CHALMERS
UNIVERSITY OF TECHNOLOGY

UNIVERSITY OF GOTHENBURG

Design of 28 nm FD-SOI CMOS 800 MS/s SAR ADC for wireless applications

Master's thesis in Embedded Electronic System Design

VICTOR ÅBERG

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Gothenburg, Sweden 2016

Master's thesis 2016

Design of 28 nm FD-SOI CMOS 800 MS/s SAR ADC for wireless applications

VICTOR ÅBERG

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Computer Science and Engineering
Chalmers University of Technology
Gothenburg, Sweden 2015

Design of 28 nm FD-SOI CMOS 800 MS/s SAR ADC for wireless applications
VICTOR ÅBERG

Examiner: PER LARSSON-EDEFORS, Computer Science and Engineering
Supervisor: LARS SVENSSON, Computer Science and Engineering
Co-supervisor: CHRISTIAN FAGER, Microtechnology and Nanoscience

Master's Thesis 2016
Department of Computer Science and Engineering
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Typeset in LaTeX
Department of Computer Science and Engineering
Gothenburg, Sweden 2016

Abstract

As user expectations for higher bandwidth continue to rise, new techniques are required. One such is the massive-Multiple Input Multiple Output (MIMO), building on dozens or hundreds of antennas all having their own antenna. Making compact, low cost transceivers then becomes essential. Silicon technology fulfils the cost while suffering in terms of performance which needs to be compensated for by active pre-distortion which require an Analog-to-Digital Converter (ADC). This converter must have low power consumption, small footprint and achieve high sample rates in order to be useful.

This work tries to fulfil these demands by implementing a Successive-Approximation-Register (SAR) Analog-to-Digital Converter (ADC) in a 28 nm Fully Depleted Silicon on Insulator (FD-SOI) Complementary Metal-Oxide Semiconductor (CMOS) process. The converter has been implemented based on the principle of alternating comparators in combination with a redundantly scaled Capacitive Digital-to-Analog Converter (CDAC) that help increase the operation speed. The implementation also includes additional circuitry in order to support testing of the circuit.

The implemented ADC shows an SNDR = 38.4 dB at a sample rate of 800 MS/s. The converter consumes 1.1 mW of power while doing this which results in a FoMW = 20.3 fJ/conversion step. This Figure of Merit (FoM) is among the lowest reported for high speed ADC.

Keywords: SAR, ADC, Redundant scaling, Alternating comparator, 800 MS/s

Acknowledgement

Firstly, I would like to thank my supervisor Lars Svensson for his support during this project, and the discussions which brought deeper insight into the work. Secondly, I would like to thank my co-supervisor Christian Fager for the insights in the intended use for the converter. I would also like to thank Erik Ryman for his help with getting started with the 28 nm process.

I would also like to thank the other students in the master thesis room for our interesting conversations and the insight I gained about my work by discussing it with them.

Lastly, I would like to thank my family for their support during these five years of studies with BSc and MSc.

Victor Åberg, Gothenburg, Sweden June 22, 2016

Contents

List of Figures

List of Tables

Acronyms

ADC Analog-to-Digital Converter

ASIC Application Specific Integrated Circuit

CDAC Capacitive Digital-to-Analog Converter

CMOS Complementary Metal-Oxide Semiconductor

DAC Digital-to-Analog Converter

DC Direct Current

DNL Differential non-linearity error

DSP Digital Signal Processing

ENOB Effective-Number-of-Bits

FD-SOI Fully Depleted Silicon on Insulator

FFT Fast Fourier transform

FoMS Schreier Figure of Merit

FoMW Walden Figure of Merit

FPGA Field-Programmable Gate Array

INL Integral non-linearity

LSB Least Significant Bit

MIMO Multiple Input Multiple Output

MOM Metal Oxide Metal

MOSFET Metal-Oxide Semiconductor Field-Effect Transistor

MSB Most Significant Bit

NMOS N-channel Metal-Oxide Semiconductor Field-Effect Transistor

PMOS P-channel Metal-Oxide Semiconductor Field-Effect Transistor

RF Radio frequency

S&H Sample and hold

SAR Successive-Approximation-Register

SFDR Spurious Free Dynamic Range

SNDR Signal-to-Noise-and-Distortion Ratio

SNR Signal-to-Noise Ratio

SOI Silicon on Insulator

T&H Track and Hold

TI Time Interleaving

1
Introduction

This chapter will give the background to the project as well as introducing the goal and delimitations of the project. The structure of this report will also be presented in this chapter.

1.1 Background

As user expectations of bandwidth and general service level continue to rise, new technologies are required to provide these services. One such technique that recently has been introduced is the concept of massive-Multiple Input Multiple Output (MIMO) transceivers, containing dozens or even hundreds of antennas. The massive-MIMO principles has the advantage that it enables high aggregate user bandwidth per spectral resource. This is achieved by the use of spatial-division multiple access where users at different places within a room can be served by a single access point using the same channels, simultaneously. The Massive-MIMO technology not only improves the bandwidth, but it also increases the radiated efficiency and robustness against jamming [6].

The high transceiver count makes their unit cost a high priority. Low-cost silicon technology allows a high transceiver count but suffers from limited performance. This limited performance can be compensated for by the use of higher-complexity techniques such as adaptive predistortion [7, 8]. This will however require a feedback path from the transmitter output back to the digital domain, thus requiring analog-to-digital conversion. The principle of using active feedback is shown in figure 1.1. The mixer and Analog-to-Digital Converter (ADC) are here used to feed the output back into the Digital Signal Processing (DSP) where non-linearities in the output are compensated for.

Current state-of-the-art high-speed ADCs are commonly based on time-interleaved arrays of unit ADCs [9]. This principle allows for scalable designs where array performance

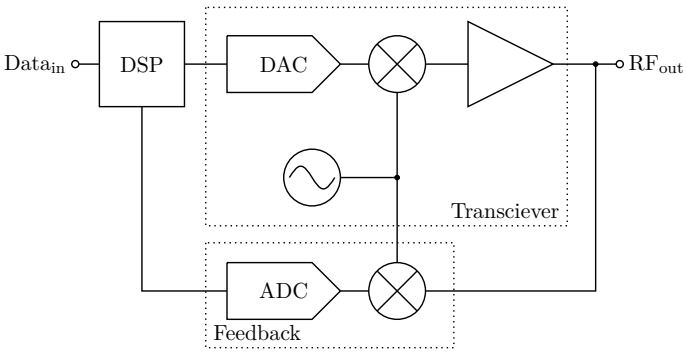


Figure 1.1: Principal sketch of a transmitter with active feedback.

can be selected by means of the number of unit converters.

Any type of ADC can be used as a unit converter, however the Successive-Approximation-Register (SAR) topology is most commonly seen in state-of-the-art implementations [10]. It has a small footprint as well as a low power consumption.

1.2 Purpose and goal

The main goal of this project is to develop a design for a unit SAR ADC in a 28 nm silicon-based technology, based on the main design principles of Kull et al. [1]. The performance specifications for the converter are an eight-bit resolution with an Signal-to-Noise-and-Distortion Ratio (SNDR) of at least 44 dB and a sample rate of 800 MS/s.

To verify these specifications, schematic-level circuit simulations will primary be used. Layout and post-layout simulations will also be performed, primary focusing on the critical blocks to verify their functionality. Layout simulations of the whole circuit will also be performed if there is time for that.

1.3 Delimitations

Chip fabrication and measurements will not be included in this work. This is due to this project's limited time together with the time required for fabrication and testing.

The circuitry required for extracting actual bits from the raw data will also not be considered. This since these circuits are naturally considered part of the overall digital signal processing of the system.

1.4 Document structure

The technical background for this project will be set in chapter 2. Next, the methodology used will be presented in chapter 3. The preliminary specifications for the central components are given in chapter 4. The different design choices made will be presented in chapter 5 followed by chapter 6 which presents the implementation of the converter. Chapter 7 will then present the results before the whole project is concluded in chapter 9.

2

Technical background

This chapter will present the theoretical and technical base for the project. It will start by presenting the fundamental properties of an ADC followed by different ADC topologies. Then different trends in the field will be presented. This will then be followed by a generic SAR topology and the state transition control. Following this will be the reference scaling, and the principle of charge redistribution. Mismatch and offset voltage calibration will then be presented followed by kickback noise. Different comparison metrics will then be presented together with the technology node used. Finally the related work will be presented.

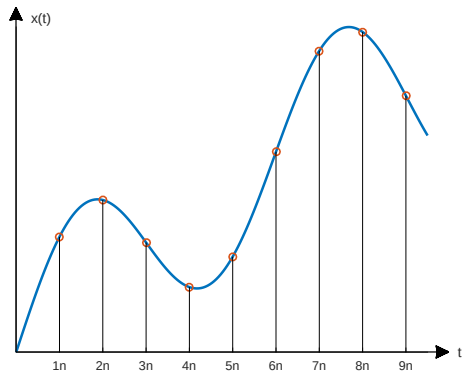
2.1 Fundamental properties of analog to digital conversion

The functionality of an ADC can be described by a set of elementary functions. These are anti-aliasing filtering, sampling, quantization and data encoding [11]. These four elementary functions can be divided into two categories: core operation and system operation. While both filtering and data encoding are important for ADC operation, in a system perspective, they are not as important for generating the core functionality. Sampling and quantization on the other hand are essential for the core operation of an ADC.

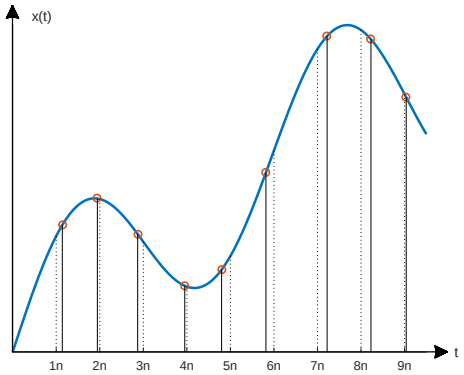
The ideal case of representing an analog signal by a digital representation would mean no errors either in sampling or quantization. This would require an infinite number of quantization levels and an infinitesimally short sampling interval given that the signal is not bandwidth limited. This is however not possible since it would require an infinite number of bits to represent the amplitude at the same time as the circuit would have to operate at an infinite frequency. Instead, both the sampling frequency and the number of quantization levels need to be finite. This means that a non-linear operation needs to be performed both in the time and in the amplitude domain, converting a continuous-time and continuous-amplitude signal into a discrete-time and quantized amplitude representation [11]. This conversion will introduce error sources and fundamental limitations which will be described below.

2.1.1 Sampling

The sampler is the part of the ADC which is responsible for converting the continuous-time input signal into a discrete-time representation. This representation is however only valid during the exact sampling moment. Ideally, the sampler yields a sequence of weighted delta pulses representing the continuous-time signal [11]. The function of generating these delta pulses is a multiplication of the input signal with a sequence of delta pulses separated by the sampling period T. This operation can also be seen as mixing between the input signal and a train of Dirac pulses. A continuous-time signal and its sampled representation



(a) Continuous signal and the sampled representation of it.



(b) The effect of jitter on the sampled representation of the continuous signal. The jitter affected sampling points are shown as solid lines while the ideal sampling points are shown as dotted lines.

Figure 2.1: A representation of sampled continuous time signal (a) and the effect of jitter during the sampling (b).

is shown in figure 2.1 (a).

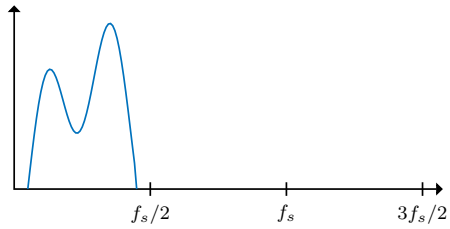
To be able to represent all the desired frequencies contained in the input signal with a sampled representation of it, a minimum sampling frequency f_s will be required [11]. This frequency is given by the Nyquist sampling theorem given in equation (2.1). This theorem also gives the highest frequency which can be represented after sampling, given that the frequency is within the range $0 - f_s/2$, and this frequency is known as the Nyquist frequency.

$$f_{sig} < f_s/2 \tag{2.1}$$

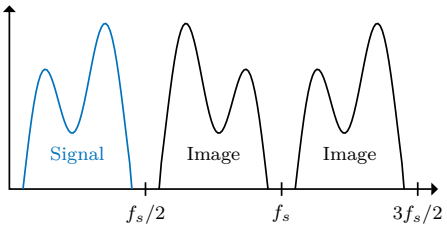
Frequencies higher than the Nyquist frequency can also be sampled and represented. However, the representation is unique only if all frequencies contained in the signal are within the same *Nyquist zone* such as the second or third Nyquist zone defined by the frequency ranges $f_s/2 - f_s$ and $f_s - 3f_s/2$ respectively [11]. The frequency range $0 - f_s/2$ is known as the Nyquist band or base band. Sampling signals using these higher order Nyquist zones is known as under-sampling and the minimum sampling frequency is calculated with a generalization of equation (2.1) found in equation (2.2). This equation will however only give the theoretical minimum sampling frequency which may not work due to the folding of the Nyquist bands.

$$f_H - f_L < f_s/2 \tag{2.2}$$

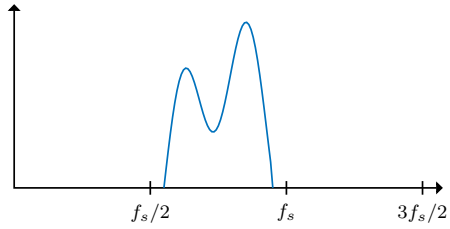
When the input signal is multiplied with the sequence of delta pulses, images of the signal emerge around all multiples of the sample frequency f_s [11]. These images are what creates baseband representation of a signal situated within a higher Nyquist zone. Figure 2.2 (a) shows the spectrum for a signal within the Nyquist band and (b) shows the sampled spectrum of the same signal. In figure 2.2 (c) the spectrum of a signal within the second Nyquist zone is shown and its sampled spectrum is given in (d). For a signal within the third Nyquist zone its spectrum is given in figure 2.2 (e) and the sampled spectrum is shown in (f). From figure 2.2 it is also possible to see that for all signals contained within odd Nyquist zones, the baseband representation of the signal spectrum is equal to



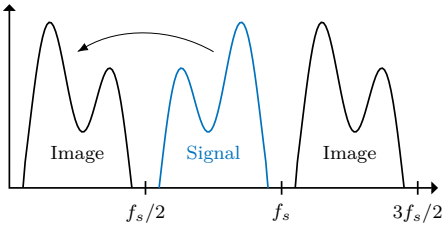
(a) Signal spectrum at Nyquist band



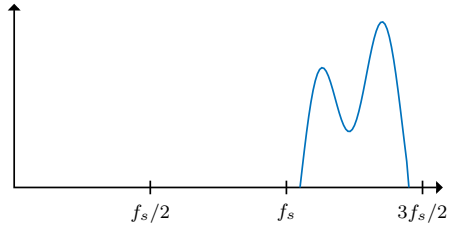
(b) Sampled representation of signal at Nyquist band



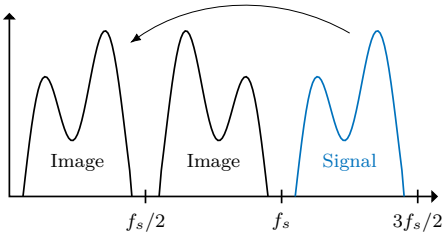
(c) Signal spectrum at second Nyquist zone



(d) Sampled representation of signal at second Nyquist zone



(e) Signal spectrum at third Nyquist zone



(f) Sampled representation of signal at third Nyquist zone

Figure 2.2: Illustration of the spectrum for bandwidth limited signals in the Nyquist band, second Nyquist zone and third Nyquist zone are shown in (a), (c) and (e) respectively. The spectrum of the sampled signal is shown in (b), (d) and (f) using the same Nyquist bands. The signal spectrum is shown in blue while images of the signal spectrum are shown in black.

the signal spectrum. For signals within even Nyquist zones this is not the case; instead the spectrum at baseband is a mirrored representation of the signal spectrum.

2.1.1.1 Jitter

In the ideal situation, sampling will take place at equally separated instances in time. This is also what is assumed during the data processing that follows. This is however not the case, since both clock variations and delay variations between the sampling control logic and sampler itself make the exact sampling instance uncertain [11]. These two sources are the root cause to sampling time variations known as jitter.

The effects of jitter can be seen in figure 2.1 (b) where variations in the sampling instance cause errors in the sampled signal. The amplitude error depends both on the sampling error $\delta(t)$ and the slope of the signal at the sampling point. The error $\Delta X(nT)$ can be calculated using equation (2.3) given the input signal $X_{in}(t) = A\sin(\omega_{in}t)$. Equation (2.3) is a Taylor expansion truncated after the first term and cosine comes from the

derivative of the sine in the input signal.

$$\Delta X\left(nT\right) = A\omega_{in}\delta\left(nT\right)\cos\left(\omega_{in}nT\right) \tag{2.3}$$

The average error power $x_{rms}\left(t\right)$ can be calculated using equation (2.4).

$$x_{rms}\left(t\right)^2 = \frac{A^2\omega_{in}^2}{2}\delta_{rms}\left(t\right)^2 \tag{2.4}$$

Then the Signal-to-Noise Ratio (SNR) can be calculated with equation (2.5) which is based on the average power for a sine wave signal being $A^2/2$. The SNR expression given in equation (2.5) is given in dB and this will be the case for all SNR values throughout the report except for when stated.

$$SNR_{ji} = -20\log\left(\omega_{in}\delta_{rms}\right) \tag{2.5}$$

2.1.1.2 kT/C noise

The sampler in an ADC is typically combined with a hold circuit to store the sampled input during quantization. This circuit typically consists of a capacitor on which the input voltage is stored until the next sample is taken. This combination of a sample switch followed by a hold capacitor is also source of one of the fundamental noise classes, kT/C noise, which limits the ADC performance [11]. The source to this noise is thermal noise inside the sample switch resistance R_s which then is stored in the sample capacitor. The spectrum of kT/C noise is white and its amplitude square is given by equation (2.6).

$$\overline{v_{n,R_s}^2} = \overline{4kTR_s} \tag{2.6}$$

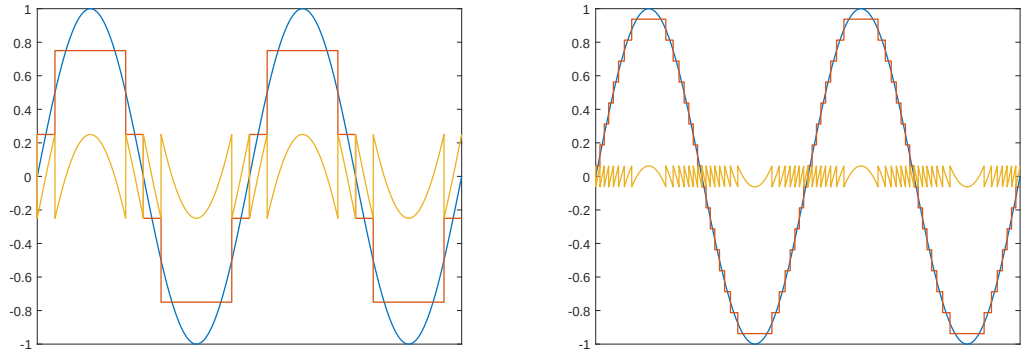
The combination of the sample switch with its internal resistance followed by a sampling capacitance forms a lowpass filter, making the noise power at the sampling capacitor frequency dependent [11]. The noise power over the sampling capacitor is given by equation (2.7), which is the thermal noise power divided by the magnitude of the filter transfer function squared.

$$v_{n,C_s}^2\left(\omega\right) = \frac{\overline{4kTR_s}}{1+\left(\omega R_sC_s\right)^2} \tag{2.7}$$

The total noise power stored in the sampling capacitor can be calculated by integrating $v_{n,C_s}^2\left(\omega\right)$ over all frequencies giving equation (2.8). This results in an expression which is only dependent on the physical temperature of the device and the size of the sampling capacitor. As can be seen, the switch series resistance has no effect on the resulting noise level. The reason for this is that while a higher resistance R_s will increase the noise floor it will also improve the lowpass filtering property, thereby eliminating the dependency on the sample switch resistance [11].

$$P_{n,C_s} = \int_0^\infty \frac{\overline{4kTR_s}}{1+\left(2\pi fR_sC_s\right)^2}\,\mathrm{d}f = 4kTR_s\left[\frac{\tan^{-1}\left(2\pi fR_sC_s\right)}{2\pi R_sC_s}\right]_0^\infty = \frac{kT}{C_s} \tag{2.8}$$

For a differential circuit using the the same sized capacitors in each leg as what would be used in the equivalent single ended case, the total noise sampled would double [12]. However the the input amplitude in the differential case will be twice that of the single ended case, resulting in 4 times as high input power. With the noise power increasing by



(a) 2-bit resolution (b) 4-bit resolution

Figure 2.3: Plot of the continuous (blue) and quantized signal (red) together with the quantization error (yellow) for 2-bit resolution (a) and 4-bit resolution (b).

a factor of 2 while the signal power increases by a factor of 4 results in an increased SNR of 3 dB. To get the same SNR as in the single ended case, it would be possible to use half the single ended sampling capacitance at each leg making the total sampling capacitance in both the differential and single ended case equal. In other words, C_s could be taken as the total sampling capacitance.

2.1.2 Quantization

Quantization is similar to sampling but rather than representing a continuous-time signal at discrete points in time, a continuous-amplitude signal is represented by quantized levels. Another difference is that while the sampled representation is only valid at that exact point in time, the discrete level on the other hand represents a range of valid input values [11].

Quantization levels are typically represented by the mid-point of their specific range, however either of the upper or lower boundary of the quantization range can also be used to represent the interval [11]. Figure 2.3 shows a continuous signal together with its quantized representation for two different number of quantization levels. The quantization intervals are in this case represented by their mid-point.

The size of the quantization intervals Δ can be calculated using equation (2.9). This equation is based on the full scale of the input $X_{FS} = X_{max} - X_{min}$ and on the number of quantization levels M. In most cases, $M = 2^N$ where N is the resolution of the converter given in bits.

$$\Delta = \frac{X_{FS}}{M} = \frac{X_{FS}}{2^N} \tag{2.9}$$

Since a range of values are represented by a single value such as the mid-point of the interval $X_n = X_{min} + (n + 0.5)\,\Delta$, there will be a discrepancy between the continuous and quantized values for all inputs not equal to X_n. This error is known as the quantization error ϵ_Q and it will together with an input signal X_{in} give the quantized output Y as can be seen in equation (2.10).

$$Y = X_{in} + \epsilon_Q = X_{min} + (n + 0.5)\,\Delta \tag{2.10}$$

The quantization error will range from $[-0.5\Delta, 0.5\Delta]$ in the case of mid-point representation. In the case of upper or lower bound representation, the quantization error will range from $[-\Delta, 0]$ and $[0, \Delta]$ respectively [11]. When the signal is outside the dynamic

range $[X_{min}, X_{max}]$, the quantization error will continue to increase linearly with the input signal. The quantization error is shown in figure 2.3. Here it is also possible to see how its amplitude depends on the number of quantization levels.

2.1.2.1 Quantization noise

While kT/C noise is one of the fundamental limitations which limits the converter performance, another one is the quantization error which is often referred to as the quantization noise. Quantization error is not really noise like kT/C noise which is completely random and jitter which in some cases is random. It is however often treated as noise due to the convenience of using the SNR concept [11].

The problem with the quantization error is that it only behaves as noise under specific circumstances. One case in which the error cannot be treated as noise is when the input is a Direct Current (DC) signal since this will give rise to a constant error. Another case is when the signal only varies within a single quantization interval. This will give rise to an error spectrum and signal spectrum that only differs by a DC term [11]. Frequent quantization level transitions will on the other hand decorrelate the noise from the signal making it behave like noise.

Four conditions enable a simple noise-based treatment of the quantization error [11]:

- There should be equal probability of reaching any quantization level.
- A large number of quantization levels are used.
- The quantization intervals should be uniform.
- Errors are uncorrelated with input signal.

To calculate the average quantization noise power, the probability distribution of the quantization error is required [11]. Since $|\epsilon_Q| < \Delta/2$ and as stated above a uniform distribution of ϵ_Q in the interval $-\Delta/2 - \Delta/2$ can be assumed giving $p(\epsilon_Q) = 1/\Delta$ in this range. Using this probability distribution makes it possible to calculate the noise power using equation (2.11).

$$P_Q = \int_{-\infty}^{\infty} \epsilon_Q^2 p(\epsilon_Q)\, \mathrm{d}\epsilon_Q = \int_{-\Delta/2}^{\Delta/2} \frac{\epsilon_Q^2}{\Delta}\, \mathrm{d}\epsilon_Q = \frac{\Delta^2}{12} \tag{2.11}$$

Here it is possible to see that the only thing that affects the power of the quantization noise is the size of the quantization step. With the power for a sine wave expressed in quantization levels being given in equation (2.12), the SNR can be calculated using equation (2.13).

$$P_{sig} = \frac{1}{T}\int_0^T \frac{X_{FS}^2}{4} \sin^2(2\pi f t)\, \mathrm{d}t = \frac{X_{FS}^2}{8} = \frac{(\Delta 2^n)}{8} \tag{2.12}$$

$$SNR_{qe} = 10\log\left(\frac{P_{sig}}{P_Q}\right) = 10\log\left(\frac{(\Delta 2^n)^2}{8}\frac{12}{\Delta^2}\right) = 10\log\left(1.5\cdot 2^{2n}\right) = 6.02n + 1.76 \tag{2.13}$$

In equation (2.13) it can be seen why quantization noise is one of the fundamental limitations in data converters. The SNR for a full scale input signal only depends on the resolution of the converter, which is reasonable since a higher resolution results in closer space quantization levels which minimizes the quantization error and thereby the noise as well.

2.2 ADC topologies

The topology of an ADC has major effects on achievable performance. Some topologies are suitable for high-resolution, low-speed applications, while others are suitable for high-speed applications with lower demands on resolution. Since a high-speed application is considered in this case, only suitable topologies will be presented here. These are the flash, pipeline, and SAR topologies, which are commonly adopted in high-speed applications alone or in different combinations [10].

The three converter topologies mentioned above will first be presented one by one and then a comparison of them will follow. After the comparison of the topologies, the Time Interleaving (TI) architecture will be presented; it can be used to increase the sample rate when a singe converter cannot deliver the performance required.

2.2.1 Flash

The flash topology is just like the name suggests a fast converter, taking the input and delivering the output within a single clock cycle. It works by comparing the input signal with all the decision levels, the levels separating the quantization intervals, at once [11]. The result of all these comparisons will be *thermometer-encoded* with ones as the result from the comparisons where the reference voltage is lower than the signal, and zeros for the cases where the reference voltage is higher than the input. The thermometer-encoded data is then usually converted into binary encoding. The topology of a flash converter is shown in figure 2.4.

Since decisions are taken simultaneously, one comparator will be required for each decision level. This means that an N-bit ADC which has $2^N - 1$ decision levels would require $2^N - 1$ comparators [11]. Each comparator will also require a unique reference voltage level to compare the input to, so $2^N - 1$ reference voltage levels are required.

The reference voltages used to define the decision levels are typically generated by a resistive network, such as a Kelvin divider as seen in figure 2.4 [11]. This way of generating the reference voltage, however, comes with some problems. First, it suffers from a high power consumption due to the direct path from positive to negative voltage supply causing a constant DC current. Second, the large area required for implementing this reference voltage divider makes the resistive network suffer not only from random mismatch but also from global processing variations. These mismatch problems mentioned will show up as a non-linear response when mapping the output of the converter to its input.

2.2.2 Pipeline

The pipeline topology is another topology commonly used in high speed applications. While a converter of flash topology performs the whole conversion within a single clock cycle, a converter of the pipeline topology spreads this task over several clock cycles. The pipeline topology builds on having one or a few reference voltage levels at each stage; performing a coarse quantization per stage; subtracting the corresponding value from the analog input voltage; and amplifying the signal before the next stage [11]. Since the output from the first stages arrives several clock cycles before the data from the last stages, digital logic will be required to delay the data delivered from the first stages before recombining. The pipeline topology is shown in figure 2.5 where the overall topology is shown in (a) and the stage topology is shown in (b). Figure 2.5 (c) shows how the data from the different stages are delivered at different points in time. Each stage here delivers two bits of data.

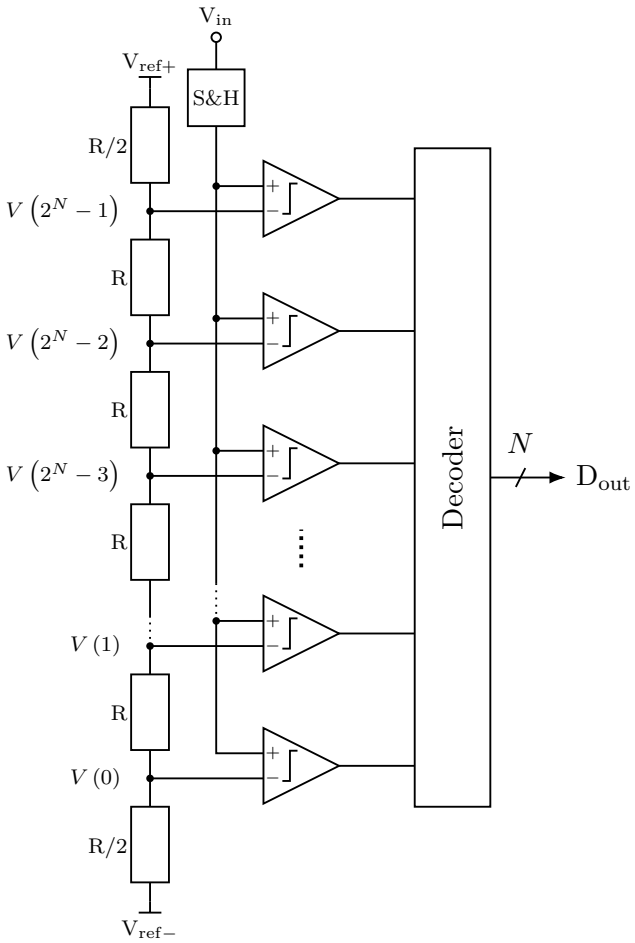


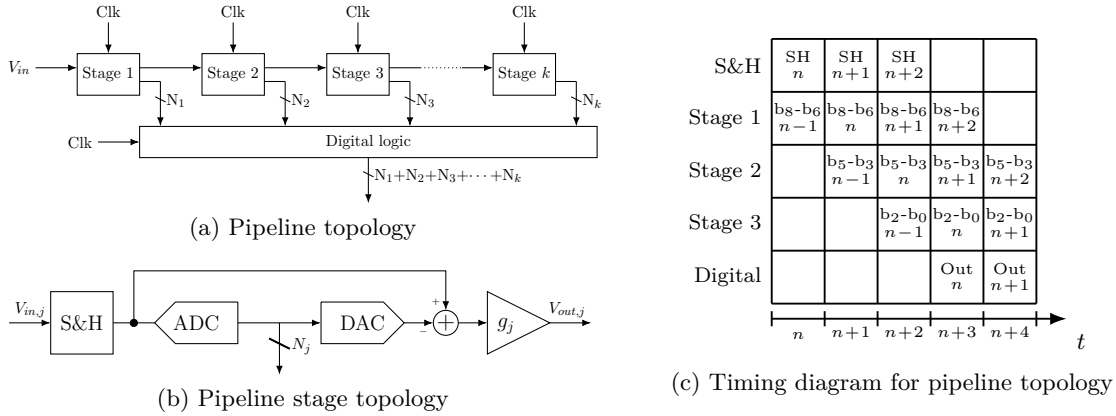
Figure 2.4: Flash ADC.

Because of the multi cycle behaviour of the pipeline converter, it will be necessary to re-sample the signal in between each stage. This means that for a N-bit converter with M bits per stage, N/M sample and hold circuits will be required. This topology will also require N/M amplifiers and DACs together with $(N/M)\,2^M$ comparators.

The stages in a pipeline ADC can be of any type, such as flash or SAR. It is not necessary that the stages are of the same type. It is also not necessary that the stages has the same resolution M. Rather, these are design parameters which can be varied depending on design trade-offs [11]. In practice, flash converters are the most common choice for the stages.

The critical components in this topology are the amplifiers, sub-ADCs, and DACs. For the amplifier, its gain is the critical factor; for the sub-ADCs and DACs, on the other hand, the linearity is the critical factor. Linearity problems can however be managed by using 1-bit stages.

The pipeline topology suffers from a high power consumption due to the biasing current required in the amplifiers. This topology also suffers from a large footprint due to the multiple instances of all components [13]. The latency from sample to output may in some cases also be troublesome such as in feedback loops where short latency is critical [11].



(a) Pipeline topology

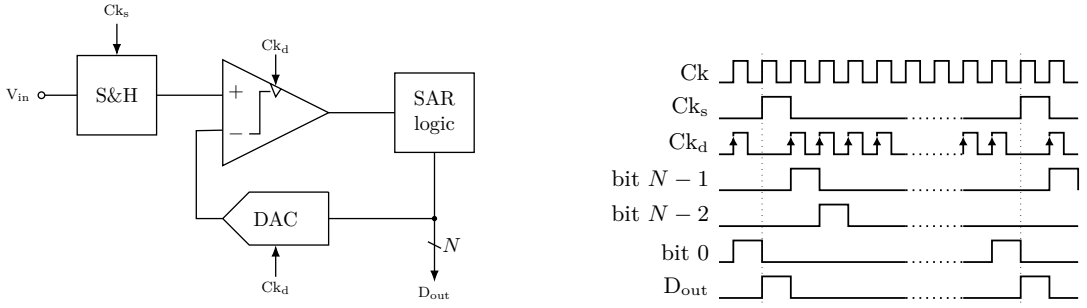
(b) Pipeline stage topology

(c) Timing diagram for pipeline topology

Figure 2.5: Pipeline ADC with its topology shown in (a) and the stage topology shown in (b). The timing diagram for the pipeline topology is shown in (c).

2.2.3 Successive-Approximation-Register

The SAR topology can be seen as an iterative version of the pipeline architecture. It is built on the principle of making a decision and then feed the result of that decision back to compensate for the result of the last comparison [11]. The compensation can either be performed by alter the reference voltage or the input signal. Another difference between the pipeline and SAR topologies is that the SAR topology does not need re-sampling, which is the case for the pipeline topology. In figure 2.6 (a), the topology of an generic SAR converter is shown. A timing graph of when the different signals are generated is shown in figure 2.6 (b). The whole conversion including the sampling is known as a *cycle*. This cycle is then divided into *stages* where each stage corresponds to a decision with its effect on the reference voltage.



(a) SAR topology

(b) Timing diagram

Figure 2.6: SAR ADC with its topology shown in (a) and the timing diagram showing the converters operation shown in (b).

The SAR topology has several advantages. There is no need for amplifiers which require biasing, giving this topology potential for both low power consumption and small footprint [13]. The small footprint is also helped by the fact that only a single sampling capacitor is required.

This topology not only comes with advantages, it also has its drawbacks. One is that the linearity of the ADC depends on the linearity of the DAC used for reference voltage generation. Another drawback is that comparator and control logic speed becomes a bottleneck in terms of operation speed. The comparator and control logic will need to

operate at frequencies higher than Nf_s, where N is the resolution of the converter in bits and f_s is the sampling frequency. How much more than N times the sample rate that is required is determined by the portion of the period used for sampling the value before quantization commences.

As more and more comparisons are performed, the reference voltage will approach the input voltage, and the difference between the input signal and the reference voltage will shrink. As the difference between the voltages shrinks, the time it takes to make a decision increases. This changing comparison time is not experienced in the same way in either the flash topology where there always will be one decision that takes long time, or in the pipeline topology where the difference is amplified between each stage making the difference more constant.

2.2.4 Flash vs Pipeline vs SAR

The topologies mentioned above will next be compared based on their potential in four areas; sample rate, power consumption, footprint, and linearity.

The first comparison concerns the potential sample rate for these different topologies. Here both the flash and pipeline topologies has potential for the highest sample rates, since they only require a single comparison per sampling instance. The SAR topology on the other hand requires N successive comparisons between each sampling instance. The pipeline topology suffers from a latency from the signal is sampled until the result is delivered [11].

Continuing with their power consumption, the SAR topology shows potential for the lowest power consumption due to its absence of amplifiers which requires biasing [14]. The flash topology also lacks amplifiers, but all its parallel comparators which constantly operate together with the resistive ladder generating the reference voltages combine to a significant power consumption [14]. When it comes to the pipeline topology, both amplifiers with their biasing and the sample and hold circuits are large contributors to the pipeline topologies power consumption [14].

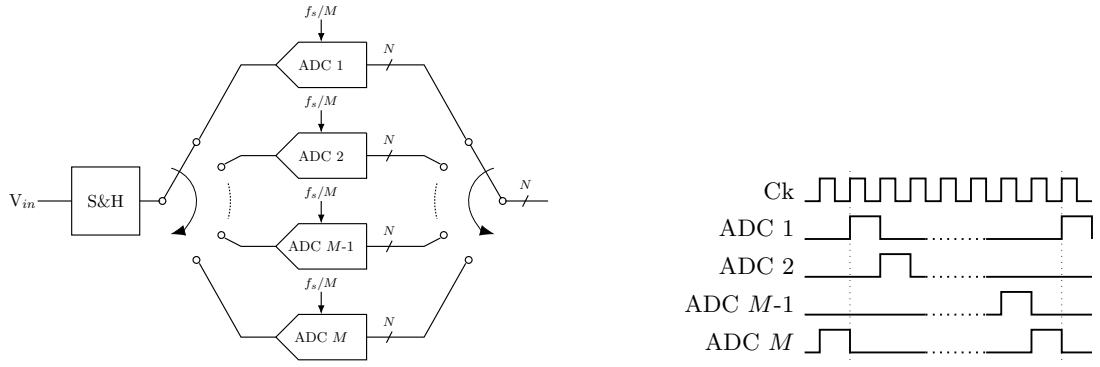
When it comes to the footprint of different topologies, the SAR topology has the best potential [13]. This is due to its single-comparator, single-sample-and-hold structure. The sampling capacitor is one major contributor to the footprint, especially for higher resolutions [11]. In the flash topology case, it is the Kelvin divider in combination with the $2^N - 1$ comparators that contributes to is large footprint. When it comes to the pipeline topology, it is quite obvious that essentially having several small ADC in series will increase the required area [13].

The final review is the linearity, for which the pipeline topology has the greatest potential. The pipeline topology will especially show small non-linearities in the case when each stage delivers 1 bit and the amplifiers has a gain of 2. For both the flash and SAR converters, their reference voltage generators are what is problematic. While a matching of a factor of two can be nicely achieved, larger ratios or more components than a few are trickier and thereby typically result in higher non-linearities. The reason that larger ratios matches worse than smaller ratios is that while small ratios build on components which are similar and can be placed close to each other, larger ratios requires components that significantly differ in size, thereby making them experience the surroundings differently which affects their values.

2.2.5 Time Interleaving

TI is used to increase the sample rate past what a single ADC can achieve. The principle

builds on having several converters operating in parallel but with a sampling instance shifted in time [11]. The converter used in these structures is typically one of those mentioned above, but it might be any other topology. The operation of a TI converter is equal to that of a single ADC whose sample rate has been increased by a factor equalling the number of converters placed in parallel. Figure 2.7 (a) shows a typical TI ADC topology and figure 2.7 (b) shows the instance at which each of the converters are sampled. The same shift in time goes for when each of the converters deliver its output.



(a) Topology (b) Timing diagram

Figure 2.7: Time-interleaved ADC with its topology shown in (a) and a timing graph showing when each converter is sampled.

While the TI structure makes it possible to achieve sample rates that are much higher than what can be achieved with a single converter, it also comes with its limitations. In addition to the fact that a TI structure will have a higher power consumption and larger footprint due to multiple devices, it will also suffer from both clock jitter and mismatch between the different converters.

Clock jitter normally shows up due to variations in the clock generation and due to asymmetrical clock paths as mentioned in section 2.1.1.1. Also, the sampling switch in each of the converters will not trig at the exact same point on the clock edge, thereby also contributing to the jitter [11].

Mismatches in both gain and offset voltage will also affect the performance of a TI ADC. This is since each converter will be affected by mismatch variation in its own way, causing differences which affects the combined performance [11].

In the case of a single converter, the jitter will typically show up as white noise, and the gain and offset errors will be relatively constant. This is not the case when multiple devices are placed in parallel. The jitter will still contain a white noise part, but also components with the frequency f_s/M (and its harmonics) where M is the number of converters in parallel. This constantly varying part comes from the asymmetries in the clock paths to the separate samplers [11]. The same frequency behaviour also shows up for gain and offset errors which also are fixed to each individual converter. These variations are typically what is limiting the performance of a TI ADC and only small effects are required for severe performance degradations.

2.3 Trends

The field of ADCs is constantly moving due to both technical and architectural trends. Technology scaling helps increasing the speed while the power consumption is decreased [14]. At the same time the converter footprint continues to decrease [13].

As technology scaling helps decrease the power consumption, demands for low power devices continues to increase, resulting in a push for new converter architectures [14]. Two clear trends are that pipeline converters are replaced with SAR converters, and flash converters are replaced with the TI topology. In the first case, the benefit lies in removing the amplifiers together with the high speeds that can be achieved in today's technology nodes. The TI converters are more beneficial than the flash converter since more power efficient implementations can be used. The problem is that mismatch can be introduced and limit the performance.

2.4 Generic SAR topology

As can be seen in figure 2.6 (a), several different components are required for a SAR ADC. The topology essentially consists of five components: Track and Hold (T&H), sampling capacitor, comparator, control logic, and feedback reference voltage generator. The functionality of these components will be described below. Possible topologies will also be described in some cases.

The reason for presenting five components here while only four are shown in figure 2.6 (a) is that the Sample and hold (S&H) has been divided into two components, namely T&H and sampling capacitor. This to connect to the actual topology used in the design and the components used there.

2.4.1 Track and Hold

The T&H circuit is responsible for making sure that the input signal is tracked during sampling. It should also hold the voltage which was at the input when sampling ended until the next round of sampling. Since this component during sampling is responsible for passing on the input signal it is important that it has a linear response: any non-linearity directly affects the converter performance. It is also important that its series resistance together with the sampling capacitance has a time constant much smaller than the sampling time [11].

2.4.2 Sampling capacitor

The sampling capacitor stores the analog input voltage between the sampling instances. When the sampling capacitor is used only to store the input signal, the main requirement is that it must be large enough so that kT/C noise will not limit converter SNR. In the case where the sampling capacitor is combined with the reference voltage divider, its design becomes more critical since it now must fulfil the linearity constraints required for the reference buffer as well.

2.4.3 Comparator

The comparator is the component responsible for generating the output by deciding which signal is larger than the other. Comparators comes in many different types which are useful in different cases. Two such types will be presented below, the Strong-ARM latch and the dynamic latch.

In the SAR topology, comparator speed is critical due to the topology's iterative behaviour with multiple comparisons in series for each sample. However, comparator speed is not always the same. Instead, it varies depending on the difference between the two differential inputs. Another critical comparator property is its accuracy [1]. Essentially, the

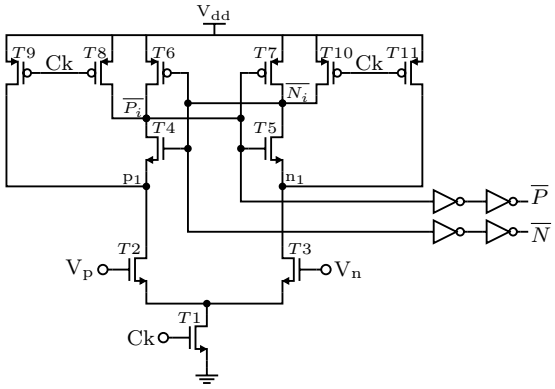


Figure 2.8: Schematic of a Strong-ARM latch, inspired by [1].

comparator determines the accuracy of the whole converter. Comparator power consumption is also important because of its rapid switching. Therefore, only topologies which are not based on pre-amplifiers will be considered here.

2.4.3.1 Strong-ARM latch

The Strong-ARM latch is one of the more popular comparator topologies used in SAR based converters. This is due to its low power consumption and high speed. This low power consumption is due to the lack of static power consumption [15]. One example of a Strong-ARM latch is shown in figure 2.8. This is the most common structure where N-channel Metal-Oxide Semiconductor Field-Effect Transistor (NMOS) transistors are used as input transistors [1,15,16]. There are however also cases where P-channel Metal-Oxide Semiconductor Field-Effect Transistor (PMOS) transistors are used as input transistors [17, 18]. The best choice depends mainly on the input common mode voltage and the desired speed.

The comparator works by having the input fed to the gates of the two input transistors $T2$ and $T3$. The signal should be offset by a common mode voltage which in the NMOS case must be large enough so that the common mode voltage minus the input signal amplitude is larger than the threshold voltage of the transistor. When the clock goes high, nodes p_1 and n_1 are pulled low at rates which depend on the voltages applied to the input transistor gates. When the nodes p_1 and $n1$ are pulled down, they will drag with them nodes $\overline{P_i}$ and $\overline{N_i}$. As one of these nodes is pulled below the switch-over voltage for the other leg's inverter it will start pulling the other node up thereby making the cross-coupled inverters latch the result. As the clock goes low, nodes p_1, n_1, $\overline{P_i}$ and $\overline{N_i}$ are all pulled up to the supply voltage in preparation for the next cycle. Because the latch is directly connected to the input, an inverter or buffer typically follows on the output so that the latch transistors can be kept small [1,17].

2.4.3.2 Dynamic latch

Another commonly-used comparator used in SAR based converters is the dynamic latch [2,19,20]. Just like the Strong-ARM latch, this comparator consumes no static power [20]. In figure 2.9, an example of a dynamic latch is shown. There also exist several alternatives of this topology where parts has been removed [21].

This comparator topology is compared to the Strong-ARM latch divided into three stages. These are input stage, inverter stage and output stage or latch stage [2]. The stage division is a way of isolating the input from the latch so that the latch will not

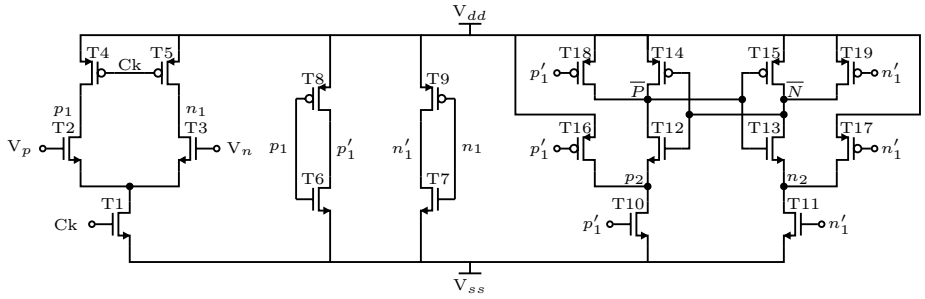


Figure 2.9: Schematic of a dynamic latch, inspired by [2].

load the input transistors. The input must also in this case be offset by a common mode voltage so that the NMOS transistors always are in saturation when the clock goes high. As the clock rises nodes p_1 and n_1 will be pulled down at a pace corresponding to the amplitude of the input at each side. As the nodes p_1 and n_1 are pulled down the inverters in stage two will flip pulling nodes p_1' and n_1' high. This activates the transistors $T10$ and $T11$ which starts to pull down nodes p_2 and n_2 which also will pull down the two output nodes until one of the nodes reaches the switch-over voltage and the cross-coupled inverters latch the result. When the clock goes low, nodes p_1 and n_1 will be pulled high, pulling the second stage outputs low. As the outputs from the inverters in the second stage are pulled low, nodes p_2, n_2, $\overline{P}$, and $\overline{N}$ will be pulled high.

2.4.4 SAR logic

The SAR logic is the component which controls the operation of the whole converter. Essentially the logic can be divided into three main functionalities: clock generation, memory, and state machine [1]. The logic can also contain some other functionalities such as circuitry for managing potential offset calibration circuits, digital correction for correcting the output for offset or gain errors or logic to provide correct binary representation in the case of redundant scaling with additional comparisons.

2.4.4.1 Clock generator

The clock generation part varies depending on the clocking principle used. The difference lies in how the state machine is controlled. The principle for what this block does is however the same. It is responsible for generating all the different clock signals required for the operation of the converter. It includes clocks for sampling, comparator activation and output delivery. The clock generator is also tightly connected to the state machine which orchestrates the converter operation.

2.4.4.2 Memory

The memory is responsible for storing the outputs from each of the comparisons until the result should be delivered to the output. The memory contents are also used to feed back information of previous decisions to the reference voltage generator. Typically, the memory plays a central role in the state machine by keeping track of in which stage the operation is at the moment.

2.4.4.3 State machine

The stage machine controls what will happen next. It decides which memory that data should be stored in. It also controls when each clock signal should be active or not. The

state machine may also be connected to potential additional functionalities such as offset calibration circuits.

2.4.5 Feedback reference voltage generator

The feedback reference voltage generator is the component responsible for generating the voltage to which the input is compared to at each stage. This voltage should change according to the results of the previous decisions, thereby compensating for the information already gained.

This voltage generator can be implemented in different ways depending on if the input is single ended or differential. In the single ended case, the voltage generator will essentially be a DAC as can be seen in figure 2.6 (a). In the differential case, it is more common to combine both the sampling capacitor and reference voltage generator into a single component utilizing the concept of charge redistribution which will be described in section 2.7 [1, 21, 22].

2.5 State transition control

The behaviour of a circuit can be controlled in two ways, either in a synchronous fashion or in a asynchronous fashion. These two method will below be described followed by a description about the metastability problem.

2.5.1 Synchronous control

Synchronous control, i.e. clocking, is the control method most commonly used in digital circuits. It builds on having transitions at known points in time such as at a rising or falling clock edge. Knowing the point at which state transitions occur also makes it possible to know the state of the circuit at any point in time.

Having knowledge about which state the circuit is in at a specific point in time simplifies both the system analysis as well as the debugging. A problem with synchronous control is that when the time it takes to perform an operation depends on the signal, the worst case time must be the base line. This means that in most cases, the operation time will be only a portion of the time assigned for the operation thereby limiting the maximum speed of the system.

2.5.2 Asynchronous control

Asynchronous control is another way of controlling the state of a system. It is used less in digital circuits than synchronous control due to added design complexity. It builds on state transitions taking place when results has been generated instead of at a specific point in time. Since state transitions can take place at any point in time, the technique makes it difficult to know in which state the circuit is at a specific point in time.

Not having knowledge about the operating state of the circuit at a specific point in time makes both system analysis an debugging more troublesome. The benefit of not having state transitions at a specific point in time is that variations in the time it takes to perform an operation can be covered by a time margin in the end of the cycle compared to at each stage which is the case for synchronous clocking. See section 2.2.3 for definition of stage and cycle.

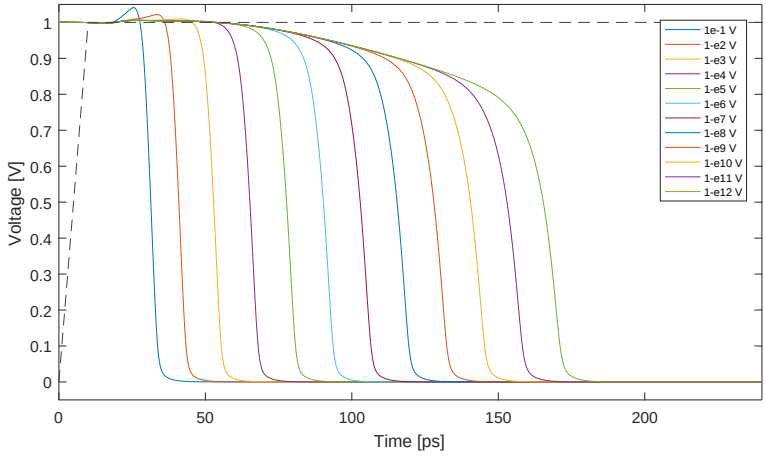


Figure 2.10: Plot of the decision time for the comparator at different input voltages.

2.5.3 Metastability

Metastability is a problem that may occur when time it will take to perform an operation is unknown. In a SAR converter, this problem shows up during the comparisons where the decision time will depend on the input voltage difference.

While the input voltage difference in an ADC at most decisions is larger than half a Least Significant Bit (LSB), there will always be at least one decision per cycle in which the voltage difference is at most 0.5 LSB. These decisions where the decision time is significantly longer than average is where the problem of metastability shows up.

An example on how a signal may affect the time it takes to perform an operation is shown is figure 2.10. In this case, the dependency between decision time and input voltage is shown for a comparator. It can here be seen that there is a large difference between the shortest and longest decision time.

2.6 Reference voltage scaling

The SAR ADC is based on comparing the input voltage to a reference voltage generated by a local DAC where the output from the comparison is used for control. The change in reference voltage that a specific result will have depends on the number of comparisons that has been performed before as well as on the scaling of the reference voltage in the DAC. The DAC scaling can both be redundant and non-redundant depending on the properties desired.

With binary scaling, each output code is unique. This means that there is only one path (sequence of decisions) that can be made in order to reach a specific output. Redundancy on the other hand provides several paths to some of the output codes, making it possible to compensate for an erroneous decision caused by for example insufficient settling of the DAC. While redundancy makes it possible to relax the constrains on settling time at each stage, it will also have a cost in terms of additional comparisons required. The amount of redundancy used is a design parameter, influencing both the settling time relaxation and the cost of additional comparisons.

During the presentation of the scaling algorithms, a set of definitions will be used. A *stage* is as defined in section 2.2.3: one iteration in the SAR algorithm. A *level* is one of the outputs from the reference voltage generator used to compare the input to. It may be so that several inputs to the reference voltage generator results in the same output level

due to the use of redundancy. A *path* is a sequence of decisions and its corresponding levels passed during the previous stages. Each path is unique although the resulting output may be the same as for other paths due to the redundancy. A *step* is the difference between the decision levels at successive stages.

I will describe three principles for scaling the reference voltage: binary scaling, which is a non-redundant technique, and the two redundant techniques: non-binary scaling and compensation.

2.6.1 Binary

Binary scaling is typically used in most applications. Just as the name suggests, its reference voltage is binarily scaled. This means that each step will be half as large as the step at the stage before.

In error-free operation, the difference between the input voltage and reference voltage will always be less than 0.5LSB when all decisions are made [23]. In the case of an erroneous decision, this error will always be larger than one LSB.

The reference voltage at stage k can be calculated using equation (2.14). The digital ADC output can then be calculated by equation (2.15) where $d\,(i)$ is the result from the comparison in stage i and may be either 0 or 1. It can be noticed that while the reference voltage is calculated based on half the range, the digital output code is calculated based on the 0 code. This principle only works for the binary scaling algorithm: the steps are placed so that they are symmetric both relative to the mid-level and to the 0 level.

$$V_{ref}\,(k) = 2^N \left(2^{-1} + \sum_{i=1}^{k} (-1)^{d(i-1)+1}\, 2^{-i}\right) \qquad (k = 1, 2, \ldots, N) \tag{2.14}$$

$$D_{out} = d\,(1)\, 2^{N-1} + d\,(2)\, 2^{N-2} + \cdots + d\,(N-1)\, 2 + d\,(N) = \sum_{i=1}^{N} d\,(i)\, 2^{N-i} \tag{2.15}$$

The decision levels for a 5-bit ADC is shown in figure 2.11 (a). The effect of an erroneous decision is shown in figure 2.11 (a). It is here possible to see that the error is larger than one LSB, compared to the correct operation where the error always is less than 0.5LSB [23].

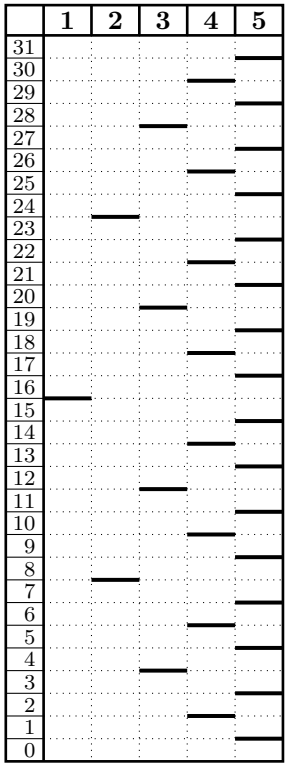
2.6.2 Redundant non-binary

Non-binary scaling is another way of scaling the reference voltage. This method also provides redundancy by having more stages than bits of resolution of the output. This means that the radix between the different steps will be less than 2.

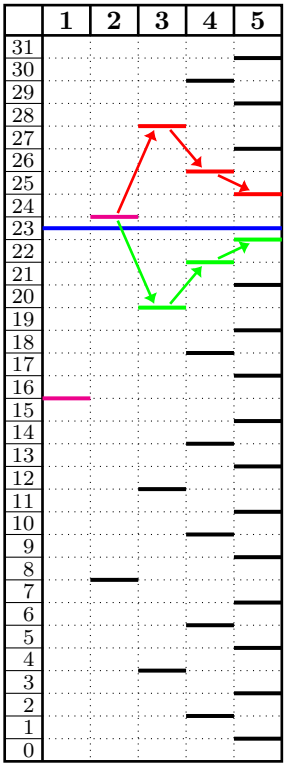
There are two scaling principles that are commonly used: the conventional one building on a radix of $2^{N/M}$, and a generalized one where any radix less than 2 can be used [24]. These two principles will be described in more detail below.

2.6.2.1 Conventional

Just like the binary search algorithm, the non-binary search algorithm uses an equal radix for all the stages. The radix for the non-binary algorithm is defined as $\gamma = 2^{N/M}$ where $1 < \gamma \leq 2$ compared to the radix of 2 for the binary scaling. γ is valid under the condition that $N \leq M$ where N is the resolution in bits and M is the number of stages. This algorithm will be equal to the binary one when $N = M$.



(a) Decision levels for binary scaling. Inspired by [24].



(b) Decision paths for both correct decisions (lower green path) and for a erroneous decision at stage 2 (upper red path). The blue line is the input and the pink levels are shared decision levels.

Figure 2.11: Illustration of the decision levels for the binary search algorithm (a) and the effect of correct and erroneous decisions (b).

Since the radix is lower than 2, additional reference levels will be required [23]. These additional reference levels will provide both new reference levels and multiple paths to existing levels. The actual split between new levels and additional paths depends on the scaling used. This means that different paths, that is, different input codes to the reference voltage generator, will yield the same reference voltage. These redundant codes makes it possible to compensate for erroneous comparator decisions at earlier stages.

The reference voltage at stage k can be calculated with equation (2.16). It will from this equation be possible to calculate the ADC output D_{out} using equation (2.17).

$$V_{ref}(k) = 2^{N-1}\left(1 + \sum_{i=2}^{k}(-1)^{d(i-1)+1}\left(\gamma^{-i+2} - \gamma^{-i+1}\right)\right), \qquad (k = 1, 2, \ldots, M) \quad (2.16)$$

$$D_{out} = 2^{N-1}\left(1 + \sum_{i=2}^{M}(-1)^{d(i-1)+1}\left(\gamma^{-i+2} - \gamma^{-i+1}\right)\right) + 0.5\,(-1)^{d(M)+1} - 0.5 \quad (2.17)$$

Here it can be noticed that the digital output code is calculated in a similar fashion as the reference voltage which is not the case for binary scaling. The reason is that the steps are here symmetric only around the mid-level and not around the 0 level which is the case for binary scaling. The additional term, $0.5\,(-1)^{d(M)+1} - 0.5$, is a result of how the digital output code and the reference voltage are related to each other. The reference level always points on the lowest level in a range. This means that the output code will always point at the range above the reference level. This principle works fine until the last stage where also the range below must be referred to, thereby adding this additional term to the equation.

2.6.2.2 Generalized

The generalized non-binary search algorithm covers both the binary and $2^{N/M}$ scaling algorithms. It is based on choosing at which stages redundancy is desired and how much. While both the binary-scaled and $2^{N/M}$ search algorithms have a range equal to their resolution, the range for the generalized search algorithm can be chosen to be either larger than or equal to its resolution. This additional range is known as the over-range r.

The generalized non-binary search algorithm builds on a set of propositions, equations (2.18) and (2.19), that specifies the properties of the redundancy [24].

$$|V_{in} - V_{ref}(k)| < q(k) \quad (2.18)$$

$$2^M - 2^N = \sum_{i=1}^{M-1} 2^i q(i) + 2r \quad (2.19)$$

The first expression, equation (2.18), specifies the range in which a correct output will be provided even if an erroneous decision previously has been taken. The other expression, equation (2.19), specifies the number of redundant paths that can be provided given the resolution N of the ADC, the number of stages M and the over-range r. The number of redundant paths limits the amount of redundancy that can be provided. Equation (2.19) provides the budget is terms of the number of redundant paths that can be placed where each redundant path is a cost. It can also be seen that the path cost for having redundancy at the last stages is much higher than having it at the first ones. This comes from the fact

that the number of decision levels doubles at each stage compared it the previous one, thereby also doubling the number of redundant paths.

When it comes to the difference between the decision levels of two successive stages $p(i)$ there are two conditions, equations (2.20) and (2.21), that must be fulfilled [24]. The first, equation (2.20), says that the first decision level must be half the range. The second one, equation (2.21), says that the sum off all step sizes must be the the same as the range minus the over-range. The -1 comes from using 0 instead of 1 as the lowest level.

$$p(1) = 2^{N-1} \tag{2.20}$$

$$\sum_{i=1}^{M} p(i) = 2^N - 1 - 2r \tag{2.21}$$

Equation (2.22) shows how to calculate p at each stage based on the redundancy decided on at each stage.

$$p(k+1) = 2^{M-k-1} - q(k) - \sum_{i=k+1}^{M-1} 2^{i-k-1} q(i) \tag{2.22}$$

The reference voltage at stage k can be calculated with equation (2.23). From this it is possible to calculate the digital output code D_{out} with equation (2.24). Just as for the conventional non-binary scaling algorithm this one also requires the additional term $0.5\,(-1)^{d(M)+1} - 0.5$ to compensate for the handling of the comparison level.

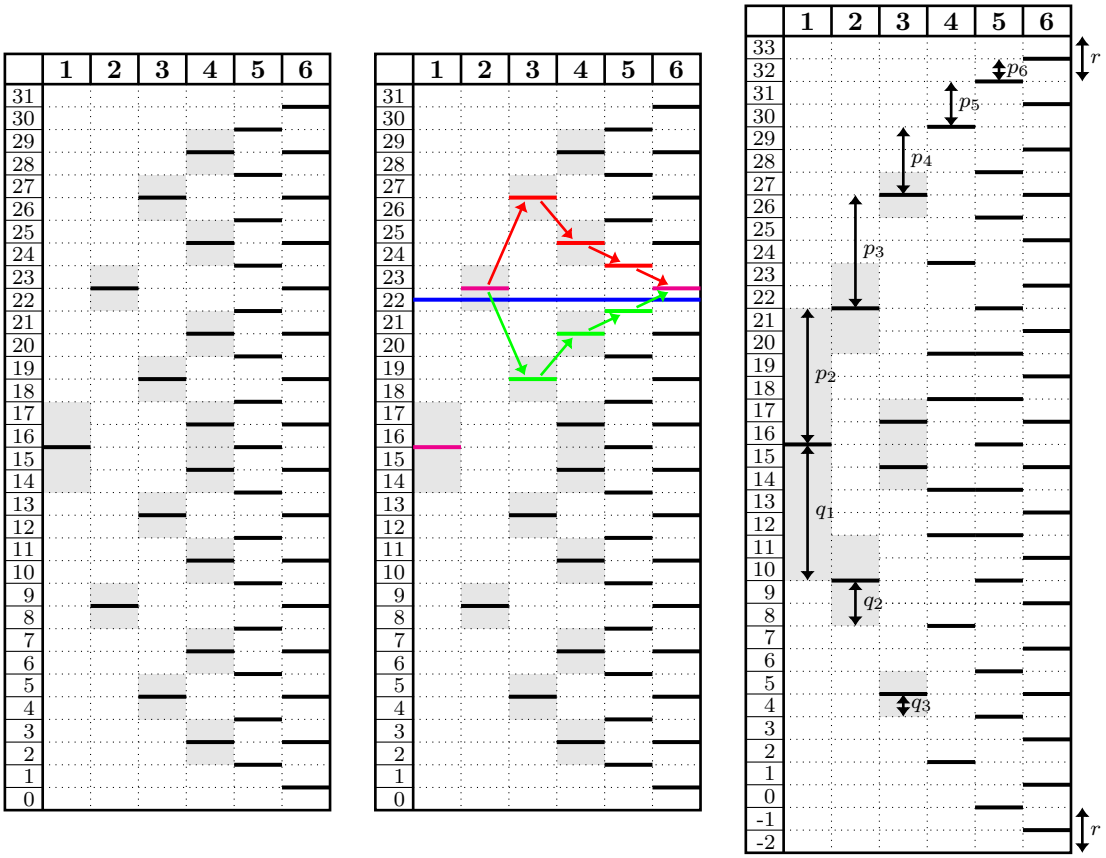
$$V_{ref}(k) = p(1) + \sum_{i=2}^{k} (-1)^{d(i-1)+1} p(i), \qquad (k = 1, 2, \ldots, M) \tag{2.23}$$

$$D_{out} = p(1) + \sum_{i=2}^{M} (-1)^{d(i-1)+1} p(i) + 0.5\,(-1)^{d(M)+1} - 0.5 \tag{2.24}$$

It is also possible to note that if $N = M$, that is $p(i) = 2^{N-i}$, this algorithm is equal to the binary scaling [24]. If $p(i) = \gamma$ with $\gamma = 2^{N/M}$ and $1 < \gamma < 2$ on the other hand, this algorithm is equal to the conventional non-binary scaling [24].

As mentioned above, the decision levels in the non-binary scaling can be placed in several ways, either with or without over-range. Both these cases are shown in figure 2.12 where (a) uses no over-range and (b) uses an over-range of $r = 2$. Starting with the case which has no over-range, shown in figure 2.12 (a), and uses $n = 5$, $m = 6$, and $r = 0$. Together with the redundancy $q = [2, 1, 1, 1, 0, 0]$ this gives the following scaling $p = [16, 7, 4, 2, 1, 1]$. In figure 2.12 (b) the same scaling is used to show what happens in the case of an erroneous decision, in this case at stage 2, compared to the correct the correct decisions.

The case when the scaling uses an over-range is shown in figure 2.12 (c) where $n = 5$, $m = 6$, and $r = 2$. This together with the redundancy $q = [6, 2, 1, 0, 0, 0]$ gives $p = [16, 6, 5, 3, 2, 1]$. It is also possible to see that while the range for a 5-bit ADC is $[0, 31]$, the range in figure 2.12 (c) is $[-2, 33]$. The ranges $[-1, -2]$ and $[32, 33]$ are known as the over-range and it is here equal to ± 2 LSB giving $r = 2$ LSB.



(a) Decision levels for the generic non-binary scaling using,$n = 5$, $m = 6$, $r = 0$, $p = [1674211]$, $q = [211100]$. Inspired by [24].

(b) Decision paths for the generic non-binary scaling presented in (a). The path is shown for both correct decisions (lower green path) and for an erroneous decision at stage 2 (upper red path). The blue line represents the input and the pink levels represent shared decision levels. Inspired by [24].

(c) Decision levels for the generic non-binary scaling with over range. This scaling uses $n = 5$, $m = 6$, $r = 2$, $p = [1665321]$, $q = [621000]$. The figure also shows the range witch is covered by redundancy shaded in gray as well as arrows indicating p, q and r. Inspired by [24].

Figure 2.12: Illustration of the decision levels for the generic non-binary search algorithm without over-range (a), with decision paths for the scaling in (a) in the case of both correct and erroneous decisions (b) and with over-range (c).

2.6.3 Compensation

Another method to redundantly scale the reference voltage is the compensation method. It builds on having a radix 2 scaling for all stages [23]. The redundancy is then implemented by adding compensation stages which have the same range as the stage before, thereby only shifting the levels which are covered by the stage.

The compensation scaling method not only introduces redundancy at the stage at which compensation is added, it also introduces redundancy at the stages before [23]. This means that when choosing how often a compensation stage should be added, the worst case settling error which needs to be covered has to be analysed. This worst case error will take place at the stage furthest away from the compensation stage and still being behind an earlier compensation stage [23]. With a compensation stage added after every third stage, the worst case settling error which can be covered for in each stage is at least 12.5 % of that stage's range. The actual settling error which can be covered is dependent on in what stage the error occurs.

Separating the compensation from the stage scaling means that each compensation stage will result in an additional comparison. The method will also result in an over-range which minimizes the effective range of the converter. The maximum input amplitude $V_{in,amp}$ can be calculated using equation (2.25) where V_{ref} is the reference voltage, C_{bin} is the capacitance for the standard binary-scaled stages, and C_{total} is the total capacitance which includes C_{bin}, the compensation capacitance, as well as the parasitic capacitance [23]. It can here clearly be seen that larger compensation for settling errors results in a smaller input range.

$$V_{in,amp} = V_{ref} \frac{C_{bin}}{C_{total}} \tag{2.25}$$

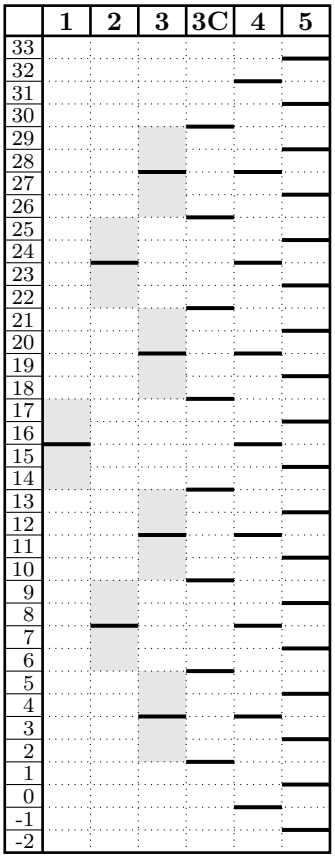
The compensation will show up in the digital output as an offset which makes it simple to remove in the addition where the result at the compensation stages needs to be added to the result at the conventional stages [23]. The offset equals half of the total weight of the compensation stages. This is since half on the compensation weight will be added below and the other half above the range defined by the resolution which can be seen in figure 2.13 (a). The digital output code for the scaling shown in figure 2.13 is shown in equation (2.26).

$$\begin{aligned} D_{out} &= 16d\,(1) + 8d\,(2) + 4d\,(3) + 4\,(d\,(3C) - 0.5) + 2d\,(4) + 1d\,(5) \\ &= 16d\,(1) + 8d\,(2) + 4d\,(3) + 4d\,(3C) + 2d\,(4) + 1d\,(5) - 2 \end{aligned} \tag{2.26}$$

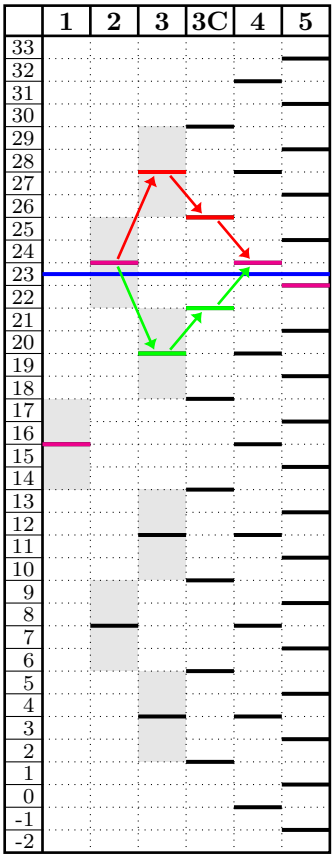
2.6.4 Binary vs Non-binary vs Compensation

All three scaling methodologies mentioned above has their advantages and disadvantages making them useful in different cases. The binary scaling algorithm uses the minimum number of decisions, as given by the resolution. The disadvantage with using binary scaling is that since it has no redundancy it cannot handle erroneous decisions. A typical source of these errors is incomplete settling in the reference voltage generator. Because there is no redundancy that can manage erroneous decisions caused by incomplete settling, a long settling time will be required.

The redundant algorithms all suffer from additional stages added to provide the redundancy. The number of extra comparisons depends on the actual scaling that has been used. The benefit of having these additional comparisons with the resulting redundancy is



(a) Decision levels for the compensation scaling method.



(b) Decision paths for the compensation method in the case of both correct decisions (lower green path) and for an erroneous decision as stage 2 (upper red path). The blue line shows the input level and the pink lines shows shared decision levels among the two cases.

Figure 2.13: A figure showing the decision levels (a) and effects of correct and erroneous decisions (b) for the compensation scaling method. In the scaling used here a compensation level has been placed after stage 3, indicated as stage 3C.

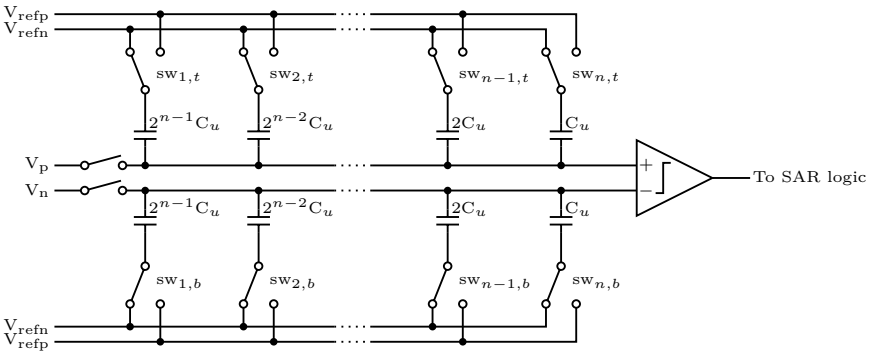


Figure 2.14: Example of a charge redistribution network.

that incomplete settling can be allowed at stages which are covered by redundancy. Thus, shorter settling times can be used, thereby speeding up the converter operation. The time gained by the shorter settling time may however not result in a shorter cycle time since the additional comparisons required consumes part of this time. Indeed, it is not beneficial to use redundant scaling when the decision time for the additional comparisons is longer than the time gained from short settling time.

Non-binary scaling features a larger flexibility in where the redundancy is placed. This also makes it possible to minimize the number of additional stages; however, the optimum scaling may not be suitable for implementation due to the radixes required. Compensation on the other hand is limited in where redundancy may be added, making some stages have unnecessarily large redundancy to provide the required redundancy at earlier stages. The benefit with the compensation method is that is builds on the binary scaling using a radix of 2 at all stages which are not compensating stages. Since all stages have a size of a power of two, implementations of all possible compensation schemes are possible since there will be no case where the component values will be troublesome.

2.7 Charge redistribution

There are several methods to generate the reference voltage used in SAR converters. The simplest one is to use a resistive network. However, this will only work for single ended input signals. Another method that works both in the single ended and differential case is charge redistribution [11]. The principle builds on splitting the sampling capacitor into an array with different stages typically binary scaled. However other scaling methods are possible, such as the ones mentioned in section 2.6.

An example of a charge distribution network for differential operation is shown in figure 2.14. It works by first having the capacitors in each leg pre-charged to one of the reference levels, in this case V_{refn}. Then the input will be sampled onto the capacitors through the sampling switches. After each decision, one of the switches at this stage, top or bottom switch, will switch over to the other reference voltage, in this case V_{refp}, depending on the result of the previous decision.

Switching $sw_{1,t}$ to V_{refp} results in adding charge to the capacitive array. The added charge can be split into two parts: charge added to the capacitor at which the switch is changed, and charge added to the rest of the capacitors. The amount of charge that will be added to each part depends on the relationship between the capacitive sizes which is defined by the scaling of the array. In the case of binary scaling, a switch at the first stage results in that half the charge becomes added to the first stage capacitance and

the rest becomes split among the other capacitors in the array. This will result in an increase of the voltage at the shared node corresponding to half of the change in reference voltage. A switch at the second or nth stage, switch $sw_{2,t}$ or $sw_{n,t}$, would result in a change corresponding to 1/4 or $1/2^n$ of the reference voltage change respectively.

If the capacitors were pre-charged to V_{refp} instead of V_{refn}, the operation would still be the same. The only difference would be that charge will be removed instead of added. The removed charge will still be split in the same way as when charge where added.

2.8 Mismatch

Mismatch is an effect caused by process variations in the manufacturing. These variations can be divided into two kinds, global variations and mismatch, which may affect different parts of the circuit functionality. While the global variations are due to where components are placed on the chip and where the chip is placed on the wafer during manufacturing, the mismatch on the other hand will be a random variation between components affecting even components that are placed next to each other. The global variation will be small for small chips affecting all components in the same way.

Global variations mostly affect the speed of the circuit and the components relying on specific component values. Mismatch on the other hand affects the matching of components, that is, it affects the components which build on ratios between component values. In an ADC, there are two cases where this mismatch will affect the performance. One is the offset voltage and the other is the linearity of the ADC.

2.8.1 Offset voltage

Offset voltage is the input voltage difference at which the comparator sees both inputs as equal. A non-zero offset voltage is a result of mismatch in the comparator. When only a single comparator with a non-zero offset voltage is used in a converter, the output code corresponding to a zero input will actually point to the input voltage difference corresponding to the offset voltage. This means that the output code will be shifted from the ideal representation by the same amount as the offset voltage when translated into digital representation. The effect is caused by random variations in the transistors making up the comparator. These mismatches cause one differential-circuit leg to be faster than the other just as if that leg would have a higher input voltage than the other one.

A large offset voltage is troublesome both since it will make part of the input range unusable since it will always result in either the maximum or minimum output code depending on it the offset error is positive or negative. This problem will however be limited if only a single comparator is used. The problem will be much more severe in the case when multiple comparators are used since they will all have different offsets showing up as varying noise instead of static difference.

2.8.2 Linearity

Linearity a measure of how evenly the digital output codes are separated. The linearity is affected by several different sources where offset voltage can be one. Another mismatch-affected part that affects the linearity is the reference voltage generator.

Since reference voltage generators are based on ratios between different components, matching is a significant contributor to linearity problems. Structures building on larger ratios are typically more vulnerable to mismatch than small ratios due to structure similarities.

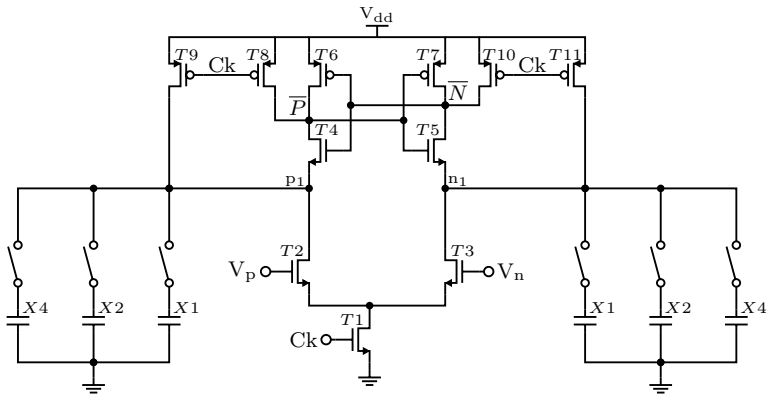


Figure 2.15: Principle for offset voltage compensation by using capacitor switching.

2.9 Offset calibration

As described above, offset voltage may be a significant contributor to the comparator noise picture. It may be the factor limiting the SNR already at small mismatches. To minimize the effects of mismatch contributions on the offset voltage, calibration techniques are often used. Four different techniques will be described here: digital post processing, capacitor switching, body biasing, current injection.

2.9.1 Digital post processing

Digital post processing is a method where the effects of the offset voltage are cancelled out in digital circuitry [21]. It relies on measuring the size of the offset error and on the assumption that this error will be static. The method is especially suitable in cases where only a single comparator is used, since the offset shows up as a shift in the digital output code. When multiple comparators are used, this error will show up as varying noise. Depending on the type of ADC, the noise may have different influence depending on the input voltage, thereby affecting the linearity and making it harder to remove with digital circuitry.

2.9.2 Capacitor switching

Capacitor switching is another method which can be used to compensate for both static and dynamic mismatch. The technique builds on adding additional capacitance either in the nodes p_1 and n_1, as can be seen in figure 2.15, or in the output nodes which are connected to the latching transistors [20, 22]. The capacitance is added to the leg that is faster, thereby slowing it down. The amount depends on how much offset voltage that the comparator is showing. To control which capacitors that should be connected, digital control logic will be required. This logic is essentially constructed around a counter with some surrounding logic [20].

The amount of offset voltage that can be compensated for is determined by the number of capacitors used and their size [20]. The unit capacitance should be smaller or of the same size as the equivalent capacitance responsible for the offset. This capacitance are typically very small and the parasitic capacitance of the switches used to control how much capacitance that are connected are typically a large contributor [22]. Even if no compensation capacitance is added, parasitic capacitance from the switches will always be present at the nodes, decreasing the speed of the comparator.

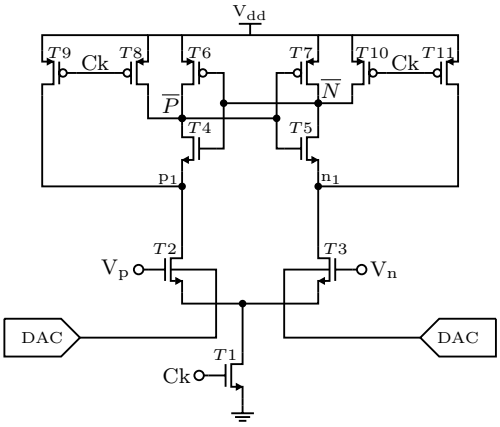


Figure 2.16: Principle for offset compensation using body biasing.

2.9.3 Body biasing

Body biasing can also be used to compensate for both static and dynamic variations. The principle works by applying biasing to the body of the transistor and thereby alter the threshold voltage [25]. The biasing voltage is typically generated by a DAC as can be seen in figure 2.16. The DAC resolution limits the amount of compensation that can be produced. To control the output voltage of the DAC, digital logic will be required.

Body biasing is typically used for coarse compensation with upto 6-bit control signal showed in literature [25,26]. The fine compensation is on the other hand typically building on another method such as digital post processing or capacitor switching [25, 26].

2.9.4 Current injection

Current injection is a method where additional current is injected into the p_1 and n_1 nodes to compensate for mismatch. The current can either be injected in parallel with the input transistors or in parallel with the reset transistors [1, 27]. The placement depends on in which phase that mismatch will be most critical.

The amount of current injected can be controlled in two ways, either with digital control or switched charge control. Both these methods will be described below.

2.9.4.1 Digital control

Digital controlled current injection is similar to the switch capacitor compensation method and can as well handle both static and dynamic variations. The difference is that instead of adding capacitance to either node p_1 or n_1, current will be injected into one of them [27]. The amount of current injected is controlled by activating small transistors as can be seen in figure 2.17. These transistors need to be significantly smaller than the input transistor that they are put in parallel with so that the operation is not affected. The smallest transistor size in combination with the size of the input transistor pair will limit the amount of compensation that can be achieved.

2.9.4.2 Switched charge control

Switched charge control offset compensation differs from the other methods presented above in the sense that it does not build on a digital control circuit. Just as for the digitally controlled current injection technique, parallel transistors are used to inject current. There will however only be one parallel pair and current will be injected into both node p_1 and

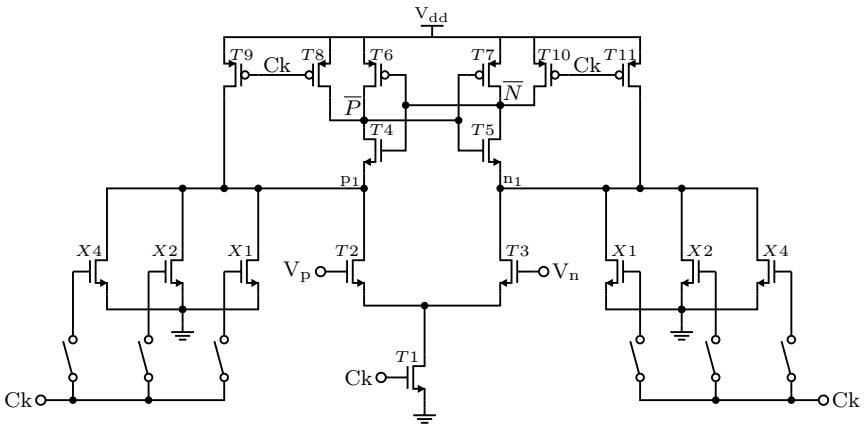
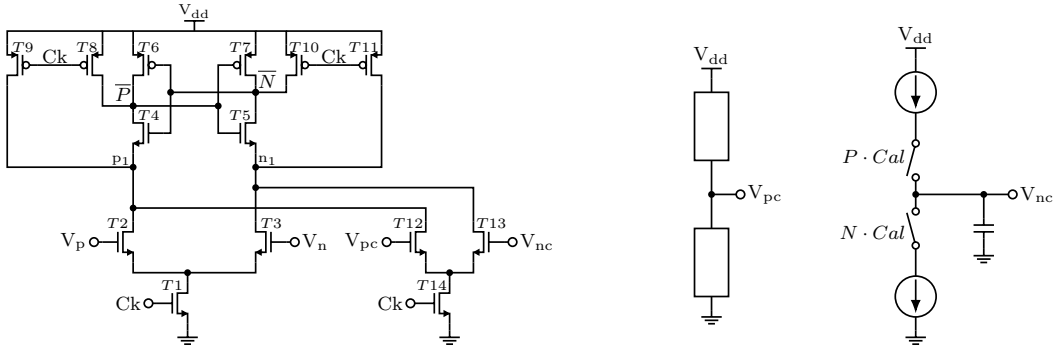


Figure 2.17: The principle of offset compensation using digitally controlled current injection transistors.



(a) Comparator with calibration transistors

(b) Control

Figure 2.18: Principle of offset compensation using switched charge control. The placement of the charge injection transistors are shown in (a) and the control circuit is shown in (b).

n_1 compared to either of them in the case of digital control as can be seen in figure 2.18 (a). The principle behind the control circuit can be seen in figure 2.18 (b).

This compensation method works by having the two compensation transistors providing a current into the nodes p_1 and n_1 to compensate for the difference in the current provided by the input pair in combination with the difference in parasitic capacitance [5]. One of the compensation transistors is set to a fixed voltage and the voltage to the other one is set by a charge switching circuit [1]. The principle of the circuit shown in figure 2.18 (b) is that if the offset voltage is positive, resulting in a discharge of the control capacitor, thereby lowering the control voltage and thereby decreasing the compensation current [5]. If the offset voltage is negative, then charge will be added to the capacitor, increasing the control voltage and thereby also the compensation current.

The resolution of the compensation is determined by the ratio between charge added or removed by the current sources and the charge stored in the capacitor which can be seen in figure 2.18 (b). The ratio between the input transistor pair and the compensation pair together with the common mode calibration voltage determines the amount offset that can be managed.

The benefit with this calibration method is that no external calibration signals will be required, which gives this method potential for a small footprint [5]. Since each comparator has its own calibration circuit, simultaneous calibration of multiple comparators will be possible which is suitable when many comparators are used [5].

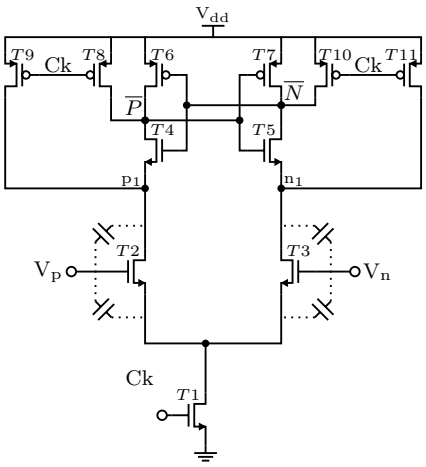


Figure 2.19: Parasitic capacitances responsible for the generation of kickback noise in the comparator.

While the fast calibration cycle makes this method good for dynamic variations, its static behaviour it is not suitable for static compensation when no calibrations take place. This is since the leakage current will slowly drain the charge storing capacitor, thereby changing the compensation state [20].

2.10 Kickback noise

Kickback noise is an effect caused by the parasitic capacitances in the comparator's input transistor pair [15]. These parasitic capacitances connect the drain and source to the gate thereby connecting the outputs to the input. This coupling is not a problem as long as the output voltage is kept stable. The problem shows up under large variations of the output voltage which injects charge in the parasitic capacitance which results in noise across the input impedance [15]. The parasitic capacitances responsible for generating this noise are shown in figure 2.19.

The sampling capacitors are connected to the input of the comparator. This means that the kickback noise will affect the sampled signal and potentially affect the accuracy [3]. The amplitude of the noise depends on the ratio of the parasitic capacitance, which connects the gate to the source and drain, and the size of the sampling capacitance (or rather half the sampling capacitance, since each of the two comparator inputs is connected to one half of the sampling capacitance). The size of the parasitic capacitances is determined by the size of the input transistor pair.

The kickback noise generation can be divided into two phases which coincide with the two operation phases of the comparator. The first phase occurs when the clock goes high, to activate the comparator. When this happens, the nodes p_1, n_1 and n_{ck} will be pulled down towards ground. Then, the polarity of the parasitic capacitances needs to be changed which gives rise to a current running through the parasitic capacitance, removing some charge from the Capacitive Digital-to-Analog Converter (CDAC) and lowering the voltage stored in it. The other phase occurs when the clock goes low to reset the comparator. When this happens, both node p_1 and n_1 will be pulled upwards towards V_{dd} at the same time as node n_{ck} will be floating. When these nodes are pulled up the voltage across the parasitic capacitance will change once again causing the injection of a current charge. This time the current is in the opposite direction, thereby adding charge to the CDAC, increasing the voltage on it.

Kickback noise can be divided into two categories: differential mode and common mode noise [15]. The differential part is due to the differential voltage applied at the input transistor pair. This means that the size of the differential kickback noise is directly related to the input signal and will vary with it. The common mode kickback noise on the other hand is due to the common pull-down of the nodes p_1 and n_1. This part of the kickback noise is directly related to the common mode voltage of the input signal and a larger voltage will result in a larger kickback noise. Mismatch in the input transistors will also increase the kickback noise [1].

From how the kickback noise is generated, it can be seen that the noise will almost have the same amplitude in both phases that it is generated. This means that in a single comparator connected to a fixed sampling capacitor will result in close to complete cancellation of the kickback noise. This will not be the case in more complex topologies. When multiple comparators are used in parallel, the kickback noise may be a factor which will limit the performance of the ADC [15]. In the case of SAR ADCs, where the sampling capacitance often is used as feedback reference voltage generator and is set before the comparator has completely reset, the state will not be the same during the two phases of the kickback noise generation, causing part of it not to be cancelled. The non-cancelled part will be indistinguishable from the voltage change in the sampling capacitor.

2.10.1 Kickback noise reduction

Reduceing the effect of kickback noise can be done in several different ways. A straightforward way would be to decrease the ratio between the size of the parasitics in the input transistor and the size of the sampling capacitance. This will however significantly limit the performance in other ways. Increasing the sampling capacitance will increase both the power consumption and the footprint of the circuit. Decreasing the size of the transistors on the other hand will increase the decision time, thereby decreasing the speed of the comparator which results in a slower converter. Reducing the kickback noise in this manner is therefore not suitable due to its effects on the performance. Instead two other methods to reduce the effects of kickback noise will be presented.

2.10.1.1 Resampling

One method to minimize kickback noise is to essentially re-sample the signal onto a new smaller sampling capacitance which is separate from the main one [3]. The principle is that by re-sampling the signal, separating the comparator from the main sampling capacitor will eliminate the kickback noise. This is since the noise will only reach the smaller capacitance, which is reset once a decision has been taken. For this technique to work, it is important that the clock controlling the re-sampling switch is active for a time which is smaller than the time which the comparator clock is inactive. This is necessary to guarantee that the re-sampling switch has closed so that nothing will not leak past it. A schematic showing how this kickback noise reduction technique could be implemented is shown in figure 2.20.

The are however several troublesome aspects of this way of reducing the kickback noise. The re-sampling capacitance must be large enough not to be kT/C-noise limited. Then, the main sampling capacitance must be several times larger than the re-sampling capacitance so that the charge drawn from this capacitance does not significantly change the voltage stored on it. There will however always be a small change due to charge being removed. Another possible problem is that there may be clock leakage from the re-sampling switch into both the re-sampling capacitor and the main sampling capacitor.

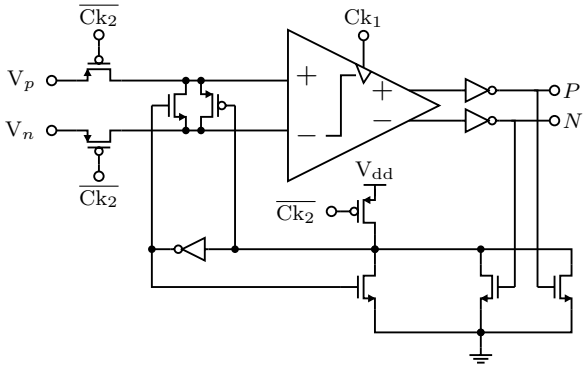


Figure 2.20: Schematic showing how the principle of re-sampling works. Inspired by [3].

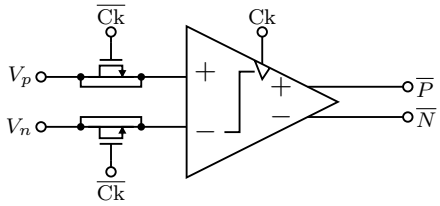


Figure 2.21: Schematic showing how switched capacitance can be used to reduce effects of kickback noise.

This leakage behaves just like kickback noise and thereby minimizes the benefit of adding the re-sampling circuit.

2.10.1.2 Switched capacitance

Another way of reducing the kickback noise is to place dummy transistors on the places which are affected either by kickback noise or clock leakage. These transistors are then switched with a signal which goes in the opposite direction of the signal which causes the noise. When it is the clock that generates the noise, the signal going to the dummy transistor must be the inverted clock signal. If it is the comparator that generates the noise, the signal going to the dummy transistors is the clock signal controlling the comparator operation. The reason for this signal not being inverted is that the signal which is source to the noise already has an inverted behaviour compared to the clock signal. Figure 2.21 shows how this technique to reduce the kickback noise can be implemented at the inputs of a comparator.

The idea is to let the parasitic capacitances in the dummy transistors counteract the effect of the parasitic capacitances causing kickback noise. This means that the dummy transistors should be of approximately half the size of the transistors from which the effect that they should compensate for originates. The actual sizing will depend on the case and how the signals causing the noise are connected to the transistor. The problem with this technique is that it builds on using parasitic capacitances which will depend on the actual implementation. Another problem is that the gate capacitance in a transistor is voltage dependent, which also makes the parasitic capacitance voltage dependent since they are related to the gate capacitance [28].

2.11 Comparison measures

Several different comparison measures can be used to compare different ADC implementations. First, the fundamental measures such as power consumption, resolution, sample rate and so on, are essentially specification measures that do not give any indication about the converter's actual dynamic performance. To compare the operative performance, other measures are required. These can essentially be divided into three categories: signal-to-noise based measures, combined measures, and linearity measures. The signal-to-noise based measures includes SNR, SNDR and Spurious Free Dynamic Range (SFDR). The linearity measures include Differential non-linearity error (DNL) and Integral non-linearity (INL). The combined measures include Walden Figure of Merit (FoMW) and Schreier Figure of Merit (FoMS).

2.11.1 Signal-to-Noise Ratio (SNR)

SNR is the ratio of the power of the signal to the total noise power within the whole Nyquist zone [11]. It is linearly dependent, given decibel scale, on the input amplitude but may not have its maximum value at the highest input amplitude to the system. SNR may also depend on the frequency of the input signal [11].

Since SNR includes all noise sources, its maximum may not coincide with the maximum allowable input signal but rather with a significantly smaller amplitude. The reason for having larger SNR at lower input levels is that distortion typically shows up when the signal gets close to the maximum range, thereby decreasing the SNR.

Since SNR can include a variety of different noise sources, this measure may be troublesome to use in comparisons. A specific noise type may be included in some cases but not in other. This means that taking just the SNR without the description under which constraints that it is valid makes it unusable.

2.11.2 Signal-to-Noise-and-Distortion Ratio (SNDR)

SNDR is similar to SNR but not only includes noise, it also explicitly includes effects from distortion which may also be included in the SNR. The distortion that is included means that SNDR will be dependent both on the amplitude and frequency of the input signal.

SNDR is typically decreasing for large amplitudes at high amplitude due to higher distortion [11]. When specified as a single value, it is often measured at an input amplitude of 1 dB or 3 dB below full scale with an input frequency close to the Nyquist frequency [1, 29].

2.11.3 Spurious Free Dynamic Range (SFDR)

SFDR is a measure which slightly differs form SNR and SNDR in the sense that it does not include all the noise sources. Instead it specifies the input signal and the worst case spur [11]. Depending on the input amplitude the source of the spur may vary [11]. At high input amplitudes, the spur will typically be caused by a higher order harmonic. At lower input levers where distortion is limited, spurs are typically caused by other non-linear effects in the circuit not related to the higher order harmonics.

2.11.4 Differential non-linearity error (DNL)

DNL is a measure of the deviation of the quantization level compared to the ideal difference. With the ideal step size being Δ and the actual step size being $\Delta_r(k) =$

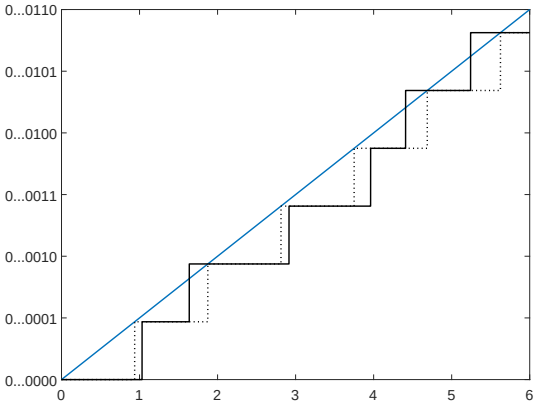


Figure 2.22: The figure shows the random variation on the decision levels causing DNL. The blue line shows the ideal transfer function. The black curve shows the quantization levels affected by random variations while the dotted black line shows the case with ideal quantization levels.

$(X_{k+1} - X_k)$, the DNL can be calculated as in equation (2.27). When talking about DNL as a value, it is typically the maximum $|DNL\,(k)|$ that is meant. The effect of DNL on the quantization levels can be seen in figure 2.22, where the ideal quantization levels are shown together with quantization levels affected by random variations.

$$DNL\,(k) = \frac{\Delta_r\,(k) - \Delta}{\Delta} \tag{2.27}$$

DNL is typically measured in LSB but it is also possible to express it in V, % or ppm [11]. The problem with the three last ones is that it is hard grasp how large the non-linearity is compared to the resolution of the converter. This is not a problem when using LSB since it is based on the resolution of the converter.

DNL can be divided into two parts, correlated and uncorrelated. The correlated part of the DNL is what is deciding the size of the INL [11]. The uncorrelated part on the other hand will not contribute to the INL. Instead it can be treated as noise and be added on top of the quantization noise.

2.11.5 Integral non-linearity (INL)

INL as a measure of the deviation of the transfer function from its ideal representation [11]. It is also known as the deviation from the line fitted between the two end-points. The first case includes the effect of gain and offset which are corrected by the second method which is most commonly used. The reason for this is that gain and offset errors which are linear errors are easy to compensate for in digital circuitry. Digital compensation for non-linear errors is much more complex and thereby this part is of more interest for estimations of the harmonic distortion. Figure 2.23 shows the case where INL has affected the transfer function together with the ideal transfer function. The difference between these two curves gives the INL.

The INL can essentially be seen as the sum of the correlated part of the DNL. The INL can be calculated using equation (2.28) where $X'\,(k)$ is the transition point between codes after correction for gain and offset errors. Equation (2.29) shows the calculation of $X'\,(k)$ where k_{os} is the offset in LSB and G is the gain error.

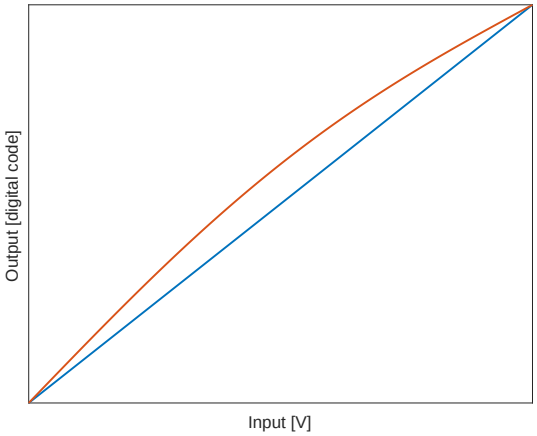


Figure 2.23: The figure shows the effect of INL on the transfer function connecting the input to the output of a converter. The blue line shows the ideal transfer function while the orange curve shows the case when the transfer function is affected by INL and the difference between these two curves is the INL.

$$INL\left(k\right) = \frac{X'\left(k\right) - k\Delta\left(1+G\right)}{\Delta} = \left(1+G\right)\sum_{i=1}^{k} DNL\left(i\right) \tag{2.28}$$

$$X'\left(k\right) = \Delta\left(1+G\right)\left(k_{os} + \sum_{1}^{k} DNL\left(i\right)\right) \tag{2.29}$$

Just as for DNL, INL is typically refereed to as the maximum $|INL\left(k\right)|$. The two measures share the unit of measure which in most cases is LSB however just like for DNL, it is also possible to use V, % or ppm.

2.11.6 Walden Figure of Merit (FoMW)

The FoMW is one of the commonly used combined measures. It overcomes the problem with SNR, SNDR and SFDR that they cannot be used to compare converter implementations of different topologies, resolution or sampling frequency ranges. The combined measures are intended to be used for comparing converters of different topologies, resolutions and sample frequencies.

The FoMW was formed by Robert Walden as a way of comparing the power dissipation for different types of converters with different resolution [30]. The initial FoM is given by equation (2.30) and is based on the resolution of the converter b, power dissipation P_{dis} and sample rate f_s.

$$FoM_w = \frac{2^b f_s}{P_{dis}} \tag{2.30}$$

The original FoMW has later evolved into equation (2.31) [31]. This equation differs in two ways from the original one presented by Walden. First, as can be seen in equation (2.31), it is the inverse of the original one. The reason for this is due to the convenience of talking about energy per conversion step instead of conversion step per energy. The other change however slightly changes the meaning of the FoM. This since it is based on the Effective-Number-of-Bits (ENOB) instead of the resolution of the converter. This means that the actual performance of the converter will be included in the FoM.

(a) Bulk CMOS (b) FD-SOI CMOS

Figure 2.24: Illustration of the transistor structure in a bulk CMOS process (a) and FD-SOI CMOS process (b). Inspired by [4].

Equation (2.32) shows the definition of ENOB based on a full-scale sine wave input signal. Despite that fact that the original FoM definition has been changed, it is still known under the same name.

$$FoM_w = \frac{P_{dis}}{2^{ENOB} f_s} \tag{2.31}$$

$$ENOB = \frac{SNDR - 1.76}{6.02} \tag{2.32}$$

FoMW is suitable for comparisons of converters which has a low resolution.

2.11.7 Schreier Figure of Merit (FoMS)

Another combined measure is the FoMS. This FoM shown in its original form in equation (2.33) which builds on the assumption that the power dissipated by the converter is determined by the thermal noise [32]. An alternative definition of this FoM is given in equation (2.34) where the SFDR has been changed for SNDR in order to include distortion [10]. This metric is more suitable for comparing high resolution converters.

$$\text{FoMS} = SFDR + 10\log_{10}\left(\frac{f_s}{2P_{dis}}\right) \tag{2.33}$$

$$\text{FoMS} = SNDR + 10\log_{10}\left(\frac{f_s}{2P_{dis}}\right) \tag{2.34}$$

Neither of these FoMs captures the whole picture and they should be used with care [31]. The reason is that the complex design tradeoffs regarding power and SNDR does not show up in these simple measures. It is also so that all FoMs are based on observations making them at best true for certain regions.

2.12 Technology node

The 28 nm Fully Depleted Silicon on Insulator (FD-SOI) Complementary Metal-Oxide Semiconductor (CMOS) process differs in the way transistors are implemented compared to standard bulk Si-CMOS technology. This can be seen in figure 2.24 where illustrations of the structure for both a bulk Si-CMOS transistor (a) and a FD-SOI CMOS transistor (b). It can here be seen that in the FD-SOI process the channel is isolated from the bulk by the use of a thin oxide layer.

The benefit with the FD-SOI process is that higher performance can be achieved while keeping leakage low [33]. This technology also reduces the input capacitance which helps to increase speed and decease power consumption. While the FD-SOI offers a reduced body effect as a result of the fully depleted body, it suffers from variations in the threshold voltage due to thickness variations in the thin oxide used [34]. Even though the FD-SOI process suffers from some problems, it has been deemed suitable for mixed-signal Radio frequency (RF) applications as the one designed in this project [35].

2.13 Related works

There exist a vast variety of different ADC implementations. The work presented by Kull et al. [9, 36–38] in their strive of reaching a sample rate of 100 GS/s are closely related to the work presented here, and then especially the unit cell presented in [36]. This since the Kull unit cell has been used as a reference design in this work. The unit cell builds on having two comparators in parallel with an alternating behaviour.

Other related works are the works presented by Lien [39], Mei et al. [40], and Wei et al. [41]. All these works have in common that they all builds on 2-bits per stage conversion.

The 8-bit converter presented by Lien [39] is based on a two stage pipeline structure with the 2 Most Significant Bit (MSB) fixed and the 6 LSB are converted at 2-bit-per-stage. The 2 MSB are used to reduce the number of reference levels required.

The 2-bit-per-stage converter presented by Mei et al. [40] is based on a reference capacitor array and a signal capacitor array in combination with three comparators. Two of the comparators used has four inputs to combine the signals from the reference and signal capacitor arrays.

The converter presented by Wei et al. [41] is based on 2-bit-per-stage conversion using three identical comparators. The design also uses a Kelvin divider rather than a CDAC for the reference voltage generation.

There also exist a vast number of works presenting TI based converters. These will however not be relevant for this work since it focuses on the design of a unit cell converter.

3
Method

This chapter will describe the different methodologies used followed by the work flow.

3.1 Methodologies used

This section presents the different methodologies used in this project.

3.1.1 Literature study

The project started with a literature study to gain insight into the field of high-speed data converters. The results from this study where then used to better understand how to translate the system requirements into component requirements and to identify components critical in terms of performance. These findings were then used to plan the work for the rest of the project.

3.1.2 Design process

After the literature study was completed, the design process started. This part combined further reading with simulations to evaluate different architectures for the different components making up the ADC.

The design started with the two most essential components in the converter, namely the comparator and CDAC. This since performance of these components sets the performance of the whole system. Also, the topology of these two components affects the requirements of the other components both in terms of required control signals and timing.

When all the different subcomponents were implemented, the process of adjusting the transistor sizes in the control logic started. The initial sizing was based on having a logic fan-out of four, since this is a commonly used trade-off between speed and power. The size of the transistors were then adjusted with the help of simulations. This adjustment were important, since some signal paths contain stages where the signal consists of a short pulse and therefore the load of each stage becomes even more important.

3.1.3 Verification process

Verification has partly been interleaved with the design process by verifying the functionality of key components such as the comparator and CDAC.

The process of verifying the functionality has also been a bit more complex due to the self-correcting behaviour of the circuit. The problem is that the redundancy which provides this self-correcting behaviour may also mask severe functionality problems with the circuit, thereby making it hard to find the actual root of the problem.

When verifying self-correcting hardware, it is even more important to characterise each component by itself before they are put together. This is to isolate potential problems with the different components which may be troublesome to find when they are all put together. Two such components which are important to characterise separately are the comparator and CDAC.

The comparator is important, since its offset voltage will be masked by the alternating comparator topology, while the kickback noise will be masked by the settling of the CDAC. For the CDAC, the linearity is the critical parameter to characterise, especially for all the alternative paths provided by the redundancy, since some alternative paths are not as likely to occur as others.

When the verification of the components has been performed, the verification of the whole system can start. Here it is important to remember that the calibration circuits used for the comparator as well as the reference voltage buffer introduce history effects which may result in different decision paths even though the input is the same. This means that the redundant encoding cannot directly be used to verify that a correct result has been delivered. Instead, the code needs to be converted into a non-redundant representation before any conclusions of the correctness in the output can be drawn.

3.2 Work flow

The project has been organized in a combination between linear and iterative planning. The stages, which mimic the main parts of the work, have been organized in a linear fashion. Each of these main stages consists of several tasks which has been performed in an iterative fashion.

Organizing these tasks in an iterative way makes it possible to make small improvements on the design based on the results from the verification.

As stated above, the work flow has been divided into three stages. This will be presented below together with the tasks that are included in them.

For a more detailed description of the project organization and work flow, see the time plan in appendix B.

3.2.1 Specification

During this stage of the project, the overall system specifications where split into specifications for each of the components. This stage consisted of three tasks: study potential implementations for the different components, analyse requirements for the components, and get started with the 28 nm process.

The study of potential implementations and the division of the components specification were two tasks which were tightly connected. To set reasonable requirements on the different components, different architectures needed to be analysed in order to gain insight into what performance could be achieved with each of the components.

The process of getting started with the 28 nm process was started here to gain insight in how the different tools worked. This was to make the start of the actual circuit design a bit simpler.

3.2.2 Circuit design

This stage of the project consisted of three tasks: circuit design, design of verification set-up, and circuit simulation.

Both the circuit design and the design of the verification set-ups were performed in a concurrent way. This was to make it possible to test the component functionality while it was implemented.

The task of simulating the different designs were iterated together with the design of the individual components to improve them based on the results from the design.

3.2.3 Circuit layout

The last stage in the project consists of moving the schematic-based design into the actual layout of the circuits. The layout consists of two tasks: doing the actual layout, and doing a post-layout simulation with the extracted parasitics from the layout added. These two tasks were iterated with the redesign task.

During redesign, improvements on the design was performed based on the results of the post-layout simulation. During this redesign, unexpected parasitic effects was taken care of in order to improve the design.

4

Specification

The specifications given in this chapter are intended as preliminary pointers of the required performance for each component during the design. This to have a starting point when starting with the design of each of the components.

The overall specifications for the system are given in table 4.1. These specifications are then used to derive the specifications for the components which make up the circuit.

A sampling rate of 800 MS/s was chosen as a balance which would make it possible to go for the high set SNDR goal of 44 dB for a 8-bit converter. The ADC power consumption is tightly connected to the sample rate and therefore a power consumption of 2 mW was considered reasonable.

The specifications for the timing constraints will be described below, followed by the specifications for the most critical parts, namely the comparator and CDAC.

4.1 Timing

The timing budget is essential to keep track of, since it affects the effort needed for each part in order for the system to work.

A duty cycle of the system clock being 1/8 with a sample rate of 800 MS/s gives 1.09 ns for the conversion. This time needs to be spread out over the different tasks which has to be performed. In table 4.2, the requirements for the most critical timing paths are given.

In the table it can be seen that the most critical paths in terms of timing are the clock generation and decision times, since they will be repeated several times during each conversion cycle. It will also be these paths that will require most attention during design, since a small improvement in one of these paths will be multiplied.

Table 4.1: Specifications for the whole system.

Parameter	Value	Comment
Sampling rate	> 800 MS/s	
Resolution	8-bits	
Input range	$0.5V_{pp}$	The input range is situated around a common mode input voltage
SNDR	> 44 dB	A tough goal with current state-of-art being around 40 dB [10].
INL	< 0.4 LSB	
Power consumption	< 2 mW	
Duty cycle clock	1/8	Sampling while active and converting while passive

Table 4.2: Timing budget for different parts of the converter.

Path		Value	Comment
From	To		
Ck	Ck_1	30 ps	Time from system clock falls to activations of comparator clock CK_1
Ck_1, Ck_2	Decision	60 ps	Duration for standard decision, $V_{diff} = 0.5$ LSB = 1 mV
Decision	Next Ck_1, Ck_2	45 ps	Time from decision to activation of next clock
Reset on	Reset off	100 ps	

Table 4.3: Specifications for the comparator.

Parameter	Value	Comment
Decision speed	< 60 ps	At differential input $V_{diff} = 0.5$ LSB = 1 mV
Reset time	< 50 ps	
Offset voltage	< 0.25 LSB = 0.5 mV	This indicates the standard deviation based on multiple runs
Kickback noise	< 0.5 LSB = 1 mV	Referees to the constant level shift and not to rapid transients
Common mode voltage	$0.5 - 0.6$ V	

The timing budget builds on achieving the case with standard decisions throughout the cycle and still be able to complete both reset and calibration. In case of a longer decision at some stage the time for the calibration will be reduced.

4.2 Comparator

The specifications set for the comparator are shown in table 4.3. Specifications have been distributed with the alternating comparator behaviour in mind which puts stricter requirements on offset voltage and kickback noise.

The requirements for the kickback noise and the offset voltage have been set based on their expected contribution to the noise when the alternating comparators are used. It is however not certain that these requirements will be enough to reduce their effect on the offset voltage.

The common-mode voltage has been chosen in such a way that the amount of kickback noise which needs to be managed is kept reasonably low while still achieving sufficient comparator speed.

4.3 CDAC

The specifications for the CDAC are given in table 4.4 and they mainly focus on the linearity and settling time. As can be seen in the table the INL for the CDAC has been set a bit stricter than the INL for the whole system has. This is since other parts of the system may also affect the linearity, such as the sampling switch and the alternating comparators with their offset voltage and kickback noise.

Table 4.4: Specifications for the CDAC.

Parameter	Value	Comment
DNL	< 0.25 LSB	
INL	< 0.3 LSB	
τ	$< 10\,\text{ps}$	For cycle switching in the SS corner to guarantee sufficient settling
τ	$< 15\,\text{ps}$	For CDAC switching in the SS corner during reset to guarantee sufficient settling
Reset time	$< 100\,\text{ps}$	Reset time for CDAC before calibration

When it comes to the τ times, it can be seen that the longer time has been allowed during reset; this is since the settling time during reset is not as critical as during conversion.

5
Design

This chapter will present some of the design choices made when designing the different components. The chapter will start with the whole design followed by the different components used. In the end different test supporting circuits will be presented together with purposed tests for evaluation of the design.

5.1 ADC topology

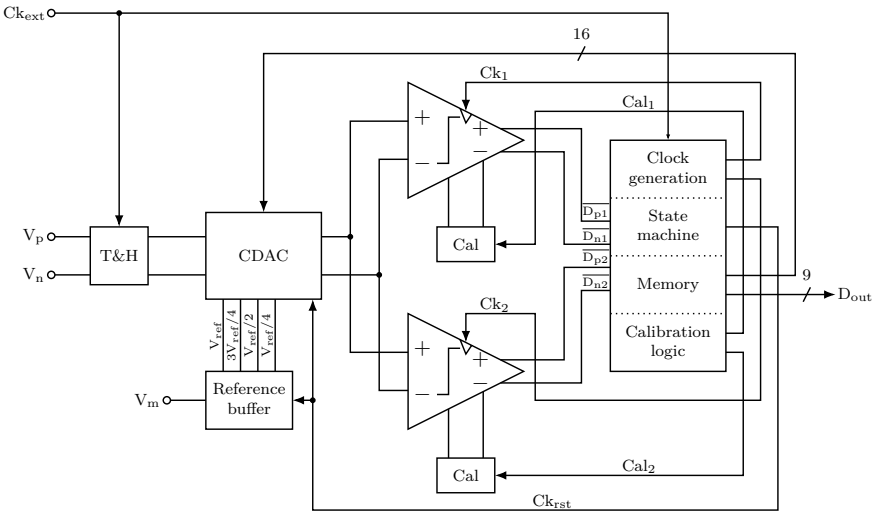
The design of the ADC starts with its overall topology and the components that are required for performing the operation. As already mentioned in section 2.4 there are five components that are essential for the operation of the converter, T&H switch, CDAC, comparator, SAR logic and reference voltage buffer. An illustration of the converter topology is shown in figure 5.1 where (a) shows the architecture and (b) shows the timing for the different clock signal used.

Each stage in the SAR converter can be based on either single-bit- or multi-bit-per-stage conversion. Multi-bit-per-stage conversion has the advantage that it minimizes the number of conversions in series by making some of them in parallel. This is highly beneficial if the settling time in the CDAC is long, since reduces the number of times at which it needs to occur. The settling time can however be decreased using redundant search algorithms as mentioned in section 5.3.1, single-bit-per-stage conversion has been chosen. As important as the CDAC settling time is the reset of the comparators, since incomplete reset results in a memory effect in the comparator, where previous decisions will affect upcoming ones. This means that a full reset of the comparator must be performed after each decision. To do this, either long time or large reset switches would be required. Large reset switches would contribute large amounts of parasitic capacitance, thereby slowing down the comparator.

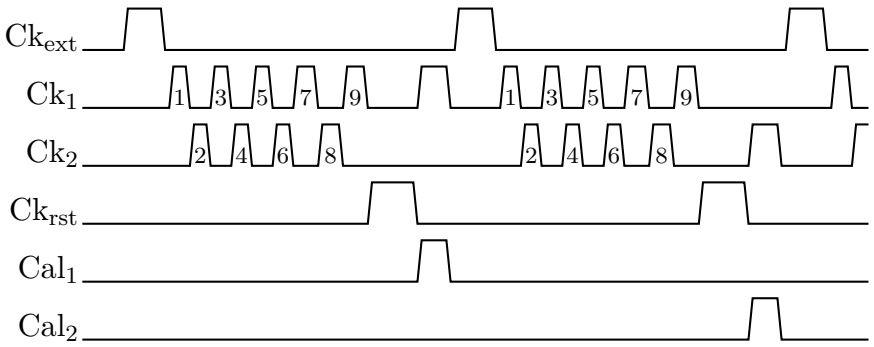
One way to remove the reset time from the critical path is to use alternating comparators, where one comparator is making a decision while the other performs reset. This way of alternating the comparators was originally presented in [1] and this method has been adopted here. The speed benefit however comes at the cost of a more complex SAR logic and in stricter requirements on the comparators.

5.2 Comparator

The comparator is together with the CDAC the most critical component in terms of performance. It decides the accuracy as well as speed of the converter and it is also a major contributor to the power consumption. The requirements on the comparator accuracy also increase due to the alternating operation, similar to the TI ADC architecture. Due to this



(a) Illustration of the SAR topology



(b) Timing diagram for the different clock signals used

Figure 5.1: Illustration of the components forming the converter topology (a) and the timing diagram for the different clock signals used (b).

component's contribution to both accuracy and speed, special attention has been paid when designing it.

The comparator design includes several aspects to be considered such as the topology, offset voltage management, and kickback noise reduction. These considerations will now be presented, starting with the topology.

5.2.1 Topology

The converter topology has a large effect on both the speed and accuracy that can be achieved as well as on the power consumption. To minimize the power consumption while still achieving high speed with good accuracy, a latch-based comparator was chosen. This is since such comparators do not have any direct path from supply to ground that constantly draws power.

Two comparator topologies where analysed: the strong-ARM latch, shown in figure 2.8, used by [1,22] and the dynamic latch, shown in figure 2.9, used by [2,19]. The strong-ARM comparator has the advantage of a lower transistor count than the dynamic latch. This reduces the power consumption and increases the speed due to the single stage behaviour. A disadvantage with this topology is that since the input transistor pair needs to directly drive the latch they need to be large which increases effects of kickback noise. The larger transistors required will however improve the matching between the transistors.

The dynamic latch on the other hand has the advantage that since the input pair is not directly driving the latch, which makes it possible to have a smaller input transistor pair without reduced speed. The disadvantage with this topology is that the stage behaviour decreases the speed since the signal needs to propagate through several stages before reaching the latch. Another disadvantage is that the increased number of transistors required increases the power consumption.

The strong-ARM latch topology was chosen for the converter. This was mainly due to its speed advantage in combination with a lower power consumption. When it comes to matching and kickback noise, it was considered that compensating methodologies where required in both cases and that these would have similar complexity, making them not affecting the decision.

The different output combinations for the comparator are listed together with explanation about their meaning in table 5.1. The output names are the one given in figure 2.8. The comparator has three valid outputs: [1,1] indicating that no decision has been taken, and [0,1] or [1,0] indicating that a decision has been taken. The output [0,0] is however a non-valid one indicating incorrect operation of the comparator.

5.2.2 Offset voltage

Offset voltage is an effect caused by random variations in the transistor properties which affect their parasitic capacitance, current factor, and threshold voltage, as mentioned in section 2.8.1. Layout asymmetries will also contribute to the offset voltage, since the parasitic capacitances will vary.

With alternating comparators, the problem with the offset voltage will become worse, since it will now be a varying effect rather then a constant offset. In the TI case, the offset voltage varies between the conversions, making the problem being an undesired frequency component. In the case with the alternating comparators on the other hand, the problem will be worse since the offset variations will now be within one conversion instead of between conversions. This means that a large offset difference between the

Table 5.1: Explanation of the meaning of the different comparator outputs. The output names correspond to the ones in figure 2.8.

Output $\overline{P}$	Output $\overline{N}$	Explanation
1	1	Comparator in reset, no decision made
1	0	Input to negative leg larger then input to positive leg
0	1	Input to positive leg larger then input to negative leg
0	0	Non-valid output, comparator operating incorrectly

converters may result in totally inaccurate results. Some of these erroneous decisions may be covered by the use of redundancy at the decision levels.

The offset voltage gets reduced by increasing the sizes of the input transistors, which will improve matching. Just making the transistors larger will however come with other problems and therefore other techniques to minimize the offset voltage needs to adopted.

5.2.3 Calibration

To reduce the offset voltage effects, an offset voltage calibration circuit will be required. This circuit not only needs to cover the static effects caused by transistor mismatch in the comparator itself; it also needs to manage dynamic effects such as the body memory effect that the Silicon on Insulator (SOI) CMOS process suffers from. This means that a dynamic calibration circuit will be required.

In section 2.9, several different offset voltage calibration techniques were presented. One of them was digital post processing. This technique will however not work in this case, since it only manages static errors. All the other methods presented in section 2.9 are however suitable candidates.

The body biasing offset compensation method has the advantage that it does not add any additional capacitance in nodes $p1$ and $n1$. The problem with this method is that the resolution of DAC that produces the biasing voltage limits the ability to correct offsets. To limit the complexity in the DAC this method is typically used only for coarse calibration [25, 26]. This means that to achieve a small offset after calibration, another calibration technique must also be used. This however introduces the problem of having overlapping in between the two calibration methods so that there would not be any gaps in the calibration. The complexity of the calibration circuit will also increase with the use of two different methodologies. This calibration methodology has not been chosen due to its limited ability to do fine correction of offset voltages.

Another promising calibration method is the switched capacitor method which has the advantage that it could cover large ranges of offset voltage. The problem with this method is that while compensating for offset voltage, additional capacitance is added to the nodes $p1$ and $n1$, in addition to this will the switches themselves introduce parasitic capacitances, and thereby the speed of the comparator is decreased. The minimum implementable capacitance limits the achievable calibration accuracy. This calibration method has not been chosen due to the fact that it may significantly reduce the speed of the comparator when offset compensated.

That only leaves the current injection compensation method, which can be implemented in two ways. The advantage with this calibration method is that it will increase the speed of the comparator rather then slowing it down, which was the case for the switched capacitor method. This is due to the fact that current is injected in the slow leg thereby increasing its speed. The digitally controlled method has the problem that the minimum transistor size limits the ability to correct small offset voltages. It is however a compact calibration method. The switched-charge-controlled method has potential for very fine calibration of the offset voltage, and the resolution is essentially determined by controlling the amount of charge being added or removed from the storage capacitor.

The switch charge control method has been chosen due to its ability to correct small offset voltages in combination with its potential for a compact calibration control.

5.2.4 Kickback noise

Kickback noise is another effect which limits the performance of the converter as mentioned in section 2.10. The reason is that it causes voltage variations on the sampling capacitance. These voltage variations consists of both a common mode and a differential mode component. The common mode component comes from the common mode voltage used while the differential mode component comes from the amplitude of the signal feed into the comparator.

In the case where only the comparator affects the voltage on the sampling capacitance, that is no CDAC compensating for the result, the effect of the kickback noise will cancel it self out when the comparator has been fully reset. The problem however arises when this is not the case, such as when the sampled voltage is changed by the CDAC; then, parts of the kickback noise will not be cancelled out since conditions have changed. The problem also shows up when multiple comparators are alternating, since this technique builds on performing the reset while another comparator performs a comparison.

With the combination of alternating comparators and a CDAC that both cause variations on the sampled voltage, parts of the differential kickback will not be cancelled, resulting in a unwanted offset which may affect upcoming decisions and cause erroneous results. This means that some type of kickback reducing circuit will be required.

5.2.5 Kickback noise reduction

Two potential techniques for reducing the kickback noise have been investigated. These techniques are mentioned in section 2.10.1.

The re-sampling method essentially builds on re-sampling the voltage on the sampling capacitor onto another capacitance, in this case onto the input capacitance on the comparator. It is important that the capacitance onto which the voltage is re-sampled is sufficiently large, so that kT/C noise will not be a factor. This method has the advantage that it separates the comparator from the sampling capacitance, making the kickback noise only affect the voltage sampled onto the input. A problem with this method is that whereas no kickback noise will come from the comparator, cross-talk from the clock controlling the sample switch will leak into the sampling capacitance. This cross-talk may be smaller than the kickback noise from the comparator how ever it depends on the size of the re-sampling switches. Another problem with this method is that to remove the kickback effects completely, it is important that the re-sampling capacitance is completely reset. Resetting this capacitance means that each time re-sampling takes place, charge will be drawn from the sampling capacitance, thereby reducing the voltage stored on it. A problem that arises for the alternating comparator topology is the time during which

re-sampling must take place. It is important to not have the sampling switch open while the comparator makes decisions, but it should still be sampling while the CDAC is settling. Due to all these complications that become worse due to the alternating comparator topology, this method for reducing the kickback noise has not been chosen.

Instead, the method building on switched capacitance was chosen. It builds on switching a dummy transistor, connected with both drain and source to the node which is affected by the leakage, in the opposite way to the signal causing the leakage and thereby letting the parasitic capacitance cause a similar effect in the opposite direction. This method has the advantage that it eliminates all the problem related to re-sampling. The problem with this technique is to match the leakage from the dummy transistor to the inverted leakage from the source. This is since the capacitances related to the transistor are not constant, rather they are highly dependent on the voltage applied [28, 42]. Another related problem is that the amount of leakage depends on the rate at which the signal rises or falls [43].

5.3 CDAC

Among with the comparator the CDAC is the most important components. This is since it not only functions as a sampling capacitance but also provides the decision levels with its charge redistribution functionality which makes the linearity a concern. The CDAC is also a large contributor to the footprint of the design, as well as to the power consumption, making the design of it require special attention.

The design of the CDAC involves several things that need to be considered: the scaling, overall topology, unit topology, sampling noise, and reference voltage adjustment for the charge redistribution process. All these considerations will be presented below.

5.3.1 Scaling

The scaling of the CDAC can be implemented in several different ways with or without redundancy. In section 2.6.4, a comparison of the advantages and disadvantages is given for the three different scaling methods: binary, redundant non-binary, and compensation. Binary scaling has the advantage that it requires the minimum number of compared to the redundant methods which needs additional stages. The problem with binary scaling is the required settling time to get a small settling error. It also suffers from the problem that an erroneous decision will result in a erroneous output.

All redundant scaling algorithms have the same advantage over the binary scaling: they can compensate for erroneous decisions cased by incomplete settling. This means that they can handle larger settling errors, thereby making it possible to utilize partial settling. They all share the same drawback of requiring additional comparisons. The number of additional comparisons depends on the amount of errors to be covered.

To gain speed with a redundant scaling of the CDAC, it is required that the time it takes to perform the additional comparisons are shorter than the time gained by having a smaller t/τ. The clock generation, which decides when a new comparison should take place, will not adapt to the individual settling time at each stage. The clock generation circuit will rather be implemented so that sufficient settling is achieved for the stage with the longest settling time. This means that redundancy will be most efficient at the first stages, in order to reduce the settling time at these stages and thereby increase the speed. The redundancy helps even out the settling time at the different stages, making it more even over all stages, as can be seen in table 5.3 where the settling time is shown for both the binary case and for two different configurations of the generic non-binary case.

Out of the redundant scaling methods, the generic non-binary was chosen over both the conventional non-binary and compensation. This was due to the fact that the generic method offered more flexibility in where to place redundancy. As mentioned in section 2.6.4, when comparing the different scaling methods, the conventional non-binary method suffers from having a fixed radix throughout all stages. This typically results in capacitance ratios that are hard to realize. To solve this, either fine fractional capacitance would be required or the radix must be slightly tuned to be able to use multiples of a unit capacitance. This however changes how the redundancy will be spread throughout the stages, potentially resulting in significantly less redundancy than expected at a stage. The compensation method suffered from the fact that to get sufficient redundancy, more than one additional stage would be required, thereby reducing the benefit of having it there to reduce settling time and increase speed.

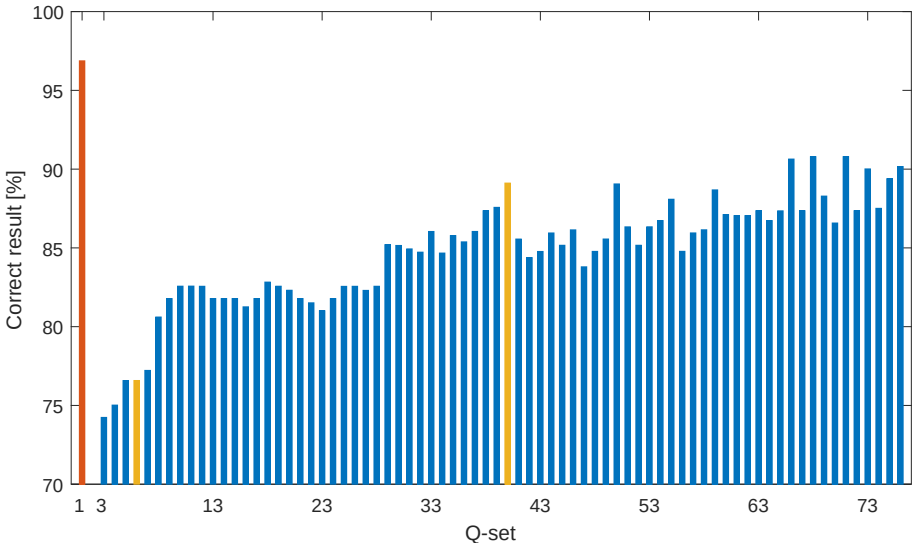
When deciding where to place the redundancy, there are several factors which affects the decision. One place where redundancy is useless is at the first stage, since the decision taken at this stage is taken on a completely settled CDAC. This means that no settling errors will affect the decision. Another place at which the effect of redundancy is limited are at the last stages. This is since the effect of an erroneous decision at one of these stage will have a limited effect on the output. It is also so that the required settling time is small at these stages compared to the requirement for the first stages which means that no speed improvement would be achieved.

To analyse suitable scalings for use in the CDAC, all valid q vectors (fulfilling the condition given in equation (2.19) and only having non-negative integer values) were analysed. From each q vector, the corresponding p vector and t/τ vector were calculated. To reduce the number of potential scalings a bit, a few limitations were introduced. First, as mentioned above, redundancy at the first stage is useless and therefore all cases where this occurs have been removed. To reduce the set even further, p vectors containing odd $p\,(i)$ were removed, with the exception for $p\,(M)$ which always is one. The reason for doing this last sorting is to make it possible to use fractional reference voltage to reduce the total capacitance without affecting the common mode voltage [1]. Finally, all scalings which resulted in a maximum t/τ larger than 3 were removed since the speed-up for scaling with a larger maximum t/τ would be limited. All the scalings that were left after this last separation are given in appendix A.

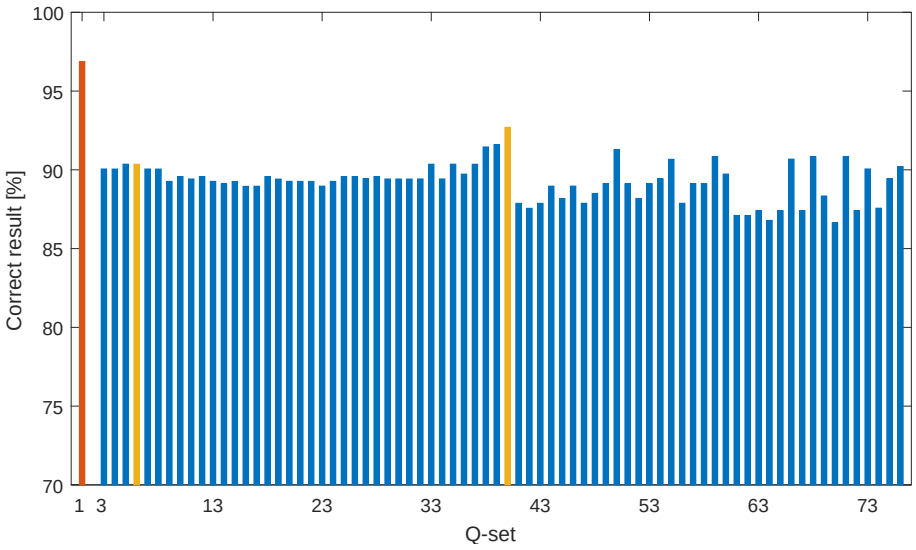
From the scalings that where left after the reduction, a comparison of the correctness in the output where checked for both the maximum individual t/τ as well as for $t/\tau = 3$. These results were then compared to the binary case. The correctness were simulated by feeding uniformly distributed random inputs to a MATLAB model of the charge redistribution functionality where decisions where taken at partial settling. The output from the partial settling was then compared with the output gained at operation at full settling. In both cases, the binary scaling was used for comparison with its maximum t/τ.

Based on the results of correctness evaluation, two different scalings were chosen for circuit implementation. These two scalings are shown in table 5.2; they will here after be known as *Generic 1* and *Generic 2*. In table 5.3 t/τ is given for all stages with both binary scaling and for the two redundant non-binary ones, *Generic 1* and *Generic 2*. The reason for not directly going for a specific scaling is that the linearity of the implementation may depend on the scaling if fractional reference voltages are used.

The settling times presented in table 5.3 have been calculated based on the case where the settling error is less then 0.5 LSB plus redundancy, if any. The calculations have been performed using equation (5.1), where SE is the settling error allowed, in this case 0.5 LSB, while $q\,(i)$ and $p\,(i+1)$ are corresponding values in the p and q vectors.



(a) Maximum individual t/τ



(b) $t/\tau = 3$

Figure 5.2: Evaluation of the output correctness for a set of scalings at each scalings individual maximum t/τ (a) and for $t/\tau = 3$ (b). The scaling vectors connected to each bar is given in appendix A.

Table 5.2: p and q values of the two scaling configuration chosen for further analysis.

Scaling	q									p								
	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9
Generic 1	0	8	4	4	4	0	0	0	0	128	64	28	16	8	4	4	2	1
Generic 2	0	4	2	2	2	2	0	0	0	128	64	30	16	8	4	2	2	1

Table 5.3: t/τ at each stage for binary scaling as well as two different configurations of the generic non-binary scaling. The settling times have been calculated for a settling error of 0.5 LSB plus redundancy if any.

Scaling	Stage							
	1	2	3	4	5	6	7	8
Binary	4.852	4.159	3.466	2.773	2.079	1.386	0.693	-
Generic 1	2.019	1.828	1.269	0.575	2.079	2.079	1.386	0.693
Generic 2	2.654	2.484	1.856	1.163	0.47	1.386	1.386	0.693

$$t/\tau(i) = -\ln\left(\frac{SE + q\,(i)}{p\,(i+1)}\right) \tag{5.1}$$

5.3.2 Topology

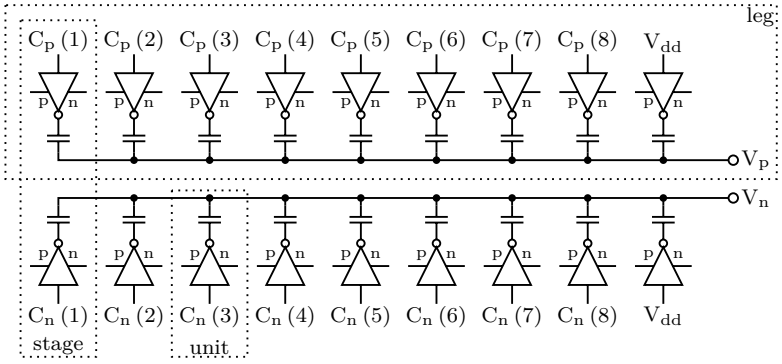
The topology of the CDAC can be divided into two parts, where one covers the whole CDAC implementation and the other one which shows what the different stages will look like. First, the overall topology will be described, followed by the topology of each stage.

5.3.2.1 Overall topology

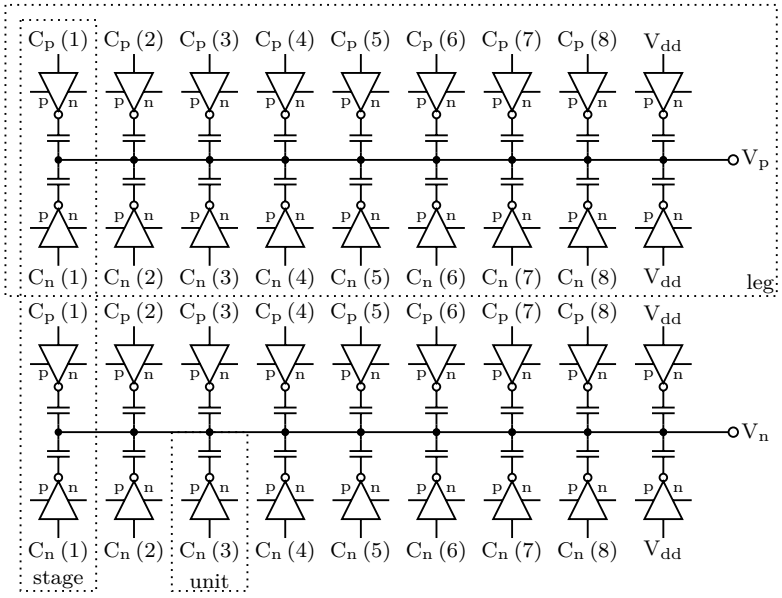
The topology of the CDAC can be divided into several levels where each level is connected to a property in the operation of the CDAC. These levels may be designated *leg*, *stage*, and *unit* and they will all be defined below. Starting with the leg, which corresponds the inputs of the comparator where the single ended case, there is only one, while the differential case has two. Essentially, the differential topology can be seen as a duplication of the single ended case, with the difference that the switching is performed in an inverted fashion in one of the two legs compared to the other one.

The leg can then be constructed in two ways either balanced or double balanced as shown in figure 5.3. The difference between having a balanced or double balanced structure is that in the balanced case, only one of the legs will be switched, up or down depending on the preset voltage. The effect of only switching one of the legs is that the common mode voltage will vary depending on the input as well as on the previous decisions. This results in large variations in the operating condition for the comparator, since it needs to operate over a large range of common mode voltages without losing performance both in terms of speed, offset voltage, and kickback noise. These common-mode variations do not show up when the double balanced array is used, since switching will occur in both legs at each stage and pull the legs in opposite directions. For example, if switching pulls the voltage in the top leg up will the voltage in the bottom leg be pulled down an equal amount, resulting in a constant common-mode voltage. A disadvantage with the double balanced is that it requires twice as many switches as well as twice as much capacitance compared to the balanced case. The doubled capacitance however only occurs when the implementation uses the smallest capacitors and is not kT/C noise limited.

The CDAC can also be divided into stages. A stage consists of all units that can be controlled by the result of one comparison. In the double balanced structure there are two units in each leg and two legs in total which means that there will be 4 units per stage. There is one stage for each comparison that will take place except for the last one, and all of them are switchable. These stages can be recognized by the control signals that are connected to them, and their number indicates the stage number. These switchable



(a) Balanced capacitor array



(b) Double balanced capacitor array

Figure 5.3: The CDAC topology including its redundant scaling and the reference voltages input to each stage. The topology is shown for a single ended array in (a) and a differential array in (b).

stages are responsible for creating the voltage changes, and their sizes determine how much the voltage is changed. The size of the capacitance at each of these stages is defined by the scaling in combination with the unit capacitance used. Except for these 8 switchable stages, there is also one fixed stage that has the same size as the last stage. This stage becomes obvious in the binary case, where the sum of the capacitances in the remaining stages is equal to the total capacitance at the current stage. Without this stage, the total capacitance after the current stage would not be equal to the capacitance at the current stage thereby resulting in a voltage change larger then half the range of the current stage. The principle is the same when redundant scaling is used, however the principle that the current stage will have the same capacitance as the sum of the stages following will not be valid at stages which have redundancy. The last stage can be seen as the capacitance which would be split into smaller fractions if higher resolution were used.

The reason that there is only 8 switchable stages are required even if 9 decisions are taken is that the first comparison level located at half the range is provided by the differential behaviour of the input. This is also the reason that there is no requirement of performing any switching prior to the first decision.

With a converter resolution of 8 bits, the ratio between the capacitance at the first and last stage becomes large. This ratio can however be reduced by only switching parts of the charge stored in the unit capacitance. The implementation of the fractional charge switching builds on switching fractional voltages rather then the full reference voltage swing [1]. The principle builds on that the charge stored in a capacitor is linearly dependent on both the capacitance as well as the voltage which can be seen in equation (5.2), so minimizing one of them reduces the charge switched. Typically this is implemented by reducing the size of the capacitance, but the same effect can be achieved with fractional reference voltages. In theory, this method can be used to implement any fractional capacitance; however, this will make the linearity of the fractional voltage generator critical. If a Kelvin divider is used will the load on the output affect the fractional voltage generation. Common-mode variations can also arise from the use of fractional voltages [1]. There is one fractional voltage that will not have effects on the common mode voltage and that is $V_{ref}/2$.

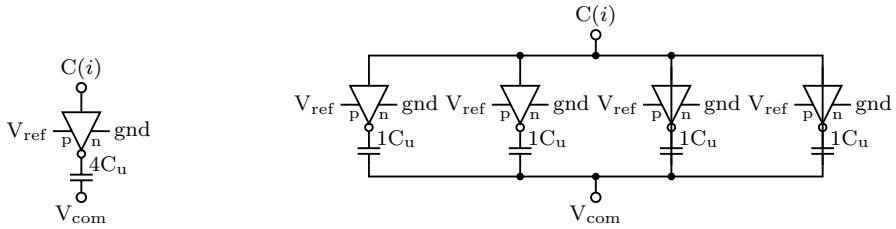
$$Q = CV \tag{5.2}$$

Using half the reference voltage for generating fractional capacitances makes it possible to reduce the capacitance ratio between the first and last stage by a factor of 2. To reduce this ratio even further, more fractional voltages are required. This however comes at the cost of potential common mode voltage variations and therefore the use of fractional voltages has been reduced to $V_{ref}/4$, $V_{ref}/2$, and $3V_{ref}/4$, where $V_{ref}/4$ and $3V_{ref}/4$ are used only at the last stage, making variations on the common mode voltage only affect the LSB.

5.3.2.2 Unit topology

The unit is what is building each stage of the CDAC and it can be constructed in two different ways, either by using a single large cell containing a single capacitance or by using smaller unit cells that are placed together to form a larger unit. In both cases, the circuit out of which the cells are realized is the same, with the only difference that one is unit sized while the other one is stage sized.

The unit is based on an inverter which is connected to the unit's capacitance. The inverter is then fed with the two reference voltages, most commonly V_{ref} and ground. The



(a) Single cell (b) Unit cell

Figure 5.4: The unit based on a single cell (a) and on unit cells (b) in the case where the capacitance should be $4C$.

node on the capacitance that is not connected to the inverter is connected to the other units as well as to both the T&H and comparator input as can be seen in figure 5.3.

The single cell structure consists of a single capacitor of the desired size together with a single inverter sized in proportion to the capacitor. An illustration of the single cell structure is shown in figure 5.4 (a) for the cell size of 4. The benefit with this topology is that any radix can be formed as long as the capacitance ratio can be physically implemented. The disadvantage is that it is more vulnerable to random mismatch variations than the unit cell topology. It is also more complex to design this topology since a unique cell will be required at each stage.

The unit cell topology on the other hand builds on having a unit sized capacitor and inverter. The desired size is then achieved by placing several cells in parallel. An illustration of the unit cell structure is shown in figure 5.4 (b) for a total size of 4. The problem with this topology is that only radixes based on multiples of the unit can be achieved. Some fractional based radixes can however be implemented by the use of fractional reference voltages. The advantage with this topology is that it is not as vulnerable to mismatch variations as the unit based topology. Since mismatch will make the capacitance at each cell either smaller or larger, putting several cells in parallel thereby typically results in a smaller total variation for the stage. The averaging can not be guaranteed since the variations are random, but statistically the likelihood of having a large mismatch when using unit cells is much smaller then in the single unit case. Another advantage with the unit cell topology is that only one cell needs to be implemented.

The unit cell topology has been chosen here, since it typically minimizes the effects of random mismatch variations which will help improve the linearity of the CDAC. Another reason for choosing this topology is that it simplifies the design since only the unit cell needs to be verified compared to the single cell structure where each cell needs to be verified. Verification here means checking that the settling time requirements are fulfilled.

5.3.3 Sampling noise

The sampling noise stored into the sampling capacitance is determined by the size of the capacitance according to equation (2.8) for the single ended case. In the differential case it is sufficient to use only half the equivalent single ended capacitance for each leg achieve the same SNR, as described in section 2.1.1.2.

To reduce the effects of kT/C noise, it is important that a sufficiently large capacitance is chosen. How large the contribution of the kT/C noise is allowed to be is a design parameter. A typical start value is to use a noise power being 3 dB below the quantization noise.

A full scale input of 0.5 V and a resolution of 8 bits gives a total sampling capacitance as in equation (5.3) given a temperature of 300 K. From this it is possible to calculate the minimum sampling capacitances in the differential case, which results in $C_{s,d} = 13\,\text{fF}$ at a kT/C noise level 3 dB below the quantization noise.

$$C_s = \frac{kT}{V_{n,C}^2} = \frac{24 \cdot 2^{2N} kT}{V_{FS}^2} = \frac{24 \cdot 2^{16} \cdot 1.38 \cdot 10^{-23} \cdot 300}{0.5^2} = 26\,\text{fF} \tag{5.3}$$

5.3.4 Reference voltage adjustment

The principle of charge redistribution builds on switching capacitors between different reference voltages and thereby controlling the charge stored in the CDAC. The reference voltage range over which the capacitor is switched should ideally be half the dynamic range, that is $V_{pp}/2$ in case of the dual half topology, where the two legs are switched in opposite directions. This ideal case however assumes that all capacitance in the CDAC is switchable. This is however not the case, since parasitic capacitances are connected to the switches themselves, the reset switch, the T&H switch, and the comparator inputs will also contribute to the total CDAC capacitance [1]. This means that the switchable capacitance will only be a fraction of the total capacitance.

To overcome this problem and still achieve the desired dynamic range, a higher reference voltage can be used. The new reference voltage can be calculated using equation (5.4) where V_{ref_c} is the new reference voltage and V_{ref} is the initial one. $C_{u,total}$ is the total capacitance per unit cell and $C_{u,switch}$ is the part that is switchable.

$$V_{ref_c} = V_{ref} \frac{C_{u,total}}{C_{u,switch}} \tag{5.4}$$

The relationship can be used as an estimate for what reference voltage that should be used. The exact voltage can however not be determined before fabrication, since random mismatch affect both the parasitic capacitance as well as the switchable part. The relationship can also be used to calculate the ratio between the switchable and parasitic parts of the capacitance after fabrication. An exact value for the parasitic contribution can however not be calculated, since the exact switchable part is not known.

5.4 SAR logic

The SAR logic is the part which controls the operation of the converter. The functionality of this logic will vary depending on how the SAR algorithm has been implemented and what types of additional circuits that has been added. The SAR logic implementation also depends on what type of control principle that is used.

The state transition control can be performed in two different ways: synchronous or asynchronous. As mentioned in section 2.5.1, synchronous control has the advantage that the time of when decisions are taken is known beforehand. This is rely convenient during design since it simplifies debugging. The problem is however that the decision time depends on the voltage applied to the input of the comparator. This means that an exact decision time can not be known. It is also not known at which stage a longer decision will occur which means that margins need to be added at each stage to handle cases where the decision time is longer than the standard decision. This means that each stage must have time which is not used just to be sure that there will be time for most of the slower decisions.

Asynchronous control on the other hand builds on having the state transitions depending on the results of the operation. The advantage is that now the results will be used to start the next stage rather than it being started at a specific point in time. This means that a margin only needs to be added at the end of the cycle, instead of at each stage. The disadvantage is however that it is more complex to implement and debug asynchronous logic.

Asynchronous control has been chosen here since it only requires margin in time to cover one slow stage per cycle. With synchronous control, it would not have been possible to retch the desired speed due the the margins that are required at each stage.

The logic implementation used in this case can be divided into three main parts: comparator clock generator, state machine, and calibration control logic. This division has been done since the dependencies between the different functions are minimal and for increased clarity.

5.4.1 Clock generation

The clock generation circuit is responsible for generating the clock signals which start a comparison as well as reset the comparator afterwards. This block is also responsible for generating the clock signals used to control the kickback reduction circuit in the comparator.

Since asynchronous control has been chosen, the clock generation circuit must operate based on the results from the comparisons. The circuit must also have a short delay from detected decision to outputted clock, since this delay has a large effect on the speed at which the converter can run. The main idea for the clock generation circuit comes from the circuit used by [1]. It is based on detecting when a decision has been completed and using that detection signal to turn off the currently active clock at the same time as the next clock is activated.

5.4.2 State machine

The state machine keeps track of how many decisions have been taken. It is also responsible for generating different control signals that are used to stop the clock generation and reset the different parts of the circuit. These control functionalities can be implemented in several different ways.

One way would be to have a state machine that consists on state memories and logic that determines the next stage. This is also the conventional way to implement a state machine. The problem here is that while the sampling rate is 800 MS/s, the state machine itself would need to operate at speeds exceeding 10 GHz. The reason for this high speed requirement on the state machine is that each stage corresponds to a state as well as reset and calibration. In addition, it will not be possible to use the entire time from one falling edge of the sampling clock to another one, since sampling must take place between these two falling edges, reducing the operation time for the logic.

To eliminate the need for a state machine operating at 10 GHz, an alternative method has been chosen. This builds on using the results stored in memory with combinational logic to control the state transitions. State transition is activated by detecting if a decision has been stored. This however requires having both the comparator outputs $\overline{P}$ and $\overline{N}$, shown is figure figure 2.8, stored but this will not be a problem since both signals will be required for controlling the CDAC. The detection of a decision generates the write-enable signal for the next memory cell, thereby propagating the state forward. The benefit of using this method is that the state transition delay is reduced. Another benefit comes

from the use of the alternating comparators and how they are connected to the memory cells. By only connecting the comparators to the memory cells that correspond to the states at which they are supposed to deliver results for. This not only relaxes the timing constraints, but also helps preventing skipped state caused by the activation of the next write enable before reset is performed, as well as preventing that the result from the next stage overwrites the stored result due to not turning of the write enable signal at the memory early enough.

An alternative where the comparators where connected to all memory cells was also analysed. This would have the benefit that it would be possible to alternate the comparator that starts a cycle. This was thought to be beneficial in terms of optimizing the reset and calibration by performing them at the same time and still guarantee that the comparator being calibrated had been fully reset. This is done by calibrating the comparator which did not preform the last conversion. However, it was discovered that calibration and reset would not be possible to perform at the same time. This is due to the fact that it would have resulted in a calibration which were not performed under the same conditions as under normal operation. This method would also have made it more complex to guarantee that new results would not affect previous ones or cause a skipped state due to a delayed reset. Thus, the alternative with dedicated memory cells was chosen over the one where both comparators where connected to all memory cells.

The state transition will be initiated by the sampling clock and will then ripple along after each detected decision. It is also important that the memories are completely reset before a new cycle is initiated, since the state transition would not work otherwise. The reset must be performed even if all stages has not been passed due to long decision times.

5.4.2.1 Memory

To implement the state machine around the memories that stores the results, it is important that the delay through the memory in minimized. The memory must also have two separate outputs which can be reset separately. This is important since the output which controls the switching in the CDAC should be reset after the last decision, while the memory contents should be reset during sampling so that it is possible to feed the result forward to the output. The load which the memory puts on the comparator must be small while it must provide driving capabilities for driving the switches in the CDAC.

The memory cell is based on the cell used in [1]. This since it provides low latency at the same time as individual resets are provided. The latency for the CDAC control signal output is especially short, which reduces the time from decision to a sufficiently settled CDAC.

5.4.3 Calibration control logic

The calibration logic has been added due to the comparator offset calibration requirement. This circuit is responsible for keeping track of which comparator is to be calibrated, since only one comparator will be calibrated in each cycle. This circuit will also be responsible for the comparator clock initiation. The reason for placing it here is that it is tightly connected to the calibration clock generation.

The calibration logic is designed to always change the comparator which will be calibrated even if the calibration of the other comparator is not completed. Changing comparator each cycle has been chosen since it is not required to have a calibration every cycle and if one calibration is missed it is not that severe. A calibration logic which keeps track of if the calibration is completed becomes much more complex.

5.5 Track and Hold

The T&H switch can be designed in several different way with more or less complex solutions to improve the linearity of it. One T&H circuit that has been analysed is presented by [28] and it builds on bootstrapping the sampling switch to increase the linearity of it while eliminating the clock leakage. The problem with the bootstrapping topology is that it requires multiple control signals for various control switches making the implementation complex when speed increases.

The T&H switch will instead be based on a simple combination of NMOS and PMOS switches placed in parallel. These will be combined with clock leakage reduction method inspired by [44]. This implementation has the advantage that it only requires a single clock signal. The clock signal will however need to be differential since both NMOS and PMOS switches are used. The reason for using a complementary switch is that it improves the linearity by providing a conductivity that is constant over a larger range than for a single NMOS or PMOS [11].

5.6 Reference voltage buffer

The reference voltage buffer is the component which is responsible for providing the reference voltage as well as the fractions of it to the CDAC. The reason for having this buffer on the chip is to guarantee that sufficient amount of charge can be delivered to the CDAC when it switches. This to make the settling depend on the inverter driver size in combination with the size of the unit cell and not on the ability to provide charge. Another reason is that the switching occur during such short amount of time that it would be problematic to deliver the charge from outside the chip. Another reason for having the reference voltage buffer is that the converter is designed to be a unit cell that can be used multiple times either in parallel by TI or at different parts of the circuit. In both cases, it is important to be able to control the reference voltage individually to each converter used, since the parasitic capacitance at each cell will affect the dynamic range of that specific converter.

The reference voltage buffer is based on feeding an external master voltage onto the chip. The reason for using an external signal is that the reference voltage needs to be set in such a way that it can be controlled and set to achieve the desired dynamic range. That the exact relationship between the swichable and parasitic parts of the sampling capacitance is not known, thereby making the exact range unknown.

Since multiple voltage should be delivered to the CDAC, the option of using a single or multiple buffers arises. The advantage with using multiple devices is that there will be no load on the net that provides the fractional master voltages, thereby not making it load dependent. The problem is however that there will be mismatch between the different comparators, introducing differences in the fractional voltages which may be large enough to cause significant errors. When multiple buffers are used each of them also needs to be sized according to the amount of charge that will be drawn from it.

Instead, a single reference voltage buffer will be used, from which the fractional voltages are generated using a resistive network. This has the advantage that only a single comparator is required, thereby making its offset voltage affect all the reference voltages in the same way. The disadvantage with this buffer alternative is that when charge is drawn through the resistive network that generates the fractional voltages, their value will change and this may have an effect on the linearity of the CDAC.

5.7 Design for test

Testing and verification of a design is just as important as the design itself. This is especially true when designing an Application Specific Integrated Circuit (ASIC) since actual performance depends on how the process variations has affected the design. To verify that the actual performance is fulfilled, tests needs to be performed on the fabricated chip. The problem however arises with blocks which are not designed for stand alone use, but only as part of a larger design.

It is also important that the test set-up does not affect the design itself. This is especially true for ADCs as well as DACs, since they operate with small analog voltages and leakage from external control signals such as the clock may have severe effect on the performance. To overcome this problem, additional circuitry will be required to support the operation of the design. In this case two supporting circuits will be used; a clock generator and an output buffer.

Supporting circuits are however not everything that needs to be considered when designing a prototype ASIC. There is always the risk of having parts of the design not working due to some bug or state that was missed during the design or due to manufacturing flaws. However, due to the high cost of manufacturing ASICs in small series, it becomes important to make sure that as much information can be gained from the design as possible.

By making it possible to turn off components in the design which is not critical for the core operation, the likelihood of getting some data increases. Sending key signals off chip is also method that can be used to verify which part of the circuit that works or not. This must however be thought of during the design phase since when the chip is fabricated it will be to late add these signals. The problem is however that pads are limited and having many additional control signals for controlling the activation of various functionalities or sending out many state indication signals will not be possible.

5.7.1 System clock generator

A system clock generator was considered required since cross-talk from a full swing digital clock may leak onto the analog input. Having short rise and fall times was also considered complicated with an external system clock feed. This is since the rise and fall times will have effect on the size of the sampling jitter.

Another thing that also complicates the clock generation is that a non-uniform duty cycle is used for the clock: the clock should have a duty cycle of 1/8, that is the clock is high 1/8 of the time and low 7/8 of the time.

To fulfil these requirements, a clock generator circuit based on a clock recovery circuit and a counter will be used. The input to the clock recovery circuit will be a differential sine wave at a common mode offset voltage in a similar way as the analog input signal. The counter will then be used to decimate the input clock into the non-uniform duty cycle that are required by the converter. The advantage with having this additional circuit is that the risk of affecting the analog input is reduce at the same time as the jitter performance is improved.

5.7.2 Output buffer

Different methods for taking care of the output has been analysed. This has included both direct output and storing of data on chip. The method of storing the samples on the chip and then read them off afterwards are a popular method used when testing high

speed converters [37, 38, 45]. However this method comes with increased on-chip complexity in combination with the fact that the size of the memory will occupy a significant chip area that is many times larger than the expected area required for the converter implementation. This method is mostly used when the sample rate reaches several GS/s.

At speeds around 1 GS/s and below it is commonly so that the data is directly fed out of the chip [1, 22, 46]. This is since it simplifies the design; at the same time, it is possible to directly capture the data at these speeds. It however comes with the problem that switching outputs at full voltage swing may affect the performance of the chip due to the amount of capacitance that is connected to the output. This is also the principle that will be used to output the data here.

To make sure that the data is kept stable for long enough to be captured, a buffer will be used. This buffer also reduces the load on the memory by taking care of driving the outputs. The buffer will be based on two *layers*, where each layer consists of a set of memories. The first one will be responsible for capturing the data from the memories during the conversion cycle. The second layer is responsible for keeping this value throughout the whole next cycle so that it can be captured off-chip.

5.7.3 Proposed tests

An ADC can be tested in several different ways depending on what results that are of interest. Some different test methods that could be used for testing the chip will here be presented. All tests are based on having a system connected to the converter that can store the results for post-processing which is required due to the redundancy used. The data acquisition system can either consist of a logic analyser or a Field-Programmable Gate Array (FPGA).

Firstly, a test method for characterising the static converter properties, which mainly consists of charactering the linearity. There are two methods to perform the measurements used to calculate the DNL and INL. These are code edge detect and the histogram technique [11]. The code edge technique builds on tuning sampling in the edge voltage and multiple times to check that half of the samples falls into each side of the edge. This is however a time-consuming process due to the number of samples required at each level in order to get sufficiently high confidence. The histogram technique builds on inputting a ramp signal into the converter and looking at the number of samples that fall into each level. This method requires only a small number of samples that would be required for the code edge method without affecting the resulting INL calculations.

Secondly, a test method for the dynamic converter properties. This part may include several different measures such as SNDR and SFDR and can be performed in several different ways depending on what results that are desired. The test principle itself does however not change; rather it is the selection of the inputs that is altered. There are essentially two main types of results that are of interest: how the converter performs at a specific sampling frequency with different input frequencies, and the performance of the converter for various sample rates with an input signal of half the sampling rate. The test involves providing a sine wave input signal to the converter at a desired frequency and storing the output data. A Fast Fourier transform (FFT) is then used to provide the signal spectrum. This output can then be used to determine the SNDR or SFDR. The FFT output can also be used to evaluate the design by analysing which effects that limits the performance.

Thirdly, a test method to determine the converter power consumption. This testing can be performed during both the above-mentioned tests. However, the power consumption may be affected by the input signal and therefore it may be so that the power consumption

for sine wave is more reasonable. It is important to make sure that only the design and not any other circuitry affects the measured power consumption.

Finally, additional tests for which would be interesting for characterising the converter. One such test is to analyse the performance with and without the offset voltage calibration. This evaluation is only possible if the calibration can be turned of externally. This evaluation will essentially be an additional part to the dynamic evaluation where the same tests are performed with and without the calibration. Another test would be to analyse the converter performance at different supply voltages and study how the performance and power consumption are affected by the supply voltage changes.

6

Implementation

All components used in the implementation of the converter are based on circuits designed at the transistor level. The reason for implementing the circuit on transistor level instead of using standard logic cells at the places where it would be suitable (such as in the state machine) is that it would put limitations on the freedom at layout. It also gives more freedom in sizing the different stages as well as optimizing the logic, thereby minimizing the transistor count as well as the delay.

This chapter starts with presenting the implementations of the different components used in the converter. Then, the implementation of the supporting circuitry for test will follow.

6.1 Comparator

The comparator has been implemented based on the strong-ARM topology; the circuit implementation can be seen in figure 6.1. This figure shows the comparator together with both the offset calibration transistors and the kickback noise reduction transistors. The sizes of the transistors used for the comparator implementation are given in table 6.1.

The comparator implementation is based on a trade-off among speed, mismatch, and kickback noise even though calibration and reduction techniques are used for both the mismatch effects and kickback noise.

The comparator can be divided into four sections with different functionalities: input, latch, reset, and output. For each of these sections, there are different factors which affect how the transistor size should be chosen. Starting with the output buffer, the size of the last stage depend on the total load connected. The output buffer also helps balance the load on the differential pair by having known loads with minimum parasitics connected to the latch. The last stage in this output buffer has been sized to approximate a fan-out of four. The same ratio has also been chosen between the first and last stage of the buffer to improve speed. This however results in having a transistor size at the first stage which is not of the minimal size that can be used, thereby resulting in a larger capacitance added to the latch reducing its speed. This capacitive cost was however considered reasonable since a buffer stage with long delay would be hard to compensate for.

In the latch section, the transistor sizes depend both on the output load and on the intrinsic load in the latch itself. The latch essentially consists of two inverters and the initial sizing was based on the principle of having PMOS transistors twice as large the as NMOS transistors. However since the latch is stacked on top of other transistors, the size of the NMOS transistors had to be increased to compensate for the slightly limited driving capability below. The transistors in the latch have been sized in a way that they have significantly larger size than for a fan-out of four, to have as little effect from the output on the latch as possible. However this is a trade-off where the speed improvement

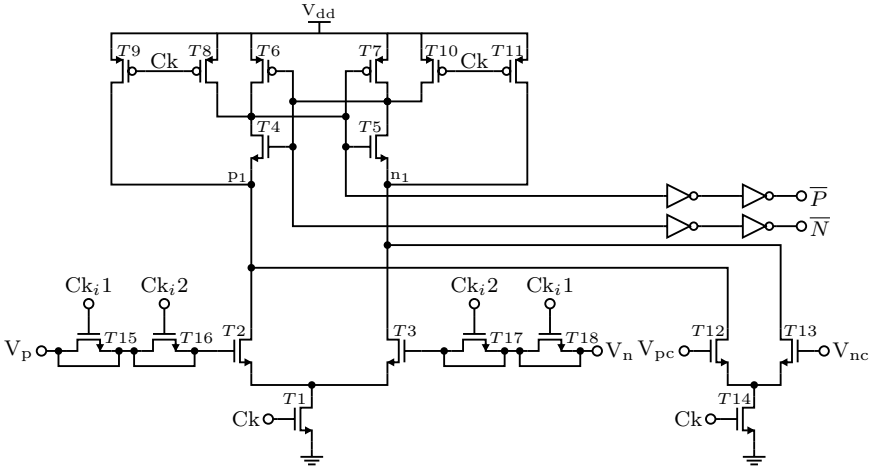


Figure 6.1: The comparator circuit implementation together with offset calibration and kickback reduction. The sizes used for the different transistors are given in table 6.1.

Table 6.1: Transistor sizes used in the comparator.

Transistor	Width	Length	Comment
$T1$	6 µm	30 nm	Decision activation
$T2$, $T3$	4 µm	30 nm	Input transistors
$T4$, $T5$	500 nm	30 nm	Latch transistors, NMOS
$T6$, $T7$	800 nm	30 nm	Latch transistors, PMOS
$T8$, $T10$	400 nm	30 nm	Latch reset transistors
$T9$, $T11$	300 nm	30 nm	Reset transistors, nodes p_1 and n_1
$T12$, $T13$	1 µm	30 nm	Offset calibration transistors
$T14$	1.5 µm	30 nm	Path activation for offset calibration
$T15$, $T18$	3.4 µm	30 nm	Main kickback reduction
$T16$, $T17$	300 nm	30 nm	Initial transient kickback reduction
PMOS	400 nm	30 nm	Buffer stage 1
NMOS	200 nm	30 nm	Buffer stage 1
PMOS	800 nm	30 nm	Buffer stage 2
NMOS	400 nm	30 nm	Buffer stage 2

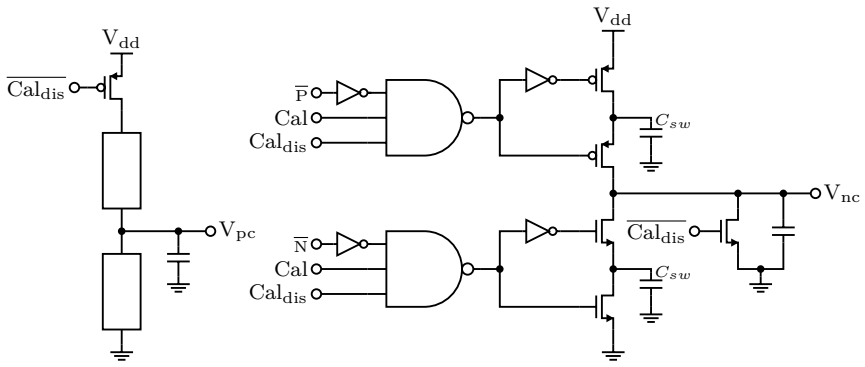


Figure 6.2: The circuit implementation of the calibration circuitry which provides the control signals for the calibration transistors in the comparator. The circuit is inspired by the circuits used is [1, 5].

gained by increasing the transistor size reduces as the size increases.

The input section of the comparator contains the transistors which have the largest size. The reason for this is that the input voltage is a signal operating around an offset voltage rather than a full swing digital signal and this has effects on the transistor transconductance. While large transistors improves the matching (thereby decreasing the offset voltage), their large size will also increase the kickback noise. Large input transistors will also cause an increase in the parasitic capacitance in the CDAC, thereby making it require a larger reference voltage. Just as for the latch stage, the benefit of increasing the transistor size decreases when the size increases.

The sizing of the reset section is essential, since the parasitic capacitance of these transistors degrades the speed of the comparator. The size also determines the speed at which the comparator can be reset. A complete reset is essential for the correctness in the comparator operation and therefore it is important to analyse the comparator reset so that a complete reset is guaranteed while not introducing unnecessary large capacitance into the nodes for which reset switches are added.

6.1.1 Offset calibration

When deciding the size of the transistors used for the offset calibration, it is important that the effect from these transistors is small enough so that they would not affect the operation of the comparator by drawing too much current reducing the effect of the main input transistor pair. At the same time, the effect caused by these transistors should be large enough to be able to handle offset variations without reaching the limits of what the calibration circuit which provides the input voltages can provide.

The circuit used to provide the control signals for the offset calibrating transistors is shown in figure 6.2. This circuit is based on the principle presented in [5] and the circuit used in [1].

The implementation consists of two parts: a resistive divider with capacitive buffer that provides the voltage to the positive leg of the calibration. The other part of the implementation is the part which changes to compensate for the offset voltage.

The voltage used in the fixed half of the calibration circuit affects the ability to correct offset voltage. This since the calibration circuit produces a voltage while the calibration itself is conducted by a current which is related to the voltage in square. This means that when the adjustable calibration voltage is larger than the fixed one the calibration precision will be limited by the minimum change in the calibration voltage. The fixed

voltage must therefore be balanced between the minimum step that can be taken and the fact that it needs to be larger than the threshold voltage of the transistor to which it is connected. In this case, a fixed voltage of 0.6 V has been chosen since it fulfils this balance.

The part which controls the changeable voltage builds on two stacked inverters, each with a small capacitance C_{sw} connected to its output. This capacitance is used to charge or discharge the larger capacitance placed between the two stacked inverters. The problem with using actual inverters stacked of top of each other is that it will limit the range in which the voltage can vary. This is since the source voltage on the NMOS transistor in the top inverter as well as for the PMOS transistor in the bottom inverter will cause the transistor to be turned off since the gate-source voltage will be less than the threshold voltage.

To overcome this problem, only PMOS transistors are used in the top inverter while only NMOS transistors are used in the bottom inverter. Inverters are used to achieve the same functionality, however, this brings the problem of leakage current due to both transistor being open at the same time. This current limits the calibration step size. The contribution from the leakage current is approximately equal to the contribution from C_{sw} in terms of added or removed charge.

It may seem like the capacitance added after the restive network is redundant since no charge are drawn from these nodes. It is however very important that this capacitance is there and that it has the same size as the one used in the switched half of the circuit since it affects how the circuit is affected by kickback noise.

The NAND-gate is used to control when calibration should be performed and it is essentially operating as a switch. The possibility to turn off the calibration has also been added to improve testability, as argued in section 5.7. This has been done by adding a control signal to the NAND-gate controlling when calibrations are performed. An NMOS transistor has also been added here to discharge the capacitor and make sure that no charge builds up due to leakage making the calibration active. For the fixed side it is sufficient to only add one transistor that turns off the supply which is done by the PMOS transistor added. The reason that no transistor is required for making sure that the capacitor is discharged is that it is connected to ground through the resistive divider and therefore there is no need for having additional connections to ground.

6.1.2 Kickback noise reduction

To reduce the effect of kickback noise going back into the sampling capacitance, dummy transistors are added to the two input nodes, as can be seen in figure 6.1. Two transistors have been added in each of the input nodes and this is due to the complexity in how the leakage is formed. The implementation of the kickback noise reduction is done for minimizing the kickback noise at a common mode voltage $V_{com} = 0.55\,\mathrm{V}$.

The kickback noise emerging from the comparator has two sources: one is the clock itself while the other is caused by pulling down nodes p_1 and n_1. The kickback from these two sources also has different polarity where the effect caused directly by the clock pulls the voltage up while the other pulls the voltage down somewhat later. The part originating directly from the clock is significantly smaller and thereby the transistors used to cancel this effect also becomes smaller which can be seen in table 6.1.

As can be seen in figure 6.1, no kickback reduction has been added to the inputs of the offset calibration transistors. There are two reasons for this. First, this input is not shared among the two comparators which means that the kickback effects caused by one of the comparators will not affect the other one. Second, the effect of kickback into the calibration voltage nodes are not as severe as for kickback into the sampling capacitance.

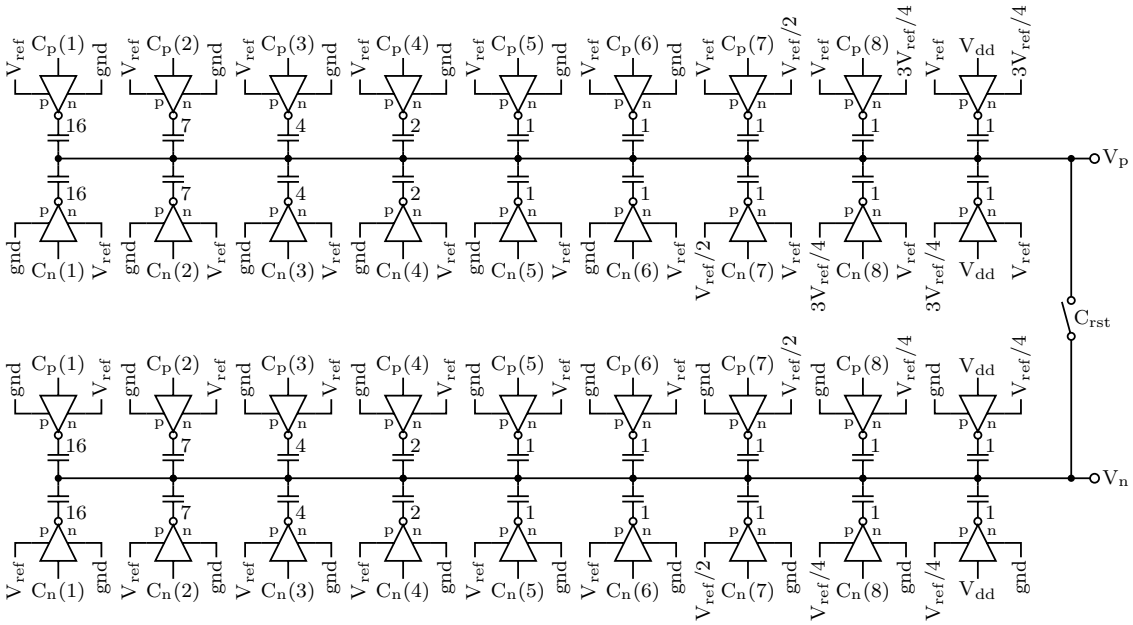


Figure 6.3: A schematic showing the implementation of the CDAC for the scaling *Generic 1*. The schematic also shows how the different reference voltages are connected to the different cells.

This is since the size of the capacitors used in the charge switching calibration control are much larger than the size of the sampling capacitance together with that the size of the calibration transistors are smaller than the input transistor pair which also contributes to the minimized kickback noise in the calibration circuit.

6.2 CDAC

The CDAC has been implemented by adding unit cells to form the required size of each of the stages that makes up the CDAC. This implementation has been performed for both the scalings *Generic 1* and *Generic 2* in order to be able to further evaluate these two scalings in terms of how mismatch affects the linearity.

A schematic of the implementation of the scaling *Generic 1* is shown in figure 6.3. The implementation of the scaling *Generic 2* is similar to the one shown in the figure. The difference is that $V_{ref}/2$ is used in stage 6 as well as for the fractional part in stage 2.

The reset switch used to reset the CDAC before the calibration of the comparators has been implemented by a NMOS and a PMOS placed in parallel. This combination of NMOS and PMOS has been used to provide a range where the conductivity is kept high, thereby shorten the reset time.

6.2.1 Unit cell

The unit cell topology in essentially based on an inverter, however a traditional inverter based on a NMOS and a PMOS was not suitable due to the low reference voltages used. Instead the unit cell had to be implemented based on two NMOS transistors which can be seen in figure 6.4. This however poses a problem since the functionality of a PMOS must be mimiced by the combination of a NMOS and an inverter which causes additional delay. The effect of this additional delay is that during a short amount of time both transistor

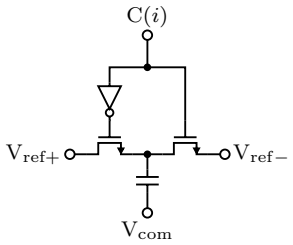


Figure 6.4: A schematic showing the implementation of the unit cell.

will be active, thereby providing a direct path between the reference buffer and ground, causing significant additional discharge of the reference buffer and lowering its voltage.

Several alternatives were analysed in order to reduce the effects on the voltage variations caused by this leakage current. First, the size of the reference buffer was increased, but this would cause problems in terms of the size required for the buffer capacitance, so it was decided to only use this method for small final compensations. The second principle that was analysed was to put circuitry between the memory and unit cell to delay the signals so that they would arrive at the same time. The problem with adding this delay is that the settling of the CDAC is included in the operations which are critical and additional delays result in the requirement of delaying the generation of the next clock. This option was therefore scrapped.

The only remaining is to look at reducing the sizes of the transistors. This will however affect the settling time and thereby also result in a delayed clock. The topology of the CDAC however partly solves this problem in the way that the reference voltages are connected to the unit cell. Changing the way reference voltages are connected to the unit cell instead of inverting the control signal makes it possible to only keep one side of the unit cell fast while the other one can be slowed down a bit to reduce leakage. This slowdown can however not be too large since this would affect the CDAC reset time.

The sizing of the two main transistors resulted in a trade-off between speed and leakage current. The charge consumed by the leakage current was chosen to be equal to the charge used to switch the unit capacitance. With this leakage current, a sufficiently short reset time was still provided.

The capacitor used in the unit cell is a Metal Oxide Metal (MOM) capacitor. The MOM capacitor is based on interleaved metal fingers. The MOM capacitor has the advantage that its capacitance is constant over the voltage range while the transistor gate capacitance varies with the voltage.

A switchable capacitance of 1.8 fF has been chosen, which is the smallest capacitance that can be implemented with this type of capacitor in this process. The non-switchable parasitic capacitances connected to the reset switch, to the input capacitance on the comparator, and to other sources will contribute with around 1.1 fF per unit cell, giving a total capacitance per cell of 2.9 fF.

The settling time for the unit cell was also an important design parameter which affects the transistor sizes. Due to the importance of meeting the settling time, additional margins have been added. A minimum settling time of 3τ has been used to guarantee that the longest settling time in both the scalings, *Generic 1* and *Generic 2*, is fulfilled. A total maximum settling time of 30 ps, for the TT corner, gives $\tau = 10$ ps. As an additional margin this maximum τ should be fulfilled also in the SS process corner.

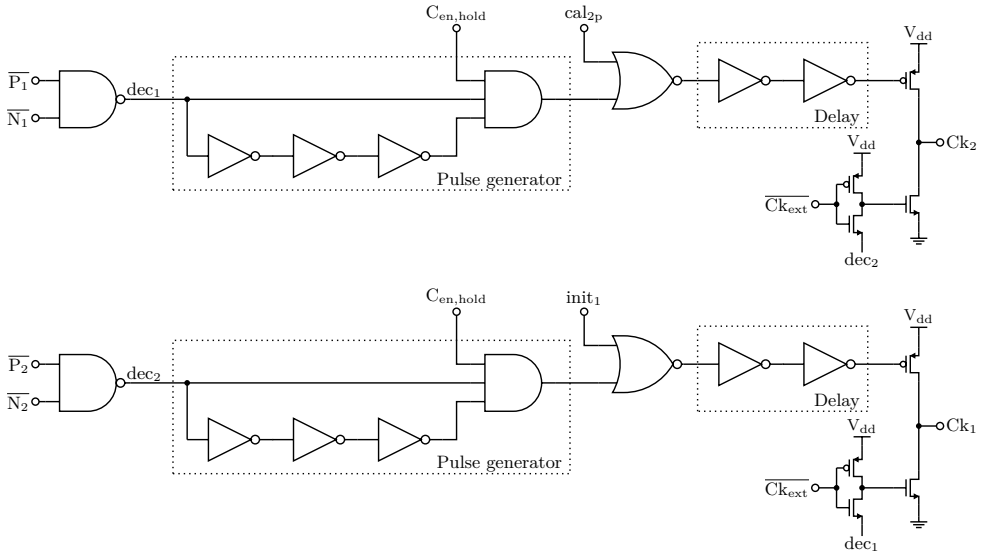


Figure 6.5: Schematic of the clock generation circuit based on logic gates.

6.3 SAR logic

The SAR logic has been implemented in three blocks and these will now be presented.

6.3.1 Clock generation

The clock generation has been implemented based on two paths where each path controls one clock. Each of these paths contains five stages: decision detection, pulse generation, clock initiation, delay, and output. These stages can be seen together with the circuit based on logic gates in figure 6.5. These logic gates has in the implementation been replaced by transistor based circuits where logic logic optimization has been used to reduce the number of circuits in the critical path.

The decision detection generates the signal which stops the currently active clock while starting the other one keeping the alternating clocks running. This stage is then followed by a pulse generating stage. This stage is required to guarantee that the clock can be turned off when the next decision arrives and not get stuck with both output transistors open. The $C_{en,hold}$ signal is used to enable and disable the clock path. This signal is used to stop the alternating clocks after all stages have completed. The initiation stage is used to initiate the clock at the beginning of each cycle as well as re-start the clock for calibration of the comparators. The reason behind one of the initiation signal being named $init_1$ while the other one is named cal_{2p} in figure 6.5 is that $init_1$ initiates both the clock in the beginning of each cycle as well as for comparator calibration. The control signal cal_{2p} on the other hand only initiates the comparator calibration. The following delay part is used to make sure that sufficient settling of the CDAC has been performed. The different delay blocks are used to delay the clock different amounts for the clocks used for the kickback noise reduction. Finally comes the output stage which is the stage that directly drives the clock signals. The switch controlled by the sampling clock is there to guarantee that the clocks are turned off before the next cycle starts thereby making sure that the comparators are reset.

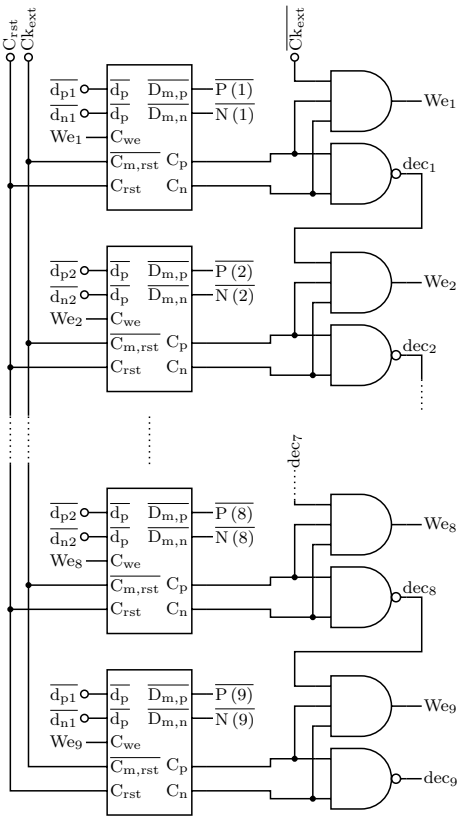


Figure 6.6: Schematic showing the implementation of the state machine using logic gates.

6.3.2 State machine

The state machine has been implemented around the memory cells storing the results from the previous comparisons as can be seen in figure 6.6. This way of implementing the state machine makes it possible to utilize only simple combinatorial logic to generate the next state signal which is the write-enable signal to the memory.

The logic has been implemented so that the delay from detecting stored result to turning off the current write enable and activating the next one is minimized. This has been done by using a parallel structure where the write enable is controlled by the previous decision detect signal dec_{i-1} together with the current stored result $\overline{P_i}$ and $\overline{N_i}$, given the current state i. This method however increases the load on the memory cells. The alternative would be to use a serialized method based on the decision detect from last stage dec_{i-1} as well as from the current stage dec_i.

The state machine is initiated by the sampling clock in a similar fashion as the decision signal from the previous state. To guarantee that the state machine always starts at the first stage even if the previous cycle is not complete, the reset signal is based on the clock.

The state machine is also responsible for the generation of the $C_{en,hold}$ signal as well as the different reset signals. $C_{en,hold}$ is activated by the clock and deactivated when the results at the eighth stage have been stored in order for the clock paths to be disabled before the decision at the ninth stage arrives. The reason for having several reset signals is that different parts of the system requires different types of reset. The reset of the CDAC must be a pulse since the calibration will follow. The reset of the CDAC control signals will however need to be active until the sampling clock rises to guarantee that the decisions stored in the memory does not leak back and sets the CDAC ones more.

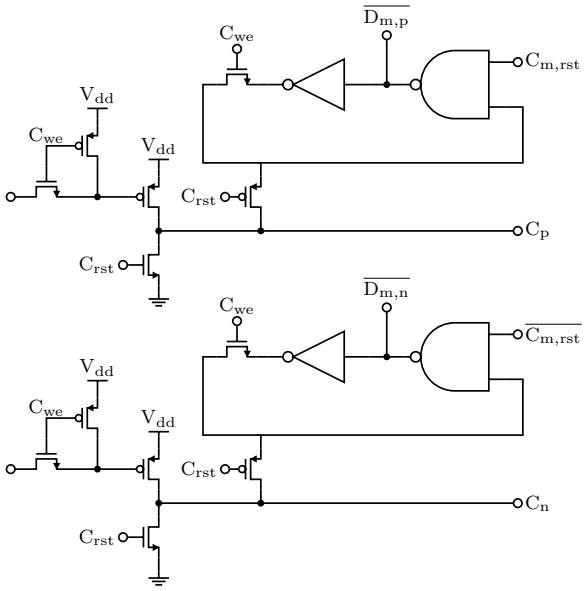


Figure 6.7: A schematic of the implementation of the memory cell.

6.3.2.1 Memory

The memory has been optimized to reduce the delay from decision delivered from the comparator to the outputs of the memory. A schematic of the memory is shown in figure 6.7. The two different outputs provide the possibility of resetting the control signals to the CDAC without resetting the output.

The output from the comparator is always one except for when a decision is delivered. This means that the default state of the memory should be one in order to be able to detect when a decision has been taken on the output. Since only low inputs propagate, a reset of the comparator would not affect the value stored in the memory in case of the write enable still being active.

The memory cell is implemented in such a way that the CDAC control signals are reset separately from the memory loop. Under normal operation will the control signals be reset first in order to reset the CDAC and the memory loop later on to make it possible to capture the stored data. However if the cycle is not completed and a reset of the CDAC has not been performed, the reset must be provided by the memory loop so that the CDAC is reset at the start of the next cycle.

Since the output controlling the switching in the CDAC is essentially separated from the comparator by a PMOS transistor, it becomes important to properly size this transistor. This is since too large a transistor will significantly load the comparator output while too small a transistor would increase the settling time of the CDAC.

6.3.3 Calibration

The calibration logic is the most diverse part of the SAR logic. In addition to providing the calibration signals, it also generates the initiation signal for the clock generator.

The clock initiation is based on a pulse generator similar to the one used in the clock generator block followed by an OR-gate which is used to combine the pulse with the calibrations initiation pulse for calibration of comparator one.

The calibration logic generation is based on two stages: a pulse generator which generates both a pulsed version and a stable version of the calibration enable signal, and the

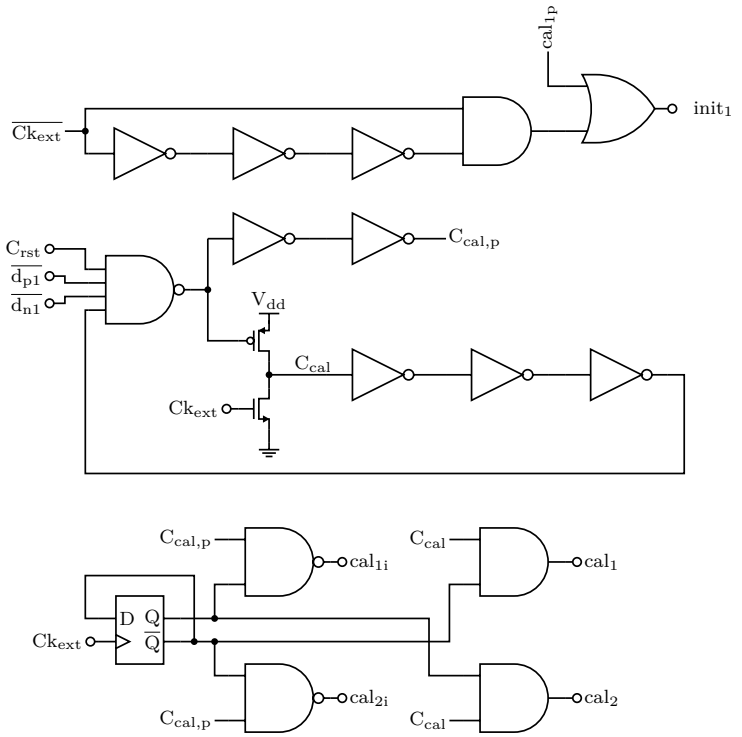


Figure 6.8: Schematic of the logic generating the calibration enable signal used to activate one of the calibration signals.

memory that keeps track of which comparator that should be calibrated. The reason that both a pulsed and a static version of the signal is required is that the generation of the comparator clocks require a pulse while the calibration circuit requires a static signal to activate the calibration so that the result can be used to change the calibration voltage. These signals are than delayed before being ANDed to the output of the flip-flop keeping track of which comparator that should be calibrated. The calibration signal for one of the comparators is connected to the inverted output of the flip-flop while the other is connected to the non-inverted one. The state in the memory is then updated by feeding the inverted output back and storing it on the rising edge of the system clock.

6.4 Track and Hold

The T&H is based on a complementary switch in each of the two input legs. Then dummy transistors have been added on each side of the transistors to reduce the clock leakage. The reason for having dummy transistors also before the switches is that the converter is designed as a unit cell which can be used to achieve even higher sampling rates by placing several devices in parallel. This means that all the inputs of the different converters are connected together and leakage from one switch may affect the voltage sampled at another switch. Figure 6.9 shows the implementation of the sampling switch together with the dummy transistors.

The placement of the dummy transistor was analysed in terms of trying to reduce the total switched capacitance. This could in theory be done by adding an NMOS dummy transistor of the same size as the NMOS transistor used and switch them in the same direction, thereby compensating for the leakage in the PMOS transistor which is twice as

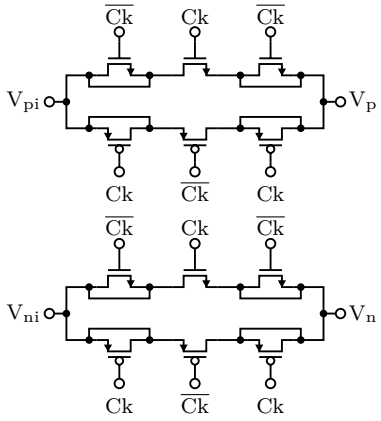


Figure 6.9: An illustration of the T&H circuit used.

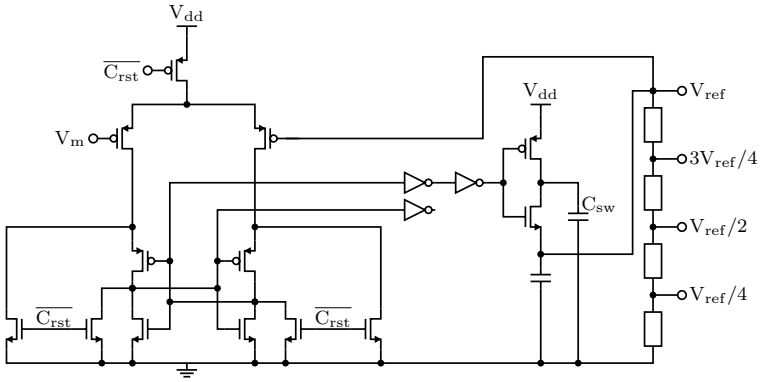


Figure 6.10: Schematic of the implementation of the reference voltage buffer.

large as the NMOS. This would however have limited effects since the capacitance of a Metal-Oxide Semiconductor Field-Effect Transistor (MOSFET) is voltage dependent [28]. The behaviour is also complicated by the leakage dependency of the rise and fall times of the control signals [43]. The combination of all these problems limits the effect of using only one dummy transistor to reduce the clock leakage.

Instead two dummy transistors has been used, one for the NMOS and one for the PMOS, both being switched in the opposite phase to the switch they are compensating for. Their sizes have been optimized for minimizing the clock leakage at a common mode voltage of $V_{com} = 0.55\,\mathrm{V}$. This is since this voltage is in the middle of the intended range for the input common mode voltage.

To guarantee that sufficient settling occurs for all input voltages, the transistors have been sized in a way such that only one of them, when operating in saturation, will be sufficient to completely settle the sampling capacitance within the sampling time. This means that in most cases the settling time will be significantly shorter.

6.5 Reference voltage

The reference voltage buffer implementation is based on a large capacitor acting as a charge buffer in combination with a comparator and circuitry to add charge to the buffer. The implementation of the buffer can be seen in figure 6.10.

The comparator used to compare the master reference voltage V_m with the locally stored reference voltage at reset is based on the same topology as the main comparator.

There is however a large difference between them: for the comparator used in the reference buffer, all NMOS transistors have been replaced by PMOS transistors and vice versa, effectively flipping the whole circuit upside down. This needs to be done in order to get a minimum input common mode voltage of 0, which is required when comparing the master voltage to the local voltage since they will be refereed to 0 and in most cases be smaller than the threshold voltage for the NMOS transistor.

The comparator then controls a similar circuit as the one used for the offset calibration. The difference is however that there is no need for the bottom half, since the circuit will always be discharged through the resistive divider used to provide the fractional reference voltages. Another difference is that the charging circuit is in this case implemented with a inverter instead of only PMOS transistors. This is since the range in which the buffer is designed to operate is well below the level at which the NMOS transistor turns off due to a too small gate-source voltage.

The size of the buffer capacitor decides how large the voltage ripple will be due to the charge removed when switching the CDAC. This capacitance is a large contributor to the footprint, making the size a trade-off between voltage ripple and size. In this case a capacitance of 5.56 pF has been used. The buffer is based on a capacitor type which makes it possible to place it beneath the capacitors used to implement the CDAC, which reduces the total size of the implementation.

The capacitance used to switch in charge C_{sw} should be sized in such a way that it charges the buffer capacitor during most of the cycles. This sizing is however only valid for the highest voltage which the reference buffer supports, since this voltage decides the amount of charge which can be added by the C_{sw} capacitor.

The size of the resistors used in the resistive divider affects both the settling time of the CDAC and the leakage from the buffer. A resistor size of 8 kΩ has been chosen. This results in a charge leakage from the buffer corresponding to approximately 1/4 of the charge used while switching the CDAC. This resistor size will also be in the same order of magnitude as the transistor on-resistance in the CDAC switches. This, in combination with the shorter required settling time at the stages where the fractional reference voltage is used, minimizes the effect of the larger series resistance.

6.6 Supporting circuitry for test

Several principles have been used in the implementation to be able to monitor the operation of the chip. The internally-generated clock signal will be fed out from the chip, not only to indicate when data are ready on the output, but also to show that the most essential part for controlling the system works. The local reference voltage buffered on the chip is also fed out. This is since this voltage is essential for charge redistribution.

It is also possible to turn off the comparator offset calibration. Calibration is not essential for the operation; rather, it is there to improve the performance. Making it possible to turn off the calibration also has the benefit that it becomes possible to evaluate the effectiveness of the calibration circuit.

The redundant behaviour of the circuit may cause problems by masking severe errors. Not converting the redundant output into binary representation on the chip also makes it possible to track the behaviour of the circuit by looking at each comparator decision. Conversion from the redundant representation to binary was excluded in the delimitations, but in that case for completely different reasons. Another benefit is that it becomes possible to track for incomplete conversion cycles in the data and thereby connect it to the resulting performance.

6.6.1 System clock generation

The system clock generator is based on two stages: a clock reconstruction circuit and a division circuit. The clock reconstruction circuit is used to reconstruct a single ended digital clock from a differential sine wave signal with non-zero offset. The division circuit is then used to generate the system clock with its duty cycle of 1/8.

The clock generation circuit is based on the circuit presented in [47] and is shown in figure 6.11. This architecture was chosen since it provides a low jitter at the same time as it has a well documented process of implementing it [48]. This was useful since the clock generator was not the core part of the design, but rather an important part required for testability.

The architecture builds on a differential amplifier with a current mirror placed around it to convert from a differential into a single ended signal. A buffer is used at the output to further decrease the rise and fall times while also minimizing the load on the output.

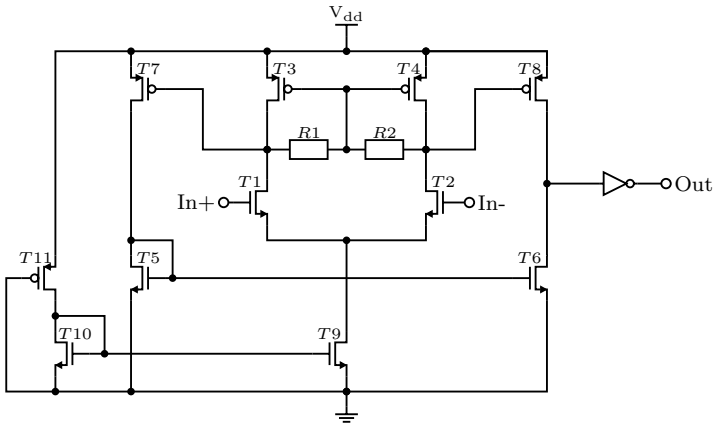
The divisor circuit is used to generate the 1/8 duty cycle clock. This circuit is based on a three-bit counter and delivers the system block by ANDing the state of all flip-flops. Following this counter is a buffer that increases the drive strength of the signal in order for it to be able to drive the converter clocks. The compact circuit is essential for making the clock generation work since the counter has to operate at eight times the speed of the converter.

6.6.2 Output buffer

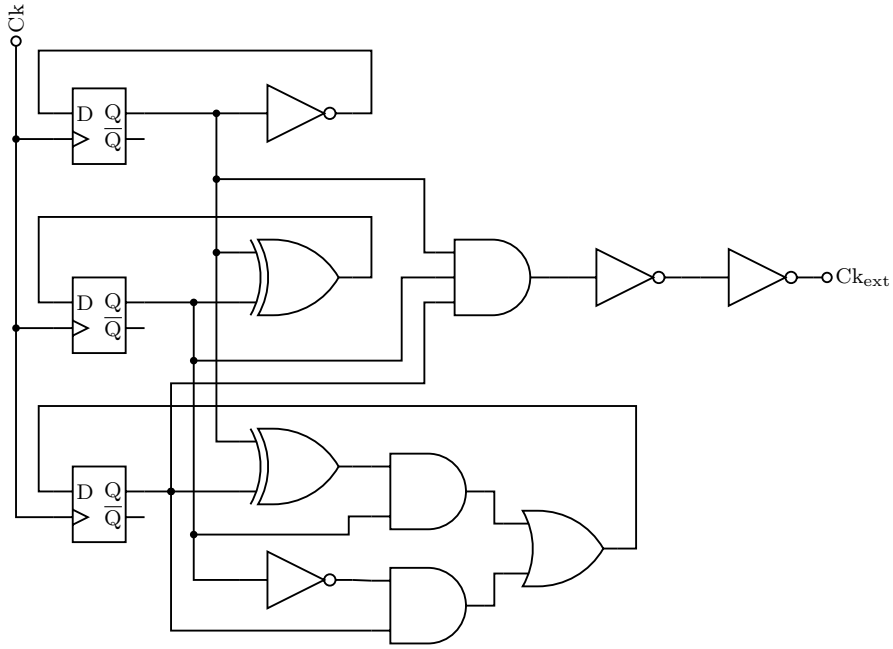
The output buffer has been implemented using one layer of latches followed by a layer of registers. The first layer is used to copy the data from in the memories directly after the data has been stored. The write-enable for this first layer of latches is implemented in a similar way as in the state machine which can be seen in figure 6.12. The reason for using this rippling write enable is that even if a cycle would not be finished, all the decisions that have already been stored will also have been copied to the output buffer. There is just one case where not all data will be stored correctly: if the last decision comes just before the clock edge so that it has time to settle in the memory but not in the output buffer. If this occurs, the converter would have been pushed to far up in speed, or there will be a decision that takes much longer than the standard decision. This first layer will be reset on a delayed version of the clock to make sure that the data has been stored in the second layer.

The coping from the memories to the output buffer could also have been implemented based on a finished signal that indicates that all stages are performed. The problem with this method is that if not all comparisons are performed no data will be copied. This would also be troublesome from a testing perspective since it would not be possible to detect why a outputs are not delivered. By getting some outputs makes it possible to analyse why the design will not work.

The second layer builds on an array of registers that are all updated at the same time. This is to update the result on the outputs for all bits at the same time rather than having them updated as they come which would make it hard to know if a bit has been updated. The data in these registers will be updated by a slightly delayed clock. The reason for this is to make sure that there will be time for even late decisions to propagate through the first layer. The outputs are however updated on the output before the sampling clock edge falls which makes it possible to use this signal to indicate when a new result is feed out from the chip. Each output from the second layer registers is then fed through an inverter buffer to improve the drive strength of the signal.



(a) Clock reconstruction circuit



(b) Clock division circuit

Figure 6.11: Schematic implementation of the clock reconstruction circuit (a) and the division circuit used to generate the uneven duty cycle (b).

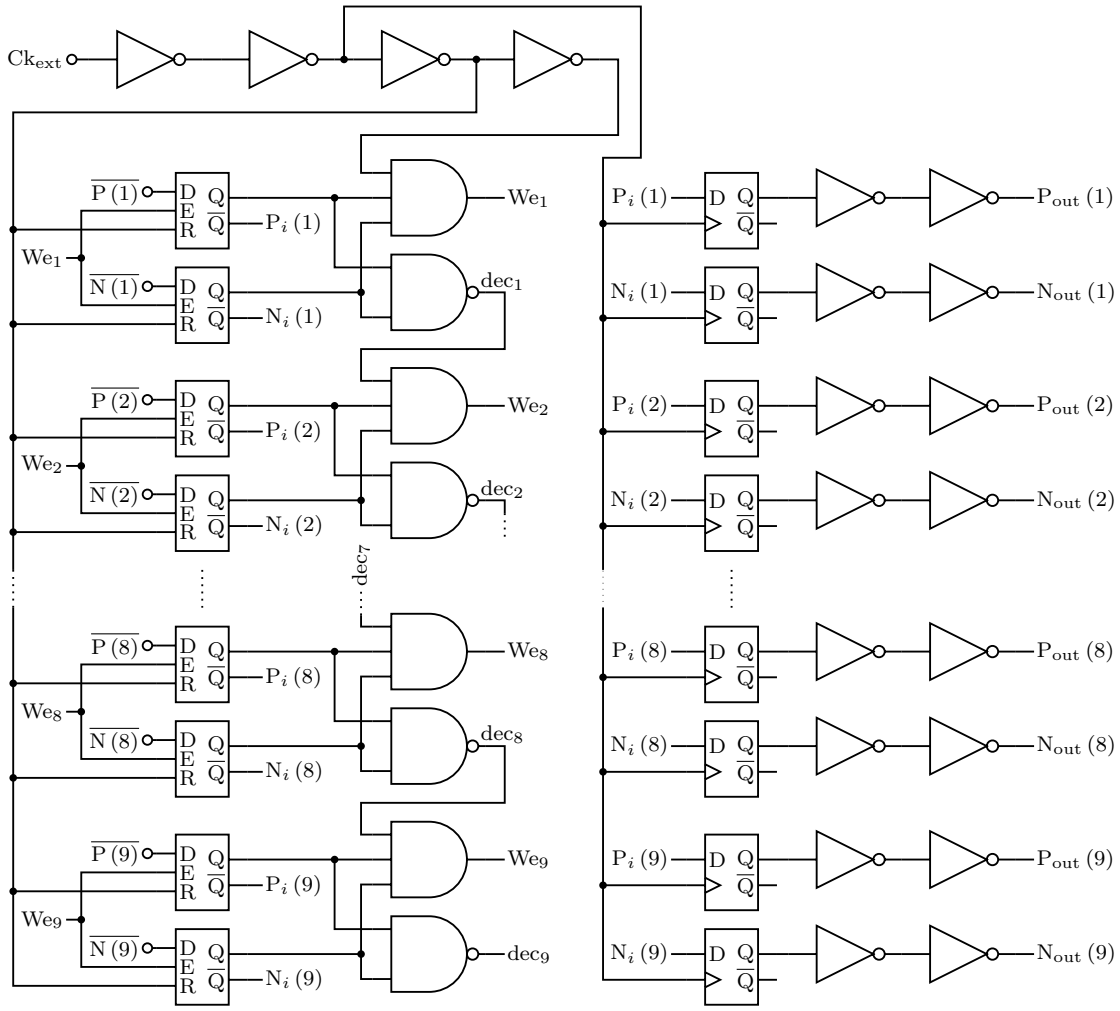


Figure 6.12: An illustration of the implementation of the output buffer with its two layers.

The signal which leaves the output buffer has been inverted compared to the signal which entered the buffer. This has been done in order to correct for the inverted behaviour of the output from the comparators that follows through the memories in the state machine.

7
Results

This chapter will present the results for from the evaluation of the implementation. The default assumption will be TT process corner and ambient temperature of 27 °C. In all cases where decision times are given, it will be assumed that a standard input voltage of 0.5 LSB $\approx$ 1 mV.

The results for the evaluation of the comparator, CDAC, and T&H will first be presented by themselves. Then the results for the evaluation of the whole design will follow.

7.1 Comparator

The evaluation of the comparator are performed based on three measures: the decision time, offset voltage, and kickback noise.

7.1.1 Decision speed

The decision speed of the comparator depends on the size of the differential input voltage. The decision speed is shown in figure 7.1 for various differential voltages in the range 100 mV to 1 pV. From this figure it can clearly be seen that the decision time is logarithmically dependent on the input voltage.

This makes it possible to calculate the expected decision time using equation (7.1) where $t_0 = 11.6\,\text{ps}$ and $\tau_{cmp} = 12.4\,\text{ps}$. This expression can them be used to estimate how much longer than the standard decision a small input voltage would require.

$$t_{dec} = t_0 + \tau_{cmp} \log\left(V_i\right) \tag{7.1}$$

The decision time dependency on the input differential voltage has been simulated with the comparator by itself using a clock with 10 ps rise and fall times and the inputs directly connected to voltage sources. The offset calibration inputs has also been connected to individual voltage sources which delivers the fixed common mode calibration voltage used. The comparator outputs are not connected to any load.

7.1.2 Mismatch

The comparator offset voltage has been evaluated with 250 MonteCarlo simulations where random mismatch was added to each of the components in the comparator as well as in the calibration circuit when this was used. The result is shown as histograms in figure 7.2. The y-axis shows the number of runs that falls into each bin.

Figure 7.2 (a) shows the results from the runs without calibration. These runs were performed as the reference showing the pure offset voltage of the comparator. The average offset voltage was in this case 700 µV and the standard deviation was 11.8 mV.

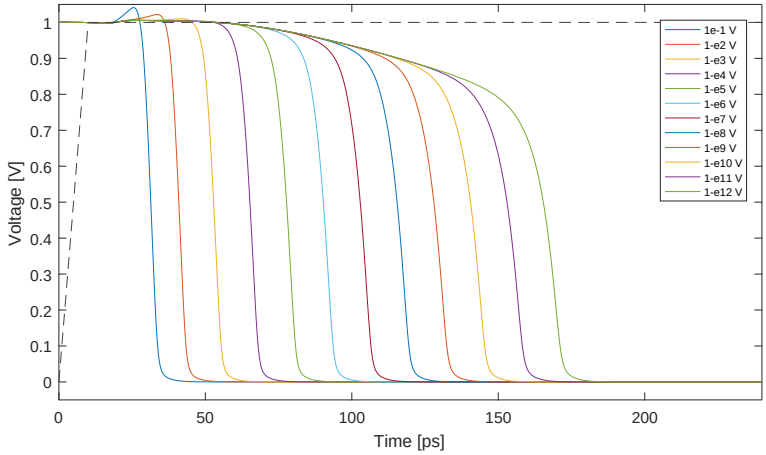


Figure 7.1: The comparator decision time for various differential input voltages in the range 100 mV to 1 pV.

Figure 7.2 (b) shows the results from the case with offset compensation used. The offset is significantly smaller than what it would be if no calibration were used. The average offset voltage is in this case $-20\,\mu V$ and the standard deviation was $217\,\mu V$, a decrease of more than 50 times compared to when no offset calibration where used.

The simulation set-up used to determine the offset voltage of the comparator is based on a sampling network similar to the actual implementation. The comparator clock is generated by the actual clock generator to make it possible to utilize the kickback reduction. When calibration is used, the comparator becomes calibrated during the initiation; then, calibration is turned off during the actual check of the offset voltage. The offset voltage is checked by sweeping the input until the comparison result changes from one to the other. When no calibration was used, these inputs were driven by voltage sources providing the fixed common mode voltage. The input voltage was changed in steps of $50\,\mu V$ for the case with calibration and $160\,\mu V$ for the case without calibration.

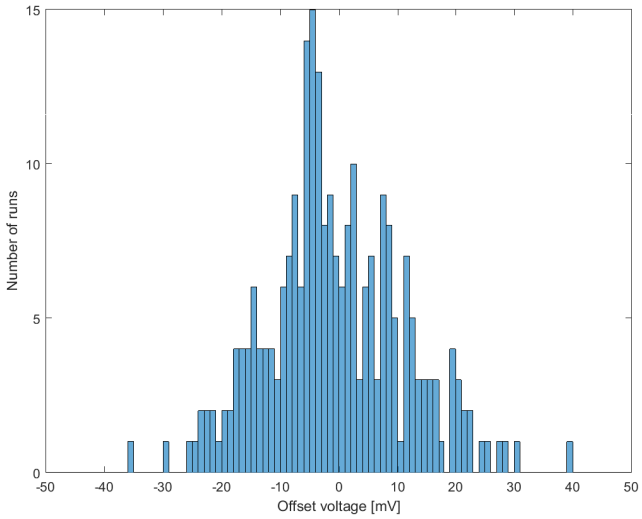
7.1.3 Kickback noise

The kickback noise has been evaluated by analysing it versus its dependency on the common mode input voltage. Choosing the kickback noise values is a bit tricky since there noise has two phases: one rapid transient one causing a large drop in voltage during a short amount of time, and the other phase consists of the effect which is visible during the whole time during which the comparator control clock is active. This effect shows up as a negative voltage shift. This second phase will also be used to calculate the kickback noise.

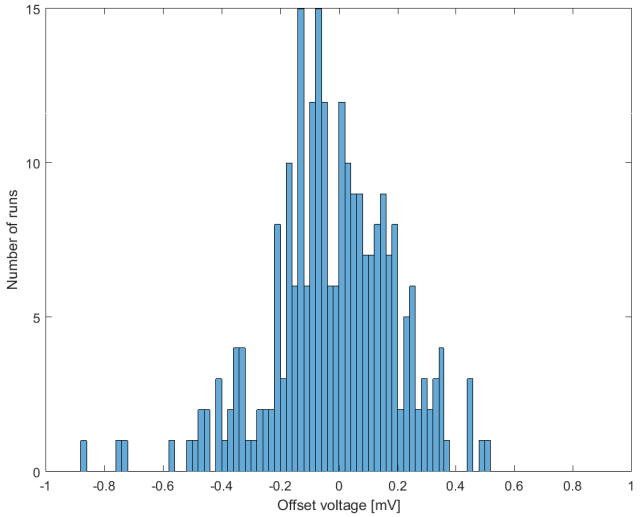
Figure 7.3 shows the kickback noise and decision time as a function of common mode voltage for both with and without kickback noise reduction.

As can be seen in the figure, the kickback noise has been reduced by at least a factor of 4 in the common mode voltage range $V_{com} = 0.5 - 0.6\,V$ for which the circuit has been designed. From the figure it can also be seen that the kickback noise is strictly increasing in the case where no compensation is used. This at the same time as the decision time is decreasing. The reason behind the rapid decrease and increase in kickback noise when reduction is used is that the reduction technique only compensates fully at a specific voltage; for all other voltages, the compensation will only be partial.

In figure 7.3 it can also be seen that the kickback noise slightly affects the decision

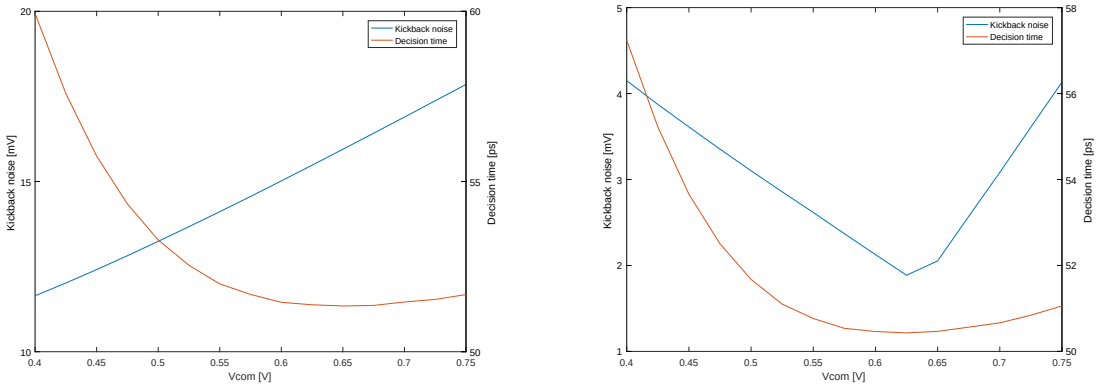


(a) Without calibration



(b) With calibration

Figure 7.2: Histogram showing the distribution of offset voltage without offset voltage calibration (a) and with offset calibration (b). The histograms are based on 250 MonteCarlo simulations and the y-axis shown the number of runs which falls into each of the bins.



(a) Without compensation

(b) With compensation

Figure 7.3: Plots of decision time and kickback noise as a function of common mode voltage for both the case without compensation (a) and with compensation (b). The differential voltage used is 0.5 LSB.

speed. This is a direct effect of the kickback noise since it produces a negative common mode voltage shift during the comparison, thereby reducing the effective common mode voltage seen by the comparator.

In figure 7.4 the kickback noise is shown for the case when alternating comparators are used. Here it can be seen that when no kickback noise reduction is used a large kickback occurs and the signal will not be fully restored before the other comparator starts its cycle. In the case with kickback noise reduction, the noise has much smaller amplitude.

The kickback noise has been analysed in a set-up similar to the actual use with the CDAC used as sampling capacitance to mimic the capacitance which the comparator will see in the actual converter. All the control signals to the CDAC were in this case grounded to ensure a constant voltage. The actual clock generator was then used to generate the clock signal for the comparator.

7.2 CDAC

When evaluating the CDAC, the linearity is the critical part. Here, the linearity will be presented by two measures, the DNL and the INL.

Starting with the DNL, figure 7.5 shows the DNL for both *Generic 1* and *Generic 2*. Due to the redundancy, a direct conversion from digital input code to output voltage was not possible to use for the DNL calculations. Instead the voltages needed to be combined in such a way that it would be possible to extract the DNL. This was performed by taking the difference between the voltage given by the current code and the closet smaller code which corresponded to the level below. Figure 7.6 shows how the actual step size has been calculated for the redundantly scaled CDAC, based on the scaling *Generic 1*. The input code corresponds to the CDAC control signals and are as can be seen not directly translatable into the output level. The arrows indicate the step size used to calculate the DNL.

The histogram in figure 7.7 shows the distribution of the maximum DNL. Here it can be noted that the distribution of the maximum DNL for *Generic 1* has a larger spread than the distribution for *Generic 2*. This can also be noted in the mean and standard deviation for the scalings. The mean maximum DNL for *Generic 1* is 0.2445 LSB and

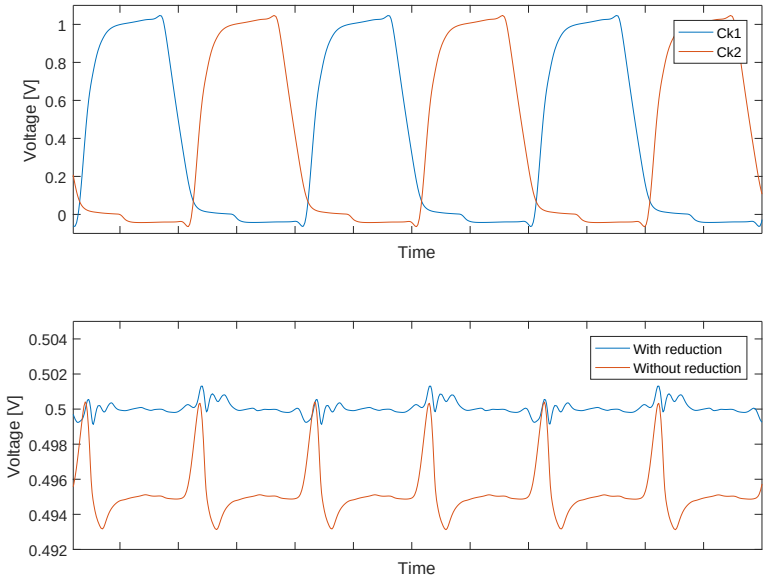
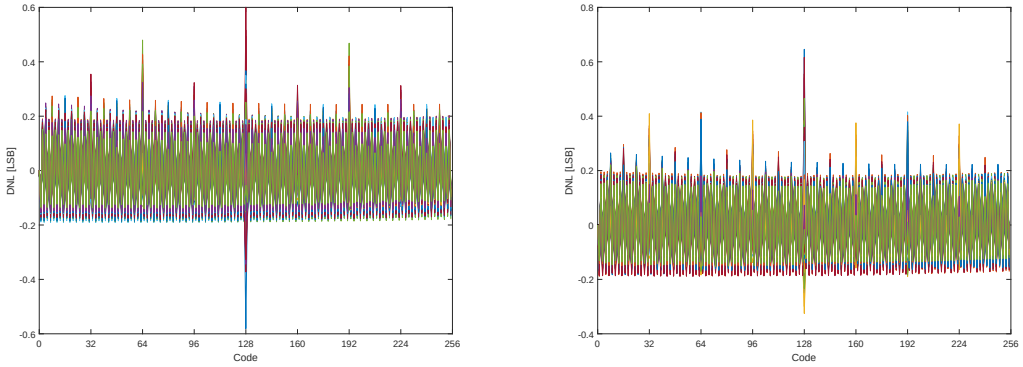


Figure 7.4: Kickback noise behaviour for alternating comparators. The top part of the plot shows the clock signals and the bottom part shows the kickback noise for both cases.



(a) *Generic 1* (b) *Generic 2*

Figure 7.5: DNL for the scalings *Generic 1* (a) and for *Generic 2* (b) for 250 MonteCarlo simulations. Due to the redundancy, the code is not translatable directly into a voltage.

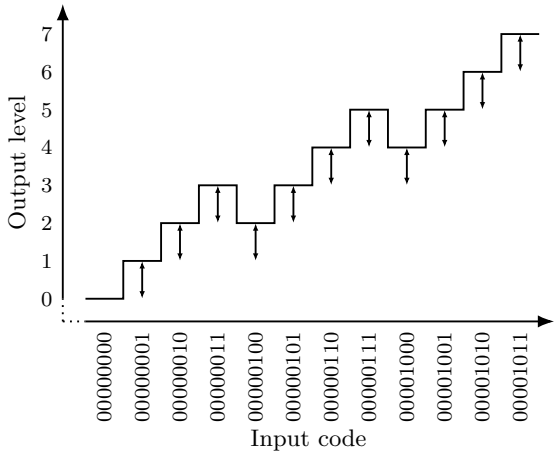
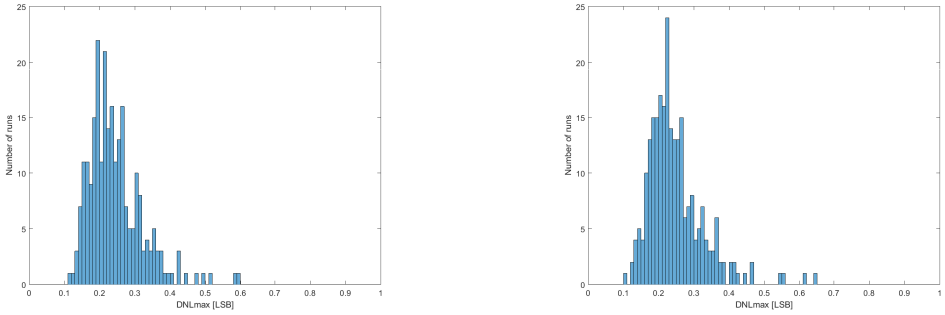


Figure 7.6: Illustration of how the DNL has been calculated with the redundant scaling. The arrows indicate the step size and the input code corresponds to the CDAC control signals. Keep in mind that the actual places where the level falls depends on the scaling used.



(a) *Generic 1* (b) *Generic 2*

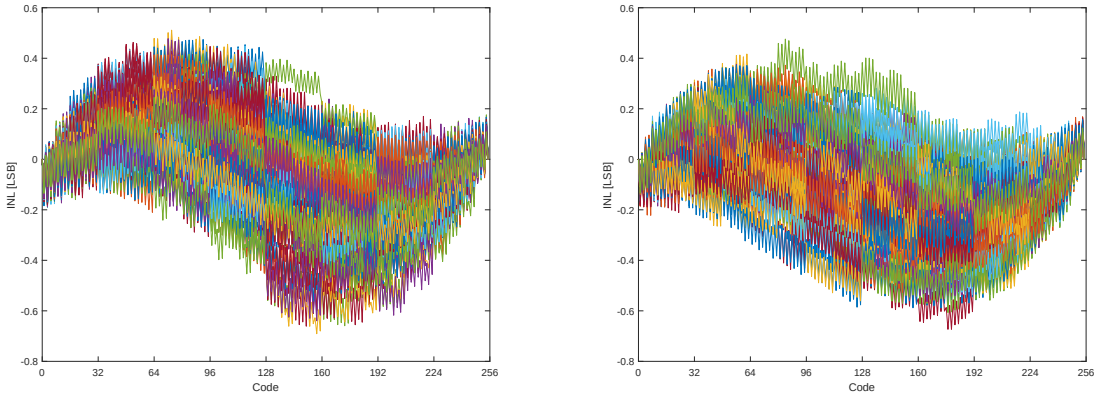
Figure 7.7: Histogram of the distribution of the maximum DNL for the scalings *Generic 1* and for *Generic 2* based on 250 MonteCarlo simulations. The y-axis shows the number of runs that falls into each bin.

the standard deviation is 0.0766 LSB while for *Generic 2* the mean is 0.248 LSB and the standard deviation is 0.08 LSB.

Figure 7.8 shows the INL for *Generic 1* and for *Generic 2*. Here it can be seen that the trend of the INL is sine shaped. The source of this INL shape has been identified as variations in the reference voltage. These variations are caused by a dependency between the leakage current through the CDAC unit cells at switching and the switching sequence performed at the stages before.

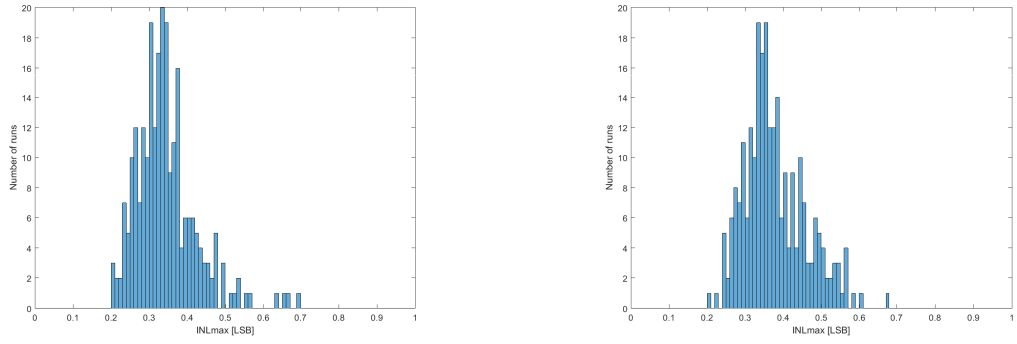
The distribution of the maximum INL is shown in figure 7.9 (a) and (b) for the scalings *Generic 1* and *Generic 2* respectively. Here it can be seen that the distribution of the maximum INL for *Generic 1* is smaller than for *Generic 2*. The mean maximum INL for *Generic 1* is 0.344 LSB and it has a standard deviation of 0.08 LSB. For *Generic 2* the corresponding values are 0.377 LSB and 0.081 LSB respectively. *Generic 1* has been used for all the other evaluations of the system.

Both the DNL and INL has been generated by simulating the CDAC, sweeping through all its inputs one by one. These tests where performed with a simulated reference buffer



(a) *Generic 1* (b) *Generic 2*

Figure 7.8: INL for the scalings *Generic 1* and for *Generic 2* for 250 MonteCarlo simulations. Due to the redundancy, the code is not directly translatable into a voltage.



(a) *Generic 1* (b) *Generic 2*

Figure 7.9: Histogram of the distribution of the maximum INL for the scalings *Generic 1* and for *Generic 2* based on 250 MonteCarlo simulations. The y-axis shows the number of runs that falls into each bin.

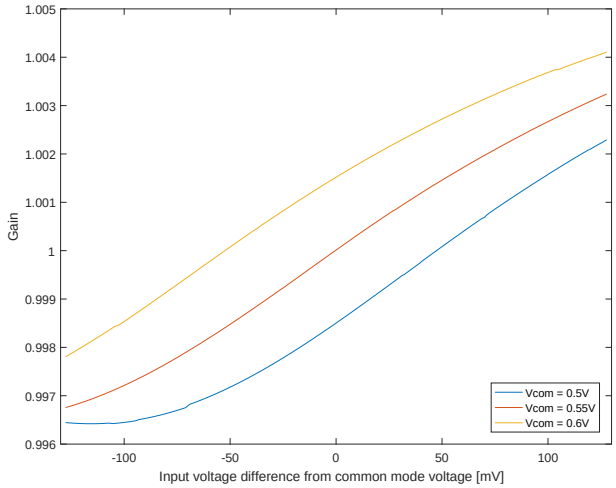


Figure 7.10: Sampling switch gain.

to get the same behaviour of changed voltage. The CDAC was also reset after each input code to ensure that the codes would not affect each other. The reference voltage was set to 400 mV to achieve the full input range.

7.3 Track and Hold

For the T&H the most important measure is the linearity or gain which should be as constant as possible. In figure 7.10 the gain for the T&H is plotted for one of the differential legs at three different common mode voltages. The input voltages given in the figure are relative to their respective common mode voltage. It can be seen here that a constant gain has not been achieved. Instead, the gains are slightly sloped and taper off for both the smallest and largest inputs. This tapering effect can especially be seen for low input voltages at $V_{com} = 0.5\,\mathrm{V}$ and for high input voltages at $V_{com} = 0.6\,\mathrm{V}$. The source of this tapering effect is the transistor conductance that is reduced due to the low and high voltages. The slope effect comes from the clock leakage which is lowest at the input which it was optimized for, that is at $V_{com} = 0.55\,\mathrm{V}$ with a 0 V input voltage.

It should be remembered that these results are for only one of the differential legs. For differential inputs, one leg will have a voltage higher than V_{com} while the other leg has input voltage which is lower than V_{com} resulting in a differential gain that is u-shaped.

The gain of the T&H has been evaluated by inputting a ramp input and sample it at various points in time. The output and input has then been compared to each other in order to

7.4 Complete design

The evaluation of the whole design focuses on three measures, the SNDR, the linearity, and the power consumption. The result from these evaluations will be presented below. All simulations of the whole system use the scaling *Generic 1*. This scaling where chosen due to its higher linearity compared to *Generic 2* while the difference in terms of correctness where small.

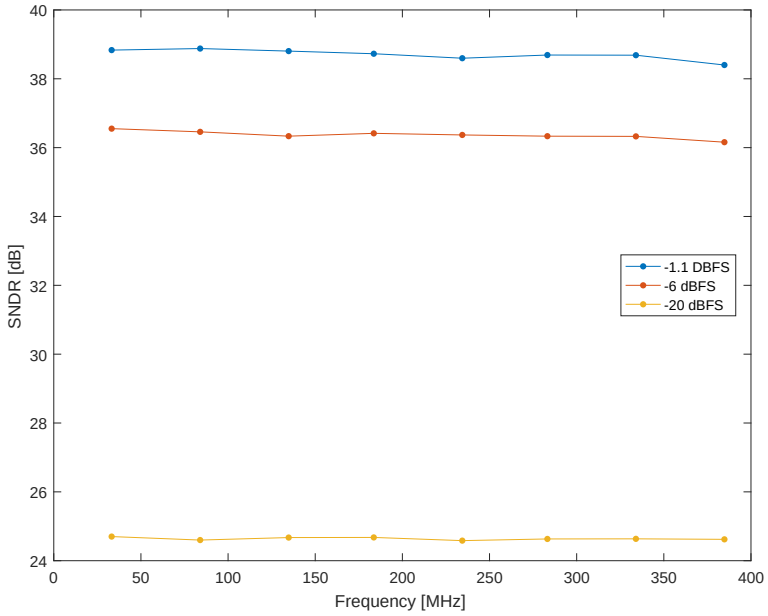


Figure 7.11: Simulated SNDR for different frequencies and amplitude at a sample rate of 800 MS/s.

7.4.1 SNDR

SNDR and SFDR are both important measures of the dynamic performance for the ADC. In figure 7.11 simulated SNDR is plotted for three input amplitudes and various frequencies. It can be seen in the figure that the highest SNDR is achieved for the highest input amplitude, the SNDR shows no dependency on the input frequency. The SNDR for the highest input amplitude -1dBFS ranges from 38.4 dB to 38.9 dB which corresponds to an ENOB of $6.09 - 6.17$ bits.

The SFDR is plotted in figure 7.12 for the same frequencies and amplitudes as in figure 7.11. It can be seen here that the SFDR for most frequencies is larger at an input amplitude of -6dBFS than at -1.1dBFS. The lowest SFDR is 44.8 dB at an input amplitude of -1.1dBFS.

The reason that the SFDR is several dB larger than the SNDR comes from their definition. The SFDR is defined as the distance between the signal and the largest spur or harmonic while SNDR is the ratio between the signal and noise, as mentioned in section 2.11. The difference indicates that there are several spurs or harmonics with similar amplitude as the one limiting the SFDR, thereby causing the lower SNDR. This effect can be seen in figure 7.13 where the spectrum for a signal at two amplitudes is shown.

Figure 7.13 shows the simulated output frequency spectrum for two input amplitudes at an input frequency of 335 MHz. The figure shows not only the input signal, it also includes the folded 2-12 harmonics as well as all other noise. The spectrum contains for both cases two significant noise spikes which limit the performance of the converter. It can also be seen that the effect is worse for the high input amplitude shown in figure 7.13 (a). These two undesired frequencies are not related to any of the harmonics since they change their frequency depending on the input amplitude which can be seen in the figure. The source of these spurs has not been discovered.

All SNDR and SFDR results are based on simulation results with 4096 sampled data points. This data has been generated by simulating the converter together with its test circuitry at various different frequency and amplitude combinations. During these sim-

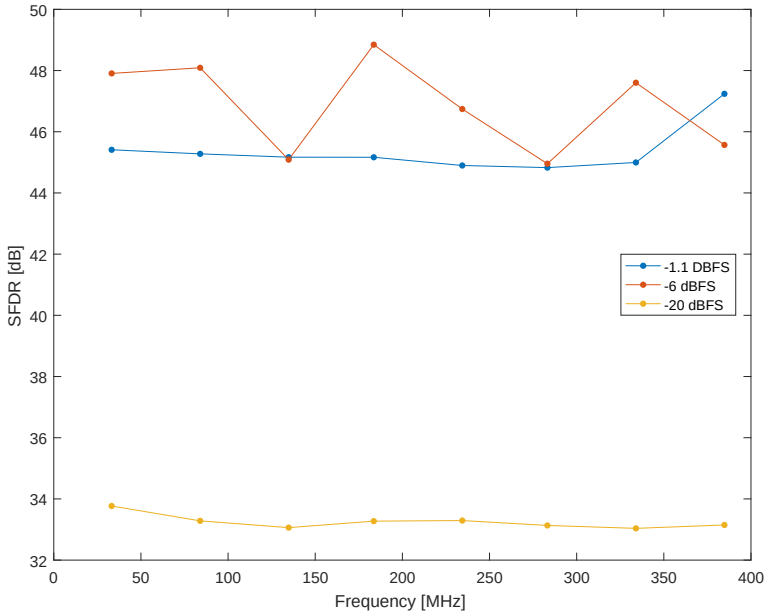
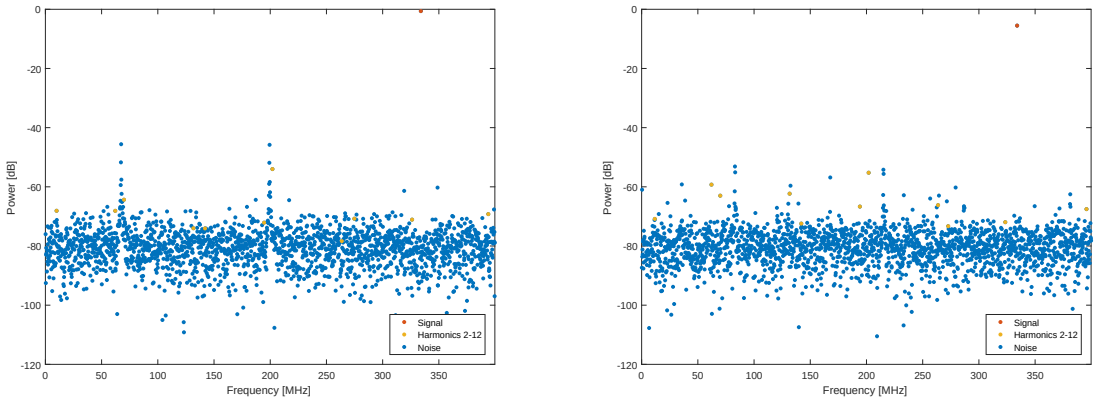


Figure 7.12: Simulated SFDR for different frequencies and amplitudes at a sample rate of 800 MS/s.



(a) $P_{sig} = -1.1$dBFS (b) $P_{sig} = -6$dBFS

Figure 7.13: Frequency spectrum for the sampled data at two amplitudes, -1.1dBFS (a) and -6dBFS (b) at a signal frequency of $f_{sig} = 335$ MHz. The spectrum includes the signal in red, the folded harmonics of order 2-12 in yellow and the rest of the noise in blue.

Table 7.1: Comparison of different SAR converters presented in literature with the one proposed here.

Work	[36]	[39]	[40]	[41]	This work
Architecture	SAR, alternating comparators	SAR, 2-bits/stage	SAR, 2-bits/stage	SAR, 2-bits/stage	SAR, alternating comparators
Technology	32 nm SOI CMOS	28 nm CMOS	65 nm	65 nm	28 nm FD-SOI CMOS
Resolution	8 bit	8 bit	8 bit	8-bit	8-bit
SNDR	39.3 dB	45.2 dB	47.7 dB	44.5 dB	38.4 dB
Sampling rate	1.2 GS/s	750 MS/s	600 MS/s	400 MS/s	800 MS/s
Power consumption	3.06 mW	4.5 mW	4.02 mW	4 mW	1.1 mW
FoMW	34 fJ/conv. step	41 fJ/conv. step	34.5 fJ/conv. step	73 fJ/conv. step	20.3 fJ/conv. step

ulations MonteCarlo mismatch variations were enabled to ensure that the circuit is not ideal.

7.4.2 Linearity

Due to simulation problems it has not been possible to provide the linearity for the whole design.

7.4.3 Power

The simulated power consumption for the ADC is 1.1 mW for the whole design. The memory with its state machine and the clock generator are the parts of the converter which consumes most power, 330 µW and 290 µW respectively. The CDAC consumes 170 µW and the two comparators together consume 170 µW. Both the T&H and the reference voltage buffer consume 55 µW each. The power consumption for the rest of the circuit is negligible.

With a sample rate of 800 MS/s and a power consumption of 1.1 mW, the energy per conversion will be 1.38 pJ. This energy consumption together with the ENOB calculated above then gives FoMW = 20.3 fJ/conversion_step. This FoMW is comparable to the highest reported for Nyquist rate converters with a sample rate over 250 MS/s [10]. The design also achieves a FoMS = 154 dB.

7.4.4 Comparison with literature

In table 7.1 is the proposed converter presented in this work compared to other converters presented in literature. The included converters are all based on a single converter of SAR topology. It can here be seen that the proposed topology outperforms the other designs in terms of both power consumption and FoMW. However, note that the results presented for this work are based on simulations while the results for the works compared with are based on measurements.

8

Discussion

This discussion chapter will be divided into three sections covering the design, recommendations for future work, and an evaluation of my work process.

8.1 Design

First of all it must be stated that since all results are based on circuit level simulations rather than on extracted parasitic values based on the layout, a small drop in the performance can be expected. These results shall however give a indication about where the actual performance of the design lies.

8.1.1 Topology

Several choices have been made during the design which relate to the topology of the converter. The most important decision here is the use of the alternating comparators. These bring additional complexity both in terms of noise sources and of required circuitry. This decision was brought up several times during the project and it resulted in the same answer each time; it would not be possible to achieve the desired sample rate using a single comparator.

The redundancy used in the CDAC also complicates the circuit operation. The redundancy was not as essential as the dual comparator structure for achieving the required speed.

8.1.2 Comparator

The comparator design shows a very low offset voltage which is essential for achieving a high SNDR when alternating comparators are used. While most of the MonteCarlo runs result in an offset voltage close to 0, there are still some cases where the offset voltage is significantly larger. These cases are however few and they are a result of the trade-off between the maximum offset that can be managed and the effects which the compensation has on the comparator operation. The offset voltage was in this case considered sufficiently low.

The kickback noise coming from the comparator is another factor which limits the comparator performance. As could be seen in figure 7.3 (b), the kickback noise was still quite large after compensation. This is a result of the complexity in reducing the kickback noise where few options where suitable. The chosen reduction method was chosen as the optimum in the actual case. However there may be better ways of reducing the effects of kickback noise than the one used here, and this needs further investigation.

A better balance between speed and kickback noise may be possible since the kickback noise increases with larger input transistors just as speed does. The improvement of such an optimization is however considered small.

8.1.3 CDAC

When it comes to the scaling of the CDAC it can be argued that the neither of the redundant options investigated was an option, since they all had a significantly lower correctness than binary scaling. However, this is the drawback with reducing the required settling time: something needs to be sacrificed. The problem is however not that severe, since the erroneous results will typically just be 1 LSB off, thereby only having a small effect on the performance.

Regarding the sine shaped behaviour of the INL, it can be argued that the size of the buffer voltage should be increased since the INL is not within the specifications. However the effect off increasing this buffer size may be that it occupies an area which is several times as large as the area required by the rest of the converter.

8.1.4 Whole design

When it comes to the scaling, the decision was to use *Generic 1* rather than *Generic 2*. *Generic 2* showed a higher correctness in terms of delivering the correct output at the same time as the INL improvement where rather small. The reason for not choosing *Generic 2* was its dependence of half the reference voltage at multiple stages which has its effects on the INL. The problem with the increased $V_{ref}/2$ dependence is that due to the additional series resistance for the resistive network creating this fractional voltage, a longer settling time is required before sufficient settling is achieved; this is not shown in the INL curves since they builds on achieving complete settling at each level before another one is switched.

From the results it can be seen that the achieved SNDR is less then what was specified originally. The specified SNDR was however an ambitious goal that was significantly higher than other works presented. The value shown are what has been reported in literature for this type of topology.

When it comes to the power consumption, it can be seen that the effect of not pushing for speed pays off in terms of a low power consumption. This power consumption is however based on simulated circuits at schematic level, and it can be expected to increase when parasitic effects from layout are included. The value is however expected to give a good indication of where the expected power consumption lies.

In general, since the results presented in this work are based on schematic level simulations the comparison to literature will not be completely fair. The comparison however gives a pointer about what the expected results and how these are compared to other implementations presented.

8.2 Recommendations for future work

For a potential continuation of this project the next obvious step would be to continue with the layout. This is to make is possible to further analyse the performance of the design. Before this is done there are however some parts of the circuit that needs some attention as stated above.

First of all the kickback noise reduction technique will need further analysis, both on the comparator as well as on the T&H, since the same technique is used at both places.

Other alternatives would need to be analysed and evaluated to see if there is a better way of reducing the kickback noise. For the T&H, it may be beneficial to look for an alternative circuit which incorporates kickback noise reduction.

The reset switch used in the CDAC should also be revisited and then especially its sizing, since it is a major contributor to the CDAC parasitic capacitance.

Lastly, as mentioned during the implementation, the sizing of the reference buffer will need to be revisited. This however needs to be done after an initial layout to trade off buffer size for area.

8.3 Work process

According the initial time plan, schematic, schematic-based simulations, layout, and post-layout simulations were supposed to be performed. However as stated in the project plan, the layout of all post-layout simulations of the whole design where only supposed to be performed if time permitted.

The original plan for the design was to first do a first version of the implementation with the key parts of the circuit and then do a layout of this before improvements on the design where implemented based on the time left. This however had to change when it was realized that the design was more complex than expected. It was also realized that due to the parasitic and mismatch effects in the 28 nm process, calibration and compensation circuits required even in the first iteration. This is since the complexity of adding them afterwards would result in significant redesign of several parts of the system.

As for the design flow which was based on starting with the key components of the system and the continued by putting it all together, the problem was that the components were put together into a complete system at a too late stage in the project. This resulted in problems related to the sizing of critical parts of the system which required. As a result, the actual project differed a bit from the original plan.

9

Conclusion

In this work, the design of a 800 MS/s 8-bit ADC is presented. The whole process from literature study to schematic-based circuit implementation has been documented. The converter has been implemented in a 28 nm FD-SOI CMOS process.

The simulation results show an SNDR which is somewhat lower than the specified value. The power consumption of the proposed converter is however significantly smaller than other designs shown in literature.

Due to the complexity in testing the performance of the converter, additional circuitry has been added to improve the testability of the design when it has been fabricated. To further improve the testability of the implemented design the ability to activate and deactivate the offset calibration has been included.

The design has been based on a previously known architecture and it has been implemented in a 28 nm FD-SOI CMOS process. Simulation results shown a SNDR = 38.4 dB at a sample rate of 800 MS/s with a power consumption of 1.1 mW resulting in a FoMW = 20.3 fJ/conversion step.

Bibliography

[1] L. Kull, T. Toifl, M. Schmatz, P. Francese, C. Menolfi, M. Brandli, M. Kossel, T. Morf, T. Andersen, and Y. Leblebici, "A 3.1 mW 8b 1.2 GS/s Single-Channel Asynchronous SAR ADC With Alternate Comparators for Enhanced Speed in 32 nm Digital SOI CMOS," *IEEE Journal of Solid-State Circuits*, vol. 48, no. 12, pp. 3049–3058, Dec 2013.

[2] S.-S. Wong, U.-F. Chio, Y. Zhu, S.-W. Sin, S.-P. U, and R. Martins, "A 2.3 mW 10-bit 170 MS/s Two-Step Binary-Search Assisted Time-Interleaved SAR ADC," *IEEE Journal of Solid-State Circuits*, vol. 48, no. 8, pp. 1783–1794, Aug 2013.

[3] P. M. Figueiredo and J. C. Vital, "Kickback noise reduction techniques for CMOS latched comparators," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 53, no. 7, pp. 541–545, July 2006.

[4] http://www.st.com, "Learn More About FD-SOI." [Online]. Available: http://www.st.com/content/st_com/en/about/innovation---technology/FD-SOI/learn-more-about-fd-soi.html [Accessed: 2016-06-01]

[5] M. Miyahara, Y. Asada, D. Paik, and A. Matsuzawa, "A low-noise self-calibrating dynamic comparator for high-speed ADCs," in *2008 IEEE Asian Solid-State Circuits Conference (A-SSCC '08)*, Nov 2008, pp. 269–272.

[6] E. Larsson, O. Edfors, F. Tufvesson, and T. Marzetta, "Massive MIMO for next generation wireless systems," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 186–195, February 2014.

[7] P. Suryasarman and A. Springer, "A Comparative Analysis of Adaptive Digital Predistortion Algorithms for Multiple Antenna Transmitters," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 62, no. 5, pp. 1412–1420, May 2015.

[8] F. Ghannouchi and O. Hammi, "Behavioral modeling and predistortion," *IEEE Microwave Magazine*, vol. 10, no. 7, pp. 52–64, Dec 2009.

[9] L. Kull, T. Toifl, M. Schmatz, P. Francese, C. Menolfi, M. Braendli, M. Kossel, T. Morf, T. Andersen, and Y. Leblebici, "A 35mW 8b 8.8 GS/s SAR ADC with low-power capacitive reference buffers in 32nm Digital SOI CMOS," in *2013 Symposium on VLSI Circuits (VLSIC)*, June 2013, pp. C260–C261.

[10] B. Murmann. (2015) ADC Performance Survey 1997-2015. [Online]. Available: http://web.stanford.edu/~murmann/adcsurvey.html [Accessed: 2016-01-21]

[11] F. Maloberti, *Data Converters*. Springer, 2007.

[12] M. Gustavsson, J. J. Wikner, and N. N. Tan, *CMOS Data Converter for Communications*. Boston, MA: Springer US, 2002, ch. Analog Functional Blocks, pp. 139–175.

[13] B. E. Jonsson, "Area efficiency of ADC architectures," in *20th European Conference on Circuit Theory and Design (ECCTD), 2011*, Aug 2011, pp. 560–563.

[14] M. Furuta and T. Itakura, "Trends in the design of high-speed, low-power analog-to-digital converters," in *2015 IEEE International Symposium on Radio-Frequency Integration Technology (RFIT)*, Aug 2015, pp. 169–171.

[15] Y. Huang, H. Schleifer, and D. Killat, "Design and analysis of novel dynamic latched comparator with reduced kickback noise for high-speed ADCs," in *2013 European Conference on Circuit Theory and Design (ECCTD)*, Sept 2013, pp. 1–4.

[16] R. Palani and R. Harjani, "A 220-MS/s 9-Bit 2X Time-Interleaved SAR ADC With a 133-fF Input Capacitance and a FOM of 37 fJ/conv in 65-nm CMOS," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 62, no. 11, pp. 1053–1057, Nov 2015.

[17] Y. Sinangil and A. Chandrakasan, "An embedded energy monitoring circuit for a 128kbit SRAM with body-biased sense-amplifiers," in *2012 IEEE Asian Solid State Circuits Conference (A-SSCC)*, Nov 2012, pp. 69–72.

[18] C.-C. Liu, Y.-T. Huang, G.-Y. Huang, S.-J. Chang, C.-M. Huang, and C.-H. Huang, "A 6-bit 220-MS/s time-interleaving SAR ADC in 0.18-μm digital CMOS process," in *International Symposium onVLSI Design, Automation and Test, 2009. VLSI-DAT '09*, April 2009, pp. 215–218.

[19] A. Bekal, R. Joshi, M. Goswami, B. Singh, and A. Srivatsava, "An Improved Dynamic Latch Based Comparator for 8-Bit Asynchronous SAR ADC," in *2015 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, July 2015, pp. 178–182.

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

[21] J.-H. Tsai, H.-H. Wang, Y.-C. Yen, C.-M. Lai, Y.-J. Chen, P.-C. Huang, P.-H. Hsieh, H. Chen, and C.-C. Lee, "A 0.003 mm^2 10 b 240 MS/s 0.7 mW SAR ADC in 28 nm CMOS With Digital Error Correction and Correlated-Reversed Switching," *IEEE Journal of Solid-State Circuits*, vol. 50, no. 6, pp. 1382–1398, June 2015.

[22] S. Lee, A. Chandrakasan, and H.-S. Lee, "A 1 GS/s 10b 18.9 mW Time-Interleaved SAR ADC With Background Timing Skew Calibration," *IEEE Journal of Solid-State Circuits*, vol. 49, no. 12, pp. 2846–2856, Dec 2014.

[23] C.-C. Liu, S.-J. Chang, G.-Y. Huang, Y.-Z. Lin, C.-M. Huang, C.-H. Huang, L. Bu, and C.-C. Tsai, "A 10b 100MS/s 1.13mW SAR ADC with binary-scaled error compensation," in *2010 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*, Feb 2010, pp. 386–387.

[24] T. Ogawa, H. Kobayashi, M. Hotta, Y. Takahashi, H. San, and N. Takai, "SAR ADC algorithm with redundancy," in *2008 IEEE Asia Pacific Conference on Circuits and Systems (APCCAS)*, Nov 2008, pp. 268–271.

[25] S. Aghaie, J. H. Mueller, R. Wunderlich, and S. Heinen, "Design of a low-power calibratable charge-redistribution SAR ADC," in *2014 10th Conference on Ph.D. Research in Microelectronics and Electronics (PRIME)*, June 2014, pp. 1–4.

[26] S. Kundu, E. Alpman, J.-L. Lu, H. Lakdawala, J. Paramesh, B. Jung, S. Zur, and E. Gordon, "A 1.2 V 2.64 GS/s 8 bit 39 mW Skew-Tolerant Time-interleaved SAR ADC in 40 nm Digital LP CMOS for 60 GHz WLAN," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 62, no. 8, pp. 1929–1939, Aug 2015.

[27] G. Souliotis, C. Laoudias, and N. Terzopoulos, "An Offset Cancelation Technique for Latch Type Sense Amplifiers," *RADIOENGINEERING*, vol. 23, no. 4, pp. 1121–1129, 2014.

[28] P. Pouya, A. Ghasemi, and H. Aminzadeh, "A low-voltage high-speed high-linearity MOSFET-only analog bootstrapped switch for sample-and-hold circuits," in *2015 2nd International Conference on Knowledge-Based Engineering and Innovation (KBEI)*, Nov 2015, pp. 418–421.

[29] J. Fang, S. Thirunakkarasu, X. Yu, F. Silva-Rivas, K. Y. Kim, C. Zhang, and F. Singor, "A 5GS/s 10b 76mW time-interleaved SAR ADC in 28nm CMOS," in *2015 IEEE Custom Integrated Circuits Conference (CICC)*, Sept 2015, pp. 1–4.

[30] R. Walden, "Analog-to-digital converter technology comparison," in *16th Annual Gallium Arsenide Integrated Circuit (GaAs IC) Symposium, 1994. Technical Digest 1994.*, Oct 1994, pp. 217–219.

[31] B. Murmann, *Analog Circuit Design: RF Circuits: Wide band, Front-Ends, DAC's, Design Methodology and Verification for RF and Mixed-Signal Systems, Low Power and Low Voltage*. Dordrecht: Springer Netherlands, 2006, ch. LIMITS ON ADC POWER DISSIPATION, pp. 351–367.

[32] R. Schreier and G. C. Temes, *Example Modulator Systems*. Wiley-IEEE Press, 2005, pp. 283–363.

[33] N. Planes, O. Weber, V. Barral, S. Haendler, D. Noblet, D. Croain, M. Bocat, P. O. Sassoulas, X. Federspiel, A. Cros, A. Bajolet, E. Richard, B. Dumont, P. Perreau, D. Petit, D. Golanski, C. Fenouillet-Béranger, N. Guillot, M. Rafik, V. Huard, S. Puget, X. Montagner, M. A. Jaud, O. Rozeau, O. Saxod, F. Wacquant, F. Monsieur, D. Barge, L. Pinzelli, M. Mellier, F. Boeuf, F. Arnaud, and M. Haond, "28nm FDSOI technology platform for high-speed low-voltage digital applications," in *2012 Symposium on VLSI Technology (VLSIT)*, June 2012, pp. 133–134.

[34] A. K. Sharma and A. Teverovsky, "Reliability evaluation of fully depleted SOI (FD-SOI) technology for space applications," *NASA Electronics Parts and Packaging Program (NEPP) report*, vol. 14, no. 9, 2001.

[35] S. S. Suryagandh, M. Garg, and J. C. S. Woo, "Comparison between bulk and SOI MOSFETs for sub-100nm mixed mode applications," in *33rd Conference on European Solid-State Device Research, 2003. ESSDERC '03.*, Sept 2003, pp. 423–426.

[36] L. Kull, T. Toifl, M. Schmatz, P. Francese, C. Menolfi, M. Braendli, M. Kossel, T. Morf, T. Andersen, and Y. Leblebici, "A 3.1mW 8b 1.2GS/s single-channel asynchronous SAR ADC with alternate comparators for enhanced speed in 32nm digital

SOI CMOS," in *2013 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*, Feb 2013, pp. 468–469.

[37] L. Kull, T. Toifl, M. Schmatz, P. Francese, C. Menolfi, M. Braendli, M. Kossel, T. Morf, T. Andersen, and Y. Leblebici, "A 90GS/s 8b 667mW 64x interleaved SAR ADC in 32nm digital SOI CMOS," in *2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*, Feb 2014, pp. 378–379.

[38] L. Kull, J. Pliva, T. Toifl, M. Schmatz, P. Francese, C. Menolfi, M. Braendli, M. Kossel, T. Morf, T. Andersen, and Y. Leblebici, "A 110 mW 6 bit 36 GS/s interleaved SAR ADC for 100 GBE occupying 0.048 mm^2 in 32 nm SOI CMOS," in *2014 IEEE Asian Solid-State Circuits Conference (A-SSCC)*, Nov 2014, pp. 89–92.

[39] Y. C. Lien, "A 4.5-mW 8-b 750-MS/s 2-b/step asynchronous subranged SAR ADC in 28-nm CMOS technology," in *2012 Symposium on VLSI Circuits (VLSIC)*, June 2012, pp. 88–89.

[40] J. Mei, J. Xiang, H. Chen, F. Ye, and J. Ren, "A 4-mW8-b 600-MS/s 2-b-per-cycle SAR ADC with a capacitive DAC," in *2013 IEEE 10th International Conference on ASIC (ASICON)*, Oct 2013, pp. 1–4.

[41] H. Wei, C. H. Chan, U. F. Chio, S. W. Sin, U. Seng-Pan, R. Martins, and F. Maloberti, "A 0.024mm2 8b 400MS/s SAR ADC with 2b/cycle and resistive DAC in 65nm CMOS," in *2011 IEEE International Solid-State Circuits Conference*, Feb 2011, pp. 188–190.

[42] C. H. Wang, "Identification and measurement of scaling-dependent parasitic capacitances of small-geometry MOSFET's," *IEEE Transactions on Electron Devices*, vol. 43, no. 6, pp. 965–972, Jun 1996.

[43] B. J. Sheu and C. Hu, "Switch-induced error voltage on a switched capacitor," *IEEE Journal of Solid-State Circuits*, vol. 19, no. 4, pp. 519–525, Aug 1984.

[44] G. Tretter, D. Fritsche, C. Carta, and F. Ellinger, "10-GS/s track and hold circuit in 28 nm CMOS," in *2013 International Semiconductor Conference Dresden-Grenoble (ISCDG)*, Sept 2013, pp. 1–3.

[45] D. Stepanović and B. Nikolić, "A 2.8 GS/s 44.6 mW Time-Interleaved ADC Achieving 50.9 dB SNDR and 3 dB Effective Resolution Bandwidth of 1.5 GHz in 65 nm CMOS," *IEEE Journal of Solid-State Circuits*, vol. 48, no. 4, pp. 971–982, April 2013.

[46] Y. Zhu, C.-H. Chan, S.-P. U, and R. Martins, "An 11b 900 MS/s time-interleaved sub-ranging pipelined-SAR ADC," in *ESSCIRC 2014 - 40th European Solid State Circuits Conference (ESSCIRC)*, Sept 2014, pp. 211–214.

[47] J. N. nez, A. J. Ginés, E. J. Peraías, and A. Rueda, "Low-jitter differential clock driver circuits for high-performance high-resolution ADCs," in *2015 Conference on Design of Circuits and Integrated Systems (DCIS)*, Nov 2015, pp. 1–4.

[48] J. N. nez, A. J. Ginés, E. J. Peralías, and A. Rueda, "An approach to the design of low-jitter differential clock recovery circuits for high performance ADCs," in *2015 IEEE 6th Latin American Symposium on Circuits Systems (LASCAS)*, Feb 2015, pp. 1–4.

A

Evaluated scalings

Table A.1: Different scalings evaluated during the design of the CDAC. The scaling are presented with their q and p-vector together with their maximum t/τ.

Index	q									p									t/τ
	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	
1	-	-	-	-	-	-	-	-	-	128	64	32	16	8	4	2	1	-	4.852
2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
3	0	16	4	2	4	0	0	0	0	128	64	24	18	10	4	4	2	1	2.0794
4	0	12	6	2	4	0	0	0	0	128	64	26	16	10	4	4	2	1	2.0794
5	0	8	8	2	4	0	0	0	0	128	64	28	14	10	4	4	2	1	2.0794
6	0	8	4	4	4	0	0	0	0	128	64	28	16	8	4	4	2	1	2.0794
7	0	20	2	2	4	0	0	0	0	128	64	22	20	10	4	4	2	1	2.1748
8	0	12	2	4	4	0	0	0	0	128	64	26	18	8	4	4	2	1	2.3418
9	0	36	2	2	2	0	0	0	0	128	64	14	24	12	6	4	2	1	2.4849
10	0	32	4	2	2	0	0	0	0	128	64	16	22	12	6	4	2	1	2.4849
11	0	28	6	2	2	0	0	0	0	128	64	18	20	12	6	4	2	1	2.4849
12	0	28	2	4	2	0	0	0	0	128	64	18	22	10	6	4	2	1	2.4849
13	0	24	8	2	2	0	0	0	0	128	64	20	18	12	6	4	2	1	2.4849
14	0	24	4	4	2	0	0	0	0	128	64	20	20	10	6	4	2	1	2.4849
15	0	20	10	2	2	0	0	0	0	128	64	22	16	12	6	4	2	1	2.4849
16	0	20	6	4	2	0	0	0	0	128	64	22	18	10	6	4	2	1	2.4849
17	0	20	2	6	2	0	0	0	0	128	64	22	20	8	6	4	2	1	2.4849
18	0	16	12	2	2	0	0	0	0	128	64	24	14	12	6	4	2	1	2.4849
19	0	16	8	4	2	0	0	0	0	128	64	24	16	10	6	4	2	1	2.4849
20	0	16	4	6	2	0	0	0	0	128	64	24	18	8	6	4	2	1	2.4849
21	0	12	14	2	2	0	0	0	0	128	64	26	12	12	6	4	2	1	2.4849
22	0	12	10	4	2	0	0	0	0	128	64	26	14	10	6	4	2	1	2.4849
23	0	12	6	6	2	0	0	0	0	128	64	26	16	8	6	4	2	1	2.4849
24	0	12	2	8	2	0	0	0	0	128	64	26	18	6	6	4	2	1	2.4849
25	0	8	16	2	2	0	0	0	0	128	64	28	10	12	6	4	2	1	2.4849
26	0	8	12	4	2	0	0	0	0	128	64	28	12	10	6	4	2	1	2.4849
27	0	8	8	6	2	0	0	0	0	128	64	28	14	8	6	4	2	1	2.4849
28	0	8	4	8	2	0	0	0	0	128	64	28	16	6	6	4	2	1	2.4849
29	0	4	22	0	2	0	0	0	0	128	64	30	6	14	6	4	2	1	2.6548
30	0	4	18	2	2	0	0	0	0	128	64	30	8	12	6	4	2	1	2.6548
31	0	4	14	4	2	0	0	0	0	128	64	30	10	10	6	4	2	1	2.6548
32	0	4	10	6	2	0	0	0	0	128	64	30	12	8	6	4	2	1	2.6548

33	0	4	10	2	4	0	0	0	0	128	64	30	12	10	4	4	2	1	2.6548
34	0	4	6	8	2	0	0	0	0	128	64	30	14	6	6	4	2	1	2.6548
35	0	4	6	4	4	0	0	0	0	128	64	30	14	8	4	4	2	1	2.6548
36	0	4	2	10	2	0	0	0	0	128	64	30	16	4	6	4	2	1	2.6548
37	0	4	2	6	4	0	0	0	0	128	64	30	16	6	4	4	2	1	2.6548
38	0	4	2	6	0	2	0	0	0	128	64	30	16	6	6	2	2	1	2.6548
39	0	4	2	2	6	0	0	0	0	128	64	30	16	8	2	4	2	1	2.6548
40	0	4	2	2	2	2	0	0	0	128	64	30	16	8	4	2	2	1	2.6548
41	0	28	2	8	0	0	0	0	0	128	64	18	22	8	8	4	2	1	2.7726
42	0	24	4	8	0	0	0	0	0	128	64	20	20	8	8	4	2	1	2.7726
43	0	20	6	8	0	0	0	0	0	128	64	22	18	8	8	4	2	1	2.7726
44	0	20	2	10	0	0	0	0	0	128	64	22	20	6	8	4	2	1	2.7726
45	0	16	8	8	0	0	0	0	0	128	64	24	16	8	8	4	2	1	2.7726
46	0	16	4	10	0	0	0	0	0	128	64	24	18	6	8	4	2	1	2.7726
47	0	12	10	8	0	0	0	0	0	128	64	26	14	8	8	4	2	1	2.7726
48	0	12	6	10	0	0	0	0	0	128	64	26	16	6	8	4	2	1	2.7726
49	0	12	2	12	0	0	0	0	0	128	64	26	18	4	8	4	2	1	2.7726
50	0	12	2	4	0	2	0	0	0	128	64	26	18	8	6	2	2	1	2.7726
51	0	8	20	0	2	0	0	0	0	128	64	28	8	14	6	4	2	1	2.7726
52	0	8	12	8	0	0	0	0	0	128	64	28	12	8	8	4	2	1	2.7726
53	0	8	8	10	0	0	0	0	0	128	64	28	14	6	8	4	2	1	2.7726
54	0	8	4	12	0	0	0	0	0	128	64	28	16	4	8	4	2	1	2.7726
55	0	8	4	4	0	2	0	0	0	128	64	28	16	8	6	2	2	1	2.7726
56	0	4	14	8	0	0	0	0	0	128	64	30	10	8	8	4	2	1	2.7726
57	0	4	10	10	0	0	0	0	0	128	64	30	12	6	8	4	2	1	2.7726
58	0	4	6	12	0	0	0	0	0	128	64	30	14	4	8	4	2	1	2.7726
59	0	4	6	4	0	2	0	0	0	128	64	30	14	8	6	2	2	1	2.7726
60	0	4	2	14	0	0	0	0	0	128	64	30	16	2	8	4	2	1	2.7726
61	0	36	2	6	0	0	0	0	0	128	64	14	24	10	8	4	2	1	2.9957
62	0	32	4	6	0	0	0	0	0	128	64	16	22	10	8	4	2	1	2.9957
63	0	28	6	6	0	0	0	0	0	128	64	18	20	10	8	4	2	1	2.9957
64	0	24	8	6	0	0	0	0	0	128	64	20	18	10	8	4	2	1	2.9957
65	0	20	10	6	0	0	0	0	0	128	64	22	16	10	8	4	2	1	2.9957
66	0	20	2	2	0	2	0	0	0	128	64	22	20	10	6	2	2	1	2.9957
67	0	16	12	6	0	0	0	0	0	128	64	24	14	10	8	4	2	1	2.9957
68	0	16	4	2	0	2	0	0	0	128	64	24	18	10	6	2	2	1	2.9957
69	0	12	18	0	2	0	0	0	0	128	64	26	10	14	6	4	2	1	2.9957
70	0	12	14	6	0	0	0	0	0	128	64	26	12	10	8	4	2	1	2.9957
71	0	12	6	2	0	2	0	0	0	128	64	26	16	10	6	2	2	1	2.9957
72	0	8	16	6	0	0	0	0	0	128	64	28	10	10	8	4	2	1	2.9957
73	0	8	8	2	0	2	0	0	0	128	64	28	14	10	6	2	2	1	2.9957
74	0	4	18	6	0	0	0	0	0	128	64	30	8	10	8	4	2	1	2.9957
75	0	4	14	0	4	0	0	0	0	128	64	30	10	12	4	4	2	1	2.9957
76	0	4	10	2	0	2	0	0	0	128	64	30	12	10	6	2	2	1	2.9957

B

Time plan

Table B.1: Time plan.

Time plan master thesis

Date: 2016-01-20											Version: 0.1												
Activities		Week number																					
No	Description	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	
	Deliverables																						
1	Planning report																						
2	Halftime report																						
3	Report to opponent																						
4	Opponate on report																						
5	Prepare presentation																						
6	Finalize report																						
	Meetings																						
7	Supervisor																						

8	Supervisor and Co-supervisor																					
	Seminars																					
9	Preparation for seminar 1																					
10	Seminar 1						Mo															
11	Seminar 2																We					
	Design																					
12	Study potential implementations																					
13	Get going with the 28nm process																					
14	Analyse requirements for system blocks																					
15	Circuit design																					
16	Design of verification setup																					
17	Circuit simulation																					
18	Layout																					
19	Post-layout simulation																					
20	Redesign and improvements																					

	Milestones																					
21	Block requirements descided				Fr																	
22	Circuit implemented							Fr														
23	Circuit verified										Th											
24	Layout implemented													Fr								
25	Implementation verified																	Fr				
	Tollgates																					
26	Planning report		Fr																			
27	Halftime report												Fr									
28	Report to opponent																				We	
29	Presentation																					We
30	Final report																					Fr
	Presentations																					
31	Presentation																					We
32	Opposition																					We