BUFR Driving Various Logic Resources**

Regional Clock Nets

In addition to global clock trees and nets, Virtex-6 devices contain regional clock nets. These clock trees are also designed for low-skew and low-power operation. Unused branches are disconnected. The clock trees also manage the load/fanout when all the logic resources are used.

Regional clock nets do not propagate throughout the whole Virtex-6 device. Instead, they are limited to only one clock region. One clock region contains six independent regional clock nets.

To access regional clock nets, BUFRs must be instantiated. A BUFR can drive regional clocks in up to two adjacent clock regions (Figure 1-26). BUFRs in the top or bottom region can only access one adjacent region; below or above respectively.

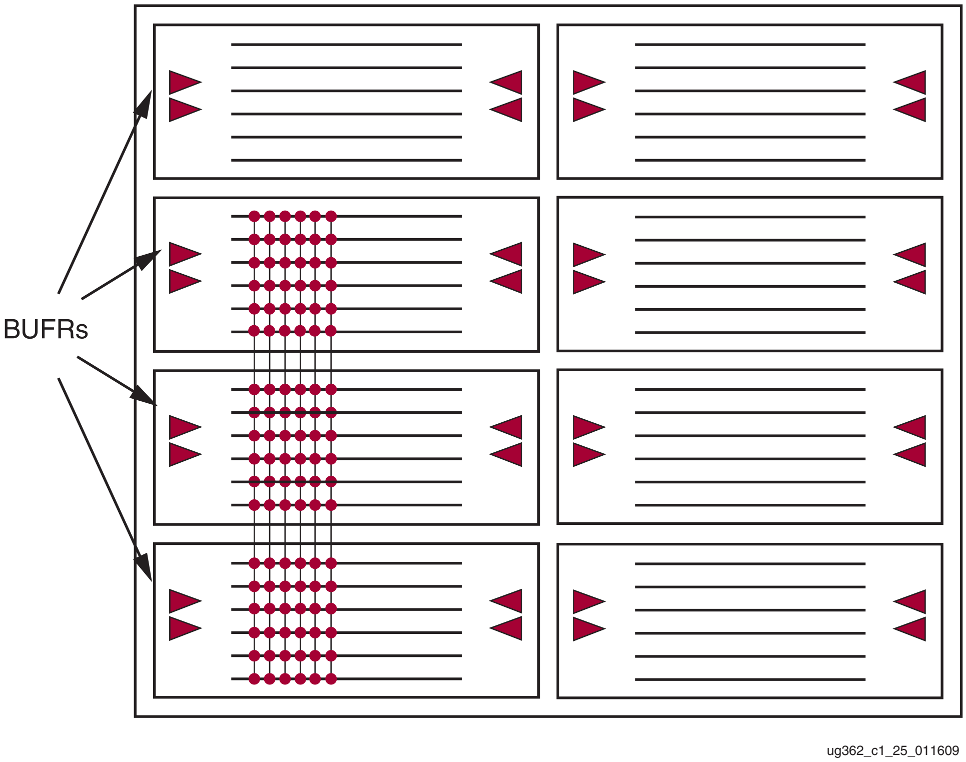


Figure 1-26: **BUFR Driving Multiple Regions**

Horizontal Clock Buffer - BUFH

The horizontal clock buffer (BUFH) drives a horizontal global clock tree spine in a single region (Figure 1-27). Each region has 12 BUFHs available. Every BUFH has a clock enable pin (CE) that allows the clocks to be turned-off dynamically. BUFHs can be driven from the center out by:

- MMCM outputs in the same region
- BUFG outputs
- Local interconnect
- Clock-capable I/Os from either the left or right inner column banks that are adjacent to the horizontal clocking region

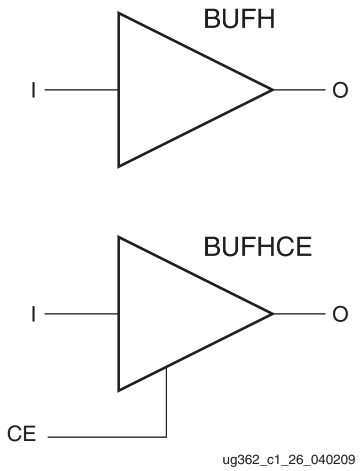


Figure 1-27: **BUFH and BUFHCE Primitives**

Table 1-9: **BUFH and BUFHCE Port List and Definitions**

Port Name	Type	Width	Definition
O	Output	1	Clock output port
CE	Input	1	Output clock enable port
I	Input	1	Clock input port

To use the BUFH, the logic must fit into the two regions adjacent to each other (left and right) as illustrated in Figure 1-28. The clock enable pin can completely turn off the clocks thus realizing potential power savings. The power consumption in a BUFH has lower power consumption when compared to a BUFG driving two adjacent regions with lower jitter and higher performance.

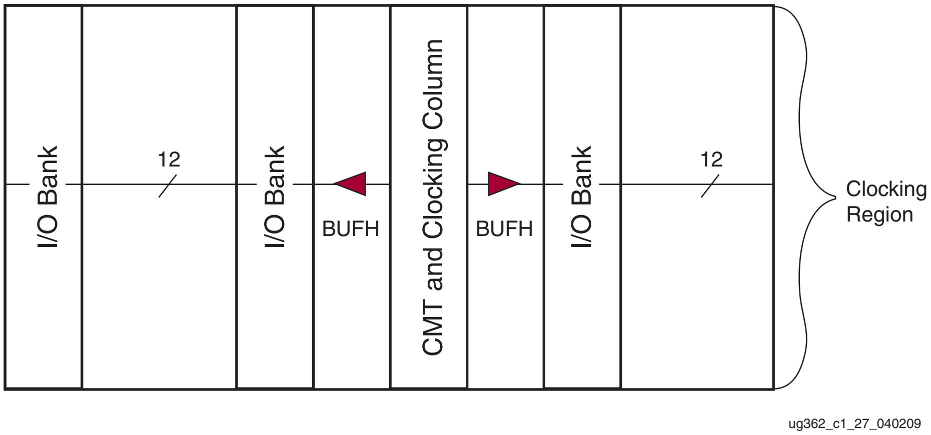
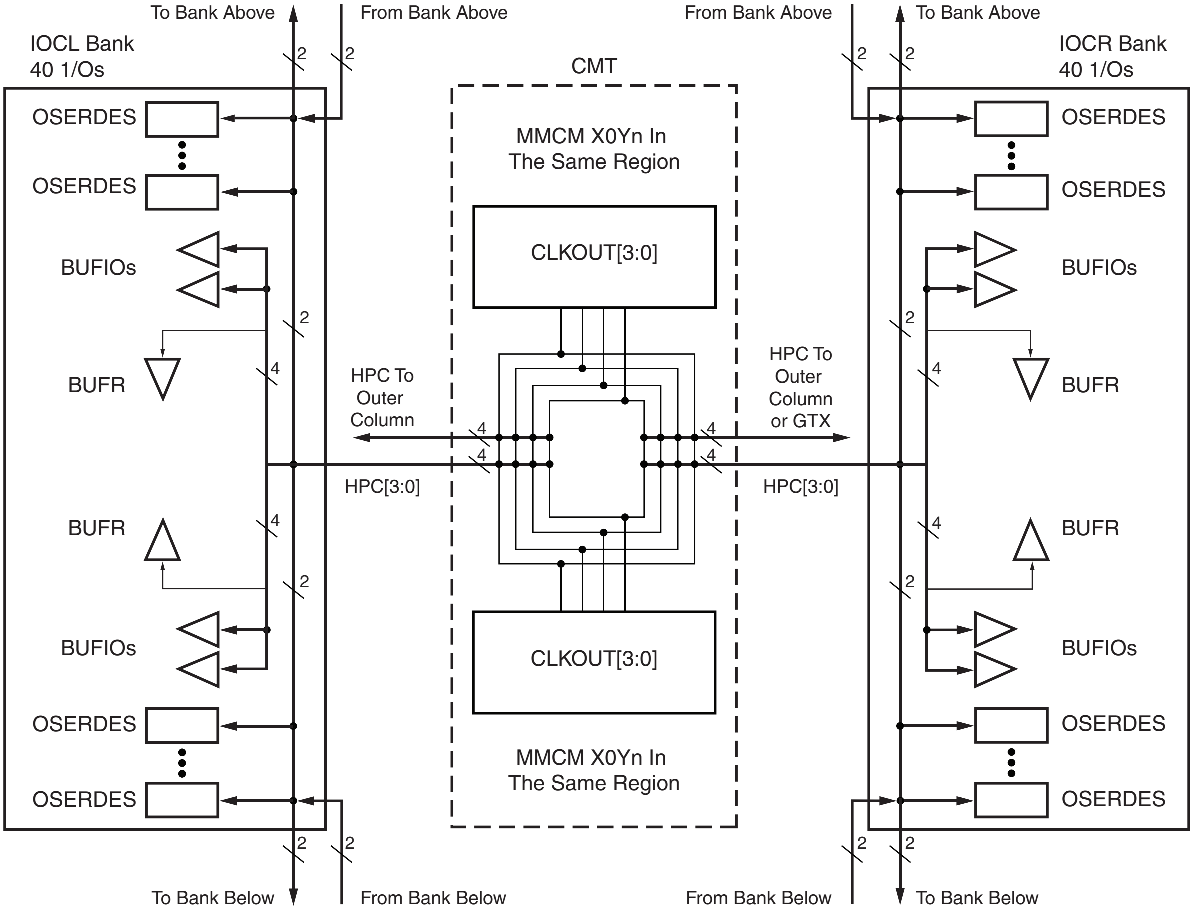


Figure 1-28: **Horizontal Clock Buffer**

High-Performance Clocks

Virtex-6 devices contain four high-performance clocks (HPC) per I/O column: in each region from the MMCMs and in the same region to the inner and outer I/O columns, including the GTX/GTH transceiver columns. These clocks (Figure 1-29) are directly driven by the MMCMs regulated power supply and never enter the V_{CCINT} supply domain. Therefore, these clocks exhibit very low jitter and minimal duty-cycle distortion. In the I/O columns the HPC connects to the BUFIO and drives the I/O logic. Two of the four HPCs can drive directly into the I/O banks (above and below) without using the multiregion BUFIO. The HPCs can also directly connect to the OSERDES without going through another clock buffer. This provides a forwarded clock with very low jitter and low duty-cycle distortion. An HPC has no buffer associated with it. The ISE software automatically determines when to use this resource by examining the connections to the IOLOGIC (OLOGIC) in the design. HPCs can also drive BUFRs in the same region to support the source-synchronous interface designs. Either MMCM in a CMT (CLKOUT0 through CLKOUT3) can drive the HPCs in the left- or right-side regions.



Notes:
1) Any of the MMCM CLKOUT[3:0] outputs can drive any of the HPC[3:0] to the inner or outer columns or GTX column (outer I/O and GTX column not shown).
2) HPCs can drive OSERDES directly. The same two HPCs can drive directly OSERDES in the adjacent Banks.
3) Any MMCM CLKOUT[3:0] can drive any BUFIO and any BUFR.

UG362_c1_28_011609

Figure 1-29: **High-Performance Clocks**

VHDL and Verilog Templates

The VHDL and Verilog code for all clocking resource primitives and ISE language templates are available in the Libraries Guide.

Chapter 2

Mixed-Mode Clock Manager

Introduction

The Clock Management Tile (CMT) in Virtex-6 FPGAs includes two Mixed-Mode Clock Managers (MMCMs). At the core of the MMCM is the Phase-Locked Loop (PLL) architecture similar to Virtex-5 FPGAs with enhanced functions and capabilities. There are dedicated routes within a CMT to chain two MMCMs. Each MMCM within the tile can be treated separately; however, there exists a dedicated routing between MMCMs. Using these dedicated routes frees up global resources for other design elements. Additionally, the use of local routes within the CMT provides an improved clock path because the route is handled locally, reducing chances for noise coupling.

The CMT diagram (Figure 2-1) shows a high-level view of the connection between the various clock input sources and the MMCM-to-MMCM. In Virtex-6 FPGAs the clock input connectivity is greatly enhanced allowing multiple resources to provide the reference clock(s) to the MMCMs. The number of output counters (dividers) has increased to eight with some of them capable of driving out an inverted clock signal (180° phase shift). For backward compatibility with DCMs, nine independent outputs can be selected for mapping the DCM outputs directly into the MMCM. The MMCM output clocks in the same CMT can be chained by multiplexing any MMCM output into a single clock signal for use as a reference clock to the other MMCM. Virtex-6 FPGA MMCMs have added infinite fine phase-shift capability in either direction and can be used in dynamic or fixed phase-shift mode. The resolution of the fine phase shift depends on the VCO frequency. Fractional divide functionality in increments of 1/8th (0.125) is added to support greater clock frequency synthesis capability. A fractional divide can be performed by combining two counters in the CLK0 output path. In addition, various ports and functions are added to make the MMCM compatible with the DCMs in previous architectures. The outputs from the MMCM are not spread spectrum, however a spread spectrum on the input clock will not be filtered and thus passed on the output clocks.

Any used MMCM requires a calibration after a user reset or user power-down is issued. Similarly, a calibration is required after power-up. Some versions of the ISE® software (v11.5 and later) automatically insert a calibration circuit that produces an additional reset of any used MMCM after the initial LOCK. This circuit disables the STARTUP_WAIT functionality after configuration for the MMCMs. The STARTUP_WAIT attribute must always be set to FALSE.

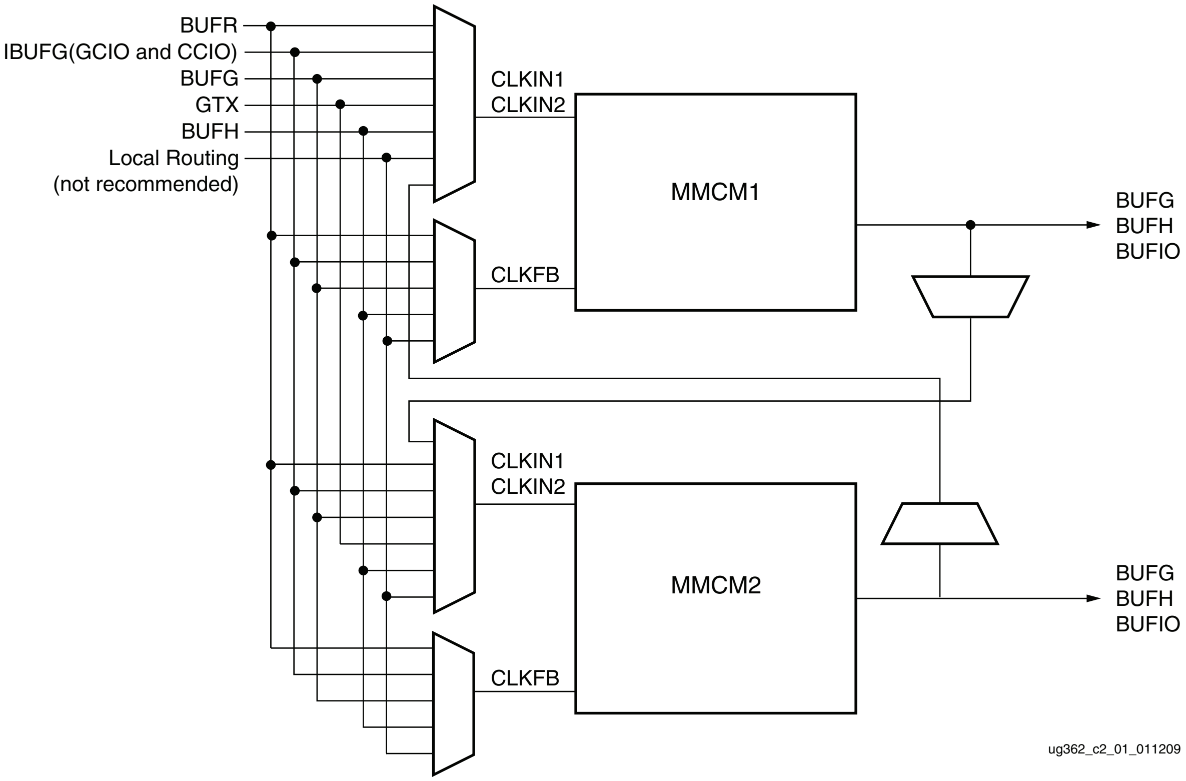


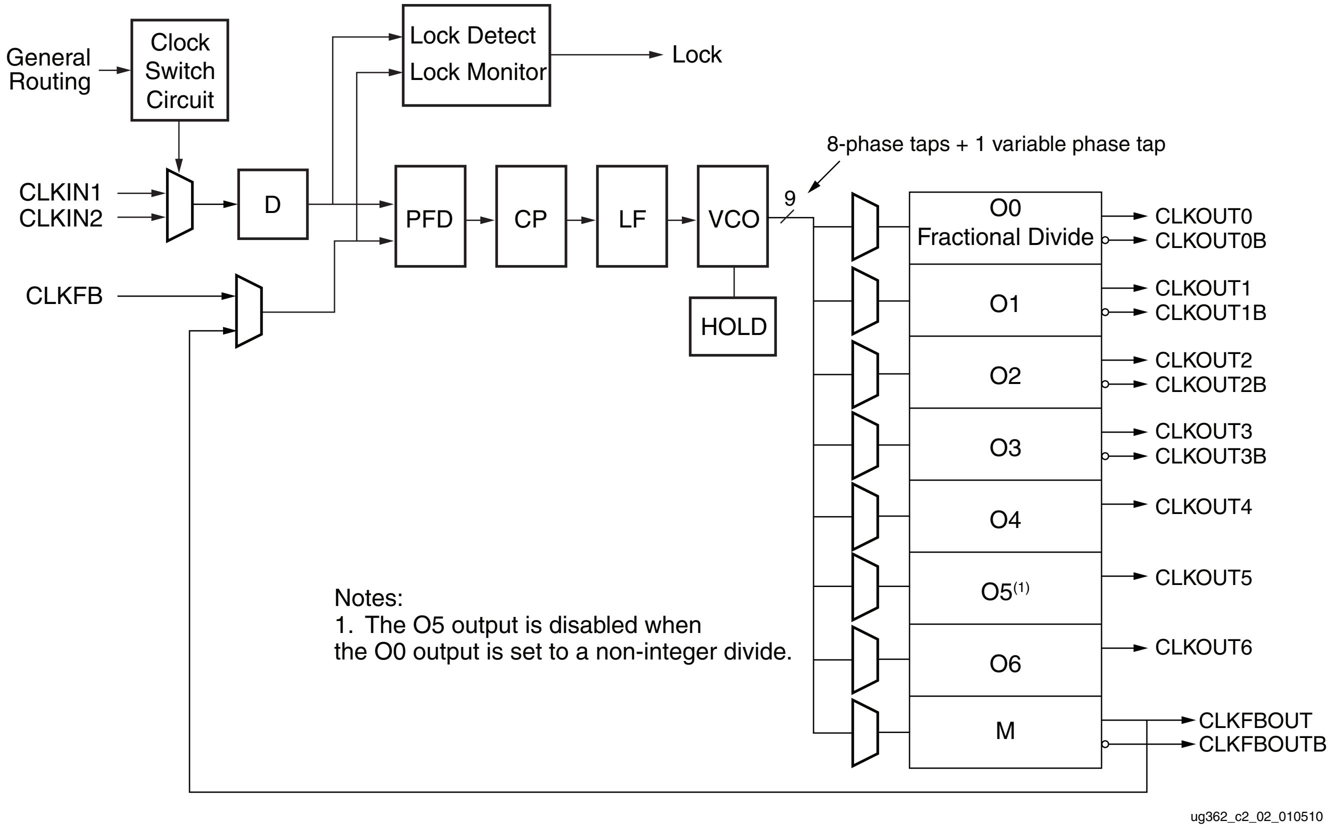
Figure 2-1: **Block Diagram of the Virtex-6 FPGA CMT**

MMCMs

Virtex-6 devices contain up to nine CMT tiles. The MMCMs serve as a frequency synthesizer for a wide range of frequencies, serve as a jitter filter for either external or internal clocks, and deskew clocks.

Input MUXes select the reference and feedback clocks from either the IBUFG, BUFG, BUFG, GTs (CLKIN only), or interconnect (not recommended). Each clock input has a programmable counter divider (D). The Phase-Frequency Detector (PFD) compares both phase and frequency of the rising edges of both the input (reference) clock and the feedback clock. If a minimum High/Low pulse is maintained, the duty cycle is ancillary. The PFD is used to generate a signal proportional to the phase and frequency between the two clocks. This signal drives the Charge Pump (CP) and Loop Filter (LF) to generate a reference voltage to the VCO. The PFD produces an up or down signal to the charge pump and loop filter to determine whether the VCO should operate at a higher or lower frequency. When VCO operates at too high of a frequency, the PFD activates a down signal, causing the control voltage to be reduced decreasing the VCO operating frequency. When the VCO operates at too low of a frequency, an up signal will increase voltage. The VCO produces eight output phases and one variable phase for fine-phase shifting. Each output phase can be selected as the reference clock to the output counters (Figure 2-2). Each counter can be independently programmed for a given customer design. A special counter, M, is also provided. This counter controls the feedback clock of the MMCM allowing a wide range of frequency synthesis.

In addition to integer divide output counters, Virtex-6 devices adds a fractional counter by combining the O0/O5 counters. When used in fractional mode, the O5 output is not available.



Notes:
1. The O5 output is disabled when the O0 output is set to a non-integer divide.

Figure 2-2: **Detailed MMCM Block Diagram**

General Usage Description

MMCM Primitives

The two Virtex-6 FPGA MMCM primitives, MMCM_BASE and MMCM_ADV, are shown in Figure 2-3.

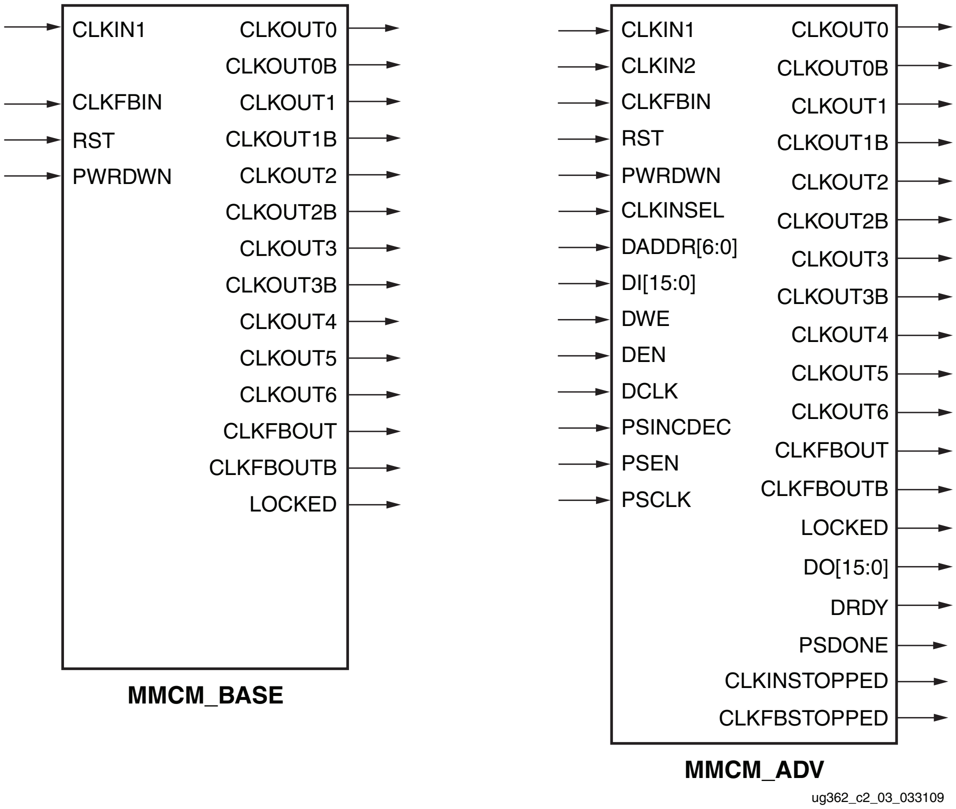


Figure 2-3: **MMCM Primitives**

MMCM_BASE Primitive

The MMCM_BASE primitive provides access to the most frequently used features of a stand alone MMCM. Clock deskew, frequency synthesis, coarse phase shifting, and duty cycle programming are available to use with the MMCM_BASE. The ports are listed in Table 2-1.

Table 2-1: **MMCM_BASE Ports**

Description	Ports
Clock Input	CLKIN, CLKFBIN
Control Inputs	RST
Clock Output	CLKOUT0 to CLKOUT6, CLKOUT0B to CLKOUT3B, CLKFBOUT, and CLKFBOUTB
Status and Data Outputs	LOCKED
Power Control	PWRDWN

MMCM_ADV Primitive

The MMCM_ADV primitive provides access to all MMCM_BASE features plus additional ports for clock switching, connectivity to the other MMCM in the same CMT, access to the Dynamic Reconfiguration Port (DRP), as well as dynamic fine-phase shifting. The ports are listed in Table 2-2.

Table 2-2: **MMCM_ADV Ports**

Description	Ports
Clock Input	CLKIN1, CLKIN2, CLKFBIN, DCLK, PSCLK
Control and Data Input	RST, CLKINSEL, DWE, DEN, DADDR, DI, PSINCDEC, PSEN
Clock Output	CLKOUT0 to CLKOUT6, CLKOUT0B to CLKOUT3B, CLKFBOUT, and CLKFBOUTB
Status and Data Output	LOCKED, DO, DRDY, PSDONE, CLKINSTOPPED, CLKFBSTOPPED
Power Control	PWRDWN

The Virtex-6 FPGA MMCM is a mixed signal block designed to support clock network deskew, frequency synthesis, and jitter reduction. These three modes of operation are discussed in more detail within this section. The Voltage Controlled Oscillator (VCO) operating frequency can be determined by using the following relationship:

$$F_{VCO} = F_{CLKIN} \times \frac{M}{D} \qquad \text{Equation 2-1}$$

$$F_{OUT} = F_{CLKIN} \times \frac{M}{D \times O} \qquad \text{Equation 2-2}$$

where the M, D, and O counters are shown in Figure 2-2. The value of M corresponds to the CLKFBOUT_MULT_F setting, the value of D to the DIVCLK_DIVIDE, and O to the CLKOUT_DIVIDE.

The seven "O" counters can be independently programmed. For example, O0 can be programmed to do a divide-by-two while O1 is programmed for a divide by three. The only constraint is that the VCO operating frequency must be the same for all the output counters since a single VCO drives all the counters.

Clock Network Deskew

In many cases, designers do not want to incur the delay on a clock network in their I/O timing budget therefore they use a MMCM to compensate for the clock network delay. Virtex-6 FPGAs support this feature. A clock output matching the reference clock CLKIN frequency (always CLKFBOUT) is connected to a BUFG in the same half of the device and fed back to the CLKFBIN feedback pin of the MMCM. The remaining outputs can still be used to divide the clock down for additionally synthesized frequencies. In this case, all output clocks have a defined phase relationship to the input reference clock.

Frequency Synthesis Only Using Integer Divide

The MMCMs can also be used for stand alone frequency synthesis. In this application, the MMCM is not used to deskew a clock network, but rather generate an output clock frequency for other blocks. In this mode, the MMCM feedback paths will be internal since it keeps all the routing local and should minimize the jitter. Figure 2-4 shows the MMCM configured as a frequency synthesizer. In this example, an external 33 MHz reference clock is available. The reference clock can be a crystal oscillator or the output of another MMCM. Setting the M counter to 32 makes the VCO oscillate at 1066 MHz (33.333 MHz x 32). The six MMCM outputs are programmed to provide (for example) a 533 MHz processor clock, a 266 MHz gasket clock, a 178 MHz clock, a 133 MHz memory interface clock, a 66 MHz interface, and a 33 MHz interface. In this example, there are no required phase relationships between the reference clock and the output clocks, but there are required relationships between the output clocks.

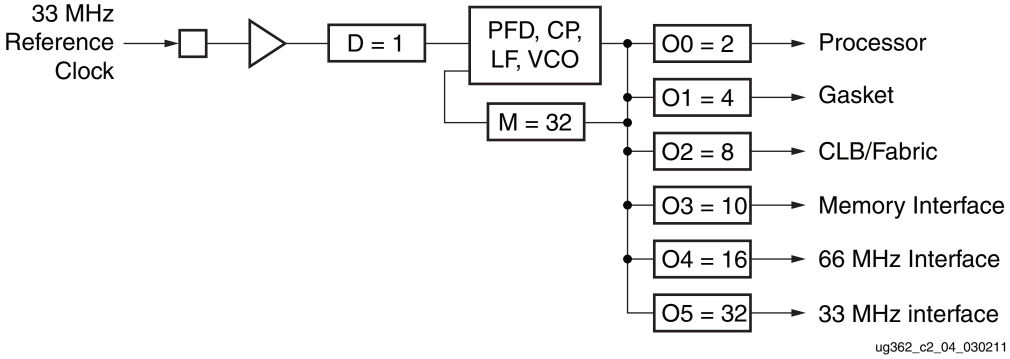


Figure 2-4: **MMCM as a Frequency Synthesizer**

Frequency Synthesis Using Fractional Divide

Virtex-6 FPGAs support fractional (non-integer) divides in the CLKOUT0 output path. If the CLKOUT0 counter is used in fractional mode, then the CLKOUT5 counter output is not available. The resolution of the fractional divide is 1/8 or 0.125 degrees, effectively increasing the number of synthesizeable frequencies by a factor of eight. For example, if the CLKIN frequency is 100 MHz and the M divide value is set to 8, then the VCO frequency is 800 MHz. CLKOUT0 can be used to further fractionally divide the 800 MHz VCO frequency (e.g., CLKOUT0_DIVIDE = 2.5 resulting in a 320 MHz output frequency).

When using the fractional divider then the duty cycle and phase shift are not programmable for outputs used in the fractional mode.

Jitter Filter

MMCMs always reduce the jitter inherent on a reference clock. The MMCM can be instantiated as a standalone function to simply support filtering jitter from an external clock before it is driven into the another block. As a jitter filter, it is usually assumed that the MMCM acts as a buffer and regenerates the input frequency on the output (e.g., F_{IN} = 100 MHz, F_{OUT} = 100 MHz). In general, greater jitter filtering is possible by using the MMCM attribute BANDWIDTH set to Low. Setting the BANDWIDTH to Low can incur an increase in the static offset of the MMCM.

Limitations

The MMCM has some restrictions that must be adhered to. These are summarized in the MMCM electrical specification in the *Virtex-6 FPGA Data Sheet*. In general, the major limitations are VCO operation range, input frequency, duty cycle programmability, and phase shift.

VCO Operating Range

The minimum and maximum VCO operating frequencies are defined in the electrical specification of the *Virtex-6 FPGA Data Sheet*. These values can also be extracted from the speed specification.

Minimum and Maximum Input Frequency

The minimum and maximum CLKIN input frequency are defined in the electrical specification of the *Virtex-6 FPGA Data Sheet*.

Duty Cycle Programmability

Only discrete duty cycles are possible given a VCO operating frequency. Depending on the CLKOUT_DIVIDE value, a minimum and maximum range is possible with a step size that is also depending on the CLKOUT_DIVIDE value. The Clocking Wizard tool gives the possible values for a given CLKOUT_DIVIDE.

Phase Shift

In many cases, there needs to be a phase shift between clocks. The MMCM has multiple options to implement phase shifting. Static phase shifting can be achieved by selecting one of the eight VCO output phases with additional fine phase shifting available in the CLKOUT output counters depending on the CLKOUT divide value. In Virtex-6 FPGAs there is also an interpolated phase shifting capability in either fixed or dynamic mode. The MMCM phase shifting capabilities are very powerful which can lead to complex scenarios. It is best to consult the software tools for selecting the proper phase-shift methodology.

Static Phase Shift Mode

The Static Phase Shift (SPS) resolution in time units is defined as:

$$SPS = \frac{1}{8F_{VCO}}period \text{ or } \frac{D}{8MF_{IN}}period \qquad \textit{Equation 2-3}$$

Since the VCO can provide eight phase shifted clocks at 45° each; always providing possible settings for 0°, 45°, 90°, 135°, 180°, 225°, 270°, and 315° of phase shift. The higher the VCO frequency is, the smaller the phase-shift resolution. Since the VCO has a distinct operating range, it is possible to bound the phase-shift resolution using from $\frac{1}{8F_{VCOMIN}}\ to\ \frac{1}{8F_{VCOMAX}}$ period.

Each CLKOUT output counter is individually programmable allowing each to have an additional phase-shift resolution in degrees based on the phase of the VCO selected and the CLKOUT counter divide value. The granularity of the CLKOUT phase-shift value can be calculated as 45°/CLKOUT_DIVIDE value. The maximum phase-shift range is also determined by the CLKOUT_DIVIDE value. The maximum phase shift is 360° when CLKOUT_DIVIDE ≤ 64. When CLKOUT_DIVIDE is > 64, the maximum phase shift is:

$$\text{Maximum Phase Shift } = \left(\frac{64}{(CLKOUT\_\ DIVIDE)} \times 360\right) + (7 \times \text{Phase Shift Value})$$

It is possible to phase shift the CLKFBOUT feedback clock. In that case all CLKOUT output clocks are negatively phase shifted with respect to CLKIN.

Interpolated Fine Phase Shift in Fixed or Dynamic Mode

Interpolated Fine Phase Shift (IFPS) mode in the MMCM has linear shift behavior independent of the CLKOUT_DIVIDE value and the phase shift resolution only depends on the VCO frequency. In this mode the output clocks can be rotated 360° round robin in linear increments of $\frac{1}{56F_{VCO}}$.

If the VCO runs at 600 MHz, then the phase resolution is approximately (rounded) 30 ps and at 1.6 GHz is approximately (rounded) 11 ps.

The phase shift value can be programmed as a fixed value set during configuration or a dynamic increment/decrement under application control after configuration. The dynamic phase shift is controlled by the PS interface of the MMCM_ADV. This phase-shift mode equally affects all CLKOUT output clocks that are selected for this mode by setting the USE_FINE_PS attribute to TRUE. It is possible for each individual CLKOUT counter to either select the interpolated, the previously described static phase-shift mode or none. Fractional divide is not allowed in this mode. Fixed or dynamic phase shifting of the feedback path will result in a negative phase shift of all output clocks with respect to CLKIN. The dynamic phase-shift interface can not be used when the phase-shift mode is set to fixed.

Dynamic Phase Shift Interface

The MMCM_ADV primitive provides three inputs and one output for dynamic fine-phase shifting. Each CLKOUT and the CLKFBOUT divider can be individually selected for phase shifting. The attributes CLKOUT[0:6]_USE_FINE_PS and CLKFBOUT_USE_FINE_PS select the output clocks to be dynamically phase shifted. The dynamic phase-shift amount is common to all the output clocks selected.

The variable phase shift is controlled by the PSEN, PSINCDEC, PSCLK, and PSDONE ports (Figure 2-5). After the MMCM locks, the initial phase is determined by the CLKOUT_PHASE attribute. Most commonly, no initial phase shift is selected. The phase of the MMCM output clock(s) increments/decrements according to the interaction of PSEN, PSINCDEC, PSCLK, and PSDONE from the initial or previously performed dynamic phase shift. PSEN, PSINCDEC, and PSDONE are synchronous to PSCLK. When PSEN is asserted for one PSCLK clock period, a phase-shift increment/decrement is initiated. When PSINCDEC is High, an increment is initiated and when PSINCDEC is Low, a decrement is initiated. Each increment adds to the phase shift of the MMCM clock outputs by 1/56th of the VCO period. Similarly, each decrement decreases the phase shift by 1/56th of the VCO period. PSEN must be active for one PSCLK period. PSDONE is High for exactly one clock period when the phase shift is complete. The number of PSCLK cycles is deterministic. After initiating the phase shift by asserting PSEN and until the completion of the phase shift signaled by PSDONE, the MMCM output clocks and the MMCM output clocks gradually drift from their original phase shift to an increment/decrement phase shift in a linear fashion. The completion of the increment or decrement is signaled when PSDONE asserts High. After PSDONE has pulsed High, another increment/decrement can be initiated. There is no maximum phase shift or phase-shift overflow. An entire clock period (360 degrees) can always be phase shifted regardless of frequency. When the end of the period is reached, the phase shift simply wraps around round-robin style.

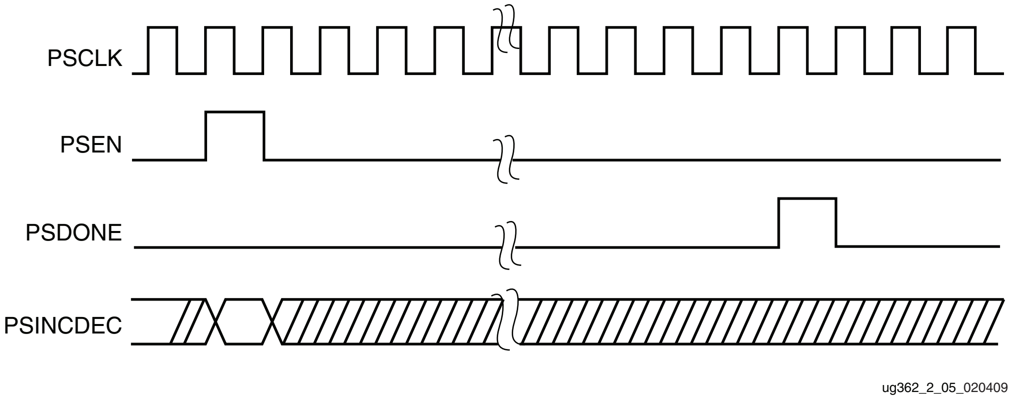


Figure 2-5: **Phase-Shift Timing Diagram**

Counter Cascading

The CLKOUT6 divider (counter) can be cascaded with the CLKOUT4 divider. This provides a capability to have an output divider that is larger than 128. CLKOUT6 simply feeds the input of the CLKOUT4 divider. There is a static phase offset between the output of the cascaded divider and all other output dividers.

MMCM Programming

Programming of the MMCM must follow a set flow to ensure configuration that guarantees stability and performance. This section describes how to program the MMCM based on certain design requirements. A design can be implement in two ways, directly through the GUI interface (the Clocking Wizard) or directly implementing the MMCM through instantiation. Regardless of the method selected, the following information is necessary to program the MMCM:

- Reference clock period
- Output clock frequencies (up to seven maximum)
- Output clock duty cycle (default is 50%)
- Output clock phase shift in number of degrees relative to the original 0 phase of the clock.
- Desired bandwidth of the MMCM (default is OPTIMIZED and the bandwidth is chosen in software)
- Compensation mode (automatically determined by the software)
- Reference clock jitter in UI (i.e., a percentage of the reference clock period)

Determine the Input Frequency

The first step is to determine the input frequency. This allows all possible output frequencies to be determined by using the minimum and maximum input frequencies to define the D counter range, the VCO operating range to determine the M counter range, and the output counter range. There can be a very large number of frequencies. When using integer divides, in the worst case, there will be 80 x 64 x 128 = 655,360 possible combinations. In reality, the total number of different frequencies is less since the entire range of the M and D counters cannot be realized and there is overlap between the various settings.

As an example, consider F_{IN} = 100 MHz. If the minimum PFD frequency is 10 MHz, then D can only go from 1 to 10.

- D = 1, M can only have values from four to 16.
- D = 2, M can only have values from eight to 32.
- D = 4, M can only have values from 16 to 64.

In addition, D = 1 M = 4 is a subset of D = 2 M = 8, D = 4 M = 16, and D = 8 M = 32 allowing these cases to be dropped. For this case, only D = 1, 3, 5, 6, 7, and 9 are considered since all other D values are subsets of these cases. This drastically reduces the number of possible output frequencies. The output frequencies are sequentially selected. The desired output frequency should be checked against the possible output frequencies generated. Once the first output frequency is determined, an additional constraint can be imposed on the values of M and D. This can further limit the possible output frequencies for the second output frequency. Continue this process until all the output frequencies are selected.

The constraints used to determine the allowed M and D values are shown in the following equations:

$$D_{MIN} = roundup\frac{f_{IN}}{f_{PFD\ MAX}} \qquad \text{Equation 2-4}$$

$$D_{MAX} = rounddown\frac{f_{IN}}{f_{PFD\ MIN}} \qquad \text{Equation 2-5}$$

$$M_{MIN} = roundup\left(\frac{f_{VCOMIN}}{f_{IN}} \times D_{MIN}\right) \qquad \text{Equation 2-6}$$

$$M_{MAX} = rounddown\left(\frac{f_{VCOMAX}}{f_{IN}} \times D_{MAX}\right) \qquad \text{Equation 2-7}$$

Note: M and D have additional limitations as listed in Table 2-4.

Determine the M and D Values

Determining the input frequency can result in several possible M and D values. The next step is to determine the optimum M and D values. The starting M value is first determined. This is based off the VCO target frequency, the ideal operating frequency of the VCO.

$$M_{IDEAL} = \frac{D_{MIN} \times f_{VCOMAX}}{f_{IN}} \qquad \text{Equation 2-8}$$

The goal is to find the M value closest to the ideal operating point of the VCO. The minimum D value is used to start the process. The goal is to make D and M values as small as possible while keeping f_{VCO} as high as possible.

MMCM Ports

Table 2-3 summarizes the MMCM ports. Table 2-4 lists the MMCM attributes.

Table 2-3: **MMCM Ports**[1]

Pin Name	I/O	Pin Description
CLKIN1	Input	General clock input. See CLKIN1 – Primary Reference Clock Input.
CLKIN2	Input	Secondary clock input for the MMCM reference clock. See CLKIN2 – Secondary Clock Input.
CLKFBIN	Input	Feedback clock input. See CLKFBIN – Feedback Clock Input.
CLKINSEL	Input	Signal controls the state of the clock input MUX, High = CLKIN1, Low = CLKIN2. Dynamically switches the MMCM reference clock. See CLKINSEL – Clock Input Select.
RST	Input	Asynchronous reset signal. The RST signal is an asynchronous reset for the MMCM. The MMCM will synchronously re-enable itself when this signal is released (i.e., MMCM re-enabled). A reset is not required when the input clock conditions change (e.g., frequency). See RST – Asynchronous Reset Signal.
PWRDWN	Input	Powers down instantiated but unused MMCMs. See PWRDWN – Power Down.
DADDR[6:0]	Input	The dynamic reconfiguration address (DADDR) input bus provides a reconfiguration address for the dynamic reconfiguration. When not used, all bits must be assigned zeros. See DADDR[6:0] – Dynamic Reconfiguration Address.
DI[15:0]	Input	The dynamic reconfiguration data input (DI) bus provides reconfiguration data. When not used, all bits must be set to zero. See DI[15:0] – Dynamic Reconfiguration Data Input.
DWE	Input	The dynamic reconfiguration write enable (DWE) input pin provides the write enable control signal to write the DI data into the DADDR address. When not used, it must be tied Low. See DWE – Dynamic Reconfiguration Write Enable.
DEN	Input	The dynamic reconfiguration enable (DEN) provides the enable control signal to access the dynamic reconfiguration feature. When the dynamic reconfiguration feature is not used, DEN must be tied Low. See DEN – Dynamic Reconfiguration Enable Strobe.
DCLK	Input	The DCLK signal is the reference clock for the dynamic reconfiguration port. See DCLK – Dynamic Reconfiguration Reference Clock.
PSCLK	Input	Phase shift clock. See PSCLK – Phase-Shift Clock.
PSEN	Input	Phase shift enable. See PSEN – Phase-Shift Enable.
PSINCDEC	Input	Phase shift increment/decrement control. See PSINCDEC – Phase-Shift Increment/Decrement Control.
CLKOUT[0:6]	Output	User configurable clock outputs (0 through 6) that can be divided versions of the VCO phase outputs (user controllable) from 1 (bypassed) to 128. The output clocks are phase aligned to each other (unless phase shifted) and aligned to the input clock with a proper feedback configuration. See CLKOUT[0:6] – Output Clocks.
CLKOUT[0:3]B	Output	Inverted CLKOUT[0:3]. See CLKOUT[0:3]B – Inverted Output Clocks.
CLKFBOUT	Output	Dedicated MMCM feedback output. See CLKFBOUT – Dedicated MMCM Feedback Output.
CLKFBOUTB	Output	Inverted CLKFBOUT. See CLKFBOUTB – Inverted CLKFBOUT.
CLKINSTOPPED	Output	Status pin indicating that the input clock has stopped. See CLKINSTOPPED – Input Clock Status.

Table 2-3: **MMCM Ports[1] *(Cont'd)***

Pin Name	I/O	Pin Description
CLKFBSTOPPED	Output	Status pin indicating that the feedback clock has stopped. See CLKFBSTOPPED – Feedback Clock Status.
LOCKED	Output	An output from the MMCM that indicates when the MMCM has achieved phase alignment within a predefined window and frequency matching within a predefined PPM range. The MMCM automatically locks after power on. No extra reset is required. LOCKED will be deasserted if the input clock stops or the phase alignment is violated (e.g., input clock phase shift). The MMCM automatically re-aquires lock after LOCKED is deasserted. See LOCKED.
DO[15:0]	Output	The dynamic reconfiguration output bus provides MMCM data output when using dynamic reconfiguration. See DO[15:0] – Dynamic Reconfiguration Output Bus.
DRDY	Output	The dynamic reconfiguration ready output (DRDY) provides the response to the DEN signal for the MMCMs dynamic reconfiguration feature. See DRDY – Dynamic Reconfiguration Ready.
PSDONE	Output	Phase shift done. See PSDONE – Phase Shift Done.

Notes:

1. All control and status signals except PSINCDEC are active High.

MMCM Port Descriptions

CLKIN1 – Primary Reference Clock Input

CLKIN1 can be driven by an IBUFG (either through a global or clock-capable clock pin), BUFG, BUFR, BUFH, interconnect (not recommended), or directly from a high-speed serial transceiver.

CLKIN2 – Secondary Clock Input

CLKIN2 is a secondary clock input that is used to dynamically switch the MMCM reference clock. CLKIN2 can be driven by an IBUFG (either through a global or clock-capable clock pin), BUFG, BUFR, BUFH, interconnect (not recommended), or directly from a high-speed serial transceiver.

CLKFBIN – Feedback Clock Input

Must be connected either directly to the CLKFBOUT for internal feedback or IBUFG (either through a global or clock-capable pin for external deskew), BUFG, BUFH, or interconnect (not recommended). For external clock alignment, the feedback path clock buffer type should match the forward clock buffer type with the exception of BUFR. BUFR can not be compensated for.

CLKFBOUT – Dedicated MMCM Feedback Output

For possible configuration see MMCM Use Models. CLKFBOUT can also drive logic similar to the CLK0 of the DCM in Virtex-5 FPGAs.

CLKFBOUTB – Inverted CLKFBOUT

This signal should not be used for feedback. It provides an additional, inverted CLKFBOUT output clock. CLKFBOUTB can drive logic similar to the CLK180 clock of the DCM in Virtex-5 FPGAs.

CLKINSEL – Clock Input Select

The CLKINSEL signal controls the state of the clock input MUXes, High = CLKIN1, Low = CLKIN2 (see Reference Clock Switching). The MMCM must be held in RESET during clock switchover.

RST – Asynchronous Reset Signal

The RST signal is an asynchronous reset for the MMCM. The MMCM will be synchronously re-enabled when this signal is deasserted.

PWRDWN – Power Down

Powers down instantiated but currently unused MMCMs. This mode can be used to save power for temporarily inactive portions of the design and/or MMCMs that are not active in certain system configurations. No MMCM power is consumed in this mode.

DADDR[6:0] – Dynamic Reconfiguration Address

The dynamic reconfiguration address (DADDR) input bus provides a reconfiguration address for the dynamic reconfiguration. The address value on this bus specifies the 16 configuration bits that are written or read with the next DCLK cycle. When not used, all bits must be assigned zeros.

DI[15:0] – Dynamic Reconfiguration Data Input

The dynamic reconfiguration data input (DI) bus provides reconfiguration data. The value of this bus is written to the configuration cells. The data is presented in the cycle that DEN and DWE are active. The data is captured in a shadow register and written at a later time. DRDY indicates when the DRP port is ready to accept another write. When not used, all bits must be set to zero.

DWE – Dynamic Reconfiguration Write Enable

The dynamic reconfiguration write enable (DWE) input pin provides the write/read enable control signal to write the DI data into or read the DO data from the DADDR address. When not used, it must be tied Low.

DEN – Dynamic Reconfiguration Enable Strobe

The dynamic reconfiguration enable strobe (DEN) provides the enable control signal to access the dynamic reconfiguration feature and enables all DRP port operations. When the dynamic reconfiguration feature is not used, DEN must be tied Low.

DCLK – Dynamic Reconfiguration Reference Clock

The DCLK signal is the reference clock for the dynamic reconfiguration port. The rising edge of this signal is the timing reference for all other port signals. The setup time is specified in the data sheet. There is no hold time requirement for the other input signals relative to the rising edge of the DCLK. The pin can be drive by an IBUF, IBUFG, BUFG, BUFR, or BUFH.

PSCLK – Phase-Shift Clock

This input pin provides the source clock for the dynamic phase-shift interface. All other inputs are synchronous to the positive edge of this clock. The pin can be drive by an IBUF, IBUFG, BUFG, BUFR, or BUFH.

PSEN – Phase-Shift Enable

A dynamic (variable) phase-shift operation is initiated by synchronously asserting this signal. It must be activated for one cycle of the PSCLK. After initiating a phase-shift the phase is gradually shifted until a High pulse on PSDONE indicates that the operation is complete. There are no glitches or sporadic changes during the operation. From the start to the end of the operation the phase is shifted in a continuous analog manner.

PSINCDEC – Phase-Shift Increment/Decrement Control

This input signal synchronously indicates if the dynamic phase shift is an increment or decrement operation (positive or negative phase-shift). PSENCDEC is asserted High for increment and Low for decrement. There is no phase-shift overflow associated with the dynamic phase shift operation. If more 360° or more are shifted, then the phase will simply wrap around starting at the original phase.

CLKOUT[0:6] – Output Clocks

These user-configurable clock outputs (0 through 6) can be divided versions of the VCO phase outputs (user controllable) from 1 (bypassed) to 128. The input clock and output clocks can be phase aligned. For possible configuration see MMCM Use Models. CLKOUT0 can be used in fractional divide mode. All CLKOUT outputs can be used in non-fractional mode to provide a static or dynamic phase shift. See Static Phase Shift Mode for more information.

CLKOUT[0:3]B – Inverted Output Clocks

Inverted (180° phase shift) of CLKOUT[0:3].

CLKINSTOPPED – Input Clock Status

Status pin indicating that the input clock has stopped. This signal is asserted within one clock cycle of clock stoppage. The signal is deasserted after the clock has restarted or switched to the alternate clock input and the MMCM has re-locked.

CLKFBSTOPPED – Feedback Clock Status

Status pin indicating that the feedback clock has stopped. This signal is asserted within one clock cycle of clock stoppage. The signal is deasserted after the feedback clock has restarted and the MMCM has re-locked.

LOCKED

An output from the MMCM used to indicate when the MMCM has achieved phase and frequency alignment of the reference clock and the feedback clock at the input pins. Phase alignment is within a predefined window and frequency matching within a predefined PPM range. The MMCM automatically locks after power on, no extra reset is required. LOCKED will be deasserted within one clock cycle if the input clock stops, the phase alignment is violated (e.g., input clock phase shift) or the frequency has changed. The MMCM will automatically re-lock if the clock stops or when the phase or frequency is changed.

DO[15:0] – Dynamic Reconfiguration Output Bus

The dynamic reconfiguration output bus provides MMCM data output when using dynamic reconfiguration. If DWE is inactive while DEN is active at the rising edge of DCLK, then this bus holds the content of the configuration cells addressed by DADDR. The DO bus must be captured on the rising edge of DCLK when DRDY is active.

DRDY – Dynamic Reconfiguration Ready

The dynamic reconfiguration ready output (DRDY) provides the response to the DEN signal for the MMCMs dynamic reconfiguration feature. This signal indicates that a DEN/DCLK operation has completed.

PSDONE – Phase Shift Done

The phase-shift done output signal is synchronous to the PSCLK. When the current phase-shift operation is completed, the PSDONE signal is asserted for one clock cycle indicating that a new phase-shift cycle can be initiated.

MMCM Attributes

Table 2-4: **MMCM Attributes**

Attribute	Type	Allowed Values	Default	Description
BANDWIDTH	String	HIGH LOW OPTIMIZED	OPTIMIZED	Specifies the MMCM programming algorithm affecting the jitter, phase margin and other characteristics of the MMCM.
CLKOUT[1:6]_DIVIDE	Integer	1 to 128	1	Specifies the amount to divide the associated CLKOUT clock output if a different frequency is desired. This number in combination with the CLKFBOUT_MULT_F and DIVCLK_DIVIDE values will determine the output frequency.
CLKOUT[0]_DIVIDE_F	Real	2.000 to 128 in increments of 0.125 or integers	1	
CLKOUT[0:6]_PHASE	Real	–360.000 to 360.000 in increments of 1/56 the F_{VCO} and/or increments depending on CLKOUT_DIVIDE.	0.0	Allows specification of the output phase relationship of the associated CLKOUT clock output in number of degrees offset (i.e., 90 indicates a 90° or ¼ cycle offset phase offset while 180 indicates a 180° offset or ½ cycle phase offset).
CLKOUT[0:6]_DUTY_CYCLE	Real	0.01 to 0.99	0.50	Specifies the Duty Cycle of the associated CLKOUT clock output in percentage (i.e., 0.50 will generate a 50% duty cycle).
CLKFBOUT_MULT_F	Real	5 to 64 (Integer values only)	5	Specifies the amount to multiply all CLKOUT clock outputs if a different frequency is desired. This number, in combination with the associated CLKOUT#_DIVIDE value and DIVCLK_DIVIDE value, will determine the output frequency.
DIVCLK_DIVIDE(1)	Integer	1 to 80	1	Specifies the division ratio for all output clocks with respect to the input clock. Effectively divides the CLKIN going into the PFD.

Table 2-4: **MMCM Attributes *(Cont'd)***

Attribute	Type	Allowed Values	Default	Description
CLKFBOUT_PHASE	Real	0.00 to 360.00	0.0	Specifies the phase offset in degrees of the clock feedback output. Shifting the feedback clock results in a negative phase shift of all output clocks to the MMCM.
REF_JITTER1 REF_JITTER2	Real	0.000 to 0.999	0.100	Allows specification of the expected jitter on the reference clock in order to better optimize MMCM performance. A bandwidth setting of OPTIMIZED will attempt to choose the best parameter for input clocking when unknown. If known, then the value provided should be specified in terms of the UI percentage (the maximum peak to peak value) of the expected jitter on the input clock.
CLKIN1_PERIOD	Real	1.25 to 100	0.000	Specifies the input period in ns to the MMCM CLKIN1 input. Resolution is down to the ps. This information is mandatory and must be supplied.
CLKIN2_PERIOD	Real	1.25 to 100	0.000	Specifies the input period in ns to the MMCM CLKIN2 input. Resolution is down to the ps. This information is mandatory and must be supplied.
CLKFBOUT_USE_FINE_PS	Boolean	FALSE, TRUE	FALSE	CLKFBOUT counter variable fine phase shift enable.
CLKOUT0_USE_FINE_PS	Boolean	FALSE, TRUE	FALSE	CLKOUT0 counter variable fine phase shift enable. CLKOUT0_DIVIDE must be an integer and therefore fractional divide is not allowed.
CLKOUT[1:6]_USE_FINE_PS	Boolean	FALSE, TRUE	FALSE	CLKOUT[1:6] variable fine phase shift enable.
STARTUP_WAIT	Boolean	FALSE	FALSE	Must always be set to FALSE.
CLOCK_HOLD	Boolean	FALSE, TRUE	FALSE	When TRUE, holds the VCO frequency close to the frequency prior to losing CLKIN.
CLKOUT4_CASCADE	Boolean	FALSE, TRUE	FALSE	Cascades the output divider (counter) CLKOUT6 into the input of the CLKOUT4 divider for an output clock divider that is greater than 128, effectively providing a total divide value of 16,384.

Table 2-4: **MMCM Attributes *(Cont'd)***

Attribute	Type	Allowed Values	Default	Description
COMPENSATION	Boolean	ZHOLD[2], CASCADE, EXTERNAL, INTERNAL, BUF_IN	ZHOLD	Clock input compensation. Must be set to ZHOLD. Defines how the MMCM feedback is configured. ZHOLD: Indicates the MMCM is configured to provide a negative hold time at the I/O registers. CASCADE: Indicates the cascading of two MMCMs. EXTERNAL: Indicates a network external to the FPGA is being compensated. INTERNAL: Indicates the MMCM is using its own internal feedback path so no delay is being compensated. BUF_IN: Indicates that the configuration does not match with the other compensation modes and no delay will be compensated. This is the case if a clock input is driven by a BUFG/BUFH/BUFR or GTX/GTH.

Notes:

1. The DIVCLK_DIVIDE values of 3 and 4 cannot be used if the MMCM input clock (CLKIN) frequency is greater than 315 MHz. Multiplying up (doubling) both the DIVCLK_DIVIDE and CLKFBOUT_MULT attribute values yield the same result for the VCO and CLKOUT frequencies.
2. The COMPENSATION attribute values are documented for informational purpose only. The ISE® software tools automatically select the appropriate compensation based on circuit topology. Do not manually select a compensation value, leave the attribute at the default value.

MMCM Clock Input Signals

The MMCM clock source can come from several sources:

- IBUFG - Global clock input buffer, the MMCM will compensate the delay of this path. IBUFG represents a global clock pin or a clock-capable clock pin in the same region.
- BUFGCTRL or BUFG - Internal global clock buffer, the MMCM will not compensate the delay of this path.
- IBUF - Not recommended since the MMCM can not compensate for the delay of the general route. An IBUF clock input must route to a BUFG before routing to a MMCM.
- BUFR - Regional clock input buffer, the MMCM will not compensate the delay of this path.
- GTX/GTH - These serial transceivers can directly connect to the MMCM in the same clocking region.

Counter Control

The MMCM output counters provide a wide variety of synthesized clocks using a combination of DIVIDE, DUTY_CYCLE, and PHASE. Figure 2-6 illustrates how the counter settings impact the counter output.

The top waveform represents the output from the VCO.

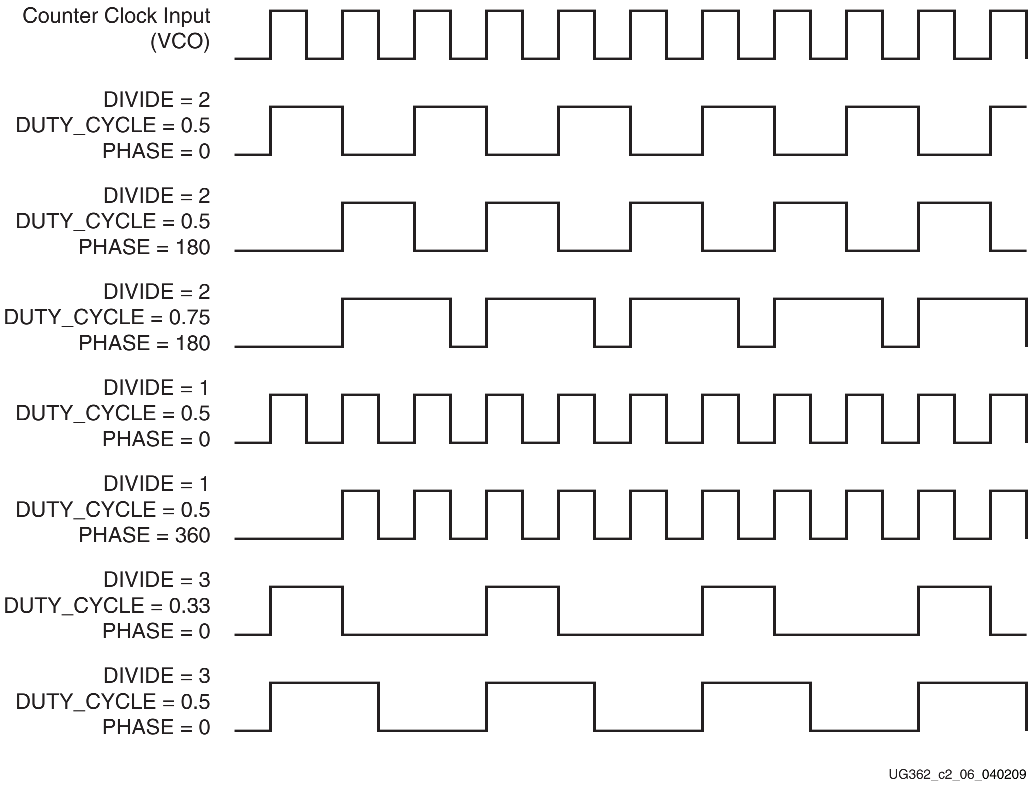


Figure 2-6: **Output Counter Clock Synthesis Examples**

Detailed VCO and Output Counter Waveforms

Figure 2-7 shows the eight VCO phase outputs and four different counter outputs. Each VCO phase is shown with the appropriate start-up sequence. The phase relationship and start-up sequence are guaranteed to insure the correct phase is maintained. This means the rising edge of the 0° phase will happen before the rising edge of the 45° phase. The O0 counter is programmed to do a simple divide by two with the 0° phase tap as the reference clock. The O1 counter is programmed to do a simple divide by two but uses the 180° phase tap from the VCO. This counter setting could be used to generate a clock for a DDR interface where the reference clock is edge aligned to the data transition. The O2 counter is programmed to do a divide by three. The O3 output has the same programming as the O2 output except the phase is set for a one cycle delay. Phase shifts greater than one VCO period are possible.

If the MMCM is configured to provide a certain phase relationship and the input frequency is changed, then this phase relationship is also changed since the VCO frequency changes and therefore the absolute shift in picoseconds will change. This aspect must be considered when designing with the MMCM. When an important aspect of the design is to maintain a certain phase relationship amongst various clock outputs, (e.g., CLK and CLK90) then this relationship will be maintained regardless of the input frequency.

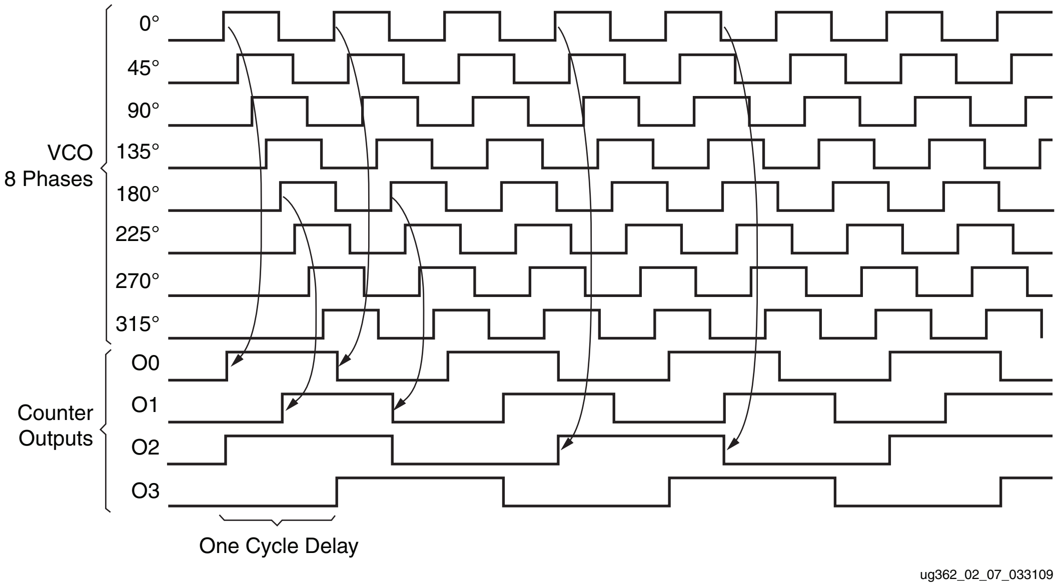


Figure 2-7: **Selecting VCO Phases**

All "O" counters can be equivalent, anything O0 can do, O1 can do. In Virtex-6 devices, the O0 counter has the additional capability to be used in fractional divide mode. The MMCM outputs are flexible when connecting to the global clock network since they are identical. In most cases, this level of detail is imperceptible to the designer as the software and Clocking Wizard determines the proper settings through the MMCM attributes and Wizard inputs.

Reference Clock Switching

The MMCM reference clock can be dynamically switched by using the CLKINSEL pin. The switching is done asynchronously. Once the clock switches, the MMCM is likely to lose LOCKED. The MMCM clock MUX switching is shown in Figure 2-8. The CLKINSEL signal directly controls the MUX. No synchronization logic is present. The MMCM must be held in RESET during clock switchover to prevent a runt pulse from violating the minimum pulse width during the switchover. Otherwise, the MMCM automatically regains LOCK.

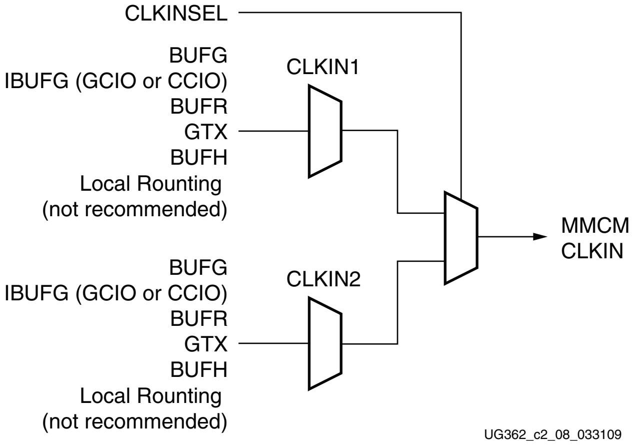


Figure 2-8: **Input Clock Switching**

Missing Input Clock or Feedback Clock

When the input clock or feedback clock is lost, the MMCM optionally maintains the VCO frequency at a slightly lower frequency than the original. The CLKINSTOPPED or CLKFBSTOPPED status signal is asserted. The MMCM deasserts the LOCKED signal. After the clock returns, the CLKSTOPPED signal is deasserted and a RESET must be applied if the MMCM VCO frequency was not maintained using the HOLD attribute. Otherwise, if the VCO frequency is held, the MMCM will relock without applying RESET.

MMCM Use Models

There are several methods to design with the MMCM. The Clocking Wizard in ISE software can assist with generating the various MMCM parameters. Additionally, the MMCM can be manually instantiated as a component. It is also possible for the MMCM to be merge with an IP core. The IP core would contain and manage the MMCM.

Clock Network Deskew

One of the predominant uses of the MMCM is for clock network deskew. Figure 2-9 shows the MMCM in this mode. The clock output from one of the CLKOUT counters is used to drive logic within the fabric and/or the I/Os. The feedback counter is used to control the exact phase relationship between the input clock and the output clock (if, for example a 90° phase shift is required). The associated clock waveforms are shown to the right for the case where the input clock and output clock need to be phase aligned. This configuration is the most flexible, but it does require two global clock networks (Figure 2-9).

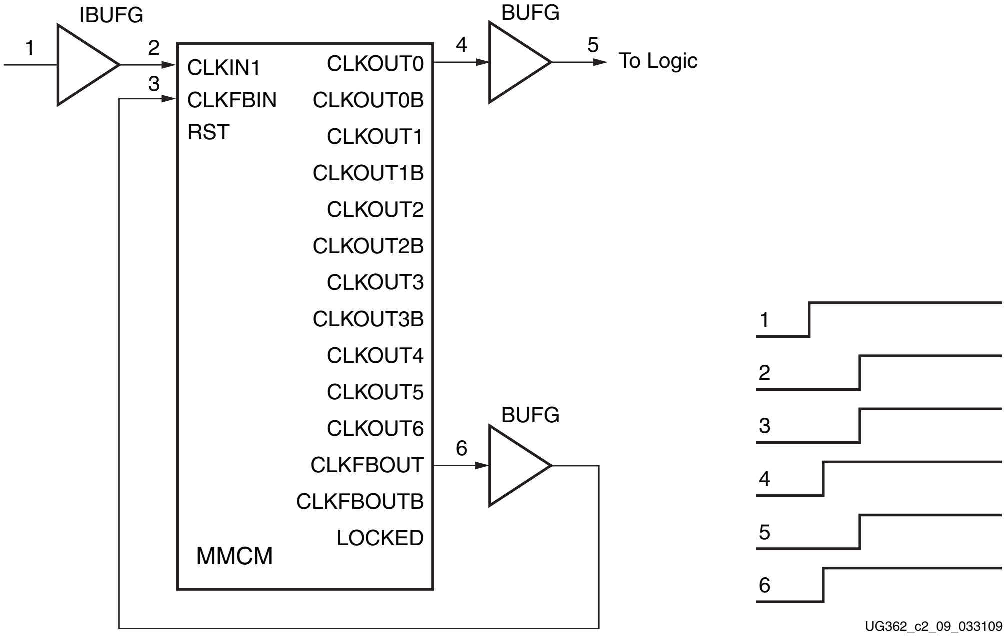


Figure 2-9: **Clock Deskew Using Two BUFGs**

There are certain restrictions on implementing the feedback. The CLKFBOUT output can be used to provide the feedback clock signal. The fundamental restriction is that both input frequencies to the PFD must be identical. Therefore, the following relationship must be met:

$$\frac{f_{IN}}{D} = f_{FB} = \frac{f_{VCO}}{M}$$ *Equation 2-9*

As an example, if f_{IN} is 166 MHz, D = 1, M = 6, and O = 2, then VCO is 996 MHz and the clock output frequency is 498 MHz. Since the M value in the feedback path is 6, both input frequencies at the PFD are 166 MHz.

In another more complex scenario has an input frequency of 66.66 MHz and D = 2, M = 30, and O = 4. The VCO frequency in this case is 1000 MHz and the CLKOUT output frequency is 250 MHz. Therefore, the feedback frequency at the PFD is 1000/30 or 33.33 MHz, matching the 66.66 MHz/2 input clock frequency at the PFD.

MMCM with Internal Feedback

The MMCM feedback can be internal to the MMCM when the MMCM is used as a synthesizer or jitter filter and there is no required phase relationship between the MMCM input clock and the MMCM output clock. The MMCM performance should increase since the feedback clock is not subjected to noise on the core supply since it never passes through a block powered by this supply. Of course, noise introduced on the CLKIN signal and the BUFG will still be present (Figure 2-10).

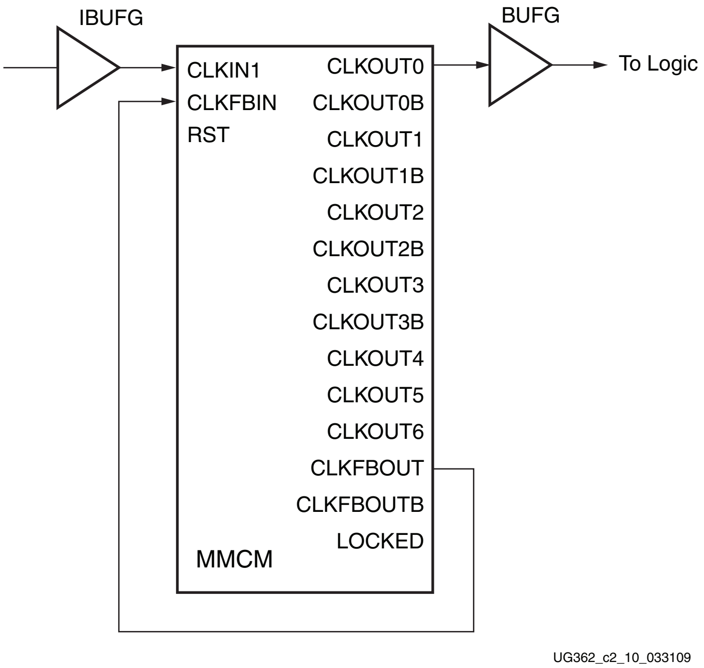
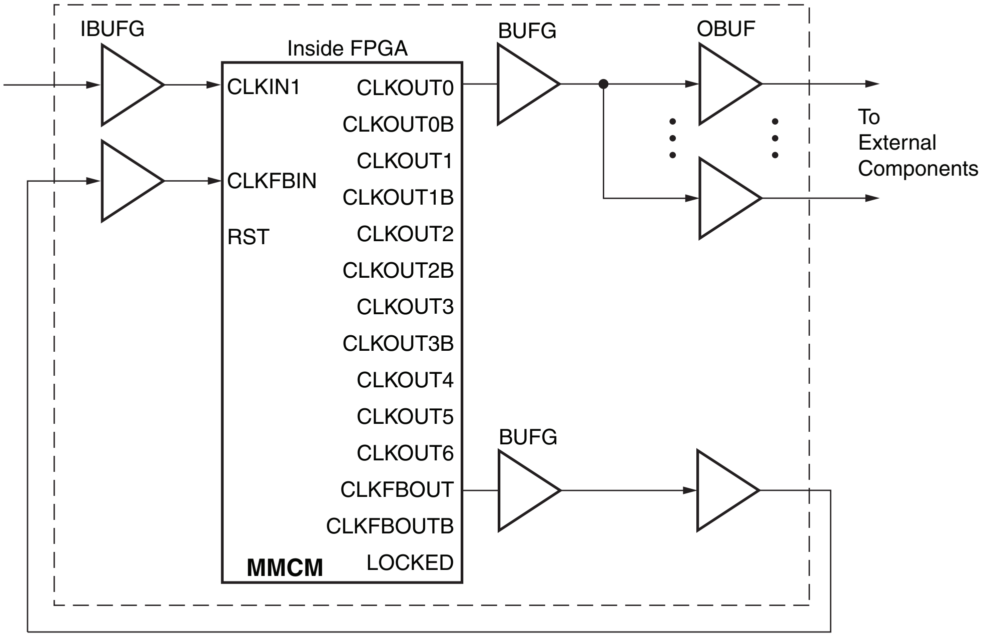


Figure 2-10: **MMCM with Internal Feedback**

Zero Delay Buffer

The MMCM can also be used to generate a zero delay buffer clock. A zero delay buffer can be useful for applications where there is a single clock signal fan out to multiple destinations with a low skew between them. This configuration is shown in the Figure 2-11. Here, the feedback signal drives off chip and the board trace feedback is designed to match the trace to the external components. In this configuration, it is assumed that the clock edges are aligned at the input of the FPGA and the input of the external component. The input clock buffers for CLKIN and CLKFBIN must be the same type. Both must either be a clock capable input (CC) in the same clocking region or both must be a global clock input (GC). There will be a limitation on the maximum delay allowed in the feedback path.



ug362_c2_11_033109

Figure 2-11: **Zero Delay Buffer**

In some cases precise alignment will not occur because of the difference in loading between the input capacitance of the external component and the feedback path capacitance of the FPGA. For example, the external components can have an input capacitance on 1 pF to 4 pF while the FPGA has an input capacitance of around 8 pF. There is a difference in the signal slope, which is basically skew. Designers need to be aware of this effect to ensure timing.

MMCM to MMCM Connection

The MMCM can be directly cascaded within a CMT to allow generation of a greater range of clock frequencies. The frequency range restrictions still apply. Equation 2-10 shows the relationship between the final output frequency and the input frequency and counter settings of the two MMCMs (Figure 2-12 and Figure 2-13). The phase relationship between the output clock of the second MMCM and the input clock is undefined. To cascade MMCMs, route the output of the first MMCM to the CLKIN pin of the second MMCM. This path provides the lowest device jitter. Cascading using the inverted CLKOUTxB outputs is not available.

$$f_{OUTMMCM2} = f_{OUTMMCM1}\frac{M_{MMCM2}}{D_{MMCM2} \times O_{MMCM2}} = f_{IN}\frac{M_{MMCM1}}{D_{MMCM1} \times O_{MMCM1}} \times \frac{M_{MMCM2}}{D_{MMCM2} \times O_{MMCM2}} \qquad \textit{Equation 2-10}$$

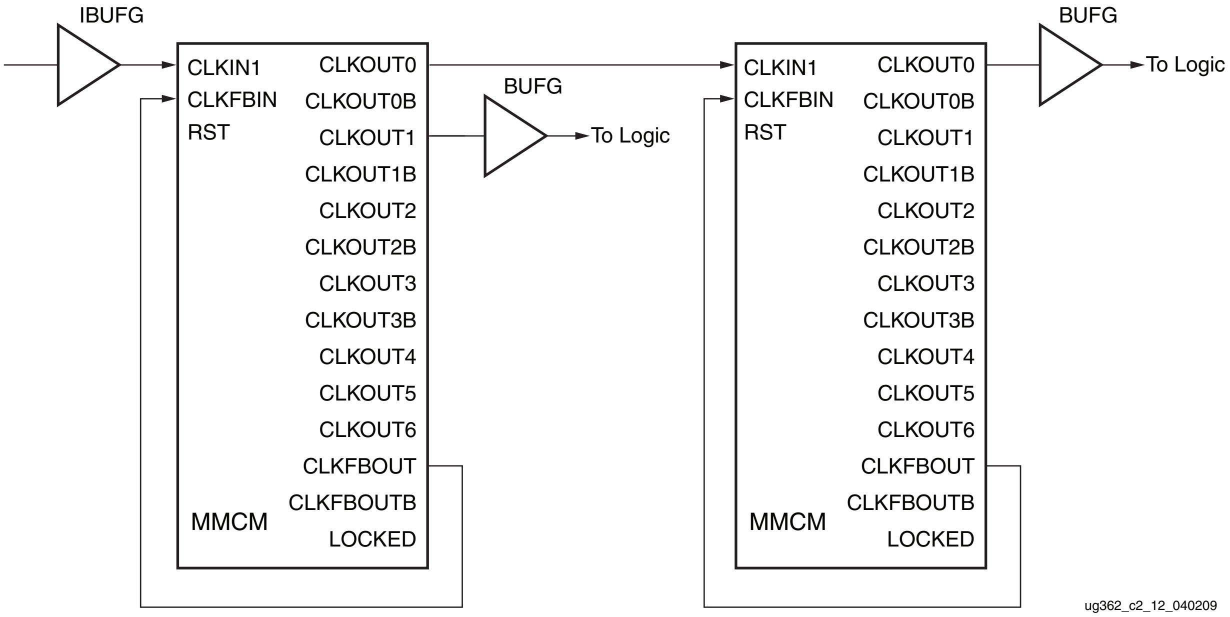


Figure 2-12: **Cascading Two MMCMs Without Any Clock Alignment**

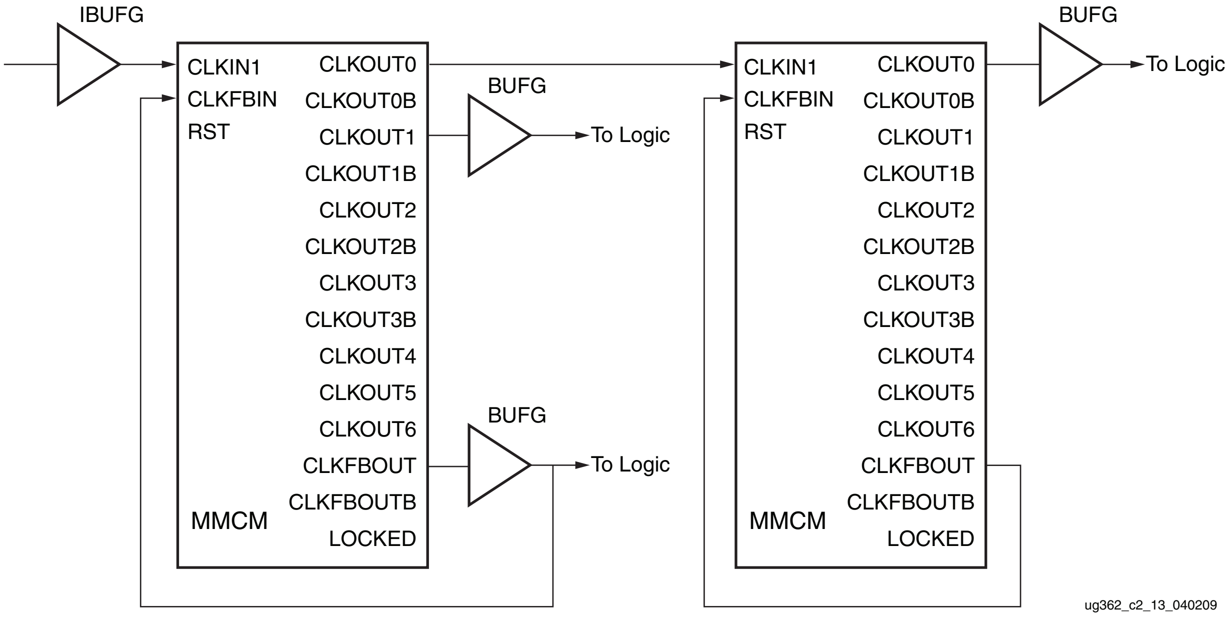


Figure 2-13: **Cascading Two MMCMs With Clock Alignment**

MMCM Application Example

The following MMCM attribute settings result in a wide variety of synthesized clocks:

```
CLKOUT0_PHASE = 0;
CLKOUT0_DUTY_CYCLE = 0.5;
CLKOUT0_DIVIDE = 2;
CLKOUT1_PHASE = 90;
CLKOUT1_DUTY_CYCLE = 0.5;
CLKOUT1_DIVIDE = 2;
CLKOUT2_PHASE = 0;
CLKOUT2_DUTY_CYCLE = 0.25;
CLKOUT2_DIVIDE = 4;
CLKOUT3_PHASE = 90;
CLKOUT3_DUTY_CYCLE = 0.5;
CLKOUT3_DIVIDE = 8;
CLKOUT4_PHASE = 0;
CLKOUT4_DUTY_CYCLE = 0.5;
CLKOUT4_DIVIDE = 8;
CLKOUT5_PHASE = 135;
CLKOUT5_DUTY_CYCLE = 0.5;
CLKOUT5_DIVIDE = 8;
CLKFBOUT_PHASE = 0;
CLKFBOUT_MULT_F = 8;
DIVCLK_DIVIDE = 1;
CLKIN1_PERIOD = 10.0;
```

Figure 2-14 displays the resulting waveforms.

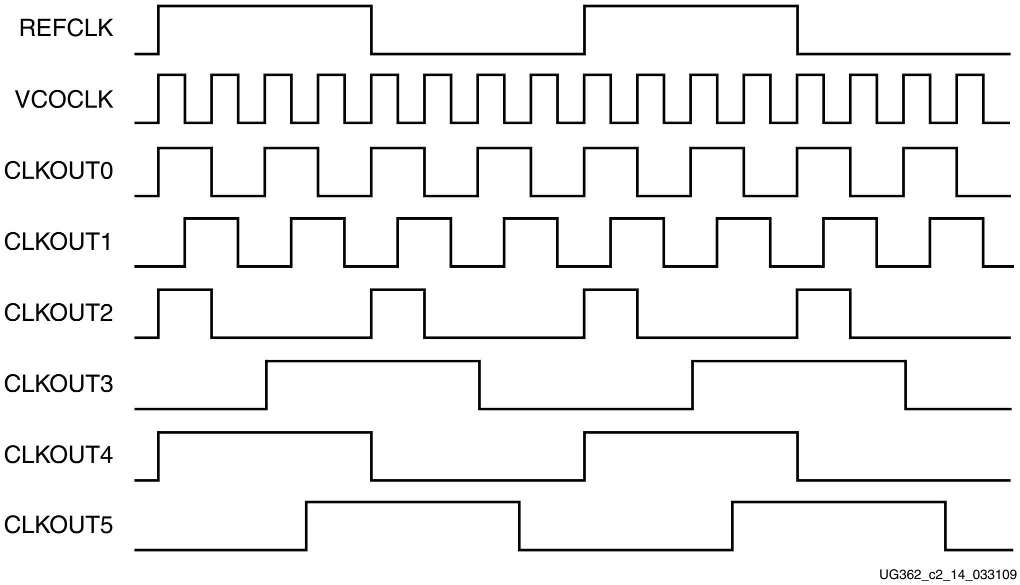


Figure 2-14: **Example Waveform**

Dynamic Reconfiguration Port

Details of the supported Virtex-6 FPGAs MMCM DRP operations are described in XAPP878, *MMCM Dynamic Reconfiguration*.

Appendix A

Summary of Clocking Connectivity

Summary

Table A-1 summarizes the Virtex-6 FPGAs clocking connectivity.

Table A-1: **Summary of Clocking Connectivity**

Clocking Function or Pin	Directly Driven By	Used to Directly Drive
Global Clock Inputs (GCs): There are 8 pin/pairs per device and two GCs per bank in banks 24, 25, 34, 35.	External Clock	GCs are located in the inner I/O columns. GCs drive all MMCMs and all 32 BUFGs.
Multi-region Clock Capable I/O (MRCC) There are two pin/pairs in each bank.	External Clock	MRCCs that are located in the outer left/right column I/Os within the same region: • Two matching multi-region BUFIOs • Four BUFRs Additionally, MRCCs are located in the center right/left column I/O: • Two MMCMs (one CMT) in same region • 16 BUFGs within the same half top/bottom
Single-region Clock Capable I/O (SRCC) There are two pin pairs in each bank.	External Clock	SRCCs that are located in the left column I/Os, within the same region: • Two matching single region BUFIO • Four BUFRs Additionally, SRCCs are used for the center right/left column I/O: • Two MMCMs (one CMT) in same region SRCCs within the same half top/bottom drive: • 16 BUFGs

Table A-1: **Summary of Clocking Connectivity *(Cont'd)***

Clocking Function or Pin	Directly Driven By	Used to Directly Drive
BUFIO	BUFIOs in the same clocking region: • MRCCs • SRCCs (dedicated 1:1) • MMCM.CLKOUT0–MMCM.CLKOUT3 (High performance path, inner I/O columns only)	When used within the same clocking region, BUFIOs: • ILOGIC.clk • ILOGIC.clkb • OLOGIC.clk • OLOGIC.clkb • OLOGIC.oclk • OLOGIC.oclkb The two multi-region BUFIOs driven by MRCCs and MMCM.CLKOUT0–MMCM.CLKOUT3 can drive loads in the regions above and below.
BUFR	Within the same clocking region BUFRs are driven by: • MRCC • SRCC • MMCM.CLKOUT0-MMCM.CLKOUT3 (High performance path, inner I/O columns only) • BUFG (not recommended)	When used within the same clocking region, BUFRs drive: • MMCM • Any clocking point in the same clock region the BUFG can drive • BUFR loads in the region above and below BUFRs can drive BUFGs can drive within the same half top/bottom.
BUFG	• Any GC • BUFG Within the same top/bottom half BUFGs are driven by: • SRCCs • MRCCs • MMCMs • GTs • BUFRs	• Any MMCM • GTs • Adjacent BUFGs in same top/bottom half
BUFH	• Any BUFG • Interconnect (not recommended) Within the same horizontal region, BUFHs are driven by: • MMCM • MRCC • SRCC	• Any MMCM • GTs • Adjacent BUFGs in same top/bottom half • Any clocking point in the same clocking region the BUFG can drive

Table A-1: **Summary of Clocking Connectivity *(Cont'd)***

Clocking Function or Pin	Directly Driven By	Used to Directly Drive
GT Transceiver Clocks	• Any BUFG Within the same region, GTs are driven by: • MMCM • A BUFR within the clock region, the region above or below	• BUFG within the same half top/bottom Within the same region GTs can drive: • MMCM • BUFR
MMCM	• Any global clock input pin (GC) • BUFG Within the same horizontal clock regions MMCMs are driven by: • SRCC • MRCC • GTs • The other MMCM in the same CMT (cascade) • Only a BUFR within the same clock region, the region above and below	• Any BUFG in same top/bottom half Within the same horizontal clock regions MMCMs can drive: • BUFIO using the performance path • BUFR using the performance path • GTs • The other MMCM in the same CMT (cascade)