LOW POWER ANALOG CIRCUITS FOR AN ALL CMOS INTEGRATED CDMA RECEIVER

by

Lapoe E. Lynn

Memorandum No. UCB/ERL M95/73

1 September 1995

LOW POWER ANALOG CIRCUITS FOR AN ALL CMOS INTEGRATED CDMA RECEIVER

by

Lapoe E. Lynn

Memorandum No. UCB/ERL M95/73

1 September 1995

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

Acknowledgments

It is not often that one gets the chance to reflect on all the people and factors that have contributed to what is, in a sense, the culmination of 18+ years of education. Therefore, while keeping in mind that there are many who must necessarily go unmentioned here (but whose influence is gratefully remembered), I would like to thank the following people for helping me arrive (finally!) at this point.

This research was conducted under the kind support of the Advanced Research Projects Agency (ARPA) in the form of an NDSEG Fellowship, as well as through ARPA contract #J-FBI 92-150 under the guidance of Professor Robert Brodersen. Thanks to Prof. Brodersen and Prof. Paul Gray for reviewing this write-up. Many thanks also go to Tom Boot, Peggy Brown, Elise Mills, and Heather Brown for their relentlessly efficient (and friendly!) administrative support.

An integral part of being a senior graduate student, although it is not necessarily included in the job description, is helping the younger students who follow after you. Not enough can be said about this function -- graduate school would have been useless for me, without the advice and guidance of my peers. I have been in the happy position of learning from some of the best and most generous. Dave Cline and Thomas Cho have each spent hours of their valuable time patiently helping me with ideas, pointing me towards papers, and even donating much of their test setup to me. However, my deepest gratitude must go to Samuel Sheng. Drawing on a breadth **and** depth of knowledge unlike any I have ever seen, Sam has lent me invaluable guidance, direction, support and friendship (despite being burnt-out like a used piece of charcoal!). The reference page for this work should

have one large reference to **[Sam Sheng's Brain]** written in 42 point font across the center.

On a less technical front, without the love and support of friends and family, I would certainly never have finished this project, and would no doubt have long since collapsed into a burnt-out pile of wreckage. The playful antics of the RF group, the often distracting 550 gang, and the hard-core analog gurus of PRG group, all helped make work a fun place to be. And without friends like James Kao, Joy Ku, John Yen(bo), Aiting Tung, Chris Rivera, Jennifer Cheng, Sam Sheng, and Sally Chen, I probably would have long since lost my mind and moved back to Colorado.

Friends like Kevin Stone, Dipanwita Deb, Susanne Wong and Sekhar Narayanaswami, are the type of friends that you know will stick with you through the end of the world. Thanks to Sus for putting up with such an awful roommate, Dip for calling and calling... even though I always flaked, Sek and Kevin for making the long hours in Cory bearable. You guys are the best.

And finally, I have to thank the four people whose love has made me who I am, and whose strength has carried me to where I am now.

Helen, who has made the last year and a half the fullest and richest of my life. Helen, thank god you have a lot of patience and a good sense of humor! :)

Aenoch, who has watched over me as carefully as any big brother could. Hang in there, big guy.

And finally, I'd like to dedicate this work to my parents with all my love. Mom, Dad... believe it or not, you guys are pretty darn cool.

Table of Contents

List of Figures

Chapter 2

Chapter 3

Chapter 4

Chapter 5

Chapter 6

CHAPTER 1

Introduction

As interest in wireless portable computation has blossomed over the past several years, the electronics industry has seen the much anticipated debut of such products as the Apple Newton fall upon lethargic sales and widespread consumer dissatisfaction. These systems have attempted to provide a networked computing environment to individuals by basically building a small computer with a cellular phone connection to provide "connectivity" to the growing internet. However, portability requirements have severely limited the amount of computing resources that can be placed in such units; this fact, coupled with the use of a low-bandwidth wireless link have resulted in products that have failed to provide the quality of service necessary to please the modern-day consumer. An alternative is the UC Berkeley InfoPad.

The Infopad system places almost all of the user's computation in large, non-portable resources on a high-speed wired network, all of which is connected to the portable unit through a wideband wireless radio link. Such a system has many attractive advantages over current implementations. By moving a majority of the computation away from the portable unit, significant power savings can be achieved, extending battery lifetime for the user. Furthermore, since voice and pen recognition are also performed by the non-portable server, information passing over the wireless link is restricted to voice and video data. Since these forms of data are inherently resistant to bit errors (i.e. the human senses may not detect bit errors below a certain level), a higher bit-error rate can potentially be toler-

ated in the radio connection. The caveat to all of these advantages is that the requirements of the wireless radio link become much more severe. Supporting multiple users demanding multimedia (e.g. video) data rates results in a system which consumes a large amount of bandwidth. The UC Berkeley Infopad Project is an attempt to design such a system. Therefore, the design of the Infopad radio needs to simultaneously achieve high speed, wide bandwidth, extremely low power, and high integration.

The work presented here represents part of the effort to design and build the infopad radio receiver as a low-power, monolithic, single-CMOS-chip. In particular, this work focuses on the design and implementation of high speed, discrete-time variable gain circuits, as well as an extremely fast, low power 4 bit analog to digital converter for use in the integrated receiver chip.

CHAPTER 2

Motivation

2.1 Architecture

In order for a true multimedia server to support video data to multiple terminals, a large amount of bandwidth is required. In fact, almost 100 MHz of bandwidth are specified for the system in order to support up to 50 users in a single cell (taking into account that high quality compressed video data rates require on the order of 1-2 Mb/sec). This is quite a sizeable amount of data, and in order to robustly support such a broadband transmission, a direct sequence spread spectrum modulation scheme was adopted [1]. While there are numerous reasons for taking advantage of the noise immunity and potential multipath interference rejection of a spread signal, the decision to transmit a CDM (Code Division Multiplexed) signal has several significant side effects on the design of the receiver circuitry (see [2], [3], and [4]). Perhaps the most obvious is that each user must now receive and decode a transmit pulse that has been modulated with a pseudorandom code, requiring high speed signal processing to despread the information. On the other hand, no frequency tuning in the receiver is required since the entire band is translated directly to baseband (actually, the signal is mixed to a low IF frequency where it is directly converted from analog to digital and mixed to baseband digitally). Also, since quantization noise is rejected by the spread spectrum processing gain, the resolution requirements on the ADC in the receiver are greatly reduced [5]. Table 1 is a summary of the specifications for the InfoPad

radio receiver design [1]. A standard superheterodyne architecture could certainly be used to implement the receiver design presented above. A representative block diagram for such a receiver is shown in Figure 1 including a transmit section. Down conversion is performed in two steps before analog-to-digital conversion -- requiring two local oscillators, multiple mixer and amplifier chips, and numerous discrete filter components. Clearly, such a design does not lend itself easily to a highly integrated, low-power implementation [6]. The power consumed by each active ele-

TABLE 1. System Parameters

Carrier Frequency	> 1 GHz
Chipping Rate	64 MChips/sec
Spreading Gain	64 (18dB)
Raw User Data Rate	2 Mbps
Symbol Rate	1 Mbaud (DQPSK encoded)
Rec'd Signal Strength (into 50 ohms)	-80 dBm to -40 dBm (dynamic range)
A/D Resolution	4 bits
A/D Conversion Rate	128 Msamples/sec

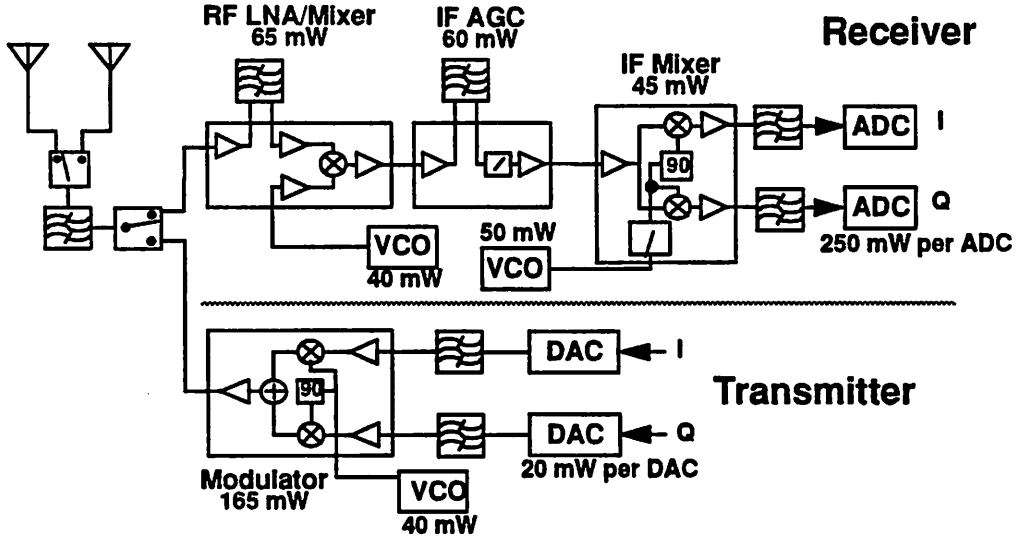

FIGURE 1. Superheterodyne Transceiver Architecture

ment is included in Figure 1. The receiver alone consumes on the order of 750 mW of power, with a large portion of that going into the discrete A/D converters. By designing low-power A/D converters in CMOS technology and by switching to a quasi-direct conversion architecture, higher integration and significant power savings can be achieved.

A homodyne conversion can be achieved by subsampling the RF carrier directly [7]. By constraining the subsampling rate to be an integer divisor of the carrier frequency, the RF signal is converted directly to a discrete-time baseband signal. Although energy from every integer multiple of the sampling frequency is mixed down to the discrete-time baseband region, prefiltering can be applied to narrowband the noise around the RF carrier band. Figure 2 shows the process of subsampling in the spectral energy domain. The InfoPad CDMA radio uses this technique to replace the superhet architecture with a single simple CMOS sampling switch being driven by a crystal oscillator. The architectural change results in the elimination of two VCOs, two bipolar mixer chips, and a quadrature tank element, with significant savings in power, area, and com-

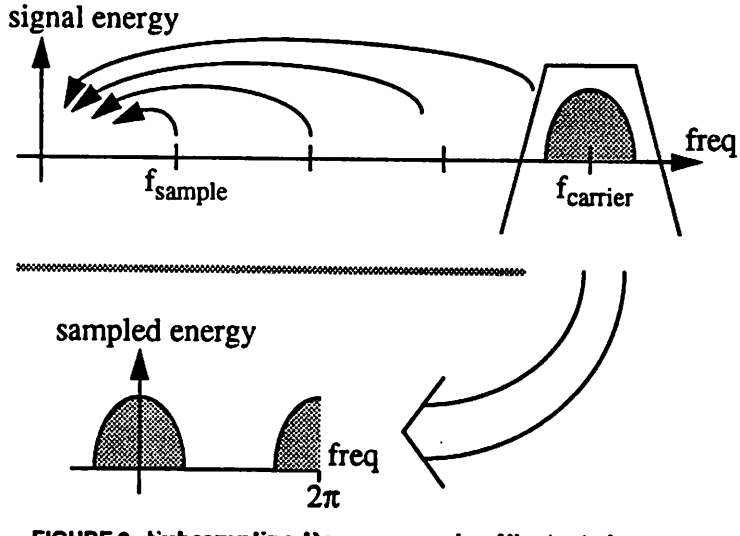

FIGURE 2. Subsampling Downconversion Illustrated

plexity. However, the InfoPad CDMA radio design does not *quite* follow this model exactly. In order to avoid the multitude of problems associated with direct conversion receivers, the subsampling mixers convert the RF signal to a low intermediate frequency of 64 MHz. This signal is directly converted to 4 bits of digital information at a 4x oversampling rate of 256 MHz, and is subsequently mixed to baseband digitally (a 4x oversampling ratio is required for the timing recovery loop). At such a high conversion rate, the design of a low-power CMOS A/D converter becomes quite challenging even for only 4 bits of resolution.

Fortunately, as Figure 3 illustrates, a 64 MHz sine wave, sampled at 256 MHz consists of mostly zero samples. Therefore, by multiplying the input signal (on a sample by sample basis) with the values shown in Figure 3 (+1, 0, -1, 0, +1...), every other sample can be eliminated. The net result is that the analog to digital conversion only needs to take place at 128 Msamples/sec. (Still a formidable number for a low power CMOS converter!)

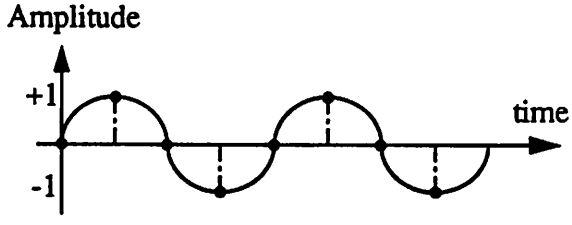

FIGURE 3. 4x oversampled sine wave

2.2 Chip Design

A block diagram of the fully integrated receiver chip is shown in Figure 4. The shaded blocks indicate the focus of this work. The automatic gain control circuitry could be placed either before or after the sampling demodulator blocks, but by placing the AGC *after* the subsample-mixing operation, the design moves to the discrete time domain. Discrete time amplifiers can implement precisely controlled gain determined by the

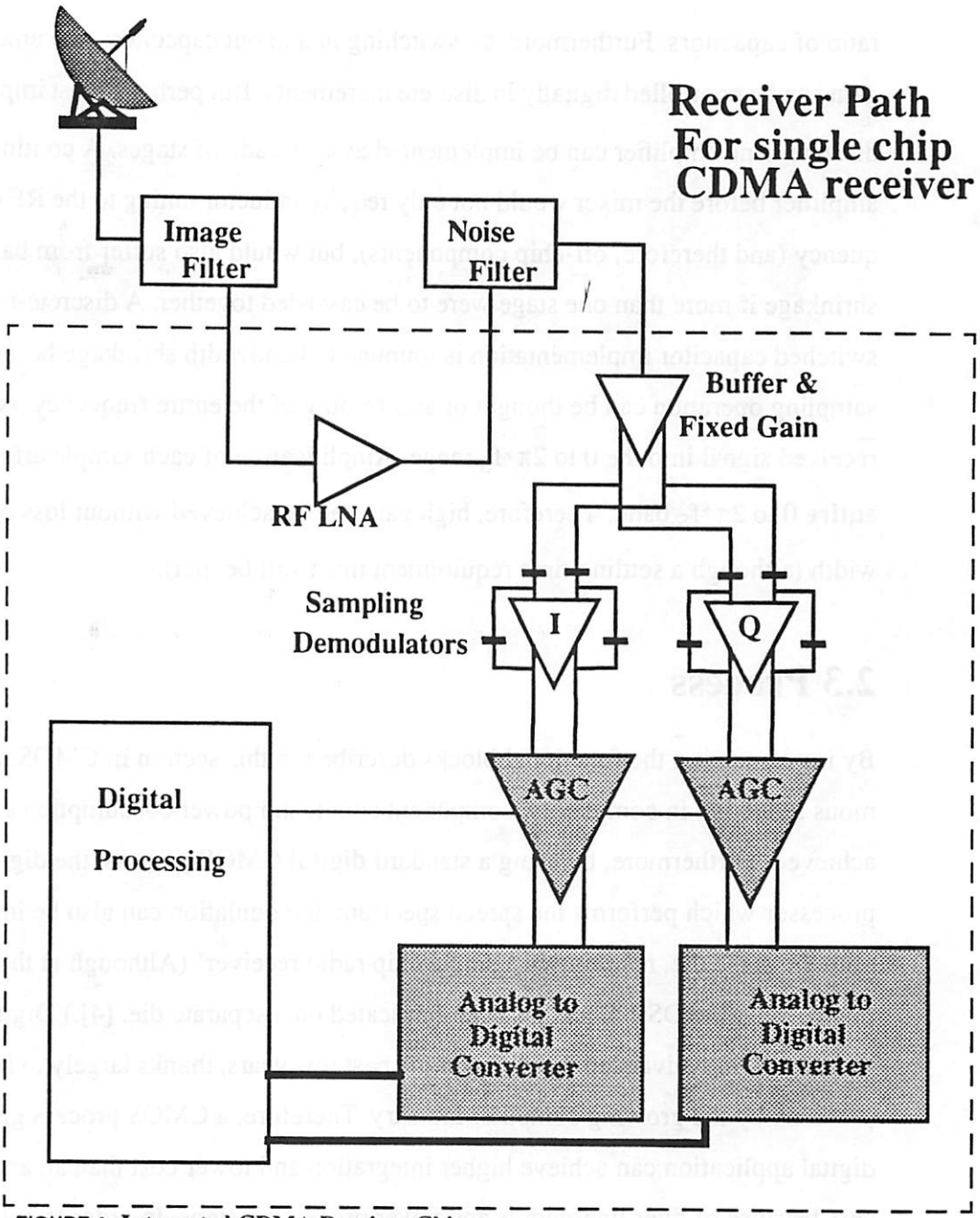

FIGURE 4. Integrated CDMA Receiver Chip

ratio of capacitors. Furthermore, by switching in and out capacitors, the amount of gain can be controlled digitally in discrete increments. But perhaps most importantly, a discrete-time amplifier can be implemented as a cascade of stages. A continuous-time amplifier before the mixer would not only require inductor tuning to the RF carrier frequency (and therefore, off-chip components), but would also suffer from bandwidth shrinkage if more than one stage were to be cascaded together. A discrete-time, switched capacitor implementation is immune to bandwidth shrinkage because the sampling operation can be thought of as a folding of the entire frequency axis of the received signal into the 0 to $2\pi*f_S$ range. Amplification of each sample affects the **entire** 0 to $2\pi*f_S$ band. Therefore, high gain can be achieved without loss of bandwidth (although a settling time requirement must still be met).

2.3 Process

By implementing the functional blocks described in this section in CMOS, an enormous reduction in complexity, component count, and power consumption can be achieved. Furthermore, by using a standard **digital** CMOS process, the digital signal processor which performs the spread spectrum demodulation can also be integrated onto the same die, resulting in a single chip radio receiver! (Although at the time of this writing, the DSP block has been fabricated on a separate die. [4].) Digital CMOS technology has advanced rapidly over the past few years, thanks largely to the impetus provided by the growing computer industry. Therefore, a CMOS process geared for a digital application can achieve higher integration and lower cost than an analog process because of finer line-widths and fewer processing steps. In order to achieve the

eventual goal of a single-chip solution, the InfoPad CDMA radio has been designed in a standard digital CMOS process. Access to this 1.0 micron Hewlett-Packard technology was provided by the MOSIS service. Unfortunately (and perhaps not surprisingly), there are several pitfalls to designing high-performance analog circuits in a technology intended primarily for digital circuit design. Precision resistors, for example, are not available in this process. Transistor output impedance is lower than might be expected from an *analog* CMOS process of comparable minimum gate length. Bipolar transistors usually provide too little benefit, usually require static power consumption, and are too costly to justify in a digital design (many analog technologies at the time of this writing include the ability to implement bipolar **and** CMOS transistors -- and are called BiCMOS processes), but perhaps most importantly, a digital technology lacks a second layer of polysilicon which is used in analog designs to implement precision capacitors. Analog CMOS circuit designs utilize two layers of polysilicon, separated by a thin layer of dielectric oxide, to create the floating capacitors used in switched-capacitor and sample-and-hold circuits. The absence of a second layer of poly makes the creation of these capacitors difficult since the dielectric oxide separating the metal and polysilicon layers is typically quite thick, resulting in a capacitance-per-unit area 100 times smaller. Therefore, capacitors created in this technology from overlapping plates of metal and/or polysilicon consume a large amount of area; and perhaps even more importantly, each capacitor implemented in this fashion, includes a large parasitic capacitance to the substrate whose value is as large or even larger than the value of the desired capacitor.

TABLE 2. NOMINAL PROCESS SPECIFICATIONS

Parameter	Value	Parameter	Value
NMOS V_T	0.74 volts	PMOS V_T	-0.85 volts
NMOS KP (μC_{ox})	119 $\mu A/V^2$	PMOS KP (μC_{ox})	34.0 $\mu A/V^2$
NMOS L_D	0.16 μm	PMOS L_D	0.105 μm
NMOS delta_W	0.36 μm	PMOS delta_W	0.39 μm
NMOS gamma	0.58 $V^{1/2}$	PMOS gamma	0.50 $V^{1/2}$
NMOS lambda	0.02	PMOS lambda	0.085
Capacitance			
t_{ox}	161 A	C_{ox}	2.17 $fF/\mu m^2$
$C_{poly\text{-}sub}$	0.058 $fF/\mu m^2$	$C_{M1\text{-}poly}$	0.055 $fF/\mu m^2$
fringe per edge	0.043 $fF/\mu m$	fringe per edge	0.049 $fF/\mu m$
$C_{M1\text{-}sub}$	0.031 $fF/\mu m^2$	$C_{M1\text{-}M2}$	0.035 $fF/\mu m^2$
fringe per edge	0.044 $fF/\mu m$	fringe per edge	0.046 $fF/\mu m$
$C_{M2\text{-}sub}$	0.015 $fF/\mu m^2$	$C_{M2\text{-}M3}$	0.035 $fF/\mu m^2$
fringe per edge	0.035 $fF/\mu m$	fringe per edge	0.049 $fF/\mu m$
Sheet Resistance			
R_{ndiff}	2.4 ohms/sq	R_{pdiff}	2.0 ohms/sq
R_{poly}	2.2 ohms/sq	$R_{M1,M2}$	0.07 ohms/sq

Table 2 is a summary of some of the key parameters associated with this process. A key point of interest is the fact that the NMOS transistors undergo a mask shrink during processing, resulting in an NMOS device with a minimum drawn gate length of 0.8 microns instead of the 1.0 micron drawn length of a minimum sized PMOS device.

Therefore, an important design consideration is the asymmetry of the technology (i.e. the n-type transistors have a significantly higher f_T than the p-type devices not only due to a higher carrier mobility, but also due to a shorter minimum gate-length.

The following chapters will detail the design, implementation and testing of the AGC and A/D converter motivated in this section.

CHAPTER 3

Background

This section is intended to give some background material on MOS switched capacitor sampling and A/D converter techniques (especially high-speed topologies). It is by no means a comprehensive discussion or tutorial on design of either type of circuit.

3.1 Analog to Digital Conversion

In a world increasingly dominated by digital signal processing, analog to digital converters play an important role as the interface between "real-world" analog signals and the digital circuitry used to process them. Conceptually, an A/D converter takes an input signal and compares it to a set of predetermined equally-spaced reference values, outputting a digital code for whichever value is closest to the input signal. In this manner, an A/D converter quantizes the range of possible values a signal can take and approximates the real signal level with one of the quantized values. This approximation function adds a natural error to the output of the A/D. Referred to as quantization noise, this error is determined by the resolution of the converter and has a major impact

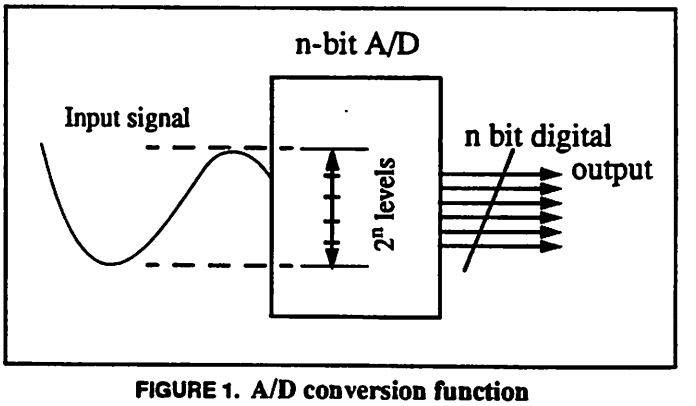

FIGURE 1. A/D conversion function

on the design of the A/D. In fact, for each additional bit of digital output desired, the resolution required (i.e. the number of reference levels used for comparison) increases by a factor of two.

Figure 1 shows the basic functionality of an A/D converter block, including the division of the input range into 2^n equally spaced levels. Clearly, in order to reduce the quantization noise introduced by the converter, more resolution is desirable. However, since complexity increases with both speed and resolution requirements, several different architectures of converters have evolved.

The most basic A/D architecture is the flash converter. A representative flash converter is a simple, logical extension of Figure 1. It consists of 2^n comparators, each comparing the same input signal against one of 2^n different reference voltages generated from

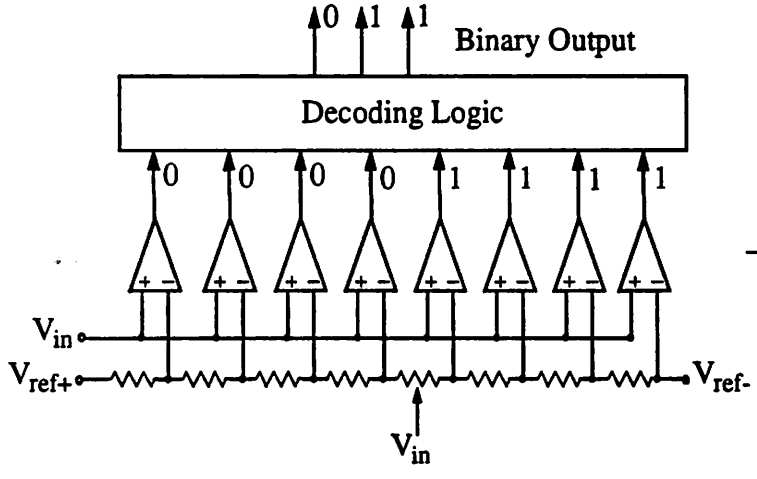

FIGURE 2. Simple 3-Bit Flash Converter

a resistive ladder bias. All of the comparators work in parallel, and are followed by digital logic converting their 2^n outputs into an n-bit number. A simplified schematic of one possible implementation of a 3-bit flash converter is shown in Figure 2. (It should be noted that since the input signal is assumed to be within the V_{ref+} to V_{ref-} range, only 2^n-1 references and comparators are really necessary.) Flash converters are fast, straightforward, and have very low latency; but they have the major drawback that size and power increase *exponentially* as the resolution is increased. A 12 bit converter requiring $2^{12} = 4{,}096$ comparators is clearly a distasteful proposition at best! Therefore, alternative architectures have been developed for high accuracy conversion.

3.2 The Subranging Architecture

Subranging and pipelined A/D converters represent two methods of reducing the area and power consumption of high resolution flash converters. A subranging architecture performs the conversion function in two steps using essential two flash converters -- a "coarse" and a "fine" converter. For example, an 5 bit comparator could be broken up into a 3 bit coarse comparison and a 2 bit fine comparison as shown in Figure 3. In this manner, the comparator count is reduced from $(2^5-1) = 31$ to $(2^3-1) + (2^2-1) = 10$. Such a savings never comes without a cost, and indeed the subranging architecture suffers a speed hit when compared to a straightforward flash converter. The fact that the conversion must occur in two steps -- with the coarse comparison necessarily finishing before the fine conversion can begin -- decreases the maximum rate at which the subranging converter can be clocked.

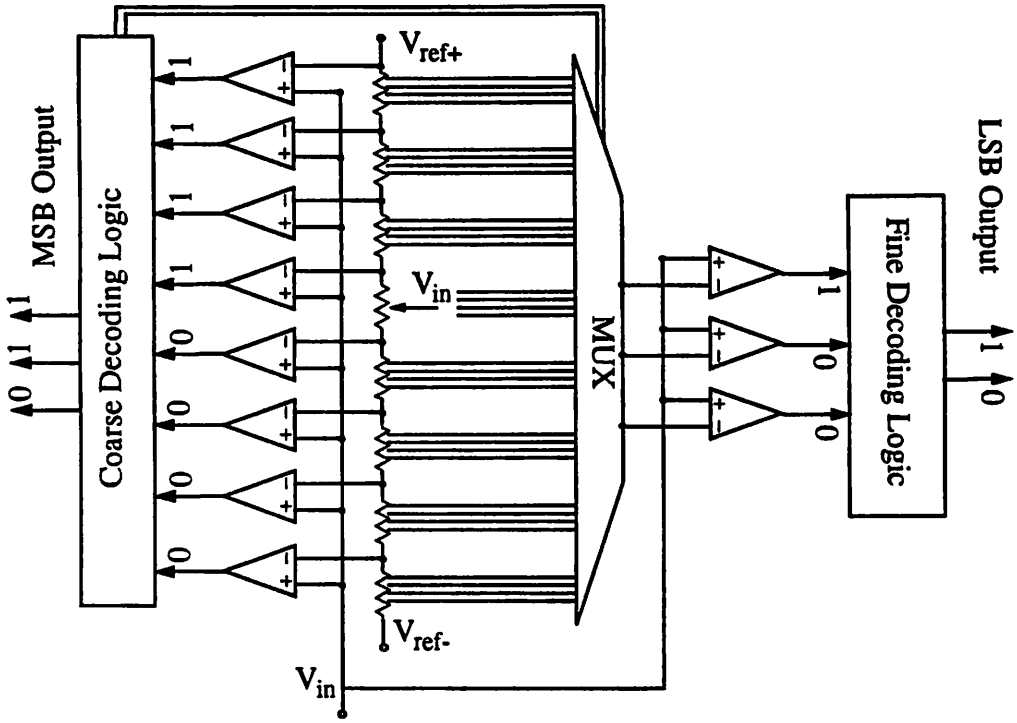

FIGURE 3. 3-Bit -- 2-Bit Subranging A/D architecture.

3.3 The Pipelined Architecture

Unlike the subranging A/D, the pipelined architecture eliminates the need for the coarse conversion to complete before the fine conversion can begin by latching the signal in between the two steps. This is exactly the same technique used in pipelining digital datapaths. By sampling and holding the signal in between functional blocks, the latency of the converter is increased, but the **throughput** is also increased. Furthermore, the pipelined converter can relax the offset requirements of the fine comparators by replacing the mux shown in Figure 3 with an analog subtracter. Once the most significant bits of the signal are known, they can be passed through a DAC and an analog representation of the MSB's

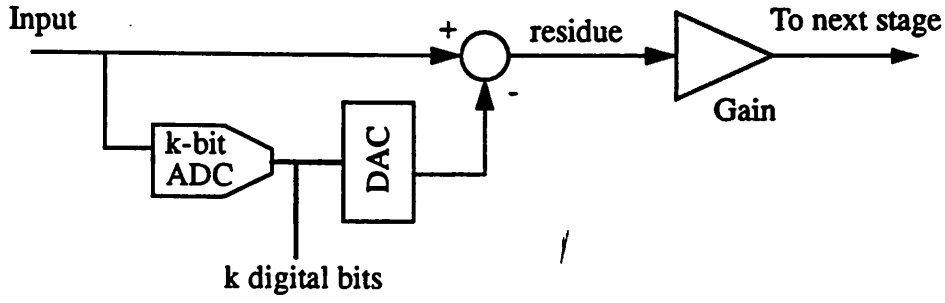

FIGURE 4. Single Stage of a Pipeline A/D Converter

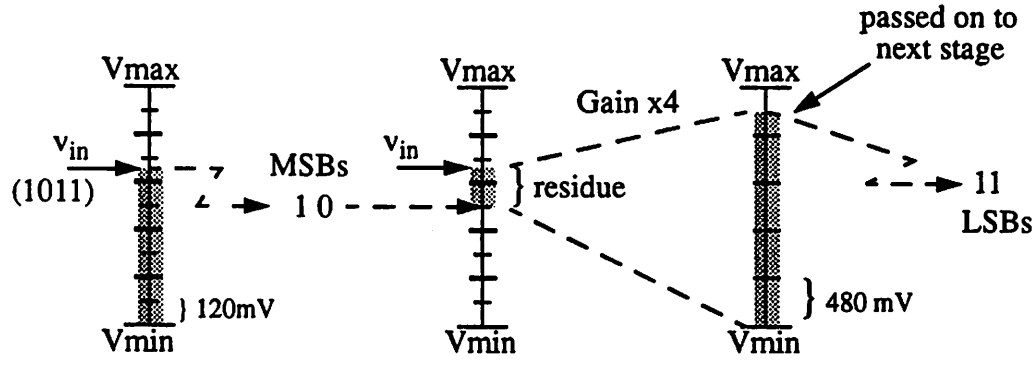

FIGURE 5. 2 bit -- 2 bit Pipeline Conversion Example

can be subtracted off from the signal. In this way, the residue that is left, which represents the LSBs of the signal, can be gained up to the original signal level. In Figure 5 for example, a 4 bit converter having an LSB size of 120 mV is broken up into a 2 bit -- 2bit pipeline. After passing through the first set of comparators, the difference between the input and the analog representation of the 2 MSBs is generated and amplified for conversion into the LSBs. In the final conversion step, the LSB size has changed to 480 mV. A sample block diagram of one stage of a pipeline A/D converter is shown in Figure 4. Multiple stages may be hooked up together, and any number of bits (limited by flash converter size) may be resolved in each stage. Finally, pipeline converters have one other feature which makes them very attractive to use -- namely, digital correction. Digital correction takes

advantage of the gain between stages of a pipeline converter in order to reject the random offset inherent in all comparators. By using extra comparators to detect when the signal has been incorrectly evaluated in a previous stage, a properly implemented digital correction scheme can make the comparator offset in all but the very last stage of the pipeline irrelevant. Furthermore, the input-referred offset of the final stage gets divided by the sum of all the interstage gain preceding it. In other words, the required comparator offset for a 1-bit per stage, 10 stage pipeline converter would be 2^{10} = 1024 times less stringent than the offset required of a flash converter! For a full explanation of the technique of digital correction, please see [8].

3.4 The Sample and Hold Circuit.

Almost all A/D converters not only convert an analog signal into a digital number, they also convert a continuous time waveform into a signal which is discrete in time. Therefore, the A/D converter samples the input signal at discrete instants of time usually at a fixed rate (the conversion rate). Sample and hold circuits typically precede A/D converters and perform the continuous-to-discrete-time conversion so that the converter does not have to deal with a rapidly changing signal -- instead, by sampling the

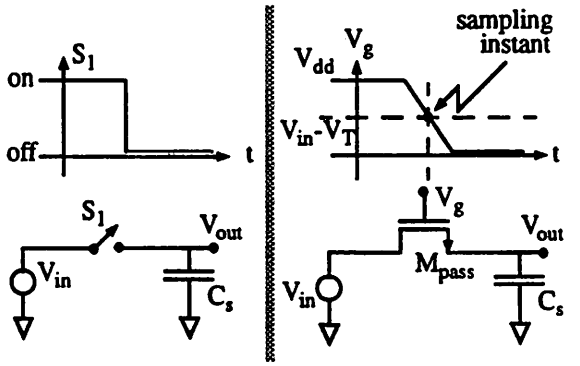

FIGURE 6. Simple Sample and Hold

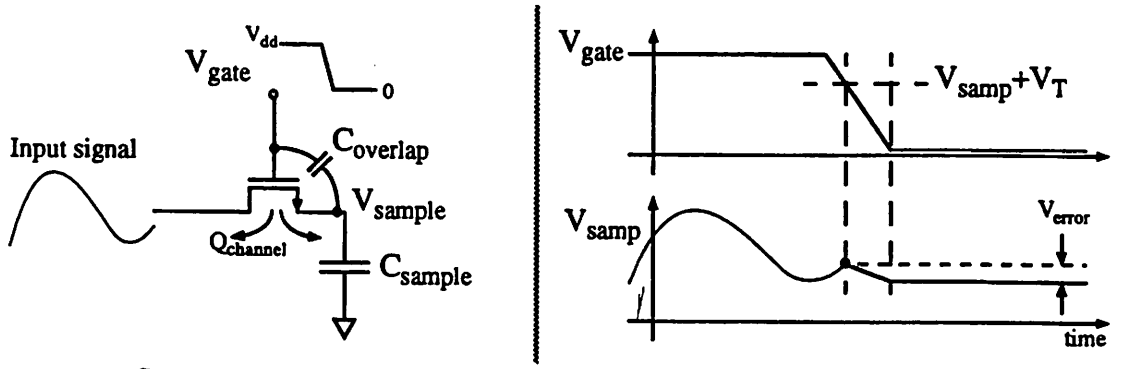

FIGURE 7. Sources of Error in Basic Sample and Hold

input signal before feeding it to the A/D, the sample and hold circuit (or S/H) provides the converter with signal which is stable and well-behaved.

The principle behind most CMOS S/H circuits is very analogous to the idea used in dynamic digital logic. One of the advantages of metal-oxide-semiconductor (MOS) technology over bipolar junction transistor technology is it's ability to store charge for relatively long periods of time. Dynamic logic families take advantage of this ability by passing "packets" of stored charge from one circuit to the next; storing each desired signal on parasitic capacitances available at each stage. S/H circuits mimic this technique, but utilize larger, more precisely controlled capacitors to manipulate the more delicate analog signals. A diagram of a simple sample and hold is shown in Figure 6. The natural characteristics of the MOS transistor make it an excellent choice for use as a switch. The right hand side of Figure 6 shows the circuit with a MOS pass gate used to replace the ideal switch. As long as transistor M_{pass} is on, the switch is closed, and the output voltage will track the input voltage. When the control voltage on the gate of M_{pass} drops below V_{in} + V_T, the transistor turns off, and a sample of charge $V_{in}*C_S$ is stored on the sampling capacitor. The right hand side of Figure 7 shows the voltage across the sampling capacitor track-

ing the input waveform until the sampling instant. After the sampling instant has passed, the value on the capacitor is held constant, and should equal the value of the input at the sampling instant. However, as Figure 7 illustrates, the act of opening the switch can introduce an error into the sampled voltage. First of all, the overlap capacitance of the pass transistor provides a capacitively-coupled path for the control voltage to inject charge onto the sampling node. As the control voltage falls (turning off the pass transistor), a capacitive divider between $C_{overlap}$ and C_{sample} is formed. The resulting error is given by:

$$\Delta v = -\left(\frac{C_{overlap}}{C_{sample}}\right)V_{dd} \qquad \text{(EQ 1)}$$

Equation 1 assumes that the control voltage switches from the supply V_{dd} to ground, and does so quickly (the switching time is on the order of the RC time constant associated with the sampling capacitor and the on-resistance of the switch). Another source of error comes from the charge stored in the channel of the MOS transistor. Since the transistor is turned off quickly, the charge in the channel of the device is left to discharge to either C_S or to the input. The amount that flows to C_S depends on the impulse response of the distributed resistance of the channel in series with the impedance on either side of it; however, a reasonable approximation

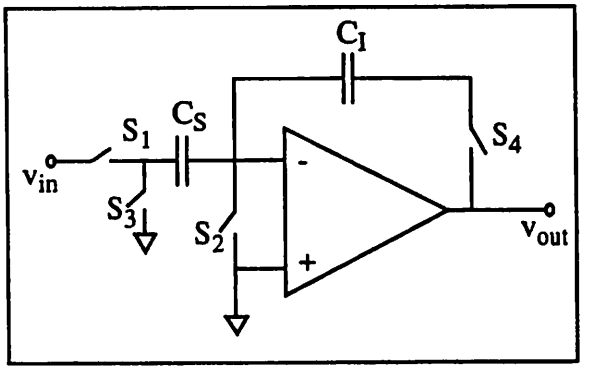

FIGURE 8. Bottom Plate Sample and Hold

assumes that 1/2 the channel charge flows to each side of the transistor for a fast V_{dd} to ground transition. The resulting error is:

$$\Delta v = -\frac{1}{2}\left(\frac{Q_{channel}}{C_S}\right) = -\frac{1}{2}\frac{(L_{eff}WC_{ox}(V_{dd}-V_{in}-V_T))}{C_S} \quad \text{(EQ 2)}$$

The fact that the error introduced is dependent on the input voltage has dire consequences. The input dependence means that the error is no longer a DC offset phenomenon, and is therefore more difficult to remove for most systems.

Figure 8 shows an improved design for a sample and hold circuit. A technique known as bottom plate sampling is employed to remove the signal dependence in the error term. The operation can be understood as follows: At first, S_1 and S_2 are closed, allowing the input signal V_{in} to be sampled onto the capacitor C_S. When S_S and S_1 are opened, an amount of charge proportional to the signal voltage and to the size of capacitor C_S is left floating on C_S exactly as in the previous circuit. However, if S_2 is opened slightly before S_1, then the second switch determines the sampling instant. Since both drain and source of transistor S_2 are at ground, no signal dependent charge injection is introduced. When S_1 is subsequently turned off, the charge in the channel sees an open circuit on the other side of C_S and therefore all the charge *must* flow to the other side of the transistor to the input source (and again no error is introduced). Finally, both S_3 and S_4 switches close, and the high gain op amp is now closed in a negative feedback loop, forcing the voltage at the inputs to the same voltage. Therefore, the charge on C_S integrates out onto C_1. If no additional circuitry is provided to remove the residual charge on C_1 after every cycle, then the circuit of

Figure 6 implements an integrator function. However, if the charge on C_I is reset after every sample, then by carefully ratioing the sizes of the two capacitors, a voltage gain of C_S/C_I can be realized. Therefore, if $C_S = C_I$, then the output voltage should equal the input signal captured at the sampling instant.

Switched capacitor sampling circuits, while traditionally used as a unity gain sample and hold preceding an A/D converter, can be easily adapted to perform a straightforward gain function. By simply ratioing C_S and C_I, the sample and hold circuit in Figure 6 effects a gain of C_S/C_I. Pipeline A/D converters perform interstage gain in this manner, since the signal travelling down the pipeline is already discrete time. An added bonus is achieved by switching S_3 to a reference voltage from a DAC instead of to ground (See section 3.3 on pipeline converters). The resulting output is $(C_S/C_I)*(V_{in} - V_{ref})$. Therefore, the interstage gain, and the subtraction of the MSBs can be efficiently combined into this one circuit.

CHAPTER 4

VGA Circuit Design

4.1 Introduction

A portable terminal such as the InfoPad **must** have variable gain placed in the receive path of its radio. The explanation for this requirement is simple: The "portability" of the terminal implies that the receiver may be moving at any given time. A moving terminal may be physically very close to the transmitter (and therefore receiving a strong signal), and then move very far away (where it receives a weak signal). Additionally, there may or may not be a direct line of sight (LOS) between transmitter and receiver -- where loss of LOS can cause severe degradation of received signal power. Wireless receivers must include circuitry to automatically adjust to these changes in environment. Simply put, the purpose of the automatic gain control circuit (AGC) is to automatically adjust the gain of the receive path so that the signal processed by the baseband circuitry appears to be of constant power regardless of the actual signal size at the antenna. If the signal level after the AGC is too large, then either the A/D or the AGC itself may begin to clip the waveform, resulting in severe distortion and loss of signal. (Although this is sometimes acceptable in certain phase-modulated systems). On the other hand, if the received signal is still too small after the last AGC stage, the A/D converter may not be able to resolve it (i.e. quantization noise will overwhelm the signal). In other words, the AGC and the A/D combine to detect a signal with a wide dynamic range. While it is true that bits in the A/D may be traded off for gain in the AGC (i.e. increased resolution in the A/D, beyond the minimum required for

SNR, allows it to detect a potentially smaller signal), higher resolution converters soon become prohibitively costly (in terms of power, area and complexity). Furthermore, the digital processing which follows the A/D (the spread spectrum demodulator in this case) must also process any additional bits added to the A/D -- again increasing power and area. In order to accommodate the entire 40 dBm of dynamic range in the received signal (see Table 1), the A/D converter would have to have 9 bits of accuracy (for a 0.3 volt signal). In the end, the A/D can not bear the full load of accepting an extremely wide dynamic range, leaving the AGC as an important part of the wireless receiver design.

Every AGC contains two critical blocks -- a variable gain amplifier (VGA) and the power detector circuit which feeds back the control signal(s) used to adjust the gain of the VGA. At the time of this writing, the power detector for the InfoPad AGC has not been implemented. However, the VGA has been designed and implemented with the eventual addition of the detector and the control loop in mind. Therefore, the rest of this chapter will describe the design of the VGA block for the InfoPad receiver.

4.2 Amplifier Topology

As mentioned before, the design of the InfoPad CDMA radio's VGA is a significant departure from traditional designs. Because of the inherent sampling incorporated into the mixing operation, a multi-stage cascade of dis-

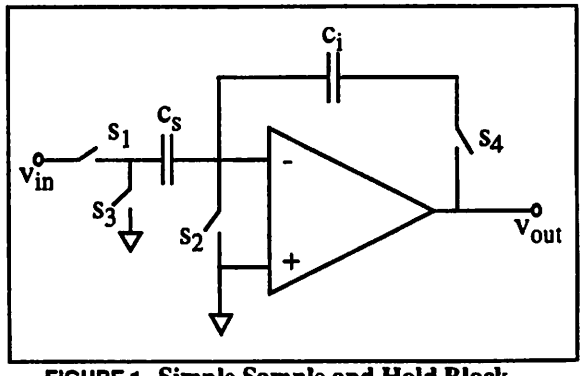

FIGURE 1. Simple Sample and Hold Block

crete-time amplifiers is used to replace a standard continuous-time VGA (variable-gain amplifier). Therefore, what was once an amplifier design problem involving a trade-off between gain and bandwidth becomes a sample-and-hold design requiring an op amp and a set of switched capacitors that can settle to four bits of accuracy within one clock period. Figure 1 shows a basic sample-and-hold amplifier as described in Section 3.4; it consists of an operational amplifier, sampling and integrating capacitors, and several switches made out of MOS transistors. However, such a simplistic design is inadequate for practical use. First, the design must be made differential. A differential circuit is necessary not only to reject charge injection from the switches (to first order, charge will inject equally into both paths becoming a common-mode offset), but perhaps more importantly, a differential topology is required to reject common mode noise coupling into the analog signal path (especially from the large digital signal processing block on chip used to demodulate the CDMA signal). Figure 2 shows how digital switching can cause common mode ringing on analog lines. If the sample is taken single-ended, a large error would result (often larger than the signal itself!). But a differential design with careful layout (so that parasitic capacitance is balanced differentially) can maintain a small differential signal amidst such large common mode noise.

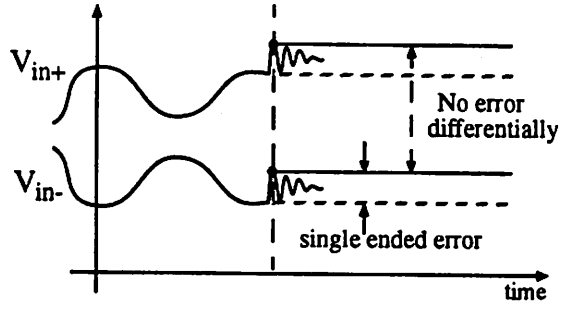

FIGURE 2. Coupling of Digital Ringing into Signal Path

As mentioned before, signal dependent charge injection can be a malignant side-effect of the sampling operation. The timing of the control voltages for

switches S_1, S_2, and S_3 in Figure 1 must be designed to effect the bottom-plate sampling described in Section 3.4. When performing bottom plate sampling, the operational amplifier shown in Figure 1 is used to drive the output to the correct value proportional to the input voltage. While quite effective for combatting signal dependent charge injection, there is another side effect to this topology. When the circuit is in track mode, the output no longer follows the input voltage. Therefore, when the hold phase arrives, the output must settle to the correct output voltage from some reset value (usually zero). If the amplifier is appropriately designed with adequate phase margin, the settling curve follows an exponential curve with a single RC time constant, and a representative curve is shown in Figure 3. Typically, only half of the clock cycle time is dedicated to the hold period (the other half goes to the tracking period). Therefore, the output has half of one cycle ($T/2$) to exponentially approach it's final value. Unfortunately, an infinite amount of time is required for an exponential curve to reach it's final *exact* value. This can be seen in Equation 3, where A is the final output value and the error goes to zero only for t = infinity:

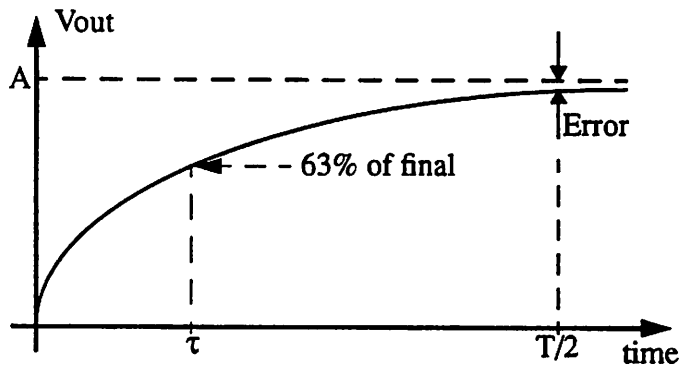

FIGURE 3. Settling Time Error

$$V_{out} = A\,(1 - e^{-t/\tau}) \qquad \text{(EQ 3)}$$

Where τ is the time constant of the circuit. Therefore, there will always be an error introduced into the signal due to incomplete settling. Fortunately, the situation is not quite as grim as it may seem. The settling error incorporated in Equation 3, $Ae^{-t/\tau}$, is linearly proportional to the final value A if t is constant. In other words, if given the same amount of time to settle, the circuit will always settle to the same percentage of its final value. Therefore, the gain of the circuit will always be reduced by the constant factor $e^{-t/\tau}$. As long as t is constant, this gain compression does not introduce a real error into the signal. In fact, the actual **exact** value of the gain is rather unimportant since the negative feedback from the AGC control loop will try to force the output of the VGA to be constant regardless of the gain of each stage. (Although the compression still causes a loss of gain in the signal path!) However, a real error does enter the signal path when the allowed time, t, is not constant. Timing jitter in the sampling instant provided by the edge of a clock causes t to vary somewhat from sample to sample, resulting in an error which is proportional to the jitter variation, Δt, and to the slope of the settling curve at the sampling instant. Given enough time or a very fast circuit, this error can be quite negligible (since the slope of the curve decreases with time). Unfortunately, the InfoPad design has neither of these luxuries. The 128 Msample per second requirement translates into a 7.8 nanosecond clock cycle time. At most half of this (and in reality, much less than half) can be used for settling time.

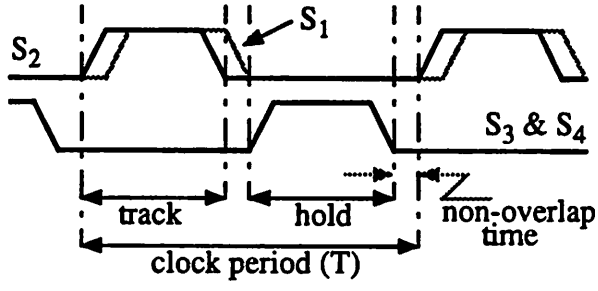

FIGURE 4. Non-overlapping clock phases used to control sample & hold.

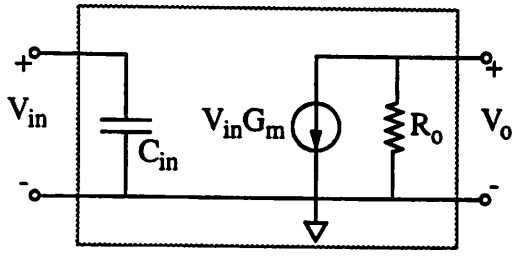

FIGURE 5. Two Port Representation of Operational Transconductance Amplifier

Figure 4 shows clock phases for the circuit in Figure 1. The hold period (and therefore, the settling time) is reduced in length by the non-overlapping period between phases (necessary for proper sampling) as well as by the finite rise and fall time of the clock's edges. The hold period for the InfoPad radio's clocks is about 2.5 ns. In order to both avoid degrading the SNR of the signal, as well as to avoid significant gain attenuation, a the settling error in the receiver VGA should be kept smaller than half the LSB of the 4 bit converter. In other words, the error must be less than 1 part in 32 (3.125%) of the signal level.

$$e^{-t/\tau} \leq 0.03125 \qquad \text{(EQ 4)}$$

$$t/\tau = \ln\left(\frac{1}{0.03125}\right) \cong 3.47 \qquad \text{(EQ 5)}$$

Equation 4 and Equation 5 show that settling to this level of accuracy requires approximately 3.5 settling time constants. Plugging in $t = 2.5$ ns into Equation 5 gives τ on the order of 0.7 ns. However, since the VGA design consists of more than one stage, each settling error will add to the others, resulting in a larger error. For example, in order to keep the error lower than one-half LSB for a four stage VGA, each stage would need to settle to less than 1 part in 128 (~0.8%) of the exact value. The result is a τ on the order of 0.5 ns... or over 310 MHz of bandwidth.

Therefore, a careful design of the amplifier is required with settling time (and therefore, closed loop bandwidth) as a primary design constraint. One of the main limiting factors in the design of a high speed sample and hold amplifier (referred to as an SHA) is the operational amplifier used to drive the output to it's final value. In order to achieve the fastest possible circuit, a single-stage transconductance topology has been chosen for the op amp. The details of the design of this operational transconductance amplifier (OTA) is described later in this chapter. But for the purposes of this section, it is enough to know that the OTA can be modeled as a G_m transconductance with a high output impedance R_o, as illustrated in Figure 5.

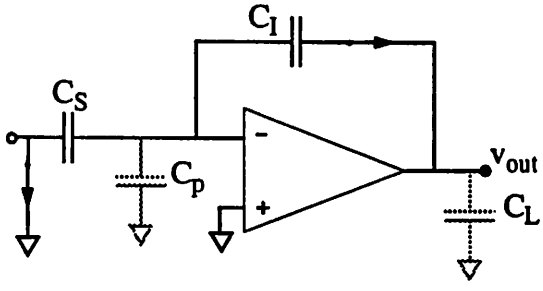

FIGURE 6. Amplifier in Hold Mode (Evaluation Phase)

Figure 6 shows the SHA during the evaluation phase of its operation. C_I encloses the OTA in a series-shunt feedback loop with feedback factor:

$$f = \frac{C_I}{C_I + C_S + C_p + C_{in}} \qquad \text{(EQ 6)}$$

Where C_p is the parasitic capacitance at the summing node, C_{in} is the input capacitance of the op amp, and C_L is the output load being driven. Given that the open loop gain of the amplifier is $G_m R_o$, the closed loop gain and bandwidth of the circuit can be determined. Including the capacitive divider before the input to the OTA, the closed loop gain becomes:

$$A_{CL} = \left(\frac{C_S}{C_S + C_P + C_{in} + C_I}\right)\left(\frac{A_{OL}}{1 + A_{OL}f}\right) = \left(\frac{C_S}{C_S + C_P + C_{in} + C_I}\right)\left(\frac{G_m R_o f}{1 + G_m R_o f}\right)\frac{1}{f} \quad \text{(EQ 7)}$$

$G_m R_o f$ is the loop gain, T, of the circuit, and f is given by Equation 6, leaving:

$$A_{CL} = \left(\frac{C_S}{C_S + C_P + C_{in} + C_I}\right)\left(\frac{C_I + C_S + C_p + C_{in}}{C_I}\right)\left(\frac{T}{1+T}\right) = \frac{C_S}{C_I}\left(\frac{T}{1+T}\right) \quad \text{(EQ 8)}$$

The loop gain term in Equation 8 is close to unity for large values of T. However, in an AGC, a small reduction in the gain is quite irrelevant since the **overall** gain in the receiver is affected very little by this variation because of the negative feedback of the control loop. Therefore, lower values of T are acceptable, and the closed loop gain of the amplifier in is close to (but not exactly) C_S / C_I. The settling time at the output node is determined by the output impedance of the closed-loop amplifier and by the output load. Therefore, the RC time constant at the output of the amplifier is given by:

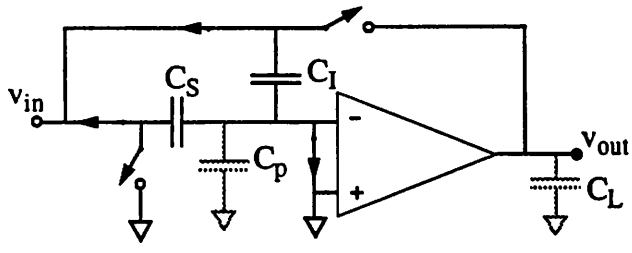

FIGURE 7. SHA topology sampling onto C_I as well as C_S.

$$\tau = \left(\frac{R_o}{G_m R_o f}\right)C_L = \frac{C_L}{G_m f} \quad \text{(EQ 9)}$$

Clearly, in order to maximize the speed of the amplifier, the feedback factor, f, should be made as large as possible, and the transconductance, G_m, should also be increased if possible. G_m is an op amp parameter, and is discussed in Section 4.3. However, the feedback factor, defined in Equation 6 relates directly to C_S and C_I. Unfortunately, in order to achieve a gain greater than one, the closed loop gain

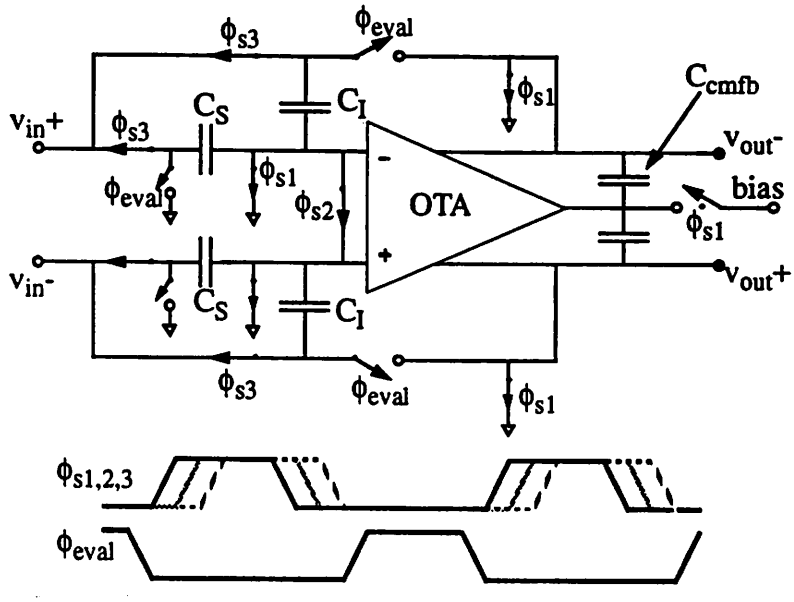

FIGURE 8. Final SHA topology including clock phasing

of this topology, C_S / C_I requires that C_S be larger than C_I, reducing the feedback factor. An alternative topology is shown in Figure 7 in which the integrating capacitor is not only used to close the feedback loop around the OTA, but is also used as a second sampling capacitor to capture a sample of charge from the input. During the track mode (as shown in Figure 7) the two capacitors are shorted together in parallel, effectively making one large sampling capacitor. When the hold, or integration phase arrives, the switches shown in the figure are reversed and the charge on C_S is transferred to C_I. The closed loop gain of the circuit is now $\frac{C_S + C_I}{C_I}$. In other words, the new topology achieves a larger loop gain for the exact same feedback factor and bandwidth (a gain of one has been added to the old C_S / C_I gain term). Alternatively, the sampling capacitor can be made smaller for the same closed loop gain, but higher bandwidth. (e.g. C_S must equal C_I to effect a gain of two in the new topology, but in the previous design C_S had to be twice as large as C_I -- decreasing the feedback factor.) The final design of the SHA for the VGA is shown in Figure 8. The

center sampling switch across the op amp's inputs and an extra clock phase, ϕ_{S2}, have been added to provide better charge injection matching from the sampling operation (charge injection from the center switch will be less dependent on geometry matching than charge from two separate switches). Not shown in Figure 8 is the control for changing the gain of this stage. Figure 9 shows the single-ended version of how this control is accomplished. When the indicated switch is closed, then the circuit is essentially the same as the circuit in Figure 7 whose operation was just described as having a closed-loop gain of $\frac{C_S + C_I}{C_I}$. However, when the switch is opened, C_S is removed from the circuit. The closed-loop gain of the circuit now becomes C_I / C_I. Therefore, each stage of the final VGA consists of a SHA providing a gain of either $\frac{C_S + C_I}{C_I}$, or a gain of unity. This capacitive ratio can be chosen to be any value desired; however, increasing C_S also decreases the feedback factor and therefore the bandwidth (Equation 6 and Equation 9). Due to the extremely high speed requirements of the InfoPad environment, stages implementing only 3 dB each have been cascaded together to provide 12 dB of total gain. 3 dB of gain (or approximately 1.4x) requires a

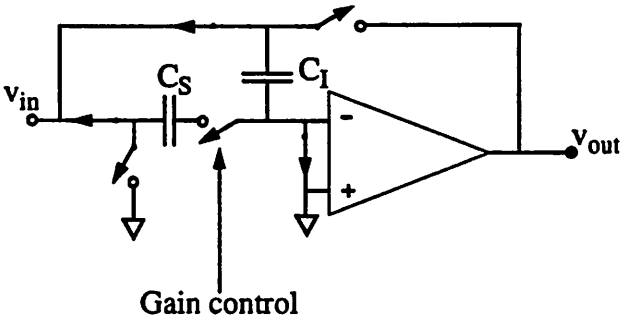

FIGURE 9. Addition of gain control (shown single ended)

C_S equal to a little less than half of C_I (again, the exact value of the gain is unimportant).

Finally, the SHA design requires common mode feedback. Figure 8 shows capacitive common-mode feedback that has been added with a switch for reset during the sampling phase. This feedback is essentially the same technique used when the C_I loop is closed (except, of course that it is common mode), since the capacitors feed back the common mode output to the tail current source of the OTA (see next section).

4.3 Operational Amplifier Design

The heart of the sample-and-hold amplifiers described in the last section is the operational amplifier. As mentioned before, the high-speed requirements of the system dictate a simple, high-bandwidth design for the OTA. While most CMOS sample-and-hold amplifiers (for use in a pipeline A/D for example) use two-stage amplifiers to achieve higher gain, the InfoPad VGA design does not have this requirement. Interstage gain amplifiers in a

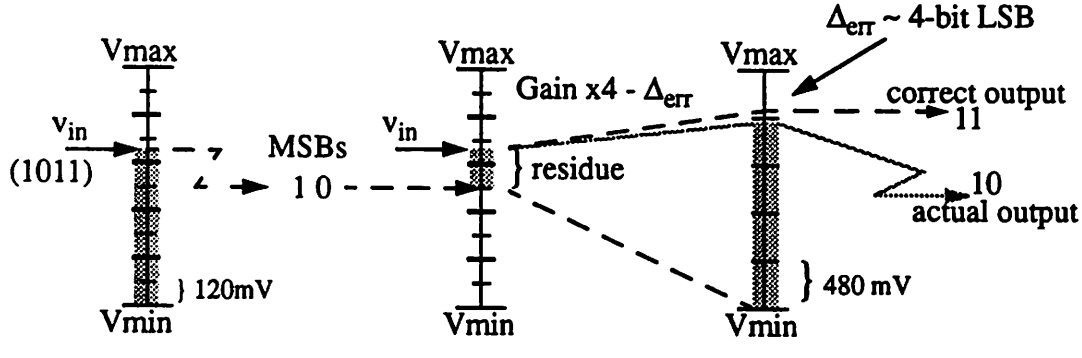

FIGURE 10. 2 bit -- 2 bit Pipeline Conversion Example with small gain error

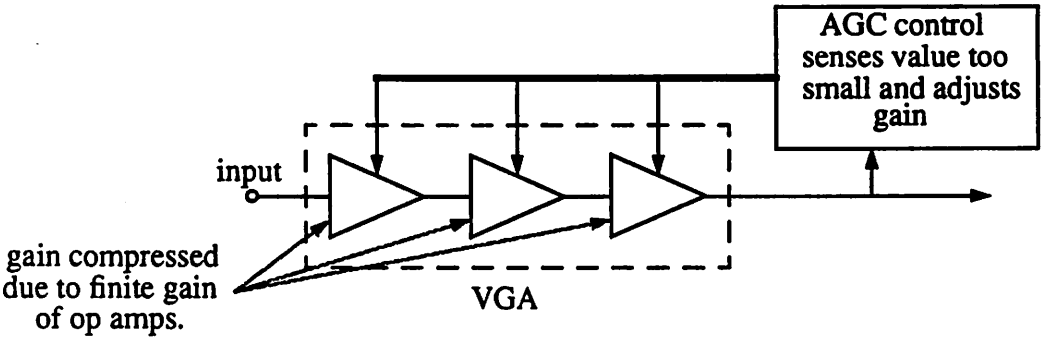

FIGURE 11. AGC compensation of gain errors by negative feedback control loop

pipeline A/D require gain accuracy on the order of the resolution of the entire converter (in other words, each stage of an N-bit pipeline A/D must have gain precisely controlled to one part in 2^N) [9]. However, as mentioned before in Section 4.2, the negative feedback provided by the overall control loop compensates for any error in the gain of individual stages of the AGC. Therefore, in the trade-off between gain and bandwidth, a lower-gain-but-higher-bandwidth design has been chosen. The effects of gain compression from finite OTA gain are illustrated in Figure 10 for a pipeline stage and in Figure 11 for the InfoPad AGC.

A single-stage telescopic cascode topology is the fastest known op amp topology available in CMOS technology. The use of common gate cascode transistors eliminates the Miller effect at the inputs, and the low impedance seen looking into the source of a cascode transistor means the circuit is essentially a single-pole system. While a folded cascode has the advantage of increased headroom capability, the inclusion of PMOS transistors in a folded design adversely affects the non-dominant poles.

This lower PMOS f_T limits the bandwidth when feedback is applied. Therefore, the telescopic cascode circuit in Figure 12 has been designed for use as an operational transconductance amplifier [3]. The input transistors and cascode transistors (M_{in+}, M_{in-}, M_{ncasc+}, and M_{ncasc-}) are n-type devices to maximize the bandwidth of the amplifier. The width of the input devices is determined by a settling time optimization analysis presented later in this chapter. PMOS transistors M_{psrc+}, M_{psrc-}, M_{pcasc+}, and M_{pcasc-} form a pair of active current source loads in order to achieve a high output impedance and therefore a high DC gain. The size of these devices is determined primarily by the required signal swing at the output of the OTA. The devices down the middle of the diagram in Figure 12 represent a high-swing bias circuit for the two NMOS cascode transistors. M_3 and M_4 are simply a cascoded PMOS current mirror to bias the two NMOS transistors. M_1 is forced to operate in the triode or linear range of operation by the diode connected transistor M_2. Therefore, M_1 acts as a source degeneration for M_2, and the W/L ratio of M_1 is chosen so that the voltage drop from drain to

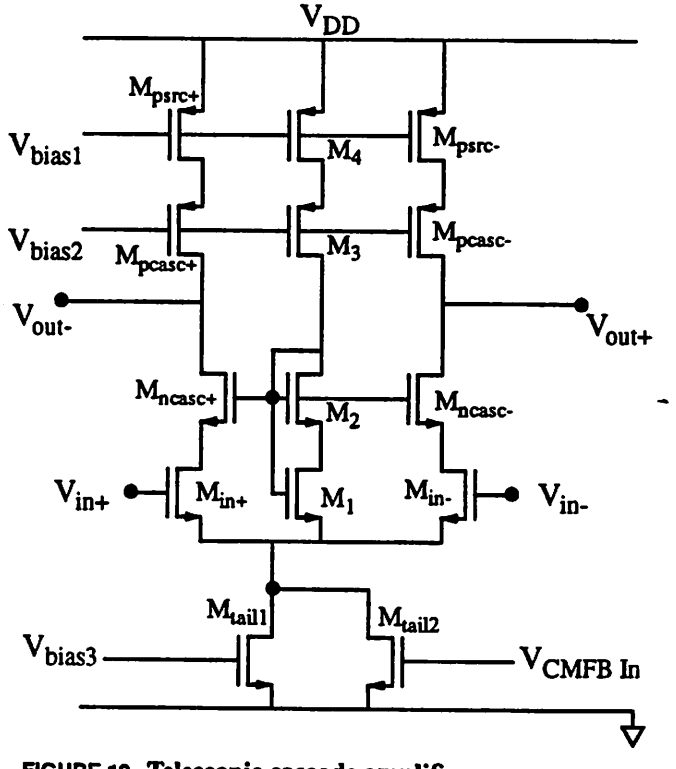

FIGURE 12. Telescopic cascode amplifier

source will match the desired V_{ds} across the two input devices. This value, V_{ds_input} should be as low as possible without pushing the input devices into the triode region of operation. Therefore, the final bias voltage at the gates of the NMOS cascode devices is designed so that the input transistors M_{in+} and M_{in-} have drain to source voltage:

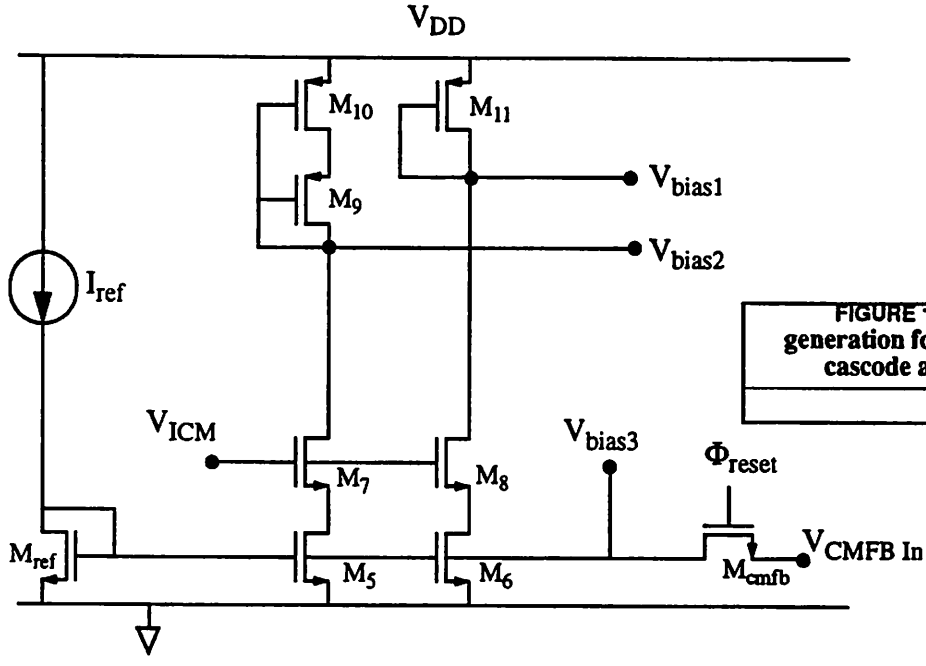

FIGURE 13. Bias generation for telescopic cascode amplifier

$V_{ds\_input} = V_{ds\_1} = V_{dsat} + V_{margin}$. A very aggressive margin of 150 mV has been chosen so that the available headroom for the signal to swing in the negative direction is maximized as long as M_2 is designed properly to match the V_{gs} drop of the cascode transistors. Therefore, the design equations for M_1 and M_2 are:

$$W/L_2 = \left(\frac{W/L_4}{W/L_{psrc}}\right)(W/L_{ncasc}) \quad \text{(EQ 10)}$$

$$I_{D1} = \mu C_{ox}(W/L_1)\left((V_{gs1} - V_T)V_{ds1} - \frac{1}{2}V_{ds1}^2\right) \quad \text{(EQ 11)}$$

Where Equation 11 can be solved for W/L_1 since I_{D1}, V_{gs} - V_T, and the desired V_{ds1} are known. The bias circuitry for the PMOS current sources have been designed with the same technique, as shown in Figure 13. Figure 13 also shows the tail current source bias generated from an off-chip current source flowing into an NMOS diode. This diode serves as a reference for all currents used by the OTAs. The tail current source of the amplifier simply mirrors the current reference, and transistors M_5, M_6, M_7 and M_8 also mirror the reference current for use in generating bias for the PMOS current source loads. M_7 and M_8 help to match the current being mirrored into the PMOS current sources with the current flowing in the tail source of the OTA by matching the V_{ds} across M_5 and M_6 to the expected V_{ds} across the tail source (this is critical for matching the currents through the devices -- the extremely poor λ of the process means I_{ds} depends strongly on V_{ds}). The tail current source, formed from transistors M_{tail1} and M_{tail2}, is split into two parts so that half of the current source can be used for common mode feedback during the evaluation phase

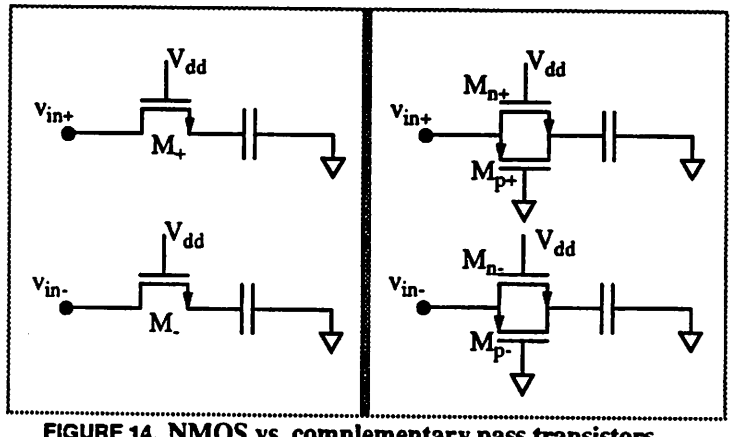

FIGURE 14. NMOS vs. complementary pass transistors

of operation (during the reset phase, both M_{tail1} and M_{tail2} are connected to the reference voltage). The use of only half the transistor in the common mode feedback loop brings down the loop gain around the loop, and helps ensure stability. It also helps protect the circuit from charge injection and ringing from turning on and off the reset switch M_{cmfb}.

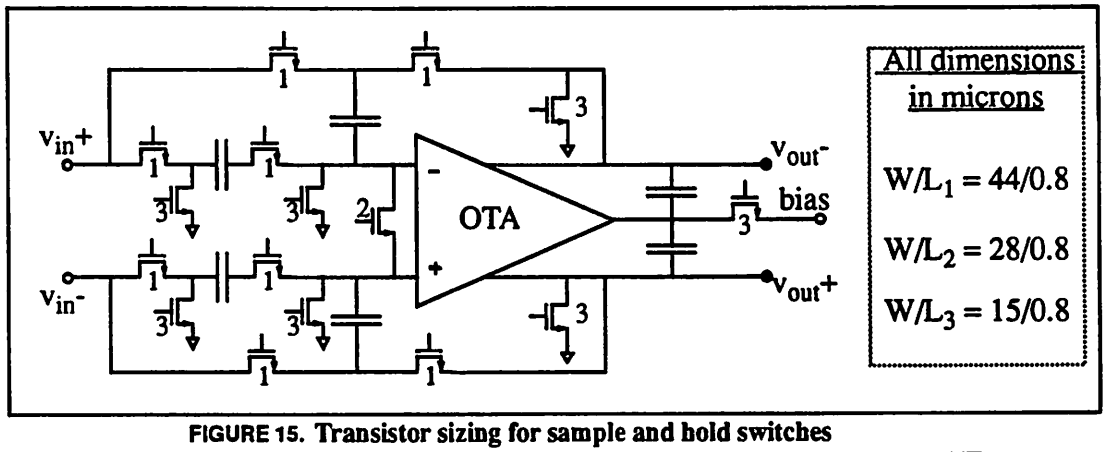

FIGURE 15. Transistor sizing for sample and hold switches

4.4 Optimizations

Almost any wireless system is designed with the expectation that the received signal will be small. The loss in signal power suffered from transmission through the air usually means that the signal seen by the antenna is significantly smaller than the baseband circuitry (the A/D) can detect. Invariably, amplifiers must be inserted in the receive path of the signal in order to gain the signal up to a detectable level. At the time of this writing, a typical A/D converter might expect an input signal which had been amplified to a maximum range of about +/- 1 volt or even larger. However, sustaining such a large signal swing, especially in a switched-capacitor circuit (e.g. the

InfoPad's VGA), requires extra power and area. Also, slewing may begin to hinder the speed of operation for the circuit, and headroom limitations in the OTA may begin to cause distortion or clipping unless a wide-output swing topology is chosen (e.g. a folded cascode). Therefore, it is clearly advantageous to apply only enough gain to the signal so that it meets the minimum power level so that the A/D can accurately resolve it. Adding more gain than this minimum amount can help relax the accuracy requirements for the A/D, but might incur all the potential headaches just outlined. Therefore, in a somewhat backwards design approach, the signal swing at the input of the A/D has been chosen to just meet the minimum detectable signal given the estimated offset characteristics of the A/D (allowing, of course, for some extra margin). The entire receive path of the CDMA radio has thus been designed for a maximum signal swing of +/- 250 mVolts (differentially, this is -0.5 volts to +0.5 volts).

Maintaining a maximum signal swing at a low level has another advantage. Typically, the pass transistors used as sampling switches for the SHAs would necessarily have to be complementary in nature because of the larger signal swing. In other words, if v_{in+} and v_{in-} differ by a significant amount, then the on resistance of the two switches (M_+ and M_- in Figure 16) may not match each other because of the different voltages biasing them (V_{dd} - v_{in}) -- this can disastrously decrease the input bandwidth of half the sampling path resulting in signal distortion and introducing a phase error. Therefore, by choosing an appropriate common mode bias voltage (v_{icm} = 1.7 to 2.5 volts in this case), all the switches in the SHA design (see Figure 8) may be constructed out of n-type devices. Not only does the elimination of the extra PMOS transistor for each pass gate simplify clock-

ing (a complimentary pass gate requires clk and $\overline{clk}$), but also increases the sampling bandwidth of the network by reducing parasitic capacitance at the drain and source nodes. The input bandwidth of the sampling network is determined primarily by the on resistance of the switches, and by the size of the capacitors used to sample the signal. In the case of Figure 15, the worst case input bandwidth is given by:

$$\tau = \left[(2R_{on1} \| R_{on1}) + \frac{1}{2}R_{on2} \right] (C_S + C_I) = \left(\frac{2}{3}R_{on1} + \frac{1}{2}R_{on2} \right) 3C_S \qquad \text{(EQ 12)}$$

Where R_{on} is determined by:

$$R_{on} = \frac{1}{\mu C_{ox}(W/L)(V_{gs} - V_T)} \qquad \text{(EQ 13)}$$

Clearly, minimizing C_S and maximizing W/L and (V_{dd} - V_{icm} - V_T) yields a faster input network. Unfortunately, the feedback factor defined in Equation 6 will decrease if C_I is made small relative to the parasitic and output load capacitance on the OTA. Therefore, a C_S = 150 fF and a C_I = 300 fF have been cho-

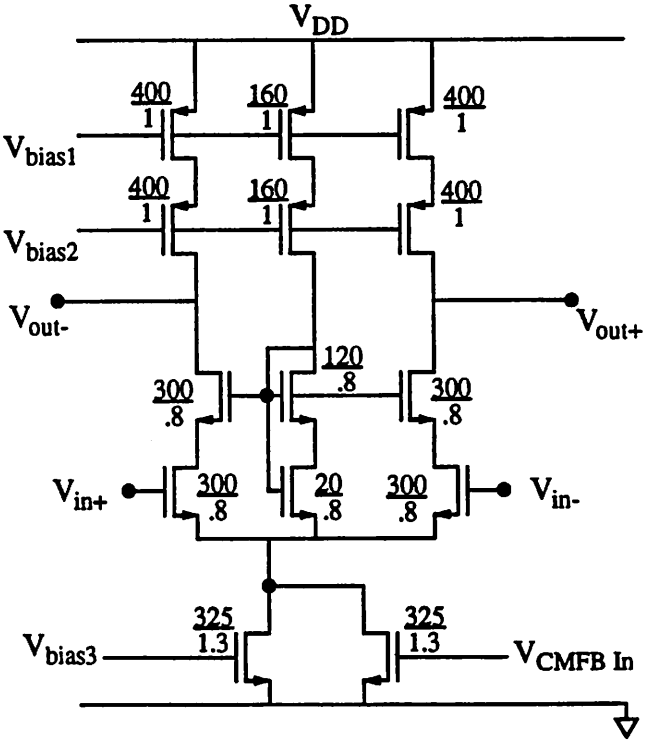

FIGURE 16. Transistor sizing for transconductance amplifier

sen. To meet the input bandwidth requirements of the system, V_{dd} has been chosen for 5 volt operation, with a V_{icm} = 2.5 volts (although the OTA can operate at a supply voltage of 3.3 volts, and a common mode voltage of 1.7, the pass transistors used in the switch array must be driven by a 5 volt clock in order to maintain a sizeable (V_{dd} - V_{icm} - V_T). Transistors have been sized for appropriate on-resistances, and their W/L ratios are shown in Figure 15. The resulting track bandwidth of the input network is on the order of 3 GHz! (Which was necessary for the sampling demodulator to admit a GHz bandwidth input, but was perhaps a little generous for the following stages).

Figure 16 shows the relative sizing of the devices used to implement the OTA. All dimensions are in microns. The numbers shown in the figure represent drawn geometries, and do not include lateral diffusion (L_D) or width shrinkage from implantation of the drain and source areas. As described in Section 2.3, the process used to fabricate the CDMA receiver chips is an asymmetric process -- all NMOS transistors undergo a mask shrink to improve maximum f_t of the n-type devices. Therefore, the actual drawn length of each NMOS transistor has been be decreased by 0.2 microns. (e.g. the input devices are 300/0.8 microns drawn instead of 300/1.0). The design process which yields the devices shown in Figure 16 proceeds as follows:

Recall that $g_m = \frac{2I_D}{(V_{gs} - V_T)}$ for an MOS transistor. Therefore, in order to achieve maximum g_m (and therefore, maximum speed) for a constrained I_D, the current density[1] of the device, V_{gs} - V_T, should be minimized without forcing the transistor into the subthreshold

region of operation. For the OTA used in the InfoPad's CDMA radio, a V_{gs} - V_T = 150 mV has been chosen. Once the current density has been fixed, the drain current through the device, $I_D = \frac{\mu C_{ox} W}{2 L} (V_{gs} - V_T)^2$, becomes proportional to the transistor's W/L ratio. Increasing the width of the input devices certainly improves the g_m of the OTA, but unfortunately the input and output parasitic capacitances also increase -- resulting in loss of feedback factor and an increased load that needs to be driven by the OTA. Equation 14 shows τ as a function of the input device sizes without taking the dependence of part of C_{in} on W/L.

$$\tau = \frac{C_L}{G_m f} = \frac{C_L (C_l + C_S + C_p + C_{in})}{C_l [\mu C_{ox} (W/L) (V_{gs} - V_T)]} \quad \text{(EQ 14)}$$

By taking the increased self-loading of a large device into account, Equation 14 becomes:

$$\tau = \frac{C_L (C_l + C_S + C_p + WLC_{ox})}{C_l [\mu C_{ox} (W/L) (V_{gs} - V_T)]} \quad \text{(EQ 15)}$$

Taking the derivative of Equation 15 with respect to W, and setting the result equal to zero, results in a W/L which corresponds to a local minimum for τ [10]. The result of the optimization performed on Equation 15 for this process is the two 300/0.8 micron input devices presented in Figure 16. The desired current level in the input devices can

1. Current density may be somewhat of a misnomer here since V_{gs} - V_T does not have the units of current per unit area. It does, however, refer to the fact that the V_{gs} - V_T of a MOS transistor is proportional to the square root of the drain current, I_D, divided by the W/L ratio of the device.

now be determined from the relation: $I_D = \frac{\mu C_{ox} W}{2 L}(V_{gs} - V_T)^2$. From this analysis, a total tail current of 2 mA has been chosen. The sizes of all other devices in the circuit are simply chosen to meet the headroom required from the output signal swing. Finally, the tail current source has been designed with a non-minimum channel length in order to increase its output impedance (and therefore, the CMRR).

The designs discussed in the previous sections have been used to implement a small four-stage VGA. Each stage introduces a gain of either 0 dB or 3 dB into the signal path -- resulting in a net controllable gain of 0 to 12 dB in 3 dB increments. The C_I and C_S used are 300 and 150 fF each respectively, and the transistor sizes are as shown in Figure 15 and Figure 16. See Section 6.1 for descriptions of the actual silicon die, and see Appendix A for relevant SPICE simulation decks.

SPICE simulations of the proposed VGA design (extracted from actual layout) revealed that the sample and hold stage met the speed and accuracy requirements dictated by the system while consuming only 10 mW of power per stage. However, when two or more stages are cascaded together, a kickback noise problem was encountered. The problem stems from the large parasitic capacitance associated with the bottom (closest to substrate) plate of the integrating capacitor C_I. At the end of each evaluation phase, C_I and its parasitic C_{Ip}, have charge equal to CV_{out} stored on them. As illustrated in Figure 7, when the next track phase arrives, C_I is switched to the input in order to sample the next value. Unfortunately, the charge stored on C_I and C_{Ip} is kicked back to the previous stage. The result is a signal dependent signal excursion at the beginning of the previous stage's set-

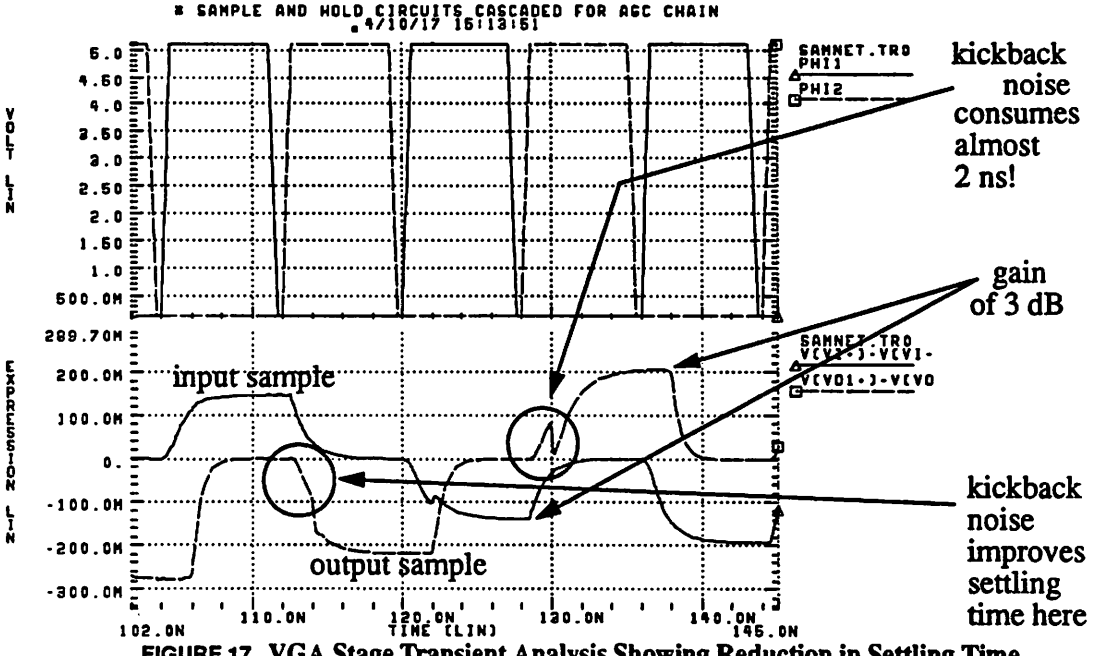

FIGURE 17. VGA Stage Transient Analysis Showing Reduction in Settling Time Due To Signal Dependent Kickback Noise.

tling curve. These kickback excursions severely reduced the amount of time available to the circuit to settle to its final value. A transient analysis of two cascaded stages of the VGA is shown in Figure 17. The clock rate has been slowed to 64 MHz because of the severe error introduced into the signal at full speed due to incomplete settling. See Appendix A for SPICE decks *samnet.sp* and *samnet.spice*. Simulations show the VGA stages settling to the required accuracy when being clocked at 1/2 the original rate (64 MHz).

CHAPTER 5

A to D Design

5.1 Introduction

One of the inherent properties of a spread spectrum system is the trade-off between bandwidth and accuracy. More specifically, a direct sequence spread spectrum (DS-SS) system such as the InfoPad radio trades off extremely high speed for reduced resolution. In other words, by dividing each transmit pulse into 64 smaller pulses (called 'chips') the radio hardware must now run 64 times faster than might previously have been required if this "spreading" had not been performed. However, by using a correlator (essentially a matched filter) on the received signal to "de-spread" it, the noise power introduced by quantizing the analog signal into a digital code is reduced by a factor of 64 (or more accurately, the signal to noise power ratio (SNR) is increased by a factor of 64). This factor is called the spreading gain of the DS-SS system, and effectively adds 3 bits of resolution to the analog to digital conversion function. In other words, if the system's signal to noise

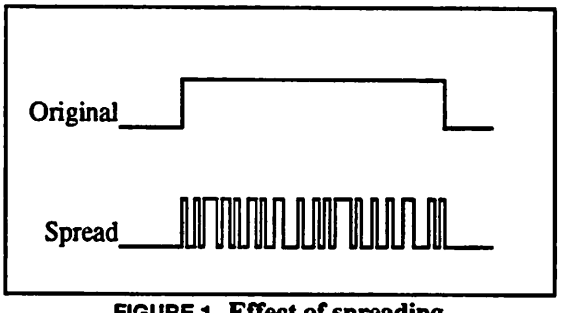

FIGURE 1. Effect of spreading a transmit pulse

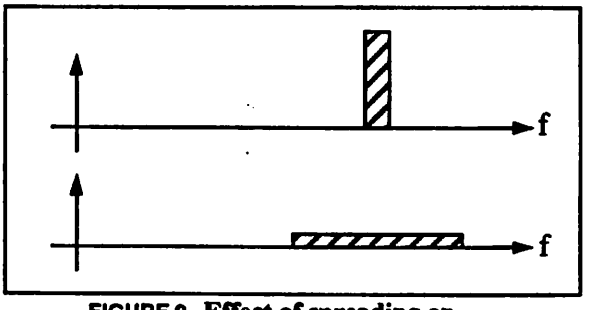

FIGURE 2. Effect of spreading on transmit power spectrum

requirements dictate that quantization noise can not exceed -77dB below the signal power, then the 12-bit A/D converter required for conventional systems only needs to have 9 bits of resolution in a spread spectrum environment (with a spreading gain of 64). Figure 1 and Figure 2 are rudimentary illustrations of the effects of spreading [11]. The spreading gain of the DS-SS matched correlator effectively adds 3 bits of resolution to the A/D converter; unfortunately, beyond a certain number of bits of resolution, the quantization noise becomes quite negligible compared to thermal noise and interference from other transmissions (the InfoPad system multiplexes several users into the same physical cell using orthogonal codes to distinguish one user from another -- in this type of system, called Code Division Multiplexing, each user receives interfering transmissions from all the other users). At the point where thermal noise and interference begin to dominate the signal to noise ratio, increasing the resolution of the converter provides no real benefit. Simulations in the U.C. Berkeley Ptolemy system have shown that an A/D converter with 4 bits of resolution (effectively 7 bits after de-spreading) is sufficient to bring quantization noise well below the interference noise floor [5]. As far as A/D converter technology goes, 4 bits could certainly be considered a fairly low resolution converter. Unfortunately, the spreading gain of the system also results in a 64-fold increase in sampling rate for the converter. Therefore, the rest of this chapter will detail the design of the 4-bit 128 Msamples/sec A/D converter for the InfoPad CDMA receiver with achieving low power as a key design constraint.

5.2 Architecture

Of all the architectures for high speed A/D converters[1], the flash architecture is the fastest. An N-bit flash converter, as described in Section 3.1, performs conversion simply by placing 2^N comparators in parallel and determining where, among the 2^N levels, the input lies. For a resolution as small a 4 bits, a flash architecture is a feasible option for a 128 Msample/sec converter. Unfortunately, the amount of hardware and power required for an N-bit flash A/D increases exponentially as 2^N, making a full flash design less attractive because of the extremely low-power requirements of the system. However, even with a very high speed comparator design (see Section 5.3), exploration of subranging and pipeline A/D converters revealed that the 7.8 ns cycle time was insufficient for performing the extra functions required in these architectures (see Section 3.2 and Section 3.3, for a description of subranging and pipeline converters). Furthermore, the addition of a high speed sample and hold amplifier (SHA) in between stages of a pipeline converter increases the power consumption due to the static current in the operational amplifier. In fact, the *increase* in power consumption added by the insertion of a SHA in a pipeline far outweighs the power *saved* by reducing the number of comparators from 15 to 8 for a 4 bit converter[2]. The conclusion that might be drawn from this analysis is that the correct architecture to choose for the A/D converter topology is a flash. Indeed, for a **stand-alone** 4-bit CMOS converter

1. Although certain other A/D topologies, including delta-sigma and successive-approximation converters, may have high clock rates, the term "high speed" here refers specifically to a class of converters called 'nyquist rate' converters. Unlike oversampled converters (e.g. delta-sigma), this class of converters processes one data sample per clock period, achieving the fastest possible data conversion rate for a given clock speed.
2. Note that for higher resolution converters, the addition of a pipeline stage might very well reduce the overall power consumption of the A/D since the number of comparators increases exponentially with the number of bits, N.

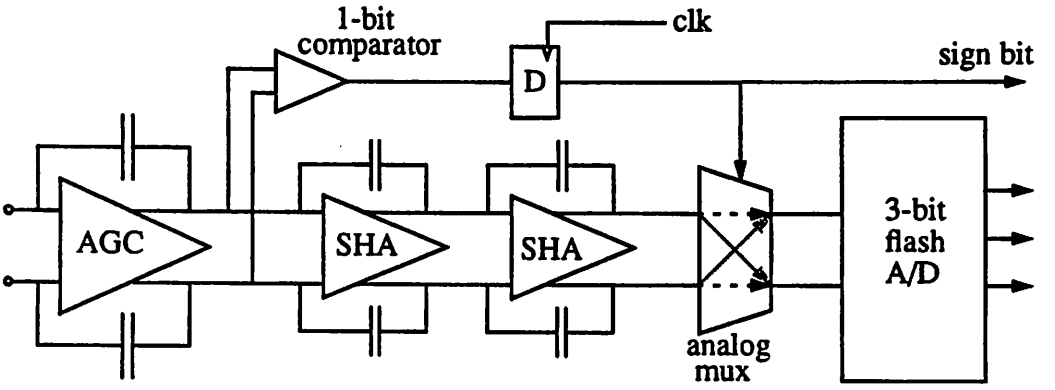

FIGURE 3. Modified 1-bit to 3-bit pseudo pipeline architecture

operating at 128 Msamples/sec, the flash design would almost certainly be the architecture of choice. However, by combining the A/D function with the AGC function, further improvements were possible. The final A/D design for the CDMA receiver pipelines one of the four bits, resulting in a 1-bit to 3-bit pipeline converter. This design is depicted in Figure 3. Since the InfoPad CDMA radio uses a discrete-time AGC immediately before the A/D converter. Each sample-and-hold amplifier in the AGC looks exactly like the interstage gain amplifier of a pipeline A/D. Therefore, the hardware to perform most of the pipelining function is already in place and could be used by the A/D **at no extra cost**. The AGC consists of several stages of gain, some of which must be small (for a reasonably fine gain resolution). Therefore, the interstage gain in the pipeline can be made from multiple sample-and-hold amplifiers, allowing the function of generating of a residue (input signal minus analog version of MSB output) to be pipelined itself. It was found that by placing **two** stages of 3dB gain in between the first comparator and the remaining 3-bit flash A/D, the residue generation

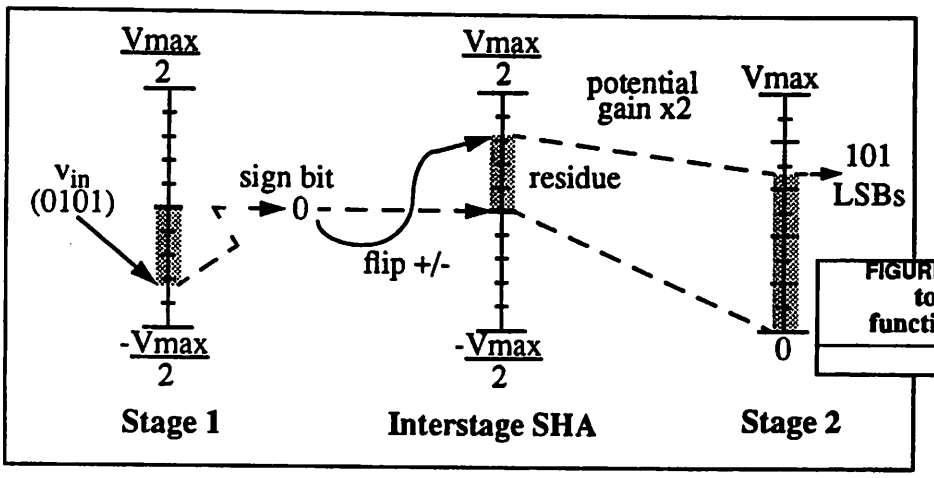

FIGURE 4. Modified 1-bit to 3-bit pipeline functionality illustrated

function could be split into two stages -- allowing enough time to generate a residue at 128 MHz.

The receive path is fully differential. Therefore, by pipelining a **single bit** of the A/D, the first stage (MSB) comparison becomes essentially a detection of the **polarity** of the incoming signal. Therefore, the 1-bit MSB subtraction function in a regular pipeline can be changed to a polarity switch (just cross-connected pass gates) based on the results of the first comparator. This concept is illustrated in Figure 4, and really represents an absolute value function on the input signal. This seemingly insignificant difference between 1/2 range subtraction and polarity swapping has a surprising effect -- since the **magnitude** of the signal has been maintained, the interstage amplifiers of the pipeline can sustain the gain compression described in Section 4.2 without any detrimental effect. In other words, since the absolute value function must be performed by the AGC control block anyways, the polarity flip does not add a non-linearity into the AGC feedback loop -- allowing the negative feedback to reject any gain variation in the amplifiers.

It should be mentioned here that the topology in Figure 3 is not *quite* a true pipeline A/D. A true 1-bit to 3-bit pipeline topology would **require** a gain of exactly 2 after the first comparator stage. As mentioned before, the two SHAs are really two variable gain stages (0 or 3 dB each) in the AGC. Therefore the total interstage gain between the two comparator stages can be either 0 dB (no gain), 3 dB, or 6 dB (gain of 2) based on the RSSI (received signal strength indicator) detected by the AGC. Indeed, since a fair amount of gain droop can be tolerated in the system, the finite DC gain of the OTAs may cause the total gain to be even less than 3 or 6 dB. Fortunately, the variability of the interstage gain does not have a significant impact on the design. Since the feedback effect of the AGC control loop tries to force the signal at the **output** of the AGC to be of constant power, the reference ladder for the last stage of the A/D is fixed. The interstage sample and hold amplifiers between the two stages may be set to provide no gain at times. When this situation exists, offset requirements of the first stage comparator are identical to those of the last stage. However, when 6 dB of gain exists between the two stages, the signal (and therefore, the allowable offset) at the first stage is half that seen in the last stage. As with a traditional pipeline, overranging comparators and digital correction could be implemented to fix any error made by excessive offset in the first stage. However, as will be described shortly in Section 5.3, the design of the single sign-bit comparator has a smaller offset characteristic than those of the other comparators. Thus, even when the signal swing at the input to the first stage of the pipeline is 1/2 of full range, the converter performs without incorrect codes even without the benefit of digital correction.

5.3 Comparator Design

The heart of any analog to digital conversion circuit is the comparator. The comparator performs the quantization function of the A/D by making a decision about the input signal relative to some fixed reference. The most basic of converters, the flash A/D, simply compares the input to each of the 2^N possible discrete values between the maximum allowed input and zero. Each of the comparators decides if the input is larger or smaller than one of the reference levels, resulting in a code of '1's and '0's which can be decoded into a digital word representing the input's magnitude. In this situation, the factor which fundamentally limits the resolution of the A/D converter is the minimum resolvable signal that the comparator can correctly make a decision upon. This key parameter is determined by several different characteristics of the comparator design, including speed of regeneration, overload recovery, and random offset. The primary difficulty in achieving high accuracy in a comparator is the inherent random offset associated with any differential structure. A differential pair, as shown in Figure 5 ideally has a differential output of zero only when the inputs are exactly equal. However, since the devices and parasitics on each side of the layout can not match each other exactly, this is never the case. The input offset voltage, V_{os}, is the input voltage that compensates for these non-idealities, and brings the outputs to zero. V_{os} is dependent on process variations, temperature gradients, and geometry mismatches of the design. For the simple MOS diff pair in Figure 5, V_{os} is given by [12]:

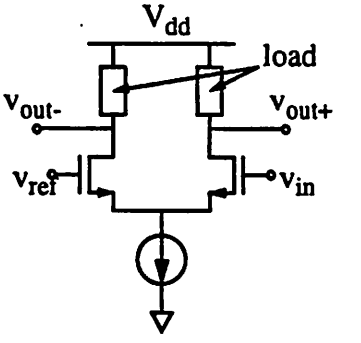

FIGURE 5. Basic Diff Pair

$$V_{os} = \frac{\Delta V_T}{V_T} + \frac{(V_{gs} - V_T)}{2}\left[\frac{\Delta (W/L)}{W/L} + \frac{\Delta Load}{Load}\right] \quad \text{(EQ 16)}$$

An error in the comparator decision can be made if $|V_{in} - V_{ref}|$ is less than or equal to $|V_{os}|$. As always, a trade-off exists between speed and accuracy. Therefore, the high speed throughput required of the A/D converters in the CDMA radio prohibits the use of most techniques for combating the offset problem (e.g. offset cancellation, large preamplification). However, a comparator topology has been proposed which achieves a very good combination of speed and accuracy and is shown in Figure 6 [13].

The comparator, as shown in Figure 6, has a single-ended input and therefore needs to be modified for the InfoPad's differential system. This could be accomplished through the use of a capacitive input sampling network included immediately preceding the

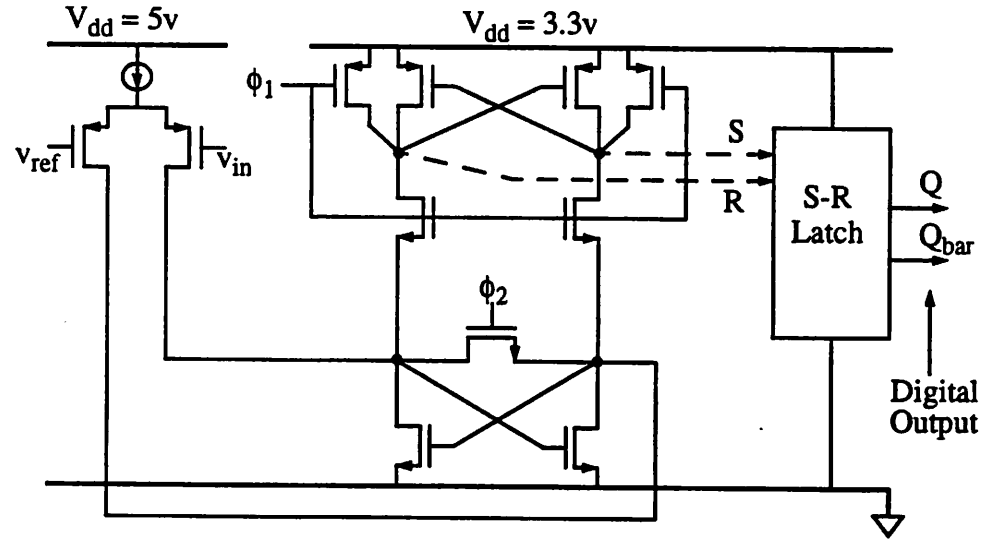

FIGURE 6. High speed, low offset comparator (after [13])

comparator, such as the one shown in Figure 7. Much like the sample and hold amplifiers described in Section 4.2, the network captures a sample of charge proportional to v_{in+} - v_{icm} on the sampling capacitors during ϕ_2. When ϕ_1 closes, one side of C_{S+} is shorted to v_{ref+}, forcing the other side of the capacitor to v_{in+} - v_{icm} - v_{ref+}. Similarly, the other side of the circuit generates v_{in-} - v_{icm} - v_{ref-}. Since v_{icm} is a common mode voltage, the diff pair generates a differential output current proportional to (v_{in+} - v_{in-}) - (v_{ref+} - v_{ref-}). Therefore, the inclusion of this additional circuit has a two-fold purpose. Not only does it provide an efficient differential-to-single ended conversion for the comparator, but it also performs a rudimentary sample-and-hold operation which might be useful if the A/D is not preceded by an active sample-and-hold circuit.The extra sampling operation provided by the ISN of Figure 7 is certainly not needed in the CDMA radio, since the input signal has already been sampled and held. Furthermore, a network which depends on large, very precisely matched capacitors becomes very unattractive in a standard digital CMOS process. Not only does the addition of large capacitors severely affect the input bandwidth of the A/D, it also becomes very costly in terms of area when the process used does not support a second layer of polysilicon (see Section 2.3). As mentioned before in Section 4.2, precision capacitors for the CDMA radio have been created using the dielectric between the first-to-second and/or second-to-third metal layers. These capacitors consume so much area, and

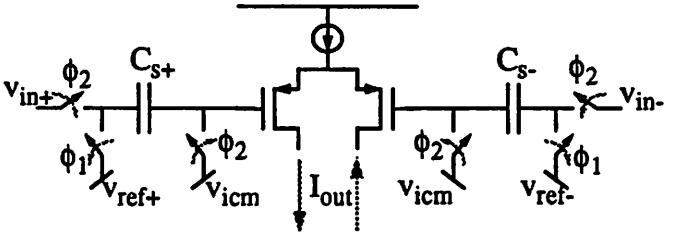

FIGURE 7. Input sampling network (ISN)

incur so much parasitic that the prospect of creating $2*2^N$such capacitors for an N-bit flash converter becomes extremely unattractive. Therefore, a modification of the original comparator has resulted in a design which is capable of accepting a fully differential input, and comparing it to a differential reference without the benefit of an input sampling network. The proposed comparator topology is shown in Figure 11. The original PMOS differential pair input has been replaced with a modified double-differential pair topology shown in Figure 8. The output of the circuit are given by the difference between the two currents I_{out_r} and I_{out_l}:

$$I_{out} = \left(\frac{-g_m}{2}\right)((v_{in-} - v_{ref-}) + (v_{ref+} - v_{in+})) - \left(\frac{-g_m}{2}\right)((v_{ref-} - v_{in-}) + (v_{in+} - v_{ref+})) \quad \text{(EQ 17)}$$

Therefore, if the bias currents match exactly, the output current is proportional to $(v_{in+} - v_{in-}) - (v_{ref+} - v_{ref-})$, which is exactly the output generated with the input sampling network. Notice that this topology does not use the differential pairs in the traditional manner. v_{in+} is not being compared directly with v_{in-}. A more standard topology is shown in Figure 9. The differential output current is identical to the output of the modified circuit given in Equation 17, but there is a significant difference between the two. Suppose a full range input signal is being compared to

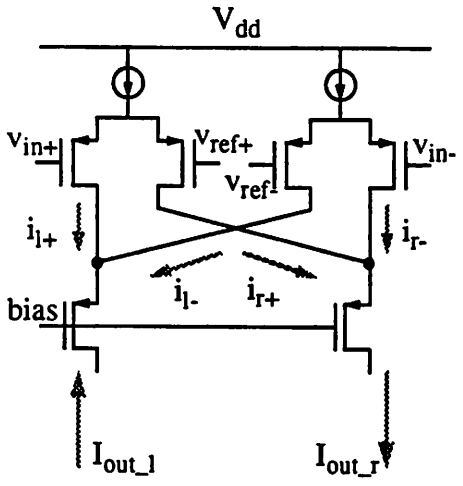

FIGURE 8. Modified doubly-differential input stage

the maximum reference (i.e. v_{in+} - v_{in-} is at its maximum value, as is v_{ref+} - v_{ref-}). In order for the differential pairs to stay in the linear range of operation (i_{out} proportional to differential input voltage), the V_{gs} - V_T bias for the input devices must be larger than one-half the differential input (see Figure 10). In other words, if the devices in the diff pair were biased with a V_{gs} - V_T = 200mV, and 500mV were to be applied across the input

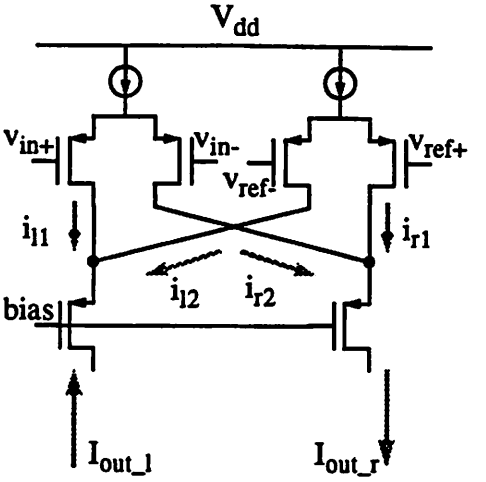

FIGURE 9. Standard double-differential pair stage

terminals, then one side of the diff pair would be completely shut off, while the other would carry the entire current from the tail source. Unfortunately, when trading off bandwidth for power, V_{gs} - V_T should be kept as small as possible (without forcing the devices into subthreshold operation) to achieve the highest bandwidth for the smallest power (since $g_m = \frac{2I_D}{(V_{gs}-V_T)}$, this is really a matter of maximizing g_m for a given current level).

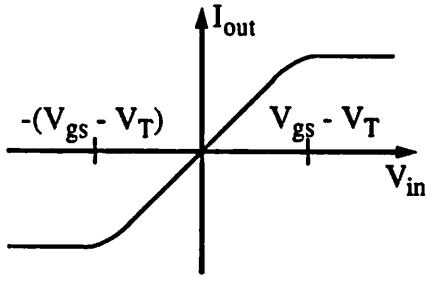

FIGURE 10. Diff pair transfer characteristic (I_{out} = A*tanh(v_{in}))

Therefore, making the V_{gs} - V_T bias as large as the full scale input voltage swing adversely affects either the speed or the power consumption of the circuit. (a large V_{gs} -V_T also reduces the available headroom on the input stage.) The proposed design in Figure 8 however, does not have this problem. Since v_{in+} is paired with v_{ref+}, and v_{in-} is paired with v_{ref-},

both differential pairs will be well within their linear range of operation even when both the input and the reference are near their maximum values. While it is true that the diff pair will become unbalanced when the input differs greatly from the reference voltage, this is of little importance because a comparator is only interested in the signal when it is near the crossing point with the reference.

The final comparator topology is shown in Figure 11. The input stage, which has just been described, performs a conversion from a doubly-differential voltage to a single differential current. PMOS cascode transistors have been added to the input stage to both improve the output impedance, as well as to eliminate the Miller effect at the input. The cascode devices also help to improve the regeneration speed of the comparator since the parasitic capacitance on the drains of the cascodes can be reduced by

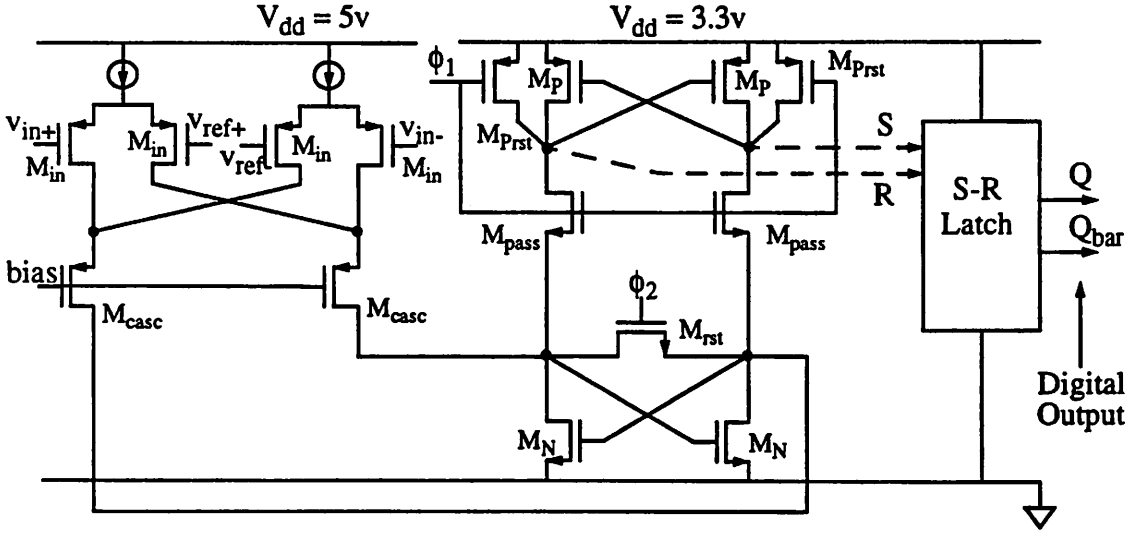

FIGURE 11. Final high-speed differential comparator

undersizing the devices (the input devices must necessarily be larger to achieve a higher g_m). Therefore, the input stage can be regarded as a single-pole voltage-to-current converter with a good high output impedance. Keeping this in mind, the operation of the rest of the circuit can now be analyzed. The comparator requires two clock phases which are shown in Figure 12. The two phases must be non-overlapping, a fact which is common for many A/D converters, but is especially important for the operation of this particular topology (as will be explained shortly).

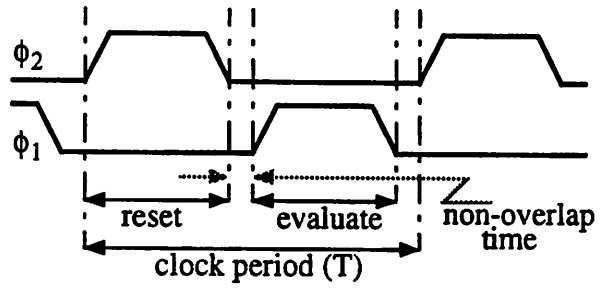

FIGURE 12. Comparator clock phases

The first phase of operation can be called the reset phase and is illustrated

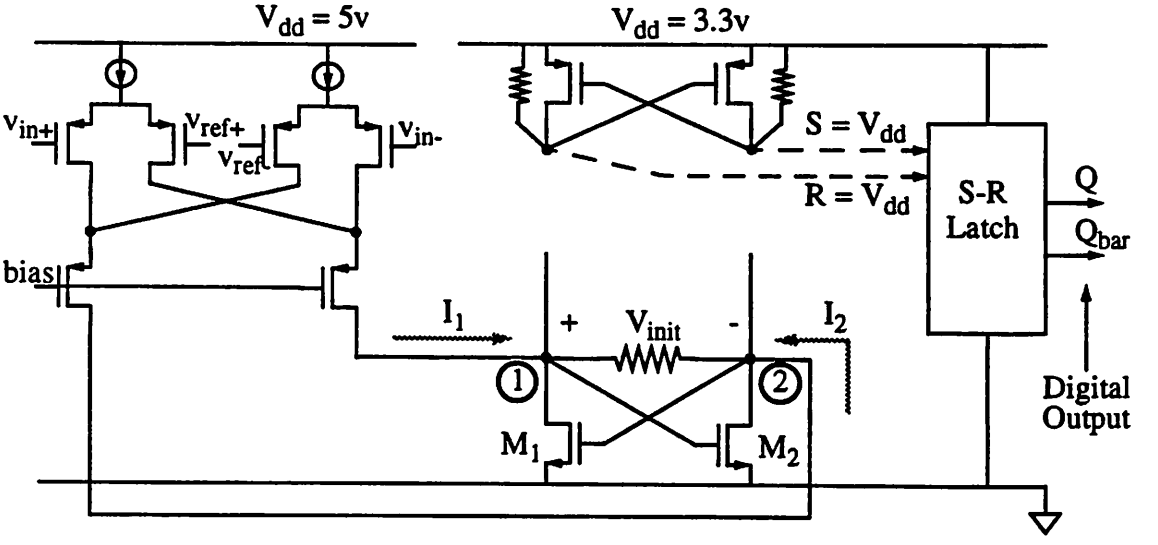

FIGURE 13. Comparator during reset phase of operation ($\phi_1 = 0$, $\phi_2 = V_{dd}$)

in Figure 13. The output nodes, which go to the S-R latch, are reset to V_{dd}. During this time, the S-R latch maintains its last value. Nodes 1 and 2 are shorted together by an NMOS switch with on resistance R_{on}, providing a good overload recovery for circuit[1]. Although M_1 and M_2 are connected in a positive feedback loop, the loop gain is forced to be < 1 by the reset switch -- making the two transistors look diode connected. However, the input stage is acting as a differential current source, forcing current down M_1 and M_2. The differential current flowing down the two devices will create a differential voltage at the two drain nodes (in other words, each device acts as a $1/g_m$ impedance load). Therefore, the voltage shown as v_{init} in the figure is proportional to the comparator's input voltage by the g_m ratio of the input devices to the NMOS latch devices (M_1 and M_2). Therefore, the function of the reset phase is to provide a good recovery from the previous sample and to set up an initial voltage proportional to the input (but smaller) as the starting point for regeneration of the NMOS latch. At the end of the reset phase, clock ϕ_2 goes to zero. After the falling edge of ϕ_2, there is a brief period of time during which both ϕ_2 and ϕ_1 are low. This 'non-overlap' period is typically regarded simply as a separator between phases of operation which guarantees that one phase will not interfere with the operation of another. However, at such a high frequency of operation, the brief non-overlap period represents a significant percentage of the clock period, and really can not be sacrificed as an idle period. The state of the proposed comparator during the period when both clocks are low is shown in

1. Overload Recovery refers to a comparator's ability to recover from evaluating a sample and then correctly evaluate a subsequent value.

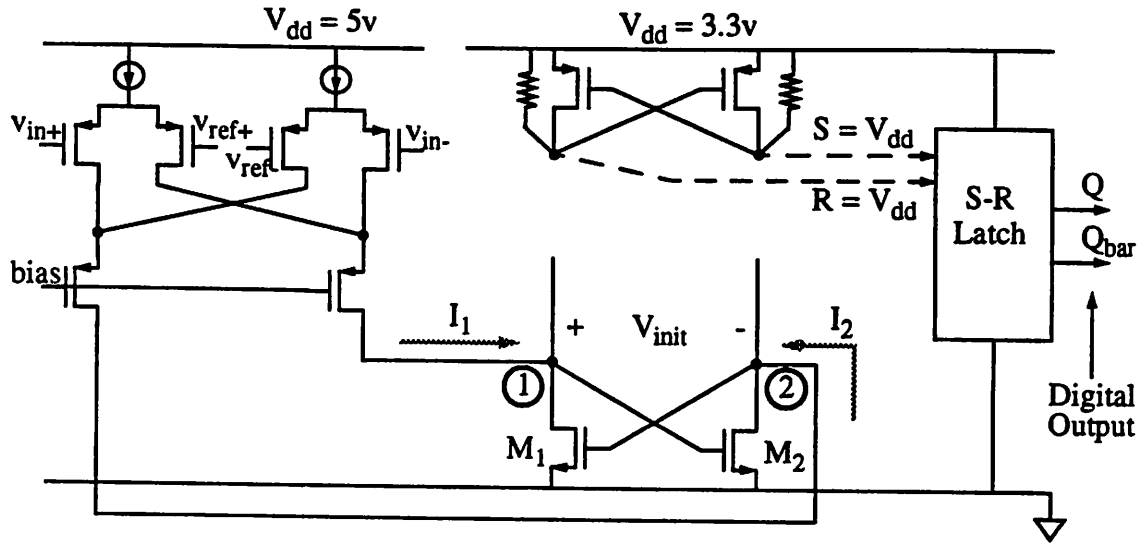

FIGURE 14. Comparator during reset phase of operation ($\phi_1 = 0, \phi_2 = 0$)

Figure 14. The reset switch across nodes **1** and **2** has been opened. Therefore, the differential current being provided by the input stage no longer sees two diode-connected loads. In fact, if the voltage v_{init} is small then M_1 and M_2 are biased at about the same level so that they act as a pair of NMOS active loads. If the initial voltage, v_{init}, is larger than a few tens of millivolts, then the positive feedback of the latch will force the two nodes to split apart further towards either supply. However, if v_{init} is small (corresponding to an input voltage very close to the reference voltage) then the currents flowing from the input stage will see the output impedance, r_o, of M_1 and M_2. Therefore, the voltage gain of the circuit changes from $A_{v1} = -\left(\frac{g_{m\_input}}{g_{m\_1,2}}\right)$ to $A_{v2} = -(g_{m\_input} r_{o\_1,2})$. While the first term, $g_{m\_input} / g_{m\_1,2}$, is actually less than one, the second term represents an amplification of the signal. The voltage across nodes **1** and **2** will begin to approach $v_{in} * A_{v2}$ until either the non-overlap

period ends (and the evaluation phase begins), or the voltage between the two nodes becomes large enough for the positive feedback of the latch to begin to regenerate it. It is not uncommon for comparator topologies to include a pre-amplifier (in the form of a differential pair) immediately before a latch because the preamp reduces the input referred offset of the latch by its gain. (A latch has a fairly bad offset characteristic because the positive feedback regeneration grows exponentially with time. Therefore, a small signal will start out with a slow rate of change.) The proposed topology folds the preamp and the latch together by using the gain achieved during the non-overlap period to reduce the input referred offset of the latch once the evaluation period begins. The evaluation phase of operation begins when ϕ_1 rises (see Figure 12). The two PMOS switches which shorted the output nodes to the supply are opened, and the two NMOS switches connecting the PMOS latch to the NMOS latch are closed. The resulting circuit is simply two inverters connected in positive feedback (a full latch) which regenerates the signal to near-digital levels. The S-R latch has been added to both bring the final output to a full digi-

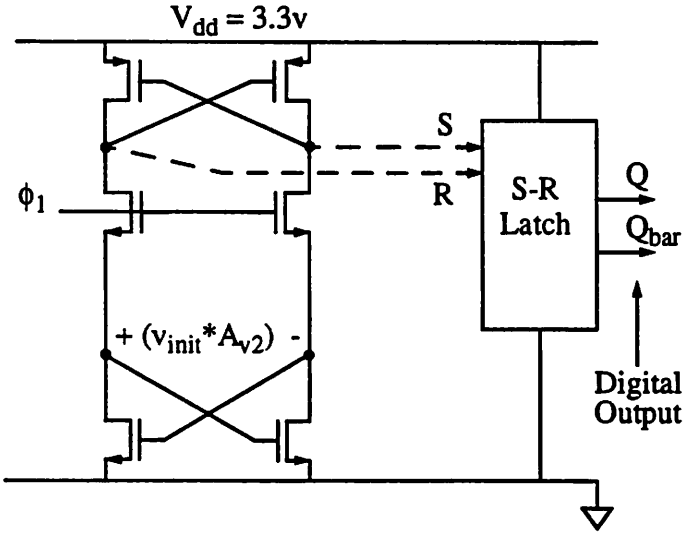

FIGURE 15. Comparator at beginning of evaluation phase of operation ($\phi_1 = V_{dd}$, $\phi_2 = 0$)

tal swing, as well as to hold the comparator's last output value during the reset phase. The final combination of doubly-differential input stage, high-impedance cascode, non-overlap period preamplification, and S-R latch results in a comparator with low offset, good overload recovery, very little kickback noise, low power, and high speed [13].

5.4 A to D Optimizations

Section 5.3 explains the advantage achieved from using the modified input stage shown in Figure 8 over the standard double-differential stage of Figure 9. The limited linear range of the differential pair limits the ability of the circuit in Figure 9 to compare a large input to a large reference voltage. The improved design in Figure 8 avoids this limitation because it places the comparator switching point exactly in the middle of the diff pair's linear range. However, if the common mode voltage of the input is different from the common mode voltage of the reference, then the switching point no longer falls exactly in the

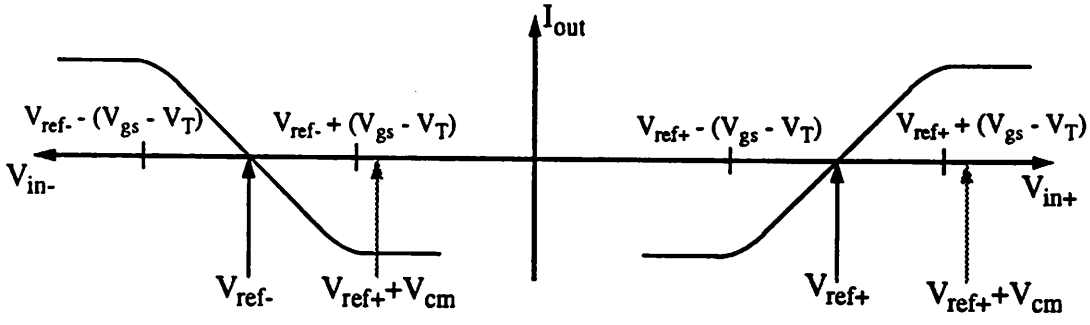

FIGURE 16. **Effect of common mode shift on output of modified input stage**

middle of the diff pair transfer function. This concept is illustrated in Figure 16. The differential output current is given by the sum of the two currents given by the curves shown

in the figure. However, if the common mode voltage, V_{cm}, shifts the comparison point outside the linear region of the circuit, as shown by the light grey arrows in Figure 16, then the differential output near the switching point will be zero, regardless of the input's value. In other words, the modified topology of Figure 8 works well even for large signal swings, but begins to fail if the common mode difference between input and reference varies significantly from zero. Therefore, the V_{gs} - V_t bias of the input devices must be chosen so that the circuit will remain linear during the largest expected common mode excursion. Unfortunately, the input offset voltage of an MOS differential pair is given by:

$$V_{os} = \Delta V_t + \left(\frac{V_{gs} - V_t}{2}\right)\left[\frac{\Delta (W/L)}{W/L} + \frac{\Delta Load}{Load}\right] \quad \text{(EQ 18)}$$

Clearly, the offset due to geometry mismatch in the devices increases with V_{gs} - V_t. Furthermore, the g_m of the input stage is $\frac{2I_D}{V_{gs} - V_t}$. Not only can the g_m for a given current can be maximized by reducing V_{gs} - V_t, but the offset can be reduced in this manner as well. Therefore, the bias on the input devices must be carefully designed in order to have a minimal V_{gs} - V_t, while still ensuring some robustness against common mode excursions. To this end, an input V_{gs} - V_t = 200mV has been chosen. In order to meet the stringent power budget allocated for the A/D, a total tail current of 40 µAmps was allowed (10 µA flowing though each input transistor). Using the drain current equation for an MOS transistor, $I_D = \frac{\mu C_{ox} W}{2 L}(V_{gs} - V_T)^2$, the W/L sizing for the input devices can be determined. The choice of 8µm/1µm geometry input PMOSes leads to

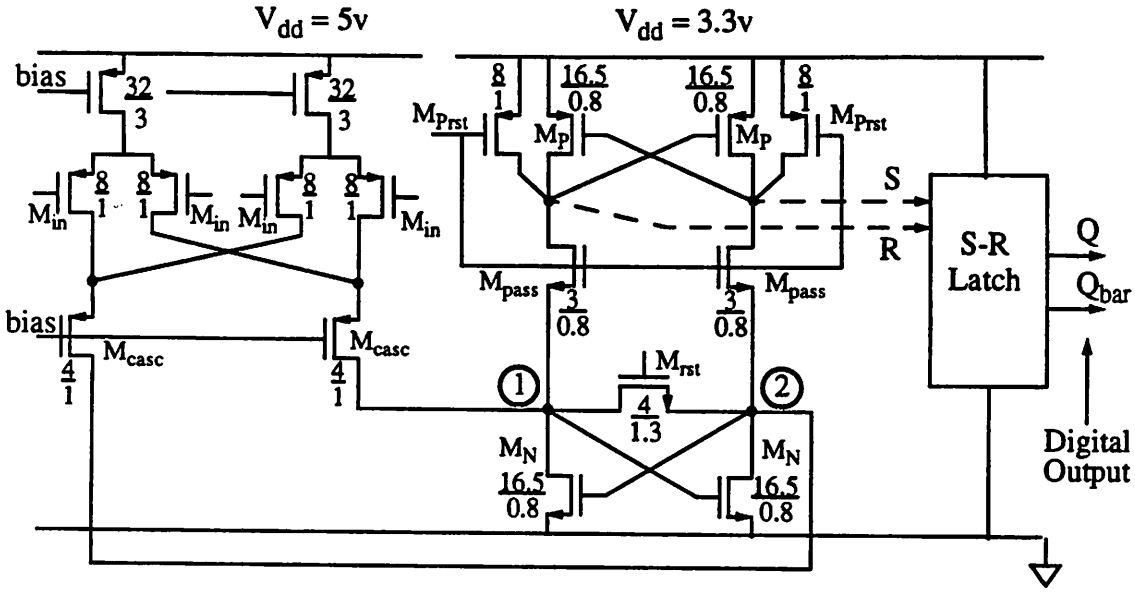

FIGURE 17. Device sizes for comparator

an estimated worst case input offset of 25-30 mV for the comparator, which is just below half of an LSB ($\frac{\Delta(W/L)}{W/L} \cong 0.05, \Delta V_t \cong 10mV$). Once the input device dimensions are determined, the cascode transistors can not be far behind, lest the nondominant pole at the source of the cascode device become significant. Therefore, the two cascode transistors have been designed to be half the size of the input devices in order to reduce the capacitance contributed to the NMOS latch. The sizes of the various transistors used in the comparator are shown in Figure 17. The NMOS latch transistors should be made as large as possible in order to achieve a faster regeneration speed by increasing current drive. However, the parasitic capacitance at the two nodes **1** and **2** is partially determined by the

width of transistors M_N. Therefore, an optimum transistor width can be found from the time constant equation:

$$\tau = \frac{C_{1total}}{g_{mN}} = \frac{C_{parasitic} + \alpha W_N}{\sqrt{2I_N \mu C_{ox}(W_N/L)}} \quad \text{(EQ 19)}$$

Where α is the capacitance per unit width contributed to node **1** (or node **2**) by the two M_N transistors. Optimization of Equation 19 yields:

$$W_N = \frac{C_{parasitic}}{\alpha} = 16.5\mu m \quad \text{(EQ 20)}$$

As described in Section 5.3, the initial voltage from which regeneration will start is determined by:

$$V_{init} = -\left(\frac{g_{min}}{g_{mN} - (2/R_{onrst})}\right)(V_{in} - V_{ref}) \quad \text{(EQ 21)}$$

Therefore, the R_{on} of the NMOS reset transistor must be made large enough to ensure a sizeable initial signal voltage, but small enough to provide a good overload recovery during the reset phase. Since the primary purpose of the reset switch is to drop the positive loop gain of the latch below one, the design equation becomes:

$$(-g_{mN}R_{onRst})\,(-g_{mN}R_{onRst}) < 1 \quad \text{(EQ 22)}$$

Equation 22 evaluates the loop gain around the latch. By taking the square root of both sides, and substituting for g_m and R_{on}, the relative sizing of transistors M_N and M_{rst} can be determined.

$$\left(\sqrt{2\mu C_{ox} I_N (W/L)_N}\right)\left(\frac{1}{\mu C_{ox}(V_{gs}-V_t)_{rst}(W/L)_{rst}}\right) < 1 \qquad \text{(EQ 23)}$$

$$(W/L)_{rst} > \frac{\sqrt{2\mu C_{ox} I_N}}{\mu C_{ox}(V_{gs}-V_t)_{rst}}\sqrt{(W/L)_N} \qquad \text{(EQ 24)}$$

For the one micron process used in the fabrication of the CDMA receiver chips, Equation 24 becomes:

$$(W/L)_{rst} > \left(\frac{1}{6}\right)\sqrt{(W/L)_N} \cong \frac{2}{3} \qquad \text{(EQ 25)}$$

A W/L_{rst} = 4/1.3 microns was chosen for the reset switch, but probably could have been more optimally designed as a smaller (i.e. lower W/L ratio) device.

Device sizes for the other transistors in the circuit have less impact on the comparator's performance. NMOS transistors M_{pass} were sized to minimize the parasitic capacitance they would add to the n-latch. The PMOS latch devices, M_P, were sized to match the current drive of the NMOS latch transistors as is done in any digital design (note that once the evaluate phase arrives, the n-latch combines with the p-latch to effectively make two cross-coupled digital inverters -- a standard digital latch).

It should be mentioned here that the sign bit comparator performs a comparison against a zero-valued reference voltage. In other words, this one comparator does not need four inputs, since it only determines if v_{in+} is <> v_{in-}. The device geometries for this comparator all remain the same as for the comparator just described in this section, but the lack of the two extra inputs gives this comparator a better offset characteristic! In fact, it is the

increased resolution of this design which allows it to be used in the 1-bit to 3-bit pipeline. Since the two interstage SHAs between the sign bit comparator and the 3-bit flash A/D represent a potential gain of 6dB, the sign bit comparator **must** have more accuracy (a smaller offset) in order to resolve a smaller signal (unless digital correction were to be implemented in the pipeline by adding comparators to the flash converter in order to detect an error due to offset. -- fortunately, this was not necessary).

The 1-bit to 3-bit pipeline converter has been implemented in the 1 micron CMOS technology described in Section 2.3. Two stages of the four-stage VGA described in Chapter 4 have been merged into the A/D as the interstage gain amplifier for the pipeline. The device sizes used are as shown in Figure 17. See Section 6.1 for descriptions of the actual silicon die, and see Appendix A for relevant SPICE simulation decks.

SPICE simulations of the proposed comparator design (extracted from actual layout) showed the comparator performing at the speed and accuracy requirements for which it was designed. Figure 18 shows a transient output from a comparator simulation. The corresponding spice decks, *casc_sr.sp* and *casc_sr.spice*, can be found in Appendix A. The top panel of Figure 18 shows the input samples to the comparator. The horizontal line across the panel corresponds to the reference voltage. The second panel shows the digital output of the SR latch switching every time the input crosses the reference voltage. The bottom panel shows the two non-overlapping 128 MHz clocks used to control the comparator. Simulation of the entire A/D converter, although too cumbersome to include here, was consistent with the simulation of each individual comparator.

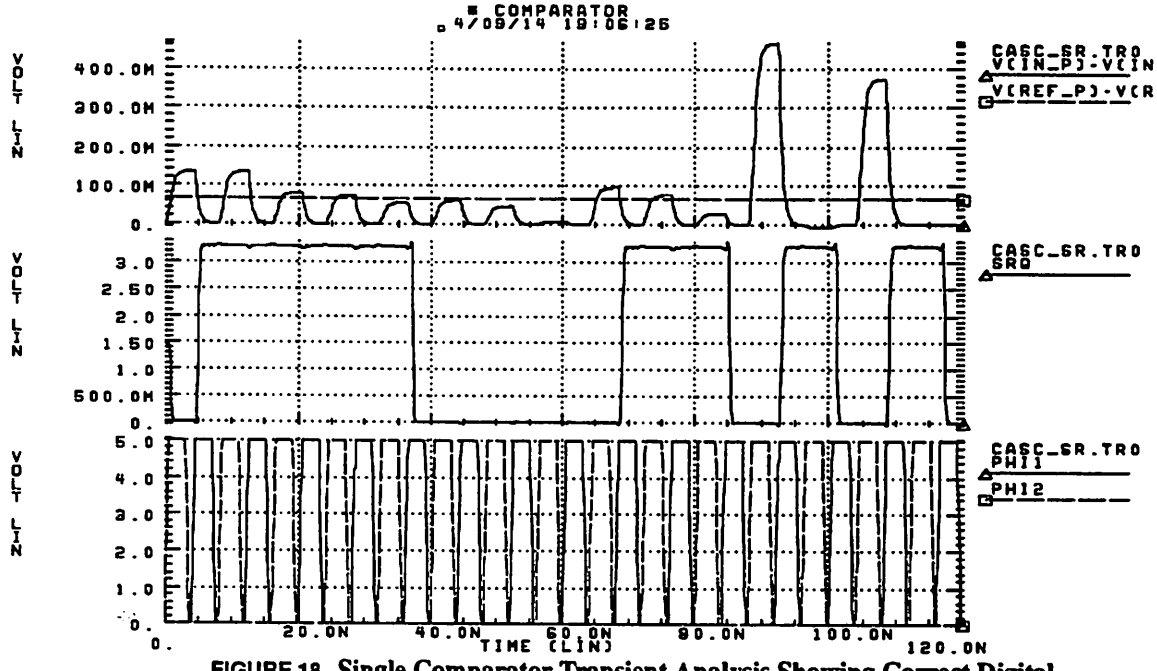

FIGURE 18. Single Comparator Transient Analysis Showing Correct Digital Output Based on Comparison of Input to Reference (top panel).

CHAPTER 6

Results

6.1 Design Prototype

Two chips have been fabricated in the standard digital CMOS process described in Section 2.3. The first chip, affectionately dubbed "minisporf", consists of the proposed 3-bit flash A/D structure. A diagram of the flash converter layout is shown in Figure 1, and a close-up diagram of one of the comparators is shown in Figure 2. It is important that the

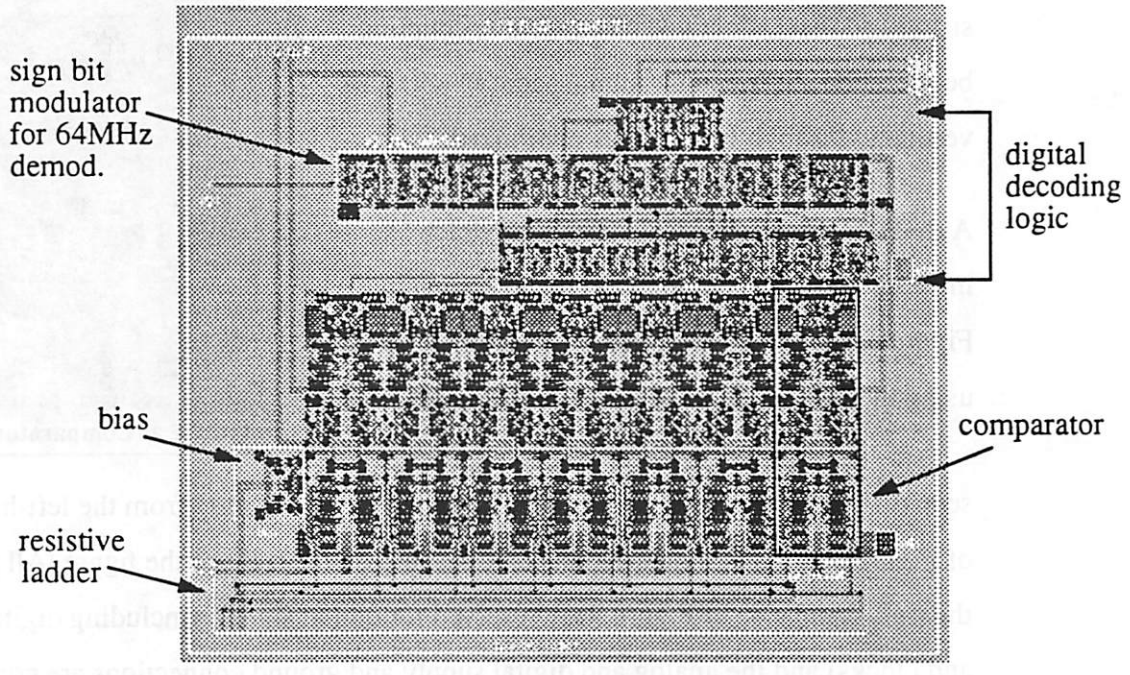

FIGURE 1. 3-Bit Analog to Digital Converter Layout

comparator layout be as symmetric as possible in order to prevent an increase in V_{os} due to geometry mismatch. Substrate and well contacts can be seen surrounding the circuits in the bottom half of the figure, forming rectangular guard rings to collect as much substrate noise current as possible. The signal flows from the bottom of Figure 2 to the top where the SR latch can bee seen (sandwiched between two bypass capacitors). It is also worth noting that the clock distribution is horizontal across the circuit (and not directly above any transistors) so that inductive and capacitive coupling between the clock and the signal path (which flows vertically through Figure 2) is minimized.

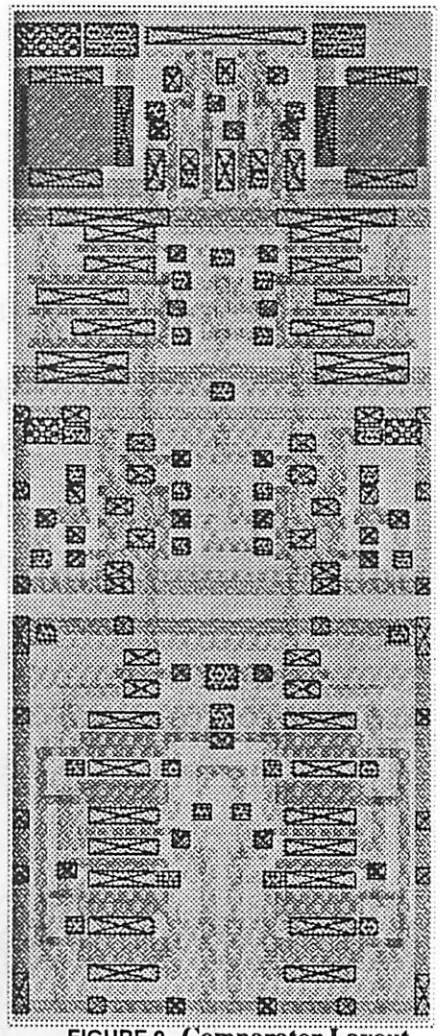

FIGURE 2. Comparator Layout

A diagram of the 3-bit A/D test chip (which includes a sample and hold circuit) is shown in Figure 3. Large on-chip bypass capacitors (created using the gate oxide of MOS transistors) can be seen distributed throughout the die. The signal enters the chip from the left-hand side of Figure 3 and the digital outputs exit the chip on the right of the figure. All pads on the right-hand side of Figure 3 are reserved for digital signals (including digital supply and clocks) and the analog and digital supply and ground connections are completely

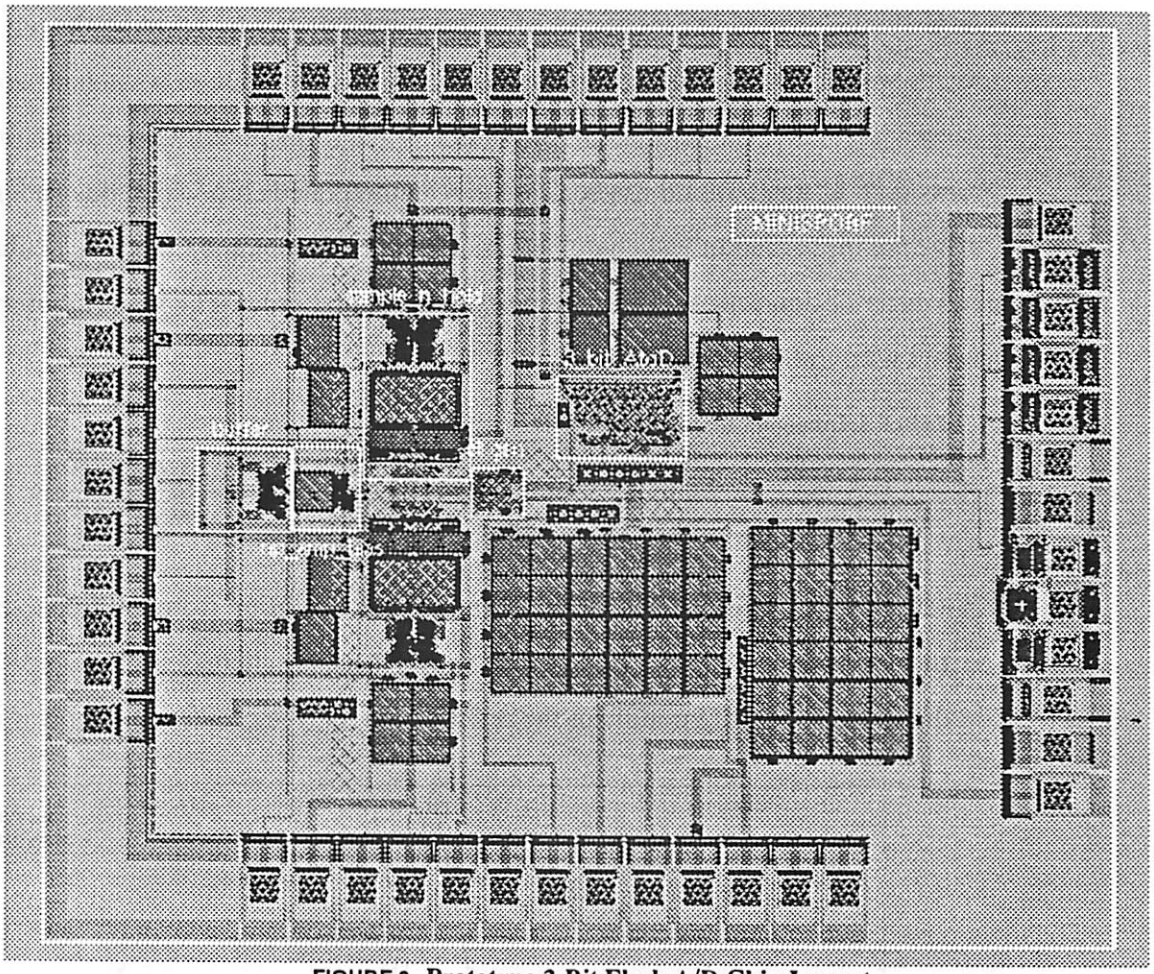

FIGURE 3. Prototype 3-Bit Flash A/D Chip Layout. (Preceeded by Sample and Hold Circuit)

separated on the chip (except for the fact that the grounds must be connected through the substrate of the die).

A high speed test board has been constructed to test the minisporf chip. Unfortunately, an error in the layout of the resistive ladder bias resulted in a small nonlinearity. Furthermore,

large digital switching noise (created by the output pads when driving the test equipment) was observed to couple into the analog signal path. Often, switching noise caused excursions larger than the amplitude of the signal itself. Large switching currents such as these would not be present in an integrated chip which does not need to drive the input load of test equipment. Fortunately the common mode rejection of the converter design rejects digital coupling noise to first order. The linearity of the converter was found to be:

TABLE 3. Linearity of 3 Bit Flash A/D

Uncorrected INL	0.967 LSB	Corrected INL	0.642 LSB
Uncorrected DNL	0.903 LSB	Corrected DNL	0.403 LSB

Where the "corrected" linearity numbers have the distortion from the resistive ladder error mathematically removed. Figure 4 shows the sharp increase in INL and DNL due to the error in the very middle of the resistive ladder (the center segment was acciden-

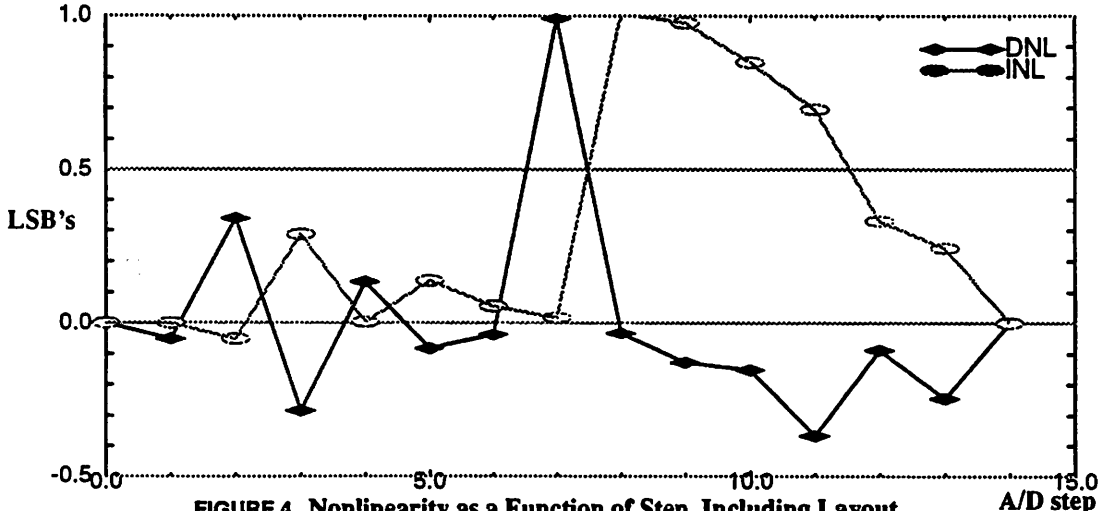

FIGURE 4. Nonlinearity as a Function of Step, Including Layout Error Between Steps Seven and Eight.

tally created two times larger than needed -- note that Figure 4 actually corresponds to the full 4 bit A/D which is described below. See Table 4 for the corrected values). Speed tests have shown that the A/D functions up to a 150 MHz clock rate (the goal of the design was 128 MHz). Beyond 150 MHz, the converter does not have sufficient time to evaluate a sig-

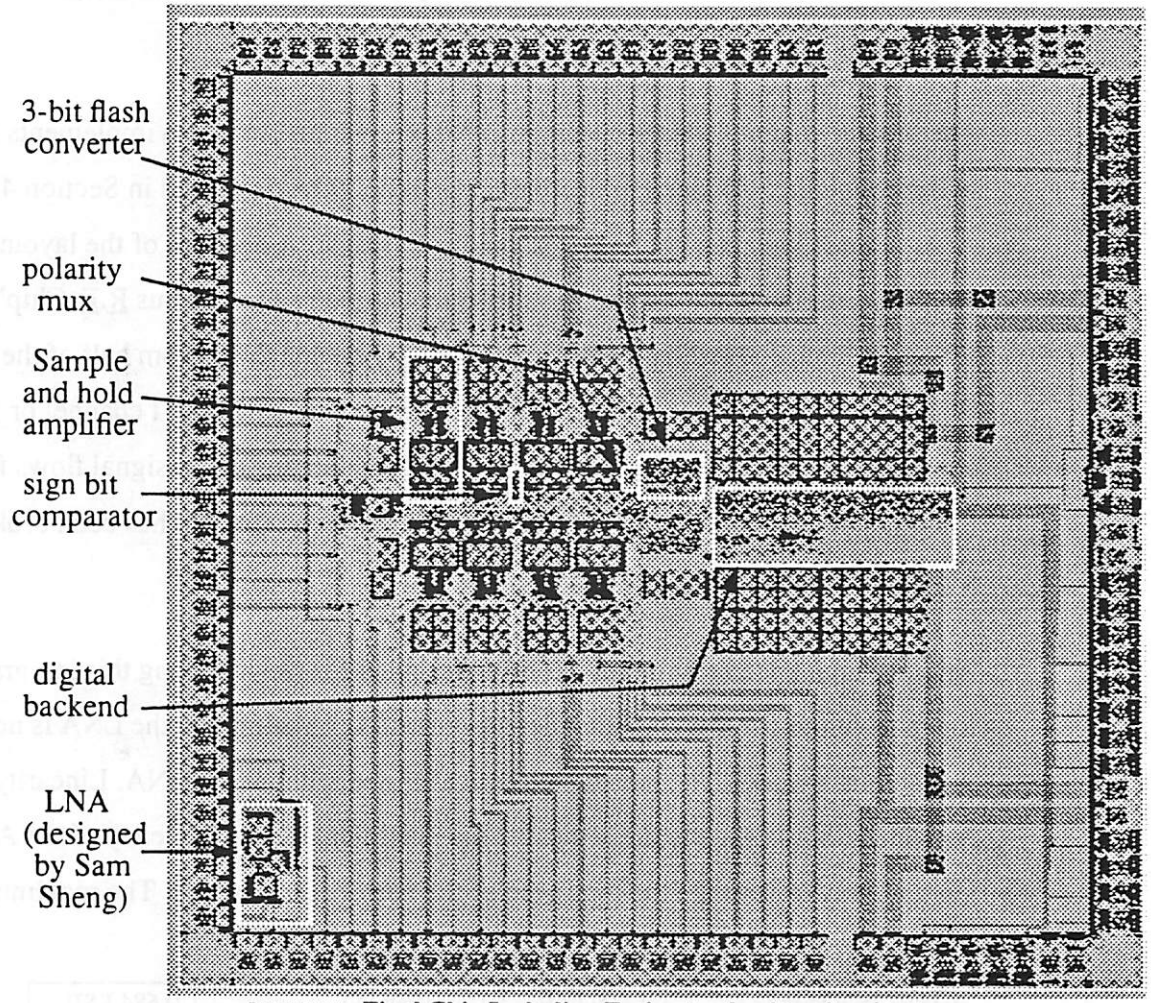

FIGURE 5. Final Chip Including Entire Analog Receive Chain

nal and convert it to a full digital level. Total power consumption from the analog circuitry is 2.2 mW. Digital power consumption for the converter could not be determined because the vast majority of digital power consumption on the test chip is created by the buffers driving the outputs off chip (These drivers should not be included in a power calculation because they would not exist in an integrated implementation).

The second chip includes the 3-bit flash A/D converter, but it also implements the 1-bit to 3-bit pipeline described in Section 5.2 and the VGA described in Section 4.2. The AGC control loop has not been designed at this time. A diagram of the layout of this chip, affectionately named "sporf" for "Sam and Poe's Outrageous R.F. Chip", is shown in Figure 5. Excluding the LNA, the top half and the bottom half of the core are symmetric copies of one another. Each half represents either the I channel or the Q channel of the receiver (designed for DQPSK modulation). The signal flows from left to right across Figure 5, and the pad ring to the right of the chip has been broken to separate analog and digital supplies.

Another high-speed test board has been designed for use in testing this integrated analog receiver chip. (The chip includes a front-end LNA, although the LNA is not part of this author's work.) The receiver chain was tested without the LNA. Linearity remained approximately the same, but the maximum clock frequency of the AGC was significantly lower than the top speed of the flash A/D converter. The maximum clock

TABLE 4. Linearity of Full 4-bit A/D Converter and AGC Chain

Uncorrected INL	1.009 LSB	Corrected INL	0.684 LSB
Uncorrected DNL	0.990 LSB	Corrected DNL	0.490 LSB

frequency at which an input sine wave was recovered at the output was determined to be 90 MHz. Beyond 90 MHz, loss of settling time due to the signal dependent kickback noise described in Section 4.4 causes the sample and hold amplifiers to fail. The first sample and hold stage was able to successfully subsample a maximum input frequency of 800 MHz (although this number is really limited by package parasitics). Figure 6 shows an FFT of an input sine wave being sampled and quantized by the prototype chip. The 100 kHz input

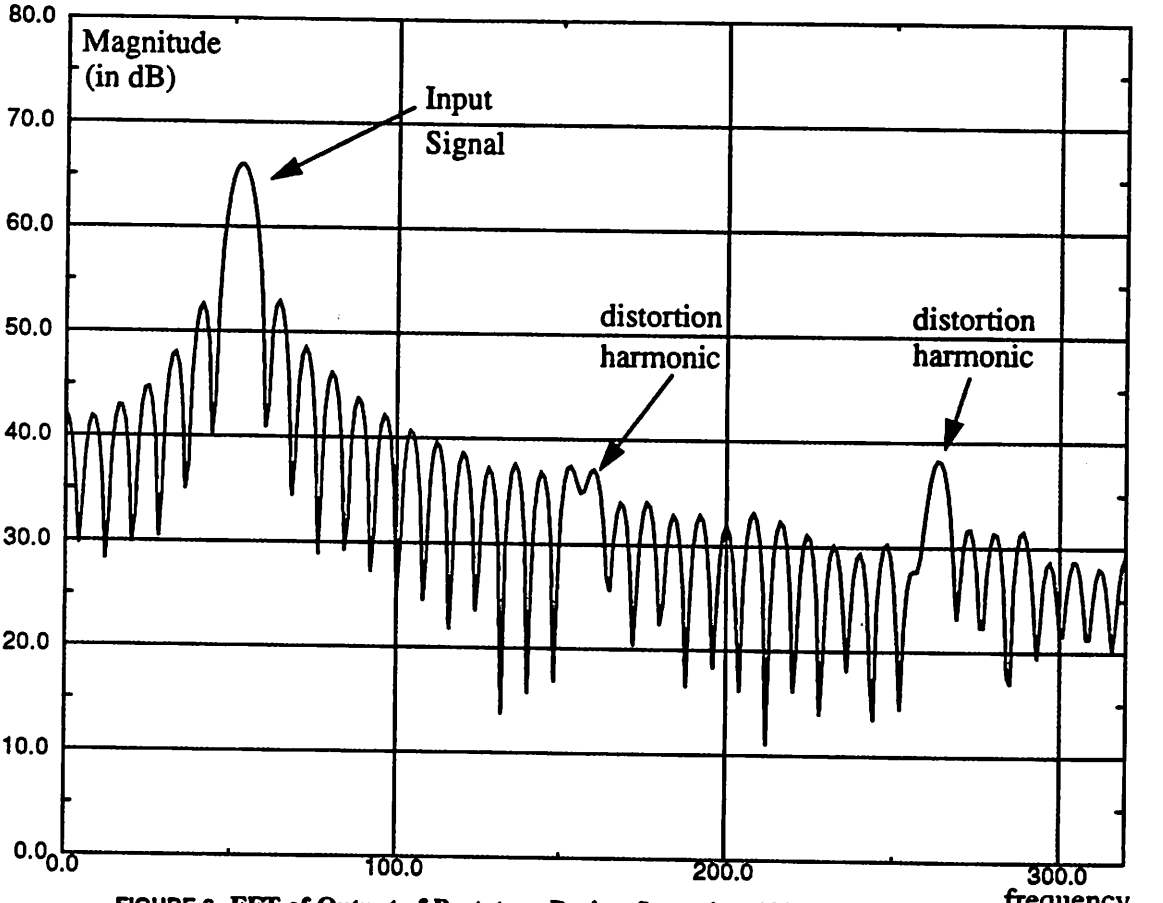

FIGURE 6. FFT of Output of Prototype Design. Input is a 100 kHz sine wave and is sampled at 16 MHz. SNDR is ~22dB.

sine wave is subsampled at 16 MHz (due to an output decimation by 4, the chip is actually clocking at 64 MHz) and then converted to 4 digital bits. By integrating the distortion harmonics (two of which are shown in Figure 6) the SNDR of the entire receiver chain has been calculated to be 22 dB (or approximately three and a half effective bits).

TABLE 5. Summary of Measured Results

Parameter	Value	Parameter	Value
Uncorrected INL	1.009 LSB	Corrected INL	0.684 LSB
Uncorrected DNL	0.990 LSB	Corrected DNL	0.490 LSB
Maximum Sampling Rate	90 MHz	Maximum A/D Conversion Rate	150 MHz
Total Measured Analog Power Consumption	70.9 mW	A/D Power Consumption (for two converters on chip)	4.4 mW
Per Stage Sample & Hold Power Consumption	8.3 mW	Digital Power Consumption (64 MHz clock rate)	101 mW
Peak SNDR	22 dB	Total Static Power Consumption (incl. test board)	102.2 mW

Table 5 shows a summary of the parameters measured from the prototype chip. Note that the digital power figure really reflects power consumed driving board parasitics, and is therefore not a valuable figure of merit. (The final integrated receiver will not need to drive off-chip loads.)

CHAPTER 7

Conclusion

A variable-gain amplifier and an analog-to-digital converter have been designed and implemented in a standard digital one-micron CMOS process for the U.C. Berkeley Infopad. In an effort to develop a next-generation wireless radio link capable of supporting video data rates to multiple users in an indoor picocellular environment, a direct-sequence spread-spectrum scheme has been adopted. By capitalizing on the signal-to-noise gain provided by the spreading and despreading process, the accuracy requirements of the A/D have been relaxed to 4 bits. In order to meet the dynamic range requirements of the system without increasing both the resolution of the A/D and the wordlength in the digital processing unit, the variable-gain amplifier has been designed for use in an AGC loop. Since the VGA follows a subsampling mixer in the receiver, it has been designed using a cascade of discrete-time sample-and-hold amplifier stages. Furthermore, by merging part of the A/D function with the VGA, a 1-bit to 3-bit pipeline architecture was chosen for the A/D converter -- resuting in significant power-savings. Low-power and high-speed were the two main design goals. Measurements of the design prototype showed the design functioning up to a 90 MHz maximum clock rate, a 800 MHz maximum input bandwidth (subsampled down to a 64 MHz clock rate), and 71 mW of static power consumption.

References

[1] S. Sheng, R. Allmon, L. Lynn, I. O'Donnell, K. Stone, R. Brodersen, "A Monolithic CMOS Radio System for Wideband CDMA Communications", Proceedings to Wireless '94 Conference, Calgary, Canada, June 1994.

[2] S. Sheng, *Wideband Digital Portable Communications: A System Design*, M.S. Thesis, Memorandum No. UCB/ERL M91/108, December 3, 1991.

[3] S. Sheng, *Wideband Digital Portable Communications*, Ph.D. Thesis, U.C. Berkeley, to be published in 1995.

[4] K. Stone, *Low Power Spread Spectrum Demodulator for Wideband Wireless Communications*, M.S. Thesis, U.C. Berkeley, 1995.

[5] C. Teuscher, *Software Simulation for the Infopad Downlink*, M.S. Thesis, U.C. Berkeley, December 1994.

[6] A. Behzad, *The Implementation of A High Speed Experimental Transceiver Module with an Emphasis on CDMA Applications*, M.S. Thesis, U.C. Berkeley, 1995.

[7] S. Sheng, A. P. Chandrakasan, R. W. Brodersen, "A Portable Multimedia Terminal", IEEE Communications Magazine, pp. 64-75, December 1992.

[8] S. Lewis, "A 10-b 20-Msample/s Analog-to-Digital Converter," *IEEE Journal of Solid-State Circuits*, Vol.27, No.3, pp.351-357, Mar. 1992.

[9] T. Cho, *Low-Power, Low-Voltage Analog-to-Digital Conversion Techniques Using Pipelined Architectures*, Ph.D. Thesis, U.C. Berkeley, April. 1995.

[10] C. Conroy, *A High-Speed Parallel Pipeline A/D Converter Technique in CMOS*, Ph.D. Thesis, U.C. Berkeley, Feb. 1994.

[11] J. Proakis, Digital Communications, USA: McGraw-Hill, 1989.

[12] P.R. Gray, R.G. Meyer, *Analysis and Design of Analog Integrated Circuits*, 3rd ed., USA: John Wiley & Sons Inc., 1993.

[13] G. Yin, F. Op't Eynde, W. Sansen, "A High-Speed CMOS Comparator with 8-b resolution," *IEEE Journal of Solid-State Circuits*, Vol. 27, No. 2, pp. 208-211, Feb. 1992.

Appendix A

<This Page Intentionally Left Blank>

samnet.sp


```
* sample and hold circuits cascaded for agc chain

.options nomod post=2
.param stepsize=16.65m bias=.3mA refnW=65u refpW=96u

*** clocks
vphi1 phi1 OTAGND pulse (0 5.0 8ns .6ns .6ns 6.5ns 16ns)
vphisamp1 phisamp1 OTAGND pulse (0 5.0 8.7ns .6ns .6ns 6.5ns 16ns)
vphieval1 phieval1 OTAGND pulse (0 5.0 9.2ns .6ns .6ns 6.5ns 16ns)
vphi2 phi2 OTAGND pulse (0 5.0 0ns .6ns .6ns 6.5ns 16ns)
vphisamp2 phisamp2 OTAGND pulse (0 5.0 0.7ns .6ns .6ns 6.5ns 16ns)
vphieval2 phieval2 OTAGND pulse (0 5.0 1.2ns .6ns .6ns 6.5ns 16ns)
*vphi1 phi1 OTAGND pulse (0 5.0 4ns .6ns .6ns 2.5ns 8ns)
*vphisamp1 phisamp1 OTAGND pulse (0 5.0 4ns .6ns .6ns 3.2ns 8ns)
*vphieval1 phieval1 OTAGND pulse (0 5.0 4ns .6ns .6ns 3.7ns 8ns)
*vphi2 phi2 OTAGND pulse (0 5.0 0ns .6ns .6ns 2.5ns 8ns)
*vphisamp2 phisamp2 OTAGND pulse (0 5.0 0ns .6ns .6ns 3.2ns 8ns)
*vphieval2 phieval2 OTAGND pulse (0 5.0 0ns .6ns .6ns 3.7ns 8ns)


*** supply
*vdd vdd_ideal 0 3.3
vdd vdd_ideal 0 5
lvdd vdd_ideal OTAVDD 5nH
lgnd OTAcGND 0 3nH
rotadamp OTAcGND OTAGND 20
vshortgnd OTAGND agnd 0
* digital's separtate supply
vdigitalvdd dvdd_ideal 0 5v
ldgnd dgnd 0 3nH
ldvdd dvdd_ideal dvdd 5nH
cdbypass dvdd cgnd 100pF
rdamp cgnd dgnd 100


*** input
.include /users/lynn/agc/input/shinputslow.sp
vAGC_ctrl AGC_ctrl OTAGND dc 5

*** bias
*Vicm vicm 0 dc 1.7
*Vcmo vcmoref 0 dc 1.7
Vicm vicm 0 dc 2.5
Vcmo vcmoref 0 dc 2.5

*Iref tailrefp1 tailrefn dc 'bias'
*mtailrefp tailrefp tailrefp OTAVDD OTAVDD pmos W=96u L=1.0u
*vhspcasc tailrefp tailrefp1 0.9V
*mtailrefn tailrefn tailrefn OTAGND OTAGND nmos30 W=65u L=1.5u
*vpshrt1 tailrefp  OTAPSRCBIAS dc 0
*vpshrt2 tailrefp1 OTAPSRCCASCBIAS dc 0
*vpshrt3 tailrefn OTATAILGATE dc 0
Iref otavdd otatailgate dc 0.32mA
mtailrefn otatailgate otatailgate otagnd otagnd nmos30 w='refnW' l=1.5u
mmirror1 otapsrcshift otatailgate otagnd otagnd nmos30 w='refnW' l=1.5u
*vdioshift otapsrcbias otapsrcshift dc 1.5
mdiodeshift otapsrcshift otapsrcshift otapsrcbias otavdd pmos w='refpW/2' l=1.0u
mpsrcref otapsrcbias otapsrcbias otavdd otavdd pmos w='refpW' l=1.0u
mmirror2 otapsrccascbias otatailgate otagnd otagnd nmos30 w='refnW*2' l=1.5u
mcascbias otapsrccascbias otapsrccascbias pdiosrc otavdd pmos w=400u l=1u
mcasctriodebias pdiosrc otapsrccascbias otavdd otavdd pmos w=50u l=1u
ctailbypass otatailgate otagnd 20pF
```

```
********** sample and hold circuit *************
.subckt SandH OTAGND OTAVDD dgnd dvdd vi+ vi- otavo+ otavo-
+            otatailgate otapsrcbias otapsrccascbias vcmoref vicm
+            phi1in phis1in phie1in phie2in AGC_ctrl

* Transistors
.include samnet.spice

** hack to set capacitors for now.
gshort1 OTAVO+ vcmoref VCR PWL(1) short 0 1v,10mega 4v,100u
gshort2 OTAVO- vcmoref VCR PWL(1) short 0 1v,10mega 4v,100u
gshort3 OTAVIN+ vicm VCR PWL(1) short 0 1v,10mega 4v,100u
gshort4 OTAVIN- vicm VCR PWL(1) short 0 1v,10mega 4v,100u
vshort short 0 pulse(5 0 10n)

.ends

**********  AGC chain  *************
xsandh1 OTAGND OTAVDD dgnd dvdd vi+ vi- vo1+ vo1- otatailgate otapsrcbias
+        otapsrccascbias vcmoref vicm phi1 phisamp1 phieval1
+                phieval2 AGC_ctrl
+        SandH

xsandh2 OTAGND OTAVDD dgnd dvdd vo1+ vo1- vo2+ vo2- otatailgate otapsrcbias
+        otapsrccascbias vcmoref vicm phi2 phisamp2 phieval2
+                phieval1 AGC_ctrl
+        SandH

cload3+ vo2+ OTAGND .5pF
cload3- vo2- OTAGND .5pF

.include /users/ssheng/rf/mosis/08hp/10model.139

*analysis
.op 80ns
.tran .25n 240n
*.tran .2n 48n
.probe
+     v(OTAVDD) v(OTAGND)
+     v(phi1) v(phieval1) v(phisamp1)
+     v(phi2) v(phieval2) v(phisamp2)
+     v(vi+) v(vi-)
+     v(vo1+) v(vo1-)
+     v(vo2+) v(vo2-)
+         v(vi+,vi-)
+         v(vo1-,vo1+)
+         v(vo2+,vo2-)
.meas tran pvdd avg p(vdd) from 12ns to 48ns
.meas tran vi_samp1 find v(vi+,vi-) when v(phisamp1)=2.5v td=15n fall=1
.meas tran vi_samp2 find v(vi+,vi-) when v(phisamp1)=2.5v td=15n fall=2
.meas tran vi_samp3 find v(vi+,vi-) when v(phisamp1)=2.5v td=15n fall=3
.meas tran vi_samp4 find v(vi+,vi-) when v(phisamp1)=2.5v td=15n fall=4
.meas tran vi_samp5 find v(vi+,vi-) when v(phisamp1)=2.5v td=15n fall=5
.meas tran vi_samp6 find v(vi+,vi-) when v(phisamp1)=2.5v td=15n fall=6
.meas tran vi_samp7 find v(vi+,vi-) when v(phisamp1)=2.5v td=15n fall=7
.meas tran vi_samp8 find v(vi+,vi-) when v(phisamp1)=2.5v td=15n fall=8
.meas tran vi_samp9 find v(vi+,vi-) when v(phisamp1)=2.5v td=15n fall=9
.meas tran vi_samp10 find v(vi+,vi-) when v(phisamp1)=2.5v td=15n fall=10
.meas tran vi_samp11 find v(vi+,vi-) when v(phisamp1)=2.5v td=15n fall=11
.meas tran vi_samp12 find v(vi+,vi-) when v(phisamp1)=2.5v td=15n fall=12
.meas tran vi_samp13 find v(vi+,vi-) when v(phisamp1)=2.5v td=15n fall=13
.meas tran vo1_samp1 find v(vo1-,vo1+) when v(phisamp2)=2.5v td=19n fall=1
.meas tran vo1_samp2 find v(vo1-,vo1+) when v(phisamp2)=2.5v td=19n fall=2
.meas tran vo1_samp3 find v(vo1-,vo1+) when v(phisamp2)=2.5v td=19n fall=3
```

```
.meas tran vo1_samp4 find v(vo1-,vo1+) when v(phisamp2)=2.5v td=19n fall=4
.meas tran vo1_samp5 find v(vo1-,vo1+) when v(phisamp2)=2.5v td=19n fall=5
.meas tran vo1_samp6 find v(vo1-,vo1+) when v(phisamp2)=2.5v td=19n fall=6
.meas tran vo1_samp7 find v(vo1-,vo1+) when v(phisamp2)=2.5v td=19n fall=7
.meas tran vo1_samp8 find v(vo1-,vo1+) when v(phisamp2)=2.5v td=19n fall=8
.meas tran vo1_samp9 find v(vo1-,vo1+) when v(phisamp2)=2.5v td=19n fall=9
.meas tran vo1_samp10 find v(vo1-,vo1+) when v(phisamp2)=2.5v td=19n fall=10
.meas tran vo1_samp11 find v(vo1-,vo1+) when v(phisamp2)=2.5v td=19n fall=11
.meas tran vo1_samp12 find v(vo1-,vo1+) when v(phisamp2)=2.5v td=19n fall=12
.meas tran vo1_samp13 find v(vo1-,vo1+) when v(phisamp2)=2.5v td=19n fall=13
.meas tran vo2_samp1 find v(vo2+,vo2-) when v(phisamp1)=2.5v td=23n fall=1
.meas tran vo2_samp2 find v(vo2+,vo2-) when v(phisamp1)=2.5v td=23n fall=2
.meas tran vo2_samp3 find v(vo2+,vo2-) when v(phisamp1)=2.5v td=23n fall=3
.meas tran vo2_samp4 find v(vo2+,vo2-) when v(phisamp1)=2.5v td=23n fall=4
.meas tran vo2_samp5 find v(vo2+,vo2-) when v(phisamp1)=2.5v td=23n fall=5
.meas tran vo2_samp6 find v(vo2+,vo2-) when v(phisamp1)=2.5v td=23n fall=6
.meas tran vo2_samp7 find v(vo2+,vo2-) when v(phisamp1)=2.5v td=23n fall=7
.meas tran vo2_samp8 find v(vo2+,vo2-) when v(phisamp1)=2.5v td=23n fall=8
.meas tran vo2_samp9 find v(vo2+,vo2-) when v(phisamp1)=2.5v td=23n fall=9
.meas tran vo2_samp10 find v(vo2+,vo2-) when v(phisamp1)=2.5v td=23n fall=10
.meas tran vo2_samp11 find v(vo2+,vo2-) when v(phisamp1)=2.5v td=23n fall=11
.meas tran vo2_samp12 find v(vo2+,vo2-) when v(phisamp1)=2.5v td=23n fall=12
.meas tran vo2_samp13 find v(vo2+,vo2-) when v(phisamp1)=2.5v td=23n fall=13
*.alter 2.5mA_tailcurrent
*.param bias=.25mA
*.alter 3mA_tailcurrent
*.param bias=.3mA
*.alter 3.5mA_tailcurrent
*.param bias=.35mA
*.alter 4mA_tailcurrent
*.param bias=.4mA
*.alter 4.5mA_tailcurrent
*.param bias=.45mA
*.alter 5mA_tailcurrent
*.param bias=.5mA
.end
```

```
+ AS=56.4P PS=11.1U
M63 otadrivsrc otatailgate otabotgnd GND! NMOS30 W=32.5U L=1.5U AD=56.4P
+ PD=11.1U AS=55.2P PS=9.9U
M64 otabotgnd otatailgate otadrivsrc GND! NMOS30 W=32.5U L=1.5U AD=55.2P PD=9.9U
+ AS=56.4P PS=11.1U
M65 otadrivsrc otatailgate otabotgnd GND! NMOS30 W=32.5U L=1.5U AD=56.4P
+ PD=11.1U AS=55.2P PS=9.9U

C1 otavin otacascdrn 2.00F
C2 otavin otacascbias 1.00F
C3 otadrivsrc otabotgnd 1.00F
C4 otatailgate otadrivsrc 1.00F
C5 OTAPSRCBIAS otatopvdd 9.00F
C6 otatailgate otabotgnd 1.00F
C7 otapsrccascbias otatopvdd 11.00F
C8 otapsrccascbias OTAPSRCBIAS 3.00F
C9 otadrivsrc otacascbias 3.00F
C10 otaCMFBin otadrivsrc 1.00F
C11 otaCMFBin 0 10.0F
C12 otatailgate 0 28.0F
C13 otacascbias 0 45.0F
C14 otacascdrn 0 47.0F
C15 otatopvdd 0 2312.0F
C16 otacascdiosrc 0 8.0F
C17 otapsrccascbias 0 51.0F
C18 otavin 0 27.0F
C19 OTAPSRCBIAS 0 58.0F
C20 otadrivsrc 0 64.0F
C21 otabotgnd 0 44.0F
*** Node Listing for subckt:  telehlf
**  0
**  1   8_12_594#
**  2   8_12_570#
**  3   8_12_546#
**  4   8_270_550#
**  5   8_12_522#
**  6   8_270_526#
**  7   8_12_498#
**  8   8_270_502#
**  9   8_12_474#
**  10  8_270_478#
**  11  8_12_450#
**  12  8_270_454#
**  13  8_12_426#
**  14  8_270_430#
**  15  8_270_406#
**  16  8_50_350#
**  17  8_270_382#
**  18  8_50_326#
**  19  8_50_302#
**  20  8_50_278#
**  21  8_50_254#
**  22  8_50_230#
**  23  8_50_206#
**  24  8_50_182#
.ENDS


*****  Subcircuit from file ./hlfswitches.ext
.SUBCKT hlfswitches phieval1 phi1 phisamp1 phieval2 vi+_ctrl AGC+_ctrl 1 2 3 4
+ 5 OTAVO- 6 7 8 vi+_1 vi+ 9 OTAVIN+ 10 11 12 vicm 13 14 vi+_2 15 refn OTACMFBIN

M1 vicm phieval1 OTAVO- GND! NMOS W=15.0U L=1.0U AD=27.3P PD=8.5U AS=37.5P
+ PS=17.5U
M2 refn phisamp1 OTACMFBIN GND! NMOS W=8.0U L=1.0U AD=20.0P PD=13.0U AS=20.0P
+ PS=13.0U
M3 vi+_1 phieval2 vicm GND! NMOS W=15.0U L=1.0U AD=37.5P PD=17.5U AS=27.3P
+ PS=8.5U
M4 OTAVIN+ phi1 vicm GND! NMOS W=14.0U L=1.0U AD=21.0P PD=3.0U AS=25.5P
+ PS=8.0U
M5 10 phisamp1 OTAVIN+ GND! NMOS W=14.0U L=1.0U AD=35.0P PD=17.6U AS=21.0P
+ PS=3.0U
M6 vi+_ctrl AGC+_ctrl 10 GND! NMOS W=22.0U L=1.0U AD=33.0P PD=3.0U AS=55.0P
+ PS=27.7U
M7 10 AGC+_ctrl vi+_ctrl GND! NMOS W=22.0U L=1.0U AD=55.0P PD=27.7U AS=33.0P
+ PS=3.0U
M8 vi+_2 phieval2 OTAVO- GND! NMOS W=45.0U L=1.0U AD=67.5P PD=3.0U AS=112.5P
+ PS=52.5U
M9 vi+ phieval1 vi+_2 GND! NMOS W=45.0U L=1.0U AD=67.5P PD=3.0U AS=67.5P
+ PS=3.0U
M10 vi+_1 phieval1 vi+ GND! NMOS W=45.0U L=1.0U AD=112.5P PD=52.5U AS=67.5P
+ PS=3.0U

C1 6 vicm 1.00F
C2 4 vicm 41.00F
C3 4 vi+_1 4.00F
C4 2 vicm 1.00F
C5 AGC+_ctrl 15 1.00F
C6 11 4 30.00F
C7 OTAVIN+ 15 1.00F
C8 4 refn 35.00F
C9 4 9 30.00F
C10 12 vicm 1.00F
C11 14 4 17.00F
C12 4 vi+_1 4.00F
C13 vicm 15 1.00F
C14 8 vicm 1.00F
C15 vi+_1 15 1.00F
C16 AGC+_ctrl vicm 1.00F
C17 4 vi+ 1.00F
C18 1 4 19.00F
C19 4 OTACMFBIN 4.00F
C20 4 3 30.00F
C21 vi+_ctrl 4 7.00F
C22 vi+_1 15 1.00F
C23 4 OTAVO- 6.00F
C24 4 5 30.00F
C25 vi+_2 4 2.00F
C26 vi+ 15 1.00F
C27 11 vicm 1.00F
C28 6 4 30.00F
C29 refn vicm 1.00F
C30 13 vicm 1.00F
C31 9 vicm 1.00F
C32 14 vicm 1.00F
C33 vi+_ctrl 15 1.00F
C34 2 4 17.00F
C35 7 4 30.00F
C36 OTAVO- 15 1.00F
C37 1 vicm 1.00F
C38 vi+_2 15 1.00F
C39 12 4 30.00F
C40 3 vicm 1.00F
C41 4 8 19.00F
C42 AGC+_ctrl 4 34.00F
C43 4 OTAVIN+ 7.00F
C44 9 0 14.0F
C45 AGC+_ctrl 0 21.0F
```

```
*****  Subcircuit from file ./telehlf.ext
.SUBCKT telehlf otacascbias otadrivsrc otacascdrn otaCMFBin otapsrccascbias
+ OTAPSRCBIAS otatailgate otavin otabotgnd otatopvdd

M1 1 OTAPSRCBIAS otatopvdd otatopvdd PMOS W=50.0U L=1.0U AD=25.0P PD=1.0U
+ AS=87.5P PS=18.3U
M2 otacascdrn otapsrccascbias 1 otatopvdd PMOS W=50.0U L=1.0U AD=75.0P PD=3.0U
+ AS=25.0P PS=1.0U
M3 2 otapsrccascbias otacascdrn otatopvdd PMOS W=50.0U L=1.0U AD=25.0P PD=1.0U
+ AS=75.0P PS=3.0U
M4 otatopvdd OTAPSRCBIAS 2 otatopvdd PMOS W=50.0U L=1.0U AD=87.5P PD=18.3U
+ AS=25.0P PS=1.0U
M5 3 OTAPSRCBIAS otatopvdd otatopvdd PMOS W=50.0U L=1.0U AD=25.0P PD=1.0U
+ AS=87.5P PS=18.3U
M6 4 OTAPSRCBIAS otatopvdd otatopvdd PMOS W=10.0U L=1.0U AD=5.0P PD=1.0U
+ AS=17.5P PS=3.7U
M7 otacascbias otapsrccascbias 4 otatopvdd PMOS W=10.0U L=1.0U AD=15.0P
+ PD=3.0U AS=5.0P PS=1.0U
M8 otacascdrn otapsrccascbias 3 otatopvdd PMOS W=50.0U L=1.0U AD=75.0P PD=3.0U
+ AS=25.0P PS=1.0U
M9 5 otapsrccascbias otacascdrn otatopvdd PMOS W=50.0U L=1.0U AD=25.0P PD=1.0U
+ AS=75.0P PS=3.0U
M10 6 otapsrccascbias otacascbias otatopvdd PMOS W=10.0U L=1.0U AD=5.0P
+ PD=1.0U AS=15.0P PS=3.0U
M11 otatopvdd OTAPSRCBIAS 6 otatopvdd PMOS W=10.0U L=1.0U AD=17.5P PD=3.7U
+ AS=5.0P PS=1.0U
M12 otatopvdd OTAPSRCBIAS 5 otatopvdd PMOS W=50.0U L=1.0U AD=87.5P PD=18.3U
+ AS=25.0P PS=1.0U
M13 7 OTAPSRCBIAS otatopvdd otatopvdd PMOS W=50.0U L=1.0U AD=25.0P PD=1.0U
+ AS=87.5P PS=18.3U
M14 8 OTAPSRCBIAS otatopvdd otatopvdd PMOS W=10.0U L=1.0U AD=5.0P PD=1.0U
+ AS=17.5P PS=3.7U
M15 otacascbias otapsrccascbias 8 otatopvdd PMOS W=10.0U L=1.0U AD=15.0P
+ PD=3.0U AS=5.0P PS=1.0U
M16 otacascdrn otapsrccascbias 7 otatopvdd PMOS W=50.0U L=1.0U AD=75.0P
+ PD=3.0U AS=25.0P PS=1.0U
M17 9 otapsrccascbias otacascdrn otatopvdd PMOS W=50.0U L=1.0U AD=25.0P
+ PD=1.0U AS=75.0P PS=3.0U
M18 10 otapsrccascbias otacascbias otatopvdd PMOS W=10.0U L=1.0U AD=5.0P
+ PD=1.0U AS=15.0P PS=3.0U
M19 otatopvdd OTAPSRCBIAS 10 otatopvdd PMOS W=10.0U L=1.0U AD=17.5P PD=3.7U
+ AS=5.0P PS=1.0U
M20 otatopvdd OTAPSRCBIAS 9 otatopvdd PMOS W=50.0U L=1.0U AD=87.5P PD=18.3U
+ AS=25.0P PS=1.0U
M21 11 OTAPSRCBIAS otatopvdd otatopvdd PMOS W=50.0U L=1.0U AD=25.0P PD=1.0U
+ AS=87.5P PS=18.3U
M22 12 OTAPSRCBIAS otatopvdd otatopvdd PMOS W=10.0U L=1.0U AD=5.0P PD=1.0U
+ AS=17.5P PS=3.7U
M23 otacascbias otapsrccascbias 12 otatopvdd PMOS W=10.0U L=1.0U AD=15.0P
+ PD=3.0U AS=5.0P PS=1.0U
M24 otacascdrn otapsrccascbias 11 otatopvdd PMOS W=50.0U L=1.0U AD=75.0P
+ PD=3.0U AS=25.0P PS=1.0U
M25 13 otapsrccascbias otacascdrn otatopvdd PMOS W=50.0U L=1.0U AD=25.0P
+ PD=1.0U AS=75.0P PS=3.0U
M26 14 otapsrccascbias otacascbias otatopvdd PMOS W=10.0U L=1.0U AD=5.0P
+ PD=1.0U AS=15.0P PS=3.0U
M27 otatopvdd OTAPSRCBIAS 14 otatopvdd PMOS W=10.0U L=1.0U AD=17.5P PD=3.7U
+ AS=5.0P PS=1.0U
M28 otatopvdd OTAPSRCBIAS 13 otatopvdd PMOS W=50.0U L=1.0U AD=87.5P PD=18.3U
+ AS=25.0P PS=1.0U
M29 15 OTAPSRCBIAS otatopvdd otatopvdd PMOS W=10.0U L=1.0U AD=5.0P PD=1.0U
+ AS=17.5P PS=3.7U
M30 otacascbias otapsrccascbias 15 otatopvdd PMOS W=10.0U L=1.0U AD=15.0P
+ PD=3.0U AS=5.0P PS=1.0U
M31 16 otavin otadrivsrc GND! NMOS W=37.5U L=1.0U AD=18.8P PD=1.0U AS=65.1P
+ PS=12.8U
M32 17 otapsrccascbias otacascbias otatopvdd PMOS W=10.0U L=1.0U AD=5.0P
+ PD=1.0U AS=15.0P PS=3.0U
M33 otatopvdd OTAPSRCBIAS 17 otatopvdd PMOS W=10.0U L=1.0U AD=17.5P PD=3.7U
+ AS=5.0P PS=1.0U
M34 otacascdrn otacascbias 16 GND! NMOS W=37.5U L=1.0U AD=56.2P PD=3.0U
+ AS=18.8P PS=1.0U
M35 18 otacascbias otacascdrn GND! NMOS W=37.5U L=1.0U AD=18.8P PD=1.0U
+ AS=56.2P PS=3.0U
M36 otadrivsrc otavin 18 GND! NMOS W=37.5U L=1.0U AD=65.1P PD=12.8U AS=18.8P
+ PS=1.0U
M37 19 otavin otadrivsrc GND! NMOS W=37.5U L=1.0U AD=18.8P PD=1.0U AS=65.1P
+ PS=12.8U
M38 otacascdrn otacascbias 19 GND! NMOS W=37.5U L=1.0U AD=56.2P PD=3.0U
+ AS=18.8P PS=1.0U
M39 20 otacascbias otacascdrn GND! NMOS W=37.5U L=1.0U AD=18.8P PD=1.0U
+ AS=56.2P PS=3.0U
M40 otadrivsrc otavin 20 GND! NMOS W=37.5U L=1.0U AD=65.1P PD=12.8U AS=18.8P
+ PS=1.0U
M41 21 otavin otadrivsrc GND! NMOS W=37.5U L=1.0U AD=18.8P PD=1.0U AS=65.1P
+ PS=12.8U
M42 otacascdrn otacascbias 21 GND! NMOS W=37.5U L=1.0U AD=56.2P PD=3.0U
+ AS=18.8P PS=1.0U
M43 22 otacascbias otacascdrn GND! NMOS W=37.5U L=1.0U AD=18.8P PD=1.0U
+ AS=56.2P PS=3.0U
M44 otadrivsrc otavin 22 GND! NMOS W=37.5U L=1.0U AD=65.1P PD=12.8U AS=18.8P
+ PS=1.0U
M45 23 otavin otadrivsrc GND! NMOS W=37.5U L=1.0U AD=18.8P PD=1.0U AS=65.1P
+ PS=12.8U
M46 otacascdrn otacascbias 23 GND! NMOS W=37.5U L=1.0U AD=56.2P PD=3.0U
+ AS=18.8P PS=1.0U
M47 24 otacascbias otacascdrn GND! NMOS W=37.5U L=1.0U AD=18.8P PD=1.0U
+ AS=56.2P PS=3.0U
M48 otacascbias otacascbias otacascdiosrc GND! NMOS W=10.0U L=1.0U AD=15.0P
+ PD=3.0U AS=16.4P PS=4.7U
M49 otacascdiosrc otacascbias otacascbias GND! NMOS W=10.0U L=1.0U AD=16.4P
+ PD=4.7U AS=15.0P PS=3.0U
M50 otacascbias otacascbias otacascdiosrc GND! NMOS W=10.0U L=1.0U AD=15.0P
+ PD=3.0U AS=16.4P PS=4.7U
M51 otacascdiosrc otacascbias otacascbias GND! NMOS W=10.0U L=1.0U AD=16.4P
+ PD=4.7U AS=15.0P PS=3.0U
M52 otacascbias otacascbias otacascdiosrc GND! NMOS W=10.0U L=1.0U AD=15.0P
+ PD=3.0U AS=16.4P PS=4.7U
M53 otacascdiosrc otacascbias otacascbias GND! NMOS W=10.0U L=1.0U AD=16.4P
+ PD=4.7U AS=15.0P PS=3.0U
M54 otadrivsrc otavin 24 GND! NMOS W=37.5U L=1.0U AD=65.1P PD=12.8U AS=18.8P
+ PS=1.0U
M55 otadrivsrc otacascbias otacascdiosrc GND! NMOS W=10.0U L=1.0U AD=17.4P
+ PD=3.4U AS=16.4P PS=4.7U
M56 otadrivsrc otaCMFBin otabotgnd GND! NMOS30 W=32.5U L=1.5U AD=56.4P PD=11.1U
+ AS=55.2P PS=9.9U
M57 otabotgnd otaCMFBin otadrivsrc GND! NMOS30 W=32.5U L=1.5U AD=55.2P PD=9.9U
+ AS=56.4P PS=11.1U
M58 otadrivsrc otaCMFBin otabotgnd GND! NMOS30 W=32.5U L=1.5U AD=56.4P PD=11.1U
+ AS=55.2P PS=9.9U
M59 otabotgnd otaCMFBin otadrivsrc GND! NMOS30 W=32.5U L=1.5U AD=55.2P PD=9.9U
+ AS=56.4P PS=11.1U
M60 otadrivsrc otaCMFBin otabotgnd GND! NMOS30 W=32.5U L=1.5U AD=56.4P PD=11.1U
+ AS=55.2P PS=9.9U
M61 otadrivsrc otatailgate otabotgnd GND! NMOS30 W=32.5U L=1.5U AD=56.4P
+ PD=11.1U AS=55.2P PS=9.9U
M62 otabotgnd otatailgate otadrivsrc GND! NMOS30 W=32.5U L=1.5U AD=55.2P PD=9.9U
```

```
C10 vi_ctrl_l agnd 13.00F
C11 vi_ctrl_r agnd 13.00F
C12 vi_ctrl_l vi_1_l 208.00F
C13 otavin_r agnd 14.00F
C14 otavo_l agnd 375.00F
C15 otavin_l vi_2_l 402.00F
C16 vi_2_l agnd 375.00F
C17 otavin_r vi_2_r 402.00F
C18 vi_1_r agnd 209.00F
C19 vi_1_l 0 35.0F
C20 vi_1_r 0 35.0F
C21 vi_2_l 0 41.0F
C22 vi_2_r 0 41.0F
C23 otavo_l 0 41.0F
C24 otavo_r 0 41.0F
C25 otavin_l 0 13.0F
C26 otavin_r 0 13.0F
C27 avdd 0 22004.0F
C28 vi_ctrl_l 0 12.0F
C29 vi_ctrl_r 0 12.0F
C30 otacmfbin_l 0 13.0F
C31 otacmfbin_r 0 13.0F
C32 agnd 0 478.0F
*** Node Listing for subckt:  samcaps
**  0
.ENDS


*****  Subcircuit from file ./samswitches.ext
.SUBCKT samswitches 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22
+ 23 24 25 26

x1 15 13 11 8 7 2 21 19 17 26 25 25 1 23 10 10 1 4 22 24 3 3 3 3 20 20 18 6 23
+ hlfswitches

x2 16 14 12 9 5 2 21 19 17 26 25 17 1 23 10 21 4 4 24 22 3 3 3 3 20 19 18 6 23
+ hlfswitches


C1 23 18 1.00F
C2 6 18 1.00F
C3 6 18 1.00F
C4 23 18 1.00F
C5 24 0 4.0F
C6 21 0 2.0F
C7 23 0 1.0F
C8 1 0 1.0F
C9 17 0 1.0F
C10 10 0 2.0F
C11 20 0 1.0F
C12 26 0 3.0F
C13 22 0 4.0F
C14 4 0 1.0F
C15 3 0 6.0F
C16 19 0 1.0F
C17 25 0 1.0F
C18 18 0 19.0F
*** Node Listing for subckt:  samswitches
**  0
**  1   9_388_21#
**  1   hlfswitches_0/vi+
**  1   hlfswitches_1/9_36_21#
**  1   hlfswitches_0/9_36_21#
**  2   hlfswitches_1/AGC+_ctrl
**  2   hlfswitches_0/AGC+_ctrl
**  3   9_3_85#
**  3   hlfswitches_1/9_36_28#
**  3   hlfswitches_1/9_36_37#
**  3   9_66_28#
**  3   9_66_37#
**  3   hlfswitches_1/vicm
**  3   hlfswitches_1/9_36_85#
**  3   hlfswitches_0/9_36_28#
**  3   9_66_85#
**  3   hlfswitches_0/9_36_37#
**  3   hlfswitches_0/vicm
**  3   9_3_28#
**  3   9_3_37#
**  3   hlfswitches_0/9_36_85#
**  4   hlfswitches_1/9_36_12#
**  4   hlfswitches_1/vi+
**  4   hlfswitches_0/9_36_12#
**  4   9_325_12#
**  5   hlfswitches_1/vi+_ctrl
**  6   hlfswitches_1/refn
**  6   hlfswitches_0/refn
**  7   hlfswitches_0/vi+_ctrl
**  8   hlfswitches_0/phieval2
**  9   hlfswitches_1/phieval2
**  10  hlfswitches_0/9_36_133#
**  10  9_324_133#
**  10  hlfswitches_0/vi+_1
**  10  hlfswitches_1/9_36_133#
**  10  9_406_133#
**  11  hlfswitches_0/phisamp1
**  12  hlfswitches_1/phisamp1
**  13  hlfswitches_0/phi1
**  14  hlfswitches_1/phi1
**  15  hlfswitches_0/phieval1
**  16  hlfswitches_1/phieval1
**  17  9_293_165#
**  17  hlfswitches_0/9_36_165#
**  17  hlfswitches_1/OTAVO-
**  17  hlfswitches_1/9_36_165#
**  18  9_4_44#
**  18  hlfswitches_0/6_5_229#
**  18  hlfswitches_1/6_5_229#
**  18  9_4_120#
**  19  9_307_53#
**  19  hlfswitches_1/9_36_53#
**  19  hlfswitches_0/9_36_53#
**  19  hlfswitches_1/vi+_2
**  20  hlfswitches_0/vi+_2
**  20  9_370_69#
**  20  hlfswitches_1/9_36_69#
**  20  hlfswitches_0/9_36_69#
**  21  hlfswitches_0/9_36_101#
**  21  9_261_101#
**  21  hlfswitches_1/9_36_101#
**  21  hlfswitches_1/vi+_1
**  21  9_343_101#
**  22  9_13_86#
**  22  hlfswitches_0/OTAVIN+
**  22  hlfswitches_1/8_96_86#
**  23  hlfswitches_0/OTACMFBIN
**  23  hlfswitches_0/9_36_213#
**  23  hlfswitches_1/OTACMFBIN
**  23  hlfswitches_1/9_36_213#
```

```
C46 6 0 14.0F
C47 vicm 0 32.0F
C48 phisampl 0 8.0F
C49 13 0 23.0F
C50 11 0 14.0F
C51 1 0 8.0F
C52 2 0 8.0F
C53 vi+_ctrl 0 10.0F
C54 14 0 8.0F
C55 OTAVIN+ 0 9.0F
C56 OTACMFBIN 0 11.0F
C57 8 0 8.0F
C58 12 0 14.0F
C59 10 0 10.0F
C60 7 0 14.0F
C61 phieval1 0 10.0F
C62 phieval2 0 7.0F
C63 3 0 14.0F
C64 phi1 0 4.0F
C65 15 0 29.0F
C66 OTAVO- 0 17.0F
C67 4 0 604.0F
C68 5 0 14.0F
C69 vi+_1 0 17.0F
C70 vi+_2 0 9.0F
C71 vi+ 0 7.0F
C72 refn 0 22.0F
*** Node Listing for subckt:  hlfswitches
**  0
**  1   9_36_101#
**  2   9_36_53#
**  3   9_36_165#
**  4   8_36_229#
**  5   9_36_197#
**  6   9_36_21#
**  7   9_36_213#
**  8   9_36_133#
**  9   9_36_12#
**  10  8_96_86#
**  11  9_36_37#
**  12  9_36_85#
**  13  9_36_28#
**  14  9_36_69#
**  15  6_5_229#
.ENDS


*****  Subcircuit from file ./stdcells/driver.ext
.SUBCKT stdcells/driver Vdd in GND out

M1 out internal Vdd Vdd PMOS W=19.0U L=1.0U AD=28.5P PD=3.0U AS=40.1P PS=16.7U
M2 Vdd in internal Vdd PMOS W=11.0U L=1.0U AD=23.2P PD=9.7U AS=16.5P PS=3.0U
M3 internal in Vdd Vdd PMOS W=11.0U L=1.0U AD=16.5P PD=3.0U AS=23.2P PS=9.7U
M4 Vdd internal out Vdd PMOS W=19.0U L=1.0U AD=40.1P PD=16.7U AS=28.5P PS=3.0U
M5 out internal Vdd Vdd PMOS W=19.0U L=1.0U AD=28.5P PD=3.0U AS=40.1P PS=16.7U
M6 Vdd internal out Vdd PMOS W=19.0U L=1.0U AD=40.1P PD=16.7U AS=28.5P PS=3.0U
M7 GND in internal GND! NMOS W=4.0U L=1.0U AD=10.0P PD=6.1U AS=6.0P PS=3.0U
M8 internal in GND GND! NMOS W=4.0U L=1.0U AD=6.0P PD=3.0U AS=10.0P PS=6.1U
M9 out internal GND GND! NMOS W=15.0U L=1.0U AD=22.5P PD=3.0U AS=37.5P
+ PS=22.9U
M10 GND internal out GND! NMOS W=15.0U L=1.0U AD=37.5P PD=22.9U AS=22.5P
+ PS=3.0U

C1 in internal 1.00F
C2 out Vdd 2.00F
C3 internal Vdd 1.00F
C4 out internal 1.00F
C5 in 0 4.0F
C6 internal 0 12.0F
C7 GND 0 10.0F
C8 Vdd 0 368.0F
C9 out 0 8.0F
*** Node Listing for subckt:  stdcells/driver
**  0
.ENDS


*****  Subcircuit from file ./teleota.ext
.SUBCKT teleota OTAVDD OTAPSRCBIAS otatailgate OTAPSRCCASCBIAS OTAGND OTAVO+
+ OTAVIN+ OTAVIN- OTACMFBIN OTAVO-

x1 1 2 OTAVO+ OTACMFBIN OTAPSRCCASCBIAS OTAPSRCBIAS otatailgate OTAVIN- OTAGND
+ OTAVDD telehlf

x2 1 2 OTAVO- OTACMFBIN OTAPSRCCASCBIAS OTAPSRCBIAS otatailgate OTAVIN+ OTAGND
+ OTAVDD telehlf


C1 2 OTAGND 2.00F
C2 OTAVDD OTACMFBIN 1.00F
C3 OTACMFBIN OTAGND 9.00F
C4 otatailgate 0 3.0F
C5 OTAVIN+ 0 3.0F
C6 OTAVIN- 0 3.0F
C7 OTACMFBIN 0 30.0F
C8 OTAVDD 0 662.0F
C9 OTAVO+ 0 3.0F
C10 OTAVO- 0 3.0F
C11 OTAPSRCBIAS 0 1.0F
C12 2 0 2.0F
C13 OTAPSRCCASCBIAS 0 1.0F
C14 OTAGND 0 37.0F
*** Node Listing for subckt:  teleota
**  0
**  1   telehlf_1/otacascbias
**  1   telehlf_0/otacascbias
**  2   10_65_38#
**  2   telehlf_1/otadrivsrc
**  2   telehlf_0/otadrivsrc
.ENDS


*****  Subcircuit from file ./samcaps.ext
.SUBCKT samcaps otavin_1 vi_ctrl_1 vi_ctrl_r otacmfbin_1 otacmfbin_r vi_1_r
+ vi_2_r vi_2_1 vi_1_1 otavo_r otavin_r otavo_1 avdd agnd

M1 avdd agnd avdd avdd PMOS W=159.5U L=279.0U AD=717.8P PD=168.5U AS=717.8P
+ PS=168.5U

C1 otacmfbin_1 agnd 14.00F
C2 otavin_1 agnd 14.00F
C3 otacmfbin_r agnd 14.00F
C4 vi_1_1 agnd 209.00F
C5 vi_2_r agnd 375.00F
C6 vi_ctrl_r vi_1_r 208.00F
C7 otavo_r agnd 375.00F
C8 otacmfbin_r otavo_r 402.00F
C9 otacmfbin_1 otavo_1 402.00F
```

```
* comparator
* from extracted layout of casc_sr_test.mag

.options nomod post=2

* level parameter specifies which bit this comparator detects
.param wmin=8.0u cs_scale=2 ro_factor=3
+ level=1
+ stepsize=33.3m

* Transistors
.include casc_sr_test.spice

mibias bias bias avdd avdd pmos w='wmin*2' l='1.0u*ro_factor'
+ ad='wmin*2*2.5u' as='wmin*2*2.5u'
+ pd='wmin*2+5u' ps='wmin*2+5u'

vcascbias cascbias agnd .75

* clocks
vphi1 phi1 dgnd pulse (0 5.0 4ns .6ns .6ns 2.5ns 8ns)
vphi2 phi2 dgnd pulse (0 5.0 0 .6ns .6ns 2.5ns 8ns)

*bias
vdda vdd_ideal 0 5.0
lavdd vdd_ideal avdd 5nH
lsub vdd_ideal subvdd 5nH
vdd dvdd_ideal 0 3.3
ldvdd dvdd_ideal dvdd 5nH
ib bias 0 10u
lagnd agnd 0 2nH
ldgnd dgnd 0 2nH

* input
vref_cm ref_cm agnd dc 1.7
vref_p ref_p ref_cm dc 'stepsize*level'
eref ref_n ref_cm ref_p ref_cm -1
.include /users/lynn/adc/flash/input/input1.sp

.include /users/lynn/adc/power_calc.sp
.include /users/lynn/adc/analpower.sp
.include /users/ssheng/rf/mosis/08hp/10model.139

*analysis
.op 10ns 11.9ns 14ns
.tran .1n 120n
.probe i(vdummy1) i(vdummy2) v(nlatch1,nlatch2) v(platch1,platch2)
.meas tran srq_samp1 find v(srq) when v(phi1)=2.5v td=14n fall=1
.meas tran srq_samp2 find v(srq) when v(phi1)=2.5v td=14n fall=2
.meas tran srq_samp3 find v(srq) when v(phi1)=2.5v td=14n fall=3
.meas tran srq_samp4 find v(srq) when v(phi1)=2.5v td=14n fall=4
.meas tran srq_samp5 find v(srq) when v(phi1)=2.5v td=14n fall=5
.meas tran srq_samp6 find v(srq) when v(phi1)=2.5v td=14n fall=6
.meas tran srq_samp7 find v(srq) when v(phi1)=2.5v td=14n fall=7
.meas tran srq_samp8 find v(srq) when v(phi1)=2.5v td=14n fall=8
.meas tran srq_samp9 find v(srq) when v(phi1)=2.5v td=14n fall=9
.meas tran srq_samp10 find v(srq) when v(phi1)=2.5v td=14n fall=10
.meas tran srq_samp11 find v(srq) when v(phi1)=2.5v td=14n fall=11
.meas tran srq_samp12 find v(srq) when v(phi1)=2.5v td=14n fall=12
.meas tran srq_samp13 find v(srq) when v(phi1)=2.5v td=14n fall=13
.meas tran srq_samp14 find v(srq) when v(phi1)=2.5v td=14n fall=14
.end
```

```
**   23   9_30_213#
**   24   hlfswitches_0/8_96_86#
**   24   hlfswitches_1/OTAVIN+
**   24   9_13_102#
**   25   hlfswitches_1/9_36_197#
**   25   9_356_197#
**   25   hlfswitches_0/9_36_197#
**   25   hlfswitches_0/OTAVO-
**   26   hlfswitches_1/8_36_229#
**   26   9_4_229#
**   26   hlfswitches_0/8_36_229#
.ENDS

*****  Subcircuit from file ./clkbuff.ext
.SUBCKT clkbuff phie2in dvdd phie2 phi1in phie1in phis1 phi1 phie1 phis1in
+ dgnd

x1 dvdd phie2in dgnd phie2 stdcells/driver

x2 dvdd phi1in dgnd phi1 stdcells/driver

x3 dvdd phis1in dgnd phis1 stdcells/driver

x4 dvdd phie1in dgnd phie1 stdcells/driver


C1 phis1 dgnd 1.00F
C2 dgnd phie2 1.00F
C3 dgnd phie1 1.00F
C4 phi1 dgnd 1.00F
C5 phie1 0 27.0F
C6 phie2 0 28.0F
C7 dvdd 0 35.0F
C8 phis1 0 16.0F
C9 phi1 0 16.0F
C10 dgnd 0 34.0F
*** Node Listing for subckt:  clkbuff
**   0
.ENDS

****** top level cell is ./samnet.ext
x1 otavdd otapsrcbias otatailgate otapsrccascbias otagnd otavo+ otavin+
+ otavin- otacmfbin otavo- teleota

x2 otavin+ 1 2 otacmfbin otacmfbin vi+_1 vi-_2 vi+_2 vi-_1 otavo+ otavin-
+ otavo- otavdd otagnd samcaps

x3 vi+ agc_ctrl vicm vi- 1 otatailgate 2 clkbuff_0/phie2 clkbuff_0/phie2 vi+_1
+ clkbuff_0/phis1 clkbuff_0/phis1 clkbuff_0/phi1 clkbuff_0/phi1
+ clkbuff_0/phie1 clkbuff_0/phie1 otavo+ otagnd vi-_2 vi+_2 vi-_1 otavin-
+ otacmfbin otavin+ otavo- otagnd samswitches

x4 phie2in dvdd clkbuff_0/phie2 phi1in phie1in clkbuff_0/phis1 clkbuff_0/phi1
+ clkbuff_0/phie1 phis1in dgnd clkbuff


C1 otagnd vi-_2 1.00F
C2 otagnd vi+_2 1.00F
C3 otagnd vi-_1 1.00F
C4 otavo+ otagnd 1.00F
C5 vi+_1 otagnd 1.00F
C6 otagnd vi-_2 1.00F
C7 otavin- otagnd 1.00F
C8 otagnd otacmfbin 4.00F
C9 otagnd vi+_2 1.00F
C10 otagnd otavin+ 1.00F
C11 otagnd otavo- 1.00F
C12 otagnd vi-_1 1.00F
C13 otacmfbin otagnd 4.00F
C14 otacmfbin otagnd 4.00F
C15 otavo+ otagnd 1.00F
C16 otagnd otavin+ 1.00F
C17 otacmfbin otagnd 4.00F
C18 otavin- otagnd 1.00F
C19 otagnd otavo- 1.00F
C20 otavdd otavo+ 1.00F
C21 otagnd otavo+ 1.00F
C22 otavo+ otagnd 4.00F
C23 otavo+ otagnd 3.00F
C24 otavin+ otagnd 1.00F
C25 otavin- otagnd 1.00F
C26 otacmfbin otagnd 2.00F
C27 otavo- otagnd 4.00F
C28 otavo- otavdd 1.00F
C29 otavo- otagnd 3.00F
C30 clkbuff_0/phie2 0 28.0F
C31 clkbuff_0/phie1 0 30.0F
C32 otagnd 0 76.0F
C33 otapsrcbias 0 40.0F
C34 otatailgate 0 48.0F
C35 clkbuff_0/phi1 0 3.0F
C36 dvdd 0 1.0F
C37 2 0 3.0F
C38 1 0 3.0F
C39 otavo+ 0 19.0F
C40 otavo- 0 19.0F
C41 otavdd 0 75.0F
C42 dgnd 0 8.0F
C43 otapsrccascbias 0 39.0F
C44 vi+_1 0 6.0F
C45 vi+_2 0 8.0F
C46 vi-_1 0 6.0F
C47 vi-_2 0 7.0F
C48 otacmfbin 0 36.0F
C49 clkbuff_0/phis1 0 5.0F
C50 otavin+ 0 10.0F
C51 otavin- 0 10.0F
*** Node Listing for subckt:  samnet
**   0
**   1   samcaps_0/vi_ctrl_1
**   1   10_556_954#
**   1   samswitches_0/hlfswitches_1/vi+_ctrl
**   2   10_1110_954#
**   2   samswitches_0/hlfswitches_0/vi+_ctrl
**   2   samcaps_0/vi_ctrl_r
```

```
C44 agnd 0 44.0F
*** Node Listing for subckt:  casc_ful
**  0
**  1   9_29_104#
**  1   casc_hlf_1/common2
**  1   casc_hlf_0/common1
**  2   9_35_126#
**  2   casc_hlf_1/common1
**  2   casc_hlf_0/common2
.ENDS

*****  Subcircuit from file ./sr.ext
.SUBCKT sr 1 Q RST1 SET2 GND Q_b Vdd

M1 Vdd SET2 Q 1 PMOS W=6.0U L=1.0U AD=12.0P PD=7.0U AS=9.0P PS=3.0U
M2 Q Q_b Vdd 1 PMOS W=6.0U L=1.0U AD=9.0P PD=3.0U AS=12.0P PS=7.0U
M3 Vdd Q Q_b 1 PMOS W=6.0U L=1.0U AD=12.0P PD=7.0U AS=9.0P PS=3.0U
M4 Q_b RST1 Vdd 1 PMOS W=6.0U L=1.0U AD=9.0P PD=3.0U AS=12.0P PS=7.0U
M5 Q SET2 2 GND! NMOS W=4.0U L=1.0U AD=9.0P PD=9.0U AS=2.0P PS=1.0U
M6 2 Q_b GND GND! NMOS W=4.0U L=1.0U AD=2.0P PD=1.0U AS=6.0P PS=3.0U
M7 GND Q 3 GND! NMOS W=4.0U L=1.0U AD=6.0P PD=3.0U AS=2.0P PS=1.0U
M8 3 RST1 Q_b GND! NMOS W=4.0U L=1.0U AD=2.0P PD=1.0U AS=8.0P PS=9.0U

C1 Q_b 0 5.0F
C2 1 0 145.0F
C3 Q 0 5.0F
C4 GND 0 4.0F
C5 Vdd 0 6.0F
C6 SET2 0 3.0F
C7 RST1 0 3.0F
*** Node Listing for subckt:  sr
**  0
**  1   6_39_46#
**  2   8_7_20#
**  3   8_18_20#
.ENDS

****** top level cell is ./casc_sr.ext
x1 sr_0/Vdd casc_ful_0/phi1 sr_0/RST1 sr_0/SET2 casc_ful

x2 sr_0/Vdd sr_0/Q sr_0/RST1 sr_0/SET2 dgnd sr_0/Q_b sr_0/Vdd sr

M1 sr_0/Vdd dgnd sr_0/Vdd sr_0/Vdd PMOS W=8.5U L=10.0U AD=21.2P PD=13.7U
+ AS=21.2P PS=13.7U
M2 sr_0/Vdd dgnd sr_0/Vdd sr_0/Vdd PMOS W=8.0U L=10.0U AD=20.0P PD=12.8U
+ AS=20.0P PS=12.8U

C1 dgnd sr_0/Vdd 5.00F
C2 sr_0/RST1 dgnd 1.00F
C3 sr_0/SET2 dgnd 1.00F
C4 sr_0/Q dgnd 1.00F
C5 sr_0/RST1 sr_0/Vdd 1.00F
C6 sr_0/SET2 sr_0/Vdd 1.00F
C7 sr_0/Q_b dgnd 1.00F
C8 dgnd casc_ful_0/phi1 6.00F
C9 dgnd sr_0/RST1 4.00F
C10 dgnd sr_0/Vdd 5.00F
C11 dgnd sr_0/SET2 4.00F
C12 dgnd sr_0/Vdd 3.00F
C13 dgnd dgnd 12.00F
C14 sr_0/RST1 dgnd 1.00F
C15 dgnd sr_0/Vdd 16.00F
C16 dgnd sr_0/Vdd 12.00F
C17 sr_0/SET2 dgnd 1.00F
C18 sr_0/SET2 0 2.0F
C19 sr_0/RST1 0 2.0F
C20 dgnd 0 44.0F
C21 sr_0/Vdd 0 226.0F
*** Node Listing for subckt:  casc_sr
**  0
```

```
***** Subcircuit from file ./casc_hlf.ext
.SUBCKT casc_hlf dvdd platch1 platch2 in_1 nlatch1 phi1 nlatch2 agnd phi2 avdd
+ subvdd refin casc common1 common2 bias

M1 platch2 phi1 dvdd dvdd PMOS W=8.0U L=1.0U AD=12.0P PD=2.9U AS=15.5P PS=6.2U
M2 dvdd platch1 platch2 dvdd PMOS W=13.5U L=1.0U AD=26.2P PD=10.4U AS=20.2P
+ PS=4.9U
M3 platch2 platch1 dvdd dvdd PMOS W=13.5U L=1.0U AD=20.2P PD=4.9U AS=26.2P
+ PS=10.4U
M4 dvdd platch1 platch2 dvdd PMOS W=13.5U L=1.0U AD=26.2P PD=10.4U AS=20.2P
+ PS=4.9U
M5 nlatch2 phi1 platch2 GND! NMOS W=3.0U L=1.0U AD=4.9P PD=3.1U AS=7.5P
+ PS=8.0U
M6 agnd nlatch1 nlatch2 GND! NMOS W=5.5U L=1.0U AD=10.1P PD=5.5U AS=8.9P
+ PS=5.8U
M7 nlatch2 nlatch1 agnd GND! NMOS W=5.5U L=1.0U AD=8.9P PD=5.8U AS=10.1P
+ PS=5.5U
M8 nlatch1 phi2 nlatch2 GND! NMOS W=1.5U L=2.0U AD=4.8P PD=7.5U AS=2.4P
+ PS=1.6U
M9 agnd nlatch1 nlatch2 GND! NMOS W=5.5U L=1.0U AD=10.1P PD=5.5U AS=8.9P
+ PS=5.8U
M10 casc 1 nlatch2 subvdd PMOS W=4.0U L=1.0U AD=6.8P PD=3.0U AS=10.0P PS=9.0U
M11 avdd bias common1 subvdd PMOS W=8.0U L=3.0U AD=12.0P PD=3.0U AS=14.7P
+ PS=6.3U
M12 common1 bias avdd subvdd PMOS W=8.0U L=3.0U AD=14.7P PD=6.3U AS=12.0P
+ PS=3.0U
M13 casc in common1 subvdd PMOS W=8.0U L=1.0U AD=13.6P PD=6.0U AS=14.7P
+ PS=6.3U
M14 common2 refin casc subvdd PMOS W=8.0U L=1.0U AD=14.7P PD=6.3U AS=13.6P
+ PS=6.0U
M15 avdd bias common2 subvdd PMOS W=8.0U L=3.0U AD=12.0P PD=3.0U AS=14.7P
+ PS=6.3U
M16 common2 bias avdd subvdd PMOS W=8.0U L=3.0U AD=14.7P PD=6.3U AS=12.0P
+ PS=3.0U

C1 casc bias 1.00F
C2 platch2 dvdd 2.00F
C3 nlatch2 nlatch1 1.00F
C4 refin avdd 1.00F
C5 nlatch2 agnd 1.00F
C6 agnd phi2 1.00F
C7 casc avdd 1.00F
C8 avdd bias 1.00F
C9 subvdd avdd 2.00F
C10 platch1 dvdd 1.00F
C11 platch2 agnd 1.00F
C12 refin bias 1.00F
C13 common2 avdd 2.00F
C14 agnd nlatch1 1.00F
C15 platch1 platch2 1.00F
C16 nlatch2 subvdd 1.00F
C17 refin 0 4.0F
C18 casc 0 6.0F
C19 bias 0 9.0F
C20 in 0 1.0F
C21 1 0 1.0F
C22 dvdd 0 196.0F
C23 phi1 0 2.0F
C24 phi2 0 4.0F
C25 subvdd 0 404.0F
C26 avdd 0 8.0F
C27 nlatch1 0 4.0F
C28 nlatch2 0 7.0F
C29 platch1 0 3.0F
C30 platch2 0 6.0F
C31 agnd 0 11.0F
C32 common1 0 4.0F
C33 common2 0 4.0F
*** Node Listing for subckt:  casc_hlf
**  0
**  1   8_50_15#
.ENDS


***** Subcircuit from file ./casc_ful.ext
.SUBCKT casc_ful dvdd phi1 platch1 platch2

x1 dvdd platch2 platch1 in_n cascbias nlatch1 phi1 nlatch2 agnd phi2
+ casc_hlf_0/avdd subvdd ref_p dummy1 1 2 bias casc_hlf

x2 dvdd platch1 platch2 in_p cascbias nlatch2 phi1 nlatch1 agnd phi2
+ casc_hlf_0/avdd subvdd ref_n dummy2 2 1 bias casc_hlf


C1 nlatch2 agnd 3.00F
C2 phi1 dvdd 1.00F
C3 subvdd agnd 3.00F
C4 nlatch1 agnd 3.00F
C5 phi2 agnd 2.00F
C6 agnd ref_n 1.00F
C7 agnd dummy1 2.00F
C8 agnd 1 3.00F
C9 agnd casc_hlf_0/avdd 24.00F
C10 bias 2 1.00F
C11 agnd subvdd 8.00F
C12 agnd ref_p 1.00F
C13 agnd dummy2 2.00F
C14 agnd 2 3.00F
C15 agnd bias 4.00F
C16 bias 1 1.00F
C17 agnd phi2 5.00F
C18 casc_hlf_0/avdd in_n 1.00F
C19 agnd casc_hlf_0/avdd 3.00F
C20 agnd 2 1.00F
C21 agnd agnd 1.00F
C22 agnd in_p 1.00F
C23 agnd subvdd 1.00F
C24 agnd in_n 1.00F
C25 agnd cascbias 6.00F
C26 casc_hlf_0/avdd in_p 1.00F
C27 agnd 1 1.00F
C28 bias agnd 6.00F
C29 in_n 0 3.0F
C30 in_p 0 3.0F
C31 bias 0 9.0F
C32 dvdd 0 71.0F
C33 cascbias 0 7.0F
C34 casc_hlf_0/avdd 0 2.0F
C35 phi1 0 11.0F
C36 phi2 0 6.0F
C37 subvdd 0 144.0F
C38 1 0 3.0F
C39 2 0 3.0F
C40 nlatch1 0 2.0F
C41 nlatch2 0 2.0F
C42 platch1 0 2.0F
C43 platch2 0 2.0F
```