High-Order Phase-Locked Loop Design and Test for Time-Mode Signal Processing Applications

Kun Chuai
Master of Engineering
Department of Electrical and Computer Engineering
McGill University, Montreal

A thesis submitted to McGill University in partial fulfillment
of the requirements for the degree of Master of Engineering

June 2010

Abstract

This thesis first presents a pole-zero placement algorithm for the systematic design of high-order phase-locked loops (PLL) serving as anti-imaging and anti-aliasing filters for time-mode signal processing applications. A 6th order PLL is designed and fabricated on a printed circuit board and is interfaced to a production mixed-signal tester. The correct filtering operation and large-signal transfer characteristic of the PLL are verified with an all-digital DFT solution. The digital test input is driven by a single clock, which can be programmed directly from an ATE high-speed digital pattern generator. As application of these high-order PLLs, an accurate and low-cost clock delay generation system is presented. With proper compensation and calibration, a delay resolution of 15 ps is achieved over an 8.4 ns range. This technique is shown experimentally to be a viable solution for clock alignment and for measuring jitter at a 50 GHz effective sampling rate.

Abrégé

Ce mémoire présente tout d'abord une approche systématique descendante pour la conception de boucles à verrouillage de phase (PLL) ayant un ordre arbitraire et opérant comme filtre anti-image ou anti-repliement pour le traitement de signal dans le domaine temporel. Un PLL de 6e ordre a été conçu et fabriqué sur une carte de circuit imprimé montée sur un tester à signaux-mixtes (ATE). La fonction de filtrage et la caractéristique de transfert de grands-signaux sont vérifiées à l'aide d'une solution de conception pour test (DFT) entièrement numérique. Le signal d'entrée numérique est cadencé par une horloge unique. Par conséquent, le signal de test peut être programmé sans effort à partir de l'instrument numérique à haute-vitesse (HSD) d'un testeur à signaux-mixtes (ATE). De plus, un système précis et économique de génération de délai d'horloge est présenté comme une application du PLL construit. A l'aide de calibration et compensation appropriées, une résolution de délai de l'ordre de 15 ps est réalisée pour un intervalle de 8.4 ns. Cette technique est démontrée expérimentalement comme étant une solution viable pour l'alignement d'horloge et pour mesurer le vacillement d'horloge à un rythme de sous-échantillonnage de 50 GHz.

Acknowledgements

First, I would like to thank my thesis supervisor, Professor Gordon Roberts for his guidance and advice, since the day he gave me an opportunity to work in his research group. His enthusiasm and dedication has motivated me throughout my research and he has always been available whenever I need any kind of advice. I learn something from him every day and I will always be grateful.

I would also like to thank the Natural Sciences and Engineering Research Council (NSERC) and Le Fonds québécois de la recherche sur la nature et les technologies (FQRNT) for supporting me financially. They made sure that conducting research is the only thing I need to worry about.

I want to thank all my friends for making masters studies such an enjoyable experience for me. I will miss all the wonderful moments. Special thanks to Sadok Aouini for his insight in digital coding, George Gal for his practical knowledge in PCB layouts and Dong An for his assistance in the testing of my design.

Most importantly, I would like to thank my parents for their endless love and support. This work is dedicated to them.

Claims of Originality

This thesis contains the following original work done in conjunction with my thesis supervisor:

- The constant-clock-input-response and the overall response of a PLL with the VCO free running term taken into consideration discussed in Chapter 2.
- The realization condition of type-II PLLs discussed in Chapter 2.
- The high-order PLL design carried out in Chapter 2.
- Effect of incoherency in loop dynamics described in Chapter 3.
- Application of phase encoding techniques for jitter transfer function measurement, voltage-to-time transfer characteristic in Chapter 3.
- Application of phase encoding techniques for skew cancellation and jitter measurement.
- The digital phase encoding techniques used in the above two claims were developed by Sadok Aouini within the scope of his Ph.D thesis.
- The method of reducing rise/fall time of the PFD output discussed in Chapter 5.
- The simulations and experiments performed in Chapter 5.
- The PCB board design and component choice / sizing discussed in the Appendix.

Table of Contents

List of Tables

List of Figures

Chapter 1
Introduction

A phase-locked loop (PLL) is a negative feedback system where an oscillator generates a signal that is phase and frequency locked to a reference signal [1]. It is extensively used in the industry, such as for frequency synthesis (e.g. generating a 1.2 GHz clock signal from a 1 GHz reference); skew cancellation (e.g. phase-aligning an internal clock to the I/O clock); and extracting the reference clock from a random data stream such as in serial-link transceivers as seen in Fig. 1.

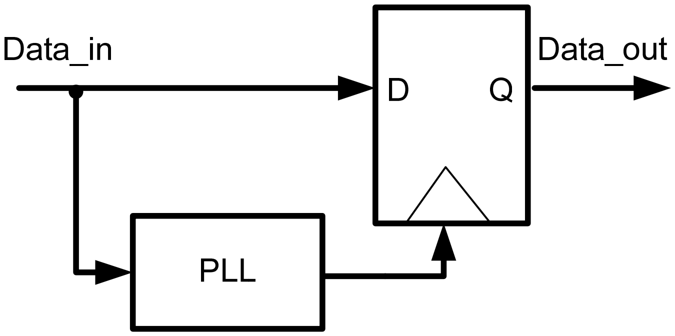


Fig. 1: Clock and data recovery circuit showing a sample PLL application.

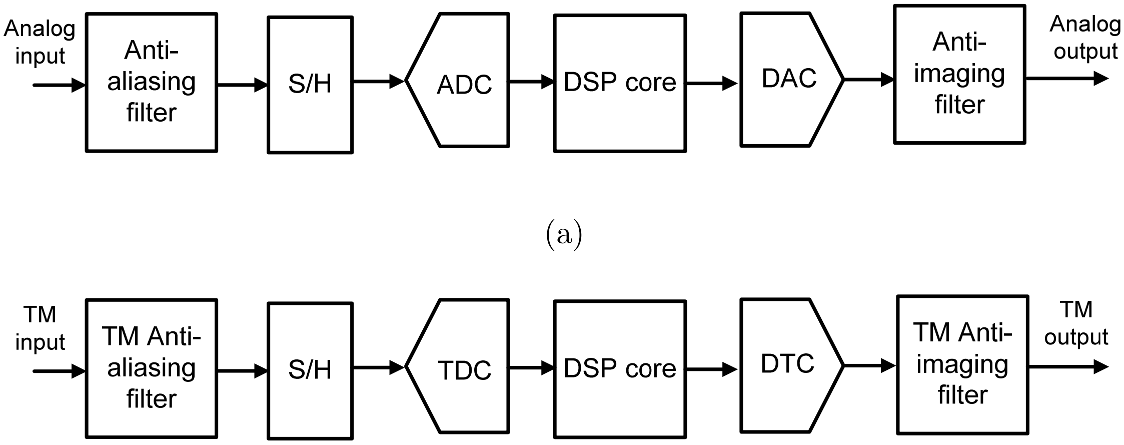
The concept of PLLs was first introduced in the 1930s, and became extremely popular in the 1960's when integrated circuit technology allowed the integration of a complete PLL on a single monolithic substrate [2]. Since then, PLLs have made significant progress and today contributes heavily to modern electronics.

In this thesis, we will focus on the PLLs of high-order in a new field of study, called time-mode signal processing (TMSP), which involves the encoding of information in the form of time-difference variables using phase modulation. The motivation for doing this will be described next.

1.1. Motivation of High-Order PLLs Design

Signal processing is the process of performing mathematical operations on signals. Digital signal processing (DSP) has become extremely popular and has a clear advantage over analog signal processing (ASP) owing it to its simplicity and versatility. Since most signals in the real world are analog in nature, the DSP core [3] needs to interface with these signals through the appropriate analog-to-digital and digital-to-analog building blocks as illustrated in Fig. 2(a). An analog input signal is first band-limited to eliminate any possible noise aliasing that may result during the sample-and-hold process prior to analog-to-digital conversion (ADC). The output after DSP is then converted back to an analog signal using a digital-to-analog converter (DAC) and an anti-imaging reconstruction filter.

Voltage has traditionally been the most common analog signal used for data transmission. However, as the technology scales down, the system voltage is forced to decrease as well; thus, imposing difficulties for the design of mixed-signal circuits such as ADCs and DACs. To resolve this issue, signals of other analog forms have been explored. For example, time-mode signal processing (TMSP) is a new field of study that involves the encoding of information in the form of time difference variables using phase modulation. Two fundamental blocks: time-to-digital converters (TDC) and digital-to-time converters (DTC) such as in [4] and [5] have been studied extensively over the past few years. DTCs convert a digital input to a time difference variable, whereas the TDCs map such a variable to the digital domain. We can see that in nature they are analogous to the ADCs and DACs used in the traditional voltage-domain processing.



(a)

(b)

Fig. 2: Illustrating the (a) modern voltage-domain signal processing architecture (b) signal processing involving time difference variables.

Referring to Fig. 2(a), we can easily see that in order to map this architecture to the time-mode one shown in Fig. 2(b), time-mode anti-aliasing and anti-imaging filters are still needed. As explained in [6], a phase-locked loop (PLL) circuit, which may be implemented as a time-mode filter, can be used for these purposes. Like any other data conversion, these filters play a critical role and directly affect the system performance. Filter theories suggest that high-order designs that are optimized for filtering, such as Elliptic, Butterworth and etc. should be considered for better performance.

Most modern high-order designs, such as described in [7], take on a dominant pole approach, as seen in Fig. 3, where all but one pole of the PLL loop filter is positioned at a frequency greater than the unity gain frequency. It is easily observed that the frequency response that results is not optimized for filtering. In [8], a pole-zero placement algorithm is used for the systematic design of several

PLLs. However, the loop filter requires a rather small DC gain and as a result, the PLL tuning range is greatly reduced. For example, for their 3rd order design, the PLL tuning range is only about 20% of the VCO range. In addition, only type-I PLLs have been considered in their work. Here a general pole-zero placement method is proposed for designing high-order PLLs with an emphasis on type-II PLLs. Realization conditions related to type-II PLLs will be given.

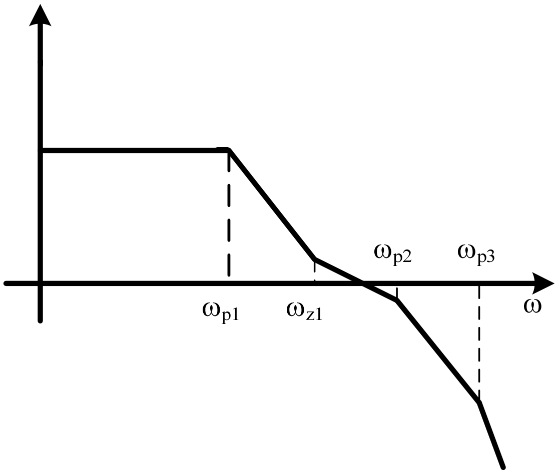


Fig. 3: Illustrating dominant pole PLL loop filter design method.

1.2. Design-For-Test Considerations

Testing a PLL circuit easily and accurately for any applications is as important as having a good design. A common way to test the PLL closed-loop transfer function is to excite the input to the PLL with a phase modulated sine wave and measure the output phase response as shown in Fig. 4. This requires a high performance analog phase modulator and is not optimized for DFT or built-in self-test (BIST) [9]. In [10], an on-chip BIST solution has been proposed. This technique employs a multiplexer and a delay element to sigma-delta encode a sinusoidal test signal of a specific frequency in the phase of the incoming clock signal as illustrated in Fig. 5.

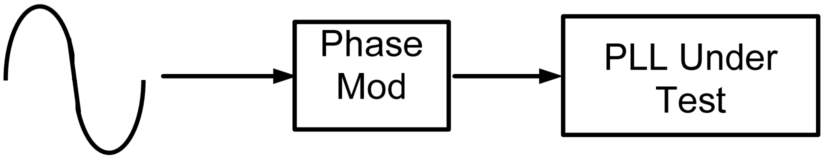


Fig. 4: Traditional PLL transfer function measurement.

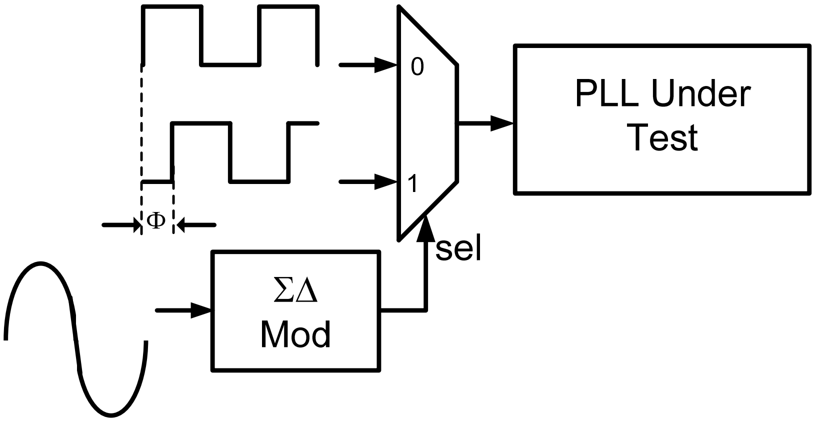


Fig. 5: An on-chip BIST solution for PLL transfer function measurement.

In this work, the multiplexer and delay element described in [10] are bypassed, and the desired phase signal are encoded directly in a cyclic memory; hence, minimizing the setup requirements. Also, arbitrary phase modulated signals can be synthesized as the signal is created in software and only a digital bit stream is passed to the PLL under test. With the added synthesis flexibility, an efficient multi-tone phase modulated test signal approach is described, which greatly reduces testing and computational time while producing almost identical results as the single tone test.

1.3. Applications in Time-Mode Signal Processing

In this thesis, several applications of high-order PLLs will be explored. Specifically, we will look at how high-order PLLs can be used for time-mode signal processing. At the heart of these applications is the creation of a programmable phase delay circuit. Phase delay generation circuits are used in different test systems as part of the signal source or measurement circuit. For example, one issue often encountered in digital testing is the problem of clock skew, i.e., the clock signal arriving to different components at a different time offset. With a phase delay generation circuit, one can tune the phase of each clock until they eventually align with one another to eliminate the skew. Another challenge that test engineers are facing nowadays is that the devices under test (DUT) are running at very high speeds, thus, requiring the Automated Test Equipment (ATE) to run at an even faster speed for a precise DUT characterization. Using a phase delay generation circuit, undersampling techniques can be utilized, and thus relax the speed requirements on the ATE. Chapter 5 will describe such an approach that achieved an effective sampling rate of 50 GHz using a system clock frequency of 66.67 MHz.

One way of generating a clock phase delay in a testing environment is through the use of digital-to-time converters (DTC) [4][11][12]. Fig. 6 attempts to capture the essence of a 3-bit DTC (more later), where a set of digital input words (e.g., 4, 1, 2, 3) is mapped linearly to the delay of a clock. This conversion process is similar to the function of a digital-to-analog converter (DAC). In subsection 1.1, we learned that such a conversion process should always be combined with an anti-imaging filter (e.g. a PLL) to eliminate the images that appear in the output spectrum. Although DTCs are crucial in measurement instrumentation, range finding and direct digital synthesis [13], their use is still relatively new and this is seldom seen in practice. Additionally, due to its lowpass characteristic, PLLs often serve the purpose of reducing the jitter present in the DTC output. This idea can be taken even further by incorporating noise shaping to achieve finer resolution, which will also be explored in this work.

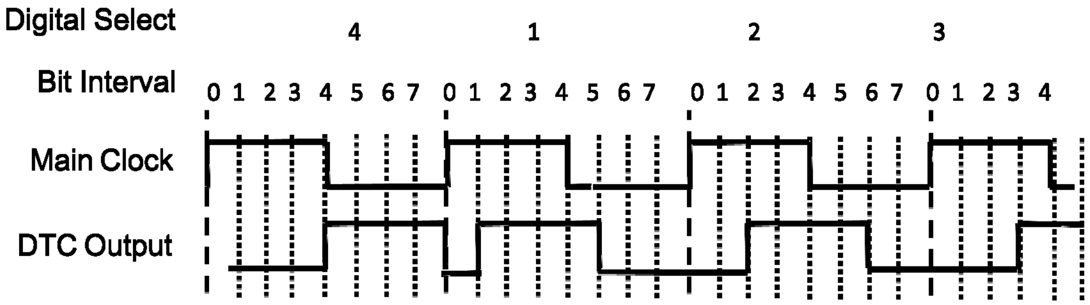


Fig. 6: Illustrating DTC input output relationship.

1.4. Thesis Overview

This thesis begins in Chapter 2 by reviewing the PLL's basic structure and the corresponding signal types. We will then examine its overall response with the VCO free running frequency component taken into consideration and break the response down into two parts: the constant-clock-input response and the excess phase response, along with the analysis of how the former affects the latter in design decision making. Furthermore, the realization conditions have been derived to restrict the type of transfer function that can be realized and the open loop parameters have been formularized in terms of the desired closed-loop parameters.

In Chapter 3, a DFT solution for testing the high-order PLL is described. Sigma-delta modulation is first reviewed, followed by an explanation of a digital-to-phase mapping technique. The test stimuli generation and output capture process are then presented. Three tests, the single-tone, multi-tone and the large signal transfer characteristic tests utilizing sigma-delta modulation and digital-to-phase mapping technique are provided in the end.

In Chapter 4, a 6th order PLL is constructed on a PCB board and interfaced to a Teradyne FLEX tester using the design methodology described in Chapter 2. The correct operation is verified experimentally using the DFT technique explained in Chapter 3. Simulations were also performed before carrying out the experiments and presented in this Chapter.

In Chapter 5, we show that a high-order PLL can be used as a phase delay generation circuit to perform clock skew cancelation and jitter measurement. Both simulation and experiments are performed to validate the proposed applications and results are presented.

Finally, thesis summary and potential future research topics are outlined in Chapter 6.

Chapter 2

PLL Design

In this chapter we shall outline the basic building blocks of PLLs and their corresponding closed-loop operation. We shall describe the closed-loop behavior in terms of a constant-clock input and one with an excess phase component. Finally, this chapter concludes by outlining how high-order PLLs can act as a frequency-selective circuit that acts on the signal carried in the phase of a clock signal.

2.1. PLL Basics Review

2.1.1. Phase Detector

A phase detector is a circuit that creates an output voltage whose average value is proportional to the phase difference of its two inputs [14]. Its implementation can be as simple as an XOR gate as shown in Fig. 7. The width of the output pulse varies linearly with the input phase difference, and therefore so too does its average value, as can be seen from Fig. 8. It can also be shown that this phase detector has a linear input range of π radians. Note that this phase detector reacts to both the rising and falling edges of its inputs and, consequently, it is desired that the two inputs have the same duty cycle.

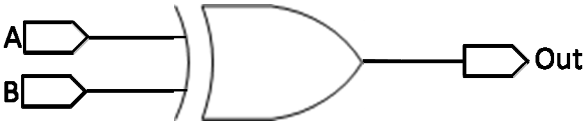


Fig. 7: An XOR phase detector.

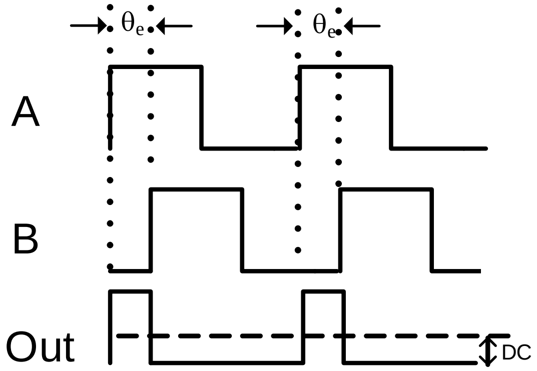


Fig. 8: A sample XOR phase detector output.

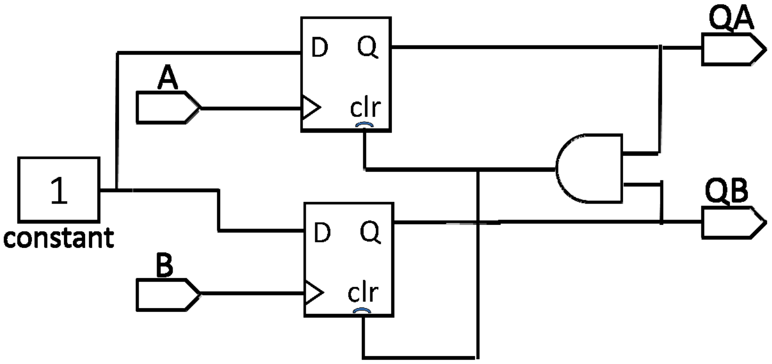


Fig. 9: A phase/frequency detector.

Another common phase detector is the 3-state phase/frequency detector (PFD) as illustrated in Fig. 9. It contains two D-flip-flops and an AND gate to perform the operation summarized by the state diagram shown in Fig. 10. Basically, PFD can detect both the phase and frequency difference of its two inputs, thus providing one more variable to aid PLL acquisition. It has a linear input range of 4π radians, which is a significant improvement over the XOR phase detector of Fig. 7. In addition, it is only sensitive to the rising edge of its inputs and therefore they do not need to have the same duty cycle. One can also note that the PFD has two outputs. In order to convert them to a single output, one can either use a tri-state buffer or a charge-pump circuit [15] as shown in Fig. 11. In general, charge pump circuits are preferred as they are insensitive to power supply variations. Also note that the charge-pump output is in the form of a current rather than a voltage.

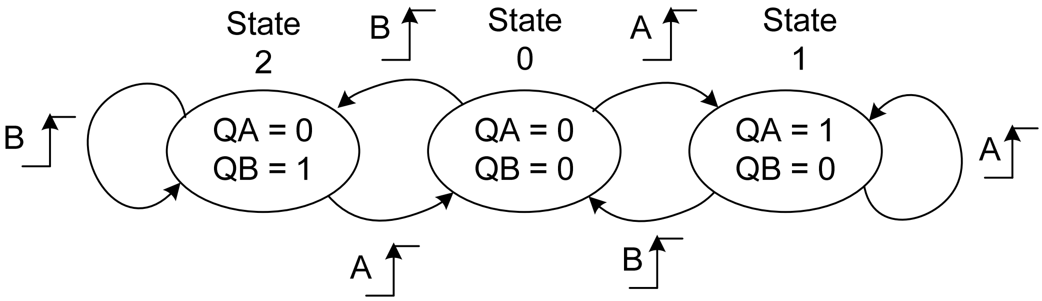
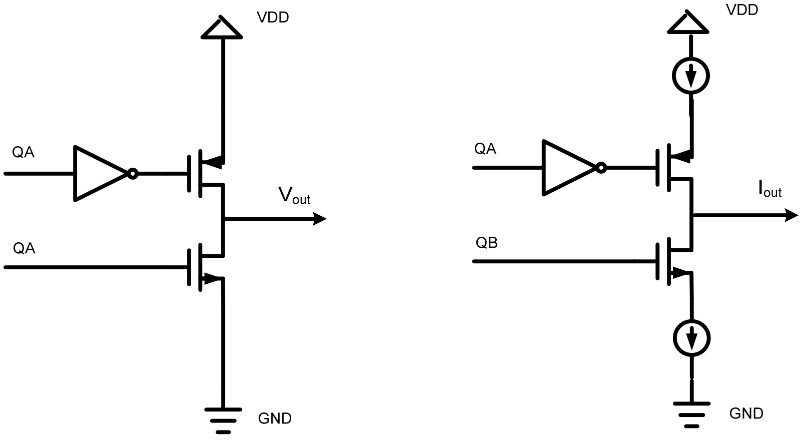


Fig. 10: PFD operation state diagram.



(a) (b)

Fig. 11: (a) A tri-state buffer circuit. (b) A charge-pump circuit.

The major drawback of the PFD is the presence of dead band region. This occurs when the two inputs have a very small phase difference as depicted by the solid line in Fig. 12 corresponding to the PDF transfer characteristic. This dead band region is highly undesirable, especially with type-II PLLs, as the steady-state output of the PFD is nearly 0 forcing the operating point of the PFD to be inside this dead band region. One common practice is to increase the delay of the AND gate shown in Fig. 9, so that dead band region shifts away from the origin, as depicted by the dash line in Fig. 12. If the designer has no control over the delay of this AND gate, i.e, when the AND gate is internal to an IC and the design is carried out at the board level, such as in this work, the PFD will have to be operated at a frequency where the impact of the dead band effect is tolerable. Such a situation is illustrated in Fig. 13(a). Assume that the PFD has a dead band region of 3 ns or ±1.5 ns about the origin. When the PFD is operating at 200 MHz so that it has a period of 5 ns, the input range of 4π radians is translated to 10 ns $\left(\frac{4\pi}{2\pi\times 200\times 10^{6}}\right)$. In this case, the dead band region of 3 ns is clearly significant. In Fig. 13(b), where the PFD is operating at 20 MHz or a period of 50 ns, the linear region becomes 100 ns, and the dead band region of 3 ns is small compared to the overall input range and therefore so too is its impact.

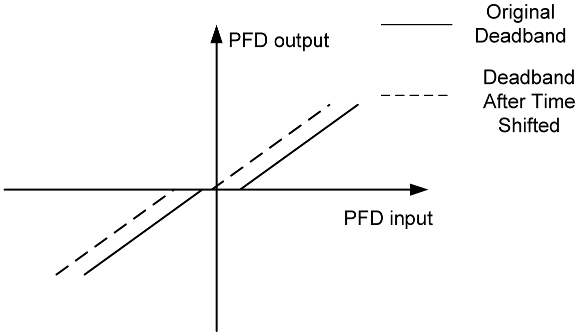
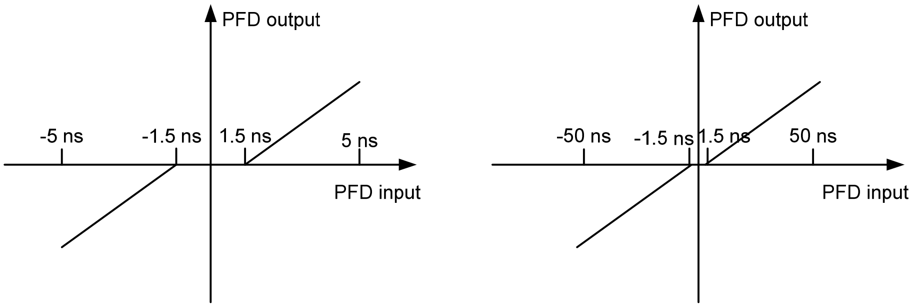


Fig. 12: Original dead band and the dead band after input phase offsetting.



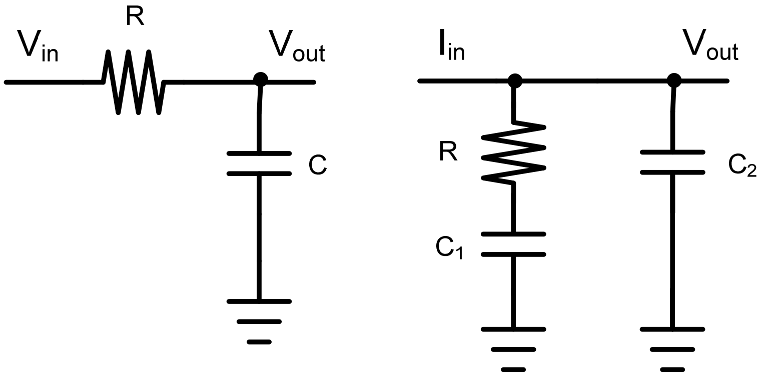
(a) (b)

Fig. 13: Showing a dead band of 3 ns when the PFD is operating at (a) 200 MHz (b) 20 MHz.

2.1.2. Loop Filter

A loop filter is a lowpass filter with general transfer function F(s). It can be constructed from a simple first-order RC filter [16] as seen in Fig. 14 (a). Its transfer function can be written as

$$F(s) = \frac{V_{out}(s)}{V_{in}(s)} = \frac{1}{1+sCR} \tag{1}$$



(a) (b)

Fig. 14: (a) A first-order all-pole RC loop filter and (b) a second-order loop filter for the PFD/charge pump combination.

In the case of PFD when a charge pump is used, the most common loop filter used is shown in Fig. 14(b). The filtering action is performed by R and C_1, while C_2 is used to suppress the ripple caused by current sources mismatch in the charge pump. The transfer function ignoring C_2 is derived to be

$$F(s) = \frac{V_{out}(s)}{I_{in}(s)} = \frac{1 + sC_1R}{sC_1R} \tag{2}$$

Note that this transfer function has a pole at DC. This is a very important feature of this type of loop filter and will be explained further later. PLLs utilizing this type of loop filter are referred to as type-II PLLs, whereas those using a loop filter without any DC poles are of type-I PLLs.

2.1.3. Voltage Controlled Oscillator

The voltage controlled oscillator (VCO) can be modeled as a voltage-to-frequency converter with a gain of K_{VCO} and frequency offset ω_{FR}. As an example, a VCO can be constructed from a ring oscillator as shown in Fig. 15. A ring oscillator needs an odd number of inverting stages N_{OS} in order to oscillate, unless a fully-differential structure is used. The frequency of oscillation f_{out} is given by

$$f_{out} = \frac{I_D}{N_{OS} \bullet C_{tot} \bullet V_{DD}} \tag{3}$$

where I_D is the drain current of each branch, V_{DD} is the supply voltage and C_{tot} is the total capacitance at each stage interface. The frequency of oscillation f_{out} is linear with respect to I_D, and consequently linear with V_{in} as I_D is controlled by V_{in} through a voltage controlled current source.

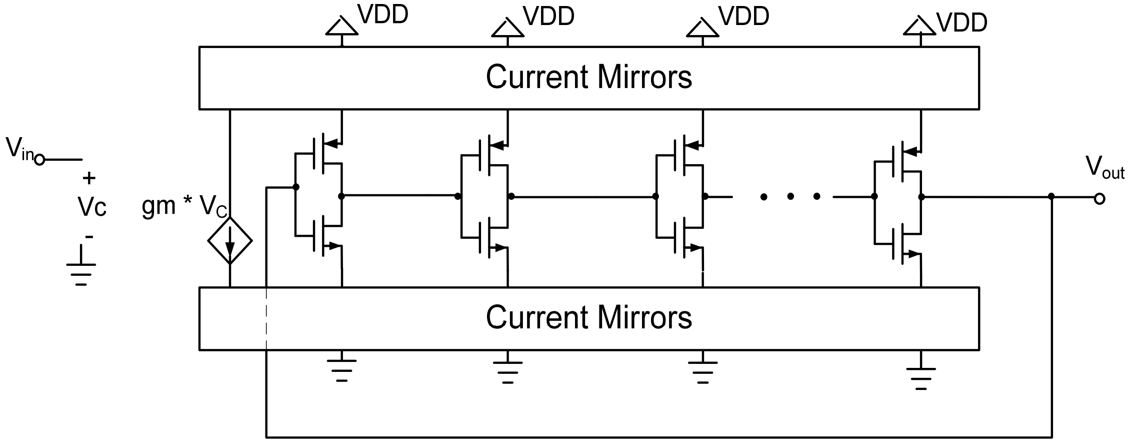


Fig. 15: A ring-oscillator-based VCO.

The circuit shown in Fig. 16 belongs to another important class of VCOs: the LC tank based VCOs. In this particular circuit, the tank is formed by an inductor and the parasitic capacitance of the MOSFET and the diode connected to it. This type of VCO oscillates at its resonance frequency given by

$$f_{osc} = \frac{1}{2\pi\sqrt{LC_{\text{var}}}} \tag{4}$$

The reverse biased diode is used as a varactor. Its capacitance C_{var} is related to the reverse-bias voltage V_R according to

$$C_{\text{var}} = \frac{C_0}{(1+\frac{V_R}{\phi_B})^m} \tag{5}$$

where C_0, ϕ_B and m are process dependent parameters. We can see that changing V_R will change C_{var} and consequently vary the oscillation frequency f_{osc}.

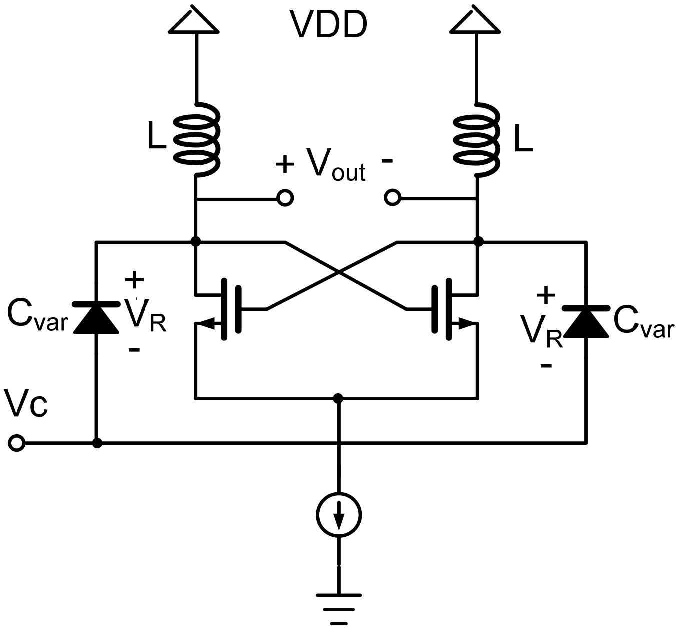


Fig. 16: An LC tank based VCO.

In general, the LC tanked based VCOs generate less noise than the ring oscillator based VCOs, but they have a reduced tuning range. These two issues compete against one another, and the choice is application specific.

2.1.4. Frequency divider

A frequency divider relates its input frequency to its output frequency by a constant dividing factor. A frequency divide-by-2 circuit is shown in Fig. 17. One can verify its operation by the fact that clk_out toggles every time a rising edge of clk_in occurs. Though not explored in this work, the dividing ratio is not restricted to integer values. Fractional values can be achieved on average by switching between two divider ratios. For example, if the divide-by-M circuit is switched to A times and the divide-by-N circuit is switched to B times, the equivalent divider ratio N_{eq} on average can be expressed as

$$N_{eq} = \frac{A+B}{A/N+B/M} \tag{6}$$

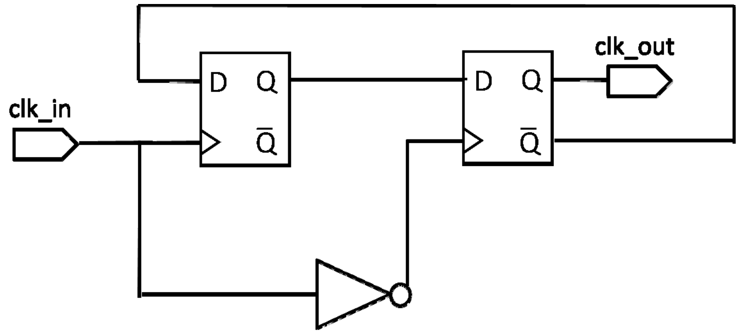


Fig. 17: A frequency divide-by-2 circuit.

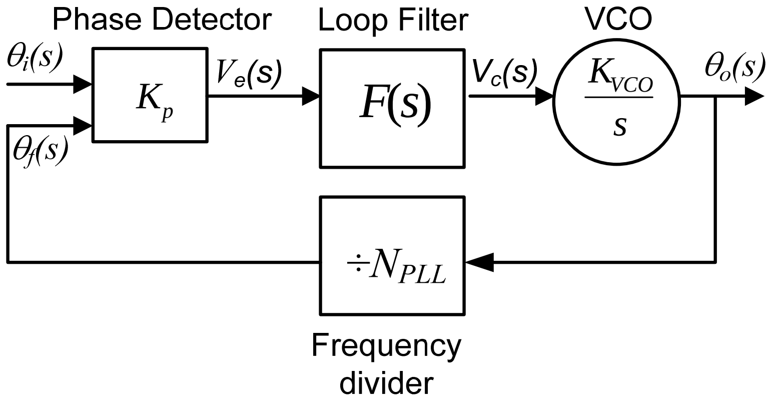


Fig. 18: A block Diagram of a PLL.

2.2. PLL Architecture and Mathematical Modeling

Now that all the PLL components have been introduced, the general architecture of the PLL is ready to be shown. A PLL consists of a phase detector, a loop filter and a voltage-controlled oscillator (VCO) in the feed-forward path and a frequency divider N_{PLL} in the feedback path as shown in Fig. 18.

Under steady state conditions, the PFD creates an output that, on average, is proportional to the phase difference of its two inputs with a gain of K_p, i.e.,

$$V_e(s) = K_p\left[\theta_i(s) - \theta_f(s)\right] \tag{7}$$

The loop filter is a lowpass filter with general transfer function F(s) as discussed earlier. Its function is to reject the unwanted high frequency components produced by the PFD. The VCO can be modeled as a voltage-to-frequency converter with a gain of K_{VCO} and frequency offset ω_{FR}. Since the input and output variables of a PLL are best described in terms of instantaneous phase, the VCO operation can be expressed in the Laplace domain as

$$\theta_o(s) = \frac{\omega_{FR}}{s^2} + \frac{K_{VCO}}{s}V_C(s) \tag{8}$$

The first term accounts for the step change in frequency from a zero-state condition and the second term accounts for the phase contribution from the input controlling voltage. Note that in most PLL literatures, the first term is ignored. In this work, we will take it into account in the analysis and describes how it impacts our design decisions.

Finally, the divider in the feedback path N_{PLL} reduces the frequency or phase of the signal feedback to the PD according to

$$\theta_f(s) = \frac{\theta_o(s)}{N_{PLL}} \tag{9}$$

2.2.1. Constant-Clock-Input Response

Now with the models of all the components described, we can derive the mathematical model for an arbitrary-order PLL. We first examine the response of the PLL when a fixed-frequency constant-phase clock is applied to the PLL.

The input-output behavior of the PLL can be described in general terms as

$$\theta_o(s) = \frac{N_{PLL}\omega_{FR}}{s\left(N_{PLL}s + K_{VCO}K_pF(s)\right)} + \frac{sN_{PLL}K_{VCO}K_pF(s)}{s\left(N_{PLL}s + K_{VCO}K_pF(s)\right)}\theta_i \tag{10}$$

For a constant-clock input ω_{in}, or equivalently, a linear ramp in phase with slope ω_{in}, we write the input signal as

$$\theta_i\left(t\right) = \omega_{in} \cdot t \cdot u\left(t\right) \tag{11}$$

Taking the Laplace transform, we obtain

$$\theta_i\left(s\right) = \frac{\omega_{in}}{s^2} \tag{12}$$

Substituting the input signal into (10) we obtain the output phase term as

$$\theta_o(s) = \frac{N_{PLL}\omega_{FR}}{s\left(N_{PLL}s + K_{VCO}K_pF(s)\right)} + \frac{sN_{PLL}K_{VCO}K_pF(s)}{s\left(N_{PLL}s + K_{VCO}K_pF(s)\right)}\frac{\omega_{in}}{s^2} \tag{13}$$

One could invert the Laplace transform and obtain the transient solution to a constant-clock input; however, this does not reveal anything interesting. Rather, let us look at the steady-state phase error between the input clock and signal feedback to the PFD input, i.e.,

$$\theta_e\left(s\right) = \theta_i\left(s\right) - \theta_f\left(s\right) \tag{14}$$

Combining (9) with (10) and substituting into (14) we write

$$\theta_\varepsilon\left(s\right) = \left[1 - \frac{sK_{VCO}K_pF(s)}{s\left(N_{PLL}s + K_{VCO}K_pF(s)\right)}\right]\frac{\omega_{in}}{s^2} - \frac{\omega_{FR}}{s\left(N_{PLL}s + K_{VCO}K_pF(s)\right)} \tag{15}$$

In steady state, the final phase error becomes

$$\theta_e(\infty) = \lim_{s\to 0} s\theta_e(s) = -\frac{\omega_{FR} - N_{PLL}\omega_{in}}{K_{VCO}K_pF(0)} \tag{16}$$

For time-mode signal processing, it is desired to have zero phase error, so that the phase delay induced in the filtering process is not dependent on the PLL running frequency. Clearly, for zero phase error, the DC gain of the filter response F(s) should be infinite. In other words, the filter requires a pole at DC. In the PLL literature, this is referred to as a type-II response and the frequency range of operation only depends on the VCO range. Another compelling reason for selecting a filter function with a pole at DC is to ensure that the output of the PFD in steady state is zero. This biases the PFD in the middle of its operating range and maximizes its signal handling capability.

2.2.2. Excess Phase Response

The behavior of the PLL excess phase over and above the constant-clock input signal is critical to the quality of the anti-imaging and anti-aliasing filtering action in the DTC and TDC conversion process, respectively, as well as for the applications that will be described in Chapter 3. Consider the input to the PLL consisting of a constant-clock input and an excess phase component, ϕ_i, such that the total instantaneous phase of the input signal can be described in the time domain as

$$\theta_i(t) = \omega_{in} \cdot t \cdot u(t) + \phi_i(t) \tag{17}$$

Taking the Laplace transform, we obtain

$$\theta_i(s) = \frac{\omega_{in}}{s^2} + \phi_i(s) \tag{18}$$

Substituting this expression back into (10), we write

$$\theta_o(s) = \frac{N_{PLL}\omega_{FR}}{s\left(N_{PLL}s + K_{VCO}K_pF(s)\right)} + \frac{sN_{PLL}K_{VCO}K_pF(s)}{s\left(N_{PLL}s + K_{VCO}K_pF(s)\right)}\frac{\omega_{in}}{s^2} + \frac{sN_{PLL}K_{VCO}K_pF(s)}{s\left(N_{PLL}s + K_{VCO}K_pF(s)\right)}\phi_i(s) \tag{19}$$

The output can be organized into two parts: one part is the response to the constant-clock input and the other is due to the excess phase, i.e.,

$$\theta_o(s) = \theta_{CC}(s) + \phi_o(s) \tag{20}$$

Isolating the excess phase terms in (19), we write

$$\phi_o(s) = \frac{N_{PLL}K_{VCO}K_pF(s)}{\left(sN_{PLL} + K_{VCO}K_pF(s)\right)}\phi_i(s) \tag{21}$$

from which we define the excess phase transfer function as

$$\Phi(s) \triangleq \frac{\phi_o(s)}{\phi_i(s)} = \frac{N_{PLL}K_{VCO}K_pF(s)}{sN_{PLL} + K_{VCO}K_pF(s)} \tag{22}$$

It is the magnitude of this transfer function that is used to remove the images or the aliases during the DTC and TDC conversion process. Also, it inherently filters out the quantization error associated with the ΣΔ encoding process for DFT and the applications, which will be described in greater details in the following chapters.

2.2.3. Realization Considerations

For an arbitrary filter function F(s) defined as the ratio of two polynomials of Nth order, i.e.,

$$F(s) = \frac{P(s)}{E(s)} \tag{23}$$

we can re-write (22) by substituting (23) to obtain

$$\Phi(s) = \frac{N_{PLL} K_{VCO} K_p P(s)}{N_{PLL} sE(s) + K_{VCO} K_p P(s)} \tag{24}$$

Here we see the zeros of F(s) are unaltered by the PLL structure and become the zeros of the PLL excess phase response. The poles together with the zeros of F(s) and the PLL parameters set the poles of the excess phase response. We can also conclude that the PLL transfer function is always proper, resulting in either a low-pass or band-pass behavior, but not high-pass.

In the previous subsection we learned that the filter function F(s) must have a pole at DC to eliminate the static phase error. Let us re-write the filter function with explicit polynomial coefficients as follows

$$F(s) = \frac{a_N s^N + a_{N-1} s^{N-1} + \ldots + a_1 s + a_0}{s\left(s^{N-1} + b_{N-1} s^{N-2} + \ldots + b_2 s + b_1\right)} \tag{25}$$

Substituting this back into (22) we obtain

$$\Phi(s) = \frac{\left(K_{VCO} K_p\right) a_N s^N + \ldots + \left(K_{VCO} K_p\right) a_2 s^2 + \left(K_{VCO} K_p\right) a_1 s + \left(K_{VCO} K_p\right) a_0}{s^{N+1} + \left[b_{N-1} + \left(K_{VCO} K_p / N_{PLL}\right) a_N\right] s^N + \ldots + \left(K_{VCO} K_p / N_{PLL}\right) a_1 s + \left(K_{VCO} K_p / N_{PLL}\right) a_0} \tag{26}$$

In order to see more clearly the structure of this transfer function we shall re-write it using two new sets of coefficients and write

$$\Phi(s) = N_{PLL} \frac{c_N s^N + \ldots + c_2 s^2 + c_1 s + c_0}{s^{N+1} + d_N s^N + \ldots + d_2 s^2 + c_1 s + c_0} \tag{27}$$

where

$$\begin{aligned} c_i &= K_{VCO} K_p a_i / N_{PLL} \\ d_i &= b_{i-1} + K_{VCO} K_p a_i / N_{PLL} \qquad for\ i \in 1 \ldots N+1 \end{aligned} \tag{28}$$

This expression reveals the impact of the pole at DC on the excess phase transfer function. Specifically, the first two coefficients of the pole and zero polynomial must be the same.

2.3. High-Order PLL Design

In this section, the goal is to design a PLL for time-domain filtering. Due to the realization condition described earlier, we need to have in Φ(s) the first order term of the numerator to be the same as that of the denominator. This will create a real left-half plane zero that is on the same order of magnitude as any of the poles and as a result, a gain peaking is observed near the band edge frequency (i.e. 3 dB frequency). To limit this impact, various filter functions have been tested as seen in Fig. 19 and the Gaussian filter, which has the most gradual passband roll-off and the lowest group delay, has been found to result in the lowest gain peaking.

A 6th order unity-feedback PLL with a 100 kHz half-power bandwidth has been designed from the following desired 6th order Gaussian filter response:

$$\Phi_{desired}(s)=\frac{2.0306\times10^{34}}{s^6+2.5613\times10^6 s^5+3.0066\times10^{12}s^4+2.0443\times10^{18}s^3+8.4065\times10^{23}s^2+1.9664\times10^{29}s+2.0306\times10^{34}} \tag{29}$$

After meeting the realization condition, the closed-loop response of the PLL becomes

$$\Phi_{actual}(s)=\frac{1.9664\times10^{29}s+2.0306\times10^{34}}{s^6+2.5613\times10^6 s^5+3.0066\times10^{12}s^4+2.0443\times10^{18}s^3+8.4065\times10^{23}s^2+1.9664\times10^{29}s+2.0306\times10^{34}} \tag{30}$$

The loop filter F(s) is then found to be

$$F(s)=\frac{6.4517\times10^{21}s+6.6625\times10^{26}}{s^5+2.5613\times10^6 s^4+3.0066\times10^{12}s^3+2.0443\times10^{18}s^2+8.4065\times10^{23}s} \tag{31}$$

It is built with two second-order Tow-Thomas biquads, together with a first-order active stage, which generates the DC pole and the zero as seen in Fig. 20.

The PFD uses the tri-state buffer in Fig. 11(b) due to the topology of the chosen PLL IC which will be discussed in Chapter 5. Note the resistor R_2 connected between ground and the output of the PFD. It is neither part of the filter circuit nor part of the PFD. Its role is to reduce the rise and fall time of the PFD output when PFD transitions into the high impedance state. Refer to Fig. 21, when the PMOS transitions from ON to OFF and the NMOS transistor is OFF, V_{out} is consequently switched from VDD to the high impedance state. The final voltage in the high impedance state is supposed to be 'gnd', set by the negative input terminal of the opamp. During the transition from VDD to 'gnd', the charge stored on the parasitic capacitor C_{para} needs to be discharged through R_1, which is a slow process as there is no direct DC path to 'gnd'. Therefore, a small resistor R_2 is introduced between V_{out} and 'gnd', so that C_{para} can discharge through it quickly. R_2 should be big enough to pull V_{out} to VDD or VSS when it is not in high impedance state. In the case of the PLL designed in this work, this resistor value was chosen to be 160 Ω. All the component values are listed in Table 2 in the Appendix.

Fig. 22 shows the resulting filter frequency responses compared to an ideal Gaussian filter with the same bandwidth. As one can see, the behaviors are similar except near the passband edge. There we see a maximum error of about 5 dB. This gain peaking is inevitable in the realization of type-II PLLs and may not be desired in some applications, such as those requiring the cascade of several PLLs since it could drastically amplify the phase noise in a small frequency region. It should consequently be taken into consideration in the overall system design. However, it is acceptable for the DFT solution and applications described in this paper as there is no cascading of PLLs and the quantization noise in the gain peaking region is deemed to be small since it is still within the bandwidth of the ΣΔ modulator.

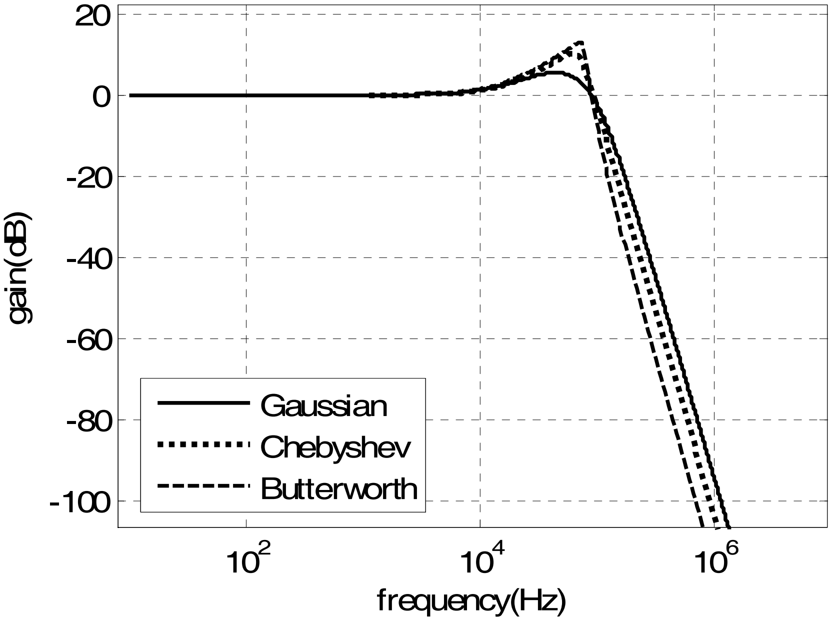


Fig. 19: Gain peaking levels resulted when modifying the Gaussian, Chebyshev and Butterworth filters to meet the PLL realization condition.

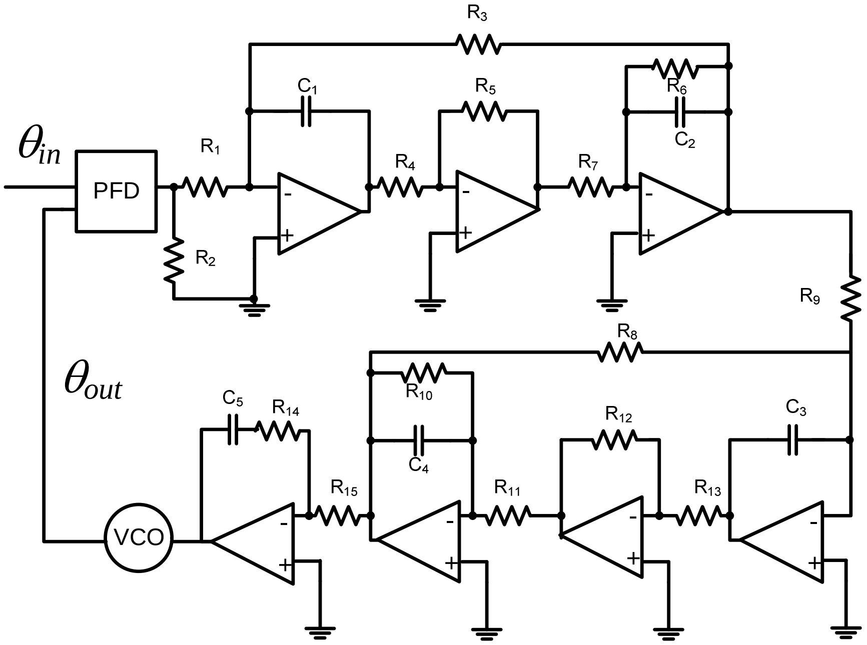


Fig. 20: Circuit realization of a 6th order PLL.

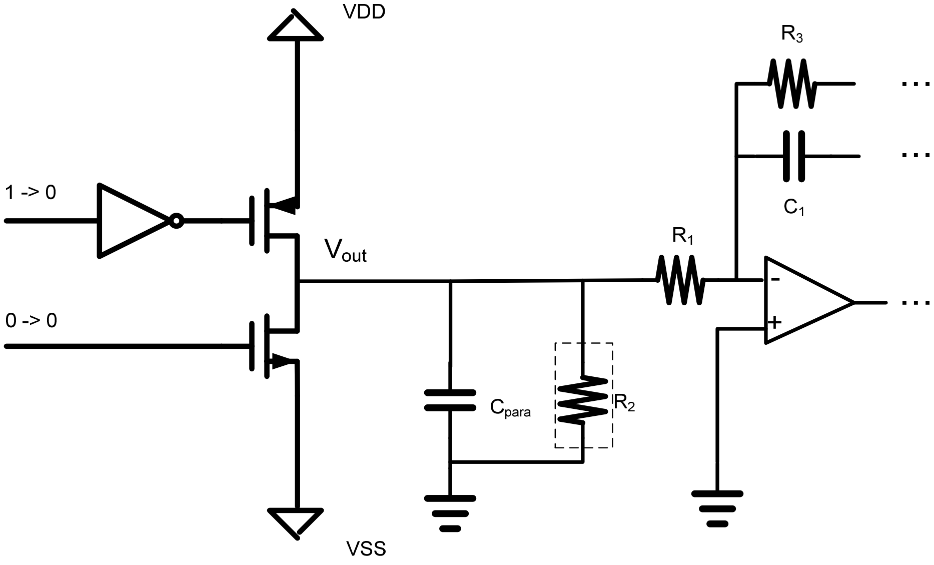


Fig. 21: R_2 is introduced to help C_{para} to discharge.

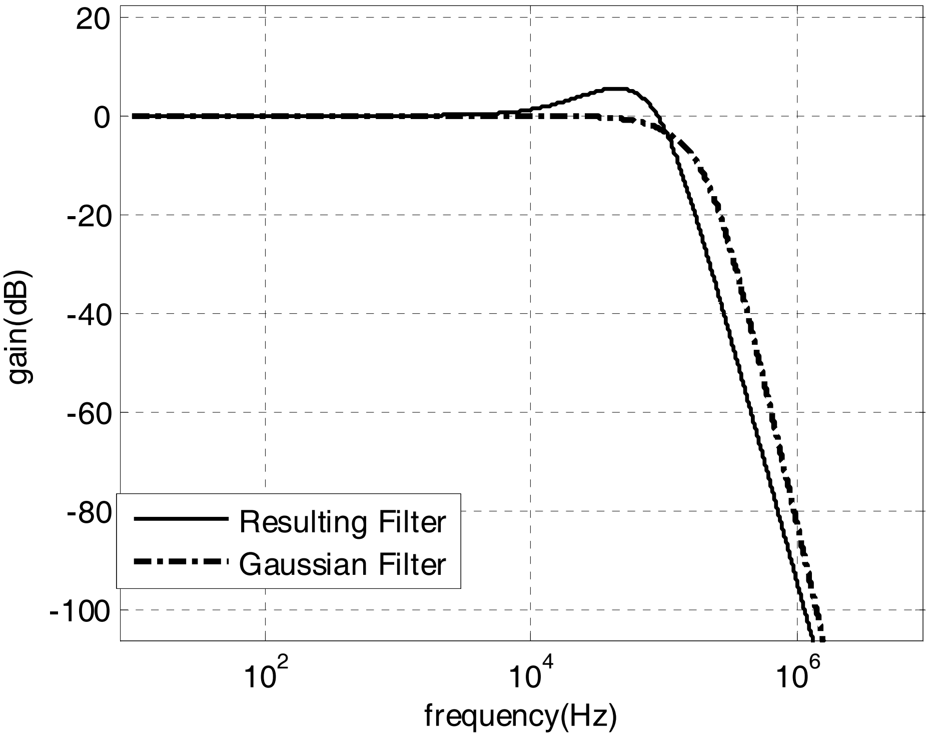


Fig. 22: Magnitude response of a Gaussian filter and that of the PLL closed-loop transfer function when forcing it to meet the realization condition.

2.4. Summary

In this chapter, a pole-zero placement algorithm for the systematic design of high-order phase-locked loops is presented. Basic mathematical models of the building blocks, namely, the PFD, VCO, loop filter and divider circuits are first reviewed. Then, the closed-loop response of the PLL was developed, first for a constant-clock input and then one with the phase-modulated clock input with an arbitrary excess phase component. According to the constant-clock-input response, we conclude that we need to have a pole at DC in the loop filter transfer function for zero phase error. Consequently, realization conditions for the loop filter were derived so that the excess phase response approaches a desired transfer function under DC pole realization conditions. A 6th order PLL was designed to approach a 6th order lowpass Gaussian filter transfer function.

Chapter 3
DFT Solution

The DFT methodology for validating the behavior of the high-order PLL is described in this section. We start by first giving a brief overview of sigma-delta modulation and show how it could be used for generating phase modulated test stimuli. The PLL test setup is then described along with the stimulus generation and response capture implementation. A discussion on single-tone and multi-tone testing followed by the investigation of the static operation of the PLL are presented.

3.1. Overview

For BIST or DFT, it is desired that the testing process be digital in nature for its simplicity, programmability and robustness. When using a digital signal to approximate an analog test signal, quantization noise is unavoidable and the accuracy of the measurement results will be affected. When a signal is quantized or digitized, the resulting signal follows approximately a second-order statistics model with independent additive white noise. In terms of power spectral density (PSD), this process yields approximately an evenly distributed noise floor, the power of which is proportional to the square of the quantizing step size.

PLLs are generally lowpass in nature. When the above digitized test signal is phase modulated and applied to the PLL, the quantization noise outside of the PLL bandwidth is attenuated, whereas those within the bandwidth are still present and influence the measurement results. One could reduce the digitizing step size; consequently, reducing the in-band quantization noise. However, a more efficient way is to use sigma-delta modulation.

3.2. Sigma-delta Modulation

The basic architecture of a first order ΣΔ modulator is depicted in Fig. 23. As can be observed, the concept consists of using feedback to reduce the quantization noise, e(n), of a quantizer. ΣΔ modulators are oversampling analog-to-digital converters, meaning that the sampling rate of the signal is increased beyond the Nyquist rate. As will be shown below, feedback pushes the noise away from the signal spectrum to the point where a low-resolution quantizer is enough for accurate digitization. The amount of oversampling is defined by the oversampling ratio (OSR)

$$OSR = \frac{f_S}{2f_B} \tag{32}$$

where f_S is the sampling frequency and f_B is the signal bandwidth. The digital output of the ΣΔ modulator corresponds to a pulse-density modulated version of the input signal.

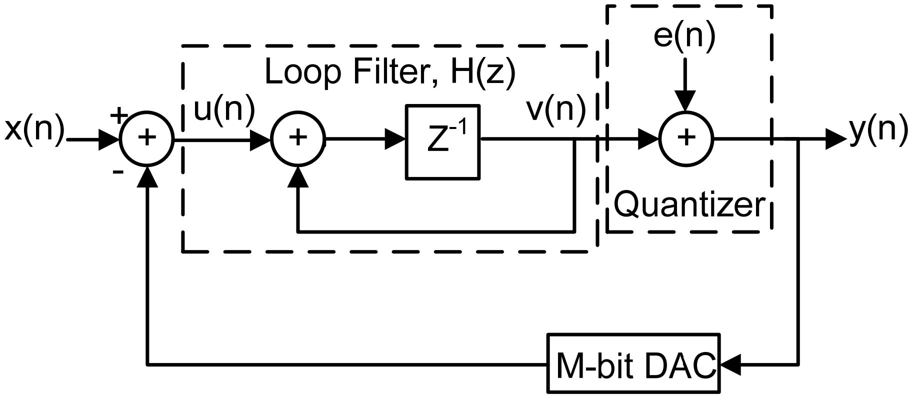


Fig. 23: First order ΣΔ modulator architecture.

Referring back to Fig. 23, the time domain difference equation mapping the input signal to the output signal is

$$y(n) = x(n-1) + [e(n) - e(n-1)] \tag{33}$$

where x(n) denotes the input, y(n) the output, and e(n) the quantization noise of the quantizer. The z-transform representation of (33) can easily be verified to be

$$Y(z) = z^{-1}X(z) + (1 - z^{-1})E(z) \tag{34}$$

Note that Y(z) is composed of two transfer functions where the first one, z^{-1}, corresponds to the signal transfer function (STF) while the second, (1-z^{-1}), is the noise transfer function (NTF). It can be noted that the STF has a unity magnitude response while the magnitude response of the NTF starts at zero and increases exponentially to a value of two as it approaches $f_S/2$. This effect of pushing the quantization noise out of the bandwidth of interest is referred to as noise shaping and is a very powerful feature of ΣΔ modulators. A typical signal and noise spectrum is shown in Fig. 24 when a sine wave is encoded.

High-order modulators can be implemented by utilizing more error history. For example, a second-order modulator can be implemented by subtracting the previous two samples of the error (weighted) signal to the current one. Generally, increasing the ΣΔ modulator order $N_{\Sigma\Delta}$ would increase the signal-to-noise ratio (SNR); however, more care with regards to the stability of the modulator need to be taken. There are other techniques to improve the SNR, such as increasing the OSR, number of output levels and etc. Note that the output is purely digital when it only has two levels. The effects of the aforementioned SNR improvement methods [17] are illustrated in Fig. 25.

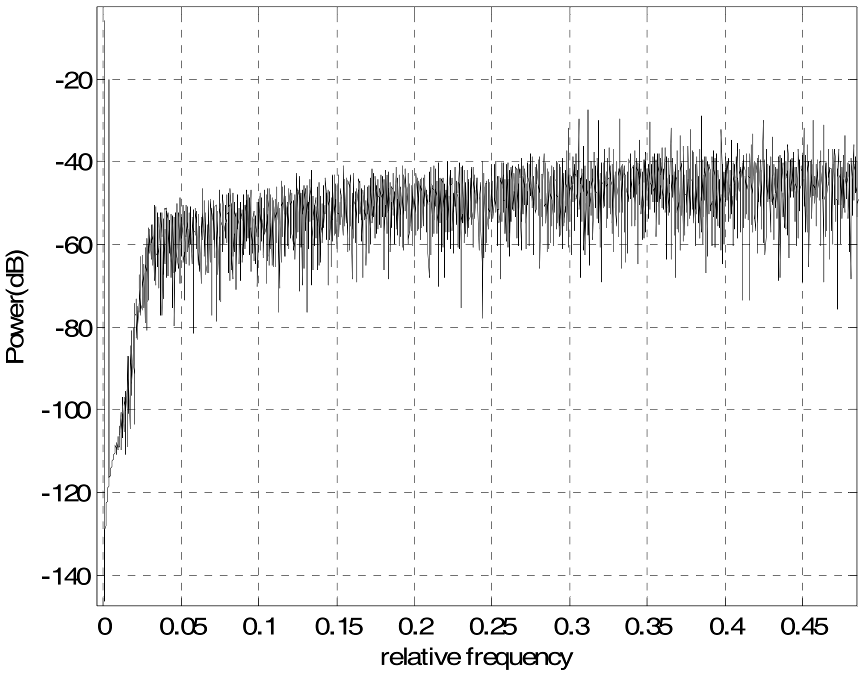


Fig. 24: Typical frequency spectrum of ΣΔ bits when a sine wave is encoded.

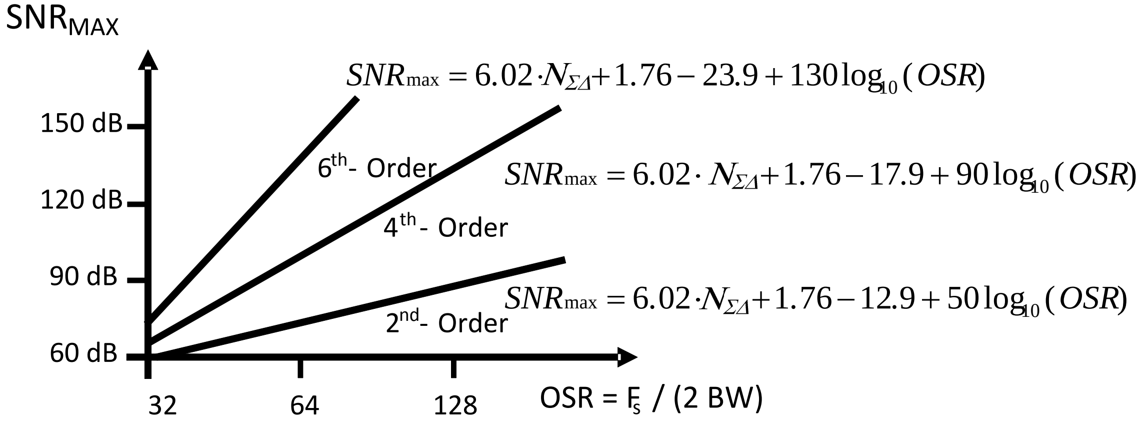


Fig. 25: SNR improvement methods and their effects.

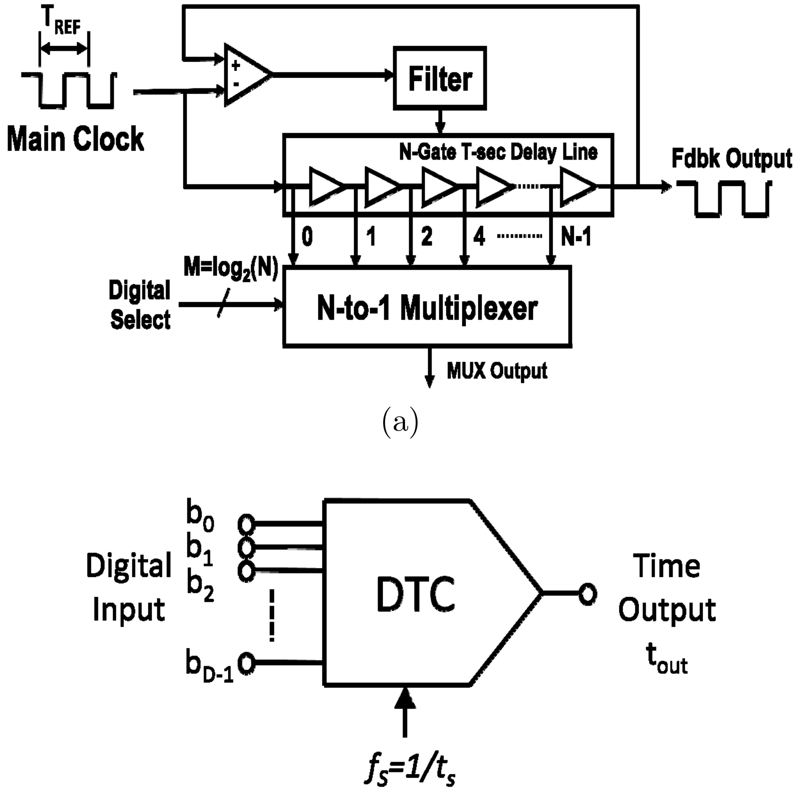
When recovering the encoded high-resolution analog information, it is straightforward to use an analog filter, as such information is present at low frequencies within the modulator bandwidth f_B and the quantization noise is mostly outside of the modulator bandwidth. This analog low-pass filter should have at least one more order than the modulator such that the slope of the rising quantization noise will be compensated by that of the filter's attenuation. If the above condition is met, a filter of the same bandwidth as the modulator will suffice. When the encoded signal has a lower bandwidth than the modulator, for example, in the case of the applications described in this work, a filter of a smaller bandwidth can be used as well. One more condition imposed on the filter is that, since most of the quantization noise is around $f_s/2$ [18], it thus needs to have high attenuation in the high frequency region.

3.3. Digital-to-Phase Mapping

In this section, we will show how a digital input can be converted to a phase modulated signal through a digital-to-time conversion process. The digital-to-phase mapping is typically done through the use of a digital-to-time converter (DTC), as introduced in [10]. The mapping process between the input bits and output time shift with respect to a reference clock is represented by

$$t_{out} = t_{ref}\left(b_0 + b_1 2^1 + b_2 2^2 + \cdots + b_{N-1} 2^{D-1}\right) + t_{os} \tag{35}$$

A general way of implementing a DTC is through the use of a delay-locked loop (DLL) and a multi-bit multiplexer (MUX). Referring to Fig. 26(a), the DLL is used to generate different delayed versions of a clock signal and the MUX takes in the digital input word and chooses the corresponding delayed clock. The symbol is presented in Fig. 26(b). For example, a 3-bit digital-to-phase mapping process with a digital input word being the sequence 4-1-2-3 would give the DTC output as shown in Fig. 27.



(a)

(b)

Fig. 26: (a) Multi-bit DTC hardware implementation consisting of a DLL and a MUX. (b) Symbol of the DTC in (a).

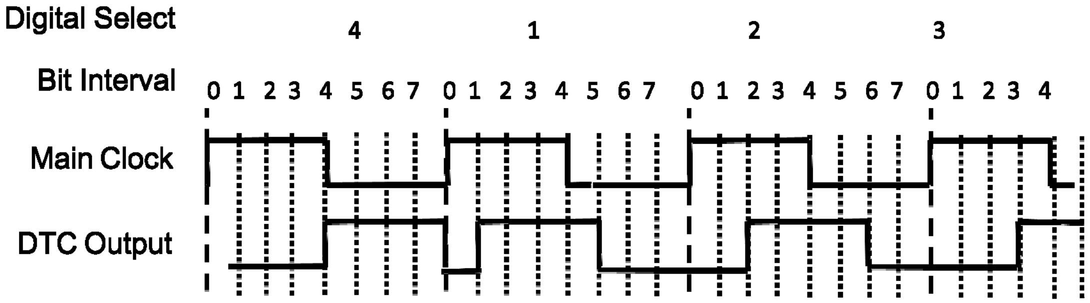


Fig. 27: Output of 3-bit DTC to a 4-1-2-3 input word.

A convenient way to understand and implement the DTC mapping process introduced in [23] is to look at the output sequence in terms of the input sequence. In the case of a 1-bit DTC that maps ‘0’ and ‘1’ in the digital domain to 0 radians and π/2 radians, respectively, every ‘0’ value in the digital domain is mapped to the bit sequence ‘1100’ and every ‘1’ is mapped to the sequence ‘0110’. In the case when a rising edge-triggered phase frequency detector (PFD) is used for the PLL, the duty cycle of the DTC output bit sequence need not to be 50%. As a result, an equivalent output pair of “1000” and “0100” can be used, as illustrated in Table 1.

As an aside, the choice of the bit sequence used in the DTC must also take into account the behavior of the PFD. As in the situation described here, the PFD is positive-edge triggered, hence all output codes must contain a well defined edge transition for every input bit. For example, as seen in Fig. 28(a), for the digital input sequence of “01”, the corresponding output code pair “1100” and “0110” would be acceptable as it contains two rising edges, whereas the output code pair “1100” and “0011” would not work, as “1100” will not be captured by the PFD when following “0011” since a rising edge is not present as seen in Fig. 28(b).

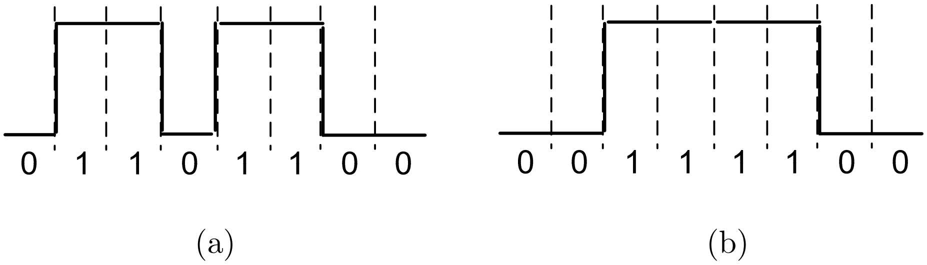


Fig. 28: Illustrating cases where the input code to the PLL (a) can be captured by an edge triggered PFD (b) cannot be captured by an edge triggered PFD.

Table 1: Illustrating the phase encoding process for a 1-bit DTC having a 90 degrees phase encoding range, for the equivalent cases of a 50% duty cycle output and a 25% duty cycle output

DTC Input	DTC Output (50% duty cycle)	DTC Output (25% duty cycle)
0	1100	1000
1	0110	0100

3.4. Phase Spectrums

Now that digital-to-time conversion has been presented, we can look at how everything fits together. Every ΣΔ modulated bit is mapped to a corresponding discrete phase through the DTC. The maximum value of the sigma-delta modulated signal in the amplitude domain, $\Sigma\Delta_{\mathrm{MAX}}$, is mapped to the maximum phase shift ϕ_{MAX}; likewise, the minimum value of the sigma-delta modulated signal in the amplitude domain, $\Sigma\Delta_{\mathrm{MIN}}$, is mapped to the minimum phase shift ϕ_{MIN}, without loss of generality if it is encoded using a single or multi-bit conversion. Assuming 'I' represent the unit used in the digital domain, the digital-to-phase mapping coefficient is defined as

$$\alpha_{\phi} = \frac{\phi_{MAX} - \phi_{MIN}}{\sum\Delta_{MAX} - \sum\Delta_{MIN}} \quad [\frac{rad}{I}] \tag{36}$$

This equation defining α_{ϕ} can also be seen as taking the full-scale range of the DTC over the full-scale range of the sigma-delta converter. Since 2π radians represent a complete period T of a clock signal, we can convert α_{ϕ} to time as

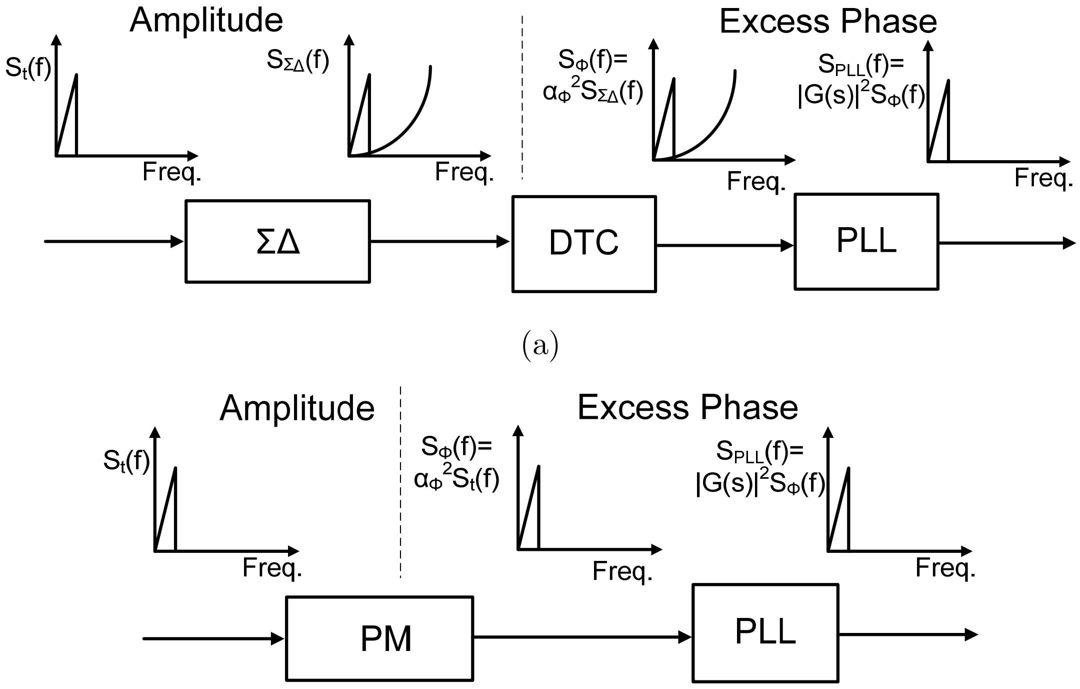
$$\alpha_{\phi t} = a_{\phi}\frac{T}{2\pi} \quad [\frac{s}{I}] \tag{37}$$

The spectrum of DTC output signal can then be written in terms of the ΣΔ PSD output, according to [23] as

$$S_{\phi}(f) = \left\{ \begin{matrix} \left(\alpha_{\phi}\sqrt{S_{\Sigma\Delta}(f)} + \phi_{OS}\right)^{2}, & f = 0 \\ {\alpha_{\phi}}^{2} S_{\Sigma\Delta}(f), & f \neq 0 \end{matrix} \right\} \tag{38}$$

Finally, the DTC spectrum is filtered in phase by the PLL with a transfer function G(s), and therefore the PLL output spectrum *$S_{PLL}(f)$* can then be written as

$$S_{PLL}(f) = |G(s)|^{2} S_{\phi}(f) \tag{39}$$



(a)

(b)

Fig. 29: (a) Illustrating the PSD of signals at each block for this testing scheme. (b) Showing the traditional phase modulation scheme and the equivalence to our method.

Fig. 29(a) illustrates the PSD at the output of the sigma-delta modulator, the DTC and the PLL. A test signal with a power spectrum $S_t(f)$ is first ΣΔ modulated and is then mapped to the excess phase through the DTC with an amplitude-to-phase mapping factor α_ϕ. The resulting signal is then applied to the PLL, which will filter out the quantization noise produced by the ΣΔ encoding process. Assuming that all the out-of-band quantization noise is filtered out and that the in-band noise is negligible, this procedure is equivalent to a direct phase modulation procedure, where the test signal is phase modulated with the same amplitude-to-phase mapping factor α_ϕ and applied directly to the PLL, as shown in Fig. 29(b).

3.5. Test Stimuli Generation and PLL Output Capture

As mentioned earlier, both the ΣΔ modulator and the DTC can be implemented in software. A short sequence of ΣΔ bits of length N_b can be generated and repeated as to approximate the output of an infinite-duration ΣΔ oscillator when driven by a periodic signal. From a DSP-based testing perspective, this signal needs to be coherent, as explained in [19]. This is ensured by respecting the coherency rule given by

$$f_{out} = \frac{M}{N_b} f_S, \quad M = 0,1,2,...,N/2 \tag{40}$$

where f_{out} is one possible frequency component of the test signal that is to complete M cycles at a sampling frequency of f_s. M and N_b need to be relatively-prime (no common factors) for a maximum number of unique samples. In addition, if the bits are not coherent, a discontinuity is created when bits are repeated. These bits are then mapped to the excess phase of a clock signal using the DTC and applied to the PLL. One can also notice that if the bit-stream is not coherent, every time when they repeat, the discontinuity will cause the PLL to undergo a transient response trying to regain steady state. This is not desired since PLLs usually have a tight bandwidth and consequently require a long time to regain steady state.

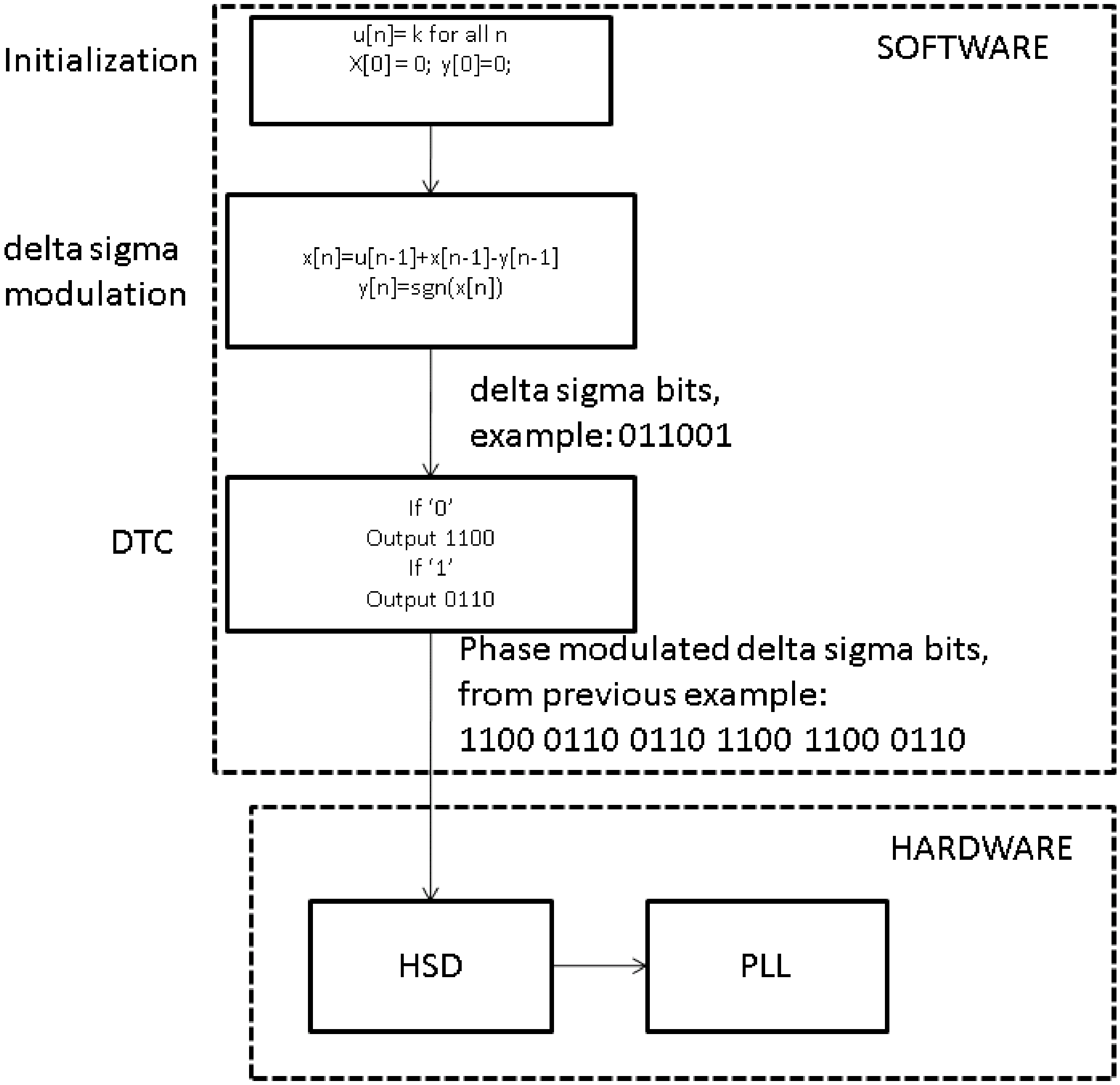


Fig. 30: Flow chart for the testing procedure.

The flow chart of the actual implementation using Matlab software and the HSD digital channels of the ATE is shown in Fig. 30. Here u[n] is the sampled version of the high resolution test signal. The sigma-delta modulation and the DTC mapping process occurs in software and the resulting bits are loaded to the HSD unit of the ATE to test the PLL. Note that this flow chart only shows a first order delta sigma modulation process. This is only for illustration purposes and in practice any order can be used when applicable.

To ensure an accurate PLL characterization, a reference clock signal that runs at the same frequency as the PLL carrier frequency needs to be introduced and the excess phase is defined with respect to it. The excess phase signal in the PLL output can be measured by using a simple TDC consisting of a counter triggered by the rising edge of this reference clock and counts the time difference to the next rising edge of the PLL output. This capture process is illustrated in Fig. 31 and the testing setup is illustrated in Fig. 32.

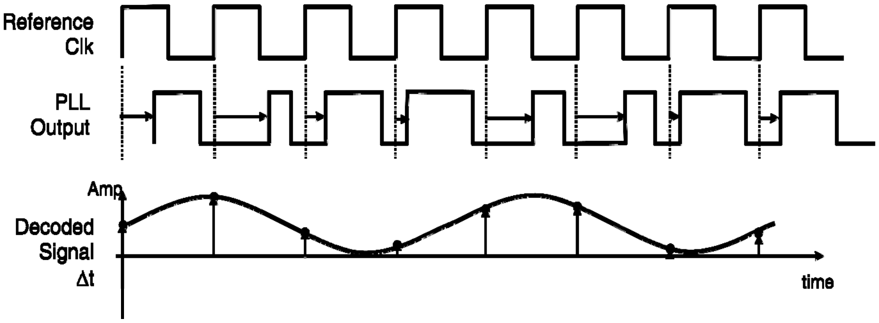


Fig. 31: Illustrating how a PM signal is extracted with respect to a reference.

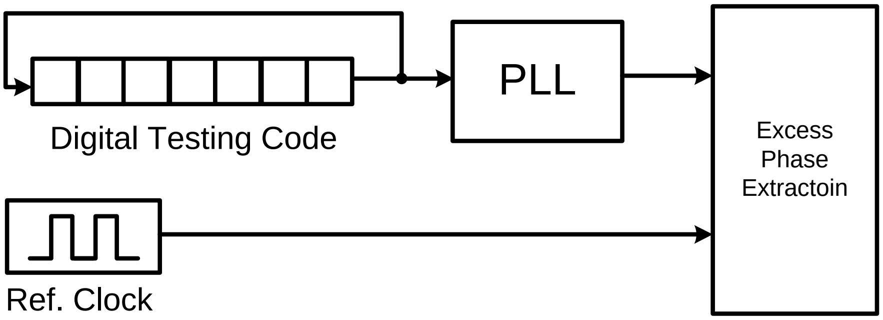


Fig. 32: Illustrating the PLL testing setup with the digital input.

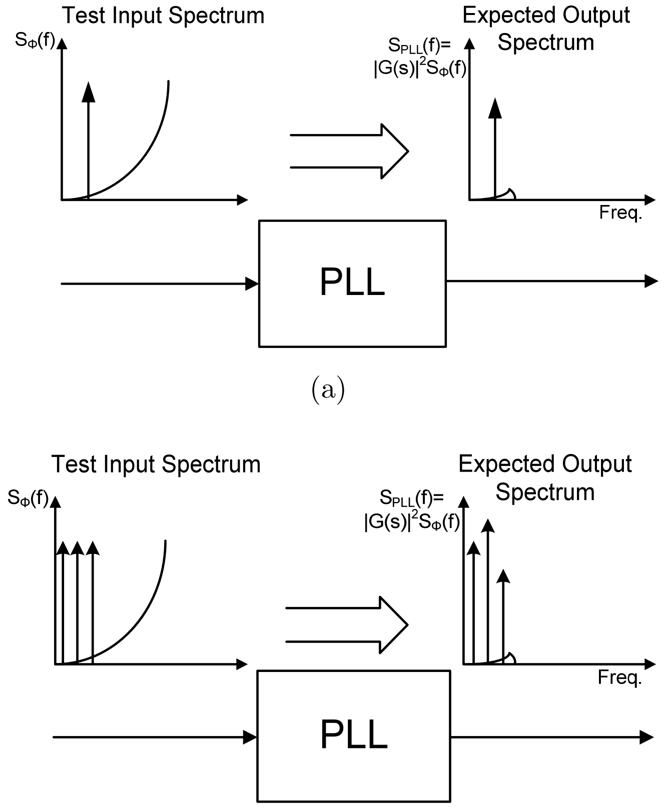
3.6. Transfer Function Measurement

In this section, PLL jitter transfer function and large signal transfer characteristic test methodologies are described, using the coding scheme mentioned in the previous sections. We start by introducing the single-tone test and expose its limitations. A more efficient multi-tone test is then described which will show that the entire transfer function can be measured with the same effort as a single tone test. Finally, the large-signal transfer characteristic test is explained.

3.6.1. Single Tone Test

In a single-tone test, a coherent sine wave is ΣΔ-encoded in the excess phase of a clock signal and applied to the PLL. Its response is captured in the excess phase of the PLL output, which is expected to be a sine wave running at the same frequency with its amplitude and phase altered by the PLL frequency response. This can be measured accurately without much interference from noise and distortion by performing a Fast Fourier Transform (FFT). This process is illustrated in Fig. 33(a) when a single-tone is applied to the PLL and its expected output is captured and analyzed. Going through a set of coherent frequencies in the input signal will give rise to an output vs. frequency plot and with enough frequency resolution, the PLL closed-loop transfer function can be determined accurately.

This approach, although very effective, has important limitations such as test running time. For example, during the frequency sweeping, each measurement must wait for the circuit to settle into steady-state before the result is valid, which is very time consuming. This problem becomes more pronounced when testing PLL circuits, as they usually have a tight bandwidth and consequently require a long time to settle.



(a)

(b)

Fig. 33: The PLL test signal input spectrum and its expected output spectrum for a) a single-tone test b) a multi-tone test.

3.6.2. Multi Tone Test

A more efficient approach is to use a multi-tone signal, which consists of a sum of coherent sinusoids. One can construct a multi-tone signal whose frequency components cover the entire frequency range of interest. Then, the PLL test input stimulus is constructed by $\Sigma\Delta$ encoding this signal in the excess phase of the reference clock signal. In addition to coherency, the tones need to be chosen such that no harmonics or low-order inter-modulation (IM) distortion collides with another test frequency. In addition, the peak-to-RMS ratio needs to be kept low to maximize dynamic range. The testing procedure is illustrated in Fig. 33(b), when 3 sinusoids are added together to generate a 3-tone signal. The output is expected to have the same frequency components with their amplitudes and phases altered by the PLL transfer function. Note that one single test can be sufficient to capture the PLL transfer function thus saving tremendous amount of test time and resources. If the test needs 10 sinusoids at different frequency to capture its frequency response, the multi-tone test will be 10 times faster than the single-tone test. Since only 1 set of bits is needed, comparing to the 10 sets in the case of single tone, the multi-tone test is also 10 times more efficient in terms of computing resources.

3.7. Static Operation Test

For investigating the static operation of the PLL, one can ΣΔ-encode a DC value in the excess phase of the signal applied to the PLL input and observe its corresponding output. At the PLL output, the excess phase information presents itself as a clock signal with a constant delay to the reference signal. By sweeping the DC code and measuring the corresponding delay, one can obtain a voltage-to-time transfer characteristic curve. This is analogous to measuring the large-signal transfer characteristics of an amplifier. Many properties, such as linearity, can be observed. During this test, the rule of coherency can be relaxed as there is no strong periodic frequency component present (since a DC signal is coherent with any frequency).

3.8. Summary

In this chapter, the DFT solution for testing a PLL is presented. We started by giving a review of sigma-delta modulation and digital-to-phase mapping techniques first introduced in [10]. The digital-to-phase mapping was then improved in [23] and used in this thesis. The theoretical background for this testing technique in the form of phase spectrum was subsequently explained. Next, how the test stimuli was generated and PLL output phase could be extracted was discussed. Finally, the details of single-tone test, multi-tone test and large-signal voltage-to-time transfer function test were described.

Chapter 4

Simulation/Experimental Verification of PLL Operation

A 6th order PLL with a 100 kHz half-power bandwidth was constructed according to the schematic of Fig. 20, interfaced to a Teradyne FLEX mixed-signal tester and its behavior was measured. The Teradyne tester was used to generate the digital bit stream to stimulate the PLL and an Agilent oscilloscope was used to analyze the PLL output data. The opamps used to realize the active filter were of the type OPA355 from Texas Instruments. The experimental board used to interface the PLL to the tester is shown in Fig. 34 and it is mounted on the tester in Fig. 35. The PFD and VCO were constructed from the components found on the TLC2932 IC chip, also from Texas Instruments. The PFD which uses a tri-state buffer has a gain K_P equal to 0.34 V/rad and the VCO has a gain K_{VCO} equal to 76.6 Mrad/V/s. The VCO output voltage range is from 0 to 5 V. The active filter was designed to have the transfer function in (31) of Chapter 2. The detailed design files (schematic, layout) are presented in the Appendix.

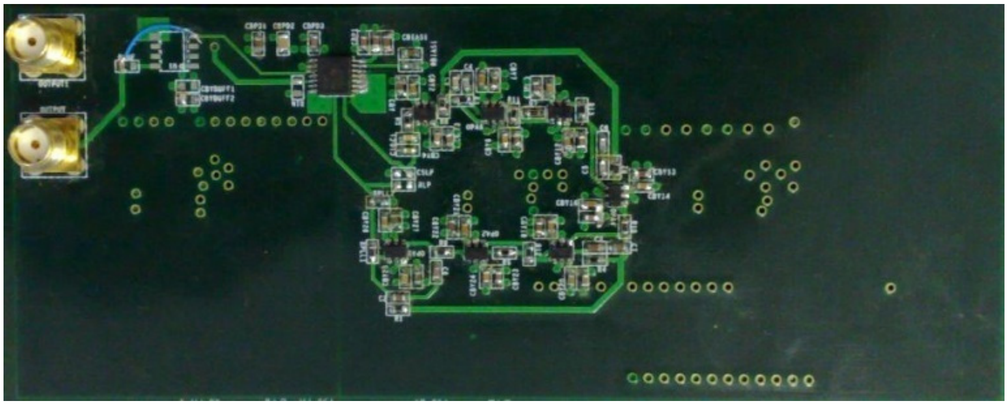


Fig. 34: Photograph of the 6th order prototype PLL.

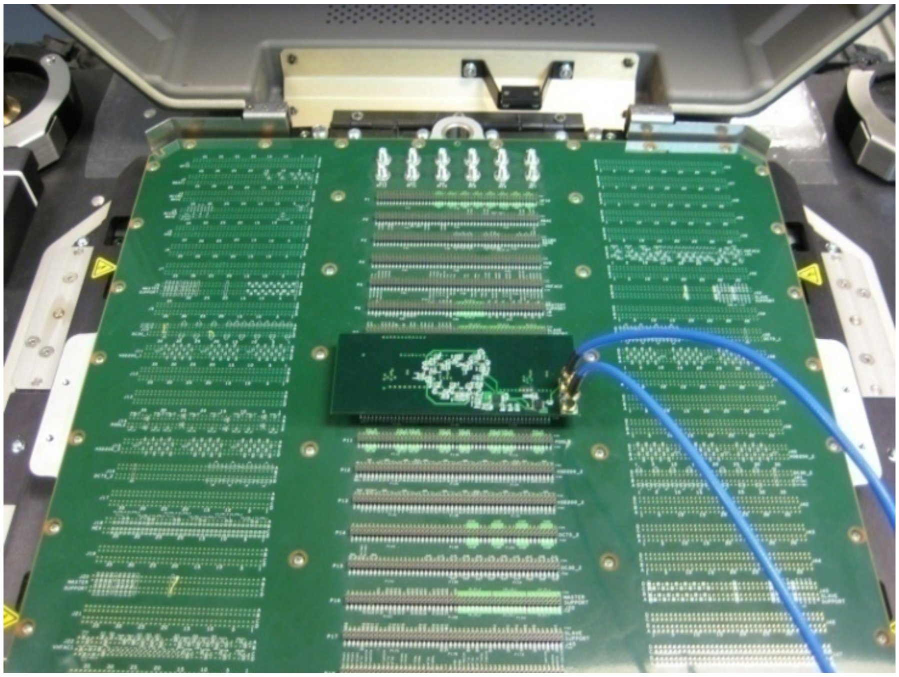


Fig. 35: Experimental setup showing the constructed PLL board on Teradyne Flex ATE.

For comparison purposes, simulations were also performed using MATLAB/Simulink. The implementation of ΣΔ modulator of arbitrary order is shown in Fig. 36 where the state variables are calculated using the tool 'DSMOD' described in [20]. The resulting bits are then exported to the Matlab workspace. In addition, the high-order PLL and the excess phase capture are implemented as shown in Fig. 37(a). The phase capture is performed by taking the time difference between the rising edges of the PLL output and those of a reference clock running at the same frequency. The time of rising edge is captured by the triggered subsystem, a unique type of system of Matlab, which is shown in Fig. 37(b), acting like a D flip flop latching the simulation time when a rising edge is present at its input. Constants of multiples of 2π radians are added to avoid issues of phase wrapping, and the final phase value is brought within the range of 0 to 2π by the "modulus" function.

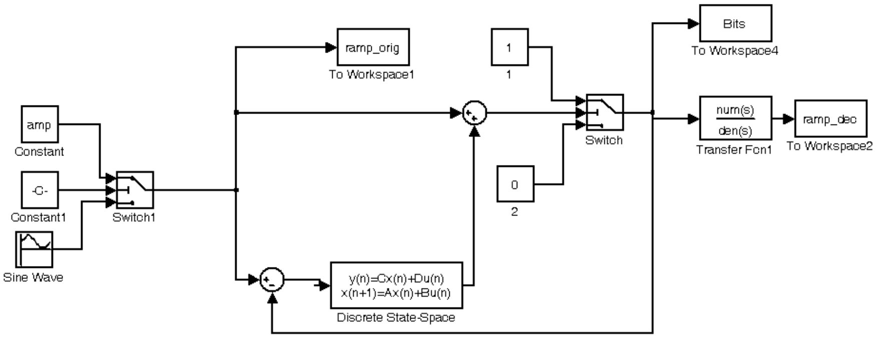
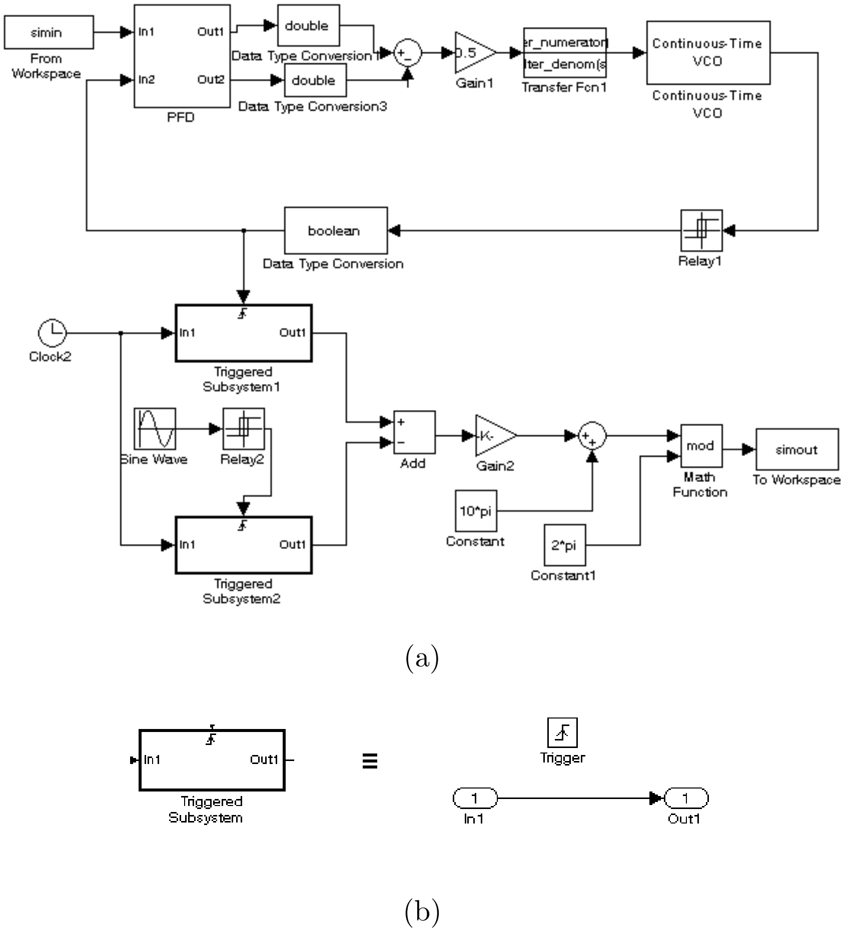


Fig. 36: A sigma-delta modulator of arbitrary order implemented in Simulink.



(a)

(b)

Fig. 37: Simulink implementation of the (a) high-order PLL and the phase extraction process which outputs the time difference between the PLL output and the reference clock in radians. (b) triggered subsystem shown in (a).

4.1. PLL Test

The bits used for testing were first generated in Simulink using a 5th-order ΣΔ modulator having a Butterworth response and an OSR of 64 together with a Matlab script that performs the DTC operation of mapping the digital sequence to the corresponding phase modulated signal. The ΣΔ modulator output ranges from 0 to 1 resulting in sigma delta parameters $\Sigma\Delta_{MIN} = 0$ and $\Sigma\Delta_{MAX} = 1$. For both the single-tone and multi-tone test, the amplitude-to-phase mapping converts every logical '0' value to the bit sequence '1000' and every '1' to the sequence '0010' for a phase range of π radians. The amplitude-to-phase mapping factor, denoted previously by α_ϕ is therefore equal to π rad/I. For the test of voltage-to-time large-signal transfer characteristics, a different coding scheme is used, where every '0' value in the digital domain is mapped to the bit sequence '1100' and every '1' is mapped to the sequence '0110'; hence, resulting in a mapping factor α_ϕ of $\pi/2$ rad/I.

The bits were generated in a cyclic fashion with a single-bit duration of 15 ns and then applied to the input of the PLL. As every bit sequence consists of 4 bits corresponding to a carrier period of 60 ns, the frequency of this phase-modulated signal is 16.67 MHz. A reference clock was also synthesized at this same rate. In simulation, this is achieved by introducing a clock signal running at 16.67 MHz, whereas during the experiment this needs to be done more carefully to ensure synchronization. Specifically, the code "1100" is stored in the source memory of the FLEX tester, repeated every 4 bits and generated in parallel with the phase-modulated bit sequence. This was achieved using the high-speed digital source unit of the FLEX tester. A block diagram of the experimental setup is shown in Fig. 38. The excess phase extraction block is implemented using the Agilent 54830D oscilloscope. Here the input to the PLL and its corresponding output are fed to the scope's channel 1 and 2, respectively, and are compared to the reference clock through the external triggering of the scope. While we choose to use the scope for extracting the excess phase, we could have just as easily fed these signals back into the digital capture unit of the FLEX tester and performed these same measurement. This was simply a matter of convenience, since the Agilent scope had Matlab running in it. In fact, a routine to extract the excess phase from the sampled data running in Matlab facilitated the analysis.

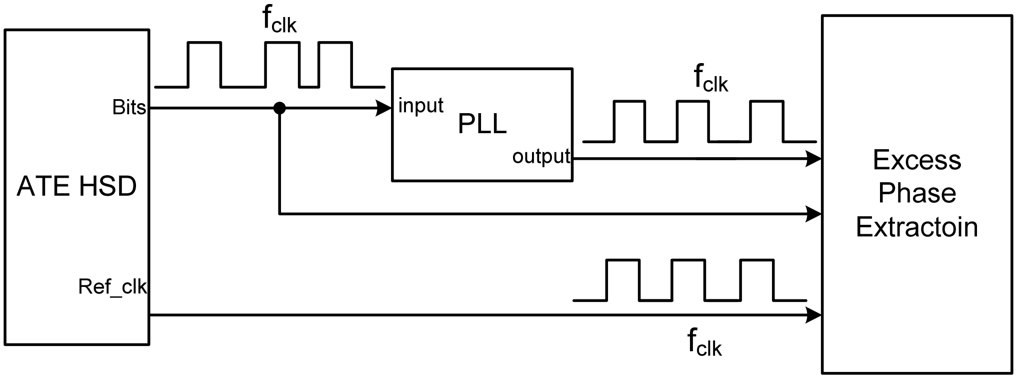


Fig. 38: Experimental setup for PLL testing in which both the excess phase of the PLL input and output are extracted with respect to a reference clock.

4.1.1. Single-tone Test

Here, a coherent sine wave with an amplitude of 0.15 on a 0 to 1 scale in the digital domain (4.5 ns in the phase domain) with a frequency of 93.59 kHz is encoded into a 81920 bit long cyclic bitstream to act as the test stimulus. Such a long bitstream is forced upon us due to the tester's lack of flexibility with regards to the looping of the bit sequence to ensure coherence. While any frequency within the ΣΔ software-encoded modulator bandwidth could have been used, in this experiment we placed the tone very near the passband edge of the PLL where the PLL is most sensitive. To verify the signal presented to the PLL, we captured the corresponding input bit stream to the PLL and measured its phase with respect to the reference signal. The PSD corresponding to this signal obtain through an FFT analysis is shown in Fig. 39. Also shown in Fig. 39 is the PSD of the input signal resulting from simulation. As is evident, the two signals are in close agreement. The level corresponding to each tone is almost identical (7.055 dB versus 7.029 dB), where 0 dB in all the spectral plots of this work corresponds to a sinusoidal phase-modulated signal with an amplitude of 2 ns. Also, evident from Fig. 39 is the high-pass nature of the PSD corresponding to the noise-shaping operation of the ΣΔ modulator. In both cases, they are very similar.

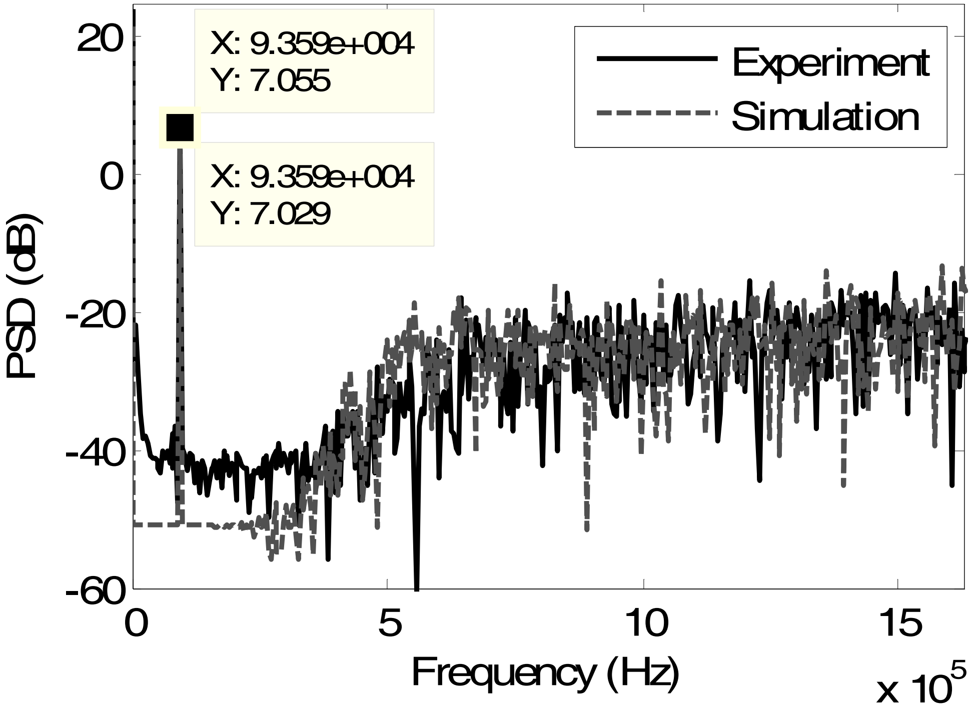
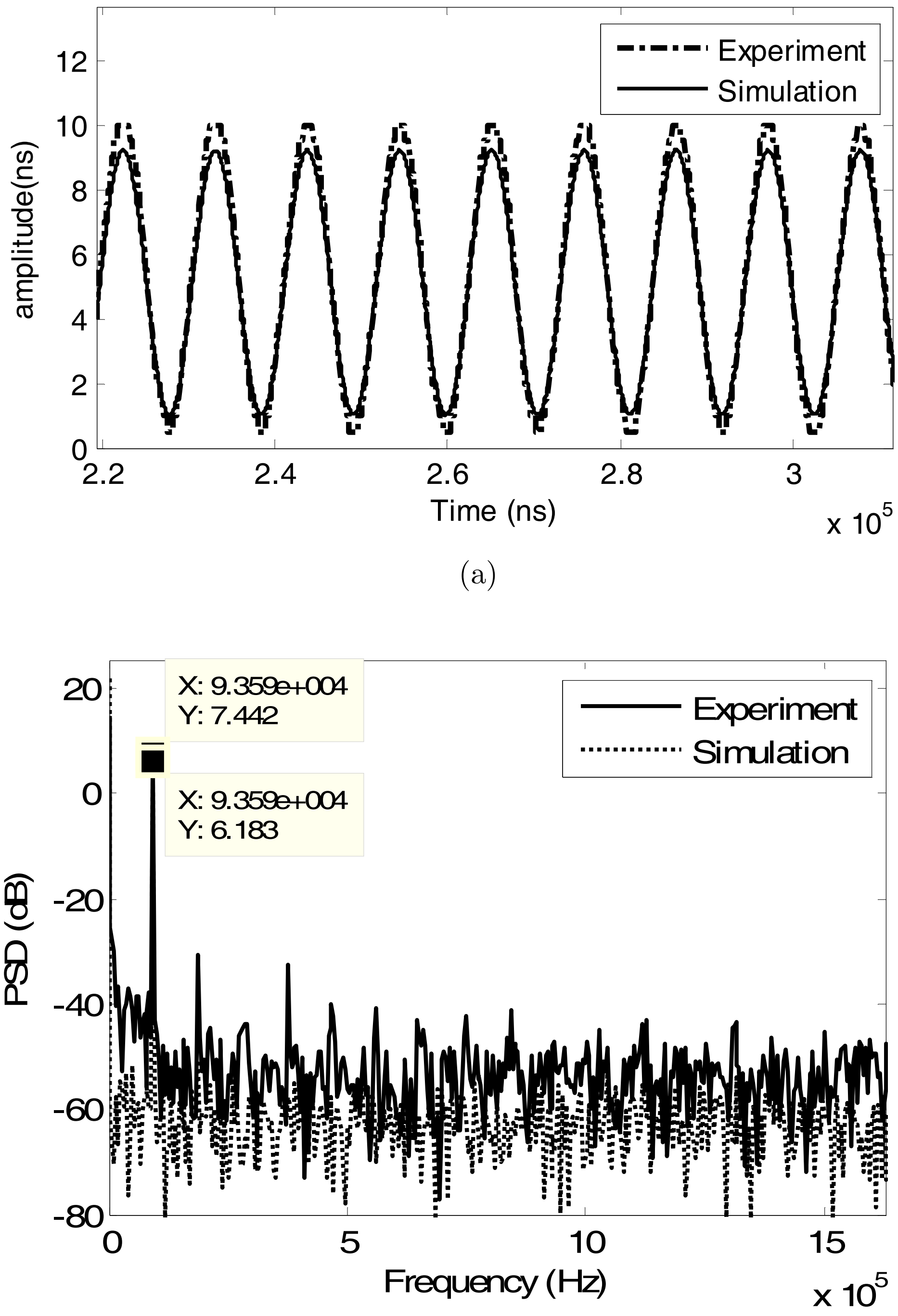


Fig. 39: Spectrum of the PLL input excess phase signal both simulated and experimentally measured for the single tone test.

The PLL output is then measured and a time domain plot of the output excess phase is obtained and plotted in Fig. 40(a). These results are also compared to those generated by simulation. Note that the simulation output had to be shifted by about 10.35 ns vertically in order to adjust for the constant delay induced from the clock skew in the experimental setup. In comparison, the two sine waves have similar amplitude; simulation reveals an amplitude of about 4.1 ns, and through direct measurement has an amplitude of 4.7 ns. The corresponding PSD of these two signals is shown in Fig. 40(b). Here the level of the experimentally measured tone is 6.183 dB whereas the simulated tone level is 7.442 dB. This is a difference of 1.26 dB corresponding to a 14% relative error. This error is attributed to the fact that the PLL model in simulation is not exactly the same as that used in the prototype. While one would never expect them to be exactly the same, these results suggest they are quite similar and the simulation results are a reasonable predictor of PLL frequency response performance. We also notice that the PSD is limited to a noise floor of about -60 dB. This can be attributed to the quantization noise associated with the time resolution of the scope. We will have more to say about this in a separate context in the next paragraph. In addition, harmonics not seen in simulation was observed in the experimental data. This is due to the nonlinear large-signal behavior of the PLL. We will investigate this further in section 4.1.3.

Next, the frequency of the encoded sine wave is swept over a range of 4069 Hz to 231.9 kHz while the input amplitude is kept constant at a level of 4.5 ns. Fig. 41 illustrates how the input amplitude as seen at the input to the PLL varies with frequency. As can be observed, the input level is relatively constant and consistent with the value encoded, with a maximum relative error of 0.05 dB or 0.6%. These results further illustrate the flexibility of the bit-stream approach as a general test stimulus.



(a)

(b)

Fig. 40: A comparison of the experimental data with simulation corresponding to PLL output in (a) time domain (b) frequency domain.

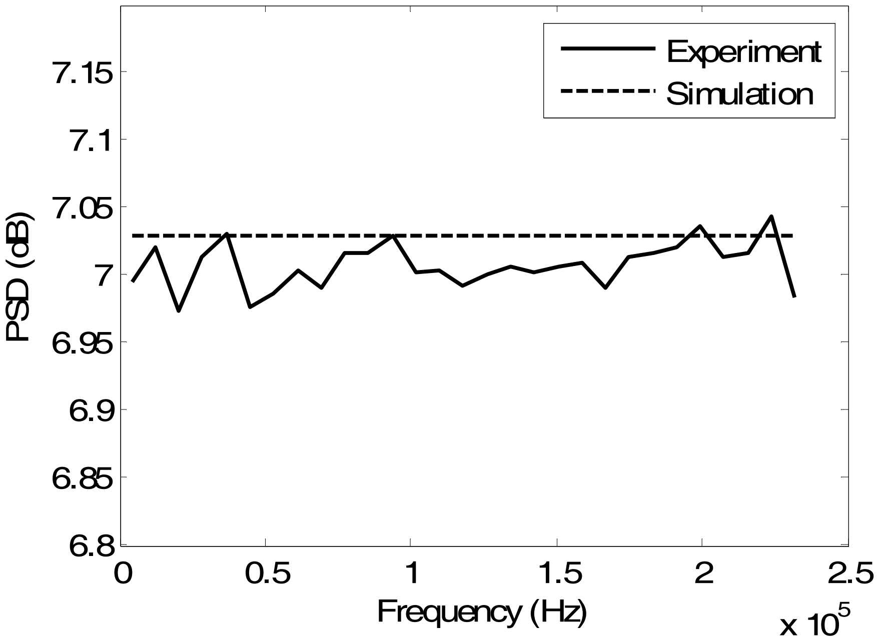


Fig. 41: Showing the input power level being consistent throughout the frequency range of interest for single tone test.

In conjunction with the frequency swept phase-modulated sine wave input stimulus, the output signal from the PLL was captured and the corresponding input-output phase magnitude frequency plot was obtained as shown in Fig. 42. Also shown in the plot is the expected or simulated phase transfer function response. It is reasonable to conclude the swept single tone test using the bitstream approach is in good agreement with that predicted by theory. This is not too surprising given that we measured the input signal characteristics and found that they were very tonal within the PLL bandwidth. We also noticed that the gain measurements could not be measured reliably 30 dB below the input level of 4.5 ns. This meant that phase signals with amplitudes less than 142 ps were lost. This is reasonable as the scope has a stated time resolution of 500 ps.

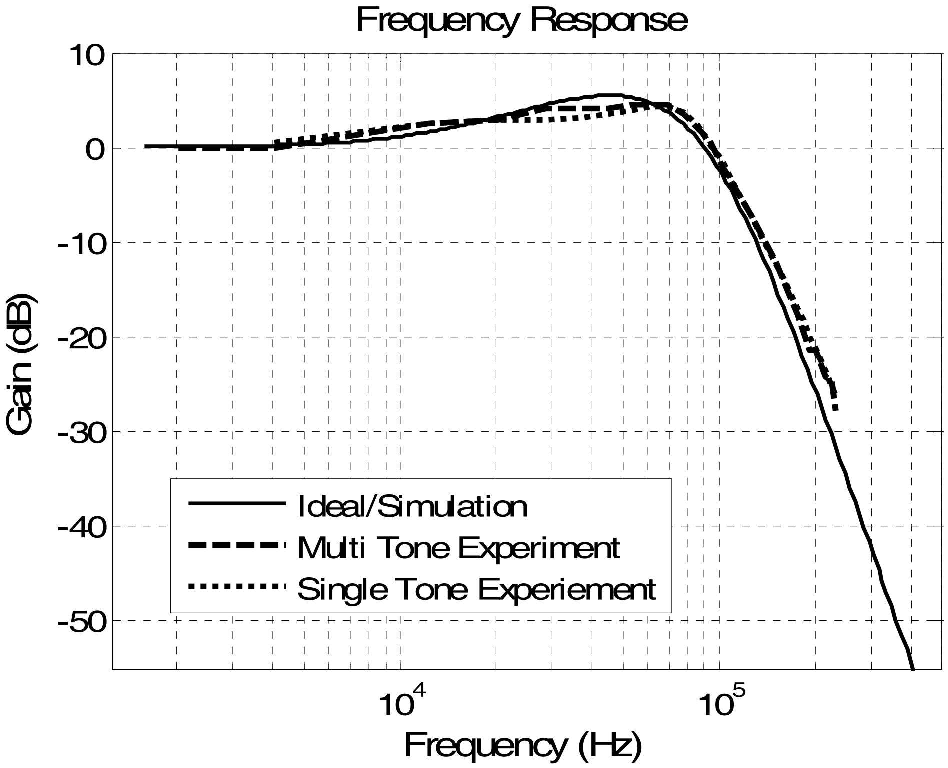


Fig. 42: A comparison between the experimentally measured frequency response and simulation using single-tone and multi-tone tests.

4.1.2. Multi-tone Test

In this subsection, 20 coherent sinusoids with equal amplitude of 0.037 in the digital domain have been added together and encoded to generate an 81920 bit cyclic bit-sequence that is phase modulated with 20 tones of 1.11 ns amplitude each. The frequencies of these tones were chosen such that there are no frequency collisions with any of their distortion products, as well as the phases of these tones were randomized to reduce the peak-to-RMS ratio[21].

The PLL input and output phase signals are captured experimentally in the same fashion as the single-tone test and their PSD are plotted in Fig. 43 and Fig. 44, respectively. One can clearly observe from Fig. 43 that the input tones are of equal amplitude across a frequency span of 0 to 250 kHz having a level of -5.10 dB corresponding to an amplitude of 1.11 ns. Also, the noise shaping action of the ΣΔ modulation is also present. In the plot of Fig. 44, one can see that the PLL frequency response to the multi-tone signal. Here the amplitude of the individual tones are altered by the PLL behavior and the out-of-band quantization noise is filtered out. Note that significant amount of energy is present on some non-testing frequencies within the PLL bandwidth. This is normal with multi-tone testing and is simply ignored in any post analysis [21]. Matlab/Simulink simulations results are also consistent with these results, as seen from Fig. 45 and Fig. 46.

The phase transfer function measurement associated with the PLL can now be easily obtained and the resulting curve is superimposed on the previous plot of expected behavior in Fig. 42. We can see that the multi-tone results are very similar to the single-tone test and ideal/simulation results. We can also conclude that the multi-tone test is faster to perform than the single tone test. In simulation, it takes 0.31 ms to complete the multi tone test, whereas the single tone tests take 6.2 ms. This stems from the fact that the tester resources are identical in both cases, requiring 81920 bits of cyclic tester memory, but the multi-tone test provides the entire phase transfer characteristic in a single test rather than after multiple iterations of the single-tone test and its associated settling time per frequency setup.

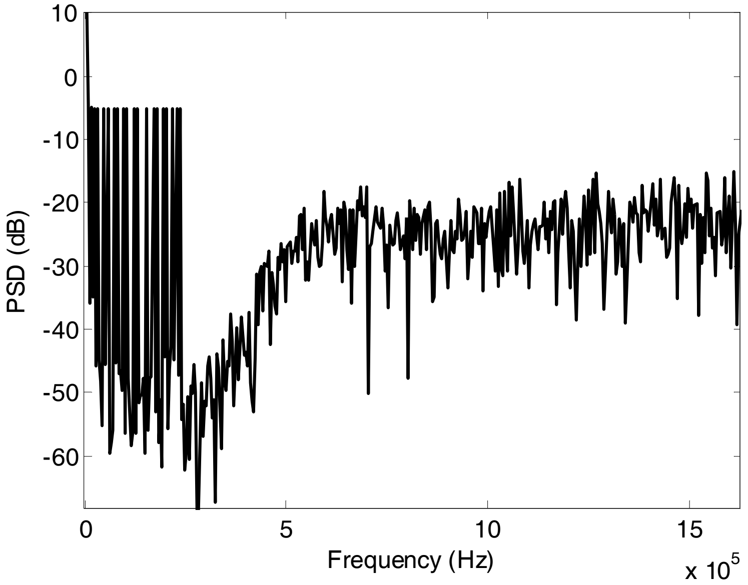


Fig. 43: Experimentally measured PSD of the PLL input excess phase signal for the multi-tone test.

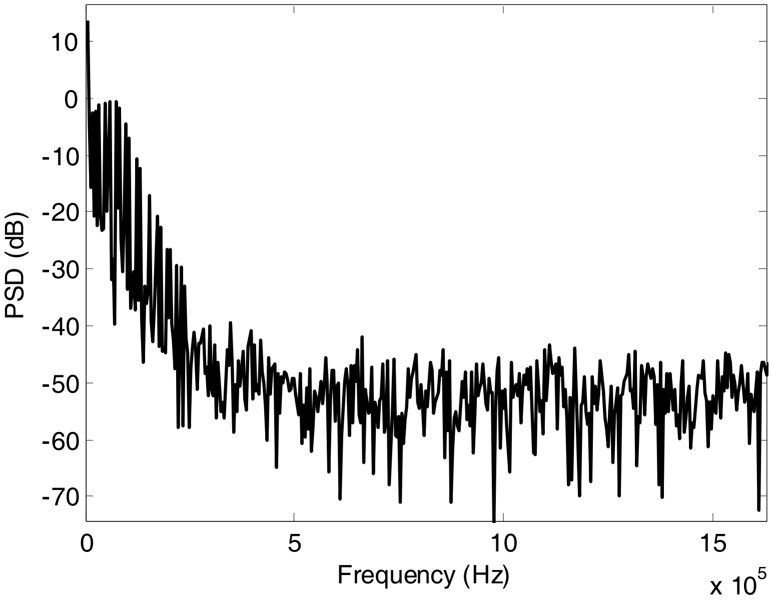


Fig. 44: Experimentally measured PSD plot of the PLL output excess phase signal for the multi-tone test.

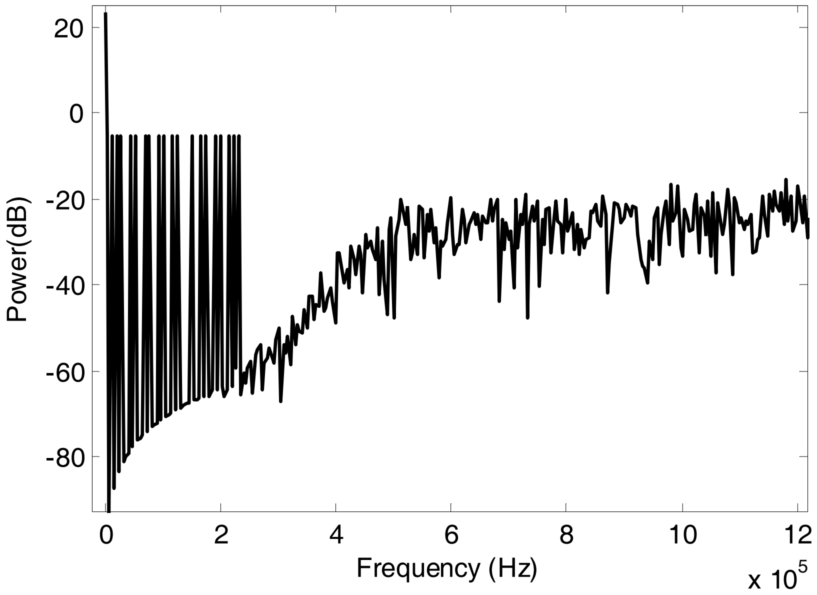


Fig. 45: PSD plot of the PLL input excess phase signal for the multi-tone test measured in simulation.

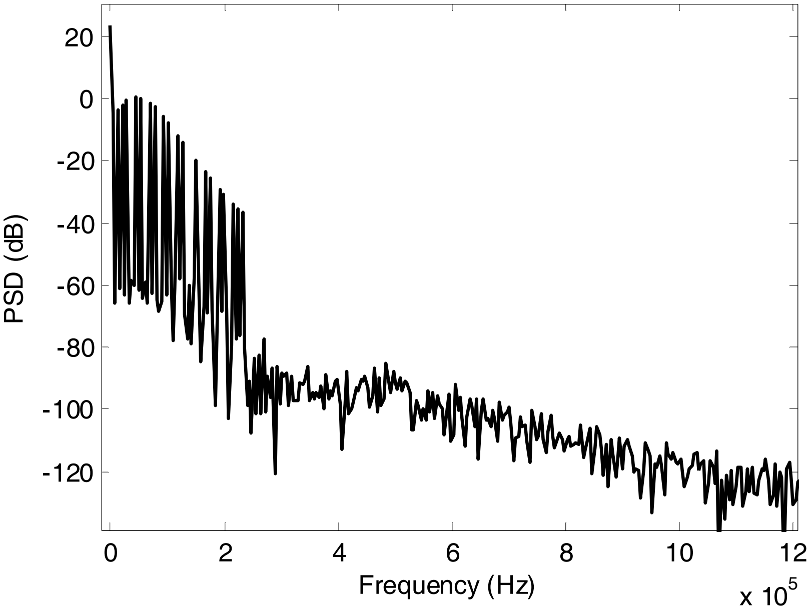


Fig. 46: PSD plot of the PLL output excess phase for the multi-tone test measured in simulation.

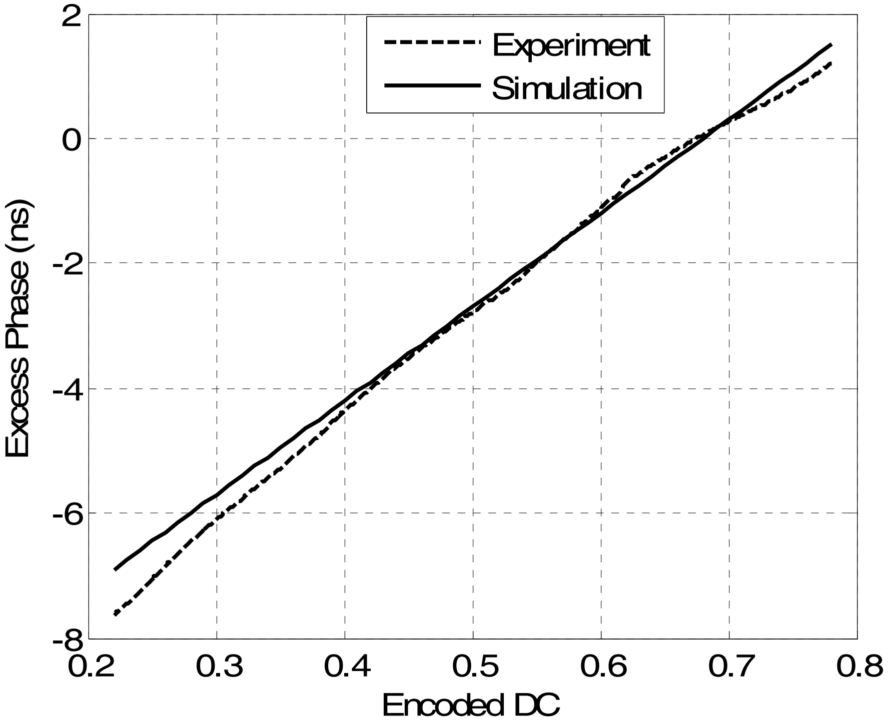


Fig. 47: Transfer characteristic curve showing output phase versus encoded DC code in both simulation and experiment.

4.1.3. Large-Signal Transfer Characteristics

In order to obtain the large-signal transfer characteristic curve, DC signals are encoded and swept over the stable region of the ΣΔ modulator used, which is about from 0.22 to 0.78 on a 0 to 1 scale. The coding scheme explained earlier with α_ϕ equal to $\pi/2$ rad/I is exercised here. The reason a smaller phase range is chosen is to use finer time resolution. In addition, it also shows the versatility of the testing encoding principle as users can zoom-in to a desired observation range using a finer phase resolution. The voltage-to-time large-signal transfer characteristic curve is plotted in Fig. 47. As can be observed, the curve is monotonically increasing with a linear trend as expected from simulation.

However, the small deviations from the straight line can be related to non-linear behaviors present in the PFD, the active filter and/or the VCO.

4.2. Summary

In this chapter, the PLL design methodology and the DFT solution were verified both in simulation and experiment. A 6th order PLL was built and measured on a Teradyne Flex ATE. In addition, simulations were performed before performing the experiment. The DFT method is verified by measuring the known input and comparing with the results capture using this method. Single-tone and multi-tone measurements are then performed using this technique, and the results agree with the theories present in Chapter 2 very well. The large signal voltage-to-time transfer function is also measured and is found to be in close agreement with the theory too.

Chapter 5

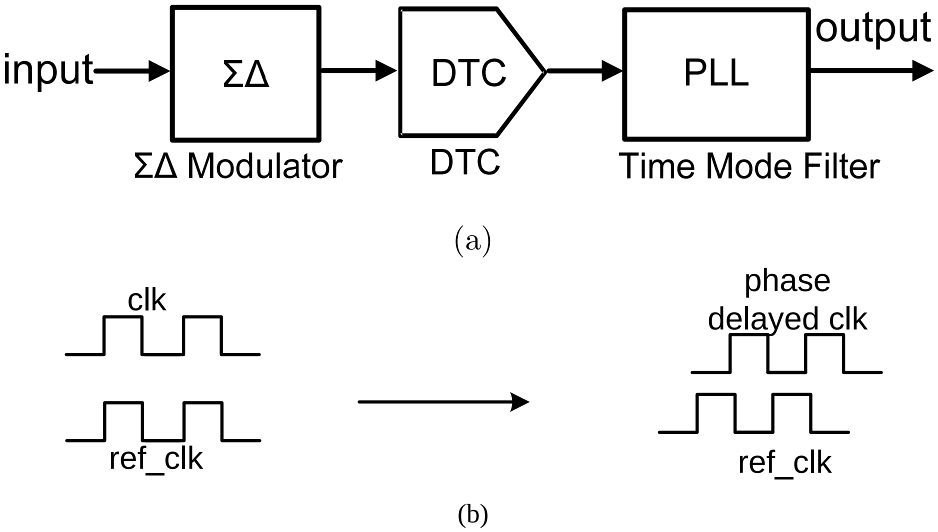
Several Applications of High-Order PLLs

In this chapter we will demonstrate several new applications of high-order PLL. This will include the design of a phase delay generator for clock alignment and jitter measurement. Simulations will first be carried out to illustrate the basic idea then experiments will be performed using the Teradyne Flex tester and the LeCroy SDA6000 oscilloscope.

5.1. Phase Generation

5.1.1. Application Description

Fig. 48(a) shows a block diagram of our proposed delay generation circuit, which is very similar to the PLL transfer function test setup. The desired phase delay is first encoded using lowpass ΣΔ modulation and it is then converted to the excess phase of a clock signal using a DTC. The unwanted quantization noise is thus shaped to high frequencies (with respect to the phase modulated signal) and is filtered out by the lowpass transfer characteristic of the PLL. Both the ΣΔ modulator and the PLL are high-order, which leads to infinitesimal quantization noise at the output, such that a clean clock signal with an arbitrary phase can be generated quite accurately. Fig. 48(b) shows a clock delayed from its original position with respect to reference clock. This clock can be thought as the clock that drives the sigma-delta modulator in a hardware implementation.



(a)

(b)

Fig. 48: (a) Proposed phase generator structure. (b) Expected input-output behavior.

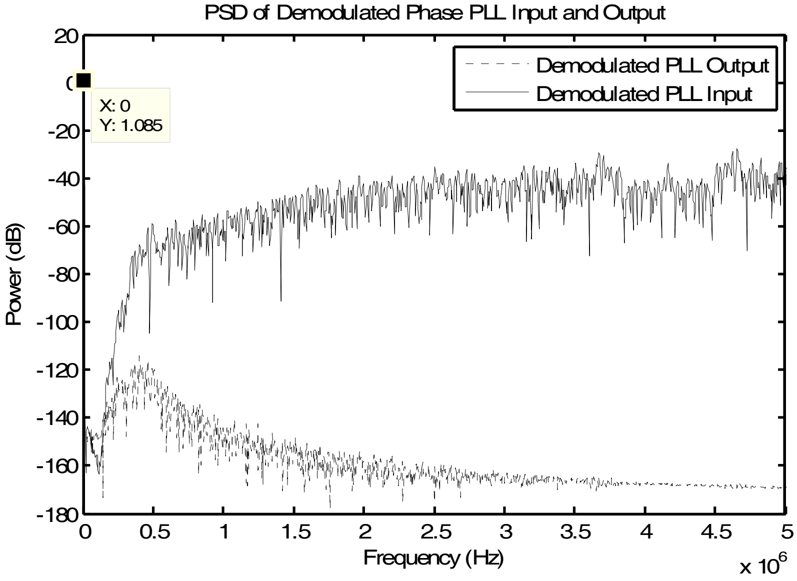


Fig. 49: PSD of the demodulated 6th order PLL input and output.

5.1.2. Simulation

The proposed phase-delay encoding scheme was first implemented in Matlab/Simulink and verified. DC encoded bit-streams are generated in the same way as the testing bit-streams for the large signal transfer characteristics test described in section 4.1.3. Here a 5th order modulator with OSR 64 was used together with a 1-bit DTC having a digital-to-phase mapping factor α_ϕ equal to $\pi/2$ rad/I. The resulting bit sequence is then applied to the input of the 6th order PLL model and the output delay is observed with respect to the reference clock.

Fig. 49 shows the PSD of the spectrums at the input and output of the PLL, when a DC value of 0.5 was encoded. A Blackman-Harris window is applied to avoid frequency leakage of the quantization noise. As can be observed, there is a tone in Bin 0 (i.e. DC) and we see the high-frequency quantization noise associated with the input and the filtering action of the PLL. The PLL rejects most of the high frequency quantization noise leading to a DC output phase signal (i.e., a constant phase shift with respect to the reference).

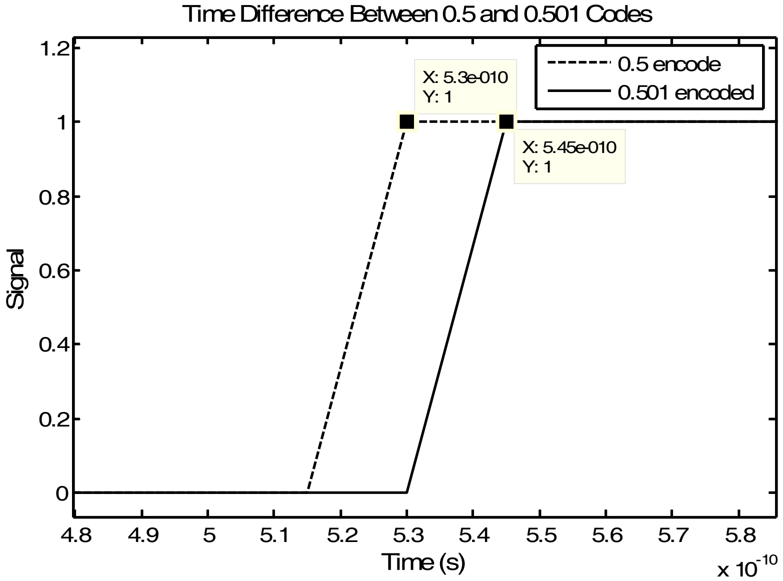


Fig. 50: Two sample outputs corresponding to 0.5 and 0.501 DC encoded values showing a phase difference of 15 ps.

Fig. 50 shows two sample outputs, where DC values of 0.5 and 0.501 are encoded. The time difference between these two outputs is calculated to be 15 ps

$((0.501-0.5)\times\alpha_{\phi}$, where α_{ϕ} is $\pi/2$ rad/I or $\alpha_{\phi t}$ of 15 ns/I for a period of 60 ns), which is consistent with the above simulation result.

Now, let us examine how various design parameters of the PLL affect the performance. As mentioned in Chapter 3, when recovering high-resolution information that is sigma-delta encoded, the order and bandwidth of the reconstruction filter, in our case, of the PLL, plays a vital role. In general, the smaller the PLL bandwidth, the more quantization noise can be filtered out and therefore the higher the performance. Higher order PLLs are also preferred, as they reject more noise out of band. In addition, higher order filter allows the use of higher order modulators, which will ultimately lead to high performance too.

Fig. 51 shows the simulation result of how time resolution varies with respect to PLL bandwidth, while all other parameters are unchanged (i.e, sigma-delta modulator order of 5, OSR 64, ΣΔ-DTC cascade gain α_{ϕ} equal $\pi/2$ rad/I, PLL order of 6 designed for Gaussian lowpass response). The resolution measure used is integral nonlinearity (INL), which is the maximum deviation of the measured result from the ideal curve. We can clearly see that the resolution gets better with tighter bandwidth.

Alternatively, instead of altering the filter bandwidth, one can also alter the order of the PLL and experience performance change. The PLL order is swept while other parameters of the delay generator are held constant (i.e, sigma-delta modulator OSR of 64, ΣΔ-DTC cascade gain α_{ϕ} equals $\pi/2$ rad/I, PLL bandwidth of 100 kHz designed to have a 6th-order Gaussian lowpass response). Note that the sigma delta modulator order is changed according to the rule described in section 3.2, as the low-order PLL cannot filter out the fast-rising high frequency quantization noise of high-order sigma-delta modulators. Fig. 52 shows the plot of resolution versus order. Not surprisingly, the higher the order, the better the resolution.

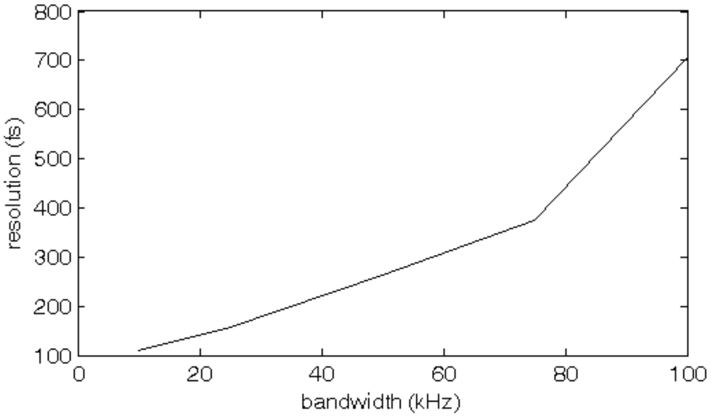


Fig. 51: Plot of time resolution vs. PLL bandwidth showing improved resolution with tightening bandwidth.

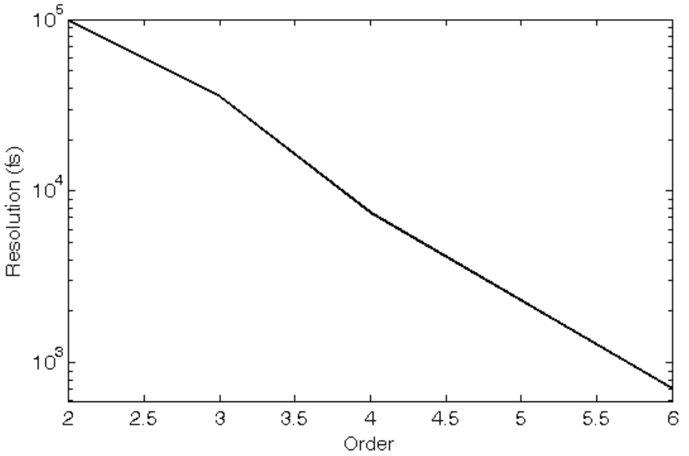


Fig. 52: Plot of time resolution vs. PLL order showing improved resolution with increasing order.

We can therefore conclude that high-order PLL design with a tight bandwidth is desired for the application of phase generation. This also allows the use of higher order sigma-delta modulator with a high OSR, which can also improve the phase delay performance.

5.1.3. Experiment

Phase delay generation experiments are performed in this section. The functional operation of all codes is first verified by sweeping the ΣΔ encoded DC value and the phase difference between the PLL output clock and the reference clock is observed. The voltage-to-time transfer characteristic curve is plotted in Fig. 54. This is the same experiment as the large signal transfer characteristic test performed in section 4.1.3. The bits chosen cover the stable region of the ΣΔ modulator, which is about from 0.22 to 0.78 using a very small DC encoded step of 0.001. Therefore, a total of 561 cases have been tested. As can be observed, the curve is monotonically increasing with a linear trend as expected. However, the small deviations from the straight line can be related to the non-linear behaviors present in the PFD, the active filter and/or the VCO. Here, the maximum absolute error compared to a best-fit line equals 380 ps. This deviation from a straight-line would correspond to a 4.4-bit linear digital-to-time conversion with an INL/DNL of less than 1 bit. However, since the curve is a monotonic function, knowing exactly the correspondence between the DC code and the phase, proper compensation can be performed. Indeed, the DC codes can be pre-distorted by a lookup table prior to applying them to the ΣΔ modulator, as depicted in Fig. 53. As shown in Fig. 54, with this compensation setup a 9-bit conversion, or a resolution of 15 ps, can be achieved with an INL/DNL of less than LSB. Note that a constant phase offset that is not present in simulation is expected since the length of the path that carries the output signal differs from that of the reference signal, which is the problem of clock skew discussed earlier. This offset is determined empirically and turns out to be -10.35 ns. A sample output is captured in Fig. 56(a), where a delay of 3 ns between the PLL output and the reference can be observed when a DC value of 0.49 is encoded.

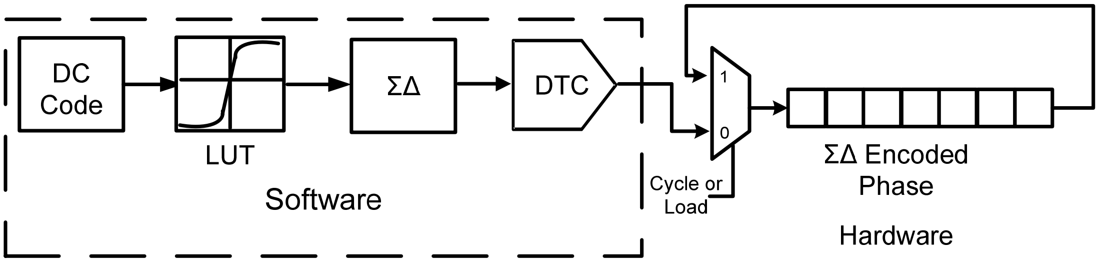


Fig. 53: Phase synthesis compensation setup where a lookup table pre-distorts the DC code in order to account for the non-linear behavior.

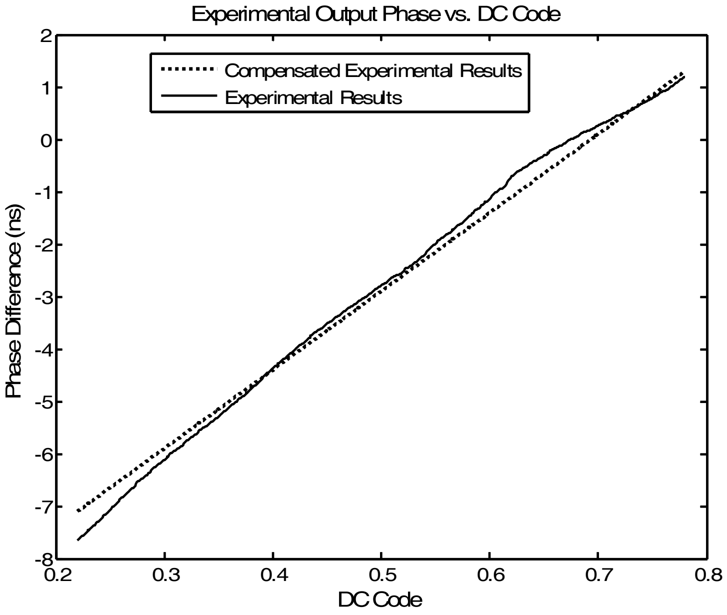


Fig. 54: Transfer characteristic curve of output phase versus encoded DC code.

5.2. Skew Cancelation

5.2.1. Application Description

One direct application of this device can be immediately identified: clock skew cancelation. When testing digital circuits, a clock signal needs to be generated from the source, for example, an ATE. Due to the spatial displacement from the source and the DUT, a phase difference is created from the desired clock.

If multiple DUTs driven by the same clock are being tested, the clock phase deviation produced due to different distance from the clock source may result in circuit malfunction. This problem becomes more pronounced as the circuits are running faster and faster. Although this problem has been addressed in most ATE testers to some extent, it is still problematic at the device-interface board (DIB) level.

Fig. 55 shows how our proposed technique can solve this problem. A clock that needs to be aligned, is first frequency up converted by 4, which can be done by a PLL with a feedback divider of 4. This clock will then drive the bits stored in a memory that is generated similarly as before. Then, the encoded phase is swept until aligned with the reference clock.

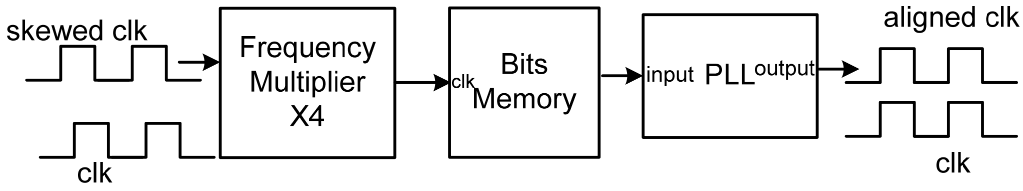
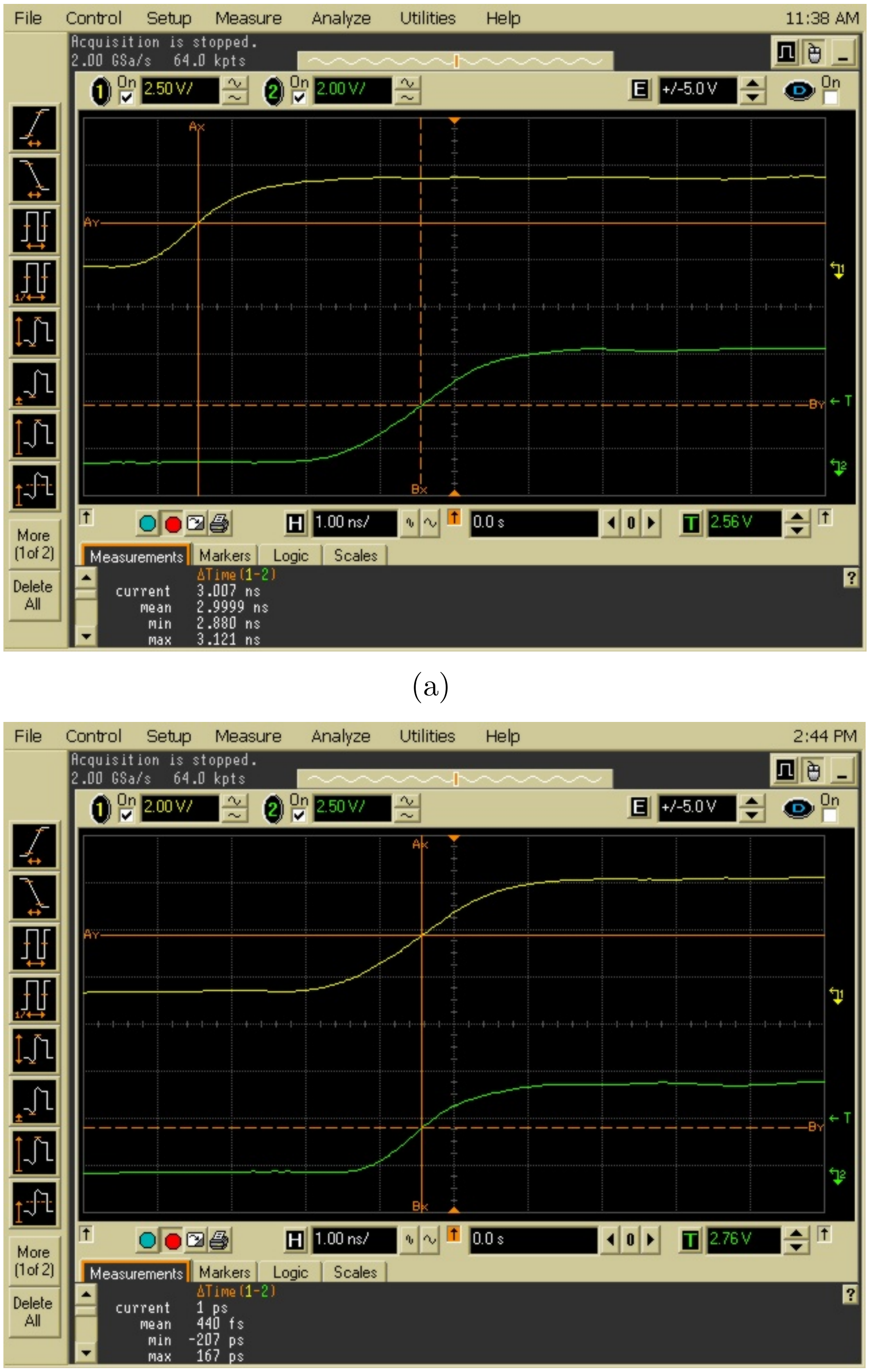


Fig. 55: Showing skew cancellation being achieved with this technique.

5.2.2. Experiment

The same setup as described in section 5.1.3 is used to perform this experiment. As described earlier in the previous section, the clock skew between the reference clock and the delivered clock is -10.35 ns due to setup delays. A calculated DC level of 0.69 (10.35 ns / $\alpha_{\phi t}$ where $\alpha_{\phi t}$ is 15 ns/I) is needed to bring the clocks to alignment. Bits are generated using this DC level and applied to the PLL and the resulting PLL output is shown in Fig. 56(b), from which a phase difference of 440 fs is observed between the two clock signals with a period of 60 ns each. This corresponds to a relative error of 0.000733% and therefore the clock skew of -10.35 ns has been reduced to near zero error.



```
Acquisition is stopped.
2.00 GSa/s   64.0 kpts
```

```
                ΔTime(1-2)
  current       3.007 ns
     mean       2.9999 ns
      min       2.880 ns
      max       3.121 ns
```

(a)

```
Acquisition is stopped.
2.00 GSa/s   64.0 kpts
```

```
                ΔTime(1-2)
  current       1 ps
     mean       440 fs
      min     -207 ps
      max       167 ps
```

(b)

Fig. 56: Oscilloscope screen capture showing: (a) programmed delay of 3 ns (b) clocks aligned.

5.3. Jitter Measurement

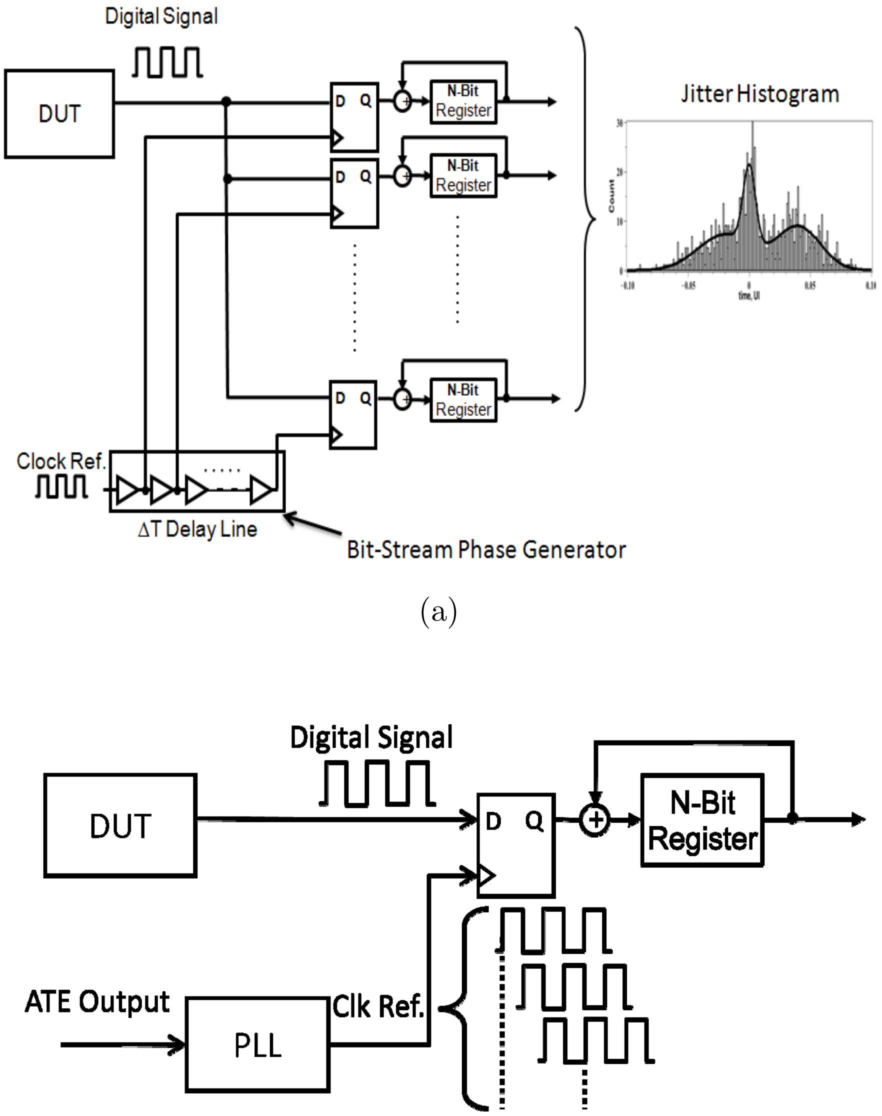
5.3.1. Application Description

Since the proposed system can generate an arbitrary delay quite accurately, another application of this circuit can be as a time sampler. As electronic circuits' speed increase, their measurements become more difficult, as test equipments are electronics themselves and consequently are required to operate at high speed. Traditional DSP-based testing techniques require the system output to repeat a few cycles to make an accurate measurement. This requires sampling much faster than the Nyquist rate. For high-speed systems seen today, it would require the time sampler on the tester to operate many times faster than the DUT, which may be running at very high frequency already. An alternative way is to use under-sampling techniques, which enables an accurate measurement using a sampling frequency less than the Nyquist rate.

One application utilizing undersampling is for jitter measurement. Referring to Fig. 57(a), a reference clock is passed through a delay line, and outputs different delayed versions of it to trigger a set of D-flip-flops. A counter is connected to each D-flip-flop to keep track of the number of times the DUT signal arrives at the D-flip-flop earlier than the trigger (DUT signal = 1 when triggering). For any delay encoded, the probability of DUT signal arriving earlier than the trigger can then be derived with another counter keeping track of the total number of such comparison instant. We can write it mathematically as

$$P(T_{DUT} < T_{trigger}) \tag{41}$$

By sweeping the position of $\mathrm{T}_{\mathrm{trigger}}$, we can obtain a cumulative distribution function (CDF) and mathematically derive the corresponding probability density function (PDF) by a differentiation operation. In this thesis, we express the PDF in terms of the histogram of the captured data set.



(a)

(b)

Fig. 57: Jitter measurement setup using various phase shifts, D-flip-flops and counters to generate a CDF of the clock jitter using: (a) parallel implementation, (b) series implementation.

An implementation that fits better with the experimental setup is outlined in Fig. 57(b). Here, a reference clock is first set to a given phase offset and a counter counts the number of times the signal under test arrives to the D-flip-flop before the reference clock for a specified duration; then, the reference clock's phase is changed to a new value and the cycle repeats. Going through a set of equidistant phase shifts would similarly give rise to a CDF and from it we derive the PDF; thus, its jitter metrics can be characterized very accurately. The equivalent sampling frequency would be the reciprocal of the phase step size.

5.3.2. Simulation

In the Matlab simulation, a clock with a jitter that is Gaussian distributed is measured using the technique proposed in the previous subsection with a phase delay step size of 20 ps. The resulting PDF is plotted in Fig. 58 and compared with the previous method of demodulating the phase using the setup shown previously in Fig. 37 and creating the histogram from the demodulated signal. As one can see, the distributions are very similar.

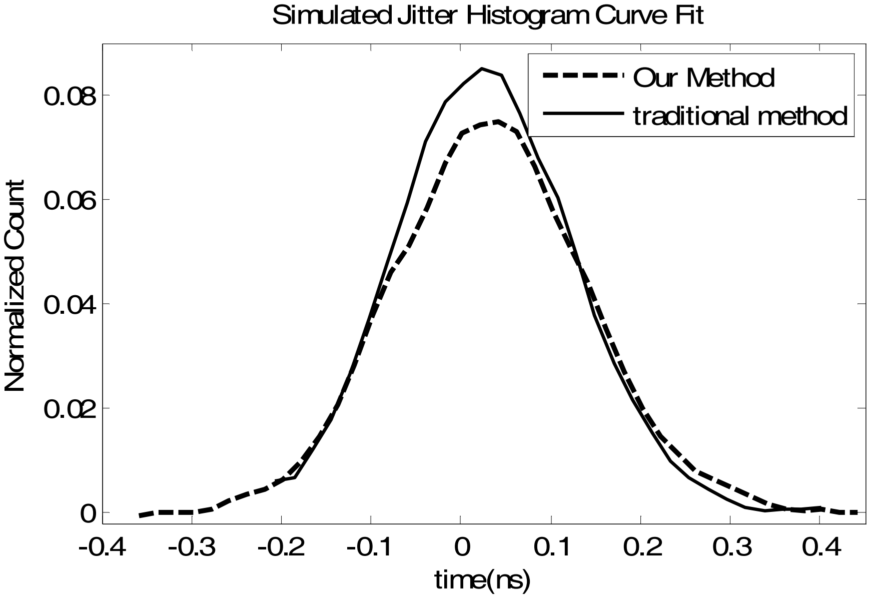


Fig. 58: Simulated jitter histogram generated using our method and the traditional method.

5.3.3. Experiment

The setup described earlier for jitter measurement has been implemented here. The phase is swept in steps of 20 ps (effective sampling rate of 50 Gs/s) and the number of times the reference clock was ahead of our encoded clock for a given time duration is counted, as shown in Fig. 57(b). Note that the clock signal for which the jitter is measured is actually the same one used to drive the bits with the encoded phase to the PLL. Consequently, an equivalent setup where 2 clocks coming out of the tester are applied to the LeCroy SDA6000 scope with one of them being used to trigger the scope while measuring the jitter of the second one as shown in Fig. 59. Fig. 60 shows the histogram distribution for both cases. Here we can see that both histograms have similar distributions. We exported the data back to Matlab to calculate the RMS jitter, and the proposed implementation shows 162 ps, which is comparable to the 195 ps obtained from the LeCroy scope. Therefore, we can conclude that our method can measure jitter performance similar to that given by the LeCroy scope.

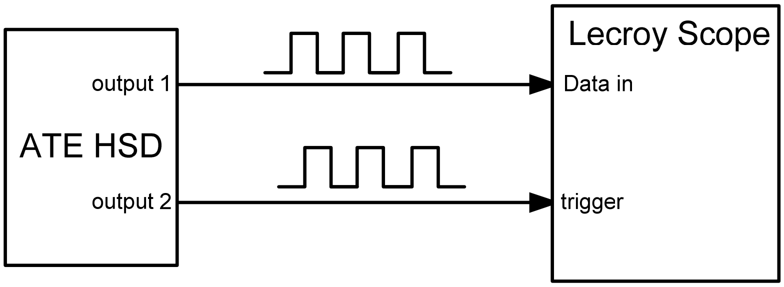


Fig. 59: Showing the jitter measurement setup using the LeCroy scope.

Other applications have also been demonstrated using the PLL designed utilizing different input coding schemes. For example, in [23], a frequency synthesis scheme is developed and a digital-to-frequency conversion resolution of 9-bit is achieved. In addition, it is also shown that, this PLL can also be used for jitter generation such as Gaussian jitter and sinusoidal jitter.

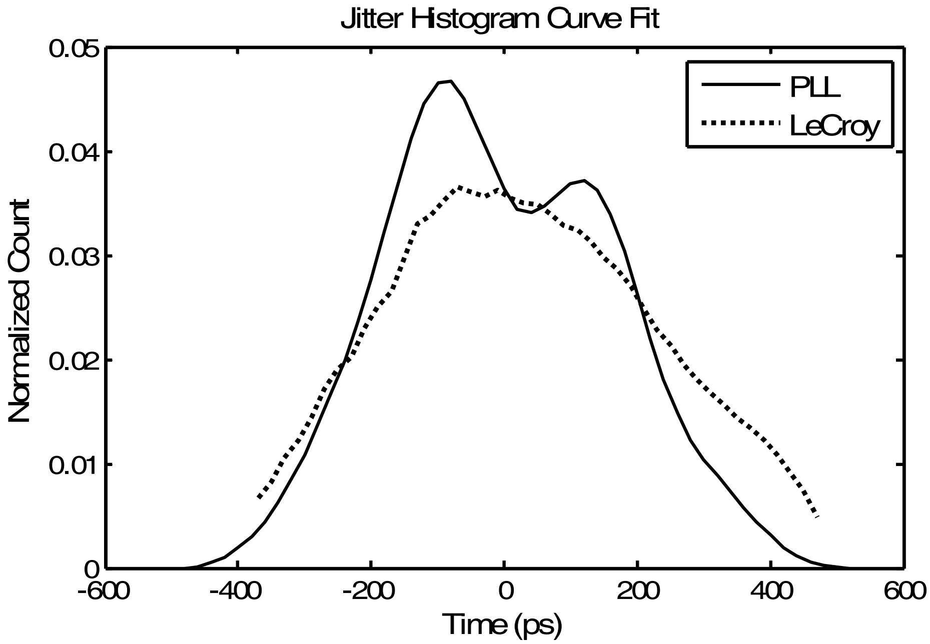


Fig. 60: Jitter histogram measured from our system and LeCroy.

5.4. Summary

In this chapter, a high-order PLL is used in the creation of an accurate and precise phase delay generation circuit. In turn, this phase delay circuit is used in the application of a skew cancelation technique on ATE tester boards and for extracting the statistics of a jittery clock signal. Matlab/Simulink simulations together with ATE experiments were performed as means of validation of the proposed techniques. Experimentally, a time resolution of 15 ps was achieved with the delay generation circuit, and clocks with a period of 60 ns were shown to be aligned within 1 ps of each other. In addition, jitter measurement was also performed using this phase delay generation circuit at an effective sampling rate of 50 Gs/s. The measured data was consistent with the result captured directly by a LeCroy scope.

Chapter 6
Conclusion and Future Work

6.1. Thesis Summary

In this work, a pole-zero placement algorithm for the systematic design of high-order phase-locked loops serving as anti-imaging and anti-aliasing filters for time-mode signal processing applications is presented. In addition, an all-digital DFT solution using either a single-tone or multi-tone test stimuli was proposed for validating the frequency characteristics of a PLL. Moreover, the large signal transfer characteristics of a PLL can also be measured using the same encoding scheme. The implementation of the test setup has three main components: a ΣΔ modulator, a DTC and the high-order PLL under test. Each building block has been investigated in detail. The first two components are implemented in software and the digital output is exported to a Teradyne FLEX ATE high-speed digital (HSD) unit and repeated cyclically and applied to the input of the PLL. A 6th order PLL interfacing with an ATE tester has been designed and fabricated. Simulations and experimental measurements have been performed that verify the correctness of the DFT testing techniques proposed and validate the PLL design methodology.

In addition, a low-cost clock delay generation technique embedded in an ATE framework has been presented as an application of the 6th order PLL. We experimentally demonstrated two applications of this technique: precise clock alignment and jitter measurement with an effective sampling rate of 50 Gs/s. Simulations were performed to verify the correct operation of the technique and then compared with experimental results. The encoded bits, driven by a single clock, are outputted repeatedly from the HSD unit and fed to the fabricated high-order PLL that is mounted on the DIB board to achieve a constant phase delay. An adjustable phase delay resolution down to 15 ps within an 8.4 ns range has been realized. This technique can be extended to accurate phase generation for under-sampling techniques.

6.2. Future Work

Since this work is the first attempt in designing high-order PLLs using a pole-zero placement algorithm for Type II PLLs, many research paths can be taken for optimization of current work or development of new applications. A few potential future research topics are listed below:

1) Gain peaking reduction. The PLL closed-loop transfer function is restricted by the realization condition described in section 2.2.3. This condition inevitably forces a gain peaking near the cut-off region. This is a common problem in all type-II PLLs. This is undesired in many other applications due to the amplification of the phase noise in that frequency region. By synthesizing a gain reduction procedure while maintaining superb filtering action, this PLL design method could become universal and could be used for all application specific PLL implementations.

2) Undersampling for measuring the PLL transfer function. During the measurement of the PLL transfer function, the oscilloscope has some difficulties with signals of smaller amplitude in phase due to its limited time sampling resolution. As a result, the PLL can only be characterized when the output level is at most 30 dB below the input amplitude of 4.5 ns for an amplitude of 142 ps. Note that for large signal transfer

characteristics test, the built-in function ‘delay’ of the oscilloscope can measure phase delay less than 1 ps. This is achieved utilizing under-sampling for a high effective sampling frequency. One can adapt it to the current measurement method so that more insight can be gained by extending the measurement more into the stopband region. Another solution is to calculate the phase directly in the voltage domain using an analytical signal method as described in [22].

3) ΣΔ multi-bit encoding. From the discussion about the ΣΔ modulators, we know the performance can be improved by increasing the OSR, modulator order and number of output levels. In this work, the number of output levels is 2. Therefore, multi-bits can be utilized as an extension of this work.

4) IC implementation. In this work, a few applications are described, and the performances are directly related to the PLL. We see a noticeable amount of non-linearity, as witnessed in Fig. 47. These are caused by the non-ideality of the ICs inside of OPA355 and TLC2932. Therefore, it is suggested that in the future a custom IC is built so that the designer has more control over the VCO, PFD and the opamps. In addition, the design will be more compact and could be taken to a much higher speed.

Appendix

A 4-layer PCB is designed using the Cadence SPB PCB Editor. All the component values for the loop filter corresponding to Fig. 20 are listed in Table 2. It can be interfaced to an ATE tester or to the bench equipments using an FPGA (for a quick functional verification). The top layer contains the main circuitry including the PLL, their bypass capacitors and an optional buffer if the PLL output is to drive a 50 Ω oscilloscope. It also has an option to use a first order loop filter instead of the 5th order active RC such that a 2nd order PLL is realized. Two SMA connectors are used, to interface the PLL output and the reference clock with the oscilloscope. The bottom layer contains the connectors interfacing with the tester and three voltage regulators (may be bypassed) to power the circuit. The middle two layers are the split power (digital VDD, analog VDD and analog ground) and ground planes (analog VSS and digital VSS). The schematic is shown in Fig. 61 and the layout of all the layers is presented in Fig. 62.

Table 2: PLL loop filter component values.

Name	*Value*	*Name*	*Value*	*Name*	*Value*	*Name*	*Value*
R1	5.7 kΩ	R6	1.1 kΩ	R11	2 kΩ	C1	450 pF
R2	160Ω	R7	2 kΩ	R12	2 kΩ	C2	450 pF
R3	2 kΩ	R8	2 kΩ	R13	2 kΩ	C3	600 pF
R4	2 kΩ	R9	9.1 kΩ	R14	1.2 kΩ	C4	600 pF
R5	2 kΩ	R10	3.05 kΩ	R15	12 kΩ	C5	8 nF

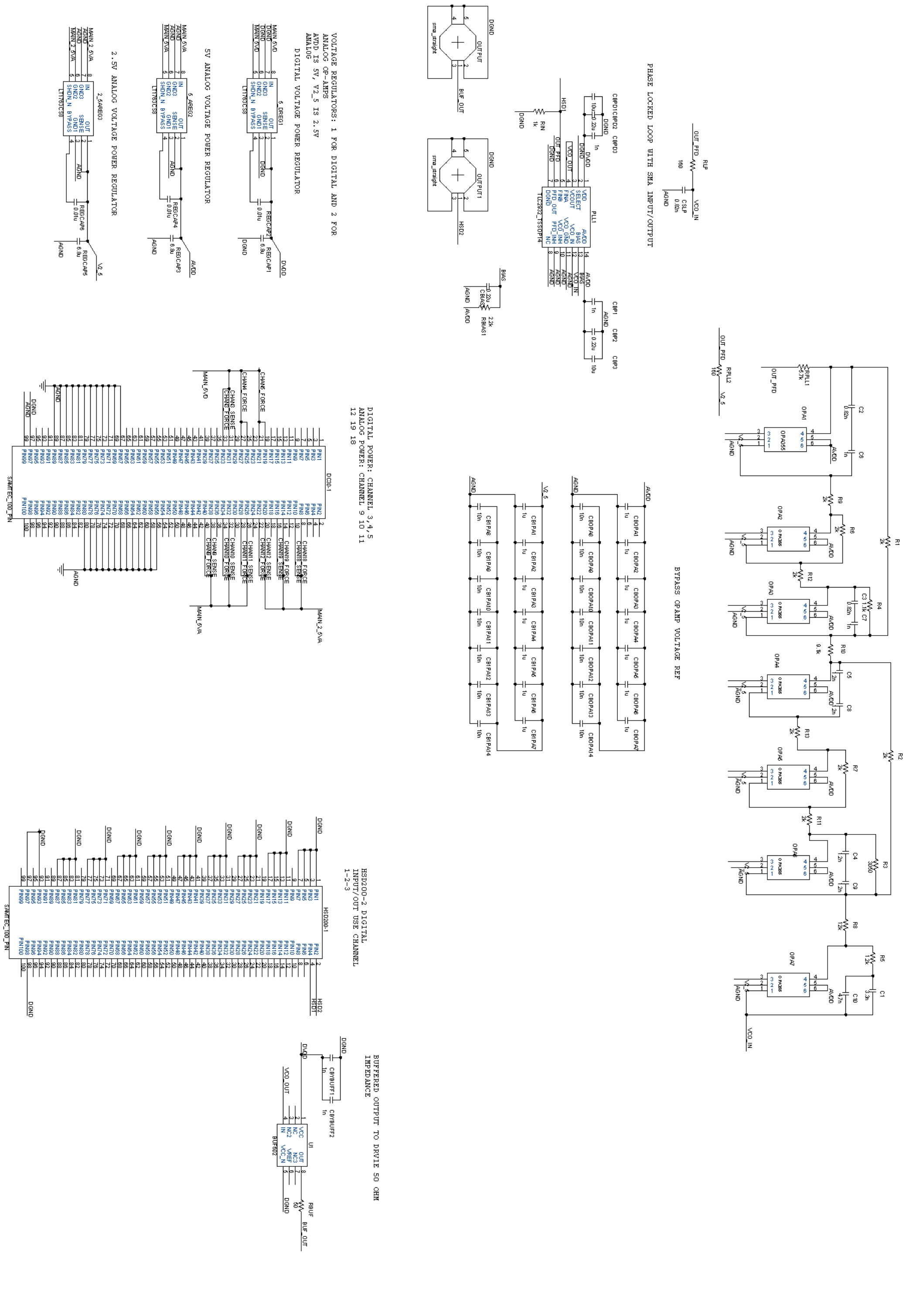
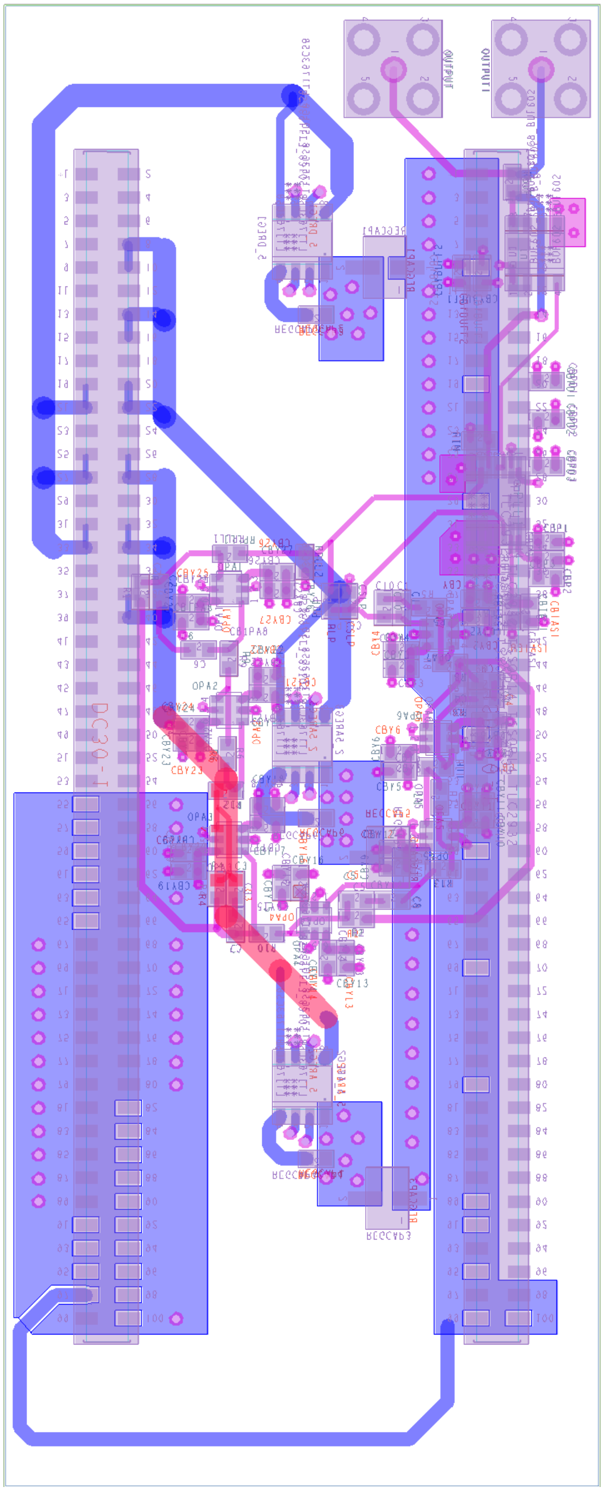
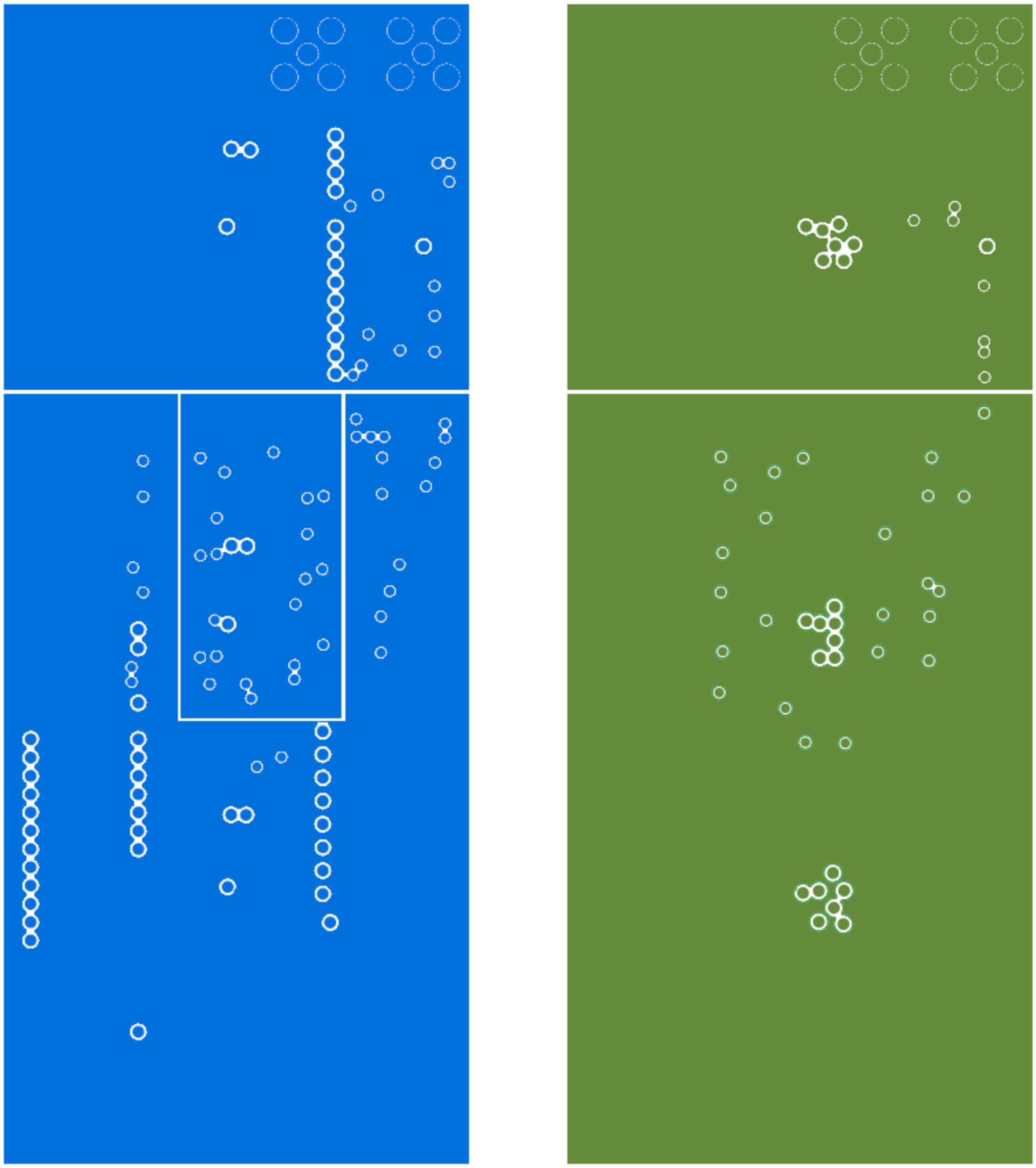

PHASE LOCKED LOOP WITH SMA INPUT/OUTPUT
BYPASS OPAMP VOLTAGE REF
VOLTAGE REGULATORS: 1 FOR DIGITAL AND 2 FOR ANALOG OP-AMPS
AVDD IS 5V, V2_5 IS 2.5V
DIGITAL VOLTAGE POWER REGULATOR
5V ANALOG VOLTAGE POWER REGULATOR
2.5V ANALOG VOLTAGE POWER REGULATOR
DIGITAL POWER: CHANNEL 3,4,5
ANALOG POWER: CHANNEL 9 10 11 12 19 18
HSD200-2 DIGITAL INPUT/OUT USE CHANNEL 1-2-3
BUFFERED OUTPUT TO DRVIE 50 OHM IMPEDANCE

Fig. 61: high-order PLL circuit schematic.


DC30-T
OPA2
OPA3
OPA4
CBY25
CBY27
CBY24
CBY23
CBY19
CBY6
CBY13
CB14

(a)



(b)

Fig. 62: PCB layout of a) the top and bottom layer B) the split power and ground planes.

References

[1] D. Fischette, R. DeSantis, and J. H. Lee, “An On-Chip All-Digital Measurement Circuit to Characterize Phase-Locked Loop Response in 45-nm SOI,” IEEE CICC, pp. 609-612, 2009.

[2] Lai MF, Nakano M. "Special section on phase-locked loop techniques". IEEE Transactions on Industrial Electronics 1996; 43(6):607–608.

[3] J.G. Proakis and D.G. Manolakis, *Digital Signal Processing, Principles, Algorithms, and Applications,* Prentice Hall, 1996.

[4] P. Chen, J.-S. Lai, and P.-Y. Chen “A FPGA vernier digital-to-time converter with 3.56 ps resolution and- 0.23 + 0.2 LSB inaccuracy,“ in Proc. IEEE CICC, Sep 2008, pp. 209-212

[5] B.K. Swann, B.J. Blalock, L.G. Clonts, D.M. Binkley, J.M. Rochelle, E. Breeding, K.M. Baldwin “100-ps time-resolution CMOS time-to-digital converter for positron emission tomography imaging applications,” IEEE Journal of Solid-State Circuits, Volume 39, Issue 11, Page(s):1839 – 1852, Nov. 2004.

[6] G. W. Roberts and M. Ali-Bakhshian, “A Brief Introduction To Time-to-Digital And Digital-to-Time Converters,” IEEE Transactions on Circuits and Systems II: Express Briefs, Volume 57 Issue 3, pp. 153-157, March 2010

[7] Roland E. Best, *Phase-Locked Loops, Theory Design, and Applications,* fifth edition, McGraw Hill, 2003.

[8] T. M. Almeida and M. S. Piedade, “High performance analog and digital PLL design,” in Proc. ISCAS, vol. 4, 1999, pp. 394–397.

[9] M.J. Burbidge, J. Tijou, A. Richardson, “Techniques for automatic on chip closed loop transfer function monitoring for embedded charge pump phase locked loops,” Design automation and test in Europe (Date), IEEE Proceedings, 2003, pp. 496–501.

[10] B. R. Veillette and G. W. Roberts, “Stimulus Generation for Built-In Self-Test of Charge Pump Phase-Locked Loops,” IEEE International Test Conference, pp. 698-707, 1998.

[11] H. Yang and R. Sarpeshkar, “A time-based energy-efficient analog-to-digital converter,” IEEE J. Solid-State Circuits, vol. 40, no. 8, pp. 1590–1601, Aug. 2005.

[12] L. Hernandez and E. Prefasi, "Analog-to-digital conversion using noise shaping and time encoding," IEEE Trans. Circuits Syst. I, Analog Digit. Signal Process., vol. 33, no. 7, pp. 2026-2037, Aug. 2008.

[13] G. Nagaraj, S. Miller, B. Stengel, G. Cafaro, T. Gradishar, S. Olson, and R. Hekmann, “A self-calibrating sub-picosecond resolution digital-to-time converter,” in IEEE MTT-S Int. Microwave Symp. Dig., Jun. 2007, pp. 2201–2204.

[14] B. Razavi: *Design of analog CMOS integrated circuits*, McGraw Hill, 2001

[15] R. J. Baker, *CMOS Circuit Design, Layout, and Simulation*, Second Edition, Wiley-IEEE Press, 2004

[16] B. Razavi, *RF Microelectronics*. Englewood Cliffs, NJ: Prentice-Hall,1998.

[17] Steven R. Norsworthy, Richard Schreier, and Gabor C. Temes. Delta-Sigma Data Converters: Theory, Design, and Simulation. IEEE Press, 2005.

[18] David Johns and Ken Martin. Analog Integrated Circuit Design. Wiley Publishers, 1997.

[19] M. Mahoney, Tutorial DSP-Based Testing of Analog and Mixed-Signal Circuits, The Computer Society of the IEEE, Washington D.C., 1987

[20] X. Haurie and G. W. Roberts, "A Design, Simulation and Synthesis Tool For Delta-Sigma-Modulator-Based Signal Sources", Proceedings of the International Symposium on Circuits and Systems, Atlanta, Georgia, pp. 715-718, May 1996.

[21] M. Burns, G. Roberts, An Introduction to Mixed-Signal IC Test and Measurement, Oxford University Press, Oxford, 2001.

[22] T. J. Yamaguchi, M. Soma, M. Ishida, T.Watanabe, and T. Ohmi, "Extraction of instantaneous and rms sinusoidal jitter using an analytic signal method," IEEE Trans. Circuits Syst. II, vol. 50, pp. 288-298, June 2003.

[23] S. Aouini, 2010. *Sigma-Delta Modulation Techniques and Circuits in the Amplitude, Time and Frequency Domain for Test Applications*. Thesis, (PhD). McGill University