Clock and Data Recovery With Coded Data Streams

Author: Leonard Dieguez

XAPP250 (v1.1) April 25, 2002

Summary

This application note and reference design outline a method to implement clock and data recovery in Virtex™-II devices. Although not limiting the implementation to a specific FPGA family, this reference design focuses on the Virtex-II architecture. With minor modifications, Clock and Data Recovery (CDR) is possible with Virtex-E and Spartan™-IIE devices. A implementation of CDR at 270 Mb/s with 8B/10B coded data is described herein. *Note: Designs not requiring a recovered clock should refer to a specific Data Recovery application note* **XAPP224**.

Introduction

This application note describes Clock and Data recovery methods allowing a receiver to extract embedded clock information and retimed data from an incoming encoded data stream. Sometimes, the received clock is also used for onward data transmission. The speed of operation is limited by the maximum input clock frequency from an external VCO and internal routing resources. The maximum operational speed is 420 MHz using a Virtex-II device (-5 speed grade). Figure 1 shows a CDR primitive.

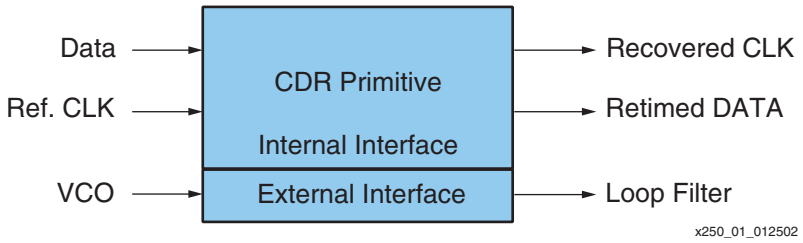

Figure 1: **CDR Primitive**

If a recovered clock is not required in a design, then XAPP224 (Data Recovery) is more appropriate. System requirements are as follows:

- Clock and data must be coded into a single data stream. This method is proven using 8B/10B coded data.
- A reference oscillator
- The availability of four FPGA pins in close proximity (two input LVDS pins and two output CMOS pins)
- An external Voltage Controlled Oscillator (VCO)

A block diagram describing the system is illustrated in Figure 2. The Maxim 2605-2609 devices, operating in frequency ranges from 45 MHz to 650 MHz, come in a SOT-23 6-pin package. These integrated VCOs have tuning diodes and differential outputs. The only external circuit components are a tuning inductor, a bypass capacitor, and an output matching network. The goal is to have a trimless phase-locked loop (PLL) design operating across temperature and device variations. Working knowledge of PLL function, design, and implementation are important when using the accompanying reference design.

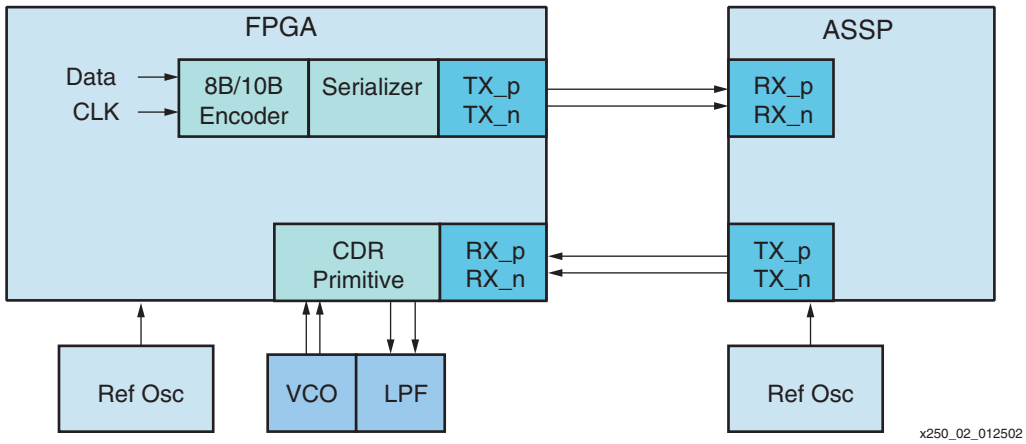

Figure 2: **CDR In Virtex-II Devices**

CDR Architecture

The Figure 3 illustrates the typical block diagram for implementing a PLL. The basic blocks for a PLL are:

- Phase detector
- Loop filter
- Voltage controlled oscillator
- Counters

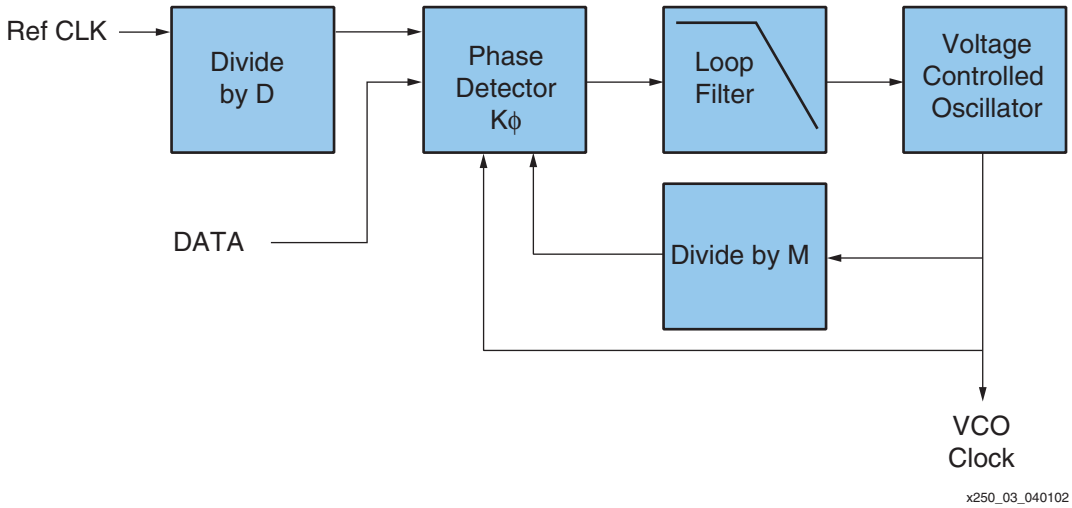

Figure 3: **PLL Block Diagram**

Phase Detectors

A combination of a 3-state phase frequency detector and a delay line phase detector are used in this design for clock recovery and data synchronization.

3-State Phase Detector

The 3-state phase detector is used because it is simple, has a linear range of ±2π radians (at the comparison frequency), and acts as both phase and frequency detector (PFD). A state diagram for the circuit is shown in Figure 4.

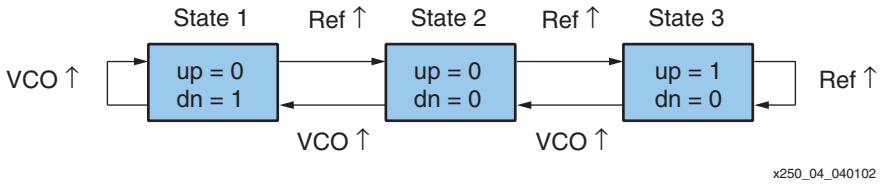

Figure 4: **State Diagram for a 3-State PFD**

One implementation of the 3-state PFD is shown in Figure 5. State 2 is the assumed initial state (both up and down are zero). A rising edge on the reference causes "up" to be asserted, therefore, forcing a transition to the third state. Then when a rising edge of VCO occurs, both up and down are high for an instant. After the gate propagation delay, the registers are both reset. The PFD is now in state 2. The maximum useful frequency for this PFD is limited by the minimum duration of state 2. This PFD is not sensitive to the duty cycle of the VCO or the reference oscillator.

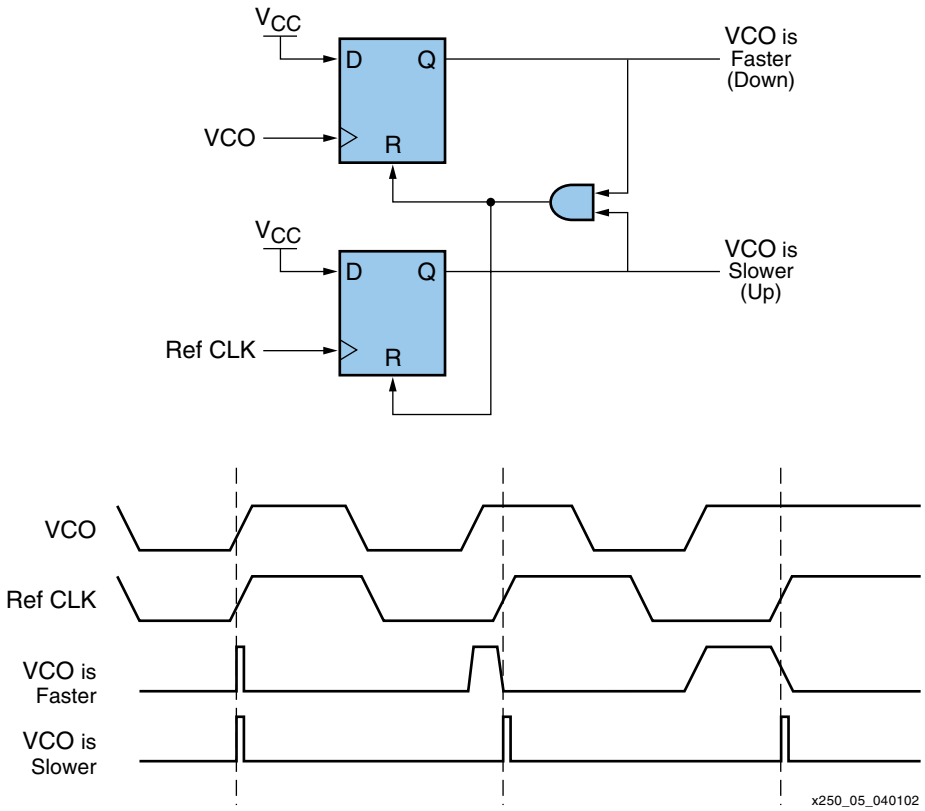

Figure 5: **3-State Phase-Frequency Detector**

When the VCO is running at a faster frequency or is leading the reference frequency in phase and needs to be slowed down or delayed, is indicated by a signal "VCO is Faster". A "VCO is Slower" signal indicates a slower VCO than the reference frequency. The phase detector gain is the following:

$$K_D = \frac{V_H - V_L}{2\pi}$$

Delay Line Phase Detector

The delay line phase detector is composed of three main blocks:

- Delay line
- Registers on the outputs of the delay line
- Phase encoder.

The two basic techniques for detecting the phase between the DCO output and the input data are:

1. Pass the data through a delay line. Then sample the delayed elements with the DCO.
2. Pass the clock through a delay line. Then sample the delayed elements with the DATA from the IOB.

The application note covers the first method. The sampled delay line is shown in Figure 6.

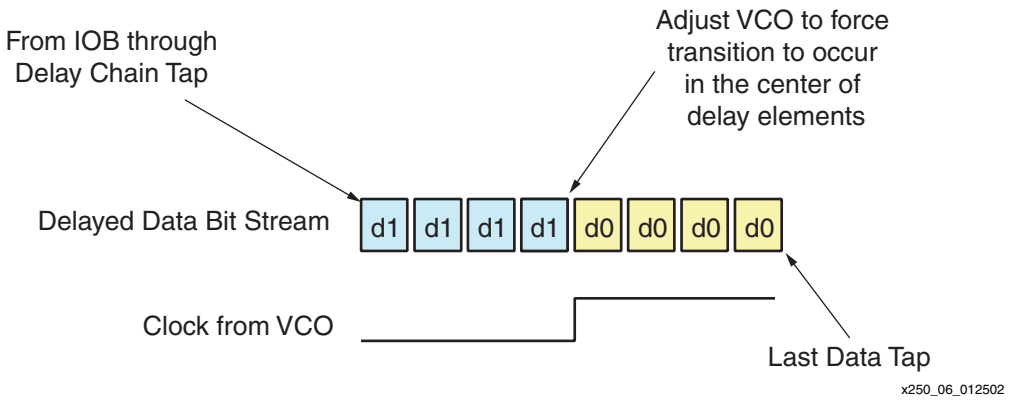

Figure 6: **Sampled Delay Line**

After the PLL has achieved lock to the reference oscillator, control circuitry attempts to recover the clock from coded data. 8B/10B coded data has an NRZ format (i.e., there may be several bit times with no transitions). With 8B/10B coded data, the maximum number of bit times without transitions (maximum run length) is limited to five. The data stream is described as a reference clock with missing transitions (a punched clock). A data stream that only transitions once during a bit time is a single data rate (SDR) signal. This makes the data stream appear as a half-rate clock when an alternating one/zero pattern is present.

The architecture used for the delay line phase detector is depicted in Figure 7. The delay line phase detector is similar to a quadrature phase detector because it outputs a frequency error by detecting phase slippage between the reference clock and the data.The delay line phase detector separates the phase differences in discrete intervals. In the Virtex-II design, a "MUXCY" carry chain is utilized for the discrete delay elements.

The phase detector provides the following functions.

- Detects if a data transition occurred within the current sample period.
- Detects if multiple transitions occurred within the current sample period.
- Produces a phase error output.
- Produces the retimed DATA output.

The delay line should have a delay of greater than one period, but not greater than two periods. The phase detector adjusts the VCO causing it to align the data transition to the center of the delay line. Outputs from the delay line are registered twice to remove any metastability.

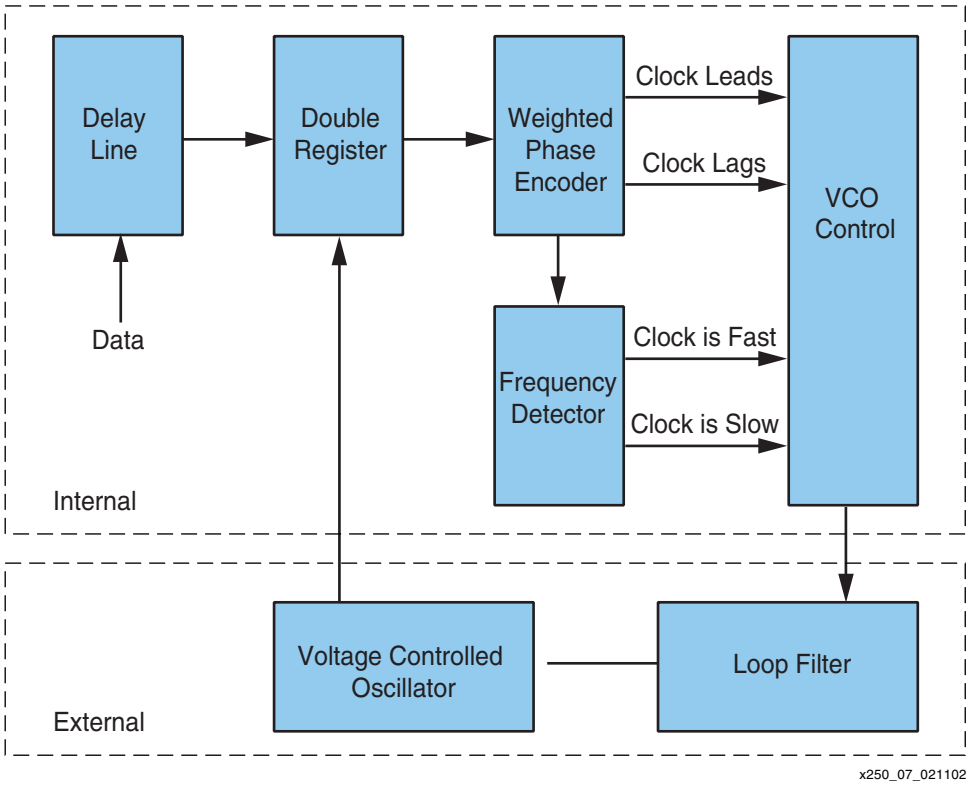

Figure 7: **Delay Line Phase Detector**

Data Transition Detection

Detection of data transitions for the current sample time can be accomplished by comparing strategically spaced sample points on the delay line. The sample points should be close enough to the adjacent sample point to guarantee that two transitions cannot occur. The sample points would be equal (all ones or all zeros) for sample time periods with no data transitions. A clock enable signal is constructed from the comparison of the sample points. The phase error signal is only updated when valid transitions occur in the data stream.

Multiple Data Transition Detection

The delay line phase detector should be designed for a maximum of two data transitions are allowed to occur in the delay line. A filtering mechanism is employed allowing the use of only one of the data transitions for phase error calculations.

For example, if two data transitions are detected at both the beginning and the end of the delay line, a phase correction is input to the VCO such that the phase of one transition is forced to the center of the delay line. The transition selected depends on the last known phase or whether the VCO is running a faster or slower frequency than the nominal bit rate frequency.

Phase Error Output

The phase error output is decoded in order to indicate the relative number of delay elements from the center of the delay line. Phase errors close to the center of the taped delay line indicate small phase errors and, hence, smaller corrections are needed. Phase errors further from the center of the taped delay line indicate larger corrections are needed. See Figure 8.

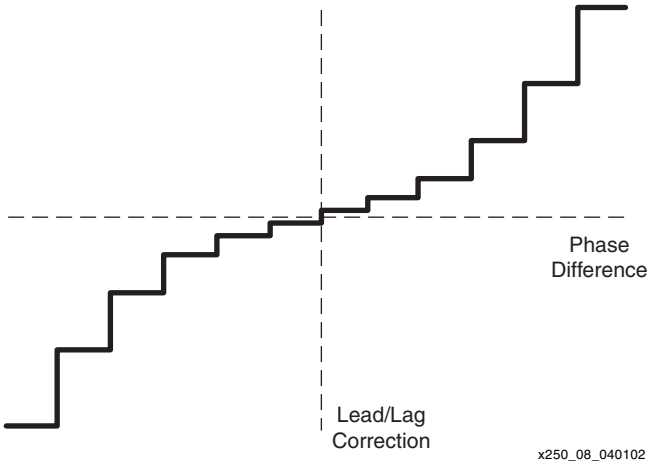

Figure 8: **Phase Error Output**

Retimed Data Output

A tap at the center of the delay line is sampled with the register clocked by the opposite edge of the VCO clock source. This ensures that the sampling of the data is 180 degrees out of phase with the jitter present on the data signal. See Figure 9.

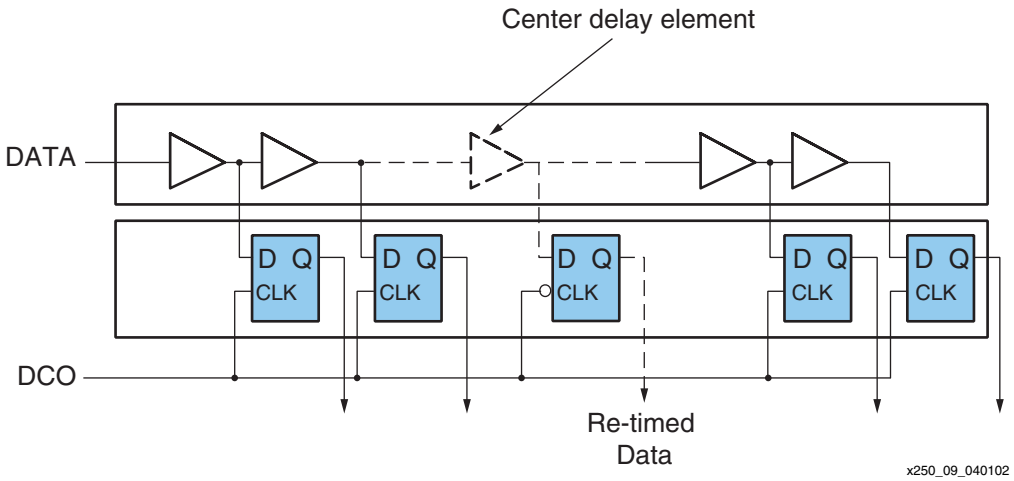

Figure 9: **Retimed Data**

External Voltage Controlled Oscillator

Although this design uses the MAX2605-2609 for the external VCO, any other device with the same functionality can be substituted. The MAX2605-2609 was chosen because it is low cost, contains an integrated tuning diode, is available in a small package (SOT23-6), produces low phase noise, and improves noise immunity by placement physically away from the FPGA. The footprint of the external VCO solution is extremely small. Figure 10 shows the basic configuration for using an external VCO.

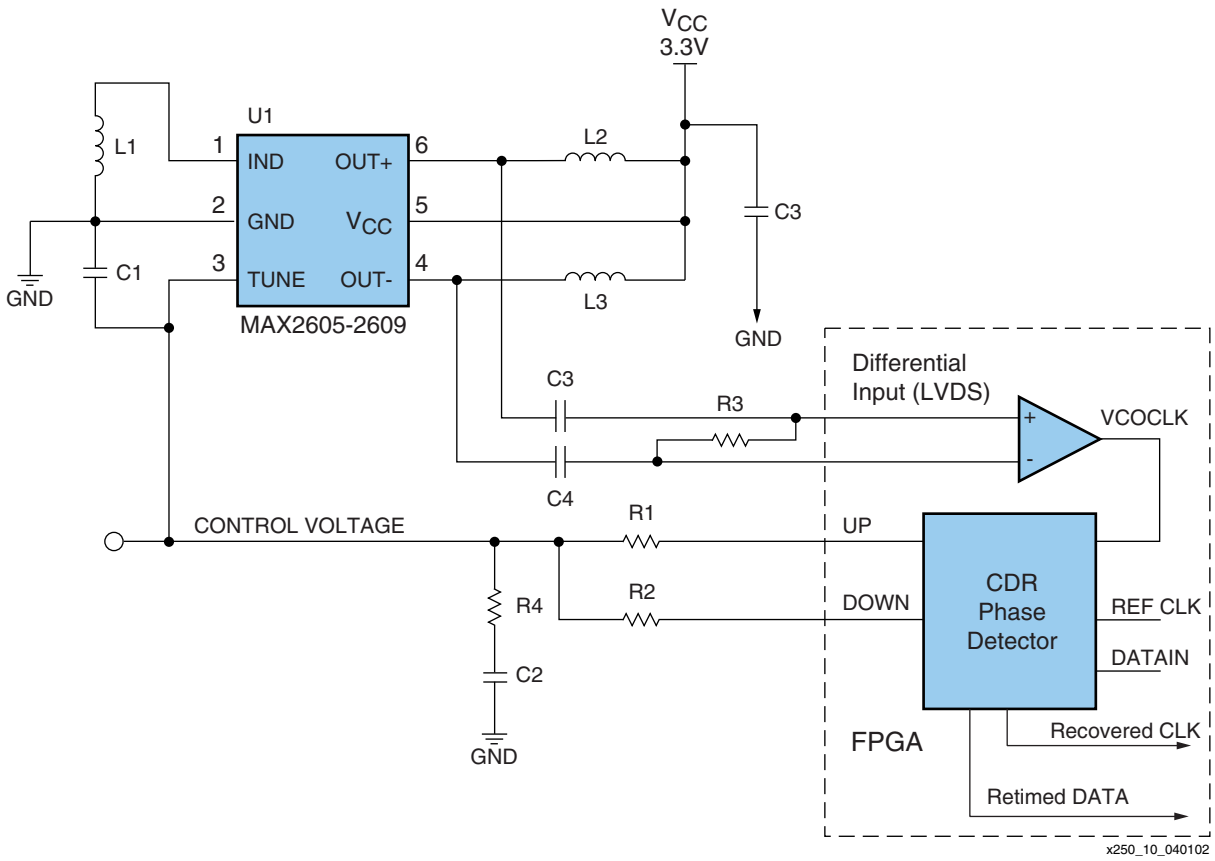

Figure 10: **External VCO Configuration**

Loop Filter

The loop filter described in this section is a passive-lag loop filter. The designer is encouraged to explore using an active loop filter. To quote Gardner from *Phaselock Techniques*[1], "Better performance almost always will be obtained through use of an active filter." Assume the transmission of the data stream is made with a stable 100 ppm crystal oscillator when selecting devices to build the loop filter. The design only has to track temperature variations in a noisy environment. Figure 11 depicts the loop filter used.

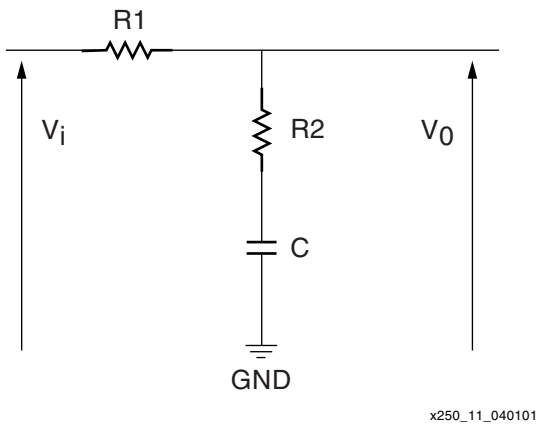

Figure 11: **Loop Filter**

The transfer function *F(s)* is given by:

$$F(s) = \frac{1 + s\tau_2}{1 + s(\tau_1 + \tau_2)}$$

If τ_2 is much smaller than τ_1, the following approximation can be used:

$$F(s) = \frac{1 + s\tau_2}{1 + s\tau_1}$$

$$\tau_1 = (R_1 + R_2)C$$

$$\tau_2 = R_2 \times C$$

Design Example

This design example is needed to track temperature related voltage changes. Normally these changes occur over long periods of time. The components were chosen to give an over damped response and a cut off frequency greater than a factor of 10 below the comparision frequency. A damping factor (ζ) of 0.707 is considered to be critically damped while values over 0.707 are considered to be overdamped. An overdamped response was chosen to avoid frequency ringing or hunting. In this design, the comparison frequency (250,000 Hz) is greater than 10 times the cutoff frequency (ω_n = 17,800), providing adequate filtering for the PLL operation.

This design example values are not intended to restrict the designer to a set of parameters. Rather, component values chosen are easy to obtain for the SMT footprints and also produce the desired results.

$$R_1 = 47,000\Omega$$

$$R_2 = 150\Omega$$

$$C = 1\mu F$$

$$K_O = 31.4 \times 10^{6}$$

$$K_d = 477 \times 10^{-3}$$

$$\tau_1 = 47.1 \times 10^{-3}$$

$$\tau_2 = 150 \times 10^{-6}$$

$$\omega_n = 17.8 \times 10^{3}$$

$$\zeta = 1.3$$

Where K_O is the VCO gain:

$$K_O = \frac{\Delta\omega}{\Delta V_C} = \frac{2\pi \times 10MHz}{2Volts}$$

K_d is the PD gain:

$$K_d = \frac{V_H - V_L}{2\pi} = \frac{3.2 - 0.2}{2\pi}$$

The natural frequency of the loop is:

$$\omega_n = \sqrt{\frac{K_O K_d}{\tau_1}}$$

The damping factor is:

$$\zeta = \frac{1}{2}\left(\sqrt{\frac{K_O K_d}{\tau_1}}\right) \times \left(\tau_2 + \frac{1}{K_O K_d}\right)$$

Reference Design

The VHDL and Verilog versions of the reference design is available on the Xilinx FTP site at ftp://ftp.xilinx.com/pub/applications/xapp/xapp250.zip.

Conclusion

Virtex-II devices are used for performing CDR functions by extracting DATA and CLOCK from encoded serial bit streams. A Virtex-E device can run at link speeds up to 200 Mb/s (SDR rates) and a Virtex-II device can run at 420 Mb/s with additional components added to the board.

References

1. Floyd M. Gardner, *Phaselock Techniques*, 2nd Edition, August 1979, John Wiley & Sons, ISBN 0-471-04294-3, page 92.

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
04/01/02	1.0	Initial Xilinx release.
04/25/02	1.1	Corrected equation for τ_1 on page 8.