The Report Committee for Robert Alexander Fontaine
Certifies that this is the approved version of the following report:

Investigation of 10-Bit SAR ADC Using Flip-Flip Bypass Circuit

APPROVED BY
SUPERVISING COMMITTEE:

Supervisor:

Nan Sun

Deji Akinwande

Investigation of 10-Bit SAR ADC Using Flip-Flip Bypass Circuit

by

Robert Alexander Fontaine, B.S.E.E

Report

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

Master of Science in Engineering

The University of Texas at Austin

December 2013

Acknowledgements

I would like to thank Dr. Nan Sun for his enthusiasm in integrated circuit and mixed signal design. His lectures and classes were of the best quality and the most memorable from my time as a graduate student.

I would also like to thank Dr. Deji Akinwande for being this report's reader. His class in analog circuit design during my first semester as a graduate student was what sparked my interest in analog and mixed-signal design.

In addition I would like to thank the teaching assistant, Arindam Sanyal, for his assistance during the simulation stages of this report.

Abstract

Investigation of 10-Bit SAR ADC Using Flip-Flip Bypass Circuit

Robert Alexander Fontaine M.S.E.

The University of Texas at Austin, 2013

Supervisor: Nan Sun

The Successive Approximation Register (SAR) Analog to Digital Converter (ADC) is power efficient and operates at moderate resolution. However, the conversion speed is limited by settling time and control logic constraints. This report investigates a flip-flop bypass technique to reduce the required conversion time. A conventional design and flip-flop bypass design are simulated using a 0.18μm CMOS process. Background and design of the control logic, comparator, capacitive array, and switches for implementing the SAR ADCs is presented with the emphasis on optimizing for conversion speed.

Table of Contents

List of Tables

List of Figures

Chapter One: Introduction: Analog to Digital Conversion Background

INTRODUCTION

The Analog to Digital Converter (ADC) enables the physical world of analog quantities to be harnessed by the computational power of digital processors. Nearly every consumer electronic devices contains at least one ADC. The purpose of the ADC is to convert a continuous analog signal into a discrete or quantized time varying signal. The analog signal could be voltage, current, or charge, and is continuous in time and in amplitude. Digital systems cannot process this infinite set of values. Accordingly, the analog signal is divided into fixed intervals in time and value by the converter. This is known as sampling and quantizing, respectively.

The Successive Approximation Register (SAR) ADC is a common architecture for its moderate resolution, relatively quick conversion time, and simple circuit complexity. First introduced as charge redistribution architecture in 1975 by McCreary[1], it has become a popular ADC design in the last decade. Compared with other moderate resolution ADC architectures, SAR designs tend to be on the lower end of power consumption. This can be observed in Figure 1.1 where the SAR ADCs highlighted in red are clustered on the lower edge of power-per-conversion and with moderate resolution, expressed as SNDR[2]. A majority of SAR architectures from the survey have an effective resolution between 5 and 12 bits with power-per-conversion under 100pJ. The SAR ADC architecture is very efficient compared to other ADCs. The figure-of-merit (FOM) lines, shown in green, represent the best performance in contemporary research.

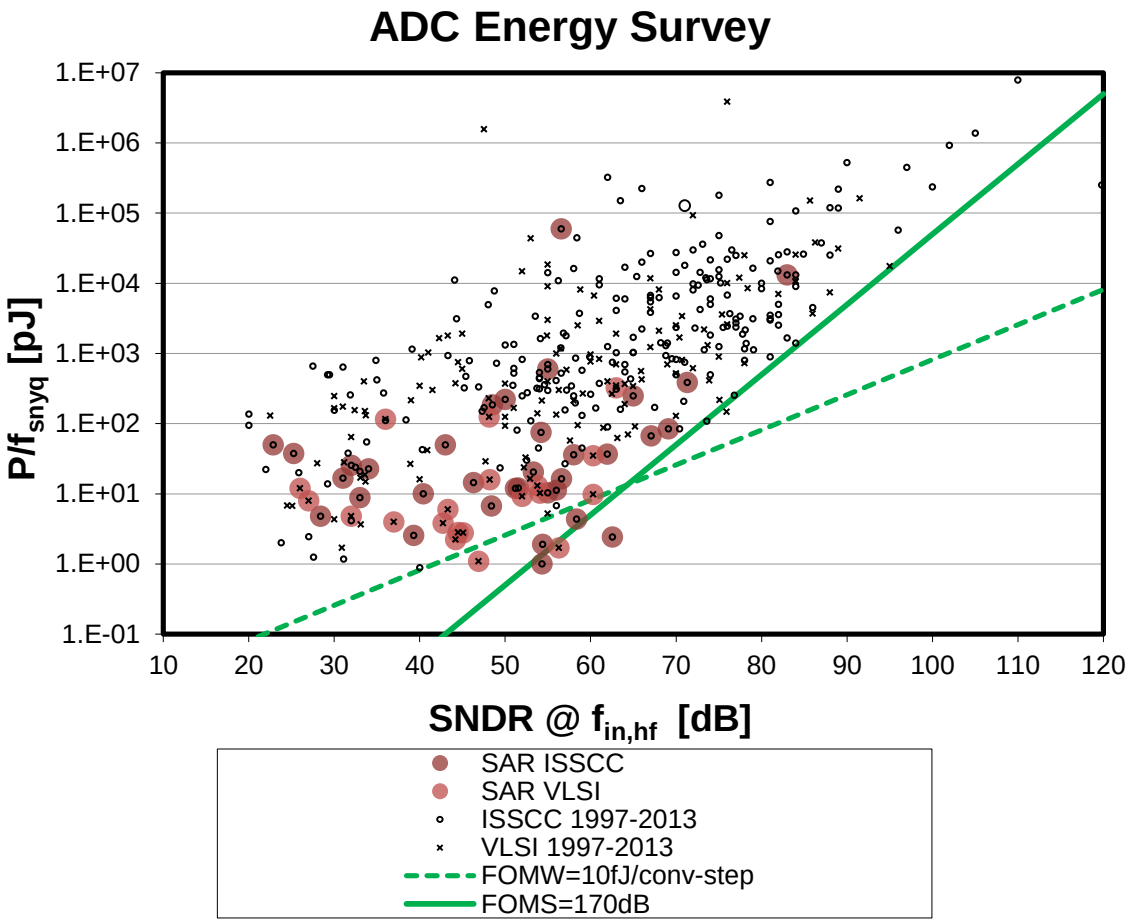

Figure 1.1: Analog to Digital Converter (ADC) Energy Survey[2] from IEEE Journals.

The conventional SAR ADC is comprised of a binary-weighted capacitor array, digital control logic and one comparator. The capacitor array performs two functions, sampling of the input and charge redistribution, thus acting as a DAC. Making use of the capacitor array as a sample circuit eliminates the need for an additional sampling circuit. The SAR ADC performs a successive search on the sampled input which results in a digitized approximation of the sample. Digital control logic is used to implement the search and to redistributing charge on the capacitive array. The comparator is driven by the capacitor array and a reference circuit, and the result is fed to the control logic. Once

a given input is sampled on the capacitor array, the control logic performs a binary search by redistributing charge on the array and comparing each step to a reference.

In this paper a 10-bit SAR ADC design will be discussed. This section will provide background required to qualitatively describe analog to digital converters and the SAR architecture. ADC concepts such as quantization, noise, sampling, and typical figures-of-merit will be presented. Then, the building blocks of charge redistribution SAR ADC will be presented. This background will be used in subsequent sections as the basis for a 10-bit SAR ADC design.

ANALOG TO DIGITAL CONVERSION BACKGROUND

In this section, the key background and qualitative descriptions of an ADC will be presented. These includes quantization noise, offset errors, gain errors, integral nonlinearity (INL), differential nonlinearity (DNL), signal-to-noise ratio (SNR), signal-to-noise and distortion ration (SNDR), dynamic range (DR), and the effective number of bits (ENOB). There are two broad categories of data convertors; Nyquist-rate converters and oversampling converters. Nyquist-rate converters are loosely described as generating a series of output values which have a one-to-one correspondence with the input. For example, a Nyquist-rate ADC generates a series of digital output codes, with each resultant code the result of a single sampled analog value. Oversampling converters typically operate at speeds 10 to 512 times greater than the input signal's Nyquist-rate[3]. The oversampled signal is used to improve output signal quality at the expense of circuit design complexity and power consumption. This report, however, is based on a Nyquist-rate design.

Sampling Theory

A sampler transforms a continuous-time signal into a discrete-time signal. The sampling frequency corresponds to uniformly spaced time intervals that the input is sampled. In order to preserve the continuous-time signal, the frequency of sampling must be twice that of the highest input frequency or bandwidth. This frequency is known as the Nyquist-rate. Signal components greater than the Nyquist-rate will have copies or aliases in the band of interest when sampled. The use of an anti-aliasing filter, typically a low-pass or band-pass, is to reject this unwanted interference from outside the band of interest.

Practical circuits are subject to non-ideal sampling. Sampling-time jitter is a deviance in the actual sampling instant around the uniform sampling period. The effects of sample jitter manifest in a degradation of SNR. However, so long as the jitter is sufficiently small, this noise source can be below the noise floor and thus not degrade SNR. The effects of sample time jitter, however, are largely ignored in this report.

Quantization

The ADC performs the function of quantizing a continuous analog value into a discrete digital value. To understand quantization, it is useful to discuss the ideal ADC. In the ideal converter, the digital code corresponds to analog signal quantities that are precisely equally spaced[3]. Quantization refers to converting the infinitely valued analog sample into a finite number of values. Typically, and in the case of this design, the quantization is uniformly spaced between a minimum and maximum voltage. The difference between the maximum and minimum voltage is known as the full scale voltage (V_{FS}). From this definition, the minimum voltage change, V_{LSB}, can be defined as the voltage change per total number of bits. The term LSB refers to a unit less value is defined as $½^N$, where N is the number of bits.

Quantization errors result from converting the analog sample to a digital representation. Quantization error, Q_e, is the difference between the actual analog input and the value of the output code given in voltage. Stated another way, quantization error is the ambiguity or loss in the actual analog value. By evaluating the quantization error from the perspective of a random variable distributed uniformly between $\pm V_{LSB}$, it can be shown that the quantization error contributes white noise in the output signal[3]. The ratio of the signal-to-quantization-noise ratio (SQNR) is equal to $20 \log 2^N$ or 6.02N dB for the ideal converter. Quantization error sets a noise floor limitation of the converter.

Noise

While quantization errors set a fundamental noise limit, other noise sources and circuit design can further degrade the quality of the converter. Thermal noise is an unavoidable factor and sets constraints on the data converter. The sampling capacitor used in many systems will not only sample the signal but also sample noise. Unavoidable thermal noise from the sampling switch is transferred to the sampling capacitor. This resultant noise, or *k*T/C noise, however, is purely a function of capacitance. The sampling capacitance must be sized sufficiently large such that the *k*T/C noise does not degrade the overall SNDR. Sizing the capacitance such that *k*T/C noise is equal to quantization noise is shown in Equation 1.1. Sized in such a way, the *k*T/C noise will only contribute to the noise floor.

$$C = 12kT \times (\frac{2^B}{V_{FS}})^2 \quad (1.1)$$

Thermal noise related to the input signal can be expressed as $SNR_{therm} = \frac{V_{FS}^2 C_s}{8kT}$. Increasing the sampling capacitance improves the thermal noise ratio at the expense of settling speed and circuit area.

Static Performance

Static Performance metrics describe the input-output characteristics of the converter. The ideal ADC input-output characteristic is a staircase of uniform steps over the entire converter range, V_{FS}, with each step size equal to V_{LSB}. Deviations from this ideal input-output transfer function contribute to static errors. Static errors are summarized as follows:

- Offset is a shift in the ideal transfer function such that all uniform steps are shifted by the same offset.
- Gain errors represent a difference in slope between the actual and ideal transfer functions.
- Differential nonlinearity, DNL, is the difference between the actual step width and the ideal step width or 1LSB. The maximum DNL is often reported as the characteristic DNL for the data converter and is a means to measure non-idealities in step width. DNL specification less than or equal to 1LSB ensures monotonicity in the ADC.
- Integral Nonlinearity (INL) is the accumulation in the difference of actual step size and ideal step size. The differences result in a deviation in the transfer function endpoint-fit line from the ideal endpoint-fit line. The use of the endpoint-fit line in measuring INL corrects for offset and gain errors. INL is informative for estimating harmonic distortion.

Dynamic Performance

Dynamic performance metrics characterize the frequency response, speed and power of the converter.

- The signal-to-noise, SNR, is the ratio of the rms value of the full-scale input signal to the rms value of the quantization error. This is a ratio of

the power of full-scale input signal to that of the quantization error. In decibels SNR can be expressed as 6.02N + 1.76, where N is the converter's resolution.

- The signal-to-noise-and-distortion, SNDR, includes distortion in addition to noise. This is the ratio of the power of the signal to the sum of the quantization noise power and distortion or harmonic power.
- The effective resolution, or ENOB, includes the effects of noise and distortion. ENOB is expressed as (SNDR – 1.76)/6.02.
- Spurious Free Dynamic Range (SFRD) is the ratio of the fundamental frequency power to the power of the largest non-fundamental power peak in the band of interest. It provides a measure of the harmonic distortion due to the offending spurious tone.

Power Performance

Power performance metrics evaluate the converter on its power efficiency. Metrics related to power efficiency versus conversion speed or energy per conversion step are used as figures-of-merit when comparing to other data converters. Commonly used is the power-per-conversion step, which evaluates how many bits a converter can convert per unit of power. Another metric useful for comparing ADCs with similar ENOB is the energy-per-Nyquist sample, a ratio of the power-per-conversion bandwidth.

SUCCESSIVE APPROXIMATION ANALOG TO DIGITAL CONVERTER BACKGROUND

The SAR ADC is a popular ADC architecture due to power efficiency, moderate speed, and moderate resolution. The power efficiency, resolution, and speed are also the major design tradeoffs. Background on the SAR algorithm and key building blocks are presented next to describe these tradeoffs

The conventional SAR ADC consists of four main blocks: sample-and-hold, comparator, a digital-to-analog converter (DAC), and a control logic block, Figure 1.2. This conventional architecture is commonly implemented with a binary weighted capacitive array. The capacitive array functions as a digital-to-analog-converter (DAC) in addition to sampling the input signal. The capacitive array stores this information in the charge domain. A control logic block implements the binary search algorithm by redistributing charge on the DAC. For each step in the search algorithm, a comparator evaluates the difference between the reference voltage and DAC voltage. The comparator decision is stored by the control logic as each successive bit decision is made.

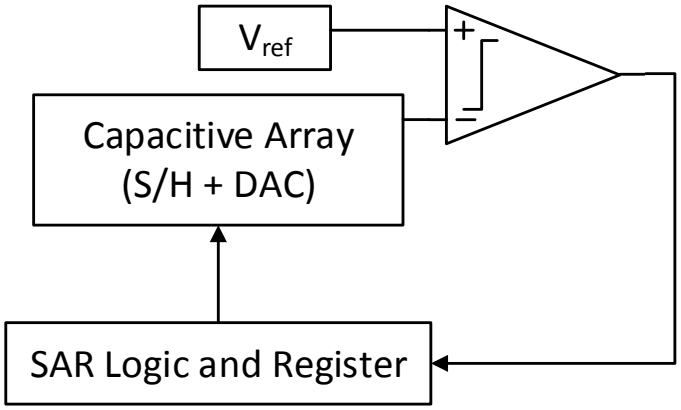

Figure 1.2: Successive Approximation Register ADC Block Diagram

A conceptual successive approximation algorithm is depicted in Figure 1.3. First, the sampled input voltage, V_{in}, is stored. The comparator evaluates if V_{in} is greater than the voltage generated by the DAC. The result of the comparison is stored as a bit_i of the digital code. The next cycle either increments or decrements the voltage generated by the DAC by a value of V_{ref} half as large as the previous step. This process of updating

V_{DAC} is a binary search algorithm. After N cycles, the algorithm completes and the resulting digital code is stored.

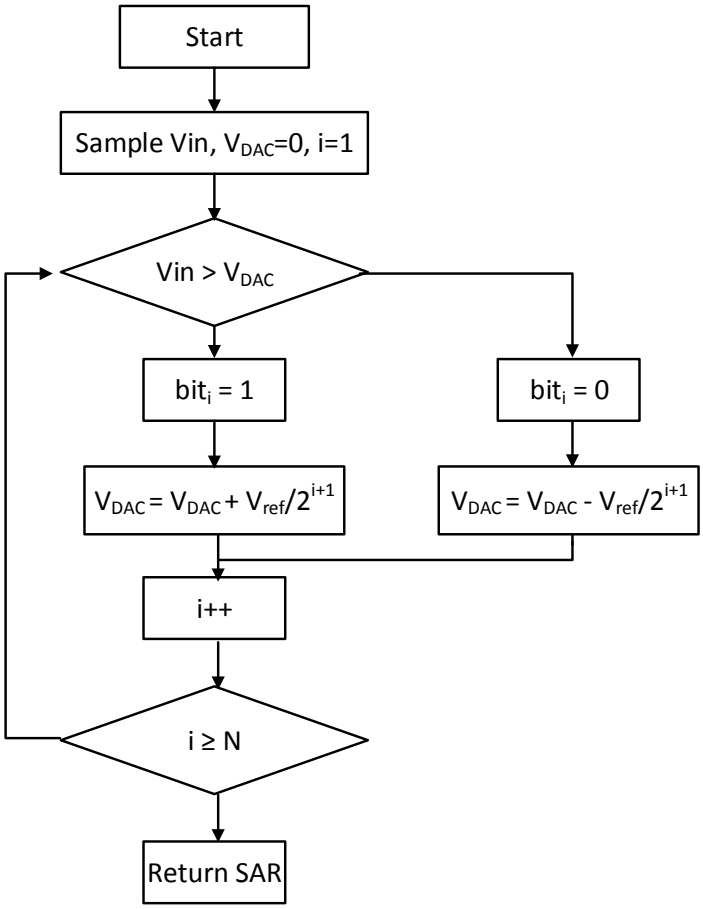

Figure 1.3: Conceptual SAR Algorithm

CHARGE REDISTRIBUTION EXAMPLE

To understand the operation and charge redistribution of a single ended SAR ADC a simplified 3-bit example will be used. In this example, V_{in} is a input signal from 0V to V_{ref}. During the sample phase, all bottom plate switches are connected to the input, V_{in}, and the top plate switch connected to ground. The DAC array consists of N + 1 capacitors, with a binary scaled capacitance from 2^{N-1} per bit plus a dummy capacitor.

The total capacitance equals 2^NC, or in this example 8C. During the sampling phase a total charge of $Q = 2^N C \times (0 - V_{in})$ is stored on the top plate connected. After sampling, the top plate is disconnected and the bottom plate switches are all connected to ground so that the input charge is held on the DAC. Due to conservation of charge, the voltage on the top plate after sampling equals $-V_{in}$, Figure 1.4.

From this point, the binary search approximation algorithm begins. The most significant bit (MSB) capacitor, with a value of 2^{N-1}C, is tested first. This is done by switching the bottom plate of the MSB switch to V_{ref}. The resulting voltage on the top plate is $V_x = \frac{V_{ref}}{2} - V_{in}$. Next, the comparator determines if $V_{ref} - V_x$ is greater than 0. The SAR logic leaves this switch connected if the comparator result is greater than zero, or otherwise the switch is connected back to ground. From this point, next bit in the conversion is tested. The resulting voltage on the top plate is either $V_x = \frac{V_{ref}}{4} - V_{in}$ or $V_x = \frac{3}{4}V_{ref} - V_{in}$, depending on the result of the previous stage. The comparator again determines if $V_{ref} - V_x$ is greater than 0 and either leaves the switch connected to V_{ref} or switches it back to ground.

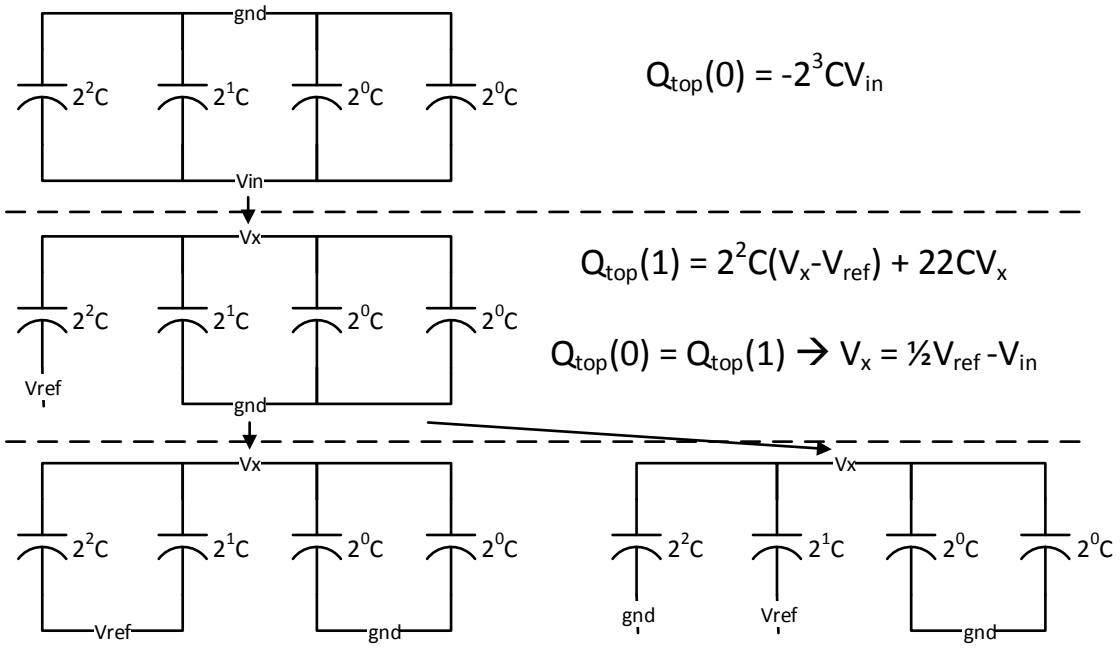

Figure 1.4: 3-bit Charge Redistribution Example

The final bit is tested based on the result of the previous stages and the converter converges on an approximate solution. A generalized solution for the impact of the reference voltage alone on the output of the DAC can be expressed as:

$$V_{DAC} = \left(\frac{Bit_N}{2} + \frac{Bit_{N-1}}{2^2} + \cdots + \frac{Bit_2}{2^{N-1}} + \frac{Bit_1}{2^N}\right) \times V_{ref}$$

Any error during a bit decision will propagate down the search path and result in an incorrect digital code. There are methods to recover from an incorrect bit decision. However, these require extra control logic to evaluate the error[4] and are beyond the scope of this report.

SWITCHED CAPACITOR SAMPLING

Sampling of the analog input signal is accomplished using a sample-and-hold circuit or track-and-hold circuit. As the first step in the data conversion process, the sample-and-hold circuit sets fundamental speed and signal integrity limits. Thus, the

design of the sampling stage is critical in achieving overall performance metrics of the data converter. Any non-idealities in the sampling stage will show up in the overall converter, so effort is made to reduce their effects. The track-and-hold function is often implemented using a capacitor and switch. During the on-phase of the switch, the input signal is followed or "tracked" on the capacitor. During the off-phase of the switch, the input signal is isolated from the capacitor and signal value on the capacitor remains constant or is "held". The sample-and-hold can be implemented by cascading two track-and-hold circuits where one track-and-hold operates in the inverse phase.

The switch and capacitor model of this sampling circuit performs essentially a passive sampling function. To achieve an accurate sample, the sampling window must be sufficiently large, such that the sample can be quantized to ½LSB. The time constant of the switch resistance and capacitance governs how much time is required for the sampling window. To achieve high speed in sampling requires small switch resistance and small sampling capacitors. The minimum sampling period from an RC time constant can be described with the following equation to settle to ½LSB:

$$T_{min} > -\ln\left(\frac{1}{2^{N+1}}\right) \times \tau$$

For 10-bit performance, 7.6 time constants are required to settle to ½LSB.

SAMPLING CAPACITOR

The trend in decreasing capacitor size is constrained by thermal noise and capacitor mismatch due to layout and parasitic capacitance. Noise associated with the switch is sampled by the capacitor and this contribution is known as *k*T/C noise. Because the corner frequency due to the sampling capacitor and switch is much higher than the Nyquist-rate frequency, the resulting noise spectrum is approximately uniform

across the band of interest. Energy is also dependent on the size of the capacitor, as depicted in the following equation:

$$E = \frac{1}{2} C_{total} V_{in}^2$$

Fabrication of capacitor in CMOS is usually done with a metal-insulator-metal, MIM capacitor. For the 0.18µm CMOS process a 1fF consumes approximately 1µm^2 of die area.

SAMPLING SWITCH

Switches are implemented using MOS transistors which introduce a number of non-ideal effects in sampling and overall performance. For one, the switch has a non-zero on-resistance. The on resistance, R_{on}, derived from the square law equations can be seen in Equation 1.2. The on-resistance has a non-linear relationship with the input voltage which can introduce harmonic distortion. The bootstrapped switch is a common method for reducing this distortion.

$$R_{on} = \frac{1}{\mu C_{ox} \frac{W}{L} (V_{in} - V_{th})} \quad (1.2)$$

In addition, the MOS transistor parasitic capacitances inject charge into the sampling capacitors and degrade linearity. When the MOS switch is on, there is a small voltage drop across the source and drain terminals. When the MOS switch is switched off, a finite amount of charge is injected into the source and drain terminals. This is a result of charge migrating out of the inversion layer as the transistor is switched off. Slow clock transition and long channel lengths can increase the amount of injected charge. There are several methods to mitigate this undesirable charge injection such as use of a dummy transistor or complimentary PMOS and NMOS transmission gate (TG) switch. For the transmission gate, a majority of the charge injection is absorbed by the complimentary devices.

Reduction in the charge injection can be also obtained by using the bottom-plate sampling technique. The bottom-plate sampling technique adds a signal independent early switch to isolate the injected charge to the parasitic or bottom-plate terminal of the capacitor. A fundamental building block of the bottom plate sampling and switched-capacitor circuits is the non-overlapping clocks generator. The non-overlapping clock generator provides clock and early-clock signals which are never on at the same time. These techniques to reduce distortion and charge injection need not be applied to every switch but are critical on the sampling switches.

There are tradeoffs in the switch design are towards power, settling time, and noise/linearity. The settling time is preferably constant for each switch network and sampling capacitor so that the signal or reference may settle within the clock period. To achieve a constant settling for a binary weighted capacitor array, the switch resistance would need to scale appropriately. Since the switch-on resistance is inversely proportional to the W/L ratio, the switch sizing would also need to scale in the same binary fashion per bit. However, much like the capacitor sizing, the switch dimensions increase on the order of 2^N. Power dissipation and input capacitance per switch are increased similarly. Since this design uses a binary weighted capacitor array, the optimizations will be considered in the design of the switches, rather than optimizing the capacitor array or using a different capacitance sizing scheme.

SAMPLING ARRAY

The sampling array is typically a binary weighted array. The total capacitance of the array is 2^N times the unit capacitance. The binary weighted array allows for a direct implementation of binary scaled charge redistribution. However, with each successive bit, the capacitance and die area increase exponentially and switching energy increases.

Capacitor matching constraints for the technology will dictate the minimum unit capacitance required. The unit capacitance is constrained by matching, for accuracy of the ADC, and the *k*T/C noise. Layout considerations of the capacitors also impact the accuracy of the final design and techniques such as centroid layout to mitigate gradient variation and dummy capacitors for shielding. The binary weighted array used in this design is presented in Figure 1.5. It consists of the top plate switch connected to a common voltage and bottom-plate switches connected to the input signal and reference voltages.

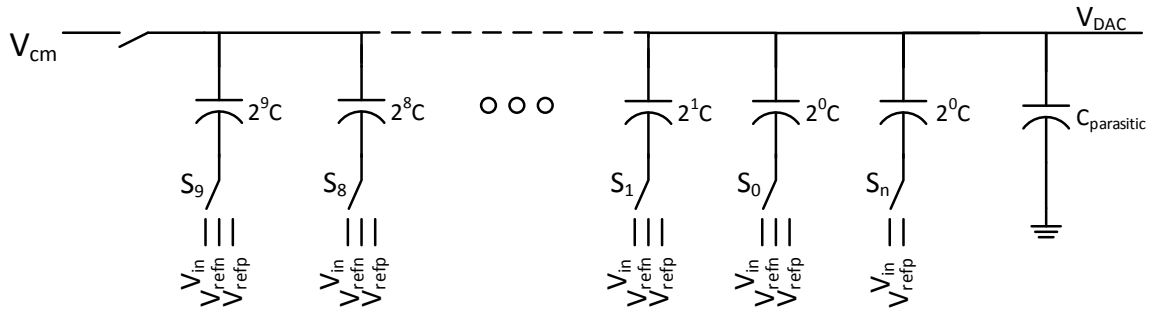

Figure 1.5: Binary Weighted Capacitor Array

One thing to note is the parasitic capacitance shown in Figure 1.5. This capacitance is the result of parasitic elements including interconnect capacitances. However, the parasitic capacitance contributes little overall charge error since the parasitic charge at the final stage of the approximation is very near the parasitic charge collected during sampling[1]. In addition, there are two instances of capacitor 2^0C, the additional instance is necessary for the last stage in the charge redistribution and to bring the total capacitance to 2^NC.

Other capacitive array topologies exist and include unary, a uniform size for all capacitors, or using other encoding techniques such as ternary[5]. Techniques to reduce

the overall capacitance are also available and include split or multi-stage capacitive topologies. A two-stage binary-weighted capacitor array consists of two binary scaled sub arrays and a scaling capacitor in series connecting the two arrays. This approximately reduces the total capacitance in half, but care must be taken to size the scaling capacitor and to mitigate parasitic capacitance[6]. Calibration techniques can also be utilized to improve precision beyond what is available by the technology alone.

COMPARATOR

The comparator is a key component to ADC designs. The SAR ADC has the benefit of using a single comparator, unlike other architectures such as flash which require multiple. Comparator metrics include input referred, decision speed, and power.

The StrongARM comparator topology is used as the main component of the comparator in this design. First introduced in [7] and used as a latch in a RISC processor design[8], it has been widely used in contemporary comparator designs and has the benefit of leakage-only static power dissipation. A regenerative latch is the key block of the StrongARM comparator, Figure 1.6. The latch is essentially a pair of back-to-back inverters and provides positive feedback. A differential pair input presents the regenerative latch with a difference signal. Positive feedback regenerates the analog input difference into a full-swing digital signal. The positive feedback configuration allows for the comparator to make a fast decision. The time constant of the latch during the regenerative phase is largely a function of the unity-gain frequency of the inverters and the capacitive load[9].

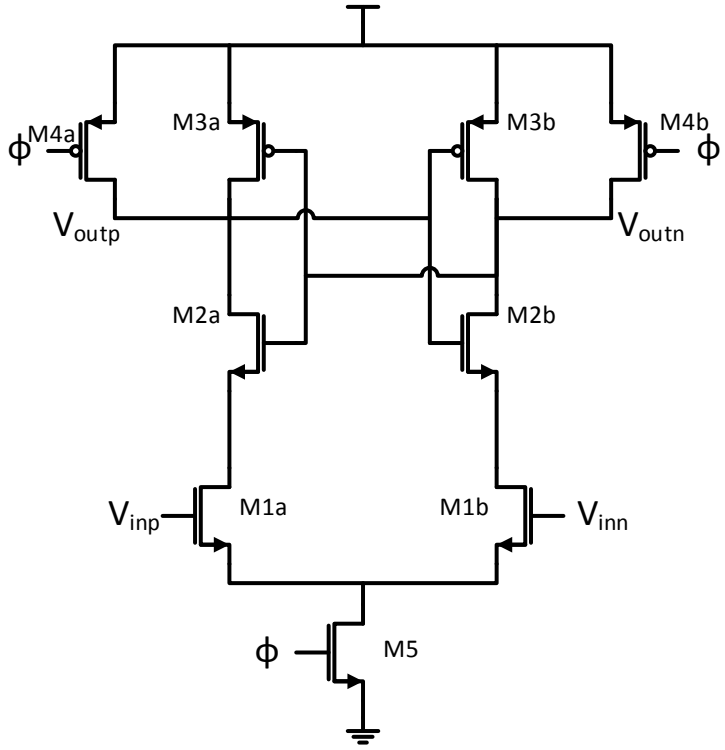

Figure 1.6: StrongARM Latch Comparator

As the input difference becomes increasingly small, the time the latch takes to regenerate increases. Metastability occurs when the input difference is sufficiently small such that the regenerative process does not produce a discernable digital output within a desired time. The input referred offset is a key parameter of the comparator which describes the voltage at which the output changes from one logic level to the other. A metastable condition will always resolve, however, due to any difference affecting the current through an inverter. Mismatch primarily in the input differential pair and noise not only affect the input referred offset, but will drive the comparator out of a metastable state. Load capacitance mismatch also contributes an offset to the latched comparator. Thus, it is important to ensure the comparator drives the same capacitive load on the output[10].

The StrongARM topology, Figure 1.6, also contains reset transistors M4 and M5. These provide a means to reset the output nodes to Vdd when the comparator clock is low. The tail transistor M5 is off during this phase and there is no supply current and thus only static leakage power dissipation. When the clock goes high, the input differential pair provides a differential current. This difference is sensed by the regenerative latch, transistors M2 and M3, whose back-to-back configuration provides positive feedback to regenerate a digital difference on the output nodes. After regeneration is complete, the output nodes will be either at Vdd or Gnd and there will only be static leakage power dissipation, which maximizes power efficiency.

Kickback noise or clock feed through can be observed on the inputs to the differential pair, and are an undesired characteristic of this design. This originates from changes in the operating region of the transistors from cutoff, when the comparator is reset, to saturation, when the comparator clock goes high, to triode during the regenerative process. The changes in operating region are accompanied with changes in the gate charge, causing input voltage variation[11]. This charge kickback will be stored on the capacitive sampling array.

A preamplifier circuit with minimal gain is often added to the regenerative latch comparator. This provides the benefits of lowering the input referred offset, common-mode rejection, and attenuation of kickback noise. However, the preamplifier will increase the comparator delay and power consumption.

CONTROL LOGIC

Conventional SAR control logic consists of a shift register used to synchronize an N-bit register, Figure 1.7. As the shift register is clocked, a bit in the N-bit register will be set. This bit drives a switch and also clocks the previous bit to latch result of the

comparator. The conversion cycle completes when the shift register reaches the last bit, and the resulting digital value of the input is stored on the N-bit register. Each bit takes one clock cycle during the conversion. Timing requirements of the switch logic, DAC settling and comparator decision must all occur during that conversion cycle.

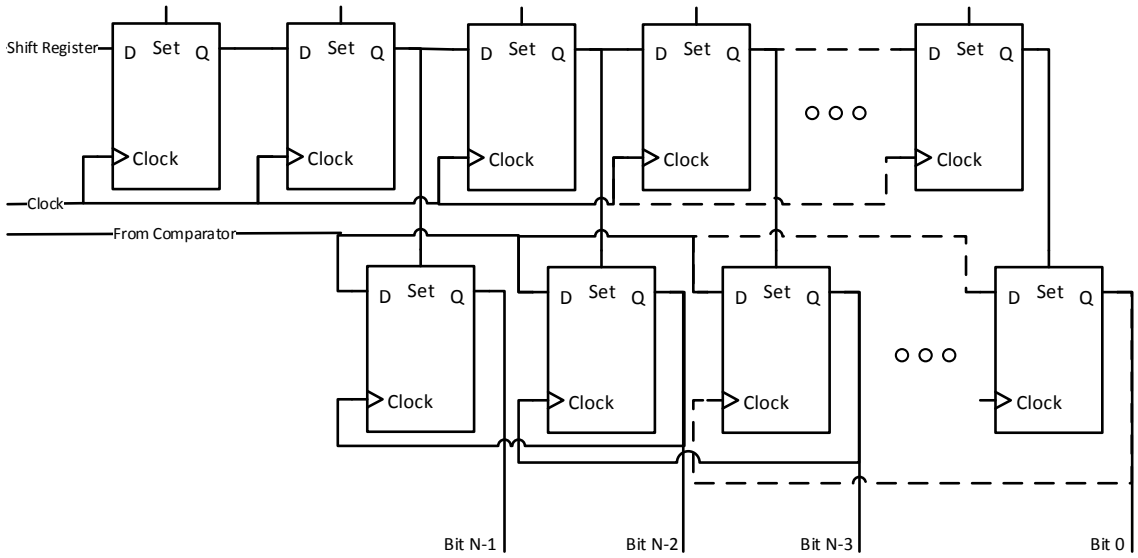

Figure 1.7: Conventional SAR Control Logic

Additional states are used for sampling of the input signal and to control switches related to sampling. Signal sampling may consume more than one clock cycle as the sampling network settling time requirements may differ from charge redistribution settling. The sampling state may also control early signals required for bottom plate sampling.

Chapter Two: SAR ADC Design Process

The design procedure made use of the idealized models as a reference to compare to transistor level designs. Changing one component or block at a time from an ideal model to a transistor level model was a useful procedure for debugging and optimizing the transistor level designs. In addition, the simulation speed of the Verilog-A and idealized models were useful in shortening simulation time. As design progressed, the Verilog-A models were updated to more accurately model the transistor level design. Verilog-A models were also used to extract data from the simulations such the digital output of the ADC. All Flip Flops were implemented using standard cell libraries provided in the 0.18 CMOS technology library. Only one type of flip flop was used to maintain consistency when comparing designs.

Two SAR ADCs were developed for this report, a conventional SAR and a flip-flop bypass SAR. The flip-flop bypass design is an attempt at increasing the speed of the converter. Aside from the control logic and comparator clock timing, the two designs are identical. All SAR ADCs are speed limited by the settling time of the DAC to settle to meet the desired precision. A unit capacitance of 1fF is used in both designs. Under ideal conditions, DAC settling would be the only speed limitation. However, there are other timing constraints that must be accounted for that limit the speed.

IDEAL BUILDING BLOCKS

A mix of ideal components using Cadence analogLib and Verilog-A models were used in the design. Ideal components are useful for a first pass at circuit design and are very fast to simulate. Initially, all components were modeled using ideal components. These include ideal switches, an ideal comparator, ideal non-overlapping clock generator, and an ideal control logic model. The ideal components should match the transistor

components as best as possible for accurate modeling. As the design progressed, the ideal components can be replaced with transistor components. For example, initial designs of the bootstrapped switch used ideal switch components act as a charge-pump on a transistor level switch. These ideal components were later replaced with transistors.

Verilog-A models were used extensively in the design. Much like digital behavioral simulation, the Verilog-A models abstracted only the behavioral function of the register and control logic to be used in an Analog simulator. Verilog-A models initially used an idealized DFF modeled after the 0.18µm CMOS library standard cell to account for rise, fall, and signal delay. This was replaced by a single Verilog-A model of the control logic which further increased simulation speed. This model took the comparator input and clock and created switch control signals for the capacitive array, with rise, fall, and propagation delays.

TECHNOLOGY CHARACTERIZATION

For a top down design methodology, it is necessary to characterize the process technology so that the design models represent the technology. The SAR design is simulated for a 0.18µm CMOS process. Figure 2.1 plots the current efficiency and transient frequency of PMOS and NMOS 0.18µm transistors with a W/L ratio of 2µm/0.18µm. This metric provides a useful insight into how the technology can perform and to scale the size of the devices. Metrics such as the transient frequency determine the regeneration speed of the comparator.

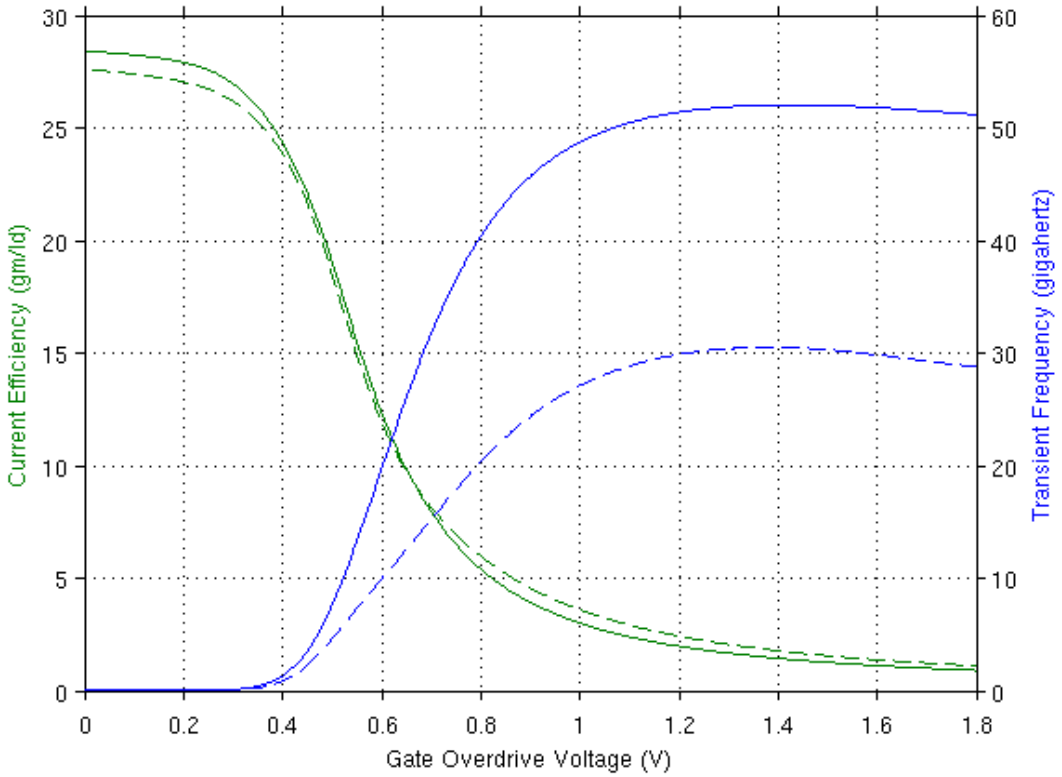

Figure 2.1: Transistor Efficiency versus Transient Frequency for $W_n/L = 2\mu m/0.18\mu m$ and $W_p/L = 4\mu m/0.18\mu m$ (Solid lines represent NMOS and dashed represent PMOS)

Similarly, the on resistance, R_{on}, is plotted in Figure 2.2. The on-resistance is useful for determining the switch sizing. The dashed line represents the on-resistance of both NMOS and PMOS devices configured as a transmission gate. The design makes heavy use of the transmission gate for a relatively constant output resistance across the entire dynamic inputs range. R_{on} varies between 1800 and 3600 Ohms across the range of gate voltages for a 2µm (nmos) and 3µm (pmos) transmission gate.

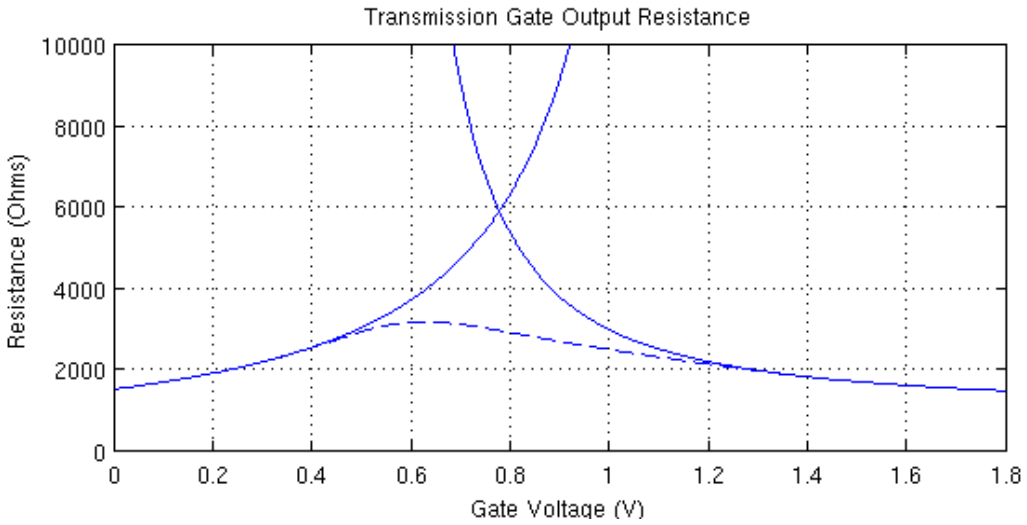

Figure 2.2: Output Resistance for $W_n/L = 2\mu m/0.18\mu m$ and $W_p/L = 3\mu m/0.18\mu m$

SWITCH DESIGN

Switches were initially designed using transmission gates to implement all switches. The transmission gates require an inverter to drive complimentary signals on the gate. This inverter used for complimentary switch signals causes a delay between the switching of NMOS and PMOS devices. Compensating for this delay can be achieved by using an appropriately sized transmission gate. In later design revisions, however, the transmission gates for input sampling and common mode sampling were replaced with bootstrapped switches. The bootstrapped switches ensure a constant on-resistance to minimize nonlinear charge injection.

Two input sampling switch designs are shown in Figure 2.3. Both designs contain a bootstrapped NMOS and a switch network for charge redistribution. The transmission gate design places a transmission gate between the bootstrapped switch and the reference switches. The second design is based on three transistors and a mux to select for sampling was tested as an alternative to the transmission gate design. During sampling of $V_{in,}$ a bootstrapped NMOS switch is turned on while the V_{ref} switch are

switched off using a mux to control their gate voltages. To redistribute charge the mux selects the switch signal which drives an inverter to select V_{refp} or V_{refn}. The transmission gate design proved more effective at isolating the sampling phase from the charge redistribution phase in the final design. However, unlike the mux design, the transmission gate on-resistance in the charge redistribution path adds settling delay.

The switch sizing can be seen in the Table 2.1 as well as the respective settling time constants. The resistance is based the output resistance of the transmission gate at 0.9V. As stated earlier, for 10-bit performance, 7.6 time constants are required for settling. For charge redistribution the DAC will settle in approximately 805ps. Additionally, a switch is required for resetting V_{DAC} to a common voltage during sampling. This switch increases the resistance of the sampling network and thus the amount of time required for ½LSB settling. For this design, the V_{CM} is also bootstrapped NMOS to improve linearity. Total settling time during sampling is approximately 1.65ns. Charge redistribution takes approximately 190ps for 10-bit performance.

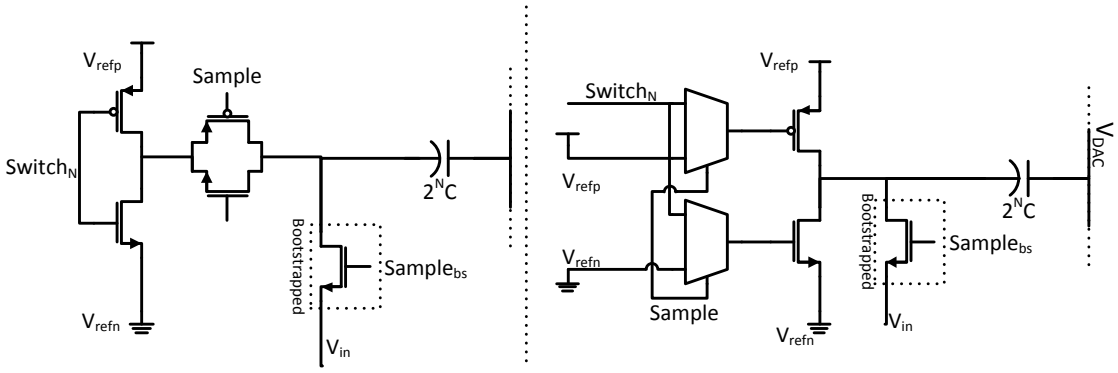

Figure 2.3: Sampling Array Switch Design, Transmission Gates (right) and Mux (left)

Bit	Nmos	Capacitor	R_{on}	$\tau = RC$
Bit 0	0.44µm	1fF	6100 Ω	244ps
Bit 1	0.44µm	2fF	6100 Ω	122ps
Bit 2	0.44µm	4fF	6100 Ω	24.4ps
Bit 3	0.88µm	8fF	3050 Ω	24.4ps
Bit 4	1.76µm	16fF	1660 Ω	25.5ps
Bit 5	3.52µm	32fF	832 Ω	26.6ps
Bit 6	7.04µm	64fF	416 Ω	26.6ps
Bit 7	14.08µm	128fF	208 Ω	26.6ps
Bit 8	28.16µm	256fF	104 Ω	26.6ps
Bit 9	56.32µm	512fF	52 Ω	26.6ps
V_{CM}	28.16 µm	1024fF	104 Ω	106ps

Table 2.1: Transistor Switch Sizing for Capacitive Array

BOOTSTRAPPED SWITCH DESIGN

Sampling switches were implemented using the bootstrapped technique similar to that of Abo[12]. The principal of operation is to keep the gate-to-source voltage, or V_{gs}, constant by using a charge-pump. Charge-pump circuits provide voltages that are higher than the supplied voltage. Transistors involved in the charge-pump are susceptible to failure resulting from gate-to-source voltages beyond the process technology. High electric fields can break down the oxide layer and short channel lengths can inject hot carriers which degrade the channel. It is necessary to ensure the gate voltages in bootstrapped circuits do not stress the technology.

The bootstrapped circuits were added to the NMOS devices on the V_{in} switches. The input sampling switches are most important for ensuring a linear sample. The

bootstrapped circuit can be seen in Figure 2.4. The charge-pump consists of a pair of coupled NMOS transistors, M_{p1} and M_{p2}, with capacitors on their source nodes. As the charge-pump is clocked, the gate-to-source voltage is approximately doubled. The bootstrapped circuit applies this boosted gate-to-source voltage to the gate of M1 during the on-phase of the clock and to Vdd during the off-phase. A boosting capacitor, C_B, stores a charge of Vdd. This charge is applied to the gate of the switch transistor via M3 and M4. The V_{gs} of M4 tracks the input signal, thus boosting the gate voltage of the switch by Vdd when clocked.

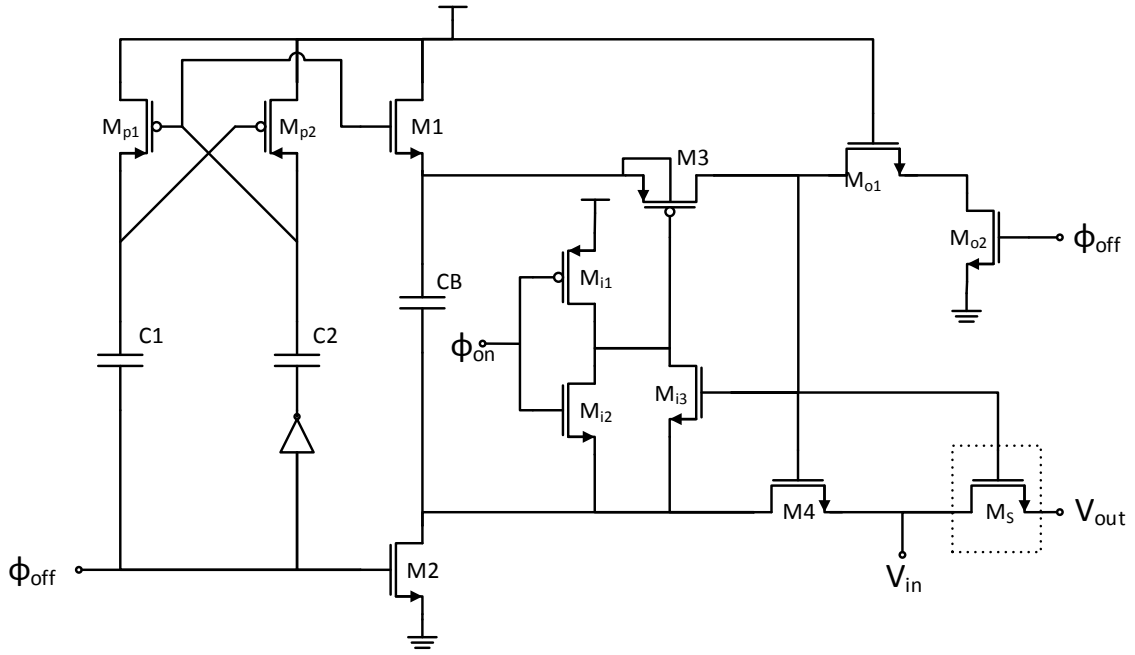

Figure 2.4: Bootstrapped Switch Circuit

A sample of the bootstrapped circuit tracking a signal can also be seen in Figure 2.5. This bootstrapped circuit is controlling a 9µm/0.18µm switch with a 500fF load. In the first clock cycle the charge pump begins to charge C_B. By the following clock cycle, C_B is fully charged to approximately V_{DD}. The bootstrapped gate voltage tracks the input

signal by approximately V_{DD} when clocked. Sizing of the transistors of the bootstrapped circuitry can be seen in Table 2.2.

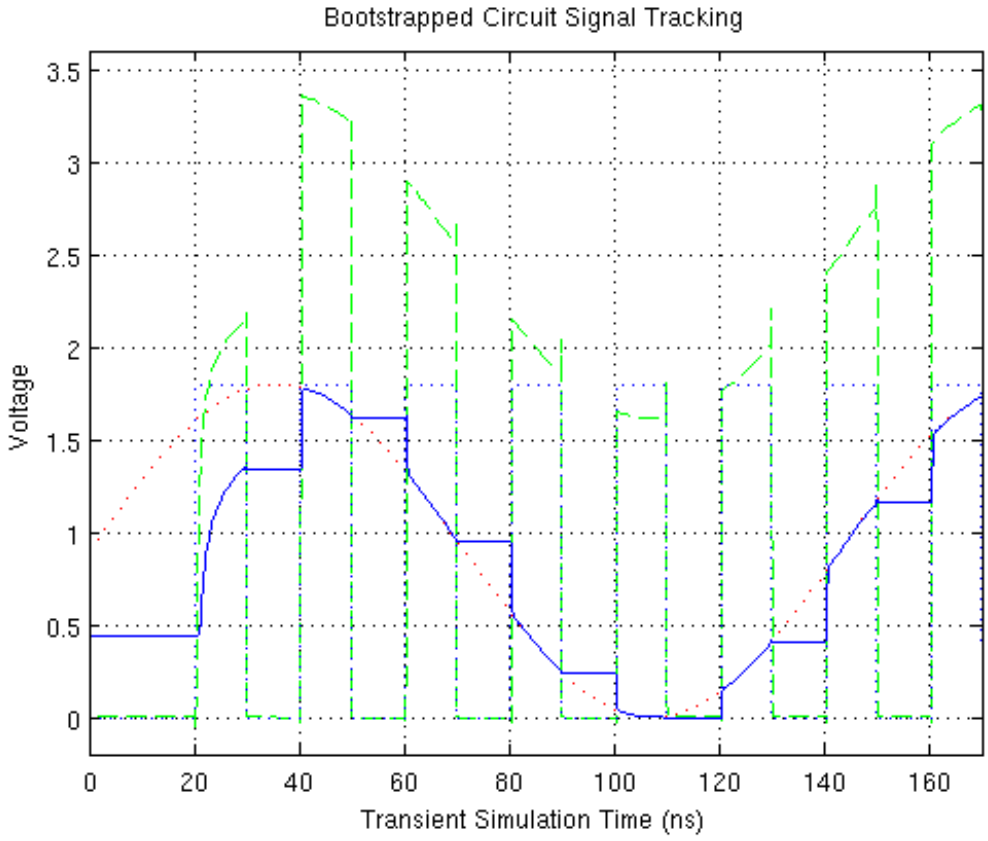

Figure 2.5: Transient Analysis of Bootstrapped 9µm NMOS Switch and 500fF Load

M_{p1}, M_{p1}	3.6µm/0.18µm
C1, C2	50fF
M1, M2, M4	3.6µm/0.18µm
M3	6.4µm/0.18µm
CB	500fF
M_{i2}, M_{i3}, M_{o1}, M_{o2}	3.6µm/0.18µm
M_{i1}	5.4µm/0.18µm

Table 2.2: Bootstrapped Circuit Sizing

FULLY DIFFERENTIAL SAR IMPLEMENTATION

The previous example presented a single ended SAR architecture. Taking advantages of differential circuit design, the SAR designs in this report are based on a fully differential topology. Differential circuit design doubles the input signal voltage range, improves common-mode noise rejection, and reduction of even-order harmonic distortions. It is advantageous to apply a differential design to the SAR for the benefits of larger voltage input range on the comparator, noise reduction, and distortion reduction. However, the differential design negatively affects the power consumption, circuit area, and circuit complexity.

The fully differential sampling array in Figure 2.6 contains two arrays for the differential inputs. The top plates of both arrays have a switch which can connect to a common mode voltage, V_{cm}, which is set to $\frac{1}{2}V_{FS}$ or 0.9V. During sampling of the inputs, the bootstrapped V_{in} switches and common mode switches are connected. The differential V_{in} signal is sampled on the arrays. The common mode switch is turned off by an early signal to allow for bottom-plate sampling of the input. For charge redistribution, the two arrays are connected to the positive and negative voltage

references, V_{refp} and V_{refn} respectively. V_{refp} and V_{refn} are set to 1.8V and ground. At the end of a conversion cycle the charge on V_{DAC+} and V_{DAC-} will equal V_{CM}, assuming no errors in conversion.

An example of the transient settling on an ideal array can be seen in Figure 2.7. In this example V_{in+} is 0V and V_{in-} is 1.8V is sampled, followed by the opposite phase. V_{DAC+} and V_{DAC-} are represented by the solid and dashed lines respectively. Waveforms for the clock, control, and comparator are presented. Three clock cycles are used for bottom-plate sampling. During each step in the charge redistribution, the V_{DAC} voltage is scaled in half until it reaches V_{CM}.

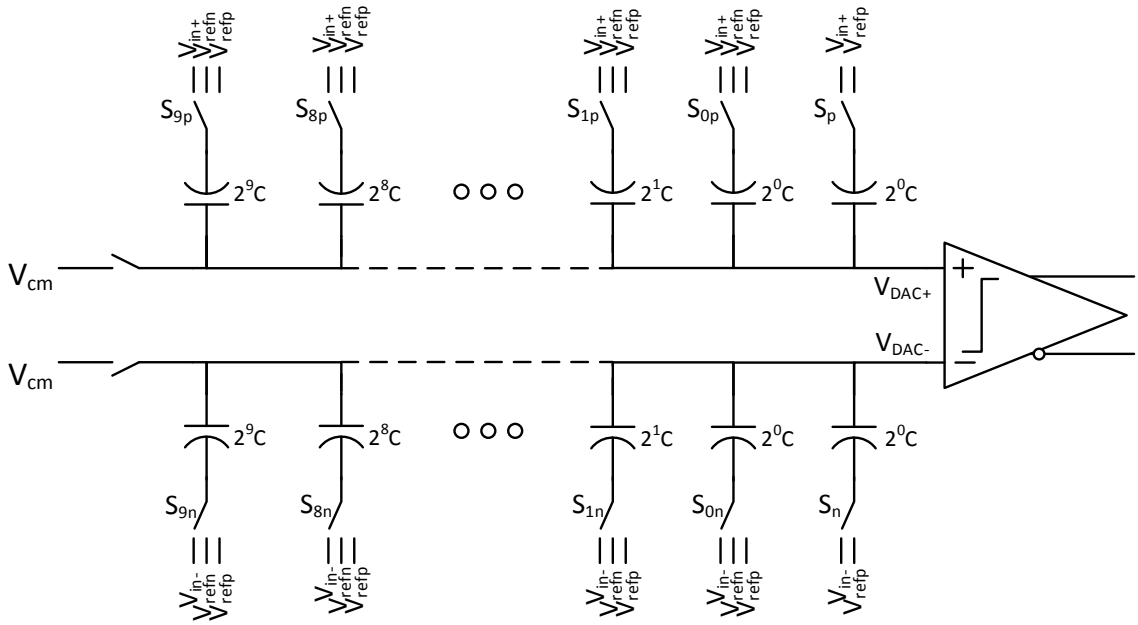

Figure 2.6: Fully Differential SAR Sampling Array and Comparator

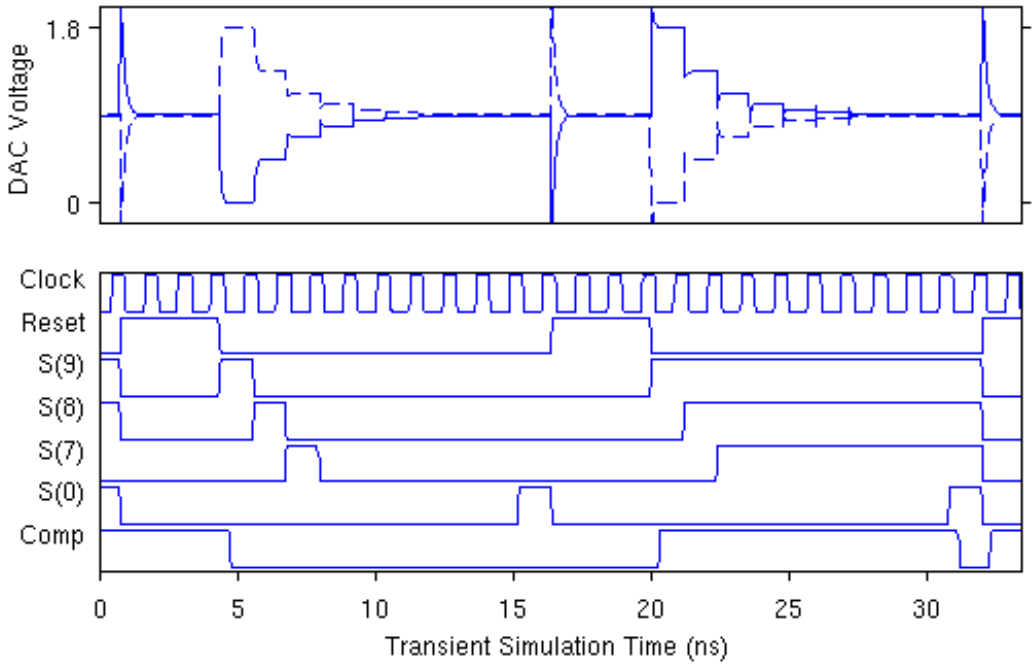

Figure 2.7: Differential Settling and Control Signals for V_{in+} = 1.8V then V_{in+} = 0V

COMPARATOR DESIGN

The comparator used in the design is shown in Figure 2.8. The design uses a dynamic pre-amplification circuit in addition to the StrongArm latch topology as discussed earlier. The pre-amplification stage consists of a dynamic preamp followed by an inverter for additional gain. The pre-amplifier is required to reduce the kickback noise and improve the input referred offset. This design is largely based off of the work in [13]. Sizing of the transistors is shown in Table 2.3. Following the comparator is an S-R latch to buffer the comparison result for the entire clock cycle, and to provide a balanced capacitive load on the comparators output.

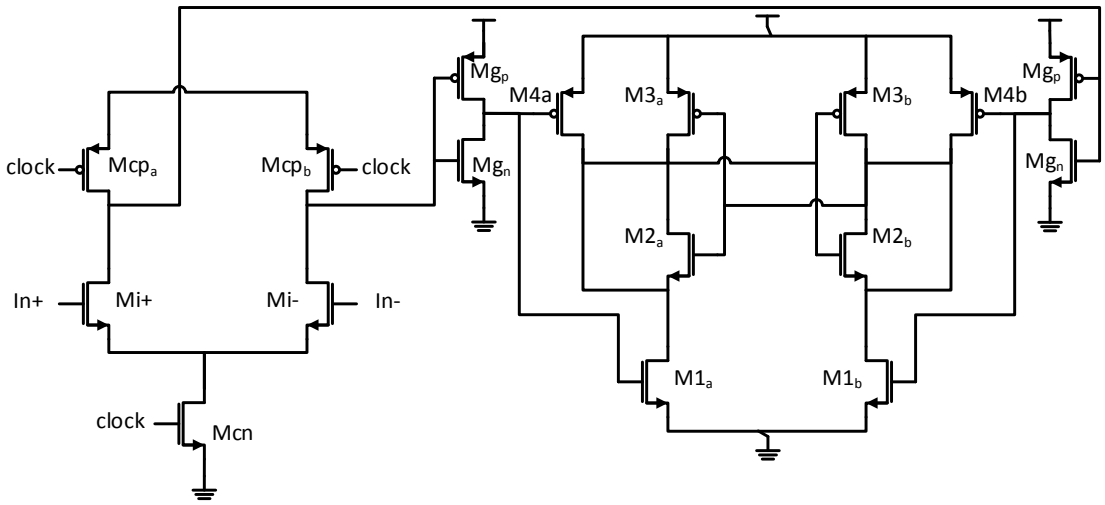

Figure 2.8: StrongArm Comparator with Dynamic Pre-amplifier

Mi	5µm	M1	2µm
Mcn	0.44µm	M2	2µm
Mcp	0.88µm	M3	4µm
Mg_p	0.40µm	M4	4µm
Mg_n	0.80µm		

Table 2.3: Comparator Transistor Sizes

CLOCK GENERATOR

The clock generator circuit, Figure 2.9, is composed of a delayed feedback of the master clock signal and passed through a pair of NAND gates. This clock provides non-overlapping signals and for generation of an early signal for bottom plate sampling. A clock signal is also used for to control the comparator so that the output is ready for the flip-flop bypass. Delay control is achieved by the inverter delay between the NAND.

Non-overlap delay is achieved by the inverter delay on the cross coupled feedback path. All clock outputs are buffered for driving the large load required by the ADC.

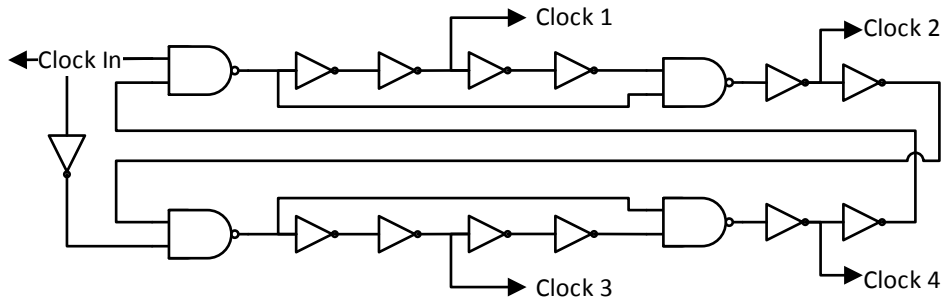

Figure 2.9: Non-Overlapping Clock Generation Circuit

Waveforms generated by the clock can be seen in Figure 2.10 for a 1.5ns reference clock. Clock 1 is used by the control logic as the early clock for bottom plate sampling of the input. Clock 2 is used by the comparator to allow time for regeneration. Clock 3 is used by the control logic to synchronize states. Clock 1 transitions to logic one 250ps early with respect to Clock 3. Clock 2 transitions to logic one about half a clock cycle early with respect to Clock 3. At a 1.5ns clock period, this allows for an offset of about 750ps.

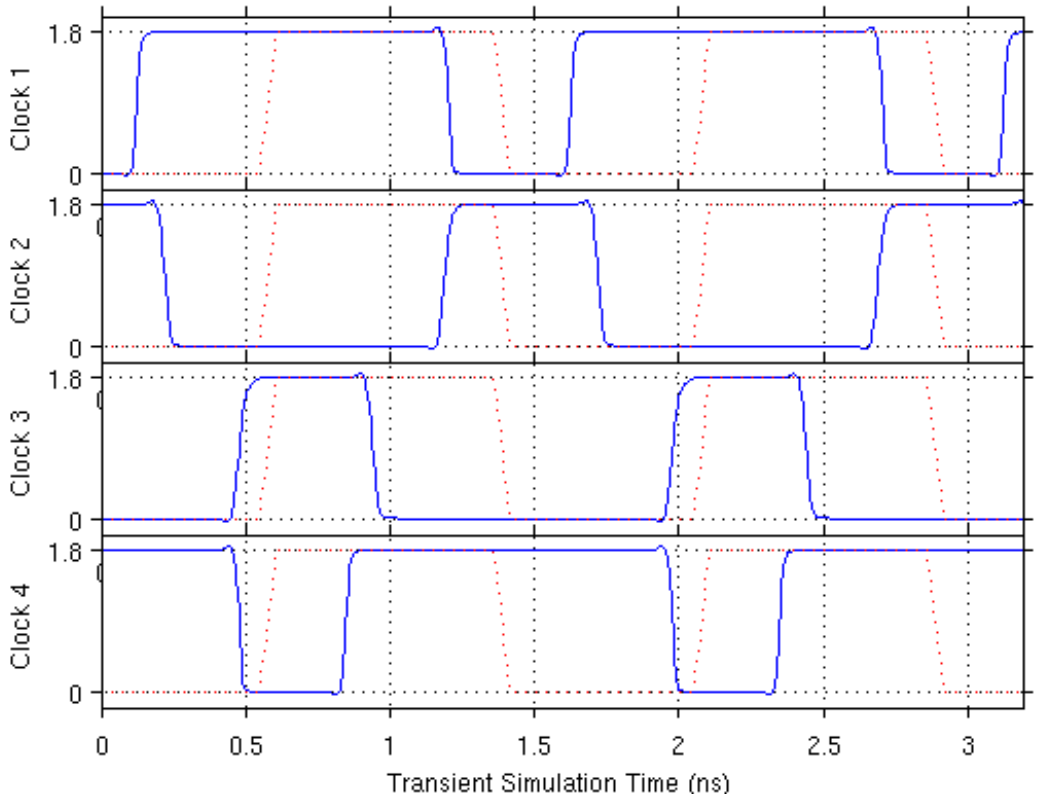

Figure 2.10: Non-Overlapping Clock Waveforms for 1.5ns Reference

TIMING CONSIDERATIONS

As explained earlier, the settling time of the DAC array is a determinate to the maximum sampling frequency. In an ideal setting, DAC settling would be the only component to sampling frequency. Additional time constraints will be set by the dynamics of the transistor level design. There are three components whose sum determines the conversion rate of the SAR ADC; DAC settling time, comparator delay, and digital control logic delay. To achieve the desired resolution, the DAC must settle to ½LSB before the comparator starts regeneration. Comparator delay sets a constraint on when the digital control logic may change states. Control logic delay includes digital logic delay and the time required to change the state of the switches. The sum of all these

delays impacts the conversion time and design effort is taken to reduce the overall conversion time.

In the conventional SAR ADC, the rising edge of the clock signal triggers the control logic to change state. In addition, the comparator result must be valid at this clock edge so that the SAR register can store the value. The delay from the rising edge to the clock through the logic to setup the switches is the logic delay. With a new state on the switch logic, the delay for the DAC to settle consumes the next portion of the clock cycle. Once the DAC has settled to the desired level, the comparator begins regeneration. The worst case comparator delay from regeneration to a valid result is the final portion of the cycle. The valid comparator result must also meet the setup and hold requirements of the SAR register.

One approach to improve the speed performance is to use and asynchronous clock to control the control logic and switching of the DAC. This clock is often based on the comparator decision to trigger the next state in the approximation. This clock can easily be generated based on a simple logic gate, such as an XOR, of the differential output signals from the comparator. The logic is to use the comparator decision to trigger the next state in the approximation. Using an asynchronous clock allows for the clock cycle to be based on the comparator delay so long as a minimum delay for logic and switching is accounted for. However, one down side is if the comparator fails to make a decision in the required time. This would effectively stall the approximation algorithm until the next input is sampled.

FLIP FLOP BYPASS

This report investigates a flip flop bypass method for controlling the SAR. This method was first reported by the work reported in [14]. The advantage compared to the

conventional SAR is to allow more time for the DAC to settle by using the result of the comparator directly on the DAC array, bypassing the flip flop. Conventional SAR logic designs allocate a worst-case half clock cycle for the comparator to regenerate, see Figure 2.11. For comparisons that can regenerate more quickly, this results in time wasted during the clock cycle[15]. Flip-flop bypass logic passes the comparator result through a multiplexer to the DAC array without a flip-flop in the logic path. The earlier the comparator regenerates, the sooner the DAC array receives its input.

For the conventional SAR in Figure 2.11, the clock triggers a bit in the shift register. This bit drives a switch on the DAC array and causes charge redistribution and V_{DAC} settling. Next, the comparator is timed such that sufficient settling has occurred and to meet the timing requirements of the SAR flip-flop register. For the flip-flop bypass SAR, the shift register also triggers a bit which drives the DAC array. However, the comparator is timed such that its result directly drives the switch as the next bit in the shift register is toggled. The clock cycle is shortened because the comparator delay and flip-flop register timing requirements are ignored with the flip-flop bypass.

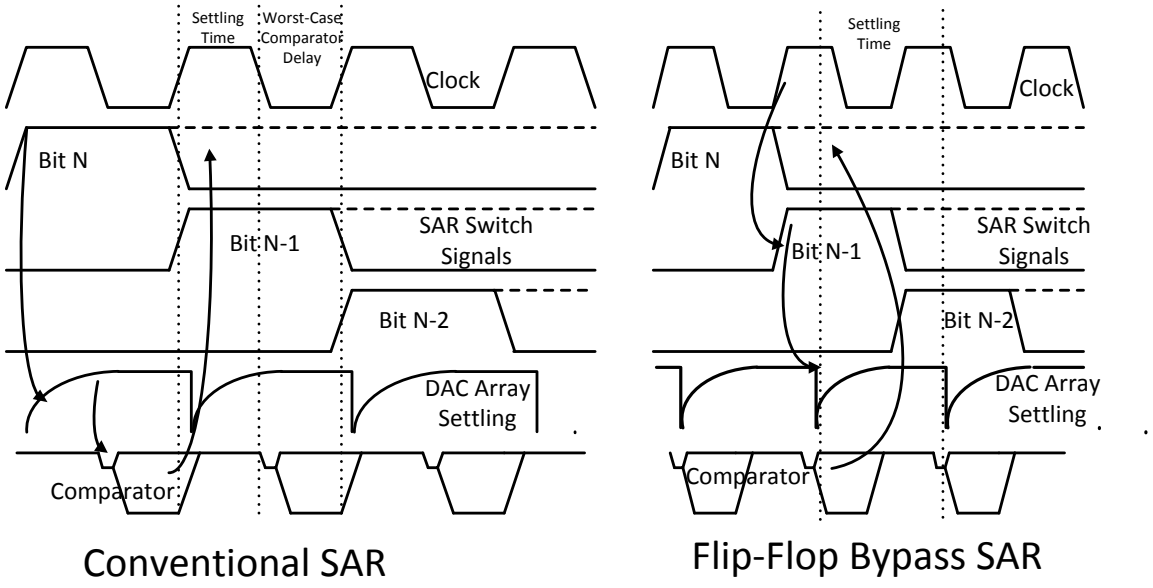

Figure 2.11: Conventional SAR Timing versus Flip-Flop SAR Timing

My implementation in Figure 2.12 utilizes some elements of the earlier research. The result of the comparator is feed into multiplexor circuit which chooses to use either the result of the shift register, comparator, or flip flop. This implements a state machine based whose state changes based on the state of the shift register. If the shift register is on the current state, the output from the shift register is selected. If the shift register is 1 ahead of the current state, the output from the comparator is selected. For all other conditions, the output from the SAR register is selected. At the end of the comparator cycle, the result is latched into a flip flop for use during subsequent approximations. The comparator is timed such that regeneration starts during at the very end of a shift register cycle, which allows for more time for settling of the DAC.

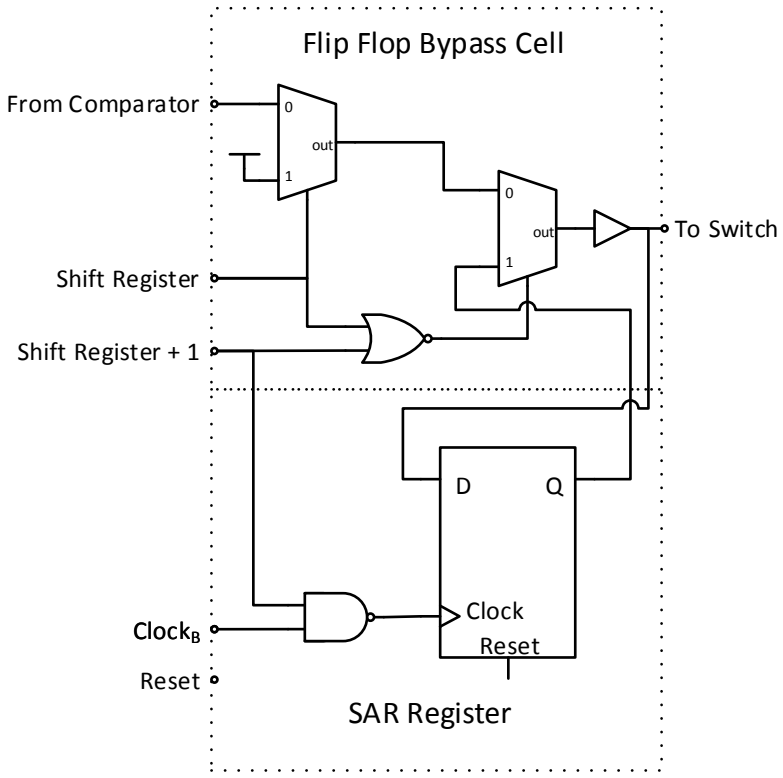

Figure 2.12: Flip Flop Bypass Control Cell

FLIP FLOP BYPASS CONTROL LOGIC

The flip flop bypass control logic uses a shift register much like the conventional design to synchronize timing of the switch logic. A diagram of the flip flop bypass control logic is presented in Figure 2.13. In the flip flop bypass cells are feed by the shift register to enable selecting between the SAR register and comparator. While the result of the comparator is driving the switch the SAR register is clocked to latch the comparator result. Reset logic to control sampling of V_{in} takes up the first four cycles of the shift register. This provides a sample and early sample signal for four clock cycles so that the input signal may settle on the capacitive arrays.

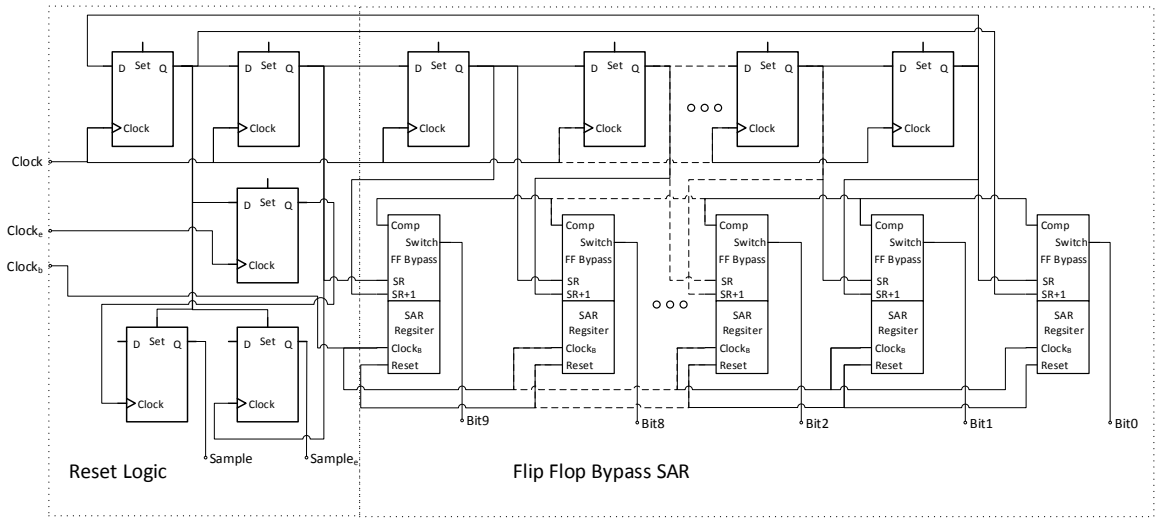

Figure 2.13: Flip-Flop Bypass Control Logic

COMPLETE DESIGN

A diagram of the complete design is presented in Figure 2.14. The SAR control logic controls the sample and charge redistribution signals for the differential sampling array. The bootstrapped circuits provide a boosted clock to the switches for V_{in+} and V_{in-}. The clock generator provides clock and early signals to the control logic for sampling and a clock signal to the comparator. The result of the comparator is stored via a SR latch and fed to the control logic. A Verilog-A model stores the conversion result from the SAR register in a data file. An ideal single to differential converter generates V_{in+} and V_{in-} waveforms. Voltage references are implemented using ideal components. Finally, an ideal maser clock is used to drive the clock generator.

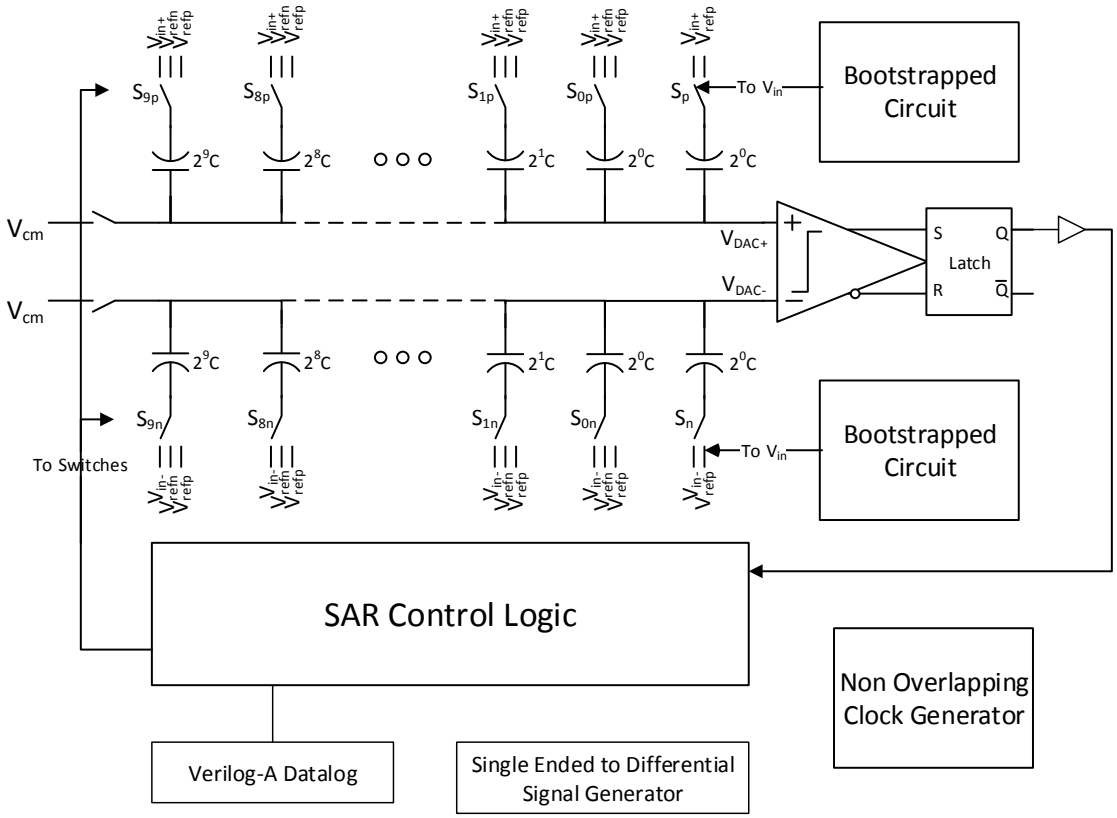

Figure 2.14: Block Diagram of Differential SAR ADC

Chapter Three: Results and Observations

COMPARATOR

Since the comparator is an essential element to both designs, it is worth discussing the key metrics of the comparator. Monte Carlo simulation allows for statistical mismatch analysis which can be used to estimate real world performance. Standard analog simulation does not take mismatch or process variation into account. Static offset resulting from mismatch and dynamic offsets resulting from parasitic capacitances will affect gain and linearity of the ADC. The test-bench described in [16] consisting of a step-wise ramp input around the common mode operating voltage was used with Monte Carlo analysis to determine the static input referred offset. Verilog-A models of step-wise ramp generator and sampler of the output for each step were created for this test bench. The result of this analysis is presented in Figure 3.1. A small degree of hysteresis is also shown in the figure. A static offset standard deviation of 4.3mV centered at 0.9V was found for this comparator design.

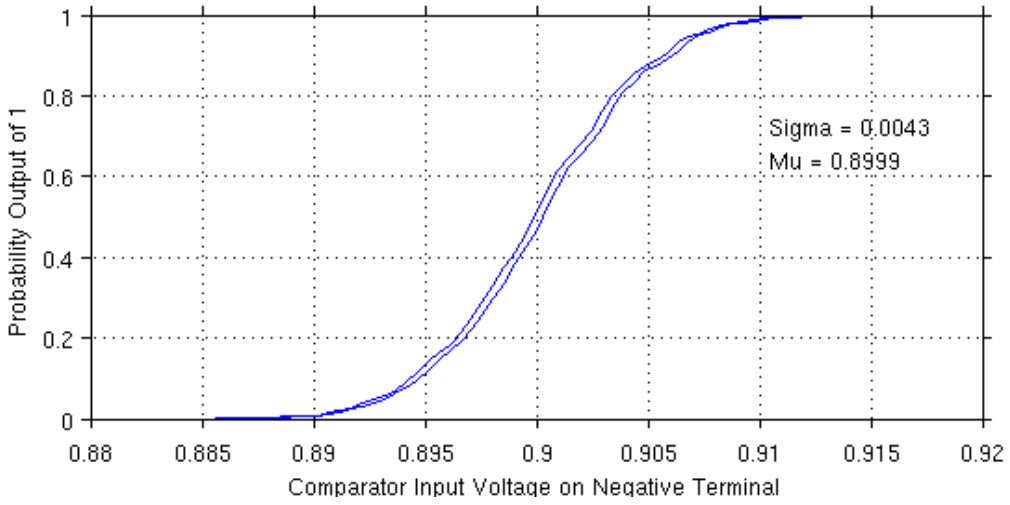

Figure 3.1: Monte Carlo Simulation of Comparator Offset (1000 Runs)

Transient analysis of the comparator in Figure 3.2 shows the delay of the latch-only comparator and pre-amplified latch, represented by the green and blue lines respectively. The sample and regenerative delays of both comparators are approximately 300ps for a ½LSB differential input. The transient waveforms also show how kickback noise is significantly reduced by the preamplifier circuit, blue versus green lines. The kickback noise waveforms show an exaggerated noise voltage on a resistor. The reduction in kickback noise was required to meet the 10-bit resolution. However, this reduction in kickback noise by the pre-amplification added a delay of 200ps before the latch circuit begins sampling. Overall a delay of 500ps is required for the pre-amplified latch. The comparator clock was timed to account for this delay and for the delay from the flip-flop bypass logic. Power consumption of the comparator for a ½LSB input was 0.26mW at a 1.4ns clock.

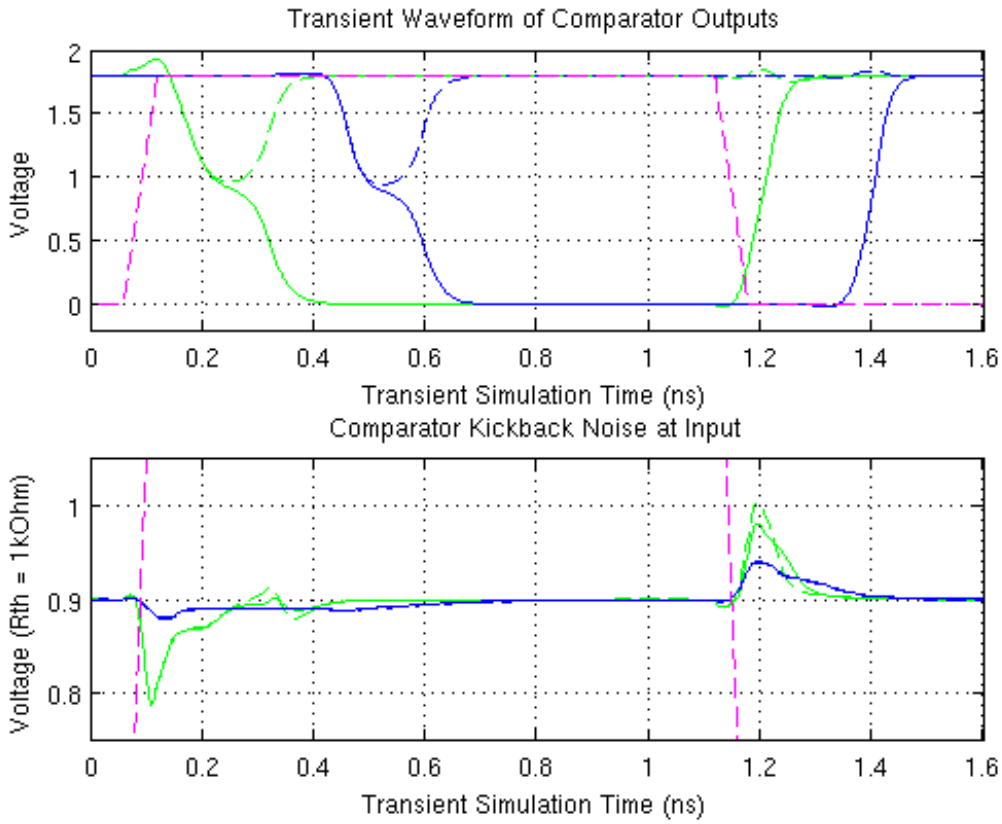

Figure 3.2: Transient Simulations of Comparator

CONVENTIONAL SAR

The control logic provides four clock cycles for the DAC to reset and settle. Sampling was achieved at a maximum clock period of 1.4ns, which equates to the 51MS/s sampling frequency. Average power consumption or the conventional SAR is measured at 54.9µW. The 512-point FFT at this sampling frequency can be seen in Figure 3.3. Simulated without noise, this design has an SNDR or more accurately, an SQNR of 60.3dB and ENOB of 9.72. The simulation input signal is 31/512 of the Nyquist-rate, or a frequency of 3.27MHz. The FFT figures were generated in Matlab using a modified script provided in [17]. Transient Waveforms are depicted in Figure 3.4.

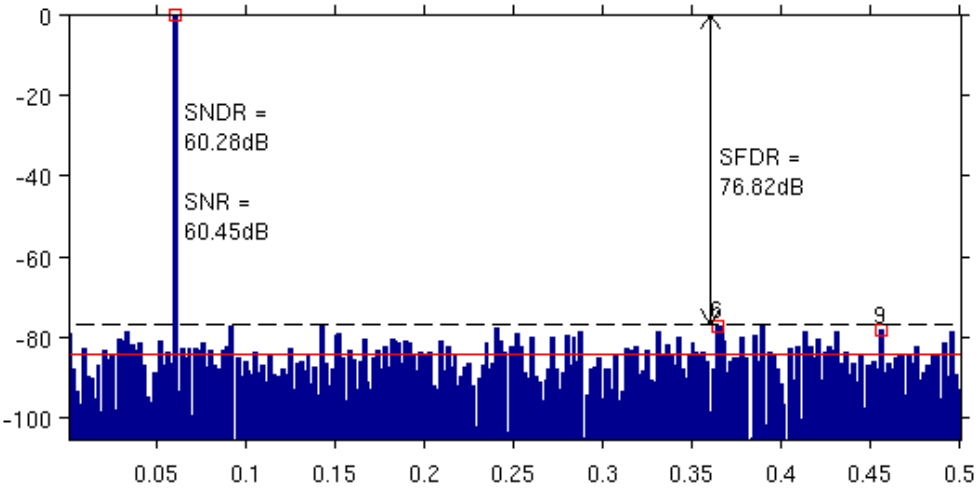

Figure 3.3: 512-point FFT of Conventional Logic SAR at F_{samp} = 54MS/s, bin 31

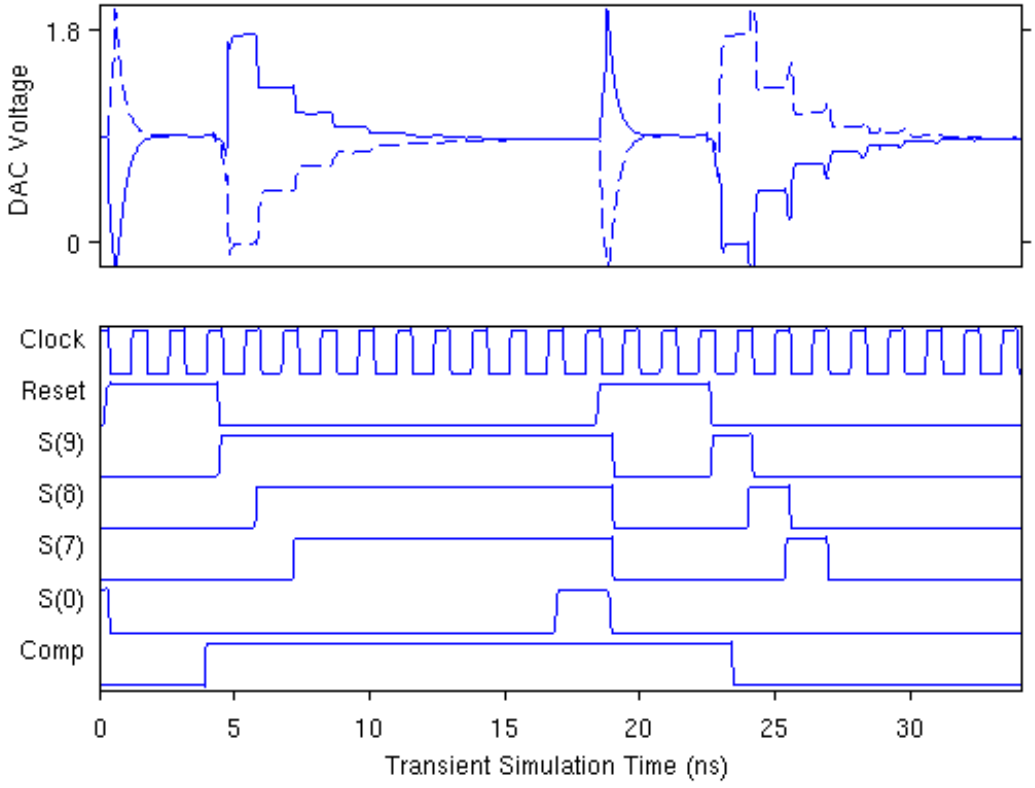

Figure 3.4: Conventional SAR Waveforms at 1.4ns Clock for V_{in+} at 1.8V and 0V

FLIP-FLOP BYPASS SAR

The FFT of the flip-flop bypass SAR, Figure 3.5, was simulated at a reference clock period of 1.2ns. The simulation input signal is 31/512 of the Nyquist-rate, or a frequency of 3.87MHz. Simulations beyond this clock frequency caused degradation in SNDR. Average power consumption or the flip-flop bypass SAR is measured at 62.1μW. Timing diagrams and DAC settling waveforms can be seen in Figure 3.6 for a 1.8V and 0V input. Spikes are visible in the DAC waveforms when the comparator signal is low. This undesired effect is due to the timing of the flip-flop bypass and comparator became more pronounced as clock frequency increases.

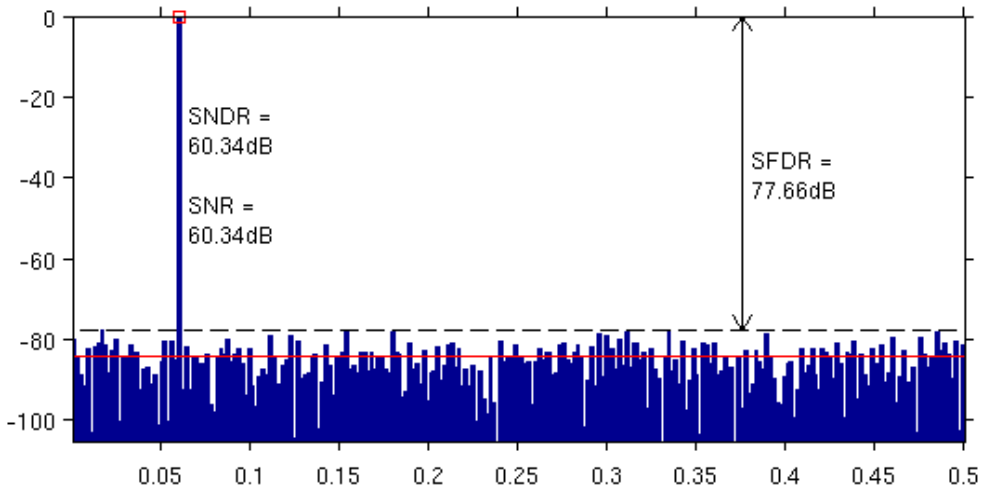

Figure 3.5: 512-point FFT of Flip-Flop Bypass SAR at F_{samp} = 64MS/s, bin 31

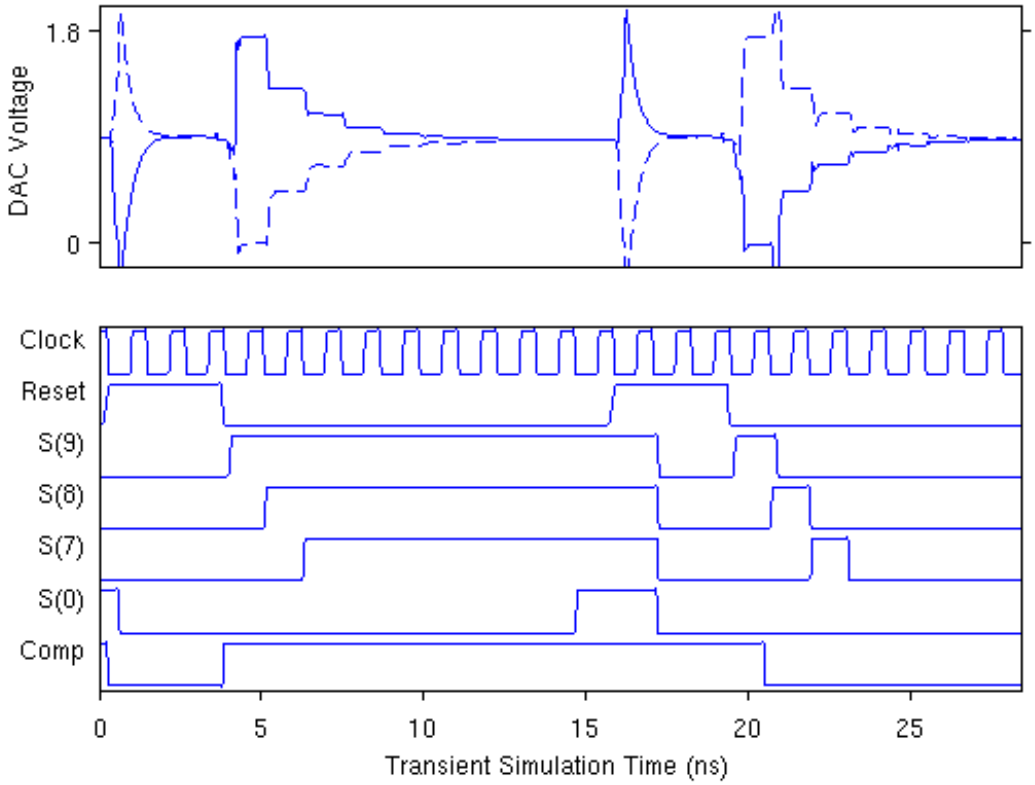

Figure 3.6: Flip-Flop Bypass Waveforms at 1.2ns Clock for V_{in+} at 1.8V and 0V

Chapter Four: Conclusions

Summary and Learnings

The flip-flop bypass design sought to increase the conversion frequency beyond the conventional design. The conventional design began to show degraded SQNR beyond a sampling frequency of 54MS/s. The flip-flop bypass design achieved a sampling frequency of 64MS/s. The increased power consumption is largely due the additional flip-flop bypass cells. Both designs were limited by the settling time of the sampling network to track the input signal. Alternatively, an additional clock cycle could be added to the sampling phase such that the overall conversion time, sampling plus charge redistribution, could be decreased. This settling time could have been improved at the expense of larger switches with increased power consumption.

Nonetheless, those optimizations would not improve on delays in the control logic signals. These delays are fundamentally what challenged the flip-flop bypass design. The finite logic transition delays do not scale as the clock period is decreased. The delay from the comparator input through the flip-flop bypass cell and to the switches, about 350ps, becomes consumes a significant portion of the clock cycle as the clock period decreases. This delay requires the comparator to make a decision earlier in the clock cycle; time that could be used for DAC settling. Figure 4.1 shows flip-flop bypass in blue and conventional in red using a 1ns clock period. Due to insufficient settling time, there are errors in the conversion. The flip-flop bypass design, in blue, is poorly timed at this frequency and spikes are seen. These spikes are due to the delay in the path from the comparator to switches.

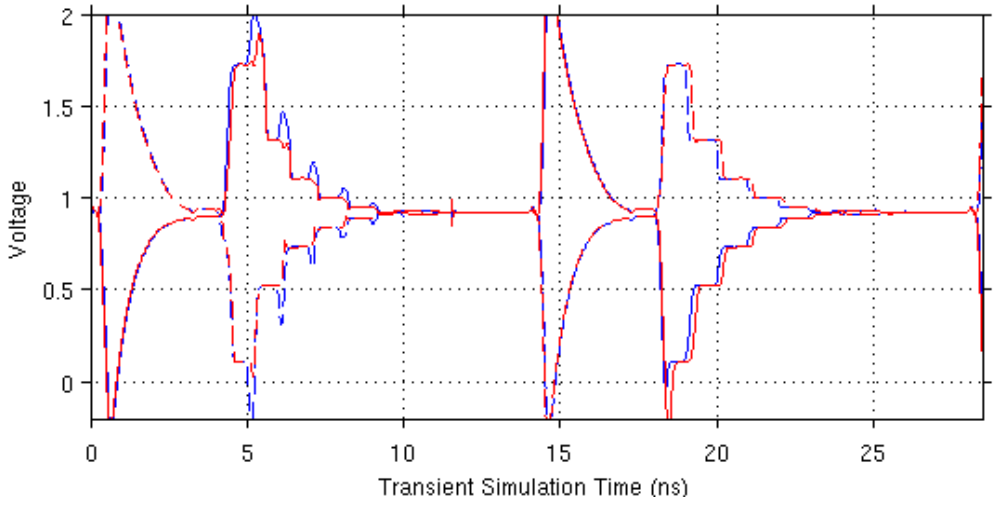

Figure 4.1: Transient Waveforms of V_{DAC} Settling for Both Designs at 71MS/s

Improvements to the flip-flop bypass design could alleviate some of these concerns. Optimizing the comparator delay is one area for improvement. The pre-amplifier used in the dynamic comparator design consumes more than half of the 500ps decision delay. The addition of an S-R latch to buffer the comparator output caused another 100ps of delay. Redesign of the flip-flop bypass cell to use a direct output from the comparator, rather than the buffered latch, could have virtually eliminated the delay. However, using the output directly would require careful synchronization in selecting between the comparator output and SAR register as well as latching the comparator result into the SAR register. Optimizing the flip-flop bypass cell design is where the greatest reduction in conversion time can be achieved.

CONCLUSION

The design and simulation of a conventional and flip-flop bypass SAR ADC has been presented. For this design to be implemented in silicon requires design optimization

and rigorous simulation across process variables and temperature. Additional circuitry for calibration to minimize parasitic effects may also be required for a real design.

The author has learned much about mixed-signal design from this project and is pleased with the results. However, this design has only been tested at a simulation level. Noise analysis has not been performed, and this will significantly degrade the SNDR and effective resolution. More design and optimization are required for the simulated results to become a reality.

Bibliography

[1] J. L. McCreary and P. R. Gray, "All-MOS charge redistribution analog-to-digital conversion techniques. I," *Solid-State Circuits IEEE J.*, vol. 10, no. 6, pp. 371–379, 1975.

[2] B. Murmann, "ADC Performance Survey 1997-2013," [Online]. Available: http://www.stanford.edu/~murmann/adcsurvey.html.

[3] T. C. Carusone, *Analog integrated circuit design*. Hoboken, NJ: Wiley, 2012.

[4] F., Maloberti, Franco, Maloberti, Franco Maloberti, *Data converters*. Dordrecht: Springer, 2007.

[5] J. Guerber, H. Venkatram, M. Gande, A. Waters, and U.-K. Moon, "A 10-b Ternary SAR ADC With Quantization Time Information Utilization," *IEEE J. Solid-State Circuits*, vol. 47, no. 11, pp. 2604–2613, Nov. 2012.

[6] F. Yuan, *CMOS circuits for passive wireless microsystems*. New York: Springer, 2011.

[7] T. Kobayashi, K. Nogami, T. Shirotori, and Y. Fujimoto, "A current-controlled latch sense amplifier and a static power-saving input buffer for low-power architecture," *IEICE Trans. Electron.*, vol. 76, no. 5, pp. 863–867, 1993.

[8] J. Montanaro, R. T. Witek, K. Anne, A. J. Black, E. M. Cooper, D. W. Dobberpuhl, P. M. Donahue, J. Eno, W. Hoeppner, and D. Kruckemyer, "A 160-mhz, 32-b, 0.5-w CMOS RISC microprocessor," *Solid-State Circuits IEEE J.*, vol. 31, no. 11, pp. 1703–1714, 1996.

[9] K. Uyttenhove and M. S. J. Steyaert, "Speed-power-accuracy tradeoff in high-speed CMOS ADCs," *IEEE Trans. Circuits Syst. II Analog Digit. Signal Process.*, vol. 49, no. 4, pp. 280–287, 2002.

[10] A. Nikoozadeh and B. Murmann, "An Analysis of Latch Comparator Offset Due to Load Capacitor Mismatch," *IEEE Trans. Circuits Syst. II Express Briefs*, vol. 53, no. 12, pp. 1398–1402, Dec. 2006.

[11] P. M. Figueiredo and J. C. Vital, "Kickback noise reduction techniques for CMOS latched comparators," *IEEE Trans. Circuits Syst. II Express Briefs*, vol. 53, no. 7, pp. 541–545, Jul. 2006.

[12] 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, 1999.

[13] H. Jeon, Y.-B. Kim, and M. Choi, "Offset voltage analysis of dynamic latched comparator," in *2011 IEEE 54th International Midwest Symposium on Circuits and Systems (MWSCAS)*, 2011, pp. 1–4.

[14] Z. Cao, S. Yan, and Y. Li, "A 32 mW 1.25 GS/s 6b 2b/Step SAR ADC in 0.13µm CMOS," *IEEE J. Solid-State Circuits*, vol. 44, no. 3, pp. 862–873, Mar. 2009.

[15] *Low-Power High-Speed ADCs for Nanometer CMOS Integration*. .

[16] A. Graupner, "A Methodology for the Offset-Simulation of Comparators," *Des. Guide Community Www Des.-Guide Org*, 2006.

[17] Skyler Weaver, "Pretty FFT: prettyFFT is a MATLAB function that plots coherently sampled data and calculates ENOB, SNDR, SNR, and SFDR." .