INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

ProQuest Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

Quantization Noise Reduction in PLLs Using Multiphase VCOs

by

Igor Miletic

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the Degree of Master of Applied Science.

Ottawa-Carleton Institute for Electrical and Computer Engineering

Department of Electronics

Faculty of Engineering

Carleton University

Ottawa, Canada

Abstract

This thesis presents the design and implementation of Sigma-Delta PLL with multiphase architecture. All the components of the system are designed in $0.18\mu m$ TCMS digital CMOS process, excluding the loop-filter which is an external component. The frequency range of the PLL ranges from 10 MHz to 120MHz and an intended application is the subharmonic injection signal in the injection locked VCOs and as a clock generator in the subharmonic ADCs. A multiphase architecture,which uses inherent multiphase outputs of the ring oscillator, is presented. The purpose of this architecture is to decrease a quantization noise from the $\Sigma\Delta$ modulator. Both single-ended VCO and differential VCO are presented and compared in terms of the phase-noise performance, power consumption, and supply sensitivity. It has been found that single-ended design gives superior performance when the power supply is isolated.

Acknowledgments

The completion of this thesis would not have been possible without the input, guidance, and support given to me by faculty members, family, and friends. First, I would like to express my gratitude to my thesis adviser, Dr. Ralph Mason, for his support and unfailing patience during the various stages of this work. Second, I would like to thank my colleagues and friends, John Danson, Victor Karam, and Tony Forzley for taking time out of their busy schedules to review my work and provide me with their insightful suggestions. Third, I would like to extend my gratitude to all faculty members in the Department of Electronics for creating a supportive environment and making my work possible.

The financial support of Carleton University, Micronet, the Ontario Graduate Scholarship Program, and Canadian Microelectronic Corporation fabrication support are also gratefully acknowledged.

On a personal note, I would also like to extend special thanks to my fiance and my family who have given me their continued love, support, and encouragement throughout my years in the Master's program and, especially during the final and, at times, hectic stages of this thesis.

Contents

List of Figures

List of Tables

Abbreviations

AC	Alternative Current
ADC	Analog to Digital Converter
BW	Bandwidth
CMOS	complementary-metal-oxide-semiconductor
DAC	Digital to Analog Converter
DC	Direct Current
DDS	Direct Digital Synthesizer
DSM	Delta-Sigma Modulator
EVM	error vector magnitude
GaAs	Gallium Arsenide
IF	Intermediate Frequency
ILO	Injection Locked Oscillator
ISF	Impulse Sensitivity Function
LC	Inductor-Capacitor
LFSR	linear feedback shift register
LNA	Low Noise Amplifier
LO	Local Oscillator
LSB	Least Significant Bit
NTF	Noise Transfer Function
PFD	Phase Frequency Detector
PLL	Phase Locked Loop
PN	Phase Noise
PSD	power spectral density
PSRR	power supply rejection ratio
RF	Radio Frequency
SiGe	Silicon Germanium
$\Sigma\Delta$	Sigma-Delta Modulator
SNR	Signal to Noise Ratio
STF	Signal Transfer Function
VCO	Voltage Controlled Oscillator

Chapter 1

Introduction

1.1 Chapter Overview

The introduction highlights the scope, motivation and technical contribution of the research conducted. It also defines specific applications of the techniques and technology and outlines the thesis organization.

1.2 Motivation

High demand of cell phones, personal digital assistants (PDA) and other mobile devices has been a main force in increased wireless system complexity. With the rapid development of wireless mobile gadgets that can send and receive great amounts of data, power consumption is a vital issue. Companies with low power devices fared better on the market than their competitors. This created both the motivation and opportunity for the mobile market to research new ideas of data transmission. These new technologies add features to their devices while keeping power consumption low and while meeting or exceeding the required specifications.

In Radio Frequency (RF) and analog design, one of the most desired features is high bandwidth. The signals have to be transfered quickly and efficiently from one point to another. In traditional designs, this usually meant increasing power in the most critical circuits, such as phase locked loops, oscillators and low noise amplifiers, to allow for enhanced performance. This approach is low risk and has been proven to work, however it decreases the mobility of the device due to a shorter battery life. Designers are currently trying a multitude of different approaches to decrease power. The subharmonic signal recovery approach has recently become of increased interest. Contrary to the standard heterodyne approach where signals are down-converted from the RF to the Intermediate Frequency (IF) and then to baseband, the subharmonic approach uses a subharmonic frequency to sample the signal into the digital domain.

In the subharmonic approach, an Analog to Digital Converter (ADC) is used to sub-sample and convert the signal into the digital domain. Ideally, this should be done at the RF frequency, however, ADCs are usually limited by the speed of switches and parasitic charge on them, which is signal dependent. Decreasing the frequency or increasing power usually diminishes the effects of limited switching speed. Therefore, the trade-off in architecture is usually met by having the ADC sub-sampling at an IF frequency. The ADC also depends on the accuracy and phase noise of the clock that is often generated by an on-chip Phase Locked Loop (PLL). Another issue with the subharmonic approach is PLL resolution. Since the frequency is divided by the subharmonic factor used, resolution of the PLL has to be increased by the same factor to insure the same accuracy of the Local Oscillator (LO) signal. Various methods of decreasing the frequency step of PLL will be discussed in following chapters.

PLLs represent an integral and considerable portion of radio transceivers. The most recent RF chips utilize an on-chip PLL with an external loop filter, which functions as a LO generator in the circuit. Reducing power consumption as well as phase noise of the PLL is an important and challenging goal toward increasing the data rate of the transceiver. A fractional-N PLL is particularly attractive in high speed data sampling and clock recovery systems such as subharmonic architectures. Due to the low operating frequency of the PLL, ring oscillators are a viable alternative to Inductor-Capacitor (LC) tank oscillators as long as the loop bandwidth and loop order can be kept high enough. The reason for this is the fact that the Voltage Controlled Oscillator (VCO) noise becomes less significant relative to the effect of increased bandwidth. A high loop bandwidth allows for direct modulation, thereby decreasing power consumption by eliminating the Digital to Analog Converter (DAC) in the transmit chain. The inherent multi-phase capability of the ring oscillator allows the quantization noise to be decreased, which is another step in widening the bandwidth of the PLL. In addition, the lower frequency saves power because the oscillation rate is lower and the number of stages in the divider is reduced.

1.3 Thesis Focus

This thesis focus is to create a low noise, wide bandwidth PLL that can be used in RF circuits. The target applications for this synthesizer are a clock generator for a subharmonic ADC and a frequency synthesizer for an injection locked VCO. This PLL will be fully integrated with the exception of the loop filter which will be external. Furthermore, it will use the multiphase capability of the ring oscillator to decrease the

quantization noise from the Sigma-Delta Modulator ($\Sigma\Delta$), while keeping the power consumption low. In addition, a comparison of the performance and trade-offs for both differential and single-ended ring oscillators will be discussed.

1.4 Contributions

The major research contributions of this thesis are as follows:

(1) Low phase noise single-ended and differential oscillators without a voltage-to-current converter are proposed. These topologies avoid current sources in order to ensure minimal flicker noise. They use a loop filter voltage in order to directly control the frequency of oscillation.

(2) A novel PLL architecture that allows the use of a higher order $\Sigma\Delta$ modulation, without decreasing the loop bandwidth, by using a multiphase ring oscillator.

1.5 Thesis Organization

This thesis is organized into six chapters.

The current chapter gives a short introduction to PLLs and their applications. Chapter 2 gives background on PLLs, including types of PLLs and their components. It introduces both the LC and ring oscillators. It also provides linear models for each component. Theory from Chapter 2 is used later in Chapter 3 to calculate the influence of the phase noise from each component on the output of a PLL.

Chapter 3 gives detailed analysis of a PLL, concentrating mostly on the phase noise. It also shows reasons for reference spurs and for noise folding. At the end, it introduces the topic of the quantization noise reduction.

Chapter 4 presents circuit design of the PLL with phase noise and power results.

Chapter 5 presents measurement results, and discuss as reasons for mismatch in simulated and measured results.

Chapter 6 presents the conclusion drawn from this work. It also presents possible future work.

Chapter 2

Phase Locked Loops

2.1 Introduction

The purpose of this section is to give an overview of PLLs and their components. Among many other things, PLLs are either used as clock generators or LO generators. When used as an LO generator in conjunction with a mixer, its purpose is to convert a signal from RF to baseband, in a receiver, and from baseband to RF, in a transmitter. Usually, the PLL consists of the following fundamental components: VCO, Phase Frequency Detector (PFD), charge pump, loop filter, divider and a control circuit for setting the divider ratio. Each of these components influence the performance of the PLL in different ways, hence the impact of each component on the overall system performance has to be considered before implementing the circuits in a PLL. A good understanding of the performance and behavior of each component is essential to building a functional PLL.

Section 2.2 describes the most common types of PLLs. Section 2.3 describes common VCO architectures, while sections 2.4 and 2.5 discuss PFDs and dividers

respectively. Section 2.7 describes fractional PLLs and section 2.6 gives an overview of basic lead-lag passive loop filters. The last two sections, 2.8 and 2.9, provide introductory information with respect to the major topic of this thesis, $\Sigma\Delta$ modulation and the minimization of quantization noise.

2.2 PLL Types

A PLL, among many other applications, can generate the LO signal for the transceiver or the clock signal for the baseband or the ADC. It is a closed loop feedback system that uses a reference signal to generate higher output frequencies, f_{out}. There are different architectures to generate the LO but the most common are integer-N and fractional-N PLLs [1], [2], [3].

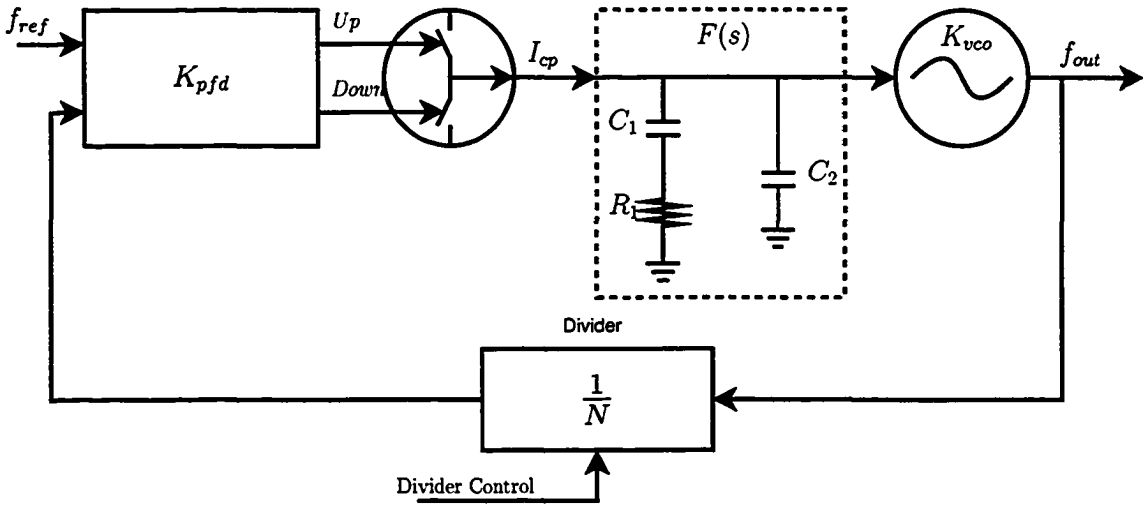


Figure 2.1: PLL block diagram

Figure 2.1 shows an integer-N PLL. It consists of a PFD, charge pump, VCO and divider. The output created by this PLL is forced to be an integer multiple of the reference frequency, f_{ref} due to the divider in the feedback loop. A PFD with a charge pump delivers a train of current pulses into the filter via the charge pump. The widths of pulses are proportional to the delay between the reference signal, f_{ref} , and the divided signal from the VCO. The loop filter integrates these pulses to create a bias voltage for the VCO. Once this bias is set correctly, the phases of the reference signal and the divided VCO signal become synchronized. At this point, the PLL is phase locked and the bias voltage is ideally fixed. However, since there is always some difference in the positive and negative charge pump pulses, small current pulses will be emitted by the charge pump and integrated by the loop filter, thereby modulating the VCO bias voltage. These pulses lead to undesirable reference spurs on the output of the VCO [4], [5].

A more advanced architecture is the $\Sigma\Delta$ fractional-N PLL. It is based on the integer-N PLL, but as opposed to the integer-N PLL, its divisor is dynamically varied among multiple values to create a frequency that can be a fractional multiple of the reference. Figure 2.2 shows a simple diagram for the fractional-N PLL. In situations where resolution of the signal must be smaller than the reference signal frequency, the fractional-N PLL is a viable option. Unlike an integer-N PLL, the fractional-N PLL never fully locks. Since the $\Sigma\Delta$ modulation is changing the division ratio from one reference period to another, there will always be charge pump current pulses dumped into the loop-filter [6].

To find the output frequency, average division ratio has to be found first. Let define integer dividing ratios as N, $N+1$, $N+2$, ..., N_{max} where N is minimum

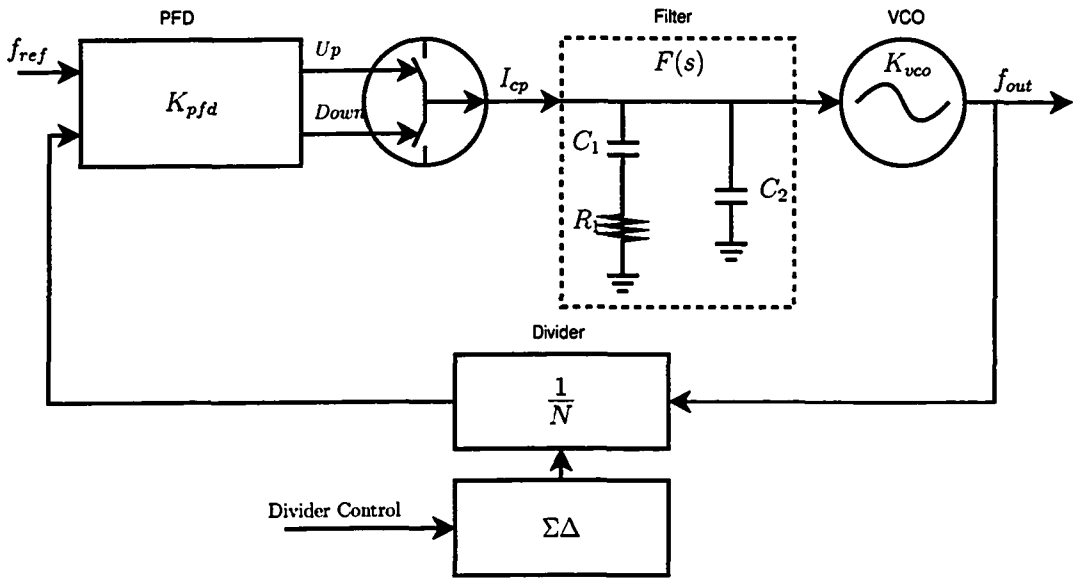


Figure 2.2: PLL block diagram - fractional-N

division ratio and N_{max} is maximum. Then, if divider during time T divided X_N times by N and X_{N+1} times by $N+1$, average divider ration can be found as:

$$N_{avg} = \frac{N \cdot X_N + (N+1) \cdot X_{N+1}}{X_N + X_{N+1}} \tag{2.1}$$

For example, if $X_N = X_{N+1}$ then average divisor will be $N + 0.5$. Similarly, this equation can be expanded to cover all the divisor ratios:

$$N_{avg} = \frac{N \cdot X_N + (N+1) \cdot X_{N+1} + (N+2) \cdot X_{N+2} + ... + N_{max} \cdot X_{max}}{X_N + X_{N+1} + X_{N+2} + ... + X_{max}} \tag{2.2}$$

And output frequency can be written as:

$$F_{out} = \frac{F_{ref} \cdot [N \cdot X_N + (N+1) \cdot X_{N+1} + (N+2) \cdot X_{N+2} + ... + N_{max} \cdot X_{max}]}{X_N + X_{N+1} + X_{N+2} + ... + X_{max}} \tag{2.3}$$

As Equation (2.3) shows, the VCO output frequency will be defined by the average divisor ratio.

2.3 VCO

An essential block in all PLL designs is the VCO. The VCO is a signal generator that can be controlled by a loop filter voltage, $V_{tune}(t)$. The frequency range of the VCO is defined by the center or free running frequency (f_c), and K_{vco}. K_{vco} is a constant that defines sensitivity of the VCO frequency to the input bias. To control the VCO, it is necessary to change the bias voltage at the input of the VCO. In general, the higher the DC level of the input signal, the higher the output frequency of the VCO (see Equation (2.4) and Figure 2.3).

$$f_{vco}(t) = \left(f_c - \frac{K_{vco}}{2}V_{range}\right) + V_{tune}(t) \cdot K_{vco} \tag{2.4}$$

Equation (2.4) shows relation between tuning voltage and output frequency. To find relation between tuning voltage and phase, it is possible to integrate frequency, since integral of frequency is phase. The relationship between the time-domain ($V_{tune}(t)$) and the phase of the VCO output (Φ_{out}) then can be written as:

$$\Phi_{out} = \int_{-\infty}^{\tau} 2\pi f_{out}(t)dt = \int_{-\infty}^{\tau} 2\pi K_{vco}V_{tune}(t)dt \tag{2.5}$$

Equation (2.5) shows that the VCO output phase is a constant, $2\pi K_{vco}$, multiplied by the integral of the tuning voltage over time. In Laplace domain, integrating is equivalent to $\frac{1}{s}$, thus, the VCO transfer function can be converted to the Laplace

domain as:

$$H_{VCO} = \frac{K_{vco}}{s}\left[\frac{V}{rad}\right] \tag{2.6}$$

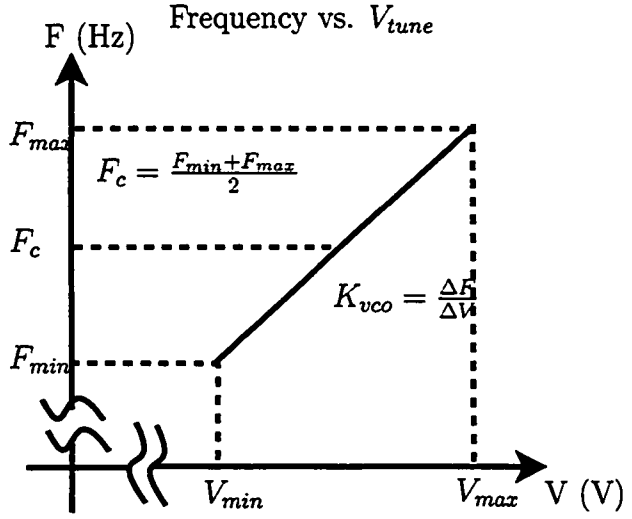


Figure 2.3: Frequency vs. tuning voltage of ideal VCO.

There are different types of oscillators used for the PLL VCO,the most common being ring oscillators, LC tank oscillators and associated variants.

The LC VCO, which is also known as a negative-g_m cell, is popular in RF design because of its good phase noise performance. However, it usually requires external pins to implement the LC tank due to the size and quality factor of the required inductors and/or capacitors. To control the oscillation frequency, the LC VCO uses tunable capacitors called varactors (see Figure 2.4). Its oscillating frequency can be found as:

$$\omega = \frac{1}{\sqrt{L \cdot (C_{constant} + C_{varactor})}} \tag{2.7}$$

where:

- $C_{constant}$ is capacitance added to move the tuning range to the proper frequency and parasitic constant capacitance together,

- $C_{varactor}$ is capacitance that changes with the applied tuning voltage, and

- L is inductance in the LC tank.

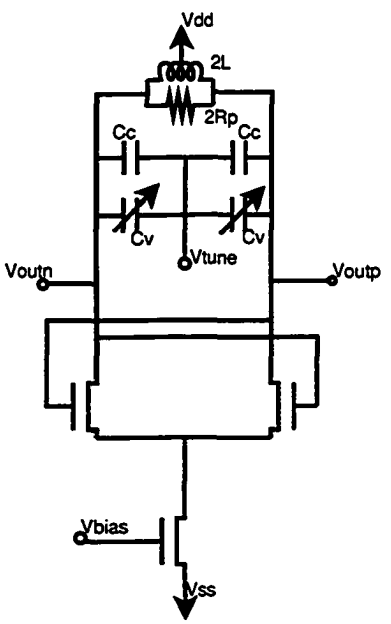


Figure 2.4: Cross-coupled LC oscillator

Given that neither inductors nor capacitors are ideal components, oscillations would dissipate over time. Therefore, extra gain is obtained from a negative g_m cell to insure oscillations. This gain is often set between 2dB and 4dB to compensate for losses from the inductor's resistance and to ensure proper start-up of the oscillations. Hence, a negative-g_m cell is used to introduce extra current into the tank to compensate for losses in the inductor and other components [7], [8].

Another popular way of realizing a VCO is to use a ring oscillator with frequency control added to it. Ring oscillators are usually designed as an odd number of

complementary-metal-oxide-semiconductor (CMOS) inverters, where minimum number of stages is 3 [9], (see Figure 2.5) placed in a feedback loop. Since each inverter has 180° phase shift, an odd number insures that the circuit will be unstable and oscillations will occur. Ring oscillators are very compact, but inherently noisy compared to the LC tank oscillators. Because of the aforementioned characteristics, they are usually used in wide-band PLLs, where in-band noise is filtered by the closed loop.

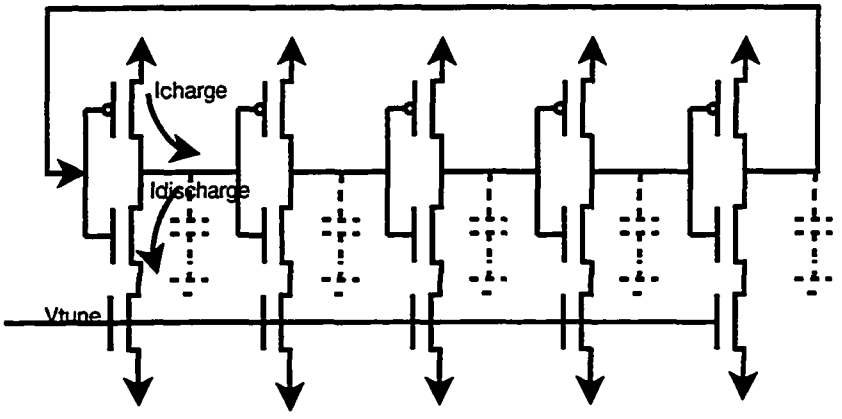


Figure 2.5: Ring oscillator

The frequency of oscillation in a ring oscillator can be found as the inversion of the total delay through all the cells. Each half period, the signal propagates through every inverter, therefore, the full period is equal to the delay of each cell multiplied by two. For example, if at the beginning of the chain the input is low, after half a period the circuit will force the output to high. This output is also the input to the chain, therefore the half period following the input will be set to low again.

To calculate oscillating frequency, first delay through each stage, t_d, has to be found. Then this delay is multiplied by N, where N defines number of stages. This gives total delay through all stages. Finally, since signal needs to pass twice through

these stages, first for low output then for high, the following equation can be applied to find oscillating frequency:

$$F_o = \frac{1}{2 \cdot N \cdot t_d} \tag{2.8}$$

The simplified delay of the CMOS inverter will be derived in the next section.

2.3.1 Inverter Delay

In case when the output value changes from high to low, we can assume that the input has already changed to high. This assumption is valid because the output of the inverter is expected to stay constant as long as the input does not cross the threshold voltage. After that, it will start changing very slowly since both PMOS and NMOS transistors are on. Hence, by the time the output changes significantly, the input will already be set. Assuming that the output switches from high to low, the PMOS transistor is turned off, while the NMOS transistor has to go through two stages. In the first stage, while $V_{out} > V_{dd} - V_{th}$, the NMOS transistor is in saturation (Figure 2.6). The second stage is when the transistor is in the linear region, when $V_{out} < V_{dd} - V_{th}$. Since the transistor threshold voltage ($\approx 0.6V$) is significantly lower than the inverter threshold voltage $V_{INV_{th}} \approx 0.9V$ (assuming $V_{dd} = 1.8V$), it can further be assumed that the transistor will stay in the saturation during the transition delay.

In the saturation region, up to time $t = t_1$, the following equation can be applied [10]:

$$I_d = K_n \cdot \frac{(V_{gs} - V_{thn})^2}{2} \approx K_n \cdot \frac{(V_{dd} - V_{thn})^2}{2} \tag{2.9}$$

where I_d represents the current through one inverter cell. Since the delay time, t_d, is defined as:

$$t_d = \frac{C \cdot \int_{V_{dd}}^{V_{dd}-V_{thn}} V(t) dV_{out}}{I} \tag{2.10}$$

by integrating voltage over time and substituting I with I_d from Equation (2.9), the fall delay when the NMOS is in the saturation, can be written as:

$$t_{fall} = \frac{2 \cdot C \cdot V_{thp}}{K_p \cdot (V_{dd} - V_{thp})^2} \tag{2.11}$$

where C is capacitance of the cell output (the next cell's input capacitance).

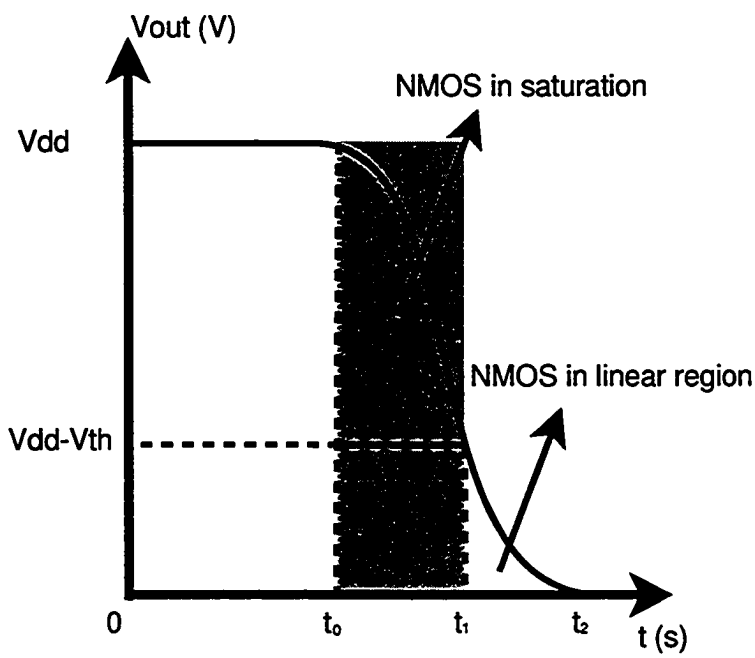


Figure 2.6: V_{out} vs. time for the inverter

Similarly, the rise time can be defined as:

$$t_{rise} = \frac{2 \cdot C \cdot | V_{thn} |}{K_n \cdot (V_{dd} - | V_{thn} |)^2} \tag{2.12}$$

The oscillating frequency can therefore be written as:

$$F = \frac{1}{2 \cdot t_d \cdot N} = \frac{K \cdot (V_{dd} - \mid V_{th} \mid)^2}{4 \cdot N \cdot C \cdot V_{th}} \tag{2.13}$$

assuming that $V_{thn} = V_{thp}$ and that $K_p = K_n$.

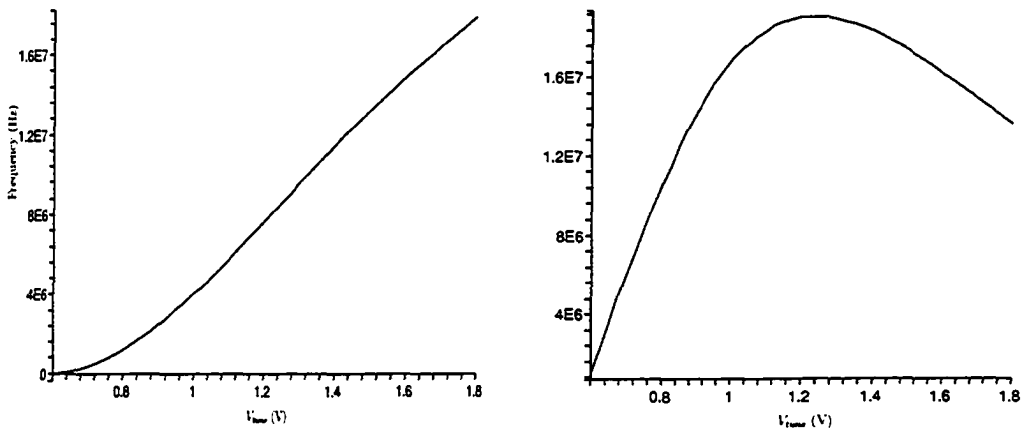
In case of a current starved inverter, the frequency can be controlled by changing V_{tune} (see Figure 2.5). In this particular case, the fall time will change because the turn on voltage for the bottom transistor is not V_{dd} but V_{tune}. Since the bottom transistor limits the current, the delay equation of (2.11) can be reformulated as:

$$t_1 = t_{fall} = \frac{2 \cdot C \cdot V_{thn}}{K_n \cdot (V_{tune} - V_{thn})^2} \tag{2.14}$$

Furthermore, with V_{tune} being lower than V_{dd}, it can be assumed that the transistor will stay in the saturation region most of the time, since $V_{ds} > V_{tune} - V_{th}$ will be true for most of the transition. The delay will be therefore defined by Equations (2.12) and (2.14), and the frequency can be approximated as seen below:

$$F \approx \frac{1}{t_{rise} + t_{fall}} = \frac{1}{\frac{2 \cdot C \cdot |V_{thp}|}{K_p \cdot (V_{dd} - |V_{thp}|)^2} + \frac{2 \cdot C \cdot V_{thn}}{K_n \cdot (V_{tune} - V_{thn})^2}} \tag{2.15}$$

Since all the parameters, except for V_{tune}, are constant, the linearity of the VCO can be checked by plotting the above function. As seen in Figures 2.7(a) and 2.7(b) (derivative of frequency with respect to V_{tune}), this VCO is not linear when being directly controlled by voltage. Usually, when using ring oscillators, the tuning voltage is converted to a current by a voltage-to-current (V-I) converter. This particular method linearizes the VCO behavior due to linearity of the V-I circuit. This current



(a) Frequency vs. V_{tune}

(b) K_{vco} vs. V_{tune}

Figure 2.7: VCO characteristics

is then mirrored inside the inverter, giving a linear dependency of the frequency on the tuning voltage. However, adding additional circuitry increases the complexity of the design, power consumption, and increases noise, as will be shown later.

2.4 PFD and Charge Pump

The purpose of the PFD and charge pump is to compare the divided VCO phase and frequency with the reference signal phase and frequency and to correct the tuning voltage of the loop to decrease the phase/frequency error. Moreover, the PFD is used to detect differences in the phase between two signals and to translate that information to the charge pump, which dumps a proportional amount of charge to or from the loop filter. When placed in a PLL, the PFD and charge pump work to decrease the difference in the phase of the two input signals. The actual amount of correction depends on the charge pump current and loop filter parameters, and

the actual phase/frequency error. There are numerous ways to implement a PFD [11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25], but the most commonly used is a tri-state PFD (Figure 2.8 and 2.9(a)).

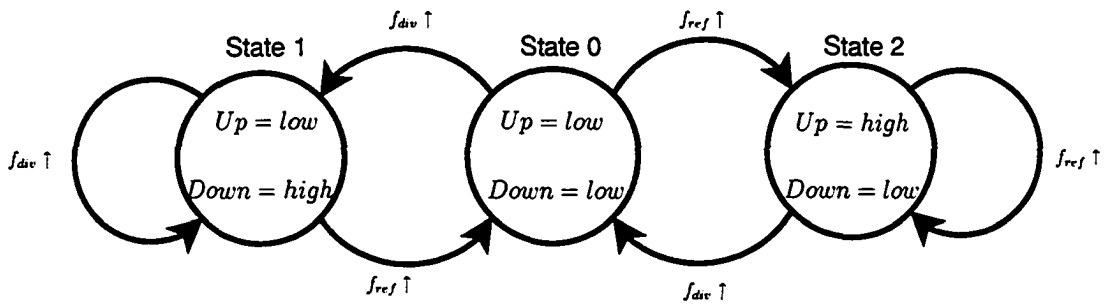
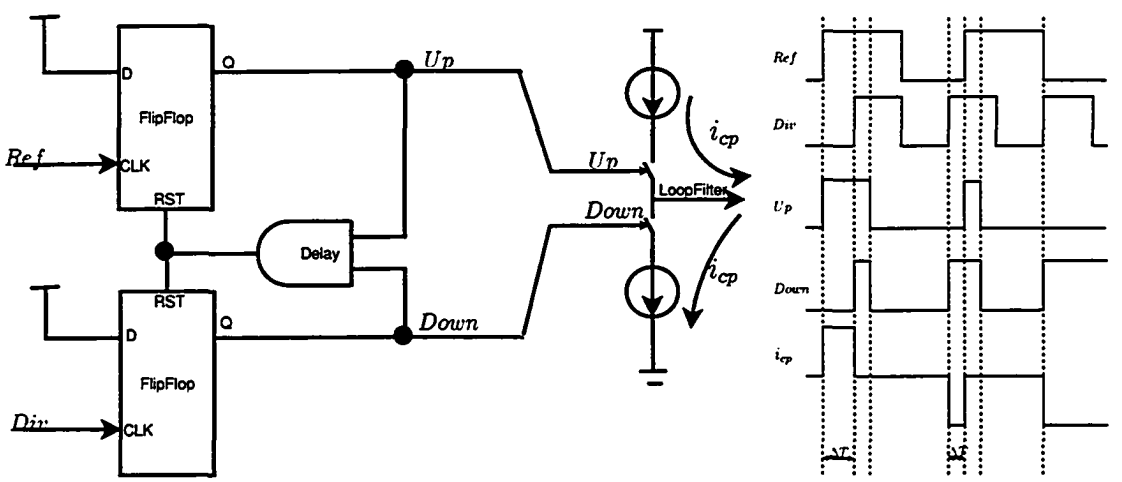


Figure 2.8: State-diagram for the PFD



(a) PFD and Charge Pump Block Diagram

(b) PFD and Charge Pump Waveform

Figure 2.9: PFD and Charge Pump

Figures 2.8 and 2.9(b) show how the tristate PFD works. For example, in a case where $f_{ref} > f_{div}$, which means that the reference signal rising edge, f_{ref}, comes before the VCO divider signal, f_{div}, the Up signal is set high. Consequently this will

force the PFD into **State 2** and turn on the positive current in the charge pump. The PFD will stay in this state until a positive edge from the divider, f_{div} arrives and returns the PFD to **State 0**. At that moment, both the *Up* and *Down* signals will be high for the duration of the delay in the Delay cell (Figure 2.9(a)) before the PFD goes back to **State 0**. This means that both positive and negative currents in the charge pump will be on and the net charge output to the loop filter will be ideally zero. Similarly, for the case when $f_{div} > f_{ref}$, negative current pulses will be pumped onto the loop filter by forcing the PFD into **State 1**. As a consequence, the tuning voltage will be adjusted to decrease frequency/phase error. This procedure will repeat until the phase difference is at a minimum detectable level, called the dead-zone (in Figure 2.10 one can see the exaggerated dead-zone). The dead-zone is caused by a finite turn on time for the charge pump switches and a finite rise and fall time of the *Up* and *Down* signals. To minimize the dead-zone effect, a delay is introduced to allow for the switches to fully turn on before resetting it.

In the instance where the frequency difference between the *Div* and *Ref* signals is large, it can be expected that most of the time the *Up* will be high, with pulses of *Up* getting very narrow during cycle-slipping (for the case that $f_{ref} \gg f_{div}$), or the *Down* will be high (for $f_{ref} \ll f_{div}$). As seen in Figure 2.10, this is caused by a vertical asymmetry in the PFD response. This frequency-steering property of the tri-state PFD is a key attribute for enabling frequency detection, and it is one of the reasons that this type of PFD is so popular. The asymmetric phase error characteristics allow the PFD to differentiate between negative and positive frequency errors (i.e. only one zero crossing) [26], consequently making the average output either positive

or negative. In the ideal case when using this type of the PFD, the PLL should always lock.

The tristate PFD has a response between -2π to 2π radians, and its characteristic is defined by a constant, K_{pfd}. K_{pfd} defines how much charge is put onto the loop filter for a specific phase error during one reference period, T. Hence, the phase error, θ_e, can be defined as:

$$\theta_e = \frac{(\theta_{ref} - \theta_{div})}{2\pi} I_{cp} \tag{2.16}$$

The term $\frac{I_{cp}}{2\pi}$ is actually the gain of the PFD. Hence, we can write:

$$K_{pfd} = \frac{I_{cp}}{2\pi} \tag{2.17}$$

for the PFD constant.

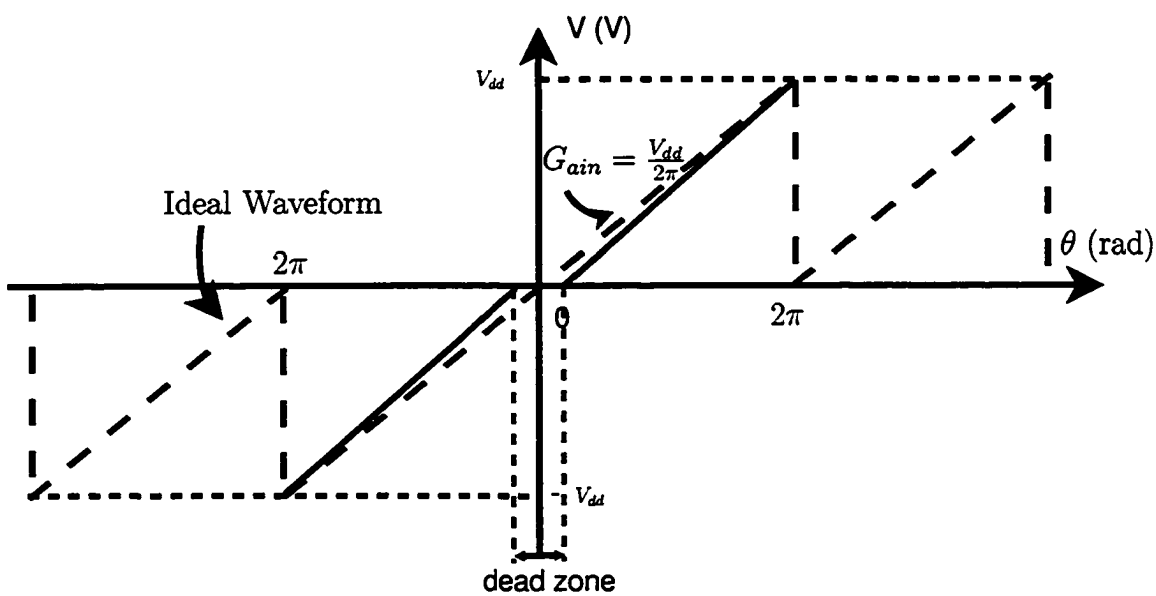


Figure 2.10: PFD response.

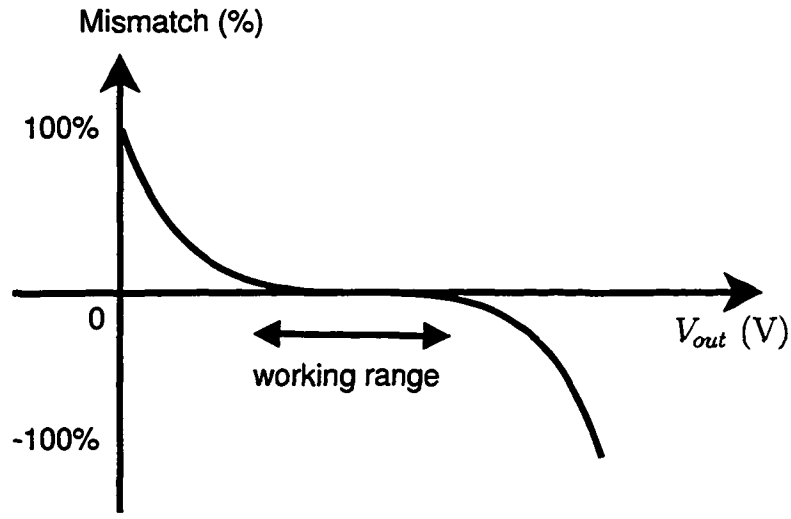


Figure 2.11: Charge pump mismatch.

Another important characteristic of the charge pump is its linearity. Due to the current dependence on the output voltage, negative and positive currents are not perfectly matched, and matching is different for every output voltage level as seen in Figure 2.11. This effect is caused by the short-channel effect, λ, and the minimum V_{ds} of the transistors to keep them in saturation. Given that neither short channel effect nor minimum drain-source voltage can be fully controlled, some non-linearity will always exist.

Another cause of nonlinearity that cannot be avoided is jitter in the digital logic. Since the PFD is an edge sensitive device, every error on the rising edge will be translated into the loop filter. This can be caused by either jitter in the PFD or the divider. If the divider is not synchronized with the VCO at the output, the difference of the delay among division ratios can become a source of nonlinearity. An example of arbitrary nonlinearity in the PFD can be seen in Figure 2.12. Unfortunately, if the mismatch is too great then the error charge introduced into the system could create spurs that can mix the out-of-band noise down to the in-band spectrum.

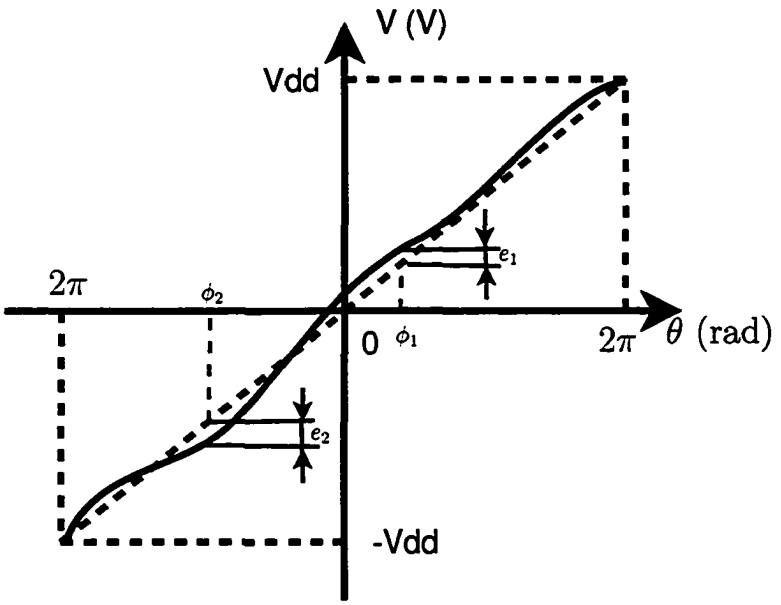


Figure 2.12: PFD nonlinearity.

Over the entire range $[-2\pi, 2\pi]$, Figure 2.12 shows an obvious nonlinear device, however, if one could ensure that only a small fraction of the range would be used in lock, then the linearity of the PFD would appear much better (see Figure 2.13). Since only a fraction of the range is used, K_{pfd} has to be redefined. In this example, the dotted line in 2.13 shows the new K_{pfd}. Nonlinearity becomes of great importance in the fractional-N PLL, given that by design these devices will use a larger range to create a fractional division ratio [27].

2.5 Dividers

Dividers are circuits that scale down the frequency of the input signal to some lower frequency. Topologies for dividers are numerous and their use depends on the frequency and mode of operation. In this thesis project, two types of dividers will be considered: counters (Figure 2.14) and multi-modulus dividers. Counter dividers

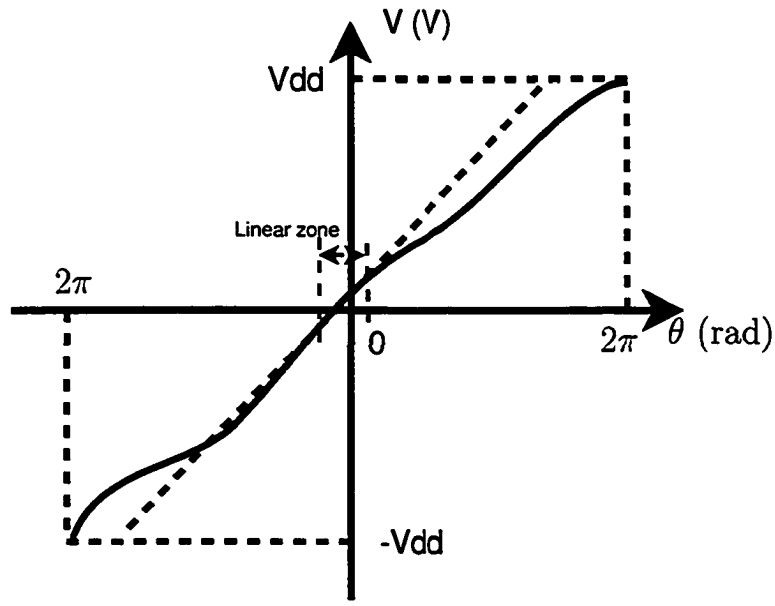


Figure 2.13: Linearity of a partial range in the PFD.

count the number of positive edges from the input signal, which are then compared with a control number. If the control number is reached, the counter resets and starts from zero. The advantage of these dividers is that they can be divided by any number in the range of $[2, 2^N]$, where N is the number of stages in the counter. This can be of great use when the range of divisors has to be large.

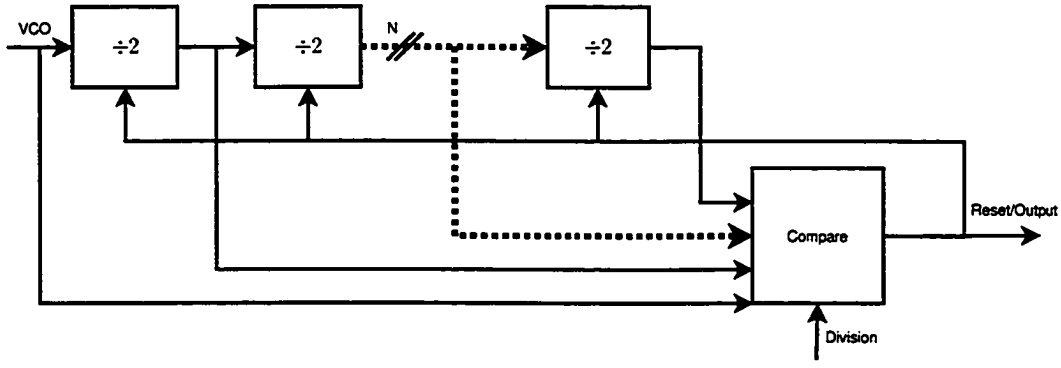


Figure 2.14: Counter divider.

In more common designs, dividers usually need to have only a fraction of the $[2, 2^N]$ range. In such cases, the multi-modulus divider, as shown Figure 2.15, is a better approach. In contrast to the counter divider, the multi-modulus divider is a much simpler circuit since it compares fewer division values. Moreover, the multi-modulus divider often incorporates a *divide-by-2-or-3* in one or more stages (see figure 2.15). This allows for a finer resolution of the divider by controlling how many times the stage divides by 2 and how many times it divides by 3. For example, if a three stage divider is present, where the first stage is a *divide-by-2-or-3* and the rest of them are simple *divide-by-2* stages, it is possible to choose a divider ratio in the range {8,12}. For a divide by 8, the first stage needs to divide by 2 only {2-2-2-2}. For a divide by 9, the first stage needs to divide once by 3 and the rest of the time by 2 {3-2-2-2}, for 10 {3-3-2-2}, and so on. A further advantage of this divider is that there is no need for a reset after the maximum count is reached, thereby avoiding possible races and glitches during reset.

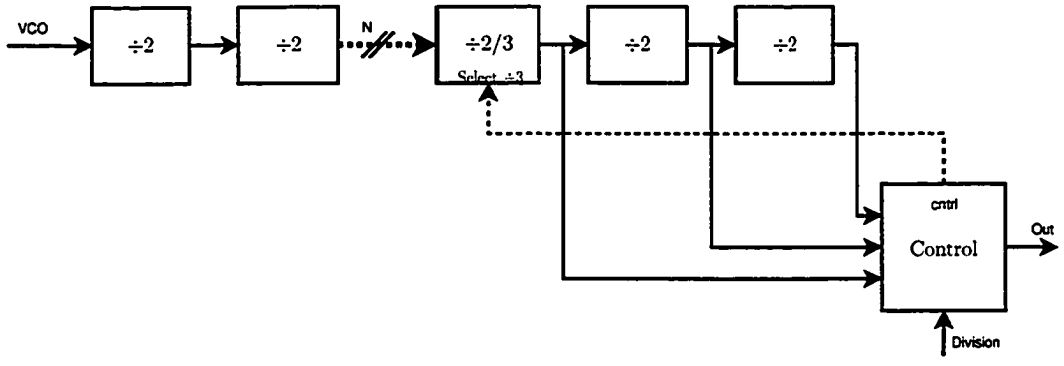


Figure 2.15: Multi-modulus divider.

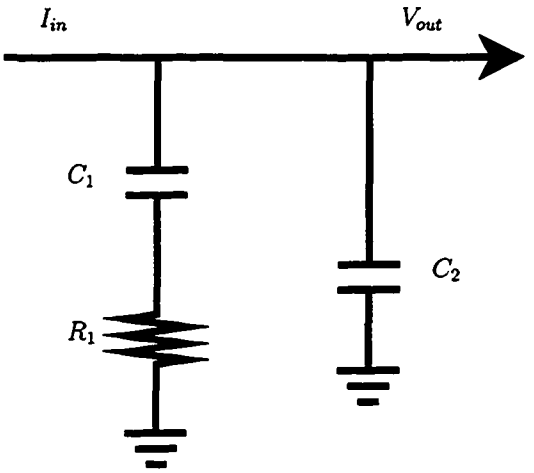


Figure 2.16: Second order loop filter.

2.6 Loop Filter

The loop filter is usually built as a low-pass filter. Its purpose is to extract the average of the PFD error pulses. Both active and passive filters can be used, however, the latter are more frequently used due to noise and power advantages. The loop filter defines the order of the PLL as well as its locking time, bandwidth, and stability.

In the frequency domain, the filter from Figure 2.16 can be represented as:

$$F(s) = \frac{s(C_1 \cdot R_1) + 1}{s^2(C_1 \cdot C_2 \cdot R_1) + sC_1 + sC_2} \tag{2.18}$$

The order and architecture of the filter depends on the design and function of the whole PLL. When using a higher order $\Sigma\Delta$, a higher order filter has to be used to ensure that noise from the $\Sigma\Delta$ is filtered properly. A rule of thumb is to at least match the order of the filter with the order of the $\Sigma\Delta$.

Most of the time, to simplify analysis and design, denominator of the second order system is represented as

$$s^2 + 2\zeta\omega_n s + \omega_n^2$$

where ω_n is natural frequency and ζ is damping constant. Depending on the filter type, proper substitution for ω_n and ζ can be achieved. In presented case from Figure 2.16, it is possible to substitute these variables as :

$$\omega_n^2 = \frac{K}{\tau_1} \tag{2.19}$$

$$2\zeta\omega_n = \frac{1 + K\tau_2}{\tau_1} \tag{2.20}$$

under conditions where $K\tau_2 \gg 1$ and $\frac{\omega_n^2}{K} \ll 1$, [28], where:

$$K = K_{vco}K_{pdf} \tag{2.21}$$

$$\tau_1 = R_1C_1 \tag{2.22}$$

$$\tau_2 = R_1C_2 \tag{2.23}$$

And total transfer function for the system can be expressed as [28]:

$$H(j\omega) = \frac{\omega_n^2 + j2\zeta\omega_n\omega}{\omega^2 + 2\zeta\omega_n\omega + \omega_n^2} \tag{2.24}$$

2.7 Fractional PLL

There are two major reasons why fractional PLLs are used. The first, in cases where the frequency resolution is smaller than the reference frequency and the second where high division ratio has to be avoided due to the high multiplying factor for the noise entering into the PFD [29]. A block diagram of the $\Sigma\Delta$ fractional-N PLL can be seen in Figure 2.2. For the most part, it is equivalent to any other integer-N divider, however the main difference lies in the fact that the divider changes its division ratio after each divided clock output to create a fractional number. Hence, the extra component is a control circuit that changes the ratio of the divider [30]. There are multiple implementations of the control circuit, but the most common is the $\Sigma\Delta$. A $\Sigma\Delta$ modulator takes a DC input and quantizes it in such a way that its average value is equal to its DC input value. In general, the input value does not necessary need to be DC; the $\Sigma\Delta$ will still give a quantized output which follows its input.

The most important issues with $\Sigma\Delta$ fractional-N synthesizers are the quantization noise and spurs. Since the actual resolution of the system equals the reference frequency, every frequency in-between will be quantized to the level of the reference signal. The division ratio alternations are very rapid, and occur in a pseudo-random fashion such that the quantization noise is pushed to a high offset frequencies, where it can be filtered by the loop filter [31]. Since the alternations are not completely random, $\Sigma\Delta$ synthesizers create spurs at the output as well. Also, in cases where the loop filter bandwidth is too high, the quantization noise can dominate other noise sources at higher frequencies.

2.8 Sigma-Delta Modulator

A basic component of the ΣΔ modulator is a uniform quantizer. Usually, the levels of the quantizer are equally spaced with a step Δ. Any input u to the quantizer will be mapped to the closest quantizer output $q(u)$. Consequently, there will be an error which is defined as the quantization error $\epsilon = q(u) - u$. This equation can be written in following form:

$$q_n = q(u_n) = u_n + \epsilon_n \tag{2.25}$$

is also known as the additive noise model of quantization [32]. Thus, the output of the quantizer can be seen as the input plus a noise term.

The quantization region is defined as $B = \{-N/2, N/2 - 1\}$, where N is total number of levels. If the input is in the region B then the maximum quantization error can be defined as $\frac{\Delta}{2}$. Outside of the region B, the quantizer error will be higher than $\frac{\Delta}{2}$ and the quantizer will saturate. For the purpose of this thesis project, it is assumed that the quantizer will not saturate, by ensuring that the inputs of the ΣΔ are in the proper range and that the ΣΔ is stable.

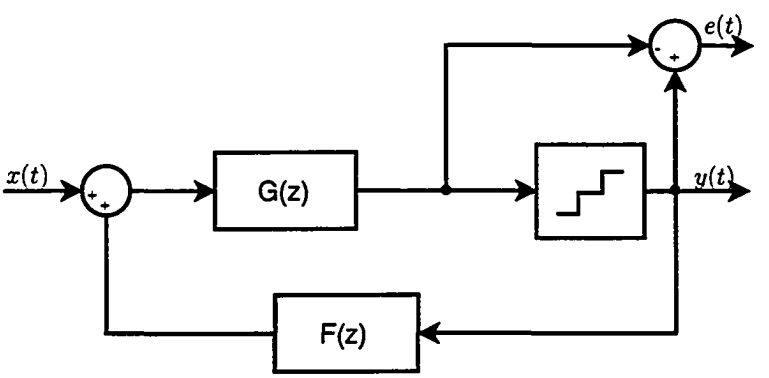


Figure 2.17: General ΣΔ modulator

Figure 2.17 shows a block the diagram of a $\Sigma\Delta$. As seen in the aforementioned figure, the input signal is quantized, then the quantization error is passed through a filter and added to the input, where the process is repeated again. The signal transfer function can be written as follows:

$$STF(z) = \frac{Y(z)}{X(z)} = \frac{G(z)}{1 + F(z) \cdot G(z)} \tag{2.26}$$

Similarly, the noise transfer function can be written as:

$$NTF(z) = \frac{E(z)}{X(z)} = \frac{1}{1 + F(z) \cdot G(z)} \tag{2.27}$$

The $\Sigma\Delta$ topology above is usually simplified by removing the $G(z)$ block. This block is used in cases where the Noise Transfer Function (NTF) needs to be different than the Signal Transfer Function (STF). Hence, the simplified NTF is as follows:

$$NTF(z) = \frac{E(z)}{X(z)} = \frac{1}{1 + F(z)} \tag{2.28}$$

The order of the $\Sigma\Delta$ is defined by the filter order, $F(z)$. In the simplest case, this is a first order low pass filter. If properly chosen, the NTF will have a high pass response while the STF will have a gain of approximately unity in the passband.

Assuming that the quantization noise is additive and uncorrelated to the signal, and that it has a white spectrum, then the in-band noise will be attenuated while the signal magnitude will stay the same according to Equations (2.26) and (2.28). This assumption is not true for the $\Sigma\Delta$ used in a PLL. The input of the $\Sigma\Delta$ in a PLL is a constant rational number. It is observed that the rational DC inputs to the

$\Sigma\Delta$ produce limit cycle oscillations and tones [33],[34] . If tones are produced out-of-band, then it is likely that these components will fold in-band due to the nonlinearity of the PFD and charge pump. This fact underlines the importance of obtaining a very smooth noise shaping without spectral tones. If a Least Significant Bit (LSB) randomizer is added to the input signal, then the quantization noise will become random and uncorrelated to the input signal. Knowing this we can obtain the mean square value of this error as follows [32]:

$$\epsilon_{rms}^2 = \frac{1}{\Delta} \cdot \int_{-\Delta/2}^{\Delta/2} \epsilon^2 d\epsilon = \frac{\Delta^2}{12} \tag{2.29}$$

The two main classifications of $\Sigma\Delta$ are predictive and noise-shaping topologies. This thesis will focus on the noise-shaping topology.

The noise-shaping topology can be seen in Figure 2.18. A general $\Sigma\Delta$ noise-shaper is such that it obeys:

$$Y(z) = X(z) + H_{NS}(Z) \cdot E(z) \tag{2.30}$$

where $Y(Z)$ is the output of the $\Sigma\Delta$, $X(z)$ is the input and $E(z)$ is the quantization error. Here we can see that the output is equal to the input term, $X(z)$, including the quantization noise, $E(z)$ that is shaped by the H_{NS}, filter. Hence, the shape of the quantization noise power spectrum can be controlled by a careful choice of the filter coefficients for H_{NS}. Consequently, the optimal coefficients for this topology are defined by following z-domain equation, [35]:

$$H_{NS}(z) = (1 - z^{-1})^N \tag{2.31}$$

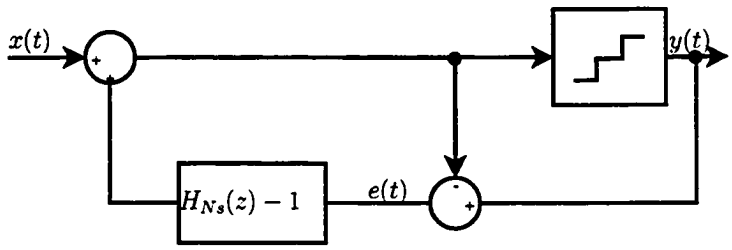


Figure 2.18: Noise-Shaping topology of ΣΔ modulator.

where N is order of the ΣΔ modulator. The actual quantization noise spectrum will depend on the sampling frequency and on the number of stages of ΣΔ modulator. The coefficients from Equation (2.31) ensure that in-band quantization noise is removed. However, in the wide-band PLL, where the bandwidth of loop filter approaches the reference frequency, the in-band term takes a larger range than these coefficients can remove. Thus, in cases like these, it might be necessary to choose coefficients that will attenuate quantization noise over a broader range, while compromising on close-in quantization noise. In general, having in-band quantization noise a bit higher is not an issue since it is usually the charge pump noise that dominates close-in phase noise.

2.9 Reduction of Quantization Noise

Of the many methods to reduce the quantization noise, the most common ones are those that use either a higher sampling rate, higher order ΣΔ, quantization step reduction, or some form of noise cancellation technique.

When a higher sampling rate is used, or in the case of a PLL when a higher reference frequency is used, the alternation speed of the divider ratio also increases.

Consequently, the quantization noise is decreased for the same bandwidth. Unfortunately, this approach means that the circuitry will work at a higher frequency, increasing its power consumption. Moreover, since the reference frequency is higher, the bit order of the $\Sigma\Delta$ has to increase to keep the resolution of the PLL the same. The complexity of the circuit does not increase significantly, but its power consumption usually increases more than 2 times for each sample rate doubling due to the simultaneous doubling of the frequency and bit order increase. Moreover, this method reduces the quantization noise by only $N - \frac{1}{2}$ bits in the $\Sigma\Delta$ for each doubling of the sampling rate [35]. For the second order $\Sigma\Delta$, the noise reduction is 9dB (see Figure 2.19).

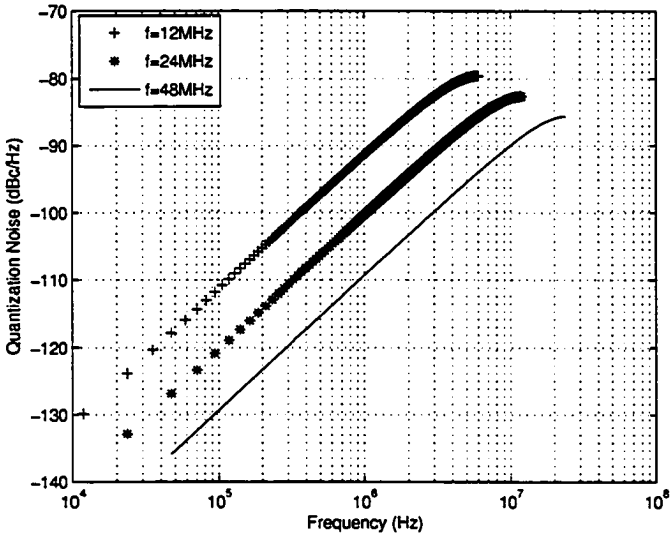


Figure 2.19: Quantization noise vs. sampling rate for second order $\Sigma\Delta$.

If a higher order loop filter is possible and noise reduction is only needed in-band, then a higher order $\Sigma\Delta$ can help reduce the quantization noise. In contrast to increasing the sampling rate, this method increases the complexity of the design. The advantage is that increasing the order from $N = 2$ to $N = 4$ only doubles power,

but, depending on bandwidth, could decrease in-band quantization noise much more than the doubling of the sampling rate, as seen in Figure 2.20.

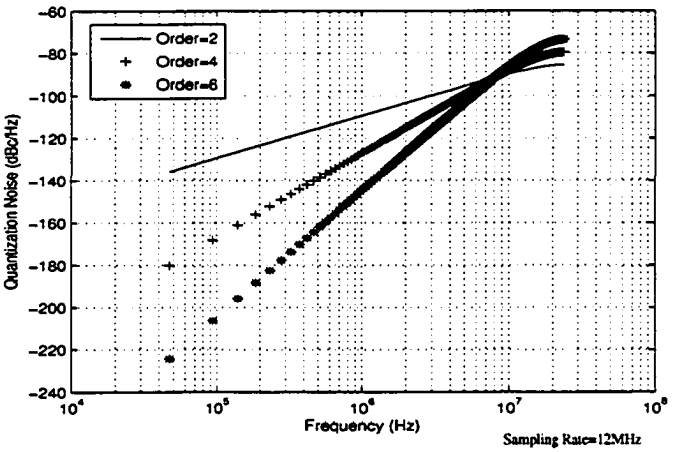


Figure 2.20: Quantization noise of the $\Sigma\Delta$ vs. order of the $\Sigma\Delta$.

The noise cancellation approach uses the predictability of the quantization noise to reduce the amount of error charge put into the loop filter [36]. In an ideal case, it is equivalent to reducing the quantization step, Δ. For example, decreasing Δ by a half is equivalent to canceling one half of the quantization noise. Its advantage over a higher order $\Sigma\Delta$ is that it cancels quantization noise over the whole spectrum. Since the input word to the $\Sigma\Delta$ is known, it is possible to obtain the quantization error by taking the difference of the input and output of the $\Sigma\Delta$. Furthermore, this error can be converted into the equivalent charge error with a DAC and dumped onto the loop filter with the opposite sign. The net quantization error reduction depends on the order of the DAC. For every bit increase in the DAC there is a 6dB decrease in quantization noise. The disadvantage of this approach is an increase in complexity of the circuitry. Moreover, the nonlinearity in the DAC could increase the in-band

phase noise due to the noise folding. Hence, the trade-off between power, order and linearity in the DAC will limit the noise reduction.

This chapter has described the essential components in a PLL. Behavior and mathematical models of each component have been detailed. Different methods of reducing quantization noise in the $\Sigma\Delta$ PLL have been introduced. The following chapter will expand on this knowledge by providing a summary on noise contributions from each component. In addition, it will introduce the topic of quantization noise reduction by the use of the multi-phase characteristic of the ring oscillator.

Chapter 3

Analysis of the Phase Locked Loop

3.1 Introduction

This chapter provides a system level analysis of a PLL. Various equations are formulated that will be used in the design of a PLL. It starts with the transfer function for the feedback system, then it expands it to a PLL application. Subsequently, details for the transfer function of each of the components are derived which explain the type of noise introduced into the system and its effect on the PLL output. Finally, the chapter puts all the equations together and provides a mathematical model for the total noise of the PLL that includes the noise contribution of every circuit in the system.

3.2 System Level Analysis

Figure 2.1 shows the block diagram of a standard PLL. The closed loop function for the feedback system in Figure 2.1 is as follows:

$$G(s) = \frac{A(s)}{1 + A(s)} \tag{3.1}$$

where $G(s)$ is the closed loop transfer function and $A(s)$ is the open loop transfer function. By considering Figure 3.1, while ignoring the $\Sigma\Delta$ block, it can be shown that $A(s)$ is given as follows:

$$A(s) = K_{pfd} \cdot F(s) \cdot \frac{K_{vco}}{s} \cdot \frac{1}{N} \tag{3.2}$$

and the closed loop equation becomes:

$$G(s) = \frac{K_{pfd} \cdot F(s) \cdot \frac{K_{vco}}{s} \cdot \frac{1}{N}}{1 + K_{pfd} \cdot F(s) \cdot \frac{K_{vco}}{s} \cdot \frac{1}{N}} = \frac{K_{pfd} \cdot F(s) \cdot K_{vco}}{sN + K_{pfd} \cdot F(s) \cdot K_{vco}} \tag{3.3}$$

Assuming that $F(s) = 1$ (i.e. loop without filter), then $A(s)$ is a low pass filter, due to the $\frac{1}{s}$ term. Therefore, $G(s)$ goes to zero for high frequencies and to one for very low frequencies, implying that $G(s)$ is a low pass filter as well. Equation (3.3) shows that the loop adds one more order to the system. For instance, if the loop filter $F(s)$ is of 2^{nd} order then the system is of 3^{rd} order.

To ensure stability in the system, it is necessary that the open loop phase is less than 180^o, while the gain is unity. The difference between the open loop phase and 180^o is called the phase margin. An example of the open loop magnitude and phase response with a second order loop filter, as shown in Figure 2.16, can be seen

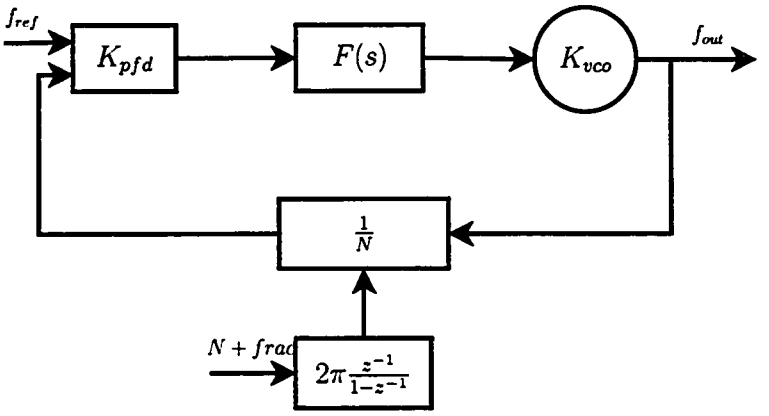


Figure 3.1: PLL Block Diagram

in Figure 3.2. The phase margin is very sensitive to the gain of the loop. Significant changes in the gain of the loop will shift the zero dB crossing point and reduce the phase margin.

Additional parasitics can further change the response of the loop. Each parasitic adds a pole to the open loop response as seen in Equation (3.4). In proper designs, the poles caused by parasitics are outside the loop bandwidth and it is possible to ignore them without any consequence to the predicted circuit performance.

$$A_{parasitic}(s) = A(s) \cdot \left(\frac{1}{1 + \frac{s}{\omega_p}} \right) \tag{3.4}$$

The closed loop response can be seen in Figure 3.3. The drop-off of the slope is a function of the loop filter order. Peaking around the $3dB$ point is caused by the mismatched zero and the pole in the transfer function. Moreover, peaking can be amplified by parasitic poles and zeros, which is undesirable in systems that require flat gain response.

Besides knowing the closed loop transfer function, to fully analyze the system it is necessary to know how noise from every source influences the system.

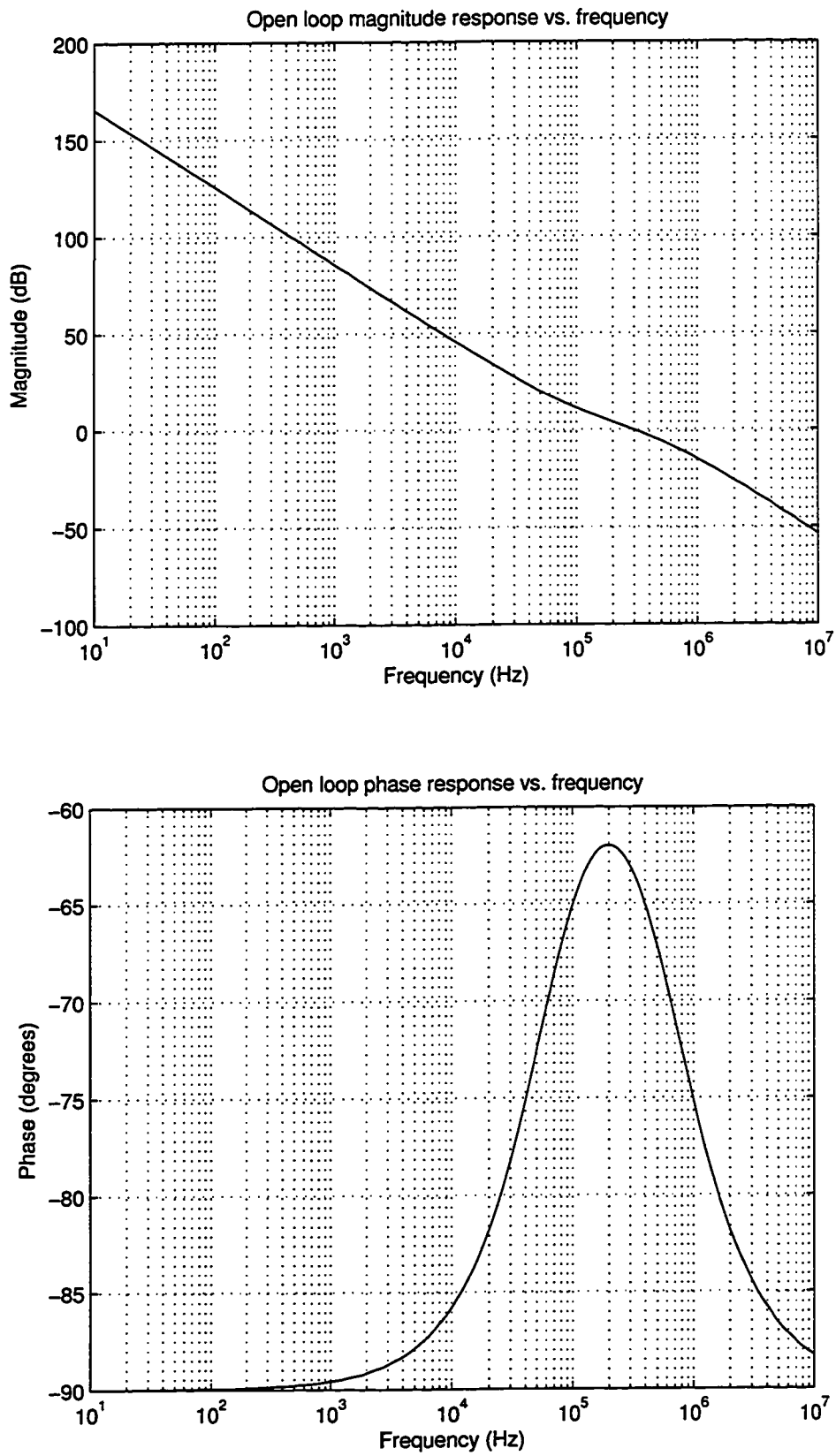


Figure 3.2: Open loop gain and magnitude.

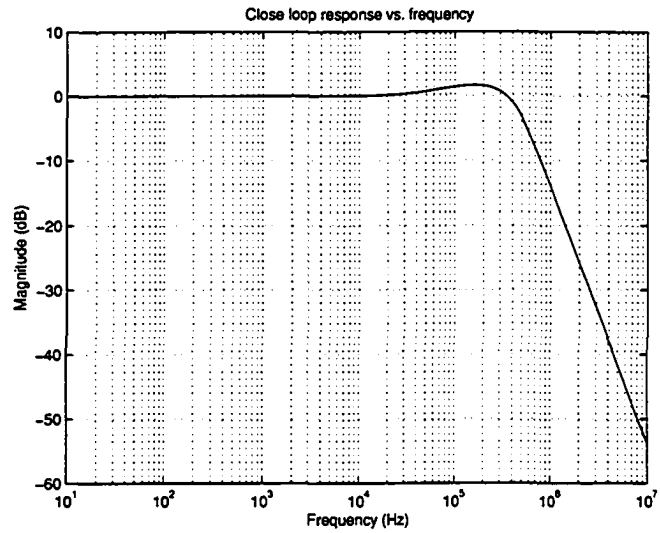


Figure 3.3: Closed loop response of the PLL
with a 2^{nd} order loop and loop bandwidth of 200kHz.

3.3 Phase Noise and Spurs

Poor phase noise can degrade the system performance. In the case of the receiver, it reduces sensitivity due to the phase noise skirts (see Figure 3.4). Likewise, on the transmitter side, it increases the spectral emissions that can affect adjacent channels. The spectral emission specifications are usually limited and very rigid, hence having high phase noise could result in the specifications not being met.

While phase noise is a non-periodic phenomena, spurs are periodic in nature. Phase noise is described as spectral density relative to carrier power (see Equation (3.5)). It is given in units of $\frac{dBc}{Hz}$, which represents the power of the noise relative to the signal power at a specified frequency offset.

$$L(f) = 10 \cdot log(S_{\Phi out}(f)) \left[\frac{dBc}{Hz}\right] \tag{3.5}$$

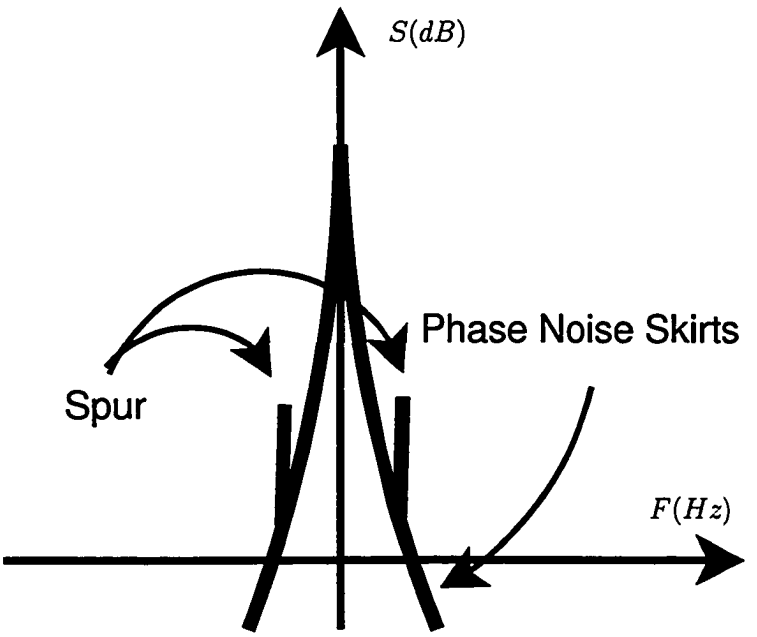


Figure 3.4: Phase noise skirt.

Spurs are defined as tone power relative to carrier power.

$$20 \cdot log\left(\frac{2 \cdot A_{spur}}{A_{carrier}}\right)[dBc] \tag{3.6}$$

where A_{spur} and $A_{carrier}$ are magnitudes of the spur and the carrier respectively. In the frequency domain they appear as tones (see Figure 3.4) that could mix down phase noise to in-band frequencies if the non-linearity of the system is significant. Hence, it is important to ensure that the combination of phase noise, spurs and nonlinearity does not cause a significant mixing of the noise in-band.

In the time domain, noise $\theta(t)$ can be represented as a phase change in the oscillation of the oscillator.

$$V_{out}(t) = cos(\omega \cdot t + \theta(t)) \tag{3.7}$$

Since the phase noise is small, $\theta(t) \ll 1$, it can be assumed that:

$$\begin{aligned} cos(\theta(t)) &\approx 1 \\ sin(\theta(t)) &\approx \theta(t) \end{aligned}$$

and using trigonometric identities Equation (3.7) can be modified to:

$$V_{out}(t) = cos(\omega \cdot t) - sin(\omega \cdot t)\theta(t) \tag{3.8}$$

Furthermore, converting this into Laplace domain:

$$S_{out}(s) = L\left(cos(\omega \cdot t)\right) - L\left(sin(\omega \cdot t)\right) * L\left(\theta(t)\right) \tag{3.9}$$

where L denominates Laplace transform and symbol $*$ represents the convolution operator. This means that the output signal will be made of a sine wave at the desired frequency convolved with the phase noise and/or spurs. This is equivalent to the representation in Figure 3.4.

Each component of the PLL can also be a noise source. Common sources of PLL noise are:

- Reference jitter
- Reference feed-through
- Charge pump noise
- VCO noise
- Divider jitter and

- $\Sigma\Delta$ noise

Furthermore, the output spectrum of the noise is shaped by the PLL system transfer function. Since each source of the noise is injected at a different point in the system, see Figure 3.5, the system will influence each source differently. Therefore, the transfer function for each noise source will be developed.

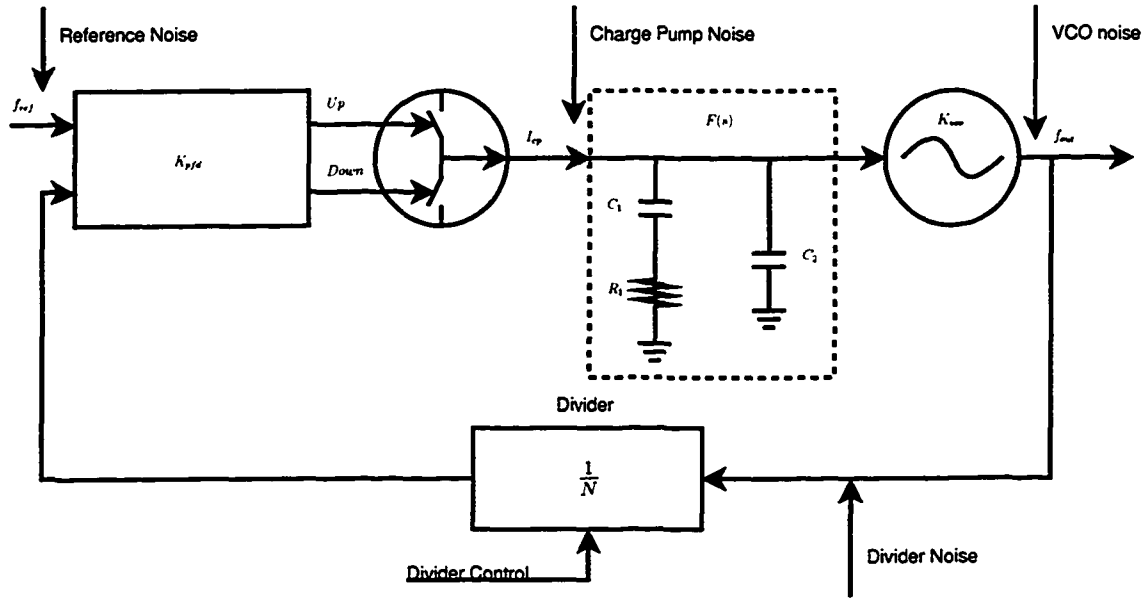


Figure 3.5: PLL block diagram with noise sources.

3.3.1 Phase Noise Due to Reference and Divider Jitter

Since the reference signal itself is an oscillator it will inherently have phase noise. Reference signals are typically based on a crystal oscillators with very narrow skirts. Their phase noise is dominated by the thermal noise, which is represented as a straight line in the frequency domain. It is not uncommon for crystal oscillators to have flat phase noise response as low as $-150\frac{dBc}{Hz}$ for the frequency offset of interest [37].

In order to determine how noise from the reference influences the system we can refer to Equation (3.12). The derivation can be seen below:

$$G_{ref}(s) = \frac{ForwardGain}{1 + OpenLoopGain} \tag{3.10}$$

$$G_{ref}(s) = \frac{K_{pfd}F(s)\frac{K_{vco}}{s}}{1 + \frac{1}{N}K_{pfd}F(s)\frac{K_{vco}}{s}} = NG(s) \tag{3.11}$$

where $G_{ref}(s)$ is the transfer function from input of the reference signal to the output of the VCO, and $G(s)$ is closed loop function of the system from Equation (3.3). Therefore, the transfer function can be written as follows:

$$G_{ref}(s) = N \cdot G(s) \tag{3.12}$$

First, given that the loop is a low pass filter, it can be expected that the signal will drop by the loop order after the $3dB$ point. Furthermore, since there is a divider in the loop, the reference signal will be multiplied by the ratio N (as seen in Figure 3.6). Hence, unless the divisor ratio of the loop is very high, for the practical implementations the reference noise is usually not an issue for the system performance.

Similar to the reference noise, the divider jitter noise is a function of thermal noise. Hence, it can also be represented by a straight line in frequency domain (Figure 3.6). Noise itself is a function of the circuit complexity in the divider. Moreover, given that the noise can be referred to the input of the divider (see Figure 3.5), the same transfer function can be used as in Equation (3.12) but multiplied by a $\frac{1}{N}$ factor in order to refer it to the input of the PFD. Thus, the transfer function for the divider can be written as:

$$G_{div}(S) = \frac{1}{N} \cdot G_{ref}(s) \tag{3.13}$$

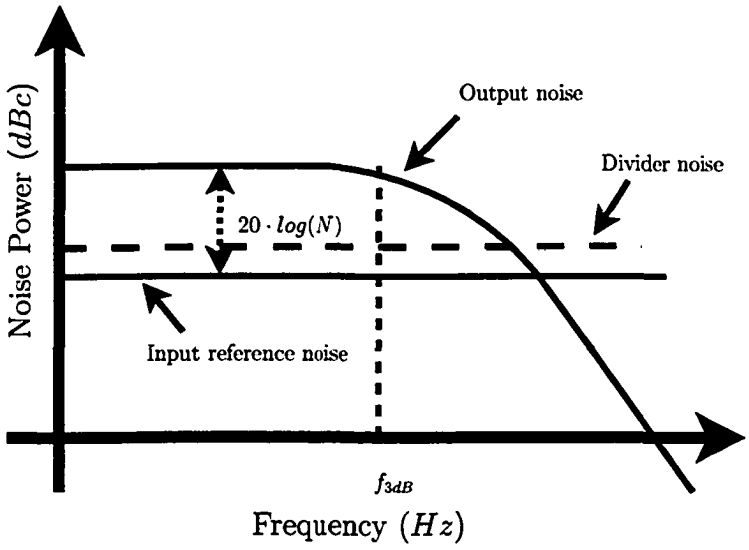


Figure 3.6: Reference signal noise

From the equation above, it can be seen that the divider jitter is independent of the division ratio. It is rather dependent on the number of gates that the VCO signal needs to pass through to get divided.

3.3.2 Reference Feed-through Noise

The reference feed-through noise refers to spurs that appear at the output of the VCO at frequency offsets equal to multiples of the reference frequency. The common cause of most spurs in the PLL is mismatch in the PFD speed and charge pump current. If one side of the PFD is faster than the other, the loop will create a periodic pattern to compensate for the extra charge dumped on the filter. Similarly, if the pull-up current is larger than the pull-down current (Figure 3.7), the loop will compensate by creating a phase offset between the reference and divided VCO signals to remove the extra charge introduced by the mismatch.

Mathematically, this can be written as follows:

$$i_{up} = I_{cp}\left(1+\frac{\epsilon}{2}\right) \quad , \quad i_{dn} = I_{cp}\left(1-\frac{\epsilon}{2}\right) \tag{3.14}$$

where I_{cp} is the ideal current magnitude and ϵ is the total error current in percent. Let's assume that the phase delay at the PFD is

$$T_{error} = T_{ref} - T_{div}$$

this means that the pulse width will be equal to T_{error} seconds, and the charge, q, dumped on the loop filter will be:

$$Q = \begin{cases} i_{up}T_{error}, & if \quad T_{error} > 0 \\ i_{down}T_{error}, & if \quad T_{error} < 0 \end{cases} \tag{3.15}$$

Since the currents are not matched, extra charge will be dumped on the loop during the reset delay in the PFD [27]. Hence, the total charge becomes as follows:

$$Q_{total} = I_{cp}T_{error} + I_{cp}\frac{\epsilon}{2}T_{error} + I_{cp}\epsilon T_{delay} \tag{3.16}$$

where T_{delay} is the reset delay in seconds. The $I_{cp}T_{error}$ term in Equation (3.16), represents the ideal model for the PFD and charge pump, while the $I_{cp}\epsilon T_{delay}$ term represents charge introduced by the current mismatch during the reset delay. This is a constant charge that adds to the phase offset. The middle term is the one that can cause problems in a $\Sigma\Delta$, since the error introduced during the pull-up or pull-down signals depends on division ratio changes. However, in integer-N PLLs, the

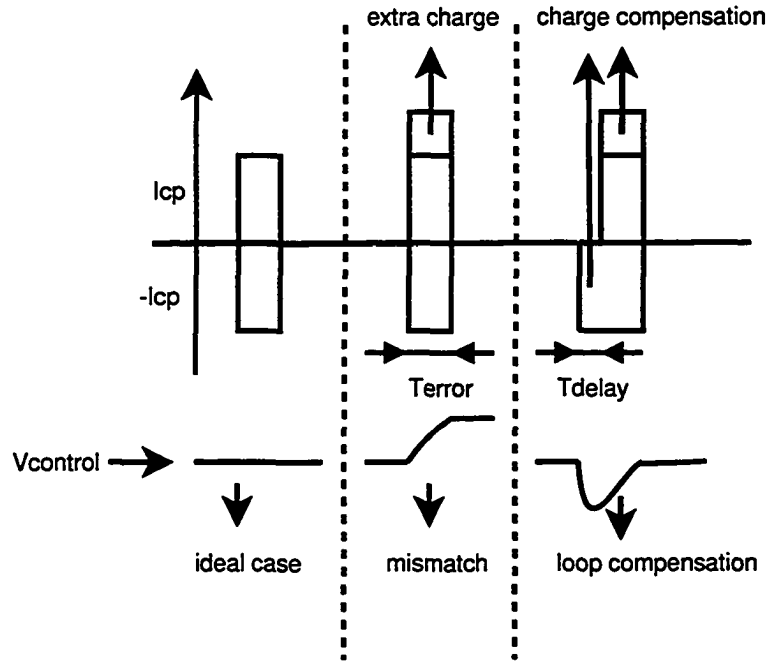


Figure 3.7: Loop compensation due charge pump mismatch.

error only adds to the phase offset due to the constant signal width. This creates a blip on the control voltage that repeats every reference period, as seen in Figure 3.7. Consequently, this will translate into the reference spurs on the VCO output. Similar to the reference noise, those spurs will be shaped by the transfer function from Equation (3.1). Therefore, aside from designing the circuit to have minimal mismatch in the PFD and charge pump, it is possible to use the loop filter to suppress most of the spurs. By controlling the bandwidth and the order of the filter, it is possible to control the degree of spurs suppression.

Unfortunately, the loop order and bandwidth cannot be used to remove the noise folded in-band due to spurs in the $\Sigma\Delta$ PLLs, since the folding usually happens inside of a non-linear device, which in this case is the charge pump or the PFD. The signal that comes into the filter has the noise folded already. In order to avoid noise

folding, it is therefore important to increase the linearity of the devices (e.g.: better matching of currents in the charge pump).

3.3.3 Charge Pump Noise

The standard charge pump is built with a pair of current mirrors, where one is turned on for the pull-up state and the other one is turned on for the pull-down state. Therefore, the noise performance of the charge pump depends on the performance of the current mirrors. From [38], the major contributor of current mirror noise is thermal noise, which is a function of the current mirror gain and complexity [38]. In general, thermal noise performance is best in a basic current mirror because a minimal number of components are being used [38].

As the bandwidth of the loop filter decreases, a significant part of the noise at the output of the filter comes from flicker noise. This is usually the case with the charge pump in the PLL, since the bandwidth of the PLL is relatively limited. The source of flicker noise is a random trapping and releasing of charge. This trapping process happens at the oxide interface of the MOSFET gate and causes fluctuations in the surface potential. This, in turn, modulates the channel carrier density and, due to a secondary effect, Coulombic scattering [39], causes fluctuations in the surface mobility. Because of the slow time constant of oxide trapping and releasing, the flicker noise is inversely proportional to frequency and hence often referred to as $1/f$ noise.

Given that the flicker noise is caused by various physical defects in the material, it is not dependent on the bias conditions for long devices. However, in case of sub-micron devices, there are reports that there exist a gate voltage dependence [39].

The influence of charge pump noise on the output of the PLL can be found in the following equation:

$$G_{cp}(s) = \frac{2\pi}{I_{cp}} \cdot N \cdot G(s) \tag{3.17}$$

This transfer function represents the charge pump noise referred to the PFD input. The amount of noise introduced into the system will be dependent on the system bandwidth.

3.3.4 VCO Noise

From Figure 3.5 we can define the transfer function for VCO noise at output of the PLL as follows:

$$G_{vco}(s) = \frac{1}{1 + A(s)} = 1 - G(s) \tag{3.18}$$

In contrast to other blocks, the phase noise from the VCO will be high-pass filtered and its transition frequency is equivalent to the low pass bandwidth of the loop.

The oscillator can be represented by Equation (3.7), where the amplitude term is purposely omitted. The reason for this is that amplitude fluctuations are attenuated by the amplitude limiting mechanism, which is particularly strong in ring oscillators. A good approximation of ring oscillator noise can therefore be achieved by focusing only on phase variations.

Considering the single-ended oscillator from Figure 2.5, during charge or discharge, a current pulse with area Δq, which represents the equivalent charge, is expected to cause an instantaneous change in the voltage ΔV as follows:

$$\Delta V = \frac{\Delta q}{C} \tag{3.19}$$

where C is the capacitance of the next stage. For a very small ΔV, the change in phase, $\phi(t)$, will be proportional to the injected charge:

$$\Delta\phi(t) = \Gamma(\omega_0 t)\frac{\Delta q}{q_{max}} \tag{3.20}$$

where q_{max} is a maximum charge across the capacitor on the next stage and $\Gamma(\omega_0 t)$ is the Impulse Sensitivity Function (ISF). $\Gamma(\omega_0 t)$ represents the sensitivity of the system to the current impulse [40]. The ISF is a periodic function of period $2\pi f$, where f is the oscillation frequency of the circuit. In order to understand the time dependence and periodicity of the ISF we can consider two extreme cases:

- **Case 1:** When the output voltage is already switched high or low, any current impulse will have a minimal effect on phase noise due to the gain saturation and the fact that the threshold voltage of the next gate is much further than the current value of the output.
- **Case 2:** When the output voltage is in transition, even the smallest current impulse will have a degrading effect on the phase of the oscillator. Hence, the ISF will have highest value during transition times.

Both cases are illustrated in Figure 3.8.

According to [41], the total phase noise for one node is expressed as follows:

$$L(\Delta\omega) = 10 \cdot log\left(\frac{\Gamma_{rms}^2}{q_{max}^2} \cdot \frac{\overline{i_n^2}/\Delta f}{4 \cdot \Delta\omega^2}\right) \tag{3.21}$$

where Γ_{rms}^2 is RMS value of $\Gamma(\omega t)$ and $\overline{i_n^2}/\Delta f$ is the noise current of the input. Equation (3.21) represents the noise in the $1/f^2$ region of the oscillator. For a full ring

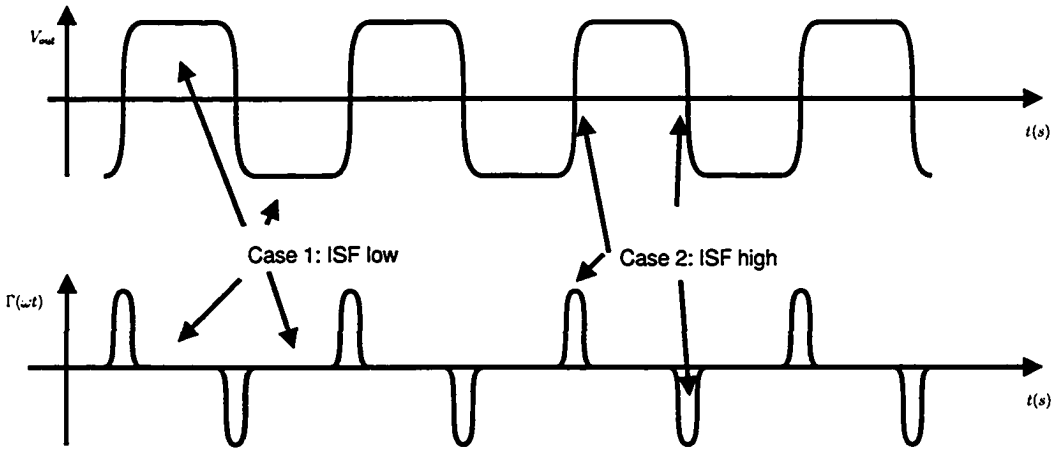


Figure 3.8: Impulse sensitivity function in ring oscillators.

oscillator it is necessary to multiply the log part of Equation (3.21) with the number of stages N, or $2N$ for a differential oscillator. However, in order to fully model the noise in the oscillator, the attributes of the ISF need to be fully understood. Since this is a periodic function, its Fourier transform will give the following:

$$\Gamma(\omega_0 t) = \frac{c_0}{2} + \sum_{n=1}^{\infty} c_n \cos(n\omega_0 \tau + \theta_n) \tag{3.22}$$

where c_n are the coefficients for the n^{th} harmonic of the oscillator. When Equations (3.22) and (3.20) are combined, it can be shown that the noise around DC, ω_0 and its harmonics will significantly contribute to the total phase change in the oscillator [41] because the components around those frequencies are converted directly to the ω_0 frequency. Moreover, for the noise around the DC term, the c_0 component of the ISF will define what portion of it will be up-converted to the oscillator noise. Similarly, for the noise around $n\omega_0$, the c_n term will influence its up-conversion. In addition, any $1/f$ noise from the current noise sources will appear as $1/f^3$ in the oscillator response (see Figure 3.9). Since it is close to DC, the c_0 term will define how much of

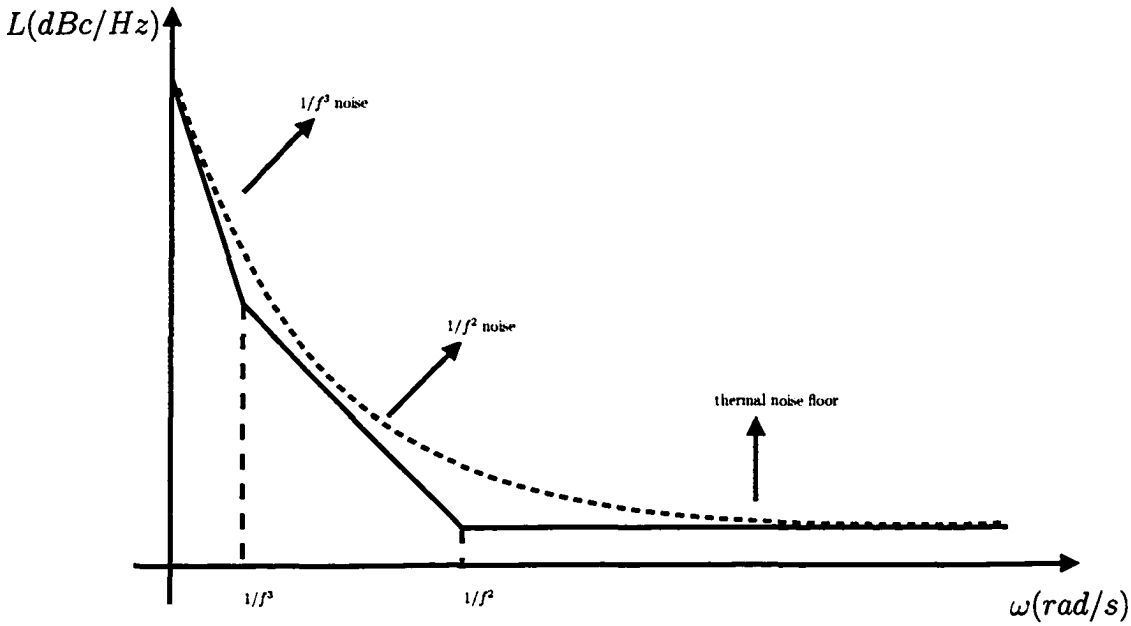


Figure 3.9: Standard VCO noise response.

it will be up-converted to phase noise. It is therefore possible to control the amount of up-converted DC noise by insuring that the DC term of $\Gamma(\omega_0 t)$ is small. Moreover, a symmetrical ISF will have small c_0 term. Symmetry can be achieved with equal rise and fall times [41], however, this does not mean that a differential ring oscillator will have a small c_0. Even though the differential output is symmetrical, the noise itself is inserted in one node at a time (i.e. noise sources are uncorrelated for each half of the circuit). This implies that each half of the circuit must have an equivalent rise and fall time in order to decrease the c_0 term. The corner frequency for the $1/f^3$ term can be found from [40]

$$f_{1/f^3} = f_{1/f} \cdot \frac{\Gamma_{dc}^2}{\Gamma_{rms}^2} \tag{3.23}$$

where Γ_{dc} is the DC value of the ISF. Equation (3.23) shows that flicker noise in the ring oscillator is defined by Γ_{dc}, which is the average value of the ISF.

In a CMOS ring oscillator the c_0 term is very important since flicker noise in MOSFET can be significant. In general, the designer usually only needs to consider the c_0 and c_1 terms. The rest of the terms are usually an order of magnitude lower.

3.3.5 ΣΔ Noise

As shown in Equation (2.29), a mean RMS value for ΣΔ noise for a given quantization step Δ can be calculated. In the case of the ΣΔ PLL, this basically means that the quantization error will have a variance of $1/12$ over the F_{ref} bandwidth. Furthermore, the power spectral density (PSD) of the quantization noise can be represented as follows:

$$e_q = \frac{1}{12 \cdot F_{ref}} \tag{3.24}$$

This noise will be shaped over the spectrum by the NTF. In an ideal case where the NTF is $(1 - z^{-1})^n$ and a ΣΔ noise shaper is used (see Equation (2.30)), the ΣΔ quantization spectrum is represented by the following equation [27]:

$$S(z) = |(1 - z^{-1})^n F_{ref}|^2 \times \left(\frac{1}{12F_{ref}}\right) = |(1 - z^{-1})^{2n}| \times \frac{F_{ref}}{12} \tag{3.25}$$

Since the phase noise shape is more relevant than the frequency fluctuation, the frequency noise can be integrated in order to get phase noise. In the discrete domain, the integration is represented by $2\pi T_{ref}/(1 - z^{-1})$, where T_{ref} is a sampling rate. Therefore,

$$S_\phi(z) = \frac{(2\pi)^2}{|1 - z^{-1}|^2 F_{ref}^2} \times \left[|(1 - z^{-1})^{2n}| \times \frac{F_{ref}}{12}\right] = \frac{(2\pi)^2}{12F_{ref}}|1 - z^{-1}|^{2n-2} \tag{3.26}$$

By converting this to the frequency domain (replacing z with $e^{j\pi\frac{f}{F_{ref}}}$), the following equation is achieved:

$$S_\phi(f) = \frac{(2\pi)^2}{12F_{ref}}\left(2\sin\left(\frac{\pi f}{F_{ref}}\right)\right)^{2n-2} \tag{3.27}$$

where n is the order of the $\Sigma\Delta$. Usually a factor β is added to the gain to compensate for dithering in the $\Sigma\Delta$. Typical β values range from 1.2 to 1.4 [27]. A limitation of Equation (3.27) is that it does not include a mechanism to deal with the nonlinearities in the PLL, such as the current mismatch in the charge pump.

From Equation (3.16) it is obvious that the charge pump current has an ideal term and an error term. For the ideal term, Equation (3.27) can predict the exact shape and magnitude of the $\Sigma\Delta$ phase noise. However, the error term will introduce noise folding which will raise the noise floor (see Figure 3.10). In order to find the noise floor due to this term, it is necessary to assume that current pulses at the output of the charge pump have a Gaussian distribution and a mean of zero when the PLL is locked [42]. It will then be possible to calculate the variance for different orders of $\Sigma\Delta$ as given in Table 3.1 [27].

	n=1	n=2	n=3	n=4	n=5
σ^2/T_{ref}^2	1/12	1/6	1/2	5/3	35/6

Table 3.1: Variance for different orders of $\Sigma\Delta$.

The actual noise floor due to the noise folding can be calculated with the equation from [27]:

$$S_{folding} = \left(\frac{\epsilon}{2}\right)^2 \frac{(2\pi)^2}{F_{ref}}\left(1-\frac{2}{\pi}\right) \times \frac{\sigma^2}{T_{ref}^2} \tag{3.28}$$

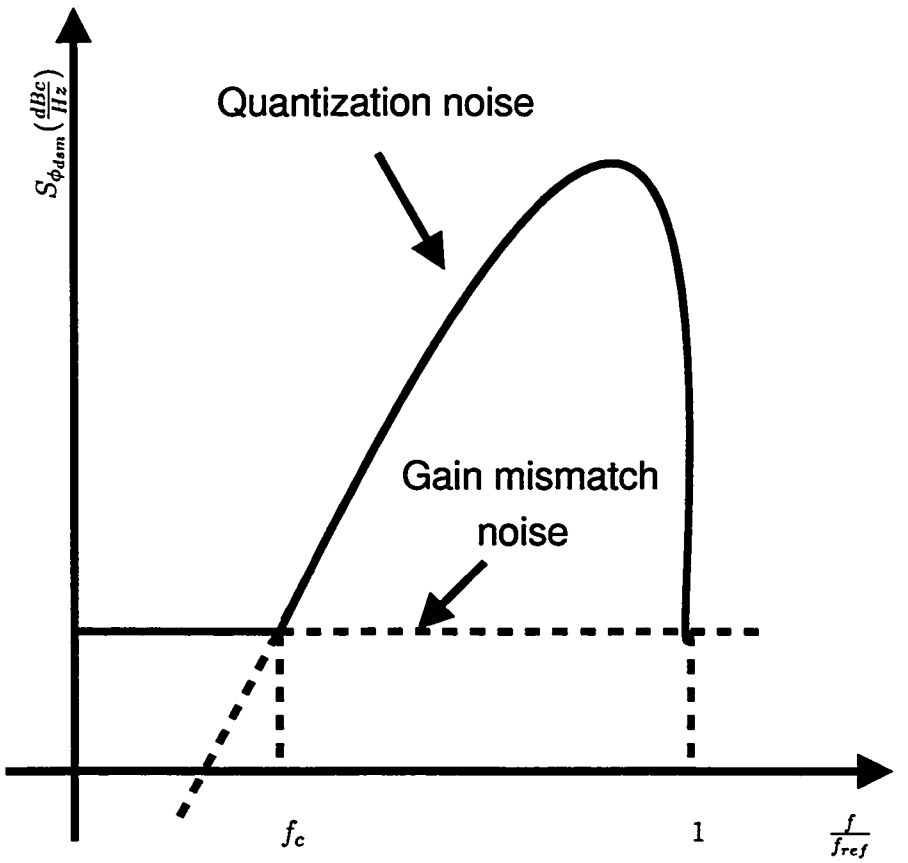


Figure 3.10: Noise folding due to charge pump mismatch in ΣΔ PLLs

Thus, in order to reduce noise folding, the current error ϵ in the charge pump has to be reduced as well. Some other options would include decreasing the order of the $\Sigma\Delta$ or increasing the reference frequency. The total phase noise at the output of the PLL due to the $\Sigma\Delta$ and the charge pump mismatch can be expressed as follows:

$$S_{\Sigma\Delta_{total}}(f) = [S_{\phi}(f) + S_{folding}] \cdot G(j2\pi f) \tag{3.29}$$

3.3.6 Summary of the Noise Sources

In order to calculate the total phase noise of the PLL, each noise source has to be shaped by the appropriate loop transfer function and added to the total output noise. As Figure 3.11 shows, each noise source will depend on the $G(s)$ factor. Therefore, the loop parameters for phase noise are of significant importance. In order to aid the design of the PLL, a mathematical model for the total noise at the PLL output can be developed based on Figure 3.11 and the equations developed in previous sections. Once all the noise from different circuits is known from simulation results, it can be

shown that total noise is represented by equation (3.30).

$$
\begin{aligned}
S_{total}(s) =& 10 \cdot \log\left(N_{Icp} \times \frac{2\pi}{I_{cp}} \cdot N \cdot G(s)\right) && Charge-Pump\ \ Noise \\
&+ 10 \cdot \log\left(N_{div} \times \frac{1}{N} \cdot G(s)\right) && Divider\ \ Noise \\
&+ 10 \cdot \log\left(N_{ref} \times G(s)\right) && Reference\ \ Noise \\
&+ 10 \cdot \log\left(N_{vco} \times [1 - G(s)]\right) && VCO\ \ Noise \\
&+ 10 \cdot \log\left(N_{sdm} \times 2\pi \frac{z^{-1}}{1 - z^{-1}} T_{ref} \cdot G(s)\right) && \Sigma\Delta\ \ Noise \\
&+ 10 \cdot \log\left(S_{folding}\right) && Folded\ \ Noise
\end{aligned}
\tag{3.30}
$$

where:

- N_{icp} is noise from the charge pump in $\frac{V^2}{Hz}$
- N_{div} is noise from the divider
- N_{ref} is the reference noise
- N_{vco} is the VCO noise, etc.

In this chapter, a mathematical model for each of the circuit components was developed. This model includes the influence of the charge pump's nonlinearity. In the next chapter, these mathematical equations will be used to verify if any of the components are not meeting the design specifications. Moreover, this model will be extended to include the multiphase VCO and static mismatch among the phases.

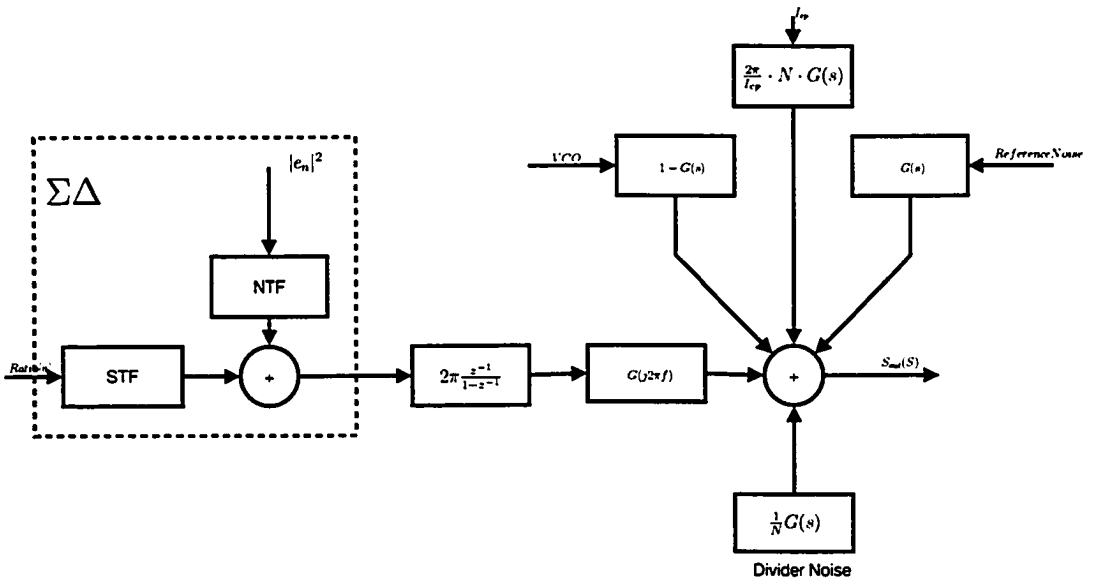


Figure 3.11: All noise sources form PLL

Chapter 4

Design

Conceptually a low frequency, low power PLL is used to control an injection locked VCO [43] at high frequencies using a harmonic of the PLL (see Figure 4.1). For example, tuning to a $2.44GHz$ RF frequency requires the PLL to be tuned to $2.44G/21 \approx 114.285MHz$, if the 21^{st} harmonic is to be used. To achieve a precision of at least $100kHz$ in the LO (e.g. $2.44GHz \pm 100kHz$), the PLL needs to have a minimum step of $100K/21 \approx 4.76k$. This is what defines the resolution of the $\Sigma\Delta$. Given that a harmonic of a PLL can be used as part of an LO generator, a PLL can be used in a device that operates for example at $2.4GHz$ or $900MHz$ by using a different harmonic or frequency of the PLL.

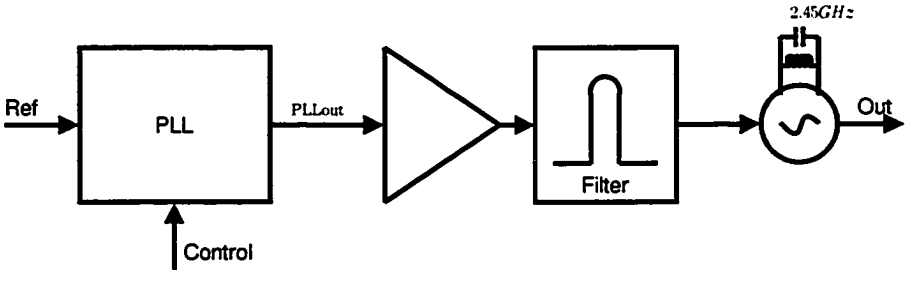


Figure 4.1: PLL with injection locked oscillator.

For the PLL described above, a reference signal $F_{ref} = 12MHz$ is assumed. For this application the output spectrum has to comply with the IEEE 802.15.4 standard [44]. Specifically, the error vector magnitude (EVM) of the system in 802.15.4 standard is defined as 35%. EVM is defined as the difference between an ideal signal vector and an observed signal vector (see Figure 4.2). Therefore, if EVM is 35%, using trigonometric equalities it is possible to calculate the maximum phase error as:

$$\sin\left(\frac{Error\_vector}{Ideal\_signal}\right) = 0.35 \tag{4.1}$$

$$\phi = \arcsin(0.35) \approx 20.5^o \tag{4.2}$$

If EVM is set to 35%, the calculated equivalent RMS jitter from Equation (4.1) is 20.5^o. Assuming that half the phase error (11^o) is assigned to PLL jitter, it is possible to define phase noise performance of the PLL. If the 21^{st} harmonic is used, the jitter of the PLL needs to be $11^o/21 = 0.53^o$. As a design aid, this jitter can be converted to phase mask. For simplicity, it is assumed that the phase noise will be flat in-band, then it will start to drop off quickly. The jitter is give by:

$$\sigma = \sqrt{10^{\int_0^{BW} L(f)\cdot df}} \tag{4.3}$$

where $L(f)$ is phase noise at frequency f. It is found that for a flat phase in-band noise of $-99dBc/Hz$, maximum possible bandwidth is $300kHz$ if the 0.53^o spec was meant to be met. Hence, the spectrum mask for this PLL is expected to look as the one shown in Figure 4.3.

In general, the PLLs design and architecture are defined by the phase noise performance of the VCO. Having a low VCO noise allows for more flexibility in the

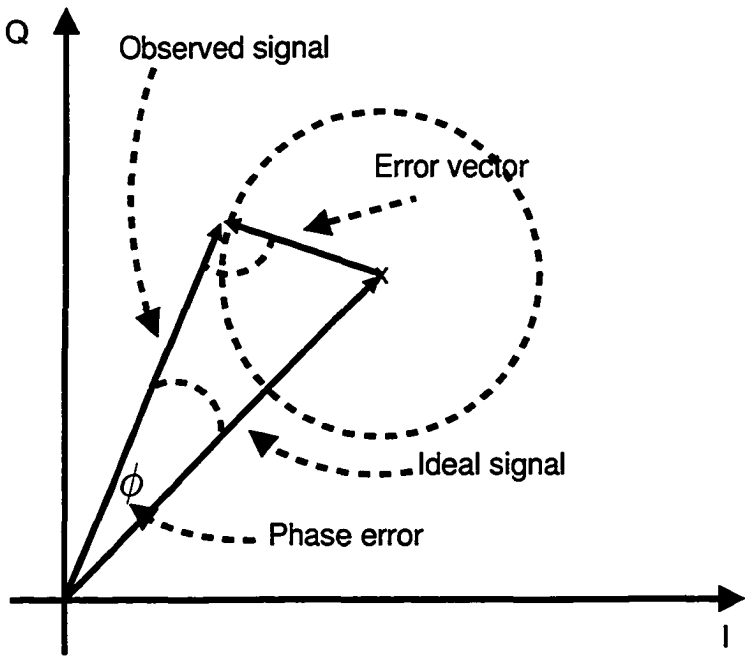


Figure 4.2: Error vector magnitude

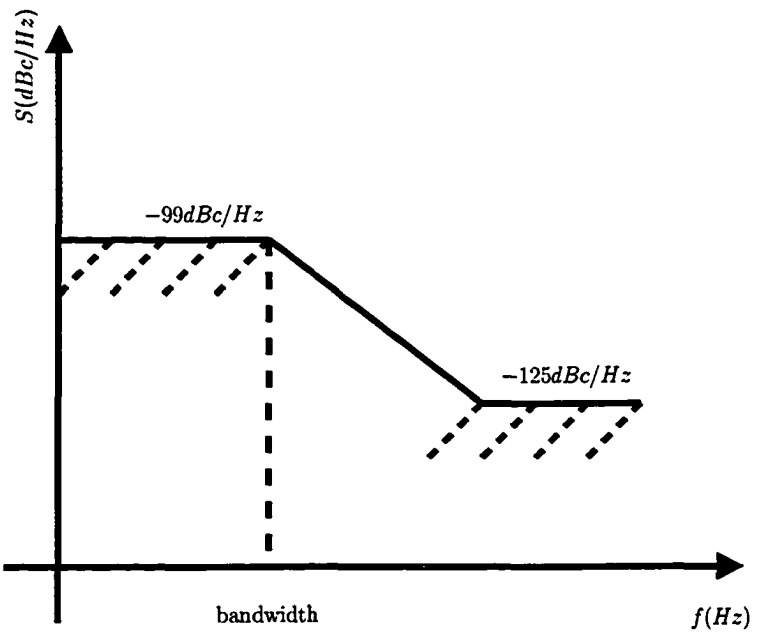


Figure 4.3: Phase noise mask for the PLL.

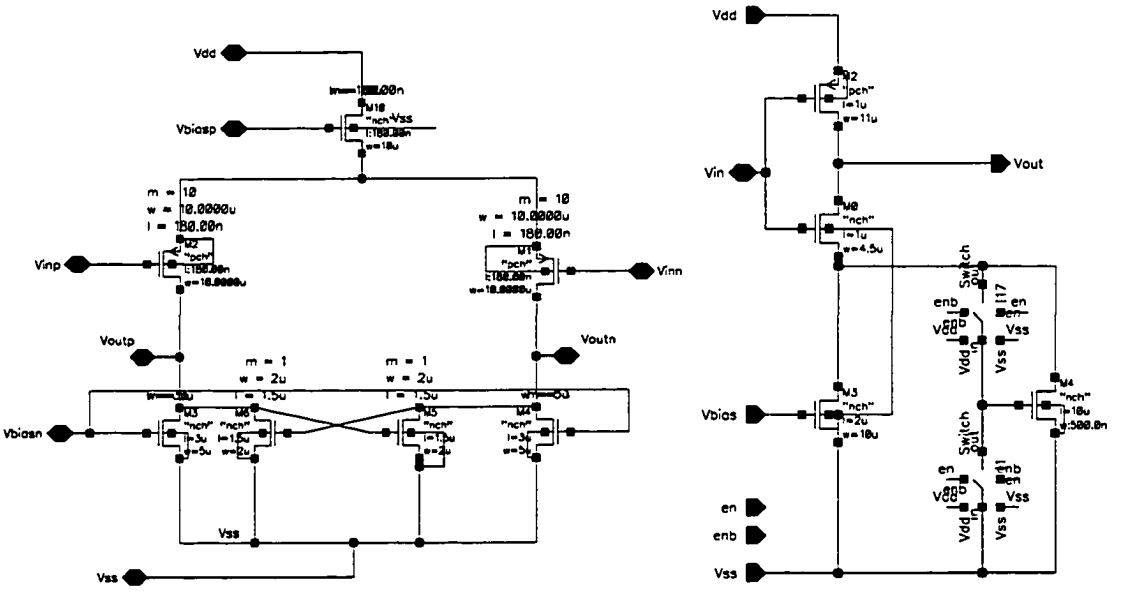
design of the PLL. Therefore, the first section in this chapter will describe the design and performance of the VCO. Following sections will describe the design procedures, and then present and discuss simulation results.

4.1 VCO Design

Choosing the type of VCO was based on the multiphase requirement and the frequency of oscillation. Given that the maximum frequency is below $200MHz$, using a LC tank oscillator for a VCO would require an external pin, due to size of inductor and/or capacitor. Moreover, if the multiphase characteristic is needed, then either the frequency would need to be increased or extra pins would be required for additional LC tanks. Both choices would increase power consumption significantly enough to render this architecture useless compared to other methods of quantization noise reduction. Therefore the design focused on ring oscillators, which inherently have multiphase outputs. Both single-ended and differential ring oscillators are considered.

There are two intrinsic noise sources in the oscillator: flicker noise and thermal noise [45], [46]. As mentioned previously, the flicker noise is a product of CMOS process imperfections and it can be controlled to a limited degree with geometry [32] and proper biasing of the transistors. Thermal noise is caused by the resistance of each component. In general, lower resistance produces less noise. Naturally, a reduced number of components inherently reduces the total intrinsic noise.

Besides intrinsic oscillator noise, there is power supply noise. In low noise design, power supply noise can become the dominant noise source. To avoid power supply noise, a designer usually chooses a differential architecture, because it has good



(a) Differential delay cell for the ring oscillator.

(b) Single-ended delay cell for the ring oscillator.

Figure 4.4: Proposed delay cell designs for differential and single ended oscillator.

power supply rejection ratio (PSRR) to common mode noise sources. Additionally, power supply noise can be minimized with an independent supply for the VCO and by isolating the VCO from the other components of the PLL. Finally, by ensuring that the ISF is very narrow (i.e. having very short rise and fall times), the time during which the circuit is sensitive to noise is reduced. Either one or a combination of the aforementioned methods can be used to decrease power supply noise.

To summarize, an oscillator delay cell with minimal phase noise has: a minimum number of components, very narrow ISF, large PSRR and, by implication of Equation (3.20), large q_{max}; that is output capacitance. Using this information, the following designs for differential [47] (see Figure 4.4(a)) and single ended (see Figure 4.4(b)) delay cells are proposed.

4.1.1 Differential Delay Cell

ISF analysis shows that to lower the phase noise of the oscillator one needs to ensure minimal noise at its harmonics and at DC (i.e. flicker noise). Furthermore, a narrow and symmetrical ISF is needed to minimize the power supply noise and the flicker noise respectively. However, this also implies that the symmetry of the ISF can be ignored if flicker noise is low. Using the same reasoning, having low power supply noise and thermal noise reduces the need for a narrow ISF. Therefore, minimizing these noise sources allows the designer to concentrate on power reduction rather than on the sensitivity of the device to those sources.

Minimizing power supply noise is as simple as having a separate power supply and ground for the VCO. However, precautions during layout have to be taken for this to be realized. Proper isolation of the VCO is still necessary since noise from other circuits can be easily introduced into the VCO through the substrate [48]. To compare the actual difference that the coupling of power noise can create, both differential and single-ended oscillators were optimized for power and phase noise performance.

Another source of noise is the tail current, which is mostly a source of flicker noise [40]. The reason for this lies in the fact that $1/f$ noise in current mirrors is high close to DC, and since PLLs are narrow bandwidth, it becomes the dominant noise [38]. In addition, the ratio of the transistors in the current mirror has to be very small to avoid the multiplication of noise, which in turn means that power consumption must increase. A simple way to remove the major flicker noise contributor is to remove the current mirror from the system. As mentioned in Section 2.3, linearity suffers from this choice. However, this causes less of a problem in the multiphase architecture for reasons explained in Section 4.2.

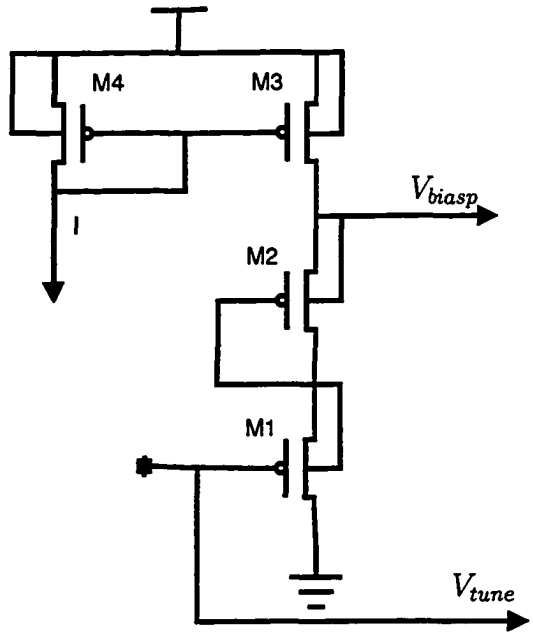


Figure 4.5: Level shifter.

The design from [47] (as seen in Figure 4.4(a)) meets the most important criteria for this system, the removal of current sources. The design consists of four parts.

The first part is the input PMOS pair. These transistors are used as switches, moving current from one side to the other. Next part is the frequency control pair, which is the bottom two NMOS transistors. The gate voltage on these transistors controls frequency of the VCO. Since the current through these transistors is actually what defines the frequency, it is possible to conclude that the frequency will increase by the square law with a linear increase of V_{biasn}. If V_{biasn} can be controlled directly from the loop voltage, which has very low noise, the extra noise introduced through the control mechanism will be negligible compared to designs with a V-I converter [49]. The cross-coupled NMOS pair is the third part. It is in parallel with the frequency control pair. Its purpose is to add extra gain during the outputs' transition,

effectively making ISF narrower. Those transistors are sized to be much smaller than the frequency control pair, hence current through them is a small portion of the total current. The top NMOS transistor represents a source follower and is used to isolate noise on the power supply by keeping its source voltage independent of the drain voltage. This is possible if the gate voltage is independent. In order to drive the gate of the top NMOS transistor, a level shifter is used (shown in Figure 4.5) [47]. The gate voltage, V_{biasp} is defined by the voltage drop through $M1$ and $M2$ (see Figure 4.5). If the current through $M1$ and $M2$ is constant, then V_{biasp} will be independent of the power supply. Therefore, the source voltage of the top NMOS transistor will be defined as:

$$V_{source} = g_m \cdot V_{gs} \cdot Z_L \tag{4.4}$$

where g_m is the transconductance of the transistor, and Z_L is the impedance of the rest of the circuitry as seen from the source of the NMOS transistor. This means that V_{source} will stay independent of the supply as long as all three factors in the equation are independent as well. Unfortunately, this is not completely true for V_{gs}, since changes in the power supply will force a slight change of the current in the current mirror. However, in this design, the frequency is controlled through an input NMOS pair, therefore changes in V_{gs} will have a secondary effect.

The voltage swing of this delay cell is defined by the power supply and the V_{ds} of the NMOS transistor. Ideally, full swing would give lower noise [50], however, minimal sacrifice in swing was chosen to lower the influence of power supply noise.

To test the delay cell, four cells are cascaded together to form the ring oscillator. According to [51] and [49], three stages would give the same phase noise as four with lower power consumption, however, it would mean that only six multiphase

	Frequency (MHz)	Total Current (uA)	Phase Noise (dBc/Hz)
VCO1	110	1500	-106 (200kHz)
VCO2	110	1000	-102 (200kHz)

Table 4.1: Differential VCO results.

outputs could be used. The compromise is to use four stages, which gives eight multiphase outputs, to minimize quantization noise, and allowing wider bandwidth, since quantization noise will be a lesser problem. Two VCOs are optimized, one for power and the other for phase noise. The results are shown in Table 4.1. A significant issue in multiphase design are the buffers required for each phase output, compared to regular VCO design that requires only one buffer. Buffers consume 50% of the total power and are the reason for increased power consumption in the VCO.

A plot of K_{vco} is shown in Figure 4.6. Careful analysis shows that the frequency has a slight square law dependence on the tuning voltage, which is expected. PSRR results show a $0.3\%/V$ rejection, which correlate well with results reporter in [47]. Figure 4.7 shows a graph that approximates the ISF by using the derivative of the VCO output signal. Thanks to the cross coupled pair, the ISF is very narrow, however it is not a symmetrical function. Therefore, it can be expected that DC noise should be up converted to the output frequency. Using a loop filter to directly tune the oscillator, most of the flicker noise is removed from the circuit.

4.1.2 Single-ended Delay Cell

An advantage of the single ended design is the lower number of components and the simplicity of the circuit. Therefore, it is possible to have the same frequency with

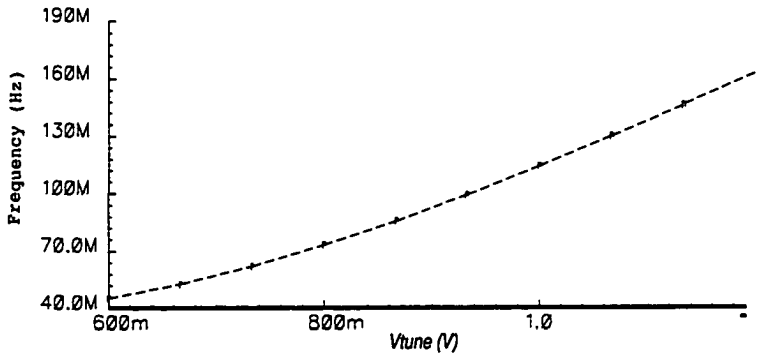


Figure 4.6: K_{vco} of VCO2.

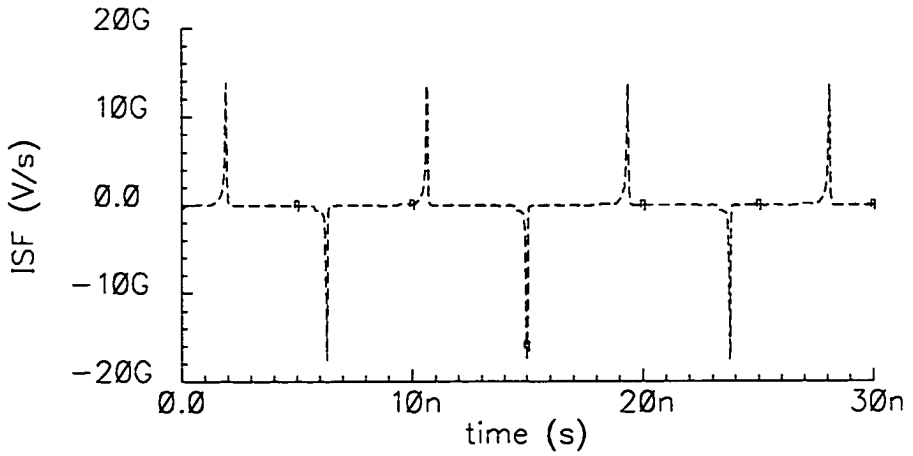


Figure 4.7: ISF of VCO2.

	Frequency (MHz)	Total Power (uA)	Phase Noise (dBc/Hz)
VCO	110	315	-99 (@ $200kHz$)

Table 4.2: Single-ended VCO results.

lower power consumption. Furthermore, being a full swing CMOS design, current only flows during transitions, minimizing static current consumption.

The design of the delay cell, shown in Figure 4.4(b), is a modified current starved delay cell. It is made out of three parts: a CMOS inverter on top, a current control transistor and a diode connected transistor with an on/off switch. The purpose of the inverter is to give a 180^o phase shift between the input and output. The modification in this inverter is that only one side is current starved, in this case the pull-down. This is to avoid the need for a voltage-to-current circuit, which is a source of flicker noise [49]. Therefore, the oscillator is controlled by V_{tune} directly. One of the side effects of this design is a non-symmetrical ISF (Figure 4.8). However, since the flicker noise is already minimized, symmetry is less important. The purpose of the diode connected transistor is to ensure that there is always current flowing in the CMOS gate.

For the same reasons as in the differential oscillator, five delay cells are used for the oscillator instead of three. This oscillator has 10 output phases if both rising and falling edges are included. Noise results are shown in Table 4.2. The power consumed in the single-ended design is almost one third of the total power consumed in the differential design for the same frequency and similar noise performance. However, PSRR is worse at $0.56\%/V$, and is approximately twice that of the differential oscillator.

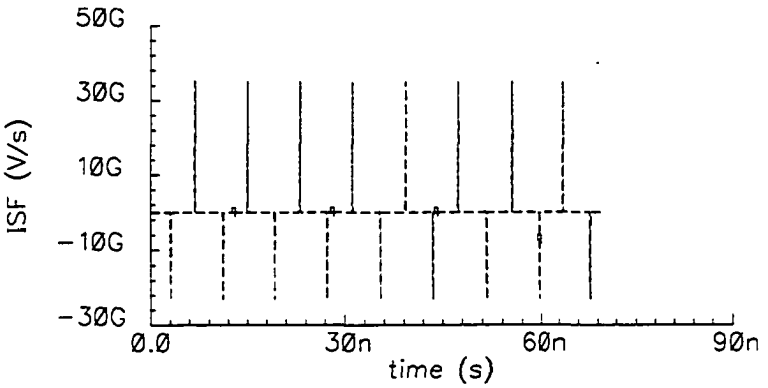
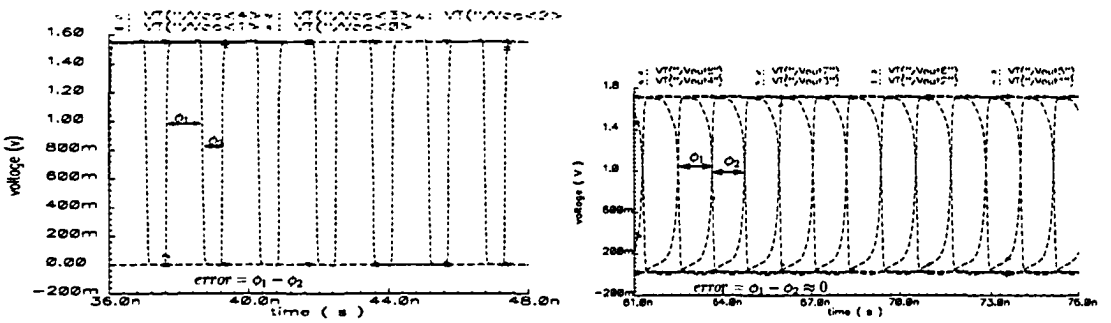


Figure 4.8: ISF of single ended oscillator.



(a) Transient response of the single-ended oscillator.

(b) Transient response of the differential oscillator.

Figure 4.9: Static phase error of the single-ended and differential oscillators.

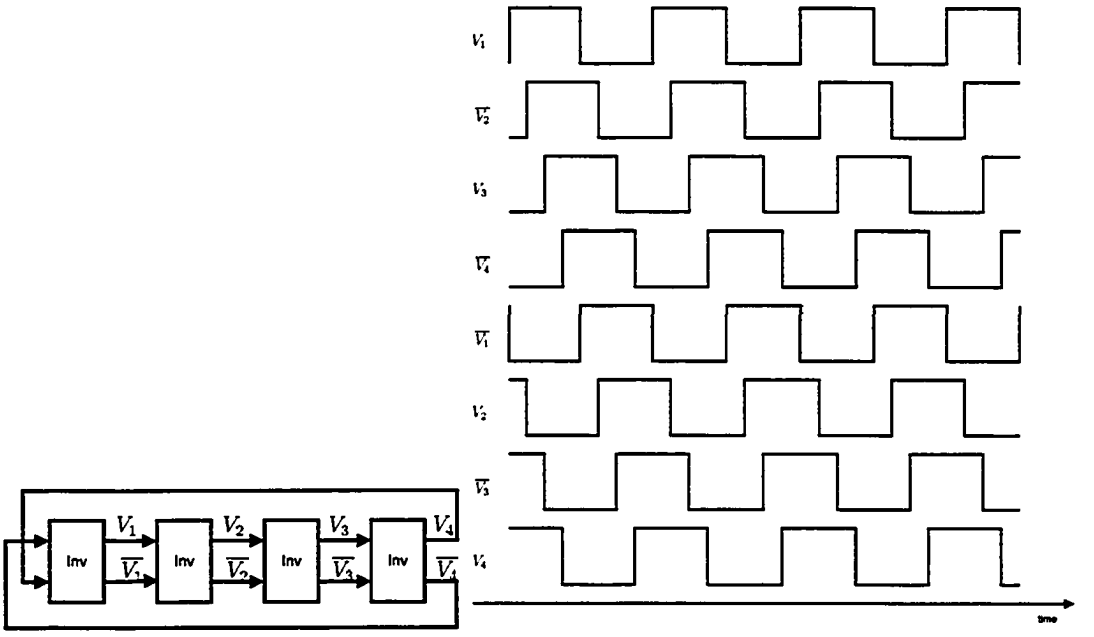
From the results above, it is possible to see that the single-ended oscillator has a major power advantage over the differential oscillator for the same frequency and phase noise. However, where differential oscillator excels is static phase error. In the single-ended oscillator, if the threshold voltage of the delay cell is not exactly $\frac{V_{dd}}{2}$, then the positive or negative edges will not occur exactly every half period (see Figure 4.9(a)). This delay of either positive or negative edges creates a static phase error among the multiphase outputs. For differential oscillator, it is possible to select only one type of edge since every output has its inverse. As Figure 4.9(b) shows, there are no significant phase differences in the differential oscillator.

Knowing the phase noise of the oscillator, one can choose the proper architecture for the rest of the PLL system.

4.2 System Level Design

An inherent problem with $\Sigma\Delta$ PLLs is high quantization noise at higher frequency offsets. This will add upper limit to optimal bandwidth if the quantization noise is dominant for out-of-band frequencies. Excluding a slower locking time, problems with a narrow bandwidth are the VCO's phase noise and reduced capability for direct modulation [52]. There are methods that can speed up locking in these cases (e.g. 5 state PFD). However, in the case that one requires 2-point modulation or direct modulation scheme, the minimum bandwidth is defined by the data rate. In modern RF circuits the data rate can reach a few Mb/s (megabits per second), which means that the bandwidth has to a increase to a few MHz to allow direct modulation. Moreover, to minimize influence of the VCO phase noise on the system, the bandwidth needs to get wider as well.

As mentioned in Section 2.9, there are different approaches to decrease the quantization noise and to increase bandwidth, which all lead to increased complexity and power consumption. This thesis will show that using the multiphase characteristic of ring oscillators to decrease the quantization noise is a viable option, with a minimal increase in complexity and power consumption. Example outputs of a differential ring oscillator with four stages is shown in Figure 4.10(b). A multiphase output allows one VCO to be split to $2N$ equal parts, where N is the number of stages. If one selects the next phase output instead of the next divisor ratio, the quantization noise



(a) Differential ring oscillator (b) Differential ring oscillator output

Figure 4.10: Ring oscillator and multiphase output

can be decreased by $\frac{1}{8}$. This is equivalent to decreasing the quantization step in $\Sigma\Delta$ quantizer by the same amount. One other way to decrease quantization steps in $\Sigma\Delta$ quantizer is to have VCO running at higher speed (8 times in this example), which would increase power consumption as well.

A block diagram of the proposed architecture is shown in Figure 4.11. In this case, for simplicity, there is a divider for each phase output. Eventually this could be replaced with a single divider and a more advanced phase selector. Figure 4.12 shows the waveforms of the proposed architecture. For a specific case, the system is set to divide by $8+\frac{1}{16}$. Previously there was only one phase used, and the difference between the two edges was at least T_{vco} (from example $9clks - 8clks = 1clk$). However, in this case, the minimum difference is $\frac{T_{vco}}{8}$ as shown in Figure 4.12. The divisor ratio is able to divide by increments of $\frac{1}{8}$. Therefore, the divider will keep an integer divisor

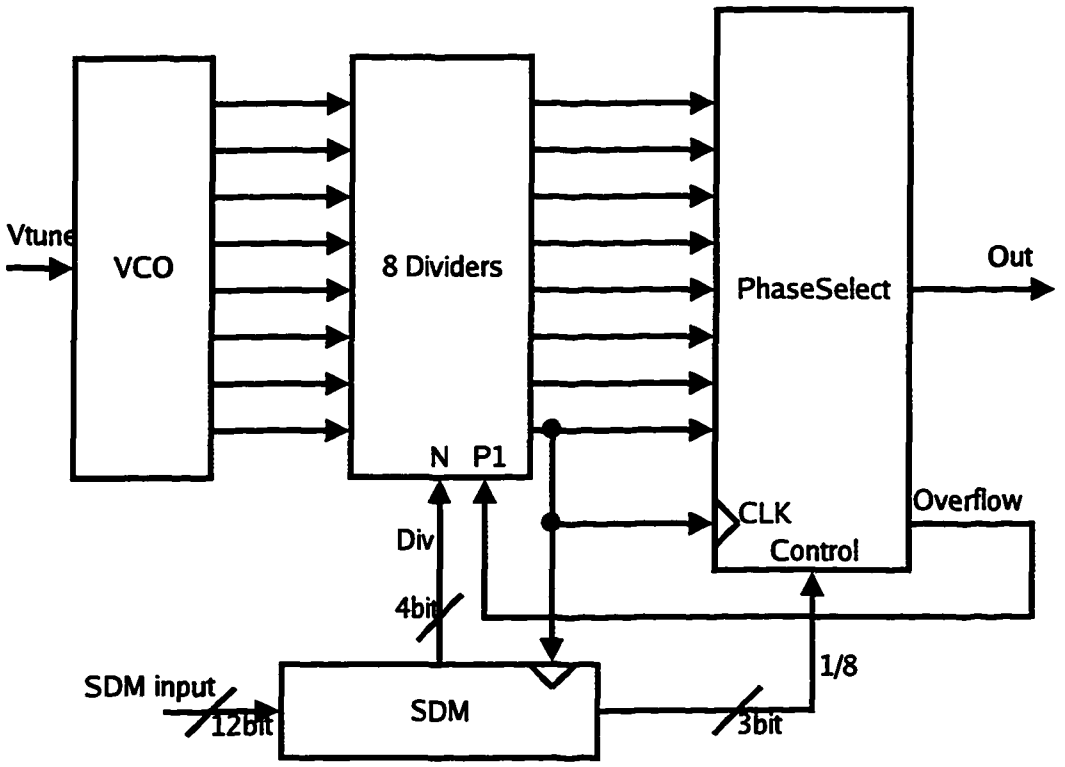


Figure 4.11: Block diagram of the multiphase architecture.

to 8 but alternate phases between phase zero and phase one (i.e. it would divide alternatively by 8 and $8+\frac{1}{8}$).

In the case that the $\Sigma\Delta$ chooses $8+\frac{1}{8}$ for the divisor, the $\Sigma\Delta$ will set the divisor for the divider to 8 while the phase selector will add one to the accumulator ($\frac{1}{8}$ part), as shown in Figure 4.11. This in turn selects a phase that is one after the previous phase (e.g. if the old phase was 4, the new phase will be 5). In Figure 4.12, we can see that the $\Sigma\Delta$ sets the divisor to 8, then $8+\frac{1}{8}$, and then back to 8 with selected phases $0, 1, 1$ respectively. The fractional part gets accumulated (Figure 4.13) and passed to a MUX. In the case of an overflow, which happens every time when a new phase is out of range, the integer part (i.e. divider ratio) is increased to cover the range. For example, if the previous phase was 5 and the new divisor is $8+\frac{6}{8}$, the new phase will

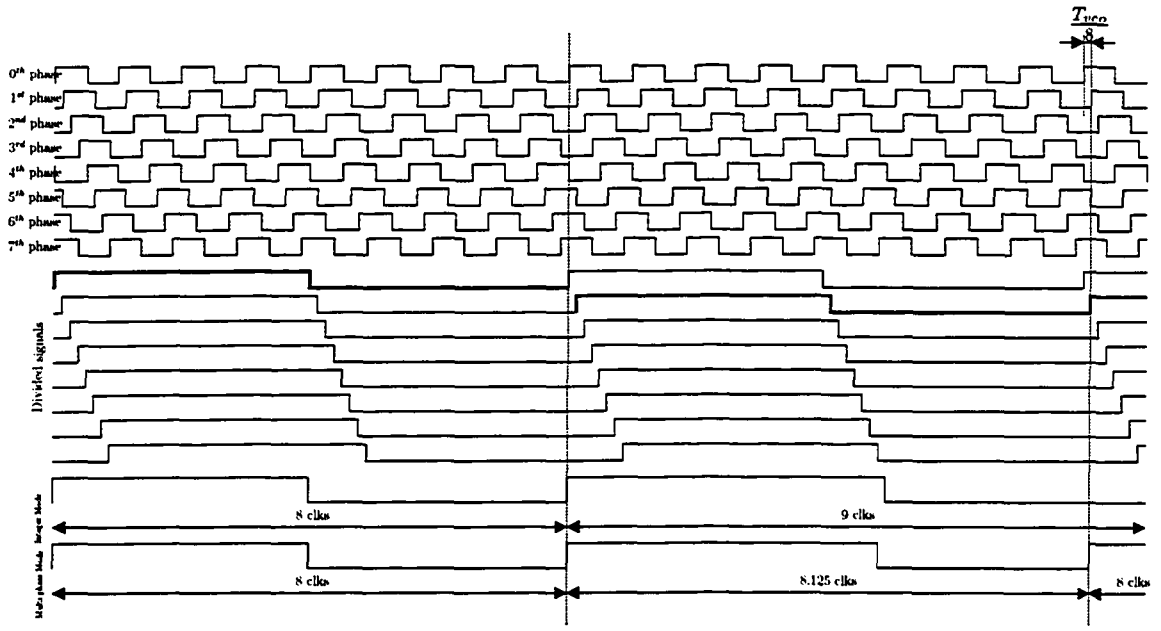


Figure 4.12: Waveform outputs for the multiphase architecture.

be set to $mod(\frac{5+6}{8}) = 3$ and the divisor for the divider will be set to $int(\frac{5}{8}+8+\frac{6}{8}) = 9$ instead of 8 due to overflow. Therefore, the system will wait for 9 divider edges and then select phase 3, which makes the difference between two output edges $8+\frac{6}{8}$.

To find out the optimum $\Sigma\Delta$ design/order and loop filter order, a Simulink model was designed. The same model is used to learn how the multiphase VCO influences the PLL performance, specifically, the influence of static phase error among the multiphase outputs of the ring oscillator. The model is shown in Figure 4.14.

Since the duration of the pulse will be cut by a multiphase factor N, it is expected that the nonlinearity of the charge pump will decrease by the same factor. The reasoning behind this is the fact that the error charge will be inserted for a time that is proportionally shorter to N. Hence, if the error in the charge pump is ϵ and the $\Sigma\Delta$ quantization noise folding due to mismatch is as described in Equation (3.28),

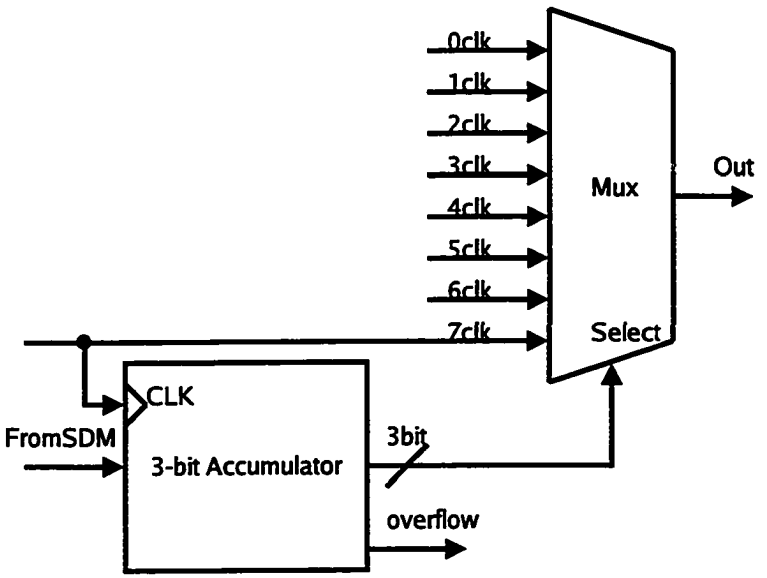


Figure 4.13: Phase-selector block diagram

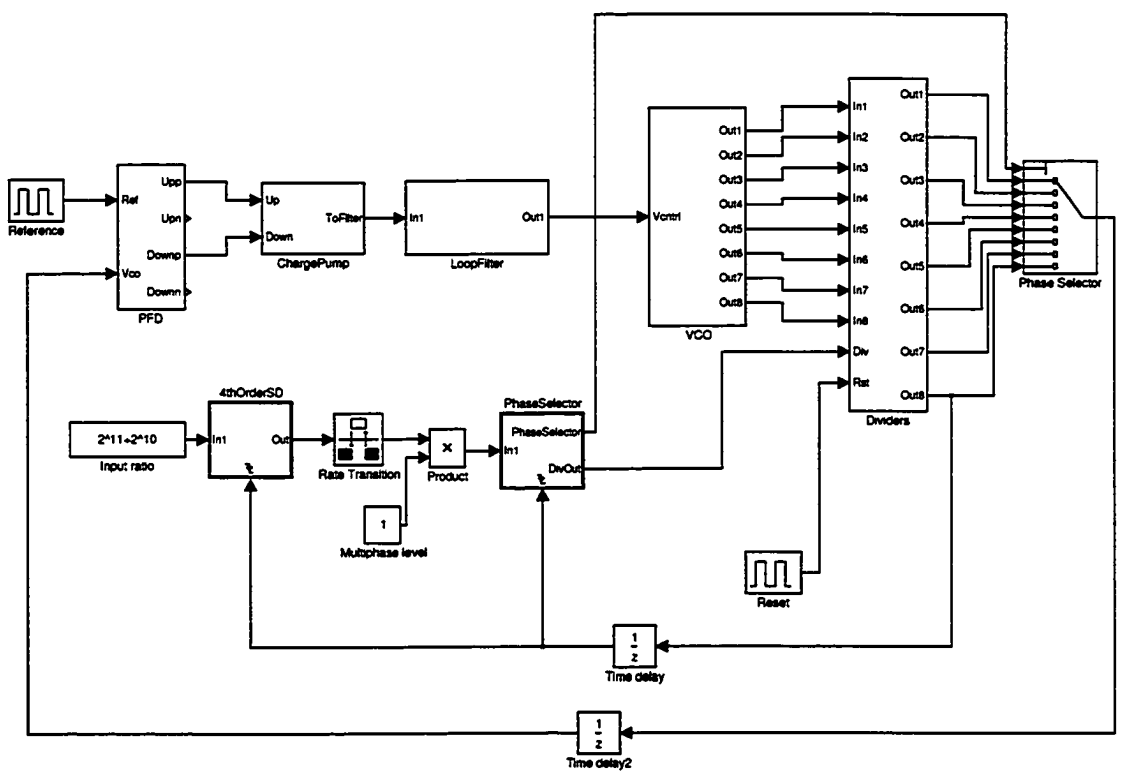


Figure 4.14: Simulink PLL model.

then the equivalent error for the multiphase architecture is:

$$\epsilon_{mp} = \frac{\epsilon}{N} \tag{4.5}$$

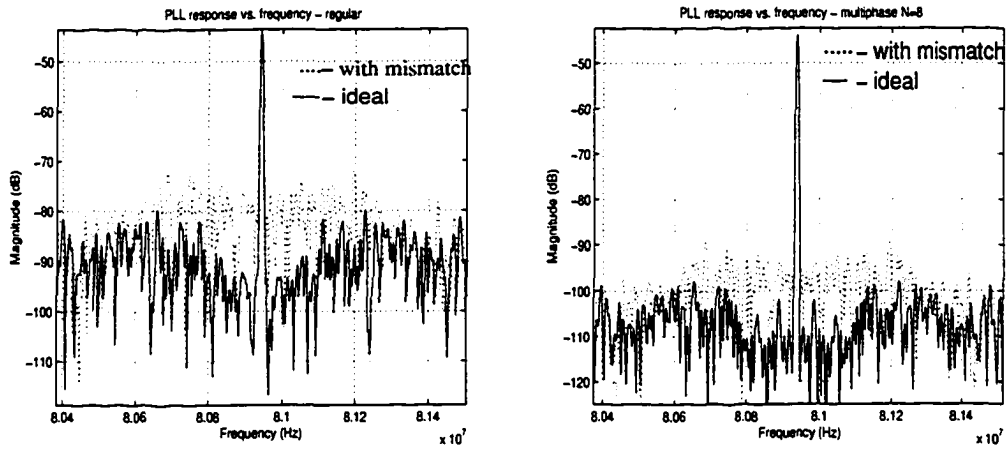
and the folded noise floor is:

$$S_{folding_{mp}} = \left(\frac{\epsilon/N}{2}\right)^2 \frac{(2\pi)^2}{F_{ref}} \left(1 - \frac{2}{\pi}\right) \frac{\sigma^2}{T_{ref}^2} \tag{4.6}$$

To test this, a 25% charge pump mismatch is modeled ($150\mu A$ for up current and $112.5\mu A$ for down current). The folding noise expected by Equation (3.28) is $S_{folding} = -75dBc/Hz$ for a 4^{th} order $\Sigma\Delta$. However, since a multiphase VCO with $N = 8$ is used, new calculations for folded noise using Equation (4.6) give $S_{folding_{mp}} = -93dBc/Hz$, a difference of $18dB$. Figure 4.15(a) demonstrates a PLL with and without the mismatch of 25%. Figure 4.15(b) shows a multiphase PLL with the same mismatch. It is clearly evident from the plots that the mismatch in the charge pump is attenuated by the expected factor of $20 \log N$.

In a ring oscillator, the signal propagates through multiple stages before it is fed back to the first stage. Ideally, each stage has the same delay. However, in reality, process variations and unequal signal paths from stage to stage introduce static differences in each stage. This difference can be translated directly to an equivalent current mismatch error in the charge pump. Assuming that the static delay error for stage n is ϵ_n, in degrees, then the average can be calculated as follows:

$$\epsilon_A = \frac{1}{N} \sum_{n=1}^{N} |\epsilon_n| \tag{4.7}$$



(a) Regular PLL response with charge pump mismatch.

(b) Multiphase PLL response with charge pump mismatch.

Figure 4.15: Multiphase noise folding vs. a regular PLL.

The average percentage error charge due to the phase offset in the VCO is:

$$\epsilon_{percent} = \frac{\epsilon_A}{360 \cdot N_{frac}} \tag{4.8}$$

where N_{frac} is fractional division ratio. Using similar procedure as was done in [27] for charge pump mismatch, it is possible to express the phase noise folding due to static phase error, referred to the input of the PFD, as follows:

$$S_{folding_{mm}} = \left(\frac{\epsilon_A}{360 \cdot N_{frac}}\right)^2 \frac{(2\pi)^2}{F_{ref}} \times \sigma^2_{|\epsilon_n|} \tag{4.9}$$

where the term $\left(\frac{\epsilon_A}{360 \cdot N_{frac}}\right)^2$ represent the equivalent current error in the charge pump and $\sigma^2_{|\epsilon_n|}$ is variance of the static phase error due to process mismatch during fabrication. $\sigma^2_{|\epsilon_n|}$ can be calculated from the Monte-Carlo simulation if standard deviation or variance for different parameters were given. To demonstrate influence of the static

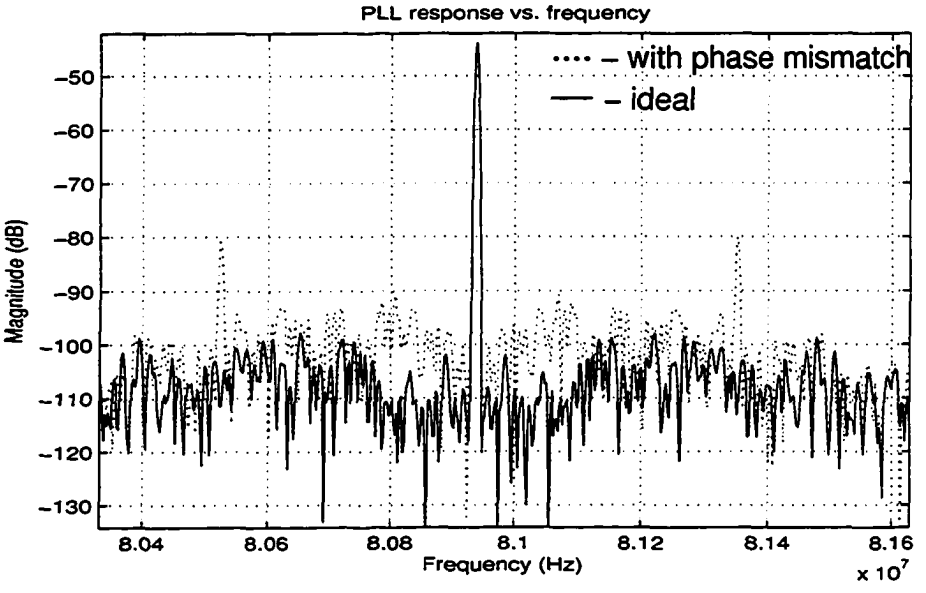


Figure 4.16: Noise folding due to static phase offset in the VCO.
Dashed line shows 10dB increase of the quantization noise in-band.

phase error in the VCO, the Simulink model phase offset for every second phase is set to 10^o, therefore the average phase error is $\epsilon_A = 5^o$. A cyclical phase error is used to emphasize and analyze possible spurs from the given architecture. Figure 4.16 shows, the Simulink model gives an output with in-band noise increased $10dB$.

Another problem with static phase mismatch is the creation of spurs that appear at the PLL output. An example of such spurs can be seen in Figure 4.16 (the dotted line). Model testing showed that the frequency of the spurs is a function of the reference frequency and the control word in the $\Sigma\Delta$. The magnitude of the spurs increases with the static phase mismatch. In general, the spurs can be controlled with proper design. Also, using randomization in the $\Sigma\Delta$, the multiphase outputs of the VCO are randomized too, hence the spurs can be greatly attenuated [27].

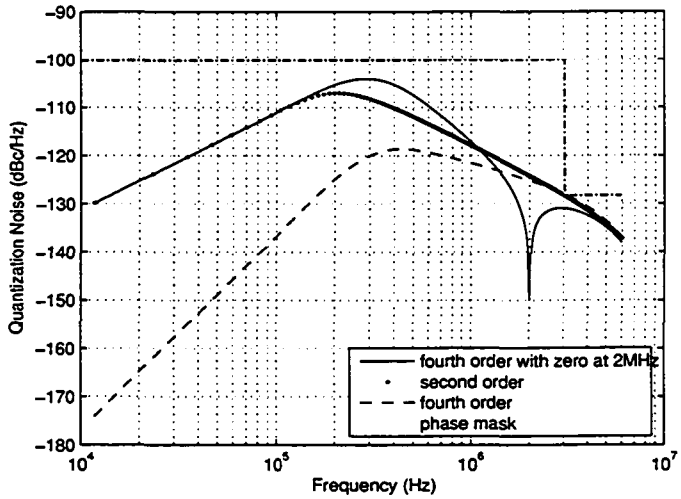


Figure 4.17: Quantization noise of different $\Sigma\Delta$ modulators

4.2.1 $\Sigma\Delta$ order

The phase noise mask from Figure 4.3 shows that the in-band noise should not be higher than $-99dBc/Hz$, and that it should drop sharply to $-125dBc/Hz$. In order to maximize the bandwidth, an optimum order and architecture for the $\Sigma\Delta$ needs to be found. Starting with Equation (3.27), using a $\Sigma\Delta$ order of 2 and 4 and an equivalent loop filter order, it is possible to find the maximum loop bandwidth that will meet the aforementioned specifications.

Assuming that multiphase VCO is not used, based on the Equation (2.31), the maximum bandwidths for the 2^{nd} and 4^{th} order $\Sigma\Delta$, with corresponding 2^{nd} and 4^{th} order loop filters, are 100kHz and 200kHz respectively (see Figure 4.17).

As Figure 4.17 shows, if the 4^{th} order $\Sigma\Delta$ is used with one zero pair at $2MHz$, the quantization noise around $2MHz$ will be minimized. In that case, the out-band noise specifications can be met without using complex filters. However, this comes with a price of increased in-band quantization noise. Given that the in-band noise

will be dominated by charge pump noise (see Section 4.4) and noise folding from higher frequencies, the increase in the in-band noise, due to the use of the 4^{th} order $\Sigma\Delta$ with a zero at $2MHz$, will be insignificant. Therefore, the 4^{th} order $\Sigma\Delta$ with a zero at $2MHz$ is used in this PLL design.

The number of bits in the $\Sigma\Delta$ is defined by the resolution required in the PLL. Given that the accuracy of the LO at $2.4GHz$ must be $100kHz$, it is possible to calculate the resolution of the $\Sigma\Delta$. Since the 21^{st} harmonic is used, the accuracy at the PLL frequency must be $100kHz/21 \approx 4762Hz$. For a reference frequency of $12MHz$, a 12-bit $\Sigma\Delta$ is required to achieve the proper resolution. Moreover, two extra bits are needed to insure that the $\Sigma\Delta$ does not saturate, one bit for the randomizer and one bit for the sign. Hence, the 4^{th} order, 16-bit $\Sigma\Delta$ was designed.

4.3 Loop Filter

Since the 4^{th} order $\Sigma\Delta$ was designed, a 4^{th} order loop filter was necessary to compensate for the increased $\Sigma\Delta$ quantization noise. Figure 4.18 shows the schematic of the filter. The filter is based on the second order lead-lag filter, with an extra two poles to reduce quantization noise. The transfer function for this filter is calculated similarly to the loop filter shown in Figure 2.16 and Equation (2.18). The difference is that two extra poles are added. These poles are far enough to not influence the stability of the system.

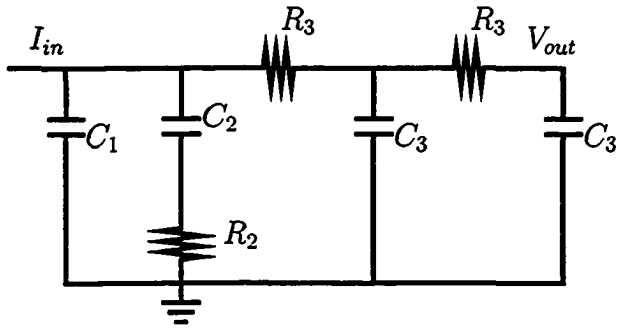


Figure 4.18: Fourth order loop filter.

4.4 PFD and Charge Pump Design

PLLs most often include a digital PFD and charge pump [53]. In this PLL a simple tri-state PFD is found to be sufficient. The design of the PFD is presented in Section 2.4. Its delay time is set to $800ps$ to insure that both U_p and D_{own} current signals are properly turned on before the reset is applied.

A schematic of the charge pump is shown in Figure 4.19. It uses cascode current mirrors to match the up and down current signals. There are five main current mirrors with switches, which behave as a simple DAC. Their purpose is to select the output current range from $10\mu A$ to $150\mu A$ in steps of $10\mu A$. These current mirrors provide an additional variable to tune if the circuit becomes unstable. The output stage uses a simple current mirror with a switch for on/off. This switch is also used as degeneration for the output current mirror. In order to match the loads for the U_p and D_{own} signals, an additional transistor with source and drain shorted is added on the D_{own} signal. It adds a capacitive load to slow the D_{own}, which, according to simulation results, is slightly faster than U_p signal. In addition, a $3pF$ capacitor is added to the output current mirror bias to minimize switching noise on the loop filter.

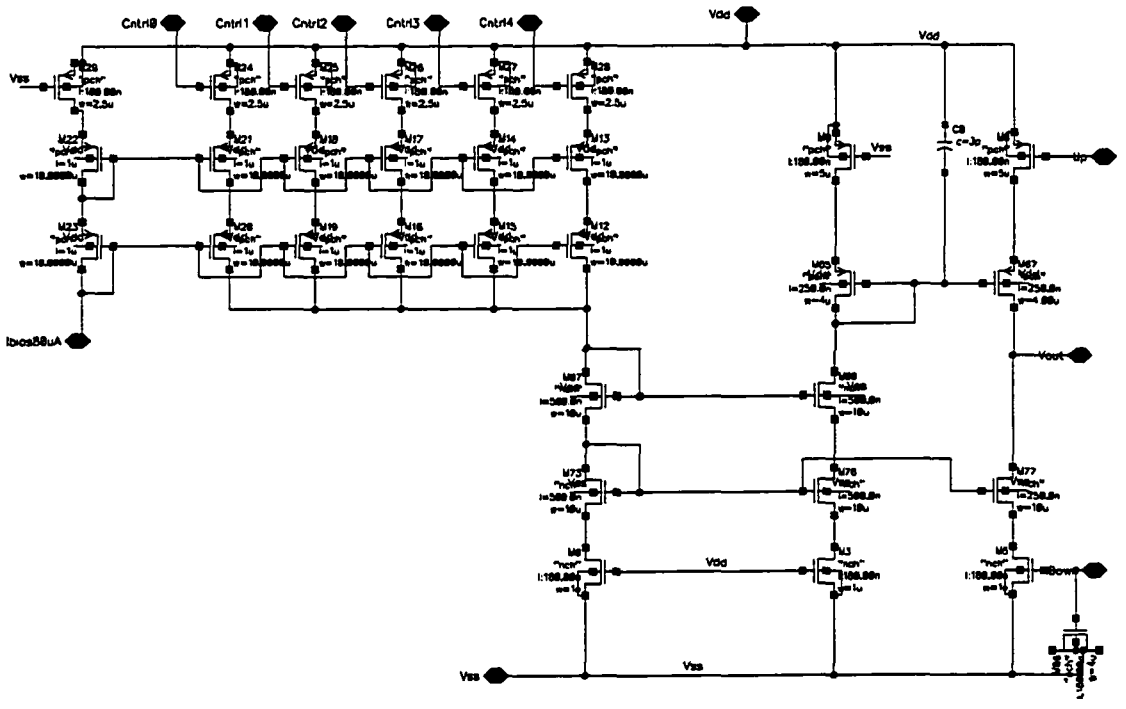


Figure 4.19: Charge pump schematic

The current matching is presented in Figure 4.20. It shows that the current mismatch for the range of $550mV$ to $1050mV$ is $\pm 5\%$ when $V_{dd} = 1.5V$. When compared to K_{vco} in Figure 4.6, the range, which ensures $\pm 5\%$ mismatch, is $40MHz - 120MHz$ that corresponds to the working range of the PLL.

To calculate the phase noise at the output of the PLL due to the charge pump, first, the phase noise of the charge pump is simulated, then converted to an output phase noise using Equation (3.17) and plotted in the Figure 4.21. As it is apparent from Figure 4.21, the charge pump phase noise is lower than $-99dBc/Hz$ over the entire range, which is used as a guideline.

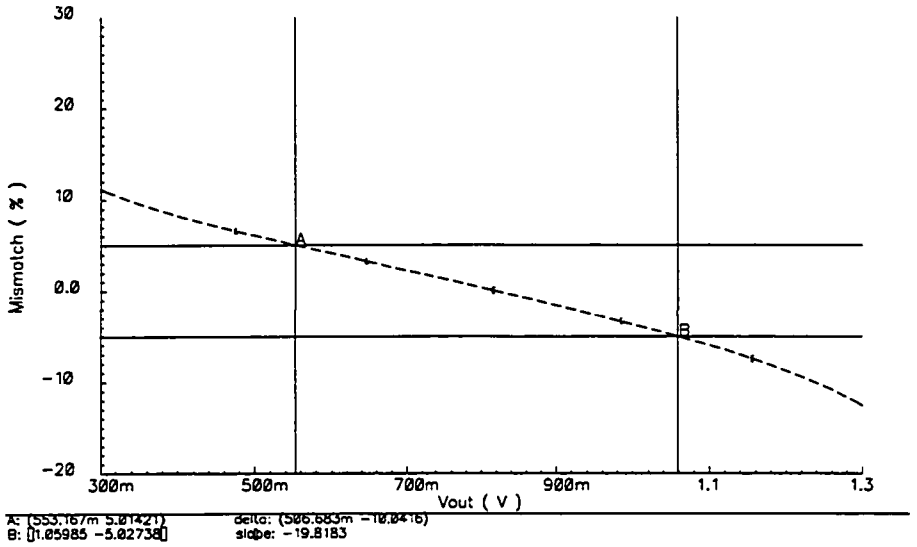


Figure 4.20: Current mismatch in the charge pump, $V_{dd} = 1.5V$.

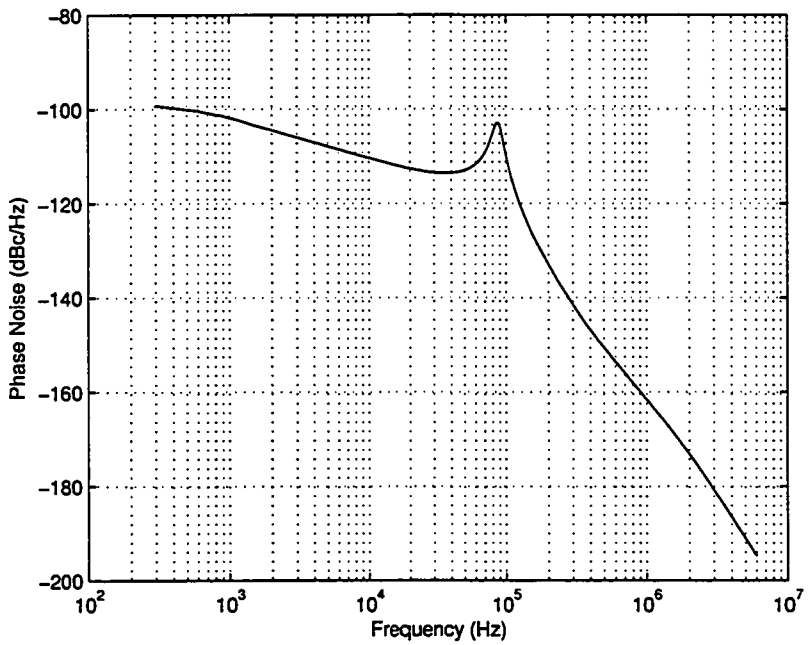


Figure 4.21: Phase noise at the output of the PLL due to charge pump.

4.5 Digital Block

The digital block consists of dividers, $\Sigma\Delta$, phase selector, randomizer and reset block. The proposed architecture is presented in Figure 4.11. The following subsections describe each of the blocks and the choices made while working on their respective design.

4.5.1 Dividers

The output frequencies of the PLL are based on the 21^{st} subharmonic of the $915MHz$ frequency band and the 21^{st} subharmonic of the $2.4GHz$ frequency band [44]. There are 27 channels covered in the 802.15.4 standard. These are:

- $F_c = 863.6MHz$
- $F_c = 906 + 2(k-1)$ in megahertz, where $k = 1, 2, .., 10$, and
- $F_c = 2405 + 5(k-11)$ in megahertz, where $k = 11, 12, ..., 26$

Hence, the minimum frequency of the PLL is $863.6MHz/21 \approx 41.12MHz$, while the maximum frequency is $2480MHz/21 \approx 118.095MHz$. For a divider to cover this range, the divisor range must be from $41.12MHz/F_{ref} \approx 3.42$ to $118.095MHz/F_{ref} \approx 9.84125$, which must be rounded down for the lowest divisor and up for the highest divisor, 3 to 10. Since the fourth order $\Sigma\Delta$ will change the output ± 5 levels from the average value, total divisor range would have to be -2 to 15 to avoid saturation of the $\Sigma\Delta$ modulator. However, minimum practical divisor is 2, therefore, a counter divider is used because it can cover a range from 2 to 16 as explained in Section 2.5 and

as presented in Figure 2.14. This means that dividing by less than 7 is not possible without multiphase mode. However, in the multiphase mode, it is possible to get up to the 8 extra fractional quantization levels between two integer levels, allowing for the use of the divide by 3 without saturating $\Sigma\Delta$ (i.e. $3 - \frac{5}{8} > 2$).

In order to avoid further complications in the design, a divider is included for every phase output. Although this inevitably increases the power consumption, it shortens the design time, since there is less concern about delays and races in the circuitry. The design that consumes much less power is shown in Figure 4.22. The only difference from the regular design is the addition of extra flip-flops that change state at the reference frequency, thereby not adding significantly to power consumption. The design from Figure 4.22 is not implemented in this tape-out due to the complex problem of the divider delay variation and the time required to solve it. This delay could cause some phase other than the phase marked zero to arrive high first at the flip-flop, hence changing the phase order at the output (see Figure 4.24 for the expected effect).

The circuits, which are not shown in Figure 4.11, are the randomizer and reset circuits. The randomizer is needed to insure that the input signal at the $\Sigma\Delta$ is not static. Having a static signal at the input makes the output very spurious. However, once random LSB noise is added to the signal, the quantization error will have the properties of an "input-independent additive white-noise approximation" [32]. It is only when these properties are true that the $\Sigma\Delta$ will have a noise shaped output.

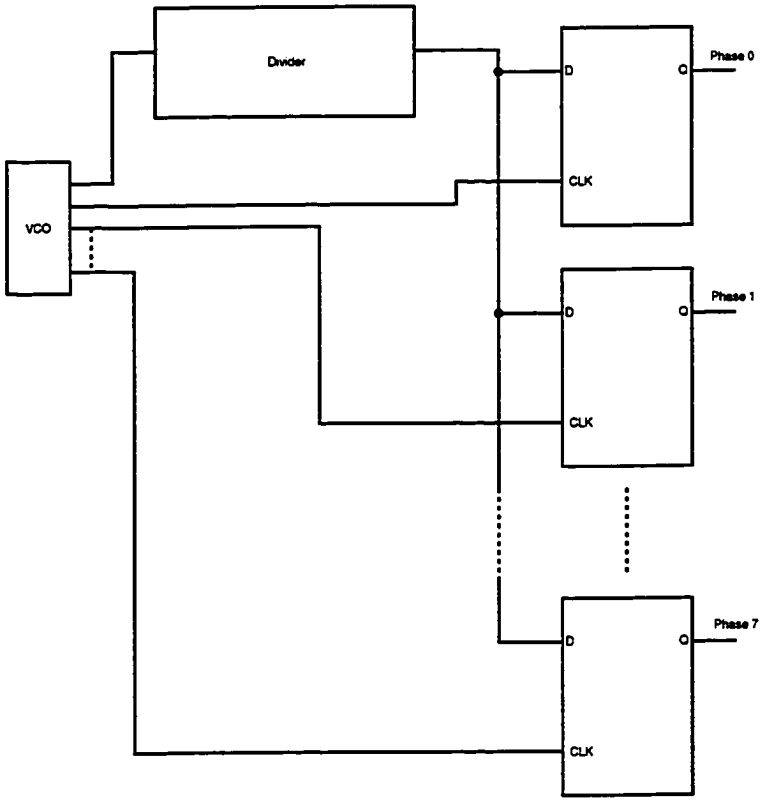


Figure 4.22: An improved multiphase divider.

4.5.2 Linear Feedback Shift Register

The randomizer is implemented with a linear feedback shift register (LFSR). The LFSR is an example of a pseudo-random bit generator. It has a stream of N serially connected registers. Some of the register outputs are XOR'd and fed back to the input (Figure 4.23). If this feedback system creates a *primitive polynomial*, which is a polynomial that cannot be factored, then the output of the LFSR will be periodic, with a period of $2^N - 1$ [54]. During this period the LFSR gives a random output stream of $2^{(N-1)}$ ones and $2^{(N-1)} - 1$ zeros. In order to get the LFSR to work properly, it is important to insure that the LFSR does not enter the $0 - state$, which is a state where each register has a zero output. Once in this state, it will stay in it until some external signal changes at least one register output. The actual periodicity of the stream can be found as:

$$T_{LSFR} = \frac{1}{(2^N - 1) \cdot T_{ref}} \tag{4.10}$$

Based on the equation above, the spur frequency caused by the LFSR can be calculated. Since the reference frequency is fixed, the period of the spur will depend on N. There are two reasons why N needs to be large. First, it is important to insure a very low spur frequency. If the frequency is close to DC, then modulation of the output will be minimal. Second, a proper randomization of the input signal is needed. For example, if one needs a 14-bit resolution in the $\Sigma\Delta$, then the minimum LFSR should be at least 15 bits to insure that the minimum fraction is randomized as well. To demonstrate this, a simple fractional divider can be considered. If $1/2^{14}$ is the required fraction, it can be assumed that the output will be a repetitive stream of $2^{14} - 1$ lows and only one high. Bit streams as this can create spurs throughout the bandwidth of the PLL. However, if a longer LFSR bit stream is added to the input

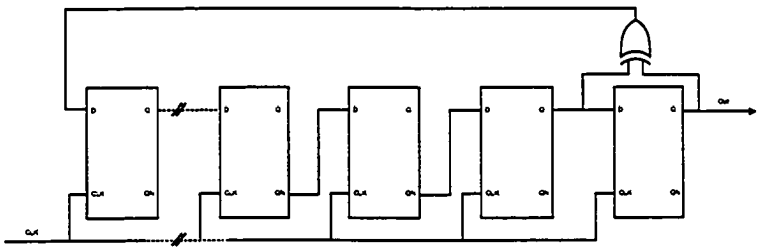


Figure 4.23: Linear feedback shift register (LSFR) diagram.

signal, then the $\Sigma\Delta$ output stream will be randomized enough, so that repetitiveness will not be visible. Since a 16-bit $\Sigma\Delta$ is used, the LFSR is chosen to be 21-bits.

4.5.3 Reset Circuit

In order for the system to work properly, it is important to determine which phase is number zero in the phase selector. To solve this problem, the inputs to the phase selector are hardwired in the proper order, where phase zero is the phase that comes first and so on. However, given that the divider starts dividing at some arbitrary time, the first edge seen at the divider input does not necessarily have to be phase zero. If this is the case, then the order of the phases coming into the divider can be changed as well. Figure 4.24 demonstrates a case where the fourth phase has been seen first by the divider. To insure a proper startup, a reset circuit is introduced, which is synchronous to the last phase of the VCO. Therefore, once the circuit is out of the reset state, the first phase will be seen by the divider before any other. The circuit ensures that output is as presented in Figure 4.25. The implementation of the reset is shown in Figure 4.26.

The whole digital block, including dividers, the phase selector, $\Sigma\Delta$ and randomizer, is implemented in Verilog code and synthesized using standard library cells.

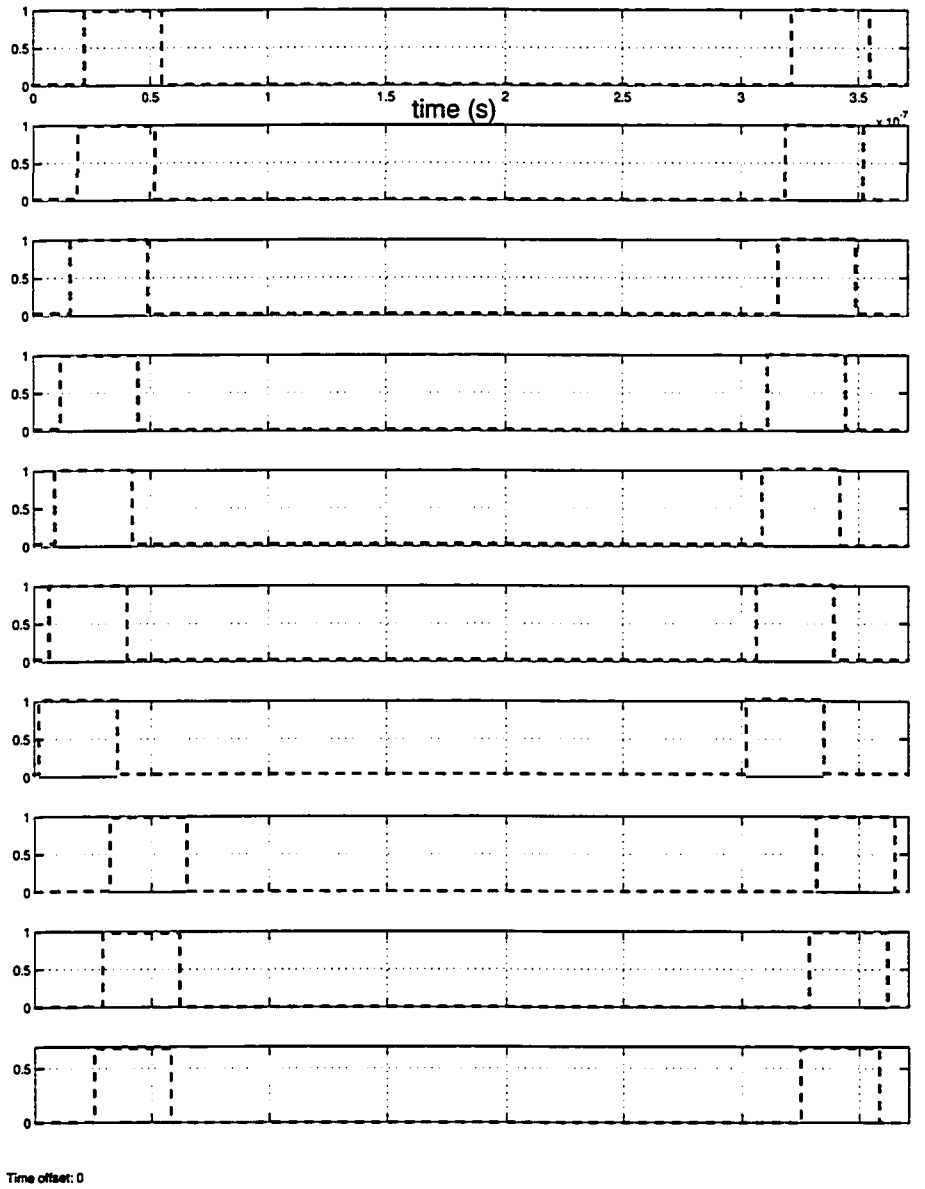


Figure 4.24: Improper order of the phases.

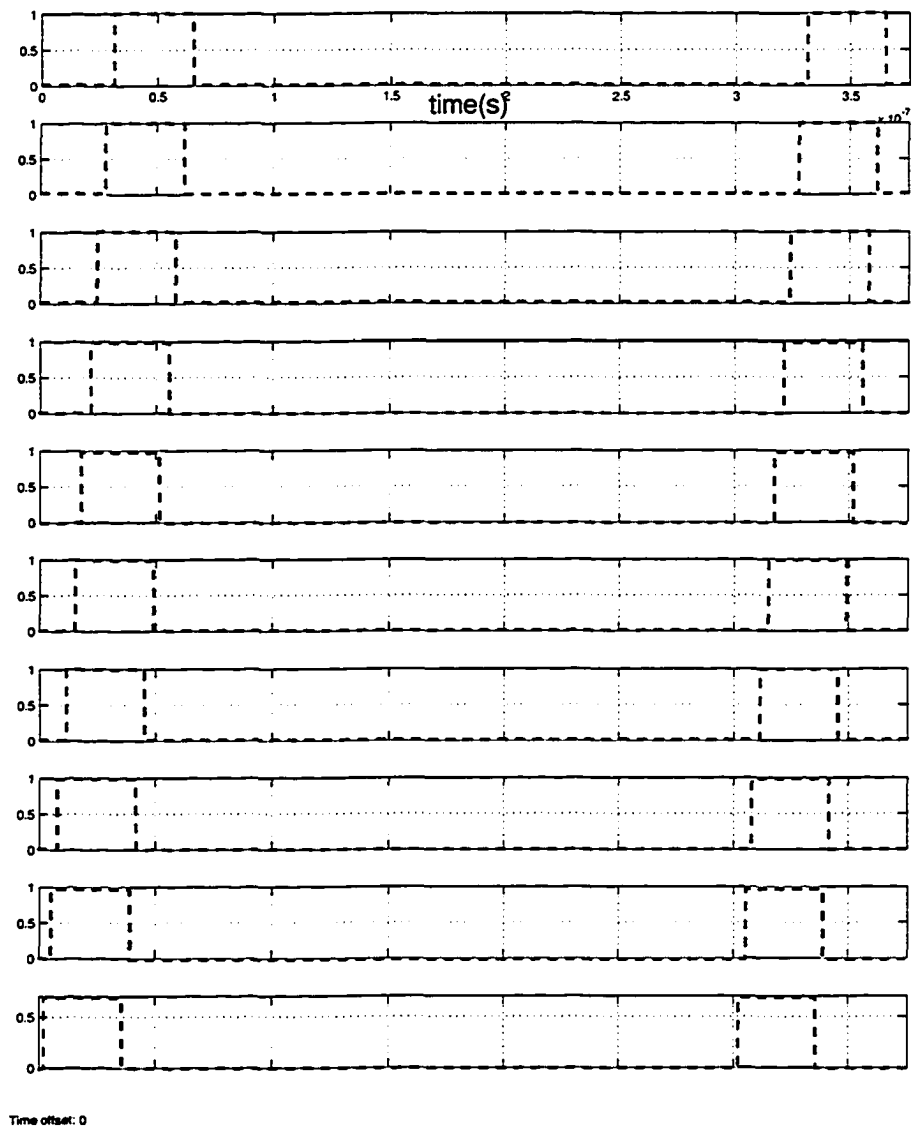


Figure 4.25: Proper order of the phases.

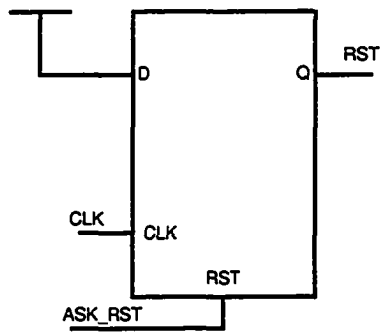


Figure 4.26: Reset circuit.

The code is included in Appendix A. A ramification of doing the design with standard cells is the increased power consumption (simulated total of $500\mu A$, or $850\mu W$).

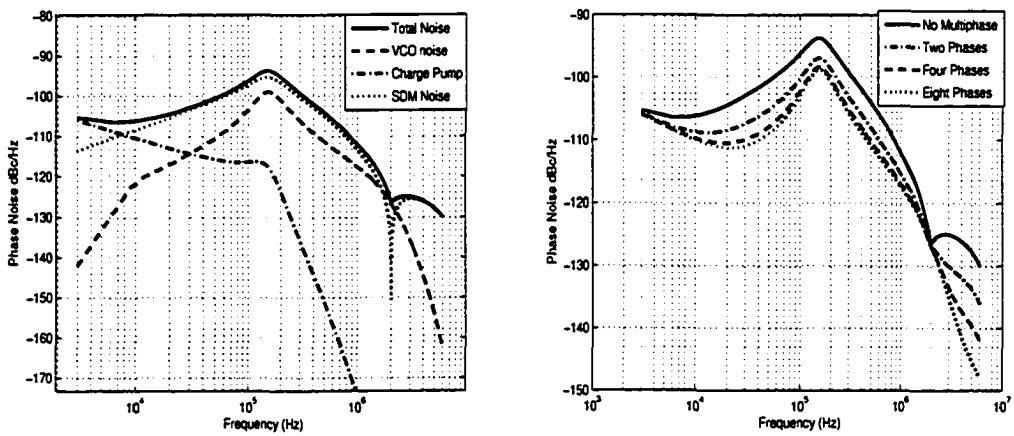
4.6 Predicted Results for Full PLL

By using the simulated results of the phase noise from the charge pump, VCO and calculated quantization noise for the sigma delta, it is possible, using Equation (3.30), to predict the total noise at the output of the PLL. Figure 4.27(a) shows the total noise from different sources. As can be seen, the $\Sigma\Delta$ noise is dominant if only one phase from the VCO is used. However, Figure 4.27(b) shows that using more than one phase significantly reduces the influence of the quantization noise. To compare an actual decrease in the phase noise from one multiphase level to the other, the phase noise was converted into rms jitter and presented in the Table 4.3. As the multiphase level is increased, the VCO noise becomes more dominant.

Where the multiphase architecture really excels is at the wide bandwidth loops. This is shown in Figure 4.28. Without the multiphase VCO, the output noise is dominated by a quantization noise but, if more than one phase was used from the VCO, the RMS jitter would decrease significantly and VCO phase noise starts dominating (see

Number of phases	RMS jitter in degrees
1	0.77
2	0.60
4	0.50
8	0.44

Table 4.3: RMS jitter for different number of phases (200kHz loop filter).



(a) Phase noise at the output of the PLL without multiphase quantization noise reduction.

(b) Phase noise at the output of the PLL with multiphase quantization noise reduction.

Figure 4.27: Noise with and without multiphase VCO.

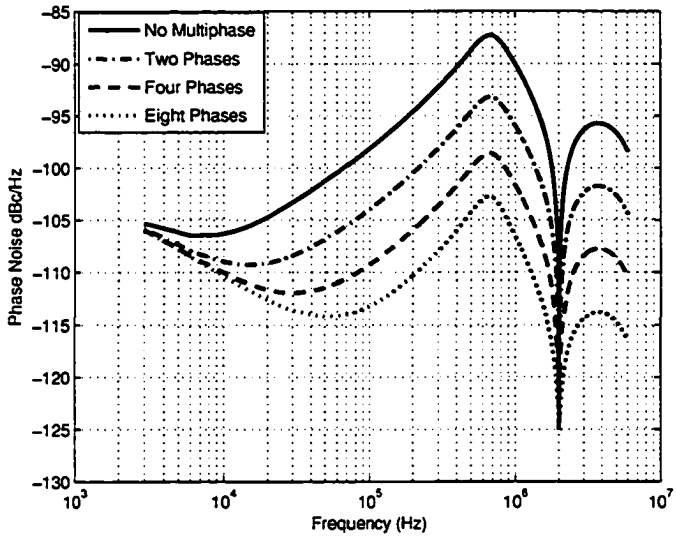


Figure 4.28: Wide bandwidth multiphase output.

Number of phases	RMS jitter in degrees
1	3.81
2	1.92
4	1.00
8	0.56

Table 4.4: RMS jitter for different number of phases (700kHz loop filter)

Table 4.4 for calculations). Therefore, for wide bandwidth systems, the multiphase architecture shows a potential for decreasing the total phase noise of the system. However, it is interesting to note that once the multiphase order is increased, even a second order $\Sigma\Delta$ will have similar RMS jitter performance, while third order seems to be optimal (see Table 4.5). Unfortunately, the third order $\Sigma\Delta$ modulator has high quantization noise at higher frequencies as shown in Figure 4.29. Therefore, a fourth order $\Sigma\Delta$ with a zero at $2MHz$ was chosen. There was an option to use the second order $\Sigma\Delta$ modulator, however, a fourth order modulator with zero at $2MHz$ allows for higher bandwidth if it is decided to deactivate the multiphase for power savings.

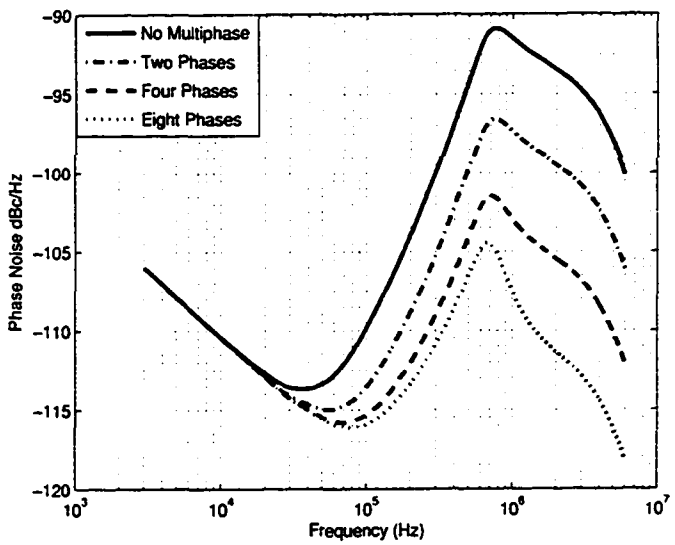


Figure 4.29: Output phase noise with 3^{rd} order $\Sigma\Delta$.

Number of phases	RMS jitter in ^{o}, 2^{nd} order	RMS jitter in ^{o}, 3^{rd} order
1	4.14	3.56
2	2.09	1.80
4	1.07	0.94
8	0.59	0.54

Table 4.5: RMS jitter for different number of phases and orders of $\Sigma\Delta$ (700kHz loop filter)

The following chapter will present results of the test chip and compare them with simulated results. It will also explain any differences in the results and discuss possible causes.

Chapter 5

Measurements and Discussion

A PLL is built according to the design highlighted in Chapter 4. Actual fabricated chip can be seen in Figure 5.1. Following parts are marked:

I - Output buffers

II - Digital circuit

III - Differential oscillators

IV - Single-ended oscillator

V - PFD and charge pump

VI - Serial to parallel interface

To control the PLL, a serial-to-parallel (SPI) interface was added on chip. This way, by using only four external inputs, it is possible to control up to a 128 digital inputs on the chip. The full test setup is shown in Figure 5.2. Measurement results are presented in the following sections.

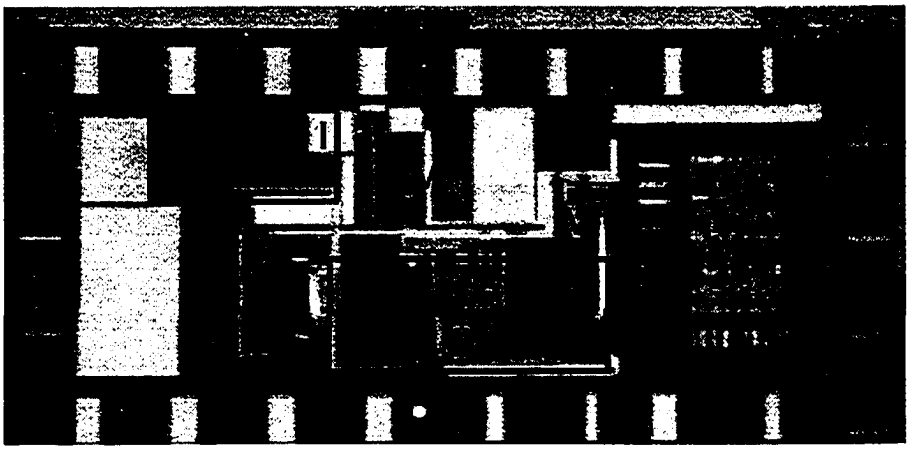


Figure 5.1: Fabricated chip.

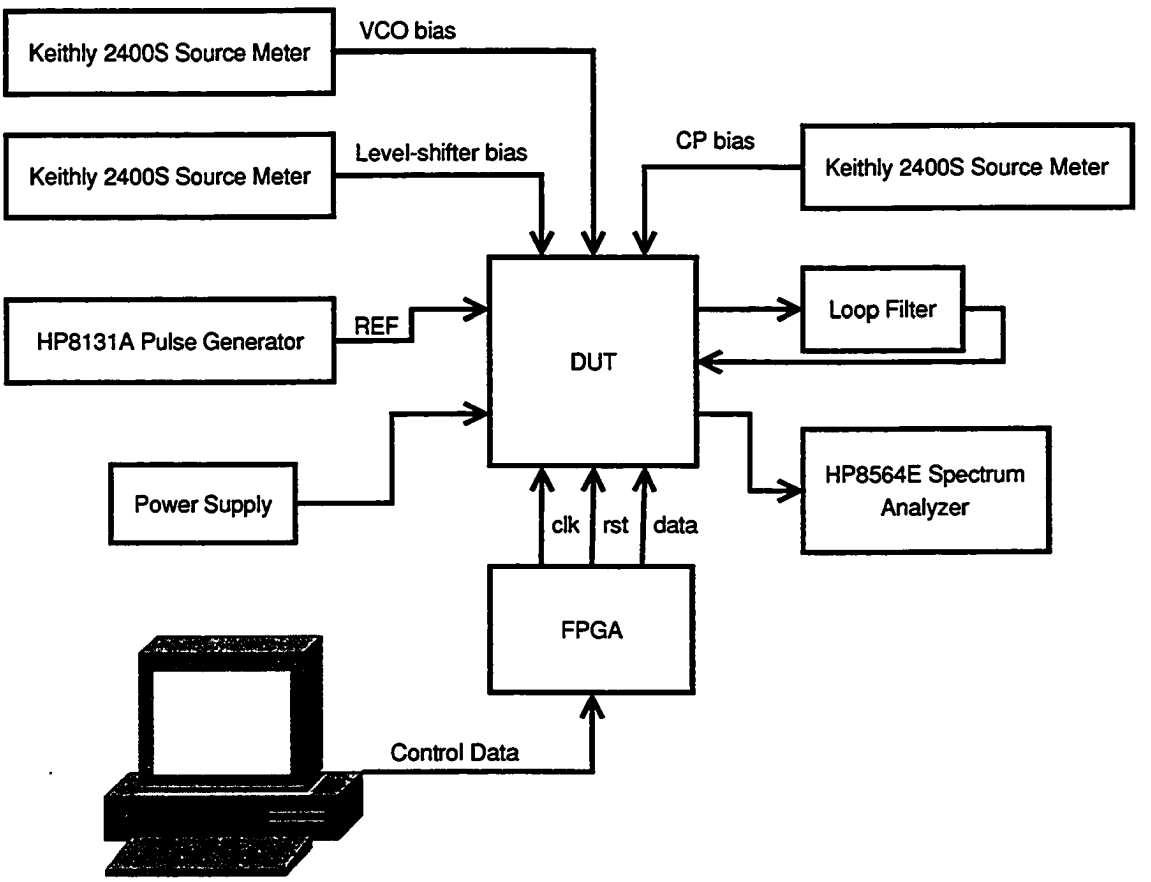


Figure 5.2: Test setup.

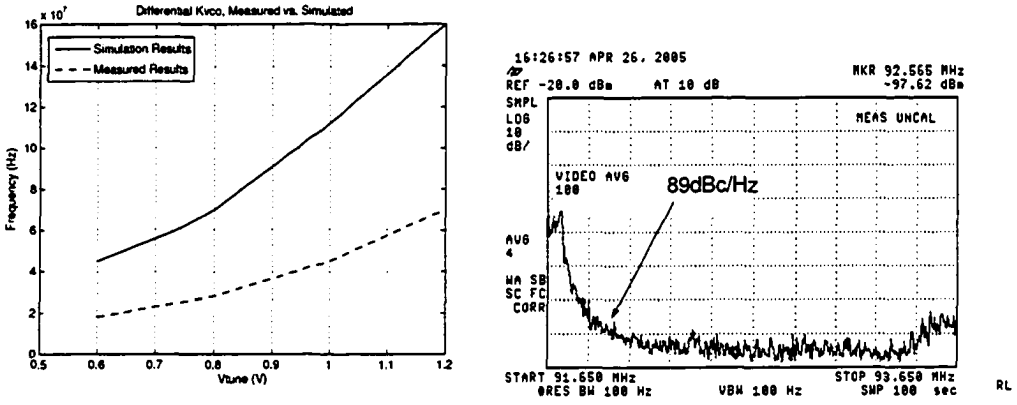
5.1 VCO Measurements

The performance of the free running VCOs were the first measurement taken. All measurements were done by controlling the V_{tune} of the VCOs with a DC voltage supply while measuring the power consumption, output frequency, and phase noise of each VCO.

5.1.1 Differential VCO Measurements

The phase noise measurement is shown in Figure 5.3(b). It was measured indirectly since the particular spectrum analyzer did not have a phase noise module. The peak of the tone, which was $-3dB$, was used as a reference point. Next, the start frequency of the spectrum analyzer was set to the tone frequency, while the end frequency was set to a $2MHz$ offset. Video bandwidth was set to $100Hz$. The noise at $91MHz$ at the offset of 200kHz was measured to be $-106dBc/Hz$. By converting this to single side band, a VCO phase noise of $-103dBc/Hz$ can be obtained.

The measured K_{vco} of VCO2 is presented in Figure 5.3(a). For comparison simulated results are put in the figure as well. If two figures are compared, it is evident that extracted simulation has about two times the frequency compared to the measured performance when the same V_{tune} is applied. The measured power consumption of VCO2 is $1.8mA$ for $110MHz$ compared to a simulated $1mA$ for the same frequency. For a given bias, the output frequency is almost half of the simulated frequency. There are two possibilities that could cause this. Either the given bias transistor is not pulling the predicted current, or the load capacitance is two times larger than simulated. However, the measured voltage on the current mirrors is the



(a) Measured vs. simulated frequency of the fast differential oscillator.

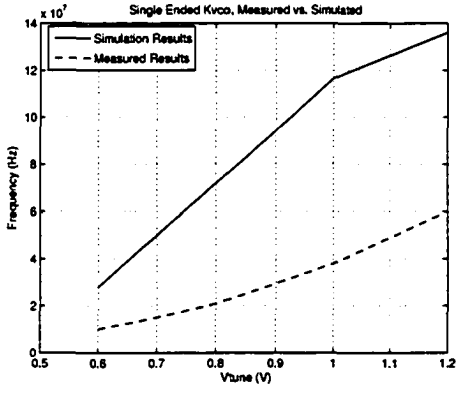
(b) Differential oscillator phase noise.

Figure 5.3: Differential VCO measurements.

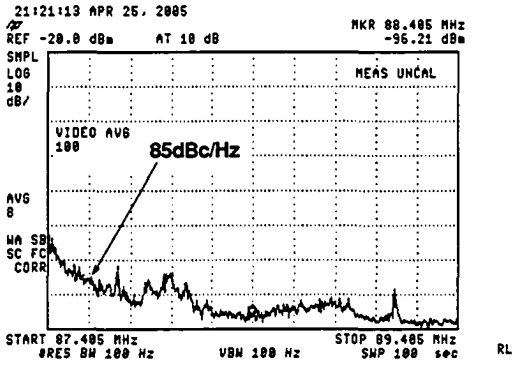
same as simulated, showing that model can accurately predict the current-voltage dependence. Therefore, there is a higher probability that the extraction tool did not report all the capacitance in the layout, which could also explain the frequency shift and the improved phase noise (measured $-103dBc/Hz$ over $-102dBc/Hz$ simulated).

5.1.2 Single-ended VCO Measurements

Figure 5.4(b) shows the phase noise results of the single ended oscillator. The noise was measured in a similar way as in case of the differential oscillator. The phase noise measured at a $200kHz$ offset from $87MHz$ is $-100dBc/Hz$ with a power consumption of approximately $350uA$. The expected number for the phase noise is the same, but with a power consumption of $250uA$ for the given frequency. Similar to the differential VCO, the difference in the power consumption is most likely due to an



(a) Frequency vs. V_{tune} of the single ended oscillator.



(b) Single-ended oscillator phase noise.

Figure 5.4: Single-ended VCO measurements.

improper extraction of the capacitance in the layout. Since routings of the single-ended VCO are shorter and simpler than the differential VCO, the increase in the power consumption is not as dramatic.

5.2 PLL Measurement

Following the VCO measurements, the full loop was turned on with the setup shown in Figure 5.2. The goal was to measure a difference in the phase noise among different multiphase levels. First, the PLL was run with only one phase coming from the VCO. In the integer mode, everything was as expected, as shown in Figure 5.5. At the loop bandwidth corner frequency ($200kHz$), phase noise is identical to differential VCO phase noise, while in-band phase noise drops of with lower offset. Spurs in Figure 5.5 are found to be from unknown external source.

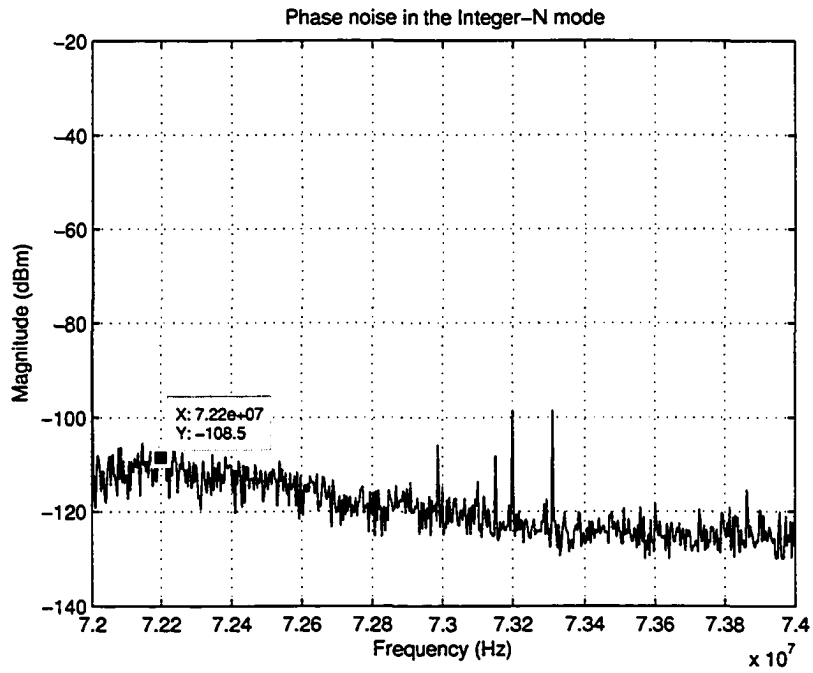
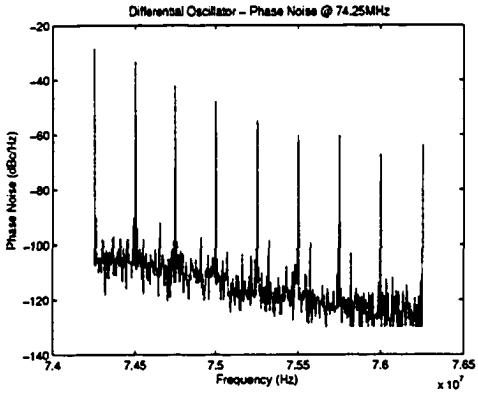
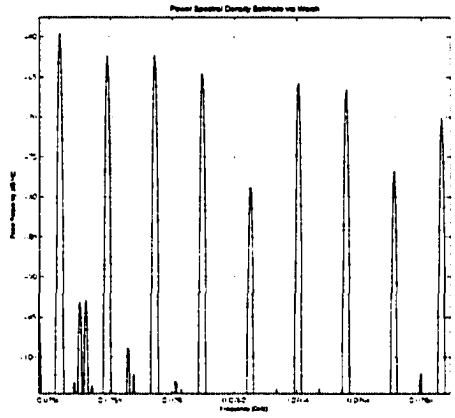


Figure 5.5: Integer-N mode phase noise (72MHz, using differential oscillator).

However, if the $\Sigma\Delta$ was turned on while randomizer was still off, the output that is shown in Figure 5.6(a) appeared. The source of the spur was found to be an unequal number of buffers in the clock tree. The synthesizer tool inserted buffers in such a way that it effectively made a second order $\Sigma\Delta$ out of the fourth order. The coefficients of the new second order $\Sigma\Delta$ are based on the fourth order $\Sigma\Delta$ and are therefore not optimal. Figure 5.7(a) shows the noise shaping filter that was coded in Verilog and Figure 5.7(b) presents the noise shaping filter that had been designed due to the clock issue. The error was not detected since the PLL was simulated in a mixed design simulator, where digital cells were simulated as verilog code and analog as the extracted layout. In order to confirm that this is an error, the matlab model was simulated with the second order $\Sigma\Delta$ with coefficients [3 -4 1], vs. optimal [1 -2 1]. It should be noted that, due to the improper clock tree, it is hard to extract proper



(a) PLL with differential oscillator at 74.25MHz, no multiphase.



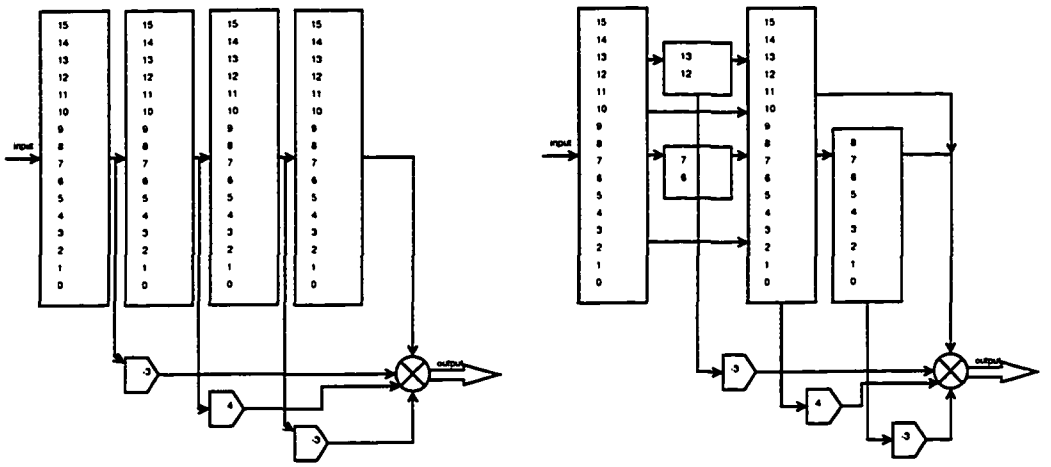
(b) PLL with differential oscillator matlab model.

Figure 5.6: PLL with differential oscillator.

coefficients, and [3 -4 1] were chosen as the best estimate. Simulation of the model shows a somewhat similar spur pattern, as seen in Figure 5.6(b), to the measured result, where a difference comes from the coefficient estimates. Once the randomizer is turned on in the $\Sigma\Delta$, a quantization noise as seen in Figure 5.8 should be expected. It becomes evident that the phase noise will be dominated by the quantization noise of the improper $\Sigma\Delta$. Although this renders the chip useless in an RF system, it should be good enough to show a quantization noise reduction if the number of phases is increased.

Other issue is a reversed order of the inputs to the dividers. This is another reason for the high spurs in the multiphase mode.

Fortunately, it was still possible to show a difference between multiphase levels. Figure 5.9, compares the use of 4 phases and the use of 8 phases. For this



(a) Proper fourth order noise shaping filter.

(b) Noise shaping filter due to clock tree problem.

Figure 5.7: Noise shaping filter problem.

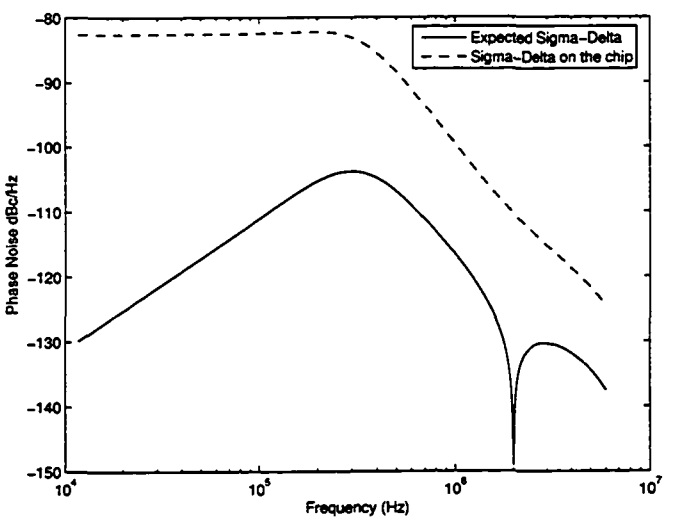


Figure 5.8: Problem with $\Sigma\Delta$.

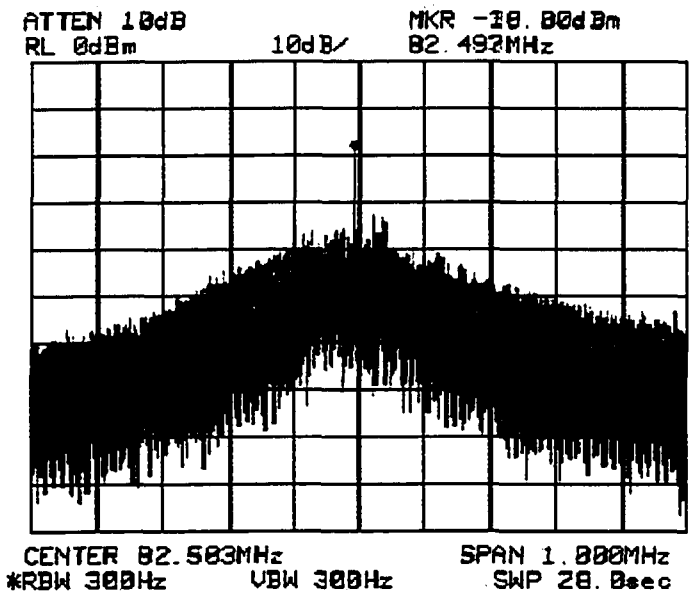


Figure 5.9: Comparision between mutliphase level 4 and level 8.

measurement, the charge pump current was decreased to $20\mu A$ to minimize quantization noise by reducing PLL bandwidth to $50kHz$. With the $200kHz$ bandwidth noise was high enough to cover the signal itself. As stated before, an expected difference is $20log(8) - 20log(4) = 6dB$ over the entire bandwidth. While this is mostly true, the in-band noise reduction is only $2dB$ lower.

5.3 Overall Performance

The results of the synthesizer are summarized in Table 5.1. The synthesizer current consumption is dominated by the VCO and the buffers in the differential oscillator, while it is dominated by dividers in the single-ended design. Although the differential oscillator gives a better PSRR performance, it was not found to be significant enough to justify the four times power consumption increase in the VCO.

Parameter	Value (Single-ended)	Value (differential)
Frequency Range	10MHz-100MHz	20MHz-120MHz
Phase Noise @200KHz (Integer-N)	-100dBc/Hz	-104dBc/Hz
Supply Current @110MHz	1.6mA	3mA
Supply Voltage	1.5-1.8V	1.5-1.8V
Tuning step	1.5kHz	1.5kHz

Table 5.1: PLL overall performance.

Parameter	PLL1 [55]	PLL2 [56]	PLL3 [57]	This-thesis (Diff/Single-ended)
Frequency	1GHz	1.25GHz	1.7GHz	110MHz
Jitter	43.20°	4.95°	0.79°	0.54°
Power Consumption	1.5mW	109mW	60mW	5.4mw/2.7mW

Table 5.2: Overall performance of other designs

For comparison purposes similar designs from papers [55], [56] and [57] that use multiphase approach are presented with results in Table 5.2. Due to different working frequency, simulation results of the PLL from this thesis is normalized for power and frequency with PLLs from papers [55], [56] and [57]. Assuming that for each twofold increase in frequency power of the VCO and divider is doubled, Table 5.3 shows normalized results. It should be noted that when increasing VCO frequency, pulses from the charge pump will become narrower and total power consumption of the charge pump will decrease. However, since it is relatively insignificant compared to VCO and divider power consumption, this fact was excluded from the calculations. From Table 5.3 it can be concluded that design presented in this thesis has better noise/power performance compared to designs from [56] and [57], while for design [55] it is hard to come to conclusion due to high jitter.

Parameter	F=1GHz	F=1.25GHz	F=1.7GHz
Jitter	0.54^o	0.54^o	0.54^o
Power Consumption	38.24mW	46.54mW	65.00mW

Table 5.3: Normalizing PLL frequency to presented papers

Chapter 6

Conclusion

This chapter will summarize what was designed and what can be improved, as well as discuss some issues in the design.

6.1 Accomplishments

In this thesis, the design architecture that promises a lower quantization noise in $\Sigma\Delta$ PLLs is presented. Both calculations and measurements for the full PLL and VCO were presented. It was shown that quantization noise is usually a limiting factor in widening the bandwidth of the PLL. Therefore, by lowering the quantization noise it is possible to use a wider bandwidth for faster locking time or, more often, for direct modulation of the signal. A direct modulation allows for an exclusion of the DAC and mixer in the transmit chain and lowers power consumption.

Furthermore, a ring oscillator without voltage-to-current control was proposed in order to create a multiphase VCO with low phase noise. Tuning was performed by using a loop filter voltage to minimize flicker noise. As a result, it was found that the

single-ended oscillator achieves a better phase noise performance and a lower power consumption. The increase in the power supply sensitivity is noticeable in the single-ended design but with proper grounding and supply isolation it was found not to be an issue. Furthermore, a phase selector circuit with an accompanying divider was designed, which allowed for a proper selection of the phases.

The PLLs consumed 5.4mW and 2.88mW from a 1.8V supply for the differential and single-ended designs respectively. The whole design occupied $800\mu m$ by $1600\mu m$ including pads, a serial-to-parallel circuit, and output buffers. The active area of the PLL is estimated to be $600\mu m \times 600\mu m$.

6.2 Issues in the Design

The multiphase design is very sensitive to the order of the phases from the VCO. Therefore, there must be an additional circuit that will insure that the phases reach the divider in a specific order. Moreover, the delay in the startup circuit should not be more than the VCO period divided by a number of the phases. This can be a major issue in multi-gigahertz designs where the reset signal will have a window of only a couple of hundreds of pico seconds to reset the dividers. In such a case, a better reset design would be required.

This design also had a phase inversion in the input of the divider that created large spurs when more than one multiphase level is used. Also, the clock tree in the $\Sigma\Delta$ was not properly synthesized, thereby inadvertently creating a very noisy design. Finally, given that standard cells were used, the $\Sigma\Delta$ and dividers were not optimized for power and speed.

6.3 Future Work

In order for multiphase architecture to work, significant improvements in the current design are required.

First, a $\Sigma\Delta$ needs to be properly synthesized. The present $\Sigma\Delta$ has the clock tree issue, which made it unstable for some frequencies, while it increased the quantization noise significantly for other frequencies. Also, if a higher number of phases are used (more than 4), even a second order $\Sigma\Delta$ would give good performance, thereby saving power consumption compared to the fourth order $\Sigma\Delta$. Furthermore, a better divider should be designed. The current design uses 8 dividers that work in parallel. However, the design presented in Figure 4.22 is expected to save a significant amount of power given that it has only one divider. Lastly, in order to further minimize the power consumption of the digital cells, custom cells should be designed and used throughout all digital parts of the PLL.

Appendix A

Verilog Code for Digital Circuits

A.1 Divider

A.1.1 Single Divider

```
////////////////////////////////////////////////////
//
//      Igor Miletic
//      100238306
//
//      A programable divider (2 to 16)
//      Inputs: clk, rst, div_ratio
//      outputs: clk_divided
//
////////////////////////////////////////////////////

module Divider(clk, rst, div_ratio, clk_divided);

input           clk;
input           rst;
input   [3:0]   div_ratio;
output          clk_divided;

wire            clk;
wire            rst;
```

```
wire    [3:0]   div_ratio;
reg             clk_divided;
reg     [3:0]   cntr;

always @(posedge clk or negedge rst)
begin
        if (!rst)
        begin
                clk_divided<=0;
                cntr<=0;
        end
        else                        //Reset counter and output
        begin
                if (cntr==div_ratio)//check if counter at its maximum
                begin               //if yes, output 1
                        clk_divided<=1;
                        cntr<=|clk_divided ? 0:1;
                end
                else                    // else count one more
                begin
                        clk_divided<=0;
                        cntr<=cntr+1;
                end
        end
end

endmodule
```

A.1.2 Divider Master Block

```
////////////////////////////////////////////////
//
//      Igor Miletic
//
//      8 divider from Divider.v
//      inputs: vco(8), rst, divider_ratio(4)
//      outputs: vco_divided(8)
//
```

```
////////////////////////////////////////////////

module Dividers(vco,rst, divider_ratio, vco_divided);

input   [7:0]   vco; // inputs to dividers
input           rst; // reset
input   [3:0]   divider_ratio; // divider ratio
output  [7:0]   vco_divided;   // outputs from dividers

wire    [7:0]   vco, vco_divided;
wire            rst;
wire    [3:0]   divider_ratio;
wire            out0, out1, out2, out3, out4, out5, out6, out7;

Divider div1(vco[0], rst, divider_ratio, out0);// divider for phase 1
Divider div2(vco[1], rst, divider_ratio, out1);// divider for phase 2
Divider div3(vco[2], rst, divider_ratio, out2);//         .
Divider div4(vco[3], rst, divider_ratio, out3);//         .
Divider div5(vco[4], rst, divider_ratio, out4);//         .
Divider div6(vco[5], rst, divider_ratio, out5);//         .
Divider div7(vco[6], rst, divider_ratio, out6);//         .
Divider div8(vco[7], rst, divider_ratio, out7);// divider for phase 8

assign  vco_divided={out7, out6, out5, out4, out3, out2, out1, out0};

endmodule
```

A.2 Sigma Delta Modulator

A.2.1 Filter

```
////////////////////////////////////////////////////////////////
//
//      Igor Miletic
//      100238306
//
```

```
//      4th order filter
//
//      coefficients numerator=[0 3 -4 3 1], denominator=[1]
//
//      Designed for use in Sigma Delta Modulator
//
////////////////////////////////////////////////////////////////////

module Filter(clk, rst, enb, datain, dataout);

   input         clk, rst, enb;
   input  [15:0] datain;
   output [15:0] dataout;

   wire         clk, rst, enb;
   wire [15:0]  datain;
   wire [15:0]  dataout;

   reg  [15:0]  mem1, mem2, mem3, mem4;   // delays

// filter implementation
   assign dataout=((mem1<<1)+mem1)-(mem2<<2)+((mem3<<1)+mem3)-mem4;

   always       @(posedge clk or negedge rst or posedge enb)
     begin
        if (enb) begin
                mem1<=mem1;
                mem2<=mem2;
                mem3<=mem3;
                mem4<=mem4;
        end
        else begin
           if (!rst) begin
                mem1<=0;
                mem2<=0;
                mem3<=0;
                mem4<=0;
           end
           else begin
                mem4<=mem3;
                mem3<=mem2;
```

```
                mem2<=mem1;
                mem1<=datain;
            end
        end
    end // always      @ (posedge clk or negedge rst)
endmodule // Filter
```

A.2.2 Sigma Delta Main Block

```
////////////////////////////////////////////////////////////////////
//
//      Igor Miletic
//      100238306
//
//      3-bit Mux
//
//      Inputs: clk, vco_divided[8], phase_selector[3]
//      Outputs: phase_out
//
////////////////////////////////////////////////////////////////////

module SigmaDelta(clk, rst, enb, datain, dataout);
   input  [15:0] datain;
   input         clk, rst,enb;
   output [4:0]  dataout;

   wire   [4:0]  dataout;
   wire   [4:0]  data1;
   wire   [15:0] diff1, filter_out;
   wire          clk, rst, enb;
   wire   [15:0] sum1, datain;

   Filter        fltr(clk, rst, enb, diff1, filter_out);

   assign        sum1=datain+filter_out;
   assign        data1=sum1[10] ? (sum1[15:11]+1):sum1[15:11];
   assign        diff1=sum1-(data1<<11);
```

```
   assign        dataout=data1;

endmodule
```

A.3 Phase Selector

```
////////////////////////////////////////////////////////////////
//
//      Igor Miletic
//      100238306
//
//      3-bit Mux
//
//      Inputs: clk, vco_divided[8], phase_selector[3]
//      Outputs: phase_out
//
////////////////////////////////////////////////////////////////

module PhaseSelector(clk,rst,enb,divider_ratio, vco_divided,
    phase_selector, step, phase_out, N);
 input [7:0] vco_divided;     // divided vco output 8 bits
 input [3:0] divider_ratio;  // divider ratio setable from outside
 input clk,rst,enb;    // clock, reset (low), enable (low)
 input [1:0] step;     // we can use either 1/8, 1/4, 1/2 or 1
                 // for resolution(step 0, 1, 2 and 3 respectively)
   input  [4:0] phase_selector; // output of sigma delta, with 1 bit
                                // for neg. and 1 bit for overflow
   output       phase_out;      // final divided output
   output [3:0] N;              // dividers division ratio, based on
                                // SD output and divider_ratio

   wire   [7:0] vco_divided;
   wire   [3:0] divider_ratio;
   wire   [4:0] phase_selector;
   wire         phase_out, clk, rst,enb;
   reg    [7:0] cntr;
```

```
    wire   [3:0] N, step2;  // step2 is a wire that is either all 1
            // or 0, used to expand phase_selector variable to 7 bits
    wire   [1:0] step;
    reg [2:0]    cntr1;     // a register used to memorize previous
                            // output
    wire   [4:0] divider_change; // how much will divider ratio change
                                 // on next step
    wire   [7:0] phase_selector_new; // expanded phase selector

    assign       step2={4{phase_selector[4]}};
    assign       phase_selector_new={step2,phase_selector};


    assign       phase_out=vco_divided[(cntr1)];
    assign       N=rst ? (divider_ratio+divider_change):divider_ratio;
    assign       divider_change=cntr[7:3];

    always @(posedge clk or negedge rst or posedge enb)
      begin
         if (enb) begin
            cntr<=0;
            cntr1<=0;
         end
         else begin
            if (!rst) begin
               cntr<=0;
               cntr1<=0;
            end
            else begin
               cntr<=cntr1+(phase_selector_new<<step);
               cntr1<=cntr1+(phase_selector[2:0]<<step);
            end
         end
      end // always @ (posedge clk or negedge rst or negedge enb)
endmodule // PhaseSelector
```

A.4 Linear Feedback Shift Register

```
        //                                  -*- Mode: Verilog -*-
// Filename      : LFSR.v
// Description   : Linear Feedback Shift Register, used as pseudo
//       random number generator, if used with 12MHz clock
//       periodicity is 5.6Hz
// Author           : Igor Miletic
// Created On       : Wed Oct 22 19:35:52 2003
// Last Modified By: .
// Last Modified On: .
// Update Count     : 0
module LFSR(clk, enb, rst, yout);
   input clk, rst, enb;
   output yout;

   reg [20:0] poly;
   wire       yout, clk, rst, enb;

   always @(posedge clk or negedge rst or posedge enb)
     begin
        if (enb) begin
          poly<=poly;
        end
        else begin
           if (!rst) begin
              poly<=21'b111100001111000011111;
           end
           else begin
              poly<={poly[0], poly[20:4], poly[0]+poly[3], poly[2:1]};
           end
        end
     end
   assign yout=poly[0];
endmodule // LFSR
```

A.5 Synchronized Reset

```
//                                -*- Mode: Verilog -*-
// Filename         : SyncUnreset.v
// Description      : It asyncronously resets, but it
//                     syncronously un resets circuit
// Author           : Igor Miletic
// Created On       : Tue Oct 28 11:30:32 2003
// Last Modified By: .
// Last Modified On: .
// Update Count     : 0

module SyncUnreset(clk, ask_rst, rst) ;
   input  clk, ask_rst;
   output rst;

   wire   clk, ask_rst;
   reg    rst;

   always @ ( posedge clk or negedge ask_rst) begin
      if (!ask_rst) begin
         rst<=ask_rst;
      end
      else  begin
         rst<=1;
      end
   end
endmodule // SyncUnreset
```

A.6 Top Level

```
// Filename         : AllDigital.v
// Description: Top Digital code, integrating dividers+sigma delta
// Author           : Igor Miletic
// Created On       : Tue Oct 21 16:08:08 2003
```

```
// Last Modified By: .
// Last Modified On: .
// Update Count    : 0

module AllDigital(ask_rst,enb, vco, divider_ratio, step, datain,
                  randomize, yout, phase_selector, vco_clk);
   input        ask_rst,enb, randomize;
   input  [1:0] step;
   input  [3:0] divider_ratio;
   input  [7:0] vco;
   input [15:0] datain;

   output       yout, vco_clk;
   output [4:0] phase_selector;

   wire         clk, rst, enb, ask_rst, randomize;
   wire   [1:0] step;
   wire   [4:0] phase_selector;
   wire   [3:0] divider_ratio;
   wire  [15:0] datain;
   wire         yout, vco_clk, rnd1;
   wire   [7:0] vco_divided;
   wire   [3:0] N;

   Dividers       divs(vco,rst, N, vco_divided);
   PhaseSelector  ps(vco_divided[0],rst, enb, divider_ratio,
                     vco_divided, phase_selector, step, yout,N);
   SigmaDelta sd(vco_divided[0], rst,enb, datain, phase_selector);
   SyncUnreset    unrst(vco[7], ask_rst, rst);
   LFSR           lfsr1(vco_divided[0], enb, rst, rnd1);

endmodule // AllDigital
```

Appendix B

Schematics

In this appendix all the schematics that were not placed in the thesis are presented. First schematic is top level.

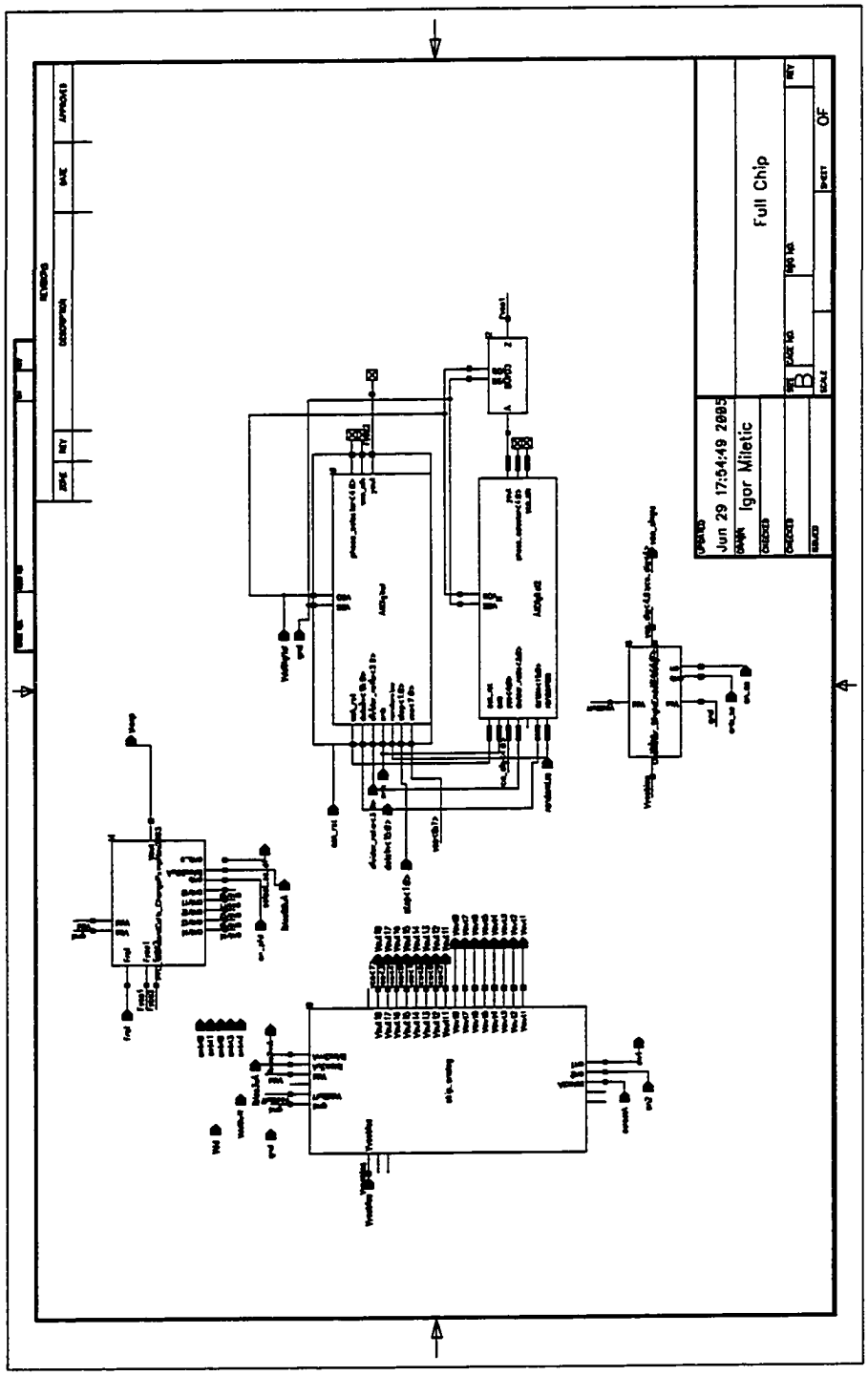


Figure B.1: Top level schematic, full chip.

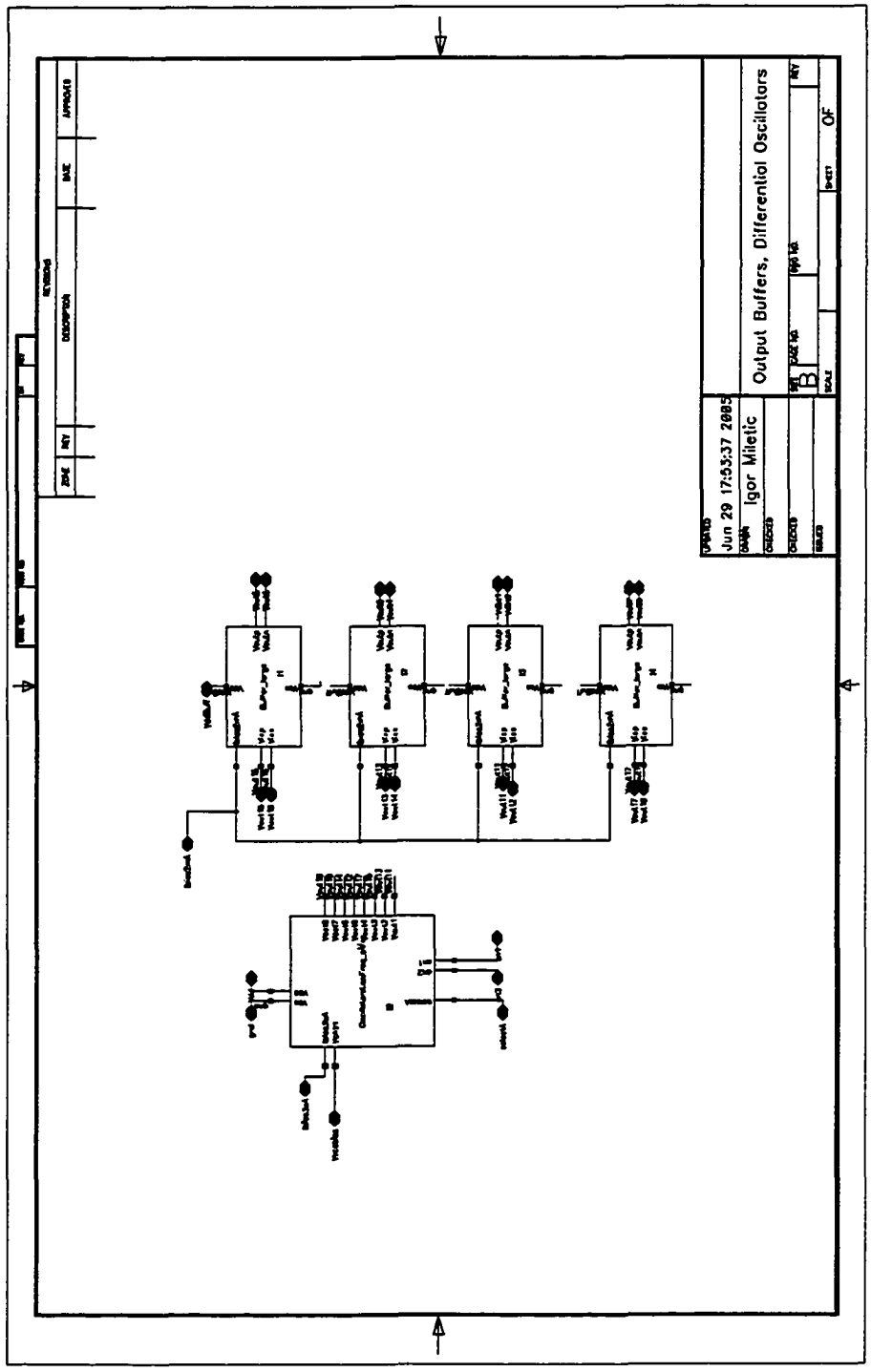


Figure B.2: Differential oscillators with output buffers.

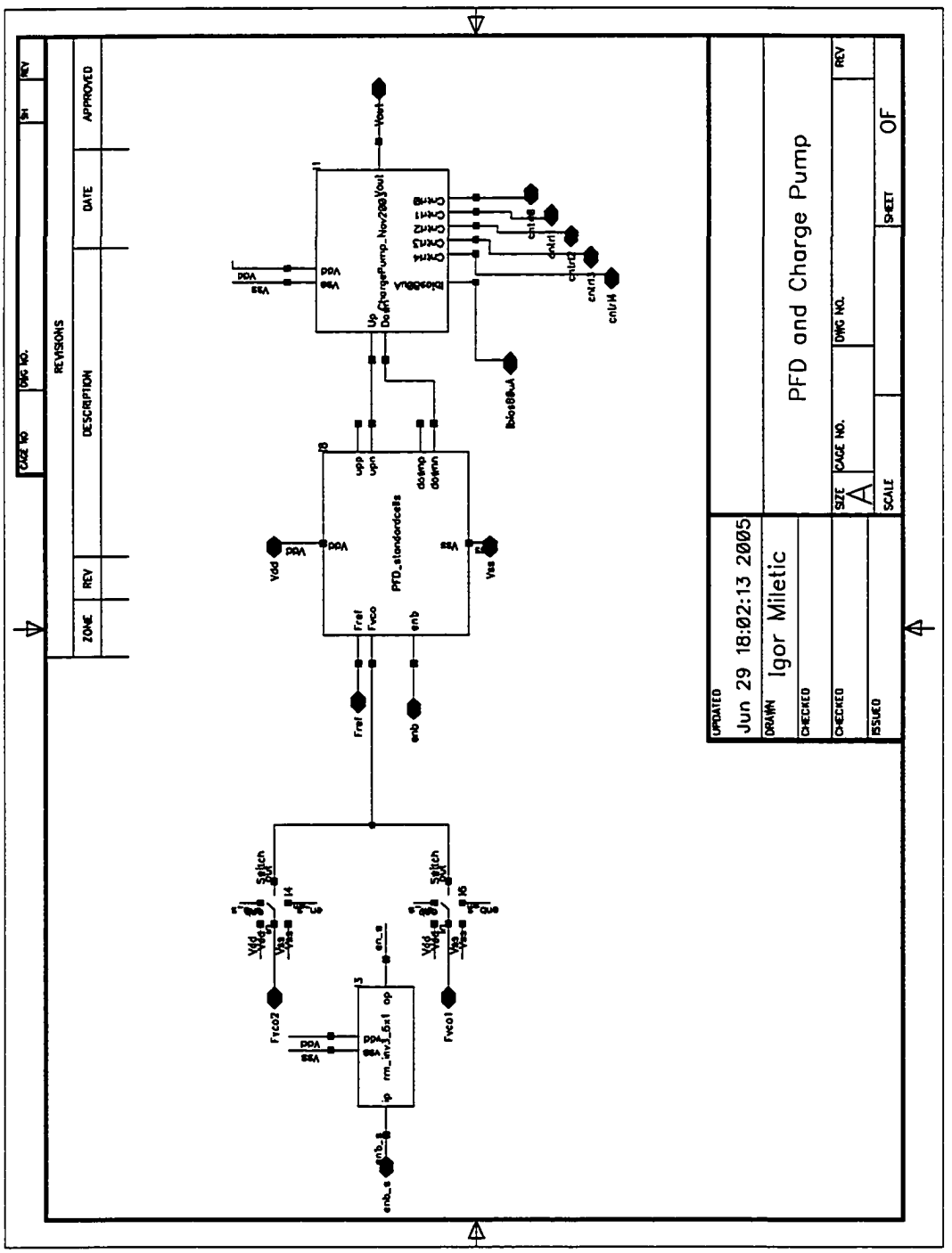


Figure B.3: Charge pump with PFD.

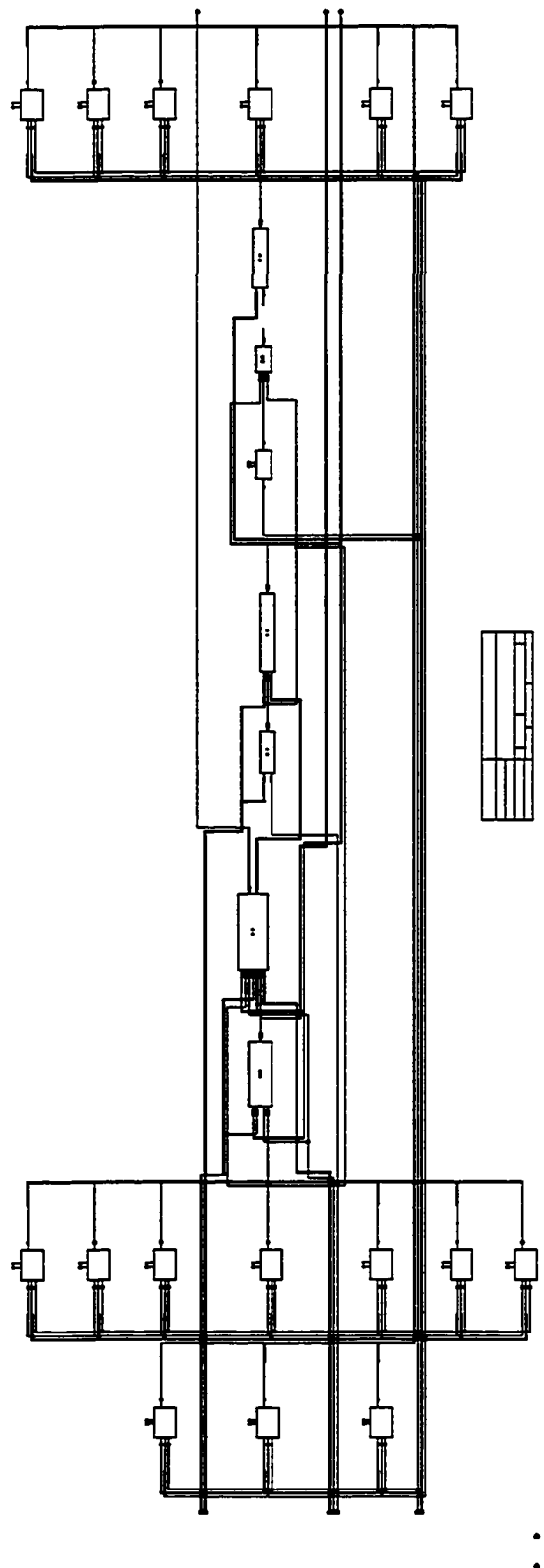


Figure B.4: Synthesized digital block for differential oscillator.

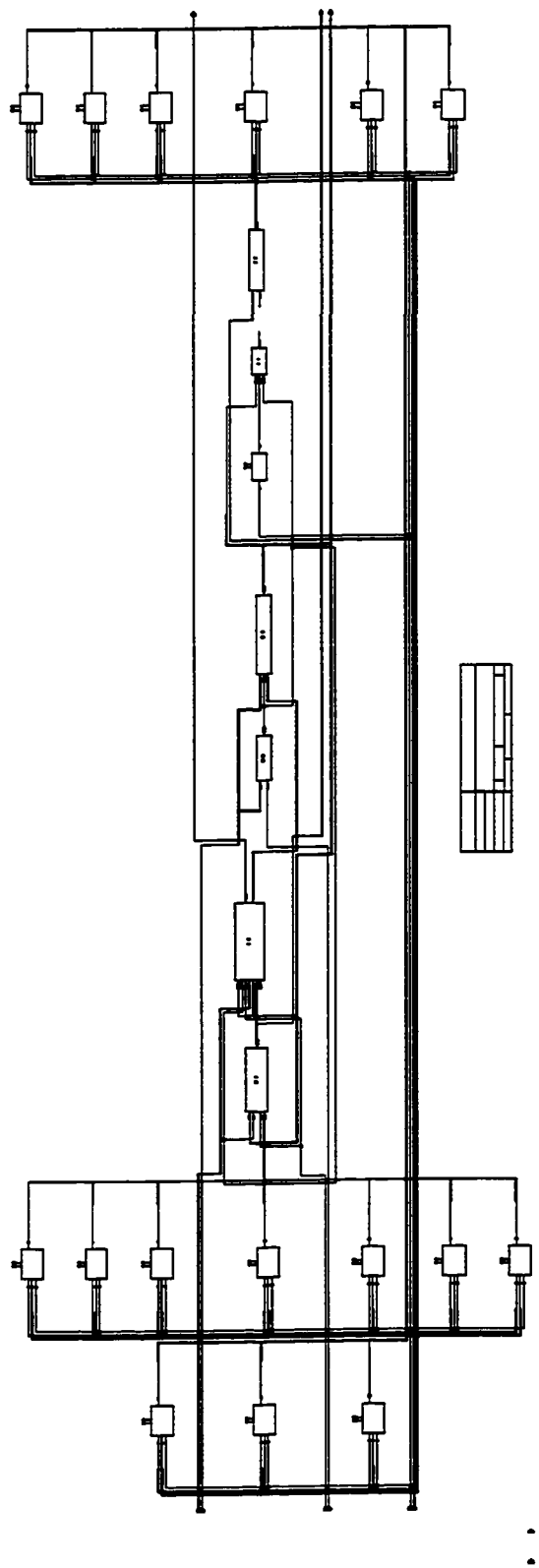


Figure B.5: Synthesized digital block for single-ended oscillator.

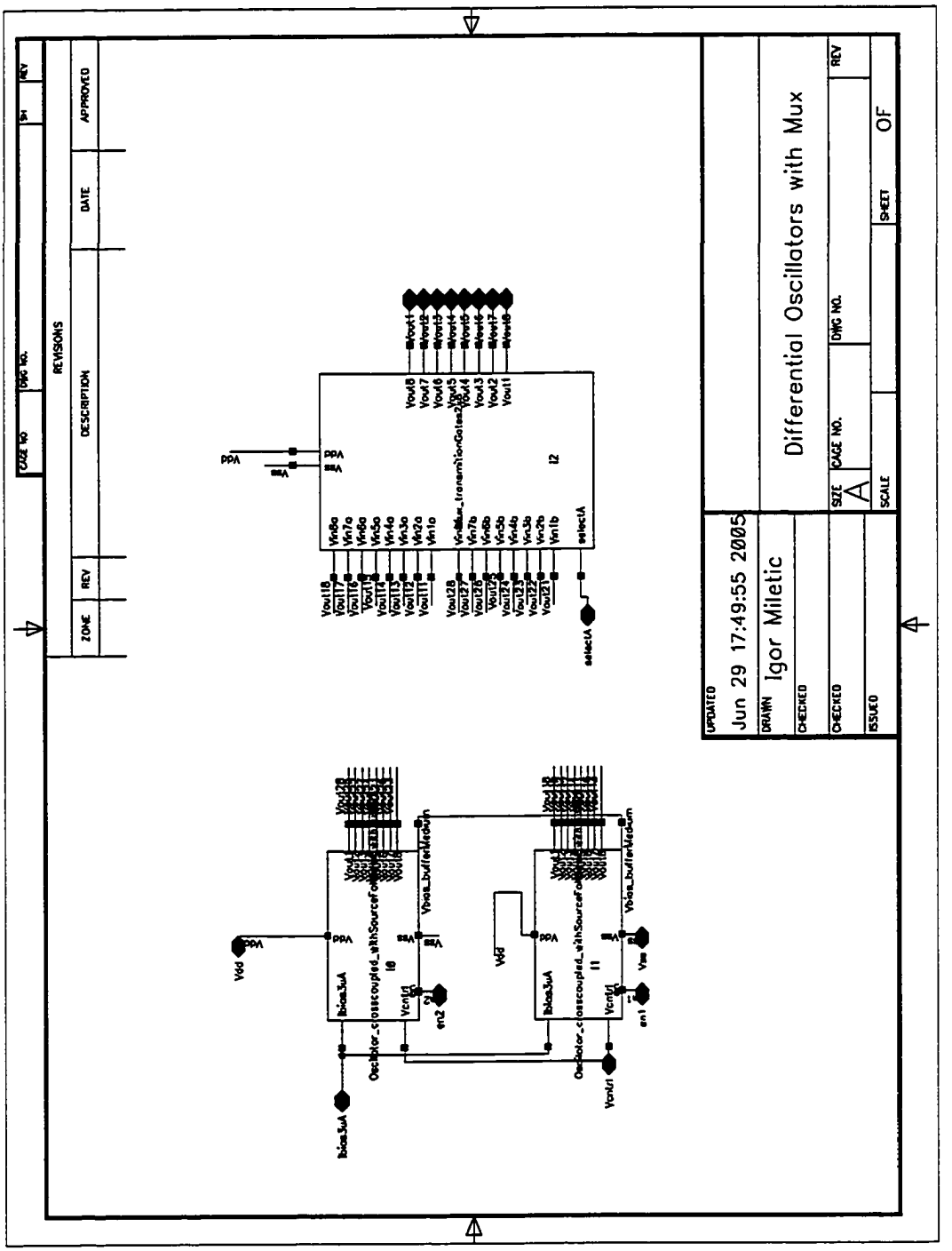


Figure B.6: Both differential oscillators with mux.

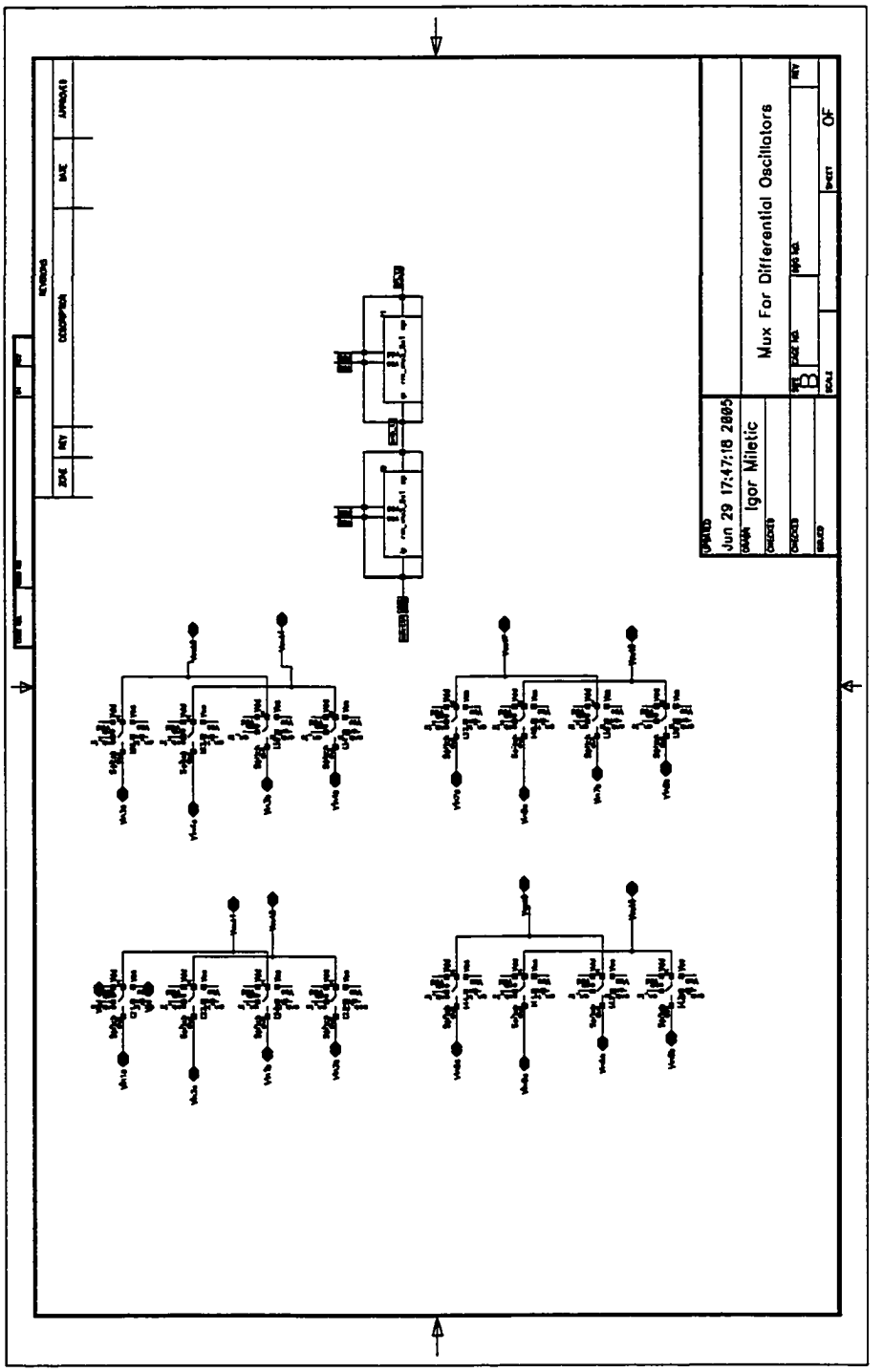


Figure B.7: Multiplexer for the differential oscillators.

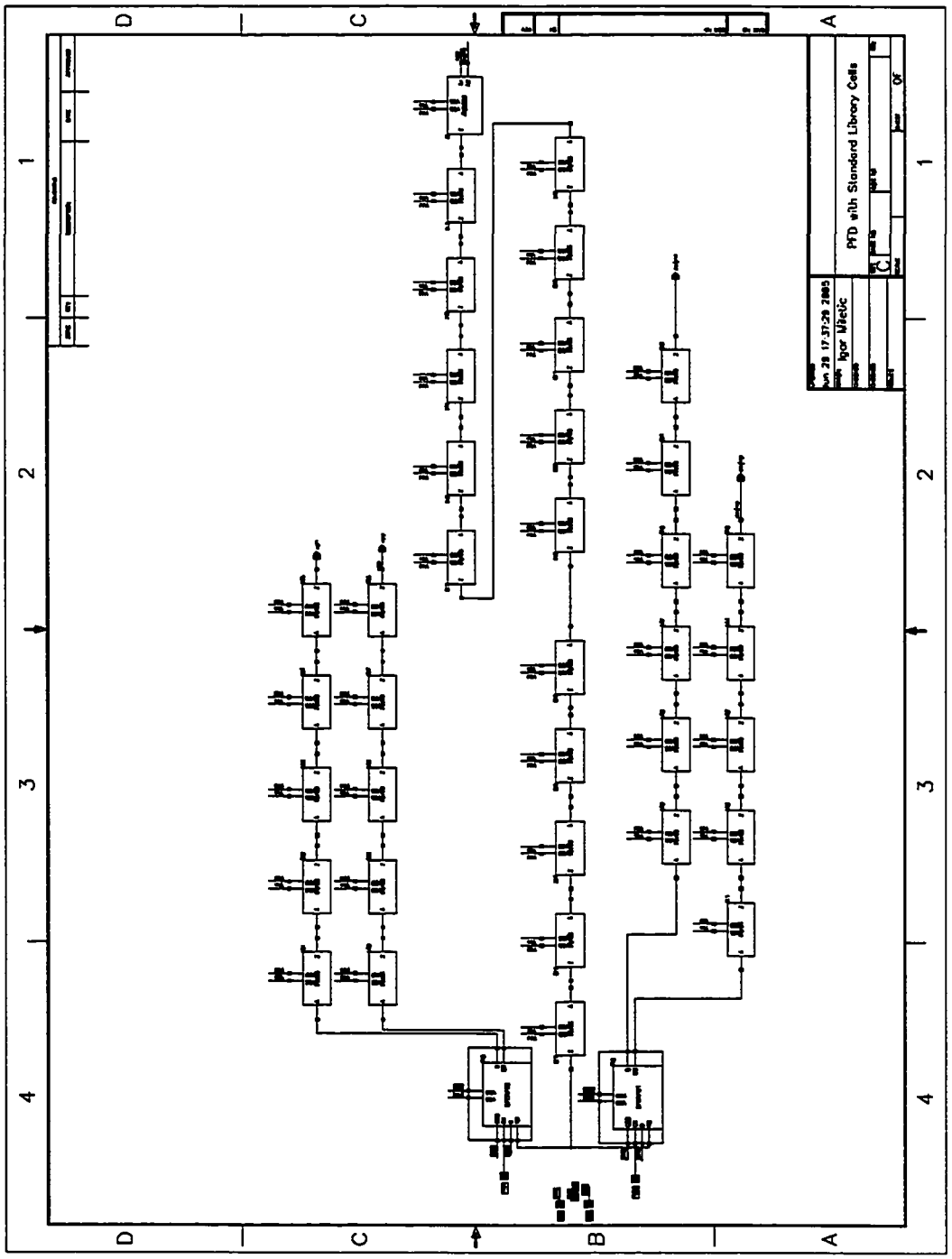


Figure B.8: Phase frequency detector.

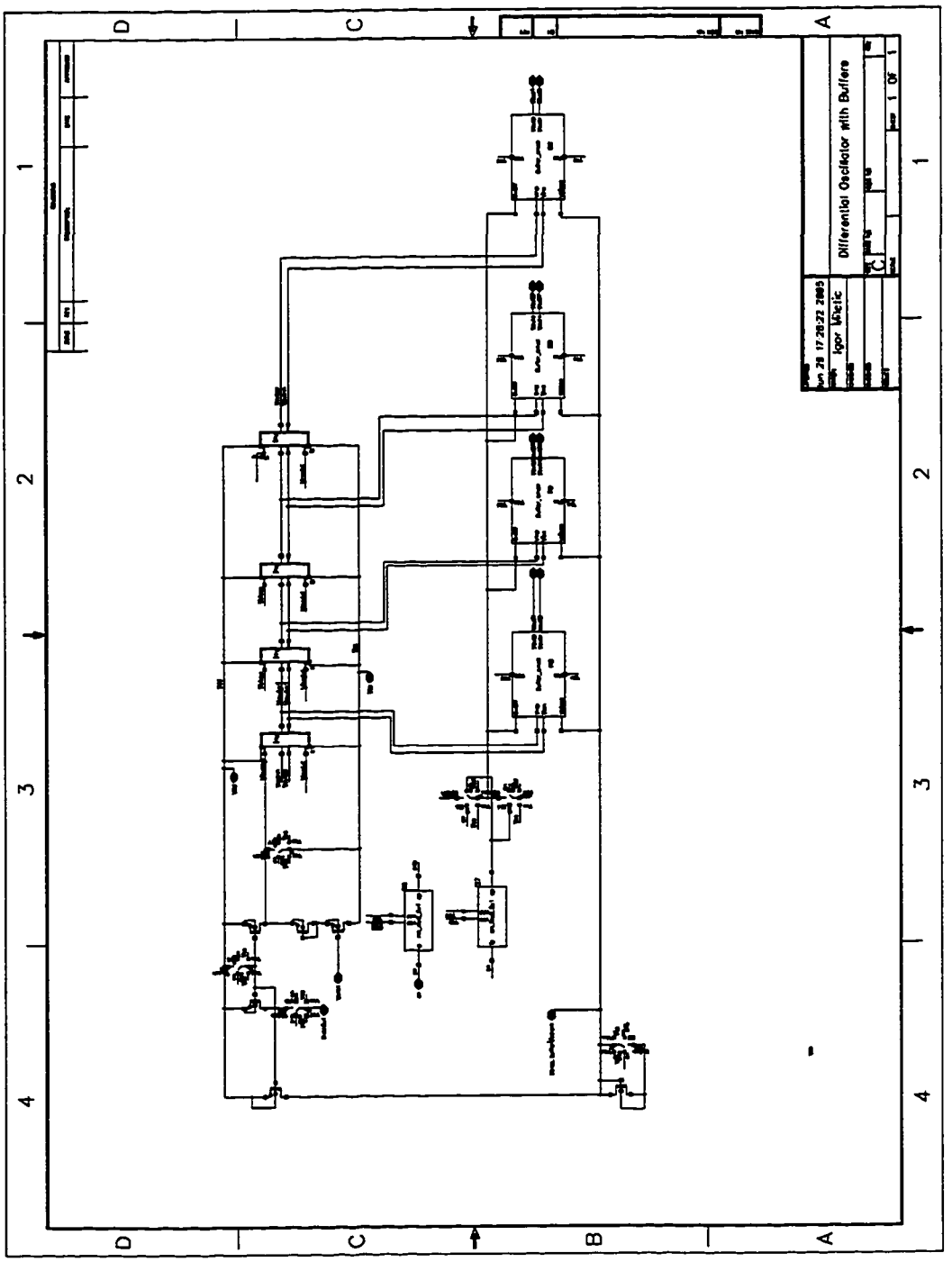


Figure B.9: Differential ring oscillator with buffers.

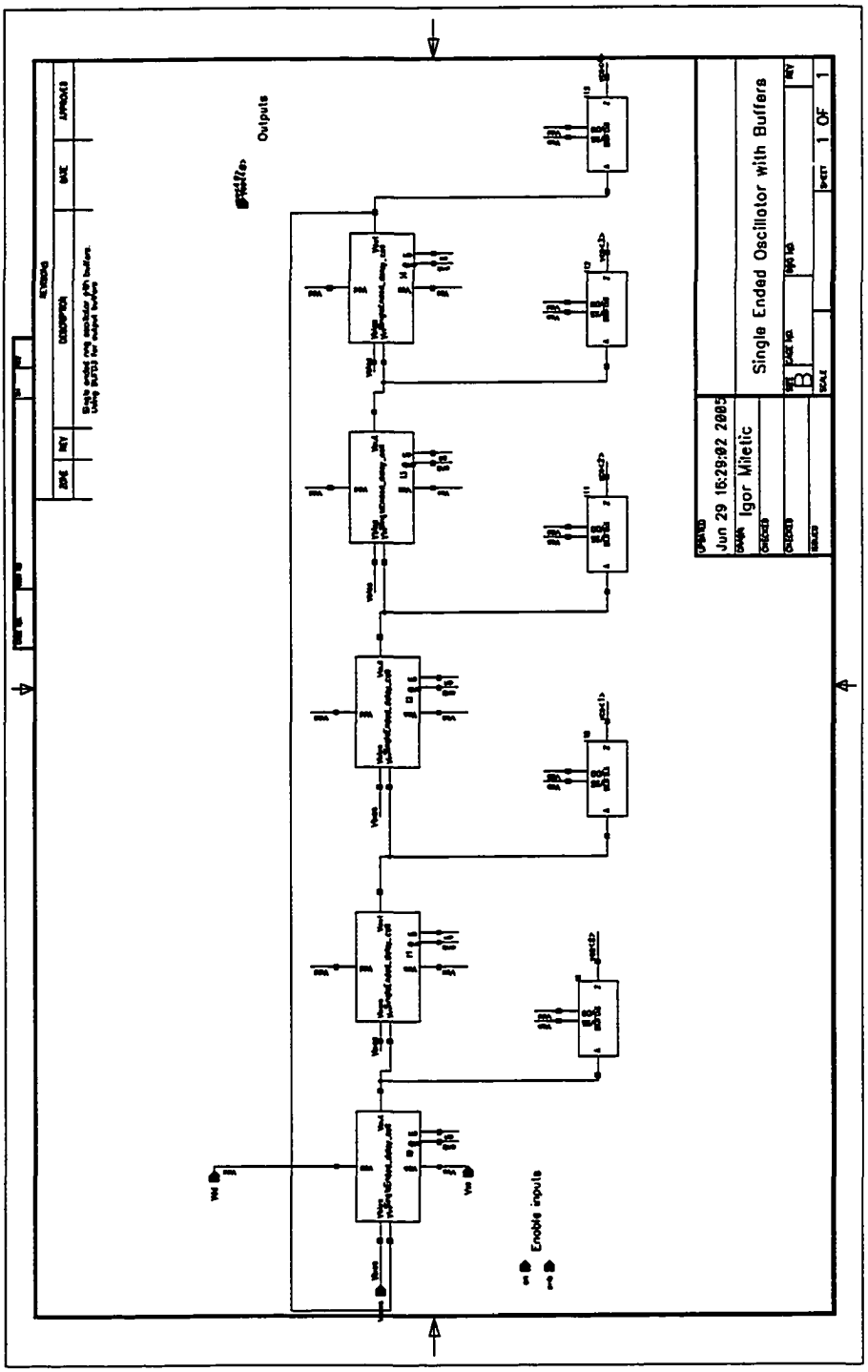


Figure B.10: Single-ended ring oscillator with buffers.

References

[1] F.M. Gardner, *Phaselock Techniques*, Wiley & Sons, New York, 1979.

[2] U.L. Rohde, *Digital PLL Frequency Synthesizers, Theory and Design*, Printice-Hall, Englewood Cliffs, NJ, 1997.

[3] Y. Kado, T. Ohno, and et. al, "An ultralow power cmos/simox programmable counter lsi," *IEEE J. Solid-State Circuits*, vol. 32, no. 10, pp. 1582–87, October 1997.

[4] W. Rhee, "Design of high-performance cmos charge pumps in phase-locked loops," in *Circuits and Systems.* 30 May - June 2 1999, vol. 2, pp. 545–548, Proceedings of the 1999 IEEE International Symposium on.

[5] Wei-Zen Chen, Jia-Xian Chang, Ying-Jen Hong, Meng-Tzer Wong, and Chien-Liang Kuo, "A 2-v 2.3/4.6-ghz dual-band frequency synthesizer in 0.35-μ m digital cmos process," *Solid-State Circuits, IEEE Journal of*, vol. 39, no. 1, pp. 234 – 237, January 2004.

[6] M.H. Perrott, "Fast and accurate behavioral simulation of fractional-n frequency synthesizers and other pll/dll circuits," *Proceedings of 39th Design Automation Conference*, p. 498, 2002.

[7] Donhee Ham and Ali Hajimiri, "Design and optimization of a low noise 2.4ghz cmos vco with integrated lc tank and moscap tuning," in *IEEE International Symposium on Circuits and Systems*, 28-31 May 2000, vol. 1, pp. 331 – 334.

[8] Nhat M. Nguyen and Robert G. Meyer, "Start-up and frequency stability in high-frequency oscillators," *Solid-State Circuits, IEEE Journal of*, vol. 27, no. 5, pp. 810–820, 5 May 1992.

[9] Behzad Razavi, "A 2-ghz 1.6-mw phase-locked loop," *IEEE Journal of Solid-State Cuircuits*, vol. 32, no. 5, pp. 730–735, May 1997.

[10] Thomas H. Lee, *The Design of CMOS Radio-Frequency Integrated Circuits*, Cambridge University Press, 1998.

[11] S.I. Ahmed and R.D. Mason, "A dual edge-triggered phase-frequency detector architecture [frequency synthesizer applications]," *Proc. of the 2003 IEEE International Symposium on Circuits and Systems*, vol. vol.1, pp. 721 – 4, 2003//.

[12] P.V. Brennan, "Performance of phase-locked loop frequency synthesiser using accumulative phase detector," *IEE Proceedings-Circuits, Devices and Systems*, vol. 143, no. 5, pp. 249 – 54, 1996/10/.

[13] D. Baker, "Reference cancelling phase/frequency detector," *Proceedings RF Technology Expo 90*, pp. 305 – 15, 1990//.

[14] J. Crawford, "The phase/frequency detector," *R.F. Design*, vol. 8, no. 2, pp. 46 – 57, 1985/02/.

[15] D. Vyroubal, "Low frequency phase-detector," *Elektrotehnika*, vol. 22, no. 1-2, pp. 25 – 32, 1979//.

[16] P. Lefevre, N. Haese, G. Lewandowski, C. Loyez, P.-A. Rolland, E. Rogeaux, and J.-C. Sarkissian, "A full gaas mmic phase-locked oscillator using sampling phase detector and a sampling frequency detector," *30th European Microwave Conference 2000. Conference Proceedings*, vol. vol.3, pp. 112 - 15, 2000//.

[17] N. Beverini, M. Prevedelli, F. Sorrentino, B. Nyushkov, and A. Ruffini, "An analog + digital phase-frequency detector for phase locking of a diode laser to an optical frequency comb," *Kvantovaya Elektronika, Moskva*, vol. 34, no. 6, pp. 559 - 64, 2004/06/.

[18] C. Piguet and V. von Kaenel, "Logic synthesis of a pll phase frequency detector," *IEE Proceedings-Computers and Digital Techniques*, vol. 144, no. 6, pp. 381 - 5, 1997/11/.

[19] M. Sie, G. Cibiel, T. Tournier, R. Plana, and J. Graffeuil, ," *2003 IEEE MTT-S International Microwave Symposium Digest*, 2003//.

[20] R. Howald and K. Marra, "A high-speed phase-frequency detector asic with programmable dividers for lo generator and indirect synthesis applications," *Proceedings of the 1995 IEEE International Frequency Control Symposium.*, pp. 250 - 8, 1995//.

[21] L. Moura and A. Borjak, "A modular phase-frequency detector design with +or- 2n pi linear range of operation," *Circuits, Systems, and Signal Processing*, vol. 23, no. 4, pp. 273 - 83, 2004/07/.

[22] Ching-Yuan Yang and Shen-Iuan Liu, "Fast-switching frequency synthesizer with a discriminator-aided phase detector," *IEEE Journal of Solid-State Circuits*, vol. 35, no. 10, pp. 1445 - 52, 2000/10/.

[23] V. Warnkross, "Faster phase and frequency detector," *Elektronik*, vol. 28, no. 21, pp. 85 – 6, 1979/10/18.

[24] Y. Sumi, K. Syoubu, S. Obote, Y. Fukui, and Y. Itoh, "Pll frequency synthesizer with multi-phase detector," *IEICE Trans. on Fund. of Electronics, Communications and Computer Sciences*, vol. E82-A, no. 3, pp. 431 – 5, 1999/03/.

[25] J.I. Brown, "A digital phase and frequency-sensitive detector," *Proceedings of the IEEE*, vol. 59, no. 4, pp. 717 – 18, 1971/04/.

[26] J.W.M. Rogers, M. Cavin, F. Dai, and D. Rahn, "A delta sigma fractional-n frequency synthesizer with a multi-band pmos vcos for 2.4 and 5ghz wlan applications," p. 651 4, 2003//.

[27] H. Arora, N. Klemmer, J.C. Morizio, and P.D. Wolf, "Enhanced phase noise modieling of fractional-n frequency synthesizers," *IEEE Transations On Circuits and Systems*, vol. 52, no. 2, pp. 379–395, February 2005.

[28] Jim Wight, "Elec5606 phase-locked loops and receiver synchronizers," Carleton University, 2002.

[29] J.W.M. Rogers, F.F. Dai, M.S. Cavin, and D.G. Rahn, "A multiband delta sigma fractional-n frequency synthesizer for a mimo wlan transceiver rfic," *IEEE Journal of Solid-State Circuits*, vol. 40, no. 3, pp. 678 – 89, 2005/03/.

[30] B. De Muer and M.S.J. Steyaert, "A cmos monolithic delta sigma -controlled fractional-n frequency synthesizer for dcs-1800," *IEEE Journal of Solid-State Circuits*, vol. 37, no. 7, pp. 835 44, 2002/07/.

[31] B.G. Goldberg, "The evolution and maturity of fractional-n pll synthesis," *Microwave Journal, Euro-Global Edition*, vol. 39, no. 9, pp. 124, 126, 128, 130, 132, 134, Sept. 1996.

[32] Gabor C. Temes Steven R. Norsworthy, Richard Schreier, *Delta-Sigma Data Converters, Theory, Design, and Simulation*, IEEE Press, 1996.

[33] James C. Candy and Oconnell J. Benjamin, "Structure of quantization noise from signma-delta modulation.," vol. CM-29, no. 9, pp. 1316 1323, 1981.

[34] Vladimir Friedman, "Structure of the limit cycles in sigma delta modulation.," *IEEE Trans. on Communications*, vol. 36, no. 8, pp. 972 979, 1988.

[35] Stuart K. Tewksbury and Robert W. Hallock, "Oversampled, linear predictive and noise-shaping coders of order n greater than 1.," vol. CAS-25, no. 7, pp. 436 447, 1978.

[36] S. Pamarti, L. Jansson, and I. Galton, "A wideband 2.4-ghz delta-sigma fractional-npll with 1-mb/s in-loop modulation," *IEEE Journal of Solid-State Circuits*, vol. 39, no. 1, pp. 49 62, 2004/01/.

[37] Fabrice Sthal, Serge Galliou, Nicolas Gufflet, and Marc Mourey, "Predicting phase noise in crystal oscillators," in *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, January 2005, vol. 52.

[38] L.N. Alves and R.L. Aguiar, "Noise performance of classical current mirrors," in *Electronics, Circuits and Systems, 9th International Conference on*, Sept. 2002, vol. 1, pp. 277–280.

[39] T.Danelle Au and Kelvin Khoo, "Improved flicker noise model for submicron mosfet devices," Department of Electrical Engineering and Computer Science Univesity of California at Berkeley.

[40] A. Hajimiri, S. Limotyrakis, and T.H. Lee, "Jitter and phase noise in ring oscillators," *IEEE Journal of Solid-State Circuits*, vol. 34, no. 6, pp. 790 – 804, 1999/06/.

[41] A. Hajimiri and T.H. Lee, "A general theory of phase noise in electrical oscillators," *IEEE Journal of Solid-State Circuits*, vol. 33, no. 2, pp. 179 – 94, 1998/02/.

[42] T.A.D. Riley, N.M. Filiol, Qinghong Du, and J. Kostamovaara, "Techniques for in-band phase noise reduction in delta sigma synthesizers," *IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 50, no. 11, pp. 794 803, 2003/11/.

[43] H. Ahmed, C. DeVries, and R. Mason, "A digitally tuned 1.1 ghz subharmonic injection-locked vco in 0.18 mu m cmos," *ESSCIRC 2004 - 29th European Solid-State Circuits Conference*, pp. 81 – 4, 2003//.

[44] "802.15.4: Wireless medium access control (mac) and physical layer (phy) specifications for low-rate wireless personal area networks (lr-wpans)," The Institute of Electrical and Electronics Engineers, Inc., 2003.

[45] T.C. Weigandt, Beomsup Kim, and P.R. Gray, "Analysis of timing jitter in cmos ring oscillators," *1994 IEEE International Symposium on Circuits and Systems*, vol. vol.4, pp. 27 – 30, 1994//.

[46] A.A. Abidi and R.G. Meyer, ," , no. 6, 1983/12/.

[47] Kamran Iravani and Gary Miller, "Vcos with very low sensitivity to noise on the power supply," *Proceedings of the Custom Integrated Circuits Conference*, pp. 515 – 518, 1998.

[48] P. Larsson, "Measurements and analysis of pll jitter caused by digital switching noise," *IEEE Journal of Solid-State Circuits*, vol. 36, no. 7, pp. 1113 – 19, 2001/07/.

[49] Liang Dai and R. Harjani, "Design of low-phase-noise cmos ring oscillators," *IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 49, no. 5, pp. 328 – 38, 2002/05/.

[50] S.L.J. Gierkink, A. van der Wel, G. Hoogzaad, E.A.M. Klumperink, and A.J.M. van Tuijl, "Reduction of the 1/f noise induced phase noise in a cmos ring oscillator by increasing the amplitude of oscillation," *ISCAS '98. Proc. of the 1998 IEEE Int. Symposium on Circuits and Systems*, vol. vol.1, pp. 185 – 8, 1998//.

[51] F. Herzel and B. Razavi, "A study of oscillator jitter due to supply and substrate noise," *IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 46, no. 1, pp. 56 – 62, 1999/01/.

[52] E. Temporiti, G. Albasini, I. Bietti, R. Castello, and M. Colombo, "A 700-khz bandwidth sigma delta fractional synthesizer with spurs compensation and linearization techniques for wcdma applications," *IEEE Journal of Solid-State Circuits*, vol. 39, no. 9, pp. 1446 54, Sept. 2004.

[53] F. M. Gardner, "Chargepump phase-locked loops," in *IEEE Trans. on Communications*, November 1980, vol. COM-28, pp. 1849–1858.

[54] P. L'Ecuyer and F. Panneton, "A new class of linear feedback shift register generators," *2000 Winter Simulation Conference Proceedings*, vol. vol.1, pp. 690 - 6, 2000//.

[55] M. Kossel, T. Morf, W. Baumberger, A. Biber, C. Menolfi, T. Toifl, and M. Schmatz, "A multiphase pll for 10 gb/s links in soi cmos technology," *2004 IEE Radio Frequency Integrated Circuits (RFIC) Systems. Digest of Papers*, p. 207, 2004.

[56] Lizhong Sun and T.A. Kwasniewski, "A 1.25-ghz 0.35- mu m monolithic cmos pll based on a multiphase ring oscillator," 2001.

[57] Chan-Hong Park, Ook Kim, and Beomsup Kim, "A 1.8-ghz self-calibrated phase-locked loop with precise i/q matching," *IEEE Journal of Solid-State Circuits*, vol. 36, no. 5, pp. 777 83, 2001/05/.