XAPP685 (v1.3) March 4, 2005

High-Speed Clock Architecture for DDR Designs Using Local Inversion

Summary

The Virtex™-II Pro family meets the requirements of high-performance double data rate (DDR) designs. This application note provides implementation guidelines for DDR interfaces using a Digital Clock Manager (DCM) and local inversion clocking techniques for Virtex-II Pro devices. It also describes the requirements for using the corner DCMs in an XC2VP100 –6 device above 360 MHz.

Introduction

Figure 1 through Figure 3 show examples of circuits found in Virtex-II Pro designs that use rising and falling edges of a clock. These circuits all use a local inverter to clock data on the negative edge of the clock. The rest of this document refers to this type of clocking as *local inversion clocking*. Local inversion clocking is an alternative to using a complementary DCM output clock for the negative edge of the clock, for example, using CLK0 and CLK180 DCM outputs to drive a DDR flip-flop.

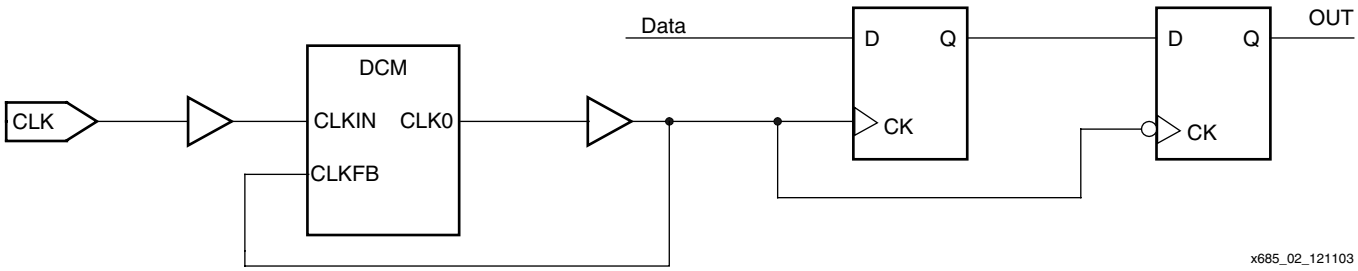


Figure 1: **Example Circuit #1: Rising to Falling Edge Clocking with Local Inversion**

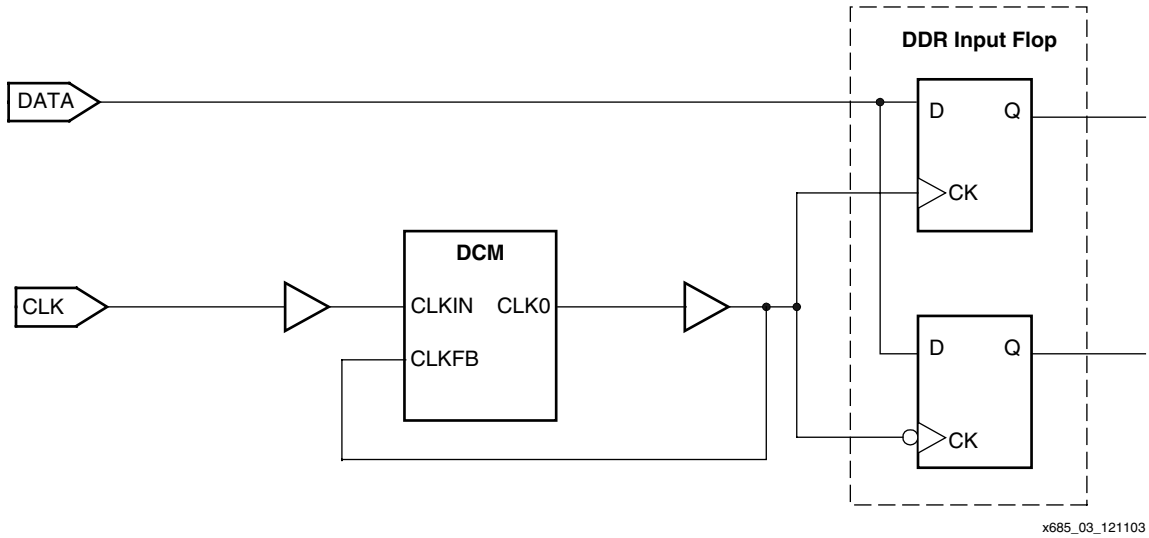


Figure 2: **Example Circuit #2: DDR Input Circuit with Local Inversion**

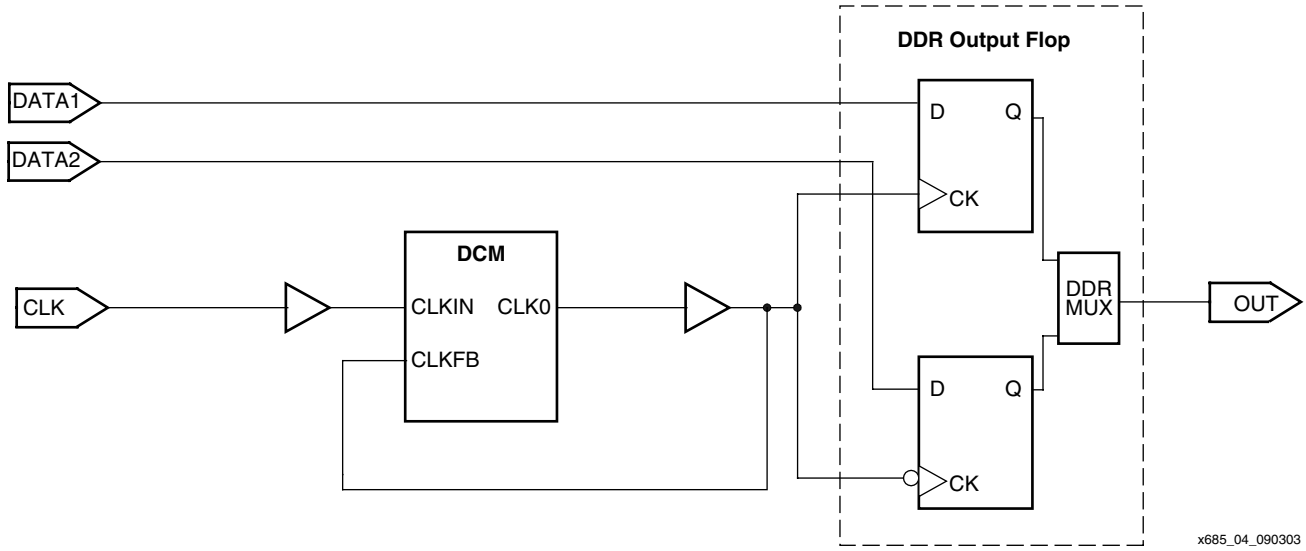


Figure 3: **Example Circuit #3: DDR Output Circuit with Local Inversion (Forwarded Clock or Data)**

In the Virtex-II Pro family, the clock net between the DCM output and the BUFGMUX (or BUFG) input must run through a macro in order to achieve high-speed DDR timing budgets when utilizing local inversion clocking. This application note describes how to implement and use these macros. Figure 4 depicts example usage of the macros.

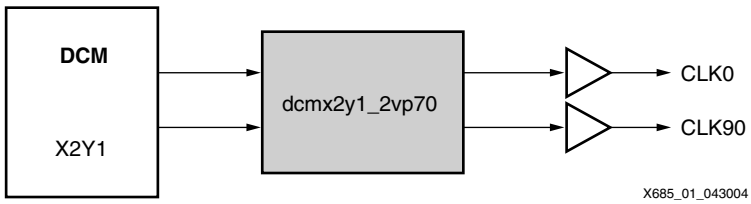


Figure 4: **Macro Usage Example**

Application Example

Figure 5 shows a usage example for the macros in a typical source-synchronous application. On the receive interface, data is captured in the input DDR flops using a forwarded clock. The receive clock is usually edge-aligned with the data, and the DCM phase-shift feature is used to center-align this clock with the data. Only one BUFGMUX is necessary for the RX interface. On the transmit interface, clock and data are transmitted with the clock center-aligned to the data. The CLK0 and CLK90 DCM output clocks are used to clock the data and generate the forwarded TX clock, respectively. Two BUFGMUXs are used for the transmit interface.

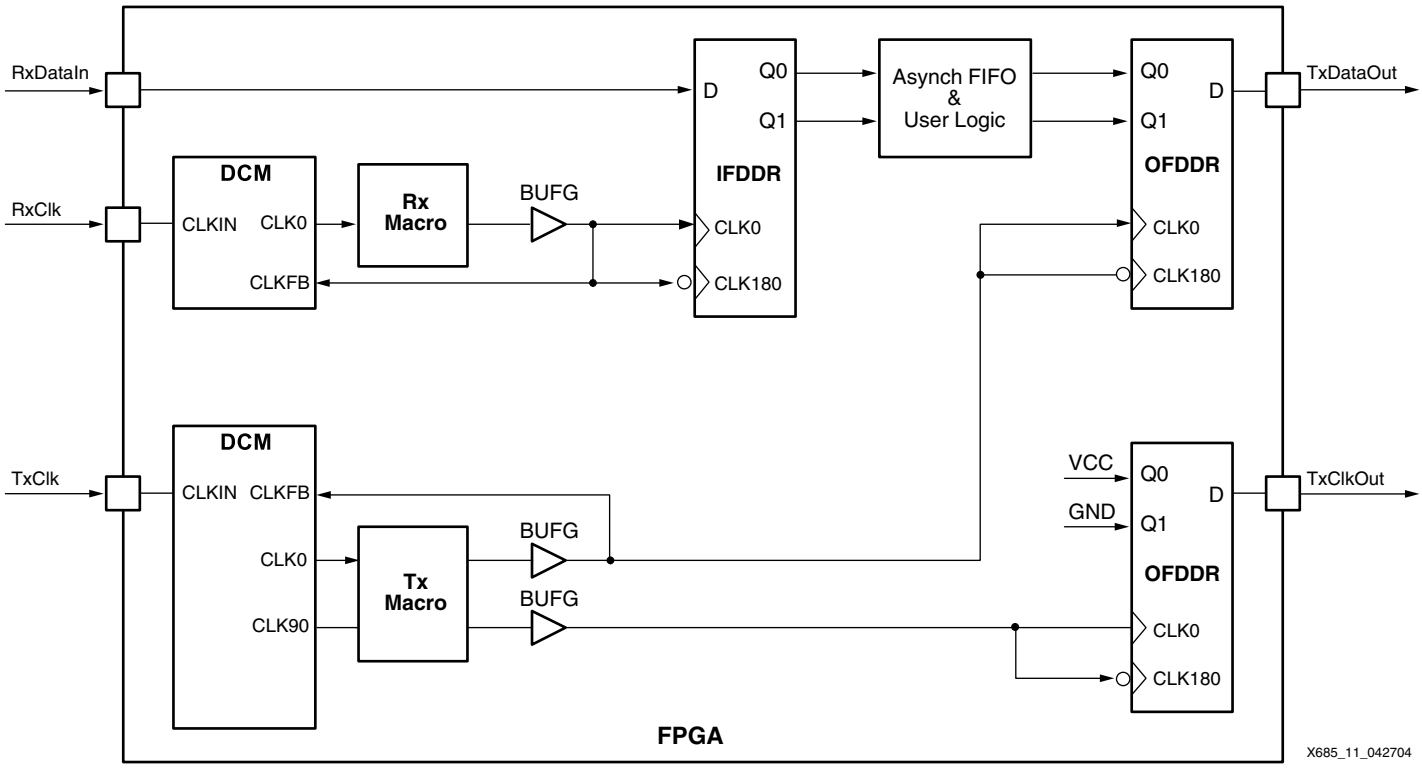


Figure 5: **Design with Local Inversion Clock Macro**

Implementation Guidelines

The macros are provided as EDIF netlists and may be downloaded as a zipped archive from http://www.xilinx.com/bvdocs/appnotes/xapp685.zip. In addition to the EDIF macros, a design example is provided in the archive. The following files are provided in the zip archive:

- `2vp2_local.zip` - ZIP file containing EDIF macros for the XC2VP2 device
- `2vp4_local.zip` - ZIP file containing EDIF macros for the XC2VP4 device
- `2vp7_local.zip` - ZIP file containing EDIF macros for the XC2VP7 device
- `2vp20_local.zip` - ZIP file containing EDIF macros for the XC2VP20 device
- `2vp30_local.zip` - ZIP file containing EDIF macros for the XC2VP30 device
- `2vp40_local.zip` - ZIP file containing EDIF macros for the XC2VP40 device
- `2vp50_local.zip` - ZIP file containing EDIF macros for the XC2VP50 device
- `2vp70_local.zip` - ZIP file containing EDIF macros for the XC2VP70 device
- `2vp100_local.zip` - ZIP file containing EDIF macros for the XC2VP100 device
- `2vpx20_local.zip` - ZIP file containing EDIF macros for the XC2VPX20 device
- `AnExample.zip` - Design example for simulation and implementation

Every DCM and device combination has an associated macro. For example, the XC2VP70 device has eight DCMs, and each DCM in the XC2VP70 has its own associated macro. The file names for the macros match the device and DCM to which they apply. For instance, the macro *dcmX1Y2_2vp70.edn* is used with the X1Y2 DCM in the XC2VP70 device.

The macros are used by instantiating them in the HDL code as "black boxes". See the design example for the correct syntax. The EDIF files must be placed in the working project directory for the tools to automatically find the macros. If placed elsewhere, the directory path must be specified in the NGDBUILD (Translate) command.

The inputs to the macros must always come from a DCM output clock, and the outputs from the macros must always connect to a BUFGMUX input on the same edge of the chip that the DCM is located. Additionally, the macro supports up to two clocks. Additional DCM output clocks may be used through the regular dedicated routes, however, the phase relationship specification CLKOUT_PHASE is not guaranteed between these clocks and the ones utilizing the macro.

Refer to the "DCM and BUFGMUX Guidelines" section for more details.

Constraint Requirements

Multiple constraint requirements *must* be followed when using the macros. These constraints can be placed in the UCF, as shown in the application example above, or embedded in the HDL:

- DCMs and BUFGMUXs must be LOC'ed
- A MAXDELAY constraint on the nets between the DCM to the MACRO of 0.450 ns
- A MAXDELAY constraint on the nets between the MACRO and the BUFGMUX of 0.755 ns
- IOBDELAY = NONE constraint on input bus flip-flops clocked with the macro

On rare occasions, PAR issues an error on the MAXDELAY timing constraints. This situation might occur because of congested routing on the routes near the BUFGMUXs. For instance, if several macros are all used on the same side of the chip. There are two simple workarounds for this situation:

1. Change the pin on the BUFGMUX, that is, I0 --> I1
2. Choose another BUFGMUX location.

Here is an example of a UCF file syntax with these constraints:

```
################################
#####    DCM CONSTRAINTS    ####
################################
# Note: must place location constraint on the DCM(s) with local routing
solutions.
INST dcm0 LOC=DCM_X0Y1;

# Note: select 2 of 8 BUFGMUXs to go with DCM location on TOP of chip
INST bufg0 LOC=BUFGMUX7P;
INST bufg1 LOC=BUFGMUX6S;

# Place maximum delay specifications on the Macro input/output routes.

# Macro input nets(DCM to macro input):
net "clk0dcm"    MAXDELAY = 0.450 NS;
net "clk90dcm"   MAXDELAY = 0.450 NS;

# Macro output nets(macro to BUFG):
net "clk0d2inv"  MAXDELAY = 0.755 NS;
net "clk90d2inv" MAXDELAY = 0.755 NS;

# Prohibit MAP from adding the IOB input delay element
# to the input bus (and strobe signals if present)
net "input_databus_name" IOBDELAY = NONE;
net "input_datastrobe_name" IOBDELAY = NONE;
```

DCM and BUFGMUX Guidelines

All DCM specifications are valid when using the EDIF macros, including jitter, input and output frequencies, and the phase offset between all DCM clock outputs. However, since the macros insert additional delay between the DCM and BUFGMUX, there is an additional phase offset at the output of the BUFGMUXs between clocks using the macros and those that use the dedicated routes. TRACE correctly calculates this phase difference and treats it as clock skew for signals that cross clock boundaries. Consider the examples in Figure 6 and Figure 7.

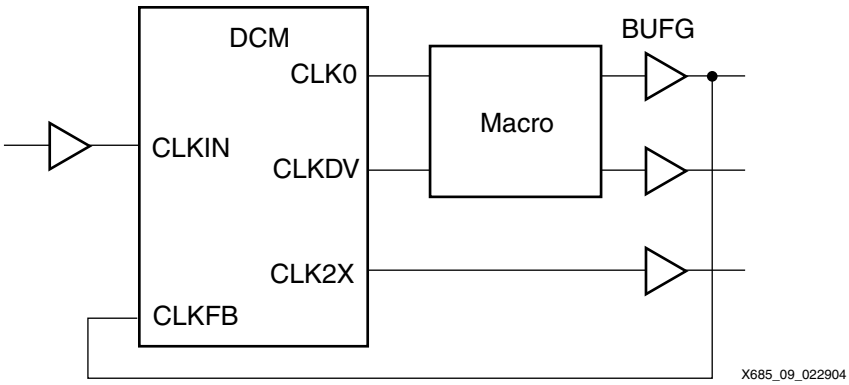


Figure 6: **Phase Offset Example 1**

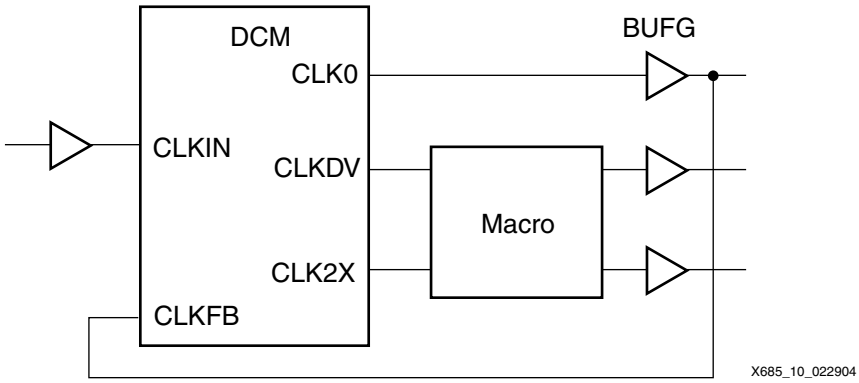


Figure 7: **Phase Offset Example 2**

In Figure 6, the outputs of CLK0 and CLKDV meet the specification for CLKOUT_PHASE and are deskewed to match CLKIN. CLK2X is not phase-aligned to any of these clocks. In Figure 7, the outputs of CLKDV and CLK2X are phase-aligned but are not deskewed to match CLK0 or CLKIN.

Only two output clocks from the DCM are recommended for use with these macros. If a third clock needs to be phase-aligned, a second DCM is needed to provide this clock.

Any BUFGMUX can be used with these macros as long as it is on the same edge of the chip as the DCM and follows the general rules as laid out in the *Virtex- II Pro Platform FPGA User Guide*. Both input pins (I0 and I1) of the BUFGMUX are accessible to the macros.

Timing Analysis

For ISE 6.3 and newer, timing analysis should be performed as normal.

For earlier ISE releases, timing analysis should be performed using the following workaround:

When using the EDIF macros, timing analysis is identical with the exception of I/O timing. As of the ISE 6.2i release, TRACE and Timing Analyzer do not report the correct I/O timing when utilizing these macros in a design, due to an incorrect calculation of Tdcmino. This issue can be addressed as follows:

1. Complete the design with the macros included.
2. Run PAR.
3. Disconnect the macro inputs and outputs and reconnect the DCM through the normal dedicated routes through FPGA_EDLINE and save the resulting NCD under a new name.

   An example FPGA_EDLINE script is provided in the ZIP file. It needs to be modified to specify the specific net names and DCM/BUFGMUX combos per design.
4. Run TRACE on this new design to determine if I/O timing is met.

Also for the 6.2i release, data must be captured and transferred within the I/O flops when utilizing the macro due to offset constraints not being correctly applied as a result of an incorrect Tdcmino. This constraint will be fixed in a future release.

XC2VP100 Corner DCM Requirements

The following DCMs in XC2VP100 devices with –6 speed grades do not lock when operated at frequencies greater than 360 MHz:

- X0Y0
- X0Y1
- X5Y0
- X5Y1

The affected DCMs are physically located in the corners of the chip (see Figure 8). Only by using the EDIF macros can the corner DCMs operate and lock at frequencies greater than 360 MHz.

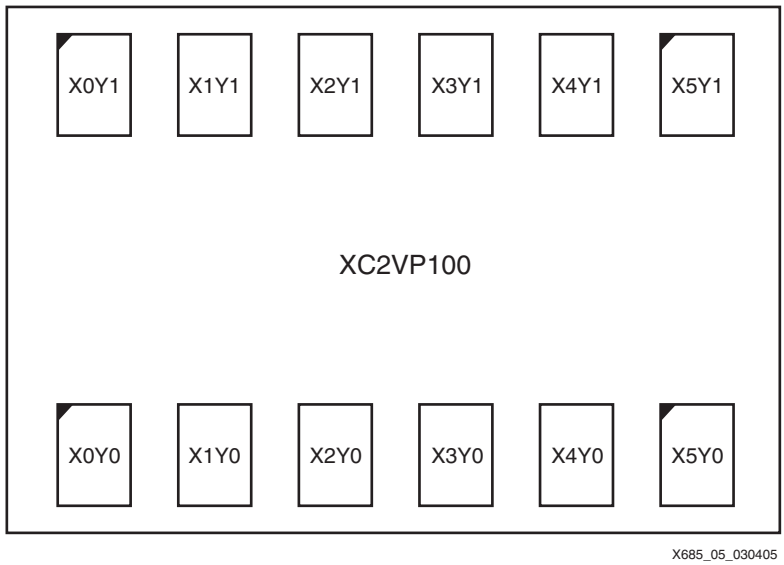
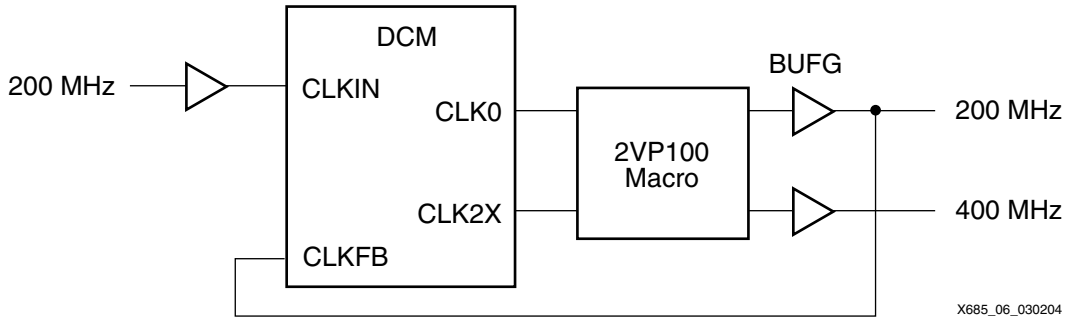


Figure 8: **XC2VP100 with Corner DCMs**

Figure 9 through Figure 11 show example circuits of the corner XC2VP100 DCMs used at frequencies greater than 360 MHz. A couple limitations with using the DCM macros exist:

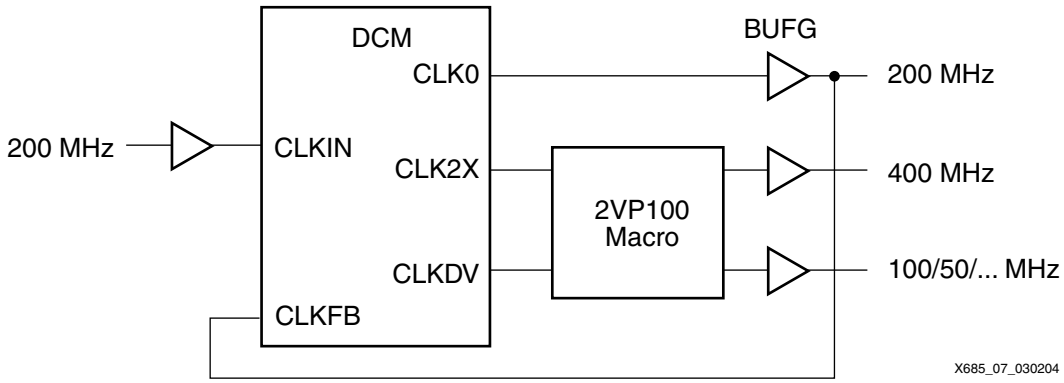
- Only two clocks can be connected to the input of the macro. Therefore, for DCMs with three or more output clocks, there is potential for the clocks not being phase-aligned or deskewed correctly. See the "DCM and BUFGMUX Guidelines" section above.
- The TRACE and Timing Analyzer tool do not report the correct I/O timing. See "Timing Analysis" section above.



Notes:

1. CLK2X cannot be used as the feedback clock in Virtex-II Pro devices.

Figure 9: **Corner XC2VP100 DCM Example Circuit 1**



Notes:

1. The macro clocks and CLK0 are not phase-aligned at the BUFG outputs.

Figure 10: **Corner XC2VP100 DCM Example Circuit 2**

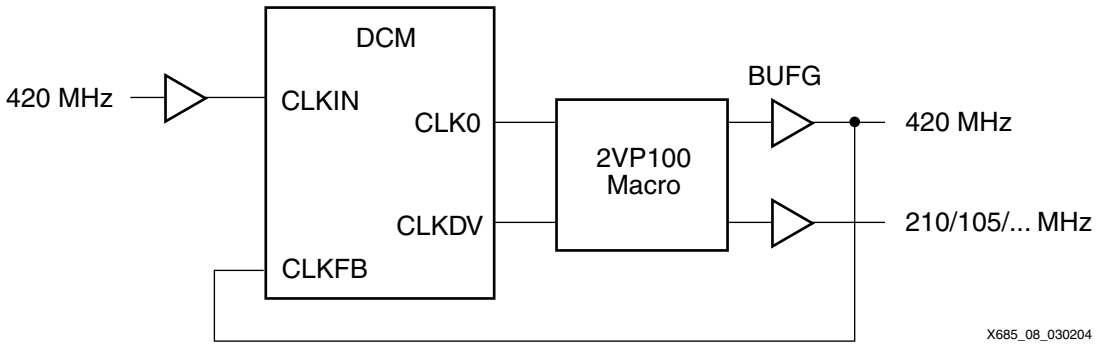


Figure 11: **Corner XC2VP100 DCM Example Circuit 3**

Reference Design

The Virtex-II Pro local inversion clock macros and reference design are available on the Xilinx website at: http://www.xilinx.com/bvdocs/appnotes/xapp685.zip. The ZIP file includes synthesized EDIF macros, an example constraint file, an example reference design, and simulation models. Tool version ISE 6.1i or greater must be used with the macros.

Conclusion

Virtex-II Pro devices support high-speed double data rate (DDR) designs using local-inversion clocking techniques that help reduce global clock resource usage. A macro is required to bypass the dedicated DCM to BUFG route in order to use local-inversion clocking. Corner DCMs in the XC2VP100 –6 devices operating above 360 MHz are required to use these macros, even for single data rate (SDR) applications. This application note provides these macros and describes how to implement them in HDL flow designs.

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
12/11/03	1.0	Initial Xilinx release.
03/03/04	1.1	Added "DCM and BUFGMUX Guidelines", "Timing Analysis", and "XC2VP100 Corner DCM Requirements" sections.
04/30/04	1.2	Added 360 MHz reference to "Summary". Added new section entitled "Application Example." In the "Constraint Requirements" section, added the multiple constraint requirements and added a new constraint to the code snippet. In the "XC2VP100 Corner DCM Requirements" section, added the limitations with DCM macros. The reference design files were updated to Revision 1.3.
03/04/05	1.3	Updated "Timing Analysis" section for ISE 6.3. The reference design files were updated to Revision 1.4.