An 8-bit Charge Redistribution SAR ADC in 45nm CMOS

John Warwar, *MSEE, USC Department of Electrical Engineering*

***Abstract*—An 8-bit charge redistribution SAR ADC is proposed as a building block for a 5GS/s application. The functional logic is synchronous and designed for straightforward time interleaving.**

I. INTRODUCTION

SAR ADCs offer many desirable characteristics such as low power, high resolution, and decent conversion rates with proper implementation and techniques. In current technology trends, SAR ADCs are being used extensively in applications that require low power and high resolutions [2]. There are several SAR architectures that all have advantages for specific applications. In this design, a binary capacitive array is implemented with a standard capacitive scaling scheme. The design goal of this project was to implement an 8-bit SAR ADC at 5Gs/s, with a low FOM. Given the fast sample rate and the relatively high precision, in order to fulfill the speed requirement, an 8-way time interleaving arrangement will be used.

II. ARCHITECTURE

Figure 1 below shows the basic 8-bit ADC topology that is being proposed. The S2 switch is part of the track-and-hold function while the pre-driver and the SAR logic are not shown. By using a differential architecture, even order harmonics are cancelled which greatly reduces the distortion. In figure 1 it can be seen that there are two switch phases of the capacitive DAC: a sampling phase and holding phase, denoted by S1 and S2, respectively. In the sampling phase, the input signal is sampled onto the bottom plate of each capacitor, and the common mode is sampled onto the top plate. It should be noted that—although not denoted in figure 1—switches sampling the common mode and the input signal, while both S1, are slightly offset in time from each other. In reality, the common mode is sampled onto the top plates first, and then at small delay in time later, the input signal is sampled onto the bottom plates.

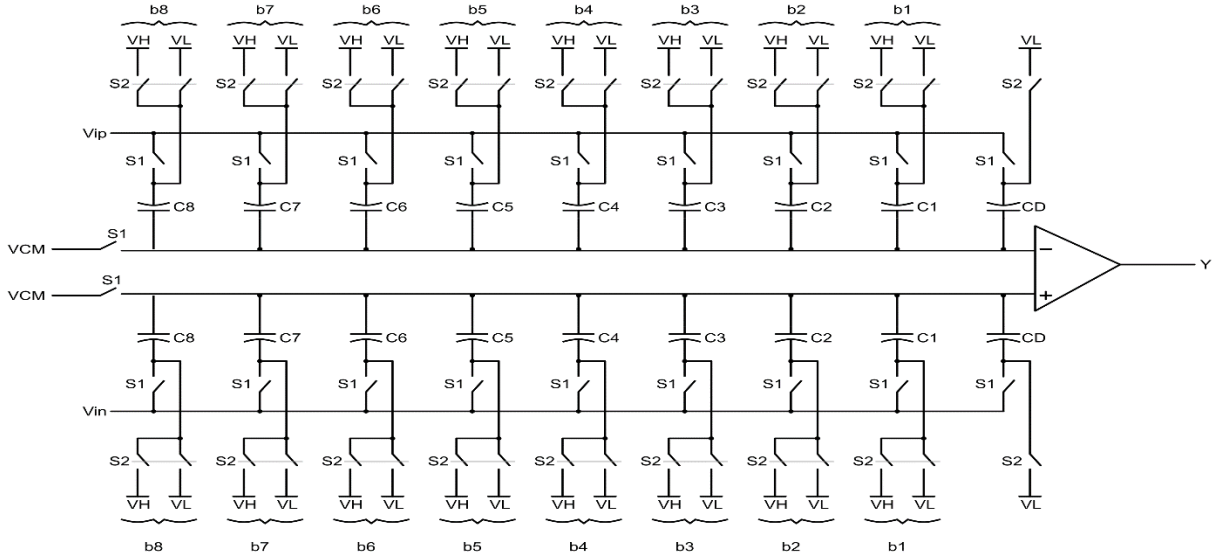

Figure 1. 8-bit SAR ADC Capacitive Array and Switching Scheme

III. Calculations of Specifications

As a worst-case estimate for the accuracy of the capacitors in the binary weighted array, I assumed that each capacitor would need to be consistent with an accuracy of $50dB \simeq \frac{1}{316} = 0.00316 = 0.32\%$. The matching property of the provided technology for the capacitor is $A_C = 0.3\%\sqrt{fF}$ and $\sigma_C = \frac{A_C}{\sqrt{C}}$. Rearranging for C, $C = \left(\frac{A_C}{\sigma_C}\right)^2 = \frac{\left(0.3\%\sqrt{fF}\right)^2}{\left(0.3\%\right)^2} \rightarrow C > 1fF$. This calculation gave a rough estimate for the value of the unit capacitor size in the binary weighted capacitive array $C_u = 1fF$, which thus gave $C_{Total} = 256C_u = 256\,fF$. After building the behavioral model and modeling the effects of inaccuracies in the capacitors, I found this estimation to be overly conservative, since the accuracy of the larger and smaller capacitors essentially average out, making the system very immune to the capacitor matching.

For the input of the capacitive DAC, $C_{in} \simeq 256\,fF$ and $V_{ref} = 0.5$. As another worst-case estimate, for the power consumption

$$P_{DAC} \simeq CV^2 f = C_{in}V_{ref}{}^2(5GHz) = \left(256\,fF\right)\left(0.5\right)^2(5GHz) = 320\times10^{-6}\frac{J}{s}$$

and $P_{Comparator} \simeq 10 \times P_{DAC} = 320\times10^{-5}\frac{J}{s}$.

Therefore, $P_{Total} \simeq 3.52\times10^{-3}\frac{J}{s}$. Since an 8-way time interleaving will be implemented to meet the speed requirements, the effective sample frequency for each individual ADC will be $f_s = 625MHz$. Assuming an $ENOB = 8$, we can make a rough estimate for the figure of merit, which does not include the impact of the logic circuitry or the source follower which is used as a driver for the track-and-hold.

$$FOM = \frac{P_{Total}}{\left(2^{ENOB}\right)\left(f_s\right)} = \frac{3.52\times10^{-3}\frac{J}{s}}{\left(2^8\right)\left(625\times10^6\right)} = 2.2\times10^{-14} \simeq 22\frac{fJ}{\text{conv-step}}.$$

For an $SNR > 50\ dB$, we need $10\log A = 50$ or $A = 1\times10^5$.

This implies that $\frac{KT}{C} < \frac{\frac{\left(V_{\max}\right)^2}{2}}{A}$ or $C > \frac{(KT)\left(1\times10^5\right)(2)}{\left(V_{\max}\right)^2}$

This estimate yields $C > 13.25\,fF = 256C_u$, which is well within our previously calculated constraint for C_u

For an ADC, $SQNR = 6.02N + 1.76$, which is the theoretical maximum signal to quantization noise power for a given number of bits. The $SNDR = 6.02(ENOB) + 1.76$, so if we assume $ENOB \simeq 8$, then $SNDR = 6.02(8) + 1.76 = 49.92$. In our 8-bit system, $1\,\text{LSB} = \frac{0.5}{256} = 0.00195$.

Additionally, we need $DNL < \frac{1}{2}LSB$ and $INL < 1\,LSB$

Since we want $SNR > 50\ dB$, we need

$$V_n{}^2 < \frac{\frac{(0.25)^2}{2}}{1\times10^5} = 312nV^2.$$

For the settling requirements, the ADC should settle to 8 bits within half of a clock cycle, or 100ps, while in unity gain feedback. It should also settle within 100ps while in the comparison mode. The offset voltage is less of a concern, as it can be calibrated out as a DC offset. The SAR logic must operate at a maximum clock rate of 5 GHz, with a power supply voltage of 1.1V.

Table 1: Desired ADC Specifications

	min	max	units
Track and Hold			
Integrated Noise (KT/C) (1 Hz - 100GHz)	-	312	nV^2
THD	-	-50	dB
DAC			
Input Cap	256*Cu	-	F
Integrated Noise (KT/C) (1 Hz - 100GHz)	-	312	nV^2
Comparator			
Settling time (unity gain feedback)	-	100	ps
Settling time (comparison mode)	-	100	ps
Offset Voltage	N/A	N/A	V
Integrated Noise (KT/C) (1 Hz - 100GHz)	-	312	nV^2
SAR Logic			
CLK Rate	-	5	GHz
Conversion Rate	-	625	MHz
Complete ADC			
VDD	-	1.1	V
SNDR	49.92	-	dB
DNL	-	0.5	LSB
INL	-	1	LSB

IV. Design Strategy

To achieve an optimal figure of merit (FOM), there are a few characteristics of the ADC that are especially important. Namely, a good FOM requires a combination of low power, a high ENOB, and a fast-sampling frequency. To reduce the non-ideal effects associated with this architecture and improve performance, a few techniques can be implemented. One technique that is used in this design is bottom plate sampling, as alluded to previously. This reduces the parasitic capacitances at the input terminal of the comparator [1]. Moreover, the use of bottom plate sampling eliminates the need for a more complicated clock bootstrapping scheme. This is because the bottom plate sampling isolates the charge injection to the top plate, which is sampling the common mode. This removes the signal dependency. Additionally, lowering the power consumption as much as possible, particularly in the DAC will also improve the FOM. Throughout the design, each circuit block was designed individually and tested for functionality, and then the entire ADC was assembled using the smaller sub-circuits. For example, when building the comparator block, 5 versions of a comparator were designed and tested, which will be discussed in a later section. Designing hierarchically definitely takes more time up front, but in the end, it saves a lot of time as changes can be made very efficiently.

V. Timing

The timing diagram seen in figure 2 details what happens on each successive clock cycle throughout the SAR operation. Given that this is an 8-bit ADC and that it takes one clock cycle for each bit, it takes a total of 8 clock cycles for one conversion. On the rising edge of the first clock, the input signal is sampled. In each successive clock cycle, starting from the MSB and ending at the LSB, each bit is set high, compared with the output of the comparator, and then set to its correct value which is held for the remainder of the eight clock cycles. To ensure that all of these clock cycles were properly timed and skewed in relation to each other, long chains of inverters were created to generate these delays, which will be discussed in a later section.

VI. Comparator

The comparator is one of the most important pieces of a SAR ADC, and its functionality is critical for correct SAR operation. During the design process, 5 different comparator topologies were designed and tested. These comparators can be seen in figure 3. Each comparator was tested for speed, correct operation and functionality, stability, kickback, and offset. All of these comparators were relatively similar in regard to many of these specifications, but the comparator seen in figure 4 was ultimately chosen for this application. When the other comparators started to show problems when running at higher speeds, this one seemed to perform the most optimally.

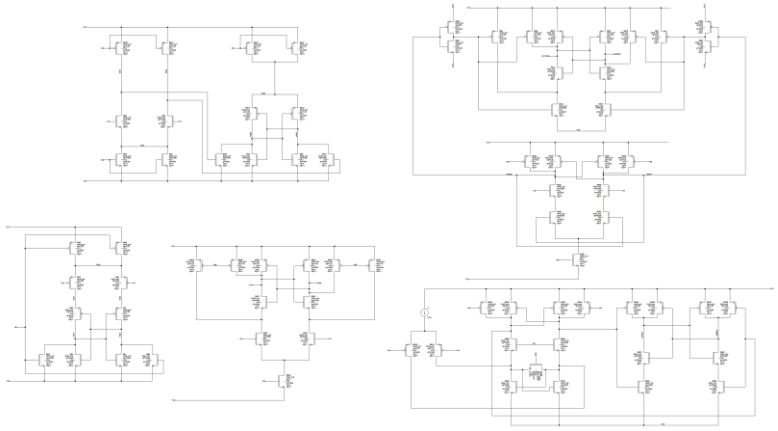

Figure 3: Various Comparators Designed and Tested

The comparator shown in figure 4 which was used in this ADC is a relatively simple topology. It consists of an input gain stage—or a preamplifier—followed by

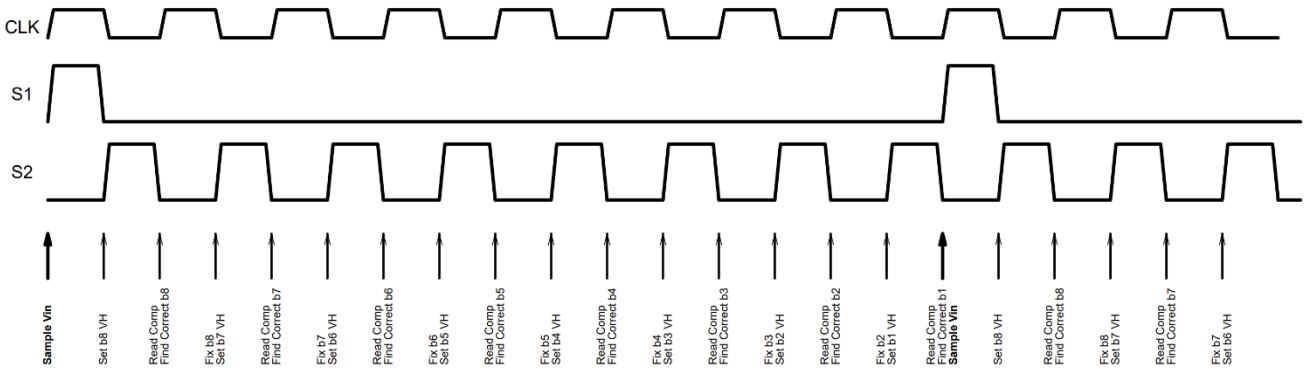

Figure 2: Timing Diagram for Switch Operation

an output latch stage. All transistors in this comparator were sized with minimum lengths and with the PFET widths sized about 2.5 times that of the NFETS.

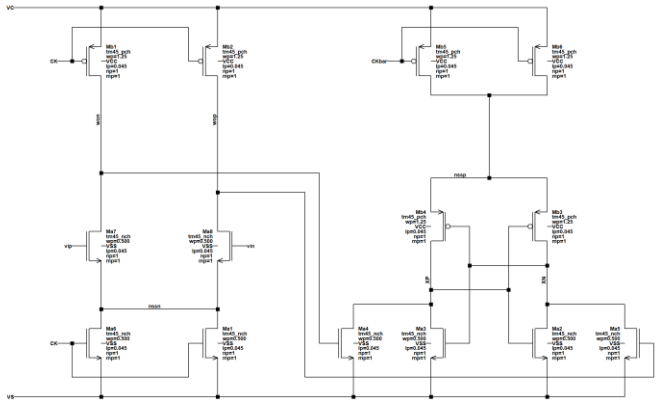

Figure 4: Chosen Comparator

This comparator was a relatively simple design which was inspired by a paper from IEEE 2007 [5]. The only modification of the comparator used in this project and the one from [5] was that the two tail devices were split into two devices at half the width.

VII. Three-bit counter

In order to generate the pulses that drive the digital logic, a 3-bit counter was designed that can be seen in figure 5. This is a synchronous counter that is fed into a shift register. Each register output is tapped off and used to drive the digital logic blocks. Since the outputs of the flip-flops are tied directly to a feedback path, inverters were added to avoid loading this path. In figure 5, a long chain of inverters is also shown. The purpose of this inverter chain is to generate clocks with proper delays.

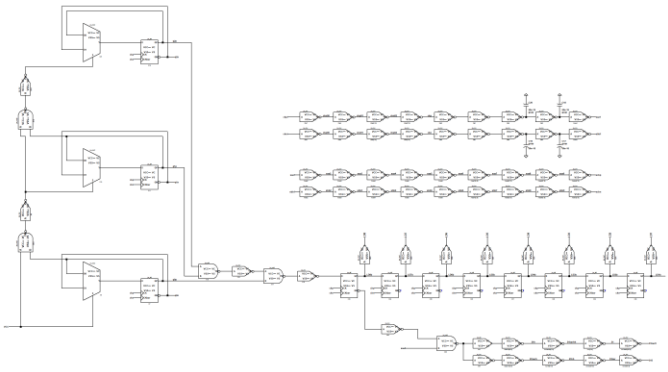

Figure 5: 3-bit Counter with Shift Register

Throughout the ADC design, various clocks are needed with small skews relative to each other and using delays of inverters were utilized for this purpose. Capacitors were also added on certain nodes to make this delay greater. Other than the generation of delays, these inverters also serve the purpose of buffering the clock signal, which is input to the system essentially as a triangle wave. The negative output of each flip-flop in the shift register is ANDed with the system clock to generate the pulses used for the switches in the capacitive array.

VIII. Digital Logic

Designing a functional digital logic system probably required the largest amount of time during the design process. Figure 6 shows the 8 digital logic blocks, and their respective inputs and outputs. Each block receives the comparator output (positive terminal), as well as the current and previous load bits generated by the shift register after the 3-bit counter. All blocks also receive the MSB load bit, LD8, to know when to reset once a conversion is completed. The blocks set and then return the digital bits (b8-b1) back to the capacitive array, and also output the digital code (Q8-Q1).

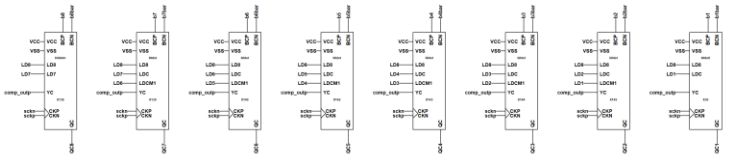

Figure 6: Digital Logic Blocks

Below in figure 7, the inside of one of the logic blocks is shown. It should be noted that all of the digital logic blocks are internally the same, except the blocks used for the MSB and LSB. The MSB block only needs LD8 and LD7 as inputs whereas the successive blocks need LD8 plus the current and successive LD bits. The LSB only requires LD8 and LD1 as load bits, because it is the last bit in the conversion. Therefore, the MSB and LSB blocks only require two multiplexers instead of three. Once the logic was determined and its functionality was tested, significant time had to be spent to buffer many of these signals, especially the ones driving the larger capacitors.

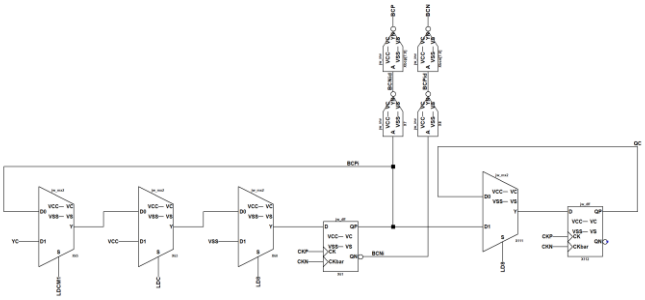

Figure 7: Example of Digital Logic Block

IX. Sizing and Buffering

A primary challenge throughout the design process was using appropriate sizing and buffering to achieve proper signal throughput. Particularly in the capacitive array, loading became a significant issue when trying to drive the larger capacitors, and the switches had to scaled up accordingly. In figure 8 below, the capacitive array can be seen in its true implementation. At the input to the switches where the digital bits b8-b1 will return from the digital logic blocks, inverters were added to help with the effects of loading. It should also be mentioned that at the output of the digital logic blocks, these signals were already buffered through several stages of inverters to further improve in this regard. In the capacitive array of figure 8, it can be seen that the switches driving the larger capacitors have been proportionally scaled up with respect to the capacitors to effectively lower their resistance. This helped in reducing the settling time, because for fast settling, a low switch resistance is desired, i.e., a large switch. Inverters were also added at the output of the comparator to be able to drive the digital logic.

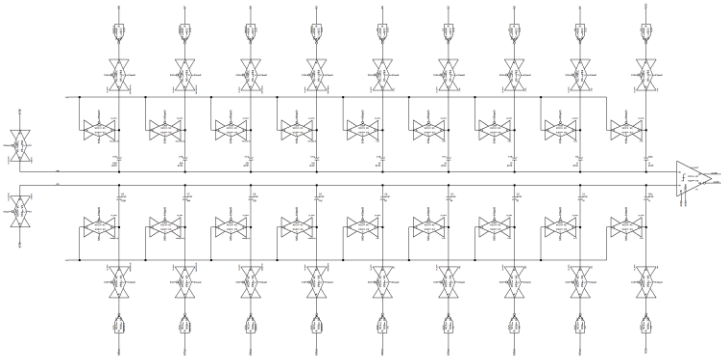

Figure 8: Capacitive Array and Switches with Buffering

X. 8 way Time Interleave

After the single ADC was tested and functioned properly, 8 ADC blocks were stacked for an 8-way time interleaving scheme. Much of the logic that was used to generate the pulses for the switches was shared between each of these individual ADC's and was all put into one block that drove each of the 8 blocks. In figure 9, this scheme is shown. Each ADC receives the same 8 load signals, LD8-LD1, but on each successive block the inputs are skewed by one bit, creating a stair-step like pattern. By skewing these bits in this way, each ADC is only responsible for 1 bit per 8 cycles, as opposed to 8, so the effective speed-up of this scheme is 8 times. The 8 ADC's do not share common switching signals, so four switch signals are sent to each ADC.

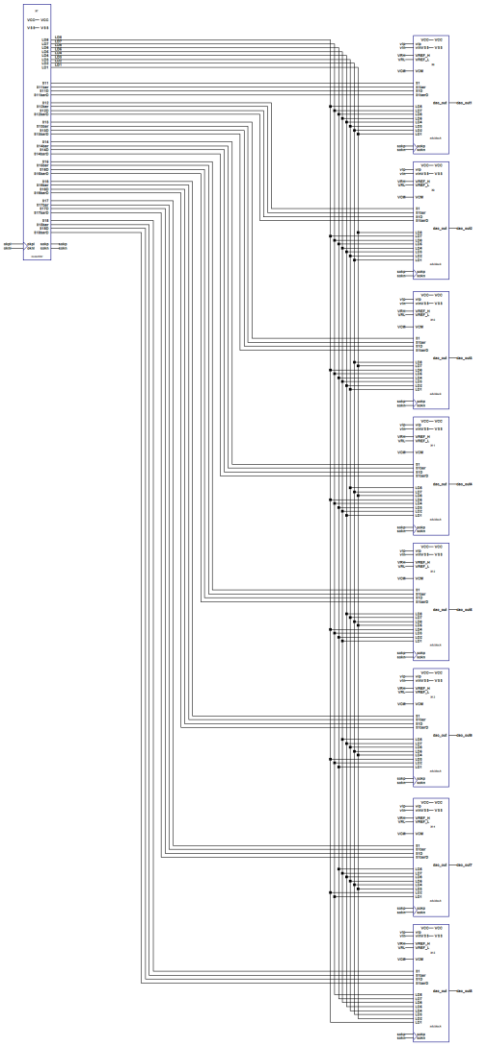

Figure 9: 8-Way Time Interleave

XI. Challenges

This project introduced a seemingly never-ending series of challenges and difficulties, some of which were remedied and some which could not be. I found that the signal loading non-idealities were very significant and required huge amounts of buffering for the fanout of those signals. Determining the timing and required skew of all the various clocks and signals so that operation was correct also took lots of time. I also had a hard time finding helpful documentation anywhere about the digital logic and spent a large amount of time through trial and error determining the correct logic. Additionally, I was presented with the problem of lengthy simulations which slowed down the design greatly, as small changes could not be quickly tested in an iterative way. One challenge that I was unable to overcome was meeting the speed requirements of the desired ADC. The objective was to build a 5 GS/s converter, and even with 8-way time interleaving, I was only able to reach a clock speed of 2 GHz. This was likely due to the critical path of some

of the logic blocks was simply too long, because above about 2 GHz, the ADC no longer functioned properly. For 5 GHz to be reached further time interleaving of more than 8 could be implemented.

XII. Results

It should be noted that on the simulation at the sampling frequency, which is 250 MHz, the signal is aliased at 7.63 MHz. In the spectrum for the sampling frequency, the fundamental tone can be seen at this frequency.

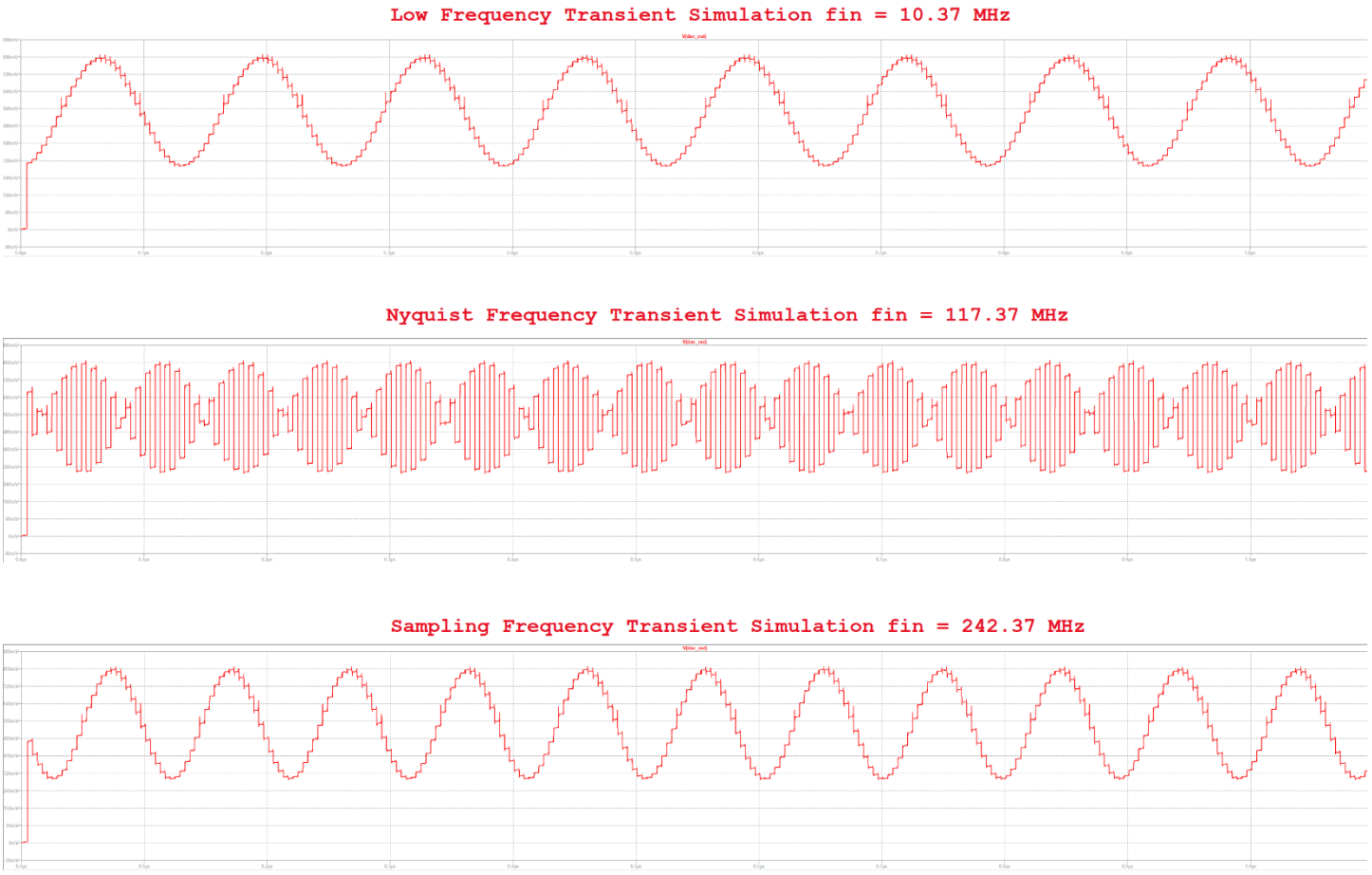

Figure 10: Transient Simulation at Various Frequencies

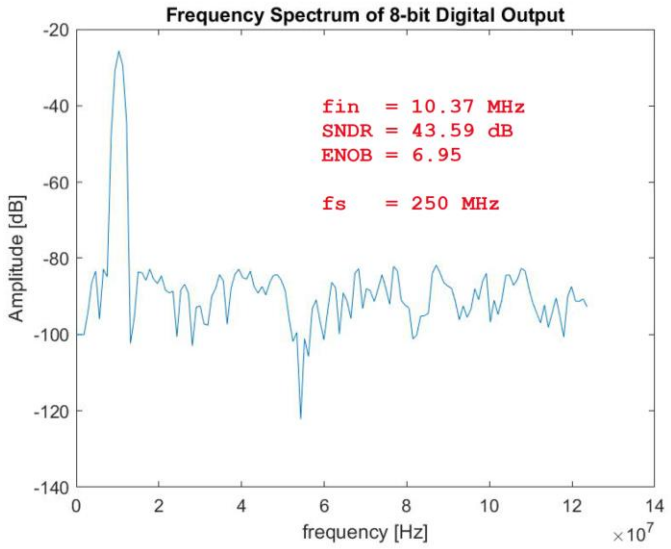

Figure 11: FFT on ADC Output (Low Frequency)

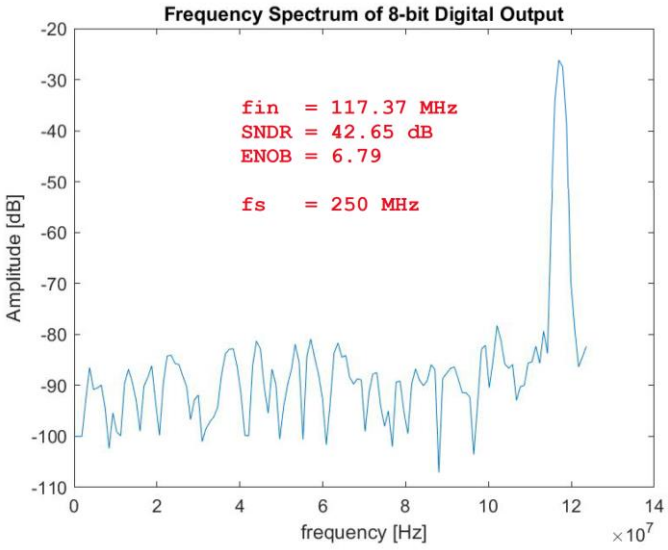

Figure 12: FFT on ADC Output (Nyquist Frequency)

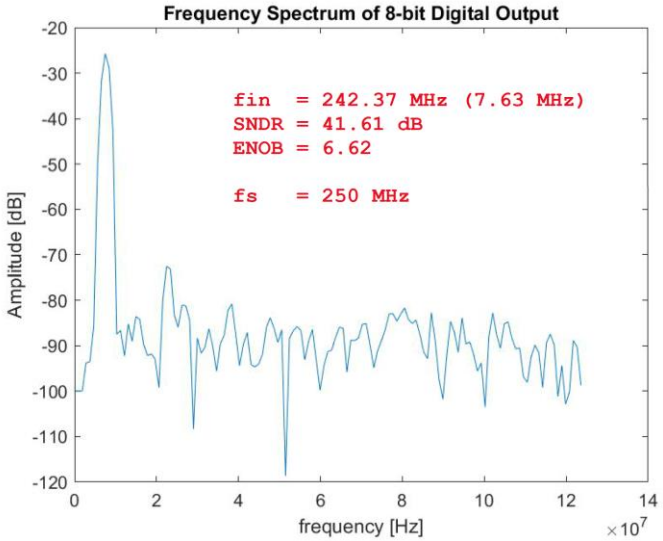

Figure 13: FFT on ADC Output (Sampling Frequency)

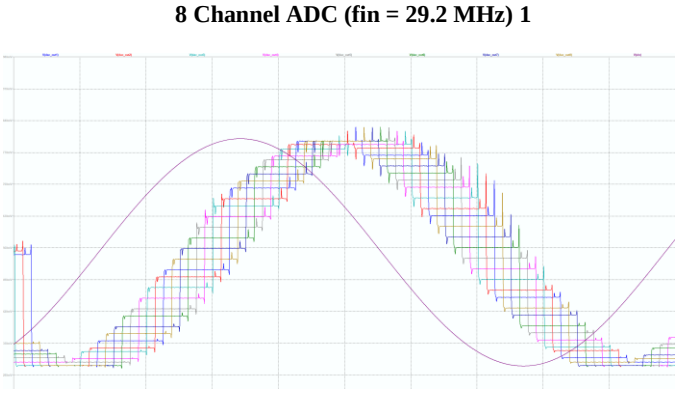

Figure 14: 8-Way Time Interleaved Transient Output

XIII. Performance Summary

ADC Specifications	Desired	Achieved	units
SNDR	49.92	43.59	dB
CLK	5	2	GHz
FOM	22	481.2	fJ/(conv*step)
ENOB	8	6.95	bits
Power	-	119	mW

Table 2: Summary Table of Results

Note: FOM and power in performance summary are calculated based on time interleaved ADC.

XIV. Conclusion

The SAR architecture provides a robust building block for an interleaved high-performance ADC. Through the design of a behavioral model in MATLAB as well as through full implementation at the schematic level, this project exposed me to a whole host of problems, challenges, and solutions that ultimately taught me the fundamentals of SAR ADC design. The behavioral model also provided metrics and a foundation for evaluating the transistor level design. Architecturally the SAR ADC provides a good framework for a time interleaved application. While the performance was acceptable, in future revisions the power consumption could be greatly reduced, and perhaps the speed could be improved upon through optimizing the logic.

XV. Appendix

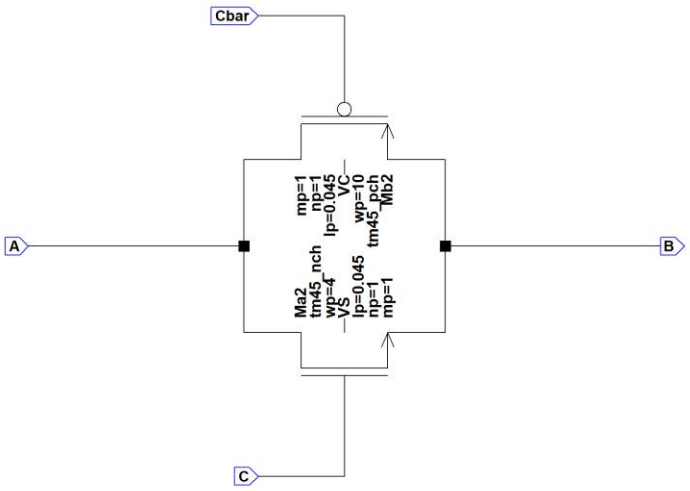

Figure 15:Transmission Gate

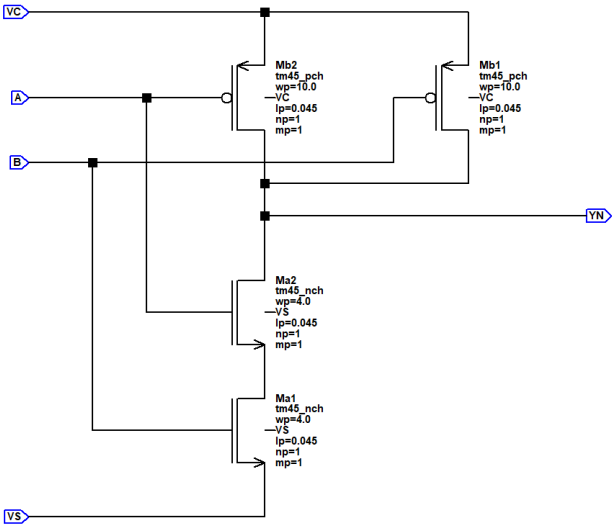

Figure 16: 2 Input NAND Gate

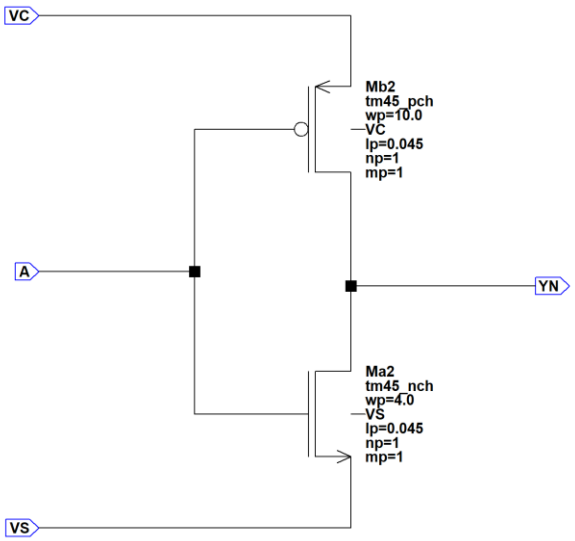

Figure 3: Inverter

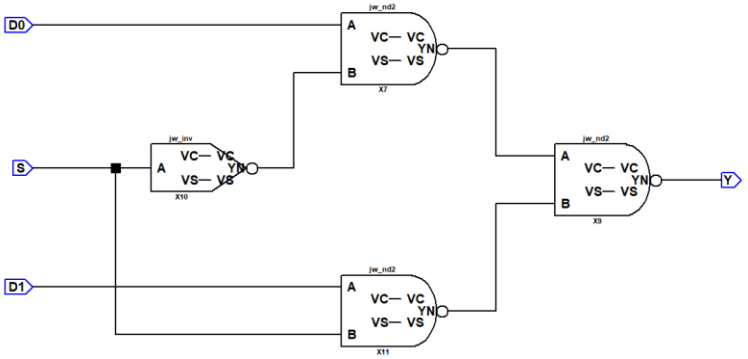

Figure 17:2-to-1 Multiplexer

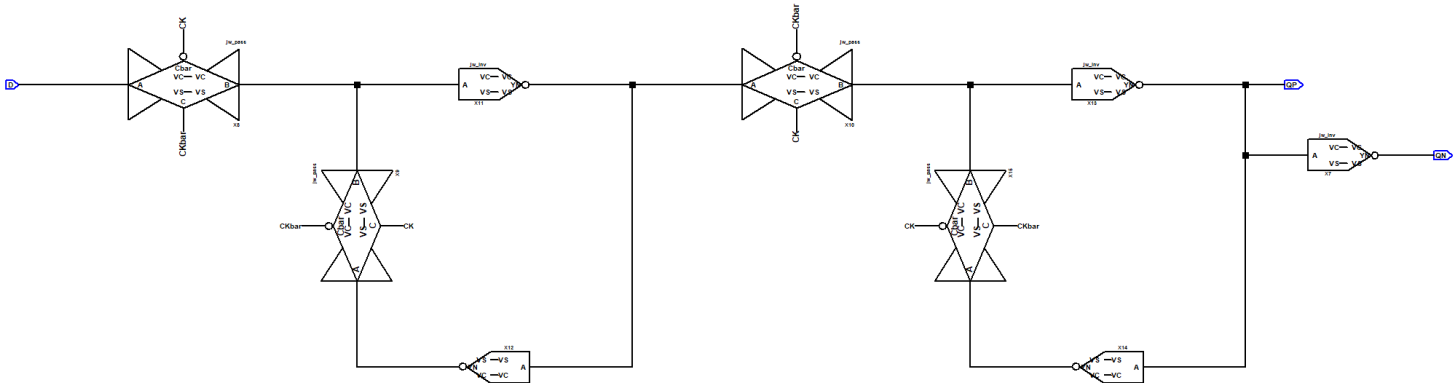

Figure 4: D-Flip-Flop

XVI. References

[1] T. C. Carusone, D. A. Johns, and K. W. Martin, “Chapter 17: Nyquist Rate A/D Converters,” in Analog Integrated Circuit Design, 2nd ed., Danvers, MA: John Wiley & Sons, 2012, pp. 646–695.

[2] B. Yazdani, A. Khorami, and M. Sharifkhani, “Low-power DAC with charge redistribution sampling method for SAR ADCS,” Electronics Letters, vol. 52, no. 3, pp. 187–188, Feb. 2016.

[3] B. P. Ginsburg and A. P. Chandrakasan, “An energy-efficient charge recycling approach for a SAR converter with capacitive DAC,” 2005 IEEE International Symposium on Circuits and Systems, pp. 184–187, Aug. 2005.

[4] B. Ginetti and P. Jespers, “Reliability of code density test for high resolution adcs,” *Electronics Letters*, vol. 27, no. 24, p. 2231, Nov. 1991.

[5] D. Schinkel, E. Mensink, E. Klumperink, E. van Tuijl, and B. Nauta, “A double-tail latch-type voltage sense amplifier with 18ps setup+hold time,” *2007 IEEE International Solid-State Circuits Conference. Digest of Technical Papers*, 2007.

I would like to thank Qiaochu and Professor Chen for their instruction, guidance, and motivation to complete this ADC. It was not easy, but it was worth it!