Design and Use of Analog, Digital, Software, and Neuronal Phase-Locked Loops

Ramsey Z. Doany

Ingram School of Engineering, Texas State University, San Marcos, TX, USA

Student Member, IEEE

Abstract

Since their initial invention, phase-locked loops (or PLLs) have been used in a wide range of electrical engineering applications from communication and digital signal processing to digital clocks and control systems. Initially, there only existed the analog PLL consisting of an analog loop filter and voltage controlled oscillator. However, modern advancements in technology have resulted in digital, software, and neuronal (i.e. neuron-based) PLLs. As such, a comparison of tradeoffs and applications of the different design techniques and uses can be valuable to those who are looking to research and use PLLs in modern design. This paper seeks to concisely compare the different PLL designs and their respective applications in history and in the modern day.

Introduction

The Phase-Locked Loop is an electronic device with a wide variety of applications. They are used in all modern cell phones, frequency synthesizers, radios, computers, microcontrollers, on-chip computer clocks, televisions, and digital communication (Abramovich, 2006). Fundamentally, phase-locked loops take an input signal and produce an output signal whose phase is “locked” to the phase of the injected signal. The PLL is a nonlinear device but simple configurations can be analyzed using control theory-based linear systems analysis. More complex configurations however often require circuit simulation software for proper analysis

(Stephan, 2015). This paper will provide a brief history of phase-locked loops as well as a brief overview on phase-locked loop theory and will compare analog, digital, software, and neuronal implementations of phase-locked loops.

History of Phase-Locked Loops

The first analysis and experimentation regarding oscillator synchronization was done by JH Vincent and Edward Appleton in the late 1910's and early 1920's. In the 1930's, Travis conducted research on the need for oscillator control, continuing the mathematical development for PLLs. In the 1940s, oscillator synchronization research was focused on FM demodulation and atomic particle accelerators (Stephens, 2002). The first recognizable PLL was conceived in 1964 with the need for coherent communication. The following year, the first PLL IC was introduced and digital PLLs were introduced in the 1970s. The most widespread application for PLLs throughout this time was in television sets in which the vertical and horizontal sweeps were controlled by a continuous signal that had to be locked to a sync pulse (Abramovich, 2006). Today, PLLs are used in MCUs, cell phones, and a whole host of other applications.

Components of the Phase-Locked Loop

The PLL contains 3 main components: the voltage controlled oscillator (VCO), the phase comparator (or phase discriminator), and the loop filter. The voltage controlled oscillator is a nonlinear device and fundamentally, it produces an oscillating output whose frequency is controlled by an input voltage. The phase comparator (or phase discriminator) is also a nonlinear device and it produces the phase difference between two signals injected into its input. The loop filter is not a required component but is usually utilized to have optimal performance. A block diagram of the basic PLL can be seen in Figure 1.

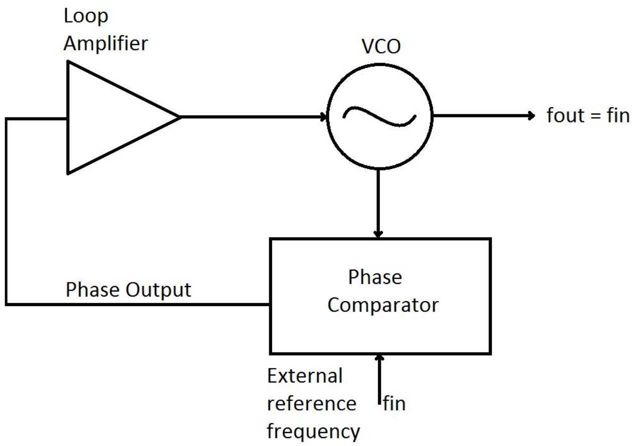

Figure 1. PLL Block Diagram (Adapted from Stephan)

Phase-locked loops are nonlinear but can be analyzed linearly using control theory when used in simple configurations. Luckily, this is often the case. These simple applications are referred to as low order. PLLs can be used to clean up and/or recover weak and noisy signals partially because even if the incoming signal is lost momentarily, the PLL can continue to function and pick the signal back up when it returns. This is of course only possible if the signal disappears for a very brief amount of time (Stephan, 2015).

Types of Phase-Locked Loops

Generally, there are four types of phase-locked loops: analog phase-locked loops, digital phase-locked loops, software phase-locked loops, and neuronal phase-locked loops. Predictably, the first phase-locked loops were analog. Since then parts of the phase-locked loop have been realizable using digital and software components creating new implementations of phase-locked loops, each with their own advantages and disadvantages. The newest implementation is the neuronal phase-locked loop that utilizes an artificial neural network.

Analog Phase-Locked Loops

Most analog phase-locked loops are not truly analog. To achieve optimal performance, they are usually mixed-signal devices (Stephan, 2015). When designing an analog (or any) PLL, it is important to decide the order of the configuration. The first order configuration is one in which the loop filter response is identical to a single pole LP filter. While this layout is inflexible, it can be analyzed using linear systems analysis. Higher order PLLs are more flexible but are also of rather high complexity. High order PLLs also have the advantage of high noise rejection due to the fact that their frequency response is very selective (Almeida, 1999).

A typical analog PLL IC is the CD4046B. It's block diagram can be seen in Figure 2.

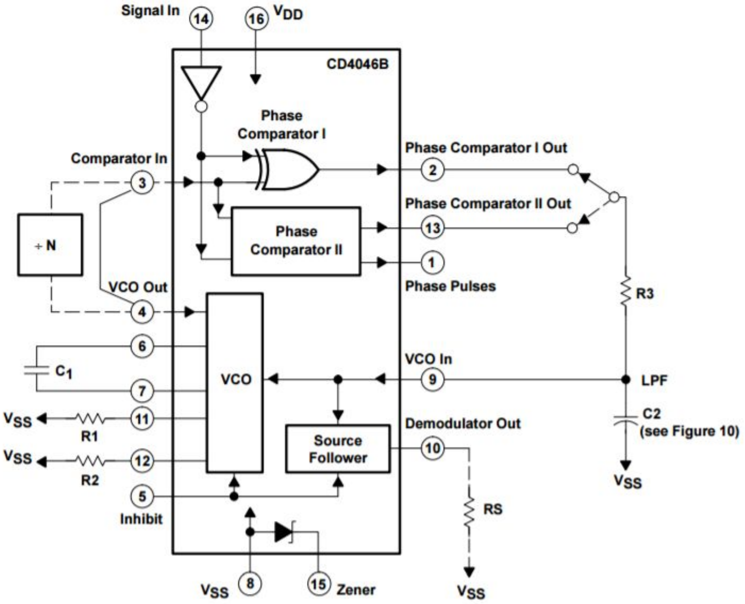

Figure 2. CD4046B PLL Block Diagram. Source:

https://johnthacher.com/2017/03/03/phase-locked-loop-lab-4046/

The block diagram of the CD4046B clearly shows the various components of the PLL. The input signal is sent to the phase comparator whose output goes to the loop filter whose output is sent to the VCO whose output is finally sent back to the phase comparator.

The CD4046B has two outputs: the XOR output and the charge pump output.

The XOR output is low when the input frequency and local oscillator are in phase and is high when the input frequency and local oscillator are out of phase. A visual depiction of this can be seen in Figure 3.

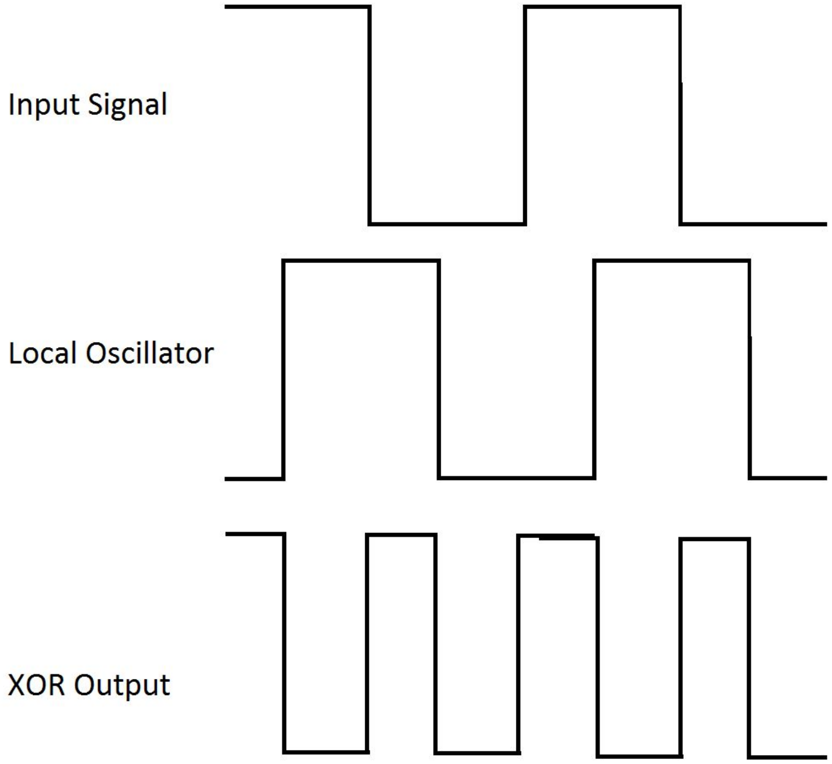

Figure 3. Phase Comparator XOR Output

The charge pump outputs the difference in frequency as a series of pulses, the width of which signify the difference in phase between the two input signals at the phase comparator. When the input frequency is higher than the local oscillator frequency, the output is positive and the local oscillator increases its frequency until locking occurs and the period of the charge pump output pulse goes to 0. When the input frequency is lower than the oscillator frequency, the output is negative and the oscillator increases its frequency until it is locked to the input and the period of the charge pump output pulse goes to 0. A visualization of this can be seen in Figure 4.

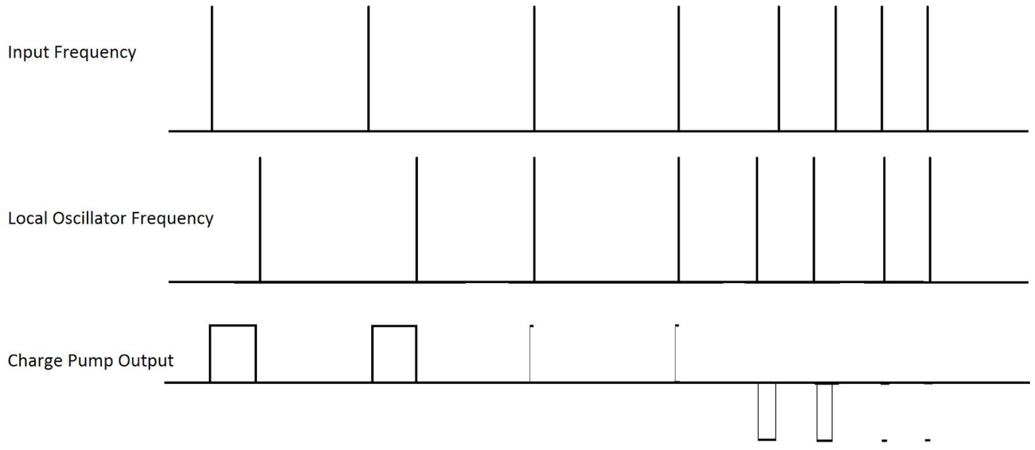

Figure 4. Phase Detector Charge Pump Output

Digital Phase-Locked Loops

A PLL is considered a digital PLL when some or all of its components are implemented using digital hardware, as opposed to analog hardware. When the PLL is entirely digital, it is referred to as an All Digital Phase-Locked Loop (or ADPLL).

In digital PLLs, logic levels, instead of analog signals, are used in a digital phase detector but the VCO and loop filter are still designed using analog electronics. In ADPLLs, the phase detector, the VCO (called DCO here), and loop filter are all implemented digitally. In the first type of ADPLL, a digitally controlled oscillator (or DCO) is controlled by an input word and outputs a frequency. That frequency is compared to a reference signal and the DCO instructs a counter to increment or decrement based on which signal was lagging (Culp, 1986). Next, the input signal is sent to the phase comparator, the outputs of which are count up/down and are sent to the DCO whose output is fed back into the phase detector. The digital loop filter and phase detector are often implemented using JK flip-flops and XOR gates and the DCO is implemented using a divide-by-N counter (Lata and Kumar, 2013).

Digital phase-locked loops and all-digital phase-locked loops are often used in situations where minimal circuit drift is desired. However, they are not ideal for input signals with poor SNR. DPLLs/ADPLLs have the additional capability of integration into small form-factor IC packages (Abramovich, 2006).

Software Phase-Locked Loops

The software phase-locked loop contains the same basic building blocks of the traditional PLL but some or all of its components are implemented in software. Designing a SPLL generally begins with designing an analog loop filter using a low pass filter and a first order integrator. Next, the transfer function is mathematically determined and the bilinear z transform is used to convert the filter to the digital domain. At this point verification of the filter's stability is done and the process can continue. A basic algorithm for the SPLL is as follows: Determine input signal and adjust its phase, compute the output of the PLL, determine the phase error (this is approximate), and finally inject the phase error into the loop filter and adjust the phase of the output (Gaeddert, 2013).

As might be clear from the number of approximations and the design of the SPLL, the SPLL is very similar to a simulation of a phase-locked loop. This is due to the fact that SPLLs often contain heuristics. Heuristics is a problem solving method that involves a significant amount of estimation. Its goal is to obtain a reasonably accurate (or optimal) solution using non-prohibitively high computational complexity, while having a low probability of an inaccurate (or non-optimal) solution (Marti, 2013).

SPLLs have significant clock rate constraints but with the speed of modern processors, SPLLs can be a cheap and flexible solution for many applications.

Neuronal Phase-Locked Loops

Neuronal phase-locked loops (NPLLs) employ artificial neuronal networks within their feedback loop. The first NPLL patent was issued in 2003 and defines an NPLL as a device "that can decode temporally-encoded information and convert it to a rate code… based on an algorithm similar to that of the electronic PLL, but is a stochastic device, implemented by neuronal networks (real or simulated)" (Ahissar, 2003). To understand what this exactly entails, a review of some of the terms used in the definition must be understood as well.

The device being stochastic means that it functions probabilistically. Rate coding and temporal coding are models to understand and simulate brain activity. In what is called a spiking neural network (SNN) where a network of neurons are firing, such as in the brain, the spikes are modeled in two main ways: rate and temporal. In rate coding, a mean of time between neuron firings is measured but the exact time is not, whereas in temporal coding, the specific time at which each neuron fires is measured but their average time between firing is not (Kiselev, 2006). The patent mentioned above seeks to use NPLLs to be able to model both and switch between them.

NPLLs are mostly used in brain modeling but they also have some advantages over the other PLL implementations. In most PLL implementations, if the input frequency has a large and fast change, the PLL is likely to unlock because the gain falls significantly enough that no feedback occurs (Raoof and Suratgar, 2008). NPLLs do not suffer from this problem because of their use of adaptive neural networks.

An adaptive neural network (or ANN) is a network of processing elements, each given a "weight" that work together to solve a problem and do so by learning by example and repetition, much in the way that the brain does (Stergiou and Siganos). In NPLLs, ANNs can be used to minimize error in the phase detector so long as the processing elements are weighted correctly and thus are able to continue to lock even with wide and fast input swings. NPLLs tend to be much more complex and more expensive than the other implementations and have very specialized applications.

Conclusion

In this research paper, the history and theory of phase-locked loops is discussed and a comparison of the design and application of analog, digital, software, and neuronal phase-locked loops is done. The comparisons include the methods of creating and implementing the phase comparator, loop filter, and voltage controlled oscillator as well as additional components such as the adaptive neural network in the NPLL and the benefits of these additional components. A general comparison of cost and complexity is also discussed.

References

Abramovich D 2006 Phase-locked loops: a control centric tutorial *Agilent Labs* [Webinar Presentation]

Ahissar E 2003 Neuronal phase-locked loops *Yeda Research and Development Co. Ltd.* Patent No. US 6581046 B1 [Patent]

Almeida T M 1999 High performance analog and digital PLL design *ISCAS* [Conference Paper]

Culp N L 1986 Digital phase locked loop *Honeywell Inc.* Patent No. US4577163 A [Patent]

Gaeddert J 2013 Writing a phase-locked loop in straight C www.liquidsdr.org [Online]

Kiselev M 2016 Rate coding vs temporal coding - is optimum between? *IJCNN* [Conference Paper]

Lata K and Kumar M 2013 All digital phase-locked loop: a survey *IJFCC* **2** [Journal Paper]

Marti R 2013 The linear ordering problem *Springer Berlin* [Book]

Raoof M and Suratgar A 2008 Adaptive control of PLL using neural network *Journal of the Franklin Institute* **346** [Journal paper]

Stephan K 2015 Analog and mixed-signal electronics *Wiley* [Book]

Stephens D R 2002 Phase-locked loops for wireless communication *Kluwer Academic* [Book]

Stergiou C and Siganos D Neural Networks *Imperial College of London Department of Computing* [Online Paper]