國 立 交 通 大 學

電信工程學系

碩 士 論 文

具有快速頻率擷取之相位頻率偵測器與低抖動
之頻率合成器設計

Design of Low Jitter Frequency Synthesizers with Fast
Frequency Acquisition Phase-Frequency Detector

研 究 生：施宜興

指導教授：唐震寰　教授

中 華 民 國 九 十 五 年 七 月

具有快速頻率擷取之相位頻率偵測器與低抖動之頻率合成器設計

Design of Low Jitter Frequency Synthesizers with Fast Frequency Acquisition Phase-Frequency Detector

研 究 生：施宜興　　Student：Yi-Shing Shih

指導教授：唐震寰　教授　　Advisor：Jenn-Hwan Tarng

國 立 交 通 大 學

電 信 工 程 學 系 碩 士 班

碩 士 論 文

A Thesis

Submitted to Department of Communication Engineering
College of Electrical Engineering and Computer Science
National Chiao Tung University
in Partial Fulfillment of the Requirements
for the Degree of
Master
in

Communication Engineering

July 2006

Hsinchu, Taiwan, Republic of China

中華民國九十五年七月

具有快速頻率擷取之相位頻率偵測器與低抖動之頻率合成器設計

研究生：施宜興　　指導教授：唐震寰

國立交通大學

電信工程學系 碩士班

摘要

在現代無線通訊世界，鎖相迴路頻率合成器在射頻前端電路中扮演一個很重要的角色。隨著無線通訊標準演進，鎖相迴路的設計涉及許多問題與折衷取捨，如何達到低抖動、快速鎖定時間以及低消耗功率等的迫切要求，愈來愈具有挑戰性。兩個關鍵重要的設計問題–死區(dead-zone)和盲區(blind-zone)均不利於鎖相迴路的效能，分別會造成時序抖動及鎖定速度變慢。特別是，兩者無法同時兼顧，縮減死區卻會增加盲區。

為克服這些問題，本篇論文的研究焦點著重於相位頻率偵測器的設計。找尋且發展一個新的方法來消除死區和盲區。藉此，提出一個新型的相位頻率偵測器。最後，結合我們所提出的相位頻率偵測器，我們利用台灣積體電路 0.18-μm CMOS 製程來實現一個操作在 2.36~2.95-GHz 的整數除頻頻率合成器。模擬結果顯示，從 2.36 GHz 震盪頻率跳頻鎖定至 2.8 GHz 的情況下，本頻率合成器在1.8伏特的電壓供應下所消耗的功率為 25.9mW，其鎖定時間為1.93μs，相較於原始的頻率合成器改善了將近25%。

Design of Low Jitter Frequency Synthesizer with Fast Frequency Acquisition Phase-Frequency Detector

Student：Yi-Shing Shih　　Advisor：Dr. Jenn-Hwan Tarng

Department of Communication Engineering
National Chiao Tung University

Abstract

In the world of modern wireless communication, phase-locked loop (PLL) based frequency synthesizers have played an important role in RF front-ends. As the wireless standards evolve, it presents an increasing challenge to meet the stringent requirements of low jitter or phase noise, fast settling time, and low power in PLL designs, which involve a lot of design issues and trade-offs. Two crucial design issues, dead-zone and blind-zone are detrimental to the performance of PLLs, increasing the timing jitter and slowing the settling speed, respectively. In particular, the decrease of one of them may cause the increase of the other.

To overcome these issues, the research described in this thesis focuses on the design of phase-frequency detectors (PFDs). A new way to eliminate the dead-zone as well as the blind-zone has been founded and developed, whereby a novel and robust fast frequency-acquisition PFD is proposed. A 2.36~2.95-GHz integer-N frequency synthesizer including our proposed PFD is implemented in a standard TSMC 0.18-μm CMOS process. Simulation results reveal that the frequency synthesizer using our

proposed PFD shows a locking time of 1.93μs, which is an improvement of up to 25% over that using a conventional PFD, while consuming 25.9mW at a 1.8V supply in the case of starting at 2.36 GHz and locking at 2.8 GHz. In addition, as compared with other PFD architectures, our proposed PFD manifests itself as a robust design for higher operating frequency, and neither dead-zone nor blind-zone.

誌　　謝

在碩士研究的這三年歲月，首先要感謝的是我的指導教授 唐震寰教授並致上我最誠摯的謝意。感謝老師在專業的通訊領域中，給予我不斷的指導與鼓勵，並賦予了實驗室豐富的研究資源與環境，使得這篇碩士論文能夠順利完成。

其次，要感謝波散射與傳播實驗室的學長們—鄭士杰學長、劉文舜學長、莊博學長在研究上的幫助與意見，讓我獲益良多。感謝實驗室的同學—和穆、孟勳、舜升等在課業及研究上的互相砥礪與切磋，以及生活上的多彩多姿。感謝學弟妹們—逸倫、建傑、懷文、奕慶、豐吉、育正、志瑋、思云、蓓縝等，讓實驗室在嚴肅的研究氣氛中增添了許多歡樂，有了你們，更加豐富了我這三年的研究生生活。另外，也要感謝助理—梁麗君小姐，在生活上的協助和籌劃每次的美食聚餐饗宴。

最後，要感謝的就是我最親愛的家人，由於他們在我求學過程中，一路陪伴著我，給予我最溫馨的關懷與鼓勵，讓我在人生的過程裡得到快樂，更讓我可以專心於研究工作中而毫無後顧之憂。

鑒此，僅以此篇論文獻給所有關心我的每一個人。

施宜興　誌予

九十五年七月

Table of Contents

List of Tables

List of Figures

CHAPTER 1

Introduction

1.1 Background and Problems

Over the past few years, the wireless communications industry has experienced rapid and incredible growth, much of which has been driven by the rapidly-growing wireless market such as mobile telecommunications. But not only has the market for mobile telecomm grown; during the last couple of years, all kinds of previously wired connections between home and office appliances have been going wireless as well. The cellular phone has a calendar function that synchronizes automatically with the desktop calendar through a Bluetooth connection; the laptop computer accesses the Internet for multimedia services through a WLAN or emerging WiMAX connection, etc. So, there introduces a growing demand for wireless communication in today's world.

In wireless communication systems, low cost, low power consumption, and high performance are the critical requirements due to the highly competitive market environment and limitation in battery life. In order to meet a growing demand for wireless communication, it's desirable to implement radio transceivers monolithically with the help of improving large-scale low-cost integration technology. At present, GaAs, silicon bipolar, and BiCMOS technologies constitute a major section of the RF transceivers market because these technologies provide useful features such as high breakdown voltage and high cutoff frequency, etc. However, they are still expensive and low-density integration technologies so as not to satisfy people's desire for low cost, small size, high portability, and good performance. Fortunately, as the deep-submicron

CMOS process evolves, CMOS technology has strong advantages of low cost and high density compared to other available technologies. Moreover, CMOS technology has the high potential to achieve a fully-integrated solution for *system-on-a-chip* (SOC), which realizes the addition of back-end digital function with the RF front-end circuit. So, in order to achieve the ultimate goal of SOC, many efforts have devoted to well design and implement a transceiver in CMOS process, as well as increase the integration level.

In the world of modern wireless communication, phase-locked loop (PLL) based frequency synthesizers have played an important role in RF front-ends. PLL-based frequency synthesizers are used to provide clean, stable, and precise carrier signals for frequency translation in wireless transceivers, as shown in Figure 1.1 which illustrates a generic transceiver architecture. As the wireless standards evolve, it presents an increasing challenge to meet the stringent requirements of low jitter or phase noise, fast settling time, and low power in PLL-based frequency synthesizer designs, which involve a lot of design issues and trade-offs.

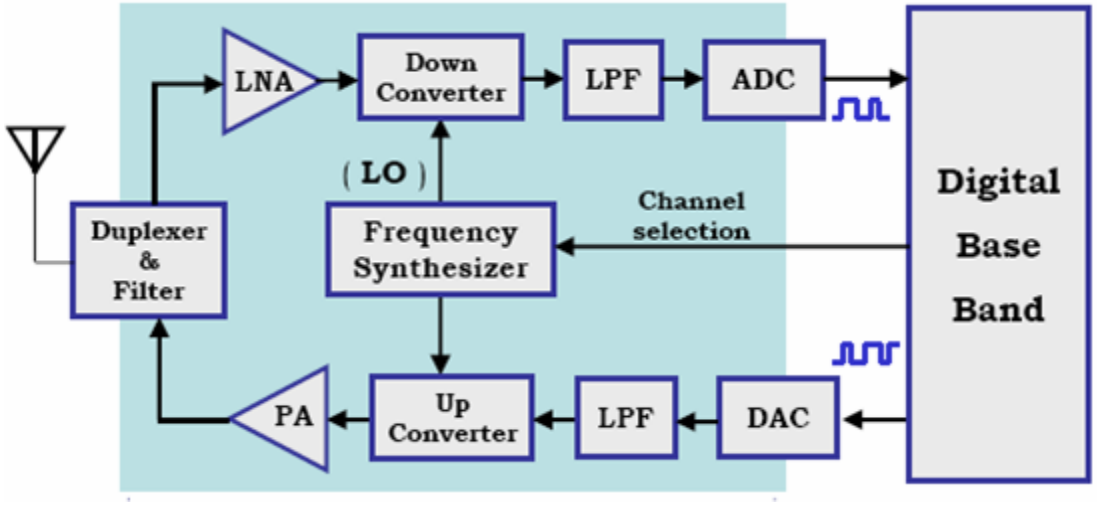


Figure 1.1 Block diagram of a generic transceiver architecture

The important system performance specifications for a frequency synthesizer are timing jitter, locking time, spectral purity, power dissipation, and manufacturing cost, etc. Within different PLL-based topologies, a popular low-cost architecture is the charge-pump PLL-based (CPPLL-based) frequency synthesizers, in which an ideal phase-frequency detector (PFD) is incorporated with an ideal charge pump (CP) to provide an infinite dc gain with passive filters, resulting in an unbounded pull-in range and zero static phase error [1]. However, in reality there exist many non-idealities in both PFD and CP or PFD/CP combination, such as dead zone, blind zone, current mismatch/leakage, charge sharing/injection, etc. These non-idealities are all detrimental to the overall performance of frequency synthesizers and should be avoid or alleviated. Here, the following brief description of how these non-idealities influence the system will give us a preliminary insight into the problems and trade-offs in CPPLL designs.

First, in a CPPLL, one of the critical building blocks is the tri-state PFD due to its frequency-detection capability. A conventional tri-state PFD suffers from the "dead zone" problem, which occurs when the loop is in a lock mode and the output of the following CP don't change for small phase changes in the input signals at PFD. Any phase error within the dead-zone will disturb the VCO control voltage and directly translate to phase jitter in the PLL output, as shown in Figure 1.2.

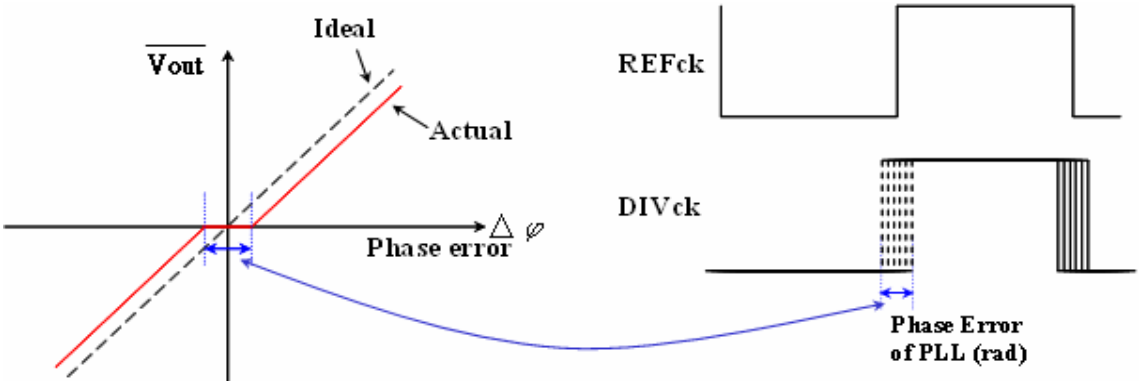


Figure 1.2 PLL jitter introduced by the PFD dead zone

Second, in order to eliminate the dead-zone, an added delay is inserted in the reset path to maintain a minimum pulse width when the two input signals are in phase. However, such a solution presents a limit on its maximum operating frequency and introduces another problem called “blind zone”, where any input transition will be overridden for large phase errors. Any input transition override results in the wrong output polarity (Figure 1.3) and longer frequency acquisition time.

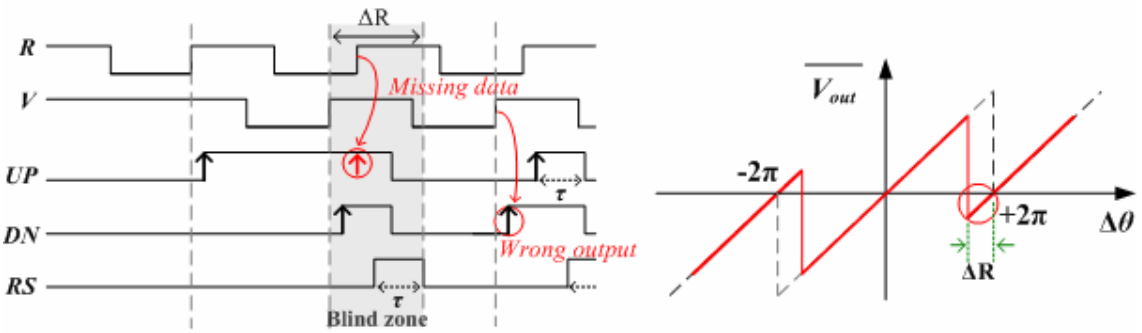


Figure 1.3 The PFD blind zone and its phase characteristic

Third, in addition to the above-mentioned issues, one of the other design issues is the unwanted FM modulation, which causes the reference spurs. Any non-idealities in the PLL itself causes periodic ripples on the VCO control line, which will in turn result in undesired spurs at the upper and lower sideband of the carrier. The dominant spurious-generating block in the PLL is the charge pump, the non-idealities of which mainly are: 1) the mismatch between the CP current sources (both random and due to channel-length modulation); 2) the mismatch between the charge injection and clock feedthrough of the pMOS and nMOS switches in the CP; 3) the mismatch between the arrival times of the input control pulses; 4) the mismatch of the widths of the input control pulses. Charge sharing also exacerbates the ripples [2].

1.2 Related Works

Over the past few years, a large amount of literature has been contributed to the study of how to solve these issues and improve the PLL performance. However, there is still a lot of work to be done in the field. In view of this, the focus of our work is on finding better ways to alleviate or even completely ameliorate these issues but except the FM-modulation issue. The following subsections comes the overview of the related works referred in our thesis.

1.2.1 Review on Phase-Frequency Detectors

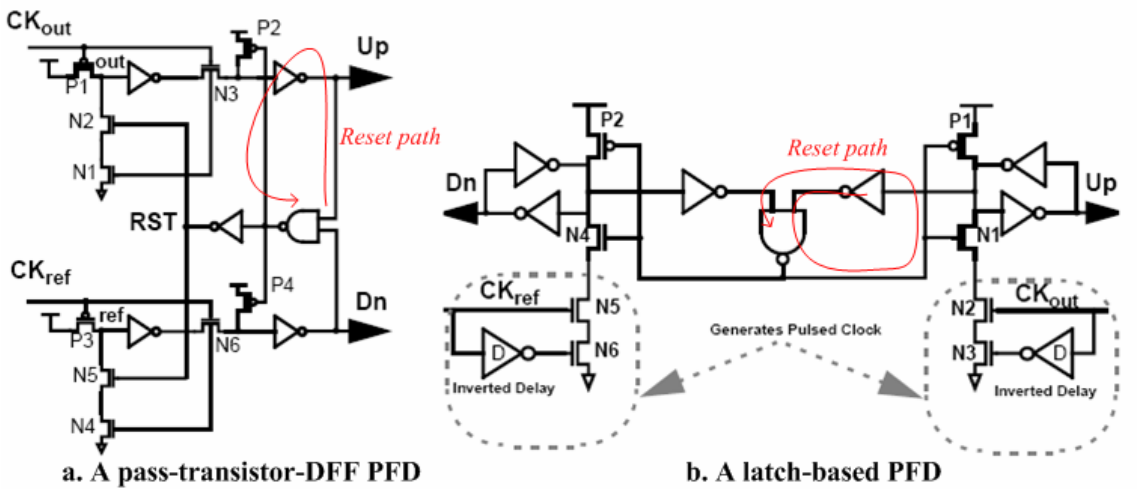


Figure 1.4 Circuit schematics [6] (a) a pass-transistor DFF PFD, and (b) a latch-based PFD

As stated in the previous section, a conventional tri-state PFD suffers from the dead-zone issue which worsens the PLL output jitter. Roughly, there are two ways to alleviate this problem. One way is to reduce the intrinsic reset time [3], [4], thus shortening the dead-zone. This in turn alleviates the speed and jitter limitations. Nevertheless, such a method offers only limited improvement on the issue. The other way is to add an additional delay in the reset path for enough pulse width to drive the

following stage [1], [3]. This would be a relatively simple and effective method, which could completely eliminate the dead-zone. Unfortunately, such a way introduces the blind-zone (Figure 1.3). It's obvious that the blind zone is detrimental to the PLL settling behavior, and will slow down the locking time. In view of this, some extensive studies have been undertaken recently and expected to address both issues together [5]-[8].

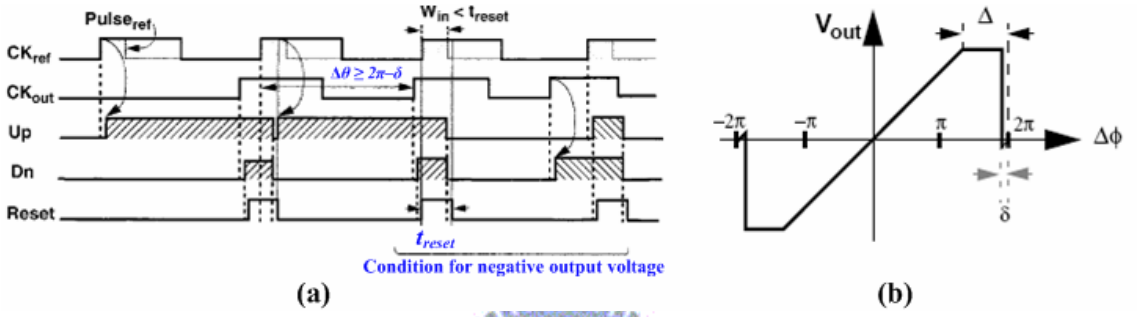


Figure 1.5 (a) Timing diagram and (b) Phase-detection characteristic of the latch-based PFD

In Figure 1.4(a) and (b) [6], two techniques for designing PFD have been presented, a pass-transistor-DFF PFD and a latch-based PFD, respectively. A pass-transistor-DFF PFD shows a smaller reset delay, only including one pass-transistor, one inverter, and one NAND gate. As a result, a reduced blind-zone and faster frequency acquisition can be achieved. In Figure 1.4(b), by using pulse latches instead of flip-flops, the latch-based PFD fundamentally changes the dependence on the reset delay. This is illustrated in the timing diagram of Figure 1.5(a).When CK_{ref} arrives during the Reset , the edge information propagates to the output as long as CK_{ref} is still high (level-sensitive) when the blind-zone duration ends. The PFD no longer loses the edge that arrives during the Reset and doesn't output wrong polarity. The phase-detection characteristic is shown in Figure 1.5(b). It should note that the input pulse widths should be designed to be slightly smaller than the reset pulse width so that an input that triggers

the Reset would not assert the output after the Reset pulse ends; otherwise, the PFD will fail to lock at zero phase error. This design criterion results in wrong output polarity for $\Delta\theta \geq 2\pi-\delta$. Thus, there still exhibits a very small blind-zone, whereas its operating frequency potentially approaches twice that of either the first proposed PFD or the conventional PFD for the same Reset time.

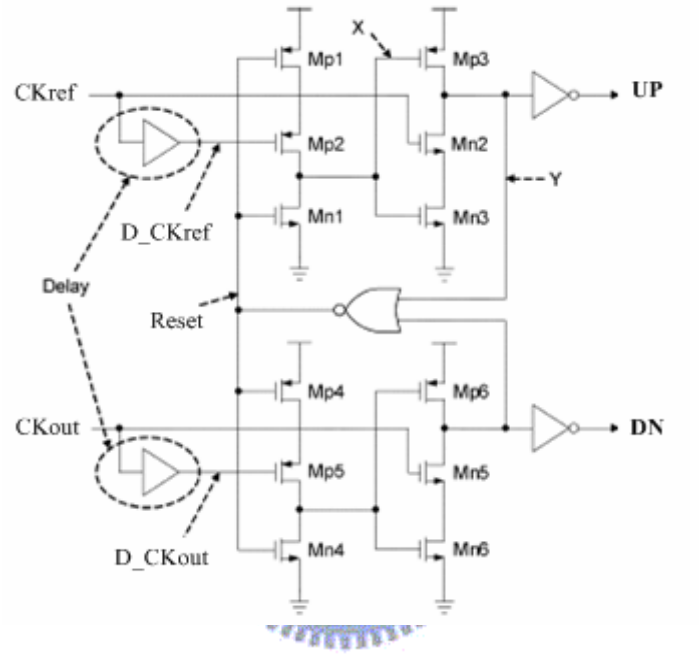


Figure 1.6 A novel precharged PFD in [7]

Figure 1.6 [7] also shows a novel precharged PFD, which employs the same idea as in [6]. Noninverting delay stages are inserted into the commonly used precharged PFD so that it generates effective control signals even when the phase error approaches 2π. The novel precharged PFD has the same phase-detection characteristic but lower power consumption and higher precision as compared to the latch-based PFD [6]. Similarly, it also has a very small blind-zone, and the maximum operating frequency is dependent on the duty ratio of each input clock. Assume 50% duty ratio, the maximum operating frequency is half that of [6].

A robust tri-state PFD architecture [8] is shown in Figure 1.7(a), which does not

rely on any assumption on the underlying VLSI technology. The proposed PFD is divided two parts: one is the classical tri-state PFD using flip-flops with asynchronous set and reset inputs, and the other is the FZ-detector, which takes over the detection process once inside the blind-zone. The proposed PFD forces a set signal when an input transition occurs inside the blind-zone, thereby avoiding setting the wrong output and enhancing the frequency acquisition capabilities. Compared with the conventional PFD, its operating frequency shows an improvement of about 36%. Figure 1.7(b) shows the corresponding phase-detection characteristic.

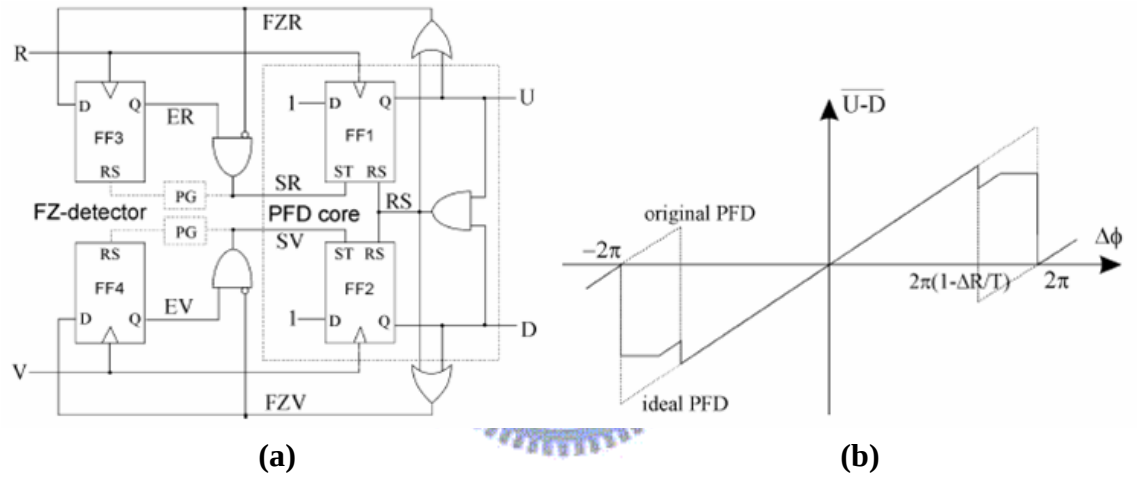


Figure 1.7 (a) The proposed PFD architecture [8]; (b) Its phase-detection characteristic

1.2.2 Review on Dual-Modulus Prescalers

A high-frequency CMOS PLL frequency synthesizer has stringent requirements on dual-modulus prescaler (DMP). High speed, high moduli, and low power dissipation are the challenges in DMP designs. Typically, a DMP usually comprises of a synchronous dual-modulus counter, followed by an asynchronous counter. The critical path delay and the speed of the DFFs in the synchronous counter limit its overall speed, particularly at high divide-by-value. High divide-by-value is, in general, achieved by adding flip-flops

in the asynchronous counter at the cost of additional loading to the synchronous counter which results in degraded performance. Therefore, there is a trade-off between the speed and the divide-by-value.

Conventionally, most high-moduli DMPs usually comprise of a synchronous divide-by-4/5 counter, followed by a chain of toggle flip-flops, which forms an asynchronous counter. The operating speed of DMPs is mainly limited by that of the divide-by-4/5 counter. Unlike the conventional divide-by-4/5 counter [9], a novel topology for a divide-by-3/4 counter using transmission gates (TGs) in the critical path for mode selection is proposed by R. S. Rana [10]. The author has demonstrated that the TG-based divide-by-3/4 counter provides higher speed compared to the conventional divide-by-4/5 counter. However, an alternative divide-by-3/4 counter using NOR gates in the critical path is also presented and taken into account for further comparison by R. S. Rana [10]. With the help of Hspice simulation, the results show that the NOR-based divide-by-3/4 counter provides higher speed than the TG-based one due to smaller feedback path delay even though the critical path delay is more. For enhancing the speed of the TG-based divide-by-3/4 counter further, the author expects to shorten the D flip-flop (DFF) delay for future improvement.

1.3 Motivation

Nowadays, modern wireless system applications have an increasing demand to fabricate low-cost high-performance RF integrated circuits. In the world of wireless communications, frequency synthesizer is one of the critical components for RF front-end transceivers. As the demands for high-performance RF front-end transceivers grow rapidly, designing high-performance PLL-based frequency synthesizers becomes challenging. In general, the challenging design requirements of a frequency synthesizer

are low jitter or phase noise, fast settling time, and low power etc.

How to design a high-frequency, fast-settling, and low-jitter synthesizer? We start this thesis with a thorough insight into the dead-zone and blind-zone problems. We then survey the recently-published related literature. Afterward, we try to find new ways of eliminating these problems and further achieving a low-jitter and faster-locking frequency synthesizer. For high-frequency applications, the need for a high-speed and high-moduli dual modulus prescaler (DMP) also introduces stringent requirements due to its limitation on the operating speed of the frequency synthesizer. So, we also make some efforts to develop a high speed DMP in this work.

Therefore, this thesis focuses on the design of the new architecture of PFD, which incorporates an auxiliary circuitry to enhance the frequency acquisition capabilities. The novel proposed architecture suffers no problem of dead zone and blind zone, and has much better phase- and frequency-detection characteristics. It is implemented in a 2.36~2.95-GHz CMOS frequency synthesizer, including an improved high-speed divide-by-3/4 dual-modulus prescaler.

1.4 Thesis Organization

To achieve a low-jitter and fast-locking frequency synthesizer, this thesis presents a new fast frequency acquisition PFD with better phase- and frequency-discriminator characteristics. It also presents a new dual-modulus prescaler for high speed operation. This thesis is organized into the following chapters. Chapter 2 will give some general design considerations of PLL-based frequency synthesizer as well as basic behavior characteristics of its individual building blocks. A comprehensive and in-depth analysis of the noise behavior and loop stability of frequency synthesizers provides an insight into the design issues and trade-offs. Chapter 3 introduces a novel fast frequency

acquisition PFD with a detailed analysis of its phase- and frequency-discriminator characteristics. With the help of HSPICE and ADS simulation, the results demonstrate its robustness in PLL designs. Chapter 4 introduces a modified high-speed divide-by-3/4 dual-modulus prescaler and provides some simulation results to prove its improvement over the original one published by Rana [10]. Chapter 5 presents the circuit design and implementation of the building blocks of the frequency synthesizer, including voltage-controlled oscillator (VCO), frequency divider, PFD, charge pump and loop filter. Chapter 6 concludes this thesis with a summary and future work.

CHAPTER 2

Basics of Frequency Synthesizers

In a typical RF front-end circuit, the local oscillator is usually embedded in a phase-locked loop (PLL) as a tunable frequency synthesizer to provide clean, stable and more precise carrier signals for frequency up/down-conversion. The frequency synthesizer needs to be tunable in order to address all frequency channels and be fast switching to perform the addressing sufficiently fast. A basic block diagram of a PLL-based frequency synthesizer is shown in Figure 2.1. The loop provides a feedback to keep the output frequency of the synchronized oscillator to be a multiple of the reference frequency, i.e. $f_{out}=M\cdot f_{ref}$, where f_{out} is the output frequency and f_{ref} is the reference frequency.

This chapter begins with the general design considerations of a PLL-based frequency synthesizer, following which comes the overview of two widely-used frequency synthesizer architectures. Subsequently, the behavior characteristics of individual functional blocks (in Figure 2.1) are described and discussed. In conclusion, the noise behavior and loop stability of a frequency synthesizer are analyzed then to draw some conclusions of design issues and trade-offs.

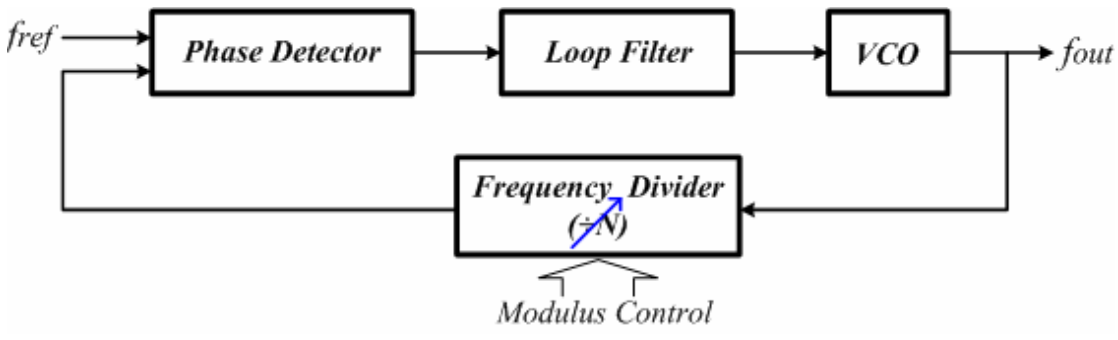


Figure 2.1 Block diagram of a PLL-based frequency synthesizer.

2.1 General Considerations

The most important design considerations of a frequency synthesizer are tuning range, phase noise, spurs, and settling time. Their impacts on general wireless communication systems are investigated in this chapter.

2.1.1 Tuning Range

The basic requirement set for a frequency synthesizer by any wireless communication system is that the synthesizer must be able to generate all required frequencies of the system with a sufficient accuracy for channel selection. Therefore, the voltage-controlled oscillator (VCO) and the prescaler must be carefully designed so as to cover the required dynamic frequency range of the synthesizer.

2.1.2 Phase Noise

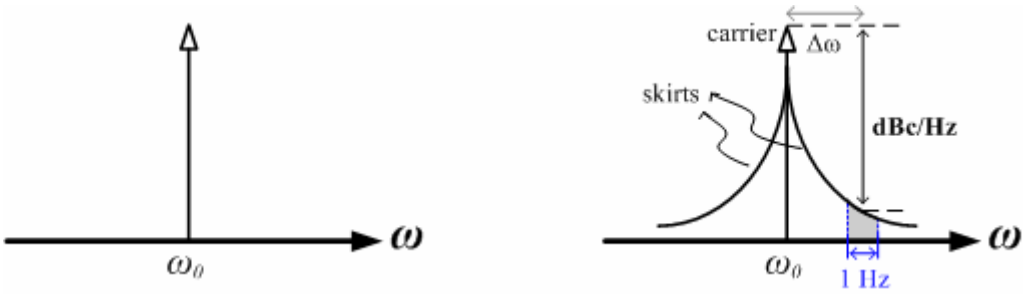
A spectral purity of synthesized output signal is the most important requirement in all wireless communication systems. Ideally, the output spectrum of a frequency synthesizer should be a pure tone at the desired frequency, as shown in Figure 2.2(a). In the time domain, the output can be expressed by Eq. (2.1).

$$v_{out}(t) = A \cdot \cos(\omega_0 t) \tag{2.1}$$

However, due to random amplitude and phase fluctuations, the actual output becomes

$$v_{out(t)} = \left[A + \varepsilon(t)\right] \cdot \cos\left[\omega_0 t + \theta(t)\right] \tag{2.2}$$

, where $\varepsilon(t)$ represents amplitude fluctuations and $\theta(t)$ represents phase fluctuations. The actual output spectrum exhibits “skirts” around the desired carrier impulse in the frequency domain, as shown in Figure 2.2(b). Because the amplitude fluctuations can be removed or greatly reduced by a limiter, the phase fluctuations, expressed in terms of phase noise, become a bigger and dominant concern in frequency synthesizer design. The phase fluctuations could be attributed to either the external noise at the frequency-



(a) **(b)**

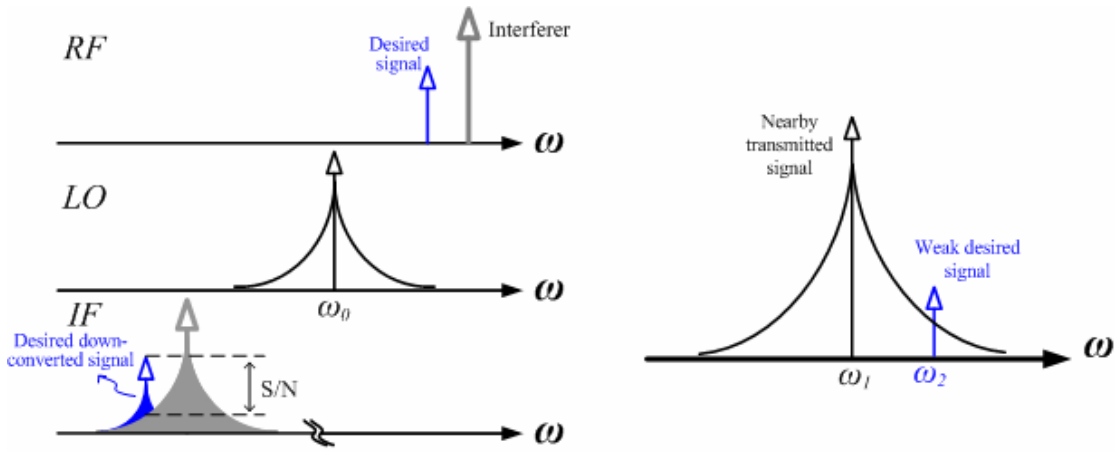
Figure 2.2 (a) Ideal; (b) Actual output spectrum of an oscillator.

tuning input of the oscillator or the noise sources such as thermal, shot, or flicker noise of the devices in the oscillator.

The phase noise limits the quality of the synthesized signal. In order to quantify the phase noise, the total noise power within a unit bandwidth at an offset frequency ($\Delta\omega$) from the carrier frequency (ω_0) is compared with the carrier power. As shown in Figure 2.2(b), this quantity is defined as Eq. (2.3) in the unit of dB_C / Hz.

$$L\{\Delta\omega\} = 10 \cdot \log\left[\frac{P_{sideband}(\omega_0 + \Delta\omega, 1Hz)}{P_{carrier}}\right] \tag{2.3}$$

, where $P_{sideband}(\omega_0 + \Delta\omega, 1Hz)$ represents the single sideband noise power within a 1Hz bandwidth at an offset frequency ($\Delta\omega$).



(a) **(b)**

Figure 2.3 Effect of phase noise in (a) the receive path, and (b) the transmit path.

Figure 2.3 illustrates the impact of the oscillator or synthesizer phase noise in both the receive path and transmit path of a transceiver. As depicted in Figure 2.3 (a), in the receive path, the weak desired signal is accompanied by a larger interferer in the adjacent channel. Ideally, the received RF signal is down-converted with a pure LO signal into the desired pure IF signal and the down-converted interferer can be easily filtered. However, in reality, there exists a phase noise skirt around the LO signal. After down-conversion, the weak desired signal could be corrupted by the tail of the interferer spectra and even possibly swamped out if the phase noise skirt is too large. This effect is called "*reciprocal mixing*", and it degrades the SNR of the desired signal. In the transmit path, the weak nearby signal of interest can be corrupted by the tail of the large-power transmitted signal, as shown in Figure 2.3 (b).

Therefore, the output spectrum of the LO or synthesizer must be extremely sharp, and a set of stringent phase noise requirements must be achieved so as to satisfy the maximum blocking signal power specified in the wireless communication system.

2.1.3 Spurs

Apart from the phase noise, the other key parameter affecting the spectral purity of synthesized output signal is the relatively high-energy spurious tones (also called spurs), appearing as spikes above the noise skirt, as shown in Figure 2.4 (a).

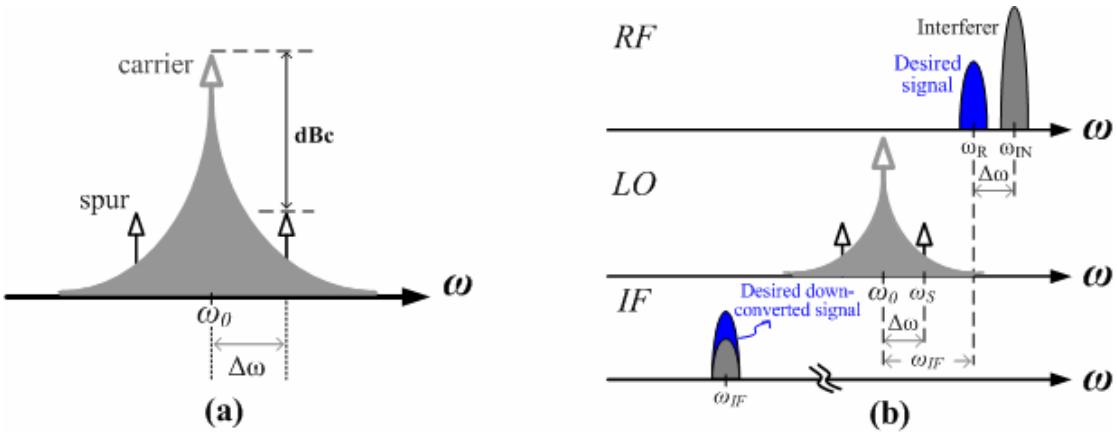


Figure 2.4 (a) Spurs, and (b) effect of spurs in the receive path

Any systematic disturbance on the tuning input of the oscillator will cause the periodic phase variation and thus modulate the synthesized output. In the frequency domain, it manifests itself as the undesired tones at the upper and lower sideband of the carrier. These tones can be quantified by the difference between the carrier power and the spurious power at certain frequency offset in the dB_C unit. The most common type of spur is the reference spur that appears at multiples of the comparison frequency. Due to the non-ideal switching nature of the synthesizer, it may cause reference frequency feed-through, and then the resulting periodic ripples on the tuning input of the oscillator induces the reference spurs at the output, as shown in Figure 2.5.

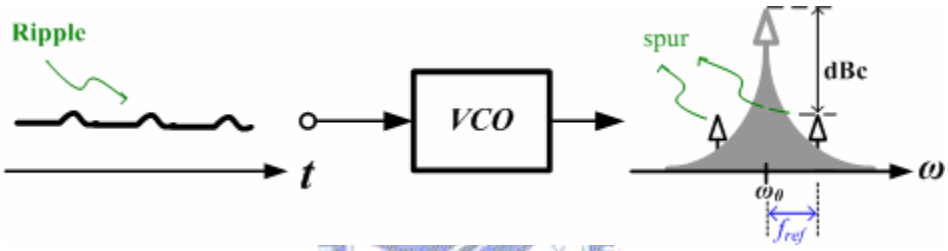


Figure 2.5 Reference frequency feed-through

As illustrated in Figure 2.4 (b), similar to the case of phase noise, if a large interferer is close to the weak desired signal and the LO signal has spurs, then both the desired signal and interferer will be mixed down to the IF. If the spacing between the desired signal and the interferer is equal to that between the LO signal and the spur, the spur in the down-converted interferer falls into the center frequency of the desired down-converted signal, and then also degrades the SNR performance.

Phase noise and spurious tones in the synthesized signal can limit the ability to receive a weak desired signal in the presence of strong interferers and this ability is called "*selectivity*". In later sections, all the contributors and causes of the phase noise and spurs will be addressed so as to specify the design trade-offs.

2.1.4 Settling Time

Transient behavior of the frequency synthesizers is also a critical performance parameter. As shown in Figure 2.1, a change in the division ratio of the frequency divider would result in a loop transient. Every time a different division ratio is set for channel selection, the synthesizer requires a finite time to lock to the new frequency. The synthesizer needs settling to certain accuracy within the specification of the wireless standard and the overall required time is called "*settling time*" (also called "*locking time*"). Also, one thing worth mentioning is that the locking speed requirement of synthesizers is even more stringent for a fast frequency-hopping spread-spectrum system. Accordingly, a detailed analysis to model the loop settling behavior will also be discussed in later sections.

2.2 Frequency Synthesizer Architectures

The well-known ways to implement a frequency synthesizer can be categorized into three types: the table-look-up synthesis, the direct synthesis, and the indirect or PLL-based synthesis. However, the growing call for miniaturization, low power, low cost and the move towards higher frequencies for emerging communication techniques become critical trends. Therefore, with the ability of high-integration in low-cost CMOS process, the PLL-based frequency synthesizer is the widely-used method in today's frequency synthesis. An overview of two PLL-based synthesizer architectures: integer-N and fractional-N synthesizer will be introduced in this section.

2.2.1 Integer-N Architecture

The generic PLL-based frequency synthesizer generates its output by phase-locking the divided output to a reference signal. Due to a low cost IC solution, a charge pump is widely used in PLL-based frequency synthesizers nowadays. As shown in

Figure 2.6, an idea charge pump combined with an ideal phase-frequency detector (PFD) provides an infinite dc gain with passive loop filters, which results in an unbounded pull-in range and zero static phase error. "Integer-N" means that the division ratio N of the frequency divider is a variable integer. In other words, the synthesized output frequency is integer multiples of the input reference frequency. In general, f_{ref} is fixed and the frequency step or channel spacing is equal to f_{ref}. Various frequencies are achieved by changing the division ratio N. In the locked state, the output frequency is as follows:

$$f_{out} = N \cdot f_{ref} \tag{2.4}$$

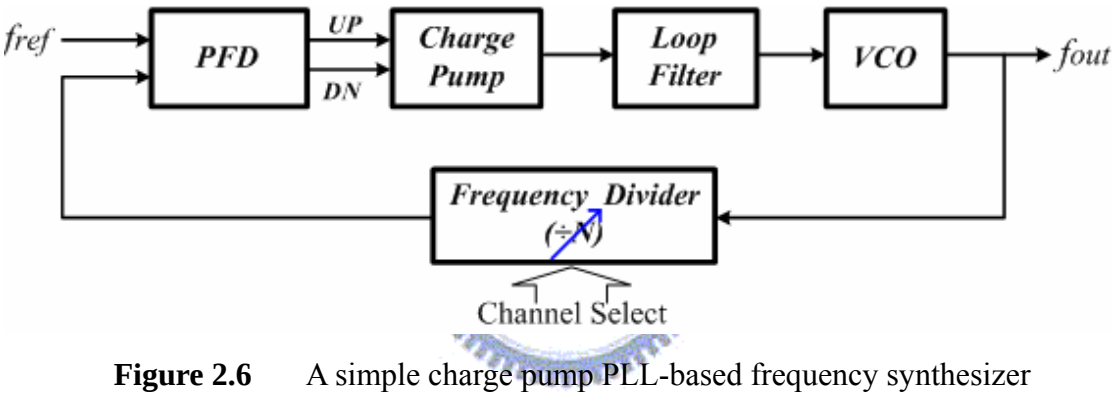


Figure 2.6 A simple charge pump PLL-based frequency synthesizer

In the design of integer-N frequency synthesizers, to achieve the fine frequency resolution, a low f_{ref} is needed. This low f_{ref} yields a high division ratio as well as a narrow loop bandwidth. However, a narrow loop bandwidth results in slower settling speed of transients and deteriorates the in-band phase noise. So, the loop performance of the integer-N architecture is intrinsically limited by the standard-specified frequency resolution. Generally, a larger loop bandwidth is desired to achieve a faster dynamic loop response and suppress the VCO close-in phase noise, but otherwise the reference frequency leakage becomes serious. Additionally, as a rule of thumb, the loop bandwidth should be *10* times less than f_{ref} for the consideration of the loop stability under linear, continuous-time approximation [1].

Figure 2.7 shows the generic structure of the full frequency divider (the block *"÷N"* in Figure 2.6), which consists of a dual-modulus prescaler (DMP), a swallow counter and a programmable counter. The dual-modulus prescaler, which is dedicated to the high-frequency operation, follows VCO so as to relieve the constraint of the operating speed of the counters. The DMP divides by $(P+1)$ until the swallow counter overflows after which the overflow bit (Modulus Control) will set the DMP in divide-by-P mode until the programmable counter overflows. Then, the overflow bit (Reset) will reset both two counters and the division process restarts. Therefore, the overall division ratio becomes $(P+1)\cdot A+P\cdot(B-A)=P\cdot B+A$. The detailed discussion of the frequency divider will be presented in section 2.3.

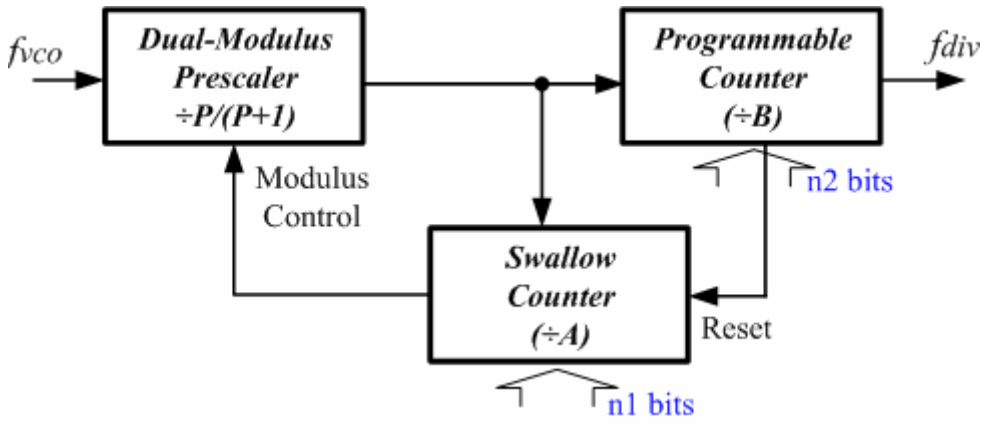


Figure 2.7 A full divider with a dual-modulus prescaler and two counters

2.2.2 Fractional-N Architecture

As mentioned above, the loop performance of the integer-N architecture is restricted by the given frequency resolution. Compared with the integer-N architecture, however, the fractional-N architecture [11]-[13] shown in Figure 2.8 allows a higher reference frequency for a desired fine frequency resolution. The higher reference frequency implies the wider loop bandwidth, yielding faster settling speed and more suppression of VCO close-in phase noise.

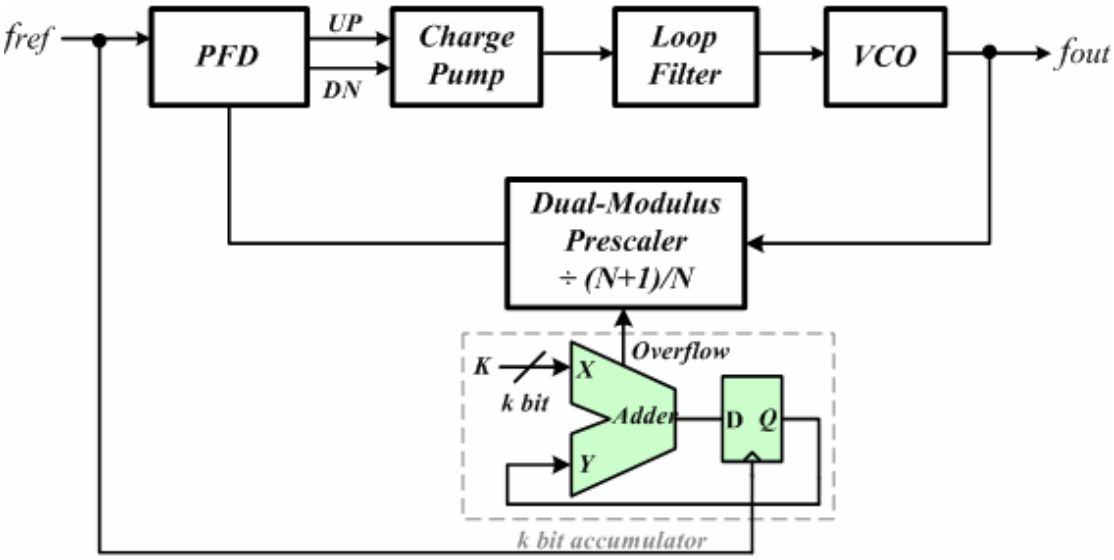


Figure 2.8 A fractional-N frequency synthesizer with an accumulator

A simplified block diagram of a fractional-N frequency synthesizer utilizing an accumulator is shown in Figure 2.9. The accumulator, which consists of an adder and a latch, is clocked by the reference frequency. The output (X+Y) of the adder is latched and then fed to the adder as an input Y. The input X contains the data (K) to be accumulated. When the total (X+Y) exceeds the maximum size (2^k) of the adder, an overflow occurs and then the division ratio of the DMP is changed. The DMP divides its input by N when the accumulator is not overflow. When an overflow occurs, the DMP divides its input by (N+1). For every 2^k clock cycles, the accumulator overflows K times. Thus, the DMP divides K cycles by (N+1) and 2^k-K cycles by N, resulting in an average division ratio:

$$N_{avg} = \frac{K \cdot (N+1) + (2^k - K) \cdot N}{2^k} = N + \frac{K}{2^k} \tag{2.5}$$

It can be seen that, therefore, f_{ref} can be many times the frequency step resulting in a higher loop bandwidth without compromising the settling speed and the in-band phase noise. But the periodically alternating process of the DMP causes a sawtooth phase error. This phase error will generate severe spurious tones, which are called “fractional spurs”

at all multiples of the offset frequency ($\frac{K}{2^k} \times f_{ref}$) if not filtered. This sawtooth phase error could be predictable and further eliminated by the classical phase interpolation method [14], as shown in Figure 2.8. This compensation technique requires accurate matching of compensation signals which is sensitive to temperature and process variations. Another solution for large fractional spurs is to randomize the prescaler modulus by using a higher-order Σ-Δ modulator [13]. Its noise-shaping property shapes the noise spectrum of the carrier such that most of the noise energy can appear at large frequency offsets and be pushed outside the loop bandwidth, hence the close-in noise can be suppressed. Arbitrarily fine frequency resolution can be achieved, limited only by the size of the digital adder.

Concerning Σ-Δ Fractional-N PLLs, abundant literatures have been published, and thereby we ends up with the above discussion due to being far away from the focus of this thesis.

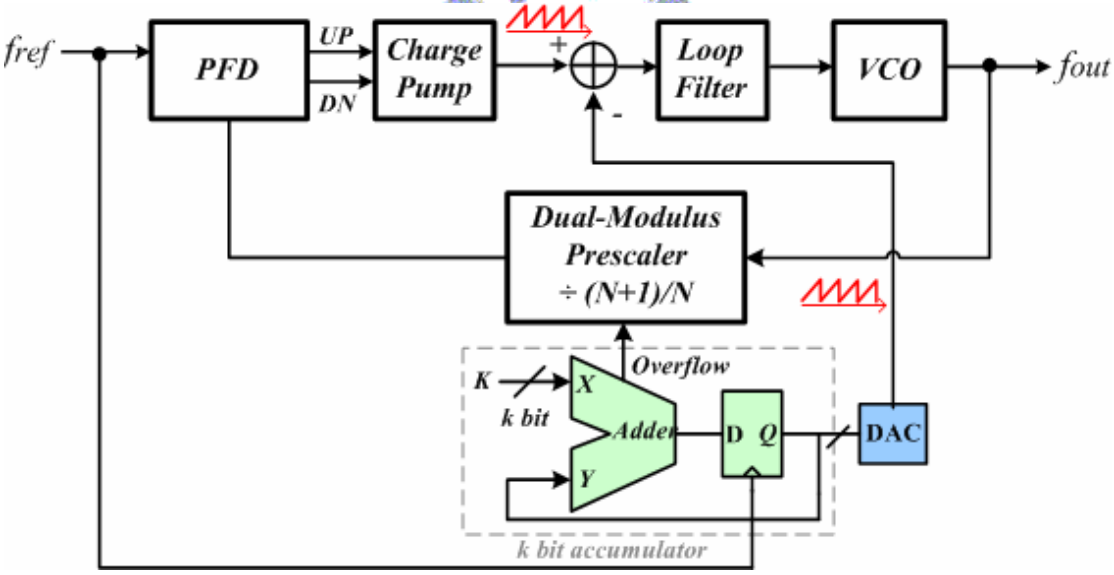


Figure 2.9 Classical phase interpolation method for spur cancellation

2.3 Fundamentals of Phase-Locked Loop (PLL)

When designing a PLL-based frequency synthesizer, it is very important to have the complete knowledge of the behaviors of each functional block as well as the overall closed-loop behavior. As discussed earlier, a charge pump is widely used in PLLs nowadays. Therefore, the following subsections focus on the discussion of integer-N charge pump PLL-based (CPPLL-based) frequency synthesizers.

2.3.1 Phase-frequency Detector (PFD)

An idea phase detector (PD), as shown in Figure 2.10, produces an output whose average dc value is linearly proportional to the phase difference between its two periodic inputs, namely the reference signal *(R)*and the divided signal *(V)*.

$$\overline{v_{out}} = K_{PD} \cdot \Delta\theta \tag{2.6}$$

, where K_{PD} is the gain of the phase detector (specified in V/rad) and $\Delta\theta$ is the input phase difference.

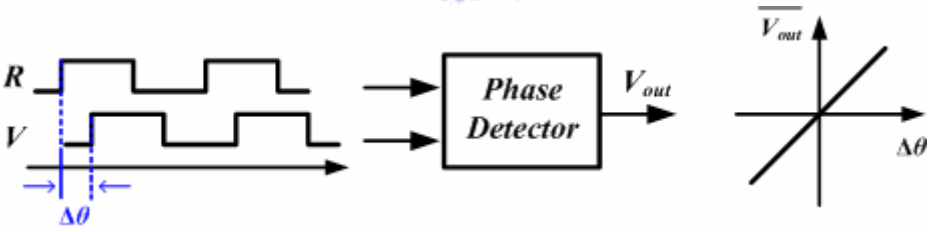


Figure 2.10 Characteristic of an ideal phase detector

Nowadays, the common phase detectors can be categorized to three types: the analog multiplier-type, the XOR-type, the sequential-type. However, the analog multiplier-type phase detector exhibits the nonlinear dependence of the output voltage on the phase difference and the instability, resulting from the inverted gain polarity beyond the phase difference range, ± π/2. Concerning the XOR-type phase detector, the instability issue still exists except for the nonlinear dependence.

Unlike the multiplier-type and XOR-type, which are only phase-sensitive, the phase-frequency detector (PFD) is a sequential phase detector triggered by else the rise or fall edge of the reference signal *(R)* and the divided signal *(V)*. As a result, the PFD has the capability of operating as a frequency discriminator for large frequency errors or as a coherent phase detector once inside the pull-in range of the PLL, allowing a fast frequency acquisition and a full linear phase difference range, ± 2π (shown in Figure 2.11).

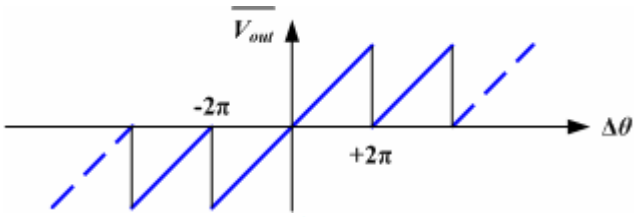


Figure 2.11 PFD characteristic

As illustrated in Figure 2.12, the operation of a typical PFD is as follows. If initially *UP* = *DN* = *0*, then a rising transition on *R* leads to *UP* = *1*, *DN* remains *0*. The circuit remains in this state until *V* goes high, upon which *UP* returns to zero simultaneously. The behavior is similar for the *V* input. Thus, the average dc value of *(UP-DN)* is an indication of the frequency or phase difference between *R* and *V*.

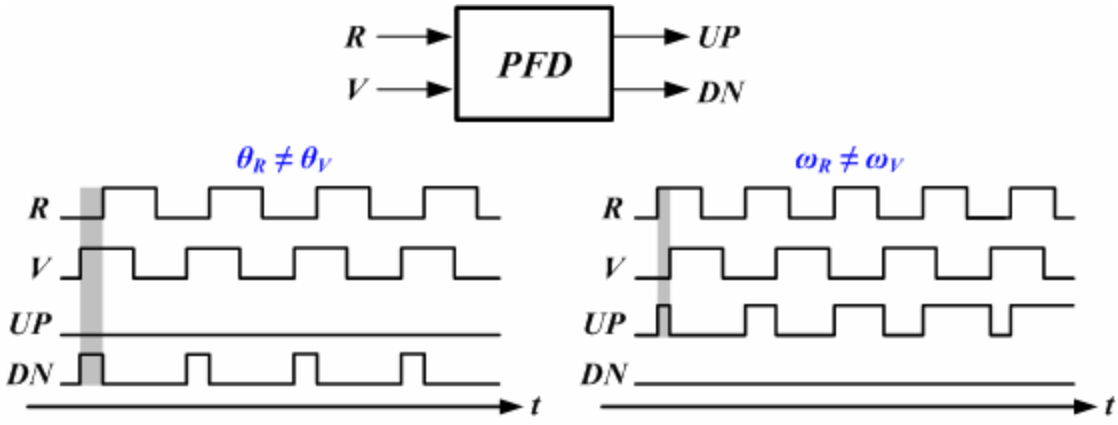


Figure 2.12 Conceptual operation of a PFD

To achieve a PFD with the above behavior, at least, three logical states are required: *UP* = *DN* = *0* (state 0); *UP* = *1*, *DN* = *0* (state I); *UP* = *0*, *DN* = *1* (state II). Figure 2.13 shows a state diagram summarizing the operations. If the PFD is in the state 0, then a transition on *R* takes it to state I. During state I, any more rising edge on *R* won't changes the state at all. The PFD will remain in this state until a transition occurs on *V*, upon which the PFD returns to state 0 immediately. The switching sequence between state 0 and state II is similar. Such a PFD is called "*tri-state phase-frequency detector*".

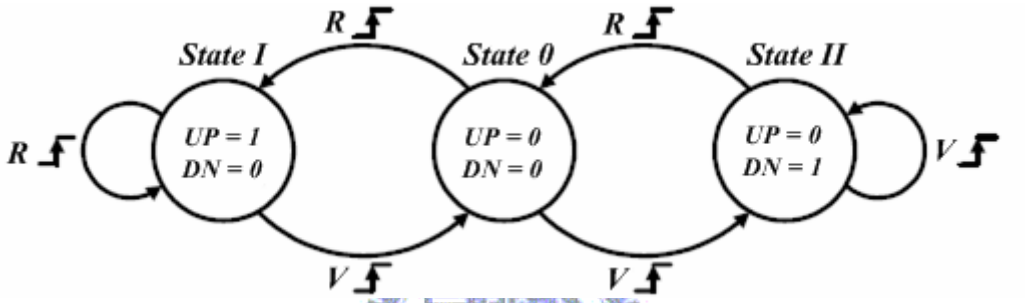


Figure 2.13 State diagram of a three-state PFD

2.3.2 Charge Pump (CP)

A PFD couldn't alone provide the exact voltage (or current) signal proportional to the phase difference at its inputs. A charge pump serves to convert the difference of the two output signal *UP* and *DN* of the PFD into the corresponding error current either sourced to or sunk from the loop filter, depending on the state of the switches S_U and S_D controlled by *UP* and *DN*, respectively. No current flows through the loop filter if both switches are off and the output node represents an infinite-impedance towards the loop filter.

A charge pump (CP) with a PFD and a capacity C_P as the loop filter is shown in Figure 2.14, which illustrates the corresponding time-domain response. One should note that the system is nonlinear and discrete time in the strict sense. To overcome this

quandary, it can be approximated by a continuous-time model only when the loop bandwidth is much less than the reference frequency [1]. Therefore, the characteristic of the PFD and charge pump can be together approximated linearly as:

$$\overline{I_e} = I_P \cdot \frac{\Delta\theta}{2\pi} \tag{2.7}$$

, where $\overline{I_e}$ is the average error current over a cycle, $\Delta\theta$ represents the phase error between the PFD inputs and $I_P = I_1 = I_2$ is the current value of the two current sources in the charge pump.

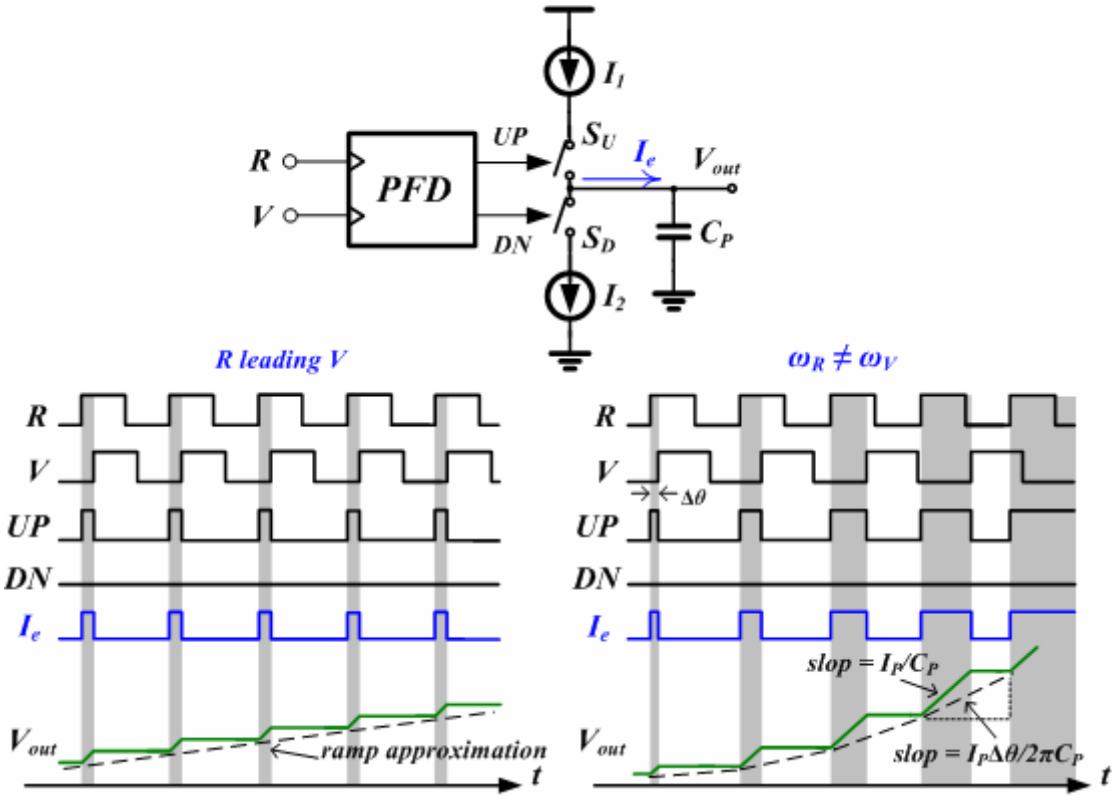


Figure 2.14 Block diagram of PFD with CP, and the timing diagram

2.3.3 Loop Filter (LF)

The loop filter (LF) determines most of the PLL's specifications. In *CPPLLs*, unlike many other feedback systems, the variable of interest changes dimension around the loop: it is converted from phase to current by the PFD/CP, processed by the LF as

such, and then converted back to phase by the VCO. More specifically, the loop filter serves to convert the error current I_e, proportional to detected phase error, into the corresponding control voltage for VCO. In general, the loop filter can be realized in either active or passive forms, and its design has a great impact on the overall system performance of *CPPLLs*, such as the loop stability and noise rejection. The stability issue of great concern tends to apply for low order filters, which conflicts with the noise-rejection requirement. Figure 2.15 shows three features of the passive filter commonly adopted in most *CPPPLs*, namely first-order, second-order, and third- order filters, respectively.

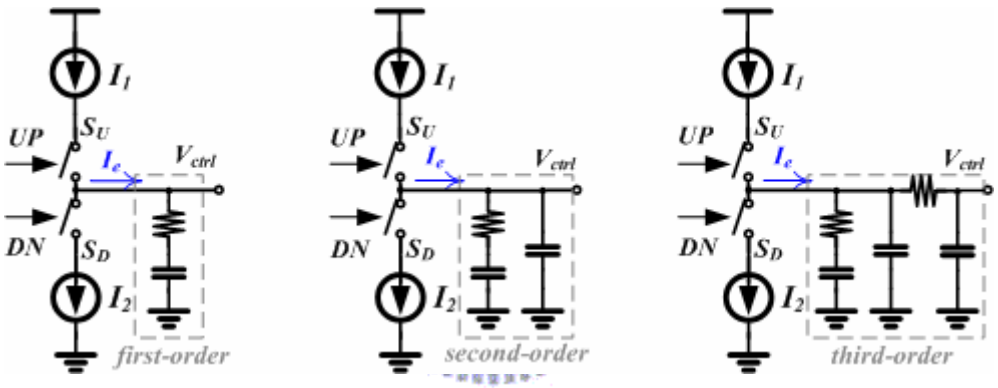


Figure 2.15 Different features of the passive filter

2.3.4 Voltage-Controlled Oscillator (VCO)

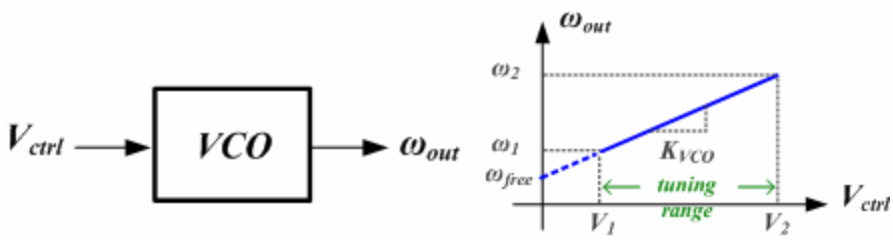


Figure 2.16 VCO characteristic

An ideal voltage-controlled oscillator, as shown in Figure 2.16, is a circuit that generates a periodic output whose frequency is a linear function of a control voltage

(V_{ctrl}):

$$\omega_{out} = \omega_{free} + 2\pi K_{VCO} \cdot V_{ctrl} \tag{2.8}$$

Here, ω_{out} represents the free-running frequency, and K_{VCO} denotes the "gain" of the VCO or "sensitivity" of the circuit (specified as Hz/V). Since the phase is the integral of frequency with respect to time, the output signal of a sinusoidal VCO can be expressed as

$$y(t) = A \cdot \cos\left[\omega_{free} t + 2\pi K_{VCO} \int_{-\infty}^{t} V_{ctrl}(t)dt\right] \tag{2.9}$$

As depicted in Figure 2.17, for example, if applying a step voltage as follows:

$$V_{ctrl}(t) = \Delta V \cdot \big(u(t) + u(t - t_1)\big), t_1 > 0 \tag{2.10}$$

, then the VCO output signal can be expressed as

$$y(t) = A \cdot \cos\left[\omega_{free} t + 2\pi K_{VCO} \int_{-\infty}^{t} \Delta V \cdot \big(u(t) + u(t - t_1)\big)dt\right]$$

$$= A \cdot \cos\left[\omega_{free} t + 2\pi K_{VCO} \cdot \Delta V \cdot \big(u(t) + u(t - t_1)\big)t\right] \tag{2.11}$$

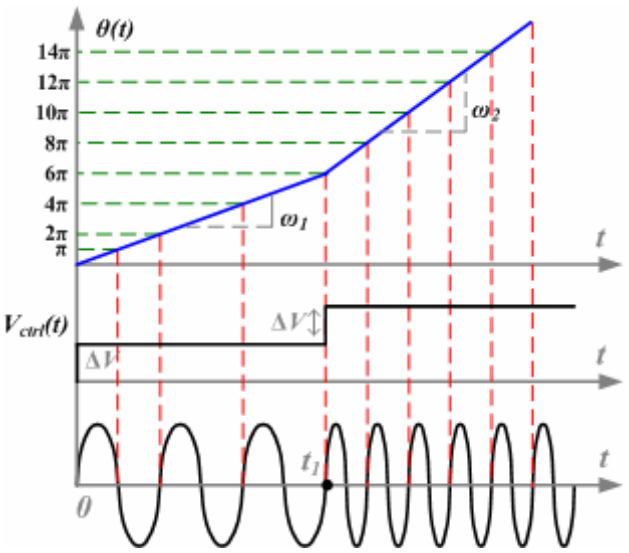


Figure 2.17 Illustration of phase variation of VCO with a voltage step ΔV.

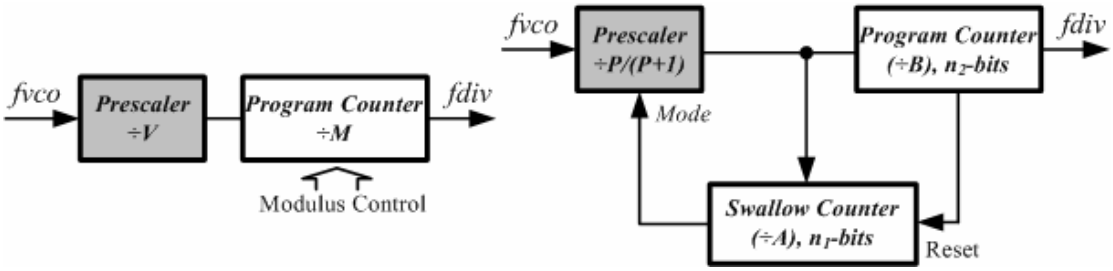
Thus, as expected, when $t = 0$, V_{ctrl} experiences a voltage step ΔV and then the VCO frequency is simply shifted up by $2\pi K_{VCO}\Delta V$. When time goes to t_1, the frequency is shifted up by $2\pi K_{VCO}\Delta V$ again. The illustration reveals that, the higher the VCO frequency, the faster the VCO phase accumulates. However, in the analysis of PLLs, the VCO is viewed as a linear time-invariant system whose input and output are the control voltage (V_{ctrl}) and the output excess phase ($2\pi K_{VCO}\int_{-\infty}^{t} V_{ctrl}(t)dt$), respectively. Thus, the input-output transfer function is

$$\frac{\theta_{out}}{V_{ctrl}}(s) = \frac{2\pi K_{VCO}}{s} \tag{2.12}$$

2.3.5 Frequency Divider

As discussed earlier, the output frequency of a PLL-based frequency synthesizer is $f_{out}=N\cdot f_{ref}$. Thus, with a fixed reference frequency, the controllability of the output frequency or channel selection depends on the controllability of the division ratio N. In order to achieve such a channel selection function, a frequency divider with full programmability of N in an arbitrary range serves to change the output frequency of the VCO according to the digital channel-selection control input.

Such a programmable frequency divider can be easily implemented with standard CMOS logic, for example, a digital programmable counter. However, with the growth of wireless communication systems, the wireless frequency band goes higher, i.e. over GHz. Thus, it is not an easy task to realize such a divider completely in standard CMOS logic, especially for a high-moduli divider. Even if the process technology evolves so that such a digital counter could operate at GHz range, the considerable power consumption prevents one from using it in most PLL designs. Usually, a simple prescaler is hence used in the front to lower the operating frequency of the actual programmable divider. Depending on the system constraints, two different types of prescalers are used: fixed modulus and dual modulus.



(a) **(b)**

Figure 2.18 Different types of prescaler: (a) fixed modulus, and (b) dual modulus

The simplest implementation of the prescaler is the first-type fixed modulus high-speed prescaler, as shown in Figure 2.18 (a). The purpose of such a high-speed architecture is to lower the frequency to some extent before the actual programmable divider. However, dividing the output frequency by a fixed factor V means N can only be chosen in steps of V. This could, of course, be compensated by decreasing the reference frequency by the same factor, resulting in a narrow loop bandwidth. This would yield slower settling of PLL's transients and less suppression of the VCO close-in phase noise. Also, a lower reference frequency implies a higher division ratio. The noise of the reference oscillator, PFD, CP, and LF is seen in the output of PLLs multiplies by N. Thus, increasing N would tighten the requirements for the rest of the PLL building blocks. So, such a fixed prescaler can only operates in systems which allow long switching times between frequency changes.

Due to the drawbacks of the fixed modulus prescalers, instead, dual-modulus prescalers (DMPs) are almost always preferred. Having two possible moduli instead of one increases the circuit complexity only slightly, and none of the problems of increasing N and decreasing the reference frequency occur in this case. In general, two types of DMPs are developed: conventional dual-modulus architecture and phase switching architecture. Nevertheless, only the dual-modulus architecture is the focus of our concern in this work. Accordingly, our work will concentrate on the popular

pulse-swallow frequency divider, which consists of a DMP, a swallow counter and a programmable counter, as shown in Figure 2.18 (b). The two counters used here are n_1-bit and n_2-bit counter, respectively.

Figure 2.19 illustrates the timing diagram of the whole frequency divider. If *Mode* = *1*, the input frequency from VCO, f_{VCO}, is divided by $(P+1)$ by the DMP. In this instant, both the counters begin to counter. After the n_1-bit swallow counter counts "*A*" clocks, it will overflow and output a control pulse to set *Mode* from *1* to *0*. Once *Mode* is set to *0*, the DMP alters to divide by *P* and the swallow counter is disabled. Then, only the n_2-bit programmable counter keeps on count in this mode. After the programmable counter counts the residual "*B-A*" pulses, it will overflow and reset both two counters. Besides, the DMP will be set back in divide-by-$(P+1)$ mode and then a new division cycle begins. During a period of the divided output signal, there are $(P+1)\cdot A+P\cdot(B-A) = P\cdot B+A$ input clocks enter the DMP. In other words, $f_{VCO} = (P\cdot B+A)\cdot f_{div}$.

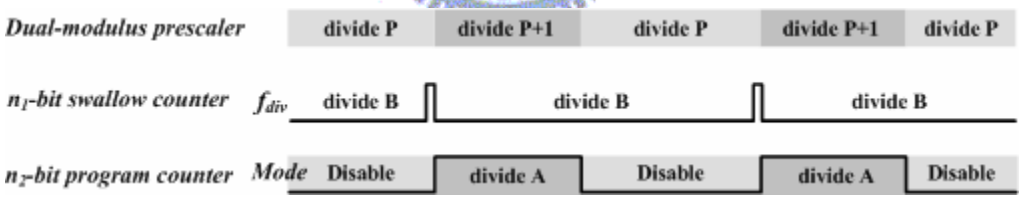


Figure 2.19 Timing diagram of the pulse swallow frequency divider

2.4 Modeling and Analysis of Frequency Synthesizer

A PLL is a negative feedback system that operates on the excess phase of nominally periodic signals. Duet to the sampling and switching nature of the PFD/CP, the transient response of PLL is generally a nonlinear phenomenon. Although the PLL is a highly nonlinear system, it has been found that when the loop is in lock and its state changes by only a very small amount on each cycle of the input, it can be reasonably well approximated as a linear system [1]. With this linear approximation, a several Laplace transfer functions will be derived to gain insight into trade-offs in PLL design and the performance of PLLs, such as loop stability, static phase error, and transient response in this section.

2.4.1 Loop Stability Analysis

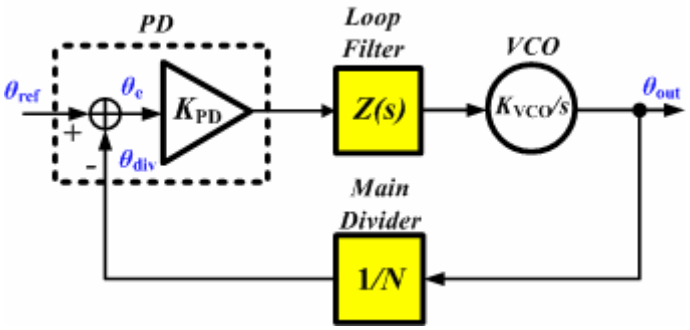


Figure 2.20 Linear model of a generic PLL

Figure 2.20 illustrates a linear model of a generic PLL. Note that the input and output variables of the PLL are phases. The phase detector compares the phase of the input reference signal θ_{ref} and the phase of the divided signal θ_{div}. It produces a phase error θ_e, and then converts this phase error with a gain K_{PD} (i.e. $I_P/2\pi$ for CPPLLs) into a corresponding error signal $V_e(s)$ or $I_e(s)$. That is,

$$I_e(s) = K_{PD} \cdot [\theta_{ref}(s) - \theta_{div}] = K_{PD}\theta_e(s) \tag{2.13}$$

Then, the loop filter $Z(s)$ filters the error signal $V_e(s)$ and produces the VCO control voltage $V_{ctrl}(s)$, equal to:

$$V_{ctrl}(s) = Z(s) \cdot V_e(s) \tag{2.14}$$

Recall from (2.12) that the VCO is modeled as a phase integrator. This results in an output excess phase θ_{out} as follows:

$$\theta_{out}(s) = V_{ctrl}(s) \cdot \frac{2\pi K_{VCO}}{s} \tag{2.15}$$

In turns, this output phase $\theta_{out}(s)$ is fed back and passes through the frequency divider. Because frequency and phase are related by a linear operator, division of frequency by a factor of N is identical to division of phase by the same factor. Thus, the feedback phase θ_{div} is

$$\theta_{div}(s) = \frac{\theta_{out}(s)}{N} \tag{2.16}$$

The open-loop transfer function can be expressed as follows:

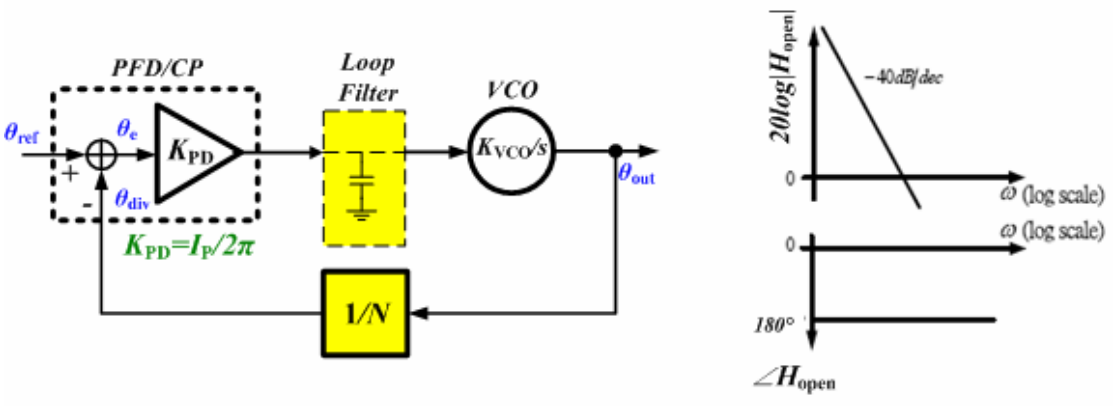
$$H(s)\big|_{open} = A(s)\beta(s) = K_{PD} \cdot Z(s) \cdot \frac{2\pi K_{VCO}}{s} \cdot \frac{1}{N} \tag{2.17}$$

As with other feedback systems, ***A(s)*** is the forward-loop gain and $\beta(s)$ is the reverse-loop gain. This open-loop transfer function determines the performance of the PLL, such as the loop stability.

Shown in Figure 2.21 (a) is the linear model of a CPPLL. Firstly, if only a single capacity is employed as the loop filter, Equation 2.17 becomes

$$H(s)\big|_{open} = \frac{I_P}{N}\left(\frac{1}{sC_p}\right)\frac{K_{VCO}}{s} \tag{2.18}$$

Since the loop gain has two poles at the origin, this topology is called a "*type-II second-order PLL*". As shown in Figure 2.21 (b), due to two poles at the origin, (i.e. two ideal integrators), each integrator contributes a constant phase shift of *90°* and hence the phase margin is zero at the gain crossover frequency, yielding instability.



(a) **(b)**

Figure 2.21 (a) Linear model of a 2^{nd} order CPPLL and (b) its open-loop Bode plot.

In order to stabilize the system, a zero must be introduced in the loop gain such that the phase shift is less than $180°$ at the gain crossover frequency (Figure 2.22 (b)). As shown in Figure 2.22 (a), this can be accomplished by adding a resistor R_P in series with the capacity C_P. Thus the open-loop transfer function can be rewritten as

$$H(s)\big|_{open} = \frac{I_P}{N}\left(R_P + \frac{1}{C_P s}\right)\frac{K_{VCO}}{s} = \frac{I_P K_{VCO} R_P}{N}\left(\frac{s+\omega_z}{s^2}\right) \tag{2.19}$$

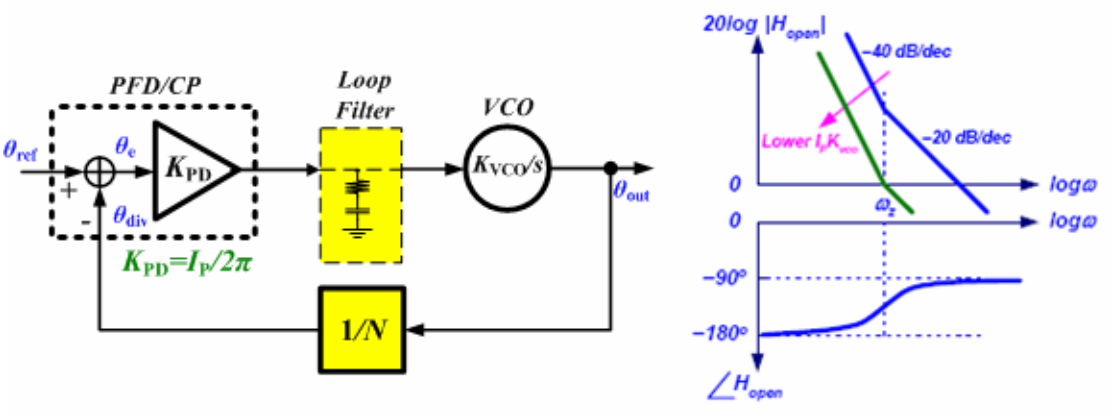
, *where a zero at* $\omega_z = -1/R_P C_P$.

The loop bandwidth K, defined as the unity gain frequency of the open-loop transfer function, can be found out assume that K is much greater than ω_z:

$$H(s)\big|_{open} = \frac{I_P K_{VCO} R_P}{N}\left(\frac{s+\omega_z}{s^2}\right) \approx \frac{I_P K_{VCO} R_P}{N}\left(\frac{1}{s}\right) = 1$$

$$\Rightarrow K = \frac{I_P K_{VCO} R_P}{N} \tag{2.20}$$

As the Bode plot indicates, if $I_P K_{VCO}$ increases, the gain crossover frequency (namely, the loop bandwidth K) moves away from the origin, enhancing the phase margin. But since the PFD/CP works in discrete time domain, the loop bandwidth should be considerably lower than the reference frequency according to *Gardner's stability limit* [1]. In other words, the loop gain should be limited to avoid instability.



(a) **(b)**

Figure 2.22 (a) Linear model of a compensated 2^{nd} order CPPLL and (b) its open-loop Bode plot.

Besides, this *compensated type-II second-order PLL* suffers from a critical drawback. Since the charge pump drives the series combination of R_P and C_P, each time a current is injected into the loop filter, the control voltage experiences a large jump of $I_P R_P$, as shown in Figure 2.23. So, the PLL suffers from serious "granularity effect", which can be seen easily in transient response (Figure 2.23). A possibly more serious problem introduced by the jumps is the potential for overload of the VCO. More specifically, any real VCO has only a finite frequency range over which it can be tuned. If this voltage jump exceeds the valid input control range of VCO, it leads to the failure of the operation of the overall charge pump PLL.

Additionally, any possibly mismatches in the CP introduce voltage jumps in V_{ctrl} even in the locked status. The resulting ripple severely disturbs the VCO, which causes large spurs. To alleviate these issues, a second capacity is added in shunt with the first-order filter. Now, the loop filter is of second-order, yielding a third-order PLL, which is adopted in our work.

Shown in Figure 2.24 is the *type-II third-order PLL* that alleviate the voltage-jump-induced problems stated above. The transfer function of the loop filter now becomes

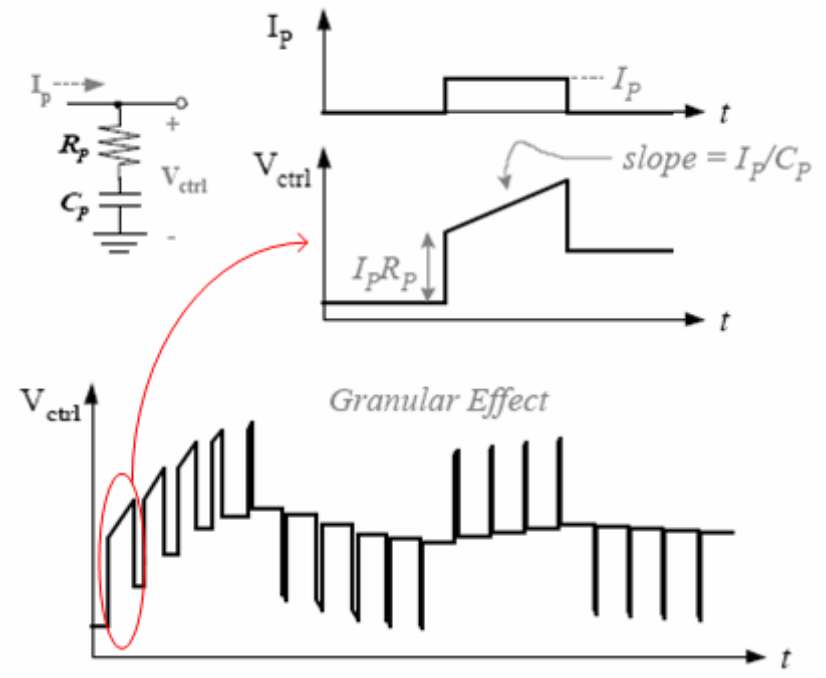


Figure 2.23 Granular transient response of a PLL with first-order loop filter

$$Z(s)=\frac{C_P}{C_2+C_P}\cdot\frac{sR_PC_P+1}{sC_P\left(\frac{sRC_P}{C_P/C_2+1}+1\right)}=K_f\cdot\frac{s+\omega_z}{s\left(s/\omega_p+1\right)} \tag{2.21}$$

, *where* $\omega_z=\frac{1}{R_PC_P}$, $\omega_p=\frac{C_2+C_P}{C_2C_PR_P}$, $K_f=\frac{C_PR_P}{C_P+C_2}$

Note that a new pole ω_p is introduced with its frequency higher than the zero ω_z according to the following formula:

$$\omega_p=\omega_z\left(\frac{C_P}{C_2}+1\right) \tag{2.22}$$

Substitute Equation 2.20 into Equation 2.17 and then the open-loop transfer function can be written as

$$H(s)\big|_{open}=\frac{I_P}{N}Z(s)\frac{K_{VCO}}{s}=\frac{I_PK_{VCO}K_f}{N}\cdot\frac{s+\omega_z}{s^2\left(s/\omega_p+1\right)} \tag{2.23}$$

The phase margin is then given by

$$PM=\tan^{-1}(\omega/\omega_z)-\tan^{-1}(\omega/\omega_p) \tag{2.24}$$

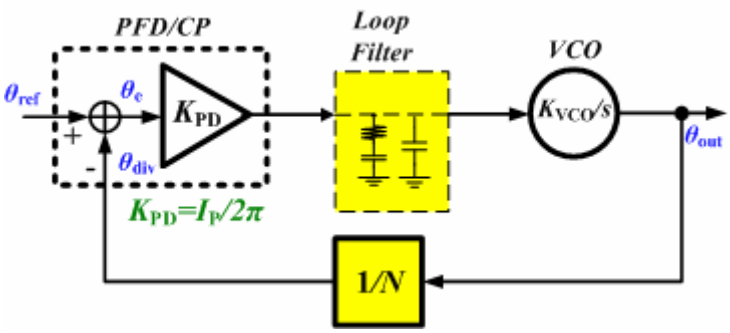


Figure 2.24 Linear model of a *type-II 3rd order* CPPLL

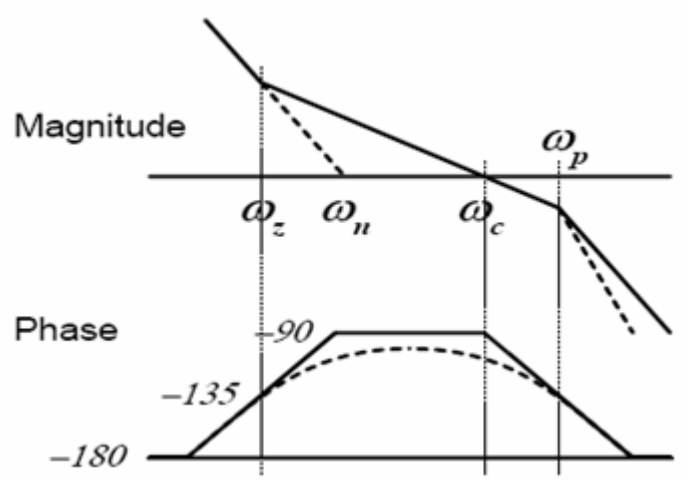


Figure 2.25 Open-loop Bode plot of *type-II 3rd order* CPPLL

The open-loop Bode plot is shown in Figure 2. 25. The phase of $H(s)|_{open}$ is *-180°* at dc. The zero ω_z and the pole ω_p introduce phase shifts of *+90* ° and *-90°*, respectively. Thus, it is essential to place the gain crossover frequency *K* between ω_z and ω_p for enough phase margin in consideration of stability. To find out the value of *K* that satisfies the optimal phase margin, equaling the derivative of the phase margin to zero gives

$$\frac{d}{d\omega}\left(\tan^{-1}(\omega/\omega_z)-\tan^{-1}(\omega/\omega_p)\right)=\frac{\omega_z}{\omega^2+{\omega_z}^2}-\frac{\omega_p}{\omega^2+{\omega_p}^2}=0$$

$$\Rightarrow \omega_{opt}=\sqrt{\omega_z\omega_p}=K_{opt} \qquad (2.25)$$

That is, if the loop bandwidth is set to the geometric average of ω_z and ω_p, the phase margin will be maximal.

Furthermore, we defines a new parameter γ as

$$\gamma = \frac{K}{\omega_z} = \frac{\omega_p}{K} \tag{2.26}$$

Table 2.1shows a useful reference table of the relationship between γ and *PM*. From Equation 2.21, the capacitance ratio of C_P and C_2 can be represented by

$$\frac{C_P}{C_2} = \gamma^2 - 1 \tag{2.27}$$

Assume the loop bandwidth K is much greater than ω_z but much smaller than ω_p. K can be found out:

$$H(s)\Big|_{open} = \frac{I_P K_{VCO} K_f}{N} \frac{s+\omega_z}{s^2\left(s/\omega_p + 1\right)} \approx \frac{I_P K_{VCO} K_f}{N}\left(\frac{s}{s^2}\right) = 1$$

$$\Rightarrow K = \frac{I_P K_{VCO} K_f}{N} = \frac{I_P K_{VCO} R_P}{N} \frac{C_P}{C_P + C_2} \tag{2.28}$$

Table 2.1

Relationship between γ and *PM*

γ	*PM*
1	*0°*
2	*36.9°*
3	*53.1°*
4	*61.9°*
5	*67.4°*

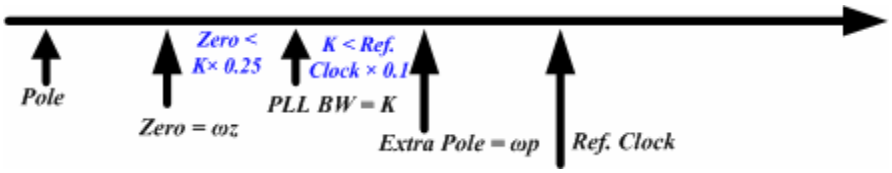


Figure 2.26 The interrelation between each pole and zero

One should note that, for second-order or higher CPPLL designs, the choice of the loop bandwidth K should treat the trade-offs between stability and spur rejection under *Gardner's stability limit* [1]. Illustration of Figure 2.26 can provide good insight into the interrelation between each pole and zero [14].

A design flow of a type-II third-order CPPLL can be summarized as follows:

(1) Determine $\boldsymbol{K_{VCO}}$: the VCO gain or sensitivity can be found from simulation results or experiment results.

(2) Determine $\boldsymbol{K}$: the loop bandwidth is then determined depending on the demand noise and transient characteristic. (In general, K is at least 10 times less than $f_{ref.}$)

(3) Determine $\boldsymbol{I_P}$: choose an appropriate pump current in consideration of trade-off between chip area and power consumption if fully integrated into single chip.

(4) Determine $\boldsymbol{PM}$: according to Table 2.1, select a value of γ to meet the required PM specification. The ω_z and ω_p can also be determined by Equation 2.26.

(5) Calculate $\boldsymbol{R_P}$: with K_{VCO}, K, I_P, and M determined, R_P can be obtained by substituting Equation 2.27 into Equation 2.28.

(6) Calculate $\boldsymbol{C_P}$ and $\boldsymbol{C_2}$: from (4), C_P can be calculated and then C_2 by Equation 2.27.

2.4.2 Static Phase Error Analysis

As discussed earlier, a PLL is a negative feedback control system, which only responds to variations in the excess phase of the input or output. Once the PLL experiences frequency or phase variations (i.e. a discontinuous step in modulus), the loop starts to replicate and track the frequency and phase at the input until lock is achieved. The loop is considered "locked" if the phase error θ_e is constant with time and preferably small. That is,

$$\frac{d\theta_e(t)}{dt} = \frac{d\theta_{ref}(t)}{dt} - \frac{d\theta_{div}(t)}{dt} = 0 \tag{2.28}$$

, and hence

$$\omega_{div} = \frac{\omega_{out}}{N} = \omega_{ref} \Rightarrow \omega_{out} = N\omega_{ref} \tag{2.29}$$

In summary, when locked, the loop operates in the steady state, exhibiting no transient but a possibly small static phase error at the PFD output. In evaluating a PLL, θ_e must be examined in order to determine if the steady state and transient characteristics are optimum and/or satisfactory. The steady state evaluation can be simplified with the use of "the final value theorem" associated with Laplace. That is,

$$\lim_{t\to\infty} \theta_e(t) = \lim_{s\to 0} s\theta_e(s) \tag{2.30}$$

Ideally, if neglecting mismatches and offsets, the static phase error is expected equal to zero no matter what type of transient phase signal is inputted. Various inputs, including phase step, frequency step, and frequency ramp, will be applied to examine the response of type-I, II, and III PLLs. As illustrated in Figure 2.20, the closed-loop transfer function can be expressed as follows:

$$H(s)\big|_{close} = \frac{\theta_{out}(s)}{\theta_{ref}(s)} = \frac{A(s)}{1 + A(s)\beta(s)} = \frac{2\pi K_{VCO}K_{PD}Z(s)}{s + \frac{2\pi K_{VCO}K_{PD}Z(s)}{N}} \tag{2.31}$$

Then, the phase error transfer function can be derived as follows:

$$\theta_e(s) = \theta_{ref}(s) - \frac{\theta_{out}(s)}{N}$$

$$\Rightarrow \frac{\theta_e(s)}{\theta_{ref}(s)} = \left(1 - \frac{H(s)\big|_{close}}{N}\right) = \frac{1}{1 + \frac{2\pi K_{VCO}K_{PD}Z(s)}{sN}} \tag{2.32}$$

According to "the final value theorem", the static phase error can be obtained as follows:

$$\lim_{t\to\infty} \theta_e(t) = \lim_{s\to 0} s\cdot\theta_e(s) = \lim_{s\to 0} \frac{s^2 N\theta_{ref}(s)}{s + 2\pi K_{VCO}K_{PD}Z(s)} \tag{2.33}$$

However, before further examining the response of type-I, II, and III PLLs, we recall how to specify the type and order of PLLs. Typically, if the loop filter *Z(s)* is of n^{nd}-order with *m* poles at the origin, the PLL can be identified as a type-*(m+1)* $(n+1)^{nd}$-order PLL. The transfer function of the loop filter can be rewritten as

$$Z(s) = \frac{1}{s^m} F(s) \tag{2.34}$$

, where *F(s)* is of *(n-m)*-order without any pole at the origin.

Three different transient phase inputs θ_{ref} is characterized as follows:

Phase step: $\theta_{ref}(t) = \Delta\theta \cdot u(t)$

Or, in Laplace domain: $\theta_{ref}(s) = \frac{\Delta\theta}{s}$

Frequency step: $\theta_{ref}(t) = \Delta\omega t \cdot u(t)$

Or, in Laplace domain: $\theta_{ref}(s) = \frac{\Delta\omega}{s^2}$

Frequency ramp: $\theta_{ref}(t) = at^2 \cdot u(t)$

Or, in Laplace domain: $\theta_{ref}(s) = \frac{2a}{s^3}$

In conclusion, applying the three inputs into type-I, II, and III PLLs and utilizing Equation 2.33, the following Table 2.2 summarizes the results to show the respective static phase errors. Thus, due to the requirement of frequency step-switching, type-II or type-III PLL would be beneficial for arriving at zero static phase error.

Table 2.2

Static phase error for various types of PLLs

	Type I	Type II	Type III
Phase step	*Zero*	*Zero*	*Zero*
Frequency step	*Constant*	*Zero*	*Zero*
Frequency ramp	*Continually increasing*	*Constant*	*Zero*

2.4.3 Phase Noise Performance Analysis

A PLL-based frequency synthesizer suffers from noise introduced at the input or generated by the other components, such as PFD, CP, VCO, loop filter, and frequency divider, etc. It is important to get insight into how these noise sources affect the overall noise performance of synthesized output signal. These noise sources may be classified into two main types to sum up: one is the noise of VCO and the other is the noise from other sources. The effect caused by each of these noise sources can be seen from the closed-loop transfer functions.

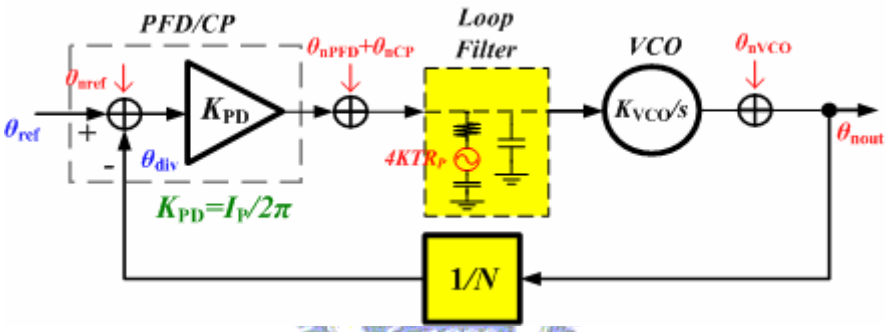


Figure 2.27 Linear model of the type-II 3rd PLL with noise sources

Figure 2.27 illustrates an analytic linear model of the type-II third-order PLL with noise sources added. Based on this linear model, the transfer function from each noise source to the output can be derived so that we can quantify how much each noise source contributes to the output signal. These noise sources can be characterized as follows:

φ_{nPFD}: the noise arises from timing jitter caused by additive noise, predominantly thermal within PFD.

φ_{nCP}: the noise comes from thermal and/or flicker noise of each transistors in each current source. (i.e. $\propto \frac{\overline{I_n^2}}{\Delta f} = 4KT\gamma g_m$)

φ_{nRp}: the equivalent thermal noise of resister R_P inside the loop filter. (i.e.

$\propto 4KTR_P$)

φ_{nVCO} and φ_{nref}: the noise sources from VCO and the crystal reference oscillator, respectively.

And then the transfer functions for these noise sources can be derived as

$$H(s)\big|_{\varphi_{nref}} = \frac{\varphi_{nout}(s)}{\varphi_{nref}(s)} = \frac{2\pi K_{VCO}K_{PD}Z(s)}{s + \frac{2\pi K_{VCO}K_{PD}Z(s)}{N}}$$

$$H(s)\big|_{\varphi_{nVCO}} = \frac{\varphi_{nout}(s)}{\varphi_{nVCO}(s)} = \frac{1}{1 + \frac{2\pi K_{VCO}K_{PD}Z(s)}{sN}}$$

$$H(s)\big|_{\varphi_{nCP}} = \frac{\varphi_{nout}(s)}{\varphi_{nCP}(s)} = \frac{2\pi K_{VCO}Z(s)}{s + \frac{2\pi K_{VCO}K_{PD}Z(s)}{N}} = H(s)\big|_{\varphi_{nPFD}} = \frac{\varphi_{nout}(s)}{\varphi_{nCP}(s)}$$

$$H(s)\big|_{\varphi_{nR_P}} = \frac{\varphi_{nout}(s)}{\varphi_{nR_P}(s)} = \left(\frac{1}{R_P + \frac{1}{sC_P}}\right)\frac{2\pi K_{VCO}Z(s)}{s + \frac{2\pi K_{VCO}K_{PD}Z(s)}{N}}$$

The overall output phase noise φ_{nout} contributed by each noise source can be expressed as

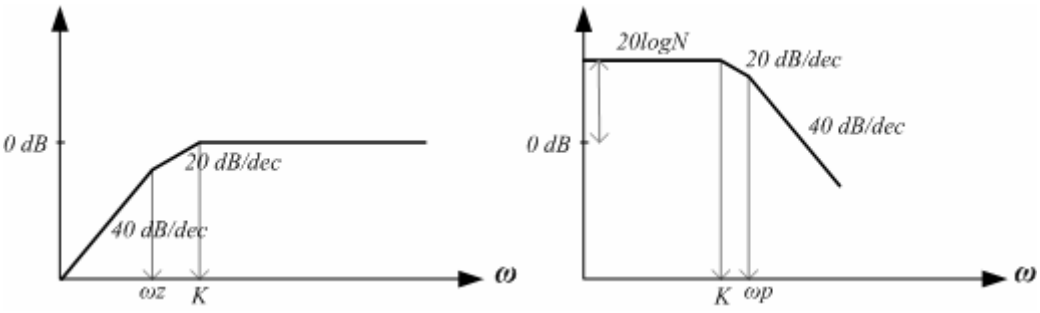
$$H(s)\big|_{\varphi_{nout}} = \left(\varphi_{nref} + \varphi_{neq}\right)\left(\frac{2\pi K_{VCO}K_{PD}Z(s)}{s + \frac{2\pi K_{VCO}K_{PD}Z(s)}{N}}\right) + \varphi_{nVCO}\left(\frac{1}{1 + \frac{2\pi K_{VCO}K_{PD}Z(s)}{sN}}\right),$$

where $\varphi_{neq} = \left(\varphi_{nCP} + \varphi_{nPFD} + \varphi_{nR_P}\left(\frac{1}{R_P + 1/sC_p}\right)\right)/K_{PD}$

The first term of φ_{nout} is a low pass term and the second is a high pass term. At low frequencies (i.e. *Z(s)* has infinite DC gain, $s \to 0$), $\varphi_{nout} \approx (\varphi_{nref} + \varphi_{neq}) \cdot N$, which implies that the noise contribution mainly comes from the reference oscillator, frequency divider, and PFD/CP. At high frequency (i.e. $Z(s) \approx 1$, $s \to \infty$), $\varphi_{nout} \approx \varphi_{nVCO}$, which reveals that the main noise contribution comes from the VCO phase noise.

In summary, the VCO phase noise experiences a high-pass filter characteristic as it

appears at the output of a PLL, whereas the phase noise at the PLL input undergoes a low-pass filter characteristic with an amplification factor of N, as shown in Figure 2.28. Note that the transfer function from the thermal noise of R_P to φ_{nout} has a band-pass characteristic. It is desirable to reduce the value of R_P and increase the value of C_P at the cost of large chip area for on-chip integration.



(a) **(b)**

Figure 2.28 PLL noise transfer functions (a) from VCO noise; (b) from Ref. noise

However, in the design of PLLs, the noise sources from other than the reference oscillator and VCO are usually relatively insignificant and hence negligible. In other words, the overall phase noise performance is mostly dominated by the reference oscillator and VCO. As discussed above, the loop bandwidth should be as wide as possible in order to minimize the output phase noise caused by the VCO inherent phase noise φ_{nVCO}. However, in order to achieve a minimum in-band output noise contributed by the reference noise φ_{nref}, the loop bandwidth should be as narrow as possible. Obviously, there exits a trade-off regarding the choice of the loop bandwidth. Therefore, to attain a minimal phase noise performance from the reference noise and VCO inherent noise, the best choice is to set the loop bandwidth to the point where the VCO phase noise intersects the reference phase noise times *N*, as illustrated in Figure 2.29.

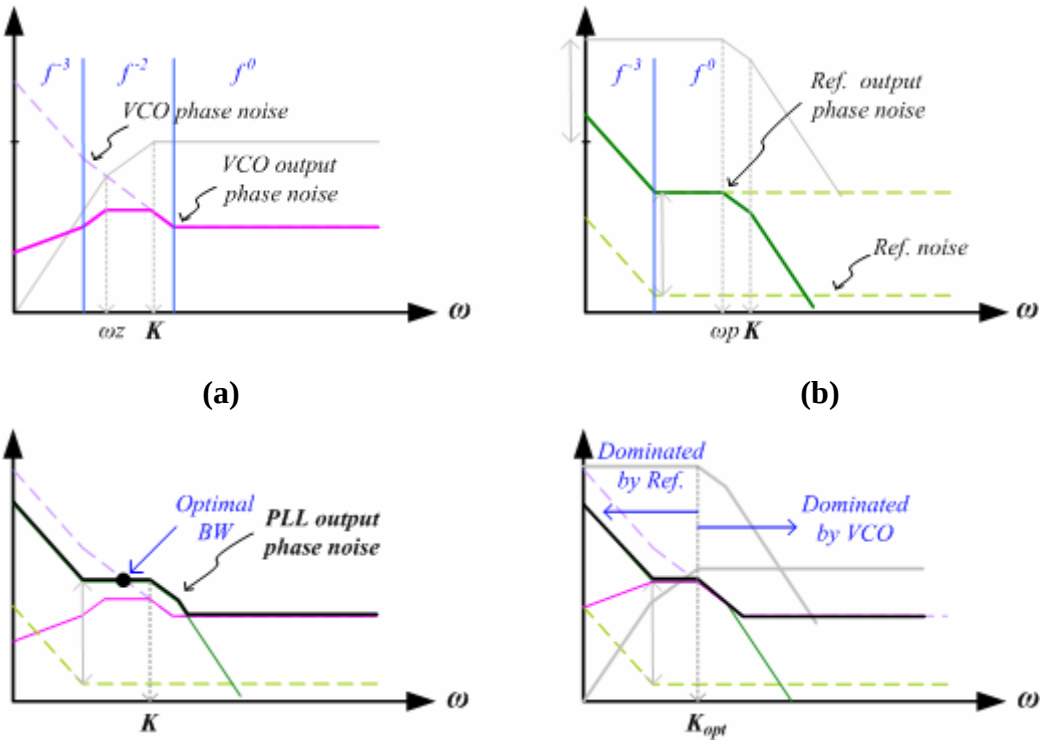


Figure 2.29 Phase noise contributions in a PLL (a) VCO output phase noise; (b) Ref. output phase noise; (c) and (d) PLL output phase noise (before and after optimizing the loop bandwidth, respectively)

CHAPTER 3

Fast Frequency Acquisition Phase-Frequency Detectors Design

3.1 Introduction

Phase-locked loops (PLLs) are widely used in frequency acquisition loops for data recovery circuits, and as frequency synthesizers for wireless communication systems. The design of integrated PLLs still remains one of the most challenging and time-consuming tasks in communication systems. In order to meet the stringent requirements of emerging communication technologies, low jitter, fast settling and low power are some of the most crucial and in-demand aspects in PLL designs, which involve a lot of design issues and trade-offs.

Within different PLL topologies, a common architecture is the charge pump PLLs (CPPLLs), in which a phase-frequency detector (PFD) is incorporated to monitor the input reference signal and the divided VCO output signal, and produces an output signal proportional to the phase and frequency differences between them. Initially, the PFD operates as a frequency discriminator for large frequency errors, and finally as a coherent phase detector to achieve loop lock once inside the pull-in range of the PLLs. As a result, a fast frequency acquisition can be achieved. However, the design of the PFD involves several design issues, such as faster operating frequency, dead-zone/ blind-zone problems, and locking time of the PLLs. In particular, applications requiring low jitter increase the difficulty of the design of the PFD because they prefer a high reference frequency and minimum division ratio.

A conventional tri-state PFD suffers from a phenomenon known as the “dead zone”, which arises from the fact that the intrinsic reset pulse is too short to allow the PFD to drive the charge pump fully into ON state at a very small phase difference. Thus, any phase error within the dead zone cannot be detected, and then directly translates to phase jitter in the PLL and should be avoided. A typical solution to this problem is to insert an added delay in the reset path of the PFD to maintain a minimum pulse width, allowing a full switching of the following charge pump. Nevertheless, unfortunately such a solution introduces another issue known as the “blind zone”, which limits the maximum operating frequency [15] and narrows the available linear phase input range. For large phase errors, the differential output of the PFD presents the wrong polarity, hence yielding longer acquisition times. The blind-zone effect will become even more serious as the tendency for high reference frequencies is further pursued.

In this chapter, an alternative architecture of a tri-state phase-frequency detector is proposed to extend the linear phase input range to would be expected and achieve much better frequency acquisition capabilities. Section 3.2 briefly describes the conventional tri-state PFD architecture and addresses its inherent constraints. Section 3.3 presents the proposed PFD architecture, and elaborates upon its phase- and frequency-discriminator characteristics. In Section 3.4, the simulation results are provided, including some comparison with the results of other recently-published related literature [5]-[6], [8]. Finally, some conclusions are drawn in Section 3.5.

3.2 Conventional tri-state PFD architecture

Figure 3.1 shows the schematic circuit and state diagram of a conventional tri-state PFD, which consists of two edge-triggered, resettable D-type flip-flops (DFFs), a delay element, and an AND gate. Note that the use of the edge-triggered DFFs is to avoid the dependence of the output upon the duty cycle of the inputs.

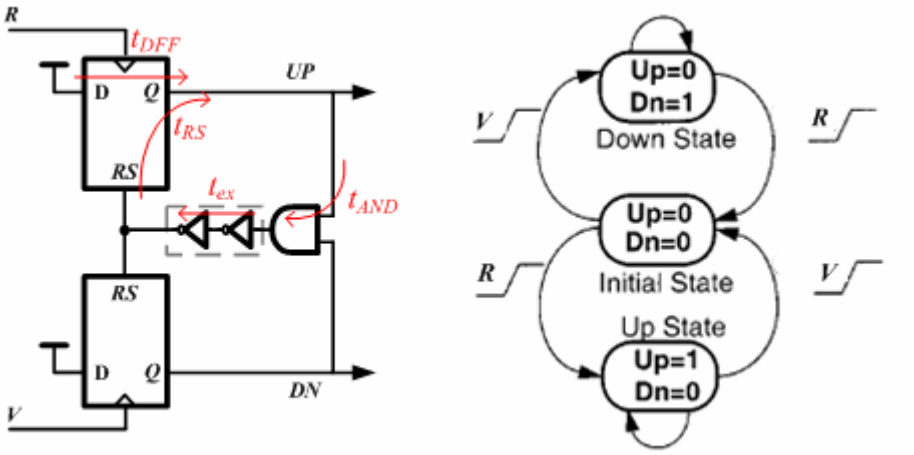


Figure 3.1 A conventional tri-state PFD and its state diagram

The D inputs of the flip-flops are tied to logic one. Signal *R* and *V* act as the clock inputs of the two DFFs, respectively. A positive transition in the input *R* sets the output *UP* "high". Similarly, a positive transition in the input *V* sets the output *DN* "high". When both *UP* and *DN* are simultaneously "high", a reset pulse generated by the AND gate resets both flip-flops, which brings *UP* and *DN* to low and then terminates the reset pulse. In order to eliminate the dead-zone, a delay element has been introduced in the reset path, so that minimum width pulses are always present in the outputs *UP* and *DN* when the inputs are in phase. The minimum duration t_{DZ} of the output pulses has to allow a full switching of the following charge pump, and is determined by the reset delay of the flip-flops t_{RS}, AND-gate delay t_{AND}, and an added delay t_{ex} given by the delay element. The minimum duration t_{DZ} is expressed as $t_{DZ} = t_{RS} + t_{AND} + t_{ex}$.

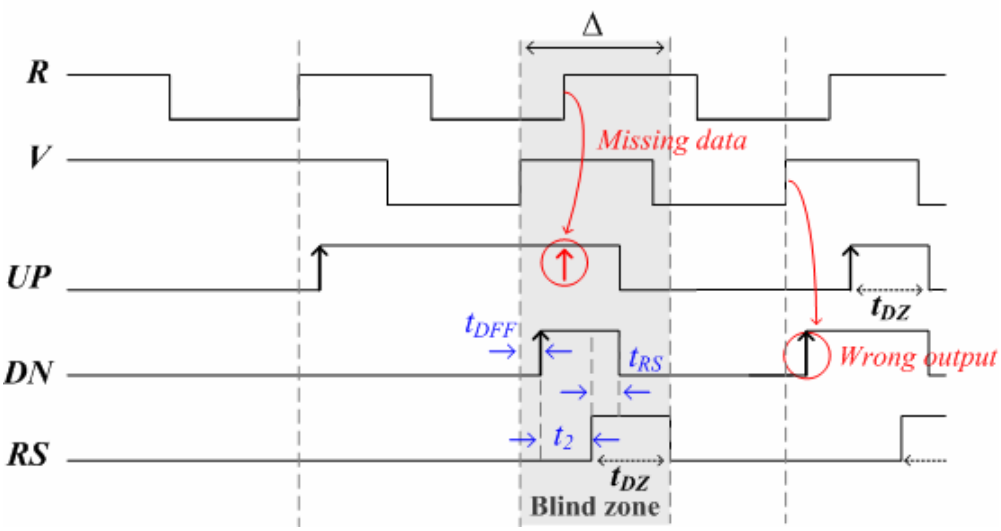


Figure 3.2 Timing diagram of a tri-state PFD, presenting its non-ideal behavior.

Figure 3.2 illustrates a timing diagram of a tri-state PFD, which presents its non-ideal behavior. The circuit operation can be described as follows. Initially, the flip-flops are in the reset status, and the *UP* and *DN* are logic low. A first leading rising-edge of *R* causes *UP* to transit to "high" and this status is held until the following rising-edge of *V* arrives, which in turn activates the AND gate to output a reset pulse. Afterward, both *UP* and *DN* will still remain logic "high" for duration of t_{DZ} until the end of resetting both flip-flops. When the *UP* and *DN* are returned to logic low, the reset pulse is terminated and the circuit is really for next cycle. It turns out that the PFD exhibits a blind zone, which starts from the lagging rising-edge of *V* and ends with the falling-edge of the reset signal *RS*. As the phase error approaches 2π, any next leading rising-edge of *R* arriving inside the blind zone will be overridden. In a subsequent cycle, the following rising-edge of *V* causes a leading *DN* signal. The effect appears as a negative output for phase differences higher than $2\pi - \Delta$, where $\Delta = 2\pi \cdot t_{BZ} / T_{ref}$. Thus, the linear phase input range is less than would be expected, as highlighted in Figure 3.3. Note that the duration of the blind zone can be expressed as $t_{BZ} = t_{DFF} + t_{DZ} + t_{AND} + t_{ex}$, where t_{DFF} represents the propagation delay of the DFFs and t_2 denotes $t_{AND} + t_{ex}$.

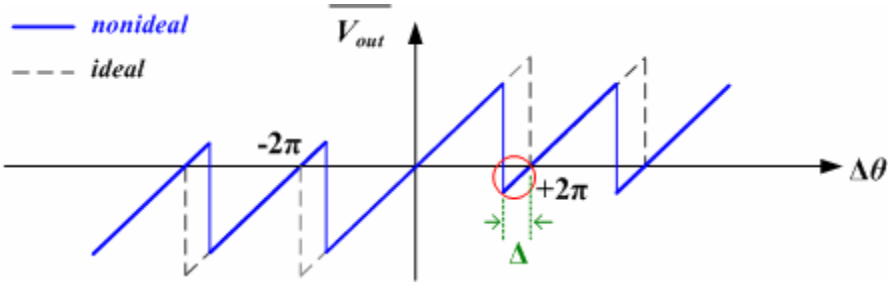


Figure 3.3 Phase characteristic of the ideal (*dashed line*)/ non-ideal (*solid line*) PFD

During acquisition, the frequency will not monotonically approach lock-in range because the nonideal PFD gives the wrong information periodically [6]. The acquisition slows by how often the wrong information occurs, which depends on the proportion of t_{BZ} and T_{ref}. Thus, as the tendency is toward increasing the reference frequency, this phenomenon will become more significant, which poses a limit on the maximum operating frequency to $f_{ref} \leq 1 / 2 \cdot t_{BZ}$ [6][15].

3.3 Proposed PFD architecture

As discussed above, in reality, a conventional tri-state PFD experiences the output polarity reversal for large phase errors due to “blind zone”, yielding longer acquisition times. In order to address this problem, we propose a novel PFD architecture which keeps track of input transitions occurring in the blind zone, thereby avoiding wrong output information and achieving faster frequency acquisition.

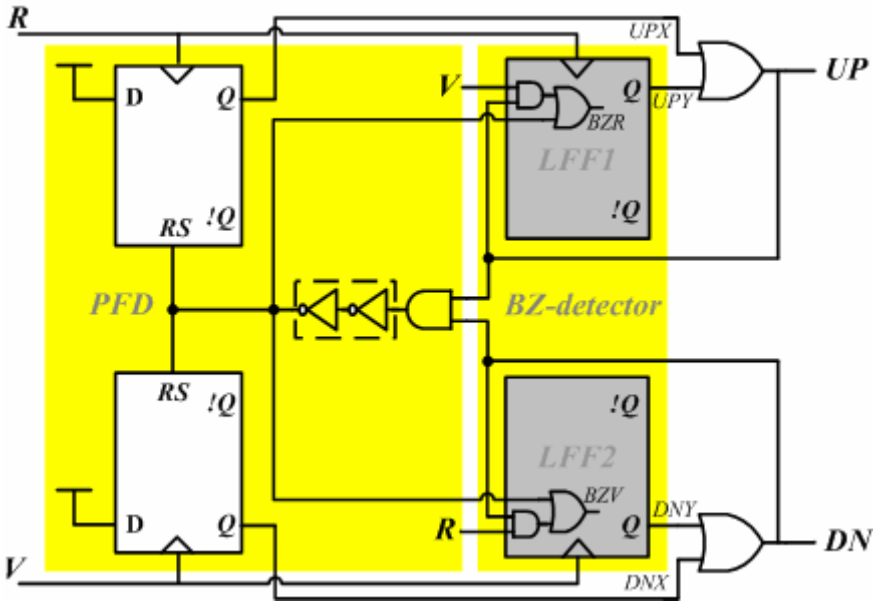


Figure 3.4 Proposed tri-state PFD architecture

Figure 3.4 shows the proposed tri-state PFD architecture, which consists of two parts: one is the "blind-zone detector" (*BZ-detector)* for aided acquisition and the other is the classical "dead-zone-free tri-state PFD" for normal operation. The BZ-detector, which comprises of several logic gates and two flip-flops, serves as an auxiliary circuit to aid the acquisition once the phase difference $\Delta\theta$ is inside the blind zone of $2\pi-\Delta < \Delta\theta < 2\pi$. The signals from the PFD core and BZ-detector logically combines to form the *UP* or *DN* signal. Figure 3.5 shows the corresponding timing diagram illustrating the operation of the proposed PFD architecture. When an input transition falls in the blind zone, the BZ-detector is set to track the event. In turn, suitable signals *BZR* (*BZV*) are sampled by the input transitions, setting the output *UPY* (*DNY*) high so that the input edge information propagations to the *UP* (*DN*). After that, the *UP* signal remains logic high until the phase difference is out of the blind-zone range. In this manner, the PFD no longer loses the input edge that arrives during blind zone and don't output the wrong information, so resulting in reduce acquisition times. Note that the signals *BZR* and *BZV* are generated by "$BZR = (V \cdot UP) + RS$" and "$BZV = (R \cdot DN) + RS$", respectively. In order

to eliminate the logic gate delay, the logic function can be merged into the flip-flops, forming logic flip-flop (LFF) pipeline stages. As a result, the linear phase input range can be fully extended to ±2π.

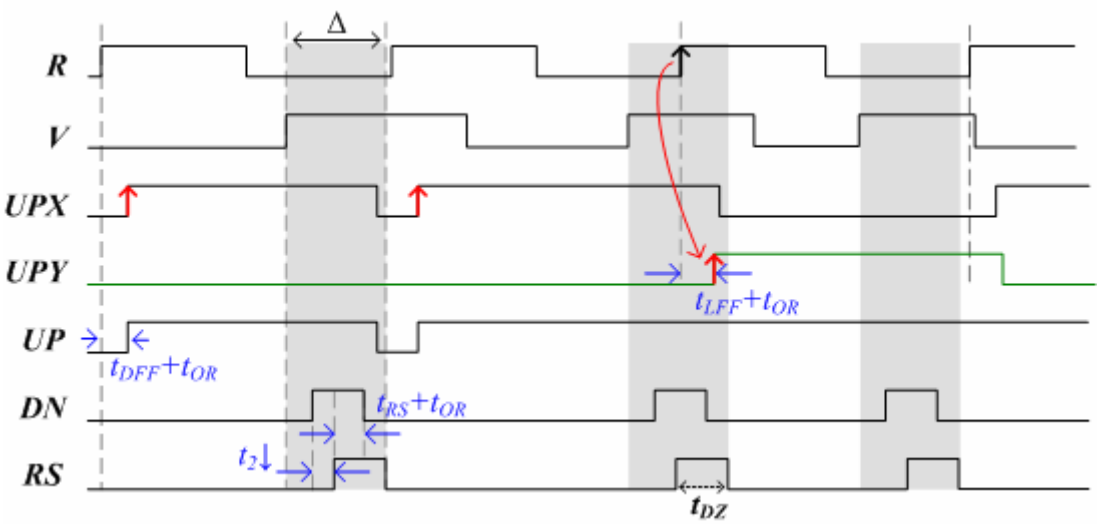


Figure 3.5 Timing diagram of the proposed tri-state PFD

Figure 3.6 illustrates the phase-discriminator characteristic of the proposed PFD, compared to those of the ideal and real PFDs. For phase errors small than $2\pi(1-t_{BZ}/T_{ref})$, no input transition falls in the blind zone, and the circuit behaves like an ideal PFD, with the average output proportional to the phase error. For $2\pi\ (1-t_{BZ}/T_{ref}) < \Delta\theta < 2\pi\ [1-(t_{BZ}-t_{LFF}-t_{OR})/T_{ref}]$, the proposed PFD still follows the linear slope. Finally, the PFD exhibits a flat gain for phase errors larger than $2\pi\ [1-(t_{BZ}-t_{LFF}-t_{OR})/T_{ref}]$.

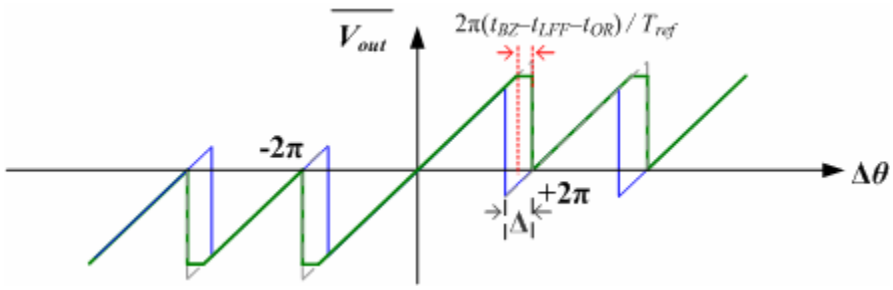


Figure 3.6 Phase-discriminator characteristic of the proposed PFD

Moreover, to evaluate a PFD, its frequency-discriminator characteristic is also of great concern in addition to phase-detection characteristic. Before further examining our proposed PFD, we recall a statistical method which is proposed in [15]. In [15], analytical expressions that correctly predict the high-frequency behavior of the PFD have been derived and demonstrated. Thus, a similar statistical analysis can be made for the frequency-detection characteristic of our proposed PFD. Firstly, let us recall and follow the approach specified in [15]. Assume that the two input frequencies, f_R and f_V, are not equal and $\alpha = f_V / f_R$. Then,

If f_R is greater than f_V and look at time interval $[t, t+T_R]$ between the two successive *R* transitions, let us define the following probabilities:

$P(0)$ = probability of no V transition in $[t, t+T_R] = 1-\alpha$ (3.1)

$P(1)$ = probability of a single V transition in $[t, t+T_R] = \alpha$ (3.2)

$P(1)|_{R\uparrow}$ = probability of the *R* transition at time *t*, setting the output *UP* high

$= 1- t_{BZ}/T_V$ (3.3)

$P(0)|_{R\uparrow}$ = probability of the *R* transition at time *t*, not setting the output *UP* high

$= t_{BZ}/T_V$ (3.4)

As stated earlier in this chapter, $P(0)|_{R\uparrow}$ arises from the fact that the first *R* transition does not set the output *UP* high if it's within the time interval $[t_V, t_V+t_{BZ}]$, where t_{BZ} is the time at which the last *V* transition appeared.

Then, $(U–D)_{AVE}$ normalized with respect to the logic swing can be written as

$$\overline{(U-D)_{AVE}} = P(1)\big|_{R\uparrow} \times P(0) \times 1 - P(0)\big|_{R\uparrow} \times P(0) \times 0 + P(1)\big|_{R\uparrow} \times P(1) \times 0.5 - P(0)\big|_{R\uparrow} \times P(1) \times 0.5 \quad (3.5)$$

Substituting (1) ~ (4) into (5):

$$\overline{(U-D)_{AVE}} = (1-0.5\alpha) - \frac{t_{BZ}}{T_V} = \frac{\beta+0.5}{\beta+1.0} - \frac{t_{BZ}}{T_V} \quad (3.6)$$

A similar analysis for $f_V \rangle f_R$ yields

$$\overline{(U-D)_{AVE}} = -\frac{\gamma+0.5}{\gamma+1.0}+\frac{t_{BZ}}{T_V} \tag{3.7}$$

Note that the above analysis is based upon the study of a conventional PFD with the blind zone problem. However, for the proposed PFD, $P(0)|_{R\uparrow}$ never occurs because the blind zone has been removed. In other words, $P(1)|_{R\uparrow} = 1$ and $P(0)|_{R\uparrow} = 0$. Thus, Equation (3.6) and (3.7) would be easily rewritten as:

$$\overline{(U-D)_{AVE}} = (1-0.506\alpha) = \frac{\beta+0.494}{\beta+1.0} \tag{3.8}$$

$$\overline{(U-D)_{AVE}} = -\frac{\gamma+0.494}{\gamma+1.0} \tag{3.9}$$

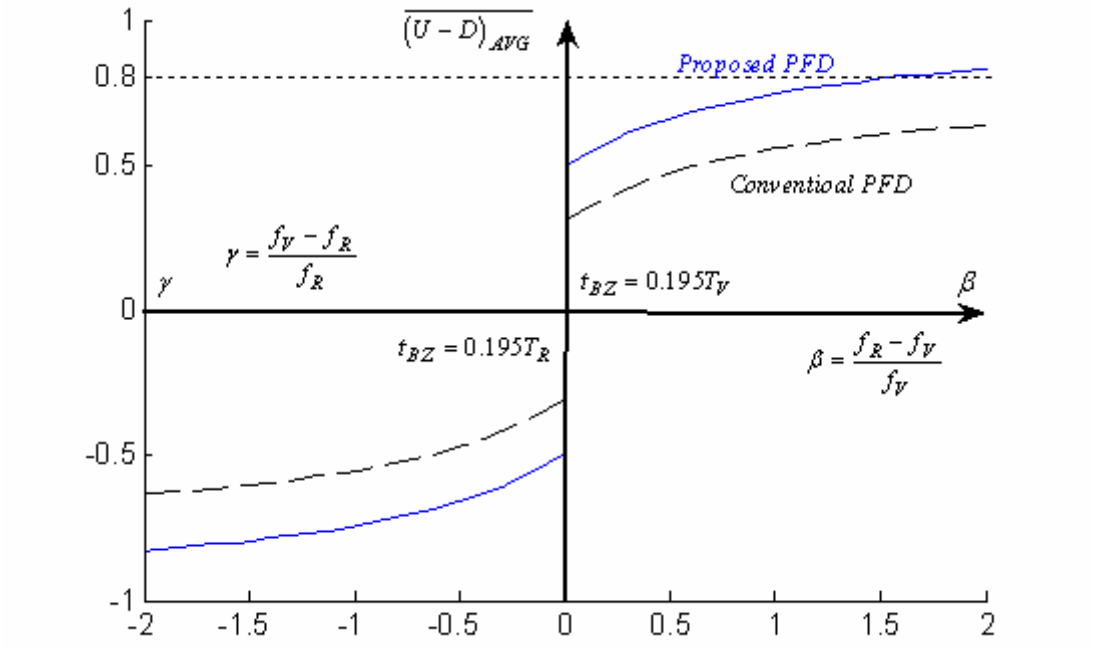


Figure 3-7 Frequency characteristics of the proposed and conventional PFDs

Figure 3.7 shows the resulting frequency-detection characteristic of the proposed PFD, compared with the convention PFD. It is clear that the proposed PFD presents a higher frequency-detector gain with respect to the conventional one, thus achieving faster frequency acquisition. ***One should note that Eq. 3.8 and 3.9 are only valid for both f_V and f_R small than $1/[2(t_{DFF} + t_{OR} + t_2)]$, which will be further discussed later.***

3.4 Circuit Design and Simulation

In order to avoid any output polarity reversal, it is required to extend the available phase input range fully to 2π. Therefore, the BZ-detector must accurately track any input transition inside the boundary timing of the blind zone. Figure 3.8 shows the implementation of *LFF*s in our proposed architecture, which is beneficial for our proposed design. Also, circuit technology evolution (e.g. TSPC ratioed DFFs) tends to prove the robustness of our proposed architecture for pursuing a high-speed operation.

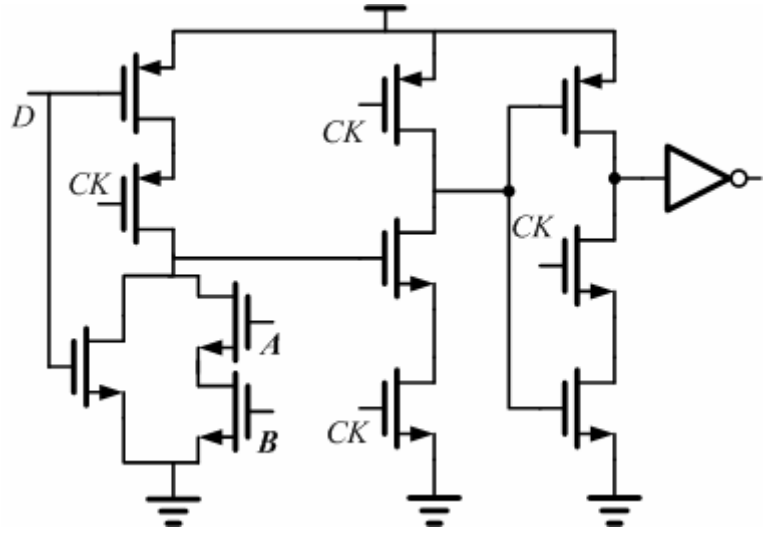


Figure 3.8 Implementation of logic flip-flops in our proposed PFD

For comparison purpose, typical propagation delay values of 0.18-μm CMOS process used by [8] are chosen and summarized in Table 3.1. With these values, the blind zone duration can be obtained from the timing diagram, as depicted in Figure 3.5. With the same t_{DZ}, the contribution of t_{OR} alleviates t_2. It turns out that t_{BZ} approximates 780*ps*, which is close to that in [8].

Analyzing the timing diagram (in Figure 3.6) allows us to examine the frequency limitations of the propose architecture. Assuming 50% duty ratio, firstly we introduce a threshold frequency, which is the maximum frequency the PFD experiences no blind

zone. In other words, the clock duration of logic "high" must be long enough to allow the reset signal *RS* to go high. Thus, $f_{thre} = 1 / [2(t_{DFF} + t_{OR} + t_2)] \approx 1.14$ GHz.

When above the threshold frequency, the PFD exhibits a blind zone δ which grows with increasing frequency until the clock period T is less than t_{BZ}. Thus, the maximum operating frequency of the proposed PFD is close to $1 / t_{BZ}$ (about 1.28 GHz). Note that, with proper design, such as higher duty ratio and/or reduced t_D, the threshold frequency can be pushed toward the maximum frequency. As a result, a fully blind-zone-free PFD can be obtained, and its resulting frequency-discriminator characteristic stated earlier can be maintained. Table 3.2 provides a comparison of the proposed architecture with some recently-published related literature.

Figure 3.9 compares the simulated frequency acquisition for the proposed and conventional PFDs, starting the VCO at 375 MHz and locking at 800 MHz. A 4 MHz bandwidth PLL with 200 MHz reference frequency is used. It can be seen that the proposed PFD shows a 25% improvement in locking time over the conventional one.

Table 3.2

Comparison of the proposed PFD with recently-published related literature

Tri-state PFD	Technology	Power Supply	f_{max}	Blind zone
[8]*	CMOS 0.18-μm	1.8V	875 MHz**	No
[6]*	CMOS 0.25-μm	1.8V	1.5 GHz	Yes, δ**
[5]	BiCMOS 0.5-μm	Not available	Not available	No
This work	CMOS 0.18-μm	1.8V	1.28 GHz	No

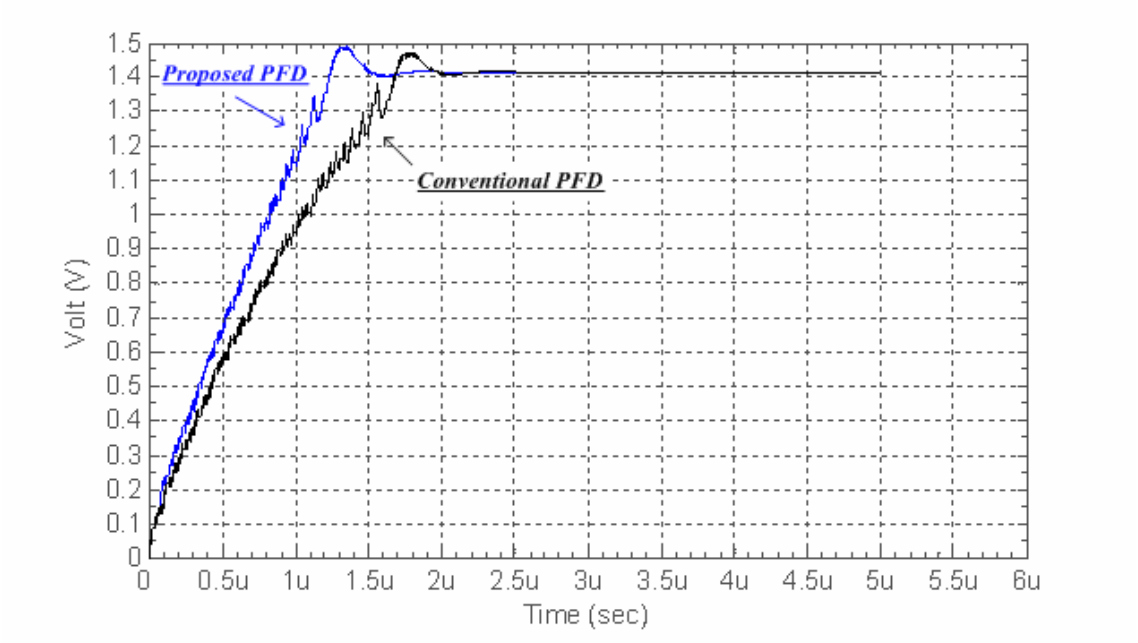


Figure 3.9 Simulated frequency acquisition for the proposed and conventional PFDs

3.5 Summary

In this chapter, we proposed a novel architecture for a tri-state PFD, which introduces an auxiliary circuit to track the phase errors inside the blind zone, thus eliminating the wrong output polarity and enhancing the frequency acquisition capabilities.

As presented in previous sections, all simulation results have demonstrated the robustness of our proposed architecture. Our proposed PFD not only solves the blind-zone problem but also achieves better phase- and frequency-discriminator characteristics. Compared with the architecture presented in [8], it also shows an excellent improvement of about 46% in the maximum operating frequency based upon the same VLSI fabrication technology in [8] (presented in Table 3.1). Additionally, as the tendency nowadays for high reference frequencies is being pursued, without the blind-zone problem, a faster frequency acquisition PLL can be achieved.

CHAPTER 4

High Speed Dual-Modulus Prescalers Design

4.1 Introduction

A high-frequency CMOS PLL frequency synthesizer has stringent requirements on dual-modulus prescaler (DMP). High speed, high moduli, and low power dissipation are the challenges in DMP designs. Typically, a DMP usually comprises of a synchronous dual-modulus counter, followed by an asynchronous counter. The critical path delay and the speed of the DFFs in the synchronous counter limit its overall speed, particularly at high divide-by-value. High divide-by-value is, in general, achieved by adding flip-flops in the asynchronous counter at the cost of additional loading to the synchronous counter which degrades the performance. Therefore, there is a trade-off between the speed and the divide-by-value.

Conventionally, most high-moduli DMPs usually comprise of a synchronous divide-by-4/5 counter, followed by a chain of toggle flip-flops, which forms an asynchronous counter. The operating speed of DMPs is mainly limited by that of the divide-by-4/5 counter. Unlike the conventional divide-by-4/5 counter [9], a new topology for a divide-by-3/4 counter using transmission gates (TGs) in the critical path for mode selection is proposed by R. S. Rana [10]. The author has demonstrated that the TG-based divide-by-3/4 counter provides higher speed compared to the conventional divide-by-4/5 counter. However, an alternative divide-by-3/4 counter using NOR gates in the critical path is also presented and taken into account for further comparison by R. S. Rana [10]. With the help of Hspice simulation, the results show that the NOR-based

divide-by-3/4 counter provides higher speed than the TG-based one due to smaller feedback path delay even though the critical path delay is more. For enhancing the speed of the TG-based divide-by-3/4 counter further, the author expects to shorten the D flip-flop (DFF) delay for future improvement.

In this chapter, a novel design for a high-speed divide-by-3/4 counter is presented. It is based on the principle of merging the flip-flop with the logic block between flip-flops to form the logic flip-flop (LFF) pipeline stage [16]-[18]. This design reduces not only the critical path delay but also the feedback path delay by sharing the delay between logic blocks and DFFs. Thus, it yields a significantly higher operating speed. In this design, a true single phase clocking (TSPC) ratioed DFF [16] is utilized.

This chapter is organized as follows. Section 4.2 describes the circuit schematics of the recently-reported counters and details the proposed design while quantifying the improvement over them. Simulation results are discussed and summarized in Section 4.3. In the end, the conclusions are provided in Section 4.4.

4.2 Circuit Topology and Principle Operation

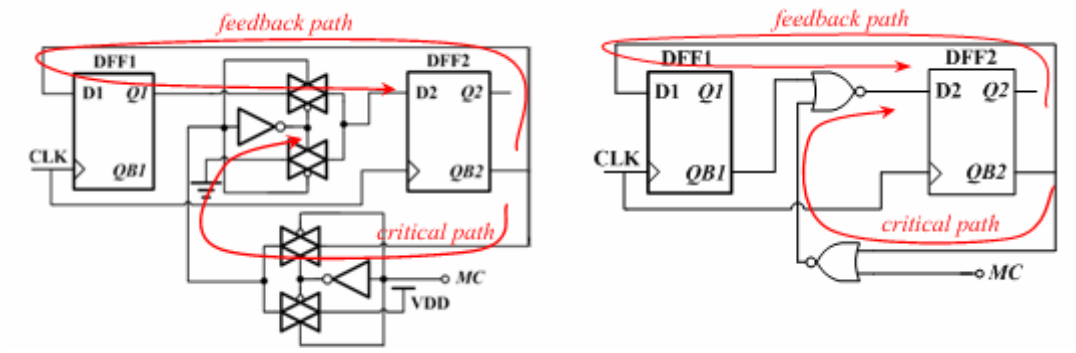


Figure 4.1 Circuit schematics: (a) TG-based divide-by-3/4 counter, and (b) NOR-based divide-by-3/4 counter

Figure 4.1 (a) and (b) show the circuit schematics of the TG-based and NOR-based divide-by-3/4 counters, respectively. As marked in the figure, it can be seen that the speed of divide-by-3/4 counter is limited primarily by two factors: 1) DFF delay and 2) logic gate delay in the critical and feedback paths. Consequently, it is believed that the speed of DMP can be further improved if the logic gate delay and/or DFF delay can be reduced. Figure 4.2 shows the proposed circuit topology of a divide-by-3/4 counter. From the figure, the proposed topology uses only a 2-to-1 multiplexer in the critical path for mode selection and adopts a ratioed-NAND structure in the counter. Table 4.1 (a) and (b) give an insight of state transitions during modulus division. The principle of circuit operation can be illustrated as follows.

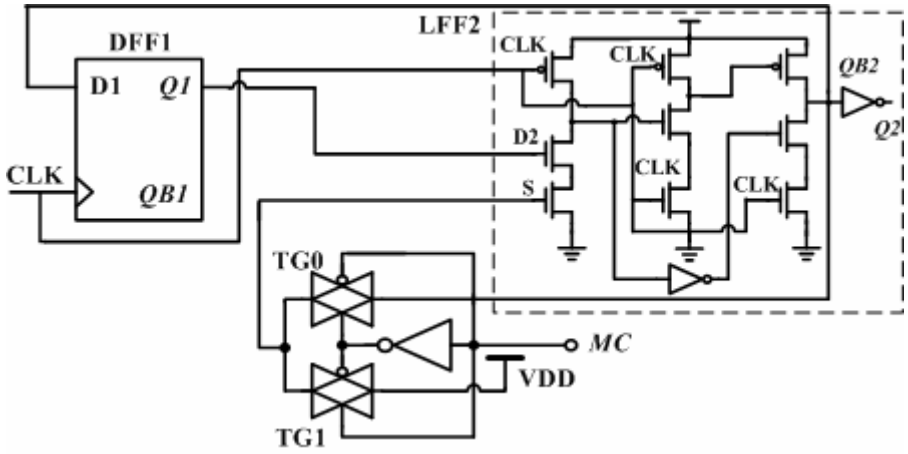


Figure 4.2 Proposed divide-by-3/4 dual-modulus prescaler

Table 4.1 State tables

(a) Divide-by-4 counter

State	Present state Q1Q2	Next state Q1'Q2'
1	0 0	1 0
2	1 0	1 1
3	1 1	0 1
4	0 1	0 0

(b) Divide-by-3 counter

State	Present state Q1Q2	Next state Q1'Q2'
1	0 0	1 0
2	1 0	1 1
3	1 1	0 1

For the divide-by-4 case: Q1Q2 cycles as 00 to 10 to 11 to 01 to 00 and so on. For the divide-by-3 case: the state "01" is skipped and changes directly from 11 to 00. Here, this can be achieved by generating the signal *S* from *MC* using 2-to-1 multiplexer. In turn, NAND-FF2 functions as "Q1 AND *S*". When TG0 is on (TG1 is off), DFF1 and NAND-FF2 function as divide-by-4 counter else as divide-by-3 one.

When the signal *MC*=1, TG0 is turned off and TG1 is on. In other words, the path from QB2 to *S* is disconnected and *S* is connected to VDD. Thus, NAND-FF2 functions as "Q1 AND 1" and the circuit operates in divide-by-4 mode. Whereas, when the signal *MC*=0, TG1 is off and TG0 is on. In turn, the path from QB2 to *S* gets connected. Therefore, NAND-FF2 functions as "Q1 AND QB2" which results in the divide-by-3 mode operation.

A block diagram of a 127/128 DMP is shown in Figure 4.3. It consists of the proposed divide-by-3/4 counter and an asynchronous divide-by-32 counter. The control logic governs the status of the signal MC based on the Mode level and outputs of divide-by-32 counter. In turn, the signal MC controls the 2-to-1 multiplexer. As a result, the whole circuit functions as divide-by-127/128.

It is worth noting here that, in the divide-by-127/128 DMP, when the signal QB2 is fed back to D1 and used to AND with Q1, the divide-by-3/4 counter functions essentially as divide-by-3. Based on this analysis, the "AND" logic block and DFF2 can further be combined to form a LFF2. The logic gate delay and DFF delay are thereby shared to result in reducing propagation delay. Thus, the divide-by-3/4 counter can be realized by only a 2-to-1 multiplexer for mode selection. The critical path delay is minimized as well as the feedback path delay. For MC=0 (Mode=0), when the clock triggers DFFs, the previously-stored signal D1 must reach LFF2 before next clock triggering. Simultaneously, the output of LFF2 also must be fed back to LFF2 which

results in the correct function. This reflects that the speed of the divide-by-3/4 counter is limited not only by the critical path delay but also by the feedback path delay.

The proposed topology shows a smaller critical path delay than the TG-based and NOR-based ones. Furthermore, the feedback path delay is also reduced based on the optimization method of merging flip-flop with the logic between flip-flops, which has been developed recently [14, 17].

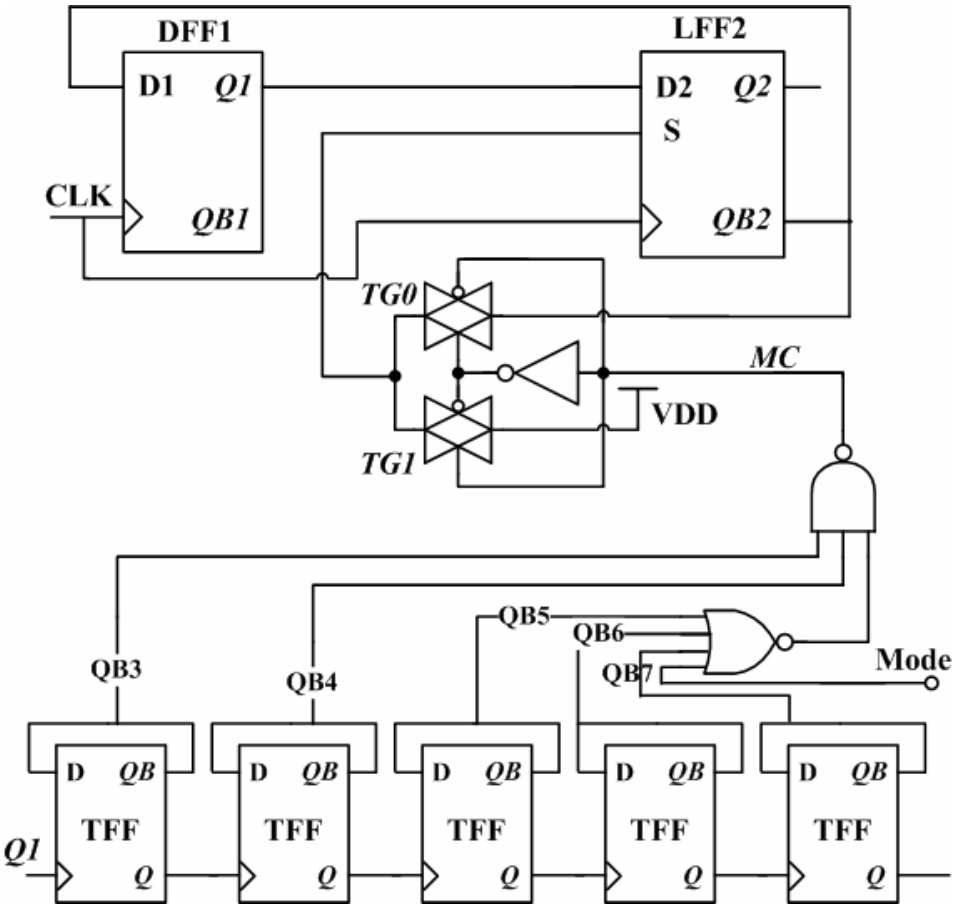


Figure 4.3 Block diagram of the divide-by-127/128 DMP

4.3 Simulation Results

To evaluate the performance of the high-moduli DMP, a divide-by-127/128 DMP using the proposed divide-by-3/4 counter is implemented in TSMC 0.18-μm CMOS process with the help of HSPICE simulation. The high-speed TSPC ratioed DFF [17] is used in the synchronous counter. As the asynchronous counter operates at about one-fourth frequency compared to the synchronous counter, TSPC DFFs [19] are used for low-power considerations. Figure 4.4 shows the simulated waveforms which correspond to the divide-by-127/128 operation at 5 GHz. This validates the fact that, during the divide-by-127 operation, the state "Q1Q2=01" is skipped and the transition in the output occurs early by one clock cycle.

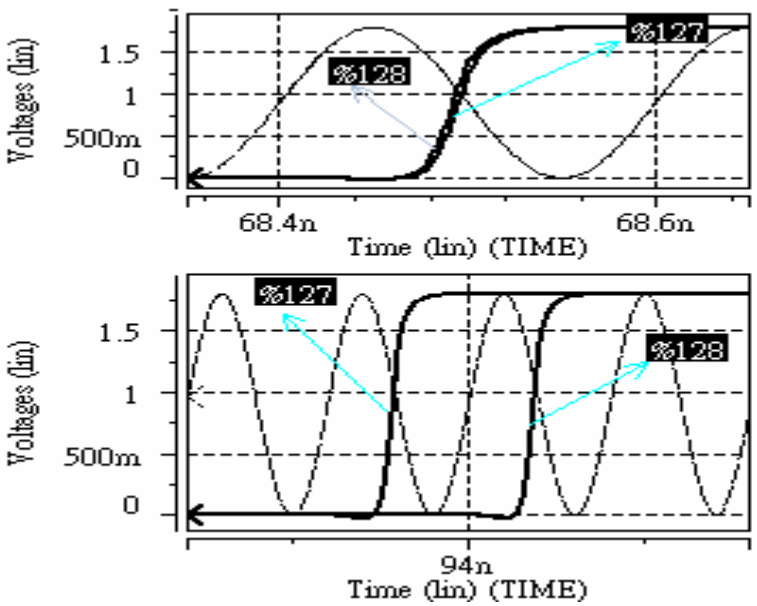


Figure 4.4 Waveforms of the divide-by-127 and the divide-by-128 outputs at 5 GHz

For comparison purpose, both NOR-based and TG-based DMPs may also be realized using the same DFF and the design of the asynchronous counters is also kept the same. In pre-layout simulation, the critical path delays for the DMP of Figure 4.1(a) and Figure 4.1(b) are observed as 112*ps* and 124*ps*, respectively. Whereas, for the

proposed DMP (Figure 4.2), it is noted as 105*ps*. It indeed can be seen that, though the critical path delay is more, the NOR-based DMP provides higher speed than that TG-based DMP. This implies that, in addition to the critical path delay, the feedback path delay also limits the DMP speed. The proposed DMP presents a good reduction in the feedback path delay (about 37%), thereby yielding a significant improvement of ~30% in speed as compared to the TG-based DMP. Table 4.2 provides a summary of the performance comparisons based on simulation results. Figure 4.5 shows the simulated maximum operating frequency versus the supply voltage for the DMPs described above, as well as the corresponding power consumption.

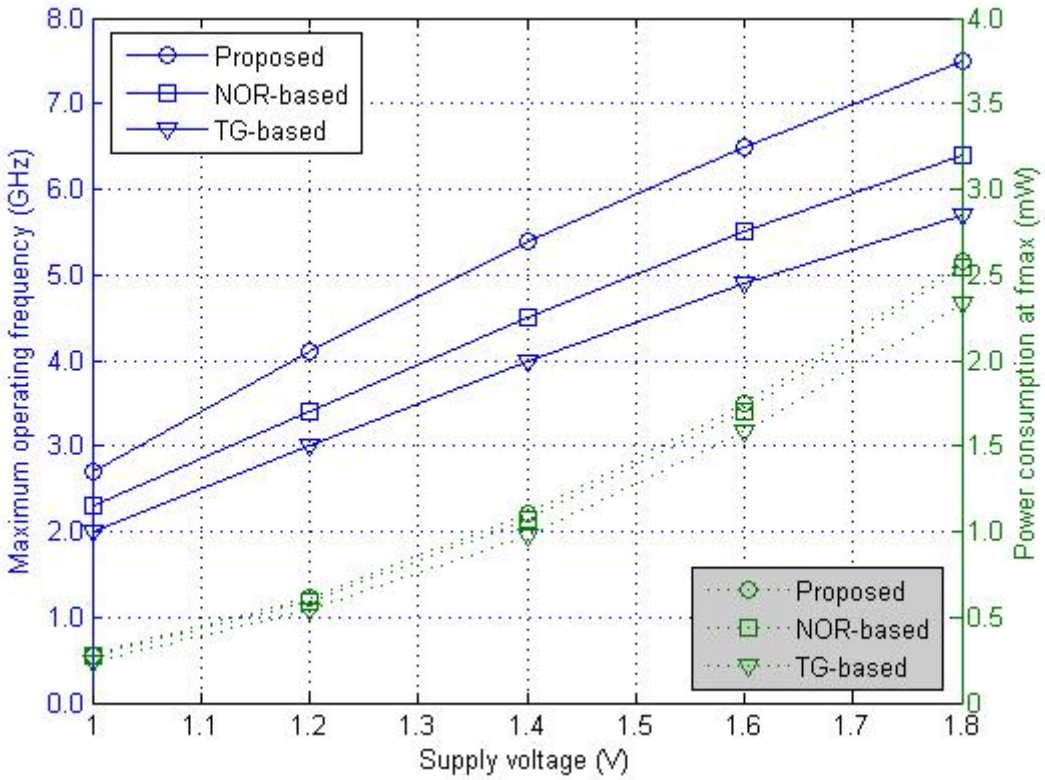


Figure 4.5 Simulated maximum operating frequency and power consumption of DMPs versus power supply voltage.

4.4 Conclusions

This chapter presented a novel and robust design for a high-speed divide-by-3/4 counter. The proposed design is suitable for the high-moduli DMP at high-speed and low-power operation. The use of ratioed-NAND structure helps enhancing the speed of the DMP as a result of smaller feedback path delay and critical path delay. The simulation results demonstrate that the high-moduli DMP using the proposed design not only has a significant speed improvement but also consumes lower power than that proposed by Rana [10] at a given operating frequency.

Table 4.2

Comparison between Rana's and proposed DMP

	NOR-based DMP	**TG-based DMP**	**Proposed DMP**
Synchronous counter flip-flop	TSPC ratioed DFF [17]	TSPC ratioed DFF [17]	TSPC ratioed DFF [17]
Synchronous counter moduli	Divide-by-3/4	Divide-by-3/4	Divide-by-3/4
Critical path element	2 NOR gates	2 Transmission gates	1 Transmission gate
Feedback path element	DFF1(D1 to QB1), 1 NOR	DFF1(D1 to Q1), 1 TG	DFF1(D1 to Q1)
Asynchronous counter flip-flop	TSPC DFF [19]	TSPC DFF [19]	TSPC DFF [19]
Maximum operating frequency	6.4 GHz	5.7 GHz	7.5 GHz
Critical path delay	124*ps*	112*ps*	105*ps*
Feedback path delay	136*ps*	165*ps*	104*ps*
Power consumption at f_{max}	2.5m*W*	2.3m*W*	2.6m*W*
Supply voltage	1.8*V*	1.8*V*	1.8*V*
Technology	0.18-μm CMOS	0.18-μm CMOS	0.18-μm CMOS

CHAPTER 5

Frequency Synthesizer Circuit Design and Simulation

5.1 Introduction

We have introduced the building blocks of PLL-based Frequency Synthesizer in Chapter 2. In this chapter, we will demonstrate the design of individual blocks. A PLL structure is the core of our frequency synthesizer design, which involves the designer's analog and digital expertise. Also, it has many design constraints, most of which have been briefly stated earlier. Here, we will utilize a full-custom design flow for the PLL design. The behavior simulation run by Matlab Simulink has implied the specification and direction for the system design. However, the circuit level design is based on the deep submicron CMOS 0.18-μm technology. In this thesis, our focus is on the integer-N architecture for consideration of the performance verification of our proposed PFD.

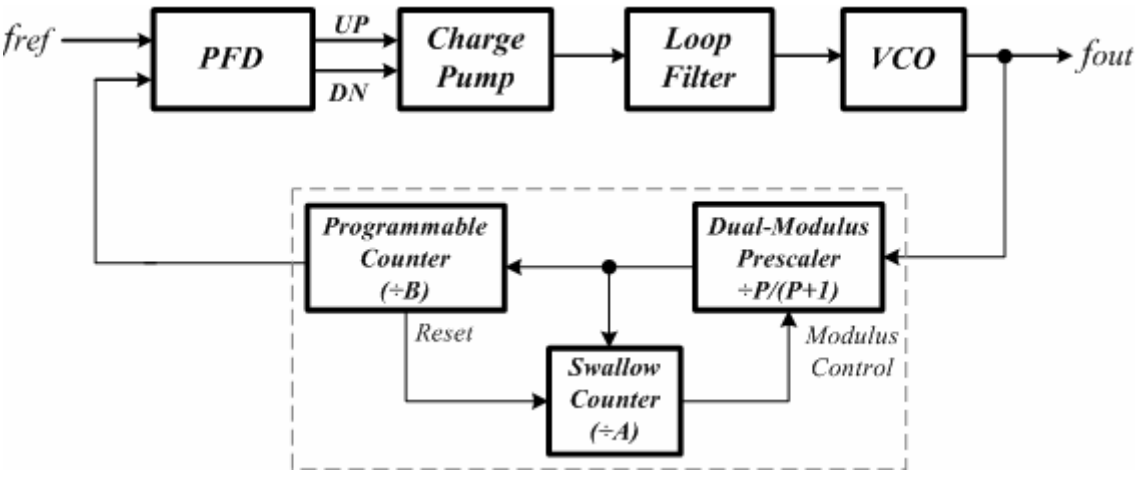


Figure 5.1 The architecture of integer-N frequency synthesizer in this thesis

Figure 5.1 shows the architecture of integer-N frequency synthesizer in this thesis. It has five building blocks, including voltage-controlled oscillator (VCO), prescaler and pulse swallow counter, phase-frequency detector (PFD), charge pump (CP) and loop filter (LF). The following sections will discuss the design and implementation of individual building blocks in a frequency synthesizer. Finally, the implementation of a complete frequency synthesizer with our proposed PFD is achieved. For comparison purpose, an alternative frequency synthesizer with a conventional PFD is also presented. With the help of ADS (Advanced Design System) simulation, the results are provided.

5.2 Voltage-Controlled Oscillator Design

In the design of the frequency synthesizer, the most critical building block is the voltage-controlled oscillator (VCO), which dominates the PLL performance, such as phase noise and tuning range. Some of the important VCO design specifications include phase noise, tuning range, tuning linearity, frequency pushing/pulling, output power, and power consumption.

The design of integrated voltage-controlled oscillators is a broad research topic in itself. A large amount of literature has devoted to the VCO designs in recent years. There are many types of oscillators, such as the relaxation oscillators, ring oscillators, and LC oscillators. The relaxation oscillators and ring oscillators, however, suffer from inherently poor phase noise performance and output spectral purity. In contrast, the integrated LC VCOs exhibit much better phase noise performance, which has improved rapidly over past years. Instead of discrete off-chip VCOs, fully integrated LC VCOs nowadays start to become the best choice for wireless communication applications in the commercial market. A more detailed analysis of the VCO is beyond the scope of this work. This section starts with a brief introduction of the phase noise theory following

which comes an overview of the oscillator topology adopted in this work, and then ends with the simulation results.

5.2.1 Phase Noise Theory

Figure 5.2 shows the well-known Leeson's phase noise model [20] which is linear, time-invariant.

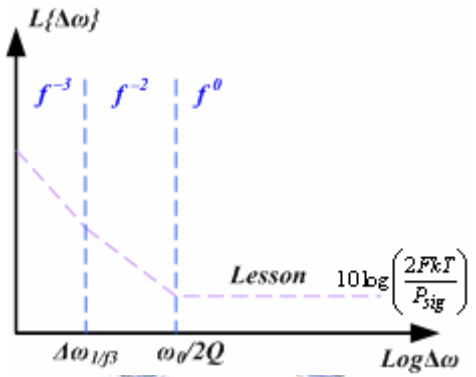


Figure 5.2 Leeson's phase noise model

It can be expressed in terms of signal power P_{sig} and quality factor Q of LC tank as follows:

$$L\{\Delta\omega\} = 10\log\left[\frac{2FkT}{P_{sig}}\{1+\left(\frac{\omega_0}{2Q\Delta\omega}\right)^2\}\left(1+\frac{\Delta\omega_{1/f^3}}{|\Delta\omega|}\right)\right] \quad (5.1)$$

, where F is an empirical factor to account for the increased noise in the $1/(\Delta\omega)^2$ region and $\Delta\omega_{1/f^2}$ is an empirical fitting parameter related to the $1/f$ corner of device noise. All oscillators have the phase noise profile shown above. The Leeson's phase noise model states that increasing the signal amplitude and quality factor of resonator are the only ways to reduce phase noise.

To analyze the oscillator circuit, the basic model is indicated in Figure 5.3, with parasitic element R_P and its current noise source represented by $\overline{I_{R_P}^2}$ [21]. An active element is included in order to compensate the loss of the tank for steady oscillator.

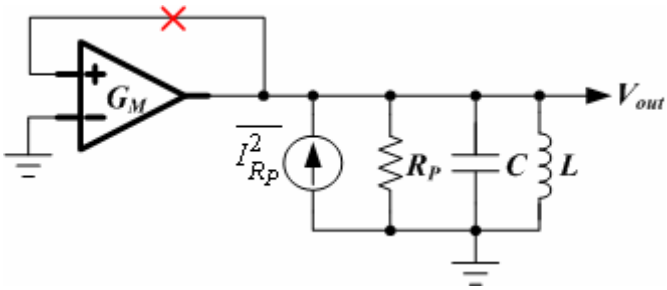


Figure 5.3 Basic model of the oscillator circuit.

To evaluate the phase noise contribution of R_P, the transfer function from the current noise source $\overline{I_{R_P}^2}$ to the output voltage V_{out} must be calculated. It can be expressed as follows:

$$T_{noise,R_P}^2(s) = \frac{\overline{V_{out}^2}}{\overline{I_{R_P}^2}}(s) = \left[\frac{sL}{1 - s(G_M - G_P) + s^2 LC}\right]^2 \tag{5.2}$$

where G_P represents the inverse of R_P. To calculate the phase noise of the oscillator, we have to evaluate this noise transfer function at a frequency offset $\omega_0+\Delta\omega$. To simplify the calculation, we will evaluate the inverse of the noise transfer function, $H_{noise,\ Rp}$ which is defined as

$$H_{noise,R_P}(\omega_0 + \Delta\omega) \doteq \frac{1}{T_{noise,R_P}(\omega_0 + \Delta\omega)} \tag{5.3}$$

We approximate this function by Taylor series expansion around the center frequency:

$$H_{noise,R_P}(\omega_0 + \Delta\omega) \approx H_{noise,R_P}(\omega_0) + \frac{dH_{noise,R_P}(\omega_0)}{d\omega} \cdot \Delta\omega \tag{5.4}$$

The first term $H_{noise,\ Rp}(\omega_o)$ is equal to zero and the second term is equal to

$$\frac{dH_{noise,R_P}(\omega_0)}{d\omega} \cdot \Delta\omega = \omega_0 \cdot \frac{dH_{noise,R_P}(\omega_0)}{d\omega} \cdot \left(\frac{\Delta\omega}{\omega_0}\right) = 2j \cdot \sqrt{\frac{C}{L}} \cdot \left(\frac{\Delta\omega}{\omega_0}\right) \tag{5.5}$$

Then, the transfer function from the current noise source associated with R_P to the output is thus given by

$$T^2_{noise,R_P}(s) \approx \left|\frac{1}{2j}\cdot\sqrt{\frac{L}{C}}\right|^2 \cdot \left(\frac{\omega_0}{\Delta\omega}\right)^2 = \frac{1}{4\cdot(\omega_0 C)^2}\left(\frac{\omega_0}{\Delta\omega}\right)^2 \quad (5.6)$$

So the noise density at a frequency offset $\omega_0+\Delta\omega$ is given by

$$\overline{dV^2_{out}}(\omega_0+\Delta\omega) = T^2_{noise,R_P}(\omega_0+\Delta\omega)\times\overline{dI^2_{R_P}} \approx \frac{1}{4\cdot(\omega_0 C)^2}\left(\frac{\omega_0}{\Delta\omega}\right)^2 \times \frac{4kT}{R_P}\cdot df \quad (5.7)$$

$$\overline{V^2_{out}}\{\Delta\omega\} \approx kT\frac{1}{R_P(\omega_0 C)^2}\left(\frac{\omega_0}{\Delta\omega}\right)^2 \cdot df \quad (5.8)$$

Note that the power needed to maintain the oscillation in the existence of R_P is given by

$$G_{M,R_P} = \frac{1}{R_P} \quad (5.9)$$

The noise can be split up into an amplitude modulation (AM) and a phase modulation (PM). Thus, the phase noise is typically half the value given by Equation (5.8). But for the worst-case analysis, a reduction factor of 1 is used for the prediction of phase noise.

For other parasitic resistors R_l (inductor series resistance) and R_c (capacitor series resistance), the noise generated by them can be calculated in a similar way. The noise contribution of R_l and R_c can be expressed as follows:

$$\overline{V^2_{out,R_l}}\{\Delta\omega\} \approx kTR_l\left(\frac{\omega_0}{\Delta\omega}\right)^2 \cdot df \quad (5.10)$$

$$\overline{V^2_{out,R_c}}\{\Delta\omega\} \approx kTR_c\left(\frac{\omega_0}{\Delta\omega}\right)^2 \cdot df \quad (5.11)$$

The power needed to maintain the oscillation in the presence of R_l and R_c is given by:

$$G_{M,R_l} = R_l(\omega_0 C)^2 \quad \text{and} \quad G_{M,R_c} = R_c(\omega_0 C)^2 \quad (5.11,12)$$

The effective resistance is then defined for the evaluation of the LC oscillators. And the phase noise due to the parasitic resistances can be summarized as follows

$$R_{eff} = R_c + R_l + \frac{1}{R_P(\omega_0 C)^2} \quad (5.13)$$

$$G_M = R_{eff}(\omega_0 C)^2 \tag{5.14}$$

$$\overline{V_{out,R_{eff}}^2}\{\Delta\omega\} \approx kTR_{eff}\left(\frac{\omega_0}{\Delta\omega}\right)^2 \cdot df \tag{5.15}$$

The Q-factor of LC-tank can be expressed in terms of the effective resistance as follows

$$Q = \frac{1}{\frac{1}{Q_{R_c}} + \frac{1}{Q_{R_l}} + \frac{1}{Q_{R_P}}} = \frac{1}{\frac{1}{R_P(\omega_0 C)} + R_l(\omega_0 C) + R_c(\omega_0 C)} = \frac{1}{R_{eff}(\omega_0 C)} \tag{5.16}$$

Similarly, the phase noise contribution of the active element can also be calculated as

$$\overline{dV_{out,G_M}^2}(\omega_0 + \Delta\omega) = T_{noise,R_P}^2(\omega_0 + \Delta\omega) \times \overline{dI_{G_M}^2} \approx \frac{1}{4\cdot(\omega_0 C)^2}\left(\frac{\omega_0}{\Delta\omega}\right)^2 \times 4kT \cdot F \cdot G_M \cdot df$$

$$\overline{dV_{out,G_M}^2}(\omega_0 + \Delta\omega) \approx kT \cdot R_{eff} \cdot F \cdot \left(\frac{\omega_0}{\Delta\omega}\right)^2 df \tag{5.17}$$

, where G_M is given by Equation (5.14) and F is the noise factor of the amplifier.

In the design of an actual oscillator, the transconductance used will be higher than theoretically needed so as to provide enough negative resistance for a safety margin. By multiplying the noise with a factor α in the equation, the amount of noise which the actual amplifier generates in excess of the ideal amplifier is included. Define a factor A as being equal to $\alpha \cdot F$, Equation (5.17) becomes as follows

$$\overline{dV_{out,G_M}^2}(\omega_0 + \Delta\omega) = kT \cdot R_{eff} \cdot A \cdot \left(\frac{\omega_0}{\Delta\omega}\right)^2 df \tag{5.18}$$

Thus, the total phase noise at the output can be expressed as

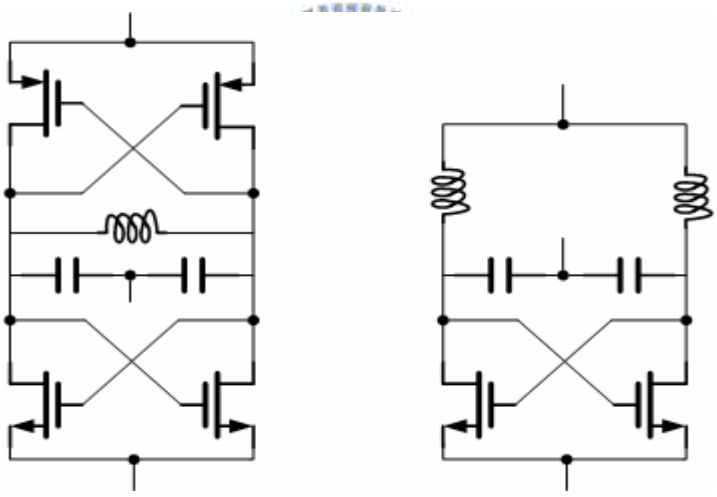
$$\overline{dV_{out,G_M}^2}(\omega_0 + \Delta\omega) = kT \cdot R_{eff} \cdot (1+A) \cdot \left(\frac{\omega_0}{\Delta\omega}\right)^2 df \tag{5.19}$$

The single sideband noise spectral density is:

$$L\{\Delta\omega\} = \frac{\int_{f_0+\Delta f-1/2}^{f_0+\Delta f+1/2} \overline{dV_{out}^2}}{P_{output}} = \frac{kT \cdot R_{eff} \cdot (1+A) \cdot \left(\frac{\omega_0}{\Delta\omega}\right)^2}{\overline{V_{amp}^2}}$$

Similar to the Leeson's model, two ways to reduce the phase noise is increasing the signal amplitude and decreasing the effective resistance, i.e., increasing the Q-factor of the LC-tank. It should note that this model only predicts the phase noise of oscillator in the $1/(\Delta\omega)^2$ region. The limitation of this model is that A is also an empirical factor and the phase noise in the $1/(\Delta\omega)^3$ region cannot be predicted. The phase noise in the $1/(\Delta\omega)^3$ region is mainly due to the up-conversion of low-frequency $1/f$ noise [22]. A time-variant model [22] has been developed to more accurately predict the phase noise in the $1/(\Delta\omega)^2$ and $1/(\Delta\omega)^3$ regions. For simplicity, the LTI model presented above is used to provide rough insight into the phase noise calculation.

5.2.2 Circuit Topology and Design



(a) Complementary cross-coupled VCO; (b) All-NMOS cross-coupled VCO

Figure 5.4 Two typical LC-tank oscillator structures

Figure 5.4 shows two typical LC-tank oscillators. The first one uses NMOS and PMOS cross-coupled pairs (complementary cross-coupled pair) to provide negative-G_M and the second employs all-NMOS cross-coupled pair. In both structures, MOS cross-coupled pair is an active element to compensate for the losses of inductor and capacitor.

There are several reasons why the complementary structure is superior to the all-NMOS structure [23]:

1. The complementary structure offers better rise- and fall-time symmetry. It makes less up-conversion of 1/*f* noise and other lower frequency noise sources.
2. The complementary structure offers higher transconductance for a given current, which results in a better start-up behavior.
3. The complementary structure also exhibits better phase noise performance for all bias points illustrated in Figure 5.5.

As long as the oscillator operates in the current-limited regime, the tank voltage swing is the same for both oscillators. However if we desire to operate in the voltage-limited region, the all-NMOS structure can offer a larger voltage swing.

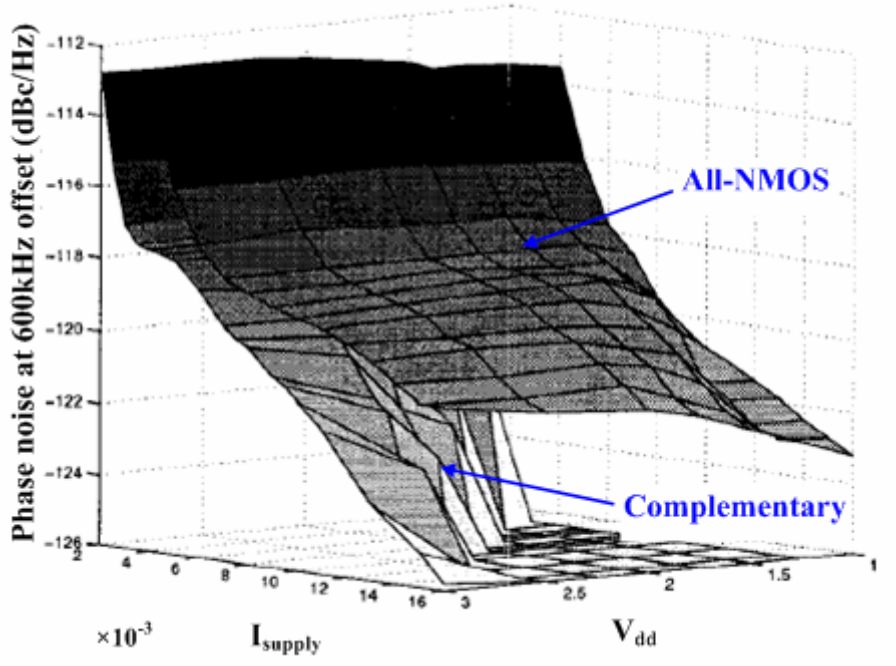


Figure 5.5 Phase noise for the complementary and NMOS-only.

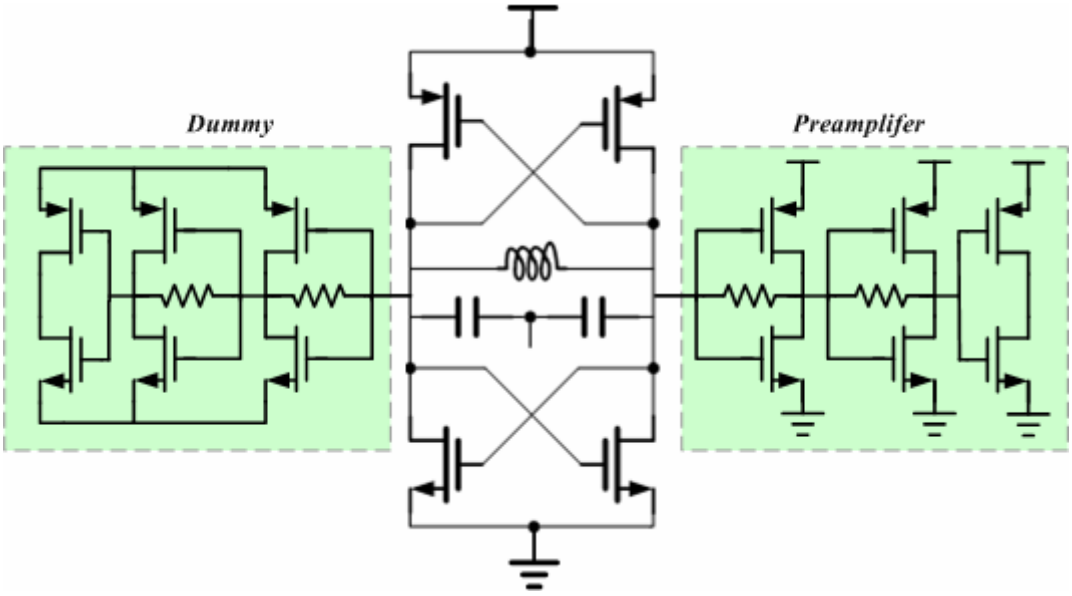


Figure 5.6 Complementary cross-coupled LC VCO without the tail current source

Figure 5.6 illustrates the schematic of the complementary cross-coupled LC VCO without the tail current source, which is adopted in this work. From the phase noise point of view, this topology reveals better noise performance than the one in Figure 5.4(a). This is due to the fact that the $1/f^3$ noise of the topology without the tail current can only originate from the flicker noise of the MOS transistor switches. These switches are expected to feature lower flicker noise than the tail current source that dominates the $1/f^3$ noise, for two main reasons. First, the switches operate in triode region for large portions of the oscillation period; hence, they exhibit lower current flicker noise than the tail transistor that continuously operates in saturation. Second, switched MOS transistors are known to have lower flicker noise than transistors biased in the stationary condition [24]. Nevertheless, the main drawback of this topology is a higher sensitivity of the frequency to the voltage supply (frequency pushing). This effect can be alleviated by using a supply voltage regulator.

5.2.3 Simulation Results

Simulation results as shown in Figure 5.7 are the phase noise of complementary cross-coupled LC VCO at 2.6 GHz. The phase noise is -99.1 dBc/Hz at 100KHz frequency offset and -121.6 dBc/Hz at 1MHz frequency offset.

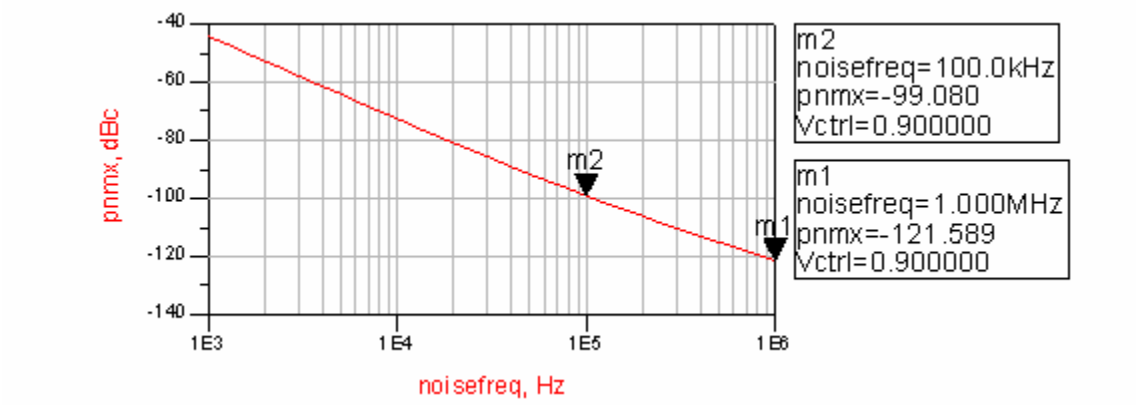


Figure 5.7 Simulated phase noise of the LC VCO

Figure 5.8 shows the output frequency tuning range, which is about 2.36 GHz to 2.95 GHz. For the designed frequency band, the average gain of VCO is about 425 MHz/V. The current consumption of the VCO circuit including the preamplifier is 17.45m*A*. The VCO core draws 10.4m*A*, thus its power consumption is 18.7m*W*.

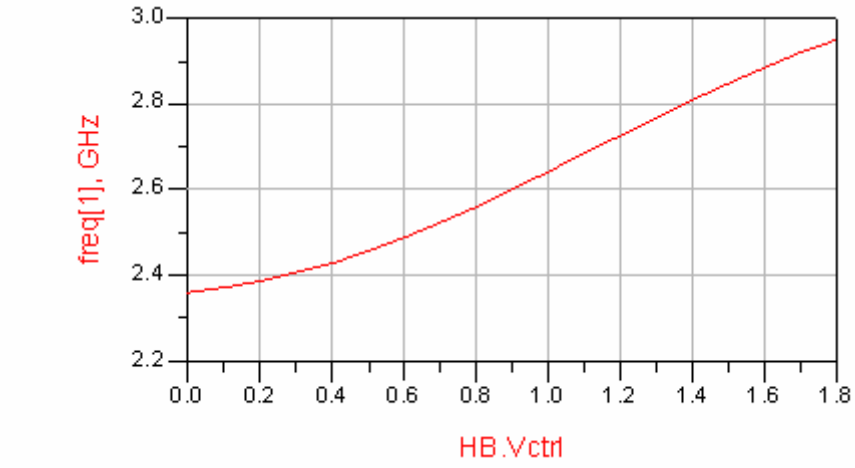


Figure 5.8 Simulated output frequency tuning range of VCO

5.3 Frequency Divider Design

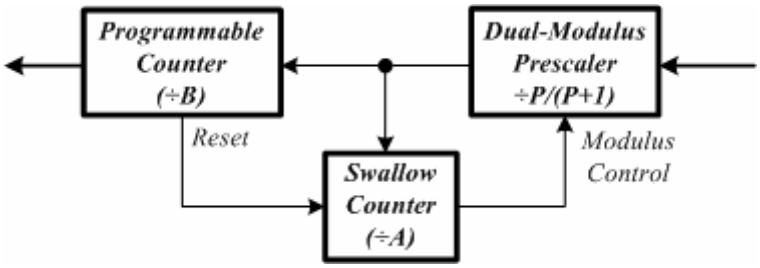


Figure 5.9 Block diagram of the pulse-swallow frequency divider

In this work, the programmable integer-*N* frequency divider is based on a pulse-swallow topology. Shown in Figure 5.9 is the block diagram of a conventional implementation, consisting of a "dual-modulus prescaler", a "programming counter", and a "swallow counter". Initially, the DMP divides by *P+1* until the swallow counter is full. Afterwards, it overflows and outputs a control pulse to set the DMP in divide-by-*P* mode. The division continues until the programming counter is full and the DMP is set back in divide-by-*P+1* mode. The pulse-swallow divider then generates one complete output cycle for total $(P+1)\cdot A+P\cdot(B-A)$ input clocks and resets both counters. Note that $B \geq A$ for the correct operation of the pulse-swallow divider.

5.3.1 Pulse-Swallow Frequency Divider

In order to verify the performance of our proposed PFD design, we intend to choose 100 MHz (200 MHz) as the reference frequency. Our VCO frequency tuning range is 2.36~2.95 GHz. Therefore, the integer division ratio is designed from 25 to 28 (from 13 to 14), which implies only four (two) channels. We construct the pulse-wallow frequency divider by a divide-by-3/4 DMP and a generic architecture of a programmable swallow counter combined with a fixed-ratio programming counter. Figure 5.10 shows the generic architecture of a programmable swallow counter and a

fixed-ratio programming counter [25]. b_0 to b_n are the channel selection bits that control the division ratio. The division ratio N is given as

$$N = (P+1)A + P(B-A) = (P+1)\underbrace{(2^{n+1} - b_n 2^n - \ldots - b_1 2 - b_0)}_{A} + P(2^{n+2} - A) \quad (5.20)$$

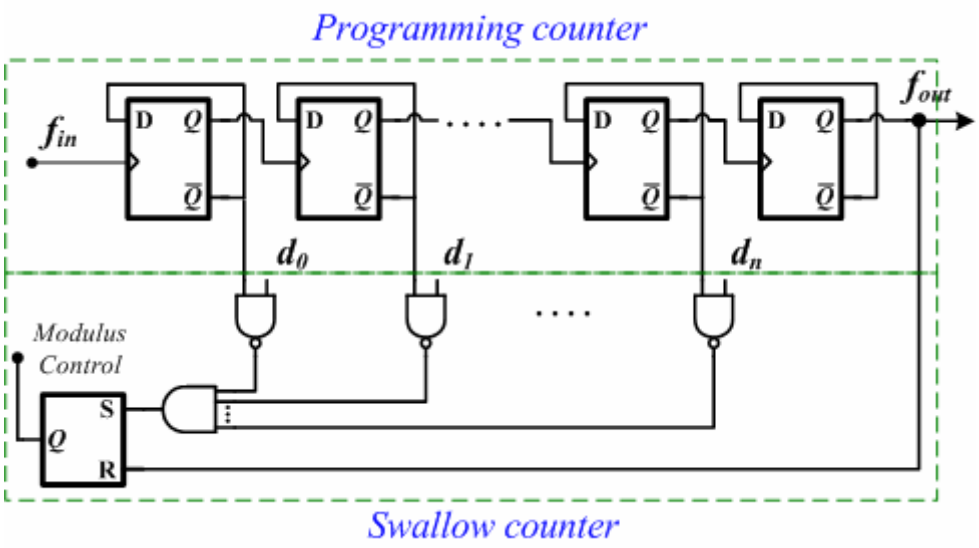


Figure 5.10 Generic architecture of the programming and swallow counters [25].

Note that A is the division ratio of the swallow counter, which is the 2's complementary of $b_n \ldots b_0$ and equal to $(2^{n+1}\text{-}b_n 2^n\text{-}\ldots\text{-}b_1 2\text{-}b_0)$, and B is the division ratio of the program counter, which is 2^{n+2}. In this work, the division ratio of the swallow counter is from 25 to 28 (13 to 14), and the division ratio of the swallow counter is 8 (4). Figure 5.11 is the resulting frequency divider with our proposed divide-by-3/4 DMP. Table 5.1 describes the pulse-swallow divider step function.

Table 5.1

(a) The pulse-swallow divider step function (100 MHz f_{ref})

Channel	Center Freq.	B	A	b_0	b_1	P	N
1	2.5 GHz	8	1	1	1	3	25
2	2.6 GHz	8	2	1	0	3	26
3	2.7 GHz	8	3	0	1	3	27
4	2.8 GHz	8	4	0	0	3	28

(b) The pulse-swallow divider step function (200 MHz f_{ref})

Channel	Center Freq.	B	A	b_0	P	N
1	2.6 GHz	4	1	1	3	13
2	2.8 GHz	4	2	0	3	14

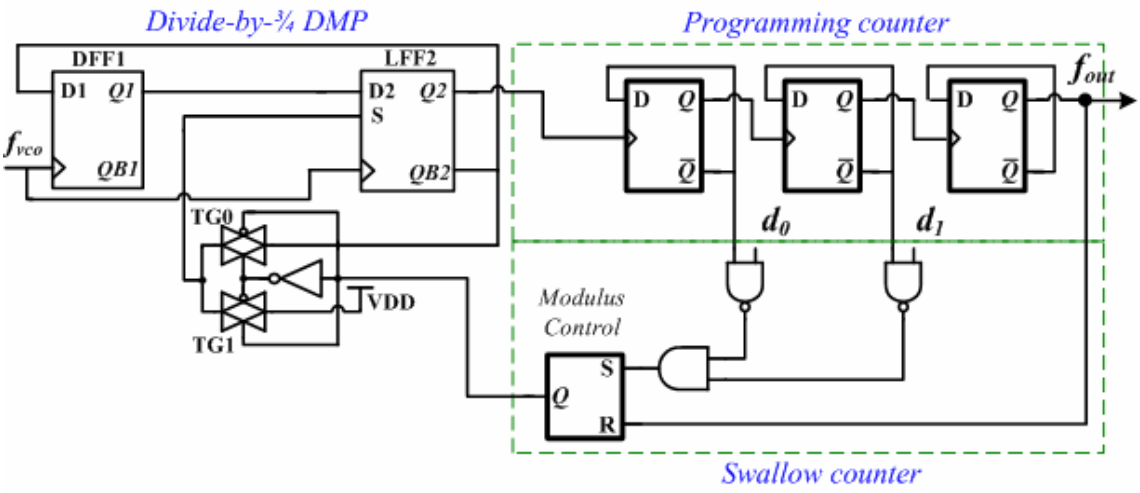


Figure 5.11 Pulse-swallow frequency divider with our proposed divide-by-3/4 DMP

5.3.2 Simulation Results

Figure 5.12 shows the simulation result of the divide-by-3/4 DMP with input frequency 2.6 GHz). In the waveforms, the divide-by-3/4 function works correctly.

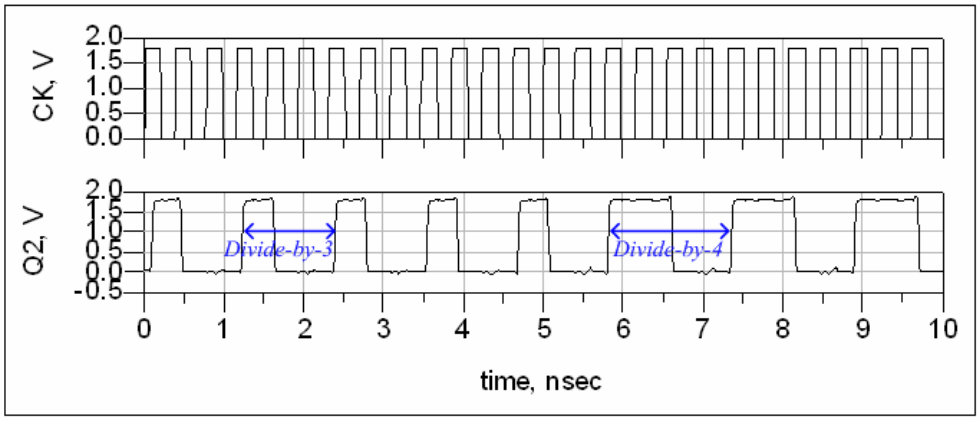


Figure 5.12 Simulated waveforms of the divide-by-3/4 DMP

Figure 5.13 shows the programming and swallow counters simulation results. Those functions work correctly in divide-by-13 and divide-by-14 modes, respectively.

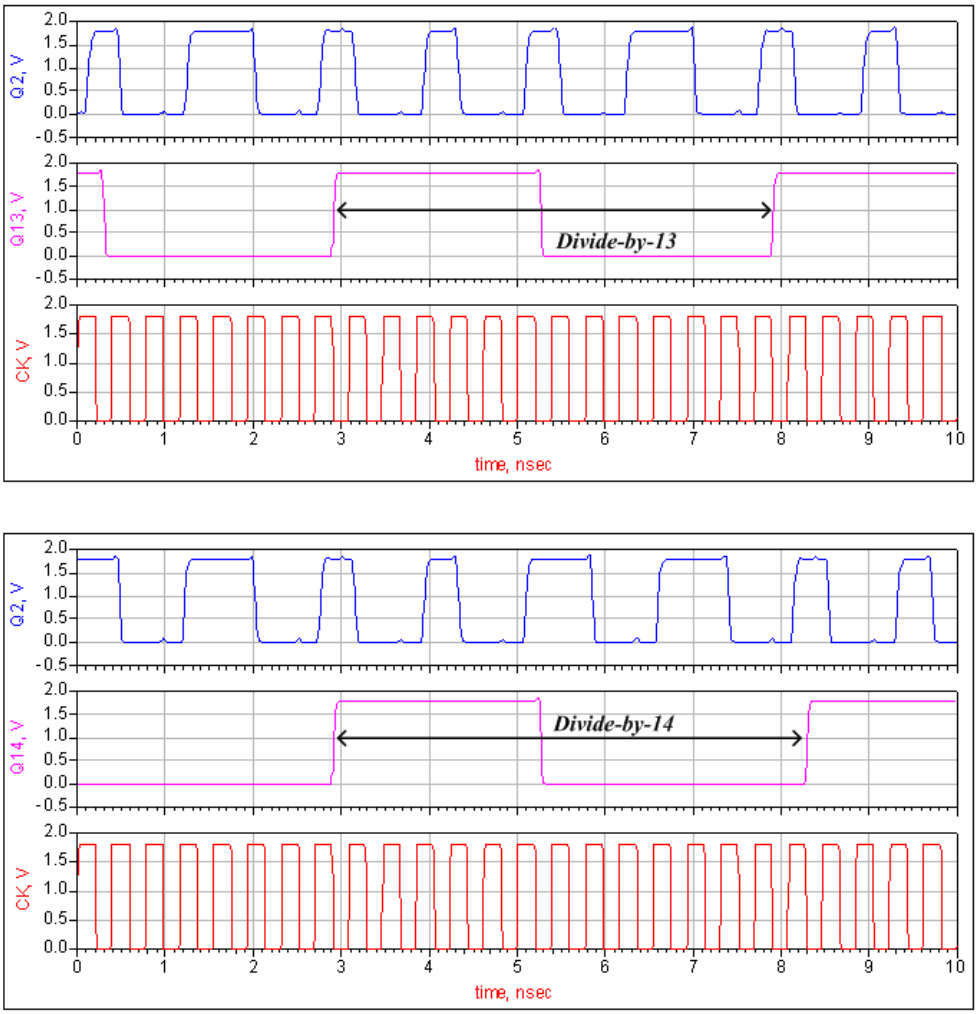


Figure 5.13 Simulation results of the programming and swallow counters (÷13/÷14)

5.4 Phase-Frequency Detector Design

The phase-frequency detector (PFD) compares the phase between the reference signal and the divided VCO signal, in turn producing the corresponding output signal proportional to the phase difference. In this thesis, an alternative fast frequency-acquisition PFD is proposed to achieve a fast-locking PLL. As discussed earlier, our proposed PFD architecture comprises two parts: one is the BZ-detector for aided

acquisition and the other is the classical dead-zone-free PFD. Figure 5.14 shows the circuit schematic of the classical dead-zone-free PFD which is the commonly used precharged PFD [7]. The implementation of the *LFFs* in the BZ-detector has been revealed in Figure 3.8.

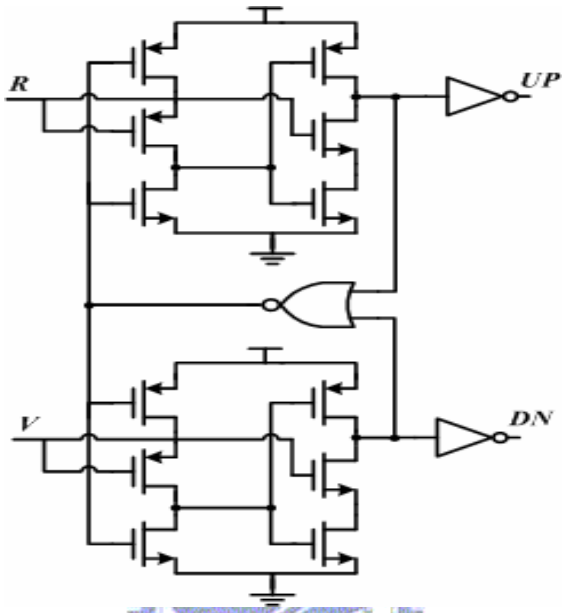


Figure 5.14 Circuit schematic of the classical dead-zone free precharged PFD

To examine the phase- and frequency-discriminator characteristics of our proposed PFD, transient simulation results under different scenarios are presented as shown in Figure 5.15. In Figure 5.15(a), the two inputs have equal frequencies but *R* leads *V*. Figure 5.15(b) reveals that *R* has a higher frequency than *V*. Figure 5.15(c) shows the comparison of the proposed and conventional PFDs at large phase errors, which demonstrates the improvement in the blind-zone problem. Note that the resulting dead-zone duration is 370*ps* and the blind-zone duration is 690*ps*.

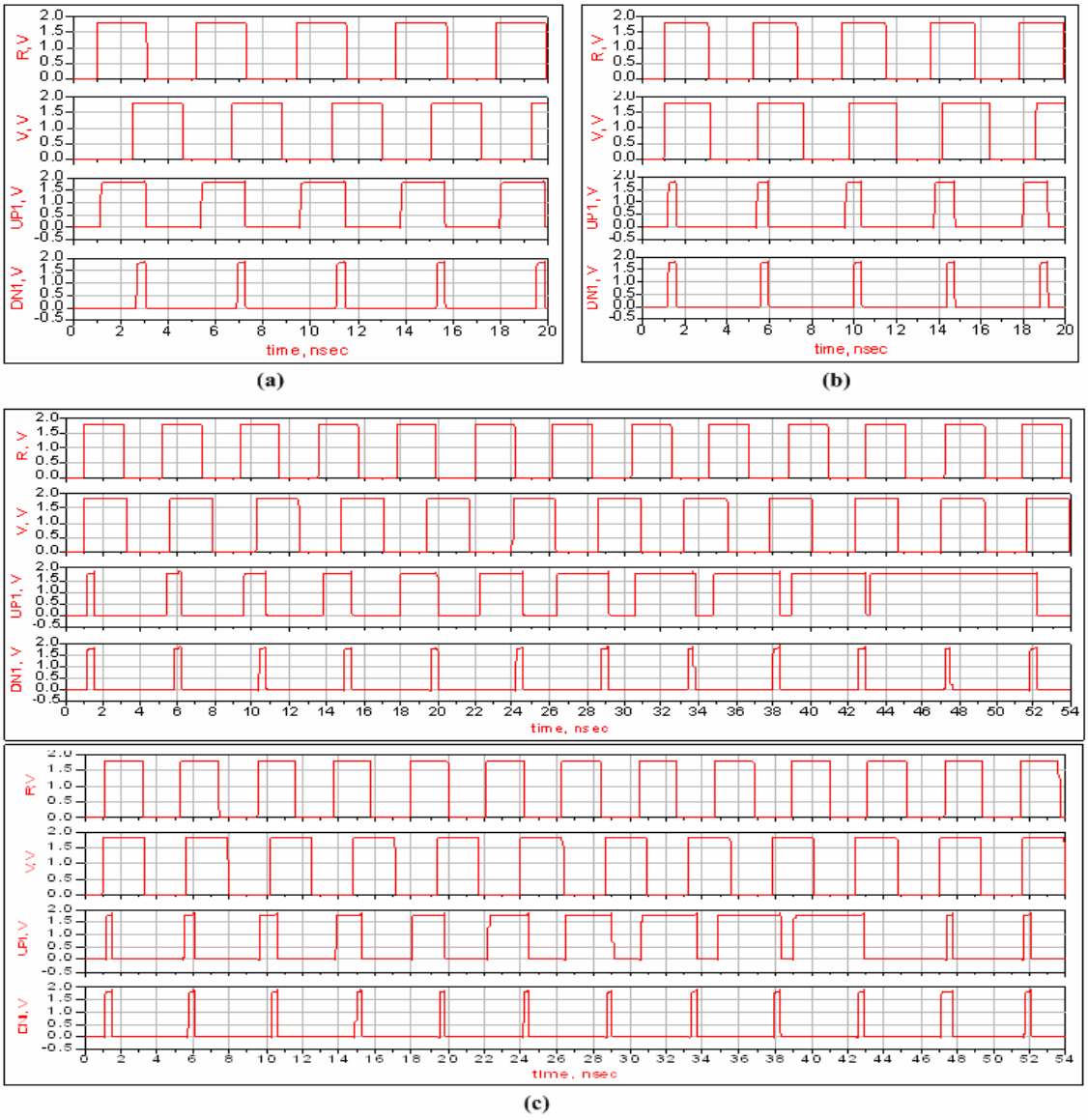


Figure 5.15 (a) Both inputs have equal frequency but *R* leads *V*; (b) *R* has a higher frequency than *V*. (c) Comparison of the proposed (Upper diagram) and conventional (Down diagram) PFDs at large phase errors.

5.5 Charge Pump and Loop Filter Design

In the design of CPPLL-biased frequency synthesizers, the advantages of using an ideally charge pump are zero static phase error and an unbounded pull-in range, and the design of the charge pump is easier than that of the operation amplifier. However, its design involves many issues as discussed roughly in Chapter 1.

5.5.1 Charge Pump Circuit Topology

In this work, the topology of a CMOS charge pump proposed in [26] is adapted and shown in Figure 5.16 (a). Positive feedback and current reuse are used to obtain faster switching speed and lower power consumption. This topology [26] evolves from some previous topologies using current steering techniques (Figure 5.16 (b)) [27].

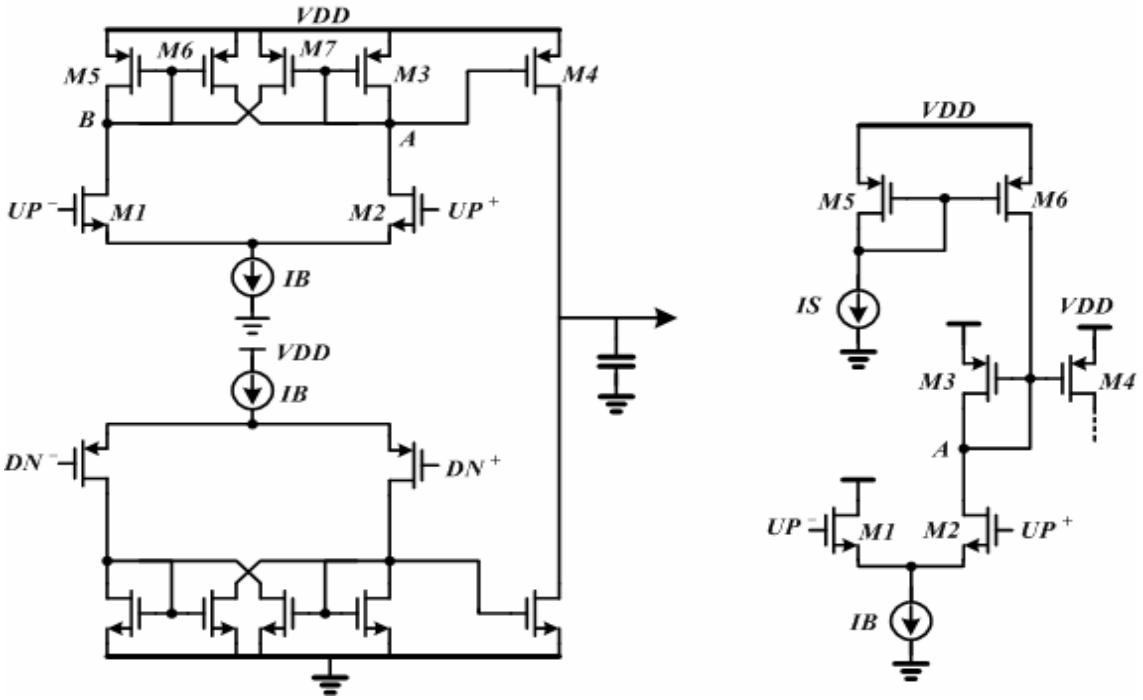


Figure 5.16 Circuit schematics: (a) positive-feedback charge pump and (b) charge pump in [27]

As shown in Figure 5.16(b), when signal in UP^+ is greater than UP^-, the I_B current source is steered on *M2*. The difference between this current and I_S is mirrored by *M3* and *M4*, producing the charge/discharge current. On the other hand, when the signal in UP^+ is lower than UP^-, the current is steered on *M1*. The pull-up circuit (*M5*-*M6*) is used to increase the charge speed of node *A* and turn off *M4*. However, such a circuit presents several drawbacks, such as a slow path (pulling up *A* to *VDD*) and excess power consumption. In Figure 5.16(a), the proposed topology [26] increases the charge

speed of node *A* and saves power by I_B current reuse. With the added pull-up transistor *M7*, a simple positive feedback configuration is achieved, thus increasing the charge speed of node *B*. It turns out that the switching speed of the charge pump is increased.

5.5.2 Loop Filter

The loop filter in this work is a second-order passive filter that consists of two capacitors and one resistor. The resulting PLL is then a type-II third-order loop. The capacitors and resistor of the loop filter should be properly chosen to perform the required filtering function and maintain the stability of the loop without introducing too much noise. The component values in the filter are calculated, which follows the design flow:

(1) The average VCO gain in this work is about 425 MHz/*V*.

K_{VCO} = 425 MHz/*V*.

(2) The input reference clock is 200 *MHz*.

f_{ref} = 200 MHz

(3) The loop bandwidth is chosen to be 2 MHz

K = 2 MHz

(4) A 56° phase margin is chosen. It corresponds to a γ of 3. In other words, the zero ω_Z is placed a factor 3 below *K*, and the pole ω_P is placed a factor 3 above *K*, to obtain a phase margin of approximately 56°.

(5) An equivalent charge pump current is 0.1m*A*.

I_{CP} = 0.1m*A*

(6) In the design of the loop filter, the loop filter design software from National Semiconductor called “PLL Loop Filter Design”. As shown in Figure 5.17 and Figure 5.18, the loop filter is calculated as follows:

R_1 = 4.24 KΩ, C_1 = 61.4 pF, C_2 = 6.33 pF.

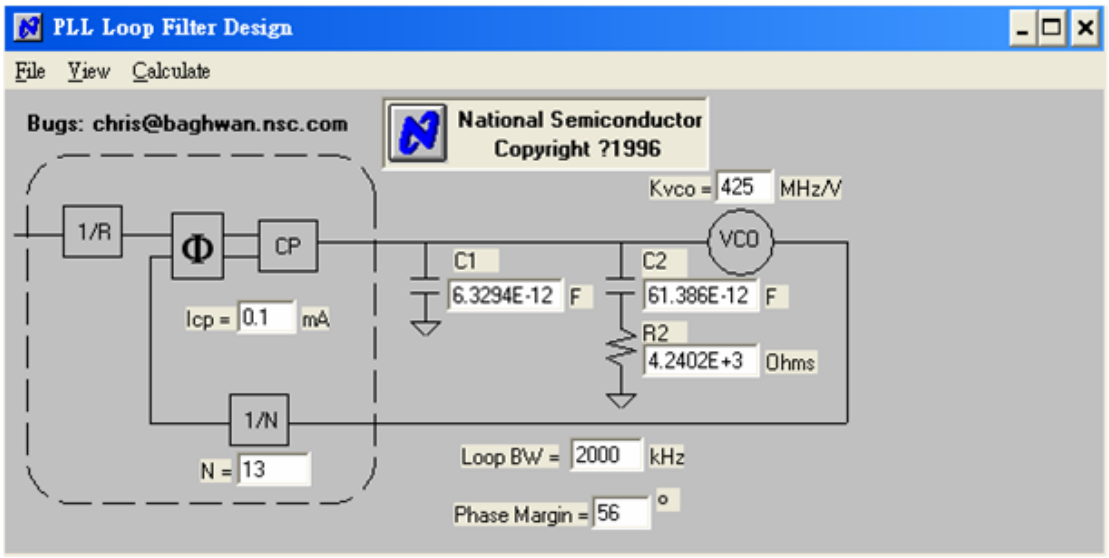


Figure 5.17 PLL loop filter design software

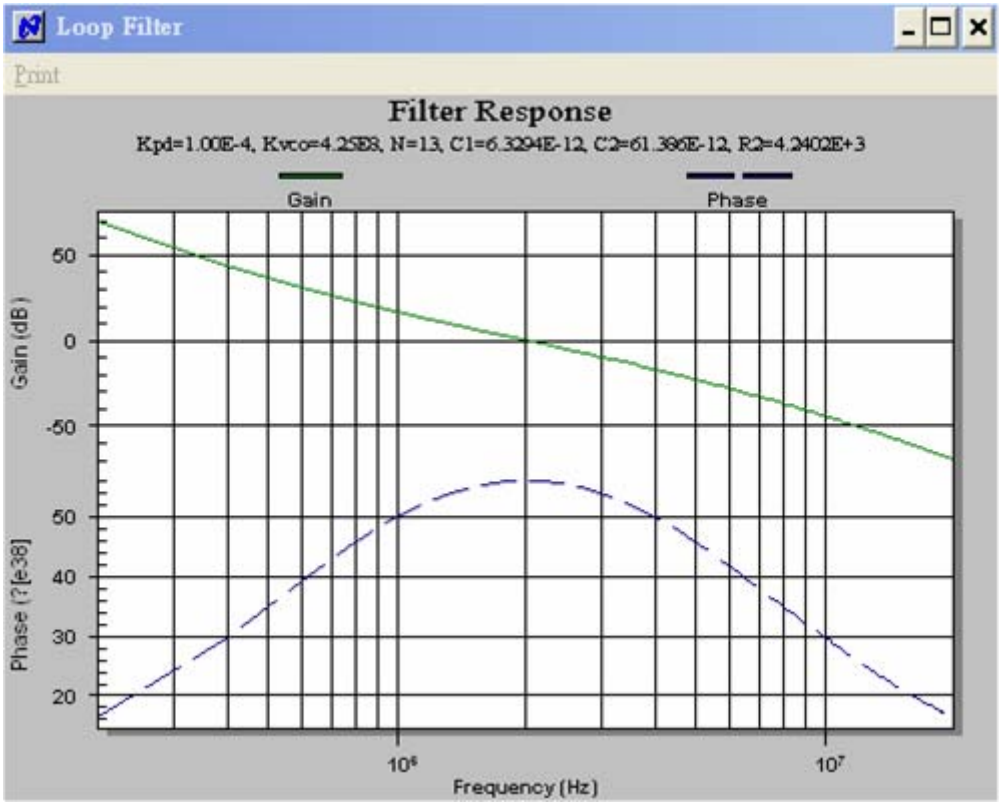


Figure 5.18 Bode plot of the loop gain and phase margin in this frequency synthesizer design

In Table 5.2, we summarize the final PLL parameters in this work.

Table 5.2

The final PLL parameters in this work

Final PLL parameters in this work		
VCO gain	K_{VCO}	425 MHz/V
Open loop gain bandwidth	K	2 MHz
Phase margin	PM	56º
Zero frequency	ω_Z	0.625 KHz
Pole frequency	ω_P	6.4 MHz
Passive elements	R_1	4.24 KΩ
	C_1	61.4 pF
	C_2	6.33 pF

5.5.3 Simulation Results

The transient simulation result of the PFD/CP combination with only a 10 pF capacitor as the loop filter is shown in Figure 5.19(a) and (b). In Figure 5.19(a), the VCO control line is charged toward V_{DD} in case of R leading V, indicating that the function works properly. In Figure 5.19(b), the VCO control line is discharged toward 0 in case of V leading R.

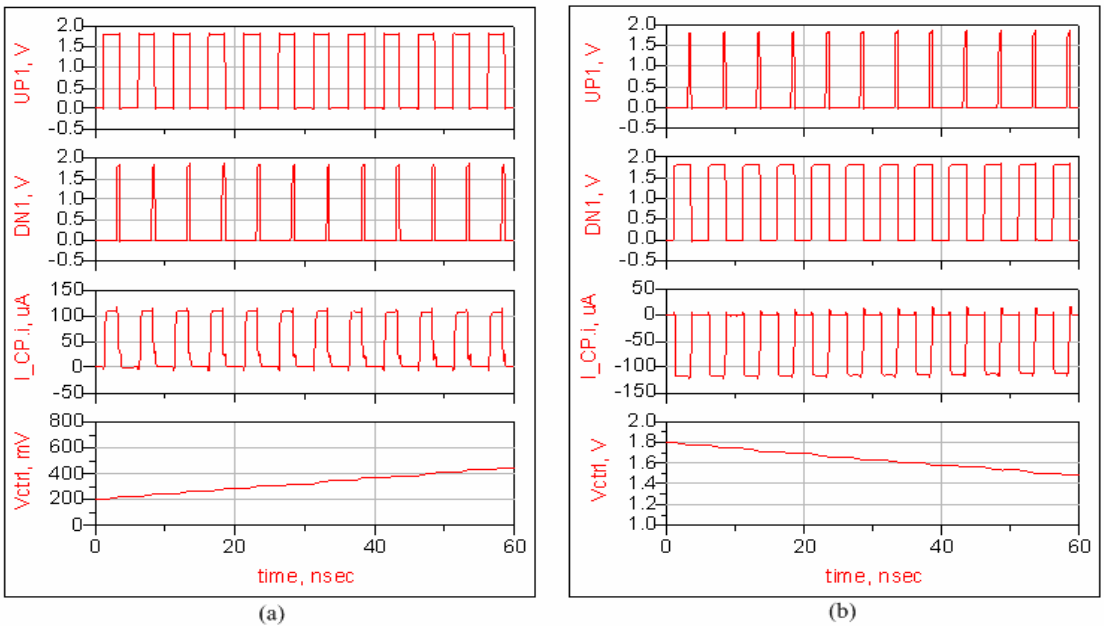


Figure 5.19 Simulation results of the charge pump (a) charging; (b) discharging

5.6 Closed-Loop Frequency Synthesizer Simulation

With all discussion of the architecture and simulation of each block above, the complete frequency synthesizer can thus be implemented by combining all of them. In this work, a third-order PLL is achieved. However, the impedance of the input control voltage of the VCO is a low-pass system with a resistor series with varactors, thus making the PLL go to the forth-order loop. This turns out to create an additional pole, and this pole is far away from another pole and zero, thus its effect can be neglected.

Because of the tremendous amount of gate counts in a frequency synthesizer (especially in the frequency divider), closed-loop simulation at the gate-level will take a lot of times. Hence, some prior architecture simulations must be done with *Simulink* in order to achieve a well-defined closed-loop behavior (system level). A behavior model for the complete PLL is implemented in *Simulink* as shown in Figure 5.20.

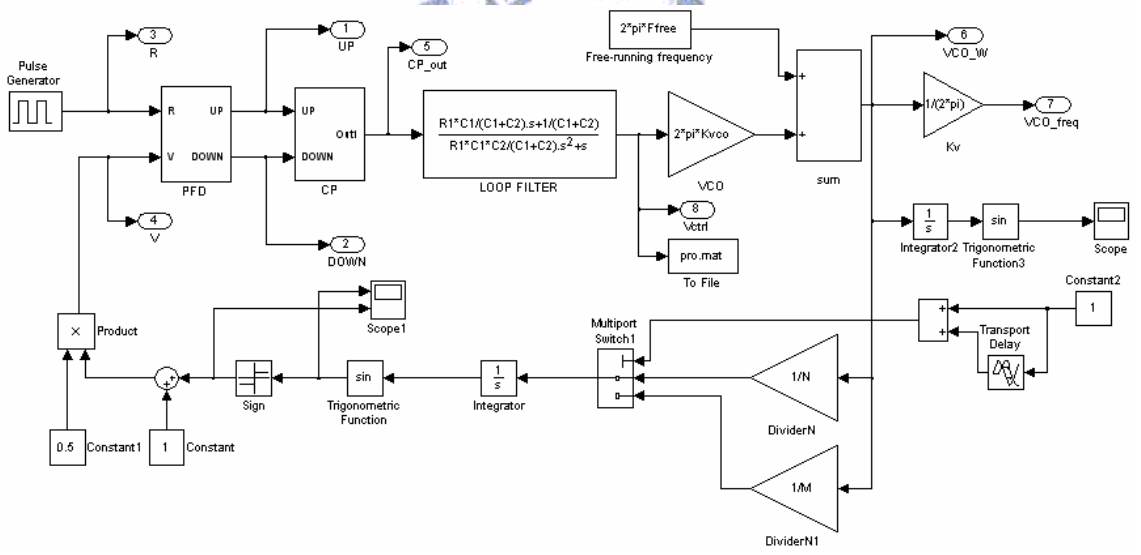


Figure 5.20 Behavior model by *Simulink*.

And the transient simulations of the behavior model are shown in Figure 5.21 and Figure 2.22. The locking times from 2.36 GHz to 2.6 GHz and to 2.8 GHz (equivalently, *0V to 0.5648V* and to *1.306V*) are about 1.36μs and 1.76μs, respectively.

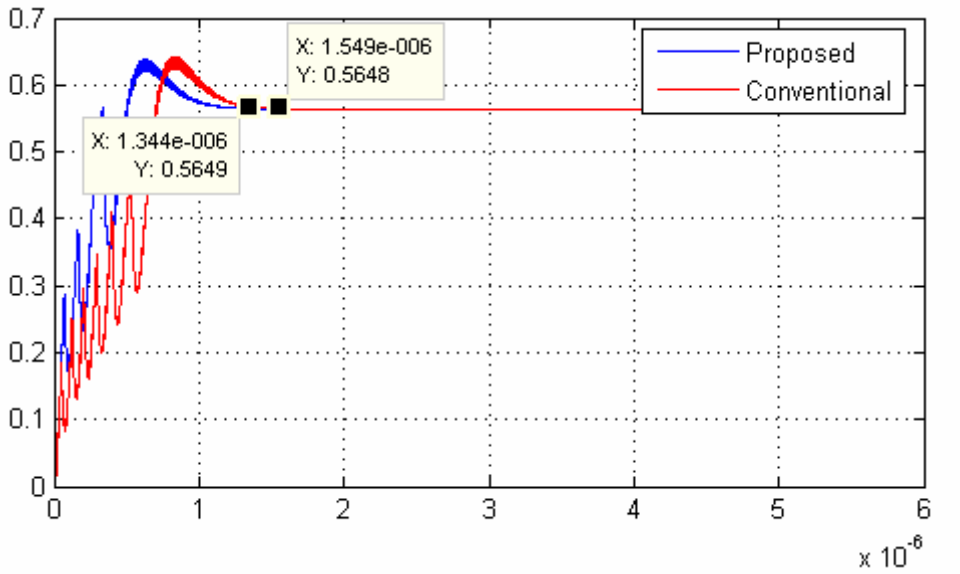


Figure 5.21 Simulated locking time from 2.36 GHz to 2.6 GHz by *Simulink*

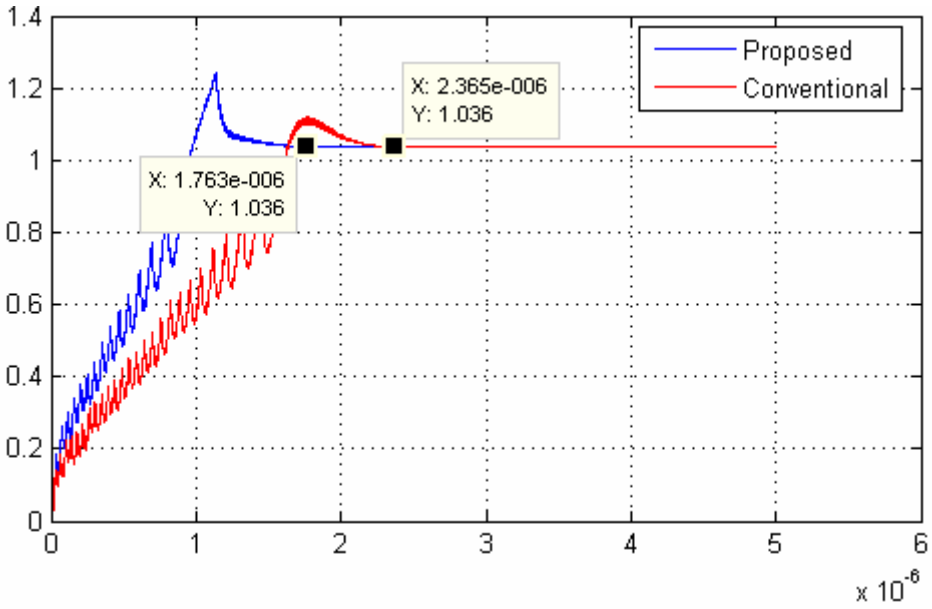


Figure 5.22 Simulated locking time from 2.36 GHz to 2.8 GHz by *Simulink*

Note that, for comparison purpose, transient simulations of a PLL employing the conventional PFD are also done and displayed. Thus, it can be seen that the frequency synthesizer using our proposed PFD has 15% and 25% improvement in locking time over the one using the conventional PFD while locking to 2.6 GHz and 2.8 GHz, respectively.

After Matlab *Simulink* simulation, we utilize ADS to complete the whole closed-loop simulation in TSMC 0.18-μm CMOS process model. Figure 5.23 and 5.24 shows the transient simulation results of the frequency synthesizer. Figure 5.23 (a) reveals that the output frequency is from 2.36 GHz to 2.6 GHz and the locking time is 1.43μs. Figure 5.23 (b) shows that the output frequency exactly locks to 2.6 GHz.

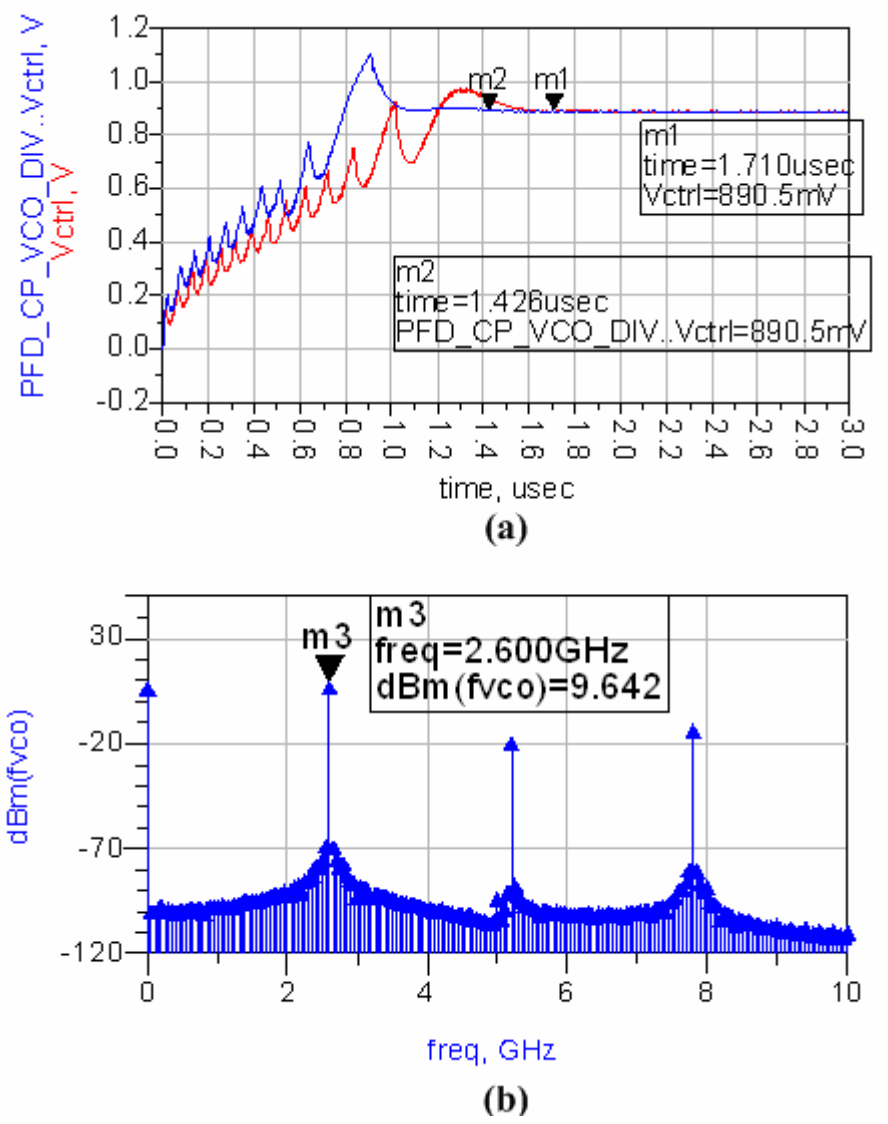


Figure 5.23 (a) Simulated locking time from 2.36 GHz to 2.6 GHz by ADS; (b) Its stable output spectrum at 2.6 GHz.

And Figure 5.24 (a) indicates that the locking time from 2.36 GHz to 2.8 GHz is 1.93μs. The output spectrum at 2.8 GHz is shown in Figure 5.24(b).

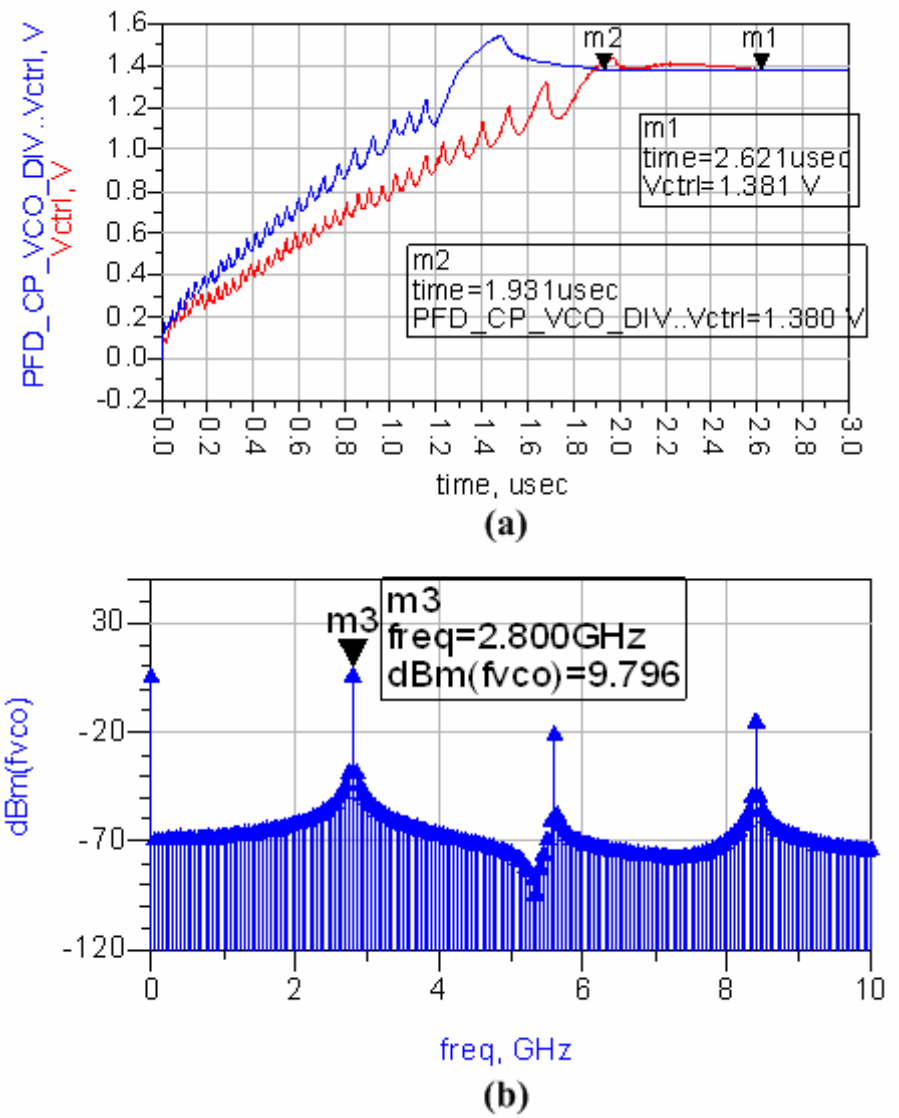


Figure 5.24 (a) Simulated locking time from 2.36 GHz to 2.8 GHz by ADS; (b) Its stable output spectrum at 2.8 GHz.

As shown in Figure 2.4 and 2.5, once again, the results prove that the frequency synthesizer using our proposed PFD shows 16% and 26 % improvement in locking time over the one using the conventional PFD while locking to 2.6 GHz and 2.8 GHz, respectively.

Finally, Table 5.3 lists all results about the simulation of frequency synthesizer in this work. The total power consumption of the frequency synthesizer is 25.9mW.

Table 5.3

Summary of the simulation results of frequency synthesizer

VDD		1.8V
Power	PFD	1.8mW
	CP	0.36mW
	VCO	18.7mW
	FD	5.06mW
	Total	25.9mW
Output frequency		2.36~2.95 GHz
f_{ref}		200 MHz
Phase margin		56º
Loop bandwidth		2 MHz
Loop filter	R_1	4.24 KΩ
	C_1	61.4 pF
	C_2	6.33 pF
VCO gain		425 MHz/V
VCO phase noise @2.6GHz		-99 dBc/Hz @100KHz
		-121.6 dBc/Hz @1MHz
Locking time	2.36 to 2.6 GHz	1.43μsec
	2.36 to 2.8 GHz	1.93μsec
Technology		0.18-μm CMOS, TSMC 1P6M

CHAPTER 6

Conclusion and Future Works

6.1 Conclusion

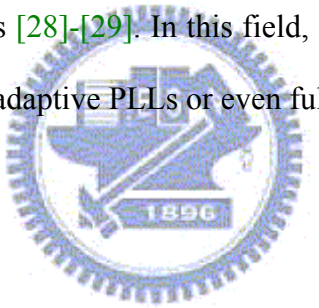
The design of integrated PLLs still remains one of the most challenging and time-consuming tasks in communication systems. In order to meet the stringent requirements of emerging communication technologies, low jitter or phase noise, fast settling and low power are some of the most crucial aspects in PLL designs, which involve a lot of design issues and trade-offs. Among these issues, the dead-zone and blind-zone issues become more critical and serious as the tendency for higher reference frequency is further pursued. As a result, they will further degrade the jitter and/or settling speed performance of PLLs.

This thesis concentrates on these two design issues and provides a solution to achieve faster settling and low jitter performance. A robust frequency acquisition PFD and a high speed dual-modulus prescaler are proposed and applied in our work. With the help of Hspice and Advanced Design System (ADS), a low-jitter and fast-locking CMOS integer-N frequency synthesizer for wireless communication applications has been developed. The synthesizer is implemented in a 0.18-μm CMOS technology at 1.8V power supply. Its working frequency is from 2.36 to 2.95 GHz and the close-loop locking time from 2.36 to 2.8 GHz is 1.93μsec. For comparison purpose, a frequency synthesizer with a conventional PFD is also implemented. Simulation result shows that the synthesizer using our proposed design has an improvement of up to 25% in settling time.

6.2 Future Works

In this work, we only focus on the issues related to the PFD in a PLL. However, to design a good performance frequency synthesizer, it is also required to consider the design issues which arise from the non-idealities of the charge pump. Neglecting those issues will cause high-energy spurious tones at the output spectrum of PLLs. Therefore, in order to achieve a good performance of low phase noise, we will make more efforts to the design of the charge pumps with perfect current matching in the future works.

In addition, although the settling speed is obviously improved, there is still a lot of space for improvement. Several extensive studies have been underway to further speed up the setting behavior of PLLs [28]-[29]. In this field, it maybe worth our effort in the future works, such as partially-adaptive PLLs or even fully-adaptive PLLs.



References

[1] F. Gardner, "Charge-pump phase-locked loops," *IEEE Trans. Comm.,* vol. 28, pp. 1849-1858, Nov. 1980.

[2] T. C. Lee and Behzad Razavi, "A stabilization technique for phase-locked frequency synthesizers," *IEEE J. Solid-State Circuits,* vol. 38, no. 6, pp. 888-894, June, 2003.

[3] S.J. Kim, K.H. Lee,Y.S. Moon, D.K. Jeong, Y.H. Choi, and H.K. Lim, "A 960-Mb/s/pin interface for skew-tolerant bus using low jitter PLL," *IEEE J. Solid-State Circuits,* vol. 32, no. 5, pp. 691-700, May 1997.

[4] S. Joen, T. Cheung, and W. Choi, "Phase/frequency detector for high-speed PLL applications," *IEEE Electronics,* vol. 34, no. 22, pp. 2120-2121, Oct. 1998.

[5] K.-S. Lee, B.-Ha Park, H.-il Lee, M. J. Yoh, "Phase-frequency detectors for fast frequency acquisition in zero-dead-zone CPPLLs for mobile communication systems," *Solid-State Circuits Conference, 2003. ESSCIRC '03 Proceedings of 29th European*, pp 525-528, Sept. 2003.

[6] M. Mansuri, D Liu, C.-K. Ken Yang, "Fast frequency acquisition phase-frequency detectors for Gsamples/s phase-locked loops," *IEEE J. Solid-State Circuits,* vol. 37, no. 10, pp. 1331-1334, Oct. 2002.

[7] G. Y. Tak, S.-B. Hyun, T. T. Kang, B. G. Choi, S. S. Park, "A 6.3–9-GHz CMOS Fast Settling PLL for MB-OFDM UWB Applications," *IEEE J. Solid-State Circuits,* vol. 40, no. 8, pp. 1671-1679, Aug. 2005.

[8] F. Centurelli, S. Costi, M. Olivieri, S. Pennisi, A. Trifiletti, "Robust three-state PFD architecture with enhanced frequency acquisition capabilities," *ISCAS Proc. 2004*, vol. 4, pp. 812-815, May 2004.

[9] B. Chang, J. Park, and W. Kim, "A 1.2GHz CMOS dual-modulus prescaler using new dynamic D-flip-flops," *IEEE J. Solid-State Circuits,* vol. 31, pp. 749-752, May 1996.

[10] Ran Singh Rana, "Dual-Modulus 127/128 FOM Enhanced Prescaler Design in 0.35-μm CMOS Technology," *IEEE J. Solid-State Circuits,* vol. 40, no. 8, pp. 1662-1670, Aug. 2005.

[11] B. Miller and B. Conley, "A multiple modulator fractional divider," *IEEE Trans. on Instrum. Meas.,* vol. 40, no. 8, pp.578-583, June. 1991.

[12] T. A. Riley, M. Copeland, and T. Kwasniewski, "Delta-sigma modulation in fractional-N frequency synthesis," *IEEE J. Solid-State Circuits,* vol. 28, no. 5, pp. 553-559, May 1993.

[13] W. Rhee, B.-S. Song, and A. Ali, "A 1.1-GHz CMOS fractional-N frequency synthesizer with a 3-b third-order ΔΣ modulation," *IEEE J. Solid-State Circuits,* vol. 35, no. 10, pp. 1453-1460, Oct. 2000.

[14] 高曜煌，射頻鎖相迴路IC設計，滄海書局，2005 年 10 月.

[15] M. Soyer, R. G. Meyer, "Frequency limitations of a conventional phase-frequency detector," *IEEE J. Solid-State Circuits,* vol. 25, no. 4, pp. 1019-1022, Aug. 1990.

[16] R. Rogenmoser and Q. Huang, "An 800-MHz 1-μm CMOS pipelined 8-b adder true-single clocked logic-flip-flops," *IEEE J. Soiid-State Circuits,* vol. 31, pp. 401-409, Mar. 1996.

[17] C. Y. Yang, G. K. Dehng, J. M. Hsu, and S. I. Liu, "New dynamic flip-flops for high-speed dual-modulus prescaler," *IEEE J. Solid-State Circuits,* vol. 33, pp. 1568-1571, Oct. 1998.

[18] B. Chi and B. Shi, "An optimized structure CMOS dual-modulus prescaler using dynamic circuit technique," in *Proc. IEEE Region 10 Conf. Computers, Communications, Control and Power Engineering,* pp. 1089-1092, Oct. 2002.

[19] J. Yuan and C. Svensson, "High-speed CMOS circuit technique," *IEEE J. Solid-State Circuits,* vol. 24, no. 1, pp. 62-70, Feb. 1989.

[20] D.B. Leeson, "A simple model of feedback oscillator noise spectrum," *Proc. IEEE,* vol. 54, pp. 329-330, Feb. 1996.

[21] J. Craninckx and M. Steyart, "Low-noise voltage-controlled oscillators using enhanced LC-tanks," *IEEE Trans.,* vol. 42, no. 12, pp. 794-804, Dec. 1995.

[22] A. Hajimiri and T. H. Lee, "A general theory of phase noise in electrical oscillators," *IEEE J. Solid-State Circuits,* vol. 33, no. 2, pp. 179-194, Feb. 1998.

[23] A. Hajimiri and T. H. Lee, "Phase noise in CMOS differential LC oscillators," *Proc. VLSI Circuits,* vol. 11-13, pp. 48-51, June 1998.

[24] S. Levantino, C. Samori, A. Bonfanti, S.L.J. Gierlink, A.L. Lacaita, and V. Boccuzzi, "Frequency dependence on biased current in 5-GHz CMOS VCOs: impact on tuning range and flicker noise upconversion," *IEEE J. Solid-State Circuits,* vol. 37, no. 8, pp. 1003-1011, Aug. 2002.

[25] C. Lam and Behzad Razavi, “A 2.6-GHz/5.2-GHz frequency synthesizer in 0.4-μm CMOS technology,” *IEEE J. Solid-State Circuits*, vol. 35, no. 5, pp.788-794, May 2000.

[26] E.J. Hernandez and S.A. Diaz, “Positive feedback CMOS charge-pump circuits for PLL applications,” *Proc. on Circuits and Systems, MWSCAS,* vol. 2, pp. 14-17, Aug. 2001.

[27] R.C. Chang and L.C. Kuo, “A new low-voltage charge pump circuit for PLL,” *IEEE, International Sym. of Circuits and Systems, ISCAS*, pp. 701-703, May 2000.

[28] C.S. Vaucher,” An adaptive PLL tuning system architecture combining high spectral purity and fast settling time,” *IEEE J. Solid-State Circuits,* vol. 35, no. 4, pp. 490-502, April 2000.

[29] L.C. Liu and B.H. Li, “Fast locking scheme for PLL frequency synthesizer,” *IEEE Electronics,* vol. 40, no. 15, May. 2004.