Virtex Delay-Locked Loops (DLL)

VTT003 (v1.1) August 7, 2000

Introduction

Supporting the highest bandwidth data rates between devices requires advanced clock management technology such as digital delay-locked loops (DLLs). The DLL circuitry allows for very precise synchronization of external and internal clocks. Xilinx was the first to deliver DLLs in programmable logic by offering four 200 MHz DLLs in every Virtex device. The Virtex-E family takes this technology to the next level with devices containing eight DLLs capable of over 311 MHz. Figure 1 shows a block diagram of the DLL circuitry. Virtex Series DLLs provide precise clock edges through phase shifting, frequency multiplication, and frequency division. The precise duty cycle generation is critical for high performance applications (like Double Data Rate, or DDR) in which a slight shift in duty cycle can dramatically decrease overall system performance.

History

Phase-locked loops (PLLs) have been used since the 1940's in analog implementations. Recent emphasis on digital methods has made it desirable to match signal phases digitally. Using a digital delay-locked loop (DLL) in place of an analog PLL eliminates the need for separate noise-free ground and power planes. Virtex DLLs also ensure a reliable frequency range over all variations in manufacturing processes, temperature, and voltages.

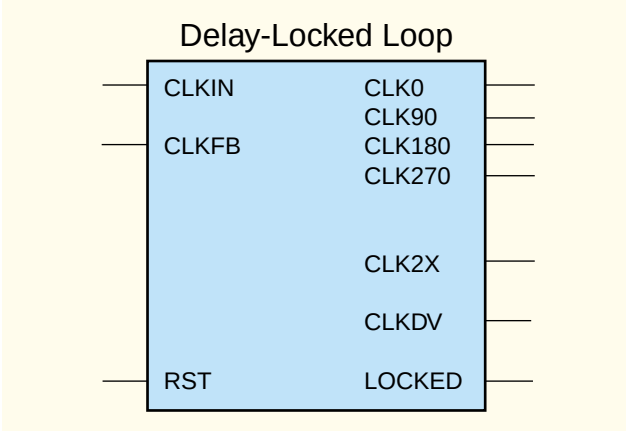


Figure 1: **DLL Block Diagram**

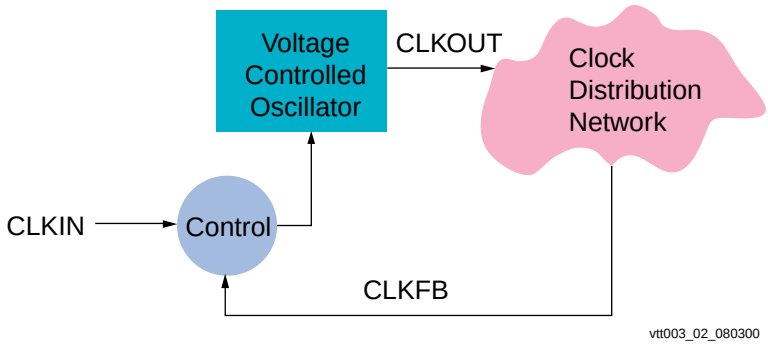


Figure 2: **PLL Block Diagram**

Phased-locked loops are implemented using either analog or digital techniques, and trade-offs exist for each implementation. An analog PLL can be designed with a finer timing resolution and more frequency synthesis features in a small silicon area. However, PLLs have disadvantages that make their use in high-speed designs problematic, particularly when both high performance and high reliability are required.

The PLL voltage-controlled oscillator (VCO) is the greatest source of problems. Variations in temperature, supply voltage, and manufacturing process affect the stability and operating performance of PLLs.

DLLs, however, are immune to these problems. A DLL in its simplest form inserts a variable delay line between the external clock and the internal clock. The clock tree distributes the clock to all registers and then back to the feedback pin of the DLL. The control circuit of the DLL adjusts the delays so that the rising edges of the feedback clock align with the input clock. Once the edges of the clocks are aligned, the DLL is locked, and both the input buffer delay and the clock skew are reduced to zero. The major advantages of DLLs are precision, stability, power management, noise insensitivity, and jitter performance.

The Virtex FPGA series provides up to eight fully digital dedicated on-chip DLL circuits which provide system clock generation, de-skewing of clock signals distributed throughout the device and/or the board, and other advanced clock domain control. In addition to frequency synthesis, a DLL optionally provides duty cycle correction and phase shift.

DLLs

Digital Delay-Locked Loops Provide Significant System Benefits

1. Achieving zero clock skew, effectively eliminating the clock-distribution delay and allowing digital closed-loop control. This control provides system clock rates ranging from 25 to 320 Mb/s with 100-ps resolution. A DLL greatly reduces the clock latency of a device, which in turn reduces the clock-to-out timing.
2. Precise control of system clocks, both internally at the device level as well as externally for other devices on the board using the independent DLLs per Virtex or Virtex-E device.
3. Clock mirroring on the PCB to synchronize external devices. This also reduces system cost by eliminating the need for external devices (Cypress Roboclock). DLLs support clock mirroring with less than 100 ps skew. The DLL output clock can be taken off-chip to synchronize external devices, thus eliminating race conditions and improving chip-to-chip performance. To assure a synchronous start of all devices, use the LOCK output signal. On startup the PROGRAMMING DONE signal can be delayed optionally, until lock is achieved. As shown in Figure 3, the clock can also be connected to the feedback pin of the DLL, thereby de-skewing the board's system clock to less than 100 ps.

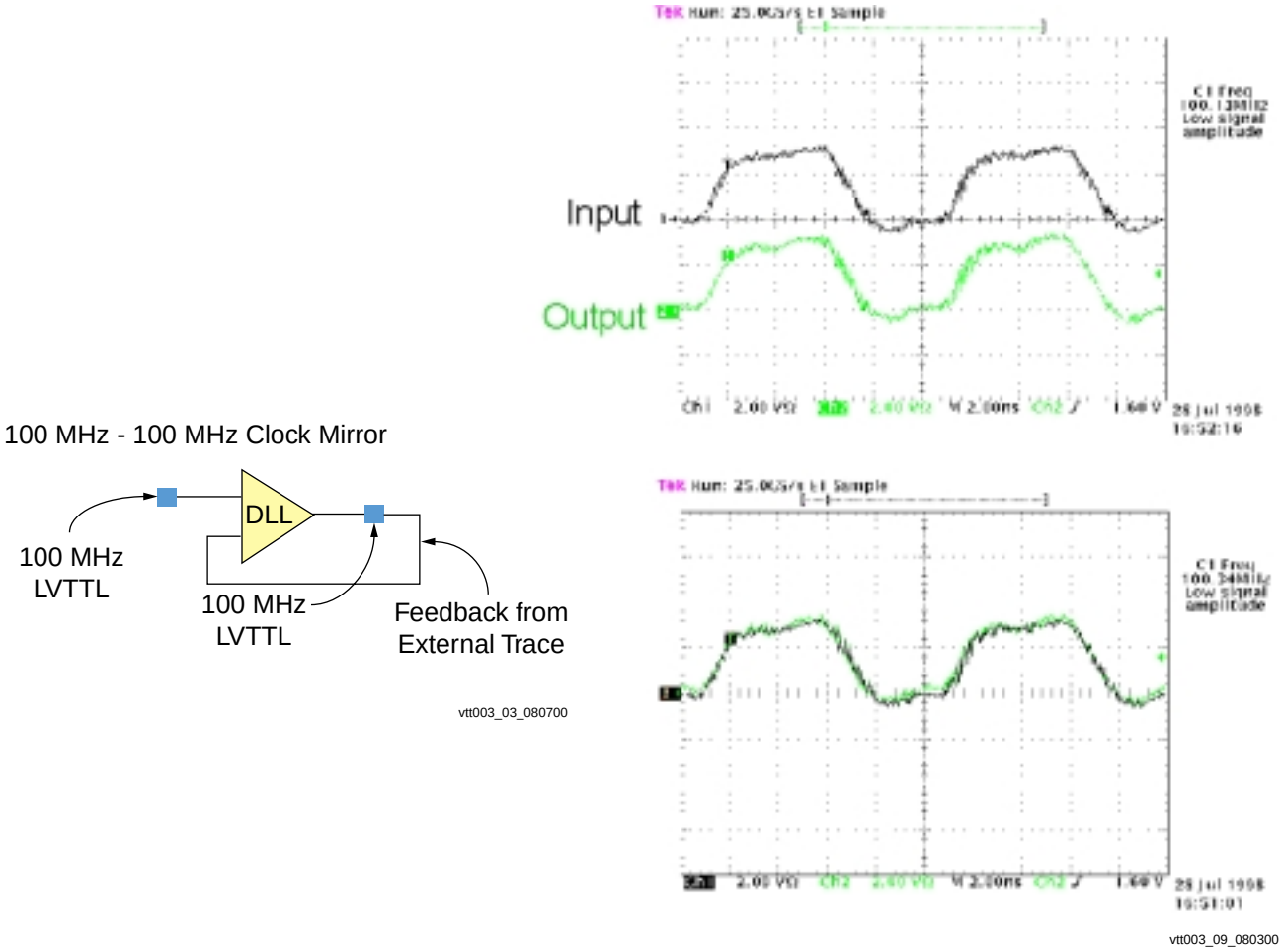
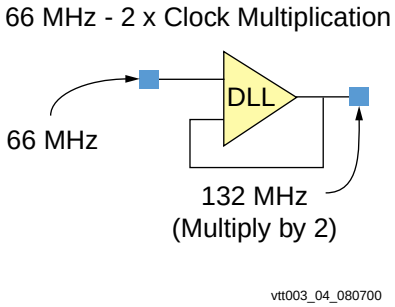


Figure 3: **Using Clock Mirror to De-Skew the System Clock**

4. Multiply or divide the external clock to produce a clock to use on or off the device, allowing for multiple clock domains. Designers can also use the DLL to phase-shift clocks in order to support clock-multiplexed systems.
   - 2X-4X: Double or quadruple the incoming clock frequency, supporting designs that need a fast internal operation but a slower external clock. This gives the designer a number of choices. For example, clocks routed on the board can be kept to lower frequencies, thus avoiding signal-to-noise issues while the FPGA simultaneously runs at maximum speed. All multiplied clocks have synchronized edges. Internal device clock speeds can be high as 320 MHz. Figure 4 shows a 2X clock multiplication example.
   - CLK_DV: Divide clock by 1.5, 2, 2.5, 3, 4, 5, 8 or 16. Applications often monitor data at a high frequency but process data at a much lower clock frequency (for example, they might read data at 155 MHz and process data at 38.5 MHz).



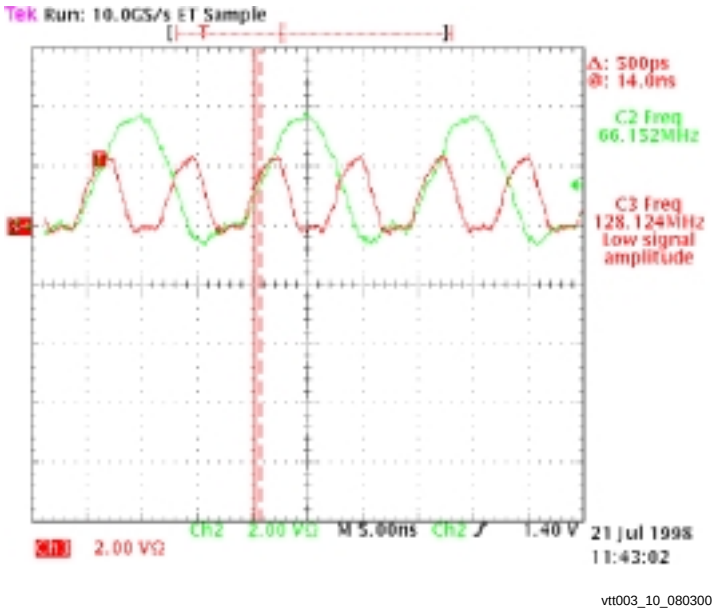
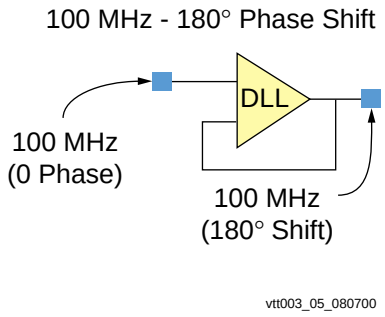


Figure 4: **Clock Multiplication**

- Support clock-multiplexed applications by creating four-quadrant clock phases (0/90/180/270). Input four sequential bits per clock period. DLL phase shift; used with the clock divider (e.g. data is applied at 200 MHz and registered at 50 MHz by four clocks shifted 90 degrees each). Figure 5 shows an example clock phase adjustment.



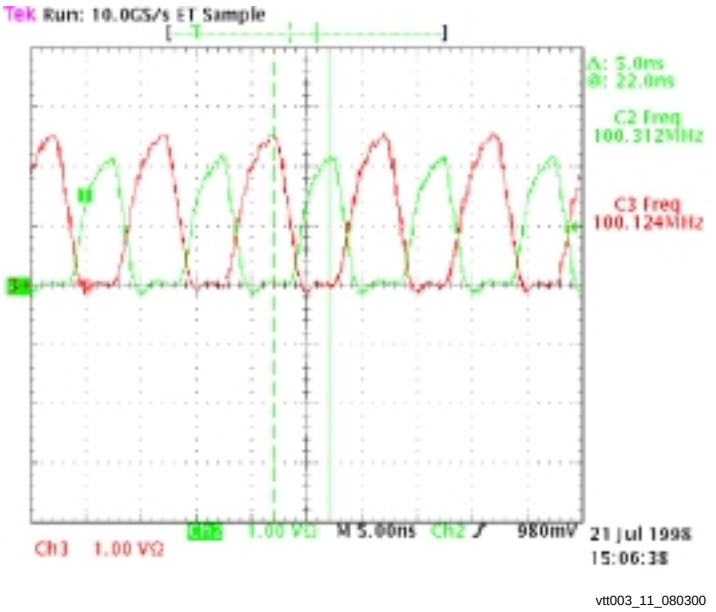
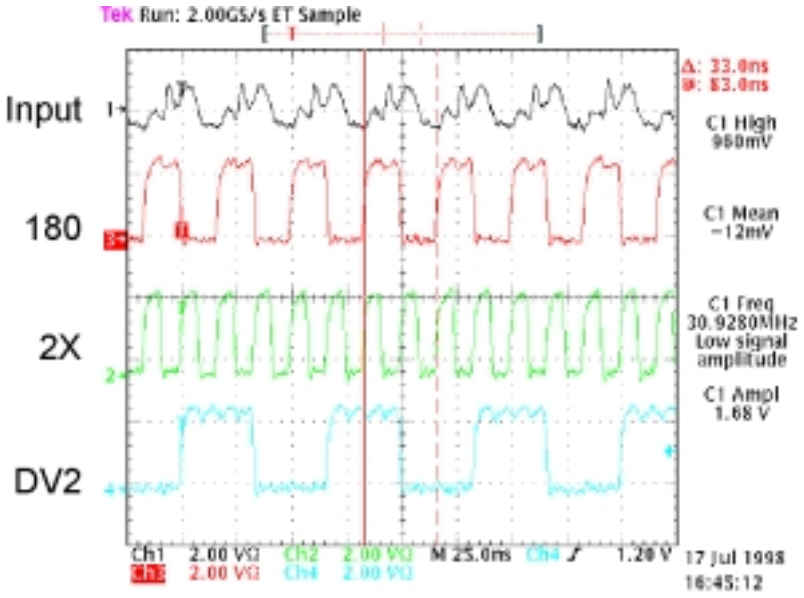


Figure 5: **Clock Phase Shift**

- Figure 6 illustrates the dramatic impact of multiply and divide techniques and phase shift on conditioning a noisy external clock.
- Selectable Division Values
  - 1.5, 2.0, 2.5, 3, 4, 5, 8, or 16
  - 50/50 duty cycle correction
  - Use DLL pair to combine functions



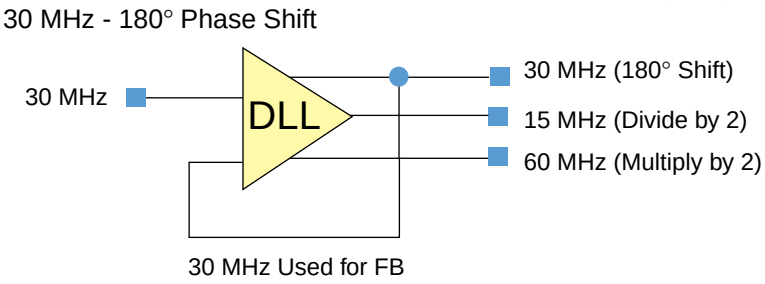


Figure 6: **Phase Shift With Clock Multiply and Divide**

5. Deliver superior chip-to-chip clock performance
   - Up to 622 Mb/s LVDS and LVPECL performance on 36 Virtex-E I/O pairs
   - SelectLink™ technology for high speed DDR Virtex-to-Virtex device communication
     - 200 Mb/s Virtex-to-Virtex device
     - 311 Mb/s for Virtex-E device to Virtex-E device (up to 804 pins)
   - 143 MHz ZBT™ SRAM, 200 MHz for Virtex-E device
   - 125MHz SDRAM/SGRAM, 133 MHz SDRAM.
   - Use precise 50/50 duty cycle to achieve high-speed DDR interface to external devices
     - 200 MHz Virtex Double Data Rate
     - 266 MHz Virtex-E Double Data Rate
6. Provide greater stability. DLLs operate reliably on waveforms with up to 1 ns frequency drifts, adding a maximum of only 60 ps jitter. This stability is far beyond the typical 100 PPM (0.01%) of most oscillators, where any input jitter is reflected on the output. Most analog PLLs can filter jitter on the reference clock, but the intrinsic oscillator noise is still present. VCOs tend to be especially sensitive to I/O switching, V_{CC} variations, and ground bounce. In addition, a less than optimal loop filter can amplify the jitter. Each loop filter must be matched to the multiplier value, the oscillator frequency, and various parameters such as analog frequency, phase, and gain. Digital delay-locked loops have no such limitations, as there are no finely tuned analog control elements. The DLL is implemented with a completely digital feedback mechanism that adjusts the phase in 50-60ps increments.

Most discrete PLLs are designed with a specific application in mind. Once external pins are connected to resistors, capacitors, power, and ground, the designer must insure no signal couples into or interferes with these pins. Any noise may result in either high jitter or the PLL not locking. Otherwise, the designer can experience significant signal integrity problems. On today's high-speed circuit boards this puts an additional burden on the design and layout engineer to provide separate power and ground connections.

Eight High-Performance DLLs

Drop-in Bandwidth Optimization With Virtex-E Devices

Supporting high-bandwidth data rates between devices requires advanced clock management technology offered by DLLs. The DLL circuitry allows for very precise synchronization of external and internal clocks. DLLs also provide precise clock edges during phase shifting, frequency multiplication, and frequency division. Critical Virtex-E DLL bandwidth specifications are listed in Table 1. Precise duty cycle generation is critical for high-performance applications, such as Double Data Rate (DDR), where a slight shift in duty cycle can dramatically decrease overall system performance. The Virtex-E family offers eight DLLs, allowing both internal and external de-skew of four systems clocks (Figure 7).

Table 1: **Bandwidth Critical Specifications of the Virtex-E DLL**

Parameter	Value
Maximum Output Frequency	320 MHz*
Maximum Output Jitter Added	60 ps
Output Frequency Duty Cycle	50% ± 100 ps

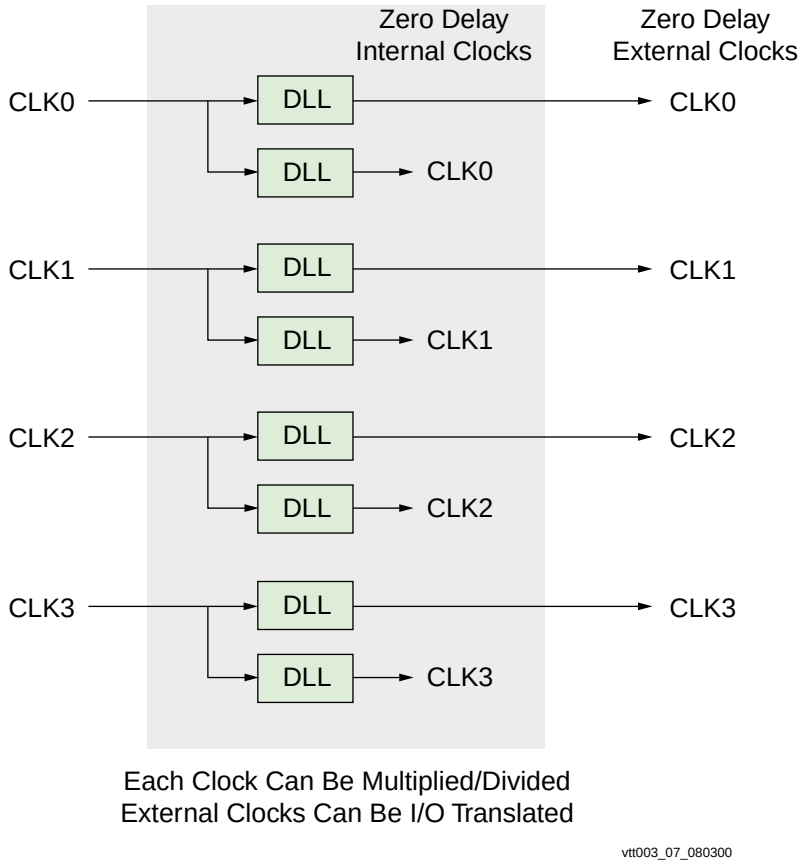


Figure 7: **Four System Clocks De-Skewed**

Maximizing DDR Bandwidth

A key technique for increasing the bandwidth of a particular data port is to have signals change on both edges of a clock, commonly referred to as the Double Data Rate technique. Memory suppliers have already started to support this type of high performance technique to increase the memory bandwidth of their devices. At high frequencies, signal integrity limits the clock performance, which limits the bandwidth of the data. Bandwidth for the port is immediately doubled if the architecture can change data at each edge of a system clock. A precise 50% clock duty cycle is critical for this technique. Since Virtex-E DLLs can generate clocks with a duty cycle guaranteed to be within 100 ps of 50%, system designers can achieve the maximum memory bandwidth in the DDR application. Figure 8 demonstrates how Virtex-E DLLs help achieve maximum bandwidth in a 266 MHz DDR application.

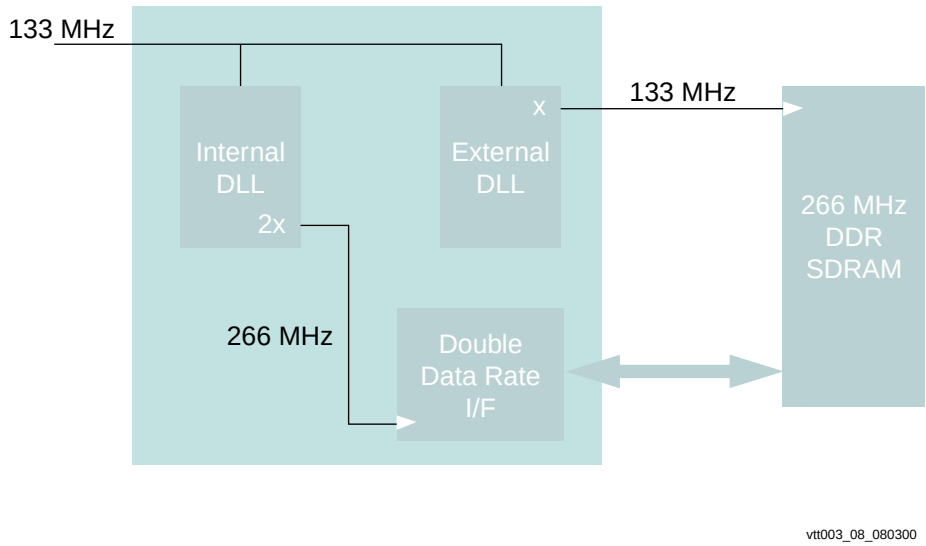


Figure 8: **Virtex-E FPGA Interfacing With a 266 MHz DDR SDRAM Memory Module**

Virtex Advantages

A comparison of the performance and flexibility of DLLs versus PLLs in Table 2 shows the benefits designers gain by using Xilinx Virtex series devices.

- DLLs are beneficial for designers who require external interface performance above 50 MHz. This includes interfacing to memory devices in numerous applications, as well as networking and telecommunications applications.
- DLLs are superior for designs that have a clock fan-out of greater than 10, or when multiple clocks are required.
- DLLs are critical to achieving maximum performance for Double Data Rate (DDR) applications.
- Two DLLs are required for complete internal and external clocks de-skew. With 8 DLLs, Virtex-E allows four system clocks to be managed. Altera's APEX E family has only two or four PLLs, and only one PLL in their APEX family.
- Only two of the maximum four PLLs available on the Altera APEX E product support LVDS. All eight DLLs on Virtex-E support LVDS.
- DLLs do not require separate power and ground planes. Altera's APEX and APEX E families' PLLs require designers to use PCBs with separate noise-free power and ground planes.

Table 2: **Virtex DLL Versus APEX PLL Technology**

Feature	Virtex	Virtex-E	APEX	APEX-E
Architecture	DLL	DLL	PLL	PLL
Technology	100% Digital	100% Digital	100% Analog	100% Analog
Quantity	4	8	1	2 – 4
Max Output (MHz)	200	311	133	200
Input Duty Cycle	n/a	n/a	40 – 60%	40 – 60%
Output Duty Cycle	50% ± 100 ps	50% ± 100 ps	40 – 60%	40 – 60%
Min Input Clock (MHz)	25	25	5	5

Related Xilinx Documents

XAPP132: "Using the Virtex Delay-Locked Loop" at: http://www.xilinx.com/xapp/xapp132.pdf

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
03/30/00	1.0	Initial Xilinx release.
08/07/00	1.1	Updated Table 1 and Figure 6.