Offset Correction for Swept Threshold Ultra-Wide-Band Sampling

Dino Vele

Thesis submitted for the degree of
Master in Electronics and Computer Science, Microelectronics
60 credits

Department of Physics
Faculty of mathematics and natural science

UNIVERSITY OF OSLO

Autumn 2019

Offset Correction for Swept Threshold Ultra-Wide-Band Sampling

Dino Vele

Offset Correction for Swept Threshold
Ultra-Wide-Band Sampling
http://www.duo.uio.no/

Preface

This Master's thesis was submitted for the degree of M. Sc. at the University of Oslo, dept. of Informatics. The work has been carried out in the period between spring 2018 and fall 2019 at the NANO group under the supervision of Professor Dag T.Wisland. and Professor Jørgen A. Michaelsen.

Oslo, August 12, 2019

Dino Vele

Acknowledgment

I would like to thank my supervisors, Dag T. Wisland and Jørgen A. Michaelsen, for guiding me through this master project.

I would also like to thank Olav S. Kyrvestad, Tohid M. Khanshan and Krisitan G. Kjelgård for their time, help me with technical support.

And finally, I would like to thank my family and friend for supporting me through this work.

Dino

TABLE OF CONTENTS

List of Figures

LIST OF TABLES

1 INTRODUCTION

With the advancement of technology scaling still following Moore's law, transistors are now being produced in ten-nanometre technology processes. The scaling of transistors, or more precisely MOSFETs (*metal-oxide-semiconductor field-effect-transistors*), has made it possible to produce compact, high-processing, high-speed and high-capacity devices such as cell phones, laptops, computers and so on. Transistors do benefit from scaling by improving their speed (due to less parasitic capacitance), their power consumption (due to lower operating voltages), and by reducing their size which again leads to better cost efficiency. However, since the technology generations reached the sub-micron era, previous higher-order effects have become more dominant and are getting more dominant with scaling. Effects such as subthreshold leakage, tunnelling current, velocity saturation and process variation have led to new design approaches being used in order to handle these effects. This master thesis will focus on the latter one of the effects, process variation, and the challenges stemming from it.

Mismatch, caused by process variation, has become a growing challenge in both digital and analog circuit design. According to Pelgrom's research [1], the potential of variation between two parameters increases with the shrinking of the parameters. This mean that the smaller a parameter, such as a metal line, gets, the greater the chances of the parameter deviating from the designed shape is. Furthermore, Pelgrom's findings also indicate that this challenge will be greater with technology scaling.

Looking at how variation, or mismatch, impacts digital circuits, one finds that variation causes a fair amount of timing issues. Long chains of digital logic gates require data to be processed in parallel in order to reach the correct conclusion. However, mismatch between the logic gates lead the data to propagate unevenly through the chains and thus can result in incorrect processing. Adding to this, variation in interconnects contributes to timing issues due to unmatched propagation delay. Also, with the industry pushing the circuits to operate at ever higher speeds, does not help the challenges.

For analog circuits, process variation tends to be a nuisance of precision rather than time, even though these two are often strongly related. Compared to digital circuits that are operating with two voltage states, analog circuits must operate with all voltage levels within the supply range. Variation affects passive elements, such as resistors, inductors, capacitors and interconnects, by altering their effective impedances which in turn alters the desired signal transfers and voltage levels. Active elements such as transistors have their operation properties change with variation. Analog blocks such as amplifiers and comparators are thus sensitive to variation because they rely on the external and internal elements to operate properly.

With that being said, this thesis will focus on variation and mismatch towards analog designs. The circuit of interest here will be the *analog-to-digital converter* (ADC). ADCs are all designed around comparators which decide whether an analog voltage is greater than another voltage, outputting a digital *HIGH* or *LOW*. Furthermore, differential amplifiers are often used to amplify the difference, when small signal are expected, so that the comparator more easily can decide the outcome. Now, as is stated above, comparators and amplifiers do suffer from variation. Differential amplifiers and comparators rely both on having two symmetrical inner current paths to function properly. These paths usually consist of transistors and resistors. If the paths are not matched, when one element is different from its counterpart on the other branch, it will lead to offsets in the blocks. These offset in turn affect the conversion of the ADCs. For ADCs that use multiple comparators, individual offset in the comparators will cause non-linearities during the quantization which will result in incorrect conversions.

Luckily design methods and layout methods have been developed to deal with the mismatch and variation challenges. Process variation can be divided in two sub-categories, *systematic* variation and *random* variation. Systematic variations refer to predictable variations that can be countered through common layout design methods. Random variations, on the other hand, are unpredictable variations that require additional circuitries or post-manufacturing adjustment to correct the variations.

Design methods for offset correction, caused by random variations, has been proposed for and implemented in sampling techniques utilizing ADC architectures such as *successive approximation*, *delta-sigma*, *pipeline* and *flash* ADCs. In this thesis, however, a less common ADC architecture based on a sapling technique called *swept-threshold sampling* (ST) will be investigated with regards to offset correction. This sampling technique is unique because it allows for direct *ultra-wide-band* (UWB) sampling [2]. A corrected offset for this type of sampler will directly improve sampling time, power consumption and quantization noise. Literature or documentation of offset correction for this particular sampling approach has not been found as of the works of this thesis and will therefore be a challenging project.

The objective of this master project is to evaluate existing offset correction techniques in regard to the swept threshold sampling approach. Resulting correction technique together with the sampler will be implemented in the TSMC 65-nm process and validated through testing. The sampling circuit presented by the paper "*A 118-mW Pulse-Based Radar SoC in 55-nm CMOS for Non-Contact Human Vital Signs Detection*" [3], will serve as a project case. The goal here will be to improve the sampling characteristics and performance of the circuit through offset correction. Resulting implementation will aim to meet the specifications that are true to the once in the case paper.

In this thesis, chapters and sections are referred to with the following hierarchy:

1. Chapter
    1.1. Section
        1.1.1. Subsection
            1.1.1.1. Subsection

The rest of the thesis is set up as follows:

- *Chapter 2* will present the background theory needed for the project. Building blocks that are going to be used will be discussed in addition to design and layout methods that can improve the final product.
- *Chapter 3* will take a closer look at the ST sampling technique and the case circuit. A presentation of offset correction techniques that are being evaluated, will follow.
- *Chapter 4* will investigate the case circuit with regard to offset, and evaluate which correction technique is best suited for the particular sampler.
- *Chapter 5* will present the design approach of every component and the 65-nm implementation of the design. There will also be some simulations that verify the design and that will be references for the hardware tests.
- *Chapter 6* will present the design of the test setup and will discuss the test approach. The end of this chapter will present test results.
- The last two chapters will, as standard, discuss the test results and the project, and draw conclusions for this master thesis.

2 BACKGROUND

2.1 VARIATION AND MISMATCH

When trying to produce multiple physical objects, like cars, chairs, spoons and so on, they will never be fully identical even if the designs and tools used to manufacture then are the same. One car may be more powerful than the next, one chair might curve more than another and a spoon may be shorter or longer than another. Also, a cars performance will change depending on the surface it is driving on. If the surface is icy, the car will steer, break and accelerate worse than it would on dry tarmac. These production imperfections and operating conditions also occur for electrical circuit and are known as manufacturing variations and environmental variation.

Variations in integrated circuit performance are mainly caused by three variation sources. Supply voltage, operating temperature and process variation. The first two are environmental variations while the latter one is introduced during fabrication of the silicon circuits.

2.1.1 Environmental variation

The external supply voltage for a circuit will rarely be fixed to one voltage level. Instead, the voltage supply or regulator will fluctuate around the target voltage level. How much the voltage changes depend on how well the supply or regulator is designed. In addition, IR-drops along the supply rails and di/dt noise add to the voltage variation. These variations effects will in turn influence the performance of a circuit by changing voltage range and current flow. Variation in operating temperature alters the mobility of electrons in the different metal types of which the circuits are composed of. Both these environmental variation sources cannot be controlled by the IC designer. Instead, the designer needs to ensure that the circuits are robust enough to perform within specified conditions to the specified specifications needed. Although these sources are important to manage when designing integrated circuits, this thesis will mainly focus the variations introduced during fabrication.

2.1.2 Process variation

As mentioned above, variation in performance due to non-ideal silicon fabrication processes are known as process variations. Since the production chain of an integrated circuit is long, from raw material to polished top layer, it is useful to differentiate the variations depending on the scale of variation or opposed to what it variates from. The four classifications are *Lot-to-lot* (L2L), *Wafer-to-wafer* (W2W), *Die-to-die* (D2D) and *Within*-die (WID) [4]. A lot is a batch of wafers, meaning that they are produced in the same instance. Lots will differ from each other because the heat treatment and cleaning process for the wafers are difficult to perform the exact same way for every batch. There are also differences between wafers within the same lot, W2W. During ion implementation, wafers may be exposed to the implanter for slightly different durations making the threshold voltages uneven across a lot.

Scaling down a step, one finds that there are variations across a wafer, D2D. This variation topology is also categorised as *inter-die* and *within-wafer* across various books, papers and datasheets. The

main variation here is the inconsistence in channel lengths of transistors due to etching. Etching on one part of the wafer will be different than the etching process on another part of the wafer. In addition, the further apart dies are from each other the greater the probability and magnitude of difference there is [1]. The result of channel length variation is variation in the speed of dies. Transistors with shorter channels can operate at higher frequencies and vice versa.

These *Not-Within-Die* (NWID) process variations (L2L, W2W and D2D) are strongly dependent on the fabrication facilities [5], meaning that it is hard to predict the circumstances that are going to affect a circuit. Instead, designers simulate their designs in the different design corners that illustrate how the circuits are going to behave in extreme NWID circumstances. Design corners are statistical values for the best- and worst-case conditions transistors can experience. The corners are defined as fast-slow, fast-fast, slow-fast and slow- slow where the first part represents how the NMOS transistor will behave and the second part represents the PMOS. *Figure 1* shows a typical design corner diagram with the target-target point in the middle.

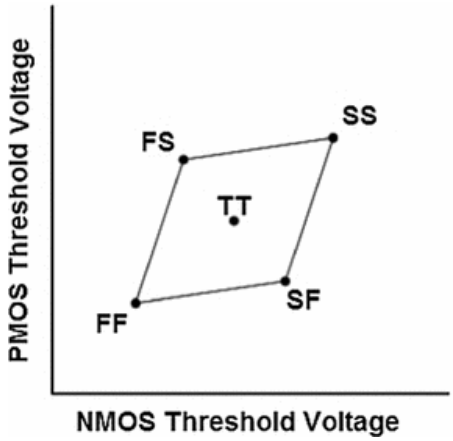


Figure 1: Design corners, provided by [6]

NWID variations need to be considered for all cases but they are more crucial when trying to match mass produced chips. Since the design and purpose of this thesis is to improve low quantity circuits the focus will be on the within-die or *intra-die* variations. This means that there will not be any requirement of matching the operating performance of dies across the wafer.

The following two subsections will introduce and define the two classifications of process variation, *systematic variation* and *random variations* [7] [5].

2.1.2.1 Systematic Variation

Systematic variations can be viewed as *known* variations, meaning that it is possible to predict the variation effects, to a degree, by using simulation tools and statistical device-models of the process technology. The ability to estimate variation gives the designer the possibility to adjust and counteract that variation during the design phase [4] [7] [5].

The definition of systematic variation is variation that systematically occurs during the silicon fabrication process. During lithography processes, *two-dimension effects* occur causing widths and lengths of devices (components) and interconnects (connections between components) to vary systematically across a wafer [7]. In some regions, layers may be etched more than in others, leading unmatched device sizes across regions. These two-dimensional effects are often referred to as *line width variations*. An example of this are regions with high density of polysilicon gate that may be etched narrower compared to regions where the gates are isolated or less dense [4]. The orientation

of a layer can also introduce systematic variations. Horizontal layers may be etched different from vertical layers. These are known as *orientation effects*.

The ion implementation process does also introduce systematic variations. Ion implanters may systematically implement different amounts of ions in different areas of a wafer. Ion implementation do also add to the two-dimension effects. During well formation, lateral diffusion cause wells to be larger than its mask and thus larger than the intended design [7]. The ion implementation process, under well forming, add some additional variations resulting in some *proximity effects*. Dopants tend to concentrate near the well edges due to atoms scattering off the photo resistive masks and down to the well surfaces. This means that the threshold voltage varies across a well. Transistors near the well edges will be slower than transistors on more distant from the edges [4].

2.1.2.2 Random variation

Random variations are all the "unknown" or unpredictable variations that influence a circuit. This unpredictability is closely related to CAD limitations, meaning that although the variation effects are known and documented, they are too hard and-or too costly to model [4]. Considering line edge roughness for instance, it is clearly visible under a microscope. However, recording the edge roughness and making a viable model that can predict it is a difficult task, especially in deep submicron processes.

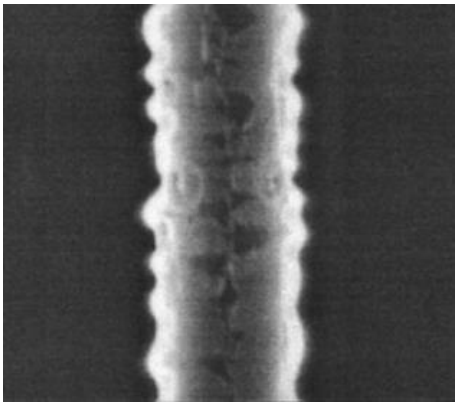
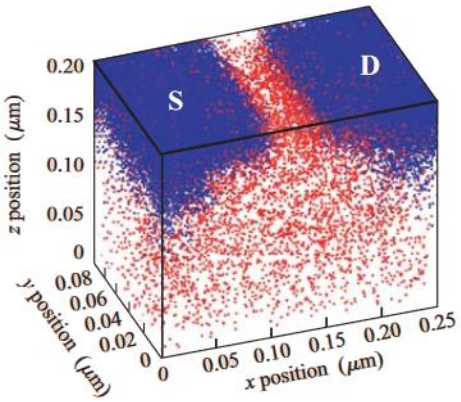
a)  *b)* 

Figure 2: a) Line edge roughness [8], b) Random dopant spread in a 50nm MOSFET channel (red dots are acceptors and blue dots are donors [9]

The two major sources of random variation are again etching and dopant implementation. Etching introduces random line edge roughness in addition to the systematic line width. These two effects cause the effective channel lengths and metal widths to vary from the intended design. For the interconnects, this means that the parasitic resistances and capacitances will be different from the calculated or simulated once. In turn, this may lead to undesired voltage drops at crucial nodes [10]. The second major unpredictable source of variation, dopant implementation, causes random number of dopants to be implemented in random locations of a channel region [4] [7]. This in turn affect the threshold voltage.

A less impactful source of random variation is oxide thickness. Although the thickness of the oxide layers is becoming thinner with transistor scaling, often just a few Å thick, the process that prints the oxide layers is very precise [4]. Because of the good control, this variation may just add about 10% to the standard deviation of the threshold voltage.

2.1.3 Mismatch

The process variations described in the previous sections result in mismatching of devices and interconnect. As will be shown later in the thesis, many designs and architectures rely on devices being equal towards one and another, matched, for the circuits to perform properly.

Mismatch in circuits that rely on symmetry, like differential amplifiers or latches, cause the characteristics of the components to degrade or shift. For differential amplifiers, mismatch in either the paired transistors or the paired loads will lead to offsets between the two current paths [10]. This offset also affects the common-mode rejection feature of the amplifier known as CMRR [11]. As for latches, device mismatch will result in unbalanced parasitic impedances. This in turn affect the decision-making ability of the latch and reduces sense, or resolution [12]. The current mirror is another widely used structure in circuit design that is relies on matching transistors. Its mirroring capability is limited by the voltage threshold variation causing the transconductance, and hence the currents, to be unequal.

Mismatch between interconnects like clock distribution networks or transmission lines do also affect a circuits performance. Variations for interconnects, or between interconnects, stem from width, height and length variations and result in mismatch of parasitic. In the case of transmission lines leading into or out of a differential amplifier, parasitic mismatch may cause one line to have different frequency response characteristics than the other. In clock distribution networks, mismatch will result in *skew* where devices trigger at slightly different times [5]. A flash analog-to-digital-converter is a good example for the importance of clock distribution. If the multiple comparators are not triggered at the same time, then the output value might be corrupted.

Mismatches caused by systematic variations are usually limited during the design process. Design tools allow designers to simulate the potential variations and thus correct them prior to production. Random variations, on the other hand, are difficult or even impossible to simulate. There is however a way of predicting the potential mismatch of devices. In 1989, Mercel J.M. Pelgrom published a paper [1] that showed the matching properties of MOS transistors. He found that the mismatch of two parameters is related to the sizes of the parameters and the distance between them. The findings were expressed with the following equation.

$$\sigma^2(\Delta P) = \frac{A_P^2}{WL} + S_P^2\, D^2$$

(2-1)

Here ΔP is the variance between two parameters P. A_P and S_P are process-determent constants that are determined by measurements of the respected process. And, D is the distance between the parameters P and WL are the width and length of the parameter. The first term of the equation represents the localized variation of the parameter while the second term determines the global variation [5]. However, the second term is often neglected since modern transistors are so small and thus can be minimally spaced. According to [1], spacing can be ignored for transistors with area less than $100\mu m^2$. Based on *equation (2-1)*, one can see that variation is inversely proportional to $\sqrt{WL}$.

2.2 BUILDING BLOCKS

In this section, the main components and circuit block needed for this master project will be presented. There will also be discussions on trade-offs and methods of improving these blocks and components.

2.2.1 Differential Amplifiers

Since the introduction of integrated circuits, differential transistor pairs have become the most used buildings blocks in analog integrated circuit design [10]. The differential amplifier configuration is a well-known input architecture used in almost all amplifiers as well as widely being used as a stand-alone amplifier itself.

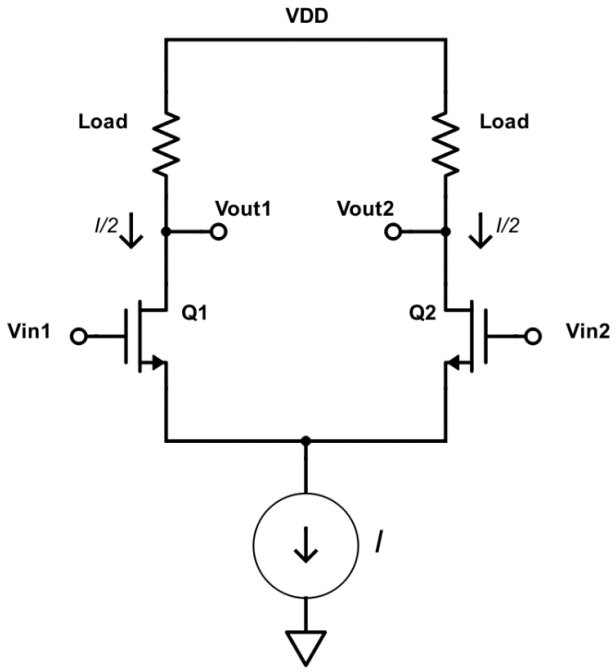


Figure 3: NMOS differential amplifier

Figure 3 illustrates the basic setup for the NMOS differential amplifier. The source-terminals of the paired transistors are connected to the same current source and the drain-terminals are connected to their own loads that are in turn terminated to a supply voltage. If the transistors and loads are matched and both have the same input-gate voltage, then the current going through the current source is equally divided through both transistors. When the voltage is equal on the two input gates it is called common-mode voltage. To ensure that the transistors divide the current equally and that the voltage at the drains are maintained, it is important that the common-mode voltage is chosen such that the transistors are in the saturation region. The following equations determine the maximum and minimum common-mode voltages for the NMOS version in order to keep the transistors in saturation.

$$V_{CM_{MAX}} = V_{th} + V_{DD} - \frac{I}{2}R_D \qquad (2\text{-}2)$$

$$V_{CM_{MIN}} = -V_{SS} + V_{CS} + V_{th} + V_{OV}$$

(2-3)

Equations (2-2) and *(2-3)* express the outer points of the common mode range. V_{th} is the threshold voltage of the transistors, V_{DD} is the positive voltage supply, V_{SS} is the negative voltage supply, I is the total current being pulled by the current supply, V_{CS} is the voltage needed for the current supply to operate properly, V_{OV} the overdrive voltage, and R_D is the resistance of the drain resistors.

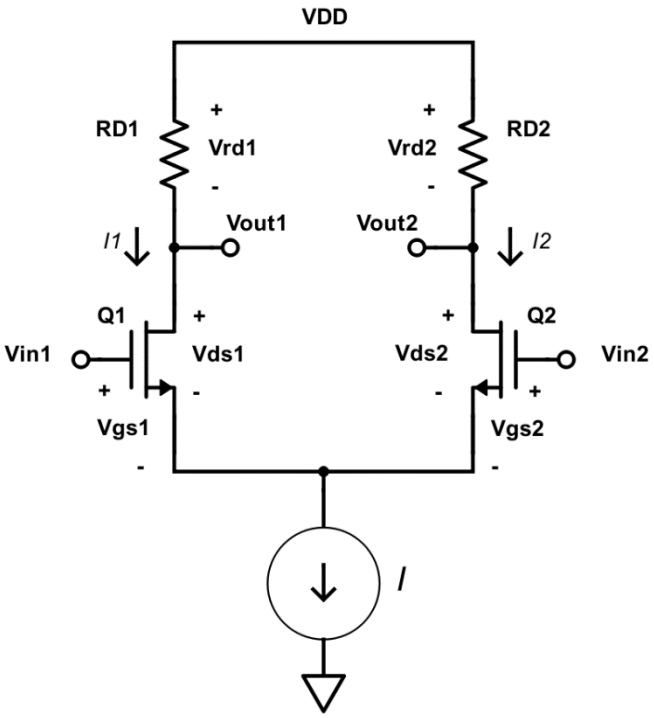


Figure 4: DC analysis of a differential amplifier

Now, considering a differential input voltage on the inputs of the amplifier. Setting one of the input voltages higher than the other affects the current balance in the two current paths. Referring to *Figure 4,* when V_{in1} is higher than V_{in2} transistor Q_1 allows more current to pass through and thus restricting the current in Q_2. This is because the total current needs to equal the current being pulled by the current source. The change in currents in the two paths result in change of the output node voltages, V_{out1} and V_{out2}. With an increased gate-to-source voltage on Q_1, V_{gs1}, the voltage across the load resistance R_{D1} increases while the drain-to-source voltage V_{ds1} across the transistor decreases. This results in the voltage of V_{out1} dropping. The opposite is true for the opposite current branch where the decrease in current decreases the voltage across R_{D2} and thus increases the voltage across the transistor Q_2. With these mechanisms established, one can see that the output voltages change with opposite polarity meaning that having V_{gs1} greater than V_{gs2} results in V_{out2} being greater than V_{out1}.

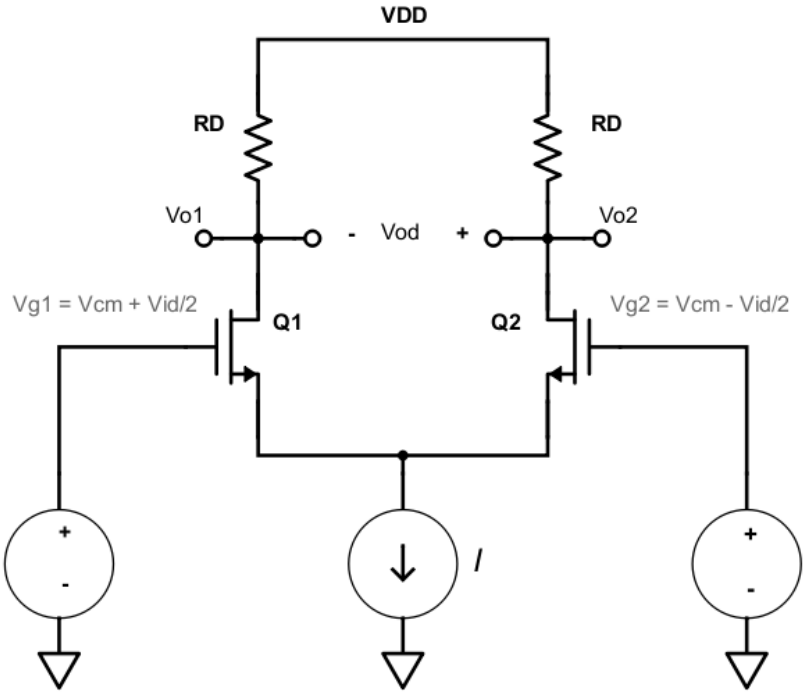


Figure 5: Differential amplifier with voltage and signal sources on the input

In *Figure 5*, the DC biasing points of the input differential pairs are set to be within the common-mode range, in order to keep the transistors in saturation. The differential input signal V_{id} between the two inputs is biased around the common mode voltage V_{CM}, meaning that half the V_{id} is added to the V_{CM} on one input while half the vid is deducted from the V_{CM} om the other. Also, since the current source is still considered ideal, the connection between the two source-gates will act as a virtual ground.

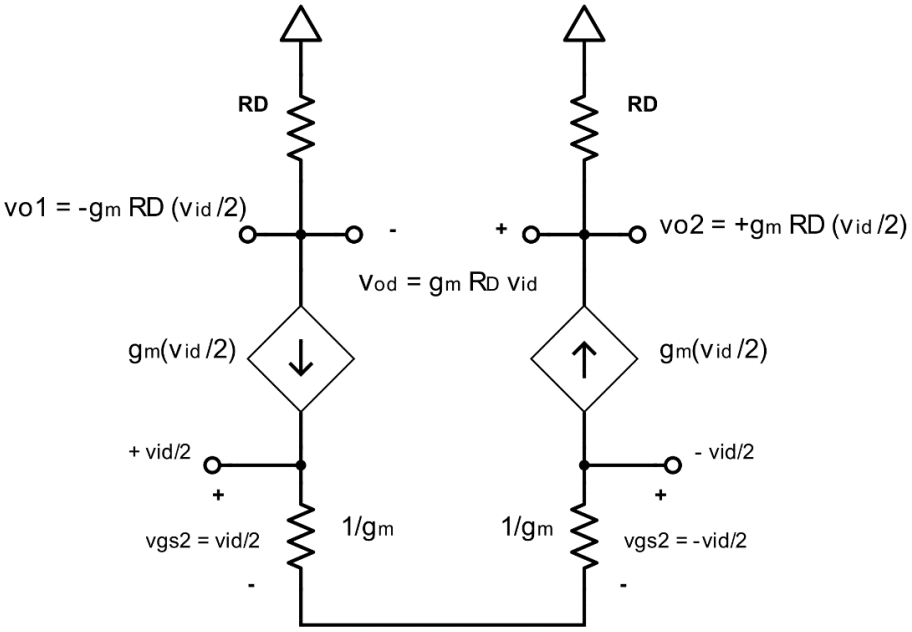


Figure 6: Small singnal analysis of the differential amplifier

By transforming the amplifier to a small-signal circuit with T-models replacing the transistors one ends up with circuit in *Figure 6*. The voltage on the output nodes can be found as follow.

$$v_{o1,2} = -g_m\left(\frac{v_{id}}{2}\right)R_D$$

(2-4)

Note that the DC voltages are shorted to ground in small signal analysis so that the gate-to-source voltages v_{gs} are $\frac{v_{id}}{2}$ complimentary around ground which again means that the polarity is inverted between the inputs. Inputting the polarity of the input signals to the equation leads to the same result as discussed earlier in this subsection where the output node of the branch changes opposite to the input signal. To find the differential gain A_d, v_{o1} can be subtracted from v_{o2} and the divided by the differential input signal v_{id},

$$A_d = \frac{v_{od}}{v_{id}} = \frac{v_{o2} - v_{o1}}{v_{id}} = g_m R_D$$

(2-5)

The two gain equations *(2-4)* and *(2-5)* are however simplified by neglecting the transistor output resistances, r_O. r_O tends to be much larger than the passive resistance R_D and thus is often removed from the equation. A more accurate representation of the gain would be,

$$A_{1,2} = \frac{v_{O1,2}}{v_{id}} = -\frac{1}{2}g_m(R_D||r_O),$$

(2-6)

and thus

$$A_d = g_m(R_D||r_O)$$

(2-7)

Overall, the equations indicate that in order to achieve higher gain, either the transconductance or the output load resistance needs to be greater. To understand how these variables can be modified, some additional transistor equations are presented in the next subsection.

2.2.1.1 MOSFET transistor theory

From MOSFET transistor theory [10], the following equations for the NMOS transistor parameters are established in order to show a illustrate all variables impacting the transistor.

The drain current through a transistor in saturation is,

$$i_D = \frac{1}{2}\mu C_{ox_n}\left(\frac{W}{L}\right)\left(v_{GS} - V_{th_n}\right)^2$$

(2-8)

Another common representation of this equation is given by the fact that $v_{GS} - V_{th_n} = V_{OV}$, the overdrive voltage,

$$i_D = \frac{1}{2}\mu C_{ox_n}\left(\frac{W}{L}\right)v_{OV}^2$$

(2-9)

The transconductance g_m of a transistor can be represented in several ways,

$$g_m = \mu C_{ox_n}\left(\frac{W}{L}\right)(v_{GS} - V_{th_n}) = \mu C_{ox_n}\left(\frac{W}{L}\right)V_{OV}$$

(2-10)

$$g_m = \sqrt{2\,\mu C_{ox_n}\left(\frac{W}{L}\right)I_D}$$

(2-11)

$$g_m = \frac{2I_D}{V_{GS} - V_{th_n}} = \frac{2I_D}{V_{OV}}$$

(2-12)

In addition to this, it is useful to know the mathematics behind the threshold voltage of a transistor,

$$V_{th} = V_{t0} + \gamma(\sqrt{2\phi_f + V_{SB}} - \sqrt{2\phi_f})$$

(2-13)

V_{t0} is the threshold voltage when the source-to-bulk voltage of the transistor is 0V. ϕ_f is a physical parameter, while γ is a fabrication-process parameter.

2.2.1.2 Improving gain of differential amplifiers

As stated above, the ways of improving the gain of these amplifiers is either through improving the transconductance or the output resistance.

Firstly, to improve the transconductance it is useful to look at the equations in the previous subsection. According to *equation (2-10)*, extending the width-to-length ratio $\frac{W}{L}$ of the transistor or increasing the overdrive voltage V_{OV} through raising the gate-to-source voltage v_{GS}, will result in better transconductance. *Equation (2-11)* also indicates that the $\frac{W}{L}$ ratio gives better g_m, in addition to pointing out that increase in drain current I_D helps improve the transconductance. *Equation (2-12)* seams to contradict the statement where the increase in overdrive voltage improves g_m at the first glance, however, *equation (2-9)* shows that I_D also is affected by the V_{OV}. Since the V_{OV} has a greater impact on the current, to the power of two, this validates the previous statement that increase in overdrive voltage results in better transconductance.

There is however a trade-off when increasing $\frac{W}{L}$ and V_{OV}. The output resistance, r_O, of the transistor is also affected by these changes [10].

$$r_O = \left[\lambda\frac{1}{2}\mu C_{ox_n}\frac{W}{L}(V_{GS} - V_{th_n})^2\right]^{-1}$$

(2-14)

The λ represents a device parameter that is dependent on the process technology and the channel length, and it is given in units of reciprocal volts (V^{-1}). Although λ is vital in short-channel -

Continuing with the effects on r_O, by increasing width-to-length ration and the overdrive voltage the output resistance of the transistor decreases. Recalling *equations (2-6)* and *(2-7)*, the gain of the

amplifier depends on the parallel resistance on the output. So, when r_O moved towards R_D the total resistance on the output is reduced and thus the gain is reduced.

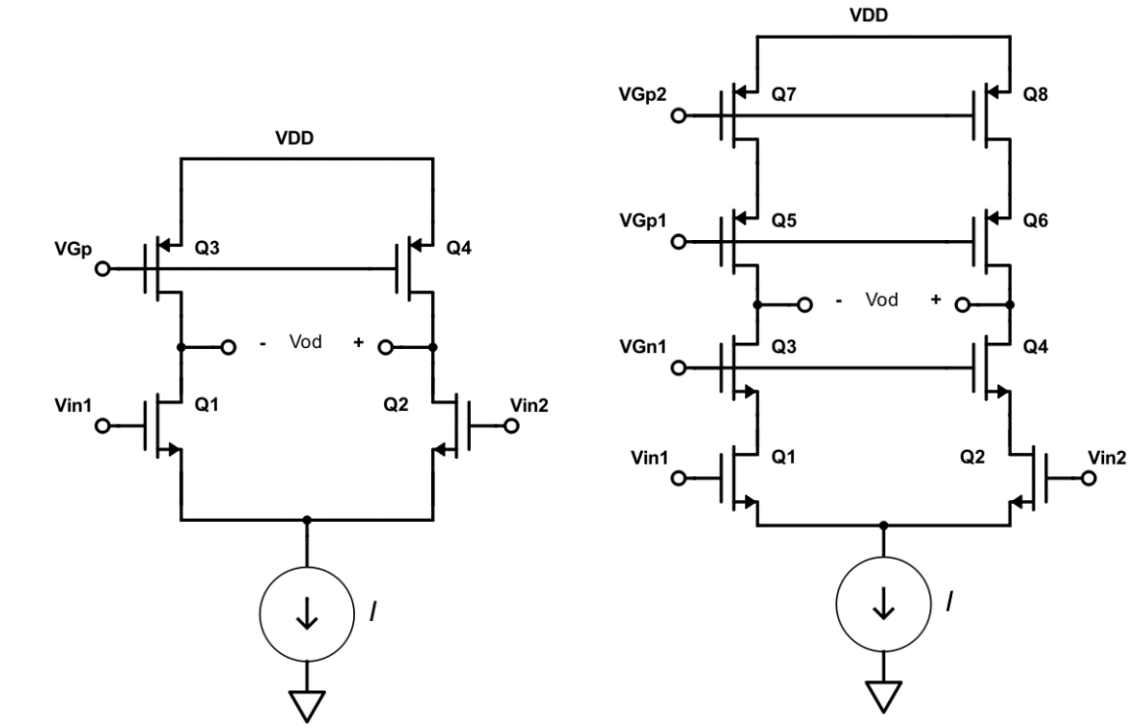


Figure 7: a) Differential amplifier with current mirror load, b) Cascoded differential amplifier

Moving on to improving the gain by changing the output resistance. There are two methods of increasing the output resistance, other than increasing the size of the passive load resistor. These methods build on replacing the passive resistor with active loads. The first one uses PMOS current mirrors instead of the passive resistors [10]. As with NMOS transistors, the output resistance of the PMOS is generally much larger than the passive resistance. The result of this is that the total parallel output resistance becomes larger. *Figure 7a* illustrates how det circuit is set up. The DC biasing voltage V_{G_p} needs to be such that the PMOS transistors deliver a current equal to I_D. The differential gain, with the current mirror loads, can be expressed as follow,

$$A_d = g_{m1}(r_{o_n}||r_{o_p})$$

(2-15)

To further improve the resistance a method called *cascoding* can be used [7], *Figure 7b*. Here the drain load of the input differential pair is connected to multiple transistors and the output signal is moved up to the node where the centre drains of the PMOS and NMOS transistors are connected. Since the transistors are stacked, they represent a larger total output resistance giving by this gain expression,

$$A_d = g_{m1}[(g_{m3}r_{o3})r_{o1}||(g_{m5}r_{o5})r_{o7}] = g_{m1}(R_{on}||R_{op})$$

(2-16)

A requirement for using active loads is to have a large enough voltage supply range to keep all the transistors in saturation. This requirement is difficult to meet with the advances in technology scaling. In addition to this, adding more transistors will increase the complexity to the circuit since

every transistor level needs its own biasing voltage, and the more transistors there are the more area is needed.

2.2.1.3 Advantages of differential amplifiers

So, what are the advantages of using differential amplifiers opposed to single-path amplifiers such as *common-source* (CS) *and common-emitter* (CE) amplifiers?

One of the reasons why differential amplifiers are preferred over single pathed *amplifiers*, amplifiers that have single inputs and single outputs, is this common mode stability. Noise and interference on a single input amplifier will be amplified with the input signal. Differential amplifier, on the other hand, have two inputs (ideally) placed close to each other. This means that the noises and interferences would affect both inputs equally and thus can be viewed as a change to the common mode voltage. The voltage difference between the two drains will not be affected, meaning that noise is not amplified but rather rejected. Of course, in the non-ideal reality there will be some noise on the differential outputs due to mismatch and the fact that the current source has a finite output resistance [10].

Another advantage of using differential amplifiers in IC design is the elimination of bypass and coupling capacitors. Discrete-circuit amplifiers use coupling capacitors to bias the inputs of the amplifiers and to couple amplifier stages together. In addition, bypass capacitors are used at the source terminals of the amplifying transistors so that the signal current go directly to source supply without affecting the source resistance. The differential amplifiers on the other hand do not need these capacitors to couple multiple stages or bypass the source load. This is because the differential output signals are biased at the output common-mode voltage, and because of the virtual ground at the source-terminals of the transistors [10]. The reason for avoiding large capacitors in integrated circuits is that they need large areas for implementation and thus are very costly.

2.2.1.4 DC offset

A critical requirement for differential amplifiers is to have two identical current paths leading into the current source. But as established in *section 2.1*, mismatch, due systematic and random variation in the production process, will always occur. To measure the offset of a differential amplifier one must simply connect the two inputs to the same voltage level. If the output voltage difference is $0V$, then, the two current paths, consisting of transistors and loads, are matched and therefor there is no DC offset. However, if there is a voltage difference between the outputs, then it is called an output offset voltage [10]. The corresponding input offset voltage can be found by dividing the output offset with the differential gain.

$$V_{OS_{in}} = \frac{V_{OS_{out}}}{A_d} \tag{2-17}$$

If multiple stages are cascaded and they all have their own internal offsets, then the by input referred offset of a stage is its own offset in addition to the next stages input offset divided by the first stage gain.

$$V_{OS_{tot}} = V_{OS_{in1}} + \frac{V_{OS_{in2}}}{A_{d1}} = \frac{V_{OS_{out1}}}{A_{d1}} + \frac{\frac{V_{OS_{out2}}}{A_{d2}}}{A_{d1}} \tag{2-18}$$

This equation is common representation for other input referred parameters as well, such as noise. It is possible to add as many stages as needed. What this equation is establishing is that offset of the first stage will always affect the total offset more than the other stages down the line. Taking a fourth stage into account, its effect on the total offset will be divided by the gain of the three prior stages. Another factor to take away from this equation is that higher gain in the initial stages helps decreases the offsets contribution of the later stages.

2.2.2 Comparators

Comparators are among the most used components in electronics [7]. The purpose of the comparator is to, as the name indicates, compare two voltage levels and then output the high or low voltage depending on the input polarity [12]. Comparators are essential components of analog-to-digital converters as well as featuring in many other circuits.

2.2.2.1 Establishing basic principals

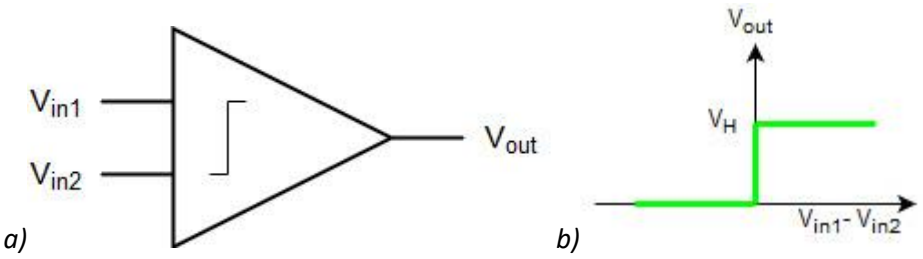


Figure 8: a) Comparator block symbol, b) Ideal characteristics of a comparator

Figure 8 show the basic comparator representation and the ideal input and output characteristic. When input V_{i1} is larger than V_{i2}, the output V_{out} goes *high* and vice versa. To achieve this instant output transition when the input polarity changes, the gain needs to be infinitely high. As is commonly known, infinite gain is difficult to achieve, especially in CMOS design where the transconductance is inherently low compared to bipolar transistors [12]. So, high gain is desired in order to reduce the transition time.

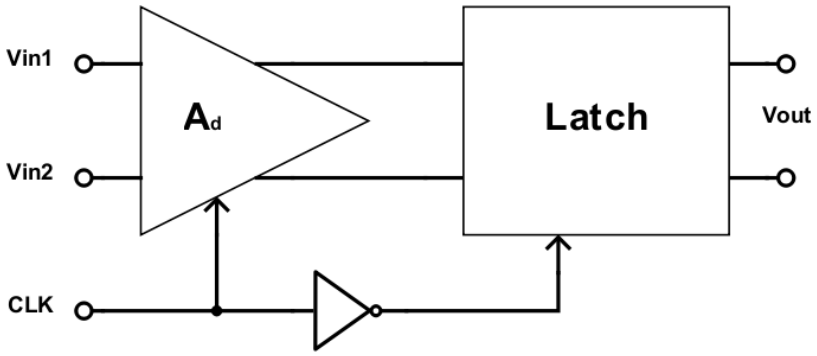


Figure 9: Typical comparator architecture

Figure 9 is a common comparator architecture where a latch is used to decide the output level while a preamplifier is used to amplify the input difference and to store the differential polarity. These two

components are controlled by the clock for them to operate separately [12]. When the clock is high, the preamplifiers *tracks* the input signal like a sample-and-hold circuit. Then when the clock goes low, the latch *latches* its outputs based on the inputs received from the preamplifier. Having the components controlled by an external signal gives the possibility to sample the signal at desired instances.

Because the components are being switched on and off, a noise appears in the inputs called *kickback* [12]. This noise arises when the preamplifier switches on or go into track-mode. In the off state the input gates will not pull any current though because the current has no way to go. Now, when the amplifier switches on, there will be a sudden current pull creating a current spike on the inputs. This current spike will in turn influence the circuit supplying the input signal. There is also charge being output from the input gates when the inputs are turned off for the latch to enter latch-mode. This charge comes from the parasitic capacitances at the gates. A common way of dealing with this noise, as will be discussed later, is to add an amplifier at the input to absorb this noise.

Since a differential pair by itself will struggle to achieve enough gain for the input signal to rail to rail output voltages, latches are used. Latches, in the form of cross coupled inverters or inverting amplifiers, create positive feedbacks in order to generate virtually infinite gain [12]. However, latches are prone to something called *hysteresis*. Hysteresis can be viewed as memory of the previous state or decision [7]. In order to *flip* a latch, one usually needs to apply a higher differential voltage with the opposite polarity of the previous state. This is because charges are built on the parasitic capacitances on the side previously outputting the high voltage. So, a small differential voltage might not be strong enough to flip the latch. A technique to alleviate this memory is to pre charge both sides of the latch to balance them out before activating them. This is regularly done with clocked switches that force the voltage on the output nodes to an equal level during the tracking mode. Then, when the latching starts, the outputs are “pulled” with equal “weight”.

Another point to consider with comparators is *metastability*. Metastability occurs when comparator does not have enough time to generate the desired logic output during the latch-mode [12] [7]. The following equations describe the relationships between time, gain and voltage difference.

$$\Delta V_{XY} = V_X - V_Y = \Delta V_0 \, e^{(A_O - 1)\frac{t}{\tau_0}}$$

(2-19)

Here A_o is the small-signal gain of the inverters, τ_0 is the characteristic time constant of the inverters, and ΔV_0 is the initial voltage difference between V_X and V_Y at $t = 0$. This equation indicates that the voltage difference ΔV_{XY} increases exponentially with time, t. It also states that initial voltage difference V_{XY0} and inverter gain A_o extend ΔV_{XY}. The next equation expresses how much time, T_{lch}, the latch needs to achieve a desired ΔV_{XY}, in this case a logic level ΔV_{LOGIC}.

$$T_{lch} = \frac{\tau_0}{A_0 - 1} \ln\left(\frac{\Delta V_{LOGIC}}{\Delta V_0}\right)$$

(2-20)

The equation shows that the time T_{lch} needed to generate ΔV_{LOGIC} increases with decrease in ΔV_0, but is improved with better A_o or lower τ_0.

So, the comparator becomes metastable if T_{lch} is larger than half the sampling clock period ($\frac{1}{2}T_{clk}$). The result of metastability is potential logic error from the comparator.

Another effect of the latch time response is the limitation of resolution [12]. For a set sampling frequency, the maximum resolution of ΔV_0 (neglecting noise and offset) is determined by the minimum regeneration time of the latch in order to output a correct logic level. In other words, the

maximum resolution is the minimum initial voltage difference needed for the latch not to become meta stable.

2.2.2.2 Offset and noise

Adding to the previous section of the latch time response determining the maximum resolution, offset and noise do also affect the resolution. The minimum input voltage difference must be raised so such that the output logic level is not affected by noise.

As with the differential amplifier, mismatch leads to voltage offsets in the components. In the case of comparators constructed by differential amplifiers and latches, the offset is referred to the input of the comparator. The input offset is determined by the differential input voltage at which the output logic level changes [7]. Ideally the logic levels would change when both inputs voltages are identical, the differential voltage being zero, but that is rarely the case. Instead the crossing point output changes is shifted.

It is possible reduce the offset impact of the comparator by adding gain stages prior to the comparators. This is why comparators often are preamplified with a number of gain stages [7]. The only downside of having many preamplifier stages is that the signal needs to propagate through all of them before reaching the comparator. This in turn slows the signal down.

Random noise on the inputs may cause the output logic to flip even if the input voltage is constant [7]. The amplitude of the noise does affect how lightly it is for the logic state to change considering the distance from the differential voltage to the threshold at which the comparator flips without the noise factor.

A way of measuring the input offset with noise is to sweep the inputs of a comparator and oversampling the output for each step of the sweep. The result will end up looking something like the chart in *Figure 10*. This chart is commonly known as a *cumulative distribution function* (CDF).

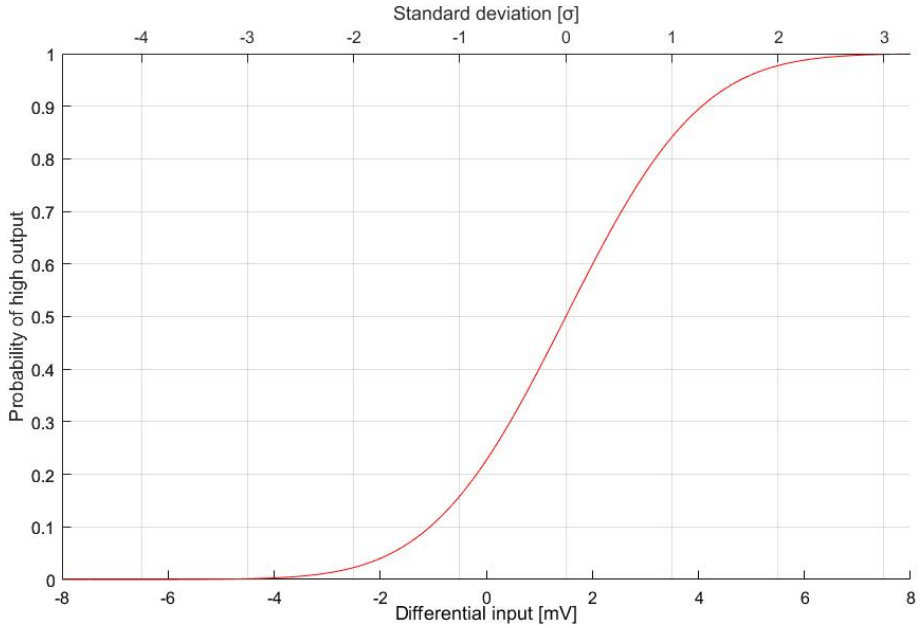


Figure 10

2.2.2.3 StrongARM latch

The StrongARM latch is widely used as comparators [13]. The reason for this is that the latch does not consume any power when static, it produces rail-to-rail outputs and the input referred offset voltage stems primarily from one transistor pair.

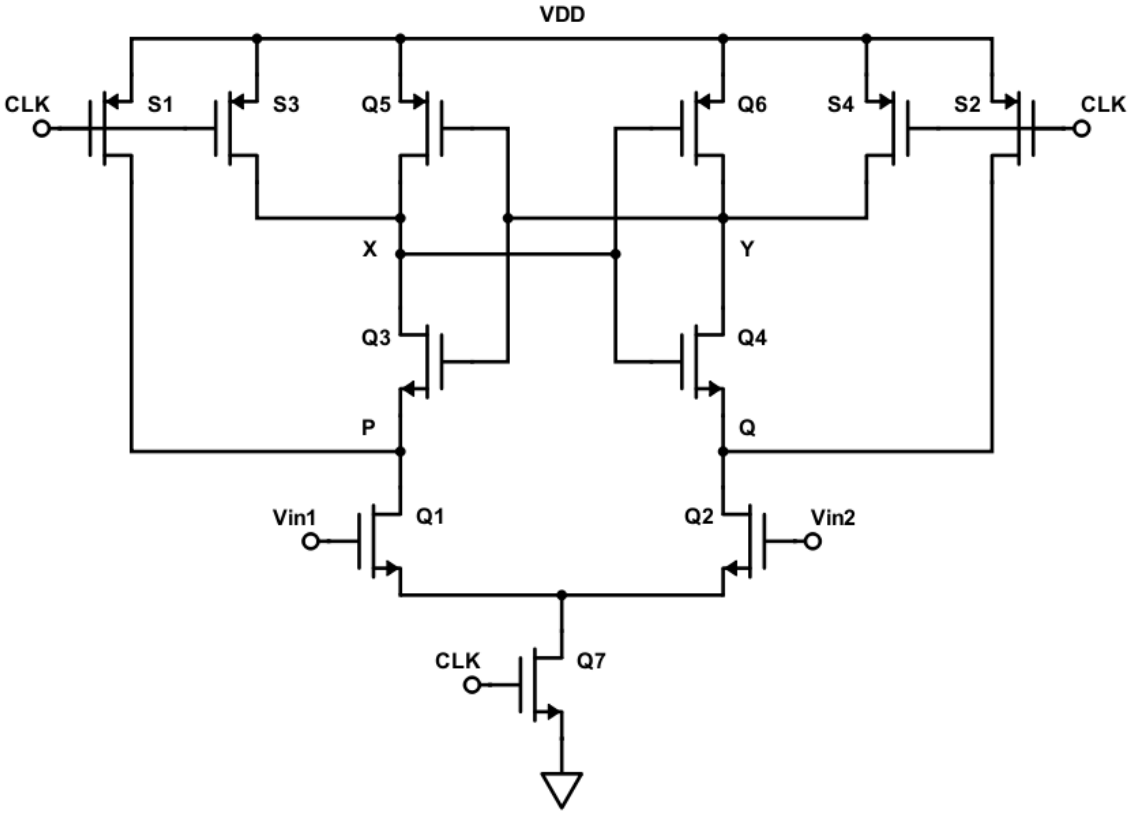


Figure 11: The StrongARM latch

Figure 11 illustrates how the latch is designed. Q1 and Q2 are a differential input pair that amplify the differential input signal of V_{in1} and V_{in2}. Transistors Q3 through Q6 are two cross coupled transistor pairs creating positive feedbacks in order to amplify the difference further so that det outputs reach logic levels. Q7 is the current source for the input pair, also acting as a switch in order to turn the latch off during the tracking mode. Finally, S1 through S4 are switches that precharging nodes X, Y, P and Q to V_{DD} when the sample clock signal (CLK) is low. The differential output is sensed between the two nodes, X and Y.

The StrongARM latch goes through four phases [13]:

1. The CLK is low, turning off the NMOS current source and turning on the PMOS switches. This allows the parasitic capacitances of node X, Y, P and Q to charge to V_{DD} because the current is blocked from passing through the input pair. By charging these nodes the previous state is erased and thus the latch is not affected by hysteresis.
2. The CLK goes high, turning on the current source and off the switches. Since node P and Q are charged to V_{DD}, Q1 and Q2 start drawing differential current and potentially amplifying the input differential signal. At this point are still off because their gate-to-source voltages are below the threshold voltage.
3. When the voltages at node P and Q fall to $V_{DD} - V_{th_n}$, then Q3 and Q4 turn on and allow Q1 and Q2 to pull current from nodes X and Y. This in turn allows the cross coupled NMOS transistors to some initial low regeneration.
4. The last phase activates when nodes X and Y fall to $V_{DD} - V_{th_p}$. At this point the cross coupled PMOS transistors get enough gate-to-source voltage for them to turn on and start the full regeneration. This results in nodes X and Y reaching the opposite rail voltages determined by the input differential signal.

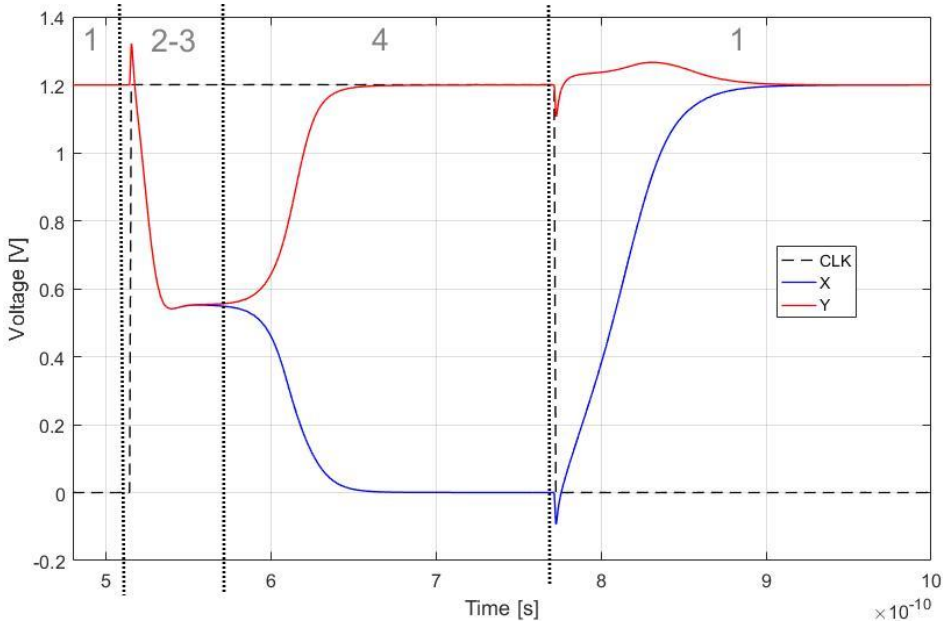


Figure 12: Transient analysis of the output voltage of a StrongARM comparator

Figure 12 illustrates the voltages of the output nodes X and Y, from the schematic of *Figure 11*, for the four phases.

For this particular latch design, the input referred offset voltage is dominated by the mismatch of the input transistor pair [13]. This is because the two cross coupled transistor pairs are initially turned off until phase three activates the NMOS pair, and eventually phase four turns on the PMOS pair. During the second phase, the amplification phase, the input pairs discharge nodes P and Q while also implementing their internal offsets in the nodes prior to the following phases. The same happens with the NMOS cross coupled pair in the first regeneration phase, allowing this offset to affect nodes X and Y before phase four kicks in. This does also comply with the *equation (2-18)* where the offset of later stages is divided by the gain of the previous stages.

According to paper [13], the input referred noise is also mostly derived from the input differential pair. The reason is the same, where noise from Q3 through Q6 is reduced because the switches precharge the four selected nodes resulting in the cross coupled transistors to turn off. However, kt/C-noise from the switches S1 and S2 does affect nodes P and Q and thus can be viewed as significant.

2.2.2.4 Improved StrongARM latch

Paper [14] introduces an improved version of the StrongARM latch. According to the paper, the proposed latch design increases the latching speed, reduces the energy consumption and reduced clock feedthrough. *Table 1*, summarizes the improvements from the proposed design compared to the design introduced by [13].

	32nm	*90nm*
Speed improvement at $V_{diff} = 1mV$	9%	14%
Speed improvement at $V_{diff} = 100mV$	7%	8%
Energy improvement at $V_{diff} = 1mV$	15%	7%
Energy improvement at $V_{diff} = 100mV$	10%	3%
Average reduction in clock feedthrough	56%	41%

Table 1: Improvements of the improved StrongARM latch compared to the regular StrongARM latch

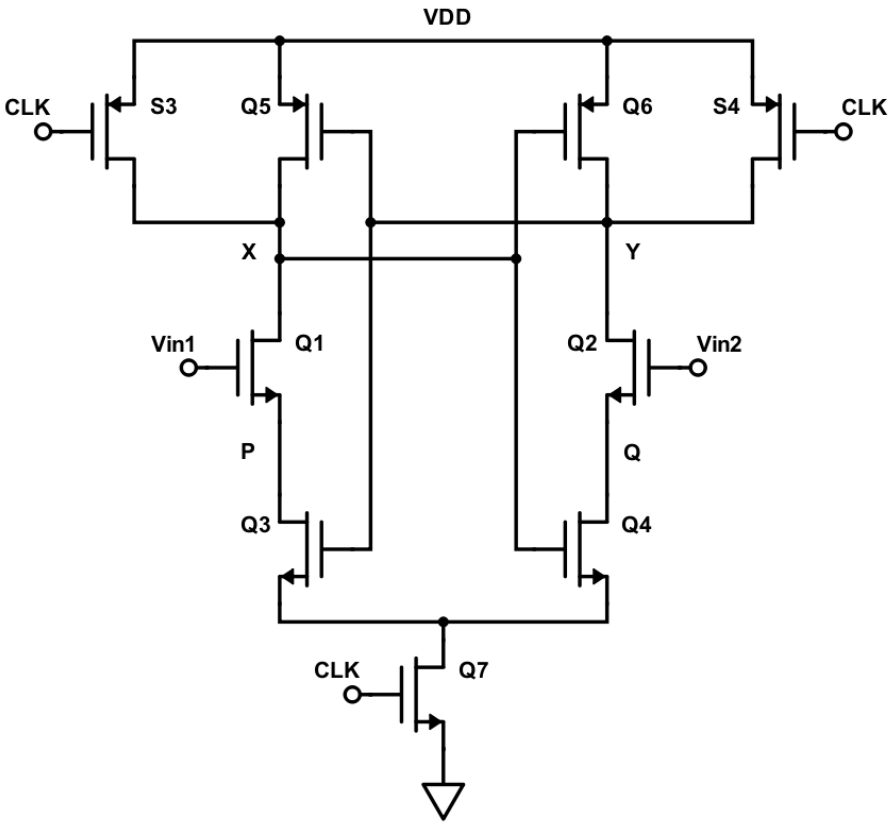


Figure 13: The improved StrongARM latch

Figure 13 illustrates the proposed latch. This design focuses on improving the latching time by reducing the capacitance in the previously mentioned nodes X, Y, P and Q. To achieve this, the input differential pair has been moved above the NMOS cross coupled pair. This way the S1 and S2 can be removed because the Q1 and Q2 are biased with a common mode voltage thus precharging nodes P and Q. The result of removing the switches is less capacitance in the nodes thus less time is needed to discharge them in order to move to the next phase. Another advantage with removing those switches is the removal of clock feedthrough. In the design of [13], S1 and S2 inject charges to nodes P and Q when the transistors were turned off. Although S3 and S4 also disburse charges when they turn off, they do not affect overall latch response due to nodes X and Y not being *active* until P and Q have fallen by V_{thn}.

However, the new stacking order of the latch changes some of the *mechanics*. Since the NMOS cross coupled pair have moved down, they are the first to be turned on. If all the differential pairs were to be perfectly matched, then there would no differential gain during the discharge period of P and Q because X and Y have the same voltage.

The removal of switches S1 and S1 does introduce a trade-off in reset time. Since nodes P and Q no longer are charged directly through S1 and S2, but instead through S3, S4, Q1 and Q2 respectively, this increases the time needed to reset the four nodes to the supply voltage. This, however, does not affect the overall response of the latch since the reset period is much faster in the first place.

2.2.3 Frequency response

Achieving wide frequency ranges as the one needed for the receiving end of the sensor in paper [3], is not an easy task. This subsection will thus introduce the effects that limit the frequency range and present techniques for improving it.

Since integrated circuit amplifiers usually avoid using bypass and coupling capacitors [10], the focus here will only be on the upper frequency range.

2.2.3.1 Internal capacitance of a transistor

As mention previously, transistors have internal capacitances. In the case of the comparators, these internal capacitances added up on the different nodes and determined the discharge rate, or speed, of the comparator. When it comes differential amplifiers, internal capacitances also limit the propagation characteristics. Here parasitic capacitance affects the frequency bandwidth of the amplifier.

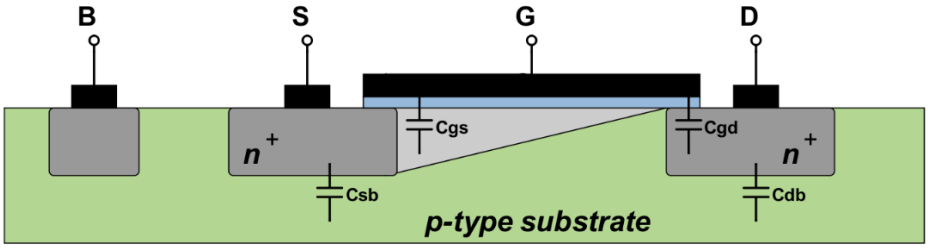


Figure 14: Cross section of a n-channel MOSFET

A cross-section of a n-channel MOSFET transistor, with its four main internal capacitances, is illustrated in the *Figure 14*. Note that the transistor is in saturation, as indicated by the pinch-off of the n-channel, thus the following equations represent the capacitances in that respected state.

To begin with, the capacitance of the overlapping area, C_{ov}, from gate to source or gate to drain is established.

$$C_{ov} = WL_{ov}C_{ox} \tag{2-21}$$

Here W is the width of the transistor, L_{ov} is the length of the overlap by gate and C_{ox} is the regular oxide capacitance per unit gate area. The length of L_{ov} is typically between 0.05 and 0.1 of the full channel length [10]. This overlapping capacitance is important because it partially or fully represents some of the other capacitances like the gate-to-drain capacitance, C_{gd}.

$$C_{gd} = C_{ov} \tag{2-22}$$

As the cross section shows, the only area a charge between gate and drain can accumulate is on the overlapping area. This area is also present for the gate-to-source capacitance C_{gs}.

$$C_{gs} = \frac{2}{3}WLC_{ox} + C_{ov} \tag{2-23}$$

Here the C_{gs} is defined by C_{ov} and the area of the channel where the length is $\frac{2}{3}$ of the full L because of the pinch off when the transistor is in saturation.

The capacitance of the depletion layer for the two reverse-biased pn junctions between the diffusions of source and drain to the *p*-type substrate. are given by these two equations.

$$C_{sb} = \frac{C_{sb0}}{\sqrt{1 + \frac{V_{SB}}{V_0}}}$$

(2-24)

$$C_{db} = \frac{C_{db0}}{\sqrt{1 + \frac{V_{DB}}{V_0}}}$$

(2-25)

C_{sb0} and C_{db0} the values of their respected capacitances when the reverse-bias voltages V_{SB} and V_{DB} are $0V$. Also, V_0 is the junction built-in voltage normally being in the region of $0.6V$ to $0.8V$. The junction capacitance C_{j0}, representing C_{sb0} and C_{db0}, is calculated by all *pn* junction sides of the diffusion.

$$C_{j0} = A\sqrt{(\frac{\epsilon_s\, q}{2})(\frac{N_A N_D}{N_A + N_D})(\frac{1}{V_0})}$$

(2-26)

Equation (2-26) is the formula for finding the exact value of the junction capacitances [10]. This thesis will not go more in depth on this subject. However, the knowledge of junction capacitance being proportional to the surface area of the diffusion, and thus the transistor area, will be useful.

2.2.3.2 High-frequency response of a transistor

With all the internal capacitances introduces, a small-signal model of the MOSFET, including the capacitances, is needed for calculating high-frequency response of amplifiers.

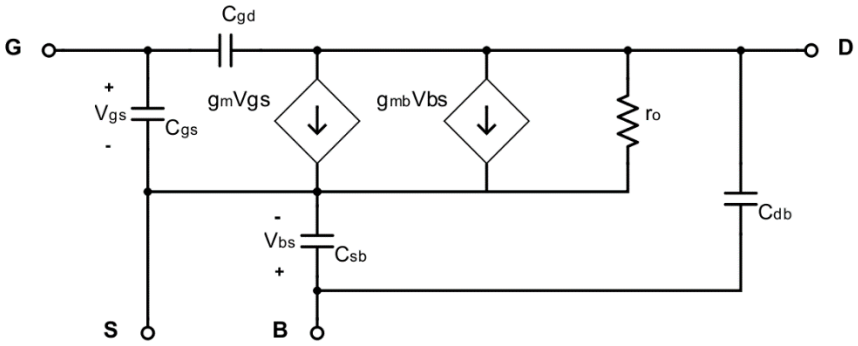


Figure 15: Small-signal model of a MOSFET with parasitic capacitances

The unity-gain frequency f_T of a MOSFET is a way of expressing how effective the transistor is [10]. f_T defines at which frequency the gain of a MOSFET is equal to 1. A simplified formula for f_T, where the body terminal is connected to the source terminal and the C_{db} is neglected, can be seen in *equation (2-27)*, below.

$$f_T = \frac{g_m}{2\pi(C_{gs} + C_{gd})}$$

(2-27)

This equation states that the f_T is proportional to the g_m and inversely proportional to the internal capacitances. As was discussed earlier, the transconductance is proportional to the size of a transistor as is the internal capacitance. This means that unity-gain frequency does not rely on the transistors size.

2.2.3.3 High-frequency response of a differential amplifier

In high-frequency analysis, finding the frequency where the gain falls by $3dB$ is the main goal [10]. Assuming that the amplifier represents a low-pass network, which it commonly is in integrated circuit design due to coupling capacitors rarely being integrated. This $3dB$ frequency, f_H, indicates at which upper frequency the signal will degrade.

Since differential amplifiers can be represented as half-circuits, because source connection of the differential transistor pair acts like a virtual ground, the frequency response will be the same as for common-source amplifiers. The following equations are derived from the common-source amplifier but are also valid for the differential amplifier.

The simplest way of finding f_H is to distinguish the dominant pole of the circuit, if one exists. The rule of thumb is that in order for a pole to be dominant it has to be at least a factor of 4 away for the nearest pole [10]. If this is the case f_H can be found with the following equation.

$$f_H = \frac{1}{2\pi\,\tau_H}$$

(2-28)

Here τ_H is the time constant of the capacitance that creates the dominant pole.

When it is not possible to define a dominant pole an approximate value of f_H can be found by adding up all time constants in the circuit. The new τ_H, that is the sum of multiple time constants, is known as the *effective high-frequency time constant*.

The *Open-Circuit Time Constant* method is a good way on determining all the time constant of each capacitance in a circuit [10]. To demonstrate the outcome of the method, common-source amplifier with input and output loads, as the one in *Figure 16*, will be analysed.

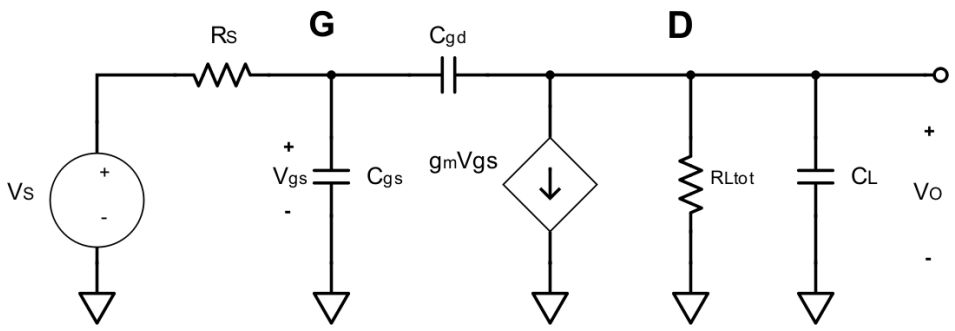


Figure 16: Generalized high-frequency circuit for a CS amplifier

The effective high-frequency time constant here is,

$$\tau_H = C_{gs}R_S + C_{gd}\left[R_S\left(1 + g_m R_{L_{tot}}\right) + R_{L_{tot}}\right] + C_L R_{L_{tot}}$$

(2-29)

This equation clearly represents the time constants for the three capacitances. Another way of presenting the equation is in reference to the input or output of this circuit.

$$\tau_H = [C_{gs} + C_{gd}(1 + g_m R_{L_{tot}})]R_S + (C_{gd} + C_L)R_{L_{tot}}$$

(2-30)

The $(1 + g_m R_{L_{tot}})$ component of these equations stems from the *Miller effect* where the C_{gd} capacitance sensed at the input is multiplied due to the voltage between gate and drain. This voltage is related to the large negative gain of the amplifier and thus multiplying the C_{gd} [10]. The gain is as usually given by the transconductance g_m and the total load resistance ($R_{L_{tot}} = r_o||R_D||R_L$).

There are a couple of things to take away from this subsection:

1. Bandwidth can be traded for gain.
2. The easiest way of finding f_H is to identify the dominant pole if possible.
3. In order achieve the widest bandwidth, the poles of a circuit should be clustered.
4. Resistances and capacitances limit the bandwidth of a circuit.
5. The Miller effect may introduce a dominant pole if the gain is high.

2.2.4 Bandwidth extension techniques

This section will introduce bandwidth extension techniques presented by [15]. These techniques are based on passive filtering and do not increasing power consumption. As an example, circuit for these techniques, the common-source amplifier will once again be used.

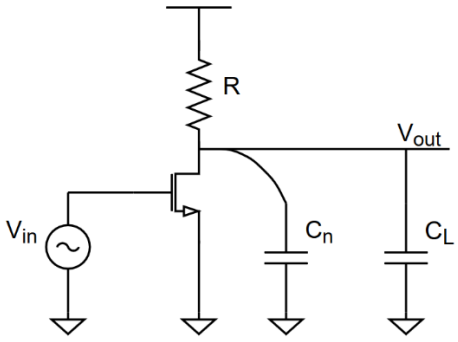


Figure 17: CS amplifier with node and load capacitance identified

For some of these techniques to be considered the node capacitance C_n will need to be distinguished from the load capacitance C_L. C_n represents the parasitic capacitance of the node while C_L represents attached capacitances, such as inputs of cascaded amplifying stages and such. The presented techniques depend on the ratio between C_n and the total output capacitance C which is denoted by the design constraint k_C.

$$k_C = \frac{C_n}{C} = \frac{C_n}{C_n + C_L}$$

(2-31)

The dependency of k_C means that different techniques yield effects for different cases. Also, in cases such as multistage amplifiers, there may be a need for using multiple techniques in order to obtain the best results.

2.2.4.1 Shunt peaking

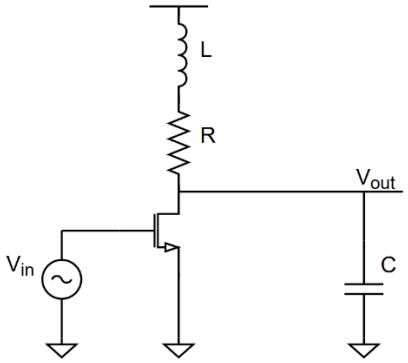


Figure 18: CS amplifier with shunt peaking

The first extension technique is called *Shunt Peaking*. Here an inductor is connected in series with the drain resistance. The reason why this technique extends the bandwidth is that the inductor hinders the small signal to pass though the resistive path and instead goes to the output. Thus, when a larger signal, or current, will charge the capacitance leading to shorter risetime. To understand the inductor value in relationship to the other impedances, variable m is introduced.

$$m = \frac{R^2 C}{L}$$

(2-32)

[15] reports a maximum *bandwidth extension ratio* (*BWER*) of 1.84 for $m = \sqrt{2}$ with a 1.5dB peak.

2.2.4.2 Bridged-shunt peaking

To reduce the peaking of previous technique, *Bridged-Shunt Peaking* can be implemented. This technique adds a capacitor in parallel with the inductor.

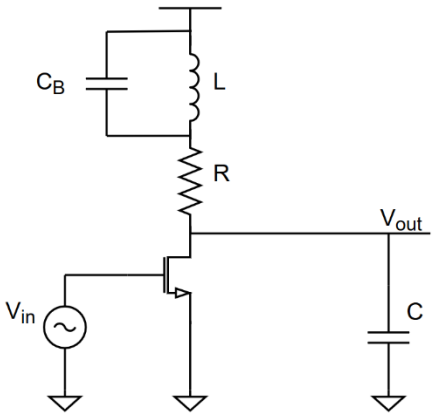


Figure 19: CS amplifier with bridged-shunt peaking

In order to determine the effect of the added capacitor, a relationship k_B to the resistances and capacitances must be established.

$$k_B = \frac{C_B}{C}$$

(2-33)

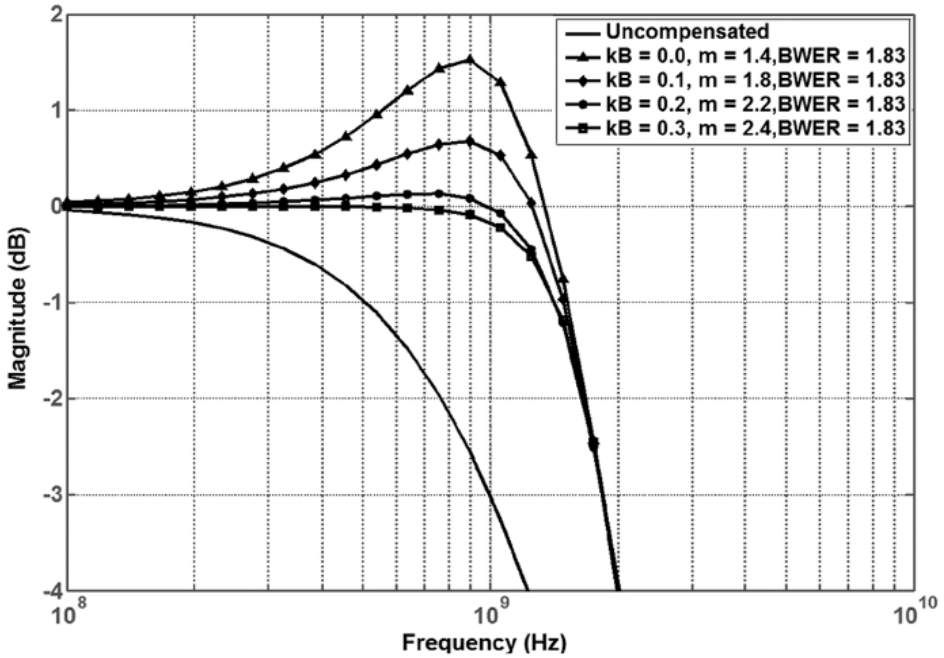


Figure 20: Frequency response of a CS amplifier with bridged-shunt peaking, provided by [15]

The graph in *Figure 20* shows that the peaking can be significantly reduced bridging the inductor with a capacitor, whilst still retaining the BWER. The table in *Figure 20* also shows that it is possible to achieve close to maximum BWER with smaller inductor values. With k_B increasing, m can also be increased, and recalling *equation (2-32)*, increased m translates to decreased L. Inductor in integrated circuits are quite large in size, so finding a way of reducing the area usage is desired.

2.2.4.3 Series peaking

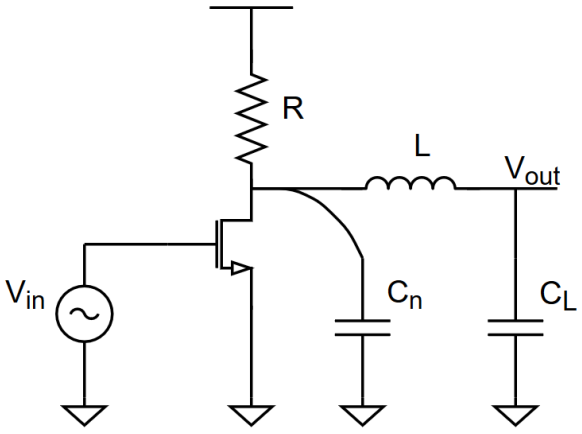


Figure 21: CS amplifier with series peaking

The next technique the [15] presents is *Series Peaking*. For this technique, the aim is to split the total capacitances C. Series peaking is preferred when the parasitic capacitance of the node C_n approaches the total capacitance C. This will happen if either the size of the transistor increasing or the load capacitance decreasing, which also translates to k_C increasing.

$k_C = \frac{C_n}{C}$	Ripple (dB)	$m = \frac{R^2 C}{L}$	BWER
0	0	2	1.41
0.1	0	1.8	1.58
0.2	0	1.8	1.87
0.3	0	2.4	2.52
0.4	1	1.9	2.75
	2	2.5	3.17
0.5	3.3	1.5	2.65

Table 2: Summary of the series peaking technique

Table 2 verifies that series peaking yields better BWER for increased k_C and worse BWER for lower k_C values. A bi effect of this technique is the ripple formation. If the circuit is fine with having ripples, then the BWER of 3.17 can be achieved.

2.2.4.4 Bridged-shunt-series peaking

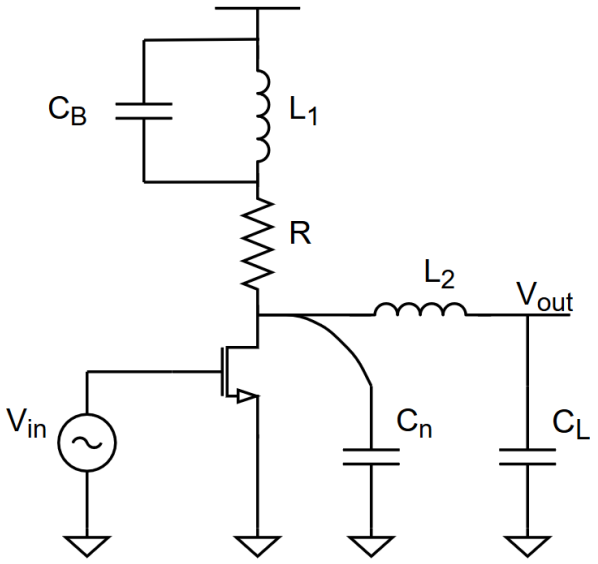


Figure 22: CS amplifier with bridged-shunt-series peaking

To achieve even better BWER, *Bridged Shunt Series Peaking* can be implemented. This technique implements all the once discussed above. Like with the series peaking, this technique is better suited for k_C being large.

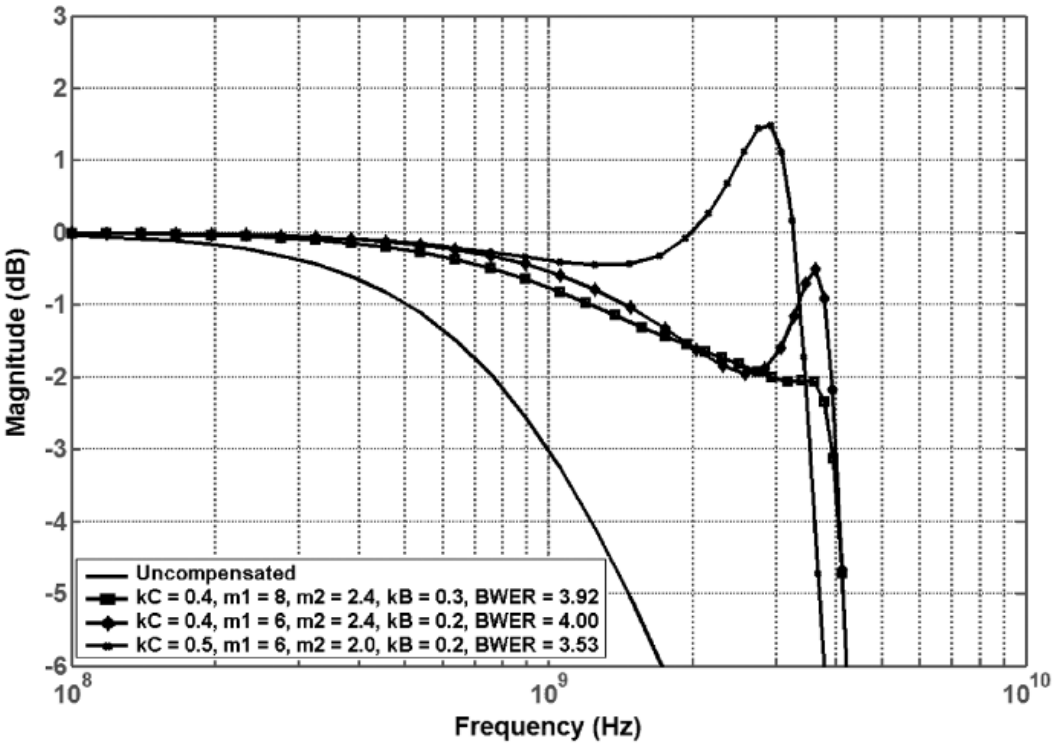


Figure 23: Frequency respons of a CS amplifier with bridged-shunt-series peaking, povided by [15]

Here another relationship is introduced, m_2, with $m_1 = m$.

$$m_2 = \frac{R^2 C}{L_2} \tag{2-34}$$

Figure 23 indicates BWER extending to 4 with less ripple than the series peaking method. The only drawback here is that an additional inductor needs to be implemented leading to even larger area usage.

2.3 Mismatch improvements and Corrections

This section will present design methods and correction techniques that can improve circuit mismatches. These methods and techniques will be directed towards the devices and architectures discussed in the thesis up till this point.

2.3.1 Layout methods for improved matching

There are methods for reducing mismatch during layout design. These methods, if used properly, can eliminate almost all mismatches due systematic variation. For random variation, layout methods have only the ability to improve mismatch to a certain degree.

2.3.1.1 *Unity sizing*

Using unity sized devices to scale device values is a good way of improving matching [5]. This is due to larger devices being less affected by variation than smaller devices. A case for this can be presented by resistor arrays. If a process systematically overetches the widths of a resistive metal by $10nm$, then the variation would have greater impact on a $100nm$ wide resistor than a $200nm$ wide resistor. The narrower resistor would experience a variation of 20% while the wider one would experience 10% variation. If instead two small resistors are connected in parallel, to achieve the same resistance as the large resistor, then the variation would remain 20%. *Figure 24* demonstrates this method where the grey area represents the over etched region of the metal. This logic can also be used for other devices such capacitors and transistors.

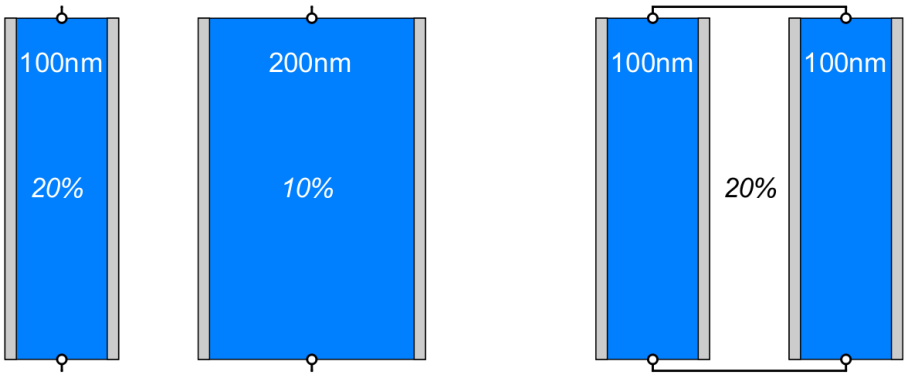


Figure 24: Unity sizing

2.3.1.2 *Dummy transistors*

Adding dummy transistors reduces proximity effects and line-width effects of the poly gates. As mentioned in *section 2.1.2.1*, polysilicon lines in more dense areas are usually etched narrower than in lower density areas. The idea here is to match the poly density of every transistors in order to achieve the same variation for all transistors [5]. *Figure 25* illustrates this mismatch.

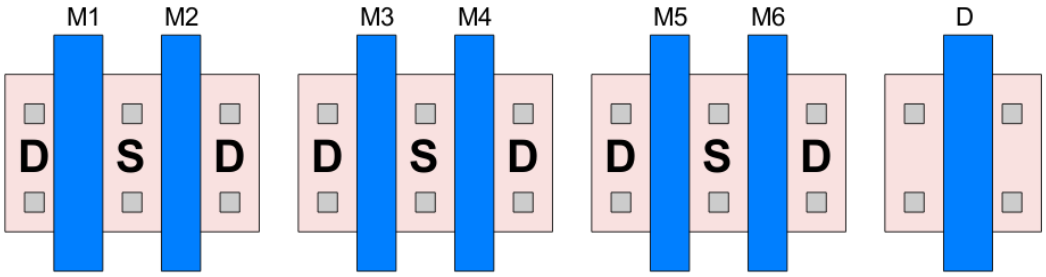


Figure 25: Dummy transistors

The figure shows a row of tree transistor pairs and a dummy transistor. Transistor pair M1-M2, is poorly matched due to M1 experiencing lower polysilicon density and thus is wider than M2. Transistors M3 and M4 experience the same density resulting in pair M3-M4 being matched. The use of dummy transistors, here transistor D, will increase the density for the outer transistor in M5-M6 so that the pair is better matched.

2.3.1.3 *Common centroid*

The common centroid method is useful for large devices where gradients introduce variation [11]. Gradients are results of the imperfect manufacturing processes that may lead to variables such as

oxide capacitance per area, C_{ox}, gradually changing across a distance. For devices along this distance, the variation will be significant when comparing devices from one side of the distance to the devices on the other side.

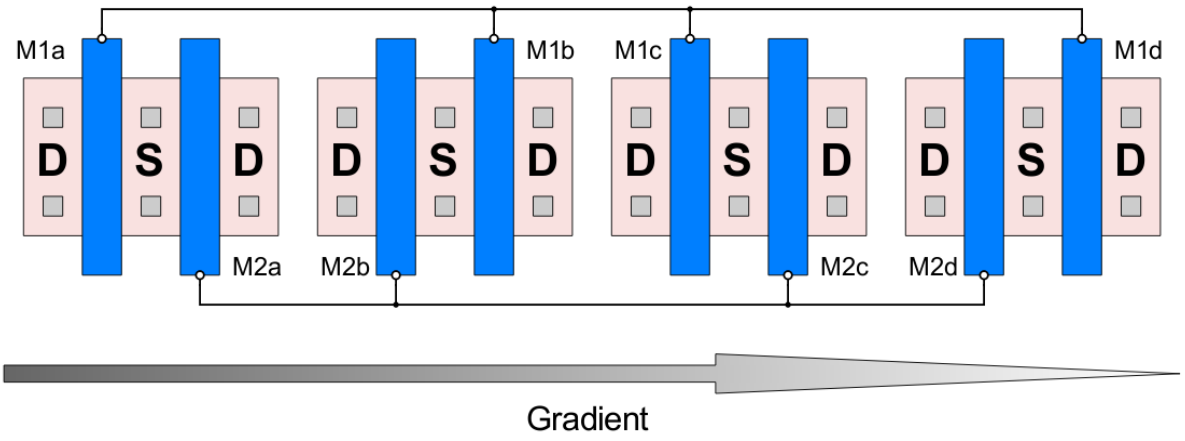


Figure 26: Common centroid

Figure 26 demonstrates how the common centroid layout method works for a large transistor split up by four fingers. If the process is gradient along the horizontal axis and every transistor past the first one adds an additional ΔC_{ox} to C_{ox} for every step, variation averaging can be achieved.

$$\Delta C_{ox}(M1) = 0\Delta C_{ox} + 3\Delta C_{ox} + 4\Delta C_{ox} + 7\Delta C_{ox} = 14\Delta C_{ox} \quad (2\text{-}35)$$

$$\Delta C_{ox}(M2) = 1\Delta C_{ox} + 2\Delta C_{ox} + 5\Delta C_{ox} + 6\Delta C_{ox} = 14\Delta C_{ox} \quad (2\text{-}36)$$

As *equations (2-35)* and *(2-36)* show, the variation of the transistors will be matched with this approach. Common centroid can also be used when the gradients are diagonal compared to the device orientation. This, however, does add complexity to the interconnects between the device parts when it comes to large devices [11].

2.3.1.4 Guidelines for layout

The theories in this subsection are guidelines, rather than methods, for better matching results.

1. Orienting devices to face the same direction is generally a good idea. This eliminates mismatch due to shadowing and other oriental effects.
2. Design devices as large as possible. This limits the effect of variation and thus makes it easier to match devices.
3. Minimum spacing. Recalling *equation (2-1)*, the probability of variation between two parameters increases with distance. So, in order to minimize the mismatch, the matching objects should be as close to each other as possible.

3 TECHNOLOGY STUDY

This chapter will start by discussing the swept-threshold sampling technique, followed by presenting the circuit of [3] and the swept-threshold sampling technique.

3.1 SWEPT-THRESHOLD SAMPLING

The swept-threshold (ST) sampling is technique suited for sampling ultra-wide-band (UWB) radar pulses [3]. It combines 1-bit quantization with a stepping threshold to convert the analog pulse to the digital domain. Elaborating on this, the input signal is compared with a threshold voltage in order to determine the digital value. To attain the full amplitude of the signal, this process needs to be repeated with the threshold voltage ramping for every quantization. This process can also be viewed as sweeping threshold. Throughout the sweep, the digital outputs are summed up, or counted, in order to profile the pulse. *Figure 27* illustrates this concept.

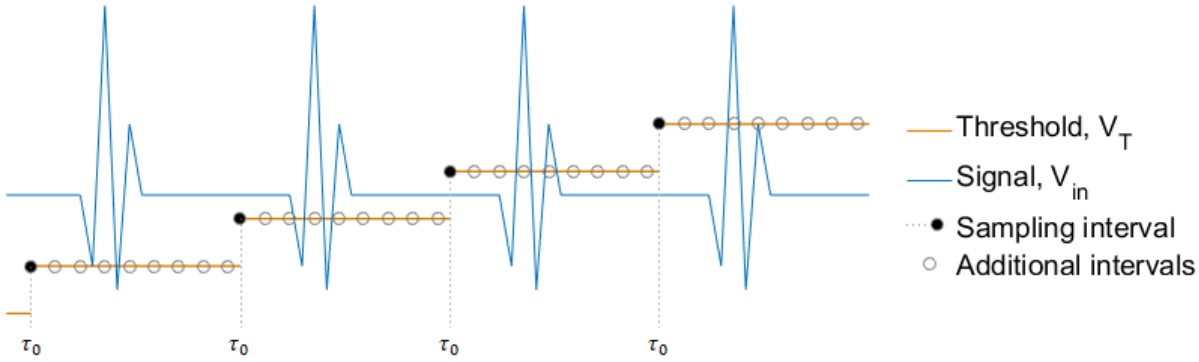


Figure 27: Principle of Swept-threshold sampling

Here V_{in} is the input pulse, V_T is the scaling threshold voltage and τ_0 is the time at which the quantization is performed. If $V_{in} > V_T$ at τ_0, then the counter counts 1 bit. Note that by only sampling at τ_0 the signal amplitude is only represented for that specific point. In order to render the whole pulse, a large number of sampling point need to be swept. The more points that are sampled, the less loss of data there will be.

To extend the number of quantization points, the swept threshold technique utilizes the time-interleaved sampling architecture. This architecture was originally created to enhance sampling speeds. Instead of relying on one converter to sample, reset, sample, and so on, multiple converter can operate in parallel, with a slight time offset, to speed up the process [12] [7]. As the *Figure 28* shown, several converters are connected in parallel with their own clock input. The sampling clocks are time interleaved in order to spread the sampling points as illustrated by circles in *Figure 27*.

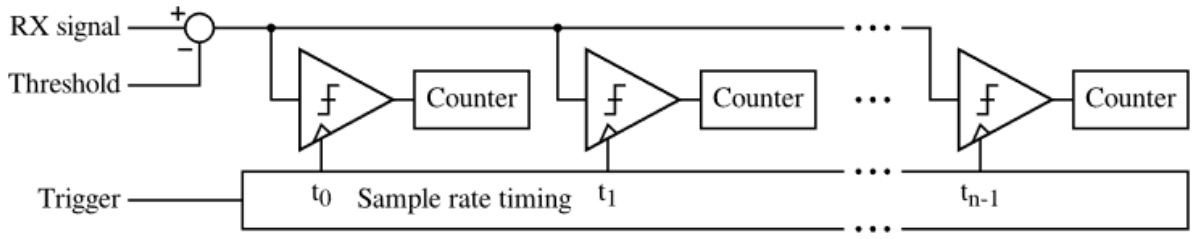


Figure 28: Block diagram of the Time-Interleaved architecture, with counters, provided by [3]

The total sampling speed of this architecture is limited by the conversion time of the analog-to-digital converter and the number of converters. Having faster converter will alleviate the number of converters needed and vice versa. Increased number of converters, however, results in larger area usage, higher power consumption, more circuit complexity and larger capacitive load. For this reason, faster ADCs are usually preferred.

Looking at the swept threshold technique again, there are some requirements that must be fulfilled in order it to work. The received pulse, that is being sampled, needs *static* throughout the whole sweep for the for the sample to not be distorted. The second requirement is that the pulses or the sampling clocks need to align for ever threshold step for the converter to profile the same sampling point throughout the sweep. This last requirement is often challenge due to skew and jitter in the clocking networks [3].

3.2 THE 118-MW PULSE-BASED RADAR

The circuit this thesis is trying to improve arrives from the paper "A 118-mW Pulse-Based Radar SoC in 55-nm CMOS for Non-Contact Human Vital Signs Detection" by Novelda AS and the University of Oslo, from 2017. This subsection will briefly explain how this sensor works. For more in-depth information about the sensor, refer to [3].

Like the title reads, this is a sensor that can detect occupants and their vital signs. This technology aims to provide remote vital signs observations of multiple subjects simultaneously. Trough that, it can also detect multiple occupants both stationary and in motion. The gains from these features are improved health monitoring in areas like hospitals, nursing homes and workplaces, and to save energy through smart building automation.

3.2.1 How it works

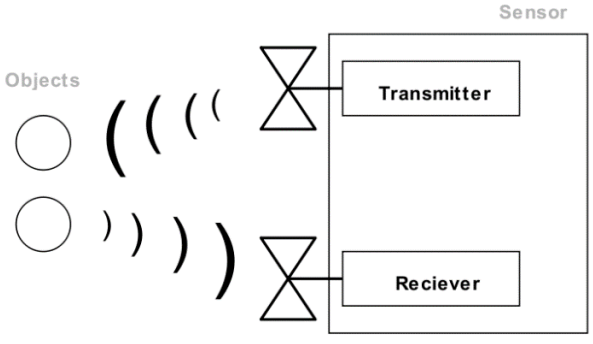


Figure 29: A simple block diagram of how a radar works

As is illustrated in *Figure 29*, the transmitter transmits frequency-shifted Gaussian pulses which reflect off objects and are sensed by the receiver. To avoid degradation of the receiver sensitivity, the pulses are transmitted in intervals giving the receiver time to read the reflected pulses without being disturbed by the transmitter. The receiver filters the incoming signals, amplifies them and finally samples them.

3.2.2 Front-end receiver

A block diagram of the receiving front-end is shown in the *Figure 30*.

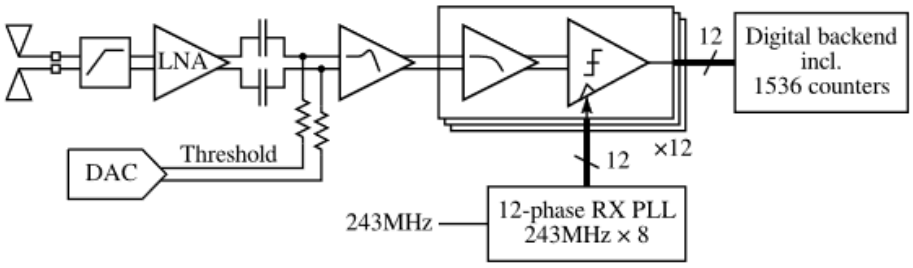


Figure 30: Block diagram of the front-end reciever, provided by [3]

The receiver in this application is required to covert signal over a wide frequency range. This is due to wide band nature of the frequency-shifted Gaussian pulses. The bandwidth of the pulses is in turn defined by the sensors required resolution for separation two objects. *Equation (3-1)* determines the relationship between the resolution ΔR and the bandwidth B.

$$\Delta R = \frac{c}{2B} \tag{3-1}$$

Here c represents the speed of light. The given equation indicates that in order to distinguish two objects separated by $50cm$, the bandwidth needs be about $3GHz$ wide. Transmitting such wideband signal, without disturbing other radio bands and not needing licenses to do so, means that the signal needs to be placed outside more regular radio bands WiFi, mobile among others. *Table 3* displays radio bands for unlicensed transmissions that can fit the required band width, for different regions of the world.

U.S.A (FFC)	3.1 – 10.6 GHz
Europe (ETSI)	6.0 – 8.5 GHz
Korea (KCC)	7.2 – 10.2 GHz

Table 3: Radio bands for unlicensed use in different parts of the world

This table indicates that the available radio band, for the ultra-wideband range required, will result in the upper frequency range exceeding $10GHz$ in some cases, and thus requiring the front-end receiver to handle these high frequencies.

Further, the receiver presented in [3] and shown in *Figure 30* is fully differential. The differential input signals go first through a high pass filter to remove noise and interferences from lower frequencies. Then a low noise amplifier amplifies the signals before removing the DC components by running the signals though their own coupling capacitors. The *digital-to-analog converter* (DAC)

biases the signals in order to perform the swept threshold sampling technique, which will be explained in the following subsection. Next, a preamplifier stage amplifies the differential signals before they are fanned out to twelve identical quantizing circuits. These parallel circuits are time-interleaved in order to profile the full signal or a pulse in this case as discussed in *section 3.1*. The parallel analog-to-digital sampling circuits are designed with an additional preamplifier and a comparator. Each parallel sampler is controlled separately by a 12-*phase phase-lock-loop* (PLL) and they each have a sampling frequency of $1.944GHz$. The binary outputs form the quantizers are finally fed into their own counter bank.

Notice that does not use sample-and-hold blocks. This is because the receiver must sample the signals directly at radio frequency in order to Nyquist criterium. [3] explains more in depth why this is the case. What this means for the sampling circuit is that the comparators need to be fast enough to in order to obtain the correct sample.

3.2.3 ST related to the case circuit

For the front-end receiver presented by [3], the pulse-sampling alignment requirement is fairly easy to control. Because the transmitter and receiver are integrated in the same chip, and use the same base oscillation, the sampling clock can be adjusted for alignment based on pulse's time-of-flight.

Furthermore, the static-pulse requirement for this application can also be assumed satisfied. This is due static positioning of the sensor, usually fixed to one position, and to the fact that the targets are moving fairly slow compared to the pulse transmission intervals of the transmitter.

The differential implementation of the sensor does introduce a small change to the swept threshold technique. Instead of the incoming pulse being swept by a threshold voltage, the differential pulses will sweep each other. This is done by making both inputs controllable and adding the incoming signal them. *Figure 30* shows the implantation of the differential sweep technique while *Figure 31* illustrates the principle of operation.

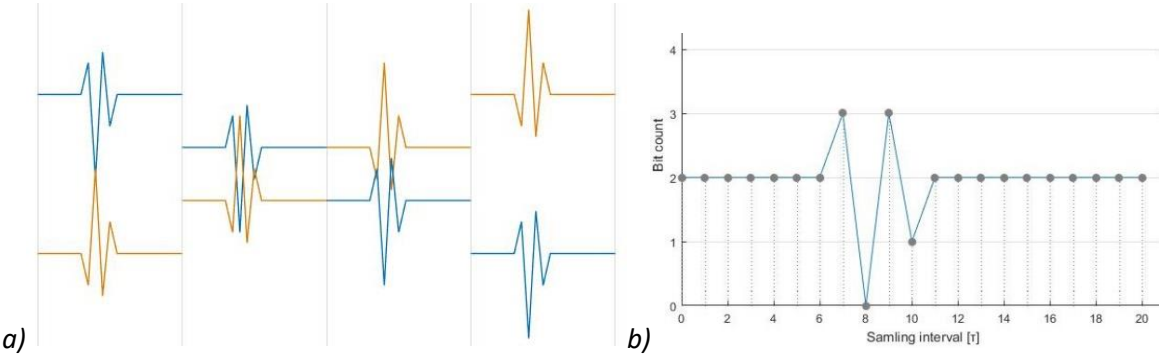


Figure 31: a) The principle of differential ST sampling, b) Resulting digital conversion of the 2-bit sweep of a)

Finally, the high frequency nature of the transmitted pulse increases the demand for fast comparators. If the comparators are slow, then an averaging effect will occur where the sampled value will represent an average value across the wide sampling period, which in turn distorts the data. Even though the case circuit utilizes 1-bit quantizes, which are inherently faster than multibit quantizers, there must be a focus on designing them as fast as possible.

3.3 OFFSET CORRECTION TECHNIQUES

In this section, the points at which offset correction can be implemented will be discussed, and the possible correction techniques will be presented.

3.3.1 Points of correction

Offset correction can basically only be implemented at the input or the output of a differential circuit component. With that being said, some components are constructed off multiple subcomponents such as multi-stage *operation amplifiers* (OpAmp) and comparators. This leads to an additional correction point which in this thesis will be defined as *in-device* correction point. The definition of the correction points may vary on behalf of the circuit perspective.

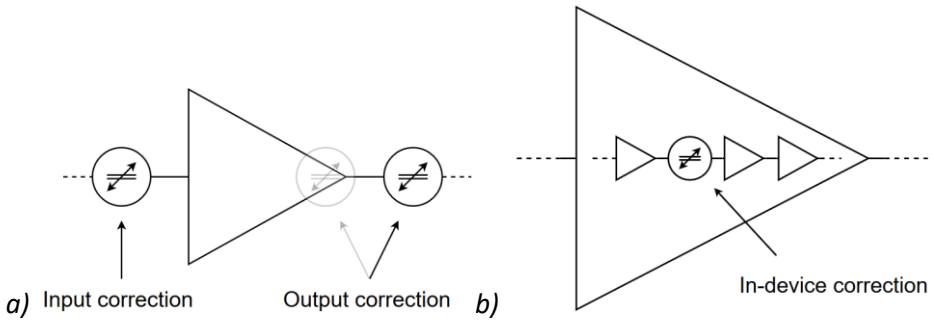


Figure 32: Definition of correction points; a) differential circuit, b) OpAmp as example

Figure 32 illustrates the discussed points. Note that the has two separate correction points. This is to illustrate that the output circuit of the component can be corrected, or an additional circuit can be implemented at the output for correction.

Correction effectiveness is a something that needs to be considered when multiple components are cascaded. Like with the case of offset of multi-stage amplifiers discussed in *subsection 2.2.1.4*, where the offset added by the later stages would impact the total offset less than prior stages, offset correction applied to the later stages will influence the overall offset less than correction earlier in the chain. In other words, correction done at the input will require less correction to correct the total offset than correction done on the output or later stages. This can be verified by adding a correction point V_{CORR} to *equation (2-18)* and solving for the total input referred offset voltage, $V_{OS_{in\,tot}}$, being zero.

$$0 = \left(\frac{V_{OS_{out1}}}{A_{d1}} - V_{CORR}\right) + \frac{\frac{V_{OS_{out2}}}{A_{d2}}}{A_{d1}} \tag{3-2}$$

In *equation (3-2)* the correction point is implemented at the input of a two-stage gain circuit.

$$V_{CORR} = \frac{V_{OS_{out1}}}{A_{d1}} + \frac{\frac{V_{OS_{out2}}}{A_{d2}}}{A_{d1}} \tag{3-3}$$

Equation (3-3) shows that the amount of voltage correction applied at the input is decreases with increasing gain from the gain circuits.

Now, implementing the correction at the output of the second stage gives these equations.

$$0 = \frac{V_{OS_{out1}}}{A_{d1}} + \frac{\frac{V_{OS_{out2}} - V_{CORR}}{A_{d2}}}{A_{d1}} \tag{3-4}$$

$$V_{CORR} = (V_{OS_{out1}} A_{d2}) + V_{OS_{out2}} \tag{3-5}$$

Equation (3-5) states that the amount of correction needed increases with the gain of the second amplifying stage.

In essence, offset correction techniques are more effective the earlier in a gain circuit they are implemented. This statement is only supported if all stages yield gain and not loss. In some cases, gain may be minimal and thus not affect the effectiveness of the correction.

3.3.2 Input correction

The principle of input offset correction is to adjust the input biasing voltage of the two input gates of a differential amplifier or a comparator. With this adjustment, the input referred offset voltage can be cancelled out.

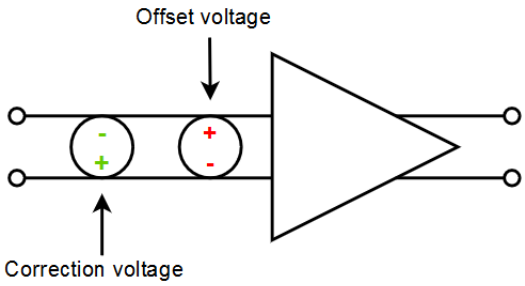


Figure 33: Concept of input offset correction

There are many ways of controlling the input biasing voltage of the inputs. *Figure 34 a)* shows a simple method which uses voltage dividers to control the biasing voltages. Here the voltage can be adjusted by controlling the resistance values of the divider. The controllable resistance can be realized through various resistive switching networks or even through postproduction film-trimming. Either way, the input gates need to be DC-separated for the original biasing points provided by the previous circuit stage. To achieve this, coupling capacitors are implemented on both input lines to only allow the signal to pass though.

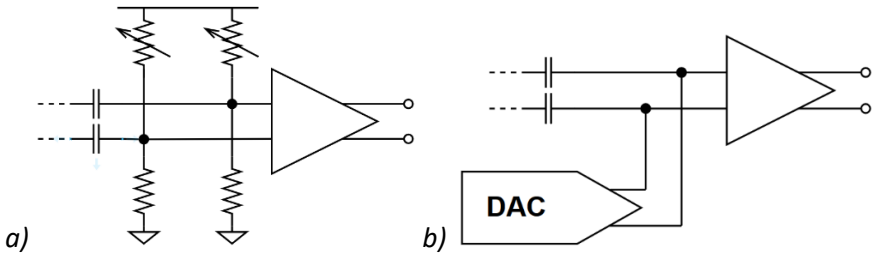


Figure 34: Input correction technique with, a) division, b) a DAC

Figure 34 b) illustrates another approach for controlling the input biasing point. Here a DAC is used to correct the offset instead of voltage dividers. Again, coupling capacitors must be used to isolate the DC voltage.

However, a more common method for input correction is to add a differential amplifier to the inputs and correct its outputs. By using a differential amplifier, the coupling capacitors can be removed. This is preferred when it comes integrated circuit because integrated capacitors are area inefficient and they add parasitic capacitance which slows down the operation speed and limits the frequency response.

There are several techniques for correction the output of a differential amplifier. The following subsections will introduce the techniques.

3.3.3 Resistive load correction

This is a technique used for differential amplifiers. The concept of this technique is to adjust the drain resistance of the amplifier. By changing the drain resistance, the current going through the resistor changes and thus the voltage across it changes. This gives the ability to set the output node voltages to desired levels for correcting the offset of the next circuit or to cancel out the amplifiers own offset, or both.

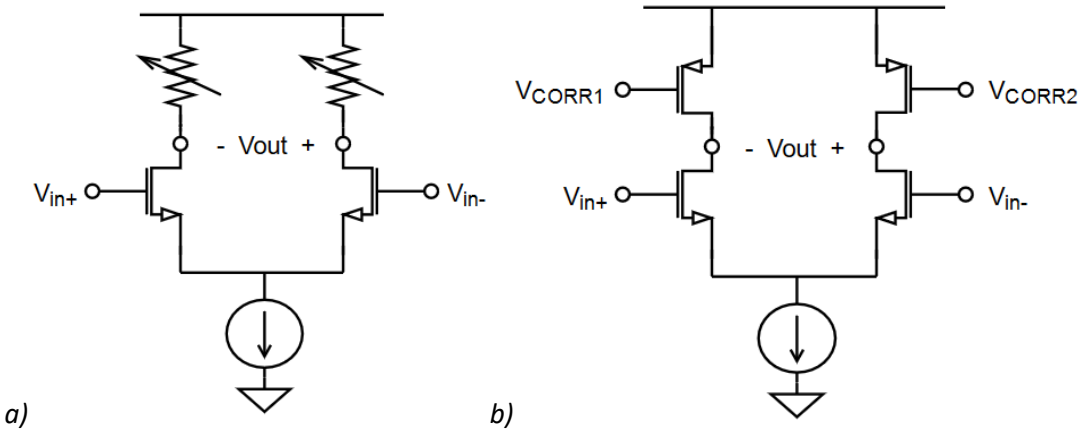


Figure 35: Drain resistance correction

Figure 35 a) illustrates the load resistance technique. The adjustable drain resistors can be realized with film-resistors that are laser trimmed post fabrication or with programable resistor networks [16]. Post-fabrication laser trimming is a costly process and thus rarely used. *Figure 35 b)* show a third option for correction where transistors are used as active loads. The drain-to-source resistance of the transistor can be changed by regulating the gate-to-source voltage.

3.3.4 Current correction

The current correction technique evolves around adding additional current paths where the current going through them is controlled. This technique can be implemented for amplifiers as well as comparators. However, the effects are not the same for those two cases.

In the case of the differential amplifier, the added current paths correct the offset by adjusting the currents going through the drain resistors. The added paths can be realized with a differential current-DAC [17] [18], with transistors in parallel to the input transistors [19], or even with transistors bypassing both the input transistors and the tail-current source. The first two are illustrated in *Figure 36 a).* One thing to consider when it comes to leading the current out of the amplifier, is that the correction will add current to the drain resistors instead of balancing the current of the differential branches. The result of this is that the voltage across the resistors only can get larger and thus reduce the available voltage range for the input transistors.

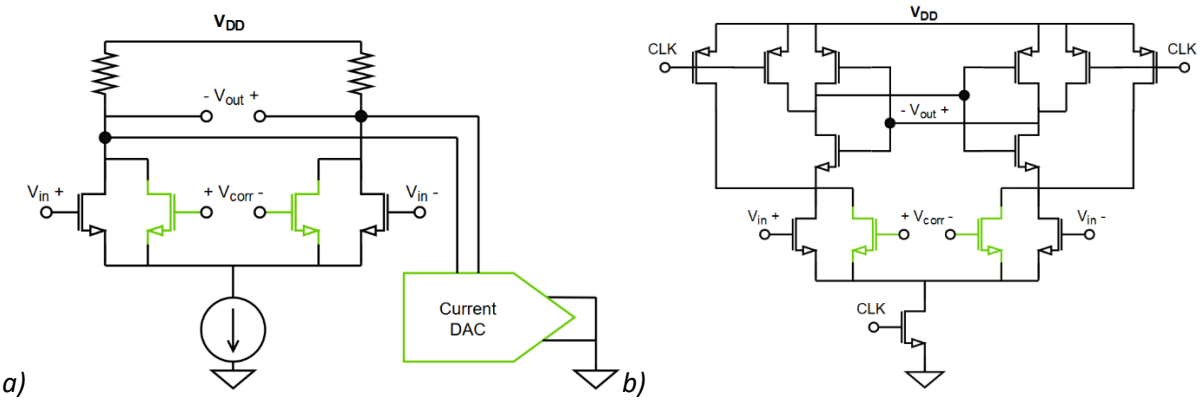


Figure 36: Current correction implementations for a) differential amplifiers, b) dynamic comparators

In the case of the dynamic comparator, the adjustable current regulated the discharge rates of the differential branches. By doing that, the sampled offset and the internal offset can be cancelled out.

$$V_{node}(t) = V_{DD} - \frac{I\,t}{C} \tag{3-6}$$

*Equation ((*3-6*)* shows that the voltage of a precharged node in the comparator, drops with the rate of $\frac{I*t}{C}$. Thus, the larger the current or smaller the capacitance, the earlier the amplification and regeneration of a branch is activated.

The only way of implementing this type of correction in a dynamic comparator while keeping the dynamic feature, is to implement transistors in parallel with the input transistors [20]. If the added current paths lead out of the comparator, then parts of the comparator will remain active when the clock goes low. This will turn the comparator into a regular latch and thus hysteresis will be a problem.

3.3.5 Threshold voltage correction

This technique aims to correct offset by altering the threshold voltages of transistors. By changing the threshold level of a transistor, the current going through the transistor affected. *Equation (*3-7 shows the relationship between the drain-to-source current I_{ds} of a NMOS transistor and the threshold voltage V_{tn}.

$$I_{ds} = \frac{\mu_n C_{ox}}{2} \frac{W}{L} (V_{GS} - V_{tn})^2$$

(3-7)

The effects of altering the currents in a differential amplifier and a dynamic comparator are the same as in the case of the current correction technique. For the differential amplifier, the change in current affects the voltage across the drain resistance and thus the output biasing point. As for the comparator, the current affects the discharge rate.

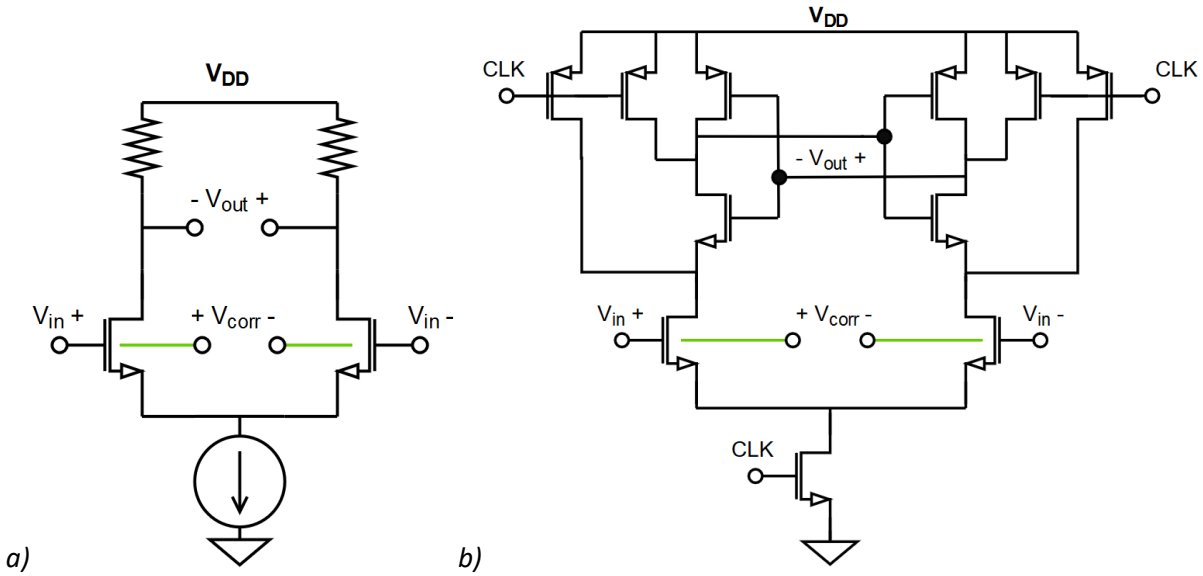


Figure 37: Threshold voltage correction in a) differential amplifier, b) dynamic comparator

The only way of change the threshold voltage of a transistor without altering any physical parameters is to adjust the bulk voltage of the transistor [21]. *Equation (3-8)* shows how the source-to-bulk voltage V_{SB} affects the threshold voltage in the case of a NMOS transistor.

$$V_{tn} = V_{tn-0} + y(\sqrt{V_{SB} + 2\phi_F} - \sqrt{2\phi_F})$$

(3-8)

Figure 37 illustrates where the correction points are for the differential amplifier and the dynamic comparator.

However, the bulk of a NMOS transistor is normally the substrate of the circuit. This means that in order to control the bulk of the transistor, the substrate of that transistor needs to be isolated for the main substrate. This can be realized by creating a triple-well for each of the transistors being corrected. The downside here is that additional wells increase the area usage of the transistors and that the transistors are wider spaced due them needing their own wells. In the case of the PMOS transistor, it will not need deeper or additional wells because it already has a N-well. However, for this correction technique the paired PMOS transistors will need their own wells.

A critical limitation for bulk correction is the voltage range that can be applied to the bulk. Since the bulk effectively creates a diode connection (pn-junction) to the source of the transistor it is important that voltage difference between those metals does not exceed the threshold voltage for the materials. Exceeding this threshold will lead to current leakage through the source gate.

3.3.6 Capacitive load correction

Capacitive load correction is a technique used in dynamic comparators. It can be categorized as output correction because the offset is adjusted for at the output of the comparator. As with the comparator current correction technique, the principle of this correction is to alter the discharge rate of the out nodes so that the correct polarity is represented after regeneration.

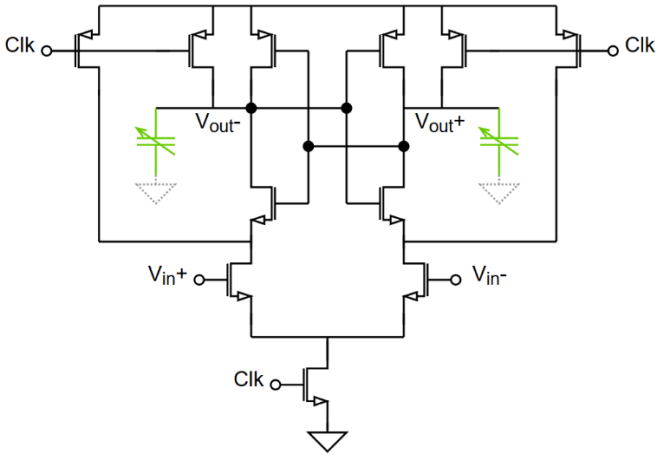


Figure 38: Capacitive load correction

In contrast to current correction, this technique changes the discharge rate by changing the capacitance of the output node, as illustrated in *Figure 38*. The capacitance can be adjusted either by programable capacitor-networks [22] or by capacitor-connected transistors [23]. The capacitance of a capacitor-connected transistors is determined by the voltage difference gate terminal and the other shorted terminal.

4 DISCUSSION FOR IMPLEMENTATION

In this chapter, the sampling circuit that needs to be corrected will be determined. Also, the correction technique that is best suited for the sampling architecture will be discussed and determined. The chapter will end with presenting the circuit that will be designed and the circuit that will be implemented on chip. Specification and requirements for the design will also be presented and discussed throughout this chapter.

4.1 IDENTIFYING THE CIRCUIT

To identify the circuit that will be improved, the potential mismatches of the front-end receiver need be discussed. *Figure 39* highlights the components that rely on matching, assuming that the signal paths are matched.

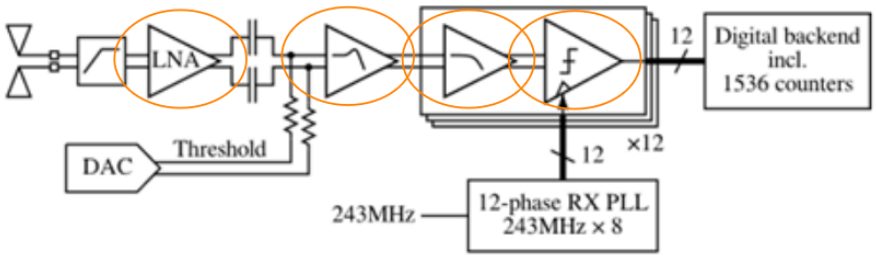


Figure 39: Block diagram of the front-end receiver with identified offset sources

Starting with the *low-noise-amplifier* (LNA). The amplifier is prone to offset but the offset does not affect the analog-to-digital conversion, assuming that the amplified offset is small enough to keep the amplified incoming signal within the supply range. This is true because the coupling capacitors block the DC voltage from the LNA and allow only the signals to pass. For this reason, the LNA will not be implemented.

Moving on to the bandwidth extending differential amplifier. The offset generated by this amplifier will affect the sampling of the of the signal moving the common-mode voltage level for the parallel quantizers. There is however two reason for not correction its offset. The first one is that the offset is common for all quantizers and thus will only move the digital representation of the signal up or down the voltage ladder. Theoretically, this would also result in need for wider sweeping range, less resolution and higher power consumption, but next reason corrects this. The second reason for not focusing on correcting this offset is that amplifier already has a potential correction circuit, the threshold sweeping DAC. As was presented in *section 3.3.2*, a DAC can be used to correct the input of an differential amplifier. Had it not been for the parallel quantizers at the output of this amplifier, the DAC could have corrected the offset for the whole sampling circuit. Either way, the bandwidth extending amplifier will not be considered for correction in this master project.

Finally, the interleaved quantizers are discussed. The quantizers are affected by offset the most in this sampling architecture. In a quantizer, both the preamplifier and the comparator contribute with offset. Also, both components are designed very small in order to reduce the total load capacitance of the bandwidth extending amplifier. According to *equation (2-1)*, this will lead to larger mismatch in these components. Furthermore, every quantizer has their own *local* offset. These offsets will result in amplitude errors at the individual sampling intervals of the full sampling profile. Because of these reasons, the goal for this project is to improve the offsets of the quantizers.

4.2 IDENTIFYING THE CORRECTION TECHNIQUE

The correction technique being implemented is chosen in regard to the main requirements needed for the swept threshold sampler of [case paper] to function properly. These requirements are:

- High frequency response, in order to sample the ultra-wide-band pulses.
- High quantization speed, in order to meet the Nyquist rate needed for the high frequency nature of the received signals, because there is no sample-and-hold circuit.

The goal here is to implement offset correction without affecting the sampling performance of the sampler. With that being said, the techniques are evaluated as follow.

4.2.1 Input correction

The input correction technique, where programable voltage dividers or a DAC is used, could be implemented at the input of the differential preamplifier or at the input of the comparator. Implementing it on the input of the amplifier would result in better correction effectiveness compared to the input of the comparator. However, both these implementations would require coupling capacitors to work. These capacitors would in turn result increased capacitance in the sampler and thus the high frequency response requirement would not be hard meet.

Using the output of the already existing differential amplifier, on the other hand, to correct the offset on the input of the comparator is a much more viable solution. This method would not require any capacitors for it to work. In addition, by using the existing amplifier would require less area usage compared to the voltage dividing method. The only slight drawback here is that the correction efficiency will be less. But, with the transistors of the amplifier being small, in order to reduce the input capacitive load, the gain will not be that high and thus there will only be small efficiency loss.

4.2.2 Resistive load correction

As for correcting at the output of the differential amplifier, the resistive load technique can be used. The three methods for implementation presented in *subsection 3.3.3* are, film-resistor trimming, programmable resistor networks and active load adjustment.

Film-resistor trimming would have no negative impact on the performance of the sampling circuit. The area usage would be minimal, and no additional capacitance would be introduced. This method would also not consume additional power because it needs no support circuits. However, this implementation allows for a one-time correction only and the cost of this correction is very high.

The programmable resistor network approach allows for easy correction because it is controlled digitally. The downside with this method, in regard to this particular sampler, is that it will add capacitance to the circuit. Resistor networks are usually realized though resistor arrays or ladders. These networks have several resistors connected to the resistor terminal which can result in large

overall parasitic capacitance, which again means that the signal propagating of the sampler would suffer. An additional point worth noting here is that in order to increase the resistive resolution more resistors must be added, and thus even more capacitance is introduced.

The active load method, realized with transistors, would require headroom to operate if it is implemented. This can cause problems because the voltage supply for the sampling circuit is only $1.2V$. Having three transistors stacked within this range would limit the operating range of the amplifying transistors. Active loads do also make the design phase more challenging due to the extra dynamic complexity they bring. Furthermore, small resistors translate to high internal resistances which in turn limit the possible currents going through the differential branches of the amplifier. To reduce the internal resistances, the transistors can be designed wider but this also means increase in parasitic capacitance.

Another thing to consider here is the effect resistor correcting has on the gain of the amplifier. Changing the drain resistances will have a first order effect on the gain of the differential amplifier. This is true for all three methods of this technique.

4.2.3 Current correction

The current correction technique introduces an additional point for correction. This point being *in-device* of the comparator.

Evaluation the current correction for the differential amplifier first. Using additional transistors or current-DACs to adjust the current flow through the parallel branches will alter the voltage over the drain resistors and thus can compensate for the offset. However, implementing the transistors or current-DACs will lead to an increase in capacitance and limit the frequency response of the sampler again. Out of these two methods, the single transistor would add less capacitance than the DAC. Also, by adding additional current paths, there will be less current going through the input transistors. The drain current is a second-order parameter of the total gain so the gain of the amplifier will be reduced. Since the gain of the sampling circuit already is small, this could become a greater concern because it will influence the metastability of the comparator.

As for implementing current correction in-device of the comparator, this technique will introduce the same unwanted effects. The parasitic capacitance added correction components will directly extend the discharge time of the nodes they are connected to. Increase in discharge time means that the time from when a sample is triggered to the time the comparator enters the regenerating phase is longer. This of course, is not desired when the circuit is sampling the signal directly. The second effect affecting the amplifier, loss in gain, also is true for the comparator where the metastability is affected. A third consequence of implementing the offset correction in the comparator, is that the correction effectiveness becomes even less than the output of the amplifier.

4.2.4 Threshold voltage correction

The threshold voltage correction technique can, like the current correction, be implemented both at the output of the amplifier and in-device of the comparator.

This technique does not introduce any additional capacitance to the signal paths. Since the correction is applied at the substrates of the input transistors, there is no need for additional components to correct the offset. The frequency response of the sampler is thus not affected. However, the concept of this technique is to adjust the transconductance of the transistors in order to correct the offset. This will of course translate to reduction in gain because change in threshold is a third-order effect. Another drawback of this technique is that in order to control the substrate voltages the substrates need to be isolated. This isolation is realized by implementing individual double-well for the transistor. In addition to increasing the area of the transistor pairs, this implementation will separate the transistor and thus increase the probability for mismatch and offset. Because the two input transistor pairs will be fairly small, this can result in large offsets.

The discussed effects here do apply for both correction points, output and in-device. In-device implementation for the comparator will, like for current correction, result in less effectiveness.

4.2.5 Capacitive load correction

The capacitive load correction technique only adjusts the capacitance of nodes which means will not have any effect of the output of the differential amplifier, other than limiting the frequency response. In other words, this technique will only work at the in-device point of the comparator.

As presented in section 3.3.6, this technique can be realized through the capacitive-network or the capacitor-connected transistor implementation method. Both these methods will slow down the sampling time of the comparator due to increased discharge time, which is not desired. Also, the point of correction is at the backend of the sampler meaning that the correction effectiveness is low.

4.2.6 Correction size comparison

This section has thus far not considered the silicon area needed for these correction techniques to be implemented. That is because the corrections techniques and implementation methods need to be set in perspective and some assumptions need to be made.

To define the area usage, one needs to consider how the corrections methods are controlled, and in which way they need to be controlled. Techniques such as the current correction technique and the active-load resistance correction technique are controlled by voltage inputs, while the capacitive-network load correction is controlled digitally. At first glance, the voltage-controlled methods seem to need much less area to be implemented than the network method does. However, the voltage-controlled methods will need have their voltages converted from digital values. This is because it is more practical to adjust or program devices in integrated circuit digitally.

In addition to the control perspective, some other parameters need to be addressed. Firstly, the sizes of the impedance networks rely on the allowed impedance-per-unit-area ratio of the fabrication process, and the correction range and correction resolution needed to correct for the offset. To simplify these parameters, this thesis will assume that capacitors and resistors of the impedance-network take up the same amount of area. Furthermore, the range is assumed to increase with the decrease of correction effectiveness, while the resolution is assumed to decrease with the decrease of effectiveness. This means that the range and resolution parameters cancel each

other out and thus are not taken into consideration. Secondly, the digital-to-analog converters used for the correction techniques and for the voltage-controlled inputs assumed to use equal amount of space. Also, they are assumed of the same size a one impedance network. This is because DACs can operate differentially, and thus serve two inputs, while the networks only have one "output". Thirdly, the active load transistors and the current correction transistors are either replacing existing components or are so small in size. For this reason, they are assumed to not affect the area usage. Finally, the two double-well implementations of the threshold correction and the two coupling capacitors for input correction, are assumed to be $\frac{1}{4}$ of the size of the DACs. And, the two capacitor-connected transistors are assumed to be $\frac{1}{8}$ of the DAC size.

To present the comparisons more easily, the discussed parameters are normalised to the DAC size and put in a table. The following abbreviations will identify the techniques and methods:

- INC, Input correction
  - VD, voltage dividers
  - DAC, digital-to-analog converter
- RLC, resistive load correction
  - FRT, film-resistor trimming
  - PRN, programable resistor network
  - AL, active load
- CC, current correction
  - TR, transistors
  - CDA, current DAC
- TVC, threshold voltage correction
- CLC, capacitive load correction
  - PCN, programable capacitor network
  - CCT, capacitor-connected transistors

	Network	DAC	Double-wells	Coupling capacitors	Capacitive transistors	Total size
INC-VD	2	-	-	0.25	-	2.25
INC-DAC	-	1	-	0.25	-	1.25
RLC-FRT	-	-	-	-	-	0
RLC-PRN	2	-	-	-	-	2
RLC-AL	-	1	-	-	-	1
CC-TR	-	1	-	-	-	1
CC-CDA	-	1	-	-	-	1
TVC	-	1	0.25	-	-	1.25
CLC-PCN	2	-	-	-	-	2
CLC-CCT	-	1	-	-	0.125	1.125

Table 4: Area usage comparison of the correction techniques

According to *Table 4*, the resistive load correction with film-resistor trimming does not contribute to any additional area usage. Furthermore, the table shows that the correction techniques that use impedance-networks are about twice the size of the techniques using DACs. The DAC using techniques are fairly similar in size with the threshold voltage correction and input DAC correction techniques using the most space out of them.

4.2.7 Technique for implementation

Now, with all the techniques and methods being evaluated and the sizes of them being compared, the best correction technique for the circuit being improved in the master project can be determined.

To begin with, the film-resistor load correction technique can be neglected as an option. Even though it does not add any area to the final implementation, the cost and time of correcting the offset this way is too great.

As for the rest of the techniques, the requirements for the sampler must be considered. To meet the first requirement of high frequency response, the signal path, form the input of the differential amplifier to the input of the comparator, should experience as little capacitance as possible. The only technique that does not affect the capacitance of the signal path is the threshold voltage technique. The technique that add the most capacitance is the input correction technique with voltage division and DAC. This is due to the coupling capacitors and the parasitic capacitances the programable networks. For this reason, they should not be used for offset correction in this sampling circuit. Also, the resistor-network load correction and the DAC current correction implementations should be avoided because of the parasitics they introduce. The other two correction options for correcting at the output of the differential amplifier, active load correction and current correction with transistors, do add capacitance to the path but much less than the programable networks do.

With the other sampler requirement, high quantization speed, in mind, the correction techniques should retain as much gain as possible and not affect the discharge rate of the comparator. As for the first goal here, retaining gain, only the capacitive load correction techniques, besides the already excluded input correction techniques, do not affect the gain. The resistive load correction techniques affect the gain the most because their adjustment is a first-order component of the amplifiers gain. As for the other possible techniques, they do affect the gains of both the amplifier and the comparator, if implemented in them, but to a lesser degree. With regard to the second contributor to reduce of quantization speed, discharge rate, the only correction techniques that influence that are the possible in-device for the comparator. Here the capacitive load correction will introduce the most discharge rate extension and should thus be avoided. The DAC current technique would also introduce a fair amount of capacitance, while transistor current technique would just slightly affect the discharge rate compared to the others. The final in-device technique, threshold voltage correction, will not add any capacitance as in the case for the amplifier.

From these discussions related to the sampling requirements, the technique that fits the sampling circuit the best is the threshold voltage correction technique implemented for the output of the differential amplifier. Even though the correction also could be implemented in-device of the comparator, implementing it for the output of the amplifier yields better correction effectiveness. Furthermore, this technique does neither affect the frequency response of the sampling circuit nor the discharge rate of the comparator. It does however affect the gain of the sampler, and it does increase the possible offset of the differential transistor pair on the amplifier. Because the offset is corrected anyway, the increase in offset will not be a problem. As for the area usage of this correction, with regard to the assumptions made for the sizes, it will be about half the size of the impedance-network technique and slightly larger than the smallest techniques.

4.3 CIRCUIT FOR IMPLEMENTATION

With the circuit for offset improvement identified, and the corresponding best fit offset correction technique. The implementation will be discussed here.

As was found in *section 4.1*, the part of the swept threshold sampler presented in [3] that is the most affected by offset is the interleaved quantizers. Each quantizer experiences their own offset which in turn results in overall sampling errors. For this reason, offset correction will be implemented for the quantizers.

In practice, only a single quantizer is needed to prove the correction ability of the threshold voltage correction technique. However, in order to have a more realistic implementation case, the full sampler should be designed. By designing the full sampling circuit, the dimensions and characteristics of the sampler, and thus the components, are better represented. This will in turn help show the effects the offset correction will have on the sampler.

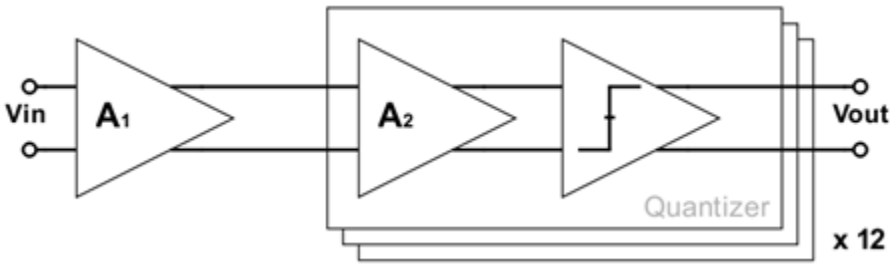


Figure 40: Interleaved sampling circuit for implementation

Figure 40 shows a block diagram of the circuit being designed. The goal here is to design this circuit as close to the specifications and performances of the one in [3]. By doing that, the requirements and limitations of the sampler will be highlighted. The specification the design will try to meet are:

- $12GHz$ frequency response, on behalf of the ultra-wide-band characteristics of the transmitted signal and the available radio band than can be used.
- Sampling rate of $2GHz$ for the interleaved quantizers, to satisfy the Nyquist sampling rate requirement for overall sampling rate of $24GHz$.
- $9dB$ gain combined for the two amplifiers leading to the comparator.

In addition to the sampling rate requirement, there will be a focus on reducing the decision time of the quantizers in order to minimize quantization averaging of the signal.

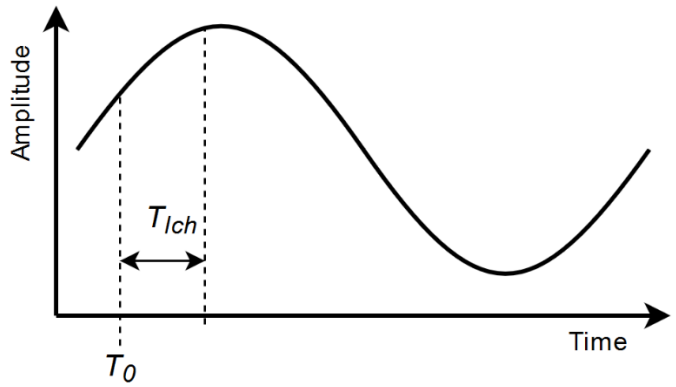


Figure 41: Quantization averaging

Figure 41 illustrates the consequence of quantization averaging, where instead of just enquiring the signal amplitude at the triggering instance, τ_0, the decision time of the comparator, τ_{lch}, introduces an averaging effect. For dynamic comparators, τ_{lch}, is determined by the discharge rate and metastability.

4.3.1 Design

With these requirements in mind, the sampling circuit will be designed with the following components.

- Differential amplifier with frequency extension, to compensate for the load of the interleaved quantizers.
- Resistor loaded differential amplifier with triple wells, in order to incorporate offset correction.
- Improved StrongARM comparator, because of its improved quantization speed compared to the original StrongARM.

The frequency extension techniques depend on the resulting capacitances of the design, and thus will be chosen during the design. Also, these techniques could be implemented for each quantizer but because that would lead to very large silicon area being used, the extension will only be limited to the input amplifier.

When it comes to realizing the correction technique, the only things that is needed, in addition to isolated bulks, are two programable voltage outputs to control the bulk voltages. To achieve this the correction circuit be design as illustrated in *Figure 42*.

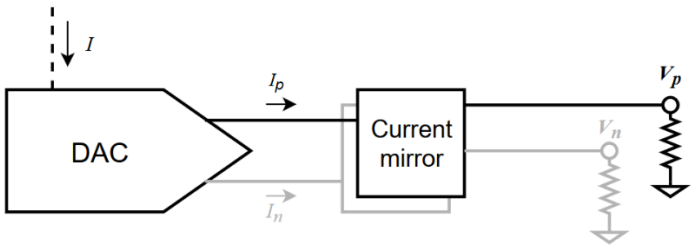


Figure 42: Programable voltage outputs for threshold voltage correction

Here a differential current-DAC will be used to control the currents going through the passive resistors. The currents will however go through current-mirror networks, in order to remove the voltage-drop across the DAC, before entering the resistors. The resistors will be sized with regard to the needed voltage range for the correction.

Additional support components such as input shift registers, output buffers and output latches will be presented in *chapter 5*.

4.3.2 TSMC 65-nm Implementation

With the characteristics of the sampling and the correction circuits being verified though the design, only a single quantizer with correction will be implemented on the chip. This is because implementing the whole circuit without the sampling clocking network would be pointless. Also,

implementing the full sampler would demand a time-frame larger than the frame of this master project. Either way, by analysing the sampler's performance though simulations, conclusions can be made.

As a conclusion of this subsection, a single quantizer with offset correction will be implemented on chip. The process used for implementation is the 65-nm TSMC process.

5 IMPLEMENTATION

In this chapter the design process will be presented as well as layout process. There will also be discussions of the parameters that validate the designs and simulations that show the characteristics of the resulting designs.

For the sake of better overview, the full design is divided into three parts, analog design, correction design and digital design. The analog design part will present the implementation of the analog-to-digital converter. Then the correction design part will present the offset correction technique that is being implemented on the analog circuit. And finally, the digital design part will present the digital components that must be in place for the sampling circuit and the correction circuit to operate.

One thing to note here is that the parasitic extraction tool was not available throughout the design and implementation process. This means that post layout simulations could not be done to verify the final implementation. However, post layout simulation will be presented at the end of the chapter. These simulations were performed after the chip was sent to fabrication.

5.1 ANALOG DESIGN

As established in *section 4.3*, the specifications the sampler will be designed to meet are:

- $12GHz$ frequency response with 12 interleaved quantizers
- $2GHz$ sampling rate for the quantizes
- $9dB$ gain in total for the two amplifying stages
- As close to $0s$ decision time for the comparator

The latter point is more of a target than a specification and is directly related to the sampling rate.

To meet these specifications and for other practical reasons, the analog design approach will be as follow:

1. Designing the comparator. Meeting the sampling rate requirements is the highest priority. By designing the comparator first, it will not need to depend on the preamplifier leading to it and thus it can be optimized for best performance. This means that rather than designing the comparator for the preamplifier, the preamplifier is designed for the comparator. Although the gain of the preamplifier improves the sampling rate and decision time of the comparator, the comparator should be able to reach requirements on its own.
2. Designing the preamplifier of the quantizer. With the comparator being designed, the preamplifier will have some pre-set requirements it has to meet. The requirements will be to bias the outputs of the amplifier to match the optimal input biasing voltage of the comparator, and to meet the bandwidth requirement with the load of the comparator. This means that the amplifier should be designed for bandwidth rather than gain.
3. Designing the first stage amplifier with bandwidth extension. With the load of the quantizers being established, the extension techniques can be evaluated, and the amplifier can be designed accordingly. Here again, the amplifier needs to be designed so that the output biasing voltage matches the required input biasing voltage of the preamplifier.

As a final note, all the transistors in the analog design will be realized with low-threshold RF-transistors. This is because RF-transistors are better isolated from noise, and because the supply voltage the circuits are designed for is $1.2V$ the low-threshold transistors will give more voltage range to design for. In other words, low-threshold transistor needs less drain-to-source voltage to reach saturation.

5.1.1 Comparator design

5.1.1.1 Design parameters

Before designing the comparator, the parameters that determine the comparators performance needs to be defined, and the methods of validating these need to be presented.

For the improved StrongARM comparator, the decision time determines the highest possible sampling rate. This decision time can further be determined by the metastability on the comparator and the discharge rate of the precharged node in the StrongARM comparator. As discussed in *section 2.2.2.3*, the precharged nodes need to drop by V_{th} in order for the transistors to start operating and thus the time needed to discharge the nodes affects the decision time. Also, the metastability of a comparator represents the gains of the amplification phase and regeneration phase. For these reasons, the two parameters characterizing the performance are discharge time and metastability.

The metastability and the discharge time will in this design process be found through the measurement method presented in the "ADC verification workshop" [24]. The method introduces a small differential voltage at the input of the comparator before triggering a sample. The resulting transient analysis, as the one in *Figure 43*, is then used to determine the parameters.

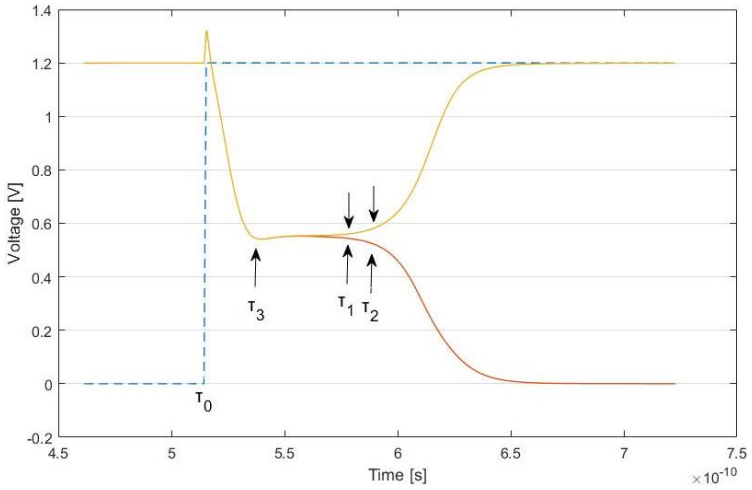


Figure 43: Transient analysis of the output of a dynamic comparator

The metastability parameter τ_m is determined by the time difference between time τ_1 and τ_2. τ_1 represents the time at which the comparator outputs the differential voltage of the input V_{in_d}, while τ_2 represents the time at which the comparator outputs the natural logarithmic gain of V_{in_d}.

$$\tau_m = \tau_2 - \tau_1 = \tau\left(V_{in_d}e^1\right) - \tau(V_{in_d}e^0) \quad (5\text{-}1)$$

The discharge time parameter τ_{dr} can be determined by the time difference of the sample being triggered τ_0 and the time outputs going flat τ_3, meaning that the discharge phase is finished.

$$\tau_{dr} = \tau_3 - \tau_0$$

(5-2)

In conclusion, the smaller the values of both τ_m and τ_{dr} are the better sampling performance of the comparator is. The design process of the comparator will aim to reduce these parameters as much as possible.

5.1.1.2 Factors considered for the improved-StrongARM comparator design

Throughout the background chapter, various factors have been discussed for transistors, amplifiers and latches. In order to have a better overview of these factor, a short summary of the factors considered during the improved-StrongARM comparator design are listed here:

- The metastability of the comparator is determined by the gain of the cross coupled inverters and the input differential transistor pair. For these components, the gain can be represented as common-source amplifiers. The gain then is a product of the transconductance g_m and output resistance r_O of the transistors. The g_m is affected by the width-to-length ration $\frac{W}{L}$ of the transistor, where greater $\frac{W}{L}$ results in better g_m. However, r_O decreases with increase of $\frac{W}{L}$.
- The discharge time τ_{dr} of a StrongARM comparator is determined by charge potential of the nodes that need to be discharged, and the discharge rate. The node charge potentials are derived by the total parasitic capacitances of the transistors connected to them. These parasitics are again proportional to the widths and lengths WL of the transistors. The discharge rates of the nodes are determined by the current going from the nodes, which in turn relies on the g_m of the transistors the current is being pulled through, hence $\frac{W}{L}$.
- The input capacitance of the StrongARM latch will affect the total output capacitive load of the preamplifier, and in turn reducing the frequency response. This input load is determined by the parasitic capacitances of input transistor. Again, WL is proportional to the capacitance.
- The potential mismatch of this comparator, and hence the offset, is inversely proportional with the WL. Of the matching transistors. Having greater WL decreases the potential offset of the paired transistors. Also, the offset dominancy follows the order on which the transistor pairs are activated.

5.1.1.3 Final design of the improved-StrongARM comparator

With the factors discussed in the previous subsections in mind, the following sizing approach is taken.

As a whole, the channel lengths L should be as short as possible. The advantages of having a short L is increased transconductance due to $\frac{W}{L}$ and decease of parasitic capacitance due to WL. On the other hand, det disadvantages are smaller output resistances and greater offset potential. However, since the aim of the sampler is to have high bandwidth and the goal of the comparator is to support high sampling rates, using minimum length channel lengths will be a good solution. Even though this will result in larger offsets, the circuit is being offset corrected anyway so it will not make a huge difference.

The input differential-pair M1-M2, seen in *Figure 44*, should be sized for gain. A slightly higher input capacitance would not affect the frequency response that much if the preamplifier is designed properly. By sizing for initial gain, the metastability is improved and thus the error probability is less.

The differential cross coupled pairs, M3-M4 and M5-M6, do not impact the input load. For this reason, they can be sized larger in order to improving the regeneration gain and reduce their offset contribution. However, these effects need to be evaluated with regard to the discharge rate.

The two precharging transistors, M8 and M9, do not affect the regeneration or the metastability with their transconductance. Their parasitic drain capacitance adds to the total output node capacitance. For these reasons it is desired to size the transistors as narrow as possible to improve the discharge time. The only factor that limits the minimum width of these transistors is the current they need to supply the precharged nodes with during the reset phase.

Another parameter that needs to be considered is the input common-mode voltage V_{CM}. If the voltage is to low, the discharge rate is reduced. On the other hand, if the voltage is too high then the preamplifier could struggle to create any gain because of the output resistance being small due to the small voltage range available for the resistors.

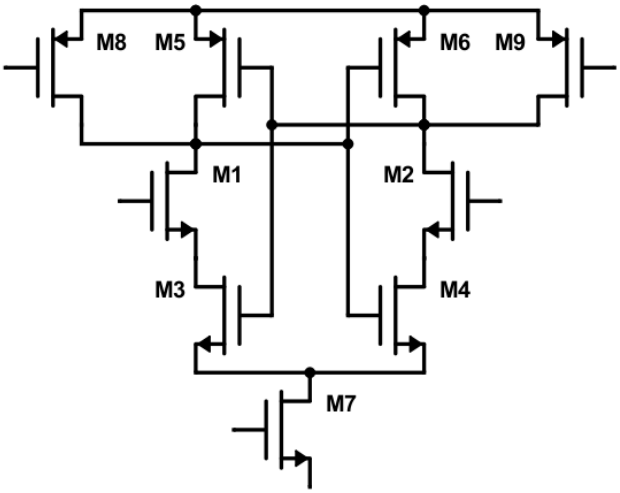


Figure 44: Schematic of the improved-StrongARM comparator

Overall, the main body of the improved-StrongARM comparator, transistors M1 through M7, must be optimized through simulations of the metastability and discharge time parameters.

Table 5 shows the final transistor sizes used for the improved-StrongARM comparator. It also indicates the optimum input common-mode voltage V_{CM}.

	M1-M2	M3-M4	M5-M6	M7	M8-M9
Type	$nmos$	$nmos$	$pmos$	$nmos$	$pmos$
No. of fingers	4	4	4	4	2
Finger width	$2.5\mu m$	$3.6\mu m$	$3.3\mu m$	$3.9\mu m$	$2\mu m$
Total width	$10\mu m$	$14.4\mu m$	$13.2\mu m$	$15.6\mu m$	$4\mu m$
Channel length	$60nm$	$60nm$	$60nm$	$60nm$	$60nm$
W/L	166.7	240	220	260	66.7
V_{CM}	$870mV$				

Table 5: Improved-StrongARM dimension and input biasing voltage

The graph in *Figure 45* is the resulting transient simulation of the designed comparator. In this simulation, the input differential voltage was set to $100\mu V$, the common-mode voltage is set to $870mV$, the sampling clock period corresponds to the $2GHz$ sampling frequency, and the rise-time of the sampling clock is set to $1ps$.

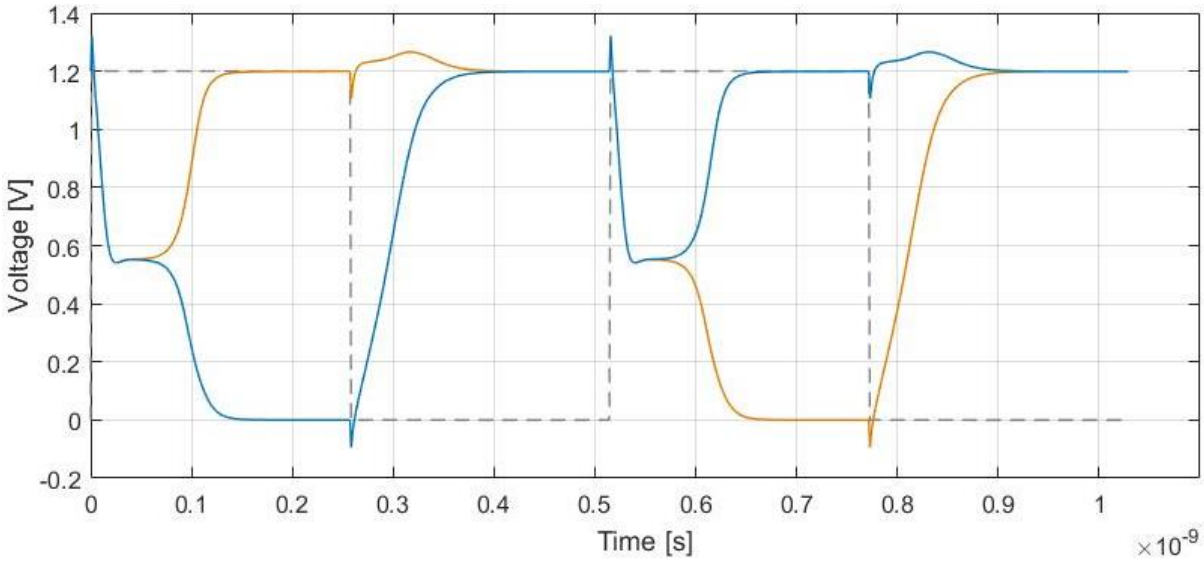


Figure 45: Transient simulation of the designed comparator

The simulation shows that τ_m is $8.44ps$, that τ_{dr} is $24.4ps$ and that time it takes for the output to reach 90% of absolute difference is $118ps$. Comparing the achieved parameters with the sampling clock period, shows that the derived comparator is good enough. In reality, the rise-time of the clock signal will be longer than then $1ps$. Due to this, further simulations where done to find the maximum risetime allowed in order for the comparator to settle within half the clock period. The simulations showed that maximum risetime is to be about $100ps$.

5.1.2 Preamplifier

As presented in *subsection 4.3.1*, the preamplifier will be realized with a resistor loaded differential amplifier where the substrates of the input transistors are separated and triple-welled.

The design goal for this amplifier is to achieve $12GHz$ bandwidth including the load of the comparator, and to bias the inputs of the comparator for $873mV$. However, the design should also try to keep the input capacitance low as possible so that the first-stage amplifier is loaded less.

5.1.2.1 Factors to considered for the differential amplifier

Here is a quick summary of the factors related to the differential amplifier:

- The gain of a differential amplifier is determined by the transistors transconductance and the total resistance on the outputs.
- The frequency response of the differential amplifier is limited by the unity gain frequency of the transistor and its load. Gain can be traded for bandwidth.
- The voltage over a passive drain resistor is proportional to the current passing through it. When the current is increased, then the resistor must become small. Attention must be given to this fact since a small resistor with large current can lead to high temperature generation, and thus potential melt downs.

- The Miller effect is directly related to the common-source gain. A larger gain will result in larger gate-to-drain capacitance.

5.1.2.2 Final design of the preamplifier

To achieve the best transistor efficiency the channel lengths are designed to be as short as possible. Transistor efficiency is here viewed as width-to-length ratio verses area. The result of this is that a higher ratio is achieved with the least amount of parasitic capacitance being added. Higher ration translates to higher gain.

By having the transistors narrower the parasitic capacitance is reduced and thus the Miller effect becomes negligible. Since there will be an amplifier stage in front of this one, with bandwidth extension, the largest gain contribution should stem from that amplifying stage.

The tail current source needs to be balanced with the differential pair in order to obtain the optimal gain and frequency response. Because of the strict voltage budget of $870m$, the current source biasing input and the input common-mode level are balanced to fit the budget.

Table 6 is in reference to *Figure 46* and lists the derived design and the operating voltages.

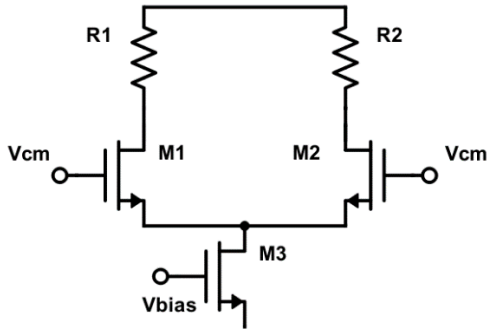


Figure 46: Schematic of the differential amplifier

	M1-M2	M3
Type	$nmos$	$nmos$
No. of fingers	4	2
Finger width	$2\mu m$	$1.9\mu m$
Total width	$8\mu m$	$3.8\mu m$
Channel length	$60nm$	$60nm$
W/L	133.3	63.3
V_{CM} or V_{bias}	$1V$	$665mV$

Table 6: Derived values for the elements of the preamplifier in Figure 46

The resistor type being used here *P+ poly without silicide*, and the resistance is equal to $1.16k\Omega$ for both. The output biasing levels achieved are at $873mV$, which is close to the desired V_{CM} by the comparator.

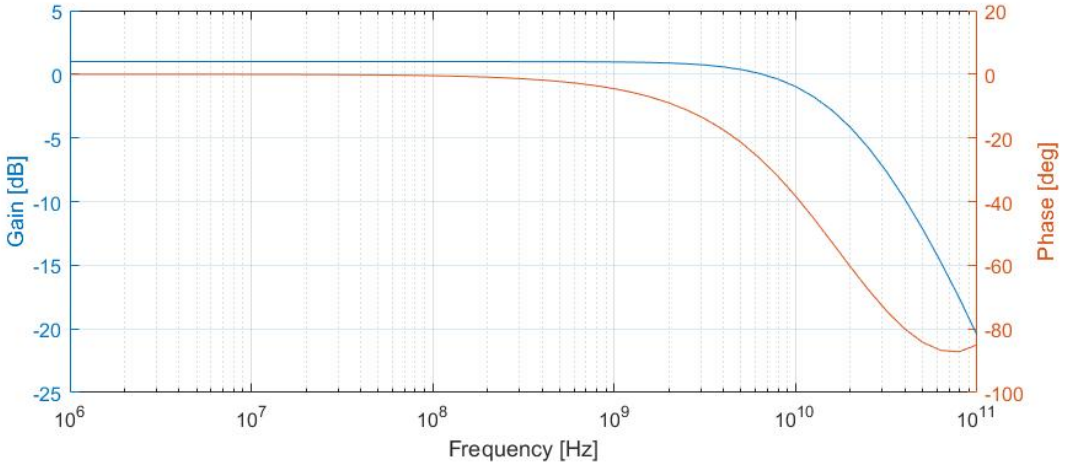


Figure 47: AC analysis of the preamplifier with the comparator connected

The simulated graph in *Figure 47* shows the frequency response of the differential amplifier with the improved-StrongARM comparator connected to the output. The graph indicates a $993mdB$ gain with the first pole appearing at about $13.5GHz$. Although the gain at f_{-3dB} is $-2dB$, the gain of the first stage amplifier will lift the total gain for the circuit.

5.1.3 First stage amplifier with bandwidth extending

For the first stage amplifier the same design factors apply as in *subsection 5.1.2.1* for the obvious reason that this is also a differential amplifier.

The only difference in this stage is the addition of the bandwidth extension elements. From the discussion in *subsection 2.2.4*, the correct technique to be used is determined by the parasitic node capacitance at the output of the amplifier compared to the load capacitance seen by the output.

5.1.3.1 The implemented first stage amplifier with bandwidth extension

The first stage amplifier is designed for bandwidth drive ability. What this means is that it should be able to supply the output capacitance with enough current for small signals to propagate to the next stages. As has been discussed, the current can be raised by expanding the $\frac{W}{L}$ of a transistor. For this reason, the transistors are designed to be quite wide.

When it comes to the bandwidth extension techniques, the shunt peaking technique was found to be the most effective bandwidth extender. The evaluation was done though implementing all the techniques and simulating their effects.

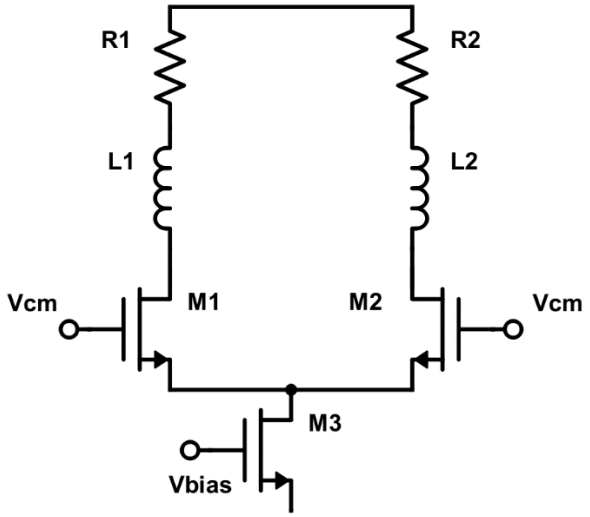


Figure 48: Schematic of the frst stage amplifier with shunt peaking

Figure 48 shows the schematic layout of the final first stage amplifier design, while *Table 7* lists the parameters of the design.

	M1-M2	M3	R1-R2	L1-L2
Type	$nmos$	$nmos$	$P_+ \ poly \ wo \ sili$	UTM
No. of fingers	18	10	1	
Finger width	$3.8\mu m$	$2.5\mu m$	$2\mu m$	
Total width	$68.4\mu m$	$25\mu m$	$2\mu m$	
Channel length	$90nm$	$60nm$	$12.96\mu m$	
W/L	760	416.7		
V_{CM} or V_{bias}	$1V$	$645mV$		
Impedance			100Ω	$2.65nH$

Table 7: Element design parameters of the first stage amplifier

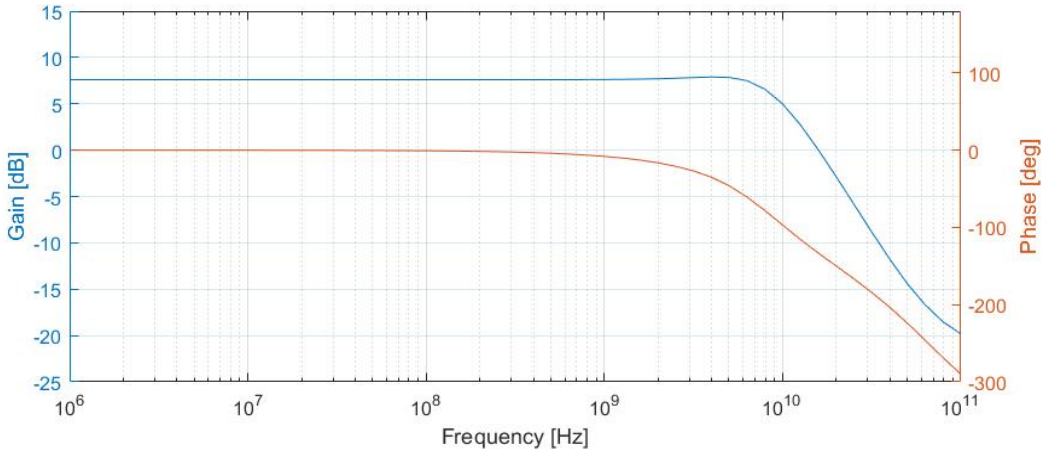


Figure 49: AC analysis of the first stage amplifier with 12 quantizers connected

Figure 49 shows the AC analysis of the final first stage amplifier design with load of 12 quantizers at the output. The analysis shows a gain of $7.597dB$ and a bandwidth of $10.4GHz$. However, the 90-degree phase shift is at about $9GHz$.

Even though the designed sampling circuit does not meet the gain and bandwidth specification, they are fairly close. This will however not affect the offset measurements that will be performed.

5.2 CALIBRATION DESIGN

As determined in *subsection 4.3.1*, the threshold voltage correction technique implemented with a current steering DAC, followed by two current mirror and two resistors. The design of these components will be presented in this section.

Before the design process is introduced, the correction range that is needed for the sampling circuit must be established. To find the required correction range, theoretical calculations and mismatch simulations were performed for the designed quantizer circuit.

To calculate the potential offset, *equations (2-1)* and *(2-18) were used.*

$$\sigma_{tot} = \sqrt{\sigma_{preamp}^2 + \frac{\sigma_{comp}}{A_{preamp}}^2} = \sqrt{(\frac{A_{VT_{preamp}}}{\sqrt{(WL)_{preamp}}})^2 + (\frac{\frac{A_{VT_{comp}}}{\sqrt{(WL)_{comp}}}}{A_{preamp}})^2}$$

(5-3)

Equation (5-3) shows the potential offset caused by mismatch of the threshold voltages in the input transistor pairs of the preamplifier and the comparator. Mismatch in threshold voltage is usually used to determine the offset because its parameters are of smallest area compared to other transistor parameters. Ideally, all parameters of a transistor should be included in the calculating, however, that would be a too complex equation to solve. To compensate for the lacking parameters, the correction circuit will be designed for six-times the deviation that is calculated and simulated. The parameter A_{VT} is about $4mV/\mu m$ for the 65-nm process. Solving with for the input transistor values of *Table 5* and *Table 6 with the gain of the preamplifier being* $993mdB$.

$$\sigma_{tot} = \sqrt{(\frac{4mV}{\sqrt{8 * 0.06}})^2 + (\frac{\frac{4mV}{\sqrt{10 * 0.06}}}{10^{\frac{993mdB}{20}}})^2} = 7.39mV$$

(5-4)

To determine the offset range, both sides of the mean offset must be represented, and the deviation multiplied by six, as discussed.

$$Offset\ range = 2 * 6\sigma_{tot} = 88.68mV$$

(5-5)

As for finding the offset range by simulation, Cadence is used. By performing Monte Carlo sampling with mismatch, the offsets can be defined. *Figure 50* is a histogram of the acquired simulation point fitted for normal distribution. MATLAB is once again used find the best fitted normal distribution and mean of the simulated data. A deviation of $6.45mV$ and a mean at $0.87mV$ was derived. For the design of 6σ, this equals to $77.4mV$ range.

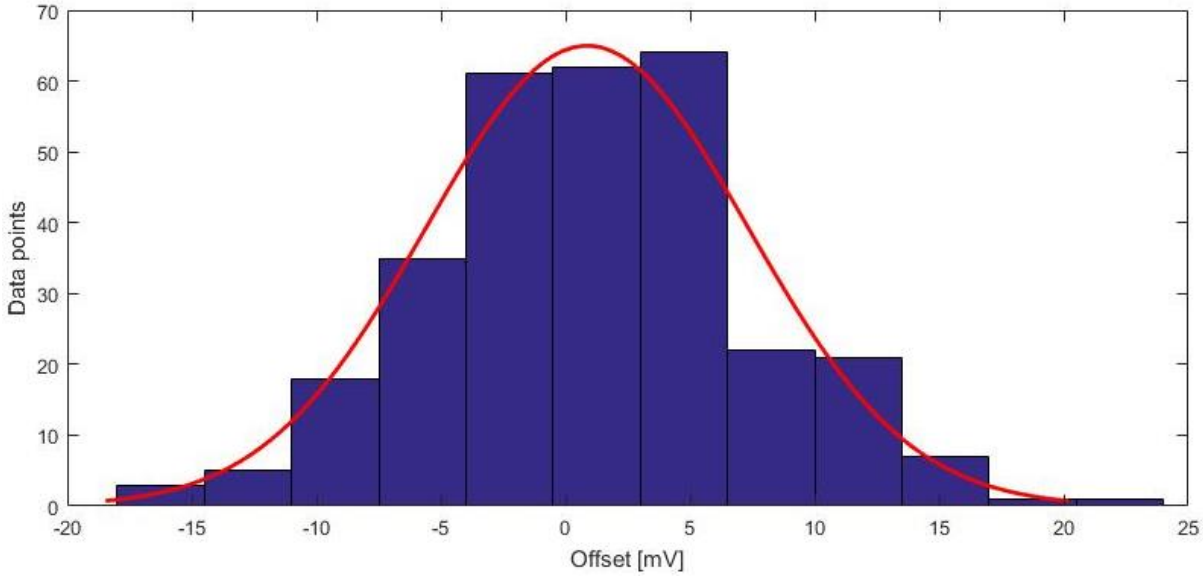


Figure 50: Histogram of the simulated offset with a fitted PDF

The two found offset ranges are slightly different. This is due to simplifications and assumptions made when they were derived. However, they give an idea of what offsets the sampling circuit may experience. Designing the correction circuit for a $90mV$ correction will satisfy both offset ranges.

5.2.1 DAC

To digitally control the substrate voltage for the threshold correction technique, a current steering DAC is implemented. The concept of current steering digital-to-analog converters is to supply them with a current and the DACs steer the current to one of their two outputs. How much is sent to which output depends on the digital code. The total current output by the DAC will be equal to the input current. *Figure 51* shows the 7-bit current steering M3M DAC used in the implementation.

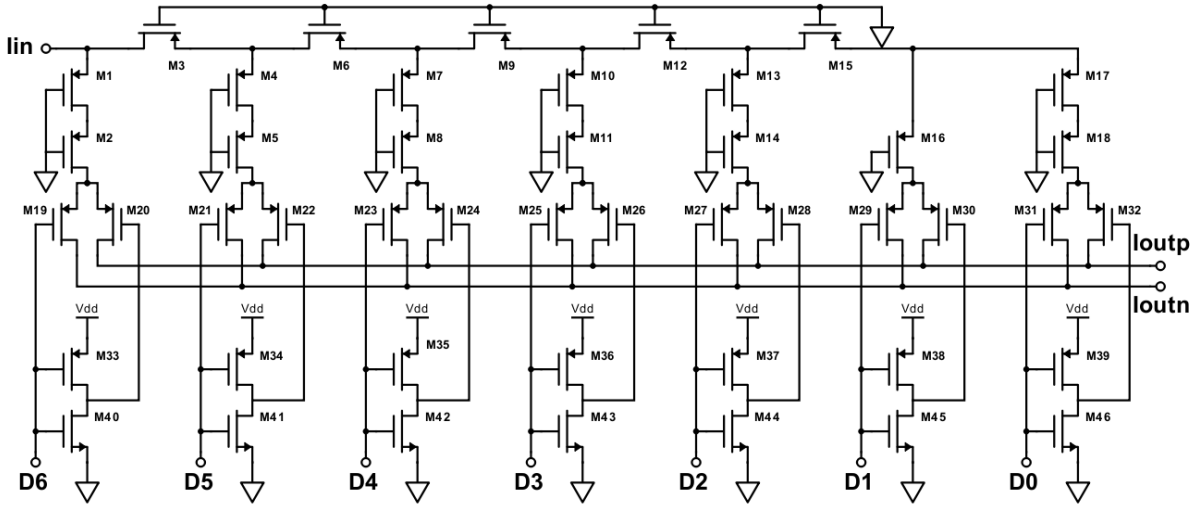


Figure 51: Schematic of a M3M current steering DAC

Here the current is supplied through I_{in} and divided amongst the branches. PMOS transistors M1 through M18 are all connected to ground at the gates for them to remain in saturation. For the current coming in, these transistors will represent resistances. At the first branch, the resistance

through M1 and M2, and eventually one of the switches M19 or M20, will be approximately equal to the rest of the circuit and thus splitting the current in half. The same will happen for the next branch where its parallel resistance will equal to its own. This division continues throughout the circuit. By doing this, the current in each branch can be viewed as binary weighted, meaning that the current in every branch is half of the current in the previous branch.

The two transistors at the bottom of every branch are used to switch the current to one of the two outputs. The digital inputs, $D0$ through $D6$, dictate which output the currents are to take. As an example, if all digital inputs are high, then all the currents are directed to I_{outp}. The inverter connected PMOS and NMOS transistors, M33 through M46, are there just to invert the input so that the switched pairs are opposite to each other.

The sizing of these transistors is easy. For the current division to operate as mentioned, all the current dividing transistors, M1 through M32, need to be equally sized. The transistors of an inverter need to be balanced for optimal operation since PMOS transistors have lower transconductance compared to the NMOS transistors. A rule of thumb is to design the PMOS transistor two or three times larger than the NMOS, to achieve good balance. The table BELOW lists the resistor sizes of the implementation.

	M1-M32	M33-M39	M40-M46
Type	$pmos$	$pmos$	$nmos$
Num. Fingers	1	1	1
Length	$60nm$	$60nm$	$60nm$
Width	$1.5\mu m$	$1.8\mu m$	$600nm$

Table 8: Design parameters for the M3M current DAC

One thing to note for this current DAC is that it is not perfectly linear. This however is not crucial for the application it is used for. As long as the resolution is good enough, a voltage correction level can be found to correct the offset.

5.2.2 Current-to-voltage translation

To convert the currents provided by the current DAC into voltages for the threshold correction, two current mirrors are implemented. The schematic of a realized current mirror is shown in *Figure 52*.

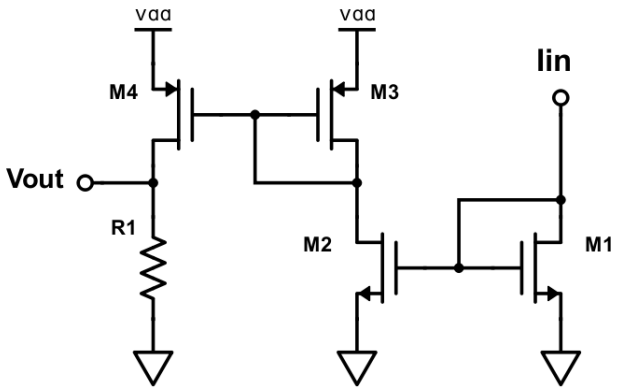


Figure 52: Schematic of the implemented current mirror

One of the output currents from the DAC, either I_{outp} or I_{outn}, is connected to the I_{in} of this circuit. This current is then mirrored to the M2-M3 branch and in turn mirrored to the branch with the resistor. The output voltage is determined by Ohms law.

Ideally the current going into the circuit is mirrored for every branch if the paired transistors are the same size. However, in reality this is not the case. In order to replicate the input current, as closely as possible some size balancing need to take please. For this implementation the balancing was done at the PMOS pair. *Table 9* lists the resulting design parameters for the transistors and the resistor.

	M1-M2	M3	M4	R1
Type	$nmos$	$pmos$	$pmos$	
Num. Fingers	2	2	2	
Length	$60nm$	$60nm$	$60nm$	
Width	$1\mu m$	$800nm$	$700nm$	
Total width	$2\mu m$	$1.6\mu m$	$1.4\mu m$	

Table 9: Design parameters of the current mirror

It is worth noting that the output voltage of the of the output resistor is not linear to the input current. The reason for this is that the transistors are operating in the triode region when the currents are small. *Figure 53* shows the output voltage of this circuit on behalf of the input current.

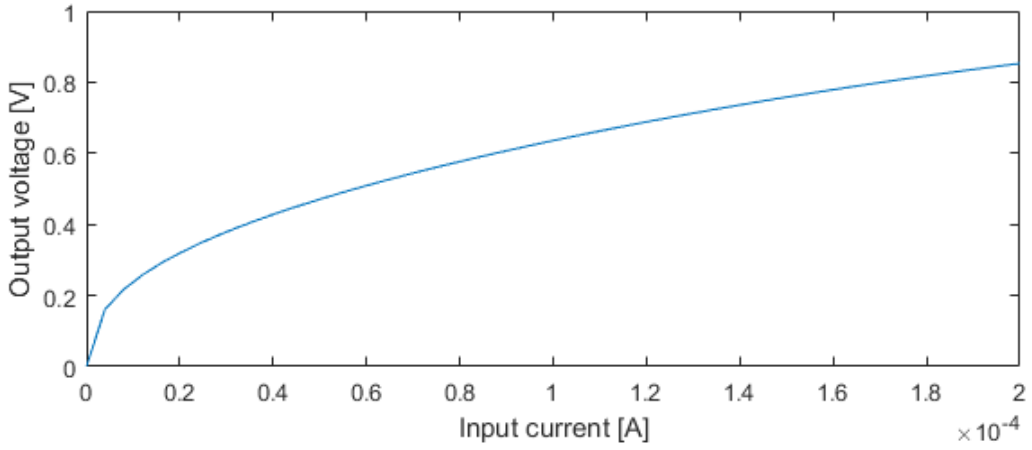


Figure 53: Simulation of the output voltage current-to-voltage translation circuit.

The graph shows the nonlinearity of the output voltage at low input currents. The voltage does, however, become more linear with the increase in current.

5.2.3 Correction circuit verification

To verify the designed correction circuit and the correction effect it has on the sampling circuit, simulations of the correction circuit and the offset correction of the quantizes are performed in this subsection.

The output voltages of the offset correction circuit are presented in *Figure 54*. The graph shows that the output voltages scale between about $100mV$ to about $860mV$. This simulation was performed with a $100\mu A$ biasing current.

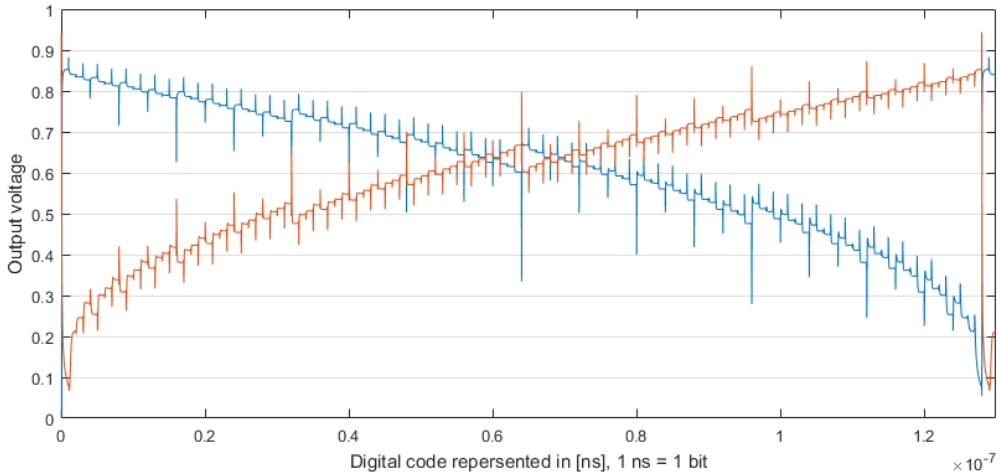


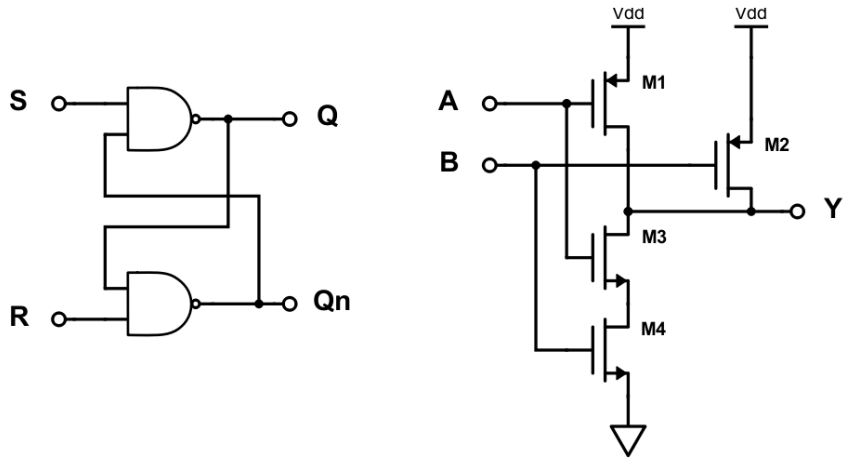
Figure 54: Simulations of the two outputs of the correction circuit with 100µA bias current

5.3 DIGITAL DESIGN

The implementations presented thus far need some additional support components to be able to function properly. The components here are common circuits and will not be discussed in much detail.

5.3.1.1 Output latch

An additional latch is implemented to the output of the comparator to hold on to the sampled digital value while the comparator is in reset mode. Without this latch, the outputs would both go $high$ due to the precharge operation done at the output nodes during the reset phase. The result of not having a latched output is a much shorter time window for "reading" the converted value.



S	R	Q	Qn
0	0	Invalid	
0	1	1	0
1	0	0	1
1	1	Unchanged	

Figure 55: Left, Block diagram of NAND gate SR latch - Middle, Schematic of NAND gate – Right, Truth table of SR latch

The latch used in the implementation is a SR latch, designed with two NAND gates. From the truth table in *Figure 55*, the outputs can be determined. When the two inputs, S and R, are high, then the output values of the previous state are held. This means that the outputs, Q and Qn, only change if

the inputs are inverse of each other. An invalid state occurs if the inputs are both low, however, the comparator being implemented in this project will never achieve this output.

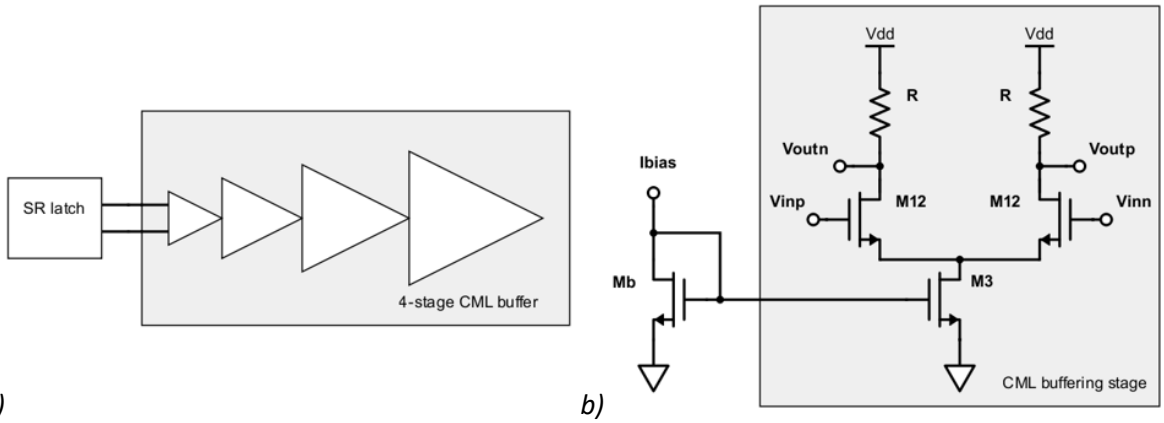
The NAND gate transistors are all minimum size. Widths are $200nm$ with one finger, and channel lengths are $60nm$. Having the transistors as small as possible reduces the parasitic capacitance being added to improved-StrongARM output nodes.

5.3.1.2 Output buffer

Now, since the output latch of the previous subsection is very small, it is not able to drive large loads. The loads the latch will needs to drive would be the interconnects to the frame pads of the chip, and the external components or devices needed to *record* the output values. The small current provided by the latch would need extensive time to reach the digital levels when changing between them. Using a sampling frequency of $2GHz$ would result in incorrect output levels.

NOTE. During the design phase of this project, neither the pad frame of the chip nor the parasitic extraction tool was available. For this reason, an assumption of the capacitive load was made to be 1pF. The assumption included capacitance form interconnects, pad frame, bonding wires, and external components.

To achieve an output response matching the sampling frequency, considering the output loads, a four-stage buffer will be implemented. *Figure 56* illustrates the buffer.



a) *b)*

Figure 56: a) Block diagram of the 4-stage CML buffer, b) Schematic of a CML buffering stage

Here is the four-stage buffer implemented on the output. It is designed with four cascaded common-mode logic buffers. Each stage uses the same differential amplifier architecture but with different sized transistors. By gradually increasing the sizes of the transistors with every step, the current is also increasing in every stage.

	Stage 1		
	M12	M3	R
Fingers	4	8	
Length	$60nm$	$60nm$	$9.1\mu m$
Width	$400nm$	$400nm$	$2\mu m$
Total width	$1.6\mu m$	$3.2\mu m$	
W/L ratio	26.7	53.3	
Current		$260\mu A$	
Resistance			$3.2k$

Table 10: Design parameters for the first stage of the 4-stageCML buffer

The table list the transistor and resistor values for the first stage, as well as the total current going through the buffer stage. The following stages are designed to increase the current by a factor of 4 for every stage. To achieve that, the widths and number of fingers of every transistor is doubled resulting in the wanted factor of 4. The resistors also need to be adjusted by dividing the total resistance by the same factor.

Figure BELOW shows the transient of the buffer with the SR latch as the digital source and $1pF$ loads at the outputs. The frequency of the alternating digital input is $1GHz$ which is equivalent to the output frequency of the SR latch.

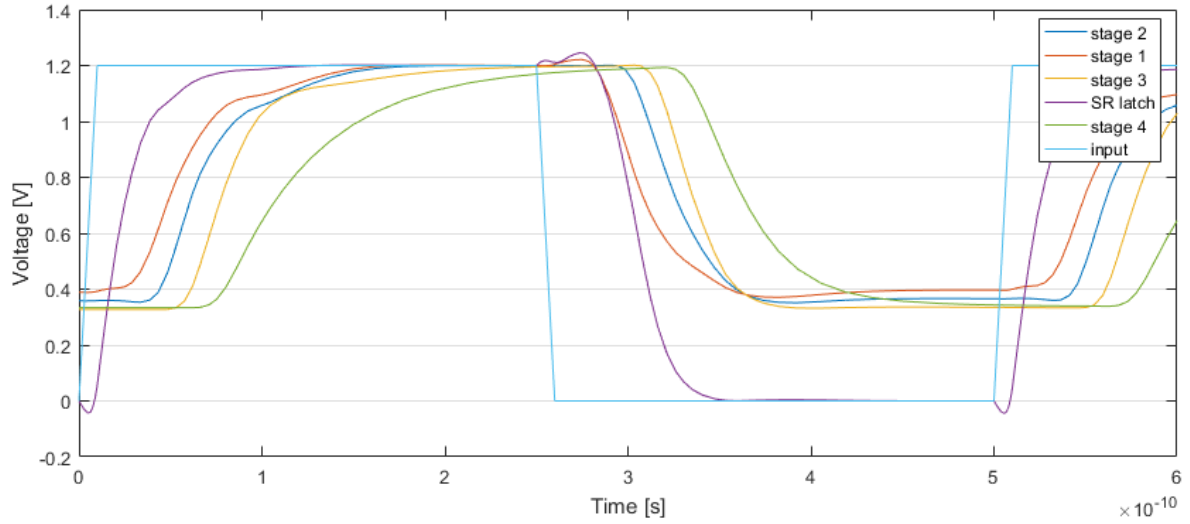


Figure 57: Transient analysis of the output stages of the implemented circuit

The analysis shows three things. Firstly, the propagation delay is about $150ps$ from the input signal to the output of fourth stage. This delay will be shorter if a smaller load is driven. Secondly, the output voltage uses about $250ps$ to reach V_{DD}. A large load would extend the settling time. And thirdly, the output voltage has a lower limit at about $350mV$, due to the drain-source voltage of the current source transistor M3. This needs to be considered when extraction the sampled data.

5.3.2 Input shift register

The DAC used to control the threshold correction voltage, needs to be digitally programmed. Programming the DAC directly by using external devices would require multiple connections out of the chip. For the DAC being implemented here, seven parallel lines would be required. To reduce the amount of connections, the clocked shift register in *Figure 58* is being implemented.

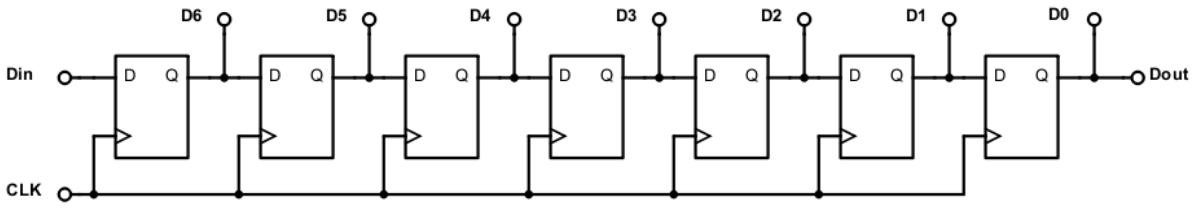


Figure 58: Block diagram of the shift register

A shift register is commonly used to translate serial data to parallel data. By shifting in a binary code, every bit can be extracted in parallel. A clock signal is used to propagate the serial data through the register. In order to transfer at binary value from D_{in} to D_0, the register needs to be clocked seven times. The shift register in this implementation will also have a data output, D_{out}, to add the possibility to check if that being sent in is correct.

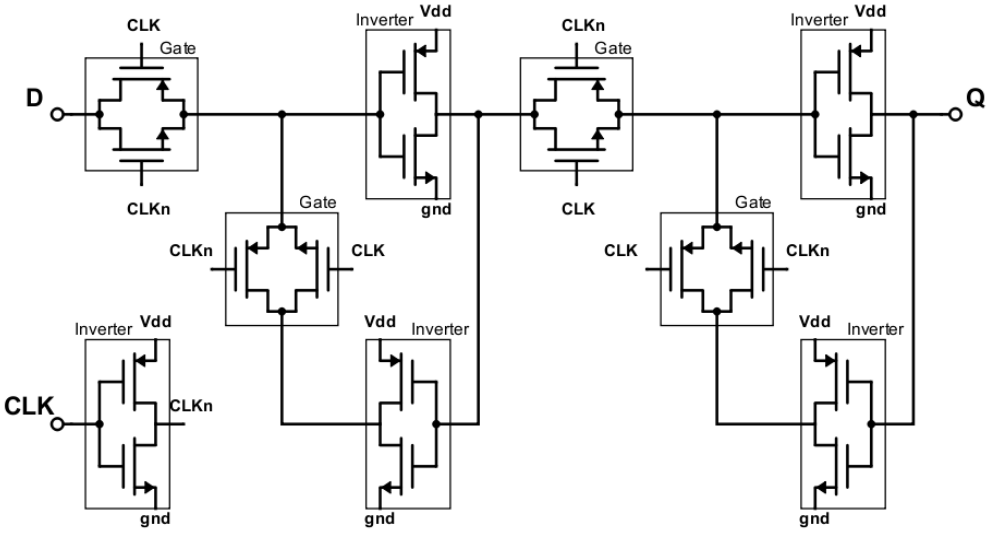


Figure 59: Schematic of the D flip flop

The D flipflops are designed with gates and inverters as shown in the *Figure 59***.** When the CLK goes low, the digital value on the input, D, is sent through the gate G_1 and inverted twice by the inverter loop INV_1 and INV_2. However, the G_2 and G_3 are not letting the digital value through because they are closed. Then, when the CLK goes high, the G_2 and G_3 open while G_1 closes. This allows the inverted value at the first loop output to propagate to the next loop where it is inverted to the original value by INV_3. The purpose of G_2 and G_4 is to block the previous value from overwriting the new value. This will happen if the previous value is high and the new value is low. Without the closed path, the high value is sensed by the main inverter and thus locking it in an infinite loop.

The shift register here is implemented with minimum widths ($200nm$) and lengths ($60nm$) due to there not being any requirements of speed for the correction operation.

5.4 BLOCK DIAGRAM OF THE FULL IMPLEMENTATION

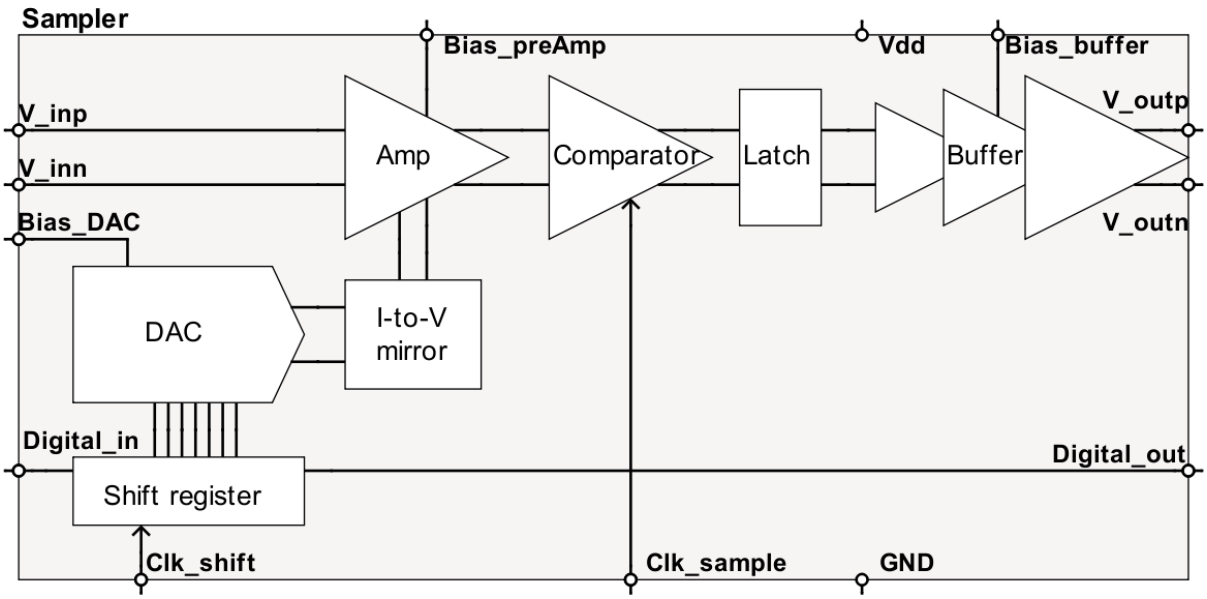


Figure 60: Block diagram of the circuit being fabrecated

Here is a quick summary of the blocks of *Figure 60*:

- *Amp* is the differential amplifier that is used to correct its own offset and the offset of the comparator. The $Bias_{preAmp}$ terminal is the current input to bias the current source transistor while the bottom terminals are for the complementary voltages for substrate correction.
- *Comparator* is the improved-StrongARM latch that decides which input is larger. It has a clock input for triggering the sample.
- *Latch* is a latch that keeps the sampled output while the comparator is resetting for the next sample.
- *Buffer* is a four-stage buffer that amplifies the output currents of the latch in order to be able to drive larger output loads.
- *Shift register* is a register that stores the binary values for the programming of the DAC. The data is shifted serially though the register by a clock signal at the Clk_{shift} terminal. Binary data is taken out in a parallel fashion to set the DAC. This register also has a serial output that can be used for verifying the data.
- *DAC* is a digital-to-analog converter that steers the current from the $Bias_{DAC}$ terminal to the two outputs according to the binary value of the shift register.
- *I->V* converts the two currents out of the DAC to voltages that control the substrates voltages of the Amp.

5.5 LAYOUT

In this section the layout approach will be discussed.

The most important part of the layout implementation is the analog part of the circuit. This is because the sampler is the most sensitive component to noise and variation. To minimize the mismatch of the whole analog part, the paired transistors are places as close to each other as possible. As Pelgrom's *equation (2-1)* dictates, mismatch is affected by the spacing of the matched parameters. So, by placing every matched device as close as possible, the potential mismatch is less.

In fact, the best way of ensuring better matching is to use the common centroid layout method. However, prebuilt RF transistors are used in the implementation hindering the use of the common centroid method. These RF transistors are design with an additional well which is used for noise isolation in radio frequency applications. The use of individual wells for the transistors adds another drawback where every transistor needs a larger area and in turn leads to larger distances between transistors.

Symmetry and equal surroundings were also considered for the analog circuit. This was done to counteract two-dimensional, proximity and oriental effects as best as possible. Another gain here is that parasitic coupling noise would ideally affect the two "sides" equally if the surroundings are the matched.

Finally, the analog circuit part is placed as close to the pad-frame as possible. This to minimize the travel distance of the RF signal. The result is less parasitic affecting the signal before it is sampled. Minimum signal traveling distance is also used in the analog circuit itself. for the same reasons.

Usually, in mixed-signal circuits, the analog and digital parts are separated. This separation tries to isolate the crosstalk, provided by the digital operations, from the analog sensitive circuitry. Digital processing does introduce a lot of noise because of the voltage levels within switching between the supply voltage and ground. Although this separation is implemented in this layout, the coupling noise can be neglected here due to the digital circuit not being active during the signal sampling.

The digital correction circuit is laid out as it is for practical reasons. The shift register is placed such that outputs are aligned with the inputs of the M3M DAC, and the DAC is placed to have an easy access to the current-voltage translators. Clock and data lines from the pad-frame and to the digital circuit are not critical and thus can be designed longer. This is due to relaxed speed requirements for the programming of the correction circuit.

The output latch is the only digital circuit that has some requirements for its placement. Since the latch is directly connected to the output of the quantizer, it needs to be implemented close to the output. This is due to the interconnect adding parasitics to the output nodes. Having long lines increases the resistance and the capacitance, which in turn can affect the quantizer for the worse. So, placing the latch as close as possible will reduce the affects. For these reasons, the latch is placed inside the analog circuit.

The output buffering circuit is placed adjacent to the output latch. Because the latch is design as small as it is, it will not be able to drive large loads. Here again the length of the interconnects comes into play by adding more load with length. Therefore, the inputs of the output buffer are placed close to the output latch. In addition, the output buffer needs large currents compared to the rest of the circuits and therefor has its own supply pads. Separating the supply of the output buffer and the rest will reduce supply noise for the analog circuit. Usually this method should also be implemented with regards to digital and analog circuit. But again, the digital circuit is static during the analog operations.

5.5.1 Post layout simulations

The post layout simulations are crucial indicators of how the circuits are going to operate once they have been manufactured. Cadence uses a parasitic extraction tool to calculate parasitics between metal layers and parasitics of these layers. This is a powerful tool that can be used to simulate the circuits as they would behave after the integrated circuits are produced. Of course, these extractions are not completely accurate, but they give a realistic representation of the end-product.

Knowing the effects, the layout introduces to the circuits, allows the designer to modify the circuit if it does not meet the requirements and to predict how other devices will interact with the device. Unfortunately, the parasitic extraction tool was not operational during the design phase of this project. The reason being that some of the extraction files used by the tool were not set up properly and were not fixed prior to the handover for production. This means that the implementation of this thesis was not verified through these simulations, and that the testbench design was delayed until the problem was fixed. That being said, the following simulations were done once the tool was in place.

Furthermore, the layout was only done for the quantizer and thus no post layout simulation was performed for the full sampling circuit.

5.5.1.1 Frequency response

Figure 61 shows the frequency and phase response of the quantizer circuit with the offset correction circuit also implemented. Also, the correction circuit is set to supply each of the current-to-source circuits with $50\mu A$. From this simulation, the first pole, or f_{-3dB}, was found to be at $9.92GHz$ and the gain to be of $964mdB$. Comparing these post layout simulation results to the circuit simulation result in *subsection 5.1.2.2*, indicated that the bandwidth has retracted by $3.58GHz$ and that the gain has fallen by $29mdB$.

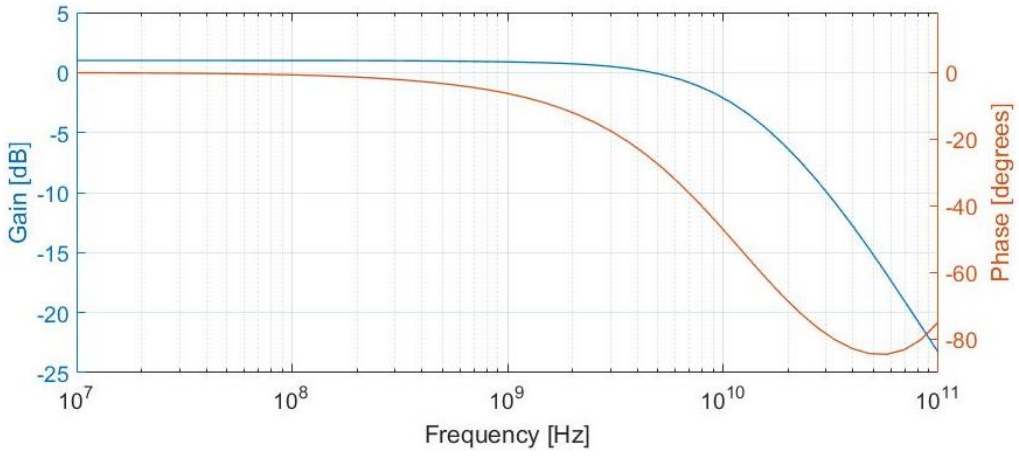


Figure 61: Post layout AC analysis of the quantizer circuit with offset correction

An additional post layout simulation was performed with the offset correction circuit being disconnected and the bulk terminals connected to ground. The results form that simulation yields a bandwidth of $10.08GHz$ and a gain of $996mdB$. Correlating the simulated results of the three discussed simulations, dictated that the extracted parasitics are responsible for $3.42GHz$ of the bandwidth retraction while the offset correction circuit only contributes $160MHz$ to the total

retraction. In the case of gain, the paracitics do actually add $3mdB$ gain and thus the correction circuit is responsible for $32mdB$ loss in gain.

5.5.1.2 Sampling rate

Figure 62 is the post layout transient simulations of the different device outputs of the implemented sampling circuit. The clock period is set to $500ps$ to represent the $2GHz$ sampling rate, and the differential input for the sampler was set to $1mV$.

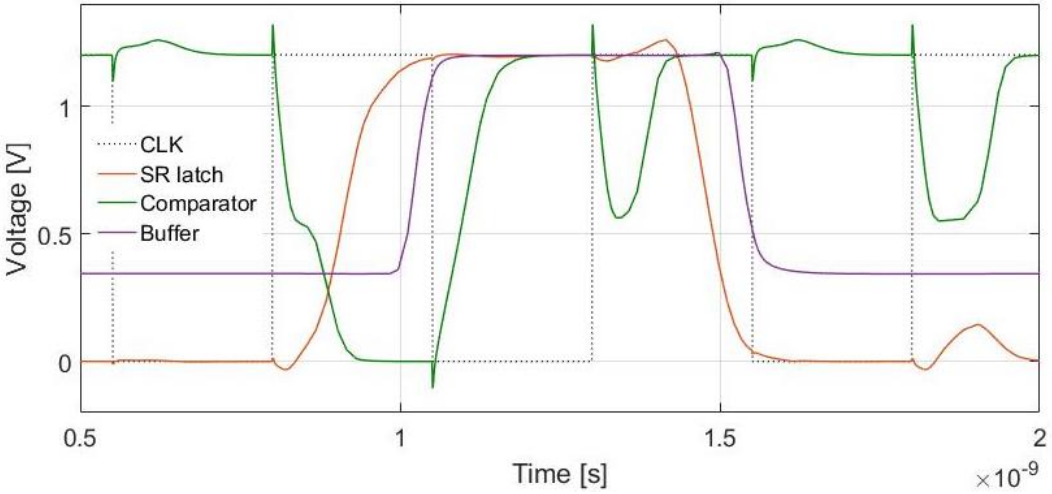


Figure 62: Post layout transient simulation of component outputs of integrated circuit

The simulations show that all the outputs are settling within the sampling period which means that the sampling rate requirements is satisfied. There is however some propagation delay in the circuit, but that is to be expected and it will not affect the sampling circuit.

5.5.1.3 Correction circuit output

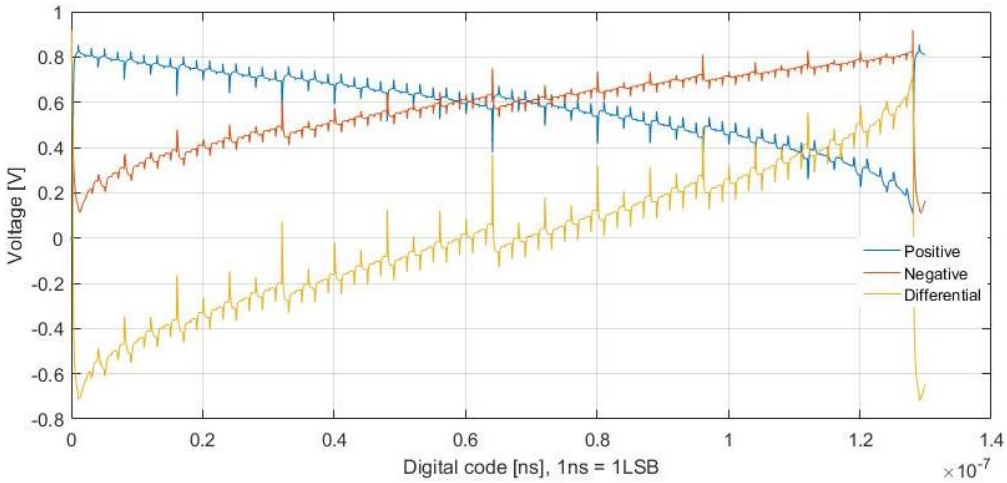


Figure 63: Post layout simulation of the correction circuit outputs.

Figure 63 shows the results of the post layout simulation performed for the correction circuit output voltage. The results match the circuit simulation result of *section 5.2.3* meaning that implementation did not affect the characteristics of this circuit.

5.5.2 Layout for manufacturing

The designed layout is to be fabricated in a *Quad-flat no-leads* package with 80 leads (QFN 80). This means that there are 80 leads going in/out of the chip, 20 for each of the four sides. Out of these 80 leads, 8 are used to power that lead-frame (2 for each side) leaving 72 leads in total to be used for implementation connections.

6 MEASUREMENTS

The fabricated chip, with the implementation of the swept threshold sampler and the offset correction circuit, is tested in this chapter.

The first section will present the design of the testbench needed to perform the measurements. *Section 6.2* will present measurements are conducted and how the results are acquired. Finally, the results will be presented in the last section.

Before, the testbench can be design the measurements parameters relevant to this master project need to the defined:

1. Offset prior to correction. It is of interest to find the offset of the sampling circuit without the influence of the correction. The findings here will highlight the offset caused by the correction technique.
2. Correction range. In order to verify the correction effectiveness of the implemented technique, the correction range must be determined.
3. Post correction offset. This measurement parameter will establish how well the offset can be corrected and the resolution of the correction technique.
4. Noise. It is of interest to find the noise of the sampling circuit in order to determine how much noise the correction circuit introduces.

6.1 DESIGN OF TESTBENCH

To measure the offsets and the impact of the correction circuit, the testbench will be set up similar to the testbench setup of the Cadence simulations.

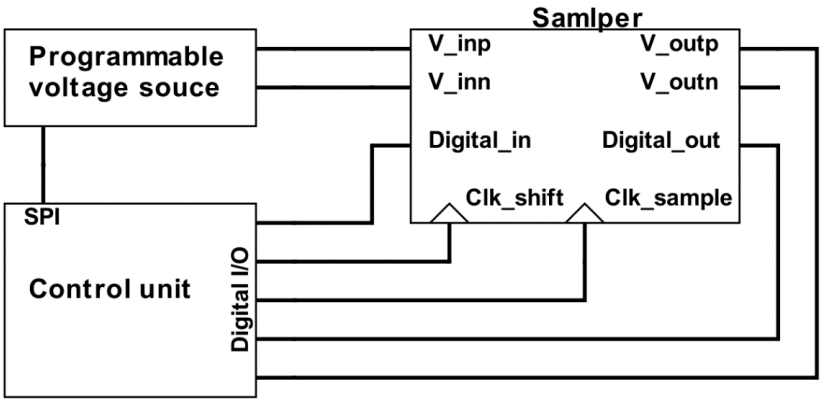


Figure 64: Block diagram of the test setup

Figure 64 shows a block diagram of the devices and connections needed for perform the measurements. Following subsections will explain the setup and functions of the two support units, programmable voltage source and control unit, and the PCB needed for mounting the chip.

6.1.1 External programable voltage source

The function of the programmable voltage source is to sweep voltages at the two differential inputs of the chip. The output voltages of the source need to ramp through a range of $400mV$, with inverse starting points, and have a common-mode voltage at $1V$. In other words, one of the outputs will start at $800mV$ and step up to $1.2V$ while the other output starts at $1.2V$ and steps down to $800mV$. The stepped outputs will cross each other at the common-mode voltage which is where the sampler ideally would change its output state.

This voltage source also needs to be controlled in order to correlate the sampled value out of the chip with the threshold levels on the inputs of the chip. To achieve this, a control unit must program every step of the range whilst also reading the output of the sampler in between every step. Since a higher sweep resolution requires a higher number of bits, a serial programming interface is a more practical option for programming the source than a parallel interface is. For this reason, a serial interface such as *I2C*/*TWI* or *SPI* can be used.

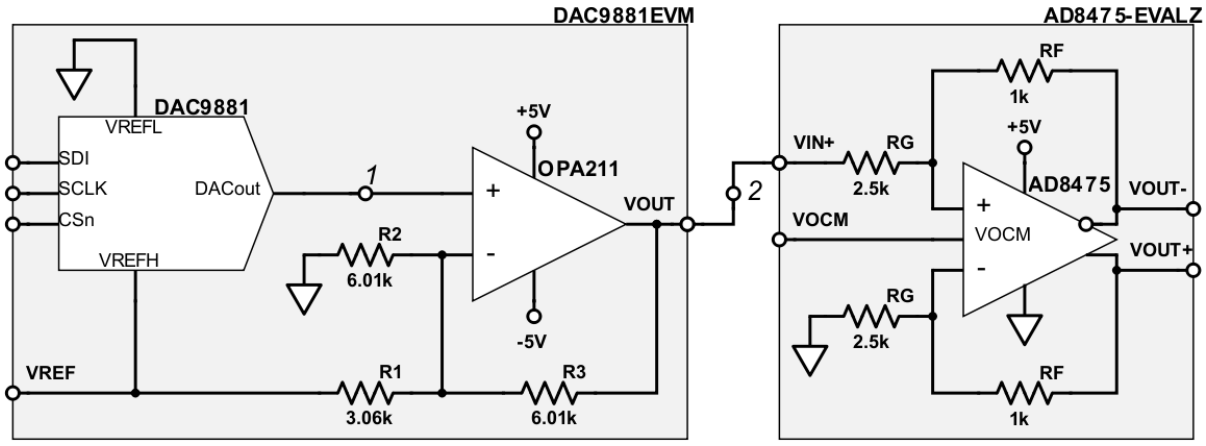


Figure 65: Block diagram of the programable voltage source

Figure 65 shows a block diagram of the programmable voltage setup for this test setup. Here two evaluation boards, *DAC9881EVM* and *AD8475-EVALZ*, are used to achieve the wanted sweep voltages [25] [26].

DAC9881EVM is designed for the DAC9881 digital-to-analog converter, with an operational amplifier at its output. DAC9881 is an 18-bit, single-channel, voltage-output DAC with SPI interface [27]. It is designed with an R-2R ladder architecture and uses two reference inputs, V_{REF_H} and V_{REF_L}, to determine the range of the output voltage. The evaluation board is built to accommodate various test setups. For this test the board is configured to operate bipolar as *Figure 65* illustrates.

$$V_{OUT}(CODE) = \left(V_{REF}\frac{CODE}{2^{18}}\right)\left(1+\frac{R_3}{R_2}+\frac{R_3}{R_1}\right)-\left(V_{REF}\frac{R_3}{R_1}\right) \quad (6\text{-}1)$$

The *equation (6-1)* represents the output voltage of the bipolar configuration [27]. $CODE$ is the binary value the DAC is programmed with to and V_{REF} is the same as V_{REF_H} due to V_{REF_L} being grounded for this operation.

AD8475-EVALZ is an evaluation board consisting of the fully differential, attenuating amplifier AD8475 [26]. It reduces the input differential voltage by $0.4x$ or $0.8x$, depending on which input pins are used, and it has a terminal, V_{OCM}, for setting the output common-mode level. The AD8475 has

integrated gate and feedback resistors, R_G and R_F respectively, which determine the gain of the amplifier. *Equation (6-2)* is the function of the voltage transfer.

$$V_{OUT_{dm}} = V_{IN_{dm}} \frac{R_F}{R_G}$$

(6-2)

The two boards, with the setup and connection of *Figure 65*, satisfy the required programmable voltage source needed for the measurements. DAC9881 converts the binary value provided through the serial interface SPI to an analog voltage. This analog voltage is then amplified and shifted down to operate bipolarly as is required by the next stage. The AD8475 takes the single-ended voltage and converts it to an attenuated differential voltage biased around the set output common-mode level.

The output common-mode here needs to match the input common-mode of the implemented sampler. For this reason, V_{OCM} is set to $1V$. Now, to find the reference voltage V_{REF} needed to achieve the wanted sweep range of $400mV$, one can combined *equation (6-1)* and *(6-2)*, where $V_{IN_{dm}} = (V_{OUT}(CODE) - 0V)$, and solve for V_{REF}.

$$V_{REF} = \frac{V_{OUT}(CODE)}{\left(\frac{CODE}{2^{18}}\right)\left(1 + \frac{R_3}{R_2} + \frac{R_3}{R_1}\right) - \left(\frac{R_3}{R_1}\right)} = \frac{V_{OUT_{dm}} \frac{R_G}{R_F}}{\left(\frac{CODE}{2^{18}}\right)\left(1 + \frac{R_3}{R_2} + \frac{R_3}{R_1}\right) - \left(\frac{R_3}{R_1}\right)}$$

(6-3)

To get the wanted $V_{OUT_{dm}}$, $CODE$ must be set to either 0 or 2^{18}. Technically $CODE$ will only go up to $(2^{18} - 1)$, however V_{REF} is the ideal reference voltage and thus serves as a theoretical guide value which will be adjusted for the measurements. For simplicity, $\frac{R_3}{R_1}$ is also viewed as 2. With $CODE$ chosen as 2^{18}, V_{REF} is calculated.

$$V_{REF} = \frac{400mV * 2.5}{(1)(1 + 1 + 2) - (2)} = 500mV$$

(6-4)

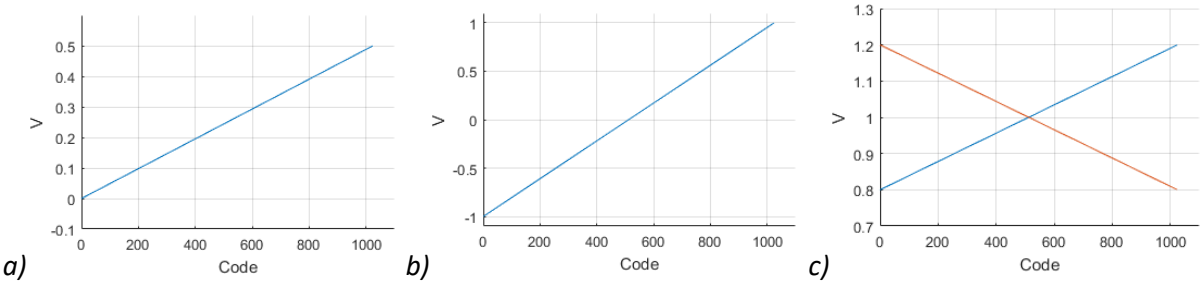


Figure 66: Ideal node voltage on behalf of programmed code value for the nodes of Figure 65, a) 1, b) 2, c) 3

The graphs of *Figure 66* illustrate the ideal voltages at the outputs of the three main components of the programmable voltage source, on behalf of the binary values the voltages are converted from. Graph *a)* of *Figure 66* illustrates the voltage at node *1* in the block diagram of *Figure 65*, *b)* illustrates node 2 and *c)* illustrates the resulting differential output of the source. Note that the code here only goes to 1023 which is only a 10-bit range. This is just to simplify the illustration. The resolution wanted resolution can be set by the control. For the measurements in this project both 10-bit and 18-bit resolution will be used.

6.1.2 Control unit – Arduino Leonardo

The control unit used for this test setup is an Arduino Leonardo [28]. It is built up around an ATmega32u4 microcontroller and has 20 input/output-digital ports, as well an USB interface to communicate with a computer. The Arduino microcontroller board was chosen because it is easy and quick to set up, and because there is plenty of easily accessible literature and example codes on the internet. In addition, the Arduino IDE has a built-in serial monitor that communicates with the Leonardo while it is operating. This is practical because it allows for direct interaction between the computer and the control unit.

The connections needed for the Leonardo to control the implemented sampling circuit are as follow:

- *Clk_sample*, a digital output port for triggering the implemented sampler to sample.
- *Digital_out*, a digital input port for reading the digital output of the sampler.
- *Clk_shift*, a digital output port for shifting a binary value through the shift register in the chip.
- *Digital_in*, a digital output port for the binary values being shifted into the shift register.

For controlling the DAC9881EVM, these SPI connections need to be supported:

- *SCLK*, a digital output port for clocking the serial data.
- *CSn*, a digital output port for enabling the interface of the DAC9881.
- *SDI*, a digital output port for the data that is being transferred to the DAC9881.
- *SDO (optional)*, a digital input port for receiving serial data form the DAC9881.

More information about the SPI interface and timing diagram for the DAC9881 can be viewed in [27].

In order to automate the measurements needed to find the parameters of interest, the control unit is programmed to operate the following four functions:

- *swp*, programs the DAC9881 to sweeps the input of the sampler with a 10-bit resolution, whilst also reading the sampled output of the sampler. The function is implemented such that it can perform multiple samples for every step of the sweep and output the average value of that step, also known as oversampling. This is done to average the noise of the circuit which is affecting the decision making of the sampler. The number of samples per step is set to be 1000 in this setup.
- *adj*, programs the correction circuit of the sampler. This function is used to adjust the correction level of the correction circuit implemented on the chip.
- *all*, is the function that performs the full correction test at 10-bit resolution. Here the entire correction range is stepped with *adj* while the *swp* function is performed for every correction step. The output results show the range and resolution of the correction implementation
- *inv*, is the same function as *swp* but with 18-bit resolution. This function will be used to investigate the noise of the sampler.
- *set*, programs the programable voltage source only. It is used to manually set the output voltages of the source in order to inspect and verify the operations of the voltage source.

The full code for the control unit is attached in Appendix B

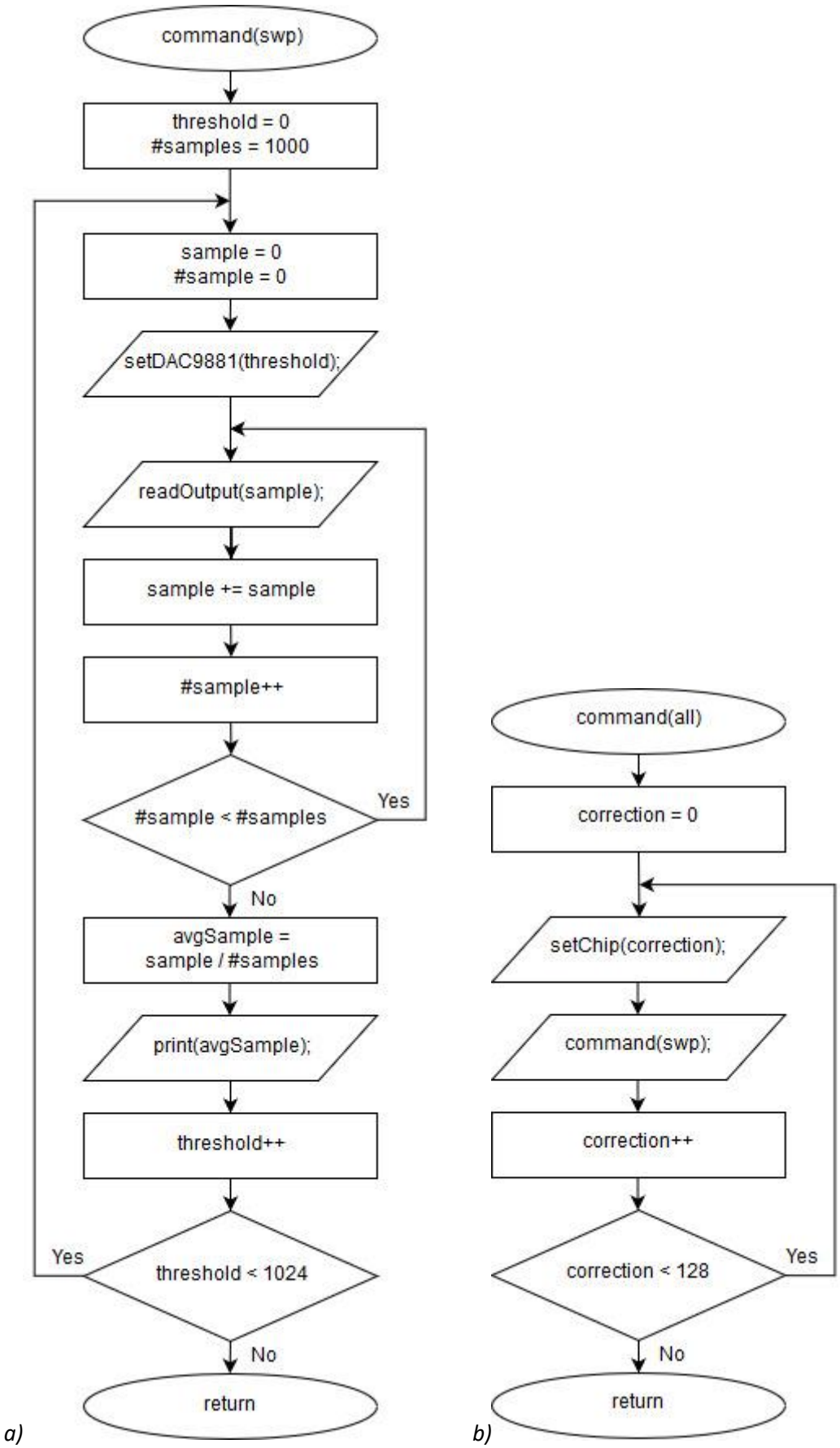


Figure 67: Flow chart for functions a) swp, and b) all

Figure 67 shows the flow chart for the functions, *swp* and *all*. What is worth noting here is that the maximum correction value is 127, which corresponds to the 7-bit current DAC implemented in the correction circuit, and that the maximum threshold value is 1023, which again represents the 10-bit range. The flow chart of function *inv* is not shown because it is almost identical to *swp*, just with higher resolution and with selectable start and end points.

6.1.3 PCB

Now, to test the implemented sampler, a *printed circuit board* (PCB) had to be produced to access the manufactured chip. As mentioned earlier, the packaging of the chip is of the type *Quad-flat no-leads* with 80 leads and a single chip contains four identical sampler circuits. Because the samplers on each side are identical, the following PCB design will be presented for a single sampler. The produced PCB will however be set up equally for all sides with minor path and terminal differences.

6.1.3.1 Chip access

Normally, a chip is soldered on to the PCB for connection to other devices. For this PCB design however, a QFN-socket is used to connect the chip to the PCB. By using a socket instead of soldering the chip directly on to the PCB some practical gains are achieved. Firstly, soldering a QFN 80 chip with 0.5mm pitch is very difficult compared to soldering a socket with through-hole pins. And secondly, using a socket allows for chips to be swapped without re-soldering chips or having to produce a PCB for every chip. With the use of a socket, multiple chips can be tested with same PCB.

There is a drawback with using sockets. Sockets such as the one used here extend the signal paths to the chip. This leads to added inductance, resistance and possibly noise signal paths for the. If the setup would have been for RF-testing, then the sockets could not have been used. But, since the tests for this thesis only evolve around DC levels and offsets, using sockets is acceptable.

6.1.3.2 Biasing circuits

The implemented sampler needs to be supplied by three bias currents to operate. As the diagram of *Figure 68* shows, one is for the preamplifier of $225\mu A$, one is for the current-DAC of $100\mu A$, and the final one is for the output buffer of $350\mu A$.

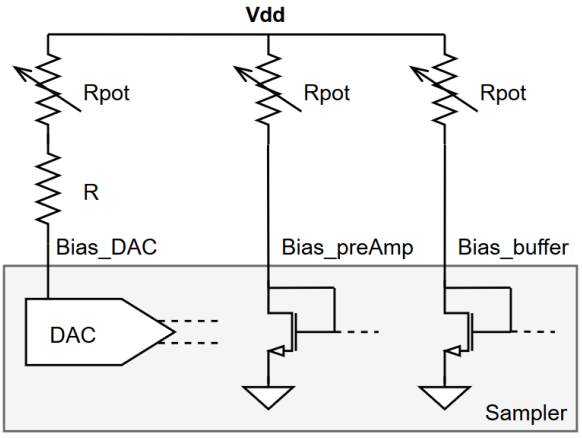


Figure 68: Current biasing networks for the chip

Figure 68 also illustrates the current-biasing realization for this PCB design. Here resistors are simply used to limit the currents going into the terminals. With the supply voltage being $1.2V$, the desired current can be set by use of Ohm's law. However, knowing the total resistance of the branches entail knowing the drain resistance of the biasing transistors and the input resistance of the current-DAC, and relying on them being constant for all instances. Because of the possible variation in resistance, trimming potentiometers are used to have the ability to adjust the resistance depending on the chip being tested. Furthermore, the selection of resistor sizes had to be done through simulations in Cadence. For the two larger bias-currents ($225\mu A$ and $350\mu A$), $5k\Omega$ potentiometers are enough, while for the $100\mu A$ an additional $1k\Omega$ resistors must be connected in series. The used potentiometers were also chosen to have 25 turns of adjustments in order to set the currents more precisely.

6.1.3.3 Digital translation

The sampler operates with a supply voltage of $1.2V$ resulting in corresponding digital levels. The Arduino Leonard control unit on the other hand, operates at $5V$. A digital *HIGH* from the control unit and into the chip may damage the chip due to the high current coming in, and digital *HIGH* from the chip will not be sensed as a digital *HIGH* by the control unit. For this reason, the input and output digital signals need to be translated, or level-shifted, for the two devices to communicate properly. *Figure 69* illustrates the translating implementation.

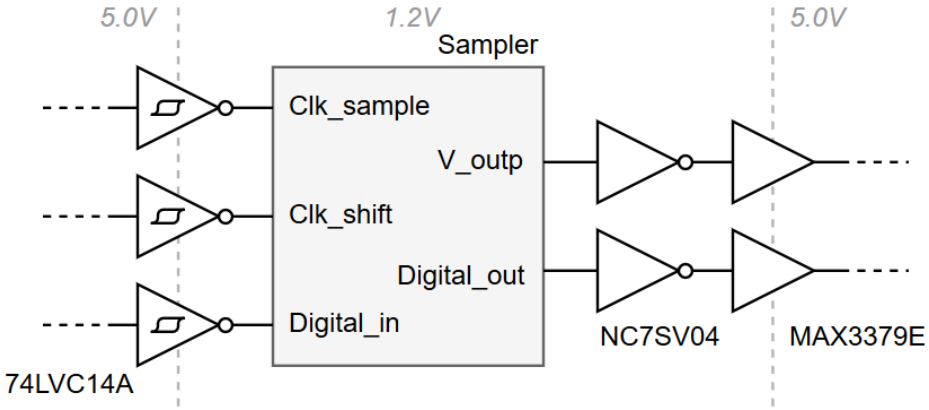


Figure 69: Voltage level translation

74LVC14A are inverting Schmitt triggers with high tolerances on the inputs. They tolerate inputs between $3.3V$ and $5V$ while outputting between $1.2V$ and $3.6V$. The output levels are set by the device supply voltage which is $1.2V$ for this setup.

MAX3379E are low-voltage level translators. They can translate voltages from $1.2V$ and up to $5.5V$. The input and output levels are set by two supply inputs.

NV7SV04 are regular inverters with high low-level-thresholds. As mentioned in *section 5.3.1.2*, the digital output voltage (V_{outp}) of the sampler has a digital *LOW* of about $350mV$. The level translator has a low-level-threshold of $150mV$ which means that it will not accept digital *LOW* of the sampler. For this reason, the NV7SV04 inverter is implemented between the sampler and the level translator. The inverter has a low-level-threshold of $420mV$ and thus can serves as a rail-to-rail extender. The digital output of the shift register is also connected to a NV7SV04 to ensure that its output also reaches the accepted levels of the level translator.

6.1.3.4 Decoupling

Decoupling capacitors are installed for the chip and for the level translators. Two parallel capacitors are placed on every voltage source terminal of the chip and one is places on each of the supply terminals of the level translator. All capacitors are $100pF$ large. The decoupling capacitors are used to filter out supply noise.

6.1.3.5 PCB implementation

Altium Designer was used to design and layout the PCB. The test PCB was implemented with four layers. The inner layers are used for $1.2V$ supply and for ground, while the outer layers are signal layers.

The layout of the PCB is fairly simple. Since the sampler chip has four sides with a sampling circuit on each of them, the sides are labelled as A, B, C and D. The layout was done for A and B, and then mirrored for C and D.

All inverters, translators, potentiometers, connectors and the socket are mounted on the top layer of the PCB, while the resistors and decoupling capacitors are mounted on the bottom layer. This was done to make adjustment and connection points easier accessible, and to make the soldering process more practical.

The PCB is also designed so that the sides not being testes can be turned off. This was done by adding three-way jumpers between the biasing resistors and the chip. With this setup, each biasing terminal of the chip can manually be connected to either the biasing paths or ground. The advantage of being able to shut down unused circuits is that they will not affect the circuit being tested.

Another major reason for having jumpers between the biasing paths and the chip is to have the ability to ground the current-DAC biasing terminal. This is required for testing the offset of a sampler without it being affected by the correction circuit. As previously mentioned, the implemented current-DAC divides the input current between the two output of the DAC and thus will not be able to turn off both correction sides simultaneously. This means that there will always be a voltage present one of the two correction substrates. The only way to ensure that no voltage is output by the correction circuit is to ground the biasing input of the current-DAC.

6.2 MEASUREMENT APPROACH

In this section methods and procedures used for obtaining the results are further discussed. But before that, the abbreviations of the circuit being tested needs to be defined.

As presented in *subsection 6.1.3.5*, every chip has one circuit for testing on each side and these circuits are alphabetically labelled for A to D. So, referencing to a specific circuit will be done by referring to the chip the circuit is implemented in by number, and by referring to the side of the chip with the circuit. For example, C1SA will refer to sampling circuit A on chip number 1.

6.2.1 Initial offset

In theory, finding the offset of the sampler without any correction would be done by grounding the biasing input of the current-DAC and performing the *swp* command with the control unit. The resulting data would show at which step of the sweep the differential voltage at the input of the sampler changes polarity. The derived step is then subtracted from the mean step of the sweeping range and multiplied by the voltage equivalent of 1LSB (least-sigificant-bit), to find the offset of the sampler.

However, during initial testing, the measurements showed some irregularities. When an offset was found, it did not match the offset of the same sampler when the inputs of the sampler were flipped. Ideally the offsets would have the same distance to the mean point, the point at which the two input voltages are equal, of the sweep. Instead, two measured offsets where unequally distanced from the mean point, as is shown in *Figure 70*. In addition, the unequal distances are circuit and chip depended, meaning that they very differently for every testing instance.

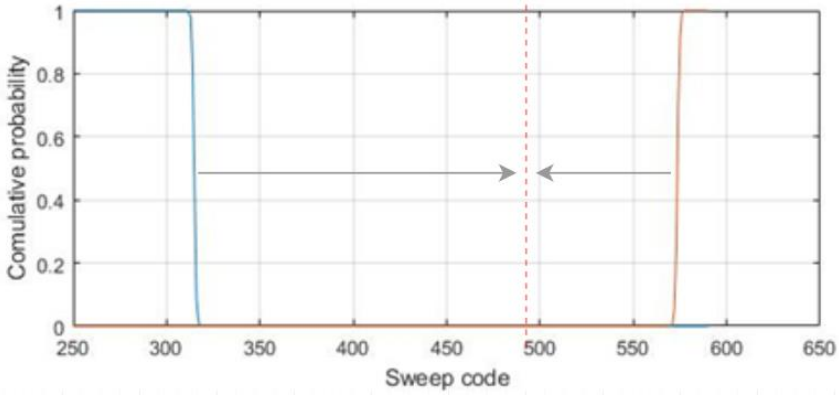


Figure 70: CDF of an offset sweep for both connection options

The causes for this uneven separation will be discussed in the *chapter 0*. However, some assumptions and some alterations must be made in order to perform the measurements. By treating this unwanted effect as an additional offset caused by the PCB and the input paths of the chip, leading to the sampler, an argument can be made that the mean point or voltage at the input of the sampler is the midpoint of the measured offsets.

So, for every sampler that is tested, a new mean value needs to be defined and the following characteristics must be corelated to that value. *Figure 71* illustrates the graph of *Figure 70* where the mean point is redefined to code value 455 and the sweep code is converted to the corresponding voltages. From the *Figure 71* one can derive the offset of the sampler, which is about $99mV$ here.

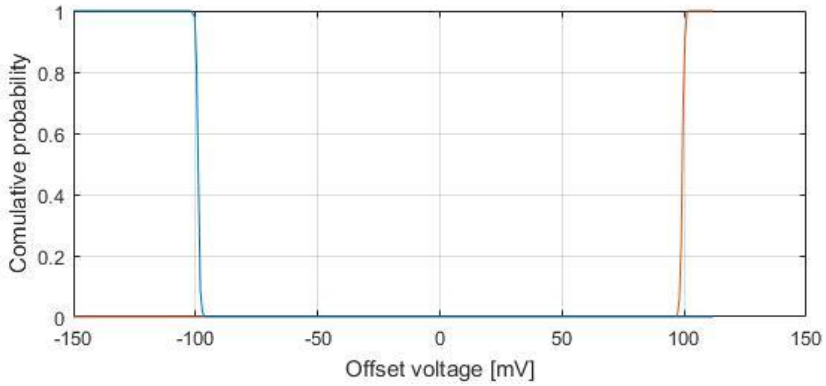


Figure 71: Normalized CDF for the redefined mean of Figure 70 with sweep code converted to mV

Another thing to note from *Figure 70* is that the mean point, the code value at which the output levels of the programable voltage supply are equal, is found to be 490 instead of 511/512 for the 10-bit sweep. This mean point was found by manually stepping the programable voltage supply, with the *set* function, and measuring the outputs with a multimeter. This small shift is caused by imperfections in the controllable voltage source but does not have any consequences for the measurements.

6.2.2 Correction range

The correction ranged is simply found by programming the correction circuit to its two outer ends and performing a *swp* at both ends. Then, it is a matter of subtracting the derived points and converting them to voltage.

6.2.3 Correction code and offset after correction

The correction code that corrects the initial offset best, is found by performing the *all* command which does a 7-bit correction sweep, with a 10-bit threshold sweep for every correction step. From the resulting data the correction code that yields the best offset correction is detected and the remaining uncorrected offset in calculated.

6.2.4 Noise

To find the nose affecting the sampler higher resolution sweep needs to be performed because the 10-bit sweep does not represent the noise well. Because of this, 18-bit sweep function *inv* is used to investigate the sweep range where the output of the sampler switches for one digital value to the other. The green graph of *Figure 72*, shows the noise from the 18-bit sweep.

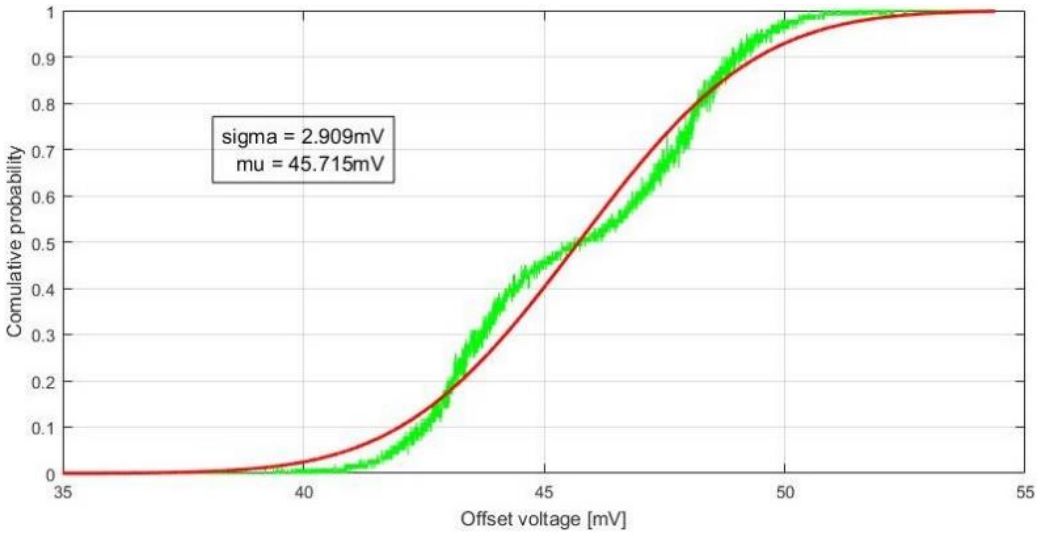


Figure 72: 18-bit threshold sweep of the transition point of C3SD with a fitted CDF

After the 18-bit sweep is performed, the data is transferred to the calculation tool MATLAB. This tool is thus used to find the cumulative probability function that fits the data the best. The standard deviation, or sigma as it is referred to in MATLAB, the function is a direct translation of the noise and thus will be used to determine the measured noise more accurately.

```
10 -    i1 = find(yyy > 0.5,1,'last');
11 -    i2 = find(yyy < 0.5,1,'first');
12 -    mu0 = (xxx(i1) + xxx(i2)) / 2;
13 -    efun = @(v) (sum((yyy-(1-cdf('Normal',xxx,v(1),v(2)))).^2));
14 -    xx = fminsearch(efun, [mu0 256]);
15
16 -    mu = xx(1);
17 -    sigma = xx(2);
```

Figure 73: MATLAB operation for finding the best fitting CDF of the sampled data

Figure 73 shows the functions used in MATLAB for determining the noise. Here *fminsearch();* is the function that does the search for the best fit. The μ, or mu, of the function is also a direct representation of the offset. *Figure 74* shows the post correction sampled data of C3SD and the corresponding noise and offset values.

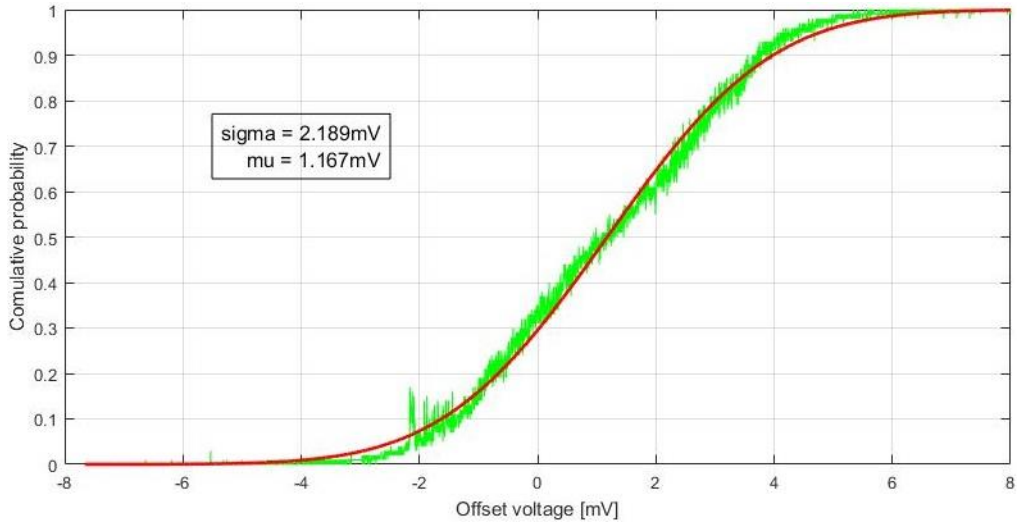


Figure 74: 18-bit threshold sweep of the transition point of C3SD after correction, with a fitted CD

6.3 RESULTS

The measured results are presented in this section.

All the test results are listed in ***Error! Reference source not found.***. However, it is of interest to have a look at an uncompressed measurement.

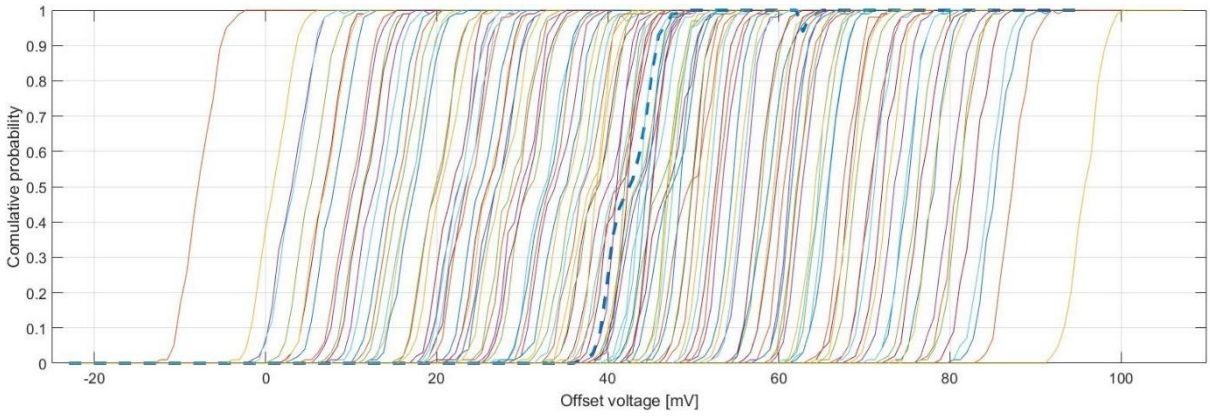


Figure 75: CDF of all correction values for C3SD

Figure 75 shows the initial offset (dark-blue stippled line) and the distribution of the correction effects. These measurements stem from C3SD. *Figure 76* illustrates the same measurements with regard to offset voltage and correction code.

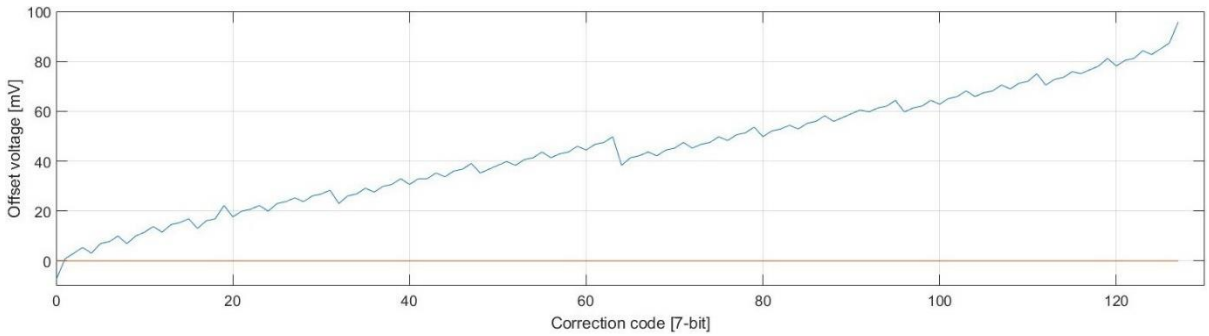


Figure 76: Offset voltage vs correction code of C3SD

	Centerpoint [CODE]	Initial offset [mV]	Initial noise [mV]	Correction [CODE]	Corrected offset [mV]	Corrected niose [mV]	Correction range
C3SB	444	99.656	1.003	N/A	N/A	N/A	
C3SD	490	45.715	2.909	1	1.167	2.189	102.552
C4SA	540	28.480	1.175	20	-0.386	1.309	107.226
C4SB	319	31.134	1.288	10	0.519	1.406	96.503
C4SC	390	89.337	0.770	N/A	N/A	N/A	

Table 11: Final results of the measureme

7 DISCUSSION AND FURTHER WORK

The aim of this master thesis has been to correct offset voltages, caused by mismatch, for the Ultra-Wide-Band (UWB) Swept Threshold (ST) sampling topology. As established throughout the thesis, offsets affect the performance of sampling circuits by reducing the quantization resolutions, increasing the needed sampling ranges and reducing the common-mode rejection-ratios (CMRR) in general. These affects translate to longer sampling sequences and larger power consumption for Swept Threshold samplers. To investigate the offset effects and to evaluate the correction possible options, an existing implementation of the swept threshold sampling technique was chosen as a case circuit [3]. Having the case circuit as a reference, highlights the challenges of offset correction implementation in UWB ST sampling design.

Through the study of the case circuit, the interleaved quantizers were found to cause the most performance reduction due to offset. The reason for this being that quantizers must be designed with close to minimal channel length, in order to support the wide bandwidth characteristics of the sampler, and that they all are affected by individual offsets. With the requirements for the UWB ST sampling circuit being identified, the possible offset correction techniques were evaluated and the Threshold Voltage Correction (TVC) technique was found to be the best approach for correction. The arguments made for the TVC technique were that it would not affect the bandwidth nor the sampling speed of the sampler, and that it would have the least effect on the input gain.

When it comes to the design of the UWB ST sampling circuit, the set specifications were $12GHz$ bandwidth, $2GHz$ sampling rate and $9dB$ gain. The frequency response simulation of *subsection 5.1.3.1* show that the bandwidth and gain specifications were not met. Even though a gain was simulated to be $7.597dB$, this is fairly close to the targeted value. The small difference in gain would only affect the metastability of the comparator a small amount, thus if the comparator is well designed it will not become a big issue. As for the bandwidth of the sampler, the simulation shows a bandwidth of $10.4GHz$ which would in theory suffice. However, the simulation was performed without parasitic and thus this frequency range would be expected to get shorter if the circuit was to be implemented in silicon. Overall, the sampler design did its job by challenging the quantizer design.

Post layout simulation were however performed for the quantizer, since this part of the sampler circuit was to be corrected for offset and implemented on chip. Simulations form *section 5.5.1* show that the bandwidth has been reduced form $13.5GHz$, from the circuit simulations, to $9.92GHz$. The implemented correction circuit did only contribute $160MHz$ of the total bandwidth reduction, which means that the rest was caused by the parasitics. As for the gain of the preamplifier, the simulations showed a $32mdB$ drop when connecting the correction circuit. By viewing these numbers in term of percentage, the selected correction technique reduced the bandwidth by 1.6% while reducing the gain by 3.2%. These effects seem very moderate, but it is difficult to define the values as good or bad when there is no comparison. In order to verify the effects, the design has on the sampling circuit, other correction techniques should be implemented with the same reference circuit for statements to be made. Also, correction effectiveness is a factor to take into consideration.

As for the correction effectiveness of the threshold voltage correction technique. The designed TVC circuit was simulated correcting offsets of $|62.25mV|$ with a biasing current being $100\mu A$. This offset correction range does exceed the calculated and simulated offset ranges of $|44.16mV|$ and $|38.7mV|$. The measured effectiveness will be discussed shortly.

The final requirement set for the design was to achieve a $2GHz$ sampling rate in order to satisfy the Nyquist sampling theorem. Even though the fundamental requirement is for the rate to be twice the frequency of the bandwidth, meaning the predefined bandwidth of $12GHz * 2$, the twelve

interleaved quantizers share this responsibility and thus the required rate is $2GHz$. With that said, there was no problem achieving the sampling rate throughout the design. S*ubsection 5.5.1.2* establishes that the comparator of the quantizer is able to covert a $1mV$ differential signal within the $500ps$ time frame, and that the following latch and output buffer are able to propagate the sample.

There are no ways of measuring bandwidth, speed nor gain of the fabricated circuit. This is because the circuit does not have an amplifying stage for the clock input signals, and because the amplifier leads straight into the comparator. In order for the limits of the comparator to be tested, a sharp clock signal would have to be presented at clock input of the comparator. Normally this is achieved by adding gain stages to the clock input. However, no gain was designed for the clock and no available instrument can provide a steep enough triggering flank, thus there are physical measurements are performed for bandwidth and sampling rate.

The measurements performed in *chapter 6* with the designed test setup, do however show the effects the implemented offset correction technique has on the quantizer circuit. *Figure 75* illustrates the measured initial offset of the quantizer and the correction capabilities of the correction circuit. First of all, the range does meet the predetermined range specification of the design. For the particular test point, the correction range is $102.552mV$. The range here is beyond the calculated and simulated potential offset range of the implemented circuit. However, comparing the range to the range of the simulated offset correction in *section 5.2.3* shows a slight reduction for the measured values. A reason for this can be that ideally current supplies with no line resistance were used in the simulation.

When it comes to the correction resolution of the offset correction, *Figure 75* shows the distribution of the offset corrected points. The resolution is highest at the middle of the range and gets lower the further form the centre the correction moves. There are two reasons for this. One being, that the correction controlling DAC is non-linear, as simulated in *Figure 63*, causing correction levels to overlap around the heavier weighed most-significant-bit change. And the second reason being that output voltages of the current-to-voltage blocks, of the correction circuit, drop exponentially when the input current approaches zero. This effect is shown in *Figure 53*. The mechanics behind this sudden drop is that the current becomes to low for the transistors to stay in saturation and thus they enter the triode-region. This latter effect is the main reason why the outer correction level steps are so large. These outer “jumps” can be avoided by setting the least-significant bit to “1” and just using the remaining bits to control the correction.

The post correction offset, or corrected offset as it is referenced to in *Table 11*, is depended on the resolution density of the offset range. The smaller the initial offset is, the better the offset can be compensated for. If the offset is larger than desired, like for C3SD, the current can be tuned for better fit. However, additional tuning is not an attractive prospect for integrated circuits since it would use even more area.

Investigating the initial offset of the quantizer, indicates that the calculated and simulated offset predictions may not be good enough. With offsets close to $100mV$, the designed offset correction circuits can not adjust for those offsets. It is possible to extend the correction ranges by raising the bias current for the correction DAC. But even that is limited due to the differential voltage levelling out with increase in current, as is illustrated in *Figure 53*. The best solution here would be to redo the layout design.

The test setup for the measurements had unexpected effects on the measurements. As presented in *section 6.2.1* the crossing point at which the two input voltages intersect do not represent the measured points. This effect might have an influence on the measured result. From what can be observed, is seems like there is some kind of offset caused by the test PCB and the and the input paths of the fabricated chip. The reason for including both PCB and chip in this assumption is that

the offset changes for every chip tested. I better investigation of the effects must be done before a conclusion can be made. However, even with these effects the designed circuits can be shown to adjust the offsets even is the offset targets are not met.

For future works, there should be put focus on test setup in order to avoid the mentioned unwanted measurement effects.

When it comes to the design of the this implemented circuit, there is much potential in improving the layout of the chip. In this project the common centroid layout method was not used which may have led to some additional mismatch. There are also more optimizations that can be done to improve the performance of the sampling circuit.

To have a better evaluation of which offset correction technique is best suited for the swept-threshold sampling architecture, all the techniques should be implemented in an identical circuit. This would of course give a better foundation for evaluation.

Also, for more practical results, the entire sampling circuit can be implemented to have a greater perspective of the correction effects and the performance gain that come form correction.

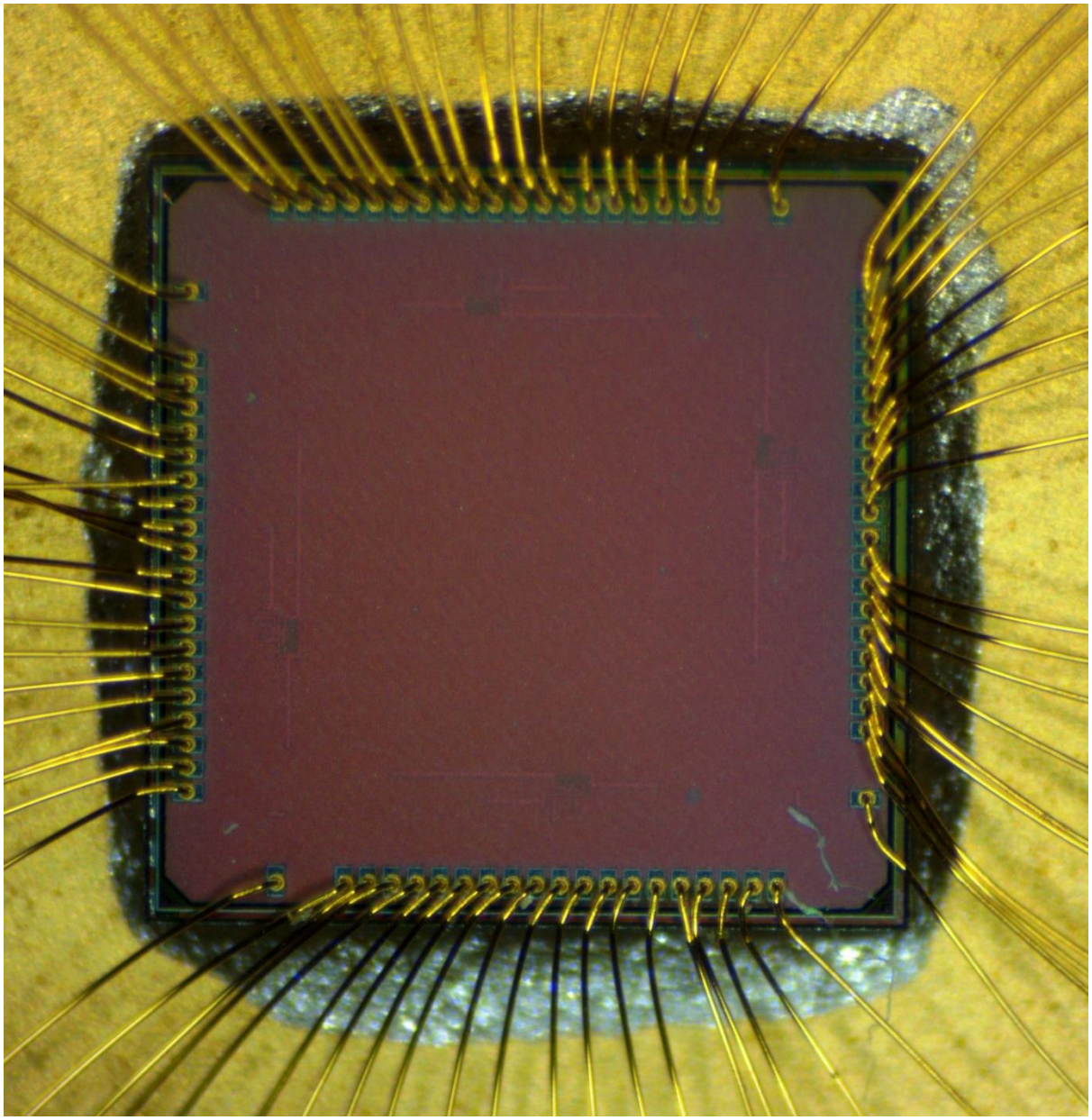
8 CONCLUSION

Through this thesis, multiple offset correction techniques were evaluated for the Ultra-Wide-Band Swept-Threshold Sampling architecture. The Threshold Voltage Correction technique was found to suit the UWB ST sampling the most. To achieve more realistic results, the sampling circuit of [3] was first designed to prove the sampling architecture. Then the quantizing circuit of the sampler was integrated in the TSMC 65-nm process.

Through simulations and measurements, the TVC correction technique was proved to be effective. Simulations demonstrated the correction technique only contributed 3.2% loss to gain and 1,6% reduction of bandwidth for the quantizing circuit. Measured results show the correction effectiveness being up to $|53.6mV|$.

APPENDIX A, FABRICATED CHIP

Project implemented in TSMC 65nm- process.



APPENDIX B, CODE FOR MEASUREMENT

The programming code used to the automated measurements with an Arduino Leonardo

```
#include<SPI.h>
//Pins for chip 456
const int sampleClock = 9;
const int dataClock = 12;
const int dataIn = 13;
const int chipOutput = 8;
//Pins for DAC
const int chipSelect = 7;
const int nRESET = 4;
const int pwrDown = 2;
int startRange = 200;
int endRange = 700;
int dacSetling = 20;
float ampleDelay = 0.2;
byte calPoint = 0;
long dacPoint = 0;
void setup() {
  Serial.begin(9600);
  //Setup for DAC
  pinMode(chipSelect, OUTPUT);
  digitalWrite(chipSelect, HIGH);
  pinMode(nRESET, OUTPUT);
  digitalWrite(nRESET, HIGH);
  pinMode(pwrDown, OUTPUT);
  digitalWrite(pwrDown, LOW);
  //Initiating SPI
  SPI.begin();
  SPI.setDataMode(SPI_MODE0);
  SPI.setBitOrder(MSBFIRST);
  //DAC reset
  digitalWrite(nRESET, LOW);
  delay(100);
  digitalWrite(nRESET, HIGH);
  //Set calibration to mid point, 0x3F, and set sample clock low
  calPoint = 0x3F;
  calibrateChip(calPoint);
  digitalWrite(sampleClock, HIGH);
  //Setup for chip pins
  pinMode(sampleClock, OUTPUT);
  digitalWrite(sampleClock, HIGH);
  pinMode(dataClock, OUTPUT);
  digitalWrite(dataClock, HIGH);
  pinMode(dataIn, OUTPUT);
  digitalWrite(dataIn, HIGH);
  //pinMode(chipOutput, INPUT);
}
void loop() {
  //Monitor instructions
  delay(1000);
  Serial.print("These are the commands:\n swp - for sweeping the DAC\n adj - for adjusting the calingration\n");
  Serial.println(" set - to set threshold\n all - sweep for all calibration points\n xit - to go back to main\n");
  bool gogo = true;
  //Endering main loop
  while (gogo) {
    if (Serial.available() > 0) { //Waiting for characters in the serial buffer
      char val1 = Serial.read(); //Reading the first three charecters
```

```
 char val2 = Serial.read();
 char val3 = Serial.read();
 while (Serial.available() != 0) { // Clearing rest of serial buffer
  Serial.read();
 }
 if (val1 == 's' & val2 == 'w' & val3 == 'p') {
  while (Serial.available() != 0) Serial.read();
  Serial.println("Threshold sweep:");
  sweepThreshold();
 } else if (val1 == 'a' & val2 == 'd' & val3 == 'j') {
  while (Serial.available() != 0) Serial.read();
  Serial.print("The current calibration point is at: "); Serial.println(calPoint);
  Serial.println("Set a new value between 0 and 127");
  while (Serial.available() == 0) {}
  int mRead = Serial.parseInt();
  calPoint = (int)mRead;
  Serial.print("The new caliration point is: "); Serial.println(calPoint);
  calibrateChip(calPoint);
  while (Serial.available() != 0) Serial.read();
 } else if (val1 == 's' & val2 == 'e' & val3 == 't') {
  Serial.println("Set MIN (0), MID (1), MAX (2)");
  Serial.println("Increase by 1 (3), Decrease by 1 (4), Increase by 10 (5), Decrease by 10 (6)");
  Serial.println("Increase by 100 (7), Decrease by 100 (8), Increase by 1000 (9), Decrease by 1000 (10)");
  //while (Serial.available() == 0) {}
  //int dacSet = Serial.parseInt();
  //setThreshold(dacSet);
  setThreshold();
  while (Serial.available() != 0) Serial.read();
 } else if (val1 == 'a' & val2 == 'l' & val3 == 'l') {
  while (Serial.available() != 0) Serial.read();
  for(int calVal = 0; calVal < 128; calVal += 4){
   if(Serial.available() != 0){
    while (Serial.available() != 0) Serial.read();
    break;
   }
   byte calByte = (int)calVal;
   calibrateChip(calByte);
   Serial.print("Sweep with calibration point "); Serial.println(calByte);
   sweepThreshold();
  }
 } else if (val1 == 'm' & val2 == 'a' & val3 == 'n') { //manual sweep
       while (Serial.available() != 0) Serial.read();
  manualSweep();
 } else if (val1 == 'x' & val2 == 'i' & val3 == 't') {
  while (Serial.available() != 0) Serial.read();
  gogo = false;
 } else if (val1 == 'i' & val2 == 'n' & val3 == 'v') {
  while (Serial.available() != 0) Serial.read();
  Serial.println("Enter start point:");
  while(Serial.available() == 0){};
  int lowerRange = Serial.parseInt();
  while (Serial.available() != 0) Serial.read();
  Serial.println("Enter end point:");
  while(Serial.available() == 0){};
  int upperRange = Serial.parseInt();
  while (Serial.available() != 0) Serial.read();
  investigateSweep(lowerRange, upperRange);
 } else {
  while (Serial.available() != 0) Serial.read();
  Serial.println("Wrong command\n");
  gogo = false;
 }
}
```

```
  }
}
void sweepThreshold() {
  byte temp = 0;
  long value = 0;
  float avgValue = 0;
  for (long cnt = startRange; cnt < (endRange + 1); cnt++) {
    value = 0;
    digitalWrite(chipSelect, LOW);
    temp = (long)cnt >> 8;
    SPI.transfer(temp);
    temp = (long)cnt;
    SPI.transfer(temp);
    SPI.transfer(0x00);
    delay(1);
    digitalWrite(chipSelect, HIGH);
    delay(dacSetling);
    for (int i = 0; i < 1000; i++){
    digitalWrite(sampleClock, LOW); delay(ampleDelay);
    digitalWrite(sampleClock, HIGH); delay(ampleDelay);
    value += digitalRead(chipOutput);
    }
    float convert = (long)value;
    avgValue = convert / 1000;
    //if (value != value2){
    Serial.print(cnt);
    Serial.print("\t");
    Serial.println(avgValue);
    if (Serial.available() != 0) {
      while (Serial.available() != 0) Serial.read();
      break;
    }
  }
  Serial.println("Sweep done");
}
void investigateSweep(int lowR, int uppR) {
  byte temp = 0;
  long value = 0;
  float avgValue = 0;
  for (int lol = lowR; lol < (uppR); lol++){
  for (int cnt = 0; cnt < 256; cnt += 1) {
    value = 0;
    digitalWrite(chipSelect, LOW);
    temp = (int)lol >> 8;
    SPI.transfer(temp);
    temp = (int)lol;
    SPI.transfer(temp);
    temp = (int)cnt;
    SPI.transfer(temp);
    delay(1);
    digitalWrite(chipSelect, HIGH);
    delay(dacSetling);
    for (int i = 0; i < 1000; i++){
    digitalWrite(sampleClock, LOW); delay(ampleDelay);
    digitalWrite(sampleClock, HIGH); delay(ampleDelay);
    value += digitalRead(chipOutput);
    }
    float convert = (long)value;
    avgValue = convert / 1000;
    Serial.print(lol);
   Serial.print("\t");
    Serial.print(cnt);
    Serial.print("\t");
```

```
   Serial.println(avgValue);
   if (Serial.available() != 0) {
     while (Serial.available() != 0) Serial.read();
     break;
   }
  }
  }
}
void manualSweep() {
  byte temp = 0;
  long value = 0;
  float avgValue = 0;
  int numOver = 10000;
  for (long cnt = 530; cnt < (560 + 1); cnt++) {
   value = 0;
   while(Serial.available() == 0){};
   if(Serial.read() == '9') break;
   if (Serial.available() != 0) {
     while (Serial.available() != 0) Serial.read();
   }
   digitalWrite(chipSelect, LOW);
   temp = (long)cnt >> 8;
   SPI.transfer(temp);
   temp = (long)cnt;
   SPI.transfer(temp);
   SPI.transfer(0x00);
   delay(1);
   digitalWrite(chipSelect, HIGH);
   delay(100);
   for (int i = 0; i < numOver; i++){
   digitalWrite(sampleClock, LOW); delay(0.5);
   digitalWrite(sampleClock, HIGH); delay(0.5);
   value += digitalRead(chipOutput);
   }
   float convert = (long)value;
   avgValue = convert / numOver;
   Serial.print(cnt);
   Serial.print("\t");
   Serial.println(avgValue);
   if (Serial.available() != 0) {
     while (Serial.available() != 0) Serial.read();
     break;
   }
  }
  Serial.println("Sweep done");
}
void setThreshold() {
  int a = 0;
  while (a != 99) {
   while (Serial.available() == 0) {}
   a = Serial.parseInt();
   if (a == 0) {
     dacPoint = 0;
   } else if (a == 1) {
     dacPoint = 500;
     //dacPoint = 0x1FFFF;
   } else if (a == 2) {
     dacPoint = 989;
     //dacPoint = 0x3FFFF;
   } else if (a == 3) {
     dacPoint++;
   } else if (a == 4) {
     dacPoint--;
```

```
    } else if (a == 5) {
      dacPoint = dacPoint + 0xA;
    } else if (a == 6) {
      dacPoint = dacPoint - 0xA;
    } else if (a == 7) {
      dacPoint = dacPoint + 0x64;
      } else if (a == 8) {
      dacPoint = dacPoint - 0x64;
      } else if (a == 9) {
      dacPoint = dacPoint + 0x3E8;
      } else if (a == 10) {
      dacPoint = dacPoint - 0x3E8;
      } else if (a == 99) {
        break;
    } else {
      Serial.println("Wrong input\n");
    }
    digitalWrite(chipSelect, LOW);
    //byte tmp = (long)dacPoint >> 16;
    //SPI.transfer(tmp);
    //Serial.println(tmp);
    byte tmp = (long)dacPoint >> 8;
    SPI.transfer(tmp);
    //Serial.println(tmp);
    tmp = (long)dacPoint;
    SPI.transfer(tmp);
    SPI.transfer(0x00);
    //Serial.println(tmp);
    Serial.print("The new DAC point: "); Serial.println(dacPoint);
    digitalWrite(chipSelect, HIGH);
    while (Serial.available() != 0) Serial.read();
  }
}
void calibrateChip(byte calValue) {
  for (int i = 0; i < 7; i++) {
    byte tmp = (calValue >> i) & 0x01; //Need to check order of chip register
    if (tmp < 1) digitalWrite(dataIn, HIGH); //data is inverted since there is an inverter on the input of the chip
    else digitalWrite(dataIn, LOW);
    delay(1); digitalWrite(dataClock, LOW); ////Maybe flip
    delay(1); digitalWrite(dataClock, HIGH); delay(1);
  }
}
```

9 BIBLIOGRAPHY

[1] M. J. M. Pelgrom, A. C. J. Duinmaijer and A. P. G. Welbers, "Matching properties of MOS transistors," *IEEE Journal of Solid-State Circuits,* vol. 24, no. 5, pp. 1433-1439, 1989.

[2] H. A. Hjortland, D. T. Wisland, T. S. Lande, C. Limbodal and K. Meisal, "Thresholded samplers for UWB impulse radar," *IEEE International Symposium on Circuits and Systems,* pp. 1210-1213, 2007.

[3] N. Andersen, K. Granhaug, J. Michaelsen, S. Bagga, H. Hjortland, M. Knutsen, T. Lande and D. Wisland, "A 118-mW Pulse-Based Radar SoC in 55-nm CMOS for Non-Contact Human Vital Signs Detection," *IEEE Journal of Solid-State Circuits,* vol. 52, no. 12, pp. 3421-3433, 2017.

[4] N. H. Waste and D. M. Harris, Integrated Circuit Design, 4th ed., Pearson Education , 2011.

[5] B. P. Wong, A. Mittal, Y. Cao and G. Starr, Nano-CMOS Circuit and Physical Design, 1st ed., John Wiley & Sons, 2005.

[6] A. Abumurad and K. Choi, "Design Procedure and Selection of TIQ Comparators for Flash ADCs," *Circuits, Systems, and Signal Processing,* vol. 37, no. 2, pp. 500-531, 2018.

[7] T. C. Carusone, D. Johns and K. Martin, Analog Integrated Circuit Design, 2nd ed., John Wiley & Sons, 2013.

[8] T. Yu-Chih and S. B. Darling, "Block Copolymer Nanostructures for Technology," *Polymers,* vol. 2, no. 4, 2010.

[9] K. Bernstein and D. Frank, "High-performance CMOS variability in the 65-nm regime and beyond," *IBM Journal of Research and Development,* vol. 50, no. 4.5, pp. 433-449, 2006.

[10] A. S. Sedra and K. C. Smith, Microelectronic Circuits, 7th, Int ed., Oxford University Press, 2015.

[11] B. Razavi, Design of Analog CMOS Integrated Circuits, 2nd ed., McGraw-Hill Education, 2016.

[12] B. Razavi, Principles of Data Conversion System Design, IEEE, 1995.

[13] B. Razavi, "The StrongARM Latch [A Circuit for All Seasons]," *IEEE Solid-State Circuits Magazine,* vol. 7, no. 2, pp. 12-17, 2015.

[14] A. Almansouri, A. Alturki, A. Alshehri, T. Al-Attar and H. Fariborzi, "Improved StrongARM latch comparator: Design, analysis and performance evaluation," *2017 13th Conference on Ph.D. Research in Microelectronics and Electronics,* pp. 89-92, 2017.

[15] S. Shekhar, J. Walling and D. Allstot, "Bandwidth Extension Techniques for CMOS Amplifiers," *IEEE Journal of Solid-State Circuits,* vol. 41, no. 11, pp. 2424-2439, 2006.

[16] Y. Lin, C. Lin and S. Chang, "A 5-bit 3.2-GS/s Flash ADC With a Digital OffsetCalibration Scheme," *IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS,* vol. 18, no. 3, pp. 509-513, 2010.

[17] L. Shiwen, H. Hua, G. Peng, G. Xiaoyan, C. Zhiming, W. Xinghua and Z. Shunan, “Design and Implementation of a CMOS 1Gsps 5bit Flash ADC with Offset Calibration,” *2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing, Beijing,* pp. 1829-1833, 2013.

[18] C. Zhu, J. Lin and Z. Wang, “Background Calibration of Comparator Offsetsin SHA-Less Pipelined ADCs,” *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: EXPRESS BRIEFS,* vol. 66, no. 3, pp. 357-361, 2019.

[19] J. Chen, S. Zheng, K. Sheng, W. Gai and J. Feng, “A Low-Power Area-Efficient Wide-Range Offset Calibration Technique for High-Speed High-Resolution Comparator,” *2017 International Conference on Electron Devices and Solid-State Circuits (EDSSC),* pp. 1-2, 2017.

[20] B. Verbruggen, J. Craninckx, M. Kuijk, P. Wambacq and G. Van der Plas, “A 2.2mW 5b 1.75GS/s Folding Flash ADC in 90nm Digital CMOS,” *2008 IEEE International Solid-State Circuits Conference - Digest of Technical Papers,* pp. 252-611, 2008.

[21] J. Yao, J. Liu and H. Lee, “Bulk Voltage Trimming Offset Calibrationfor High-Speed Flash ADCs,” *IEEE Transactions on Circuits and Systems II: Express Briefs,* vol. 57, no. 2, pp. 110-114, 2010.

[22] D. Li, Q. Meng, F. Li and L. Wang, “An Analysis of Offset Calibration Based Additional Load Capacitor Imbalance for Two-stage Dynamic Comparator,” *2016 6th International Conference on Information Communication and Management (ICICM),* pp. 264-267, 2016.

[23] C. Chan, Y. Zhu, U. Chio, S. Sin, U. Seng-Pan and R. Martins, “A voltage-controlled capacitance offset calibration technique for high resolution dynamic comparatorA voltage-controlled capacitance offset calibration technique for high resolution dynamic comparator,” *2009 International SoC Design Conference (ISOCC),* pp. 392-395, 2009.

[24] Cadence, “ADC Verification Workshop, Version 2.0,” Cadence Design Systems, Inc, 2014.

[25] T.I., “User's Guide, DAC9881 Evaluation Module,” Texas Instruments, 2009.

[26] A.D., “Data Sheet AD8475,” Analog Devices, 2017.

[27] T.I., “Data Sheet, DAC9881,” Texas Instruments, Inc, 2016.

[28] Arduino, “Arduino_BoardLeonardo,” 2019. [Online]. Available: https://www.arduino.cc/en/Main/Arduino_BoardLeonardo. [Accessed 8 August 2019].