A subset of this appears in the Proceedings of the 2002 ACC.

Phase-Locked Loops: A Control Centric Tutorial

Daniel Abramovitch
Agilent Laboratories
Communications and Optics Research Lab
3500 Deer Creek Road, M/S: 25U-9
Palo Alto, CA 94304
Phone: (650) 485-3806
FAX: (650) 485-4080
E-mail: danny@labs.agilent.com

June 24, 2003

Abstract

This paper will present a tutorial on phase-locked loops from a control systems perspective. It will start with an introduction of the loop as a feedback control problem, with both the similarities and differences to traditional control problems. Chief among the differences is the necessary inclusion of two nonlinearities in the loop that are not parasitic, but essential to the loop's operation. Analysis methods, both linear and nonlinear will be discussed. Then digital loops will be discussed, followed by loop components and a cursory look at noise. Finally, the paper will end with a discussion of different applications of PLLs and their relatives.

1 Introduction

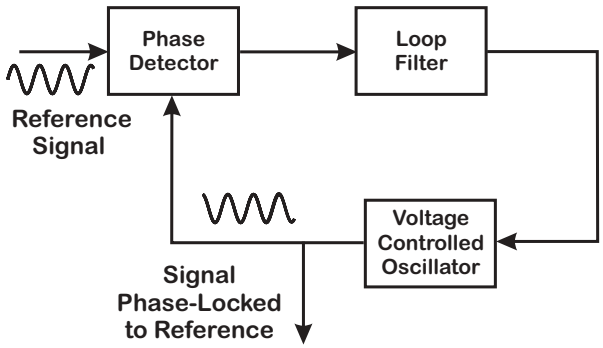

Figure 1: A general PLL block diagram.

Phase-locked loops (PLLs) have been around for many years[1, 2]. Gardner's short history links the earliest widespread use of PLLs to the horizontal and vertical sweeps used in television, where a continuous clocking signal had to be synchronized with a periodic synch pulse [3]. In many respects, the field is mature, with widespread application to almost every type of communication and storage device and a large number of books on the subject [3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]. By the same token, PLLs and their relatives are included in so many bleeding edge applications that their designs are anything but stagnant.

An incomplete list of specific tasks accomplished by PLLs include carrier recovery, clock recovery, tracking filters, frequency and phase demodulation, phase modulation, frequency synthesis, and clock synchronization. PLLs find themselves into a huge set of applications, from radio and television, to virtually every type of communications (wireless, telecom, datacom), to virtually all types of storage device, to noise cancellers, and the like. With the widespread use by the public of such devices, one can claim that PLLs are the most ubiquitous form feedback system built by engineers.

The most basic block diagram of a PLL is shown in Figure 1. This diagram shows the components that every PLL must have, namely:

- A phase detector (PD). This is a nonlinear device whose output contains the phase difference between the two oscillating input signals.
- A voltage controlled oscillator (VCO). This is another nonlinear device which produces an oscillation whose frequency is controlled by a lower frequency input voltage.
- A loop filter (LF). While this can be omitted, resulting in what is known as a first order PLL, it is always conceptually there since PLLs depend on some sort of low pass filtering in order to function properly.

- A feedback interconnection. Namely the phase detector takes as its input the reference signal and the output of the VCO. The output of the phase detector, the phase error, is used as the control voltage for the VCO. The phase error may or may not be filtered.

PLLs have several unique characteristics when viewed from a control systems perspective. First of all, their correct operation depends on the fact that they are nonlinear. The loop does not exist without the presence of two nonlinear devices, namely the phase-detector and VCO. These devices translate the problem from signal response to phase response and back again. Accompanying this is a time scale shift, as PLLs typically operate on signals whose center frequency is much higher than the loop bandwidth. Secondly, PLLs are almost always low order. Not counting various high frequency filters and parasitic poles, most PLLs in the literature are first or second order. There are a few applications where third or fourth order loops are used, but these are considered fairly risky and sophisticated devices. Finally, with the exception of PLL controlled motors, the PLL designer is responsible for designing/specifying all the components of the feedback loop. Thus, complete feedback loop design replaces control law design, and the designer's job is governed only by the required characteristics of the input reference signal, the required output signal, and technology limitations of the circuits themselves. In the case of PLL control of motors, the motor and optical coupler takes the place of the VCO, leaving all other parts of the PLL to the designer's discretion.

With that, one would expect that the study of PLLs would be strongly steeped in control theory and that control theorists would have the highest expertise in PLLs. In fact, the control theory used in most PLL texts is straight linear system design with a small amount of nonlinear heuristics thrown in [3, 4, 5, 6, 8, 9]. The stability analysis and design of the loops tends to be done by a combination of linear analysis, rule of thumb, and simulation[1, 2, 3, 4, 5]. The experts in PLLs tend to be electrical engineers with hardware design backgrounds. The general theory of PLLs and ideas on how to make them even more useful seems to cross into the controls literature only rarely [14, 15, 16, 17].

This tutorial will take a control engineer's view of PLLs. The idea is to map out what is in the common literature on the devices against the background of control design to see how the problem breaks down and what tools of modern control theory can be applied to these devices.

1.1 PLL Basics

The basic idea of a phase-locked loop is that if one injects a sinusoidal signal into the reference input, the internal

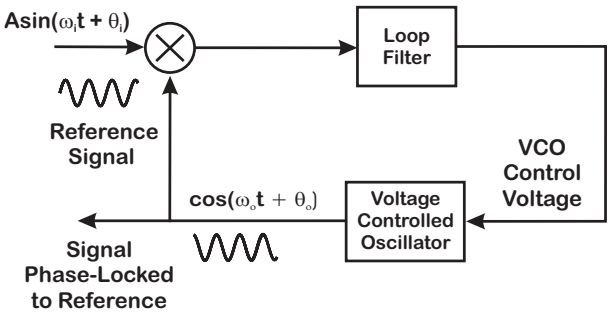

Figure 2: A classic mixing phase-locked loop.

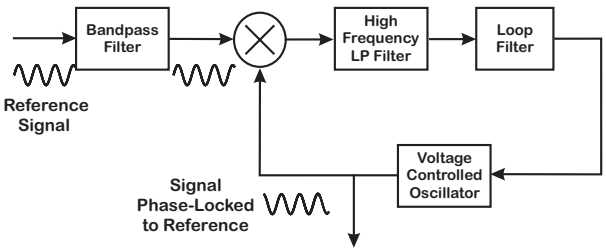

Figure 3: A practical version of the classic mixing phase-locked loop: note the addition of a bandpass filter preceding the loop to limit input noise and a high frequency low pass filter within the loop to attenuate the 2X frequency component with minimal impact on the loop dynamics.

oscillator in the loop will lock to the reference sinusoid in such a way that the frequency and phase differences between the reference sinusoid and the internal sinusoid will be driven to some constant value or 0 (depending on the system type). The internal sinusoid then represents a filtered or smoothed version of the reference sinusoid. For digital signals, Walsh functions replace sinusoids.

Typical block diagrams of PLLs in the literature resemble Figure 2, however practical loops often more closely resemble Figure 3, in which a high frequency low pass filter is used to attenuate the double frequency term and a bandpass filter is used to limit the bandwidth of input

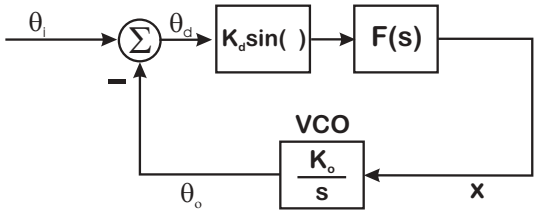

Figure 4: Conceptual block diagram of PLL with sine detector. This is a transition stage in the analysis of the classical mixing loop. This model represents the effect of the multiplying detector once the high frequency component has been attenuated.

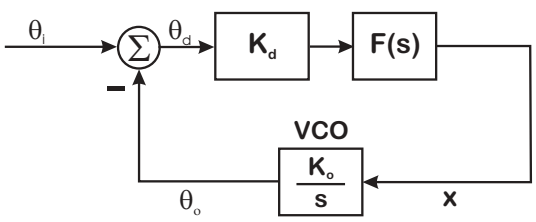

Figure 5: Conceptual block diagram of linear PLL. This is derived from the sine detector loop by assuming that the phase error is small and thus $\sin(\theta_d) \approx \theta_d$. This is the model with which most analyses of phase-locked loops are done.

signals to the loop. A general sinusoidal signal at the reference input of a PLL as shown in Figure 3 can be written as:

$$v_i = R_1(t) = A\sin(\omega_i t + \theta_i). \quad (1)$$

Without loss of generality, we can assume that the output signal from the Voltage Controlled Oscillator (VCO) into the mixer is given by

$$v_o = VCO_{out}(t) = \cos(\omega_o t + \theta_o). \quad (2)$$

The output of the mixer in Figure 3 is then given by

$$v_d = Mixer_{out}(t) = AK_m \sin(\omega_i t + \theta_i)\cos(\omega_o t + \theta_o), \quad (3)$$

where K_m is the gain of the mixer.

Typically, analysis of such a PLL is done by taking several simplifying steps. Using the familiar trigonometric identity in terms of the PLL:

$$2\sin(\omega_i t + \theta_i)\cos(\omega_o t + \theta_o) = \quad (4)$$
$$\sin((\omega_i + \omega_o)t + \theta_i + \theta_o) + \sin((\omega_i - \omega_o)t + \theta_i - \theta_o)$$

and then making two fundamental assumptions leads to the commonly used model of the analog PLL. Let $\theta_d = \theta_i - \theta_o$. Then these assumptions are:

1. The first term in (4) is attenuated by the high frequency low pass filter in Figure 3 and by the low pass nature of the PLL itself.
2. $\omega_i \approx \omega_o$, so that the difference can be incorporated into θ_d. This means that the VCO can be modeled as an integrator.

Making these assumptions leads to the PLL model shown in Figure 4.

The problem is that this is still a nonlinear system, and as such is in general difficult to analyze. The typical methods of analysis include:

1) Linearization: For θ_d small and slowly varying

$$\sin\theta_d \approx \theta_d,\ \cos\theta_d \approx 1,\ \text{and}\ \dot{\theta_d}^2 \approx 0.$$

While this is useful for studying loops that are near lock, it does not help for analyzing the loop when θ_d is large.

2) Phase plane portraits [3, 5]. This method is a classical graphical method of analyzing the behavior of low order nonlinear systems about a singular point. The disadvantage to this is that phase plane portraits can only completely describe first and second order systems.

3) Simulation. Note that explicit simulation of the entire PLL is relatively rare. Because the problem is stiff, it is more typical to simulate the response of the components (phase detector, filter, VCO) in signal space and then simulate the entire loop only in phase space.

The linearized model is shown in Figure 5. This is what is used for most analysis and measurements of PLLs. As will be seen in Section 6, changing the phase detector and VCO can result in a system for which this model is very accurate. It is possible to learn quite a bit about the phase behavior of the PLL from linear analysis. However, this model has some very important omissions that come into play when simulating or constructing the classical PLL:

1) The texts typically omit the input bandpass filter shown in Figure 3. While this is not in the loop itself and the actual input frequency is often not known or is variable, it is most often the case that the designer has some idea of the range of the signal. In this case, an input bandpass filter can considerably reduce the broadband noise entering the system.

2) The texts typically omit the high frequency low pass filter shown in Figure 3. This is important because this filter is highly useful in attenuating the effects of the $2\omega_o t$ signal. The loop filter itself is optimized for the stability and performance of the baseband (phase). The prevalence of the linear phase model often leads designers and simulation tool builders to forget this important component. However, experienced PLL circuit designers include this feature.

3) As seen in (3), the amplitude of the phase error is dependent upon A, the input signal amplitude. The linearized model has a loop gain that is dependent upon the loop components. Thus, in practical loop design, the input amplitude must either be regulated or its affects on the loop must be anticipated.

4) The equations of a PLL are stiff. That is, the loop has a component at baseband and one at $2\omega_o t$. Simulations that sample fast enough to characterize the latter are often far too slow (due to the huge number of sample points) to effectively characterize the former.

2 Linear Analysis Methods for Classical PLLs

The PLL model in Figure 5 is a closed-loop feedback system. The complimentary sensitivity transfer function from reference phase input to VCO phase output, $T(s)$, can be obtained as

$$T(s) = \frac{\theta_o(s)}{\theta_i(s)} = \frac{K_d F(s) K_v / s}{1 + K_d F(s) K_v / s} \quad (5)$$

$$= \frac{K_d K_v F(s)}{s + K_d K_v F(s)}. \quad (6)$$

Similarly, the sensitivity transfer function from the reference phase input to the phase error, $S(s)$, is

$$S(s) = \frac{\theta_d(s)}{\theta_i(s)} = \frac{1}{1 + K_d F(s) K_v / s} \quad (7)$$

$$= \frac{s}{s + K_d K_v F(s)}. \quad (8)$$

Among the basic properties of interest in this transfer function are the loop stability, order, and the system type.

The order of the PLL system should be obvious from the denominator of Equation 6. The stability of the system can be determined by a variety of classical methods including root locus, Bode plots, Nyquist plots, and Nichols charts [18].

2.1 The Hold Range

The hold range, $\Delta\omega_H$, is defined as that frequency range at which the PLL is able to statically maintain phase tracking. It is determined by calculating the frequency offset at the reference input that causes the phase error to be beyond the range of linear analysis. For a multiplying or XOR phase detector, this phase error is $\pi/2$. For sequential detectors, it will be larger. Best states that since loops will be permanently out of lock if the frequency offset at the input is greater than the hold range, this quantity is more of an academic matter than a practical one, but it can be calculated for a classical PLL (sinusoidal phase detector) as [8, 19]

$$\Delta\omega_H = K_o K_d F(0). \quad (9)$$

2.2 The Lock Range

The lock range, $\Delta\omega_L$, is defined as that frequency range within which the PLL locks within one single-beat note between the reference frequency and output frequency [8]. The lock range must be calculated from a nonlinear equation, but there are several useful approximations that are made. In particular, if the relative order of numerator and denominator of the PLL are 1, then the loop can be said to behave like a first order loop at higher frequencies, and thus the lock range can be estimated as [19]

$$\Delta\omega_L \approx \pm K_o K_d F(\infty). \quad (10)$$

2.3 The Pull-In and Pull-Out Range

The pull-in range, $\Delta\omega_P$, is defined as the frequency range in which the PLL will always become locked. The pull-out range, $\Delta\omega_{PO}$, is defined as the limit of dynamic stability for the PLL [8]. Unfortunately, there are no simple relationships for these.

2.4 The Steady-State Error

Steady state errors of PLLs are obtained from the linear analysis via use of the Final Value Theorem, i.e.

$$\lim_{t\to\infty} \theta_d(t) = \lim_{s\to 0} s\theta_d(s) \quad (11)$$

$$= \lim_{s\to 0} s\theta_i(s) S(s). \quad (12)$$

As seen from Equation 7, the presence of a VCO makes every PLL at least a Type 1 system, achieving zero steady state error to a phase step at θ_i. For a phase ramp or (equivalently) a frequency step, there must be another integrator in the forward path, and the natural place for this is the loop filter, $F(s)$.

It is worth noting that third order PLLs, which can be Type 3 systems and have zero steady state error to a frequency ramp, are relatively rare. There are two apparent reasons for this. First of all, the applications that require that type of performance are typically only found in deep space communications, where the Doppler shift of the signal produces a frequency ramp. The second potential reason has to do with the stability of the third order loop versus that of the second order loop. It turns out that the parameter values that make the linear model of second and first order PLLs stable also guarantee stability of the nonlinear PLL model shown in Figure 4. However, for third order loops and higher, this is not the case [20, 21].

3 Nonlinear Analysis Methods for Classical PLLs

3.1 Phase Plane

One of the earliest methods of nonlinear system analysis is the graphical method of phase plane design. This has been used in the early days of PLLs [5], before the widespread use of computers for such calculations. The use of phase plane portraits for PLLs is made more practical by the fact that most PLLs are first or second order, which doesn't clash with the order restrictions on phase plane techniques.

3.2 Lyapunov Redesign

Starting with the the sinusoidal phase detector model shown in Figure 4, it has been possible to apply the technique of Lyapunov Redesign [22] to phase-locked loops [20, 21]. More recently, this method has been extended to classical digital PLLs [23].

3.3 Circle/Popov Criteria

The difficulty of applying Lyapunov methods to higher order loops has led to the exploration of nonlinear analysis methods suitable for numerical techniques. In particular, the Circle Criterion [14] and the Popov Criterion [24] have been used to check the stability of higher order PLLs.

4 Digital Signals

Generally speaking, there are a variety of reasons to use digital circuitry to implement PLLs rather than the classical methods above. In this case analog voltage levels are often replaced by digital logic levels. For example, clock signals to drive digital circuitry, computers, and digital communications systems all run better with Walsh functions (rectangular waves) rather than sinusoids. Furthermore, these digital circuits are easier to integrate and verify than their analog counterparts. Finally, as the speed of the logic outstrips the speed requirements of the applications, such implementations become far more reliable than the classical methods.

PLLs that deal with digital signals have one or more of their components replaced by digital circuitry. This results in analysis that is considerably different from what we have seen up to now. However, once one converts the mindset to that of digital signals, the analysis is often linear.

5 Digital PLLs

The definition of most digital feedback loops is fairly straightforward. Digital loops sample the input, converting it to a digital quantity using an ADC, perform the control law calculation using some type of computer, and output the resulting control signal through a DAC. However, the definition of a digital PLL depends quite a bit upon which text one reads. Digital PLLs may consist entirely of analog components with the exception of using one of the digital phase detectors described in Section 6. In other cases, the loop consists of a digital phase detector, a digital filter, and a numerically controlled oscillator. Best's book distinguishes these as the Classical Digital PLL and the All Digital PLL, respectively [8]. Finally, an all software PLL is possible on reference signals that are entirely digitized. The loop components themselves are implemented entirely in computer code.

5.1 Classical Digital PLLs

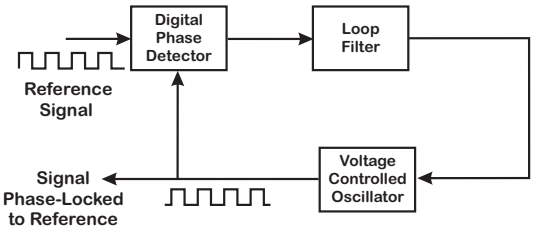

Figure 6: A classical digital phase locked loop.

The classical digital PLL (CDPLL) shown in Figure 6 is somewhat of a misnomer from the controls perspective. It is not a digital, sampled data system as the term digital would imply to control theorists. Instead, it is an analog PLL implemented with a digital phase detector, such as one of those in Section 6. In this case, the output of the digital phase detector is seen as a continuous time voltage and this voltage is fed to an analog loop filter. PLL authors point out that this type of PLL has all the disadvantages of the classical PLL due to its analog components. Still, this loop has advantages in that it can be implemented at very high frequencies (multiple Gigahertz) with fairly reliable logic. Furthermore, these loops can be analyzed using continuous time linear feedback theory. It is for this reason that some authors do not treat these loops as digital at all [6].

5.2 All Digital PLLs

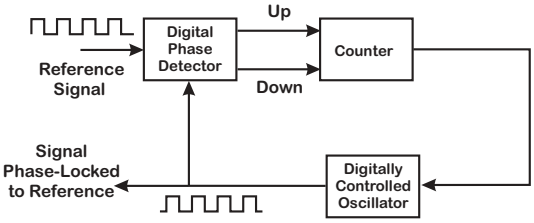

Figure 7: An all digital phase locked loop. The digital phase detector produces pulses that go into the count up or count down inputs of the counter, which acts as the loop filter. The counter then adjusts the frequency of the digitally controlled oscillator (DCO).

The first difference between the all digital PLL (ADPLL) [8] and the classical digital PLL of Section 5.1 is in the use of the digital phase detector (DPD). In the latter case, the DPD was used to generate analog voltages in continuous time. In the ADPLL, the DPD's output is considered a digital quantity, either pulses or multi-bit values. The ADPLL [8] replaces the analog filter with some sort of a digital filter and the VCO with a digitally

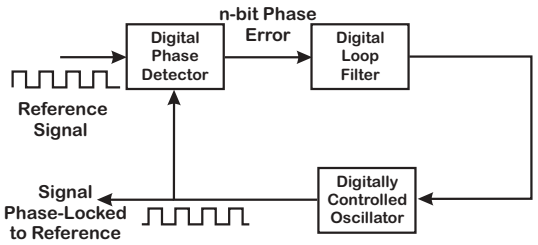

Figure 8: Another all digital phase locked loop. The digital phase detector produces samples of phase error in an n-bit value. This value is fed to a digital filter whose output adjusts the the frequency of the digitally controlled oscillator (DCO).

controlled oscillator (DCO).

The DPD and the loop filter are chosen together at this point. For a DPD that produces pulse streams on its outputs that correspond to either high or low phase error, the filter used is some type of counter. Note that while the pulse streams are essentially continuous time in nature, the counter relies on an input clock. This clock makes the DPD/counter combination a sample data system. This is shown in Figure 7.

For a DPD that produces a sampled stream of multi-bit valued numbers, a digital filter in the classical sense can be used [25]. This case, shown in Figure 8, corresponds most closely to what the controls world considers a digital feedback loop.

5.3 Software PLLs

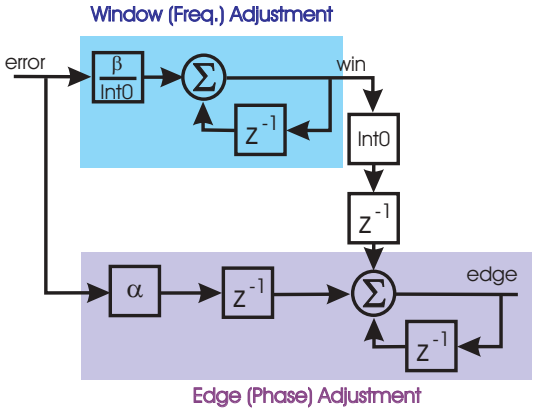

Figure 9: An example of a software PLL that does clock/data recovery. This heuristic loop uses a zero crossing detector on the sampled input. The effective sample rate is derived from the average bit zero crossing rate.

When data can be sampled at a rate substantially faster than the loop center frequency, the entire loop operation can be implemented in software. This has the advantage of flexibility. Any type of PLL can be implemented in software provided the sample rate is high enough. Software loops have a lot in common with simulation. One key difference is that the software loops deal with real data. Software PLLs may operate on the data in real time, but can also be used in the post processing of measured data. One cautionary note is that certain operations which are highly effective in hardware, such as limiters which have a lot of high frequency content, create real sampling issues for software loops.

6 Phase Detectors

The analysis methods of Section 2 above were applied to the classic mixing loop. This has the property that once an ideal multiplication is done, the analysis of the baseband signal can be more rigorous. This section will explore phase detectors constructed from digital logic for which the initial reduction to baseband relies on arguments of pulse-width modulation and averaging [3, 6, 8]. While these phase detectors have worse noise performance than the classic mixing detectors, they often have better pull in range and are much more manufacturable, especially for high speed applications. Furthermore, most of these phase detectors have advantage that their low frequency response is actually linear over some range rather than sinusoidal. The exception to this group is the Alexander or Bang-Bang phase detector [26], which as it's name implies produces a response similar to that of a relay.

Analysis of digital phase detectors requires a different view from that of classical mixing detectors. First of all, while the exact behavior of these digital phase detectors is necessarily nonlinear, the low frequency behavior is often linear. Secondly, the circuitry of the phase detectors are constructed more to deal with specific circuit conditions than to make analysis simpler. Finally, no one type of phase detector is best for all situations. Thus, vastly different circuit designs are chosen to implement largely the same functionality for different applications.

6.1 Mixing

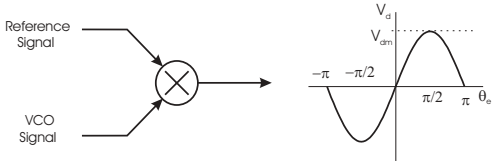

Figure 10: Classical mixing phase detector

The mixing (multiplying) phase detector shown in Figure 10 and discussed in Sections 1.1–3, has superior noise performance to all the other detectors discussed here [6, 9], due to the fact that it operates on the entire amplitude of the input and VCO signals, rather than

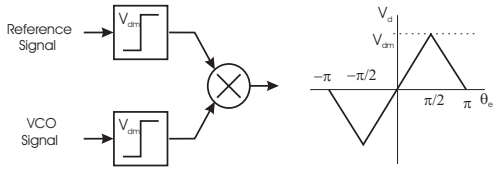

Figure 11: Over driven mixing phase detector

quantizing them to 1 bit. Balanced mixers are best suited for PLL applications in the microwave frequency range as well as in low noise frequency synthesizers. However, as mentioned earlier, this results in a loop whose gain is dependent upon the signal amplitude. Furthermore, non-idealities in the circuit implementation of the mixer result in responses that are far from linear. When noise is not an issue, it is advantageous to move to a detector that has immunity to these effects.

6.2 XOR

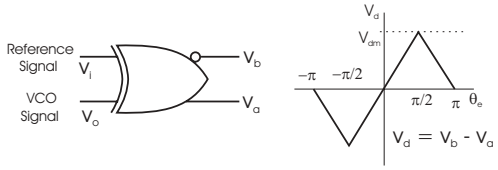

Figure 12: Phase detection using an XOR gate. Note that this accomplishes the same thing as an over driven mixer, but with digital circuitry.

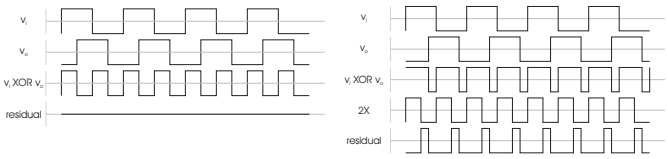

Figure 13: Phase detection using a XOR gate. On the left, a phase shift between reference and VCO output of $\pi/2$ produces an output of the phase detector whose baseband component is 0. On the right a relative phase shift of $\pi/4$ results in an output of the phase detector whose baseband component is $v_d/2$. The output is broken up here into a 2X frequency signal and a residual. The 2X signal averages to 0, while the residual averages to the baseband phase error.

For a variety of reasons, it may be desirable to have a loop which does not produce a sinusoidal clock but instead a square wave clock. If one over-drives the mixer circuit, that is if one uses signals so large that the amplifiers saturate, the output signals stop looking like sinusoids and start looking like Walsh functions (rectangular signals). Such a phase detector is shown in Figure 11. Understanding the output of such a phase detector relies on a combination of averaging analysis and heuristics. However, one of the more interesting features of such a phase detector is that it can be implemented using an Exclusive-OR (XOR) gate as shown in Figure 12. One advantage of such a phase detector is that the loop gain is now independent of input signal amplitude. Furthermore, an XOR phase detector's response can have a larger linear range than a sinusoidal detector (mixer). The disadvantage is that the linearity of the baseband response is affected by the relative duty cycles of the input and VCO signals [6, 8]. The standard analysis done by PLL engineers involves drawing out square waves as shown in Figure 13 and then doing some heuristic "analysis" to convince themselves that the baseband (low frequency) component of the signal behaves with the triangular phase response shown in the right of Figure 12 (for a 50% duty cycle of the input signal).

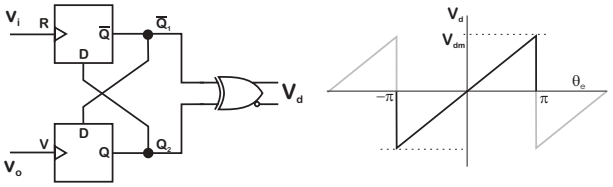

Figure 14: Two state phase detection using gates. The two logic combinations result in the same PD characteristic.

6.3 Two State Phase Detectors

To eliminate the duty cycle dependence of the XOR phase detector, detectors using logic gates can be used. An example of this is found in Wolaver [6] is shown on the left side of Figure 14. The addition of the two flip flops to the XOR gate has several results. First, the phase detector is only sensitive to the rising edges of the input signals, rather than their duty cycles. Secondly, the linear region of the phase detector is expanded to $\pm\pi$ from $\pm\pi/2$. Finally, the phase detector is no longer memoryless. Thus, noise spikes that are large enough to trigger a change of state have a larger effect than they do with the XOR phase detector. The state analysis of this phase detector is a bit involved.

Because this phase detector uses only the leading edge of the input signals, the linear region is increased as mentioned above. The resulting baseband response can be understood from square wave manipulations as described above. The resulting baseband component of the phase detector output now has a sawtooth, rather than triangle wave response, and so this detector is often called a sawtooth detector.

6.4 Phase-Frequency Detector

An extremely popular phase detector is the combination of the tri-state phase-frequency detector (PFD) with a charge pump shown in Figure 15. The charge pump can be viewed as a 3 position switch controlled by the phase-

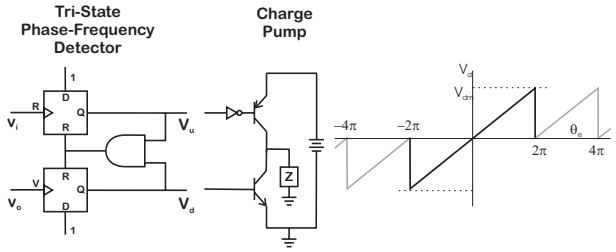

Figure 15: The combination of a tri-state phase-frequency detector and a charge pump. Note that the loop filter is often implemented in the Z block of the charge pump.

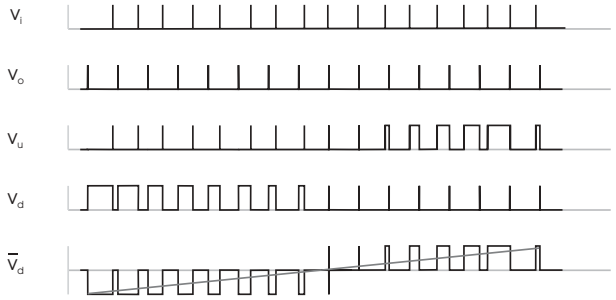

Figure 16: Phase detection using an phase-frequency detector (PFD). As only the leading edges are significant, these are compared to show the phase behavior of the detector.

frequency detector. The action of the charge pump is to alleviate any loading of the phase detector in driving the rest of the circuit. This allows the response to be smoother than without the charge pump. Note that the loop filter is often implemented within the charge pump as shown at the right of Figure 15. Figure 17 shows the use and misuse of the PDF. In the case on the left, a difference in frequencies is detected and the phase is ramped that eventually the frequencies match. However, this same property means that if this detector were used in clock/data recovery (Section 10.3), the missing transitions are misinterpreted as a lower input frequency, resulting in the phase detector ramping the VCO frequency down.

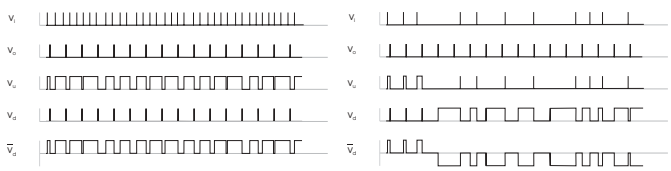

Figure 17: The left diagram shows how the PFD responds to frequency errors. The response rapidly slews the frequency towards the correct value. This same property makes the PFD ineffective for use in clock data recovery (CDR) as the "missing" transitions in the data trick the PFD into slewing the frequency to a lower clock rate, as shown in the right diagram.

6.5 Sample and Hold

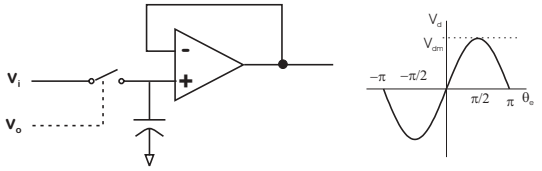

Figure 18: Sample and hold as a phase detector.

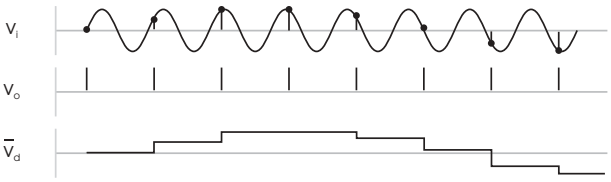

Figure 19: Signals through a sample and hold phase detector.

A sample and hold circuit may be used as a phase detector as shown in Figure 18. The basic idea here can be seen as follows. Assume the input signal, v_i is sinusoidal as shown in the top plot of Figure 19. Furthermore, assume the VCO signal, v_o is used to trigger sampling of v_i as shown in the middle plot of Figure 19. If the frequency of v_o matches or is close to the input frequency, then the value of the sampled signal will depend only on the relative phase of v_i and v_o. Because the sample rate is below the Nyquist frequency, the samples alias down. However, these aliased samples create a phase error signal as seen in the lower plot of Figure 19. Note that the shape of the phase detector characteristic is based on the shape of the input signal, v_i, so that if v_i is sinusoidal, $\bar{v}_d$ is sinusoidal. If v_i is triangular, $\bar{v}_d$ is triangular [6]. Finally, if v_i is a square/rectangular, then $\bar{v}_d$ has a relay characteristic. Note also that there is no high frequency component resulting from this phase detector. This can be seen analytically due to the fact that the zero-order hold has a zero at the sample frequency [27].

6.6 A Linear Clock Phase Detector

Clock recovery from a data stream, known as clock/data recovery (CDR), requires a special type of phase detector. One of the most popular is the so-called Hogge [28] detector, show in Figure 20. An improvement with lower jitter was generated by Shin et. al. [29] and that version is often called the Hogge-Shin detector.

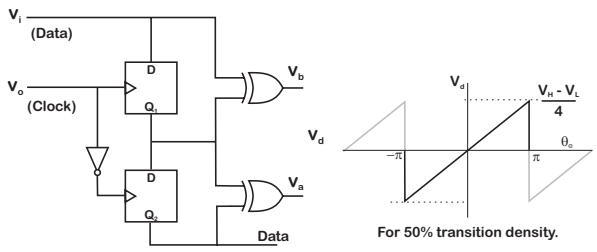

Figure 20: The Hogge phase detector. Used primarily in clock data recovery applications (CDR), the Hogge detector has a linear characteristic. $\bar{V}_b$ is modulated by the signal phase while $\bar{V}_a$ is not. The difference gives a phase error for the data signal.

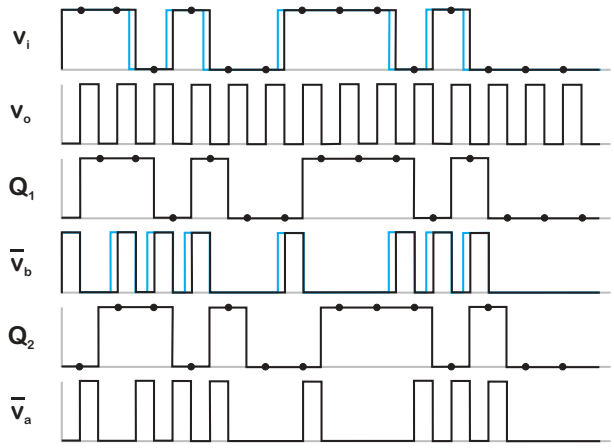

Figure 21: Phase detection using a Hogge phase detector.

6.7 A Bang-Bang Clock Phase Detector

The Bang-Bang phase detector [26] shown in Figure 22 is unique among the detectors presented here in that its baseband behavior is never linear. Instead, the detector acts as a relay over the region from $-\pi$ to π. The behavior can be seen from the timing diagram analysis shown in Figure 23. The behavior can be described as follows: the signals a, b, and c are re-timed versions of the data signal. a and c are one bit period apart, b is sampled at the half period between a and c. Basically, if a and c are the same, then no transition has occurred and the output of the phase detector is tri-stated. If not, then the state depends on b. If b = a, then the clock is early. If b = c, then the clock is late.

While the nonlinear behavior of the detector is a disadvantage, it has advantages for high speed clock data recovery applications over the Hogge detector of Section 6.6 in that it does not need to be calibrated on an individual basis. This improves manufacturability when the circuit technologies are being pushed to their limits.

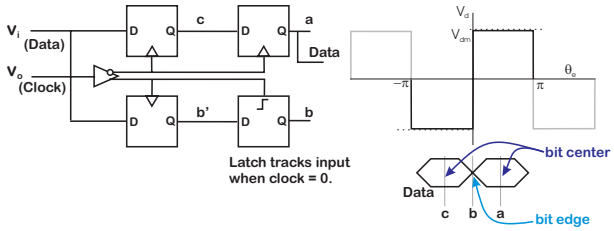

Figure 22: The Alexander (bang-bang) phase detector. The original version made of component flip flops. The version shown here is a circuit well suited to integration which substitutes a latch for the last flip flop, thereby saving one latch. On the right is the phase detector characteristic.

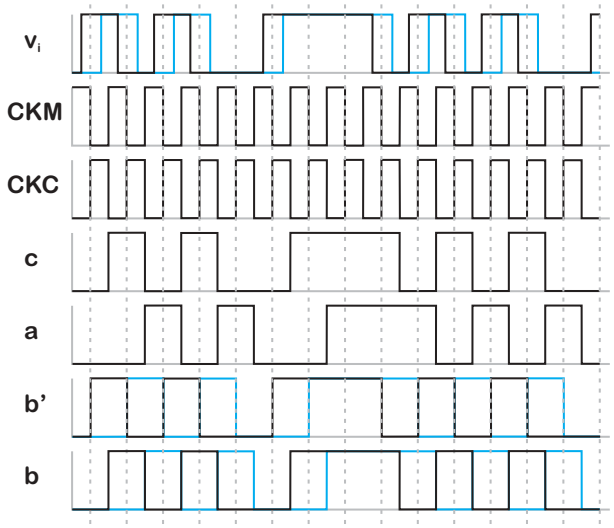

Figure 23: Phase detection using a bang-bang phase detector.

6.8 Conclusions

What should be clear from the above section is that the mere existence of such wide variety of phase detectors indicates that no one phase detector is optimal or even applicable in each situation. Their usefulness depends greatly on the type of PLL they will be used in and on the input signals that they will be encountering.

7 Voltage Controlled Oscillators

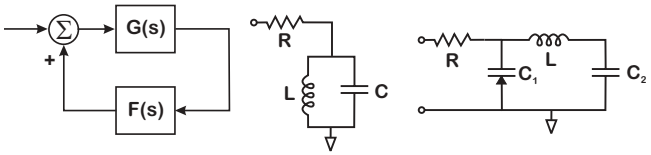

Figure 24: On the left, a block diagram of an oscillator implemented as a positive feedback loop between a voltage to current amplifier through a resonant circuit. On the right, examples of resonant circuits: a LC tank and a π network.

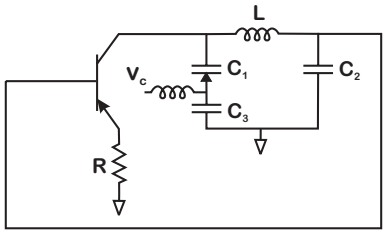

Figure 25: A VCO implemented through a π network. The frequency is adjusted by adjusting the reverse bias on the varactor diode, C_1.

The actual clock generated by a PLL comes from the voltage controlled oscillator (VCO). The VCO is a nonlinear device which generates a periodic oscillation. The frequency of this oscillation can be controlled by modulating some control voltage (hence the name). In a PLL, the control voltage corresponds to some filtered form of the phase error. In response to this, the VCO adjusts its frequency. As the VCO frequency is slewed by the control voltage, the phase is error is driven towards 0. This frequency adjustment to achieve phase lock results in the model of a VCO as an integrator.

VCOs are generally of the form of a ring oscillator, relaxation oscillator or a resonant oscillator. The ring oscillator, common in monolithic topologies takes the form of an odd number of inverters connected in a feedback loop [30]. The relaxation oscillator uses a Schmitt-trigger to generate a stable square wave [6]. The latter puts a resonant circuit in the positive feedback path of a voltage to current amplifier as shown in Figure 24. The amplifier shown for these circuits is a voltage to current amplifier with close to unity gain. The resonant circuit in the positive feedback path has poles close to the $j\omega$ axis. Consider the bandpass filter:

$$F(s) = \frac{2\zeta\omega_0 s}{s^2 + 2\zeta\omega_0 s + \omega_0^2}, \tag{13}$$

and $G(s) = K < 1$. Then

$$VCO(s) = \frac{G(s)}{1 - G(s)F(s)} = K\frac{s^2 + 2\zeta\omega_0 s + \omega_0^2}{s^2 + 2\zeta_1\omega_0 s + \omega_0^2}, \tag{14}$$

where $\zeta_1 = (1 - K)\zeta$. The lowering of the damping ratio is called "Q amplification" ($Q = \frac{1}{2\zeta}$) and moves the poles even closer to the $j\omega$ axis. (In the case of the π network, there is a complex pair of poles and one pole on the negative real axis. The dominant effect, Q amplification, takes place on the complex pair.) The frequency is controlled by altering the capacitance of the resonator, typically by using a varactor diode as a capacitor. A simple circuit diagram for a resonant circuit VCO is shown in Figure 25. Other forms of VCOs, such as crystal oscillators and YIG oscillators essentially run on the same principle, but modify the resonant circuit.

For the all digital and software PLLs, the VCO is replaced by a digitally or numerically controlled oscillator (DCO/NCO) [8]. In this case, the input voltage is replaced by some digital value. The output is a digital oscillating waveform.

8 Loop Filters

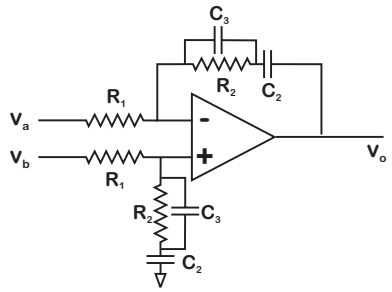

Figure 26: Analog loop filter for differential inputs. For single ended input, the + terminal can be tied directly to ground.

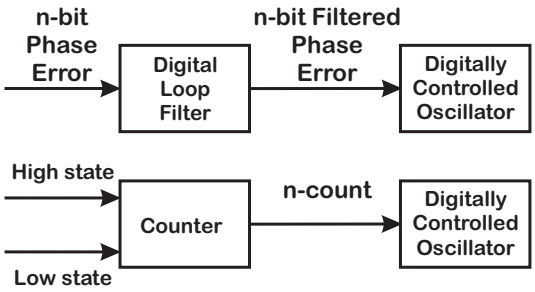

Figure 27: Digital PLL filters depend upon the output of the phase detector and the input of the DCO. The integration function can be accomplished with a counter.

As noted earlier, there is – at least conceptually – always a loop filter. Typical analysis ignores the high frequency low pass filter and other dynamics that do not affect the behavior of the loop at the time constants of the phase.

As the vast majority of PLLs are second order and as the actions of the VCO are modeled as an integrator, loop filters are typically first order. More specifically, since a Type II system will track a phase ramp and this corresponds to tracking a step in frequency, the loop filter almost always contains an integrator. For a double integrator system, the loop filter needs a minimum phase zero to obtain stability. This is true whether the filter is implemented as an analog or digital filter. Higher order loops (which are rarer) can be obtained by adding extra pole/zero pairs to the filter.

The analog circuit (classical or classical digital PLL) shown in Figure 26 shows the general form of a loop filter. This particular filter is fairly general with a transfer

function of:

$$\frac{V_o}{V_a - V_b} = -\frac{sR_2(C_2 + C_3) + 1}{sR_1C_2(sR_2C_3 + 1)}. \quad (15)$$

For a typical second order loop, we let $C_3 = 0$. For single ended input (V_a only), simply tie the positive terminal of the op-amp to ground.

Digital filters used in all digital PLLs very much depend upon the type of phase detector being used and the type of DCO used at the output. There are many different types [8]. Figure 27 shows two. When the phase detector output is a n-bit sampled number, then it is reasonable to construct a digital filter with classic Z-plane techniques. However, as seen in Section 6, many phase detectors simply put out pulses. In this case, a n-bit counter can be used. The operation of this counter can be described as follows. Assuming the DCO has a center frequency that is set for the nominal counter value, N, if up pulses add to and down pulses subtract from N, then the counter output can be seen as an average of the PD pulses:

$$n_{out}(z) = (1 + z^{-1} + z^{-2} + z^{-2} + \ldots)\theta_d(z), \quad (16)$$

which can be interpreted as a digital integrator with a zero at z = 0:

$$\frac{n_{out}(z)}{\theta_d(z)} = \frac{1}{1 - z^{-1}} = \frac{z}{z - 1}. \quad (17)$$

Combining this with the digital integrator of the DCO and the PD gain yields a PLL open loop of:

$$\frac{\theta_{out}(z)}{\theta_d(z)} = \frac{K_dK_vT}{2}\frac{z(z+1)}{(z-1)^2}, \quad (18)$$

which should have poles that stay within the unit circle. A large excess in up or down pulses would saturate the counter in one direction or another, which would lower the effective open loop gain but not destabilize the loop.

9 Noise

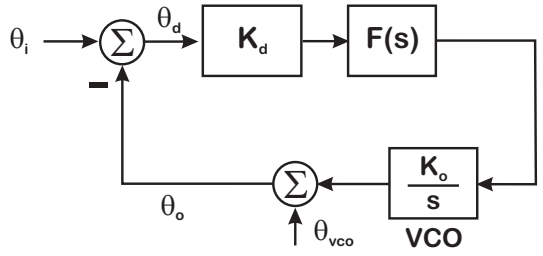

Figure 28: Linear model of input and VCO noise passing through a PLL.

Since a PLL locks to the phase of the input signal, one of the key measures of the performance of the loop is the phase noise (or jitter) in the output, θ_o. Figure 28 shows the linear PLL model with noise at both the reference input, θ_i, and the VCO, θ_{vco}. In particular one may design the loop to minimize θ_o or to have θ_o track θ_i precisely. As with a standard linear feedback loop,

$$\theta_o = T(s)\theta_i + S(s)\theta_{vco} \quad (19)$$

where $T(s)$ and $S(s)$ are obtained from Equations 5 and 7, respectively. As we can assume that the PSDs of θ_i and θ_{vco} are independent, the PSD of the output phase is given by:

$$G_{oo}(j\omega) = \|T(j\omega)\|^2 G_{ii}(j\omega) + \|S(j\omega)\|^2 G_{vv}(j\omega). \quad (20)$$

As with most feedback systems, the loop designer has some control over the effect of θ_i on θ_o through the shaping of the loop, but beyond the loop bandwidth, θ_o is dominated by θ_{vco}.

This discussion has touched on two noise sources, but every component of a PLL is a potential source of phase noise, from the phase detector to the resistors in the filter. A survey of these sources is found in Kroupa [31]. The study of VCO noise is a field unto itself. Perhaps most commonly used model was presented by Leeson [32]. Among the other sources of information on the analysis of noise in PLLs is Wolaver's book [6] and a host of papers [33, 34].

10 Applications

The ubiquity of PLLs is due to their usefulness in so many applications that proliferate through everything from communications systems, to computer clocks, to wireless systems, to consumer electronics. A set of representative examples is listed here.

10.1 Carrier Recovery

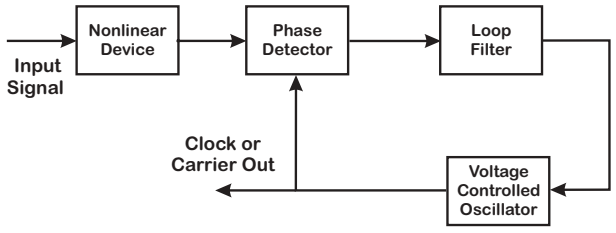

Figure 29: General block diagram of frequency recovery from a modulated signal.

A common process in communications systems is to modulate a signal onto a carrier frequency. In order to demodulate the signal from the carrier, often called RF in a reference to the early days of radio, one must first recover the carrier signal from the composite signal. If

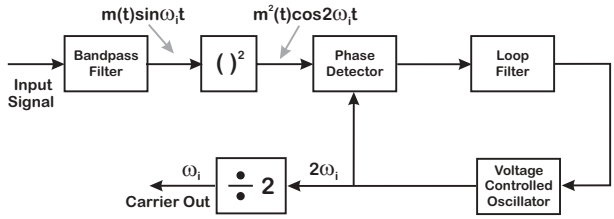

Figure 30: Squaring loop to recover carrier from a modulated signal.

the signal spectrum contains a strong component at the carrier frequency, then this is easily accomplished with a PLL. However, quite often there modulation removes the carrier from the signal spectrum. The restoration of a carrier in this case is generally accomplished by preceding the PLL with some sort of nonlinear element. A specific example of one of these is carrier recovery when the signal has the form:

$$r(t) = m(t) \sin \omega_i t \tag{21}$$

and $m(t)$ is ± 1. Known as binary phase shift keying (BPSK), this simple communication method results in the spectrum of r having no component at ω_i (for equally probable $+1$ and -1 bits). In this case, the squaring loop shown in Figure 30 is able to lock to $2\omega_i$. The divide-by-2 circuit recovers the carrier.

10.2 Costas Loop

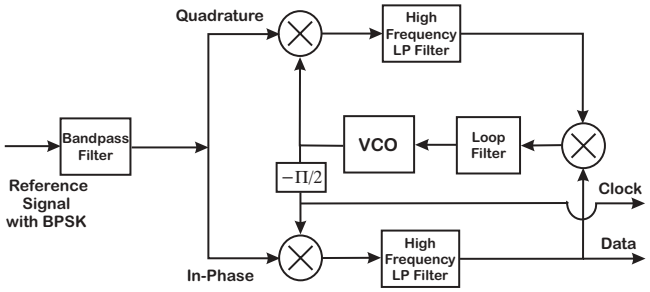

Figure 31: A Mixing Costas Loop

In the example of BPSK in Section 10.1, it was shown how a squaring loop can recover the carrier. A Costas loop, shown in Figure 31 is able to both recover the carrier and demodulate the data from such a signal. If there were no modulation, the upper arm could be considered simply a PLL which could lock to a carrier. The effect of the lower arm of the loop is to lock to the modulation and cancel it out of the upper arm of the loop.

10.3 Clock/Data Recovery

An issue that pervades communication systems is that of extracting a data clock and the data itself from an incoming signal. Known as clock/data recovery (CDR), this problem presents some unique issues not found in other PLL applications. Two common encoding methods, Return to Zero (RZ) and Non-Return to Zero (NRZ) are shown in Figures 32. Depending upon which format is used, a different version of clock recovery must be used.

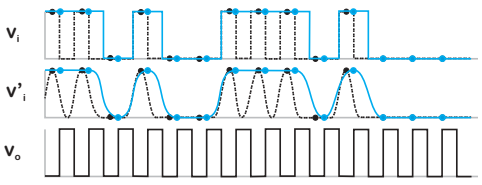

Figure 32: RZ data (dark dashed) and NRZ data (solid light)

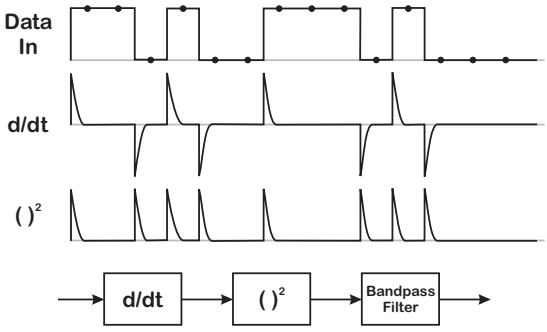

Figure 33: Conversion from NRZ data to RZ data using analog circuits.

RZ formats have the advantage that the clock signal shows up in the data signal spectrum. However, in doing so, they require twice the bandwidth of data encoded using NRZ format.

The tasks of recovering the clock frequency is often separated from that of recovering the data and the clock phase. With RZ data a PLL can usually lock onto the clock directly. However, with NRZ data, one must first generate a signal whose spectrum contains the clock frequency. The circuit shown in Figure 33 detects the edges of the NRZ data by differentiating and then uses a squarer

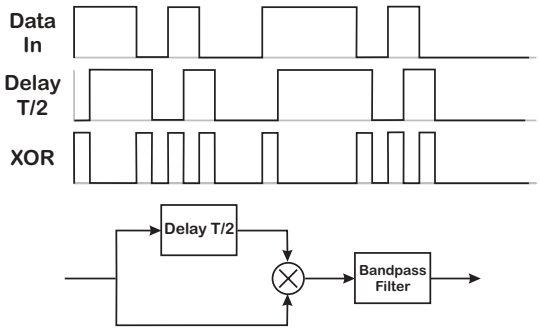

Figure 34: Conversion from NRZ data to RZ data using digital circuits.

to rectify this signal. (A full wave rectifier could also have been used.) The rectified signal is fed through a bandpass filter to obtain an input signal for a PLL. Depending upon the frequencies involved and noise in the signal, a similar result can be obtained with digital logic as shown in Figure 34.

10.4 Frequency Synthesis

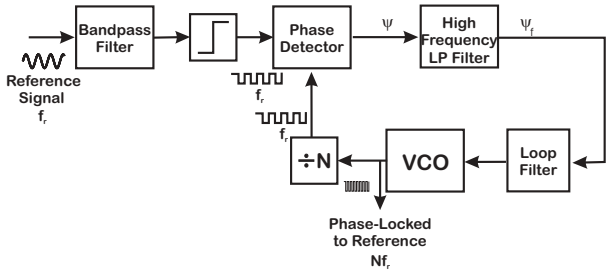

Figure 35: A Harmonic Locking Phase-Locked Loop which uses a phase detector.

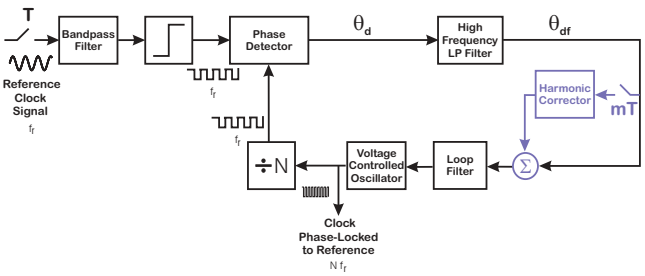

Figure 36: A harmonic locking PLL with optional multi-rate harmonic correcting capability.

In some cases, it is desirable to have a clock which is phase locked with an input signal of some different frequency. A common example of this is in synthesizing a frequency from an input signal at a different (often lower) frequency. A variant of a PLL called a harmonic locking loop where the VCO frequency is at some multiple of the input signal frequency, as shown in Figure 36. The output of the divide by N operation is at the same frequency as the input signal and thus the phase detector can provide an error signal. For this to happen, however, the VCO must run at a frequency N times the input frequency. It is also possible to have values of N which are not integers.

A recent example from the storage industry is the case of linkless editing in the DVD+RW optical disk drive format [35, 36] is enabled by locking a harmonic locking loop to a high frequency oscillation of the track walls on the disk as shown in Figure 37.

In some applications, the clocking loop is affected by

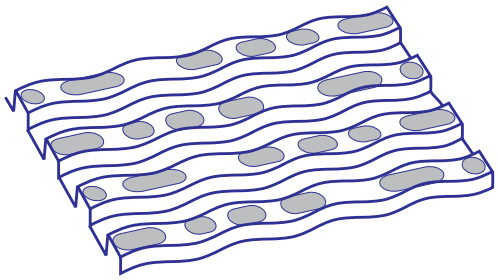

Figure 37: A perspective schematic of high frequency wobbles. This allows a rewritable DVD optical drive to use the DVD-ROM format. There are no sector marks or edit gaps. The address and timing information are encoded in the wobble.

harmonic disturbances. The same type of harmonic compensation done in rotating storage systems can be applied to remove this component of the phase error [37] as shown by including the highlighted harmonic corrector in Figure 36.

10.5 Modulation/Demodulation (Phase/Frequency)

11 PLL Applications in Control Problems

Several applications of PLLs relate to controlling moving objects, rather than only the tracking of signals. Three examples of these will be presented in this section.

11.1 Disk Drive Control

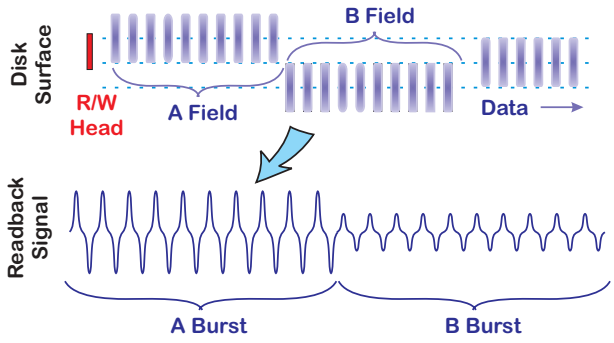

Figure 38: Amplitude encoded position error signal (PES) in a hard disk. PLLs are used to time the acquisition of the readback signal. Correct demodulation of the A and B fields depends upon proper timing.

One broad use of PLLs is in storage systems. This discussion will focus on hard disks, although similar issues exist for optical disks and tape drives. Disk drives encode the cross-track position in a variety of ways, but they all

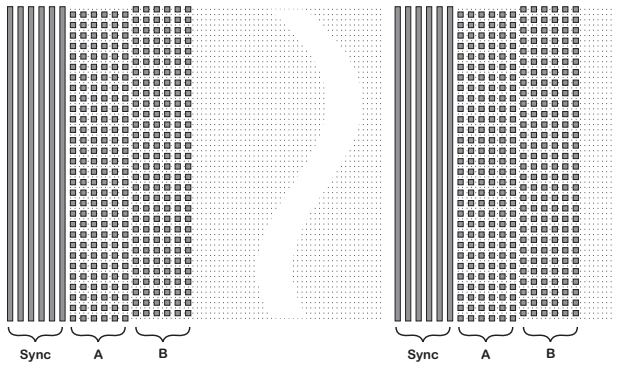

Figure 39: Layout of amplitude encoded position information on a hard disk. The role of the PLL is to resynchronize the timing before the servo signals and the data can be read.

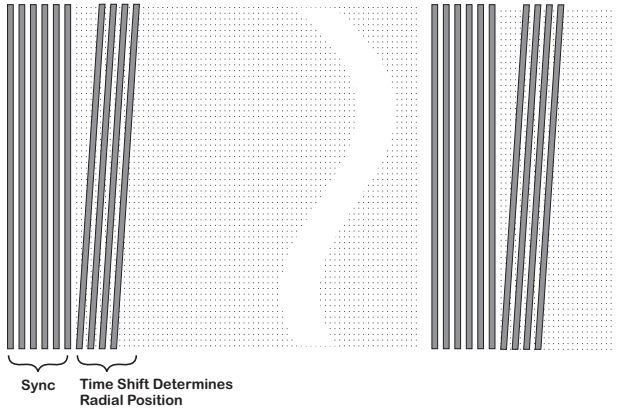

Figure 40: In phase encoding of position error the phase difference between the reference mark and the position mark gives a measure of the cross track position.

require some type of PLL to synchronize the reading of the position signal with the rotation of the disk [38]. In sectored servo – the dominant format for modern hard disks, the data and the position information are interlaced on the same disk. In amplitude encoded position information (Figures 38 and 39), the read/write head demodulates the amplitude of the A and B fields separately. If the head is on track, the amplitudes of the demodulated signals are nominally equal. If the head is off to one side, then either the A amplitude or the B amplitude will be larger. It should be clear that correct demodulation of these fields depends upon proper timing information, so that the disk drive can distinguish between the different servo position fields, as well as the data. The synchronization pattern can be seen in Figure 39. It precedes the position fields and the data fields. This allows the clock to be recovered at the beginning of each sector. Although the most common encoding is called amplitude encoding, an alternate example uses phase encoding of position error [39], shown in Figure 40. This method was used on IBM's first Magneto-Resistive (MR) Head drive, the Corsair [40]. The earlier MR heads had highly nonlinear cross track behavior. Rather than deal with this, the IBM scheme wrote reference timing patterns followed by slanted position marks. The PLL would lock to the reference pattern and then the drive would do a time of flight calculation to the position marks. If the relative phase was 0, the head was on track. Deviations from the on track position produced phase differences which could be used to servo the head.

11.2 Harmonic Compensation

One of the control oriented applications of PLLs that has arisen the in the past few years is the one cancellation of harmonic disturbances [41, 42]. Most of these algorithms assume knowledge of the frequency of the periodic disturbance. However, when that information is not available, a PLL-like approach can be used to first estimate the disturbance frequency and feed this into the harmonic corrector [16, 43, 44]. Furthermore, this approach is also useful in frequency estimation [45, 46].

11.3 Motor Control

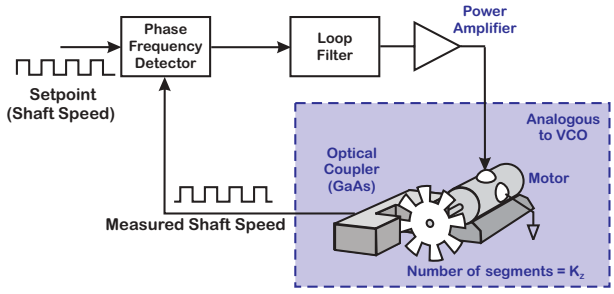

Figure 41: Motor control using PLL techniques.

The use of PLL techniques for motor speed control is described in Best [8]. Basically, in place of a linear control system with a speed setpoint as the reference input and a tachometer to measure the motor's rotational velocity, a system as shown in Figure 41 is used. Here, the tachometer has been replaced by an "optical tachometer" consisting of a segmented wheel attached to the rotor shaft and an optical coupler. The optical coupler consists of a photodiode followed by a Schmitt Trigger which produces an oscillation proportional to the motor speed. The motor and optical coupler together replace the VCO of the PLL. A phase-frequency detector (Section 6.4) is used as it has an infinite pull in range and barring damage to the segmented wheel there will be no missing samples.

There are several interesting points about PLL based motor speed control. First, the motor model itself is sec-

ond order (rather than the first order model of the VCO):

$$G_m(s) = \frac{K_m K_z}{s(1 + sT_m)}, \tag{22}$$

where K_m is the motor torque constant, T_m is the motor time constant, and K_z is the number of segments in the wheel. Allowing for the gain of the power amplifier to be K_a, the gain of the PFD to be K_d and the loop filter to be an integrator with a minimum phase zero, the open loop linear response of the system is defined by [8]:

$$H_{OL}(s) = K_a K_d \left(\frac{1 + s\tau_2}{s\tau_1}\right)\left(\frac{K_m K_z}{s(1 + sT_m)}\right). \tag{23}$$

During normal operation, the PFD will be in a nonlinear regime [17] as the motor speed is ramped to different setpoints. Some improvements to the steady state response have recently been reported by modifying the behavior of the PFD [47].

12 Some Advanced Topics

With all the available topologies and technologies, it should be clear that the choice of what to use is very application dependent. Digital techniques have the advantages that they are relatively immune to circuit drift and easy to integrate into small chip packages. However, they are not suitable for every application.

For high noise input signals, the traditional mixer is still preferred as it makes the best use of information in the signal amplitude to reject noise. Classical digital PLLs are extremely useful in high speed digital communications systems.

Some of the tradeoffs can be seen in high speed digital communications systems. As the standard for these systems reaches beyond 10 gigabits per second (Gbps), to 40 and 100 Gbps, the circuit technologies are hard pressed to produce the short pulses of phase detectors such as the linear Hogge detector (Section 6.6). Thus, the nonlinear Alexander detector (Section 6.7) is often preferred. This is because the latter detector's components are easy to integrate and the detector itself has pulses that are no shorter than half a bit interval.

At the other end of the spectrum, as processing power goes up, it becomes more practical to sample the data and perform all the relevant operations in software (Section 5.3). Not only is this the ultimate in flexibility, but it can also be the lowest in cost.

13 Areas for Contribution

It seems that there are several areas where use of advanced control methods might improve PLLs. The first obvious one is in the area of nonlinear analysis. While it is true that many loops can be considered to have linear phase detectors close to lock, this is not true when the loop is unlocked and is never true for the Bang-Bang PLL. Furthermore, more analysis of the high frequency detector effects on the loop performance and the injection of noise into the clock signals would be useful. Design tools to optimize designs of PLLs for both phase performance and signal performance would be another area of strong contribution. Some of these tools might also be useful in designing new phase detectors that are optimized for a particular type of loop.

For a variety of reasons mentioned earlier, PLLs are always low order. However, knowing how to obtain stable high order nonlinear PLLs should allow for extra loop shaping so common in other control systems.

Finally, efficient simulation of the entire PLL is a broad area to study. Most packages break the problem into signal simulation and phase response simulation. Methods to allow for complete simulation of the system despite the two time scales would be very useful and could further be applied to software PLLs.

14 Useful References

Andrew Viterbi's classic book (sadly out of print), *Principles of Coherent Communication* [5], has a wonderful first introduction to the analysis of a PLL. Floyd Gardner's famous little book, *Phaselock Techniques* [3], has been the classic first book for PLLs. It provides basic analysis and applications for PLLs. Another of the classic PLL text is Alain Blanchard's book, *Phase-Locked Loops* [4].

Dan Wolaver's book, *Phase-Locked Loop Circuit Design* [6] provides excellent coverage of the different circuits used in PLLs, including many different phase detector models. Wolaver tends to focus on classical analog and classical digital PLLs.

Roland Best's book, *Phase-Locked Loops: Design, Simulation, and Applications* [8] provides a more classical analysis of PLLs. It does an excellent job of describing the various classes of PLLs, including classical analog, classical digital, all digital, and software PLLs. Furthermore, a software disk is included. However, the treatment of actual circuits is far more cursory than Wolaver's book.

The IEEE Press has published two books containing papers on PLLs, both co-edited by William C. Lindsey. *Phase-Locked Loops and Their Application* [11], co-edited with Marvin K. Simon, contains many of the seminal papers on PLLs. *Phase-Locked Loops* [12], co-edited with Chak M. Chie, has a larger emphasis on digital loops. A third book from the IEEE, edited by Behzad Razavi on *Monolithic Phase-Locked Loops and Clock Recovery Circuits: Theory and Design* [13] goes into much more depth on issues of integration of PLLs and CDR circuits into silicon. The collection starts with an excellent tutorial [30].

Hsieh and Hung have a nice tutorial on PLLs that not only includes the basic theory, but also some applications, particularly to motor control [19].

Paul Brennan's book, *Phase-Locked Loops: Principles and Practice* [7] is a brief book with a practical bent. It gives an excellent explanation of phase-frequency detectors and charge pumps.

Donald Stephens' book, *Phase-Locked Loops for Wireless Communications: Digital and Analog Implementation* [10] has an interesting history section in the front as well as a method of approaching analysis of digital PLLs that is closer to a typical sampled data approach than most books. This second edition of the book includes information on optical PLLs.

Acknowledgments

The understanding of PLL circuit components presented here has been greatly enhanced by discussions with Rick Karlquist, Rick Walker, and Len Cutler of Agilent Laboratories and with Salam Marougi of Agilent Technologies' Electronic Products and Solution Group. The control systems interpretation of various loop components improved dramatically after discussions with Gene Franklin of Stanford University. To them I owe a debt of gratitude for their insight and their patience.

References

[1] S. C. Gupta, "Phase-locked loops," *Proceedings of the IEEE*, vol. 63, pp. 291–306, February 1975.

[2] W. C. Lindsey and C. M. Chie, "A survey of digital phase-locked loops," *Proceedings of the IEEE*, vol. 69, pp. 410–431, April 1981.

[3] F. M. Gardner, *Phaselock Techniques*. New York, NY: John Wiley & Sons, second ed., 1979. ISBN 0-471-04294-3.

[4] A. Blanchard, *Phase-Locked Loops*. New York, NY: John Wiley & Sons, 1976.

[5] A. J. Viterbi, *Principles of Coherent Communication*. McGraw-Hill Series in Systems Science, New York, NY: McGraw-Hill, 1966.

[6] D. H. Wolaver, *Phase-Locked Loop Circuit Design*. Advanced Reference Series & Biophysics and Bioengineering Series, Englewood Cliffs, New Jersey 07632: Prentice Hall, 1991.

[7] P. V. Brennan, *Phase-Locked Loops: Principles and Practice*. New York: McGraw Hill, 1996.

[8] R. E. Best, *Phase-Locked Loops: Design, Simulation, and Applications*. New York: McGraw-Hill, third ed., 1997.

[9] J. A. Crawford, *Frequency Synthesizer Design Handbook*. Norwood, MA 02062: Artech House, 1994.

[10] D. R. Stephens, *Phase-Locked Loops for Wireless Communications: Digital and Analog Implementation*. Understanding Science and Technology, Boston/Dordrecht/London: Kluwer Academic Press, second ed., 2002.

[11] W. C. Lindsey and M. K. Simon, eds., *Phase-Locked Loops and Their Application*. IEEE Selected Reprint Series, New York, NY: IEEE Press, 1978.

[12] W. C. Lindsey and C. M. Chie, eds., *Phase-Locked Loops*. IEEE Selected Reprint Series, New York, NY: IEEE Press, 1986.

[13] B. Razavi, ed., *Monolithic Phase-Locked Loops and Clock Recovery Circuits: Theory and Design*. IEEE PRESS Selected Reprint Series, New York, NY: IEEE Press, 1996.

[14] N. E. Wu, "Circle/Popov criteria in phaselock loop design," in *Proceedings of the 1998 American Control Conference*, (Philidelphia, PA), pp. 3226–3228, AACC, IEEE, June 1998.

[15] D. Y. Abramovitch, "A method for obtaining MIMO open-loop plant responses from closed-loop measurements," Invention Disclosure 191166, Hewlett-Packard Co., Corporate Patent Department, M/S 20B-O, Palo Alto, CA, February 1991.

[16] M. Bodson, J. S. Jensen, and S. C. Douglas, "Active noise control for periodic disturbances," *IEEE Transactions on Control Systems Technology*, vol. 9, pp. 200–205, January 2001.

[17] C. A. Adkins and M. A. Marra, "Modeling of a phase-locked loop servo controller with encoder feedback," in *Proceedings of IEEE Southeastcon'99*, pp. 59–63, IEEE, IEEE, March 25-28 1999.

[18] K. Ogata, *Modern Control Engineering*. Prentice-Hall Instrumentation and Controls Series, Englewood Cliffs, New Jersey: Prentice-Hall, 1970.

[19] G.-C. Hsieh and J. C. Hung, "Phase-Locked Loop techniques — A survey," *IEEE Transactions on Industrial Electronics*, vol. 43, pp. 609–615, December 1996.

[20] D. Y. Abramovitch, "Analysis and design of a third order phase-lock loop," in *Proceedings of the IEEE Military Communications Conference*, IEEE, October 1988.

[21] D. Y. Abramovitch, "Lyapunov Redesign of analog phase-lock loops," *The IEEE Transactions on Communication*, vol. 38, pp. 2197–2202, December 1990.

[22] P. C. Parks, "Liapunov redesign of model reference adaptive control systems," *IEEE Trans. on Automatic Control*, vol. AC-11, July 1966.

[23] D. Abramovitch, "Lyapunov Redesign of classical digital phase-lock loops," in *Proceedings of the 2003 American Control Conference*, (Denver, CO), pp. 2401–2406, AACC, IEEE, June 2003.

[24] N. E. Wu, "Analog phaselock loop design using Popov critereon," in *Proceedings of the 2002 American Control Conference*, (Anchorage, AK), AACC, IEEE, May 2002.

[25] A. V. Oppenheim and R. W. Schafer, *Digital Signal Processing*. Englewood Cliffs, N. J.: Prentice Hall, 1970.

[26] J. Alexander, "Clock recovery from random binary signals," *Electronics Letters*, vol. 11, pp. 541–542, October 1975.

[27] G. F. Franklin, J. D. Powell, and M. L. Workman, *Digital Control of Dynamic Systems*. Menlo Park, CA: Addison-Wesley, second ed., 1990.

[28] C. R. Hogge, Jr., "A self correcting clock recovery circuit," *IEEE Journal of Lightwave Technology*, vol. LT-3, pp. 1312–1314, December 1985.

[29] D. Shin, M. Park, and M. Lee, "Self-correcting clock recovery circuit with improved jitter performance," *Electronics Letters*, vol. 23, pp. 110–111, January 1987.

[30] B. Razavi, "Design of monolithic phase-locked loops and clock recovery circuits - A tutorial," in *Monolithic Phase-Locked Loops and Clock Recovery Circuits: Theory and Design*, IEEE PRESS Selected Reprint Series, pp. 1–39, New York, NY: IEEE Press, 1996.

[31] V. F. Kroupa, "Noise properties of PLL systems," *IEEE Transactions on Communications*, vol. COMM-30, pp. 2244–2252, October 1982.

[32] D. B. Leeson, "A simple model of feedback oscillator noise spectrum," *Proceedings of the IEEE*, vol. 54, pp. 329–330, February 1966.

[33] W. Rosenkranz, "Phase-locked loops with limiter phase detectors in the presence of noise," *IEEE Transactions on Communications*, vol. COMM-30, pp. 2297–2304, October 1982.

[34] C. Y. Yoon and W. C. Lindsey, "Phase-locked loop performance in the presence of CW interference and additive noise," *IEEE Transactions on Communications*, vol. COMM-30, pp. 2305–2311, October 1982.

[35] D. Y. Abramovitch and D. K. Towner, "Re-writable optical disk having reference clock information permanently formed on the disk," United States Patent 6,046,968, Hewlett-Packard, Palo Alto, CA USA, April 4 2000.

[36] D. Abramovitch, D. Towner, C. Perlov, J. Hogan, M. Fischer, C. Wilson, I. Çokgör, and C. Taussig, "High Frequency Wobbles: A write clock generation method for rewritable DVD that enables near drop-in compatibility with DVD-ROMs," *The Japanese Journal of Applied Physics*, February 2000.

[37] D. Y. Abramovitch, "Turning the tracking problem sideways: Servo tricks for DVD+RW clock generation," in *Proceedings of the 2000 American Control Conference*, (Chicago, IL), pp. 2615–2620, AACC, IEEE, June 2000.

[38] D. Y. Abramovitch, "Magnetic and optical disk control: Parallels and contrasts," in *Proceedings of the 2001 American Control Conference*, (Arlington, VA), pp. 421–428, AACC, IEEE, June 2001.

[39] Z.-E. Boutaghou, D. H. Brown, K. J. Erickson, and R. Greenberg, "Digital servo signal demodulation method and apparatus utilizing a partial-response maximum-likelihood (PRML) channel in a disk file," United States Patent 5,343,340, International Business Machines Corporation, Armonk, NY USA, August 1994.

[40] E. D. Daniel, C. D. Mee, and M. H. Clark, eds., *Magnetic Recording: The First 100 Years*. New York, NY: IEEE Press, 1997.

[41] T.-C. Tsao and M. Tomizuka, "Adaptive and repetitive digital control algorithms for noncirculating machining," in *Proceedings of the 1988 American Control Conference*, (Atlanta, GA), pp. 115–120, AACC, IEEE, June 1988.

[42] C. Kempf, W. Messner, M. Tomizuka, and R. Horowitz, "Comparison of four discrete-time repetitive control algorithms," *IEEE Control Systems Magazine*, vol. 13, pp. 48–54, December 1993.

[43] M. Bodson, "A discussion of Chaplin & Smith's patent for the cancellation of repetitive vibrations," in *Proceedings of the 37th IEEE Conference on Decision and Control*, (Tampa, FL), pp. 1560–1565, IEEE, IEEE, December 16-18 1998.

[44] B. Wu and M. Bodson, "Multi-channel active noise control for periodic disturbances," in *Proceedings of the 38th IEEE Conference on Decision and Control*, (Phoenix, AZ), pp. 4971–4975, IEEE, IEEE, December 7-10 1999.

[45] B. Wu and M. Bodson, "A magnitude/phase locked loop approach to parameter estimation of periodic signals," in *Proceedings of the 2001 American Control Conference*, (Arlington, VA), pp. 3594–3599, AACC, IEEE, June 25-27 2001.

[46] B. Wu and M. Bodson, "Frequency estimation using multiple sources and multiple harmonic components," in *Proceedings of the 2002 American Control Conference*, (Anchorage, AK), AACC, IEEE, May 8–10 2002.

[47] C. A. Adkins, M. A. Marra, and B. L. Walcott, "Modified phase-frequency detector for improved response of PLL servo controller," in *Proceedings of 2002 American Control Conference*, (Anchorage, AK), AACC, IEEE, May 8-10 2002.

[48] H. D. Bellescize, "La réception synchrone," *L'onde électrique*, vol. 11, pp. 225–240, May 1932.

[49] K. R. Wendt and G. L. Fredendall, "Automatic frequency and phase control of synchronization in television receivers," *Proceedings of the IRE*, vol. 32, pp. 7–15, January 1943.

[50] D. Richman, "Color-carrier reference phase synchronization and accuracy in NTSC color television," *Proceedings of the IRE*, vol. 42, pp. 106–133, January 1954.

[51] W. C. Lindsey and C. M. Chie, "A survey of digital phase-locked loops," *Proceedings of the IEEE*, vol. 69, pp. 410–430, April 1981.