Low-Power Techniques for
High-Speed Wireless Baseband Applications

by

Keith Ken Onodera

B.S. (Harvey Mudd College) 1974
M.S. (Stanford University) 1976

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy
in

Engineering - Electrical Engineering
and Computer Sciences

in the

GRADUATE DIVISION

of the

UNIVERSITY OF CALIFORNIA, BERKELEY

Committee in charge:

Professor Paul R. Gray, Chair
Professor Robert W. Brodersen
Professor Charles J. Stone

Fall 2000

The dissertation of Keith Ken Onodera is approved:

Paul R. Gray — Dec 13 00
Chair — Date

Robert W Brodersen — 12/13/00
Date

Charles G Sodini — 12/14/00
Date

University of California, Berkeley

Low-Power Techniques for
High-Speed Wireless Baseband Applications

by

Keith Ken Onodera

Abstract

Low-Power Techniques for
High-Speed Wireless Baseband Applications

by

Keith Ken Onodera

Doctor of Philosophy in Electrical Engineering
and Computer Sciences

University of California, Berkeley

Professor Paul R. Gray, Chair

The application of digital techniques to every facet of electronic design continues to be a goal in the engineering community due to the many benefits of digitization (noise tolerance, precision control, increased functionality, programmability, etc.). As this goal is applied to the newest electronic markets, the highest-performance analog-to-digital (A/D) converters are required which tend to consume a large amount of power. In the growing area of wireless communications, large power consumption is contrary to battery life, motivating the search for low-power alternatives.

One possible low-power alternative is to perform some of the initial signal processing in the analog domain thereby relaxing the speed and/or resolution requirements of the A/D converter, thus lowering its power consumption. If the analog processing circuits dissipate roughly the same amount of power as the digital circuits they displaced, then the overall system power will be lower. The potential benefits of this analog approach is the motivation of this research.

The ideas and techniques developed in this work are applied to a integrated circuit (IC) prototype that implements a baseband data recovery system for a wireless direct-sequence, code-division multiple access (DS-CDMA) system. In the course of this research, the following contributions are presented: (1) a first-order relationship between power of an A/D converter and its speed (sampling rate) and resolution is derived, (2) a passive charge-error-cancellation technique is developed, permitting switched-capacitor integration without an operational amplifier, (3) low-power digital techniques are developed that further lower the dynamic power, (4) a CMOS DS-CDMA IC prototype capable of operation at 128MHz and dissipating 75mW is demonstrated.

Paul R. Gray, Chairman of Committee

To my parents, Ken and Joan Onodera

Table of Contents

Acknowledgements

I feel very fortunate to have had the opportunity to do my graduate research at the University of California, Berkeley, in an outstanding EECS program. The incredible educational and personal growth that I experienced these past years will forever be responsible for any success I may have. It was a long, tough trail but the many good friendships I made and the great times we shared, more than compensated for the long hours of research work. I would like to mention a few of the many people responsible for making my graduate experience so rewarding.

First and foremost is my advisor, Professor Paul Gray, whose encouragement, guidance and trust was invaluable to my academic and personal development. When and if the time comes, I hope that I will be able to offer someone assistance with as much wisdom. I would also like to thank Professor Robert Brodersen for all his time and advice on my research project as well as chairing my qualifying examination committee and reviewing my dissertation. I also appreciate Professor Bernhard Boser and Professor Charles Stone for being part of my qualifying committee and for all their useful discussions.

An important reason that I chose Berkeley to do my graduate work was the excellent courses in integrated circuit design. Two professors stand out as superb instructors; they are Professor Robert Meyer and Professor Chenming Hu who both present material in an intuitive and simple way. A real pleasure to attend their lecturers.

A couple of people that are indispensible to us grad students are the graduate assis-

tants (our moms away from home). When I say, "I couldn't have done it without you", I'm talking about Heather Brown and Ruth Gjerde. Thank you both for all that you did for me and that you do for all the Cory grad students.

Other people that get to see us come and go, and who are responsible for making the wheels turn in Cory Hall, are the staff members. A very special "thank you" to Tom Boot and Peggye Brown for all the administrative assistance and more importantly their friendship over the years. "Thank you" goes to our building manager, Eleta Cook, for all her help and patience, especially with the environmental controls in 550. Thanks to Loretta Lutcher, Carol Sitea, Diane Chang, Carol Block and Elise Mills for all their help over the years and thanks to Tito Gatchalian and Jeff Wilkinson for their help with ERL memorandums. And to Pam Atkinson and Isabel Blanco, "thanks" for running such a useful and beneficial remote learning program and for letting me take part in it.

A special "thanks" to those friends and fellow students with whom I shared a cubicle and a lot of good times; Andy Abo, Jeff Ou and Sekhar Narayanaswami. I'm glad I opted out of the "window cubicle" sweepstakes. To all the other close friends from Cory and Keeler: Srenik Mehta, Chris Rudell, Dennis Yee, Tom Burd, Joyce Law, Arya Behzad, Tony Stratakos, Carol Barrett, Dave Lidsky, Lisa Guerra, Arnie Feldman and Jeff Weldon; thanks for all the great times and being there for the tough times. I'll forever remember the pick-up basketball games we had, and for that I can thank: Andy, Sek, Tom, Dave, Jeff, Tony, Dennis, Kevin Stone, Lapoe Lynn, Eric Boskin, Craig Teuscher, Dave Sobel, Robert Neff and Todd Weigandt. And to all the many fellow Cory students over the years:

Ed Liu, Scarlett Wu, Tom Wongkomet, Anna Ison, Marlene Wan, Renu Mehra, Arthur Abnous, Sam Sheng, Darrin Young, Ingrid Ma, Cynthia Keys, Jennie Chen, Crist Lu, Henry Sheng, James Chen, Manolis Terrovitis, Edwin Chan, Iasson Vassiliou, Tony Miranda, Dave Rodriguez, Nathan Yee, Sovarong Leang and fellow PRG students; thanks for keeping me sane with suitable diversions from an insane scholarly quest.

I was very fortunate to be in a research group with many excellent colleagues and good friends. I have the "veterans" to thank (Cormac Conroy, Ken Nishimura, Greg Uehara, Ed Liu, Weijie Yun, Gani Jusuf, Tim Hu, Thomas Cho, Dave Cline, Robert Neff and Henry Chang) for helping me to adjust to graduate life; my contemporaries (Andy, Sekhar, Chris, Srenik, Jeff O., Jeff W., Arnie, Carol, Todd, George Chien, Li Lin, Martin Tsai, Luns Tee, Caesar Wong, Steve Lo, Greg Wardle, Kelvin Khoo, Danelle Au and Troy Robinson) for helping me enjoy graduate life; and the "rookies" (Ryan Bocock, Ken Wojciechowski, Yun Chiu, Nathan Sneed, Tim Wongkomet, Gabe Desjardins and Cheol Lee) for making me glad that I'm done.

I'd like to mention a few people outside of Berkeley that gave me moral support through my graduate years: my friends in the BBBS and Adventure Trips program (Dave Axthelm, Ed Mahler, Ed Ferrero and Diane Ruch), my friends in Japan (Tsutomu Kikuchi, Chiaki Shioda, Tomoko Onuma and Noriko Hanawa). my basketball friends (Rich & Janis Yamaguma, John & Suanne Higashidani, Tim & Lee Garverick, Bill Shirado and Dan & Atsuko Kojiro), my National friends (Mike Rana, Thai Nguyen, Shu-Ing Ju, Alan Kondo, Kumar Sivasothy, Neal Domen, Charlie Carinalli, Vickie & Andrew Pagnon, Bill Vil-

lamor, Bill Ching and Prasun Paul) and my house-mates (Leslie Lee, Silvia Uchida, Mei-Fang Chou and Cintia Miura). I'm indebted to you all for the friendship and support you've given me over the years. And a very special note of gratitude is reserved for my close friend, LeAnn Sucht, who inspired me to return to school by her example.

My foundation through the whole program were my Little Brothers: Ken Davis, Colin (Cru) Yocum, Michael Barnes, Anthony Barnes, and Shawn Bird. My strength to persevere the rigors of the program came from them; "Thank you, Bro's". Their mom's (Wendy, Mary Lou, Shelley and Rhonda) are also due some "thanks" for raising such great young men.

And nothing I have ever accomplished would have ever been realized were it not for the love and support from my parents, Ken and Joan, my sisters, Trish and Shirl, and my brother, George. I love you all and dedicate this to you.

Chapter 1

Introduction

In modern society, time is arguably the most precious commodity and it motivates us to look for ways of doing things faster. This is particularly true in the electronics industry where increasing processing speeds has been a never-ending goal. However, the result of this high-speed quest has been an attendant increase in power which raises a new concern about continued progress. This fact coupled with the trend towards battery-powered personal communication systems (cellular phones, portable computers, wireless web access and hand-held global positioning satellite (GPS) systems) creates a conflicting desire for low-power electronics that can deliver high performance. The search for techniques to reduce the power without decreasing the performance or to improve the performance without increasing the power, or both, is the subject of this research.

1.1 Background

In addition to the progression towards higher processing speeds, the electronics industry continues a trend of maximizing the digital circuitry content of integrated circuits

and of minimizing the analog content. This is due to several advantages that digital circuitry has over its analog counterpart, such as, better noise immunity, faster development time (via computer automated software), more flexible & precise monitor/control capabilities and easier extensibility. For these reasons, the industry's approach is one of continuing the migration towards 100% digital content and of relying on fabrication process advancements to improve the speed-power performance.

While this strategy works well with mature technology markets (consumer audio, telephony and voice-band data communications), it produces less than optimal speed-power solutions for the newest high-speed markets (wireless, optical and high-speed wired data communications). The difficulty lies mainly with the analog-to-digital (A/D) conversion process. The high conversion (sampling) rate and/or large dynamic range (resolution) requirements of these new high-speed applications push the latest A/D converters (ADC's) to their limits, resulting in high converter power consumption.

This is illustrated by Fig. 1-1 which shows the extent of current A/D converters' capabilities on a plot of published[1] ADC performance. The *converter boundary* separates the region where both analog and digital processing is possible from where only analog processing is possible. In the former region, a choice can be made to use analog or digital circuitry; digital being chosen often for reasons given above. For applications that do not require A/D performance close to the boundary, i.e., requiring lower speed and resolution,

1. See Chapter 2 for a list of references.

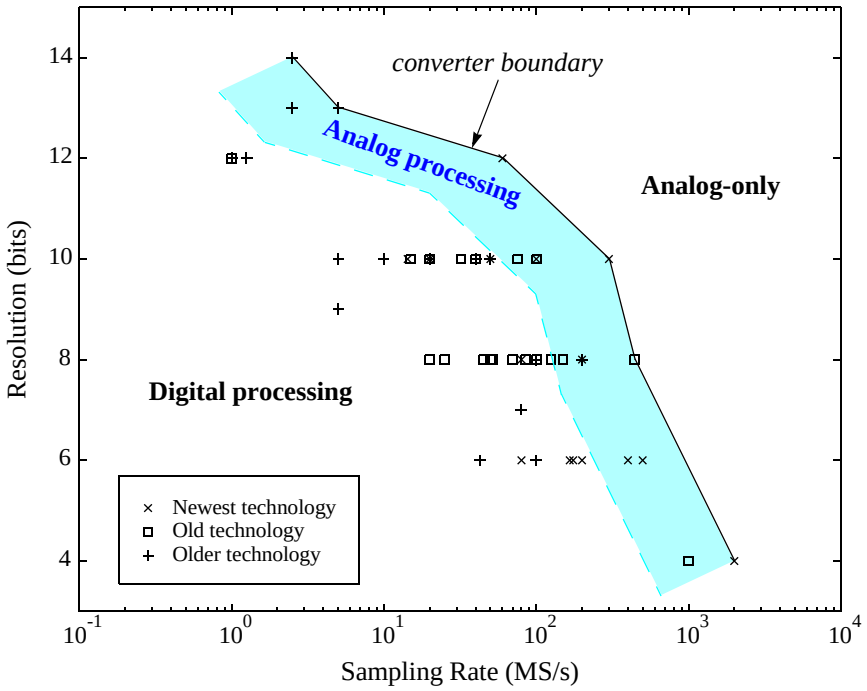


Figure 1-1 Plot of published ADC performance (1988-1999).

low-power A/D conversion exists, permitting digital processing with all its benefits. However, for those applications requiring A/D performance near the boundary (in the shaded region)[1] in order to implement digital processing, a significant power cost is exacted.

In these cases, analog circuitry can perform the initial processing, reducing the signal's dynamic range and bandwidth. Then the subsequent ADC will have lower resolution and/or sampling rate requirements (below the shaded region) resulting in lower power consumption. Assuming that analog processing consumes as much or less power than digital processing for the same task, as suggested by [1], the overall system power consump-

1. The shaded region is of arbitrary width intended to give a qualitative idea of the region where analog processing may be used to lower converter requirements and thus power.

tion can be lowered by using this analog pre-processing approach.

1.2 Research Goals

The above assertions appeal to ones intuition but quantitative analyses and results can be more useful; so this work attempts a first-order analysis of the power consumption of high-speed A/D converters. With this information and some examples of current high-speed applications, a rough quantitative analysis of the above ideas is possible. The results of this analysis can be further tested by fabricating a prototype system and compiling the measured results.

The following is a list of contributions from this research:

1. Derived first-order relationships of ADC power as a function of resolution and sampling rate for high-speed converter architectures (Flash, Folding/Interpolating, Two-step/Subranging and Pipelined converters).

2. Developed a *passive* charge-error-cancellation technique for switched-capacitor integrators that enables implementation of a high-speed low-power *passive* (amplifier-free) PN correlator.

3. Developed low-power digital techniques (low-parasitic timing shift-register chain, local control signal generation, automatic state decoding by multi-tapping timing chain, split-gate static-logic loads, double-latched PN generator chain and bank-select clocking) to reduce dynamic power.

4. Demonstrated that a DS-CDMA baseband recovery integrated circuit, fabricated in a 1.2μm double-metal double-poly CMOS process, which used analog processing

prior to the A/D conversion is capable of operating at 128*MS/s* (*I* and *Q* channels at 64*Mchips/s*) with an output *SNR* of 46.6*dB* while dissipating 75*mW.* This compares favorably to the power of two ADC's (*I* and *Q* channel) required for a digital implementation.

1.3 Dissertation Organization

The ADC power analysis for high-speed A/D converter architectures is carried out in Chapter 2. The application of the resulting analysis to three high-speed communication system examples can be found at the end of the chapter. The DS-CDMA baseband recovery system example is chosen as the demonstration-vehicle prototype. Chapter 3 gives a background description of the DS-CDMA system. Chapter 4 covers design topics dealing with implementation of PN correlators in the analog domain; specifically in the analog sampled-data domain. A description of the prototype vehicle is discussed in Chapter 5 which includes descriptions of low-power digital design techniques. Chapter 6 presents the experimental results measured from the prototype integrated circuit and concluding comments are given in Chapter 7.

1.4 References

[1]. K. Nishimura, "Optimal Partitioning of Analog and Digital Circuitry in Mixed-Signal Circuits for Signal Processing," *Memo UCB/ERL M93/67*, Univ. Calif., Berkeley, July 1993.

Chapter 2

Power Dissipation of Analog-to-Digital Converters

The power dissipation of an analog-to-digital converter (ADC) is a function of many variables, such as sampling rate (f_S), resolution, architecture, process, voltage supply and technology. This chapter will attempt to establish the power dependence on sampling rate and resolution as its primary goal. To make this tenable, the scope of this task will be narrowed in the following two ways:

1. Architectures: Only those ADC's suitable for use in high-speed signal processing applications, i.e., capable of attaining high Nyquist sampling rates, such as Flash, Two-step, Subranging, Folding, Interpolating and Pipelined architectures will be considered.

2. Process: Coverage will be restricted to high-integration capable IC processes such as bipolar, BiCMOS and CMOS processes which allow embedding of the ADC function in a monolithic signal processing chip.

Even with a narrower scope, only a first-order analysis is attempted in light of the many variables that influence the power of an ADC. With this first-order relationship established between speed, resolution and power for an ADC, how system power varies as a

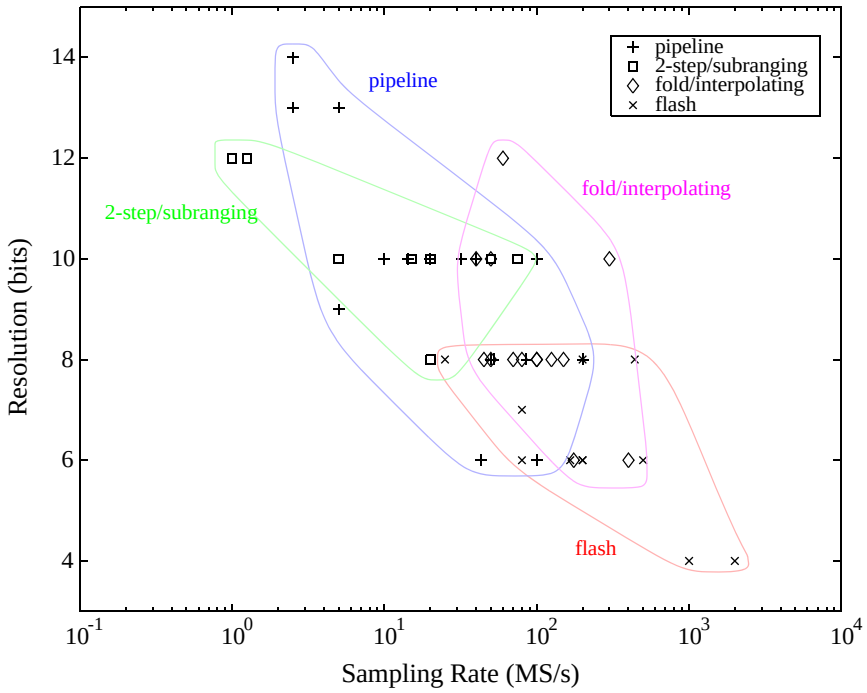


Figure 2-1 Empirical data from published research of ADC's from 1987 to 1999.

function of architectural choices in the implementation of some high-speed signal processing applications can be attempted. These will be carried out at the end of the chapter.

2.1 High-Speed ADC Architectures

Before describing each architecture type, data gathered from published research of these types of ADC's is presented for reference. In Fig. 2-1, the resolution of the ADC's is plotted against their sampling rate (f_S) and organized by ADC type. Two-step and Subranging ADC's [1]-[12] are grouped together due to their similarity in operation and performance. Folding and Interpolating ADC's [13]-[25] are also grouped together because most implementations employ interpolation with their folding structures. The other two

groups are Flash [26]-[37] and Pipelined [38]-[56] converters.

From Fig. 2-1, the trend from low-speed high-resolution to high-speed low-resolution converters is clear. While Pipeline converters cover a large speed and resolution range, they tend to dominate at the lower-speed higher-resolution end of the spectrum while Flash converters dominate at the higher-speed lower-resolution end. Two-step/Subranging and Folding/Interpolating converters fall in between. The reason for this partitioning will be clearer as each ADC type is discussed below. It is interesting to note that this trend line is fairly continuous despite the different ADC architectures and process types (bipolar, BiCMOS and CMOS) of its constituents.

As noted in [57], this trend line represents the boundary for processing signals in the digital domain, for monolithic silicon-based ADC's, since functions requiring a higher resolution or speed above this trend line must be processed in the analog domain. The movement of this boundary as fabrication technology evolves and improves, can be roughly shown by a re-grouping of the data points by technology (see Table 2-1) as illustrated in Fig. 2-2. Each group encompasses about four years of fabrication technology

TABLE 2-1 Technology Grouping Ranges

	CMOS $L_{min}\,(\mu m)$	BiCMOS $f_T\,(GHz)$	Bipolar $f_T\,(GHz)$
Newest technology	$x \le .7$	$10 \le x$	$11 \le x$
Old technology	$.7 < x \le 1.1$	$6 \le x < 10$	$6 \le x < 11$
Older technology	$1.1 < x$	$x < 6$	$x < 6$

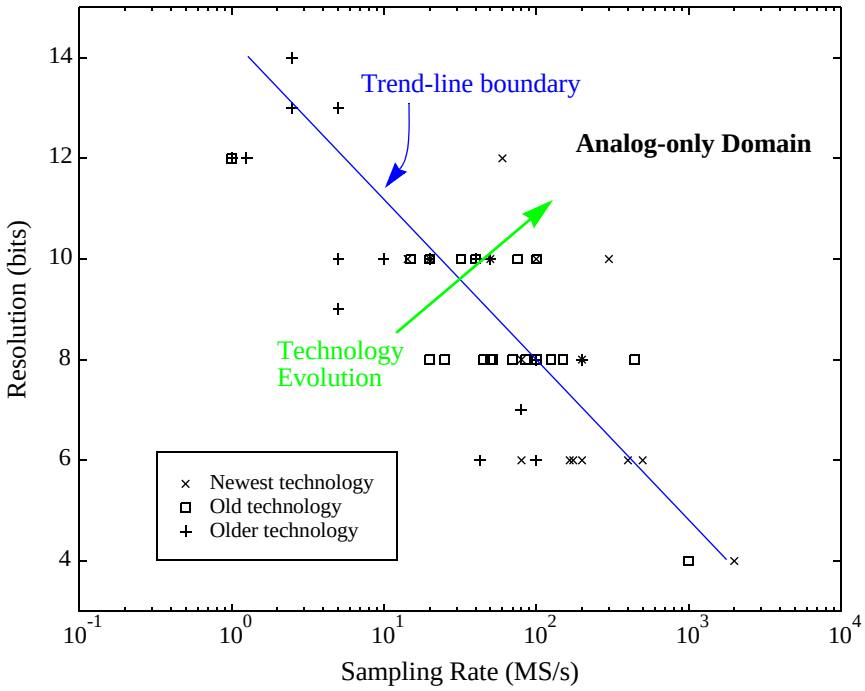


Figure 2-2 ADC performance improvement with technology evolution.

evolution. An estimate from linear regression and graphical analysis of this data, suggests that for a given resolution, the speed (f_S) increases by roughly 50% in four years, alternatively, for a given speed, the resolution increases by roughly half a bit in four years.

2.2 Flash A/D Converters

Each of the high-speed architectures covered in this chapter, has evolved from the Flash ADC architecture. The Flash ADC, which is the simplest and the fastest of all the converters, is shown schematically in Fig. 2-3. In this 3-bit example, the input voltage is compared simultaneously to voltage steps created by the resistor-ladder network. This creates a *thermometer* code at their outputs when the comparators are strobed by the sam-

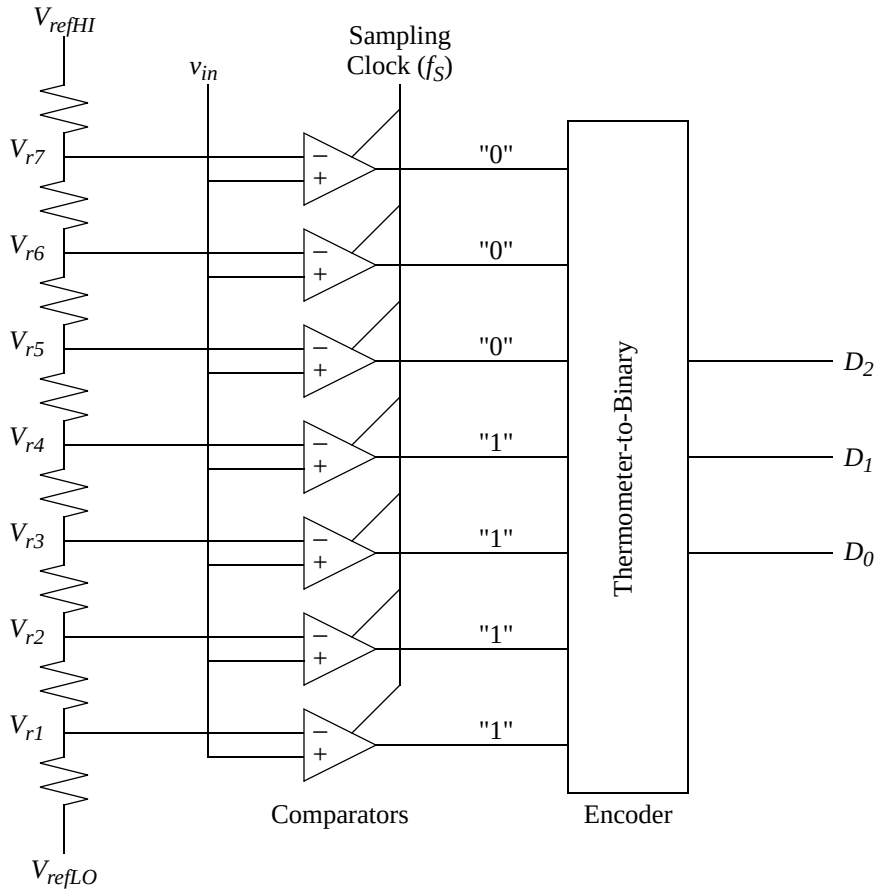


Figure 2-3 Simplified schematic of a Flash ADC (3-bit example).

pling clock. The thermometer code is then converted to a binary code by the encoder's combinational logic. It is the parallel nature of the comparisons that permits the high sampling and conversion rate, but which also limits the resolution of the Flash, as the size of the converter quickly becomes unmanageable due to its exponential growth (a factor of 2 for each additional bit) dependence.

With any parallel input structure, input impedance or loading degrades the input bandwidth and requires a high drive-capable signal source. To remedy this situation, a

sample-and-hold amplifier may be required. However, the power necessary for such an amplifier, with high linearity, wide bandwidth and high drive capability, can be severe. So while there are other problems associated with high resolution Flash converters, such as reference ladder errors, kick-back noise and metastability [60], the dominant reasons that limit its resolution are high power, large silicon area, degradation of SNDR[1] and input loading problems. Thus the Flash converter can be very fast, but remains a low resolution converter.[2]

2.2-1 Power versus resolution

To a first order, the *power* of a Flash converter, with its parallel structure, *increases as a power of 2* for each bit of resolution.[3] Note that, this is an underestimate and over-simplification, that omits any additional power needed for more accurate comparisons (low offset voltage comparators with higher degree of settling) at higher resolutions, and thus, errors on the side of the Flash converter. But it will help keep the complexity of the analysis reasonable.

To evaluate the power-versus-speed trade-off, it will be necessary to partition the discussion by fabrication type; bipolar, BiCMOS and CMOS processes.

1. See Appendix A.
2. Flash implementations often take advantage of being in the low resolution realm; using small, fast comparators with large offsets, fast open-loop amplifier stages with low linearity and front-ends without a sample-and-hold (S/H) amplifier, all of which optimize the power-speed trade-off.
3. Assuming no sample-and-hold amplifier power.

2.2-2 Power versus speed (bipolar process)

In this chapter, the terms *speed* and *sampling rate* (f_S) are used interchangeably. What is meant by *speed* or *sampling rate* is the highest processing rate possible under a given set of operating conditions. So in this context, how power scales with speed is not simply finding the power at a lowering the sampling rate for a given design, but rather, determining the lowest possible power for a given sampling rate and ADC architecture. With this criterion established, the first fabrication type to be considered is the bipolar process.

Most bipolar Flash converters have three main power consuming circuits; a resistive reference ladder, a bank of comparators and digital encoding logic. The typical bipolar comparator is composed of a front-end preamplifier and latch (see Fig. 2-4). The gain of the preamp reduces the input referred offset voltage due to the latch, while the latch's positive feedback increases its gain and, thus, its switching speed.

Pre-amp stage:

In a first order analysis, the switching speed of the simple diff-pair preamp stage is dictated by its exponential output response (via R and C_p[1]). The value of R is dependent on the bias current (I_o) and the small-signal gain, a_o, by the following equation:

$$a_o = g_m \cdot R = \frac{I_c \cdot R}{V_T} = \frac{I_o \cdot R}{2 \cdot V_T} \qquad (2 - 1)$$

1. C_p represents the total (parasitic) capacitance at the output node.

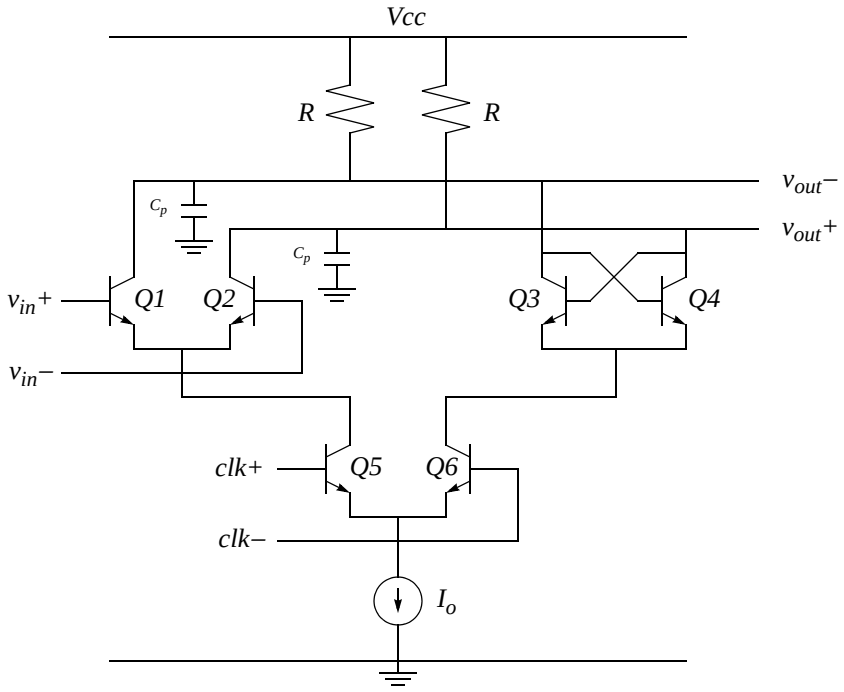


Figure 2-4 Bipolar comparator: preamp (*Q1* & *Q2*) feeds latch (*Q3* & *Q4*).

or rearranging:

$$R = \frac{a_o \cdot 2 \cdot V_T}{I_o} \qquad (2 - 2)$$

where V_T is the thermal voltage (kT/q). The small-signal gain is usually set by the amount of input-referred offset voltage that can be tolerated for a given converter resolution. Equation (2 - 2) exhibits the inverse relationship of R and I_o for a given a_o.

Relating this to sampling rate (f_S), the time it takes the preamp to respond with respect to the sampling period ($T_S = 1/f_S$) depends upon its time constant ($\tau_{pre} = R{\cdot}C_p$) so that the following can be derived (for a desired, fixed a_o):

$$f_S = \frac{1}{T_S} = \frac{K_{A1}}{\tau_{pre}} = \frac{K_{A1}}{R \cdot C_p} = \frac{K_{A2} \cdot I_o}{V_T \cdot C_p} = \frac{K_{A2} \cdot P_A}{V_T \cdot C_p \cdot V_{cc}} \quad (2 - 3)$$

where the *K*'s are proportionality constants[1], V_{cc} is the supply voltage and P_A is the power of the preamp stage. From (2 - 3), *power is* ***proportional*** *to speed* for a given parasitic capacitance and fixed gain (a_o) and supply (V_{cc}). It is interesting to note that (2 - 3) can be rearranged as:

$$P_A = K_{A3} \cdot C_p \cdot V_T \cdot V_{cc} \cdot f_S \quad (2 - 4)$$

which has the same form as the familiar CMOS dynamic power equation, $P = CV^2f$.

Latch stage:

The latching speed is related to the dominant pole, p_{dl} (or equivalently the time constant, $\tau_{dl} = 1/p_{dl}$), and the low frequency gain, a_{ol}, of the cross-coupled amplifiers, *Q3* and *Q4*. The latching speed is measured in terms of the latch *regeneration* time, T_L, which from [60], is:

$$T_L = \frac{\tau_{dl}}{a_{ol} - 1} \ln\left(\frac{v_{outf}}{v_{outi}}\right) \quad (2 - 5)$$

where v_{outi} is the initial value of v_{out}, determined by the preamp at the moment when the sampling clock switches the comparator from the preamp to the latch state, and v_{outf} is the

1. Throughout this chapter constants will be represented by K_α, where α is any alphanumeric subscript.

voltage value needed to qualify as a valid logic level by the encoder that follows the latch. From (2 - 5), for a given $\left(\frac{v_{outf}}{v_{outi}}\right)$, T_L is:

$$T_L \propto \frac{\tau_{dl}}{a_{ol}-1} \cong \frac{1}{a_{ol}\cdot p_{dl}} = \frac{C_p}{g_m} = \frac{1}{2\pi f_u} \qquad (2 - 6)$$

where f_u is the unity-gain frequency and where the last two equalities assume a single-pole response for *Q3* and *Q4* [61]. Thus, ignoring the dependence on $\left(\frac{v_{outf}}{v_{outi}}\right)$ for the moment, latching speed is related to f_u, or equivalently, gain-bandwidth ($a_{ol}\cdot p_{dl}$) in the single-pole case. However, the value of $\left(\frac{v_{outf}}{v_{outi}}\right)$ affects the value of T_L dramatically, especially for very small values of v_{outi} when metastability dominates the response [60]. Despite this complication, v_{outi} is a function of the preamp gain[1], and for a given level of metastability tolerance, these factors are fixed so (2 - 6) is still generally valid for our power derivation.

The latch time has to occur within half the sampling period so the following sampling rate to power relationship can be derived using (2 - 6):

$$\begin{aligned} f_S &= \frac{1}{T_S} = \frac{K_{A4}}{T_L} = \frac{K_{A5}\cdot g_m}{C_p} = \frac{K_{A5}\cdot I_o}{V_T\cdot C_p} \\ &= \frac{K_{A5}\cdot P_A}{V_T\cdot C_p\cdot V_{cc}} \end{aligned} \qquad (2 - 7)$$

Again as in the preamp case, *power is* ***proportional*** *to speed*[2] so by manipulating (2 - 7),

1. Higher preamp gain decreases the metastability problems.
2. It must be noted that since the same bias current powers the preamp and latch, it is more correct to show both the preamp time constant and latch time as a function of the power. Fortunately they both have the same linear dependence on power.

the equation for P_A will again have the same form as the CMOS dynamic power equation.

Resistor reference ladder:

The resistor reference ladder's power is dependent on the value for each resistor section. For reasons of kick-back noise and settling time (see [60]), the time constant of the resistor sections[1] and tap capacitance must be proportional to the desired sampling period. Assuming that the reference voltage and power is being supplied on-chip, the following speed-power relationship can be made:

$$I_{ref} = \frac{V_{ref}}{n \cdot R_{tap}} \tag{2 - 8}$$

$$P_B = I_{ref} \cdot V_{cc} \tag{2 - 9}$$

$$f_S = \frac{1}{T_S} = \frac{K_{B1}}{R_{tap} \cdot C_{tap}} = \frac{K_{B2} \cdot I_{ref}}{V_{ref} \cdot C_{tap}} = \frac{K_{B2} \cdot P_B}{V_{cc} \cdot V_{ref} \cdot C_{tap}} \tag{2 - 10}$$

$$P_B = K_{B3} \cdot C_{tap} \cdot V_{cc} \cdot V_{ref} \cdot f_S \tag{2 - 11}$$

Like the comparator, the result is: *power is **proportional** to speed*.

Digital encoder logic:

Lastly, for the digital encoder logic, emitter-coupled logic (ECL) or some variant is usually employed. Since ECL logic uses diff-pair gain stages, it is very similar to the

1. Though this is not precisely accurate, since the time constant in question is a function of more than just one resistor section, the trend is still correct and sufficient for this analysis.

comparator's preamp gain stage and the speed-power trade-off is the same. So in summary, for a bipolar Flash converter: ***power is proportional to speed***.

2.2-3 Power versus speed (BiCMOS process)

BiCMOS Flash converters use the same comparator topology as its bipolar counterpart, due to the high gain and low input-referred offset characteristics of bipolar devices, and thus achieve the same power-speed relationship. However, the digital encoder logic can use lower power CMOS circuits, when they are fast enough. Whether bipolar ECL or CMOS circuits are used, the same CV^2f-type relationship holds so again *power is proportional to speed*.[1]

2.2-4 Power versus speed (CMOS process)

CMOS Flash converters differ from the previous converters mainly by the type of comparator circuits that are used. This stems from the fact that MOS transistors have poorer transconductance characteristics than bipolar transistors. This leads to larger input-referred offsets and slower latch times of circuit with lower gain (see (2 - 5)). To its advantage, MOS technology's inherent capacitor-sampling capabilities can be used to auto-zero the large offset voltage during a calibration cycle.

CMOS comparators fall into two different categories; auto-zero chopper inverters and master-slave dynamic latches.

1. This is not precisely correct for digital CMOS circuits as will be shown in the next section.

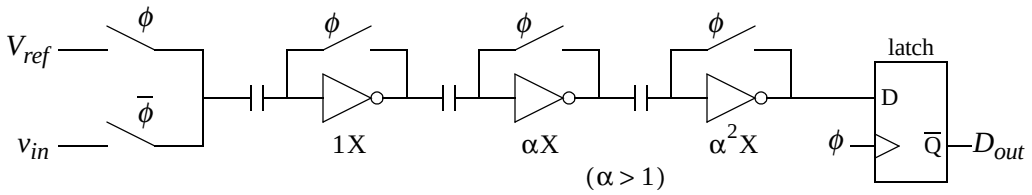


Figure 2-5 Auto-zero chopper inverter CMOS comparator (three inverter example).

Chopper-inverter comparator:

Fig. 2-5 is a simplified schematic of a auto-zero chopper inverter comparator. The preamplifying function is performed by the inverter chain which feeds a D-type flip flop that serves as the latching function. When the clock, ϕ, is high, the CMOS inverter's inputs are shorted to their outputs placing them in a linear amplifier mode where both NMOS and PMOS transistors are turned on and their input/output voltage set precisely to the inverter's switching threshold (auto-zero cycle). At the same time V_{ref} is sampled onto the first (input) capacitor. In the opposite clock phase, v_{in} is connected to the input capacitor and the inverter chain amplifies the difference between V_{ref} and v_{in} and feeds the result to the latch. On the rising edge of ϕ, this result is captured and held by the latch for the subsequent encoding section. The low gain of the MOS transistor is compensated by the cascade of inverter gain stages. Any static offsets due to inverter or device type mismatches are cancelled automatically during the auto-zero cycle.

There are a number of disadvantages with this circuit that are worth mentioning. Foremost, the single-ended nature of this comparator makes the comparison susceptible to power supply and substrate noise, causing *sparkles* in the thermometer code [60]. There is a large current spike during the auto-zero cycle as both NMOS and PMOS transistors are

turned on which causes supply and ground bounce as well as high power dissipation. The cascaded stages, needed to boost the overall gain, create a propagation delay that limits the speed of the converter.

For optimum drive capabilities and propagation characteristics, the inverters are usually scaled up as they go down the chain. In addition to the normal CV^2f dynamic power, the comparator dissipates static power during the auto-zero cycle via the bias current which is set by the size of the inverter's transistors. Since the bias current and the parasitic capacitance both scale with transistor size, only a fractional increase in speed is obtained for a given increment in power. The relationship between the speed of the inverter chain and its power is derived to be (see Appendix A):

$$P_C = \frac{K_{C2}C_m(V_{GS}-V_t)V_{dd}\cdot f_S}{1-K_{C1}f_S\tau_{min}} \qquad (2\text{ - }12)$$

where K_{C1} and K_{C2} are constants, C_m is the inter-stage metal capacitance and τ_{min} is the minimum time constant of a linear (inverter) gain-stage in the limit as the power approaches infinity (see Appendix A for details). From (2 - 12), unlike the bipolar comparator, the *power of a CMOS chopper-inverter comparator is* ***not*** *proportional to speed* but increases asymptotically to infinity as f_S approaches $f_{S(max)}$ given (in Appendix A) by:

$$f_{S(max)} \equiv \frac{1}{K_{C1}\tau_{min}} \qquad (2\text{ - }13)$$

The significance of this power trend can be viewed both positively and negatively. From

the view point of increasing power to obtain faster circuit operation, a CMOS converter would require a disproportionately higher power increment than its bipolar counterpart. Alternatively, if slower circuit operation is allowed by a given ADC architecture, the incremental power would be disproportionately less, potentially resulting in significant power savings.[1] In Appendix A, it is also pointed out that the *break-even* point occurs when the total intrinsic parasitic capacitance equals the total external parasitic capacitance at the output node. Thus comparator speed can be increased somewhat proportionately with power increase (via increases in transistor width) until this *optimum-power*, P_{opt}, breakpoint is reached. However, further power/width increases will result in diminishing speed improvements.

Master-slave dynamic latch comparator:

There are many slight variations to the *master-slave dynamic-latch* comparators that are used in the other category of CMOS Flash converters. Fig. 2-6 is one variant that encompasses the main features of this type of comparator [33][2]. The first stage is a CMOS version of the bipolar comparator of Fig. 2-4 with a preamp (*M1* and *M2*) multiplexed with a (*master*) latch (*M3* and *M4*). This feeds a pipelined *slave* stage which is a dynamic latch made up of *M7–M15*. Due to the lower transconductance of MOS transistors, CMOS converters are not as fast as bipolar or BiCMOS Flash converters. However,

1. The caveat is that once the transistor sizes reach a minimum, the power will also reach a minimum and remain constant, assuming that the auto-zero cycle stays set to half the sampling period, as the sampling frequency decreases. From this point further power savings can be realized by setting the auto-zero period constant thereby reducing its duty cycle as the sampling frequency decreases.
2. The first stage loads are diode-connected transistors that are modeled as a resistor (*R*) for simplicity.

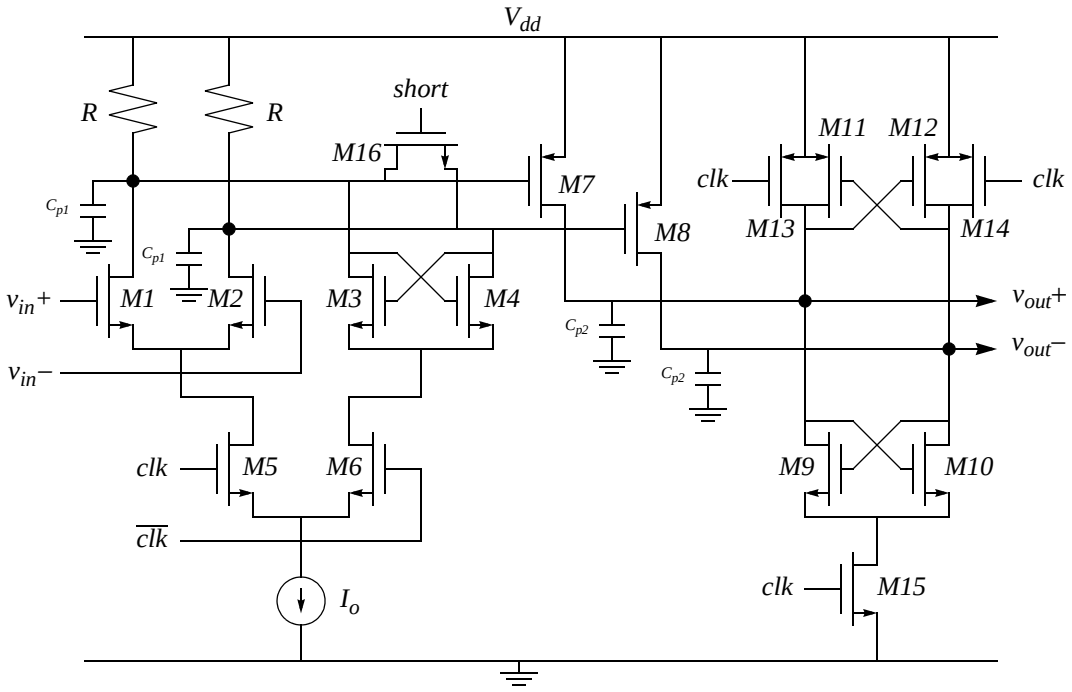


Figure 2-6 CMOS *master-slave* comparator with a dynamic *slave* latch.

by exploiting the advantages of CMOS technology through the use of dynamic latches and digital logic, lower power CMOS converters are possible for lower sampling or higher resolutions.

Another advantage of CMOS technology is the availability of near-ideal switches (such as transistors *M13*, *M14* and *M16* in Fig. 2-6) which can be used to short the output nodes to improve the overload-recovery time. Besides dissipating no static power, the dynamic slave latch generates rail-to-rail outputs for interfacing to the digital CMOS encoder logic. Unfortunately, dynamic latches suffer from large input offset voltages due to the large V_{gs} value as the latch is energized [60] which dictates the need for offset cancellation techniques if medium-to-high resolution is desired.

The analysis of the power-speed relationship for the first stage can proceed in a similar manner to that for the bipolar comparator of Fig. 2-4. Long-channel equations will be used since the short-channel criterion, $(V_{gs}\text{-}V_t) >> (E_{sat} \cdot L)$, is usually not satisfied with analog-type circuits such as the preamp/master-latch stage. A relationship between the load (R), the small-signal gain (a_o) and bias current (I_o) follows the derivation of (2 - 1):

$$a_o = g_m \cdot R = \frac{2I_d \cdot R}{(V_{gs}-V_t)} = \frac{I_o \cdot R}{(V_{gs}-V_t)} \qquad (2 - 14)$$

which is useful if $(V_{gs}-V_t)$ remained constant independent of bias current (and power) change. This case is true only if the current density, $\left(\frac{I_d}{W}\right)$, remains constant, i.e., the transistor size scales with the bias current. When this is not true, the other limiting case to consider is when the transistor size remains constant while the bias current changes. In this case, the alternative formula for g_m can be used in (2 - 14):

$$a_o = g_m \cdot R = R\sqrt{2I_d \cdot k' \cdot \frac{W}{L}} = R\sqrt{I_o \cdot k' \cdot \frac{W}{L}} \qquad (2 - 15)$$

[constant current density case]

Consider the first case of constant current density. From the previous bipolar analysis, the power for both the preamplifier and master latch, as a function of sampling rate, can be shown to be:

$$P_D = K_{D1} \cdot C_{p1} \cdot (V_{gs}-V_t) \cdot V_{dd} \cdot f_S \qquad (2 - 16)$$

where K_{D1} is a constant dependent mainly on upon settling characteristics, i.e., metastabil-

ity tolerance, and small-signal gain, a_o. The parasitic capacitance at the output node, C_{p1}, is made up of the capacitance from the transistors (*M1-M4*, *M7*, *M8* and *M16*) and metal trace capacitance. Since the sizes of *M1-M4* scale with bias current, their contribution to C_{p1} increases with increasing power, i.e., C_{p1} is a function of the power. Thus:

$$C_{p1} = C_{p1f} + C_{p1s} \tag{2 - 17}$$

where C_{p1f} and C_{p1s} are the fixed and scaled portions of C_{p1}, respectively. C_{p1s} includes the power dependence and when combined with (2 - 16), yields the following (from Appendix A):

$$P_D = \frac{K_{D1} \cdot C_{p1f} \cdot (V_{gs} - V_t) \cdot V_{dd} \cdot f_S}{1 - f_S / f_{S(max)}} \tag{2 - 18}$$

where f_{Smax} is given by:

$$f_{S(max)} = \frac{f_T}{4K_{D1}\left(\frac{C_{gd} + C_{db}}{C_{gs} + C_{gd}}\right)} = \frac{f_T}{4K_{D1}\rho} \tag{2 - 19}$$

Notice that (2 - 18) has the same form as the power equation for the chopper-inverter comparator, equation (2 - 12), and is characteristic of circuits whose power varies proportionately with transistor width and, hence, intrinsic parasitic capacitance. Thus, again the same *optimum-power* breakpoint condition exists when C_{p1f} and C_{p1s} are equal. In terms of sampling rate, for frequencies much lower than $f_{S(max)}$, the power is directly proportional to f_S, and as the sampling frequency approaches $f_{S(max)}$, the power increases asymp-

totically to infinity as (2 - 18) indicates.

ρ is somewhat independent of transistor sizing, since both the numerator and denominator approximately scale with the transistor width (*W*), and ranges from .5 to 2 depending on process technology. And since K_{D1} includes the small-signal gain (a_o), $f_{S(max)}$ is an order of magnitude lower than the device f_T. From Appendix A (for long-channel operation):

$$f_T = \frac{\mu(V_{gs}-V_t)}{2\pi L^2} = \frac{\mu}{2\pi L^2}\cdot\sqrt{\frac{2L}{k'}\cdot\frac{I_d}{W}} \qquad (2 - 20)$$

indicating that by increasing the current density, $f_{S(max)}$ can be increased, albeit in a square root manner, supporting the intuition that smaller devices for a given power will improve speed (at the expense of headroom because of the larger ($V_{gs}-V_t$)).

A by-product of maintaining a constant current density while increasing the power, is the increase in input capacitance imposed by the gates of *M1* and *M2*. In addition to increasing the loading on the input signal source, it also presents a larger parasitic capacitance to the reference ladder. This increase in capacitance must be compensated by a reduction in the ladder segments resistance, to limit the settling response of a kick-back impulse, thereby increasing the ladder power. From (2 - 11), the power in the resistor ladder is:

$$P_E = K_E \cdot C_{tap} \cdot V_{dd} \cdot V_{ref} \cdot f_S \tag{2 - 21}$$

and since C_{tap} is a function of the power in the first stage, from (A - 35) in Appendix A:

$$C_{tap} = \frac{K_{E1} P_D}{V_{dd}} \tag{2 - 22}$$

so that the ladder power becomes:

$$\begin{aligned} P_E &= K_{E2} \cdot P_D \cdot V_{ref} \cdot f_S \\ &= \frac{K_{E3} \cdot C_{p1f} \cdot (V_{gs} - V_t) \cdot V_{dd} V_{ref} \cdot f_S^2}{1 - f_S / f_{S(max)}} \end{aligned} \tag{2 - 23}$$

which increases much faster than the power in the first stage with increasing f_S.

[constant transistor size case]

The second case to consider is when the transistor sizes stay constant and only the bias current changes. In this case, C_{p1} is constant and following the previous derivation of (2 - 1) through (2 - 4) using (2 - 15), yields:

$$P_F = K_{F1} \cdot (C_{p1})^2 V_{dd} \cdot (f_S)^2 \tag{2 - 24}$$

where the constant K_{F1} incorporates the fixed small-signal gain, a_o, and transistor size, $\frac{W}{L}$. So for constant transistor sizes the power increases by the square of the sampling rate which can be traced back to the transconductance's square root dependence on bias current (see (2 - 15)).

[dynamic slave latch]

The second stage *slave* latch dissipates only dynamic power when it is energized. From [60], the regeneration time constant, τ_{reg}, for this type of latch is given by:

$$\tau_{reg} \cong \frac{C_{p2}}{g_{m9} + g_{m11}} \qquad (2 - 25)$$

where g_{m9} is the transconductance of *M9* (or *M10*) and g_{m11} is that of *M11* (or *M12*). For this dynamic latch, (V_{gs}-V_t) is a function of V_{dd}, i.e., fixed, and like the chopper-inverter comparator, the power is a function of transistor *W*. Since C_{p2} can be split into a portion that scales with *W* and a portion that is fixed and/or dependent up the load (C_{p2f}), the power-speed relationship will be similar to the chopper-inverter comparator and will have the form:

$$P_G = \frac{K_{G1} C_{p2f} (V_{GS} - V_t) V_{dd} \cdot f_S}{1 - f_S / f_{S(max)}} \qquad (2 - 26)$$

where C_{p2f} is composed of metal interconnect capacitance and the loading of the digital encoder logic that follows.

Power-speed relationship of digital CMOS circuits:

The well-known power-speed relationship of digital CMOS circuits is given by:

$$P_H = C_{eff} V_{dd}^2 \cdot f_S \qquad (2 - 27)$$

where C_{eff} is the total *effective* capacitance that is being switched at f_S. As before, C_{eff} can

be partitioned into a part that scales with transistor *W* and a part that is fixed. Since (V_{gs}-V_t) is equal to (V_{dd}–V_t), the current density is fixed[1] and thus the charging and discharging currents are proportional to the widths of the transistors. For CMOS digital gates, the propagation delay, t_{pd}, is approximately:

$$t_{pd} = \frac{C_o \cdot (V_{dd}/2)}{I_d} \qquad (2 - 28)$$

where C_o is the capacitance load at the output of the gate. The maximum switching frequency, $f_{sw(max)}$, is the inverse of twice the propagation delay , so:

$$f_{sw(max)} = \frac{I_d}{C_o \cdot V_{dd}} \qquad (2 - 29)$$

(2 - 29) indicates that switching speed is directly related to the drain current, and thus also to transistor width. Appendix A shows that with (2 - 27) and these three conditions: (1) capacitance with a fixed portion and a portion that scales with device size, (2) constant device current density, and (3) speed of operation directly proportional to current, the power is:

$$P_H = \frac{C_{feff} V_{dd}^2 \cdot f_S}{1 - f_S / f_{S(max)}} \qquad (2 - 30)$$

which has same form as previous CMOS circuits that also satisfied the above three conditions.

1. See Appendix A.

2.2-5 Flash converter summary

By their parallel nature, the power dissipated by a Flash converter increases by the number of quantization levels desired. So to a first order, power increases exponentially as a power of two for every additional bit of resolution.

The relationship between power dissipation and sampling rate depends upon the process used and the method used to vary the circuit speed. When a bipolar or BiCMOS process is used, a bipolar transistor is used in the comparator. Since the g_m is independent of device size and proportional to bias current, the comparator power is directly proportional to sampling rate. When the encoder is implemented with bipolar transistors, its power is also directly proportional to sampling rate. In the case of a BiCMOS encoder which uses digital CMOS circuits, its power is asymptotic as described below. For a constant tap capacitance, the power for the reference ladder is also proportional to sampling rate.

For CMOS circuits with constant device sizes, the power of a linear-type amplifier comparator (as in Fig. 2-6) increases by the square of the sampling rate. For most other CMOS circuits, where the transistor drive voltage (V_{gs}-V_t) is fixed (either by designing to a fixed current density or by some voltage constraint such as V_{dd}) and the speed of the circuit is proportional to the driving current (either static or dynamic), then power is of the form:

$$P = \frac{C_{fixed} \cdot (V_{gs} - V_t) V_{dd} \cdot f_S}{1 - f_S / f_{S(max)}} \qquad (2 - 31)$$

where C_{fixed} is the fixed extrinsic capacitance (any capacitance that does not scale with transistor width, internally or off-chip) and $f_{S(max)}$ is some factor (4 to 50) lower than the device f_T. When f_S is much lower than $f_{S(max)}$, the power is directly proportional to f_S. This occurs when the intrinsic capacitance (C_{scaled}), which scales with transistor *W*, is much smaller than C_{fixed}. When C_{scaled} is much larger than C_{fixed}, the asymptotic behavior of (2 - 31) takes over as f_S approaches $f_{S(max)}$. In this regime, power increase results in a diminishing increase in f_S and is inefficient from a power utilization standpoint. The transition or break-even point between these two regimes occurs when C_{scaled} equals C_{fixed}. In a power-speed efficiency sense, this is the optimum power point, P_{opt}.

2.3 Folding / Interpolating A/D Converters

All the derivative converters address the per-bit exponential growth problem of Flash converters by partitioning the single-step parallel approach into a *coarse-fine* segmented or a *coarse-fine* multi-step approach. The motivation behind this strategy is the realization that most of the comparators in a Flash converter do not contribute to the conversion result, thus wasting area and power. Only those few, whose comparisons are near the input signal value, determine what the final conversion value will be, so a *coarse-fine* approach decreases area, power and complexity of the non-Flash converter, and/or permits conversions at higher resolutions, at the expense of lower conversion rates.

2.3-1 Folding converters

The *folding* technique started out as a way of "folding" the input signal range into

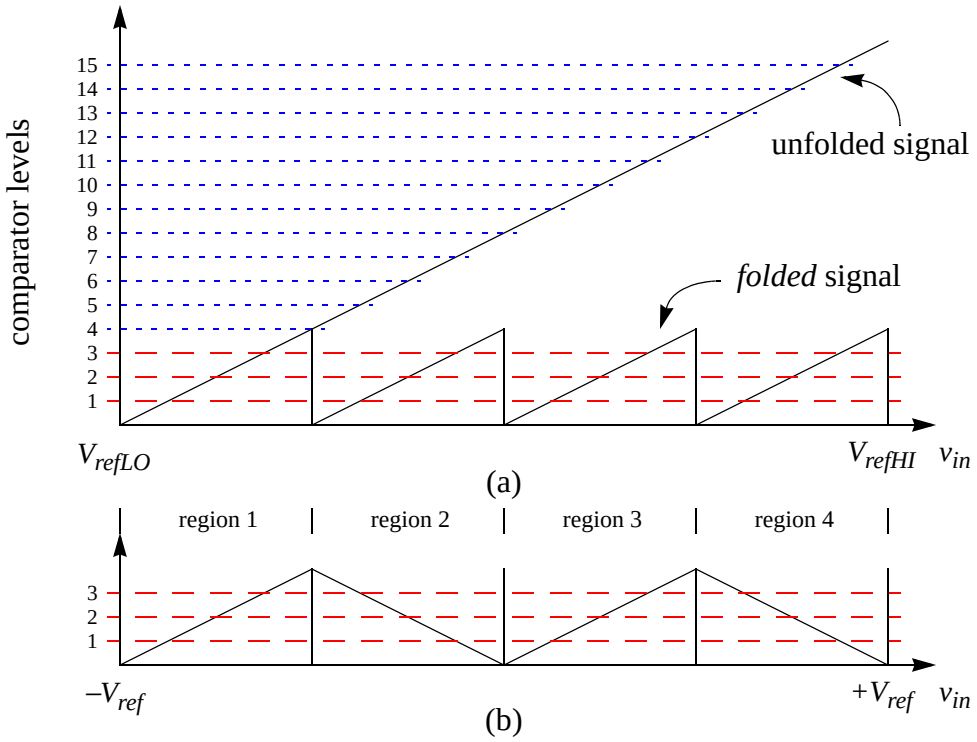


Figure 2-7 Folding transfer function (4X folding example); (a) theoretical, (b) practical (ideal) realization.

smaller repeating ranges much like *modulo* arithmetic (see Fig. 2-7(a)). In the 4-bit ADC example of Fig. 2-7, the input signal is folded into four equal regions. Only 3 comparators are needed to decode any of the 2 least-significant bits (LSB) of a region. Three additional comparators are used to identify in which region the signal lies, to resolve the 2 most-significant bits (MSB). In this way only 6 comparators are needed instead of 15 for a Flash converter. For implementation reasons, the folds for regions 2 and 4 are commonly generated symmetric to regions 1 and 3, respectively, as shown in Fig. 2-7(b).

Problems generating the desired transfer function of Fig. 2-7(b) with adequate bandwidth and linearity led to a modification that relies only on accurate zero-crossing wave-

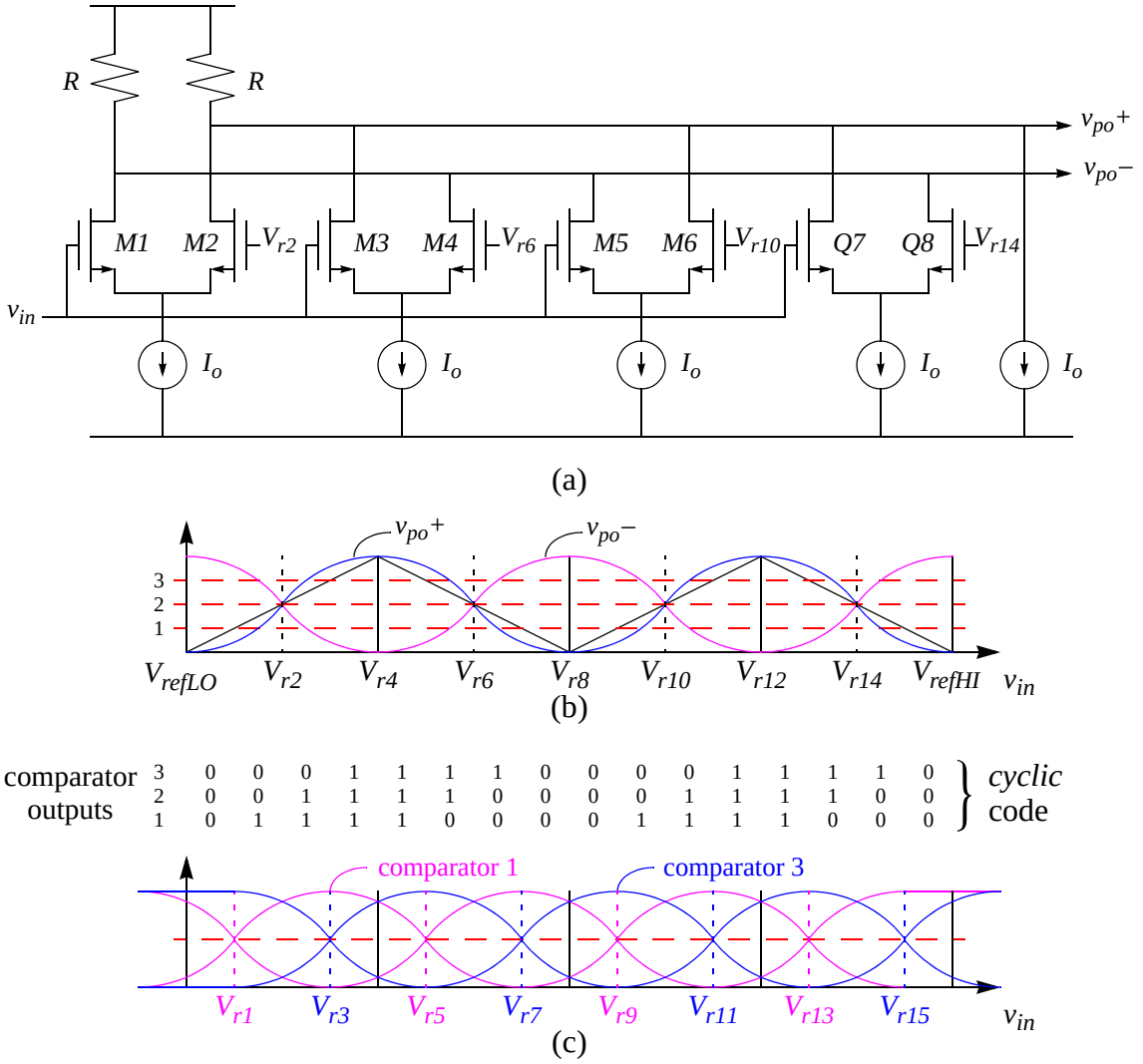


Figure 2-8 A "4 times" CMOS folding block; (a) schematic, (b) transfer characteristics, (c) *cyclic-thermometer* code and transfer curves of folding blocks for comparators 1 and 3.

forms (see [20] and [13]). The most common way of implementing the folding transfer function is by cross-coupling several diff-pair stages as shown in Fig. 2-8. The actual characteristic (v_{po}+) differs from the straight-line ideal characteristic and would cause outputs from comparators 1 and 3 to introduce non-linear errors in the digital output codes. However, the zero-crossing point of the differential output (v_{po}+ minus v_{po}–) exhibits no

error regardless of amplitude or slope variations of the transfer characteristic, i.e., the output of comparator 2 will be insensitive to process and circuit bandwidth variations. By adding two more folding blocks offset one-quarter of a region before and after the previous (comparator 2's) folding block, error-free outputs from comparators 1 and 3, respectively, are possible.[1]

The three MSB comparators would use taps V_{r4}, V_{r8} and V_{r12} to decode the four regions. This modified folding converter would require 12 preamp (diff-pair) stages and 6 comparators as opposed to 4 preamp stages and 6 comparators in the original folding converter or 15 comparators in a full 4-bit Flash converter. From a power perspective, the preamp stages in the folding block consume more power than a comparator at a given speed for two reasons; (1) the output node capacitance of the folding block is about $(n/2)$ times larger due to the n preamp stages in a "n times" folder,[2] and (2) the bandwidth of the folding block needs to be about $(n/2)$ times higher than a comparator from the $(n/2)$ times input signal frequency multiplication by the *folding* action. So assuming that g_m is proportional to I_o implies an increase in I_o by a factor of $(n/2)^2$ for the folder to match the comparator's bandwidth. These two reasons are largely responsible for the lower speed of Folding converters versus its Flash cousin. However Folding converters are generally faster than the other *coarse-fine* converters because the MSB and LSB A/D conversions can be done in parallel rather than in sequence as in the other converters (see

1. Comparator 1's folding block would be connected to reference taps V_{r1}, V_{r5}, V_{r9} and V_{r13} while comparator 3's block would be connected to V_{r3}, V_{r7}, V_{r11} and V_{r15}.
2. If there is enough headroom, a cascode stage can be added to eliminate this problem as in [17].

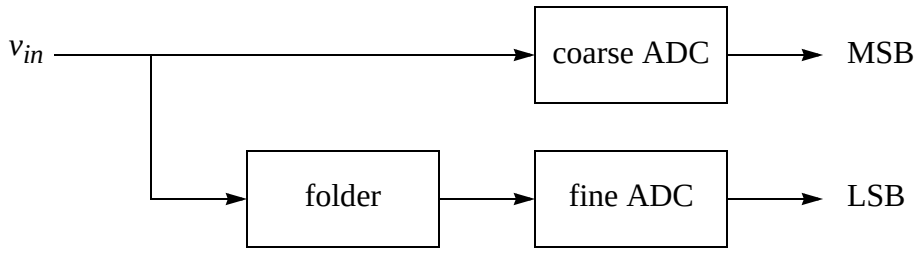


Figure 2-9 Conceptual block diagram of a Folding converter.

Fig. 2-9).

For the sake of comparison, assume that the folder's preamp stage consumes the same amount of power as a comparator, making it possible to see how Folding converter configurations and a Flash converter stack up to one another. Let a preamp or comparator represent one *power load* (PL). Then an "*n* times" folding block represents (*n*+1) power loads. From the prior description of the modified Folding converter, an (*m*+*l*)-bit converter with *m* MSBs and *l* LSBs, requires (2^m–1) MSB comparators (cmps), 2^m times folding per block and (2^l–1) folding blocks and LSB comparators. The 2^m times folding block requires (2^m+1) PLs due to the extra current source needed to set the initial output value. Table 2-2 tabulates the power loads for various configurations.

It is clear from Table 2-2 that the modified Folding converter consumes more power and is slower than a regular Flash converter, so aside from possibly a slight area advantage, there seems to be little reason for using a Folding converter. The original Folding architecture did have an advantage in area and power but it suffered from distortion in generating a precision folding characteristic. Moving to the modified Folding architecture

TABLE 2-2 Power loads

	MSB/LSB	PLs for MSB cmps	PLs per folding block	# of folding blocks	PLs for LSB cmps	Power Load totals
4-bit Flash						15
4-bit Folder	2/2	3	5	3	3	21
"	3/1	7	9	1	1	17
6-bit Flash						63
6-bit Folder	3/3	7	9	7	7	77
"	2/4	3	5	15	15	93
"	4/2	15	17	3	3	69
"	5/1	31	33	1	1	65
8-bit Flash						255
8-bit Folder	4/4	15	17	15	15	285

solved the distortion and precision problems but resulted in a degradation in bandwidth and power performance. To improve the performance of the modified Folding converter, *interpolation* is used to generate half or more of the folded waveforms.

2.3-2 Interpolation technique

In the 4-times Folding converter example of Fig. 2-8, if a differential waveform was generated by taking the average of the differential outputs from the comparator 1 and 3 folding blocks, its zero-crossing would coincide exactly with the differential outputs of the comparator 2 folding block. Since only the zero-crossing point is important, this *interpolated* waveform could replace the comparator 2 folding block waveform, allowing its removal. Although in this case only one block is eliminated, in the case of 8-times folding, 3 of 7 blocks could be removed using interpolation, thus approaching a saving of half the power and area as the number of folds increases. Ignoring the overhead due to interpo-

lation, the PL for the 4-bit 2/2 case would decrease from 21 to 16, the 6-bit 3/3 case from 77 to 50 and the 8-bit 4/4 case from 285 to 166, yielding a more favorable power performance than for a Flash converter. For this reason, most Folding converter use the modified architecture for low-distortion coupled with interpolation to reduce the power and area.

The interpolation technique in its simplest form can be realized by buffering the outputs of two folding blocks and driving a resistor divider to obtain the interpolated signal at its midpoint [60]. An interpolation factor higher than two can be implemented by increasing the number of taps of the resistor divider but some distortion in the zero-crossings of all but the midpoint will have to be tolerated (increasing the non-linearity of the converter).

2.3-3 Power versus resolution

It is hard to generalize or formulate a relationship between power and resolution for Folding/Interpolating converters because of the multiple degrees of freedom (amount of folding, power compensation (for bandwidth multiplication), interpolation, etc.) that exists when changing the converter resolution. However, by examining the above results for the restricted case of an interpolation factor of 2, the power increase per 2-bits increase in resolution is less than 4 (as in a Flash) but more than 3.

2.3-4 Power versus speed

Only a qualitative description will be given for this category of converters. The fact

that the folding blocks use the same diff-pair preamp stage as the comparator in a Flash converter, to a first order, the power-speed relationships should follow those of the Flash converters (bipolar, BiCMOS and CMOS). It appears that Folding architectures can be the fastest converters next to Flash converters but also power consumption of the same order as them.

2.4 Two-step and Subranging A/D Converters

Two-step and subranging converters perform a *coarse* flash conversion to determine the most-significant bits and incorporate this information in the subsequent *fine* (LSB) conversion process. Two-step converters use the MSB information to reconstruct a quantized version of the signal which is subtracted from the original signal, effectively translating it to a fixed sub-range of the reference voltage and bank of LSB comparators (see Fig. 2-10(a)). Subranging converters use the MSB information to select one of 2^{MSB} sub-ranges of the reference voltage via a bank or network of switches effectively translating the LSB comparator bank to the selected sub-range (see Fig. 2-10(b)). Because of the two-step nature of each conversion, the input signal must be sampled and held to process high frequency signals.

2.4-1 Two-step converters

A conceptual (M+L)-bit Two-step Flash requires only 2^M-1 and 2^L-1 comparators for the *coarse* and *fine* Flash sub-converter blocks, respectively, as opposed to $2^{(M+L)}$-1 comparators for a full Flash implementation. Thus Two-step converters can be of higher reso-

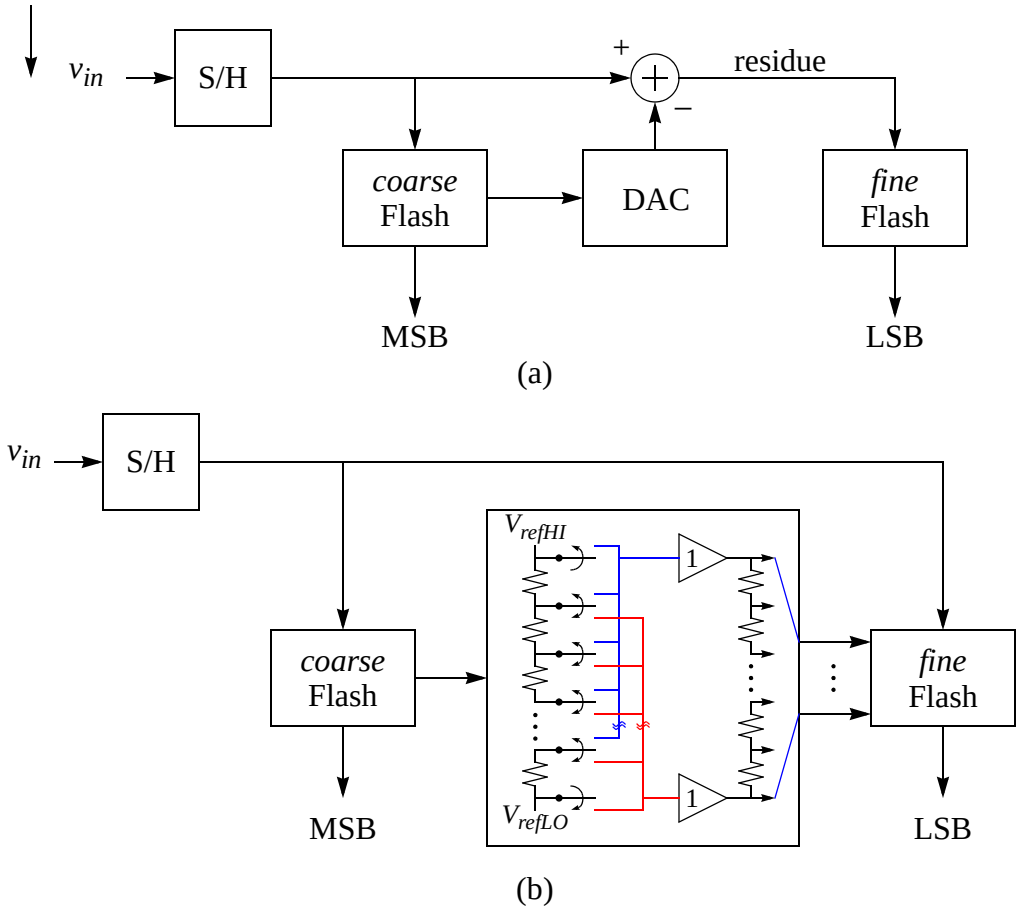


Figure 2-10 Conceptual block diagrams of; (a) Two-step and (b) Subranging Flash architectures.

lution than a full Flash without incurring its area and power penalty but its sequential nature substantially reduces its speed. Within one sample time, the Two-step must perform a *coarse* A/D conversion for the MSBs, reconstruct a MSB quantized analog signal, subtract it from the original signal and perform a *fine* A/D conversion. The duration of this process is further aggravated by the subtraction step which usually requires a closed-loop amplifier to perform a subtraction at the full precision of the converter. Unlike the Flash or Folding converters that can use fast open-loop amplifiers, the speed of

the Two-step (and Pipelined) converters are limited by the need to use low-bandwidth closed-loop amplifiers.

A couple of enhancements can be added to the Two-step architecture which eases its design and improves its overall power and performance.[1] The first enhancement is having extra comparators in the *fine* Flash block that extends its coverage, creating overlaps in adjacent *coarse* (MSB) regions. In this way, offset or settling time errors made by the *coarse* comparators can be detected by the extra comparators and corrected using digital logic. This is called ***digital correction***. By relaxing the offset and settling requirements of the *coarse* comparators, smaller devices and less power can be used. However, the *fine* comparators along with the DAC and subtractor circuits still need the precision required by the full resolution. The second enhancement is to insert a gain stage between the subtractor and *fine* Flash sections. By gaining up the residue signal, the offsets of the *fine* comparators can be relaxed. The gain function can usually be integrated into the subtraction function with little cost in power; however, a proportional penalty in bandwidth is usually incurred as a result of a constant gain-bandwidth product.

2.4-2 Power relationships of a Two-step converter

The relationships between power and resolution or speed for a Two-step follows closely those of the Pipelined converter due to their similarity in architecture; so this discussion is postponed to that section.

1. For a more detailed description see [60] or [58].

2.4-3 Subranging converter

The Subranging converter is an attempt to shorten the conversion time by eliminating the closed-loop operational amplifier (opamp) block needed for the precision subtraction process. Unfortunately the settling time of the switch-matrix network and nodal capacitances compromises this attempt. One implementation [12], uses a dual DAC method to generate the desired sub-range, eliminating the switch-matrix settling problem. Recent Subranging converters have exploited the no-opamp architecture to optimize the power rather than the sampling rate with very good success [9], [10].

Like Two-step converters, *digital correction* can be used to relax the precision of the *coarse* comparators, thus saving power for the high-resolution implementations where these converters are used.

2.4-4 Power versus resolution

The number of comparators increases as $2^{(B/2+1)}$ for an *B*-bit converter. With digital correction the *coarse* comparator's accuracy is relaxed but the *fine* comparators still need accuracy to the full resolution $(1/2^{B})$ of the converter. If offset voltage is the main inaccuracy, then input device area needs to be increased in inverse proportion to the required precision, increasing power of the fine comparators in a square fashion. Alternatively, some offset cancellation technique can be used (with some timing and speed impact), increasing power only by the extra circuitry needed to do the cancellation. The switching-matrix voltage reference will add power similar to that required by the reference ladder for a full Flash converter.

2.4-5 Power versus speed

The *coarse* and *fine* Flash comparators would follow the same relationship as those for the full Flash converter. The power relationship of the buffer amplifiers in the switching matrix that drive the *fine* reference taps should also follow those of the diff-pair stages developed in the full Flash converter section. The power of the resistor ladders in the switching-matrix should also follow the same characteristics as those in the Flash section.

2.5 Pipelined A/D Converters

The Two-step converter's speed limitation is imposed by having to complete all the conversion tasks (*coarse* A/D conversion, D/A conversion, subtraction, possibly amplification and *fine* A/D conversion) within one sampling period. The Pipelined ADC architecture breaks this limitation by adding a S/H block as shown in Fig. 2-11(a) allowing the tasks to be pipelined. Now the results from the coarse and fine conversions are out-of-sync with each other so the coarse result is delayed one clock period[1] through a latch before being digitally corrected and merged with fine conversion result. Thus pipelining trades off latency for throughput (speed). Adding more stages does not affect the throughput so a generalized multi-stage pipeline architecture (Fig. 2-11(b)) can be realized which allows flexibility in the trade-off of bits-resolved-per-stage versus the amount of latency through the converter.

1. With two-phase clocking, it is possible to pipeline at half the clock period, reducing the latency by a factor of 2 (see [63]).

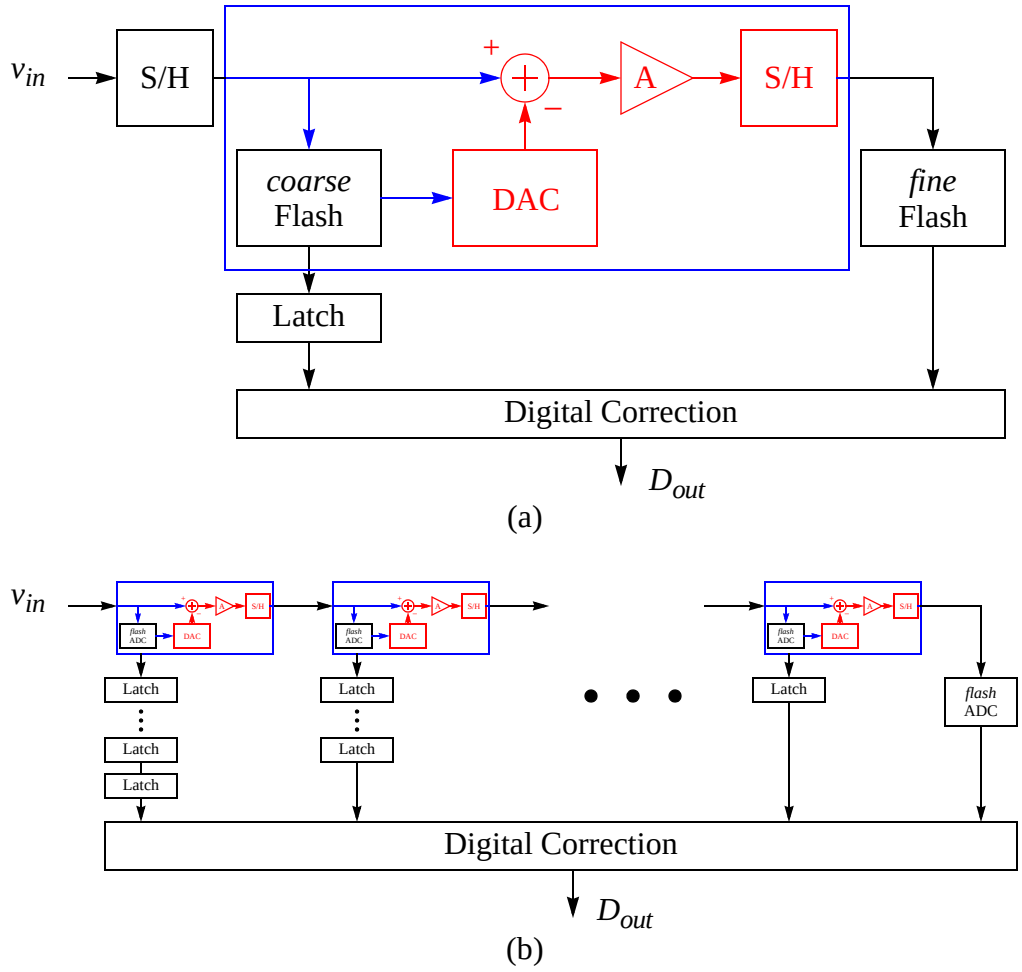


Figure 2-11 Pipelined ADC's; (a) two-stage, (b) multi-stage.

2.5-1 Relaxed accuracy benefit

An advantage of a multi-stage pipelined ADC over the Two-step converter in terms of increasing resolution is the role of the amplifier blocks. As in the Two-step converter, the gain stage relaxes the accuracy requirements of the *fine* comparators. Likewise each gain stage in the Pipelined converter relaxes the requirements progressively as one proceeds down the pipeline. But whereas half the circuits (those in the first stage) have to be accurate to the full precision of the Two-step converter, a smaller percentage of the circuits

(again those in the first stage) have to have full precision for the Pipelined converter. For example, a 10-bit pipelined ADC can be implemented in 5-stages with each stage resolving 2-bits (assume no digital correction). Then the amplifier of each stage would have a gain of four. After the first stage resolves its 2-bits and gains up its residue by four, the rest of the converter (second stage and on) can be viewed as an 8-bit pipelined converter where its accuracy requirements are reduced by a factor of four. From each stage on, the resulting converter needs 2-bits less precision than the preceding one. This can significantly reduce the power requirements by progressively relaxing the design of each succeeding stage. The *first* stage flash ADC, DAC, subtractor, amplifier and S/H blocks have to be accurate to the *full resolution* of the converter but they compose only 20% as opposed to 50% of the circuits in the Two-step.

In terms of power per bit-of-resolution; while adding another stage to a Pipeline to increase the resolution does not impact the speed (throughput) at all, the additional stage cannot be tacked-on at the end of the chain but rather must be added to the *front* of the chain; and as the *new* first stage, it requires more power and precision (at the *new* higher resolution) than the *old* first stage.

2.5-2 Merged DAC, subtraction, amplification and S/H functions

Adding an analog S/H function may seem expensive in terms of power, settling time and accuracy but if MOS technology is used, with its inherent sampled-data capability, the S/H function is included automatically. With CMOS (or BiCMOS) technology the D/A conversion, subtraction, gain and S/H functions can all be merged into one opamp circuit

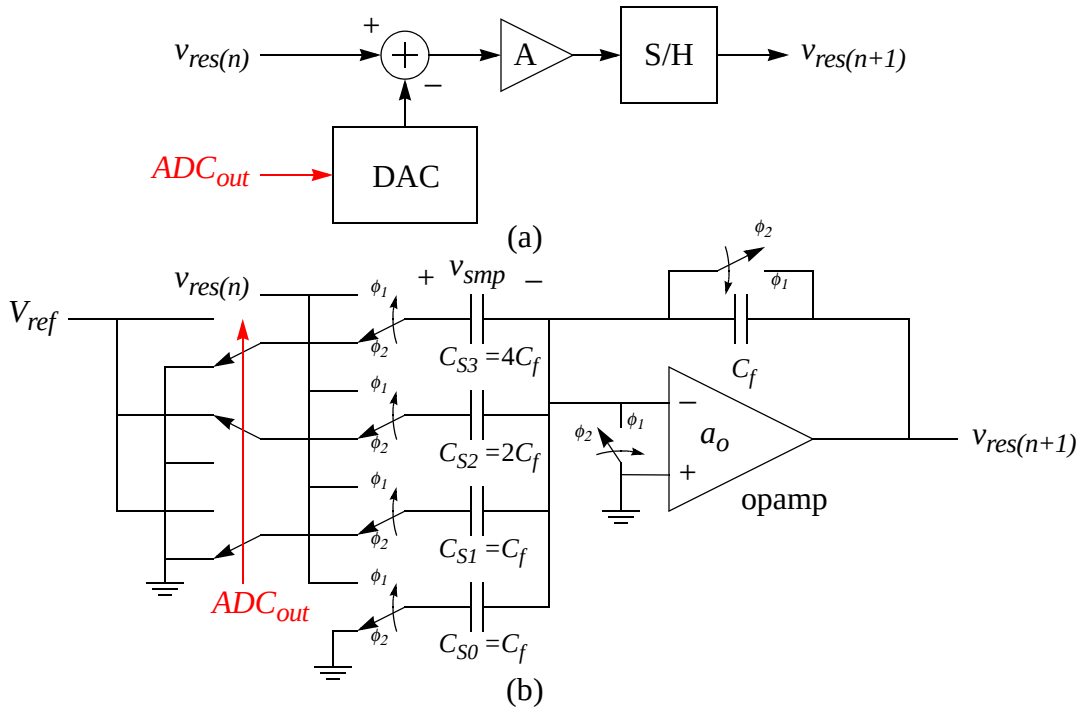


Figure 2-12 CMOS SC integrator implementation of DAC-subtractor-amp-S/H functions (gain-of-eight example); (a) schematic, (b) conceptual single-ended circuit.

as shown in Fig. 2-12. This figure is an example of a single-ended implementation for a switched-capacitor (SC) integrator with a closed-loop gain of eight. For the DAC function it uses a capacitive reference ladder[1] instead of a resistive ladder to avoid static power consumption. Using non-overlapping two-phase timing, the sampling capacitors (C_{S0}-C_{S3}) are charged to the input signal, $v_{res(n)}$, and the feedback capacitor (C_f) is reset (discharged) during the *sampling* phase (ϕ_1). In the *integrate-and-hold* phase (ϕ_2), the sampling capacitors are connected to either ground or V_{ref} depending on the result of the 3-bit flash conversion (ADC_{out}) while the opamp computes the result. It can be shown that after the opamp settles to its final value, the resulting gained-up residue ($v_{res(n+1)}$) is:

1. Capacitive references use a charge-redistribution scheme [62] which dissipates only dynamic power.

$$v_{res(n+1)} = 8\left(v_{smp} - \frac{i}{8}V_{ref}\right) \qquad (i = 0, 1, \ldots, 7) \qquad (2 - 32)$$

where v_{smp} is the sampled value of the input ($v_{res(n)}$) and i is the value of ADC_{out}.

With a two-phase non-overlapping clock system, the *sampling* and *integrate-and-hold* phases of adjacent pipelined stages can be interleaved so that a *integrate-and-hold* phase is being performed on each half cycle (see [63]). Between the flash ADC and opamp functions, the settling time of the opamp integration process occupies most of the *integrate-and-hold* period. Thus the sampling rate is directly related to the bandwidth (settling behavior) of the opamp.

2.5-3 No separate front-end S/H amplifier required

Another benefit of a MOS implementation, worth mentioning, is the inherent sample-and-hold function of the first-stage sampling capacitors (C_S) which eliminates the need for a front-end S/H amplifier. If the first-stage Flash comparators are strobed at the same time that the input signal is sampled (onto C_S), no high frequency degradation in the output *SNR* is observed from lack of a front-end S/H amplifier [63].[1]

2.5-4 Thermal (*kT/C*) noise consideration

In the sampling process, the thermal noise from the MOS transistor switches is cap-

1. In practice, the comparators are strobed slightly after the input sample is taken to ensure a quiet sampling event. With digital correction, this offset in sampling time can be corrected as long as the input signal frequency does not create a sampling error that exceeds the digital correction range. The maximum input signal frequency can be calculated based on the time between input sampling and comparator strobing and is a design parameter that can be set based on the desired bandwidth as was done in [63].

tured onto a sampling capacitor along with the signal. The power of these random noise samples, N_{th}, can be shown to be [64]:

$$N_{th} = \frac{kT}{C} \tag{2 - 33}$$

where k is Boltzmann's constant, T is absolute temperature (Kelvin) and C is the value of the sampling capacitor. At room temperature, the *rms* (square root of N_{th}) value of the kT/C noise for a $1pf$ capacitor is approximately $64\mu V$ so a high-speed[1] converter's *SNR* can begin to degrade as resolutions increase above 8-bits. Thus with the previous discussion on per-stage accuracy, the first stage of a pipelined converter would be most sensitive to this noise. For example, [63] shows that for a 10-bit pipelined converter, the sampling capacitor sizes of first few stages are dictated by the kT/C noise consideration. The remainder of the stages are able to use minimum capacitor sizes due to relaxed accuracy requirements. [63] also shows that in kT/C limited cases, to a first order, the power of a switched-capacitor integrator (as in Fig. 2-12) is proportional to the sampling capacitor size which agrees with the results of (2 - 34) below. Since for each additional bit of resolution (factor of two in precision), the sampling capacitor needs to quadruple, the power of each additional *first* stage also quadruples with respect to the last *first* stage added to the front of the pipeline.

2.5-5 Power versus resolution

The power of a Pipelined converter as a function of resolution will vary depending

1. Therefore small sampling capacitors.

on the number of stages and the number of bits allocated per stage. For convenience and to make this task tenable, a 1*bit*-per-stage architecture will be assumed. As pointed out previously, to increase the resolution of a pipelined converter, an additional stage is added at the front of the chain. So for a 1*bit*-per-stage converter, each preceding stage adds one bit of resolution so the increase in power is due to the power of the additional stage.

In the kT/C limited regime (high-order bit stages), the power of a given stage is four times the power of the following stage. When not in the kT/C limited regime, each preceding stage still requires more power for its SC integrator to settle to a higher accuracy required by the additional bit of resolution, though not by a factor of four.[1] Making some simplifying assumptions, the per-stage power as a function of converter resolution, capacitor sizes and sampling frequency covering both regimes can be derived based on the SC integrator settling-time behavior (see Fig. 2-13). This derivation is done in

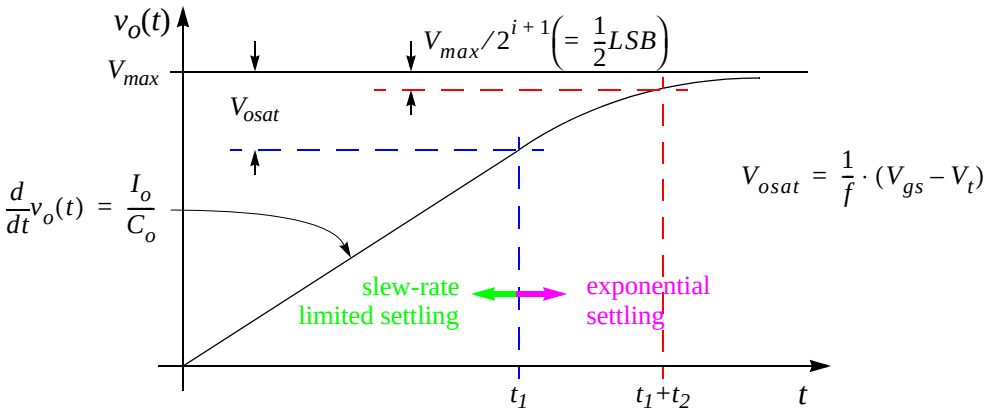


Figure 2-13 Settling behavior of SC integrator.

1. In the *kT/C* limited regime the higher precision does cause an additional power increase per each preceding stage but it is over-shadowed by the four-fold increase due to the sampling capacitors.

Appendix A and yields the following result for the power of a stage, P_i, with a required precision of *i*-bits:

$$P_i = 2C_x V_{dd}(V_{gs}-V_t)f_S \cdot \left((i+1)\ln(2) + \frac{V_{max}}{V_{osat}} - 1 - \ln\left(\frac{V_{max}}{V_{osat}}\right)\right) \quad (2\text{ - }34)$$
$$= P_x \cdot [(i+1)\ln(2) + \alpha]$$

where V_{max} is the worst-case maximum voltage excursion of the output, V_{osat} is the voltage range within V_{max} in which single-pole (exponential) settling occurs (see Fig. 2-13), C_x is given by:

$$C_x = C_S\left(1 + \frac{C_L}{C_f}\left(1 + \frac{1}{|A_I|}\right)\right) + C_{op}\left(1 + \frac{C_L}{C_f}\right) \quad (2\text{ - }35)$$

and α is given by:

$$\alpha = \frac{V_{max}}{V_{osat}} - 1 - \ln\left(\frac{V_{max}}{V_{osat}}\right) \quad (2\text{ - }36)$$

A_I is the closed-loop (inverting) gain of the SC integrator and is equal to $(-C_S/C_f)$. The value for α ranges typically from 0 (when V_{osat} equals V_{max}, i.e., no slew rate limiting) to 3 (for $V_{dsat} = 175mV$, $f = 0.5$, $V_{max} = 2V$).

The SC integrator model used for the above derivation is shown in Fig. 2-14. Equation (2 - 34) shows that P_i varies linearly with bit resolution (*i*) due to the need for more complete settling by each preceding stage. It also incorporates the direct dependence on

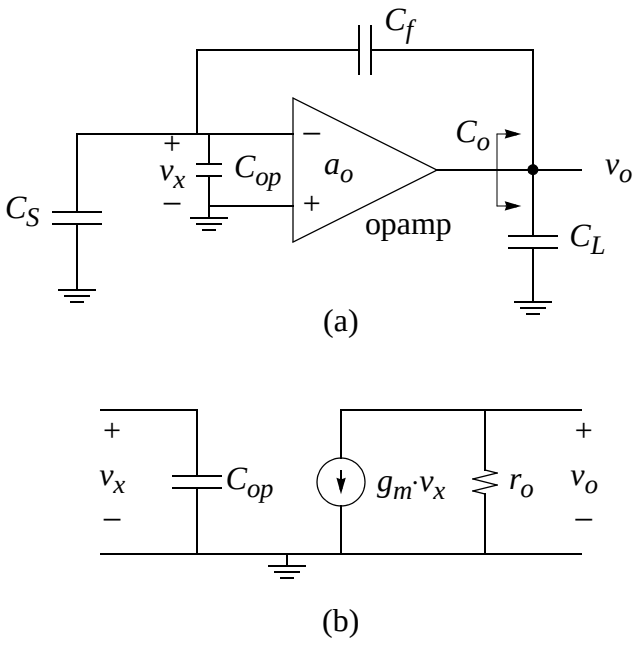


Figure 2-14 Settling-time calculation circuits; (a) simplified DAC-sub-gain-S/H diagram, (b) small-signal model of opamp.

capacitance, C_x, which gives rise to the kT/C power trend, i.e., quadrupling P_i per additional bit of resolution (via C_S & C_f). (Note that the ratio C_L/C_f remains constant, independent of i, which is also generally true for V_{max}, $(V_{gs}-V_t)$ and V_{osat} for a given design.)

Pipelined converters can be designed where all stages are identical so that the per-stage power is identical, then for a 1*bit*-per-stage design, the power increases *linearly* for each additional bit of resolution. In this case, (2 - 34) can be used to calculate the power of the first stage and all following stages since the first stage design (meeting the most stringent requirements) can be used for all subsequent stages. Though this is efficient in terms of design time, it can be very inefficient in terms of power since it does not leverage the relaxed accuracy requirement of the latter stages. Alternately the power can be optimized by designing each stage only as accurate as needed. When this is the case,

the per-stage power is not constant but follows (2 - 34) as a function of the needed accuracy, specified by i (number of bits of settling accuracy) and C_x (indirectly the size of the sampling capacitor needed to keep the kT/C noise low enough for i-bits of accuracy). Then the power for all N stages of this optimized converter, P_N, is:

$$P_N = \sum_{1}^{N} P_i = \sum_{1}^{N} (P_x \cdot [(i+1)\ln(2) + \alpha]) \qquad (2 - 37)$$

2.6 Power Trend Summary

To a first order, converter power is directly proportional to sampling rate (f_S) when f_S is much lower than the device technology transition (*cutoff*) frequency (f_T). However power increases asymptotically to infinity as the design pushes f_S close to f_T. In the case of *constant current-density* designs, there is an optimum power point (P_{opt}), when total intrinsic capacitance[1] equals total extrinsic capacitance, beyond which power increases yield diminishing returns in speed improvements.

Power dependence on converter resolution is not as straight-forward as its dependence on f_S because converter architecture also varies with resolution. Flash converters dominate at the high-speed low-resolution end of the spectrum while Pipelined converters are usually employed at the low-speed high-resolution end. If one ignores the other mid-range converters, this architectural dependence can be factored into a disjoint resolution-versus-power function. Thus for converters of less than 8bits, the power dependence

1. Intrinsic capacitances is device capacitances that scale with transistor size and hence also bias current.

would be proportional to the number of quantization levels needed for a given number of bits (B), i.e., 2^B, if a Flash converter is used. And for converters greater than or equal to 8bits, the dependence would be a nonlinear function following the per-stage power function given by (2 - 34) if a 1-bit-per-stage Pipelined architecture is used.

2.7 ADC Power in High-speed Application Examples

Based on the results of the preceding sections, a rough quantitative comparison of system power as a function of architectural choices in the implementation of three high-speed signal processing applications[1] can be carried out. The three applications are: (1) a high-speed local area network (LAN) system (100BaseTX), (2) a direct-sequence spread-spectrum code-division multiple access (DS-CDMA) system and (3) a radio-frequency (RF) wireless front-end system as commonly found in cellular phones.

2.7-1 High-speed LAN application

The first application is based on an 100BaseTX CMOS transceiver design as described in [65]. This LAN system is designed to enable computer network connectivity at 100*Mb/s* via data-grade "category 5" (cat 5) unshielded twisted-pair telephone wiring as found in most newly wired office buildings and schools. Due to the lack of wire shielding, the transmitted signal must be filtered and wave-shaped to transform the wideband signal spectrum to a narrower bandwidth to meet FCC emissions requirements. Prior to

1. In all three examples, only the receive (demodulating) section is considered, since this is the section that contains the ADC function.

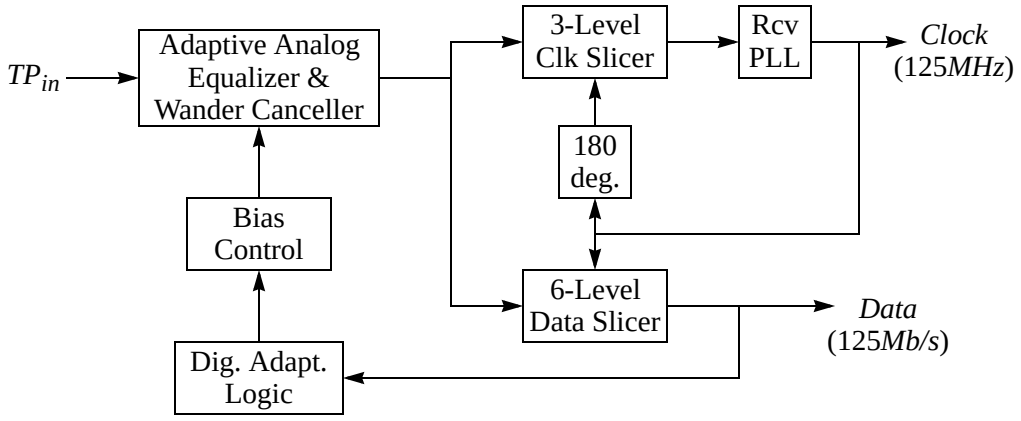


Figure 2-15 Simplified block diagram of 100BaseTX receiver (redrawn from [65]).

wave-shaping, the transmitted signal is encoded using a 4bits-to-5bits (4b5b) NRZI scheme to create extra symbols that can be used for control and error management while boosting the data rate to 125*Mb/s*. At the receive end of the twisted-pair cable, the signal recovery circuits must, (1) compensate for distortion caused by the channel (twisted-pair cable) using an adaptive equalizer, (2) adjust the DC level shift due to AC-coupled asymmetric data patterns (*baseline wander* compensation), (3) restore an adequate signal level via automatic gain control (AGC), (4) phase-lock to recover timing information, (5) detect the multi-level (MLT3) wave-shaped signal and (6) reverse the 4b5b encoding. Fig. 2-15 shows a simplified block diagram of the 100BaseTX receiver of [65].

Instead of digitizing the twisted-pair input (TP_{In}) directly, the signal is equalized and *wander* compensated in the analog domain. Next it is converted into a digital signal using a six-level data slicer[1] which requires five comparators. To phase-lock for clock recovery,

1. Equivalent to the front-end of a six level Flash converter.

a three level clock slicer (two comparators) is used that is strobed midway between the data slicer comparisons. From [65], to accomplish the same task digitally would require at least a 6-bit ADC sampling TP_{in} at 125*MHz* or higher.[1]

Thus in this example, by digitizing **later** in the signal processing stream, *the ADC power ratio is **64** comparators (6-bit Flash) to **7** comparators which is more than a factor of **9***. The dynamic range of the signal is reduced by the analog processing which eases the requirements of the digitizing task thus requiring less circuitry (comparators), and hence, less power.

2.7-2 Baseband DS-CDMA application (all analog)

The second high-speed example is a baseband DS-CDMA system that is part of the design of the *InfoPad* research project at UC Berkeley[2] (see Chapter 3 for a more detailed discussion of a DS-CDMA system). This baseband system modulates the data with a high-frequency *pseudo-random noise* (PN) code, which "spreads" the baseband data spectrum across a larger bandwidth, prior to RF up-conversion and transmission. This wider *spread spectrum* makes the information more tolerant to spectrum fading and interference from multipath and narrowband signals when it is transmitted into the air. The bandwidth expansion factor is determined by a design parameter called the *spread factor* (F_S). For example, the 1*MHz* data bandwidth of the InfoPad system is expanded to a 64*MHz spread*

1. To achieve clock and data recovery would typically require sampling the 100BaseTX signal at twice the data rate, or using two interleaved ADC's sampling at the data rate, offset by half a sample period as in the clock and data slicers described above.
2. For more information go to: http://infopad.EECS.Berkeley.EDU/infopad/

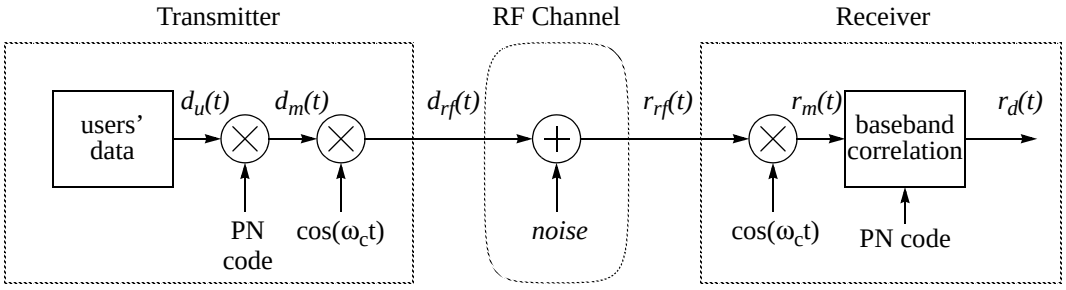


Figure 2-16 DS-CDMA simplified system block diagram.

spectrum bandwidth by using an F_S of 64. A simplified system block diagram is shown in Fig. 2-16.

The baseband data recovery (after RF carrier demodulation) is accomplished using correlation of the input signal with the *same* PN code that is used in the transmitter. PN code synchronization is accomplished by using a delay-locked loop (DLL) which employs two additional correlators. Fig. 2-17 shows a simplified diagram of the baseband correla-

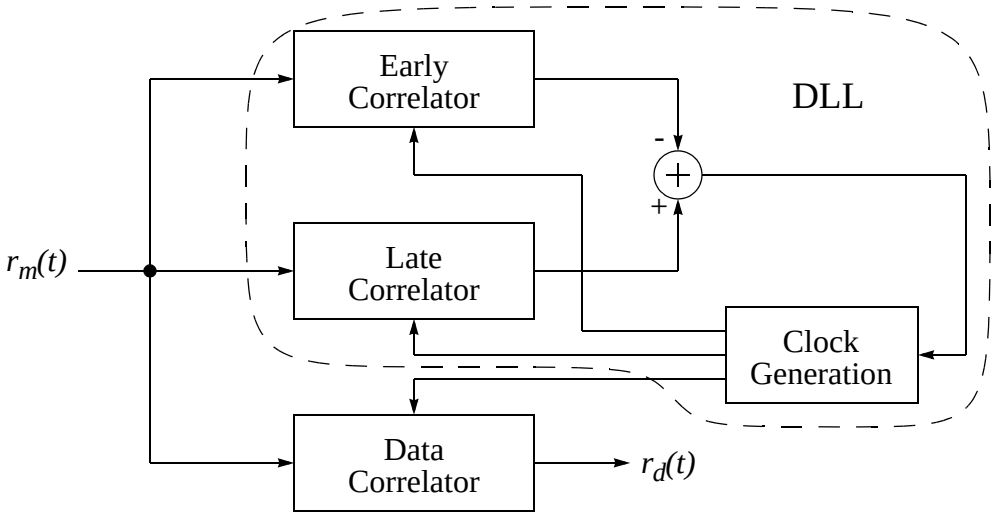


Figure 2-17 Baseband correlation block diagram with delay-locked loop (DLL).

tion block. In the InfoPad system, each correlator outputs one result after processing 64 input samples,[1] therefore, from input to output, the data rate experiences a decimation by a factor of 64 which results in a $1MHz$ output bandwidth.

Now a comparison can be made between a *pre-correlation* ADC which digitizes the input baseband signal, $r_m(t)$, and a *post-correlation* ADC which digitizes the output signal, $r_d(t)$. Since the input bandwidth is 64 times larger than the output bandwidth, the *pre-correlation* ADC would have to sample 64 times faster than the *post*-ADC. In addition, since the timing information for proper operation of the DLL requires half period sampling, either two *pre*-ADC's are needed or one sampling 128 times faster than the *post*-ADC is needed. In either case, a 128 to 1 power ratio of *pre*- to *post*-ADC results when only sampling rate is considered. In terms of resolution, the dynamic range requirements of the input requires 4 or more bits of resolution [66], whereas at the output, only 1 bit of resolution is needed to recover the data. These low resolutions suggest using Flash converters to do the ADC function, resulting in a 16 to 1 power ratio when only taking resolution into consideration. Combining both the sampling rate and resolution ratios results in a **2048-to-1** power ratio between a *pre*-ADC and *post*-ADC in this example.

As before, signal processing prior to analog-to-digital conversion appears to favor a lower power architecture.

1. Analog sampled-data processing (discrete-time, continuous amplitude) is assumed for the analog-domain correlation in the *post*-ADC case.

2.7-3 Alternative realization of a DS-CDMA application (digital synchronization)

Before moving on the to the last example, a comparison of another *post*-ADC realization of the above system is possible. This realization stems from the current trend to maximize digital processing when it is desirable (in terms of performance and flexibility) as well as practical and economical (in terms of time, area and power). In the diagram of Fig. 2-17, the clock generation block which controls the tracking and acquisition behavior of the DLL, is responsible for the overall performance of the recovery process. Bit errors are generally not caused by errors in the data correlation (detection) block but by loss of synchronization caused by large tracking errors that are functions of the signal strength of the tracking signals (Early and Late correlations) and type of algorithms used in the clock generation block. For these reasons, digital implementation of the clock generation block is sometimes preferable (as shown in Fig. 2-18).

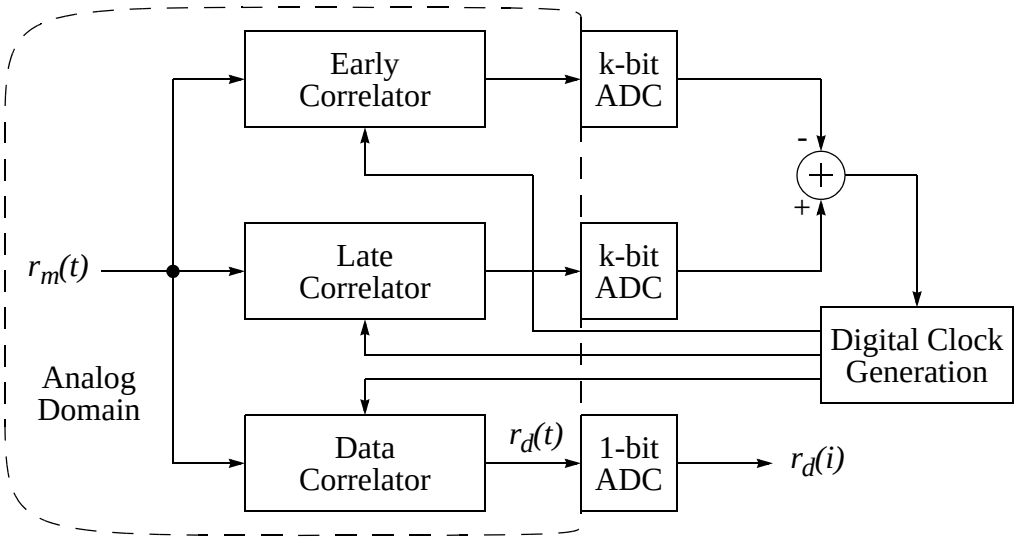


Figure 2-18 *Post-correlation* ADC block diagram with digital clock generation.

The resolution of the *post*-tracking ADC's, has to be better than *pre*-tracking ADC's by *i* bits, where *i* is:

$$\begin{aligned} i &= \log_2(\sqrt{F_S}) \\ &= \frac{1}{2} \cdot \log_2(F_S) \end{aligned} \qquad (2 - 38)$$

in order to preserve the *SNR* improvement due to the processing gain (see Chapter 3). This translates to 3 bits for an F_S of 64. Thus 4 bits of resolution for the *pre*-ADC's implies *post*-ADC's with 7 bits of resolution which falls within the regime of using Flash converters. Assuming two *pre*-ADC's, Table 2-3 summarizes the characteristics for the *pre*- and *post*-ADC's (assuming two ADC's, operating at 64*MS/s* offset by half a sample period for tracking purposes, are used in the *pre-correlation* case). From this table, the

TABLE 2-3 ADC Characteristics for DS-CDMA System

	Pre ADC's	*Post* ADC's	
		tracking	data
Number required	2	2	1
Sampling rate (f_S)	64*MS/s*	1*MS/s*	1*MS/s*
Resolution	4*bits*	7*bits*	1*bits*

post-ADC comparators will consume less power than the *pre*-ADC comparators by a factor of 64. However, there will be roughly 8 times more *post*-ADC comparators resulting in a *pre*-versus-*post* power ratio of about **8-to-1**. So although the power difference between the *pre*- and *post*-ADC's is not as large as in the all-analog case, in this more realistic case the difference is still about a factor of 8.

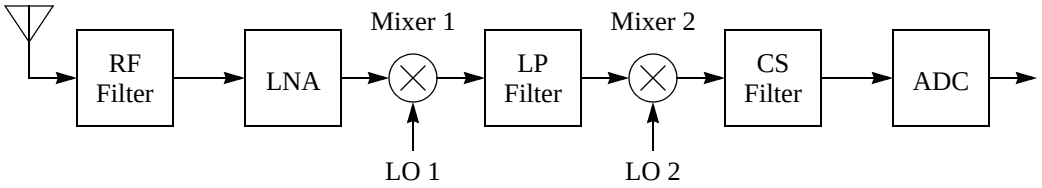


Figure 2-19 Wide-band IF with double conversion receiver architecture.

2.7-4 RF wireless front-end application

The last example is based on the *front-end* chip of an RF receiver for the Digital Enhanced Cordless Telecommunications (DECT) specifications [67]. This receiver does not use the typical superheterodyne architecture; instead it uses an architecture that is more suitable for monolithic integration in CMOS technology called *wide-band IF with double conversion* (WIFDC). A block diagram of the WIFDC architecture is shown in Fig. 2-19. The RF filter is the only component that is hard to integrate. The low-noise amplifier (LNA) gains up the incoming signal to decrease the input-referred noise contributed by Mixer 1 and subsequent blocks. The combination of Mixer 1 and 2 uses complex signal processing to reject the *image* signal, i.e., with image-reject mixing. The low-pass (LP) filter attenuates all energy higher than the intermediate frequency (IF) channel information and the channel-select (CS) filter attenuates all adjacent channels thereby passing only the desired channel information to the ADC (see [67]).

The desired channel information (700*kHz* bandwidth) is translated down to baseband after demodulation by Mixer 2. The CS filter must be a high-order high-Q low-pass filter in order to attenuate the adjacent channels. The 700*kHz* bandwidth is modest enough to allow using analog switched-capacitor (SC) or digital filtering thus providing stable

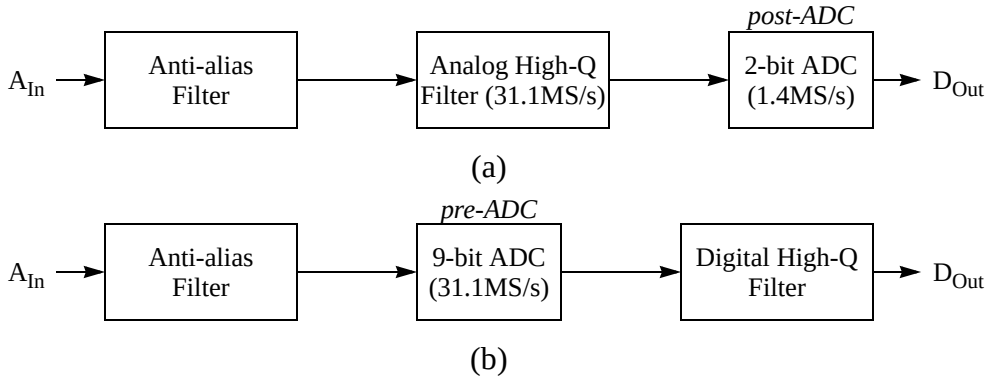


Figure 2-20 Two possible realizations of the CS filter; (a) analog filter, (b) digital filter.

high-Q filter functions.[1] Fig. 2-20 shows how the CS filtering function can be implemented in the analog[2] or digital domains.

To relax the filtering requirements of the continuous-time anti-alias filters, the subsequent blocks oversample the analog signal by about a factor of 22 in [67] ($f_S = 31.1MS/s$). To obtain at least a *SNR* of 10.3*dB* after the CS filters, the input dynamic range of these blocks must be at least 50.3*dB* to accommodate the highest adjacent channel energy allowed by the DECT standard (40*dB* over the minimum input signal). This translates to at least 9*bits* of resolution required of the *pre-ADC* in Fig. 2-20(b). After analog channel selection, the *post-ADC* needs a minimum dynamic range of 10.3*dB* (which translates to at least 2*bits*[3] of resolution) and Nyquist sampling rate of 1.4*MS/s*[4] (700*kHz* signal band-

1. It is more difficult to realize high-Q filter functions using integrated continuous-time analog filters.
2. This approach was used in [67] and is described in [68].
3. Of course in an actual realization, extra bits of resolution would be added for system margin but are not essential for this exercise.
4. In [67] and [68], the pipelined ADC has 10*bits* of resolution and samples at 10.37*MS/s* to allow subsequent digital processing the ability to correct for timing errors due to incoherent demodulation. This analysis will assume coherent demodulation to avoid the extra complication.

width).

At these sampling rates, an ADC power comparison can be made assuming a pipelined ADC architecture. The power ratio of the *pre-* to the *post-*ADC as a function only of sampling rate is 31.1 to 1.4 or **22 to 1**. A conservative estimate of the power ratio as a function of resolution can be derived using (2 - 37). To make the derivation tenable, the power trend attributed to kT/C noise considerations is ignored making the results more modest. With this assumption, P_x is constant and the following can be derived (see Appendix A):

$$P_N = P_x \cdot \frac{\ln(2)}{2}\left[N^2 + N\left(3 + \frac{2\alpha}{\ln(2)}\right)\right] \qquad (2 - 39)$$

So the power ratio of a 9*bit* converter versus a 2*bit* converter is:

$$\frac{P_9}{P_2} = \frac{\left[9^2 + 9\left(3 + \frac{2\alpha}{\ln(2)}\right)\right]}{\left[2^2 + 2\left(3 + \frac{2\alpha}{\ln(2)}\right)\right]} \qquad (2 - 40)$$

For $\alpha = 0$, this ratio is 10.8 and for $\alpha = 3$, this ratio is 6.8. Combining the speed and resolution trends results in a power ratio of at least **149 to 1**, again favoring analog processing prior to digital conversion.

2.7-5 Summary

From these examples, the power ratio of a *pre-processing* ADC versus a *post-processing* ADC can range from 2048-to-1 to 8-to-1. Assuming that analog signal processing

and digital signal processing dissipates about the same power to perform the same function implies that there may be an overall system power minimization by delaying the analog-to-digital conversion for high-speed applications. To validate these ideas, a prototype DS-CDMA baseband recovery system similar to the one described in Section 2.7-3 has been implemented and is subsequently described. But before describing the prototype, some background information on DS-CDMA systems is presented next.

2.8 References

[1]. T. Shimizu, M. Hotta, K. Maio and S. Ueda, "A 10-bit 20-MHz Two-Step Parallel A/D Converter with Internal S/H," *IEEE J. Solid-State Circuits*, vol. 24, no. 1, pp. 13-20, February 1989.

[2]. R. Petschacher, B. Zojer, B. Astegher, H. Jessner and A. Lechner, "A 10-b 75-MSPS Subranging A/D Converter with Integrated Sample and Hold," *IEEE J. Solid-State Circuits*, vol. 25, no. 6, pp. 1339-1346, December 1990.

[3]. D. A. Mercer, "A 12-b 750-ns Subranging A/D Converter with Self-Correcting S/H," *IEEE J. Solid-State Circuits*, vol. 26, no. 12, pp. 1790-1799, December 1991.

[4]. T. Miki, H. Kouno, T. Kumamoto, Y. Kinoshita, T. Igarashi and K. Okada, "A 10-b 50 MS/s 500-mW A/D Converter Using a Differential-Voltage Subconverter," *IEEE J. Solid-State Circuits*, vol. 29, no. 4, pp. 516-522, April 1994.

[5]. D. A. Kerth, N. S. Sooch and E. J. Swanson, "A 12-bit 1-MHz Two-Step Flash ADC," *IEEE J. Solid-State Circuits*, vol. 24, no. 2, pp. 250-255, April 1989.

[6]. M. Yotsuyanagi, H. Hasegawa, M. Yamaguchi, M. Ishida and K. Sone, "A 2 V, 10 b, 20 Msample/s Mixed-Mode Subranging CMOS A/D Converter," *IEEE J. Solid-State Circuits*, vol. 30, no. 12, pp. 1533-1537, December 1995.

[7]. B. S. Song, S. H. Lee and M. Tompsett, "A 10-b 15-MHz CMOS Recycling Two-Step A/D Converter," *IEEE J. Solid-State Circuits*, vol. 25, no. 6, pp. 1328-1338, December 1990.

[8]. M. K. Mayes, S. W. Chin and L. L. Stoian, "A Low-Power 1 MHz, 25 mW 12-Bit Time-Interleaved Analog-to-Digital Converter," *IEEE J. Solid-State Circuits*, vol. 31, no. 2, pp. 169-178, February 1996.

[9]. S. Hosotani, T. Miki, A. Maeda and N. Yazawa, "An 8-bit 20-MS/s CMOS A/D Converter with 50-mW Power Consumption," *IEEE J. Solid-State Circuits*, vol. 25, no. 1, pp. 167-172, February 1990.

[10]. K. Kusumoto, A. Matsuzawa and K. Murata, "A 10-b 20-MHz 30-mW Pipelined Interpolating CMOS ADC," *IEEE J. Solid-State Circuits*, vol. 28, no. 12, pp. 1200-1206, December 1993.

[11]. M. Ito, T. Miki, S. Hosotani, T. Kumamoto, Y. Yamashita, M. Kijima, T. Okuda and K. Okada, "A 10 bit 20 MS/s 3V Supply CMOS A/D Converter," *IEEE J. Solid-State Circuits*, vol. 29, no. 12, pp. 1531-1536, December 1994.

[12]. J. Doernberg, P. R. Gray and D. A. Hodges, "A 10-bit 5-Msample/s CMOS Two-Step Flash ADC," *IEEE J. Solid-State Circuits*, vol. 24, no. 2, pp. 241-249, April 1989.

[13]. R. J. van de Plassche and P. Baltus, "An 8-bit 100-MHz Full-Nyquist Analog-to-Digital Converter," *IEEE J. Solid-State Circuits*, vol. 23, no. 6, pp. 1334-1344, December 1988.

[14]. H. Kimura, A. Matsuzawa, T. Nakamura and S. Sawada, "A 10-b 300-MHz Interpolated-Parallel A/D Converter," *IEEE J. Solid-State Circuits*, vol. 28, no. 4, pp. 438-446, April 1993.

[15]. C. W. Moreland, "An 8b 150MSample/s Serial ADC," in *ISSCC Digest of Technical Papers*, Feb. 1995, pp. 272-273.

[16]. P. Vorenkamp and R. Roovers, "A 12-b, 60-MSample/s Cascaded Folding and Interpolating ADC," *IEEE J. Solid-State Circuits*, vol. 32, no. 12, pp. 1876-1886, December 1997.

[17]. G. Hoogzaad and R. Roovers, "A 65-mW, 10-bit, 40-Msample/s BiCMOS Nyquist ADC in 0.8 mm^2," *IEEE J. Solid-State Circuits*, vol. 34, no. 12, pp. 1796-1802, December 1999.

[18]. M. Steyaert, R. Roovers and J. Craninckx, "A 100 MHz 8 Bit CMOS Interpolating A/D Converter," in *CICC Digest of Technical Paper*s, May. 1993, pp. 28.1.1-28.1.4.

[19]. R. Roovers and M. Steyaert, "Design of CMOS A/D Converters with Folding and Interpolating Techniques," in *Advanced A-D and D-A Conversion Techniques and their Applications - Conference Pub.*, July. 1994, pp. 76-81.

[20]. B. Nauta and A. G. Venes, "A 70-MS/s 110-mW 8-b CMOS Folding and Interpolating A/D Converter," *IEEE J. Solid-State Circuits*, vol. 30, no. 12, pp. 1302-1308, December 1995.

[21]. R. Roovers and M. Steyaert, "A 176 Ms/s, 6 b, 160 mW 3.3 V CMOS A/D Converter," *IEEE J. Solid-State Circuits*, vol. 31, no. 7, pp. 938-944, July 1996.

[22]. M. P. Flynn and D. J. Allstot, "CMOS Folding A/D Converters with Current-Mode Interpolation," *IEEE J. Solid-State Circuits*, vol. 31, no. 9, pp. 1248-1257, September 1996.

[23]. A. G. Venes and R. J. van de Plassche, "An 80-MHz, 80-mW, 8-b CMOS Folding A/D Converter with Distributed Track-and-Hold Preprocessing," *IEEE J. Solid-State Circuits*, vol. 31, no. 12, pp. 1846-1853, December 1996.

[24]. K. Bult and A. Buchwald, "An Embedded 240-mW 10-b 50-MS/s CMOS ADC in 1-mm^2," *IEEE J. Solid-State Circuits*, vol. 32, no. 12, pp. 1887-1895, December 1997.

[25]. M. Flynn and B. Sheahan, "A 400-Msample/s, 6-b CMOS Folding and Interpolating ADC," *IEEE J. Solid-State Circuits*, vol. 33, no. 12, pp. 1932-1938, December 1998.

[26]. D. Daniel, U. Langmann and B. Bosch, "A Silicon Bipolar 4-bit 1-Gsample/s Full Nyquist A/D Converter," *IEEE J. Solid-State Circuits*, vol. 23, no. 3, pp. 742-749, June 1988.

[27]. V. E. Garuts, Y-C S. Yu, E. O. Traa and T. Yamaguchi, "A Dual 4-bit 2-Gs/s Full Nyquist Analog-to-Digital Converter Using a 70-ps Silicon Bipolar Technology with Borosenic-Poly Process and Coupling-Base Implant," *IEEE J. Solid-State Circuits*, vol. 24, no. 2, pp. 216-222, April 1989.

[28]. C. W. Mangelsdorf, "A 400-MHz Input Flash Converter with Error Correction," *IEEE J. Solid-State Circuits*, vol. 25, no. 1, pp. 184-191, February 1990.

[29]. Y. Nejima, M. Hotta and S. Ueda, "An 8-b ADC with Over-Nyquist Input at 300-Ms/s Conversion Rate," *IEEE J. Solid-State Circuits*, vol. 26, no. 9, pp. 1302-1308, September 1991.

[30]. H. Reyhani and P. Quinlan, "A 5 V, 6-b, 80 Ms/s BiCMOS Flash ADC," *IEEE J. Solid-State Circuits*, vol. 29, no. 8, pp. 873-878, August 1994.

[31]. K. Ono, T. Matsuura, E Imaizumi, H. Okazawa and R. Shimokawa, "Error Suppressing Encode Logic of FCDL in a 6-b Flash A/D Converter," *IEEE J. Solid-State Circuits*, vol. 32, no. 9, pp. 1460-1464, September 1997.

[32]. M. J. M. Pelgrom, A. C. J. v. Rens, M. Vertregt and M. B. Dijkstra, "A 25-Ms/s 8-bit CMOS A/D Converter for Embedded Application," *IEEE J. Solid-State Circuits*, vol. 29, no. 8, pp. 879-886, August 1994.

[33]. J. Spalding and D. Dalton, "A 200MSample/s 6b Flash ADC in 0.6μm CMOS," in *ISSCC Digest of Technical Papers*, Feb. 1996, pp. 320-321,466.

[34]. C. L. Portmann and T. H. Y. Meng, "Power-Efficient Metastability Error Reduction in CMOS Flash A/D Converters," *IEEE J. Solid-State Circuits*, vol. 31, no. 8, pp. 1132-1140, August 1996.

[35]. S. Tsukamoto, I. Dedic, T. Endo, K. Kikuta, K. Goto and O. Kobayashi, "A CMOS 6-b, 200 MSample/s, 3 V-Supply A/D Converter for a PRML Read Channel LSI," *IEEE J. Solid-State Circuits*, vol. 31, no. 11, pp. 1831-1836, November 1996.

[36]. Y. Tamba and K. Yamakido, "A CMOS 6b 500MSample/s ADC for a Hard Disk Drive Read Channel," in *ISSCC Digest of Technical Papers*, Feb. 1999.

[37]. K. Yoon, S. Park and W. Kim, "A 6b 500MSample/s CMOS Flash ADC with a Background Interpolated Auto-Zeroing Technique," in *ISSCC Digest of Technical Papers*, Feb. 1999.

[38]. K. Nakamura, M. Hotta, L. R. Carley and D. J. Allstot, "An 85 mW, 10 b, 40 Msample/s CMOS Parallel-Pipelined ADC," *IEEE J. Solid-State Circuits*, vol. 30, no. 3, pp. 173-183, March 1995.

[39]. K. Nagaraj, H. S. Fetterman, J. Anidjar, S. H. Lewis and R. G. Renninger, "A 250-mW, 8-b, 52-Msamples/s Parallel-Pipelined A/D Converter with Reduced Number of Amplifiers," *IEEE J. Solid-State Circuits*, vol. 32, no. 3, pp. 312-320, March 1997.

[40]. S. H. Lewis, H. S. Fetterman, G. F. Gross, Jr., R. Ramachandran and T. R. Viswanathan, "A 10-b 20-Msample/s Analog-to-Digital Converter," *IEEE J. Solid-State Circuits*, vol. 27, no. 3, pp. 351-358, March 1992.

[41]. D. Fu, K. C. Dyer, S. H. Lewis and P. J. Hurst, "A Digital Background Calibration Technique for Time-Interleaved Analog-to-Digital Converters," *IEEE J. Solid-State Circuits*, vol. 33, no. 12, pp. 1904-1911, December 1998.

[42]. K. Y. Kim, N. Kusayanagi and A. A. Abidi, "A 10-b, 100-MS/s CMOS A/D Converter," *IEEE J. Solid-State Circuits*, vol. 32, no. 3, pp. 302-311, March 1997.

[43]. T. B. Cho and P. R. Gray, "A 10 b, 20 Msample/s, 35 mW Pipeline A/D Converter," *IEEE J. Solid-State Circuits*, vol. 30, no. 3, pp. 166-172, March 1995.

[44]. C. S. G. Conroy, D. W. Cline and P. R. Gray, "An 8-b 85-MS/s Parallel Pipeline A/D Converter in 1-μm CMOS," *IEEE J. Solid-State Circuits*, vol. 28, no. 4, pp. 447-454, April 1993.

[45]. M. Ishikawa and T. Tsukahara, "An 8-bit 50-MHz CMOS Subranging A/D Converter with Pipelined Wide-Band S/H," *IEEE J. Solid-State Circuits*, vol. 24, no. 6, pp. 1485-1491, December 1989.

[46]. D. W. Cline and P. R. Gray, “A Power Optimized 13-b 5 Msamples/s Pipelined Analog-to-Digital Converter in 1.2 μm CMOS,” *IEEE J. Solid-State Circuits*, vol. 31, no. 3, pp. 294-303, March 1996.

[47]. W-C. Song, H-W. Choi, S-U. Kwak and B-S. Song, "A 10-b 20-Msample/s Low-Power CMOS ADC," *IEEE J. Solid-State Circuits*, vol. 30, no. 5, pp. 514-521, May 1995.

[48]. G. T. Uehara and P. R. Gray, "A 100 MHz A/D Interface for PRML Magnetic Disk Read Channels," *IEEE J. Solid-State Circuits*, vol. 29, no. 12, pp. 1606-1613, December 1994.

[49]. Y.-M. Lin, B. Kim and P. R. Gray, “A 13-b 2.5-MHz Self-Calibrated Pipelined A/D Converter in 3-μm CMOS,” *IEEE J. Solid-State Circuits*, vol. 26, no. 4, pp. 628-636, April 1991.

[50]. S. H. Lewis and P. R. Gray, "A Pipelined 5-Msample/s 9-bit Analog-to-Digital Converter," *IEEE J. Solid-State Circuits*, vol. SC-22, no. 6, pp. 343-350, December 1987.

[51]. A. M. Abo and P. R. Gray, "A 1.5-V, 10-bit, 14.3-MS/s CMOS Pipeline Analog-to-Digital Converter," *IEEE J. Solid-State Circuits*, vol. 34, no. 5, pp. 599-606, May 1999.

[52]. Y. Sugimoto and S. Mizoguchi, "An Experimental BiCMOS Video 10-Bit ADC," *IEEE J. Solid-State Circuits*, vol. 24, no. 4, pp. 997-999, August 1989.

[53]. K. Sone, Y. Nishida and N. Nakadai, "A 10-b 100-Msample/s Pipelined Subranging BiCMOS ADC," *IEEE J. Solid-State Circuits*, vol. 28, no. 12, pp. 1180-1186, December 1993.

[54]. K. Amano, S. Matsuba and A. Yukawa, "An 8-bit 200Ms/s 500mW BiCMOS ADC," in *CICC Digest of Technical Papers*, May 1995, pp. 207-210.

[55]. D. A. Mercer, "A 14-b 2.5 MSPS Pipelined ADC with On-Chip EPROM," *IEEE J. Solid-State Circuits*, vol. 31, no. 1, pp. 70-76, January 1996.

[56]. T-H. Shu, K. Bacrania and R. Gokhale, "A 10-b 40-Msample/s BiCMOS A/D Converter," *IEEE J. Solid-State Circuits*, vol. 31, no. 10, pp. 1507-1510, October 1996.

[57]. K. Nishimura, “Optimal Partitioning of Analog and Digital Circuitry in Mixed-Signal Circuits for Signal Processing,” *Memo UCB/ERL M93/67*, Univ. Calif., Berkeley, July 1993.

[58]. D. W. Cline, “Noise, Speed, and Power Trade-offs in Pipelined Analog to Digital Converters,” *Memo UCB/ERL M*95/94, Univ. Calif., Berkeley, November 1995.

[59]. F. G. Stremler, *Introduction to Communication Systems*, 3rd ed., Reading, Mass. USA: Addison-Wesley Pub. Co., 1990.

[60]. B. Razavi, *Principles of Data Conversion System Design*, New York, N.Y.: IEEE Press, 1995.

[61]. P. R. Gray, and R. G. Meyer, *Analysis and Design of Analog Integrated Circuits*, 3rd Ed., New York, N.Y.: John Wiley and Sons, Inc., 1993.

[62]. Y.-M. Lin, “Performance Limitations on High-Resolution Video-Rate Analog-Digital Interfaces,” *Memo UCB/ERL M*90/55, Univ. Calif., Berkeley, June 1990.

[63]. T. B. Cho, “Low-Power Low-Voltage Analog-to-Digital Conversion Techniques Using Pipelined Architectures,” *Memo UCB/ERL M*95/23, Univ. Calif., Berkeley, April 1995.

[64]. B. Razavi, *Design of Analog CMOS Integrated Circuits*, Preview ed., Boston, Mass.: McGraw-Hill, 2000.

[65]. J. Everitt, J. F. Parker, P. Hurst, D. Nack and K. R. Konda, "A CMOS Transceiver for 10-Mb/s and 100-Mb/s Ethernet," *IEEE J. Solid-State Circuits*, vol. 33, no. 12, pp. 2169-2177, December 1998.

[66]. C. M. Teuscher, “Software Simulation of the InfoPad Wireless Downlink,” *Memo UCB/ERL M95/16*, Univ. Calif., Berkeley, March 1995.

[67]. J. C. Rudell, J-J Ou, T. B. Cho, G. Chien, F. Brianti, J. A. Weldon and P. R. Gray, "A 1.9-GHz Wide-Band IF Double Conversion CMOS Receiver for Cordless Telephone Applications," *IEEE J. Solid-State Circuits*, vol. 32, no. 12, pp. 2071-2088, December 1997.

[68]. T. B. Cho, G. Chien, F. Brianti and P. R. Gray, "A Power-Optimized CMOS Baseband Channel Filter and ADC for Cordless Applications," in *Symp. on VLSI Circuits Digest of Technical Papers*, June 1996, pp. 64-65.

Chapter 3

DS-CDMA System Overview

CDMA is a method to transmit multiple users' data simultaneously using coding, ideally orthogonal, to add dimensionality to the transmit space. This CDMA technique is combined with direct-sequence spread spectrum (DSSS) modulation to yield what is know as direct-sequence code division multiple access communication (DS-CDMA). Fig. 3-1 shows a conceptual block diagram of a wireless DS-CDMA system. The unmodulated users' data, $d_u(t)$, can be either a serial binary data stream (single-user case) or a serial multilevel data stream (multi-users case). This data stream is then modulated by a higher bit-rate code sequence, $p(t)$, which increases or *spreads* the bandwidth of the baseband

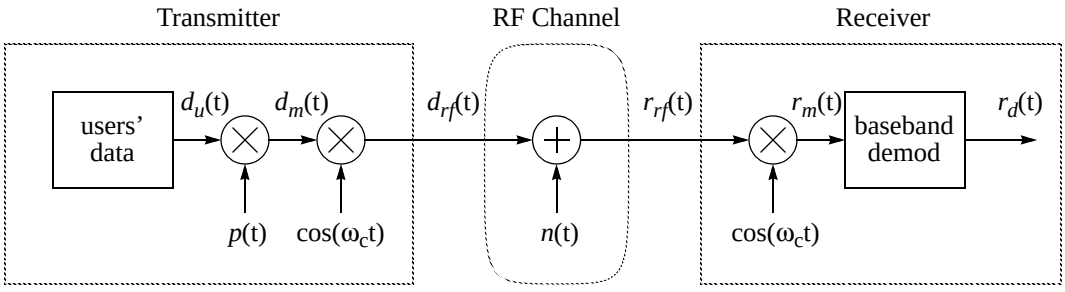


Figure 3-1 Conceptual block diagram of DS-CDMA system

data signal. This baseband spread-spectrum (SS) signal, $d_m(t)$, is finally modulated with a radio frequency (RF) carrier, ω_c, before being transmitted into the air (RF channel). During transmission through the channel the signal gets corrupted by the addition of multipath interference, random noise, and other interfering signals all of which are simply represented as $n(t)$. The received signal, $r_{rf}(t)$, is then RF demodulated[1] yielding the baseband SS signal, $r_m(t)$. The final baseband demodulation despreads and recovers the original data signal resulting in $r_d(t)$.

Since the RF transmission and demodulation steps are outside the scope of this research, the rest of this chapter will focus only on the baseband SS modulation and demodulation steps. The baseband received signal, $r_m(t)$, can be separated into three types of signals based on how the SS demodulator handles them. The first signal type is the desired user data that is to be ultimately recovered. The second is the other users' data (multiuser case) that can be rejected by the demodulator. These signals are special in that they are *synchronously orthogonal* to the desired signal (as will be explained later). And lastly are all other signals (noise, interference, multipath, etc.) that corrupt the reception of the desired user data. Since the SS demodulator treats each of these signal types very differently, it is natural to use this signal division as the basis for the organization of this chapter.

The chapter starts with an overview of SS communications and its advantages, fol-

1. For simplicity a direct conversion, or homodyne, system is illustrated (omitting the low-pass filter that usually follows the conversion). For our purposes the important result is that $r_m(t)$ is a baseband spread-spectrum signal corrupted by passband noise and interference.

lowed by a description of pseudo-random noise (PN) code sequences which are the key elements in the SS process. Next DSSS modulation and demodulation for the simplest case of a single user are presented before describing the more complex multiuser case (CDMA). How the demodulator processes noise and interference will be addressed next, where the concept of processing gain (G_P) will be introduced. The chapter will finish with a section on synchronization, explaining how the demodulator aligns itself to the incoming received signal and maintains that alignment.

3.1 Spread Spectrum

Spread-spectrum communication refers to modulating a signal for the purpose of spreading its energy across a frequency range that is wider than its original unmodulated bandwidth. The two most common methods of performing this modulation are *direct sequence* (DS) and *frequency hopping* (FH). In DSSS modulation, a high-frequency PN sequence is used to modulate the signal directly, yielding a wide-bandwidth baseband signal (see Fig. 3-2(a)). In FH spread-spectrum (FHSS) modulation, a modulating frequency is varied according to a PN sequence so that the signal occupies a frequency *slot* for a short time before hopping to another slot in a pseudo-random order (see Fig. 3-2(b)). Because of the random nature of the PN sequence, the spectrum of the DSSS signal is "white" or spectrally flat; in the case of FHSS, the hopped frequency band is evenly distributed across the expanded spectrum over time.

Expanding the bandwidth with either method will achieve the following benefits as

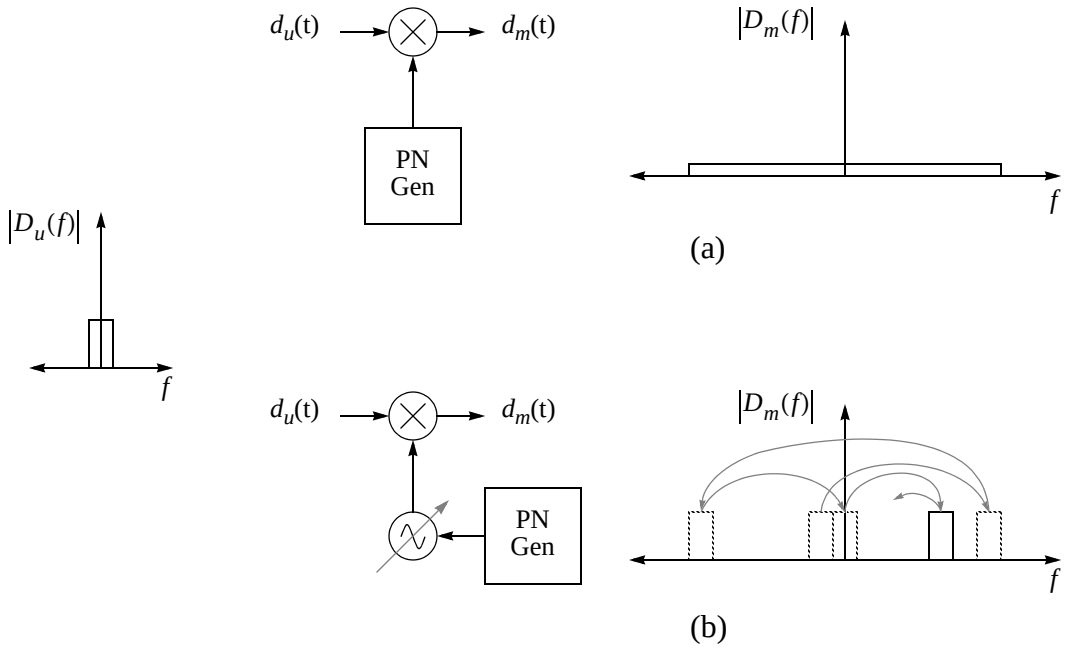


Figure 3-2 Spectrum expansion for (a) DSSS modulation and (b) FHSS modulation.

the signal passes through the RF channel:

1. Immunity to frequency-selective narrowband fading due to multipath interference.
2. Immunity from jamming and other external interfering signals.
3. Reusability of spectrum.
4. Simultaneous multi-user access of the same spectrum

Both DSSS and FHSS systems permit spectrum reusability, or multi-access, but by different means. In DSSS modulation, other signals or interferers in the same RF band only corrupt a small portion of the spread-spectrum signal, only slightly degrading its *signal-to-noise ratio* (*SNR*). And conversely, the DSSS signal looks like *white* noise to these

other signals thus permitting spectrum sharing. FHSS systems accomplishes spectrum reuse by orthogonal frequency *hopping* to avoiding interference with other fixed frequency signals in the same band. Thus, multiple SS services with unique random codes will be able to communicate simultaneously over the same spectrum, even while other non-SS services occupy parts of the band.[1] For these reasons SS modulation is often a popular choice for data and voice communications.

3.2 Pseudo-random Noise (PN) Code Sequences

The unique characteristics of PN sequences[2] make them useful in spectral whitening, random test-data generation, data scrambling and spectral expansion (as in SS). It is their close-to-ideal randomness and ease of generation that makes them so useful. These sequences are easily generated by using an *M*-bit shift register with the appropriate feedback taps, e.g. as shown in Fig. 3-3 for $M = 5$. With the appropriate taps, the length (N) of

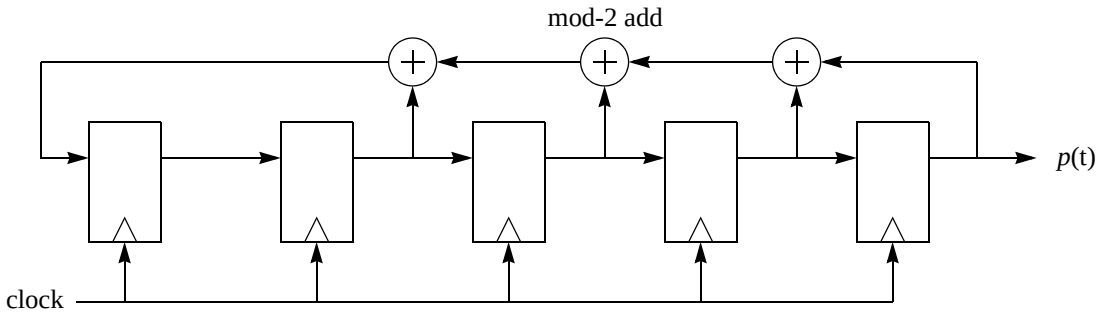


Figure 3-3 Maximal-length PN sequence generator (M=5, N=31).

1. A more in-depth description of spread spectrum systems can be found in [1] & [2].
2. Most of what follows on PN sequences is summarized from [1]

the serial bit stream at the output will be a maximum (L_{max}):

$$N = L_{max} = 2^M - 1 \qquad (3 - 1)$$

The meaning of bit-stream length in this context is the maximum length of the bit sequence before it starts repeating itself. PN sequences of maximum length are called maximal linear code sequences, but because non-maximal PN sequences are rarely used in SS systems, “PN sequences” will be used to denote maximal linear code sequences for this document. Likewise, “PN codes” or “PN code sequences” will be used synonymously with “PN sequences”. The feedback taps are added modulo-2 (exclusive OR’ed) and fed to the input of the initial shift register. Only particular tap connections will yield a maximum length for a given shift register length. A good compilation of maximal length tap connections for given register lengths can be found in [1].

These maximal length PN codes have the following properties:

1. Code balance: The number of ones and the number of zeros differ by only 1, i.e., there is 1 more one than the number of zeros. This particularly useful when the channel is AC coupled (no DC transmission).

2. Run-length distribution: The run-lengths of ones, i.e., the number of ones in a row, and zeros are distributed in a deterministic and somewhat statistically balanced way. Table 3-1 illustrates this distribution for M=7. In words, there are exactly 1 run of M ones, 1 run of M-1 zeros, 1 each of runs of M-2 ones & zeros, 2 each of runs of M-3 ones & zeros, 4 each of runs of M-4 ones & zeros, 2^3 each of runs of M-5 ones & zeros, ..., 2^K each of runs of M-2-K ones & zeros, ... and 2^{M-3} each of runs of 1 ones & zeros.

TABLE 3-1 Run-length Distribution (M = 7)

Run-length	Runs of Ones	Runs of Zeros	Bit Total
7	1	0	7
6	0	1	6
5	1	1	10
4	2	2	16
3	4	4	24
2	8	8	32
1	16	16	32
All Runs			**127**

3. Autocorrelation: Using signaling values of ±1, the autocorrelation of a PN sequence has a value of -1 for all phase shifts of more than one bit time. For no phase shift (perfect alignment with itself), the autocorrelation has a value of N, the sequence length (see Fig. 3-4).

4. Modulo-2 addition: Modulo-2 addition of a PN sequence with a shifted version of itself results in a differently-shifted version of itself.

5. Shift register states: The binary number represented by the M bits in the shift register randomly cycle through all 2^M values, except for 0, in successive 2^M-1 clocks. If the value of 0 (all shift register bits are 0) is ever present in the shift register, it will stay in that state until reloaded with a nonzero value.

The second and third properties are the most important properties of PN sequences for use in SS systems. The run-length distribution property is responsible for the randomness and for the spectral “whiteness” of the PN sequence. The randomness is ideal for generating pseudo-random data for testing purposes while its whitening characteristic makes it useful as a scrambling modulator to flatten the spectral content of a signal prior to

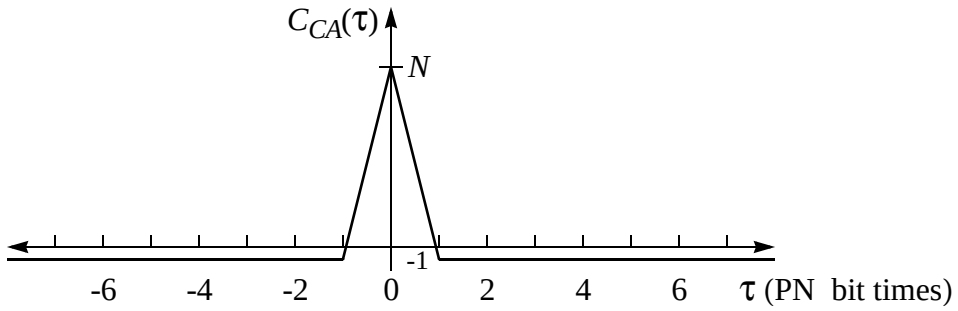


Figure 3-4 Continuous-time autocorrelation function

transmission or as a spreading modulator to expand as well as flatten a signal.

The continuous-time (CT) autocorrelation of a PN sequence is defined as:[1]

$$C_{CA}(\tau) = \frac{N}{N \cdot T_c} \cdot \int_{N \cdot T_c} p(t) \cdot p(t+\tau) dt \qquad (3\text{ - }2a)$$

$$= \frac{1}{T_c} \cdot \int_{N \cdot T_c} p(t) \cdot p(t+\tau) dt \qquad (3\text{ - }2b)$$

where p(t) is the PN sequence as a function of time with value ±1, N is the number of PN sequence bits, T_c is a PN-bit time and $N \cdot T_c$ is its length. Fig. 3-4 is a plot of the CT autocorrelation function, $C_{CA}(\tau)$. This plot assumes that p(t) wraps around when computing the integral. Timing synchronization in a DSSS receiver takes advantage of the unique shape of C_{CA}(t) by searching for the peak autocorrelation value which signals perfect alignment ($\tau = 0$) of the incoming received signal to the DSSS receiver's internally generated PN sequence. The longer the PN code length, N, the larger the correlation peak and

1. This equation has been scaled by N so that the continuous-time and discrete-time correlations give equivalent results.

hence the more resistant to false synchronization from noise and interference (See section 3.6 “Synchronization”).

3.3 Single-user DSSS Modulation

The simplest case of DSSS modulation is that of transmission and reception of a single (user) data stream. In this case, a PN code directly modulates the binary data signal, d_u(t) (see Fig. 3-1). The bandwidth expansion of the signal’s spectrum (Fig. 3-2) is a result of the higher bit rate of the PN code. The expansion or ***spread factor*** (F_S) is the ratio of the PN code bit rate to the data signal bit rate. The modulation process is illustrated in Fig. 3-5. The modulated output d_m(t) has the same bit rate as the PN code. To avoid confusion with the data bits (d_u(t)), the PN code bits and modulated data bits (d_m(t)) are commonly referred to as ***chips***. Thus bit time (T_b) will refer to the duration of one unmodulated data bit and chip time (T_c) to that of a PN code or modulated data bit time. With this convention, an alternative definition for spread factor is:

$$F_S = \frac{T_b}{T_c} \qquad (3 - 3)$$

The modulation can be viewed two ways depending on the type of implementation used. If the implementation uses a differential type of signaling, i.e., $\pm A$, where A is the amplitude of the signal, then a multiplication operation is equivalent to modulation (Fig. 3-5(a)). For standard digital logic with binary signaling, i.e., {0,1}, an exclusive OR operation (XOR) can be used for modulation (shown as a modulo-2 addition in

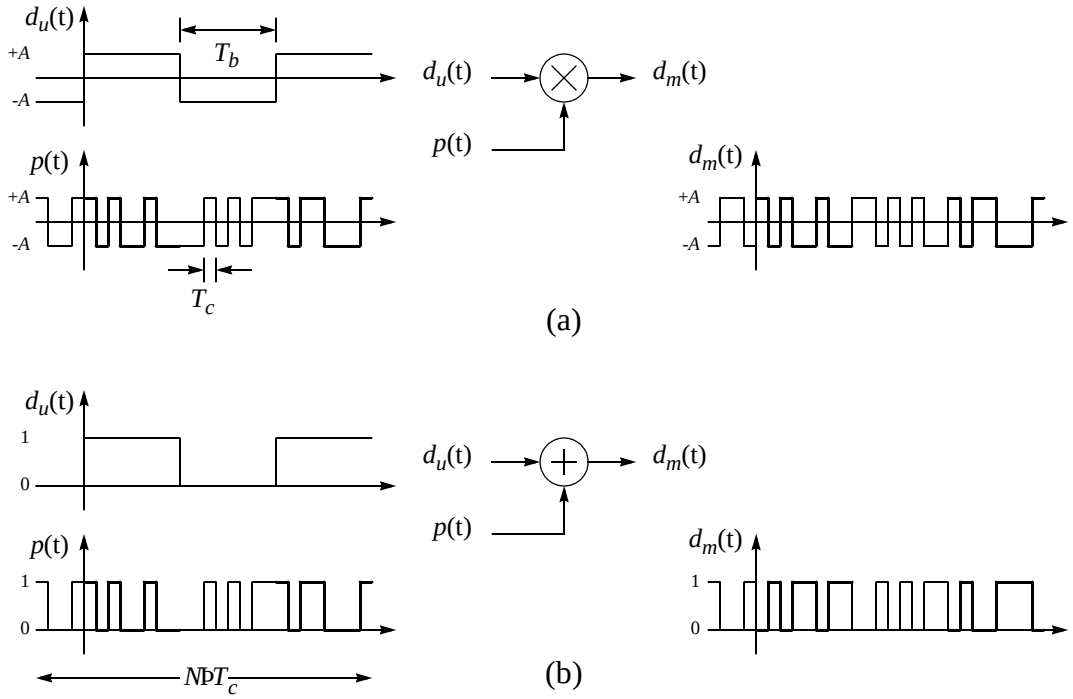


Figure 3-5 DSSS modulation representation ($N \cdot T_c > T_b$); (a) multiplication for binary antipodal signaling {±*A*}; (b) XOR for digital logic {0,1}

Fig. 3-5(b)[1]). Normally the PN code length, $N \cdot T_c$, is greater than or equal to the data bit time, T_b. Fig. 3-5 shows the case when $N \cdot T_c > T_b$. When $N \cdot T_c = T_b$, the modulation can be viewed as mapping bits of d_u(t) to one polarity of the entire PN sequence or the other (see Fig. 3-6).

3.3-1 Demodulation of a single DSSS user

Recovering the original data stream requires timing alignment of the receiver's internal PN code (p(t)) to the embedded PN code in the baseband received signal (r_m(t)), then demodulation, or despreading, of the received signal and finally detection of the data bits.

1. Actually the equivalent operation for binary numbers is an exclusive NOR operation, but for simplicity an XOR operation will be used in this document along with a modulo-2 addition as its symbol.

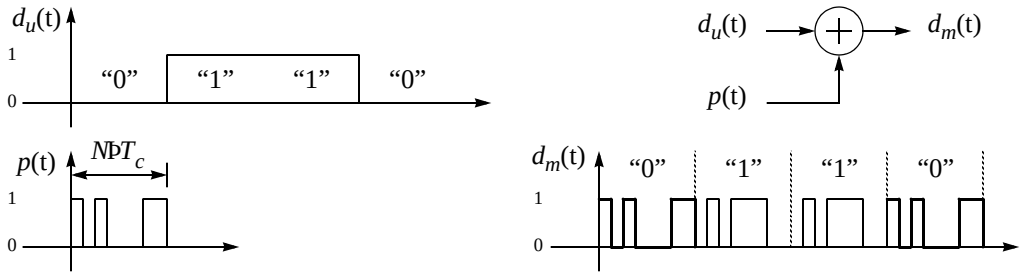


Figure 3-6 DS modulation for $N \cdot T_c = T_b$ (XOR representation)

For this discussion, perfect PN code alignment during demodulation is assumed, deferring the topic of timing alignment, or synchronization, to the end of the chapter.

Demodulation of the DS signal converts the wideband modulated signal back to a narrowband signal indicating some type of decimation or filtering function. Correlation, or more specifically cross-correlation, of the received signal ($r_m(t)$) with the desired PN code ($p(t)$) accomplishes this decimation. Fig. 3-7 shows a simplified block diagram of a DSSS receiver where a correlator is used as the baseband demodulator. CT cross-correla-

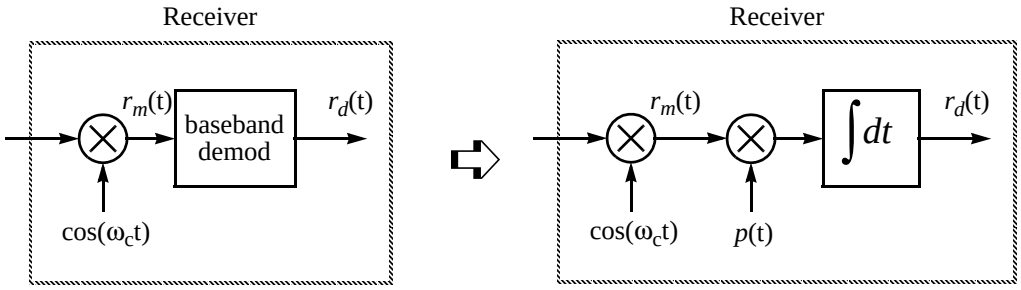


Figure 3-7 Correlation as DSSS baseband demodulator

tion for **data demodulation** is defined as:[1]

$$C_{CX}(\tau) = \frac{1}{T_c} \cdot \int_{T_b} r_m(t) \cdot p(t+\tau)dt \qquad (3\text{ - }4)$$

where the extent of the integration in this case is restricted to the duration of one data bit (T_b) as opposed to the PN sequence length.

For clarity, this discussion of the demodulation process will be restricted to an ideal noiseless transmission and reception of a single user's data signal. For this case, the baseband receive signal would be:

$$r_m(t) = A \cdot d_m(t) = A \cdot d_u(t) \cdot p(t) \qquad (3\text{ - }5)$$

where A represents the signal amplitude, $d_m(t)$ is the PN modulated baseband binary signal, and $d_u(t)$ is the original unmodulated binary data signal. For simplicity also assume that $A = 1$, so $r_m(t)$ looks exactly like the baseband modulated data signal, $d_m(t)$. Then the CT cross-correlation with perfect synchronization ($\tau = 0$) is:

$$C_{CX}(0) = \frac{1}{T_c} \cdot \int_{T_b} r_m(t) \cdot p(t+0)dt \qquad (3\text{ - }6a)$$

$$= \frac{1}{T_c} \cdot \int_{T_b} d_u(t) \cdot p(t) \cdot p(t+0)dt \qquad (3\text{ - }6b)$$

1. This equation has been scaled by F_S so that the continuous-time and discrete-time correlations give equivalent results.

$$= d_u\Big|_{T_b} \cdot \frac{1}{T_c} \cdot \int_{T_b} p^2(t)dt \qquad \text{(3 - 6c)}$$

$$= d_u\Big|_{T_b} \cdot \frac{T_b}{T_c} = d_u\Big|_{T_b} \cdot F_S \qquad \text{(3 - 6d)}$$

So the correlator output is equal to the original data bits evaluated at T_b intervals and scaled by the spread factor. Fig. 3-8 shows input and output waveforms for this example when F_S=8 and N=31. From the output waveform it is clear that the final data recovery is simply sign detection of the correlator output at T_b intervals. The output magnitude contains no data information itself but does represent the strength of the signal which can be

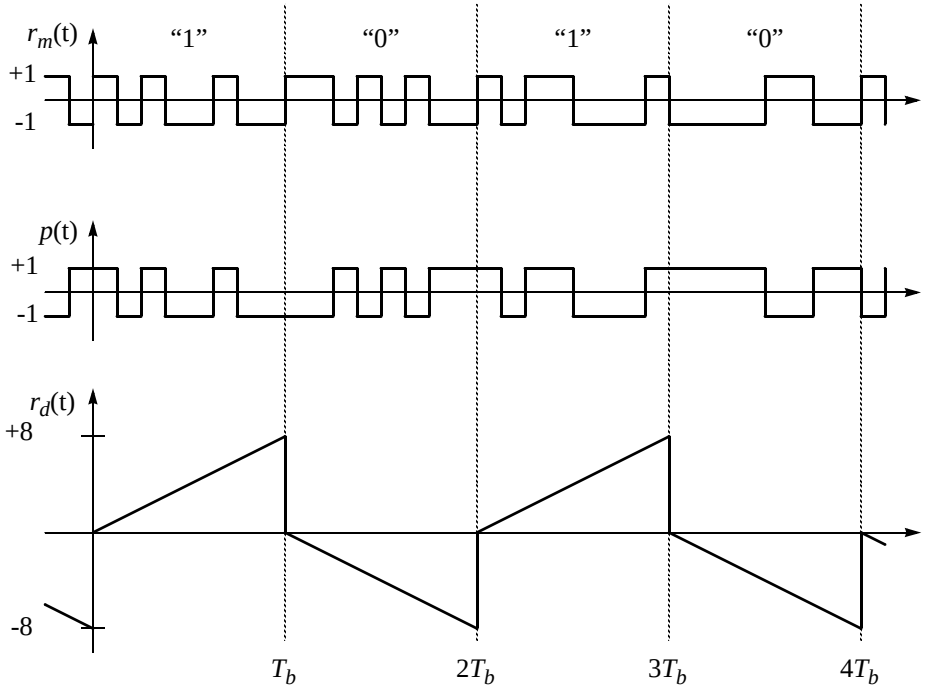


Figure 3-8 Correlator input [r_m(t) and p(t)] and output [C_{CX}(t)] waveforms for F_S=8 and N=31.

used in calculating the post-demodulation *SNR* to give an indication of the correlation bit-error rate performance.

In a more realistic example, the received signal could be corrupted by interference and noise signals as well as distorted by the RF channel characteristics. This effect will randomly vary r_m(t) such that the product $r_m(t) \cdot p(t)$ will be ±1 with some variance. Integrating this variance over the interval T_b results in an averaging effect which reduces the variance, or noise power, while not affecting signal power. This reduction in the noise power is called the *processing gain* (G_P) of a SS system and will be looked at more closely in a later section.

The effect of timing misalignment in an actual system will result in the reduction of the correlation output magnitude. The output magnitude must decrease with misalignment since its peak value only occurs for perfect alignment. So instead of increasing the noise power of the correlator output as in the additive noise case above, misalignment reduces the signal power but the result is the same; the *SNR* is reduced.

The preceding analysis has been presented in the continuous-time domain. Alternatively, a sampled-data system can perform a correlation in the discrete-time (DT) domain which may be more resistant to timing misalignment error. With a sampling interval of T_c the DT correlation for **data demodulation** is defined as:

$$C_{DX}(n) = \frac{1}{T_c} \cdot \sum_{1}^{r_S} r_m(i) \cdot p(i+n) \cdot T_c \qquad (3\text{ - }7a)$$

$$= \frac{1}{T_c} \cdot T_c \cdot \sum_{1}^{r_S} r_m(i) \cdot p(i+n) \qquad (3\text{ - }7b)$$

$$= \sum_{1}^{r_S} r_m(i) \cdot p(i+n) \qquad (3\text{ - }7c)$$

where $r_m(i)$ are the sampled values of $r_m(t)$ and $p(i)$ are the chips of the PN code. Fig. 3-9 illustrates how the sampled values of the received signal can tolerate almost ±0.5Tc misalignment without affecting its value. Fig. 3-10 is a comparison of the CT and DT autocorrelation functions (both ideal). Note that the DT curve is the loci of possible correlation values as the phase is incrementally changed. In summary, with CT correlation

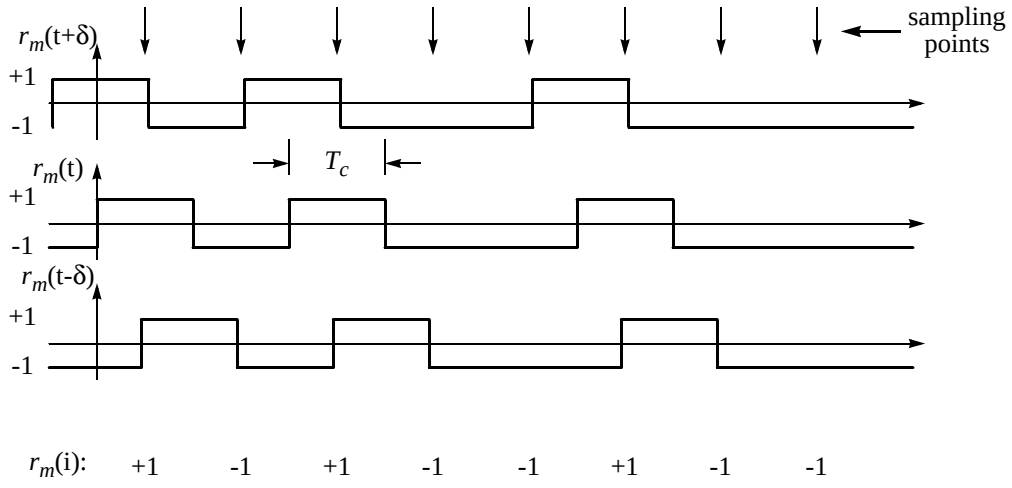


Figure 3-9 ±0.5T_c misalignment tolerance of samples for DT correlation.

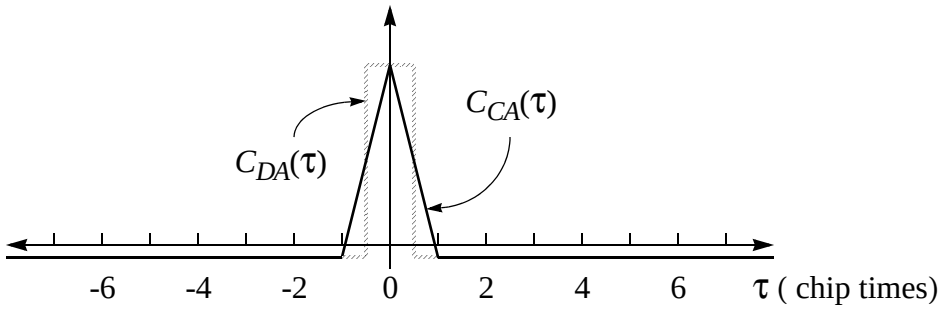


Figure 3-10 Comparison between CT and DT autocorrelation function of a PN sequence.

there is a slight decrease in the correlation output amplitude with slight timing misalignment, whereas the DT correlation output remain unaffected.

3.4 Code Division Multiple Access

In this section, transmission of data from multiple users via CDMA is addressed. In DSSS, multiple users can simultaneously transmit and receive over the same frequency spectrum since each user's modulated data looks somewhat like white noise to the other users. Differentiation of each user's data is achieved by using a different PN code to modulate and spread the data. Ideally, if *K* codes are orthogonal they span a *K*-dimensional space which can accommodate *K* users. Orthogonality in this context means that the cross-correlation of any of the *K-1* undesired users' DS-modulated signal with the PN code of the desired user is zero. In this sense, the receiver's correlator performs a dot-product between the desired PN code and the incoming received signal. So only the desired incoming signal will have a non-zero correlation resulting in recovery of that user's data.

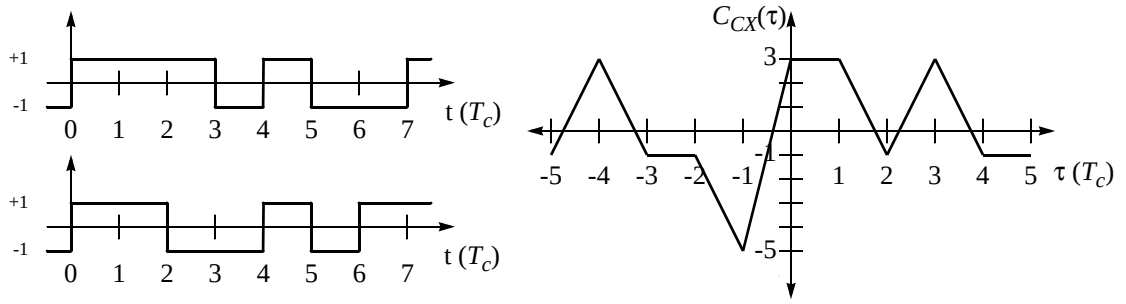


Figure 3-11 CT cross-correlation of two 7-chip PN codes.

The flaw with this scenario is that it is difficult to find a set of binary codes that are orthogonal in an asynchronous multi-user transmission environment. A simple example of the cross-correlation of two 7-chip PN codes ("1110100" and "1100101", coded as ±1) can illustrate this. Fig. 3-11 is a plot of the cross-correlation versus time shifts between the two codes. A correlation value of -1 in this case is essentially zero, yielding only 3 (-3, -2, and 2) of 7 possible chip phases (τ) where the two PN codes are “orthogonal”. Since asynchronous transmission cannot guarantee phase alignment, the two transmissions could interfere with one another. In the case of long PN codes, orthogonal alignment over the length of the entire code is not possible if a large enough mismatch exists in the time references of the receiver and the interfering transmitter. So although complete orthogonality is not possible,[1] the white-noise attribute of DSSS signals does reduce the degradation on SNR between interfering signals, permitting multi-user communication if the number of users can be limited to guarantee a desired minimum SNR. This reduced degradation of SNR is described in more detail in Section 3.5 “Processing Gain”.

1. It is possible to find a set of PN codes, called Gold Codes, for which the cross-correlation is bounded thus providing partial orthogonality for asynchronous multi-user transmission (see [1])

There is a special case of ***synchronous*** multi-user transmission that maintains the phase relationship between different user's signals and where orthogonality remains valid. This case is the downlink transmission of a base-station type architecture similar to those used in cellular phones [3] and pico-cellular LAN [4] systems. In this situation the data transmitted to each of the users can be code-modulated and packaged together before being sent synchronously. Under these conditions the codes used need only to be orthogonal ***synchronously***. An example of synchronous orthogonality is the relationship between the sine and cosine functions at a given frequency, i.e., they are orthogonal as long as the 90° phase shift difference is maintained (this is the basis for quadrature phase shift keying (QPSK) communication). For a given length of digital codes, many mutually synchronously-orthogonal codes exist and can be used to create a *K*-dimensional code space for the downlink transmission of the base-station. *Walsh* functions are one set of synchronously-orthogonal 2^M codes that can be generated by the Hadamard Matrix recursion formula [5]. Fig. 3-12 is a display of 7 of the 16 Walsh functions of length 16. The first code is just a constant or "DC" value; codes 2 and 3 are digital equivalents to sine and cosine of the fundamental (lowest frequency); codes 4 and 5 are digital equivalents to sine and cosine of the second harmonic; but code 6 has no sine or cosine equivalent. Code 6 illustrates the flexibility of digital code representation to provide many more synchronously-orthogonal sequences than simply deriving the digital equivalent from a Fourier series expansion. By examination, all 6 codes are *mutually* orthogonal, i.e., their cross-correlation over 16 chips is zero (assuming ±1 signal values); and this is also true of all 16 Walsh functions of length 16.

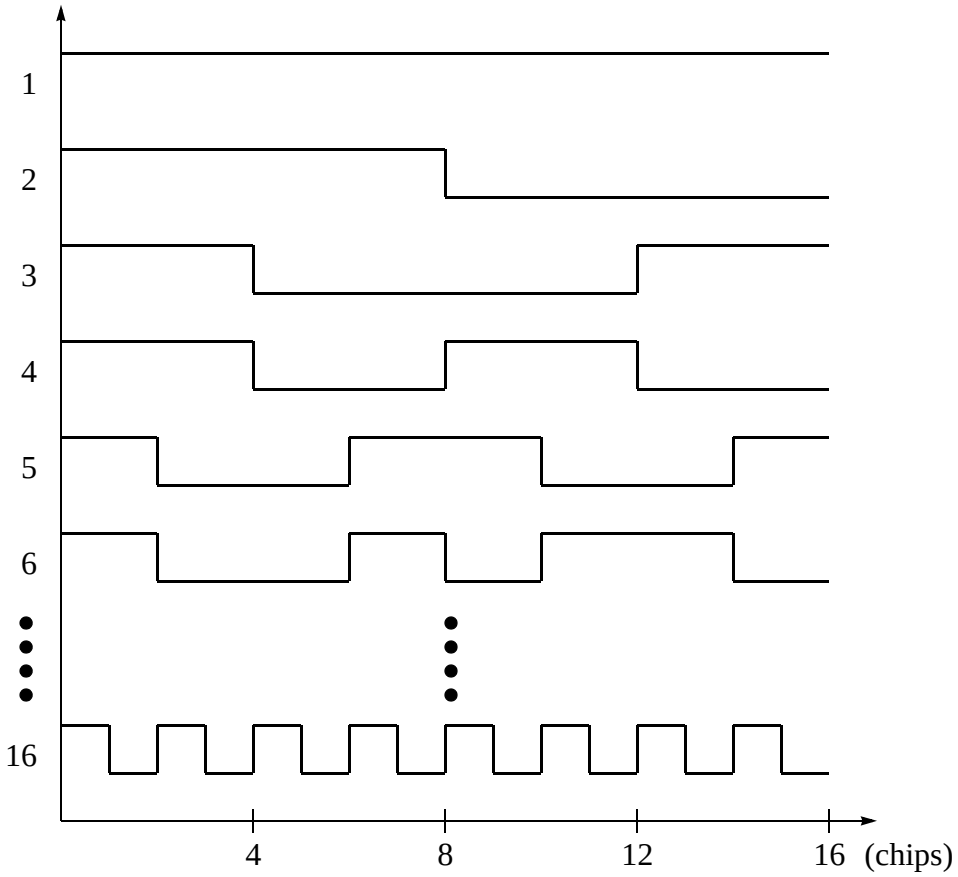


Figure 3-12 7 of 16 Walsh functions of length 16.

Selecting a unique Walsh code to modulate each user's signal, CDMA is analogous to quadrature modulation using the sine and cosine functions, where in this case a 2-dimensional space is created. Fig. 3-13 is a simplified block diagram of the CDMA transmitter.[1] Normally the orthogonal code rate is the same as the PN code rate so the orthogonal code spreads the spectrum of the user's data. However, in general, these orthogonal codes have a very poor autocorrelation characteristic and therefore cannot be

1. Note that unlike in the single-user case, the analog combined signal, $d_w(t)$, would be a multilevel signal.

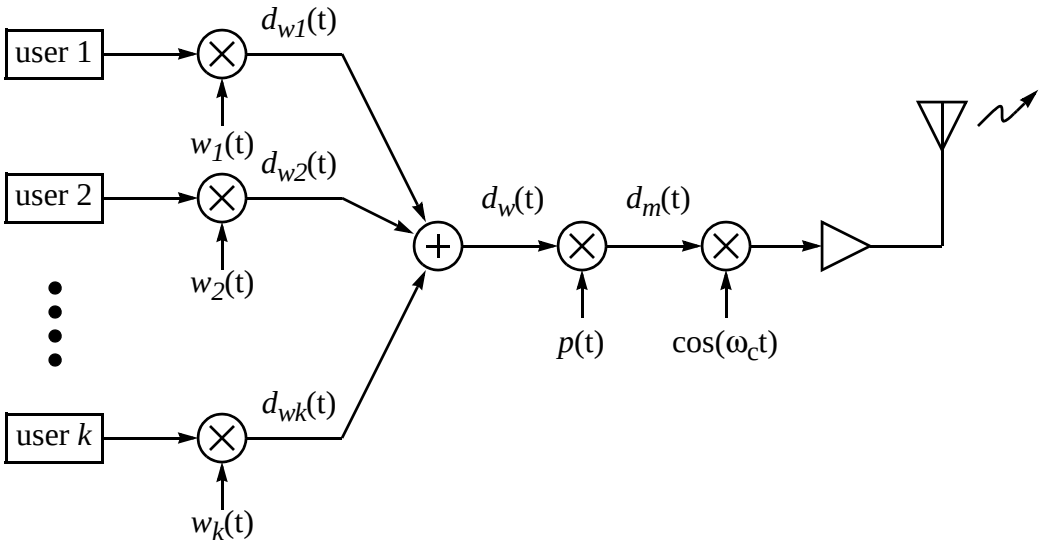


Figure 3-13 An example of a CDMA transmitter

used for synchronization in the receiver. Thus a PN code modulation is added to aid the synchronization block in achieving timing alignment by detecting the peak of the autocorrelation characteristic.[1] Fig. 3-14 shows an example of the signal waveforms for the simple case of 3 users using the Walsh functions {2, 3 and 8} for $F_S = 8$ and $N = 31$. Since the PN code modulation is common to all signals it does not disrupt the orthogonality of the signals.

3.4-1 Demodulation of CDMA signals

Recovery of this doubly-modulated baseband signal can be accomplished by double demodulation as shown in Fig. 3-15(a). The Walsh and PN codes can be pre-mixed (pw_k(t)) to ease implementation (Fig. 3-15(b)). Fig. 3-16 illustrates the waveforms[2] for

1. This fact will be covered in more detail later in Section 3.6 “Synchronization”.
2. Assuming idealized noise-free and interference-free reception

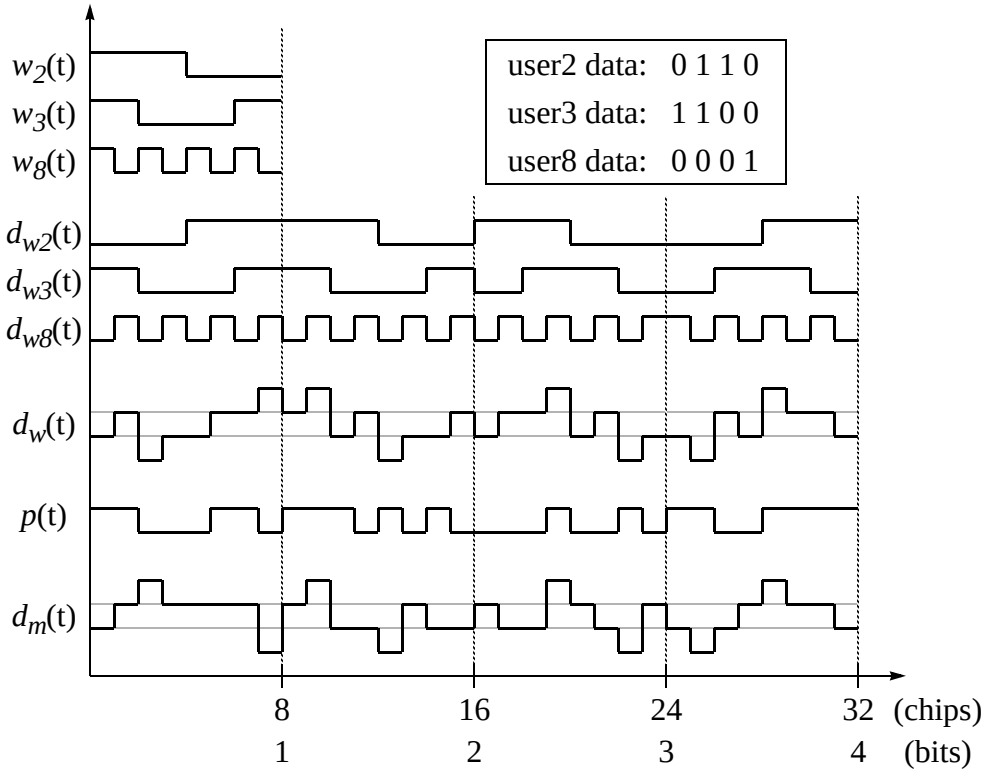


Figure 3-14 Example of CDMA waveforms (F_S = 8).

demodulating the data of user3 in the example of Fig. 3-14. Note that unlike the single-user case, the integration is not a linear ramp but a crooked one due to integrating the

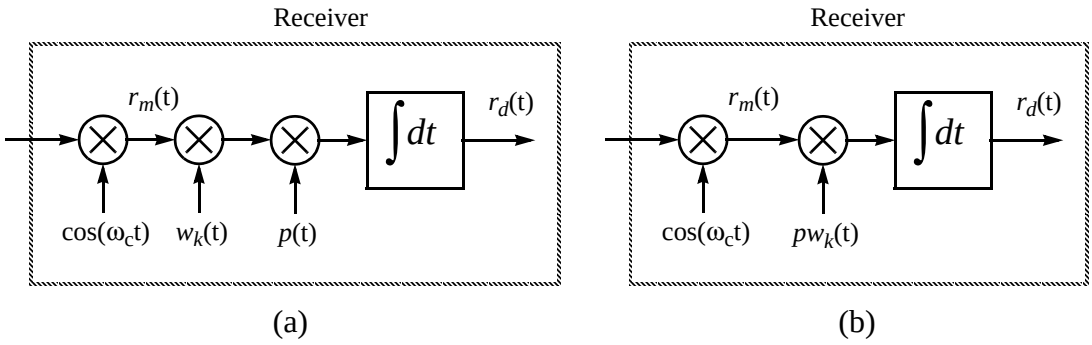


Figure 3-15 Multi-user CDMA receivers: (a) straight-forward implementation; (b) pre-mixed Walsh and PN codes (pw_k(t)) implementation.

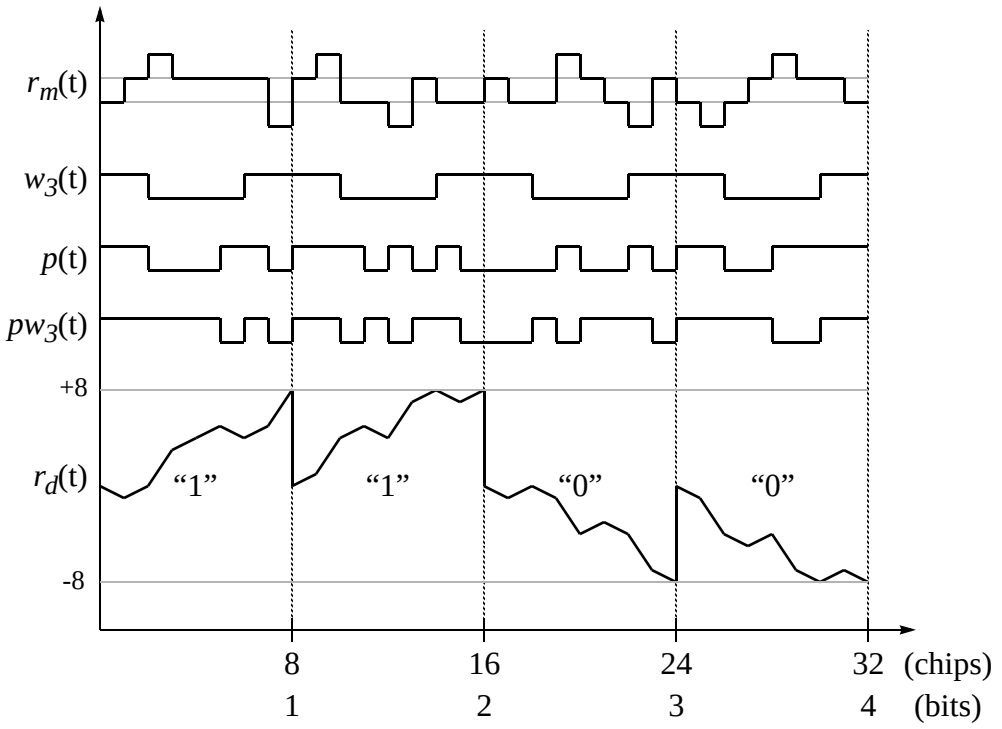


Figure 3-16 Input and output waveforms of multi-user demodulation process (user3 data recovered) for the Fig. 3-14 example.

other users' signals. However the final value is the same since the other users' signals integrate to zero.

There are two other things of note. First, linearity of the correlator is important for maintaining the orthogonal relationship of the signals, i.e., to ensure that the other users' signals integrate exactly to zero. Second, for the same desired-user signal strength, the dynamic range of the correlator must be larger in the multi-user case to accommodate the larger multilevel signal.

3.5 Processing Gain

The previous two sections explained how recovery of the desired user's signal and rejection of undesired (orthogonal) users' signals were accomplished. Now rejection of other signals that might be found in the received signal are now considered. These signals can be divided into two categories: narrowband and wideband interference. Narrowband interference can be simplified to a carrier wave signal, as in a broadcast radio transmission. Wideband interference can encompass white noise sources, multipath SS signals, FHSS signals or asynchronous DS-CDMA signals. In the case of a carrier wave interferer, the correlation process mixes the signal with a PN code and then integrates the result. The PN modulating step of the correlation randomly scrambles the sine wave and spreads its spectrum exactly like the DSSS transmission process, making it look like a pseudo-white wideband signal prior to integration. The integrating step then averages or low-pass filters the interference, decreasing its variance or mean-square signal power. In the second case of wideband interference, the PN modulating step will tend to whiten, or flatten, the spectrum while the integrating step will again lower the variance or power of the interfering signal. In either case the averaging effect by the correlator lowers the overall interference power and thus increases the *SNR* (assuming the desired signal power remains constant) by what is known as the processing gain, G_P. Thus G_P can be defined as:

$$G_P = \frac{SNRo}{SNRi} \tag{3 - 8}$$

where *SNRo* and *SNRi* are the output and input *SNR* of the correlator, respectively.

Since the reduction in the noise power is by a factor equal to the length of the correlation and thus the amount of spectrum compression, a somewhat more useful but narrower definition for G_P is:

$$G_P = \frac{BW_{SS}}{BW_D} \cong \frac{T_b}{T_c} = F_S \tag{3 - 9}$$

where BW_D and BW_{SS} are the bandwidth of the data before and after SS modulation, respectively. This equation applies primarily to the demodulation process as opposed to the synchronization process as will be explained in the next section. A more precise derivation of G_P can be found in [2] but a less rigorous derivation, and one of a narrower scope, now follows that may yield more intuition.

With respect to this research, it is appropriate to restrict the scope of this derivation to sampled-data or discrete-time processing. This derivation is further simplified to the case where samples are taken at the chip rate (every T_c) and each sample of the received input signal, $r_m(i)$, corresponds to one PN code chip, $p(i)$. Assuming the single-user case, $r_m(i)$ can be decomposed as:

$$r_m(i) = d_m(i) + n(i) \tag{3 - 10}$$

where $d_m(i)$ is the sampled value of $d_m(t)$ (desired signal) and $n(i)$ is the undesired (noise) portion of $r_m(i)$. Assuming a ±1 PN code chip, PN modulation is just a random "sign" modulation prior to integration. For $n(i)$, this sign modulation does not change its characteristics since its value and sign are assumed to be random. Thus discrete-time integration

of the $n(i)$ samples results in a summation of these random values. Treating each sign-modulated $n(i)$ sample as a random variable X_i, a new random variable Y can be defined as the correlation of X_i, so from (3 - 7a):

$$Y = \sum_{1}^{N} X_i \tag{3 - 11}$$

Assuming that each X_i is independent and identically distributed (iid) (especially in light of the random modulation by the PN code prior to summation) then the variance of Y is:

$$var(Y) \equiv \sigma_y^2 = \sum_{1}^{N} \sigma_x^2 = N \cdot \sigma_x^2 \tag{3 - 12}$$

where σ_x is the variance of X_i, σ_y^2 represents the total post-correlation noise power and N is the correlation length. There are no cross-correlation terms in the above equation since the X_i variables are independent of each other. From (3 - 7a), the recovered desired signal is $A \cdot N$ where A is the amplitude of the desired signal and N is the correlation length. Thus *SNRo* is:

$$SNRo = \frac{(A \cdot N)^2}{N \cdot \sigma_x^2} = \frac{A^2 \cdot N}{\sigma_x^2} \tag{3 - 13}$$

Comparing this to the input *SNR* which is:

$$SNRi = \frac{A^2}{\sigma_x^2} \tag{3 - 14}$$

yields a processing gain of:

$$G_P = \frac{SNRo}{SNRi} = N \tag{3 - 15}$$

Thus the processing gain or improvement in *SNR* via the correlation process is equal to the correlation length *N*. The correlation length for demodulation is the same as the spread factor, F_S, so the wider the transmitted bandwidth the larger the improvement in *SNR* during demodulation.

A subtlety that may be overlooked, when considering the benefits of SS modulation, is the lack of improvement in the overall system *SNR* when only additive white Gaussian noise (AWGN) is considered. This happens because AWGN has energy at all frequencies that are allowed to pass through the wideband front-end receiver required by SS systems, permitting more input noise power than in the narrowband case. Since the noise power is exactly proportional to F_S, the *SNRi* degradation exactly cancels the G_P improvement yielding no net *SNR* improvement over a narrow-band system. The advantage of SS modulation occurs when considering other types of noise sources. This is best understood by examining the SS demodulation process.

Though the SS demodulation result seems to indicate that the signal power increases by *N*, it is the noise variance that actually decreases by *N* via the integration process, i.e., (3 - 13) can be rewritten as:

$$SNRo = \frac{(A \cdot N)^2}{N \cdot \sigma_x^2} = \frac{A^2}{\sigma_x^2 / N} \tag{3 - 16}$$

Thus the power of any noise signal at the input of the correlator is reduced by *N*, i.e., the correlation length or, equivalently, the processing gain. So the benefit of SS modulation is that any finite-energy or narrow-band noise introduced in the channel or by the front-end SS receiver circuitry will be attenuated by the processing gain factor yielding an improvement in the overall *SNR*.

3.6 Synchronization

Synchronization is the two-phase process of obtaining and maintaining the timing alignment of the PN code internally generated by the DSSS receiver to the PN code embedded in the received signal, r_m(t). In the first (acquisition) phase a coarse alignment to within one chip time (T_c) is achieved. Once acquisition is attained the second (tracking) phase begins. The tracking phase performs a fine alignment to within a fraction of T_c and maintains that alignment while the demodulation is in progress.

3.6-1 Acquisition Mode

As mentioned earlier, acquisition and tracking rely heavily on the autocorrelation characteristic of PN sequences (see Fig. 3-4). For this to be true, the tracking correlation length must equal the PN sequence length, $N \cdot T_c$. However for demodulation, the correlation length must equal the data bit length, T_b. Thus when referring to tracking correlations, the above definitions for autocorrelation and cross correlation must be modified appropriately. Acquisition is carried out by monitoring the correlator's output for successive chip (T_c) phase alignments until the peak region is detected, signaling an alignment

within $\pm T_c$ of peak center. This amounts to tracing the autocorrelation curve of the PN sequence by the correlator. Two difficulties are sometimes encountered in practice; data modulation of the PN sequence and a sequence that is longer than a data-bit time ($N \cdot T_c > T_b$). If the PN sequence is only data modulation so that its length is equal to T_b ($N \cdot T_c = T_b$), then acquisition can be attained by including negative as well as positive peak detection. Or when the sequence is longer than T_b ($N \cdot T_c > T_b$) but is not modulated by data, then performing a longer correlation (beyond that needed for data demodulation) will suffice for acquisition. However, when both conditions occur, the peak detection process can be degraded or completely impaired. In this case, data demodulation has to be done first before a full length (*N*-chip) correlation can be done (see Fig. 3-8 for reference). Unfortunately the data demodulation cannot occur until synchronization is achieved so a paradox is reached.

A couple of approaches can be used to overcome this problem.[1] One approach is to calculate partial correlation quantities of length T_b and sum their absolute values to get the full length correlation. While this yields the correct peak amplitude, the off-peak correlation values are not all -1 (see Fig. 3-17) as in the ideal case. From Fig. 3-17, the noise margin is now reduced (by about half the ideal value in this example) which increases the chance for false acquisition lock. However if the noise margin is adequate and the threshold for peak detection is placed appropriately, successful acquisition may be possible.

1. These are not meant to be exclusive solutions to this problem and are only presented for informational purposes.

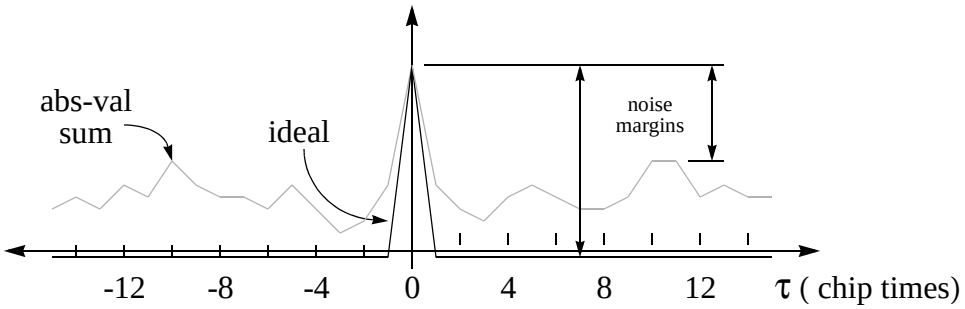


Figure 3-17 Ideal versus "absolute-value sum" CT autocorrelation functions for the waveforms of Fig. 3-8.

Another approach is to use orthogonal modulation as in CDMA (See Section 3.4 "Code Division Multiple Access") and reserve one of the user channels for sending a "pilot tone" that can be used for timing alignment. This pilot tone would have no data modulation, and thus it can produce an ideal autocorrelation characteristic. This approach is easy to implement and robust in operation so it is often used in both the cellular phone [3] and wireless LAN [4] DS-CDMA systems. As the prototype chip for this research also uses this approach, its description will be covered in detail in Chapter 5.

Acquisition time (τ_{acq}) is the time it takes the receiver to achieve initial coarse PN code alignment (lock) or re-achieve alignment after losing lock. For the two most popular correlator architectures, the ***sliding correlator*** and the ***matched filter***, τ_{acq} is strongly influenced by the implementation. The sliding correlator is simpler and more straight forward where, in the acquisition mode, it processes the input signal serially, finishing the correlation after processing *N* samples (assuming a DT correlation length of *N)*. If the correlation peak is not detected, the correlator shifts the PN code by one T_c with respect to the

input signal and serially processes the next N samples. Since there are N possible chip phases and each correlation takes $N \cdot T_c$, the maximum τ_{acq} with this approach would be:

$$\tau_{acq}(max) = N \times N \cdot T_c \tag{3 - 17}$$

The matched filter approach uses a pipelined architecture where different chip-phase correlations are being computed simultaneously. Discrete-time block diagrams of these two approaches are shown in Fig. 3-18 which reveals their differences. The matched filter has an initial latency of $N \cdot T_c$ to output the first chip-phase correlation, but then outputs successive chip-phase correlations at successive chip times, so τ_{acq}(max) is:

$$\tau_{acq}(max) = 2 \times N \cdot T_c \tag{3 - 18}$$

Note in both cases the minimum τ_{acq} is the time for a single correlation, or:

$$\tau_{acq}(min) = N \cdot T_c \tag{3 - 19}$$

Thus the matched-filter implementation has a much shorter acquisition time but is proportionately more complex and costly (circuitry and power) to implement than the sliding

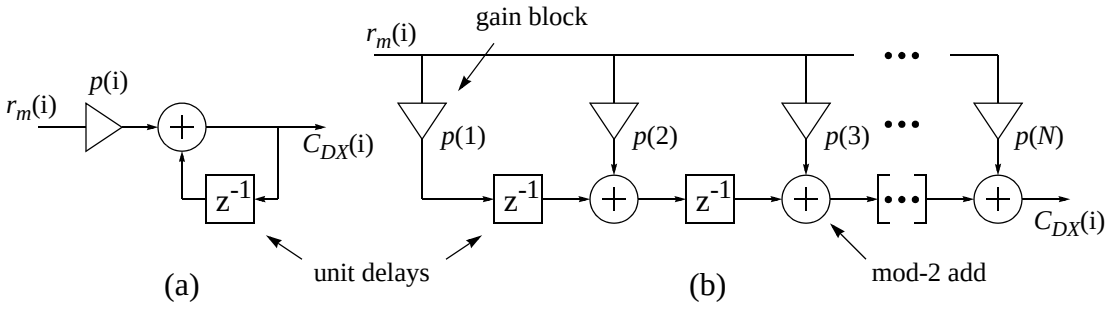


Figure 3-18 Discrete time correlators; (a) *sliding correlator*; (b) *matched filter*

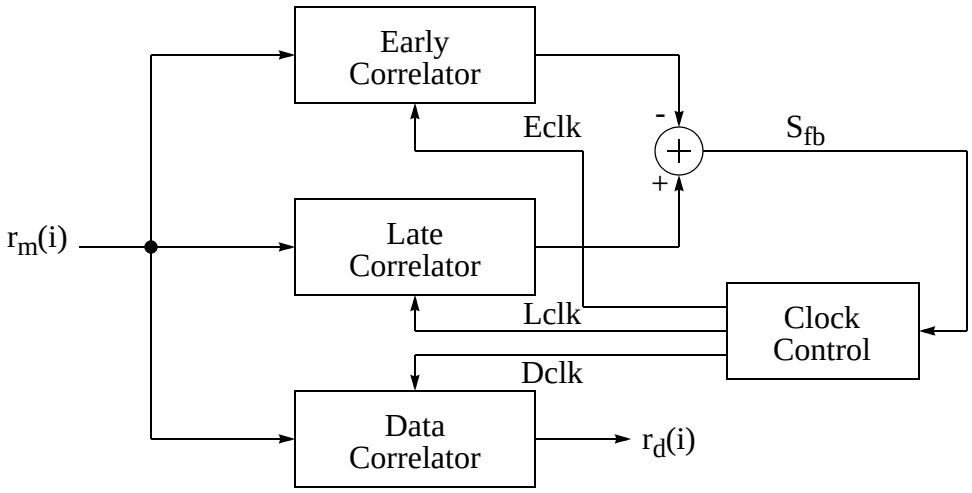


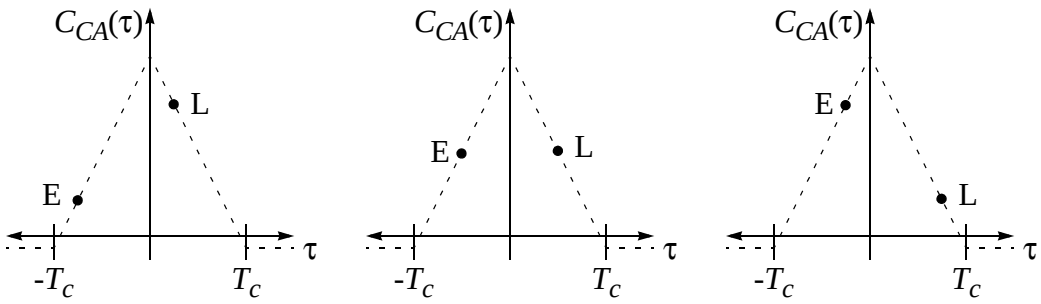
Figure 3-19 Delay-Locked Loop with the Data demodulator

correlator version.

3.6-2 Tracking Mode

PN code alignment is maintained by using two *tracking* correlators in a feedback configuration called a *delay-locked loop* (DLL).[1] One tracking correlator (Early) samples the input a half chip time ahead of the data demodulator (Data), while the other correlator (Late) samples one half chip time behind the data correlator[2] (see Fig. 3-19). The feedback attempts to equalize the output values of the two tracking correlators. This can best be understood by considering the CT C_{CA}(t) function and Early & Late correlator results for three cases: internal PN code ahead of the input signal, aligned to the input signal and behind the input signal (Fig. 3-20). The Early, Data, and Late correlators perform a corre-

1. Though it unfortunately shares the same name, this DLL is completely different from the one used to lock a inverter delay line to a reference frequency as commonly found in microprocessor clock systems.
2. The distinction between correlators is that the correlation length of a tracking correlator is usually the length of the PN code whereas the correlation length of the data (demodulating) correlator is always the length of a data bit, T_b.



(a) ahead (b) aligned (c) behind

Figure 3-20 Early and Late correlator output values for different timing alignments; (a) internal PN code is ahead of the received input signal; (b) aligned to input signal; (c) behind input signal.

lation on three successive chip phases of the input signal spaced $0.5 \cdot T_c$ apart, so when the Early and Late output amplitudes are equal, the Data correlator will be perfectly aligned to the input signal and will produce the peak output value.

In the ideal DT case, the autocorrelation function's rectangular characteristic (Fig. 3-10) modifies the tracking characteristics. In this case (see Fig. 3-21), since the spacing of the Early and Late correlations is exactly the same as the width of the $C_{DA}(\tau)$ characteristic, i.e., one T_c, the DLL can never make the Early and Late output values equal. This causes the loop to *dither* or toggle back and forth between being ahead or behind the ideal alignment. It is insightful to look at the consequence of this rectangular characteristic on the of sampling of the PN modulated input stream. Fig. 3-22 illustrates the relative sampling positions for the Early, Data and Late correlators for three conditions. When the Early correlator is sampling the correct chip, the Late correlator is sampling the incorrect chip, and vice versa, giving rise to the rectangular characteristic. In the

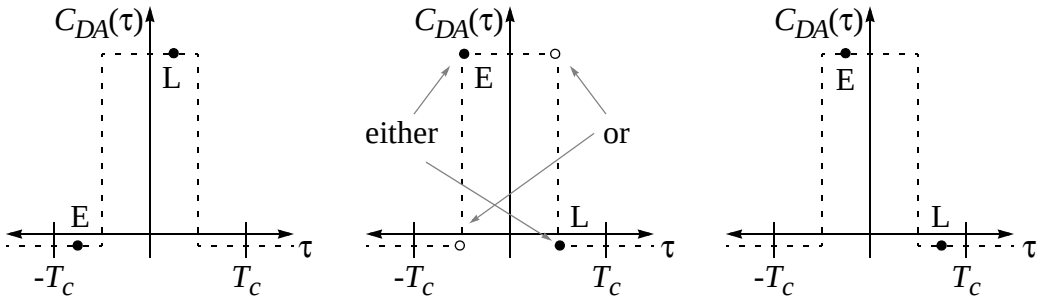


Figure 3-21 Discrete-time Early and Late correlation outputs.

case of perfect alignment both the Early and Late correlations may be close to zero since they fall in the signal transition region so the tracking algorithm should take steps to maintain alignment lock in this event.

It appears that because of the flat-top characteristic of $C_{DA}(\tau)$ that this dithering should not affect the Data correlation value, but in practice it does introduce a *dithering error* due to finite sampling bandwidth.

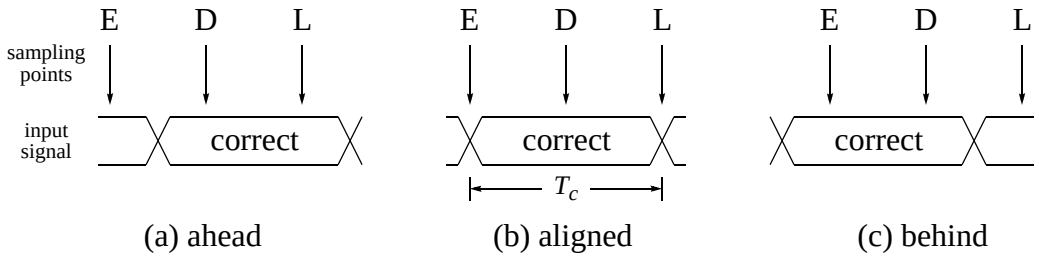


Figure 3-22 Relative sampling points for Early, Data and Late correlators.

3.7 References

[1]. R. C. Dixon, *Spread Spectrum Systems*, 3rd ed. New York, USA: John Wiley & Sons, 1994.

[2]. R. L. Pickholtz, D. L. Schilling, and L. B. Milstein, "Theory of spread-spectrum communications--a tutorial," *IEEE Trans. Commun.*, vol. COM-30, pp. 855-884, May 1982.

[3]. R. L. Pickholtz, L. B. Milstein, and D. L. Schilling, "Spread spectrum for mobile communications," *IEEE Trans. Veh. Technol.*, vol. 40, pp. 313-322, May 1991.

[4]. S. Sheng, L. Lynn, J. Peroulas, K. Stone, and I. O'Donnell, "A low-power CMOS chipset for spread-spectrum communications," *ISSCC Digest of Technical Papers*, Feb. 1996, pp. 346-347.

[5]. D.T. Magill, T. G. Kirke, ”Optimal short-range CDMA codes of length 16 for wireless communications,” *1992 IEEE International Conf. on Selected Topics in Wireless Communications*. Conference Proceedings, Vancouver, BC, Canada, June 25-26, 1992.

Chapter 4

Design Considerations of Analog Correlation

This chapter will look at ways to implement an analog correlator in a very power efficient manner. First a typical switched-capacitor (SC) architecture will be examined, followed by a lower power yet area inefficient version, finally arriving at a *passive* structure with even lower power dissipation. This is followed by a discussion of the design issues that must be addressed for correct operation by a *passive* correlator. At the end of the chapter non-ideal issues that accompany SC designs will be addressed.

4.1 Minimum Power Architecture

4.1-1 Active chip-rate correlator

Before looking at SC correlators, a choice between a *matched filter* or *sliding correlator* architecture (see Fig. 4-1) needs to be made. Because of its much lower power consumption,[1] the sliding correlator is preferred despite its slower acquisition performance.[2]

1. The matched filter is roughly *N*-1 times higher power since it require *N*-1 times more stages (opamps).
2. In the prototype, both the Early and Late correlators were used during acquisition which improved the synchronization time by a factor of two.

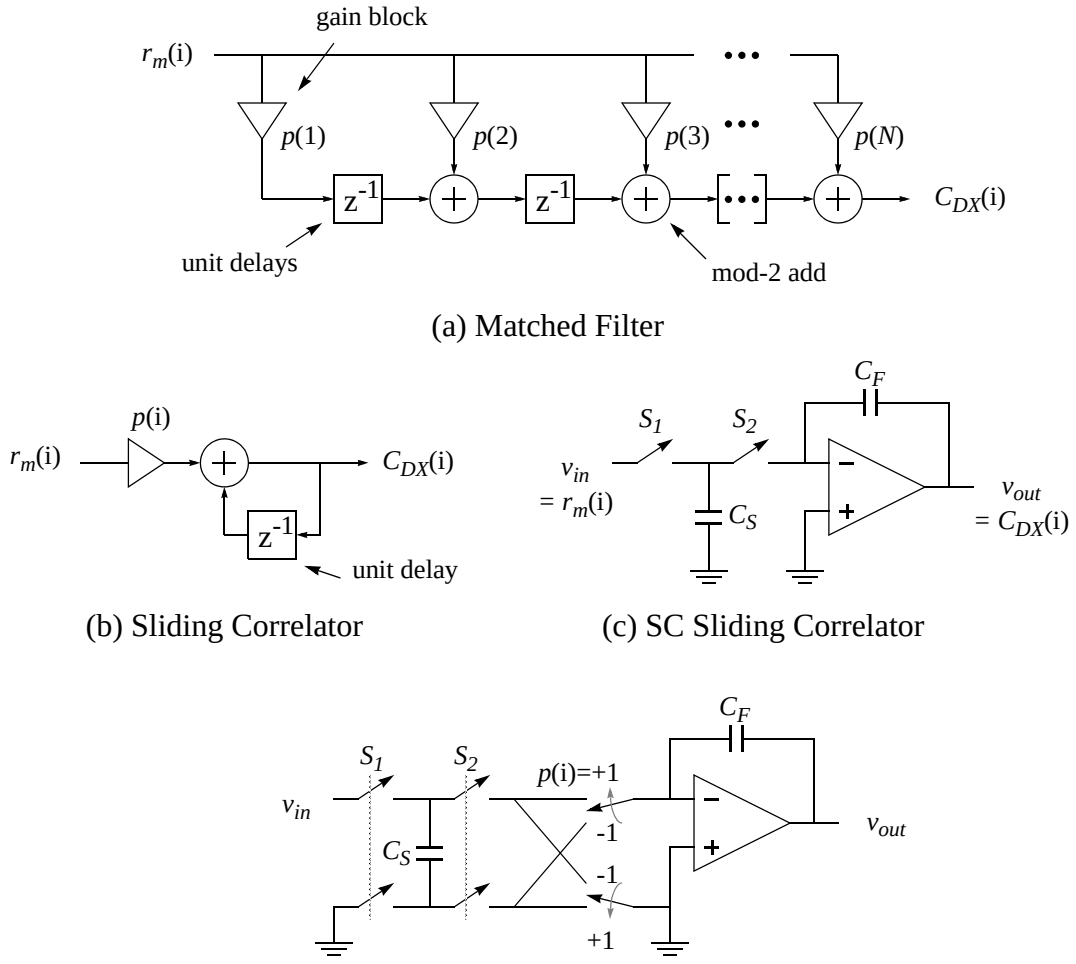


Figure 4-1 Analog correlator architectures and SC implementations.

Fig. 4-1(c) shows a conceptual schematic of how a SC integrator can implement a sliding correlator (neglecting the PN modulation). Recalling that PN modulation is simply *sign modulation*,[1] Fig. 4-1(d) adds the PN modulation for a complete schematic of the sliding correlator. A few notes on its operation; first, the samples are being captured and inte-

1. For PN bit values of {−1,+1}.

grated at the ***chip rate***, thus the opamp must settle within a fraction of a chip time (T_c). Second, the sliding correlator is an integrate-and-dump process, where after N samples are integrated, the output is evaluated and then the integrating feedback capacitor (C_F) is reset ("charge is dumped"), thus with proper scaling integration overflow is not a problem.

4.1-2 Active data-rate correlator

The problem with this *chip-rate* correlator is that for high chip rates, e.g. 100*Mc/s*, the opamp has to settle to the desired resolution within a fraction of T_c, e.g. 5*ns*, which in turn means an opamp with a time-constant near 1*ns* or a gain-bandwidth exceeding 10*GHz* with even minimal gain. Such an opamp would dissipate a large amount of power with today's technologies. Another approach is to implement the sliding correlator as shown in Fig. 4-2. In this configuration the single sampling capacitor is now replaced by two banks of N capacitors that alternate between the sampling and integrating operations. During the sampling operation, N successive input samples are stored sequentially on the N capacitors

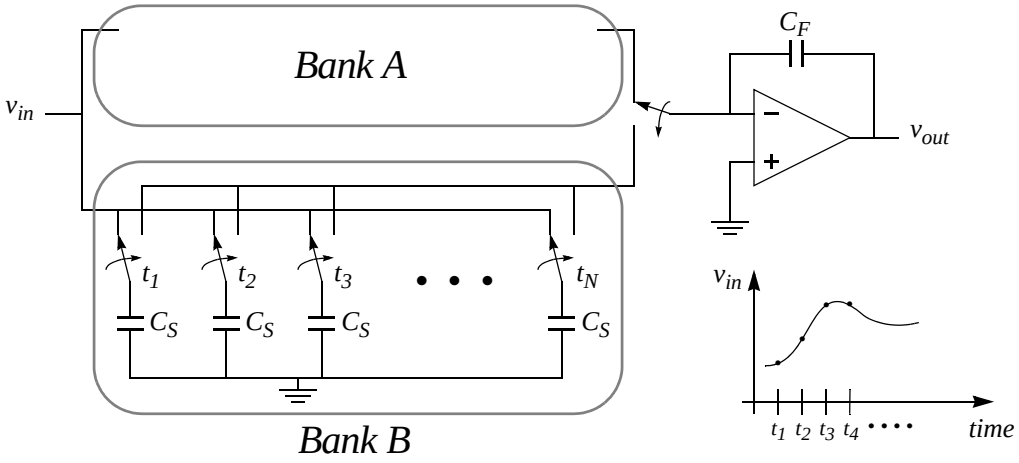


Figure 4-2 Active data-rate correlator; banks alternate sampling and integrating.

of one bank so that sampling is still performed at the chip rate. This allows continuous sampling of the input while permitting the opamp $2{\cdot}N$ times longer to settle.[1] Since processing one data bit requires N samples, the opamp is integrating at the data rate rather than the chip rate, hence the term data-rate correlator.

Although it appears that this greatly relaxes the settling time requirement of the SC integrator, only a direct factor of two improvement in settling behavior is realized because the sampling capacitor[2] has increased by a factor of N. However, due to a MOSFET's somewhat square-law characteristic, power savings of upwards to four fold may be possible depending on the proximity to long-channel operation. Thus this architecture does yield some power improvement over the chip-rate correlator but at a disproportionately high cost in area.

4.1-3 Passive data-rate correlator

Further power reduction is possible by a modification that follows directly from the active data-rate correlator of Fig. 4-2. Unlike that structure, where integration is performed by transferring the signal charge of each sampling capacitor to the integrating capacitor, C_F, via an opamp, this correlator integrates the charge by simply connecting the sampling capacitors together passively, i.e., without an opamp (see Fig. 4-3). Thus this implementation can be called a *passive data-rate correlator.* The following observations can be made about this passive approach:

1. The opamp has N full periods to settle instead of half a sample period as in the *chip-rate* correlator.
2. The settling time-constant of a SC integrator is proportional to $1/gm$ and roughly proportional to C_S, especially if C_S is large compared to C_F and other capacitors.

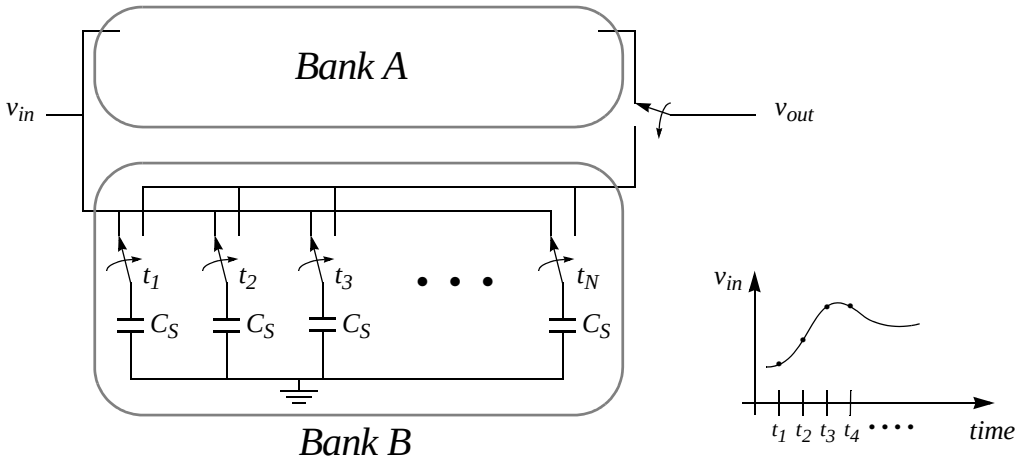


Figure 4-3 Passive data-rate correlator; integration without an opamp.

1. No integrating capacitor is needed. The combination of all the sampling capacitors forms the integrating capacitor.

2. No opamp is needed. Switches perform the charge transfer.

3. No settling time restriction by limited opamp bandwidth.

4. No opamp static power dissipation.

Hence, this new *passive* architecture is both faster and lower in power (only dynamic power).

4.2 Passive SC Integration

The cost of this power/speed improvement is twofold; a charge-error cancellation scheme is required and the gain is restricted to a value that is less than unity (as is common for passive structures). In active SC integrators, the opamp prevents charge errors from

corrupting the summing node that contains the signal charge. In the passive integrator, any error charge deposited on the summing node can be cancelled by proper sequencing of the sampling and additional *zeroing* switches. In the process of removing this error charge some signal charge is destroyed and results in a signal strength reduction. Being passive, the correlator itself cannot restore or amplify the signal which results in a less-than-unity gain that is determined by the ratio of sampling to parasitic capacitances. The next sub-sections covers the details involved with SC sampling and integration.

4.2-1 Charge errors in active SC integration

Before considering charge errors in the passive SC integration approach, it may be helpful to review what happens to these errors in an *active* SC approach. Consider the simple SC integrator of Fig. 4-4. The input signal is applied to the sampling capacitor (C_S) when switch S_1 is closed and a sample of the input is captured on C_S when S_1 opens. Next S_2 is closed and the signal charge from C_S is transferred onto the feedback integrating capacitor (C_F). Ignoring the parasitic capacitors, C_{p1} & C_{p2}, and assuming an ideal opamp, the change in the output voltage will be the change in the voltage across C_F due to

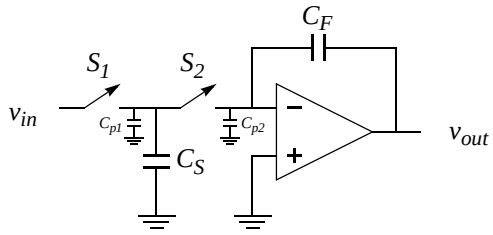


Figure 4-4 Single-ended *active* SC integrator

the signal charge, q_S. Thus:

$$\Delta v_{out} = -\frac{q_S}{C_F} = -v_S \cdot \frac{C_S}{C_F} \tag{4 - 1}$$

where v_S is the sampled value of v_{in} that is stored on C_S prior to integration. The minus sign is due to the inverting configuration of the opamp. Thus the ideal closed-loop gain for this integrator is:

$$A_{CL} = -\frac{C_S}{C_F} \tag{4 - 2}$$

Now the parasitic capacitors, which may include the source/drain diffusion-to-bulk capacitance of the switches, the parasitic peripheral capacitance of C_S[1] or any stray metal/poly capacitance, are added to the analysis. The input sample is now captured on both C_S and C_{p1} so subsequent integration will yield:

$$\Delta v_{out} = -\frac{q_S}{C_F} = -v_S \cdot \frac{C_S + C_{p1}}{C_F} \tag{4 - 3}$$

and thus:

$$A_{CL} = -\frac{(C_S + C_{p1})}{C_F} = -\frac{C_S}{C_F} \cdot \left(1 + \frac{C_{p1}}{C_S}\right) \tag{4 - 4}$$

So C_{p1} adds a gain error term relative to the value of C_S making the gain *parasitic sensi-*

1. This peripheral capacitance is the parasitic capacitance that is unavoidably formed by the lower plate of the capacitor to substrate or well, as such it is usually some constant percentage of the main capacitance.

tive. Since C_{p1} is partly composed of non-linear junction capacitance (source/drain to bulk of the switches), this error, and thus the gain itself, has a voltage dependence component. While increasing C_S will reduce this error,[1] nonetheless, the non-linearity will give rise to harmonic distortion. (Notice that C_{p2} does not cause any error since its voltage is unchanged before and after the integration of q_S.)

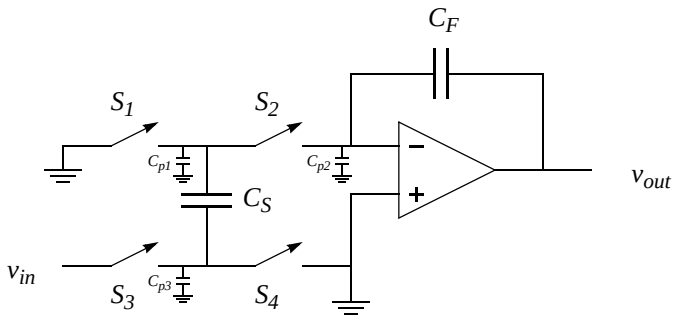
4.2-2 Bottom-plate sampling

The way to avoid this problem and achieve *parasitic* ***insensitive*** gain, is to use *bottom-plate sampling* [1]. This is accomplished by using the configurations of Fig. 4-5. Fig. 4-5(a) is the configuration used when ***continuous integration*** is needed as in sigma-delta ADC's [2] and SC filters [3]. In contrast, Fig. 4-5(b) is used for ***integrate-and-dump*** type operations as in pipelined ADC's [4] and other sample-and-hold applications. In both configurations, it is the bottom plate that captures the input sample rather than the top plate of C_S (S_1 and S_3 closed). Consequently it is C_{p3} rather than C_{p1} that also captures the input signal, however the charge on C_{p3} is shorted to ground by S_4 during the integration period instead of being included with q_S (see Fig. 4-5(c)).

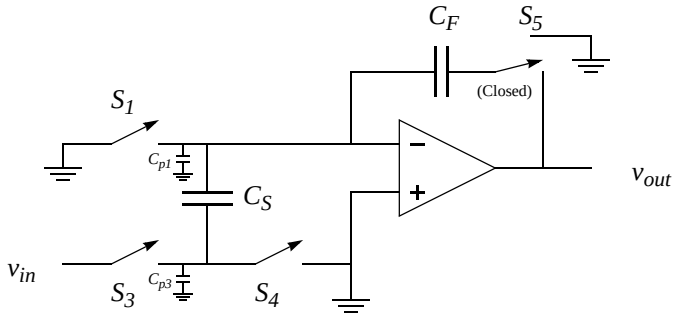
Notice that in contrast to equation (4 - 2), bottom-plate sampling inverts the sampled charge, q_S, resulting in a *positive* closed-loop gain of:

$$A_{CL} = \frac{C_S}{C_F} \qquad (4 - 5)$$

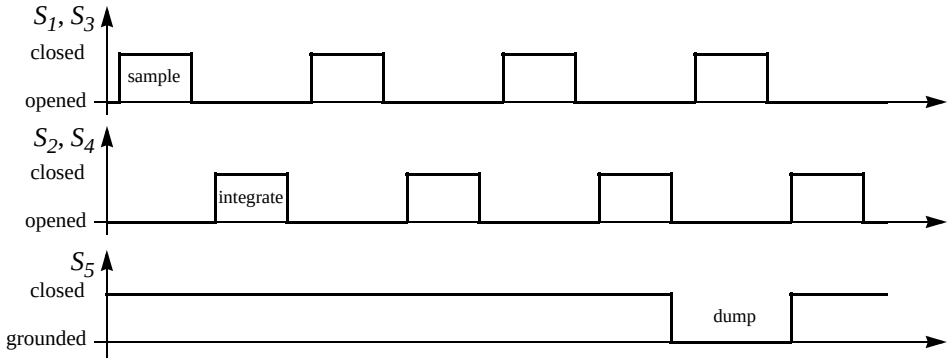
1. To the extent that C_{p1} is independent of C_S



(a) Continuous SC Integrator



(b) “Integrate-and-dump” SC Integrator



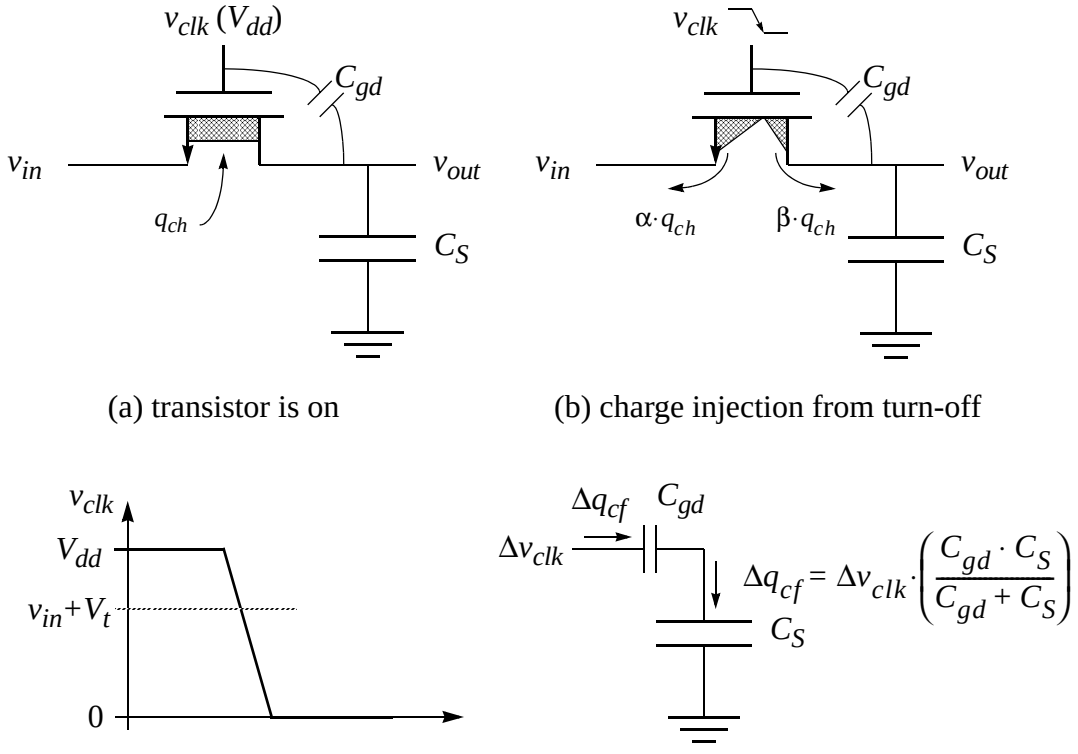
(c) Timing diagram (2-phase non-overlapping clocks)

Figure 4-5 Bottom-plate sampling for parasitic insensitivity.

Note also that the voltage across C_{p1} and C_{p2} is the same before and after the integration cycle so they do not contribute any error charge to C_F, assuming an ideal opamp.

4.2-3 Charge injection and clock feed-through errors

Charge errors associated with MOS transistors that are used as switches in SC integrators are addressed next. This is best illustrated with a single NMOS transistor[1] and capacitor as shown in Fig. 4-6. Charge injection refers to that portion of the switch's channel charge, q_{ch}, that is injected onto the sampling capacitor when the switch is turned off (Fig. 4-6(b)). The clock feed-through is due to the coupling of charge via the parasitic



(a) transistor is on (b) charge injection from turn-off

(c) clock falling-edge waveform (d) clock feed-through

Figure 4-6 Charge injection and clock feed-through

1. For simplicity, this discussion uses NMOS switches, but everything applies equally to PMOS or CMOS switches.

gate-to-drain capacitance of the transistor switch when the clock makes a transition (Fig. 4-6(d)).

For charge injection, the channel charge is:

$$q_{ch} = W \cdot L \cdot C_{ox} \cdot ((v_{in} + V_t) - V_{dd}) \qquad (4 - 6)$$

Due to its dependence on v_{in}, injection of q_{ch} is also known as ***signal-dependent*** charge injection. When the switch is turned off, some of this channel charge is injected back into the input and some onto the sampling capacitor. How much is injected depends on the relative impedance that the switch sees looking into C_S and v_{in}, and the fall time, t_f, of the clock signal [5]. In [5], it was found that for slow fall times, a higher fraction of q_{ch} flows into the lower impedance node. As the fall times get faster, the channel collapses quicker and q_{ch} divides more evenly between the two nodes (from Fig. 4-6(b), $\alpha = \beta$) independent of their impedances. In the typical situation where the input impedance is lower than the impedance of the sampling capacitor, for slow fall times, the channel exists longer, allowing the channel charge to flow back into the input, further decreasing the amount of injected charge[1] onto C_S. Thus for a low impedance input, the charge injection onto C_S for two limiting cases of fall times are:

1. Extremely slow t_f: $\beta \cdot q_{ch} = 0$
2. Extremely fast t_f: $\beta \cdot q_{ch} = 0.5 \cdot q_{ch}$

1. Note that for a NMOS switch, the channel charge is negative so that the net charge on C_S will decrease (assuming q_S is positive) when the switch turns off.

The actual magnitude of the charge injection falls somewhere between these two values and is dependent on the clock fall time.[1]

The clock feed-through charge, q_{cf}, that is deposited onto C_S is the same amount that is deposited on the top-plate of C_{gd}, by Kirchoff's current law, so it is determined by the equivalent capacitance seen by Δv_{clk}, i.e., the series combination of C_{gd} and C_S (Fig. 4-6(d)). However, it too is modulated by the speed of the fall time.[2] For slow t_f, while V_{clk} is higher than $v_{in}+V_t$, the input will absorb any Δq_{cf} that is deposited on C_S, thus only for V_{clk} lower than $v_{in}+V_t$ will charge be coupled onto C_S. When t_f is fast, the input will not have time to absorb the charge so all the charge will get coupled to the sampling capacitor. This can be summarized as:

1. Extremely slow t_f: $$\Delta q_{cf} = -(v_{in} + V_t) \cdot \left(\frac{C_{gd} \cdot C_S}{C_{gd} + C_S} \right)$$

2. Extremely fast t_f: $$\Delta q_{cf} = -V_{dd} \cdot \left(\frac{C_{gd} \cdot C_S}{C_{gd} + C_S} \right)$$

Combining these results yields the following:

1. Extremely slow t_f:

1. If the input impedance is higher than the impedance of C_S then it is possible that the charge injection onto C_S can be higher than half the channel charge according to [5] but this is not usually the case.
2. Even though there is also clock feed-through on the rising edge of the clock, it has no affect since once the switch is turned on, the input signal will determine the value on C_S.

$$v_{out} = \frac{q_S + \Delta q_{cf}}{C_S} = v_{in} \cdot \left(1 - \frac{C_{gd}}{C_{gd} + C_S}\right) - V_t \cdot \left(\frac{C_{gd}}{C_{gd} + C_S}\right) \qquad (4 - 7)$$

2. Extremely fast t_f:

$$v_{out} = v_{in} \cdot \left(1 + \frac{W \cdot L \cdot C_{ox}}{2 \cdot C_S}\right) - V_{dd} \cdot \left(\frac{W \cdot L \cdot C_{ox}}{2 \cdot C_S} + \frac{C_{gd}}{C_{gd} + C_S}\right) + V_t \cdot \left(\frac{W \cdot L \cdot C_{ox}}{2 \cdot C_S}\right) \qquad (4 - 8)$$

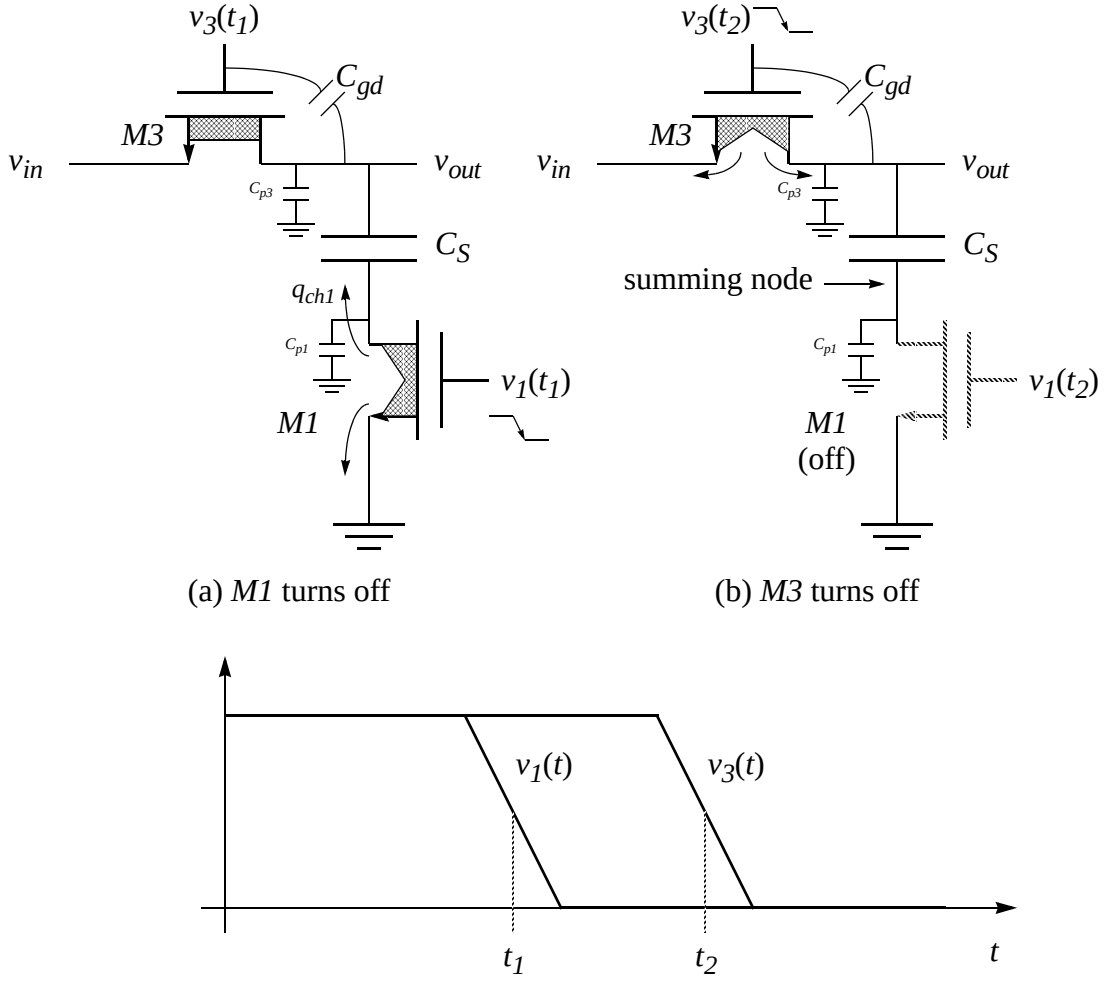
The first term in both cases represents a gain error, where ideally the gain should be unity. The latter terms represent offset errors. Since the actual values for both the gain and offset errors vary between these two extremes, depending on the speed of the falling clock edge, the unpredictability of these errors requires a method for their cancellation or avoidance if accurate SC integration is to be achieved.

4.2-4 Avoidance of signal-dependent charge injection

[6] and [7] present a technique for avoiding the signal-dependent charge injection by adding another switch and delaying the turn-off of the input switch (see Fig. 4-7). In this technique, when the new switch (*M1*) is turned off first, it injects a charge, q_{ch1}, onto the sampling capacitor equal to (assuming a fast falling edge for $v_1(t)$):

$$q_{ch1} = \frac{1}{2} \cdot W \cdot L \cdot C_{ox} \cdot (V_t - V_{dd}) \qquad (4 - 9)$$

which is independent of the input signal and can be negated by using a differential archi-



(a) *M1* turns off

(b) *M3* turns off

(c) falling edge of $v_3(t)$ delayed from sampling clock, $v_1(t)$

Figure 4-7 Charge injection error avoidance by turn-off delay of input switch.

tecture as described later. Once *M1* is off, the ***net*** charge on the "summing node" (see Fig. 4-7(b)) cannot be disturbed by any changes in the input circuit, preventing corruption of the sampled signal charge. Thus despite changes by the input signal after *M1* turns off but before *M3* turns off, or charge injection and clock feed-through when *M3* turns off (which both cause charge to be transferred between C_S and C_{cp1}) the ***net*** charge at the

summing node remains ***unchanged***. This net charge, q_{sn}, is:

$$q_{sn} = -q_S + q_{ch1} + q_{cf1} = -q_S + Q_1 \qquad (4 - 10)$$

where q_S is the signal charge and Q_1[1] is the sum of injected and clock feed-through charge due to the turn off of *M1*.

To understand how q_S can be recovered, the bottom-plate sampling circuit of Fig. 4-5(b) is used, except redrawn as shown in Fig. 4-8. During the sampling period, *M1* turns off first, effectively isolating the summing node and preserving the net charge, q_{sn},

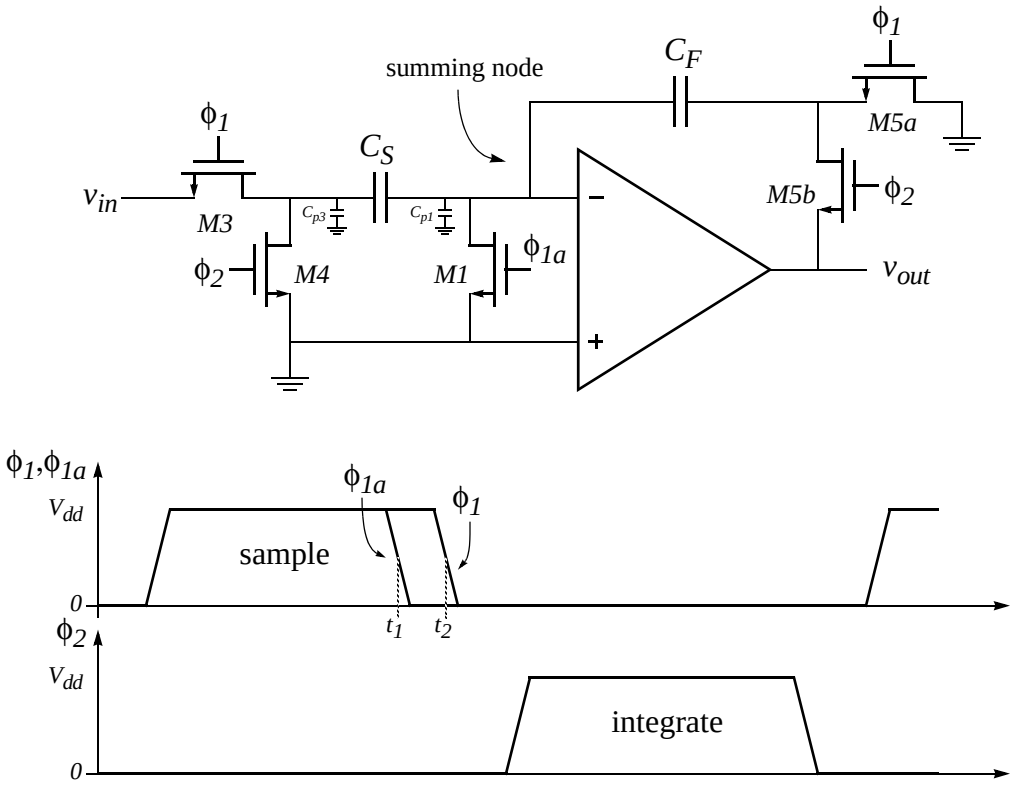


Figure 4-8 Bottom-plate sampling with input switch turn-off delayed.

1. Note that Q_1 is negative in this instance.

as given in (4 - 10) (refer back to Fig. 4-7). Although there is an additional capacitance, C_F, attached to the summing node, it does not add any charge to the node since it is shorted out prior to turning off *M1*.[1] As before, when ϕ_1 turns off *M3* (and now *M5a*), there is charge transfer between C_S and C_{p1} (and C_F) due to charge injection and clock feed-through but no change in the net charge at the summing node. Similarly, when the integration cycle begins by turning on *M4* & *M5b* and proceeds to completion, charge is redistributed but no change in the net charge at the summing node occurs.

At the end of the integration cycle, the opamp (assumed ideal) forces the summing node to zero volts through the feedback connection of C_F, resulting in the transfer of all the charge on C_S and C_{p1} onto C_F. The output voltage is then:

$$v_{out} = \frac{q_S - Q_1}{C_F} = v_{in}(t_1) \cdot \left(\frac{C_S}{C_F}\right) - \frac{Q_1}{C_F} \qquad (4 - 11)$$

where:

$$Q_1 = q_{ch1} + q_{cf} = -\frac{WLC_{ox} \cdot (V_{dd} - V_t)}{2} - V_{dd} \cdot \left(\frac{C_{gd} \cdot C_S}{C_{gd} + C_S}\right) \qquad (4 - 12)$$

In (4 - 11), the first term is the desired gain component and the second term is an offset error due to charge when *M1* turns off, that need to be eliminated in order to obtain an accurate result.

1. Assuming an integrate-and-dump scheme.

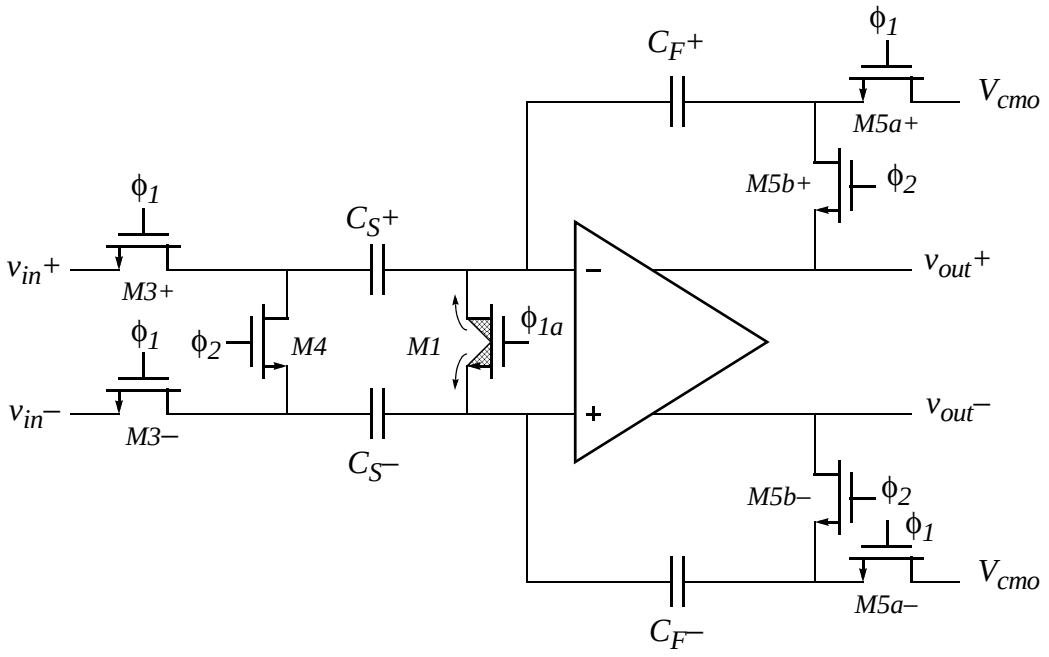


Figure 4-9 Differential SC integrator

4.2-5 Elimination of charge offset errors

Cancellation of charge offset errors can be accomplished by using a differential structure as in Fig. 4-9, that transforms this offset error into a common-mode signal, effectively eliminating it. Before considering Q_1, a few observations about the differential integrator should be made. Unlike the single-ended version, the output side of C_F is now connected to the output common-mode voltage (V_{cmo}) of the differential amplifier during ϕ_1 rather than ground. And although not shown (for the sake of clarity), the input side of C_F would normally be clamped to the input common-mode voltage (V_{cmi}) of the amplifier during ϕ_{1a} via two additional matched switches. This implies that C_F would contribute additional charge to the summing node if V_{cmi} and V_{cmo} were not equal as assumed in the single-ended case. While this is true, since this charge is common to both the plus and

minus sides of the integrator, it is a common-mode signal and hence does not affect the differential output.

This same argument can be applied to the components of Q_1. From (4 - 12), it appears that the clock-feed-through component, q_{cf}, is identical for both sides of the integrator and thus represent a common-mode entity. And while this assumes fast fall times, it is true even for slower fall times provided that the overlap capacitances of *M1* are equal[1] and C_S+ matches C_S–.

The charge injection component, βq_{ch}, is also a function of fall time,[2] but again relying on symmetry and matching, it can yield a common-mode charge contribution. However, as pointed out in [7], when the fall time is not fast enough to guarantee a 50% split, the relative impedance on both sides of *M1* will determine the charge injection mismatch. The dominant impedance mismatch can be attributed to the input (*M3*) switches' "on" resistances (R_{on}) which will be unequal due to dissimilar source voltages (v_{in}+ and v_{in}–). [7] offers one way to minimize this effect by using a transmission gate (CMOS switch) instead of a single transistor switch which has less variation in R_{on} as the source voltage varies. It is hard to quantify this error since it depends on the value of t_f, v_{in} and matching of *M1* but it is greatly reduced from an absolute error (single-ended case) to a mismatch error, and eliminated completely if *M1* is turned off fast enough. The mismatch injected charge error of the common-mode input switches (not shown in Fig. 4-9 but connected to

1. This is best accomplished by making *M1* out of two identical cross-coupled transistors.
2. β of $\frac{1}{2}$ is used in (4 - 12) due to a fast fall-time assumption.

the summing nodes and V_{cmi}) can be eliminated by adding another (falling edge) clock phase prior to ϕ_{1a}, to turnoff those switches as described in [8].

4.2-6 Isolation of the summing nodes

Before ending this section, a point needs to be highlighted about the importance of isolating the summing node from noise coupling or other extraneous charge perturbations. Charge injection avoidance relies on isolating and thus preserving the charge on the summing node. The most common way to corrupt this net charge is by capacitive coupling from stray signal, supply or clock lines and the substrate (bulk). Shielding by a "clean" (dedicated non-supply) ground line for the summing node traces and a bottom-plate shield[1] under the capacitors (C_S and C_F) will preserve the integrity of the summing node. Fortunately, another benefit of the differential integrator configuration is that substrate coupling usually occurs as a common-mode signal to the differential summing nodes (if they are close to each other) so its impact is greatly reduced.

By taking the above precautions, the above techniques work well enough to reduce the charge errors to levels suitable for 13 bit A/D converters [8],[9].

4.2-7 Charge errors in passive SC integration

In a passive sampling and integration scheme, many of the same considerations apply as in the active scheme. The main difference is in the way the integration is performed; by transferring charge onto an integrating capacitor, C_F, from the sampling capac-

1. A N-well, P-well or lower layer metal tied to "clean ground" depending on the type of capacitor.

itor, C_S, using an opamp (in the active approach), versus equilibrating charge on the sampling capacitors (in the passive approach). From a charge error perspective, the lack of an opamp implies the loss of a feedback mechanism to force the summing node back to a well-defined potential (virtual ground) and thus preventing the reversal of any charge-error transfers that occurred at the summing node. The passive approach compensates by employing an additional switch and "zeroing" sequence to obtain a similar result.

Fig. 4-10 shows a single-ended version of a passive integrator. Observe that the clock signals are different for each set of sampling cells. Unlike the active SC integrator, the sampling capacitor will hold the integrated signal charge so both of its terminals cannot be grounded. Thus, at least C_{p1}, or C_{p3}, will hold some charge that is included as part of the final integrated charge. This implies that the integration result will be dependent on at least one of these parasitics so there is less benefit from bottom-plate sampling. However, it is still desirable to eliminate signal-dependent charge injection and clock feed-through, by delaying the turn off of the input switch, *M3*, until after the sampling switch, *M1*, has isolated the summing node [7] with the configuration shown in Fig. 4-10. This structure by itself, is not sufficient for eliminating the charge errors, but is useful for explaining the passive integration process. A simple modification to the structure will allow cancellation of the errors as described later.

Consider the operation of a single cell (see Fig. 4-11). As in the active case, a sample is taken when *M1* turns off (which will be denoted as time t_1). In the process, *M1* injects an error charge, Q_1, so the net charge, q_{n1}, at node 1 is:

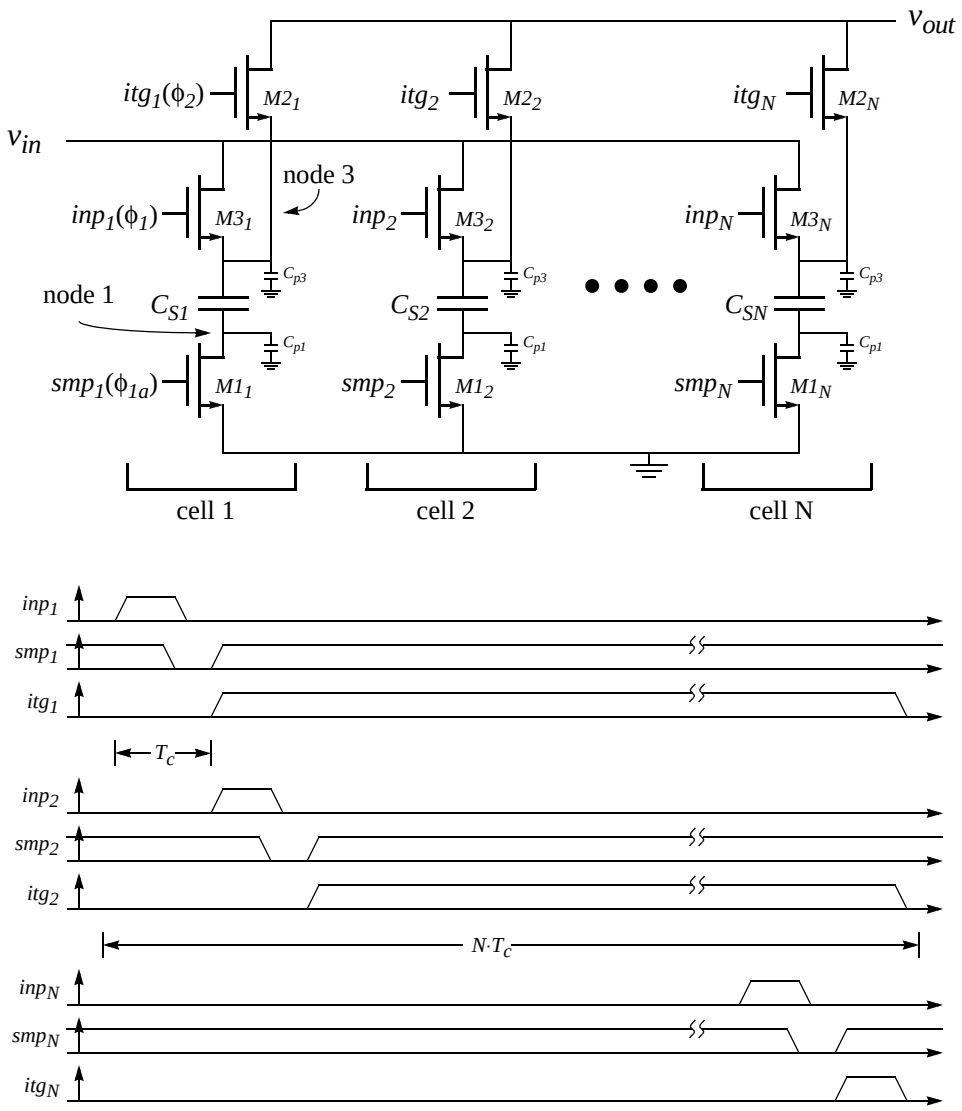


Figure 4-10 Single-ended passive integrator and timing waveforms.

$$q_{n1} = -q_S + \beta_1 \cdot q_{ch1} + q_{cf1} = -q_S + Q_1 \tag{4 - 13}$$

and the charge, q_{Csb}, on the bottom plate of C_S is:

$$q_{Csb} = -q_S + Q_1 \cdot \left(\frac{C_S}{C_S + C_{p1}} \right) \tag{4 - 14}$$

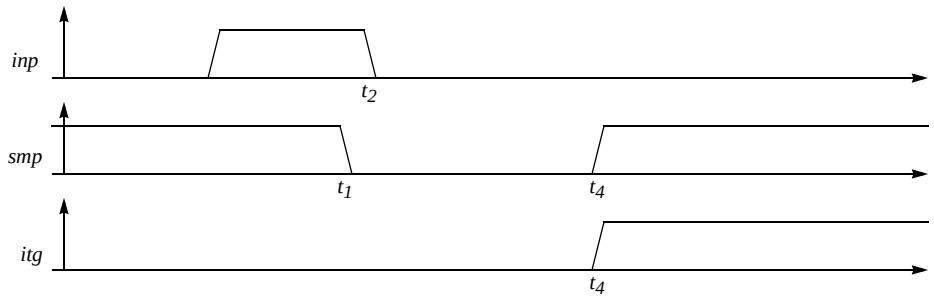


Figure 4-11 Expanded view of one timing cycle in Fig. 4-10.

After a delay, *M3* opens, disconnecting the input from C_S at time t_2. In addition to *M3*'s turn-off charge, charge is deposited on C_{p3} and the series combination of C_S and C_{p1} due to a change in the input signal during the delay between the turn-off of *M1* and *M3*. So the resulting charge on the top plates of C_S and C_{p3}, q_{Cst} and q_{Cp3t}, respectively, are:

$$q_{Cst} = q_S - Q_1 \cdot \left(\frac{C_S}{C_S + C_{p1}} \right) + \Delta v_{in} \cdot C_S' + Q_3 \cdot \left(\frac{C_S'}{C_S' + C_{p3}} \right) \qquad (4 - 15)$$

$$q_{Cp3t} = v_{in}(t_1) \cdot C_{p3} + \Delta v_{in} \cdot C_{p3} + Q_3 \cdot \left(\frac{C_{p3}}{C_S' + C_{p3}} \right) \qquad (4 - 16)$$

$$= v_{in}(t_2) \cdot C_{p3} + Q_3 \cdot \left(\frac{C_{p3}}{C_S' + C_{p3}} \right)$$

where Q_3 is the turn-off charge of *M3* and:

$$\Delta v_{in} = v_{in}(t_2) - v_{in}(t_1) \qquad (4 - 17)$$

$$C_S' = \frac{C_S \cdot C_{p1}}{C_S + C_{p1}} \qquad (4 - 18)$$

Once *M3* is off, node 3 is isolated and the net charge at that node is preserved. That net charge, q_{sn3}, is:

$$q_{sn3} = q_{Cst} + q_{Cp3t} \tag{4 - 19}$$

$$= q_S - Q_1 \cdot \left(\frac{C_S}{C_S + C_{p1}} \right) + Q_3 + v_{in}(t_2) \cdot C_{p3} + Q_{e5}$$

where:

$$Q_{e5} = \Delta v_{in} \cdot C_S' \tag{4 - 20}$$

In (4 - 19), the second term can be minimized or eliminated entirely by using a differential structure as in the active integration case. However, the last three terms are dependent on the input signal at time t_2 and represent an uncorrelated error to the signal charge, q_S, that was sampled at t_1 and must be cancelled before the charge integration step at t_4.

4.2-8 Zeroing sequence

Fig. 4-12 shows a single cell of a differential passive integrator along with the associated timing signals. Transistor *M4* is added to the integrator and a new clock pulse is added to the cell timing between the sampling (t_2) and integrating (t_4) intervals. This zeroing clock pulse will turn on switch *M4*, effectively cancelling the charge represented by the last three terms of (4 - 19), converting them to common-mode signals. In the case of the charge on C_{p3}+ and C_{p3}– (fourth charge term), shorting them together clearly equates their voltages and cancels any differential charge, but how the third, Q_3,[1] and fifth, Q_{e5},

1. To be precise, some of Q_3 is on C_{p3}, so this pertains to that portion of Q_3 on C_S.

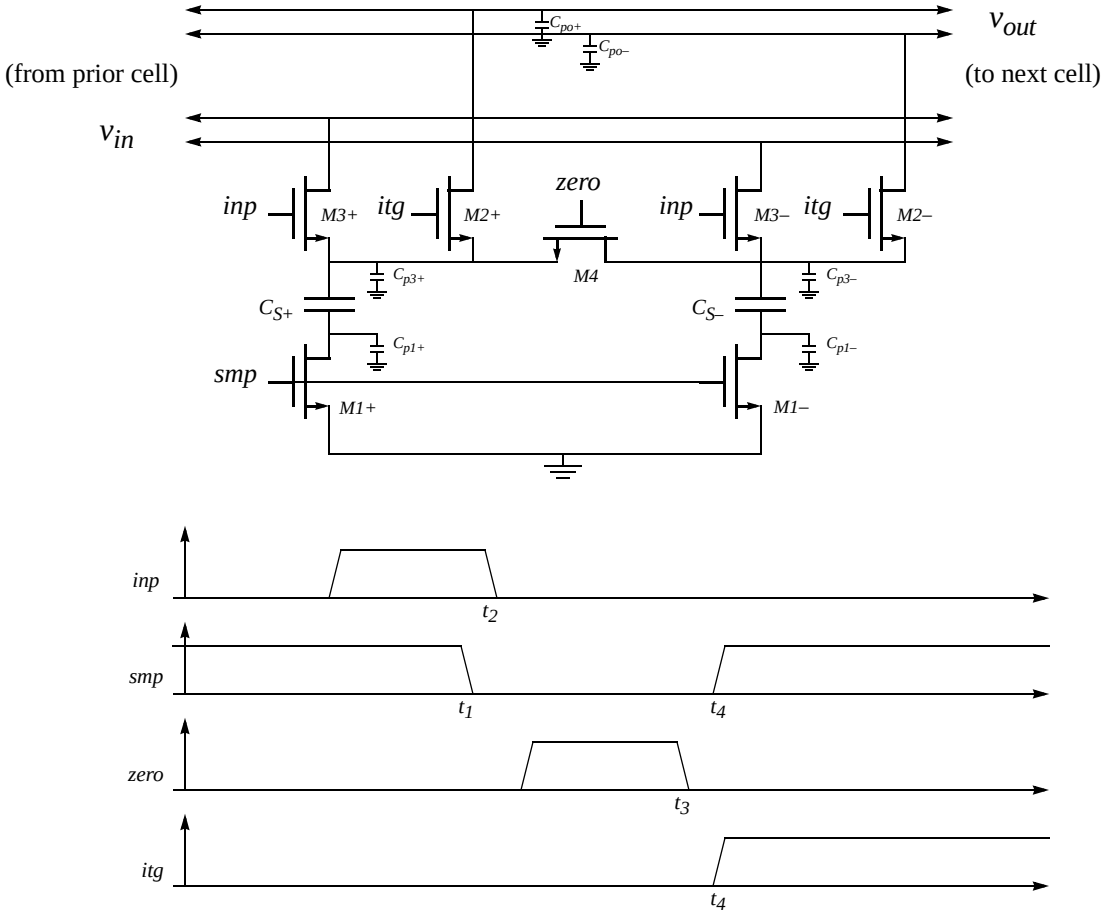


Figure 4-12 One cell of a differential passive integrator with timing diagram.

terms are cancelled is not as clear to see.

How Q_3 and Q_{e5} can be cancelled without destroying the signal charge, q_S, relies on the fact that q_S is isolated and therefore preserved on the bottom-plate of C_S and top-plate of C_{p1} (node 1) despite changes on the top-plate of C_S (node 3). Consider the diagram of Fig. 4-13, which is the resulting circuit when *M4* closes, excluding C_{p3}. Intuitively, if V_{3+} and V_{3-} were grounded, then V_{1+} and V_{1-} would only reflect the net charge at the respective node 1's, capacitively divided between C_S and C_{p1}; thus q_{3+} and q_{3-} would not con-

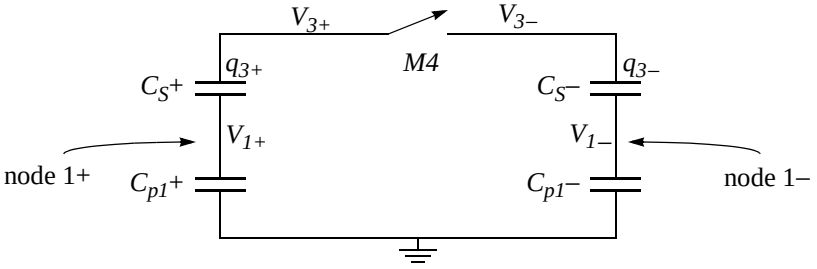


Figure 4-13 Cancelling charge errors prior to passive integration.

tain any error charge (Q_3 or Q_{e5}). If V_{3+} and V_{3-} were equal and set to some voltage (V_z), q_{3+} and q_{3-} would contain an additional but equal charge, q_{cm}:

$$q_{cm} = V_z \cdot \left(\frac{C_S \cdot C_{p1}}{C_S + C_{p1}} \right) \tag{4 - 21}$$

and thus not altering the differential signal component, q_S. So by simply making V_{3+} equal to V_{3-}, the differential charge, q_{df}, on $C_S\pm$ is restored to a charge proportional to q_S:

$$q_{df} = q_{3+} - q_{3-} = 2 \cdot q_S \cdot \left(\frac{C_S}{C_S + C_{p1}} \right) \tag{4 - 22}$$

This result is derived in Appendix B.

The final step is to turn on *M1* and *M2*, thereby integrating the signal charge with the other samples. Due to three parasitic capacitances, the signal charge is once more attenuated. The three parasitic capacitors are C_{p3}, C_{po} (per cell output bus capacitance)[1] and

1. For a correlator, *N* cells in length, C_{po} is the total output bus capacitance divided by *N*.

C_{ox2} (gate-oxide capacitance of *M2*). The final differential charge, q_{df}', is:[1]

$$q_{df}' = 2 \cdot q_S \cdot \left(\frac{C_S}{C_S + C_{p1}} \right) \cdot \left(\frac{C_S}{C_S + C_{p3} + C_{po} + C_{ox2}} \right) \qquad (4 - 23)$$

With *M1* and *M2* on, the corresponding output differential voltage, v_{out}, for the cell is:

$$\begin{aligned} v_{out} &= \frac{q_{df}'}{C_S} = \frac{2 \cdot q_S}{C_S} \cdot \left(\frac{C_S}{C_S + C_{p1}} \right) \cdot \left(\frac{C_S}{C_S + C_{p3} + C_{po} + C_{ox2}} \right) \\ &= \frac{(v_{in}+ - v_{in}-) \cdot C_S}{C_S} \cdot \left(\frac{C_S}{C_S + C_{p1}} \right) \cdot \left(\frac{C_S}{C_S + C_{p3} + C_{po} + C_{ox2}} \right) \qquad (4 - 24) \\ &= v_{in} \cdot A_v \end{aligned}$$

where:

$$A_v = \left(\frac{C_S}{C_S + C_{p1}} \right) \cdot \left(\frac{C_S}{C_S + C_{p3} + C_{po} + C_{ox2}} \right) \qquad (4 - 25)$$

It is evident by (4 - 25) that the gain, A_v, is less than unity and parasitic dependent (as expected for a passive system). This implies that good layout matching (as discussed in Section 4.3-1) is needed not only for the main capacitors, C_S, but the parasitic capacitors as well, to minimize cell gain variations, i.e., correlator PN code variations. As mentioned previously (See “Isolation of the summing nodes” on page 119), it is important to shield the summing nodes (nodes connected to C_S) from extraneous charge coupling that can

1. The derivation of (4 - 25), and hence the validation of (4 - 23), can be found in Appendix B.

introduce errors that cannot be cancelled.

4.2-9 Bottom-plate passive integrator

The passive integrator can also be implemented using a differential *bottom-plate* architecture which is slightly less complex than the *top-plate* architecture described above. Fig. 4-14 is the corresponding diagram to Fig. 4-12 for a passive bottom-plate integrator

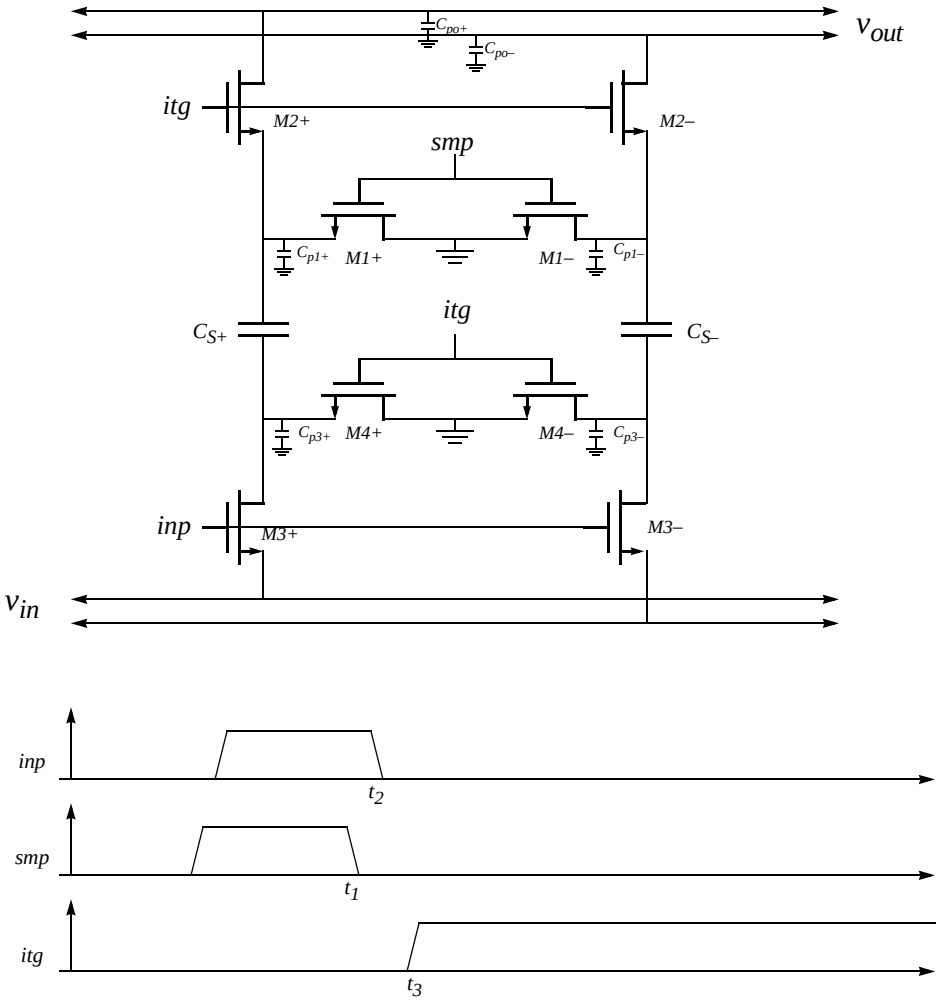


Figure 4-14 Passive differential bottom-plate integrator cell.

cell. Using this structure, a separate zeroing step is not needed since any error charge on the bottom-plate of C_S and top-plate of C_{p3} is shorted out during the integration step when *M4* turns on. Again the summing node (top-plate of C_S) charge is preserved once *M1* turns off when the signal sample is taken so the only corrupting charge will be as a result of *M1* turning off (Q_1) as before, which will be common-mode to the extent that the impedances of the positive and negative sides match and/or the fall time of *smp* is "fast". By avoiding the zeroing step, the signal loss due to charge-sharing with C_{p1} is eliminated so the voltage gain is:

$$A_v = \left(\frac{C_S}{C_S + C_{p3} + C_{po} + C_{ox2}} \right) \qquad (4 - 26)$$

4.3 Other Non-ideal Considerations in SC Designs

It is evident from (4 - 25) and (4 - 26) that layout and matching have a significant impact on performance of a passive integrator, in particular, and of SC circuits in general. This section will briefly cover those and other issues that one must address when working with these analog circuit blocks.

4.3-1 Matching

Like most analog circuits, good matching of device geometries plays an important role in circuit performance. While matching of main components (capacitors, resistors and transistors) is the primary objective, matching of circuit parasitics is usually of second-order concern. However, for the passive integrator, the gain is a direct function of the

parasitic capacitances as (4 - 25) shows, so their matching is just as important. Fortunately, good matching of main components relies on using the same cell for all layout instances so the parasitics tend to match as well as the main components except at the cell boundaries. Since boundary conditions influence the parasitics more than the main components, care needs to be taken to ensure that each layout cell instance "sees" the same neighboring structure as all other cell instances for each of its edges. This usually means using dummy cells at the ends of rows and columns of the layout.

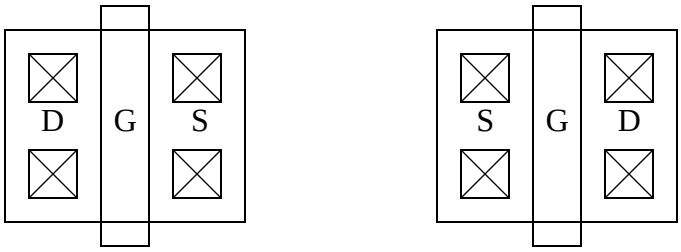
Another important matching dimension is the two sides, positive and negative, of a differential structure. As shown in the above derivation of error cancellation, good common-mode rejection relies on this positive to negative matching. It is important to avoid incorrect mirror symmetry which is a natural tendency when laying out differential structures. Fig. 4-15 illustrates incorrect mirror symmetry as well as two correct alternatives (mirrored and non-mirrored) for matching of transistors.

4.3-2 Summing node integrity

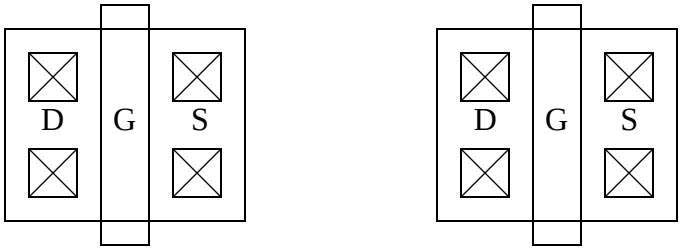
As mentioned previously (“Isolation of the summing nodes” on page 119), shielding with grounded traces should be used to preserve the charge integrity on the summing nodes of SC circuits.

4.3-3 Non-linearity of parasitic junction capacitance

The parasitic capacitances associated with a MOS transistor switch are the non-linear capacitances formed by the pn-diode junctions of the source and drain diffusions to the



(a) incorrect mirrored-symmetry matching



(b) correct non-mirrored matching

(c) correct mirrored symmetry matching

Figure 4-15 Mirror symmetry matching for differential structures.

bulk material. These capacitances are inversely proportional to the depletion width of the reverse-biased junction which in turn is dependent upon the reverse-biased[1] voltage, V_{sb}, giving the following non-linear voltage dependence:

$$C_j(V_{sb}) = \frac{C_{j0}}{\left(1 + \frac{V_{sb}}{\phi}\right)^m} \qquad (4 - 27)$$

1. For a NMOS transistor, the source(or drain)-to-bulk voltage would be the reversed-biased voltage.

where C_{j0} is the value when $V_{sb} = 0$, and ϕ (built-in potential) and m (grading factor) are process parameters. For the passive integrator, C_{p1} is predominantly the drain-bulk junction capacitance of switch *M1*. C_{p3} is made up of the junction capacitors from *M2*, *M3* and *M4*, plus the parasitic peripheral capacitance of C_S and the metal-interconnect capacitance of that node. Normally the parasitic peripheral capacitance of C_S will dominate C_{p3}, lowering the influence of any junction capacitance non-linearity.

The voltage dependence of the junction capacitors translates to a variation of the gain (see (4 - 25)) depending on the bias voltages of C_{p1} and C_{p3}. In the case of C_{p1}, it is the bias voltage during the "zeroing" phase that determines the value of C_{p1}, since this is the capacitance value that determines the amount of charge sharing with C_S as given by (4 - 22). As for C_{p3}, its effect on the gain is only dependent upon the final integrated output voltage value, i.e., the C_{p3} gain factor[1] is not dependent on the individual sampled values. In contrast, the C_{p1} gain factor (4 - 22) varies for each sampled value and thus diminishes the ability of the *top-plate* passive correlator to completely reject undesired orthogonal signals (users) of a CDMA waveform which decreases the SNR of the output signal. [Note that the gain of the *bottom-plate* passive integrator (page 127) is not a function of C_{p1} and thus does not have this problem.] Thus the effect of C_{p3}'s non-linearity is not a function of the individual sampled signals, is attenuated by the larger parasitic peripheral capacitance of C_S and can thus be neglected; whereas for C_{p1}, its non-linearity can decrease SNR for CDMA waveforms and needs to be addressed.

1. Second factor in (4 - 25).

Assuming that C_{p1} is only composed of *M1*'s junction capacitance, the differential charge error, E_{dq}, due to the non-linearity can be written as (see Appendix B):

$$E_{dq} = E_1 + E_2 + E_3 \tag{4 - 28}$$

where:

$$E_1 = \frac{1}{2} \cdot [E_a + E_b] - 1 \tag{4 - 29}$$

$$E_2 = -\frac{Q_1}{2 \cdot q_S} \cdot [E_a - E_b] \tag{4 - 30}$$

$$E_3 = \frac{V_z \cdot C_{p1o}}{2 \cdot q_S} \cdot \left[\frac{C_{p1}^{+}}{C_{p1o}} \cdot E_a - \frac{C_{p1}^{-}}{C_{p1o}} \cdot E_b \right] \tag{4 - 31}$$

$$E_a = \frac{\left(1 + \frac{C_{p1o}}{C_S}\right)}{\left(1 + \frac{C_{p1}^{+}}{C_S}\right)} \tag{4 - 32}$$

$$E_b = \frac{\left(1 + \frac{C_{p1o}}{C_S}\right)}{\left(1 + \frac{C_{p1}^{-}}{C_S}\right)} \tag{4 - 33}$$

$$\frac{C_{p1o}}{C_S} = \frac{C_{j0}}{C_S} \cdot \frac{1}{\left(1 + \frac{V_{cm}}{\phi}\right)^m}$$

$$\frac{C_{p1}^{+}}{C_S} = \frac{C_{j0}}{C_S} \cdot \frac{1}{\left(1 + \frac{V_{cm} + \Delta V}{\phi}\right)^m} \qquad (4 - 34)$$

$$\frac{C_{p1}^{-}}{C_S} = \frac{C_{j0}}{C_S} \cdot \frac{1}{\left(1 + \frac{V_{cm} - \Delta V}{\phi}\right)^m}$$

ΔV is the voltage difference, with respect to the common-mode voltage, V_{cm}, of the top-plates of C_{p1}^{+} and C_{p1}^{-}, respectively and V_z is the voltage of the top-plate of C_S when *M4* is closed (see Fig. 4-12 or Fig. 4-13). The first term (E_1) represents the direct gain error due to the variation of C_{p1}^{+} and C_{p1}^{-}; the other terms (E_2 and E_3) can be attributed to unequal Q_1 and V_X/V_Y components, respectively, during the zeroing operation (*M4* is closed).

From the above equations, the error terms are ultimately all scaled by the ratio of C_{j0} to C_S which can be expressed as:

$$\frac{C_{j0}}{C_S} = \left(\frac{W}{C_S}\right) \cdot L_{dfu} \cdot \overline{C_j} \qquad (4 - 35)$$

where W is the switch size (width), L_{dfu} is the length of the source/drain diffusion for a given technology, and $\overline{C_j}$ is junction capacitance area density for a given technology. (4 - 35) suggests that increasing C_S with respect to the sampling switch size can make the

error arbitrarily small. However, the processing speed of the passive SC integrator is directly related to the sampling-mode bandwidth, ω_S, given by:

$$\begin{aligned}\omega_S &= \frac{1}{R_{on} \cdot C_S} \\ &= \frac{\mu \cdot C_{ox} \cdot \frac{W}{L_{min}} \cdot \left(V_{gs} - V_t - \frac{1}{2} \cdot V_{ds}\right)}{C_S} \\ &= \left(\frac{W}{C_S}\right) \cdot \frac{\mu \cdot \varepsilon_{ox} \cdot (V_{dd} - V_t)}{t_{ox} \cdot L_{min}}\end{aligned} \qquad (4 - 36)$$

where R_{on} is the switch on-resistance for a transistor in the triode region ($V_{ds} \cong 0$V). For a given technology (oxide thickness (t_{ox}) and L_{min} fixed) and supply (V_{dd} fixed), sampling speed is proportional to $\frac{W}{C_S}$. Combining this fact with the above equations for E_{dq} creates a trade-off between processing speed and error, i.e., increasing the speed by decreasing C_S and/or increasing the sampling switch size increases E_{dq}, or conversely, decreasing E_{dq} decreases the speed of the passive integrator. Fortunately, (4 - 36) indicates that if technology (t_{ox} and L_{min}) scales downward faster than supply voltage (V_{dd}), the processing speed will increase for a given level of non-linear error due to C_{p1}.

4.3-4 Sampled thermal noise (*kT/C*)

When switched capacitor circuits sample a signal, the thermal noise of the switch is also captured on the sampling capacitor. The power (variance) of these noise samples, N_{Si}, is:

$$N_{Si} = \frac{k \cdot T}{C_S} \tag{4 - 37}$$

where k is Boltzmann's constant and T is absolute temperature. For a typical SC active integrator circuit, this kT/C noise can be comparable to the opamp noise and limits how small C_S can be. In a passive SC architecture with no opamp, this can be the dominant circuit noise source and can determine the minimum size of C_S. However, for the passive integrator, the noise samples are added together during the integration cycle which effectively reduces the total noise power with respect to the signal power. This can be viewed in two ways. First, from Chapter 3 using the concept of processing gain for a SS correlator, the power of any uncorrelated signal will be reduced by the processing gain, G_p, i.e.:

$$\begin{aligned} G_P &= N \\ &= 10 \cdot \log(N) \text{ (dB)} \end{aligned} \tag{4 - 38}$$

where N is the correlation length. Thus the output or post-integration noise power, N_{So}, is

$$N_{So} = \frac{k \cdot T}{N \cdot C_S} \tag{4 - 39}$$

A second way to view the processing of the kT/C noise samples is to realize that discrete-time correlation or integration averages the noise samples. Assuming each noise sample is independent and has the same variance (power), then their average will have a variance that is lower by a factor of N, i.e., the output noise power will conform to (4 - 39). An intuitive interpretation of this result is that the output noise is attributed to the entire sampling array rather than just a single sampling capacitor. Thus for large correlation lengths, small sampling capacitors can be used with reasonable noise power.

4.4 References

[1]. D. J. Allstot and W. C. Black, Jr., "Technological Design Considerations for Monolithic MOS Switched-Capacitor Filtering Systems," *Proc. IEEE*, vol. 71, no. 8, pp. 967-986, Aug. 1983.

[2]. B. E. Boser and B. A. Wooley, "The Design of Sigma-Delta Modulation Analog-to-Digital Converters," *IEEE J. Solid-State Circuits*, vol. 23, no. 6, pp. 1298-1308, Dec. 1988.

[3]. R. W. Brodersen, P. R. Gray and D. A. Hodges, "MOS Switched-Capacitor Filters," *Proc. IEEE*, vol. 67, no. 1, pp. 61-75, Jan. 1979.

[4]. C. S. G. Conroy, D. W. Cline and P. R. Gray, "An 8-b 85-MS/s Parallel Pipeline A/D Converter in 1-μm CMOS," *IEEE J. Solid-State Circuits*, vol. 28, no. 4, pp. 447-454, April. 1993.

[5]. J.-H. Shieh, M. Patel and B. J. Sheu, "Measurement and Analysis of Charge Injection in MOS Analog Switches," *IEEE J. Solid-State Circuits*, vol. SC-22, no. 2, pp. 277-281, Apr. 1987.

[6]. P. W. Li, M. Chin, P. R. Gray and R. Castello, "A Ratio-Independent Algorithmic Analog-to-Digital Conversion Technique," *IEEE J. Solid-State Circuits*, vol. SC-19, no. 6, pp. 828-836, Dec. 1984.

[7]. K.-L. Lee and R. G. Meyer, "Low-Distortion Switched-Capacitor Filter Design Techniques," *IEEE J. Solid-State Circuits*, vol. SC-20, no. 6, pp. 1103-1113, Dec. 1985.

[8]. Y.-M. Lin, B. Kim and P. R. Gray, "A 13-b 2.5-MHz Self-Calibrated Pipelined A/D Converter in 3-μm CMOS," *IEEE J. Solid-State Circuits*, vol. 26, no. 4, pp. 628-636, April 1991.

[9]. D. W. Cline and P. R. Gray, "A Power Optimized 13-b 5 Msamples/s Pipelined Analog-to-Digital Converter in 1.2 μm CMOS," *IEEE J. Solid-State Circuits*, vol. 31, no. 3, pp. 294-303, March 1996.

Chapter 5

DS-CDMA Baseband Receiver Prototype

To demonstrate the high-speed and low-power capabilities of the passive switched-capacitor integrator, six analog binary correlators were integrated as part of a DS-CDMA baseband receiver integrated circuit (IC) operating at 64*Mchip/s*. The prototype chip was fabricated in a 1.2μm double-poly double-metal CMOS process.

This chapter is organized into four sections. The first section gives a brief description of the InfoPad Physical Layer subsystem for which this prototype was defined, followed by an overview of the entire baseband receiver prototype in Section 5.2. Section 5.3 presents low-power digital design techniques that minimize the dynamic power dissipation which comprises the bulk of the chip's power due to the passive nature of the correlators. To conclude the chapter, Section 5.4 covers those blocks involved with the synchronization (acquisition and tracking) process.

5.1 Brief Overview of the InfoPad Physical Layer Subsystem

The prototype IC was defined as part of the InfoPad [1] project's physical layer sub-

system. The InfoPad project at UC Berkeley was research in low-power IC methodologies applied to the design of an indoor cellular-like base station and portable pen-based terminal system, with high-speed (2*Mb/s*) downlink data transfer, hardware hand-writing recognition and A/V compression capabilities. The downlink physical layer subsystem was a DS-CDMA system patterned after Qualcomm Inc's cellular phone system. Since the prototype was a baseband system, the RF modulation and transmission as well as the RF reception and demodulation details are omitted but the reader is directed to [1] or the InfoPad project website at UC Berkeley (http://infopad.eecs.berkeley.edu/) for more information.

The prototype system had a spread factor of 64, a PN-sequence length of 32,768 chips, a data rate of 2*Mb/s* with quadrature phase-shift keying (QPSK) modulation, a tracking correlation length of 896 and a multi-user modulation using a 64-code Walsh function. A standard delay locked-loop (DLL) employing *Early, Late* and *Data* correlators required sampling the input signals at twice the chip rate (128*MS/s*) to maintain chip-phase tracking.

5.2 General Description of the Baseband Receiver

The realization of the DS-CDMA receiver system was partitioned into three parts; the correlator and state-machine (COR) block, the analog-to-digital converter (ADC) and back-end digital processing (FPGA) block (see Fig. 5-1). The COR and ADC blocks were integrated as a single CMOS chip whereas the FPGA was an off-the-shelf field-program-

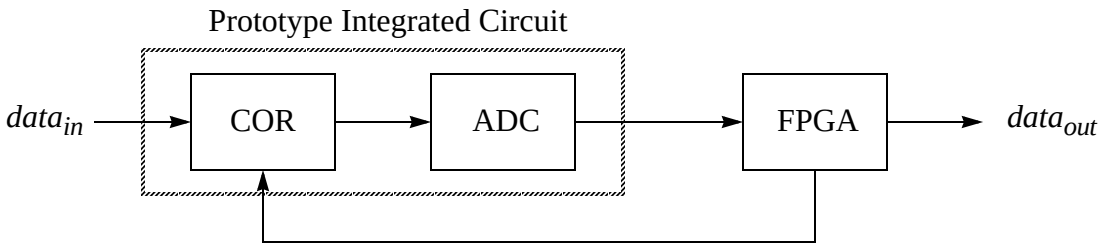


Figure 5-1 Experimental DS-CDMA baseband receiver (simplified block diagram - clock connections omitted).

mable gate array. The prototype chip developed in the research project by T. Cho [2] was used as the ADC block with only minimal modifications to its interfaces to combine it with the COR block. All timing for both the ADC and FPGA was generated in the COR block while the COR block derived all its timing from an external 256*MHz* frequency source.

The FPGA block performed the output-word alignment,[1] as well as the *digital correction*, needed for a pipelined A/D converter. It also demultiplexed the six correlator outputs, accumulated the partial-correlation sums, compared the *Early* to *Late* sums (tracking mode) and compared the *Early* and *Late* sums to the acquisition threshold (acquisition mode). Its outputs (*Shift*, *Up* and *delay2Tc*) were sent to the COR block to close the synchronization timing loop. Because it only needed to operate as fast as the ADC block (8*MHz*), and it could be reprogrammed for different acquisition and tracking algorithms during the testing phase, this block was implemented as an external FPGA rather than as

1. If not done internally, the outputs bits from each pipeline stage need to be *pipe-organ* buffered to align all the bits for each complete A/D conversion.

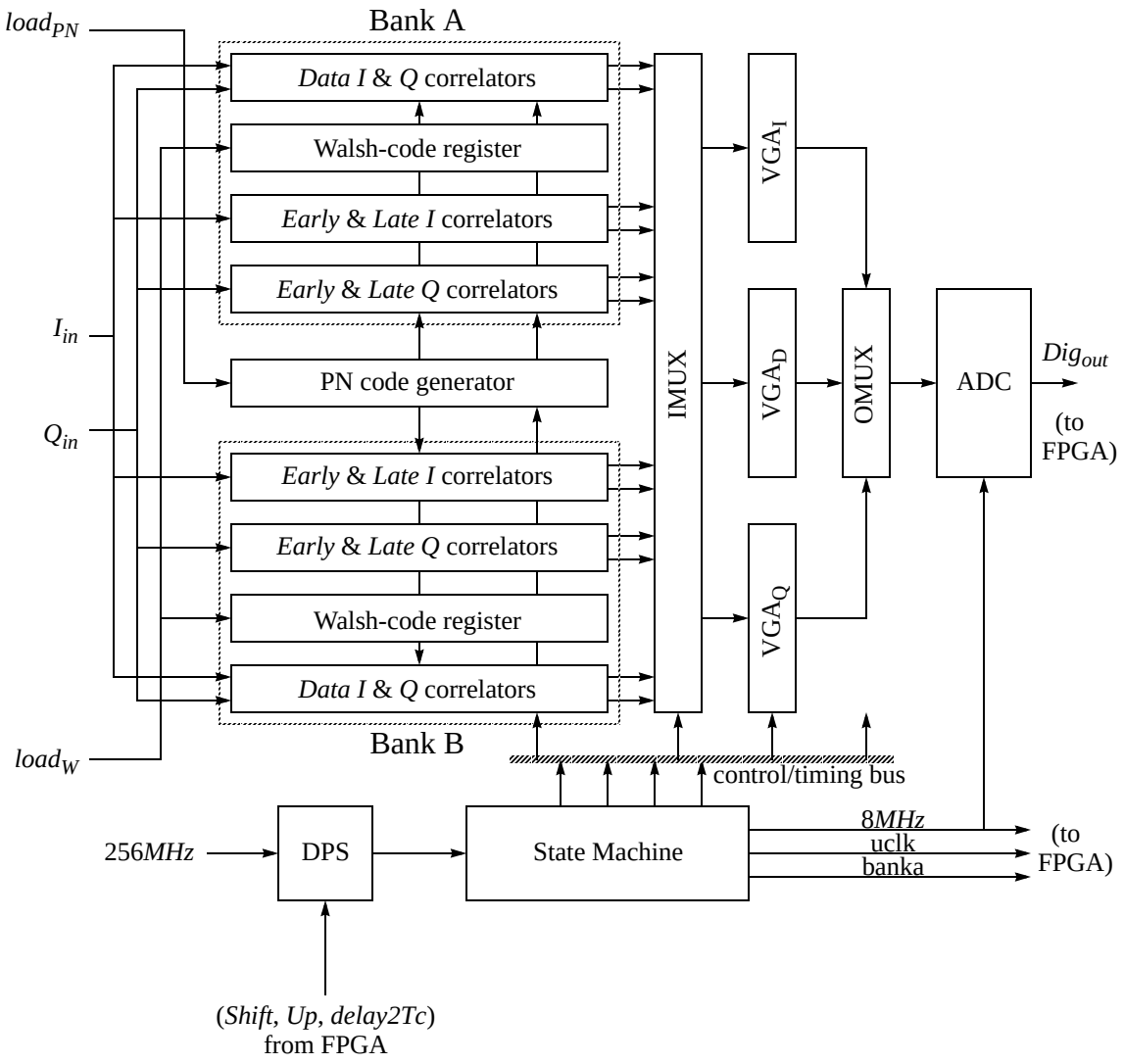


Figure 5-2 Simplified block diagram of prototype IC.

one of the blocks in the prototype chip.

Fig. 5-2 is an expanded view of the prototype IC chip. It incorporated six passive correlators, a 16-stage PN-code generator, two sets of Walsh-code registers, a 256*MHz* digital phase shifter (DPS), a 1024-state machine, three 4-to-1 multiplexers (IMUX), three variable gain amplifiers (VGA), a 3-to-1 multiplexer (OMUX) and a single 10-*bit* pipe-

lined ADC. The chip used a dual power supply; 5*V* for the COR block and 3.3*V* for the ADC. While the ADC incorporated voltage-boosted supplies for the switches it needed, allowing operation at 3.3V, the number of switches needed by the COR block was too great to justify voltage boosting, so a 5V supply was used. The chip was designed to process both in-phase (*I*) and quadrature (*Q*) signals at 1*Mb/s*, to accommodate QPSK type modulations. This yielded an aggregate data rate of 2*Mb/s*. In each signal path (*I* and *Q*), there were three correlators (*Early, Late* and *Data*), which were arrayed in a delay-locked loop (DLL) configuration for synchronization and demodulation [3]. With a spreading factor of 64, each passive correlator was composed of 64 differential sampling/integrating cells. Twelve physical correlators were arranged as two banks of six logical correlators to accomplish the time-interleaved sampling and integrating operations needed for a passive correlator (see "Passive data-rate correlator" in Chapter 4). Data for the on-chip shift registers were downloaded from a lab computer to alter the PN-sequence generator seed, the Walsh-code value and the VGA gain values.

5.3 Low Power Digital Design

The advantage of a low-power high-speed passive-analog correlation technique can be diminished if the power of the digital circuitry is excessive or extravagant, therefore some effort was devoted to developing low-power digital techniques. The basic approach was to address the power consumption at the circuit level by minimizing the capacitance (*C*) or frequency (f) of the digital blocks based on the dynamic power equation:

$$P = C \cdot V_{dd}^2 \cdot \mathrm{f} \qquad (5 - 1)$$

With a power goal in the range of tens of milliwatts, the strategy was to operate as much circuitry as possible at the lowest clock frequencies.

5.3-1 Timing and control shift-register chain

The timing and control signal generation for the twelve 64-cell correlator chains consumed the highest dynamic power, because of its size (70% of the COR block) and operational speed (64*MHz*), so minimizing its power was most important. A single correlator chain was composed of 64 differential sampling/integrating cells (see Fig. 5-3).[1] A differential input and differential output bus connected all 64 cells in common. Each cell required four dynamic timing signals (*input*, *sample*, *zero* and *integrate*) and one static control signal (PN-code bit). The successive-sampling scheme translated to staggering each cell's timing signals in time as shown in Fig. 5-4. How these signals are generated and distributed has a large impact on the power consumption, so how to accomplished this, in the most power efficient way, was the main focus.

Centrally generating all the control signals and routing to each cell of each correlator chain was not practical because of timing skew and layout logistic problems. The most straight-forward approach is to generate the signals at one end of a correlator chain and propagate them down using four shift-register chains (see Fig. 5-5a). The drawback with this approach is that each shift register is being clocked at 64*MHz*, which translated to

1. Two such chains (one per bank) are needed to implement a single correlator (see Fig. 4-3).

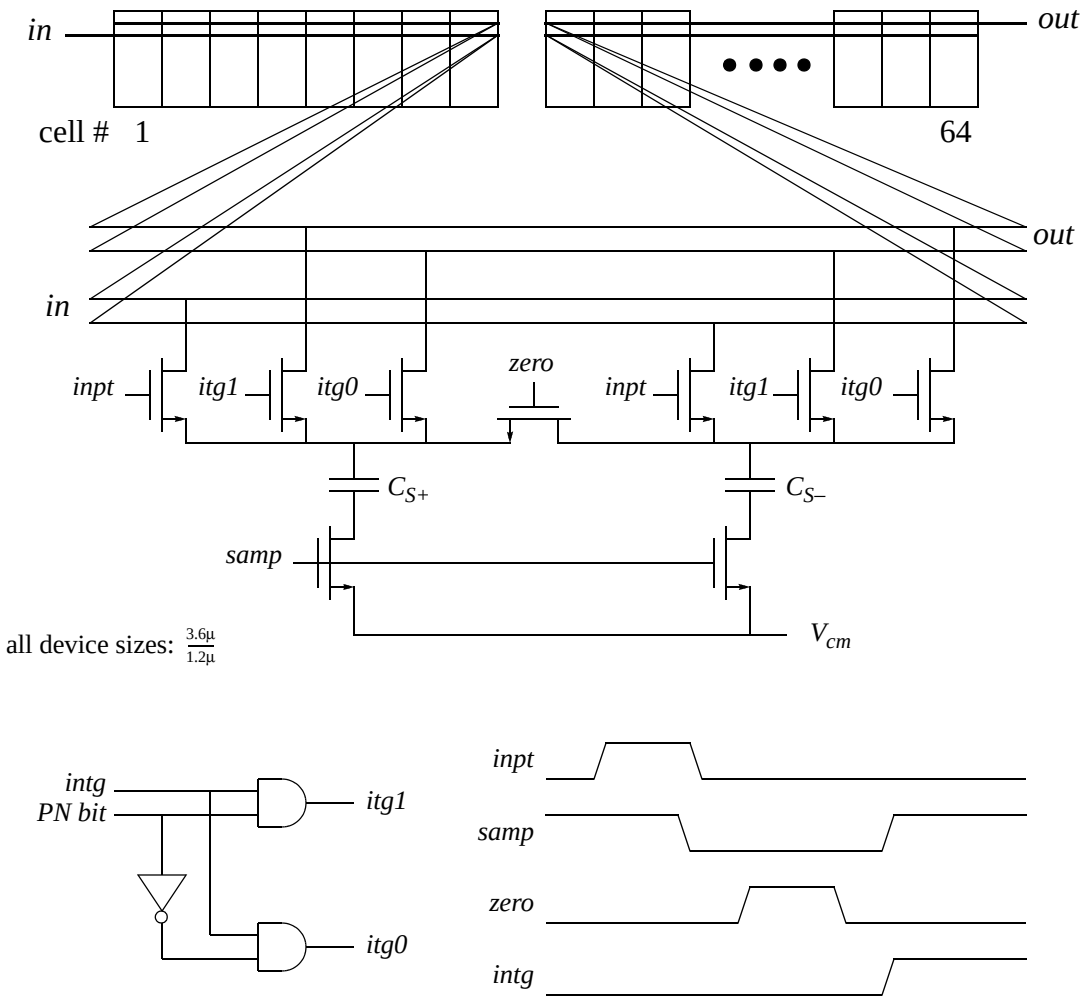


Figure 5-3 Detailed view of one differential sampling cell with timing diagram.

2.56*mW* per correlator chain in clock power alone, assuming a 25*fF* clock load per shift register. Another approach is to generate each set of signals locally[1] using the clock and a single timing (*start-pulse*) signal (see Fig. 5-5b). Local generation is well suited for this application because of the large amount of idle time (low duty cycle) for the timing signals. Local generation also reduces the clock load, and hence the clock power, at the

1. Local to a cell.

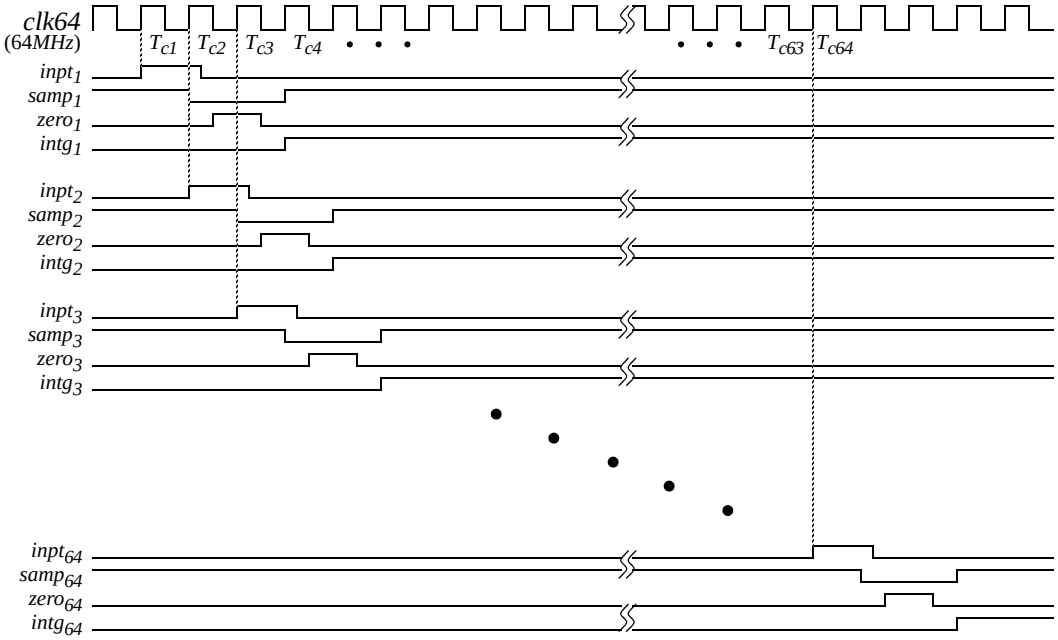


Figure 5-4 Staggered timing signal waveforms for successive sampling.

expense of some additional combinational-logic power. However, the clock load of the combinational logic will likely exceed that of a single shift register resulting in little power

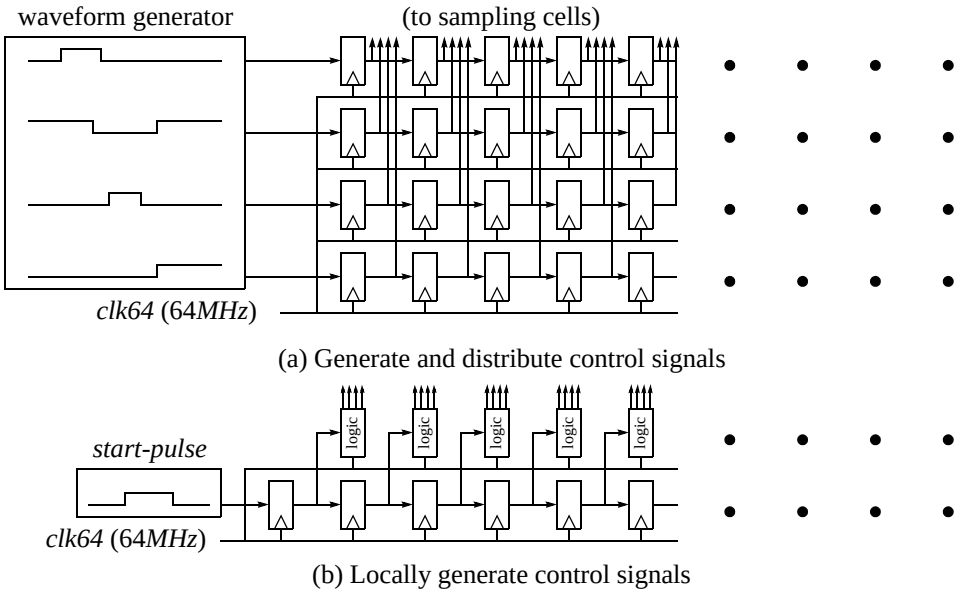


Figure 5-5 Two approaches to generating staggered timing signals.

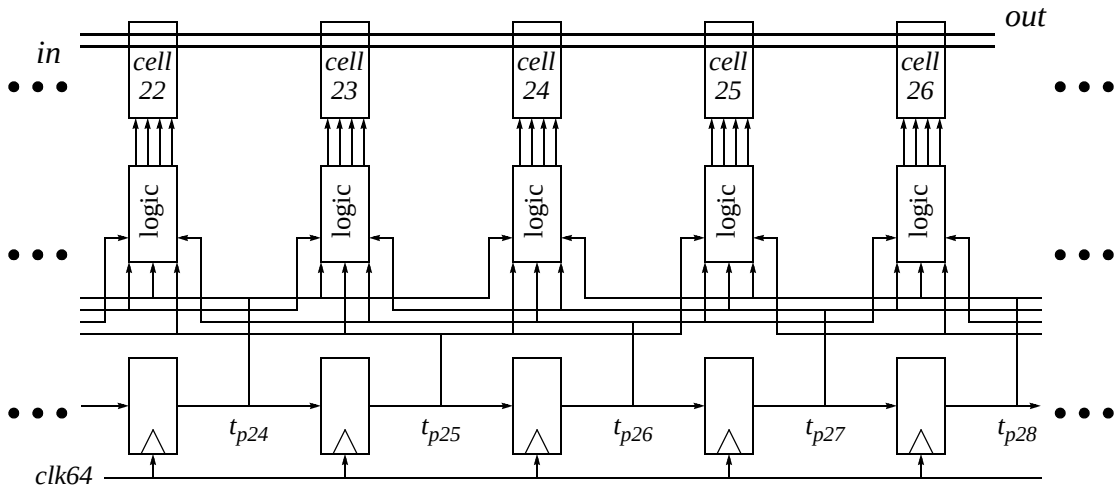


Figure 5-6 Minimize clock load by *multi-tapping* propagating *start-pulse* shift register chain (mid-chain example).

savings, if not increased power dissipation, over the generate-and-distribute approach. A modification to the local-generation approach is needed that will reduce the capacitance that is being clocked at 64*MHz*.

Two methods are possible to solve this clock-loading problem; both using the *start-pulse* signal that propagates down the correlator chain to start each cell's sampling sequence. The first method is to use self-timed circuits where the sequencing of the generation logic can rely on inherent gate-delays. These types of circuits have the potential problems of race conditions, metastability and increased transitions[1] from long timing-delay chains. The second method is to derive the necessary timing information by ***multi-tapping*** the *start-pulse* shift-register chain (see Fig. 5-6). This effectively creates a finite-length clock signal to run each local signal generator with the following two bene-

1. Increasing the number of transitions increases the effective frequency and hence the power.

fits:

1. The high-speed clock (64*MHz*) is loaded by a single shift-register chain, i.e., it does not have to drive the logic.

2. **Automatic state decoding**: the occurrence of each *tap* pulse automatically identifies a specific chip time. This significantly reduces the complexity of the generator logic and the associated switching power.

With its better power performance, the latter method was adopted in the prototype IC. In the actual implementation, timing signal edges were aligned to the clock and a negative *start-pulse* was used to ease the design by favoring NAND logic (see Fig. 5-7). The

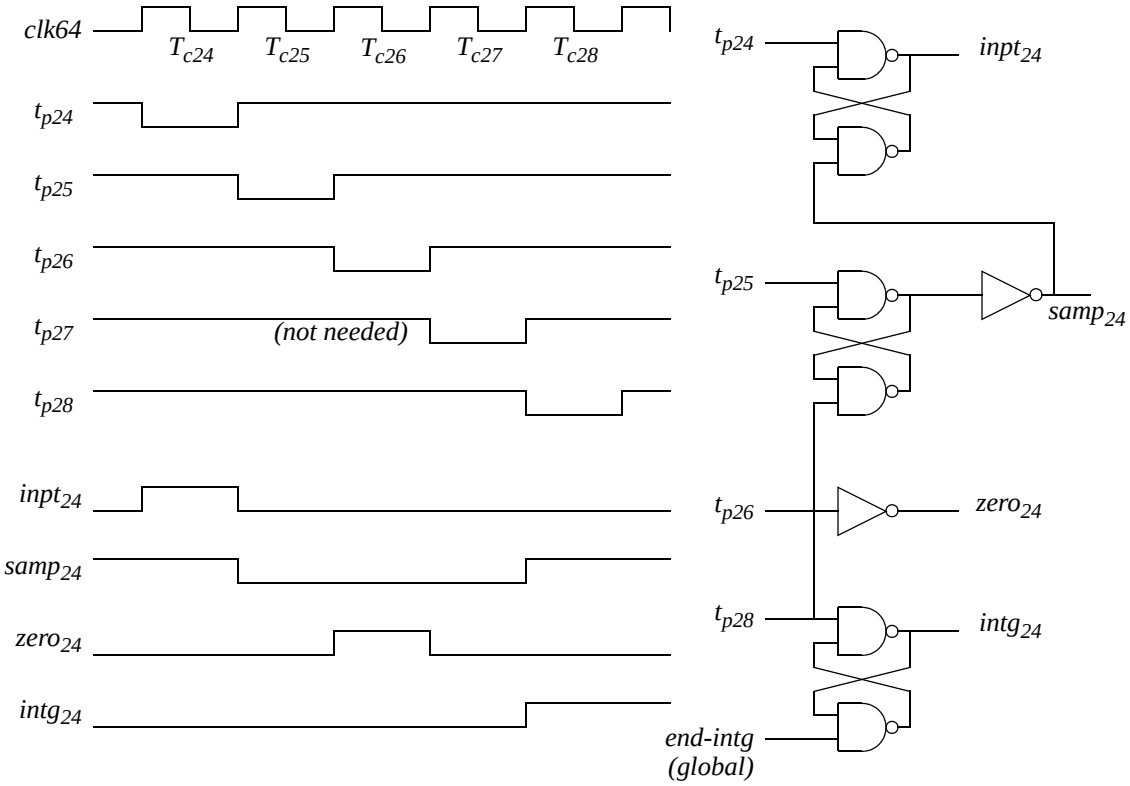


Figure 5-7 Local-generation logic and timing diagrams for one cell (*cell 24* example).

simplicity of the logic benefited greatly from not having to embed a state machine in each logic block.[1]

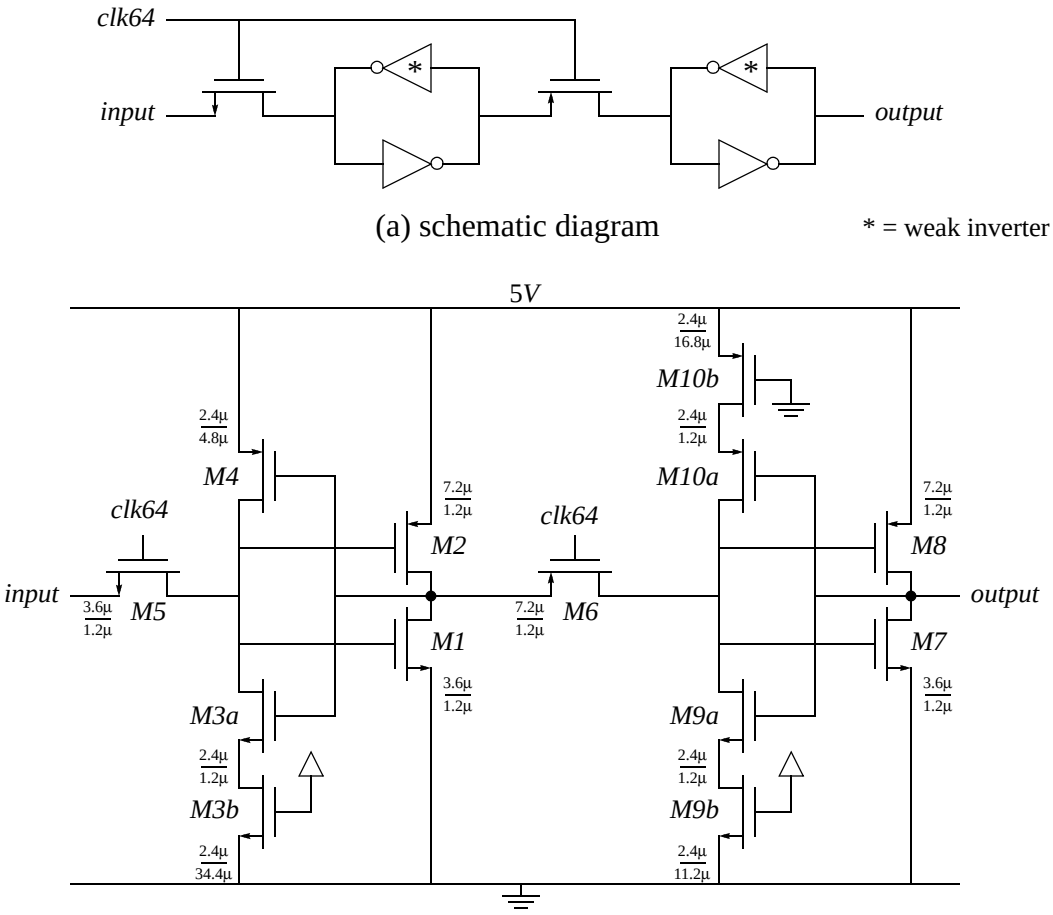
The next decision is the type of shift-register element to use that dissipates minimal power. Any D-type or master-slave flip-flop circuit can be used but exploiting the characteristics of the *start-pulse* SR chain will permit additional power savings. The SR chain has the following characteristics (+) and requirements (–):

1. (+) Low duty cycle. A given register is idle most of the time.
2. (+) Edge-triggering is not crucial. Input signal is stable for a long time.
3. (–) Low clock input capacitance. This directly impacts the dynamic power.
4. (–) No internal node transitions while flip-flop is being clocked and input is not changing (IDLE). Extraneous internal transitions increase the dynamic power.
5. (–) Static design. This will permit stopping the clock for a correlator bank during the integration phase (described later in this section).

The simple master-slave flip-flop (MSFF) of Fig. 5-8 met the above requirements and took advantage of the relaxed characteristics (1 and 2 above). The key features of this structure are:

1. Simplicity and static design. Minimizing the number of elements (transistors) and their sizes, in general, reduces the parasitic capacitance thus increasing the speed

1. Note that only four taps are needed (t_{p27} is not used in the example of Fig. 5-7). "*end-intg*" is a global signal for each correlator signaling the end of the integration period.



(a) schematic diagram

(b) circuit diagram

Figure 5-8 Shift register chain flip-flop (MSFF).

while reducing the power. The static design eliminates any glitching during IDLE.

2. A clock input load limited to 2 transistors (about 20*fF*). Together with another 10*fF* (per cell) of metal clock-line capacitance results in a clock power of 3.1*mW* per correlator or 16*mW*[1] for all correlators. Most other flip-flop implementations have 4 or more transistors loading the clock inputs and thus contribute to higher clock power.

1. There are 8 timing SR chains for the 12 correlators but they run at a 62.5% duty cycle due to another power saving technique that is described later (see “Bank-selected correlator clocks” on page 153).

3. Feedback inverters can be slow and weak, thus dissipating less short-circuit switching power. This is a direct consequence of having a stable input signal during the master or slave latching cycle. The feedback inverters re-establish full-swing outputs allowing the use of *single polarity* switches (M5 and M6) which would otherwise have to be transmission gates (CMOS switches) that would double the clock power.

4. *Split-gate* transistors. Weak feedback inverters imply minimum-width long-channel devices which present a high gate capacitance to the forward inverters, decreasing speed and increasing power. To improve the performance in speed and power, a ***split-gate*** technique was used for *M3*, *M9* and *M10*, where a minimum-length transistor segment was connected to an "always on" larger transistor segment reducing the switching load for the forward inverters.[1]

To quantify the results, Table 5-1 presents the power estimates[2] for the generate-and-distribute and modified local-generation approaches. These results show that the local-generate approach can reduce the power by a factor of three over the straight-forward generate-and-distribute approach due primarily to reducing the 64*MHz* clock capacitance.

TABLE 5-1 Timing-Signal Generation Power

Approach	Power
Central generation and distribution	81.89 *mW*
Local generation with multi-tapped clock	27.54 *mW*

1. From simulations, this gave the latch a 3X improvement in power-delay product. The gate of M4 was not long enough to benefit from splitting. Minimum gate width was 2.4μ.
2. These estimates are the power for all correlator timing chains and local generating logic. Their derivation can be found in Appendix C.

5.3-2 Double-latched PN generator outputs

Another circuit that operated at the highest clock rate, and therefore could benefit from *power-conscious* design, was the PN generator block. Though the absolute power savings using this *double-latched* approach turns out to be modest, the relative or percentage savings is significant (see Table 5-2 at end of section).

The PN-sequence length of 32,768 chips, was chosen to be a multiple of the spreading factor (64 chips) to ease implementation and acquisition time. This sequence length meant using a 16-stage, rather than 15-stage, PN generator, and selecting a 32,768 sub-sequence from the 65,535 chip-sequence total (see Chapter 3). To ease the end-of-sequence detection, the PN generator is loaded with a 16-chip *seed* word such that the sequence ends when all 16 registers contain a "1". Optionally, this *seed* word can be loaded externally to alter the generated sequence length for testing purposes. Fig. 5-9 is a

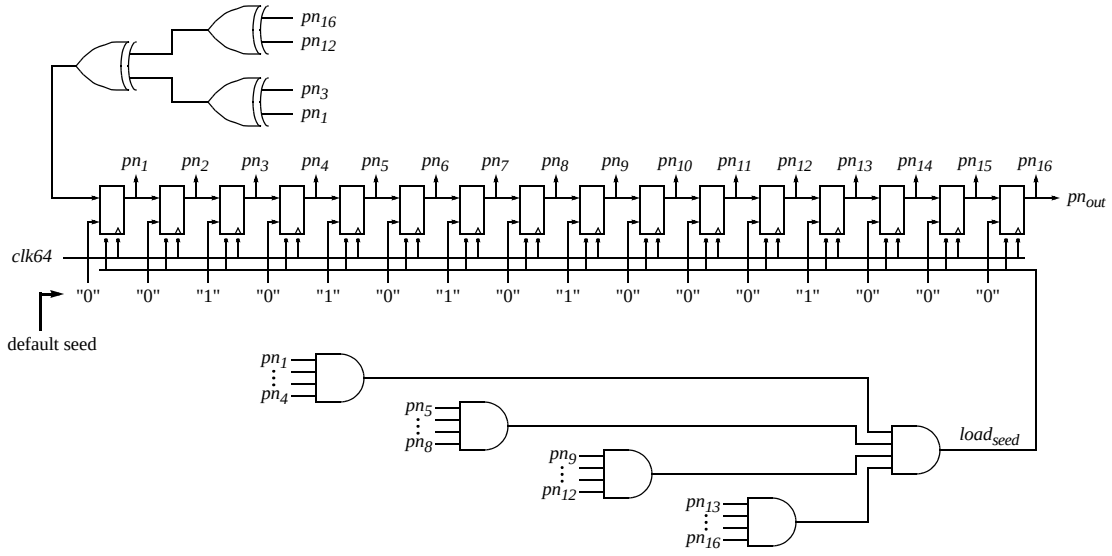


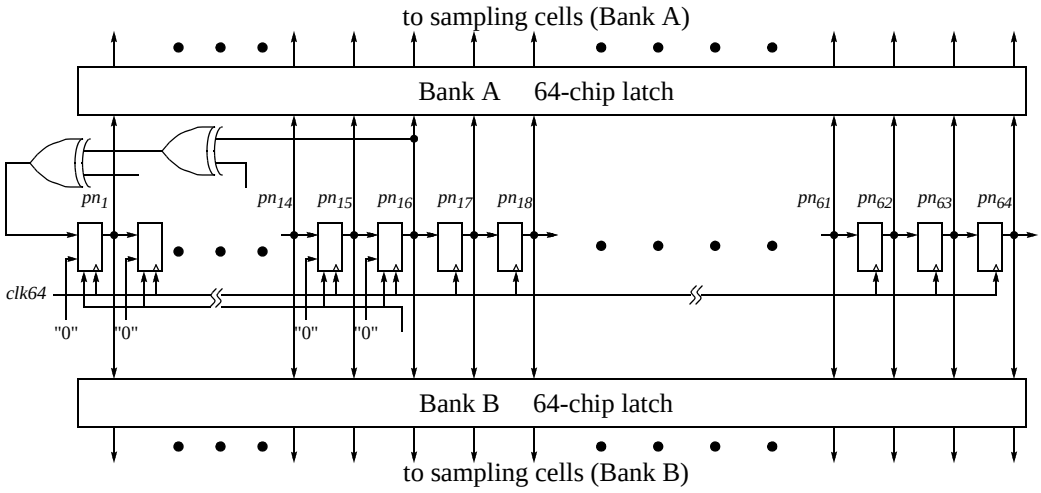
Figure 5-9 16-stage PN generator schematic with parallel load of the "*seed*" word at end-of-sequence (all ones) detection.

simplified schematic of the 16-stage PN generator.

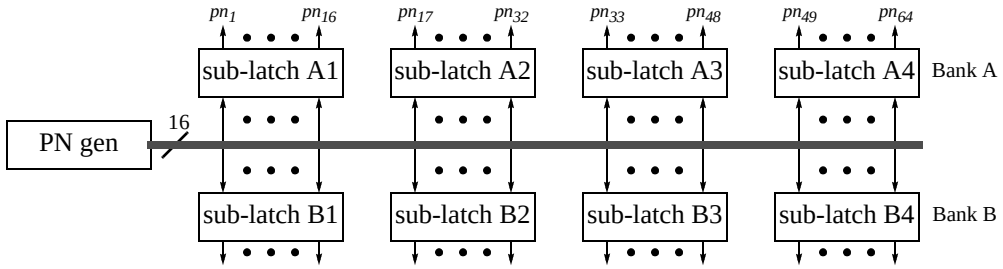
When a 64-chip sub-sequence is generated, it is latched and held for the duration of the integration period. The alternating sample-and-integrate dual-banking scheme required two 64-chip latches, one for each bank. The most straight-forward way of loading the latches is to extend the PN generator by adding an additional 48 shift-registers and performing a 64-chip *parallel* load into the latches (see Fig. 5-10(a)). This has the disadvantage of having to clock the additional 48 shift-registers at $64MHz$ (an estimated $3.4mW$ in clock power). A lower power approach may be to distribute the PN generator outputs to eight 16-chip sub-latches (see Fig. 5-10(b)) where four sub-latches comprise one 64-chip bank latch. Now the $64MHz$ clock load is limited to the 16 PN generator shift-registers which lowers the clock power by a factor of 4. However this result ignores the power dissipated by the outputs of the PN generator registers that have to drive the latches. As shown in Appendix C, when this power is included in the analysis, this second approach dissipates an equivalent amount of power due to the capacitance of the 16 bus lines.[1] A characteristic of the PN sequence is that the generator's outputs have an average frequency of 1/4 of the clock frequency,[2] i.e., $16MHz$ in the case of the IC prototype. It is apparent that this frequency is still high enough to warrant some method of reducing the load capacitance in order to reduce the power.

By adding another 16-chip (*master*) latch between the PN generator and the 8

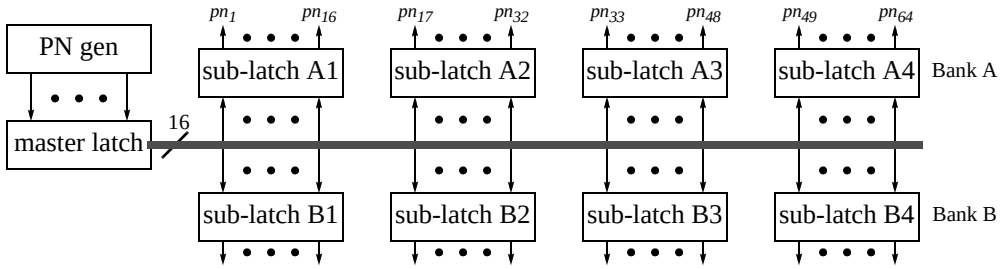
1. All inferences to a process assume a $1.2\mu m$ CMOS double-poly double-metal technology unless specified otherwise.
2. Based on code distributions for maximal length pseudo-random noise sequences (see Appendix C).



(a) Extend PN generator to 64 registers



(b) Single latch using 16-chip sub-latches



(c) Double latch using 16-chip sub-latches

Figure 5-10 Latching a 64-chip output from the PN generator.

sub-latches, the bus line and sub-latch capacitance can be isolated from the 16*MHz* outputs thus reducing the power. Now the generator outputs see only a single register load and negligible metal-line capacitance (assuming that the master latch is located next to the PN generator). Further, the clocking frequency of the master latch is 1/16th of the PN

TABLE 5-2 Latching Scheme Power Comparison

Latching Scheme	Power
PN Generator with 48 SR extension	6.25 mW
Single-latch configuration	7.04 mW
Double-latch configuration	1.71 mW

generator (4MHz), so the average rate of its outputs is 1MHz *(*1/4 the clocking frequency) which reduces the power to drive the sub-latch bus by a factor of 16. Table 5-2 presents the relative power dissipation of the three PN generator and latch subsystems of Fig. 5-10 based on calculations from Appendix C. Thus, as stated previously, the absolute power savings of the *double-latched*[1] approach over the two other approaches is modest, but the relative power savings is large (factor of 3.5 to 4).

5.3-3 Bank-selected correlator clocks

The next technique saved more absolute power than the previous scheme, but it is a fairly common technique and less significant in terms of percentage power savings, so it is presented last.

In the dual-banking scheme for a given correlator, one bank samples while the other integrates the previous 64 samples. Since the integrating bank cells are idle during the integration cycle, the 64MHz clock can be stopped to save power. However a power reduction of 37.5% instead of 50% was realized because the multi-tapping clock configuration of the local signal generators (see Fig. 5-6) required some overlap of the clock sig-

1. In the prototype IC, the latency caused by the double-latch configuration was compensated by shifting the PN generator ahead in time with respect to the main-state machine.

TABLE 5-3 Timing-Signal Power & Bank Select

Approach	Power
Central generation and distribution	81.89 *mW*
(with bank select)	51.18 *mW*
Local generation with multi-tapped clock	27.54 *mW*
(with bank select)	17.22 *mW*

nals to the banks.[1] The bank select power savings is applied to the results of Table 5-1 and presented in Table 5-3. In this instance, the percentage power savings is modest but the absolute savings is significant.

5.3-4 True Single Phase Clocked (TSPC) D-type flip-flop (DFF)

The DFF structure (Fig. 5-11) used throughout the prototype IC (other than the static shift-register used in the correlator timing chain - see “Timing and control shift-register chain” on page 142), was the one based on the TSPC-2 latch described in [4]. Although not a low-power design technique developed in this work, its efficient design and good performance merits mention. After simulating a number of candidates, it was found to have the lowest switching power and clock input capacitance while being one of the fastest implementations.

5.3-5 Summary

Significant power reduction in the digital circuitry can be achieved by using design techniques that shift capacitive loads (*load-shifting*) to lower operating frequencies (*multi-tapped* timing chain with local signal generation and double-latched outputs) or

1. The power reduction could have been increased a few more percent but the added complexity to the clocks were not justified.

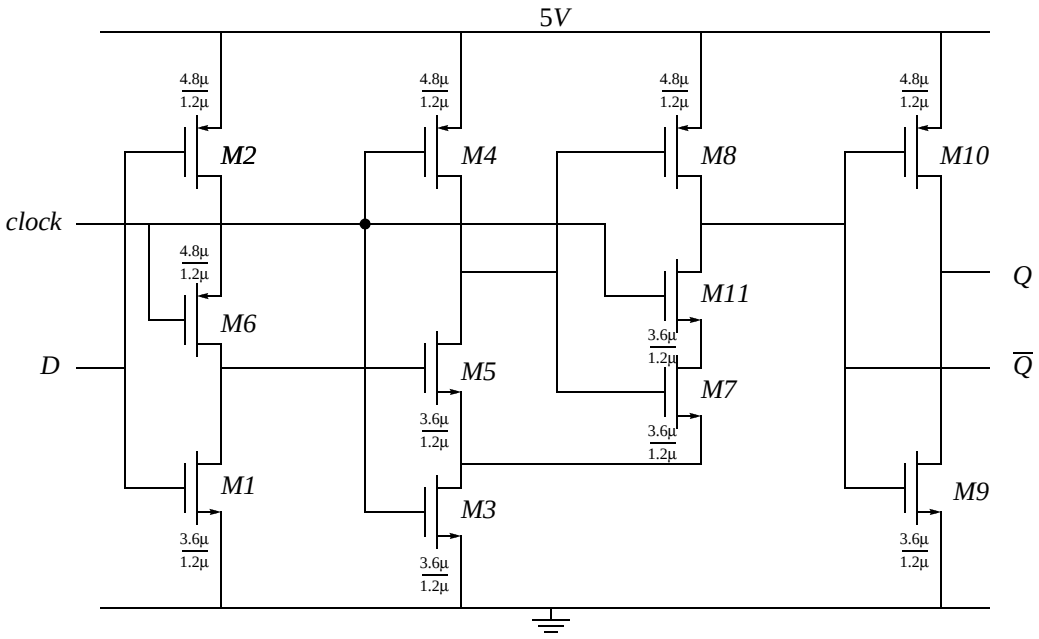


Figure 5-11 Dynamic D-type flip-flop based on the TSPC-2 latch.

reducing the effective frequency (bank-selected clocks). The key technique of *multi-tapping* the timing chain provided the generation logic with *automatic-state-decoding* and clocking information, thereby helping to minimize its gate-count. This coupled with a *split-gate* low-input-capacitance MS flip-flop (MSFF) resulted in a three-fold reduction in the power for generating and routing the clock signals (Table 5-1). Double-latching the PN generator outputs effectively accomplished *load-shifting* to achieve about a four-fold power savings for the PN generator block (Table 5-2). And stopping the clocks to the correlator bank in the integration phase helped to save 37.5% of the correlator power. The combined impact of all these techniques was a 79% reduction in power for these circuits or almost 50% (69*mW*) reduction in power for the whole prototype IC.

5.4 Synchronization and Demodulation Circuits

Before demodulation can proceed, the internal PN sequence must be aligned, or synchronized, to the same sequence embedded in the incoming received signal. This is accomplished by first acquiring a coarse open-loop alignment then closing a feedback loop to track and maintain the alignment. The correlation lengths and modulating signals used for the synchronization and demodulation differed due to the dual modulation scheme of the DS-CDMA system.

5.4-1 PN-sequence and Walsh-function modulations

To accommodate up to 50 users in an indoor cellular area, the system was designed to use a spreading factor of 64 which results in two 64*Mchips/s* CDMA data streams for a 2*Mbit/s* QPSK data rate. Using Walsh functions to modulate the users' data not only yielded 64 distinct codes but also made the transmitted signals mutually orthogonal due to the *synchronously-orthogonal* nature of Walsh functions.[1] However, the "whitening" and near-ideal autocorrelation property of the PN code was still necessary for successful phase acquisition and tracking, consequently, a second modulation by a long PN sequence was employed. The PN code was 32,768 chips which is longer than the 64-chip data bit so the PN code recycled every 512 data bits. This longer PN code permitted processing-gain (G_P) improvement of the acquisition and tracking correlation, of upwards to 45 dB, by increasing the correlation length if more robust synchronization is desired. Of course, longer tracking correlation lengths translate to infrequent timing updates which can be

1. See Section 3.4 for more information.

problematic due to the mismatch between the transmit and receive timing-reference signals.[1] A correlation length of 1024 chips for tracking and acquisition was chosen as a good compromise between timing drift and adjacent-cell interference problems based on simulations in [5]. A 1024-chip correlation is accomplished by summing sixteen 64-chip partial correlation results. However, in the actual implementation only 14 partial-correlation results (896 chips) were summed, allowing time ($2\mu s$ = 128 chips) for the comparison and update-signal generation within the FPGA block. Thus the correlation length for tracking and acquisition was 896 chips ($G_P = 30dB$) and for data demodulation was 64 chips ($G_P = 18dB$).[2]

5.4-2 Acquisition

A correlation length (896 chips) larger than a data bit (64 chips) would normally compromise the acquisition process through random data modulation of the partial correlation values (see Section 3.6). However, modulation by the Walsh functions avoids this difficulty by reserving one orthogonal code for a tracking and acquisition signal (*pilot-tone*) that is free of any data modulation. To ease the receiver design, the first Walsh code (all ones) was used for this signal, yielding a *pilot-tone* composed solely of the PN

1. The InfoPad downlink system was designed as an open-loop timing system requiring accurate external transmit and receive reference (chip) clocks. In contrast, the synchronization DLL was a closed-loop system used to obtain symbol (PN-sequence) lock. Implementing a chip clock phase-locked loop would have required a complex dual-loop system that was beyond the scope of this research.
2. The benefits of the 30dB processing gain for the tracking and acquisition correlations were not completely realized because the full correlation length was not used, i.e., the ideal autocorrelation characteristic is only achieved when the correlation length is equal to the maximal PN-sequence length (2^M-1). The autocorrelation of a PN subsequence will introduce side-spurs to the autocorrelation characteristic (much like those shown in Fig. 3-17) reducing the acquisition margin. In light of this fact, a PN-sequence length of 32,768 chips was not necessary nor beneficial, but only implemented to comply with the definition of the downlink system.

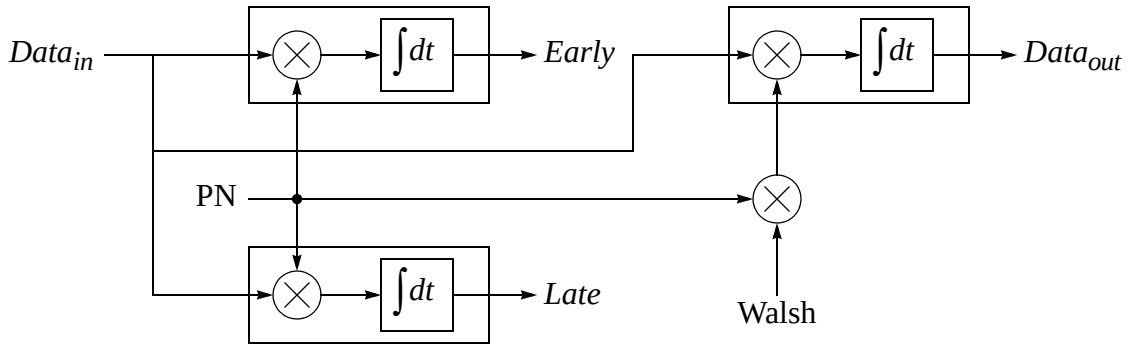


Figure 5-12 By using a *pilot-tone,* only *Data* correlation has to be Walsh-code demodulated.

sequence.[1] As Fig. 5-12 illustrates, pilot-tone (*Early* and *Late*) correlations need no Walsh codes in contrast to *Data* demodulation.

Acquisition is performed as a straightforward sequential search through all possible chip phases. Although there were six correlators ($Early_I$, $Late_I$, $Data_I$, $Early_Q$, $Late_Q$ and $Data_Q$), at most only three chip phases could be checked in one pass because vector rotation of the *I* and *Q* signals is possible, requiring complex correlation. However using all three (*Early, Late* and *Data)* correlator complex-pairs would have required changing their timing separation from half a chip-time ($T_c/2$) to one chip-time (T_c) and disabling the Walsh demodulation for the *Data* correlators. Therefore, only the *Early* and *Late* correlator complex-pairs were used in the acquisition search. By checking 2 chip-phases per pass, the maximum acquisition time to check all 32,768 chip-phases was 262*ms* with an average of 131*ms*. The two correlation results were compared to a programmable acquisi-

1. Modulation by all ones is equivalent to no modulation, hence the pilot-tone had no Walsh-function modulation yet was still orthogonal to all other Walsh-modulated signals. See Fig. 3-12.

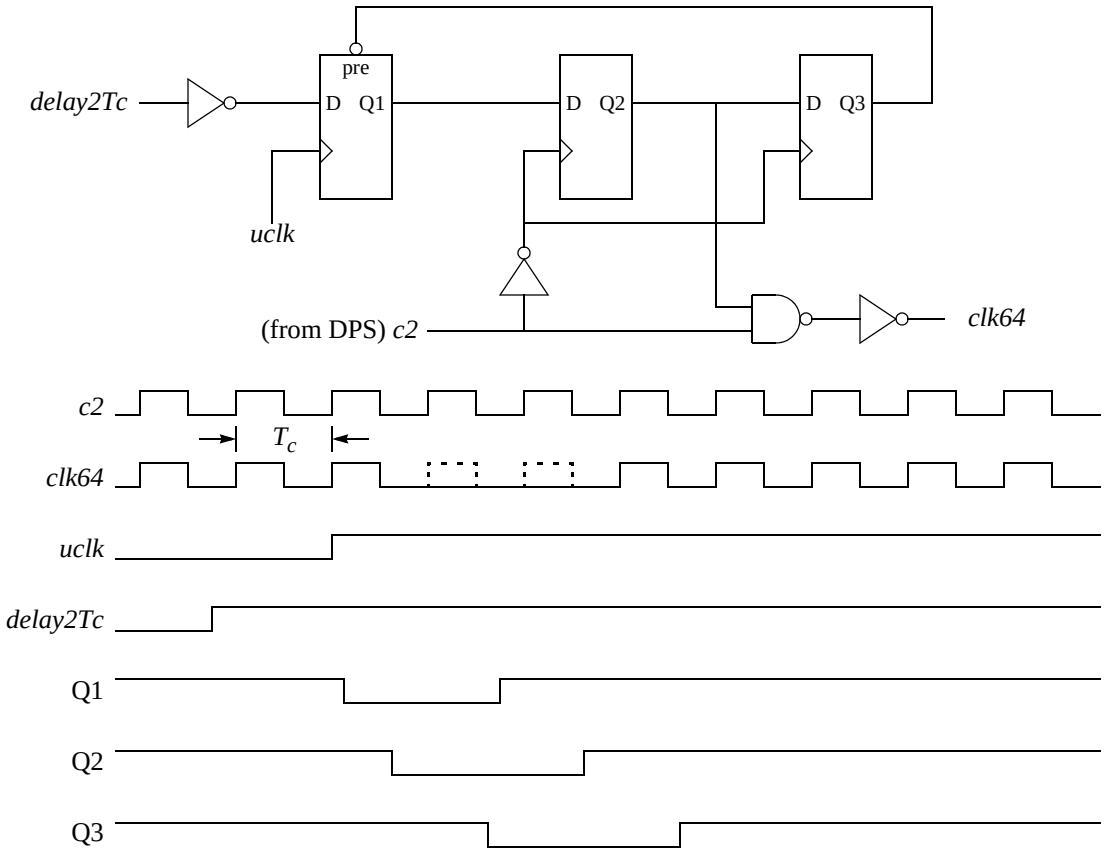


Figure 5-13 Simplified schematic of "slip"-2·T_c logic (acquisition mode).

tion threshold in the FPGA which fed back a signal (*delay2Tc* signal) to the COR block in response to the comparison result. When a correlation peak from either of the two correlator pairs was detected the clock phase was locked (*delay2Tc* set "low") and the tracking process was started. The circuit in the COR block that "slipped" 2 chip-times when the *delay2Tc* input signal was sampled "high" by *uclk* (update clock) is shown in Fig. 5-13. The *uclk* signal was generated by the main state machine to signal an internal chip-phase adjustment every 1024 T_c (16μs).

5.4-3 Tracking

In the tracking DLL architecture, the *Early* correlator sampled the input one half T_c ahead of the *Data* correlator and the *Late* correlator sampled one half T_c behind it yielding an aggregate 128*MS/s* sampling rate. The feedback of the DLL maintained alignment by keeping the magnitudes of the *Early* and *Late* correlations equal thus ensuring that the *Data* correlation was being sampled at the correlation peak for the highest SNR (see Fig. 3-20). The outputs of the *Early* and *Late* correlators were compared against each other by the FPGA which fed back an *up* signal value based on the comparison outcome. The COR block advanced or retarded the chip phase in response to the *up* signal value that was sampled by *uclk*.

A 256*MHz* external clock was required to offset the sampling of the I_{in} and Q_{in} signals by $\frac{T_c}{4}$ to accommodate the sub-sampling RF demodulator of the InfoPad system that was to precede the prototype chip. This complemented the need to adjust tracking by $\pm\frac{T_c}{4}$ which was at the processing-speed limit for a 1.2μm CMOS process. The circuit for the digital phase shifter (DPS) logic had to be custom designed to "evaluate" in the less-than-4ns clocking period. Critical timing also made it necessary to merge the advance $\left(-\frac{T_c}{4}\right)$ and retard $\left(+\frac{T_c}{4}\right)$ logic with the divide-by-4 needed to generate the 64*MHz* clock. The restrictive timing window precluded the use of a 2-stage ripple counter for the divide-by-4 function, requiring instead, a synchronous counter design. Rather than use a standard synchronous counter configuration, a SR architecture (see Fig. 5-14) was developed to allow more time for removing or adding a $\frac{T_c}{4}$ edge. The key feature of the

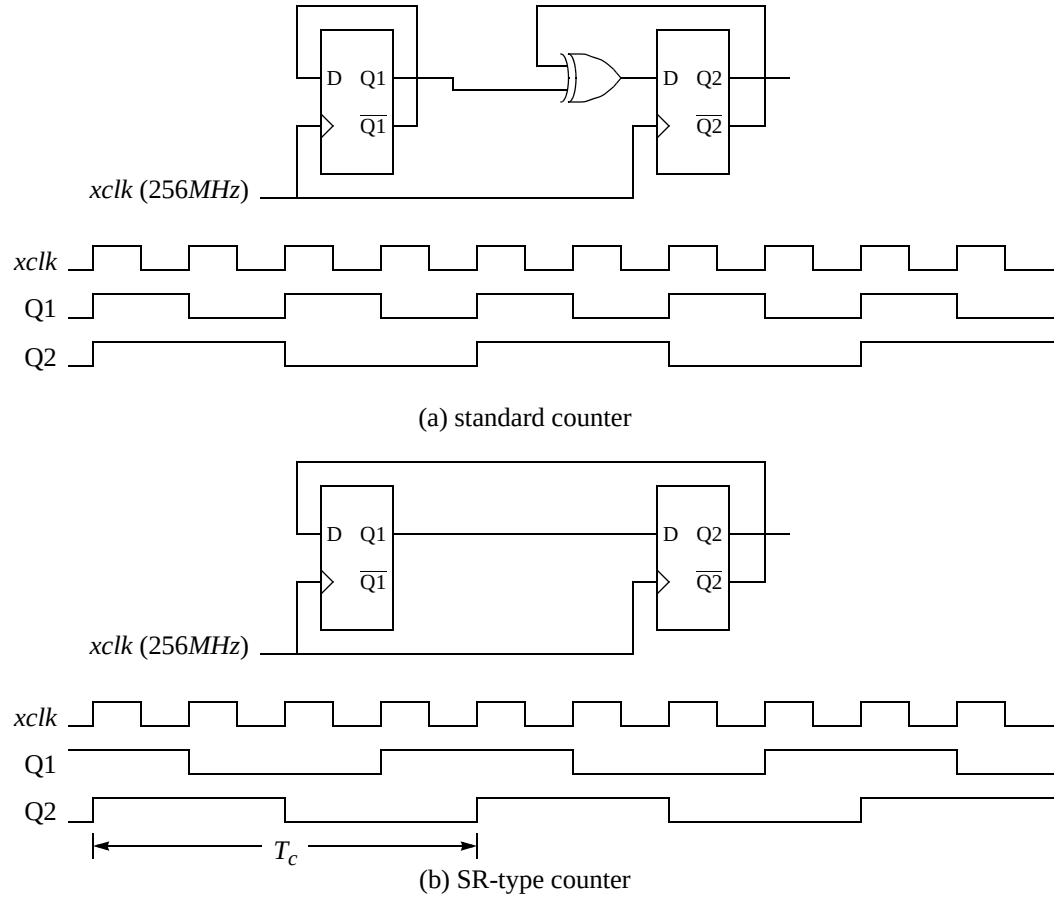


Figure 5-14 Divide-by-4 synchronous counter implementations.

SR-type counter is the fewer (half as many) transitions of the Q1 output providing a more stable signal with which to generate a **glitch-free** Q2 output. A simplified schematic of the DPS logic and its timing diagram is shown in Fig. 5-15. The DPS shifted the phase of the 64*MHz* clock (*c2*), by either advancing or retarding it, dependent upon the value of the *up* input signal, as long as the *shift* input remained high.

5.4-4 Demodulation

The 64-chip Data correlation involved a dual demodulation with both the PN

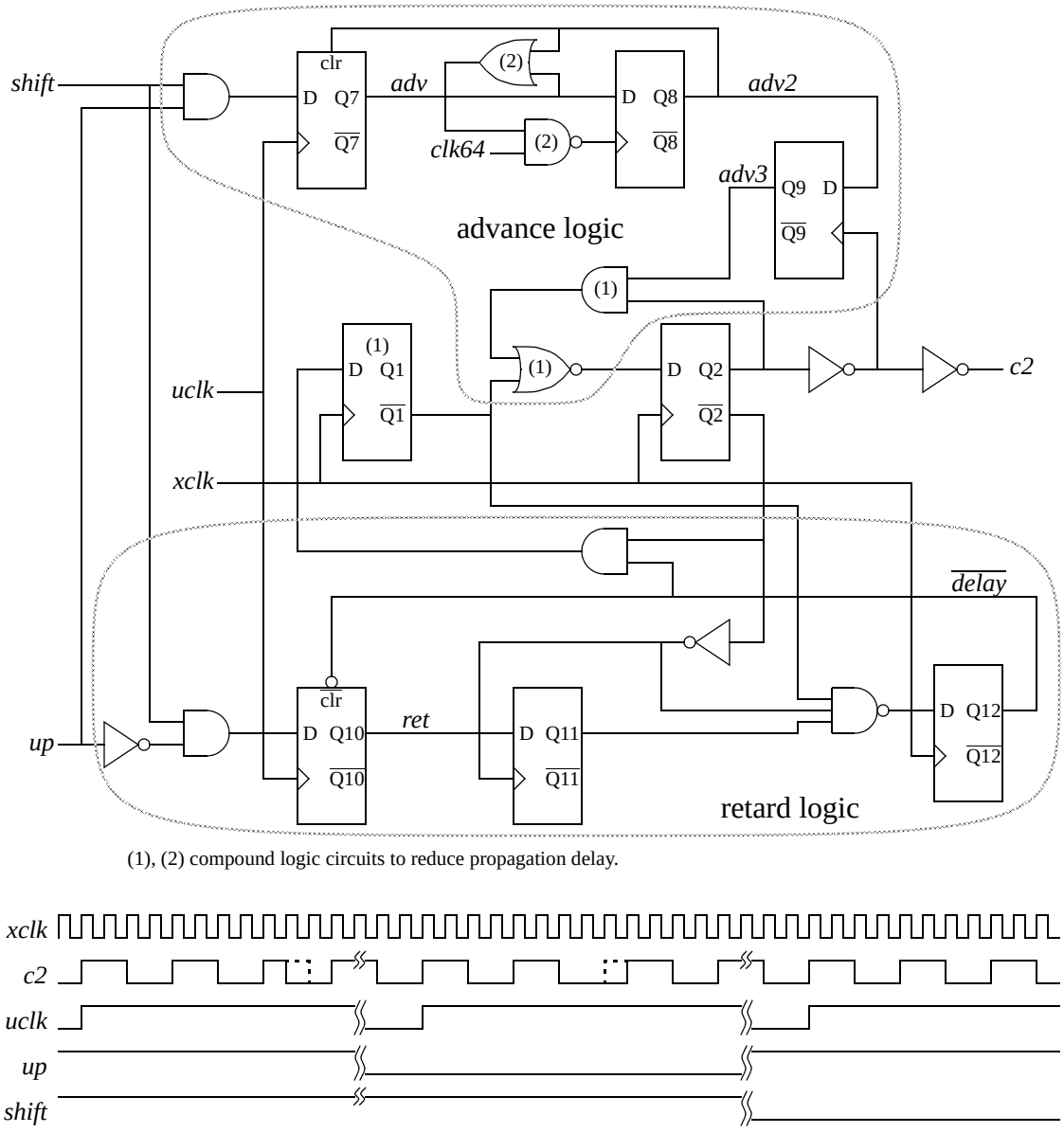


Figure 5-15 Simplified schematic of Digital Phase Shifter (DPS) logic.

sequence and a Walsh code. Any one of the sixty-four Walsh codes could be loaded externally into the onboard 64-chip Walsh-code registers (one per bank) that would determine which user signal to extract from the CDMA input signal. The PN and Walsh codes are pre-mixed (Exclusive-NOR) and applied to each sampling cell as a single PN-Walsh code.

5.4-5 Back-end interface circuits

A 10-bit pipelined ADC [2] was used to digitize the six correlator outputs allowing digital implementations of the acquisition and tracking algorithms as well as providing a digitally demodulated data output. Since each correlator[1] provided an output at a 1*MHz* rate, the ADC has to sample at 6 MS/s or higher in order to multiplex between the outputs. To simplify multiplexing and clock generation, 8 MS/s was chosen for the sampling rate of the ADC with the two extra conversions reserved for calibration. The ADC was fed by a 6-to-1-multiplexer / variable-gain-amplifier block. Since each correlator was configured as two interleaved (sample/integrate) banks the overall MUX was 12-to-1 and was split into two stages, 12-to-3 (IMUX) and 3-to-1 (OMUX), preceding and following three VGAs (see Fig. 5-2). The OMUX stage was actually a 4-to-1 multiplexer with an extra differential zero-volt input added to digitize any ADC input offset for calibration purposes. Three VGAs are used rather than one VGA running three times faster to lower the power dissipation since power is a square function of frequency. Due to the need for relative gain accuracy in the *Early*/*Late* correlation, no more than three VGAs could be used. Thus *I*-channel *Early*/*Late* correlators shared one VGA (VGA_I) as did the *Q*-channel *Early*/*Late* (VGA_Q) and the *I/Q Data* correlators (VGA_D).

Though the passive correlator outputs could have been fed directly to the ADC, a VGA was inserted for flexibility in testing inputs with different signal strengths. The

1. In this context, one correlator, e.g., $Early_I$ correlator, is made up of two passive correlators (one per bank) that require $1\mu s$ for sampling and $1\mu s$ for integrating thus each providing an interleaved output at $.5MHz$ for an aggregate output of $1MHz$.

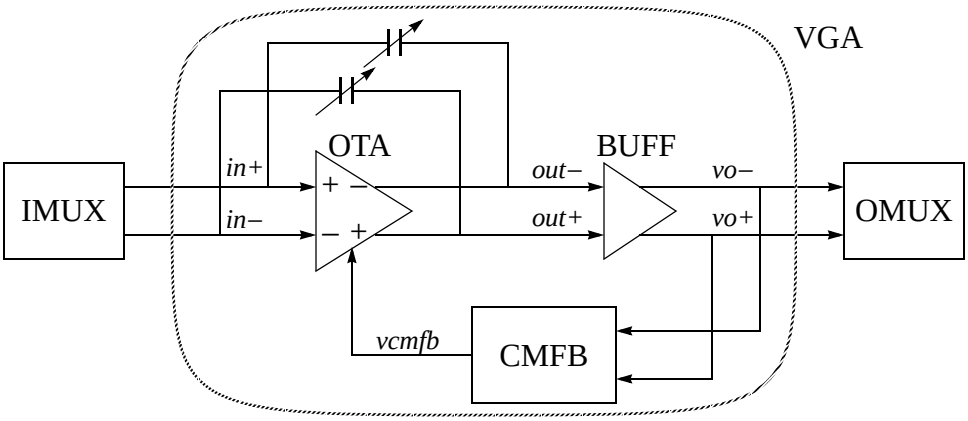


Figure 5-16 Variable gain amplifier.

VGA (Fig. 5-16) was implemented with a fully-differential *telescopic* operational transconductance amplifier (OTA), a source-follower buffer (BUFF) and common-mode feedback (CMFB) circuit. Since the correlator output signal was in the form of charge on a capacitor, a typical SC-type charge integrator configuration was used. Thus the gain was adjusted and determined by the size of the feedback capacitor. The feedback capacitors were binary-weighted to adjust the gain from 4 to 32 ($4pF$ to $.5pF$ in increments of $.5pF$).[1] A simplified schematic of the OTA, BUFF and CMFB circuits are shown in Fig. 5-17 and Fig. 5-18. The buffer was needed to drive the output multiplexer (OMUX) and the ADC input, so the feedback capacitors were connected to the outputs of the OTA to avoid the instability of a dual-dominant-pole situation.

Since each VGA amplified two correlator outputs during the integration phase ($64 \cdot T_c = 1\mu s$) of a given bank, the OTA should have had $500ns$ in which to settle. However, due to scheduling for the ADC, only $250ns$ (worst case) was available for the settling time of

1. The actual capacitor values were slightly higher (+3.6%) but the numbers used here were rounded down for notational convenience.

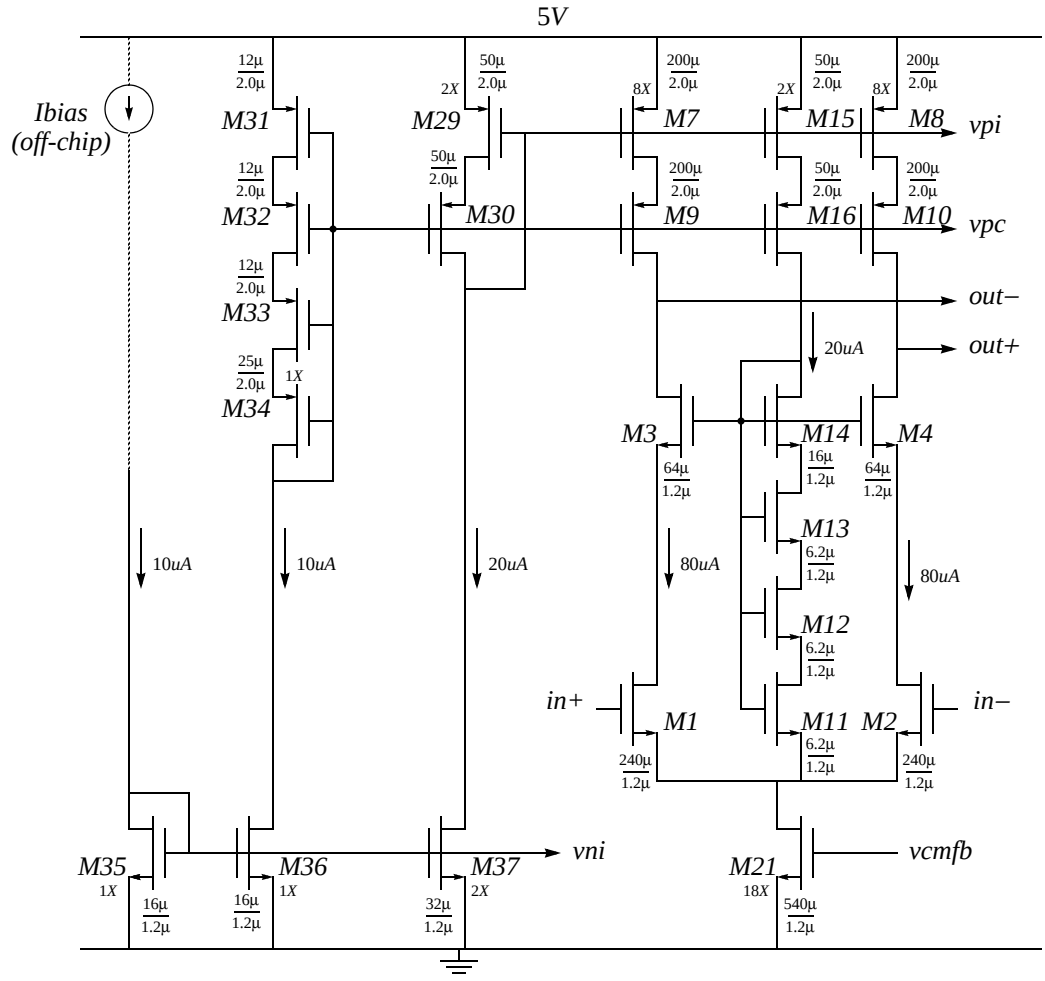


Figure 5-17 Schematic of OTA used in the VGA blocks.

the OTA. For 10-bit linear (single-pole exponential) settling[1], 7.6 time-constants (τ) are needed which translates to a desired τ of less than 33*ns*. Since the *telescopic* amplifier results in a single-pole response, τ is given by:

1. The VGA_I is connected to the first correlator ($Early_I$) while it is still taking samples so the OTA has closer to 500*ns* to settle. In addition, the integration process itself has an exponential settling nature as the integration capacitance increase with each successive sample, preventing any slew-rate limiting by the OTA.

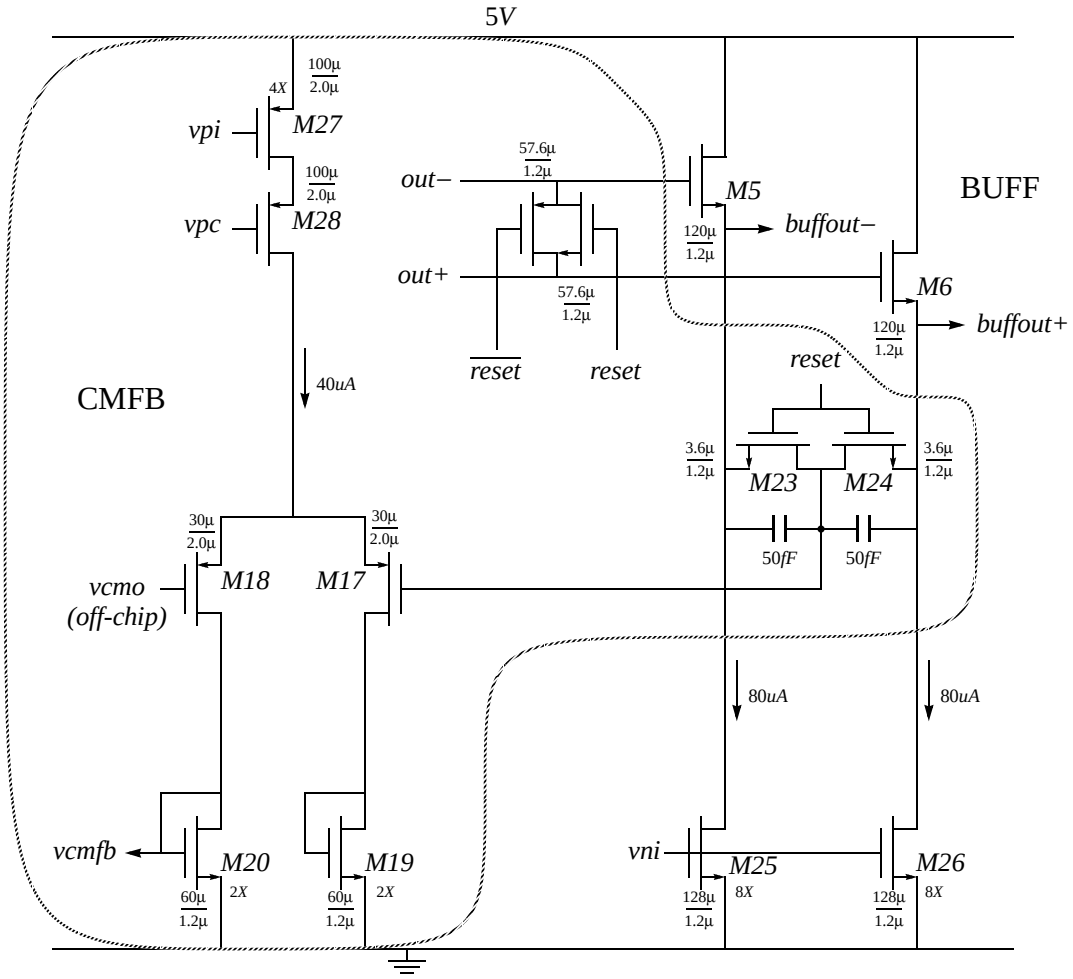


Figure 5-18 Schematic of BUFF and CMFB circuits used in the VGA blocks.

$$\tau = \frac{1}{g_m} \cdot \left(C_S + C_I + \frac{1}{f} \cdot C_L \right) \qquad (5 - 2)$$

where C_S is the total sampling-array capacitance (= 16*pF*), C_I is the input capacitance of the OTA (= 300*fF*), g_m is the transconductance of the OTA, f is the feedback factor and C_L is the load capacitance (= 400*fF*) at the output of the OTA (at *out±*). The feedback factor has the following relationship to the closed-loop gain (A_{cl}):

$$\frac{1}{f} = \frac{C_S + C_I + C_F}{C_F} \cong \frac{C_S + C_F}{C_F} = A_{cl} + 1 \qquad (5 - 3)$$

where C_F is the feedback capacitance (C_I is negligible). Thus (5 - 2) can be rewritten as:

$$\tau \cong \frac{1}{g_m} \cdot (C_S + (A_{cl} + 1) \cdot C_L) \qquad (5 - 4)$$

The worst case τ occurs for the largest value of A_{cl} (= 32), requiring a minimum transconductance of:

$$g_m \geq \frac{(16pF + 33 \cdot 400fF)}{33ns} = 885uA/V \qquad (5 - 5)$$

The OTA used a *continuous-time* CMFB employing a pair of 50*fF* capacitors to extract the output common-mode voltage. Between the end-of-integration and start-of-sampling times, the feedback and the 50*fF* capacitors were reset to zero volts. An external 1.5*V* reference (*vcmo*) set the output common-mode voltage.

5.5 References

[1]. S. Sheng, L. Lynn, J. Peroulas, K. Stone and I. O'Donnell, "A Low-Power CMOS Chipset for Spread-Spectrum Communications," in *ISSCC Dig. Tech. Papers*, Feb. 1996, pp. 346-347.

[2]. T. Cho and P. R. Gray, "A 10-b, 20 Msamples/s, 35mW pipeline A/D converter," *IEEE J. Solid-State Circuits*, vol. 30, no. 3, pp. 166-172, March 1995.

[3]. R. C. Dixon, *Spread Spectrum Systems*, 3rd ed. New York: Wiley, 1994.

[4]. J. Yuan and C. Svensson, "High-Speed CMOS Circuit Technique," *IEEE J. Solid-State Circuits*, vol. 24, no. 1, pp. 62-70, Feb. 1989.

[5]. C. M. Teuscher, "Software Simulation of the InfoPad Wireless Downlink," *Memo UCB/ERL M95/16*, Univ. Calif., Berkeley, March 1995.

Chapter 6

Experimental Results

The power performance of the prototype IC in comparison to a digital implementation is the main focus of this chapter. To make this comparison, it is necessary to establish a comparable performance level for the two implementations. The metric that will be used is the output signal-to-noise ratio (SNR_o) since it encompasses the dynamic range and noise performance of both architectures. Before presenting the *SNR* and power measurements, a brief description of the test setup will be given followed by some functional measurements of the prototype.

6.1 Test Setup

Fig. 6-1 is a diagram of the setup used to measure performance and power dissipation of the prototype IC. The key block in the setup was the arbitrary waveform generator (AWG) which created the baseband DS-CDMA signal. The AWG was a Tektronics model AWG2020 arbitrary waveform generator capable of producing two outputs at a 250*MHz*[1]

1. The particular AWG2020 used for the measurements was able to operate successfully up to 256*MHz*.

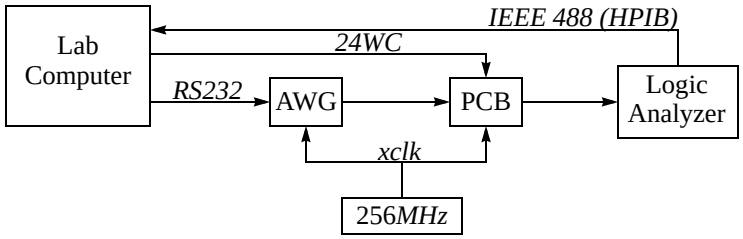


Figure 6-1 Prototype test configuration.

output rate. One output was designed to supply the *I*-channel data stream and the other to supply the *Q*-channel data stream to the IC inputs. Since the IC had differential inputs, there were high-frequency transformers on the printed circuit board (PCB) placed between each AWG output and the corresponding IC input to do the single-ended-to-differential conversion[1]. A lab computer was used to:

1. generate the DS-CDMA data file,
2. download the data file to the AWG via an RS232 serial link,
3. download the configuration file to the FPGA (Xilinx XC4008) on the PCB via a 24-bit wide control (24WC) bus,
4. download the three 3-bit VGA gain settings, 16-chip PN seed value and 64-chip Walsh code value to the PIC's onboard registers via the 24WC bus,
5. run the control program that interfaced to the FPGA to set acquisition thresholds and execute a master reset via the 24WC bus, and
6. retrieve and process data from the logic analyzer via an IEEE 488 (HPIB) bus.

1. This had to be altered as explained in Section 6.3 (See "Measurement notes" on page 179).

Depending upon the configuration file, the FPGA could be programmed to output any of its internal registers; the demodulated *I* and *Q Data* correlator outputs or the *Early* and *Late* correlator outputs. These outputs were captured by the logic analyzer (HP16500B with a 1-*Megaword* memory module) for subsequent data reduction by the lab computer. The FPGA also provided the three DLL feedback signals (*shift*, *up* and *delay2Tc*) to the IC. A common frequency source was used to feed both the AWG (input data timing) and the prototype which constrained testing to a synchronous recovery mode. Separate frequency sources could have been used to measure the capture range of the DLL but this was outside the scope of this research and had no bearing on the results.

Dedicated voltage regulators (LM 317) were used on the PCB to isolate the digital, analog and buffer[1] power sources for both the 5*V* (COR) and 3.3*V* (ADC) supplies. In addition to these six supplies, there were separate regulated supplies for the common-mode voltage (V_{cm} = 1.5*V*), the ADC full-scale reference (1*V*) and *xclk* (256*MHz*) squaring-buffers. Besides isolations, multiple supplies permitted separate power measurements of the various circuits. A lab regulated 5*V* supply was used to power the FPGA. Two adjustable current source regulators (LM 334) were used to set the bias currents for the amplifiers in the ADC and the VGAs.

6.2 Basic Functionality of the Prototype

To check the basic functionality and operation of the prototype, three measurements

1. The 5*V* and 3.3*V* buffer supplies powered the digital buffers that were used to drive signals off-chip.

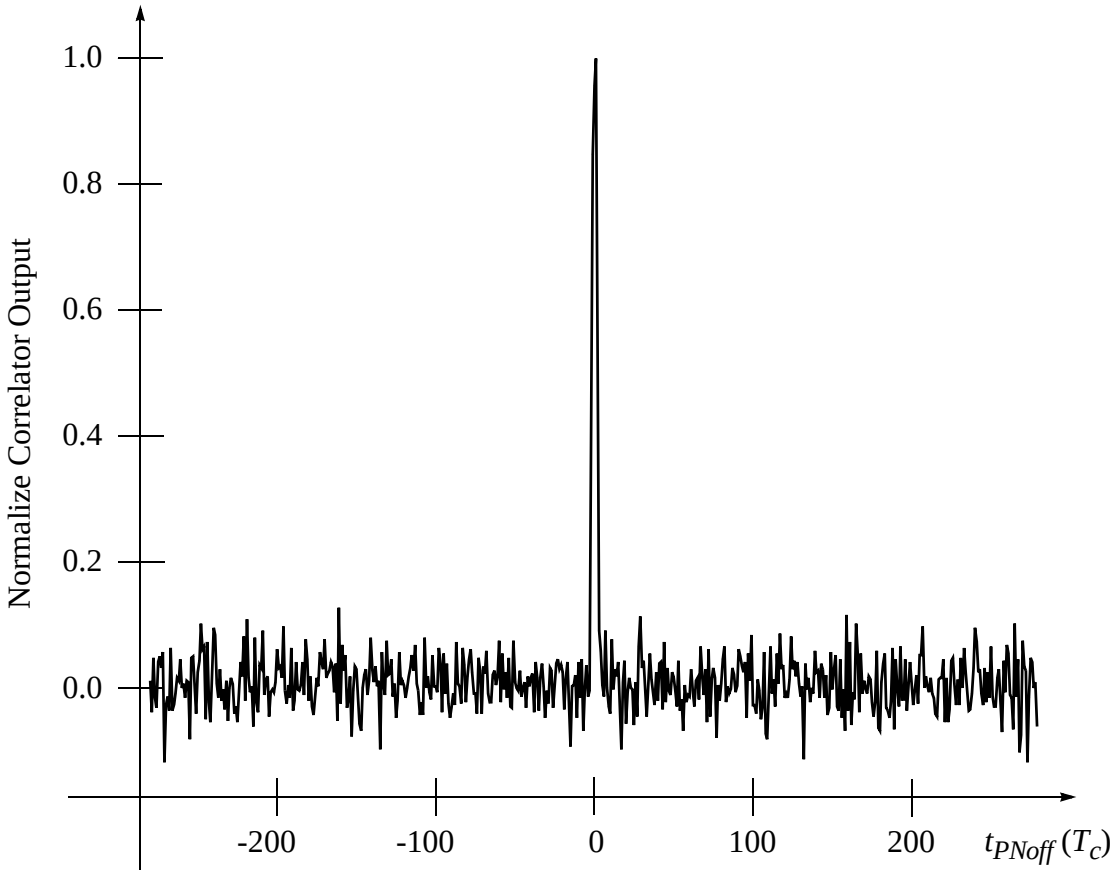


Figure 6-2 Correlator output from a serial search during the PN sequence acquisition phase (acquisition inhibited).

were made:

1. Acquisition search.
2. Frequency response.
3. Input to output transfer function.

Fig. 6-2 illustrates the acquisition search measurement which is a plot of the *Early* and *Late* correlators as a function of the offset time, t_{PNoff}, (in T_c), between the on-chip PN

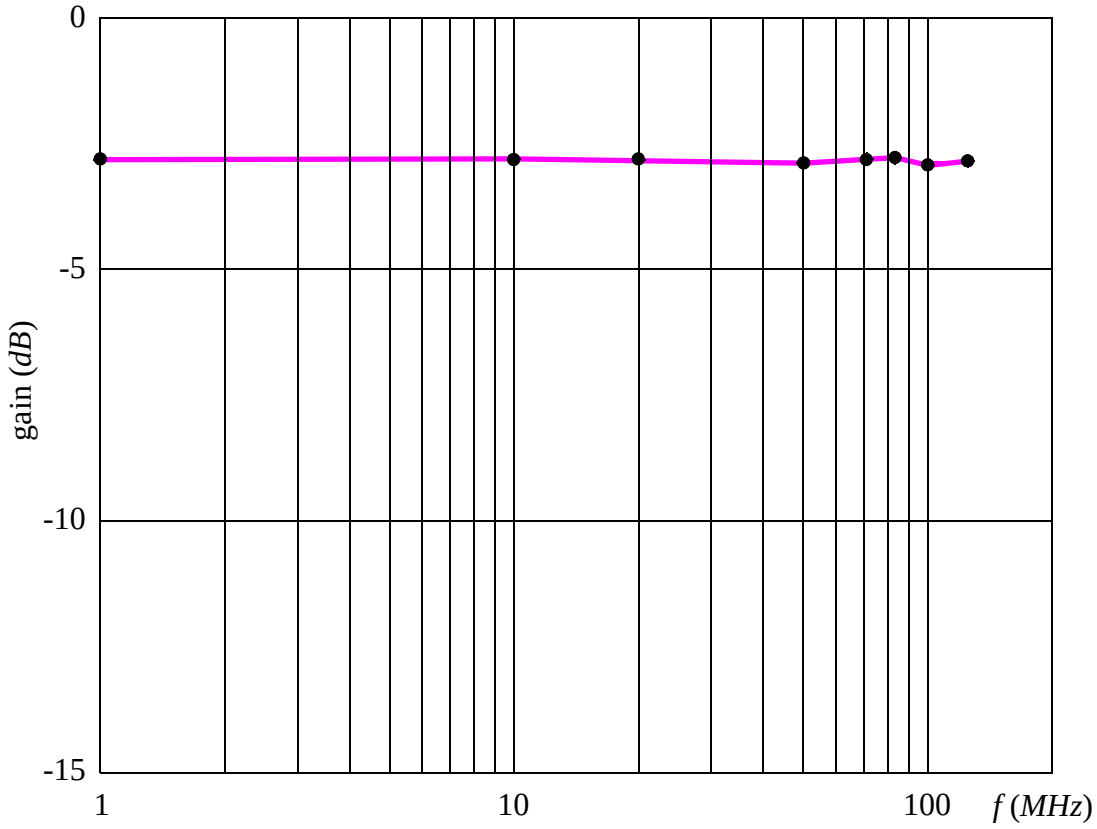


Figure 6-3 Frequency response of the *Data* demodulator gain.

sequence and the *pilot-tone* PN sequence. This plot was obtained by setting the acquisition threshold to the maximum value so that peak detection was never achieved thus inhibiting the transition from the acquisition to the tracking phase. The noise (peak-to-sideband) margin was less than the processing gain (30*dB*) because neither the correlation length (896 chips) nor the PN sequence (32,768) was maximal (2^M-1) as explained in Section 3.2.

The frequency response of the *Data* correlator's (demodulation) gain is shown in Fig. 6-3. This plot was obtained by varying the external frequency source (*xclk*) to change

the chip-rate of the input signal and correlator sampling operation. The gain response was flat out to the maximum sampling frequency of 128*MHz* (f_{xclk} = 256*MHz*). From (4 - 25), the voltage gain, A_v, of the passive correlator is given by:

$$A_v = \left(\frac{C_S}{C_S + C_{p1}}\right) \cdot \left(\frac{C_S}{C_S + C_{p3} + C_{po} + C_{ox2}}\right) \qquad (6 - 1)$$

From process data and layout dimensions, the following calculated values for the above capacitors are:

C_S = 250 *fF*[1] C_{p1} = 7 *fF* C_{p3} = 66 *fF*[2]

C_{po} = 10 *fF*[3] C_{ox2} = 7 *fF*

Using these values, the correlator gain is:

$$A_v = 0.73$$

The **measured** gain (at f_{xclk} = 256*MHz)* was:

$$A_v = 0.72 \ \ (-2.85\ dB)$$

which agrees well with the calculated value.

The linearity of the *Data* correlator is illustrated in Fig. 6-4, which is the transfer function characteristic of the *Q*-channel input for f_{xclk} = 2*MHz* and 256*MHz*.

1. With fringe capacitance, the value is closer to 259*fF* but for convenience a rounded value of 250*fF* is used.
2. Includes bottom-plate parasitic of C_S (44*fF*) and C_{db} of *M2*, *M3* and *M4* (see Fig. 4-12).
3. Per cell metal capacitance of the output bus.

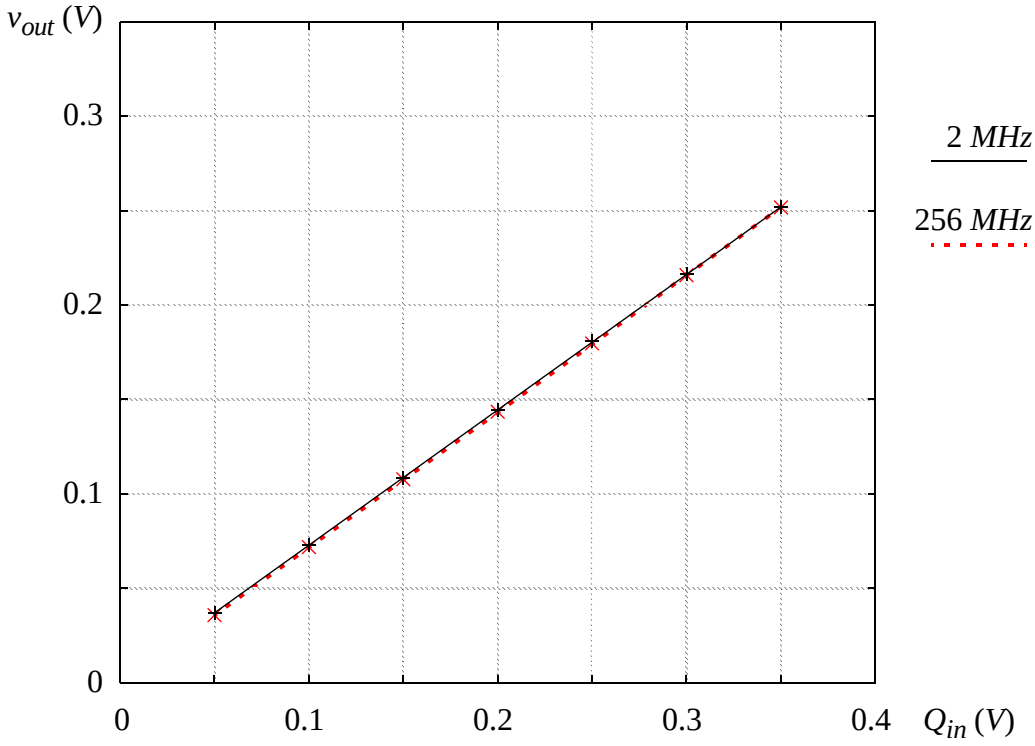


Figure 6-4 Input / output transfer function of *Q*-channel *Data* Correlator (for *xclk* = 2*MHz* and 256*MHz*).

6.3 System Signal-to-Noise (*SNR)* Measurements

Ultimately the goal of this prototype is to be able to make a comparison in the power dissipation of the two architectures of Fig. 6-5. Since the sampling rate requirement is given, only the resolution requirement of the ADC in the digital correlator implementation is needed to estimate its power. To indirectly arrive at the equivalent resolution required by a front-end ADC, a system comparison of the signal-to-noise performances of the two architectures was made.

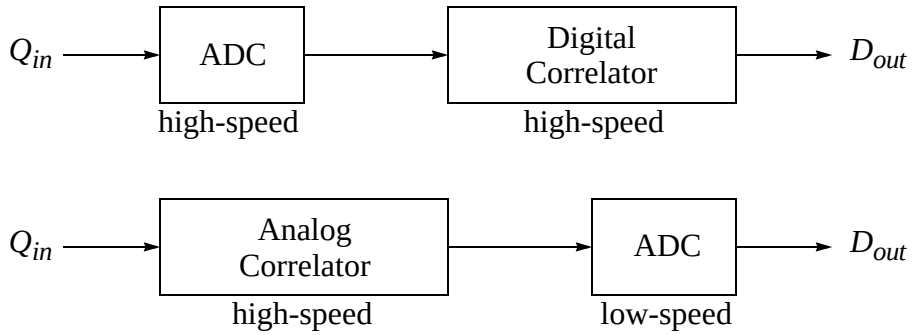


Figure 6-5 Digital and analog correlator architectures.

6.3-1 *SNR* of an ideal digital correlator

In this ideal case, the only noise contribution is assumed to be that of the quantization noise due to granularity of the A/D conversion process. If it is also assumed that the noise amplitude has a uniform distribution, then the quantization noise power, σ_{qn}^2, is (from [1]):

$$\sigma_{qn}^2 = \frac{\Delta^2}{12} \tag{6 - 2}$$

where Δ is the step (granularity) size of the quantization. So for a full-scale range of F and a resolution of B bits:

$$\Delta = \frac{F}{2^B} \tag{6 - 3}$$

The signal power of the demodulated data will depend on the number of users and how the power is partitioned among those users and the pilot tone. For simplicity, the test case for the comparison will consist of one user plus the pilot tone where each is allotted half the

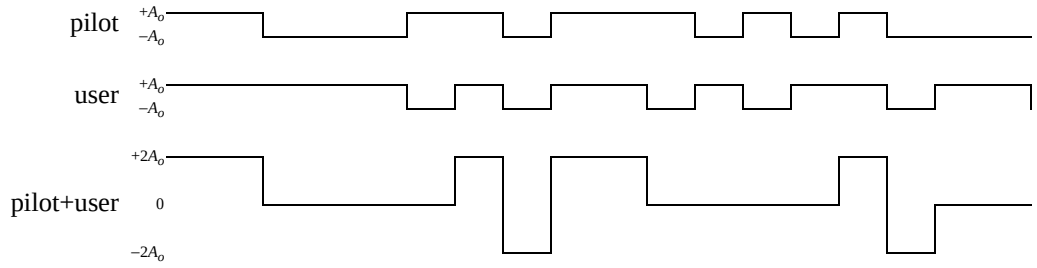


Figure 6-6 A sample DS-CDMA baseband test signal.

transmit power (see Fig. 6-6). For NRZ[1] data with a signal swing of $\pm A_o$, the mean-squared signal power for the user, $\overline{s_u^2(t)}$, is:

$$\overline{s_u^2(t)} = A_o^{\,2} \tag{6 - 4}$$

Thus the quantization-noise-limited[2] input-referred *SNR* for an ideal digital correlator, SNR_{id}, is:

$$SNR_{id} = \frac{\overline{s_u^2(t)}}{\sigma_{qn}^2} = \frac{12 \cdot A_o^{\,2} \cdot 2^{2B}}{F^2} = \left(\frac{4A_o}{F}\right)^2 \cdot \frac{3}{4} \cdot 2^{2B} \tag{6 - 5}$$

or in *dB*:

$$\begin{aligned} SNR_{id}\Big|_{dB} &= 20 \cdot \log\left(\frac{4A_o}{F}\right) + 10 \cdot \log(0.75) + B \cdot 20 \cdot \log(2) \\ &= \left(\frac{4A_o}{F}\right)\Bigg|_{dB} - 1.25dB + B \cdot 6.02dB \end{aligned} \tag{6 - 6}$$

1. Non-return to zero.
2. This is the maximum input *SNR* assuming that the *SNR* of the input signal is not the limiting quantity.

After correlation, the user's signal power remains unchanged, but the quantization noise is reduced by the processing gain (G_P) increasing the *SNR*, as described in Section 3.5, so the output *SNR* is:[1]

$$\begin{aligned} SNR_{od}\Big|_{dB} &= SNR_{id}\Big|_{dB} + 18.06dB \\ &= \left(\frac{4A_o}{F}\right)\Bigg|_{dB} + 16.81dB + B \cdot 6.02dB \end{aligned} \tag{6 - 7}$$

The first term represents the input signal magnitude in *dB* with respect to *F*. The 0*dB* input value is when the input peak-to-peak value is equal to the ADC's full-scale, i.e., when:

$$A_o = \frac{F}{4} \tag{6 - 8}$$

6.3-2 *SNR* of prototype analog correlator

The output *SNR* of the prototype analog correlator, SNR_{oa}, was calculated by taking measurements of the demodulated output (from the FPGA). With the user data bits set to one value (one or zero), computing the signal power was equivalent to calculating the mean value of the demodulated *Data* output and squaring it. Similarly, squaring the standard deviation of the output, i.e., calculating its sample variance, yielded the noise power from which the SNR_{oa} could then be determined. For these calculations, an all ones data pattern and a sample size greater than 8000 data bits (512,000 chips) per data point was used. Table 6-1 lists the SNR_{oa} measurements and calculations.

1. For a correlation length of 64 (10·log(64)).

TABLE 6-1 SNR_{oa} Measurements(1)

A_o (user)(2)	A_o (referenced to 1.5V)	Sample size (bits)	$\overline{s_u(t)}$ (3)	σ_{qn}^2	σ_{qn}	SNR_{oa}
12.5 *mV*	-41.6 *dB*	8192	18.6	2.4	1.5	21.6 *dB*
25.0 *mV*	-35.6 *dB*	8192	37.0	2.6	1.6	27.2 *dB*
50.0 *mV*	-29.5 *dB*	8192	72.7	1.4	1.2	35.8 *dB*
125.0 *mV*	-21.6 *dB*	8185	180.6	2.8	1.7	40.6 *dB*
250.0 *mV*	-15.6 *dB*	8185	361.7	4.5	2.1	44.6 *dB*
350.0 *mV*	-12.6 *dB*	8186	504.6	5.6	2.4	46.6 *dB*

(1). Conditions: VGA gain = 4; f_{xclk} = 256MHz; applied offset = -85mV
(2). Total input (user + pilot) amplitude was twice as large.
(3). Digital output value (unitless). The conversion factor was 2.0V/1024 = 1.95mV/unit.

Using the equations from the preceding section and the results in the table, the SNR_{od} curves are plotted along with the SNR_{oa} data in Fig. 6-7. This plot shows that the analog correlator has performance comparable to a digital correlator with a front-end ADC having quantization noise at the 7 to 8 bit level. Unlike the SNR_{oa} data, the SNR_{od} curves do not include the spectrum folding noise due to aliasing of the wideband CDMA input signal. If this noise was factored into the SNR_{od} curves, the analog correlator performance would be closer to the 8 to 9 bit quantization level.

6.3-3 Measurement notes

To get good SNR_{oa} measurement results, it was necessary to create as accurate an input DS-CDMA signal as possible so that the signal's own *SNR* did not limit the SNR_{oa}. Reflections and band-limiting affects by the input transformers and PCB layout made it necessary to use both AWG outputs to create a single differential signal source. A prob-

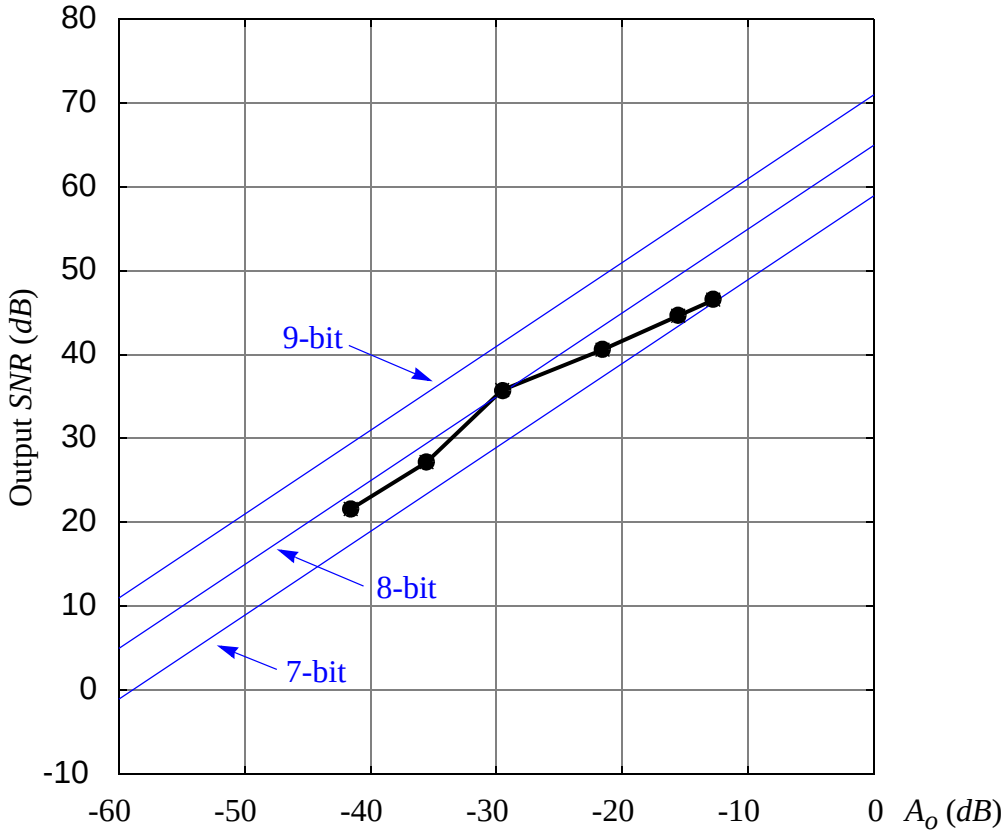


Figure 6-7 Ideal SNR_{od} curves (for 7, 8 and 9 bits) versus measured SNR_{oa} data.

lem with the package used to house the prototype IC compromised the *I*-channel input so all measurements were made on the *Q*-channel. Even though I_{in} was not used (inputs shorted), the I-channel correlators still dissipated the same amount of power as the *Q*-channel correlators so power measurements reflect dissipation from all six correlators.

There was also approximately 80*mV* of offset detected in all correlators. Though not confirmed as the cause, there was a mismatch in the layout of the differential sampling cell where, unlike the positive summing node, the negative summing node was not isolated from a switching signal. Though this input offset was constant, mixing by the PN-Walsh

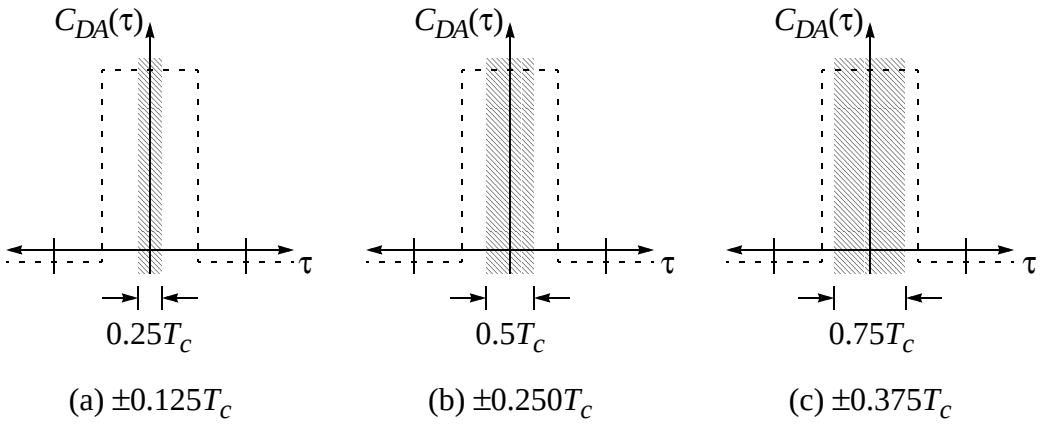


Figure 6-8 *Data* correlator sampling-uncertainty window due to *dithering error* for three different phase correction amounts.

modulation converted it to noise, increasing the output noise power. An offset was applied to the input to minimize this noise component when making the SNR_{oa} measurements.

Another variation in the measurements was caused by the *dithering* action of the tracking (DLL) loop and the finite sampling bandwidth of the correlator inputs. This ***dithering error*** is a result of the discrete-time nature of the tracking system so it affects both the analog correlator and the ADC front-end of the digital correlator equally. As was shown in Fig. 3-21, in the tracking state, the *Early* and *Late* correlations cannot be equal so the loop alternates tracking adjustments between advancing and retarding $0.25 \cdot T_c$.[1] This causes an uncertainty in the sampling point of the *Data* correlator dependent on the adjustment amount. Fig. 6-8 shows the *Data* correlator sampling window for different adjustment values assuming that the adjustment amount is greater than the correction needed due to timing drift and update frequency. The problem occurs when the sampling

1. This correction amount $(T_c/4)$ is specific to the prototype system.

bandwidth is not large enough for sampling points near the front edge of the uncertainty window, i.e., error is introduced due to lack of settling time. There was a 1*dB* to 2*dB* difference in the SNR_{oa} values due to this error. The results shown above used the higher *SNR* value.

6.4 Power Measurements

A breakdown of the power consumption for the prototype IC is shown in Table 6-2. From this table, the dynamic power makes up 81% of the total power.[1] The correlators

TABLE 6-2 Power Consumption(1)

Blocks	Dynamic	Static	V_{dd}
6 Correlators(2)	24 *mW*	---	5.0*V*
PN Gen, DPS & MUX (2)	33 *mW*	---	5.0*V*
VGA_I, VGA_Q and VGA_D	---	6 *mW*	5.0*V*
ADC (10-bit 8*MS/s*)	4 *mW*	8 *mW*	3.3*V*
Total	61 *mW*	14 *mW*	

(1). f_{xclk} = 256*MHz*

(2). How the power is partitioned between these 2 groups is estimated from computer simulations.

consumed an average of 4*mW* each[2] (this included the clock power to sequence the switches). The overhead control logic (PN Gen, DPS and MUX), which consumed 33*mW*, was digital logic so its power scaled as CV^2f. If the overhead logic is included, the corre-

1. Thus the overall power can benefit from supply voltage scaling.
2. Per correlator power was 6.4*mW* but with the bank select duty cycle of 62.5% (see Section 5.3-3), the average was 4*mW*.

lators required 9.5*mW* each to operate at 256*MHz*. This compares to 5.3*mW* per correlator[1] for a similar digital correlator [2] which represents a 25*mW* power difference in a six-correlator implementation. The 10-bit ADC required 12*mW* when sampling at 8*MS/s*.

To implement an equivalent DS-CDMA demodulator in the digital domain would require two (*I* and *Q*) ADCs operating at 128*MS/s* each. To match the performance of the analog correlator, these ADCs would need 7 to 8 bits of resolution. Table 6-3 is a list of

TABLE 6-3 High Speed ADCs

Resolution (bits)	Sampling Rate	Power for 2 ADCs	Technology	Reference
10 bits	100 *MS/s*	2.2 *W*	1.0µ CMOS	3
10 bits	100 *MS/s*	1.9 *W*	0.8µ BiCMOS	4
8 bits	650 *MS/s*	1.7 *W*	1.0µ Bipolar	5
8 bits	300 *MS/s*	6.6 *W*	2.5µ Bipolar	6
8 bits	200 *MS/s*	1.0 *W*	0.8µ BiCMOS	7
8 bits	150 *MS/s*	0.7 *W*	1.0µ BiCMOS	8
8 bits	100 *MS/s*	1.6 *W*	2.0µ Bipolar	9

high-speed ADCs that are suitable for this comparison. Using the fact that a six-correlator analog block is 25*mW* more than an equivalent digital one, a power comparison can be made based solely on the ADC(s) required by each approach. From Table 6-3 the 12*mW* post-correlation ADC is about two orders of magnitude lower power than the pre-correlation ADCs required by the digital approach. Thus, based on this comparison, the analog pre-processing approach applied to the baseband demodulation of a DS-CDMA signal could be a lower-power alternative to a comparable digital approach.

1. Extrapolated to 128MS/s.

6.5 References

[1]. F. G. Stremler, *Introduction to Communication Systems,* 3rd ed. Reading, Mass. USA: Addison-Wesley Pub. Co., 1990.

[2]. S. Sheng, L. Lynn, J. Peroulas, K. Stone, and I. O'Donnell, "A low-power CMOS chipset for spread-spectrum communications," *ISSCC Digest of Technical Papers*, Feb. 1996, pp. 346-347.

[3]. K. Y. Kim, N. Kusayanagi and A. A. Abidi, "A 10-b, 100-MS/s CMOS A/D Converter," *IEEE J. Solid-State Circuits*, vol. 32, no. 3, pp. 302-311, Mar. 1997.

[4]. K. Sone, Y. Nishida and N. Nakadai, "A 10-b 100-Msample/s Pipelined Subranging BiCMOS ADC," *IEEE J. Solid-State Circuits*, vol. 28, no. 12, pp. 1180-1186, Dec. 1993.

[5]. J. van Valburg and R. J. van de Plassche, "An 8-b 650-MHz Folding ADC," *IEEE J. Solid-State Circuits*, vol. 27, no. 12, pp. 1662-1666, Dec. 1992.

[6]. Y. Nejime, M. Hotta and S. Ueda, "An 8-b ADC with Over-Nyquist Input at 300-Ms/s Conversion Rate," *IEEE J. Solid-State Circuits*, vol. 26, no. 9, pp. 1301-1308, Sept. 1991.

[7]. Y. Nishida, K. Sone, K. Amano, S. Matsuba and A. Yukawa, "An 8-bit 200 Ms/s 500 mW BiCMOS ADC," in *Proc. Custom Integrated Circuits Conf.*, May 1995, pp. 10.4.1-10.4.4.

[8]. C. W. Moreland, "An 8b 150MSample/s Serial ADC," *ISSCC Digest of Technical Papers*, Feb. 1995, pp. 272-273, 378.

[9]. R. J. van de Plassche and P. Baltus, "An 8-b 100-MHz Full-Nyquist Analog-to-Digital Converter," *IEEE J. Solid-State Circuits*, vol. 23, no. 6, pp. 1334-1344, Dec. 1988.

Chapter 7

Conclusions

Progress in the electronics field continues to be towards higher performance and functionality. This has been accomplished through integrated circuits with higher processing speeds and transistor count. A necessary by-product of both these progressions is the increase in power dissipation (raising reliability and lifetime issues with the escalating internal die temperatures) and consumption (taxing the battery life of portable electronics products). These power concerns threaten continued progress in electronic development but have, thus far, been kept at bay by advances in process technology. However, demands by today's personal communication market motivate other non-technology solutions to further address the power situation.

This research presents a non-process advancement approach to lowering power. In particular, this research demonstrates how lowering the signal bandwidth and dynamic range through analog signal processing, prior to the A/D conversion, can lower the converter power, and thus, the overall system power. In the course of this research, the following contributions have resulted: **(1)** A first-order analysis of the power dependence on

resolution and sampling rate for high-speed A/D converter architectures was derived. It was found that power was proportional to sampling rate for most architectures until the sampling rate approached the device f_T, at which point power had to increase asymptotically toward infinity. The power increase as a function of converter resolution was highly dependent upon architecture. Its dependence varied from an exponential function of resolution in the Flash converter case to slightly more than linear in a 1*bit*-per-stage Pipelined structure. **(2)** Developed a *passive charge-error cancellation* technique for switched-capacitor integrators that enabled implementation of a high-speed low-power *passive* (amplifier-free) PN correlator. **(3)** Low-power digital design techniques were developed to further optimize the power without degrading the operating speed. The predominant theme was *load-shifting* which minimized the circuits that operated at the highest frequencies. *Load-shifting* realizations included *multi-tapped* single timing-SR-chain, local control-signal generation with *automatic-state* decoding, and *double-latched* PN generator. Clock duty-cycle reduction was also used to lower the effective operating frequency. **(4)** Demonstrated that a DS-CDMA baseband recovery integrated circuit, fabricated in a 1.2μm double-metal double-poly CMOS process, which used analog processing prior to the A/D conversion is capable of operating at 128*MS/s* (*I* and *Q* channels at 64*Mchips/s*) with an output *SNR* of 46.6*dB* while dissipating 75*mW*. This compares favorably to the power of two ADC's (*I* and *Q* channel) required for a digital implementation.

The results of this research assume that the power of performing a task using analog signal processing is approximately equal to that of using digital signal processing,[1] i.e, an

ADC power comparison reflects a system power comparison. This is somewhat true today but as process technology improves allowing supply voltages to drop, the power of performing a given task in the analog domain increases [2] while that of equivalent digital function decreases or remains the same. This affect diminishes the applicability of these results to future technology suites and should be considered in any extension of this work. Nonetheless, the results appear significant enough to predict a continued power benefit of using analog circuit techniques in the region closest to the *converter boundary* of Fig. 1-1.

7.1 References

[1]. G. T. Uehara, “Circuit Techniques and Considerations for Implementation of High Speed CMOS Analog-to-Digital Interfaces for DSP-Based PRML Magnetic Disk Read Channels,” *Memo UCB/ERL M93/96*, Univ. Calif., Berkeley, August 1993.

[2]. A. M. Abo, "Design for Reliability of Low-voltage, Switched-capacitor Circuits,” *Memo UCB/ERL M99/25*, Univ. Calif., Berkeley, May 1999.

1. Excluding the post-ADC and pre-ADC power, respectively.

Appendix A

Addendum to Chapter 2

A.1 Velocity Saturation Equations for CMOS Transistors

CMOS transistors experience velocity saturation for short channel lengths and high electric fields along the channel. This modifies the normal *forward-active* square-law drain current formula as follows [1]:

$$I_d = \frac{k'}{2} \cdot \left(\frac{W}{L}\right)\left(\frac{(V_{gs} - V_t)^2 (E_{sat} \cdot L)}{(V_{gs} - V_t) + (E_{sat} \cdot L)}\right) \qquad \text{(A - 1)}$$

where E_{sat} is the critical electric field[1] when the transistor enters the velocity saturation regime. In all analyses in this dissertation, two limiting cases shall be considered; *long-channel* case:

$$I_d = \frac{k'}{2} \cdot \left(\frac{W}{L}\right)(V_{gs} - V_t)^2 \qquad (V_{gs} - V_t) \ll (E_{sat} \cdot L) \qquad \text{(A - 2)}$$

and *short-channel* case:

1. [1] uses E_c for this field.

$$I_d = \frac{k'}{2} \cdot \left(\frac{W}{L}\right)(V_{gs} - V_t)(E_{sat} \cdot L) \qquad (V_{gs} - V_t) \gg (E_{sat} \cdot L)$$
$$= \frac{k'}{2} \cdot W(V_{gs} - V_t)E_{sat} \tag{A - 3}$$

The second form of the short-channel drain current equation (A - 3) suggests that I_d is independent of channel length. This is only true as long as the channel length stays small enough to satisfy the short-channel criterion that $(V_{gs}\text{-}V_t) >> (E_{sat} \cdot L)$. The respective formulas for transconductance (g_m) are:

$$g_{ml} = k' \cdot \left(\frac{W}{L}\right)(V_{gs} - V_t) = \frac{2I_d}{(V_{gs} - V_t)} = \sqrt{2I_d \cdot k' \cdot \left(\frac{W}{L}\right)}$$
$$g_{ms} = \frac{k'}{2} \cdot W \cdot E_{sat} = \frac{I_d}{(V_{gs} - V_t)} \tag{A - 4}$$

Note that the short-channel g_m is dependent only upon W (again with the restriction on L satisfying the short-channel criterion). The device cutoff frequencies (f_T) are, respectively:

$$f_{Tl} = \frac{\mu(V_{gs} - V_t)}{2\pi L^2}$$
$$f_{Ts} = \frac{\mu E_{sat}}{4\pi L} \tag{A - 5}$$

Unlike the long-channel case, the short-channel f_T is not dependent upon the bias condition, $(V_{gs}\text{-}V_t)$. And the formulas for the transistor drive voltages, $(V_{gs}\text{-}V_t)$, are, respectively:

$$(V_{gs} - V_t)_l = \sqrt{\left(\frac{I_d}{W}\right) \cdot \frac{2L}{k'}}$$
$$(V_{gs} - V_t)_s = \left(\frac{I_d}{W}\right) \cdot \frac{2}{k' \cdot E_{sat}} \qquad \text{(A - 6)}$$

In both cases, (V_{gs}-V_t) is a function of the drain current density $\left(\frac{I_d}{W}\right)$.

Derivations using short-channel equations will note any significant long-channel differences and vice versa.

A.2 Peak SNDR

One performance specification of an ADC is the signal-to-noise-plus-distortion ratio (*SNDR*) [2] as measured at the output of the converter. For an ideal converter, there would be no distortion and the only noise contribution would come from quantization errors due to the finite step sizes of the digital codes, i.e., the difference between the actual input voltage value and the discrete voltage step represented by the digital code value. Thus this quantization noise, N_q, decreases with the increasing resolution of the converter. For uniformly spaced step sizes and assuming that the probability density function of the quantization error is uniformly distributed over a step size, the quantization noise power is [3]:

$$N_q = \frac{\Delta^2}{12} \qquad \text{(A - 7)}$$

where Δ is the step size given by:

$$\Delta = \frac{V_{refHI} - V_{refLO}}{2^B} = \frac{F}{2^B} \tag{A - 8}$$

B is the number of bits of resolution and *F* is the full-scale input range for the ADC. For a given resolution, the step sizes are fixed so the quantization error power is bounded and independent of the input signal amplitude. This implies that as the input signal increases, the *SNDR* will increase. For a pure sinewave input, v_{in}, i.e.:

$$v_{in} = A_o \sin(2\pi f_{in} t) \tag{A - 9}$$

the input signal power, S_i, would be:

$$S_i = \frac{A_o^2}{2} \tag{A - 10}$$

so the *SNDR*[1] for this quantization-noise-only case would be:

$$SNDR = \frac{S_i}{N_q} = \frac{A_o^2 \cdot 12 \cdot 2^{2B}}{2 \cdot F^2} = \left(\sqrt{\frac{3}{2}} \cdot \frac{A_o \cdot 2^B}{(F/2)} \right)^2 \tag{A - 11}$$

or in decibels:

$$\begin{aligned} SNDR|_{dB} &= 20\log(A_o) - 20\log\left(\frac{F}{2}\right) + 20\log(2^B) + 20\log\left(\sqrt{\frac{3}{2}}\right) \\ &= 20\log(A_o) - 20\log\left(\frac{F}{2}\right) + B \cdot 6.02dB + 1.76dB \end{aligned} \tag{A - 12}$$

1. In this case, it would actually be *SNR* since we are assuming no distortion but the *SNDR* notation is retained for generality.

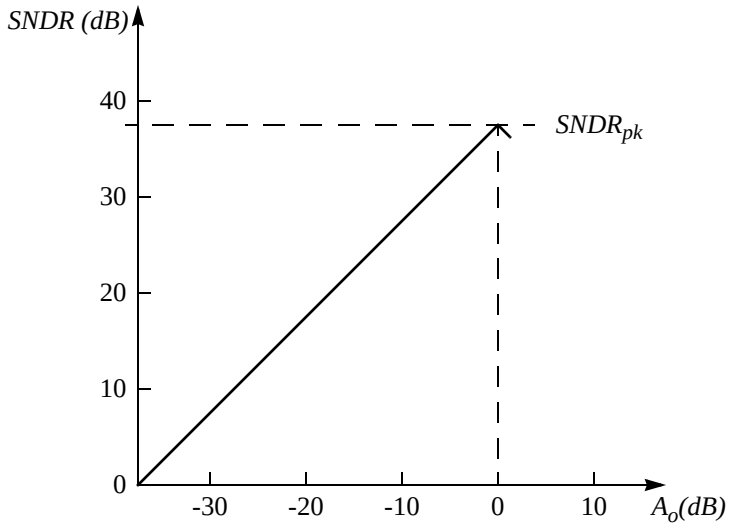


Figure A-1 Signal-to-quantization-noise curve for ideal 6-bit ADC with sinusoidal input.

The last three terms of (A - 12) are constant for a given full-scale range (F) and resolution (B), so the *SNDR* for this ideal case is proportional to the input amplitude, offset by the value of the last three terms, and thus increases as the input signal level increases. This is shown in Fig. A-1 for a 6-bit ideal ADC example, where the *SNDR* is plotted against the input amplitude (A_o).[1] When the input amplitude peak-peak signal exceeds the full-scale range, the digital output representation of the sinusoidal input experiences *clipping*[2] resulting in harmonic distortion and a decrease in SNDR. Thus conservatively, the peak *SNDR* ($SNDR_{pk}$) occurs when the input sinusoidal peak-to-peak amplitude ($2A_o$) equals the full-scale value (F) and is given by (from (A - 12)):

1. The 0*dB* value for A_o is taken as the point when the peak-to-peak value of the sinewave is equal to the ADC's fullscale value, i.e., when $A_o = F/2$.
2. This can equivalently be viewed as increasing the quantization error, as the error at the extremes exceeds the step size, Δ, when the signal swings beyond the full-scale range. In this case the quantization noise power exceeds the value given by (A - 7).

$$SNDR_{pk}\Big|_{dB} = B \cdot 6.02dB + 1.76dB \quad \text{(A - 13)}$$

This equation is often rearranged to determine an effective number of bits, B_{eff}, for an actual converters based on measured values of their $SNDR_{pk}$.

The fact that the input is a sinusoidal waveform means that a number of different $SNDR_{pk}$ values can be obtained corresponding to different input test frequencies, f_{in}. Since timing errors (predominantly jitter and skew) translate to larger voltage sampling errors for higher values of f_{in}, the frequency responses of $SNDR_{pk}$ and B_{eff} tend to fall-off at higher frequencies. Thus a converter designed for, say, 6 bits may have a B_{eff} of 5 or less at the Nyquist frequency ($f_{in} = f_S/2$). Jitter in the sampling clock can be caused by digital switching noise in the substrate or power supply. Sampling clock skew originates from clock buffering and metal routing differences. Both of these timing errors are aggravated exponentially by increasing the resolution of the converter. A front-end sample-and-hold (S/H) amplifier can eliminate or reduce both these errors but at the cost of additional power and area.

A.3 Power-Speed Relationship of CMOS Auto-zero Inverter Comparators

The speed of the Flash converter is dependent upon the inverter chain's evaluation time of the difference between V_{in} and V_{ref}. Since the inverters are configured as linear gain stages during this evaluation, small-signal analysis (for all but possibly the last stage) will predict their performance. For the last stage with an output swing of several hundred

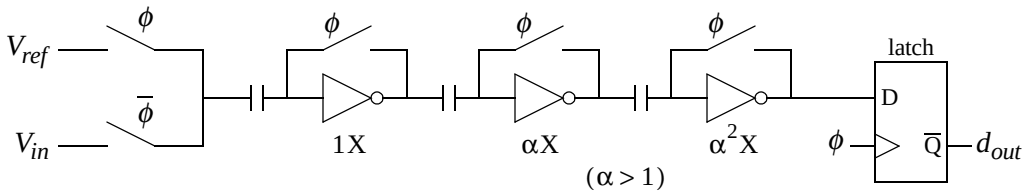


Figure A-2 Auto-zero chopper inverter CMOS comparator (three inverter example).

millivolts (in order for correct binary interpretation by the latch), it is best described by large signal equations. A CMOS inverter gain stage is capable of at least a gain of 20 which suggests that the next-to-last stage is still within the domain of small-signal analysis.

The inverter stages are commonly scaled up (α factor in Fig. A-2) to optimize their driving capacity and propagation delay. The value for α is driven by the number of inverter stages needed, the capacitive load of the latch and the input load capacitance as seen by V_{in}. The speed and power of an inverter is determined by the sizes of its transistors and the capacitive load that it must drive (see Fig. A-3). Assuming the inter-stage coupling capacitor is large enough to be a short circuit, Fig. A-3 can be used to calculate the speed of one inverter stage. The following assumptions will be made for this analysis; (1) all transistors of an inverter stage including the auto-zero switch (*M3* and *M4*) scale together, (2) the next stage transistors are α times larger than the current stage, and (3) the V_{gs} and V_t of *M1* equals those of *M2*. As with most CMOS digital circuits, the short-channel criterion is met, so the following analysis is carried out using the short-channel equations.

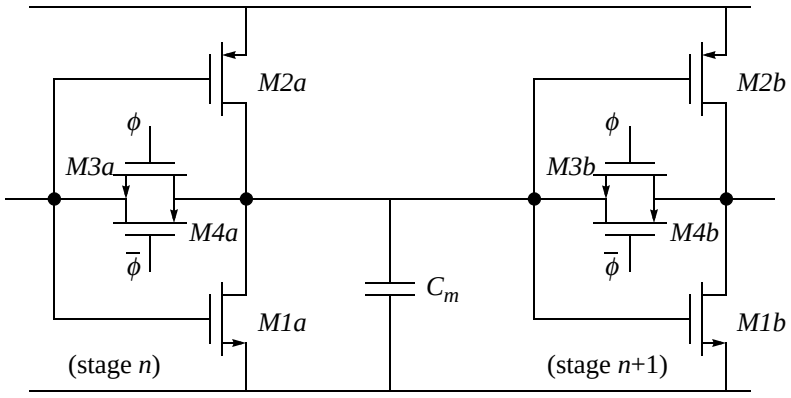


Figure A-3 Two Inverter gain-stages: circuit schematic.

The linear (inverter) amplifier can be modeled as a single-pole operational transconductance amplifier (OTA) that has the small-signal schematic[1] of Fig. A-4, where the values of the circuit parameters are given by:

$$\begin{aligned} G_m &= g_{m1} + g_{m2} \\ &= \frac{I_d}{(V_{gs} - V_t)} + \frac{I_d}{(V_{gs} - V_t)} \\ &= 2g_{m1} = 2g_{m2} \equiv 2g_m \end{aligned} \qquad \text{(A - 14)}$$

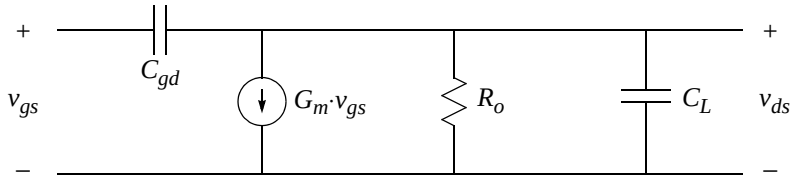


Figure A-4 Inverter gain-stage: small-signal schematic - single-pole OTA.

1. The feed-forward zero, created by C_{gd}, is assumed to occur much higher than the unity-gain frequency.

$$R_o = r_{o1} \parallel r_{o2} = \frac{\frac{V_{A1}}{I_d} \cdot \frac{V_{A2}}{I_d}}{\frac{V_{A1}}{I_d} + \frac{V_{A2}}{I_d}} = \frac{1}{I_d} \cdot \left(\frac{V_{A1} \cdot V_{A2}}{V_{A1} + V_{A2}} \right) \quad \text{(A - 15)}$$

where V_A is the early voltage and:

$$C_{gd} = c_{gd1} + c_{gd2} \quad \text{(A - 16)}$$

The load capacitance[1], C_L, is the sum of the inter-stage metal-line capacitance, C_m, the output capacitance of the current (n) stage and the input capacitance of the following (n+1) scaled-up stage (including its Miller capacitance):

$$C_L = C_m + C_o[n] + C_i[n+1] \quad \text{(A - 17)}$$

$$\begin{aligned} C_o[n] &= c_{db1} + c_{db2} + c_{db3} + c_{db4} \\ &= C_{db} + c_{db3} + c_{db4} \\ &= C_{db} + C_{sb} \end{aligned} \quad \text{(A - 18)}$$

$$\begin{aligned} C_i[n+1] &= \alpha \cdot [c_{gs1} + c_{gs2} + c_{sb3} + c_{sb4} + (1 + a_o)(c_{gd1} + c_{gd2})] \\ &= \alpha \cdot [C_{gs} + C_{sb} + (1 + a_o)C_{gd}] \end{aligned} \quad \text{(A - 19)}$$

The low-frequency gain, a_o, is:

1. For simplicity, the overlap capacitances of the "off" switches are ignored.

$$\begin{aligned} a_o &= G_m \cdot R_o = 2g_m \cdot \frac{1}{I_d} \cdot \left(\frac{V_{A1} \cdot V_{A2}}{V_{A1} + V_{A2}} \right) \\ &= 2 \cdot \left(\frac{I_d}{V_{gs} - V_t} \right) \cdot \frac{1}{I_d} \cdot \left(\frac{V_{A1} \cdot V_{A2}}{V_{A1} + V_{A2}} \right) \\ &= \left(\frac{2}{V_{gs} - V_t} \right) \cdot \left(\frac{V_{A1} \cdot V_{A2}}{V_{A1} + V_{A2}} \right) \end{aligned} \quad \text{(A - 20)}$$

which is independent of bias current and scaling factor, i.e., all stages have the same gain.

For this single-pole amplifier, its speed is related to its unity-gain frequency [1] which is given by:

$$f_u = \frac{G_m}{2\pi(C_L + C_{gd})} \quad \text{(A - 21)}$$

or alternatively by its time-constant, τ_u:

$$\begin{aligned} \tau_u &= \frac{1}{2\pi f_u} = \frac{C_L + C_{gd}}{G_m} \\ &= \frac{C_m + C_{db} + C_{sb} + \alpha \cdot [C_{gs} + C_{sb} + (1 + a_o)C_{gd}] + C_{gd}}{2g_m} \\ &= \frac{C_m}{2g_m} + \frac{C_{db} + C_{sb} + C_{gd} + \alpha \cdot (C_{sb} + a_o C_{gd})}{2g_m} + \frac{\alpha(C_{gs} + C_{gd})}{2g_m} \end{aligned} \quad \text{(A - 22)}$$

The transconductance, g_m, is proportional to the width, W, of the transistors (which also true for long-channel assuming the length, L, is constant (L_{min}) since (V_{gs}-V_t) is constant for small-signal analysis) and all the capacitors except the metal capacitance, C_m, also

scale with *W*, so the last two terms are independent of the bias current and power. The last term can be identified as the cutoff frequency of the NMOS and PMOS capacitors scaled by half α. The middle term can be associated with the intrinsic delay, τ_I, of the inverter for a given scaling factor, α. Using these quantities, (A - 22) can be rewritten as:

$$\begin{aligned}\tau_u &= \frac{C_m}{2g_m} + \tau_I + \frac{\alpha}{2}\left(\frac{1}{2\pi f_{TN}} + \frac{1}{2\pi f_{TP}}\right) \\ &= \frac{C_m(V_{gs} - V_t)}{2I_d} + \tau_I + \frac{\alpha}{2}(\tau_{TN} + \tau_{TP})\end{aligned} \tag{A - 23}$$

Assuming that the auto-zero cycle is half the sampling period, the average power dissipated by the comparator, P_C is half the power dissipated during the auto-zero cycle, so:

$$P_C = \frac{V_{dd} \cdot I_d}{2} \tag{A - 24}$$

and:

$$\begin{aligned}\tau_u &= \frac{C_m(V_{gs} - V_t)V_{dd}}{4P_c} + \tau_I + \frac{\alpha}{2}(\tau_{TN} + \tau_{TP}) \\ &= \frac{C_m(V_{gs} - V_t)V_{dd}}{4P_C} + \tau_{min}\end{aligned} \tag{A - 25}$$

From (A - 25), it is clear that increasing the power can only reduce the first term of the inverter's unity-gain time-constant and will reach a limit of τ_{min} as power approaches infinity. Since τ_u will be proportional to the sampling period, the sampling rate is found to be:

$$f_S = \frac{1}{T_S} = \frac{1}{K_{C1}\tau_u} = \frac{1}{K_{C1}\left(\frac{C_m(V_{gs}-V_t)V_{dd}}{4P_C} + \tau_{min}\right)} \quad \text{(A - 26)}$$

$$P_C = \frac{K_{C2}C_m(V_{gs}-V_t)V_{dd}\cdot f_S}{1-K_{C1}f_S\tau_{min}} \quad \text{(A - 27)}$$

for constants K_{C1} and K_{C2}. Thus the *power is **not** directly proportional to speed* but goes to infinity when the sampling rate reaches its maximum value, $f_{S(max)}$, or alternatively, when the sampling period is at a minimum, i.e.:

$$f_S = f_{S(max)} \equiv \frac{1}{K_{C1}\tau_{min}} \quad \text{(A - 28)}$$

$$T_S = T_{S(min)} = K_{C1}\tau_{min} \quad \text{(A - 29)}$$

A useful and more intuitive way of viewing the above result is obtained by comparing equations (A - 22) and (A - 25). The first term of both equations describe the time component associated with the extrinsic capacitance (C_m) which is the only time term that can be reduced by increasing the power (P_C), i.e., the g_m. The other terms of (A - 22) are intrinsic capacitance terms that are independent of power variations because a larger transistor width (W) is needed to increase the power, which also increases the intrinsic capacitance proportionately. A *3dB breakpoint* can be identified where below this point the sampling rate is proportional to power (first term of (A - 22) dominates), and where above this point, the sampling rate is relatively constant (last two terms dominate). This point occurs when the first term equals τ_{min}, or equivalently, when the intrinsic and extrinsic

capacitance are equal.

In other words, the comparator's speed will benefit from a power increase (via increasing *W*) while the intrinsic capacitances are smaller than the external parasitic capacitances, but beyond that break-even point speed improvements reach a diminishing return. Thus the power at this point is optimum, P_{opt}, from a speed-power point of view.

A.4 CMOS Preamp/Master Latch Power for Constant Current Density

From (2 - 16) in Chapter 2, the power for the first stage of the CMOS comparator of Fig. 2-6, is:

$$P_D = K_{D1} \cdot C_{p1} \cdot (V_{gs} - V_t) \cdot V_{dd} \cdot f_S \qquad \text{(A - 30)}$$

But part of C_{p1} scales with transistor width, *W*, so let:

$$C_{p1} = C_{p1f} + C_{p1s} \qquad \text{(A - 31)}$$

where C_{p1f} is the fixed portion and:

$$C_{p1s} = W \cdot K_{D2} \qquad \text{(A - 32)}$$

where K_{D2} is the constant of proportionality (capacitance density) for C_{p1s}. C_{p1s} is composed of C_{gd} and C_{db} of transistors *M1 - M4* (assumed to be the same size) so:

$$K_{D2} = \frac{4(C_{gd} + C_{db})}{W} \qquad \text{(A - 33)}$$

For a constant current density value of K_{D3}, the width can be written as a function of power:

$$W = \frac{I_d}{K_{D3}} = \frac{I_o}{2K_{D3}} = \frac{P_D}{K_{D4}V_{dd}} \qquad \text{(A - 34)}$$

Combining (A - 32) and (A - 34) gives:

$$C_{p1s} = \frac{K_{D2}P_D}{K_{D4}V_{dd}} \qquad \text{(A - 35)}$$

Then combining this with (A - 30) and (A - 31) yields:

$$P_D = K_{D1} \cdot \left(C_{p1f} + \frac{K_{D2}P_D}{K_{D4}V_{dd}} \right) \cdot (V_{gs} - V_t) \cdot V_{dd} \cdot f_S \qquad \text{(A - 36)}$$

$$P_D = \frac{K_{D1} \cdot C_{p1f} \cdot (V_{gs} - V_t) \cdot V_{dd} \cdot f_S}{1 - \left(K_{D1} \cdot \frac{K_{D2}}{K_{D4}} \cdot (V_{gs} - V_t) \cdot f_S \right)} \qquad \text{(A - 37)}$$

The parenthetical portion of the denominator can be evaluated by expanding the constants K_{D2} and K_{D4}:

$$K_{D1} \cdot \frac{K_{D2}}{K_{D4}} \cdot (V_{gs}-V_t) \cdot f_S = K_{D1} \cdot \frac{4(C_{gd}+C_{db})}{W} W \cdot \frac{(V_{gs}-V_t)}{2I_d} \cdot f_S$$

$$= K_{D1} \cdot 4(C_{gd}+C_{db}) \cdot \frac{1}{g_m} \cdot f_S \qquad \text{(A - 38)}$$

$$= K_{D1} \cdot \frac{4(C_{gd}+C_{db})}{(C_{gs}+C_{gd})} \cdot \frac{f_S}{f_T} = \frac{f_S}{f_{S(max)}}$$

where $f_{S(max)}$ is:

$$f_{S(max)} = \frac{f_T}{4K_{D1}\left(\frac{C_{gd}+C_{db}}{C_{gs}+C_{gd}}\right)} = \frac{f_T}{4K_{D1}\rho} \qquad \text{(A - 39)}$$

Then (A - 37) can be written as:

$$P_D = \frac{K_{D1} \cdot C_{p1f} \cdot (V_{gs}-V_t) \cdot V_{dd} \cdot f_S}{1 - f_S / f_{S(max)}} \qquad \text{(A - 40)}$$

Short-channel analysis would yield the same result except for a factor of 2 in the value of K_{D1}.

A.5 Power-speed derivation for Digital CMOS Circuits

The well-known dynamic power equation for digital CMOS circuits is:

$$P_H = C_{eff} V_{dd}^2 \cdot f_S \qquad \text{(A - 41)}$$

where C_{eff} is the total *effective* capacitance that is being switched at f_S. As before, C_{eff} can

be partitioned into a part that scales with transistor *W* and a part that is fixed so:

$$\begin{aligned} C_{eff} &= C_{feff} + C_{seff} \\ &= C_{feff} + W \cdot K_{H1} \end{aligned} \quad \text{(A - 42)}$$

where it is assumed that *W* is a generic scaling factor for all the digital logic. Since $(V_{gs}\text{-}V_t)$ is fixed to $(V_{dd}–V_t)$, the current density is fixed[1] so for some constant, K_{H2}:

$$\frac{I_d}{W} = K_{H2} \quad \text{(A - 43)}$$

thus the current is proportional to the size of the transistors. From these two facts, the effective capacitance is a function of the charging and discharging current so what remains is to relate the current to the switching speed of the digital gates. For CMOS digital gates, the propagation delay, t_{pd}, is approximately:

$$t_{pd} = \frac{C_o \cdot (V_{dd}/2)}{I_d} \quad \text{(A - 44)}$$

where C_o is the capacitance load at the output of the gate. The maximum switching frequency, $f_{sw(max)}$, is the inverse of twice the propagation delay , so:

$$f_{sw(max)} = \frac{I_d}{C_o \cdot V_{dd}} \quad \text{(A - 45)}$$

Combining (A - 42), (A - 43), and (A - 45), and equating C_o to C_{eff} and $f_{sw(max)}$ to f_S via a constant (K_{H3}), C_{eff} as a function of f_S can be found:

1. See section A.1 “Velocity Saturation Equations for CMOS Transistors” above.

$$f_S = \frac{K_{H3} I_d}{(C_{feff} + W \cdot K_{H1}) \cdot V_{dd}} = \frac{K_{H3} \cdot W \cdot K_{H2}}{(C_{feff} + W \cdot K_{H1}) \cdot V_{dd}} \quad \text{(A - 46)}$$

Solving for *W*:

$$W = \frac{C_{feff} \cdot V_{dd} \cdot f_S}{K_{H3} \cdot K_{H2} - K_{H1} \cdot V_{dd} \cdot f_S} \quad \text{(A - 47)}$$

Using (A - 42):

$$C_{eff} = C_{feff} + \left(\frac{C_{feff} \cdot V_{dd} \cdot f_S}{K_{H3} \cdot K_{H2} - K_{H1} \cdot V_{dd} \cdot f_S} \right) \cdot K_{H1} = \frac{C_{feff} \cdot K_{H3} \cdot K_{H2}}{K_{H3} \cdot K_{H2} - K_{H1} \cdot V_{dd} \cdot f_S} = \frac{C_{feff}}{(1 - f_S / f_{S(max)})} \quad \text{(A - 48)}$$

where:

$$f_{S(max)} = \frac{K_{H3} \cdot K_{H2}}{K_{H1} \cdot V_{dd}} = \frac{K_{H3} \cdot \frac{I_d}{W}}{\frac{C_{seff}}{W} \cdot V_{dd}} = \frac{K_{H3} \cdot I_d}{C_{seff} \cdot V_{dd}} \quad \text{(A - 49)}$$

Combining (A - 41) with (A - 48) gives:

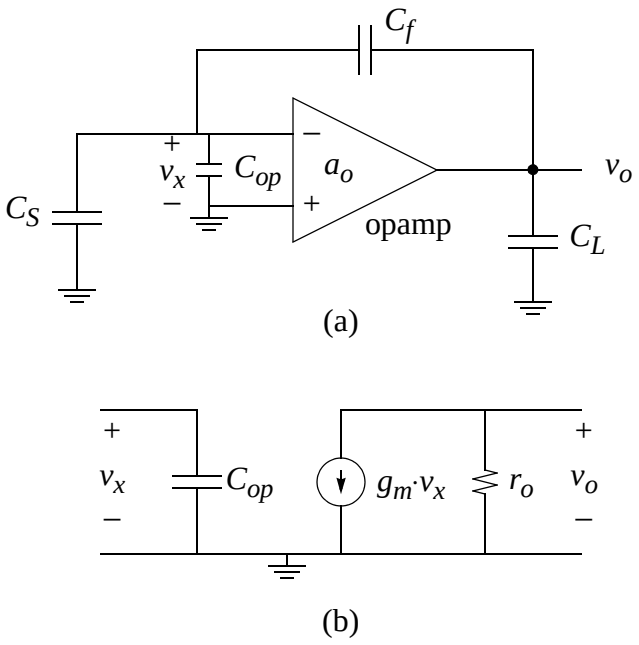


Figure A-5 Settling-time calculation circuits; (a) simplified DAC sub-gain S/H diagram, (b) small-signal model of opamp.

$$P_H = \frac{C_{feff} V_{dd}^2 \cdot f_S}{(1 - f_S / f_{S(max)})} \qquad \text{(A - 50)}$$

A.6 Pipeline Stage Power Calculation

To calculate the per-stage power of a Pipelined converter, the first assumption made is to ignore the flash ADC power (negligible relative to the opamp power) so that the stage power is that of the opamp. The opamp is modeled as a common-source amplifier with a current source load. The DAC-subtraction-gain-S/H block diagram used for the settling time derivation is shown in Fig. A-5 along with the small signal model of the opamp. The circuit of Fig. A-5(a) has a single-pole open-loop response, with a pole, p_1, equal to:

$$p_1 = \frac{1}{r_o \cdot C_o} = \frac{1}{\tau_1} = \frac{1}{r_o \cdot \left(C_L + \left(\frac{C_f(C_S + C_{op})}{C_f + C_S + C_{op}} \right) \right)} \quad \text{(A - 51)}$$

where C_o is the total capacitance at the output node, C_L represents all extrinsic capacitance at the output node (primarily the input capacitance of the next stage), τ_1 is the time-constant associated with p_1 and C_{op} is the input capacitance of the opamp. The low-frequency open-loop gain, a_o, is:

$$a_o = g_m \cdot r_o \quad \text{(A - 52)}$$

For a single-pole response, the unity-gain bandwidth, f_0, is equal to the gain-bandwidth (GBW) product so:

$$2\pi f_0 = a_o \cdot p_1 = \frac{1}{\tau_0} = \frac{g_m}{C_L + \left(\frac{C_f(C_S + C_{op})}{C_f + C_S + C_{op}} \right)} = \frac{g_m}{C_L + f(C_S + C_{op})} \quad \text{(A - 53)}$$

where f is the feedback factor [1] and is given by:

$$f = \frac{C_f}{C_f + C_S + C_{op}} \tag{A - 54}$$

Assuming that a_o is high enough for a given stage's accuracy and the low-frequency loop-gain, $(T_o = a_o \cdot f)$ is much larger than one, then the closed-loop gain, A_{cl},[1] is approximately the inverse of the feedback factor, i.e., $1/f$, and the closed-loop bandwidth, f_{cl}, is the GBW product divided by the closed-loop gain:

$$\begin{aligned} 2\pi f_{cl} &= \frac{1}{\tau_{cl}} = \frac{\text{GBW}}{A_{cl}} = \frac{2\pi f_0}{1/f} \\ &= \frac{g_m}{C_o} \cdot f = \frac{g_m \cdot f}{C_L + f(C_S + C_{op})} \end{aligned} \tag{A - 55}$$

Thus the closed-loop settling time-constant, τ_{cl}, is:

$$\tau_{cl} = \frac{C_o}{g_m} \cdot \frac{1}{f} = \frac{C_x}{g_m} \tag{A - 56}$$

where C_x can be written as:

1. Note that this closed-loop gain pertains to the **non-inverting** opamp configuration which is relevant when considering stability and settling performance. This should not be confused with the **inverting**-opamp configuration's closed-loop gain, $A_I (= -C_S/C_f)$, which is the quantity that determines the "operational" gain of the SC circuit.

$$C_x = \frac{C_o}{f} = C_S + C_{op} + \frac{1}{f} \cdot C_L$$
$$= C_S\left(1 + \frac{C_L}{C_f} + \frac{C_L}{C_S}\right) + C_{op}\left(1 + \frac{C_L}{C_f}\right) \qquad \text{(A - 57)}$$
$$= C_S\left(1 + \frac{C_L}{C_f}\left(1 + \frac{1}{|A_I|}\right)\right) + C_{op}\left(1 + \frac{C_L}{C_f}\right)$$

A_I is the closed-loop (inverting) gain of the SC integrator and given by:

$$A_I = -\frac{C_S}{C_f} \qquad \text{(A - 58)}$$

The worst-case settling time of a closed-loop SC integrator occurs when the output experiences its largest voltage excursion. Assuming a Class-A type opamp in this case, the settling event will be divided into a *slew-rate limited* (large signal) period followed by a *exponential settling* (small-signal) period [1] (see Fig. A-6). The finite bias current of

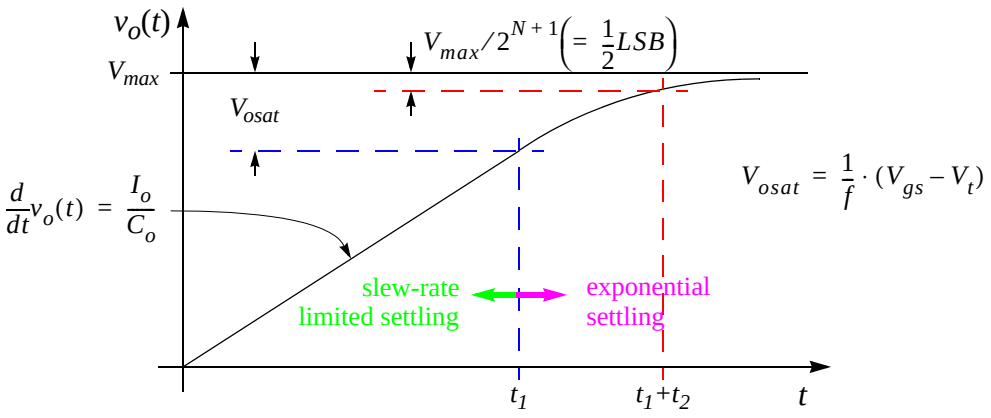


Figure A-6 Settling behavior of SC integrator.

the Class-A opamp initially causes *slew-rate limited* settling because the output cannot transition as fast as that demanded by a single-pole response (*exponential* settling). The settling behavior switches to exponential settling once the time-derivative of the single-pole response equals the slew-rate limit. For a maximum output voltage excursion of V_{max}, the point of this transition can be derived as follows. The linear single-pole response at the output is:

$$v_o(t) = V_{max} \cdot (1 - e^{-t/\tau_{cl}}) \tag{A - 59}$$

and its derivative (ramp rate) is:

$$\frac{d}{dt}v_o(t) = \frac{V_{max}}{\tau_{cl}} \cdot e^{-t/\tau_{cl}} \tag{A - 60}$$

By setting the above equation equal to the slew-rate, the voltage when exponential settling begins can be solved for:

$$\frac{d}{dt}v_o(t_x) = \frac{V_{max}}{\tau_{cl}} \cdot e^{-t_x/\tau_{cl}} = \frac{I_o}{C_o} \tag{A - 61}$$

$$t_x = \tau_{cl} \cdot \ln\left(\frac{V_{max}}{\tau_{cl}} \cdot \frac{C_o}{I_o}\right) \tag{A - 62}$$

Substituting (A - 62) back into (A - 59), the output voltage is:

$$
\begin{aligned}
v_o(t_x) &= V_{max} \cdot \left(1 - e^{-\left(\tau_{cl} \cdot \ln\left(\frac{V_{max}}{\tau_{cl}} \cdot \frac{C_o}{I_o} \right) \right) / \tau_{cl}} \right) \\
&= V_{max} \cdot \left(1 - \frac{1}{\frac{V_{max}}{\tau_{cl}} \cdot \frac{C_o}{I_o}} \right) \\
&= V_{max} - \frac{\tau_{cl} \cdot I_o}{C_o} = V_{max} - \left(\frac{C_o}{g_m} \cdot \frac{1}{f} \right) \cdot \frac{I_o}{C_o} \\
&= V_{max} - \left(\frac{(V_{gs} - V_t)}{I_o} \cdot \frac{1}{f} \right) \cdot I_o = V_{max} - A_{cl} \cdot (V_{gs} - V_t)
\end{aligned}
\qquad \text{(A - 63)}
$$

where $A_{cl} = \frac{1}{f}$ is assumed. (A - 63) shows that the output voltage swing is slew-rate limited until the voltage is within V_{dsat} ($=V_{gs}-V_t$) times the closed-loop gain of its final value. In terms of the input voltage to the opamp, $v_x(t)$ (which is smaller by the closed-loop gain), slew-rate limited settling ends when the input voltage is equal to a V_{dsat}.[1] Knowing the output voltage of (A - 63), the slew-rate limited settling period, t_1, is simply:

$$
\begin{aligned}
t_1 &= \left(V_{max} - \frac{1}{f} \cdot (V_{gs} - V_t) \right) \cdot \frac{C_o}{I_o} \\
&= (V_{max} - V_{osat}) \cdot \frac{C_o}{I_o}
\end{aligned}
\qquad \text{(A - 64)}
$$

where V_{osat} is V_{dsat} referred to the output.

1. This is true since the final value of $v_x(t)$ is zero. Note that I_o is typically the bias of a diff-pair so that it is equal to $2I_d$ for small-signal analysis (which is a somewhat gross over-simplification)

Given V_{max}, the time, t_2, for the exponential settling to within $\frac{1}{2}$ LSB for *i*-bits of resolution can be found by solving the following equation for t_2:

$$\begin{aligned} v_o(t_2) &= V_{osat} \cdot (1 - e^{-t_2/\tau_{cl}}) \\ &= V_{osat} - \frac{V_{max}}{2^{(i+1)}} \end{aligned} \tag{A - 65}$$

$$\begin{aligned} t_2 &= \tau_{cl} \cdot \left((i+1)\ln(2) - \ln\left(\frac{V_{max}}{V_{osat}}\right)\right) \\ &= \frac{C_o}{I_o} \cdot \frac{(V_{gs} - V_t)}{f} \cdot \left((i+1)\ln(2) - \ln\left(\frac{V_{max}}{V_{osat}}\right)\right) \end{aligned} \tag{A - 66}$$

In a worst-case design, the total settling time (the sum of t_1 and t_2) is equal to half the sampling period so:

$$\begin{aligned} t_1 + t_2 &= \frac{C_o}{I_o} \cdot V_{osat}\left(\frac{V_{max}}{V_{osat}} - 1 + (i+1)\ln(2) - \ln\left(\frac{V_{max}}{V_{osat}}\right)\right) \\ &= \frac{C_o V_{dd}}{P_i} \cdot V_{osat}\left(\frac{V_{max}}{V_{osat}} - 1 + (i+1)\ln(2) - \ln\left(\frac{V_{max}}{V_{osat}}\right)\right) \\ &= \frac{T_S}{2} = \frac{1}{2f_S} \end{aligned} \tag{A - 67}$$

where P_i is the power of a pipelined stage that needs to meet an accuracy of *i*-bits and V_{osat} is:

$$V_{osat} = \frac{1}{f} \cdot (V_{gs} - V_t) \cong A_{cl} \cdot (V_{gs} - V_t) \qquad \text{(A - 68)}$$

(A - 67) can be rearranged to give:

$$\begin{aligned} P_i &= 2C_o V_{dd} V_{osat} f_S \cdot \left((i+1)\ln(2) + \frac{V_{max}}{V_{osat}} - 1 - \ln\left(\frac{V_{max}}{V_{osat}}\right)\right) \\ &= 2C_x V_{dd} (V_{gs} - V_t) f_S \cdot \left((i+1)\ln(2) + \frac{V_{max}}{V_{osat}} - 1 - \ln\left(\frac{V_{max}}{V_{osat}}\right)\right) \\ &= P_x \cdot [(i+1)\ln(2) + \alpha] \end{aligned} \qquad \text{(A - 69)}$$

where:

$$\begin{aligned} C_x &= \left(C_S \left(1 + \frac{C_L}{C_f}\left(1 + \frac{1}{|A_I|}\right)\right) + C_{op}\left(1 + \frac{C_L}{C_f}\right)\right) \\ P_x &= 2C_x V_{dd} (V_{gs} - V_t) f_S \\ \alpha &= \frac{V_{max}}{V_{osat}} - 1 - \ln\left(\frac{V_{max}}{V_{osat}}\right) \end{aligned} \qquad \text{(A - 70)}$$

α ranges typically from 0 (when V_{osat} equals V_{max}, i.e., no slew rate limiting) to 3 (for $V_{dsat} = 175mV$, $f = 0.5$, $V_{max} = 2V$). The *kT/C* noise influence on the power is incorporated in C_x via the value of C_S.

A.7 Derivation of P_N (Ignoring *kT/C* Noise)

The total power[1] (P_N) of a 1*bit*-per-stage *power-optimized* pipelined converter of

N-bits can be obtained from (A - 69) as follows:

$$P_N = \sum_{1}^{N} P_i = \sum_{1}^{N} (P_x \cdot [(i+1)\ln(2) + \alpha]) \tag{A - 71}$$

If *kT/C* considerations are ignored, then P_x is constant and can be factored out of the summation, yielding:

$$\begin{aligned} P_N &= P_x \sum_{1}^{N} [(i+1)\ln(2) + \alpha] \\ &= P_x \left[\ln(2) \sum_{1}^{N} i + (\ln(2) + \alpha) \sum_{1}^{N} 1 \right] \\ &= P_x \left[\ln(2) \left(\frac{N(N+1)}{2} \right) + (\ln(2) + \alpha) N \right] \\ &= P_x \cdot \frac{\ln(2)}{2} \left[N^2 + N \left(3 + \frac{2\alpha}{\ln(2)} \right) \right] \end{aligned} \tag{A - 72}$$

A.8 References

[1]. P. R. Gray, and R. G. Meyer, *Analysis and Design of Analog Integrated Circuits*, 3rd Ed., New York, N.Y.: John Wiley and Sons, Inc., 1993.

[2]. D. W. Cline, “Noise, Speed, and Power Trade-offs in Pipelined Analog to Digital Converters,” *Memo UCB/ERL M*95/94, Univ. Calif., Berkeley, November 1995.

[3]. F. G. Stremler, *Introduction to Communication Systems*, 3rd ed., Reading, Mass. USA: Addison-Wesley Pub. Co., 1990.

1. Considering only opamp power.

Appendix B

Addendum to Chapter 4

B.1 Error cancellation of "zeroing" sequence

In Section 4.2, *top-plate passive* integration uses a zeroing sequence to eliminate error charge on the top-plate of the sampling capacitors, C_S. This sequence momentarily closes a switch (*M4*) after capturing a sample of the input signal and prior to integrating it. It relies on the fact that the net charge on the node connecting C_S and C_{p1} is preserved after the sampling operation (see Fig. B-1).

Fig. B-1 is a reprint of Fig. 4-13 and will help to explain the process. The charges on

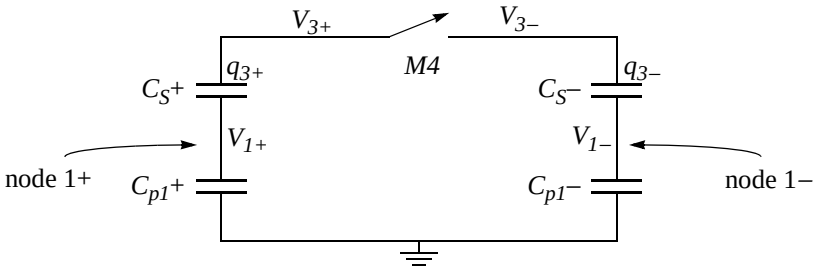


Figure B-1 Cancelling charge errors prior to passive integration.

the top-plates of $C_S\pm$ are indeterministic and random as are the corresponding voltages (V_{3+} and V_{3-}). However the net charges, q_{1+} and q_{1-}, at nodes 1+ and 1– are known (see equation (4 - 13) and accompanying section):

$$q_{1+} = -q_S + Q_1 \tag{B - 1}$$

$$q_{1-} = q_S + Q_1 \tag{B - 2}$$

where Q_1 is the common-mode charge error[1] due to charge injection and clock feedthrough when the sampling switch is turned off.

Using the above facts, the equation for the charge on the top-plate of C_S+ (and C_S–), q_{3+} (and q_{3-}), as a function of V_{3+} (and V_{3-}) can be derived as follows:[2]

$$\begin{aligned} V_{3+} &= \frac{q_{3+}}{C_S} + V_{1+} \\ &= \frac{q_{3+}}{C_S} + \frac{(-q_S + Q_1) - (-q_{3+})}{C_{p1}} \\ &= \frac{q_{3+}}{C_S} + \frac{-q_S + Q_1 + q_{3+}}{C_{p1}} \end{aligned} \tag{B - 3}$$

Solving for q_{3+} from (B - 3):

$$q_{3+} = q_S \cdot \left(\frac{C_S}{C_S + C_{p1}}\right) - Q_1 \cdot \left(\frac{C_S}{C_S + C_{p1}}\right) + V_{3+} \cdot \left(\frac{C_S \cdot C_{p1}}{C_S + C_{p1}}\right) \tag{B - 4}$$

1. Assuming $Q_1+ = Q_1- = Q_1$.
2. Assuming $C_S+ = C_S- = C_S$ and $C_{p1}+ = C_{p1}- = C_{p1}$.

Similarly, q_{3-} is:

$$q_{3-} = -q_S \cdot \left(\frac{C_S}{C_S + C_{p1}}\right) - Q_1 \cdot \left(\frac{C_S}{C_S + C_{p1}}\right) + V_{3-} \cdot \left(\frac{C_S \cdot C_{p1}}{C_S + C_{p1}}\right) \quad \text{(B - 5)}$$

From (B - 4) and (B - 5), only the last term adds a variable and unknown charge to the top-plates of C_S+ and C_S– and is dependent upon the value of V_{3+} or V_{3-}, respectively. Thus by forcing V_{3+} to equal V_{3-}, the last terms cancel and the net differential charge, q_{df}, is:

$$q_{df} = q_{3+} - q_{3-} = 2 \cdot q_S \cdot \left(\frac{C_S}{C_S + C_{p1}}\right) \quad \text{(B - 6)}$$

B.2 Gain attenuation of integration step

After the zeroing sequence removes the error components from the top-plate of C_S, the final step is to turn on *M1* and *M2* (time t_4 in Fig. B-2), thereby integrating the signal charge with the other samples. As stated in Chapter 4, due to three parasitic capacitances, the signal charge is further attenuated. The three parasitic capacitors are the top-plate summing node parasitic (C_{p3}), the per-cell output-bus capacitance (C_{po}) and the gate-oxide capacitance of *M2* (C_{ox2}). To show this, first consider the differential output voltage, v_{odf}, of a passive integrator resulting from integration of N input samples:[1]

1. In a passive approach there is an automatic normalization by the number of samples integrated, N.

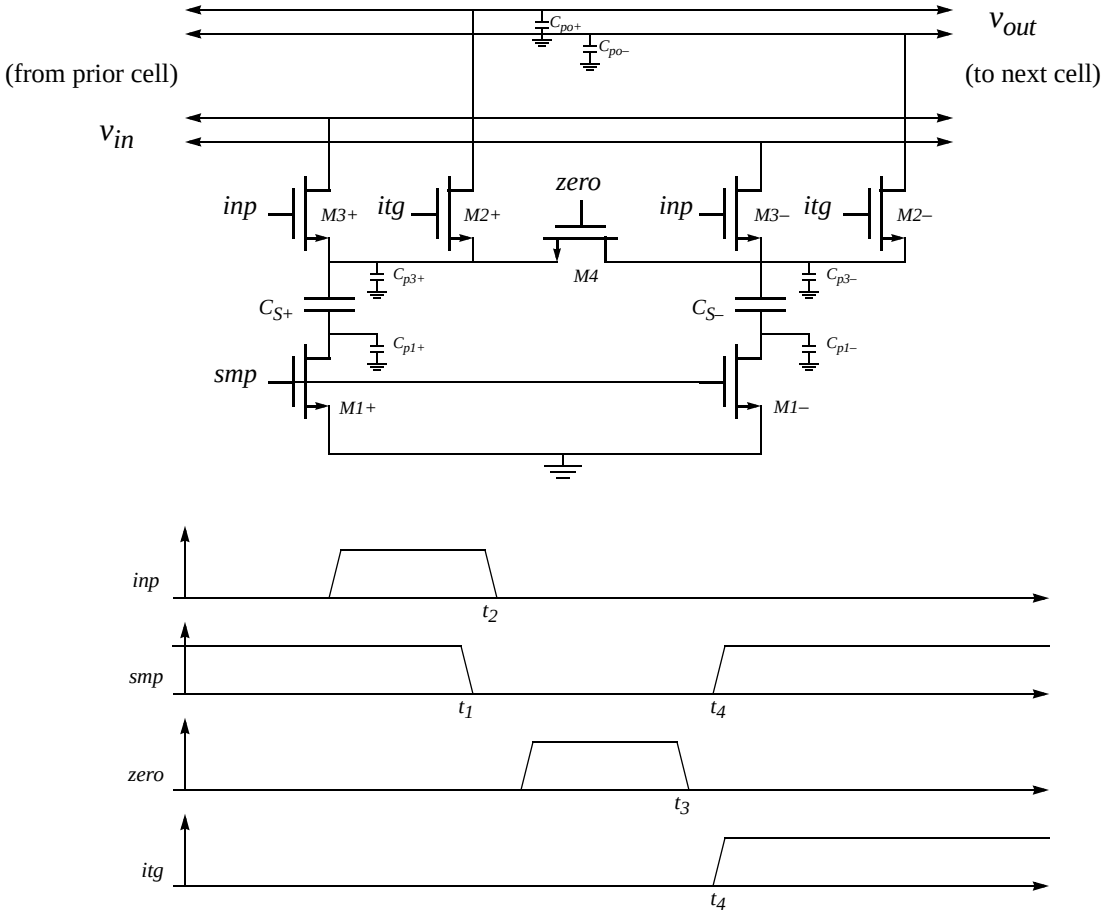


Figure B-2 Signal charge is integrated when *M1* and *M2* turn on at time t_4.

$$v_{odf} = \frac{1}{N} \cdot \sum_{1}^{N} (v_{idf}(i) \cdot A_v) = \frac{A_v}{N} \cdot \sum_{1}^{N} v_{idf}(i) \qquad \text{(B - 7)}$$

where $v_{idf}(i)$ is the differential input voltage samples and A_v is the per-cell voltage gain. For an ideal passive integrator, A_v equals one, and is less than unity for any actual implementation. (B - 7) can be compared to the derived result to extract the value for A_v.

Let q_S' be defined as:

$$q_S' = q_S \cdot \left(\frac{C_S}{C_S + C_{p1}} \right) \tag{B - 8}$$

Then from (B - 4) and (B - 5), the differential charges, q_{3+}(i) and q_{3-}(i), on the top-plates of C_S+ and C_S–, respectively, for the *i*-th sample can be written as:

$$\begin{aligned} q_{3+}(i) &= q_S'(i) + Q_\varepsilon(i) \\ q_{3-}(i) &= -q_S'(i) + Q_\varepsilon(i) \end{aligned} \tag{B - 9}$$

where Q_ε is the sum of the last two terms in (B - 4) and (B - 5) (with V_{3+} equal to V_{3-} from the zeroing operation). Thus the total differential integrated charges, Q+ and Q–, from N samples are:

$$\begin{aligned} Q+ &= \sum_{1}^{N} (q_S'(i) + Q_\varepsilon(i)) = N \cdot (\overline{q_S'} + \overline{Q_\varepsilon}) \\ Q- &= \sum_{1}^{N} (-q_S'(i) + Q_\varepsilon(i)) = N \cdot (-\overline{q_S'} + \overline{Q_\varepsilon}) \end{aligned} \tag{B - 10}$$

where $\overline{q_S'}$ and $\overline{Q_\varepsilon}$ are the average values of their respective N quantities.

When switch *M2* is on, the signal charge is changed by the amount of charge that is stored in the transistor's channel, $-q_{ox2}$.[1] The amount of charge in the channel is bias

1. For NMOS switches, the channel charge is negative so an equal and opposite charge is added to the signal charge. Hence the channel charge is $-q_{ox2}$.

dependent and hence dependent on the output voltage. Since the total amount of the channel charge (from the N switches) is important only at the end of the integration, it only depends on the final output voltage, V_{out}. Thus $V_{out}+$ is:

$$V_{out}+ = \frac{(Q+) + (N \cdot q_{ox2}+)}{N \cdot (C_S + C_{p3} + C_{po})} \tag{B - 11}$$

where q_{ox2} is the charge due to one ($M2$) switch:

$$\begin{aligned} q_{ox2}+ &= W \cdot L \cdot \frac{\varepsilon_{ox}}{t_{ox}} \cdot (V_{dd} - V_{out}+ - V_t) \\ &= C_{ox2} \cdot (V_{dd} - V_{out}+ - V_t) \end{aligned} \tag{B - 12}$$

Substituting (B - 12) in (B - 11) yields:

$$V_{out}+ = \frac{(Q+) + N \cdot C_{ox2} \cdot (V_{dd} - V_t) - N \cdot C_{ox2} \cdot (V_{out}+)}{N \cdot (C_S + C_{p3} + C_{po})} \tag{B - 13}$$

$$\begin{aligned} V_{out}+ &= \frac{(Q+) + N \cdot C_{ox2} \cdot (V_{dd} - V_t)}{N \cdot (C_S + C_{p3} + C_{po} + C_{ox2})} \\ &= \frac{N \cdot (\overline{q_S'} + \overline{Q_\varepsilon}) + N \cdot C_{ox2} \cdot (V_{dd} - V_t)}{N \cdot (C_S + C_{p3} + C_{po} + C_{ox2})} \\ &= \frac{\overline{q_S'} + \overline{Q_\varepsilon} + C_{ox2} \cdot (V_{dd} - V_t)}{(C_S + C_{p3} + C_{po} + C_{ox2})} \\ &= \frac{\overline{q_S'}}{C_T} + \frac{\overline{Q_\varepsilon}}{C_T} + \frac{C_{ox2} \cdot (V_{dd} - V_t)}{C_T} \end{aligned} \tag{B - 14}$$

Similarly, $V_{out}-$ is:

$$V_{out}- = -\frac{\overline{q_S'}}{C_T} + \frac{\overline{Q_\varepsilon}}{C_T} + \frac{C_{ox2}\cdot(V_{dd}-V_t)}{C_T} \qquad \text{(B - 15)}$$

And thus the final differential output voltage, V_{odf}, is:

$$\begin{aligned} v_{odf} &= (V_{out}+)-(V_{out}-) = \frac{2\cdot q_S'}{C_T} = \frac{2\cdot q_S}{C_T}\cdot\left(\frac{C_S}{C_S+C_{p1}}\right) \\ &= \left\{\frac{1}{N}\cdot\sum_1^N (v_{idf}(i)\cdot C_S)\right\}\cdot\frac{1}{C_T}\cdot\left(\frac{C_S}{C_S+C_{p1}}\right) \qquad \text{(B - 16)} \\ &= \frac{1}{N}\cdot\sum_1^N v_{idf}(i)\cdot\left(\frac{C_S}{C_S+C_{p3}+C_{po}+C_{ox2}}\right)\cdot\left(\frac{C_S}{C_S+C_{p1}}\right) \end{aligned}$$

By comparing (B - 16) to (B - 7), the voltage gain is:

$$A_v = \left(\frac{C_S}{C_S+C_{p1}}\right)\cdot\left(\frac{C_S}{C_S+C_{p3}+C_{po}+C_{ox2}}\right) \qquad \text{(B - 17)}$$

as desired.

B.3 Error Derivation due to Junction Capacitance Non-linearity

In Section 4.3, the non-linearity of the parasitic junction capacitance, C_{p1}, for the top-plate passive SC integrator contributes a differential error component to the gain of a cell depending on the magnitude of the sampled signal. In this section that fractional error (as given in (4 - 28)) is calculated.

From (B - 6), it appears that the non-linearity of C_{p1} only affects the differential gain in a direct manner; however in the derivation of (B - 6), the last terms in (B - 4) and (B - 5) do not cancel if C_{p1} is non-constant, producing additional error terms. To re-derive (B - 6) including the non-linearity of C_{p1}, let C_{p1}^+ and C_{p1}^- be the values of C_{p1} for q_{3+} and q_{3-}, respectively. C_{p1}^+ and C_{p1}^- are the values of the respective C_{p1} capacitors when *M4* is closed and $V_{3+} = V_{3-} = V_3$. These charges, q_{3+} and q_{3-}, on C_S^+ and C_S^- are *captured* once *M4* is opened so subsequent variations of C_{p1}^+ and C_{p1}^- do not affect the result. Let the corresponding voltages across C_{p1}^+ and C_{p1}^- be $V_{cm} \pm \Delta V$, where ΔV is roughly the sampled input voltage. Then the differential charge on $C_S^{\pm}$ is:

$$\begin{aligned} q_{df} &= q_{3+} - q_{3-} \\ &= q_S \cdot (A_v^+ + A_v^-) \\ &\quad - Q_1 \cdot (A_v^+ - A_v^-) \\ &\quad + V_3 \cdot [(C_{p1}^+ \cdot A_v^+) - (C_{p1}^- \cdot A_v^-)] \end{aligned} \tag{B - 18}$$

where:

$$\begin{aligned} A_v^+ &= \frac{C_S}{C_S + C_{p1}^+} \\ A_v^- &= \frac{C_S}{C_S + C_{p1}^-} \end{aligned} \tag{B - 19}$$

Referring to (B - 6), the fractional error of the differential charge, E_{dq}, can be calculated as follows:

$$q_{df} = 2 \cdot q_S \cdot A_{vo} \cdot (1 + E_{dq}) \tag{B - 20}$$

$$E_{dq} = \frac{q_{df}}{2 \cdot q_S \cdot A_{vo}} - 1 \tag{B - 21}$$

where:

$$A_{vo} = \frac{C_S}{C_S + C_{p1o}} \tag{B - 22}$$

$$C_{p1o} = C_{j0} \cdot \frac{1}{\left(1 + \frac{V_{cm}}{\phi}\right)^m} \tag{B - 23}$$

therefore from (B - 18) and (B - 21):

$$\begin{aligned} E_{dq} &= \frac{1}{2} \cdot \left(\frac{A_v^+ + A_v^-}{A_{vo}}\right) - 1 - \frac{Q_1}{2 \cdot q_S} \cdot \left(\frac{A_v^+ - A_v^-}{A_{vo}}\right) \\ &\quad + \frac{V_3}{2 \cdot q_S} \cdot \left[\frac{(C_{p1}^+ \cdot A_v^+) - (C_{p1}^- \cdot A_v^-)}{A_{vo}}\right] \end{aligned} \tag{B - 24}$$

$$\begin{aligned} &= \frac{1}{2} \cdot [E_a + E_b] - 1 - \frac{Q_1}{2 \cdot q_S} \cdot [E_a - E_b] \\ &\quad + \frac{V_3 \cdot C_{p1o}}{2 \cdot q_S} \cdot \left[\frac{C_{p1}^+}{C_{p1o}} \cdot E_a - \frac{C_{p1}^-}{C_{p1o}} \cdot E_b\right] \end{aligned} \tag{B - 25}$$

where:

$$E_a = \frac{A_v^+}{A_{vo}} = \frac{C_S + C_{p1o}}{C_S + C_{p1}^+} = \frac{\left(1 + \frac{C_{p1o}}{C_S}\right)}{\left(1 + \frac{C_{p1}^+}{C_S}\right)} \quad \text{(B - 26)}$$

$$E_b = \frac{\left(1 + \frac{C_{p1o}}{C_S}\right)}{\left(1 + \frac{C_{p1}^-}{C_S}\right)} \quad \text{(B - 27)}$$

$$C_{p1}^+ = C_{j0} \cdot \frac{1}{\left(1 + \frac{V_{cm} + \Delta V}{\phi}\right)^m} \quad \text{(B - 28)}$$

$$C_{p1}^- = C_{j0} \cdot \frac{1}{\left(1 + \frac{V_{cm} - \Delta V}{\phi}\right)^m} \quad \text{(B - 29)}$$

An example will help quantify the relative magnitudes of the error components.

B.3-1 Example

This example is taken from the prototype chip which was fabricated in a 1.2μm CMOS double-poly process.

$C_S = 250fF$, $v_{in}^+ = +0.5V$ and $v_{in}^- = -0.5V$ so:

$\Delta V = 0.5V$ and $q_S = C_S \cdot v_{in}^+ = 125fC$

For V_{cm} = 1.5*V*, ϕ = 0.8*V*, *m* = 0.4 and a *M1* switch size of $\frac{W}{L} = \frac{3.6um}{1.2um}$, yields:

$$Q_1 = \frac{1}{2} \cdot W \cdot L \cdot C_{ox} \cdot (V_{gs} - V_t) = 9fC$$

$$C_{j0} = 11.23fF \qquad C_{p1o} = 7.36fF$$
$$C_{p1}+ = 6.81fF \qquad C_{p1}- = 8.12fF$$

For V_3 = 1.5V:

$$V_3 \cdot C_{p1o} = 11fC$$

From the above we find:

$$E_a = 1.0021 \qquad E_b = 0.9971$$

and:

$$E_{dq} = -0.00040 - 0.00018 - 0.00760 = -0.00818 = -0.82\%$$

which permits about 42*dB* of dynamic range. The biggest error component is due to the last term (V_3 term) since the series combination of C_S and C_{p1} is dominated by C_{p1} and its non-linearity during the zeroing operation (see Fig. B-1).

Appendix C

Addendum to Chapter 5

C.1 Power Estimates of Timing-Signal Generation and Distribution

In this section the calculations for two approaches to the timing-signal generation and distribution will be presented. One approach is the generate-and-distribute technique which generates the four timing signal at one end of the correlator chain and uses four SR chains to distribute the signals to each of the sampling cells of the correlator. The other approach is to propagate a single pulse (*start-pulse*) down a SR chain, use a multi-tap technique to create a clock sequence for each generation logic block and locally generate the timing signals. Since the second approach minimizes the high-frequency switching capacitance, i.e., the 64*MHz* clock, it results in a power savings over the first approach.

Simulation results shown in Table C-1 can be used to quantify this savings.[1] The generate-and-distribute scheme requires four 64-stage SR chains. The locally-generate

1. Metal line capacitance was not included in the simulation nor was the capacitive loading of the differential sampling cells. Both would have the same effect on the two approaches and not affect the relative results.

TABLE C-1 Power Breakdown

Description	Power
Clock power (64-stage SR)	1.899 *mW*
Switching power (64-stage SR)	.660 *mW*
subtotal	2.559 *mW*
Generation logic switching power(1)	.884 *mW*

(1). This also included the additional power due to loading of this block on the multi-tapped SR chain.

approach requires only one 64-stage SR chain but has additional switching power from the per-cell generation logic. For the entire chip (12 correlators, i.e., 6 correlators per bank), eight sets of timing-chains were needed (4 per bank; one each for the In-phase Data correlator, the Quadrature-phase Data correlator, the I-phase Early and Late correlators and the Q-phase Early and Late correlators).

Table C-2 tabulates the power totals.

TABLE C-2 Power Comparison

Approach	Single 64-stage SR Chain	Four 64-stage SR Chains	Additional Generate Logic	Eight Sets of Timing Chains
Generate-and-distribute	2.559 *mW*	10.236 *mW*	–	81.888 *mW*
Locally generate	2.559 *mW*	–	3.443 *mW*	27.544 *mW*

C.2 Average Frequency of PN Generator Outputs

Fig. C-1 shows example waveforms of the output from a shift-register (positive edge-triggered D-flip flop) that is used in a PN generator along with the input clock waveform. The fastest output rate possible, out_{fast}, is half the frequency of the clock. Since any

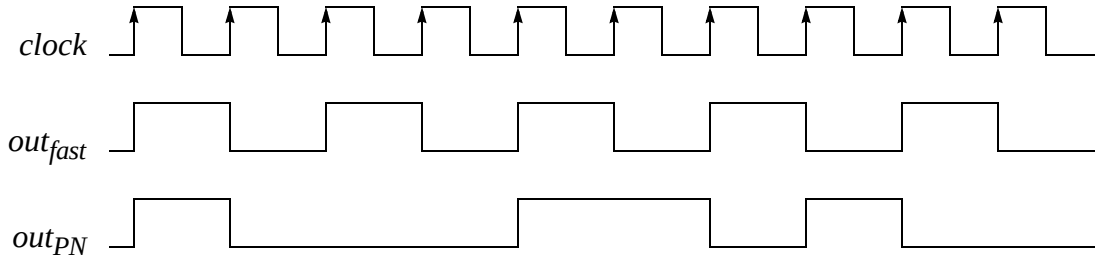


Figure C-1 Clock and output waveforms of a shift-register used in a PN generator.

SR output is just a shifted version of the previous stage's output, all outputs have the same average frequency for the PN generator. out_{PN} is an example of one of the PN generator outputs. The output pattern is a run of 1 ones, then a run of 3 zeros, 2 ones, 1 zero, 1one, etc. Since there is exactly one edge transition associated with each run of ones or zeros, knowing the distribution of the runs will allow a calculation of the average frequency with respect to the clock. From Section 3.2, the distribution of ones and zeros has a binary weighting function which will make the calculation simpler. From Table 3-1, as an example (M=7)[1], there are 32 (2^5) runs of ones and 32 runs of zeros when summing the second and third columns. Since two opposite edge transitions make up one cycle, there are 32 cycles of the PN output for 127 cycles of the clock. Thus the average frequency of the PN output is about 1/4 the clock frequency or by induction for any M-stage ***maximal***-length PN-sequence generator:

$$\frac{f_{out}}{f_{clock}} = \frac{2^{M-2}}{2^M - 1} \approx \frac{1}{4} \qquad \text{(C - 1)}$$

1. M is the number of stages (shift registers) in the PN generator.

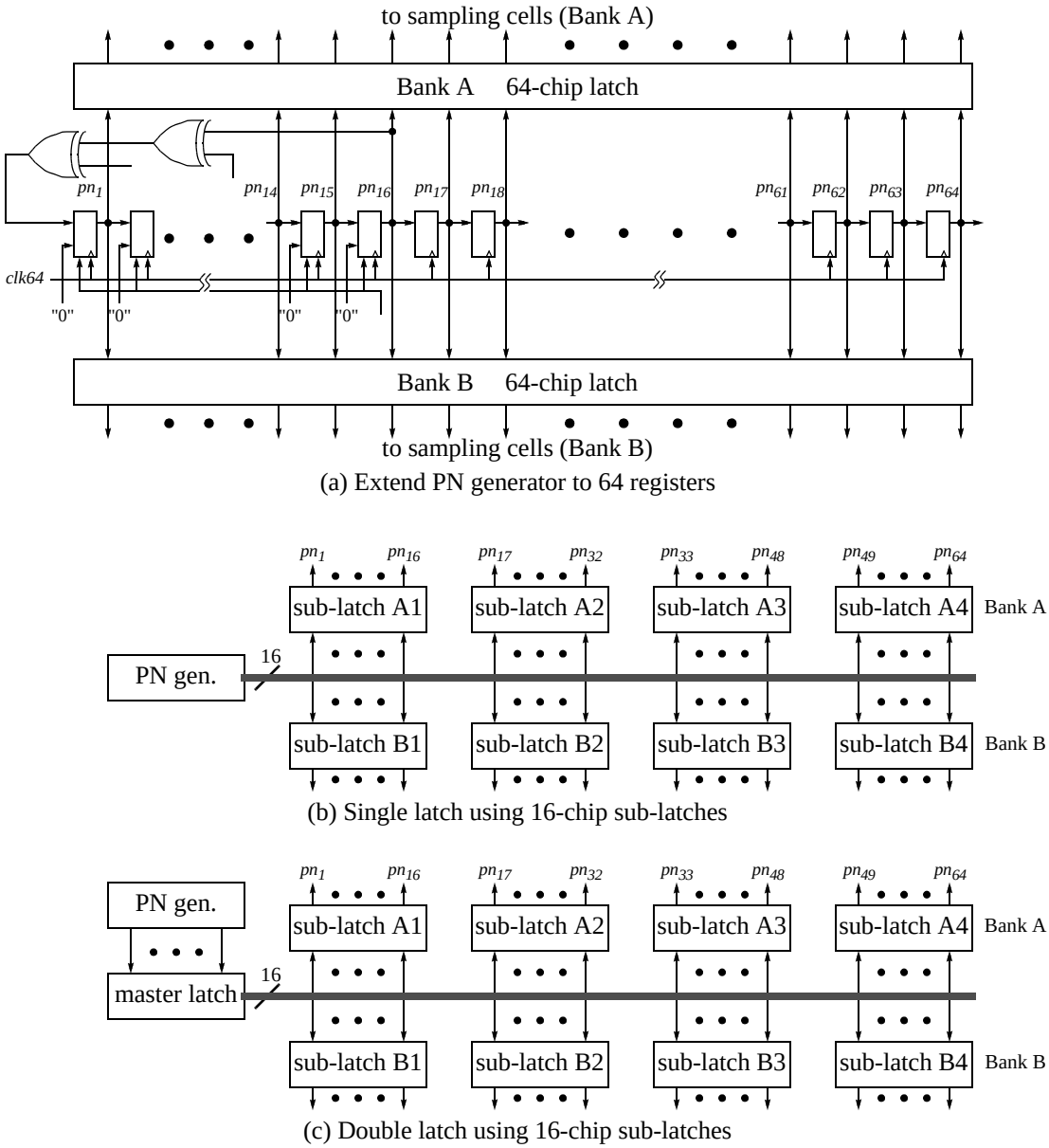


Figure C-2 Latching a 64-chip output from the PN generator.

C.3 PN Generator Power Calculations

This section will quantify the power dissipation of the three PN generator output latching approaches of Fig. C-2. This analysis will restrict the comparison to clock and

output power of the PN generator and the master latch. All process values pertain to the 1.2μm CMOS double-poly double-metal 5V technology used for the prototype IC.

From the process data, Table C-3 of capacitance values were calculated:

TABLE C-3 Capacitance values

Description	Symbol	Value
D-input of SR or Latch	C_D	16 fF
Clock input of SR or Latch	C_{clk}	33 fF
Metal line per SR or Latch width	C_m	10 fF

C.3-1 PN generator with 48 shift-register extension

In this configuration, the 64MHz clock would have a clock load of 64 SR clock-inputs and a metal capacitance load of 64 C_m (640fF). The register outputs would have a load of 128 SR D-inputs and an estimated metal capacitance of 4 C_m (40fF) per output for a total of 2.56pF. The outputs would have an average frequency of 16MHz. Table C-4 presents the power calculations based on the data above.

TABLE C-4 PN Generator with 48 SR Extension

Description	Freq	Load	Power
PN generator clock - SR	64 MHz	2.112 pF	3.379 mW
PN generator clock - metal line	64 MHz	.640 pF	1.024 mW
PN gen. outputs - SR D-inputs	16 MHz	2.048 pF	.819 mW
PN gen. outputs - metal lines	16 MHz	2.560 pF	1.024 mW
total			6.246 mW

C.3-2 Single-latch configuration

The clock would have a load of 16 SR clock-inputs and metal capacitance of 16 C_m

(160*fF*). The register outputs would have a load of 128 SR D-inputs, an estimated metal capacitance of 16 lines each of length 64 C_m feeding 2 latches (4 C_m) for 12.80*pF* and an average frequency of 16*MHz*.

TABLE C-5 Single-latch Configuration

Description	Freq	Load	Power
PN generator clock - SR	64 *MHz*	.528 *pF*	.845 *mW*
PN generator clock - metal line	64 *MHz*	.160 *pF*	.256 *mW*
PN gen. outputs - SR D-inputs	16 *MHz*	2.048 *pF*	.819 *mW*
PN gen. outputs - metal lines	16 *MHz*	12.800 *pF*	5.120 *mW*
total			7.040 *mW*

C.3-3 Double-latch configuration

The PN generator clock would have the same loads as in the single-latch configuration above and the master latch would have the same loads as the PN generator outputs above. The PN generator outputs would have a load of 16 SR D-inputs with an estimated metal load of 16 C_m and an average frequency of 16*MHz*. The clock into the master latch would have a load of 16 SR clock-inputs with a metal load of 16 C_m and a frequency of 4*MHz*. Thus the master latch outputs would have an average frequency of 1*MHz*.

The double-latching scheme significantly reduces the power dissipation and is a good example of the benefits of shifting circuitry to a lower operating frequency (*load-shifting*).

TABLE C-6 Double-latch Configuration

Description	Freq	Load	Power
PN generator clock - SR	64 *MHz*	.528 *pF*	.845 *mW*
PN generator clock - metal line	64 *MHz*	.160 *pF*	.256 *mW*
PN gen. outputs - SR D-inputs	16 *MHz*	.256 *pF*	.102 *mW*
PN gen. outputs - metal lines	16 *MHz*	.160 *pF*	.064 *mW*
Master latch clock - SR	4 *MHz*	.528 *pF*	.053 *mW*
Master latch clock - metal line	4 *MHz*	.160 *pF*	.016 *mW*
Master latch outputs - SR D-inputs	1 *MHz*	2.048 *pF*	.051 *mW*
Master latch outputs - metal lines	1 *MHz*	12.800 *pF*	.320 *mW*
total			1.707 *mW*