

---

# **Building Blocks for A Wide-Band Phase-Locked Loop**

by Zhiming Deng

---

## **Research Project**

Submitted to the Department of Electrical Engineering and Computer Sciences,  
University of California at Berkeley, in partial satisfaction of the requirements for the  
degree of **Master of Science, Plan II**.

Approval for the Report and Comprehensive Examination:

### **Committee:**

---

Professor Ali M. Niknejad  
Research Advisor

---

(Date)

\* \* \* \* \*

---

Professor Robert G. Meyer  
Second Reader

---

(Date)



# Acknowledgment

As with any endeavor, this research project could not have been completed without the help and support of others. Foremost, I wish to thank my research advisor Prof. Ali Niknejad. Your guidance throughout the course of this project is greatly appreciated. I would also like to thank Professor Robert Meyer for reviewing this thesis.

I am indebted to many present and past members of my research group: Axel Berny, Mounir Bohsali, Yun Hui Chee, Wei-Hung Chen, Mohan Dunga, Gang Liu, Nuntachai Poobuapheum, Stanley Wang, Mark Chew, Eddie Ng and Hanching Fuh.

Finally, I wish to thank my friends and family for their support and encouragement. I am very lucky to have a wonderful group of friends here in Berkeley, and I will never forget our happy days in Chinese People Union. Mom and Dad, thank you for your continual love and support throughout my life. I dedicate this work to the both of you!



# TABLE OF CONTENTS

CHAPTER 1 Wide-Band PLL Introduction .....	1
1.1 Wireless communication applications .....	2
1.2 Design specifications .....	4
1.3 PLL building blocks .....	5
CHAPTER 2 Frequency Divider .....	11
2.1 Overall divider architecture .....	11
2.2 High frequency divider .....	12
2.3 Programmable divider .....	19
2.4 Simulation result .....	24
CHAPTER 3 Sigma-Delta Modulator .....	25
3.1 General theory of Sigma-Delta modulator .....	26
3.2 Mash 1-1-1 Sigma-Delta modulator .....	30
CHAPTER 4 Band-Searching Scheme .....	35
4.1 Principle of band-searching scheme .....	36
4.2 Simulation of PLL with band-searching scheme .....	39
CHAPTER 5 Conclusion .....	41

REFERENCE ..... 45

# Chapter 1

## Wide-Band PLL Introduction

Phase-locked loops (PLL) have become essential components in wireless communication systems. They are used as frequency synthesizers with precise and convenient digital control in both traditional electronics, such as televisions and AM/FM radios, and modern consumer products among which cellular mobile phone is a striking example.

IC fabrication technology advances have made monolithic integration possible. More and more electronic devices can be put on the same chip to reduce the number of external components and then the costs. Therefore, on a single chip we can accomplish many functions for which we might need to make several chips work together a few years ago. A monolithic wide-band PLL is of great interests to wireless communication applications due to both its low cost and convenience to switch between different communication standards. The focus of this work is to implement a monolithic wide-band PLL using as few as possible building blocks and also as simple as possible structure.

## 1.1 Wireless Communication Applications

New radio frequency (RF) wireless communication applications have been emerging in these years due to technology improvements. These applications include cordless phones, cellular phones, global positioning systems (GPS), wireless local area networks (WLAN) and personal communication systems (PCS). They rely on both devices and communication standards as well. A communication standard tells a device how to talk to each other. For every application numerous standards exist which are optimized for different implementations. Standard examples include AMPS, GSM, DECT and Bluetooth.

Two basic building blocks are used in all RF systems: transmitters and receivers. Figure 1.1 shows the block diagrams of an RF front end with a transmitter and receiver respectively. The local oscillators in the block diagrams are usually frequency synthesizers. In a frequency synthesizer a PLL is used to lock the voltage-controlled oscillator (VCO) to the desired frequency precisely. A short review of PLL building blocks will be given later in the chapter.

Each Communication standard has its own complete set of performance requirements on the transceivers and some of them are design specifications for the PLL [Vaucher02], a very important part in the RF transceiver. First is the tuning range, which denotes the range of frequencies to be synthesized. It can be used to classify a PLL as being either narrow-band or wide-band. A coarse criteria can be used: a narrow-band PLL is characterized by a ratio  $f_{out,max} / f_{out,min} \cong 1.5$ , with  $f_{out,max}$  the maximum output frequency and  $f_{out,min}$  the minimum output frequency; a wide-band

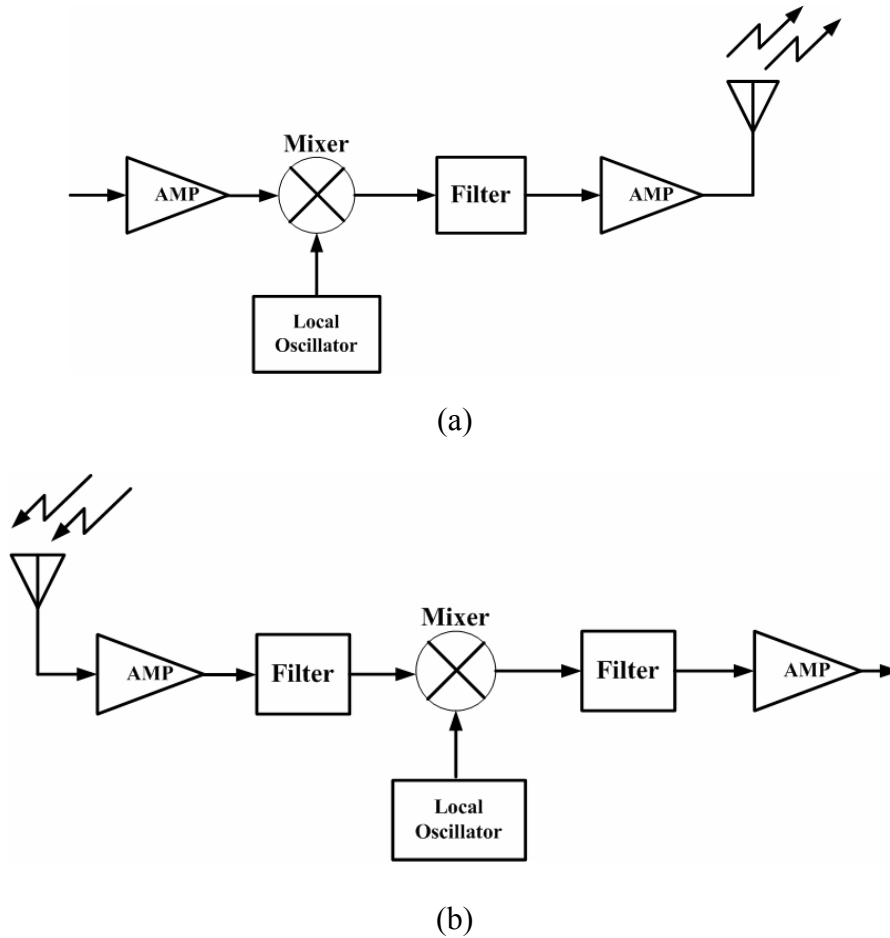


Figure 1.1 Block diagram of a typical RF front end: (a) transmitter, and (b) receiver.

PLL then has ratio of  $f_{out,max} / f_{out,min} > 1.5$ . Second is the frequency resolution or minimum frequency step size. It is very often equal to the frequency difference between two adjacent RF channels. The required minimum step size value varies widely with the application from 1 kHz to 1 MHz. The third specification is settling time, which is the time necessary for the tuning system to settle within a frequency window delimited by  $\pm f_{error}$  from the desired frequency  $f_{lock}$  after a “change frequency” command has been received. The settling time must be smaller than the locking time  $t_{lock}$ , and  $t_{lock}$  is often defined for the maximum frequency step, which is the approximately the tuning range. The last is the spur and phase noise requirement.

Smaller spurious signals and higher side-band phase noise roll-off rate can alleviate both direct mixing and reciprocal mixing, which might bury the useful signal.

## 1.2 Design Specifications

The desired wide-band PLL should be able to cover different standards with a large frequency range. Our design will be based on a wide-band VCO [Berny04]. And since the VCO shows an output frequency range of 1.14-2.47 GHz, we will try to design the PLL to cover all digital cell phone standards including TDMA IS-54/IS-136, CDMA, GSM, DC1800/DC1900, PDC.

We can now obtain the PLL specifications and it is listed in Table 1.1.

Table 1.1 PLL design specifications

PLL Specifications:	Requirements
Output Frequency	600MHz ~ 2.7 GHz
Reference Frequency	40 MHz
Frequency Resolution	25 kHz
Settling Error	0.1ppm
Settling Time	< 100 $\mu$ s
Phase Noise	GSM Specifications

## 1.3 PLL Building Blocks

A conceptual block diagram of a typical PLL is shown in Figure 1.2. It consists of five blocks: VCO, feedback frequency divider, phase frequency detector (PFD) / charge pump (CP), low-pass loop filter (LPF) and crystal oscillator (Xtor).

Comparing to other RF front-end circuits, PLL is a far more complicated system, and a combination of many individual small blocks. It is worth of discussing more details and to examine each block carefully.

### 1.3.1 LC VCO

An LC VCO generates the output signal of the PLL. The frequency of the VCO signal is dependent on the external control signal at the input. For a mixed-signal tuning VCO [Berny04], there are both weighted capacitor array for digital coarse tuning and MOS varactors for continuous fine tuning. The output frequency expression has the form of  $f_{out} = f_{center} + f_{digit}(D_{control}) K_{VCO}(V_{tune}) V_{tune}$ , where  $D_{control}$  is the digital control command word with each word representing a discrete frequency bank and  $K_{VCO}$  is called the gain factor in [Hz/V]. The advantage of using such a mixed-signal tuning VCO is that it decouples the needs for both wide tuning range and low  $K_{VCO}$ , which will be explained more thoroughly when we analyze the phase noise performance later.

A time domain formula (1.1) shows the phase of the VCO output signal is a perfect integration of the control signal  $V_{tune}$ . Here we drop the terms containing the center frequency and the digitally controlled frequency bank since these are not

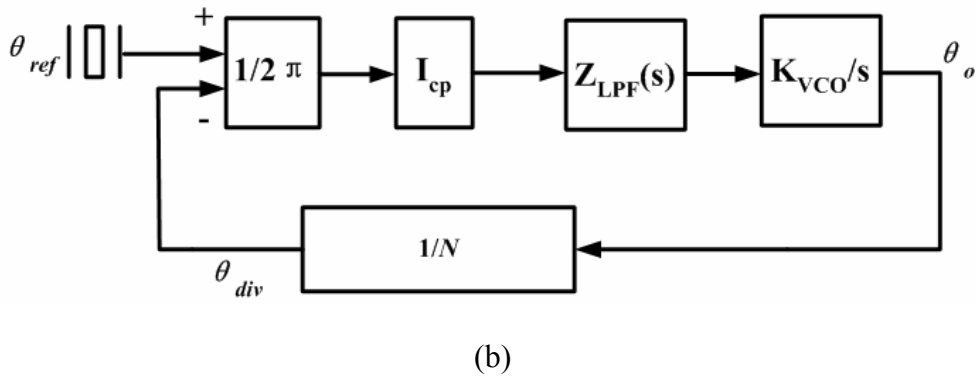
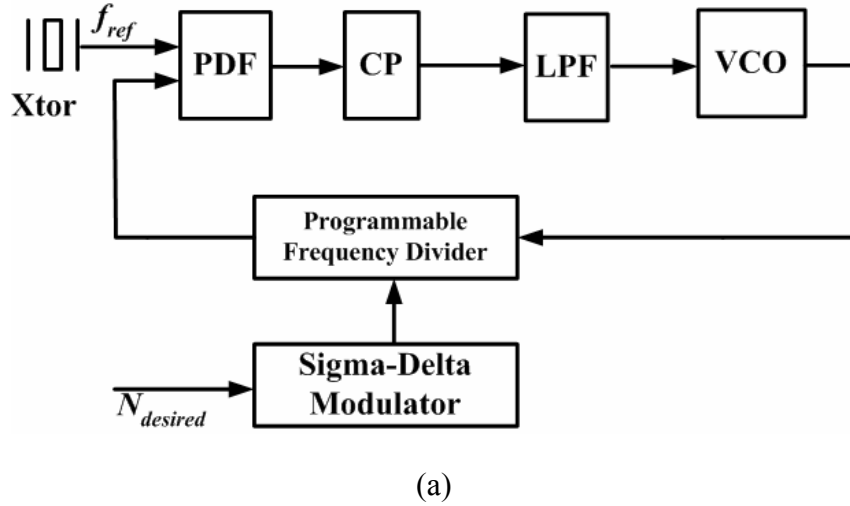


Figure 1.2 A single-loop PFD/CP Fractional-N PLL: (a) Block diagram and (b) frequency domain model

presented in the continuous tuning loop.

$$\theta_o = \int 2\pi \cdot f_{out}(t) dt \quad (1.1)$$

Taking the Laplace transform of (1.1) yields a frequency domain representation

$$\theta_o(s) = \frac{2\pi \cdot K_{vco} \cdot V_{tune}(s)}{s} \quad (1.2)$$

### 1.3.2 Frequency Divider

The frequency divider is responsible for frequency scaling within the loop. The input frequency  $f_{in}$  is divided by some programmable number  $N$ , which is an integer in

an integer-N PLL and can contain a fractional part in a fractional-N PLL, and the output frequency is  $f_{div}$ . A frequency domain description of the function of a frequency divider is

$$\theta_{div}(s) = \frac{\theta_{in}(s)}{N} \quad (1.3)$$

As the VCO output signal has higher and higher frequency, the high-frequency capability of the frequency divider is attracting more attention. To achieve both high-frequency capability and divider number programmability, a complete frequency usually consists of two parts: high frequency divider-by-2 divider and low frequency programmable digital divider. The former divides the high frequency VCO signal down to low frequency signal, which can be handled by the latter.

### 1.3.3 PFD and CP

Signal at the output of the frequency divider should be in the range comparable to the reference frequency. Then a PFD is used to compare the phase difference of  $f_{div}$  and  $f_{ref}$  and generates an error signal which is proportional to their phase difference.

A CP converts the error signal from voltage to current pulses, with the mean value also proportional to the phase difference.

Though the combination of them has a lot of practical design issues need to be taken into account when a real circuit is to be built, the frequency domain transfer function is very simple, shown in (1.4):

$$I_{out} = I_{cp} \frac{\theta_{ref} - \theta_{div}}{2\pi} = K_{pd} (\theta_{ref} - \theta_{div}) \quad (1.4)$$

$I_{cp}$  is the nominal CP current. This formula shows the function of the combination of PFD and CP in frequency domain is just a linear gain stage with combined gain  $K_{pd}$ .

### 1.3.4 LPF

The output of the CP cannot be directly connected to the tuning node of the VCO for two reasons. First, it is a current output signal, and without any path to ground it will charge the input capacitor of the tuning node to infinity eventually. Second, it is a pulse instead of a constant. Fundamental and its harmonics will affect the VCO output by phase modulation.

Therefore a LPF is necessary here to extract the mean value from the output pulse. There are many implementations of LPF with each of them focusing on different aspects of the PLL performance. But we can use a transfer function  $Z_{LPF}(s)$  to describe the function of a LPF in frequency domain.

To investigate the phase locking phenomenon of the PLL more deeply, it is helpful to examine the feedback loop in frequency domain. Also a simple assumption is made that every block in the diagram is linear. Then we can derive the open loop transfer function  $G(s)$  and the close loop transfer function  $H(s)$  of the PLL respectively. See

$$G(s) = K_{pd} \cdot Z_{LPF}(s) \cdot \frac{2\pi \cdot K_{VCO}}{s} \cdot \frac{1}{N} \quad (1.5)$$

$$H(s) = \frac{G(s)}{1 + G(s)} = \frac{2\pi K_{pd} Z_{LPF}(s) K_{VCO} / N}{s + 2\pi K_{pd} Z_{LPF}(s) K_{VCO} / N} \quad (1.6)$$

The loop bandwidth of the PLL is defined as the frequency  $f_b$  where  $|G(j2\pi f_b)| = 1$ .

The dynamic behavior and the spectral purity are highly dependent on the  $f_b$ .

In the following chapters, building blocks including the frequency divider, the Sigma-Delta modulator and the band-searching scheme will be discussed.



# Chapter 2

## Frequency divider

In this chapter, frequency divider implementations will be discussed. Efforts will be put on optimize the high frequency capability while achieving programmable flexibility. With a single divider, it is difficult to accomplish these two functionalities simultaneously. Thus a multi-stage divider will be built with high frequency divide-by-2 divider followed by a programmable low frequency divider.

Source-coupled logic (SCL) is used in the high frequency divider design due to its both high frequency capability and wide frequency range. The phase-select approach is used in the programmable divider due to its higher operating speed [Craninckx96].

Also in order to achieve lower power operation an asynchronous approach is used. This minimizes the number of devices working at high frequency.

### 2.1 Overall divider architecture

Depending on the VCO, we have an output frequency range of 1.3 GHz ~ 2.7 GHz. And we choose the reference frequency to be 40 MHz. so the division ratio  $N_{div} = f_{out} / f_{ref} \approx 32 \sim 68$ . In order to cover the GSM standard, a divide-by-2 stage following the VCO is necessary, so the rest division ratio  $N_{div}' = N_{div} / 2 = 16 \sim 34$ . A programmable

divider then follows the high frequency divide-by-2 stage. It has a  $P / P+1$  prescaler and an  $A / B$  accumulator, whose detail designs will be covered in section 2.3. So we have  $N_{div}' = A + BP$ . To guarantee continuous  $N_{div}'$ , we must have  $N_{div}'_{(min)} = P(P-1) \leq 16$ , i.e.  $P \leq 4$ . Making  $P = 4$ , the ranges of A and B are 0~3 and 3~8 respectively. Eventually we get a division range of 24~70 with a step of 2.

One more thing to notice is that a 4/5 prescaler has 2 stages of divide-by-2 divider, here they are both categorized together with the first high speed divider into a divide-by-8 divider and the 4/5 prescaler becomes a 1/1.25 prescaler.

## 2.2 High frequency divider

A high frequency divider in a wide band PLL system needs high frequency capability as well as wide frequency range. Injection locked frequency dividers have been proposed due to their high frequency capability [Rategh99]. However, the drawback of narrow input frequency range or locking range makes it unsuitable in this wide band system design. Therefore, a DFF-based frequency divider has been chosen to be the high frequency divider design. Figure 2.1 shows the block diagram of DFF-based divide-by-2 frequency divider. The D Flip-Flop consists of two D-Latch, which are driven by the same clock signal but with opposite phases. The inverted output of the DFF is fed back to the input to generate waveform with half of the input frequency.

Though we have chosen the DFF-based structure, the logic style to implement the D-Latch and other accessory circuits is still to be decided. The major concern is high

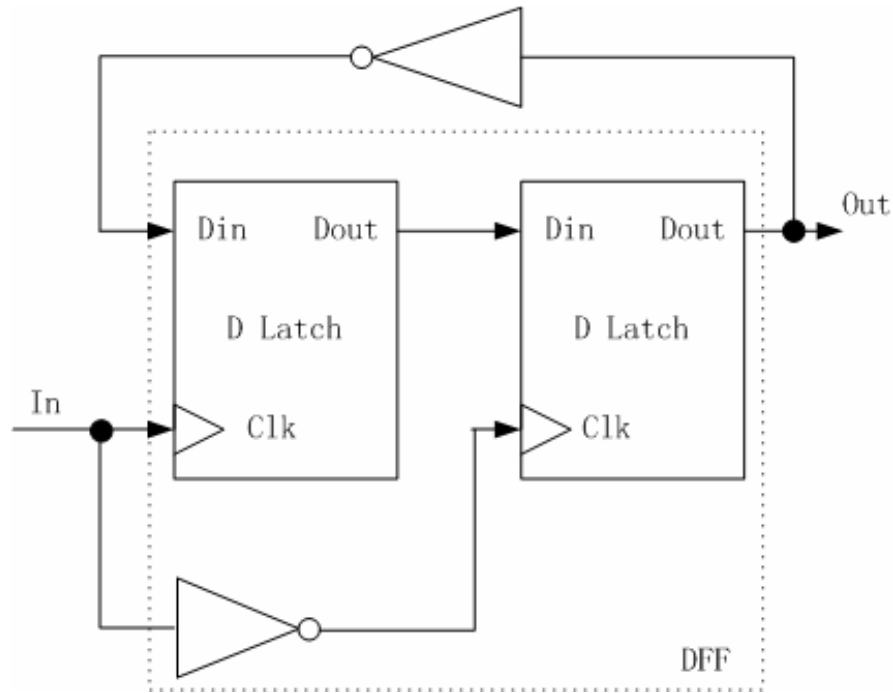


Figure 2.1 Block diagram of a DFF-based divide-by-2 divider

speed. Without using the normal complementary CMOS logic style, very fast logic style, SCL, has been used to successfully implement a fast frequency divider up to very high frequency [Wohlmuth02]. Besides high frequency capability, we also benefit from its low noise level coupled to the substrate, which will be shown later. Furthermore, the differential nature of SCL also reduces the complexity of the logic implementation. For example, one can have signals with both phases simultaneously without the use of an inverter.

### 2.2.1 Elementary SCL cell design

A buffer is the elementary SCL cell. The approach used in its design is basis for the designs of more complex logics. It is worth more attention. Figure 2.2 gives a

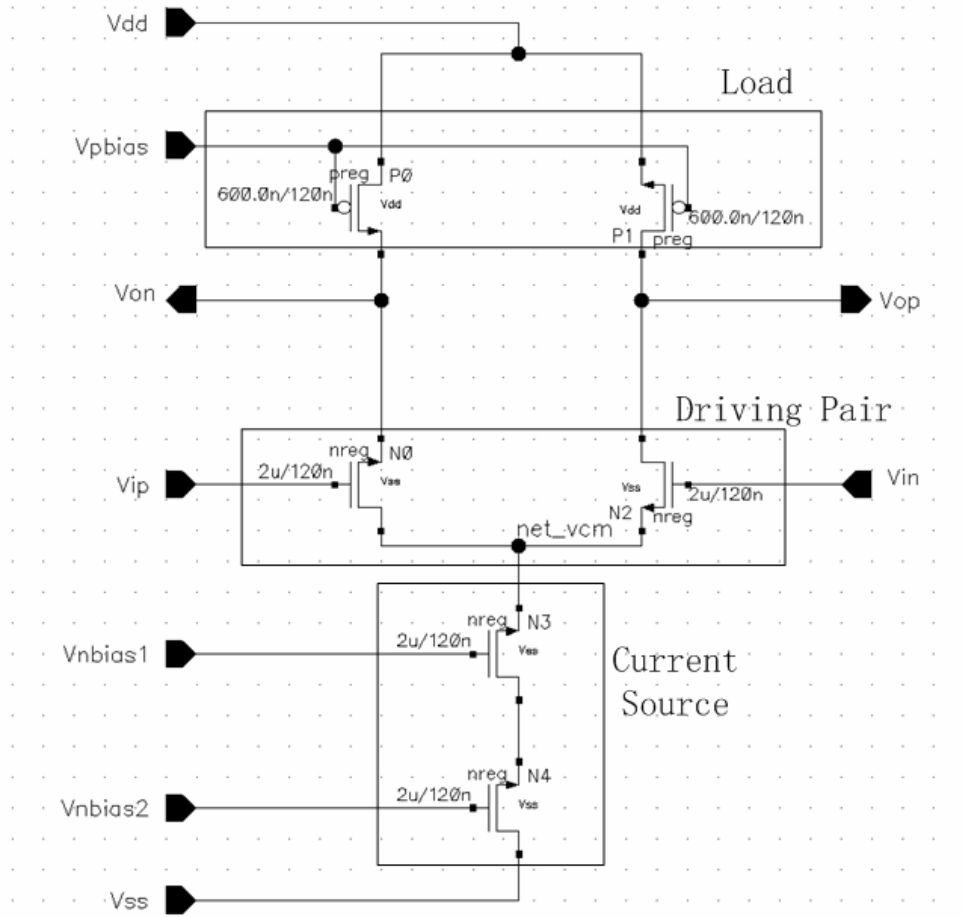


Figure 2.2 Schematic of a SCL buffer designed for 3 GHz

schematic of an SCL buffer. The main concerns of this circuit include propagation delay, power consumption, and output voltage swing, while taking bias current variation into consideration. As can be seen from the schematic, an SCL buffer consists of three parts: bias current source, the NMOS driving pair and the last is the PMOS load. We will look at the detail design considerations and tell how each part contributes relative to our major concerns mentioned above.

The propagation is the most important specification of the design. It consists of two aspects: the falling delay and the rising delay. The falling delay is determined by the discharging current, which is the bias current. And the rising time is determined by

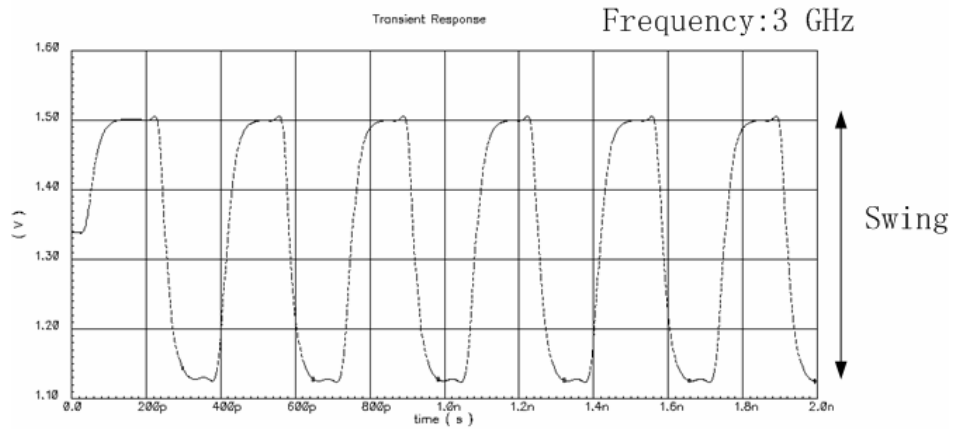
the charging current from the supply through the PMOS load transistor, which can be controlled through the external gate bias voltage.

Power consumption is decided by both supply voltage and bias current. The lower the supply voltage, the lower is the power consumption. However, using too low a supply voltage will make it very hard to put more transistors in stack when more complex logics are required. For bias current, we can pick up the smallest bias current that satisfies the timing requirement.

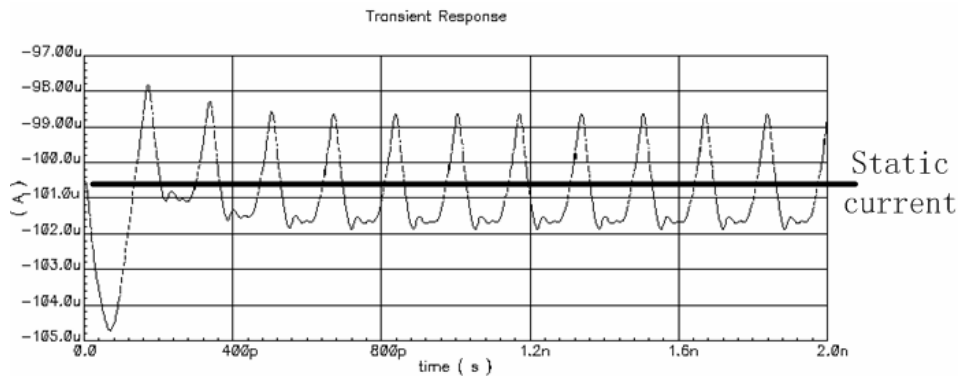
The voltage swing can determine the dynamic power and also the sensitivity to noise. In this topology, the maximum output is always the supply voltage. And the minimum output level is decided by the product of the bias current and the resistance of the load PMOS transistor.

One of the most significant advantages of SCL over complementary CMOS logic is its low transition current spike. To make the bias current source more like an ideal source, or to make the output resistance of the current source larger, two transistors in stack are used. To completely switch the bias current from branch to the other, the output swing cannot be too low. Or equivalently speaking, the driving pair has to be strong enough to have low enough turn off voltage,  $V_{dsat}$ .

Figure 2.3 shows the transient response of an SCL buffer driving by a 3 GHz signal. It has a voltage swing of less than 0.4 V and a bias current of about 101  $\mu$ A. And the bias current transient variation is less than 2%, which is measured from the peak of the swing from the static value



(a)



(b)

Figure 2.3 Transient response of an SCL buffer: (a) Output voltage with swing less than 0.4V, (b) bias current with static value of 100  $\mu$ A and variation less than 2%

## 2.2.2 SCL D-Latch and D Flip-Flop

Based on the design of an SCL buffer, we can build an SCL D-Latch with the same approach. See Figure 2.4 for the schematic. When the clock signal is at high state, it works like a buffer. When the clock signal is low, the cross-coupled pair holds the existing state by positive feedback principle.

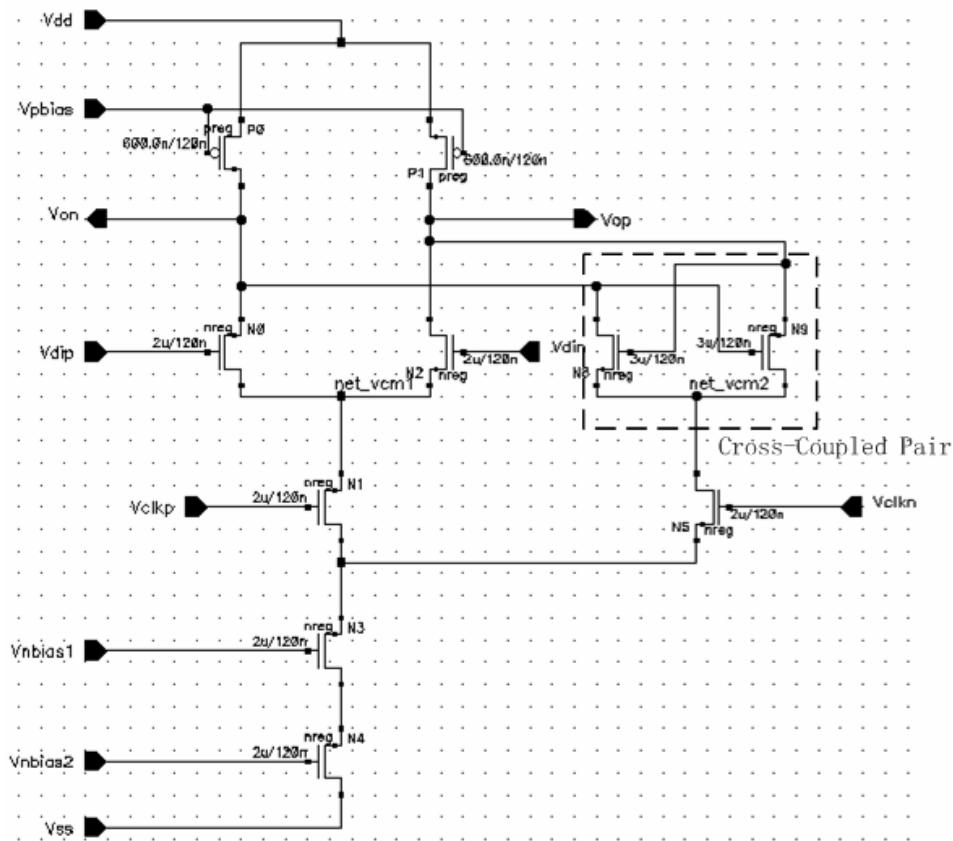


Figure 2.4 Schematic of a SCL D-Latch

Figure 2.5 gives the transient plot of the D-Latch when it is driven by a 3 GHz clock signal and 1.5 GHz data signal.

Then a DFF can be constructed by cascading two equivalent D-Latches. With the inverted output signal fed back to the data input. The high speed property of this topology comes from the differential nature of the SCL. No additional inverters are needed and hence the number of stages as well as the propagation delay is reduced. Figure 2.6 shows the transient response of the DFF-based divide-by-2 divider. The output provides four phases of output signal, which is helpful to the phase selection method for programmable divider. This will be covered later in this chapter.

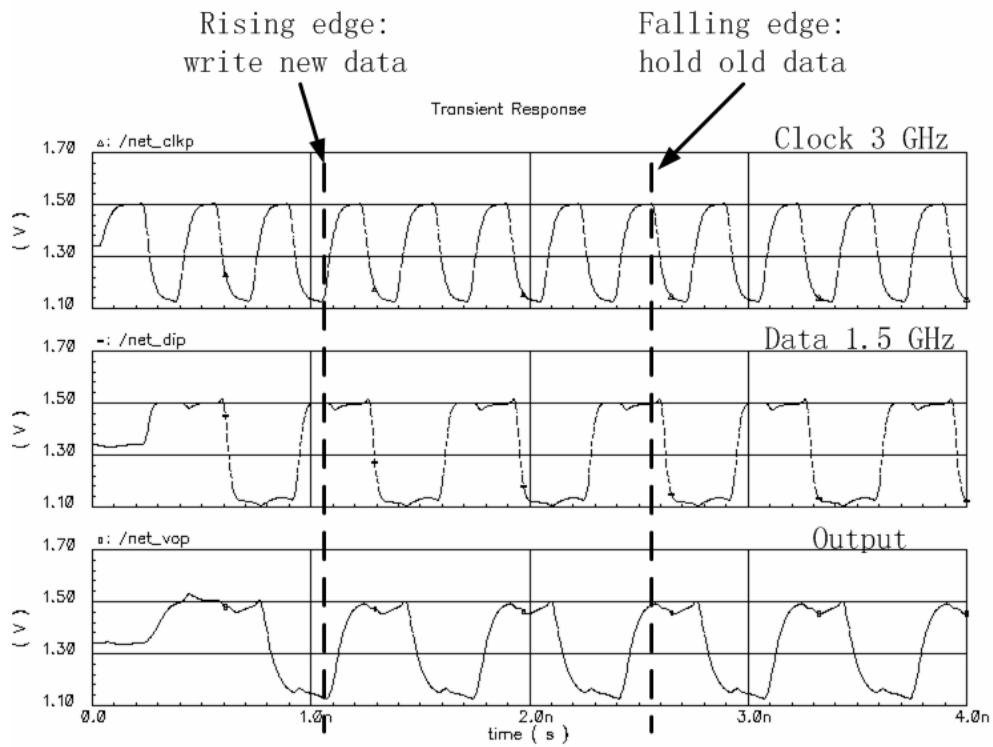


Figure 2.5 Transient response of an SCL D-Latch

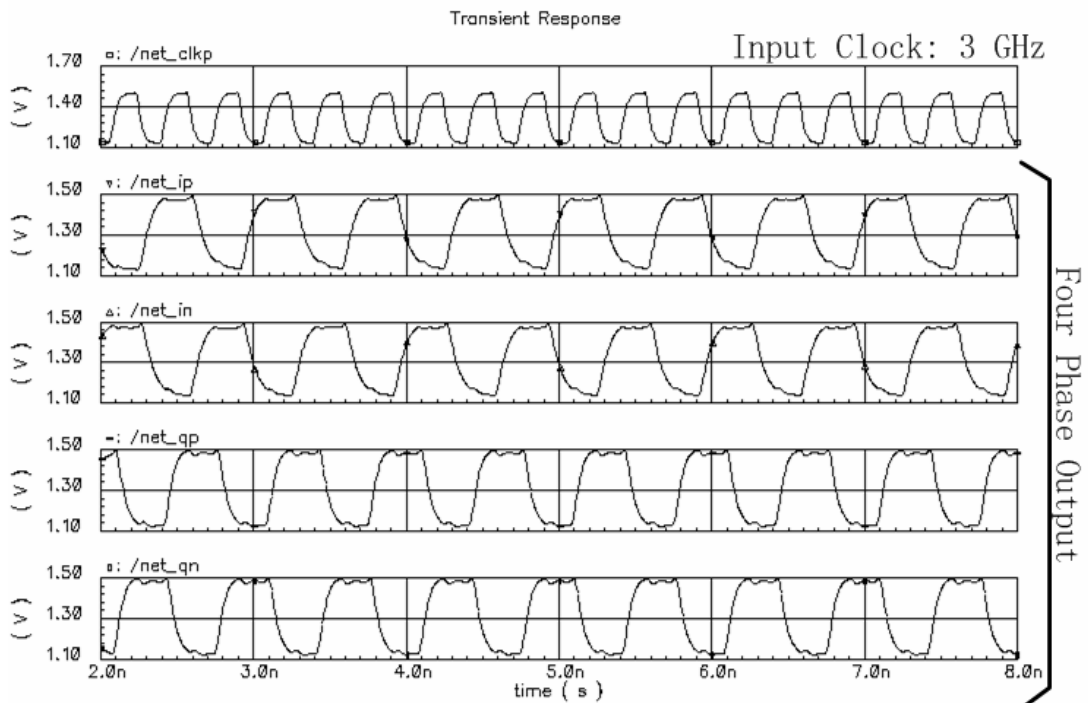


Figure 2.6 Transient response of an SCL DFF-based divide-by-2 divider

### 2.2.3 SCL high frequency divide-by-8 divider

The schematic of the divide-by-8 divider is shown in Figure 2.7. It consists of three cascaded divide-by-2 divider with each has the same structure and the preceding buffer is used to convert the VCO output voltage level to the SCL voltage level.

The output frequency of the divide-by-8 divider will be 1/8 of the VCO output frequency. And its output is then fed to the input the programmable divider.

## 2.3 Programmable divider

The foundation of a programmable divider is a dual-modulus prescaler and an accumulator. The prescaler can divide the input signal frequency by  $N$  or  $N+1$  depending on a control signal. And this control signal is from the accumulator. The accumulator has two data input,  $A$  and  $B$ , which satisfy the relationship  $A \leq B$ . The accumulation starts at  $0$  and with the control signal's initial value of logic "0", which means the divider number is  $N+1$ . When the accumulator reaches  $A$ , the control signal turns into logic "1" meaning a divider number of  $N$ . This operation lasts until the accumulator counts to  $B$ , then the accumulator goes back to  $0$  and the control signal becomes logic "0" again. If the output of the accumulator is used as the total divider output, then the total divider number is  $N_{\text{tot}} = A(N+1) + (B-A)N = A + BN$ . Figure 2.8 is a conceptual diagram of the programmable divider.

### 2.3.1 Phase-select divide-by-1/1.25 prescaler

Phase-select-based dual modulus prescaler has the advantages of higher

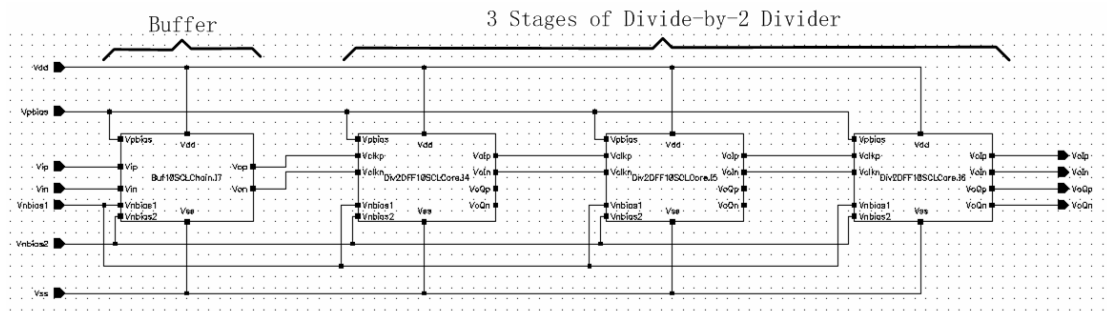


Figure 2.7 Divide-by-8 high frequency divider

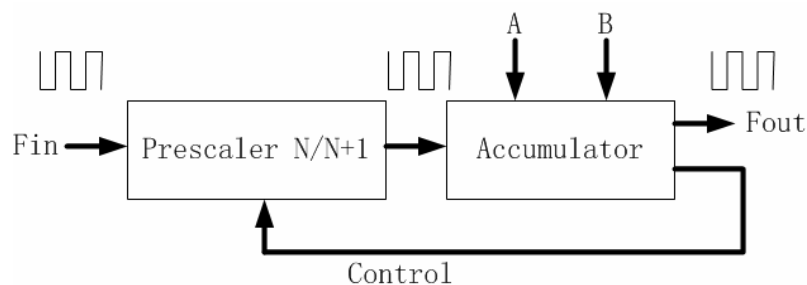


Figure 2.8 Conceptual diagram of the programmable divider

maximum input frequency and only the first stage of the circuit working at the maximum frequency comparing to the traditional gating technology [Craninckx96].

Figure 2.9 shows a diagram of a divide-by-1/1.25 prescaler. The four input signals are at the same maximum frequency but with phase shift. When the “Mode” signal keeps low, the output keeps connecting to only one input, and we get a divide-by-1 operation. When the “Mode” signal keeps high, at each positive edge of the output signal, the control signal will change the connection of the output from the present input to the one with 90 degree delay. Then the divider number becomes 1.25.

As can be shown in Figure 2.9, the clock signal of the phase selection control logic comes from the output of the phase selection. For low input frequencies, the risk of creating spikes does exist [Craninckx96]. This comes from the transition from one

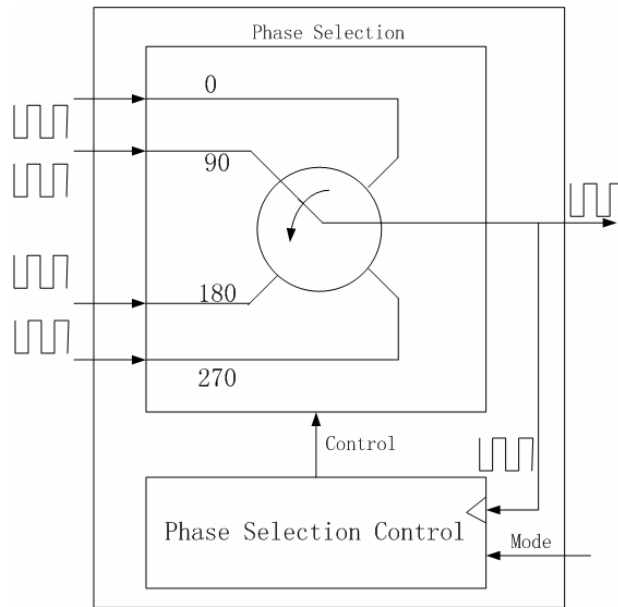


Figure 2.9 Diagram of divide-by-1/1.25 prescaler

input to another is too fast. A proposed solution, called “parallel phase selection”, is shown in Figure 2.10. The difference is that two parallel phase selection blocks are used simultaneously, and the overall output is the OR operation of the two phase selection outputs. The two phase selection MUX’s do the transitions at different clock edges. The earlier one makes the transition at the positive edge and the other does at the negative edge of the clock. So if we look at the output waveform of the two MUX’s, when the earlier output connection changes at the positive clock edge, the other output does not change. Therefore, the output of the OR operation of them does not change, and the risk of spike is eliminated. One additional requirement for this approach is that the initial states of the two parallel selection logics must be set to connect to the same input at the beginning. A simple initial reset pulse can accomplish this task.

Figure 2.11 gives the schematic of and SCL multiplexer 4:1, and Figure 2.12

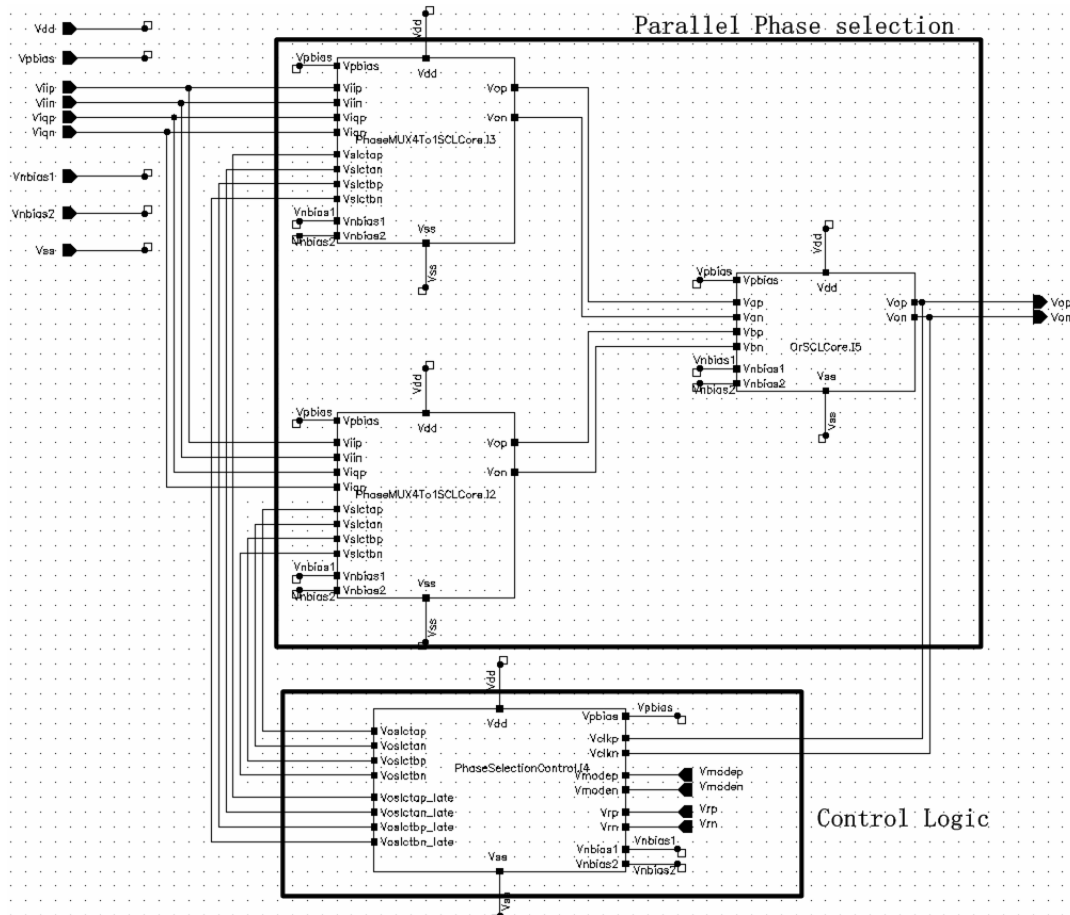


Figure 2.10 Parallel phase selection spike-free prescaler

shows the transient plot of the whole prescaler with the “Mode” signal keeping high, i.e. a quarter period of waveform is swallowed in every period.

### 2.3.2 Accumulator

Accumulation operation depends on two input parameters  $A$  and  $B$ . They come from the Sigma-Delta modulator. Within one output cycle,  $A$  and  $B$  are fixed. And the resulted divide number is  $A + BP$  with  $P = 4$ . To guarantee the range of integer divider number  $N_{div}$  from 16 to 34 continuously,  $A$  has the range from 0 to 3 and  $B$  has the range from 3 to 8. Then the total divider number range is 12 to 35. For each set of

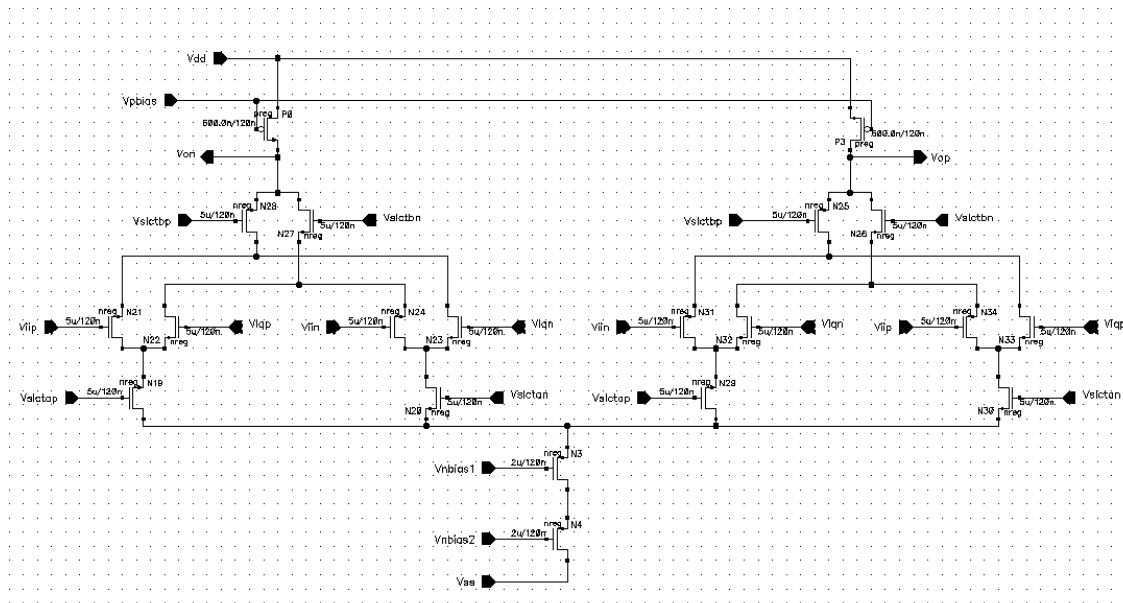


Figure 2.11 Schematic of a 4:1 multiplexer

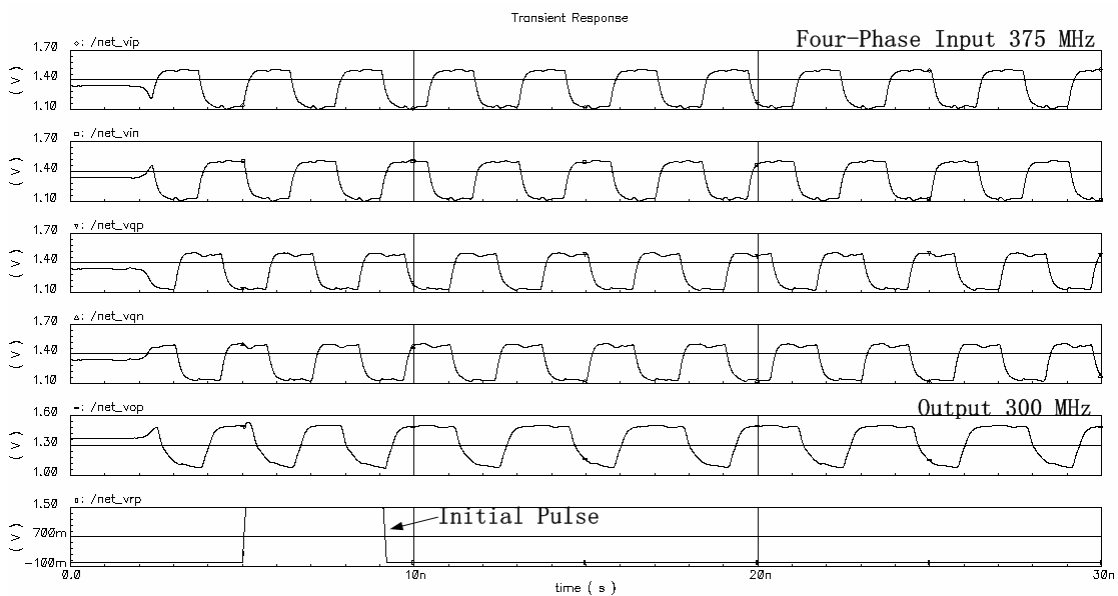


Figure 2.12 Transient response of the divide-by-1.25 operation

values of  $A$  and  $B$ , the accumulator starts from  $B-1$ , and after  $B$  cycles it reaches 0 and then is reset to  $B-1$  by a synchronous reset signal. If we take the output of accumulator

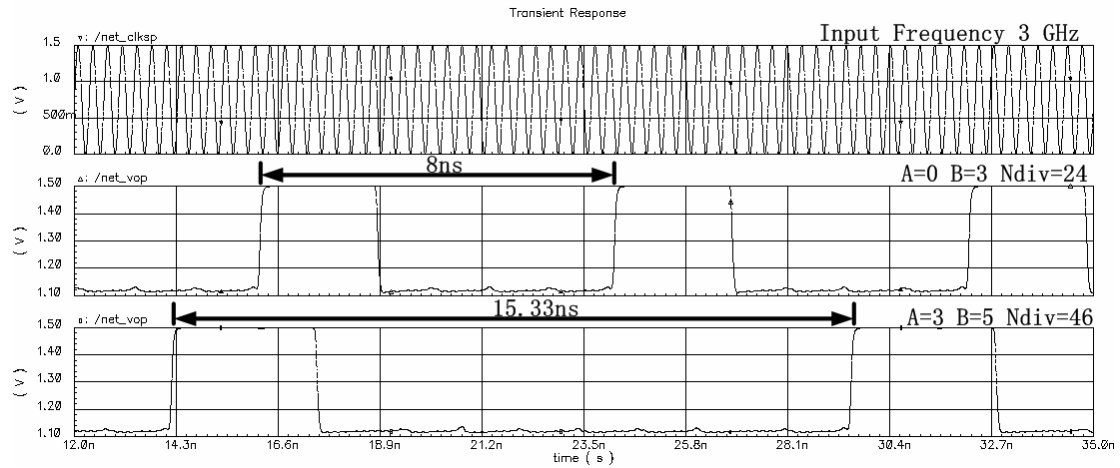


Figure 2.13 Divider transient output

and subtract it by  $A$ , the result has  $(B-A)$  cycles of positive sign and  $A$  cycles of negative sign. Then this sign signal can be used as the phase selection control signal and the output can use it too.

## 2.4 Simulation result

Combining the three parts: high frequency divide-by-8 divider, phase-select prescaler and the accumulator, we can have the whole divider. By setting different values of  $A$  and  $B$ , we can have different results. Figure 2.13 shows two situations with input frequency 3 GHz and  $A, B$  equal to 0, 3 and 3, 5 respectively.

# Chapter 3

## Sigma-Delta Modulator

In a simple integer- $N$  frequency synthesizer, there exists one major drawback: the frequency resolution, or the channel spacing, is equal to the reference frequency, meaning that only integer multiples of the reference frequency can be synthesized. Fine frequency resolution needs very small reference frequency. At the same time, stability requirements limit the loop bandwidth to about one tenth of the reference frequency. To ensure sufficient suppression of spurious signals at the reference frequency, the loop bandwidth needs to be even smaller. As a result, the dynamic behavior of the PLL is seriously degraded. Besides small reference frequency, fine resolution also requires large modulus  $N$ . Since the noise contributions of almost all PLL building blocks, except the VCO, are multiplied by  $N$ , the in-band noise of the PLL becomes unacceptable. In short, the design of integer- $N$  frequency synthesizers poses a severe trade-off between frequency resolution, PLL dynamic behavior and spectral purity – spurious as well as phase noise.

One of the most promising techniques to decouple the different PLL specifications and supply more degrees of freedom in frequency synthesizer design is fractional- $N$  synthesis. In fractional- $N$  synthesizers, fractional multiples of the reference frequency

can be synthesized, allowing a high reference frequency for a given frequency resolution. The principle behind the fractional-N frequency synthesis is that the modulus  $N$  to the frequency divider input is controlled to be changing periodically when the desired modulus has fractional number and is fixed. For example, the period has  $K$  cycles, and for each cycle the modulus is an integer  $N[k]$  with  $k$  from 1 to  $K$ . Then the fractional modulus  $N = \text{average}(N[k])$ . That means the resolution of modulus  $N$  in time can be exchanged for resolution of modulus  $N$  in amplitude, which is very similar to the idea behind a Sigma-Delta modulator.

In this chapter, techniques of Sigma-Delta modulators in fractional-N frequency synthesis will be discussed. A mash 1-1-1 Sigma-Delta modulator will be implemented for the wideband PLL.

### 3.1 General theory of Sigma-Delta modulator

Figure 3.1 shows a model of a Sigma-Delta modulator. It is a pure digital implementation, so its behavior can be formulated in both discrete-time domain or in  $z$ -transform domain. The sampling frequency is the reference frequency,  $f_{ref}$ , from the PLL. Besides the basic arithmetic operations such addition and subtraction, the model also include two functional blocks. The quantizer truncates the input signal and the output will only be the integer part. It is equivalent to adding a noise signal  $e(z)$  and the quantization error is  $-e(z)$ .  $J(z)$  is a given discrete-time filter, it takes the responsibility of shaping the quantization noise. The feedback will force the quantized output  $N_o$  to track the input  $N_i$ . In addition a shaped quantization error will

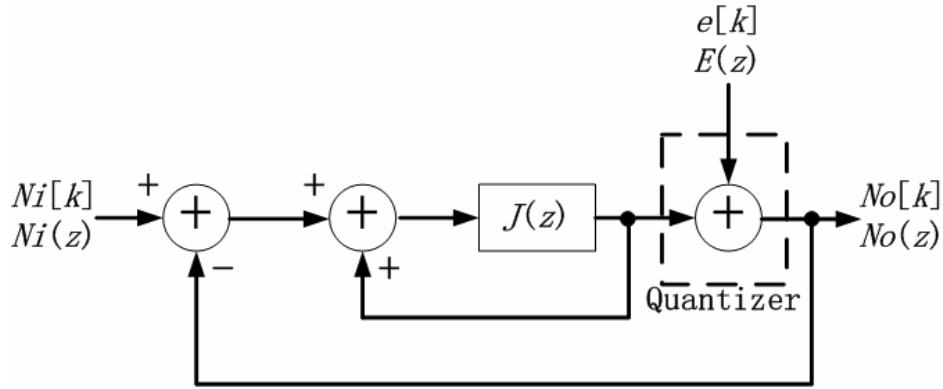


Figure 3.1 An equivalent model of a sigma-delta modulator

also be added to the  $N_o$ . By properly selecting the shaping function, which is related to  $J(z)$ , the shaped quantization noise can be high passed, i.e. most error energy is situated around  $f_{ref}/2$ .

The discrete-time domain expression for the output of the shown model is:

$$N_o(z) = J(z) \cdot N_i(z) + (1 - J(z)) \cdot E(z) \quad (3.1)$$

To guarantee the output can track the input signal, an additional requirement is needed for the  $J(z)$ .  $J(1)$  must be 1 at 0 frequency ( $z = 1$ ). Simultaneously the noise transfer function  $(1 - J(1))$  equals to zero. Equivalently speaking, the output exactly equals to the input in the average sense. Also  $(1 - J(z))$  must be a high pass shaping function.

If  $J(z) = z^{-1}$ , a unit cycle delay, the system is called a first-order Sigma-Delta modulator. In the following discussion, we use the first-order Sigma-Delta modulator as an example to discuss some general theory of Sigma-Delta modulator.

### 3.1.1 Noise transfer function and noise shaping

From (3.1) we see noise is shaped by the function  $(1 - J(z))$ , which is called the

noise shaping function. This filter can reduce the error  $E(z)$ . Especially at low frequencies a large reduction is obtained. Total amount of the quantization error is found by integrating  $(1 - J(z)) \cdot E(z)$  over the signal bandwidth  $f_b$ .  $f_b$  is also the PLL loop bandwidth. Here we assume that any signal outside  $f_b$  can be completely removed by the LPF presented in the loop. Then we have the integration result:

$$E_{tot}^2 = \int_0^{f_b} \left| E^2 \left( e^{j2\pi \frac{f}{f_{ref}}} \right) \cdot \left[ 1 - J \left( e^{j2\pi \frac{f}{f_{ref}}} \right) \right] \right|^2 df \quad (3.2)$$

The integration is in frequency domain,  $z$  is replaced by  $e^{j2\pi \frac{f}{f_{ref}}}$ . The quantization error  $E(z)$  is actually a deterministic signal for a given input. But the white noise assumption is still a good enough approximation for it. That means  $E^2(z)$  can be approximated as a constant,  $\frac{\Delta^2}{12}$ , over the interested bandwidth.  $\Delta$  is the weight of the least significant bit which is 1 for a integer quantizer. So here  $E^2(z) = 1/12$ .

Furthermore, we call the system an  $n$ th order system if  $(1 - J(z)) = (1 - z^{-1})^n$ . And we

also have  $\left| (1 - z^{-1})^n \right|^2 = \left| 1 - e^{j2\pi \frac{f}{f_{ref}}} \right|^{2n} = \left[ 4 \sin^2 \left( \pi \frac{f}{f_{ref}} \right) \right]^n$ . Combining the above two

assumption, (3.2) can be rewritten as

$$E_{tot}^2 = \frac{4^{n-1}}{3} \int_0^{f_b} \sin^{2n} \left( \pi \frac{f}{f_{ref}} \right) df \quad (3.3)$$

If the loop bandwidth  $f_b$  is much smaller than the reference frequency  $f_{ref}$ , (3.3) can be

simplified further by making  $\sin \left( \pi \frac{f}{f_{ref}} \right) \approx \left( \pi \frac{f}{f_{ref}} \right)$ . So we have

$$E_{tot}^2 \approx \frac{4^{n-1}}{3} \int_0^{f_b} \left( \pi \frac{f}{f_{ref}} \right)^{2n} df = \frac{4^{n-1} \pi^{2n}}{3(2n+1)} \left( \frac{f_b}{f_{ref}} \right)^{2n} f_b \quad (3.4)$$

Comparing (3.4) to case where no noise shaping is applied, we can see the equivalent resolution of the Sigma-Delta modulator output is

$$R = \sqrt{\left[ \frac{4^{n-1} \pi^{2n}}{3(2n+1)} \left( \frac{f_b}{f_{ref}} \right)^{2n} f_b \right] / \left( \frac{f_b}{12} \right)} = \frac{2^n \pi^n}{\sqrt{2n+1}} \left( \frac{f_b}{f_{ref}} \right)^n \quad (3.5)$$

Define *ENOB* to be the equivalent number of binary bits and *ENODB* to be the equivalent number of decimal bits. Then

$$ENOB = -\log_2 \frac{2^n \pi^n}{\sqrt{2n+1}} \left( \frac{f_b}{f_{ref}} \right)^n = 0.5 \log_2(2n+1) + n(\log_2 M - 2.65) \quad (3.6)$$

$$ENODB = -\lg \frac{2^n \pi^n}{\sqrt{2n+1}} \left( \frac{f_b}{f_{ref}} \right)^n = 0.5 \lg(2n+1) + n(\lg M - 0.798) \quad (3.7)$$

where  $M = \frac{f_{ref}}{f_b}$  is called oversampling ratio.

### 3.1.2 Sigma-Delta modulator with DC inputs

What we discussed above assumes the input of the Sigma-Delta modulator is busy enough, so that the white noise assumption for the quantization error signal won't deviate from the real situation too much. When the input of the Sigma-Delta modulator is connected to some fixed DC inputs, the white noise approximation can still be used, but some distinct phenomena will happen. The output will be a periodic signal, which is called idle pattern. The period of the idle pattern depends on the order of the Sigma-Delta modulator.

One way to break the idle pattern is move the Sigma-Delta modulator to higher order. This is because higher-order modulator will generate less quantization noise to the input due to (3.4). Also the randomization provided by higher-order modulator is

much better, since the predicting procedure of higher-order Sigma-Delta modulator is more spread over time, thereby decorrelating the succeeding noise samples. In short, the white noise assumption is more applicable.

### 3.2 MASH 1-1-1 Sigma-Delta modulator

We have talked about how the order of a Sigma-Delta modulator can affect the resolution  $R$ ,  $ENOB$  and  $ENODB$  in (3.5), (3.6) and (3.7) respectively. Now we reverse the problem, i.e. given the resolution requirement and find out what is the minimum order that can satisfy this requirement.

First, we have a frequency resolution or channel spacing specification of 25 kHz and settling error of 0.1ppm. So the resolution  $R$  satisfies

$$2 \times [(N + R) - N] \times f_{ref} \leq \min\{f_{spacing}, 10^{-7} \times f_{out}\} \quad (3.8)$$

$$R \leq \frac{1}{2} \left( \frac{10^{-7} \times f_{out}}{f_{ref}} \right) \approx \frac{1}{2} \left( \frac{10^{-7} \times 1.0 \times 10^9}{40 \times 10^6} \right) = 1.25 \times 10^{-6} \quad (3.9)$$

The factor of 2 in (3.8) comes from the fact that a divide-by-2 stage is in front of the continuous integer programmable divider. From  $R$  we can get  $ENOB$ :

$$ENOB = -\log_2(R) \approx 20 \quad (3.10)$$

That means 20 fractional bits are necessary to accurately represent the modulus at the input of the Sigma-Delta modulator.

The next step is to determine the order of the Sigma-Delta modulator. We make the oversampling ratio  $M$  to be 2000, i.e. the loop bandwidth of the PLL  $f_b = 20$  kHz. Then by (3.5) we can get resolutions for  $n = 1, 2, 3$  respectively. See Table 3.1 for results. From the table, we see the order of the Sigma-Delta modulator has at least to

Table 3.1 Sigma-Delta modulator resolutions for different orders

Order of Sigma-Delta modulator: $n$	Resolution: $R$	$ENOB$
1	0.02518	5
2	0.00001259	16
3	0.000000006295	27

be 3 to satisfy the resolution requirement.

A good way to implement a 3<sup>rd</sup>-order Sigma-Delta modulator is to cascading 3 1<sup>st</sup>-order Sigma-Delta modulators, which is called the multi-stage noise-shaping 1-1-1 (MASH 1-1-1) structure, which is shown in Figure 3.2. The most important quality of a MASH modulator is its unconditional stability for any modulator order, because of its first-order nature. The input to the modulator is a 20-bit fractional signal. The function of the quantizer is to keep the integer part and throw the fractional part off. Another advantage of MASH modulator is simple integration in plain CMOS technology, since only adders and registers are needed to implement the noise shaping function. In each stage of the MASH 1-1-1 structure, the quantization error  $E_i$  is the input to the next modulator. By summing the filtered versions of the first-order outputs, the quantization errors of the first and the second modulator is cancelled. Since the Sigma-Delta modulator in fractional-N PLLs is an all digital implementation, the cancellation is perfect. Let us look at the equations for the MASH modulator:

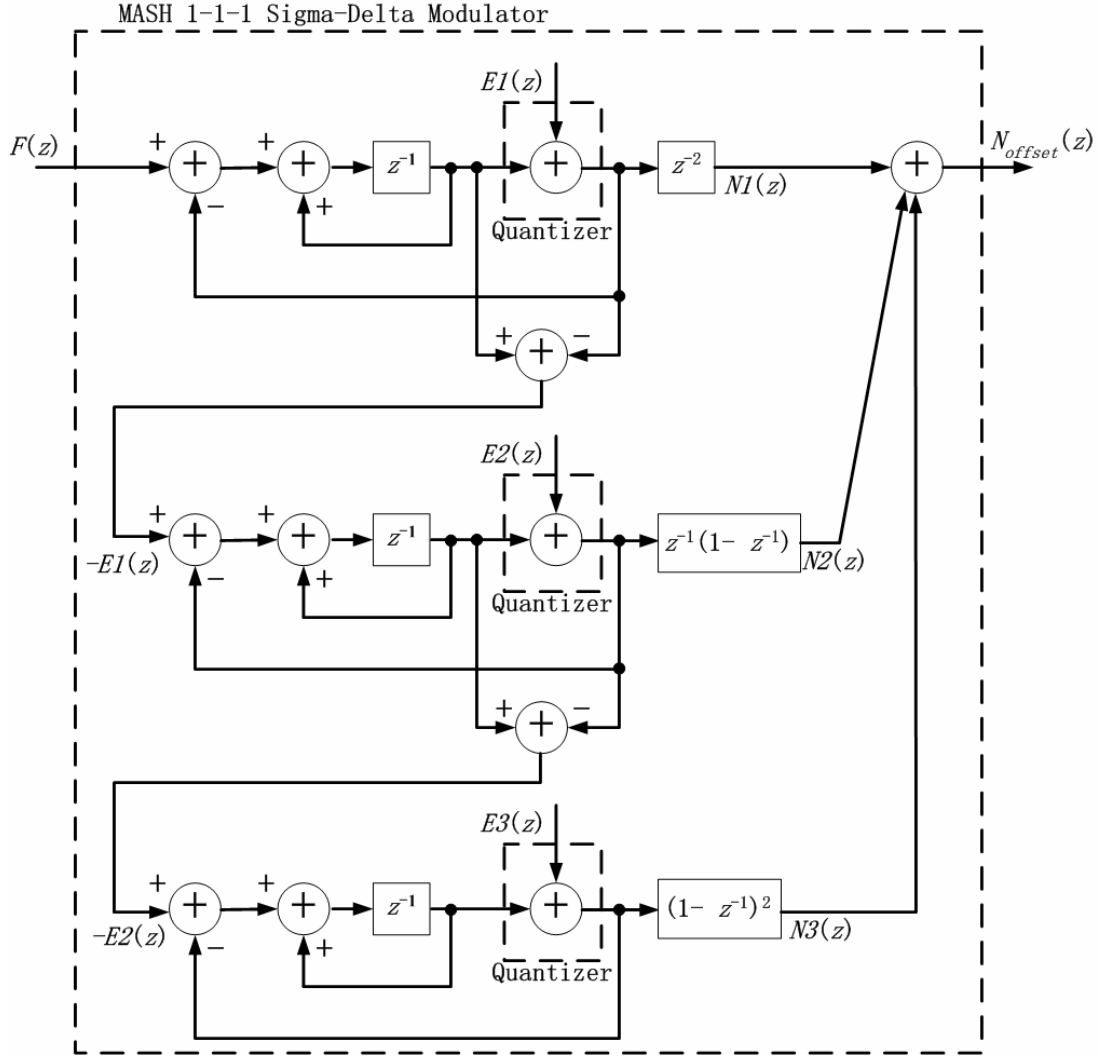


Figure 3.2 The MASH 1-1-1 Sigma-Delta modulator

$$\begin{aligned}
 N_{offset} &= N1(z) + N2(z) + N3(z) \\
 &= z^{-2} [z^{-1} F(z) + (1 - z^{-1}) E1(z)] + z^{-1} (1 - z^{-1}) [-z^{-1} E1(z) + (1 - z^{-1}) E2(z)] \\
 &\quad + (1 - z^{-1})^2 [-z^{-1} E2(z) + (1 - z^{-1}) E3(z)] \\
 &= z^{-3} F(z) + (1 - z^{-1})^3 E3(z)
 \end{aligned} \tag{3.11}$$

Based on the MASH 1-1-1 modulator, the complete fractional-N divider with both the Sigma-Delta modulator and the programmable divider can be built. It is shown in Figure 3.3. The types and sources of the denoted data in Figure 3.3 are listed and explained in Table 3.2.

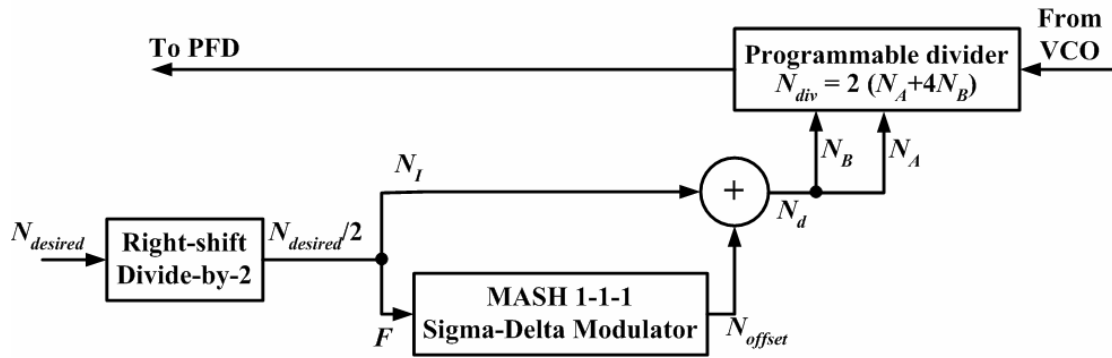


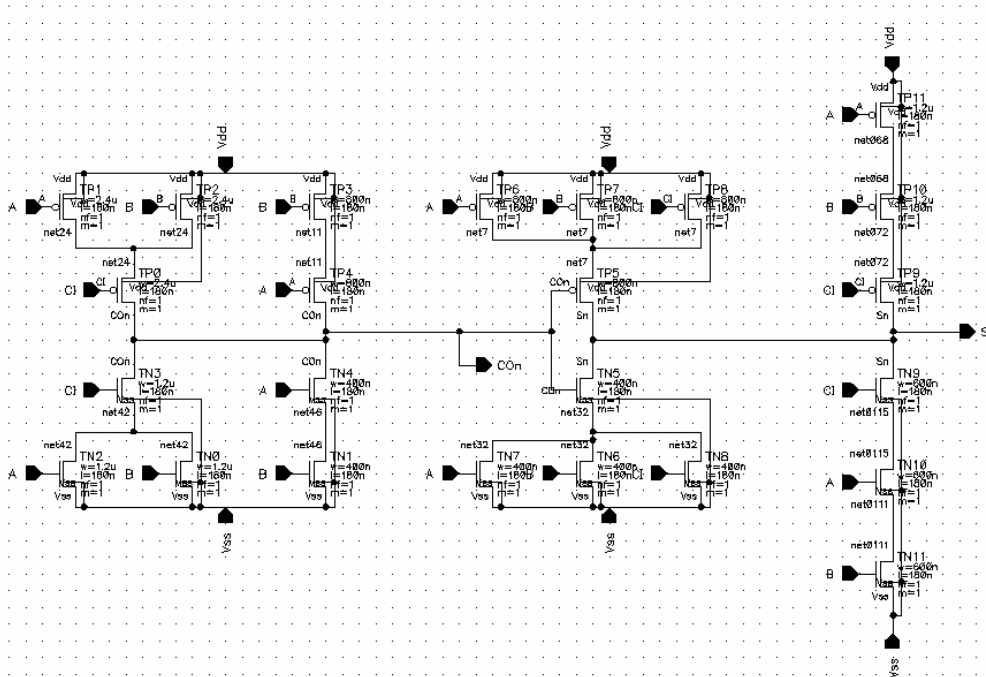
Figure 3.3 The fractional-N divider with MASH 1-1-1 Sigma-Delta modulator

Table 3.2 Data type and operation source of fractional-N divider

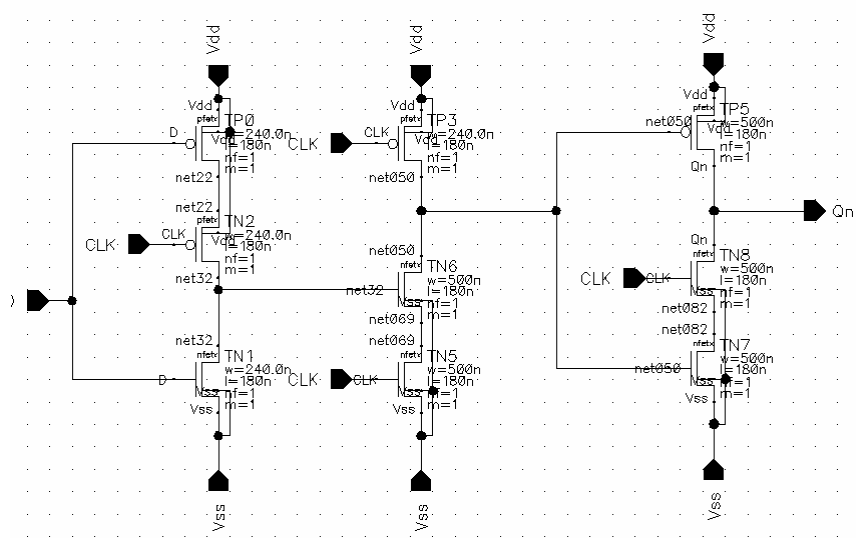
Data name	Data type	Source
$N_{desired}$	Unsigned 7-bit integer + 19-bit fractional	System input
$N_{desired}/2$	Unsigned 6-bit integer + 20-bit fractional	Shift $N_{desired}$ right by 1 bit
$N_I$	Unsigned 6-bit integer	Integer part of $N_{desired}/2$
$F$	Unsigned 20-bit fractional	Fractional part of $N_{desired}/2$
$N_{offset}$	Signed 3-bit integer	Output of S-D modulator
$N_d$	Unsigned 6-bit integer	Sum of $N_I$ and $N_{offset}$
$N_A$	Unsigned 2-bit integer	2 LSB of $N_d$
$N_B$	Unsigned 4-bit integer	4 MSB of $N_d$

Plain CMOS technology has been used to implement the Sigma-Delta modulator.

Figure 3.4 shows the schematics of the two basic circuits of the modulator: mirror full adder and true single-phase clock (TSPC) DFF.



(a)



(b)

Figure 3.4 Schematic of basic building blocks of Sigma-Delta modulator: (a) Mirror

full adder; (b) TSPC D Flip-Flop

# Chapter 4

## Band-Searching Scheme

The band-searching scheme used in the Sigma-Delta fractional-N PLL is to help coarse locking of the VCO to the correct band before the beginning of the fine tuning. The VCO has both coarse digital tuning and fine analog tuning. The traditional PLL locking scheme can only change the voltage of the analog tuning node, and do the analog tuning. However, if the digital control bits are not staying at the correct code, no matter how the analog tuning voltage is changed the VCO cannot be locked to the right frequency. So an additional digital control code locking scheme is required before the analog locking.

The band-searching scheme is indeed another feedback control loop. Since only coarse tuning is required, it provides all digital implementation. That means band-searching scheme is not able to find the exactly correct frequency for the VCO but will help it find the correct frequency range after the coarse tuning step is finished, the band-searching scheme locking loop will be broken, and the traditional locking lock begins to work.

This chapter will show the flow of the band-searching scheme and give simulation results.

## 4.1 Principle of band-searching scheme

A PLL with band-searching scheme is a double loop system. Its locking process has two steps. The first step is coarse tuning, i.e. band searching to find out the correct VCO band, and the second step is traditional analog fine tuning. Figure 4.1 shows a diagram of a double loop PLL. The components outside the dotted line region are the same with those of a normal PLL. Of course, the VCO and the fixed divider are also in the normal loop. Those inside the dotted line region form the band-searching loop. VCO and fixed divider are reused. The D latch and the finite state machine (FSM) are both sequential logics with the clock from the reference clock. The counter counts the output of the latch and acts as a phase accumulator. The counter's reset signal is also the reference clock. The comparator is a combinational logic. It simply compares the result of the counter with desired modulus from system input, and the result of the comparison will be sent to the FSM.

The locking procedure starts from a synchronous reset signal. If the system detects a new modulus input, the “*Hold*” signal becomes low and the system state is reset to the band-searching start state. The switches are controlled to break the normal loop and close the band-searching loop. Figure 4.2 shows the connection state for the band-searching procedure. During this procedure, the analog tuning node of the VCO is disconnected from the output of the LPF. Instead, it is connected to some reference voltage  $V_{min}$ .  $V_{min}$  is set to the minimum tuning voltage available from the output of the LPF. And the digital control bits are set to all zero which means the lowest frequency band,  $Band_0$ , is chosen. See Figure 4.3 for the details of VCO tuning

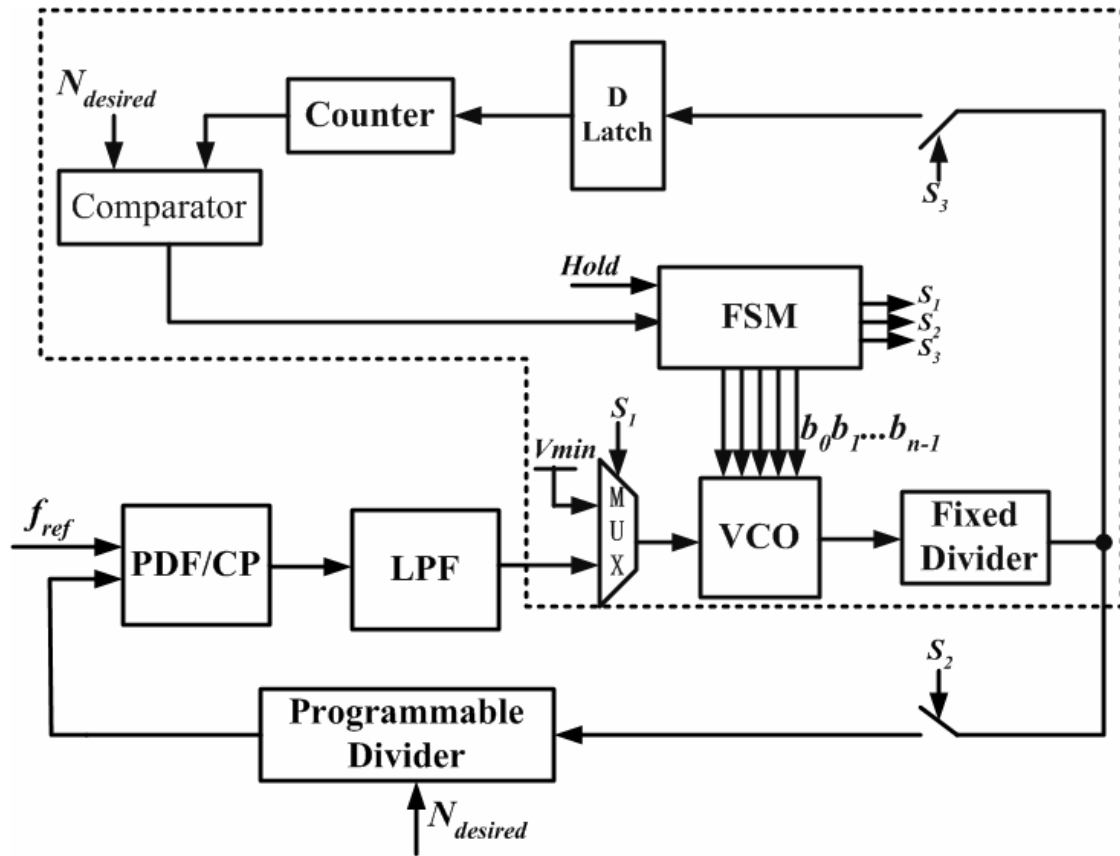


Figure 4.1 Diagram of a double loop PLL with band-searching scheme

characteristic curve. So output frequency of the VCO is now locked at  $f_{m0}$ . Then the counter will count the number of pulse from the output of the VCO within half cycles of the reference clock and compare the counter result with the system input,  $N_{desired}$ . If the result is smaller than  $N_{desired}$ , we can say the desired frequency is out of the locking range. Then the band-searching procedure ends and the system goes back to the normal locking status. Otherwise, the digital control bits plus one and the next band is selected. After another comparison, if the frequency is still too small, go on to the next band, otherwise go back to the previous band and quit the band-searching procedure. If we have reached the maximum available band and the comparison result

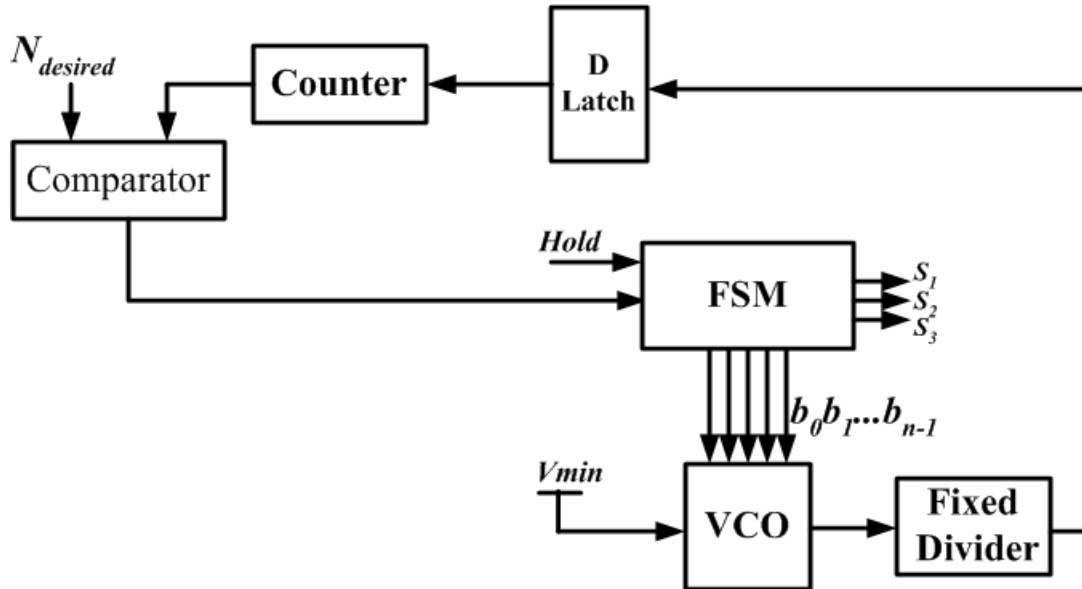


Figure 4.2 Diagram of operation status of band-searching procedure

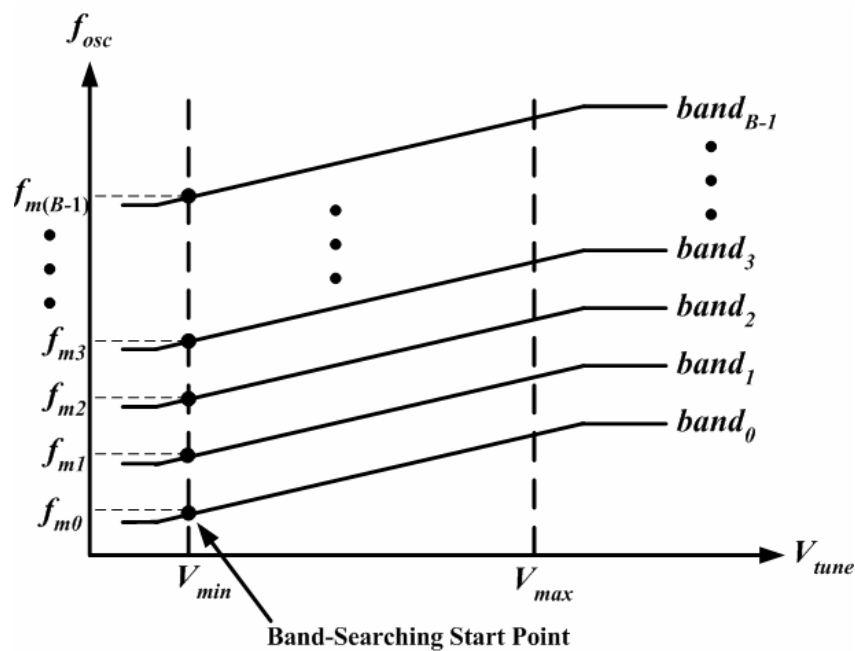


Figure 4.3 Multi-band VCO frequency tuning characteristic curve

is still higher than the reference, the band-searching procedure also ends and digital control bits remains at the maximum value.

After the band-searching procedure, the control logic freezes the digital control

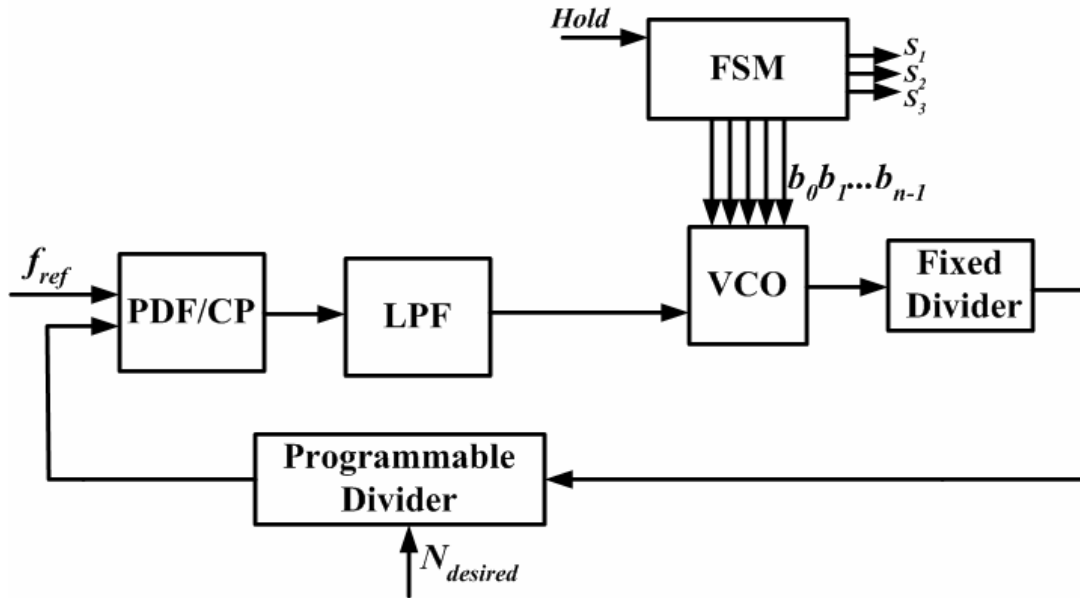
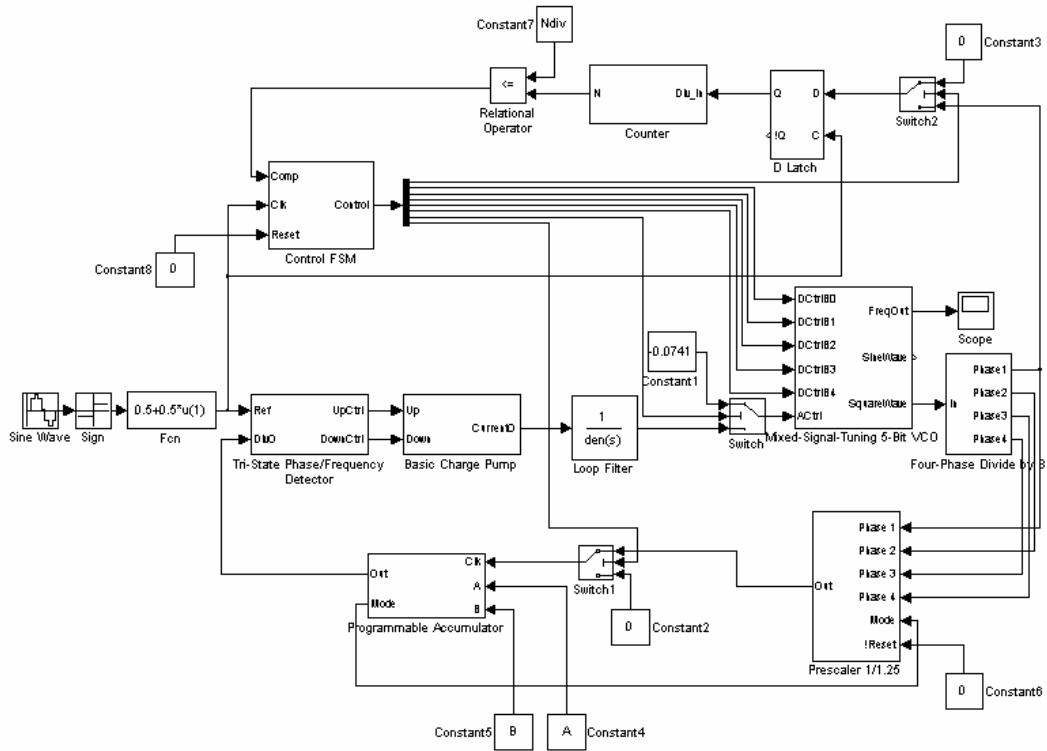


Figure 4.4 Diagram of operation status of normal analog locking procedure

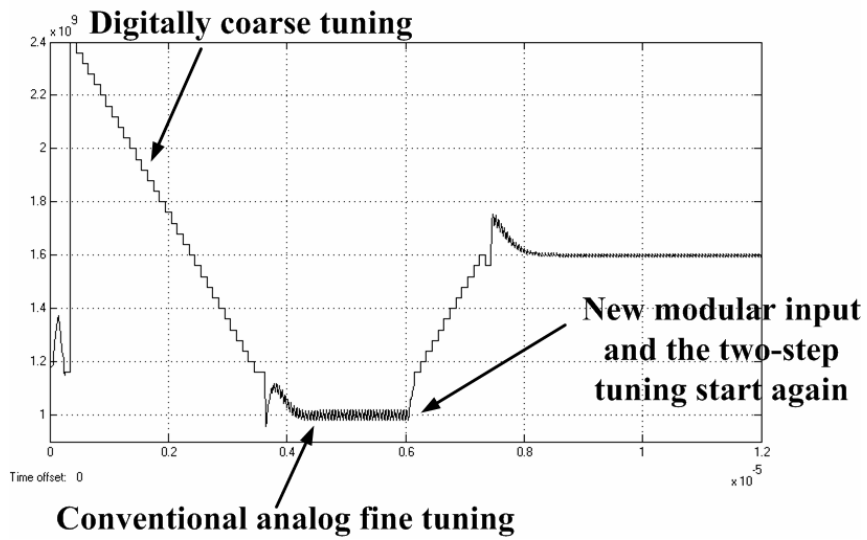
bits, breaks the band-searching loop and turn on the normal locking loop again. Then the normal analog locking procedure begins. Figure 4.4 shows the connection status of the normal locking procedure.

## 4.2 Simulation of PLL with band-searching scheme

Matlab® Simulink model for the PLL with band-searching has been built to verify the idea of the double loop PLL. The VCO used in the model has 5 digital control bits, which means there are total of 32 frequency sub-bands. So the maximum number of band-searching comparison steps is 32. Figure 4.5 gives the model blocks and the result of the locking procedure for two modulus input cases. The output is the oscillation frequency of the VCO changing with time.



(a)



(b)

Figure 4.5 Simulink (a) Model; (b) simulation result of a dual-loop PLL locking procedure

# Chapter 5

## Conclusion

In this project, efforts have been made in the design of a wide-band phase-locked loop. Including only one VCO with very wide frequency tuning range, a wide-band phase-locked loop has advantages of low power and small chip area. It is a key building block for the universal radio, which is the trend of the development of the telecommunication. Multiple communication standards can be satisfied by only one frequency synthesizer. This also makes cognitive communication standard control and switching possible. Users can use the same cell phone all around the world for voice, data, and personal communication needs.

The wide-band implementation of the phase-locked loop puts special requirements to its building blocks. In previous chapters, two primary building blocks for a wide-band Sigma-Delta fractional- $N$  have been discussed. They are feedback frequency divider and the Sigma-Delta modulator. Also, a double-loop solution for combining the coarse digital tuning and fine analog tuning of the VCO is proposed.

The feedback frequency divider consists of two important components: high frequency divider and programmable divider. The high frequency pre-scalar divides the output of the VCO signal to lower frequency while it can also generate

four-phase output signal, which is important for the programmable divider based on phase-selection. Source coupled logic is chosen as the logic style of the high frequency divider to achieve wide input frequency range and low transient current fluctuation. It can operate for the VCO frequency from 1.3 GHz up to 3 GHz. At the same time, a very small bias current fluctuation has been observed through transient simulation, thus minimal transient switching noise will propagate to the power supply line. However, the source coupled logic consumes more static power than other high frequency dividers. Therefore, a lot of effort has been put into careful sizing of the devices to use the minimized power while satisfying the frequency requirement. The most important part for the programmable divider is the phase-select 1/1.25 pre-scaler. It has a four-phase signal input but only one phase can be selected during one time. The selection process has very low propagation delay, thus it is very suitable for high frequency operation. In order to minimize of the risk of spike, a parallel phase selection solution has been proposed in this project.

The Sigma-Delta modulator is built as a pseudo random number generator. MASH 1-1-1 structure is chosen for its high resolution and unconditional stability. A careful detail calculation has been done to find the required number of binary bits.

A coarse tuning is necessary for a VCO with multiple digitally controlled bands. Using a double loop, at the beginning of the phase-locking procedure, only the coarse locking loop is working with the fine locking loop broken. After several comparison cycles the digital control bits are set and the fine locking procedure begins. The fine locking procedure is exactly the same as the normal phase locking. A Simulink model

has been built and the behavior simulation have been done to verify the approach.

Future work of this project includes implementing other blocks of the wide-band phase-locked loop, such as the phase frequency detector, charge pump, low-pass loop filter and the most importantly, the VCO.



# REFERENCE

[Berny04] A.D. Berny, A.M. Niknejad and R.G. Meyer, "A 1.8 GHz LC VCO with 1.3 GHz Tuning Range and Mixed-Signal Amplitude Calibration," *IEEE Symposium on VLSI Circuits, Digest of Technical Papers*, pp. 54-57, 17-19 June 2004.

[Craninckx96] J. Craninckx and M. S. J. Steyaert, "A 1.75-GHz / 3-V Dual Modulus Divide-by-128/129 Prescaler in 0.7- $\mu$ m CMOS," *IEEE Journal of Solid-State Circuits*, Volume: 31, Issue: 7, July 1996.

[Rategh99] H.R. Rategh, H. Samavati and T.H. Lee, "A 5 GHz, 32 mW CMOS frequency synthesizer with an injection locked frequency divider," *IEEE Symposium on VLSI Circuits, Digest of Technical Papers*, pp. 12.1.1-12.1.4, 1999.

[Vaucher02] C.S. Vaucher, *Architectures for RF Frequency Synthesizers*. Boston: Kluwer Academic Publishers, 2002.

[Wohlmuth02] H.-D. Wohlmuth and D. Kehrer, "A High Sensitivity Static 2:1 Frequency Divider up to 27 GHz in 120 nm CMOS," *IEEE European Solid-State Circuit Conference*, pp. 823-826, Firenze, Italy, September 2002.