EE343 CMOS Mixed Signal Design
Lab 4: 4 bit Flash ADC Layout/Simulation

Jeffrey Cool 006442018
Dr. M. Wagdy

12-15-2015

Contents

1 Introduction

The following describes the design, layout and simulation of a CMOS 4 bit flash ADC within the Microwind/DSCH environment. A flash ADC is conceptually simple, yet the design becomes component-intensive (and slow) as the resolution is increased. A 4 bit device has manageably few parts: 15 comparators, 15 voltage references, and a priority encoder consisting of 20 or so gates. The development of each of these components, plus several peripherals, are documented in detail in this report.

2 Quantization

A 4 bit ADC has 16 possible output states corresponding to 16 analog voltage ranges, with a difference of 1 LSB at the output representing a difference of one quantizing step (or $1/16^{th}$ of the full scale input voltage) at the input. Linear quantization can be accomplished using either truncation or rounding. An ADC that uses truncation requires that the input be one full quantizing step above zero for a one to appear at the output, while one that uses rounding only requires one half of a quantizing step to do the same thing [3]. Either is easily accomplished with properly designed voltage references.

3 Voltage References

The simplest way to generate 15 evenly spaced voltage references is to use a voltage divider made of 16 resistors in series. Below is a figure comparing schemes for rounding and truncation. Resistor layout is

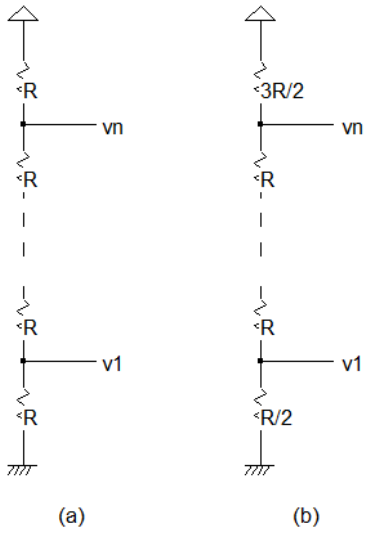

Figure 1: Voltage Reference Schemes for Truncation (a) and Rounding (b)

accomplished in Microwind using the *Generate→Resistance* command. There are a few considerations to keep track of when sizing the resistors. Most obviously, it is undesirable for the network to draw excessive current, and at the same time, it is necessary that each reference voltage not have too high of an impedance. Also, matching is improved by increasing both an integrated resistor's width and length [1].

For convenience, the design settled upon allows for operation in rounding or truncating mode. Each value R, as depicted in figure 1 is composed of 2 poly resistors with nominal resistances of 200Ω and a width of 10λ in series. Fractional values of R are obtained by selectively connecting 2 of 4 buses to Vdd or Vss. The layout is shown in figure 2. Notice that the actual resistance value of the poly resistors is 196Ω. This is due to the finite feature size. Also, two dummy resistors had to be included at the extremities of the divider so that all of the others matched well.

The divider functions in truncation mode with VddT connected to Vdd and VssT connected to Vss, and operates in rounding mode with VddR and VssR connected to the respective power buses. Figure 3 shows simulation results for operation in both modes.

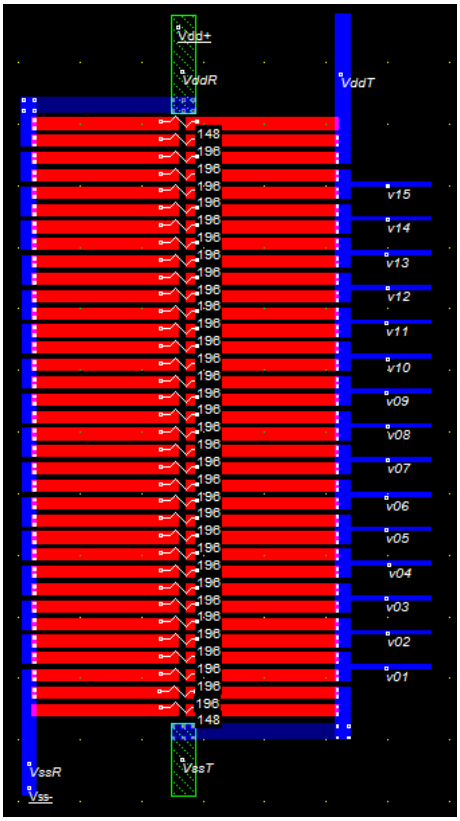

Figure 2: Layout of Resistor Divider Circuit for Voltage Reference Generation

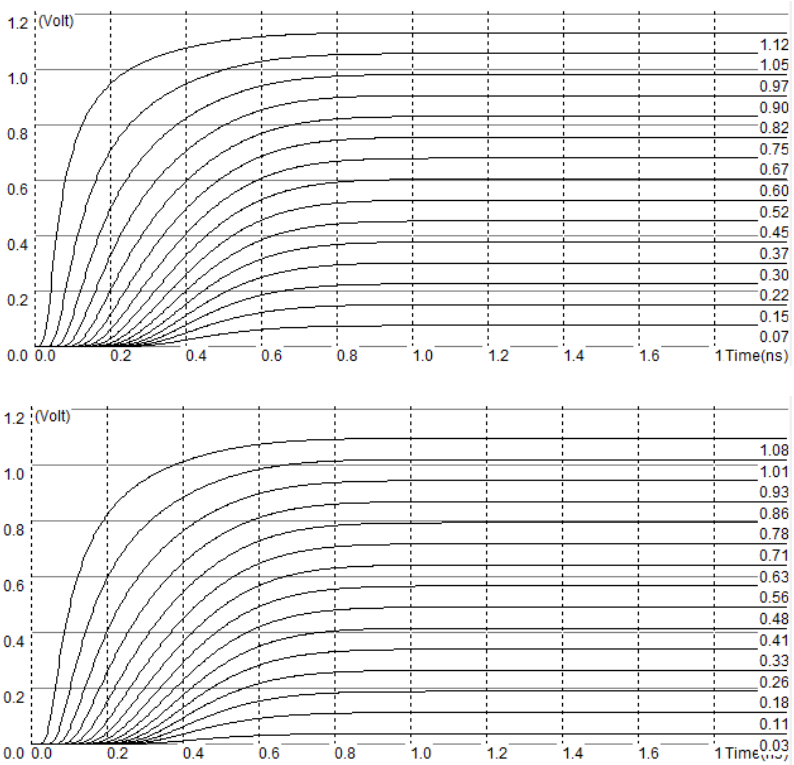

Figure 3: Voltage references. Top: Truncation mode, Bottom: Rounding mode

4 The Comparator

The comparator is essentially a high gain differential amplifier. Since the lowest and highest voltage references are near the supply rails, it is important that the comparator have rail to rail input range. A simple N-channel self biased differential amp performs fine with common mode voltages well above V_{TN}, but doesn't really work with low CM inputs. The opposite is true for P-channel diff amps, but putting these two complementary circuits in parallel as shown in figure 4 gives good results.

Automatic layout of circuits like this in Microwind yields unappealing results. To minimize the size

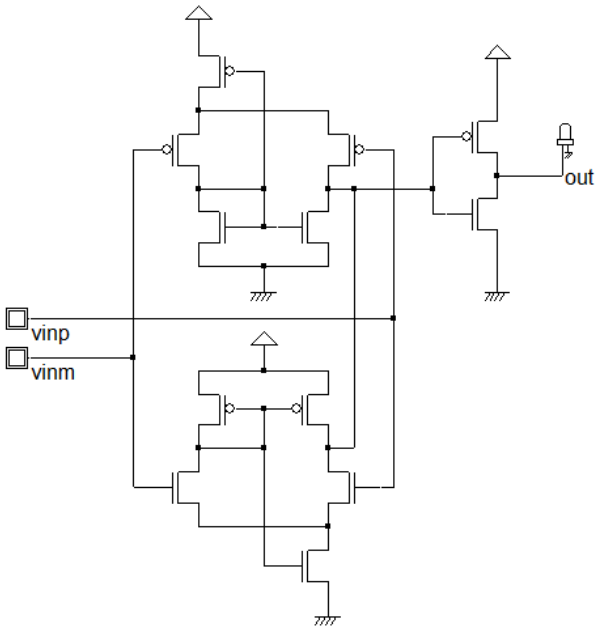

Figure 4: Comparator Schematic

and maximize the attractiveness of the comparator, the layout was performed manually with the help of the *MOS Generator*. The sizes of the MOSFETs were chosen to be the minimum allowable by the design rules ($W/L_{NMOS} = 4/2, W/L_{PMOS} = 12/2$) to minimize the input capacitance. The final layout is shown in figure 5.

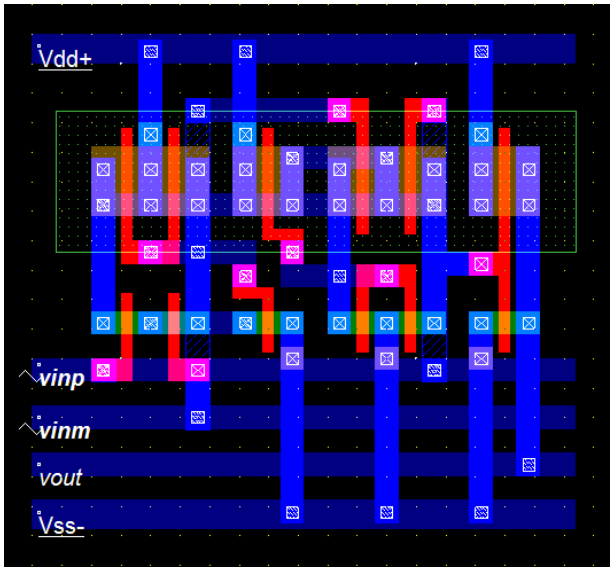

Figure 5: Comparator Layout

4.1 Comparator Transient Response

The three figures below show the response of the comparator to a differential input of just over 1/4 of a single quantizing step below and above three representative reference voltages (from truncation mode): v1–one quantizing step above Vss, v8–the midpoint, and v15–one quantizing step below Vdd.

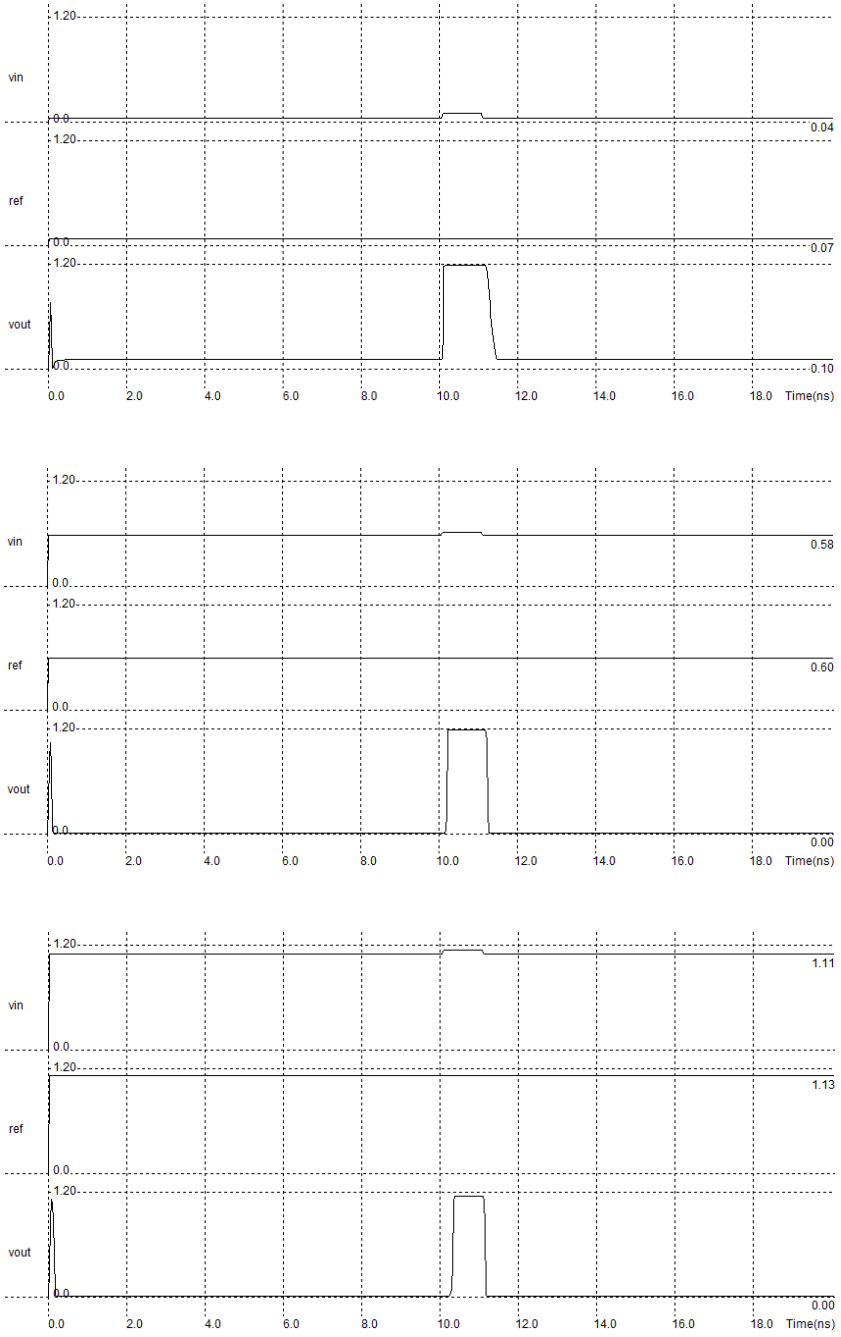

Figure 6: Simulation of Comparator Transient Response at Three Reference Voltages

The plots above demonstrate that the comparator performs well within the requirements. We notice an appreciable delay of around 150ps on the rising edge when the reference voltage is v15 and a similar delay on the falling edge when the reference voltage is v1. This will put an upper limit on the speed of conversion, which will be discussed later in detail.

4.2 Comparator Common Mode Performance

It is expected that this simple comparator circuit will have some nonzero common mode gain. For common mode signals, the comparator is essentially a common source amplifier with source degeneration, followed by a CMOS inverter. Figure 7 shows the response of the comparator to a common mode sweep from 0V to Vdd. Note that v_x is the input to the inverter. A negative common mode gain means that with

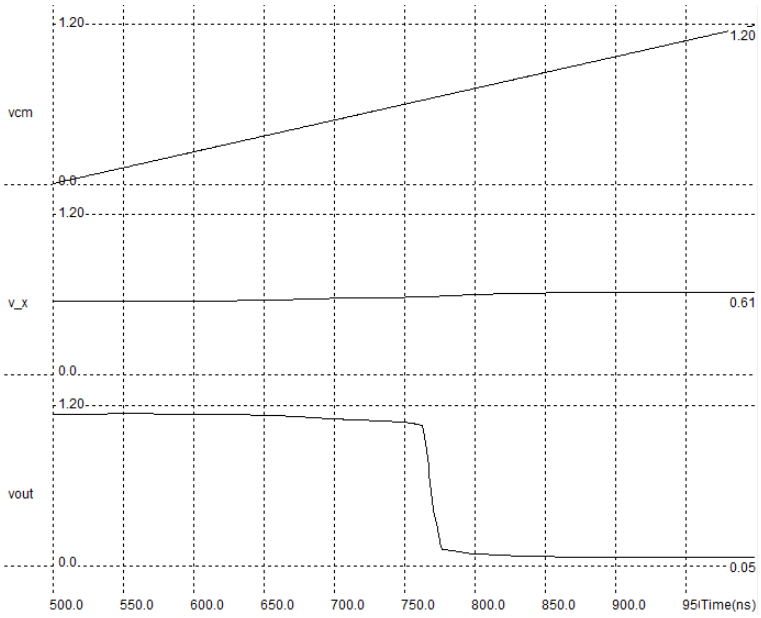

Figure 7: Simulated Response of Comparator to Common Mode Input (vcm)

a reference voltage close to Vdd, the comparator output will remain low for sufficiently small positive differential inputs. To combat this, additional small resistors can be added to the ends of the reference voltage divider. Given that R≈400Ω, it is expected that a good value for the compensation resistor would be in the tens of ohms. More on that Later

5 Thermometer Code

Quantization of the analog signal results in a 16 bit signal known as thermometer code. The circuit used for quantization is simply an array of comparators, as shown in figure 8. The partial schematic describes the simultaneous comparison of an analog voltage at Vin with 15 reference voltages. The reference voltages are applied at the inverting inputs, and the analog voltage is applied to the noninverting inputs, so the comparator output will swing high when the input exceeds its reference.

Considering the relative sizes of the individual comparators and the resistor array, a layout of 3 rows of 5 comparators adjacent to the resistor array makes relatively good use of chip area. The comparator was invoked 5 times using the *Insert Layout* command, then these were arranged in a tightly packed horizontal row. After measuring the length of the array, 10 I/O buses were generated in metal 2 using the *Generate→ Metal Bus* tool. Inputs and outputs were connected to the buses using *Generate→ Metal Path* tool and *Ctrl - W* to quickly connect metal layers, and excess metal was trimmed manually. This structure was copied and pasted 3 times and aligned vertically, then all of the noninverting inputs were connected to a vertical path at the left. The voltage divider was inserted, and the outputs were routed to the inverting comparator inputs. Precise placement is accomplished using the *Move by a Step* tool in Microwind. The preliminary layout is shown in figure 9. Final floor planning decisions – namely the location and shape of the power buses – will be made when the full ADC circuit is complete. Note that the vertical green stripes are metal 4 paths, not N wells.

Proper function is verified in the simulation diagram in figure 10, which shows the response of the circuit to a symmetrical triangular pulse with a slope of ±0.24V/ns.

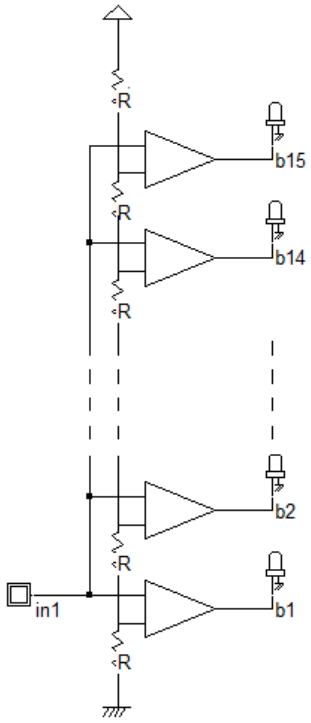

Figure 8: Quantization Circuit Schematic

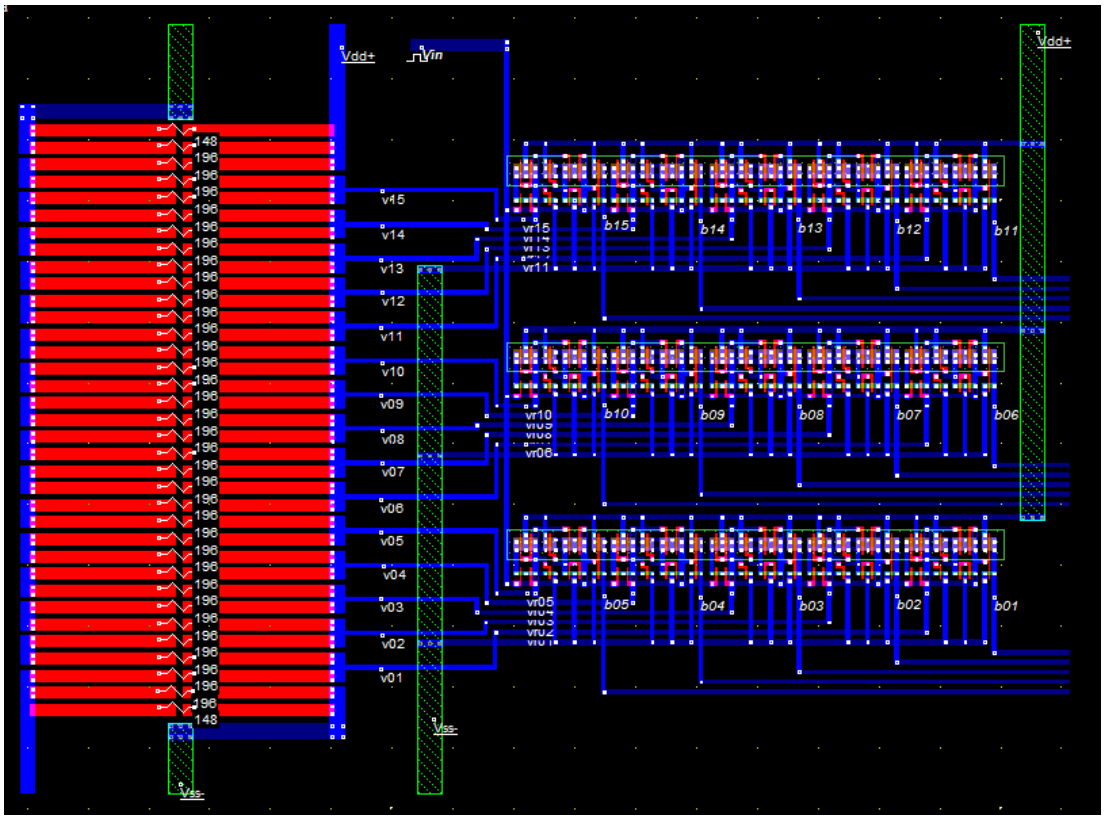

Figure 9: Quantization Circuit Layout

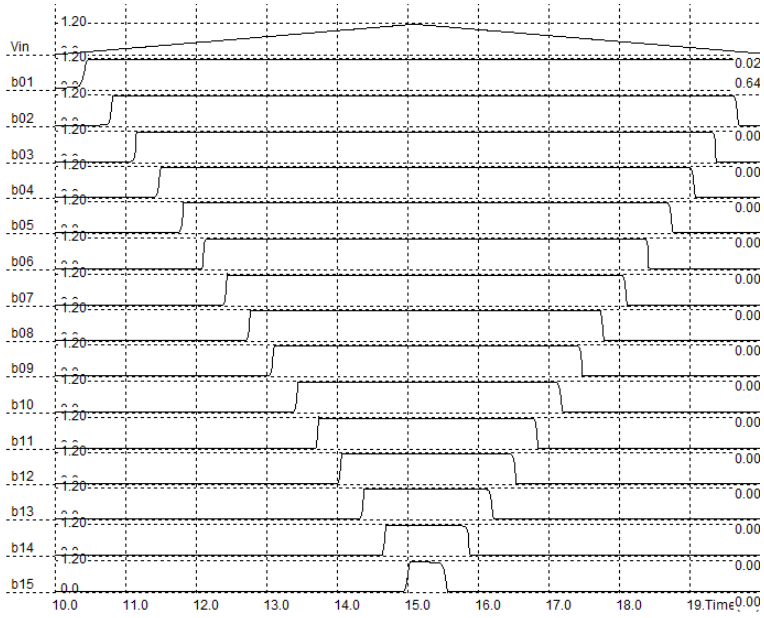

Figure 10: Simulation Diagram: Thermometer Code Circuit Response (truncation mode)

6 The Priority Encoder

The thermometer code can be completely represented as 4 bit binary code. Encoding the 16 bits from the comparator outputs into a 4 bit signal is accomplished with a logic circuit derived from the truth table shown in figure 12, using a method described in [3]. A 4 input NAND gate is available in the DSCH Library, so the logical expressions derived were modified to develop a mostly NAND structure, by observing that a sum of products can be accomplished with a nand gate if the inputs are inverted. The transformed

y3	y2	y1	y0		b15	b14	b13	b12	b11	b10	b9	b8	b1	b6	b5	b4	b3	b2	b1
0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1		0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0	0	1	0		0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
0	0	1	1		0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
0	1	0	0		0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
0	1	0	1		0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
0	1	1	0		0	0	0	0	0	0	0	0	0	1	1	1	1	1	1
0	1	1	1		0	0	0	0	0	0	0	0	1	1	1	1	1	1	1
1	0	0	0		0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
1	0	0	1		0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
1	0	1	0		0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
1	0	1	1		0	0	0	0	1	1	1	1	1	1	1	1	1	1	1
1	1	0	0		0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
1	1	0	1		0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	0		0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

y3 = b8
y2 = b4.~b8 + b12
y1 = b2.~b4 + b6.~b8 + b10.~b12 + b14
y0 = b1.~b2 + b3.~b4 + b5.~b6 + b7.~b8 + b9.~b10 + b11.~b12 + b13.~b14 + b15

Figure 11: Thermometer to Binary Truth Table

logic expressions are shown below. Layout for the priority encoder was generated automatically using the *Compile Verilog File* function in Microwind. The logic circuit is constructed at the gate level in DSCH,

then converted into a Verilog file, and compiled in Microwind. The result is shown in figure 13.

$$y_3 = b_8 \tag{1}$$

$$y_2 = \overline{\overline{b_4 \cdot \overline{b_8}} \cdot \overline{b_{12}}} \tag{2}$$

$$y_1 = \overline{\overline{b_2 \cdot \overline{b_4}} \cdot \overline{b_6 \cdot \overline{b_8}} \cdot \overline{b_{10} \cdot \overline{b_{12}}} \cdot \overline{b_{14}}} \tag{3}$$

$$y_0 = \overline{\overline{b_1 \cdot \overline{b_2}} \cdot \overline{b_3 \cdot \overline{b_4}} \cdot \overline{b_5 \cdot \overline{b_6}} \cdot \overline{b_7 \cdot \overline{b_8}}} + \overline{\overline{b_9 \cdot \overline{b_{10}}} \cdot \overline{b_{11} \cdot \overline{b_{12}}} \cdot \overline{b_{13} \cdot \overline{b_{14}}} \cdot \overline{b_{15}}} \tag{4}$$

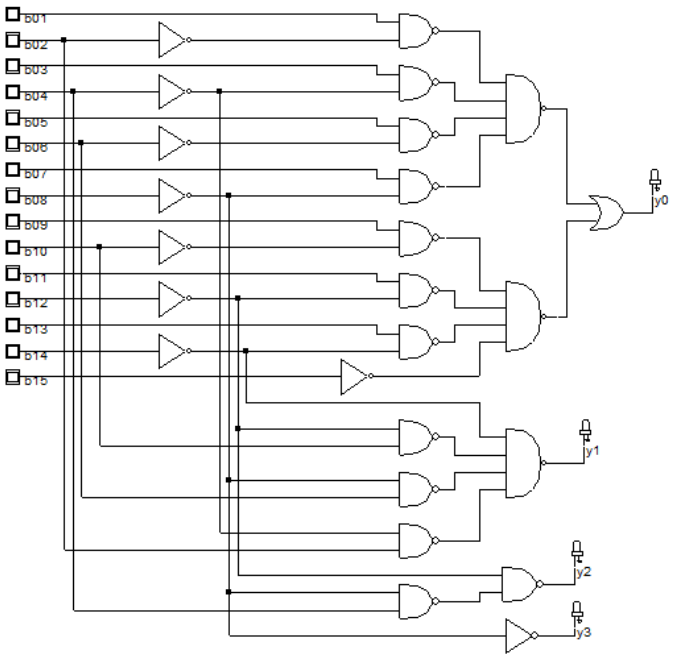

Figure 12: Priority Encoder Logic Diagram

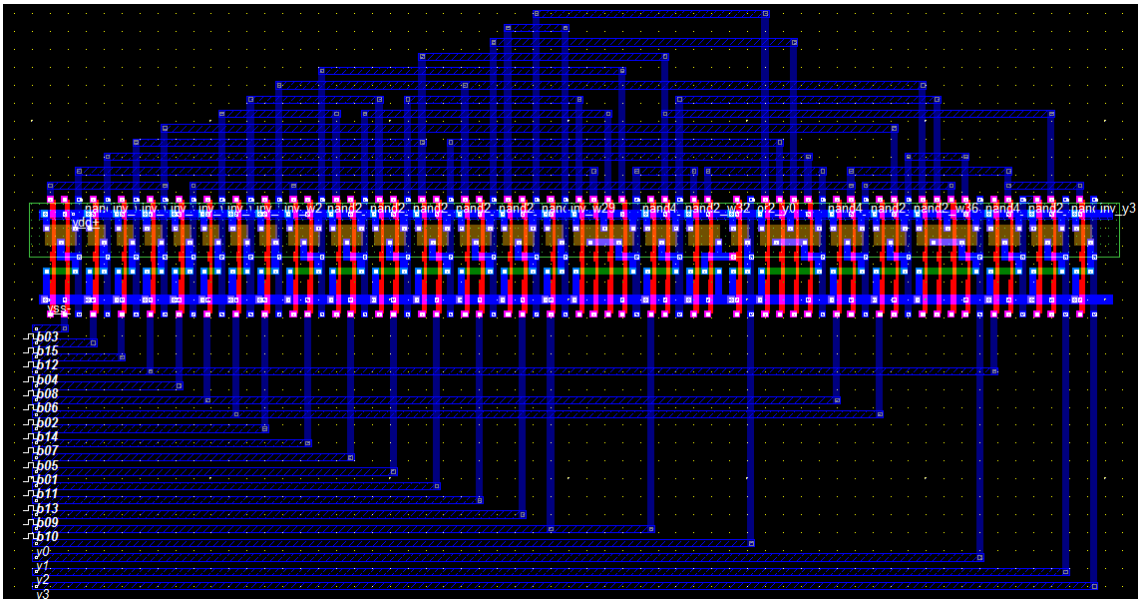

Figure 13: Priority Encoder Layout (Automatic)

7 A Minimum Flash ADC

Routing the Comparator outputs to the inputs of the priority encoder completes the basic circuit of the 4 bit flash ADC. This, again, was performed using the *Generate→Metal Path* tool in Microwind. The layout is shown with a simplified resistor connection for truncation in figure 16. A simulation diagram in figure 14 shows the binary output (y_i) with the thermometer code (bi) to the same triangular pulse as in figure 10. The total area on chip occupied by the minimal ADC is 805000 λ^2 or 2898 μm^2 using 0.12μm

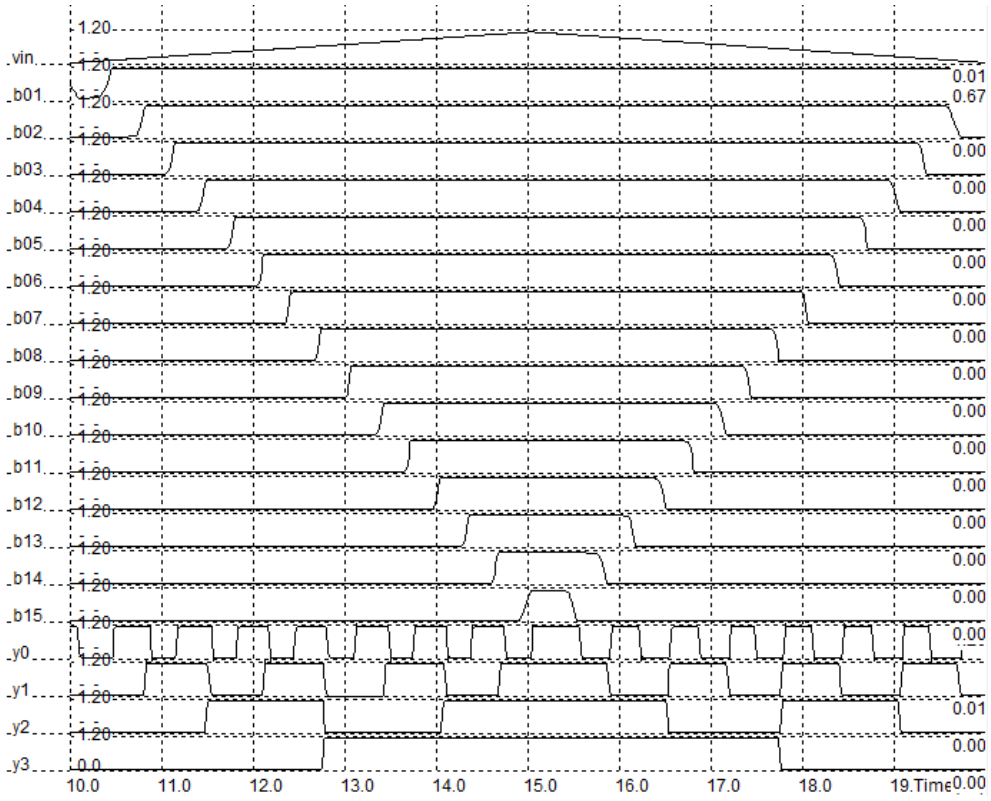

Figure 14: ADC Transient Response

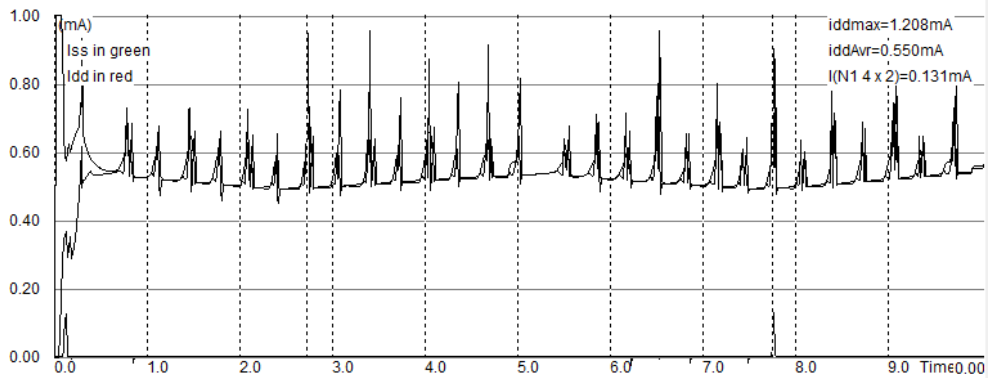

Figure 15: Supply Currents in Minimal ADC

CMOS technology. Average Power consumption can be estimated using the *Simulate Voltage, Current, v. Time* tool in Microwind. A plot showing only the currents as a response to the same triangular pulse used earlier is shown in figure 15. Average power can be calculated as $V_{dd} \cdot i_{ddAVG} = 1.2V \cdot 0.55mA = 660\mu W$. Spikes in the current result from switching: there is a brief interval during switching when both N and P channel devices of inverters and logic gates conduct. So the total power consumption is derived from steady state biasing currents and the switching frequency.

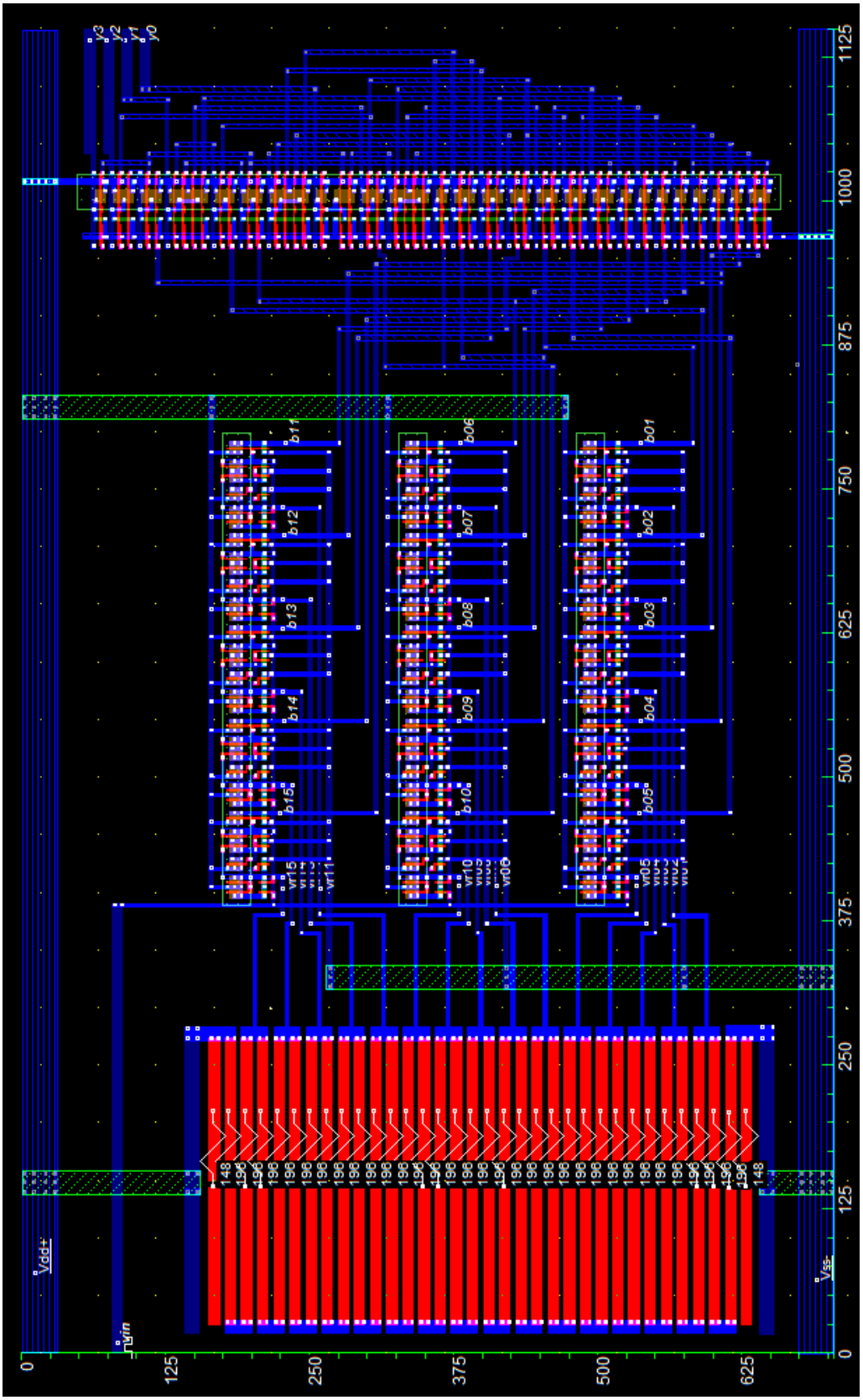

Figure 16: Minimum ADC Layout

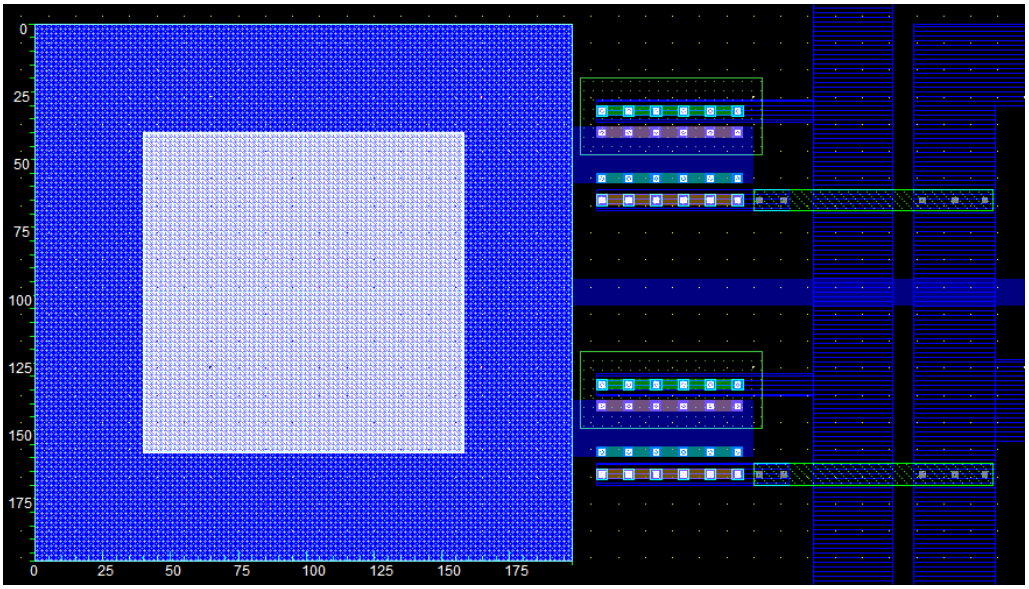

Figure 17: Detail of Input Pad with ESD Protection

8 Pads and Output Conditioning

If the device described were to be manufactured, there would be no way to test it since there are no I/O or test pads. Given the size of the layout, traditional IO pads ($100\mu m$ X $100\mu m$) would be gratuitous; however, MOSIS specifies that test pads need be only ¿$6\mu m$ on a side [1]. To fit the layout with a pad frame with 7X7μm pads, the following procedure is used:

1. *Generate→I/O Pads→Single Pad*, specifying the length in microns.
2. Add a metal 2 path extending out from the pad using *Generate→Metal Path*, specifying a width of 10λ.
3. Power buses created using *Generate→Metal Bus*, specifying width, length, separation and number of buses. ($1.8\mu m$, $100\mu m$, $0.48\mu m$, 2, respectively).
4. Copy, paste, and rotate buses and arrange around the main layout, trimming manually.
5. Copy, paste, and rotate pads, placing them around the pad frame using *Ctrl - M*.
6. Select Vdd/Vss pads and connect their respective metal 2 paths to the inner/outer buses using *Ctrl - W*.
7. Connect ADC Vdd and Vss buses to the main Vdd and Vss buses by manually stretching and using *Ctrl - W*.
8. Select input pads and add ESD protection diodes as shown in figure 17.
9. Manually stretch input and output buses to connect with appropriate pads.

The ADC layout is rectangular, so the pad frame is made rectangular, with 4 pads on each side. Input pads are on the left, output pads on the right, and Vdd/Vss pads are on the top and bottom.

8.1 ESD protection

ESD protection in CMOS circuits is accomplished by laying out clamping diodes connected to the input pad and the power bus to create the circuit shown in figure 18. The diode connected to Vss consists two parallel diodes formed from N and P diffusion inside an N well with the P diffusion connected with contacts (using *Ctrl - W*) to the pad, and the N diffusion connected to Vss, forming a reverse biased N+ - N - P+ junction. The diode connected to Vdd is only different in that it is laid out directly on the

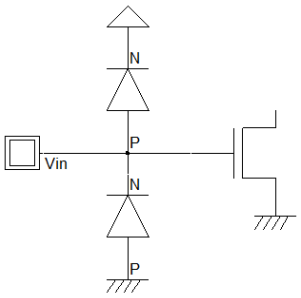

Figure 18: Clamping Circuit for ESD Protection

substrate, forming a reverse biased N+ - P - P+ junction [1]. A single set of diodes was laid out manually, and then copied and pasted to the remaining input pads.

8.2 Output Conditioning

An output pad is effectively a capacitive load. Using the *View Node* tool in microwind, it is discovered that a single pad has a capacitance to the substrate of 130fF. The priority encoder uses NMOS transistors with a W/L=4/2. According to the *MOS generator Dialog Window* a transistor this size can source a maximum of $145\mu A$. The minimum delay can thus be calculated as:

$$C = \frac{Q}{V} \; ; \; Q = I \cdot T \; \rightarrow \; T_{delay} = \frac{C \cdot Vdd}{I} = \frac{130fF \cdot 1.2V}{145\mu A} \approx 1ns$$

This is nearly an order of magnitude greater than the maximum delay from the comparator output. Simulation Results are shown in figure 19. As can be seen, with the same triangular pulse input used

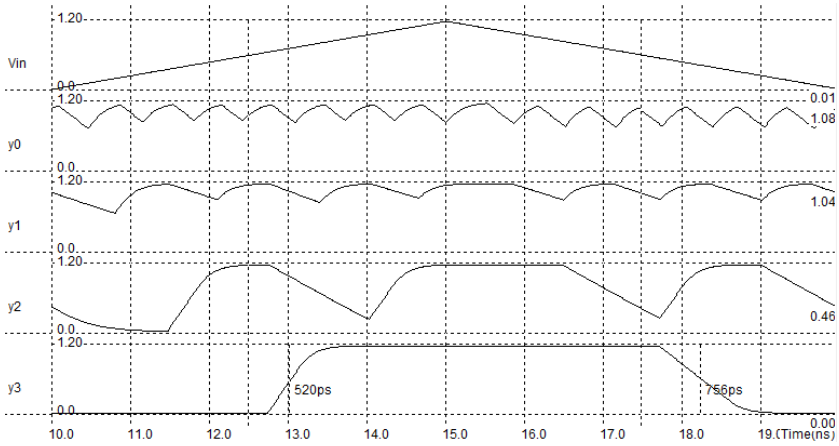

Figure 19: Simulation Diagram Showing Uncompensated Effect of Capacitive Load

before, the output just can't keep up with the switching frequency.

It is possible to obtain minimal delay by cascading progressively wider inverters between the digital output and the capacitive load. According to [1], using N inverters, each one with a width A times that of the previous one gives the minimum delay if:

$$N = \ln \frac{C_{load}}{C_{in}}$$

$$A = \left[\frac{C_{load}}{c_{in}} \right]^{\frac{1}{N}}$$

Since the gate capacitance of the minimally sized inverter is approximately 1fF...

$$N = \ln \frac{130fF}{1fF} = 4.868 \rightarrow 5\, stages$$

$$A = \left[\frac{130fF}{1fF}\right]^{\frac{1}{4.868}} = 2.718 = e$$

Since we don't want logical inversion, we will add an additional small inverter at the beginning of the chain, ensuring that every successive inverter has approximately e times the width of the previous one, as in the logic diagram of figure 20. Layout for this circuit was accomplished using the *MOS generator*.

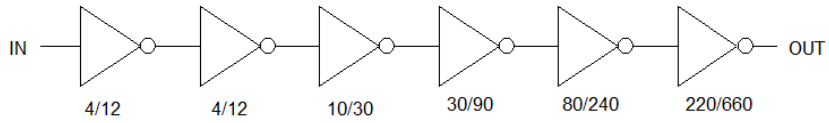

Figure 20: Output Buffer Showing Transistor Widths (N/P)

First, the two small inverters were generated (1 two finger NMOS and 1 2 finger PMOS). Next, a single finger 10/2 NMOS and a 30/2 PMOS were generated for the next inverter. Pairs of N and P channel devices were generated with 3, 8 and 22 fingers for the remaining inverters. The devices were arranged using *Ctrl - M* and substrate ties and ties to the power bus were created with *Ctrl - W*. The layout is shown in figure 21. This circuit is invoked 4 times in the main layout and inserted horizontally in two

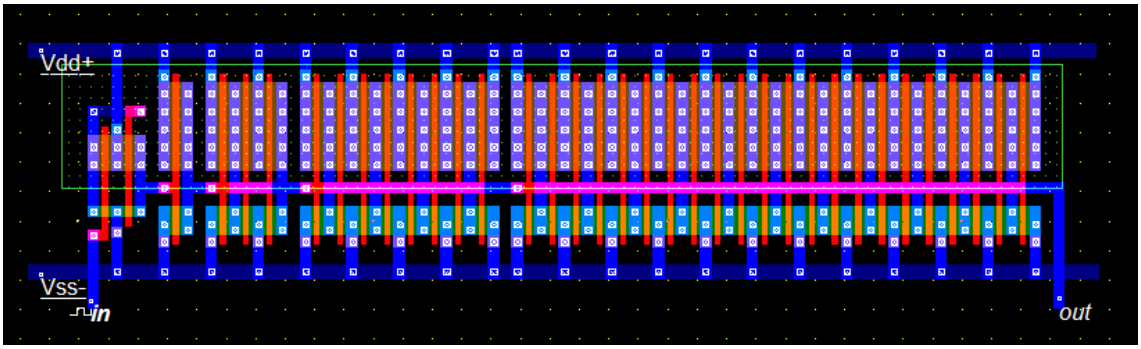

Figure 21: Output Buffer Layout

rows of two into the layout area. Since there are only 4 inputs and 4 outputs, routing was done manually. Simulation results verify that delay is dramatically reduced. See figure 22. In addition to the improved speed, there is a dramatic increase in supply current draw. Observing the voltage and current versus time plot shown in figure 23 indicates that the instantaneous current draw during switching can reach nearly 40mA! Average current with this particular input is $I_{AVG} = 1.527mA$. With a DC input (no switching) the quiescent current is $I_Q = 550\mu A$. We can define the total current draw as:

$$I_{supply} = I_Q + I_{sw}$$

The simulation results in figure 23 show 29 switching events in a period of 10ns, which corresponds to a switching frequency of 2.9GHz. Since the current due to each switching event is derived from adding or removing a capacitor charge equal to $C_{out} \cdot Vdd$, the switching current must be linearly related to the switching frequency:

$$I_{sw} = \frac{(1.527mA - 0.55mA)}{2.9GHz}$$

$$I_{sw} = 3.67 \times 10^{-13} \frac{A}{Hz} \cdot f_{sw}$$

So the total power consumption as a function of switching frequency is:

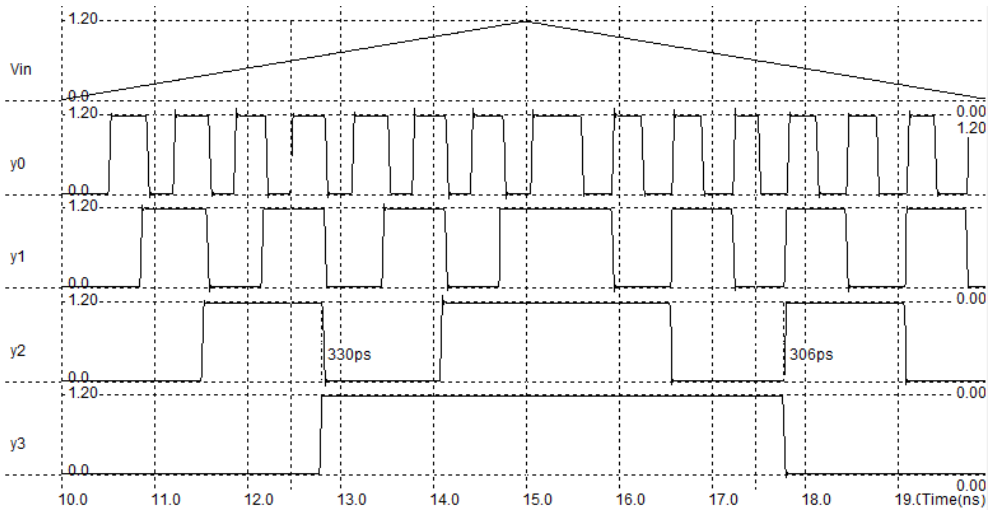

Figure 22: Simulation Diagram Showing Effect of Output Buffer

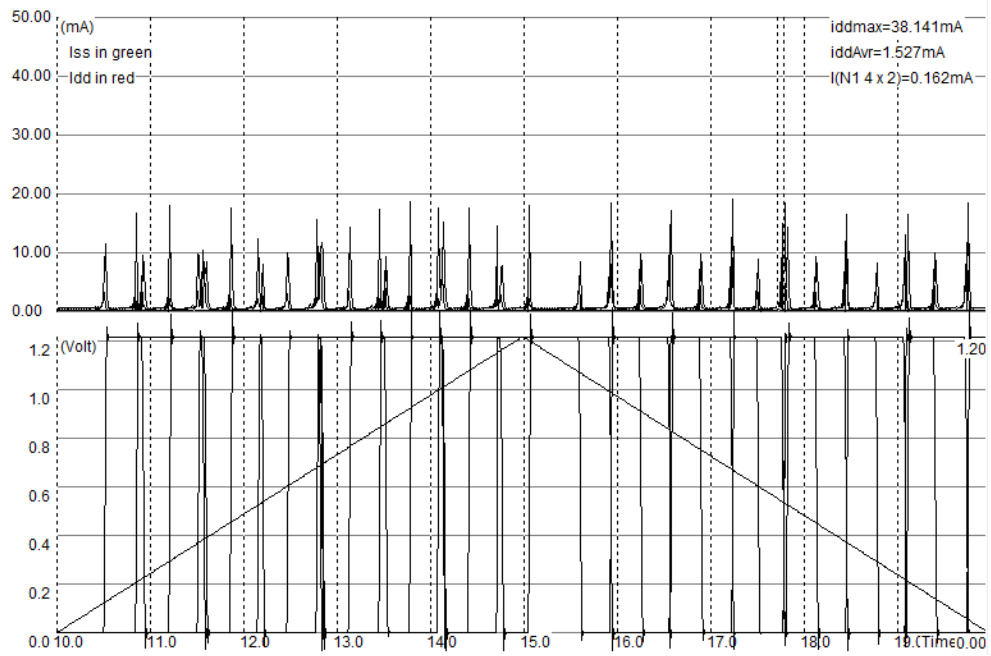

Figure 23: Voltages and Currents v. Time Simulation for Circuit with Output Buffers

$$P_{AVG} = 1.2V \cdot \left[0.55mA + 3.67 \times 10^{-13} \frac{A}{Hz} \cdot f_{sw}\right]$$
$$P_{AVG} = \left[6.6 \times 10^{-4} + 4.4 \times 10^{-13} \cdot f_{sw}\right] W$$

Before we look at the complete layout, there are two issues to address:

1. The common mode gain of the comparators

2. The mode of quantization (truncation or rounding)

9 Programmable Quantization

All of the results obtained so far demonstrated the ADC operating in truncation mode. It is cumbersome to switch between the two modes via metal routing, so in the interest of user friendliness, 4 switches can be added added to the voltage divider to make it easy. Perfect switches are not necessary: The negative common mode gain of the comparators can be compensated for by adding small resistances in series with the voltage divider in between the divider and the power buses. This is accomplished with non-ideal CMOS switches (transmission gates), as shown in figure 24. This layout was generated using the *MOS*

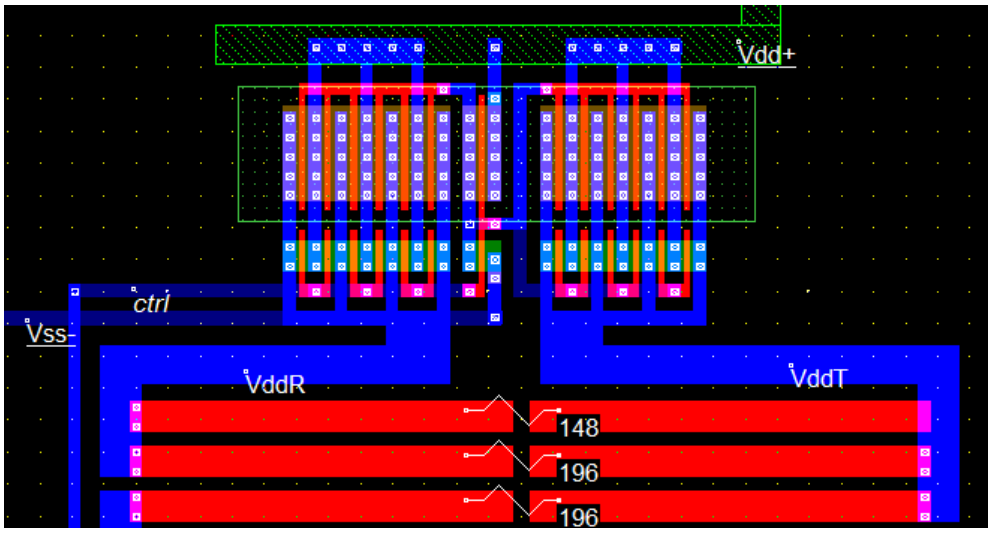

Figure 24: Detail of Transmission Gates Used to Select Quantization Mode

Generator, and was routed manually. Note also that an identical network is connected between VssR and VssT and Vss (as shown in figure 2

With this addition, quantization via rounding or truncation can be set with a logic signal. Simulation results are shown in figure 25. Notice that when rounding is used, switching occurs at a lower voltage than with truncation. The complete layout for the ADC with testing I/O pads, output conditioning, and

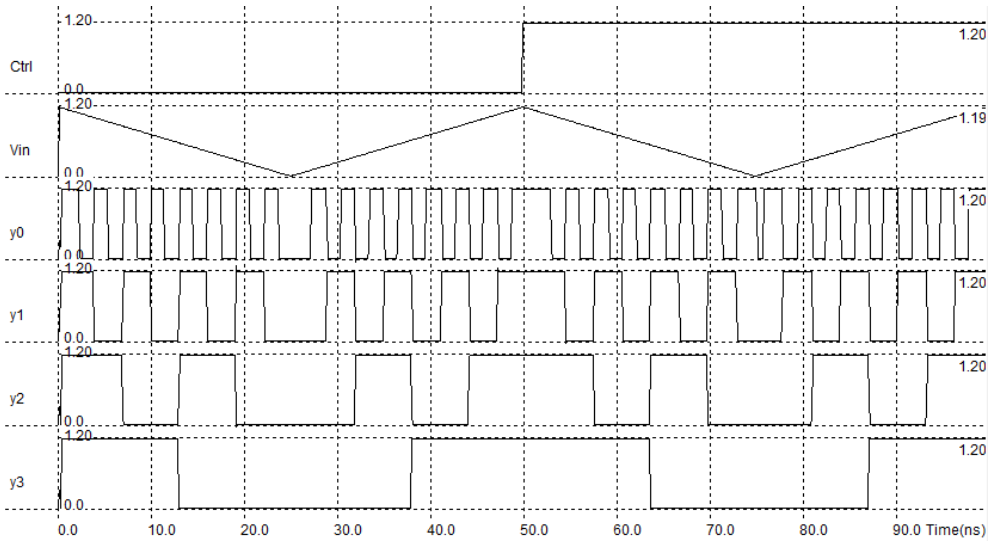

Figure 25: Simulation showing Truncation and Rounding (rounding starts at 50ns)

programmable quantization is shown in figure 26. The output driver circuits were intentionally placed as far from the resistors as possible, to minimize noise coupling to the reference voltages. The total area on chip is approximately 1575X2150λ which is equal to 12255μm^2.

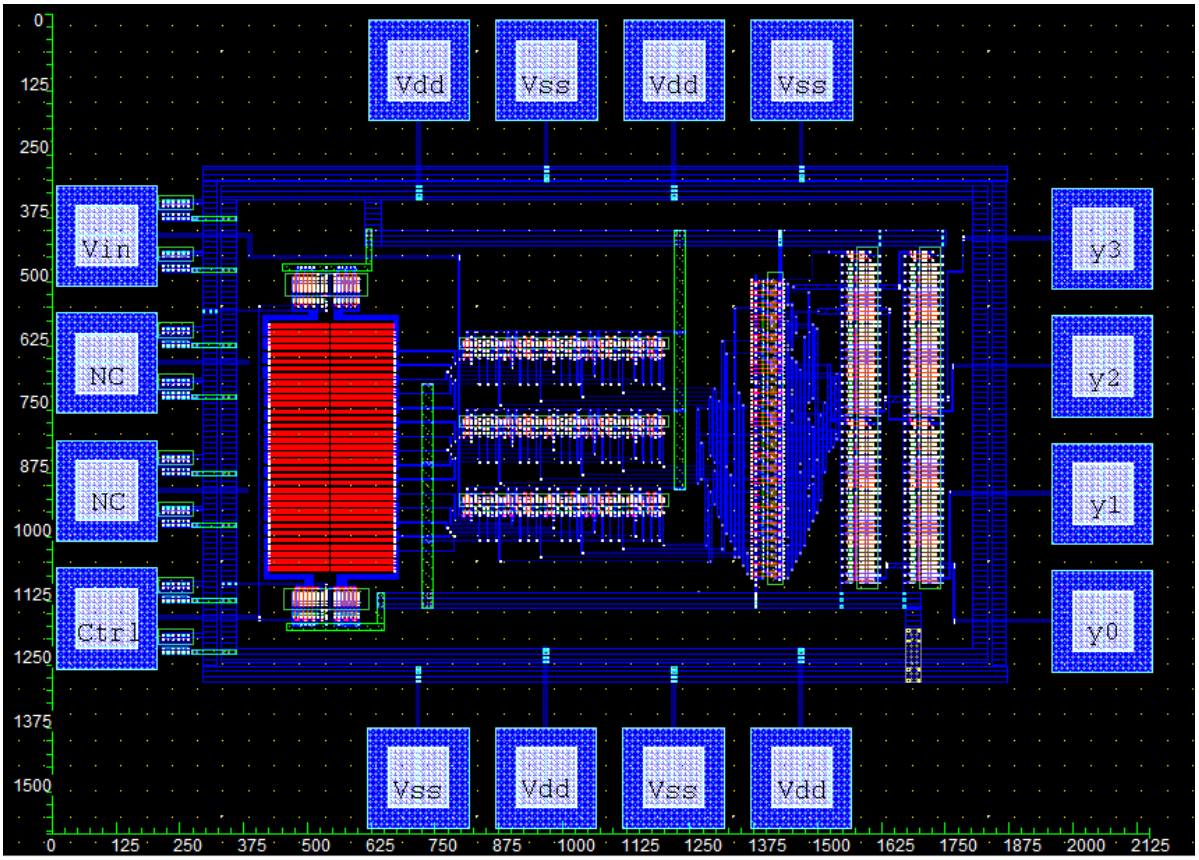

Figure 26: Complete ADC layout

10 Summary of Design and Performance

1. Voltage references were generated with a resistor divider configured with CMOS switches to allow for operation in either truncation or rounding mode with control via a single pin.

2. A parallel set of N and P channel differential amplifiers followed by an inverter served as the comparator. Layout for this was performed using the *MOS Generator* in Microwind and manual routing. The comparator exhibited some negative common mode gain, the effect of which was opposed by resistance (the channel resistance of the MOS switches) introduced in series with the main resistor array.

3. The priority encoder was designed as described in [3], with the sum of products accomplished with mostly nand static logic. Layout was performed automatically using DSCH to create a logic diagram and to extract a Verilog file, then the file was compiled in Microwind.

4. Output conditioning was accomplished with a minimal delay network of cascaded inverters, as described in [1]. This was needed due to the introduction of a capacitive load (output pads).

5. I/O pads and a pad frame were laid out using tools in Microwind, in an "software-assisted" manual layout. ESD protection diodes were added to the inputs using a scheme described in [1].

6. The total chip area was determined using the measuring tool in Microwind to be $12255\mu m^2$.

7. The total power consumption was determined using the Microwind *Voltage and Current v. Time* simulation tool. Based on measurements of the current at different switching frequencies, The total power consumption was found to be a function of the switching frequency:

$$P_{AVG} = \left[660 + 4.4 \times 10^{-7} \cdot f_{sw}\right] \mu W$$

11 Conclusion

The design presented performs well in simulation (SPICE level 3), and the layout generated has reasonably good floor planning. There is still more investigation to do concerning the maximum conversion rate and conversion accuracy. If a sample and hold circuit is to be used, the maximum settling time of the comparators with small differential inputs must be determined. One notable observation is that there really is no need for a general purpose comparator with rail to rail common mode performance. The reason being that the common mode input into each comparator does not swing rail to rail, but is fixed in a range of at most 1/2 Vdd about the fixed reference voltage. So a different comparators can be designed for different common mode performance, perhaps conserving chip area. Also, more careful floor planning and routing could result in reduced chip area, perhaps by as much as a factor of 2.

References

[1] Baker, R. Jacob, *CMOS Circuit Design, Layout and Simulation*, John Wiley & Sons, Inc., Hoboken, New Jersey, 3rd edition, 2010.

[2] Uyemura, John Paul, *Chip Design for Submicron VLSI : CMOS Layout and Simulation*, Thomson/Nelson, Toronto, 1st edition, 2006.

[3] Wagdy, Mahmoud Fawzy, *Comparative ADC Performance Evaluation Using a New Emulation Model for Flash ADC Architectures*. IEEE Xplore, 1995.