DESIGN AND FABRICATION OF A 6-BIT CURRENT-MODE ADC FOR LIDAR AND HIGH-SPEED APPLICATIONS

By

Jason Silic

Bachelor of Science – Electrical Engineering

University of Nevada, Las Vegas

2015

A thesis submitted in partial fulfillment

of the requirements for the

Master of Science in Engineering – Electrical Engineering

Department of Electrical and Computer Engineering

Howard R. Hughes College of Engineering

The Graduate College

University of Nevada, Las Vegas

December 2020

Abstract

The design, simulation, fabrication, and testing of a 6-bit current-mode ADC is presented with related calculations. Capable of producing samples at greater than 100MHz, the proposed design should excel at measuring fast current transients such as those produced by arrays of avalanche photodiodes (APDs). An array of 63 current comparators feed digital logic gates, via "thermometer code," that produce a 6-bit output word that is driven off-chip by appropriate buffers.

The circuit was designed using Cadence Virtuoso design software. Simulated Differential Non-Linearity (DNL), with an 8µA reference current, is 1.656 LSB at the worst transition involving a scale change. Excluding scale changes (where different current mirror sizes are used for different sections of the input range) the worst DNL is 0.229 LSB. Worst Integral Non-Linearity (INL) is -2.29 LSB. Simulations indicate delay for digital logic to encode results from comparators into a 6-bit word is less than 1ns. Output buffers are designed to drive 10pF load capacitance at a frequency of at least 100MHz.

Fabrication was in a 180nm SiGe BiCMOS process. The complete design is 830µm x 395µm for a layout area of 0.328mm^2. Layout techniques and strategy are described, including isolation structures, power delivery, and standard cells for digital design. Metal routing and shielding techniques are described. Integration with a test Printed Circuit Board (PCB) is described. The ADC is characterized and test results are presented. Oscilloscope measurements indicate buffers are capable of driving signals at frequencies in excess of 100MHz. The complete system power consumption can exceed 50mW. Conditions that affect power consumption are investigated. Methods to stabilize power delivery and associated challenges are described as well as possible future design mitigations. The proposed ADC is demonstrated to have a response sufficient for high-speed operation.

Acknowledgements

I would like to start by thanking committee chair Dr. R. Jacob Baker, a cynosure for many engineering students here, who has been a luminous example of a hard-working professor. Yet for all his lecturing skill an even more memorable quality may be the genuine care he shows for the growth of the students he teaches. Dr. Baker is also a noteworthy purveyor of desiccated humor.

I have deep gratitude for the other members of the committee as well. Dr. Robert Schill is a high-performance professor who has opened my eyes to the austere science of electromagnetics. I am indebted to his efforts teaching this (arguably) hardest of the hard sciences as well as his support as graduate coordinator at the beginning of my graduate career. Dr. Biswajit Das is another professor whose classes are always enjoyable. I have only touched the edges of nanotechnology and the luminous depths of research in that field but his passion and expertise there are contagious. I particularly want to thank Dr. Monika Neda as the Graduate College representative. It cannot be easy to sit on a committee for one of those weird engineering students! From my experience Dr. Neda has been a keen teacher and I am grateful for her time here.

There are many others from the UNLV community who have helped me so much over the years. Jennifer Reff has answered so many questions guiding me through various processes. Dr. Greg and Dr. Zhu have always been great to work with in the laboratories. I would like to thank Dr. Henry Selvaraj as the current graduate coordinator for help with forms and mishaps.

My fellow students in Dr. Baker's group have been helpful. I would especially like to thank Sachin Namboodiri for giving me insight and tips on navigating the graduate student life. James Skelly and David Santiago (who is also great at decorating the office and making school fun) have helped me think through homework and been friends as well. Jazmine Boloor did timely work designing the test PCBs for this project, and I appreciate that effort. The inimitable Angsuman Roy has provided given me much

valuable advice and encouragement. Thank you Angsuman for your support with my project and other RA projects!

I would also like to acknowledge my family and friends. My Dad has always been supportive of my engineering education; without him I would probably not have made it this far. My Mom, siblings, and friends have been a consistent source of joy (very important for EE). Finally, I would like to thank my Lord Jesus, without whom none of these accomplishments would endure.

Table of Contents

List of Tables

List of Figures

1. Introduction

This thesis will describe the design and fabrication of a fast current-mode comparator primarily intended for use with an array of avalanche photodiodes (APDs) in a lidar application. The high-speed current mode comparator provides an effective low-impedance interface with a six-bit digital output. A quick response to input pulses in timeframes less than ten nanoseconds is demonstrated in simulations with drivers capable of an output sample rate of 100 megasamples per second.

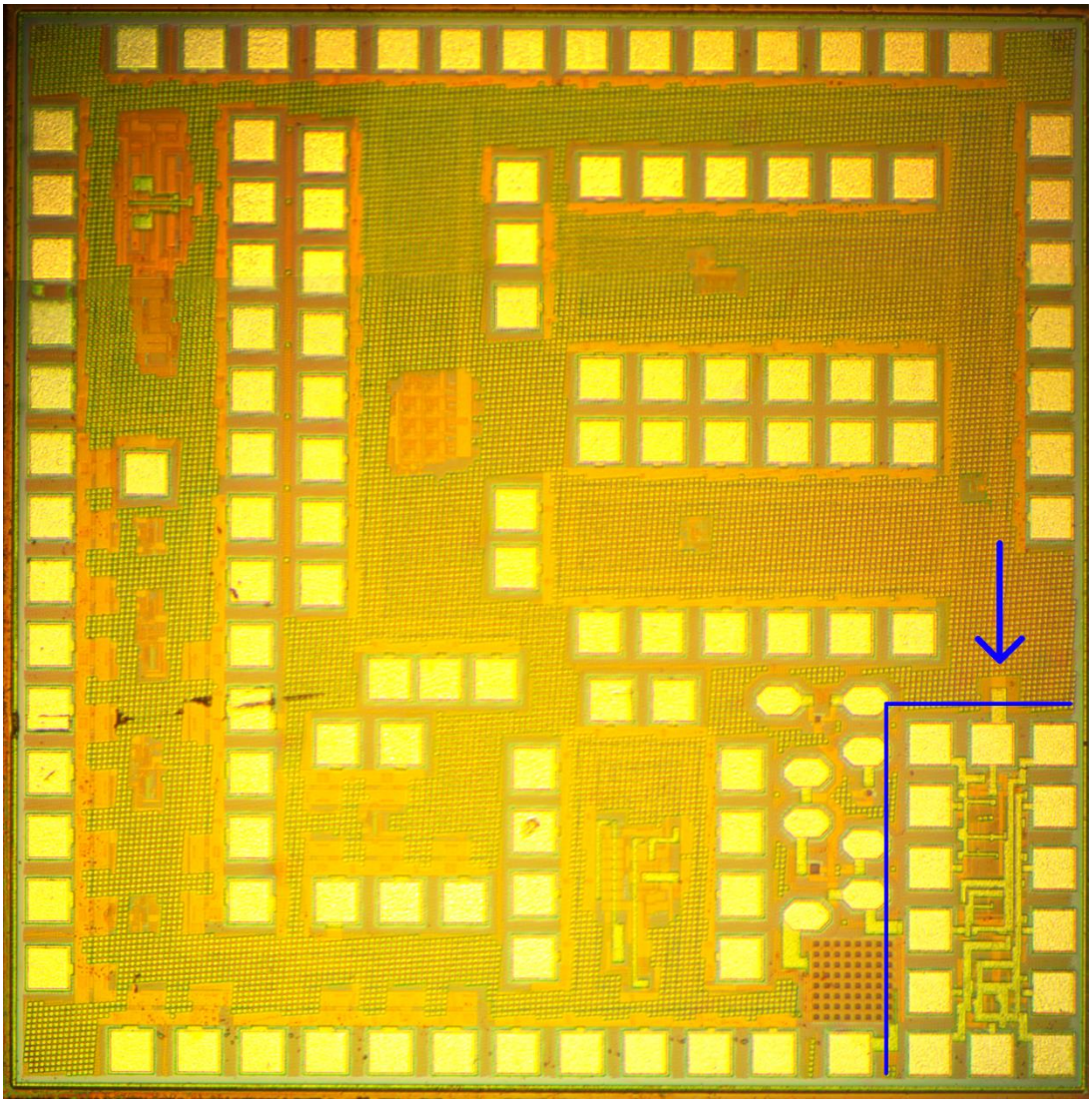

Figure 1 - Micrograph of the fabricated chip

The analog input is critical to the performance of this ADC and is based on [1]. We discuss the design procedure and evaluate the performance of various designs. The analog front end sends a 63-level thermometer code to the digital circuitry. A digital circuit processes this information with a propagation delay of less than a nanosecond and produces a six-bit digital output. The design can respond in simulations to current pulses in timeframes below 10ns. In a lidar application this can be used to determine the time-of-flight of a laser pulse and thus determine range (described in [2]), similar to how radar works. This is described in more detail in the application section below.

The design was sent to TowerJazz for fabrication using the SiGe BiCMOS SBC18HA technology node which has a 0.18µm (180nm) process size as noted in [3]. The tape out happened in mid-January and the chips were finished in April 2020. Testing was performed in October and November of 2020. Figure 1 above shows a composite image of the die. The indicated area in the lower right corner is the circuit discussed in this thesis. The rest of the die is used for other research projects.

2. Applications of a high-performance ADC

The primary application envisioned for this project is as a detector for a lidar sensor. Lidar is similar to radar and sonar in that it sends out energy and detects that energy reflecting from the environment. In the case of sonar the energy is in the from of a physical wave while radar and lidar use electromagnetic energy. Radar obviously uses much lower frequencies compared with the visible wavelengths of lidar.

In Figure 2 below we show the basic operation of lidar. A laser sends out a short pulse of light that reflects off objects in the environment. Light travels approximately as fast in the atmosphere as it does in a vacuum, at 3.0 x 10^8 m/s. This means it travels about 3 meters in 10 ns. The rock in Figure 2 is at a distance of 6m from the laser and detector. The round trip distance is 12m and a reflection is detected after 40ns. The figure also shows a smaller reflection in the output current graph from the tree at 60ns.

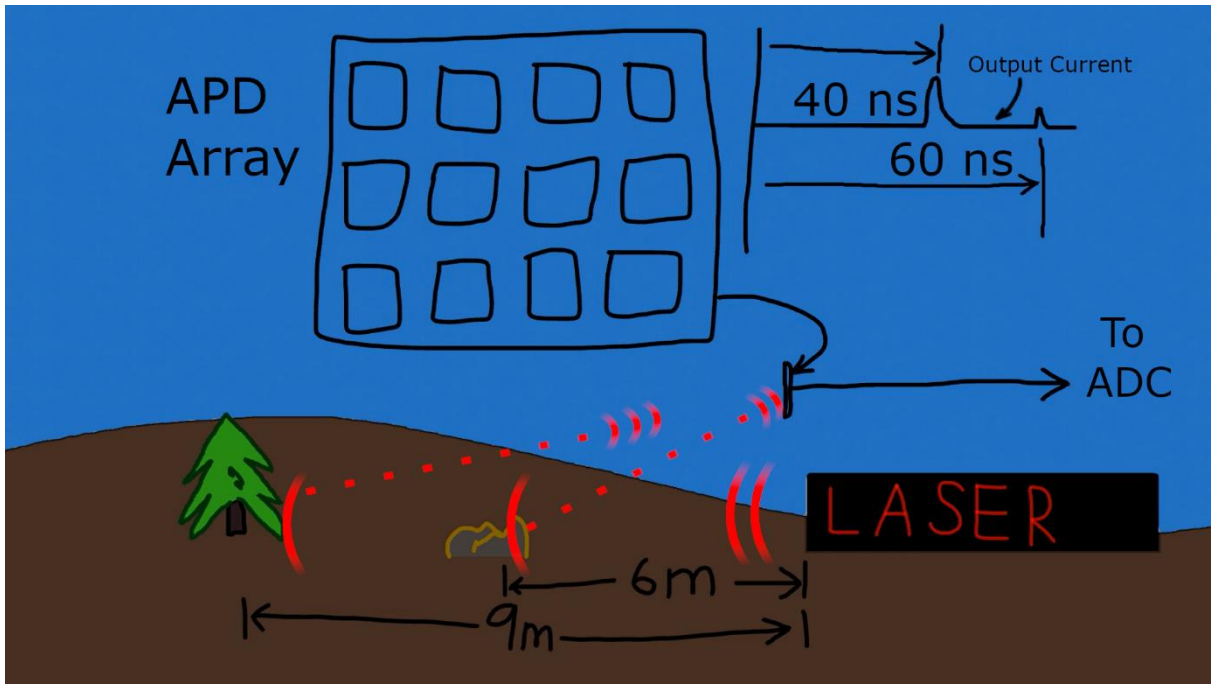

Figure 2 – Illustration of a lidar system

The amplitude of the reflected signal can vary depending on the distance, albedo, and surface orientation of the reflecting object. The advantage of a six-bit digital output is the ability to differentiate between different input currents corresponding to the strength of the returning signal.

In the illustrated configuration the current input is provided by a large array of Avalanche Photo Diodes (APDs) connected in parallel. A large voltage is applied in the reverse direction across the diodes. Figure 3 shows a schematic of this array of diodes. In a Lidar application a laser pulse is sent out and reflects off a target. The returning light causes some of the APDs to avalanche and current to flow. We expect the actual number of diodes to be approximately 64 so we can theoretically determine how many APDs are breaking down with our six-bit output although, as noted below, we will see that the precise number of states we can detect is $2^6 - 1 = 63$. This is due to one state being reserved to indicate "no current detected." This current is directly correlated to the intensity of the light hitting the sensor. If we can measure this current in timeframes under 10ns and the APDs have a sufficiently rapid recovery time we will have an effective sensor that can determine range and other data from the incoming light pulse.

The voltage at the input to the comparator should not increase too much or the diodes will not have a sufficient reverse voltage drop to avalanche correctly. Problems with too much voltage across the diode include extended recovery time, where the diode takes too long to stop avalanching once it has been triggered. The voltage across the diodes needs to be relatively precise. Our design uses a current mirror to provide a low impedance input and a stable voltage drop across the APDs.

Because we are trying to time pulses of light to measure distance any error on even the nanosecond level results in significant errors. In one nanosecond a photon will travel approximately 30cm. A fast and consistent response to changes in the input current is therefore critical to the success of this design.

A typical application of the proposed design is to measure extremely rapid fluctuations of current in a Lidar sensor. In Lidar, laser light pulses are reflected off objects to determine the distance, which is easily calculated based on the speed of light in the atmosphere. If your measurement of the round-trip time is off by 1 nanosecond, your calculated distance (accounting for departing and returning time) will have an error of about 15 cm.

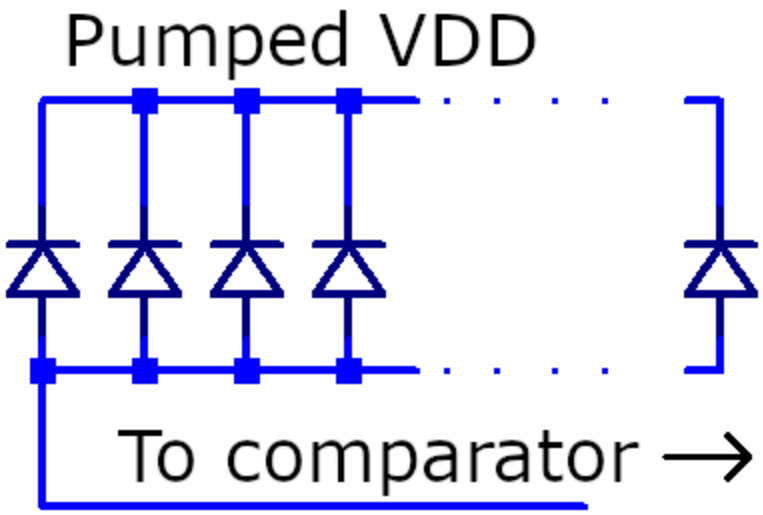

Figure 3 – Diagram of font-end APDs

The ultimate goal of the circuit in this application is to be able to precisely measure the timing and amplitude of returning laser pulses hitting the avalanche photodiode. This requires measuring the current produced by the APD. If we can quickly response to rapid pulses of current then we can accurately infer the amount of incoming light.

This design could also be used in other applications requiring a high-performance ADC. The input current could come from a photodiode in a communications context. Ideally the application would make use of the precision offered by the six bits of the converter. For example, a wide range of values could be detected at a sensor with a current mode output. Our primary concern, however, is developing the circuit with the specific application of lidar in mind.

3. Previous research on current-mode lidar detectors

There has been significant research in the field of lidar over the past decades, yet comparatively little work seems to have been done with multi-bit continuous-time current-mode ADCs. In the next section we note our work is based on a previous design in [4]. Most of the literature surveyed uses single bit outputs although designs such as [8] do utilize multiple channels. The output from multiple channels is equivalent to the information contained in a multi-bit output. If there are *n* output channels then they can be aggregated as a word with:

$$\#bits = \text{floor}(\log_2 n) + 1$$

Therefore a 3-channel ADC can represent all its states in in a 2-bit word. The "+1" in the above equation is due to reserving one state as the "zero" state where all channels are inactive.

Many of the examined designs use a Transimpedance Amplifier (TIA) or a variation as an input stage. One popular design is found in [9] and focuses on using the TIA with a Silicon Photo-Multiplier (SiPM) input stage in applications like PET scanners for medical imaging.

An unusual approach is taken in [10] where a programmable TIA gain is used to increase the linearity of the input range. The current input from an APD or SiPM can have a large dynamic range and this enables increased linearity in the TIA. This design also incorporates a peak detector circuit and appears to have good bandwidth and power consumption figures.

Yet another TIA design is presented in [11] for lidar (there referred to as "LADAR") applications. This paper does note that an array of TIAs are used but focuses on the design of a single channel. No papers were found that presented a complete monolithic design for reading the input current and converting it to a digital output. This is understandable as the input stages are a critical part of any complete design.

The current research presented in this thesis will also present the design of a current comparator, while adding the complexity of the array of 63 comparators and the associated logic. Complex effects arise when multiple circuits are working together. The comparators may interfere with each other in a way that is difficult if not impossible to simulate.

Therefore we conclude the fabrication and testing of the circuit is important to validate design principles and explore the performance of the design under conditions closer to real-world use cases. We will therefore devote a significant portion of this research to testing the design and examining experimental results noting that no research was discovered that presented such a complete system overview.

4. Selecting the comparator design

The design of the circuit involved two parts. First, an analog frontend is used to detect 63 individual levels of current (the 64th value available from a 6-bit word is used to indicate no current detected, i.e. zero). The thermometer code (explained shortly) is then converted to a conventional 6-bit binary output with appropriate digital circuitry.

The analog input is essentially a current-mode comparator as we are measuring the current coming from the APD. The proposed circuit is an expansion of an earlier design by Vikas Vinayaka in [4]. That design mirrored the input current 16 times and compared that with 16 reference currents to produce a 4-bit output. To produce the reference current an NMOS current mirror with 3µm width is used for the first stage. Subsequent stages use the reference voltage on the gate with linearly increasing width to generate multiples (2x, 3x, etc.) of the reference current. For example, the second stage uses 6µm width to ostensibly sink twice the reference current.

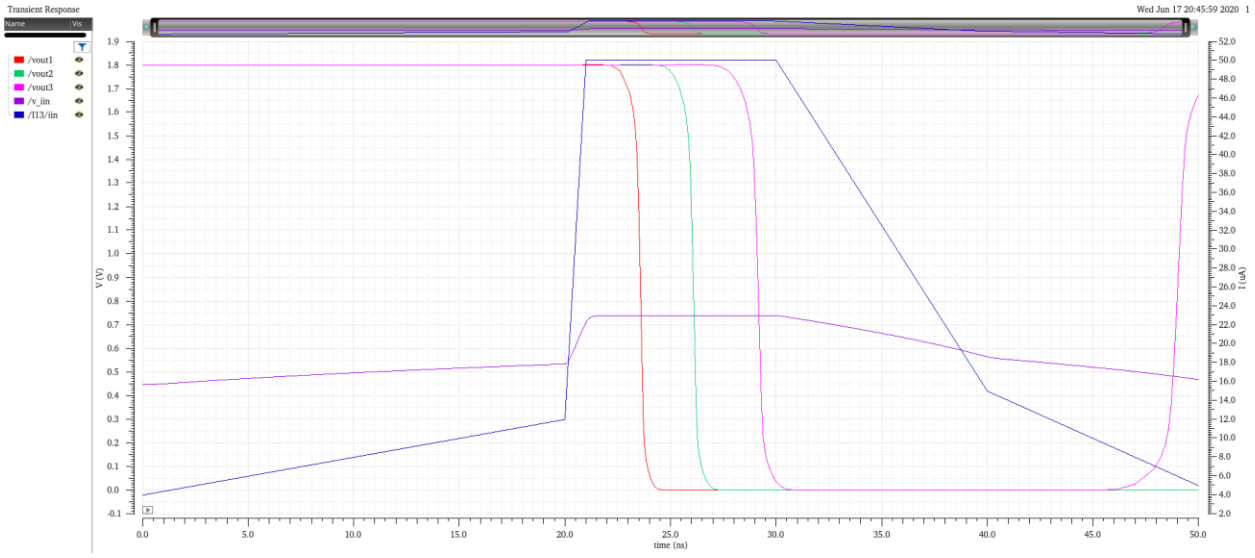

Figure 4 – Slow response of original design

In Figure 4 we see the drawback of the simple “current mirror” comparator, which we explore in more detail in a later section. Note how the input current pulse (dark blue) of 50µA should activate all

three bits, yet the third bit (vout3) has a delay (measured from the midpoint of the transition) of almost 9ns (29.1ns – 20.4ns). Even the first transition takes about 3.3ns. Note that this does include propagation through an inverter (which adds minimal delay), resulting in the inversion seen where the signals transition low when current goes above threshold.

This design was developed with consideration for previous research. Several papers were examined in the search for a faster current comparator. The earliest paper is [5], written by Träff in 1992, and is frequently cited in later research on current-mode comparators. It is interesting to note that A/D conversion was an important application then and is also the subject of this current thesis. It appears that far from being a "solved problem" the engineering world is still ready three decades later for a faster, smaller, and cheaper comparator with low power consumption.

We find that [2] first describes shortcomings with the current mirror comparator. The basic configuration is shown in Figure 5. Two current mirrors are in series and the voltage at Vout will go up if the comparison current through the PMOS is larger than the current being sunk by the NMOS reference current mirror. The output node has high impedance, leading to poor performance at higher frequencies when a capacitive load is present. Output buffers provide this load with their gate capacitance. Delay is cited as about 10ns for a 2µm gate length in the paper's simulations.

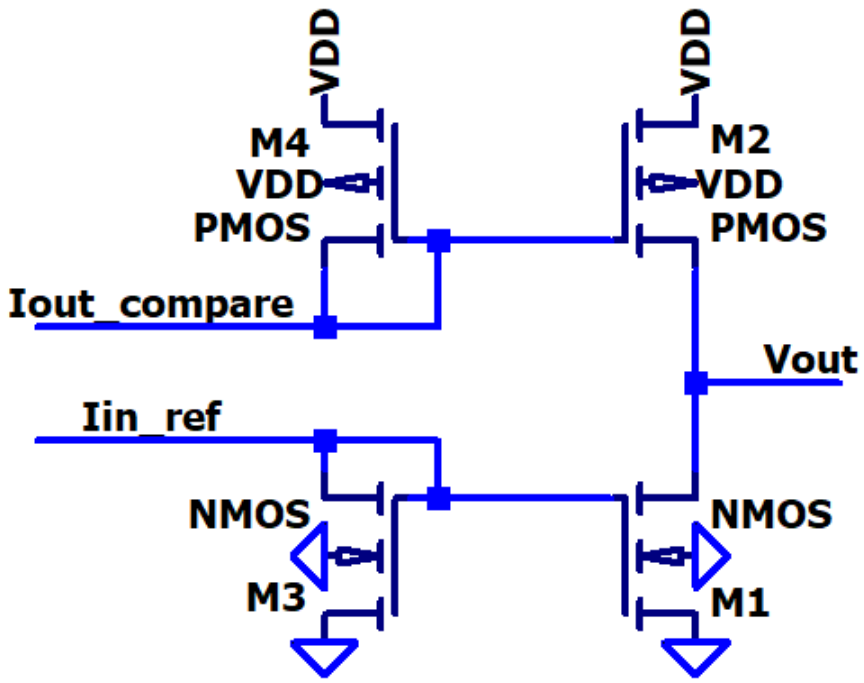

Figure 5 – Basic Current Mirror Comparator

The paper's proposed design tries to reduce the impedance of slewing nodes. The output node needs low impedance for better high-frequency performance when driving capacitive loads such as buffers. A source follower stage applies feedback to the input node and reduces input resistance. Finally, a single current input is the result of subtracting the input current from the reference current. The five transistors on the left of Figure 6 are these current mirrors.

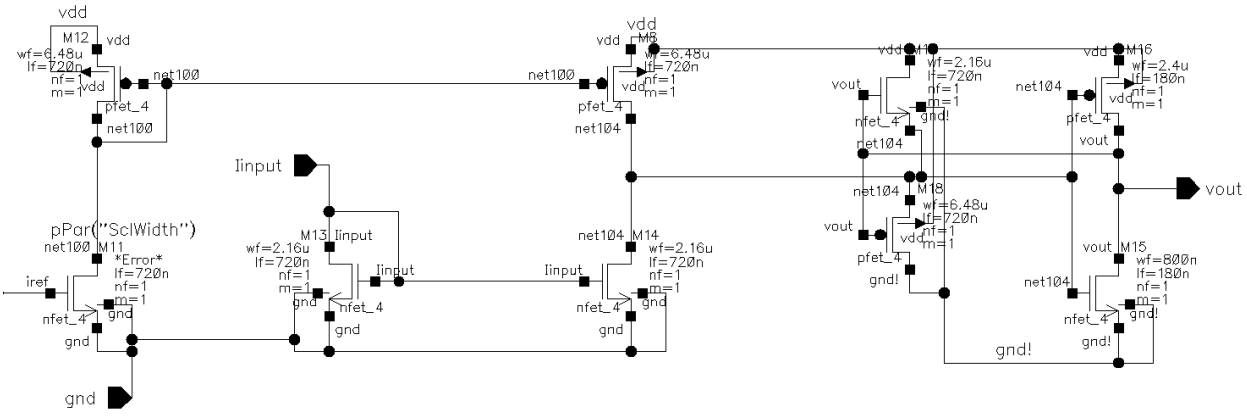

Figure 6 – Schematic of Träff's comparator

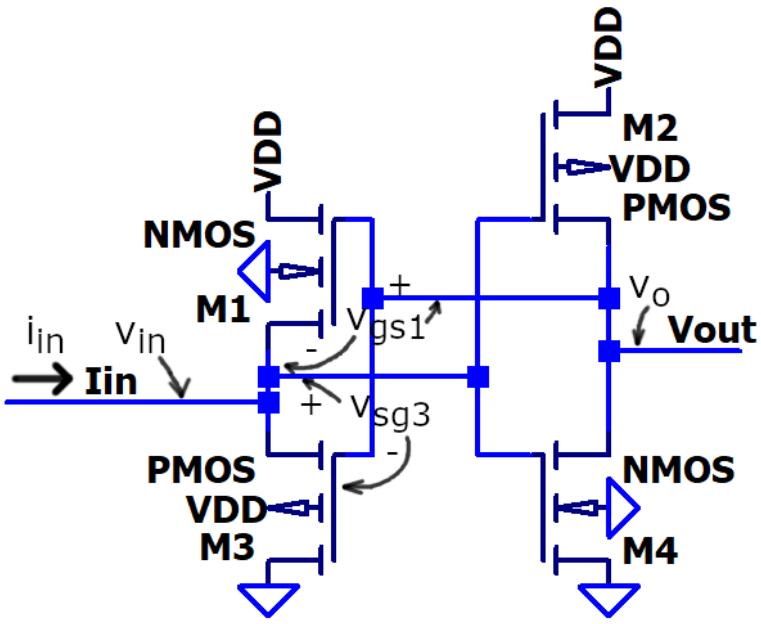

Figure 7 – Core transistors of Träff's comparator, annotated

We now discuss the calculation of the input resistance for Träff's comparator. The paper gives this as approximately $1/g_m$. Consider Figure 7 with marked voltages for the following derivation. We will assume that the inverter is biased at the switching point and provides a relatively low amount of gain, here for convenience we give as unity so that $v_o = -v_{in}$ for small-signal analysis. We ignore body effect in the source follower transistors as well. The input resistance, $R_{in} = v_{in}/i_{in}$. Given the drain currents of M1 and M3 as, respectively, i_{d1} and i_{d3} we write:

$$i_{in} = -i_{d1} + i_{d3}$$

In saturation we can write the drain currents in terms of v_{gs1}, v_{sg3}, and g_m, assuming transconductance of PMOS and NMOS are similar.

$$i_{in} = -g_m v_{gs1} + g_m v_{sg3} = -g_m(v_o - v_{in}) + g_m(v_{in} - v_o)$$

Letting $v_o = -v_{in}$:

$$i_{in} = -g_m(-v_{in} - v_{in}) + g_m\big(v_{in} - (-v_{in})\big) = 4v_{in}g_m$$

$$R_{in} = \frac{v_{in}}{i_{in}} = \frac{1}{4g_m}$$

This result generally agrees with the $1/g_m$ input resistance given in the paper but note this is only an estimate. There will be a scaling factor depending upon the specific sizes of the transistors. Another factor not taken into account by this analysis is capacitive effects which become important at higher frequencies.

Träff's current comparator has improved results, as demonstrated below, but another possibility is the design proposed in [1]. This design, published nearly a decade later, references Träff's paper and develops a more complex design with additional transistors. See Figure 8 below.

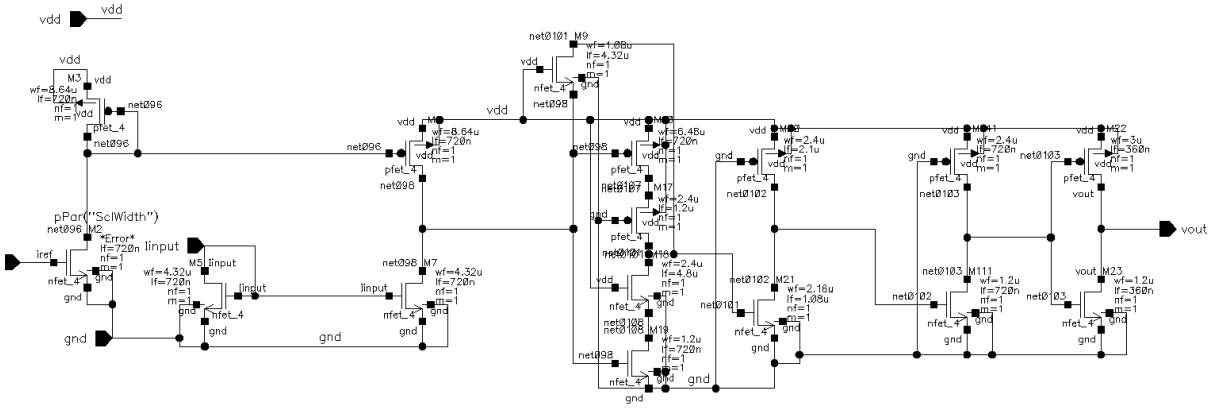

Figure 8 – Design from [1]

The paper discusses the problems with Träff's circuit and the need to avoid adding complexity and power consumption with additional current sources. This is important for our purposes as we have 63 comparators. The design proposed has a "CMOS complementary amplifier" with an NMOS providing resistive feedback. The input and output resistance are relatively small. This helps at the output when driving the resistive-load amplifiers. The low input impedance is also beneficial at the input so the

current mirrors can easily subtract the input and reference currents without large amounts of nonlinearity.

We now confirm the results noted in the paper for input and output resistance. In Figure 9 we present a simplified schematic with the two resistive central transistors removed (they are not included

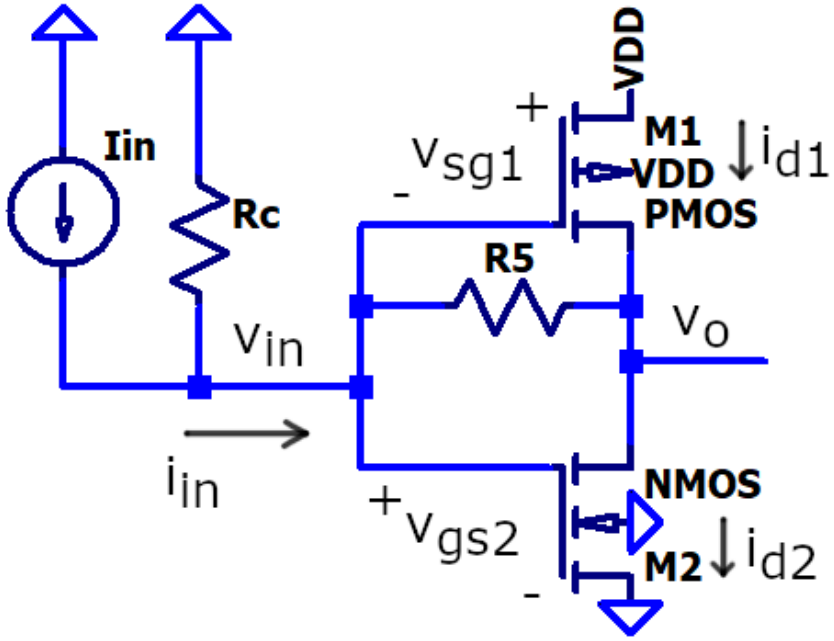

Figure 9 – Simplified schematic for calculating input/output resistance

in this calculation as suggested by the paper) and the feedback transistor represented as R_5. In this derivation we combine the output resistances of M1 and M2 into R_p, which equals $R_{o1}||R_{o2}$.

To calculate the input resistance, R_{in}, we apply a test input current, denoted above (Note this does not refer to the input current source and R_c which are used deriving the output resistance) as i_{in}. This current flows exclusively through R5 and then splits to the algebraic sum of the drain currents and current through the output resistances. Thus:

$$i_{d1} = g_{m1}v_{sg1} = -g_{m1}v_{in}$$

$$i_{d2} = g_{m2}v_{gs2} = g_{m2}v_{in}$$

$$i_{in} = i_{d2} - i_{d1} + \frac{v_o}{R_p}$$

$$i_{in} = v_{in}(g_{m1} + g_{m2}) + \frac{v_o}{R_p}$$

We can also write i_{in} as the current through R_5:

$$i_{in} = \frac{v_{in} - v_o}{R_5}$$

$$v_o = v_{in} - i_{in}R_5$$

With this substitution for v_o in the previous equation and then solving for $R_{in}=v_{in}/i_{in}$:

$$i_{in} = v_{in}(g_{m1} + g_{m2}) + \frac{v_o}{R_p} = v_{in}(g_{m1} + g_{m2}) + \frac{v_{in} - i_{in}R_5}{R_p}$$

Multiply both sides by R_p and group terms of i_{in} and v_{in}.

$$i_{in}(R_p + R_5) = v_{in}(R_p(g_{m1} + g_{m2}) + 1)$$

$$R_{in} = \frac{v_{in}}{i_{in}} = \frac{R_5 + R_p}{1 + R_p(g_{m1} + g_{m2})}$$

For the output resistance we must also consider the resistance of the input current source. As noted in Figure 9 we use R_c to represent this, following the convention of the paper. A test current source, i_T, is applied to the output which splits into the drain currents and a current back into the input source.

$$i_T = i_{d2} - i_{d1} + \frac{v_o}{R_p} + \frac{v_o}{R_5 + R_c} = v_{in}(g_{m1} + g_{m2}) + \frac{v_o}{R_p} + \frac{v_o}{R_5 + R_c}$$

Noting that v_{in} can be written in terms of v_o from the simple voltage divider formula, it is possible to rewrite the above and group factors of v_o together.

$$v_{in} = \frac{v_o R_c}{R_5 + R_c}$$

$$i_T = \frac{v_o R_c}{R_5 + R_c}(g_{m1} + g_{m2}) + \frac{v_o}{R_p} + \frac{v_o}{R_5 + R_c} = v_o\left(\frac{1}{R_p} + \frac{1 + R_c(g_{m1} + g_{m2})}{R_5 + R_c}\right)$$

$$i_T = v_o\left(\frac{\frac{R_5 + R_c}{R_p} + 1 + R_c(g_{m1} + g_{m2})}{R_5 + R_c}\right)$$

Finally, we write the output resistance. Our result agrees with the paper.

$$R_{out} = \frac{v_o}{i_T} = \frac{R_5 + R_c}{1 + R_c(g_{m1} + g_{m2}) + (R_5 + R_c)/R_p}$$

We note [1] argues that generally $R_5 << R_p$ and R_c, $(g_{m1} + g_{m2})R_p >> 1$, and thus R_{in} and $R_{out} \approx 1/(g_{m1} + g_{m2})$. As an example of what we can expect we run a simulation on a PMOS (3.24u/540n) and NMOS transistor (1.44u/540n). Both transistors have 900mV V_{ds} and we sweep the gate voltage from 800mV to VDD (1.8V) for the NMOS and from 1V to 0V for the PMOS. Our simulation time is 1µs leading to a linear change in gate voltage of 1V/µs. Taking the derivative of the drain (or source) current we can calculate g_m for these parameters.

Looking at the simulation results in Figure 10 below we note that as v_{gs} increases we see more transconductance. To be conservative let's take the smaller values from this range. For the NMOS a value of 30 indicates 30µA/µs. We divide by our constant ramp of 1V/µs we get g_{mn} = 30µA/V. Similar logic for the PMOS leads to a value of g_{mp} = 90µA/V. Our formula above will therefore give approximate input and output resistance of

$$R_{out} = R_{in} = \frac{1}{g_{mn} + g_{mp}} = \frac{1}{30\mu\frac{A}{V} + 90\mu A/V} = 8.33k\Omega$$

We will check this value in simulations with our final design.

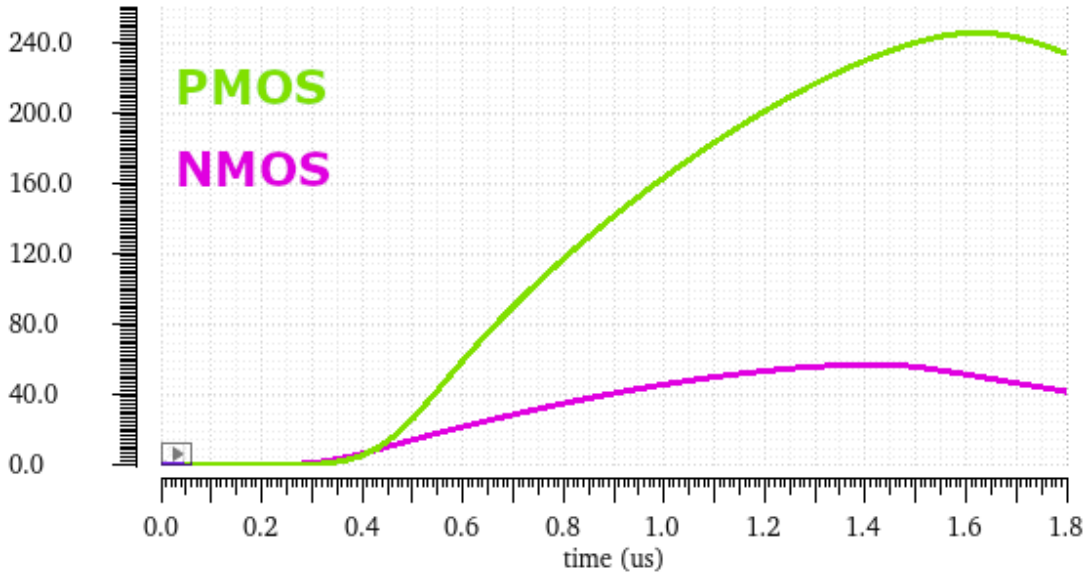

Figure 10 – Drain current vs steady v_{gs} ramp (1V/µs)

5. Refining the comparator design

The performance of the comparators is the most critical aspect of the design. Any deficit here will have a deleterious effect that cannot be ameliorated by heroic efficiency gains in the digital logic or output stages. Note that matching is less of a concern so we will be trying to use smaller and faster transistors. Another concern is layout area as we have 63 comparators and would like to keep things compact if possible. For the following discussion refer to Figure 11 for the specific labels.

The paper gives suggested values for the length and width of the transistors in the comparator. These were modified to produce a more compact layout with smaller transistors. The original paper is from the year 2000 and is based on an older process. By making many of the transistors smaller simulations show an increase in speed as seen below. We are able to run parametric simulations to determine specific sizes for our transistors.

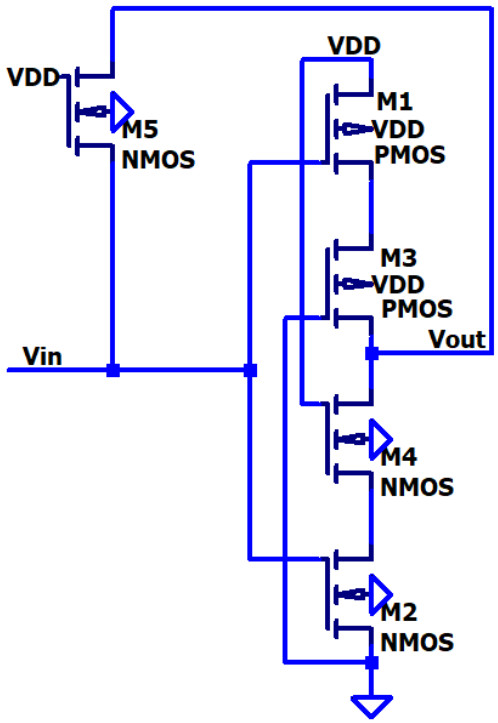

Figure 11 – Core Comparator Transistors

As a starting point we multiply the suggested sizes given in the paper by a factor 0.6 to produce reasonable values for our process. Technically the TowerJazz process has a 180nm minimum gate length but we generally use much longer sizes in the analog circuitry to improve matching. A gate length around 720nm is what we have used for many of the current mirrors although in some cases we even use 540nm (3x minimum length). For M1 and M2, which are the transistors producing gain, we multiply the paper's 1.2µm by 0.6 to get 720nm. Some dimensions were not directly followed; M1's width is supposed to be 7.5µm * 0.6 = 4.5µm and M2's should be 2.4µm * 0.6 = 1.44µm but in an attempt to balance the switching point of the "inverter" the PMOS is made stronger with a width of 6.48µm and the NMOS weaker at 1.2µm. The relative strength of NMOS and PMOS transistors is highly dependent on the process used so this is not a surprise but we will return to this in a moment.

We use this circuit from Figure 8 as a baseline so we can experiment with changes to various circuit parameters and determine if we have improved performance. for our first experiment we will decrease the sizes of M3 and M4. These transistors are used to reduce current flowing in the comparator (less power consumption) and if they are smaller they will have less parasitic capacitance. We now determine if this increases the speed of the amplifier by examining the simulation seen in Figure 12.

For the second circuit (vint2, voutBuffer2) we have decreased size of PMOS from 2.4µ/1.2µ W/L to 2.16µ/1.08µ. The NMOS goes from 2.4µ/4.8µ to 1.08µ/2.16µ. Note the NMOS has a W/L ratio 4 times greater than the PMOS. Total drawn active area (approximately proportional to capacitance between gate and channel) decreases from 14.4µm² to 4.67µm², a reduction of 68%.

For this set of simulation we use a uniform input. A steady 8µA current is used as the reference. The input current starts at 2µA for 10ns. Then the first ramp of 4µA/ns is applied from 10ns to 13ns for a final current of 14µA. At 20ns this abruptly drops back to 2µA, testing the comparators performance on a falling input current. Finally at 30ns a 10µA/ns ramp is applied for 3ns for a final current of 32µA. This

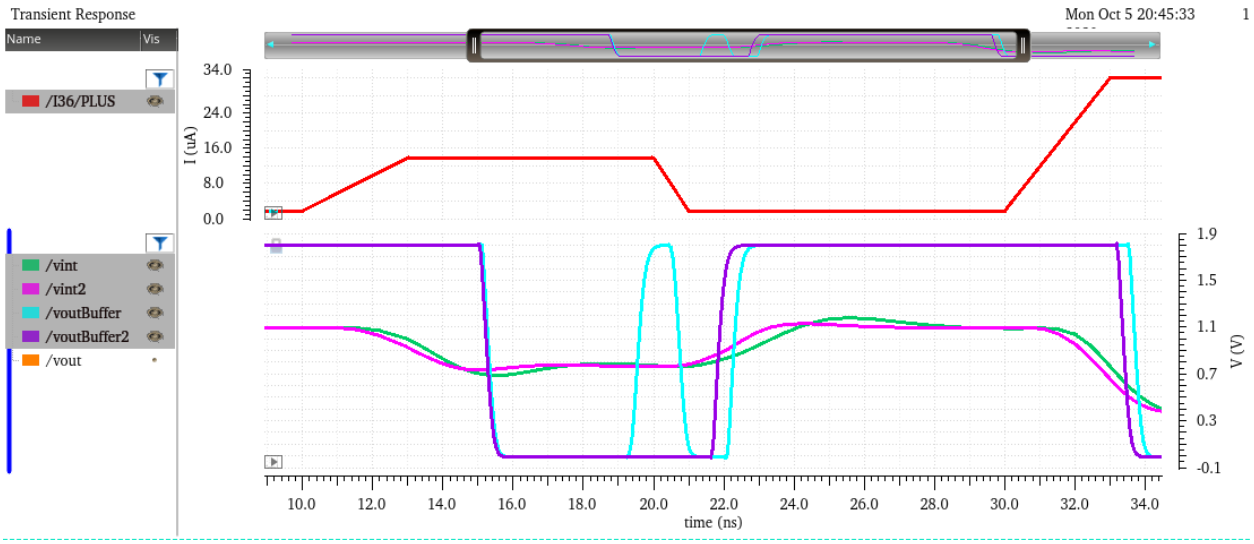

Figure 12 – Improving the speed of the comparator

is a faster ramp and will help determine comparator performance under conditions that may be closer to the current transients produced by APDs.

In Figure 12 we can see the input current as the red trace in the upper plot. In the lower plot we first examine the intermediate voltages, *vint* and *vint2*. Signal *vint2* is clearly quicker to respond and appears to have less oscillation. Approximate measurements indicate *vint2* is up to a half-nanosecond ahead of *vint*. One peculiar result here is the instability in the output *voutBuffer* in the first circuit. Our modification has not only increased the speed of the comparator it may have also helped stabilize it.

One issue we address now is the switching point of the comparator while keeping the change of smaller M3 and M4. Ideally it should change its output when the input current is zero. In other words, a net positive input current will produce one output and a net negative input will result in the complementary output. Note the fact that a positive input current produces a logical ‘zero’ output in these simulations is immaterial. A simple inverter will correct this in the digital logic.

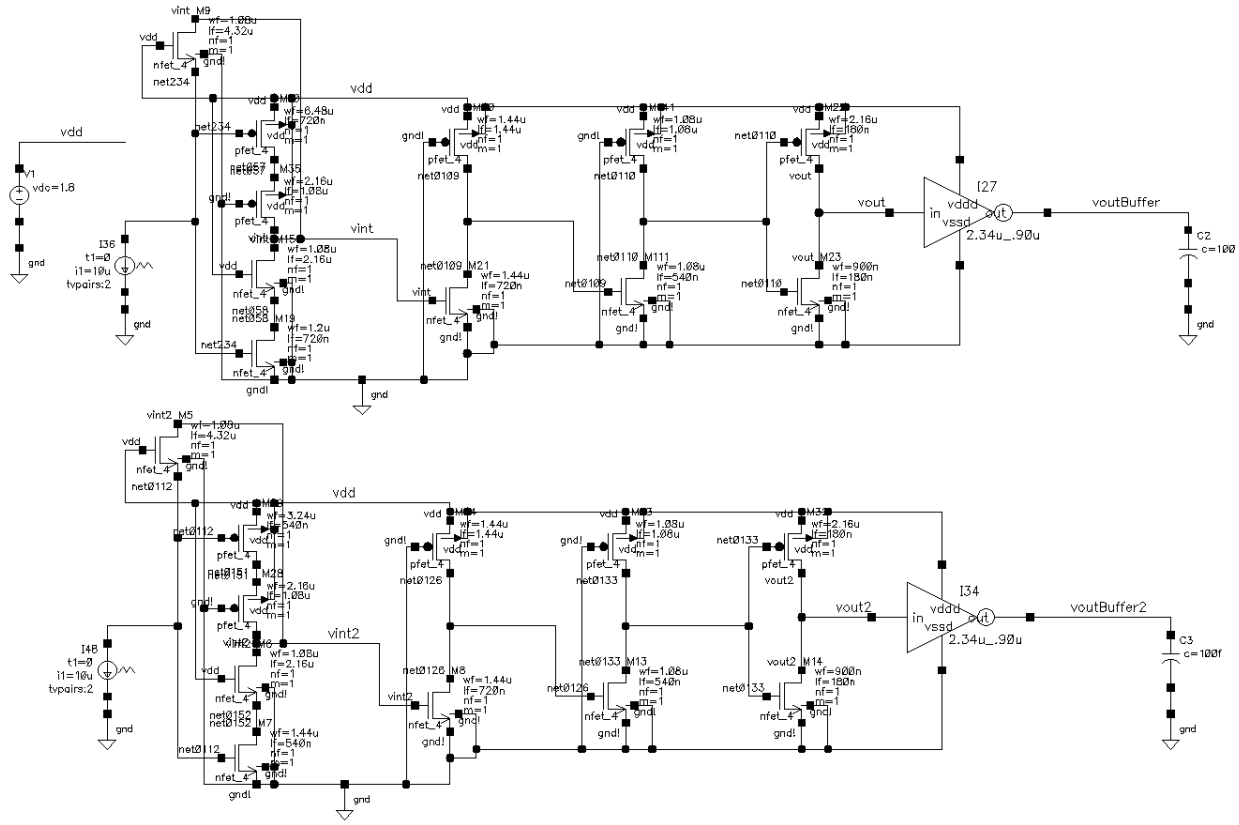

Figure 13 – Adjusting the switching point of the comparator (simulation schematic)

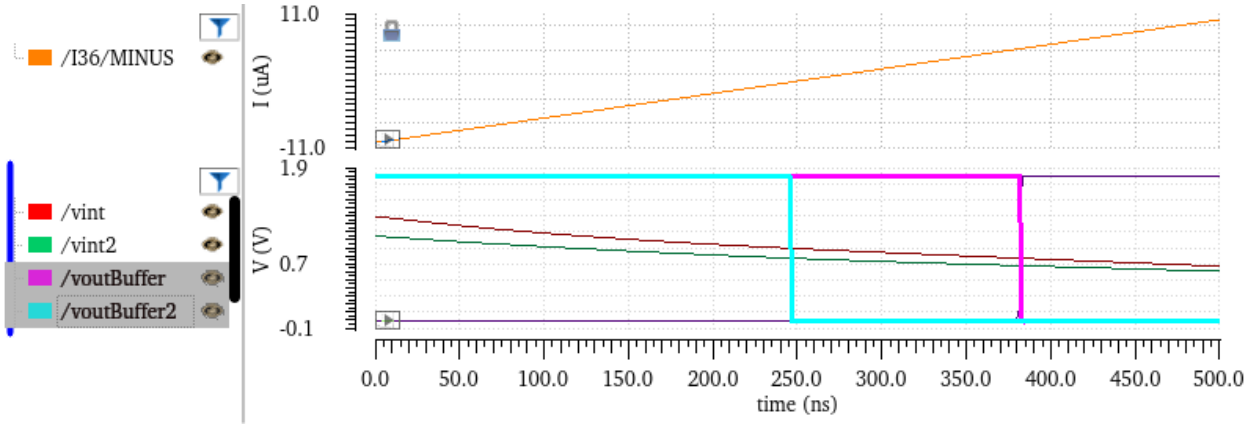

Figure 14 - Adjusting the switching point of the comparator (simulation result)

In Figure 14 we see that the first circuit switches at nearly 5.3µA of input current! To rectify this situation we note that we would like to lower the switching point of the 'inverter' formed by M1 and

M2. This is accomplished by increasing strength (proportional to W/L, also known as “Beta” of the transistor) of the NMOS and weakening the PMOS. Our updated sizes are shown in Table 1 below and decrease the PMOS ratio W/L from 9 to 6 and increase the NMOS ratio from 1.67 to 2.67. The result is to move the switching point almost exactly to zero current (in this sim about -150nA, which is negligible). Our final comment is on the reduced channel length from 720nm to 540nm. This change was mixed in with the others to further improve the speed of the transistors.

	PMOS (W/L)	**NMOS (W/L)**
Circuit 1	6.48µm/720nm	1.2µm/720nm
Circuit 2	3.24µm/540nm	1.44µm/540nm

Table 1 – Changing transistor sizes to modify switching point

The only major design choice remaining for this circuit is the size of the feedback transistor, M5 in Figure 11. In general, a smaller resistance will result in a smaller input resistance as the feedback current stabilizes the voltage at the input node. For example, if the input current goes up the voltage on the input node will increase, driving the output node down. The feedback from the output will again try stop the input node from rising. Given that (note: lowercase here denotes small signal approximation):

$$R_{in} = v_{in}/i_{in}$$

We note that a decrease in the swing of v_{in} will reduce the effective input resistance. This is also derived above in the formula for v_{in}. The drawback of a smaller resistance is that the output voltage does not swing as much and the output stage amplifiers may not detect the input change (depending on switching point) or may add additional delay.

To try to determine optimal values we run a parametric simulation. Based on our approximate scaling we set the width of M5 to 1.08µm and change the length to ten values from 1.08µm to 10.8µm with an increment of 1.08µm. By changing the length like this we are effectively changing M5's resistance. The results in Figure 15 indicate that a length of 3.24µm leads to the fastest response. (Note that only 5 representative lengths are shown in the figure for clarity.) Detailed information on response times is found in Table 2.

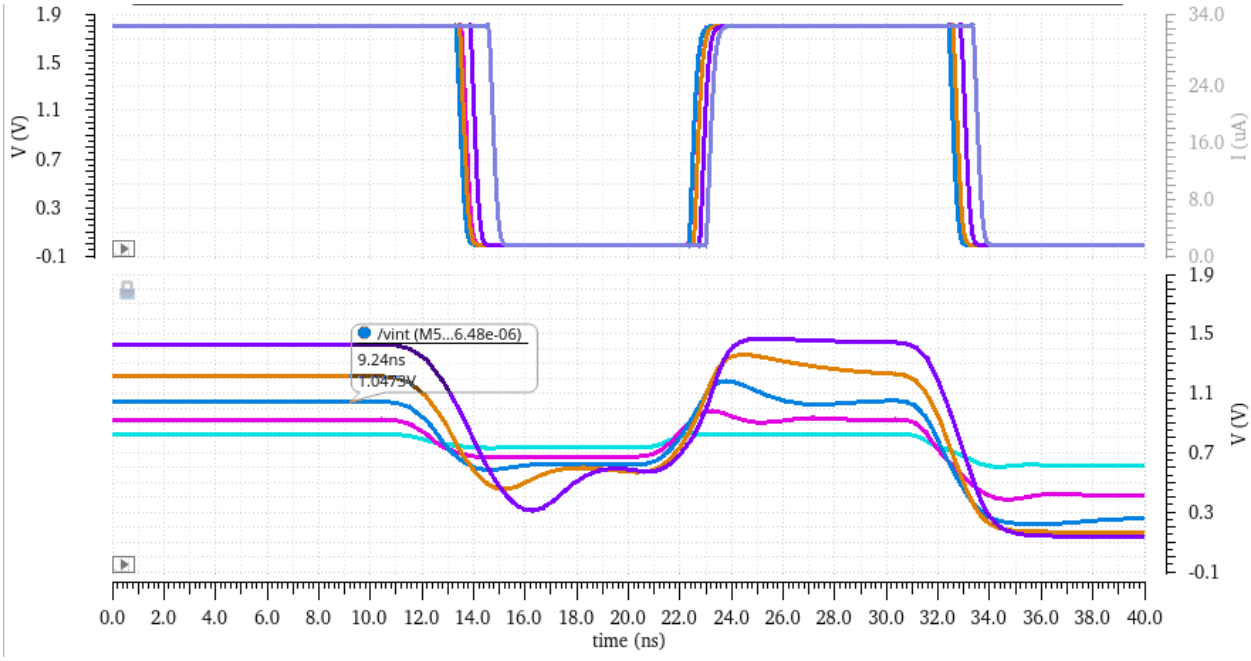

Figure 15 – Parametric simulation of M5's length

The first activation is when the comparator detects the current ramp from 2µA to 14µA in 3ns starting at 10ns (output goes low). The deactivate time is when the comparator output goes high again after current drops back to 2µA in 1ns starting at 20ns. Finally, the second activation is the time the output goes low following the input current ramp to 32µA in 3ns starting at 30ns simulation time. All times given (rounded to nearest 0.1ns) are when the buffer output crosses VDD/2 or 900mV.

Length of M5 (µm)	Time (first activation, ns)	Time (deactivate, ns)	Time (second activation, ns)
1.08	13.9	23.7	33.3
2.16	13.7	22.5	32.7
3.24	13.6	22.5	32.6
4.32	13.5	22.5	32.6
5.4	13.5	22.6	32.7
6.48	13.6	22.7	32.8
7.56	13.8	22.8	32.9
8.04	14.1	22.9	33.1
9.72	14.4	23.1	33.3
10.8	14.8	23.2	33.5

Table 2 – Response time of comparator depending on M5 length

The optimal length for M5 appears to be 3.24µm or 4.32µm. A length of 3.24µm is similar in speed to 4.32µm and will take up less layout space. Our final design uses a length of 3.24µm.

We can now develop simulations to test the input and output resistance of the comparator. Note that we will compare our calculations with the input resistance of the core transistors of the comparator, excluding the current mirrors that subtract the reference current from the input current. The input resistance of those mirrors is also an important question (the APD array sees this resistance) and results for that are presented as well.

Note that input resistance can be assumed to be the change in voltage divided by the change in current. A large input resistance will lead to a sharp rise in voltage at the input node as additional current is pushed in. Mathematically:

$$R_{in} = \frac{\Delta v_{in}}{\Delta i_{in}}$$

Figure 17 shows the voltage at and current into the input node for a "two speed" current test. A slow ramp of 4µA/ns at 10ns and a faster ramp of 20µA/ns at 30ns. Note that a small 1mΩ resistor is used to give the simulation engine a component to measure the net current entering the input node. A partial schematic excluding output transistors is shown in Figure 16.

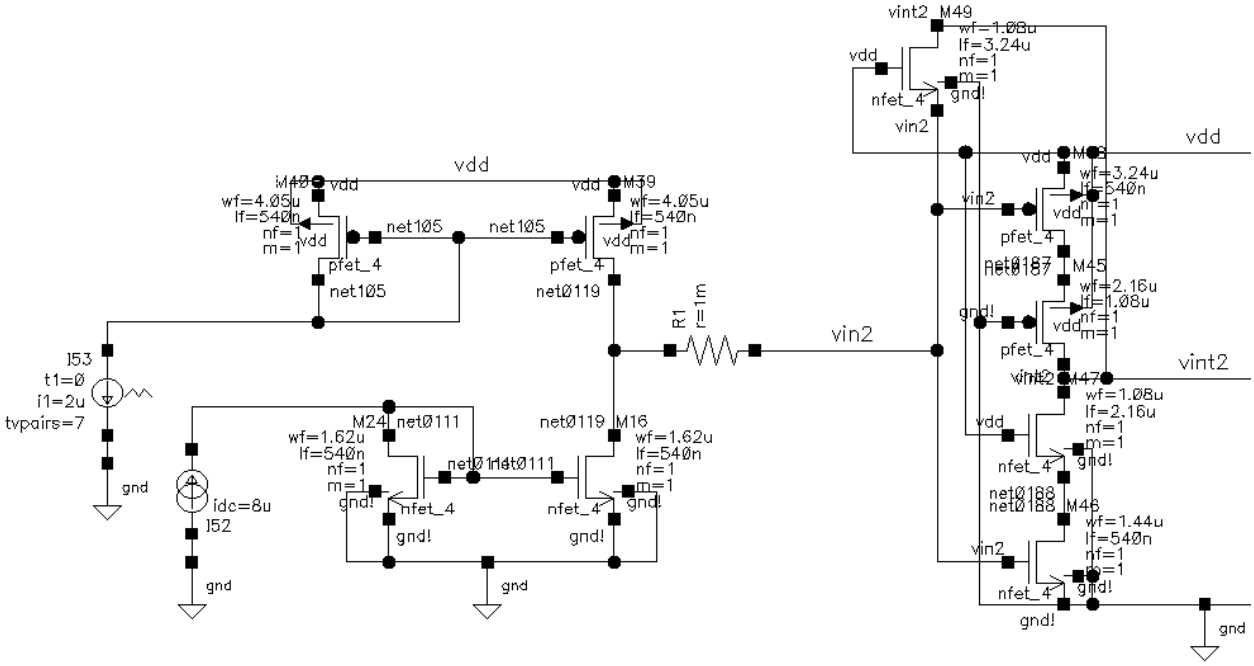

Figure 16 – Input resistance test schematic

Table 3 below lists voltages and currents before and after the two ramps. We can then easily calculate input resistance by taking the difference in the input voltage divided by the difference in the input current. We also run a simulation with a 100MHz sine wave current input from 0 to 20µA. Note that an input current of zero minus the 8µA reference current results in current being pulled out of the

comparator. These simulation results are displayed in Figure 18. For this sine wave a good place to measure input resistance is where there is a rapid rise in input current from -10µA to the peak value. This should result in a value for R_{in} that is probably a little too small as a little bit of that dip looks like it is due to capacitive effects and non-ideal behavior at 100MHz.

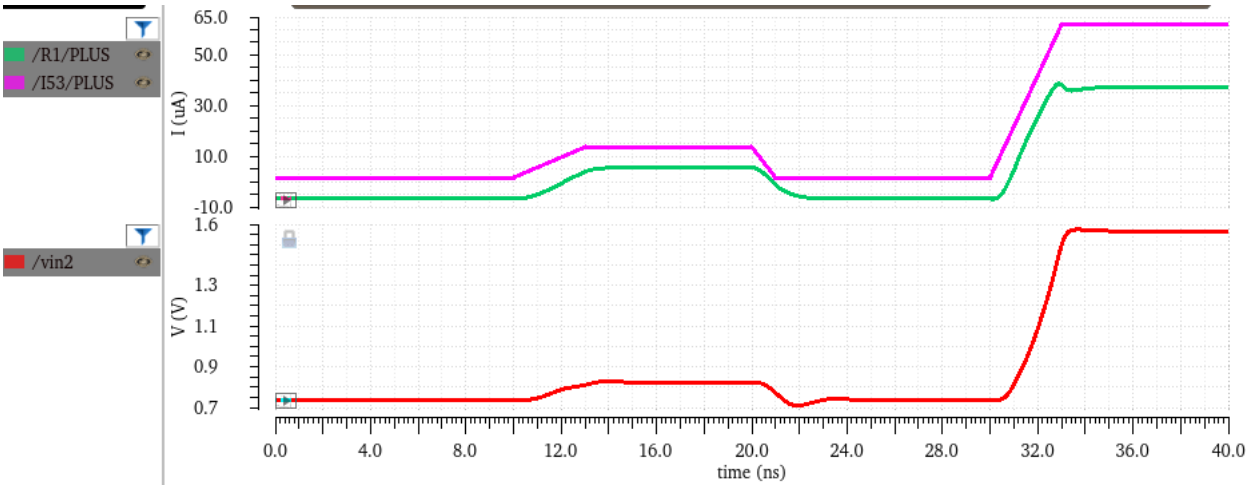

Figure 17 – Ramp input resistance test

Results clearly show the input resistance is dependent on the frequency or speed of the input signal. It appears that a larger and faster signal will result in a disproportionate rise in voltage. The second ramp test demonstrates this with an input resistance more than double the first (slower, smaller) ramp. Although there is quite a bit of variation these values agree quite well with our calculated value of 8.33kΩ.

	i_{in}, initial (µA)	i_{in}, final	v_{in}, initial	v_{in}, final	$\Delta v_{in}/\Delta i_{in}$	R_{in}
Ramp1	-6.17	5.88 µA	742 mV	833 mV	91mV/12.05µA	7.55kΩ
Ramp2	-6.17	38.7 µA	742 mV	1.573 V	831mV/44.87µA	18.5kΩ
Sine wave (100MHz)	-10.7	21.8 µA	713 mV	1.084 V	371mV/32.5µA	11.4kΩ

Table 3 – Data and results for input resistance

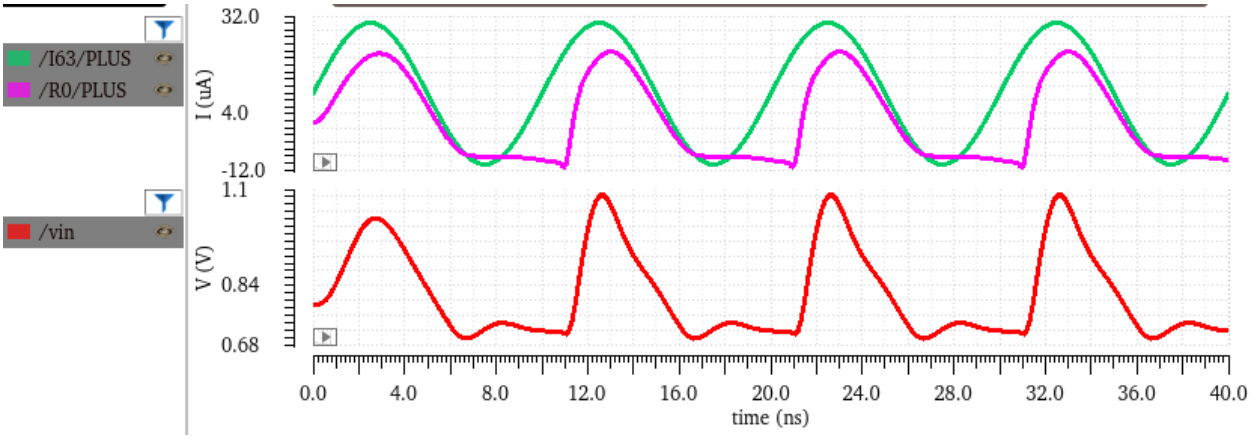

Figure 18 – Sine wave input resistance simulation

6. Operation and simulation of comparator

Now that we have our design, it is helpful to run simulations, to validate our design decision choice vs the other comparators. One critical measure of a comparator's performance is how fast it responds to input signals. One easy way to get a rough idea of performance in this area is to run an AC simulation. We note that there are significant problems with this approach. AC simulation uses the small signal approximation and is not ideal for circuits with a digital output like our comparator. It can perhaps instead give an intuition on how well the circuit performs at high frequencies. Note that additional buffer stages after the comparator can introduce additional gain as long as the comparator output swings around their switching point. It is also important for these AC simulations that the circuits are biased near where they have good gain.

Design	**Gain @ 100MHz**	**Gain @ 1GHz**
Basic Current Mirror	34k	1.7k
Träff's Comparator	298k	4.1k
Design based on Chen	3.8M	12.8k

Table 4 – AC simulation for various designs

Table 4 above lists some of the results. Gains to the primary output of the comparator listed, except for our design which has a few amplifier stages as seen in the schematic. Additional buffers often do not improve the response at high frequencies but again, our output is digital which can be difficult to understand in an AC simulation. One way to think about these results is to consider a 1µA input current (much more than this would occur during a transient event when the APDs break down). For the design in our chip the AC response at 100MHz would nominally lead to an output voltage amplitude of 3.8V

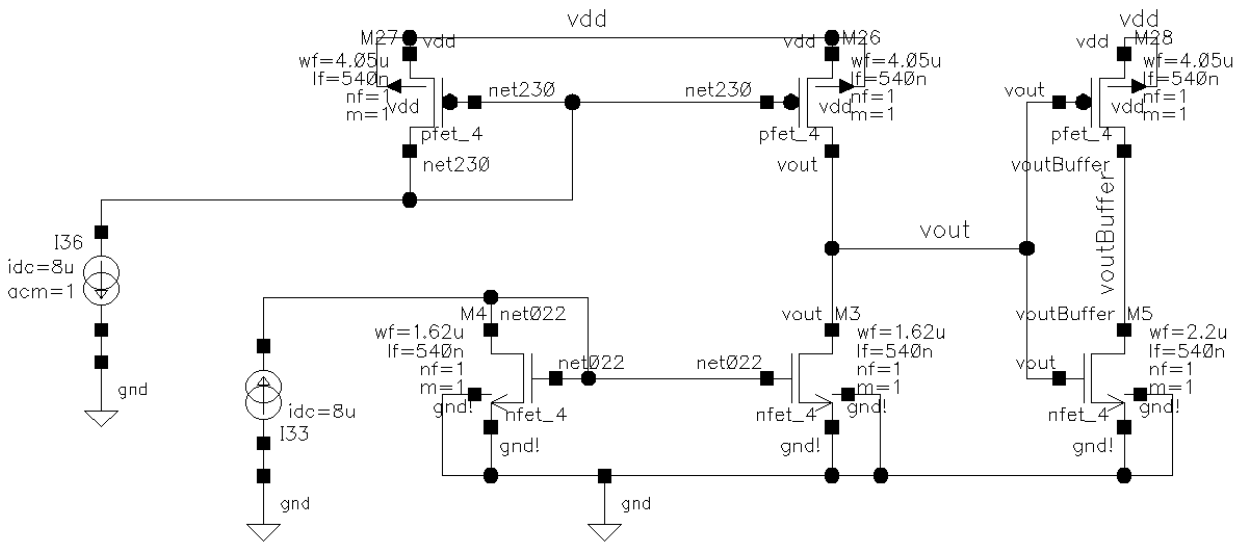

Figure 19 – Basic current mirror simulation

(clearly not realistic). With additional buffer stages even a small signal should be enough to create a decisive output signal.

Figure 19 shows the schematic of a basic current mirror comparator with a single inverter as a buffer on the output. In the simulation results in Figure 20 we note a roll off above 1MHz and overall low gain at higher frequencies.

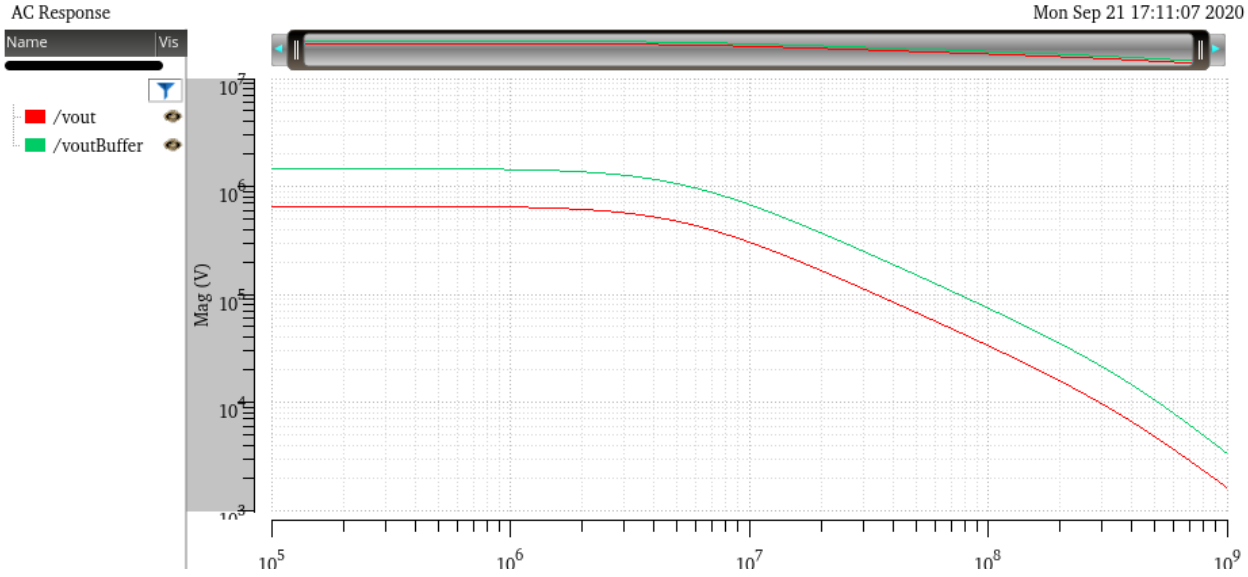

Figure 20 – Basic current mirror sim results

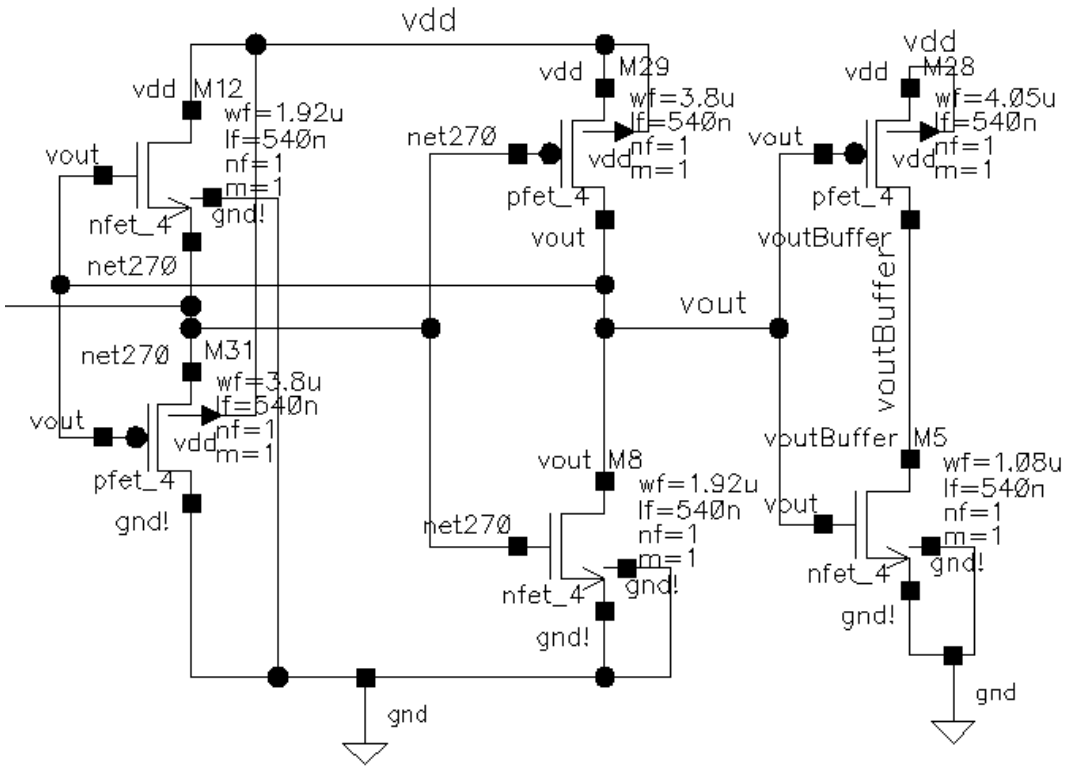

Figure 21 – Träff comparator schematic

Figures 21 and 22 show Träff's comparator. The schematic has been truncated for space considerations, the input on the left is a current (from a pair of current mirrors similar to those in Figure 8) either into or out of the comparator. The gain is approximately an order of magnitude greater at 100MHz. Note that in Figure 20 we see that the buffer provides additional gain. This circuit actually has a slightly reduced gain after the buffer stage, perhaps because the output is not biased near the buffer's switching voltage. Still, a nominal gain of 300k will produce an output swing of 300mV with an input current delta of 1µA. This will easily be driven to full logic levels by buffers.

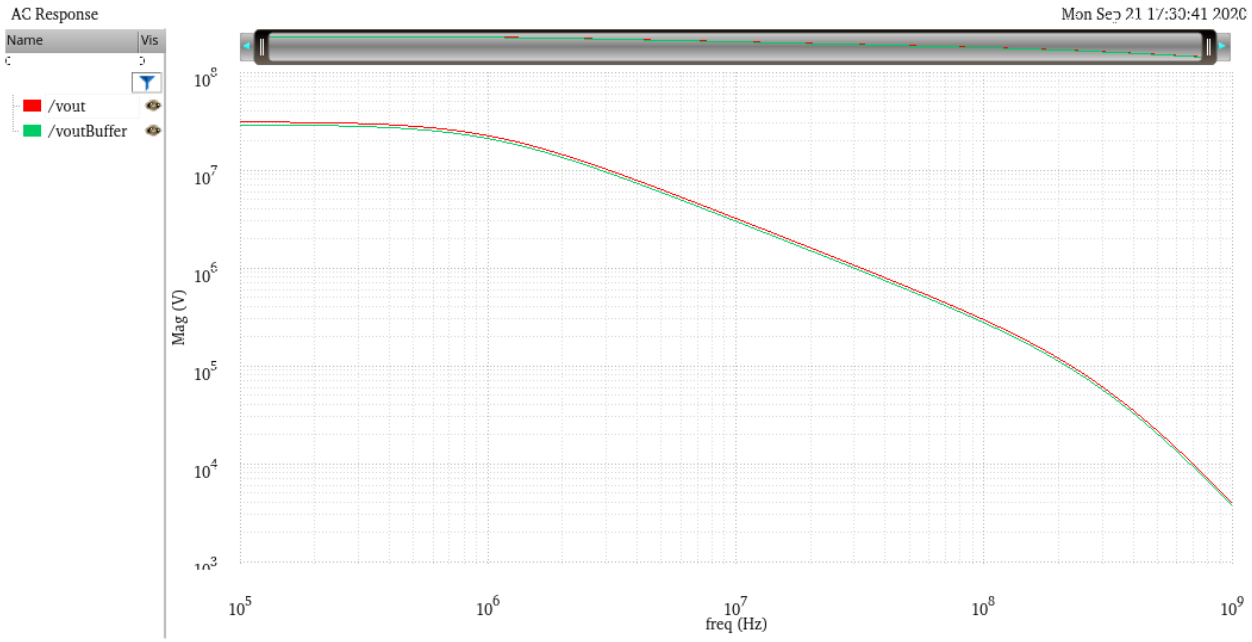

Figure 22 – Träff comparator AC simulation

Finally we can examine our chosen design. As the table below shows we have a massive increase in gain at 100MHz and even at 1GHz this design still has three times the gain of Träff's design. Figure 23 is

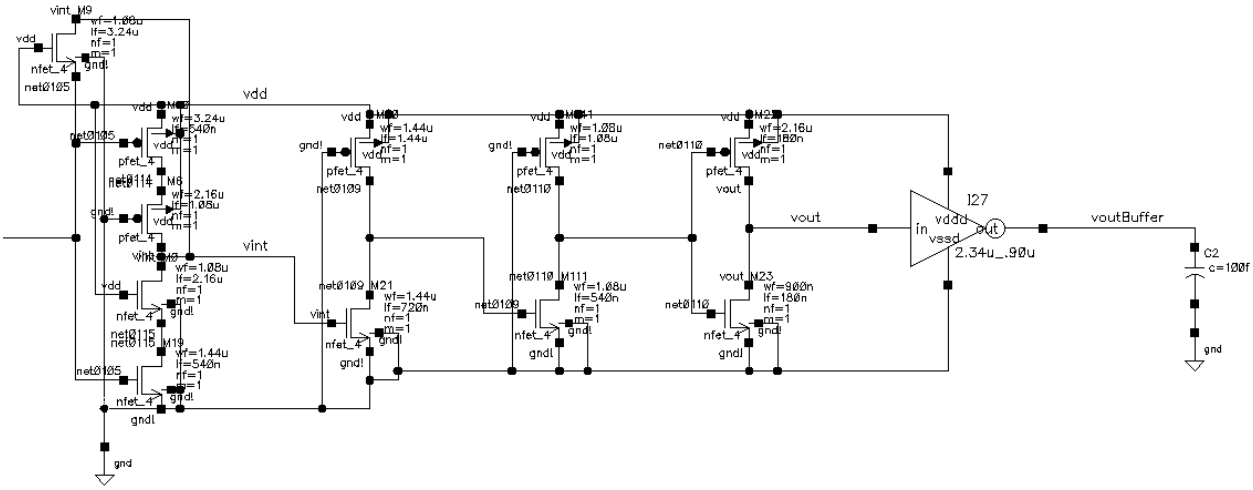

Figure 23 – Schematic of our design

the schematic, excluding the current mirror subtraction (just like Figure 21). Figure 24 shows the frequency response. Note that this is the circuit used in the final design.

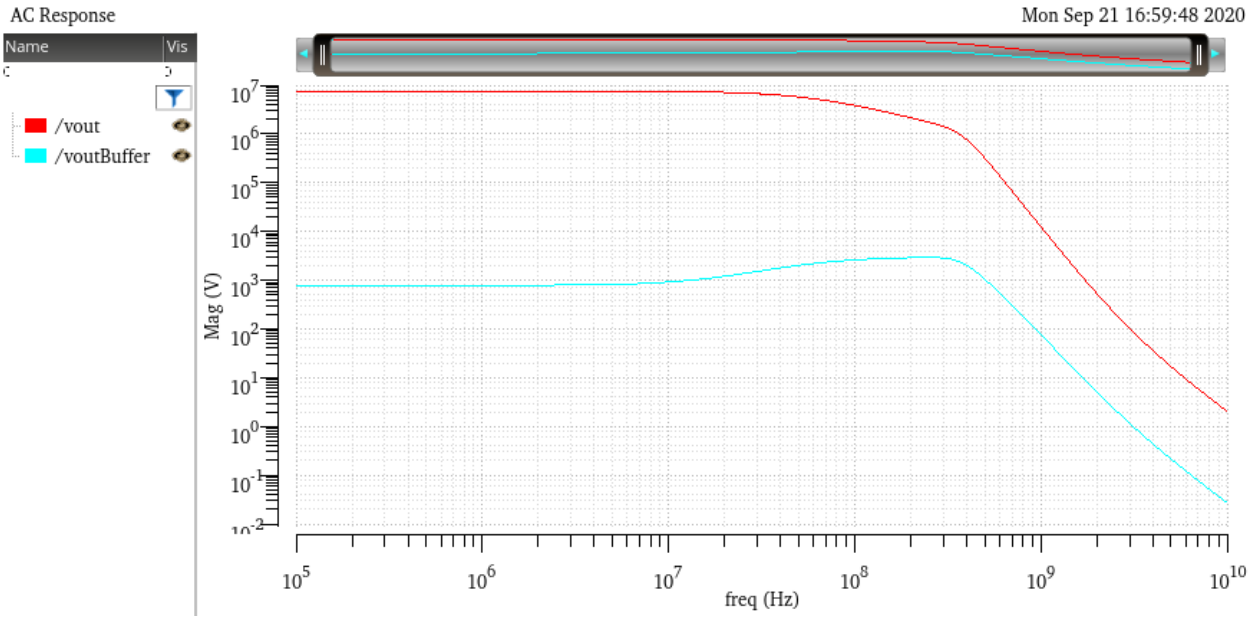

Figure 24 – AC response of our design

To confirm our design choices we present a supplementary simulation in table 5 below. The result of our development is a reduction in the time it takes the comparator to respond to a current pulse for almost all categories. In order to provide a benchmark for measuring speed we use identical inputs with each of the circuits. The rest of the ADC circuitry remains constant while swapping out different comparator blocks. The resulting changes in timing are thus due to the comparator design and as much as possible are not influenced by the rest of the circuitry. The results confirm our design is best. For example, the original design with the standardized input has the first output passing VDD/2 (1.8V/2 = 900mV) at 21.8ns. The final design manages to reduce that time by about 3ns. Similar results are found for the other outputs.

Design	Time to VDD/2 for vout<0>	Time to VDD/2 for vout<1>	Time to VDD/2 for vout<7>	Time to VDD/2 for vout<15>
Current Mirror	21.8ns	27.4ns	51.0ns	N/A
Träff	25.9ns	28.3ns	37.6ns	57.9ns
Chen (2000)	N/A	23.1ns	37.0ns	53.5ns
Final design	18.7ns	21.7ns	34.4ns	54.2ns

Table 5 - Timing results of simulated designs with identical input

7. Comparator current mirroring

Another question is how to supply the two currents to be compared to the comparator. Our reference current is nominally expected to be about 8µA. At full scale the input current would be 63 times larger, or 504µA. If we generated a linearly increasing reference current for each comparator block we would burn an extremely large amount of power (nominally summing from 8µA to 504µA we can calculate power as 1.8V * 16,128µA = 29mW, although in many cases the full current would not be flowing). In addition, the transistor sizes begin to get unwieldy and layout area becomes a concern with a simple mirroring approach.

The solution I decided on was to partition the comparators and scale current to different sections. The first 16 comparators use a full reference current. The next 16 use half the input and reference current. The final 31 comparators are at one-quarter of full current values. The power savings of this approach can be seen with a simple example.

Consider the 20^{th} comparator. It needs to trigger if the input current reaches 20 times the reference current, which we assume to be 8µA. If we mirrored the input current at a 1-to-1 ratio we would need to compare it to a reference current of 160µA. However, if we scale the input down by a factor of 2 we need to trigger when the scaled input current reaches 80µA. We have thus cut power consumption in half for this specific current path.

Variation in the current supplied to the comparator cells due to mismatch in the current mirrors is a major source of non-linearity. One measure that is taken is making the length of the current mirror transistors longer than the minimum 180nm. Most input current mirrors have lengths of 540nm (3x minimum) and the final scaling NMOS mirror in the comparator cell has a length of 720nm (4x minimum). A longer transistor will reduce the effect that variations in length and width have on transistor mismatch.

One idea that was rejected was to use cascode structures to improve the accuracy of the current mirrors. An excellent introduction is found in [6, Chap. 20]. The output resistance of the cascoded current mirror will be greatly increased but unfortunately, as we will soon see, the speed of this topology leaves much to be desired.

A simulation is presented below comparing a standard current mirror with cascode on both NMOS and PMOS and a hybrid design with cascode only on the PMOS. A cascode NMOS for the initial current input requires a much higher input voltage (in this simulation almost 900mV compared to 412mV for the standard and hybrid designs). This is not helpful as a higher voltage here will prevent the APDs from breaking down and recovering properly.

The major problem, as seen in the simulation results in Figure 26, is the additional delay introduced by the cascode structure. The additional parasitic capacitances introduced by the transistors make this

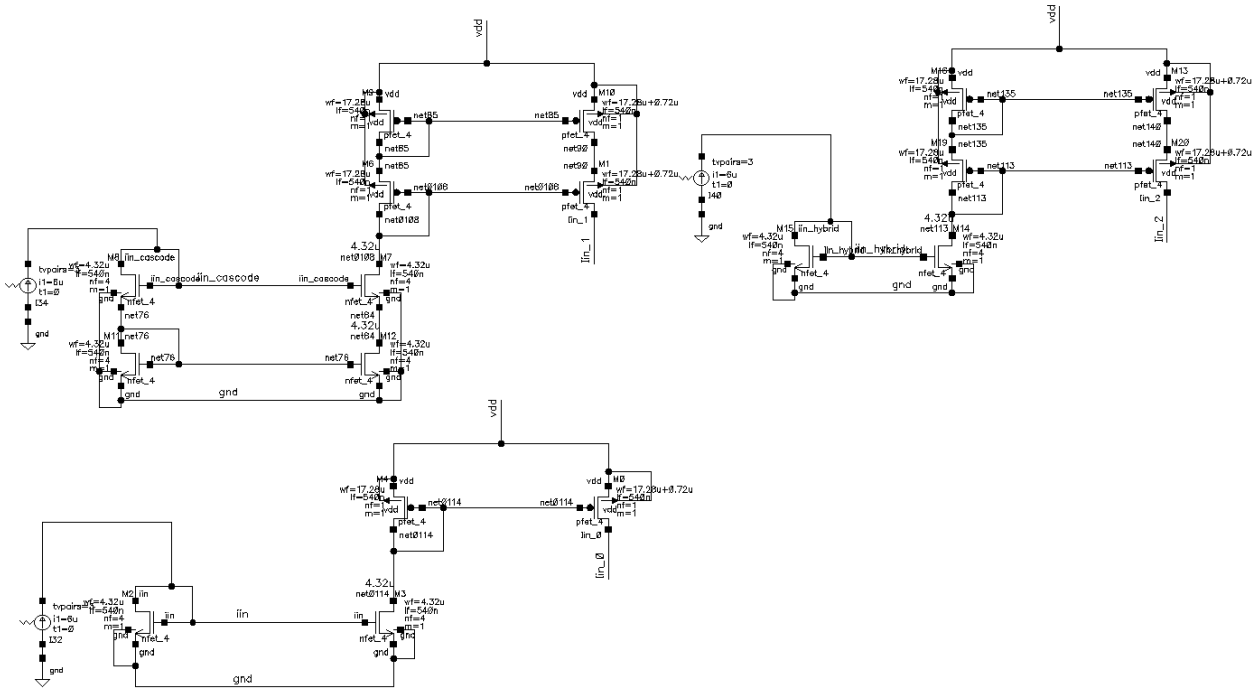

Figure 25 – Cascode test schematic

unsuited for high-speed operation.

In addition, the additional layout area is also significant, almost doubling for even the hybrid circuit (the PMOS mirror makes up the majority of the transistors and area for the mirror, with a dedicated branch for each of the 63 comparators). In this simulation, the standard mirror has an output transition before 10.2ns, while the hybrid and cascode are at 12.8ns and 14.2ns, respectively. This additional 2.5-4ns is a large part of the total response time, given that the current input begins changing at 5ns, and is therefore rejected for this design.

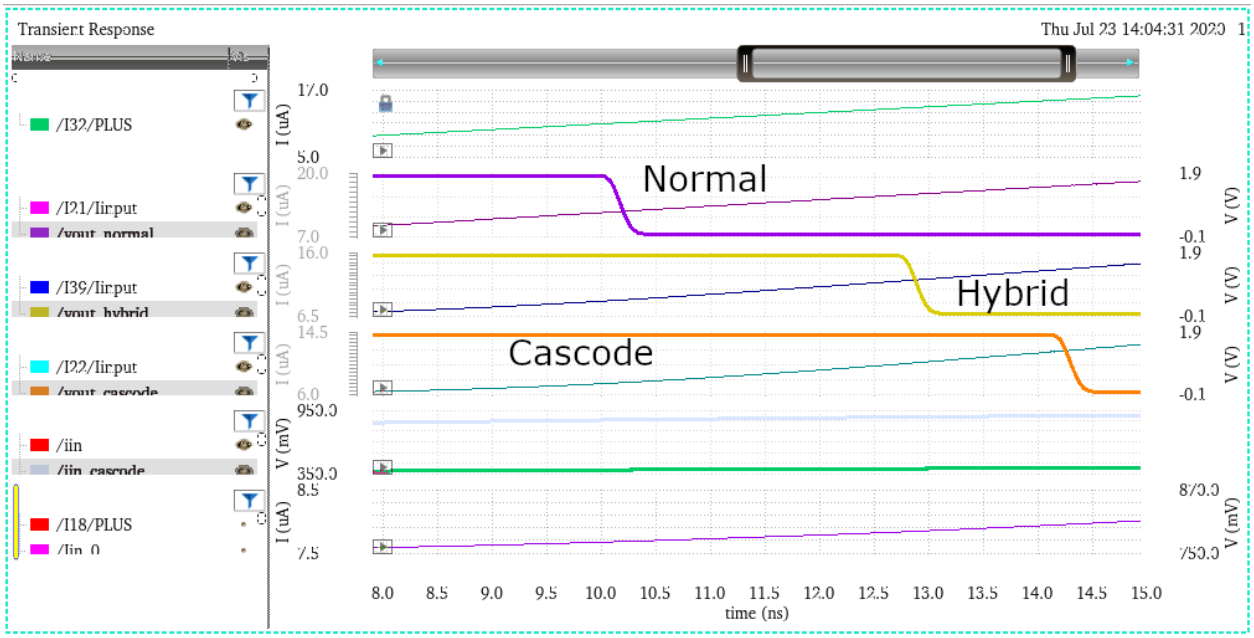

Figure 26 – Cascode testing simulation results

8. Thermometer code

Moving on from the analog input circuitry toward the digital side of the design we now discuss the output of the current comparators, which is a "Thermometer Code." This is a simple "base 1" representation of the detected current level. A simple eponymous example is presented in Figure 27, showing a thermometer with eight output signals. Whenever the temperature rises above a level the output code is '1'. In the example, the output codes for 10 through 40 degrees are '1', matching the level of mercury. In our comparator we apply the same principle to the measurement of the input current level.

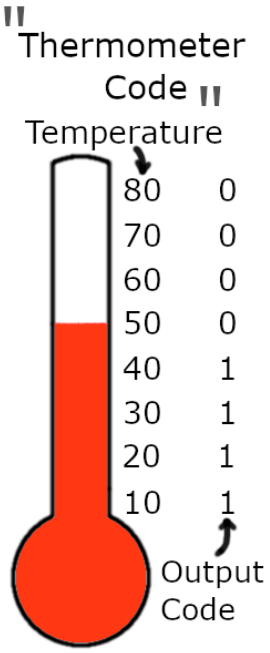

Figure 27 – Thermometer Illustration

Ideally this means that with 63 lines we can detect 63 levels of current plus a "no current" state with all lines low. These 64 states are encoded in the output binary word with 6 bits to uniquely represent $2^6 = 64$ states. Because each comparator is detecting a larger level of current we expect that if output 5 is active then all previous comparators have driven their outputs active (outputs 0-4 would be high in our design).

However, due to manufacturing variation, current mirror mismatch, and other defects, we can have a situation where there are gaps (or "bubbles") in the output code. Before generating a digital output we pass the output from the comparators through a layer of logic designed to remove these irregularities. We generate 63 signals selecting a level only if two consecutive outputs are active. Denoting X(N) as the X*th* selection signal the general logic is:

$$X(N) = V(N) \cdot V(N-1)$$

This ensures the previous comparator also agrees the current has reached its threshold. An isolated active output will not trigger an output. The final design also expects that only one (at most) of the selection lines is active (low) at a time. The active selection line will have (ideally) all ones below it and all zeros above it. Using our previous notation:

$$X(N) = V(N) \cdot V(N-1) \cdot \overline{V(N+1)}$$

This formula is modified for the first signal as there is no previous signal to include in the formula. The last line likewise does not have V(N+1) available. This logic function is implemented with 3-input NAND gates. The inverted V(N) is taken from the inverters that buffer the comparator output. Figure 28 is a representation of how the first three selection signals are generated and includes the inverters that buffer the output of the comparators.

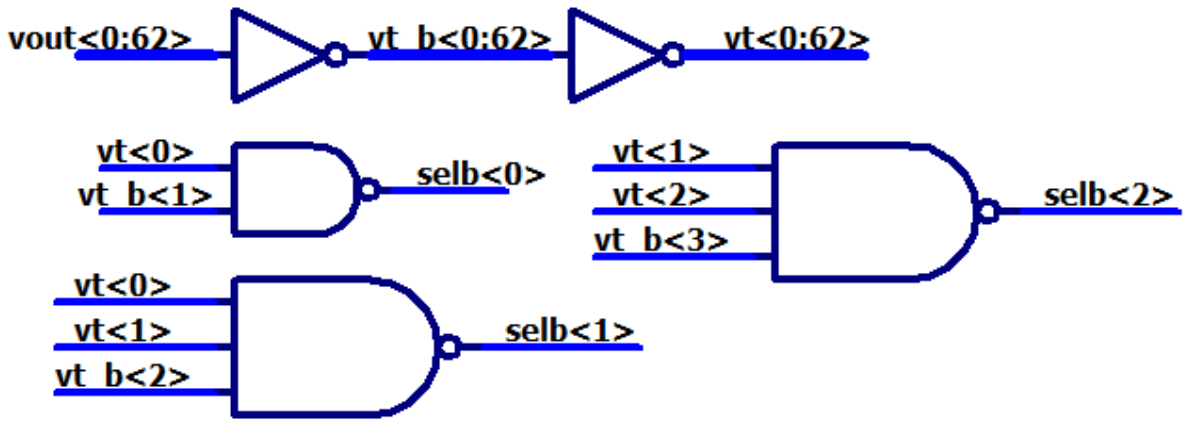

Figure 28 – Schematic of selection logic for first three signals

Note the first line is labeled *selb<0>* (“Selection Bar,” i.e., inverted) and goes low when the buffered comparator output *vt<0>* is high and the next comparator has not triggered so *vt_b<1>* (again, “b” indicates bar, used in Boolean logic to indicate inversion) is high. As noted above the next selection is generated by a three-input NAND gate because there is now a previous input to consider. This continues

until we reach the final gate as noted above. This is again a two-input NAND gate, like the first one, that has *vt<61>* and *vt<62>* as inputs to generate *selb<62>*.

It is important to note that this bubble prevention logic can generally only handle a single bubble. As an example of this limitation, consider if comparators 30 and 31 trigger abnormally due to process variation while the previous highest comparator is comparator 27 (See Figure 29). This leaves a "double bubble" where outputs 28 and 29 are still low leading to a situation with multiple selection lines going high. The MSB, B5, will go high based on its logic while B0 and B1 are still generating an output based on select signals from comparator 27. This can lead to a situation where the output value "jumps" up to three values higher than the "correct" level as B5 (or any of the outputs for the four most significant bits) goes high and B0 and B1 have two selections lines feeding their logic.

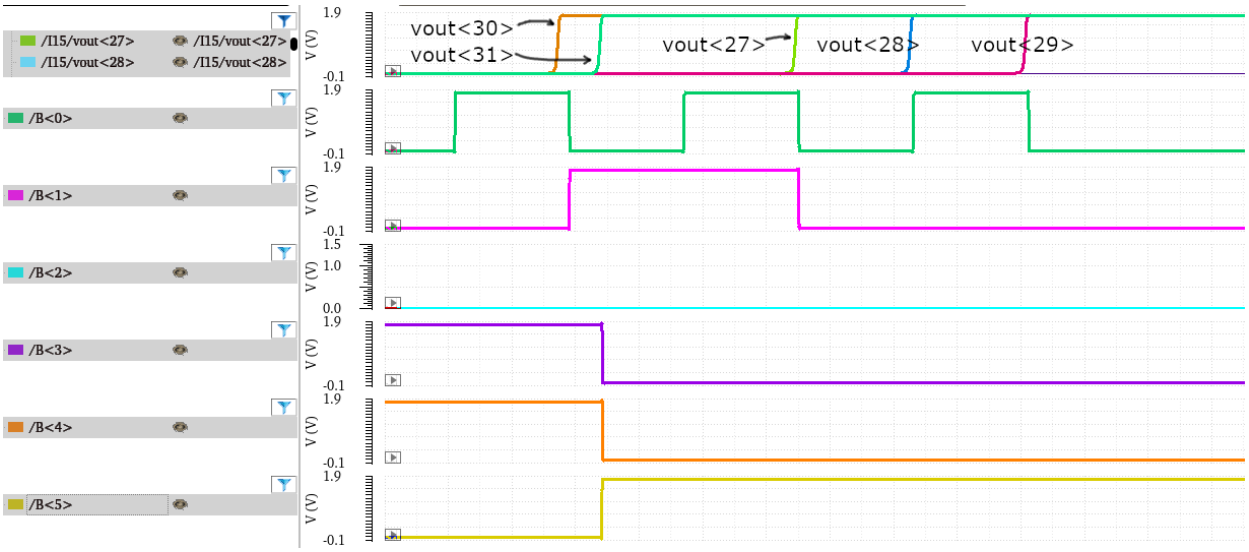

Figure 29 – Dangers of "Double Bubbles"

Fortunately, only the two least significant bits use the selection lines. In Figure 28 we can see that the logic for B3 and B4 deactivates those output bits when vt<30> and vt<31> go high. Note that the correct output in this situation is '32' or '100000' in binary. Because of the double bubble in the outputs of the

comparators the LSBs B0 and B1 continue to operate on erroneous data. At one point they both output a high signal, resulting in the output being ‘35’ or binary ’100011’. As noted, this is three values larger compared to the correct value of ’32.’

Overall we consider this a fairly minor concern as the error is relatively small and requires truly massive process variation unlikely to occur in actual fabrication. Our simulation here shrinks the width of transistors about 33µm in width by over 5µm (over 15% change!) to get the above simulation results. It is not worth dedicating additional hardware to correct this problem.

9. Generating the digital output

The selection lines are used to generate the final 6-bit output word. The original design used a matrix of transistors with M x N total, where M is number of selection lines (not counting complements) and N is number of output bits. For Vikas' design, this was 16 transistors on each of 4 bitlines for 64 transistors total. However, if no selection lines were active no transistors would be "on" and the bitlines connected to the output buffers would essentially be in a high-impedance state.

For a 6-bit design this method would have required 64 transistors on each bitline and the associated

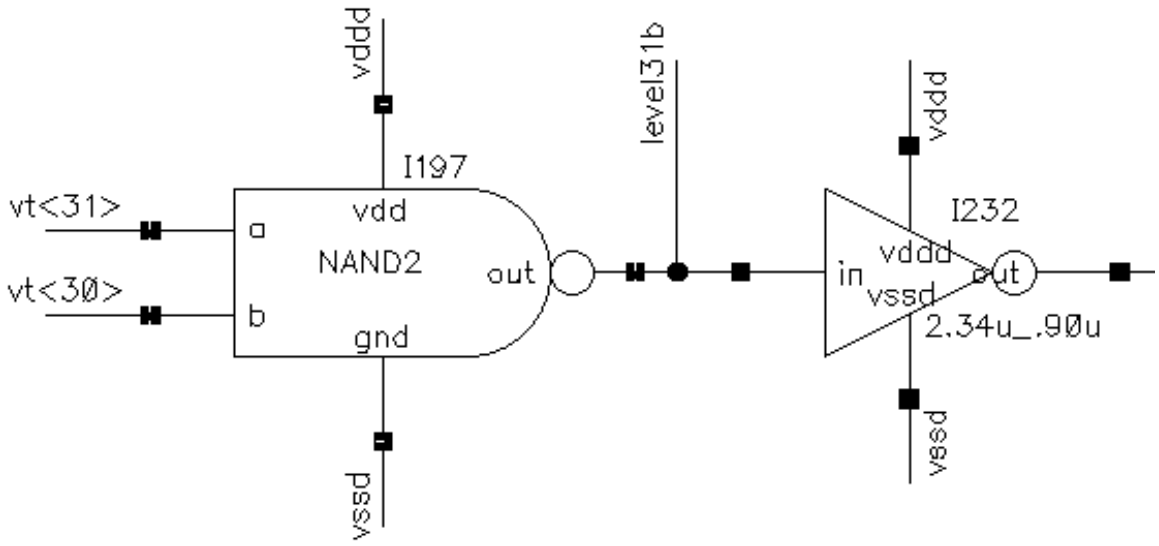

Figure 30 – B5 Generation Logic

parasitic capacitance. Instead a fully digital design is used with logic gates implementing the appropriate Boolean function for each bit. The overall result is very fast as the digital logic has a propagation delay of less than a nanosecond.

Some optimizations are possible for the higher-order bits. For example, if comparators 30 and 31 are active (again, checking two adjacent comparators to make sure we do not trigger on bubbles in the thermometer code) we know the final value will be 31 or higher. We can therefore set bit 5 (B5, the most significant bit) to high without a lot of complicated logic (See figure 30 above). Unfortunately, for the lowest two bits (B0 and B1) we did not find any optimization and a brute force approach is used.

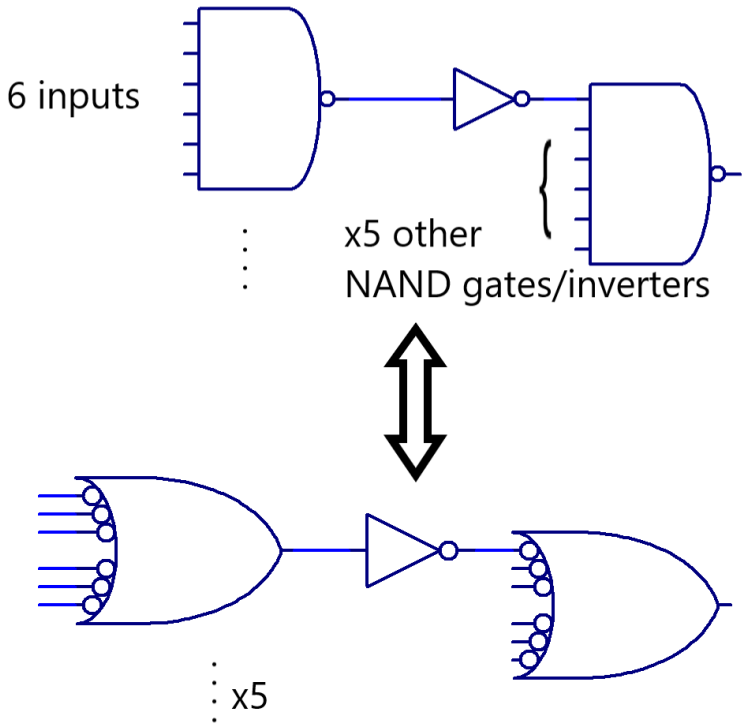

Figure 31 – Equivalent Logic

To understand the following design we first note that a NAND gate can be logically replaced with an OR gate with inverted inputs, as illustrated by Figure 31. For the least significant bit we know the output will be high if any of the even selection lines is high. Recall that when no comparator is detecting current the output of the ADC is all zeros. When the first comparator detects current, selb<0> goes low and we need to activate bit 0 of the output. Note that B0 is high when any of selb<0>, selb<2>, … , selb<62> is low. As seen in the diagram, the active low is inverted by the bubbles on our equivalent OR gate inputs.

We have 32 selection signals we need to OR together. These are routed to four 6-input NAND gates and two 4-input NAND gates (not shown in figure) to reach this total. (4x6 + 2x4 = 32) The output of the first level of gates is inverted and sent to a final 6-input NAND gate.

For the B1 output we use identical logic, only changing which select lines we look at. For the next significant bit we should be high for levels 1, 2, 5, 6, … , 61, 62. We therefore route those connections to our network of gates. Note that every fourth selection code is not needed by the B0 and B1 logic. Selection logic for levels 3, 7, 11, … , 59 are not needed and we thus eliminate these gates from the final design.

For the remaining bits of the output (B2 through B5) we use a different strategy. Examining table 6 below we note that B2 is active for four states from when vt<3> goes high until vt<7> goes high. We still need to check for "bubbles" in the thermometer code because this logic is based directly on the buffered signals from the comparators (*vt<*>* and *vt_b<*>*). The selection signals, with their bubble detection, are only used for the two least significant bits of the output, B0 and B1.

The final logical expression for the first group of four states where B2 is high is:

$$vt<2>\cdot vt<3>\cdot \overline{(vt<6>\cdot vt<7>)}$$

In plain language, this expression is true when at least level 3 has been reached, but level 7 has NOT been reached. Note from Table 6 that when level 7 is reached (vt<6> and vt<7> high) B3 goes high and B2 is back low. there are 8 groups of 'ones' for B2 so similar expressions for these are formed and all 8 signals are logically "OR-ed" together. Figure 32 shows a simplified schematic of this logic.

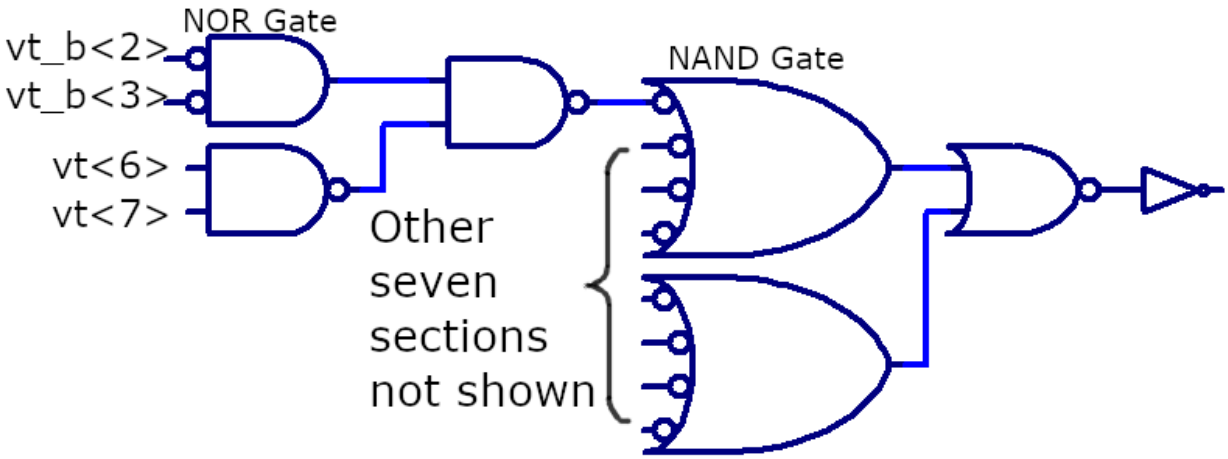

Figure 32 – Partial B2 Logic

Note that the NOR gate in the first level of gates is equivalent to an AND gate with inverted inputs per De Morgan's Theorem in [7]. We simply use the inverted signals from comparator 2 and 3 to make the logic here compact. The NAND gate at this level implements the end of the "block" so that when *vt<7>* is active with no bubble the output of this gate goes low. This deactivates the signal (logic '0') at the output of the second level of gates. There are eight blocks as noted above which must be OR-ed together. We again note the convenient feature that two levels of NAND gates act like a cascade of AND gates followed by OR gates. As shown above the third level NAND gates negate the inversion from the second level and group all the signals into the final NOR gate which has its inversion corrected by an inverter.

A similar process is used for B3, except there are only 4 groups of 8 'ones' so the logic is much simpler. By the time we get to B5 we only have one condition. Any level at or above 31 has the most significant bit set so our logical equation is simply $vt < 30 > \cdot vt < 31 >$. Complete schematics can be found in Appendix A. One method that saves on gates is reusing signals in the logic for multiple output bits. For example, the above equation for B5 is identical to expressions used in equations for B2, B3, and

B4. This is easy to see as those signals go low when level 31 is reached. We can therefore save quite a few gates by reusing this signal.

Highest detector active	B5	B4	B3	B2	B1	B0
none	0	0	0	0	0	0
vt<0>	0	0	0	0	0	1
vt<1>	0	0	0	0	1	0
vt<2>	0	0	0	0	1	1
vt<3>	0	0	0	1	0	0
vt<4>	0	0	0	1	0	1
vt<5>	0	0	0	1	1	0
vt<6>	0	0	0	1	1	1
vt<7>	0	0	1	0	0	0
vt<8>	0	0	1	0	0	1
vt<9>	0	0	1	0	1	0
vt<10>	0	0	1	0	1	1
vt<11>	0	0	1	1	0	0
vt<12>	0	0	1	1	0	1
vt<13>	0	0	1	1	1	0

vt<14>	0	0	1	1	1	1
vt<15>	0	1	0	0	0	0

Table 6 – First 16 output codes

For reference, the complete expressions for bits B2 through B5 are now listed. A special case is the final group from level 59 through 62 for B2. Because there are no further states where the output B2 is low there is no need for the deselect logic (the second part of each expression that "turns off" the output at the end of each group of four). Similar situations are found for the other outputs.

$$B2 = vt<2>\cdot vt<3>\cdot\overline{(vt<6>\cdot vt<7>)} + vt<10>\cdot vt<11>\cdot\overline{(vt<14>\cdot vt<15>)}$$

$$+vt<18>\cdot vt<19>\cdot\overline{(vt<22>\cdot vt<23>)} + vt<26>\cdot vt<27>\cdot\overline{(vt<30>\cdot vt<31>)}$$

$$+vt<34>\cdot vt<35>\cdot\overline{(vt<38>\cdot vt<39>)} + vt<42>\cdot vt<43>\cdot\overline{(vt<46>\cdot vt<47>)}$$

$$+vt<50>\cdot vt<51>\cdot\overline{(vt<54>\cdot vt<55>)} + vt<58>\cdot vt<59>$$

$$B3 = vt<6>\cdot vt<7>\cdot\overline{(vt<14>\cdot vt<15>)} + vt<22>\cdot vt<23>\cdot\overline{(vt<30>\cdot vt<31>)}$$

$$+vt<38>\cdot vt<39>\cdot\overline{(vt<46>\cdot vt<47>)} + vt<54>\cdot vt<55>$$

$$B4 = vt<14>\cdot vt<15>\cdot\overline{(vt<30>\cdot vt<31>)} + vt<46>\cdot vt<47>$$

$$B5 = vt<30>\cdot vt<31>$$

10. Practical concerns with the digital logic

Simulations indicate that the propagation delay through the digital logic is very small. In the simulation in Figure 33 (below) we can see that vout<31> goes high at about 25.1ns. There is a bit of skew as the six output bits transition but the last one (B0) goes low at 25.7ns, resulting in a delay for this code of about 600ps. Other output codes have similar delays as the transitioning logic levels propagate through the same number of gates.

One observation is that the most significant bit reacts first. We expect this because (as noted previously) B5 is set due to a simple AND of two signals, vt<30> and vt<31>, which demonstrate the second half of the thermometer code has been reached. It is undesirable that the different bits of the output word transition at different times as this leads to short periods of time where the output is wildly incorrect. If the output happens to be sampled during these transition periods we would have inaccuracy. To alleviate this factor additional inverters are added to introduce additional delay to the fast paths such as the B5 logic.

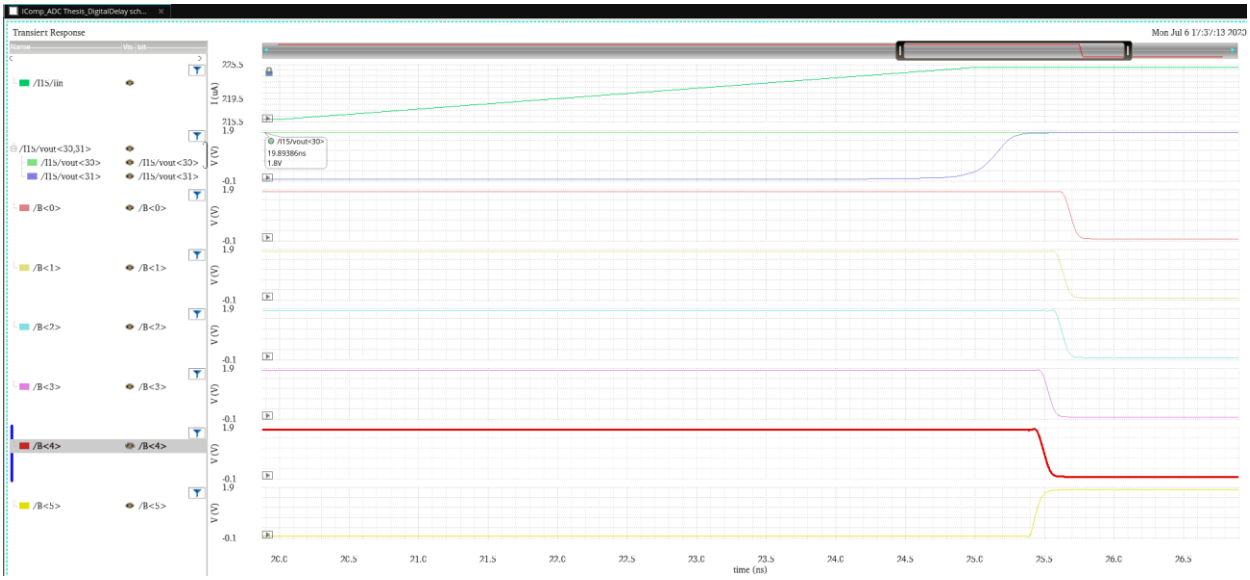

Figure 33 – Digital Delay

Another concern is determining how large to make the output buffers. The six digital bits need to be sent off-chip and require large buffers to drive this external load capacitance. If we require an output frequency up to 100MHz then our RC time constant should be on the order of 1ns or better. We design for a load capacitance up to 10pF:

$$\tau = RC$$

$$R \leq \frac{\tau}{C} = \frac{1ns}{10pF} = 100\ Ohms$$

Simulations show that for 180nm length transistors with Vds and Vsd of VDD/2 (900mV) the on resistance is 1.6k/L for NMOS, 4.1k/L for PMOS, where L is in µm. Note from the graph in Figure 34 that on resistance is a function of the voltage applied across the transistor so these numbers are only approximations. Our output buffer final stage has an effective width of 38.88µm for a resistance of about 1.6k/38.88 = 41 Ohms. The PMOS with a width of 97.2µm has a calculated on resistance of 4.1k/97.2 = 42 Ohms. These values are well below 100 Ohms to have some margin and ensure we meet design specifications.

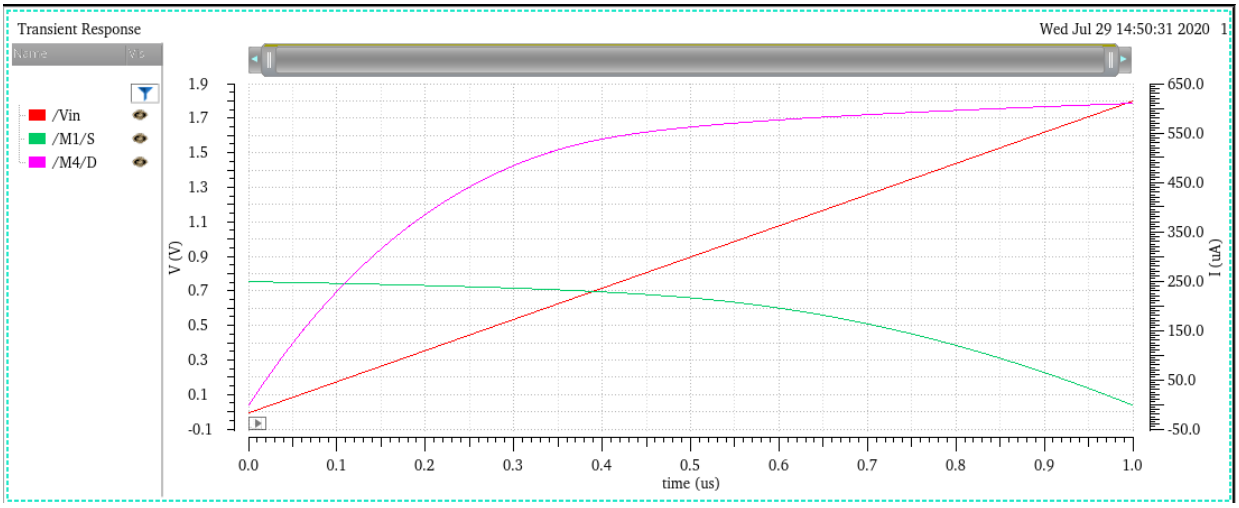

Figure 34 – On Resistance Simulation

One final issue to consider is the possibility of power supply noise. The large buffers driving the output pads will draw large amounts of current, especially when multiple output bits are changing state simultaneously (most transitions will have this result). Figure 35 shows a simulation with estimates for bond wire and internal parasitics included. Current consumed by the buffers can exceed 50mA for short durations. If this current (combined with current drawn by the digital and analog circuitry) was sourced from a single pad it is likely that the resistance and other parasitics would lead to an unstable VDD and VSS power supply. In the simulation the voltage supplied to the buffers dips as low as 1.25V, which is significantly below the nominal VDD of 1.8V.

The simulated bond wire has a 2nH inductance and a 300 milliohm. These estimates were based off typical values for bonding wire. Additional resistance is calculated from the process PDK and the actual typical metal paths in the chip layout. Actual results will hopefully be better than the simulation as conservative values were chosen. However, we can see that voltage fluctuations are very likely to add noise and inconsistency to the results of the analog comparators. It was therefore decided to provide a separate power supply for the analog portion of the circuit.

Although the digital logic is not as sensitive to power supply variations there was room to provide an additional power supply for that part of the circuit as well. The final design is six pads (three supplies with VDD and VSS) that should provide a stable power supply and improve the reliability and performance of the circuit.

After laying out the design additional space was available that was utilized to implement decoupling capacitors. Although this only adds a few pF of capacitance it is “on chip” and can supply current transients without worrying about bond wire parasitics.

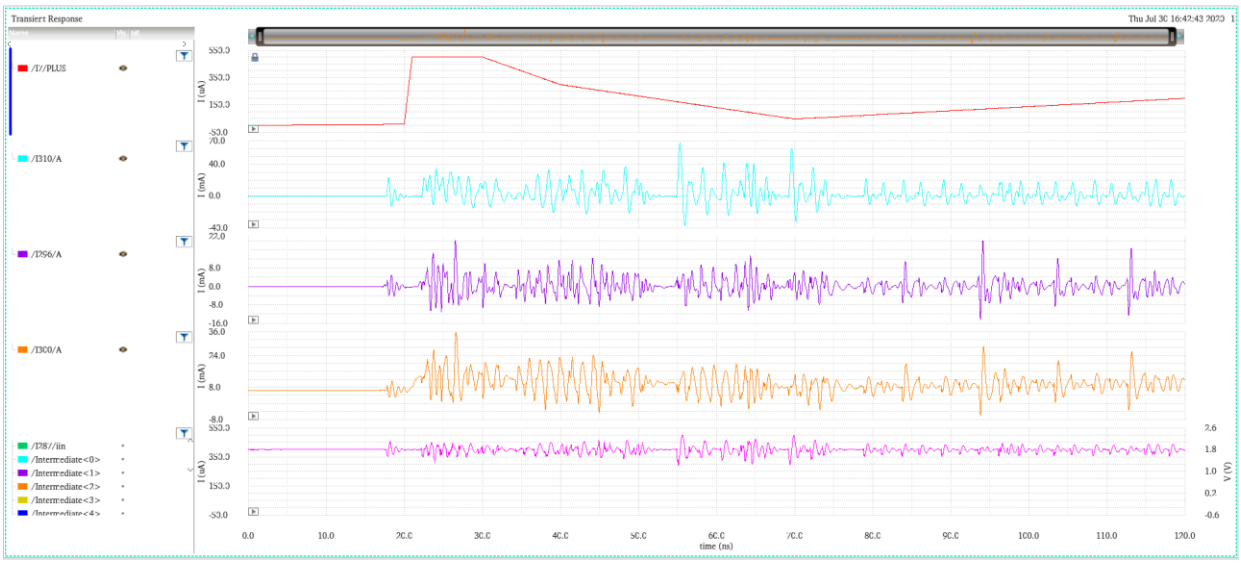

Figure 35 – VDD Droop/Noise Simulation

Preliminary test results from the chip indicate the presence of fluctuations on the power supply rails under certain conditions. These oscillations occur at frequencies up to approximately 100MHz. An amplitude of 100mV was observed on the oscilloscope. It is probable this is caused by the output buffers drawing large amounts of power as the output bits change state. These oscillations can also feed into the analog circuitry through the power connections on the circuit board.

To determine how likely induced instability is through this mechanism we run a simulation with a 100mV amplitude, 100MHz sine wave superimposed on VDD. A relatively slow input signal ramps from 6µA to 35µA. As Figure 36 indicates, there are multiple glitches as the lower three bits change state. The variations in supply voltage affect the comparators' switching points and when current is slowly ramping there is a significant time where oscillations in the comparator's output may occur.

Mitigation strategies could include larger decoupling capacitors, special logic to prevent the output codes from changing wildly, or even digital processing of the binary output data to remove oscillations and clean up the waveforms. For example, as soon as B0 goes high we could note that in the context of

an increasing trend it is unlikely that the input signal had rapid decreases resulting in the observed oscillation. Software processing could therefore remove those glitches (seen around 100-150ns in the simulation). Hardware modifications would essentially do something similar in digital logic and may be worth thinking about in some applications.

It should be noted that parasitic inductance and resistance in the bond leads and board traces play an important part in these power supply fluctuations. Consider how much charge must be supplied to decoupling capacitors to supply the buffers. If 50mA is drawn for 20ns, total charge is:

$$Q = I * t = 50mA * 20ns = 1000pC = 1n\ Coulomb$$

If we want to keep the power supply voltage drop withing 1mV of nominal we see that the decoupling capacitor must be sized according to the capacitor charge formula, Q = C*ΔV:

$$C = \frac{Q}{\Delta V} = \frac{1nC}{1mV} = 1\mu F$$

Given that the test circuit board has capacitance equivalent to this we can conclude that parasitic effects or power supply noise are more likely causes of this ripple and increasing the decoupling capacitor size will have little effect.

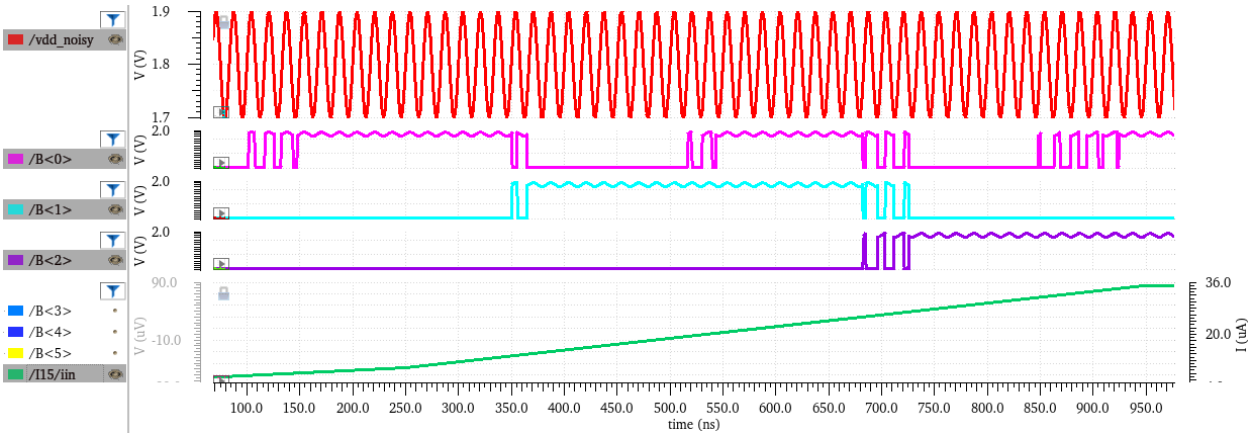

Figure 36 – Power Supply Noise Effect on Comparator

11. Layout of the chip

Layout of the circuitry was performed using the Cadence Virtuoso Layout Suite. The design was fabricated using a TowerJazz 180nm SiGe BiCMOS process. The complete design is about 830µm x 395µm, including the pads, for a total area of 0.328mm^2. The core circuitry (excluding pads and buffers) is about 300µm x 125µm (0.0375mm^2). The process used has six layers of metal. Metal1 through Metal4 are relatively thin and used for most of the signal routing. Metal5 and Metal6 are proportionally thicker and can handle the large currents in the power and ground connections. Although the process used offers a 3.3V transistor we only used the standard 1.8V MOSFETs and did not investigate if the 3.3V transistors would provide robustness or matching advantages.

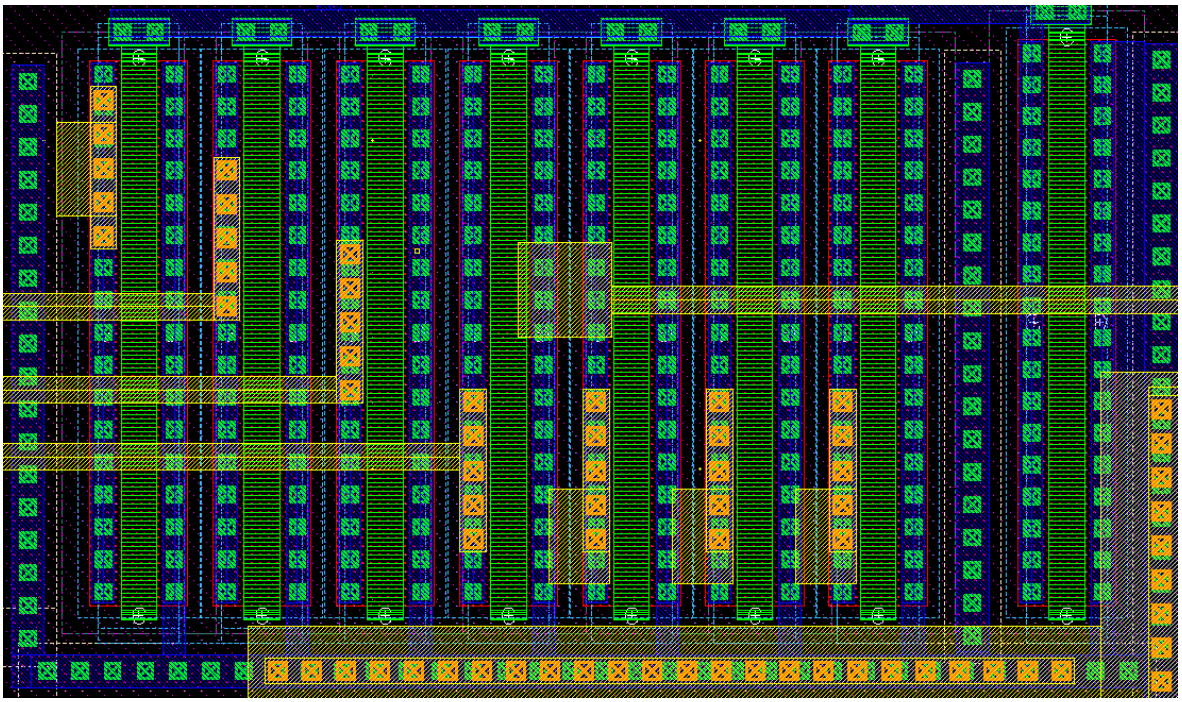

Figure 37 – Layout view of current mirrors

Several fundamental principles were applied to increase the reliability of the fabricated circuit and improve its performance. One of the most basic failures is a connection failure caused by a fabrication defect or mask misalignment. If a gate were connected by only a single contact to Metal1 then a defect

there could break the connection and potentially render the entire chip inoperable. For this reason most of the Poly-Metal1 connections have at least two contacts. Similarly, most connections between metal layers have multiple vias. This strategy also improves reliability when masks are not properly aligned. If one via misses a connection to a metal layer then it is probable another via from the same connection will overlap both metal layers.

Noise reduction is also important, especially for the analog comparators. Digital logic has significant noise margin, which partly explains its speed. However, the fast edge transitions in digital designs (where a conductor has a rapid change in voltage from logic '0' to '1' or vice versa) can introduce transients to slower analog circuitry if they are capacitively coupled. This source of noise could change the output state of a comparator close to its switching threshold. One method used to mitigate this noise is routing the digital output lines as far away from the substrate and analog circuitry as possible, using Metal3 or higher layers in our design.

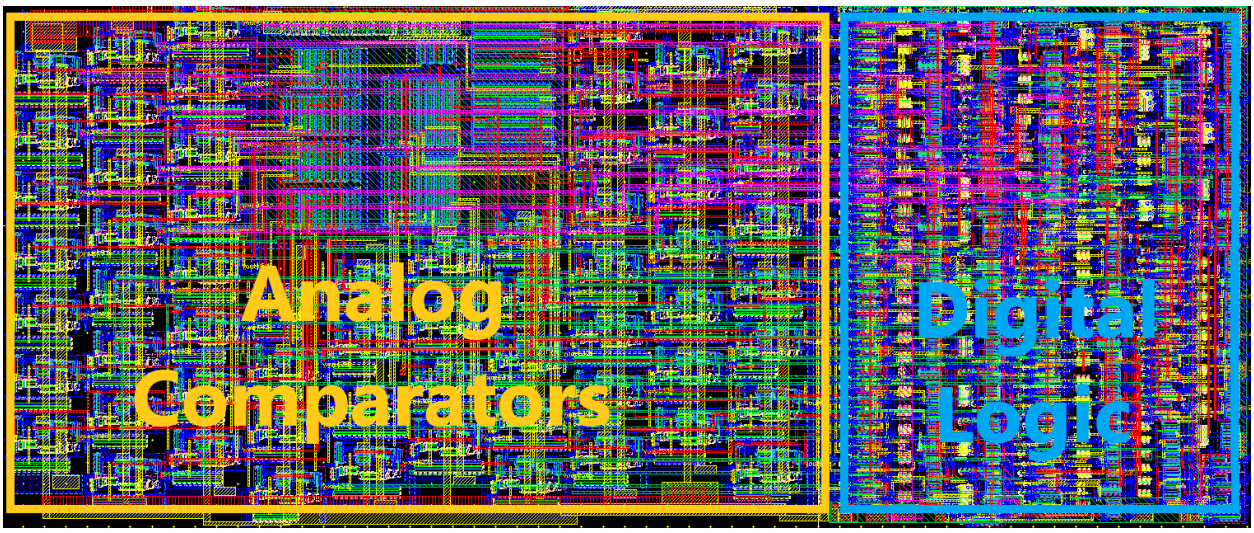

Figure 38 – Overview of the chip excluding pads and buffers

Another more complex method is to place a metal shield (usually grounded) between the digital wires and the analog circuitry. In the central part of the analog section there are current mirrors that

supply individual comparators with individual sources of the input and reference currents. Comparator blocks to the left of this section need to run high-speed digitals lines to the digital logic at the right side of the chip (See Figure 38). In this region a plane of Metal4 is connected to ground and digital lines are run above this shield on Metal5. Any voltage changes on the digital lines will charge the capacitance from the line to the metal plane but will have almost no effect on the other side as the potential of the grounded plane This should provide a significant level of protection to the current mirrors below.

Another way that the digital logic can feed noise to the analog circuitry is through the substrate. In the area between the analog and digital circuitry, although not noted in Figure 38 above, we have placed an isolation guard region. As shown in Figure 39 it consists of two guard regions with grounded P-taps, each 1µm wide. Sandwiched in-between is a N-Well with a 2µm wide active N-tap region connected to VDD. The purpose of this is to collect any minority carriers (electrons) diffusing through the substrate. When these electrons reach the depletion region they are swept out and collected by the connection to VDD. The electric field in the depletion region goes from positively charged atoms in the N-Well to negatively charged atoms in the P-Well substrate. Any electrons that drift into this “space charge” region are pulled to the N-Well side where they are eventually removed at the VDD connection.

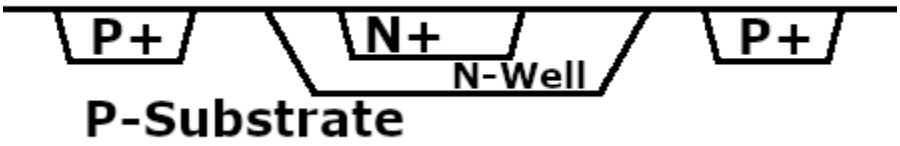

Figure 39 – Cross-section view of guard region

Efforts were made to provide a stable power supply to the analog circuitry to avoid inaccuracy caused by noise on VDD or VSS. Three separate VDD/VSS pairs are provided. The output buffers are so large they have their own power supply to avoid causing severe fluctuation in the supply voltage for the

other circuits on the chip. The digital circuitry also has its own supply as digital logic draws current in large bursts and the resulting VDD droop or ground bounce might cause comparators to detect incorrect current levels. This was already mentioned previously.

Other layout techniques are used to promote good matching between transistors in current mirrors. In figure 40 we can see one of the 1-to-8 mirrors for the input current. All transistors are drawn in the same direction (parallel to each other) with similar surrounding structures. To be even more precise interdigitation can be used to counteract process gradients, but this can use a large amount of layout area.

In modern sub-micron processes the structures surrounding a transistor can have a significant effect on its characteristics. A transistor and those it is matched with should have (as much as practicable) identical or similar patterns of active, poly, and other layers surrounding it. In our design one transistor in the current comparator block has a variable width that results in the differentiated switching points. In order to keep the matching and current ratios through these transistors as consistent and linear as

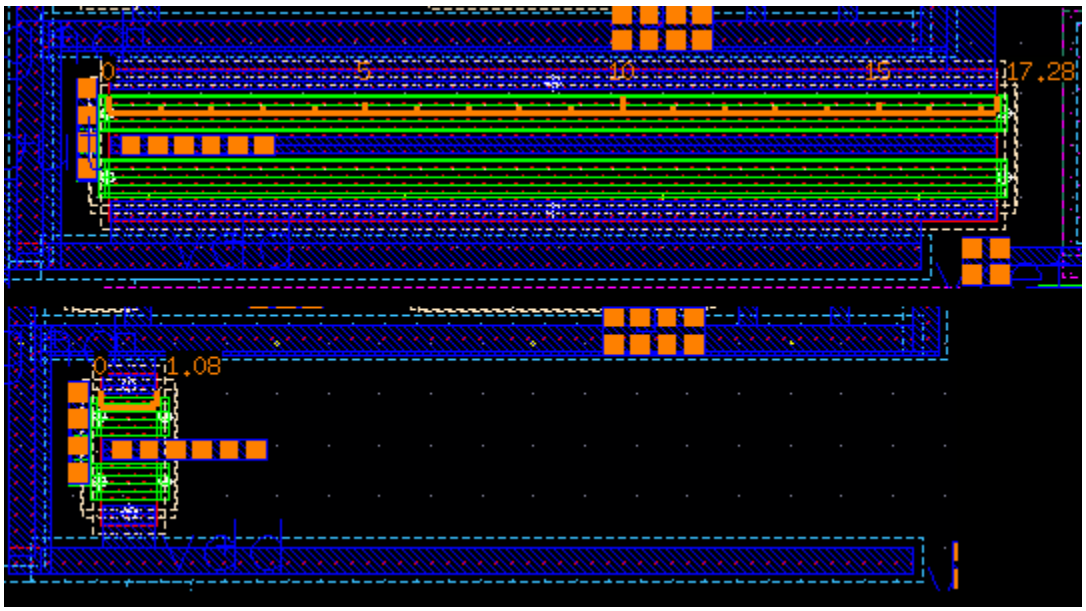

Figure 40 – Current mirror transistor size variation

possible each transistor has identical rows of p-well contacts above and below. As seen in Figure 40 the width of this transistor can vary from 1.08µm to 17.28µm, with a significant void for the smaller transistor. This actually doesn't waste too much layout area as it is difficult to fit anything into irregular spaces like this. Also notice that two fingers are used for this transistor to produce a more compact layout.

The compact layout of a complete comparator is shown in Figure 41. The VDD and VSS nets are outlines and the core transistors (See Figure 11) are noted. It is important to have many taps into both the N- and P-wells to avoid latch-up, a potentially damaging condition exacerbated by long distances

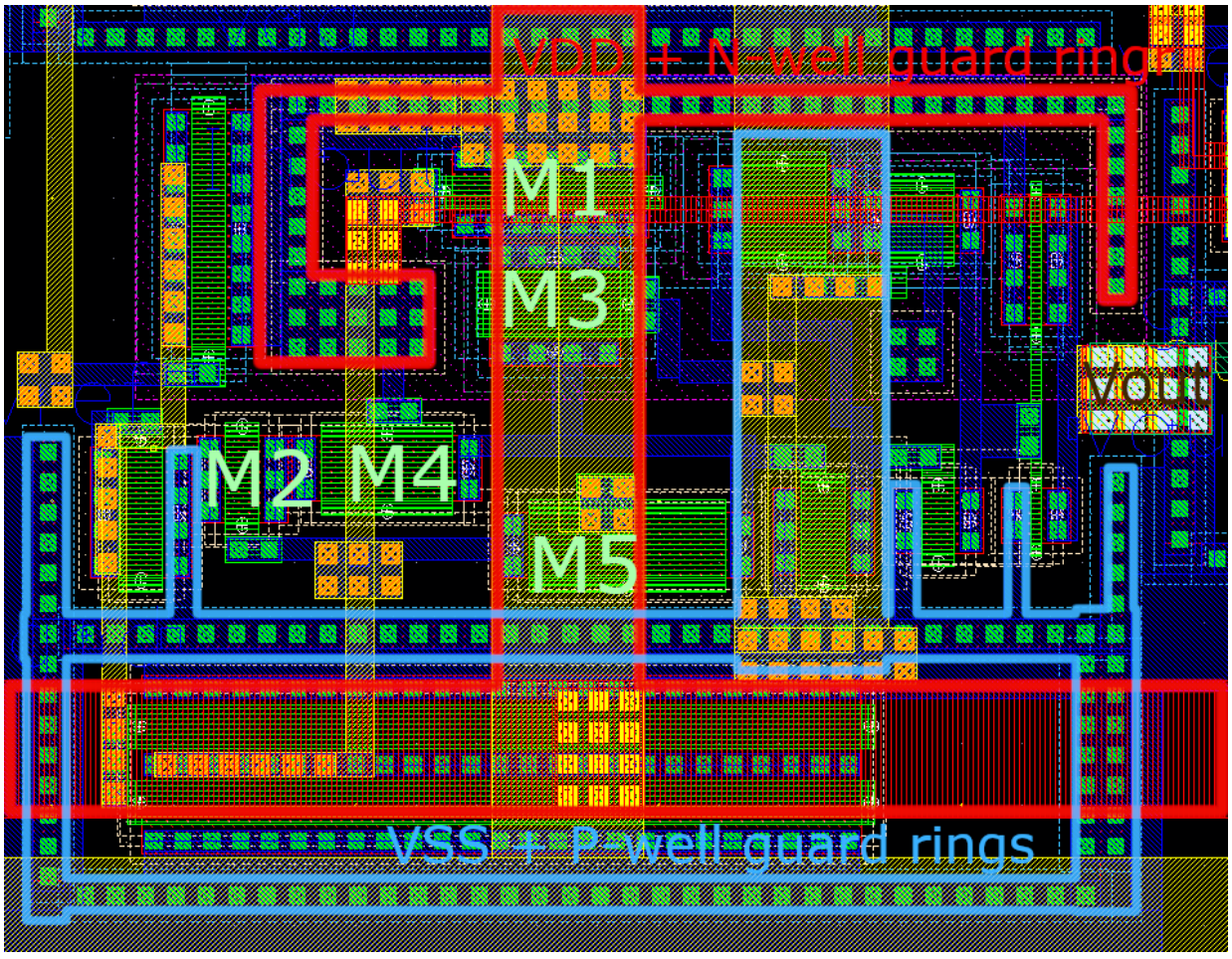

Figure 41 – Annotated layout of single comparator

from transistors to well ties. Another reason for these “guard rings” is the reduction in noise coming from other parts of the chip.

Another notable feature of the above layout is the power delivery via the vertical runs of Metal 2 (yellow). It is important to have wide and preferably redundant runs of metal to deliver power with minimal resistance.

The digital logic is based on a small number of standard logic elements. NAND gates with up to six inputs as well as inverters and NOR gates are used to implement a logic function to convert thermometer code to the six-bit binary output. We created a layout for each gate in a “standard cell” layout. This means a row of N-well contacts connected to Metal1 provide a VDD rail and a complementary arrangement on the other side with P-well and VSS.

The spacing between these active contact areas is 10.52µm. Every circuit and digital block used must fit in this space. Extra Metal1 is used past these rows of contacts to provide a lower resistance power supply. Width of these supply rails is 1.4µm. With all this extra distance and a gap of 0.6µm between adjacent power supply metal the total pitch of this standard cell design is 13.92µm. There are seven columns of these standard cells and the total size of the digital logic is 96.84µm wide and approximately 121.2µm high.

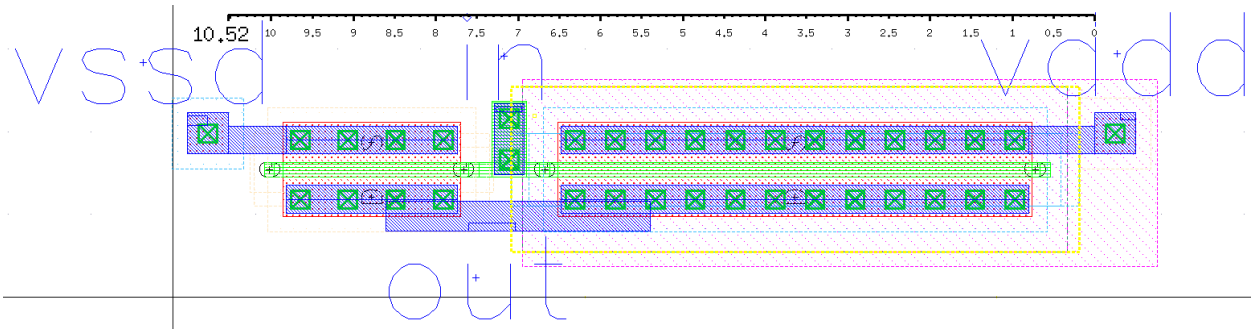

Figure 42 – Standard Cell Inverter

Figure 42 shows the layout of a common inverter used extensively in the design as an example of a standard cell design. Both NMOS and PMOS transistors have a minimum length of 180nm. The PMOS has a width of 5.76µm and the NMOS has a width of 2.16µm for a ratio of 2.67.

We now conduct simulations to validate our designs and provide helpful design parameters for both inverters and gates. A DC sweep (Figure 43) indicates a switching point of about 895mV, which is extremely close to the ideal of VDD/2 = 900mV. This indicates that our PMOS/NMOS ratio is reasonable.

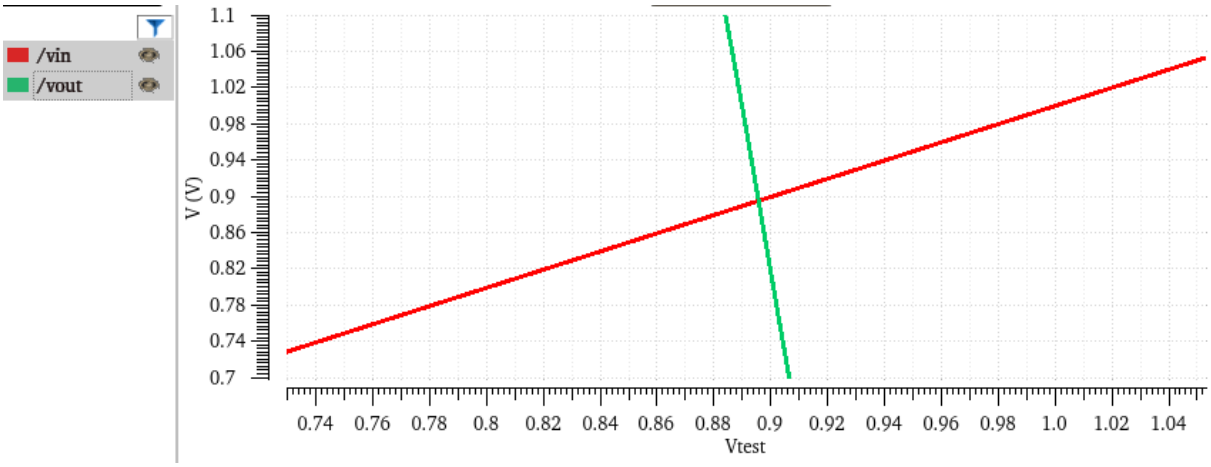

Figure 43 – Inverter Switching Point Simulation

Another useful simulation is to determine the drive strength of the NMOS and PMOS transistors. We tie the output of the inverter to a voltage source of VDD/2 (900mV). As the input voltage sweeps from ground to VDD we examine the current flowing into the voltage source at the output. Figure 44 shows that when the input is at 0V (PMOS is fully on) the voltage source is sinking 1.26mA of current. Dividing 1260µA/5.76µm = 219µA/µm of gate length. When the input voltage is 1.8V the voltage source supplies

current, leading to a calculation of 1,214µA/2.16µm = 562µA/µm. We can use these current ratings to estimate how quickly gate capacitances are changed when sizing our inverters and gates.

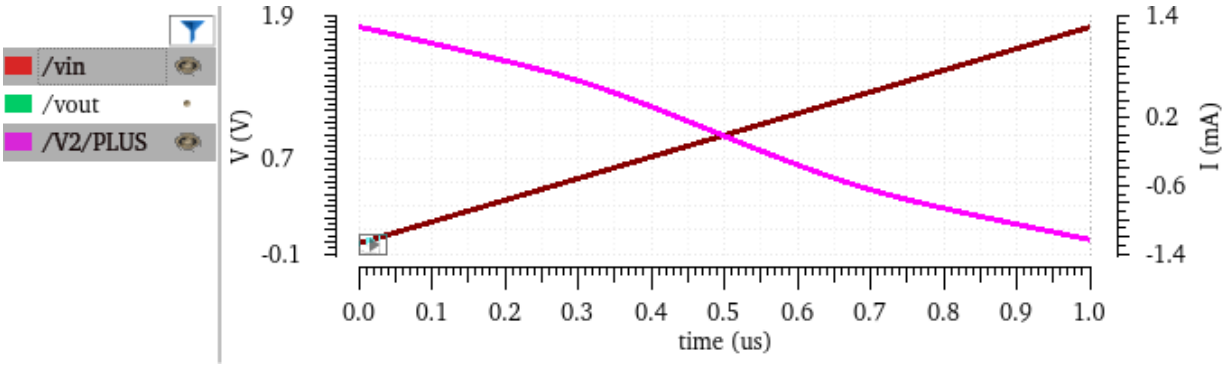

Figure 44 – Transistor Drive Strength

The overall layout routes power vertically from the perspective of the design editor. This allows signals to come in horizontally from the analog comparators on the left and immediately connect to inverters to buffer the signal and generate complementary logic levels if needed. All 63 input lines cannot connect in the first column; these additional signals are routed horizontally on metal layers four and five to inverters located in a variety of locations.

As noted above, Metal1 is used for power routing in the vertical standard cell rails as well as internal logic block connections. Metal2 is generally run horizontally with Metal3 handling vertical connections. This strategy of alternating horizontal and vertical wires on alternate metal layers helps avoid congestion and keep routing organized.

The basic layout of the design is 14 pads surrounding the core circuitry and output buffers. In Figure 45 the core area is highlighted with a dotted yellow line. The small groups of devices near the pads are ESD (electrostatic discharge) protection. These are diodes that will start conducting if the potential at the pad goes much above VDD or below ground. They are designed to prevent damage to the physical circuit if a static discharge occurs on one of the device's pins.

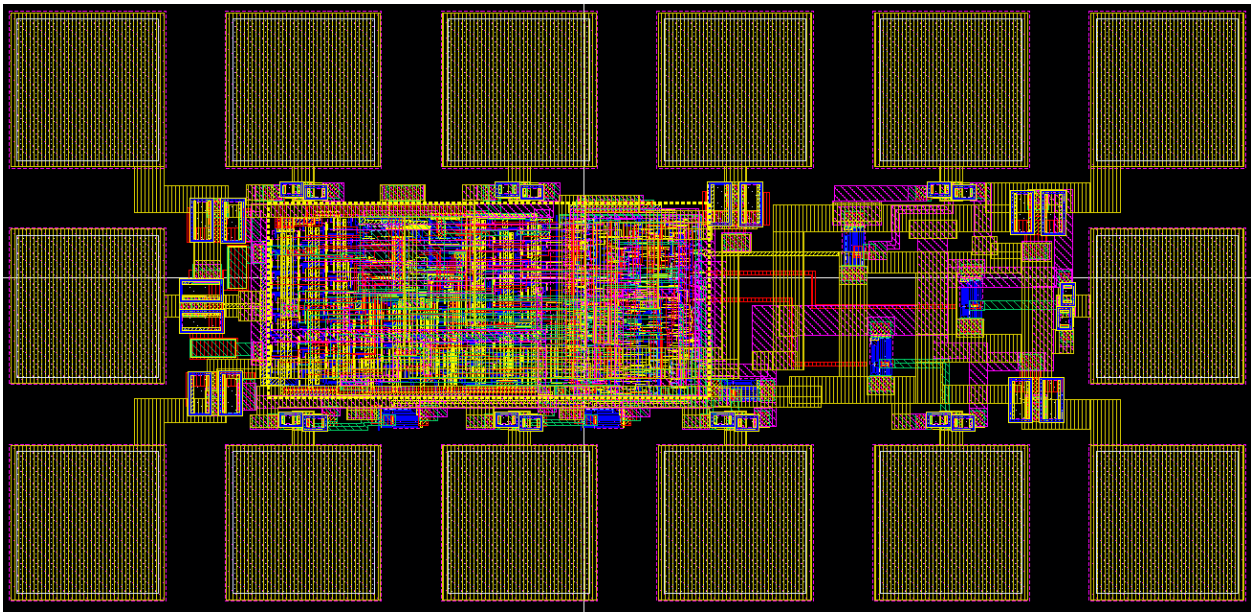

Figure 45 – Complete Layout

12. Simulation of differential and integral nonlinearity

One metric is often used to determine the quality of an Analog-to-Digital converter (ADC) is Differential Nonlinearity, or DNL. An ideal ADC will have a constant change in input signal (current for our circuit, but voltage in other cases) between output code transitions. For our example this "code width" is ideally 8µA but will have to be determined from simulation or measurement results. Each digital output code change indicates that the input current has increased by 8µA. The difference between the actual change in input current and the ideal value is the DNL for that point.

Because our reference current is 8µA we would expect the output code to change when the input changes by that amount. Due to the layers of current mirrors and the non-linearity of MOSFETs we will need to determine the actual linear code width for our simulation and testing results. The most common method is to take the first and last points were the ADC produces an output and calculate a linear gradient between these two points.

For the ideal simulation (not Monte Carlo) the first transition from code 0 to 1 is at 7.361µA. Note that code zero indicates no current and also does not have a DNL associated because that is our default output. The final transition is at 462.123µA. Therefore, 62 output transitions in 454.762µA results in a code width of:

$$\frac{462.123 - 7.361}{63 - 1} = \frac{454.762}{62} = 7.335\mu A$$

The expected transition for output 2 would be at 7.361µA + 7.335µA (code width) = 14.696µA. Results are presented below in Table 7.

Output Code	Transition Current (µA)	DNL (µA)	INL (µA)	Output Code	Transition Current (µA)	DNL (µA)	INL
0	N/A	N/A	N/A	32	217.960	-0.249	-16.78
1	7.361	N/A	0.000	33	237.440	12.145	-4.637
2	13.015	-1.681	-1.681	34	244.762	-0.013	-4.650
3	18.849	-1.501	-3.182	35	252.097	0.000	-4.650
4	24.827	-1.357	-4.539	36	259.446	0.014	-4.635
5	30.918	-1.244	-5.782	37	266.807	0.026	-4.609
6	37.101	-1.152	-6.934	38	274.181	0.039	-4.570
7	43.362	-1.074	-8.008	39	271.568	0.052	-4.518
8	49.690	-1.007	-9.015	40	288.967	0.064	-4.454
9	56.076	-0.949	-9.964	41	296.378	0.076	-4.378
10	62.513	-0.898	-10.86	42	303.801	0.088	-4.290
11	68.996	-0.852	-11.71	43	311.236	0.100	-4.190
12	75.521	-0.810	-12.52	44	318.681	0.110	-4.079
13	82.083	-0.773	-13.30	45	326.138	0.122	-3.957
14	88.679	-0.739	-14.04	46	333.606	0.133	-3.824
15	95.307	-0.707	-14.75	47	341.085	0.144	-3.680
16	101.966	-0.676	-15.42	48	348.574	0.154	-3.526

17	113.407	4.106	-11.31	49	356.074	0.165	-3.361
18	120.238	-0.504	-11.82	50	363.584	0.175	-3.186
19	127.093	-0.480	-12.30	51	371.105	0.186	-3.000
20	133.972	-0.456	-12.75	52	378.636	0.196	-2.803
21	140.872	-0.435	-13.19	53	386.177	0.206	-2.597
22	147.793	-0.414	-13.60	54	393.728	0.216	-2.381
23	154.734	-0.394	-13.99	55	401.289	0.226	-2.155
24	161.693	-0.376	-14.37	56	408.859	0.235	-1.920
25	168.670	-0.358	-14.73	57	416.440	0.246	-1.674
26	175.665	-0.340	-15.07	58	424.030	0.255	-1.419
27	182.676	-0.324	-15.39	59	431.630	0.265	-1.154
28	189.703	-0.308	-15.70	60	439.239	0.274	-0.879
29	196.745	-0.293	-15.99	61	446.857	0.283	-0.596
30	203.803	-0.277	-16.27	62	454.485	0.293	-0.303
31	210.874	-0.264	-16.53	63	462.123	0.303	0.000

Table 7 – Simulated DNL and INL

The Integral Non-Linearity (INL) is calculated by drawing a line from the first to the last point on a graph of input current versus output code. Any deviation of the current values the actual ADC changes codes from the ideal results in INL. If a series of codes are reached sooner than they are supposed to

(i.e., negative DNL) this error can accumulate (thus the name integration) and a large INL error can result in the middle of the output range of the ADC. We see this phenomenon in the simulation results and part of this is the result of the design.

To improve the probability that the ADC remains monotonic additional space is needed where the different “ranges” of the ADC meet. For example, the first sixteen comparators use a different scale for the input current and during Monte Carlo simulation it was determined that if was possible for poor matching to cause “missing codes.” It was possible for the first comparator of the next group to activate before the final comparator of the previous group, or else the resulting outputs were very close. To counteract this the lengths of the current mirror transistors were adjusted to distance the current levels at which adjacent comparators in different scaling groups trigger.

13. Testing the design

In preparation for testing each chip goes through an extensive preparation process. The actual silicon die has multiple designs so only specific pads are bonded out (i.e., connected to the padframe so we can use them) depending on what design we are testing. Once the chip is connected to the padframe it is soldered to a custom PCB that is used to support external components (resistors, capacitors, and power regulators) and connectors. Two power regulators supply 1.8V DC to the analog and digital/buffer pins of the chip to try to isolate the sensitive analog components. Capacitors are used to decouple the power supply rails. The PCB (version 2 notation is at top right corner of the board) can be seen in figure 46 with banana plugs on the left for power delivery and input currents.

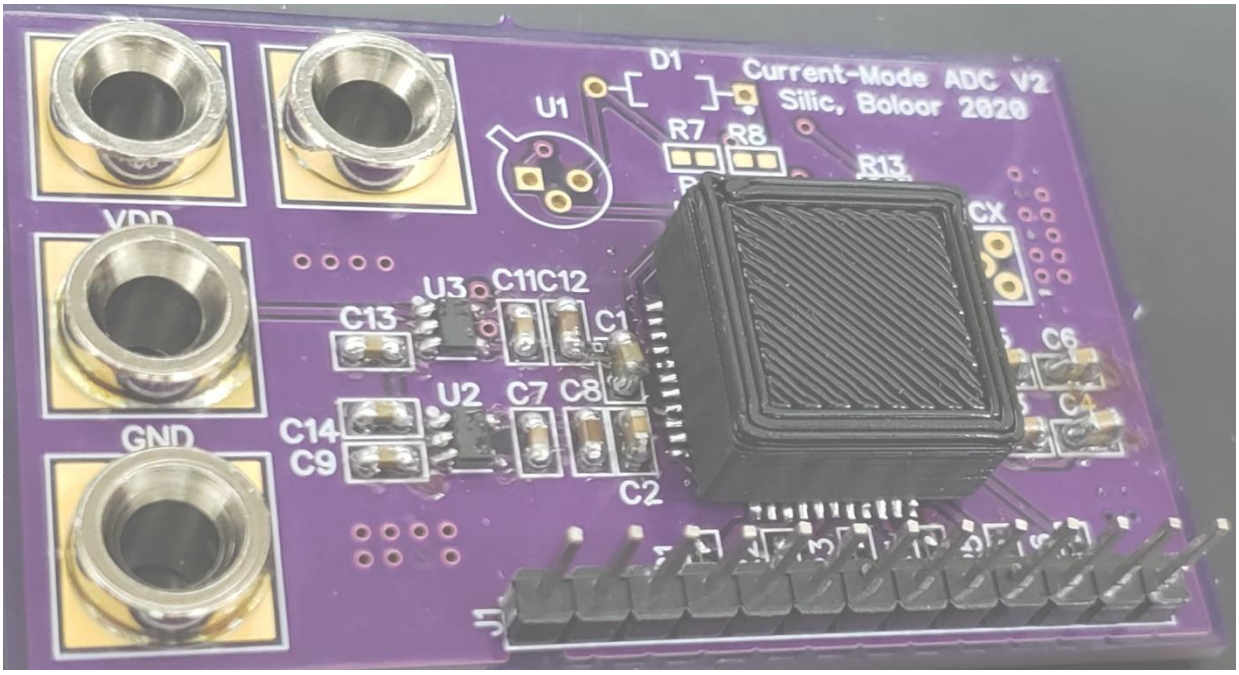

Figure 46 – V2 PCB for testing the design

Testing takes place by connecting a power supply as well as reference and input current sources. Initially an oscilloscope was used to determine if the circuit was functioning. We also employed a logic

analyzer to capture signals concurrently from all six output bits. A testing setup with the two current sources on the right is shown in Figure 47.

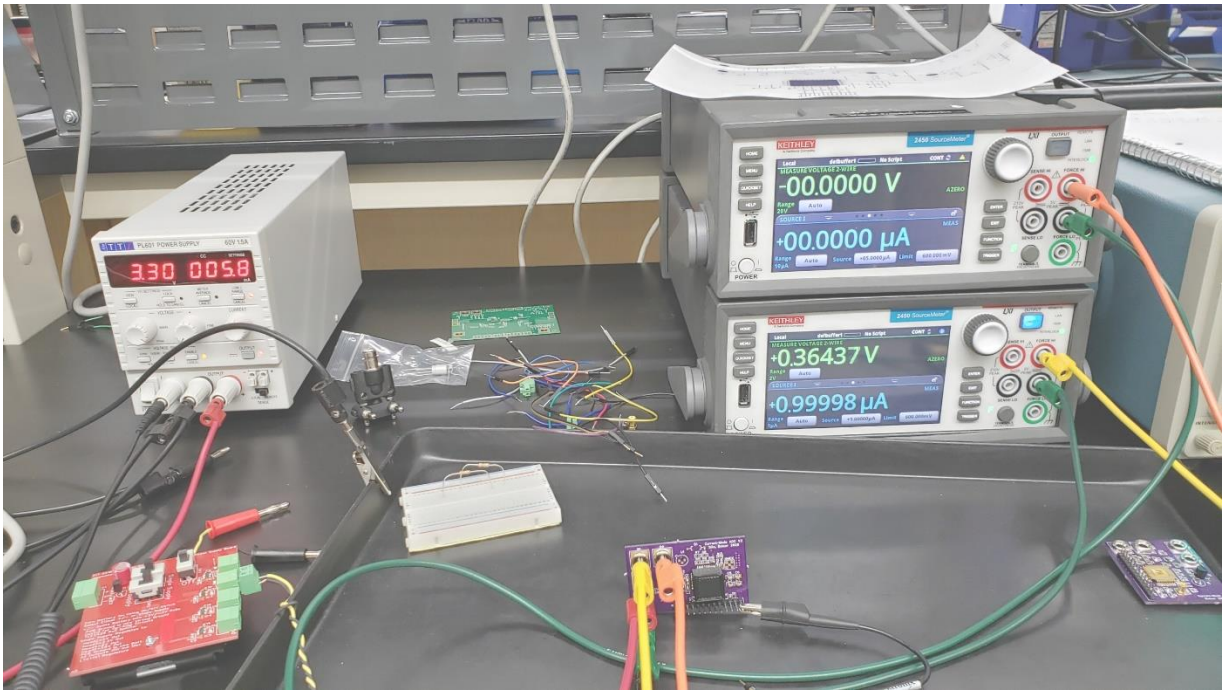

Figure 47 – Example Test Setup

Unfortunately, preliminary results indicate severe instability in the output bits whenever the output is not either all zeros or all ones. It appears that large amounts of noise from the buffers or digital logic feed into the analog comparators and cause absolute havok. For example, in Figure 48 below we can see the least significant bit (B0) rapidly varying. The frequency of the variation can exceed 100MHz which at least demonstrates the effectiveness of our buffers. This screenshot is not an outlier and in many cases even bits up the scale toward the MSB have just as much instability. Again, this is unfortunate because we would expect the comparators that are far away from switching to be more stable, leading to only the least significant bits exhibiting oscillation.

The only way to not have this instability is to have the input current be very small or large compared to the reference. If the reference current is 1µA then the outputs will be quietly zero when the input

current is zero. With a reference current of 2µA and an input of280µA the output is stable with all ‘ones.’ Note that this is a ratio of 140 in the input current where the circuit was designed to be roughly linear (i.e., anything past ~64 times the reference current should lock the output high.

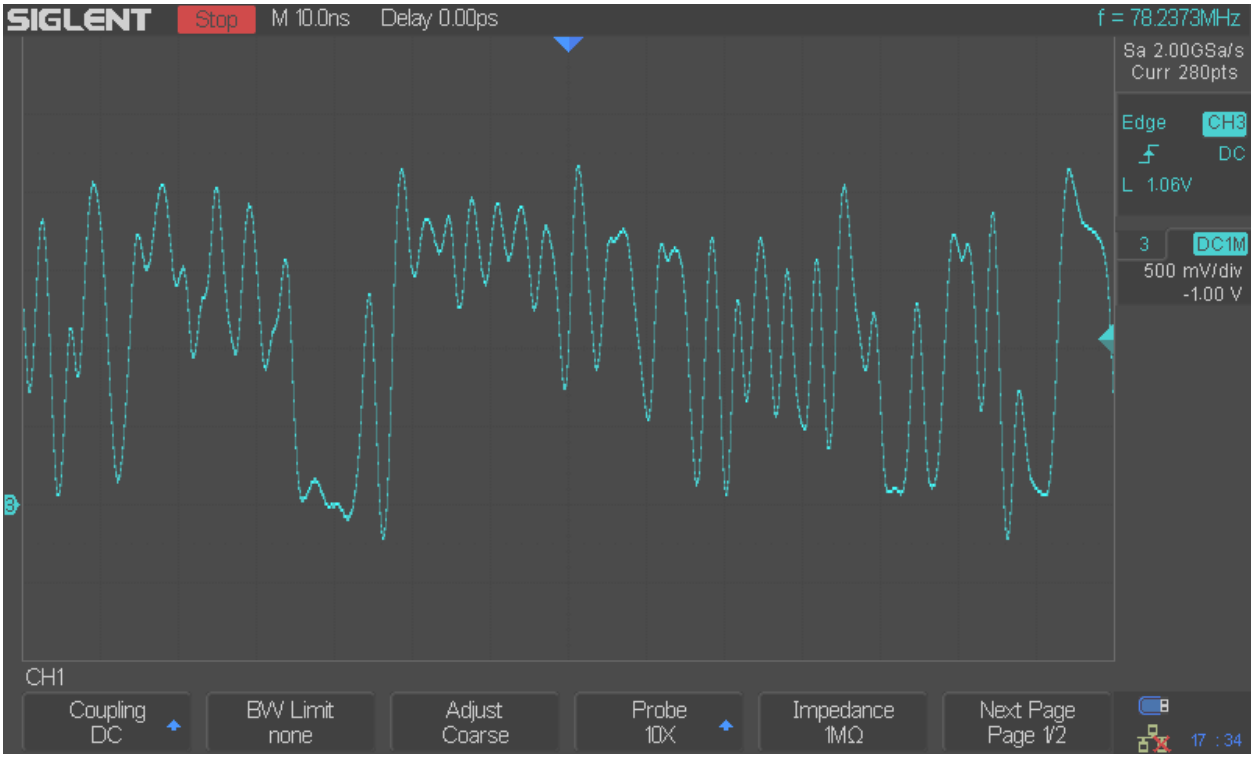

Figure 48 – Instability with B0 output

Measurements of the voltage on the power supply rails reveal variation with peak-to-peak amplitudes in excess of 100mV. A measurement on the C4 capacitor in Figure 49, which is connected to the digital supply VSS, shows variation of 140mV in a brief spike of “noise.”

One idea was that possibly the power supply is introducing noise from the power line, or perhaps a ground loop is forming with the current sources. In order to test this theory the test PCB was powered by a battery pack source. Table Another obvious fix would be to add additional decoupling capacitors to the board. SMD capacitors are preferred because of their low parasitics (they work at high frequencies). Figure 50 illustrates how with some creative soldering work SMD capacitors can be stacked on top of

each other. We also added two 100µF solid capacitors in a mostly futile attempt to smooth the power delivery.

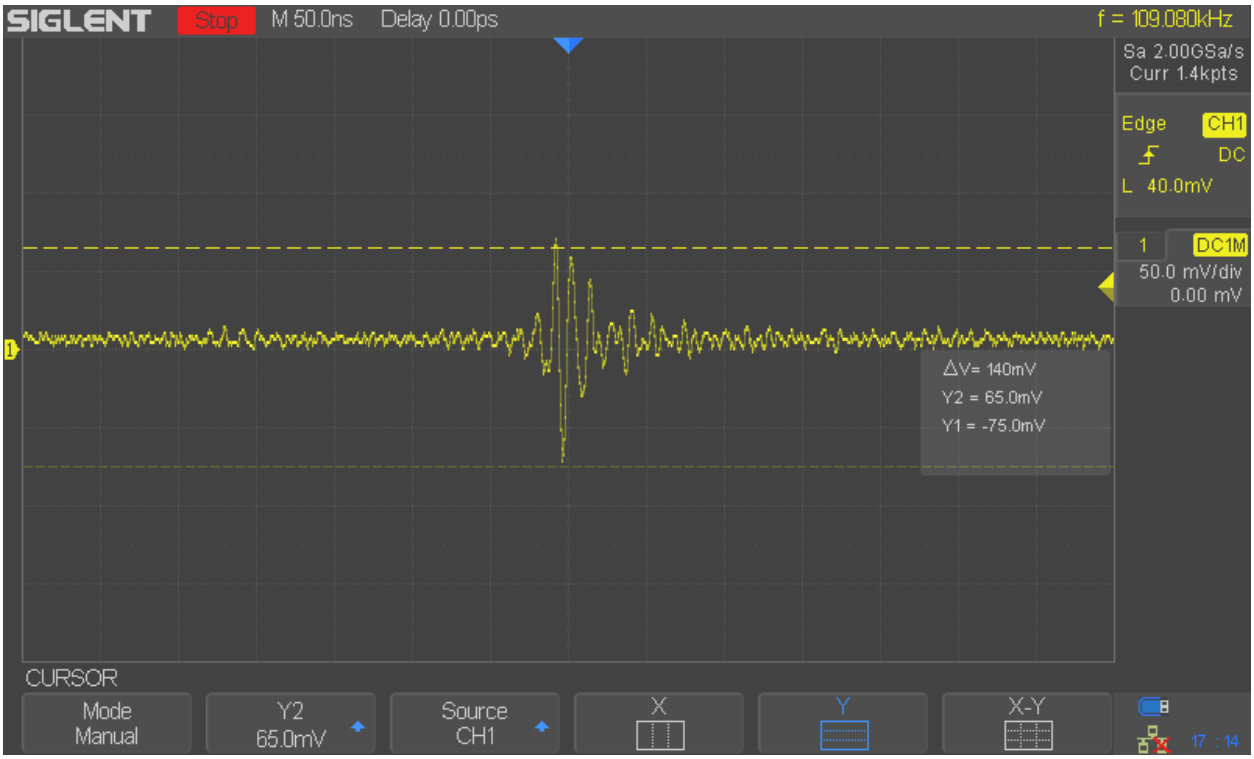

Figure 49 – Voltage ripple

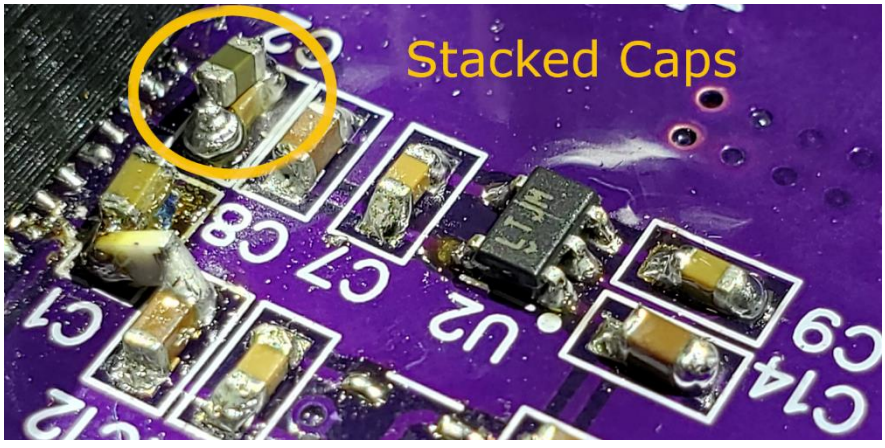

Figure 50 – Stacking SMD Capacitors

Figure 51 shows the board with added 100µF capacitors.

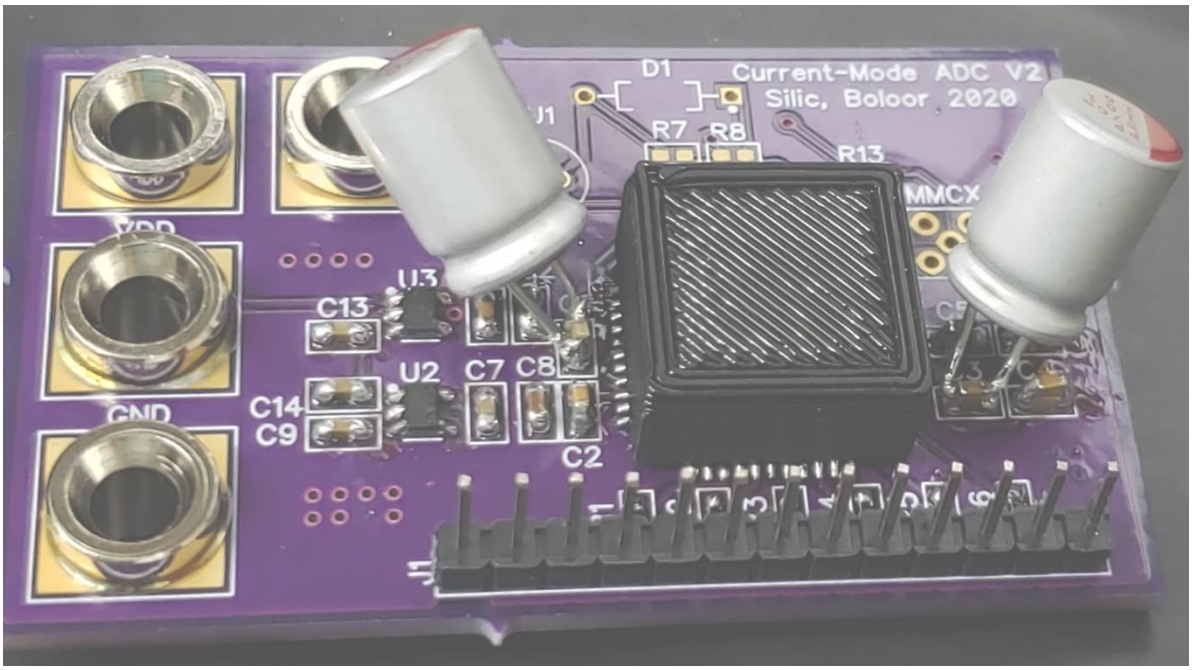

Figure 51 – New Capacitors

After testing the chip with power supply, battery source, and finally adding the through-hole capacitors we get the results in Table 8. The battery power supply actually has worse ripple on the supply rails, possibly due to the strange configuration of regulators feeding regulators. (The battery supply uses linear regulators to supply 3.3V to the PCB, which of course has its own regulators to produce our 1.8V supply.) We are pleased to report that the capacitors have reduced supply ripple overall. In many cases it appears the ripple is reduced by up to 50%. However, even 90mV of ripple is far too much for comfort when dealing with analog electronics. During this testing process the outputs of the chip remained as unstable as before.

Ref./Input Current	Bench Power Supply	Battery Power Supply	Bench Power w/ New Caps
2µA/0.1µA	160/155	182/150	105/92
2µA/2µA	108/150	165/150	101/97
2µA/50µA	160/110	155/120	108/129
2µA/120µA	132/152	160/200	110/129

Table 8 – Peak-to-Peak ripple (mV) on C1/C5 (Analog/Digital supply)

One possibility of extracting useful data from the output of this chip is inspired by noise-shaping modulators. If we treat the unwanted interference as noise then we have a faint signal at a low frequency or DC that we can extract by filtering out the high frequencies. One way to implement this low pass behavior is perhaps better known as averaging. This depends on the assumption that the "interference" takes the form of nice random (i.e., white) noise without tones in our bandwidth of

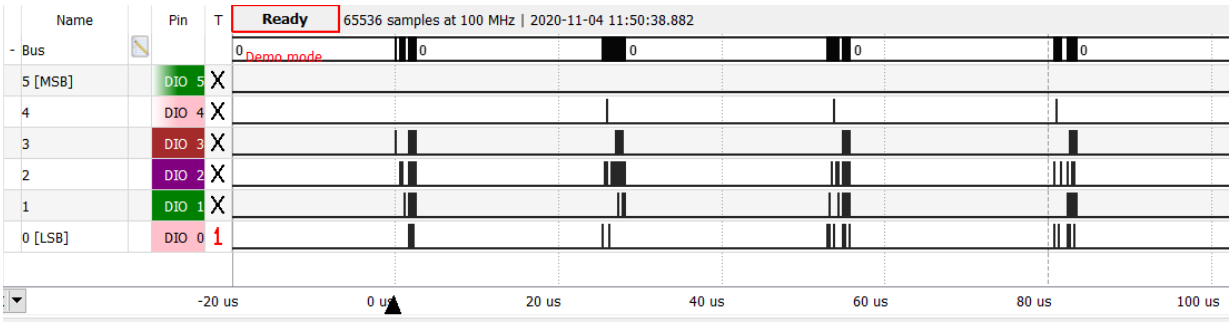

Figure 52 – Digital Logic output with 8µA/8µA input

interest. Given that our input is DC with a frequency of zero our assumptions should be reasonable.

In pursuit of this idea, consider the data displayed by the logic analyzer in Figure 52. With an 8µA current for both the reference and input we see the five least significant bits exhibit instability. However, notice the pattern where approximately every 20µs we have a brief period of oscillation

followed by a stable output for about 85% of that cycle. It's difficult to understand what could cause this behavior but it is clear that the design will never work well with this issue. A probable cause is power supply fluctuations at a frequency of approximately

$$\frac{1}{30\mu s} = 33\text{kHz}$$

due to RLC characteristics of the power delivery traces.

The logic analyzer is able to record samples on multiple channels at a sample frequency of 100MHz. This means it is taking samples every 10ns and storing them. It's connection to the host computer consists of a USB cable which is too slow to transfer data at the native sample rate. Instead we record sequences of data and store them locally before offloading to a computer. We record sequences of 65536 samples and export them to a .csv (Comma Separated Value) file. This file can then be processed by a Python script (See Appendix A). For now we do the simplest analysis and simply average all the values recorded. Table 9 show the raw results for various values of reference and input current. These results come from version 2 (V2) of the test board.

Ref./Input current (µA)	Average Output	Ref./Input current (µA)	Average Output
2/4	0.1058	8/112	6.8123
2/10	2.2673	8/128	32.6517
2/20	17.3256	8/144	39.1427
2/30	30.3081	8/160	40.1623
2/40	37.1794	8/184	43.7563
2/50	41.1782	8/208	45.6716

2/100	54.1567	8/232	42.3547
6/300	52.4649	8/256	21.5041
8/3	0.0258	8/280	21.9651
8/8	0.0702	8/304	22.4819
8/16	0.1778	8/328	22.5382
8/24	0.4123	8/360	22.7446
8/32	1.2523	8/392	22.9509
8/40	1.9613	8/424	22.7618
8/48	1.5189	8/456	22.5612
8/56	2.3527	8/488	22.4924
8/64	2.9358	8/512	22.7026
8/72	3.5549	12/38	0.8837
8/80	4.4478	16/200	12.9291
8/96	7.8962		

Table 9 – Average output vs. input

One interesting thing that emerges is that after a certain point (about 208µA) the average output actually drops back to 22 and basically doesn't change significantly as the input current increases beyond that. You can see this graphically in Figure 53 where the beginning looks very nice up until about 100µA when there is suddenly a huge spike and then a flatline as noted above. This is clearly not the

behavior we want and is clearly nowhere near ideal DNL or INL. Perhaps there is a “limit cycle” or mechanism that prevents the output value from averaging near where it should. Unless there is a way to

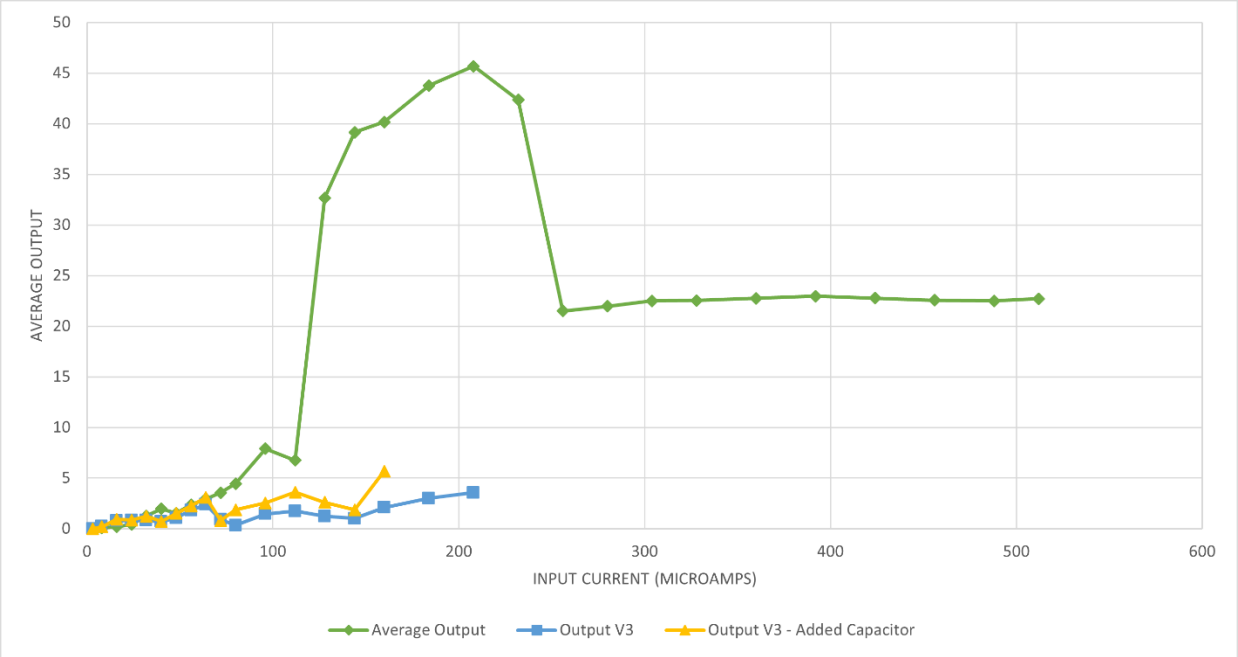

Figure 53 – Average Output Value vs. Input Current

suppress these oscillations in the output there is little chance of this design revision achieving promising performance.

Figure 53 also shows data from the next revision of the test board. To avoid damaging the input current mirrors the input current is not tested all the way to 512µA. As discussed in the next section the third version of the test board has added capacitors. While testing this board we noted large oscillations on the current inputs. A capacitor was added to the reference input for the third set of measurements (yellow triangles). Figure 54 shows a detailed view of these measurements near the origin.

Although there are hints of expected behavior near the origin we quickly lose any semblance of useful results once the input current becomes larger. The lack of even moderate stability for the more

significant bits of the output indicate that power supply stability is not the primary cause of the rapid changes in the output codes.

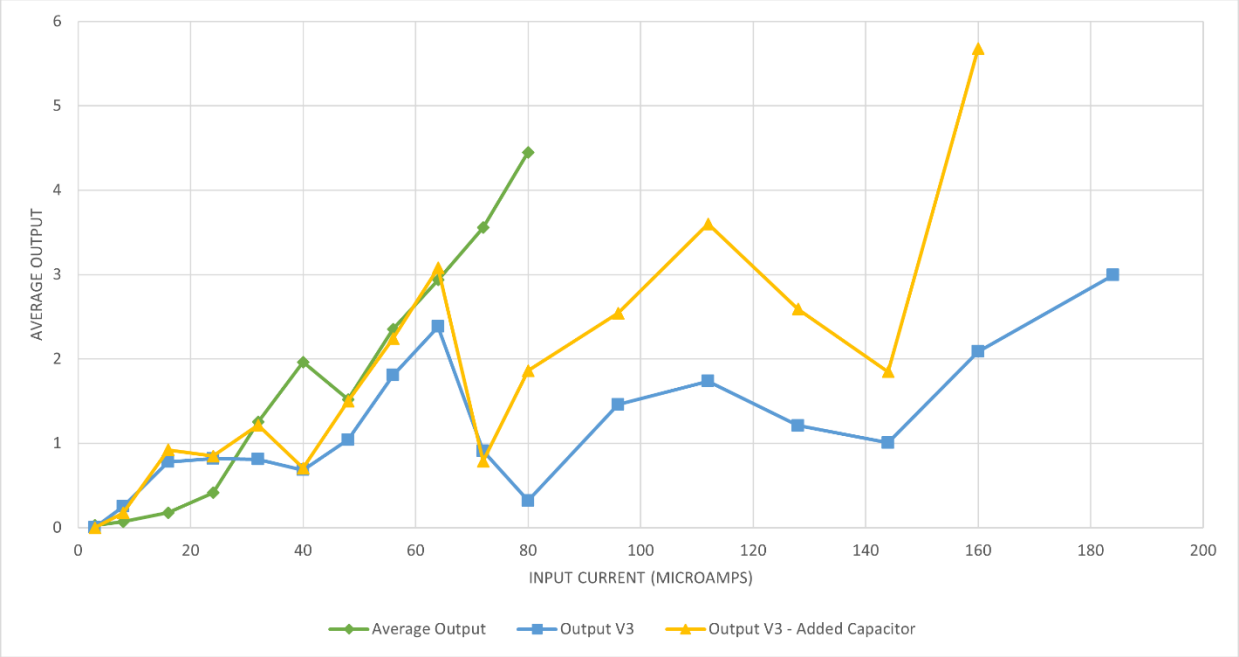

Figure 54 – Detail view of output vs. input current

To demonstrate this consider the simulation in Figures 55 and 56. A 60MHz, 400mV amplitude sine wave source adds “noise” to the VDD rail. We can see that as the input current increase toward a final value of 35µA there is some vacillation in the lower output bits. However, this never spreads beyond the comparator adjacent to the correct level. When the second comparator is switching its output due to noise on the input we see that the first comparator is already made a decision and does not change its output state at anywhere near the same point in the ramp. This makes it seem plausible that only large changes in the input and reference currents would be able to explain the massive irregularities we observe in our output bits.

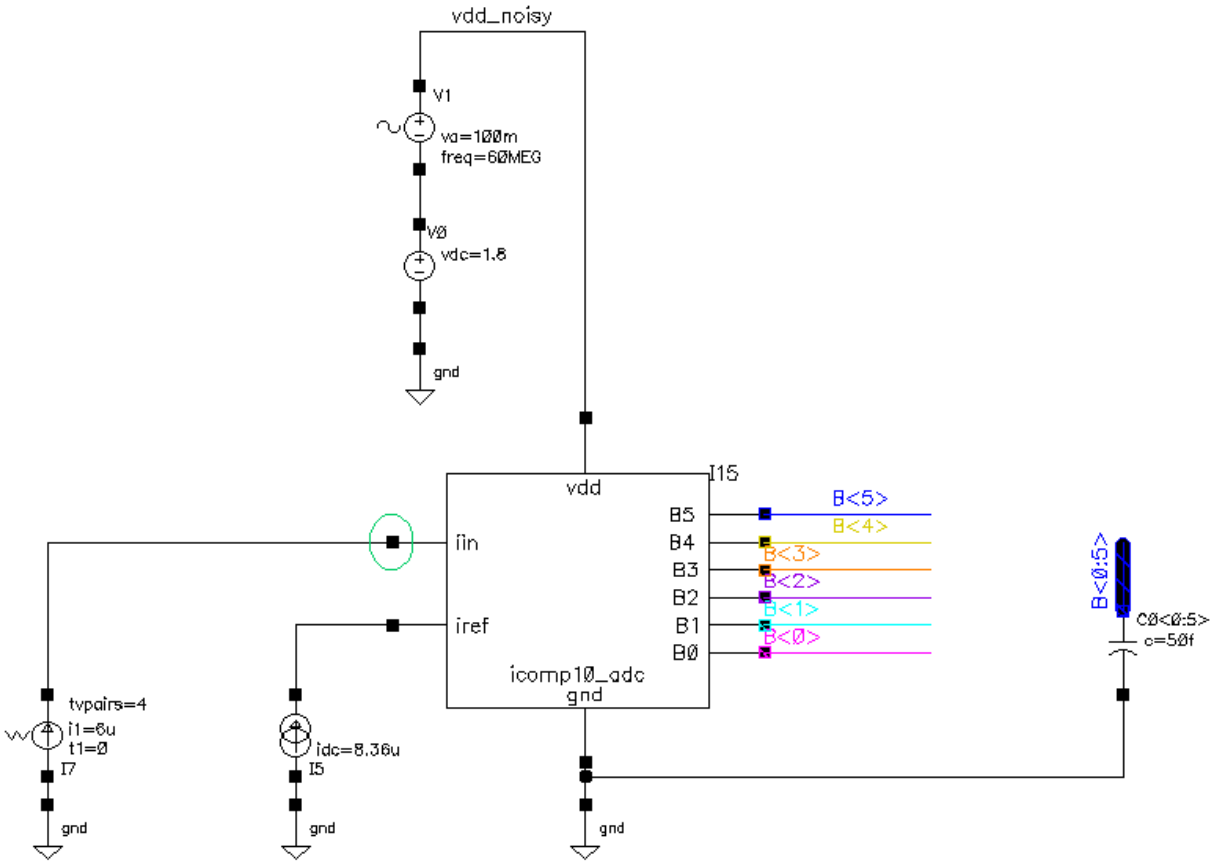

Figure 55 – VDD noise test schematic

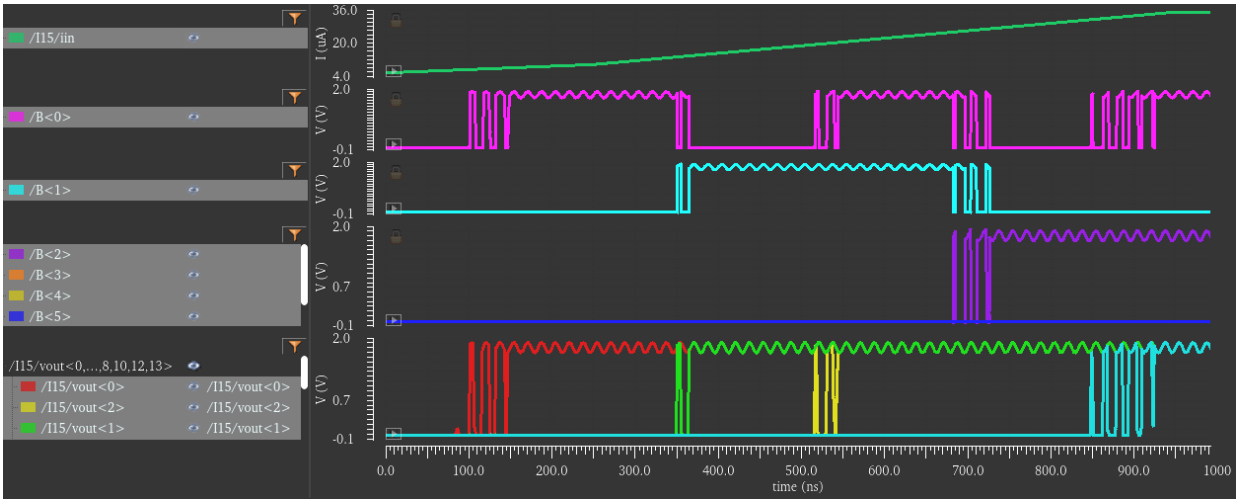

Figure 56 – VDD noise test simulation results

14. Testing with the version 3 test board

In an effort to add additional decoupling capacitors a new revision of the test board was developed. One feature of this “V3” revision was additional capacitors on the bottom of the board, placed very close to the padframe’s power supply connections. Figure 57 shows the bottom of the new board with three SMD capacitors.

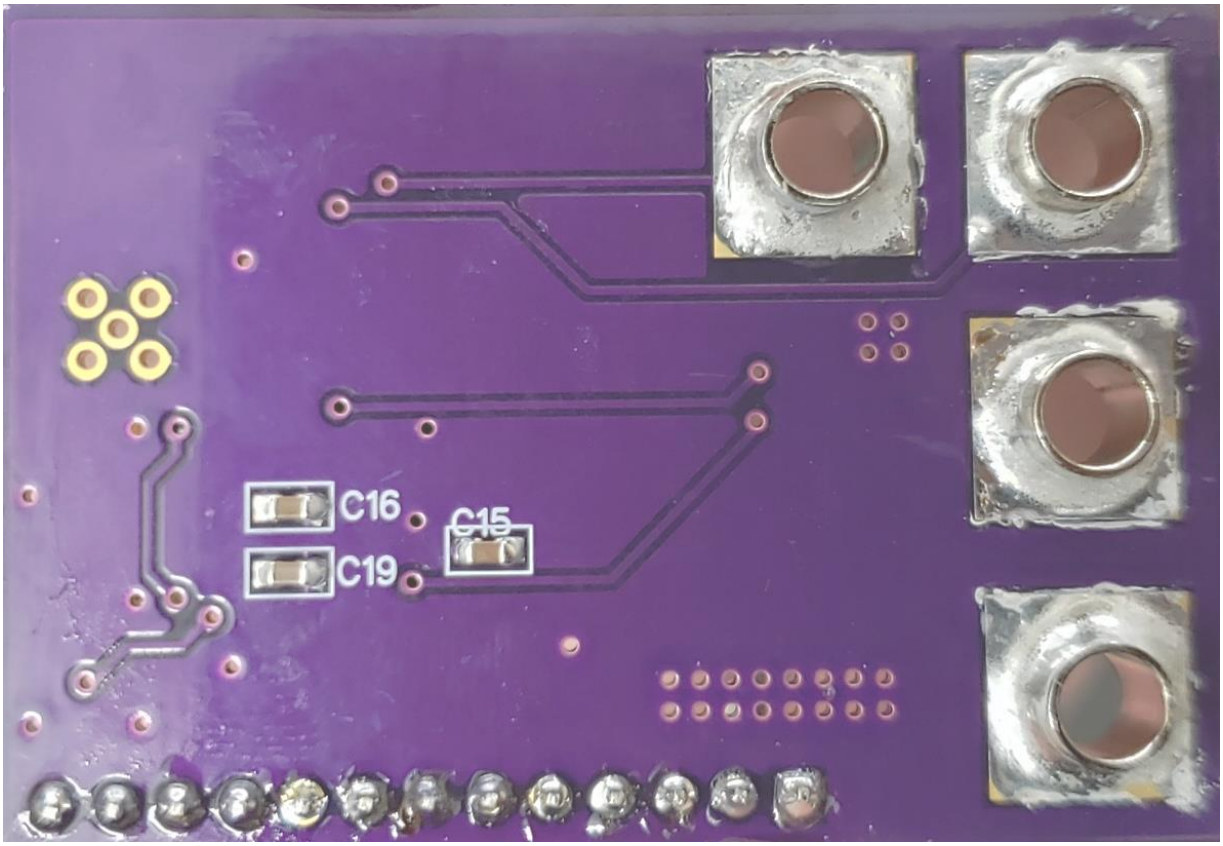

Figure 57 – Extra capacitors on bottom of V3 test board

One of the more interesting results is the transient response to a step current input. All of our other results have been with a static DC input. This was an expedient to extract any useful data that indicates an operational ADC but ideally we would like to test the response time of our circuit. The results seem to indicate we are pushing the boundaries of how fast we can generate off-chip stimuli. It is likely that parasitic inductance and capacitance limit how quickly a current input can rise. The circuit is designed to

work with an on-die APD array which is not subject to the limitations of padframes and external wiring. A test APD was fabricated on the same design but no integrated test results are currently available.

The basic test setup is shown in Figure 58. A simple contact connects the 3.3V source to the current

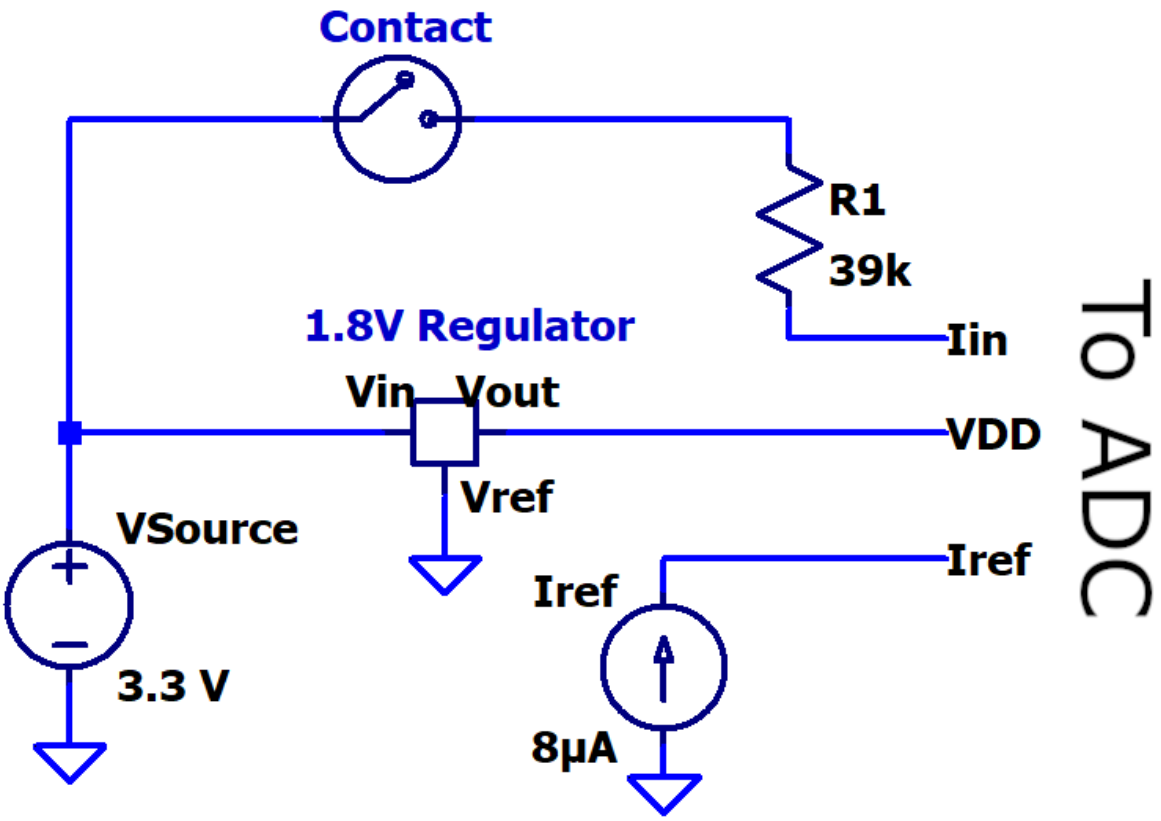

Figure 58 – Transient testing diagram

input through a 39kΩ resistor. This limits maximum current input to

$$I = \frac{V}{R} = \frac{3.3V}{39k\Omega} = 84.6\mu A$$

Note the actual maximum current would be a bit lower as the input voltage rises to somewhere over 400mV for these levels of current (See input voltage discussion below). The current input calculated above should be more than sufficient to trigger an output from the least significant bit, B0. We place

oscilloscope probes to measure the voltage at B0 as well as at the input current port. After setting the oscilloscope trigger and closing the contact we get the result shown in Figure 59.

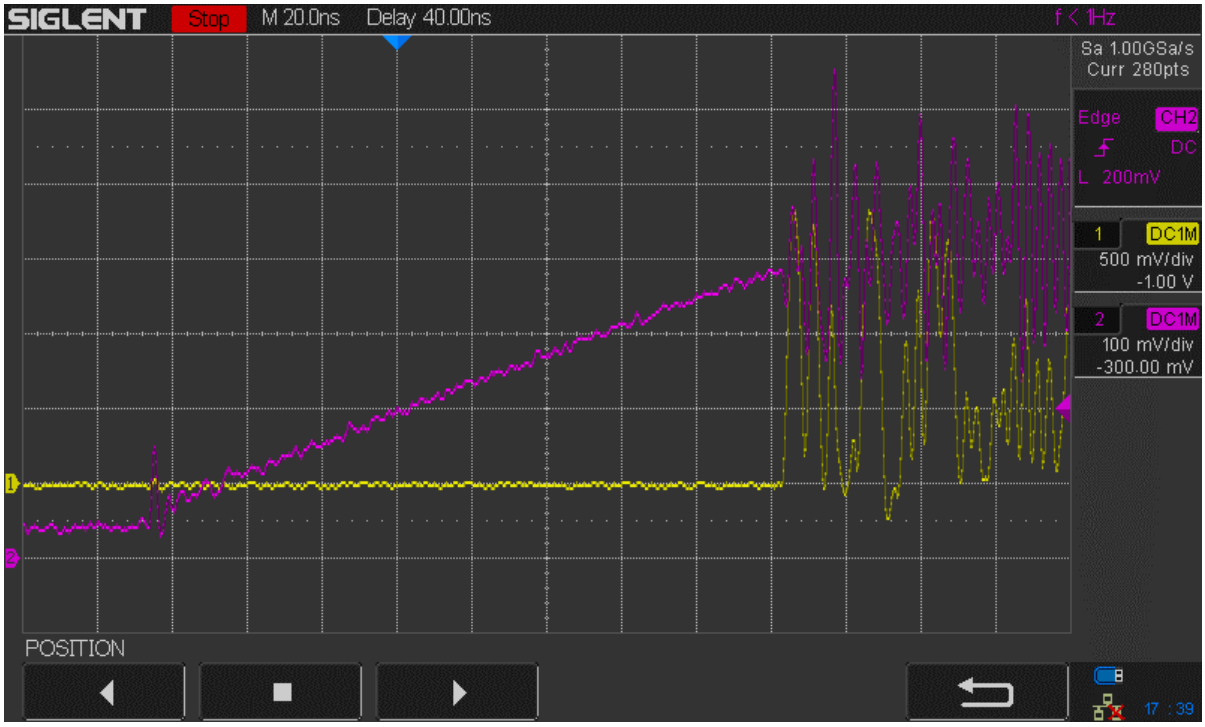

Figure 59 – Slow transient input

One interesting characteristic of this screenshot is the almost perfect linear ramp observed before the output buffers activate. After that point we have highly chaotic fluctuations which lead to an unstable output. We can estimate the parasitic capacitance on the input node by observing that the input current through the resistor is roughly

$$\mathrm{I} = \frac{\mathrm{V}}{\mathrm{R}} = \frac{3.3\mathrm{V} - 0.2\mathrm{V}}{39\mathrm{k}\Omega} \approx 80\mu\mathrm{A}$$

Working from the formula for charge on a capacitor, Q = CV, noting that current is change in charge per unit time (Coulomb/second), and estimating from Figure 59 that the voltage increases from 80mV to 380mV in 160ns, we can calculate:

$$C = \frac{Q}{V} \text{ or } \frac{\Delta Q}{\Delta V} = \frac{80\mu C/s}{(380mV - 80mV)/160ns} = \frac{80\mu F}{1.875x10^6} = 42.7pF$$

It is very reasonable that 40pF of stray capacitance could exist on the input node. To illustrate the difficulty in measuring the performance of our circuit when the smallest amount of extra parasitic capacitance will prevent accurate results consider the results in Figure 60. Here we have also displayed the second bit (B1) on an additional channel.

In the previous measurement the orange cable seen in Figure 47 was left connected to the test board during the transient experiment. We were not using the Keithley 2450 current source and the cable was simply left open at the opposite end. When this cable was removed from the circuit we see that the current ramp is roughly twice as fast (80ns). We also see in Figure 60 that current increases from approximately 100mV to 400mV. (Note that channel 2, the pink trace representing the input voltage, has a scale of 200mV/div. Figure 59 had a scale of 100mV/div.)

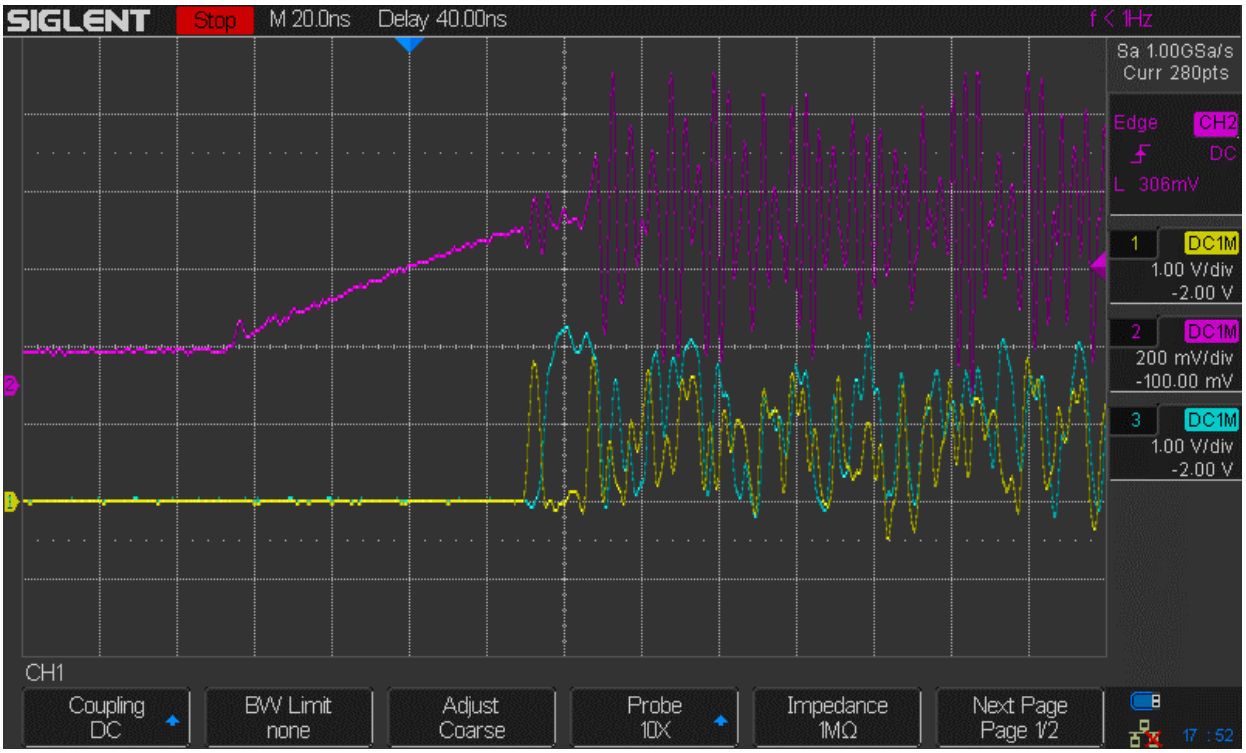

Figure 60 – Faster transient input

For this experiment we will assume average input voltage of 200mV even though results show 300mV might be a better choice. This will bias our assumed current to be larger, which will result in a larger calculated capacitance. This is a conservative assumption to avoid exaggerating the change in capacitance from Figure 59.

The calculated current is therefore again 80µA = 80µC/s. Using the method discussed above:

$$C = \frac{Q}{V} \text{ or } \frac{\Delta Q}{\Delta V} = \frac{80\mu C/s}{(400mV - 100mV)/80ns} = \frac{80\mu F}{3.75x10^6} = 21.3pF$$

This is exactly half of the previously calculated value but still prevents us from determining the exact transient response time of the ADC. Consider Table 10 below and the graph of the same data in Figure 61. If we assume the first comparator activates around 10µA then we can examine the time from the corresponding voltage of 386mV to the activation of the output. This is approximately 10ns but we should not place any confidence in this value given the uncertainty of what input current the ADC is actually experiencing given our external voltage measurements with the oscilloscope. We can probably deduce an upper bound of perhaps 60ns measuring from the start of the input voltage ramp

Input Current (µA)	Input Voltage (mV)
10	386
20	422
30	445
40	463
60	474
80	503
100	530

Table 10 – Input current vs. voltage

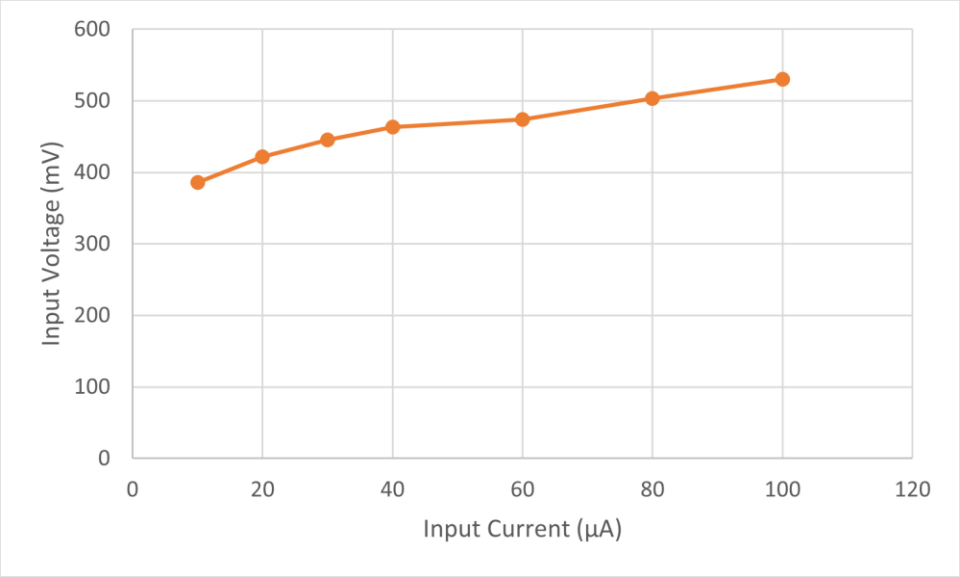

Figure 61 – Graph of input current vs. voltage

Note that a small variation in the input voltage results in a large change in the input current. This is expected and desired behavior indicating a low input impedance. As seen in Figure 60 and more directly in Figure 62, our oscilloscope measurements indicate large voltage oscillations on the input and reference current pins. For example, Figure 62 shows the voltage at the I_{in} pin with an 8µA input and reference current. The peak-to-peak oscillations exceed 300mV. Obviously this kind of variation could have a dramatic impact on the current going into the comparator current mirrors (. It is important to note that we cannot see the voltage or currents inside the chip and the oscilloscope probe itself adds capacitive load to the circuit. The magnitude of these variations indicates a serious problem regardless.

One possibility is that the current source meters are introducing this variation through their active source management, perhaps interacting with the resonant elements of the physical test board. However, we tested with passive resistors from the 3.3V supply to the two current inputs and found similar output instability. The only remaining possibility is to use the built-in APDs on the same die to supply the ADC's input current through a short bond wire. This setup would hopefully have an effective

way of supplying the reference current and would possibly lead to more hopeful results. As of this writing the described testing has not been done.

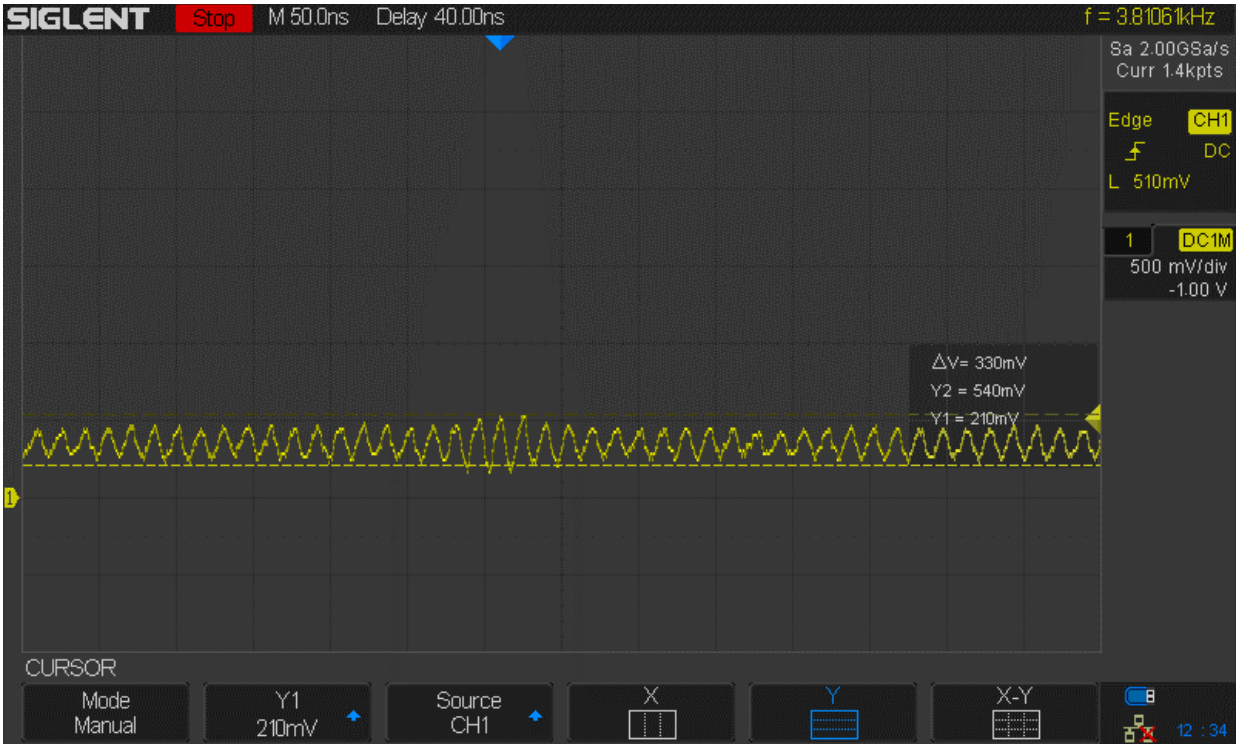

Figure 62 – Voltage oscillation at the I_{in} port

15. Conclusion

It is gratifying that the design worked, even if only on a limited basis. There were certainly things that could have been done better but one of the things that derailed this design seems to simply be the difficulties of analog design. It may have been a mistake to try to develop a complete ADC without also fabricating smaller test circuits. If we had done this then we could isolate problems with a specific part of the design and probably would have been able to more accurately characterize the comparators. It also seems like the output buffers may be too large for the capacitive loads we are driving in testing. If they were smaller we would see less power consumption in situations were the output is oscillating. An added benefit may be less power supply fluctuation on the test board side.

It was encouraging to see some positive results after averaging the outputs, showing that at least some parts of the circuit are working. With appropriate isolation measures, perhaps even a chiplet design, it seems possible that the proposed circuit could achieve the envisioned performance.

This research has been surprisingly hands-on. Besides the standard work learning Cadence there was also significant work both soldering and desoldering SMD components. Python scripts were very useful and easy to write for data processing. The effort trying to debug the chips high-frequency issues was also a very practical experience. Not many people can say they have designed, fabricated, and tested an integrated circuit of their (mostly) own design. The IC design experience will prove useful in a variety of analog and digital design situations.

Appendix A – Python code for reading .csv files

```
# Python 3 script to extract data from Digital Discovery .csv files
print("Extracting data from .csv files...")

import os
#from os import listdir

filenames = os.listdir(".\\CSV_Data") # CSV files must be stored in this folder
filenames.sort()

with open("AvgBusValues.txt", "w") as F2: # Output file
    for filename in filenames:
        if len(filename) < 5 or filename[-4:] != ".csv":
            print("Error, non-*.csv-file found: " + filename)
        else:
            fPath = ".\\CSV_Data\\" + filename
            print("Opening \'" + filename + "\'")
            with open(fPath) as F: # Open .csv file
                lines = F.readlines()
                listSize = len(lines)
                if listSize != 65537: # For now we only support 65536 samples (extra is column headers)
                    print("Unexpected size of",listSize)
                outputSum = 0
                numberList = []
                for i in range(1,listSize):
                    ln = lines[i].split(',')
                    BusValue = int(ln[1])
                    numberList.append([float(ln[0]), BusValue]) # Parse entry to time (float) and value
```

```
    outputSum += BusValue # Running total
avg = outputSum / 65536
print("Avg value: " + str(avg));
F2.write("Avg for " + filename + "\t\t" + "{:.4f}".format(avg) + "\n") # Write average to file
```

Appendix B – Python script for GIMP image processing

While developing the figures used in this thesis I needed to modify screenshots of Cadence Virtuoso's schematic capture program. The default black background with blue and green wires and components was not ideal. I wrote a python plugin/script that selects regions of certain colors and changes them to black or white to produce a clean schematic as seen in many figures. The code is presented here just in case someone finds it useful. The script (FileName.py) is placed in:

'C:\Users\<username>\AppData\Roaming\GIMP\2.10\plug-ins'

```
# Try it out!

from gimpfu import *

def SingleImgMod(image, drawable):
    # List of replace colors
    replaceables = [
        [gimpcolor.RGB(0,0,1), gimpcolor.RGB(255,255,255)],
        [gimpcolor.RGB(254,0,0), gimpcolor.RGB(0,0,0)],
        [gimpcolor.RGB(56,190,254), gimpcolor.RGB(0,0,0)],
        [gimpcolor.RGB(198,198,198), gimpcolor.RGB(0,0,0)],
        [gimpcolor.RGB(0,204,102), gimpcolor.RGB(0,0,0)],
        [gimpcolor.RGB(254,254,254), gimpcolor.RGB(0,0,0)],
        [gimpcolor.RGB(254,128,0), gimpcolor.RGB(0,0,0)],
        [gimpcolor.RGB(216,204,0), gimpcolor.RGB(0,0,0)],
```

```
        [gimpcolor.RGB(254,254,0), gimpcolor.RGB(255,255,255)],
        [gimpcolor.RGB(140,140,166), gimpcolor.RGB(255,255,255)]
    ]

    #parent function for changing colors on single image
    pdb.gimp_image_undo_group_start(image)
    pdb.gimp_context_push() #Save old context
    pdb.gimp_context_set_antialias(FALSE)
    pdb.gimp_context_set_sample_threshold(0) # Don't sample similar colors
    pdb.gimp_context_set_feather(FALSE)

    for ColorPair in replaceables:
        pdb.gimp_selection_none(image) # If no color (image doesn't have it) is selected we don't want to start filling in random color
        pdb.gimp_context_set_foreground(ColorPair[1]) # Color we are changing TO
        pdb.gimp_image_select_color(image, CHANNEL_OP_REPLACE, drawable, ColorPair[0]) # Color we are changing FROM
        is_empty = pdb.gimp_selection_is_empty(image)
        if is_empty == False:
            pdb.gimp_drawable_edit_fill(drawable, FILL_FOREGROUND)
        #else:
            #gimp.message("No color")

    pdb.gimp_selection_none(image)
    pdb.gimp_context_pop() # Restore context
```

```
    pdb.gimp_image_undo_group_end(image)


def SingleSimMod(image, drawable):
    # List of replace colors
    replaceables = [
        [gimpcolor.RGB(0,0,0), gimpcolor.RGB(255,1,255)], # Temp color
        [gimpcolor.RGB(254,0,0), gimpcolor.RGB(0,0,0)],
        [gimpcolor.RGB(56,190,254), gimpcolor.RGB(0,0,0)],
        [gimpcolor.RGB(198,198,198), gimpcolor.RGB(0,0,0)],
        [gimpcolor.RGB(0,204,102), gimpcolor.RGB(0,0,0)],
        [gimpcolor.RGB(254,254,254), gimpcolor.RGB(0,0,0)],
        [gimpcolor.RGB(254,128,0), gimpcolor.RGB(0,0,0)],
        [gimpcolor.RGB(216,204,0), gimpcolor.RGB(0,0,0)],
        [gimpcolor.RGB(254,254,0), gimpcolor.RGB(255,255,255)],
        [gimpcolor.RGB(255,1,255), gimpcolor.RGB(255,255,255)]
    ]

    #parent function for changing colors on single image
    pdb.gimp_image_undo_group_start(image)
    pdb.gimp_context_push() #Save old context
    pdb.gimp_context_set_antialias(FALSE)
    pdb.gimp_context_set_sample_threshold(0) # Don't sample similar colors
    pdb.gimp_context_set_feather(FALSE)

    for ColorPair in replaceables:
```

```
        pdb.gimp_selection_none(image) # If no color (image doesn't have it) is selected we don't want to start filling in random color
        pdb.gimp_context_set_foreground(ColorPair[1]) # Color we are changing TO
        pdb.gimp_image_select_color(image, CHANNEL_OP_REPLACE, drawable, ColorPair[0]) # Color we are changing FROM
        is_empty = pdb.gimp_selection_is_empty(image)
        if is_empty == False:
            pdb.gimp_drawable_edit_fill(drawable, FILL_FOREGROUND)
        #else:
            #gimp.message("No color")

    pdb.gimp_selection_none(image)
    pdb.gimp_context_pop() # Restore context
    pdb.gimp_image_undo_group_end(image)

register(
    "python-fu-SingleImgMod",
    "Modify image colors",
    "Change black background to white on circuit diagram",
    "", "", "2020",
    "Mod Single Image",
    "*", #Type of image
    [
        (PF_IMAGE, "image", "Input image", None),
        (PF_DRAWABLE, "drawable", "Input drawable", None)
```

```
    ],
    [],
    SingleImgMod,
    menu="<Image>/Processing")

register(
    "python-fu-SingleSimMod",
    "Modify image colors",
    "Change sim background to white on circuit diagram",
    "", "", "2020",
    "Mod Sim Image",
    "*", #Type of image
    [
        (PF_IMAGE, "image", "Input image", None),
        (PF_DRAWABLE, "drawable", "Input drawable", None)
    ],
    [],
    SingleSimMod,
    menu="<Image>/Processing")

main()
```

References

[1] L. Chen, B. Shi, and C. Lu, “A robust high-speed and low-power CMOS current comparator circuit,” *IEEE APCCAS 2000, 2000 IEEE Asia-Pacific Conference on Circuits and Systems, Electronic Communication Systems.* (Cat. No.00EX394), 174–177. https://doi.org/10.1109/APCCAS.2000.913435

[2] “Time of flight,” Wikipedia. https://en.wikipedia.org/wiki/Time_of_flight. (Accessed Nov. 4, 2020)

[3] "TowerJazz Expands its High Speed SiGe, SOI and RF CMOS Design Kits for use with Agilent Technologies’ Advanced Design System 2011 Software Release," towersemi.com. http://towersemi.com/wp-content/uploads/2018/08/pr10062011.pdf (Accessed Nov. 4, 2020)

[4] V. Vinayaka, “Analysis and Design of Analog Front-End Circuitry for Avalanche Photodiodes (APD) and Silicon Photo-Multipliers (SiPM) in Time-of-Flight Applications,” *ProQuest Dissertations Publishing,* 2018.

[5] H. Träff, “NOVEL-APPROACH TO HIGH-SPEED CMOS CURRENT COMPARATORS,” *Electronics Letters*, 28(3), 310–312. https://doi.org/10.1049/el:19920192

[6] R. J. Baker, *CMOS Circuit Design, Layout, and Simulation*, Fourth Edition. Wiley-IEEE Press, 2019.

[7] “De Morgan's laws,” Wikipedia. https://en.wikipedia.org/wiki/De_Morgan%27s_laws (Accessed Nov. 5, 2020)

[8] S. Namboodiri, G. Arteaga, J. Skelly, F. Mata-carlos, A. Roy, and R. J. Baker, "A Current-Mode Photon Counting Circuit for Long-Range LiDAR Applications,". 2020 IEEE 63rd International Midwest Symposium on Circuits and Systems (MWSCAS), 146–149. https://doi.org/10.1109/MWSCAS48704.2020.9184584

[9] M. d. M. Silva, and L. B. Oliveira, "Regulated Common-Gate Transimpedance Amplifier Designed to Operate With a Silicon Photo-Multiplier at the Input," IEEE Transactions on Circuits and Systems. I, Regular Papers, 61(3), 725–735, March 2014. https://doi.org/10.1109/TCSI.2013.2283992

[10] R. Ma, M. Liu, H. Zheng, and Z. Zhu, "A 66-dB Linear Dynamic Range, 100-dB·Ω Transimpedance Gain TIA With High-Speed PDSH for LiDAR," IEEE Transactions on Instrumentation and Measurement, 69(4), 1020–1028, 2020. https://doi.org/10.1109/TIM.2019.2907776

[11] S. Kim, S. Cho, and S. M. Park, "Dual-mode CMOS feed-forward transimpedance amplifier for LADARs," Electronics Letters, 50(23), 1678–1679, 2014. https://doi.org/10.1049/el.2014.2678

Curriculum Vitae

Jason Silic

Contact Information:

Email: jason.silic.ee@gmail.com

Webpage: http://cmosedu.com/jbaker/students/jason/jason.htm

Education:

University of Nevada, Las Vegas

Bachelor of Engineering – Electrical Engineering, 2015

Senior Design Project:

Compact Integrated Processor – Design and fabrication of a SRAM memory + simple CPU.

Work Experience:

Software Engineer – EC2 Software Solutions, Inc. 2015-2019

Thesis Title:

Design and Fabrication of a 6-bit Current-Mode ADC for LIDAR and High-Speed Applications

Thesis Examination Committee:

Committee Chair, Dr. R. Jacob Baker, PhD

Committee Member, Dr. Robert A. Schill Jr., PhD

Committee Member, Dr. Biswajit Das, PhD

Graduate College Representative, Dr. Monika Neda, PhD