A Low Phase Noise Ring Oscillator Phase-Locked Loop for Wireless Applications

by

Colin Weltin-Wu

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Masters of Engineering in Electrical Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 2005 [June 2005]

Author .....

Department of Electrical Engineering and Computer Science
May 6, 2005

Certified by .....

James K. Roberge
Professor of Electrical Engineering
Thesis Supervisor

Accepted by .....

Arthur C. Smith
Chairman, Department Committee on Graduate Students

A Low Phase Noise Ring Oscillator Phase-Locked Loop for Wireless Applications

by

Colin Weltin-Wu

Submitted to the Department of Electrical Engineering and Computer Science
on May 6, 2005, in partial fulfillment of the
requirements for the degree of
Masters of Engineering in Electrical Engineering

Abstract

This thesis describes the circuit level design of a 900MHz $\Sigma\Delta$ ring oscillator based phase-locked loop using 0.35μm technology. Multiple phase noise theories are considered giving insight into low phase-noise voltage controlled oscillator design. The circuit utilizes a fully symmetric differential voltage controlled oscillator with cascode current starved inverters to reduces current noise. A compact multi-modulus prescaler is presented, based on modified true single-phase clock flip-flops with integrated logic. A fully differential charge pump with switched-capacitor common mode feedback is utilized in conjunction with a nonlinear phase-frequency detector for accelerated acquisition time.

Thesis Supervisor: James K. Roberge
Title: Professor of Electrical Engineering

Acknowledgments

Through the course of my thesis, I have been fortunate to receive assistance from many people. Rather than trying to exhaustively list them all only to leave a few conspicuous in absentia, I will now thank those critical to the completion of my thesis. For all of you with whom I had conversations or bounced ideas around, thank you and I hope you had as much fun as I did.

First, I would like to thank my advisor, Professor James K. Roberge. It is his class on feedback systems which first drew me into analog circuit design. As a teacher, his clarity helped me quickly understand more than I could have possibly hoped to learn on my own. As an advisor, his deep insight and infinite experience was an invaluable tool and time saver in my thesis process. For every time I tried to throw another digital gate into my thesis, his steadfast (and only half joking) advice of "why not do it in analog?" set me straight.

Professor Rahul Sarpeshkar was not only a fantastic teacher, his obvious enthusiasm and love for his work inspired me to do the same with mine.

Michael Perrott's simulation tools were an integral part of this thesis, and his detailed class notes helped clarify concepts where textbooks failed to do so.

I also want to thank Professor Serge Lang at Yale University, for always insisting I know the difference between a fact and a hole in the ground, and Anne Hunter in the undergraduate office, for being the most reliably comforting and encouraging support a scatterbrained student could ask for.

Finally I thank my parents, for letting me have power tools over a Nintendo.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

The wireless communication world has so far been dominated by resonant tank based phase-locked loops. The reasons for this, as cited in numerous publications, is the bandpass filtering nature of resonant tanks helps reduce noise in the oscillator output, which dominates at frequencies above the loop bandwidth. Because wireless systems have such stringent phase noise specifications, it has so far been easier to use resonant oscillators even when the extra effort of designing integrated inductors is taken into account. The ring oscillator, meanwhile, is still widely used in CDR (clock and data recovery) and processor clock generation applications, where low phase noise (or jitter, in the digital world) takes second priority to wide tuning range. In this paper, we would like to develop a PLL which takes advantage of all the benefits of a ring-oscillator topology: wide tuning range, small size, and the ability to use an aging 0.35μm CMOS process. At the same time, we attempt the achieve as many GSM900 (Global System for Mobile communications) specifications as possible.

1.2 History

The Phase Locked Loop(PLL) in its current form was first mentioned in a 1932 paper by a French engineer, de Bellescize.[1] His paper described a way to send and receive

signals by demodulating the received signal by the carrier frequency with which it was originally modulated. As opposed to the heterodyne technique introduced in the 20's, the approach bypasses the IF (intermediate frequency) stage and directly brings the RF down to audio range in one step. The problem here is that not only does the demodulator need to have the exact frequency of the carrier, it needs to have a constant phase with respect the the carrier. Actually we'd like to avoid quadrature phase, because

$$\sin(\omega t)\sin(\omega t + \Delta t) = \sin(\omega t)\left(\sin(\omega t)\cos(\Delta t) + \cos(\omega t)\sin(\Delta t)\right) \tag{1.1}$$

so to ensure maximum signal power, we actually want exactly in or out of phase local oscillators. This idea has its share of problems, and its conception coincided with the rising success of the superheterodyne. The first widespread use of the PLL was in CRT (cathode ray tube) based television sets. In a CRT, several magnetic fields modulate a thin beam of electrons to "paint" the screen, and the screen is coated with chemicals to produce light. The motion of the beam follows a linear path across the screen, then returns to the beginning but shifted down by a line's width and repeats. Since televisions process incoming data in real time, the motion of the beam is integrally linked with the timing of the incoming data. To get this timing right, early sets used a primitive form of injection locked oscillators, whose performance suffered when faced with a degraded received signal. Signal degradation caused the oscillator to lose synchronization, resulting in visual artifacts such as line tearing, frame shaking, and moving bars across the picture. The introduction of the PLL allowed these circuits to be much more robust with respect to signal integrity. Later, when the NTSC color standard was developed, engineers decided to encode the whole color spectrum into the phase shift of a three odd MHz sine wave. To determine the phase shift, a PLL is again necessary.

Another important application of the PLL is to combat Doppler frequency shifts in space communication systems. As IC technology improved, so did the uses of the PLL increase. PLL's facilitate stereo FM radio, by utilizing phase as a method of

encoding a second variable of information in a signal. Nowadays, PLLs are used as frequency synthesizers for processors, signal generators, and pretty much anything else that relies on a digital clock.

1.3 Linear PLL Model

1.3.1 Dynamic Behavior

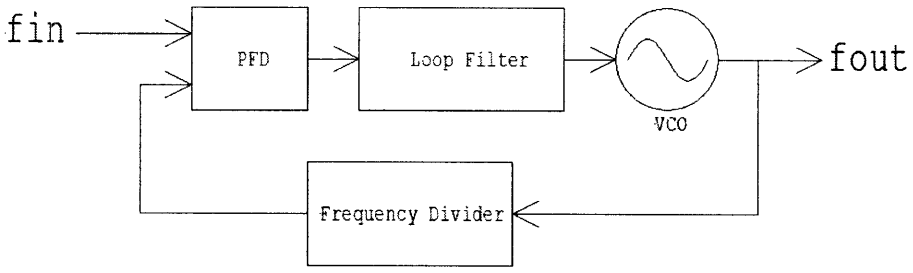

Figure 1-1: A general block diagram for an integer-N PLL.

A simple block diagram for an integer-N PLL is shown in figure 1-1. Such a PLL would be typically used as a frequency synthesizer. When used as a frequency synthesizer, the PLL takes as input a pure tone such as a crystal oscillator output, and multiplies this to create a periodic output N times higher in frequency. The application of this is the oscillation frequency of crystals does not extend much beyond 100Mhz, so modern microprocessors and communication circuits all require a precise way to multiply frequency. The additional benefit of using a PLL is by changing PLL parameters, an infinitum of output frequencies can be obtained.

The PLL is a feedback system which differs from the classic op-amp feedback example in that the variable of interest changes unit as we traverse the loop. Many factors contribute to the nonlinearity of this system, which makes accurate modeling of the system dynamics difficult. Before considering these higher-order effects, insight can be gained by treating the PLL as a classic feedback system.

Figure 1-2 shows figure 1-1 redrawn to facilitate feedback analysis. It may be

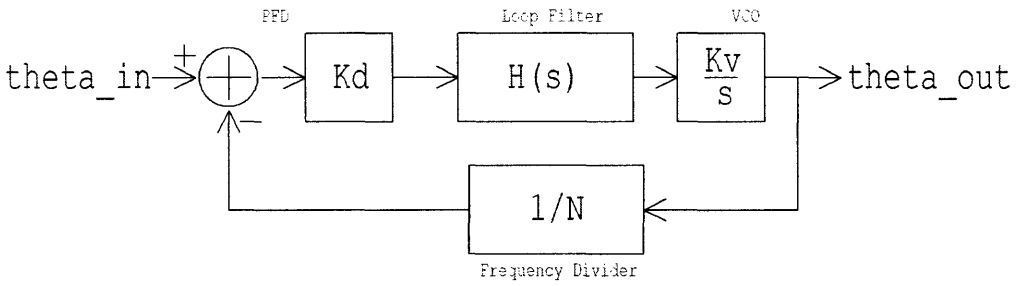

Figure 1-2: Block diagram for linear feedback analysis of a PLL.

confusing why we are concerned with phase detection if we want to do frequency multiplication. There are two reasons. The fundamental reason is that $\omega = d\theta/dt$, so if the feedback system works and drives the error to zero, we have "locked phase" and therefore the frequencies must also be equal. Secondly, as mentioned in section 1.2, some communication architectures such as those which employ quadrature modulation require precise phase alignment.

For the analysis, let's begin with the "plant" of the PLL, which is the oscillator. The oscillator has a monotonic relationship between its control signal, a current or voltage, and the frequency of its output. Let us assume without loss of generality that we are dealing with a voltage controlled oscillator, or VCO. (some oscillators use current to control the output frequency, or CCO) The output can be near square, as in the case of a ring or relaxation oscillator, or nearly sinusoidal, as is the case with resonant tank based oscillators. Just as with the op-amp, the oscillator input-output parameter K_v (units of Hz/Volt) is difficult to precisely control. This is where feedback comes in. The integrator converts the frequency output into a phase output, which is then divided in the feedback path to feed into the difference block. The difference block is called a phase detector(PD), or phase-frequency detector(PFD), depending on its implementation. From this difference in phase we get a control voltage, which then passes through a filter. It will be clear that the filter is necessary in that it is the primary way by which the designer can affect the behavior of the PLL. The filtered output voltage is then fed back into the VCO, which closes the loop around the VCO. It is important to stress the difference between our PLL model

and the actual circuit implementation. Even though we have drawn an integrator and "phase divider" these are not actually present in the circuit. The reality is that the output of the VCO, which is a periodic waveform, is divided in frequency, which is the same as saying the output of the divider has a rising edge for every N input rising edges. This frequency divided wave is then fed into the PFD, which produces an output proportional to the difference in phases of the two inputs. Therefore it is more consistent with the circuit to place the integrator right before the PFD. The reason we put the integrator in the forward path is most PLL performance metrics are concerned with the θ_{out}/θ_{in} relationship, and placing the integrator before or after the divider makes no difference as integration is a linear operation.

Denoting the PFD gain as K_d and the filter transfer function as $H(s)$, we arrive at the input-output relation of

$$\frac{\theta_{out}}{\theta_{in}} = \frac{K_d K_v H(s)}{s + K_d K_v H(s)/N} \tag{1.2}$$

or alternatively,

$$\frac{\theta_{out}}{\theta_{in}} = \frac{NL(s)}{1 + L(s)} \tag{1.3}$$

where the loop gain, $L(s)$ is defined as

$$L(s) = \frac{K_d K_v H(s)}{Ns} \tag{1.4}$$

We will assume right now that the loop filter $H(s)$ is not a highpass filter; the reason why will obvious in a moment. We immediately notice several things. The system is at least a first order system, due to the integrator from the VCO. The PLL acts like a lowpass filter with respect to input phase changes. This is good news, because we do not want a frequency synthesizer to pass input noise on to the output. Furthermore, even when the loop filter $H(s) = A_0$ is a simple gain element, the system has only a constant phase error equal to

$$L(0) = \frac{N}{K_d K_v A_0} \tag{1.5}$$

and thus the frequency error is zero. By adding an integrator to the loop filter, the phase error can be driven to zero as well; again zero phase error is necessary for accurate quadrature clocks.

We also see the trade-off between settling speed and input-referred output noise. If we model $H(s)$ as a dominant pole, i.e.

$$H(s) = \frac{A_0}{\tau s + 1} \tag{1.6}$$

then

$$\frac{\theta_{out}}{\theta_{in}} = \frac{\frac{A_0 K_d K_v}{\tau}}{s^2 + \frac{1}{\tau}s + \frac{A_0 K_d K_v}{N\tau}} \tag{1.7}$$

Converting the denominator into a canonical form $s^2 + 2\zeta\omega_n s + \omega_n^2$ yields

$$\omega_n = \sqrt{\frac{A_0 K_d K_v}{N\tau}} \qquad \zeta = \sqrt{\frac{N}{4\tau A_0 K_d K_v}} \tag{1.8}$$

Recalling that settling time is inversely proportional to the damping ratio and natural frequency, we get

$$t_s \approx \frac{4}{\zeta\omega_n} = \frac{4}{1/2\tau} = 8\tau \tag{1.9}$$

So we see that settling time is inversely proportional to the bandwidth of the loop filter, which makes sense since forward path roots contribute as roots of the whole system. From this it seems simple to suggest that we make our loop bandwidth high, to enable fast phase locking. Unfortunately, as the next section will show doing so trades off spectral purity of the output signal.

1.3.2 Linear Noise Model

The contribution of noise from the circuit devices to the output noise can be analyzed just like any other feedback system, except that noise which appears on the PLL output is neither voltage noise nor current noise, but phase noise. More precisely, while the signal we will measure will be a voltage and thus noise appears as voltage

fluctuations in the output waveform, we are concerned with how those fluctuations affect the spectrum of the output. The concept of phase noise will be formally covered in chapter 3, however an intuitive introduction may be to think of it as deviations in frequency, where instantaneously one period of the output signal is different than the next, even though the average period length is constant. In the frequency domain, these deviations are centered around the output frequency, Nf_{ref} so the noise is FM modulating the output signal. Since we will be using this oscillator to modulate or demodulate data, these "skirts" around Nf_{ref} will distort our data.

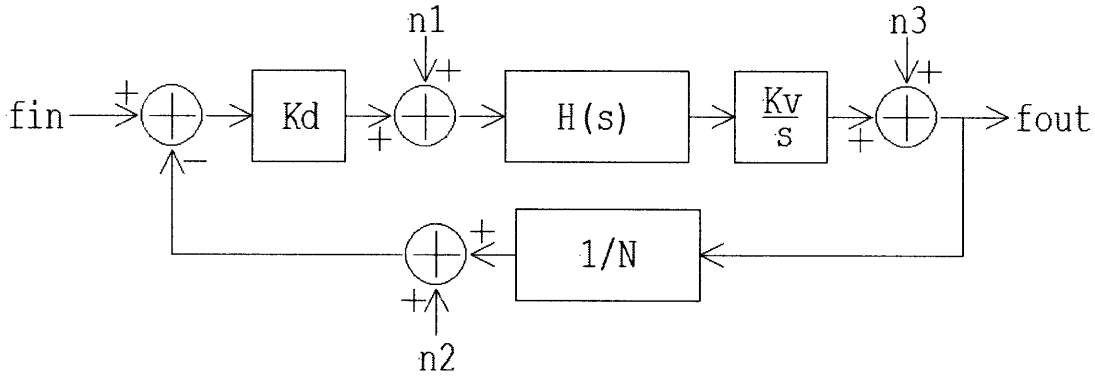

Figure 1-3: Block diagram with noise sources. We are neglecting the noise source from the loop filter, because we will design our loop filter such that its noise sources are negligible compared to the noise of the previous stages.

We revise the diagram as shown in figure 1-3 to include random noise sources injected at each node of the circuit. The transfer functions from the respective noise sources are as follows.

$$\frac{\theta_{out}}{n_1} = \frac{\frac{K_v H(s)}{s}}{1 + L(s)} \tag{1.10}$$

$$\frac{\theta_{out}}{n_2} = \frac{\frac{K_d K_v H(s)}{s}}{1 + L(s)} \tag{1.11}$$

$$\frac{\theta_{out}}{n_3} = \frac{1}{1 + L(s)} \tag{1.12}$$

Notice that noise from the output of the divider, source n_2, has the same transfer function as noise at the input. This noise is lowpass filtered by the feedback, since

$H(s)$ which is still assumed to be of the form 1.6 is in the forward path. The noise n_1 present at the output of the detector is even more attenuated since the factor of K_d/N in the feedback path is stronger than just $1/N$. However, the noise at the output of the VCO is highpass filtered. Thus our output noise performance in frequencies above our loop bandwidth is limited by how good we can make our oscillator. Intuitively, slow moving disturbances can be corrected by the loop's feedback, but noise outside the loop bandwidth is passed as at these high frequencies the loop is effectively open.

1.4 PLL Topologies

Let us consider the implementation of the PFD and loop filter. Until now, we have vaguely stated that the PFD produces an output proportional to the difference between input phases, and $H(s)$ lowpass filters this difference. We have not specified if the output of the PFD is a proportional amount of voltage, current, or chocolate syrup. There are three standard topologies of detectors.

1.4.1 Analog Multiplier Phase Detector

One implementation of a phase detector is to simply multiply the two inputs together. Using trigonometry,

$$\sin(\omega t)\sin(\omega t+\phi) = \frac{1}{2}\left(\sin\left(2\omega t+\phi+\frac{\pi}{2}\right)+\sin\left(\phi+\frac{\pi}{2}\right)\right) \tag{1.13}$$

If we assume $H(s)$ is designed to filter the second harmonic out, we have

$$\sin(\omega t)\sin(\omega t+\phi) \approx \frac{1}{2}\sin\left(\phi+\frac{\pi}{2}\right) \tag{1.14}$$

Thus this detector produces zero output when the oscillator is in quadrature with the reference. Unfortunately the proceeding result is also true with two inputs that are harmonic or sub-harmonic with respect to each other; the average output of the PD will still be zero. It is because the phase detection characteristic is modulo 2π and

can not discriminate between harmonically related frequencies that this is called a PD, not a PFD. Furthermore, unless there is an integrator in $H(s)$, this PD requires a finite frequency error to maintain lock. Since the PD has a voltage output, the integrator has to be implemented actively, which results in more noise in the loop filter. This becomes a problem in high performance PLL's.

1.4.2 XOR Phase Detector

Like the multiplicative PD, the XOR takes the digital product of the two input signals and produces a digital output, whose average is proportional to the phase difference between the two inputs. Again, this PD will exhibit finite phase error and harmonic locking. This PD is attractive because of its simplicity and linearity, so some designs overcome the harmonic locking problem by adding additional frequency detection circuitry that "coarse locks" the PLL until the PD can take over.[2, 3]

1.4.3 Charge Pump PLLs

The two main problems of the previous PDs were injection locking and finite input error. The analog multiplier PD has its own problems of biasing and DC offset error as well. The charge pump PLL solves both these problems in addition to presenting a topology that is very amenable to integration into a digital process. A basic PFD and loop filter in charge pump form are shown in figure 1-4.

The PFD is a digital circuit which creates two outputs, UP and $DOWN$. The time these two signal are on is related to the phase difference between the reference frequency phase and the local oscillator's phase, so the average duty cycle of current into the capacitor is given as $I_C = I_{UP}T_{UP} - I_{DN}T_{DOWN}$. In general, we make $I_{UP} = I_{DN} = I_D$ so $I_C = I_D\Delta\theta$. The advantage of using a current proportional to phase difference instead of a voltage is that by hanging a capacitor as shown, we get a free noiseless integrator. This drives steady state phase error to zero. The problem of phase/frequency detection is overcome by the PFD. Since it has two outputs which drive switches instead of analog filters as was the case with the multiplier based PD's,

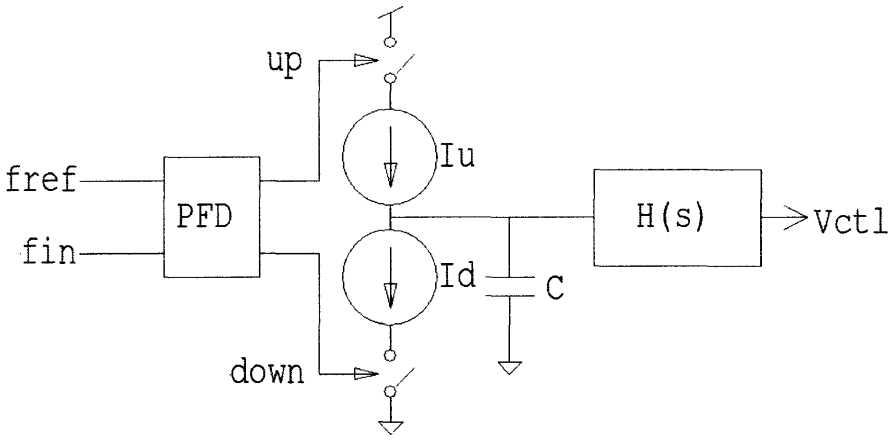

Figure 1-4: A simple charge pump based phase/frequency detector and loop filter. In general, the capacitor is part of $H(s)$ but it is drawn outside for explicitness.

the PFD has relaxed driving requirements. The PFD usually contains some state storage elements such as latches which help extend the detection range to prevent harmonic locking.

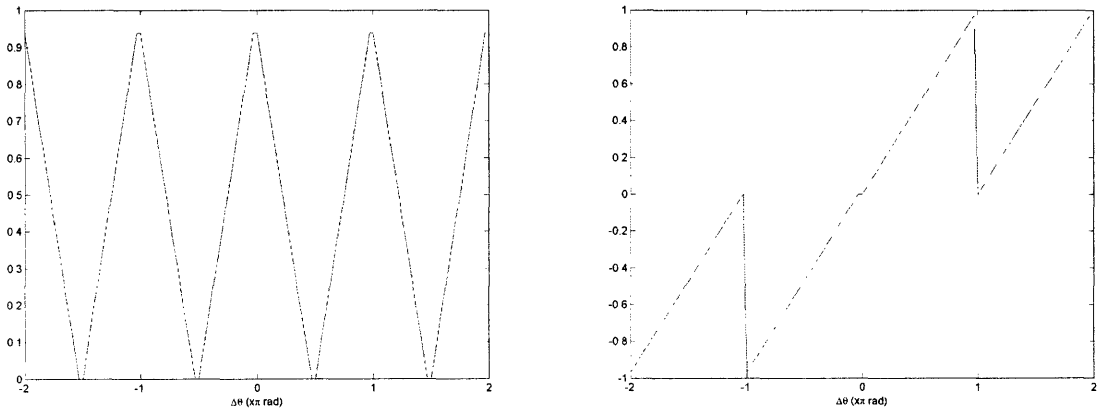

Figure 1-5: Comparison between typical PD and PFD characteristics. On the left, a PD has phase response modulo 2π, so it can not discriminate between harmonics of frequencies. The PFD on the right not only has extended range, but avoids the harmonic locking problem.

An additional advantage of the charge pump based PLL is it is a digital circuit. While the analog multiplier has bias sources, output dynamic range problems, and dc offset requirements, a charge pump's primary limitation is the voltage swing across

the capacitor must keep the source transistors in saturation to remain linear. The MOSFETs in the PFD and charge pump switches are all used as digital devices. For its circuit simplicity and the volume of preexisting documentation, we will use a charge pump topology for our PLL.

1.5 GSM900 Standards

The GSM900 standard is a comprehensive set of specifications by which all cellular communication systems in its category must adhere. This includes specifications for the whole path of the signal, from audio at one mouthpiece, to coding, to modulation, to transmission, receiving, demodulation and decoding at the other end. We will focus on a basic subset of the specifications regarding frequency synthesizers in this thesis. This covers oscillator operating range, transient performance, and phase noise specification. As some of these terms have not been rigorously defined, they will be introduced throughout the thesis when appropriate.

1.6 Thesis Organization

Chapters 2 and 3 deal with preliminary noise calculations, which provide insight into PLL design. Chapter 4 discusses the design of a ring-oscillator based voltage controlled oscillator, chapter 5 discusses a high speed programmable frequency divider, chapter 6 talks about the PFD and charge pump design, chapter 7 goes into the design of $H(s)$ and how it affects the loop performance, and we conclude with chapter 8 which shows complete system analysis. Performance results for each block are given in the final chapter rather than distributed throughout the thesis.

Bibliography

[1] Floyd M. Gardner. *Phaselock Techniques.* Wiley Interscience. John Wiley & Sons, New York, NY, second edition, 1979.

[2] Daniel H. Wolaver. *Phase-Locked Loop Circuit Design.* Prentice Hall Advanced Reference Series. Prentice Hall, Upper Saddle River, NJ, first edition, 1991.

[3] David Johns and Ken Martin. *Analog Integrated Circuit Design.* John Wiley & Sons, New York, NY, third edition, 2004.

Chapter 2

Device Noise

Intuitively, noise is the random, perturbation of a signal which degrades the original content. In all but a few specific applications, noise is an unwanted quantity and we design our circuit to reduce it. Precisely, noise is the disturbance of the output signal due to intrinsic properties of the devices within, properties which are also fundamental to the operation of the device, and thus unavoidable. The three types of such noise are thermal noise, shot noise, and flicker noise. Thermal noise and shot noise are both white noise sources, while flicker noise power is inversely proportional to frequency, leading some to call it $1/f$ noise.

Engineers are often careless and lump any undesired signal appearing at the output of a circuit as noise. This is counterproductive to understanding what causes noise, how we can deal with it, and what fundamentally limits our performance in a given circuit. For example, an audio amplifier which hums when no signal is applied is probably not humming due to device noise, rather some internal wiring is picking up the 60Hz stray magnetic field from a nearby power transformer. A sensitive analog section of a mixed signal chip which has a strong component of the digital section clock frequency in its output is not exhibiting internal noise. Poor shielding between sections is allowing coupling through the substrate across the chip, and thus not an intrinsic circuit effect. A simple minded approach is if the offending signal is well defined or narrowband in the spectra domain, it most likely isn't intrinsic device noise. In this section we will look at how device noise arises, and how it affects circuit

operation.

Fundamental to noise is the concept of random versus deterministic sources, and basic probability theory. Instead of including what is a rehash of every basic math text, the reader is urged to consult references [1] and [2].

2.1 Noise Types

2.1.1 Thermal Noise

Thermodynamics tells us that the energy of a system is proportional to its temperature. Thus, for an electron at non-zero temperature, it undergoes brownian motion with excitation proportional to its temperature. As the flow of electrons gives rise to currents, randomly moving electrons produce a randomly varying current. Because the thermal velocities of electrons exceed drift velocities,[1] the noise generated is independent of dc current. Indeed the noise power has been shown empirically to be $P_{NA}/\Delta f = kT$ where k is Boltzmann's constant, $1.3807 \cdot 10^{-23}$J/K.[3] We define noise power to be maximum power deliverable to a load. This was explained theoretically using the equipartition theorem by Nyquist at the same time.[4] From the maximum power transfer theorem, this is when the load impedance equals the source impedance, so if we model a noisy resistor by an ideal resistor in series with a noise voltage source[2] $\overline{v_n^2}$ or in parallel with a noise current source $\overline{i_n^2}$, our expressions for noise voltage and current become[2]

$$\frac{\overline{v_n^2}}{\Delta f} \cdot \frac{G}{4} = \frac{\overline{i_n^2}}{\Delta f} \cdot \frac{R}{4} = kT \tag{2.1}$$

Of course if this were true for all frequencies our noise power would be infinite, which is impossible. However it has been shown to be true up to 10^{13}Hz, which is three orders of magnitude above our device f_T, so for our purposes we can consider

[1] Although diffusion electron velocity exceeds drift velocity, under special conditions the drifting of electrons generates noise which can not be ignored, more on this later.

[2] When speaking of random zero-mean sources, it is more informative to speak in terms of mean-squared density, with units of V^2/Hz or A^2/Hz.

this white noise, with Gaussian amplitude distribution.[1] Note in general that the amplitude distribution (PDF) of the random variable is unrelated to the spectral distribution (PSD) of the random variable. Thermal noise has a flat PSD and a Gaussian PDF.[5]

2.1.2 Shot Noise

Although MOSFET noise has been traditionally considered to be dominated by thermal and flicker noise, it is useful to present an interesting result by Sarpeshkar, Delbrück, and Mead [6].

Shot noise is due to the quantization of current into the flow of discrete particles, electrons. Imagine electrons sitting in a charge gradient. Diffusion current will flow as a poisson process with arrival time $\lambda = \frac{I}{q}$.[7] At some time t_0 an electron travels from one terminal of a device to the other. The circuit surrounding the device sees an impulse response of current due to the single electron transfer, $h(t - T_0)$ with the property that $\int h(t)dt = q$. Current then is the superposition of all of these impulse responses, one for each arrival time

$$i(t) = \sum_{i}^{t_i<t} h(t - t_i) \tag{2.2}$$

Due to the probabilistic arrival nature of each electron, we can compute the spectral density of current noise, which is computed to be

$$\frac{\overline{i_n^2}}{\Delta f} = 2qI \tag{2.3}$$

Note that the PSD of shot noise is also white, which is true until the we approach the time constant of the transit time across the device, from one terminal to the other. In a diode, this is the transit time across the depletion region. In a sub-threshold MOSFET, this is the transit time from source to drain. Further investigation reveals that the PDF of shot noise is also Gaussian.[1]

The result presented in [6] suggests that thermal noise and shot noise are fun-

damentally the same noise generator under different names. While the underlying source of the noise has little consequence for the rest of this paper, unifying theories are nice.

2.1.3 Flicker Noise

Unlike thermal and shot noise, whose spectra is well defined by fundamental physical constants such as q and k, flicker noise is not as well understood. Several theories exist to the source of flicker noise. Flicker noise has been observed to be higher in devices which rely on surface interactions such as the Si-SiO_2 interface in a MOSFET than the diffusion conduction in a BJT.[2] This observation has lead to a theory called the McWhorter number fluctuation model,[8] which says flicker noise is generated by interface impurities which randomly trap and release charge, leading to fluctuations in current. The other well accepted model, the Hooge mobility fluctuation model, says variations in the surface mobility cause random scattering of carriers, and thus noise. Either way, the spectral density takes on the form[1]

$$\frac{\overline{i_n^2}}{\Delta f} = K_f \frac{I^a}{f^b} \tag{2.4}$$

Where K_f is a fitting constant which varies between device types, and even from device to device on the same substrate. The constant a varies between 0.5 to 2, and thus flicker noise is dependent on direct current flow. The constant b is around 1, which is why flicker noise is sometimes called 1/f noise. Note that unlike thermal and shot noises, which do not have free parameters in their models, 1/f noise has three, which makes analysis by hand and simulation difficult. Even if test data is available from previous fabrication runs from the same process, the great variance in the fitting parameters makes estimation crude at best.

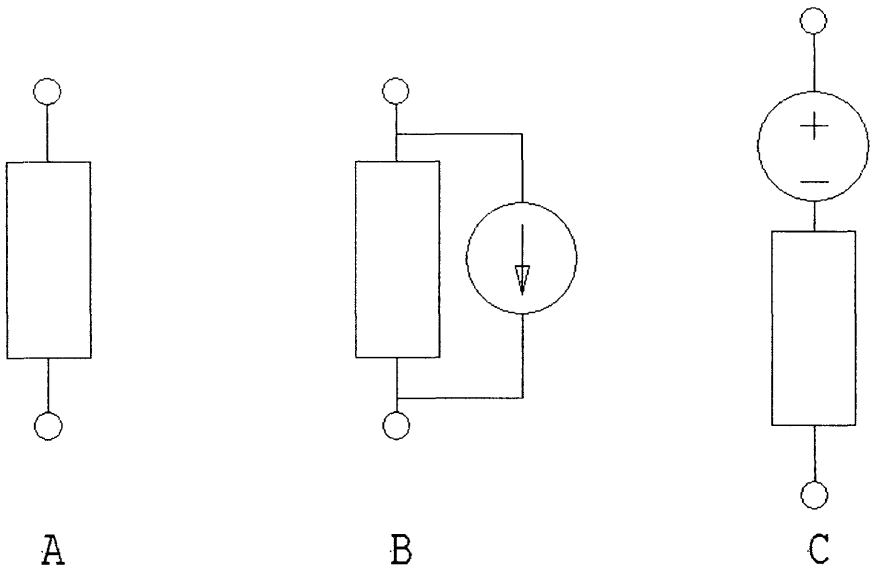

Figure 2-1: (A) shows a device which has internal noise. We model this in two ways, (B) shows an ideal noiseless device with a shunt current source, (C) is the same device but wit a series noise voltage source. Note that although the sources have polarity, since we only define noise by its RMS, the signals are non-polar.

2.2 Resistors

When modeling a device's noise for circuit analysis, it is standard to replace the device with an "ideal" noiseless device with a either a voltage noise source in series with or a current noise source in parallel with its terminals, as shown in figure 2-1.

The main source of noise in resistors is thermal noise, so the equation for the noise source is given by 2.1. For resistors with DC currents flowing through them, 1/f noise begins to appear as well, in varying degrees depending heavily upon resistor construction. In particular, carbon composition resistors show strong 1/f noise.[1] This 1/f noise is called "excess noise" and is an additive term to the existing thermal noise spectra.[2] However, since the integrated resistors in our CMOS process do not show strong 1/f noise, nor do we have DC currents flowing through them, we will neglect the excess noise of resistors.

2.3 Capacitors and Inductors

Ideal capacitors and inductors generate no noise of their own, a result of the fluctuation-dissipation theorem.[9] Loosely, a system is coupled to the external world by dissipating energy into thermal energy; hence, if a system dissipates no energy, there can be no thermal energy transfer, and thus no noise. Since ideal capacitors and inductors are lossless, they have no noise. However, no real device has ideal characteristics. Capacitors have both series resistances due to resistive contacts as well at leakage paths across the two plates.

In integrated capacitors, the insulator is SiO_2, which is an excellent insulator. The plates of the integrated capacitor are either metal layers, polysilicon, or sections of heavily doped substrate. Metal is most conductive, so it has very low series resistance. Unfortunately, the metal-metal layer spacing is highest, so this capacitance is lowest per unit area. At the other extreme, capacitors formed with poly-substrate plates are simply MOSFETs with the drain and source tied together as a terminal. The unit capacitance is the gate capacitance, which is very large due to the thinness of the gate oxide, 78Å in a 0.35μm process. Unfortunately, one of the plates is polysilicon, which is nowhere nearly as conductive as metal, and the other is the inverted MOSFET channel, also far from ideal.

In integrated inductors, flat spirals are made with a metal layer. Given some processes' lack of any angle other than right, and the heavy parsitic capacitances present in such a structure, the Q of integrated inductors are deplorably low. This aside, the metal wire which forms the inductor has a resistance, and this resistance in series with the ideal inductance has noise. Conveniently, a ring oscillator PLL avoids using inductors completely.

For both capacitors and inductors, we must find its parasitic dissipation elements, and from these calculate their contributions to noise. For tank based integrated oscillators, not only do the parasitic elements kill the tank Q, they add noise!

2.4 MOSFET

The MOSFET is dominated by two noise sources, thermal and flicker. Thermal noise arises from the finite conductance of the channel in which the mobile carriers flow. Since there is a vertical electric field in the channel due to the gate-body voltage differential, carriers are attracted to the Si-SiO_2 interface, where they interact with impurities, thus creating 1/f noise as outlined in section 2.1.3. Several models and expressions for these noise sources exist in the literature. A sampling will be presented, and in subsequent calculations, the worst case analysis for the given operating point will be used.

2.4.1 Thermal Noise Above Threshold

Saturation

Among the standard texts which deal with noise, the case of above threshold saturation is given the most coverage, since it is the most common bias condition for MOSFETs in analog circuits. The standard equation given in many analog design textbooks for the drain current noise in this case is given in [1] as

$$\frac{\overline{i_n^2}}{\Delta f} = \frac{8}{3} kT g_m \tag{2.5}$$

where

$$g_m = \sqrt{2\mu_n C_{ox} \frac{W}{L} I_D} \tag{2.6}$$

is the gate transconductance. For a sanity check, we can convert equation 2.1 into a similar form, $\frac{\overline{i_n^2}}{\Delta f} = 4kTG$, then compare it with to equation 2.5 to see that our effective drain-source conductance is $\frac{2}{3} g_m$, which is odd, because the gate-source transconductance is independent of drain-source transconductance. Before delving further, we need to first define long versus short channel devices. Long channel devices are considered MOSFETs with channel lengths 1μm and above, while short channels constitute anything below. As our channel lengths shorten, second order physical effects come

into play, such as channel length modulation, velocity saturation, and drain-induced bias lowering.

The model used above was developed before the existence of short channel devices, and unfortunately several popular texts do not differentiate between the two. The expression 2.5 is derived from a model found in [9] which is given below:

$$\frac{\overline{i_n^2}}{\Delta f} = 4kT\gamma g_{d0} \tag{2.7}$$

Where $\gamma = \frac{2}{3}$ when the model was originally developed. The zero voltage drain-source transconductance g_{d0} is defined as

$$g_{d0} = \mu_n C_{ox}\frac{W}{L}(V_{GS} - V_{Tn} - V_{DS})_{V_{DS}=0} = \mu_n C_{ox}\frac{W}{L}(V_{GS} - V_{Tn}) \tag{2.8}$$

Now we can see that $g_{d0} = g_m$, which can also be explained intuitively by thinking of how the carriers are distributed in the channel of a MOSFET for zero V_{DS}. The drain current due to drift is proportional to $-\mu_n QE$, where Q is the carrier density in the channel, and E is the drift field due to V_{DS}. As V_{DS} increases, it loses control over I_D relative to V_{GS} because incremental increases in E have less effect when E is already large. However when V_{DS} is zero, changes in Q and changes in E have equal weight, hence g_M=g_{d0}.

As short channel devices became available, the model for noise was modified to allow γ to change. As the device length shrinks, the electric field increases to the point that further increases do not increase the carrier velocity. This velocity saturation leads to hot carrier effects.[10] Furthermore, channel length modulation becomes more noticeable, causing drain voltage to control the drain source noise.

Triode

Some texts[11] use equation 2.5 to model both saturation and triode. Other texts[1] take the noise model given in equation 2.7 and extend it to include the triode region of operation by substituting g_{d0} by g_{ds} whenever in triode.

Using equation 2.5 to model triode noise says noise decreases with decreasing V_{DS}, because in triode the expression for g_m becomes $\mu_n C_{ox} \frac{W}{L} V_{DS}$, linearly dependent on V_{DS}. Note that although we generally don't speak of g_m for a transistor in triode, this expression is perfectly valid since we define $g_m = \frac{dI_D}{dV_{GS}}$. However, intuitively noise should not decrease with decreasing V_{DS}; if anything, decreasing V_{DS} should increase the amount of charge in the channel, thereby increasing the conductance and noise generation. For an elegant supporting argument, consult [6]. Although it deals with the subthreshold case, the derivation is general enough to apply here.

With the above reasoning, the alternative triode noise formula is

$$\frac{\overline{i_n^2}}{\Delta f} = 4kT\gamma g_{ds} \tag{2.9}$$

for g_{ds} set by V_{DS}. The problem with this is that as we approach saturation, $g_{ds} = \mu_n C_{ox} \frac{W}{L} (V_{GS} - V_{Tn} - V_{DS})$ approaches zero, and once we are in saturation, the chosen saturation noise equivalent conductance is once again g_{d0}, which is non-zero. So we have a discontinuity in the noise value through saturation, which seems somewhat suspicious.

A More Accurate Model

A more accurate model for thermal noise for both the triode and saturation regions was developed in [12], to account for both short channel effects and valid in both saturation and triode. The results in [12] also show that while equation 2.5 is invalid in the triode region, 2.7 is accurate in saturation and triode for long channel devices, but does not account for the increase in noise as V_{DS} increases in short channel devices. The reason for this is increases in V_{DS} in short channel devices reduces the channel length, increasing the channel conductance accordingly. The complete model for thermal noise in triode is

$$\frac{\overline{i_n^2}}{\Delta f} = 4kT\mu_{eff}^2 C_{ox}^2 \frac{W^2}{L^2 I_D} \left[(V_{GS} - V_t)^2 V_{DS} - \gamma (V_{GS} - V_t) V_{DS}^2 + \frac{\gamma^2}{3} V_{DS}^3 \right] \tag{2.10}$$

$$- 4kT \frac{W C_{ox} \mu_{eff}}{L^2 E_c} \cdot \left[(V_{GS} - V_t) V_{DS} - \frac{\gamma}{2} V_{DS}^2 \right] \tag{2.11}$$

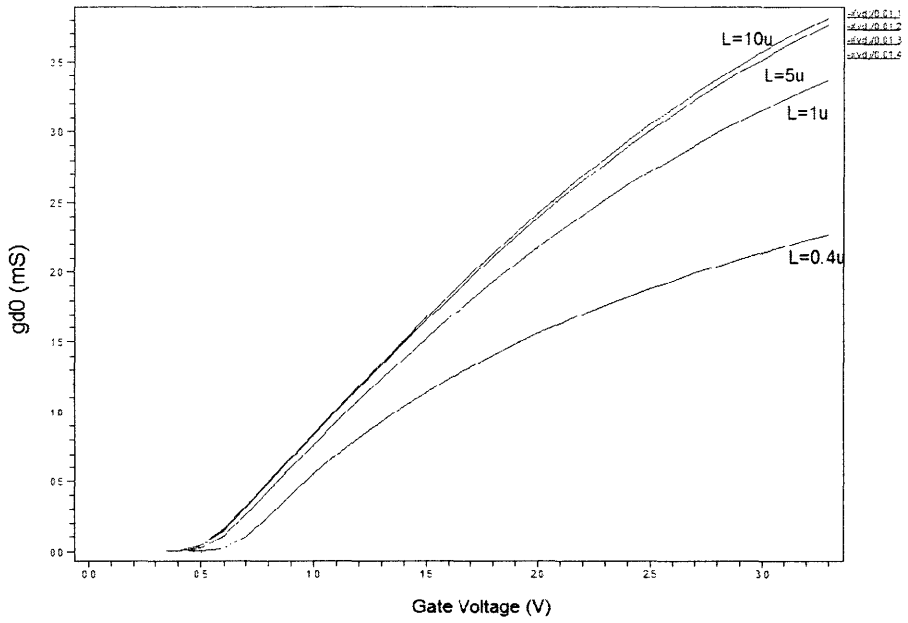

Figure 2-2: Measurements of zero voltage drain conductance versus gate voltage are plotted for several gate lengths, while scaling up width proportionally. As the channel gets longer, g_{d0} loses dependence on W/L.

To account for saturation, replace L with L_{eff}, the effective channel length due to channel length modulation, and V_{DS} with V_{Dsat}, since in saturation the channel potential is clamped to V_{Dsat} regardless of V_{DS}. This model has similar terms to the model given in 2.7, the difference is that the model given in 2.10 calculates the effective conductance of the channel by integrating the channel charge, which is more accurate than approximating with just g_{d0}.

The model we will use for our devices is derived from 2.7, since it accounts reasonably well for current noise in triode and saturation.[12] By using g_{d0} instead of g_m, we avoid issues of g_m lowering in short channel devices. Figure 2-2 is a plot of g_{d0} for varying gate lengths. Note that as we leave the short channel regime, the conductance normalizes to be independent of W/L, as it should.

Knowing this, we use MATLAB to generate a polynomial fit to the simulated data for g_{d0}, and produce the function

$$\frac{\overline{i_n^2}}{\Delta f} = 4kT\gamma g_{d0}(V_{GS}) \tag{2.12}$$

With this, we can use HSPICE plots for V_{GS} values of the transistors to get transistor noise. The choice of γ for the 0.4μ minimum size devices is estimated from [13, Figure 2] to be $\gamma = 1$ in saturation, a factor of 1.5 times greater than long channel $\gamma = 2/3$. We will correct this later with simulations, but it will be sufficient for hand analysis.

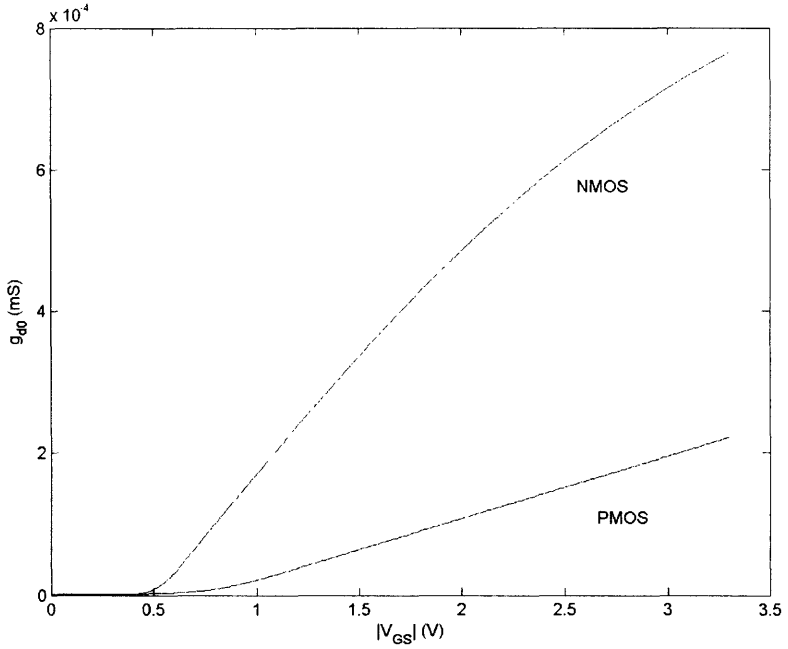

Figure 2-3: Using a 20^{th} order polynomial in MATLAB for the g_{d0} data. Both conductance values have been normalized to W/L=1. Note the conductance and hence noise of a P-MOSFET of the same size is lower, as expected. Also note that the NMOS has a more nonlinear slope, due to increased mobility saturation effects in N-type devices.

2.4.2 Flicker Noise Above Threshold

We know the equation for flicker noise given in 2.4, all we need to do is determine the constants K_f, a, and b. Unfortunately, these constants vary so greatly, the best we can do is make safe approximations. In general, as channel lengths shorten, the noise shows an increasing dependence on V_{GS} for a constant I_D. This is due to g_m nonlinearities as channel lengths shorten. Furthermore, shorter channel lengths

increase the drain noise.[8] An investigation into 0.35μm devices show that noise in n-MOSFETs is an order of magnitude higher than in p-MOSFETs, for both flicker an thermal noises. This is due to the buried channel of the p-MOSFETs.[14] Values of K_f range in the literature between 10^{-28}A·F to 10^{-24}A·F.[2, 8] We will use $a = b = 1$, and $K_f = 10^{-24}$A·F for conservative calculations.

Bibliography

[1] Paul R. Gray, Paul J. Hurst, Stephen H. Lewis, and Robert G. Meyer. *Analysis and Design of Analog Integrated Circuits.* John Wiley & Sons, New York, NY, fourth edition, 2002.

[2] Thomas H. Lee. *The Design of CMOS Radio-Frequency Integrated Circuits.* Cambridge University Press, New York, NY, second edition, 2004.

[3] John B. Johnson. Thermal agitiation of electric charge in conductors. *Physical Review*, 32:97–109, July 1928.

[4] Harry Nyquist. Thermal agitiation of electric charge in conductors. *Physical Review*, 32:110–113, July 1928.

[5] Behzad Razavi. *RF Microelectronics.* Prentice Hall Communications Engineering and Emerging Technology Series. Prentice Hall, Upper Saddle River, NJ, first edition, 1998.

[6] Rahul Sarpeshkar, Tobias Delbrück, and Carver A. Mead. White noise in mos transistors and resistors. *IEEE Journal of Circuits and Devices*, 9(6):23–29, November 1993.

[7] Roger T. Howe and Charles G. Sodini. *Microelectronics.* Prentice Hall, Upper Saddle River, NJ, second edition, 1996.

[8] Dingming Xie, Mengzhang Cheng, and Leonard Forbes. Spice models for flicker noise in n-mosfets from subthreshold to strong inversion. *IEEE Transactions on*

Computer-Aided Design of Integrated Circuits and Systems, 19(11):1293–1303, November 2000.

[9] Aldert Van Der Ziel. *Noise in Solid State Devices and Circuits.* John Wiley & Sons, New York, NY, first edition, 1986.

[10] Christian C. Enz, François Krummenacher, and Eric A. Vittoz. An analytical mos transistor model valid in all regions of operation and dedicated to low-voltage and low-current applications. *Analog Integrated Circuits and Signal Processing*, 8:83–114, August 1995.

[11] David Johns and Ken Martin. *Analog Integrated Circuit Design.* John Wiley & Sons, New York, NY, third edition, 2004.

[12] Bing Wang, James R. Hellums, and Charles G. Sodini. Mosfet thermal noise modeling for analog integrated circuits. *IEEE Journal of Solid State Circuits*, 29(7):833–835, July 1994.

[13] Jung-Suk Goo, Chang-Hoon Chi, Antonio Abramo, Jae-Gyung Ahn, Zhiping Yu, Thomas H. Lee, and Robert W. Dutton. Physical origin of the excess thermal noise in short channel mosfets. *IEEE Journal of Solid-State Circuits*, 22(2):101–103, February 2001.

[14] P. Sakalas, H. Zirath, A. Litwin, and M. Schröter. On-wafer low frequency noise investigation of the 0.35μm n and p type mosfets, dependence upon the gate geometry. In *Talk at International Conference on Noise in Physical Systems, Gainesville, FL*, 2001.

Chapter 3

PLL Phase Noise

In our ideal PLL, the frequency of the output is multiplied by the feedback ratio of the divider, producing an output of frequency $N \cdot f_{ref}$. (section 1.3.1) From feedback theory, we know how perturbations at various points within the loop affect the output. However our problem with phase noise is now voltage and current noise within the circuit degrade the spectral purity of the output. If we write the output of our PLL as

$$LO = A \cdot cos(2\pi f_0 t + \phi(t)) \tag{3.1}$$

then f_0 is the carrier frequency, and ϕ_0 represents the phase noise, or random disturbances in phase. In the ideal case, $\phi(t)$ is zero, our output signal is a pair of impulses in the spectral domain. Now suppose $\phi(t)$ is nonzero. If we use the cosine addition formula, we get

$$LO = A \cdot cos(\phi(t))cos(2\pi f_0 t) - A \cdot sin(\phi(t))sin(2\pi f_0 t) \tag{3.2}$$

Now we will assume that $\phi(t)$ is much smaller than $1/2\pi f_0$, which has to be true if our oscillator is worth anything. Thus we can approximate our output as

$$LO = A \cdot cos(2\pi f_0 t) - A \cdot \phi(t)sin(2\pi f_0 t) \tag{3.3}$$

From basic signal processing, this shows that the spectrum of the phase noise,

$\phi(t)$, is modulated by the carrier to appear centered around f_0. This leads to a problem called reciprocal mixing. In modern frequency division architectures, the frequency band is divided into many closely spaced channels to maximize bandwidth usage. Noise in the local oscillator will heterodyne with a channel adjacent to the channel we wish to receive, thus down-converting a portion of the unwanted channel along with the desired channel. Since the RF signal power can be very small, any such interference degrades the SNR of the down-converted signal. For further reading on receiver topologies, [1, 2] are useful references.

3.1 Phase Noise Definition

Since wireless applications usually also have power requirements, when we speak of phase noise we usually reference it to the power of the carrier, which is analogous to SNR. The units of phase noise are then dBc/Hz, which is defined as

$$\mathcal{L}(\Delta f) = 10 \cdot \log\left(\frac{\mathcal{P}_{sideband}(f_0 + \Delta f, 1\text{Hz})}{\mathcal{P}_{carrier}}\right) \tag{3.4}$$

$\mathcal{P}_{sideband}(f_0 + \Delta f, 1\text{Hz})$ is defined as the power contained in a 1 Hz band centered at Δf off f_0. This is also called the power spectral density, and the measurement bandwidth of 1 Hz is assumed. From equation 3.3 we can plug in to 3.4 to get the following expression of phase noise:

$$\mathcal{L}(\Delta f) = 10 \cdot \log\left(\frac{1}{2} \cdot \frac{\frac{A^2}{2}\mathcal{P}_\phi}{\frac{A^2}{2}}\right) = 10 \cdot \log\left(\frac{\mathcal{P}_\phi}{2}\right) \tag{3.5}$$

There are two points to make here. The first to note is the addition of a 1/2 term after the first equality. This is because up until now, we have not distinguished between phase noise, $\phi(t)$, and possible amplitude distortion $A(t)$ contributing to noise. If we only want to look at the effect of phase noise, the equipartition theorem of thermodynamics states that the noise energy will divide evenly between amplitude and

phase distortion.[1][3, 4, 5] The second point is by removing the temporal dependence of $\mathcal{P}_\phi$ we have implied that $\phi(t)$ is periodic, or at least stationary. This will be clear later, when we see that $\phi(t)$ is a combination of periodic ripple from the VCO control voltage and random device noise.

3.2 Jitter

Phase noise is the spectral description of the purity of the output signal, whereas jitter is the temporal description. Suppose we are counting the zero crossings of a signal such as 3.1, with $\phi(t)$ zero. Since phase is changing linearly with time, each zero crossing is timed exactly $1/2\pi f_0$ after the last. Now suppose there is some phase noise. A positive $\phi(t)$ will cause a zero crossing to occur too soon, while a negative $\phi(t)$ will cause the crossing to arrive late. Digital designers are not concerned with how the oscillator output will mix with an adjacent frequency, they require a minimum clock period in which their gates must complete switching. Thus digital designers use the quantities σ_{RMS} and $\sigma_{\text{pk-pk}}$ as the RMS and peak-to-peak values of period after subtracting out the mean period. The idea of peak-to-peak may not make strict Gaussian sense, considering any peak-to-peak value is theoretically possible. Therefore designers choose a BER, (bit error rate) threshold which gives the probability that the jitter will be greater than $\sigma_{\text{pk-pk}}$.

Since $\mathcal{L}(\Delta f)$ is a continuum of data, it generally contains more information than the σ parameters, unless we make several assumptions on the shape of either $\mathcal{L}$ or the probabilistic distribution of $\phi(t)$.[6]

3.3 Oscillator Phase Noise Theories

In the previous section we saw how to formulate the concept of phase noise. From the transfer functions given in equation 1.10, we see that the linear analysis holds for sources $\overline{i_{n1}^2}$ and $\overline{i_{n2}^2}$, but not $\overline{i_{n3}^2}$. The first two sources modulate the VCO control

[1]We will see that amplitude noise is more controllable, and thus we focus on phase noise.

voltage, and from this modulation we can determine an expression for $\phi(t)$. However, the effect of $\overline{i_{n3}^2}$ is not clear, because it acts upon devices and nodes within the oscillator, and our linear model does not have the required "modeling resolution" to explain how the VCO output is affected by its internal noise.

3.3.1 Linear Time-Invariant Models

Tuned Oscillators

A simple model for tuned oscillator phase noise was developed by David Leeson in 1966.[7] The idea is that noise from the active devices is filtered by the tuned tank, and thus shapes the spectral output. Consider a parallel LC tank with some lossy parallel resistance. The active negative resistance restores energy to the tank, such that the negative resistance perfectly cancels the tank's loss. Thus, the noise generated by the tank sees a perfect LC tank. In this case, we would like to know the impedance of the tank for small offsets of frequency from resonance:

$$Z(\omega_0 + \Delta\omega) = \frac{jL(\omega_0 + \Delta\omega)}{1 - LC(\omega_0^2 + 2\omega_0\Delta\omega + \Delta\omega^2)} \tag{3.6}$$

Since $\omega_0 = \frac{1}{\sqrt{LC}}$,

$$Z(\omega_0 + \Delta\omega) = \frac{jL(\omega_0 + \Delta\omega)}{-2LC\omega_0\Delta\omega - LC\Delta\omega^2} \approx \frac{jL\omega_0}{-2LC\omega_0\Delta\omega} = \frac{-j}{2C\Delta\omega} \tag{3.7}$$

Since the tank in reality has finite Q, we will do the substitution $Q = \omega_0 RC$ to arrive at

$$Z(\omega_0 + \Delta\omega) = \frac{-j\omega_0 R}{2Q\Delta\omega} \tag{3.8}$$

Now that we know the tank impedance for small deviations off the carrier, we can compute the total voltage noise as

$$\frac{\overline{v_n^2}}{\Delta\omega} = \frac{\overline{i_n^2}}{\Delta\omega} \cdot |Z(j\omega)|^2 = \frac{\overline{i_n^2}}{\Delta\omega} \cdot R^2 \frac{\omega_0^2}{4Q^2\Delta\omega^2} \tag{3.9}$$

It is common practice to write current noise as the noise of one resistor times a

multiplicative factor F, where F is the equivalent number of devices in a circuit. Since we already have $G = 1/R$ the tank parasitic conductance, let's denote the noise from the tank resistance in addition to any noise from the energy restoring circuit as

$$\frac{\overline{i_n^2}}{\Delta f} = F \cdot 4kTG \tag{3.10}$$

We have can use $\frac{\overline{i_n^2}}{\Delta f}$ since the spectrum of noise is flat. Substituting this back into our expression for voltage noise gets us

$$\frac{\overline{v_n^2}}{\Delta\omega} = F \cdot 4kTG \cdot R^2 \frac{\omega_0^2}{4Q^2\Delta\omega^2} = F \cdot 4kTR \frac{\omega_0^2}{4Q^2\Delta\omega^2} \tag{3.11}$$

Now we cite equation 3.5 to get the following expression for phase noise

$$\mathcal{L}(\Delta\omega) = 10 \cdot \log\left(\frac{F \cdot 2kTR\frac{\omega_0^2}{4Q^2\Delta\omega^2}}{\frac{1}{2}V_{pk}^2}\right) = 10 \cdot \log\left(\frac{F \cdot 2kT\frac{\omega_0^2}{4Q^2\Delta\omega^2}}{\frac{V_{pk}^2}{2R}}\right) \tag{3.12}$$

The denominator of the logarithm is nothing but the average power dissipated in the tank. Also, we need to account for the fact that although the tank shapes the output spectrum of noise, there is a limit to the conductivity of the tank for large $\Delta\omega$, so there is a noise floor. We account for this by adding 1 to the noise shaping function, imposing a lower bound on the magnitude of the impedance.

$$\mathcal{L}(\Delta\omega) = 10 \cdot \log\left[\frac{F \cdot 2kT}{\mathcal{P}_{diss}}\left(\left[\frac{\omega_0}{2Q\Delta\omega}\right]^2 + 1\right)\right] \tag{3.13}$$

Note that increasing the tank Q causes $\mathcal{P}_{diss}$ to decrease inversely. However, the phase noise is only linearly dependent on $\mathcal{P}_{diss}$, and quadratically dependent on Q. Thus increasing tank Q reduces phase noise, as expected.

We have derived Leeson's model for white noise sources. Further investigation reveals an additional multiplicative factor is necessary to account for the $1/f$ corner.

Leeson's full phase noise model is

$$\mathcal{L}(\Delta\omega) = 10 \cdot \log \left[\frac{F \cdot 2kT}{\mathcal{P}_{diss}} \left(\left[\frac{\omega_0}{2Q\Delta\omega} \right]^2 + 1 \right) \left(\frac{\Delta_{1/f^3}}{|\Delta\omega|} + 1 \right) \right] \tag{3.14}$$

The problem with this model is it is difficult to compute the fitting parameter, F. (F stands for "fudge") Also, because this model is linear, it does not explain modulation effects such as why ripple in the VCO control voltage is modulated into carrier sidebands.

Ring Oscillators

Since the above theory of phase noise is for tuned oscillators, the assumption that only noise around the carrier frequency contributes seems reasonable, even if it is not the case. In a ring oscillator however, because there is no bandpass tank, we can not discard noise at other frequencies.[8]

The noise at frequencies around the carrier is filtered similarly to the tuned oscillator, but not by a tuned tank, rather by the oscillator's closed loop transfer function. This filtered noise superimposes upon the output spectrum, and is called additive noise.

To deal with the noise at low frequencies, it is instructive to look at each stage separately as a linear amplifier. When analyzing the noise of a linear circuit such as an op-amp, it is common practice to reflect current noise at each node of the circuit to an equivalent gate noise voltage. This is done by dividing the current noise by the square (because we are dealing with mean-squared quantities) of the transconductance from the input to the node in question. All these gate voltage noise sources are summed up to get an equivalent voltage noise source. We can then express the control voltage into our VCO as

$$v_{CTL} = V_{CTL} + \frac{\overline{v_n^2}}{\Delta\omega} \tag{3.15}$$

We know the output of our oscillator is periodic with frequency K_v times the control

voltage, so

$$LO = A \cdot \cos(K_v v_{CTL} \cdot t) \tag{3.16}$$

expanding this out to a noise term and a carrier $\omega_0 = K_v V_{CTL}$,

$$LO = A \cdot \cos\left(\omega_0 t + K_v \frac{\overline{v_n^2}}{\Delta\omega} t\right) \tag{3.17}$$

The proceeding heuristic argument is rigorously carried out in [8] to yield the final result,

$$\mathcal{L}(\Delta\omega) = 10 \cdot \log\left[\frac{1}{4} I_m^2 \left(\frac{K_v}{\Delta\omega}\right)^2\right] \tag{3.18}$$

Razavi goes on to characterize the effect of noise at frequencies around multiples of ω_0 in a separate case using Taylor approximations. While this is proven to be an accurate theory by simulation results, it is somewhat cumbersome to break wideband noise into three sections to analyze separately. The following is a unified phase noise theory for all oscillators.

3.3.2 Linear Time-Varying Model

The Leeson model for phase noise was developed for tuned tank oscillators, and not as useful for ring oscillators. Razavi's phase noise theory was conceived with ring oscillators in mind, but it analyzes noise with a series of cases. Now we discuss Thomas Lee and Ali Hajimiri's unified linear time-varying (LTV) phase noise model. The advantage in this model is it does not depend on any oscillator topology, and presents a normalized metric, the ISF, with which one can compare relative performance between different oscillators. Furthermore, the LTV theory gives intuitive insight into circuit methods to improve the phase noise of oscillators. Finally, it explains why ring oscillators have much poorer phase noise performance than tuned oscillators. The following derivation for phase noise follows the theory presented in [2], [6], and [9] with some minor modifications.

The LTV model relaxes time-invariance enough to account for noise phase mod-

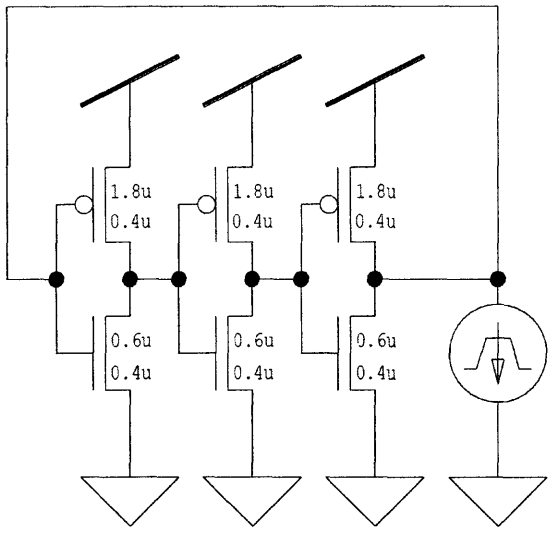

Figure 3-1: A ring oscillator with an impulse of current injected into one node.

ulation, yet remains linear so that we may still use superposition. The time variance of the model is not troublesome at all, for in the case of an oscillator in steady-state oscillation, all features of the oscillator are periodic with the output frequency. Informally, the model determines the transfer function of noise at a node to phase disturbances at the output through a brute force measurement method. For a fixed injected disturbance current, the amount of phase disturbance varies with time within a period, hence the model's time variance. If the ratio of phase disturbance to injected current is calculated for a full period, the phase disturbance impulse response is calculated. Armed with this impulse response, and leveraging the assumed linearity of this model, we can then calculate the phase disturbance to an arbitrary injected current, in particular current noise.

Starting with an arbitrary oscillator, if we inject an impulse of current onto a node, this disturbs the node voltage by $\Delta V = C_{node}/q_{test}$ where q_{test} is the area of the current impulse, and C_{node} is the capacitance of the node at time of injection. This voltage disturbance causes the oscillator output waveform to change momentarily, however the oscillator returns to steady state oscillation provided the disturbing current is sufficiently small. The voltage disturbance however leads to a phase shift of the output. Figure 3-2 is the resulting phase shift for a 25fC impulse of current injected

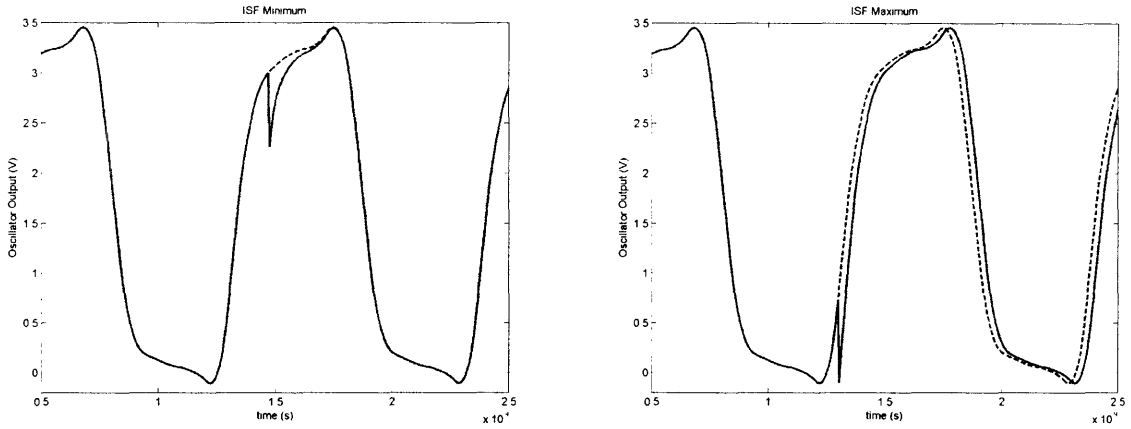

Figure 3-2: The same ring oscillator simulated with current injected at different times into one node. The dashed line represents the undisturbed oscillator output.

into one node of the 3 stage minimum size ring oscillator shown in figure 3-1 at two different times. On the left, the current has no effect, and the phase is not disturbed. On the right, the impulse leads to a constant phase offset.

If we repeat the measurement of phase disturbance to current injection many times over one oscillator period, we can obtain $h_\phi(t,\tau)$ which is the time dependent impulse response of phase with respect to current. Note that $h_\phi(t,\tau)$ is periodic with respect to τ. We can write $\phi(t) = h_\phi(t,\tau) * i(\tau)$. Referring back to figure 3-2, we see that $\phi(t)$ is a step response delayed by time τ, with magnitude equal to the resulting phase shift from the current injection. Therefore $h_\phi(t,\tau)$ is also a scaled step response delayed by τ.

Lee and Hajimiri go on to define the ISF $\Gamma(\tau)$, which is $h_\phi(t,\tau)$ normalized by the charge swing at the node. The reasoning is that if two oscillators have identical $h_\phi(t,\tau)$ responses but one has twice the node capacitance as the other, the oscillator with lower node capacitance experiences twice the voltage disturbance for a given charge injection. If that oscillator still manages to have the same $h_\phi(t,\tau)$ then simply doubling the node capacitance should halve its $h_\phi(t,\tau)$. Thus the ISF is an intrinsic measure of oscillator noise properties. We define

$$h_\phi(t,\tau) = \frac{\Gamma(\tau)}{q_{max}} u(t-\tau) = \frac{\Gamma(\tau)}{C_{node} V_{max}} u(t-\tau) \tag{3.19}$$

One problem with the ISF is that the node capacitance C_{node} varies within one period of the oscillator output. Let us define

$$\Lambda(\tau) = h_\phi(\infty, \tau) \tag{3.20}$$

Remember that $h_\phi(\infty, \tau)$ is simply the phase disturbance for a current injected at τ, since $h_\phi(t, \tau)$ is just a τ dependent step function.

Before looking at phase noise for a general $\overline{i_n^2}/\Delta\omega$ white noise source, let's consider what happens when we inject a sinusoidal current $I_n = A_n \cos((n\omega_0 + \Delta\omega)t)$ into a node. We assume that $\Delta\omega \ll \omega_0$. The equation for phase gives us

$$\begin{aligned}
\phi(t) &= h_\phi(t, \tau) * I_n(t) && (3.21)\\
&= \int_{-\infty}^{\infty} \Lambda(\tau) u(t-\tau) A_n \cos((n\omega_0 + \Delta\omega)\tau) d\tau && (3.22)\\
&= \int_{-\infty}^{\infty} (u(t-\tau)) \cdot (A_n \Lambda(\tau) \cos(\omega_n \tau))\, d\tau && (3.23)\\
&= u(t) * \Lambda(t) I_n(t) && (3.24)
\end{aligned}$$

We have transformed our time-varying model into a standard linear time-invariant system. We can show that $\Lambda(t)I_n(t)$ has a low frequency component $\Delta\omega$. To see this, consider the Fourier series expansion of $\Lambda(t) = \frac{c_0}{2} + \sum_{m=1}^{\infty} c_m \cos(m\omega_0 t + \theta_m)$. Because we are dealing with noise which has random phase, we will assume that θ_m is such that the product of a Fourier component and $I_n(t)$ is maximal. For example in this case $\theta_m = 0$ because our input is a cosine. If we expand out the product we get

$$\begin{aligned}
\Lambda(t) I_n(t) &= \left(\frac{c_0}{2} + \sum_{m=1}^{\infty} c_m \cos(m\omega_0 t) \right) A_n \cos((n\omega_0 + \Delta\omega)t) && (3.25)\\
&= A_n \left(\frac{c_0}{2} \cos((n\omega_0 + \Delta\omega)t) \right. && (3.26)\\
&\quad \left. + \sum_{m=1}^{\infty} c_m \cos(m\omega_0 t) \left[\cos(n\omega_0 t) \cos(\Delta\omega t) - \sin(n\omega_0 t) \sin(\Delta\omega t) \right] \right) && (3.27)\\
&= A_n \left(\frac{c_0}{2} \cos((n\omega_0 + \Delta\omega)t) \right. && (3.28)
\end{aligned}$$

$$+\sum_{m=1}^{\infty} c_m \left[\cos(m\omega_0 t)\cos(n\omega_0 t) - \cos(m\omega_0 t)\sin(n\omega_0 t)\right]\cos(\Delta\omega t)\Big) \quad (3.29)$$

The only dc component of the proceeding expansion occurs when $m = n$. At this frequency, the left cosine product in equation 3.29 squares, giving us

$$\begin{aligned}
&\left[\cos(n\omega_0 t)\cos(n\omega_0 t) - \cos(n\omega_0 t)\sin(n\omega_0 t)\right]\cos(\Delta\omega t) && (3.30)\\
&= \left[\cos^2(n\omega_0 t) - \cos(n\omega_0 t)\sin(n\omega_0 t)\right]\cos(\Delta\omega t) && (3.31)\\
&= \left[\frac{1}{2} + \frac{1}{2}\cos(2n\omega_0 t) - \frac{1}{2}\sin(2n\omega_0 t)\right]\cos(\Delta\omega t) && (3.32)\\
&= \left[\frac{1}{2} + \frac{1}{\sqrt{2}}\cos\left(2n\omega_0 t + \frac{\pi}{4}\right)\right]\cos(\Delta\omega t) && (3.33)
\end{aligned}$$

This seems like useless algebra until we take the next step, which is calculating the phase noise at $\Delta\omega$ due to current source $I_n(t)$. In the time domain, we can calculate $\phi(t)$ to be

$$\begin{aligned}
\phi(t) = u(t) * \Lambda(t) I_n(t) &= \int_{-\infty}^{t} \Lambda(\tau) I_n(\tau) d\tau && (3.34)\\
&= \frac{A_n c_0}{2}\int_{-\infty}^{t} \cos((n\omega_0 + dw)\tau d\tau && (3.35)\\
&\quad + \sum_{m=1}^{\infty} A_n c_m \int_{-\infty}^{t} \Big[\cos(m\omega_0\tau)\cos(n\omega_0\tau) && (3.36)\\
&\qquad - \cos(m\omega_0\tau)\sin(n\omega_0\tau)\Big]\cos(\Delta\omega\tau)d\tau && (3.37)
\end{aligned}$$

Now we take advantage of equation 3.33. Convolving with $u(t)$ is like taking a moving sum. Therefore, only the lowest frequency components are significant. This tells us that the only component of the integral which is preserved is for $m = n$. Therefore we can simplify equation 3.37 to

$$\begin{aligned}
\phi(t) &\approx \int_{-\infty}^{t} A_n \frac{c_0}{2}\cos((n\omega_0 + \Delta\omega)\tau)d\tau + \int_{-\infty}^{t} A_n \frac{c_n}{2}\cos(\Delta\omega\tau)d\tau && (3.38)\\
&= \frac{A_n c_0 \sin((n\omega_0 + dw)t)}{2(n\omega_0 + \Delta\omega)} + \frac{A_n c_n \sin(\Delta\omega t)}{2\Delta\omega} && (3.39)
\end{aligned}$$

Note that the term on the left is negligible if $n \neq 0$, and if $n = 0$, the term on the right is zero since the lowest Fourier coefficient c_m is c_1 in the sum in equation 3.37. So our final value for $\phi(t)$ resulting from a current $I_n(t) = A_n \cos((n\omega_0 + \Delta\omega)t)$ is

$$\phi(t) = \frac{A_n c_n \sin(\Delta\omega)}{2\Delta\omega} \tag{3.40}$$

Where c_n is the n^{th} coefficient of the $\Lambda(t)$ Fourier expansion.

If we recall equation 3.5, we can now easily write the phase noise contribution by a sinusoidal current:

$$\mathcal{L}(\Delta\omega) = 10 \cdot \log\left(\frac{\mathcal{P}_\phi(\Delta\omega)}{2}\right) = 10 \cdot \log\left(\frac{1}{2}\frac{\frac{A_n c_n}{2\Delta\omega}}{2}\right) = 10 \cdot \log\left(\frac{A_n c_n}{4\Delta\omega}\right)^2 \tag{3.41}$$

In the frequency domain, white noise is flat, which means all frequencies are represented equally in power. Therefore, if we would like to know the white noise current produced phase noise at a $\Delta\omega$ offset from the carrier, the components of noise which we must consider are precisely those $I_n(t) = A_n \cos((n\omega_0 + dw)t)$, for n ranging from zero to infinity. Mathematically, the phase noise at a $\Delta\omega$ offset from ω_0 arising from a white noise source of square manitude $\overline{i_n^2}/\Delta\omega$ is equal to the phase noise generated by the following signal:

$$I = \sqrt{2\frac{\overline{i_n^2}}{\Delta\omega}} \sum_{n=-\infty}^{\infty} \cos((n\omega_0 + \Delta\omega)t) \tag{3.42}$$

We can substitute this into equation 3.41 to get an expression for the total amount of phase noise at a given offset:

$$\mathcal{L}(\Delta\omega) = 10 \cdot \log\left(\sum_{n=-\infty}^{\infty} \frac{2\frac{\overline{i_n^2}}{\Delta\omega}c_n^2}{(4\Delta\omega)^2}\right) = 10 \cdot \log\left(\frac{\overline{i_n^2}}{\Delta\omega} \cdot \frac{\sum_{n=-\infty}^{\infty} c_n^2}{8\Delta\omega^2}\right) \tag{3.43}$$

$$= 10 \cdot \log\left(\frac{\overline{i_n^2}}{\Delta\omega} \cdot \frac{\sum_{n=0}^{\infty} c_n^2}{4\Delta\omega^2}\right) \tag{3.44}$$

However, we know by Parseval's theorem,

$$\sum_{n=0}^{\infty} c_n^2 = 2\omega_0 \int_0^{\frac{1}{\omega_0}} |\Lambda(t)|^2 dt = 2\Lambda_{RMS}^2 \tag{3.45}$$

Hence our phase noise for a current noise into one node is

$$\mathcal{L}(\Delta f) = 10 \cdot \log\left(\frac{\overline{i_n^2}}{\Delta f} \cdot \frac{\Lambda_{RMS}^2}{8\pi^2 \Delta f^2}\right) \tag{3.46}$$

Flicker Noise

This model also makes analysis of flicker noise contribution simple. If we denote the corner frequency $\omega_{1/f}$, we know at $\omega = \omega_{1/f}$, the flicker noise has value equal to $\overline{i_n^2}/\Delta\omega$, the thermal noise. Equating these gives us an expression for flicker noise

$$\frac{\overline{i_{n\frac{1}{f}}^2}}{\Delta f} = \frac{\omega_{1/f}}{\Delta f} \frac{\overline{i_n^2}}{2\pi\Delta f} \tag{3.47}$$

Therefore we can just substitute this noise equation into 3.46 to obtain

$$\mathcal{L}(\Delta f) = 10 \cdot \log\left(\frac{\overline{i_n^2}}{\Delta f} \cdot \frac{c_0^2 \omega_{1/f}}{32\pi^3 \Delta f^3}\right) \tag{3.48}$$

Now just as we used the thermal noise to get an expression for flicker noise, we will equate equations 3.46 with 3.48 to obtain the phase noise corner frequency ω_{1/f^3}. This corner is so named because it divides the $1/f^2$ and $1/f^3$ regions of the phase noise plot.

$$\omega_{1/f^3} = 2\pi\left(\frac{c_0^2 \omega_{1/f}}{4\pi\Lambda_{RMS}^2}\right) = \left(\frac{c_0}{2\Lambda_{RMS}}\right)^2 \cdot \omega_{1/f} \tag{3.49}$$

This tells us that to ensure minimal upconversion of flicker noise into phase noise, the DC component of Λ must be small. Simply, we would like to make Λ odd symmetric.

3.4 Method for Phase Noise Calculation

Now that we know equation 3.46, we can use this to calculate the complete phase noise for an oscillator. The method is as follows. Compute Λ for each node in the oscillator. Since oscillators are fairly symmetric, we usually only have to do this once. Compute the current noise from at each node. Again, each node will likely have the same current noise. Because the noise at a node is a function of the node voltage and hence periodic just like Λ, we can define $\frac{\overline{i_n^2}}{\Delta f}(\tau) = \alpha(\tau) \cdot \frac{\overline{i_n^2}}{\Delta f}_{pk}$, Where $\alpha(\tau)$ is a scaling function with peak magnitude unity. Because α and Λ both are periodic with period $\frac{1}{\omega_0}$, we simply replace $\Lambda(\tau)$ with $\Lambda'(\tau) = \alpha(\tau) \cdot \Lambda(\tau)$ and do all our calculations with Λ'. Therefore our final equation for the white noise $(1/f^2)$ region of oscillator phase noise is given by

$$\mathcal{L}(\Delta f) = 10 \cdot \log \left(N \cdot \frac{\overline{i_n^2}}{\Delta f}_{pk} \cdot \frac{\Lambda'^2_{RMS}}{8\pi^2 \Delta f^2} \right) \tag{3.50}$$

Note that we could multiply by N because we are still assuming linearity, and hence superposition still holds.

3.5 GSM900 Phase Noise Specification

The GSM900 standard uses time-division multiple-access (TDMA) as well as frequency-division multiple-access (FDMA) protocols. Because adjacent channel spacing is 200KHz, oscillator phase noise must fall very quickly to avoid the reciprocal mixing problems mentioned at the beginning of this chapter. The plot in figure 3-3 shows the maximum allowable phase noise at a specified offset.

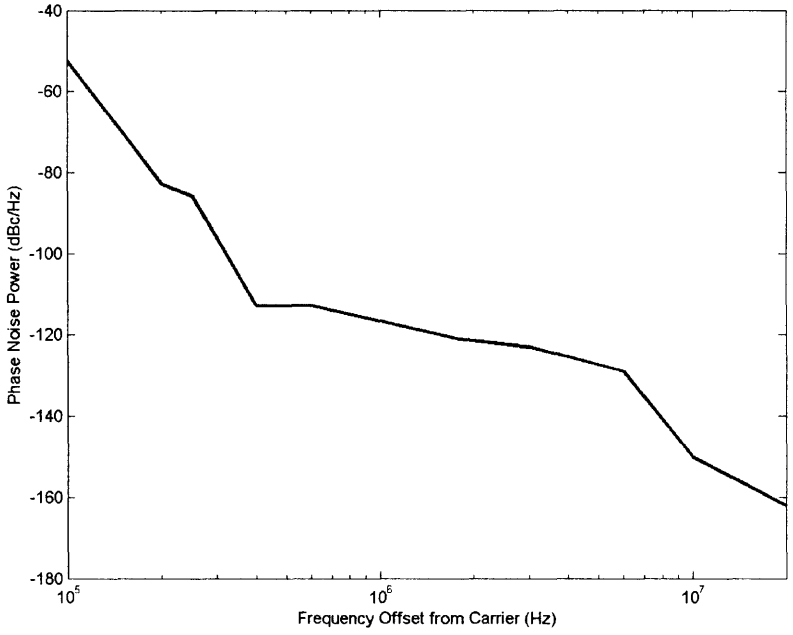

Figure 3-3: The line represents the maximum phase noise allowed for a given offset frequency.

Bibliography

[1] Behzad Razavi. *RF Microelectronics*. Prentice Hall Communications Engineering and Emerging Technology Series. Prentice Hall, Upper Saddle River, NJ, first edition, 1998.

[2] Thomas H. Lee. *The Design of CMOS Radio-Frequency Integrated Circuits*. Cambridge University Press, New York, NY, second edition, 2004.

[3] Thomas H. Lee. *Planar Microwave Engineering*. Cambridge University Press, New York, NY, first edition, 2004.

[4] Reza Navid, Thomas H. Lee, and Robert W. Dutton. Minimum achievable phase noise of rc oscillators. *IEEE Journal of Solid-State Circuits*, 40(3):630–637, March 2005.

[5] Donhee Ham, William Andress, and David Ricketts. Phase noise in oscillators. In *Talk at International Workshop on SiP/SoC Integration of MEMS and Passive Components with RF-ICs*, 2004.

[6] Thomas H. Lee and Ali Hajimiri. A general theory of phase noise in electrical oscillators. *IEEE Journal of Solid-State Circuits*, 33(2):179–194, February 1998.

[7] David B. Leeson. A simple model of feedback oscillator noise spectrum. *Proceedings of IEEE*, 54(2):329–330, 1966.

[8] Behzad Razavi. A study of phase noise in cmos oscillators. *IEEE Journal of Solid-State Circuits*, 31(3):331–343, March 1996.

[9] Ali Hajmiri, Sotirios Limotyrakis, and Thomas H. Lee. Jitter and phase noise in ring oscillators. *IEEE Journal of Solid-State Circuits*, 34(6):790–804, June 1999.

[10] John A. McNeill. Jitter in ring oscillators. *IEEE Journal of Solid-State Circuits*, 32(6):1367–1371, June 1997.

[11] Alan V. Oppenheim, Ronald W. Schafer, and John R. Buck. *Discrete-Time Signal Processing*. Pearson Education, New York, NY, second edition, 1999.

Chapter 4

Voltage Controlled Oscillator

From the proceeding chapter, it is clear that the phase noise performance of the VCO will set an upper bound for the performance of the system. We therefore focus our attention first to the VCO. Although we know we would like to use a ring oscillator, within this category are many choices of topology. Before choosing one, let us begin with some theory of oscillators.

4.1 Oscillator Theory

We define an oscillator as a circuit which produces a sustained periodic output. From a system perspective, it is an unstable feedback system. Let us consider a simple amplifier connected in feedback configuration, as shown in figure 4-1. Through Nyquist diagrams or similar linear analysis, we can show that the system will oscillate at the frequency when the phase shift around the loop is exactly zero, and the loop gain is

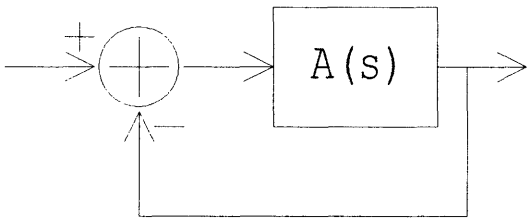

Figure 4-1: An amplifier with unity gain negative feedback.

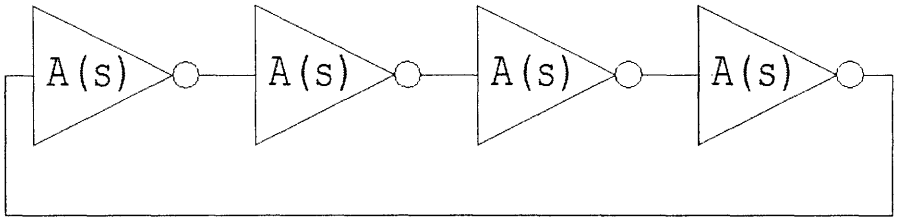

Figure 4-2: Four inverters, each with $3\pi/2$ phase shift at 100MHz in a ring.

exactly one. These two criterion are difficult to meet together, and if they are not the circuit will not sustain oscillation. If the gain at the crossover frequency is anything but exactly one, the amplitude of oscillation will exponentially rise or decay.

Instead of trying to thread the needle, we design the amplifier with too much gain at the zero phase point, and employ some sort of circuit to limit the gain. A very crude limiting mechanism is present in all electrical circuits, namely supply railing. The simplest method of extending our linear analysis techniques to such nonlinear limiting systems is through use of describing functions. These are covered extensively in [1]. Alternatively, an automatic gain control (AGC) circuit may be used, which may be linear or nonlinear.

In the case of ring oscillators, we can cascade stages in a chain, such that the gain goes with the power of the number of stages, and the phase shift adds. When doing this, we must be careful to avoid conditionally stable systems. For example, consider a single stage inverter with a $A_v = 1$ at 100MHz, and the phase shift is -270° at this frequency. Remember, an inverter already has -180° of phase shift at DC. In other words, we can model this as a single pole system with

$$A(s) = \frac{v_o}{v_i}(s) = -\frac{10^8}{\tau s + 1} \tag{4.1}$$

where τ is a low ferquency pole several decades below 100MHz. Since the gain is exactly one at 100MHz, and each inverter contributes $3\pi/2$ radians of phase shift, it makes sense that cascading four such inverters in a ring will produce an oscillator.

Unfortunately, common sense tells us that there is simple solution to this circuit.

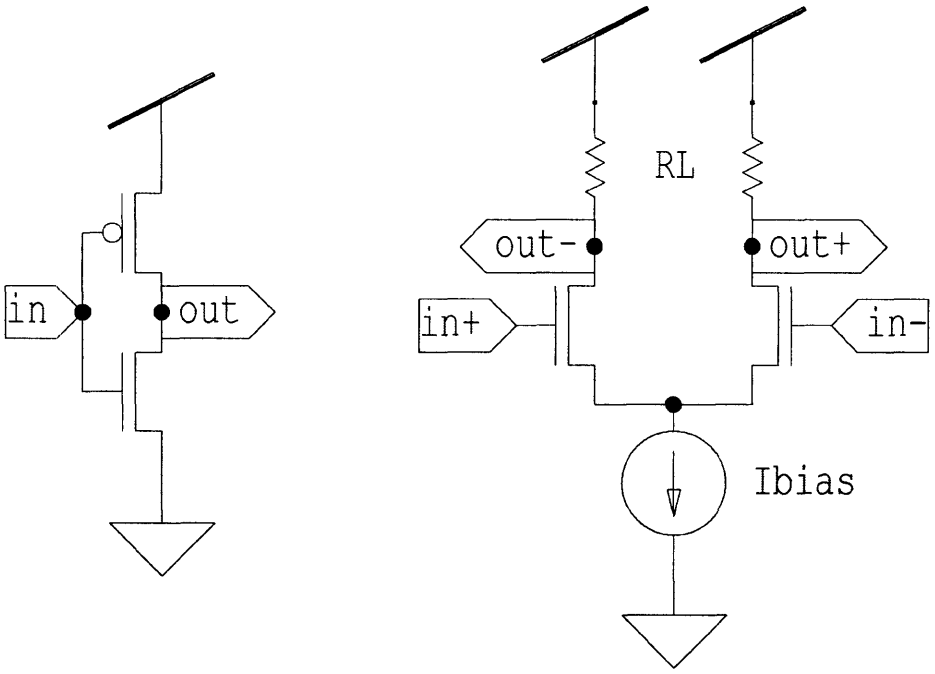

Figure 4-3: On the left is a CMOS inverter, the simplest single-ended topology. The right is a simple differential inverter stage. Note that the frequency tuning mechanism is not shown. Sometimes R_L is replaced with an active load.

The circuit has a stable DC point, which means two of the nodes will be low, and two will be high. This is an example of a system which is conditionally stable. The solution to this is for single-ended ring oscillators, the number of stages must be odd. In differential oscillators, regardless of the number of stages, inverting the connection between two stages yields a π phase shift, so any number of stages may be used.

4.2 Ring Oscillators

4.2.1 Single-Ended vs. Differential

We would like to choose a topology for our oscillator, and the first question we ask is would we like our oscillator to be single-ended or differential. Here we will compare the advantages and disadvantages of each.

From the proceeding section, we know that differential oscillators can have an even

number of stages. This is useful, because it means we can have quadrature clocks without additional circuitry. Tapping the signal at two opposite nodes in a differential oscillator will be $\pi/2$ radians apart. Single-ended ring oscillators, on the other hand, require phase shifting or frequency dividing circuits to generate a quadrature clock, since the phase shift at each node is π/N apart, where N is the odd number of stages in the ring oscillator.

Referring to figure 4-3, a problem with the differential topology is limited output swing. The differential amplifier has problems with excursions toward ground, whereas the single-ended stage has rail to rail swing.

One major concern with integrated circuits is substrate noise injection from switching circuits. In this regard, the differential topology has the advantage, due to constant biasing with I_{bias}. The current remains relatively constant independent of the differential pair switching. In the single-ended case, crowbar current flows during each transition, which causes the supply to ripple with the current transient drops across the resistive supply lines.

By the same token, the power supply insensitivity of differential oscillators is superior to that of the single-ended oscillators, in the same way that differential signalling has superior external noise rejection over single-ended signalling.

The phase noise performance of single-ended ring oscillators is very weakly dependent on the number of stages in the ring, whereas increasing the number of stages in a differential oscillator increases the phase noise, when keeping oscillation frequency and power constant.[2, 3]. The fundamental reason for this is that the dv/dt of the node voltage must increase with the number of stages, if oscillation frequency is kept constant. We also know dv/dt is proportional to the current into the node and the node capacitance. In a single ended ring oscillator, we increase dv/dt by decreasing Δt and increasing charging current, I, thus increasing dv/dt. Since we have decreased Δt, the current per transition which is $I \cdot \Delta t$ remains constant. In a differential topology, since I which is limited by I_{bias} decreases with the number of stages (if we want total power to stay constant) the only way to increase dv/dt is to decrease the node capacitance. Using a hand-waving argument, as the noise bandwidth of capacitive

systems is related by kT/C, decreasing C makes the amount of noise increase.

Another noise concern in differential oscillators is the noise from the tail current source, I_{bias}. As shown in [4, 2] the low frequency noise from the current source transistors gets folded into close-in phase noise with varying magnitude dependent on the symmetry of the ISF for oscillator. Low frequency noise is dominated by flicker noise, (section 2.1.3) and this is dependent on DC current, I_{bias}. Since I_{bias} is the largest current flowing in the differential oscillator, this noise contribution is significant. Since differential LC oscillators also employ a tail current source, efforts have been made at reducing the effect of this noise.[5]

Table 4.1: Comparing the merits of differential versus single-ended ring oscillator topologies.

Topology	Quadrature Clocks	Output Swing	Noise Injection	Supply Sensitivity	Stage Dependent Noise	Flicker Noise
Single-Ended	hard	full	high	high	no	low
Differential	easy	limited	low	low	yes	high

4.3 Oscillator Design

From the previous work in published literature, in conjunction with the device noise theory developed in chapter 2 and phase noise theory presented in chapter 3 we are now ready to design our oscillator. As a design guide, we will begin by enumerating a set of goals which we hope will lead to a low phase noise oscillator.

1. The following goals make the oscillator easier to integrate with the remainder of the PLL:

   (a) For ease of interfacing with the frequency divider, full swing outputs are desired, which implies a single-ended oscillator.

   (b) For lower supply sensitivity, a differential topology is desirable.

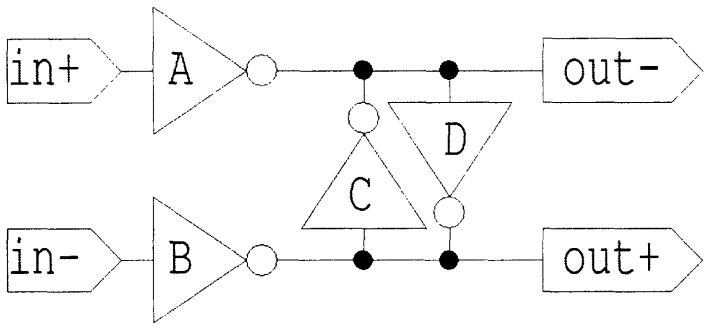

Figure 4-4: The delay cell is two main inverters A and B with a latch comprised of C and D between them to ensure synchronized switching.

(c) For ease of implementing quadrature clocks, a differential topology with an even number of stages is desired.

2. The following items are to ensure good phase noise performance:

   (a) No tail current source, to reduce $1/f$ modulation.

   (b) Make Λ symmetric.

   (c) The MOSFETs in the inverters must be full-on or full-off for as much of the period as possible.[6] Doing so reduces Λ^2_{RMS}.

We propose the following topology, which we claim simultaneously satisfies all the integration goals, while sacrificing very little from the phase noise goals. A single delay cell is shown in figure 4-4.

This delay cell is differential, so it is less supply-sensitive and we can use an even number of stages. However, it retains many of the benefits of a single ended topology, such as rail-to-rail output swing and there is no tail current source. The only disadvantage of this pseudo-differential topology as compared to the traditional differential pair is this switching cell injects more supply noise, as crowbar current still flows.

The design for this cell is basically an extension to the differential oscillator cell presented in [6]. Their work uses a similar pseudo-differential pair, however instead of a full latch, a PMOS cross-coupled pair is used. Furthermore, negative skew clocks

are used as presented in [7]. In this oscillator, the negative-skewed clocks are not used because this increases power consumption and also makes the ISF asymmetric. Also, a cross-coupled NMOS pair is added to the extant cross-coupled PMOS pair to form a complete latch. This also aids in the symmetry of the ISF, as will be shown.

The latch has internal positive feedback which is used to restore the dv/dt at each node. By switching fast, we ensure that the main inverters A and B experience full switching for as much of the period as possible.

4.3.1 Frequency Tuning

So far the cell only has a fixed delay. To implement a variable delay, we will adjust the strength of the latch inverters C and D. Traditionally, this was done by current starving the inverters. We propose a new method of current starving the inverters, by placing the current control transistors at the drains of the switches. Borrowing a page from charge pump notation, this is called source-switching, and has several advantages over drain switching.[8] The main advantage for our circuit is the noise reduction it affords. Referring to figure 4-5, drawing the small signal model of the cascode shows that transistor M2's noise source $\overline{i_{n2}^2}$ is shorted across its own small signal resistance, since $1/g_m$ is so much smaller than r_o. Thus only M1's noise contributes to the overall circuit noise. Putting all this together we arrive at the circuit level schematic of our delay cell, shown in figure 4-6.

The nodes v_r and v_f independently control the latch's pull-up and pull-down strengths, respectively. This way we can fine-tune the rise and fall times of the delay cell to account for process variations by adjusting the common mode voltage of v_r and v_f. In the ideal case, the devices are sized such that

$$\frac{v_r + v_f}{2} = \frac{VDD}{2} \tag{4.2}$$

The complete oscillator schematic is shown in figure 4-7.

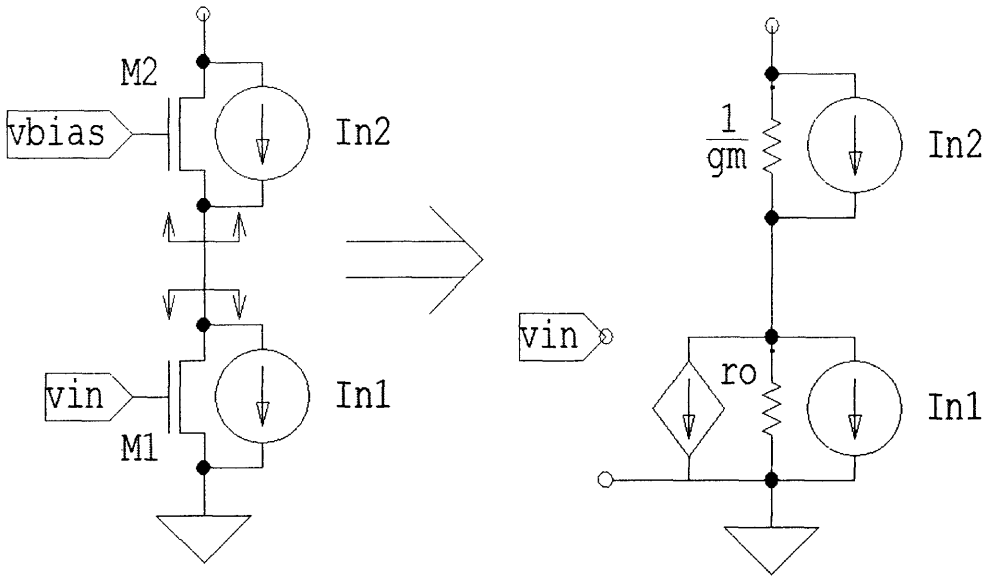

Figure 4-5: Looking into the drain of M1, we see resistance r_o, and into the source of M2 we see roughly $1/g_m$.

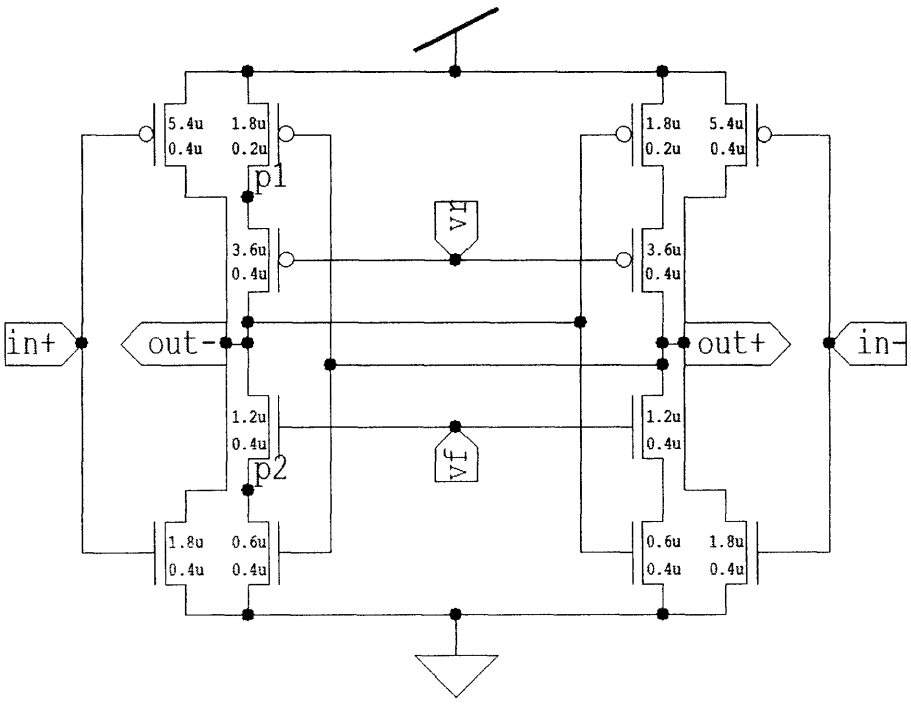

Figure 4-6: Note that the ratio of PMOS to NMOS sizing is equal to μ_n/μ_p.

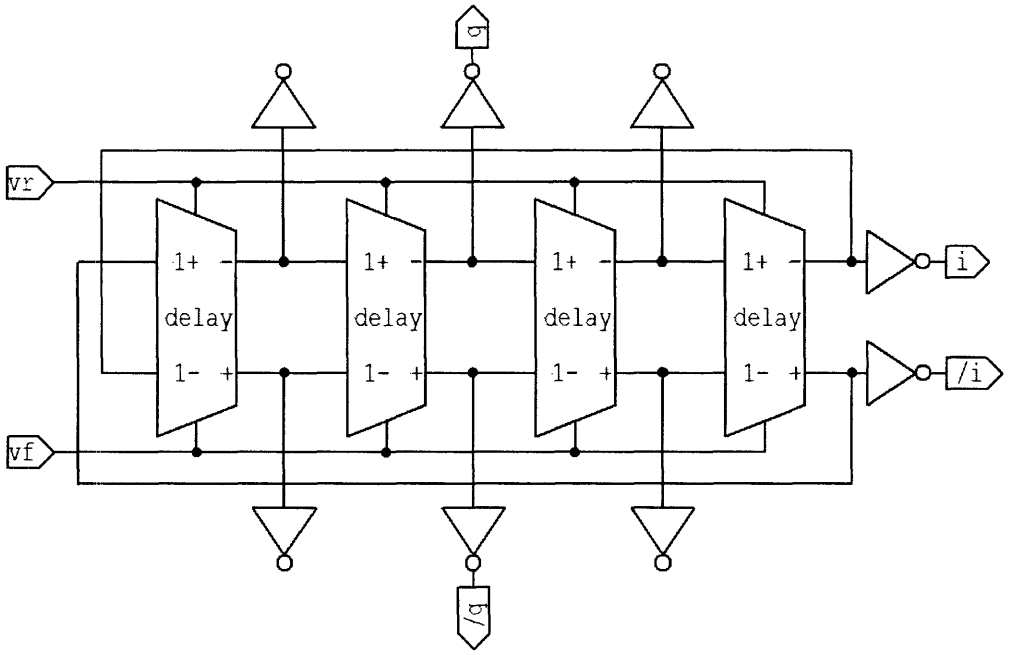

Figure 4-7: The voltage controlled oscillator. Note the inclusion of dummy loads to ensure delay symmetry between stages.

Bibliography

[1] James K. Roberge. *Operational Amplifiers.* John Wiley & Sons Inc., New York, NY, first edition, 1975.

[2] Ali Hajmiri, Sotirios Limotyrakis, and Thomas H. Lee. Jitter and phase noise in ring oscillators. *IEEE Journal of Solid-State Circuits*, 34(6):790–804, June 1999.

[3] John A. McNeill. Jitter in ring oscillators. *IEEE Journal of Solid-State Circuits*, 32(6):1367–1371, June 1997.

[4] Thomas H. Lee and Ali Hajimiri. A general theory of phase noise in electrical oscillators. *IEEE Journal of Solid-State Circuits*, 33(2):179–194, February 1998.

[5] Pietro Andreani and Henrik Sjöland. Tail current noise suppression in rf cmos vcos. *IEEE Journal of Solid-State Circuits*, 37(3):342–348, March 2002.

[6] Chan-Hong Park and Beomsup Kim. A low-noise 900mhz vco in $0.6\mu m$ cmos. *1998 Symposium on Circuits and Systems, Digest of Technical Papers*, pages 28–29, June 1998.

[7] Seog-Jun Lee, Beomsup Kim, and Kwyro Lee. A novel high-speed ring oscillator for multiphase clock generation using negative skewed delay scheme. *IEEE Journal of Solid-State Circuits*, 32(2):289–291, February 1997.

[8] Woogeun Rhee. Design of high-performance cmos charge pumps in phase-locked loops. In *Proceedings of the 1999 IEEE International Symposium on Circuits and Systems*, number 2, pages 545–548, May 1999.

Chapter 5

Frequency Divider

The frequency divider is analogous to the feedback path gain in normal feedback systems; reducing it makes the output larger, and increasing it reduces the output. In this case increasing the output means making the phase faster, which means a higher output frequency.

5.1 Fixed Division

5.1.1 Asynchronous Dividers

The simplest divider is a chain of D flip-flops which form an asynchronous counter. The division ratio is then 2^N, where N is the number of stages. However, asynchronous designs have higher jitter in their output because the timing uncertainty of each stage is compounded upon the uncertainty of its input.

5.1.2 Synchronous Dividers

A synchronous divider has each flip flop clocked by the input clock, with static logic between each stage to implement the division. The advantage of this is the jitter is a factor of $\sqrt{N}$ less than in the asynchronous case, since the output is dependent on the reference clock instead of the input from the previous stage.

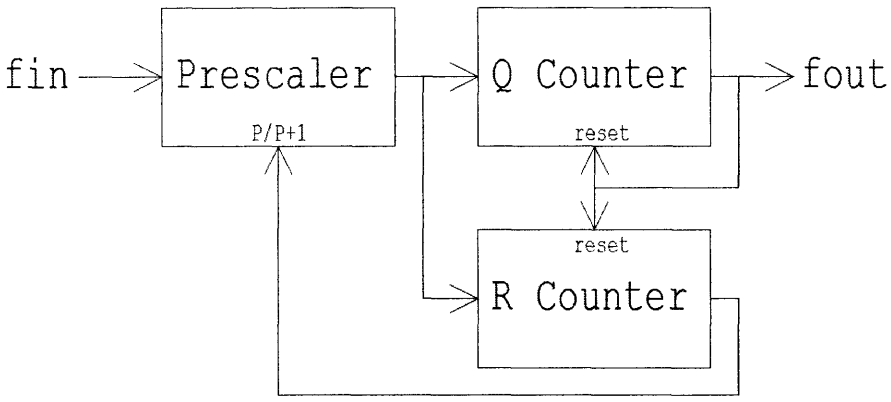

Figure 5-1: The frequency input only feeds the prescaler, while the divided output of the prescaler feeds the later stages. The prescaler divides by P when the input is high, and $P + 1$ when the input is low.

5.2 Variable Division

Increasing the number of possible division ratios in a circuit increases its complexity, thereby causing power consumption to increase and maximum operating frequency to decrease. For this reason, we would like to make the high frequency circuits as simple and small as possible.

5.2.1 Dual-Modulus Prescalers

Dual-modulus prescalers follow this principle by using one high frequency circuit, the prescaler, followed by several low frequency circuits. The prescaler has only two possible division ratios, P and $P+1$. By varying the modulus of the prescaler, many division ratios are possible.

To analyze the operation of this circuit, suppose we have just completed a cycle, i.e. f_{out} has just gone high, and thus counters Q and R have just reset, bringing f_{out} back low again. Clearly, we will need to have $R < Q$, otherwise f_{out} will reset it before it ever goes high. So in one cycle of f_{out}, we will count by $(P + 1)$ R times, and P $(Q - R)$ times. So the total divide ratio is given by

$$N = R(P + 1) + (Q - R)P = QP + R \tag{5.1}$$

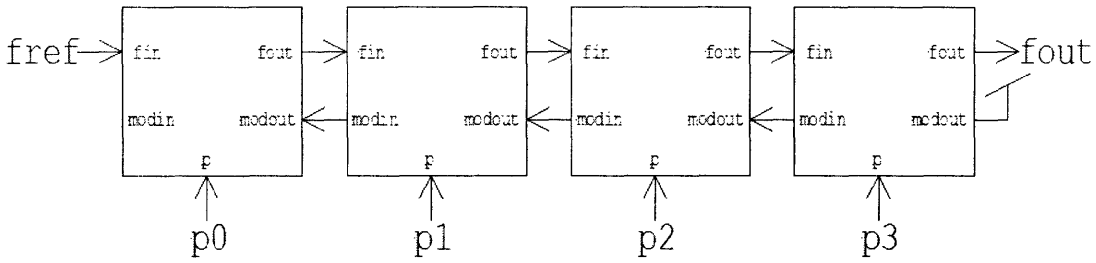

Figure 5-2: In a multi-modulus prescaler, the counters Q and R of the dual modulus prescaler are just nested dual modulus prescalers.

The advantage of this divider is that only the prescaler, sees the high frequency input, the two programmable counters see at most f_{in}/P.

For a given average divide ratio N, there is a trade-off between a wide division range around N and power consumption. Supposing $N = QP+R$ for some Q, P, and R. If we want to have a continuous divide range, we need $Q > P$ because $Q > R$ so to ensure overlap as we change Q to $Q+1$ we need $R \geq P$. This means we have an upper bound on P, and thus a minimum bound on the clock frequency of the programmable dividers Q and R, which gives us a lower bound on power consumption.

5.2.2 Multi-Modulus Prescalers

The concept of the dual-modulus prescaler is extended in the work by Vaucher et. al.[1] to both extend division range and reduce power consumption. The divider is comprised of a cascade of N 2/3 blocks, as shown in figure 5-2. Each 2/3 block has a 2/3 prescaler and some control logic. In addition to the f_{in} and f_{out} signals, each block has an input p which tells it to count by 2 or 3, and a mod_{in} signal and mod_{out} signal.

When the input mod_{in} goes high for a particular block, the output mod_{out} goes high either 2 or 3 cycles afterward depending on the input p. At the same time, f_{out}, which was until this point just $f_{in}/2$, skips a pulse and outputs $f_{in}/3$ for one period. Since the block is counting by its own clock f_{in}, and if this is the N^{th} block then $f_{in} = f_{ref}/2^N$, dividing by 3 instead of 2 means adding an additional 2^N cycles of

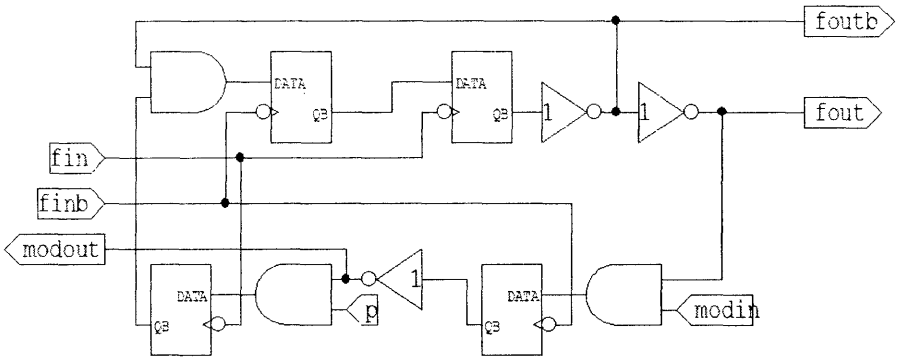

Figure 5-3: The circuit implementation of a 2/3 block in a multi-modulus prescaler.

f_{ref} worth of delay. We see that the delay propagates both left to right in the upper path and right to left in the lower path. If we add up all the delays along the chain, we get the total division ratio

$$D = 2^N + p_{n-1}2^{N-1} + p_{n-2}2^{N-2} \cdots p_1 2 + p_0 \tag{5.2}$$

This has a full octave of division range, and can be extended even more with techniques shown in [1].

5.3 2/3 Block

The operation of the 2/3 block was outlined in the previous section, now we will consider the details of its implementation. A schematic for a general 2/3 block is shown in figure 5-3. To analyze this, it's best to break it into pieces. If we assume that p is zero for now, we can simplify the figure to be as shown in figure 5-4. The top two inverters are connected in a negative feedback configuration to form a divide-by-2 counter, and the output f_{out} changes every falling edge of f_{in}. Note furthermore that mod_{out} changes on the falling edge of $\overline{f_{in}}$, which is the rising edge of f_{in}. We can then assume that the block following this one does the same thing, and thus state that mod_{in} will rise on the rising edge of f_{out}. We just said that f_{out} changes on the falling edge of f_{in}, so mod_{out} will rise half a period of f_{in} later. With that said, figure 5-5 is

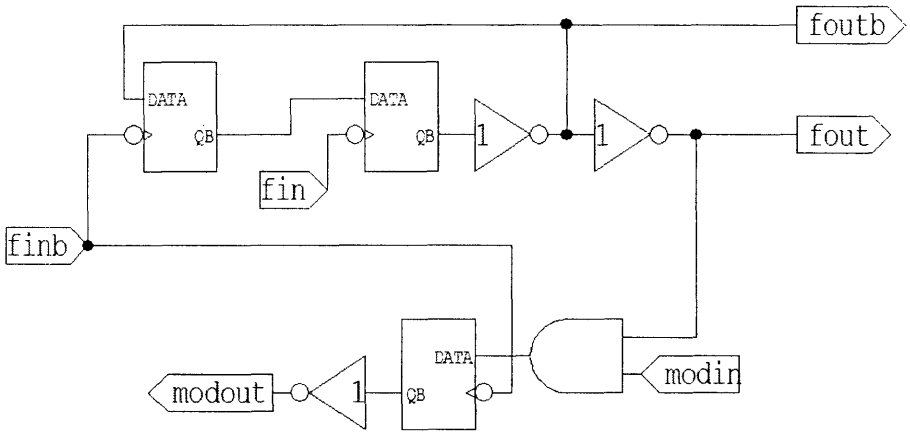

Figure 5-4: Here we have assumed that p is always 0, so we can remove one of the flip-flops since its output is stuck high.

a timing diagram showing the behavior of the 2/3 block with $p = 0$.

Now let us examine how the 2/3 block behaves with $p = 1$. Ignoring the bottom two flip-flops for a moment, we see that as before, the two two flip-flops produce f_{out} which is negative edge clocked and half the frequency with respect to f_{in}. Now, as before consider what happens when mod_{in} goes high. We know mod_{in} will go high on the rising edge of f_{out}. Once again, this brings mod_{out} low half a cycle of f_{in} later. But this time, bringing mod_{out} high has the effect of making x low, which brings the input of the upper left flip-flop low for an extra cycle. This means that f_{out}, which previously changed every falling edge of f_{in}, will now skip one falling edge. Thus, f_{out} is now divided by three. However, this only lasts one cycle of f_{out}, because mod_{out} goes low again, making x high, and we are back in divide-by-2 operation. A timing diagram of this is shown in figure 5-7.

5.3.1 Implementation of D Flip-Flops

A primary goal of this circuit implementation is small size, so modified TSPC flip-flops were chosen for the core circuits. The TSPC technique is outlined in articles [2, 3].

The dominant high frequency logic type is SCL (source-coupled logic) which uses

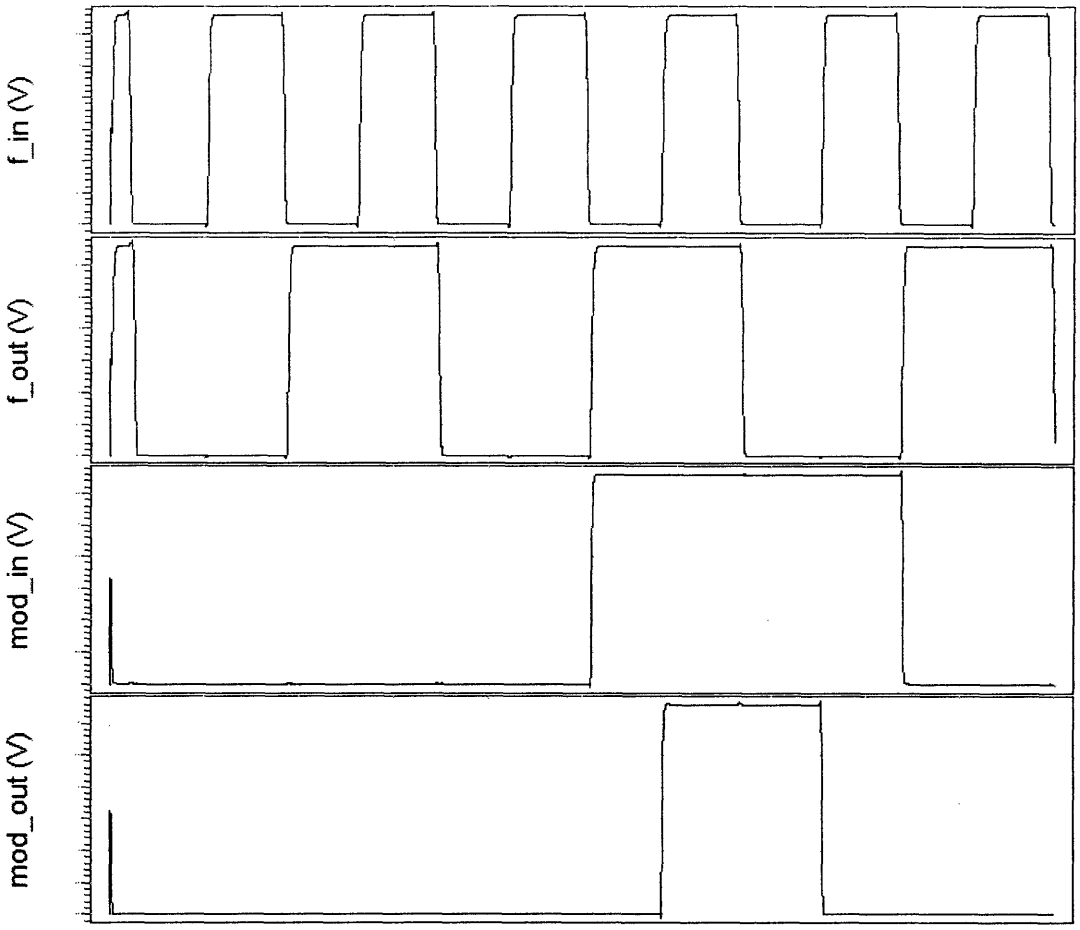

Figure 5-5: We only show f_{in} and f_{out} and not their compliments for clarity.

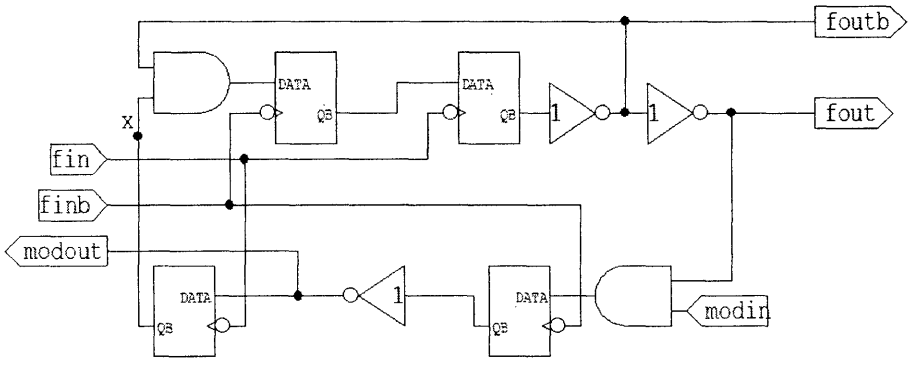

Figure 5-6: With $p = 1$, we can drop the AND gate in front of one of the flip-flops.

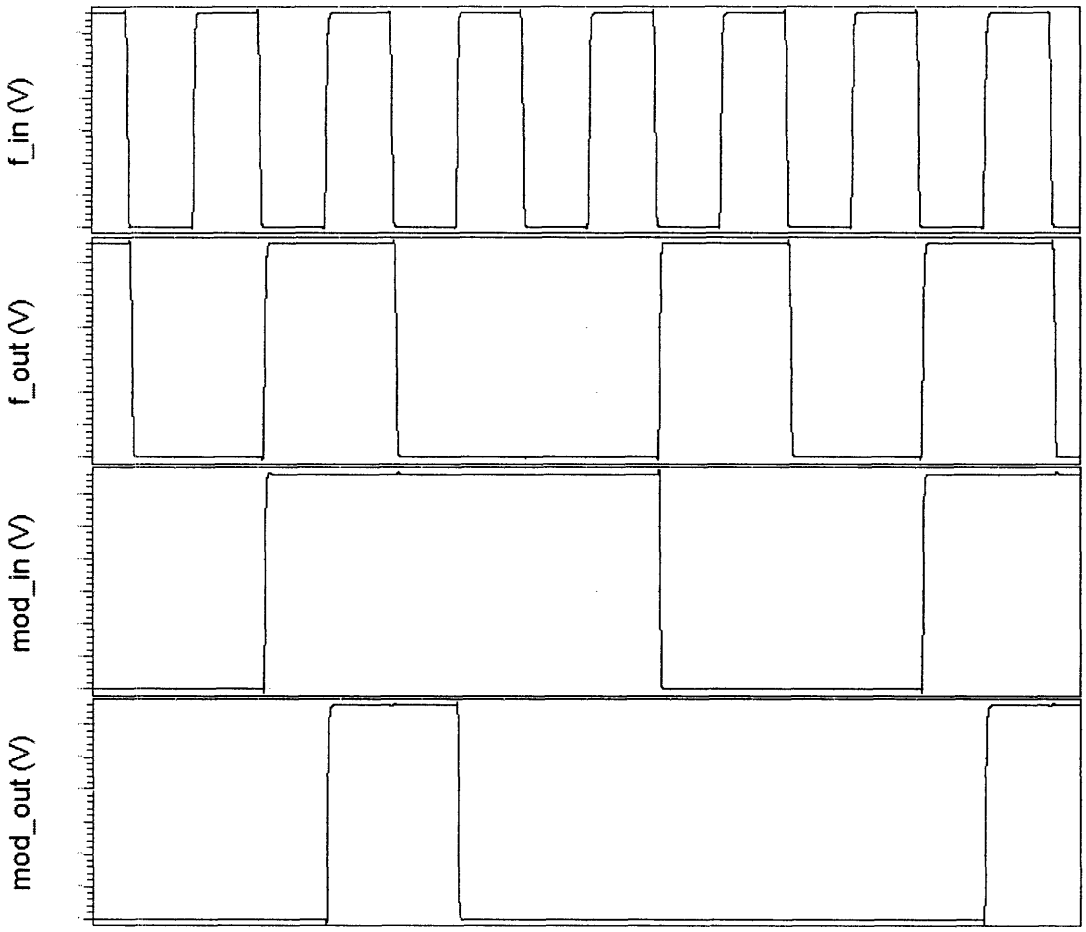

Figure 5-7: When $p = 1$ the signal x is allowed to go low, thus making f_{out} skip one period of f_{in}, thereby counting by 3 instead of 2.

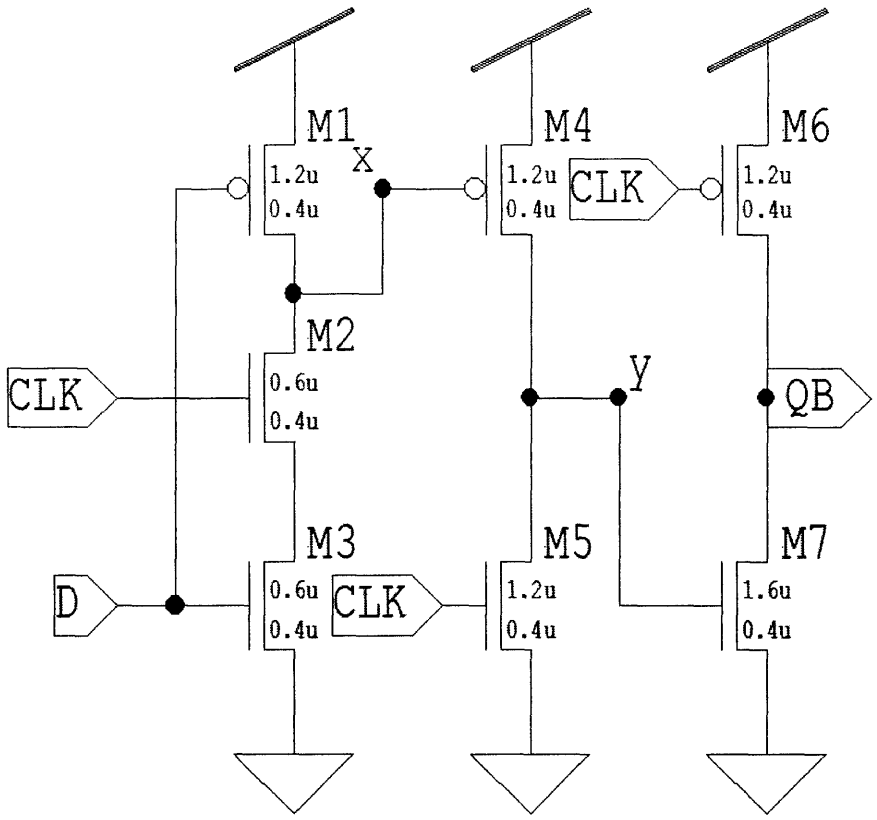

Figure 5-8: The complete flip-flop requires only 7 transistors.

transistors as current steering switches. Since logic levels are implemented in current instead of voltage, there is little voltage swing on nodes, and thus power is saved and higher operating frequencies are attained. However, these circuits require an interface driver to output normal logic levels, a converter at the input, and they consume static power in the tail current.

The modifications to the TSPC flip-flop are first proposed in [4], and the schematic for a negative-edge triggered, inverting output D flip-flop is shown in figure 5-8.

The circuit is really the sum of a gated inverter comprised of transistors M1, M2, and M3, as well as a latch, which is made up of transistors M4, M5, M6, and M7. The heart of operation of the flip-flop is that when the first stage can pull down, the latch can only be pulled up. Let us examine the operation of the latch, shown in figure 5-9.

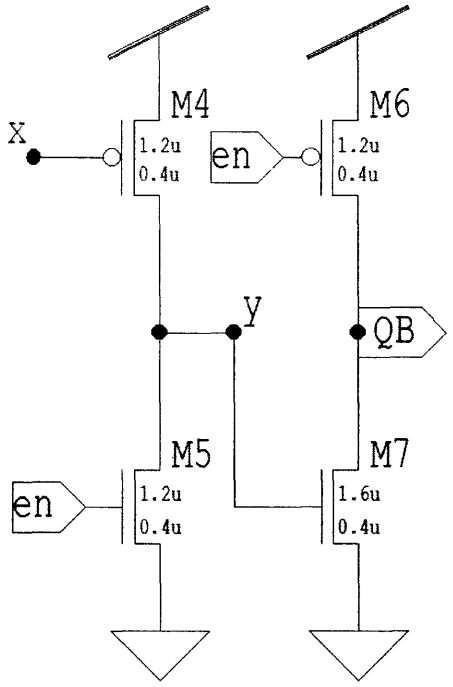

Figure 5-9: The latch is active when *en* is low.

When *en* is high, since M5 is stronger than M4, the node y is pulled down regardless of x. This means the latch is in "hold" mode, as QB is held by the node capacitance. When *en* is low, pulling x low anytime will pull y up, and since M7 is stronger than M6, QB will fall. Note that since y is only actively pulled up, once it rises while *en* is low, QB will not rise again until *en* goes low.

So we see that this latch can pull QB down if x falls when *en* is low. But referring back to figure 5-8, x can only fall when *en* is high. This is why this is a falling edge flip-flop; the output can only change when *clk* changes from high to low.

5.3.2 D Flip-Flop with AND

Fortunately this design for a flip-flop makes integrating logic very easy. The 2/3 block is constructed with AND functions, which are traditionally slower than their inverting counterparts. However, with the inclusion of M2 and M5 into the D flip-flop as shown in figure 5-11, we have a 9 transistor D flip-flop with AND function.

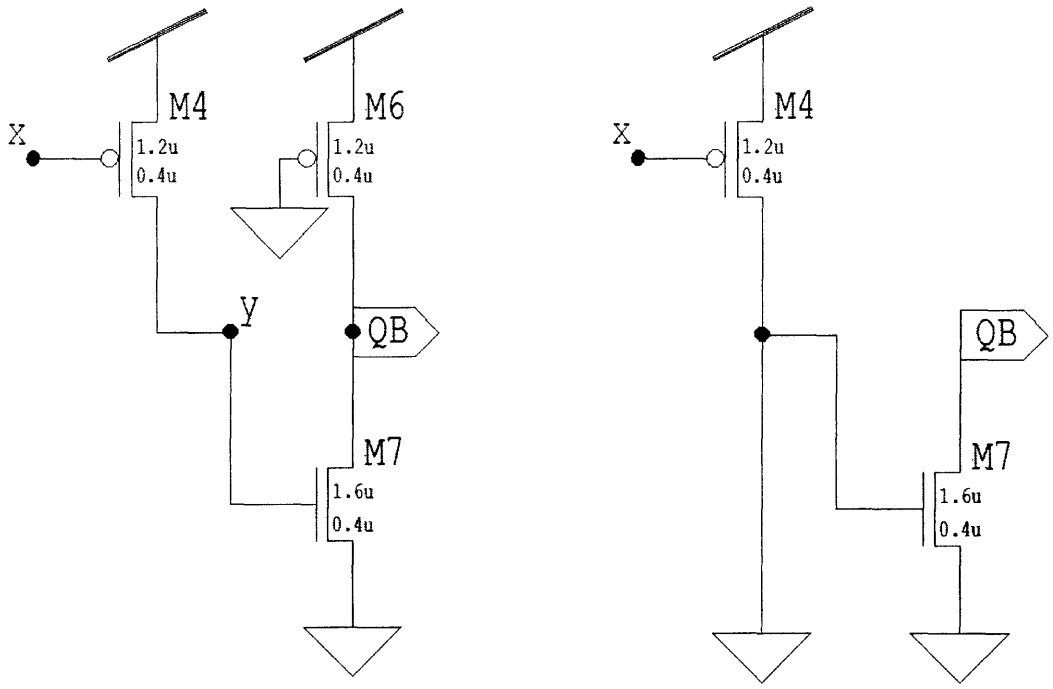

Figure 5-10: Equivalent circuits of latch operation. On the left, *en* is low, and on the right *en* is high.

5.4 5 Stage Multi-Modulus Prescaler

The final divider design used in this PLL is a five stage multi-modulus topology, which is capable of continuous division range between $2^5 = 32$ and $2^6 - 1 = 63$. The divider is shown in figure 5-12. The final block does not have a mod_{in} input, as that is just tied to V_{DD} so some extraneous logic has been removed. The first 2/3 block is the highest frequency block, and the delay path has been slightly rearranged in order to extend its maximum operating frequency at the cost of higher power consumption. The schematic for the first block is shown in figure 5-13.

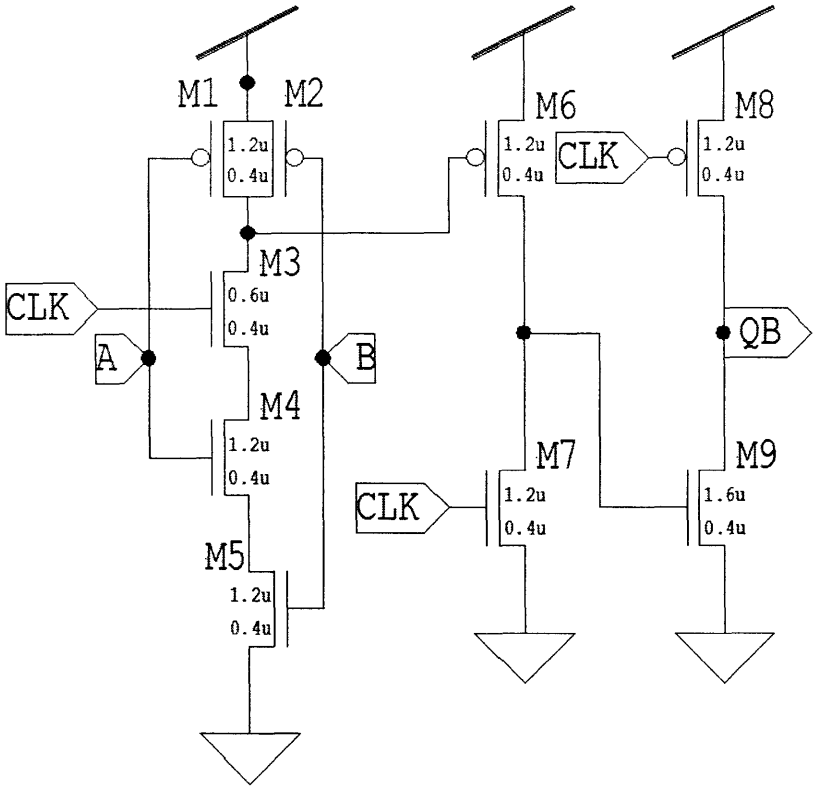

Figure 5-11: The inclusion of two transistors in the first column implements the AND function.

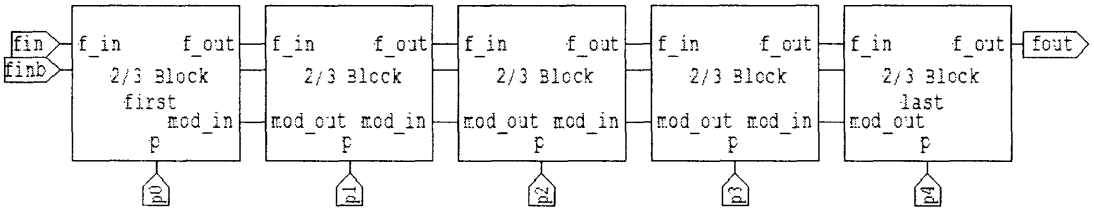

Figure 5-12: The final topology of the frequency divider.

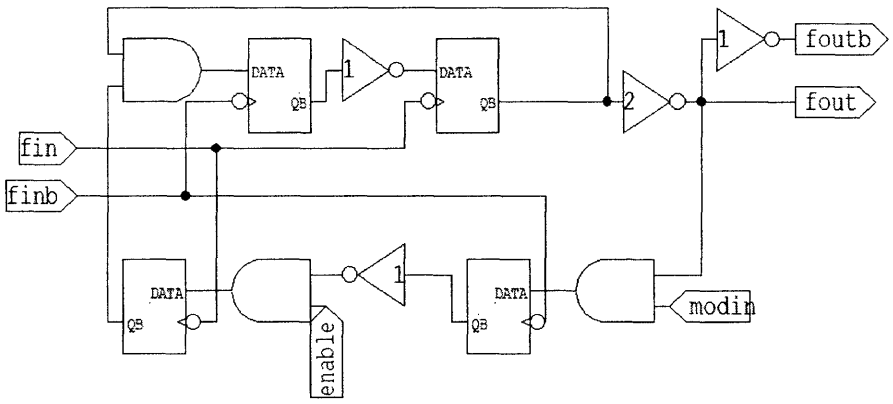

Figure 5-13: The first 2/3 block places an inverter between the two upper flip-flops to remove a critical delay path, as well as compensate for skew between f_{out} and $\overline{f_{out}}$.

Bibliography

[1] Cicero S. Vaucher, Igor Ferencic, Matthias Locher, Sebastian Sedvallson, Urs Voegeli, and Zhenhua Wang. A family of low-power truly modular programmable dividers in standard 0.35-μm cmos technology. *IEEE Journal of Solid-State Circuits*, 35(7):1039–1045, July 2000.

[2] Ji ren Yuan, Ingemar Karlsson, and Christer Svensson. A true single-phase-clock dynamic cmos circuit technique. *IEEE Journal of Solid-State Circuits*, 22(5):899–901, October 1987.

[3] Ji ren Yuan and Christer Svensson. New single-clock cmos latches and flipflops with improved speed and power savings. *IEEE Journal of Solid-State Circuits*, 32(1):62–69, January 1997.

[4] Byungsoo Chang, Joonbae Park, and Wonchan Kim. A 1.2 ghz cmos dual-modulus prescaler using new dynamic d-type flip-flops. *IEEE Journal of Solid-State Circuits*, 31(5):749–752, May 1996.

[5] Jan M. Rabaey, Anantha Chandrakasan, and Borivoje Nikolić. *Digital Integrated Circuits: A Design Perspective.* Prentice Hall Electronics and VLSI Series. Prentice Hall, Upper Saddle River, NJ, second edition, 2003.

[6] Hamid R. Rategh, Hirad Samavati, and Thomas H. Lee. A cmos frequency synthesizer with an injection-locked frequency divider for a 5-ghz wireless lan receiver. *IEEE Journal of Solid-State Circuits*, 35(5):780–787, 2000 2000.

Chapter 6

PFD and Charge Pump

The job of the PFD and charge pump is to create a current proportional to the difference in phase between its inputs. Based on the design of the previous chapters, we already have several constraints imposed upon our PFD and charge pump. From the VCO design in chapter 4, since our VCO take a differential control signal, we will use a differential charge pump. Next, because our divider outputs both a 50% and 33% duty cycle square wave depending on the divide ratio of the final 2/3 block, the PFD must be duty cycle insensitive. This suggests the PFD is edge triggered.

6.1 Tristate PFD

The tristate PFD is useful because it uses D flip-flops, which have the twofold benefit of making it duty cycle insensitive, and extending the detection range to 4π. To see how this is, consider the circuit in figure 6-1. Curiously, each latch is clocked on the inputs. With the D input tied high in both cases, the signals UP and DOWN rise when f_{ref} and LO rise, respectively. After both have risen, the NAND gate output goes low, resetting both flip-flops. So the outputs rise with the respective input, but fall together. The resulting pulse-width difference between the UP and DOWN signals is proportional to the phase difference in the edges, with respect to the slower input frequency.

A finer point to consider is the delay in the reset path. Similar to linear class

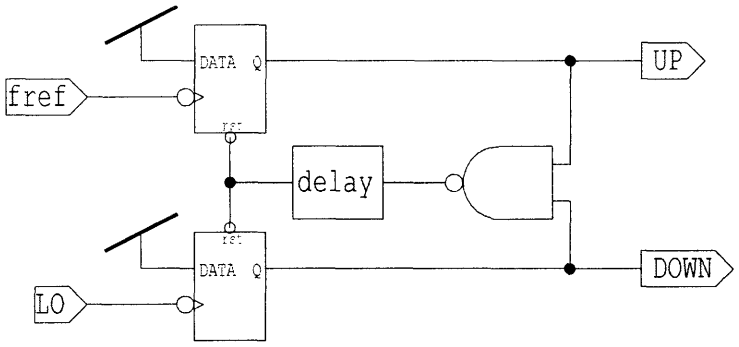

Figure 6-1: The tristate PFD employs latches to widen detection range and eliminate duty cycle sensitivity.

B output stage operation, without some overlap in the UP and DOWN signals, the phase detector characteristic exhibits some crossover distortion. The reason for this distortion lies in the charge pump. Just as transistors have non-zero V_{BE} before turning on, charge pump current switches have finite turn-on and turn-off times. If the UP or DOWN signals are shorter than this time, a non-linear amount of charge is deposited on the loop filter. By inserting a delay in the reset path, we ensure a minimum switch period is observed. Just as with Class B output stages, increasing the overlap reduces distortion at the cost of hotter transistors, or more power waste. Switching power supply pulse-width modulation controllers also allow a minimum and maximum duty cycle for the same reasons.

6.2 Adaptive Bandwidth

As is generally the case with all systems, for a fixed power dissipation we have a trade-off between speed and accuracy. For PLL's, we trade off between settling time and noise rejection, which we will explore in detail in chapter 7. In effort to get the best of both worlds, several approaches have been tried to adaptively control the PLL bandwidth during different modes of operation. There are two common methods of implementing this control; by altering the charge pump current, and changing the loop filter bandwidth.

In [1], a PLL is designed as both the transmitter and receiver synthesizer for a GSM system. Loop bandwidth control is required to optimally meet the specifications of both modes. This is done in two ways: the charge pump has a selectable current, which is equivalent to gain control and the loop filter has switchable capacitor and resistor arrays. While this does enable the PLL to have two different transfer functions, the modes are not switched during operation; it is more of a method to use the same VCO for two applications. In [2], fast lock time is achieved by simultaneously increasing the charge pump current and changing the zero locations in the loop filter. By adjusting the loop filter, the system allows much larger charge pump currents without instability problems. When the divide ratio changes, separate control logic tells the PLL to enter "fast lock" mode for a specified time period, then return to normal bandwidth for noise suppression.

Another approach is to make a PFD with a third output in addition to the UP/DOWN pulses, which signals when the PFD input error has exceeded some threshold. The error signal can then turn on another current source, increasing the charge pump current. This PFD is documented in [3], and a PLL utilizing this PFD is proposed in [4]. A similar approach it to utilize a pair of parallel PFD and charge pumps driving the same loop filter, perhaps coming in on different inputs to the loop filter. By having different loop dynamics for each charge pump, independent control of loop parameters is gained. This approach appears in [5]. A drawback to this strategy is the requirement for more circuitry, increasing chip area and power dissipation.

An interesting method to modulate loop bandwidth in an analog fashion is given in [6]. Here, a switched capacitor filter sets the charge pump current based on the absolute value of the phase error. Unfortunately, this approach may have problems with ripple in the control voltage modulating the charge pump current, as the output of the switched capacitor filter is always setting the bias of the charge pump current source.

The approach we will take is a hybrid method, combining the dual charge pump architecture with a modified PFD. The PFD will send a second set of control signals

to a simple charge pump, which outputs an analog current proportional to the absolute *frequency* difference between the two inputs. This method uses a dual charge pump similar to [5], except the second charge pump is not in the main loop so its design specifications are relaxed, and its power dissipation is lower. We also use analog bandwidth control via charge pump current as shown in [6], except we will implement a dead zone for the second charge pump so that when in lock, the second loop is effectively inactive. This prevents noise from the second charge pump from modulating the source current in the main charge pump while we are in lock. As we will see in section 6.3, this leads to reference spurs in the output.

6.2.1 Frequency Detector

The final PFD design is shown in figure 6-2. The circuitry in the lower left determines which signal is faster. It clocks a flip-flop off the UP and DOWN signals coming from the PFD. Because the PFD's outputs have variable rising edges and synchronized falling edges, we can determine which signal leads another in phase. The pair of inverters in one path helps avoid a race condition when the two inputs are close to lock.

The reset flip-flops used in this design are similar to the flip-flops used in the divider. Because the D input is tied to V_{DD}, we can get away with using only 7 transistors per flip-flop. A schematic of the flip-flop is shown in figure 6-3.

6.2.2 Pulse Based Charge Pump

The subtleties of charge pump design will follow in the next section, but for now assume the charge pump shown in figure 6-4 gets the job done.

The novelty of this charge pump are the pulse circuits before each switch. By converting a square wave into a fixed-width pulse train, we now have a wave whose average value is proportional to the input frequency. If we always input the faster wave into the sinking current source, we will be pulling a current through the PMOS/Capacitor parallel combination proportional to the absolute frequency difference between the

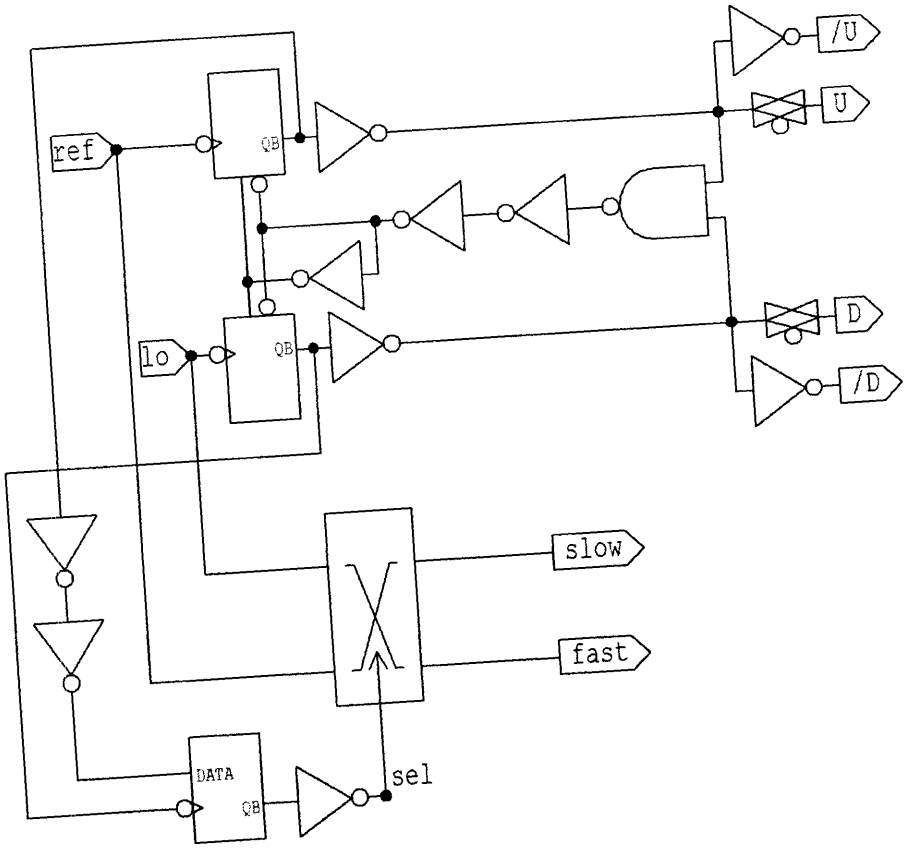

Figure 6-2: Note the inclusion of dummy delay pass gates to better match pulse arrival times at the charge pump.

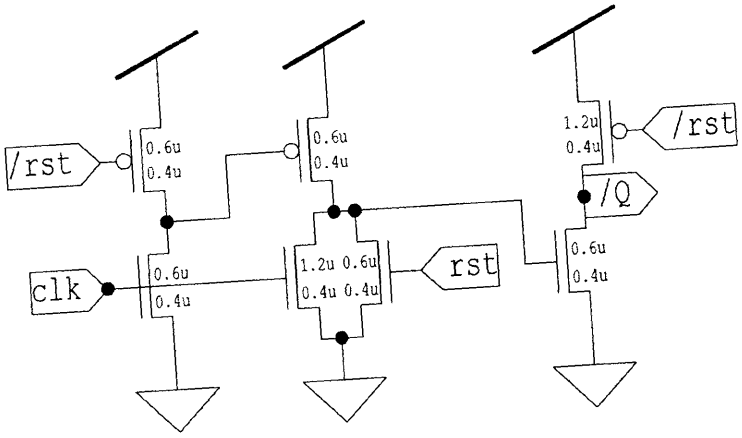

Figure 6-3: Flip-flops used in the PFD with reset signal and input tied to V_{DD}.

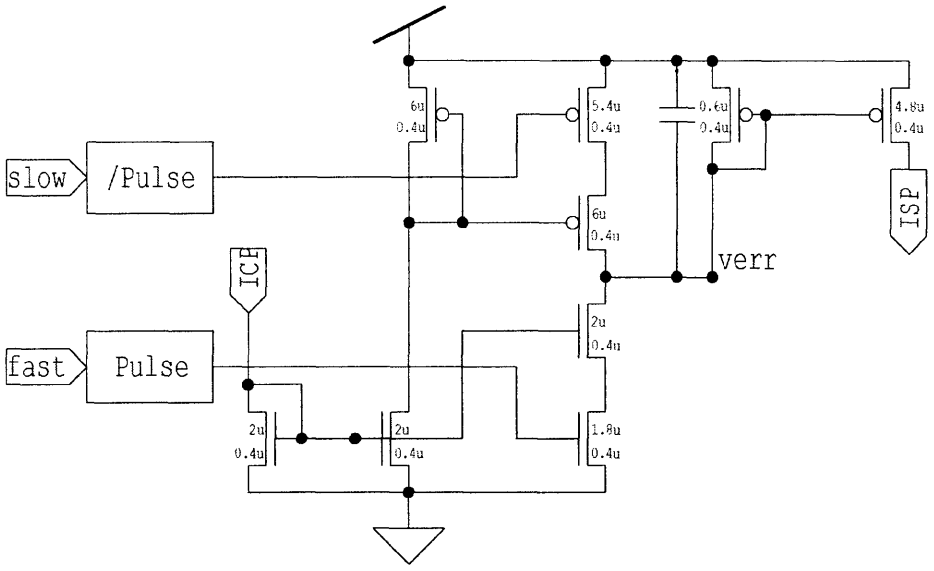

Figure 6-4: Charge pump which converts frequency difference into current.

two inputs. This current is then mirrored over to another PMOS, which will become a source transistor for the main charge pump. The concept of pulse-based computing originates from spiking neuron circuits, see [7].

The circuit which implements the spike is shown in figure 6-5. The pulse width is determined by the size of the diode connected transistor in the first leg in relation to the capacitor size. Again because precise matching is not critical in this section, small devices may be used.

6.3 Charge Pump Considerations

While a charge pump is conceptually simple, the implementation poses several challenges which make good charge pump design difficult. As a result, the few good ones are reused frequently across designs. Let us think of a few reasons why a charge pump is so hard to design.

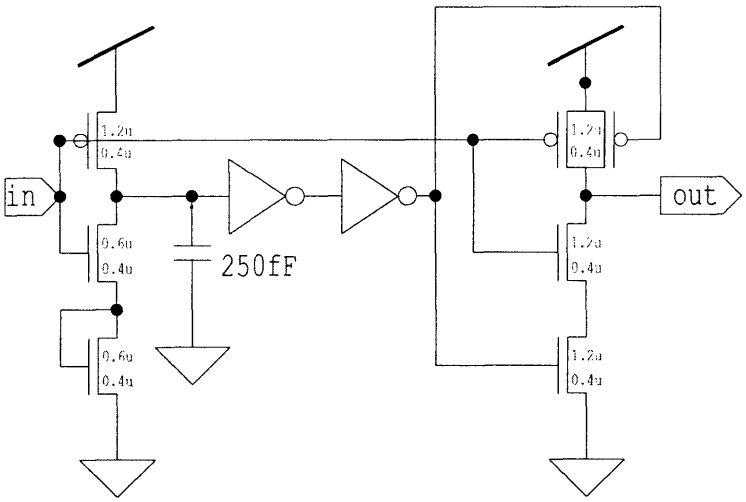

Figure 6-5: A self-biased circuit which generates fixed width pulses.

6.3.1 Current Mismatch

As we saw in figure 1-4, a charge pump is just two switched current sources. Here we want to know what happens if I_{UP} is not equal to I_{DOWN}. In locked condition, the control voltage to the VCO is constant, so the average capacitor charging current is zero. thus,

$$t_{up}I_{UP} - t_{down}I_{DOWN} = 0 \tag{6.1}$$

Suppose we have current source mismatch, so $I_{DOWN} = I_{CP} = I_{UP} + \Delta I$. Then we write,

$$t_{up}\left(I_{CP} + \Delta I\right) - t_{down}I_{CP} = 0 \tag{6.2}$$

This leaves us with non-equal up and down pulse times,

$$t_{down} - t_{up} = \frac{\Delta I}{I_{CP}} t_{up} \tag{6.3}$$

since we just argued that t_{up} has a minimum pulse width to prevent crossover distortion. To get the corresponding phase error for this pulse difference, we need to think about what K_d is for the tristate PFD and charge pump combination. We saw that the tristate PFD has detection range from -2π to 2π, so it is natural to say that

for those extreme phase errors, the charge pump is sourcing or sinking its maximum current, I_{CP}. Therefore $I_{OUT} = I_{CP}/2\pi\Delta\theta$, or

$$K_v = \frac{I_{OUT}}{\Delta\theta} = \frac{I_{CP}}{2\pi} \tag{6.4}$$

Returning to equation 6.3,

$$\Delta\theta = \frac{t_{down} - t_{up}}{\frac{1}{f_0}} = 2\pi f_0 \frac{\Delta I}{K_d} t_{up} = \omega_0 \frac{\Delta I}{K_d} f_{up} \tag{6.5}$$

This equation is helpful because it tells us what design parameters we should change to minimize the effect of a constant mismatch. Other than increasing the PFD gain, there is a trade-off between minimizing crossover distortion, and allowing mismatch based phase error. Not only does it lead to static phase error, it produces a sideband in our phase noise. If we use the noise transfer functions from section 1.3.2 along with equation 3.41 we get the result shown in [8],

$$\mathcal{L}(\Delta f_{ref}) = 20 \cdot \log\left[\frac{1}{\sqrt{2}} \cdot \frac{f_{BW}}{f_{ref}} \cdot N \cdot \Delta\theta\right] - 20 \cdot \log\left[\frac{f_{ref}}{f_P L}\right] \tag{6.6}$$

Where f_{BW} is the loop filter bandwidth, N is the average division ratio in a $\Sigma\Delta$ PLL, and f_{PL} is the dominant pole frequency. This will be discussed in more detail in chapter 7.

6.3.2 Output Resistance

All real MOSFETs have finite output resistance. This can be modeled as two resistors shunting the current sources, as shown. In the ideal case, $I_{UP} = I_{DOWN} = I_{CP}$. With the resistors in place, the actual current is modulated by the voltage at the node, V_{CP}. We get the following equation for actual charge pump current, in lock:

$$t_{up} \cdot \left(I_{CP} + \frac{V_{DD} - V_{CP}}{r_{up}}\right) - t_{down}\left(I_{CP} + \frac{V_{CP}}{r_{down}}\right) = 0 \tag{6.7}$$

Since the output resistance of a MOSFET is proportional to drain current, it is

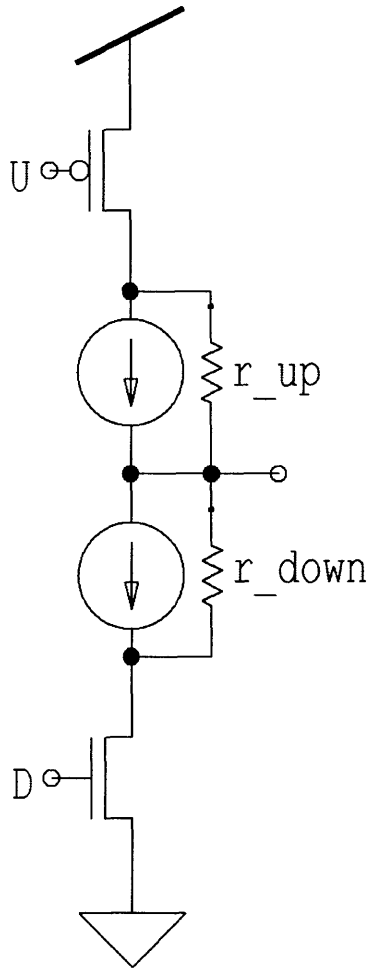

Figure 6-6: The charge pump has current sources with finite output resistance.

safe to assume $r_{up} = r_{down} = r_o$, so we simplify to get

$$(t_{up} - t_{down})I_{CP} - \frac{t_{up}V_{DD} - (t_{up} + t_{down})V_{CP}}{r_o} = 0 \tag{6.8}$$

Thus, the only way $t_{up} = t_{down}$ is if $V_{CP} = V_{DD}/2$. Otherwise, we get a finite phase error as well as fractional spurs, with an equation similar to 6.6.

Recalling $r_o = \frac{1}{\lambda I_D}$, rearranging gives us

$$\frac{t_{up} - t_{down}}{t_{up}V_{DD} - (t_{up} + t_{down})V_{CP}} = \lambda \tag{6.9}$$

Unlike current source mismatch error, which can be drowned out with an increase in current, we must reduce λ for greater performance. This can only be achieved by making devices longer, or cascoding.

6.3.3 Switch Issues

We will now investigate the previously alluded to minimum turn on and turn off switching time. Before we can do that, how should we implement the switches? Three basic configurations are source switching, drain switching, and gate switching depending on where the switch is placed in relation to the current source transistor. A diagram of this is shown in figure 6-7.

The drain switched charge pump has many negative parasitic effects. Because the source voltage of the switching transistors changes as the switch turns on, large current spikes are generated on each transition. Furthermore, turning the switch on and off causes the current source transistor to move in and out of saturation, modulating the charge pump current. The drain-bulk capacitances of the source transistors also affect the switching speed since the switches have floating sources. Finally, because the switches are so close to the output, clock feedthrough and charge injection is a large problem.[9].

An alternative is switching at the source. Because the current source transistors are always in saturation, charge sharing is smaller than the switch at the drain topol-

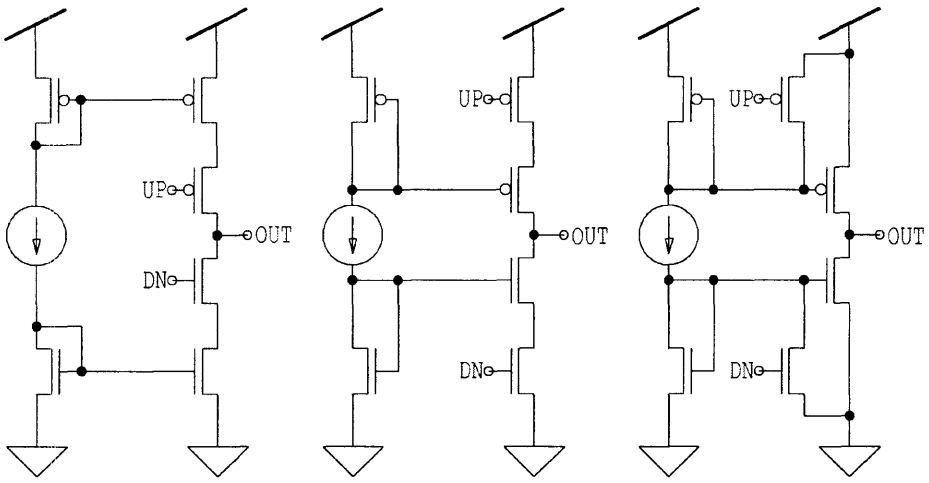

Figure 6-7: Three common single-ended charge pump switch topologies: drain switched, source switched and gate switched. Note that the current source implementation is unrealistic.

ogy. Also, gate charge injection is less of a problem, because the switch has a greater overdrive voltage.[10]. Also note that the drain bulk capacitance problem of the drain switched topology has been eliminated.

The third basic topology is grounding the gate of the current source transistors, turning them on and off. An advantage here is extremely low injection problems, as well as the widest output swing of the three topologies. A drawback is the switching speed. For matching purposes, the current source transistors will be very large. Large gates lead to high node capacitances, which need to be charged and discharged on each transition. Increasing node capacitance for a fixed V_{DD} and frequency lead to increased power consumption.

Current Steering Switches

The speed problems of the previous topologies are all due to the need to swing a large voltage across a capacitance. One way around large voltage swings is to use current-mode switches, by replacing single switches with differential pairs. In this topology, the node voltage swing is minimal, and hence switching speed is increased. The drawback is a constant power dissipation, even in lock, through the unused

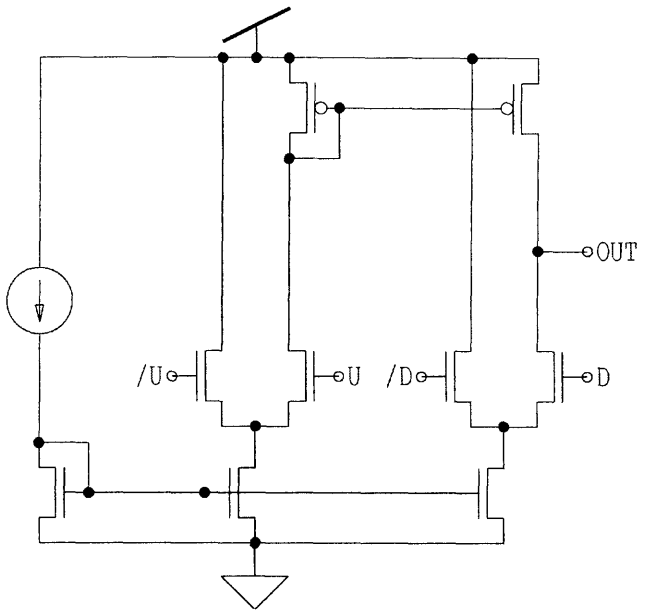

Figure 6-8: A charge pump that uses current-mode logic and NMOS only switches.

differential transistor. Perhaps the biggest advantage to this topology is the ability to use switches of all one type. This leads to a better match in switching time. In the case where we had complimentary switches, the PMOS device is always two to three times larger than an equivalent NMOS device. The PMOS's increased gate capacitance makes the pull-up path slower than the pull-down path. The current-mode NMOS only charge pump is shown in figure 6-8.

While this is a good idea, a large problem lies in the mismatched current path between the UP and DOWN signals. The DOWN current is pulled directly from the OUT node, whereas the UP current is reflected across with a PMOS mirror. Although current mirrors are fast, the delay is significant enough to cause output spurs.

6.3.4 Differential Topologies

It is possible to eliminate many of the mismatch problems mentioned above with a differential topology. The costs are increased power consumption and substantially greater chip area, both due to increased circuit complexity and the need for two loop filters.

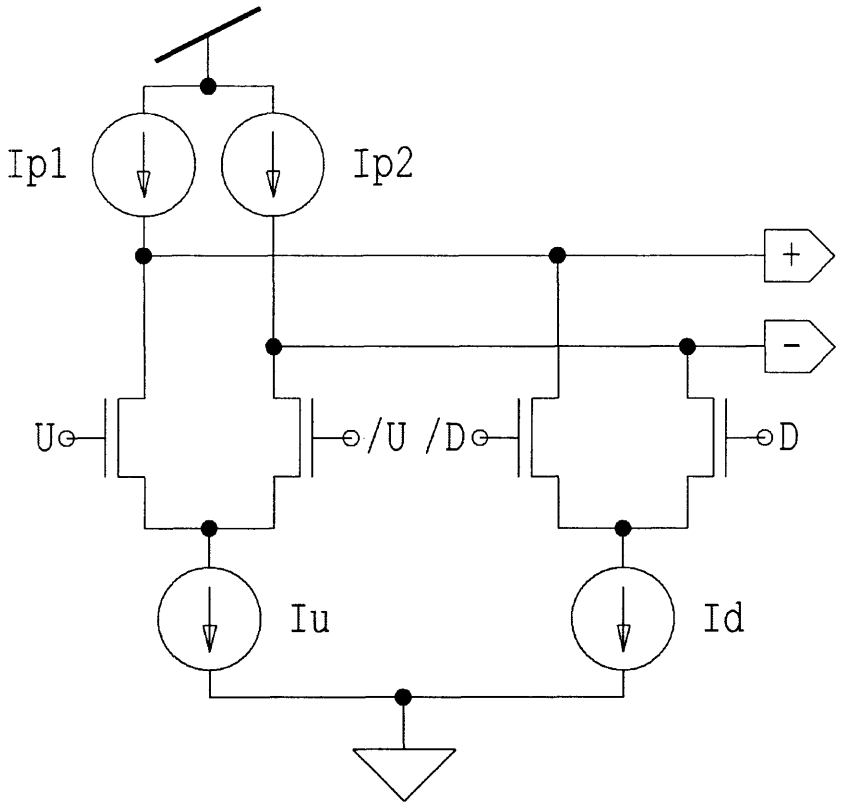

Figure 6-9: The simplest differential current steering charge pump.

Consider the charge pump shown in figure 6-9.

We are using all NMOS switches in a current steering configuration. This carries all the benefits of the single ended case, with the added benefit that the UP path is now identical to the down path. The differential topology has alleviated the need for a PMOS mirror. Furthermore, each switch has the same source voltage, so the switching times are well matched. This helps mitigate the impact of drain switching, which we previously determined was inferior to source switching. By having all parameters as well matched as possible, the large current spikes resulting from drain switching are hopefully matched as well, and are therefore a common mode effect.

The common mode is something we need to address in the differential case, which was not present in the single ended charge pumps. For completeness let us write out the differential and common mode currents flowing for each set of inputs.

We note that the common mode current is identical in all cases, and is equal

Table 6.1: Charge Pump Current Matrix

		$\overline{U}$	U
$\overline{D}$	V_{DIFF}	$\frac{1}{2}(I_{P1} - I_{P2}) + \frac{1}{2}(I_U - I_D)$	$\frac{1}{2}(I_{P1} - I_{P2}) - \frac{1}{2}(I_U + I_D)$
	V_{CM}	$\frac{1}{2}(I_{P1} + I_{P2}) - \frac{1}{2}(I_U + I_D)$	$\frac{1}{2}(I_{P1} + I_{P2}) - \frac{1}{2}(I_U + I_D)$
D	V_{DIFF}	$\frac{1}{2}(I_{P1} - I_{P2}) + \frac{1}{2}(I_U + I_D)$	$\frac{1}{2}(I_{P1} - I_{P2}) - \frac{1}{2}(I_U - I_D)$
	V_{CM}	$\frac{1}{2}(I_{P1} + I_{P2}) - \frac{1}{2}(I_U + I_D)$	$\frac{1}{2}(I_{P1} + I_{P2}) - \frac{1}{2}(I_U + I_D)$

to the difference between the average current sourced into the loop filter, minus the average current pulled from the loop filter. We can therefore implement a common mode feedback (CMFB) loop to regulate these two currents.

6.4 Charge Pump Design

The final charge pump design is shown in figure 6-10.

We have used the differential current steering topology. To combat the problem of output resistance, we use wide-swing cascoding.[10, 11] This technique allows the maximum voltage swing by biasing the cascode transistors with a current dependent voltage, so that the source transistors are always on the edge of saturation. The cascode voltages are shunted with a capacitor to the supply, to help eliminate noise feedthrough from the variable I_{CTL} and CMFB circuit.

6.4.1 Common Mode Feedback Circuit

The CMFB design chosen is a switched capacitor (SC) type. In terms of fully differential op-amps, designers usually use SC CMFB when the op-amp will be used for a SC circuit. The big advantage of the SC CMFB variety is a huge common mode voltage range. Although a charge pump falls under the SC circuit category, we can not directly use SC CMFB such as shown in [10], because the circuit draws a DC current from the outputs of the charge pump. Considering the great lengths we have taken to eliminate DC leakage, this is counterproductive. Another possibility is to implement the loop filter resistor with the CMFB circuit. While this looks promising

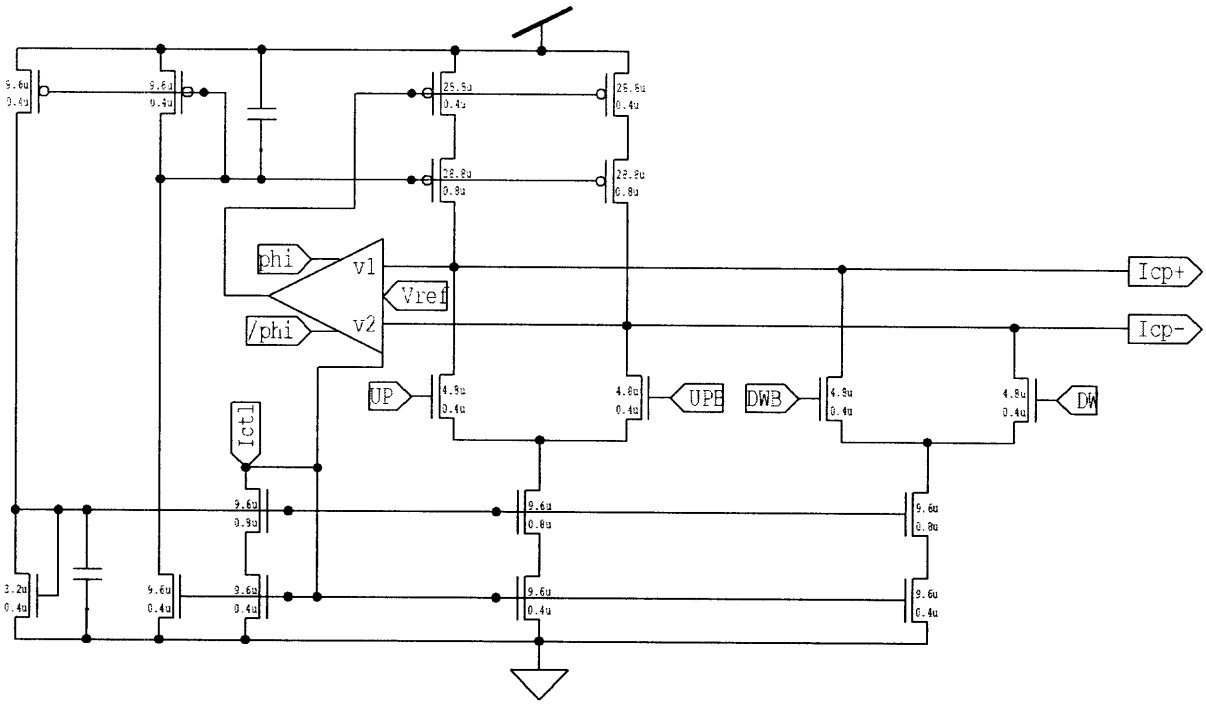

Figure 6-10: The final charge pump design with common mode feedback circuit.

at first, the loop filter must have excellent noise performance which is one of the reasons why we would like to avoid active loop filters.[1] To achieve comparable noise performance from a SC resistor would require an inconveniently high clock frequency, and result in large power dissipation.

Implementing the SC CMFB as shown in figure 6-11, we have opted to buffer the charge pump voltage with source followers prior to the SC circuit. While this does cut into our voltage swing, our swing is already reduced by the active cascoding, and thus not the limiting path.

The switched capacitors sample the voltage at the source follower at phase one, and average the charge onto the middle capacitor at phase two. This common mode voltage is compared with the reference voltage, which also experiences a voltage drop of one v_{gs} with the same bias current. The differential pair is set up in a low gain configuration. To stabilize the loop, a lower gain is desired; however the low gain leads

[1] See chapter 7

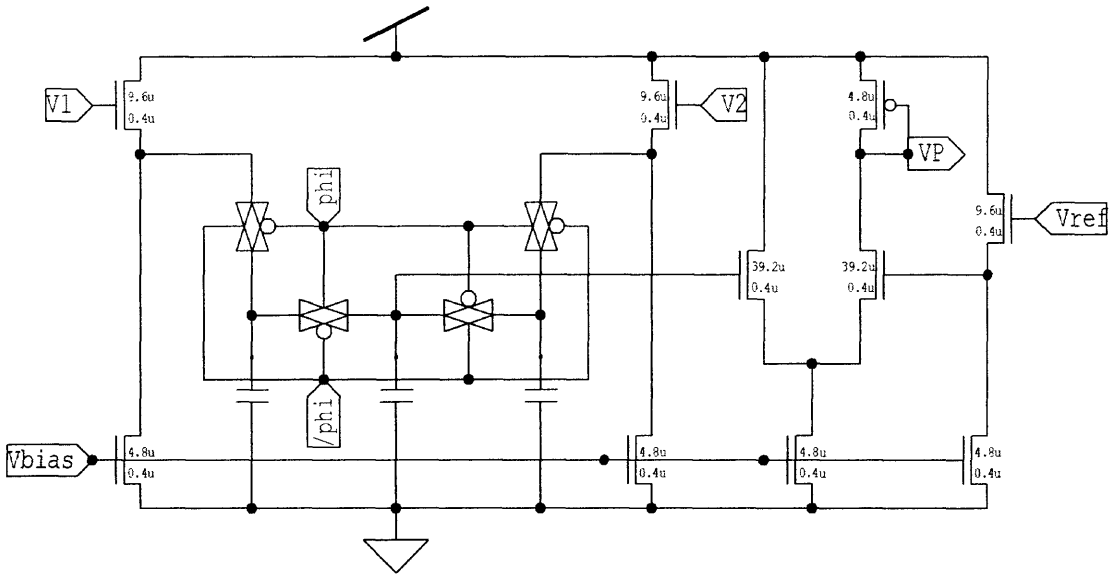

Figure 6-11: The common mode feedback circuit uses source follower buffers driving the switched capacitor circuit.

to 30mV of DC error between the reference voltage and the common mode output. Fortunately, the VCO is insensitive to this low common mode offset. Figure 6-12 is a plot showing the stability of the common mode loop.

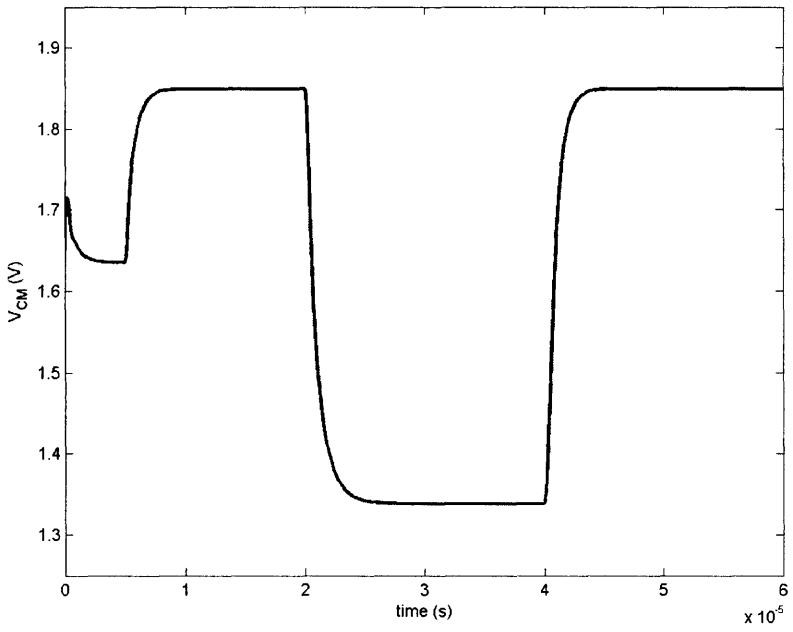

Figure 6-12: The fast rise time indicates sufficiently high bandwidth for our purpose, and the lack of peaking indicates good phase margin.

Bibliography

[1] Emad Hegazi and Asad Abidi. A 17-mw transmitter and frequency synthesizer for 900-mhz gsm fully integrated in 0.35μm cmos. *IEEE Journal of Solid-State Circuits*, 38(5):782–792, May 2003.

[2] M. Keaveney, Patrick Walsh, Mike Tuthill, Colin Lyden, and Bill Hunt. A 10μs fast switching pll synthesizer for a gsm/edge base-station. In *Proceedings of the 2004 IEEE International Solid-State Circuits Conference*, number 10, February 2004.

[3] P. Larsson. Reduced pull-in time of phase-locked loops using a simple nonlinear phase detector. In *IEEE Proceedings on Communication*, volume 142, pages 221–548, August 1995.

[4] Ching-Yuan Yang and Shen-Iuan Liu. Fast-switching freuquency synthesizer with a discriminator-aided phase detector. *IEEE Journal of Solid-State Circuits*, 35(10):1445–1452, October 2000.

[5] Cicero S. Vaucher. An adaptive pll tuning system architecture combining high spectral purity and fast settling time. *IEEE Journal of Solid-State Circuits*, 35(4):490–502, April 2000.

[6] Joonsuk Lee and Beomsup Kim. A low-noise fast-lock phase-locked loop with adaptive bandwidth control. *IEEE Journal of Solid-State Circuits*, 35(8):1137–1145, August 2000.

[7] Rahul Sarpeshkar, Lloyd Watts, and Carver Mead. Refractory neuron circuits. Computation and Neural Systems Technical Report CNS-TR-92-08, California Institute of Technology, Pasedena, California, 1992.

[8] Woogeun Rhee. Design of high-performance cmos charge pumps in phase-locked loops. In *Proceedings of the 1999 IEEE International Symposium on Circuits and Systems*, number 2, pages 545–548, May 1999.

[9] Mohamad El-Hage and Fei Yuan. Architectures and design considerations of cmos charge pumps for phase-locked loops. *Canadian Conference on Electrical and Computer Engineering*, 1:223–226, May 2003.

[10] David Johns and Ken Martin. *Analog Integrated Circuit Design.* John Wiley & Sons, New York, NY, third edition, 2004.

[11] William S. T. Yan and Howard C. Luong. A 2-v 900mhz monolithic cmos dual-loop frequency synthesizer for gsm receivers. *IEEE Journal of Solid-State Circuits*, 37(8):1012–1020, August 2002.

Chapter 7

Loop Filter

Up to this point, the components we have designed all have fairly boring transfer functions. The VCO is just a gain plus integrator, the charge pump and PFD is a gain, and the divider is another gain. The loop filter is the only block whose transfer function is completely up to us, and therefore strongly determines the closed loop transfer function of the whole PLL. Let us begin by categorizing some commonly used loop filters. Before we do this, we need to define the type of the filter. From servomechanism terminology, the loop type specifies the number of perfect integrators in the loop.[1]. The VCO has an integrator, so a PLL is at least a type I system. From feedback theory, we know that adding integrators to a system simultaneously removes steady state error for increasingly high order inputs, and makes the loop harder to compensate.[2] We will focus on Type II PLL's. These have zero steady state phase error to an input step in frequency. For land-based wireless applications, this is all we need. Type III and higher order PLL's have found uses in satellite communication systems, where zero phase error in response to a frequency ramp is needed. This arises from Doppler shifting, a phenomenon hard to achieve on land.

The next term to define is the PLL order. This is simply the order of the PLL loop transfer function. Thus, the loop filter order is one less than the order of the PLL.

One issue related to charge pump based PLL's is the delay that comes between pulses from the PFD. The charge pump acts like a sample-and-hold, and there are

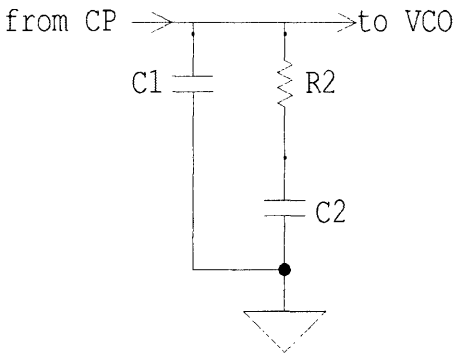

Figure 7-1: Capacitor with shunt lag network.

stability concerns regarding such systems. It turns out that as long as the loop bandwidth is two or more decades below the reference frequency f_{ref}, we can use the average charge pump current. Designs with lower higher bandwidth to reference ratios are also acceptable, but the added phase shift from the sampling can not be neglected.

7.1 Third Order Type II PLL

A type II PLL will have one integrator in the loop filter, and one from the VCO. Because we are using a charge pump, the charge pump capacitance already provides this second integration for us. However, this can not be the whole loop filter, for a system with two poles at zero is unstable. We need a zero in the left half plane to draw the poles out. This can be implemented by shunting the integration capacitor with a lag series capacitor and resistor. A schematic for the loop filter is shown in figure 7-1, and the corresponding root-locus for the whole PLL is shown in 7-2.

The transfer function from charge pump current to VCO control voltage is given by the impedance of the filter,

$$Z(s) = \frac{R_2 C_2 s + 1}{s(R_2 C_1 C_2 s + C_1 + C_2)} \tag{7.1}$$

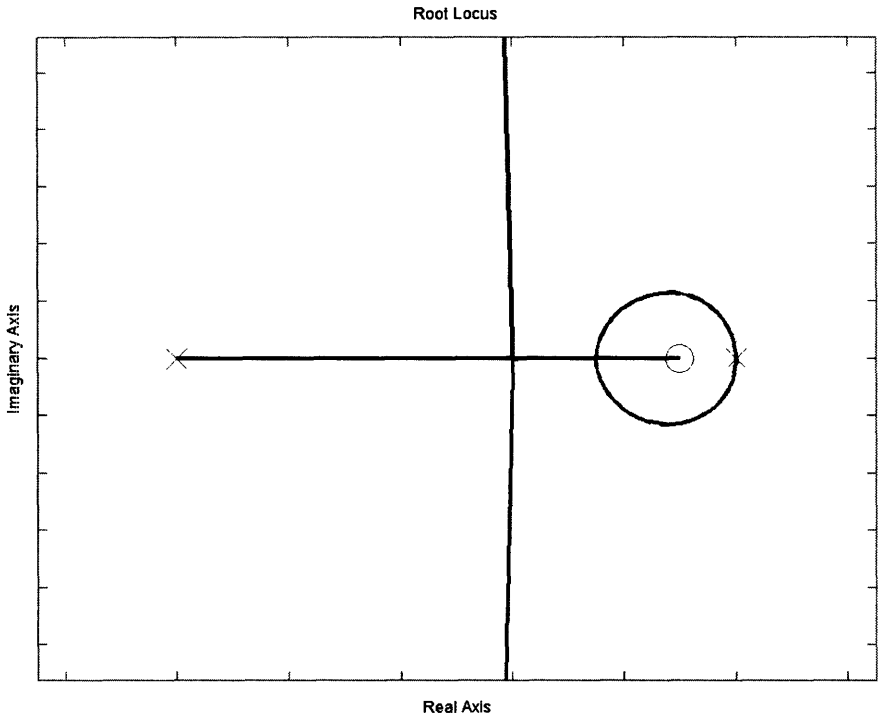

Figure 7-2: The addition of a loop-stabilizing zero draws the double pole at zero out into the left half-plane.

7.1.1 Zero Location Trade-Off

From the root-locus plot and the impedance, we see it is necessary to have $C_2 \gg C_1$, so that the added pole is sufficiently far out to have good phase margin. When the singularities are so separated, the zero is at R_2C_2, and the pole is at $R_2(C_1||C_2) \approx R_2C_1$. Unfortunately, along with stability, lag compensation brings pitfalls. From figure 7-2, there are loop gains for which the zero is the lowest frequency singularity. This causes an increase in closed loop gain, until the additional poles bring the gain back down. This region of greater-than-unity gain results in jitter peaking. If an input has some phase noise in the band of greater-than-unity gain, the PLL will amplify this phase noise. In systems with cascaded PLL's, this is problematic.

The simple solution to this is to increase loop gain further, driving the pole closer to the zero and thus narrowing the band of greater-than-unity gain. Unfortunately, this leads to a pole-zero doublet, which we know causes excessively long settling times. Thus our job is to trade-off between jitter peaking and settling time.

A more complex solution it to implement the zero-producing resistor artificially using delay lines. This method avoids the pole-zero doublet, and removes the settling-peaking relationship. For an example, see [3].

7.2 Fourth Order Type II PLL

To further attenuate high frequency noise from the charge pump switching, we can cascade a first order $R-C$ network after the previously investigated second order loop filter. This third order loop filter still has only one integrator from C_1, so we have a fourth order type II PLL.

The complete transfer function for this loop filter is the following:

$$Z(s) = \frac{R_2C_2s+1}{s(R_2R_3C_1C_2C_3s^2+(R_2C_3+R_3C_3+R_2C_1+R_3C_1)C_2s+(C_1+C_2+C_3))} \tag{7.2}$$

Clearly this equation is too messy to gain any useful intuition. One condition which makes analysis more tractable is to specify that the R_3C_3 pole is much higher

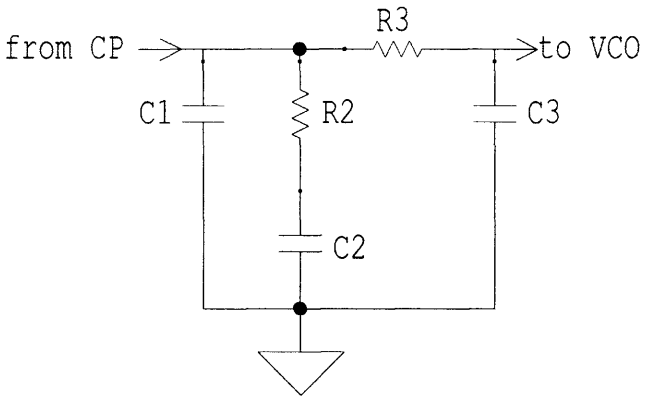

Figure 7-3: A second order loop filter with the addition of an $R - C$ network.

than the poles of the second order filter. This ensures the added $R - C$ leg has minimal loading effect on the previous stage, and can thus be treated as the cascade of a second order loop filter with an $R - C$ filter. Detailed analysis shows that a fast R_3C_3 pole is not just a convenience, it is a necessity to ensure loop stability. For the complete analysis, see [4].

7.3 GSM900 Transient Specification

As we mentioned in chapter 3, GSM900 uses a TDMA protocol. This means the synthesizer must be able to switch between frequencies in a given time, which sets an upper bound on settling time. The transient settling time is specified as within 10ppm in 150μs, however we will strive for 0.1ppm in the same time interval, which is something competitive synthesizers achieve.[5]

7.4 Loop Filter Design

An alternative to paper design is using a behavioral simulation tool. Unlike circuit level analysis which takes too long to be useful in an iterative design flow, a behavioral simulator is fast enough to be run multiple times to aid in parameter specification. One such simulator is Michael Perrott's CppSim.[6] In order to use this tool, we need

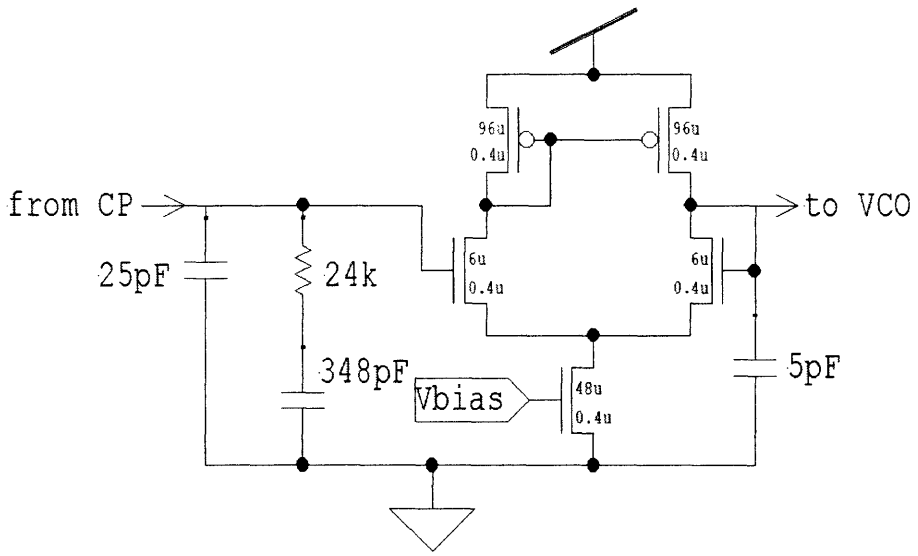

Figure 7-4: The g_mC filter provides isolation between filter sections, so that the higher order pole does not load the lower poles.

to decide upon the specifications of our PLL. For now, we will not explicitly address the phase noise specifications and simply require that the output phase noise be lower-bounded by the VCO phase noise, the best we can do. We begin with loop filter design by using the PLL Design Assistant to locate our poles and zeros. Note that for loop filter design we are ignoring detector noise, because the circuit neglecting the VCO is similar to other works, and the ring oscillator VCO will have worse performance than resonant VCO's in other PLL's. For our reference frequency we will use 26MHz, which is standard.

Given the above parameters, the loop filter values we get are a gain K of $2.677 \cdot 10^9$, a zero at 18.75KHz, a pole at 271.3KHz, and a second pole at 1MHz.

We will have difficulty placing these singularities with a completely passive implementation and stay within component values reasonable for on-chip integration. Therefore we have no choice but to use a second order loop filter, figure 7-1, followed by a g_mC OTA lowpass filter. The loop filter schematic is shown in figure 7-4.

We first note that in reference to figure 7-1 $K = C_1 + C_2$, and $R_2C_2 = f_z$. Also,

$$C_1 = \frac{C_2}{2\pi R_2 C_2 f_p - 1} \tag{7.3}$$

The exact values can therefore be found with some iteration in MATLAB. As for the g_mC filter section, we know

$$\frac{g_m}{C} = 2\pi \cdot 1\text{MHz} \tag{7.4}$$

We purposely chose large devices for two reasons, one to ensure maximum common mode swing and the other to let us use a large C. Recall that the noise of this g_mC filter is proportional to $\frac{kT}{C}$, independent of g_m. Larger C values help offset the presence of the active devices in our loop filter by reducing our noise at the cost of more power. If we choose $C = 5\text{pF}$, the required tail current is 800nA.

Bibliography

[1] Floyd M. Gardner. *Phaselock Techniques.* Wiley Interscience. John Wiley & Sons, New York, NY, second edition, 1979.

[2] James K. Roberge. *Operational Amplifiers.* John Wiley & Sons Inc., New York, NY, first edition, 1975.

[3] Tai-Cheng Lee and Behzad Razavi. A stabilization technique for phase-locked frequency synthesizers. *IEEE Journal of Solid-State Circuits*, 38(6):888–894, May 2003.

[4] An analysis and performance evaluation of a passive filter design technique for charge pump pll's. Application Note 1001, National Semiconductor, July 2001.

[5] Michael H. Perrott. *Fractional-N Frequency Synthesizer Design Using the PLL Design Assistant and CppSim Programs.* MIT Microsystems Technology Laboratory, July 2004.

[6] Michael H. Perrott. *PLL Design Using the PLL Design Assistant Program.* MIT Microsystems Technology Laboratory, July 2004.

Chapter 8

Performance

8.1 GSM900 Frequency Range

The GSM900 specification calls for 124 channels of 200KHz bandwidth each, spaced at 200Khz whose frequency is determined by the equation

$$f_{RF} = 935.2 + 0.2(N-1)\text{MHz} \qquad \text{for} \quad N = 1, \ldots, 124 \tag{8.1}$$

This received signal is downmixed by the local oscillator to an intermediate frequency (IF) of 70MHz.[1] Thus, the corresponding local oscillator frequency is given by

$$f_{LO} = 865.2 + 0.2(N-1)\text{MHz} \qquad \text{for} \quad N = 1, \ldots, 124 \tag{8.2}$$

8.2 VCO Performance

These performance results refer to the VCO designed in chapter 4, whose schematic is shown in 4-7.

8.2.1 Tuning Range

We obtain the K_V characteristic from the oscillator by plotting frequency versus control voltage, in figure 8-1. We have obtained a cubic equation representing K_V

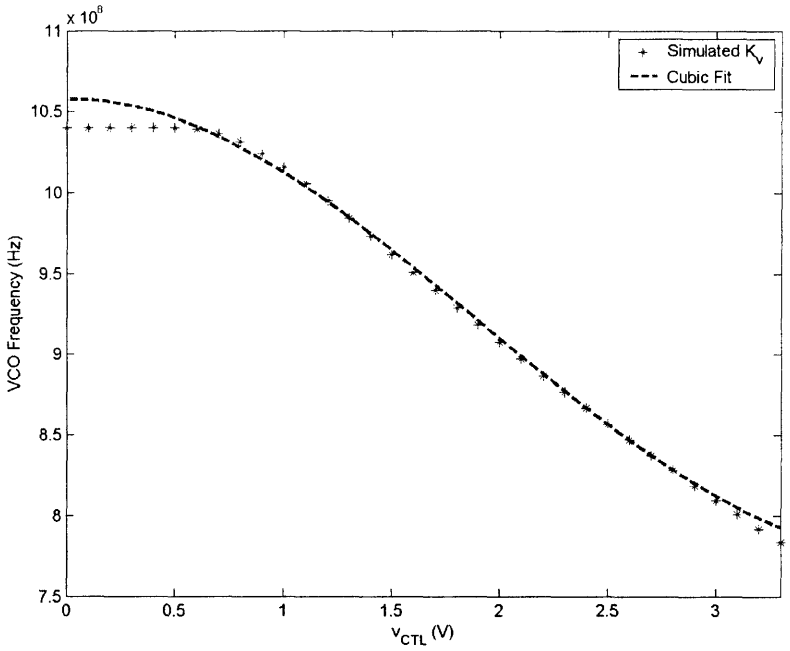

Figure 8-1: The oscillator's control voltage characteristic. Because this oscillator has differential control lines, the other node is at $V_{DD} - V_{CTL}$.

versus V_{CTL} for accurate simulation, however for most of the frequency range of interest $K_V = -102MHz/V$. We see that we have covered both the transmitter frequency range and the local oscillator frequency range for GSM900.

8.2.2 Phase Noise

To evaluate the phase noise performance of the VCO, we need to determine $\frac{\overline{i_n^2}}{\Delta f}_{pk}$, $\alpha(\tau)$, and $\Lambda(\tau)$. $\Lambda(\tau)$ is fairly easy to determine by method one given in [2]. We simulate the oscillator for a few periods of oscillation, injecting a current into one node at a variable time and measuring the phase shift. The following Λ was obtained for $\omega_0 = 900MHz$, figure 8-2.

Current Noise Normalization

As we discussed in chapter 2, equation 2.12 contains a parameter γ which increases with short channel devices. To estimate γ, a ring oscillator was replicated from

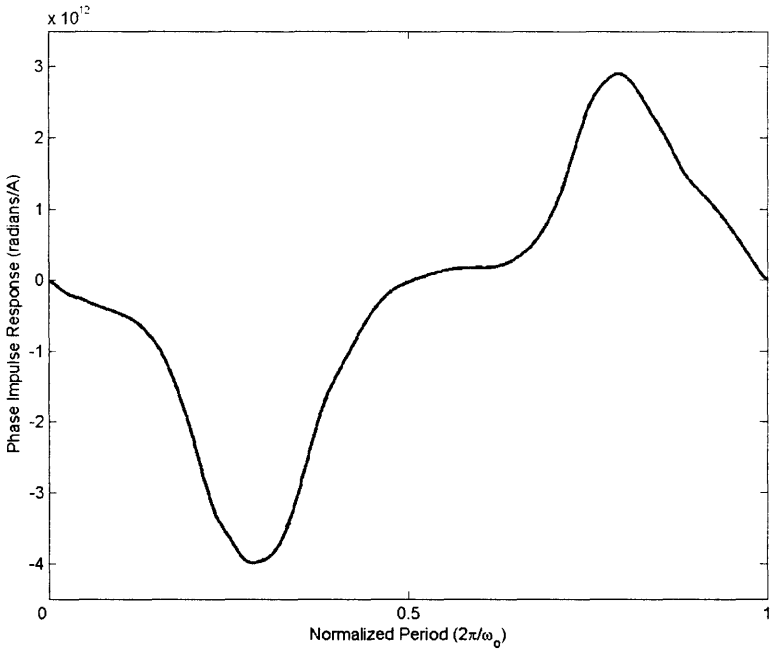

Figure 8-2: Λ over one period.

one presented in [3], and γ was adjusted until our simulated results matched their experimental results. With this value of γ, we then computed the total device noise into one node of our oscillator over one period of oscillation, shown in figure 8-3.

8.2.3 Total Phase Noise

Combining the results from the previous two sections and using equation 3.50 with $N = 8$, we get a plot of the phase noise in the $1/f^2$ region in figure 8-4. We have not included the effect of $1/f$ noise, because of the lack of reliable $1/f$ noise parameters for our process. However, using equation 3.49, we find that the relation between device corner and phase noise corner is

$$\frac{\omega_{1/f^3}}{\omega_{1/f}} = 0.0053 \tag{8.3}$$

This almost certainly puts the $1/f^3$ corner within the bandwidth of the loop filter, where it will be attenuated by the PLL.

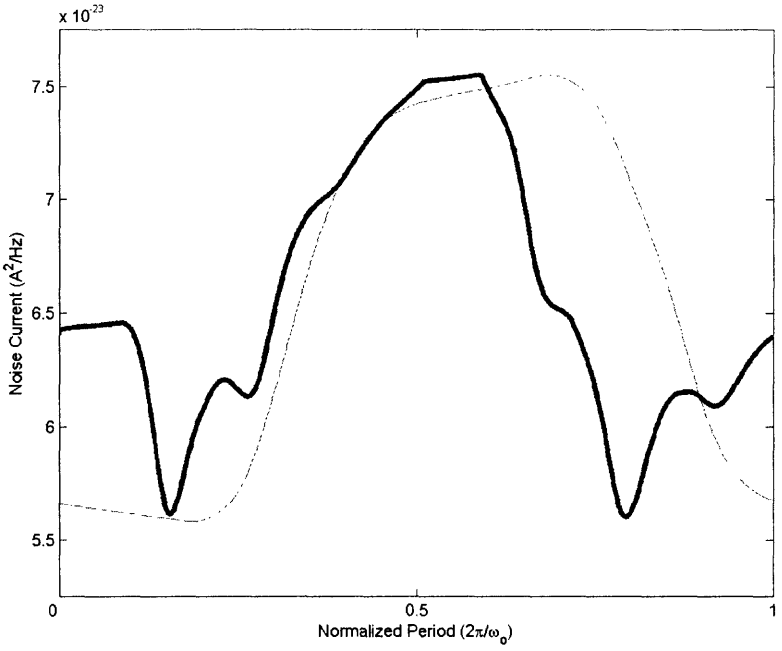

Figure 8-3: Amplitude of the mean square current noise. The faint line is the oscillator's node voltage at the corresponding time.

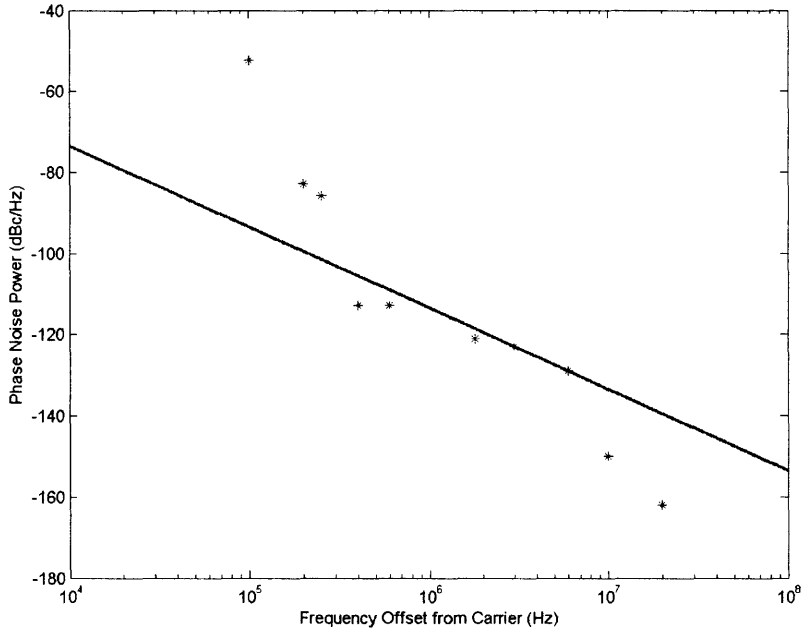

Figure 8-4: Oscillator phase noise. The stars represent the GSM900 phase noise specification.

As shown in figure 8-4, for higher frequencies the VCO does not meet GSM specification. However, this oscillator offers comparable performance to that presented in [4] at lower power dissipation, even when process scaling is taken into account. The power dissipation of this oscillator is 5.19mW from a 3.3V supply at 900MHz.

8.3 Divider Performance

The divider designed in chapter 5 is shown in figure 5-12. The divider consumes 5.14mW from a 3.3V supply, with a divide ratio of 32. With a divide ratio of 63, power consumption increases to 5.34mW, again from a 3.3V supply. The maximum operating frequency is 1.2GHz.

8.4 PLL Dynamic Behavior Simulation

We have simulated the whole PLL using CppSim, including the noise sources from the PFD, charge pump, and active loop filter. We are using a third order $\Sigma\Delta$ modulator to feed the divider input. A transient simulation for small signal frequency steps is shown in figure 8-5.

To simulate the PLL's large frequency step behavior, a custom PLL simulation was written using MATLAB, documented in appendix B. When the divider changes divide ratios too quickly, the phase detector's output exceeds the first modulus of 2π, and the PLL experiences what is called cycle slipping. In figure 8-6 we see that the adaptive loop bandwith controller described in chapter 6 indeed improves the transient behavior for such large steps.

8.5 PLL Phase Noise Simulation

Using the same simulation tool as the transient response, we have the complete phase noise plot in figure 8-7. The charge pump has a 0.8ns reset pulse, and contributes a total of $5.7 \cdot 10^{-23} \frac{A}{Hz}$ of current noise.

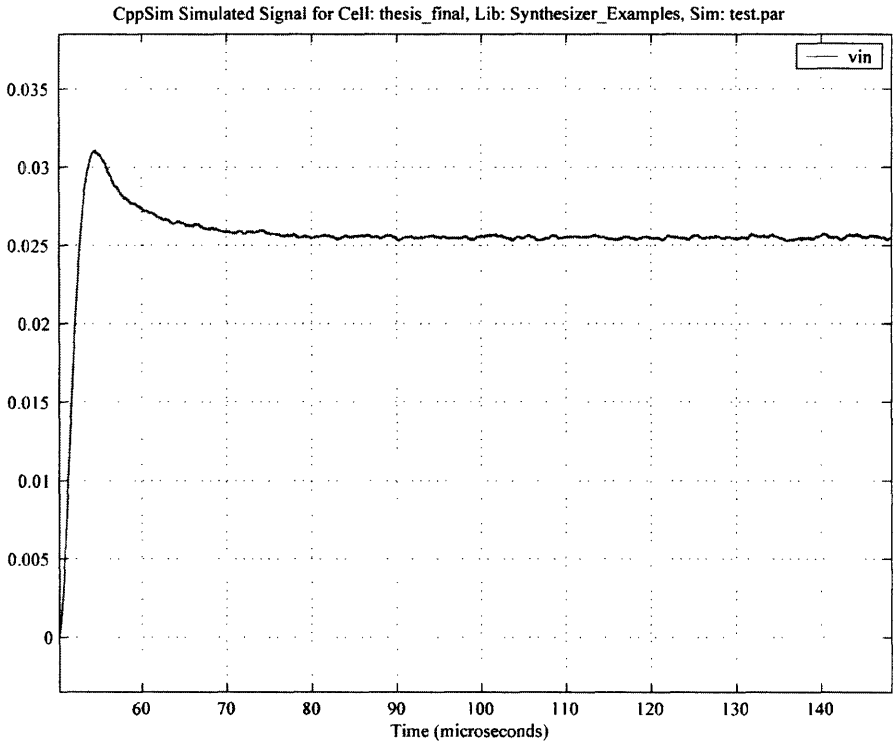

Figure 8-5: PLL response to small steps in divider value.

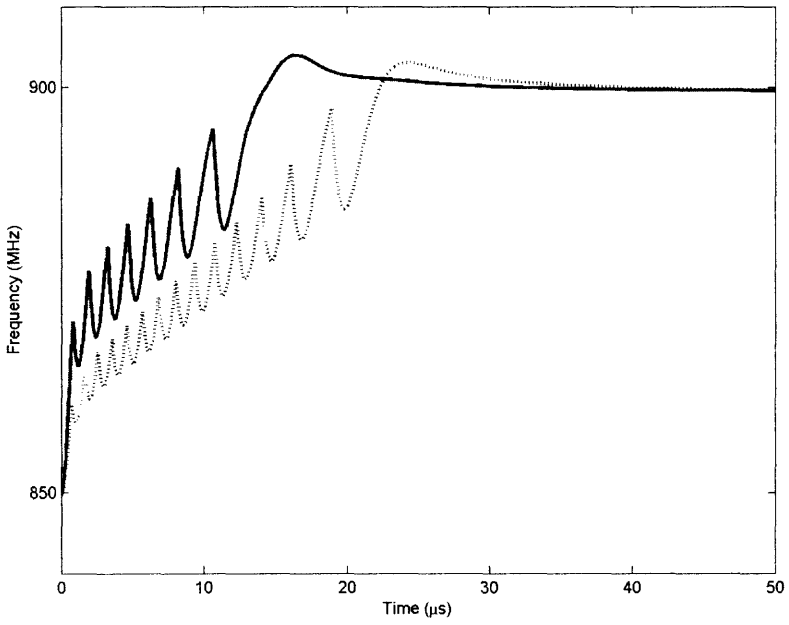

Figure 8-6: The dotted line represents a PLL with a constant charge pump current, and the solid line is the output of the PLL with the adaptive bandwidth controller.

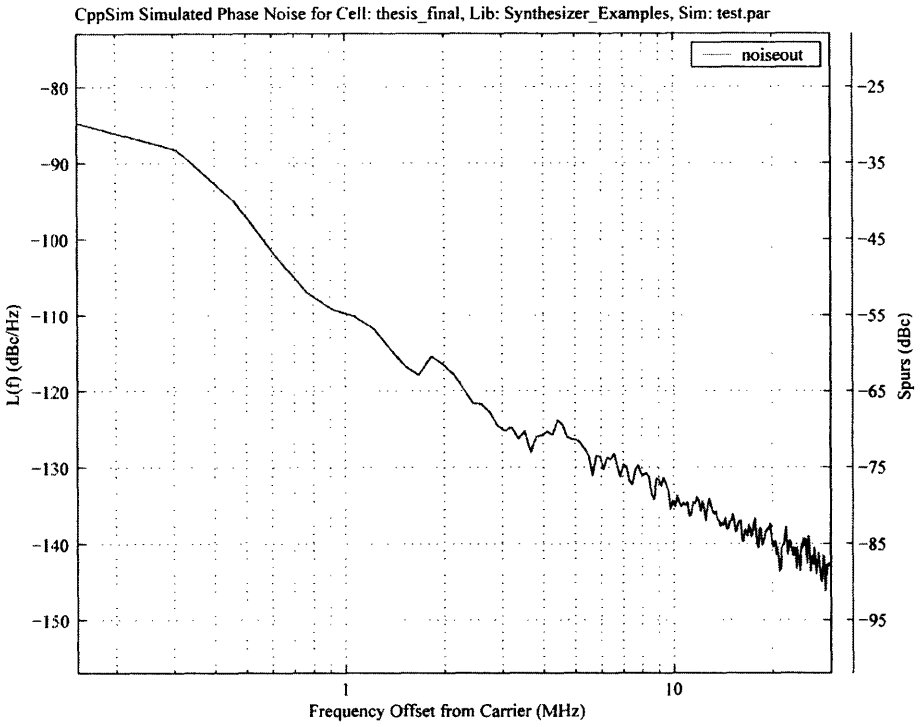

Figure 8-7: PLL phase noise.

Bibliography

[1] William S. T. Yan and Howard C. Luong. A 2-v 900mhz monolithic cmos dual-loop frequency synthesizer for gsm receivers. *IEEE Journal of Solid-State Circuits*, 37(8):1012–1020, August 2002.

[2] Thomas H. Lee and Ali Hajimiri. A general theory of phase noise in electrical oscillators. *IEEE Journal of Solid-State Circuits*, 33(2):179–194, February 1998.

[3] Ali Hajmiri, Sotirios Limotyrakis, and Thomas H. Lee. Jitter and phase noise in ring oscillators. *IEEE Journal of Solid-State Circuits*, 34(6):790–804, June 1999.

[4] Chan-Hong Park and Beomsup Kim. A low-noise 900mhz vco in $0.6\mu m$ cmos. *1998 Symposium on Circuits and Systems, Digest of Technical Papers*, pages 28–29, June 1998.

[5] Michael H. Perrott, Theodore L. Tewksbury III, and Charles G. Sodini. A 27-mw cmos fractional-n synthesizer using digital compensation for 2.5-mb/s gfsk modulation. *IEEE Journal of Solid-State Circuits*, 32(12):2048–2060, December 1997.

[6] Michael H. Perrott. Behavioral simulation of fractional-n frequency synthesizers and other pll circuits. *IEEE Design & Test of Computers*, 19(4):74–83, July 2002.

[7] Michael H. Perrott, Mitchell D. Trott, and Charles G. Sodini. A modelling approach for $\sigma - \delta$ fractional-n frequency synthesizers allowing straightforward noise analysis. *IEEE Journal of Solid-State Circuits*, 37(8):1028–1038, August 2002.

Appendix A

Notation

Throughout this paper, we will use the following notation to readily differentiate between average quantities and small-signal quantities.

Table A.1: Variable Notation

Variable	Total Quantity	DC Average	Small Signal
Voltage	v_{GS}	V_{GS}	v_{gs}
Current	i_D	I_D	i_d

In this paper equations governing MOSFETs are presented. Because NMOS and PMOS physics are mostly the same, when possible only the NMOS equation will be given, with the assumption that the PMOS form can be derived with the appropriate sign and node label changes. In cases where there is a difference between the devices, both cases will be presented.

Noise is written as a lowercase variable v or i for voltage or current noise, with a subscript always proceeded by n for noise, then averaged with an over-line, and squared to represent mean squared density. Thus, the label $\overline{i^2_{n\mathrm{M4}}}$ means the drain current noise of MOSFET M4, with units (A^2/Hz).

The variable LO used in some of the oscillator chapters refers to the local oscillator, or the output of the VCO. The crystal frequency reference is denoted f_{ref} or ω_{ref}.

We denote the fourier transform of a signal $x(t)$ by $\mathbf{F}_x(j\omega)$.

Appendix B

Simulating the PLL in MATLAB

It is possible to simulate PLL dynamic behavior in MATLAB, using a discrete time model. Here we will develop this model. Let Δt be the time increment with which we will discretize our system. A capacitor's voltage can be expressed as

$$\frac{v[n+1]-v[n]}{\Delta t} = \frac{i[n]}{C} \tag{B.1}$$

From here, we can express the impedance of a second order loop filter whose schematic is given in figure 7-1 as

$$v_1[n+1] = \frac{i[n]}{C_1}\Delta t - \frac{v_1[n]-v_2[n]}{R_2C_1}\Delta t + v_1[n] \tag{B.2}$$

$$v_2[n+1] = \frac{v_1[n]-v_2[n]}{R_2C_2}\Delta t + v_2[n] \tag{B.3}$$

Where $v_1[n]$ is the control voltage, $v_2[n]$ is the lag capacitor voltage, and $i[n]$ is the average charge pump current. Note that we require two state variables to model the filter since there are two capacitors.

Once we have the control voltage, the oscillator's output frequency is given by

$$f_{LO}[n+1] = K_v \cdot v_1[n] \tag{B.4}$$

The reference phase of the input is

$$\theta_{ref}[n+1] = 2\pi f_{ref}\Delta t + \theta_{ref}[n] \tag{B.5}$$

Now we can write the local oscillator phase, after the divider, as

$$\theta_{LO}[n+1] = \frac{2\pi f_{LO}[n]}{N}\Delta t + \theta_{LO}[n] \tag{B.6}$$

where N is the average division ratio.

Finally, the charge pump current is given by

$$i[n+1] = \frac{\theta_{ref}[n]I_{CP}}{2\pi} - \frac{\theta_{LO}[n]I_{CP}}{2\pi} \tag{B.7}$$

This can be simply written as a matrix,

$$\begin{bmatrix} v_1[n+1] \\ v_2[n+1] \\ f_{LO}[n+1] \\ \theta_{ref}[n+1] \\ \theta_{LO}[n+1] \\ i[n+1] \end{bmatrix} = \begin{bmatrix} 1-\frac{1}{R_2C_1s} & \frac{1}{R_2C_1s} & 0 & 0 & 0 & \frac{1}{C_1s} \\ \frac{1}{R_2C_2s} & -\frac{1}{R_2C_2s}+1 & 0 & 0 & 0 & 0 \\ K_v & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{2\pi}{Ns} & 0 & 1 & 0 \\ 0 & 0 & 0 & \frac{I_{CP}}{2\pi} & -\frac{I_{CP}}{2\pi} & 0 \end{bmatrix} \begin{bmatrix} v_1[n] \\ v_2[n] \\ f_{LO}[n] \\ \theta_{ref}[n] \\ \theta_{LO}[n] \\ i[n] \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{2\pi f_{ref}}{s} \\ 0 \\ 0 \end{bmatrix} \tag{B.8}$$