Using Global Clock Networks

Introduction

Virtex-II devices support very high frequency designs and require low-skew advanced clock distribution. With device density up to 10 million system gates, numerous global clocks are necessary in most designs. Therefore, to provide a uniform and portable solution (soft-IP), all Virtex-II devices from XC2V40 to XC2V10000 have 16 global clock buffers and support 16 global clock domains. Up to eight clocks can be used anywhere in the device by the synchronous logic elements (that is, registers, 18Kb block RAM, pipeline multipliers) and the IOBs. The software tools place and route automatically the global clocks, up to eight single clocks.

If the design uses between 8 and 16 clocks, it must be partitioned into quadrants, with up to 8 clocks per quadrant. If more than 16 clocks are required, the backbone (24 horizontal and vertical long lines routing resources) can be used as additional clock network.

In addition to clock distribution, the 16 clock buffers are also "glitch-free" synchronous 2:1 multiplexers. These multiplexers are capable of switching between two asynchronous (or synchronous) clocks at any time. No particular phase relations between the two clocks are needed. The clock multiplexers can also be configured as a global clock buffer with a clock enable. The clock can be stopped High or Low at the clock buffer output.

Clock Distribution Resources

The various resources available to manage and distribute the clocks include:

- 16 clock pads that can be used as regular user I/Os if not used as clock inputs. The 16 clock pads can be configured for any I/O standard, including differential standards (for example, LVDS, LVPECL, and so forth).
- 16 "IBUFG" elements that represent the clock inputs in a VHDL or Verilog design.
- 8 "IBUFGDS" elements (that is, attributes LVPECL_33, LVDS_25, LVDS_33, LDT_25, or ULVDS_25) that represent the differential clock input pairs in a VHDL or Verilog design. Each IBUFGDS replaces two IBUFG elements.
- 4 to 12 DCMs, depending on the device size, to de-skew and generate the clocks. For more information on DCMs, see "Using the Digital Clock Manager (DCM)" on page 161.
- 16 "BUFGMUX" elements that can consist of up to 16 global clock buffers (BUFG), global clock buffers with a clock enable, or global clock multiplexers.

Figure 2-1 illustrates the placement of these clock resources in Virtex-II devices (the XC2V250 through the XC2V2000) that have eight DCMs.

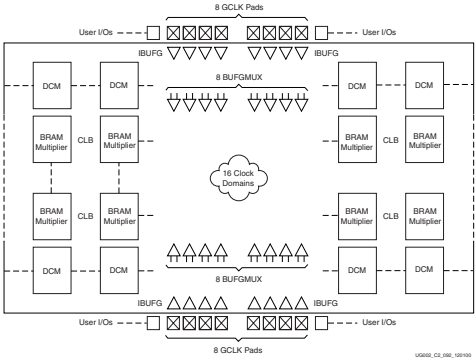

Figure 2-1: **Clock Resources in Virtex-II Devices**

The simple scheme to distribute an external clock in the device is to implement a clock pad with an IBUFG input buffer connected to a BUFG global buffer, as shown in Figure 2-2.

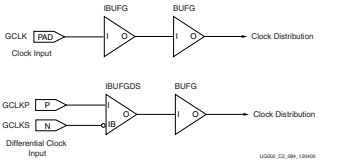

Figure 2-2: **Simple Clock Distribution**

Major synthesis tools automatically infer the IBUFG and BUFG when the corresponding input signal is used as a clock in the VHDL or Verilog code.

A high frequency or adapted (frequency, phase,…) clock distribution with low skew is implemented through a DCM between the output of the IBUFG and the input of the BUFG, as shown in Figure 2-3. "Using the Digital Clock Manager (DCM)" on page 161 provides details about the DCM and the conditions needed to use it, for example, the need for a clock feedback (DLL capability).

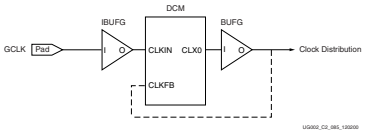

Figure 2-3: **Clock Distribution with DCM**

Clock distribution from internal sources is also possible with a BUFG only or a DCM, as shown in Figure 2-4.

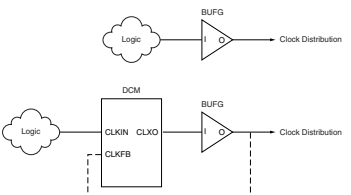

Figure 2-4: **Internal Logic Driving Clock Distribution**

Global Clock Inputs

The clock buffer inputs are fed either by one of the 16 clock pads (refer to the *Virtex-II Data Sheet*), by the outputs of the DCM, or by local interconnect. Each clock buffer is a synchronous "glitch-free" 2:1 multiplexer with two clock inputs and one select input. Internal logic (or alternatively a regular IOB) can feed the clock inputs. Any internal or external signal can drive the select input or clock enable input.

The possible inputs driving a global clock buffer or multiplexer are summarized in Table 2-1.

Table 2-1: **Inputs Driving Global Clock Buffers or DCMs**

Source	Destination				
	BUFG(I) or BUFGCE(I)	**BUFGCE (CE)**	**BUFGMUX (I0 or I1)**	**BUFGMUX (S)**	**DCM (CLKIN)**
External Clock via IBUFG(O)	Dedicated in same quadrant[1]	NA	Dedicated in same quadrant[1]	NA	Same quadrant
DCM Clock Outputs	Same edge (top or bottom)[2]	NA	Same edge (top or bottom)[2]	NA	General interconnect[3]
Internal Logic	General interconnect	General interconnect	General interconnect	General interconnect	General interconnect[3]
User I/O Pad via IBUF(O) (not IBUFG)	General interconnect	General interconnect	General interconnect	General interconnect	General interconnect[3]
BUFG(O)	NA	NA	NA	NA	Global clock net[3]
BUFGMUX(O)	NA	NA	General interconnect	NA	Global clock net[3]

Notes:
1. Not all IBUFGs in the quadrant have a dedicated connection to a specific BUFG. Others would require general interconnect to be hooked up.
2. Same edge (top or bottom) enables use of dedicated routing resources.
3. Pad to DCM input skew is not compensated.

All BUFG (BUFGCE, BUFGMUX) outputs are available at the quadrant boundaries.

The output of the global clock buffer can be routed to non-clock pins.

Primary and Secondary Global Multiplexers

Each global clock buffer is a self-synchronizing circuit called a clock multiplexer.

The 16 global clock buffers or multiplexers are divided as follows:

- Eight primary clock multiplexers
- Eight secondary clock multiplexers

There is no difference in the hardware between a primary and a secondary clock multiplexer. However, some restrictions apply to primary/secondary multiplexers, because they share input connections as well as access to a quadrant.

Each Virtex-II device is divided into four quadrants: North-West, South-West, North-East, and South-East. Each quadrant has two primary and two secondary clock multiplexers. The clock multiplexers are indexed 0 to 7, with one primary and one secondary for each index, alternating on the top and on the bottom (i.e., clock multiplexer "0P" at the bottom is facing clock multiplexer "0S" at the top).

In each device, the eight top/bottom clock multiplexers are divided into four primary and four secondary, indexed 0 to 7, as shown in Figure 2-5.

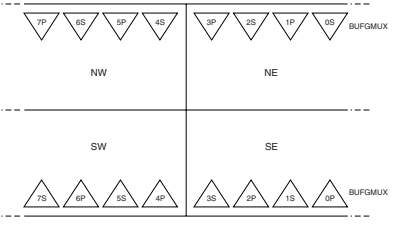

Figure 2-5: **Primary and Secondary Clock Multiplexer Locations**

Primary/Secondary: Rule 1

Considering two "facing" clock multiplexers (BUFG#P and BUFG#S), either one of these clock outputs can enter any quadrant of the chip to drive a clock within that quadrant, as shown in Figure 2-6. Note that the clock multiplexers "0P" and "0S" compete for quadrant access. Thus, BUFGOP output cannot be used in the same quadrant as BUFGOS.

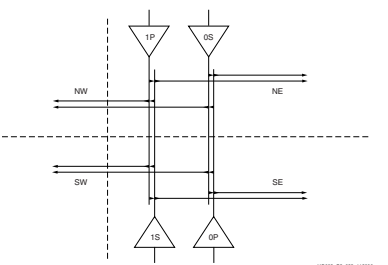

Figure 2-6: **Facing BUFG#P and BUFG#S Connections**

Primary/Secondary: Rule 2

In a BUFGCE or BUFGMUX configuration, shared inputs have to be considered. Two adjacent clock multiplexers share two inputs, as shown in Figure 2-7. The clock multiplexer "1P" and "0S" have common I0/I1 and I1/I0 inputs.

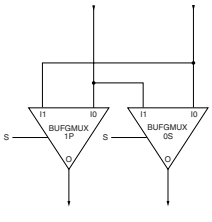

Figure 2-7: **Clock Multiplexer Pair Sharing Clock Multiplexer Inputs**

Table 2-2 lists the clock multiplexer pairs in any Virtex-II device. The primary multiplexer inputs I1/I0 are common with the corresponding secondary multiplexer inputs I0/I1 (i.e., Primary I1 input is common with secondary I0 input, and primary I0 input is common with secondary I1 input).

Table 2-2: **Top Clock Multiplexer Pairs**

Primary I1/I0	1P	3P	5P	7P
Secondary I0/I1	0S	2S	4S	6S

Table 2-3: **Bottom Clock Multiplexer Pairs**

Primary I1/I0	0P	2P	4P	6P
Secondary I0/I1	1S	3S	5S	7S

Primary/Secondary Usage

For up to eight global clocks, it is safe to use the eight primary global multiplexers (1P, 3P, 5P, 7P on the top and 0P, 2P, 4P, 6P on the bottom). Because of the shared inputs, a maximum of eight independent global clock multiplexers can be used in a design, as shown in Figure 2-8.

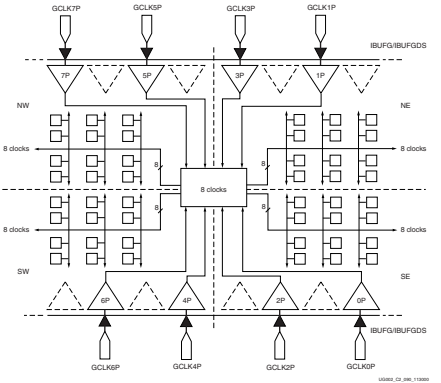

Figure 2-8: **Eight Global Clocks Design**

DCM Clocks

The four clock pins (IBUFG) in a quadrant can feed all DCMs in the same quadrant. The clock-to-out and setup times are identical for all DCMs. Up to four clock outputs per DCM can be used to drive any clock multiplexer on the same edge (top or bottom), as shown in Figure 2-9.

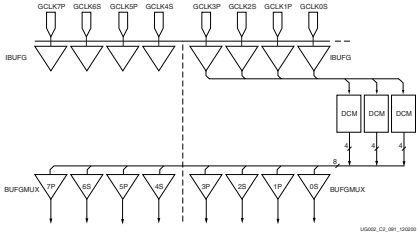

Figure 2-9: **DCM Clocks**

Clock Output

The clock distribution is based on eight clock trees per quadrant. Each clock multiplexer output is driving one global clock net. The Virtex-II device has eight dedicated low-skew clock nets. The device is divided into four quadrants (NW, NE, SW and SE) with eight global clocks available per quadrant.

Eight clock buffers are in the middle of the top edge and eight are in the middle of the bottom edge. Any of these 16 clock buffer outputs can be used in any quadrant, up to a maximum of eight clocks per quadrant, as illustrated in Figure 2-10, provided there is not a primary vs. secondary conflict.

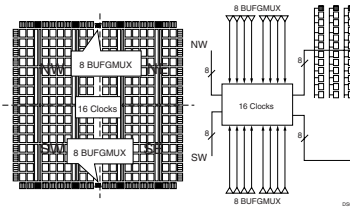

Figure 2-10: **Clock Buffer Outputs per Quadrant**

Designs with more than eight clocks must be floorplanned manually or automatically, distributing the clocks in each quadrant. As an example, a design with 16 clocks can be floorplanned as shown in Figure 2-11.

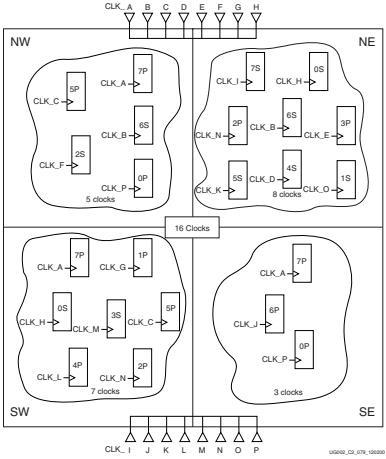

Figure 2-11: **16-Clock Floorplan**

The clock nets and clock buffers in this example are associated as shown in Table 2-4.

Table 2-4: **Clock Net Association With Clock Buffers**

Clock Net (top edge)	CLK_A	CLK_B	CLK_C	CLK_D	CLK_E	CLK_F	CLK_G	CLK_H
BUFG	7P	6S	5P	4S	3P	2S	1P	0S
Clock Net (bottom edge)	CLK_I	CLK_J	CLK_K	CLK_L	CLK_M	CLK_N	CLK_O	CLK_P
BUFG	7S	6P	5S	4P	3S	2P	1S	0P
Quadrant NW	CLK_A	CLK_B	CLK_C	–	–	CLK_F	–	CLK_P
Quadrant SW	CLK_A	–	CLK_C	CLK_L	CLK_M	CLK_N	CLK_G	CLK_H
Quadrant NE	CLK_I	CLK_B	CLK_K	CLK_D	CLK_E	CLK_N	CLK_O	CLK_H
Quadrant SE	CLK_A	CLK_J	–	–	–	–	–	CLK_P

CLK_A is used in three quadrants, and the other clocks are used in one or two quadrants, regardless of the position of the clock buffers (multiplexers), as long as they are not competing to access the same quadrant. (That is, CLK_A (BUFG7P) cannot be used in the same quadrant with CLK_I (BUFG7S). Refer to "Primary/Secondary: Rule 1" on page 146.) In other words, two buffers with the same index (0 to 7) cannot be used in the same quadrant.

Each register, block RAM, registered multiplier, or DDR register (IOB) can be connected to any of the eight clock nets available in a particular quadrant.

Note that if a global clock (primary buffer) is used in four quadrants, the corresponding secondary buffer is not available.

Power Consumption

Clock trees have been designed for low skew and low-power operation. Any unused branch is disconnected, as shown in Figure 2-12.

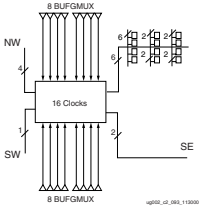

Figure 2-12: **Low-Power Clock Network**

Also available to reduce overall power consumption are the BUFGCE feature, for dynamically driving a clock tree only when the corresponding module is used, and the BUFGMUX feature, for switching from a high-frequency clock to a low-frequency clock.

The frequency synthesizer capability of the DCM can generate the low (or high) frequency clock from a single source clock, as illustrated in Figure 2-13. (See "Using the Digital Clock Manager (DCM)" on page 161).

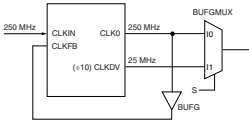

Figure 2-13: **Dynamic Power Reduction Scheme**

Library Primitives and Submodules

The primitives in Table 2-5 are available with the input, output, and control pins listed.

Table 2-5: **Clock Primitives**

Primitive	Input	Output	Control
IBUFG	I	O	–
IBUFGDS	I, IB	O	–
BUFG	I	O	–
BUFGMUX	I0, I1	O	S
BUFGMUX_1	I0, I1	O	S

Refer to "Using Single-Ended SelectI/O Resources" on page 245 for a list of the attributes available for IBUFG and Refer to "Using LVDS I/O" on page 300 for a list of the attributes available for IBUFGDS.

The submodules in Table 2-6 are available with the input, output, and control pins listed.

Table 2-6: **Clock Submodules**

Primitive	Input	Output	Control
BUFGCE	I	O	CE
BUFGCE_1	I	O	CE

Primitive Functions

IBUFG

IBUFG is an input clock buffer with one clock input and one clock output.

IBUFGDS

IBUFGDS is an input clock buffer with two clock inputs (positive and negative polarity) and one clock output.

BUFG

BUFG is a global clock buffer with one clock input and one clock output, driving a low-skew clock distribution network. The output follows the input, as shown in Figure 2-14.

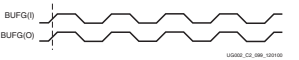

Figure 2-14: **BUFG Waveforms**

BUGMUX and BUFGMUX_1

BUFGMUX and BUFGMUX_1 are global clock buffers incorporating a smart multiplexer function that avoids output glitches or runt pulses, even when the enable signal changes asynchronously.

BUFGMUX is the preferred circuit for rising edge clocks, while BUFGMUX_1 is preferred for falling edge clocks.

Operation of the BUFGMUX Circuit

When the S input has been High for a while, the output follows input I1.
When S goes Low, the next falling edge of I1 causes the output to go Low and stay Low until after the next falling edge of I0. From then on, the output follows the I0 input.
When S goes High, the next falling edge of I0 causes the output to go Low and stay Low until after the next falling edge of I1. From then on, the output follows I1.
This means, the output stays Low while switching between the two clock domains, but it always completes the full clock period (High followed by Low) when switching between clocks. See Figure 2-15.

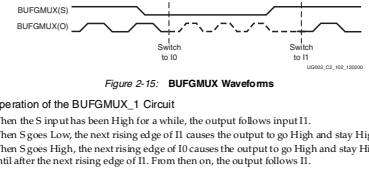

Figure 2-15: **BUFGMUX Waveforms**

Operation of the BUFGMUX_1 Circuit

When the S input has been High for a while, the output follows input I1.
When S goes Low, the next rising edge of I1 causes the output to go High and stay High
When S goes High, the next rising edge of I0 causes the output to go High and stay High until after the next rising edge of I1. From then on, the output follows I1.
This means, the output stays High while switching between the two clock domains, but it always completes the full clock period (High followed by Low) when switching between clocks. See Figure 2-16.

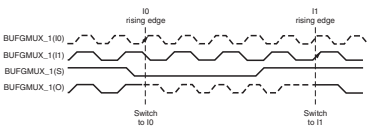

Figure 2-16: **BUFGMUX_1 Waveforms**

The default input at startup is I0. When BUFGMUX (or BUFGMUX_1) is used with DCM outputs, a second BUFG can be used for clock feedback. A buffer sharing the inputs with BUFGMUX is the preferred solution.

GSR

When a Global Set-Reset (GSR) is asserted, BUFGMUX (or BUFGMUX_1) is initialized to the default state of the I0 input.

Submodules

BUFGCE and BUFGCE_1

BUFGCE and BUFGCE_1 are submodules based on BUFGMUX and BUFGMUX_1, respectively. BUFGCE and BUFGCE_1 are global clock buffers incorporating a smart enable function that avoids output glitches or runt pulses. The select signal should meet the setup time for the clock.

BUFGCE is the preferred circuit for clocking on the rising edge, while BUFGCE_1 is preferred when clocking on the falling edge.

Operation of the BUFGCE Circuit

When the CE input is High, the output follows the input.

When CE goes Low while the input is Low, the output stays Low.

When CE goes Low while the input is High, the output continues to follow the input until it goes Low, and the output then stays Low.

When CE goes High while the input if High, the output stays Low, then starts following the input with the next rising edge.

When CE goes High while the input is Low, the output follows the input.

This means the output stays Low when the clock is disabled, but it completes the clock-High pulse when the clock is being disabled, as shown in Figure 2-17.

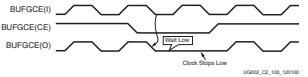

Figure 2-17: **BUFGCE Waveforms**

Operation of the BUFGCE_1 circuit

When the CE input is High, the output follows the input.

When CE goes Low while the input is High the output stays High.

When CE goes Low while the input is Low, the output continues to follow the input until it goes High, and the output then stays High.

When CE goes High while the input if Low, the output stays High, then starts following the input after the next falling edge.

When CE goes High while the input is High, the output follows the input.

This means the output stays High when the clock is disabled, but it completes the clock-Low pulse when the clock is being disabled, as shown in Figure 2-18.

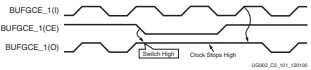

Figure 2-18: **BUFGCE_1 Waveforms**

When BUFGCE (or BUFGCE_1) is used with DCM outputs, a second BUFG can be used for clock feedback. Buffer sharing the inputs with BUFGCE is the preferred solution.

Summary

Table 2-7 shows the maximum resources available per Virtex-II device.

Table 2-7: **Resources per Virtex-II Device (from XC2V40 to XC2V10000)**

Resource	Maximum Number
Single-ended IBUFG (pads)	16
Differential IBUFGDS (pairs)	8
BUFG (Global Clock Buffer)	16
BUFGCE (or BUFGCE_1)	8
BUFGMUX (or BUFGMUX_1)	8

Characteristics

The following are characteristics of global clocks in Virtex-II devices:

- Low-skew clock distribution.
- Synchronous "glitch-free" multiplexer that avoids runt pulses.

  Switching between two clock sources can easily produce glitches without a self synchronous system like this Virtex-II global clock multiplexer.
- Setup requirement for S (or CE) versus the current clock. If the setup time is violated, the clock switches (or stops) at the next clock edge. The setup time for BUFGCE or BUFGMUX is relative to the the falling edge of the clock, and relative to the rising edge for BUFGCE_1 or BUFGMUX_1. The setup time is with reference to the selected clock.
- Two BUFGMUX (or BUFGMUX_1) resources that can be cascaded to create a 3 to 1 clock multiplexer.

Location Constraints

BUFGMUX and BUFGMUX_1 (primitives) and IBUFG (IBUFGDS) instances can have LOC properties attached to them to constrain placement. The LOC properties use the following form to constrain a clock net:

```
NET "clock_name" LOC="BUFGMUX#P/S";
```

Each clock pad (or IBUFG) has a direct connection with a specific global clock multiplexer (input I0). A placement that does not conform to this rule causes the software to send a warning.

If the clock pad (or IBUFG) has LOC properties attached, the DCM allows place and route software maximum flexibility, as compared to a direct connection to the global clock buffer (BUFG).

Secondary Clock Network

If more clocks are required, the 24 horizontal and vertical long lines in Virtex-II devices can be used to route additional clock nets. Skew is minimized by the place and route software, if the USELOWSKEWLINES constraint is attached to the net.

VHDL and Verilog Instantiation

VHDL and Verilog instantiation templates are available as examples (see "VHDL and Verilog Templates" on page 156) for all primitives and submodules.

In VHDL, each template has a component declaration section and an architecture section. Each part of the template should be inserted within the VHDL design file. The port map of the architecture section should include the design signal names.

VHDL and Verilog Templates

The following are templates for primitives:

- BUFGMUX_INST
- BUFGMUX_1_INST

The following are templates for submodules:

- BUFGCE_SUBM
- BUFGCE_1_SUBM

As examples, the BUFGMUX_INST.vhd, BUFGMUX_1_INST.vhd, BUFGCE_SUBM.vhd, and BUFGCE_1_SUBM.vhd VHDL templates are shown. In addition, the BUFGMUX_INST.v, BUFGMUX_1_INST.v, BUFGCE_1_SUBM.v, and BUFGCE_SUBM.v Verilog templates are shown.

VHDL Template

```
------------------------------------------------------------------
-- Module: BUFGMUX_INST
-- Description: VHDL instantiation template
-- Global Clock Multiplexer (Switch Low)
--
-- Device: Virtex-II Family
------------------------------------------------------------------
-- Component Declarations:
--
component BUFGMUX
  port (
  I0    : in std_logic;
        I1    : in std_logic;
        S     : in std_logic;
        O     : out std_logic
  );
end component;
--
-- Architecture  section:
--
-- Global Clock Buffer Instantiation
U_BUFGMUX: BUFGMUX
  port map (
  I0     => , -- insert clock input used when select (S) is Low
  I1     => , -- insert clock input used when select (S) is High
  S      => , -- insert Mux-Select input
  O      =>   -- insert clock output
  );
--
------------------------------------------------------------------
-- Module: BUFGMUX_1_INST
-- Description: VHDL instantiation template
-- Global Clock Multiplexer (Switch High)
--
-- Device: Virtex-II Family
------------------------------------------------------------------
-- Component Declarations:
component BUFGMUX_1
  port (
  I0    : in std_logic;
        I1    : in std_logic;
        S     : in std_logic;
        O     : out std_logic
  );
end component;
```

```
--
-- Architecture  section:
--
-- Global Clock Buffer Instantiation
U_BUFGMUX_1: BUFGMUX_1
  port map (
  I0     => , -- insert clock input used when select (S) is Low
  I1     => , -- insert clock input used when select (S) is High
  S      => , -- insert Mux-Select input
  O      =>   -- insert clock output
  );
--
-------------------------------------------------------------------
-- Module: BUFGCE_SUBM
-- Description: VHDL instantiation template
-- Global Clock Buffer with Clock Enable:
-- Input Clock Buffer to BUFGMUX - Clock disabled = Low
--
-- Device: Virtex-II Family
-------------------------------------------------------------------
library IEEE;
use IEEE.std_logic_1164.all;
--
-- pragma translate_off
library UNISIM;
use UNISIM.VCOMPONENTS.ALL;
-- pragma translate_on
--
entity BUFGCE_SUBM is
  port (
        I:  in std_logic;
        CE: in std_logic;
        O:  out std_logic
       );
end BUFGCE_SUBM;
--
architecture BUFGCE_SUBM_arch of BUFGCE_SUBM is
--
-- Component Declarations:
component BUFGMUX
  port (
  I0   : in std_logic;
        I1   : in std_logic;
        S    : in std_logic;
        O    : out std_logic
  );
end component;
--
-- signal declarations
signal GND : std_logic;
--
signal CE_B : std_logic;
--
begin
GND <= '0';
--
CE_B <= not CE;
--
-- Global Clock Buffer Instantiation
U_BUFGMUX: BUFGMUX
  port map (
  I0     => I,
```

```
  I1      => GND,
  S       => CE_B,
  O       => O
  );
--
end BUFGCE_SUBM_arch;
-----------------------------------------------------------------------
-- Module: BUFGCE_1_SUBM
-- Description: VHDL instantiation template
-- Global Clock Buffer with Clock Enable:
--  Input Clock Buffer to BUFGMUX_1 - Clock disabled = High
--
-- Device: Virtex-II Family
-----------------------------------------------------------------------
library IEEE;
use IEEE.std_logic_1164.all;
--
-- pragma translate_off
library UNISIM;
use UNISIM.VCOMPONENTS.ALL;
-- pragma translate_on
--
entity BUFGCE_1_SUBM is
  port (
         I:  in std_logic;
         CE: in std_logic;
         O:  out std_logic
       );
end BUFGCE_1_SUBM;
--
architecture BUFGCE_1_SUBM_arch of BUFGCE_1_SUBM is
--
-- Component Declarations:
component BUFGMUX_1
  port (
  I0   : in std_logic;
         I1   : in std_logic;
         S    : in std_logic;
         O    : out std_logic
  );
end component;
--
-- signal declarations
signal VCC : std_logic;
--
signal CE_B : std_logic;
--
begin
VCC <= '1';
--
CE_B <= not CE;
--
-- Global Clock Buffer Instantiation
U_BUFGMUX_1: BUFGMUX_1
  port map (
  I0      => I,
  I1      => VCC,
  S       => CE_B,
  O       => O
  );
--
end BUFGCE_1_SUBM_arch;
```

Verilog Template

```
//------------------------------------------------------------------
// Module:      BUFGMUX_INST
// Description: Verilog Instantiation Template
// Global Clock Multiplexer (Switch Low)
//
//
// Device: Virtex-II Family
//------------------------------------------------------------------
//
//BUFGMUX Instantiation
BUFGMUX  U_BUFGMUX
             (.I0(),  // insert clock input used when select(S) is Low
              .I1(),  // insert clock input used when select(S) is High
               .S(),   // insert Mux-Select input
               .O()    // insert clock output
              );
//------------------------------------------------------------------
// Module:      BUFGMUX_1_INST
// Description: Verilog Instantiation Template
// Global Clock Multiplexer (Switch High)
//
//
// Device: Virtex-II Family
//------------------------------------------------------------------
//
//BUFGMUX_1 Instantiation
BUFGMUX_1  U_BUFGMUX_1
             (.I0(),  // insert clock input used when select(S) is Low
              .I1(),  // insert clock input used when select(S) is High
               .S(),   // insert Mux-Select input
               .O()    // insert clock output
              );
//------------------------------------------------------------------
// Module:      BUFGCE_SUBM
// Description: Verilog Submodule
// Global Clock Buffer with Clock Enable:
//  Input Clock Buffer to BUFGMUX - Clock disabled = Low
//
// Device: Virtex-II Family
//------------------------------------------------------------------
module BUFGCE_SUBM (I,
                    CE,
                    O);
input   I,
        CE;

output  O;

wire GND;

assign GND = 1'b0;

BUFGMUX U_BUFGMUX
             (.I0(I),
              .I1(GND),
              .S(~CE),
              .O(O)
             );
//
endmodule
```

```
//----------------------------------------------------------------------
// Module:      BUFGCE_1_SUBM
// Description: Verilog Submodule
// Global Clock Buffer with Clock Enable:
//  Input Clock Buffer to BUFGMUX_1 - Clock disabled = High
//
//
// Device: Virtex-II Family
//----------------------------------------------------------------------

module BUFGCE_1_SUBM (I,
                      CE,
                      O);
input   I,
        CE;

output  O;

wire VCC;

assign VCC = 1'b1;

BUFGMUX_1 U_BUFGMUX_1
             (.I0(I),
              .I1(VCC),
              .S(~CE),
              .O(O)
             );
//

endmodule
```