Appendix A

Phase Lock Loops

A.1 Introduction

Since the phase lock loop (PLL) is the core component in many synchronization subsystems, it is important to understand how the PLL functions and how to analyze a PLL. This appendix summarizes the functionality and the methods commonly used to analyze a PLL. The material presented here is designed to complement the material contained in Chapter 3. The continuous-time PLL is analyzed first in Section A.2. These results are then applied to the discrete-time PLL analyzed in Section A.3.

A.2 The Continuous-Time PLL

PLLs are used in digital communication systems to synchronize the local oscillators in the receiver to the oscillators used by the transmitter (carrier phase synchronization) or to synchronize the data clock in the receiver to the data clock used at the data source (symbol timing synchronization). In either case, the PLL can be thought of as a device that tracks the phase and frequency of a sinusoid. A simple PLL that is designed to track the pahse of an input sinusoid is illustrated in Figure A.1. The PLL consists of three basic components: the phase detector, the loop filter, and the voltage controlled oscillator (VCO).

The input to the loop is the sinusoid

$$x(t) = A\cos\left(\omega_0 t + \theta(t)\right) \tag{A.1}$$

and the output of the VCO is

$$y(t) = \cos\left(\omega_0 t + \hat{\theta}(t)\right). \tag{A.2}$$

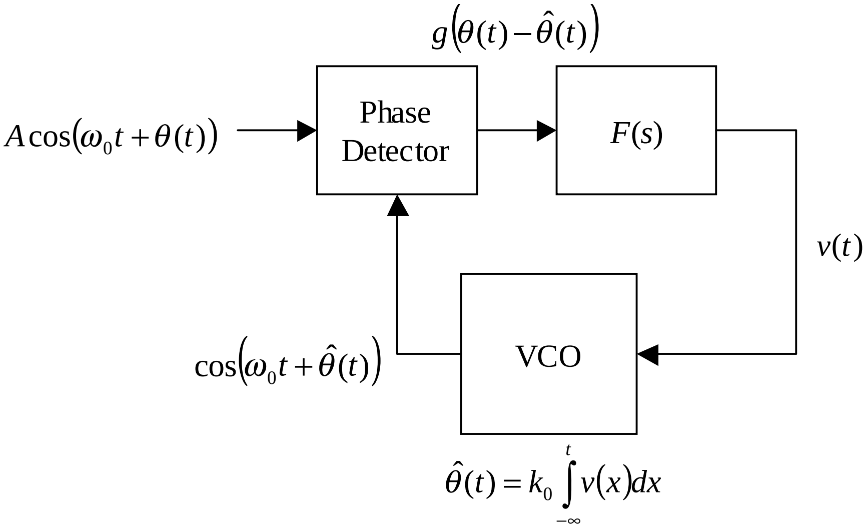

$g\left(\theta(t)-\hat{\theta}(t)\right)$
$A\cos\left(\omega_0 t+\theta(t)\right)$
Phase Detector
$F(s)$
$v(t)$
$\cos\left(\omega_0 t+\hat{\theta}(t)\right)$
VCO
$\hat{\theta}(t)=k_0\int_{-\infty}^{t}v(x)dx$

Figure A.1: Basic structure of a phase lock loop showing the three basic components: 1) the phase detector, 2) the loop filter, and 3) the voltage-controlled oscillator or VCO.

The phase detector is a device whose output is a function $g(\cdot)$ of the phase difference between the two inputs. Since the loop input and VCO output form the inputs to the phase detector, the phase detector output is $g\left(\theta(t)-\hat{\theta}(t)\right)$. The difference $\theta(t)-\hat{\theta}(t)$ is called the *phase error* and is denoted $\theta_e(t)$. The phase error is is filtered by the loop filter to produce a control voltage $v(t)$ which is used to set the phase of the VCO. The VCO output $y(t)=\cos\left(\omega_0 t+\hat{\theta}(t)\right)$ is related to the intput $v(t)$ via the phase relationship

$$\hat{\theta}(t)=k_0\int_{-\infty}^{t}v(x)dx \tag{A.3}$$

where k_0 is a constant of proportionality, called the VCO gain, that has units radians/volt.

When functioning properly, the loop adjusts the control voltage $v(t)$ to produce a phase estimate $\hat{\theta}(t)$ that drives the phase error to zero. To see why this is so, consider the plot of $g\left(\theta_e\right)$ versus θ_e for a typical phase detector[1] shown in Figure A.2. When the phase error is positive, $\theta(t)>\hat{\theta}(t)$ which means the phase of the VCO output lags the loop input and must be increased. The positive phase error produces a control voltage $v(t)$ that is also positive at the loop filter output. The positive control voltage increases the phase of the VCO output $\hat{\theta}(t)$ thus producing the desired result. Likewise, when the phase error is negative, $\theta(t)<\hat{\theta}(t)$. The negative phase error produces a negative control voltage which decreases the phase of the VCO output.

[1] Note the shape of the curve looks like the letter “S.” For this reason a plot of $g\left(\theta_e\right)$ versus θ_e is often called an *S-curve*.

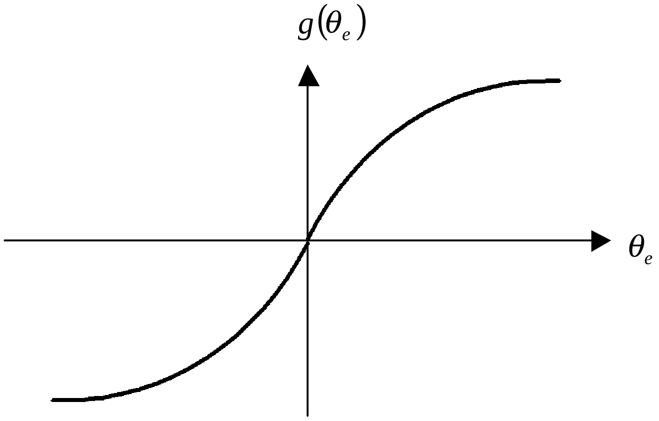


Figure A.2: Typical phase detector input-output relationship. Since the shape of the curve looks like the letter "S," this curve is called an "S Curve."

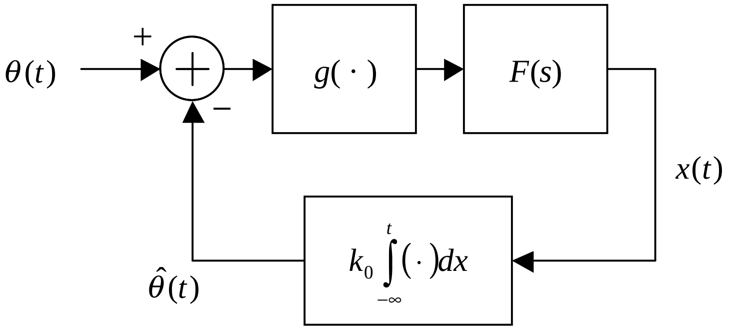


Figure A.3: Phase equivalent PLL corresponding to the PLL illustrated in Figure A.1.

The PLL of Figure A.1 has a *phase equivalent* representation as shown in Figure A.3. The phase equivalent PLL is derived from the actual PLL by writing down the phases of all the sinusoids and tracking the operations on those phases through the loop. Since the phase detector is, in general, an non-linear function of the phases of its inputs, the phase equivalent loop is a non-linear feedback control system that can be analyzed using advanced techniques.

A common method of analyzing a non-linear feedback control system is to linearize the system at a desired operating point. When this is done, the loop can be analyzed using standard linear system techniques. The desired operating point is at $\theta_e = 0$. In most cases, $g(\theta_e) \approx k_p \theta_e$ for small θ_e, so the loop is a linear feedback system when θ_e is small. In this case, the phase equivalent loop assumes the form shown in Figure A.4 (a). Since the phase equivalent loop is a linear system, it can be analyzed using LaPlace transform techniques. Computing the LaPlace transform of all signals and systems in the loop of Figure A.4 (a) produces the frequency domain version of the loop shown in Figure A.4 (b).

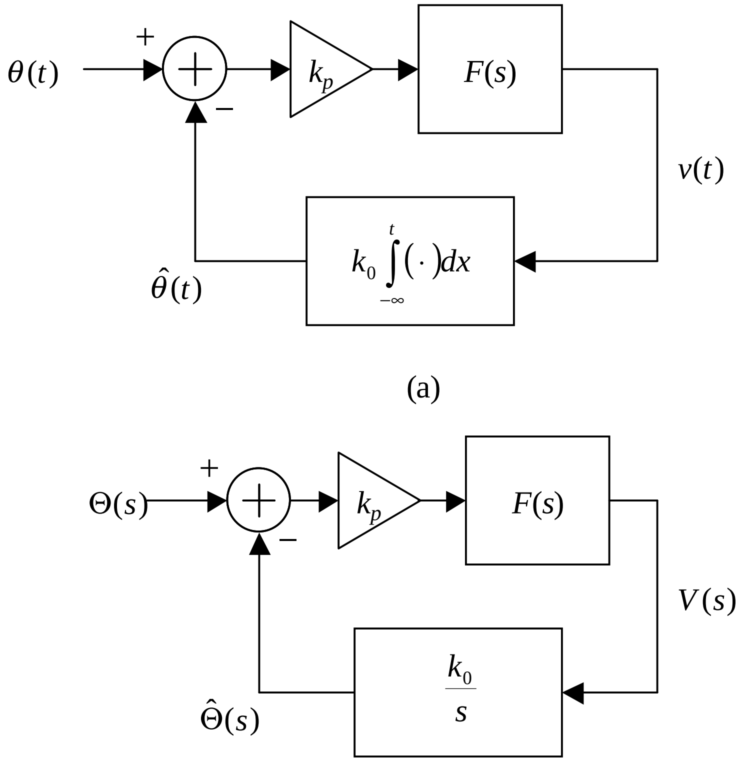


Figure A.4: Linearized phase equivalent PLL showing time-domaing signals (a) and frequency domain signals (b).

PLL operation is characterized using both the phase error $\theta_e(t)$ and the VCO output $\hat{\theta}(t)$. After some LaPlace domain algebra, the transfer functions for the phase error and VCO output phase are

$$H_a(s) = \frac{\Theta_e(s)}{\Theta(s)} = \frac{s}{s + k_0 k_p F(s)} \tag{A.4}$$

$$G_a(s) = \frac{\hat{\Theta}(s)}{\Theta(s)} = \frac{k_0 k_p F(s)}{s + k_0 k_p F(s)} \tag{A.5}$$

Ideally, the PLL should produce a phase estimate that has zero phase error. This characteristic, together with the phase error transfer function, will be used to determine the desirable properties of the loop filter. The phase estimate transfer function, or loop transfer function, is used to characterize the performance of the PLL.

A.2.1 PLL Inputs

Two PLL inputs are of special interest in characterizing PLL performance. The first represents the case where the loop input differs from the VCO output by a simple phase difference $\Delta\theta$. This is modeled by setting

$$\theta(t) = \Delta\theta u(t) \tag{A.6}$$

where $u(t)$ is the unit step function. This case is illustrated in Figure A.5 (a) where the argument of the input sinusoid, or instataneous phase, is plotted as a function of time. The slope of the line is the frequency of the input sinusoid. Note that it does not change. This means the frequency does not change. At time $t = 0$, a phase step occurs that changes the intercept point of the line. The loop outputs corresponding to the phase step input are called the *step responses* of the loop. The LaPlace transform of the input is

$$\Theta(s) = \frac{\Delta\theta}{s}. \tag{A.7}$$

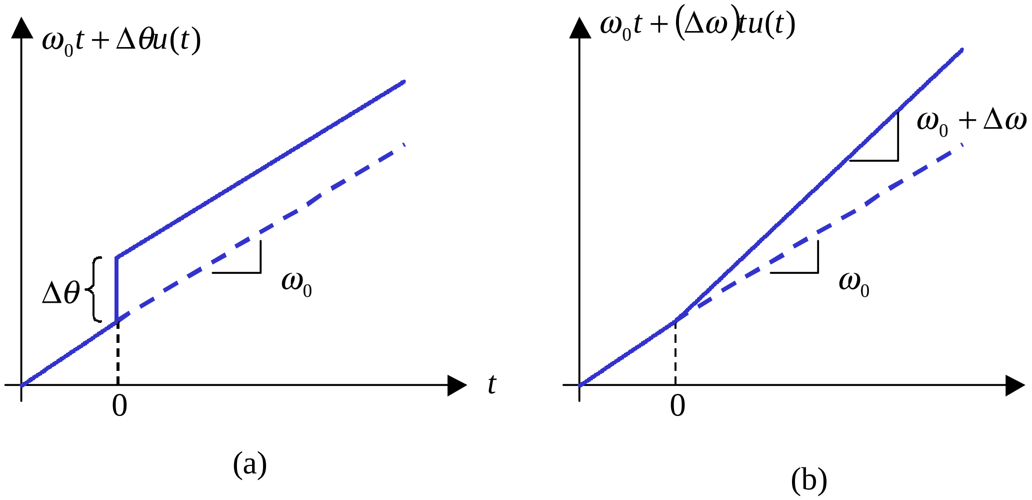
The second PLL input of interest is the case where the input sinusoid differs from the VCO output by a frequency shift of $\Delta\omega$ radians/second. The frequency offset is modeled by observing that

$$\cos\left((\omega_0 + \Delta\omega)t\right) = \cos(\omega_0 t + \underbrace{\Delta\omega t}_{\theta(t)}). \tag{A.8}$$

Thus,

$$\theta(t) = (\Delta\omega)tu(t) \tag{A.9}$$

which is called a “ramp” input. This case is illustrated in Figure A.5 (b) where the argument of the input sinusoid is plotted as a function of time. The slope of the instantaneous phase is instantaneous frequency. Prior to $t = 0$, the slope of the line is ω_0 and after $t = 0$ the slope fo the line is $\omega_0 + \Delta\omega$.



(a)

(b)

Figure A.5: Two phase inputs to be analyzed: (a) a phase step of $\Delta\theta$, (b) a frequency step of $\Delta\omega$ which is equivalent to a phase ramp with slope $\omega + \Delta\omega$.

This change in slope is the frequency change at $t = 0$. The loop outputs corresponding to the ramp input are called the *ramp responses* of the loop. The LaPlace transform of the ramp input is

$$\Theta(s) = \frac{\Delta\omega}{s^2}. \tag{A.10}$$

A.2.2 Phase Error and the Loop Filter

The properties of the phase error are used to determine the desirable characteristics of the loop filter. In particular, it will be shown that to achieve a zero steady-state phase error for a constant phase offset, the loop filter must have a non-zero DC gain and to achieve a zero staday state phase error for a constant frequency offset, the loop filter must have an infinte DC gain.

Since a constant phase offset is modeled by a step function for $\theta(t)$, the starting point is the derivation of an expression for the phase error step response. Substituting (A.7) in (A.4), and solving for $\Theta_e(s)$, the LaPlace transform of the phase error step response is

$$\Theta_{e,\text{step}}(s) = \frac{\Delta\theta}{s + k_0 k_p F(s)}. \tag{A.11}$$

The steady-state phase error may be obtained by computing the inverse LaPlace transform of (A.11) and taking the limit as $t \to \infty$. Unfortunately, this approach requires knowledge of the loop filter transfer function $F(s)$. An alternate approach is to use the final value theorem for

LaPlace transforms. Applying the final value theorem to (A.11), the steady state phase error is

$$\theta_{e,\text{step}}(\infty) = \lim_{s\to 0}\left\{s\Theta_e(s)\right\} \tag{A.12}$$

$$= \lim_{s\to 0}\left\{\frac{s\Delta\theta}{s+k_0k_pF(s)}\right\} \tag{A.13}$$

$$= 0 \quad \text{if } F(0)\neq 0. \tag{A.14}$$

This means that as long as the loop filter has a non-zero DC gain, the steady-state phase error step response is zero.

The expression for the steady state phase error for a constant frequency offset is obtaining following the same procedure except that a phase ramp is used instead of a phase step for the input. Substituting (A.10) into (A.4) and solving for $\Theta_e(s)$, the LaPlace transform of the phase error ramp response is

$$\Theta_{e,\text{ramp}}(s) = \frac{\Delta\omega}{s^2+sk_0k_pF(s)}. \tag{A.15}$$

Using the final value theorem, the steady-state phase error ramp response is

$$\theta_{e,\text{ramp}}(\infty) = \lim_{s\to 0}\left\{s\Theta_e(s)\right\} \tag{A.16}$$

$$= \lim_{s\to 0}\left\{\frac{\Delta\omega}{s+k_0k_pF(s)}\right\} \tag{A.17}$$

$$= 0 \quad \text{if } F(0)=\infty. \tag{A.18}$$

This means that as long as the loop filter has an infinte DC gain, the steady-state phase error step response is zero.

The two conditions (A.14) and (A.18) indicate that the loop filter must have a non-zero DC gain to drive the steady-state phase error to zero in the presence of a phase offset and an infinite DC gain to drive the steady-state phase error to zero in the presence of a frequency offset. A loop filter which satisfies these conditions is the *proportional-plus-integrator* loop filter[2] with transfer function

$$F(s) = k_1 + \frac{k_2}{s}. \tag{A.19}$$

The first term in (A.19) is a simple gain of k_1 and contributes, to the filter output, a signal that is proportional to the filter input. Hence the word "proportional" in the filter name. The second term in (A.19) is an ideal integrator with infinite memory and a gain k_2. This portion of the loop

[2] A commonly used filter in feedback control is the "proportional-integrator-differentiator" or PID filter. Feedback control systems that use a PID filter are commonly called PID controllers. PID controllers are the most commonly used form of linear feedback control.

filter contributes, to the filter output, a signal that is proportional to the integral of the input signal. Hence the word “integrator” in the filter name.

Not all PLLs use a proportional-plus-integrator loop filter. With continuous-time processing, it is challenging to produce an integrator with an infinite DC gain. Since the VCO is characterized by a single-pole element, the use of a loop filter with p poles will produce a PLL whose linear phase equivalent system is a $p+1$ order feedback control system. Thus a loop filter with no poles leads to a first order feedback system and a loop filter with one pole leads to a second order feedback control system. Since second order systems are easy to understood and have well developed design procedures, loop filters with one pole are preferred in most applications. As a consequence, loop filters of the form

$$F(s) = \frac{a_1 s + a_0}{b_1 s + b_0} \tag{A.20}$$

are examined. While all first and second order PLLs are special cases of the PLL that uses (A.20) as a loop filter, four special cases are of interest:

$F(s) = k$	($a_1 = b_1 = 0$):	This filter is a simple gain and produces a first order PLL.
$F(s) = \dfrac{k}{s+k}$	($a_1 = 0$):	This filter is a first order low-pass filter where $1/k$ is the time constant. It is often called a “leaky integrator.” It produces a second order PLL.
$F(s) = k_1 + \dfrac{k_2}{s}$	($b_0 = 0$):	This is the proportional-plus-integrator filter introduced earlier. It produces a second order PLL.
$F(s) = \dfrac{k_1 + s}{k_2 + s}$	($k_1 = a_1/a_0$, $k_2 = b_1/b_0$):	This is the general first order filter expressed in a more convenient form. It produces a second order PLL.

The properties of these filters and the resulting steady-state phase errors for step and ramp inputs are summarized in Table A.1. All of the filters produce a PLL with zero steady-state phase error for a step input. However, only the proportional-plus-integrator loop filter produces a PLL with zero steady-state phase error for a ramp input.

There are design techniques for higher order systems, but are more difficult than the design techniques for second order systems. One application where higher order systems are preferred is GPS receivers. The signal received from a satelliate in low-earth orbit experiences a Doppler shift (or carrier frequency offset) that is not constant in time, but linear in time (i.e., $\theta(t) \sim t^2$). In order

Table A.1: Summary of loop filter characteristics and steady state phase errors.

$F(s)$	$F(0)$	$\theta_{e,\text{step}}(\infty)$	$\theta_{e,\text{ramp}}(\infty)$
k	k	0	$\dfrac{\Delta\omega}{k_0 k_p k}$
$\dfrac{k}{s+k}$	1	0	$\dfrac{\Delta\omega}{k_0 k_p}$
$k_1 + \dfrac{k_2}{s}$	∞	0	0
$\dfrac{k_1+s}{k_2+s}$	$\dfrac{k_1}{k_2}$	0	$\dfrac{\Delta\omega k_2}{k_0 k_p k_1}$

to track the frequency offset with zero phase error, a loop filter with two poles is required. The resulting PLL is modeled by a third order feedback control system.

A.2.3 Transient Analysis

The loop performance is characterized by the properties of the loop output $\hat{\theta}(t)$. The LaPlace transform of the loop output is given by

$$\hat{\Theta}(s) = \frac{k_0 k_p F(s)}{s + k_0 k_p F(s)} \Theta(s) \tag{A.21}$$

where

$$\Theta(s) = \begin{cases} \dfrac{\Delta\theta}{s} & \text{for a phase offset} \\ \dfrac{\Delta\omega}{s^2} & \text{for a frequency offset} \end{cases} \tag{A.22}$$

and $F(s)$ is one of the loop filters in Table A.1. When $F(s) = k$, the loop transfer function is

$$H_a(s) = \frac{k_0 k_p k}{s + k_0 k_p k}. \tag{A.23}$$

The step response is

$$\hat{\Theta}(s) = \frac{k_0 k_p k \Delta\theta}{s\left(s + k_0 k_p k\right)} \tag{A.24}$$

$$\hat{\theta}(t) = \Delta\theta \left(1 - e^{-k_0 k_p k t}\right) u(t) \tag{A.25}$$

and the ramp response is

$$\hat{\Theta}(s) = \frac{k_0 k_p k \Delta\omega}{s^2\left(s + k_0 k_p k\right)} \tag{A.26}$$

$$\hat{\theta}(t) = \left[\Delta\omega t - \frac{\Delta\omega}{k_0 k_p k}\left(1 - e^{-k_0 k_p k t}\right)\right] u(t). \tag{A.27}$$

The loop transfer functions for the other three filters are of the form

$$H_a(s) = \frac{b_2 s^2 + b_1 s + b_0}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{A.28}$$

and are thus second order systems. The poles of the system are

$$p_1, p_2 = -\zeta\omega_n \pm \omega_n\sqrt{\zeta^2 - 1}. \tag{A.29}$$

The characteristics of the loop output are governed by the value of ζ. When $\zeta < 1$, the poles are complex conjugate pairs and the loop response exhibits damped oscillations. For this reason, ζ is called the *damping factor* and ω_n is called the *natural frequency*. A second order system damped sinusoidal transients is termed *underdamped*. When $\zeta > 1$, the poles are real and distinct and loop response is the sum of decaying exponentials. Such a systems is *overdamped*. When $\zeta = 1$, the poles are real and repeated and the loop responses are on the boundary between damped oscillations and decaying exponentials. The PLL is *critically damped* in this case.

As an example, the loop transfer function for the “proportional-plus-integrator” loop filter is

$$H_a(s) = \frac{k_0 k_p k_1 s + k_0 k_p k_2}{s^2 + k_0 k_p k_1 s + k_0 k_p k_2} \tag{A.30}$$

which may be expressed in the form

$$H_a(s) = \frac{2\zeta\omega_n s + \omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{A.31}$$

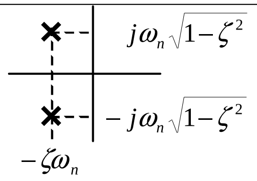
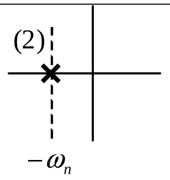
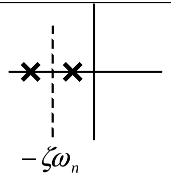
where

$$\zeta = \frac{k_1}{2}\sqrt{\frac{k_0 k_p}{k_2}} \tag{A.32}$$

$$\omega_n = \sqrt{k_0 k_p k_2}. \tag{A.33}$$

The step response as a function of ζ is summarized in Table A.2.3. The frequency response is plotted in Figure A.6 for four different values of the damping factor ($\zeta = 0.5, 0.7071, 1, 2$). The frequency response shows that the PLL acts as a low-pass filter when operating as a linear system.

Table A.2: Summary of pole locations and corresonding loop responses for a second order loop.

damping factor	poles	step response
$\zeta < 1$ (underdamped)	 $j\omega_n\sqrt{1-\zeta^2}$ $-j\omega_n\sqrt{1-\zeta^2}$ $-\zeta\omega_n$ complex conjugates	$\Delta\theta\left(1-\left[\cos\left(\sqrt{1-\zeta^2}\omega_n t\right)-\frac{\zeta}{\sqrt{1-\zeta^2}}\sin\left(\sqrt{1-\zeta^2}\omega_n t\right)\right]e^{-\zeta\omega_n t}\right)u(t)$
$\zeta = 1$ (critically damped)	 real & repeated	$\Delta\theta\left[1-e^{-\omega_n t}+\omega_n t e^{-\omega_n t}\right]u(t)$
$\zeta > 1$ (overdamped)	 real & distinct	$\Delta\theta\left(1-\left[\cosh\left(\sqrt{1-\zeta^2}\omega_n t\right)-\frac{\zeta}{\sqrt{1-\zeta^2}}\sinh\left(\sqrt{1-\zeta^2}\omega_n t\right)\right]e^{-\zeta\omega_n t}\right)u(t)$

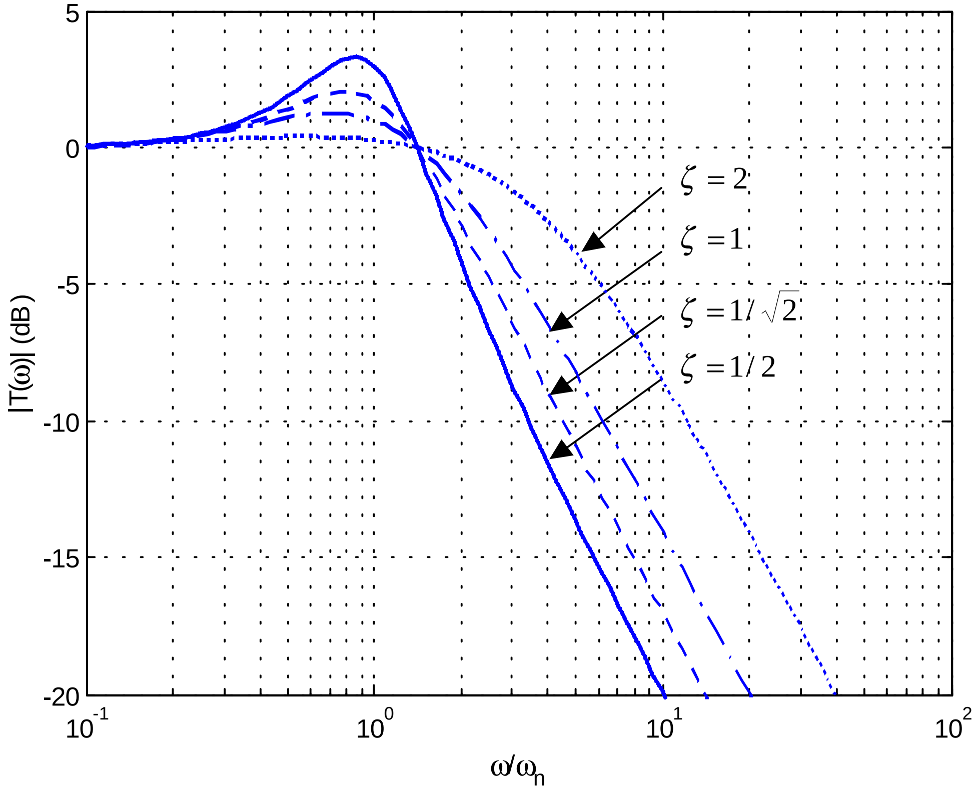


Figure A.6: The frequency response of the second order PLL with transfer function (A.31) for $\zeta = 0.5, 1/\sqrt{2}, 1, 2$.

The damping factor controls the magnitude of the frequency response at $\omega = \omega_n$. For underdamped systems ($\zeta < 1$), the frequency response displays a peak at $\omega = \omega_n$. The amplitude of this peak grows as ζ decreases. In fact, as $\zeta \rightarrow 0$, the peak grows to an impulse and the loop response is purely sinusoidal[3]. For overdamped systems, the peak at $\omega = \omega_n$ disappears and the loop response looks more like a traditional low-pass filter.

The effect of the spectral peak in $H_a(j\omega)$ is best observed by examining the step response in the time domain as illustrated in Figure A.7. The step response for the same four values of ζ are shown by the heavy lines and the step input is the thin dotted line. The ideal loop response is one that follows the input exactly. For small values of ζ, the step response overshoots the ideal response and oscillates about the ideal response. The amplitudes of the oscillations decrease with time so that in the steady state, the loop output matches the loop input. Thus, the presence of a spectral peak in $H_a(j\omega)$ produces a damped sinusoidal component in the time-domain loop response. For

[3] This is how an oscillator is designed using a second order feedback system. However, the goal is to build synchronizers, not oscillators. Underdamped systems with $\zeta = 0$ will not be examined further.

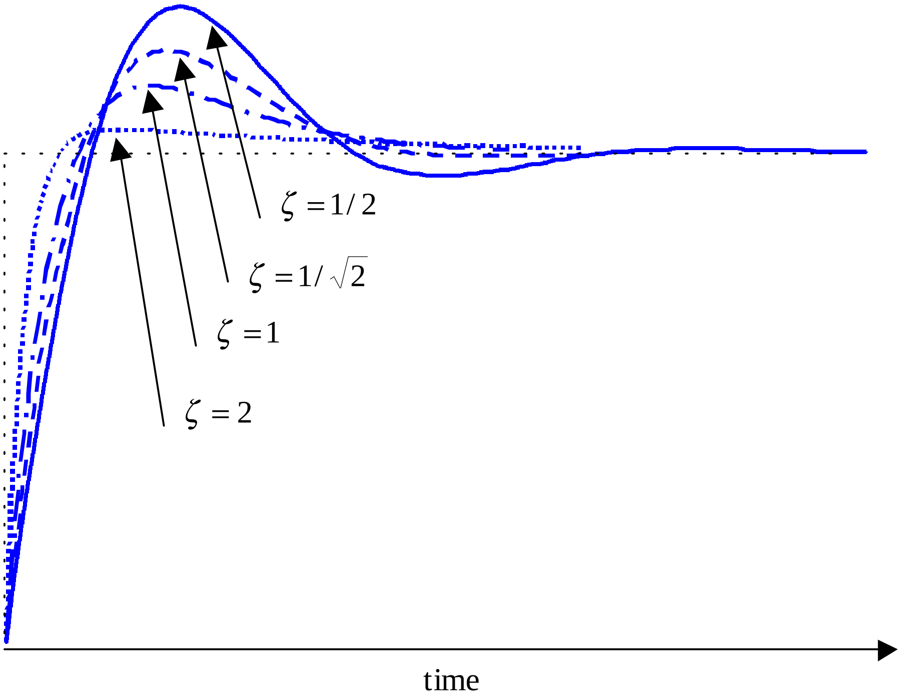


Figure A.7: Step response for the second order PLL for $\zeta = 0.5, 1/\sqrt{2}, 1, 2$.

overdamped systems ($\zeta > 1$), the step response does not overshoot the ideal response but gradually approaches the ideal response as time increases. This demonstrates the classic trade-off for second order systems: underdamped systems have a fast rise-time, but overshoot and oscillations; overdamped systems have a slow rise time, but no overshoot and no oscillations.

A.2.4 PLL Bandwidth

Another way to characterize the PLL is to compute the bandwidth of the frequency response $H_a(j\omega)$. The loop bandwidth is a function of both ζ and ω_n. The 3-dB bandwidth ω_{3dB} is obtained by setting $|H_a(j\omega)|^2 = 1/2$ and solving for ω. The 3-dB bandwidth is

$$\omega_{\text{3dB}} = \omega_n \sqrt{1 + 2\zeta^2 + \sqrt{(1 + 2\zeta^2)^2 + 1}}. \tag{A.34}$$

While the 3-dB bandwidth is a familiar concept, it is not a very useful measure of bandwidth for a PLL. A more useful measure of bandwidth is the the equivalent noise bandwidth B_n. The equivalent noise bandwidth of a linear system with transfer function $H_a(j2\pi f)$ is the bandwidth (measured in Hz) of a fictitous rectangular low-pass filter with the same area as $|H_a(j2\pi f)|^2$. Calculation of the noise bandwidth is illustrated in Figure A.8. The magnitude-squared of the

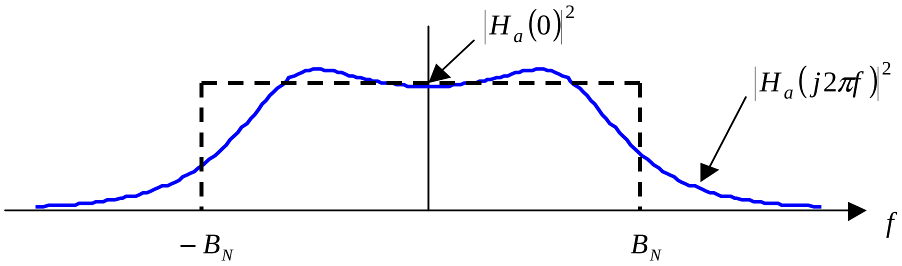


Figure A.8: Conceptual diagram for the calculation of noise bandwidth.

transfer function $|H_a(j2\pi f)|^2$ is represented by the solid line and the area under $|H_a(j2\pi f)|^2$ is obtained by integrating $|H_a(j2\pi f)|^2$ over the frequency variable f:

$$\text{area} = \int_{-\infty}^{\infty} |H_a(j2\pi f)|^2 df. \tag{A.35}$$

The fictitious retangular low-pass filter is represented by the dashed line in Figure A.8. Fixing the height of this filter at $|H_a(0)|^2$ and the width at $2 \times B_n$, the area under this filter is $2B_n|T(0)|^2$. Equating the two areas and solving for B_n produces the desired expression for noise bandwidth:

$$B_n = \frac{1}{2|H_a(0)|^2} \int_{-\infty}^{\infty} |H_a(j2\pi f)|^2 df. \tag{A.36}$$

For the “proportional-plus-integrator” loop filter, the noise bandwidth is

$$B_n = \frac{\omega_n}{2}\left(\zeta + \frac{1}{4\zeta}\right). \tag{A.37}$$

The noise bandwidth B_n is a non-linear function of ζ. For fixed ω_n, B_n assumes a minimum of $\omega_n/2$ at $\zeta = 1/2$ and is larger for all other values of ζ. For $\zeta \geq 1/2$, B_n grows monotonically with ζ.

The transfer functions and equivalent noise bandwidth of PLLs using the four loop filters in Table A.1 are summarized in Talle A.3

A.2.5 Acquisition

Any PLL requires a non-zero period of time to reduce a phase error to zero. During the initial stages of acquisition, the input voltage to the VCO is adjusted to produce an output whose frequency matches that of the input. This initial phase of the acquisition process is called *frequency lock*. Once frequency lock is achieved, an additional period of time is required to reduce the loop phase error to an acceptably low level. This phase of the acquisition process is called *phase lock*.

Table A.3: Transfer function, loop parameters and equivalent noise bandwidth for a phase lock loop using the four loop filters of interest.

Loop Filter $F(s)$	ζ	ω_n	$H_a(s)$	B_n (Hz)
k	–	–	$\dfrac{k_0 k_p k}{s + k_p k_0 k}$	$\dfrac{k_0 k_p k k}{4}$
$\dfrac{k}{k+s}$	$\dfrac{1}{2}\sqrt{\dfrac{k}{k_0 k_p}}$	$\sqrt{k_0 k_p k}$	$\dfrac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$	$\dfrac{\omega_n}{8\zeta}$
$k_1 + \dfrac{k_2}{s}$	$\dfrac{k_1}{2}\sqrt{\dfrac{k_0 k_p}{k_2}}$	$\sqrt{k_0 k_p k_2}$	$\dfrac{2\zeta\omega_n s + \omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$	$\dfrac{\omega_n}{2}\left(\dfrac{1}{4\zeta} + \zeta\right)$
$\dfrac{k_1 + s}{k_2 + s}$	$\dfrac{1}{2}\dfrac{k_0 k_p + k_2}{\sqrt{k_0 k_p k_1}}$	$\sqrt{k_0 k_p k_1}$	$\dfrac{\omega_n^2 + \omega_n\left(2\zeta - \dfrac{k_2}{\omega_n}\right)}{s^2 + 2\zeta\omega_n s + \omega_n^2}$	$\dfrac{\omega_n}{8\zeta}\left[1 + \left(2\zeta - \dfrac{k_2}{\omega_n}\right)^2\right]$

Consequently, the acquisition time, T_{LOCK} is well approximated by the sum of the time to achieve freqeuncy lock T_{FL} and the time to achieve phase lock T_{PL}:

$$T_{\text{LOCK}} \approx T_{\text{FL}} + T_{\text{PL}} \tag{A.38}$$

where

$$T_{\text{FL}} \approx 4\frac{(\Delta f)^2}{B_N^3} \tag{A.39}$$

$$T_{\text{PL}} \approx \frac{1.3}{B_N}. \tag{A.40}$$

Δf in Equation (A.39), is the frequency offset (in Hz) and B_n in Equations (A.39) and (A.40) is the noise bandwidth. It is possible for the frequency offset Δf to be so large that the loop can never acquire lock. The range of frequency offsets for which the loop can acquire lock is called the *pull-in* range $(\Delta f)_{\text{pull-in}}$, and is well approximated by

$$(\Delta f)_{\text{pull-in}} \approx \left(2\pi\sqrt{2}\zeta\right) B_n \tag{A.41}$$

A.2.6 Tracking

Tracking performance is quantified by the variance of the phase error. Conceptually, the phase error variance, $\sigma^2_{\theta_e}$, is

$$\sigma^2_{\theta_e} = \text{E}\left\{\left|\theta - \hat{\theta}\right|\right\}. \tag{A.42}$$

A linear PLL which has a sinusoidal input with power P_{in} W together with additive white Gaussian noise with power spectral density $N_0/2$ W/Hz, the phase error variance is

$$\sigma^2_{\theta_e} = \frac{N_0 B_n}{P_{\text{in}}}. \tag{A.43}$$

Since the noise power at the PLL input (within the frequency band of interest to the PLL) is $N_0 B_n$, the ratio $P_{\text{in}}/N_0 B_n$ often called the loop signal to noise ratio. Thus, for a linear PLL with addtive white Gaussian noise, the phase error variance is inversely proportional to the loop signal to noise ratio.

Equations (A.39) and (A.40) indicate that acquisition time is inversely proportional to a power of B_n. This suggests that the larger equivalent loop bandwidth, the faster the acquisition. Equation (A.43) shows that the tracking error is proportional to B_n. This suggests that the smaller the equivalent loop bandwidth, the smaller the tracking error. Thus fast acquisition and good tracking place competing demands on PLL design. Acquisition time can be decreased at the expense of increased tracking error. Tracking error can be decreased at the expense of increased acquisition time. A good design balances the two performance criteria. Where that balance is depends on the application, the signal to noise level, and system-level performance specifications.

A.2.7 Loop Constant Selection

Appropriate choice of loop constants is a critical step in PLL design. Most often, PLL design specifications define the nature of the response (i.e. underdamped, critically damped, or overdamped) and the noise bandwidth B_n. The procedure is demonstrated using the proportional-plus-integrator loop filter. The resulting second order loop, with transfer function given by (A.30), or the equivalent canonical form (A.31), offers the designer two degrees of freedom: the damping constant and the natural frequency. But there are four loop constants (k_0, k_p, k_1, k_2) that must be selected. The loop constants, however, are not all independent. Solving equations (A.32) and (A.33) for the loop constants gives

$$\begin{aligned} k_0 k_p k_1 &= 2\zeta\omega_n \\ k_0 k_p k_2 &= \omega_n^2 \end{aligned} \tag{A.44}$$

Observe that the VCO sensitivity k_0 and the phase detector gain k_p always occur as a pair. Since ζ is a parameter chosen by the designer, all that remains is to find an expression for ω_n. Solving (A.37) for ω_n gives

$$\omega_n = \frac{2B_n}{\zeta + \dfrac{1}{4\zeta}} \tag{A.45}$$

Substituting (A.45) for ω_n in (A.44) produces the desired expression for the loop constants in terms of the damping constant and the equivalent noise bandwidth:

$$\begin{aligned} k_p k_0 k_1 &= \frac{4\zeta B_n}{\zeta + \dfrac{1}{4\zeta}} \\ k_p k_0 k_2 &= \frac{4B_n^2}{\left(\zeta + \dfrac{1}{4\zeta}\right)^2} \end{aligned} \tag{A.46}$$

In many cases, the phase detector is the most complicated component in the loop and possesses a gain, k_p that cannot be adjusted. So, for a given phase detector, k_p can be treated as fixed. The other three loop constants should be adjusted to satisfy the relationships (A.46). One approach is to fix the VCO sensitivity at a value that ensures good VCO performance and adjust the filter constants k_1 and k_2 to obtain the desired loop response.

As an example, consider the design of a critically damped PLL with a noise bandwidth of 25 Hz that uses a proportional-plus-integrator loop filter. Using the relations (A.46) with $\zeta = 1$ and $B_n = 25$ produces the desired conditions for the loop constants:

$$\begin{aligned} k_0 k_p k_1 &= 80 \\ k_0 k_p k_2 &= 1600 \end{aligned} \tag{A.47}$$

A.3 Discrete-Time Phase Lock Loops

Discrete-time PLLs are used in sampled data systems. Typically the design of a discrete-time PLL begins with a continuous-time PLL design and uses a continuous-time to discrete-time transformation to produce the discrete-time PLL. The design process begins with the continuous-time PLL since there are well developed design techniques for continuous-time systems. The continuous-time to discrete-time transformations are intended to produce a discrete-time system with the same behavior as the continuous-time system. Some of the more popular continuous-time to discrete-time transformations include the following.

- Impulse Invariance: In this method, the descrete-time system is designed to have an impulse response that is equal to T-spaced samples of the impulse response of the continuous-time system. If $h_a(t)$ is the impulse response of the continuous-time system, then the impulse response of the discrete-time system is $h_d(n) = h_a(nT)$.

- Step Invariance: In this method, the descrete-time system is designed to have a step response that is equal to T-spaced samples of the step response of the continuous-time system. If $s_a(t)$ is the step response of the continuous-time system, then the step response of the discrete-time system is $s_d(n) = s_a(nT)$.

- Forward Difference: This method uses Simpson's rule to approximate integration as described in Chapter **??**. This transformation uses the substitution

$$\frac{1}{s} \rightarrow T \frac{z^{-1}}{1 - z^{-1}}. \tag{A.48}$$

- Bilinear Tranformation: This method uses the trapezoid rule to approximate integration as described in Chapter **??**. This transformation uses the substition, sometimes called Tustin's equation

$$\frac{1}{s} \rightarrow \frac{T}{2} \frac{1 + z^{-1}}{1 - z^{-1}}. \tag{A.49}$$

The problem with applying these approaches directly to the linearized phase equivalent PLL is that the linear phase equivalent PLL is only a model of the actual PLL. For example, in a real system the actual phases $\theta(t)$ and $\hat{\theta}(t)$ are not available. What is available to the system are samples of the input sinusoid and the discrete-time DDS output.

As a consequence, a hybrid approach will be demonstrated here. This approach assumes a discrete-time PLL whose structure mimics that of the continuous-time PLL. The starting point is the continuous-time PLL and its phase-equivalent loop are illustrated in Figure A.9. The discrete-time PLL that mimics the continuous-time PLL of Figure A.9 is illustrated in Figure A.10. In Figure A.10 (a), the integral portion of the loop filter uses a simple filter with a pole at $z = 1$. The discrete-time PLL uses a direct digital synthesizer (DDS) in place of the VCO. A single-pole filter is used to integrate the DDS input to calculate the instanteous phase. Note the use of upper case letters for the phase detector gain, loop filter constants, and DDS gain to distinguish them from their counterparts in the continuous-time PLL. The phase equivalent discrete-time loop is illustrated in Figure A.10 (b).

The loop transfer functions for the continuous-time loop of Figure A.9 and the discrete-time loop of Figure A.10 are

$$H_a(s) = \frac{2\zeta\omega_n s + \omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{A.50}$$

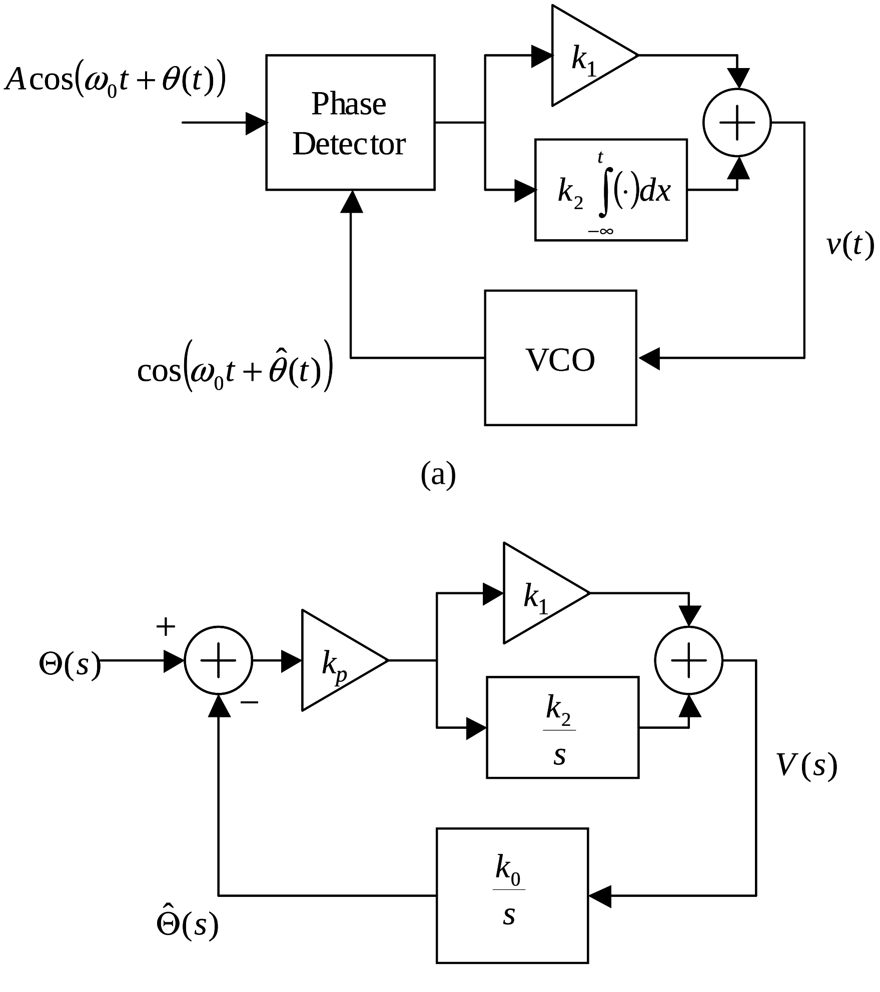
$$H_d(z) = \frac{K_pK_0(K_1+K_2)z^{-1} - K_pK_0K_1z^{-2}}{1 - 2\left(1 - \frac{1}{2}K_pK_0(K_1+K_2)\right)z^{-1} + \left(1 - K_pK_0K_1\right)z^{-2}}, \tag{A.51}$$

where ζ and ω_n in (A.50) are given by (A.32) and (A.33), respectively. Applying Tustin's equation to $H_a(s)$ produces a discrete-time version of the continuous-time loop denoted $H_a\left(\frac{2}{T}\frac{1-z^{-1}}{1+z^{-1}}\right)$. After some algebra, $H_a\left(\frac{2}{T}\frac{1-z^{-1}}{1+z^{-1}}\right)$ may be expressed as

$$H_a\left(\frac{2}{T}\frac{1-z^{-1}}{1+z^{-1}}\right) = \frac{\dfrac{2\zeta\theta_n + \theta_n^2}{1+2\zeta\theta_n+\theta_n^2} + 2\dfrac{\theta_n^2 - \zeta\theta_n}{1+2\zeta\theta_n+\theta_n^2}z^{-1} + \dfrac{\theta_n^2}{1+2\zeta\theta_n+\theta_n^2}z^{-2}}{1 - 2\dfrac{\theta_n^2-1}{1+2\zeta\theta_n+\theta_n^2}z^{-1} + \dfrac{1-2\zeta\theta_n+\theta_n^2}{1+2\zeta\theta_n+\theta_n^2}z^{-2}} \tag{A.52}$$

where

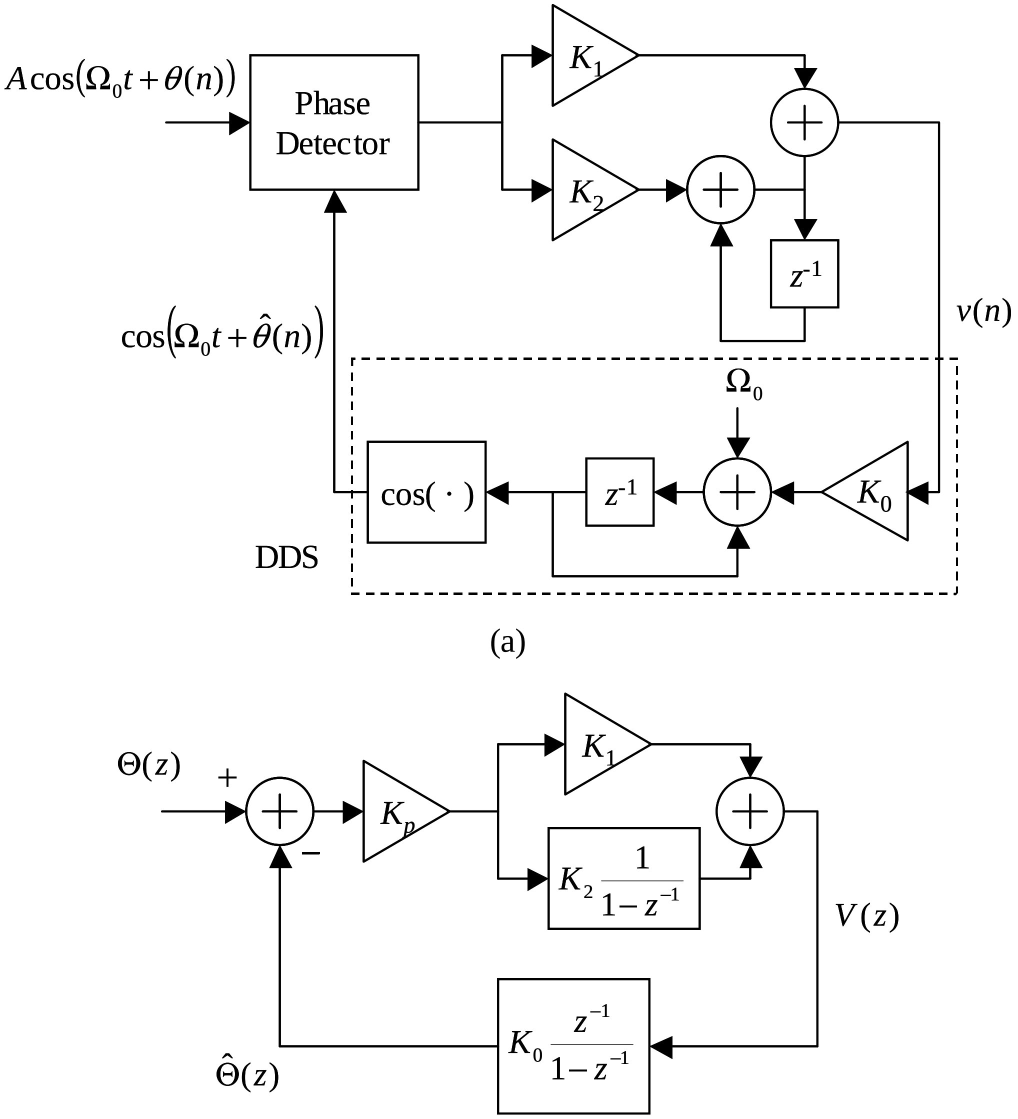
$$\theta_n = \frac{\omega_n T}{2}. \tag{A.53}$$



(a)

(b)

Figure A.9: Second order continuous-time phase lock loop with proportional-plus-integrator loop filter (a) and the linearized phase equivalent PLL (b).



(a)

(b)

Figure A.10: Second order discrete-time phase lock loop with that mimics the continuous-time PLL illustrated in Figure A.9: (a) the discrete-time PLL with proportional-plus-integrator loop filter and DDS; (b) the linearized phase equivalent discrete-time PLL.

Equating the denominator polynomials in $H_d(z)$ and $H_a\left(\frac{2}{T}\frac{1-z^{-1}}{1+z^{-1}}\right)$ produces the following relationships:

$$\begin{aligned} 1-\frac{1}{2}K_pK_0(K_1+K_2) &= \frac{\theta_n^2-1}{1+2\zeta\theta_n+\theta_n^2} \\ 1-K_pK_0K_1 &= \frac{1-2\zeta\theta_n+\theta_n^2}{1+2\zeta\theta_n+\theta_n^2}. \end{aligned} \tag{A.54}$$

Finally, solving for the loop constants gives

$$\begin{aligned} K_pK_0K_1 &= \frac{4\zeta\theta_n}{1+2\zeta\theta_n+\theta_n^2} \\ K_pK_0K_2 &= \frac{4\theta_n^2}{1+2\zeta\theta_n+\theta_n^2} \end{aligned} \tag{A.55}$$

The expressions for the loop constants are a function of the damping ratio ζ, the natural frequency θ_n, and the sampling period T. (The functional dependence on θ_n and T is through θ_n.) The loop constants can be expressed in terms of the ζ, T, and equivalent loop bandwidth B_n by solving the expression for B_n for ω_n, expressing θ_n in terms of B_n and substituting into (A.55). For the proportional-plus-integrator loop filter, B_n is related to ω_n by (A.37). Solving for ω_n and substituting produces

$$\theta_n = \frac{B_nT}{\zeta+\dfrac{1}{4\zeta}}. \tag{A.56}$$

Substituting into the relations (A.55) produces

$$\begin{aligned} K_0K_pK_1 &= \frac{4\zeta\left(\dfrac{B_nT}{\zeta+\dfrac{1}{4\zeta}}\right)}{1+2\zeta\left(\dfrac{B_nT}{\zeta+\dfrac{1}{4\zeta}}\right)+\left(\dfrac{B_nT}{\zeta+\dfrac{1}{4\zeta}}\right)^2} \\ K_0K_pK_2 &= \frac{4\left(\dfrac{B_nT}{\zeta+\dfrac{1}{4\zeta}}\right)^2}{1+2\zeta\left(\dfrac{B_nT}{\zeta+\dfrac{1}{4\zeta}}\right)+\left(\dfrac{B_nT}{\zeta+\dfrac{1}{4\zeta}}\right)^2} \end{aligned} \tag{A.57}$$

Note that when the equivalent loop bandwidth is small relative to the sample rate, $B_nT << 1$ so that the relations (A.57) are well approximated by

$$\begin{aligned} K_0K_pK_1 &\approx \frac{4\zeta}{\zeta + \dfrac{1}{4\zeta}}\left(B_nT\right) \\ K_0K_pK_2 &\approx \frac{4}{\left(\zeta + \dfrac{1}{4\zeta}\right)^2}\left(B_nT\right)^2. \end{aligned} \tag{A.58}$$

Comparing (A.58) with (A.46) shows that for the case where the sample rate is large relative to the loop equivalent bandwidth, the expressions for the loop filter constants for the discrete-time loop are the same as those for the continuous-time loop except that the loop bandwidth is normalized by the sample rate.

This design procedure requires the noise bandwidth be specified relative to the sample rate $1/T$. In digital communications, it is common practice to specify the the noise bandwidth relative to the symbol rate $1/T_s$ (i.e. B_nT_s is specified instead of B_nT or B_n). The expressions (A.57) can be adjusted to account for this by using $N = T_s/T$ so that

$$\theta_n = \frac{B_NT_s}{N\left(\zeta + \dfrac{1}{4\zeta}\right)}. \tag{A.59}$$

Substituting (A.59) into (A.55) produces

$$\begin{aligned} K_0K_pK_1 &= \frac{\dfrac{4\zeta}{N}\left(\dfrac{B_nT_s}{\zeta + \dfrac{1}{4\zeta}}\right)}{1 + \dfrac{2\zeta}{N}\left(\dfrac{B_nT_s}{\zeta + \dfrac{1}{4\zeta}}\right) + \left(\dfrac{B_nT_s}{N\left(\zeta + \dfrac{1}{4\zeta}\right)}\right)^2} \\ K_0K_pK_2 &= \frac{\dfrac{4}{N^2}\left(\dfrac{B_nT_s}{\zeta + \dfrac{1}{4\zeta}}\right)^2}{1 + \dfrac{2\zeta}{N}\left(\dfrac{B_nT_s}{\zeta + \dfrac{1}{4\zeta}}\right) + \left(\dfrac{B_nT_s}{N\left(\zeta + \dfrac{1}{4\zeta}\right)}\right)^2} \end{aligned} \tag{A.60}$$

A.3.1 Examples

Two examples are presented to illustrate ways a phase detector could be realized in a discrete-time system. The first example is the PLL illustrated in Figure A.11. This PLL is designed to track the phase of a complex exponential. The DDS output is also a complex exponential whose phase is the loop estimate of the input phase. The DDS produces a complex exponential by using a cosine and sine lookup table to output the real and imaginary parts, respectively. The phase detector computes the phase error by first computing the product of the input and the complex conjugate of the DDS output. This product is given by $\exp\left\{j\left(\theta(n)-\hat{\theta}(n)\right)\right\}$. The $\arg\{\cdot\}$ function computes the phase of the complex exponential using a four-quadrant arctangent. The output of the phase detector is thus $g\left(\theta_e(n)\right)=\theta_e(n)=\theta(n)-\hat{\theta}(n)$. The corresponding S-curve is shown in the lower portion of Figure A.11. Note that this phase detector is linear over the interval $-\pi\le\theta_e\le\pi$ and has a slope of one. For this reason, $K_p=1$. The loop filter is the discrete-time version of the familiar proportional-plus-integrator loop.

The loop responses are illustrated in Figure A.12 for an input with

$$\Omega_0=\frac{2\pi}{10} \tag{A.61}$$

$$\theta(n)=\pi u(n) \tag{A.62}$$

The real part of the loop input is illustrated by the dashed line in the top plot of Figure A.12. The loop filter constants were chosen to produce a critically damped loop with $B_nT=0.05$. Using (A.57) produces $K_1=0.1479$ and $K_2=0.0059$. The real part of the loop output (which is the DDS output) is illustrated by the solid line in the top plot of Figure A.12. Note that in the beginning, the two sinusoids are π radians out of phase. As time progresses, DDS output becomes phase-aligned with the PLL input. The phase error $\theta_e(n)$ is plotted in the middle plot of Figure A.12. The phase error starts at π radians, decreases to -0.5 radians then slowly approaches 0. The bottom plot of Figure A.12 illustrates the arugments of the two complex exponentials. The dashed line is the argument of the input sinusoid, $\Omega_0 n+\pi u(n)$, which is the equation of a straight line with slope Ω_n and intercept π. The solid line is the argument of the complex exponential at the output of the DDS. It starts at 0 and eventially aligns itself with the PLL input. The difference between these two curves is the phase error, which is illustrated in the middle plot of this Figure. Note that the solid line is below the dashed line for samples 0 through 12. This means the phase error is positive as illustrated in the phase error plot. For samples 13 though 75, the solid line is above the dashed line which means the phase error is negative. This is is also confirmed by the phaer error plot. The fact that the DDS output matches the PLL input is confirmed in all three plots in Figure A.12.

In the second example, a real-valued discrete-time PLL is explored. This PLL is illustrated in

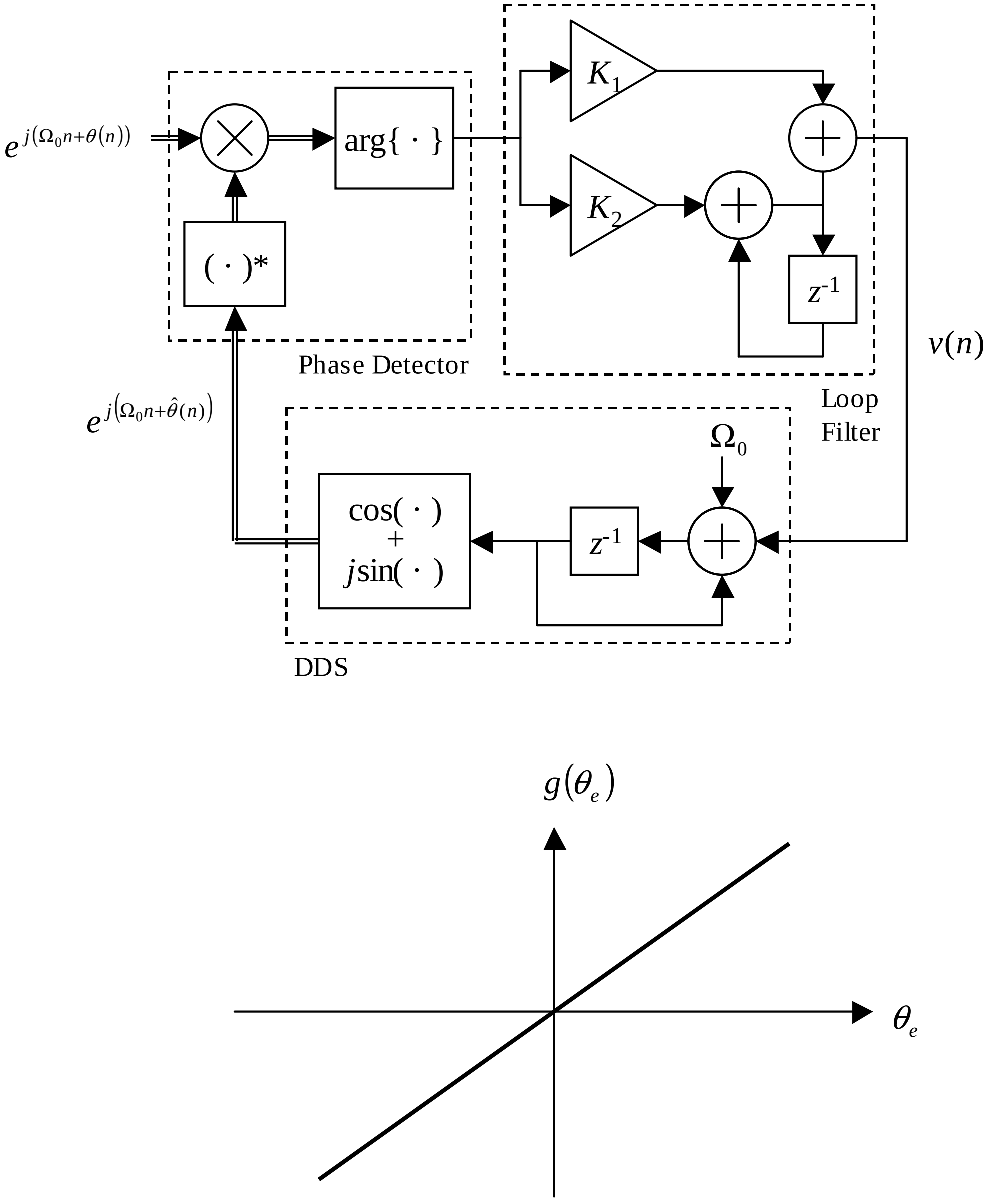


Figure A.11: A second order discrete-time PLL with complex input and output. The phase detector is realized by computing the argument (or phase) of the conjugate product of the loop input and the DDS output. A four-quadrant arc-tangent function can be used to compute the argument. The S-curve corresponding to this phase detector is shown below the block diagram.

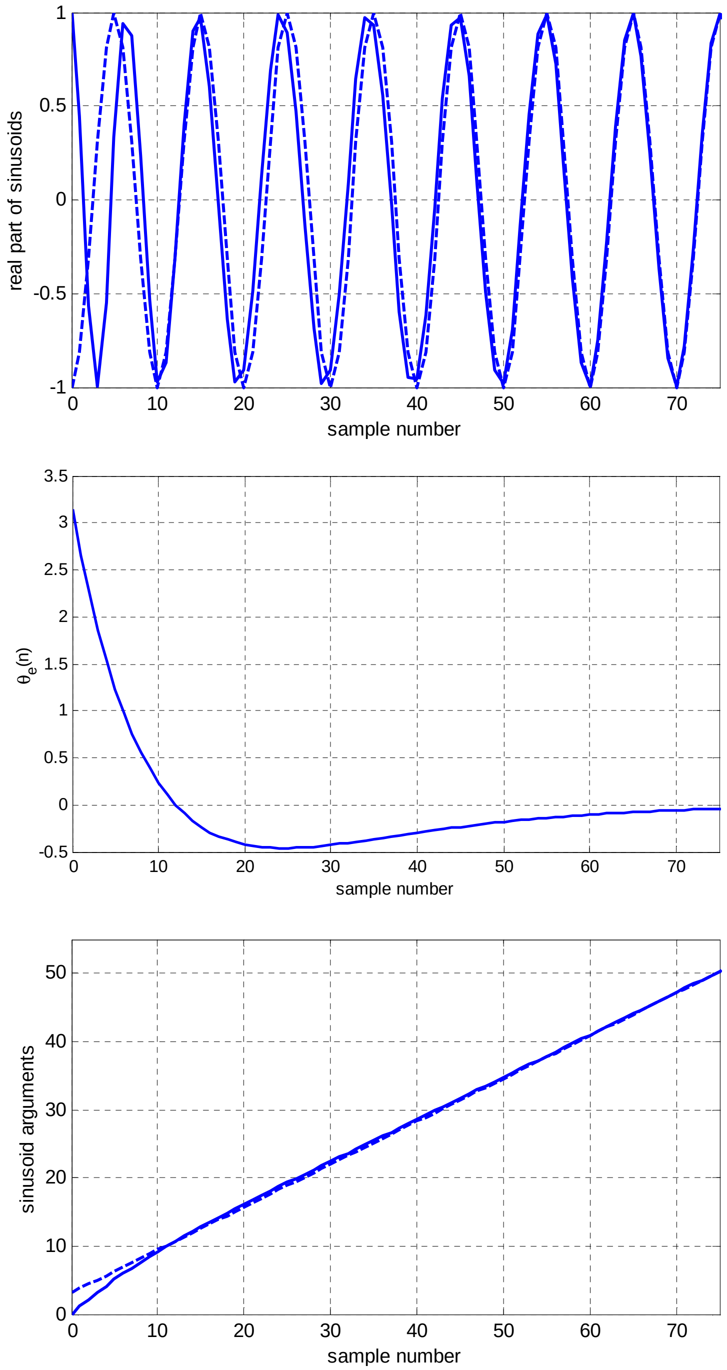


Figure A.12: A plot of loop inputs and outputs for the second-order discrete-time PLL illustrated in Figure A.11. Top: the real part of the loop input (dashed line) and the real part of the DDS output (solid line); Middle: the phase error $\theta_e(n)$; Bottom: the argument of the input complex exponential (dashed line) and the argument of the DDS output complex exponential (solid line).

Figure A.13. The input to the loop is the sinusoid

$$x(n) = A\cos\left(\Omega_0 n + \theta(n)\right). \tag{A.63}$$

The DDS is designed to output two sinusoids in phase quadrature:

$$y_c(n) = \cos\left(\Omega_0 n + \hat{\theta}(n)\right) \tag{A.64}$$

$$y_s(n) = -\sin\left(\Omega_0 n + \hat{\theta}(n)\right). \tag{A.65}$$

The phase detector forms the product

$$x(n)y_s(n) = -A\cos\left(\Omega_0 n + \theta(n)\right)\sin\left(\Omega_0 n + \hat{\theta}(n)\right) \tag{A.66}$$

$$= \frac{A}{2}\sin\left(\theta(n) - \hat{\theta}(n)\right) - \frac{A}{2}\sin\left(2\Omega_0 n + \theta(n) + \hat{\theta}(n)\right). \tag{A.67}$$

Since the loop filter is a low-pass filter, only the low-pass component of the product $x(n)y_s(n)$ needs to be tracked. Thus, the S-curve is

$$g\left(\theta_e\right) = g\left(\theta(n) - \hat{\theta}(n)\right) \tag{A.68}$$

$$= \frac{A}{2}\sin\left(\theta(n) - \hat{\theta}(n)\right) \tag{A.69}$$

and is illustrated in the lower part of Figure A.13. First note that the S-curve is given by the sine of the phase error. Since $\sin(X) \approx X$ for small X, the phase detector is approximately linear for small values of the phase error:

$$g\left(\theta_e\right) \approx \frac{A}{2}\theta_e \qquad \text{for small } \theta_e. \tag{A.70}$$

From this we see that the phase detector gain K_p is $\frac{A}{2}$ which depends on the amplitude of the sinusoid at the PLL input. This illustrates an important caveat in PLL design: Care must be taken to insure that the input signals levels are controlled. Otherwise, the PLL may not performed as designed.

The loop responses are illustrated in Figure A.14 for an input with

$$\Omega_0 = \frac{2\pi}{10} \tag{A.71}$$

$$\theta(n) = \pi u(n) \tag{A.72}$$

The loop input for $A = 1$ is illustrated by the dashed line in the top plot of Figure A.14. As in the previous example, the loop filter constants were chosen to produce a critically damped loop with $B_N T = 0.05$. Using (A.57) produces $K_1 = 0.2958$ and $K_2 = 0.0118$. The loop output (which is the inphase component of the DDS output) is illustrated by the solid line in the top plot of Figure A.14. Note that in the beginning, the two sinusoids are π radians out of phase. As time progresses, DDS output becomes phase-aligned with the PLL input. The phase error $\theta_e(n)$ is plotted in the middle plot of Figure A.14. In this case, the phase error can be thought of as a slowly varying DC component plus a constant amplitude sinusoid. The sinusoid, whose frequency is $2 \times \Omega_0$, is a result of the double frequency component that results from the multiplication performed in the phase detector as predicted by Equation (A.67). This double frequency term is eliminated by the low-pass filter action of the loop filter so that the loop responds only to the average value of the phase error[4]. The average phase error starts small and increases from samples 0 to 20. From samples 20 to 40, the average phase error decreases, passing through zeros at sample 27. It increases again from samples 40 to 60 thereafter settling to an average value of zero. The shape of the average phase error is different from that in the previous example (see the middle plot of Figure A.12) even though the loop input is the same. This difference is due to the non-linear nature of the phase detector. Since the S-curve is non-linear, the loop response to large phase errors is also non-linear. The non-linear characteristics of the response are not predicted by the linear PLL analysis.

The bottom plot of Figure A.14 illustrates the arugments of the input sinusoid (dashed line) an the DDS output (solod line). As before, the dashed line is a plot of $\Omega_0 n + \pi u(n)$, which is the equation of a straight line with slope Ω_n and intercept π. The solid line is the argument of the quadrature DDS output which *is* the DDS input. The solid line tracks the behavior of the average phase error. It starts at 0 with a slope slightly less than Ω_0. The slope then becomes greater than Ω_0 as the PLL output tries to catch up with the PLL input. Note that the two lines cross at sample 27, precisely where the average phase error crossed zero. The solid line and dashed line eventially merge as the DDS output aligns itselft with the PLL input. The fact that the DDS output matches the PLL input is confirmed in all three plots in Figure A.14.

[4] A low-pass filter *could* be inserted after the mixer to eliminate the double frequency term. This filter would then be considered part of the phase detector, but the phase detector would not be memoryless (i.e. it has its own poles and zeros). The poles and zeros of this low-pass filter increase the order of the system. While higher order systems are necessarily bad, the design procedure is much more involved and loop stability harder to achieve. When the inclusion of such a filter is unavoidable, designers usually force the loop bandwidth to be much smaller than the bandwidth of the low-pass filter. This forces the transients due to the loop filter to dominate the transients of the low-pass filter so that the poles and zeros of the low-pass filter can be ignored.

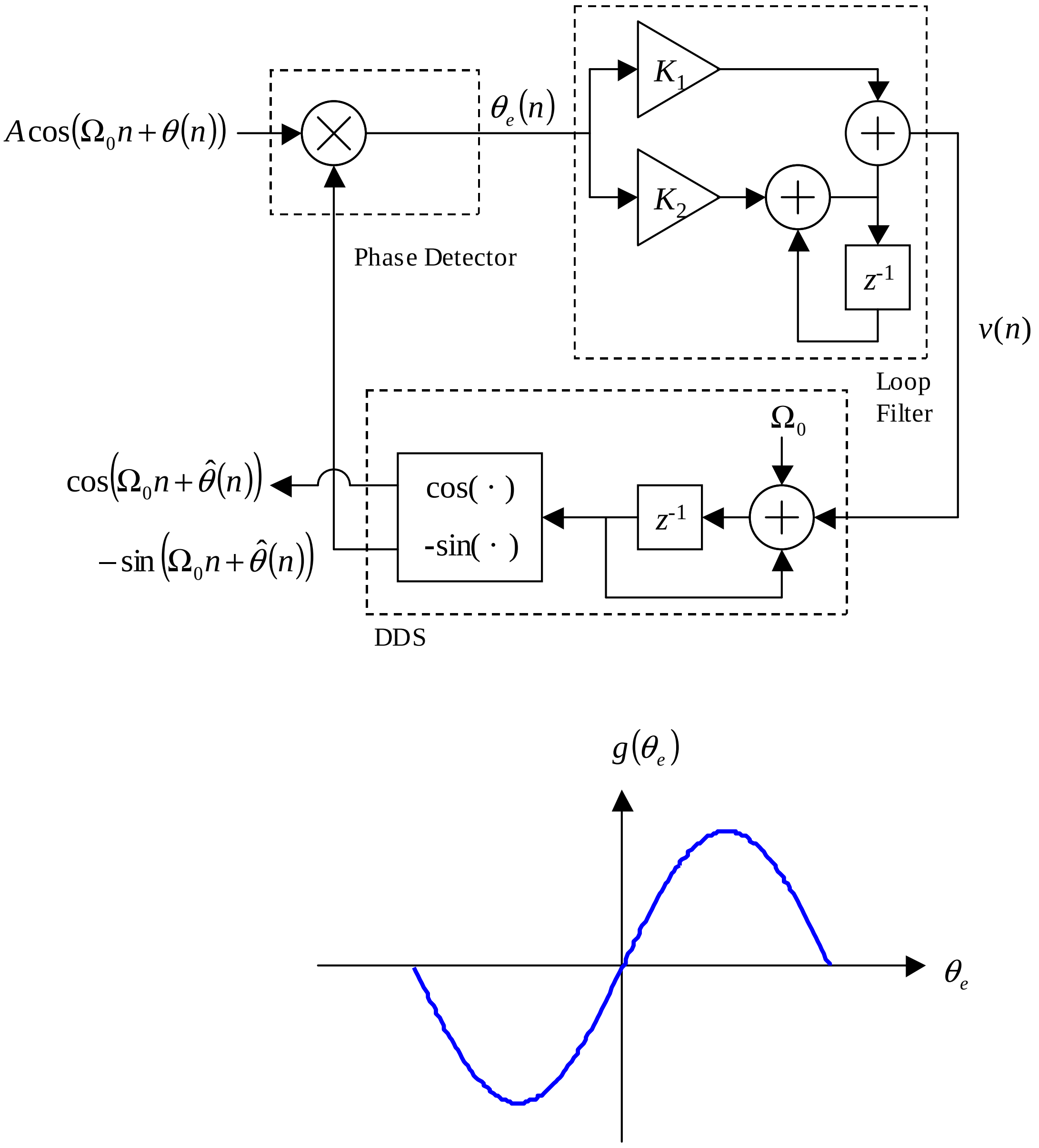


Figure A.13: A second order discrete-time PLL with complex input and output. The phase detector is realized by forming the product between the input sinusoid and the quadrature DDS output. The S-curve corresponding to this phase detector is shown below the block diagram.

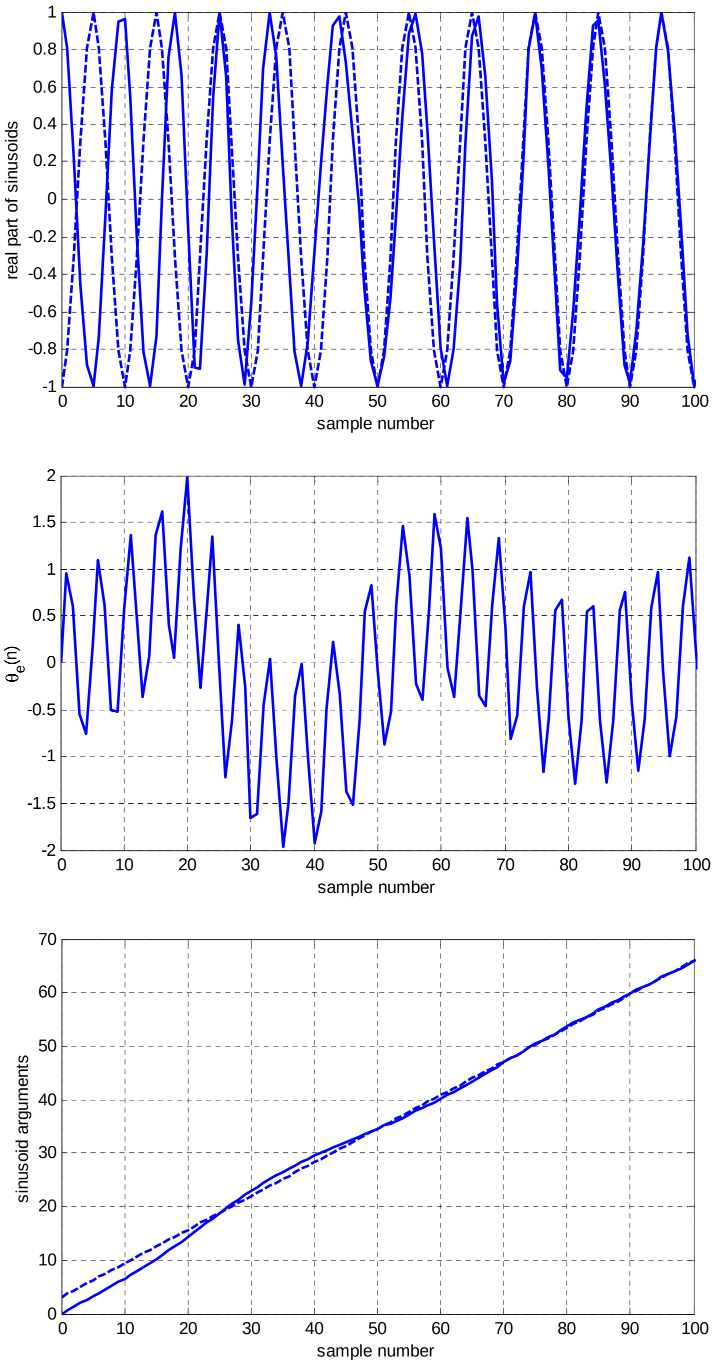


Figure A.14: A plot of loop inputs and outputs for the second-order discrete-time PLL illustrated in Figure A.13. Top: the loop input (dashed line) and the DDS output (solid line); Middle: the phase error $\theta_e(n)$; Bottom: the argument of the input sinusoid (dashed line) and the argument of the quadrature DDS output (solid line).

A.4 Notes and References

The classic text on phase locked loops is Garnder’s [1]. The basics of continuous-time phase locked loops are also covered in reasonably well written texts by Best [2] and Stephens [3]. Waggener [4] provides a nice applications oriented explanation of PLLs in digital communication systems. Discrete-time PLLs and their relationship to continuous-time PLLs are discussed in the seminal paper by Lindsey [**?**]. The conversion from continuous-time PLLs to discrete-time PLLs is described in [3].

Bibliography

[1] F. Gardner, *Phaselock Techniques,* John Wiley & Sons, New York, 1979.

[2] R. Best, *Phase-Locked Loops: Design, Simulation, and Applications,* McGraw-Hill Professional, New York, 1999.

[3] D. Stephens, *Phase-Locked Loops for Wireless Communications,* Kluwer Academic Publishers, Norwell, MA, 2002.

[4] W. Waggener, *Pulse Code Moudlation Systems Design,* Artech House, Boston, 1999.

A.5 Exercises

A.1 Consider a linear continuous-time PLL with loop filter $F(s) = k$.

(a) Determine the time-domain expression for the output $\hat{\theta}(t)$ for a step input; i.e., $\theta(t) = \Delta\theta u(t)$.

(b) Determine the time-domain expression for the output $\hat{\theta}(t)$ for a ramp input; i.e., $\theta(t) = \Delta\omega t u(t)$.

A.2 Consider a linear continuous-time PLL with loop filter $F(s) = \dfrac{k}{s+k}$.

(a) Determine the time-domain expression for the output $\hat{\theta}(t)$ for a step input; i.e., $\theta(t) = \Delta\theta u(t)$. Note there are three cases to consider: $\zeta < 1$, $\zeta = 1$, and $\zeta > 1$.

(b) Determine the time-domain expression for the output $\hat{\theta}(t)$ for a ramp input; i.e., $\theta(t) = \Delta\omega t u(t)$. Note there are three cases to consider: $\zeta < 1$, $\zeta = 1$, and $\zeta > 1$.

A.3 Consider a linear continuous-time PLL with loop filter $F(s) = k_1 + \dfrac{k_2}{s}$.

(a) Determine the time-domain expression for the output $\hat{\theta}(t)$ for a step input; i.e., $\theta(t) = \Delta\theta u(t)$. Note there are three cases to consider: $\zeta < 1$, $\zeta = 1$, and $\zeta > 1$.

(b) Determine the time-domain expression for the output $\hat{\theta}(t)$ for a ramp input; i.e., $\theta(t) = \Delta\omega t u(t)$. Note there are three cases to consider: $\zeta < 1$, $\zeta = 1$, and $\zeta > 1$.

A.4 Consider a linear continuous-time PLL with loop filter $F(s) = \dfrac{k_1 + s}{k_2 + s}$.

(a) Determine the time-domain expression for the output $\hat{\theta}(t)$ for a step input; i.e., $\theta(t) = \Delta\theta u(t)$. Note there are three cases to consider: $\zeta < 1$, $\zeta = 1$, and $\zeta > 1$.

(b) Determine the time-domain expression for the output $\hat{\theta}(t)$ for a ramp input; i.e., $\theta(t) = \Delta\omega t u(t)$. Note there are three cases to consider: $\zeta < 1$, $\zeta = 1$, and $\zeta > 1$.

A.5 Determine the 3-dB bandwidth of a continuous-time PLL with the following loop filters

(a) $F(s) = k$.

(b) $F(s) = \dfrac{s}{s+k}$.

(c) $F(s) = k_1 + \dfrac{k_2}{s}$.

(d) $F(s) = \dfrac{k_1 + s}{k_2 + s}$.

In each case, compare the 3-dB bandwidth with the noise equivalent bandwidth.

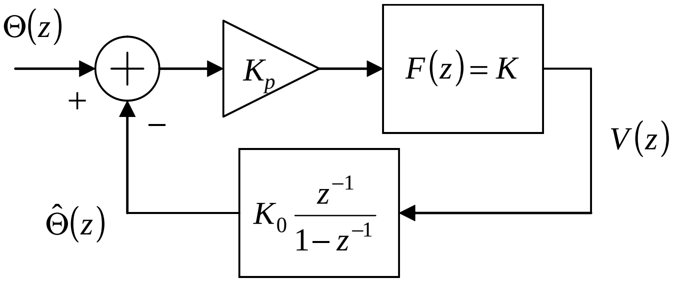
A.6 Consider a linear continuous-time PLL with a proportional-plus-integrator loop filter $F(s) = k_1 + \dfrac{k_2}{s}$. Why is the proportional component necessary? (Hint: Determine the transfer function for this loop in terms of both k_1 and k_2 and set $k_1 = 0$.)

A.7 Consider a continuous-time to discrete-time transformation of a first order continuous-time linear PLL with loop filter $F(s) = k$.

(a) Derive the transfer function of the continuous-time loop $H_a(s)$.

(b) Use Tustin's equation to convert the continuous-time transfer function $H_a(s)$ to a discrete-time transfer function $H_a\left(\frac{2}{T}\frac{1-z^{-1}}{1+z^{-1}}\right)$.

(c) Derive the transfer function $H_d(z)$ for the linearized discrete-time PLL shown below.



(d) Equate the denominators of $H_a\left(\frac{2}{T}\frac{1-z^{-1}}{1+z^{-1}}\right)$ and $H_d(z)$ and solve for the loop filter constant K.

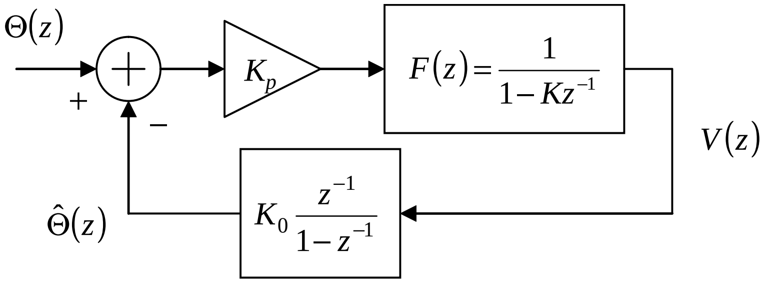
(e) Express the equation for K in terms of the equivalent noise bandwidth of the linearized continuous-time PLL.

A.8 Consider a continuous-time to discrete-time transformation of a second order continuous-time linear PLL with loop filter $F(s) = \dfrac{k}{s+k}$.

(a) Derive the transfer function of the continuous-time loop $H_a(s)$. Express $H_a(s)$ in the standard form using $s^2 + 2\zeta\omega_n s + \omega_n^2$ for the denominator.

(b) Use Tustin's equation to convert the continuous-time transfer function $H_a(s)$ to a discrete-time transfer function $H_a\left(\frac{2}{T}\frac{1-z^{-1}}{1+z^{-1}}\right)$.

(c) Derive the transfer function $H_d(z)$ for the linearized discrete-time PLL shown below.



(d) Equate the denominators of $H_a\left(\frac{2}{T}\frac{1-z^{-1}}{1+z^{-1}}\right)$ and $H_d(z)$ and solve for the loop filter constant K.

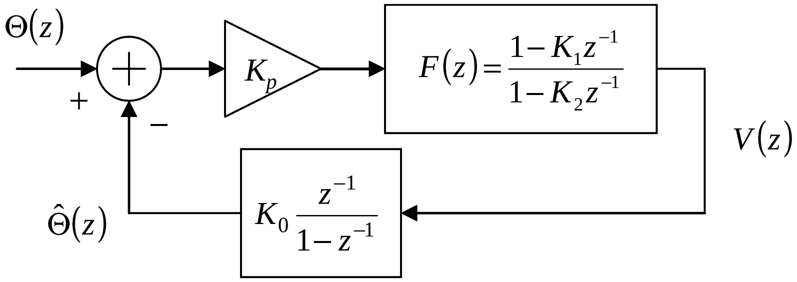
(e) Express the equation for K in terms of the equivalent noise bandwidth and damping factor ζ of the linearized continuous-time PLL.

A.9 Consider a continuous-time to discrete-time transformation of a second order continuous-time linear PLL with loop filter $F(s) = \dfrac{s+k_1}{s+k_2}$.

(a) Derive the transfer function of the continuous-time loop $H_a(s)$. Express $H_a(s)$ in the standard form using $s^2 + 2\zeta\omega_n s + \omega_n^2$ for the denominator.

(b) Use Tustin's equation to convert the continuous-time transfer function $H_a(s)$ to a discrete-time transfer function $H_a\left(\frac{2}{T}\frac{1-z^{-1}}{1+z^{-1}}\right)$.

(c) Derive the transfer function $H_d(z)$ for the linearized discrete-time PLL shown below.


$\Theta(z)$
$+$
$-$
K_p
$F(z)=\frac{1-K_1z^{-1}}{1-K_2z^{-1}}$
$V(z)$
$\hat{\Theta}(z)$
$K_0\frac{z^{-1}}{1-z^{-1}}$

(d) Equate the denominators of $H_a\left(\frac{2}{T}\frac{1-z^{-1}}{1+z^{-1}}\right)$ and $H_d(z)$ and solve for the loop filter constants K_1 and K_2.

(e) Express the equation for K_1 and K_2 in terms of the equivalent noise bandwidth and damping factor ζ of the linearized continuous-time PLL.