ANALYSIS AND HARDENING OF ALL-DIGITAL PHASE-LOCKED LOOPS (ADPLLS) TO SINGLE-EVENT RADIATION EFFECTS

By

Yanran (Paula) Chen

Dissertation

Submitted to the Faculty of the

Graduate School of Vanderbilt University

in partial fulfillment of the requirements

for the degree of

DOCTOR OF PHILOSOPHY

in

Electrical Engineering

September 30, 2017

Nashville, Tennessee

Approved:

Lloyd W. Massengill, Ph.D.

Bharat L. Bhuva, Ph.D.

W. Timothy Holman, Ph.D.

Michael L. Alles, Ph.D.

T. Daniel Loveless, Ph.D.

Mark N. Ellingham, Ph.D.

ACKNOWLEDGEMENTS

I would like to express my thanks and appreciation to all of my committee members for their guidance, encouragement and support throughout my entire PhD program. Dr. Massengill has been a great advisor and a role model to look up to. He has been providing me with extraordinary guidance and advice during my graduate years on research topics, career paths and life in general. He taught me in his own ways on how to become a good researcher and engineer. Dr. Bhuva has been always around for me to ask questions and discuss technical details with. Dr. Holman has been a great source of new ideas and new perspectives for looking at research problems. Dr. Alles' abundant knowledge on device physics and different technology nodes definitely brings fresh opinions into this work. Dr. Ellingham's strong background on Mathematics is a great support for my circuit modeling work. Last but not least Dr. Loveless is a great source of inspiration for my work. His in-depth understanding of phase-locked loop (PLL) circuitries have been really useful when it comes to designing and debugging the circuit during electrical or radiation testing.

All the other professors in the RER group are on my list of people to pay appreciation to as well. They have contributed in various aspects to the completion of this work.To name a few: I would like to thank Dr. Schrimpf for his support and numerous motivating conversations. I thank Dr. Fleetwood for pushing me to become better at writing technical papers and better at doing research in general. I also would like to thank Dr. Zhang for her friendship and also for her help with packaging all my test parts, Dr. Kauppila for countless discussions on my research paths and Dr. Sternberg for his help with numerous times of TPA laser experiment

conducted at Vanderbilt.

I thank the Defense Threat Reduction Agency (DTRA), the Institute of Space Defense Electronics (ISDE) and Vanderbilt School of Engineering for providing the funding for this work and allowing me to complete this dissertation. I would also like to pay thanks to my fellow colleagues in the Radiation Effects Research (RER) Group, including a lot of those that have graduated already. It has been a pleasure working with everyone. I have been receiving constant help from my friends in this group since the I visited the campus before I even got admitted. Without you all, I could not have adapted to the life in the States and completed the program so smoothly.

Most importantly, I would like to express my deepest appreciation to my husband and my parents for their never-ending support and encouragement. This PhD program brought my husband and I together. I am really grateful that we met, fell-in-love and got married with the best wishes from our advisors and all of our families and friends during the course of it. Thank you to my families for believing in me when I did not believe in myself. Their love have been my sources of confidence and strength through all the ups and downs of my graduate career.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

A phase-locked loop (PLL) is a closed-loop feedback system that is capable of tracking the fixed phase relationship between the phase of output and the reference clock[1]. It is widely used for clock generator or clock recovery, as a frequency synthesizer, jitter attenuator and synchronization in the fields of communications, instrumentation, control systems, and multimedia apparatus, to name just a few[2][3]. An accurate clock signal is an important guarantee of the correct functionality of a system on chip (SoC). Design flow and circuit techniques of contemporary PLL circuits are typically quite analog intensive, which usually requires usages of resistors or capacitors. This is difficult to integrate with other digital-intensive parts of the SoC such as a digital baseband (DBB) and application processors (AP) and cumbersome to port between technologies. In addition, other issues like device mismatching and voltage headrooms start to exacerbates with CMOS technology scaling and degrades the performance of conventional analog/mixed signal (A/MS) PLLs [4]. All of those mentioned issues have been pushing researchers to seek digitally-intensive alternatives to conventional analog/RF functions in the most advanced deep-submicron process to reduce cost. In recent years, all-digital phase-locked loops (ADPLLs), as the digital counterparts for conventional A/MS PLLs, are becoming favored in the deep-submicron CMOS technologies [5] because of a variety of inherent advantages, i.e. high level of integrality and portability from technology to technology. In the past decade, ADPLLs have been applied in mobile phones, Bluetooth, and other

communication applications [6] [5].

As a type of reliability issues for integrated circuits (ICs) implemented in both terrestrial and space-bound systems, a single-event effect (SEE) occurs when a high-energy ionizing particle, such as a heavy ion, passes through the circuit. If the SE ion deposits charge near a transistor, the deposited charge may potentially change the nodal voltage that is associated with that transistor leading to a single-event upset (SEU) in memory storage elements or a single-event transient (SET) in combinational logic [7]. With the scaling of technology, integrated circuits (ICs) have been reported to exhibit increased susceptibilities to SEEs due to the decreasing feature sizes and increasing operating frequencies [8].

Objective of Research

As a potential candidate for space applications, radiation-harden-by-design (RHBD) solutions for ADPLLs are of prominent significance to preserve clock signals against SEEs.

While there are numerous studies on single-event effects in charge pump PLLs [9][10][11][12][13], little is published on radiation-effect in ADPLL based clock systems. The early endeavor can trace back to 1990s when D.J. Van Alen et. al discussed how ADPLLs-based clock system could be fault tolerant using the triple modular redundancy (TMR) technique [14]. There was a quiescent period after that for about a decade. Then in 2005, right after the ADPLLs were successfully employed in commercial communication applications, a few researchers have addressed issues related to radiation hardness of ADPLL topologies. A. N. Nemmi proposed in [15] hardening techniques of ADPLLs. But the lack of hardware results degrades the

credibility of the theory.

While there are many similarities between CPPLLs and ADPLLs, major functional modules in CPPLLs and ADPLLs are still greatly different. In this work, circuit-level simulation and experimental testing were conducted to characterize the subcircuits of different types ADPLLs to distinguish and analyze their individual contribution to the overall ADPLL SE vulnerability. Different ADPLLs with complex system architectures were also characterized and analyzed for SE vulnerability. Additionally, a novel time-domain analytical model for SEU-induced errors in ADPLLs was proposed which allows designers to distinguish the most SE sensitive modules in the ADPLL topology and apply selective hardening solutions pre-tapeout. RHBD hardening guidelines for different types of ADPLLs for different operating environment and targeted design specifications were proposed based on SE characterization and modeling of the ADPLL designs. Last but not least, the proposed model and hardening techniques are compared with existing work on A/MS PLLs to provide conventional PLL designers with insights on RHBD ADPLL designs.

Organization of the Dissertation

The research effort presented in this dissertation is organized as follows:

Chapter II presents general background information and a detailed discussion of ADPLL topologies.

Chapter III is a chapter on the back ground information for single-event effects (SEEs) and previous work on SEEs in A/MS PLLs.

Chapter IV illustrate the SET-induced errors and SEU-induced errors in different modules in ADPLLs respectively.

Chapter V details the overall SE characterization results and analysis on common ADPLL topologies.

A generalized time-domain model for SEU-induced errors for ADPLLs is proposed in Chapter VI and Chapter VII goes on by proposing RHBD design techniques for different types of ADPLLs.

Chapter VIII concludes.

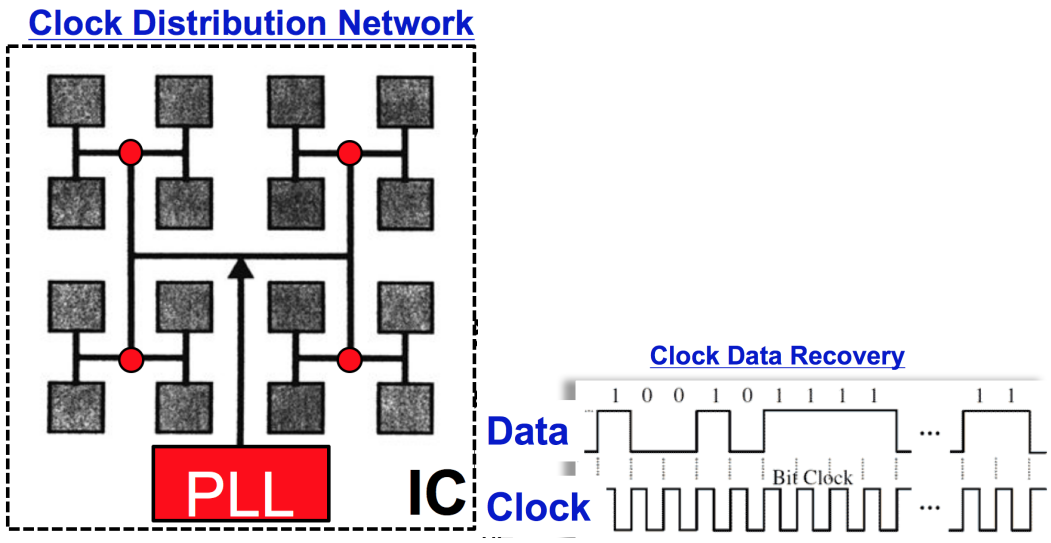
CHAPTER II

ALL-DIGITAL PHASE-LOCKED LOOPS (ADPLLS)

The functionality of modern integrated circuits (ICs) is highly reliant on the timing accuracy of the system clock signal. While standalone crystal or quartz oscillators are common clock sources for system clock signals from a few tens of kilohertz to hundreds of megahertz, clock signals in a frequency range higher than hundreds of megahertz are usually generated from active circuits. However, standalone active oscillators are subject to supply voltage changes and ambient temperature variations. Therefore, the active oscillator is usually put in a feedback system to allow users to have control on the oscillation frequency, which is defined as a phase locked loop (PLL).

PLLs are ubiquitous in modern SoCs. As indicated in Fig. 1a, PLLs are not only the most common clock sources for modern SoCs. Depending on the complexity of the system, PLLs can also be inserted locally into the clock distribution network for local frequency multiplication or active skew cancellation. In clock and data recovery systems, as shown in Fig. 1b, PLLs are utilized for providing the clock signal at the exact frequency and phase for deciphering the correct data out of the bitstream. And PLL-based frequency synthesizers are commonly deployed as local oscillators (LOs) to perform frequency translation between baseband (BB) and radio frequency (RF) in wireless transceivers.

Depending on the circuit configuration, PLLs can be classified into analog/mixed-signal PLLs (A/MS PLLs) and all-digital PLLs (ADPLLs). Simplified block diagrams for an A/MS PLL and an ADPLL are presented in Fig. 2. What distinguishes an

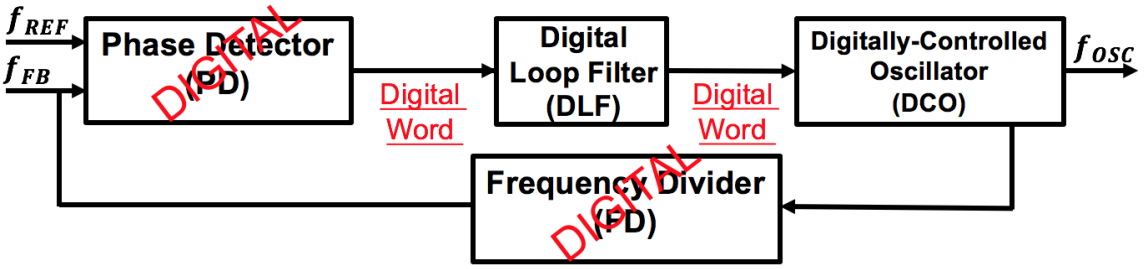


(a) Clock generator for SoCs (b) Clock and data recovery

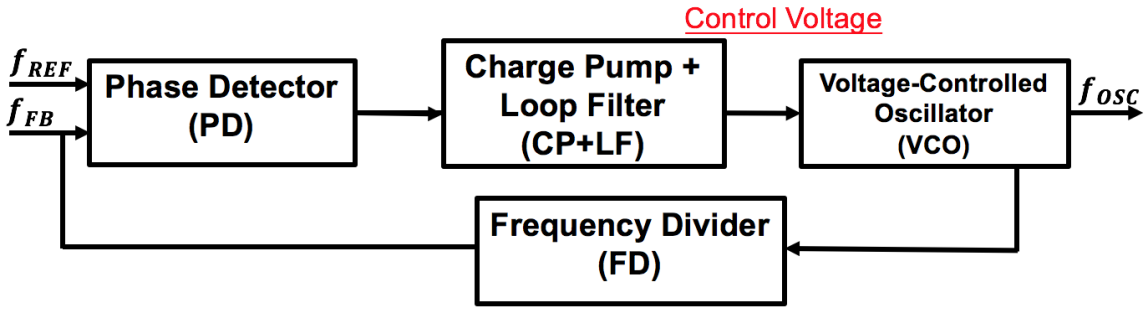
Figure 1: PLL usages in (a) modern SoCs and (b) clock and data recovery systems.

ADPLL from an A/MSPLL is that an ADPLL achieves fully digital frequency tuning instead of conventional analog voltage tuning. Conventionally, the output frequency changes linearly with the analog control voltage for a voltage-controlled oscillator (VCO). Frequency and phase tuning is achieved by adjusting analog control voltages through the A/MS control blocks in the feedback loop. In ADPLLs, the output frequency of the digitally-controlled oscillator (DCO) is usually controlled by a digital word. As shown in Fig. 2a, to accomplish "all-digital" frequency tuning, all the major subcircuits are replaced by their digital counter parts comparing to A/MS PLLs.

In an ADPLL, as shown in Fig. 2a, the digital phase detector (PD) compares the phase of the feedback signal (f_{FB}) to the phase of a reference signal (f_{REF}), and outputs a signal representing the frequency or phase error. The digital loop filter (DLF) filters out high-frequency noises in the digital control word and sends it to a DCO to adjust the oscillation frequency(f_{OSC}). The combined process thus tracks the frequency or phase of the reference signal. Frequency multiplication is completed



(a) ADPLLs



(b) A/MS PLLs

Figure 2: Block diagrams for (a) ADPLLs and (b) A/MS PLLs.

through the digital frequency divider (FD) in the feedback path in the PLL.

In A/MS PLLs, as shown in Fig. 2b, despite switching from DCO to VCO, a charge pump and a loop filter is used instead of a DLF. The loop filter is a simple RC low-pass circuit, and the charge pump (CP) generates current or voltage pulses proportional to the pulse width of the PD output and integrates it onto current control voltage. Till now, most PLLs are based on the charge pump architecture [16].

A broader definition for ADPLL includes both "all-digital" PLLs and "digital-intensive" PLLs. In "digital-intensive" PLLs, all the input and output signals for each module in the PLL are digital [4], which means the analog function can be contained inside the modules and only digital signals propagate inter-modularly. Some common topologies for "digital-intensive" PLLs include PLLs using VCOs with analog-digital converter (ADC) and digital-analog converter (DAC) wrappers [17][18][19] and PLLs

using digitally-controlled LC-tank oscillators [20] [21].

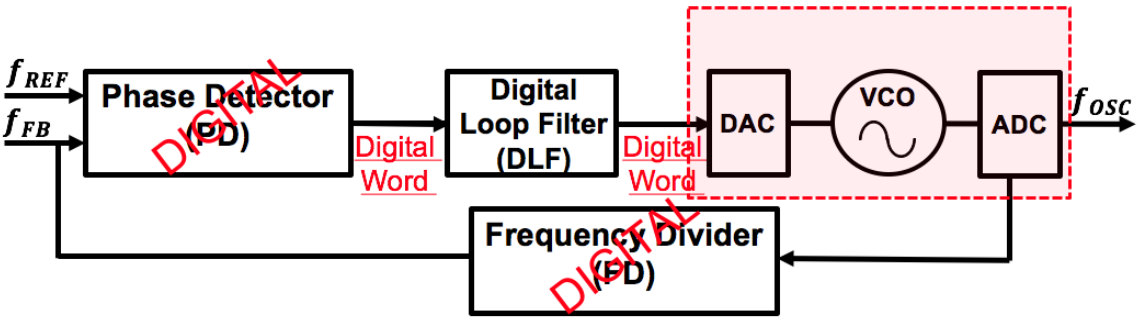


Figure 3: A simplified structure of a digital-intensive ADPLL using VCOs with analog-digital converter (ADC) and digital-analog converter (DAC) wrappers.

Basic ADPLLs

The locking characteristics and important specifications for ADPLLs are presented in the first subsection. The descriptions of each of the four modules of a basic ADPLL - namely phase detector (PD), digital loop filter (DLF), digitally-controlled oscillator (DCO) and frequency divider (FD) - are detailed in the following subsections.

Locking Characteristics and Important Specifications

Plotted in Fig. 4 is the output frequency of a typical ADPLL over time. The ADPLL goes through initial acquisition time during which it tries to find or track the expected output frequency and eventually reaches a steady state, which is referred to as the PLL "in-lock".

Important Specifications

A. Locking time and loop bandwidth

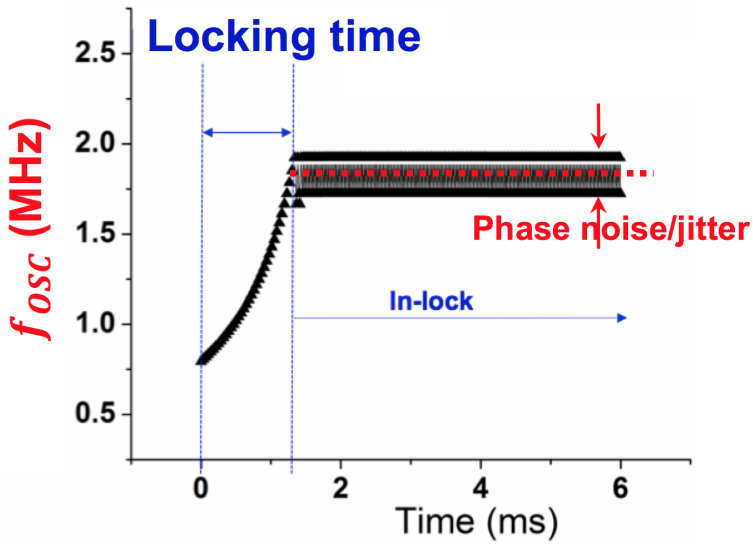


Figure 4: Output frequency of a typical ADPLL during initial acquisition time and steady-state operation.

As shown in Fig. 4, this time duration (around 1.2 ms) the PLL takes to reach a steady state is referred to as the "locking time" or "acquisition time". A PLL acts as a low-pass filter with respect to the reference modulation. Essentially, high-frequency reference noise is rejected. At the same time, a PLL acts as a high-pass filter with respect to VCO noises. ?Loop bandwidth? is the modulation frequency at which the PLL begins to lose lock with the changing reference (-3dB). PLL loop bandwidth essentially corresponds to the time it takes for the loop to respond to any changes at the input [22]. In general, higher loop bandwidth is recommended if the input clock reference is clean and stable, such as crystal oscillators. A lower PLL loop bandwidth is typically recommended if the input clock is noisy and cleaning is required.

B. Phase noise and phase jitter

After the frequency and phase acquisition, the ADPLL eventually enters a steady state, which is referred to as "in-lock", as shown in Fig. 4. Phase noise and jitter

determines the quality of the PLL output clock signal when the PLL is in-lock. The phase noise is typically expressed in dBc/Hz and represents the amount of signal power at a given sideband or offset frequency from the ideal clock frequency. Phase noise is the frequency domain representation of clock noise. Phase jitter, on the other hand, is the time domain instability of the clock signal and is often expresses in picoseconds (ps) or fractions of the ideal clock period. Two types of phase jitter are commonly used - period jitter and cycle-to-cycle jitter. Period jitter is the worst-case deviation from the ideal clock period. And cycle-cycle jitter is the worst-case clock period difference between adjacent clock cycles. In this work, the term "jitter" refers to period jitter.

ADPLL implementations suffer from tradeoffs between loop bandwidth and phase noise. Essentially, fast locking time of ADPLL is achieved through narrowing the loop bandwidth by tuning the loop parameters of the DLF at the expense of enhanced phase noise and spurs[23]. In Fig. 4, the ADPLL output frequency dithers between two frequencies, which results in jitter and phase noise, as it has reached the bandwidth limitation of the design.

C. Order and type

The order of a system refers to the highest degree of the polynomial expression in the denominator system transfer function in the phase domain. The type of a system refers to the number of poles of the open-loop transfer function located at the origin. The most commonly used A/MS PLLs are 1st-order type-I, 2nd-order type-I and 2nd-order type-II PLLs. When implementing loop filters in the digital fashion, adding a closed-loop pole is accompanied by the addition of an open-loop

pole. Therefore, 1st-order type-I and 2nd-order type-II ADPLLs are commonly used.

Locking Characteristics

The implementations of the PD and the DLF in an ADPLL dictate the loop tracking characteristics.

A. PD

The PD determines whether an ADPLL is in phase-lock or frequency-lock in the steady-state. As indicated in Fig. 5, frequency-lock is referred to a state when an ADPLL is in-lock, the ADPLL outputs only the correct frequency but the time-difference between edges of the reference clock signal (f_{REF}) and PLL output clock (f_{OSC}) is unknown. However, in the phase-lock state, an ADPLL not only produces the desired frequency, but also aligns the clock edges of f_{OSC} and f_{REF}.

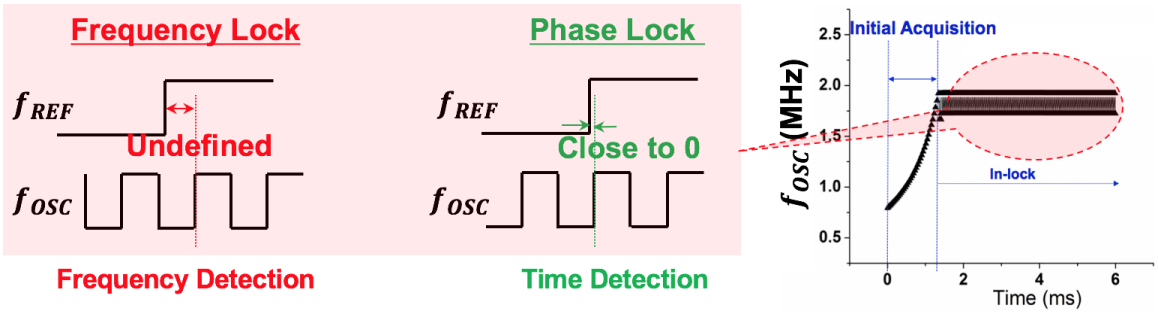


Figure 5: Time detection is needed for ADPLLs to reach phase-lock and frequency detection is needed for frequency-lock.

Phase-lock is superior to frequency-lock because of the additional edge-alignment feature. Phase-lock ADPLLs are used in clock-data-recovery systems [24] or active skew cancellation applications [25], where accurate frequency and phase are required. However, because that edge-alignment generally requires large design complexity and long settling time, frequency-lock ADPLLs are implemented in applications where

the phase accuracy of the PLL output clock signal is not required, such as LOs in transceivers or high-speed clock generators.

Different detection schemes are required for frequency-lock and phase-lock. To achieve only frequency-lock in steady state, a PD deploying frequency-detection scheme, i.e. a "frequency-based" PD, is used, whereas phase-lock requires usages of time-based PDs with the time-detection schemes. Frequency detection schemes are based on frequency-counting algorithms, which are generally more efficient than time detection schemes for frequency locking.

As stated above, using frequency-based PDs can drastically reduce the system settling time, but lack of phase tracking capability. While time-based PDs can perform both frequency and phase, the loop bandwidth is usually made wide meet the system phase noise requirement. To solve the conflicting requirements of the PLLs, "FSM-based PDs" were proposed to allow the loop to switch between different tracking modes with different loop bandwidths controlled by a gear-shifting FSM. Using the frequency-based PD in the frequency-tracking mode of the PLL with large loop bandwidth and the time-based PD in its phase-tracking mode with narrow loop width allows the loop to achieve locking state in an optimum timely fashion.

Plotted in Fig. 6 is the ADPLL output frequency over time in an example case when the ADPLL with FSM-based PD undergoes frequency and phase acquisition. The locking process monitor (LPM) is the FSM controller for mode switching in [26]. The light curve represents the case when the LPM is on, while the darkened curve indicates when the LPM is off, i.e. the loop stays in phase-locking mode the entire time. The loop bandwidth decreases from frequency acquisition mode to phase acquisition to allow for less abrupt digital control word changes during phase-tracking

mode. As shown in Fig. 6, the overall locking time improves with LPM on. Sometimes, the loop bandwidth could be gear-shifted several time for the most optimal acquisition performance[4].

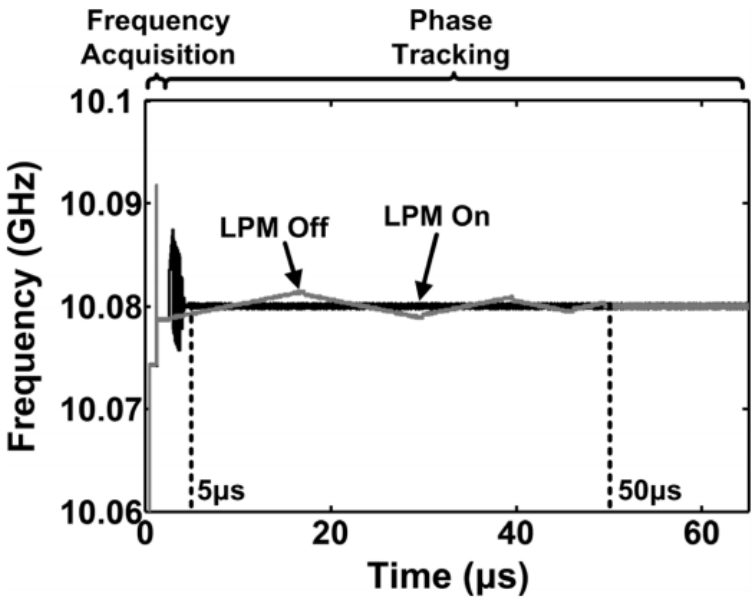
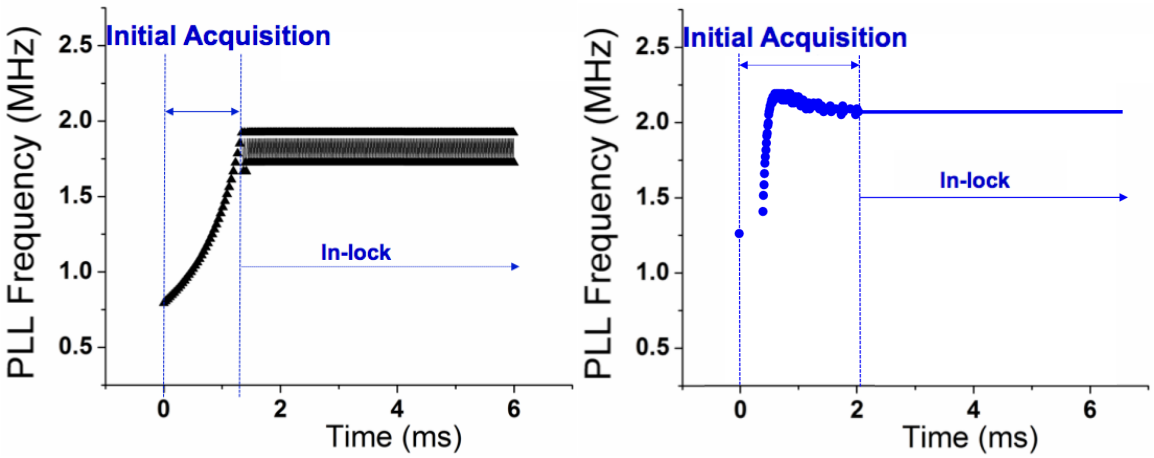


Figure 6: Comparison of locking behaviors with and without the aid of locking process monitor (LPM) [26].

B. DLF

The DLF in an ADPLL determines the order of the system. Plotted in Fig. 7 are examples of two simulated ADPLL designs undergoing initial acquisition at startup and eventually settling in locking state. Essentially, this is a plot demonstrating the system input frequency step response [22]. In Fig. 7a, the 1st-order bang-bang ADPLL exhibit linear 1st-order system behaviour during initial phase and frequency acquisition. In Fig. 7b, the output frequency of the 2nd-order frequency-linear ADPLL overshoots the input frequency during acquisition, which corresponds to a underdamped 2nd-order system behaviour ($\zeta < 1$).



(a) 1st-order bang-bang ADPLL (b) 2nd-order frequency-linear ADPLL

Figure 7: Simulation results on loop locking characteristics for (a) a 1st-order bang-bang ADPLL and (b) a 2nd-order frequency-linear ADPLL on IBM 180 SOI technology.

Phase Detector (PD)

A phase detector (PD) is an indispensable element of the ADPLL. The PD generates a digital word representing the difference in phase or frequency between two input signals. PDs can be categorized into three basic categories based on how the errors are detected - frequency-based PD, time-based PD, and FSM-based PD.

A. Frequency-based PD

Frequency-based PD detects the frequency error between the current and the expected output period. They are sometimes termed frequency detectors. This term is not used in this dissertation to avoid the confusion between frequency detectors and frequency dividers. A PLL uses a frequency-based PD locks on the frequency rather than phase, i.e. small phase offset exists between the output signal and the reference clock signal. This type of PD is used in wireless applications such as a local oscillator (LO)[4] or clock generator for Globally-Asynchronous Locally Synchronous

(GALS) architectures [27], where there is no need to align the clock phase.

If we define the period of the oscillator output as T_{OSC} and the the reference clock period as T_{REF}. It is convenient in practice to normalize the transition timestamps in terms of actual T_{OSC} since it is easy to observe and operate on actual oscillator output clock events. The clock "phase" for the oscillator clock at timestamp t can be defined as:

$$\theta_{OSC} = \frac{t}{T_{OSC}} \tag{1}$$

The "phase" of the oscillator output clock could be estimated as M by accumulating the number of significant (rising or falling) edge transitions over a reference clock cycle. The frequency error at that reference clock cycle can be calculated by comparing the actual (M) and expected number (N) of output clock cycles (T_{OSC}) in one reference cycle (T_{REF}).

One straightforward implementation outputs the direct difference between the actual (M) and expected number (N) of output clock cycles in one reference cycle, i.e. N-M [28]. This configuration is referred to as "integer-based frequency PD" in the following text. The other, more complicated, configuration of PD is referred to as "fraction-based frequency PD" in the following text. The phase error E_f is given by

$$E_f = \frac{T_{REF}}{N} - \frac{T_{REF}}{M}. \tag{2}$$

For a common implementation of DCO, where the digital control word is linearly proportional to the output period, the adjustment needed for the control word is

given by

$$\frac{D}{N} - \frac{D}{M}, \tag{3}$$

where D is the digital control word corresponding to the desired output frequency [4]. For different applications, D may vary within the pull-in frequency range of the ADPLL. D_{center}, the control word that corresponds to the center frequency, is chosen in replacement of D for simplicity of design implementation. Therefore, the output of the PD is shown in

$$\frac{D_{center}}{N} - \frac{D_{center}}{M}, \tag{4}$$

ADPLLs with integer-based frequency PD suffer from limited frequency pull-in range due to the asymmetry between the positive and negative frequency tuning steps when N is fixed. While fraction-based frequency PD operates more linearly in frequency domain comparing with integer-based frequency PD, they could be referred to as integer-based linear PD and fraction-based linear PD comparing to non-linear PDs, such as bang-bang PDs described in the following section. Other frequency-based PDs are uses advanced algorithms, including binary-search algorithm [29][30] and frequency estimation algorithm[31], which are not discussed in the scope of this work.

B. Time-based PD

Time-based PD detects the phase error between the feedback and the reference clock signal based on the relative clock edge locations of the two signals. Comparing to a PLL with a frequency-based PD, a PLL with a time-based PD not only locks onto the correct frequency but also aligns the clock edges of the output clock signal with the reference clock signal, i.e. phase offset between two clock edges is close to zero.

Even though phase-lock is superior to frequency-lock, phase alignment generally takes a longer time than locking onto the correct frequency. This type of PD is required in clock and data recovery systems[], because inaccuracies in the phase of the clock signal can introduce extra delay and possibly cause bit offset when interpreting the data stream.

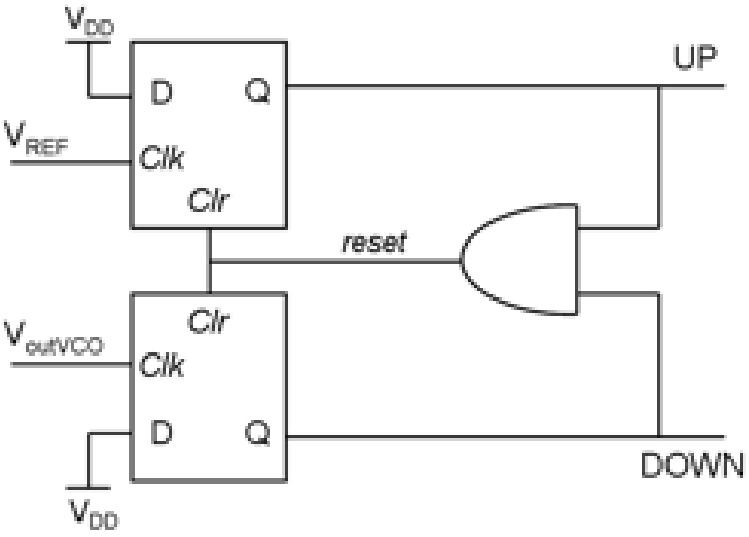
B.1 Bang-bang PD

A bang-bang PD is one of the most commonly used time-based PDs due to its simplicity in circuit design[32]. Different types of bang-bang (also called single-bit or lead-lag) PDs are used in bang-bang ADPLLs as well as CPPLLs. As shown in Fig. 8a and Fig. 8b, bang-bang PDs generally contain one or more D-flip-flops. Depending on whether the clock edge of the feedback signal (f_{FB}) is leading or lagging that of the reference signal (f_{REF})), a '0' or '1' is generated for both up or down signals indicating an increase or a decrease on current digital control word.

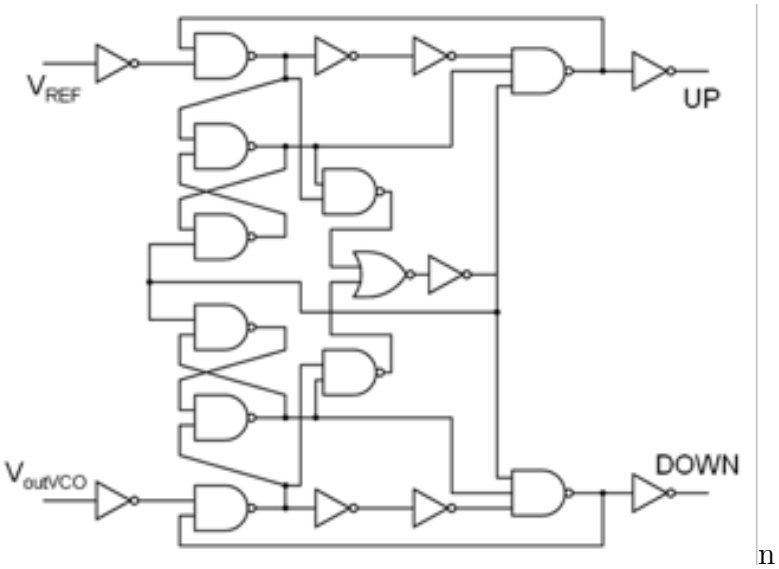
The nonlinearity resulted from single-bit resolution can be responsible for huge undesirable spurs and noise at the PLL output. In addition, this nonlinearity contributes to the nonlinear frequency tracking behavior of the loop.

B.2 Time-to-Digital Converter (TDC)

Another type of time-based PD is a time-to-digital converter (TDC). A TDC takes the time difference between the reference clock signal and the feedback clock signal (i.e. the accumulated phase error between the two signals) and directly converts that to a multi-bit digital word. As depicted in Fig. 9, the feedback clock signal HCLK passes through a string of non-inverting delay elements, such as buffers. An array of flip-flops sample the delayed clock vector D(1:L) on the rising edge of the reference clock FREF and output pseudo-thermometer-coded output Q(1:L)



(a)



n

(b)

Figure 8: Different digital implementation of bang-bang PDs [33].

containing information on the timing separation between the rising edge of FREF and the rising and falling edges of FFB. The pseudo-thermometer coded output could be converted to binary that measures the HCLK-to-FREF delay in units of a buffer delay or the pulse width of HCLK. Hence, a TDC outputs a digital word DW that satisfies Eqn. 5

$$E_p = DW \cdot \Delta_{TDC}, \tag{5}$$

in which E_p is the phase error between the two signals and Δ_{TDC} is the resolution of the TDC, i.e. the inverter delay in Fig. 9.

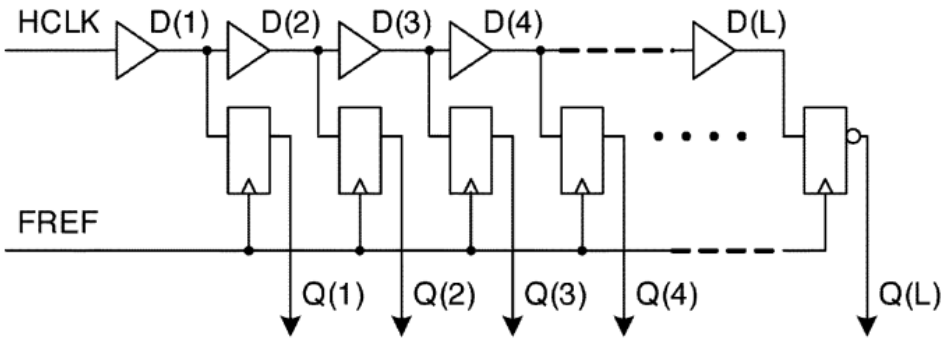


Figure 9: A simplified structure of TDC core [34].

Several variations of TDC implementations are commonly used, including delay-line-based [35] TDCs and gated-ring-oscillator-based [36] TDCs. TDCs can also be cascaded after a bang-bang PD to facilitate in converting a single-bit UP/DOWN pulse to a multi-bit digital word based on the time duration of the UP/DOWN pulses[37][38], as indicated in Fig. 10. However, the structural differences of the TDCs have minimal impacts on the circuit behavior. Eqn. 5 holds true for all TDC design implementations. Since, the TDC output is linearly proportional to the detected phase error, a PLL using a TDC is able to reach locking state faster comparing to

ones using bang-bang PDs only.

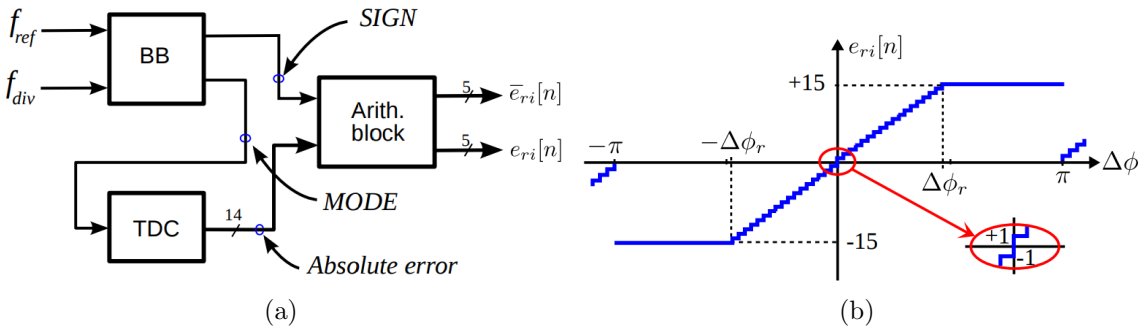


Figure 10: PFD: (a) block diagram and (b) its transfer function [37].

C. FSM-based PD

An FSM-based PD is a combination of a frequency-based PD and a time-based PD controlled in a finite-state machine (FSM)[39][26]. This gear shifting idea in analog system can result in voltage or charge losses due to mismatches. However, it is easy to implement in digital systems. (NEED REFERENCE!!!!!!!!!!!!!!!!!!!)

As shown in Fig. 11, at the beginning of operating, this ADPLL is in the frequency acquisition mode utilizing the frequency-based PD only. After frequency acquisition is completed, the ADPLL enters the phase-acquisition mode. The ADPLL increments or decrements the DCO control word based on the output of the time-based PD.

Recently, TDCs are also used as FSM-based PDs, as TDCs can perform both frequency detection and phase detection. As shown in Fig. 12, the DCO clock (CKV) through a chain of inverters such that each inverter output would produce a clock slightly delayed from that of the previous inverter [5]. The staggered clock phases are then sampled by the same reference clock. By detecting the transitions from '1' to '0' and from '0' to '1' , the rising edge and falling edge of the DCO clock are detected, based on which half-period of the DCO clock can be calculated in terms of inverter

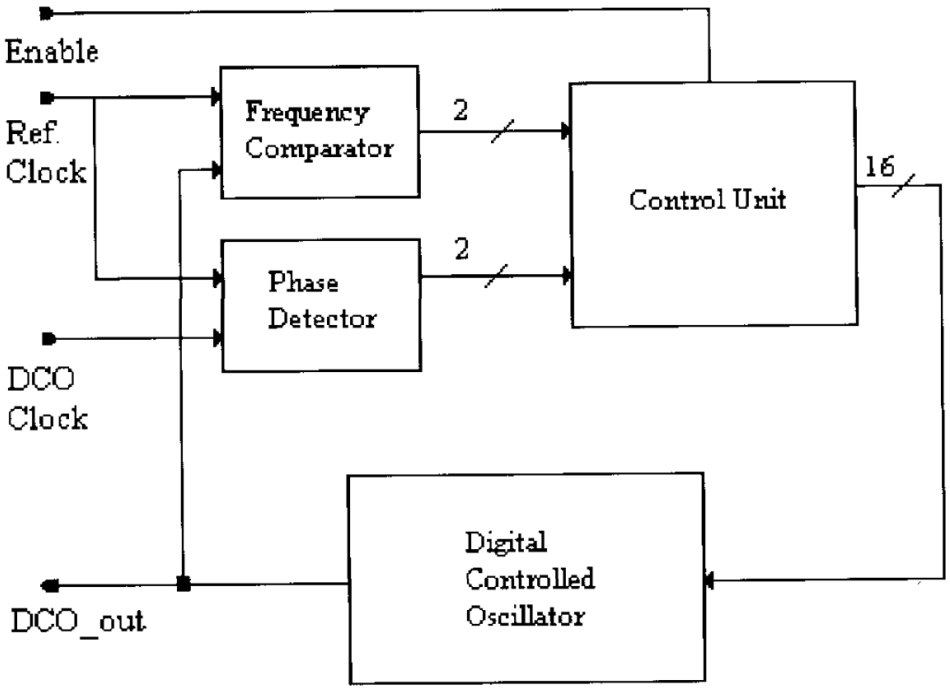


Figure 11: A block diagram of an ADPLL using frequency-based PD (i.e. frequency comparator in the figure) and time-based PD (i.e. phase detector in the figure) [39].

delays. Therefore, frequency tracking can be conducted based on the difference between current digital word and anticipated digital word for the half-period. As used in the time-detection mode, the TDC detects the time difference between the reference edge and the following rising edge of CKV. With that information, phase-tracking can be performed.

The above three categories of PDs are classified based on their phase detecting mechanisms. PDs can also be categorized based on the linearity of the operation of the PD in frequency or phase domain, and the corresponding ADPLL s can be classified into linear and non-linear ADPLLs. Intuitively, frequency-based PDs are frequency-linear PDs due to their operation linearity in frequency domain. TDCs are phase-linear PDs because TDCs output a digital word that linearly proportional to

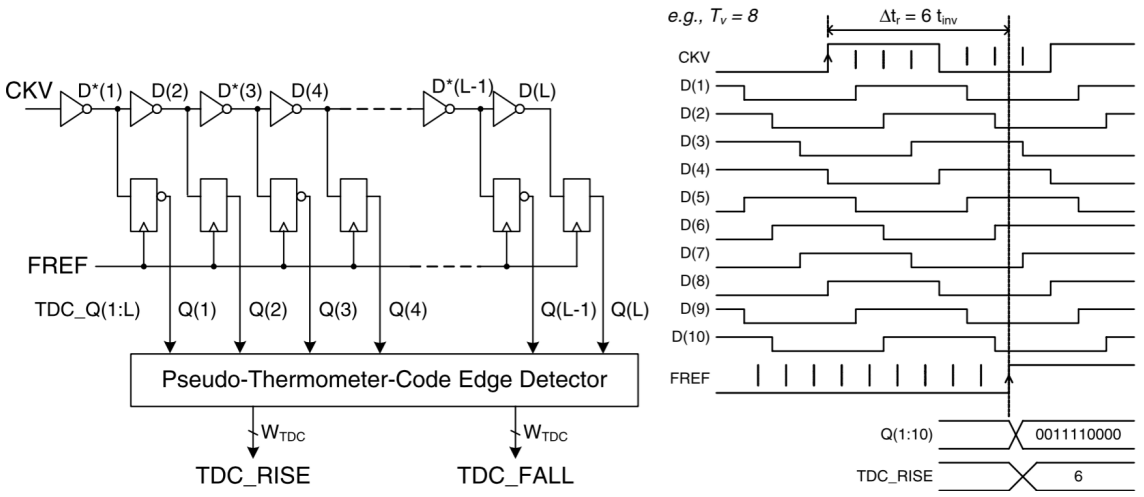


Figure 12: Time-to-digital converter serves as a "FSM-based PD" [5].

the incoming accumulated phase error. And finally, bang-bang PDs are non-linear PDs.

Digital Loop Filter (DLF)

The digital loop filter (DLF) is the core digital control unit for an ADPLL, which is analogous to the combination of the charge pump and the loop filter in analog/mixed-signal PLL [4]. The purpose of the DLF is two-fold - integrating the PD output on to the current digital control word and keeping the locking control word as steady as possible, i.e. filtering out fluctuations when PLL is in-lock.

Integrating of digital words in performed through an integrator, which corresponds to a pole in the DLF. An ADPLL with a 1st-order DLF (containing only one digital word integrator with proportional gain of α) is referred to as a 1st-order ADPLL. 1st-order ADPLLs generally feature fast dynamics and are used where fast frequency/phase acquisition is required, such as direct transmit modulation[40].

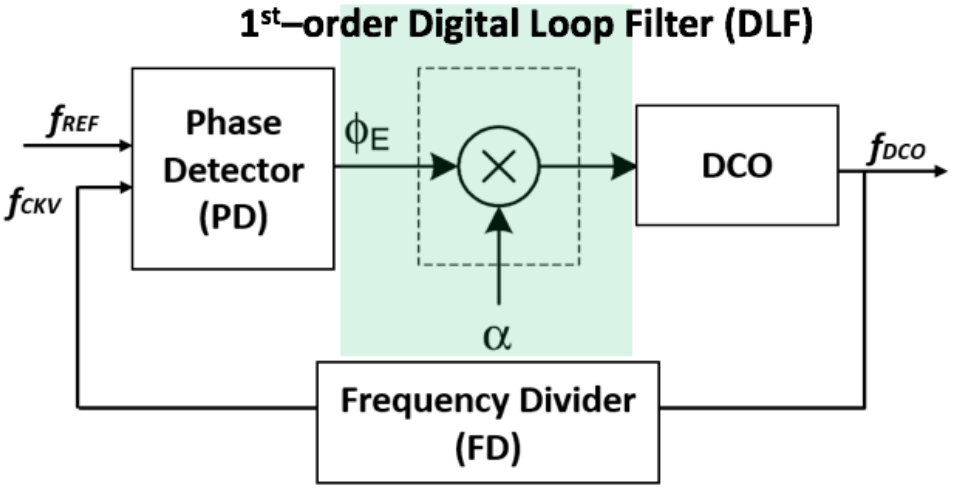
1st-order loops react fast to incoming digital word changes and their filtering

capability is low. An extra integrator, i.e. an extra pole, with integral gain of ρ, can be added to the DLF for filtering purposes, as shown in Fig. 13b. Together with the proportional path, this forms a 2nd-order proportional and integral (PI) filter. Similarly, an ADPLL with a 2nd-order DLF is referred to as a 2nd-order ADPLL. A chief advantage of 2nd-order loop is that the steady-state phase error goes to zero for a step frequency change, while the phase error in a 1st-order PLL loop is proportional to the frequency offset. Therefore, 2nd-order ADPLLs are commonly used as frequency synthesizers [4] to allow rapid frequency hoping without any residual phase errors. In addition, 2nd-order loop has better filtering capabilities of oscillator noise, leading to improvements in the overall phase-noise performance, comparing with 1st-order loops. As such, this topology is often used in applications with stringent phase noise requirements [41][42][43][44] [45].

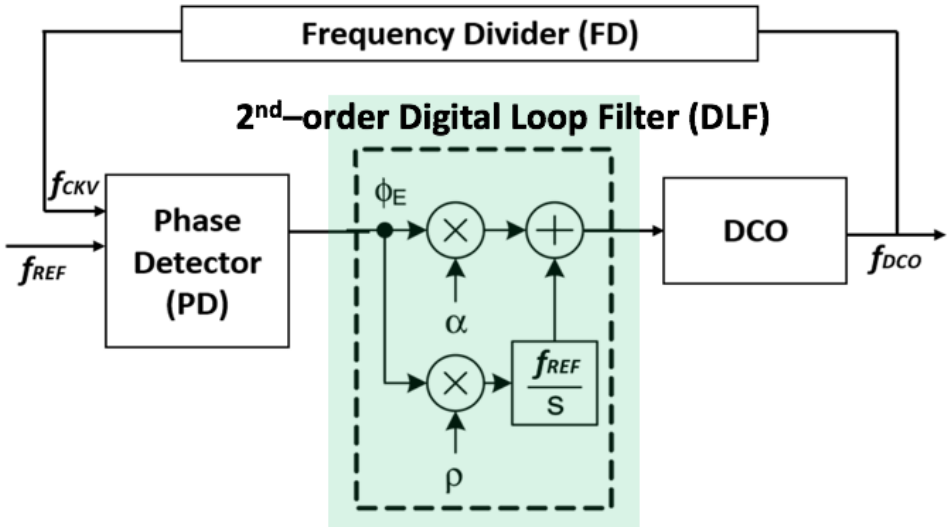
To give rise to 2nd-order DLF configuration and provide even more filtering for incoming word changes, DLF could be constructed as a combination of finite impulse response (FIR) and infinite impulse response (IIR) filters cascaded with the PI filter. Fig. 14 demonstrates the basic architectures of an FIR filter and an IIR filter. The operation of an FIR filter of order N (as shown in Fig. 14a) is illustrated in Eqn. 6

$$y[n] = b_0x[n] + b_1x[n-1] + \cdots + b_Nx[n-N], \tag{6}$$

where x[n] is the input signal, y[n] is the output signal and b_i is the value of the impulse response at the corresponding i^{th} instant (i=0,1,2...,N). FIR filters only have feedforward paths. IIR filters corresponding to Fig. 14b are often described as Eqn. 7



(a)



(b)

Figure 13: Block diagram of a typical (a) 1st-order ADPLL and (b) 2nd-order ADPLL with 2nd-order DLF.

$$y[n] = \frac{1}{a_0}(b_0x[n]+b_1x[n-1]+\cdots+b_Px[n-P])-a_1y[n-1]-a_2y[n-2]+\cdots+a_Qy[n-Q], \tag{7}$$

where P, Q are the filter order of feed forward and feedback, respectively. b_i are feed forward filter coefficients and a_i are feedback filter coefficients.

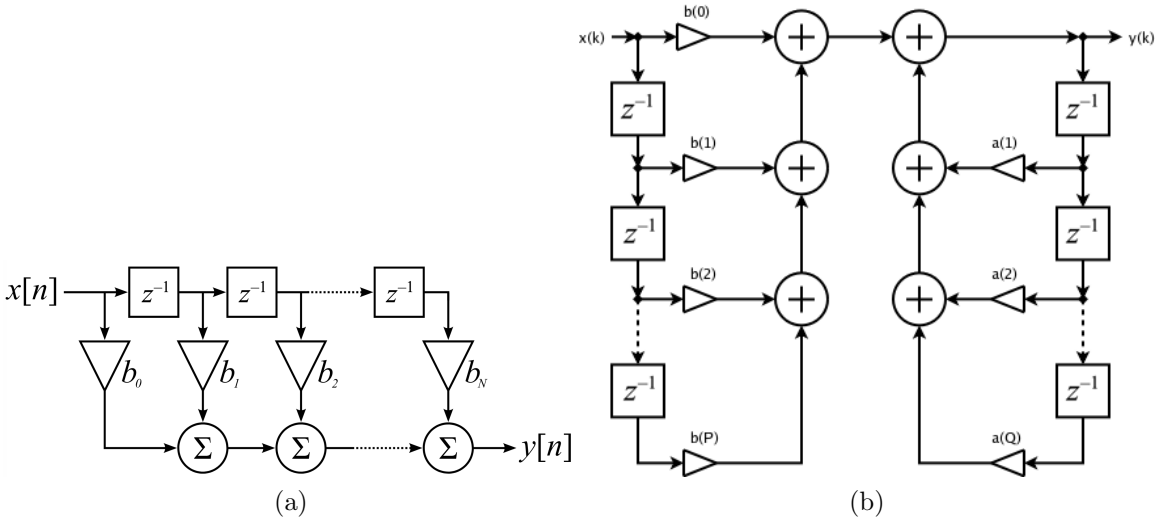


Figure 14: Basic architectures of (a) an FIR filter [46] and (b) an IIR filter [47].

Therefore, the transfer functions for the FIR filter above are given in Eqn. 8,

$$H(z)_{FIR} = \sum_{i=0}^{N} b_i z^{-i}, \tag{8}$$

in which all the poles are located at origins so that FIR filters are unconditionally stable. Similarly, the transfer function for the IIR filter is shown in Eqn. 9

$$H(z)_{IIR} = \frac{\sum_{i=0}^{P} b_i z^{-i}}{\sum_{j=0}^{Q} a_j z^{-j}}. \tag{9}$$

IIR filters could easily become unstable due to the complex pole-zero relationship in the transfer function. However, IIR filters are usually more compact and provide

stronger filtering capabilities. This problem is usually solved by using a cascade of single-pole IIR filters, which are unconditionally stable, as shown in Fig. 15. The cascaded IIR filter attenuates the noises from reference signal and the digital PD at (20·n)-dB/dec slope, in which n is the number of poles introduced by the IIR filter [4][48].

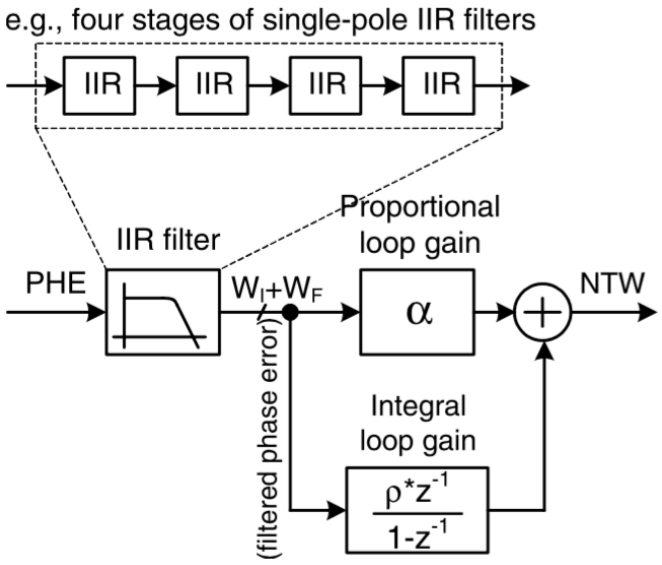
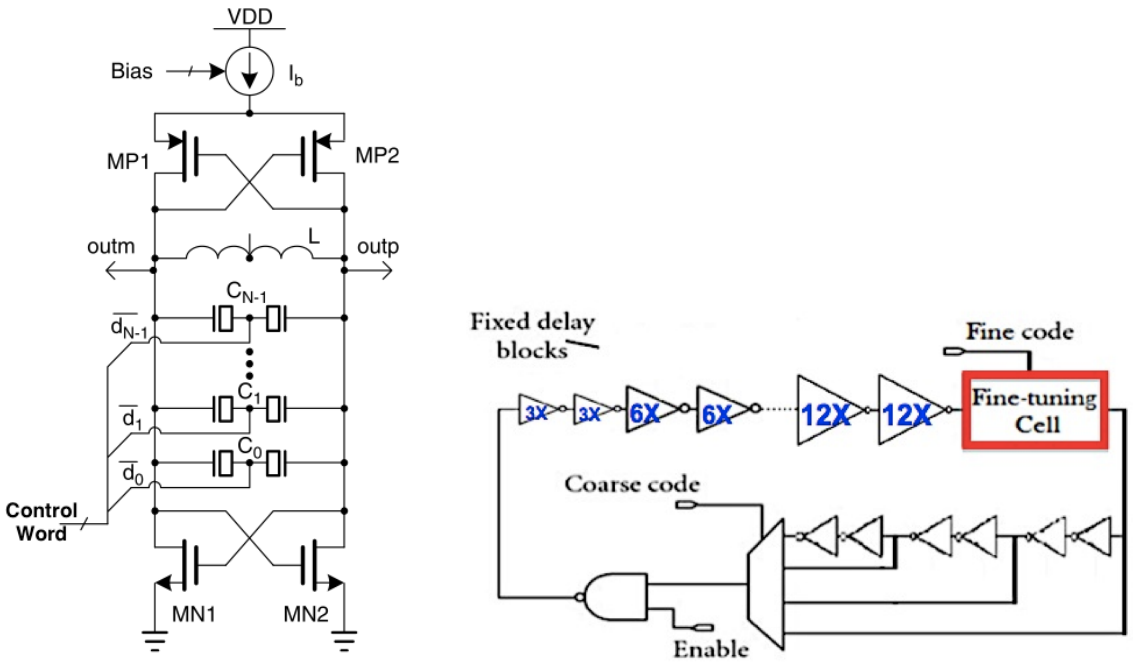


Figure 15: Block diagram DLF consists of cascaded single-pole IIR filters and PI filters in an ADPLL [4].

Digitally-Controlled Oscillator (DCO)

The design of digital controlled oscillators (DCOs) can be derived from the design of voltage-controlled oscillator (VCO). Two common topologies of DCOs are LC-tank-based and ring-based DCOs. The LC-tank oscillator is based on resonance between L and C components in the circuit. As shown in Fig. 16a, the oscillating frequency is controlled by using control words to control how much capacitance from the varactor bank is resonating with the inductor during operation. On the other hand, in Fig. 16b,

the ring-based is based on the resonance of odd number of inverting gates tied in a loop fashion. The frequency of oscillation is manipulated by either adjusting the number of inverting gates in the ring or the delay of each inverting gates through the control word. Limited by the phase-noise and jitter performance of pure digital oscillator, i.e. ring oscillator, LC-tank resonance based ADPLL are more commonly used in communication applications[49]. As stated previously, in this case, an ADPLL is really a digital-intensive PLL. For the purpose of this work, we use ring-oscillator-based ADPLL as an example to characterize the single-event sensitivity of ADPLLs. However, the conclusion of the work could be extended to LC-tank-based digital-intensive PLLs.



(a) LC-tank DCO

(b) Ring DCO

Figure 16: Different DCO structures: (a) LC-tank based DCO [4] (b) Ring-based DCO [50].

Frequency Divider (FD)

A frequency divider (FD) is commonly used in the feedback path to accomplish frequency multiplication at the output of the ADPLL. In an ADPLL with an integer-N FD, the PLL generates an output clock signal at an integer multiple of the reference frequency ($f_{PLL} = Nf_{REF}$), while in an ADPLL with an fraction-N FD, the output frequency can increment by fractions of the reference frequency.

Many digital implementations of integer-N frequency dividers have been developed for mixed-signal PLLs (charge-pump PLLs). Cascading divide-by-2 dividers in [51] to form divide-by-N frequency divider, where N is a multiple of 2, is the most commonly used topology for both charge-pump PLLs and ADPLLs.

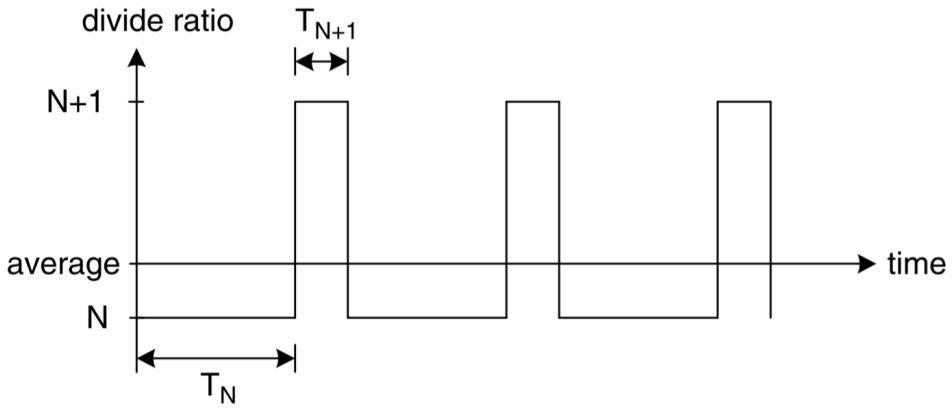


Figure 17: Alternating division ratio of fractional-N PLL.

Fractional-N PLL can achieve arbitrarily fine time-averaged frequency-division ratio of (N_{avg}) by modulation of the instantaneous integer division ratio of N and N+1. Fig. 17 reveals the principle in which the integer division is periodically altered from N to N+1. The resulting average divide ratio will be increased from N by the

duty cycle of the N+1 division:

$$N_{avg} = \frac{NT_N + (N+1)T_{N+1}}{T_N + T_{N+T}} = N + \frac{T_{N+1}}{T_N + T_{N+1}} \tag{10}$$

Analyzed Modular Design Implementations

Selective implementations of each module in common ADPLLs are analyzed in this work:

(1) PD: All frequency-based, time-based and FSM-based PDs are analyzed in this dissertation. RHBD design considerations and tradeoffs are presented for different types of PDs.

(2) DLF: Different orders of DLFs (i.e. from 1st-order to 3rd-order) are analyzed in this dissertation. RHBD design guidelines are proposed to make design choices on the design parameters of DLFs.

(3) DCO: DCROs are the focus of this dissertation due to their synthesizability and area efficiency compared with LC-tank oscillators.

(4) FD: Integer-N FDs are the focus of this dissertation due to their popularity. RHBD considerations can be extended for fraction-N FDs.

ADPLL Modeling

A top-down design flow starting with system behavior modeling is commonly applied to large-scale digital system like ADPLLs. Despite the highly quantized digital nature of all the components used in ADPLL topologies, behavioral modeling of ADPLL is useful for a fundamental understanding of the functionality of the design. System modeling of ADPLLs in literature usually falls into one of the two categories

- phase-domain modeling and time-domain modeling.

Phase-Domain ADPLL Modeling

ADPLL is a discrete-time sampled system implemented with all digital components connected with all digital signals. Consequently, the z-domain representation of the system is the most natural and accurate fit[52][53][54][55][56].

Most of existing phase-domain ADPLL modeling has been conducted on TDC ADPLLs due to their behavioral similarity with conventional CPPLLs. In fact, Kratyuk *etal*. presented in [48] a direct translation between TDC-based ADPLLs and conventional CPPLLs. Essentially, in a typical TDC ADPLL, the transfer function of TDC-based PD can be approximated as the equivalent of the combination of a PD and a CP in the CPPLL, shown in Eqn. 11,

$$I_{CP} = \frac{T_{REF}}{\Delta_{TDC}}, \tag{11}$$

in which I_{CP} is the nominal charge-pump current,T_{REF} is the reference clock period and the resolution of the TDC is Δ_{TDC}.

The proportional path gain (α) and integral path gain (ρ) in the DLF can also be approximated with an analog filter R and C by using bilinear transform, as shown in Eqn. 12 and Eqn. 13,

$$\alpha = R - \frac{T_{REF}}{2C}, \tag{12}$$

$$\rho = \frac{T_{REF}}{C}, \tag{13}$$

Therefore, when the gain of the TDC cancels out with the gain of the DCO [57], the open-loop function $H_{ol}(z)$ and close-loop transfer function $H_{cl}(z)$ of a 2nd-order

type-II ADPLL are demonstrated in Eqn. 14 and Eqn. 15, in which N is the frequency division.

$$H_{ol}(z) = \frac{\alpha(z-1)+\rho}{(z-1)^2} \tag{14}$$

$$H_{cl}(z) = N\frac{H_{ol}(z)}{1+H_{ol}(z)} = N\frac{\alpha(z-1)+\rho}{(z-1)^2+\alpha(z-1)+\rho} \tag{15}$$

Based on the translation between z-operator and s-operator shown in Eqn. 16, the linear s-domain approximation of ADPLL z-domain model is constructed[52]:

$$z = e^s \approx 1 + \frac{s}{f_R}, \tag{16}$$

in which f_R is the sampling rate of the system, which is usually the reference clock frequency f_{REF}.

Therefore, in the s-domain, Eqn. 15 becomes

$$H_{cl}(s) = N\frac{\alpha f_R s + \rho {f_R}^2}{s^2 + \alpha f_R s + \rho {f_R}^2} \tag{17}$$

Therefore, the damping factor ζ and the natural frequency ω_n of the 2nd-order type-II loop are shown in Eqn. 18 and Eqn. 19.

$$\omega_n = \sqrt{\rho} f_R \tag{18}$$

$$\zeta = \frac{\alpha f_R}{2\omega_n} = \frac{1}{2}\cdot\frac{\alpha}{\sqrt{\rho}} \tag{19}$$

When the gain of the TDC (k_{TDC}) is not normalized against the gain of the DCO (k_{TDC}), Eqn. 18 becomes Eqn. 20 and Eqn. 19 stays the same.

$$\omega_n = \sqrt{\rho k_{DCO} k_{TDC}} f_R \tag{20}$$

Even though few studies were found on phase-domain modeling of ADPLLs with frequency-based linear PDs and bang-bang ADPLLs [55][58], the modeling of the DLF module, which is the main control for the loop dynamics, provides insights on the loop behaviour in the formerly mentioned two implementations of ADPLLs.

Similar to an A/MS PLL, phase noise and locking time are important performance parameters for ADPLL frequency synthesizers. When used in a communication system, low-quality phase noise will reduce the effective signal to noise ratio, cause large bit error and reduce effective data rates. Therefore, phase-domain models are also utilized to investigate the relationship between phase noise performances and ADPLL variables, such as TDC resolution, bit-width of different digital units, DLF coefficients and DCO resolution [52][59][60].

Time-Domain ADPLL Modeling

In addition to phase-domain models, time-domain models for ADPLLs were proposed in [61][62]. The proposed time-domain models in the literature use behavioral model for each sub-block of the ADPLL realized using high-level programming languages such as Matlab, C, or SystemC to accurately predict the transient, steady-state, and phase-noise performance of the ADPLLs with minimal run-time complexity.

However, the models are presented in the form of pseudocode to achieve the level of modeling accuracy which tends to obfuscates designers by all the detailed functions in the ADPLL topology. An analytical model is lacking to provide designers with decent modeling accuracy for system response, but also allows designers to locate the key loop design parameters to tune when design for radiation environment.

ADPLL vs CPPLL: Application and Limitations

Charge-pump PLLs (CPPLLs) have been popular for over three decades due to architectural simplicity and ease of meeting the most demanding performance requirements. Until about a decade ago, almost all high-performance PLLs incorporate the charge-pump architecture[16].

However, when implementing a conventional CPPLL in advanced nanoscale CMOS technologies, the following points make it less appealing comparing to older technologies:

(1) Due to the low supply voltage constraint and poor drain dynamic resistance of MOS transistors, it is very difficult to implement near-zero input resistance for the current sources at advanced technology nodes.

(2) Due to the MOS gate leakage, it is difficult to use high-density MOS varactors. At the same time, on-chip metal-to-metal capacitors consume large area. In addition, external capacitors are typically acceptable adds extra I/O interface, routing and signal integrity issues. Therefore, capacitors for the loop filters in the CPPLLs are difficult to integrate.

(3) Ensuring wide linear tuning range of a VCO is very difficult in low-voltage technologies [63].

(4) CPPLL has rather a very limited bandwidth [64].

On the other hand, ADPLL architecture does not significantly affected by the issues addressed above. Besides, ADPLLs are more flexible and precise with the digital circuitries to meet the diverse and strict requirements of advanced systems-on-chip (SoCs). However, in ADPLLs, quantization in all of the digital control modules can lead to phase noise increase. Digital activity for switching the DCO produces

harmonic frequencies nearby the DCO resonant frequency, which can also create spurs [65].

Since 2000, many successful ADPLLs have been presented both in academia and industry. ADPLLs have started to be implemented in heating control system that used to rely heavily on CPPLLs[66]. ADPLLs have been applied in industrial, scientific and medical (ISM) band applications in wireless systems such as WLAN, Bluetooth and Zigbee where low-power is required: ADPLL is used for FM demodulaton [67], with FSK decoder in digital communications [68] and for wide-band frequency tracking and noise reduction in [69]. ADPLL-based mobile phone applications were developed in [6]. And ADPLL is used in high-speed clock generation [70] [32], clock recovery circuit [57] [71] and in frequency synthesizers [43][72][73][74][75].

In fact, there are many active on-going research topics involving ADPLLs as well. Extensive research is conducted with a focus on proposing novel modular designs of ADPLLs, such as DCOs[36], to improve overall loop phase noise performance. Some other studies focus on targeting specific PLL performance. Design specification comparisons between state-of-art ADPLLs and CPPLLs and the achieved targeting performance are listed in Table 1. The table shows the areas where CPPLLs or ADPLLs are competitive. Essentially, frequency synthesis at mm-Waves is still dominated by analog PLLs. However, ADPLLs are showing advantages in targeting design specifications, such as low power, wide tuning range and fast locking time. It is also worth noting that a number of novel CPPLLs utilize ADPLL techniques to achieve higher bandwidth and wide tuning ranges[76].

Table 1: Design specification comparisons between state-of-art ADPLLs and CPPLLs

Targeting Performance	ADPLLs	CPPLLs
High Frequency	60 GHz[77]	>160 GHz[78]
Low Power	78 μw @ 480 MHz[79]	6μw @ 2.5 GHz[80]
Low Jitter (RO)	1.25 ps [81]	0.4ps@ 2.5 GHz[82]
Low Jitter (LC)	145 fs @ 4.4-7.2GHz[83]	0.15 ps@ 2.21 GHz[84]
Wide Tuning Range	1 GHz-15 GHz [85]	35 GHz-41.88 GHz[86]
Wide Bandwidth	3.4 MHz[87]	5.5 MHz[88]
Fast Locking Time	2 cycles[31]	<10 cycles[89]
Reported Technology	16 nm FinFET[81]	45 nm SOI [82]

CHAPTER III

SINGLE-EVENT EFFECTS IN INTEGRATED CIRCUITS

Single-event effects (SEEs) are circuit behaviors resulted from ionized free changes generated from single particle incidence, as shown in Fig. 18. The primary particles of concern in the space environment are protons, heavy ions, alpha particles, and electrons. Typically, these particles are a result of cosmic ions, solar flares, products of secondary interactions, or from a natural radiation decay [7]. Some SEEs, such as single-event burnout (SEB) or single-event gate rupture (SEGR), are permanent or hard errors. Most commonly observed SEEs in CMOS circuits are transient effects, which are the main focus of this dissertation. In analog circuits, SEEs can cause unwanted noises, while in digital circuits, SEEs can result in direct data corruption, i.e. turning a "1" into a "0", or vice versa. In this chapter, the basic SEE mechanisms are discussed in section III.1. Section III.2 briefs on a literature overview of single-event transients (SETs) and single-event upsets (SEUs) in ICs. In addition, section III.4 details previous work on SEEs in phase-locked loops.

Single-Event Mechanisms

As an energetic particle passes through the semiconductor material, carriers are generated through coulombic interaction (i.e. direct ionization) or indirectly through nuclear reactions with the lattice (i.e. indirect ionization). In direct ionization, electron-hole pairs (EHPs) are created along the particle?s tracking path until it has lost all its energy or left the semiconductor[90] as shown in Fig. 18[91] .

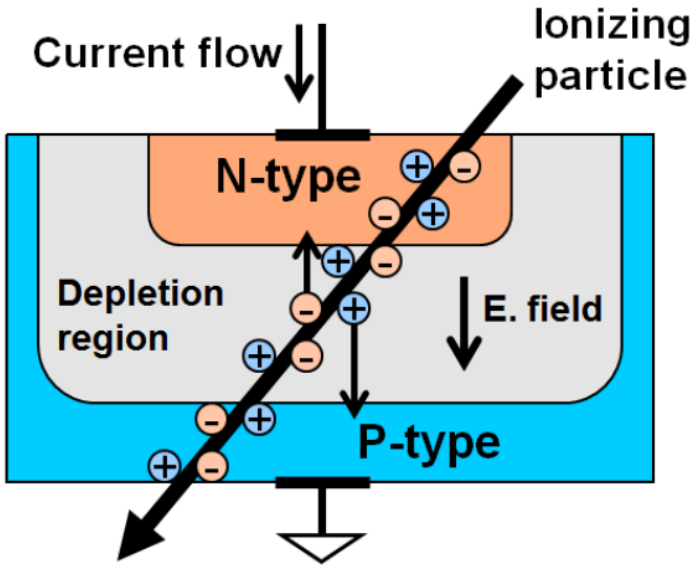
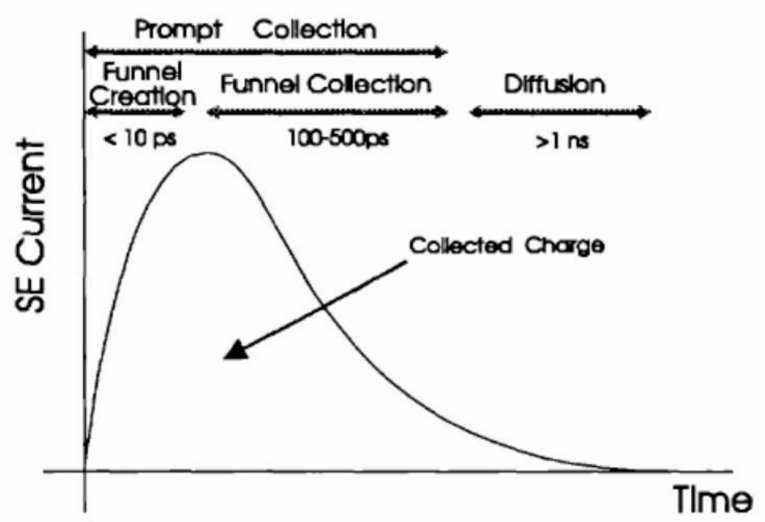


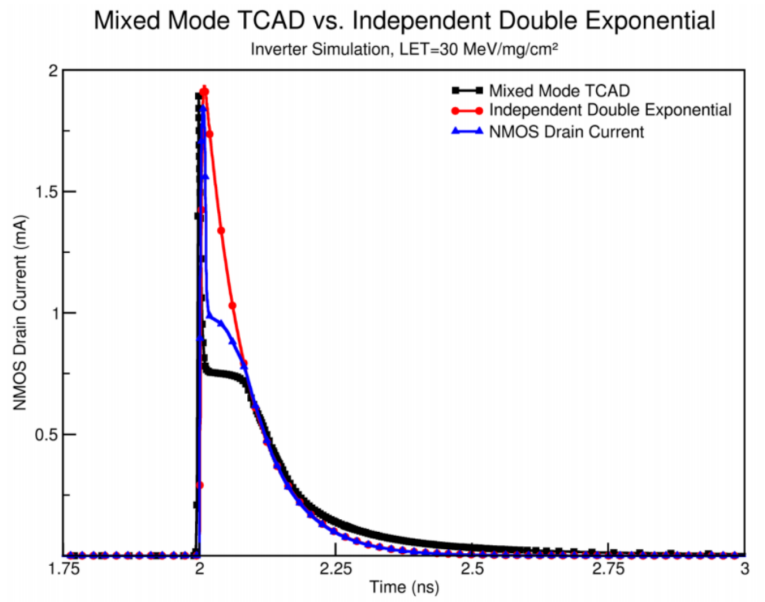
Figure 18: Collection of charge deposited by heavy ion in a reverse-biased junction [91].

The linear energy transfer (LET) is the metric used to define the energy loss per unit path length of the energetic particle passing through the semiconductor and has units of MeV·cm^2/mg.The LET of a particle can be easily associated to its charge deposition per unit path length [7]. As a reference, in silicon, a charge deposition of 1pC/μm corresponds to an LET of 97 MeV·cm^2/mg. This conversion factor of about 100 MeV·cm^2/mg is very handy and frequently used to convert LET in charge deposition.

After charge is liberated into the reverse-biased p-n junction, the collection process is divided into drift and diffusion transport. Drift transport is a quick process on the order of picoseconds in duration. In this process, the carriers are limited only by their saturation velocity. During diffusion transport, which is on the order of microseconds, charge deposited within a diffusion length of the junction can be collected and contribute to the voltage transient at the node. Charge collection



(a)



(b)

Figure 19: (a) Typical shape of the SE current at a junction. The total collected charge corresponds to the area under the curve [90] and (b) Comparison of NMOSFET drain current in TCAD mixed-mode and SPICE simulation of an inverter, where the SPICE simulation used an independent current source to model the single-event pulse [92].

may occur in multiple nodes depending on the size of the diffusion length and the spacing of transistors [93]. The two mechanisms are illustrated in Fig. 19a. The shape of the current pulse in Fig. 19a is the direct result of the charge collection mechanisms discussed above. The current spike is due to the prompt collection of charge via drift whereas the tail part of the current pulse is due to the diffusion induced charge collection. Finally, the total charge collected by the node corresponds to the integral of the current over the total duration of the single event. If the SE charge is deposited in a simple block of silicon, it will eventually recombine and equilibrium will be restored. However, if the charge is deposited at or near a p-n junction, then separation of charge carrier types, collection of this charge in different semiconductor regions, and propagation to the device terminals occur and a single-event effect is resulted. In more advanced technology the originated single-event current spike will be followed by a plateau effect resulting from the circuit load, as shown in Fig. 19b.

Single-Event Transients (SETs) and Single-Event Upsets (SEUs)

In analog circuits, an analog single-event transient (ASET) can be quite long in duration and large in magnitude due to the large time constants from the large sizes of the transistors in the circuits. In addition, since the system is continuous, ASETs could be largely indistinguishable from a legitimate signal. Extensive research has been published on characterizing, classifying and mitigating ASETs [94][95][96].

In digital circuits, digital single-event transient (DSETs) can be latched by storage devices and be read out as incorrect data (SEU). Electrical masking [97][98], temporal masking[99], logical masking[100] and operational masking [101] are factors that affect the latching of the originating SET in the storage elements. Every storage device or

latch has a ?window of vulnerability?. The window of vulnerability is the period of time that determines whether the SET is latched or is not latched as shown in Fig. 20[102][103]. Fig. 20 illustrates how SET is latched within the window of vulnerability. In the 1st, 2nd and 4th cases, the SET is not latched because the transient occurs outside of the sampling time of the latch. Only in the 3rd case, the SET is latched in the storage element. As frequency increases, the SET pulse width remains but the probability of latching SETs increases because of more frequent occurrence of time vulnerability window [104]. Therefore, SETs start to dominate chip-level single-event error rates at high frequencies.

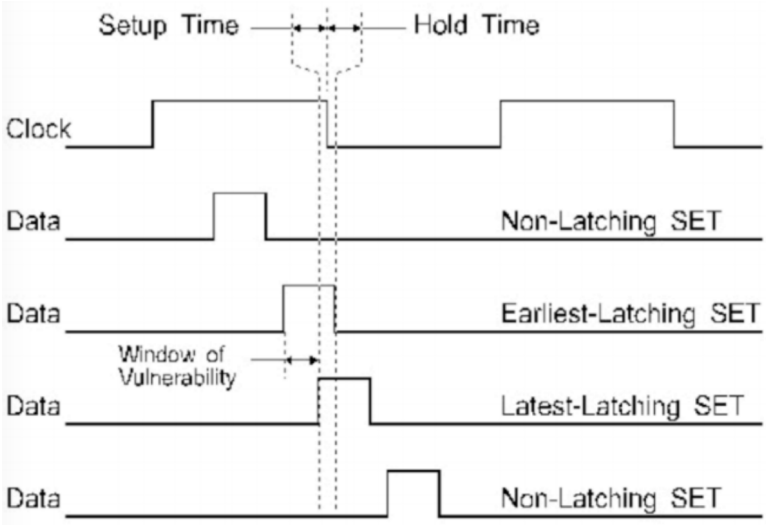


Figure 20: An illustration showing how a pulse may or may not be latched by a storage [91].

Prior Work Related to SEEs in PLLs

While limited work has been published regarding the SE vulnerability of ADPLLs, extensive work regarding SEE characterization, modeling, and mitigation in A/MS PLLs has been published. Due to structural and functional similarities between

ADPLLs and A/MS PLLs, understanding prior work on A/MS PLLs (mainly charge-pump PLLs) facilitates in understanding the SE vulnerability of ADPLLs. In fact, the single-event model and hardening guidelines proposed for ADPLLs in this work are compared with previous work on CPPLLs in Chapter VIII to provide conventional PLL designers with a comprehensive understanding of RHBD PLLs. In this section, subsection III.3.1 details previously reported single-event characterization results and mitigation techniques of A/MS PLLs. And subsection III.3.2 focuses on modeling of SETs in CPPLLs.

Single-Event Characterization and RHBD Techniques for CPPLLs

Since ASETs are the main concerns in SEEs for A/MS circuits like CPPLLs, ASETs are commonly simulated by applying a current source representing the radiation-induced photocurrent at the perturbed voltage node. Through SET simulations and radiation experiments, the single-event vulnerability of conventional CPPLLs to single-particle strikes was reported to be dominated by the SET response of the charge pump module[11][105]. In a conventional current-based charge pump, active devices are directly connected to the source, ground, and a low-pass filter. By replacing that with a SET-resistant tri-state voltage-switching charge pump [105], the removing speed of the charge stored on the struck node is no longer limited by the driving capability of the current source. The SET induced phase displacement is reduced by approximately 2 orders of magnitude[105]. However, the topology makes the PLL more susceptible to noise from the voltage sources. The jitter performance was shown to be 10 times worse than a conventional CPPLL when the operating speed is at 400 MHz and increasing sensitivity with operation frequency was predicted.

Other than the charge pump, most efforts have been focused on radiation hardening VCOs. A way to radiation hardened a VCO by introducing redundancy into the biasing stage and optimizing the number of stages of ring-oscillator is proposed in [106]. The proposed technique for both charge pump and VCO combined was shown to reduce the output phase displacement by 66% [107]. The VCO design proposed by [108] utilized two current starved ring oscillators, whose virtual ground node is controlled separately. The internal signals of any rings are connected to the other ring as well, resulting in a situation where one ring negates a radiation strike in the other ring. After a radiation particle incidence, the PLL design with this hardening approach exhibited an overall worst-case jitter performance of 18.7% and a fast locking time[108]. This hardening technique can result in higher immunity towards variation and noise of voltage source but bigger areal penalty compared with the PLL proposed by Loveless et al [12].

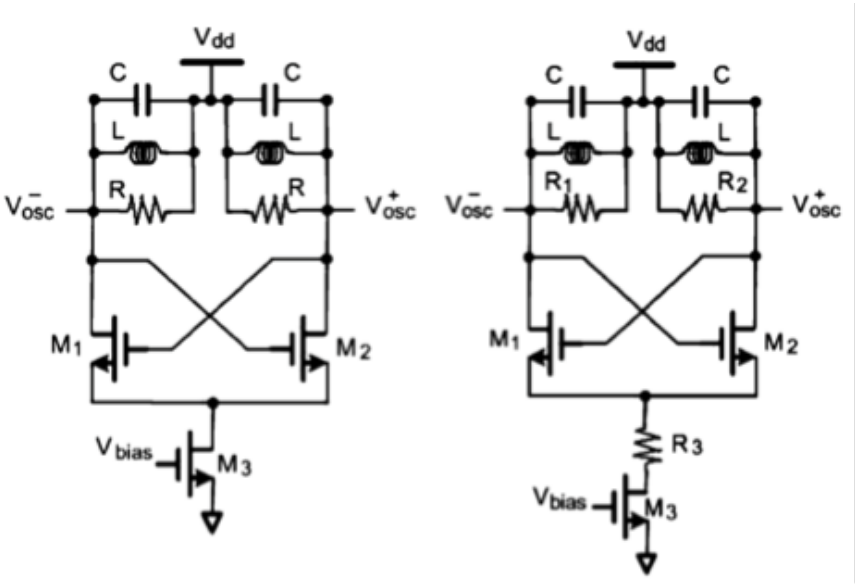


Figure 21: Original LC-Tank VCO topology is shown on the left, while the right figure is the schematic of the RHBD VCO with a decoupling resistor R3 [109].

Compared to a RO, an LC-tank oscillator requires more area and consumes more power but has outstanding phase noise and jitter performance even at very high frequencies [109]. T. Wang et al. showed the vulnerability of active devices to single event strikes, i.e. transistors, in one topology of LC-Tank oscillators, shown in the left figure of Fig. 21. An RHBD method of introducing a decoupling resistor between the oscillating stages and the biasing stages, as in the right figure of Fig. 21, was proposed in the paper and proven to be effective with experimental results.

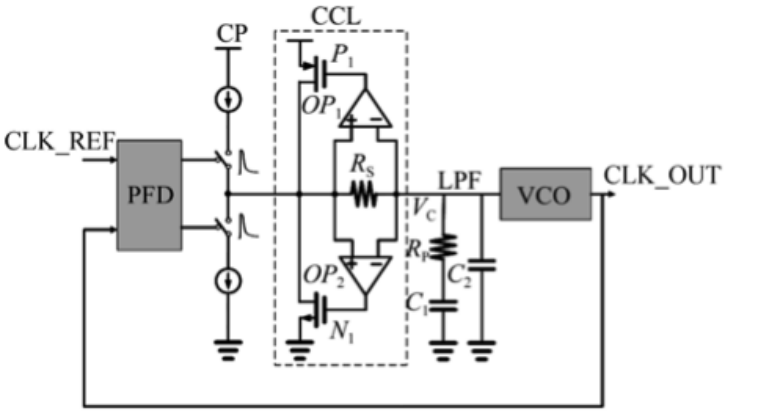


Figure 22: Single-event hardened PLL with the complementary current limiter (CCL)[110].

The above approaches are effective in mitigating the effects that SET imposed on PLLs. Nevertheless, the design of PLLs becomes more complex, as the charge pump, LPF or VCO should be re-designed. [110] presented a hardening technique of utilizing a SET-resistant complementary current limiter (CCL) between the charge pump and VCO, which can result in minimal impact on the PLL loop parameters and design flow. Fig. 22 illustrates the structure of the CCL. Based on the SET failure mechanism, the proposed CCL, which is composed of a sense amplifier and a resistor, is active only when the SET current from CP is greater than $2I_P$ ($2I_P$ is a margin

to keep the CCL circuit from falsely switching), thus mitigating the effect the SET current imposes on the control voltage of VCO. Simulation results showed that this RHBD technique was able to reduce the voltage perturbation on the input of the VCO, either induced by single-event strike, or loss of lock due to phase or frequency shift in general, by up to 93.1% and reduce the recovery time by up to 79.0%[110].

Modeling of SETs in CPPLLs

H. H. Chung et al. investigated the behavior of a CPPLL if an SE strike at the output of PFD/CD resulted in VCO control voltage perturbation. The overall model of CPPLL for SET characterization is shown in Fig.23[111], where K_{pfd} is the gain of the PFD, which includes the PFD and CP. $F(s)$ is the transfer function of loop filter (integrator), K_{vco} is the gain of the VCO, and 1/N is the feedback factor. K_{pfd} has units of volts/radian or amps/radian (dependent on the type of PFD) and K_{vco} has units of radians/(second·volt). The SET induced current is presented using an ideal rectangular current pulse with the pulse with the pulse height of I_{pulse} and the duration of time T. For an integer divider value N, when the PLL is locked the error response is zero. Therefore, any disturbance inside the loop can always be presented as an error signal, which excites the PFD/CP as well as the control voltage V_{ctrl}. The transfer function of the PLL error response (φ_e) in terms of the single event strike current (I_{in}) is derived below in Eqn. 21.

$$\frac{\varphi_e(s)}{I_{in}(s)} = \frac{K_{vco}F(s)}{Ns + K_{pdf}K_{vco}F(s)} \tag{21}$$

By plugging the SET current Laplace profile into the equation above, the author

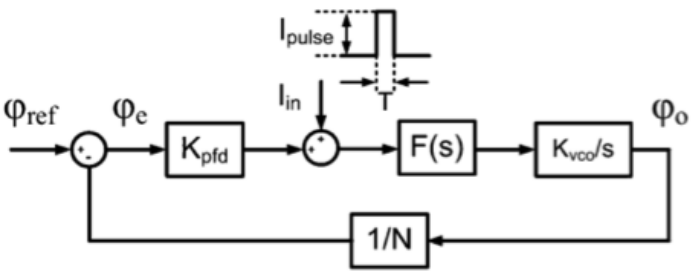


Figure 23: Overall model of a CPPLL for SET characterization [111].

obtained the uniform time domain response of error response of the PLL.

$$\varphi_e(s) = \frac{I_{pulse}}{K_{PFD}}(1-e^{-\xi\omega_n t}(cos(\omega_n\sqrt{(}1-\xi^2))t-\frac{\xi}{\sqrt{(}1-\xi^2)}sin(\omega_n\sqrt{(}1-\xi^2))t)(u(t)-u(t-T)) \tag{22}$$

where the natural frequency ω_n and the damping ratio ξ are:

$$\omega_n = \sqrt{\frac{I_P K_{VCO}}{2\pi C_l}}, \xi = \frac{R_P}{2}\sqrt{\frac{I_P C_l K_{VCO}}{2\pi}} \tag{23}$$

where I_P and R_P are the resistance and current of the CP respectively. C_l is the capacitance of the LPF. A conclusion both from theoretical analysis and experimental results was that the settling time is proportional to the peak control voltage deviation VC_{MAX} and the SET pulse width T. The authors also showed that the frequency disturbance caused by a single event strike increases as the PLL bandwidth increases [111]. Using similar approaches, Z. Zhao et al. found the settling time is also inversely proportional to the settling time constant $\xi\omega_n$. And VCMAX is dominated by the SET pulse amplitude I_{pulse} and the loop filter resistance R_P [112].

Loveless et al. modeled SE hits in VCOs by two current sources (I_D) representing the restoring device current in a current-starved inverter, an output node capacitance (C), and a current source representative of the current induced by the SE (I_{hit}), as

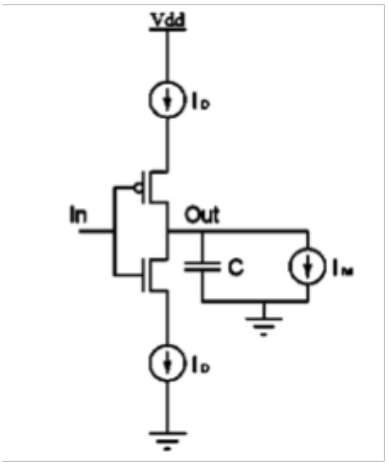


Figure 24: A simple model for a SE hit in a current-starved inverter can be presented by two current sources (I_D) representing the restoring device current in a current-starved inverter, an output node capacitance (C), and a current source representative of the current induced by the SE (I_{hit}) [106].

shown in Fig. 24[106]. In the paper, phase displacement (Φ_{disp}) was first presented as a measure of the severity of SET resulting from ion strikes in the CPPLL. An SET propagation model for CPPLLs was proposed later in [107] based on conventional phase-domain transfer functions of linear PLLs for noise analysis. Through the proposed analytical model, the output phase displacement of the CPPLL (ϕ_{disp}) due to SETs in different sub-circuit was presented, in units of radians. The output phase displacement can be presented as a function of the PLL critical time constant (τ_{crit}), as shown in Eqn. 24,

$$\tau_{crit} = \frac{\beta f_{lock}}{{\omega_n}^2}, \tag{24}$$

in which τ_{crit} is defined as the minimum time constant of the initial perturbation required to maximally disturb the closed-loop PLL and t_{rec} is the ideal recovery time of the loop. As shown in Eqn. 25,

$$\phi_{disp} = \frac{2\pi {\omega_n}^2 t_{rec}}{\beta f_{lock} \pm {\omega_n}^2 t_{rec}} = \frac{2\pi t_{rec}}{\tau}_{crit} \pm t_{rec}, \tag{25}$$

ω_n is the PLL's natural frequency, β is the feedback factor, and f_{lock} is the steady-state output frequency in phase lock. This vastly simplifies the analytical analysis of SETs in CPPLLs, which allows designers to make tradeoffs on design parameters to meet both electrical and radiation-hardness requirements. Critical time, i.e. the recovery time from upset, is defined as the amount of time it takes for I_{hit} to be larger than I_D.

CHAPTER IV

ADPLL MODULAR SINGLE-EVENT (SE) CHARACTERIZATION AND ANALYSIS

SE characterization of ADPLL was carried out in order to understand the behavior of different types of ADPLLs operating in the radiation environment, which is essential for developing RHBD solutions. Different modules in the ADPLLs are susceptible to different types of SE-induced errors. Therefore, SE characterization of ADPLL is divided into two portions based on the types of errors. Shown in Fig. 25, as stated in Chapter II, f_{REF} is often less than 500 MHz generated from crystal or quartz oscillators[5][113][114]. Since PD and DLF operate by the reference clock, SEU-induced errors are the main concerns. Similarly, the FD and other global registers consist of mostly storage elements, while DCO, FREF, and RST trees consist of only logic elements. Therefore, PD, DLF, FD and other global registers (global reset RST and oscillator enable ENABLE) are SEU-sensitive modules (indicated with light yellow color), while DCO, FREF, and RST trees are prone to SET perturbations (indicated with red color). Since SETs from RST and CLK tree affects digital systems in complex ways and can be hardened by sizing the buffers in the distribution network bigger, they are not discussed in the scope of this work.

In this chapter, section IV.1 discusses the observed single-event error signatures and proposed error metric. The SET characterization of DCOs in ADPLLs is completed and presented in Section IV.2. Section IV.3 goes into details about the SEU characterization results on the PD and DLF. The overall SE characterization of

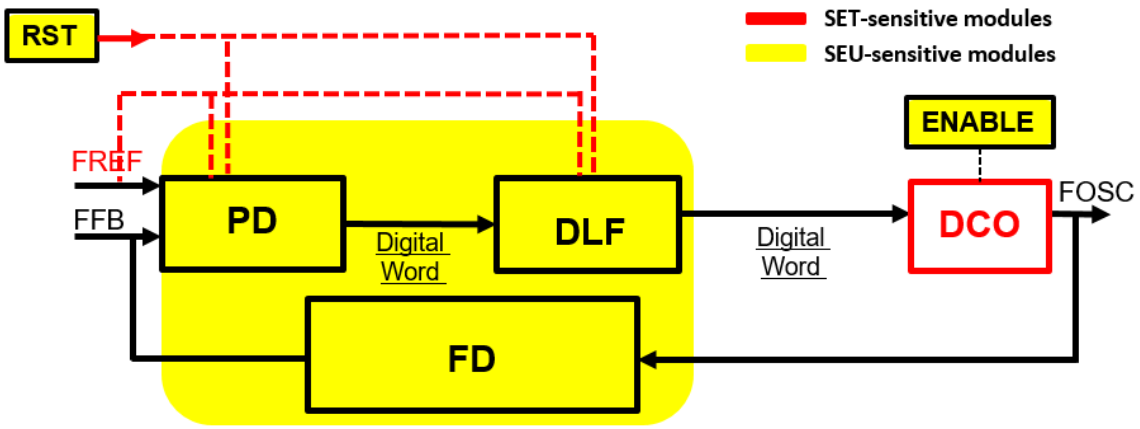


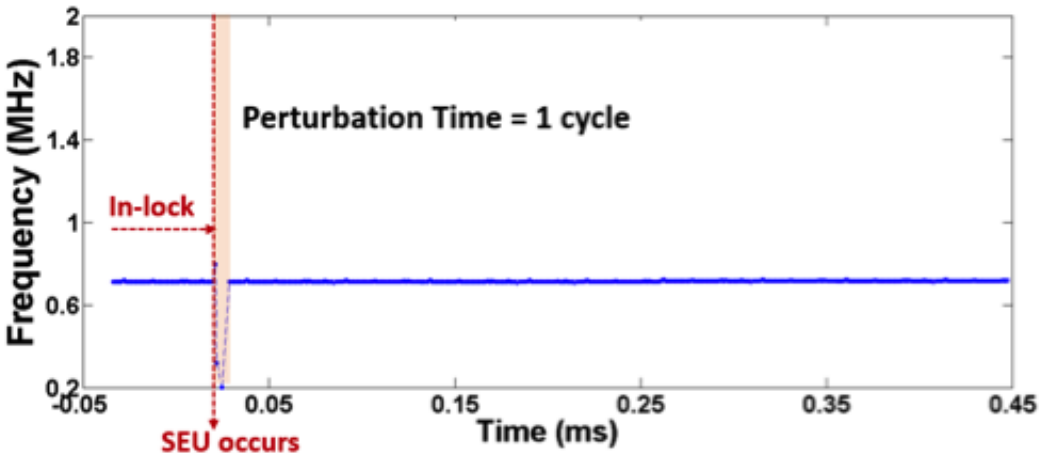
Figure 25: SEU-sensitive modules and SET-sensitive modules in an ADPLL.

common 1st and 2nd-order ADPLLs is completed and presented in Section IV.4.

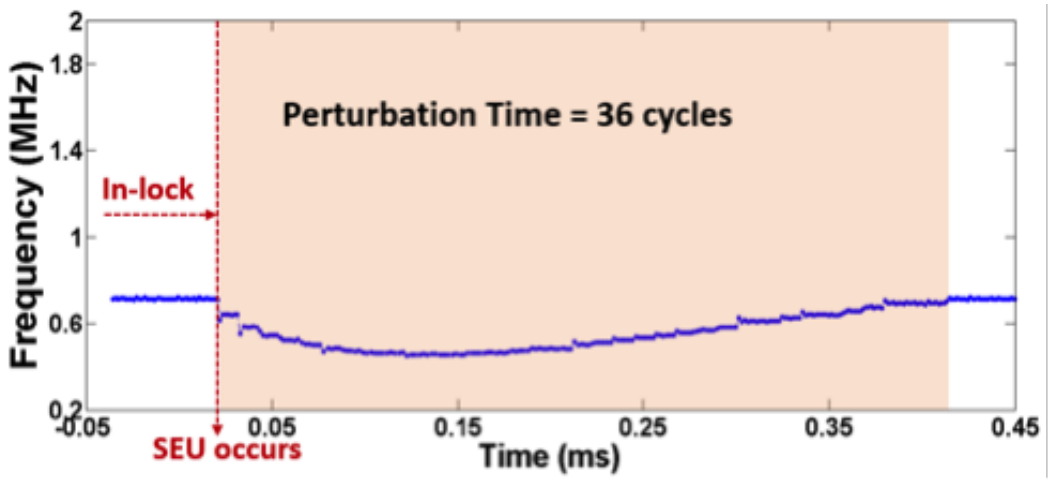
Error Signatures and Error Metric

Three primary types of errors signatures following an single-event perturbation in ADPLLs have been observed: (1) loss-of-lock errors, (2) temporary-frequency errors and (3) limit cycle error [115]. An error metric - perturbation time metric - was proposed to quantify the severity of SE-induced errors[116]. The perturbation time is defined as the time period when the ADPLL output frequency is outside of the original jitter tolerance due to the SEU occurrence. And perturbation time is quantified in terms of reference clock cycles.

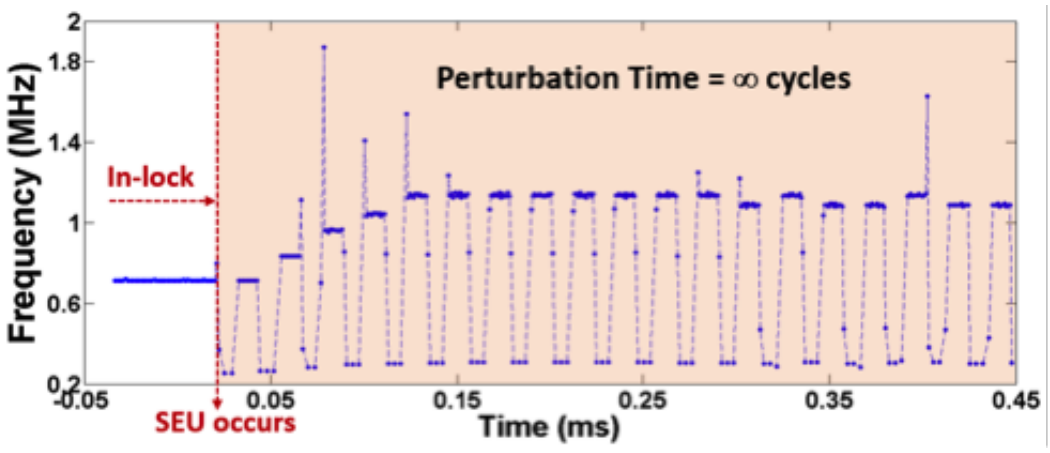
An example of the three error types is shown in Fig. 26 [117] where the ADPLL output frequency is plotted versus time. The ADPLL is in-lock within the period with a relatively constant control word versus time. Temporary-frequency errors (Fig. 26a) and loss-of-lock errors (Fig. 26b) have been analyzed previously in [116]. Temporary-frequency errors are not able to cause loss of frequency lock, because they only last for a time period that is shorter than the bandwidth of the ADPLL. Limit cycle error



(a)



(b)



(c)

Figure 26: SEU signature transient waveforms in terms of ADPLL output frequency plotted over time[117].

occurs when the ADPLL engages in frequency and phase tracking but never locks, resulting in output frequency oscillating spanning over the frequency tuning range. Limit cycle error often requires system reset if the oscillating persists.

Essentially, limit-cycle errors are the results of non-linearity in the system[118], in which the design keeps looping from frequency f_0 to f_m and back with fixed steps every time from δf_0 to δf_m. Therefore, the ADPLL stays in the loop forever until another perturbation happens, such as noise or system reset.

When a large perturbation is introduced to a high-order ADPLL by perturbing the bits in the register that corresponds to the system poles directly or indirectly, the system damping ratio is changed, which leads to an oscillating behavior (underdamping), as shown in Fig. 26c. The relationship between other registers and the pole register is design dependent; however, the more abrupt the damping ratio changes, the longer time it takes the system to correct the error.

Even though in systems driven by PLLs with large frequency division factor M, temporary-frequency errors can still pose a threat, as the PLL output is deviated from the desired frequency for at least 1 reference clock period, i.e. M oscillation period. Temporary-frequency errors are still considered less severe comparing with the other two types of errors.

ADPLL SET Sensitivity Analysis and Verification

For advanced technologies, the error rates for combinational logic have begun to dominate the overall single-event error rate (SER) at high frequencies[119][120][121][122]. This is because shrinking of feature sizes of transistors lowers the charge requirements to represent a logic HIGH state (resulting in higher number of SETs). And at higher

frequencies, increased number of clock edges makes it more possible for SETs to be latched. SET sensitivity of the DCO and other digital blocks in ADPLLs are discussed in this section.

SET Sensitivity Analysis of DCROs

SET-induced Error Signatures in Standalone DCROs

To characterize SET-induced errors in standalone DCROs, a 13-stage DCRO was designed in the UMC 40 nm Bulk CMOS process comprised of inverters of varying size, multiplexers, and a NAND gate, as shown in Fig. 27. This design uses coarse and fine frequency tuning schemes. Coarse tuning is achieved by using a control word (digital code) to select different feedback paths through a multiplexer to adjust the output frequency. Fine-tuning is achieved by using a control word (digital code) to select different capacitive loads from the capacitor bank in the fine-tuning cell. A fixed delay block is utilized to set the center frequency.

Single-event transient (SET) simulations and TPA laser experiment were performed on the DCRO designs using ion-induced current profiles obtained from calibrated 3D TCAD models [92]. The current profile is injected to every node in the circuits over the clock period to determine the error signatures and worst case SET responses of the output oscillating signal. In addition, the SET simulation is conducted basing on the assumption of only one node in the circuit is collecting charges at one time.

When a single-particle perturbs the actual oscillator, three different types of transient errors can occur from the resulted SET: erroneous (missing) pulses, duty

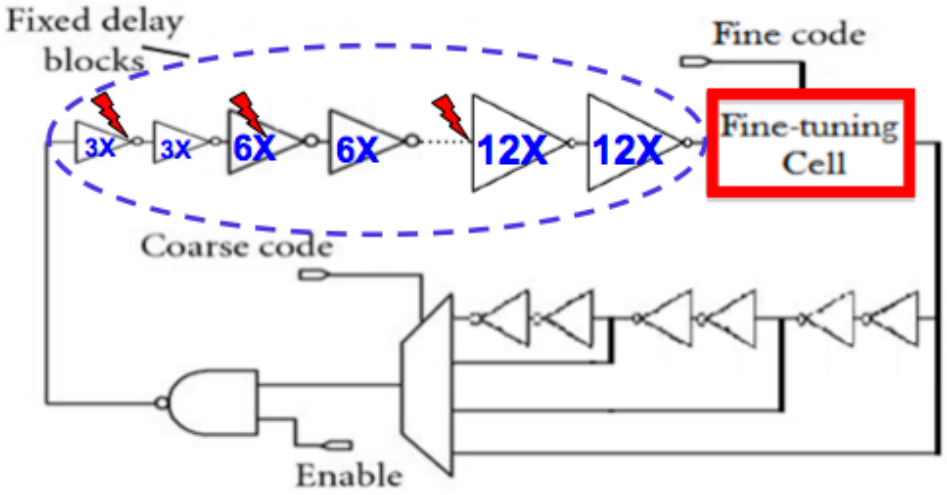


Figure 27: Block diagram of the DCRO design implemented in a 40 nm bulk CMOS technology[50].

cycle errors and harmonic errors, as shown in Fig. 28.

Duty-cycle errors refer to cases when the logic HIGH/LOW pulse widths differ from the original signal resulted from voltage perturbations. Similarly, missing pulses represent the case when one or more pulse(s) are absent from the output. These error signatures have been reported in clock circuits like CPPLLs and digital-locked loops (DLLs)[123][105]. As shown in Fig. 28, these types of transient errors are usually not persistent, but they can affect the original signal by shifting the signal in phase.

SE harmonic errors may be induced by SETs that result in the presence of one or more additional pulses occurring within the typical oscillation period, resulting in the oscillator operating at a harmonic frequency (usually an odd multiple of the original frequency), as illustrated in Fig. 28. The introduction of harmonic frequencies may result in system errors and/or synchronization problems and it is imperative to characterize, model, and develop mitigation schemes to address these errors.

A phase displacement metric has been used to quantify missing pulse and duty

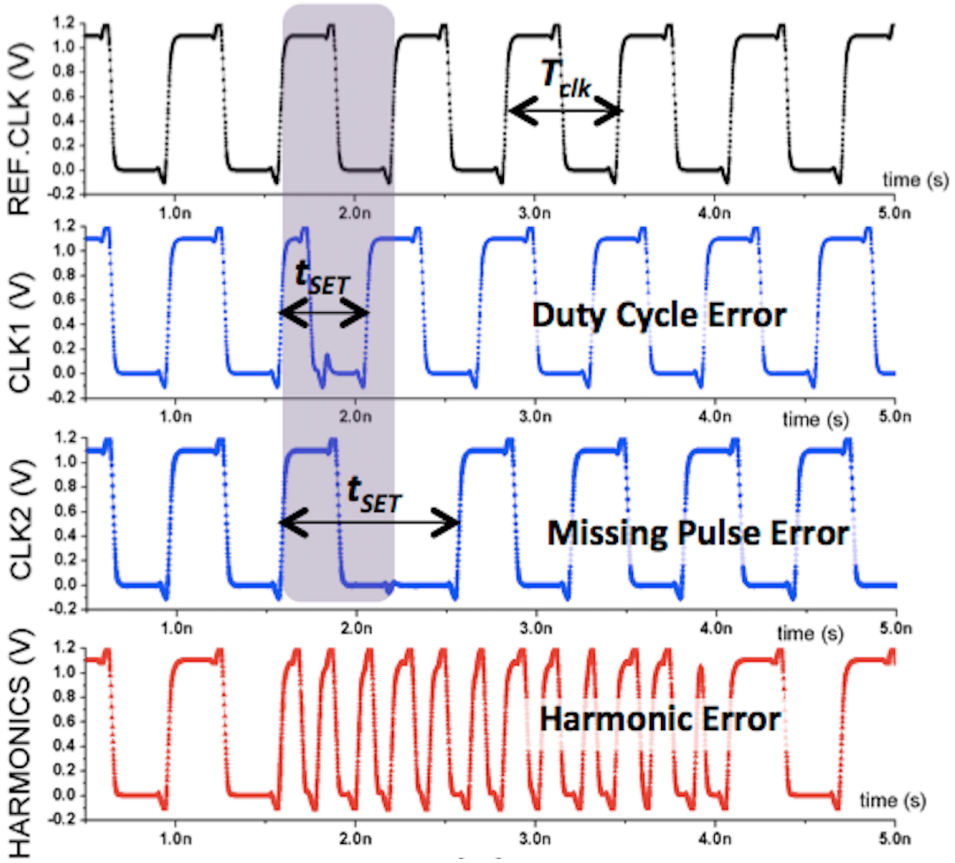


Figure 28: SET-induced duty cycle error (second), missing pulse error (third) and harmonic errors (bottom) in reference to the unperturbed clock (top)

cycle errors[123][105]. In the case of Fig. 29a, the phase difference between perturbed and unperturbed oscillation signal is cumulative in time. The accumulated phase error, ϕ_i, for i^{th} oscillation period, corresponding to δt_i(i^{th} perturbed oscillation period) is shown in Fig. 29b.

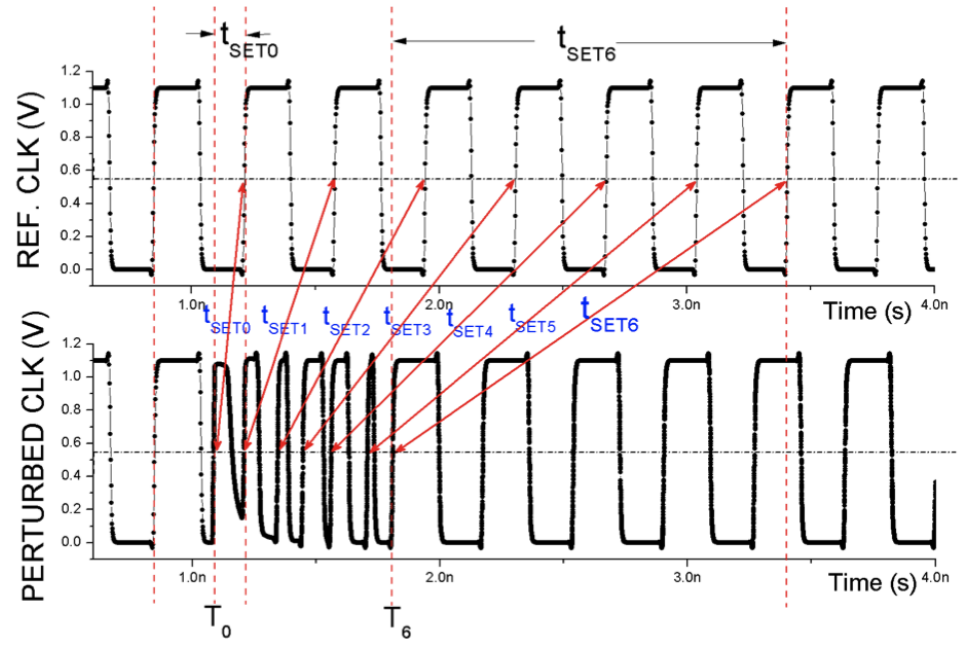
The perturbed signal shown in Fig. 29 eventually settles back to the original period, therefore the maximum accumulated phase difference (ϕ_{max}) resulting from a single particle hit is finite (and in this case is equal to 7π). A harmonic response is determined to have occurred when the accumulated phase difference exceeds 2π radians, indicating at least one additional erroneous oscillation period, thus appearing as the third harmonic. The harmonic response is always observed to eventually dampen due to parasitic and pulse stretching/broadening effects in DCROs.

The mathematical relationship between ϕ_i and t_{SETi} is illustrated in Eqn. 26,

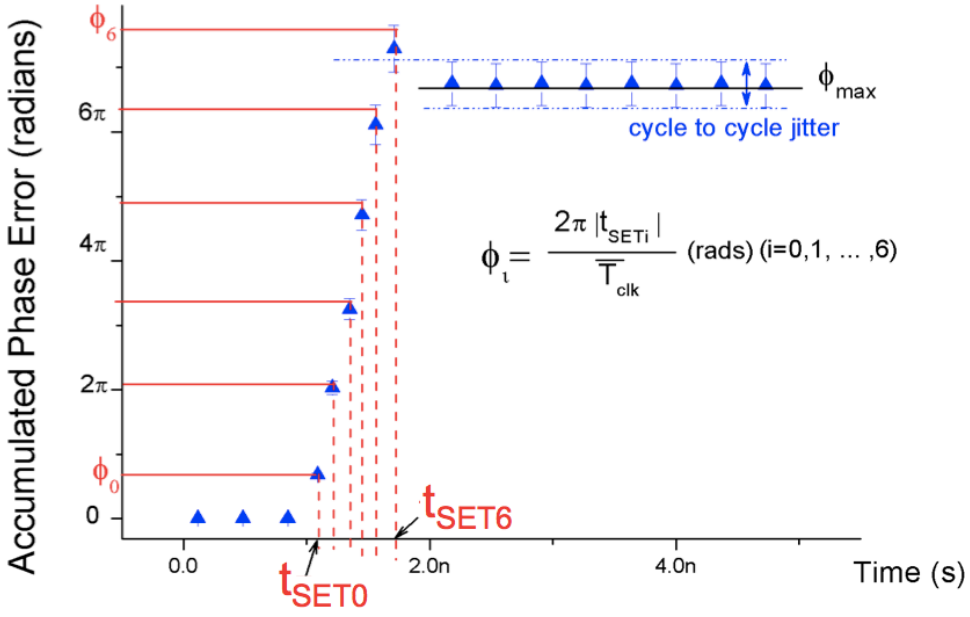
$$\phi_i = \frac{2\pi|t_{SETi}|}{T_{clk}}, (i = 0, 1, \cdots, 6) \tag{26}$$

in which T_{clk} is the average clock period without any perturbation and the absolute value of t_{SETi} is used. While harmonic errors would usually result in positive t_{SETi}, duty cycle or missing pulse errors could sometimes lead to negative t_{SETi}.

Laser-induced carrier generation for radiation testing based on two-photon absorption (TPA) has been demonstrated [124][125]. A 13-stage DCRO was tested using the TPA laser technique at Vanderbilt University with an estimated beam spot size of 1.2 μm^2. The laser system used in this work employs optical parametric generation (OPG) in a BBO crystal to convert 800nm, 120 fs pulses at a repetition rate of 1 kHz from a Titanium/Sapphire chirped-pulse amplifier (Titan, Quantronix, Inc) into signal and idler wavelengths. The OPG (TOPAZ, Light Conversions Inc) is turned to



(a) t_{SETi}



(b) ϕ_i

Figure 29: The time differences t_{SETi} between the rising edges of the perturbed and unperturbed clock (a) and phase differences ϕ_i corresponding to them (b).

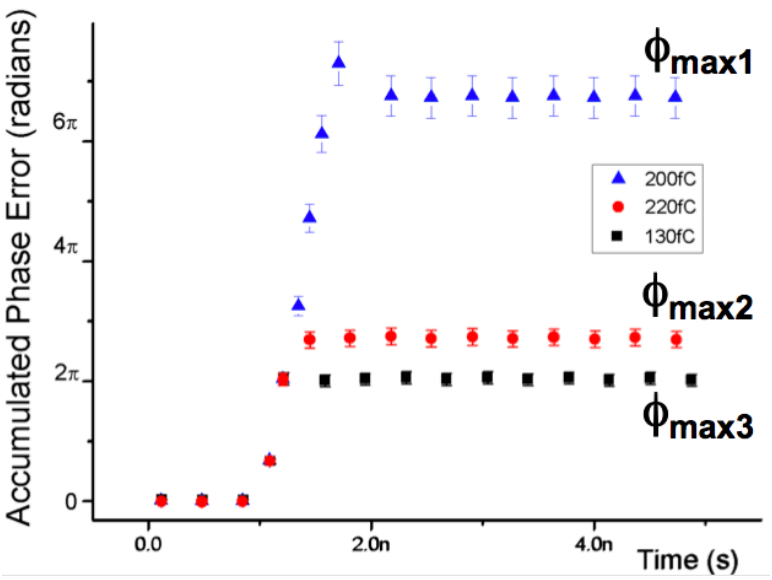


Figure 30: The maximum accumulated phase error for different collected charge values.

a signal wavelength of 1260 nm; available pulse energies at this wavelength can exceed 150 μJ [126]. The output nodal voltage and photodiode peak voltage were recorded using a high-speed Tektronix TDS6124C oscilloscope with a 12 GHz bandwidth and 25 ps sampling resolution. The test circuit was mounted in a custom-milled metal package with microstrip transmission lines and K-connectors. Biasing was supplied using a Keithley 2410 Source Meter. For all measurements, the device under test (DUT) was mounted on an automated precision linear stage with a minimum step size of 0.1 μm. The DCRO design was tested at 0.9 V, with a digital control word of all-0s set to digitally bias the RO to operate at around 1 GHz. All experiments were performed at room temperature.

Laser strikes were performed at one specific location of a specific gate in the DCRO with varied laser energies. The incident laser pulse was asynchronous from the oscillation period to enable strikes performed at the node spanning over the

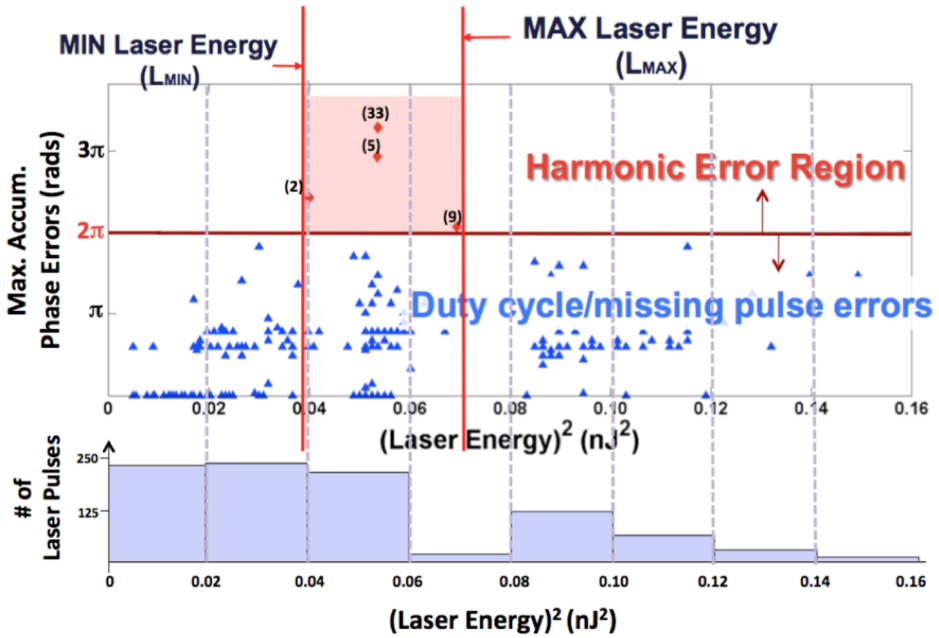


Figure 31: Measured maximum accumulated phase error versus laser energy squared for the DCRO design operated at 0.9 V.

oscillation period. The laser testing results are shown in Fig. 31. The top plot in Fig. 31 shows the measured maximum accumulated phase error over the squared energy of corresponding injected laser pulse. The bottom plot shows the number of laser strikes capable of inducing phase errors (during the half of oscillation period when the transistor is off), as a function of binned laser energies. The top figure shows that laser irradiation is able to induce harmonic errors (denoted by red diamonds in Fig. 31) and duty-cycle errors /missing pulse errors (denoted by blue triangles in Fig. 31) in the DCRO design. The numbers in parenthesis indicate the number of events observed at the same laser energy and same output accumulate phase error, i.e. error counts symbolized by the clusters of red diamonds. The harmonic errors are distinguished from the duty-cycle errors because they exhibit maximum accumulated phase errors (ϕ_{MAX}) of greater than or equal to 2π. And they are distinguished from

the missing pulse errors because they lead in phase rather than lagging in phase in the case of a missing pulse error. It is observed that only a window of laser energy from the minimum to the maximum ($L_{MIN} \sim L_{MAX}$) can induce harmonic errors for this logic gate in the DCRO. The bottom figure is provided to illustrate the statistical difference in the likelihood of observing harmonic errors in and outside of this laser energy window. From the experimental data, for the binned laser-energy-squared of [0.06 nJ^2, 0.08 nJ^2], the likelihood of observing harmonic errors is around 30% (9 counts out of 25 counts), while that for the binned laser-energy-squared of [0.08 nJ^2, 0.10 nJ^2] is close to 0 (0 out of 118 counts). The x-axis in Fig. 31 is in nJ^2 because for TPA laser experiments, carrier deposition is proportional to the square of with the laser pulse energy [124].

Since, as stated in [127][128], SET pulse width is proportional to incident laser pulse energy, the observed laser energy window corresponds to an SET pulse width window (harmonic vulnerability window[50]), which validates the harmonic analytical model. In addition, all duty cycle errors, missing pulse errors and harmonic errors are observed for the DCRO design.

DCO SETs Induced ADPLL Errors

As stated in Chapter IV, a 1st-order frequency-based linear and a bang-bang ADPLL was designed and synthesized using the IBMCS7RF standard cell library. Only PI filters with proportional path gain of 2^{-1} are used in the DLFs in the ADPLLs (i.e. no cascading FIR/IIR filters). The DCO used in both ADPLLs are ring-based designs using a 7-bit control word through a multiplexer to control the length of the DCRO thus controlling its output frequency. The gain of the DCO is in the range

of 1 $\sim$ 4 MHz/LSB. SET-induced errors for the DCOs in the ADPLL designs were studied by injecting SEE current sources using the ISDE bias dependent SEE model [92] at all the internal nodes of DCO over the oscillating period when the ADPLL is locked at the desired frequency. The output of the ADPLL was monitored to determine the SET-induced error signatures. Frequency division of 8 was used for all the SET simulations.

As shown in Fig. 32, when the ADPLL is locked at each locking frequency that corresponds to digital control word from 0-120 at interval of 20, the maximum perturbation cycles for SETs injected at all the internal nodes of DCO over clock period are plotted for both ADPLL with integer-based linear PD and bang-bang PD.

During the SET simulation, all duty-cycle, missing-pulse and harmonic errors were observed for SETs injected in DCOs in a closed-loop design. Duty-cycle and missing-pulse errors in DCROs manifest as temporary-frequency errors in ADPLLs due to its small phase deviation from the locking states. However, SET-induced harmonic oscillation result in different loop behavior for the two simulated designs.

In Fig. 32, for bang-bang ADPLLs, due to the time-based phase detection and limited frequency-tuning step sizes when reacting to the harmonic oscillation, the ADPLL only increments or decrements the digital control word for the DCRO for a few reference clock cycles and the harmonic oscillation deceases. Therefore, minimal perturbation, i.e. temporary-frequency error, is resulted at the ADPLL output. On the contrary, for linear ADPLLs, the integer-based linear PD counts the reacts to the changing of output frequency significantly more abruptly than bang-bang PDs, which results in longer frequency perturbation time and can induce loss-of-lock errors in ADPLLs.

In addition, as stated in previous work in [129], a mathematical equation has to be satisfied between the total loop delay of the ring and the SET pulse width for the sustaining of SE harmonic oscillation. In the simulated ADPLL design, changing the digital control word for the DCRO is equivalent to changing its loop delay. Extremely long harmonic-error-induced output perturbation time was observed at locking control word of 30 and 90. This is due to the accumulated frequency error input for the integer-based linear PD from harmonic oscillation over several reference cycles, which occurs only when the harmonic oscillation can be sustained according to the mathematical model in [129].

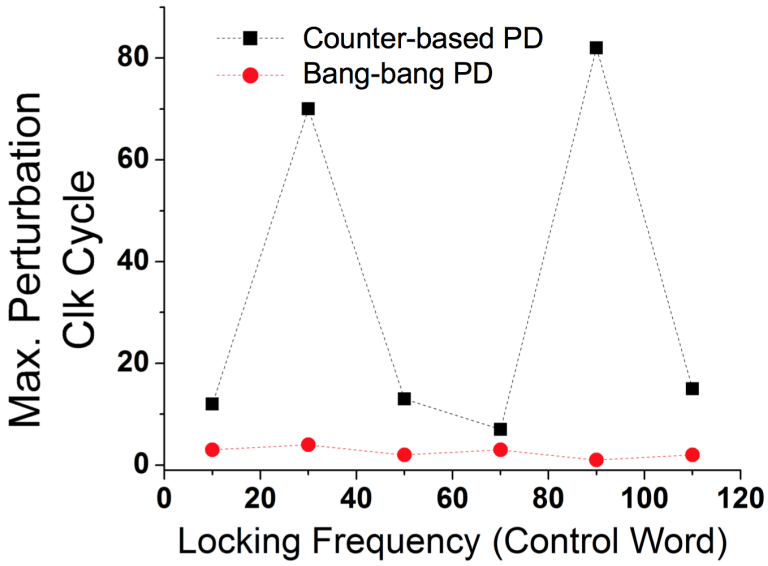
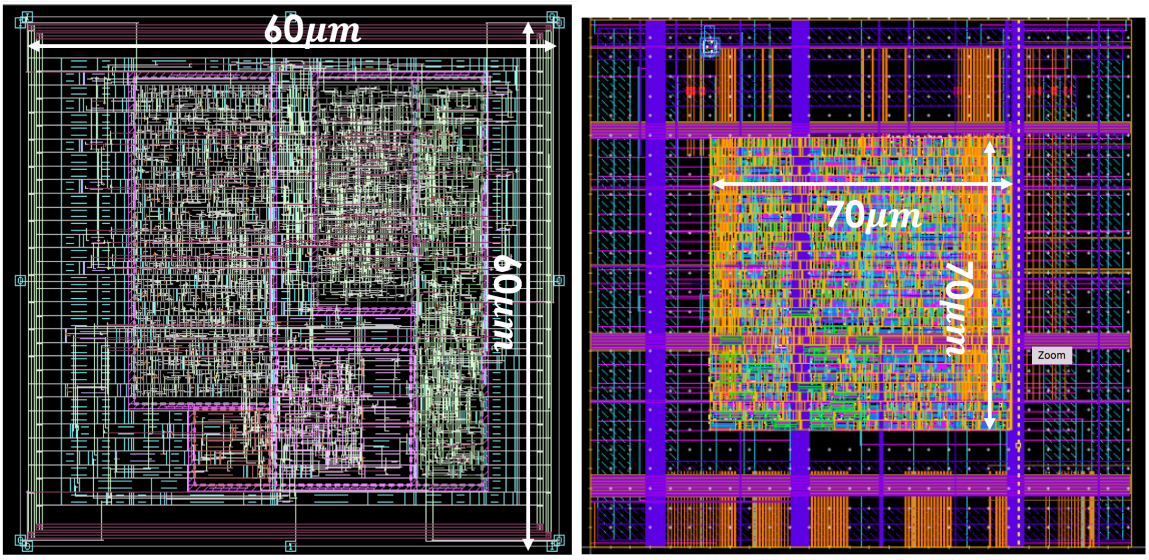


Figure 32: Maximum perturbation cycles for SETs injected at all the internal nodes of DCO over clock period are plotted for both ADPLLs with integer-based linear PD and bang-bang PD.

SET Sensitivity of Other Logic Blocks

It is stated previously that since the logic blocks PD and DLF operates at low frequencies (less than 500 MHz), SEUs those logic blocks are the main concerns for

those modules. Detailed analysis for this statement is presented in this section.



(a) (b)

Figure 33: Layout screenshots of the ADPLL design on (a) 32nm SOI technology and (b) 65nm bulk technology.

A typical bang-bang ADPLL design was synthesized and laid-out across three technologies - 32nm SOI, 65nm bulk, and 180nm SOI technologies. The design used a 7-bit digitally-controlled ring oscillator. The DLF module of the ADPLL consisted of a proportional path with a gain of 2^{-1} and an integral path with a gain of 2^{-4}. The operating frequency range for the reference clock frequency of the ADPLL design was from 40 MHz to 100 MHz. Layout screenshots of designs on 32nm SOI and 65nm bulk technologies are presented in Fig. 33. The simulated design specifications for the four ADPLLs are shown in Table 2.

The maximum SET pulse widths for particle LET of less than 60 $MeV - cm^2/mg$ for each technology are estimated from literature. For instance, Chen *etal.* showed TCAD simulation results on digital SET pulse widths for 180nm PD SOI. At room temperature for LET of 60 $MeV - cm^2/mg$, SET pulse widths for floating-body and

body-tied inverters are 0.42 ns and 0.05 ns respectively[130]. Therefore, 0.42 ns is the reported maximum SET pulse width for 180 nm SOI in that paper. The summarized maximum SET pulse widths for all three technologies are listed in the table below in Table 2, listed in which are comparisons of area and gate counts for logic gate and FFs in the logic blocks (PD and DLF combined), and also the setup and hold time for FFs of ADPLL designs.

Table 2: Comparisons of ADPLL designs and maximum SET pulse widths (PWs) across technologies.

	32nm SOI	65nm Bulk	180nm SOI
MAX SET PWs (ps)	107 [131]	250 [132]	420 [130]
Logic Gate Counts	110	111	119
FF Counts	22	22	22
Logic Gate Area (μm^2)	57.6	3219	12554
FF Area (μm^2)	135	341	5280
FF Setup-Hold Time (ps)	43	72	120

As stated in Chapter II, a window of vulnerability exists for the originating SET to be latched in a FF. In fact, the SET-induced error probability can be described as follows[133]:

$$P_{ERROR} = \frac{t_{pw} + t_{SH}}{T + t_{pw}}, \tag{27}$$

where t_{pw} is the generated pulse width, t_{SH} is the set-up and hold time, T is the clock period (T=1/f) and f is the operating frequency. For low operational frequency, static upsets are dominant. As frequency increases, temporal masking factor related to the pulse generated and the set-up and hold time of specific circuits becomes larger. This leads to increased vulnerability to SET-induced soft errors and logic upsets may

dominate overall soft errors.

Assume average logic masking probability is P_{mask}, i.e. every originating SET in the logic path will be masked when propagating to the registers with a probability of P_{mask}. Using the maximum SET pulse widths for logic gates and the FF setup-hold time for the specific technology, P_{ERROR} can be calculated for a specific clock frequency. The SET-induced error probability is dependent on P_{mask}, P_{ERROR}, the logic gate area, the SE susceptibility of transistors and radiation environment, while SEU-induced error probability is only dependent on the FF gate area, the SE susceptibility of transistors and radiation environment. Therefore, the ratio between them is technology invariant and is only dependent on the design topologies, which is shown in Eqn. 28

$$\frac{P_{SET}}{P_{SEU}} = \frac{A_{logic} \cdot P_{ERROR} \cdot (1 - P_{mask})}{A_{FF}}, \tag{28}$$

in which A_{logic}, A_{FF} are the area of the logic gates and FFs, respectively.

SET latching probability (P_{ERROR}) (indicated with black triangles) across technology nodes is shown in the black curve in Fig. 34 at clock frequency of 100 MHz. As expected, SET latching probability decreases from 180nm SOI to 32nm SOI technology due to the decrease of maximum SET pulse width. The curve with blue squares in Fig. 34 is the ratio of SET- and SEU-induced error probabilities for the ADPLL designs across technology nodes at clock frequency of 100 MHz with logic masking probability P_{mask}=0, which indicates that every originating SET in the logic path will be able to registers without masking. However, reported logic masking probability for a digital circuit generally stays in the range of 10% $\sim$ 50%[134]. Therefore, SEUs are more than 10x more likely to cause an output errors for PD

and DLF comparing to SETs in theses modules and the trend exacerbates with the scaling of technology.

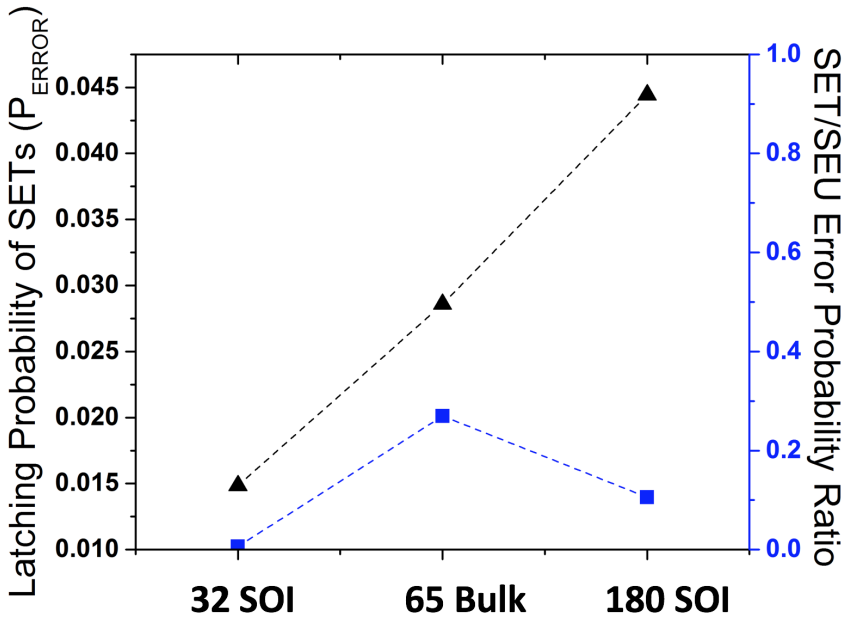


Figure 34: The ratio of SET- and SEU-induced error probabilities for the ADPLL designs (blue) and the SET latching probability (P_{ERROR}) (black) across technology nodes with logic masking probability P_{mask}=0 at reference clock frequency of 100 MHz.

ADPLL SEU Sensitivity Analysis and Verification

For PD, DLF and FD design modules, SETs in the digital logic have to be latched in FFs to manifest as a design output error. Since stable, low frequency crystal oscillators are often chosen as the reference clock (f_{REF}) to minimize injected reference period jitter, f_{REF} is often less than 500 MHz[5][113][114]. SEU-induced errors are the main concerns for PD and DLF, since they operate by the reference clock.

In this section, as shown in Fig. 35, the response signatures of common ADPLL topologies to SEUs in the ADPLL sub-circuits are compared through FPGA-based fault injection experiments [135]. The impact of different sub-circuit topological

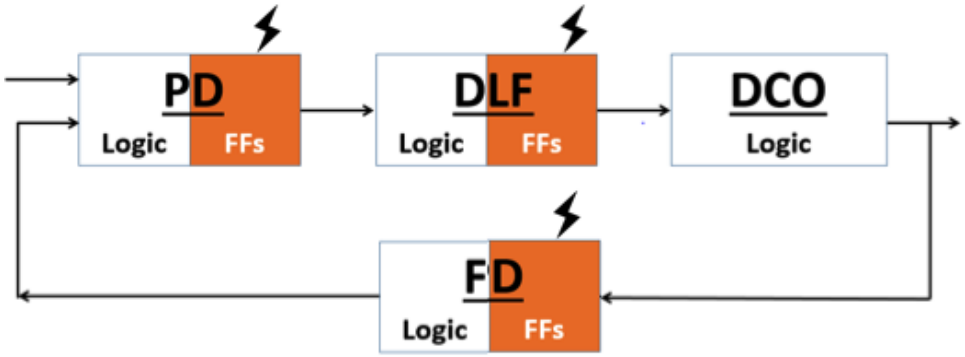


Figure 35: Block diagram illustration of conducted FPGA-based fault injection experiment.

design choices (not physical implementation) on the overall ADPLL SEU performance is analyzed. The maximum perturbation time and frequency error of the output signal is used to quantify the SEU responses of each of the sub-circuits. The most sensitive registers to SEUs are identified and mitigation strategies proposed.

FPGA-based Fault Injection Experimental Setup

Through instantiation of the ADPLL designs using the IBMCS7RF standard cell library, the number and topological locations of the storage elements (i.e. flip-flops) utilized in each design module are identified. Similar to the SET/SEU simulation setup stated in the previous chapter, ADPLL design topologies with different PD and DLF implementations are synthesized onto the FPGA in order to perform at-speed hardware fault experiments and verify the SEU simulation characterization results.

The ADPLL designs utilize counter-based DCOs[136], of which the period of the output signal is linearly proportional to the digital control word. These DCOs are representative of a typical ring or LC-tank based DCO [5] for translating the digital word change to output frequency change(as stated in Chapter II) and understanding

the overall ADPLL SEU responses. Measured specifications of the synthesized ADPLL designs are listed in tables presented in following subsections. SEU-induced error signatures were studied by performing an exhaustive FPGA-based fault injection campaign[135] based on the SEU Simulation Tool (SST)[137]. Over 50 faults were injected randomly over time during locking state for each flip-flop used in the ADPLL sub-circuits. During these tests, the output of the ADPLL was monitored to determine the worst-case SEU responses.

The FPGA-enabled hardware study described here represents an analysis of the implication of topological design choices on SEE response, not the physical details of mechanisms. Clearly, FPGA hardware implementations of the ADPLLs of study here do not represent the details of transistor-level ASIC design instantiations, nor is that the goal of this study. The intent of this study is to elucidate the impact of logic topology choice on error response.

SEU Sensitivity Analysis

In this section, design choices for each module in a typical ADPLL are evaluated based on their impact on overall ADPLL SEU performance. One module is chosen as the variable every time for comparing design choices to avoid overlapping of parameters. The maximum SEU-induced perturbation duration of the output in terms of the number of reference clock cycles is monitored in order to compare the ADPLL SEU responses across several design topologies.

Experimental fault injection results are elaborated in the following subsections for the SEUs injected at the registers in the PD, DLF and FD. For SEUs within the PD, registers are used for up and down bits in a bang-bang topology and for counter values

in the frequency-based PD topology. For SEUs within the DLF, feedback registers are used corresponding to the number of poles in the DLF. The SEU response was plotted as a function of the bit weight (from the least significant bit or LSB to the most-significant bit or MSB) of the register. SEU in the single-bit bang-bang PD is plotted in the same manner so as to compare directly to the response of the DLF. Limit cycle errors are denoted by arrows pointing to infinity where the frequency perturbation is uncorrectable and requires restarting of the system.

The design choices this work has evaluated are listed below:

- **PD:** bang-bang PD, integer-based frequency PD and fraction-based frequency PD
- **DLF:** 1st-order PI filter, 2nd-order PI filter, and high-order DLF with PI filter cascaded with FIR/IIR filters
- **DCO:** number of bits in the digital control word (i.e. gain of the DCO k_{DCO})
- **FD:** counter-based FD with integer divisor of 4/8/16

PDs

As stated in Chapter II, three major types of PDs are commonly utilized in ADPLLs - frequency-based PDs, time-based PDs and FSM-based PDs. TDC is one of the most common PD topologies used for time-based phase detection or both frequency or phase error detection. However, TDCs implemented on FPGAs are faced with differential nonlinearity (DNL) from limited circuit granularity and resolution. [138][139]. [140] proposed a TDC with sub-cell-delay resolution. However, the proposed TDC topology is drastically different from the TDC version in application-specific integrated circuits (ASICs), making it difficult for SEU fault injection and not

comparative to ASIC design instantiations. Therefore, three types of PDs are chosen to be discussed in this section - bang-bang PD, integer-based frequency PD and fraction-based frequency PD. FSM-based PDs, which can switch between frequency-detection mode and time-detection mode, are discussed in Chapter V.

To discuss the impact of the different topological choices of PDs on the SEU performance of the overall ADPLL, six ADPLL designs are analyzed consisting of 1st-order and 2nd-order ADPLLs with the three different types of PDs. Since the frequency-based PDs operates linearly in the frequency domain, ADPLLs with integer-based and fraction-based linear PDs are referred to as "linear ADPLLs" in this section. An 8-bit counter-based DCO and a frequency divider with frequency division of 8 are shared across all six designs. The experimental measurements of the designs are provided in Table 3.

Table 3: Design measurements for 1st-order and 2nd-order 7-bit linear and bang-bang ADPLLs at 713 Hz.

ADPLL (# of pole)		1	2	1	2	1	2
PD		integer-based		BB		fraction-based	
Period jitter		<0.1%	<0.1%	3.4%	3.4%	<0.1%	<0.1%
Frequency range		285Hz∼10.5MHz					
Frequency tuning resolution		40 ns/LSB					
Loop filter gain	α	1/2	1/2	1/4	1/2	1/2	1/2
	ρ	N/A	1/16	N/A	1/16	N/A	1/16
Gain implementation	k	7					
	n	1	4	2	4	1	4
Lock-in speed (# of ref. clk cycles)		40	45	60	111	35	40

1) SEUs in the 1st and 2nd order bang-bang ADPLLs

Fig. 36 shows the SEU responses of the 1st and 2nd order bang-bang ADPLL for

faults injected in each PD and DLF register bit. As expected, SEUs in the bang-bang PD only result in temporary-frequency errors at the output for less than 5 reference clock cycles. SEUs in the registers corresponding to the first and second poles of the DLF exhibit a bit weight dependence on the length of the perturbation from LSB to MSB. In addition, SEU-induced limit cycle errors were witnessed for the highest 2 bits of register corresponding to the second pole in the 2nd-order ADPLL. The ADPLL recovered for SEUs occurring in bits 7 and 8, as shown in Fig. 36b.

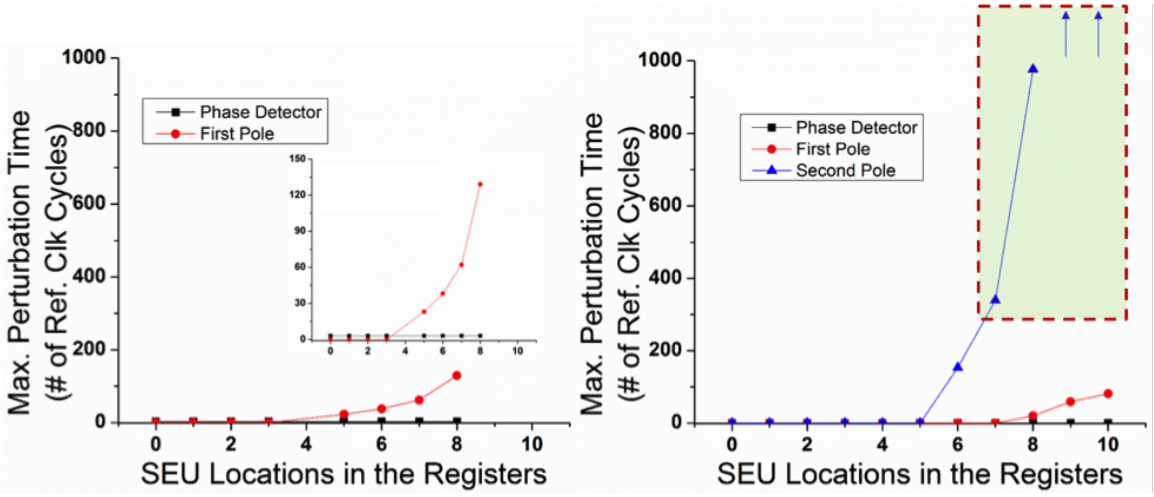
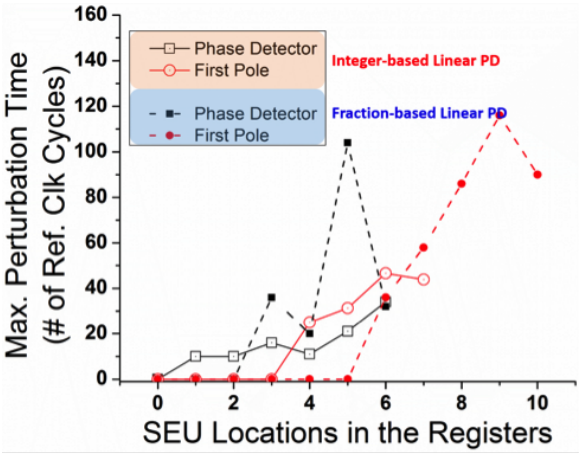


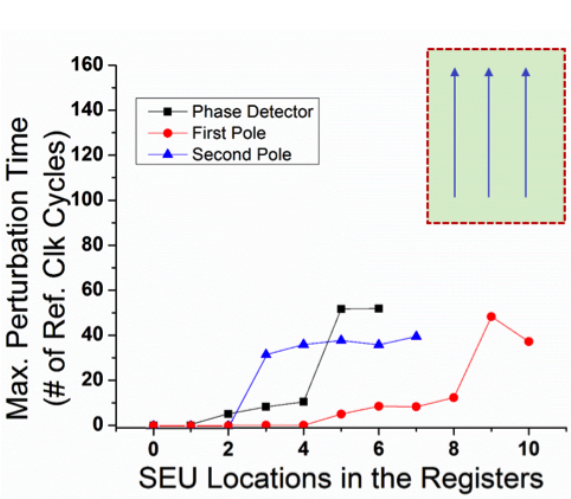
Figure 36: FPGA fault injection results on SEU sensitivity of different registers in (a) 1st-order bang-bang ADPLL (the inset figure is the zoomed-in version)and (b) 2nd-order bang-bang ADPLL at the same output frequency of 713 Hz.

2) SEUs in the 1stand 2nd order linear ADPLLs

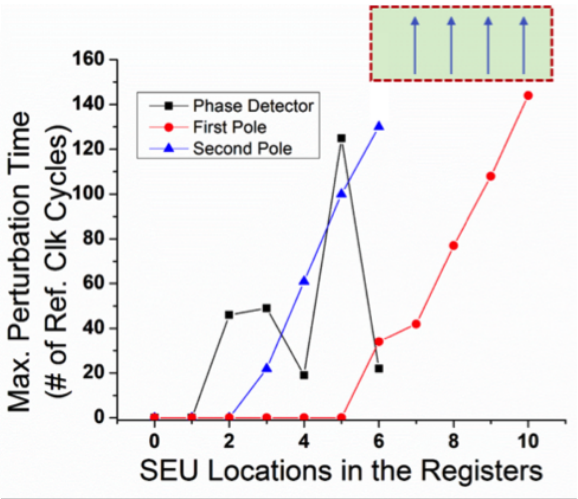
Fig. 37 shows the comparison between SEU responses of the 1st and 2nd order linear ADPLL with fraction-based linear PD and integer-based linear PD for faults injected in each PD and DLF register bit. In contrast to bang-bang ADPLLs. SEUs in the linear PD are able to induce loss-of-lock errors in both 1st and 2nd order linear ADPLLs. This is due to the large control word or frequency deviation as a result of



(a)



(b)



(c)

Figure 37: FPGA fault injection results on SEU sensitivity of different registers in (a) 1st-order ADPLL with fraction-based linear PD and integer-based linear PD, (b) 2nd-order ADPLL with integer-based linear PD and (c) 2nd-order linear ADPLL with fraction-based linear PD at the same output frequency of 713 Hz.

integrating the PD output through the DLF proportional path (shown in Fig. 13). SEU-induced limit cycle errors were also observed for the highest bits of the register corresponding to the second pole in the 2nd-order ADPLL, as shown in Fig. 37b.

In addition, Fig. 37a shows that a 1st-order ADPLL with integer-based linear PD exhibit better single-event performance than a comparable ADPLL with a fraction-based linear PD. This is because the former design exhibits shorter maximum perturbation time at bits with large bit weights. The former design also has fewer the overall sensitive bits both in PD and DLF compared with the latter design.

A similar comparison is observed by comparing Fig. 37b with Fig. 37c for 2nd-order linear ADPLLs. However, the length of the perturbation exhibits less bit weight dependence from LSB to MSB for fraction-based linear ADPLLs comparing with bang-bang ADPLLs. This is because the divider in the fraction-based linear PD introduces another level of non-linearity to the system, which requires longer time to settle.

DLF

As stated in previous chapters, two major design parameters for the DLF in an ADPLL are the order of the DLF and the gain to the integral path and proportional path. A higher order DLF uses more integrators as the poles for higher filtering capability. Similarly, larger incoming changes are required with a lower gain in the paths to increase or decrease the current control words, which means higher filtering capability as well.

A. Gain implementation

It is convenient to implement the proportional path gain (α) and integral path

gain (ρ) in power-of-2 values, i.e. 2^{-n}, since the multiplying operation then simplifies to a trivial right-shift by n bits. However, the right-most n bits are preserved along with the k-bit control word during calculation for precision. For instance, 11-bit register is used for a 7-bit ADPLL for a gain of 2^{-4}. The integral gain ρ is required to be much smaller than α to avoid system overdamping [141].

To illustrate the impact of gain implementations on the ADPLL SEU performance, a 1st-order ADPLL design using a 1-bit (bang-bang) PD, a 8-bit counter-based DCO, a 1st-order DLF and a FD was synthesized on an Altera FPGA board for fault injection experiments. The proportional path gain (α) of the DLF is varied from 1/2, 1/4, 1/8 to 1/16. As illustrated in Fig. 38, ADPLL SEU performance worsens with decreasing of gain. For high-order DLFs, the integral gains for other paths are set by the gain for the proportional path for system stability. Therefore, the results can be extended for high-order DLFs.

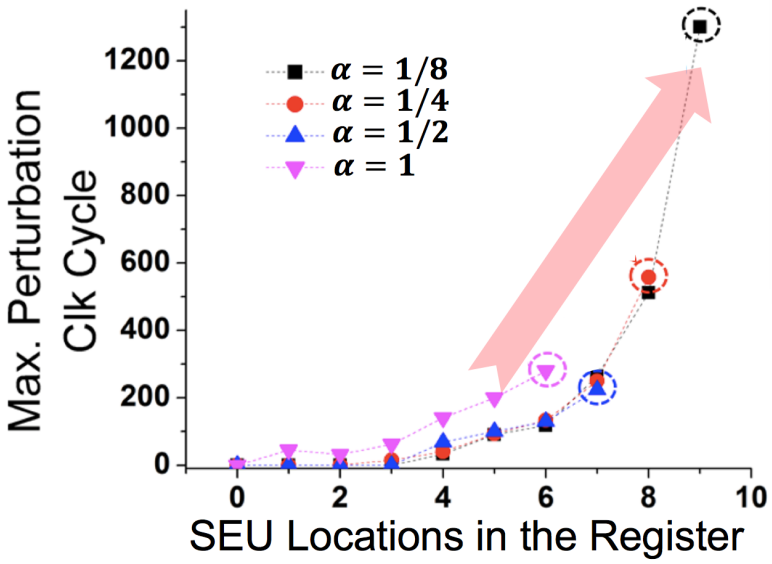


Figure 38: FPGA fault injection results on DLF in a 1st-order ADPLL with different proportional gain (α).

B. Order of the DLF

Two designs sharing the same a 1-bit (bang-bang) PD, a 8-bit counter-based DCO, and FD, but differentiated by the DLFs, were synthesized on Altera FPGA board for fault injection experiment. DLFs in both designs use PI filters only (i.e. without cascading FIR/IIR filters). One of the two designs is a 1st-order bang-bang ADPLL, which incorporates a PI filer with proportional path with a proportional gain of 2^{-1} only. And the other design, a 2nd-order bang-bang ADPLL, utilizes a PI filer with proportional path with a gain of 2^{-1} and a integral path with a gain of 2^{-3}.

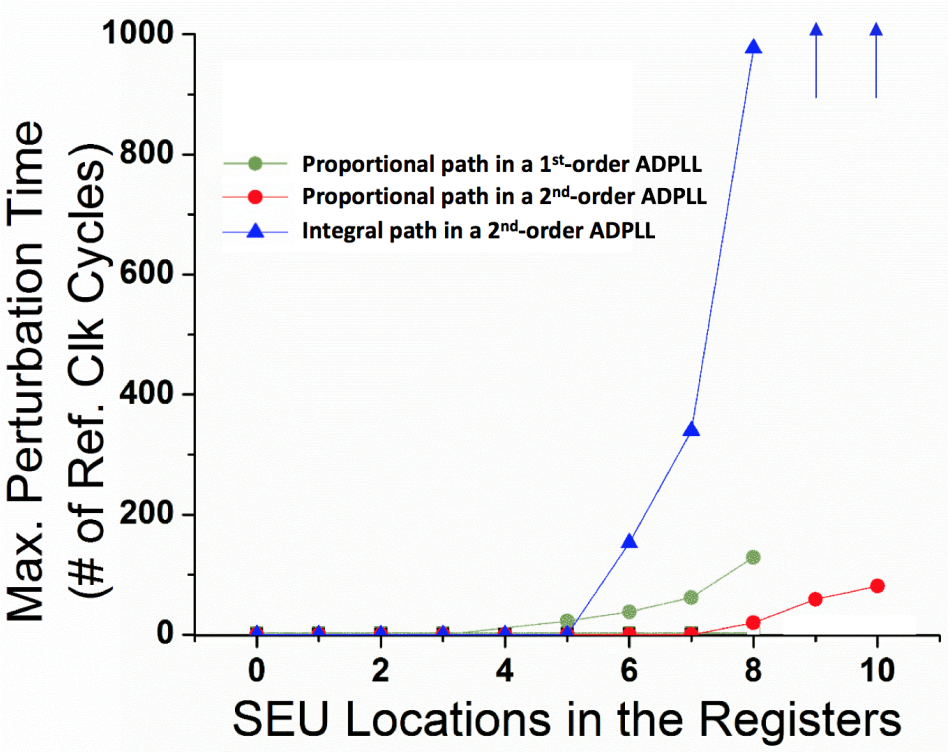


Figure 39: FPGA fault injection results on SEU sensitivity of PI filters of the DLFs in 1st-order and 2nd-order ADPLLs.

For SEUs occurring in the PI filter, Fig. 39 plots the maximum perturbation time duration in terms of reference clock cycles for SEUs in proportional and integral paths. The figure shows the experimental fault injection results for the SEU response as a

function of the bit weight (from the least significant bit or LSB to the most-significant bit or MSB) of the register. For both a 1st-order ADPLL and 2nd-order ADPLL, SEUs in the MSBs of the register in the proportional path in the DLF can induce loss-of-lock errors and those in the LSBs can only result in temporary-frequency errors. However, for the 2nd-order ADPLL, SEUs in the MSBs of the register in the integral path in the DLF can induce limit-cycle errors and loss-of-lock errors, while those in the LSBs can only result in temporary-frequency errors. Even though SEUs in the proportional path of DLF for 1st-order ADPLL results in longer perturbation time compared with those for 1st-order ADPLL and less output perturbation time compared with those for 2nd-order ADPLL.

As stated in Chapter II, high-order DLF can be constructed by cascading the PI filter with FIR/IIR filters. The logic-level design topologies of four ADPLLs were implemented on Altera DE2-115 FPGA. All of them shared the same fraction-based linear PD, a 10-bit counter-based DCO, and a divide-by-8 FD. They are differentiated by the design implementations of the DLFs, as shown in Table 4.

Table 4: DLF designs in four synthesized ADPLL topologies.

	FIR	IIR	PI
Design 1	y[k]=0.5(x[k]+x[k-1])		$\alpha = 2^{-1}, \beta = 2^{-4}$
Design 2	y[k]=0.25(x[k]+2x[k-1]+x[k-2])		$\alpha = 2^{-1}, \beta = 2^{-4}$
Design 3		y[k]=0.5(x[k]+y[k-1])	$\alpha = 2^{-1}, \beta = 2^{-4}$

The DLFs in design 1 and design 2 were constructed with PI filter and FIR filters - one utilized a 2-tap FIR filter while the other utilized a 3-tap FIR filter. For SEUs occurring in the FIR filter, Fig. 41 shows the worst-case SEU response of the two ADPLL designs. Erroneous ADPLL output clock signals are observed to span only

a few clock cycles, as shown within the shaded boxes of Fig. 40. Specifically, the maximum number of clock cycles affected is equal to the number of taps used in the FIR filter, thus SEUs in FIR filter are always temporary frequency errors.

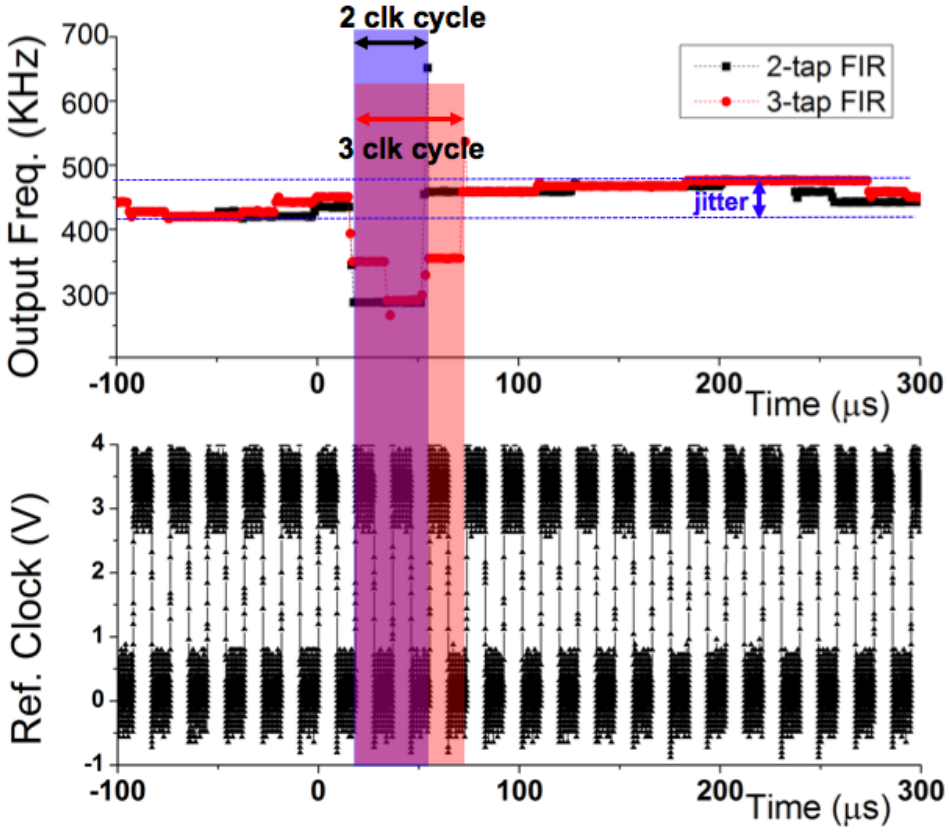


Figure 40: Worst-case ADPLL SEU response in terms of output frequency. The output signal(above) and reference clock signal (below) are plotted over the same period of time when SEUs occur in registers in FIR filters for two ADPLLs using a 2-tap and a 3-tap FIR filter respectively. Erroneous ADPLL output clock frequency errors are is observed only over a few reference clock cycles as shown in the colored boxes.

DLF in design 3 was constructed with PI filter and IIR filters. Therefore, the single pole in the IIR filter becomes the 3rd pole in the ADPLL system in addition to the 2 poles in the PI filter. As shown in Fig. 41, maximum perturbation time is plotted for SEUs occurring in every bit location in the register corresponding to the three poles in the ADPLL. It is shown in the figure that SEUs in the most significant

bits in the pole registers can lead to limit cycle errors. SEUs in higher order pole result in longer perturbation time at ADPLL output. Comparing Fig. 41 with Fig. 39, 3rd-order ADPLLs are more vulnerable towards SEUs compared with 2nd-order or 1st-order design topologies.

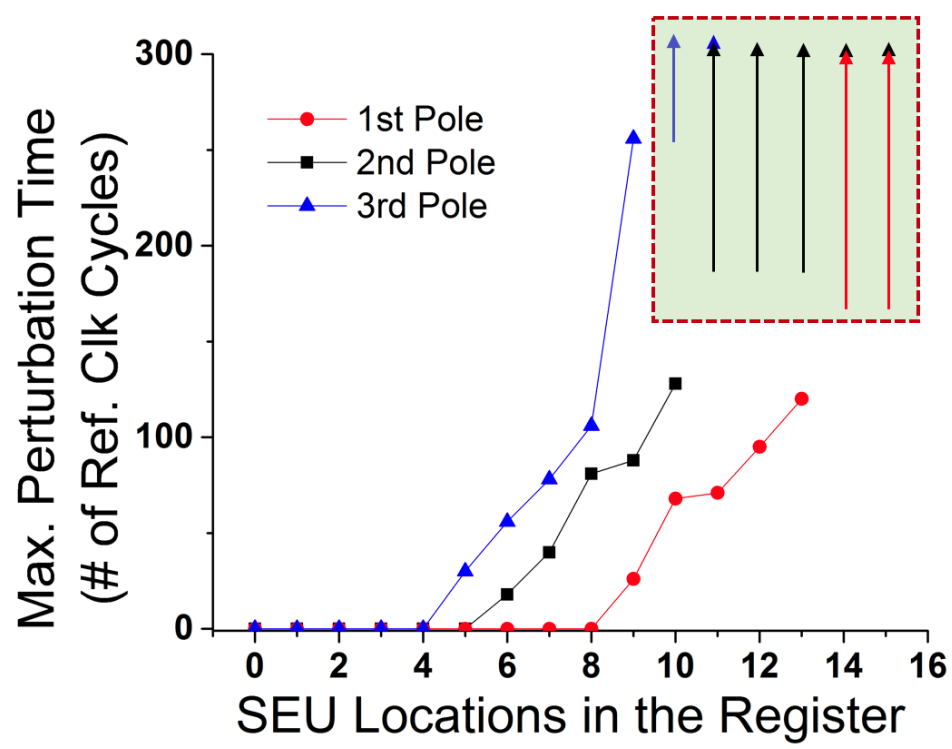


Figure 41: Maximum output perturbation time is plotted for SEUs occurring in every bit location in the register corresponding to the three poles in the ADPLL.

SEUs in IIR filters results in more severe error signatures comparing to SEUs in FIR filters. However, it sometimes takes more than a 10-order FIR filter to achieve similar filtering capability as a single-pole IIR filter.

DCOs

In this subsection, four ADPLLs consisting of 1st-order and 2nd-order ADPLLs with 7-bit or 10-bit DCOs are analyzed to discuss the impact of the different topological choices of DCOs on the SEU performance of the overall ADPLL. The four designs share the same frequency tuning range,fraction-based linear PDs and frequency divider design with frequency division of 8. The experimental measurements of the designs are provided in Table 5. The gain implementation of the designs are listed in the table to illustrate the number of bits in the integral register in the DLF.

Table 5: Design measurements for 1st and 2nd-order fraction-based linear ADPLLs with different DCOs at 128 Hz.

DCO		7-bit		10-bit	
ADPLL (# of poles)		1	2	1	2
Period jitter		<0.1%	<0.1%	<0.1%	<0.1%
Frequency range		35.6Hz 10.5MHz			
Tuning resolution		686ns/LSB		40ns/LSB	
Loop filter gain	α	1/16	1/2	1/16	1/2
	ρ	x	1/16	x	1/16
Gain implementation	k	7		10	
	n	4		4	
Lock-in speed (# of ref. clk cycles)		23	40	35	85

The ADPLL output period is plotted over digital control word for all the analyzed designs in Fig. 42. The ADPLL design with 10-bit DCO is noted by blue line and the ADPLL design with 7-bit DCO red line since they have different tuning ranges. Locking frequencies of the analyzed ADPLLs are marked with circles in the plot. The difference of the locking frequencies does not significantly affect the analysis since the locking frequency has minimal impact on the SEU performance PDs and DLFs, as

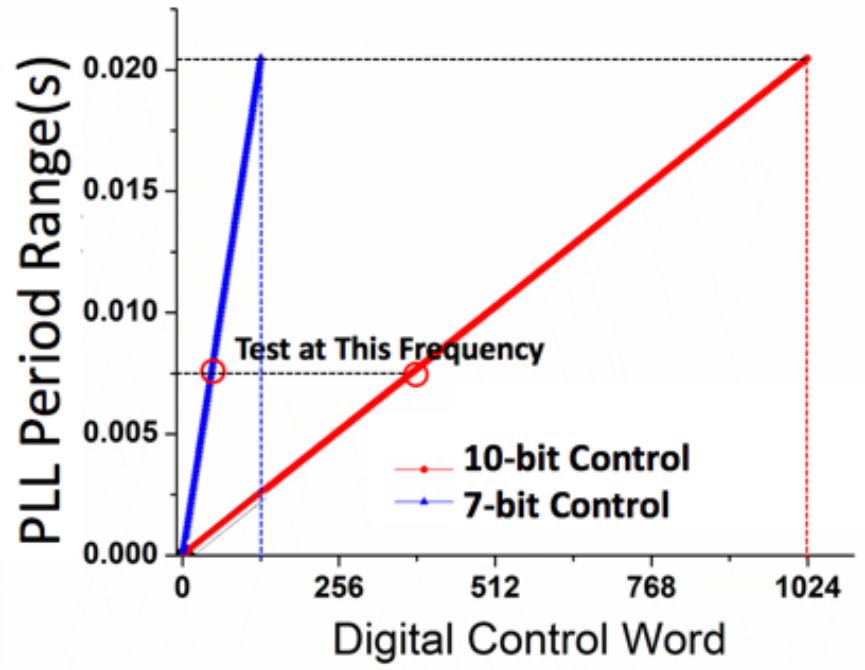
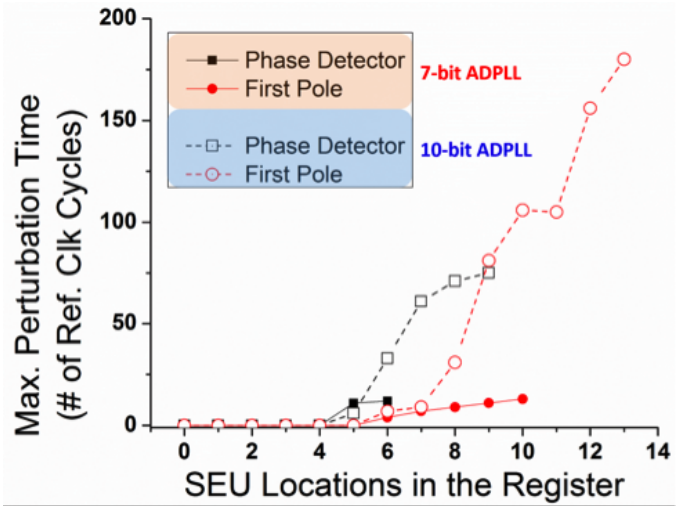


Figure 42: ADPLL output period over digital control word for the analyzed design topologies. Locking frequencies of the analyzed ADPLLs are marked with circles in the plot.

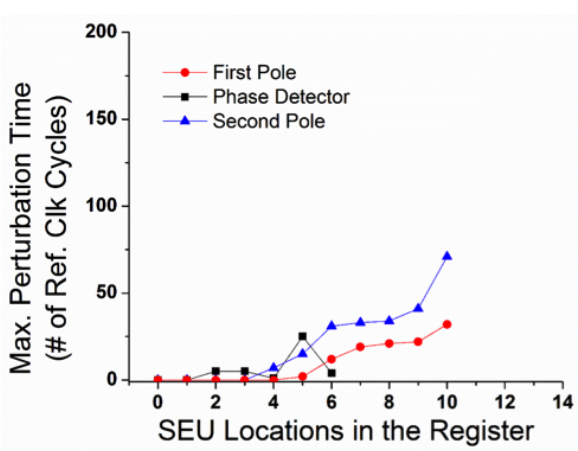
they are synchronous to the reference clock. Despite the control word differences, SEUs at a specific bit locations (i.e. bit i) in the register still result in the same control word deviation (i.e. 2^i).

As stated, four linear ADPLLs with different DCOs (design measurements listed in Table 5) are analyzed in this subsection. Fig. 43 demonstrate the comparisons between SEU responses of the 1st and 2nd order linear ADPLL with 7-bit and 10-bit DCOs for faults injected in each PD and DLF register bit.

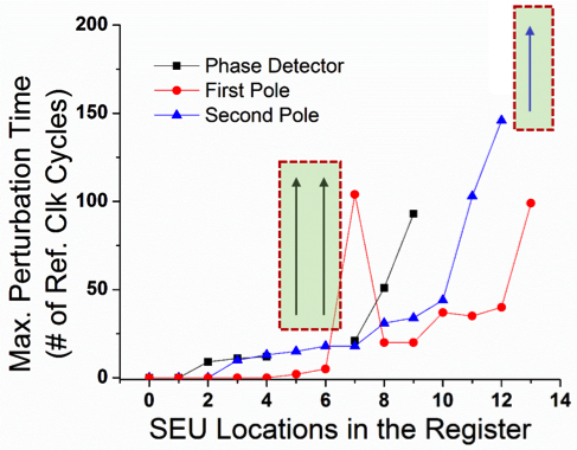
SEUs in the linear PD are able to induce loss-of-lock errors in both 1st and 2nd order linear ADPLLs. SEU-induced limit cycle errors were observed for the highest bits of the register corresponding to the second pole in the 2nd-order ADPLL, as shown in Fig. 43b. In addition, Fig. 43a shows that 1st-order ADPLL with 7-bit DCO



(a)



(b)



(c)

Figure 43: FPGA fault injection results on SEU sensitivity of different registers in (a) 1st-order ADPLL with 7-bit and 10-bit DCO, (b) 2nd-order linear ADPLL with 7-bit DCO and (c) 2nd-order linear ADPLL with 10-bit DCO at the same output frequency of 128 Hz.

exhibits better single-event performance than the one with 10-bit DCO. As stated previously, this is because the former design exhibits shorter maximum perturbation time at bits with large bit weights and has fewer the overall sensitive bits both in PD and DLF comparing with the latter design.

Similar comparison is observed by comparing Fig. 43b with Fig. 43c for 2nd-order linear ADPLLs. This can be explained from Eqn. 4. When the ADPLL is in-lock, both N and M equal to 8 and the result of Eqn. 4 is 0, i.e. the PD outputs no adjustment on the current control word. When an SEU occurs in the PD, M in Eqn. 4 is perturbed, while D_{center} and N are fixed. The ADPLL perturbation time is directly proportional to the initial PD output perturbation, i.e. δD_{center}/M. Since both designs have the same frequency tuning range, the ratio between their denominators D_{center} is the same as the ratio (λ) between the gain of the DCOs (K_{DCO}). In the case discussed here, the ratio (λ) between the 7-bit and 10-bit ADPLL is 8. The perturbation in M (δM) is $2^i K_{DCO}$, if the perturbed bit i is between bit 0 and 6, as shown in Fig. 43 (b-c),

$$\delta\frac{D_{center1}}{M1} = \frac{D_{center1}}{8+\delta M1} = \frac{D_{center1}}{8+2^i K_{DCO1}}, \tag{29}$$

$$\delta\frac{D_{center2}}{M2} = \frac{\lambda D_{center1}}{8+2^i \lambda K_{DCO1}} = \frac{D_{center1}}{1+2^i K_{DCO1}}. \tag{30}$$

Since M is significantly larger than 8 (or 1), the PD output perturbation for both designs are close to each other as shown in Eqn. 29 and Eqn. 30, i.e. maximum ADPLL output perturbation duration for both designs are similar.

In addition, the large bit weight of the perturbed bit, the larger resulted perturbation in the PD output and the larger perturbation time for the ADPLL. Hence, for the 10-bit ADPLL, from bit 0-6, the SEU-induced maximum perturbation

time is similar to that of the 7-bit ADPLL, while for bit 7-10, the perturbation time continues to increase.

SEU-induced limit cycle errors were observed for the register in the fraction-based linear PD in the 2nd-order 10-bit ADPLL (bit 6 and 7) but not for the 2nd-order 7-bit ADPLL, as shown in Fig. 43(b-c). Limit cycle errors occur because the SEU perturbation in the fraction-based linear PD pushed the loop to converge in an oscillating state. Large control word span, i.e. a large number of bits in the digital control word, corresponds to large combinations for possible frequency states for limit cycle errors. Theoretically, it is easier for SEU perturbation to cause limit-cycle errors at bit locations with large bit weight than at those with small bit weight. Due to the limiting number of testing cases in the experiment, limit-cycle errors resulted from SEUs at bits with higher bit-weight than 2^7 were not observed.

FD

It is common practice to include a programmable FD in the feedback path of the PLL to provide different selections of frequencies at the output of the PLL. In fact, Loveless *etal*. and Hafer *etal*. recently showed data indicating in A/MS PLLs both the location and gain of the FD in the PLL configuration strongly influence the predicted error rate[142][143].

This section analyzes the SEU signatures of an integer-N frequency divider and the impact of frequency divisor on the overall ADPLL SEU performance of the loop. Several programmable PLL topologies are designed and synthesized on Altera DE2-115 FPGA board for fault injection experiment to corroborate the discussed analyses.

Since the frequency divisor is usually contained inside the frequency-based PD,

which allows the omitting of an actual FD in the feedback path in the ADPLL, thus resulting in an architectural difference when compared with ADPLLs with time-based PDs. The impact of FD configuration on the ADPLL SEU performance is discussed for these two architectures separately.

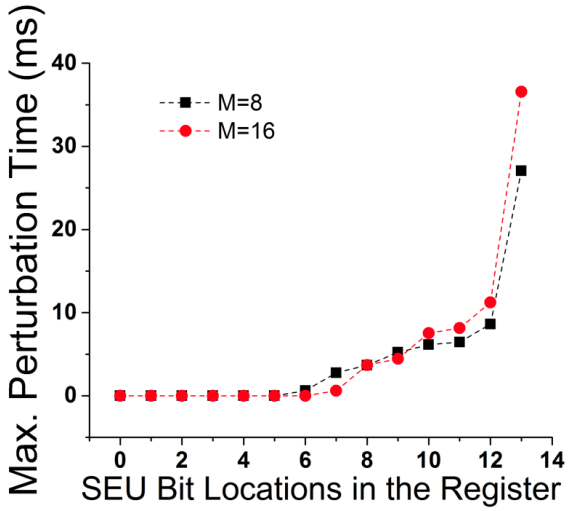
I. ADPLLs with frequency-based PDs

An ADPLL design incorporates a fraction-based PD, a 10-bit DCO and a DLF with proportional gain (α) of 2^{-1} and integral gain (ρ) of 2^{-4} is designed and synthesized on Altera DE2-115 FPGA. The programmable frequency divisor (M) of 8 and 16 is achieved inside of the fraction-based linear PD by changing the expected number of oscillator clock cycles in Eqn. 3. SEU fault injection experiment is performed when the ADPLL locking frequency equals to 104 kHz for both frequency division modes. In other words, the reference clock frequency for frequency divisor (M) of 16 is twice that of the frequency divisor of 8.

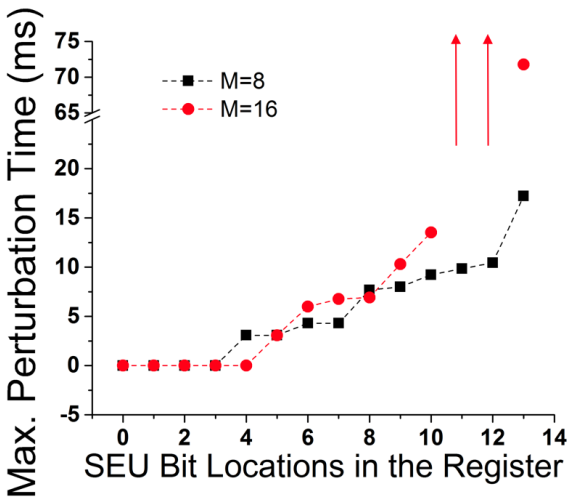
Plotted in Fig. 44 is the maximum output perturbation time over the SEU bit locations in DLF and PD. As indicated in Fig. 44a, changing the frequency divisor M does not have significant impact on the loop SEU performance when SEU occurs in the 1st-order pole of the DLF. However, with the increasing of frequency multiplication factor M, even though the perturbation time corresponding to each bit location of the SEU occurrences does not change drastically, it is more likely to observe limit-cycle errors the loop, i.e. becomes loop becomes more easily to be perturbed, as shown in Fig. 44b and Fig. 44c.

II. ADPLLs with time-based PDs

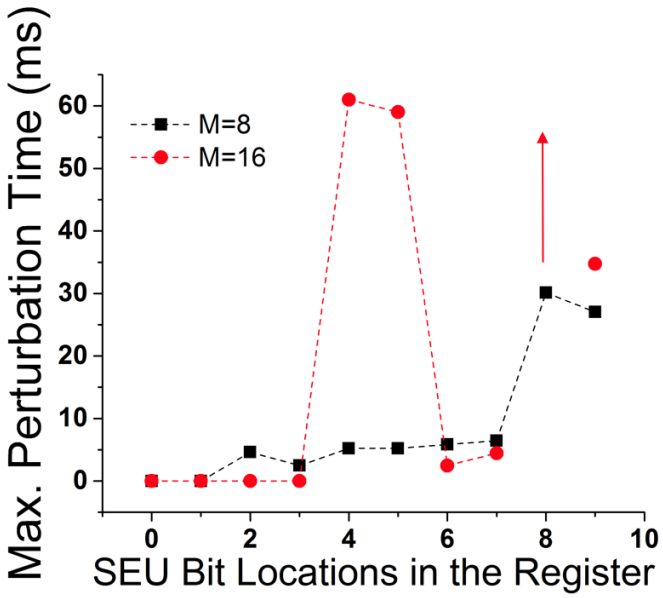
An ADPLL design incorporating a bang-bang PD, a 7-bit DCO and a DLF



(a) 1st-order Pole in DLF



(b) 2nd-order Pole in DLF



(c) PD

Figure 44: FPGA fault injection results in terms of maximum ADPLL output perturbation time for SEUs in different registers in (a) 1st-order pole in the proportional path of DLF, (b) 2nd-order pole in the integral path of DLF and (c) PD when the ADPLL is programmed to achieve frequency multiplication factor (M) of 8 or 16 at the same output frequency of 104 kHz.

with proportional gain (α) of 2^{-1} and integral gain (ρ) of 2^{-4} was designed and synthesized on Altera DE2-115 FPGA. The ADPLL contains a programmable FD with multiplexed frequency divisor (M) of 4 and 8. SEU fault injection experiment is performed when the ADPLL locking frequency equals to 128 kHz for both frequency division modes. Similar to the fault injection experiment setup for ADPLLs with frequency-based PDs, the reference clock frequency for frequency divisor of 8 is twice that of the frequency divisor of 4 to keep the output clock signal frequency the same.

Plotted in Fig. 45 is the maximum output perturbation time over the SEU bit locations in the 1st and 2nd-order pole in the DLF. Unlike ADPLLs with frequency-based PDs, SEU-induced output perturbation time exhibit a significant increase with the increase of frequency divisor for each bit location in the register. This is because the loop recovery time is dependent on the gain of the PD and the frequency divider for time-based PDs, while in ADPLLs with frequency-based PDs the frequency multiplication factor M only serves as a reference, which only imposes second order impact on the loop settling time.

In addition, only temporary-frequency errors were observed for SEUs in the FFs in the FD. This is because that the SEUs in the FD can only result in the erroneous operation of bang-bang PD for 1 reference clock cycle, which is within the jitter requirement in its normal operation.

Design Considerations

SEU performances of ADPLLs with different configurations of PDs, DLFs and DCOs are analyzed in this chapter. Different design choices can be made for difference space applications to trade off electrical performance versus SER performance for

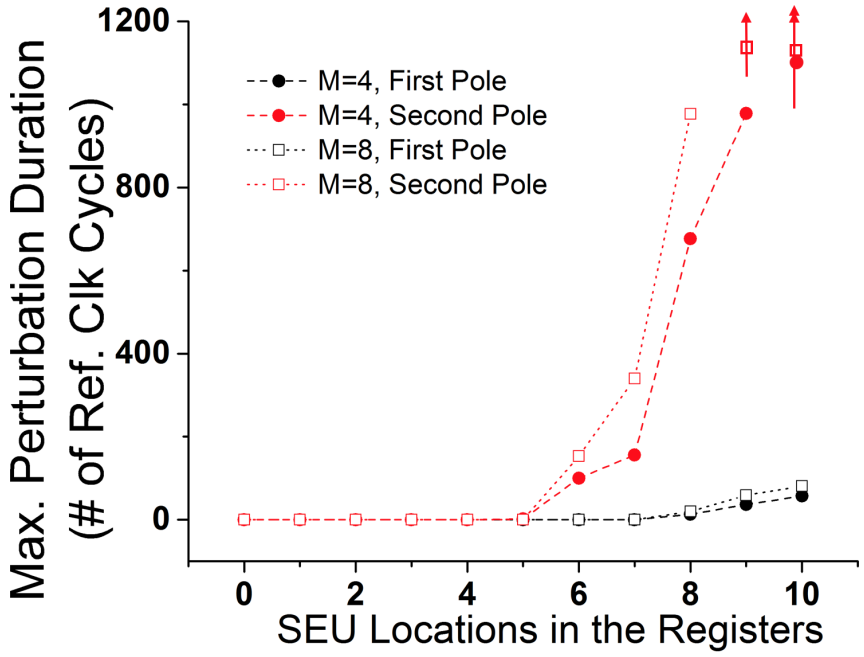


Figure 45: Maximum output perturbation time over the SEU bit locations in the 1st and 2nd-order pole in the DLF at output clock frequency of 128 kHz.

ADPLLs.

Bang-bang PDs are easy to implement in ADPLL design and SEUs in bang-bang PDs only cause temporary-frequency errors. However, due to its inherent resolution limitation, bang-bang ADPLLs often have very narrow frequency tracking ranges. Comparing to bang-bang ADPLLs, ADPLL topologies with frequency-based PDs (i.e. linear PDs) yield a lot better jitter performance over larger frequency tracking ranges. Fraction-based PDs perform better than integer-based PDs due to its linearity in frequency tracking. However, linear PDs generally are more complex to design. SEUs in linear PDs can cause loss-of-lock errors. SEUs in a fraction-based linear PD can even lead to limit-cycle errors because of the non-linearity resulted from using divider.

ADPLLs using lower-order DLFs has faster acquisition time than ones using

higher-order DLFs. 1st-order and 2nd-order DLFs are achieved by the proportional path and integral path in DLFs and higher-order DLFs can be formed by cascading IIR/FIR filters with PI filters. Despite of different types PDs and DCOs used in the ADPLL, the ADPLL output perturbation time usually exhibits a dependence on the bit weight of the perturbed bit. SEUs in the high-order poles (i.e. pole order larger than 1) in the DLF can result in limit-cycle errors, while other bits with large bit weight can result in loss-of-lock errors. The rest of the bits with small bit weights can only result in temporary-frequency errors.

In general, high-order ADPLLs are only recommended for meeting stringent reference and PD noise reduction requirements given the design complexity. However, thorough design considerations needs to be made before implementing a high order filter in an ADPLL for space applications, because the higher order filter, the more registers (any registers but the one in the proportional path) could result in limit cycle errors. In addition, comparing with ADPLLs with bang-bang or integer-based linear PD, ADPLLs with fraction-based linear PD exhibit more non-linearity and less obvious perturbation time dependence on bit-weight of the perturbed bits.

ADPLLs with a DCO with large DCO gain (K_{DCO}), i.e. few number of digital control bits, covering the same frequency range is preferred in terms of SEU performances. The configuration of FD does not affect the overall SEU perturbation time of the loop significantly. However, it is observed that the loop tend to yield more limit-cycle errors with larger frequency divisors.

If selective radiation-harden-by-design (RHBD) techniques were to be implemented in a 2nd-order ADPLL due to tight area and power requirement, using RHBD flip-flops in the most n significant bits of the integral register in the integral path with

gain of $2^{(-n)}$ and in the register in the fraction-based PD can effectively mitigate limit cycle errors.

Conclusions

In this chapter, three types of single-event-induced errors in ADPLLs are defined - temporary-frequency errors, loss-of-lock errors and limit-cycle errors. Perturbation time metric is proposed to quantify the error signatures.

SETs are analyzed in DCROs and other digital blocks in ADPLLs. SET-induced harmonic oscillation in standalone DCROs are observed in circuit simulation and TPA laser experiment. Closed-loop SET simulations on the DCRO show that SE harmonic oscillations may result in loss-of-lock errors in ADPLLs incorporating a frequency-based linear PD. However, only temporary-frequency errors are observed for bang-bang ADPLLs. Moreover, the upper boundary of probability for the SETs in the other digital block to cause an output error is analyzed showing that SEUs are at least 5-10 times more likely to cause an output error in ADPLLs in modern technologies.

Since SEUs are the main concerns for the digital blocks of the ADPLLs, individual contributions of PD, DLF and FD to ADPLL overall SE performance are analyzed using fault-injection experiment. The most SEU-sensitive module is identified to be the high-order poles in the DLF, which can cause limit-cycle errors at the ADPLL output. SEUs in linear PDs can result in loss-of-lock errors while those in bang-bang PD can only cause temporary-frequency errors. At the same output frequency, changing the frequency divisor does not have significant impacts on linear ADPLLs. However, bang-bang ADPLLs exhibit worse SE performance with increasing of

feedback frequency divisor.

CHAPTER V

OVERALL ADPLL SE CHARACTERIZATION AND ANALYSIS

ADPLL modular SEU and SET sensitivities were discussed in the previous sections. SEU and SET simulations are performed in this section to compare the contribution of each module to others to the overall SE performance of the ADPLL on the same scale. In this chapter, SET simulations are performed in transistor level in Cadence Spectre and SEU injections are performed in mixed Verilog/SPICE simulations with Cadence AMS to achieve fast simulation time and fine simulation accuracy. SEU/SET simulation setups for the ADPLL designs are detailed in section V.1. Section V.2 presents the overall SE characterization results on the ADPLL designs with three different PDs. Analysis and SE characterization for ADPLLs with FSM-based PD is presented in section V.3. The final section concludes the chapter.

SEU and SET Simulation Setup

All the analyzed ADPLL designs in the chapter were designed and synthesized using the IBMCS7RF standard cell library. As shown in Fig. 46, SEU-induced error signatures were studied by performing an exhaustive fault injection campaign[135] by conducting bit flipping at all the bits of the registers in simulation. Over 50 faults were injected randomly over time during phase lock for each flip-flop used in the ADPLL sub-circuits. During these simulation, the output of the ADPLL was monitored to determine the worst-case SEU responses.

SET-induced error for the DCO in the ADPLL designs were studied by injecting

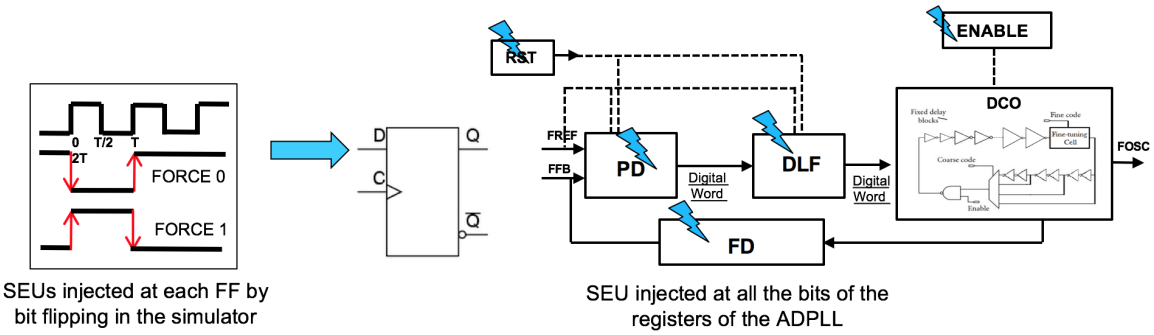


Figure 46: Block diagram illustration of conducted SEU simulation on the ADPLL implementations.

SEE current sources using ISDE bias dependent SEE model at all the internal nodes of DCO over oscillating period when the ADPLL is locked at the desired frequency. As shown in Fig. 47, the output of the ADPLL was monitored to determine the SET-induced error signatures.

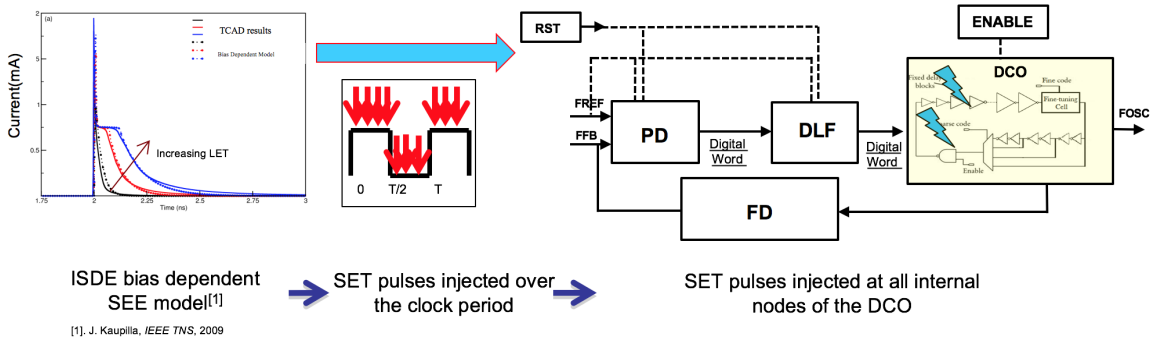


Figure 47: Block diagram illustration of conducted SET simulation on the ADPLL implementations.

Frequency-based and Time-based ADPLLs

Three distinctive implementations of ADPLL were designed and synthesized using the IBMCS7RF standard cell library - bang-bang ADPLL, TDC ADPLL, and linear

ADPLL. The former two are time-based ADPLLs, and the latter one is frequency-based ADPLL. All the designs share the same type of DLF, DCO and FD. They are differentiated by the types of PDs.

All designs use the same type of 10-bit ring-based DCO. The digital tuning of the DCRO is implemented through multiplexing of different ring lengths. The gain of the DCO is 40ns/LSB. The frequency divider is a divide-by-8 module based on a digital counter. The DLF of the the TDC ADPLL and linear ADPLL has a proportional path gain of $2^{(-3)}$ and a integral path gain of $2^{(-8)}$, while that of the bang-bang ADPLL has a proportional path gain of $2^{(-1)}$ and a integral path gain of $2^{(-4)}$. Additionally, a global RESET signal is used to reset all the modules to the initial state, and an ENABLE signal is used to allow for the DCO to be switched on or off.

A fraction-based frequency-linear PD is used in the linear ADPLL and a bang-bang PD is used in the bang-bang ADPLL. In the TDC ADPLL, VHDL behavioral model of the TDC and thermometer-to-binary decoder is used. The TDC is consisted of 512 stages with 9-bit binary output (3 fraction bits). The resolution of the TDC is 200 ns/LSB. SEUs are directly injected at the output 9-bit binary word register of the TDC since all the internal SETs/SEUs must be able to show up at the output register to cause perturbation in the loop. This allows the research to understand how is TDC comparing to other modules in the loop in terms of SEU/SET-induced maximum perturbation time at the output of the ADPLL without going through too much designing and simulation efforts. And as a result, SET vulnerability of the TDC module is not studied.

When single-event perturbation happens in different modules in the ADPLL designs, it can induce different error signatures at the output of the ADPLL. From

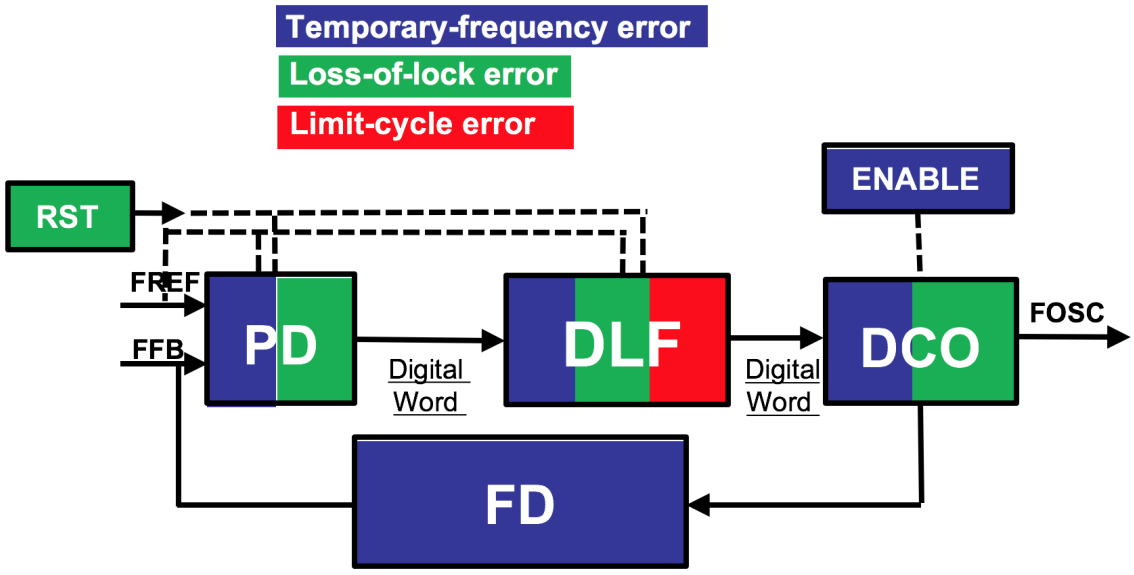


Figure 48: Modular illustration of single-event-induced error signatures for each main subcircuit of the ADPLL designs.

the SE characterization simulations, a colored-mapping of single-event-induced error signatures to subcircuits of ADPLL designs is presented in Fig. 48. As shown in the figure, single-event perturbations in RST global register can induce loss-of-lock errors, while those in FD and ENABLE for the DCO can only result in temporary frequency errors. Meanwhile, SE perturbations in all PD, DLF and DCO modules can lead to temporary frequency errors and loss-of-lock errors but only those in DLF can cause limit-cycle errors.

The overall SE characterization results through SEU/SET simulations are presented in Fig. 49 for the three ADPLL designs. As shown in Fig. 49, the SE performance of all three 2nd-order ADPLLs is dominated by SEUs in corresponding to the 2nd-order pole in the PI filter of the DLF, as limit-cycle errors are observed only in this register. SE perturbation in the 1st-order pole of the DLF, RST and DCO can result in both temporary frequency errors and loss-of-lock errors, while the other

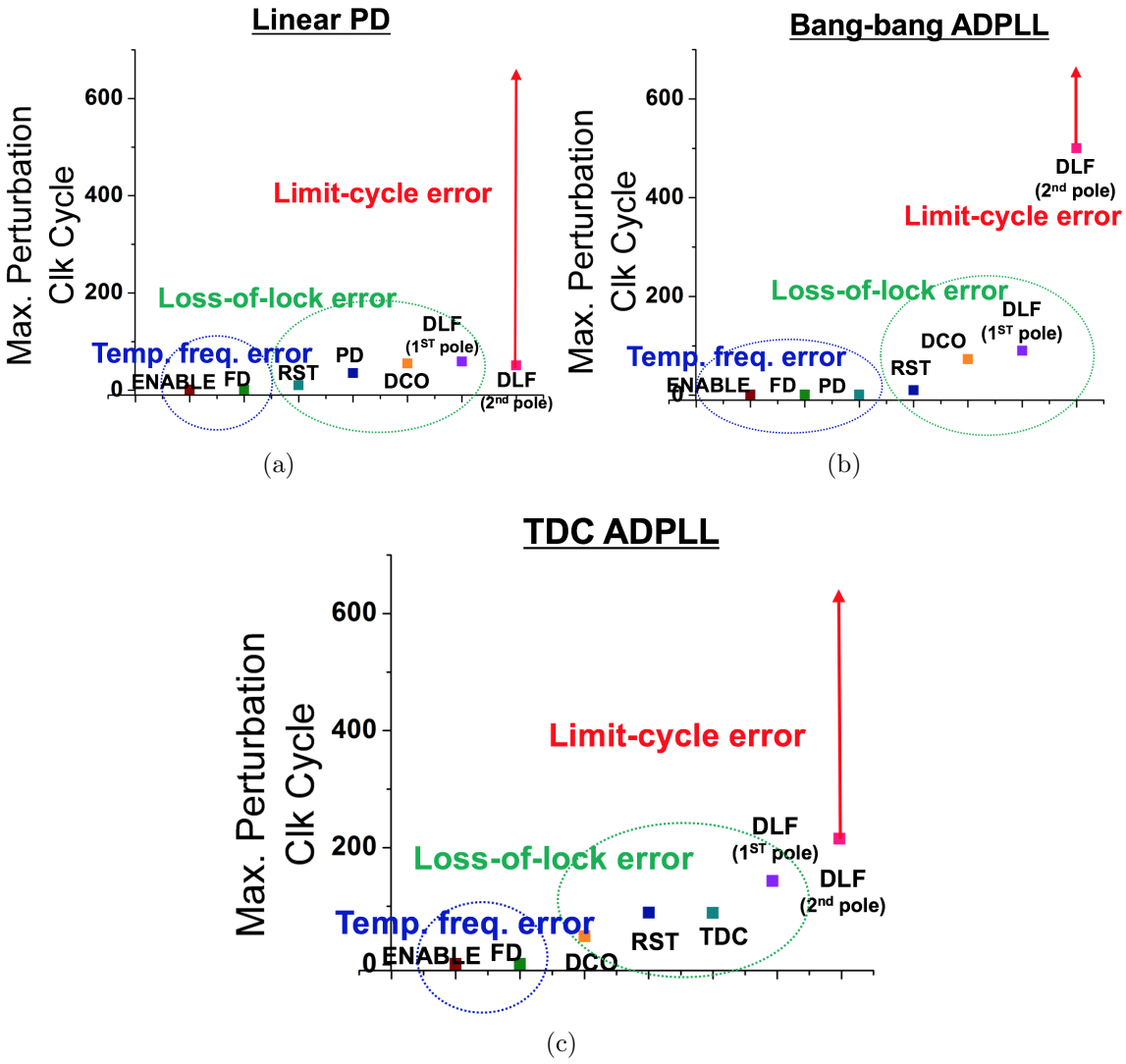


Figure 49: SE-induced maximum perturbation time at the output of ADPLL in terms of reference clock cycles for each module (a) linear 2nd-order ADPLL, (b) bang-bang 2nd-order ADPLL and (c) TDC 2nd-order ADPLL.

modules (ENABLE and FD) can only result in temporary frequency errors. As the ADPLL must undergo complete re-acquisition following faults in the RST register, the output perturbation time is equal to the start-up time of the ADPLL, therefore causing a loss-of-lock error. However, when faults are injected in FFs contained in FD or bang-bang PD sub-circuits, this abrupt change must propagate through the DLF before having an impact on the output frequency. The integrating nature of DLF removes effects of all such SEUs over the control word within a few clock cycles, resulting in temporary-frequency errors lasting only a few clock cycles. Similarly, an SEU in the ENABLE circuit results in the ADPLL output stuck at logic "1" or "0" for 1 clock cycle, during which the DCO does not accept any changes in the control signals. During the next clock cycle, the ENABLE signal is restored to its original value while the information stored in the all other internal registers of ADPLL remains unchanged, resulting in restoration of proper operation of ADPLL.

Bang-bang PD has better performance than linear PD and TDC, as SEUs in the bang-bang PD can only result in temporary frequency errors while those in the latter implementations can induce loss-of-lock errors. In total, time-based ADPLLs yield worse SE performance, i.e. longer perturbation time, comparing to frequency-based ADPLLs, since phase tracking usually dominates the tracking time. The analyzed bang-bang performs slightly better in system settling time due to its larger bandwidth, which is determined by the ratio of the proportional and integral path gain in the loop filter, comparing to TDC ADPLL.

FSM-based ADPLLs

As stated in Chapter II, it is common practice in ADPLLs to use frequency-based PD for frequency error correction and then switch to time-based PD (such as bang-bang PD or TDC) for phase alignment. Because this allows the ADPLL to achieve true phase-lock state comparing with the loops using just frequency-based PDs, and also result in optimum loop settling time comparing with those using just time-based PDs. A block diagram for such ADPLL design is shown in Fig. 50.

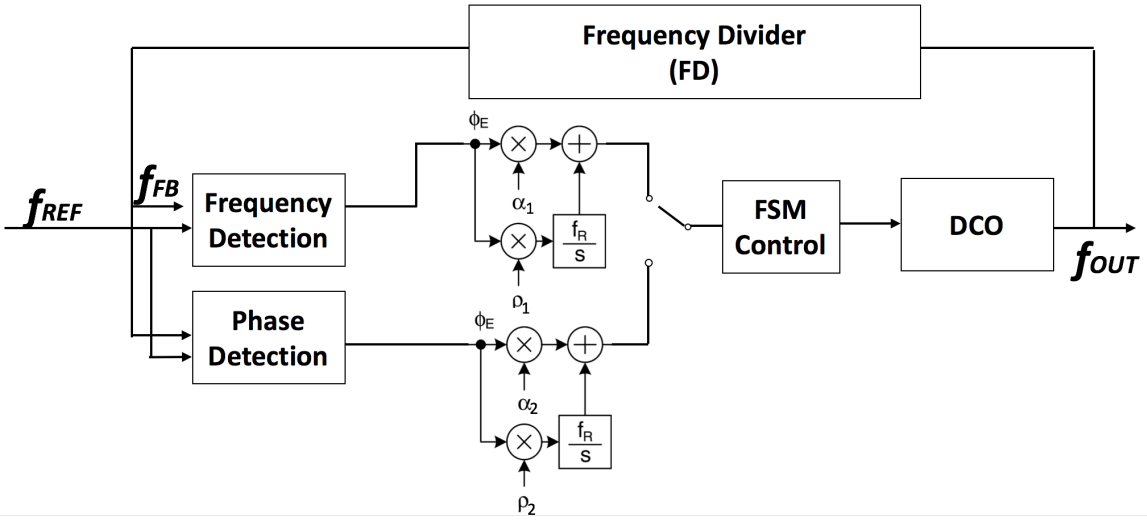


Figure 50: Block diagram illustration of an ADPLL incorporating an FSM switching between frequency detection mode and phase detection mode.

As shown in the figure, an FSM is in charge of switching between the frequency detection mode and the phase detection mode. When the frequency-based PD continues to outputs a digital control word within a small range for over several reference clock cycles, it indicates the loop is in frequency lock. The FSM thus generate a flag signal to switch from frequency detection to phase detection with slower word adjustment and narrower loop bandwidth ($\alpha_1 \ll \alpha_2$ and $\rho_1 \ll \rho_2$). Usually, the PLL undergoes the frequency tracking and then phase tracking at start

up. And the phase tracking process will correct any phase shifts of the output clock signal resulted from noise perturbations during the operation of the design. Therefore, the FSM is usually designed in a manner that the loop goes through frequency tracking once at startup and stays in phase tracking during operation.

An ADPLL with an FSM controller was designed for SE characterization. The ADPLL uses fraction-based linear PD for frequency detection and a bang-bang PD cascaded with TDC for phase detection. Other specifications for the design is listed in the table below.

Table 6: FSM ADPLL design specifications.

Design Module	Design Specifications
DCO	10-bit, 40ns/LSB
TDC	512 stages, 9 bits (3 fraction bits), 10ns/LSB
DLF1	$\alpha_1 = 2^{-1}$, $\rho_1 = 2^{-4}$
DLF2	$\alpha_2 = 2^{-3}$, $\rho_2 = 2^{-9}$
FD	division of 8

Since the FSM controls the ADPLL to operate either in frequency-tracking or phase-tracking mode, only one path in Fig. 50 is active at a specific time. In fact, when the PLL is locked, the loop stays in the phase-tracking mode, which means the ADPLL is not sensitive to SEUs or SETs in the frequency-tracking path. When the FSM is designed such that the loop goes into the phase-tracking mode and stays there, the ADPLL responds to SEUs/SETs in PD, DLF, DCO, and FD the same way as a TDC ADPLL analyzed in Section V.2. However, if the FSM is designed such that the loop can switch between phase and frequency tracking freely, when the ADPLL goes out of lock due to SEU/SET perturbations, the loop would go through frequency tracking and then phase tracking.

SETs and SEUs in the FSM controller perturb the flag signal for controlling the

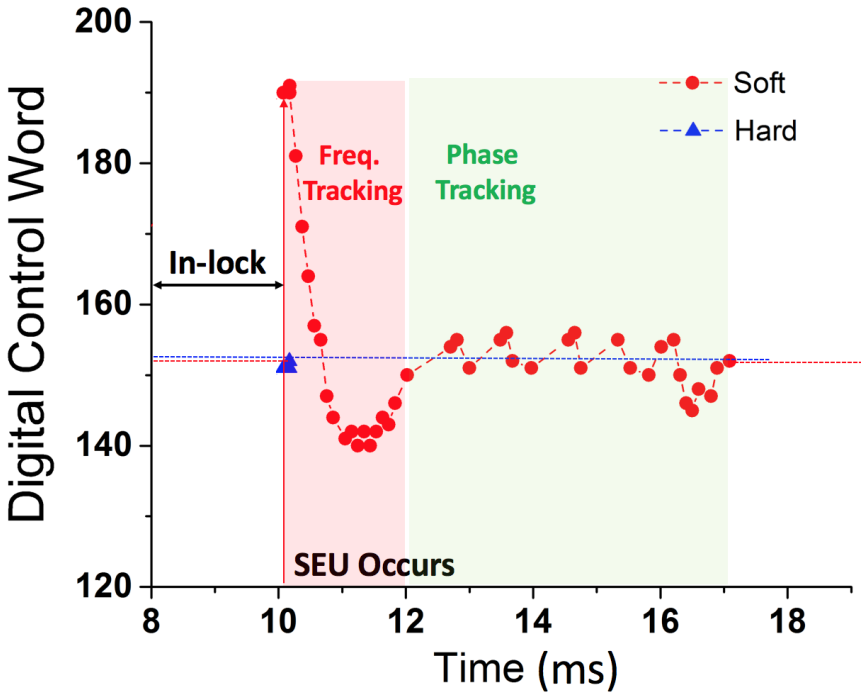


Figure 51: ADPLL responses to a SEU perturbation for the flag signal in the FSM controller. The red curve indicates the ADPLL go through frequency tracking (light pink block) and phase tracking (light green block) after SEU perturbation in the flag signal. The blue data points are for the modified ADPLL indicating the ADPLL stayed in lock the entire time in spite of the perturbation.

tracking modes of the ADPLL. Fig. 51 indicates the ADPLL response to an SEU perturbation for the flag signal. For a "soft" design, upon the occurrence of an SEU in the flag signal at around 10 ms, the ADPLL goes through frequency tracking (light pink block) and phase tracking (light green block). This is because the SEU in the flag signal resulted in the ADPLL changing from phase tracking mode to frequency mode, while, as shown in Fig. 50, the DLF corresponding to the frequency-based PD is not tracking the current digital control word when the loop operates in the phase tracking mode. By implementing a multiplexer in the feedback path of the original DLF, as shown in Fig. 52, the DLF in the frequency-tracking path can be updated with the current control word when the loop is in-lock in the phase-tracking mode. As stated, since this is implemented in the frequency-tracking path, this is not added sensitive area to SEUs/SETs during normal operation of ADPLL. The blue data point in Fig. 50 is for the "hard" ADPLL with the added MUX, indicating the ADPLL stayed in lock the entire time in spite of the perturbation.

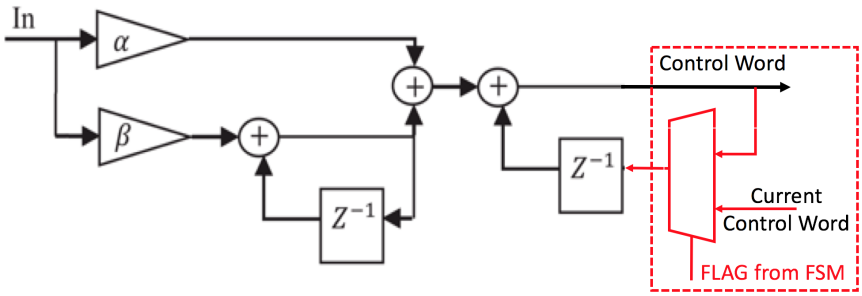


Figure 52: Storing the current digital control word in the DLF in the frequency-tracking path by implementing a multiplexer in the feedback path of the DLF.

As stated in chapter II, this type of FSM-based ADPLL can also be implemented using TDCs [4], on which similar analysis can also be applied.

Conclusions

Overall SE characterization using SEU/SET simulation was performed on three 180nm 2nd-order ADPLLs using bang-bang, TDC and fraction-based linear PDs. The three designs exhibit very similar SE performance in terms of output perturbation time. Their SE performance is dominated by the 2nd-order pole in the loops. However, SEUs in the bang-bang PD can only result in temporary frequency errors while those in the TDC or linear PD can induce loss-of-lock errors.

SE characterization is also applied to ADPLLs with FSM-based PDs. During steady-state operation, the loop stays in phase tracking mode and it is not sensitive to SEUs/SETs in the modules on the frequency-tracking path. Adding a multiplexer in the DLF in the frequency-tracking path of the design allows the design to be tolerant to SEUs in the FSM controller.

CHAPTER VI

TIME-DOMAIN MODEL FOR SEUS IN ADPLLS

As stated in previous chapters, ADPLLs have been proven to be vulnerable to single-event effects. As the operational reliability of all electronic systems is highly dependent on the reliability of the clock signal [23], it is crucial to understand how different design parameters affect the SEU response of typical ADPLLs and thus enable designers to design for SEU-tolerant ADPLLs.

Most of modeling work in the literature directly translates the phase-domain linear model for analog PLLs to discrete-time equivalent z-domain for ADPLLs [58][52], which do not apply well for non-linear ADPLLs like bang-bang ADPLLs. In addition, Z-domain models do not easily yield information related to timing of signals and their interactions. For single event (SE) environments, timing of SE transients and their effects on the circuit operation are best modeled in time-domain for accuracy. Existing time-domain models [61][144] are often topology-specific and described in algorithms that do not yield designers closed-form relationship between system output and loop design parameters. This chapter presents a novel time-domain modeling methodology for frequency-based and time-based ADPLLs, which is independent of technology process. In the presence of SEUs, the model can also be used to quantify the perturbation due to SEUs originating from different modules on the system in locked state, in terms of phase errors. The model provides designers with full characterization of ADPLL SEU responses during the design stage and identifies the very sensitive sub-circuits in the system where hardening techniques

should be applied with priority. The model results in a simplified expression of the phase error at each reference cycle to provide means to estimate the settling speed (i.e. relock time) of the loop, or whether the loop would regain lock or not (i.e. loop stability) in response to an SEU. The correlation between SEU location, perturbed loop parameters, and the resulting phase error is also discussed. Verification of the model was conducted through FPGA-based fault injection experiment and TPA laser tests on ADPLL using TDC circuit. The proposed time-domain methodology can also be easily used within a statistical design flow incorporating complex radiation effects.

System Modeling for ADPLLs

A unified time-domain model is proposed for both frequency-based and time-based ADPLLs that can be used for behavioral analyses, as in [145][61].

In a frequency-based ADPLL, the frequency tuning (Y_{fi}) by the frequency-based PD and DCO (with gain of k_{DCO}) and FD (with frequency division of M) for reference cycle i is given by

$$Y_{fi} = \frac{1}{M} PD_{OUT(i-1)} k_{DCO}, \tag{31}$$

in which $PD_{OUT(i-1)}$ is the output from the PD for at reference cycle i. Consequently, at reference cycle i, the frequency tuning (Z_{fi}) by a 1st-order ADPLL, in which the DLF only has a proportional path with a gain of α, is given by

$$Z_{fi} = \alpha Y_{fi}. \tag{32}$$

And if the ADPLL is 2nd-order incorporating a DLF with proportional gain of α and

integral gain of ρ is given by

$$Z_{fi} = \alpha Y_{fi} + \rho \sum_{j=1}^{i-1} Y_{fj}. \tag{33}$$

At the same time, the frequency error, i.e. the cycle-to-cycle phase error, between consecutive reference cycles i-1 and i results from the frequency tuning (Z_{fi}). Therefore,

$$E_{fi} = E_{f(i-1)} - Z_{fi}. \tag{34}$$

From Eqn. 33 and Eqn. 34, we have

$$E_{fi} - E_{f(i-1)} = -(\alpha Y_{fi} + \rho \sum_{j=1}^{i-1} Y_{fj}). \tag{35}$$

Based on the ADPLL design topology described in Chapter II, the loop requires one reference cycle to react to the perturbation, which means one clock cycle latency exists between the detection of the frequency error and the output for the frequency tuning. The actual recursive function for the design becomes

$$E_{fi} - E_{f(i-1)} = -(\alpha Y_{f(i-1)} + \rho \sum_{j=1}^{i-2} Y_{fj}), \tag{36}$$

where E_{f0} and E_{f1} are initial frequency errors induced by SEU perturbations.

For a time-based ADPLL, since all the control modules operate on cycle phase errors instead of frequency errors (i.e. cycle-to-cycle phase errors), the system function Eqn. 36 becomes Eqn. 37,

$$E_{pi} - E_{p(i-1)} = -(\alpha Y_{p(i-1)} + \rho \sum_{j=1}^{i-2} Y_{pj}), \tag{37}$$

in which Y_{pi} is phase-tuning by the time-based PD and DCO (with gain of k_{DCO}) and FD (with frequency division of M) for reference cycle i, and cycle phase error E_{pi}

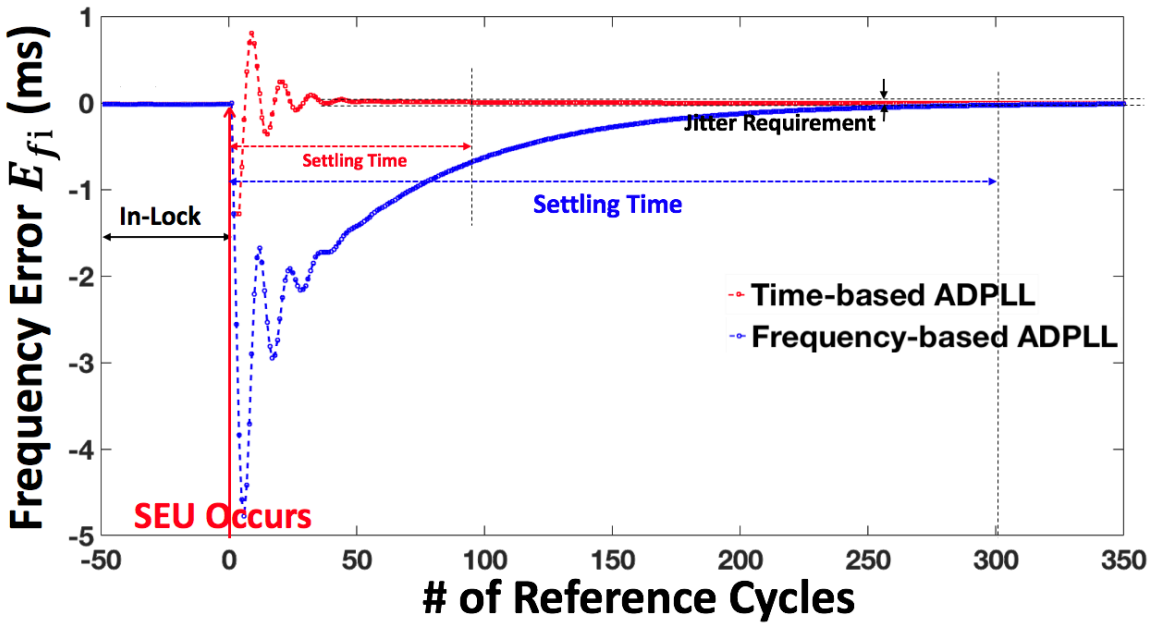
is the integral of frequency error, E_{fi} is the summation of all the cycle phase errors from startup of the PLL, shown in equation Eqn. 38,

$$E_{pi} = \sum_{j=1}^{i} E_{fj}. \tag{38}$$

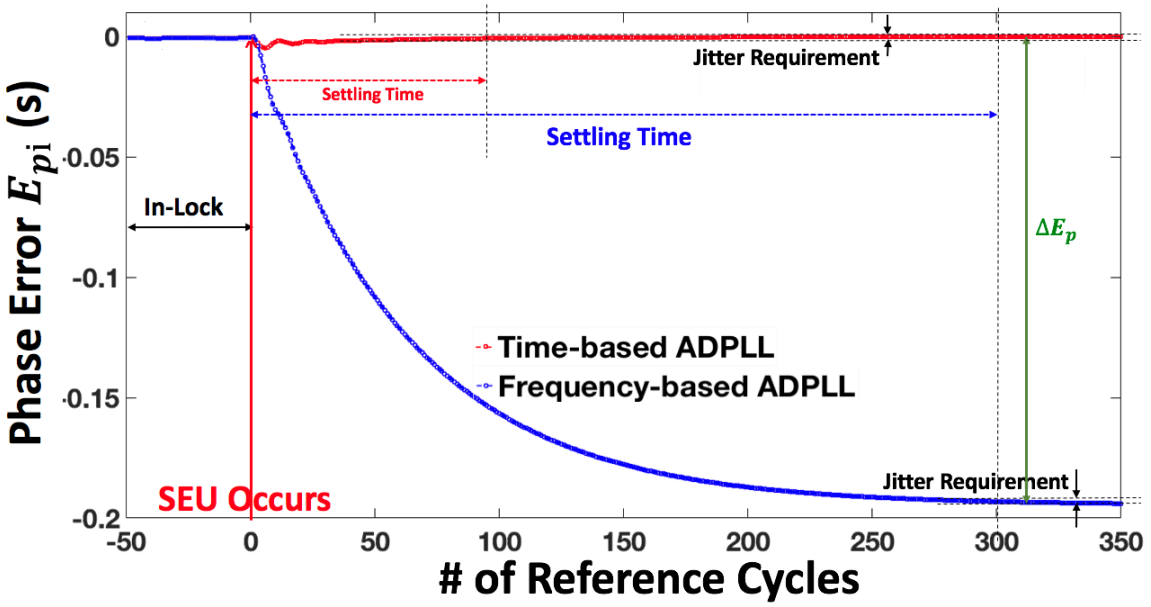
The typical responses of both a 2nd-order frequency-based ADPLL and a 2nd-order time-based ADPLL in a locked state to an initial step phase/frequency error with magnitude E_0 at time 0 (which can be resulted from SEU or simply noise perturbantion) are shown in Fig. 53. Plotted in Fig. 53a is the frequency error (E_{fi}) at the current reference cycle (i) and in Fig. 53b the phase error at the current reference cycle (i), i.e. E_{pi}.

Before time 0, the ADPLL is in-lock, i.e. $E_{f0}^{-} = 0$, and E_{p0} is a small, finite number bounded by the resolution of the PD for both frequency-based and time-based ADPLLs. After the initial phase/frequency perturbation, the systems starts to self-compensate for the phase/frequency error. Locked state of the system is achieved when E_{fi} or changes in E_{pi} is within the defined jitter requirement (usually <5% of the desired period). And system settling time is defined from the time of perturbation to the time it regains lock. As expected, when the loop regains lock, the time-based ADPLL exhibits $E_f = 0$ and $E_p = 0$, while the frequency-based ADPLL exhibits $E_f = 0$ with a constant phase shift ΔE_p.

In fact, the two example responses plotted in Fig. 53a are for systems with the same coefficients in the same recursive function. The only difference is that phase error E_{pi} is the system function operator for time-based ADPLL instead of frequency error E_{fi} for frequency-based ADPLL. However, it is shown on the plot that the time-based ADPLL settles significantly faster comparing with frequency-based ADPLL



(a)



(b)

Figure 53: System response in terms of (a) frequency error and (b) phase error to step phase/frequency error magnitude E_0 over reference cycles for both frequency-based and time-based ADPLLs. Settling time is defined from the time of the SEU to when the system phase error settles back to within jitter requirement.

corresponding to the same initial phase/frequency perturbation.

A. Frequency-based ADPLL

A frequency-based linear PD with a gain of k_{PD} outputs $k_{PD}E_{fi}$ corresponding to an input frequency error of E_{fi}. Since usually for frequency-based ADPLLs, the frequency-division is embedded in the PD, there for the 1/M factor is omitted from Eqn. 31. Therefore, Eqn. 31 for the frequency tuning (Y_{fi}) by the PD,DCO and FD for reference cycle i is given by

$$Y_{fi} = k_{PD}k_{DCO}E_{f(i-1)}. \tag{39}$$

Therefore, the system function Eqn. 36 becomes

$$E_{fi} - E_{f(i-1)} = -(\alpha k_{PD}k_{DCO}E_{f(i-2)} + \rho \sum_{j=1}^{i-2} k_{PD}k_{DCO}E_{fj}). \tag{40}$$

From Eqn. 31 and Eqn. 39, values of E_{fi} can be found, in the form of

$$E_{fi} = \frac{1}{A-B}[A^i(AE_{f1} - BE_{f0}) - \frac{B}{A}^i(E_{f1} - AE_{f0})]. \tag{41}$$

where E_{f0} and E_{f1} system initial conditions and parameters A and B are dependent on the loop parameters, i.e., α, ρ, k_{PD} and k_{DCO}. And criteria for system stability can be determined from A and B for a given E_{f0} and E_{f1}.

$$A, B = \frac{1}{2}[(2 - \alpha k_{PD}k_{DCO}) \pm \sqrt{\alpha k_{PD}{k_{DCO}}^2 - 4\rho k_{PD}k_{DCO}}]. \tag{42}$$

B. Time-based ADPLLs

One implementation of time-based ADPLLs utilized TDC as the PD, which operates linearly with a gain of k_{PD} corresponding to incoming phase error E_p (i.e. $\sum E_f$). The gain of the TDC in time domain is shown in Eqn. 43

$$k_{PD} = \frac{1}{\Delta_{TDC}}, \tag{43}$$

in which Δ_{TDC} is the TDC resolution. Therefore, for a TDC, the input and output relationship becomes

$$Y_{pi} = \frac{1}{M} PD_{OUT(i-1)} k_{DCO} = \frac{1}{M} k_{PD} E_{p(i-1)}, \tag{44}$$

Therefore, the recursive function Eqn. 37 for the system becomes

$$E_{pi} - E_{p(i-1)} = -(\alpha k_{PD} k_{DCO} E_{p(i-2)} + \rho \sum_{j=1}^{i-2} k_{PD} k_{DCO} E_{pj}). \tag{45}$$

It is worth noting that Eqn. 40 and Eqn. 45 are essentially the same functions. The difference between them is that Eqn. 40 operates on frequency error (E_{fi}) and Eqn. 45 on phase error (E_{pi}).

The other implementation of time-based ADPLLs utilized bang-bang PD. As stated, a bang-bang PD always outputs a single bit for any given input phase error.

$$PD_{OUT} = \begin{cases} k_{PD} & \sum_{j=1}^{i-1} E_{fj} = E_{p(i-1)} > 0 \\ -k_{PD} & \sum_{j=1}^{i-1} E_{fj} = E_{p(i-1)} < 0 \end{cases} \tag{46}$$

Combining Eqn. 46 with Eqn. 31, Eqn. 31 becomes

$$Y_{pi} = \begin{cases} k_{DCO} k_{PD} & E_{p(i-1)} > 0 \\ -k_{DCO} k_{PD} & E_{p(i-1)} < 0 \end{cases} \tag{47}$$

Based on Eqn. 47 and Eqn. 37 , the system model can be constructed for bang-bang ADPLLs.

Modeling for SEUs in Different Sub-modules of ADPLLs

While loop response has been shown to dominate the error response of the system in continuous time/analog PLLs [107], this work uses a similar approach for creating

a model to describe the SEU-related error response by combining the time-domain equations with SEU-induced system initial condition and loop parameter shifts. An SEU in the n^{th} least significant bit in the register can induce digital word perturbation at the output of the register of $\pm\delta = \pm 2^n$ (n=0,1,2...) depending on if it is a 1-0 or a 0-1bit flip. Based on the SEU error analysis in previous chapters and the proposed time-domain system response model, SEU perturbations in different sub-modules of both frequency-based and time-based ADPLLs are analyzed as follows.

A. SEUs originating at different modules of the ADPLL system are all modeled by the SEU-induced initial phase/frequency error E_0 at the input of the PD.

B. SEUs in the PD module are modeled differently for frequency-based and time-based ADPLLs

(1) SEUs in frequency-based PD can only induce erroneous $\pm\frac{D_{center}}{\delta}k_{PD}$ for 1 cycle at PD output. Therefore, after propagating through the loop, E_{f0} for SEUs in PD is

$$E_{f0} = \pm\frac{D_{center}}{\delta}\alpha k_{PD}k_{DCO}E_{f(i-1)}, \tag{48}$$

(2) SEUs in time-based PD can only induce erroneous $\pm\delta k_{PD}$ for 1 cycle at PD output. Bang-bang PD is a special case of time-based PDs, where $\delta = 1$. However, the loop operates on phase rather than frequency. Therefore, after propagating through the loop, initial phase error input E_{p0} for PD is

$$E_{p0} = \pm\delta\alpha k_{PD}k_{DCO}E_{p(i-1)}, \tag{49}$$

C. SEUs in any modules but PD can modeled in the same fashion for both frequency-based and time-based ADPLLs

(1) SEUs in FD can cause duty cycle errors on the output clock signal from the FD (f_{FB} in Fig. 2a). Since the output of FD directly serves as the input to PD, therefore,E_0 for SEUs in FD is

$$E_0 = \pm\frac{1}{\delta}\frac{T_{REF}}{2}. \tag{50}$$

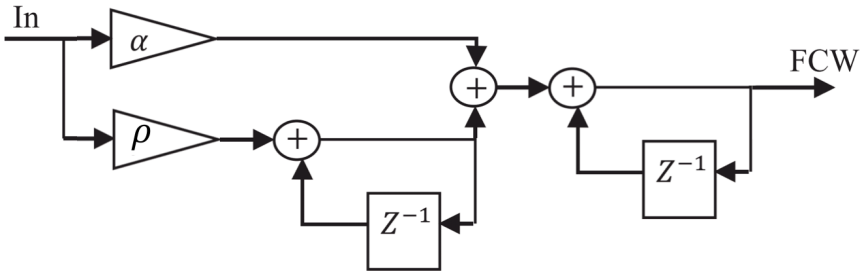


Figure 54: Block diagram of a 2nd-order digital loop filter with proportional path gain of α and integral path gain of ρ.

(2) SEUs in DLF can result in different ADPLL responses based on the location. A word perturbation 2^n in all the registers in the DLF (represented as z^{-1} in Fig. 54) can induce a $\delta = 2^n$ frequency perturbation. Therefore, the SEU-induced initial phase/frequency error for SEUs in DLF becomes

$$E_0 = \pm\delta k_{DCO}. \tag{51}$$

(3) In addition to introducing a word perturbation $\delta = 2^n$, SEUS in the DLF integrator register with gain of ρ (shown in the left of Fig. 54) can also affect the proceeding frequency tuning steps due to the integrating effect. This results in Eqn. 40 becomes Eqn. 52 and Eqn. 45 becomes Eqn. 53 for frequency-based and time-based ADPLLs.

$$E_{fi} - E_{f(i-1)} = -(\alpha k_{PD}k_{DCO}E_{f(i-2)} + \rho\sum_{j=1}^{i-2} k_{PD}k_{DCO}E_{fj}) + \delta k_{DCO}. \tag{52}$$

$$E_{pi} - E_{p(i-1)} = -(\alpha k_{PD} k_{DCO} E_{p(i-2)} + \rho \sum_{j=1}^{i-2} k_{PD} k_{DCO} E_{pj}) + \delta k_{DCO_2}. \quad (53)$$

(4) SEUs in DCOs are not modeled, since DCOs are conventionally implemented with logic-only circuits such as LC-tank oscillators and ring oscillators. Even though SETs in DCOs may potentially induce phase or frequency errors, fault injection on FPGA tiles are not representative of the SETs in typical integrated circuit designs, they are not considered here and SEU response of ADPLLs is the focus of this work.

For convenience, the above system of equations that represent the proposed model can be easily implemented in MATLAB to provide the system response for a given SEU perturbation location. Overflow in the design, i.e. wrap around issues due to not enough bits in the accumulator, can also be modeled in the program using conditional statements to facilitate modeling accuracy.

Model Verification

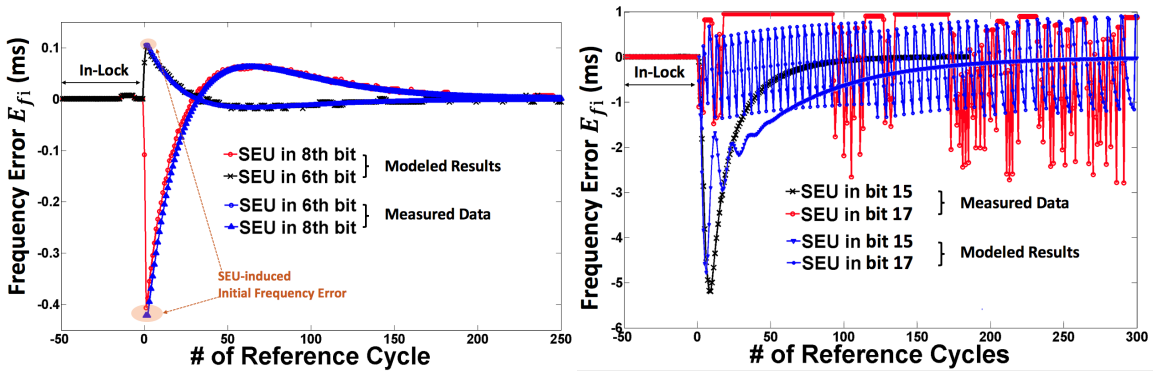
In this section, hardware-based experimental results are gathered for both frequency-based and time-based ADPLLs to verify the proposed time-domain model. FPGA fault injection experiments were performed on frequency-based ADPLL and a bang-bang ADPLL, while TPA laser experiment was performed on a TDC ADPLL. Since a 1st-order ADPLL is a special version of 2nd-order ADPLL (i.e. with no register bits implemented for the integral path), though results on 2nd-order ADPLL designs are presented, the proposed model and verification can also be applied to 1st-order ADPLLs.

Frequency-based ADPLL

A frequency-based ADPLL design using fraction-based frequency PD was synthesized using the IBMCS7RF standard cell library to instantiate flip-flop topologies representative of each design module. The design was then synthesized on an Altera DE2-115 FPGA and the fault-injection experimental setup was the same with that in Chapter IV and V.

The control word that corresponds to the center frequency, D_{center}, is chosen in replacement of D in Eqn. 4 for design simplicity of the synthesized fraction-based frequency PD. In addition, the output of the frequency-based PD outputs is normalized against the DCO tuning resolution, in which case $k_{PD}k_{DCO} = 1$. The ADPLL uses DLF with proportional gain of 2^{-3} and integral gain of 2^{-8}, implemented with 18-bit register (bit 0-17) with 8 fraction bits. The FD incorporates a divide-by-8 counter structure. The ADPLL designs utilize a 10-bit counter-based DCOs with a frequency tuning range of 35.6Hz$\sim$10.5MHz. Fault injections were conducted at when ADPLL was locked at DCO frequency of 128 Hz.

SEUs in DLF registers are used as examples to illustrate the modeled and measured time responses of the synthesized design in Fig. 55. The system is modeled as explained in Section III. The modeled and measured ADPLL response times, in terms of output frequency error with respect to the number of reference cycles, for SEUs originating in the DLF output register bit 6 and 8 are plotted in Fig. 55a. The initial frequency error for bit 6 is estimated to be 100 μs according to Eqn. 36, which agrees well with the measured data. The overall time response of the ADPLL system shows good agreement between the modeled results and measured data. Similar trend was observed for SEU at bit 8.



(a) SEUs in DLF output register

(b) SEUs in DLF integral register

Figure 55: Measured time response of frequency-based ADPLL in terms of frequency error over reference cycles towards SEUs in (a) bit 8 and bit 6 in DLF output register and (b) bit 17 and bit 15 in DLF integral register.

Shown in Fig. 55b is the output frequency error for SEUs in DLF integrator register bit 15 and 17. The model successfully predicts that for current design topology, SEU in bit 17 can result in limit cycle errors, which requires system reset [115], while SEU in bit 15 only results in loss-of-lock errors. In fact, this is illustrated in Fig. 56 as well. For SEUs in both PD and DLF, Fig. 56 compares modeled values with measurements for the perturbation time in number of reference cycles over the SEU bit location. Overall, calculated values from the proposed model show good agreement with measured values. The model successfully predicts that SEUs in the most significant two bits in the DLF integral register can induce limit-cycle errors for current design topology. Also, limit-cycle errors resulting from SEUs in bit 16 dies out after hundreds of reference cycles, while limit-cycle errors continue until system reset for SEUs originating in bit 17, which agrees perfectly with the measured data as indicated in the shaded region in Fig. 56. Through modeling, it is found that an overflow of adders is the main cause of limit-cycle errors. Without this issue, the ADPLL will eventually re-lock onto the desired frequency after a long period of

time. During the fault injection experiment, only temporary-frequency errors were observed for SEUs in the FD module of the ADPLL, for which the modeled and measured perturbation time both show less than 3 reference cycles.

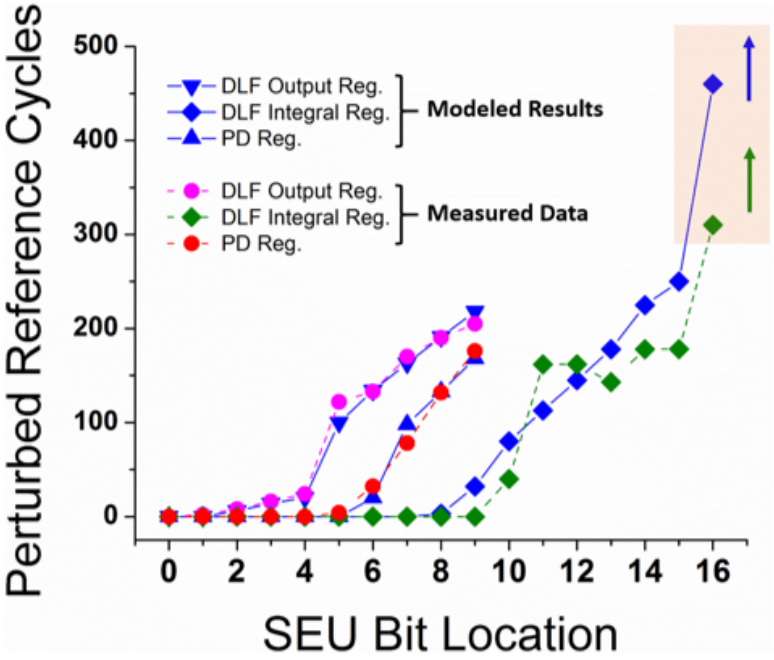


Figure 56: Modeled and measured perturbation time in number of reference cycles over the SEU bit location in different registers in DLF and PD for the synthesized linear ADPLL design. Limit-cycle errors are indicated in the shaded region, where the everlasting limit-cycle errors are indicated with arrows pointing to infinity.

TDC ADPLL

Two photon absorption (TPA) laser experiments [146] were conducted at Vanderbilt University on an ADPLL design from Boeing Company fabricated on IBM 32nm SOI technology. The laser spot size (diameter of the laser where the incident energy is 1/e of the peak value) is estimated to be approximately 1.2 μm at a signal wavelength of 1260 nm; available pulse energies at this wavelength can exceed μJ levels [126].

The functional block diagram of the ADPLL design is shown in Fig. 57. As

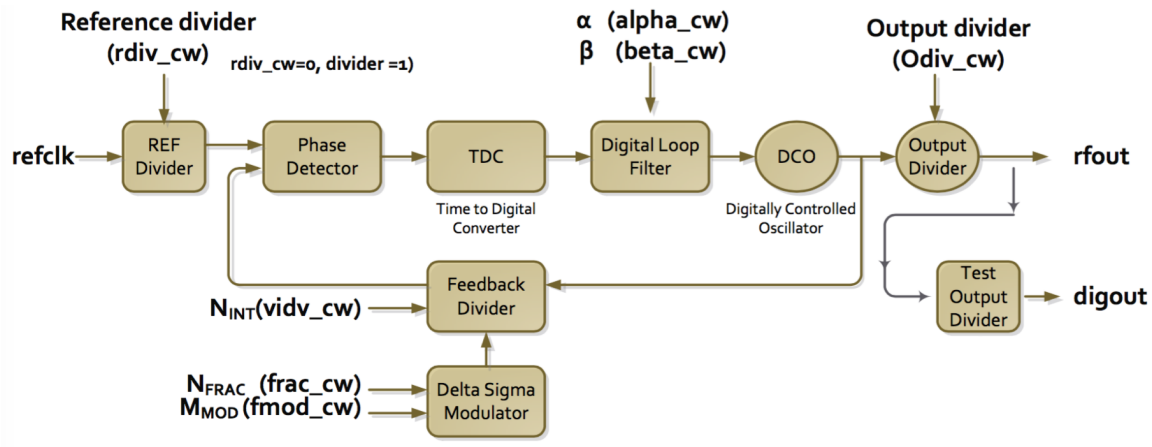


Figure 57: Functionality block diagram of the 32nm SOI ADPLL under test.

indicated in the diagram, the reference clock (refclk) goes through a reference clock pre-divider (rdiv) before it enters the ADPLL loop. The 2nd-order ADPLL incorporates a DLF with proportional path gain of $2^{(-\alpha)}$and integral path gain of $2^{(-\rho)}$). In the feedback path, an integer divider and a delta-sigma modulator are incorporated to allow for both integer and fractional mode operation of the ADPLL. DIGOUT output is monitored, which is the ADPLL output divided through the combination of output divider (i.e. divide-by-4) and test output divider (i.e. divide-by-4). Therefore, the operation of the ADPLL under test is summarized as Eqn. 54 below

$$f_{DIGOUT} = \frac{1}{16}\frac{f_{REFCLK}}{rdiv}(N_{INT} + \frac{N_{FRAC}}{N_{MOD}}). \tag{54}$$

During testing, the core supply voltage of the design was 0.92 V and the IO voltage was 1.73 V. Since the ADPLL is optimized for reference frequencies from 60 MHz to 200 MHz and the LC-tank DCO operates from 8 GHz to 12 GHz, a 120 MHz reference signal (f_{REF}) was generated from an Agilent AFG3252 function generator for the ADPLL. TPA laser testing was performed on the design at parameters listed

in Table 7. The expected frequency at the output was 600 MHz, according to Eqn. 54. All experiments were performed at room temperature when the PLL was locked at the desired frequency.

Table 7: Operating mode of Boeing's ADPLL design under laser test.

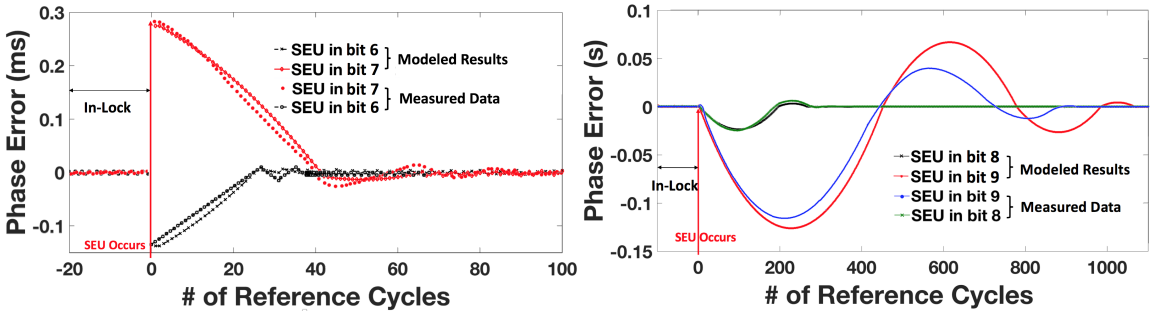
Parameters	α	ρ	N_{INT}	N_{MOD}	rdiv	N_{FRAC}
Values	4	12	80	1	120	1
f_{REFCLK} (MHz)	120					
f_{DIGOUT}(MHz)	600.06					
f_{ADPLL} (MHz)	9601					

Raster scanning was performed on the feedback frequency divider module (as shown in Fig. 57) with step size of 1 μm in both x and y direction. During the scanning, for each incidence locations, 300 fast frames of the monitored DIGOUT output were acquired for different incidence laser energies. Worst ADPLL response to laser-induced SEUs at laser incident energy in the range of 2.6 nJ^2 to 28.6 nJ^2 is calculated based on the output perturbation time for each laser incidence at each scanned location.

The maximum observed perturbation time for SEUs in FD of the ADPLL under test is 10.2 ns, which matches the modeled perturbation time results of around 8.3 ns with calibrated TDC resolution of 10 ps/LSB and DCO gain of 2 Hz/LSB.

Bang-bang ADPLL Model Verification

Similar to the linear ADPLL, a bang-bang ADPLL design was first synthesized using the IBMCS7RF standard cell library and then synthesized onto an Altera DE2-115 FPGA. Similar at-speed fault injection test campaign was performed on each



(a) SEUs in DLF output register

(b) SEUs in DLF integral register

Figure 58: Measured time response of bang-bang ADPLL in terms of phase error over reference cycles towards SEUs in (a) bit 6 and bit 7 in DLF output register and (b) bit 8 and bit 9 in DLF integral register.

flip-flop used in the ADPLL sub-circuits. And ADPLL output was monitored to compare with the modeled ADPLL SEU responses.

This bang-bang ADPLL uses DLF with proportional gain of 2^{-1} and integral gain of 2^{-4}, implemented with 11-bit register (bit 0-10) with 4 fraction bits. The frequency tuning range of the ADPLL design is 569.6 Hz ∼10.5 MHz, which is realized through a 7-bit counter-based DCOs. Fault injections were conducted at 1.25 MHz DCO frequency.

As Fig. 55, the modeled and measured ADPLL response times to SEUs in DLF registers for SEUs originating in the DLF output register bit 6 and 7 are plotted in Fig. 58a. For SEU at bit 6 and 7, the estimated initial frequency error according to Eqn. 42 and the overall time response agrees well with the measured data.

Shown in Fig. 58b is the output phase error for SEUs in DLF integrator register bit 7 and 9. In fact, SEUs in either bit 9 or bit 7 can result in limit cycle errors and they both die away with time. The model successfully predicts that for this bang-bang ADPLL. And the measured and modeled overall perturbation time for both

cases show good agreement.

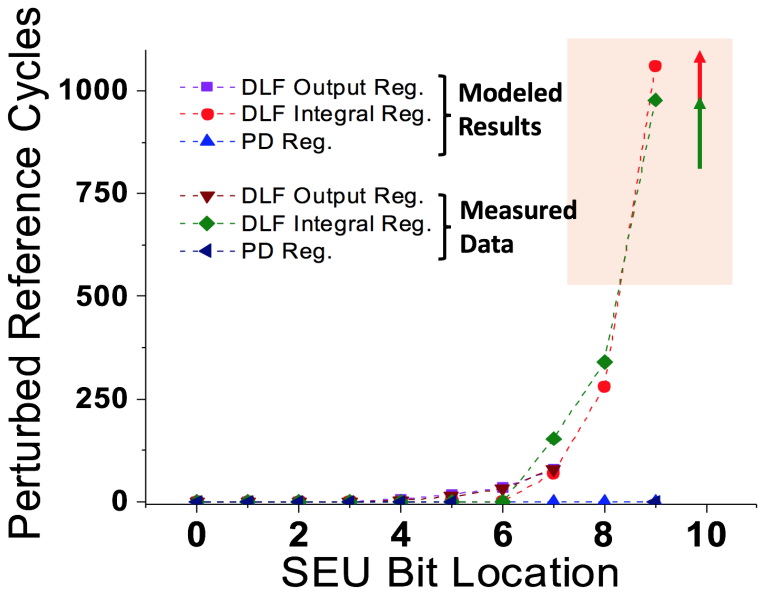


Figure 59: Modeled and measured perturbation time in number of reference cycles over the SEU bit location in different registers in DLF and PD for the synthesized bang-bang ADPLL design. Limit-cycle errors are indicated in the shaded region, where the everlasting limit-cycle errors are indicated with arrows pointing to infinity.

For SEUs in both PD and DLF, Fig. 59 compares modeled values with measurements for the perturbation time in number of reference cycles over the SEU bit location. Overall, calculated values from the proposed model show good agreement with the measurement. The model successfully predicts that SEUs in the most significant two bits in the DLF integral register can induce limit-cycle errors for current design topology. Also, limit-cycle errors resulting from SEUs in bit 9 dies out after hundreds of reference cycles, while limit-cycle errors continue until system reset for SEUs originating in bit 10, which agrees perfectly with the measured data as indicated in the shaded region in Fig. 59. Again, for SEUs in the FD module of the bang-bang ADPLL, the modeled and measured perturbation time both show less than 2 reference cycles.

Implications for RHBD and Limitations of the Model

The proposed time-domain model has many implications for RHBD techniques. Firstly, based on the requirement for converging to 0 in Eqn. 42,

$$|A| \leq 1, \left|\frac{B}{A}\right| \leq 1. \tag{55}$$

Therefore, Eqn. (23),

$$\alpha^2 k_{PD} k_{DCO} \geq 4\rho. \tag{56}$$

is the requirement on the loop parameters for the ADPLL to converge. The closer A is to 0, the faster the loop converges, i.e. the faster the loop locks on to the desired frequency after SEU perturbation.

As stated in previous chapters, an embedded tradeoff exists between the locking time and filtering capability of the DLF. The loop regains lock a lot faster if fewer register bits are implemented for filtering purposes. However, fewer register bits in the filter limits the frequency tuning resolution and the damping of the ADPLL system. The proposed time-domain model clearly identifies the sensitive bits in main registers used in an ADPLL. This provides the designers a time-efficient method to understand the single-event behavior of the ADPLL during the design process without carrying out tedious SEU/SET simulations. Using the model in a statistical design flow in combination with a modeled radiation environment can enable designer to understand the system output error distribution in a complex radiation environment.

In addition, this model provides output perturbation time information associated with each register bit. Depending on the circuit application requirement and usages, hardened flip-flops (FFs) can be used for selected sensitive bits in the registers, for optimized performance. For instance, when output frequency divider with large

frequency divisor (M) is implemented within the ADPLL, loss-of-lock errors over short period of duration (i.e. less than M output clock cycles) can show up as duty-cycle errors in the actual output clock signals. Hence, any register bits that cause loss-of-lock errors with perturbation duration of less than M output clock cycles do not have high priority when implementing hardening solutions.

In this section results are mainly presented for 2nd-order integer-N ADPLLs. However, by setting the proper SEU- induced initial phase errors and taking into consideration of different orders of integrating effect in the system model, the modeling methodology applies for higher-order ADPLLs with cascaded infinite impulse response (IIR) filters in the DLF and also fraction-N ADPLLs with sigma-delta modulators in the FDs. Note that noise from the reference signal and DCO may cause slight errors in the calculations. However, the model provides a simplified design-appropriate analysis with manageable degradation in accuracy. Additionally, these slight errors have minimal impact on the model qualitatively predicting SEU vulnerabilities of different modules in the ADPLL. In addition, the proposed modeling methodology inclusively applies for common types of ADPLLs, while potentially using phase-domain model for SEU predictions is not a good fit for bang-bang ADPLLs due to their non-linearity in the phase domain. Comparing with using existing time-domain models for SEU prediction, the proposed model provides a simplified design-appropriate analysis with manageable degradation in accuracy.

Conclusions

A generic time-domain modeling methodology was proposed in this work to quantify the SEU-induced phase error of ADPLLs in the locked state. The proposed

model ties the SEU response of ADPLLs to the SEU-induced initial phase error and loop parameters, and was verified for accurate predictions of system stability and settling speed by FPGA fault injection experiment and TPA laser experiment. The model provides designers with recommendations for key loop parameters for improved reliability for general purpose ADPLL systems. The model also identifies the most sensitive flip-flops in the system for selective hardening to achieve optimized performance.

CHAPTER VII

RHBD DESIGN CONSIDERATIONS FOR ADPLLS

The previous chapters demonstrated the single-event-induced error signature in ADPLLs - temporary-frequency errors, loss-of-lock errors and limit-cycle errors - through circuit simulation and hardware results. In order to develop a radiation hardened by design (RHBD) ADPLL, these errors must be mitigated or reduced. Since some of them are more severe than others, the hardening effort will be primarily directed toward the worst-case SEE response, i.e. the loss-of-lock errors and limit-cycle errors. In this chapter, ready-to-implement design techniques are extracted from the model and previous characterization results ,and provided for RHBD ADPLL designers. Since the DCO of an ADPLL is prone to SET-induced errors, hardening approaches for DCROS are firstly presented in Section VII.1. SEU hardening guidelines for PD, DLF and FD are detailed in Section VIII.2. Section VIII.3 provides designers designing an ADPLL from a CPPLL background with in sight information on the similarities and differences between the single-event modeling and hardening considerations for ADPLLs and CPPLLs.

Hardening Approach Against Harmonic Errors in DCROs

As previously stated, harmonic oscillation in DCROs can induce loss-of-lock errors in ADPLLs. As DCROs are prone to harmonic errors in the SET pulse width window [129], by eliminating the pulse width window could effectively eliminate the occurrences of harmonic errors. In fact, after closely examining the harmonic

oscillation model, it is found that a 3-stage DCRO inherently satisfy the criterion to eliminate the harmonic window.

Two common methods have been reported in the literature to implement a 3-stage DCRO. One implementation is based on digitally controlling of current starving inverters [147]. The other is based on tri-state inverter banks, as shown in the schematic in Fig. 60[148][149][150]. In the uniformly-sized inverter array, each row contains 3 stages of tri-state inverters. All of the three inverters are controlled by the same bit from the digital control word. The number of rows implemented in the inverter array equals to the number of bits in the digital control word for the DCO. Essentially, since all the tri-state inverters are connected, only three voltage nodes are in this circuits. Therefore, it is a 3-stage DCRO.

In fact, a 6-bit DCRO using the above-mentioned topology based on tri-state inverter array was implemented in UMC40nm technology. The relationship between the output frequency and the control code is shown in Fig. 61. This 3-stage DCRO was simulated for all the possible combination of digital control code. The DCO exhibit excellent frequency tuning linearity over its frequency range of 3.6 GHz to 4.9 GHz. A wider frequency range and even better frequency tuning linearity can be achieved by manually tuning the sizes of the inverters in the inverter array.

Due to the symmetry of circuit design, calibrated ISDE bias dependent SEE current model [92] is used for injecting SET pulses at 1 of the 3 internal nodes of the DCRO over clock period. No SE-induced harmonic oscillation was observed for injected SET current pulses corresponding to 1-60 MeV-cm^2/mg.

Therefore, 3-stage DCROs are suggested for RHBD ADPLL designs to avoid harmonic-oscillation-induced loss-of-lock errors. If 3-stage DCROs are not able

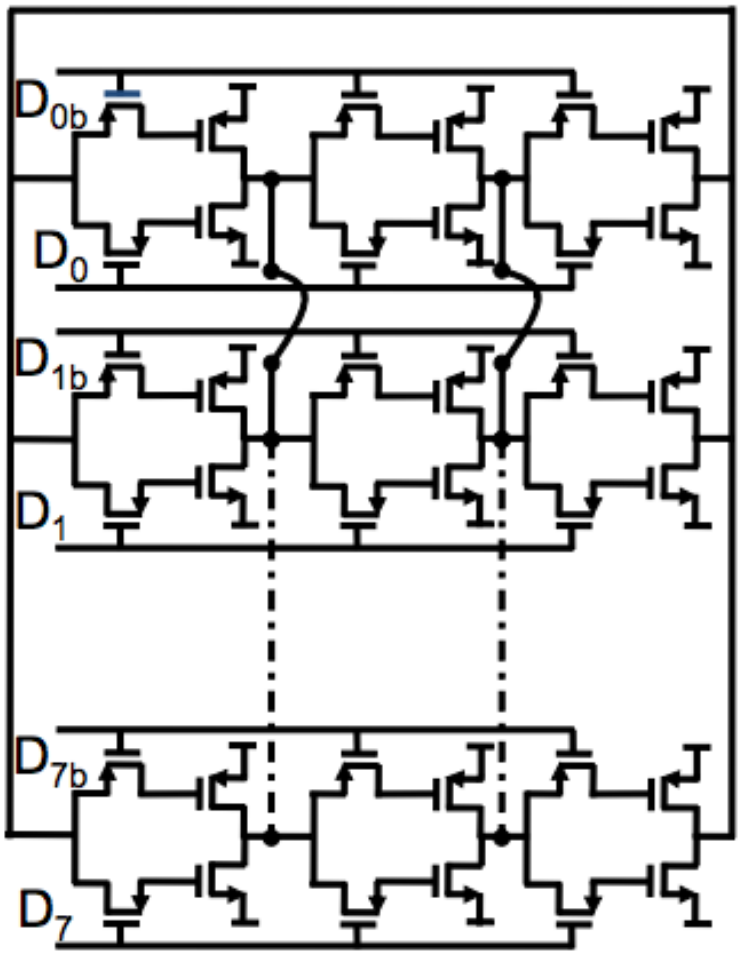


Figure 60: Schematic of harmonic-oscillation proof 3-stage DCRO [148].

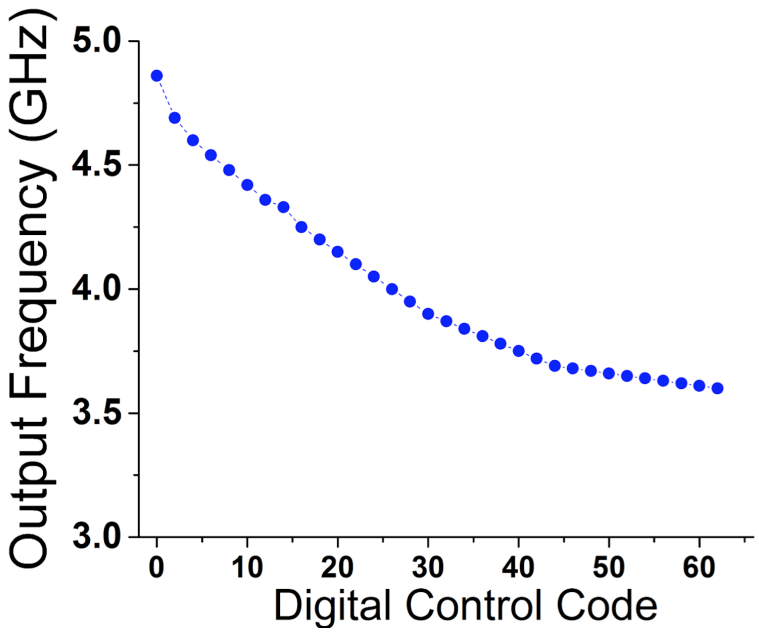


Figure 61: Relationship between output frequency and digital control code showing linearity of the designed DCRO.

to provide the required frequency tuning resolution, DCROs with few stages are suggested over ones with a large number of stages to minimize the occurrance of SE harmonic oscillations.

SEU Tolerant Hardening Approaches

As stated in previous chapter, SEUs are more concerning in digital control blocks such as PD, DLF, FD and global register modules. Therefore, SEU hardening techniques needs to be applied. Based on the SEU characterization results and proposed time-domain model, the SEU tolerant hardening approaches can be summarized as following:

(1) ADPLLs with frequency-linear PD are recommended for applications requiring mainly frequency locking rather than phase alignment between the PLL output signal (such as LO for transceivers in RF applications) due to its operation linearity

and design simplicity comparing with TDCs. However, RHBD FFs need to be implemented for the MSBs for the registers in frequency-linear PDs. Fraction-based frequency-linear PD achieves better operation linearity and wider locking range with more design complexity comparing with integer-based frequency-linear PD. And the former design implementation potentially requires RHBD FFs in more bits than integer-based PDs due to the usage of divider.

(2) ADPLLs with FSM-based PDs are recommended over pure time-based PDs for applications requiring phase alignment (such as clock data recovery systems). Even though ADPLLs with FSM-based PDs switch between frequency-tracking and phase-tracking mode and have higher design complexity comparing to ADPLLs with time-based PDs, it achieves optimum system settling time with minimum added sensitive area to SEEs since only the phase-tracking path of the the ADPLL is active during normal operation. Bang-bang PDs can be used in the phase-tracking mode of this ADPLL implementation due to its low design complexity and low SEU vulnerability. If a TDC is used as the time-based phase detector, most of the register bits in the PD needs to be replaced with SEU-tolerant FFs to guarantee the output robustness. Also a multiplexer needs to be implemented in the 1st-pole integrator of the DLF in the frequency-tracking path to allow for loop tolerance against SEUs/SETs in the FSM controller.

(3) Even though changing the frequency divisor for a frequency-based ADPLL does not affect the loop SE vulnerability, the SE vulnerability of a bang-bang ADPLL increases with increasing of the frequency divisor. In other words, small feedback frequency divisor and large reference clock frequency is recommended for bang-bang ADPLLs for SE tolerance considerations.

(4) For an ADPLL that works over certain frequency range, the fewer number of bits there are in the control word, i.e. the larger the gain of the DCO (K_{DCO}), the fewer number of bits that could result in loss of lock errors at ADPLL output. However, K_{DCO} may not be entirely adjustable for a given application with particular output phase noise requirements.

(5) Single-event tolerance of the 1-bit RESET signal should be achieved by deploying a rad-hard flip-flop to avoid SEU-induced loss-of-lock errors.

(6) For a 1st-order or 2nd-order ADPLL, the DLF module usually consists of a PI filter.

(i) The larger the integral path gain in the PI filter, the fewer bits in the PI filter needs to be hardened against SEUs. However, the PI filter has less filtering capability for incoming perturbation from PD and FD. On the other hand, if more bits are implemented in the integral path of the PI filter, the PI filter can filter out larger word perturbation from FD and PD comparing to when less bits are used. And in this case, RHBD FFs are needed for most significant bits in the register corresponding to the integral path.

(ii)The proportional path gain of the PI filter is determined by the system damping and the integral path gain. MSBs in the register corresponding to the proportional path of the PI filter also needs to be replaced with RHBD FFs to mitigate loss-of-lock errors.

(7) To construct high-order ADPLLs (i.e. ADPLL order higher than 2), DLF of an ADPLL usually consists of a PI filter and a cascaded FIR/IIR filter. The number of tap registers in the FIR/IIR filter should be optimized with respect to noise performance.

By selectively implementing TMR or the proposed hardening strategies to the sensitive modules, a significant improvement in SE susceptibility can be produced without introducing much area and power penalty. In fact, assuming RHBD FF cells are twice in area compared with standard DFFs, the hardening area penalty for the ADPLLs on 180nm, 65nm and 32nm (mentioned in chapter VI) is 30%, 10% and 70%, if the RHBD FF cell is implemented for all the FFs in the designs. By selectively hardening the only FFs which can result in loss-of-lock errors, the area penalty can be reduced by half for each technology node.

Though implementing the hardening techniques can result in certain electrical performance and design tradeoffs such as phase noise and jitter must be considered. As stated previously, since the transfer function of DLF and DCO plays a significant role in the output phase noise of ADPLL, proper attention must be paid to overall performance before applying design guidelines (4) and (7).

Comparisons Between A/MS PLLs and ADPLLs

Despite the apparent digital nature and emerging popularity of ADPLLs, a great majority of PLL applications are based on A/MS PLLs, namely charge-pump PLL structures. To provide conventional A/MS PLL designers with some in sight information on how ADPLLs compare with CPPLLs in radiation performance and the hardening tradeoffs, SE characterization and modeling of the ADPLL were performed to make an apple-to-apple comparison with existing work on SE characterization of CPPLLs and analytical SET propagation model of CPPLLs proposed in [107]. Thorough comparisons in terms of single-event modeling and hardening tradeoffs between A/MS PLLs and ADPLLs are detailed in this section.

Since a CPPLL uses analog voltage for continuous frequency tuning, the output perturbation time, i.e. output phase displacement, is directly related to the originating SET duration and voltage perturbation resulting from a single ionization radiation [151].

In fact, in [151], the equations for the ideal loop recovery time (t_{PD} and t_{CP}) following transients in PD and CP of a CPPLL are as following:

$$t_{PD} = t_{SET}, \tag{57}$$

$$t_{CP} = t_{SET} + \frac{Q_{SET}}{I_{CP}}, \tag{58}$$

where t_{SET} represents the length of the initial transient pulse, Q_{SET} is the amount of charge in/out of the CP sub-circuit as a result of the perturbation and I_{CP} is the nominal CP current.

However, in all of the digital control blocks of ADPLLs, namely the PD, DLF and FD, the SETs need to translate to a digital word perturbation to manifest as an output frequency error. Therefore, unlike SETs in CPPLLs, SEU/SETs in PD or DLF in ADPLLs cause a digital word shift only proportional to the significance of the the perturbed bit, which is independent from the length of the initial transient pulse or the other design parameters of the loop. In other words, the actual pulse width, or collected charge of the SET, does not have a direct relationship with the resulted output perturbation time at the ADPLL output.

A simplified equation from [151] for the voltage perturbation V_e from SETs is shown in Eqn. 59,

$$V_e = \frac{2\pi k_{PD} t_{rec}}{C}, \tag{59}$$

in which t_{rec} is the loop recovery time, k_{PD} is the gain of the phase detector and C is the major capacitor in the loop filter for the CPPLL.

The concept of Eqn. 59 applies well in the scenario of ADPLLs: voltage perturbation V_e is equivalent to the digital control word perturbation resulted from SEUs or SETs in the loop. Therefore, if more bits are implemented for the integrating register, i.e. equivalent to smaller capacitor C in A/MS PLL, the longer it takes for the loop to correct that digital word perturbation, thus resulting in longer loop recovery time t_{rec}. And also, if smaller steps from the phase detector (K_{PD}) is taken every time for frequency/phase tuning, again, the longer time it takes for the loop to correct the digital word perturbation.

However, while a SET in the loop filter of a CPPLL become diminished because of the filtering of passive resistance and capacitors, an SEU/SET-induced digital word perturbation in the DLF of an ADPLL can not only cause a digital control word change at the input of the DCO, it also changes the proportional gain or the integral gain of the DLF. As stated in Chapter II, the relationship between the proportional path gain (α) and integral path gain (ρ) can be related back to the R and C of an analog loop filter using bilinear transform [48]. Therefore, the equivalent R and C in the loop filter is changed. And similar rules apply for SEUs or SETs in the PD, which changes the gain of the PD (k_{PD}) as well.

All of above are difficult to be incorporated in Eqn. 58. Therefore, SEU/SET-induced perturbation is easier to be analyzed with the proposed model in this work.

Based on previous discussion, similarities and differences coexist in RHBD considerations for ADPLLs and CPPLLs. Comparisons between the RHBD techniques and their tradeoffs are presented in this subsection.

(1) DCOs are inherently more immune to SET perturbations comparing with VCOs with the sacrifice of poorer phase noise performance due to digital switching noises.

(2) Using large capacitors in the loop filter for A/MS PLL is a method of making the PLL more SE-tolerant, which trades area/power with radiation performance. In ADPLLs, this is equivalent to using small gain in the integral path, i.e. using large number of bits for the integral register. Even though this means that using larger number of bits for the integral register can help filtering out more incoming word perturbation from prior modules (i.e. PD and FD), the integral path itself is with larger SEU susceptibility.

(3) As in (2), the SETs in CPPLLs can be filtered out using larger capacitors and larger I_{cp} in the loop. However, since the analog functions are implemented using digital circuitries in ADPLLs, the SEUs can corrupt the functions of capacitances and resistances by corrupting the bits in the registers.

(4) In RHBD CPPLLs, decreasing the gain of the VCO, K_{VCO}, decreases the bandwidth of the VCO and increases the SE tolerance of the PLL. However, in ADPLLs, increasing the gain of the DCO, K_{DCO}, decreases the usage of bits for digital control word, thus decreasing the single-event vulnerability of the design.

(5) Most of the hardening techniques for ADPLLs regard the registers in the digital control blocks, which can be easily implemented using RHBD FF cells instead of regular DFFs without sacrificing a lot of design performance. However, the RHBD techniques for CPPLL involves changing a lot of the loop parameters, which adds a great deal of design complexity.

Conclusions

This chapter summarizes some RHBD design guidelines from SE modeling of ADPLLs and SE characterizations of different types of ADPLLs. Suggestions are given to designers based on different requirement from different types of space applications. SE modeling and RHBD hardening techniques are compared between ADPLLs and CPPLLs to provide PLL designer with in depth understanding of RHBD ADPLLs.

CHAPTER VIII

CONCLUSIONS

In this dissertation, we have addressed single-event-induced reliability concerns for all-digital phase-locked loops (ADPLLs) through circuit simulation, experimentation and analysis.

Based on the modular single-event vulnerabilities, SET and SEU characterization was performed on different types of ADPLLs. FPGA-based fault injection was utilized to perform topological analysis and SEU characterization of ADPLLs. SET-induced errors in ADPLLs were investigated through circuit analysis, simulation and experimentation. The major single-event-induced error signatures in ADPLLs were identified to be temporary-frequency errors, loss-of-lock errors, and limit-cycle errors. The digital loop filter (DLF) was identified as the sub-circuit that is most sensitive to single-event irradiation in any ADPLL topology. SEUs in the registers corresponds to the lower-order poles can generate loss-of-lock errors at the ADPLL output and those in the registers corresponds to the higher-order poles can lead to limit-cycle errors. This is the first work that has reported limit-cycle errors for clock circuitries and identifies the vulnerable sub-circuits for ADPLLs towards SEEs.

At the sub-circuit level, different design choices were evaluated for each design module in the ADPLL and hardening approaches (if any) were proposed for the subcircuits inside ADPLLs. Comparison of the contribution of each module to the overall SE performance of closed-loop ADPLL was also evaluated for different types of ADPLLs.

In addition, a novel time-domain model was developed for the very first time for SEU-induced errors in linear and non-linear ADPLLs that quantifies the perturbation due to SEUs originating from different modules in the system while it is in locked state. The model provides designers with full characterization of ADPLLs response to SEUs during the design stage and identifies the most sensitive sub-circuits in the system where hardening techniques should be applied with priority. The model was verified through FPGA-based fault injection experiment and TPA laser tests on ADPLLs using time-to-digital converter (TDC) circuits. The proposed time-domain methodology can also be easily used within a design flow incorporating complex radiation effects.

Finaly, we develop a list of general RHBD design guidelines for different types of ADPLL circuits. These are a set of design rules based on the topological designs of ADPLLs that are independent of technology scaling. Various types of PLLs are compared in electrical and single-event performances and suggested for applications with different design specifications and rad-hardness level requirements. ADPLLs are also compared with A/MS PLLs in various aspect to provide conventional A/MS PLL designers with in sight information on designing an RHBD ADPLL.

REFERENCES

[1] B. Razavi, *Design of Analog CMOS Integrated Circuits*. United States of America: McGraw-Hill, 2001.

[2] M.-F. Lai and M. Nakano, "Special section on phase-locked loop techniques," *IEEE Transactions on Industrial Electronics*, vol. 43, no. 6, pp. 607–608, Dec 1996.

[3] B. Razavi, *Phase-Locking in High-Performance Sytems - From Devices to Architectures*. Hoboken, New Jersey, United States of America: John Wiley & Sons, Inc., 2003.

[4] R. B. Staszewski, *All-digital Frequency Synthesizer in Deep-submicron CMOS*. Hoboken, New Jersey, United States of America: John Wiley & Sons, Inc., 2006.

[5] R. Staszewski, K. Muhammad, D. Leipold, C.-M. Hung, Y.-C. Ho, J. Wallberg, C. Fernando, K. Maggio, R. Staszewski, T. Jung, J. Koh, S. John, I. Y. Deng, V. Sarda, O. Moreira-Tamayo, V. Mayega, R. Katz, O. Friedman, O. Eliezer, E. de Obaldia, and P. Balsara, "All-digital TX frequency synthesizer and discrete-time receiver for bluetooth radio in 130-nm CMOS," *IEEE J. Solid State Circuits*, vol. 39, no. 12, pp. 2278–2291, 2004.

[6] R. Staszewski, J. Wallberg, S. Rezeq, C.-M. Hung, O. Eliezer, S. Vemulapalli, C. Fernando, K. Maggio, R. Staszewski, N. Barton, M.-C. Lee, P. Cruise, M. Entezari, K. Muhammad, and D. Leipold, "All-digital pll and transmitter for mobile phones," *IEEE J. Solid State Circuits*, vol. 40, no. 12, pp. 2469–2482, 2005.

[7] P. Dodd and L. Massengill, "Basic mechanisms and modeling of single-event upset in digital microelectronics," *IEEE Trans. Nucl. Sci.*, vol. 50, no. 3, pp. 583–602, June 2003.

[8] S. P. Buchner and M. P. Baze, "Single-event transients in fast electronic circuits," in *IEEE NSREC Short Course*, vol. 50, 2001.

[9] D. Pan, H. Li, and B. Wilamowski, "A radiation-hard phase-locked loop," in *IEEE ISIE '03*, vol. 2, June 2003, pp. 901–906 vol. 2.

[10] G. Lyons, G. Wu, T. Mellissinos, and J. Cable, "A high performance rad hard 2-3 GHz integer N cmos phase lock loop," in *Radiation Effects Data Workshop, 1999*, 1999, pp. 41–45.

[11] Y. Boulghassoul, L. Massengill, A. Sternberg, B. Bhuva, and W. Holman, "Towards SET mitigation in RF digital PLLs: From error characterization to

radiation hardening considerations," *IEEE Trans. Nucl. Sci.*, vol. 53, no. 4, pp. 2047–2053, Aug 2006.

[12] T. Loveless, L. Massengill, B. Bhuva, W. Holman, R. Reed, D. McMorrow, J. Melinger, and P. Jenkins, "A single-event-hardened phase-locked loop fabricated in 130 nm CMOS," *IEEE Trans. Nucl. Sci.*, vol. 54, no. 6, pp. 2012–2020, Dec 2007.

[13] K. Poltorak, F. Tavernier, and P. Moreira, "A radiation-hard PLL for frequency multiplication with programmable input clock and phase-selectable output signals in 130 nm CMOS," *Journal of Instrumentation*, vol. 7, no. 12, p. C12014, 2012.

[14] D. Van Alen and A. Somani, "An all digital phase locked loop fault tolerant clock," in *Circuits and Systems, 1991., IEEE International Sympoisum on*, Jun 1991, pp. 3170–3173 vol.5.

[15] A. N. Nemmani, "Design techniques for radiation hardened-phase locked loops," Ph.D. dissertation, 2005.

[16] F. M. Gardner, "Charge-pump phase-lock loops," *Communications, IEEE Transactions on*, vol. 28, no. 11, pp. 1849–1858, Nov 1980.

[17] M. S. W. Chen, D. Su, and S. Mehta, "A calibration-free 800 MHz fractional-N digital PLL with embedded TDC," *IEEE Journal of Solid-State Circuits*, vol. 45, no. 12, pp. 2819–2827, Dec 2010.

[18] W. Khalil, S. Shashidharan, T. Copani, S. Chakraborty, S. Kiaei, and B. Bakkaloglu, "A $hbox700-muhboxA$ 405-MHz all-digital fractional- n frequency-locked loop for ISM band applications," *IEEE Transactions on Microwave Theory and Techniques*, vol. 59, no. 5, pp. 1319–1326, May 2011.

[19] W. Deng, D. Yang, T. Ueno, T. Siriburanon, S. Kondo, K. Okada, and A. Matsuzawa, "A fully synthesizable all-digital PLL with interpolative phase coupled oscillator, current-output DAC, and fine-resolution digital varactor using gated edge injection technique," *IEEE Journal of Solid-State Circuits*, vol. 50, no. 1, pp. 68–80, Jan 2015.

[20] G. Marzin, S. Levantino, C. Samori, and A. L. Lacaita, "A 20 Mb/s phase modulator based on a 3.6 GHz digital PLL with 36 dB EVM at 5 mW power," *IEEE Journal of Solid-State Circuits*, vol. 47, no. 12, pp. 2974–2988, Dec 2012.

[21] L. Vercesi, L. Fanori, F. D. Bernardinis, A. Liscidini, and R. Castello, "A dither-less all digital PLL for cellular transmitters," *IEEE Journal of Solid-State Circuits*, vol. 47, no. 8, pp. 1908–1920, Aug 2012.

[22] D. Banerjee, *PLL Performance, Simulation, and Design.* United States of America: Dog Ear Publishing, 2006.

[23] R. E. Best, *Phase-Locked Loops: Design, Simulation, And Applications.* New York, United States of America: McGraw-Hill, 2003.

[24] K. Fukuda, H. Yamashita, G. Ono, R. Nemoto, E. Suzuki, T. Takemoto, F. Yuki, and T. Saito, "A 12.3mw 12.5gb/s complete transceiver in 65nm cmos," in *2010 IEEE International Solid-State Circuits Conference - (ISSCC)*, Feb 2010, pp. 368–369.

[25] S.-H. Wang, J. Kim, J. Lee, H. S. Nam, Y. G. Kim, J. H. Shim, H. K. Ahn, S. Kang, B. H. Jeong, J. H. Ahn, and B. Kim, "A 500-mb/s quadruple data rate sdram interface using a skew cancellation technique," *IEEE Journal of Solid-State Circuits*, vol. 36, no. 4, pp. 648–657, Apr 2001.

[26] S. Y. Yang, W. Z. Chen, and T. Y. Lu, "A 7.1 mW, 10 GHz all digital frequency synthesizer with dynamically reconfigured digital loop filter in 90 nm CMOS technology," *IEEE Journal of Solid-State Circuits*, vol. 45, no. 3, pp. 578–586, March 2010.

[27] C. Albea, D. Puschini, S. Lesecq, E. Beigne, and V. P., "Architecture and control of a digital frequency-locked loop for fine-grain dynamic voltage and frequency scaling in globally asynchronous locally synchronous structures," *J. Low Power Electronics.*, vol. 7, no. 3, pp. 328–340, 2011.

[28] F. M. Gardener, *Phaselocked Techniques.* New York, United States of America: John Wiley & Sons, Inc., 1979.

[29] K.-H. Cheng and Y.-J. Chen, "A novel all digital phase locked loop (ADPLL) with ultra fast locked time and high oscillation frequency," in *Proceedings 14th Annual IEEE International ASIC/SOC Conference (IEEE Cat. No.01TH8558)*, 2001, pp. 139–143.

[30] T.-C. Chao and W. Hwang, "A 1.7mW all digital phase-locked loop with new gain generator and low power DCO," in *2006 IEEE International Symposium on Circuits and Systems*, May 2006, pp. 4 pp.–4870.

[31] C. T. Wu, W. C. Shen, W. Wang, and A. Y. Wu, "A two-cycle lock-in time ADPLL design based on a frequency estimation algorithm," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 57, no. 6, pp. 430–434, June 2010.

[32] J. Zhuang, Q. Du, and T. Kwasniewski, "A 4GHz low complexity ADPLL-based frequency synthesizer in 90nm CMOS," in *Custom Integrated Circuits Conference, 2007. CICC'07. IEEE.* IEEE, 2007, pp. 543–546.

[33] R. J. Baker, *CMOS: circuit design, layout, and simulation.* John Wiley & Sons, 2008, vol. 1.

[34] R. Staszewski, S. Vemulapalli, P. Vallur, J. Wallberg, and P. Balsara, "1.3V 20 ps time-to-digital converter for frequency synthesis in 90-nm CMOS," *Circuits and Systems II: Express Briefs, IEEE Transactions on*, vol. 53, no. 3, pp. 220–224, March 2006.

[35] P. Dudek, S. Szczepanski, and J. V. Hatfield, "A high-resolution CMOS time-to-digital converter utilizing a vernier delay line," *IEEE Journal of Solid-State Circuits*, vol. 35, no. 2, pp. 240–247, Feb 2000.

[36] M. Z. Straayer and M. H. Perrott, "A multi-path gated ring oscillator TDC with first-order noise shaping," *IEEE Journal of Solid-State Circuits*, vol. 44, no. 4, pp. 1089–1098, April 2009.

[37] E. Zianbetov, D. Galayko, F. Anceau, M. Javidan, C. Shan, O. Billoint, A. Korniienko, E. Colinet, G. Scorletti, J. M. Akr, and J. Juillard, "Distributed clock generator for synchronous soc using adpll network," in *Proceedings of the IEEE 2013 Custom Integrated Circuits Conference*, Sept 2013, pp. 1–4.

[38] C. C. Chung and C. Y. Ko, "A fast phase tracking ADPLL for video pixel clock generation in 65 nm CMOS technology," *IEEE Journal of Solid-State Circuits*, vol. 46, no. 10, pp. 2300–2311, Oct 2011.

[39] J.-S. Chiang and K.-Y. Chen, "The design of an all-digital phase-locked loop with small DCO hardware and fast phase lock," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 46, no. 7, pp. 945–950, Jul 1999.

[40] V. Kratyuk, P. K. Hanumolu, U. k. Moon, and K. Mayaram, "Frequency detector for fast frequency lock of digital PLLs," *Electronics Letters*, vol. 43, no. 1, pp. 13–14, Jan 2007.

[41] H. Brugel and P. F. Driessen, "Variable bandwidth DPLL bit synchronizer with rapid acquisition implemented as a finite state machine," *IEEE Transactions on Communications*, vol. 42, no. 9, pp. 2751–2759, Sep 1994.

[42] J. Dunning, G. Garcia, J. Lundberg, and E. Nuckolls, "An all-digital phase-locked loop with 50-cycle lock time suitable for high-performance microprocessors," *IEEE Journal of Solid-State Circuits*, vol. 30, no. 4, pp. 412–422, Apr 1995.

[43] T.-Y. Hsu, B.-J. Shieh, and C.-Y. Lee, "An all-digital phase-locked loop (ADPLL)-based clock recovery circuit," *IEEE J. Solid State Circuits*, vol. 34, no. 8, pp. 1063–1073, 1999.

[44] M. Olivieri and A. Trifiletti, "An all-digital clock generator firm-core based on differential fine-tuned delay for reusable microprocessor cores," in *ISCAS 2001. The 2001 IEEE International Symposium on Circuits and Systems (Cat. No.01CH37196)*, vol. 4, May 2001, pp. 638–641 vol. 4.

[45] F. Spagna, "Phase locked loop using delay compensation techniques," in *Proceedings ISCC 2000. Fifth IEEE Symposium on Computers and Communications*, 2000, pp. 417–423.

[46] Finite impulse response. [Online]. Available: https://en.wikipedia.org/wiki/Finite_impulse_response

[47] Infinite impulse response. [Online]. Available: https://en.wikipedia.org/wiki/Infinite_impulse_response

[48] V. Kratyuk, P. Hanumolu, U.-K. Moon, and K. Mayaram, "A design procedure for all-digital phase-locked loops based on a charge-pump phase-locked-loop analogy," *Circuits and Systems II: Express Briefs, IEEE Transactions on*, vol. 54, no. 3, pp. 247–251, March 2007.

[49] R. Staszewski, C.-M. Hung, N. Barton, M.-C. Lee, and D. Leipold, "A digitally controlled oscillator in a 90 nm digital CMOS process for mobile phones," *IEEE J. Solid State Circuits*, vol. 40, no. 11, pp. 2203–2211, 2005.

[50] J. Zhao and Y.-B. Kim, "A low-power digitally controlled oscillator for all digital phase-locked loops," *VLSI Design*, vol. 2010, pp. 1–11, 2010.

[51] S. Levantino, L. Romanò, S. Pellerano, C. Samori, and A. L. Lacaita, "Phase noise in digital frequency dividers," *IEEE J. Solid State Circuits*, vol. 39, no. 5, pp. 775–784, 2004.

[52] R. B. Staszewski and P. T. Balsara, "Phase-domain all-digital phase-locked loop," *Circuits and Systems II: Express Briefs, IEEE Transactions on*, vol. 52, no. 3, pp. 159–163, 2005.

[53] J. Shen, F. Jonsson, J. Chen, H. Tenhunen, and L. Zheng, "Phase noise improvement and noise modeling of type-I ADPLL with non-linear quantization effects," in *NORCHIP, 2014*, Oct 2014, pp. 1–4.

[54] M. Chan and A. Postula, "Transient analysis of bang-bang phase locked loops," *IET Circ. Dev. Sys.*, vol. 3, no. 2, pp. 76–82, April 2009.

[55] N. Da Dalt, "A design-oriented study of the nonlinear dynamics of digital bang-bang PLLs," *IEEE Trans. Circ. and Sys. I: Regular Papers,*, vol. 52, no. 1, pp. 21–31, Jan 2005.

[56] B. Jiang and T. Xia, "ADPLL design parameters determinations through noise modeling," *Integration, the VLSI Journal*, vol. 48, pp. 138–145, 2015.

[57] R. B. Staszewski, C.-M. Hung, K. Maggio, J. Wallberg, D. Leipold, and P. T. Balsara, "All-digital phase-domain tx frequency synthesizer for bluetooth radios in 0.13μm CMOS," in *Solid-State Circuits Conference, 2004. Digest of Technical Papers. ISSCC. 2004 IEEE International.* IEEE, 2004, pp. 272–527.

[58] N. D. Dalt, "Markov chains-based derivation of the phase detector gain in bang-bang PLLs," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 53, no. 11, pp. 1195–1199, Nov 2006.

[59] S. Mendel and C. Vogel, "A z-domain model and analysis of phase-domain all-digital phase-locked loops," in *Norchip, 2007.* IEEE, 2007, pp. 1–6.

[60] J. Shen, F. Jonsson, J. Chen, H. Tenhunen, and L. Zheng, "Phase noise improvement and noise modeling of type-I ADPLL with non-linear quantization effects," in *NORCHIP, 2014.* IEEE, 2014, pp. 1–4.

[61] I. Syllaios, R. Staszewski, and P. Balsara, "Time-domain modeling of an RF all-digital PLL," *Circuits and Systems II: Express Briefs, IEEE Transactions on*, vol. 55, no. 6, pp. 601–605, June 2008.

[62] C. Joubert, J. F. Bercher, G. Baudoin, T. Diverl, S. Ramet, and P. Level, "Time behavioral model for phase-domain ADPLL based frequency synthesizer," in *2006 IEEE Radio and Wireless Symposium*, Jan 2006, pp. 167–170.

[63] Q. Wu, S. Elabd, J. J. McCue, and W. Khalil, "Analytical and experimental study of tuning range limitation in mm-wave cmos lc-vcos," in *2013 IEEE International Symposium on Circuits and Systems (ISCAS2013)*, May 2013, pp. 2468–2471.

[64] S. Sidiropoulos, D. Liu, J. Kim, G. Wei, and M. Horowitz, "Adaptive bandwidth dlls and plls using regulated supply cmos buffers," in *2000 Symposium on VLSI Circuits. Digest of Technical Papers (Cat. No.00CH37103)*, June 2000, pp. 124–127.

[65] R. B. Staszewski, K. Waheed, F. Dulger, and O. E. Eliezer, "Spur-free multirate all-digital PLL for mobile phones in 65 nm CMOS," *IEEE Journal of Solid-State Circuits*, vol. 46, no. 12, pp. 2904–2919, Dec 2011.

[66] Q. Zhang, K. Huang, Z. Liu, and Z. Li, "Research and application of all digital phase-locked loop," in *Intelligent Networks and Intelligent Systems, 2009. ICINIS '09. Second International Conference on*, Nov 2009, pp. 122–125.

[67] J. P. M. Brito and S. Bampi, "Design of a digital fm demodulator based on a 2nd-order all-digital phase-locked loop," *Analog Integrated Circuits and Signal Processing*, vol. 57, no. 1-2, pp. 97–105, 2008.

[68] R. E. Best, *Phase locked loops.* McGraw-Hill Professional, 2007.

[69] S. Chang-hong, Z. ze Chen, and J. Jin-xiong, "An all digital phase-locked loop system with high performance on wideband frequency tracking," in *Hybrid Intelligent Systems, 2009. HIS '09. Ninth International Conference on*, vol. 3, Aug 2009, pp. 460–463.

[70] C.-C. Chung and C.-Y. Lee, "An all-digital phase-locked loop for high-speed clock generation," *IEEE J. Solid State Circuits*, vol. 38, no. 2, pp. 347–351, 2003.

[71] G.-N. Sung, S.-C. Liao, J.-M. Huang, Y.-C. Lu, and C.-C. Wang, "All-digital frequency synthesizer using a flying adder," *Circuits and Systems II: Express Briefs, IEEE Transactions on*, vol. 57, no. 8, pp. 597–601, 2010.

[72] Y. Shayan and T. Le-Ngoc, "All digital phase-locked loop: concepts, design and applications," in *IEE Proceedings F (Radar and Signal Processing)*, vol. 136, no. 1. IET, 1989, pp. 53–56.

[73] D. Sheng, C.-C. Chung, and C.-Y. Lee, "An all-digital phase-locked loop with high-resolution for SoC applications," in *VLSI Design, Automation and Test, 2006 International Symposium on*. IEEE, 2006, pp. 1–4.

[74] A. Neyer, J. H. Mueller, S. Kaehlert, R. Wunderlich, and S. Heinen, "A fully integrated all-digital PLL based fm-radio transmitter in 90 nm CMOS," in *NEWCAS Conference (NEWCAS), 2010 8th IEEE International*. IEEE, 2010, pp. 225–228.

[75] N. A. Mollen, "All-digital phase-locked loop used in a clock recovery algorithm," 1999.

[76] K. Woo, Y. Liu, E. Nam, and D. Ham, "Fast-lock hybrid PLL combining fractional- N and integer-N modes of differing bandwidths," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 2, pp. 379–389, Feb 2008.

[77] W. Wu, X. Bai, R. Staszewski, and J. Long, "A 56.4-to-63.4GHz spurious-free all-digital fractional-N PLL in 65nm CMOS," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2013 IEEE International*, Feb 2013, pp. 352–353.

[78] S. Shahramian, A. Hart, A. C. Carusone, P. Garcia, P. Chevalier, and S. P. Voinigescu, "A D-band PLL covering the 81-82 GHz, 86-92 GHz and 162-164 GHz bands," in *2010 IEEE Radio Frequency Integrated Circuits Symposium*, May 2010, pp. 53–56.

[79] Y. Ho, Y. S. Yang, C. Chang, and C. Su, "A near-threshold 480 MHz 78 μW all-digital PLL with a bootstrapped DCO," *IEEE Journal of Solid-State Circuits*, vol. 48, no. 11, pp. 2805–2814, Nov 2013.

[80] S. a. Yu and P. Kinget, "A 0.65V 2.5GHz fractional-N frequency synthesizer in 90nm CMOS," in *2007 IEEE International Solid-State Circuits Conference. Digest of Technical Papers*, Feb 2007, pp. 304–604.

[81] T. H. Tsai, M. S. Yuan, C. H. Chang, C. C. Liao, C. C. Li, and R. B. Staszewski, "14.5 a 1.22ps integrated-jitter 0.25-to-4ghz fractional-n adpll in 16nm finfet

cm0s," in *2015 IEEE International Solid-State Circuits Conference - (ISSCC) Digest of Technical Papers*, Feb 2015, pp. 1–3.

[82] Z. Cao, Y. Li, and S. Yan, "A 0.4 ps-rms-jitter 1-3 GHz ring-oscillator PLL using phase-noise preamplification," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 9, pp. 2079–2089, Sept 2008.

[83] B. Shen, G. Unruh, M. Lugthart, C. H. Lee, and M. Chambers, "An 8.5 mW, 0.07 mm^2 ADPLL in 28 nm CMOS with sub-ps resolution TDC and 230 fs RMS jitter," in *2013 Symposium on VLSI Circuits*, June 2013, pp. C192–C193.

[84] X. Gao, E. A. M. Klumperink, M. Bohsali, and B. Nauta, "A low noise sub-sampling PLL in which divider noise is eliminated and PD/CP noise is not multiplied by n^2," *IEEE Journal of Solid-State Circuits*, vol. 44, no. 12, pp. 3253–3263, Dec 2009.

[85] A. Rylyakov, J. Tierno, G. English, M. Sperling, and D. Friedman, "A wide tuning range (1 GHz-to-15 GHz) fractional-N all-digital PLL in 45nm SOI," in *2008 IEEE Custom Integrated Circuits Conference*, Sept 2008, pp. 431–434.

[86] O. Richard, A. Siligaris, F. Badets, C. Dehos, C. Dufis, P. Busson, P. Vincent, D. Belot, and P. Urard, "A 17.5-to-20.94GHz and 35-to-41.88GHz PLL in 65nm CMOS for wireless HD applications," in *2010 IEEE International Solid-State Circuits Conference - (ISSCC)*, Feb 2010, pp. 252–253.

[87] E. Temporiti, C. Weltin-Wu, D. Baldi, M. Cusmai, and F. Svelto, "A 3.5 GHz wideband ADPLL with fractional spur suppression through TDC dithering and feedforward compensation," *IEEE Journal of Solid-State Circuits*, vol. 45, no. 12, pp. 2723–2736, Dec 2010.

[88] S. Levantino, G. Marzin, C. Samori, and A. L. Lacaita, "A wideband fractional-N PLL with suppressed charge-pump noise and automatic loop filter calibration," *IEEE Journal of Solid-State Circuits*, vol. 48, no. 10, pp. 2419–2429, Oct 2013.

[89] J. Lee and B. Kim, "A low-noise fast-lock phase-locked loop with adaptive bandwidth control," *IEEE Journal of Solid-State Circuits*, vol. 35, no. 8, pp. 1137–1145, Aug 2000.

[90] L. W. Massengill, "SEU modeling and prediction techniques," *IEEE NSREC Short Course*, pp. III–1III–93, 1993.

[91] N. Atkinson, "System-level radiation hardening of low-voltage analog/mixed-signal circuits," Ph.D. dissertation, Vanderbilt University, 2013.

[92] J. Kauppila, A. Sternberg, M. Alles, A. Francis, J. Holmes, O. Amusan, and L. Massengill, "A bias-dependent single-event compact model implemented into BSIM4 and a 90 nm CMOS process design kit," *IEEE Trans. Nucl. Sci.*, vol. 56, no. 6, pp. 3152–3157, 2009.

[93] S. DasGupta, A. F. Witulski, B. L. Bhuva, M. L. Alles, R. A. Reed, O. A. Amusan, J. R. Ahlbin, R. D. Schrimpf, and L. W. Massengill, "Effect of well and substrate potential modulation on single event pulse shape in deep submicron CMOS," *IEEE Transactions on Nuclear Science*, vol. 54, no. 6, pp. 2407–2412, Dec 2007.

[94] R. Koga, S. D. Pinkerton, S. C. Moss, D. C. Mayer, S. LaLumondiere, S. J. Hansel, K. B. Crawford, and W. R. Crain, "Observation of single event upsets in analog microcircuits," *IEEE Transactions on Nuclear Science*, vol. 40, no. 6, pp. 1838–1844, Dec 1993.

[95] A. L. Sternberg, L. W. Massengill, R. D. Schrimpf, Y. Boulghassoul, H. J. Barnaby, S. Buchner, R. L. Pease, and J. W. Howard, "Effect of amplifier parameters on single-event transients in an inverting operational amplifier," in *RADECS 2001. 2001 6th European Conference on Radiation and Its Effects on Components and Systems (Cat. No.01TH8605)*, Sept 2001, pp. 398–404.

[96] L. Najafizadeh, S. D. Phillips, K. A. Moen, R. M. Diestelhorst, M. Bellini, P. K. Saha, J. D. Cressler, G. Vizkelethy, M. Turowski, A. Raman, and P. W. Marshall, "Single event transient response of SiGe voltage references and its impact on the performance of analog and mixed-signal circuits," *IEEE Transactions on Nuclear Science*, vol. 56, no. 6, pp. 3469–3476, Dec 2009.

[97] H. Cha and J. H. Patel, "A logic-level model for alpha;-particle hits in CMOS circuits," in *Computer Design: VLSI in Computers and Processors, 1993. ICCD '93. Proceedings., 1993 IEEE International Conference on*, Oct 1993, pp. 538–542.

[98] M. P. Baze and S. P. Buchner, "Attenuation of single event induced pulses in CMOS combinational logic," *IEEE Transactions on Nuclear Science*, vol. 44, no. 6, pp. 2217–2223, Dec 1997.

[99] N. Seifert, X. Zhu, D. Moyer, R. Mueller, R. Hokinson, N. Leland, M. Shade, and L. Massengill, "Frequency dependence of soft error rates for sub-micron CMOS technologies," in *International Electron Devices Meeting. Technical Digest (Cat. No.01CH37224)*, Dec 2001, pp. 14.4.1–14.4.4.

[100] P. Liden, P. Dahlgren, R. Johansson, and J. Karlsson, "On latching probability of particle induced transients in combinational networks," in *Proceedings of IEEE 24th International Symposium on Fault- Tolerant Computing*, June 1994, pp. 340–349.

[101] K. W. Li, J. R. Armstrong, and J. G. Tront, "An HDL simulation of the effects of single event upsets on microprocessor program flow," *IEEE Transactions on Nuclear Science*, vol. 31, no. 6, pp. 1139–1144, Dec 1984.

[102] S. Buchner, K. Kang, D. Krening, G. Lannan, and R. Schneiderwind, "Dependence of the seu window of vulnerability of a logic circuit on magnitude of

deposited charge," *IEEE Transactions on Nuclear Science*, vol. 40, no. 6, pp. 1853–1857, Dec 1993.

[103] D. G. Mavis and P. H. Eaton, "Soft error rate mitigation techniques for modern microcircuits," in *2002 IEEE International Reliability Physics Symposium. Proceedings. 40th Annual (Cat. No.02CH37320)*, 2002, pp. 216–225.

[104] N. N. Mahatme, N. J. Gaspard, S. Jagannathan, T. D. Loveless, B. L. Bhuva, W. H. Robinson, L. W. Massengill, S. J. Wen, and R. Wong, "Impact of supply voltage and frequency on the soft error rate of logic circuits," *IEEE Transactions on Nuclear Science*, vol. 60, no. 6, pp. 4200–4206, Dec 2013.

[105] T. Loveless, L. Massengill, B. Bhuva, W. Holman, A. Witulski, and Y. Boulghassoul, "A hardened-by-design technique for RF digital phase-locked loops," *IEEE Trans. Nucl. Sci.*, vol. 53, no. 6, pp. 3432–3438, 2006.

[106] T. Loveless, L. Massengill, W. Holman, and B. Bhuva, "Modeling and mitigating single-event transients in voltage-controlled oscillators," *IEEE Trans. Nucl. Sci.*, vol. 54, no. 6, pp. 2561–2567, 2007.

[107] T. D. Loveless, L. W. Massengill, W. T. Holman, B. L. Bhuva, D. McMorrow, and J. H. Warner, "A generalized linear model for single event transient propagation in phase-locked loops," *IEEE Trans. Nucl. Sci.*, vol. 57, no. 5, pp. 2933–2947, 2010.

[108] R. Kumar, V. Karkala, R. Garg, T. Jindal, and S. P. Khatri, "A radiation tolerant phase locked loop design for digital electronics," in *Computer Design, 2009. ICCD 2009. IEEE International Conference on.* IEEE, 2009, pp. 505–510.

[109] T. Wang, K. Wang, L. Chen, A. Dinh, B. Bhuva, and R. Shuler, "A RHBD LC-tank oscillator design tolerant to single-event transients," *IEEE Trans. Nucl. Sci.*, vol. 57, no. 6, pp. 3620–3625, 2010.

[110] Z. Zhenyu, Z. Minxuan, C. Shuming, C. Jihua, and L. Junfeng, "A radiation-hardened-by-design technique for improving single-event transient tolerance of charge pumps in PLLs," *Journal of Semiconductors*, vol. 30, no. 12, p. 125009, 2009.

[111] H. H. Chung, W. Chen, B. Bakkaloglu, H. Barnaby, B. Vermeire, and S. Kiaei, "Analysis of single events effects on monolithic PLL frequency synthesizers," *IEEE Trans. Nucl. Sci.*, vol. 53, no. 6, pp. 3539–3543, Dec 2006.

[112] Z. Zhenyu, L. Junfeng, Z. Minxuan, and L. Shaoqing, "Modeling and analysis of single-event transients in charge pumps," *Journal of Semiconductors*, vol. 30, no. 5, p. 055006, 2009.

[113] A. V. Rylyakov, J. A. Tierno, D. Z. Turker, J. O. Plouchart, H. A. Ainspan, and D. Friedman, "A modular all-digital PLL architecture enabling both 1-to-2GHz and 24-to-32GHz operation in 65nm CMOS," in *2008 IEEE International Solid-State Circuits Conference - Digest of Technical Papers*, Feb 2008, pp. 516–632.

[114] J. A. Tierno, A. V. Rylyakov, and D. J. Friedman, "A wide power supply range, wide tuning range, all static CMOS all digital PLL in 65 nm SOI," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 1, pp. 42–51, Jan 2008.

[115] Y. Chen, L. Massengill, A. Sternberg, E. Zhang, J. Kauppila, M. Yao, A. Amort, B. Bhuva, W. Holman, and T. Loveless, "Single-event characterization of 1st and 2nd-order linear all-digital phase-locked loops (ADPLLs)," in *IEEE Radiation Effects on Components & Systems*. IEEE, 2016.

[116] Y. Chen, L. Massengill, B. Bhuva, W. Holman, T. Loveless, W. Robinson, N. Gaspard, and A. Witulski, "Single-event characterization of bang-bang all-digital phase-locked loops (ADPLLs)," *IEEE Trans. Nucl. Sci.*, vol. 62, no. 6, pp. 2650–2656, 2015.

[117] Y. P. Chen, L. W. Massengill, J. S. Kauppila, B. L. Bhuva, W. T. Holman, and T. D. Loveless, "Single-event upset characterization of common first- and second-order all-digital phase-locked loops," *IEEE Transactions on Nuclear Science*, vol. 64, no. 8, pp. 2144–2151, Aug 2017.

[118] G. Nicolis and I. Prigogine, *Self-organization in Nonequilibrium Systems*. John Wiley & Sons, 1977, vol. 19.

[119] S. Buchner, M. Baze, D. Brown, D. McMarrow, and J. Melinger, "Comparison of error rates in combinatorial logic and sequential logic," *IEEE Trans. Nucl. Sci*, vol. 35, no. 6, pp. 1517–1522, 1988.

[120] D. G. Mavis and P. H. Eaton, "Soft error rate mitigation techniques for modern microcircuits," in *IEEE international reliability physics symposium*, 2002, pp. 216–225.

[121] M. J. Gadlage, R. D. Schrimpf, J. M. Benedetto, P. H. Eaton, D. G. Mavis, M. Sibley, K. Avery, and T. L. Turflinger, "Single event transient pulse widths in digital microcircuits," *IEEE Trans. Nucl. Sci.*, vol. 51, no. 6, pp. 3285–3290, 2004.

[122] N. Seifert, X. Zhu, D. Moyer, R. Mueller, R. Hokinson, N. Leland, M. Shade, and L. Massengill, "Frequency dependence of soft error rates for sub-micron CMOS technologies," in *Electron Devices Meeting, 2001. IEDM'01. Technical Digest. International*. IEEE, 2001, pp. 14–4.

[123] P. Maillard, W. Holman, T. Loveless, B. Bhuva, and L. Massengill, "An RHBD technique to mitigate missing pulses in delay locked loops," *IEEE Trans. Nucl. Sci.*, vol. 57, no. 6, pp. 3634–3639, 2010.

[124] D. McMorrow, W. Lotshaw, J. Melinger, S. Buchner, and R. Pease, "Subbandgap laser-induced single event effects: carrier generation via two-photon absorption," *IEEE Trans. Nucl. Sci.*, vol. 49, no. 6, pp. 3002–3008, Dec 2002.

[125] D. McMorrow, W. Lotshaw, J. Melinger, S. Buchner, Y. Boulghassoul, L. Massengill, and R. Pease, "Three-dimensional mapping of single-event effects using two photon absorption," *IEEE Trans. Nucl. Sci.*, vol. 50, no. 6, pp. 2199–2207, Dec 2003.

[126] N. Hooten, W. Bennett, L. Edmonds, J. Kozub, R. Reed, R. Schrimpf, and R. Weller, "The impact of depletion region potential modulation on ion-induced current transient response," *IEEE Trans. Nucl. Sci.*, vol. 60, no. 6, pp. 4150–4158, Dec 2013.

[127] A. Balasubramanian, D. McMorrow, S. Nation, B. Bhuva, R. Reed, L. Massengill, T. Loveless, O. Amusan, J. Black, J. Melinger, M. Baze, V. Ferlet-Cavrois, M. Gaillardin, and J. Schwank, "Pulsed laser single-event effects in highly scaled CMOS technologies in the presence of dense metal coverage," *IEEE Trans. Nucl. Sci.*, vol. 55, no. 6, pp. 3401–3406, Dec 2008.

[128] V. Ferlet-Cavrois, P. Paillet, D. McMorrow, N. Fel, J. Baggio, S. Girard, O. Duhamel, J. Melinger, M. Gaillardin, J. Schwank, P. Dodd, M. Shaneyfelt, and J. Felix, "New insights into single event transient propagation in chains of inverters evidence for propagation-induced pulse broadening," *IEEE Trans. Nucl. Sci.*, vol. 54, no. 6, pp. 2338–2346, Dec 2007.

[129] Y. P. Chen, T. Loveless, P. Maillard, N. Gaspard, S. Jagannathan, A. Sternberg, E. Zhang, A. Witulski, B. Bhuva, T. Holman *et al.*, "Single-event transient induced harmonic errors in digitally controlled ring oscillators," *IEEE Trans. Nucl. Sci.*, vol. 61, no. 6, pp. 3163–3170, 2014.

[130] C. Shuming, L. Bin, L. Biwei, and L. Zheng, "Temperature dependence of digital SET pulse width in bulk and SOI technologies," *IEEE Transactions on Nuclear Science*, vol. 55, no. 6, pp. 2914–2920, Dec 2008.

[131] J. A. Maharrey, R. C. Quinn, T. D. Loveless, J. S. Kauppila, S. Jagannathan, N. M. Atkinson, N. J. Gaspard, E. X. Zhang, M. L. Alles, B. L. Bhuva, W. T. Holman, and L. W. Massengill, "Effect of device variants in 32 nm and 45 nm SOI on SET pulse distributions," *IEEE Transactions on Nuclear Science*, vol. 60, no. 6, pp. 4399–4404, Dec 2013.

[132] M. J. Gadlage, J. R. Ahlbin, B. Narasimham, B. L. Bhuva, L. W. Massengill, R. A. Reed, R. D. Schrimpf, and G. Vizkelethy, "Scaling trends in SET pulse widths in sub-100 nm bulk CMOS processes," *IEEE Transactions on Nuclear Science*, vol. 57, no. 6, pp. 3336–3341, Dec 2010.

[133] R. Quinn, J. Kauppila, K. Warren, Y. Chen, B. Bhuva, M. Bounasser, K. Lilja, and L. Massengill, "Probability of latching an SET in advanced technologies," in *IEEE Radiation Effects on Components & Systems*. IEEE, 2016.

[134] S. S. Mukherjee, C. Weaver, J. Emer, S. K. Reinhardt, and T. Austin, "A systematic methodology to compute the architectural vulnerability factors for a high-performance microprocessor," in *Proceedings. 36th Annual IEEE/ACM International Symposium on Microarchitecture, 2003. MICRO-36.*, Dec 2003, pp. 29–40.

[135] H. Quinn, D. Black, W. Robinson, and S. Buchner, "Fault simulation and emulation tools to augment radiation-hardness assurance testing," *IEEE Trans. Nucl. Sci.*, vol. 60, no. 3, pp. 2119–2142, June 2013.

[136] A. Babu, B. Daya, B. Nagasundaram, and N. Veluchamy, "All digital phase locked loop design and implementation," 2009, project report.

[137] O. Ruano, J. A. Maestro, and P. Reviriego, "Performance analysis and improvements for a simulation-based fault injection platform," in *2008 IEEE International Symposium on Industrial Electronics*, June 2008, pp. 2299–2304.

[138] A. M. Amiri, A. Khouas, and M. Boukadoum, "On the timing uncertainty in delay-line-based time measurement applications targeting FPGAs," in *2007 IEEE International Symposium on Circuits and Systems*, May 2007, pp. 3772–3775.

[139] J. Song, Q. An, and S. Liu, "A high-resolution time-to-digital converter implemented in field-programmable-gate-arrays," *IEEE Transactions on Nuclear Science*, vol. 53, no. 1, pp. 236–241, Feb 2006.

[140] J. Wu and Z. Shi, "The 10-ps wave union TDC: Improving FPGA TDC resolution beyond its cell delay," in *2008 IEEE Nuclear Science Symposium Conference Record*, Oct 2008, pp. 3440–3446.

[141] B. Razavi, *Monolithic Phase-Locked Loops and Clock Recocvery Circuits - Theory and Design*. Hoboken, New Jersey, United States of America: John Wiley & Sons, Inc., 1996.

[142] C. Hafer, J. Pfeil, D. Bass, A. Jordan, and T. Farris, "Single event transient event frequency prediction model for a next generation PLL," in *2008 IEEE Radiation Effects Data Workshop*, July 2008, pp. 85–89.

[143] T. D. Loveless, B. D. Olson, B. L. Bhuva, W. T. Holman, C. C. Hafer, and L. W. Massengill, "Analysis of single-event transients in integer-n frequency dividers and hardness assurance implications for phase-locked loops," *IEEE Transactions on Nuclear Science*, vol. 56, no. 6, pp. 3489–3498, Dec 2009.

[144] E. Zianbetov, M. Javidan, F. Anceau, D. Galayko, E. Colinet, and J. Juillard, "Design and vhdl modeling of all-digital PLLs," in *NEWCAS Conference (NEWCAS), 2010 8th IEEE International*, June 2010, pp. 293–296.

[145] M. Abdelfattah, M. Ghoneima, Y. I. Ismail, A. Lotfy, M. Abdel-moneum, N. A. Kurd, and G. Taylor, "Modeling the response of bang-bang digital PLLs to phase error perturbations," in *Proceedings of the IEEE 2012 Custom Integrated Circuits Conference*, Sept 2012, pp. 1–4.

[146] S. Buchner, D. McMorrow, A. Sternberg, L. Massengill, R. L. Pease, and M. Maher, "Single-event transient (SET) characterization of an LM119 voltage comparator: an approach to SET model validation using a pulsed laser," *IEEE Transactions on Nuclear Science*, vol. 49, no. 3, pp. 1502–1508, Jun 2002.

[147] R. K. Pokharel, P. Nugroho, A. Anand, K. Kanaya, and K. Yoshida, "Digitally controlled cmos quadrature ring oscillator with improved fom for ghz range all-digital phase-locked loop applications," in *2012 IEEE/MTT-S International Microwave Symposium Digest*, June 2012, pp. 1–3.

[148] A. Tomar, R. Pokharel, O. Nizhnik, H. Kanaya, and K. Yoshida, "Design of 1.1 GHz highly linear digitally-controlled ring oscillator with wide tuning range," in *Radio-Frequency Integration Technology, 2007. RFIT 007. IEEE International Workshop on*, Dec 2007, pp. 82–85.

[149] W. Kim, J. Park, J. Kim, T. Kim, H. Park, and D. Jeong, "A $0.032mm^2$ 3.1mW synthesized pixel clock generator with 30psrms integrated jitter and 10-to-630MHz DCO tuning range," in *2013 IEEE International Solid-State Circuits Conference Digest of Technical Papers*, Feb 2013, pp. 250–251.

[150] B. G. Jeon, Y. Moon, and T. W. Ahn, "A study on 11 MHz-1537 MHz DCO using tri-state inverter for DAB application," in *TENCON 2009 - 2009 IEEE Region 10 Conference*, Jan 2009, pp. 1–5.

[151] T. Loveless, "A generalized single-event analysis and hardening options for mixed-signal phase-locked loop circuits," Ph.D. dissertation, Vanderbilt University, 2009.

Appendix A

TECHNICAL ANACHRONISMS

ANACHRONISM	**DEFINITION**
AMS	Analog Mixed-Signal
CP	Charge Pump
DCO	Digitally-Controlled Oscillator
DLF	Digitally Loop Filter
FIR	Finite Impulse Response
FD	Frequency Divider
FSM	Finite State Machine
IIR	Infinite Impulse Response
PD	Phase Detector
PLL	Phase-Locked Loop
RHBD	Radiation Hardened By Design
SEE	Single-Event Effect
SET	Single-Event Transient
SEU	Single-Event Upset
TDC	Time-to-Digital Converter
TPA	Two-Photon Absorption
VCO	Voltage-Controlled Oscillator

Appendix B

VHDL/VERILOG SOURCE CODE

Phase Detectors

Bang-bang Phase Detectors

```
module P_D
  (
  input wire clkf,clkr,
  input wire rst,
  output wire inc_out, dec_out);
wire reset;
reg inc, dec;

assign ti=1'b1;
assign inc_out = (rst) ? ti  : inc;
assign dec_out = (rst) ? ti : dec;
assign reset= inc_out && dec_out ;

always @ (posedge clkr or posedge reset)
     begin
         if (reset)
    inc<=1'b0;
         else inc<=1'b1;
```

```
     end
always @ ( posedge clkf or posedge reset)
     begin
if (reset)
    dec<=1'b0;
else dec<=1'b1;
     end
endmodule
```

Time-digital Converter (TDC)

```
1.Delay element used in the delay line of TDC
‘timescale 100ps / 10ps
module buffd4(
input wire I,
output wire Z
);
assign $\#$100 Z=I;
endmodule

2. TDC
library ieee;
use ieee.std_logic_1164.all;
entity tdc is
      generic (
      number_of_bits : integer := 64
```

```
        );
        port (
        retimed_clk : in std_logic;
        variable_clk : in std_logic;
        tdc_out : out std_logic_vector (number_of_bits-1 downto 0);
        reset : in std_logic
        );
end entity;

architecture behavior of tdc is
     component buffd4 is port (
     I : in std_logic;
    Z : out std_logic
      );
     end component;
signal buf_inst_out : std_logic_vector (number_of_bits downto 0);

begin
buf_inst_out(0) <= variable_clk;
tdc_loop : for i in 1 to (number_of_bits) generate
begin
buf_inst : buffd4 port map (
I => buf_inst_out(i-1),
Z => buf_inst_out(i)
);
```

```
end generate;

process (reset,retimed_clk)

begin

if reset = '1' then

tdc_out <= (others => '0');

elsif retimed_clk'event and retimed_clk = '1' then

tdc_out <= buf_inst_out(number_of_bits downto 1);

end if;

end process;

end architecture;
```

3. TDC thermometer-binary decoder

```
library ieee;
  use ieee.std_logic_1164.all;
  use ieee.numeric_std.all;

entity TDC_dec is
  generic (
SELQ : integer := 4; -- TDC_Q index used for edge selection
DTDC : integer := 48; -- latched TDC array bus width
WTDC : integer := 6 -- decoded TDC output bus width
);

port (
```

```
  q: in std_logic_vector (DTDC downto 1);
  ckr: in std_logic;
  tdc_rise: out unsigned (WTDC-1 downto 0);
  tdc_skip : out std_logic;
  tdc_hper: out unsigned (WTDC-1 downto 0)
  );
end;

architecture rtl of TDC_dec is
constant SLV_0: std_logic_vector (SELQ downto 1):=(others=>'0');
begin

process (ckr, q)
variable rise: integer range DTDC-1 downto 0;
variable fall: integer range DTDC-1 downto 0;
variable half_period: integer range DTDC-1 downto 0;
variable skip: std_logic;

begin
if ckr='1' then
rise := 0;
for k in 2 to DTDC loop
       if q(k-1)='1' and q(k)='0' then
          rise := k-1;
  exit;
```

```
       end if;
end loop;


fall := 0;
for k in 2 to DTDC loop
if q(k-1)='0' and q(k)='1' then
fall := k-1;
          exit;
  end if;
end loop;
tdc_rise <= to_unsigned(rise, WTDC);
if q(SELQ downto 1) = SLV_0 then
       skip := '1';
else
      skip := '0';
end if;
tdc_skip <= skip;
if rise > fall then
       half_period := rise - fall;
else
       half_period := fall - rise;
end if;
tdc_hper <= to_unsigned(half_period, WTDC);
end if;
end process;
```

```
end;
```

Fraction-based Linear Phase Detector

```
1. Divider

module division(A,B,Res);

//generic size of input and output ports of the division module

     parameter WIDTH = 8;

    //input and output ports.

    input [WIDTH-1:0] A;

    input [WIDTH-1:0] B;

    output [WIDTH-1:0] Res;

    //internal variables

    reg [WIDTH-1:0] Res = 0;

    reg [WIDTH-1:0] a1,b1;

    reg [WIDTH:0] p1;

    integer i;


    always@ (A or B)

    begin

        //initialize the variables.

a1 = A;

        b1 = B;

        p1= 0;

        for(i=0;i < WIDTH;i=i+1)    begin //start the for loop

    p1 = {p1[WIDTH-2:0],a1[WIDTH-1]};
```

```
            a1[WIDTH-1:1] = a1[WIDTH-2:0];
            p1 = p1-b1;
            if(p1[WIDTH-1] == 1)    begin
                a1[0] = 0;
                p1 = p1 + b1;   end
            else
                a1[0] = 1;
        end
        Res = a1;
    end
endmodule
```

2. Fraction-based Linear PD

```
//divide by 8
`include "./division.v"
module counter(
input wire fin,
input wire reset,
input wire flag,
output reg [9:0] counter,
output reg overflow
);

initial counter=0;
```

```
always @ (posedge fin)
begin
if(reset) begin
counter<=0;
overflow<=0;
end else
begin
  if (counter==10'b1111111111)
begin
        counter <= 0;  // reset to 0
overflow<=1'b1;
end
 else begin
counter <= counter+1;  // increment counter
if (flag)
overflow<=1'b0;
end
end
end
endmodule


module P_D
  (
  input wire clk_before_div,clkr,
  input wire rst,
```

```
  input wire [1:0] speed,


  output reg  [10:0] adjust);
  reg [9:0] counter_prev;
  wire [9:0] counter_fast;
  wire [10:0] delta_counter,temp1,temp2;
  wire overflow;
  reg overflow_reg;
  wire flag;
  wire [9:0] Res;
counter fast(clk_before_div,rst,flag,counter_fast,overflow);
division #(11) uut (
        .A(11'b01000000000),
        .B(delta_counter),
        .Res(Res)
    );
initial adjust=11'b0;
initial counter_prev=10'b0;
assign delta_counter=overflow?1024+counter_fast-counter_prev:
counter_fast-counter_prev;
assign flag=overflow_reg && overflow;
always @ (posedge clkr)
begin
if (rst)
begin
```

```
counter_prev<=0;

adjust<=0;

overflow_reg<=0;

end

else begin

overflow_reg<=overflow;

counter_prev<=counter_fast;

adjust<=64-Res;


end

end

endmodule
```

Integer-based Linear Phase Detector

```
//divide by 8

module counter(

input wire fin,

input wire reset,

input wire flag,

output reg [9:0] counter,

output reg overflow

);

initial counter=0;

always @ (posedge fin)

begin
```

```
if(reset) begin

counter<=0;

overflow<=0;

end else

begin

 if (counter==10'b0111111111)

begin

       counter <= 0;  // reset to 0

overflow<=1'b1;

end

 else begin

counter <= counter+1;  // increment counter

if (flag)

overflow<=1'b0;

end

end

end

endmodule


module P_D

  (

  input wire clk_before_div,clkr,

  input wire rst,

  input wire [1:0] speed,
```

```
  output reg  [10:0] adjust);
  reg [9:0] counter_prev,counter_fast;
  wire [10:0] delta_counter,temp1,temp2;
  wire overflow;
  reg overflow_reg;
  wire flag;
counter fast(clk_before_div,rst,flag,counter_fast,overflow);
initial adjust=11'b0;
initial counter_prev=10'b0;
assign delta_counter=overflow?1024+counter_fast-counter_prev:
counter_fast-counter_prev;
assign flag=overflow_reg && overflow;
always @ (posedge clkr)
begin
if (rst)
begin
counter_prev<=0;
adjust<=0;
overflow_reg<=0;
end
else begin
overflow_reg<=overflow;
counter_prev<=counter_fast;
adjust<=delta_counter-8;
```

```
end

end

endmodule
```

Digital Loop Filter

```
1. PI filter

module DLF

  (input wire flag,

   input wire [9:0] word_in,

    input wire clkr_in,

   input wire [10:0] adjust,

 input wire rst,

 output wire [9:0] word_out);

parameter integral_bit=4;

parameter proportional_bit=1;

reg [10+integral_bit:0] word_reg;

wire [10+integral_bit:0] word,word_temp,word_reg_temp;

reg [10+integral_bit:0] adjust_dco_pole;

wire [10+integral_bit:0] adjust_dco,adjust_prev,adjust_wire,

word_overflow_low,word_overflow_high;

reg rst_reg;


assign word_out=word_reg[9+integral_bit:integral_bit];

assign adjust_dco={{(proportional_bit){adjust[10]}},adjust,

{(integral_bit-proportional_bit){1'b0}}}+adjust_prev;
```

```
assign word=(rst)?
{10'b0011000000,{(integral_bit){1'b0}}}:word_reg_temp;

assign adjust_prev=(rst)?0:adjust_dco_pole;
assign adjust_wire=(rst)?0:{{(integral_bit){adjust[10]}},adjust};
assign word_temp=word+adjust_dco;

assign word_overflow_low=(word[10+integral_bit]&&
~adjust_dco[10+integral_bit]&&~word_temp[10+integral_bit])?
{(proportional_bit+10){1'b1}}:word_overflow_high;

assign word_overflow_high=(~word[10+integral_bit]&&
adjust_dco[10+integral_bit]&&
word_temp[10+integral_bit])?0:word_temp;

assign word_reg_temp=
(1'b0)?{word_in,{(integral_bit){1'b0}}}:word_reg;

always @ (posedge clkr_in)
 begin
rst_reg<=rst;
adjust_dco_pole<=adjust_wire+adjust_prev;
word_reg<=word_overflow_low;
end
endmodule
```

2. IIR filter

```
module IIR
  (
  input wire clkr_in,
  input wire [10:0] adjust,
  input wire rst,
  output wire [11:0] iir_out);
parameter integral_bit=1;
reg [10+integral_bit:0] word_reg;
wire [10+integral_bit:0] sum;
assign sum={{(integral_bit){adjust[10]}},adjust}+word_reg;
assign iir_out=(rst)?{(integral_bit+11){1'b0}}:word_reg;

always @ (posedge clkr_in)
 begin
if (rst)
word_reg<={(integral_bit+11){1'b0}};
else
word_reg<=sum-{{(integral_bit){word_reg[10+integral_bit]}},
word_reg[10+integral_bit:integral_bit]};
end
endmodule
```

```
//Multiplexed divide-by-2/4/8/16

module freq_div(

input wire fin,

input wire reset,

output wire fbuf2, fbuf4 ,fbuf8 ,fbuf16

);

reg[3:0] counter;

reg reset_prev;

initial counter=4'b0;

assign fbuf2=counter[0];

assign fbuf4=counter[1];

assign fbuf8=counter[2];

assign fbuf16=counter[3];


always @ (posedge fin)

begin

reset_prev<=reset;

if(~reset_prev&&reset) begin

counter<=0;

end else begin

 if (counter==4'b1111)

        counter <= 4'b0000;  // reset to 0

 else counter <= counter+1;  // increment counter

end
```

```
end

endmodule


module FD (

   inout wire clk_in,

  input reset,

  input wire [1:0] speed,

output wire clk_out,

output wire clk_counter);

reg [4:0] counter;

reg en;

wire clk_choice1,clk_choice2;

wire clk_2,clk_4,clk_8,clk_16;


assign clk_out=(speed==3)?~clk_16:clk_choice1;

assign clk_choice1=(speed==2)?~clk_8:clk_choice2;

assign clk_choice2=(speed==1)?~clk_4:~clk_2;

assign clk_counter=clk_2;


freq_div df(clk_in,reset,clk_2,clk_4,clk_8,clk_16);


endmodule
```

FSM Controller

```
module control(adjust_div,clkr,clkf, flag,rst,inc_reg,dec_reg,
```

```
inc_bangbang,dec_bangbang);

  input wire clkr,clkf, rst, inc_bangbang,dec_bangbang;
  input wire [10:0] adjust_div;
  output reg flag;

  parameter gain=2;

  wire [10:0]  adjust_bangbang,bb_flag;
  wire [9:0]    temp1,temp2,temp3,temp4,temp5,temp6;
  reg  [10:0]  adjust_reg,adjust_reg_1,adjust_reg_2;
  reg  [10:0]  adjust_reg_3,adjust_reg_4;
  output reg  inc_reg,dec_reg;
  reg enable;

assign adjust_bangbang=(dec_reg$\^$inc_reg)?bb_flag:0;
assign bb_flag=(dec_reg)?gain:-gain;
assign temp1[9:0]=adjust_reg[10:1];
assign temp2[9:0]=adjust_div[10:1];
assign temp3[9:0]=adjust_reg_1[10:1];
assign temp4[9:0]=adjust_reg_2[10:1];
assign temp5[9:0]=adjust_reg_3[10:1];
assign temp6[9:0]=adjust_reg_4[10:1];

always @ (posedge clkr)
```

```
 if (rst) begin
flag<=1'b0;
 end
 else if ((enable)&&(temp1==10'b1111111111||temp1==10'b0000000000)
 &&(temp2==10'b1111111111|| temp2==10'b0000000000)&&
 (temp3==10'b1111111111|| temp3==10'b0000000000)&&
 (temp4==10'b1111111111|| temp4==10'b0000000000))&&
 (temp5==10'b1111111111|| temp5==10'b0000000000)&&
 (temp6==10'b1111111111||temp6==10'b0000000000))
 begin
flag<=1'b1;
 end
 else begin
flag<=1'b0;
 end
always @  (posedge  clkf)
if (rst) begin
inc_reg<=1'b0;
end else begin
inc_reg<=inc_bangbang;
   end
always @  (posedge  clkr)
if (rst) begin
dec_reg<=1'b0;
```

```
end else begin

dec_reg<=dec_bangbang;

   end

always @ (posedge clkr)

begin

if (rst)

begin

adjust_reg<=0;

adjust_reg_1<=0;

adjust_reg_2<=0;

adjust_reg_3<=0;

adjust_reg_4<=0;

enable<=0;

end

else begin

adjust_reg<=adjust_div;

adjust_reg_1<=adjust_reg;

adjust_reg_2<=adjust_reg_1;

adjust_reg_3<=adjust_reg_2;

adjust_reg_4<=adjust_reg_3;

if (adjust_reg_4!=0)

begin

enable<=1'b1;

end

end
```

```
end

endmodule
```

Appendix C

TIME-DOMAIN MODEL MATLAB SOURCE CODE

The MATLAB code for modeling the time-domain SEU response of a frequency-based ADPLL to SEUs in the DLF integral register is presented in this section. The code can be modified to adapt to modeling the system time response of either a frequency-based ADPLL or time-based ADPLL towards SEUs in any modules in the ADPLL using the methodology presented in the dissertation.

```
function []=plot_array_pole(n)
i=[1:1:n];
freq_error=[];
cycle_freq_error=[];
alpha=2^(-3);
beta=2^(-9);
delta_time=40*10^(-6);
DW_init=152;
DW_center=512;
PD=1/delta_time*DW_center/DW_init;
C1=alpha*PD*delta_time;
C2=beta*PD*delta_time;


figure;
%%%%N is the SEU bit number in the register
for N=5:8
```

```
    error=-2^N*delta_time;


freq_error(1)=0;
cycle_freq_error(1)=0;


sum_beta=0;
sum_phase=0;
%%initial digital word
for j=2:n
  %  sum=0;
    if (j>4)
              sum_beta=sum_beta+freq_error(j-4);
    end
    %%positive freq error
   % freq_error(j)=freq_error(j-1)-C1*freq_error(j-1)-C2*sum
    if j>4
    %% for control word overflow check
          DW_judge=DW_init-fix((C1*freq_error(j-2)+C2*sum)/delta_time);
          if DW_judge>1023
               DW_init=DW_init-fix((C1*freq_error(j-2)+C2*sum)/delta_time)-2^N-1024;
               freq_error(j)=freq_error(j-1)-(C1*freq_error(j-2)+C2*sum)+error-1024*delt
          elseif DW_judge<0
               DW_init=DW_init-fix((C1*freq_error(j-2)+C2*sum)/delta_time)-2^N+1024;
               freq_error(j)=freq_error(j-1)-(C1*freq_error(j-2)+C2*sum)+error+1024*delt
          else
```

```
            DW_init=DW_init-fix((C1*freq_error(j-2)+C2*sum_beta)/delta_time)-2^N;
            freq_error(j)=freq_error(j-1)-(C1*freq_error(j-3)+C2*sum_beta)+error;
          end


    else
         DW_init=DW_init-2^N;
         freq_error(j)=freq_error(j-1)+error;
    end
    cycle_freq_error(j)=freq_error(j)-freq_error(j-1);
end
plot(i,freq_error);
figure;plot(i,cycle_freq_error);


end
end
```