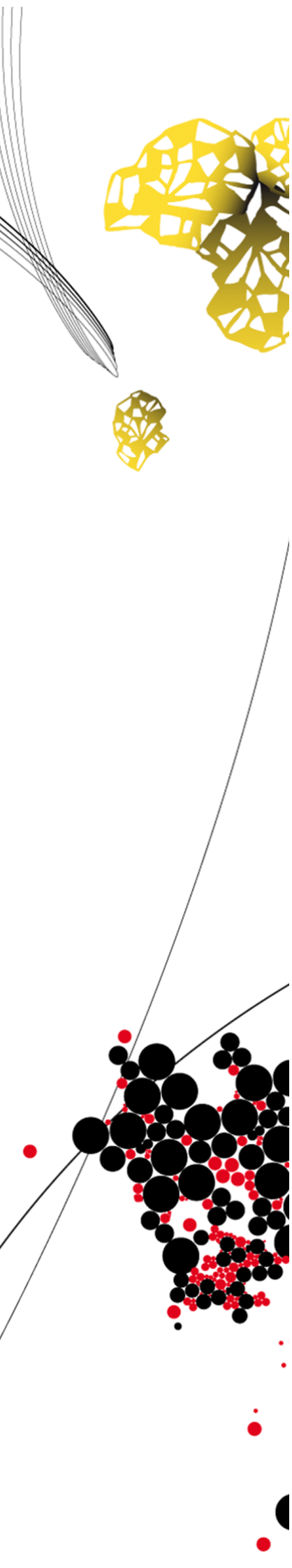

UNIVERSITY OF TWENTE.

Faculty of Electrical Engineering,
Mathematics & Computer Science

A TDC-assisted SAR ADC

Jelmer Bas Kosters

MSc. Thesis
October 2017

Supervisors
prof. dr. ir. B. Nauta
dr.ir. A.J Annema
H.S. Bindra, Msc.
ir. Bert Molenkamp

Report number: 067.3756
Chair of Integrated Circuit Design
Faculty of Electrical Engineering,
Mathematics and Computer Science
University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands

Abstract

For medium-speed, medium-resolution analog-to-digital conversion, a successive-approximation-register analog-to-digital converter (SAR ADC) is a popular choice due to its low figure of merit (FOM). Recent developments focus on hybrid architectures to improve the state-of-the-art.

In this thesis, an improvement to the SAR algorithm is proposed utilizing a TDC, to be used in a so-called TDC-assisted SAR ADC. By measuring the speed of the comparator during conversion steps, information about the sampled signal can be extracted and used to skip unnecessary steps in the SAR algorithm. The logarithmic voltage-to-time behavior of the comparator, in combination with the exponential scaled binary voltages quantized by the SAR algorithm, result in linearly spaced timings related to each SAR bit. Several algorithms that leverage the extra information are proposed, of which one is further researched. Based on calculations, it can decrease the average number of conversion steps in a 10-bit SAR ADC from 10 to 3, drastically improving the figure of merit due to reduced power consumption.

A comparator and 9-stage TDC are designed in a 65nm technology to test the feasibility of the idea. Additionally, a possible SAR logic implementation for the algorithm is given. Simulations show that the comparator requires 60fJ per comparison, and has an expected logarithmic decision speed. The TDC consumes 40fJ per comparison, and can be calibrated to match the LSB decision time of the comparator in multiple corners or mismatch scenarios. Examining the non-ideal voltage-to-time-to-digital behavior of the circuit, an input-dependent effective number of conversion steps is expected. Based on these results, expected FOMs are provided when possibly implemented in a future TDC-assisted SAR ADC.

Contents

Chapter 1

Introduction

The modern electronics space is still dominated by the use of CMOS integrated circuits. The current technology paradigm has shifted towards increasing integration, portability and power efficiency. At the same time, applications require ever-increasing speed and high resolution electronics to satisfy the needs of the user. A vivid example would be the smart-phone, with its multiple (highly sensitive) wireless systems, large computing power, high resolution camera(s), sensitive multi-touch digitizer, HiFi audio capability, accurate accelerometer, etc. These high-demanding systems all have to be powered by a small battery in a portable device.

As process technology improves and transistor feature sizes decrease, this trend does indeed give way to low-voltage, low-power, digital-heavy integrated circuitry. However, analog and mixed-signal circuits, necessary at the boundary of any embedded system, become increasingly difficult to design due to the decreased voltage headroom and increasing second-order effects. Therefore, there exists a large effort in the research community specifically on the topic of these designs.

One of these mixed-signal blocks needed in countless modern electronic system is the analog-to-digital converter (ADC). As shown in Figure 1.1, an ADC is a device that converts a physical (time continuous) analog signal to a (time discrete) digital signal to use in a processor or other digital circuit.

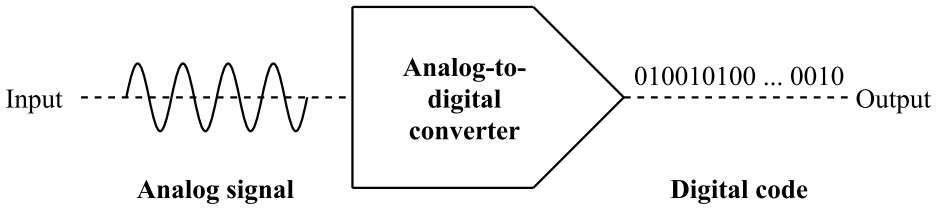

Figure 1.1: Basic principle of an ADC.

ADCs are needed in a wide range of applications and therefore have a wide range of specifications and resulting implementations. For example, low-speed high-resolution 'Sigma-Delta' converters such as [1] are found in audio applications ($192kHz$, 16-32-bit). On the other end of the spectrum, high-speed low-resolution 'Flash' converters such as [2] (2-8-bit, $< 100GHz$) are found in radar and optical transmission systems. This trade-off in speed and resolution is not arbitrary. It is generally considered to be bound by physical limitations described by Walden et al. in [3]. Figure 1.2 shows a range of ADC architectures in their respected area of resolution and speed.

Between these extremes, the so called 'successive approximation register' (SAR) ADC is found. This converter architecture is usually used to build moderate-speed ($100kHz$-$100MHz$) medium-resolution (8-16-bit) converters with extremely low power requirements. These specifications make

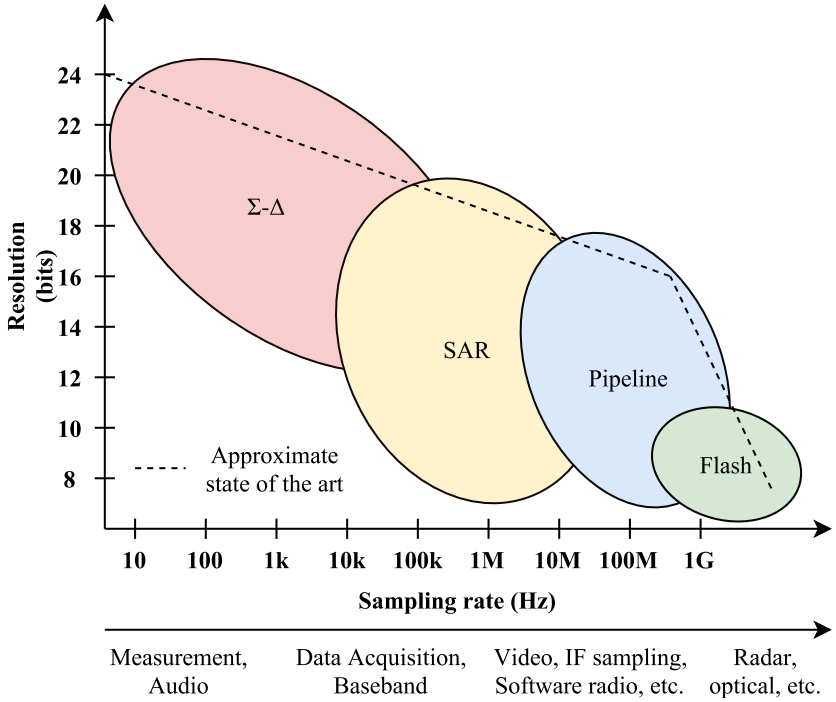

Figure 1.2: Comparison of ADC architectures with respect to accuracy and speed.

this architecture very attractive for sensor interfaces, data acquisition and radio base-band. Recent developments, such as [4] and [5], report Walden [3] figure of merits (FOMs) of (less than) a femto-joule per conversion-step. The Walden FOM is shown in Equation 1.1.

$$FOM_w = \frac{P}{f_s \cdot 2^{ENOB}} = \frac{P}{2 \cdot BW_{eff} \cdot 2^{ENOB}} \quad \text{[J/conv-step]} \tag{1.1}$$

where the ENOB (Effective number of bits) is defined by the signal-to-noise-and-distortion ratio:

$$ENOB = \frac{SNDR - 1.76}{6.02} \tag{1.2}$$

To realize these low figures, trends point towards hybrid architectures instead of scaling, because of the aforementioned increasing limitations of analog circuit design in small technologies.

In this work, a hybrid SAR ADC is described which uses a time-to-digital converter (TDC) to shorten the successive approximation algorithm. By measuring the decision speed of the comparator in the SAR ADC, steps can be skipped in the SAR algorithm, resulting in a lower power consumption and higher possible converter speed. To realize this, a comparator has been designed that can also act as a voltage-to-time converter, a time-to-digital converter has been designed to measure this time, and a possible logic implementation is provided.

1.1 Scope of Research

The scope of this thesis is limited to the following research questions:

1. Is it possible to leverage information from the logarithmic decision speed of a comparator to improve the SAR algorithm?
2. How can the decision speed of a comparator be measured with a low power CMOS circuit?
3. What are improvements to the SAR algorithm and what improves the FOM?
4. What is a possible implementation of this system in a SAR ADC?

1.2 Motivation

The proposed TDC-assisted SAR ADC has several aspects which make it an attractive improvement to existing work.

1. First and foremost, improving the SAR algorithm can result in power improvements because in recent SAR ADCs, most power is lost in switching, comparing and logic [6]. Decreasing the amount of conversion steps will decrease the power of the most demanding parts of the converter.
2. Secondly, most implementations of TDCs do scale well with technology [7]. If the performance of the converter becomes more dependent on the TDC, the SAR ADC could benefit from scaling.
3. Thirdly, TDC implementations are generally quite simplistic and will not result in a significant area and power penalty, especially considering the improvements in converter power consumption.
4. Lastly, Improving the maximum conversion speed of the converter due to the skipping of SAR steps also results in an improved FOM.

1.3 State-of-the-art and Uniqueness of Idea

Recent research in low power SAR design focuses on hybrid converter solutions to push the FOM envelope. Hsieh et al. propose several hybrid converters with the aim of reducing power such as a smart sampling scheme [8], an implementation using an early-late detector (ELD) [9], and prediction of the MSB [4]. Harpe et al. propose improving FOMs by means of improving ENOB using different mechanisms in [10] and [5], for example by oversampling. At the ICD group, a SAR design by Elzakker et al. [6] showed improvement of power consumption in the DAC, comparator and logic. In this work, the DAC and comparator are not the main contributors to conversion power anymore. The implemented comparator is based upon the Double-Tail Sense Amplifier by Schinkel et al. [11], which exhibits a logarithmic time-to-conversion (clock-to-Q delay) characteristic with regards to input differential voltages.

The idea of leveraging this time domain data as a means for improving the SAR switching scheme seems to be quite rare in literature. [12] propose using a TDC to measure the slowest behaviour of the comparator metastability to leverage an extra bit of information at the LSB-end to increase the converter's effective number of bits (ENOB). [13] implements a 3-stage TDC-assisted SAR that improves the SAR algorithm by skipping steps in many cases with redundancy, very similar to this work. These papers report a promising improvement in power and ENOB, but seem to suffer from bad INL/DNL, and have FOMs that cannot compete with today's standards. In this work, different time quantization algorithms and TDC implementations will be discussed that might further push the envelope. Besides this fact, the implementation is done in a 65nm technology, and several state-of-the-art building blocks (DAC, Comparator, etc.) will be assumed. Thus, this can be considered unique research.

1.4 Thesis Outline

The opening of this thesis will consist of several theoretical explanations to form a well-founded understanding of the basic SAR ADC and its algorithm. Using the combination of the aforementioned sub-circuits and additional insight of the voltage-to-time characteristic of a comparator, techniques to improve the SAR algorithm will be discussed.

Using the knowledge gained from theory, the next chapter deals with the actual circuit design. A comparator and TDC have been designed and motivation is given for its transistor-level implementation and layout.

The design chapter is followed by a section about simulation and its results. Each sub-block is simulated and the combination of a comparator and TDC is also tested.

Finally, this work is concluded and several recommendations for further research are given.

Chapter 2

Theory of a traditional SAR ADC

In this chapter, the operation and theory of a traditional SAR ADC is explained. Firstly, the algorithm which dictates the technique of quantization is described. Secondly, the sub-circuits typically used in SAR ADCs are briefly analyzed. These include the track-and-hold, DAC, comparator and logic circuits. Armed with this knowledge, some conclusions are made that give way to a behavior of the SAR ADC that can be exploited to make the system more efficient.

2.1 The SAR algorithm

SAR ADCs are based on the principle of a binary search algorithm. This algorithm, also known as the half-interval search or logarithmic search [14], approximates a target value by several binary comparisons. Essentially, the target value is compared to the mid-point of an array. If the target value is not equal to this midpoint, the half of the array that certainly does not include the target value is excluded, and a comparison is again done using the mid-point of the newly resulting half-sized array. This method of elimination is repeated until the target value is equal to the mid-point, or the maximum resolution of the system is reached.

Translating this algorithm to electronic systems, the target value could be a certain V_{in}, and the mid-point, or reference value V_{ref}. Because an ADC is specified at a fixed resolution, the binary search algorithm is bound by this value. An N-bit value can be found in N steps. To clarify the algorithm and its implementation in electronics, a 3-bit example is shown in Figure 2.1. In this example, the closest digital code for V_{in} is obtained by using a 3-step binary search. In this example, the full-scale of the analog input is $V_{fs} = 2V$ with a mid-point of $0V$. As described, at the first step with $V_{ref} = 0$, the comparison yields $V_{in} > V_{ref}$, producing a binary 1 for the most significant bit (MSB). Next, the bottom of the tree is discarded because we know V_{in} is situated between $0V$ and $1V$, and thus the reference voltage shifts to $V_{ref} = 0.5V$. The second comparison now yields a binary 0, since $V_{in} < V_{ref}$. In the next cycle, the algorithm is repeated, this time yielding a 1. The full-resolution quantization is now achieved, providing a binary code 101.

Using this example, one can asses that several components are necessary to implement this system electronically. Firstly, a comparator is needed to provide a means of comparing two values and quantizing that result to a logic value. Secondly, a reference voltage has to be created, requiring a digital-to-analog converter (DAC). Thirdly, the input voltage has to be sampled and held stationary during the conversion phase, which is done using a track-and-hold or sample-and-hold. Lastly, all components have to be controlled using some form of logic. Figure 2.2 shows a general example of a SAR ADC, in this case implemented differentially. One important difference is that instead of having a reference voltage that is compared to an input voltage, this mechanic is built-in by subtracting the V_{ref} from V_{in} before comparison. By this means, several simplifications are done at once. Firstly, the comparator output now automatically defines the sign, or direction of the algorithm together with the binary value. Secondly, the DAC, sample capacitor and summation block can be implemented

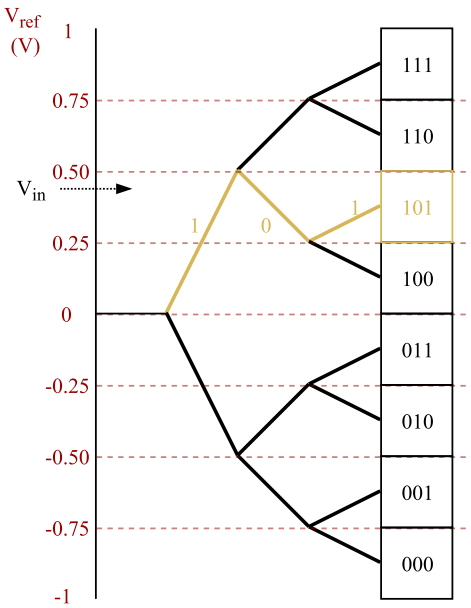

Figure 2.1: Example of a 3-bit binary search to quantize V_{in}.

using a single capacitive array, described later in this chapter. Thirdly, the common-mode voltage on the comparator input does not change, which is essential for this design and is discussed at a later point.

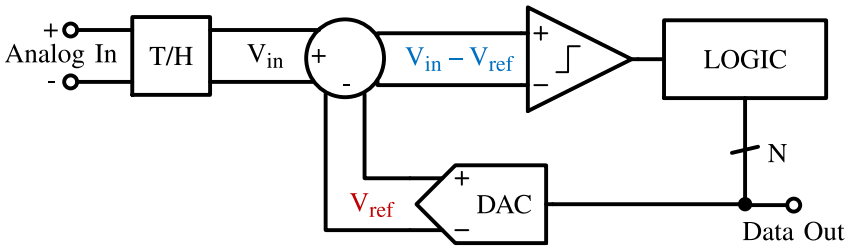

Figure 2.2: A generalized block diagram of an N-bit differential SAR ADC.

Before proceeding to go into detail about the discussed subsystems, some thought has to be given with regards to the clocking of the system. As explained, a SAR ADC requires N-steps to quantize a signal with N-bit resolution. Given a certain sample frequency f_s, the DAC, comparator and logic have to be clocked at as least $N \cdot f_s$, since all steps have to be completed within this sample period. In designing a SAR ADC, the speed requirements for the sub-systems have to be obtained concerning this sub-clock.

2.2 Track and Hold

The track-and-hold (T/H) is an essential building block for any ADC. It provides the means to sample an input signal onto a storage element (e.g. a capacitor), realizing a time-continuous to time-discrete conversion. After sampling, the ADC can quantize the stored value. When the ADC is done, the sample is released, and the input signal is tracked until the next sample moment. Figure 2.3 shows the essential diagram of a T/H circuit consisting of a sampling switch and a sampling capacitor C_{hold}.

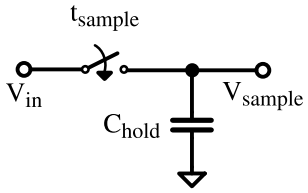

Figure 2.3: A generalized T/H circuit using an ideal sampling switch and a sampling capacitor C_{hold}.

In the track-phase, the switch is closed. The input signal is tracked on the top plate of the capacitor. In the hold-phase, the switch is opened, storing the last known value of the input signal on C_{hold}. This behavior is shown in Figure 2.4.

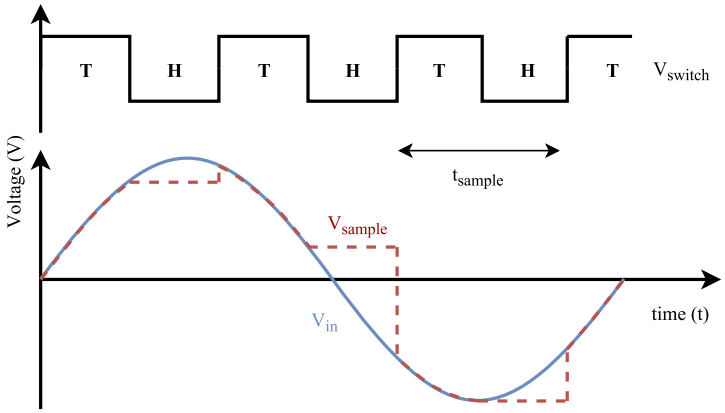

Figure 2.4: The signal behavior of an ideal T/H circuit.

Unfortunately, an ideal switch does not exist. The switch is usually implemented using an n-channel metal-oxide semiconductor (NMOS), and this gives way to several performance degradations. The three main performance limitations of the NMOS-switch are because of clock timing mismatch, parasitic capacitances and (non-linear) on-resistance. Given the fact that the input signal is sampled on a capacitor, these parasitics result in a bandwidth limited system, and the addition of noise and frequency-dependent distortion, degrading (for example) the achievable signal to noise ratio (SNR). A decreased SNR results in a worse effective number of bits (ENOB) and thus decreases the FOM in Equation 1.1.

On-resistance

The on-resistance of an NMOS device can be written as:

$$R_{on} = \frac{1}{\mu_n C_{ox} \frac{W}{L}(V_{gs} - V_{th})} \tag{2.1}$$

Several observations can be made from this equation. Firstly, the on-resistance is dependent on the overdrive voltage $V_{gs} - V_{th}$. As this can be rewritten to $V_{clk} - V_{in} - V_{th}$, it becomes clear R_{on} is dependent on the input voltage for a constant clock-voltage. A second observation is the fact that the on-resistance can be changed by varying the W and L of the NMOS.

The first observation gives way to a well-known and widely used solution: The bootstrapped switch proposed by Abo and Gray in [15]. The bootstrapped switch is a circuit that keeps $V_{gs} = V_{clk} - V_{in}$ constant, e.g. the input voltage is added to the switch clock voltage: $V_{clk} = V_{DD} + V_{in}$. This provides a constant on-resistance and an insurance that the NMOS is operating in a safe region for all input voltages.

Clock Feed-through

Because of the gate-drain and gate-source overlap capacitance C_{ov}, some charge of the clock signal is fed through to the output node. Using basic circuit analysis, it follows that the output error voltage ΔV_{sample} can be described as:

$$\Delta V_{sample} = V_{clk} \cdot \frac{WC_{ov}}{WC_{ov} + C_{hold}} \tag{2.2}$$

Given the fact that a bootstrap circuit is used, this error is dependent of the input level as $V_{clk} = V_{DD} + V_{in}$.

Charge Injection

The channel of the mosfet itself also has a capacitance. When the clock signal goes low, the remaining charge in the channel Q_{ch} has to flow out to either or both of the channel terminals. Assuming the worst case, where all charge flows to the output node, the output error voltage ΔV_{sample} can be formulated as:

$$\Delta V_{sample} = \frac{Q_{ch}}{C_{hold}} \tag{2.3}$$

This voltage is input dependent. If we assume $V_{in} \approx V_{sample}$ at the time of injection [16]:

$$Q_{ch} = WLC_{ox}(V_{clk} - V_{in} - V_{th}) \tag{2.4}$$

Because a bootstrap circuit is used , the input dependent behavior is lifted, as the overdrive voltage is kept constant. Therefore, in a differential circuit, this is a common-mode error, which should ideally not decrease performance but rather result in an offset in the code.

kT/C Noise

The last non-ideal behavior consists of the noise generated by the on-resistance previously described. Figure 2.5 shows the non-ideal model that explains this fact. The RMS noise voltage stored on the sampling capacitor can be described as:

$$\overline{v_{n,kTC}} = \sqrt{\frac{kT}{C_{hold}}} \tag{2.5}$$

This noise voltage is non-dependent on R_{on}, however it does show that the sampling capacitor needs to be sufficiently large to ensure a good noise performance.

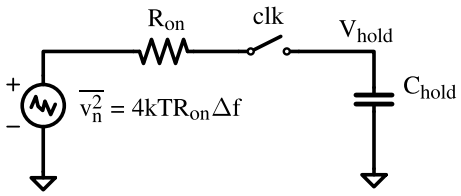

Figure 2.5: Thermal noise in an elementary track and hold circuit.

2.3 DAC

The DAC in a SAR is normally a charge redestribution network implemented using switchable capacitors. This gives the DAC a double function. Firstly, the total network can be used as a sampling capacitor. Secondly, by switching parts of this network 'in' or 'out', the charge stored on the capacitors can be reconfigured to perform the mathematical operation shown in Figure 2.2. There are many switching schemes with different merits. All schemes provide a means to do a binary search algorithm, but by implementing smart switching schemes [17], step-wise charging [6], etc., power consumption can be decreased or other beneficial effects can be achieved such as a constant common mode voltage, needed in this work.

Figure 2.6 shows a possible basic implementation of a charge redistribution DAC proposed by McCreary and Gray in [18]. The bottom plates of a binary weighted capacitor array sample the input in the track phase. During the hold phase, based on the binary search algorithm, capacitors are switched to redistribute the charge and change the output voltage V_{dac}. If the switched capacitor is chosen C_n and the total sample capacitance is C_{total}, the resulting output voltage is:

$$V_{dac} = V_{in} \pm V_{ref} \frac{C_n}{C_{total}} \tag{2.6}$$

The algorithm works as follows: After sampling, the top plates are disconnected from their reference voltage. The MSB capacitor is connected to the positive reference voltage and all other capacitors are connected to the negative reference voltages. Based on the comparison made, the algorithm propagates towards the LSB switching capacitors to different reference voltages.

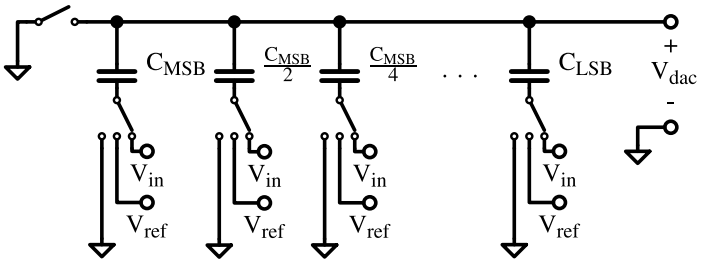

Figure 2.6: Example of a traditional binary-weighted capacitive DAC.

Figure 2.7 shows a more recent implementation, based on [17]. In this so called split-capacitor DAC, several improvements are observed. Firstly, the input signal is directly sampled on the top plate of the capacitor, which is also connected to the comparator. The bottom terminals need two reference voltages e.g. ground and VDD which can be provided by a simple inverter. By splitting the MSB capacitor into another binary weighted array of MSB-1 to LSB, and connecting half of the array to VDD and half of it to ground during reset, the DAC can create positive and negative shifts in sample voltage, or do nothing at all in certain SAR steps. This can be beneficial for more complex SAR algorithms.

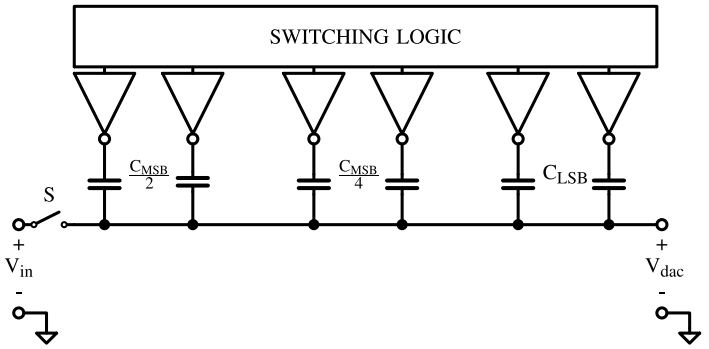

Figure 2.7: Example of a split-cap DAC as in [17].

DAC Accuracy

The DAC has several non-idealities which need to be taken into account to obtain the required accuracy for the converter. The most critical non-idealities are thermal noise and capacitor mismatch. The thermal noise sampled on the capacitor array has been formulated in Equation (2.5). As the resolution of the ADC should be limited by its LSB value and not by non-idealities, the total capacitance of the DAC should be scaled in such a way that the error is smaller enough.

Often, this is done by finding the C_{total} for which the thermal noise voltage is smaller than the quantization error. If a quantization error voltage of $\Delta/\sqrt{12}$ is taken, where Δ is the quantization step of an N-bit ADC defined as $V_{range}/2^N$, the resulting sampling capacitance is obtained:

$$C_{total} \geqslant \frac{12 * kT}{V_{range}^2} \cdot 2^{2N} \tag{2.7}$$

The mismatch between unit capacitors because of process variation, layout irregularity and parasitics results in non-linearities. Based on the Pelgrom model for mismatch [19], if a normal distribution is assumed, the standard variation of the unit capacitance can be written as:

$$\frac{\sigma_C}{C_u} = \frac{A_C}{\sqrt{WL}} \tag{2.8}$$

Where σ_C is the variation of the unit capactiance, C_u the unit capacitance, and A_C a technology-specific parameter. The critical transition point in the matching of the DAC occurs at MSB/2. [20] shows the maximum standard deviation of the DNL and INL for both traditional and split-capacitor DACs based on the Pelgrom model:

$$\sigma_{DNL,conventional} = 2^{\frac{N}{2}} \cdot \frac{\sigma_C}{C_u} \tag{2.9}$$

$$\sigma_{INL,conventional} = 2^{\frac{N}{2}-1} \cdot \frac{\sigma_C}{C_u} \tag{2.10}$$

$$\sigma_{DNL,splitcap} = 2^{\frac{N-1}{2}} \cdot \frac{\sigma_C}{C_u} \tag{2.11}$$

$$\sigma_{INL,splitcap} = 2^{\frac{N-2}{2}} \cdot \frac{\sigma_C}{C_u} \tag{2.12}$$

Where N is the DAC resolution and the INL and DNL standard deviations have to be divided by a factor $\sqrt{2}$ in case of a fully-differential implementation. The decrease in DAC linearity because of capacitor mismatch in turn decreases the linearity of the converter.

2.4 Comparator

In a SAR ADC, a comparator is used to compare the sampled input voltage to the DAC-generated reference voltage. In case of the fully differential architecture, it resolves the sign of the difference voltage $V_{in} - V_{ref}$. Common comparator topologies such as the StrongARM comparator [21] and Double-tail comparator [11] share a similar topology: a pre-amplifier followed by a latch. By clocking the comparator, low power can be achieved. Since the comparator decides on the sign of each bit, the accuracy of the comparator largely dictates the accuracy of the full converter. Therefore, it is of the essence that a comparator is designed to have low noise and offset, whilst keeping a low power and high speed performance. An example of a comparator design by Schinkel et al. [11] is given in Figure 2.8 and will be the initial topology for this work.

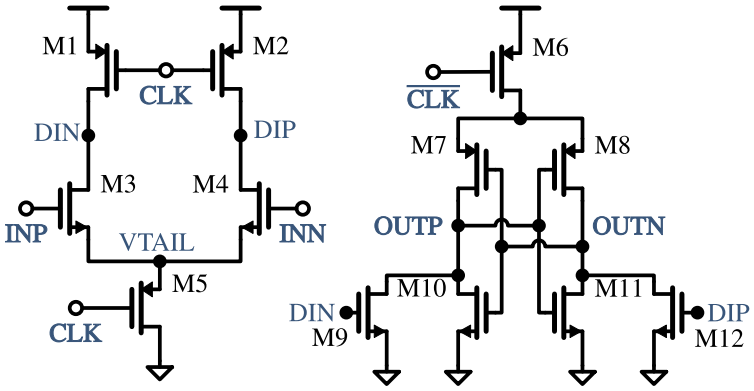

Figure 2.8: Schematic of a Double-Tail Dynamic Comparator from [11].

Operation of a Double-Tail Dynamic Comparator

The operation of the circuit in Figure 2.8 is as follows. During the reset phase, the tail switches $M5$ and $M6$ are open, and nodes DIN and DIP are pre-charged to VDD. Because these nodes are charged to VDD and tail switch M6 of the latch is opened, the output nodes $OUTP$ and $OUTN$ are discharged to ground through $M9$ and $M12$ resulting in no necessary reset transistors in the latch.

In the active phase, when CLK is high and $\overline{CLK}$ is low, the pre-amplifier is enabled and a differential voltage is generated between DIP and DIN whilst the common mode of these nodes drop towards ground. $M9$ and $M12$ pass this differential value to the latch. The latch is enabled by $M6$ at the same time as the preamp and remains in a meta-stable state until the common mode voltage of DIN and DIP drop below a certain value. This occurs when $M9$ and $M12$ biased by this common mode voltage decreases to such a level that the transistors cannot overcome the regenerative behavior of the cross coupled inverters. When this happens, a differential voltage is created on output nodes $OUTN$ and $OUTP$ and is increased until a rail-to-rail output is obtained. This behavior is shown in Figure 2.9.

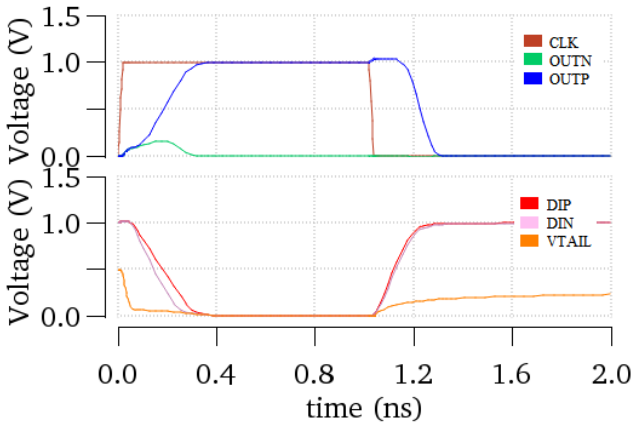

Figure 2.9: Large-signal behavior of a Double-Tail Dynamic Comparator.

Pre-amplifier

The pre-amplifier stage of this comparator is effectively a differential pair with a tail current source. The amplification of the input signal is realized by (dis)charging a capacitor on the drain node of the input transistor. To understand the behavior of the output nodes, a single half of this circuit can be taken. Figure 2.10 shows a large-signal and small-signal model of a single common source stage.

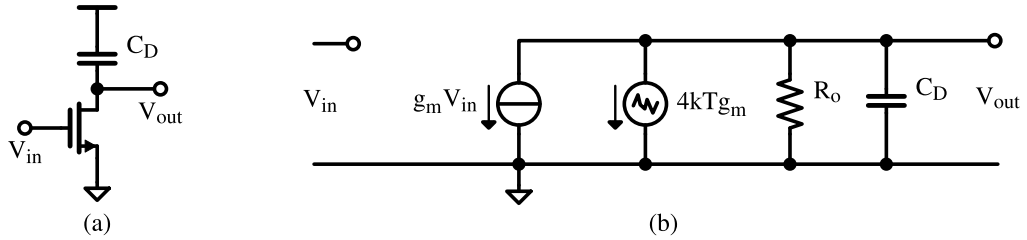

Figure 2.10: Large-signal (a) and small-signal (b) circuit of a single common source stage with capacitive load.

The voltage gain of this amplifier is time-dependent for $t << R_0C$ as proven in the thesis of [6]. The static input differential voltage is integrated as follows:

$$v_{out} \approx -v_{in}\frac{g_m}{C}t \quad [V] \tag{2.13}$$

And thus the voltage gain equals:

$$A_v = -\frac{g_m}{C}t \tag{2.14}$$

The actual implementation of the pre-amplifier is a differential pair, meaning that the g_m can be derived from the tail transistor current:

$$g_m = \sqrt{\mu_n C_{ox}\frac{W}{L}I_{M5}} \quad [S] \tag{2.15}$$

The rate of discharge of the drain capacitors on which the aforementioned output differential voltage is built up over time, is also dictated by the capacitance and tail current proportional to a factor:

$$\frac{I_{M5}}{C_{DI}} \quad \left[\frac{V}{s}\right] \tag{2.16}$$

Regenerative Latch

When the DIP and DIN nodes reach low enough and M9 and M12 are disabled, the latch stage in 2.8 can be modeled as a cross-coupled pair of inverters with an initial voltage on its output nodes. The analysis of this circuit has been described in [22]. A small signal model of this latch is shown in Figure 2.11.

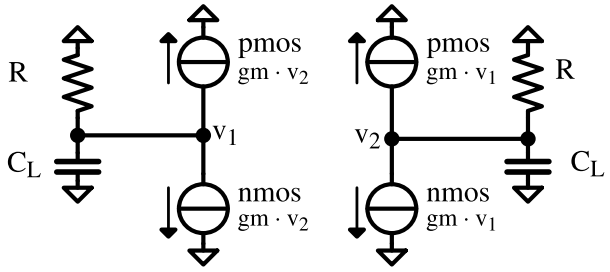

Figure 2.11: Small-signal equivalent model of cross-coupled inverters.

Based on the model shown in Figure 2.11, the output differential voltage can be described by a homogeneous differential equation:

$$V_{diff,out}(t) = V(0) \cdot e^{t/\tau} \quad [V] \tag{2.17}$$

where

$$\tau \approx \frac{C_L}{g_{m,p} + g_{m,n} - 1/R} \quad [s] \tag{2.18}$$

This equation can be re-written to obtain the time t_{latch} required for an acceptable voltage difference, where the circuit has essentially successfully latched. Including this minimum voltage $V_{out,min} = V_{DD}/2$ for which the latch has successfully decided its output state, the result becomes:

$$t_{latch} = \tau \cdot ln\left(\frac{V_{out,min}}{V(0)}\right) = \tau \cdot ln\left(\frac{V_{DD}/2}{V(0)}\right)[s] \tag{2.19}$$

In [23] this equation is linked to the physical implementation of the comparator ommiting output resistance. The initial voltage given to the latch can be written as:

$$V(0) = \left(\frac{2V_{th,n}}{I_{tail,latch}}\right)^2 \cdot \frac{V_{th,n}C_L}{C_{DI}} \cdot g_{m,n} \cdot g_{m,p} \cdot \Delta V_{in,comp} \quad [V] \tag{2.20}$$

Adding this to the initial time $t_0 = 2V_{th,n}C_L/I_{tail,latch}$ of the latch activating, the final result is obtained:

$$\begin{aligned} t_{comp} &= t_0 + t_{latch} \\ &= \frac{2V_{th,n}C_L}{I_{tail,latch}} + \frac{C_L}{g_{m,p} + g_{m,n}} \cdot ln\left(\frac{V_{DD} \cdot I_{tail,latch}^2 \cdot C_{DI}}{8V_{th,n}^2 \cdot C_L \cdot g_{m,M9,M12} \cdot g_{m,M3,M4} \cdot \Delta V_{in,comp}}\right) \quad [s] \end{aligned} \tag{2.21}$$

Which can be rewritten to accommodate the input voltage in the numerator:

$$\begin{aligned} t_{comp} &= t_0 + t_{latch} \\ &= \frac{2V_{th,n}C_L}{I_{tail,latch}} - \frac{C_L}{g_{m,p} + g_{m,n}} \cdot ln\left(\frac{8V_{th,n}^2 \cdot C_L \cdot g_{m,M9,M12} \cdot g_{m,M3,M4} \cdot \Delta V_{in,comp}}{V_{DD} \cdot I_{tail,latch}^2 \cdot C_{DI}}\right) \quad [s] \end{aligned} \tag{2.22}$$

Inspecting this result, the comparator time exhibits a logarithmic clock-to-output (clock-to-Q) behavior depending on the initial input differential voltage and given a specific common mode voltage.

Input Referred Noise

The input terminals of the comparator are connected to the sampling capacitor. Input referred noise of the comparator compromises the sampled voltage, resulting in possible wrong decisions around LSB. Therefore, the input referred rms noise of the comparator together with the rms noise of the sampling circuit, should be lower than half an LSB to achieve the required performance. The input-referred noise during the comparison cycle of the comparator is mainly due to the noise of the input pair. If these transistors are close to or in weak inversion, the input referred noise can be calulated as proven in the thesis of Van Elzakker [6]. By dividing the noise due to the input pair at the DI nodes by the gain A_v (Eq. 2.14), and noting that the bandwidth of the DI nodes can be related to the rate of discharge:

$$BW \approx \frac{I_{tail}}{2 \cdot C_{DI} \cdot V_{DI}} \tag{2.23}$$

the noise can be written as:

$$\overline{v_{n,in,rms}} \approx \sqrt{\frac{kT}{C_{DI}}} \cdot \sqrt{\frac{16V_{Th}}{V_{DI}}} \quad [V_{RMS}] \tag{2.24}$$

Where C_{DI} is the capacitance on a DI node and $V_{th,n}$ the threshold voltage of an NMOS, and V_{DI} the differential output voltage on the DI nodes. This result is remarkable because it is not dependent on the bias current. It is dependent on the drain capacitance, which can be increased artificially to decrease noise as a trade of with power consumption.

Mismatch of the Input Pair

The input differential pair will have mismatch due to process variations. This mismatch will result in a offset voltage. For an easy estimation of the offset voltage due to mismatch, the Pelgrom model [19] for threshold mismatch can be used:

$$\sigma^2(V_{th}) = 2\frac{A^2_{V_{th}}}{WL} \quad [V^2] \tag{2.25}$$

Where $A_{V_{th}}$ is a process-specific parameter. The area of the input transistors should thus be chosen large enough for the specified maximum offset. It must be noted however, that input referred offset of the comparator results in a 'DC' output code shift and can be considered not to affect the linearity of the ADC.

2.5 Logic

As the name of this type of converter suggests, the logic of a SAR ADC evolves around a register. A N-bit register both stores the quantized output code and the capacitor switch values. This leads to a very compact control system, systematically shown in Figure 2.12. It consists of N states, where each state has a measure (comparator) and execute (register and switch) action. A state machine such as this can be implemented in several ways. Traditionally, a state-select is implemented as a straight ring (or Overbeck) counter using D-flip-flops, and the registers are also D-flip-flops or D-latches. This is shown in Figure 2.13.

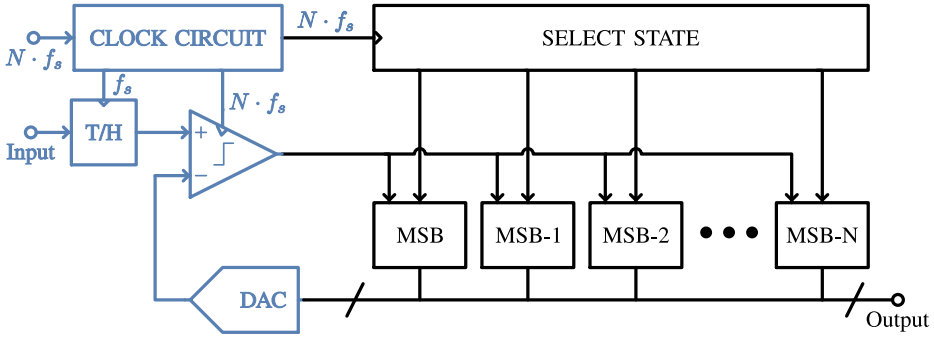

Figure 2.12: System level overview of a traditional SAR logic block.

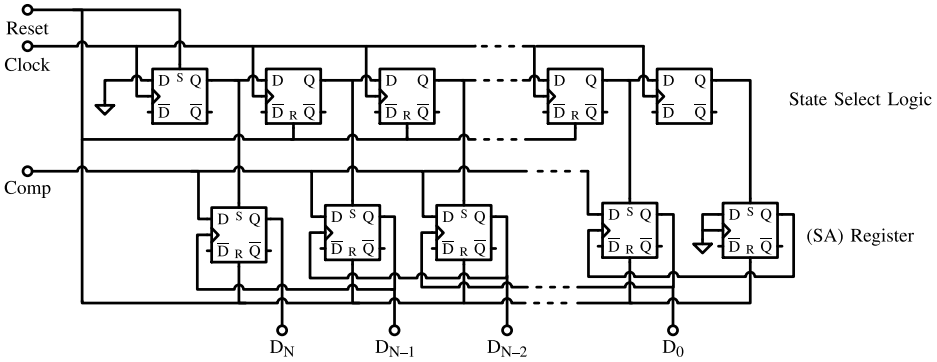

Figure 2.13: Implementation example of a traditional SAR logic block.

A more recent approach is to implement a single sequential logic circuit which uses conditional logic to progress instead of an external pointer. The systematic view of this implementation is shown in Figure 2.14. This implementation does not require an external sub-clock at $N \cdot f_s$ for the logic, and can be very low power. This has been proven in works such as [6]. A more concrete example of such a circuit is shown in Figure 2.15.

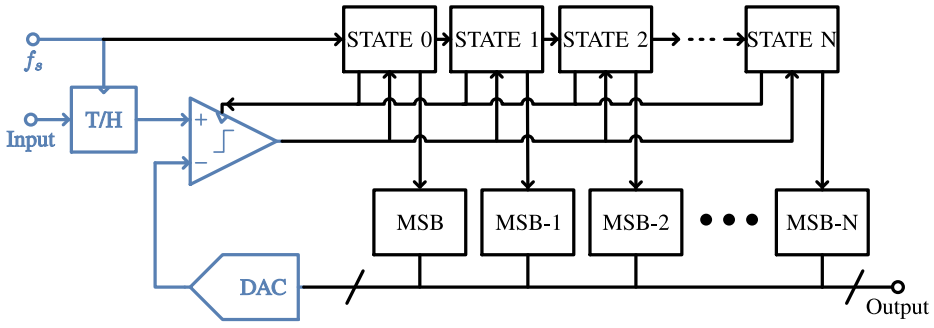

Figure 2.14: System level overview of a conditionally sequenced SAR logic block.

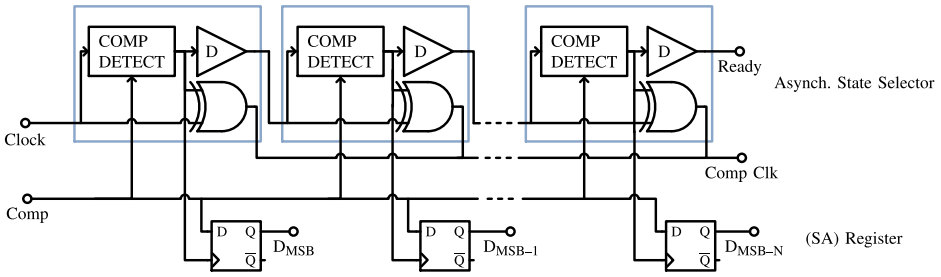

Figure 2.15: Implementation example of a conditionally sequenced SAR logic block.

2.6 Accuracy limitations of the SAR ADC

The accuracy of a SAR ADC in terms of ENOB is limited by the SNDR as proposed in Equation 1.2. Three main contributions of noise limit need to be taken in to account. Firstly, the quantization noise, which is inherent to any ADC. The RMS quantization noise voltage can be written as:

$$\overline{v_{n,q,rms}} = \frac{V_{LSB}}{\sqrt{12}} \tag{2.26}$$

As explained in this chapter, the non-ideal sample circuit also introduces a form of noise, given that $C_{hold} = C_{DAC}$:

$$\overline{v_{n,kTC,rms}} = \sqrt{\frac{kT}{C_{DAC}}} \tag{2.27}$$

Additionally, the input-referred noise of the comparator is also seen on the sample nodes:

$$\overline{v_{n,in,rms}} \approx \sqrt{\frac{kT}{C_{DI}}} \cdot \sqrt{\frac{16V_{Th}}{V_{DI}}} \tag{2.28}$$

The RMS value for a full-swing sinusoidal input signal is:

$$\overline{v_{signal,rms}} = \frac{\frac{2^N}{2}}{\sqrt{2}} \cdot V_{LSB} \tag{2.29}$$

And thus the SNDR including these effects can be written as:

$$SNDR = 20log\left(\frac{\overline{v_{signal,rms}}}{\sqrt{\overline{v_{n,q,rms}}^2 + \overline{v_{n,kTC,rms}}^2 + \overline{v_{n,in,rms}}^2}}\right) \tag{2.30}$$

This calculation does not take into account the mismatch of DAC elements, but from the calculation it becomes clear that the SNDR and therefore the performance of the ADC can be tuned by increasing the sample capacitor (in turn also decreasing the capacitor mismatch as explained), increasing the DI-node capacitors, or increasing the maximum signal swing. From this it follows that the accuracy of the SAR ADC can be traded-off with power consumption (and layout size).

2.7 Conclusions

In this chapter, the workings of a traditional SAR ADC have been described. The SAR ADC uses a binary search algorithm to find a quantized output value closest to the sampled value. This is implemented as follows: The input voltage is sampled using a Track and Hold circuit. This input voltage is compared to a reference voltage created by a DAC. The comparison is done using a comparator. Based on the output of the comparator, bits are set and the reference voltage is changed according to the algorithm. The sampled voltage and reference voltage converge from MSB to LSB in N steps. A traditional and more recent implementation of a SAR logic block are described. The equations in the chapter show that the accuracy of the ADC is limited by the quantization noise, the noise of the comparator, the noise in the sample and hold circuit and possibly the mismatch of the DAC capacitors.

Chapter 3

Theory of the TDC-assisted SAR ADC

Traditional SAR ADCs as described in the previous chapter are already very efficient and most recent works therefore focus on hybrid architectures. Hybrid architectures usually exploit energy saving schemes, oversampling or a secondary conversion mechanism. In this work, the focus lies on a non-ideal behavior in the comparator that can be exploited to gain more than one bit of information per conversion step. In this chapter, this non-ideal behavior is explained, a technique to measure this behavior is given, and three algorithms for using this information are proposed. Based on performance calculations, one of these algorithms is chosen for further implementation.

3.1 The Comparator and DAC as a Voltage-to-Time Converter

As discussed in Section 2.4, when a comparator is modeled as a pre-amp with a latch, the comparator time (or clock-to-Q delay) can be ideally described as a logarithmic function of the input differential voltage:

$$t_{comp}(\Delta V_{in}) = t_0 - \frac{C_L}{g_m} \cdot ln\left(\frac{\Delta V_{in,comp}}{\beta_{latch}}\right) \quad [s] \tag{3.1}$$

Where t_0 and β_{latch} are circuit-specific constants explained in Chapter 2. This means that a comparator with this behavior essentially acts as a logarithmic voltage-to-time converter (VTC). Exponentially increased voltages given to the input of the comparator result in a linearly increasing delay. Incidentally, the binary weights tested by a SAR ADC are exponential in nature. Since each bit of the code is tested separately, the comparison can be equated as follows:

$$Output_N = \begin{cases} 0, & \text{if } \Delta V_{in} - \Delta V_{DAC} < V_{DD} \cdot 2^{-N} \\ 1, & \text{if } \Delta V_{in} - \Delta V_{DAC} > V_{DD} \cdot 2^{-N} \end{cases} \tag{3.2}$$

As seen in this equation, for each bit that is examined in the SAR algorithm, there exists a boundary condition in the form of an exponential voltage $\Delta V_{in,comp} = V_{DD} \cdot 2^{-N}$. Plugging this exponential behavior into Equation 3.1, linearly-spaced delay times are expected:

$$t_{comp} = t_0 + \frac{C_L \cdot ln(\beta_{latch}/V_{DD})}{g_m} - \frac{C_L \cdot ln(2)}{g_m} \cdot N \quad [s] \tag{3.3}$$

This principle is elaborated in Figure 3.1.

The binary DAC in a SAR ADC is the exponential voltage generator that creates these boundaries. By itself, given that one of the binary scaled capacitors is switched, it can generate the following voltages on the input of the comparator:

$$V_{DAC} = V_{DD} \cdot 2^{-N}[V] \tag{3.4}$$

Where N is the selected bit $[MSB : MSB - N]$ to be switched in the capacitive DAC.

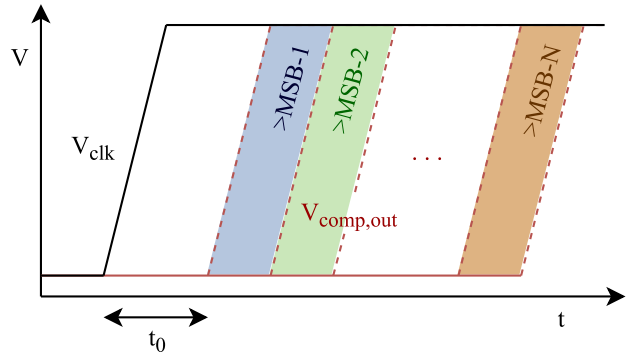

Figure 3.1: Binary-weighted input voltages ideally result in linearly spaced clock-to-Q delays.

As can be seen from this result, binary-weighted input signals to the comparator are resolved in a linearly increasing time, where an MSB-sized differential voltage is fast and (MSB-N)-sized input differential voltages take N times longer to resolve.

If this information is measured before or during the conversion in a SAR ADC, an estimate can be made of the voltage present at the comparator input. This estimate can be used to improve the traditional SAR algorithm. The system-level schematic for TDC-assisted SAR ADC described here is shown in Figure 3.2.

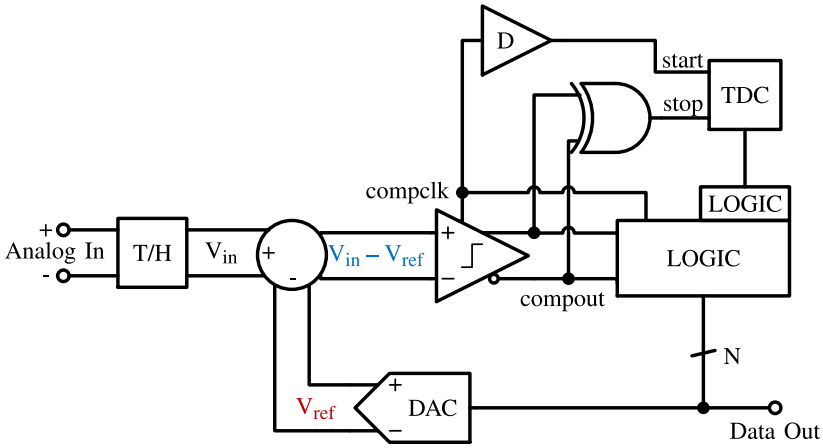

Figure 3.2: System level schematic of a TDC-assisted SAR ADC.

Before discussing several algorithm improvements, thought has to be given to the implementation of a time-measurement circuit in CMOS.

3.2 Time-to-Digital Converter

To measure the clock-to-Q delay, a time-to-digital converter (TDC) has to be implemented. In [24], many TDC implementations have been described. Almost all TDCs operate on a similar principle: A 'start' signal initiates the creation of a series of delayed edges (for example using a delay line made of buffers). A 'stop' signal initiates a state where the position of the last delay is quantized (for example by sampling the delay line with early-late-detectors (ELDs)). This basic circuit principle is elaborated in Figure 3.3.

The resulting output code is a thermometer code, where a 10 or 01 transition marks the position of the last delayed edge.

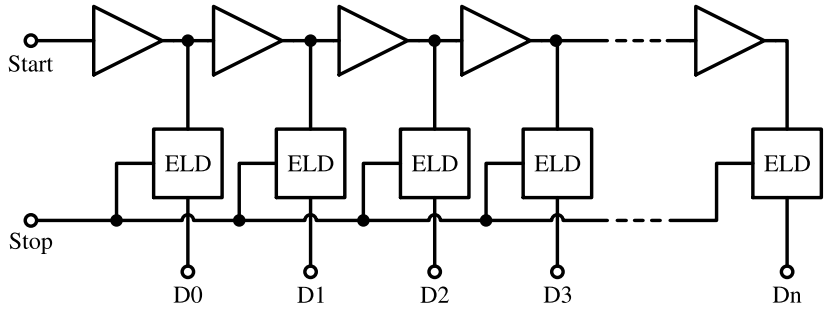

Figure 3.3: Basic TDC implemented as a delay line sampled by ELDs.

TDC time resolution

The dynamic range of a TDC can be described as:

$$DR = N \cdot T_{LSB}[s] \tag{3.5}$$

Where, in case of the delay-line approach, N is the number of TDC stages and $T_{LSB} = t_{delay-cell} = 2t_{gate}$. If the number of TDC stages is matched to the number of registers in the SAR ADC, and the dynamic range of the TDC is matched to the clock-to-Q delay range of the comparator, the boundaries shown in Figure 3.1 can be quantized by the TDC. This means the voltage region attributed to a SAR step can be obtained through the TDC.

TDC mismatch

Mismatch due to process variations in the delay line result in a deviation in time per TDC step. Additionally, the delay mismatches are accumulated and become increasingly worse in a non-calibrated delay line. [24] extensively analyzes these effects. Figure 3.4 shows the arrival-time uncertainty for an example of a linear delay line with different types of calibration.

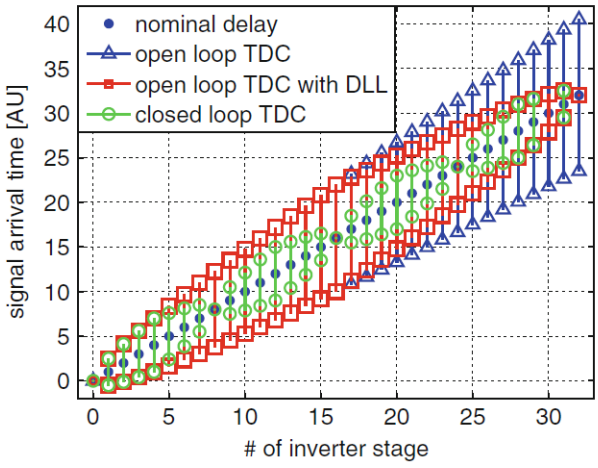

Figure 3.4: Arrival-time uncertainties in a linear TDC given several types of calibration. [24].

When designing a TDC that has to measure clock-to-Q delays of a comparator, the comparator itself exhibits a timing uncertainty due to mismatch and PVT corners. The timing uncertainties of the comparator delay and the TDC together should be kept smaller than the time between the boundaries of the binary weighted values. This has to be accomplished using one or several of the following techniques:

1. The TDC has to be calibrated to match the timing of the comparator.
2. The mismatch of the comparator and delay line can be decreased by increasing transistor sizes at the cost of power.
3. The speed of the comparator can be decreased to allow more margin of error.

3.3 New SAR Algorithms

By measuring the comparison time, a coarse estimate of the signal present at the input of the comparator can be done. With this information, obvious steps in the SAR algorithm can be skipped. An easy example would be that if the comparator makes a very slow decision directly after a sample is taken, the sample must have a value near the LSB. The MSBs of the output code can be assumed, and MSB-sized DAC changes are therefore not necessary and can be skipped. An example is shown in Figure 3.5. In this figure, the voltages at the comparator input nodes of a traditional and TDC-assisted SAR ADC are shown. The initial sampled voltage is not very large. To resolve the output code, the traditional SAR algorithm works from MSB to LSB. However, in this example the largest bits do not contain the sampled signal. The algorithm however, will switch all of these bits (and corresponding DAC capacitors) as seen in (a). Skipping these MSB-sized parts of the algorithm would result in less energy consumed in almost all ADC components: Less comparisons are needed to find the LSB value, less DAC switching is needed, and less logic has to be activated. In the example of the TDC-assisted SAR ADC, the range of the initial value is estimated and unnecessary DAC steps are skipped, shown in (b).

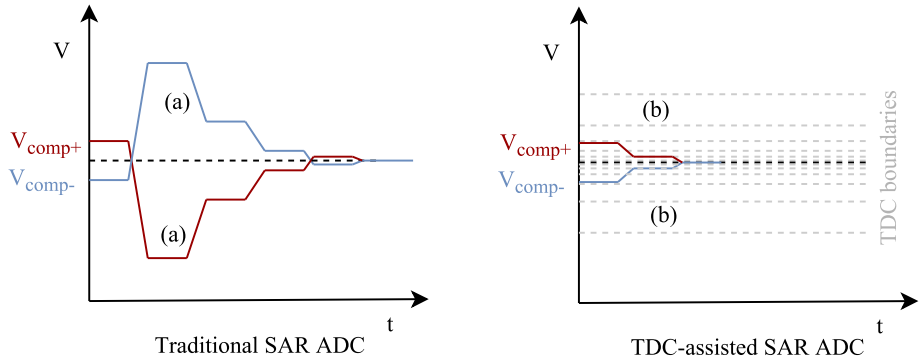

Figure 3.5: Example of the voltages at the comparator input of a traditional SAR ADC and TDC-assisted SAR ADC, starting with a sampled value and the resulting DAC steps based on the SAR algorithm.

During this research, three TDC-assisted SAR algorithms have been proposed:

1. **Initial Range Estimation**
   After sampling, the comparator is clocked as usual to find the MSB (which is the sign of the signal). During this conversion-step, the TDC is used to find the range in which the input signal is present. Obvious unnecessary steps are skipped, and the remaining algorithm is equal to that of a traditional SAR. In Appendix A.2, the expected improvement in conversion steps and power consumption are elaborated.

2. **LSB Detection**
   A traditional SAR algorithm is the basis for this variant. However, during each comparison, a TDC is used to check if a LSB-sized differential voltage is present on the input of the comparator. If this is the case, the algorithm can be stopped because the output code is already correct. Depending on the resolution of the TDC, several bits can be quantized with the TDC.

In Appendix A.3, the expected improvement in conversion steps and power consumption are shown.

3. **Path-finding Algorithm**
   The path-finding algorithm also uses a TDC in each conversion step. However, the TDC is set up to quantize the linear time steps from MSB to LSB, proposed in the previous sections. During each comparison, the range of the comparator input voltage is estimated. Depending on this range, instead of simply going to the next bit-position, the position is found for which the bit-value is not already certain. Depending on the resolution of the TDC, this can result in an average decrease in conversion-steps.

A single algorithm has to be chosen for implementation. Algorithm (1) is very easy to implement and can still work if the estimation is not accurate (as long as the TDC is set up conservatively). However, it has very marginal improvements in power consumption. Algorithm (2) is also relatively easy to implement, but do not improve the power consumption as drastically as (3). Therefore, the path-finding algorithm is chosen and elaborated in the next section.

The Pathfinding Algorithm

Figure 3.6a shows a decision tree of a traditional SAR ADC. As can be seen in this figure, the algorithm has only 2 routes per comparison, 0 or 1. Now, imagine that a TDC is added to the SAR ADC that can detect the $[MSB : MSB - 1]$ and $[MSB - 1 : MSB - 2]$ region based on the speed of the preceding comparison. Figure 3.6b shows the possible routes the new algorithm can take. There is still a direction (0 or 1), but there is also extra knowledge about the closest 'next step'.

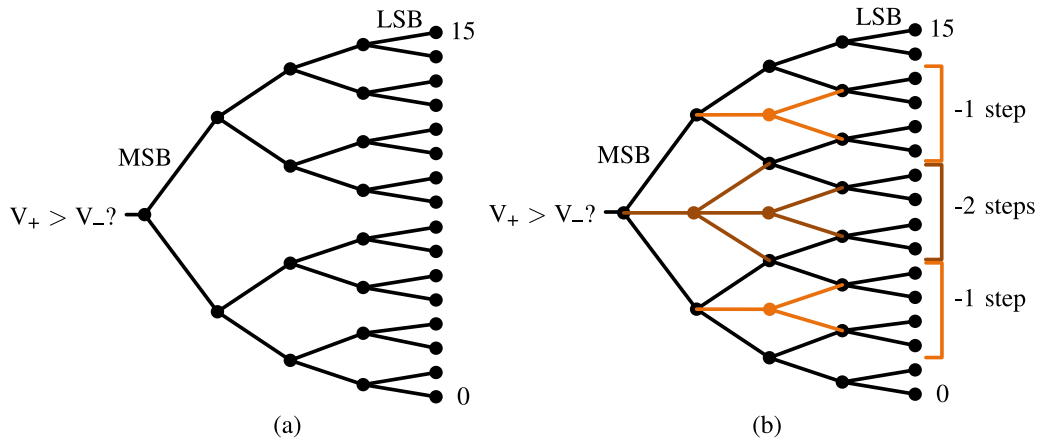

Figure 3.6: Decision tree of a 4-bit (a) traditional SAR ADC and (b) 2-step path-finding TDC-assisted SAR ADC.

As shown in Figure 3.6, the path-finding algorithm results in less necessary steps to reach the LSB for most codes. Figure 3.7 shows the amount of steps needed for each code of a 10-bit ADC, given a certain TDC resolution. Note that the largest and smallest codes still require the original amount of conversion-steps, no matter the resolution of the TDC. The average number of steps drop systematically for each additional stage of TDC resolution. These averages are given in Table 3.1.

Table 3.1: Average number of conversion steps for a given number of TDC stages in a 10-bit ADC.

TDC resolution (stages)	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
Avg. conversion steps	10.00	9.50	9.00	8.38	7.62	6.78	5.88	4.93	3.96	2.98

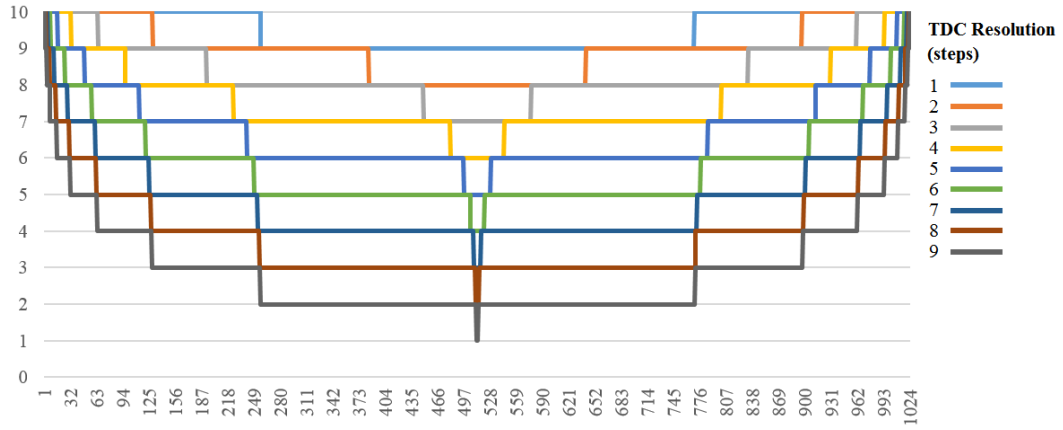

Figure 3.7: Number of conversion steps for a 10-bit SAR ADC given a TDC resolution.

In [13], a similar system leveraging comparator time is shown. It also provides with plots showing the number of conversion steps per output code. The implementation has an average of about 8 conversion steps with the added advantage that no 10-step code is present at the borders. The path-finding algorithm shows similar average steps already around a TDC resolution of 3-4 stages, but the outermost codes still require a 10-step conversion.

The decrease in average number of steps results in an improvement of the Walden FOM as long as the TDC (and its additional logic) does not consume more power than is saved by skipping certain conversion-steps. The improvement is therefore heavily dependent on the implementation of the rest of the converter. For the rest of this thesis, the ADC described in [6] will be used as as a benchmark. The energy consumption calculations are elaborated in Appendix A.1. The resulting normalized expected FOMs are shown in Table 3.2. They are based on the following equation for the total conversion energy:

$$E_{PathfindingADC,total} = \\ E_{TH} + M_{steps,avg}(E_{DAC,avg} + E_{comp} + E_{logic,stage} + O_{tdcstages}(E_{TDC,stage} + E_{logic,extra})) \tag{3.6}$$

Where E_{TH} is the energy consumed by the track and hold, $M_{steps,avg}$ is the average number of conversion steps, $E_{DAC,avg}$ is the average energy consumption for a DAC switch, E_{comp} the energy consumption of a single comparison and reset, $E_{logic,stage}$ is the energy consumption of a single logic stage, $O_{tdcstages}$ is the number of TDC stages, $E_{TDC,stage}$ is the energy consumption of a single TDC stage including its logic, and $E_{logic,extra}$ an estimate of additional 6 minimum sized logic gates for the algorithm change.

From this result, it can be seen that adding the additional logic and TDC initially adds some power but breaks even at a path-finding resolution of 4 stages, before providing an improvement in the FOM purely based on decreased power consumption.

Table 3.2: Normalized FOMs for a 10-bit TDC-assisted SAR ADC based on [6] given a number of TDC stages.

TDC resolution (stages)	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
FOM (normalized)	1.00	1.04	1.05	1.04	1.00	0.94	0.86	0.76	0.65	0.50

3.4 Additional Logic

The logic circuit of a traditional SAR ADC is made to always operate from the MSB state to the LSB state as described in the previous chapter. While a TDC-assisted SAR ADC also works from MSB to LSB, states have to be skipped based on the information received by the TDC. This is less obvious than it might seem.

TDC Logic

Firstly, the thermometer code from the TDC is converted to a one-hot code by the means of the circuit shown in Figure 3.8.

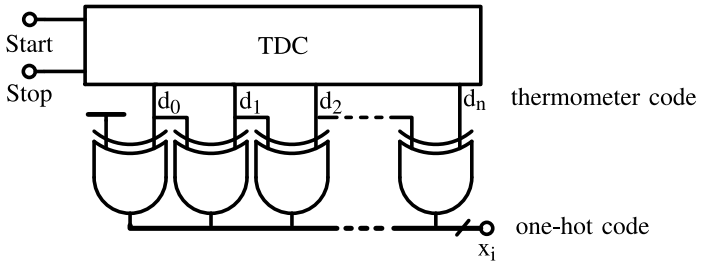

Figure 3.8: A thermometer to one-hot code converter.

The resulting one-hot code x_i is a pointer signal of N-stages which represents the voltage range of the previously compared signal and therefore can be used to select the next DAC/SAR step.

SAR Logic

Figure 3.9 shows a system representation of a possible logic block for the TDC-assisted SAR ADC. The logic implementation is built around the conditionally sequenced logic seen in Figure 2.14. The system works as follows: When a sample is taken, the start signal goes high. During state 0, a comparison is made of the sampled signal to find the MSB value (which is the sign of the signal). In terms of figure 3.7, one half of the graph is left. Together with this first comparison, the TDC is used to estimate which consecutive binary-weighted DAC value is closest to the sample. When the MSB output register is filled, the DAC is switched as 'requested' by the TDC, and the next state is enabled. Instead of this state standardly being the MSB-1 state, now it is the MSB-x state, where x is the TDC-found binary region. Based on the outcome of the new comparison, all skipped bits can be set with the same value. This algorithm is repeated until an LSB comparison is done or a less-than-LSB timing is detected. When the system records this signal from the TDC, represented as 'end-of-conversion', the 'ready' signal is forced and the consecutive states are disabled. This way, in the best case, the MSB and LSB are detected during the first conversion step and the converter is done. In the worst case, the complete traditional SAR algorithm is run with the increased power consumption of the TDC and additional logic. An example of a traditional SAR algorithm, a pathfinding algorithm with full 9-stage TDC accuracy and a pathfinding algorithm with a 6-stage TDC accuracy is given in Appendix B. The 6-stage example shows that it is of the essence that if the TDC refers to a state that has already been filled or passed, it should be ignored and the algorithm should move one step as in a traditional SAR algorithm.

3.5 Converter Accuracy

For the specific logic implementation proposed in the previous section to work, at all 9 steps, the TDC needs to accurately decide on which consecutive DAC switch will make the comparator input voltage converge to common mode as efficiently as possible. If a wrong decision is made by the TDC

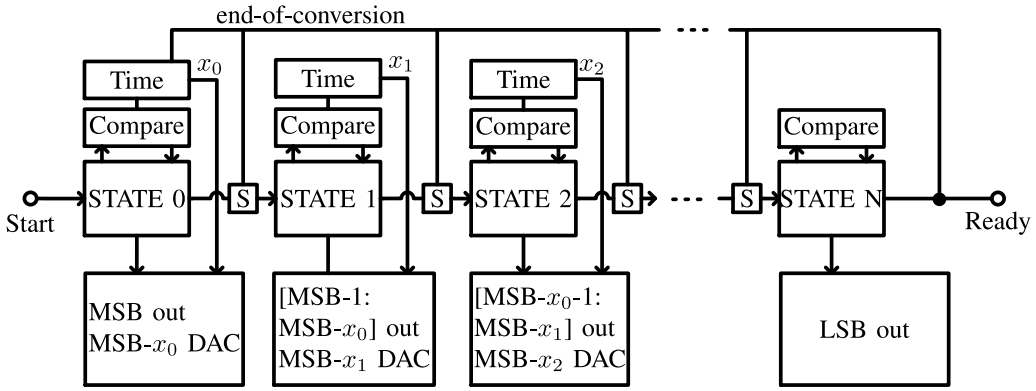

Figure 3.9: A system-level view of the TDC-assisted SAR ADC logic.

(e.g. it skips a DAC step too much, meaning the comparator voltage will never converge to within an LSB at common mode), the voltage can even possibly diverge and stall the algorithm because it will refer back to a previous state. This problem can be solved by making sure the traditional logic path of the SAR has priority over the TDC pointer signal.

The question then arises where the decision boundaries should ideally lie such that the aforementioned algorithm is obtained at maximum 9-stage resolution. Referring to Figure 3.7, the boundaries should in this case be located at (less than) 1LSB past (meaning at a smaller input voltage or 'longer' delay) the binary-spaced transition steps to avoid false output codes. This way, the system will rather skip too little than too much. The critical input voltages then are equal to the voltages that the DAC can create. Given a full dynamic range input of -1V to 1V (so 0 to 1V for the MSB-1:MSB-9) in a 10 bit converter, these boundaries coincide to the comparator input voltages shown in Table 3.3. If a 1-LSB accuracy cannot be obtained around these boundaries, the TDC should be set up in such a way that it will not skip too many steps (e.g. a lower effective resolution is required).

Table 3.3: Necessary input voltage boundaries corresponding to TDC outputs.

TDC output	**Ideal Boundary voltage (mV)**
0 (MSB-1 $<V_{comp}<$ MSB)	
1 (MSB-2 $<V_{comp}<$ MSB-1)	500
2 (MSB-3 $<V_{comp}<$ MSB-2)	250
3 (MSB-4 $<V_{comp}<$ MSB-3)	125
4 (MSB-5 $<V_{comp}<$ MSB-4)	62.5
5 (MSB-6 $<V_{comp}<$ MSB-5)	31.25
6 (MSB-7 $<V_{comp}<$ MSB-6)	15.63
7 (MSB-8 $<V_{comp}<$ MSB-7)	7.81
8 (MSB-9 $<V_{comp}<$ MSB-8)	3.91
9 ($V_{comp}<$ MSB-9)	1.95

During later stages in this research, it has been found that there are possibilities to require even less accuracy in the TDC. This is elaborated in the recommendations.

3.6 Conclusions

In this chapter, the proposed TDC assisted SAR ADC has been described. By viewing the comparator as a logarithmic voltage to time converter, and noting that the binary weighted DAC in a SAR provides exponentially scaled voltages during a conversion cycle, the clock-to-Q delay expected for binary scaled input values increases linearly. Measuring the comparator time can yield an estimate of the voltage at the input of comparator. If the measurement is quantized linearly in time, this can be linked to the binary values quantized in the SAR algorithm. This knowledge gives the opportunity to devise several new successive approximation algorithms, of which three are proposed. The third algorithm proposed concerns measuring the comparator-time with each comparison, giving the converter the possibility of skipping unnecessary DAC steps each time. This algorithm - coined the 'Path-finding Algorithm' - is chosen for further implementation. A possible implementation for the SAR logic that can provide this new algorithm is given, but because of time constraints not fully researched.

Chapter 4

Design and Implementation

To test the feasability of using the comparator as a voltage-to-time converter and measuring this time with a TDC to shorten the SAR algorithm, a comparator and a TDC are designed and combined. For limiting the specifications in such a way that it can be later implemented into a TDC-assisted SAR ADC, it is based around the traditional SAR ADC proposed in [6]. This means the following specifications are used:

- 10-bit differential SAR ADC
- 1V supply voltage
- 500mV common mode
- The DAC is a split-capacitor variant [17] which provides a constant common mode voltage on the comparator inputs, resulting in a logarithmic comparator delay based on differential voltages. Also, it can provide positive, negative or no shifts in DAC voltage, necessary for the proposed algorithm.
- A full-scale input voltage of 2V ($\pm$ 1V)
- RMS quantization noise is 0.56mV.
- ENOB is 8.75.
- A 65nm CMOS technology is used.

4.1 Comparator

The comparator needs the following characteristics:

1. A logarithmic clock-to-Q delay at a given V_{cm}.
2. Low power ($<$100fJ per comparison) and noise ($<$0.5LSB).
3. PVT characteristics similar to the TDC for matching over corners.

The comparator has to operate on a 1V supply voltage, with a common mode of 500mV. Figure 4.1 shows the implemented variant of the Double-Tail Sense Amplifier in [11]. Table 4.1 show the transistor sizes.

The pre-amplifier input transistors are scaled similarly to [11] to meet the required noise and offset specifications. The pre-amplifier tail is kept small to limit charge injection and noise. The capacitance at the DIN and DIP nodes are tuned for speed and noise. The latch does not have a single tail transistor as shown in Figure 2.8. Instead, the source nodes of M8 and M9 are not connected and two separate tail transistors are used. This essentially degenerates the PMOS transistors in the latch, decreasing the latch gain without the penalty of minimum size transistors or larger load

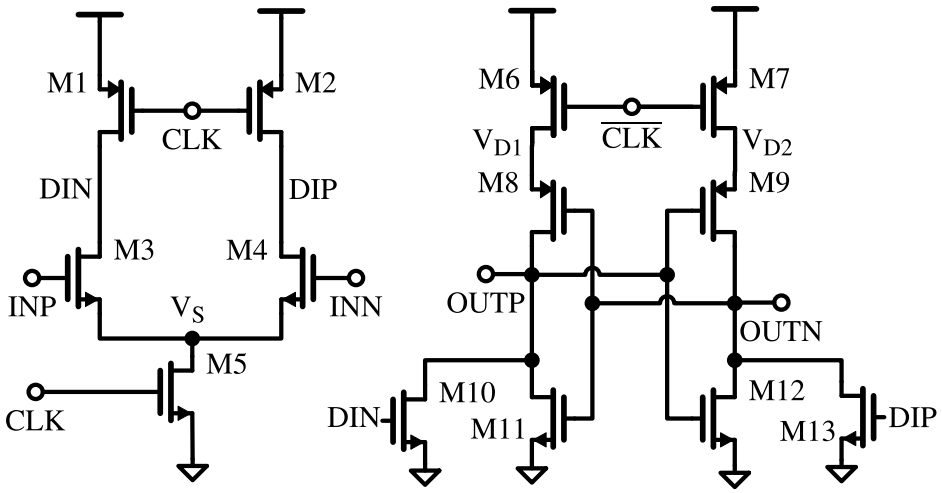

Figure 4.1: Implemented Double-Tail-type comparator with degenerated latch.

Table 4.1: Transistor sizes of the comparator in Figure 4.1

Transistor	**Size (W/L) [m]**
M1,M2	$120n/60n$
M3,M4	$2.8\mu/120n$ (x3)
M5	$120n/100n$
M6,M7	$250n/60n$
M8,M9	$400n/60n$
M10,M11,M12,M13	$200n/60n$

capacitance. Additionally, decreased gain results in a longer integration time and smaller noise bandwidth. Besides this change, the latch operates virtually equal to the previous art as explained in Chapter 2. A layout for the comparator is created to examine if the logarithmic time behavior holds including parasitics, and to examine realistic power consumption. The layout of the comparator is shown in Appendix C.1. The transistor sizes and DI-node capacitors are tweaked such that the comparator behaves as close to the theoretical logarithmic voltage-to-time converter as possible, given binary scaled input voltages.

4.2 TDC

The TDC needs the following general specifications:

1. Quantize linear time steps over the clock-to-Q-delay-range of the comparator (MSB-1:MSB-N).

2. Some form of calibration is needed for mismatch/PVT corners.

3. Low power (no significant penalty to system power).

The TDC is implemented as a delay line of calibratable buffers. However, instead of a separate delay line and a separate set of Early-Late detectors as seen in Figure 3.3, both functions are built into the same circuit, an implementation not found before in literature. By using a tri-state inverter and an inverter with a controllable speed to form a delay cell, the delay line can be 'frozen' at the moment a stop-signal is given to the TDC. All nodes are latched by a cross-coupled tri-state inverter at the same moment. This cell is shown in Figure 4.2 and the transistor implementation is shown in Figure 4.3. These delay cells are connected in series to form the TDC. M11-M15 are transistors that control the speed of the middle inverter during the falling-edge operation of the TDC. Each of these transistor size is tweaked such that this single active transistor gives the total delay line a less-than 1LSB accuracy in the final TDC stage in separate corners (TT, FF, SS, FS, SF), provided a control voltage of $V_{gs} = 0.66V$. That way, a simple single-time tuning can be performed by powering one of the transistors prior to using the converter, given a certain process corner. Given this reference voltage is accurate, the TDC time-to-digital transfer function is mapped to the logarithmic voltage-to-time transfer curve of the comparator. The reference voltage could for example be generated using a current mirror, or a different accurate reference. Another possible implementation could be a switchable resistive load.

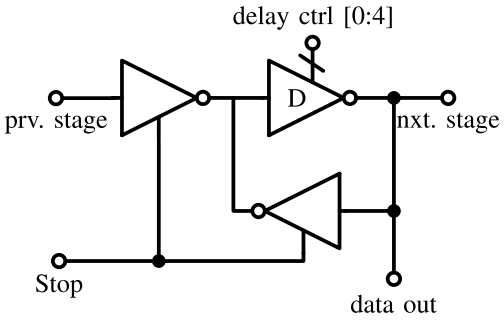

Figure 4.2: Single TDC stage with a controllable delay and built-in ELD.

The remaining transistors are chosen to meet the minimum requirement for speed (relative to the clock-to-Q delay of the comparator). The sizes are given in Table 4.2.

9 of these stages are connected in series to form a 9-stage TDC, capable of quantizing the $[MSB - 1 : MSB - 9]$ clock-to-Q delays in a 10-bit SAR ADC. The first stage boundary of the TDC is placed at a delay corresponding to 500mV input voltage, and the last stage boundary of the TDC is tuned up just past the 1.95mV LSB input voltage. This way, assuming the comparator generates linear time steps for exponentially decreasing input voltages, the middle TDC stages should correspond to the binary boundary voltages.

The thermometer code output of the TDC has to be converted to a one-hot code to act as a register pointer for the SAR. This is implemented with a decoder as proposed in Chapter 3. The layout of the TDC is shown in Appendix C.2 and is made to examine if the additional parasitics negatively affect power consuption and if the increased delay is proportional to that of the layed-out comparator. To minimize power consumption, a (minimum-sized) 6-transistor XOR is used from [25].

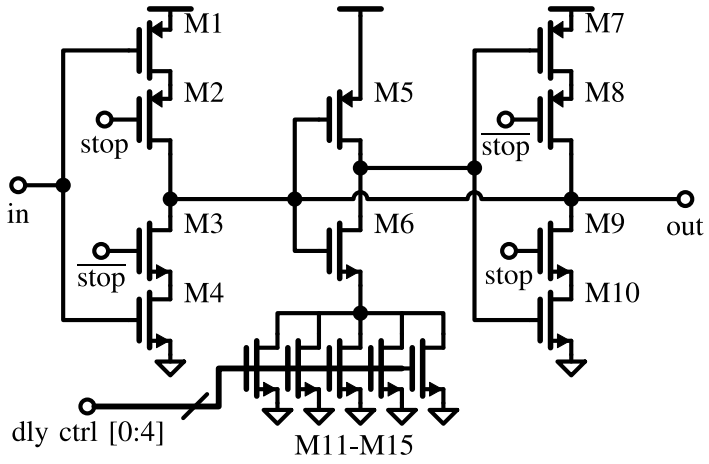

Figure 4.3: Transistor-level implementation of a single TDC stage.

Table 4.2: Transistor sizes of a TDC stage in Figure 4.3

Transistor	**Size**
M3,M4,M5,M7,M9,M10	$150n/60n$
M1,M6	$380n/60n$ (LVT)
M2,M8	$480n/60n$
M11 (for SS)	$120n/380n$ (LVT)
M12 (for FF)	$120n/240n$ (LVT)
M13 (for TT)	$240n/120n$ (LVT)
M14 (for FS)	$380n/120n$ (LVT)
M15 (for SF)	$380n/120n$ (LVT) (x2)

4.3 Combination of Comparator and TDC

The designed comparator and TDC are combined as shown in Figure 4.4. The outputs of the comparator are connected to an XOR gate, to provide a 'stop' signal for either a positive or negative comparison. A matched delay element is needed to align the clock pulse and the constant comparator delay t_0. Practically, this matched delay element should be designed during the layout phase to ensure the correct timing due to mismatch and corners. Another possibility is to use a copy of the comparator with a fixed MSB input voltage. This way, there is an equal propagation 'offset' delay for both the start and stop signal, so that the TDC only converts the timings from t_0 and onwards. Assuming that the TDC starts running at t_0, assuming that the comparator clock-to-Q behavior is linear for binary-sized input voltages, and the linear TDC is calibrated such that the decision boundary of the last stage is at LSB, the ideal decision boundaries shown in Table 3.3 are expected as the voltage-to-time and time-to-digital digital conversion gains are matched. The resulting ideal transfer function is shown in Figure 4.5.

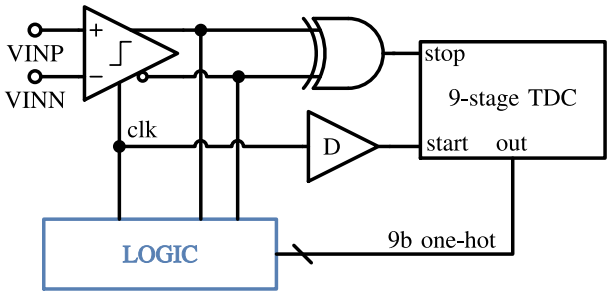

Figure 4.4: Schematic of the comparator and TDC together.

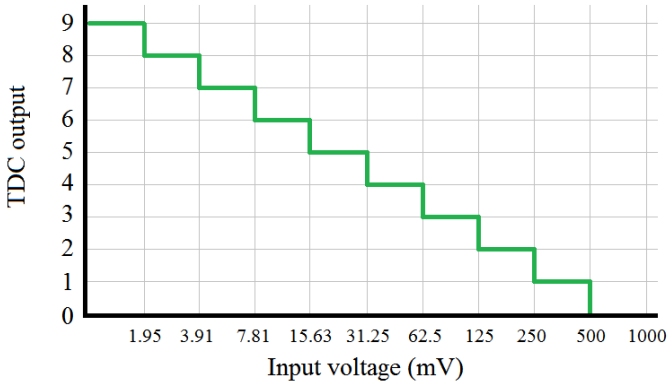

Figure 4.5: Ideal voltage-to-time-to-digital transfer function for the implemented comparator and 9-stage TDC.

Chapter 5

Simulations and Results

The testbench for simulations on the comparator and TDC is shown in Figure 5.1.

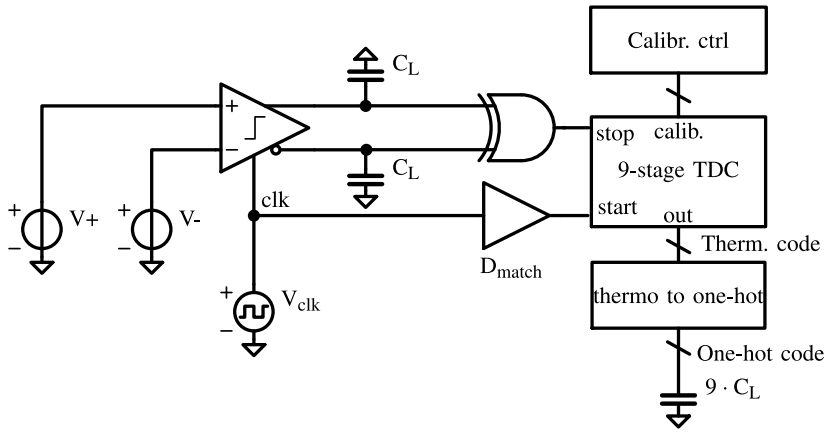

Figure 5.1: Testbench for simulations on the comparator and TDC.

5.1 Comparator

An example transient response of the comparator similar to Figure 2.9 is shown in Figure 5.2. As can be seen in this picture, it operates similar to the work in [11] described in the theory. Additionally, Figure 5.3 shows the clock-to-Q delay of the comparator for the binary-weighted input signals corresponding to the DAC voltages.

The comparator has been simulated on power consumption for several input voltages (Figure 5.4), relative clock-to-Q delay (Figure 5.5) and input-referred noise. All simulations are done using a 1V supply voltage and a 500mV common mode voltage.

The relative clock-to-Q delay has also been simulated over mismatch and process-corners using a monte-carlo simulation. This results in a range of possible transfer curves, of which the nominal, and 1-sigma fast and slow cases based on random sampling are also shown in figure 5.5.

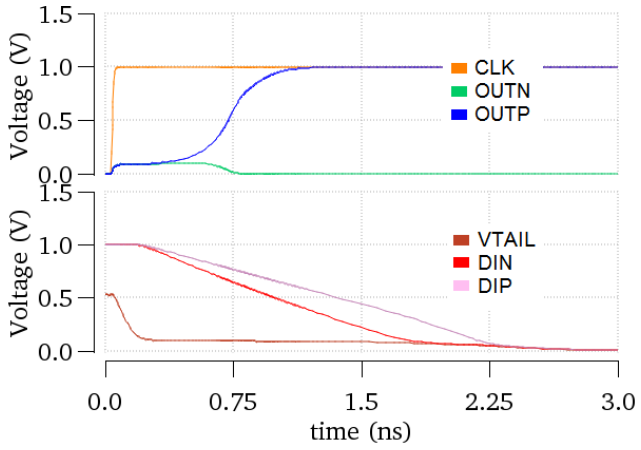

Figure 5.2: Example transient behavior of the designed comparator.

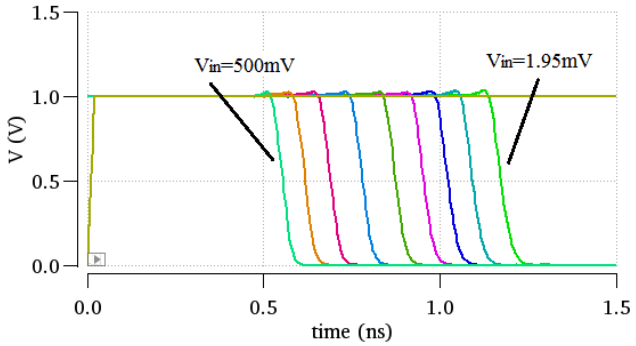

Figure 5.3: Comparator clock-to-Q delays for the binary weighed boundary voltages. Post-layout, the t_{comp} delays are slightly higher than expected, possibly because of additional parasitic capacitances, but still follow a logarithmic behavior.

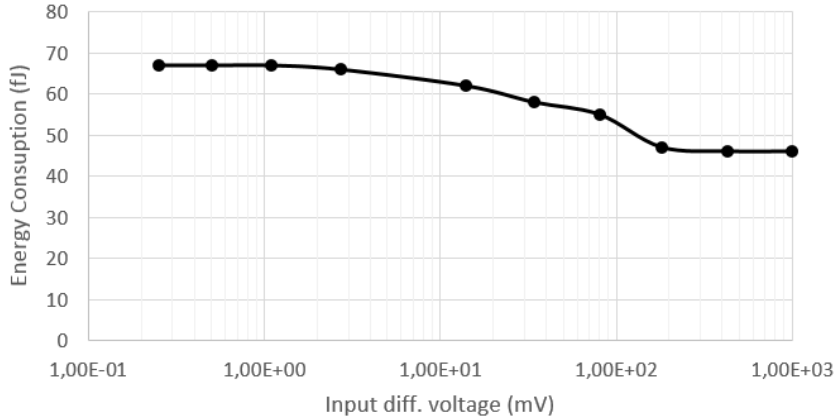

Figure 5.4: Comparator power for different input voltages.

The input referred noise has been simulated using both a transient noise and a PNOISE simulation in Cadence, because they often provide different noise figures because of the type of simulation. For PNOISE, a 0.25LSB signal is presented to the input of the comparator, and the differential-mode noise at the DI nodes is divided by the gain of the pre-amplifier during comparison. This results in a

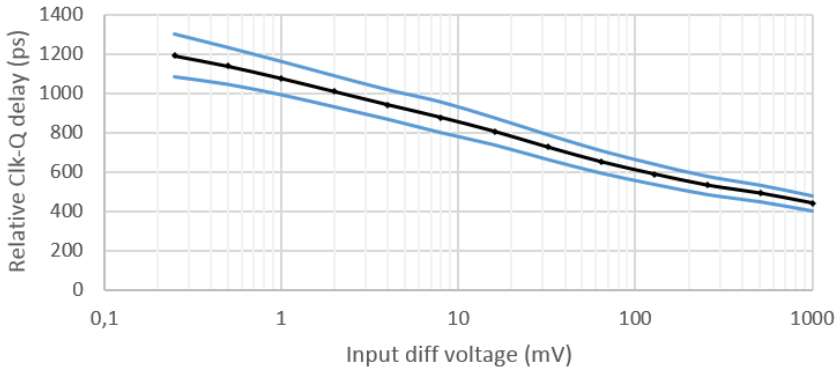

Figure 5.5: Relative clock-to-Q delay of the comparator. Nominal and 1-sigma fast and slow transfer functions are shown.

differential input referred noise voltage of $0.27mV_{RMS}$. For transient noise, a 0.25LSB input signal is presented, and the number of wrong decisions are counted over a large amount of periods. Based on the percentage of wrong decisions, a noise figure can be calculated. In this case, this results in $0.25mV_{RMS}$. The could possibly be attained to a different simulation technique or simulation accuracy. The worst case of these two results can be taken, meaning the comparator noise is about 48% lower than the $\Delta/\sqrt{12}$ quantization noise explained in theory.

The comparator has been tested and works in TT, FF, SS, FS and SF process corners. The worst-case clock-to-Q delays (both fast and slow) are noted for design of the calibration system in the TDC.

5.2 TDC

The implemented 9-stage TDC has been simulated on power consumption (Figure 5.6) and operation in the slowest and fastest mode is shown in Figure 5.7 and Figure 5.8). 1-sigma spread of the converter gain around the nominal resulting from a monte-carlo simulation are also included. The calibration extremes are based on corner simulations of the comparator clock-to-Q delay as mentioned before. The TDC is simulated and operates in TT, FF, SS, SF and FS corners and can match the comparator output in these corners by calibration using the specifically tuned transistor explained in the previous chapter.

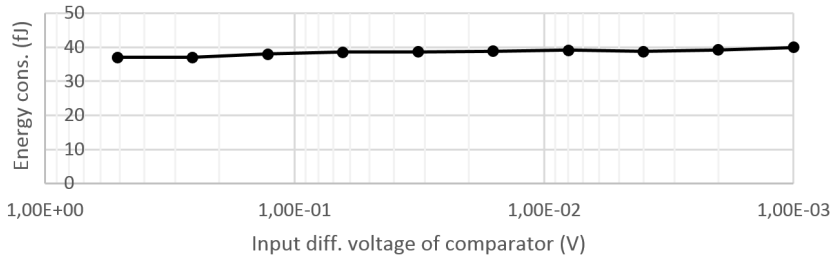

Figure 5.6: Power consumption of the 9-stage TDC including XORs.

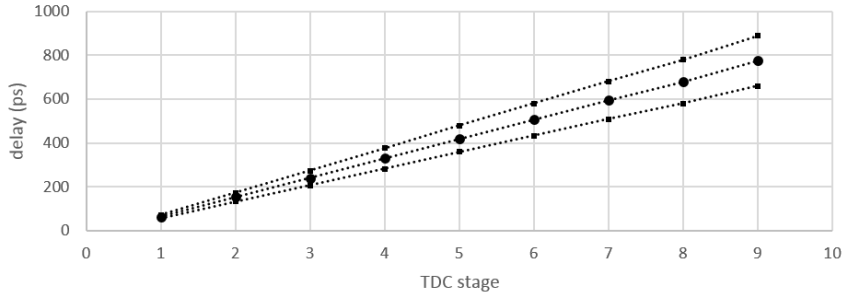

Figure 5.7: TDC delay transfer curves for nominal and 1-sigma mismatch on the fastest calibration setting.

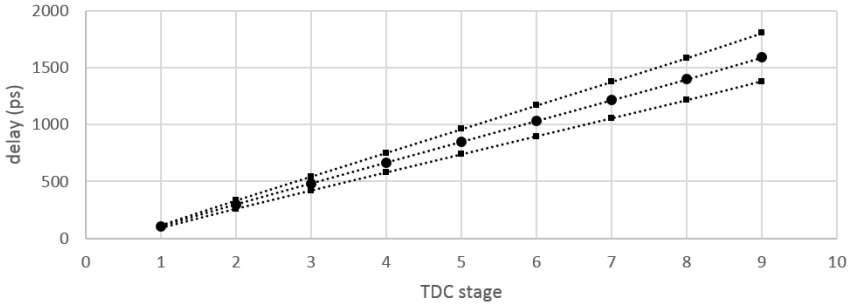

Figure 5.8: TDC delay transfer curves for nominal and 1-sigma mismatch on the slowest calibration setting.

Figure 5.9 shows an example of the output thermometer code of the TDC for a delay longer than 9-stages, so the start signal has rippled through all stages before a stop signal is received. This shows the TDC boundary timings as the logic output of each stage is the same node as the input for the next stage. Figure 5.10 shows a worst-case metastability condition of the one-hot output around a

sub-LSB input differential voltage. It also shows that there are bumps in the output code as a result of the specific XOR implementation. Care must be taken when designing the logic that these effects do not compromise the timing and consecutive logic values of the total circuit.

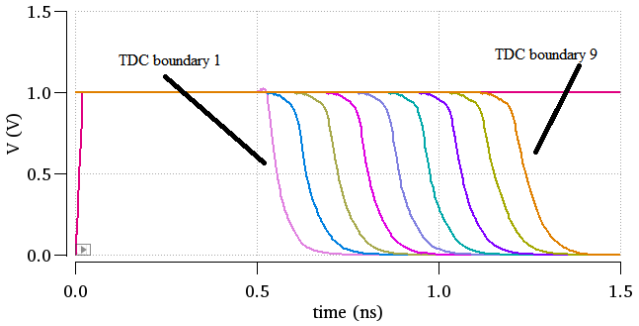

Figure 5.9: Transient simulation showing the TDC thermometer code output of a clock-to-Q delay that is longer than 9 stages, revealing the internal decision boundaries.

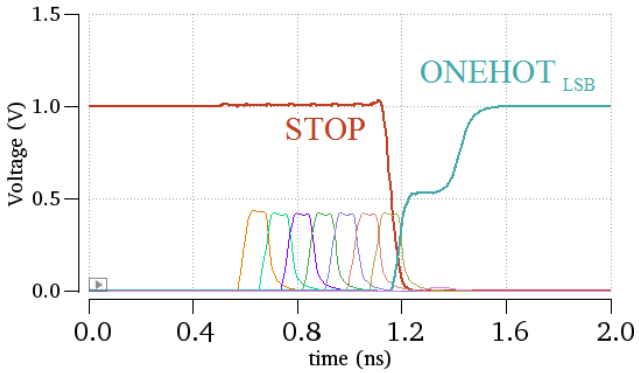

Figure 5.10: Transient simulation showing the worst-case metastability condition of the one-hot output of the TDC around the 9th stage decision boundary. The ripple of the TDC is fed through the XORs resulting in voltage bumps for non-active one-hot bits.

5.3 Comparator and TDC

The comparator and TDC are connected and the TDC is calibrated so the 1st stage coincides with the MSB-1 boundary, and the 9th stage output of the TDC coincides with an LSB differential voltage. The voltage-to-time-to-digital transfer curve is simulated by finding the actual boundary voltages for which the TDC outputs specific codes. This is shown in Table 5.1. The simulation is done in the TT corner with 1V power supply.

Table 5.1: Ideal and simulated boundary input voltages for TDC output codes.

Ideal boundary (mV)	**Simulated boundary (mV)**	**TDC code**	**Offset in LSBs**
>500	>498	0 0000 0000	
500	498	1 0000 0000	1.03
250	210	0 1000 0000	20.61
125	101	0 0100 0000	12.37
63	55	0 0010 0000	3.86
31	29	0 0001 0000	1.16
16	14	0 0000 1000	0.99
8	6	0 0000 0100	0.93
4	3	0 0000 0010	0.57
2	2	0 0000 0001	0.23

As seen in Table 5.1, for several values, the TDC output does not coincide exactly with the ideal input voltage levels. The ideal and simulated transfer functions are shown in Figure 5.11. The simulated transfer curve will be different for other corners or mismatch scenarios, but this has not been simulated.

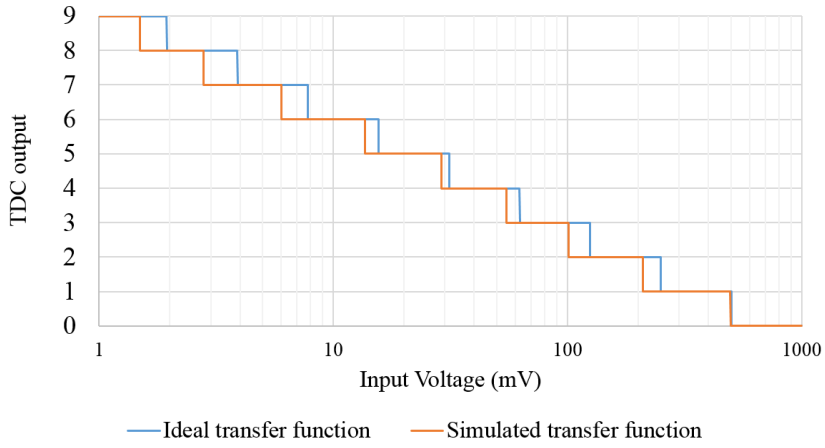

Figure 5.11: Transfer function between comparator input voltage and TDC output.

Based on this this result, several remarks can be made. Fistly, the first stage and last stage boundaries are placed close to the ideal boundary. Secondly, Several stages show a large deviation from the ideal boundaries. For the 9-stage path-finding algorithm to work at all codes with full efficiency, a 1-LSB accuracy is needed at all stages. This is not the case. In theory, wrong-decisions are generated when the TDC over-estimates a skipped DAC step. This would occur if a non-ideal boundary lies at a higher input voltage than the ideal transfer curve. In the tested circuit, this is also not the case, meaning that, although it cannot perform fully accurate at 9-stages, it will not provide skipping steps that are too large. To elaborate on this, a conversion example of the path-finding algorithm using an ideal and simulated transfer curve is given in Table 5.2 and Table 5.3 with a

sample voltage that lies between the worst-case non-linearity shown in Table 5.1, 220mV, showing the non-destructive effect of the different transfer curves on the algorithm.

Table 5.2: Path-finding algorithm for a 220mV input signal, based on the simulated transfer curve. X denotes the bits that can be set according to the TDC. At *, the TDC senses a voltage larger than the MSB-2 binary state it has reached, and therefore dismisses the TDC information and advances the algorithm a traditional step.

Step	V_{comp} **(mV)**	**Output Code**	**DAC shift (mV)**
1	220	1X ???? ????	-500
2	-280*	10 X??? ????	+250*
3	-30	10 0XX? ????	+62.5
4	32.5	10 011X ????	-31.25
5	1.25	10 0111 XXXX	-1.95
6	-0.7	10 0111 0000	

Table 5.3: Path-finding algorithm for a 220mV input signal, based on the ideal transfer curve. X denotes the bits that can be set according to the TDC.

Step	V_{comp} **(mV)**	**Output Code**	**DAC shift (mV)**
1	220	1X X??? ????	-250
2	-30	10 0XXX ????	+31.25
3	+1.25	10 0111 XXXX	-1.95
4	-0.7	10 0111 0000	

The question then arises what the effective obtainable path-finding resolution is. Unfortunately, this value is now code dependent, as the offset is different per stage, and not all stages are used in each conversion. The worst-case obtainable path-finding resolution has not been fully researched, and could be done in future work.
A possible worst-case scenario happens as shown in the previous example, where the starting voltage is one that is between the ideal and non-ideal transfer function, resulting in several extra conversion steps. The effective path-finding resolution of the aforementioned example, based on Figure 3.7, is 5. If the best case path-finding resolution then is taken at 9 (an assumption that can be made because several of the simulated stages are within 1 LSB of the ideal transfer function), all effective path-finding resolutions between 5 and 9 on average would result in an improved FOM. This is based on Table 3.2, given all its assumptions.

5.4 Conclusion

This chapter has provided simulation results for the designed comparator and TDC. Individually, the circuits work within specification. The comparator operates at 60fJ per comparison, has a differential input referred noise voltage $0.27mV_{RMS}$ which is lower than the quantization noise, and a logarithmic clock-to-Q delay. The implemented TDC consumes relatively low power (40fJ per comparison) and is step-wise tunable across the delay range of the comparator to perform in several corners. Combining the two circuits together, a voltage-to-time-to-digital transfer curve is obtained in a nominal corner. Examining the voltage-to-time-to-digital transfer function, 1-LSB accuracy across input voltages is not guaranteed. It is expected that the offset will not result in wrong codes, but will rather make the converter use more steps to converge to LSB if fail-safe logic is implemented where the traditional algorithm is preferred for one step if a register has already been filled. Based on these findings, it can be concluded that an effective path-finding resolution can be obtained between 9 and 5 stages. Relating this to the number of conversion steps, an average number between 3 and 7 steps respectively is found based on Table 3.1. Translating this to a Walden FOM normalized to a traditional TDC as shown in Table 3.2, if the proposed circuit is implemented into a TDC-assisted SAR ADC, it could have a 0.95 to 0.50 times lower FOM than its traditional counterpart.

Chapter 6

Conclusions and Recommendations for Future Research

In this thesis, a voltage-to-time-to-digital converter that can be utilized in a so-called TDC-assisted SAR ADC is proposed. The comparator in a SAR ADC exhibits a logarithmic decision speed based on the input voltage. By measuring this decision speed, an estimate can be made of the voltage present on the comparator nodes. With this estimate, unnecessary steps in the SAR algorithm can be skipped. Because the SAR ADC works with a binary (exponential) search algorithm, the voltage boundary for a SAR step is exponential in nature. These exponential steps, combined with the logarithmic nature of the comparator, result in linearly spaced timings. These can be measured with a linear TDC.

The thesis presents several algorithms that can be used to shorten the SAR algorithm based on the output of a TDC connected to the comparator. One of these algorithms is worked out. By measuring the comparator time each conversion step, only the binary scaled DAC switch closest to the comparator input voltage needs to be changed. Depending on the accuracy of the TDC, the average number of conversion steps for a 10-bit SAR ADC can be as low as 3 steps.

To confirm that it is possible to measure the speed of the comparator accurately without a large penalty in power consumption, a comparator and TDC combination are designed in 65nm CMOS. The comparator is designed for a 10-bit ADC in terms of noise, and exhibits a logarithmic voltage-to-time behavior. A low-power 9-stage TDC with simple calibration is implemented that can match the comparator MSB-1 to LSB timing in different corners and mismatch scenarios. A possible logic implementation is given in theory, but not fully worked out due to time constraints.

Given ideal accuracy and the power consumption of the designed blocks, a possible FOM improvement of up to 50% should achievable. This is possible because there are less DAC-changes, less comparisons and less logic steps enabled during a conversion. However, it is found that for the given implementation, the voltage-to-time-to-digital conversion is not accurate enough to ensure a full 9-stage resolution for all input voltages. This is because the TDC has to be accurate to 1 LSB for each convertible input voltage.

The simulated voltage-to-time-to-digital transfer function exhibits several non-linearities which degrade the effective path-finding resolution and therefore the average number of conversion steps will be larger than the ideal case. Given the implementation, it is expected that the realistic effective path-finding resolution lies between 5 and 9 depending on the sampled voltage, resulting in an effective average number of conversion steps between 7 and 3. The resulting Walden FOM is expected to be between 0.95 and 0.50 times that of a traditional SAR ADC with the same components.

6.1 Recommendations for Future Research

- The logic implementation for the path-finding algorithm should be carefully researched and optimized for low power. It is of the essence that it is fail-safe for all input voltages / output codes. Since the logic implementation in this thesis is only given in theory, it should be researched if it works properly for all possible scenarios.
- In this work, a double-tail type comparator has been used. Many other comparators exist, and there will be implementations possible that yield lower power (such as charge-biased comparators) or a better voltage-to-time characteristic.
- The TDC tuning can be implemented in several ways. Current starving, switchable load resistance, switchable output capacitance, etc. It should be researched which implementation is both robust and low power.
- Instead of tuning the TDC, the load capacitance of the comparator can be calibrated to achieve a matched system. As shown in theory, the voltage-to-time gain is dependent on the load seen at the comparator output. This calibration might be easier to achieve, or consume less power.
- Another way to achieve good matching of the comparator and TDC, is a complex calibration cycle that tunes each delay element separately to its required timing during startup. This way, a 1-LSB accurate transfer curve for all TDC stages might be possible.
- The model for calculating the power consumption and resulting FOMs is simplistic and could be improved. Firstly, the average DAC switching energy could be modeled more accurately based on the specific code and SAR steps. Secondly, it is currently a mix of pre- and post-layout power consumptions which is not ideal for a strong proof.
- Another way to improve the FOM is increasing converter speed. If the outermost codes are discarded (for example by employing over-ranging in the converter, or decreasing the input voltage range), only lower-step codes are left. As long as the ENOB is not compromised, a 10-bit TDC-assisted SAR ADC could be run in for example 5 steps, effectively doubling its possible speed. This can increase the FOM.
- To improve the ENOB, the LSB comparison could be done several times, taking the average of the comparisons as the binary value (majority voting, as seen in [10]). This effectively decreases the noise floor by oversampling. Improving the ENOB improves the FOM more drastically than lower power or higher sample rate.

Appendix A

FOM-model and Algorithms

A.1 FOM

To calculate the improvement in figure of merit for different algorithms, the power consumptions of each subcircuit in [6] (and tests with self-made sub-circuits) have been used. The power consumptions are first normalized to energy per conversion. Then, based on the average number of steps per algorithm given a uniformly distributed input signal, new power consumptions for each sub block are calculated. For example, the track and hold in a circuit is used once at the start of a conversion, whether the algorithm has changed or not. However, depending on the algorithm, parts of the logic, a number of DAC-switches, and the comparator are used less.

This model is not particulary accurate, mainly because the DAC-switch energy is averaged-out, even though switches towards the MSB side require more energy and are skipped more often.

The energy consumption per sub-circuit in a traditional SAR ADC are given in Table A.1 and the total converter energy per conversion is given by:

$$E_{trad,total} = E_{SH} + M_{steps,avg}(E_{DAC,avg} + E_{comp} + E_{logic,stage}) \tag{A.1}$$

Where E_{TH} is the energy consumed by the track and hold, $M_{steps,avg}$ is the average number of conversion steps (traditionally always N steps for an N-bit SAR ADC), $E_{DAC,avg}$ is the average energy consumption for a DAC switch, E_{comp} the energy consumption of a single comparison and reset and $E_{logic,stage}$ is the energy consumption of a single logic stage.

Table A.1: Energy consumption of different sub-circuits in targeted SAR ADC

Subcircuit	**Energy cons. (fJ)**	**Comment**
Track and Hold	83	Post-layout simulation of own TH circuit
Comparator	60	Post-layout simulation of designed comparator
Average Logic stage	64	Pre-layout from own simulated logic circuit
Average DAC switch	55	Pre-layout from own simulated DAC
TDC (1 stage+ 1 XOR)	6	Post-layout simulation of designed TDC
Additional Logic	6	Estimate based on 6 additional gates of minimum-sized logic for the TDC-assisted part

Filling in the equation with the numbers from the table, a total energy consumption of 1778fJ is obtained for the traditional SAR, close to the amount reported in [6]. The normalized figure-of-merit is calculated given an equal ENOB and sample-rate.

A.2 Initial Range Estimation

Figure A.1 shows the number of conversion steps depending on the output code for the Initial Range Estimation algorithm for a 10-bit SAR ADC with a 9-stage TDC. As seen in this figure, the algorithm only effectively decrease conversion steps around the mid code. This means that on average, there are not many skipped conversion steps. However, the TDC is only used once. Still, the normalized FOMs show a negligible improvement.

The normalized FOMs for 1 to 9 stages of TDC given the 10-bit converter are shown in Table A.2 based on the following equation:

$$E_{IRE,total} = E_{TH} + M_{steps,avg}(E_{DAC,avg} + E_{comp} + E_{logic,stage}) + O_{tdcstages}(E_{TDC,stage} + E_{logic,extra}) \tag{A.2}$$

Table A.2: Normalized FOMs for an M-stage TDC-assisted SAR ADC

TDC stages	1	2	3	4	5	6	7	8	9
Conv.steps (avg)	9.50	9.45	9.40	9.38	9.35	9.30	9.20	9.10	9.00
FOM (normalized)	0.99	0.99	0.99	1.00	1.00	1.00	1.00	0.99	0.99

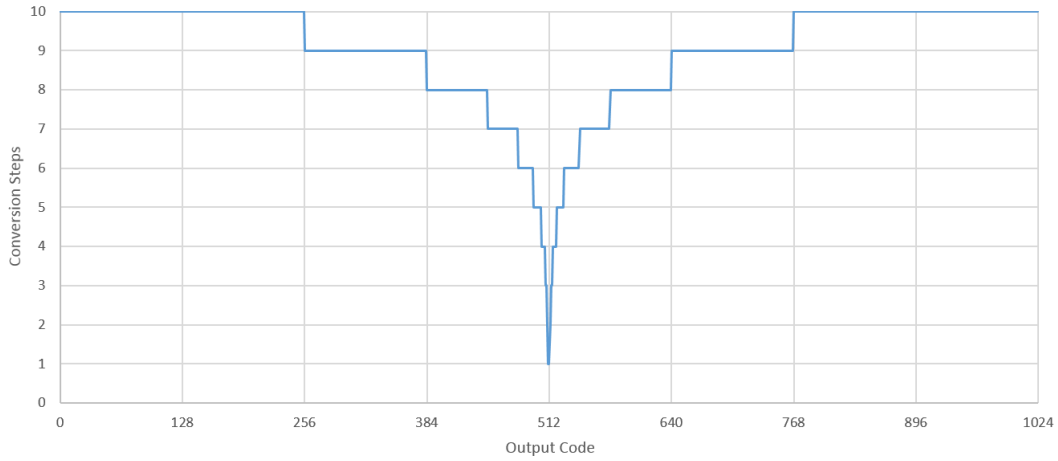

Figure A.1: Number of conversion steps for each output code given a TDC-assisted SAR ADC with a 9-stage initial range estimation.

A.3 LSB Detection

Figure A.2 shows the number of conversion steps depending on the output code for the LSB-detection algorithm for a 10-bit SAR ADC with a 1-stage TDC. Multiple TDC stages would require a difficult implementation as LSB-steps in voltage do not correspond to linear time steps. The average number of conversion steps for this type of system is around 8. The normalized FOM is then 0.88. This is based on the following equation:

$$E_{LSBD,total} = E_{TH} + M_{steps,avg}(E_{DAC,avg} + E_{comp} + E_{logic,stage} + E_{TDC,stage} + E_{logic,extra}) \quad \text{(A.3)}$$

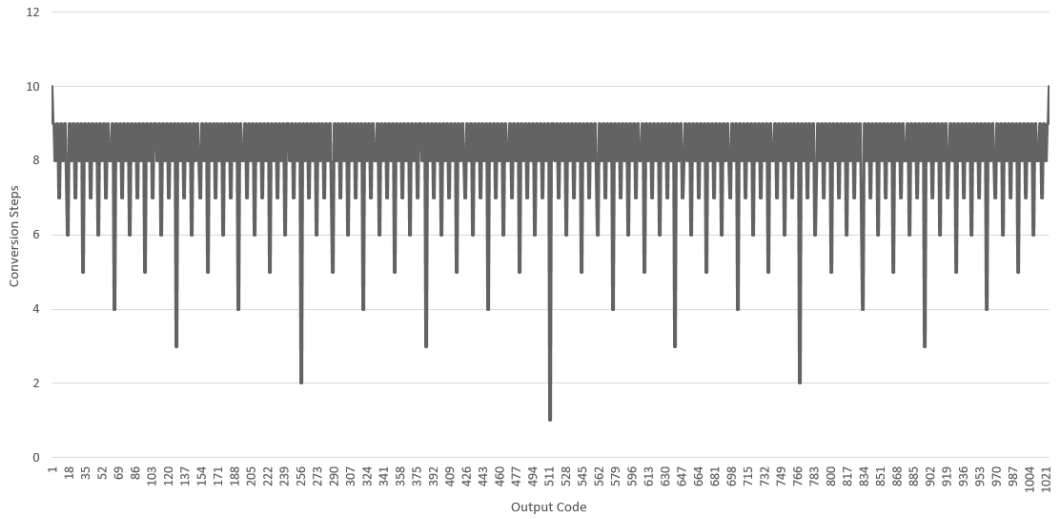

Figure A.2: Number of conversion steps for each output code given a TDC-assisted SAR ADC with LSB detection.

Appendix B

Example of several SAR ADC algorithms

Table B.1: Voltages and output codes during a traditional 10-bit SAR conversion of a $+488mV$ input voltage.

Step	V_{comp} **(mV)**	**Code**	**DAC shift (mV)**
1	488	1? ???? ????	-500
2	-12	10 ???? ????	+250
3	238	10 1??? ????	-125
4	113	10 11?? ????	-62.5
5	50.5	10 111? ????	-31.25
6	19.25	10 1111 ????	-15.63
7	3.62	10 1111 1???	-7.81
8	-4.19	10 1111 10??	+3.91
9	-0.28	10 1111 100?	+1.95
10	1.67	10 1111 1001	

Table B.2: Voltages and output codes during a 9-stage TDC assisted 10-bit SAR conversion using the pathfinding algorithm. X denotes the states filled at once after comparison.

Step	V_{comp} **(mV)**	**Code**	**TDC out**	**DAC step (mV)**
1	488	1? ???? ????	MSB-1 (skip 0)	-500
2	-12	10 XXXX X???	MSB-6 (skip 4)	+15.63
3	3.63	10 1111 1XX?	MSB-8 (skip 1)	-3.91
4	-0.28	10 1111 100?	MSB-9 (skip 0)	+1.95
5	1.67	10 1111 1001		

Table B.3: Voltages and output codes during a 6-stage TDC assisted 10-bit SAR conversion using the pathfinding algorithm. X denotes the states filled at once after comparison. No codes are skipped after the 'active' register has surpassed the TDC output pointer.

Step	V_{comp} **(mV)**	**Code**	**TDC out**	**DAC step (mV)**
1	488	1? ???? ????	MSB-1 (skip 0)	-500
2	-12	10 XXX? ????	MSB-4 (skip 2)	+62.5
3	50.5	10 111? ????	*MSB-2 (skip 0)*	-31.25
4	19.25	10 1111 ????	*MSB-4 (skip 0)*	-15.63
5	3.62	10 1111 1???	*MSB-5 (skip 0)*	-7.81
6	-4.19	10 1111 10??	*MSB-5 (skip 0)*	+3.91
7	-0.28	10 1111 100?	*MSB-6 (skip 0)*	+1.95
8	1.67	10 1111 1001	*MSB-6 (skip 0)*	

Appendix C

Layouts

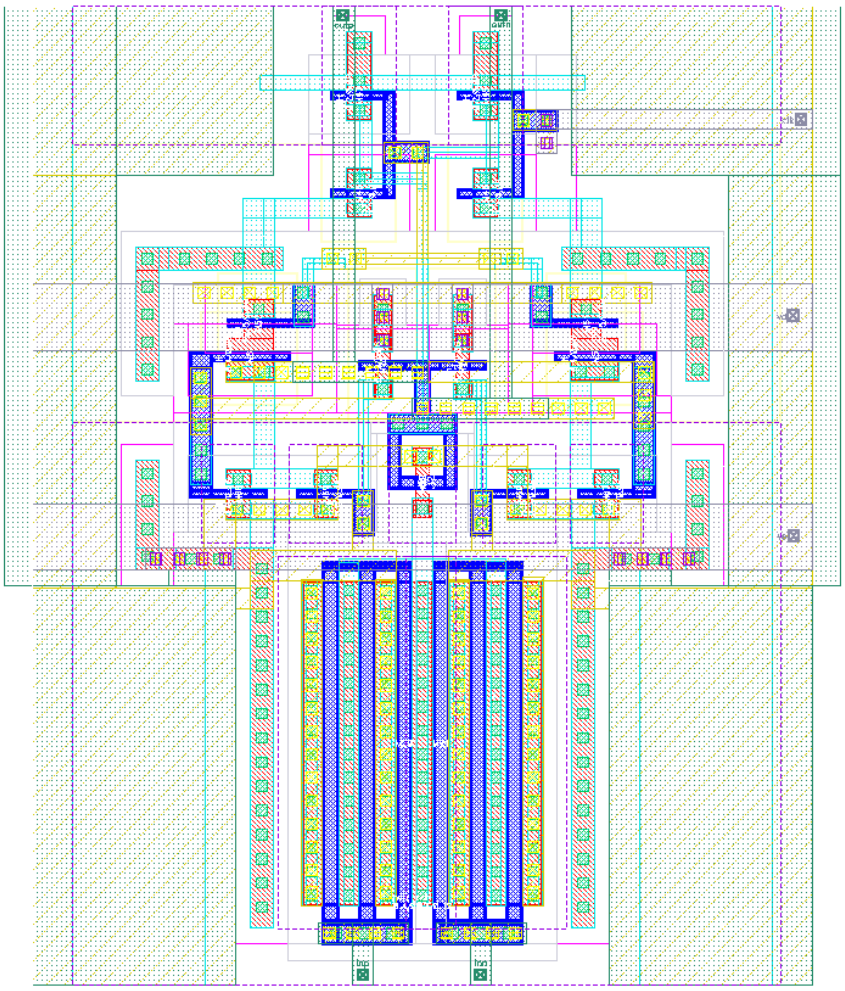

Figure C.1: Layout of the designed comparator. Custom plate capacitors around the perimeter of the circuit are placed to increase the capacitance at the DI nodes. The perimeter is also filled with bulk contacts to minimize bulk resistance. The layout is completely symmetrical.

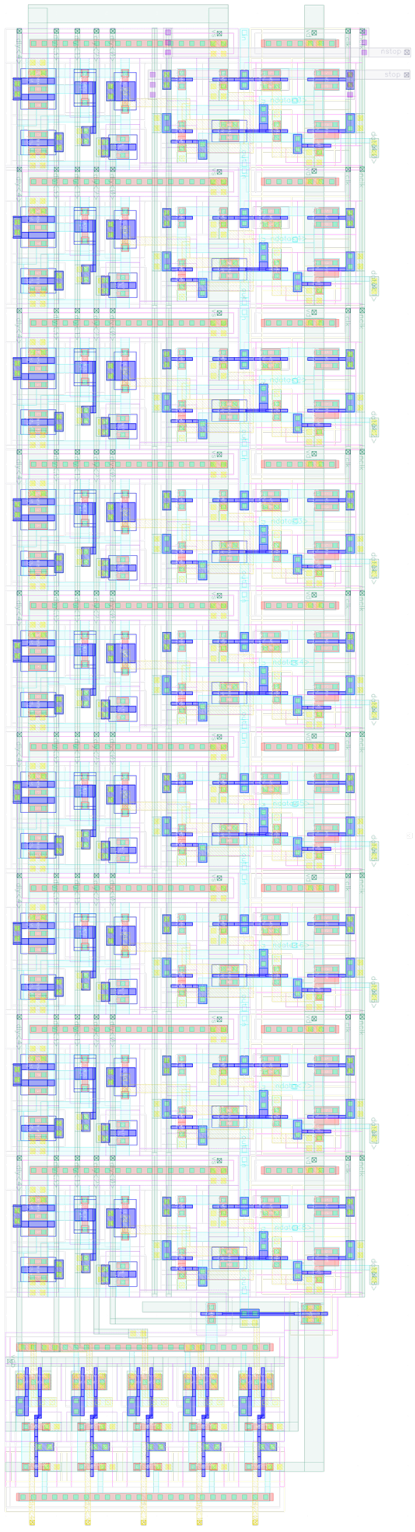

Figure C.2: Layout of the TDC. From top to bottom, 9 stages are visible with local bulk contacts between the stages. At the bottom side is the control logic for the calibration.

Bibliography

[1] YuQing Yang, A. Chokhawala, M. Alexander, J. Melanson, and D. Hester. A 114 db 68 mw chopper-stabilized stereo multi-bit audio a/d converter. In *2003 IEEE International Solid-State Circuits Conference, 2003. Digest of Technical Papers. ISSCC.*, pages 56–477 vol.1, Feb 2003. doi: 10.1109/ISSCC.2003.1234205.

[2] W. Cheng, W. Ali, Moon-Jung Choi, K. Liu, T. Tat, D. Devendorf, L. Linder, and R. Stevens. A 3b 40gs/s adc-dac in 0.12 mu/m sige. In *2004 IEEE International Solid-State Circuits Conference (IEEE Cat. No.04CH37519)*, pages 262–263 Vol.1, Feb 2004. doi: 10.1109/ISSCC.2004.1332694.

[3] R.H. Walden. Analog-to-digital converter survey and analysis. *IEEE Journal on Selected Areas in Communications*, 17(4):539–550, April 1999.

[4] P. C. Lee, J. Y. Lin, and C. C. Hsieh. A 0.4 v 1.94 fj/conversion-step 10 bit 750 ks/s sar adc with input-range-adaptive switching. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 63(12):2149–2157, Dec 2016. ISSN 1549-8328. doi: 10.1109/TCSI.2016.2617879.

[5] M. Liu, K. Pelzers, R. van Dommele, A. van Roermund, and P. Harpe. A106nw 10 b 80 ks/s sar adc with duty-cycled reference generation in 65 nm cmos. *IEEE Journal of Solid-State Circuits*, 51(10):2435–2445, Oct 2016. ISSN 0018-9200. doi: 10.1109/JSSC.2016.2587688.

[6] M. van Elzakker, E. van Tuijl, P. Geraedts, D. Schinkel, E. A. M. Klumperink, and B. Nauta. A 10-bit charge-redistribution adc consuming 1.9μ w at 1 ms/s. *IEEE Journal of Solid-State Circuits*, 45(5):1007–1015, May 2010. ISSN 0018-9200. doi: 10.1109/JSSC.2010.2043893.

[7] S. Henzler, S. Koeppe, D. Lorenz, W. Kamp, R. Kuenemund, and D. Schmitt-Landsiedel. A local passive time interpolation concept for variation-tolerant high-resolution time-to-digital conversion. *IEEE Journal of Solid-State Circuits*, 43(7):1666–1676, July 2008. ISSN 0018-9200. doi: 10.1109/JSSC.2008.922712.

[8] H. Y. Huang, J. Y. Lin, C. C. Hsieh, W. H. Chang, H. H. Tsai, and C. F. Chiu. A 9.2b 47fj/conversion-step asynchronous sar adc with input range prediction dac switching. In *2012 IEEE International Symposium on Circuits and Systems*, pages 2353–2356, May 2012. doi: 10.1109/ISCAS.2012.6271768.

[9] P. C. Lee, J. Y. Lin, and C. C. Hsieh. A 0.4 v 1.94 fj/conversion-step 10 bit 750 ks/s sar adc with input-range-adaptive switching. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 63(12):2149–2157, Dec 2016. ISSN 1549-8328. doi: 10.1109/TCSI.2016.2617879.

[10] P. Harpe, E. Cantatore, and A. van Roermund. A 10b/12b 40 ks/s sar adc with data-driven noise reduction achieving up to 10.1b enob at 2.2 fj/conversion-step. *IEEE Journal of Solid-State Circuits*, 48(12):3011–3018, Dec 2013. ISSN 0018-9200. doi: 10.1109/JSSC.2013.2278471.

[11] D. Schinkel, E. Mensink, E. Klumperink, E. van Tuijl, and B. Nauta. A double-tail latch-type voltage sense amplifier with 18ps setup and hold time. In *2007 IEEE International Solid-State Circuits Conference. Digest of Technical Papers*, pages 314–605, Feb 2007. doi: 10.1109/ISSCC.2007.373420.

[12] A. Shikata, R. Sekimoto, T. Kuroda, and H. Ishikuro. A 0.5v 1.1ms/sec 6.3fj/conversion-step sar-adc with tri-level comparator in 40nm cmos. In *2011 Symposium on VLSI Circuits - Digest of Technical Papers*, pages 262–263, June 2011.

[13] J. Guerber, H. Venkatram, M. Gande, A. Waters, and U. K. Moon. A 10-b ternary sar adc with quantization time information utilization. *IEEE Journal of Solid-State Circuits*, 47(11): 2604–2613, Nov 2012. ISSN 0018-9200. doi: 10.1109/JSSC.2012.2211696.

[14] Donald E. Knuth. *The Art of Computer Programming, Volume 3: (2Nd Ed.) Sorting and Searching*. Addison Wesley Longman Publishing Co., Inc., Redwood City, CA, USA, 1998. ISBN 0-201-89685-0.

[15] A. M. Abo and P. R. Gray. A 1.5-v, 10-bit, 14.3-ms/s cmos pipeline analog-to-digital converter. *IEEE Journal of Solid-State Circuits*, 34(5):599–606, May 1999. ISSN 0018-9200. doi: 10.1109/4.760369.

[16] Behzad Razavi. *Design of Analog CMOS Integrated Circuits*. McGraw-Hill, Inc., New York, NY, USA, 1 edition, 2001. ISBN 0072380322, 9780072380323.

[17] B. P. Ginsburg and A. P. Chandrakasan. An energy-efficient charge recycling approach for a sar converter with capacitive dac. In *2005 IEEE International Symposium on Circuits and Systems*, pages 184–187 Vol. 1, May 2005. doi: 10.1109/ISCAS.2005.1464555.

[18] J. L. McCreary and P. R. Gray. All-mos charge redistribution analog-to-digital conversion techniques. i. *IEEE Journal of Solid-State Circuits*, 10(6):371–379, Dec 1975. ISSN 0018-9200. doi: 10.1109/JSSC.1975.1050629.

[19] M. J. M. Pelgrom, A. C. J. Duinmaijer, and A. P. G. Welbers. Matching properties of mos transistors. *IEEE Journal of Solid-State Circuits*, 24(5):1433–1439, Oct 1989. ISSN 0018-9200. doi: 10.1109/JSSC.1989.572629.

[20] S. Brenna, A. Bonetti, A. Bonfanti, and A. L. Lacaita. A simulation and modeling environment for the analysis and design of charge redistribution dacs used in sar adcs. In *2014 37th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pages 74–79, May 2014. doi: 10.1109/MIPRO.2014.6859536.

[21] T. Kobayashi, K. Nogami, T. Shirotori, Y. Fujimoto, and O. Watanabe. A current-mode latch sense amplifier and a static power saving input buffer for low-power architecture. In *1992 Symposium on VLSI Circuits Digest of Technical Papers*, pages 28–29, June 1992. doi: 10.1109/VLSIC.1992.229252.

[22] A. Hajimiri and R. Heald. Design issues in cross-coupled inverter sense amplifier. In *Circuits and Systems, 1998. ISCAS '98. Proceedings of the 1998 IEEE International Symposium on*, volume 2, pages 149–152 vol.2, May 1998. doi: 10.1109/ISCAS.1998.706863.

[23] S. Babayan-Mashhadi and R. Lotfi. Analysis and design of a low-voltage low-power double-tail comparator. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 22(2): 343–352, Feb 2014. ISSN 1063-8210. doi: 10.1109/TVLSI.2013.2241799.

[24] Stephan Henzler. *Time-to-Digital Converters*. Springer Publishing Company, Incorporated, 1st edition, 2010. ISBN 9048186277, 9789048186273.

[25] Jyh-Ming Wang, Sung-Chuan Fang, and Wu-Shiung Feng. New efficient designs for xor and xnor functions on the transistor level. *IEEE Journal of Solid-State Circuits*, 29(7):780–786, Jul 1994. ISSN 0018-9200. doi: 10.1109/4.303715.