WP190 (v1.0) February 25, 2003

System Clock Management Simplified with Virtex-II Pro FPGAs

By: Chris Ebeling, Thane Koontz, Ralf Krueger, and Anil Telikepalli

Clock management in electronic designs is as important, if not more, as in football, synchronized swimming, or even flying airplanes and rockets. Without proper management of various clocks and clock circuitry, the system performance can be totally disrupted not to mention data loss. Virtex-II Pro™ FPGAs provide Digital Clock Management circuitry to handle all clock management requirements at the device, board, and system level resulting in simplified designs and reduced costs.

System Clocking

Today's systems consist of multiple master clocks. Each master clock is multiplied or divided and also distributed to various devices in the system. A typical box (Figure 1) in networking, storage, or telecom systems includes line cards, clock card, and backplane.

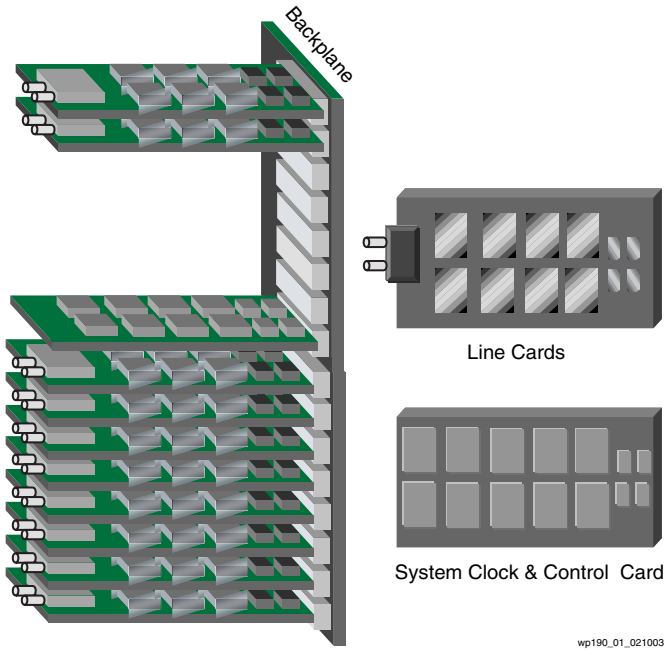

Figure 1: **Typical System Showing Clock Control Card, Backplane, and Line Cards**

The clock card has to generate and distribute clocks that travel via the backplane to the various line cards. Similarly, each line card has to generate and distribute clocks to different devices on the card, such as FPGAs, ASICs, ASSPs, and processors. As clocks propagate through the system, accumulating jitter(1) and skew(2) reduce the maximum clock frequency, cause setup/hold violations, and result in data synchronization problems. It takes numerous verification iterations and sometimes board re-spins to fix these problems. On the other hand, with proper clock management, the designer can reduce jitter and skew associated with the clocks to capture accurate data and save costs. The key to achieving this is good clock management circuitry.

Notes:

1. Jitter (Figure 2) occurs due to system noise and signal crosstalk and causes phase uncertainty resulting in ambiguity in the rising and falling edge of a signal. Jitter can be both random and deterministic.

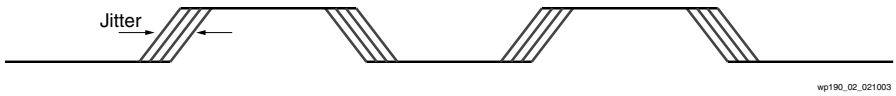

Figure 2: **Jitter**

2. Skew (Figure 3) is a time offset of the clocks. It is a fixed difference between technically identical clocks caused by intra-device clock network, board layout, device process variations, and unbalanced loading.

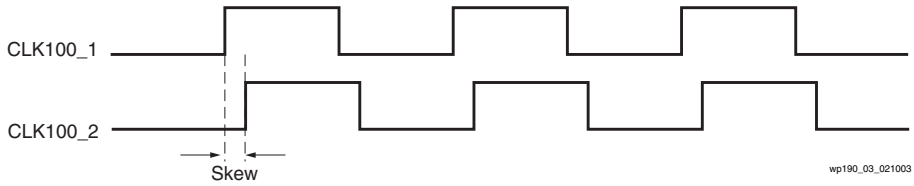

Figure 3: **Skew**

For maximum system performance, the clock must arrive at all line cards with skew and jitter matched up. Backplanes serve as system highways and are especially prone to noise due to the high density of signal traffic through them. Backplanes also consist of multiple layers and connectors contributing to significant signal delay, skew, and jitter. Due to varied distances of the line cards from the clock card, the skew and jitter on the clock arriving at each line card can also vary. To mitigate this problem, various backplane trace topologies are applied but reduce skew and jitter only to a limited extent leaving the remaining job to the individual line card clock management circuitry.

With Virtex-II Pro FPGAs and embedded Digital Clock Managers (DCMs), the designer can de-skew the clock at the input of the line card. DCMs help in clock management not just between the blocks implemented in the FPGA but also among the devices on the board. This includes frequency synthesis, board de-skewing, precise clock data synchronization for double-data rate (DDR) interfaces as well as emerging source-synchronous standards, such as POS-PHY, RapidIO, and others. Using DCMs can eliminate long verification cycles, numerous iterations, and expensive board re-spins to achieve a desired performance goals.

Virtex-II Pro Digital Clock Managers

Virtex-II Pro FPGAs contain up to 12 DCMs (Figure 4) that offer a wide range of powerful clock management features. Each DCM uses fully digital delay lines to deliver high-precision control of clock phase and frequency. Fully digital feedback systems dynamically compensate for temperature and voltage variations during operation and provide unparalleled system clock management capability.

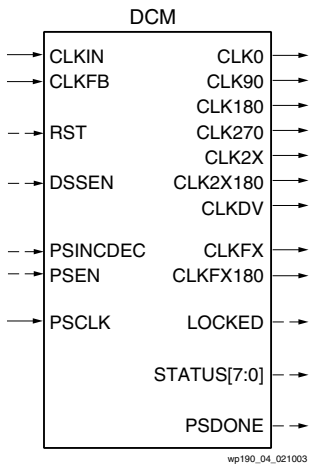

Figure 4: **Digital Clock Manager**

Multiple DCMs help manage several independent clocks in the system by providing the following management features:

- **Clock De-skew**. Generates new system clocks (internal or external to the FPGA), phase-aligned to the input clock for eliminating clock distribution delays.
- **Frequency Synthesis**. Generates a wide range of output clock frequencies performing very flexible clock multiplication and division.
- **Phase Shifting**. Performs forward or backward (positive or negative) phase shift with respect to input clock. Implement coarse 90° phase shifting (0°, 90°, 180°, and 270°), as well as precision fine-grain (1/256 of the clock period) phase shifting in either fixed or dynamic mode.

ISE Architecture Wizards

Xilinx ISE tools provide an Architecture Wizard to customize the DCM parameters (Figure 5). The wizard helps the user to choose all the required features of the DCM for the end application by using an easy-to-use graphical interface. For more information, refer to **http://www.xilinx.com/ise**.

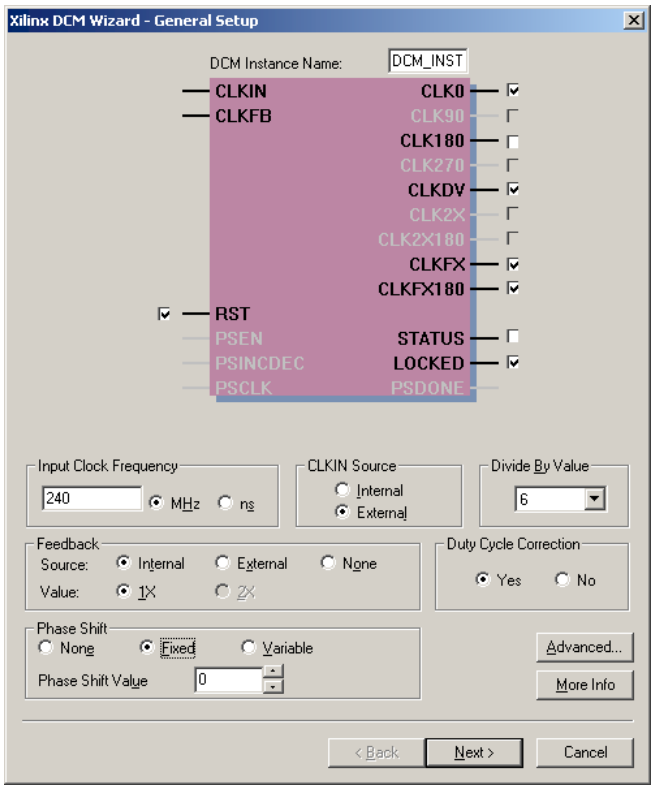

Figure 5: **DCM Wizard**

Why Digital Circuit Circuitry?

With shrinking process technologies and availability of multi-million gate densities in a single device, multiple clock domains, jitter, and skew have become a reality even at the device level. So, logic indicates that integrating clock management circuitry into the device is immensely beneficial. Analog circuits, such as Phase Locked Loops (PLLs) are popular for clock management, but integrating PLLs poses a problem. PLLs are analog circuits and are themselves highly susceptible to the noise resulting from thousands of transistors in the device. This noise susceptibility causes clock locking problems, jitter, unwarranted drift, voltage and temperature vulnerability, and coarse phase-shifting capability. By contrast, a completely digital DCM block integrated into

Virtex-II Pro FPGAs delivers noise immunity and eliminates all the problems posed by an integrated PLL.

Precision Phase Control

Good clock management must ensure that the clock is synchronized with data correctly. The goal is to make the clock fall at the center of the data eye first time, every time. Virtex-II Pro DCMs provide unparalleled clock and data synchronization immune to process, temperature, and voltage fluctuations. Some solutions use training patterns to make the clock fall into the data eye, but not necessarily at the center. However, without dynamic and fine phase shift control, the clock might not even fall in the data eye during operation. The DCM provides precision phase shift at the 1/256th step of input clock so that the designer can move the clock exactly into the center of the data eye (Figure 6).

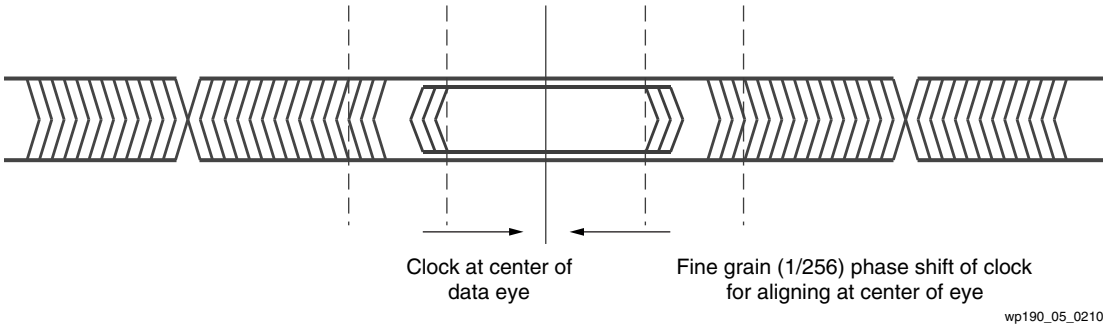

Figure 6: **Fine-Grain (1/256) Phase Shift of Clock for Aligning at Center of Data Eye (Sample Window)**

DCMs provide fixed as well as variable phase shift control. Fixed mode can be used to adjust the clock phase by a fixed value (*N* x 1/256 of clock period, where *N* is an integer) if the phase adjustment has been calculated using measurements. On the other hand, variable mode can be used to dynamically increment and decrement the clock phase (in 1/256 units) to determine the data eye edges and hence the preferred center. Refer to **XAPP268** application note and a free reference design for help in using phase shift control.

The Virtex Edge with Digital Clock Managers

With powerful clock management features, Virtex-II Pro FPGA DCM blocks enable today's high density, high performance designs. Whether it is adhering to source synchronous standards such as SPI4.2, interfacing to external high-speed memories, clock switching, or just eliminating board re-spins, Virtex-II Pro DCMs provide the edge.

Clock Management in OIF SPI4.2 (POS PHY L4) Interface

The Optical Internetworking Forum's (OIF) SPI4.2 protocol uses a 17-bit double data rate source synchronous data bus. Source synchronous interfaces eliminate clock distribution issues associated with high-speed interfaces that have historically used a centrally distributed system clock. In a source synchronous system, the transmitting interface provides a clock (forwarded clock) that is synchronous and typically in phase with the data bits. The receiving device in turn uses this forwarded clock to capture the incoming data. Because the clock follows the same path on the PCB as the data, the

PCB can be designed so that the propagation of the clock and the data match, maintaining their original phase relationship.

The SPI4.2 static alignment specification requires shifting (delaying) of the forwarded clock between the transmitter and the receiver. The 0 degree clock-data phase relationship (clock and data edges align) at the transmitter becomes a 90 degree phase relationship (clock is centered in data eye) at the receiver. While it is possible to implement the clock delay either in PCB traces or via active devices, these methods have a number of limitations. Relying on PCB delays to synchronize clock and data is complicated, not to mention ambiguity associated with temperature and process variation. If clock is not at the center of the eye, but somewhere within the eye, it can lead to setup/hold problems and result in missing data bits. To fix this can take many engineering man-months resulting in expensive debug and PCB difficulties.

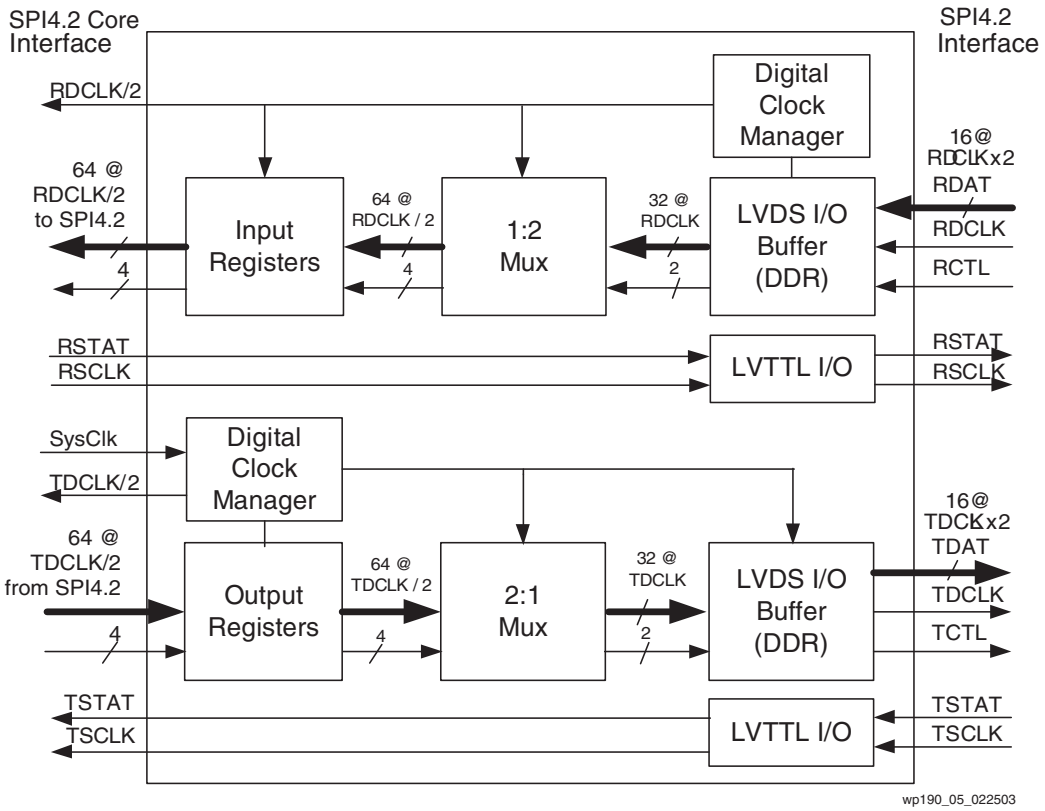

Figure 7: **SPI4.2 Interface Using DCM**

The Xilinx SPI4.2 core overcomes the difficulties associated with attempting to precisely delay the forwarded clock on the PCB by using the Fixed Phase shift capability of the Virtex™-II/Virtex-II Pro DCM to phase shift the internally distributed version of forwarded source synchronous clock relative to the external clock.

At the receiver, the internal clock can be made to lag the external input by 90° to provide the equivalent of the external delay without having to engineer it into the PCB. This approach has the added advantage of actually being able to accommodate any clock/data phase relationship at the input pins and to allow tuning the interface after fabrication of the PCB to the target platform's own particular characteristics. The DCM's capability to delay the clock up to 360° (180° only required for DDR

applications) in steps as small as 50 picoseconds with repeatable accuracy enables the SPI4.2 core to meet the aggressive timing requirements of the SPI4.2 specification, while simultaneously simplifying the design effort for the associated PCB.

For more information on the Xilinx SPI4.2 core, go to:
http://www.xilinx.com/ipcenter/posphyl4/spi42_core.htm

For all connectivity cores, go to: http://www.xilinx.com/connectivity

High Speed Memory Interfaces

Almost every application requires interfacing to some type of an external memory. Memory devices drive strobe signal and data in response to a read command. In some memories such as RLDRAM I /II and FCRAM II, the QDR SRAM strobe signal is free running and in others such as DDR SDRAM I/II and FCRAM I, it is not. In most memories, the data and strobe are driven together and they arrive edge aligned at the destination device (Virtex-II Pro FPGA). To capture the data at the destination, the strobe has to be delayed to center it in the data window.

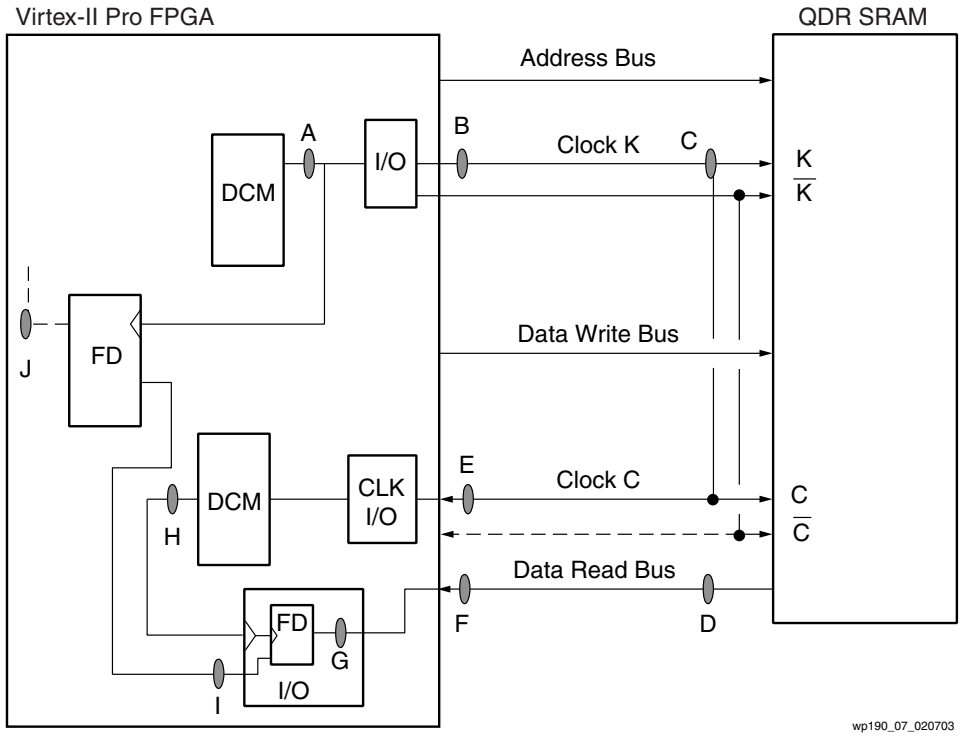

Figure 8: **Virtex-II Pro Memory Interface Diagram**

While this delay can be achieved using PCB traces, it poses problems similar to the SPI4.2 interface delaying the design completion and resulting in costly verification/debug and poor performance. The Virtex-II Pro DCM fine and quadrature phase shift control makes interfacing Virtex-II Pro devices to external memories easy. During a write to the memory, memory interfaces use the quadrature phase-shift capability of the DCM. The system clock is fed as an input to the DCM and Clocks 0 and 180 outputs are used to clock write data through the IOB DDR flip-flops. Clocks 90 and 270 outputs of the DCM are used to clock the write strobe through the IOB DDR flip-flops.

During a read from memory, both the quadrature output and fine-phase adjustment capabilities of the DCM are employed. There are two different methods that can be

used to capture read data. Although each of these methods uses the same DCM features, they are applied in different ways.

In memories with free running strobes, a dedicated DCM is used for each strobe. The free running strobe signal is input to the DCM as if it is a clock. The DCM quadrature phase outputs (CLK90/CLK270) and/or the fine phase shift feature in the DCM can be used to center this strobe in the data window. These phase shifted strobe signals from the DCM outputs clock the IOB DDR flip-flops and capture read data. The system clock is used to clock the captured read data from the IOB DDR flip-flops into the CLB flip-flops and complete the read data recapture process.

Strobes that are not free running cannot be input to the DCM as a clock. In this scenario, the strobe can be ignored and the system clock can be used to clock the DCM input. A single DCM is used for both reading and writing and the number of strobes is irrelevant. The DCM quadrature phase outputs (CLK90/CLK270) along with fine-phase adjustment can be used to center this clock in the data window. These phase shifted system clocks from DCM outputs clock the IOB DDR flip-flops and capture read data. The fine-phase adjustment enables compensation for phase offset between the quadrature clock outputs and the read data from memory.

For free memory application notes and reference designs, go to:
http://www.xilinx.com/apps/virtexapp.htm

Clock and Data Mismatch After Board Layout? No Problem!

If there is a clock and data mismatch with any on-board device talking to the Virtex-II Pro FPGA, the designer can eliminate costly board re-spins. The DCM block can be individually reconfigured with coarse and fine-phase control to eliminate any mismatch. The designer can manually shift the phase forward or backward in steps even after the board layout.

Clock Switching and Redundant Clocks

Certain applications in telecom systems need high reliability and require redundant/alternate clocks to facilitate clock switching in the event one of the clocks fails. Virtex-II Pro FPGAs enable high reliability system design—a pair of Virtex-II Pro DCMs can drive a single clock buffer (BUFGMUX). As soon as one clock fails, the second clock takes over. Since redundant clocks are usually not phase-aligned, PLLs generally have to re-lock during which the clock is ambiguous, causing application failure. DCMs provide seamless and glitch-free transition from the failed clock to the other without any ambiguity. Refer to *Global Clock Networks* in the Virtex-II Pro user guide (**UG024**) for guidance.

Summary

Virtex-II Pro FPGAs with their embedded DCMs solve clock management problems in high-speed design, giving the designer an edge over other alternatives. Whether it is system-clocking needs including frequency synthesis, phase alignment, board de-skew, adhering to source synchronous interface standards, clock-data synchronization, clock switching, or all of the above, DCMs make designing simpler, saving development and testing costs. Plus, the designer can meet high-performance standards requirements without any setbacks. The DCM blocks in the Virtex-II Pro FPGA, along with ISE tools, IP, and reference designs, work together to solve all clock management challenges.

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
02/25/03	1.0	Initial Xilinx release.