Department of Physics and Astronomy
University of Heidelberg

Master Thesis in Physics
submitted by

Milena Czierlinski

born in Bad Segeberg (Germany)

2022

The implementation of a dual-mode DAC for an SAR ADC

This Master Thesis has been carried out by
Milena Czierlinski
at the
Kirchhoff Institute for Physics in Heidelberg
under the supervision of
Dr. Johannes Schemmel

Abstract

This thesis presents the development of a dual-mode successive approximation register analog-to-digital converter for the deployment in the neuromorphic system BrainScaleS-2. On the one hand, a sampling frequency of 62.5 MS/s is provided with a 7-bit resolution. On the other hand, the precision can be increased to 8-bit, in exchange for a reduced sampling frequency of 15.6 MS/s. This is achieved by the combination of two capacitive 7-bit digital-to-analog converters, either operating in a time-interleaved fashion in the fast mode, or jointly in the precise mode. For the latter, a single-ended set-and-down switching procedure is implemented to save time and energy. Moreover, the LSB capacitance is split, saving further energy and reducing the area. The focus of this thesis lies on the development of the two conversion methods, as well as the implementation of the digital-to-analog conversion sub-circuit. Furthermore, the performance of the full analog-to-digital converter is analyzed in simulations under various operation conditions and the impact of physical effects is evaluated.

Zusammenfassung

Diese Arbeit präsentiert die Entwicklung eines dualen Analog-Digital-Wandlers, welcher das Verfahren der sukzessiven Approximation verwendet, für den Einsatz im neuromorphen System BrainScaleS-2. Einerseits wird eine Aufnahmefrequenz von 62.5 MS/s mit 7-Bit Auflösung zur Verfügung gestellt. Andererseits kann die Präzision auf 8-Bit erweitert werden auf Kosten einer reduzierten Aufnahmerate von 15.6 MS/s. Dies wird durch die Kombination von zwei kapazitiven 7-Bit Digital-Analog-Wandlern erreicht, die entweder abwechselnd im schnellen Modus oder gemeinsam im präzisen Modus betrieben werden. Für den letzteren wird ein referenzbezoger set-and-down Schaltungsablauf implementiert, um Zeit und Energie zu sparen. Außerdem wird die LSB Kapazität geteilt, was weitere Energie sowie Fläche spart. Das Hauptaugenmerk dieser Arbeit liegt auf der Entwicklung dieser beiden Umwandlungsmethoden und der Realisierung der Schaltung, welche die Digital-Analog-Wandlung ausführt. Des Weiteren wird das Verhalten des gesamten Analog-Digital-Wandlers in Simulationen unter verschiedenen Betriebsumständen analysiert und die Auswirkungen physikalischer Effekte evaluiert.

Contents

1. Introduction

In today's world analog-to-digital conversion is everywhere around us. The digitization over the last decades has allowed us to communicate with people all over the world, to replay images and sounds, to visualize medical scans and to do much more by the means of only a hand-held device (Pelgrom [2017]). Nevertheless, all these functionalities are based on analog implementations. Thus, analog-to-digital converters (abbreviated by ADCs) form an important connection between the physical and digital world.

The particular application we consider in this work is the BrainScaleS-2 neuromorphic hardware (Schemmel et al. [2020]). It constitutes a semiconductor device, comprising integrated circuits that emulate the behavior of neurons. For analysis it is necessary to digitize analog traces, e.g. representing the neurons' potentials or correlation data. Desirably, this is performed with a high sampling rate, but a low energy consumption at the same time. Hence, the goal of this thesis is to develop an ADC, striving for certain target specifications.

To do so, after a general introduction to the topic of analog-to-digital conversion, different ADC architectures are examined. Following from this, the successive approximation register analog-to-digital converter (Kugelstadt [2005]) is found to constitute the best choice for our application. Thus, the subsequent chapter treats the implementation of its main component, the digital-to-analog converter. Therefore, we apply the principle of charge redistribution, utilizing a single binary-weighted capacitor array for both sampling and to perform the conversion. To reduce the total capacitance, furthermore, different configurations of capacitors are analyzed with regard to impairments of the performance, given physical circumstances (Allen and Holberg [2002]).

In the following chapter, various methods are discussed to implement multiple modes of operation in a single ADC. Namely, on the one hand, a fast mode with a target sampling frequency of 125 MS/s is aspired. For this, a 6-bit resolution covering the upper half of the full scale voltage is aimed for. On the other hand, also a precise mode with 8-bit resolution is desired. Here, however, a reduced sampling frequency is accepted as a trade-off. A solution is found utilizing two 7-bit capacitive arrays. Either, they function in a time-interleaved fashion in the fast mode, or they are operated jointly

to achieve an 8-bit precision. The conversion methods are optimized with regard to a low energy consumption.

The last chapter, then, treats the physical implementation of the ADC. Moreover, its performance given different operation conditions is evaluated, as well as the impact introduced by the choice of layout. Additionally, the energy consumption of both modes of operation is simulated and analyzed for various input voltages. Finally, the obtained results are discussed and modifications to achieve improvements are suggested.

2. Background

Analog-to-digital conversion has gained increased importance with the development of electronic systems in many areas (Rapuano et al. [2005]). By now, it can be found in nearly every aspect of today's everyday life. How exactly is this possible? At the basis of everything we will encounter physics. The behavior of electronic charge carriers in conductors can be controlled by applying appropriate potentials. Making use of different materials, we can construct electronic components that fulfill various tasks. Connected in the right way, they allow us to build circuits, whose measurable features like voltages, currents or charges represent quantities of interest in real-world applications (Baker [2010]). However, purely analog processing quickly meets its limits, compared to the possibilities offered by digital processing (Vardhini and Makkena [2021]). In contrast to time-continuous analog signals, digital signals consist of time-discrete samples. They can be represented well-distinguishable, making them easy to store and insensitive to noise. Moreover, digitization allows advanced processing like applying complex calculations flexibly, which is not possible in analog applications, where circuits are fixed once they are manufactured. Hence, in order to use these advantages, analog signals need to be converted to the digital domain. This draws a principal focus to analog-to-digital converters as they are the key component linking the physical and digital realms.

Since the range of requirements for ADCs is as wide as the range of applications, various types of architectures have been developed to fulfill the conversion (Bashir et al. [2016]). Their working principles are optimized for certain aspects, such that it is necessary to choose the architecture that fits the requirements of a considered application best.

2.1. Analog-to-digital conversion

Converting analog traces to digital signals, always a loss of information will occur. The reason for this is that digital signals are discrete in time, as well as in amplitude, which is referred to as quantization (Pelgrom [2017], Carusone et al. [2013], Allen and Holberg [2002]). Besides the impairments introduced by quantization, static errors in

the gain and offset of a conversion might occur. Furthermore, an important measure of the performance of an analog-to-digital converter is its linearity. In case an ADC is used for converting a periodic signal, which is common in applications concerning communication, also the power ratio between the signal and noise or other disturbances is analyzed. From this, further quantities characterizing the ADC, e.g. the effective number of bits, can be derived. However, since in our application we don't have the case of period input signals, the main focus will be on the linearity.

2.1.1. Basics

Now, how does the conversion between the analog and digital domain work? Analog signals have a continuous amplitude at every given moment in time. Digital signals, however, are quantized values that only exist at the moments their were sampled. So, firstly the analog signal needs to be sampled at given points in time. The closer these intervals, the more accurate the reconstruction will be. Then, for each sample, the digital code representing the analog input best is determined. Since digital values are discrete by definition, this approximation will never represent the analog signal perfectly (see figure 2.1). The inaccuracy introduced by quantization is intrinsic to analog-to-digital conversion and is known by the name quantization error or quantization noise. Increasing the resolution reduces this error, but it can never be fully avoided. So, let's take a look at it in more detail.

Figure 2.2a illustrates the ideal transfer function of a 3-bit ADC. With increasing input voltage V_{in}, also the digital output D_{out} rises monotonically. The output code can take discrete values that are represented by a three bit binary encoding. Here, the first bit indicates whether the input voltage is higher or lower than half of the reference voltage V_{ref}. Therefore, it is the most significant bit (abbreviated by MSB). In contrast, the last bit shows the smallest difference that can be dissolved. Hence, it is called the

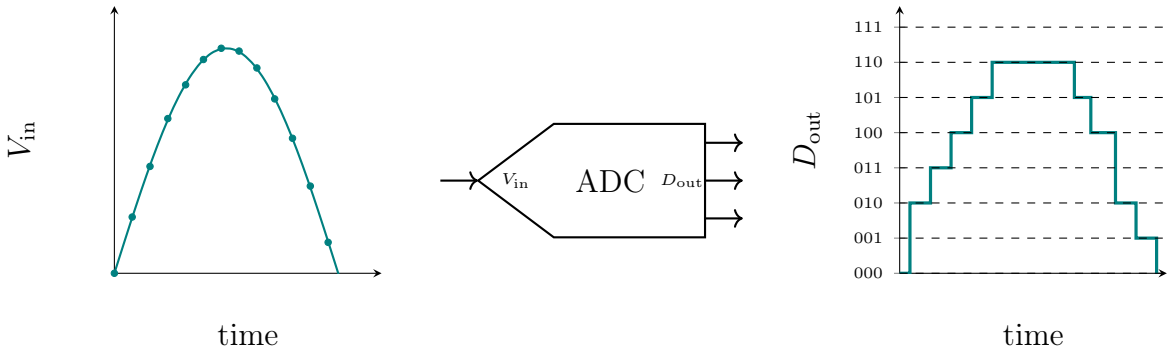
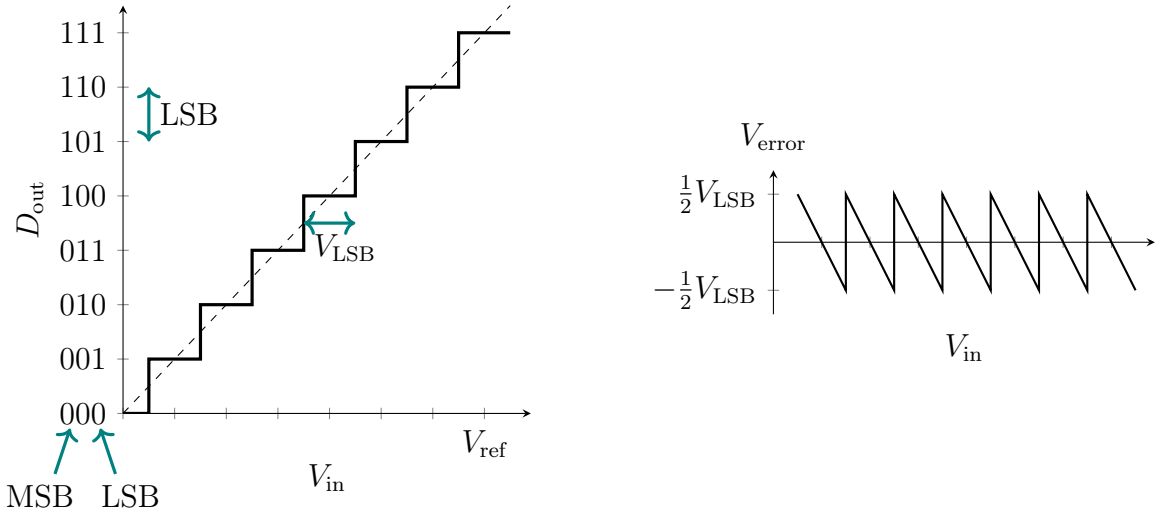


Figure 2.1.: Functioning of an analog-to-digital converter: The continuous input voltage is sampled and approximated by quantized output codes.



(a) Ideal transfer function

(b) Quantization error

Figure 2.2.: The ideal behavior of a 3-bit ADC shows an error that is intrinsic to quantization.

least significant bit or LSB. Consequently, the step size between two digital output levels amounts to one LSB. The associated voltage step we'll denote by V_{LSB}. If the conversion range spans the reference voltage completely, the minimum step length will be

$$V_{\text{LSB}} = \frac{V_{\text{ref}}}{2^n}, \tag{2.1}$$

given a resolution n. In the ideal case, a change in the digital output from i to $i+1$ will occur at the trip or decision level

$$V_{\text{ideal}}(i) = \left(i + \frac{1}{2}\right) \cdot V_{\text{LSB}}. \tag{2.2}$$

In the plot the dashed line visualizes the ideal continuous output. The error introduced by quantization, hence, is the difference between this continuous and the discrete output. It is depicted in figure 2.2b. To estimate the impact of the quantization error, we'll assume that every input voltage is equally probable. Following from this, the occurrence of an input is uniformly distributed between two trip levels. And consequently, so is the deviation V_{error} from the ideal output. This allows us to calculate the estimation value for the variance

$$\text{Var}(V_{\text{error}}) = \frac{1}{V_{\text{LSB}}} \int_{-0.5V_{\text{LSB}}}^{+0.5V_{\text{LSB}}} V_{\text{error}}(\epsilon)^2 d\epsilon = \frac{1}{V_{\text{LSB}}} \int_{-0.5V_{\text{LSB}}}^{+0.5V_{\text{LSB}}} \epsilon^2 d\epsilon = \frac{V_{\text{LSB}}^2}{12}. \tag{2.3}$$

The higher the resolution, i.e. the smaller V_{LSB}, the smaller also the effect of the quan-

tization error will be. Despite nothing can be done to prevent quantization noise, in designs with moderate resolution or higher, it usually poses no problem, since other distortions will be more severe as we'll discuss later.

Generally, analog-to-digital converters can be divided into two categories: Nyquist and oversampling converters. Extending the mathematical theory of communication of Nyquist [1928], Shannon [1949] stated that in order to unambiguously reconstruct a signal, the sampling frequency must be at least twice as large as the bandwidth of the signal. This is also know as the Nyquist criterion and the associated sampling frequency as Nyquist frequency. Consequently, analog-to-digital converters functioning close to the Nyquist frequency, are called Nyquist converters. Here, the bandwidth can start at DC and range over a wide spectrum. Oversampling converters, however, function at a sampling frequency that is significantly larger, than the Nyquist frequency. They aim at compensating a reduced accuracy per sample in a limited bandwidth by using a higher sampling rate. Since in this work we have an application that doesn't usually show a periodic signal, we will focus on Nyquist analog-to-digital converters. So, let's take a look at further effects introduced by quantization.

2.1.2. Offset and gain error

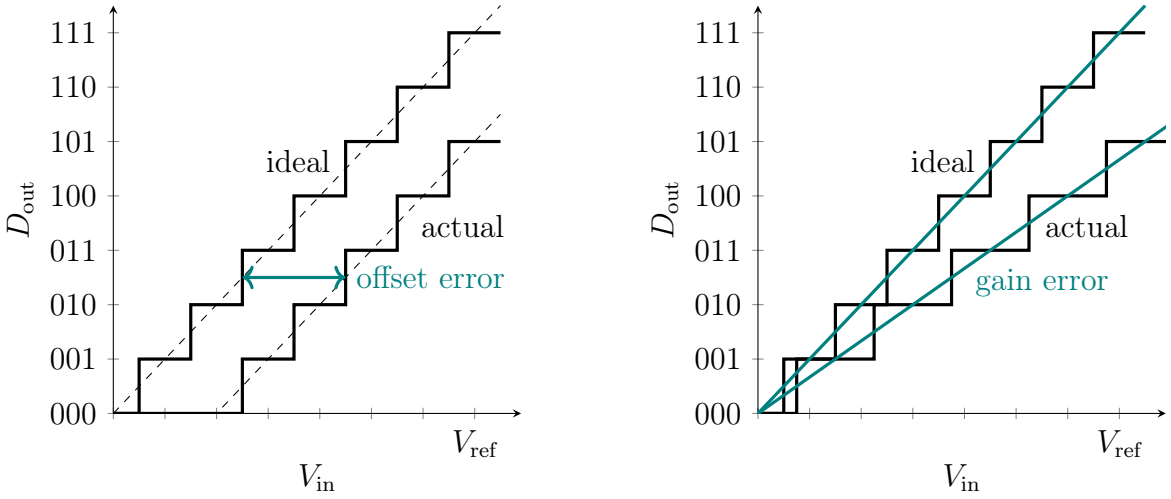
The ideal ADC behavior as depicted in figure 2.2a in most cases is not what we observe in reality. There are various sources leading to non-idealities in the conversion. Consequently, the actual transfer curve will display these errors. For now, let's assume the linearity is not affected.

Static errors e.g. caused by mismatch may lead to an offset in the conversion characteristic. This is indicated by a shift of the transfer function, when sweeping the analog input voltage. Hence, in physical implementations the output does not increase at every $(i+\frac{1}{2})\cdot V_{\mathrm{LSB}}$, but the voltage, when the output code changes from i to $i+1$ we'll denote as $V_{\mathrm{real}}(i)$. The amount by which the measured finite resolution characteristic differs from its ideal counterpart is called the offset error. Mathematically, this can be expressed by

$$E_{\mathrm{offset}} = \frac{V_{\mathrm{real}}(0) - V_{\mathrm{ideal}}(0)}{V_{\mathrm{LSB}}}. \tag{2.4}$$

We note that the sign of E_{offset} indicates the direction in which the curve is shifted.

Another way of how the transfer function may be influenced is by a change in its slope. If the input voltage range representing the same digital output code is larger than V_{LSB}, the total slope of the ADC curve will be flatter. Analogously, if the input



(a) Offset error

(b) Gain error

Figure 2.3.: The offset error indicates a shift of the starting point of the conversion, whereas the gain error shows the deviation of its slope.

range is shorter than V_{LSB}, the total slope will be steeper. This mismatch is measured by the gain error:

$$E_{\text{gain}} = \frac{(V_{\text{real}}(2^n - 2) - V_{\text{real}}(0)) - (V_{\text{ideal}}(2^n - 2) - V_{\text{ideal}}(0))}{V_{\text{LSB}}} \tag{2.5}$$

Often, these two errors occur jointly, e.g. if the conversion doesn't use the full range of the reference voltage, but only starts at a higher voltage and saturates early. Then, in order to counteract the offset error, the curve will be steeper, introducing a gain error. However, since both these errors are static and influence all codes at every time equally, they can be handled fairly well.

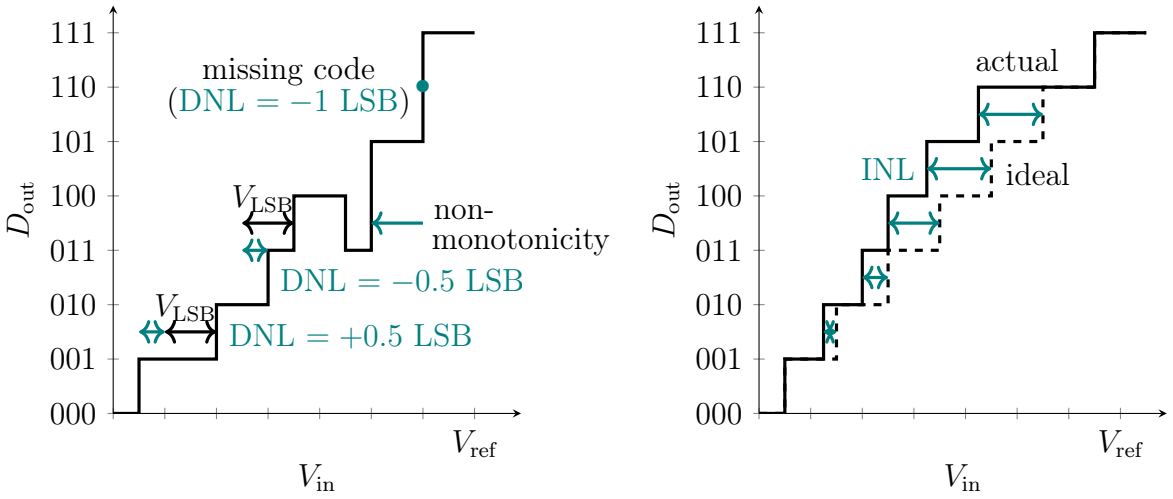
2.1.3. Differential and Integral non-linearity

Furthermore, there are also non-idealities that affect the linearity of the ADC. Before, we only considered cases, where the overall behavior was influenced, but where each individual output level had the same input width. In real ADCs, however, this usually is not the case, but discrepancies are present. They typically are characterized by two measures: differential and integral non-linearity.

The differential non-linearity (DNL) indicates the deviation of each step with respect to the ideal LSB size V_{LSB}. Its mathematical expression is

$$\text{DNL}(i) = \frac{V_{\text{real}}(i+1) - V_{\text{real}}(i)}{V_{\text{LSB}}} - 1, \ \forall i = 0, 1, ...(2^N - 2). \tag{2.6}$$

Sometimes, the DNL is not stated in units of LSB, but it can also be written as ratio



(a) Differential non-linearity

(b) Integral non-linearity

Figure 2.4.: The differential non-linearity indicates deviations from the ideal step size, while the integral non-linearity shows discrepancies from the ideal transfer function.

of the full scale voltage. Moreover, instead of the DNL spectrum, it is also common to only indicate its minimum and maximum or absolute maximum. Figure 2.4a shows an exemplary ADC transfer function. Depicted in the lower part there is a step that maps the input voltage to the identical output code for 1.5 times as long as in the ideal case. Thus, for the corresponding code the DNL is +0.5 LSB. This is followed by a step that ranges half times shorter than V_{LSB}, which is equivalent to a DNL of −0.5 LSB.

Furthermore, the transfer curve in figure 2.4a displays two extreme situations of code errors. Firstly, even though the input voltage increases, the output code steps back to a smaller value. This is unwanted in every application and, therefore, should be avoided by all means. Lastly, a case is shown where the digital output rises by multiple levels at the same time. As a consequence, the code that is skipped doesn't map to any analog input. For that reason, it is called a missing code. Since its step size is non-existing and, thus, one times V_{LSB} too short, a missing code is equivalent to a DNL of −1 LSB. Obviously, a step cannot be smaller than non-existing, so −1 LSB also represents the lower bound for the DNL.

Besides the differential non-linearity, there is the integral non-linearity (INL) that gives information about the finite resolution characteristic. Instead of the deviations regarding the step width, the INL expresses the deviations of the measured trip levels compared to their ideal position.

$$\text{INL}(i) = \frac{V_{\text{real}}(i) - V_{\text{ideal}}(i)}{V_{\text{LSB}}}, \ \forall i = 0, 1, ...(2^N - 2) \tag{2.7}$$

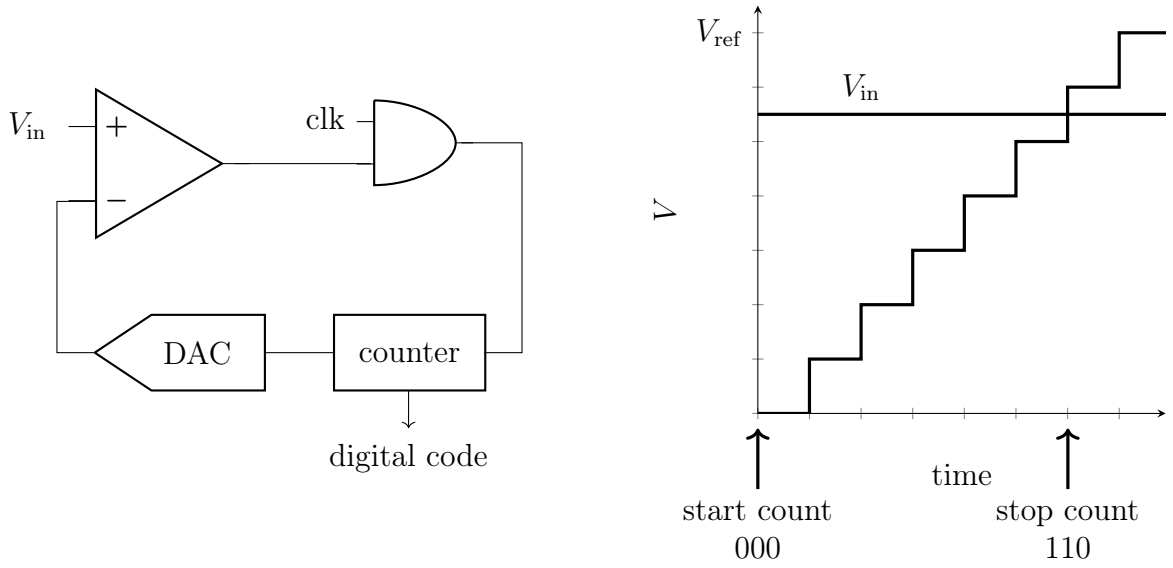
Here, $V_{\text{ideal}}(i)$ already takes into account the offset and gain error and can be defined using either the end-point of the conversion range or the best fit through it. Analogously to the DNL, the INL can be expressed in units of LSB or as percentage of the full scale voltage. Furthermore, also for this spectrum sometimes only its minimum and maximum or absolute maximum are stated. Figure 2.4b visualizes an exemplary transfer function and its ideal counterpart, using the end-points as a reference.

2.2. ADC architectures

Analog-to-digital converters can be found in a wide range of electronic applications, e.g. signal processing, measurement techniques, cameras, and electronic systems for biomedicine, communication and entertainment (Bashir et al. [2016], Rapuano et al. [2005]). All these different use cases come with their own requirements: Sometimes a certain resolution is necessary, other times the main focus is the conversion speed, energy consumption, size, accuracy, complexity or cost. These different factors are not independent, but correlated to each other. Some will come hand in hand, others, however, are coupled by an inverse relationship. The task of a designer is to find an implementation that fulfills the design requirements at the lowest possible trade-offs. For this, a variety of different ADC architectures have been developed over time, each coming with their own advantages and disadvantages regarding certain aspects (Pelgrom [2017], Carusone et al. [2013], Bashir et al. [2016]). In the following we will take a look at some of the commonly used architectures.

2.2.1. Counting ADCs

A simple analog-to-digital converter architecture is the counting ADC. Its main components are a digital-to-analog converter (DAC), a comparator and a counter (see figure 2.5a). Its operation is as follows: The unknown input voltage V_{in} is applied at the comparator, where it is compared to the output of the digital-to-analog converter, starting at its lowest level. If the DAC output does not exceed V_{in}, the value of the counter will be increased by one step. Since its output is connected to the DAC, which itself is connected to the comparator, also the voltage V_{in} is compared to rises. This is repeated until the output of the digital-to-analog converter exceeds the input voltage. At that moment, the counter is configured, such that its value is proportional to V_{in} (figure 2.5b). Hence, the counter directly provides the digital output code. An advantage of this design is that, due to its simplicity, it doesn't demand a large area. However, compared to other analog-to-digital converters, the counting ADC only has a low sampling rate. The time it takes to convert the input voltage behaves linearly regard-



(a) Block diagram of a counting ADC.

(b) Operation principle of a counting ADC.

Figure 2.5.: In a counting ADC the unknown input voltage is compared to a gradually increasing reference voltage. When the reference level reaches the input, the counter portrays the digital output code

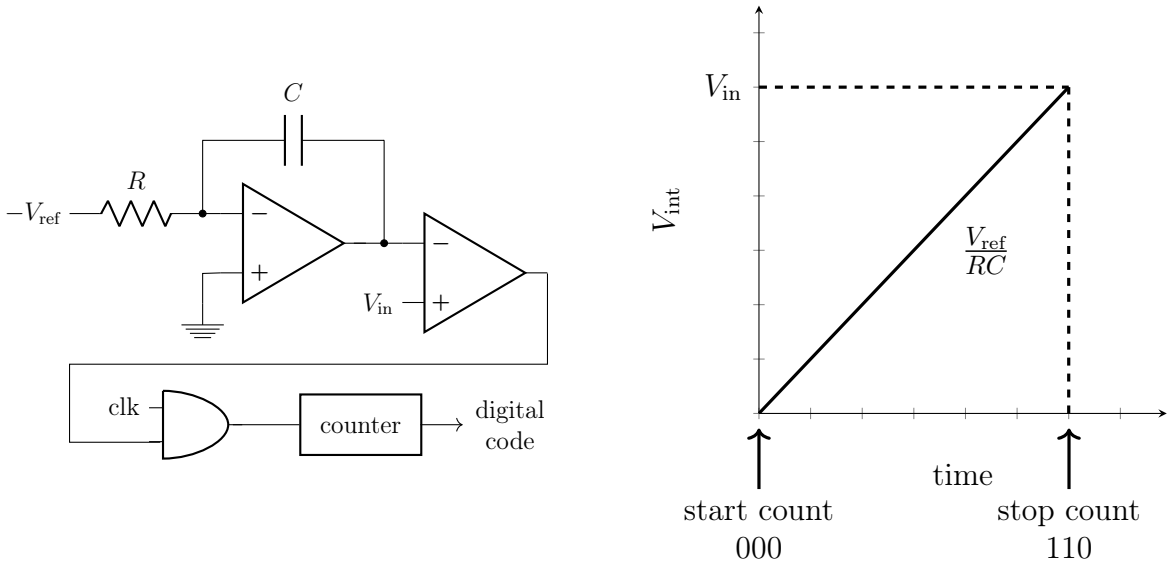
ing the possible output codes. Especially for higher resolutions this is not very efficient.

A modified version of the ramp ADC, that to some degree overcomes this problem, poses the tracking ADC. It offers an alternative behaving advantageous for signals that don't change much between subsequent samples. Here, instead of starting the search for each new sample again from the beginning, the previous comparison result is taken as a starting point. Therefore, the sampling rate of the tracking ADC can be sped up compared to the ramp ADC. This speed up comes at the cost of a more complex logic which is needed to implement the two-way search. And, a problem that may arise using tracking ADCs is that the output of a constant analog input may oscillate between two states.

2.2.2. Integrating ADCs

The integrating analog-to-digital converter can be found in two variants: single slope and dual slope. What both versions have in common are an integrator connected to a comparator. Firstly, let's take a look at the single slope integrating ADC.

In the beginning of a conversion the integrator is charged by the negative reference potential $-V_{ref}$ (see figure 2.6a). This goes on until the voltage on its capacitor is equal to the input voltage. At that point it holds that



(a) Block diagram of a single slope integrating ADC.

(b) Operation principle of a single slope integrating ADC.

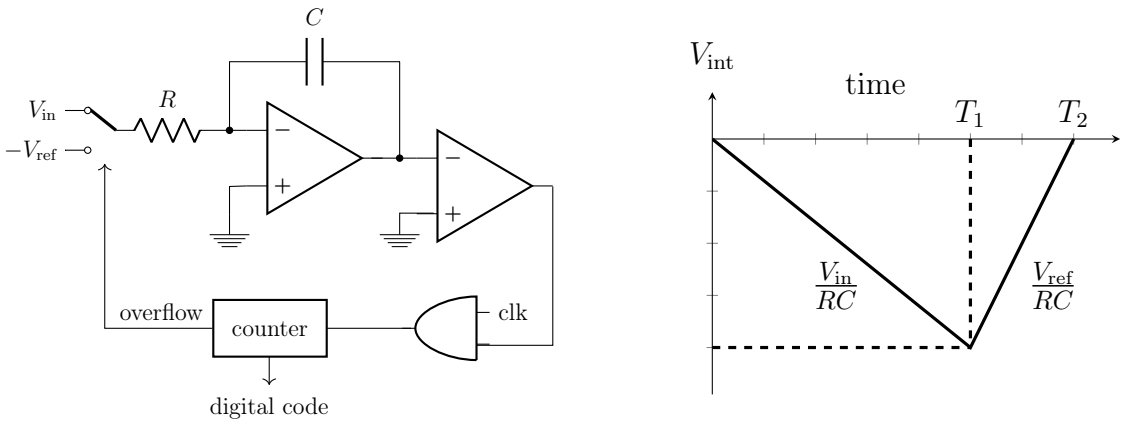
Figure 2.6.: In a single slope integrating ADC an integrator is charged by the reference voltage, unit it reaches the input voltage. The time to do so is proportional to the digital output code.

$$V_{\text{int}} = -\int_0^T \frac{-V_{\text{ref}}}{RC}\, dt = \frac{V_{\text{ref}}}{RC}\, T \stackrel{!}{=} V_{\text{in}} \Rightarrow V_{\text{in}} \propto T \tag{2.8}$$

Hence, if we assume R, C and V_{ref} to be constant, the input voltage is proportional to the time the integrator needed to charge. Consequently, it can be represented by the value of a counter that is active as long as the slope of V_{int} rises (see figure 2.6b).

This analog-to-digital converter charms by its simplicity which allows to implement it in a small area, similar to the counting ADC. The generation of the steadily increasing reference voltage makes up the main part concerning its energy consumption. In a highly-parallel system, however, this voltage may be generated globally, such that it can be used by multiple single slope channels simultaneously. Thus, operated this way, the overall energy consumption can be held at a low level. However, a disadvantage of this architecture is that in physical applications the assumption of a constant resistance and capacitance is impaired, due to changes in temperature and aging effects. This affects the slope of V_{int} and consequently also the time until the input voltage is reached. Hence, the digital output code will be compromised.

To overcome this shortcoming, the dual slope integrating ADC can be used (figure 2.7a). Here, firstly the positive input voltage is connected to the integrator for a fixed time T_1, leading to a decrease of V_{int}. T_1 is implemented as the time, until the counter



(a) Block diagram of a dual slope integrating ADC.

(b) Operation principle of a dual slope integrating ADC.

Figure 2.7.: In dual slope ADCs firstly the input voltage is integrated for a fixed time. Afterwards, the time unit the initial value is regained generates the digital output code.

reaches its maximum and the overflow bit is enabled. This triggers a switch that disconnects V_{in} and replaces it with $-V_{\text{ref}}$ at the input of the integrator. Due to the different sign, V_{int} rises again until it becomes zero at time T_2. The resulting function looks as follows:

$$V_{\text{int}} = -\int_0^{T1} \frac{V_{\text{in}}}{RC} dt \; - \int_{T1}^{T2} \frac{-V_{\text{ref}}}{RC} dt \; = \; -\frac{V_{\text{in}}}{RC} \; T_1 + \frac{V_{\text{ref}}}{RC} \; (T_2 - T_1) \stackrel{!}{=} 0. \tag{2.9}$$

Solving for V_{in}, we find that RC cancels out and we get

$$V_{\text{in}} = \frac{T_2 - T_1}{T_1} \; V_{\text{ref}}. \tag{2.10}$$

Thus, for a fixed T_1 and V_{ref}, the input voltage is proportional to $T_2 - T_1$. Since at time T_1 the counter is reset, $T_2 - T_1$, and thereby the digital representation of the analog input, can be directly read off the counter.

Equally to the single slope integrating ADC, the area and power consumption of this analog-to-digital converter are low. Furthermore, since this method is independent of the exact values of R and C, also its accuracy is very high. Therefore, it is commonly used in measurement systems. However, a disadvantage poses its long conversion time. For this reason, it is usually not used in signal processing or other applications requiring continuous sampling.

2.2.3. Flash ADCs

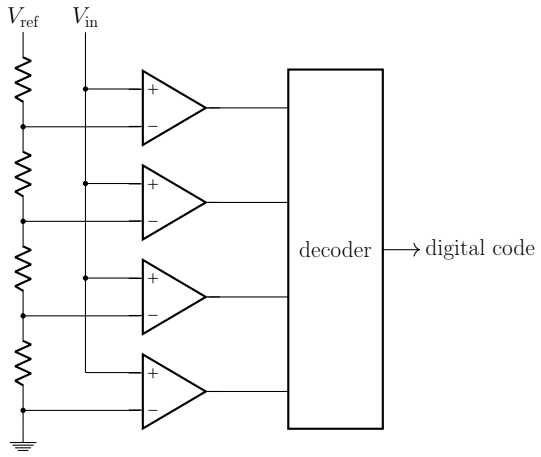


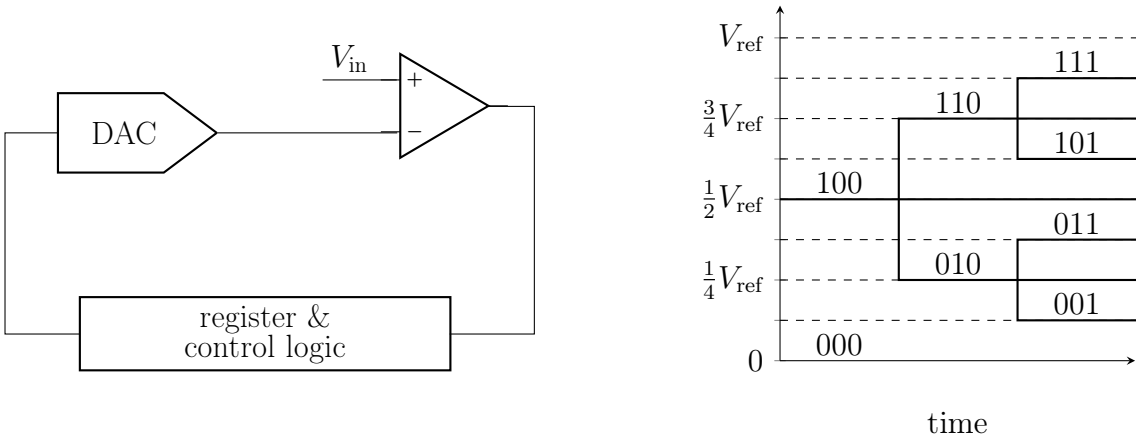
Figure 2.8.: Block diagram of a flash ADC. The input is compared to multiple reference voltage levels simultaneously, allowing to perform the conversion in a single cycle.

To overcome the limitation of a low sampling rate, flash analog-to-digital converters can be used. In this architecture the unknown input voltage is compared to multiple reference voltages simultaneously and the individual comparison results are converted to the output code (see figure 2.8). Hence, the conversion can be realized in a single cycle. Of course this speed demands a trade-off: The generation of the reference voltage levels, which for example can be implemented using a resistor ladder, consumes a lot of power. Moreover, the number of reference levels and comparators scales exponentially with the number of bits. Thus, flash ADCs are usually only feasible for designs with a small resolution.

2.2.4. Successive approximation register ADCs

One of the most versatile analog-to-digital converters is the successive approximation register (SAR) ADC. It offers a reasonable quick conversion time or can be optimized for high accuracy, while in every case providing a very low power consumption. This is due to its simple circuit. All it requires is a single comparator, a digital-to-analog converter and some digital control logic, including the output register (figure 2.9a).

The successive approximation procedure, that is applied to determine the digital representation of the input voltage, is based on the binary search algorithm. The first decision the comparator makes is between the input voltage and $\frac{1}{2}V_{\text{ref}}$, which is provided by the digital-to-analog converter. If V_{in} is larger, in the next step it will be compared



(a) Block diagram of an SAR ADC.

(b) Operation principle of an SAR ADC.

Figure 2.9.: The successive approximation principle applies a binary search procedure to determine the output code.

to $\frac{3}{4}V_{\text{ref}}$. However, if the comparator decision is negative, the next comparison will be with respect to $\frac{1}{4}V_{\text{ref}}$. This procedure is repeated in every iteration i, such that the digital-to-analog converter gradually approximates the unknown input voltage. If V_{in} exceeds the output of the DAC, it will be increased by $\frac{1}{2^i}V_{\text{ref}}$. Otherwise, $\frac{1}{2^i}V_{\text{ref}}$ will be subtracted. To do so, the digital-to-analog converter can be implemented e.g. using an array of binary scaled capacitors. After n cycles, the DAC will display the n-bit approximation of the input voltage. Hence, considering higher resolutions, the conversion times are far lower than those of ADCs with linear conversion speed. But, due to the iterative procedure, it will never be able to perform the conversion within a single clock cycle like the flash ADC.

2.2.5. Pipelined and Algorithmic ADCs

Pipelined analog-to-digital converters offer another approach, where a new sample can be taken every clock cycle. Like the successive approximation register ADC, it applies an iterative search. However, instead of dividing the reference voltage in half every iteration, here, the approximated input voltage at the comparator V_{comp} is doubled. For this, an accurate gain of two is required. The input voltage is sampled and compared to a common mode voltage. Depending on the result, in the next step the voltage at the comparator is doubled and $\frac{1}{2}V_{\text{ref}}$ is either added or subtracted from it, such that V_{comp} stays within the same range. This is repeated by n stages for an n-bit conversion. Since all stages convert subsequent samples in parallel, the pipelined ADC can provide an output code every clock cycle. Nevertheless, due to the iterative procedure, the latency to perform an n-bit conversion stays n cycles.

The pipelined analog-to-digital converter with its multiple stages trades sampling speed for area. However, also in cases, where area is critical, the iterative procedure of doubling voltages can be applied. Then, the same stage is simply used for all conversion steps. This is known by the name algorithmic analog-to-digital converter. As a consequence, the conversion of a subsequent sample can only begin, after the former has finished, decreasing its throughput. This example nicely visualizes the trade-off between different design targets.

2.2.6. Time-interleaved ADCs

In applications depending on a very high-speed conversion, this can be achieved by operating multiple analog-to-digital converters in parallel. Such designs are generally called time-interleaved ADCs. Often, they make use of successive approximation register analog-to-digital converters. The sampling of the input voltage happens staggered by the sub-ADCs. Then, each of them performs the conversion with its lower rate, before the outputs are combined again, resulting in higher total conversion frequency. The speed-up factor k equals the number of sub-ADCs that are used. What is most important in time-interleaving is an accurate matching between the single analog-to-digital converters. If one of them shows an offset or gain error, this will be visible in every k^{th} sample. But, if the mismatches can be identified precisely, it is possible to cancel them out digitally. Besides the additional logic for assigning inputs and outputs, time-interleaved analog-to-digital converters require the sum of area and energy of all of their sub-ADCs. However, if the conversion frequency is the critical design target, they'll offer a simple way to achieve this speed up.

2.2.7. Hybrid ADCs

So far, we have seen that all architectures have their own advantages and disadvantages. Hybrid designs combine two or more different analog-to-digital converters with the aim to make use of the best of all of them. For example, a resistor-capacitor hybrid analog-to-digital converter combines a flash ADC with a capacitor-based successive approximation register ADC. Firstly, the more significant bits are determined in one clock cycle by the flash analog-to-digital converter. This speeds up the conversion and ensures an accurate determination of the more significant bits. Thereafter, the less significant bit decisions are made using the SAR ADC. This saves energy, since it prevents that a large amount of reference voltage levels for the flash ADC needs to be generated. Hence, the overall performance of a hybrid analog-to-digital converter

presents a compromise between the properties of its sub-ADCs.

2.2.8. Comparison

Concluding, we see that every architecture presents an individual trade-off between design aspects. The most important findings are summarized in table 2.1. Counter and integrating ADCs represent simple architectures, that can be implemented without much expense in systems demanding only a low sampling rate. Contrarily, flash ADCs are optimized for a short conversion time. However, achieving this high sampling frequency comes at the cost of a limited resolution and, moreover, a large demand for area and energy. If a high sampling frequency, as well as a high resolution is required, pipelined ADCs will pose the best solution. Nevertheless, also their area and power demand are significant. The solution for low-power applications is the SAR ADC. It settles in the medium region of both sampling frequency and resolution, and can be applied in a small area.

Of course, it is possible to optimize every architecture with a certain focus. This way, implementations can be achieved that exceed the ranges stated here. Nevertheless, this comparison provides an impression of typical specifications for the different types of ADCs. Thus, it is possible to evaluate the suitability of various architectures given the constraints of an application.

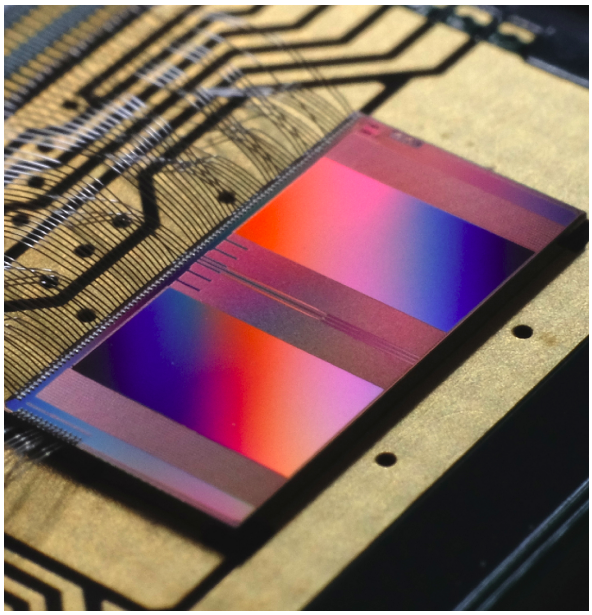
architecture	max. resolution [bits]	sampling frequency [MS/s]	power	area
counter ADCs	medium 10-12	low $10^0 - 10^3$	medium-high	low
integrating ADCs	medium-high 12-18	low $10^2 - 10^3$	low	low
flash ADCs	low 6-8	high $10^9 - 10^{10}$	high	high
SAR ADCs	medium-high 12-18	medium $10^5 - 10^7$	low-ultralow	low
pipelined ADCs	medium-high 12-18	medium-high $10^7 - 10^8$	high	high

Table 2.1.: Comparison of different ADC architectures. Values taken from Bashir et al. [2016].

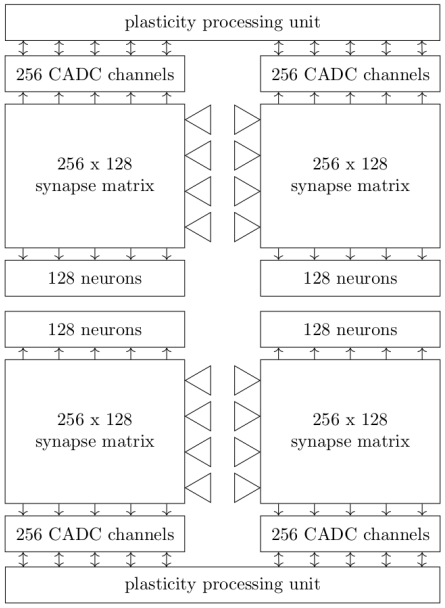
2.3. BrainScaleS-2 neuromorphic hardware

The goal of this work is to design an analog-to-digital converter for the BrainScaleS-2 neuromorphic chip (Schemmel et al. [2020], Pehle et al. [2022]). Neuromorphic hardware presents an approach to overcome the limitations of traditional von-Neumann computing architectures that is inspired by the information processing of the nervous system. In living creatures information are mainly transmitted by the means of spikes, which are discrete events in time characterized by a short rapid increase of the membrane potential of a neural axon (Gerstner et al. [2014]). But, also the membrane potential's development below the spike threshold can be of interest for certain information encodings or to analyze learning rules (Zenke and Ganguli [2018], Cramer et al. [2022]). This neural behavior can be modeled and simulated, e.g. using NEURON (Hines and Carnevale [2006]), NEST (Gewaltig and Diesmann [2007]) or Brian (Stimberg et al. [2019]). However, in a human brain the number of neurons is in the order of 10^{11}, each having up to 10^4 interconnections via synapses (Zhang [2019]). Wanting to simulate such a vast network exceeds the capabilities of current computers not only in terms of storage, but more importantly in terms of time and energy (Jordan et al. [2018], Cremonesi and Schürmann [2020]). Therefore, an approach to get around solving numerous coupled differential equations is emulating them in electrical circuits. Given the capabilities of today's semiconductor technologies, this does not only provide a tremendous energy saving, but even speeds up the processing compared to biological time (Schmitt et al. [2017]). The BrainScaleS-2 65 nm chip has been developed at the Kirchhoff Institute for Physics in Heidelberg within the efforts of the European Human Brain Project. Besides biological modeling (Billaudelle et al. [2022], Göltz et al. [2021]), it can also be used as an analog vector-matrix multiplier for neural network applications in machine learning (Stradmann et al. [2022], Weis et al. [2020]). In this case, the membrane potentials represent the multiplication results. So, either way it is important to read out the analog membrane voltages and digitize them for analysis or further processing.

The analog neuromorphic core of the BrainScaleS-2 chip is divided into four quadrants, as depicted in figure 2.10. Each of them contains 128 neuron circuits that receive input from 256 synapse rows. In the spiking case, synapses generate a current that is scaled by their configured weight for a fixed amount of time. This charges a capacitor representing the neuron's membrane. In the vector-matrix multiplication case, the input events are no longer binary, but carry an activation value. Therefore, the time during which synapses are generating currents is modulated row-wise by the input activation. The input is sent by the synapse drivers to all synapses in a row, that



(a) Photograph of the BrainScaleS-2 chip. Picture taken by Eric Müller.



(b) Block overview of the BrainScaleS-2 chip.

Figure 2.10.: The BrainScaleS-2 neuromorphic chip contains 512 neuron circuits, emulating the behavior of spiking cells.

individually apply their weights, thus, realizing the vector-matrix multiplication. The result, then, is represented by the membrane voltages. For reading them out there exist two possibilities: Either a single voltage trace is digitized with a 10-bit resolution and high sampling frequency, using the membrane-analog-to-digital converter (MADC). Or, multiples of them are read out in parallel, however with decreased resolution and speed, using the column-analog-to-digital converter (CADC). Sometimes, the CADC is also referred to as correlation-analog-to-digital converter, since it can also perform the conversion of the synaptic correlation measurements. They are relevant for numerous learning rules, e.g. spike-timing-dependent plasticity (Petrovici [2016]). From the digitization on, the results can be processed directly on chip by the plasticity processing units or are read out externally.

2.4. Design targets

The goal of this work is to redesign the CADC, such that it is possible to read out all neurons in parallel with a sampling rate that is orders of magnitude higher. Therefore, let's take a look at how the currently existing analog-to-digital converters on BrainScaleS-2 are implemented. On the one hand, the membrane-analog-to-digital

	MADC	CADC	future ADC	
			fast mode	precise mode
resolution [bits]	10	8	6	8
max. sampling frequency [MS/s]	65	1.85	125	62.5
input range [V]	0.2 - 1	0 - 1.2	0.6 - 1.2	0 - 1.2
power per channel [µW]	2500	13	195	195
energy per conversion [pJ]	38	7.3	1.56	3.12
area per channel [µm^2]	50 000	269	1176	1176

Table 2.2.: Specifications of the analog-to-digital converters currently implemented, as well as targets for the future ADC. The values of the CADC are stated per channel. Values taken from Microelectronic Systems Laboratory [2015] and Schreiber [2021].

converter realizes the successive approximation register principle, enabling the relatively high sampling frequency (Microelectronic Systems Laboratory [2015]). On the other hand, the column-analog-to-digital converter is made up of two single-slope integrating channels per column, i.e. 1024 channels in total (Schreiber [2021]). This is the case, so they can convert the causal and acausal correlation measurements in parallel. For the single slope ADCs the reference voltage ramp is generated only once per quadrant, providing a low energy consumption. The specifications of both ADCs are given in table 2.2.

Also, the targets for our new ADC are listed there. First of all, we would like to achieve an increased sampling rate of approximately 125 MS/s. This will enable achieving a better performance in applications, using the fast vector-matrix multiplication mode. But also in biologically inspired experiments, a faster CADC can provide advantages, e.g. reducing the time for training neural networks (Cramer et al. [2022]). However, achieving such a high sampling frequency at an 8-bit precision is difficult. Moreover, in some cases where an increased sampling rate is desired, a resolution of 8-bit is not necessary. For example, in the application as analog vector-matrix multiplier the entries of the input vector can only be inserted as 5-bit values. Hence, using the output of a calculation as input for a subsequent one, it is not necessary to determine the result with a higher resolution. Nevertheless, we'd also like to have the possibility of reading out multiple neurons in parallel with the full CADC resolution of 8-bit. Thus, we desire two modes of operation in the new column-analog-to-digital converter: a fast mode with reduced resolution and a precise mode that may convert slower. For the fast mode, being mainly used for vector-matrix multiplications, 6-bit are an adequate resolution. Furthermore, an input range covering only the upper half of the full scale spectrum is sufficient. For the precise mode our aim is to keep an

unreduced resolution of 8-bit. Here, however, the maximum sampling rate is not of significant importance. Still, we strive for an enhancement of one order of magnitude. As a target we try achieving half of the speed of the fast mode.

Since in both cases we plan on being significantly faster than the currently implemented CADC, in the future it will be sufficient to have only one channel per neuron, that can perform the conversion of the causal and acausal correlation subsequently. Hence, we can use the width of two single-slope channels, namely 11.76 µm. Moreover, to implement the two modes of operation, we'll try to stay within a height of 100 µm.

Concerning the power consumption, we make the following estimation: Desirably, the total chip power won't exceed one Watt. We endeavor the power consumption of all CADC channels in total to contribute by approximately 10%. Given 512 channels, this yields 195 µW per ADC. Considering the specified target frequencies, the aspired energies per conversion for the fast and precise mode are 1.56 pJ and 3.12 pJ, respectively.

Having said that, all of these estimations are not hard constraints, but serve as targets for orientation. This thesis is also about analyzing different trade-offs and discussing, what can be achieved at which cost.

Having specified the targets of the analog-to-digital converter for our application, the first step is choosing an appropriate architecture. Since we want to achieve a high sampling frequency at moderate resolution, but at the same time keep the energy consumption low, we conclude from section 2.2.8 that the successive approximation register ADC constitutes the best choice for us. Hence, in the following we will focus on how to implement a successive approximation register analog-to-digital converter that fulfills the targets of our application best. Therefore, firstly we are gonna take a look at how to optimally realize the digital-to-analog conversion part of the SAR ADC. Then, different approaches for putting the two desired modes of operation into practice are discussed, before describing and evaluating the final implementation.

3. Advanced DAC design for an SAR ADC

Just like for analog-to-digital converters, there are numerous ways of implementing digital-to-analog converters (Pelgrom [2017], Carusone et al. [2013]). One approach is digital-to-analog conversion in the voltage domain. This usually utilizes resistors that are arranged e.g. along a string or in a binary fashion as R-2R ladder. Another way to convert digital signals to analog voltages is in the charge domain. Here, switched capacitor circuits come into play. Moreover, DAC designs were introduced that function in the current or the time domain. In principle, successive approximation register ADCs can be build with almost any type of digital-to-analog converter.

When SAR ADCs were introduced, firstly for the most part resistive structures were used (Hamade [1978]). However, a shortcoming of resistors is that energy is transformed into heat. On the one hand, this means that this energy can't be reused, unlike in the case of capacitors, where energy is stored in an electric field. On the other hand, a change of temperature influences electrical components, like the resistivity itself and leads to mismatch. Thus, resistors must be handled with care in CMOS technology. Nevertheless, nowadays they start to regain more interest for high-frequency designs, where DACs may be used by multiple converters in parallel (Wei et al. [2012]).

Most attention for SAR ADCs in CMOS technology, however, is focused on capacitive DAC structures. They take advantage of the strengths of this technology, good switches and capacitors, as well as their low-voltage properties, which results in a low energy consumption. One of the early fully integrated CMOS successive approximation register ADCs applies a principle called charge redistribution (McCreary and Gray [1975]). It is still commonly used nowadays, due to its simplicity and reliability.

An alternative is offered by the charge sharing approach (Craninckx and van der Plas [2007]). By pre-charging a single or multiple reference capacitors, interfacing a possibly noisy supply during the conversion is avoided. Furthermore, charging energy is only consumed in the sampling period, since afterwards the capacitors interact exclusively by passive charge sharing. This allows to reduce the energy consumption in certain

applications. However, the reference capacitor needs to be chosen very large, in order to ensure a stable supply. Moreover, in simulation the effect of component mismatch and parasitic capacitances was found to be very severe, which will cause malfunctions in physical implementations.

Thus, in the following we will focus on charge redistribution designs.

3.1. Charge redistribution principle

Charge redistribution is the principle that is most commonly used for digital-to-analog converters in SAR ADCs. It contains a binary-weighted capacitor array, where all top terminals are connected to a comparator on a shared line V_{comp}. The output of the comparator is connected to a control logic which sets the potentials at the bottom terminals of the capacitors, as well as to the digital output register. Figure 3.1 visualizes this configuration for an n-bit ADC.

The functioning, now, is as follows: Firstly, the input is sampled. For this, the same capacitors are used as for the conversion. In this example, the input voltage is connected to the top terminals of the capacitors, such that $V_{\text{comp}} = V_{\text{in}}$ (see figure 3.2). During this time, all bottom terminals are connected to ground. Then, to test the MSB, the bottom terminal of the largest capacitor is switched to the reference voltage. This results in a rise of V_{comp} by the amount of capacitance connected to V_{ref} compared to the total capacitance in units of the reference voltage. Due to a dummy capacitor,

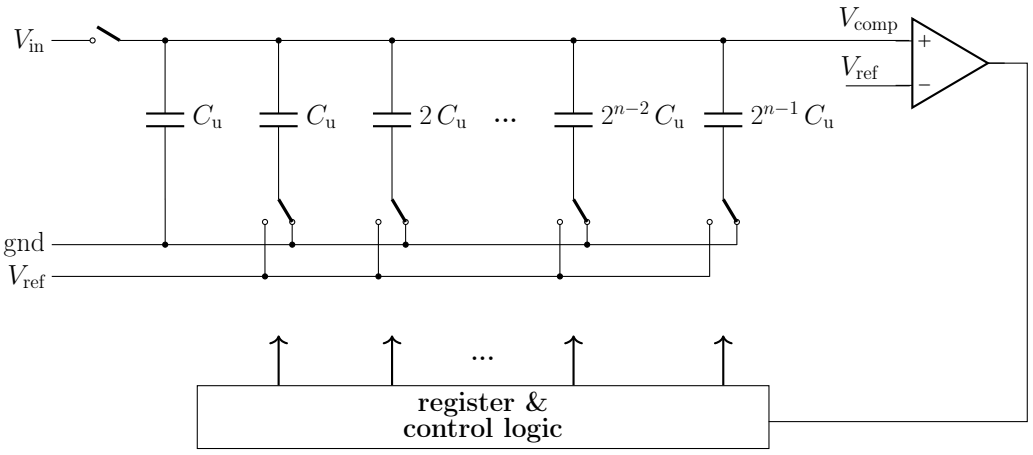
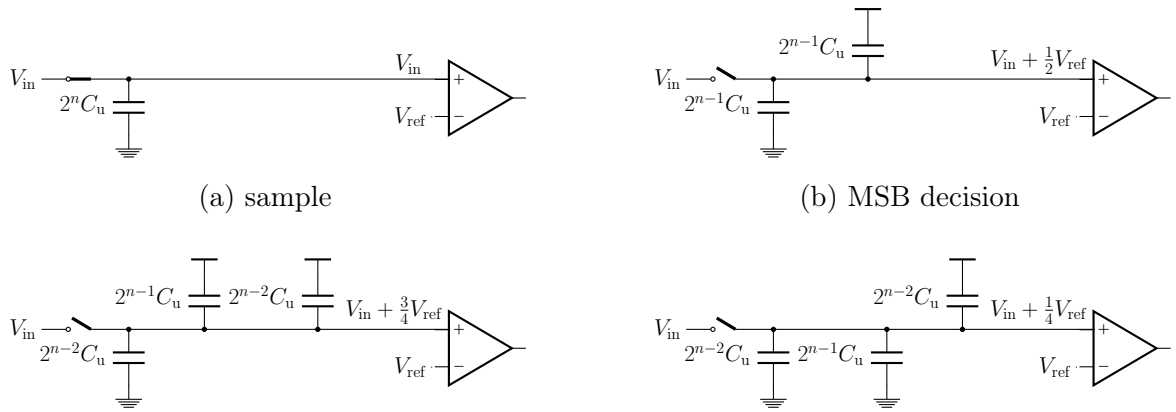


Figure 3.1.: Circuit of an SAR ADC with a capacitive charge redistribution DAC. The unknown input voltage is compared to a reference and depending on the comparator's decision, the binary-weighted capacitors are configured to compensate the input. Thus, the DAC configuration equals the register holding the digital output code.



(a) sample (b) MSB decision

(c) second bit decision (MSB = 0) (d) second bit decision (MSB = 1)

Figure 3.2.: Conversion principle of the standard charge redistribution SAR ADC. (a) The input is sampled with all capacitors connected to ground. (b) To make the MSB decision, the largest capacitor is switched to the reference voltage. This voltage division lifts the input of the comparator by $\frac{1}{2}V_{\text{ref}}$, causing an effective comparison between the input and half the reference voltage. (c) If the MSB = 0, the subsequent capacitor will be switched to V_{ref}, too, for making the second bit decision. (d) If the MSB = 1, in addition to switching the second largest capacitor to the reference, the MSB capacitor will be reconnected to ground. This principle is repeated for all bits.

duplicating the unit capacitance C_{u} of the smallest binary-weighted capacitor, this fraction shortens to $\frac{1}{2}V_{\text{ref}}$ in the ideal case, where no parasitic capacitances are present.

$$V_{\text{comp}} = V_{\text{in}} + \frac{2^{n-1}\, C_{\text{u}}}{(\sum_{i=0}^{n-1} 2^i\, C_{\text{u}}) + C_{\text{u}}} V_{\text{ref}} = V_{\text{in}} + \frac{1}{2} V_{\text{ref}} \tag{3.1}$$

Applying the reference voltage at the other side of the comparator, this is equivalent to comparing V_{in} to $\frac{1}{2}V_{\text{ref}}$. Hence, the output of the comparator depicts the MSB result. If it is zero, the input voltage is lower than $\frac{1}{2}V_{\text{ref}}$ and V_{comp} needs to be increased, in order to reach the potential at the other side of the comparator. Thus, the switch of the largest capacitor remains in its position and the bottom terminal of the second largest capacitor is connected to V_{ref}, too. Again, V_{comp} settles at the level determined by the voltage division between the capacitance connected to V_{ref} and the overall capacitance. In this case, we have

$$V_{\text{comp}} = V_{\text{in}} + \frac{2^{n-1}\, C_{\text{u}} + 2^{n-2}\, C_{\text{u}}}{(\sum_{i=0}^{n-1} 2^i\ C_{\text{u}}) + C_{\text{u}}} V_{\text{ref}} = V_{\text{in}} + \frac{3}{4} V_{\text{ref}}. \tag{3.2}$$

Consequently, the effective decision of the comparator is between V_{in} and $\frac{1}{4}V_{\text{ref}}$.

If the MSB decision is one, however, the input voltage is higher than $\frac{1}{2}V_{\text{ref}}$ and V_{comp} needs to be decreased. Hence, the switch of the first capacitor is reconnected to

ground, lowering V_{comp} to V_{in} again. Thereafter, to test the second bit, the switch of the associated capacitor is connected to V_{ref}. This lifts V_{comp} by the newly configured voltage division:

$$V_{\text{comp}} = V_{\text{in}} + \frac{2^{n-2}\,C_{\text{u}}}{(\sum_{i=0}^{n-1} 2^i\,C_{\text{u}}) + C_{\text{u}}} V_{\text{ref}} = V_{\text{in}} + \frac{1}{4} V_{\text{ref}}. \tag{3.3}$$

Comparing this to V_{ref}, now, is equivalent to comparing V_{in} to $\frac{3}{4}V_{\text{ref}}$.

This procedure is continued for all subsequent capacitors: If a bit decision is one, the bottom terminal of the considered capacitor will be reconnected to ground. Otherwise, it will stay in place. Then, regardless of the previous decision, the next capacitor will be switched to V_{ref}, preparing the determination of the following bit.

Of course, there are also other configurations how this procedure can be applied. For example, instead of initially connecting all bottom terminals to ground, they could also be connected to V_{ref}. In this case, switching capacitors to ground, will lower to potential of V_{comp}. Thus, the voltage at the other input of the comparator needs to be changed to the ground potential. Then, the procedure can be applied analogously to the explanation above. However, in this case the comparator needs to deal with voltages of different signs, which is not a trivial task. Since this configurations doesn't present any notable advantages over the former, we won't pursue it further.

Another variation concerns the side at which the input is sampled. Instead of connecting V_{in} to the top terminals of the capacitors, it can also be applied at the bottom terminals, as depicted in figure 3.3. During the sampling phase, then, the non-inverting input of the comparator needs to be held at a well-defined potential. Thereafter, the top terminals are left floating and the bottom terminals are connected to ground. Since the charge on the capacitors is conserved, this shifts V_{comp} by $-V_{\text{in}}$. In the conversion phase, then, the capacitors are configured, such that they optimally compensate this shift. In order to do so, the common mode voltage V_{cm}, that is connected to the inverting input of the comparator, must be equal to the potential, that was applied at the top terminals during the sampling phase. Generating this additional voltage may cost power in high performance designs. But, it also allows to set the optimal input voltage for the comparator.

Both sides are legitimate choices for sampling, however, they come with different consequences. But, to properly discuss them, it is useful to take a look at the effect of parasitic capacitances first.

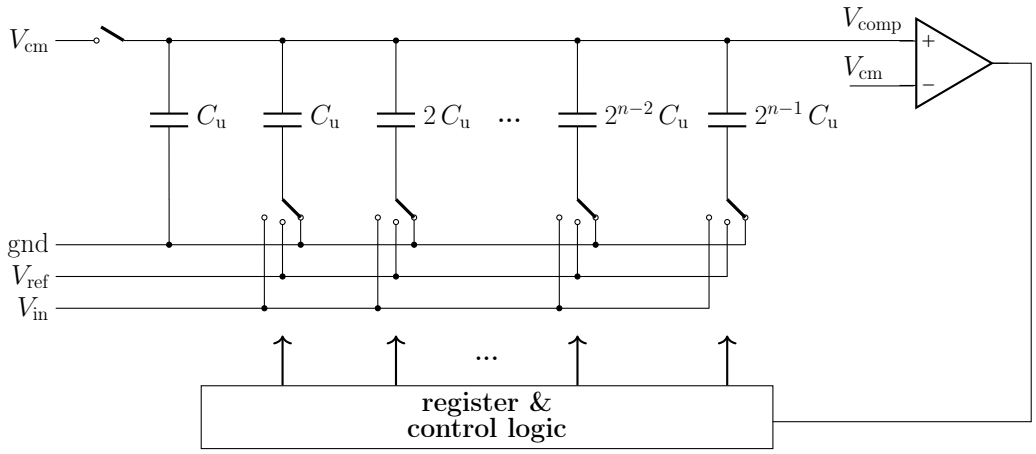


Figure 3.3.: Circuit of an SAR ADC with a capacitive charge redistribution DAC. The input voltage can also be applied at the bottom terminals. Then, however, an additional common mode voltage V_{cm} is required.

3.2. Parasitic capacitance

All those capacitances that are present in a physical implementation, however not by design, are called parasitic capacitances. They always occur where conductors with different potentials face each other e.g. wires running closely in parallel or near over the substrate. In modern technologies with shrinking feature size, the height of wires has exceeded its width, causing the coupling capacitances to components in the same layer to be larger than to those in adjacent layers (Pelgrom [2017]). This can be taken advantage of for intentionally designing capacitors, as we will discuss in chapter 5.2.2. However, closely routed wires are also the main cause for unintentional capacitive coupling. Since these capacitors are introduced by the layout, but are not meant to be there by design, they are called parasitic capacitors. The severity of their effect differs depending on the application. However, in our case the conversion is based on a precise voltage division. Hence, it is crucial to analyze the consequences of parasitic capacitances not only after the layout is finished, but also to take them into consideration during the design process. This allows to identify sensitive parts and to rule out unreliable designs early on.

In capacitor circuits a simple way to model parasitic capacitances is using the pi-model (Carusone et al. [2013]). Here, simply a capacitor is added to both the top and the bottom terminal with a parasitic capacitance coupled to ground. Depending on the way the intentional capacitor is implemented, those two parasitic capacitances may vary. In order to be more accurate, capacitances to other potentials can be taken into account, too. But, in many cases an accurate modeling early on in the design process is difficult, because before implementing the layout, usually it is not known

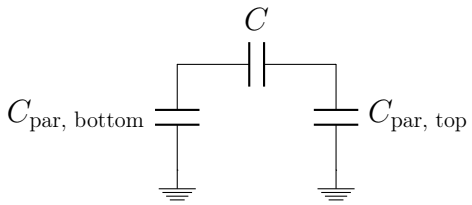


Figure 3.4.: Pi-model for parasitic capacitances of a capacitor. Both terminals couple to the substrate ground. However, depending on the implementation of the capacitor, the strength of this coupling can differ between the terminals.

where individual nets and components will be located. So, the pi-model is a useful approach for getting a first estimation of the effect of parasitic capacitances. Later when the layout is finished, the actual parasitic capacitances between all nets can be extracted and utilized for a placement-aware simulation of the design.

For now, applying the pi-model to all capacitors of the charge redistribution SAR ADC as depicted in figure 3.1, we can perform some simplifications. On the one hand, all parasitic capacitors on the top side are operated in parallel. Hence, we can combine them to a single capacitor holding the sum of all capacitances. On the other hand, the bottom terminals are connected to either ground or V_{ref}. Since these are static potentials, they don't influence the conversion and the according parasitic capacitors can be neglected. Thus, we end up with a parasitic capacitance C_{p} in parallel to all intentional capacitances, as depicted in figure 3.5.

Now, taking a look at what happens, when a capacitance $2^k\,C_{\mathrm{u}}$ is switched, we'll find a voltage change at V_{comp} of

$$\Delta V_{\mathrm{comp}} = \frac{2^k\,C_{\mathrm{u}}}{\left(\sum_{i=0}^{n-1} 2^i\,C_{\mathrm{u}}\right) + C_{\mathrm{u}} + C_{\mathrm{p}}} V_{\mathrm{ref}}. \tag{3.4}$$

The first thing to notice is that independent of the value of the parasitic capacitance and the probed bit, i.e. independent of k, all bits are influenced equally. In other words, this single added parasitic capacitor does not cause any non-linearities. The second thing to notice is that with the additional capacitor C_{p} to ground the denominator, now, is slightly larger. Thus, the amount of capacitance connected to ground is always larger than intended by design for the voltage division. This causes an attenuation at the top terminals, which is reflected in an offset and gain error. Concluding, parasitic capacitances to the ground potential narrow the conversion range depending on their value. But, since the linearity is not affected, they are fairly manageable.

Let's take a look back at the question, how the conversion is influenced by the choice of side that is used to sample the input. The discussed reduction of the conversion range

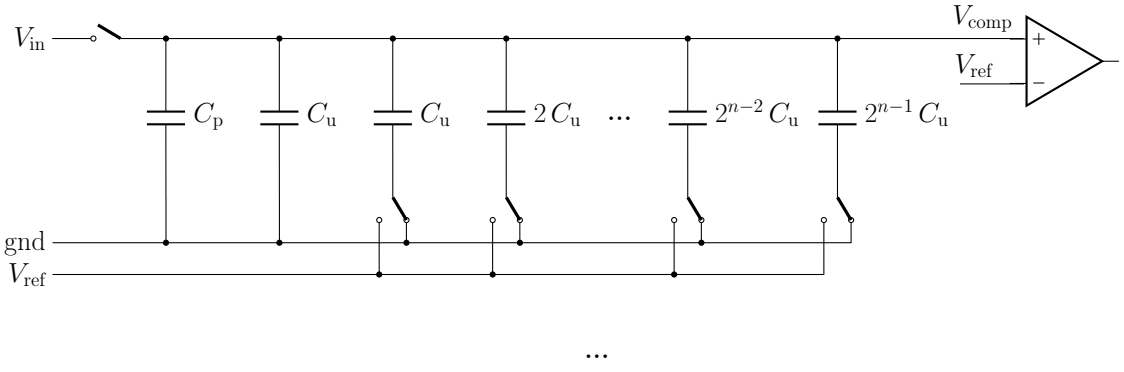


Figure 3.5.: Circuit of a charge-redistribution SAR ADC with parasitic capacitance. The coupling of the bottom terminals can be neglected, since it only affects static potentials. At the top terminals, all parasitic capacitors are arranged in parallel, such that they can be absorbed into a single capacitor C_{p}.

only occurs, if the signal is sampled at the top terminals, as depicted in figure 3.1. The reason is that V_{in} won't undergo the same attenuation as the reference voltage, which is applied at the bottom terminals. If the signal is sampled at the bottom terminals as shown in figure 3.3, however, it will take the same path as the reference voltage, canceling this effect.

Nevertheless, for our implementation we chose to sample the input at the top terminals. On the one hand, the reduced conversion range is not critical for our application. On the other hand, choosing the appropriate voltage at the inverting input of the comparator allows to determine the MSB without prior switching, if the input voltage directly is sampled directly at the top terminals. This is an attractive feature which will be further discussed in section 4.2.

3.3. Unit capacitance

So far, we have discussed that the digital-to-analog converter of our SAR ADC can be constructed using a binary-weighted array of capacitors. Now, let's have a look at what must be taken into consideration deciding on its smallest value, the unit capacitance. Intuitively, we want to choose C_{u} as small as possible. This is, because the energy scales linearly with the capacitance. For a conversion as explained above, all capacitors, except for the dummy capacitor, will be switched to V_{ref} and back to ground, before the next input is sampled. Hence, the charging of the capacitive array alone will result in an energy consumption of

$$E_{\text{conv}} = \frac{1}{2} \cdot (2^n - 1) C_{\text{u}} \cdot V_{\text{ref}}^2. \tag{3.5}$$

Consequently, the smaller we choose C_u, the less energy will be consumed.

Furthermore, also the area scales with the value of the capacitance. Even though in CMOS technology there exist different ways of how capacitors can be implemented, almost in every case a larger capacitance comes with an increased area demand. Moreover, in designs requiring precise matching, often larger capacitors are made up by multiple unit capacitors in parallel. This means that the total area scales directly with the unit capacitance. So, for an energy- and area-efficient design, we want to have a unit capacitance as small as possible.

However, if we choose a value that is too low, not only the effect of parasitic capacitances aggravates, but also intrinsic noise could become prevalent over the signal, corrupting the result. Besides the quantization noise discussed in section 2.1.1, we also need to take into consideration the Johnson–Nyquist noise (Allen and Holberg [2002]). It describes the thermal motion of charge carriers in electric conductors, that is always present at temperatures higher $T = 0\,\mathrm{K}$. Therefore, it is also called thermal noise. Ideal capacitors are lossless devices and do not carry thermal noise themselves. Nevertheless, in physical implementations they typically are connected to resistors, whose noise they accumulate. A resistor carries a root mean square noise voltage of

$$V_\mathrm{rms}^2 = 4k_\mathrm{B}TR\Delta f. \tag{3.6}$$

The noise bandwidth Δf can be calculated by integrating the absolute square of the transfer function of an RC low pass filter in the positive frequency domain.

$$\Delta f = \int_0^\infty |H(f)|^2 df = \int_0^\infty \frac{1}{1 + (RC)^2(2\pi f)^2} df = \frac{1}{4RC} \tag{3.7}$$

Inserting this into equation 3.6 yields an accumulated noise on the capacitor of

$$V_\mathrm{rms}^2 = \frac{4k_\mathrm{B}TR}{4RC} = \frac{k_\mathrm{B}T}{C} \tag{3.8}$$

We see that the Johnson–Nyquist noise is independent of the resistance value R. This is due to the fact, that even though small resistances have less spectral noise density, their noise bandwidth scales inversely proportional. However, we find that with rising temperature fluctuations increase. But, their effect can be mitigated by increasing the capacitance. Hence, in order for the $\frac{k_\mathrm{B}T}{C}$-noise not to influence the conversion, we must choose the unit capacitance large enough, such that $\sqrt{V_\mathrm{rms}^2}$ is smaller than $\frac{1}{2}V_\mathrm{LSB}$. In the ideal case, where the full conversion range is made use of, V_LSB is given by equation 2.1. If we consider the thermal noise of the total capacitance of an n-bit array, this

will result in the condition

$$\sqrt{V_{\text{rms}}^2} = \sqrt{\frac{k_{\text{B}}T}{2^n C_{\text{u}}}} \overset{!}{<} \frac{1}{2} \cdot \frac{V_{\text{ref}}}{2^n}. \tag{3.9}$$

Solving for the unit capacitance, we get

$$C_{\text{u}} \overset{!}{>} \frac{2^{n+2} k_{\text{B}} T}{V_{\text{ref}}^2}. \tag{3.10}$$

Considering the 8-bit case with $V_{\text{ref}} = 1.2\,\text{V}$ and a maximum operation temperature of 70°C, the unit capacitance must be at least $3{\cdot}10^{-18}\,\text{F}$. However, these small capacitances are usually not implementable. On the one hand, it is difficult to avoid parasitic capacitances in this range, that wouldn't significantly intrude the conversion. On the other hand, components can only be manufactured reasonable precisely down to a certain dimension. There are physical limitations in the production process, allowing to achieve only a certain feature size and accuracy.

In the 65 nm process used by us, for example the smallest metal-oxide-metal capacitor provided by the manufacturer has a capacitance of $2.4 \cdot 10^{-15}\,\text{F}$. Beyond that, it is also possible to design smaller capacitors oneself. However, then it can't be guaranteed that they are produced without considerable mismatch. Hence, if we stick to the smallest capacitor of the manufacturer, we won't have to worry about $\frac{k_{\text{B}}T}{C}$-noise.

3.4. Performance analysis

After having discussed the principle of charge redistribution and important aspects, that should be considered for its implementation, let's analyze its performance in simulation. To do so, Dauer [2022] developed a framework that allows to efficiently evaluate different analog-to-digital converter designs. In the beginning, an input voltage is generated that increases gradually. The resolution of these voltage steps can be configured, as well as their start and stop value. Then, after the ADC design under test has performed the conversion, the output code is stored. On the one hand, this way the transfer function can be analyzed visually. On the other hand, the output codes are used to automatically calculate properties like the differential and integral non-linearity, as well as the offset and gain error. This framework is used for the analysis of ADCs throughout this thesis.

Firstly, we consider the charge redistribution design as shown in figure 3.1 for the ideal 6-bit case without parasitic capacitance nor mismatch. The reference voltage

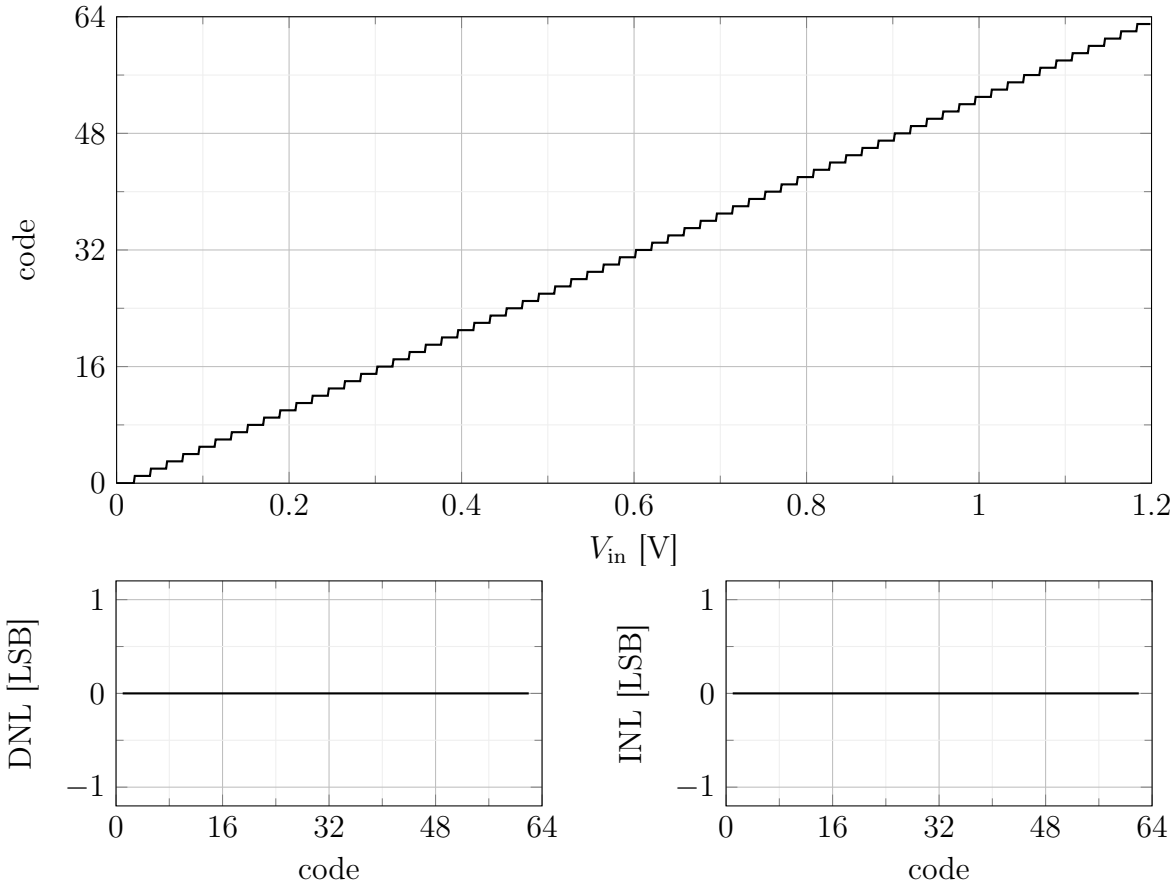


Figure 3.6.: Charge redistribution ADC characteristics for the ideal case. There is no offset and gain error and all code steps are equally wide, resulting in a vanishing DNL and INL.

equals the supply voltage of 1.2 V and the unit capacitance is chosen as $2.4 \cdot 10^{-15}$ F. Having said that, all components are simulated ideally for now. Furthermore, the input voltage is generated with a step size six times higher than the resolution of the ADC. Figure 3.6 shows its perfect transfer function with equally sized steps, which is also reflected in the vanishing non-linearities. This is exactly the behavior we expect for an ideal ADC.

Next, we consider each capacitor to have a parasitic capacitance to ground amounting to 20% of its nominal value. This estimation is obtained from the parasitic extraction of a metal-oxide-metal capacitor with grounded metal shieldings on its adjacent layers. Different ways of implementing capacitors in CMOS technology are discussed in more detail in section 5.2.2. The configuration containing additional parasitic capacitors can be simplified to the case depicted in figure 3.5. Given a total array capacitance of approximately $C_{\text{tot}} = 150 \cdot 10^{-15}$ F, this yields $C_{\text{p}} = 30 \cdot 10^{-15}$ F. As discussed, this effects that switching a capacitance $2^k\, C_{\text{u}}$ results in a voltage division of $\frac{2^k\, C_{\text{u}}}{C_{\text{tot}}+C_{\text{p}}}$, which

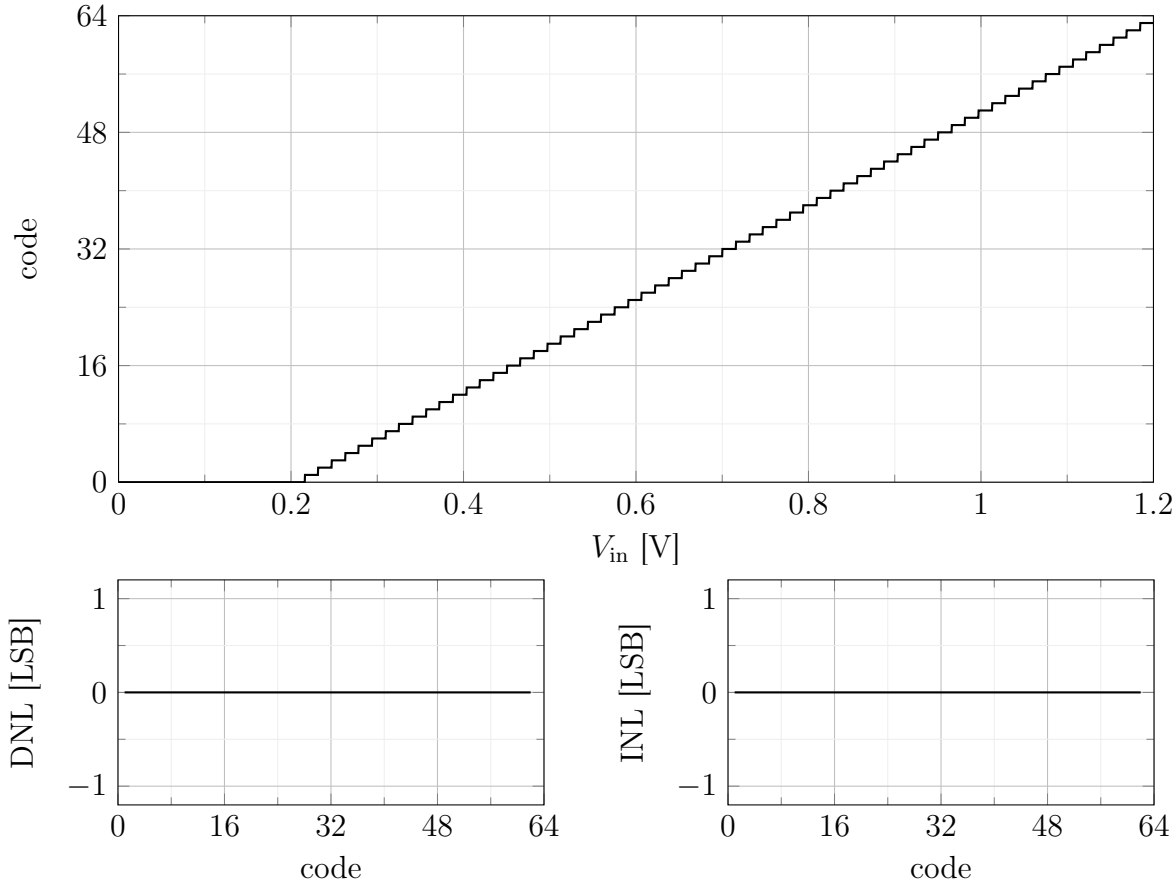


Figure 3.7.: Charge redistribution ADC characteristics for the case with parasitic capacitance of 20%. The additional capacitance causes an uneven voltage division, leading to an offset error. However, the linearity is not affected.

is smaller than the ratio intended by design. Hence, V_{comp} is increased less, such that it only exceeds the reference voltage at the inverting input of the comparator at higher input voltages compared to the case without parasitic capacitances. This causes a significant offset of approximately 13 LSB, which is visible in figure 3.7. Since all output codes can be mapped nonetheless, also a gain error is present. The linearity, however, is not affected.

Lastly, we want to analyze the effect of mismatch between the capacitors. For this, we neglect the parasitic capacitance again, but draw the capacitance values of the binary-weighed array from Gaussian distributions, ranging around their nominal values with a standard deviation of $\sigma = 0.05$. This value is chosen according to TSMC [2011]. The effect of this mismatch is most severe at codes, where a single capacitor ought to compensate many smaller capacitors. In figure 3.8 this can be observed best at the MSB transition. The according step is significantly wider, indicating that the MSB capacitance is larger than the sum of all other capacitances. In contrast, the second

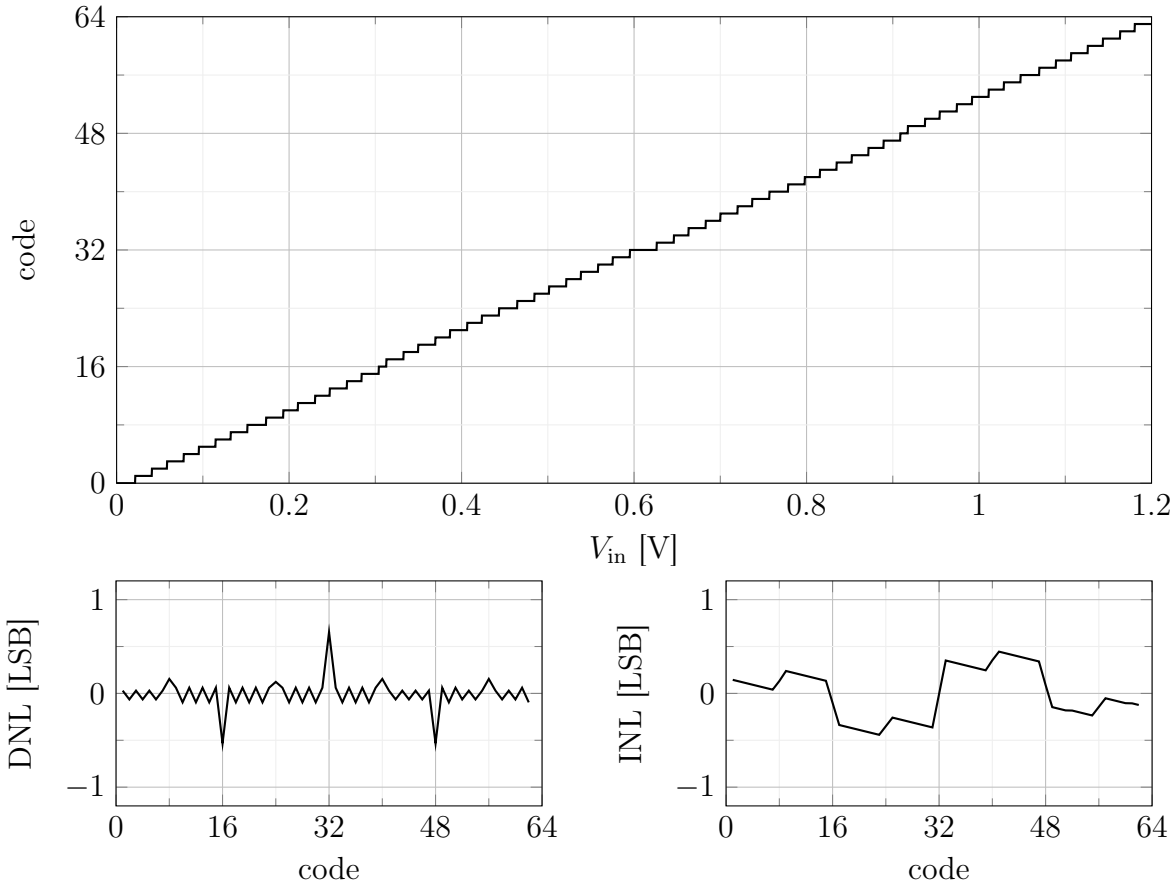


Figure 3.8.: Charge redistribution ADC characteristics for the case with $\sigma = 0.05$ mismatch. At the large bit transitions the mismatch causes errors in the DNL, leading to an INL divided into separate blocks.

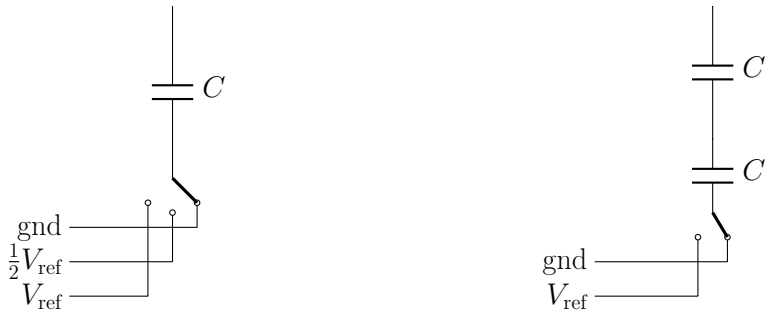
largest capacitor is smaller than the accumulation of its following bit capacitors, as the negative differential-non-linearity displays. The blocks in the transfer curve separated by the DNL errors are also reflected in the integral non-linearity. Here, the effect of mismatch scales with the capacitance value. Moreover, the course of the mismatch, i.e. the sign of the DNL, is random, because the deviations may happen likewise: Either a capacitance is larger or smaller than its nominal value. Thus, the same configuration e.g. could show a negative DNL at the MSB transition, if the mismatch was drawn such that the MSB capacitor was smaller than the sum of all other capacitors. Consequently, in order to quantitatively analyze the effect of mismatch, it would be necessary to repeat this simulation multiple times. However, since we only want to obtain a general impression of the effect of mismatch, it is sufficient to get a qualitative overview.

Concluding, we find that, on the one hand, parasitic capacitances should be avoided as well as possible, since they significantly limit the conversion range by introducing

offset and gain errors. On the other hand, a precise capacitor matching is necessary, otherwise the linearity of the conversion will be compromised.

3.5. Modifications of the DAC array

We have seen, that energy and area scale with the unit capacitance, which is why we want to make it as small as possible. If we configure it too low, however, the effect of noise and other distortions will be aggravated, leading to errors in the ADC characteristic as we have just seen. Even choosing an acceptable unit capacitance, the total capacitance will double with every bit the resolution is increased by in a binary-weighted array. Hence, another approach for reducing area and energy is to adjust the array using a suitable scaling of voltages. For this, either the reference voltage at the bottom terminals of the capacitors can be scaled, or the output voltage that is present at the top terminals, as depicted in figure 3.9 (Pelgrom [2017]). Analytically, both methods yield the same result. Since the generation of further voltages for low-impedance loads costs additional energy, in the following we will explore concepts that use appropriate voltage scaling at the top terminals to fulfill the same conversion as the binary-scaled array.



(a) Reference voltage scaling (b) Output voltage scaling

Figure 3.9.: Different approaches of voltage scaling. Either further reference levels at the bottom terminal can be added or the output voltage at the top terminal can be scaled.

3.5.1. Split least significant bit

In a binary-weighted n-bit array, the capacitance of the MSB is 2^{n-1} times the capacitance of the LSB. Thus, reducing the LSB capacitance by a factor of $\frac{1}{2}$ also decreases the MSB capacitance by the same amount, as well as the total capacitance of the array. This reduction can not only be achieved by shrinking the unit capacitance, but also by operating multiple capacitors in series. In this configuration, the effective capacitance

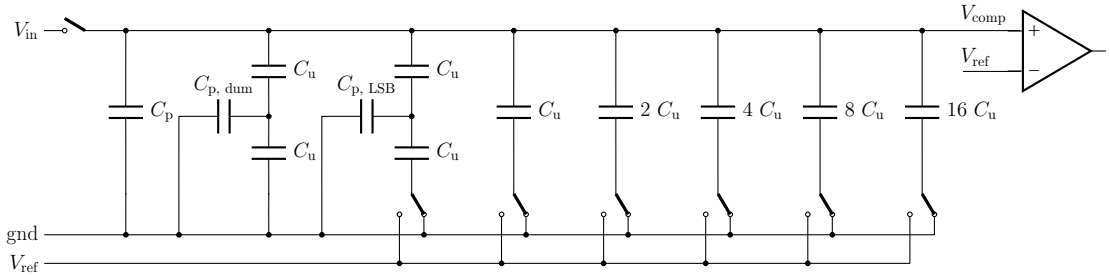


Figure 3.10.: Circuit of a 6-bit SAR ADC with split LSB and parasitic capacitance. By introducing the LSB split, additional nodes for parasitic capacitors $C_{\text{p, LSB}}$ and $C_{\text{p, dum}}$ are generated.

is given by the reciprocal sum of its components. Hence, putting two unit capacitors in series results in an LSB capacitance of $\frac{1}{2}\,C_{\text{u}}$. This way, C_{LSB} can be reduced without decreasing the unit capacitance. For the correct voltage division, the dummy capacitance has to be implemented in the same way.

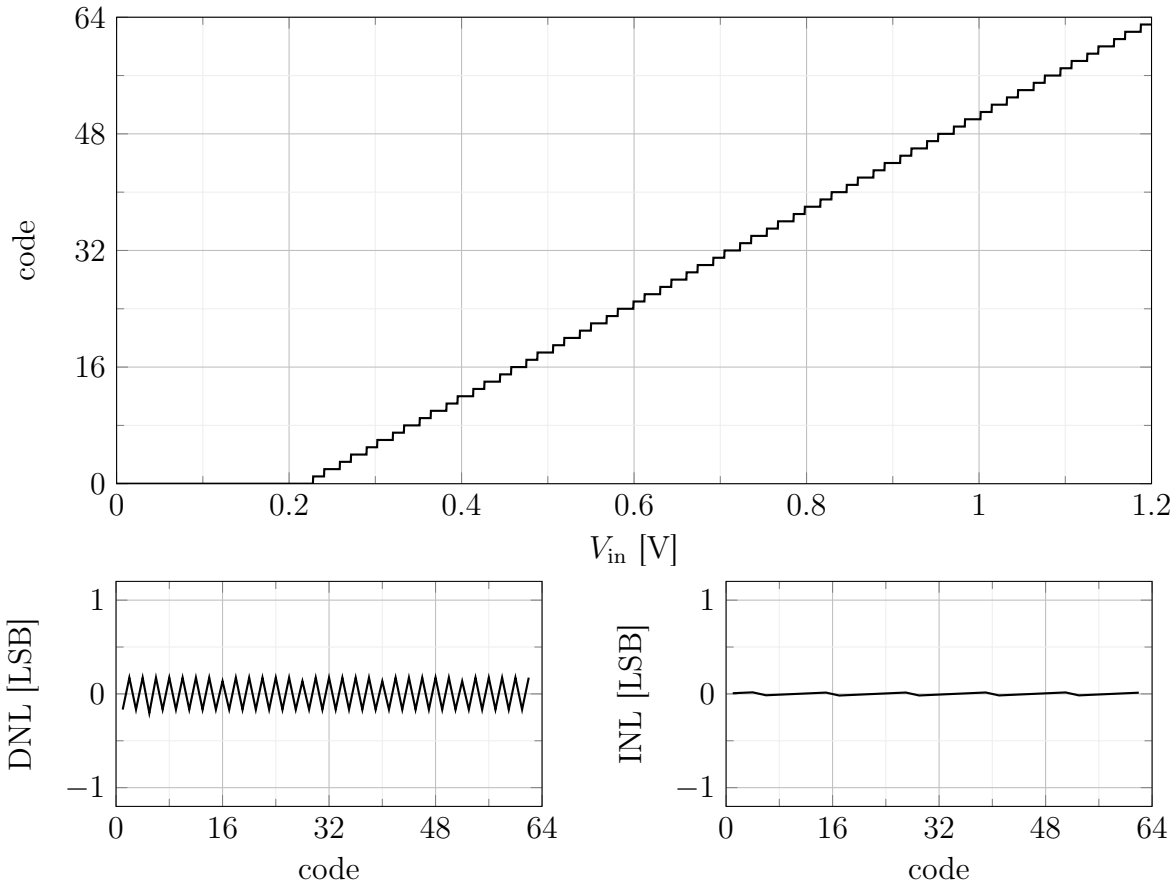


Figure 3.11.: Split LSB ADC characteristics for the case with parasitic capacitance of 20%. Besides the offset error, also the linearity is affected, since the parasitic capacitor between the LSB capacitors can't be absorbed into the joint parasitic capacitance C_{p}.

However, now we need to take a closer look at parasitic capacitances. Applying the pi-model now, also introduces parasitic capacitors between the unit capacitors arranged in series, as depicted in figure 3.10. These can no longer be absorbed into the single parasitic capacitance C_p and will introduce non-linearities in the conversion.

So, let's take a look at their severity for constructing the LSB from two unit capacitors in series. Initially, without any parasitic capacitances or mismatch, the behavior is equal to the case depicted in figure 3.6. This is, since both cases are analytically equal and no physical effects were considered. Also, the effect of mismatch is comparable to the situation with only a single LSB capacitor, using the same random seed for drawing the deviations. The reason for this is that only a fraction of the least significant capacitance changes. However, if we add a parasitic capacitance of again 20% the nominal value to each terminal, we will see the result displayed in figure 3.11. On the one hand, an offset error comparable to the case discussed before is present. On the other hand, the width of every second step varies slightly. This is due to the additional capacitance in between the two capacitors in series, which affects the LSB decision, i.e. every second step. Having said that, the linearity errors are still in a well acceptable range.

3.5.2. Linearly split two bits

As a next step, we linearly extend this procedure beyond the least significant bit. This means, we operate four unit capacitors in series for the LSB and two for the penultimate bit. Again, the ideal behavior is known to us. Adding parasitic capacitors in the same fashion as before introduces even more components which can't be substituted into a joint capacitor C_p, as shown in figure 3.12. These flaws can be clearly seen

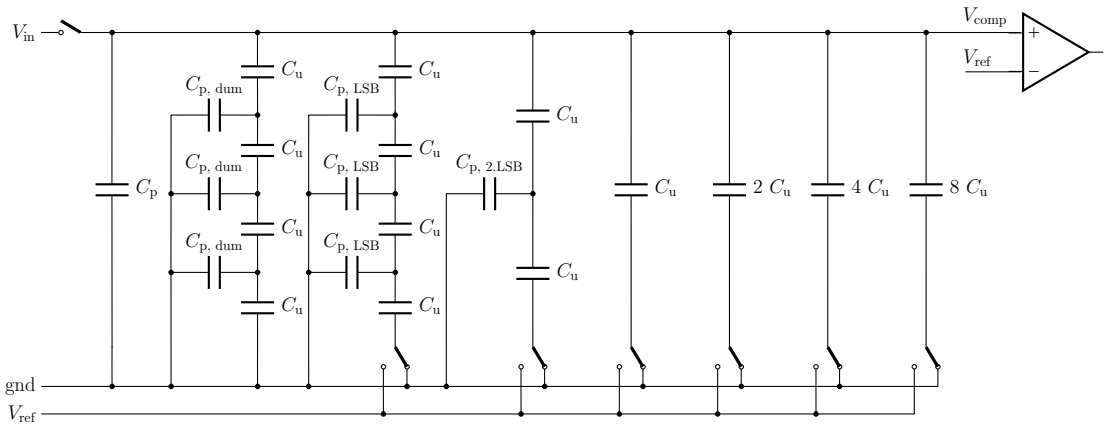


Figure 3.12.: Circuit of a 6-bit SAR ADC with linearly split two bits and parasitic capacitance. Here, even more nodes for parasitic capacitors are present.

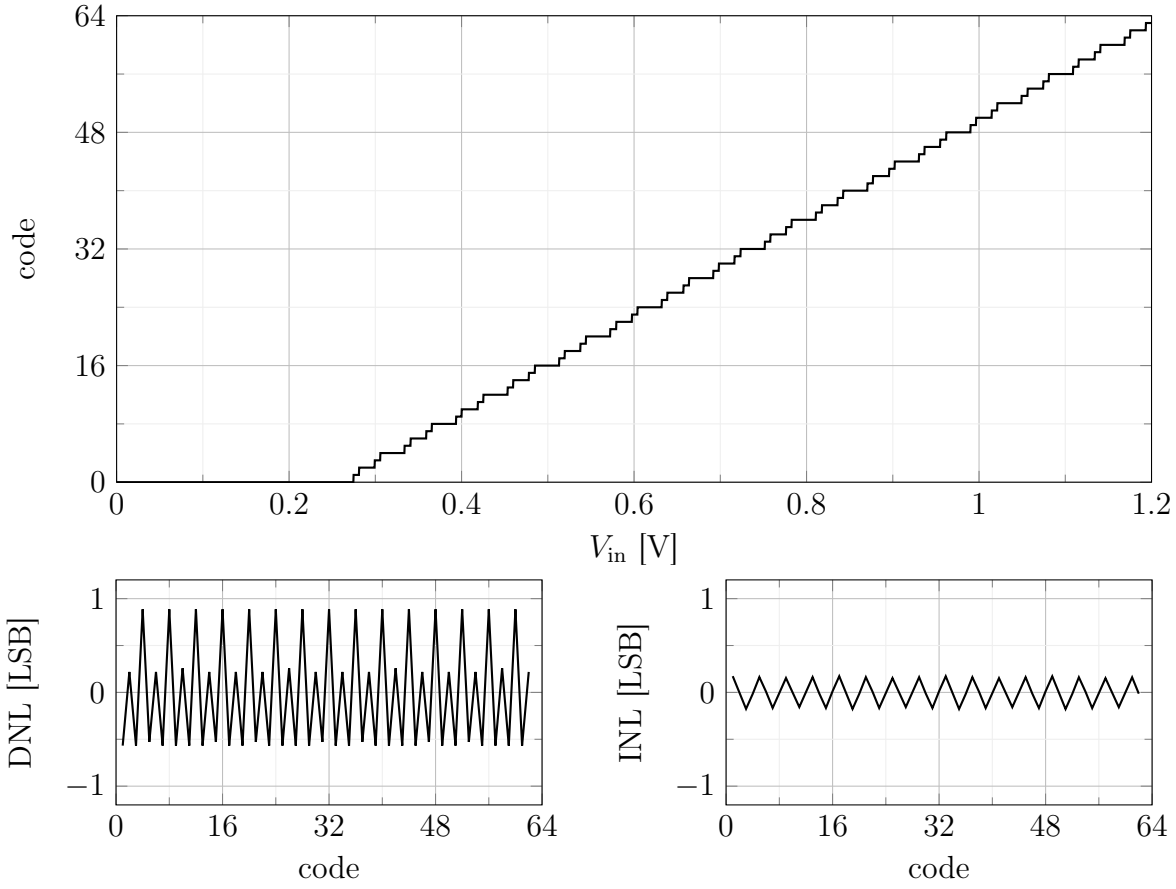


Figure 3.13.: Linearly split two bits ADC characteristics for the case with parasitic capacitance of 20%. Besides an increased offset error, the transfer curve shows clear differences in the step sizes that are reflected in the linearity. In the DNL both bits impaired by additional parasitic nodes are reflected. The jags of the INL reveal the binary structure of the ADC.

in the transfer function and non-linearities depicted in figure 3.13. Besides the offset and gain error, the effect of varying step sizes for every two adjoining output codes aggravates. Moreover, also deviations between the wider steps are visible, denoting the split of the penultimate bit, as well. They are reflected in the integral non-linearity by regular jags, which is typical for binary architectures. Since now even more parasitic capacitance is present, the differential non-linearity error reaches a significant level.

Furthermore, if we take a look at the case with mismatch, we can see a clear division in a 4-bit coarse and a 2-bit fine structure (figure 3.14). Using more capacitors to implement the two less significant bits allows more room for mismatch to impair the conversion. In addition, varying widths of the plateaus again depict mismatches at the transitions, where single capacitors are switched for multiple smaller ones.

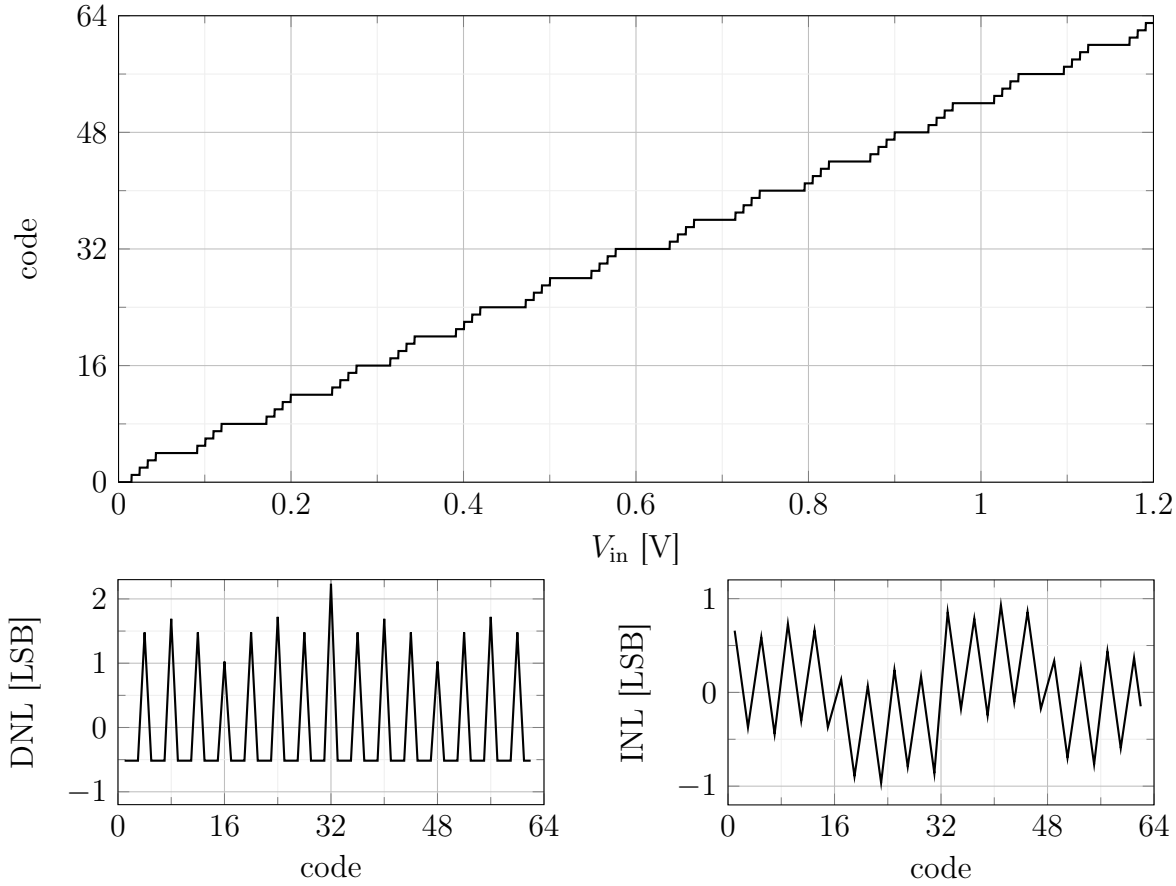


Figure 3.14.: Linearly split two bits ADC characteristics for the case with $\sigma = 0.05$ mismatch. The increased number of correlated components is more prone to mismatch, leading to large deviations in the linearity. Note here the scaled axis of the DNL.

Overall, we see that the implications of linearly scaling two bits with capacitors in series are much more severe than splitting only a single bit. Thus, there is not much reason in continuing this procedure for more bits. Instead, we take a look, if the result for scaling the output voltage of multiple bits can be improved by applying a binary division instead of a linear one.

3.5.3. Binarily split two bits

For implementing a binary division of the less significant bits, a second binary-weighted array is constructed with a total capacitance $C_{\text{LSB, tot}}$, whose top terminals all connect to a bridge capacitor C_{b}. It fulfills the purpose of scaling the output of the LSB array to appropriately match the MSB array. Figure 3.15 illustrates this configuration. Connected to the comparator are the top terminals of the MSB array. Switching a

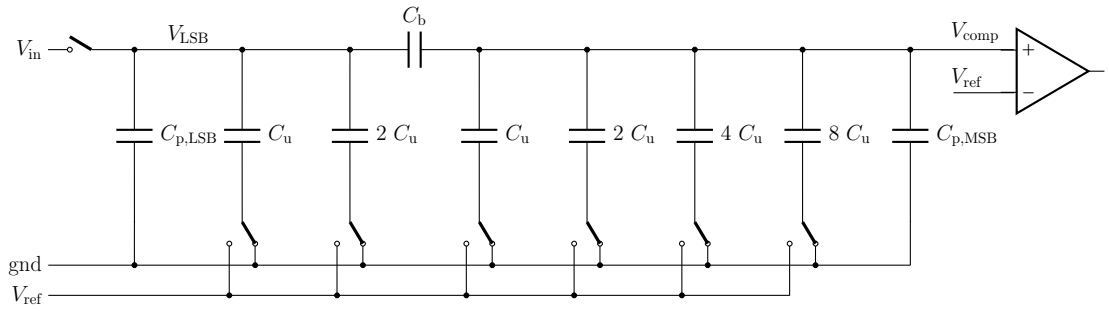


Figure 3.15.: Circuit of a 6-bit SAR ADC with binary split two bits and parasitic capacitance. Here, both MSB and LSB side have a parasitic capacitor in parallel.

capacitor of this array, results in a voltage change at the comparator of

$$\Delta V_{\text{comp, MSB}} = \frac{2^k\, C_{\text{u}}}{C_{\text{MSB, tot}} + \frac{C_{\text{b}} C_{\text{LSB, tot}}}{C_{\text{b}} + C_{\text{LSB, tot}}}} V_{\text{ref}}, \tag{3.11}$$

where $C_{\text{MSB, tot}}$ is the total capacitance of the MSB array and the right-hand term in the denominator represents the serial connection of C_{b} and $C_{\text{LSB, tot}}$. Switching a capacitor of the LSB array, however, results in a change of voltage at their top terminals of

$$\Delta V_{\text{LSB}} = \frac{2^k\, C_{\text{u}}}{C_{\text{LSB, tot}} + \frac{C_{\text{b}} C_{\text{MSB, tot}}}{C_{\text{b}} + C_{\text{MSB, tot}}}} V_{\text{ref}}. \tag{3.12}$$

Scaling this by the bridge capacitance yields a voltage change at the comparator of

$$\Delta V_{\text{comp, LSB}} = \frac{C_{\text{b}}}{C_{\text{b}} + C_{\text{MSB, tot}}} \Delta V_{\text{LSB}}. \tag{3.13}$$

If we now demand that switching the smallest capacitor of the LSB array changes V_{comp} by a fraction of $\frac{1}{2^{N_{\text{LSB}}}}$, where N_{LSB} is the number of bits in the LSB array, compared to the effect switching the smallest capacitor of the MSB array has, we'll derive a value for the bridge capacitance of

$$C_{\text{b}} = \frac{C_{\text{LSB, tot}}}{2^{N_{\text{LSB}}} - 1}. \tag{3.14}$$

It can be seen that C_{b} is independent of the MSB array, meaning that the split introduced by the bridge capacitor can be applied at any position. Furthermore, it is also possible to insert multiple bridge capacitors, dividing the capacitive array into more sections. For the value of C_{b} this means, in the case, where the dummy capacitor is neglected as depicted in figure 3.15, the bridge capacitance is equal to the unit capacitance. On the one hand, this property comes in handy, since it avoids having to match a capacitance of a complicated fractional value. On the other hand, it introduces a

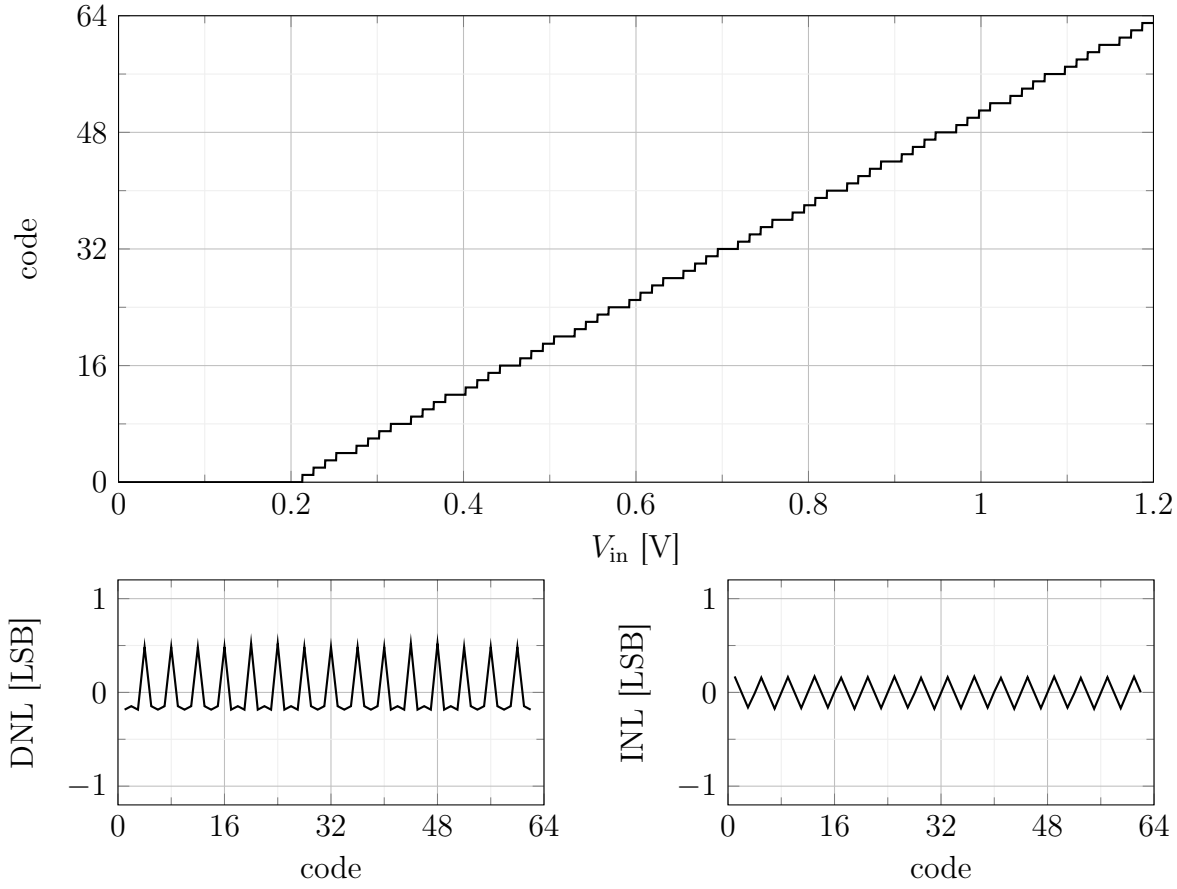


Figure 3.16.: Binarily split two bits ADC characteristics for the case with parasitic capacitance of 20%. Besides the offset error, the matching between MSB and LSB side is impaired, due to the additional parasitic capacitor on the LSB side. This causes error in the linearity.

slight gain error. Inserting $C_{\text{b}} = C_{\text{u}}$ and equation 3.12 into equation 3.13 yields

$$\Delta V_{\text{comp, LSB}} = \frac{2^k}{2^n - 1} V_{\text{ref}}. \tag{3.15}$$

The slight offset of one LSB can be observed, when the circuit is simulated with ideal components. However, this alone would be a cheap trade-off, considering the capacitance saving. Hence, we need to look at physical effects again. Adding mismatch to the capacitors, the situation again is comparable to the case, where no modifications were applied. But looking at the case with parasitic capacitances, considerable DNL errors are present like before (see figure 3.16). Due to the bridge capacitor, there is a parasitic capacitance present in the MSB array, and one in the LSB array. Since the latter contributes to $C_{\text{LSB, tot}}$, having fixed the bridge capacitance at C_{u} violates the demand for correct voltage scaling as formulated by equation 3.14. Thus, parasitic capacitances on the LSB side will attenuate these steps and cause linearity errors at

the transitions between both arrays.

Concluding, we find that the DNL errors can be mitigated by splitting the last two bits in a binary fashion compared to linear one. However, the INL is comparable to the case discussed before.

3.6. Conclusion

Since the capacitance of the digital-to-analog converter scales exponentially with the number of bits, it is desirable to modify the capacitive array, such that the number of unit capacitors can be reduced. To do so, a simple way is operating multiple capacitors in a way that their effective capacitance makes up a fraction of the unit capacitance. Reducing the LSB capacitance this way also leads to a lower total capacitance ans consequently saves area and energy. Table 3.1 summarizes the number of required unit capacitors for the standard DAC and different modifications, depending on the resolution n. Furthermore, the ratio of area and conversion energy compared to the case without modifications is shown.

However, these savings come at a cost in the performance. We have seen in simulation that parasitic capacitances in all configurations cause an offset and, consequently, a gain error. But, looking at the adjusted arrays, also the linearity is impaired, since parasitic capacitors are introduced at further nodes. Before, only mismatch caused non-linearities, because a large capacitor didn't accurately compensate the sum all smaller ones. With the modifications, now, a precise matching between components is even more crucial to avoid linearity errors. The corresponding simulation results are depicted in table 3.2.

Comparing the performances, we find that given the assumptions of a relative par-

capacitive array	no. of unit capacitors	rel. area $n = 6$	rel. area $n = 8$	rel. energy $n = 6$	rel. energy $n = 8$
standard	2^n	1	1	1	1
LSB split	$2^{n-1} + 3$	0.55	0.51	0.49	0.50
linearly split two bits	$2^{n-2} + 9$	0.39	0.29	0.24	0.25
binarily split two bits	$2^{n-2} + 3$	0.30	0.26	0.24	0.25

Table 3.1.: Comparison between different implementations of the capacitive array. By operating multiple unit capacitors in series, the total capacitance and, consequently, the area and conversion energy can be reduced.

capacitive array	20% parasitic capacitance			mismatch $\sigma = 0.05$	
	offset [LSB]	max(DNL) [LSB]	max(INL) [LSB]	max(DNL) [LSB]	max(INL) [LSB]
standard	13.3	0.01	0.01	0.65	0.45
LSB split	14.1	0.20	0.02	0.69	0.45
linearly split two bits	17.5	0.89	0.18	2.23	0.97
binarily split two bits	12.7	0.52	0.18	0.66	0.45

Table 3.2.: Performance of different implementations of the capacitive 6-bit array. Nonlinearities increase in the presence of parasitic capacitance for all modified arrays. Mismatch only has a meaningful influence in the case, where two bits are split linearly.

asitic capacitance of 20% and component mismatch of $\sigma = 0.05$, only the array with a single LSB split behaves acceptably close to the initial case. Even though the DNL increases in the presence of parasitic capacitance, its value is still in a tolerable range. Being able to reduce the area, as well as the conversion energy by approximately 50%, we decide to apply this LSB split in our final implementation. In the following chapter, however, it will be neglected temporarily for simplification.

4. Dual-mode conversion methods

The challenge of this project, besides achieving low energy and low area, is the task of developing an implementation that supports two modes of operation. On the one hand, the ADC should be able to perform a high-speed conversion with a target sampling rate of 125 MS/s at a resolution of 6-bit. For this, we can consider the input voltage to lie in the upper half of the full scale voltage spectrum. On the other hand, the ADC should additionally have a mode supporting an enhanced resolution of 8-bit, however, at a decreased sampling rate of desirable 62.5 MS/s. Here, the input voltage may span the whole possible spectrum. In the following, various concepts to implement these two modes in a single ADC will be discussed. We'll consider their effects regarding the energy and number of clock cycles required for one conversion, as well as their demands concerning the comparator.

Since the only static potentials available in our application are the supply and ground voltage, we start by discussing conversion methods only interfacing these two potentials. Examining energy-saving alternatives, however, we find that further static voltages can yield advantages not only concerning energy, but also in terms of speed. Having said that, it is costly to generate these potentials locally in a way that high currents can be drawn from them. However, a generation for high-impedance loads is feasible. Hence, in a second step we'll expand our considerations to conversion methods additionally utilizing a further static potential at the input of the comparator.

4.1. Considerations without reference voltage generation

Firstly, we take a look at the case, where besides the input voltage only the supply and ground potential are available for the conversion. Note here, that we use the supply voltage, which is equivalent to the full scale voltage, as reference. In our implementation it amounts to 1.2 V, however, these methods are applicable independent of the actually used voltage value. For generality, in the following we'll refer to it as V_{ref}. These boundary conditions allow for different conversion designs.

4.1.1. 6-bit mode with full input range

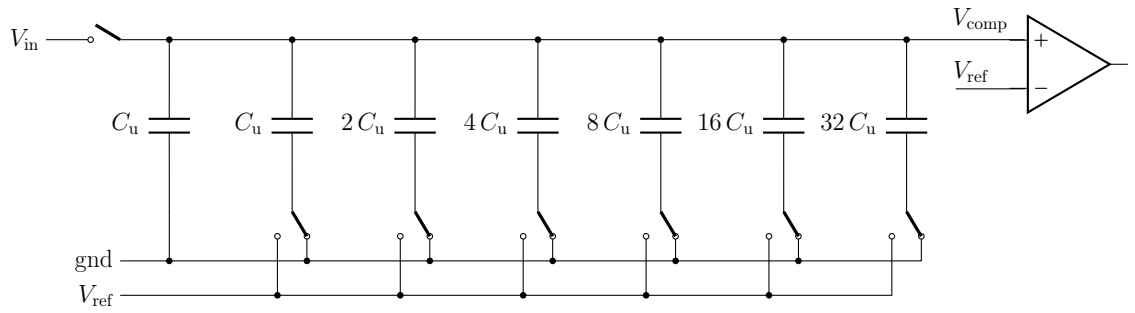
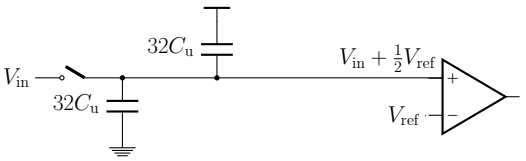


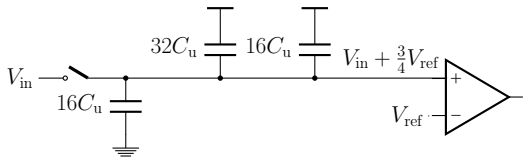
Figure 4.1.: Circuit of a single-ended 6-bit charge-redistribution SAR ADC.

Initially, let's consider the case of a 6-bit SAR ADC, where the input voltage may take any value between ground and V_{ref}. This is a concrete application of the general charge redistribution procedure introduced in section 3.1. Figure 4.1 depicts the schematic of a standard 6-bit binary DAC with a dummy unit capacitor. Its line shared by the top terminals V_{comp} is connected to the non-inverting input of the comparator, while the inverting input is held at the reference voltage. During the sampling phase, the bottom terminals of all capacitors are connected to ground. As soon as the input switch is opened, the MSB capacitor is connected to the reference voltage, resulting in a voltage division of $\frac{1}{2}V_{ref}$. Thus, V_{comp} has the potential $V_{in} + \frac{1}{2}V_{ref}$. Comparing this to the reference is equivalent to the comparison between V_{in} and $\frac{1}{2}V_{ref}$. Consequently, the output of the comparator displays the MSB decision. If it is zero, V_{comp} is less than V_{ref} and needs to be further increased. Therefore, the MSB capacitor will stay connected to the reference potential and in the next step the second largest capacitor will be connected to V_{ref}, as well. This results in a potential of $V_{comp} = V_{in} + \frac{3}{4}V_{ref}$.

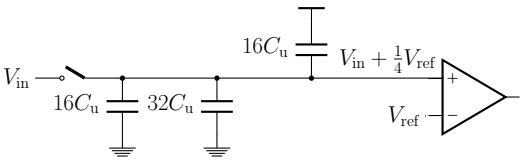
(a) sample



(b) MSB decision



(c) second decision (MSB = 0)



(d) second bit decision (MSB = 1)

Figure 4.2.: Conversion procedure for the 6-bit case with full input range.

The comparison to V_{ref}, then, is equivalent to probing V_{in} against $\frac{1}{4}V_{\text{ref}}$. Hence, the comparator can make the second bit decision.

However, if the MSB decision is one, the potential on the shared line was too high and needs to be reduced. For this, the MSB capacitor is reconnected to ground and the second largest capacitor is connected to V_{ref}. Having a potential of $V_{\text{comp}} = V_{\text{in}} + \frac{1}{4}V_{\text{ref}}$ at the non-inverting input, the effective comparison is between V_{in} and $\frac{3}{4}V_{\text{ref}}$. The comparator, then, makes the second bit decision. All subsequent bit trials follow the same procedure: The capacitor that was just switched will either stay connected to V_{ref}, if the bit decision was zero, or gets reconnected to ground, if it was one. Then, the next capacitor is switched to V_{ref}. Figure 4.2 visualizes this operation principle.

Besides the time needed for the sample and hold operation, the conversion of six bits takes up six cycles. During this time, each capacitor except for the dummy capacitor is charged to V_{ref} once and discharged again, before the next sample is taken. This results in a charging energy per conversion of

$$E_{\text{conv}} = \frac{1}{2} \cdot 63 C_{\text{u}} \cdot V_{\text{ref}}^2. \tag{4.1}$$

Due to the inverting input of the comparator being constantly at V_{ref}, the common mode voltage will approximate this potential. The largest deviation from V_{ref} at the comparator input is present in the first cycle, when $V_{\text{comp}} = V_{\text{in}} + \frac{1}{2}V_{\text{ref}}$. Since V_{in} may take any value between ground and the reference voltage, the non-inverting input of the comparator will lie between $\frac{1}{2}V_{\text{ref}}$ and $\frac{3}{2}V_{\text{ref}}$ for the first bit decision. Thereafter, $\frac{1}{4}V_{\text{ref}}$ will be added in the direction of V_{ref}, such that this input gradually converges to the reference potential. So, the comparator needs to be design to function in the range of $\frac{1}{2}V_{\text{ref}}$ to $\frac{3}{2}V_{\text{ref}}$ with the highest precision at V_{ref}.

4.1.2. 6-bit mode with half input range / 7-bit mode

Next, we take a look at the case if the input voltage ranges only from $\frac{1}{2}V_{\text{ref}}$ to V_{ref}. Now, in order to make the MSB decision, the input needs to be compared to $\frac{3}{4}V_{\text{ref}}$. Since the resolution of six bits should be kept, a simple way to generate this initial voltage division is by adding an additional binary weighted capacitor to the DAC (see figure 4.3). Like in the first approach, during sampling the bottom terminals of all capacitors are connected to ground and the inverting input of the comparator is connected to V_{ref}. When the input switch is opened, instead of the MSB capacitor as before, now the second largest capacitor is connected to V_{ref}, as depicted in figure 4.4.

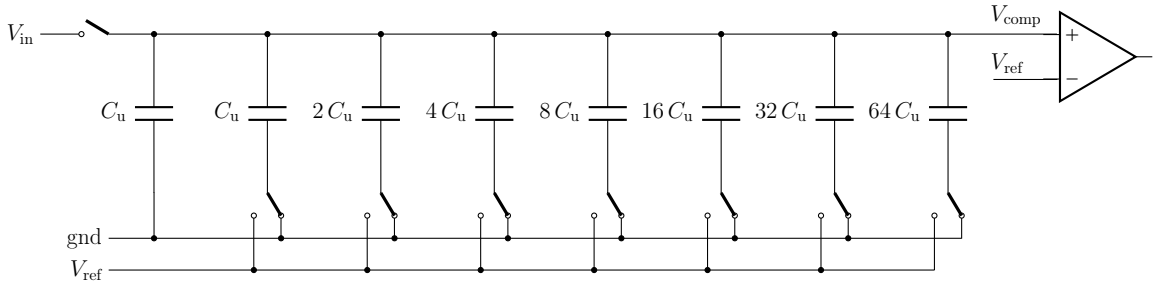
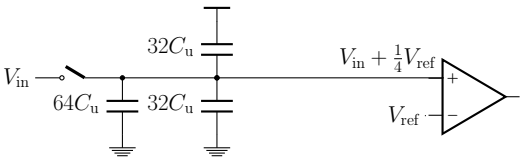


Figure 4.3.: Circuit of a single-ended 7-bit charge-redistribution SAR ADC.

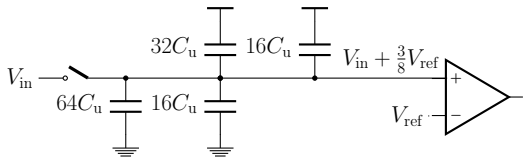
This voltage division results in a potential at the top terminals of $V_{comp} = V_{in} + \frac{1}{4}V_{ref}$. So, the effective comparison is between V_{in} and $\frac{3}{4}V_{ref}$. The following procedure is the same as in the previous section, the only difference being that the additional largest capacitor constantly stays connected to ground. And, since it neither gets charged, nor discharged during the conversion, also the energy and number of cycles per conversion stay constant. However, the total capacitance that is needed doubles, which affects the area of the ADC. But, the requirements regarding the input range of the comparator eases. With V_{in} ranging from $\frac{1}{2}V_{ref}$ to V_{ref}, the non-inverting input for the MSB decision lies between $\frac{3}{4}V_{ref}$ and $\frac{5}{4}V_{ref}$. This spectrum is only half as wide compared to the case, where the input range covers the full scale voltage.

Moreover, it should be noted that this capacitor array will also allow for a 7-bit mode covering the full input range, if the procedure starts with connecting the MSB capacitor to V_{ref} and takes the corresponding comparator result into account.

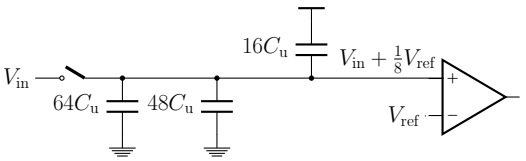
(a) sample



(b) MSB decision



(c) second bit decision (MSB = 0)



(d) second bit decision (MSB = 1)

Figure 4.4.: Conversion procedure for the 6-bit case with half input range.

4.1.3. Single-ended 8-bit mode

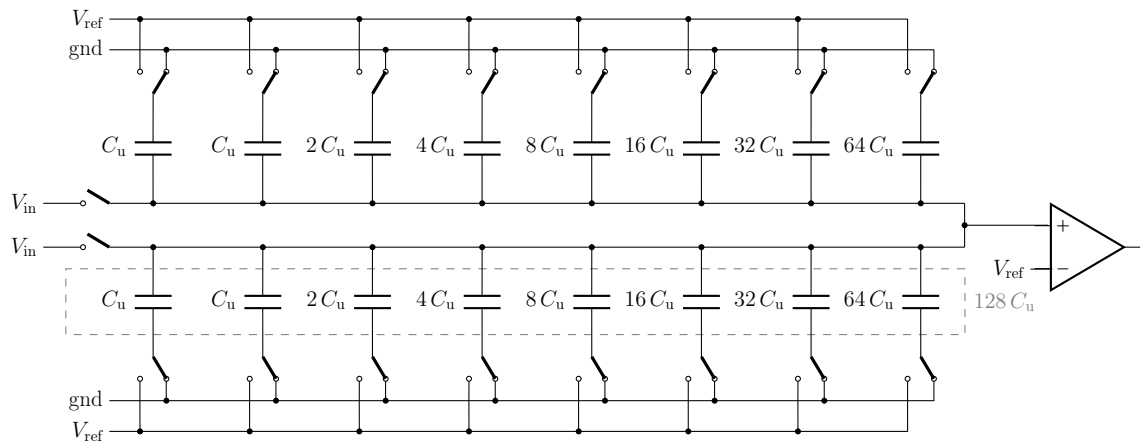


Figure 4.5.: Circuit of a single-ended 8-bit charge-redistribution SAR ADC. The MSB capacitance is made up by a jointly operated 7-bit capacitive array.

Doubling the total capacitance to achieve the same resolution for half the input range is no cheap trade-off for a design with restricted area. However, in an environment, where two modes with different resolutions are desired, this offers the possibility for interesting design concepts. Because looking closely, we find that two 7-bit arrays have a total capacitance of one 8-bit array. So, we are able to combine two ADCs of the fast mode to operate together in the precise mode with increased resolution. The simplest way is depicted in figure 4.5. Here, all capacitors of one 7-bit DAC simply make up the MSB capacitor of the 8-bit DAC. Hence, all of their switching must happen simultaneously. The conversion procedure follows exactly the routine already presented.

Since now two more bits need to be determined, the number of conversion cycles also increases by two. Due to the binary weighting of the capacitance, this influences the conversion energy exponentially:

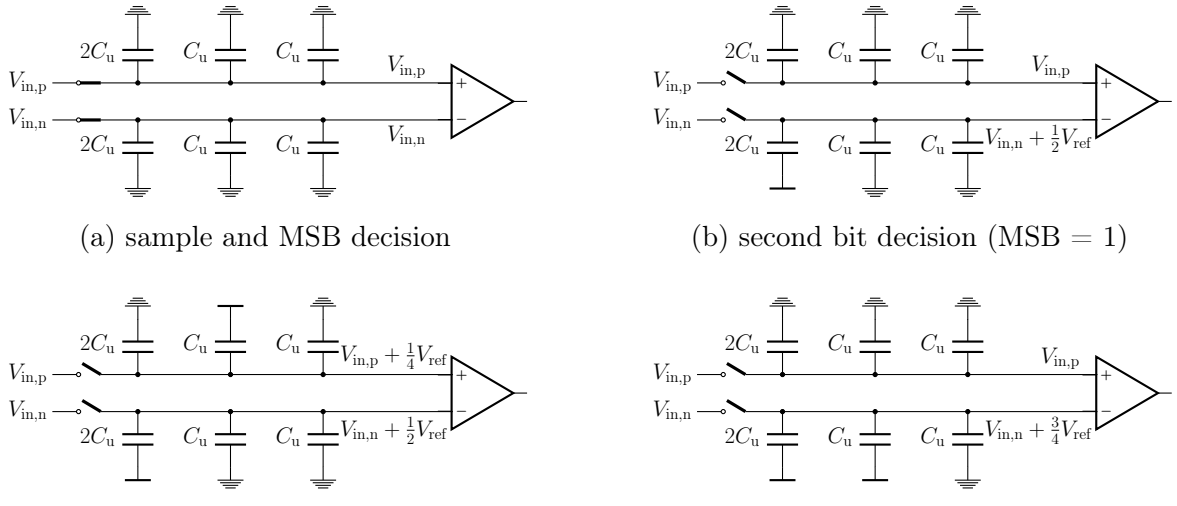
$$E_{\mathrm{conv}} = \frac{1}{2} \cdot 255 C_{\mathrm{u}} \cdot V_{\mathrm{ref}}^2. \tag{4.2}$$

The requirements for the comparator are the same as in section 4.1.1: The input range comprises all voltages between $\frac{1}{2}V_{\mathrm{ref}}$ and $\frac{3}{2}V_{\mathrm{ref}}$ and the common mode voltage approximates V_{ref}.

4.1.4. Differential 8-bit mode

We see that with extended resolution the total capacitance grows exponentially, which increases the energy consumption. Since our ADC ought to be used in a low-power

application, we want to make use of the set-and-down capacitor switching procedure presented by Liu et al. [2009]. This method was developed for differential designs and tackles the issue of conventional capacitor switching based on a trial-and-error search procedure, that a lot of energy will be consumed, if the probing of bits is unsuccessful. Liu et al. [2009] argue that if a trial, i.e. switching a capacitor to V_{ref} at one input side of a differential design, is rejected, this capacitor will be discharged and its counterpart on the other input side needs to be charged in addition. Thereafter, the next capacitor will be probed on the former input side again. To avoid this amount of charging and discharging, they propose to keep the tested capacitor connected to V_{ref} and in case the comparison is unsuccessful, the next capacitor to be switched won't be on the same, but on the other side. This procedure is depicted in figure 4.6. Since the input voltages are directly connected to the comparator, the MSB decision can be made without prior switching. Without loss of generality we assume it to be one. Hence, in the next step the largest capacitor on the negative input side is connected to V_{ref}. The following decision the comparator makes is, whether or not the difference between the two input voltages is larger than $\frac{1}{2}V_{\text{ref}}$. In case this trial is successful, the next capacitor to be switched is on the same input side. However, if the trial is unsuccessful, the second capacitor on the opposing input side will be connected to V_{ref}. This is contrary to the trial-and-error search procedure, where in an unsuccessful case the largest capacitor would be discharged and its counterpart on the opposing input side would be connected to V_{ref}, before the second capacitor would be probed again on the former side. Here, it shall be noted that logically it is equivalent, if the capacitors are initially



(a) sample and MSB decision

(b) second bit decision (MSB = 1)

(c) third bit decision (second bit = 0)

(d) third bit decision (second bit = 1)

Figure 4.6.: Set-and-down capacitor switching method for the differential 8-bit case. The MSB can be determined without prior switching. For simplification we assume without loss of generality MSB = 1.

connected to ground and subsequently switched to the reference potential or vice versa.

Now, if we want to make use of this capacitor switching procedure, there are some adjustments that need to be made. One significant advantage of using a differential design is the possibility to determine the MSB without prior switching. This saves time, as well as energy. However, in order to do so, the input needs to be compared to the appropriate voltage. Since in the 8-bit mode the full input range is allowed, that comparison voltage must be $\frac{1}{2}V_{\text{ref}}$. Consequently, the subsequent capacitor switching needs to introduce a voltage change of $\frac{1}{4}V_{\text{ref}}$. But, if we stick to the arrays of seven binary weighted capacitors, we won't be able to achieve seven binary voltage divisions starting with a factor of $\frac{1}{4}$. This initial fraction would be equivalent to an eighth binary voltage division, however, this is not possible to achieve with seven capacitors. To work around this problem, a solution is to introduce the additional binary split using reference voltage scaling, i.e. connecting the bottom terminals of the capacitors to $\frac{1}{2}V_{\text{ref}}$ instead of V_{ref}. The corresponding circuit and switching scheme are visualized in figure 4.7 and 4.8, respectively.

Since the MSB can be determined directly, the conversion consists of seven cycles, where in each step one capacitor is charged by $\frac{1}{2}V_{\text{ref}}$. This results in a total conversion energy of

$$E_{\text{conv}} = \frac{1}{2} \cdot 127 C_{\text{u}} \cdot \left(\frac{1}{2} V_{\text{ref}}\right)^2 . \tag{4.3}$$

Compared to the single-ended 8-bit case, a significant energy saving can be achieved. One thing that changes for the comparator now is that both its inputs may vary,

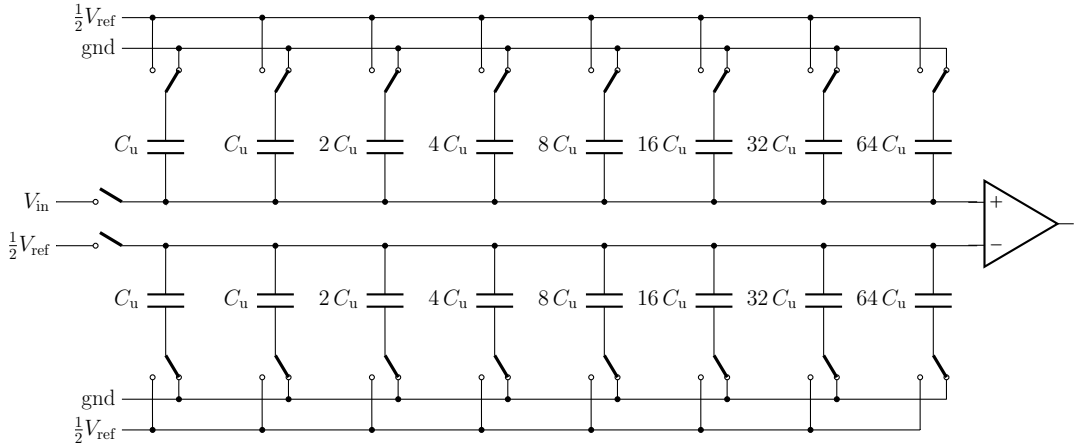


Figure 4.7.: Circuit of a differential 8-bit charge-redistribution SAR ADC. In order to make use of the set-and-down capacitor switching method, it is necessary to load half the reference voltage.

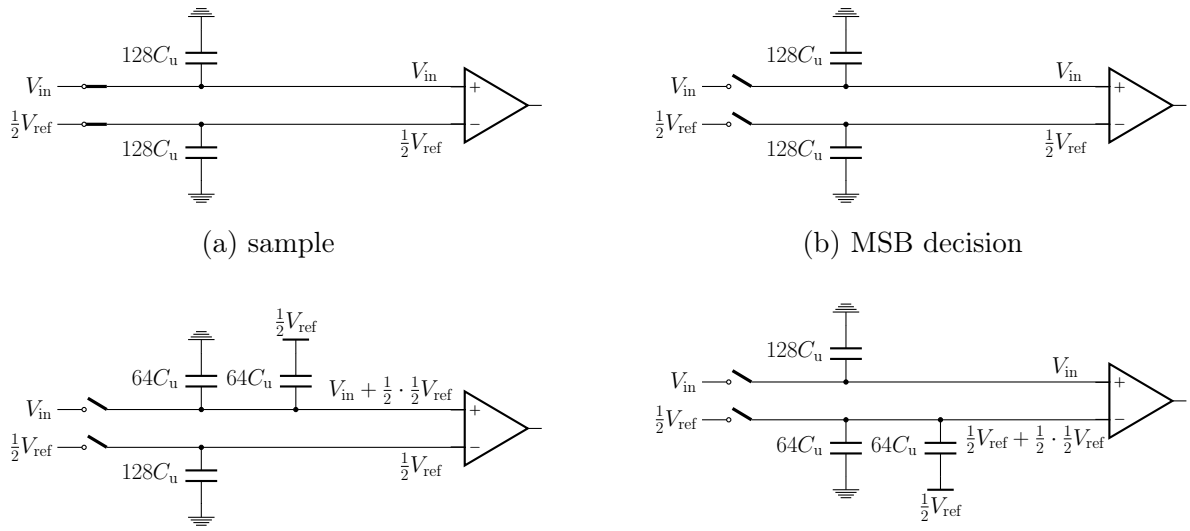
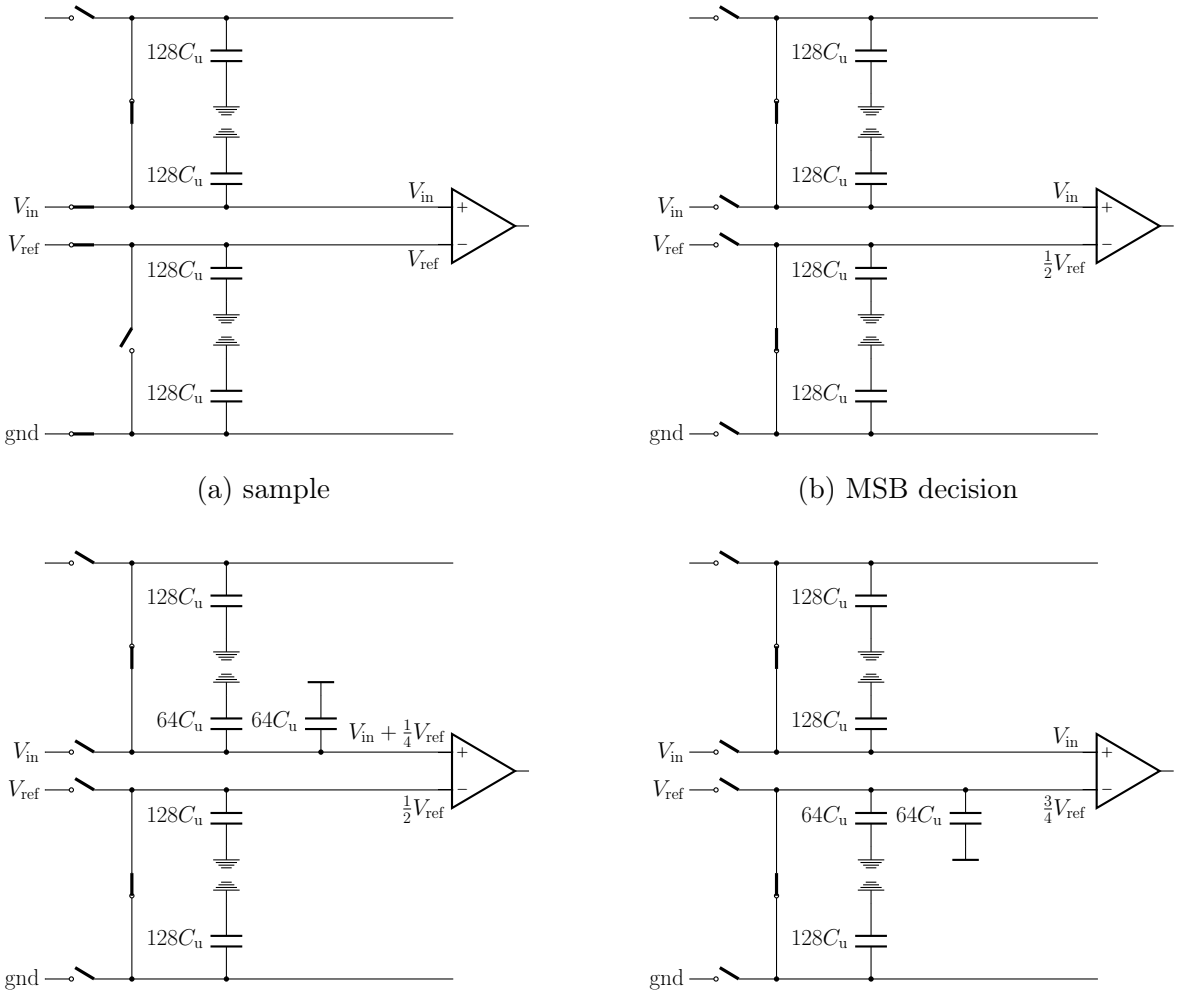


Figure 4.8.: Conversion procedure for the differential 8-bit case. In order to make use of the set-and-down capacitor switching method, it is necessary to generate half the reference voltage and draw current from it.

such that there is not static common mode voltage. Even though they will always settle somewhere between $\frac{1}{2}V_{\text{ref}}$ and V_{ref}, this variability will impair the quality of the comparator, because it is designed to function best at a fixed common mode voltage.

Concluding, we find that on the one hand, this set-and-down switching procedure aggravates the requirements on the comparator. But on the other hand, since less capacitors are switched and the voltage they are charged by is lower, we have an energy conservation of a factor of eight compared to the single-ended method discussed in the previous section. Also, in comparison to other advanced differential designs, this procedure performs energetically superior Liu et al. [2009].

Wanting to apply this procedure, only one problem remains: We started by supposing the only potentials available are ground and V_{ref}. However, for this method it is also necessary to draw currents from $\frac{1}{2}V_{\text{ref}}$. If we keep our initial constraints, it will be necessary to come up with a method to generate half of the reference potential. An approach to do so is by voltage division. But, for this we need more capacitors that can share their charge. Figure 4.9 shows a proposal that in addition uses two further 7-bit arrays. In a highly-parallel application, these can be provided by half of each adjacent 8-bit DACs. In the beginning, the bottom terminals of all capacitor are connected to ground. V_{in} is applied at one input of the comparator and, furthermore, at its adjacent 7-bit capacitor array. On the other input side of the comparator the reference voltage is connected. The top capacitors of its neighboring array are grounded and the switch



(a) sample

(b) MSB decision

(c) second bit decision (MSB = 0)

(d) second bit decision (MSB = 1)

Figure 4.9.: Conversion procedure for the differential 8-bit case. In order to use only the supply and ground potential, it is necessary to utilize the adjacent 7-bit arrays, too.

connecting both lines is open. After sampling, the voltages of all top terminals are disconnected and the named switch gets closed. Thereby, the charge between these two arrays gets shared, resulting in a voltage of $\frac{1}{2}V_{\text{ref}}$ at the comparator input. Now, the MSB decision can be made. Depending on its outcome, one side needs to be varied by $\frac{1}{4}V_{\text{ref}}$. This voltage change can be simply achieved by switching the largest capacitor on the appropriate side of the comparator to V_{ref}. Due to connecting adjacent arrays, the total capacitance has doubled, such that changing half of the voltage in one of the two 7-bit arrays leads to an overall change by a factor of $\frac{1}{4}$.

During the seven steps of one conversion, effectively all the capacitors of a 7-bit array

will be charged to V_{ref}, resulting in a conversion energy of

$$E_{\mathrm{conv}} = \frac{1}{2} \cdot 127 C_{\mathrm{u}} \cdot V_{\mathrm{ref}}^2. \tag{4.4}$$

This means, avoiding the usage of $\frac{1}{2}V_{\mathrm{ref}}$ results in a conversion energy increased by a factor of four, besides an area demand twice as large. In case of a highly-parallel design where the area per se is not increased, but capacitor arrays are shared, this will impact either the number of usable ADC channels, if always the same channels are used for sampling and every second only for charge sharing. Or, the maximum sampling frequency is reduced, if the channels are used in an alternating fashion.

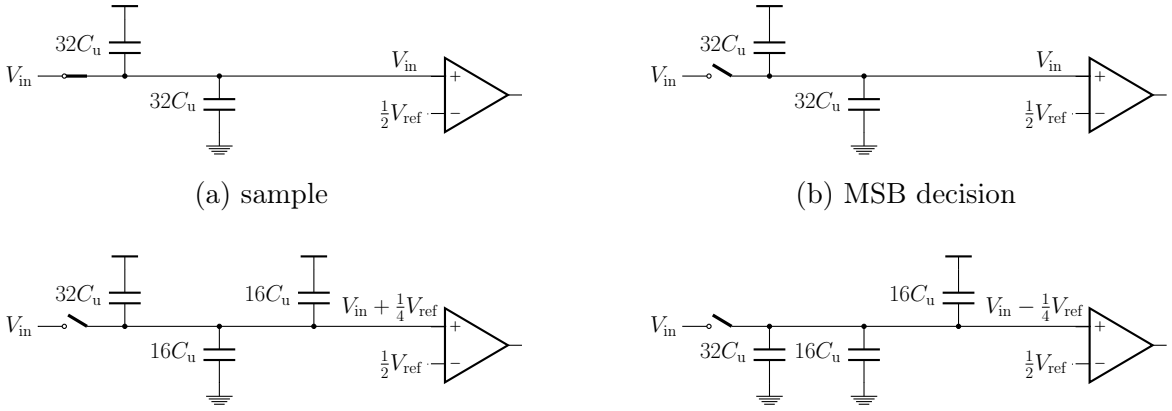
Since no prior switching is necessary, the input of the comparator will always lie between ground and V_{ref}. As discussed, both inputs vary their potential, such that the common mode voltage approaches some value between $\frac{1}{2}V_{\mathrm{ref}}$ and V_{ref}, depending on the input voltage.

4.2. Advanced procedures utilizing reference voltage generation

We started by supposing the only potentials available are the supply and ground voltage. However, as we have seen in the previous section, it is possible to reduce the conversion energy, if a further voltage can be interfaced. Having said that, a local voltage generation is costly, in case high currents are drawn from it. Nevertheless, it can already be beneficial if only the comparator, presenting a high-impedance load, is connected to a different potential. Thus, in the following we will explore further methods making use of an additional reference at the comparator, aspiring to achieve a superior conversion procedure.

4.2.1. 6-bit mode with full input range

If we desire a 6-bit resolution where the input may range from ground to V_{ref}, the new starting configuration, now, is depicted in figure 4.10. Instead of comparing to V_{ref}, the inverting input of the comparator is clamped to $\frac{1}{2}V_{\mathrm{ref}}$. Connecting V_{in} to the other side allows to determine the MSB directly. This is advantageous, since it saves one clock cycle in the conversion. After the first decision was made, $\frac{1}{4}V_{\mathrm{ref}}$ either has to be added to or subtracted from the input voltage. In order to achieve this ratio, the bottom terminal of the MSB capacitor has to be connected to the reference voltage during the sampling phase, while those of all other capacitors are connected to ground.



(a) sample

(b) MSB decision

(c) second bit decision (MSB = 0)

(d) second bit decision (MSB = 1)

Figure 4.10.: Conversion procedure for the 6-bit case with full input range.

If the MSB is zero, the second largest capacitor will simply be connected to V_{ref}. This will lead to a rise at V_{comp} of $\frac{1}{4}V_{\text{ref}}$. Comparing this to $\frac{1}{2}V_{\text{ref}}$ is effectively the same as comparing V_{in} to $\frac{1}{4}V_{\text{ref}}$. However, if the MSB is one, not only will the second largest capacitor be connected to V_{ref}, but also the MSB capacitor will be disconnected from it and switched to ground. On the one hand, this leads to an increase of V_{comp} of $\frac{1}{4}V_{\text{ref}}$, on the other hand to a decrease of $\frac{1}{2}V_{\text{ref}}$. This means, effectively $\frac{1}{4}V_{\text{ref}}$ will be subtracted. The decision of the comparator, then, is equivalent to the outcome of the comparison between V_{in} and $\frac{3}{4}V_{\text{ref}}$.

Taking a look at the charging energy for a single conversion, we find that it now depends on the MSB decision. If the it is one, all capacitors will be charged to V_{ref} and discharged to ground again, before the next conversion starts. If the MSB is zero, however, the largest capacitor won't be switched and thereby won't use any energy.

$$E_{\text{conv}} = \begin{cases} \frac{1}{2} \cdot 31C_{\text{u}} \cdot V_{\text{ref}}^2 & , V_{\text{in}} \leq \frac{1}{2}V_{\text{ref}} \\ \frac{1}{2} \cdot 63C_{\text{u}} \cdot V_{\text{ref}}^2 & , V_{\text{in}} > \frac{1}{2}V_{\text{ref}} \end{cases} \tag{4.5}$$

Consequently, if the comparison voltage is $\frac{1}{2}V_{\text{ref}}$, less or at most an equal amount of energy will be consumed as in the case comparing to V_{ref}.

Besides improving speed and energy consumption, also the requirements for the comparator ease. Previously, we added voltage to the input, before having any information about it. This lead to comparator voltages potentially being higher than the reference voltage. Here, this is avoided by explicitly adding voltage in the direction of the comparison potential from the first step on. Having input voltages just in the range between ground and V_{ref}, a precise design of the comparator is easier to achieve.

4.2.2. 6-bit mode with half input range / 7-bit mode

Considering the case with input voltages ranging only over half of the full scale, a further capacitor is added like in section 4.1.2 for achieving the same resolution in a window only half as wide. Now, making the MSB decision means comparing to half of the allowed range, i.e. $\frac{3}{4}V_{\text{ref}}$. Hence, if we want to make use of the advantageous switching scheme from the previous section, $\frac{3}{4}V_{\text{ref}}$ needs to be applied at the comparator. Since only the comparison voltage is varied and a capacitor is added, that doesn't get switched during the conversion, the number of cycles and energy per conversion are equal to the previous section. The comparator again only has input voltages up to V_{ref}, but now starting at just $\frac{1}{2}V_{\text{ref}}$. However, since the comparison voltage changes, so does the optimal working point it must be designed for.

Having said that, different working voltages of the comparator for different modes are not desirable, because its precision depends on the common mode voltage. In the 8-bit mode, where the full input range shall be covered, a comparison voltage of $\frac{3}{4}V_{\text{ref}}$ is not reasonable. Thus, we will also take a look at how the 6-bit mode with half input range can be implemented, applying $\frac{1}{2}V_{\text{ref}}$ to the comparator. For this, we start with the bottom terminals of all capacitors connected to V_{ref}, except for the MSB capacitor, which is connected to ground. After sampling, the second largest capacitor is connected to ground, too, such that we get $V_{\text{comp}} = V_{\text{in}} - \frac{1}{4}V_{\text{ref}}$. Comparing this to $\frac{1}{2}V_{\text{ref}}$ is equivalent to comparing V_{in} to $\frac{3}{4}V_{\text{ref}}$. If the MSB is zero, the second largest capacitor will be reconnected to V_{ref}. If it is one, however, it'll stay connected to ground. The same procedure follows for all subsequent capacitors.

Due to the first switching, before being able to make the MSB decision, this routine needs six conversion cycles as in section 4.1.2, comparing to V_{ref}. This is one more cycle with respect to a comparison voltage of $\frac{3}{4}V_{\text{ref}}$. Also, the conversion energy is the same as in section 4.1.2:

$$E_{\text{conv}} = \frac{1}{2} \cdot 63C_{\text{u}} \cdot V_{\text{ref}}^2 \tag{4.6}$$

The working point of the comparator, as well as its input range, are simply shifted by $\frac{1}{2}V_{\text{ref}}$.

Again, making the largest capacitor switchable, this array can also be used for a 7-bit resolution with input voltages ranging from ground to V_{ref}. Since this doesn't affect the conversion speed, we decide for this increased precision. The resulting conversion procedure effectively equals the operation depicted in figure 4.10 with capacitance

values twice as large. Also, the charging energy is dependent on the MSB decision, like in equation 4.5:

$$E_{\text{conv}} = \begin{cases} \frac{1}{2} \cdot 63 C_{\text{u}} \cdot V_{\text{ref}}^2 & , V_{\text{in}} \leq \frac{1}{2} V_{\text{ref}} \\ \frac{1}{2} \cdot 127 C_{\text{u}} \cdot V_{\text{ref}}^2 & , V_{\text{in}} > \frac{1}{2} V_{\text{ref}} \end{cases} \tag{4.7}$$

Even though the energy consumption increases for high input voltages, the enhanced precision and conversion range come at no additional costs in terms of area and speed. Therefore, instead of considering a 6-bit resolution with an input range spanning only half of the full scale voltage, in the following we will treat a 7-bit resolution with full input range for the fast mode.

4.2.3. Single-ended set-and-down 8-bit mode

The greatest advantage of using $\frac{1}{2}V_{\text{ref}}$ as comparison voltage, however, shows in the 8-bit mode. The set-and-down switching procedure for differential ADCs presented in section 4.1.4 had the shortcoming, that either $\frac{1}{2}V_{\text{ref}}$ needs to be available for drawing current from or that the total capacitance needs to be doubled, in order to share charge. Having $\frac{1}{2}V_{\text{ref}}$ available at the comparator input, now, allows to overcome these shortcomings. Figure 4.11 depicts the configuration for the high resolution mode. Apart from the potential at the comparator, this is effectively the circuit shown in 4.1.3: Both 7-bit arrays are connected to one side of the comparator, while the other side, here, is connected to $\frac{1}{2}V_{\text{ref}}$. But, instead of using one array as MSB capacitance, now we want to make use of the set-and-down switching procedure, however, not in a differential, but in a single-ended approach. To do so, initially all bottom terminals of one 7-bit array are connected to ground, while those of the other are connected to V_{ref} (see figure 4.12). The input voltage is applied at the top terminals, i.e. directly at the

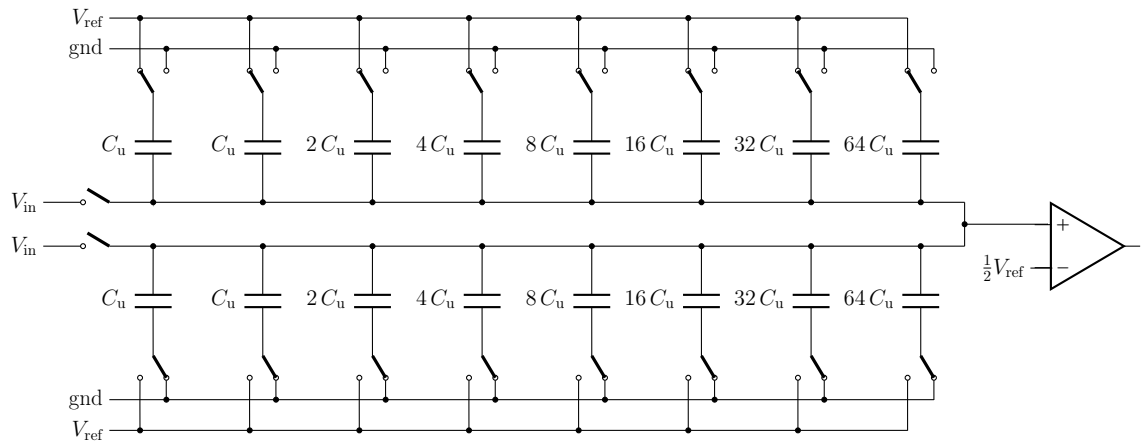
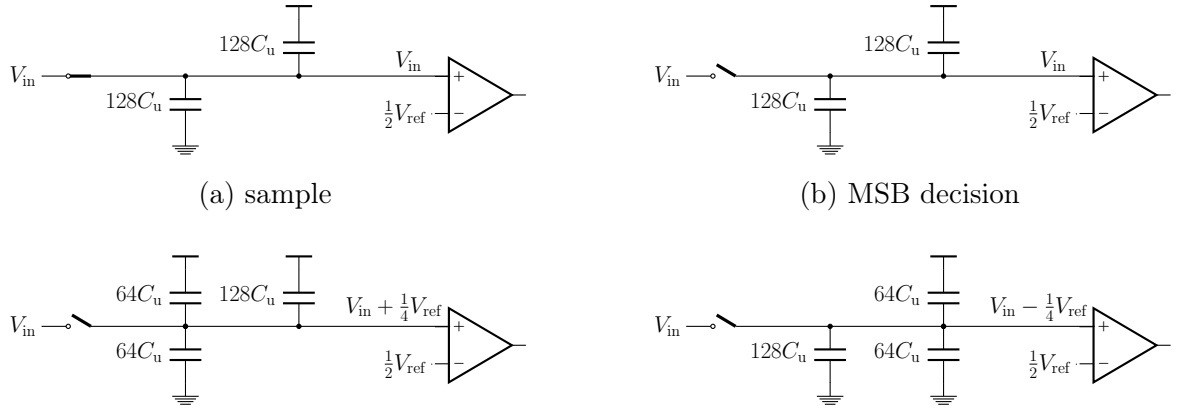


Figure 4.11.: Circuit of a single-ended set-and-down 8-bit charge-redistribution SAR ADC.



(a) sample (b) MSB decision

(c) second bit decision (MSB = 0) (d) second bit decision (MSB = 1)

Figure 4.12.: Conversion procedure for the 8-bit case applying the set-and-down procedure in a single-ended fashion.

comparator input, such that the MSB decision can be made without prior switching. If V_{in} is less that $\frac{1}{2}V_{\text{ref}}$, the potential of this line will need to be increased. Therefore, the largest capacitor of the array, that initially was connected to ground, is switched to V_{ref} . Since this changes one quarter of the total capacitance, the next decision of the comparator effectively is between V_{in} and $\frac{1}{4}V_{\text{ref}}$.

However, if V_{in} is greater than $\frac{1}{2}V_{\text{ref}}$, the potential needs to be decreased. Hence, the largest capacitor of the array initially connected to V_{ref} is switched to ground. This lowers the potential to $V_{\text{in}} - \frac{1}{4}V_{\text{ref}}$, which compared to $\frac{1}{2}V_{\text{ref}}$ is the same as comparing V_{in} to $\frac{3}{4}V_{\text{ref}}$. This procedure continues until one of each binary-weighted capacitors is switched once.

Thus, seven conversion cycles are required and the energy per conversion is

$$E_{\text{conv}} = \frac{1}{2} \cdot 127C_{\text{u}} \cdot V_{\text{ref}}^2. \tag{4.8}$$

This is the same result we have seen in section 4.1.4, however, without comprising neither the number of simultaneously usable ADC channels, nor the maximum sampling rate. Furthermore, the common mode voltage at the comparator always approaches $\frac{1}{2}V_{\text{ref}}$ irrespective of the input voltage, allowing to optimize the implementation for this working voltage.

In conclusion we find that being able to apply half the reference voltage at the comparator allows to save time and charging energy in the 6-bit mode with full input range. In the case, where only half the input range is expected, the same advantages are present utilizing $\frac{3}{4}V_{\text{ref}}$. However, to ease the requirements for the comparator, we

decide to also use $\frac{1}{2}V_{\text{ref}}$ as comparison voltage here. Even though this doesn't yield a time- or energy saving compared to applying V_{ref} at the comparator, it gives us an additional bit for free. Thus, instead of a 6-bit precision with half input range, we'll implement a 7-bit precision with full input range. The speed-up compared to the 8-bit mode not only comes from having to perform one decision less, but we use two 7-bit ADCs in a time-interleaved fashion. In the 8-bit mode, then, both capacitive arrays are operated together. By applying the set-and-down switching principle at a single input of the comparator, we are able to save half of the charging energy with respect to a standard 8-bit charge redistribution ADC.

5. Full ADC implementation and performance evaluation

Finally, the full analog-to-digital converter can be assembled with the considerations presented so far and the work carried out by Dauer [2022]. Therefore, we take a look at the implementation of the components individually and analyze the full performance on a transistor-level. Afterwards, the design is put into a physical implementation. This allows to accurately model layout-dependent effects and inspect their impact on the ADC's performance. Furthermore, its behavior is evaluated regarding the targets defined in the beginning.

5.1. Schematic circuit design

Firstly, we discuss the full ADC design independently of its layout implementation. For this, we look at the chosen dual-mode conversion method in combination with the modification applied to the LSB capacitance of the DAC. Moreover, the design of the comparator and the generation of the reference voltage is briefly described.

5.1.1. Digital-to-analog converter

For the digital-to-analog converter of the SAR ADC we combine the conversion method presented in section 4.2 with the capacitive array having one LSB split discussed in section 3.5.1 to save area and energy. Hence, the MSB capacitor of a 7-bit array has a value of $2^5\,C_\mathrm{u}$ and for the implementation of each array 67 unit capacitors are required, according to table 3.1. Figure 5.1 shows the complete arrangement for one channel. In the fast mode, both arrays function in a time-interleaved fashion, i.e. alternating, for achieving a higher sampling rate. In the precise mode, however, the total capacitance of both arrays is required for the conversion. Thus, they need to operate jointly.

The timing of the control signals for the time-interleaved 7-bit mode is depicted in figure 5.2. In the beginning, the input voltage is sampled on the capacitors of array A.

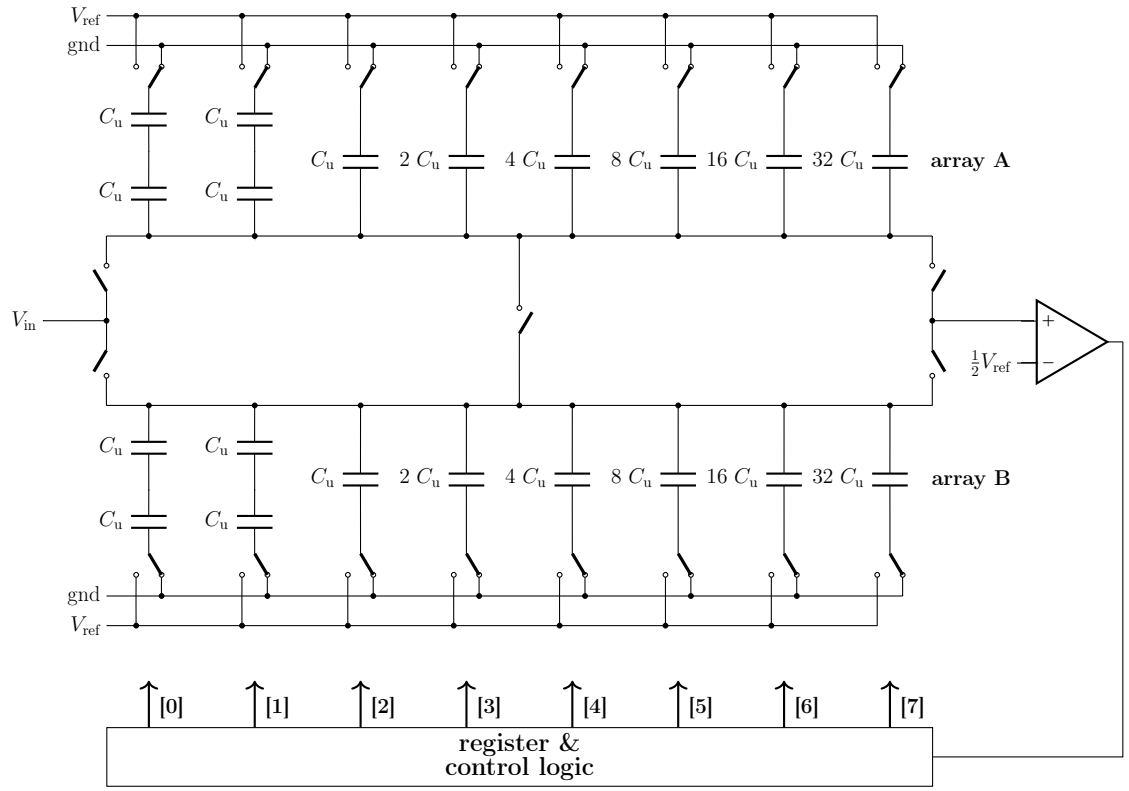


Figure 5.1.: Circuit diagram of the full ADC. The two capacitive arrays can either function in a time-interleaved fashion for a fast 7-bit conversion, or jointly with a decreased sampling frequency at an 8-bit resolution.

For this, all bottom terminals are connected to ground, except for the MSB capacitor, that is connected to V_{ref} as discussed in section 4.2.2. This is indicated by the associated control signal `array A [7]`. After the input is sampled, the DAC is connected to the comparator. Since half of the reference voltage is present at its other terminal, the MSB decision can be made right away. If it is one, the MSB capacitor will be switched to ground, otherwise, it will stay in place. Then, the second largest capacitor is switched to V_{ref} by `array A [6]` and the procedure repeats, until every binary-weighted capacitor is tested. In the end, the digital output code is represented by `array A [7:1]`. `array A [0]` comes into play in the 8-bit mode. While array A performs its bit trials, the next input sample is taken by array B. Once the conversion of array A ended, array B is connected to the comparator. During this time, array A is reset and, then, samples the next input voltage. Like this, a continuous operation is achieved.

For the precise mode with 8-bit resolution both capacitor arrays function jointly as discussed in section 4.2.3. Furthermore, to achieve faster settling times, both arrays are shorted directly. Now, array A and B are controlled simultaneously, as shown in figure

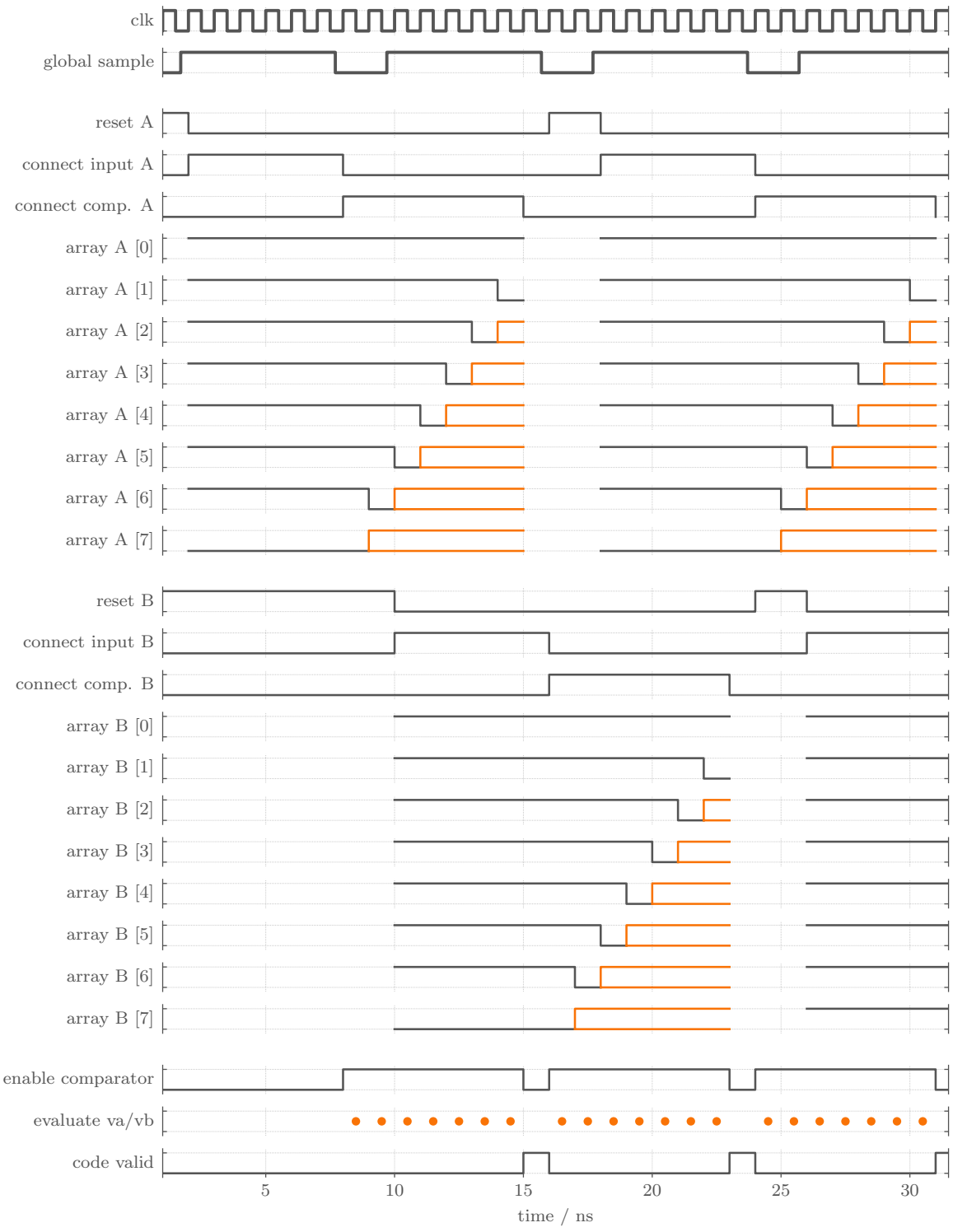


Figure 5.2.: Timing diagram for the 7-bit mode. The capacitive arrays work in a time-interleaved fashion and share the same comparator. Figure taken from Dauer [2022].

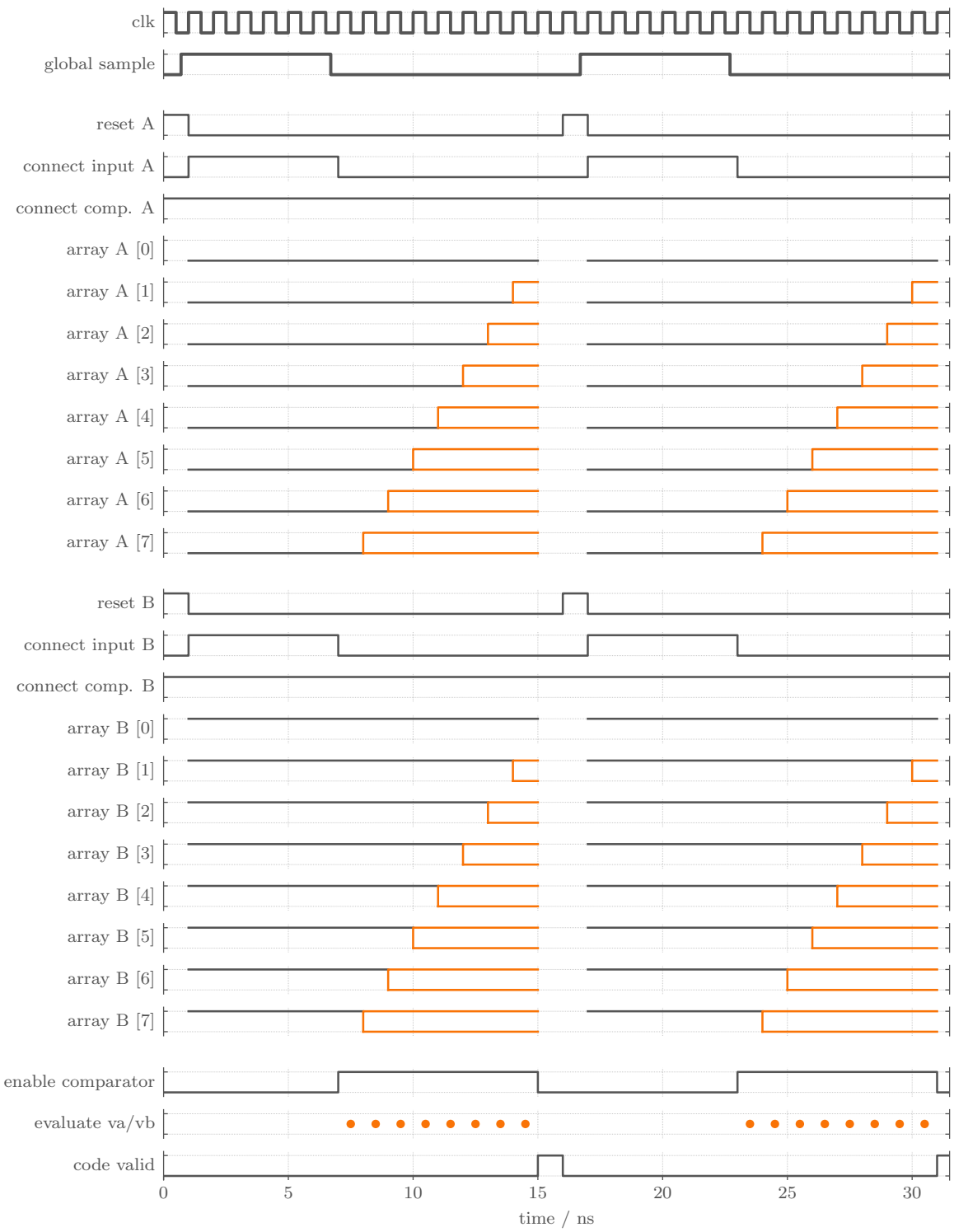


Figure 5.3.: Timing diagram for the 8-bit mode. Both arrays function together to perform the conversion. Figure taken from Dauer [2022].

5.3. After resetting, both capacitor arrays are connected to the input voltage. Thereby, the bottom terminals of array A interface the reference voltage and those of array B the ground potential. Afterwards, both DACs are connected to the comparator and the MSB decision can be made without prior switching. If the output of the comparator is positive, the largest capacitor of array A will be switched to ground. However, if it is negative, the largest capacitor of array B will be switched to the reference voltage. When the new potential at the comparator input has settled, the next decision can be made. This switching procedure repeats for all seven binary-weighted capacitors and after the last one is switched, the comparator makes its final decision.

For a detailed insight into the implementation of the digital control logic, the reader is referred to Dauer [2022].

5.1.2. Comparator and reference voltage generator

Besides the digital-to-analog converter, one of the main components of the SAR ADC is the comparator. Its task is to determine the level of the input voltage with respect to a reference voltage. Various implementations of this component were examined and evaluated by Dauer [2022]. The selected design is a double-tail sense amplifier which can be seen in figure 5.4 (van Elzakker et al. [2008]). It consists of two stages: Firstly, a differential pair senses the voltage difference between the inputs V_p and V_m. For this, the nodes `dip` and `dim` are pre-charged to the supply voltage, which is interrupted by the trigger signal. Then, the amplified input voltage difference is represented by Δ`di` = `dip - dim`. To generate a recognizable logic level at the output, the potentials `dip` and `dim` are passed to a latch. In the beginning of the decision cycle, the output nodes `outp` and `outm` were pre-charged to the ground potential. So, at the arrival of the differential voltages `dip` and `dim`, they settle in the configuration determined by their difference. Thus, a stable output voltage is ensured. Furthermore, due to the triggering, a quiescent current can be avoided.

In our application, the static potential the inputs are compared to amounts to half of the supply voltage. However, since this potential is not applied externally as discussed in section 4, we need to generate it locally. For this, a reference DAC is used. It consists of two 5-bit capacitive arrays that apply the principle of voltage division, like the capacitive DAC arrays serving as the foundation for the SAR ADC itself. By the means of its configuration the offset of the analog-to-digital conversion can be adjusted. This allows to compensate for mismatch-induced offsets or to optimize the conversion range for individual applications. During the conversion, the kick-back of the compara-

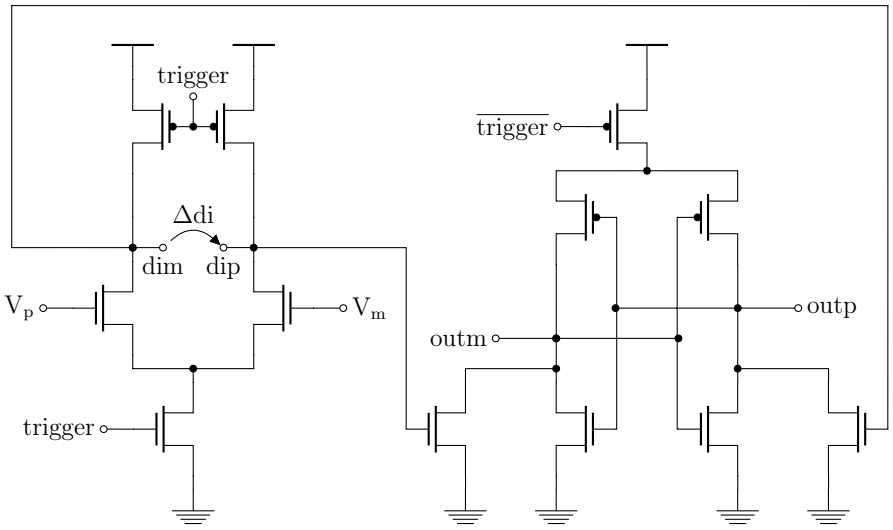


Figure 5.4.: Circuit diagram of the comparator. In this double-tail sense amplifier the difference of the input voltages is enhanced by a differential pair, followed by a latch. Figure taken from Dauer [2022]

tor draws a small amount of current from the reference DAC, reducing its potential. Hence, for the voltage loss not to influence the output, it is necessary to refresh the reference DAC during the conversion. At the target frequency, the period for this is determined to 400 clock cycles. To allow a continuous operation anyways, always one of the two capacitive arrays is connected to the comparator, while the other is given time to be refreshed.

In the following, for evaluating the performance of the SAR ADC, the components presented in this work are operated together with the components designed by Dauer [2022], namely the comparator, reference DAC and the digital controller.

5.2. Physical implementation

So far, the ADC design was presented generally, such that it can be implemented in any technology. Now, we choose a specific process and develop a concrete realization. Since our aim is to incorporate the ADC into the BrainScaleS-2 (Schemmel et al. [2020]) environment, some constraints need to be respected. On the one hand, of course the same technology needs to be utilized, namely the 65 nm low power process from the Taiwan Semiconductor Manufacturing Company (TSMC). On the other hand, the limitations within the chip structure need to be considered. Since this ADC is supposed to replace the column-analog-to-digital converter, its width is restricted to 11.76 µm, the size of a neuron column. With a height of 111.98 µm, a total area of 1317 μm^2

is achieved for the full ADC. Furthermore, on every hemisphere of the chip 256 ADC columns will be placed adjacently. Thus, the input can only be applied on one side and the digital control signals on the opposing edge. This needs to be respected, creating the layout.

5.2.1. Layout Overview

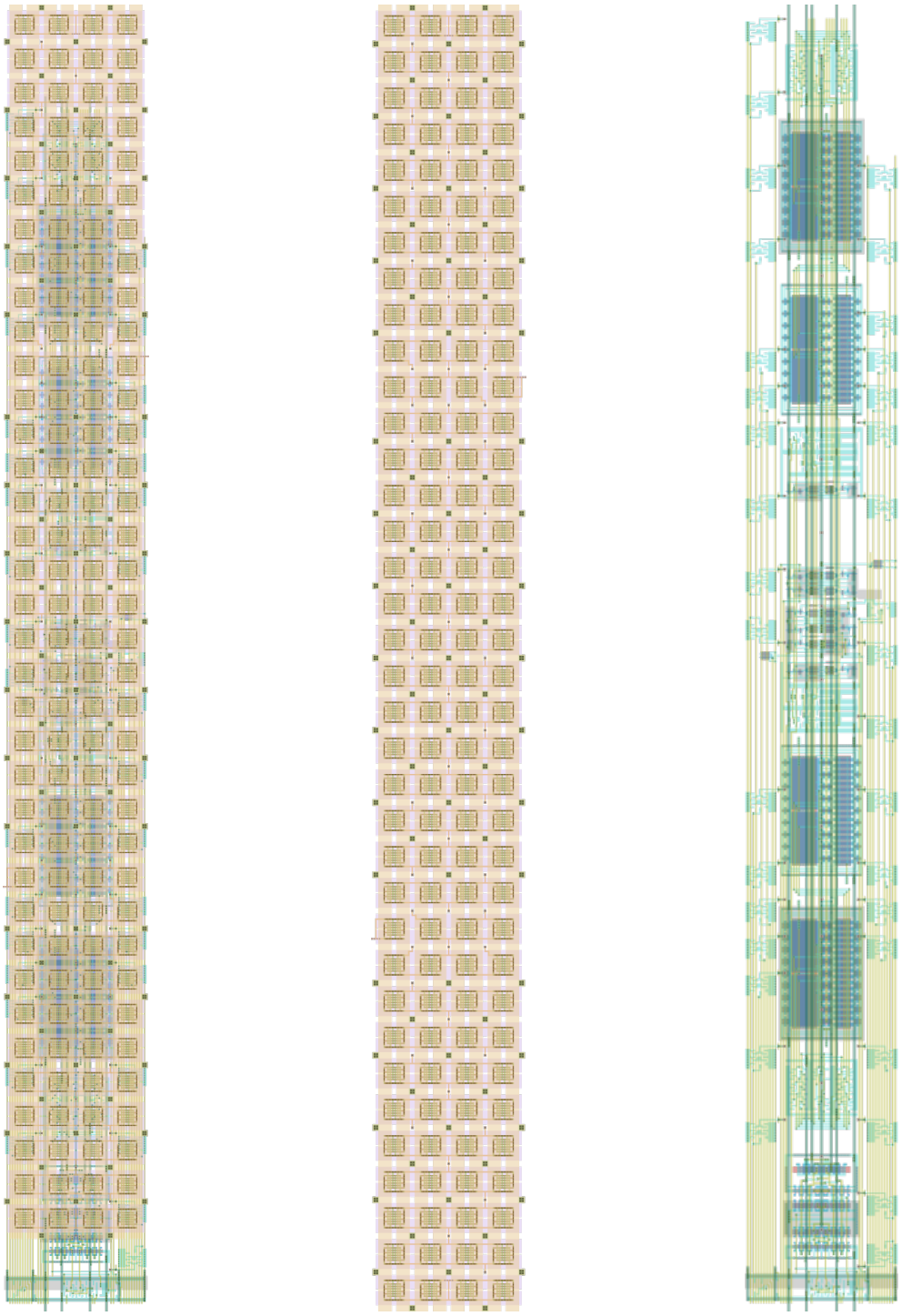
The full layout of an ADC column is depicted in figure 5.5. On upper metal layers, the capacitors of the DAC arrays are implemented. Underneath, the switches controlling them are placed in the outer 2.50µm of the column. Furthermore, the comparator and reference voltage generator are located in the central region. More specifically, the arrangement look as follows:

For the layout there are nine metal layers available. Since the chip power is applied externally via pins on the surface, top metal layer nine distributes the supply and ground voltages horizontally over the chip.

To realize the capacitors, we decided to use metal-oxide-metal (MOM) capacitors. Typically, they are implemented using multiple adjacent metal layers and make use of the parasitic capacitance between close wires. This choice and the implementation of MOM capacitors will be discussed in more detail in the following section. Here, the key point is that we'd like to utilize at least three adjacent metal layers to implement the capacitors. Using metal eight is inconvenient, because with respect so the metal layers underneath the minimum wire size on this layer increases. Hence, we want to use metal seven to five for realizing the MOM capacitors.

This just raises one problem: Since the analog-to-digital converter shall be deployed in the environment of BrainScaleS-2, there are some constraints we need to take into account. Most prominently concerning the metal stack, there are 16 wires running vertically on metal six over each ADC column, connecting the synapse array on its one side with the plasticity processing unit on its other side. Moreover, four further wires are routed in parallel carrying control signals. Since implementing the metal-oxide-metal capacitors on lower layers would block or significantly impede the routing to underlying transistors, we need to redirect these signals from metal six to another layer. The best solution would be routing them on metal eight, so they are to some extend independent of our ADC implementation. But, as mentioned earlier, the minimum wire size and, furthermore, the minimum wire spacing on metal eight increases, which has the consequence that not all signals would fit into the width of one ADC column on metal eight anymore. Hence, unfortunately the signals need to be redirected to a lower layer.

Instead, metal eight is used as a vertical supply layer, that at the same time serves



(a) (b) (c)

Figure 5.5.: Layout of the full ADC channel. (a) All metal layers are depicted. (b) The MOM capacitors are implemented on M5-M7. Its adjacent layers are used for shielding. The supply voltages are passed through via stacks. (c) On M3 and below the switches controlling the DAC are placed in the outer regions. At the bottom the comparator is located next to some buffers. The reference DACs utilize a large polysilicon area. In the center are the transmission gates connecting the DAC to the input and the comparator. Note that at some places the active diffusion is not shown.

layer	purpose
metal 9	horizontal power supply
metal 8	vertical power supply / shielding
metal 7	metal-oxide-metal capacitor
metal 6	metal-oxide-metal capacitor
metal 5	metal-oxide-metal capacitor
metal 4	horizontal power supply / shielding
metal 3	connections to capacitors, (synapse RAM and control signals)
metal 2	control signals, comparator and reference DAC
metal 1	control signals, comparator and reference DAC

Table 5.1.: Metal layer purposes in the physical implementation of the ADC.

as shielding for the capacitors. In addition, so does metal four. Consequently, the metal-oxide-metal capacitors will always see static potentials on the layers adjacent to them, preventing signal dependent voltages from corrupting the conversion.

On metal three, then, the synapse RAM and control signals from metal six are redirected, alongside connections interfacing the capacitors. Metal two and one carry control signals for the switches and are used for the implementation of the comparator and its reference DACs. A summary of the purpose of all metal layers is given in table 5.1.

A detailed description of the physical implementation of the comparator and its related components is given in Dauer [2022]. Here, we will discuss the realization of the digital-to-analog converter. For this, firstly we will take a look at the implementation of a unit capacitor, before describing the arrangement of the complete capacitor array.

5.2.2. Unit capacitor

For our digital-to-analog converter we need capacitors with values scaling binarily from $1\,C_\mathrm{u}$ to $2^5\,C_\mathrm{u}$. What is commonly done in designs where a precise matching of capacitance ratios is important, is that larger capacitors are made up of multiple unit capacitors, that are operated in parallel (Carusone et al. [2013]). The reason for this is overetching. Due to the production process, the edges of the capacitor on chip are narrowed with respect to their size in the layout mask. This will effect the ratio of two capacitors in the physical implementation to differ from their originally targeted value, if they are simply implemented as single capacitors with different sizes. If they are realized as multiple unit capacitors in parallel, however, all of them will be affected equally, such that the overetching effect will cancel out. Since in our application precise

matching is very important, we make use of this principle.

There are various methods of designing capacitors in CMOS technology (Pelgrom [2017], Chen et al. [2017]). One approach is to make use of the gate capacitance of transistors. In the inversion mode there are charge carriers present in the channel underneath the gate, such that they form a capacitor with the charge at the gate contact. This way, the highest capacitance per unit area can be achieved. However, in order to exceed the threshold voltage, a significant turn-on voltage is required. Otherwise, the inversion layer would disappear.

So, another option pose metal-insulator-metal (MIM) capacitors. They are plate capacitors that are implemented over at least two metal layers with a special dielectric in between. This insulator has fantastic linearity properties, but it introduces extra cost, since it requires an additional mask. Moreover, the capacitance value of MIM capacitors is comparably low. And, even though the parasitic coupling to the substrate is low, unintentional fringe capacitance to components in the same layer will have a significant effect, if no wide spacing is applied between them.

Taking advantage of this fringe capacitance, another type of capacitor can be implemented: Metal-oxide-metal (MOM) capacitors mainly couple to wires in the same layer. In 65 nm technology the height of wires has exceeded its width, which makes it more efficient to use horizontal capacitors compared to vertical ones. Nonetheless, stacking multiple fringe capacitors on top of each other allows to increase their capacitance per area. In modern low power designs MOM capacitors are a popular choice, since they show good linearity and the effect of unintentional parasitics is low. We want to make use of these advantages for our application. So, let's take a more detailed look at different ways, how metal-oxide-metal capacitors can be implemented.

Figure 5.6 shows some commonly used capacitor structures. For reference, in contrast to the MOM capacitor illustrations, (a) depicts a MIM capacitor. It is apparent that its capacitance density is inferior, however, the top-terminal-to-substrate parasitic capacitance C^{TS} is very small. Subfigure (b) displays a MOM capacitor with interdigitated wires. Besides making use of the fringe capacitance within a layer, also the capacitance between adjacent layers is optimally used. A comparable approach is the woven MOM capacitor in subfigure (c). Here, the orientation between layers is alternated by 90°. On the one hand, this reduces the vertical capacitance slightly, but on the other hand, it allows an easier routing, since the terminals don't need to be interfaced at each layer. Nevertheless, in general the interdigitated and woven MOM capacitor are well-comparable. Furthermore, vias can play a bigger role in the capacitor

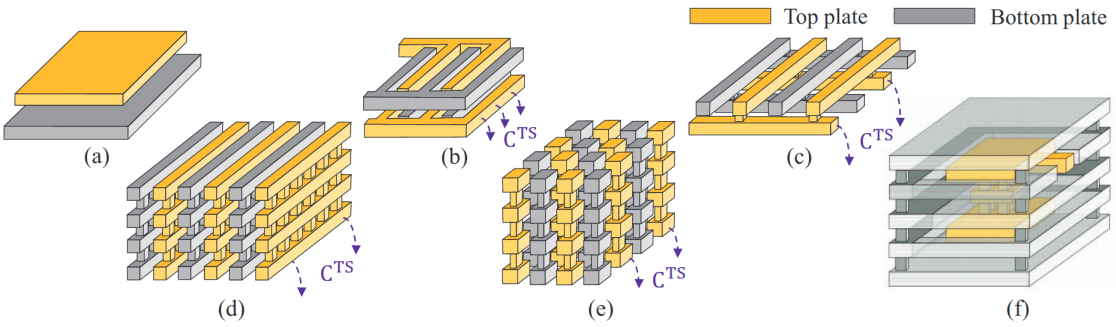


Figure 5.6.: Different approaches of realizing capacitors with metal layers. (a) MIM capacitors have a negligible top-terminal-to-substrate parasitic capacitance, but only a low capacitance density, (b) Interdigitated and (c) woven wires mainly utilize the fringe capacitance to other components in the same layer. This way their height is used more efficiently, however, they have a larger parasitic capacitance, too. (d) Parallel stacked wires and (e) vertical bars utilize vias in addition, which allow to achieve a higher capacitance density. To minimize the parasitic capacitance, a (f) multi-layer sandwich capacitor can be used. Figure taken from Chen et al. [2017].

design. In (d) a MOM capacitor is formed from parallel stacked wires that are almost like vertical plates. Due to the thinner separation in the lateral direction for advancing process technologies, a higher capacitance density can be achieved like this. Beyond that, in subfigure (e) the stacked wires are exchanged for vertical bars, allowing to extent the fringe capacitance in both horizontal directions. This allows to further increase the capacitance per area. However, now even more contacts are necessary to interface the terminals. All of these designs show a non-negligible parasitic capacitance of the top terminal to the substrate. A design offering a solution for this is the multi-layer sandwich capacitor in subfigure (f). Here, the bottom terminal almost completely encloses the top terminal, significantly reducing its parasitic substrate capacitance. This comes at the cost of a reduced density and, also, the routing complexity isn't alleviated.

After considering the advantages and disadvantages of the different designs, we decide on using the woven MOM capacitor structure from figure 5.6 for our implementation. On the one hand, we have seen in simulation that parasitic capacitance of 20% keeps the linearity of our implementation acceptable, meaning it is not indispensable to utilize the multi-layer sandwich capacitor. On the other hand, we don't want to interrupt the shielding of the capacitors at too many places. This leads us to the woven capacitor structure as it is the easiest to interface. To keep the unit capacitance small, but at the same time not risking mismatch to have a significant effect on the conversion, we decided for the smallest woven MOM capacitor offered by the vendor. It has a

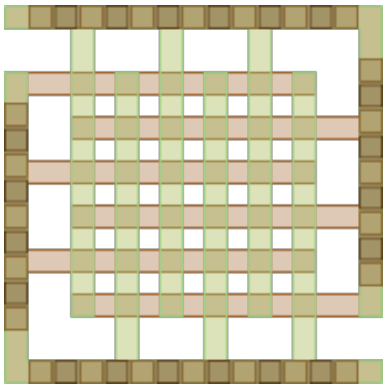


Figure 5.7.: Layout of the unit capacitor. It uses the fringe capacitance on metal five to seven between the fingers of the its two terminals. The layers are connected by vias in the half frames, allowing to contact the terminals on just a single layer.

unit capacitance of approximately $C_\mathrm{u} = 2.4 \cdot 10^{-15}\,\mathrm{F}$ and is illustrated in figure 5.7. Applying the margin of the cell symmetrical around the MOM capacitor, we can achieve an overall area of the capacitor array close to our target specification.

5.2.3. Capacitor column

By making use of unit capacitors the chances of mismatch due to overetching can be minimized. However, in designs where precise matching is crucial, furthermore the gradient of the oxide thickness should be taken into account, which typically occurs in the production process (Carusone et al. [2013]). It causes slightly variations between components that are placed at different sides of a die. To counteract this effect, a common centroid layout can be applied. Concretely, this means that the bit capacitors are split and arranged in a way, such that their centers of area are as close as possible. Thus, the gradient will affect all bits likewise. Figure 5.8 depicts the chosen configuration for one of the two 7-bit capacitor arrays. For realizing the full ADC column as depicted in figure 5.1, two of these arrays are placed vertically adjacent to each other. In addition, grounded dummy capacitors are placed at the outer rows for symmetry reasons. Optimizing strictly for the common centroid, some placements could be improved. However, we also considered a simple interfacing of the capacitors, leading to this arrangement.

On metal eight, there are vertical stripes shielding the capacitors. Over each unit capacitor, one stripe with ground and one with supply voltage are placed next to each other with minimum spacing. They directly contact the supply lines on metal nine. On metal four, the same shielding is applied horizontally. The potentials are transmitted

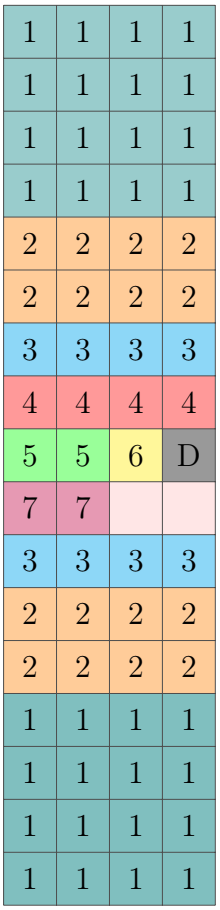


Figure 5.8.: Arrangement of the unit capacitors of a 7-bit DAC. It is chosen to combine a common centroid approach with an easy interfacing of the capacitors. The capacitors are labeled ascending from the MSB. The two unit capacitors mirroring the LSB configuration don't have a label. One shorted dummy capacitor is included for implementing the 67 unit capacitors in a 4×17 grid.

through via stacks running at the corners, where four unit capacitors meet (see figure 5.5). From the ground and power stripes on metal four, the underlying layers receive their static potentials. Now, wanting to connect to the capacitors, the shielding needs to be passed. Hence, it is not possible to contact the terminals directly from below, but gaps in the shielding between rows of capacitors are used. In the center there runs the line of the top terminals, while the bottoms terminals are connected through the spaces between the first and second, as well as the third and fourth column.

The potential of the bottom terminals is set by switches that are placed in the outer regions of the ADC column. They are implemented as inverters that are located directly behind buffers to ensure steep edges of the signals, since the longest connection is approximately 100µm. The corresponding signals are forwarded by wires running

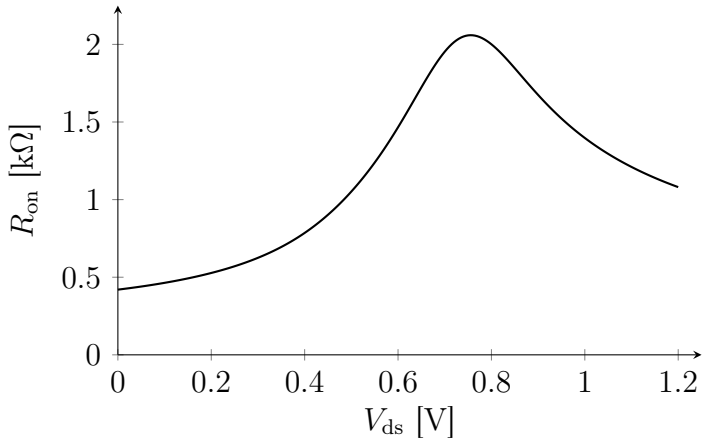


Figure 5.9.: On-resistance of the implemented transmission gates. For high and small drain-source voltages the PMOS and NMOS is conducting, respectively. Thus, in the transition range the resistance is the highest.

vertically on metal two. To minimize the effect of cross-talk, signals belonging to the two 7-bit arrays are arranged in an alternating fashion. The connections of the top terminals are controlled by transmission gates. They allow to connect the capacitors either to the input voltage, the comparator or to each other. In all cases, the inverse signal is produced locally by a buffer, on the one hand, to increase its slope and, on the other hand, to reduce the number of control wires running over large distances. The transmission gate itself uses equally sized transistors with a width of 1.3µm and minimum length, such that the charges injected due to switching are compensated. These dimensions were determined from simulations given typical conditions. Its on-resistance is plotted in figure 5.9 for varying drain voltages, while its source in held at the supply. The curve indicates the conducting states of the individual transistors. For small drain-source voltages the current flows through the NMOS transistor, but the higher V_{ds}, the higher becomes the resistance. The opposite holds true for the PMOS transistor. Thus, in the transition region, where the gate-source voltage of both transistors is the lowest, the total resistance of the transmission gate has a maximum. Since N- and PMOS possess different electrical properties, the curve is asymmetrical.

In the 8-bit case, both arrays are operated together at one input side of the comparator. For faster settling times, an additional transmission gate connects them directly. It is supposed to be conducting exactly in the case, when both arrays are connected to the comparator. Therefore, an AND-gate is used. In the physical implementation it is placed in the outer region, where also the switches controlling the bottom terminals are located. Furthermore, the reset of the DACs is controlled by NMOS transistors with the same width as in the transmission gates. They are located besides the bottom terminal switches, too. On the contrary, the transmission gates are placed in the

central region, in between the reference DACs of the comparator. These use equally sized transmission gates for presenting the same capacitive load at both input sides of the comparator. The complete arrangement is shown in figure 5.5.

5.3. Performance evaluation

Finally, we analyze the performance of the full ADC. For this, we start by considering different operation conditions, varying in their supply voltages, process corners and temperatures. While this is carried out on a transistor-level, in a next step the layout-dependent effects obtained by a parasitic extraction are taken into account, too. Lastly, the energy consumption of the ADC is examined. All these simulations focus on the analog part of the implementation and treat the digital control unit stated in a hardware description language. Further simulations including a synthesized digital controller can be viewed in Dauer [2022].

5.3.1. Transistor-level simulations

To get an impression of the performance of the full analog-to-digital converter, firstly, we evaluate its behavior on a transistor-level. This means that on the one hand, physical effects in transistors making them non-ideal switches are considered. But on the other hand, wire resistances and coupling to other nets is not, since no layout information are present, yet. The digital control unit is only considered at a behavioral-level. For simulating a design, mainly, there are three aspects that need to be contemplated: variations in the supply voltage, different process corners and varying operation temperatures.

In our application the supply voltage amounts to 1.2 V. However, it might be the case that the chip needs to be operated at a slightly shifted potential. Thus, in addition to 1.2 V, we also test how the performance is influenced, if the supply voltage varies from its ideal value by 10%.

Furthermore, different process corners, i.e. variations of fabrication parameters applying an integrated circuit design to a semiconductor wafer, are considered. Mainly, the doping concentrations in transistors on silicon wafers might vary significantly from their nominal values, affecting the mobility of charge carriers. Thus, N- and P-channel MOSFETs can behave faster or slower than in the typical case. The different process corners are indicated by firstly stating the NMOS and secondly the PMOS corner. Consequently, besides the typical-typical (tt) case, there are four corners: fast-fast (ff), fast-slow (fs), slow-fast (sf) and slow-slow (ss). Before the production, it should be ensured that for every corner the design works, even if this means restricting the operation

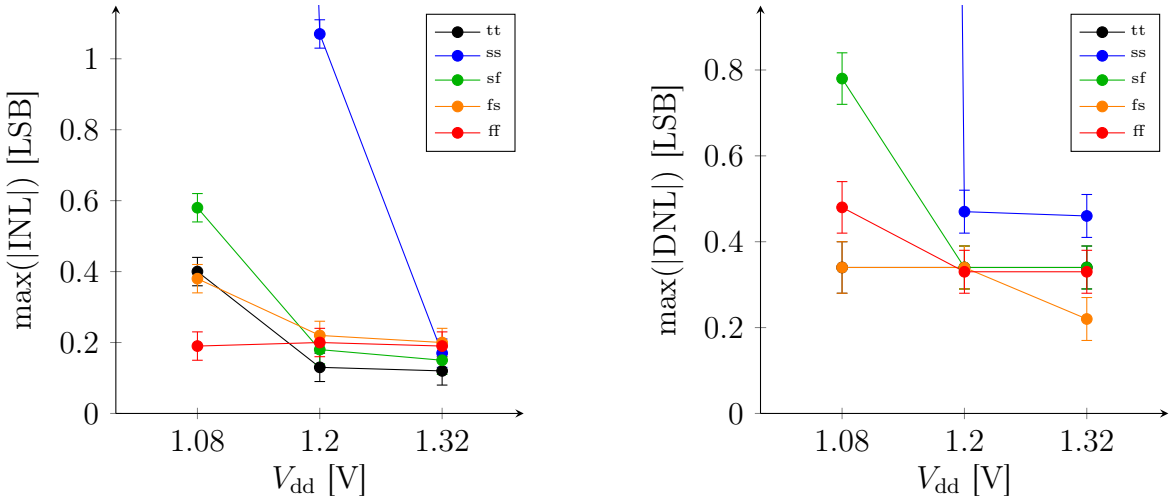


Figure 5.10.: Non-linearities in the 7-bit mode for different process corners and supply voltages. If the supply is decreased, the input won't be sampled quick enough, leading to linearity errors. The same can be observed with increased severity in the slow-slow corner. If both occurs simultaneously, the operation of the ADC will fail. In order to avoid this, the frequency needs to be reduced then.

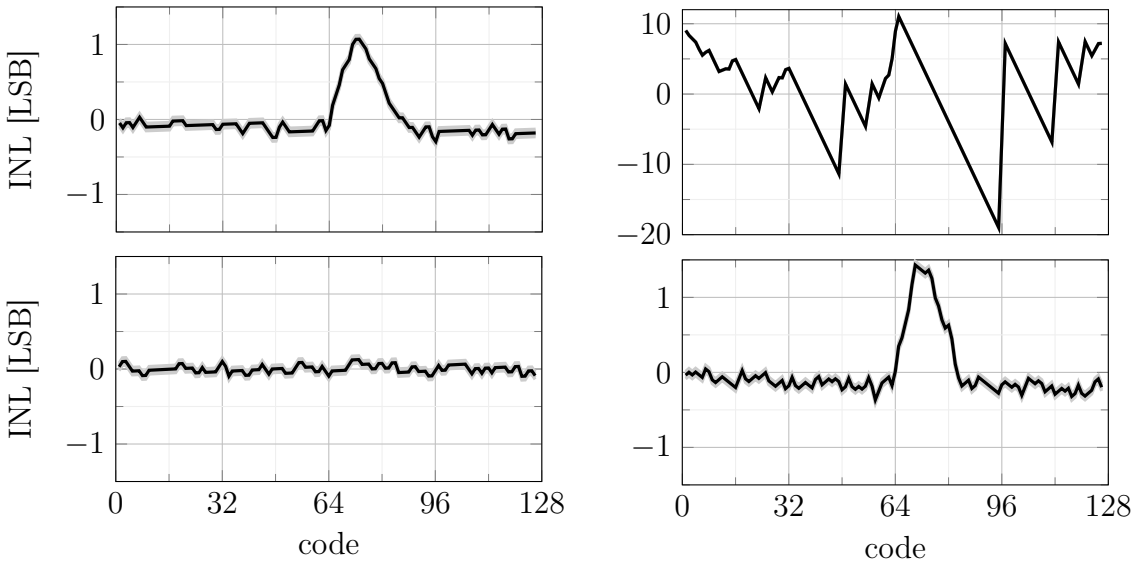
conditions, e.g. adjusting the speed.

Lastly, also the temperature has an impact on the performance, since it influences electric properties like the mobility of charge carriers and the resistivity. Therefore, we analyze the ADC's behavior for different temperatures starting at 10°C and ranging up to 70°C.

Firstly, let's take a look at the behavior of the 7-bit mode. The simulations were performed using input voltage steps corresponding to a precision of 10-bit, causing a non-disappearing uncertainty of the differential and integral non-linearity. In the typical case at 50°C and 1.2 V supply, a few non-monotonicities are visible (shown in figure A.1). Analyzing the analog traces, it can be seen that even though the input voltages of the comparator behave as expected, it makes an incorrect decision (see figure A.2 and A.3). Here, a further examination of the comparator is necessary.

For analyzing various operation conditions, the absolute maximum integral and differential non-linearities are depicted in figure 5.10 for different process corners and supply voltages at a constant temperature of 50°C. For 1.2 V and 1.32 V the behavior looks very similar for all cases. Only in the ss-corner, the INL increases significantly. In all other corners, except for the ff-corner this happens, too, if the supply voltage is reduced. Here, in the ss-corner the ADC stops to function correctly.

In order to explain this behavior, the integral non-linearity needs to be analyzed



(a) Supply voltage of 1.2V

(b) Supply voltage of 1.08 V

Figure 5.11.: Integral non-linearity in the slow-slow corner at different supply voltages. The uncertainty is depicted in gray. (a) At the target sampling frequency of 125 MS/s a rise can be observed in the upper plot at the input voltage range corresponding to the maximum on-resistance of the transmission gate. This can be mitigated, as shown in the plot below, by reducing the frequency by half. (b) If the ADC is in the ss-corner and operated at a reduced supply voltage, it is dysfunctional. Reducing the frequency by half mitigates this effect, leading to a similar behavior as for the full supply voltage. A further reductions yields the typical behavior (not shown).

more closely. Figure 5.11 depicts the INL spectrum in the ss-corner for supply voltages of 1.2 V and 1.08 V. At the full supply voltage in subfigure 5.11a a rise is visible shortly after half of the output code range, signifying that the corresponding input voltages are assigned a lower value compared to the linear interpolation through all output codes. Again, analyzing the analog traces, we find that this is caused by a too high speed. For sampling the input voltage, a transmission gate needs to be passed. However, as we have seen in figure 5.9 its resistance is the highest at the transition between the inversion modes of the NMOS and PMOS transistors. As a consequence, the input voltages corresponding to this range are not completely sampled onto the capacitors in the given time. Thus, an output value slightly lower is determined by the conversion, causing the increase of the INL. Nevertheless, this can be fixed by reducing the clock frequency. The lower plots show the integral non-linearity at only half of the initial rate. Now, the behavior is equivalent to the typical case.

In subfigure 5.11b the supply voltage is reduced to 1.08 V. This slows down the analog-to-digital converter in addition, since the gate-source voltage of the transistors

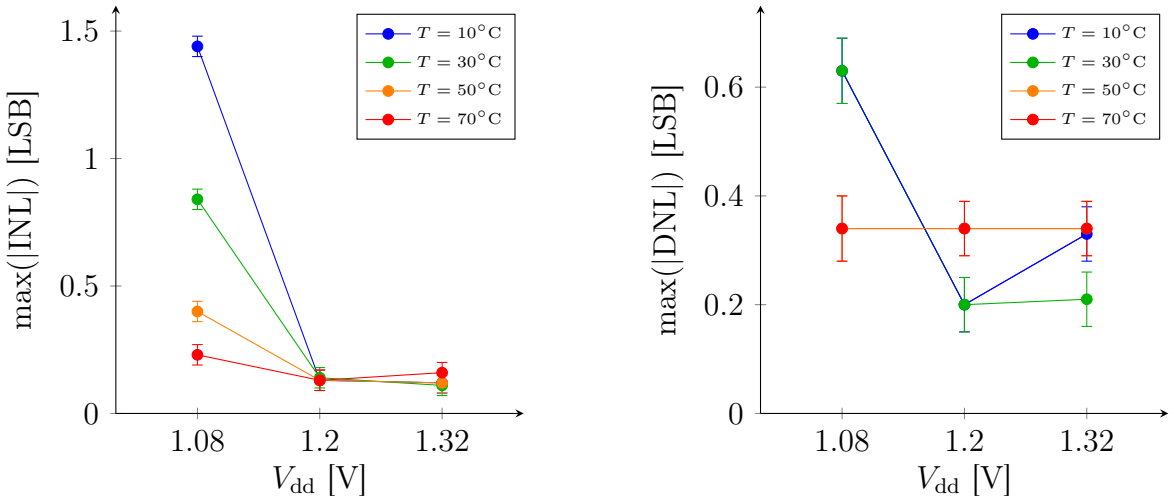


Figure 5.12.: Temperature dependence of the non-linearities in the 7-bit mode. Lower temperatures lead to a reduced mobility of the charge carriers, effecting linearity errors in combination with a reduced supply voltage.

in reduced. Consequently, in the slow-slow corner the conversion is impaired more severely, leading to fatal defects. But, again the situation can be eased providing more time. Reducing the clock frequency by half, a behavior similar to the case with full supply voltage can be observed. Further decreasing yields the typical linearity.

The differential non-linearities don't show as large deviations compared to the integral non-linearities, excluding the dysfunctional case of slow-slow at 1.08 V. At this reduced supply voltage, also the DNL in other corners increases. Here, in all cases, as expected the maximum DNL is present at the MSB code transition.

Next, we consider the effect of different operation temperatures, assuming the tt-case. The simulation results are visualized in figure 5.12. Again, the integral non-linearity looks very similar for a supply voltage of 1.2 V and 1.32 V. At 1.08 V the temperature dependency of the mobility of the charge carriers is well visible. Here, the INL spectrum looks qualitatively like in the slow-slow case, indicating an insufficient sampling time. This effect aggravates with decreasing temperature. However, it doesn't destruct the functioning of the ADC, since the solution is known to us: Reducing the speed again mitigates the linearity errors.

Moreover, the differential non-linearity at 50°C and higher is constant for all tested supply voltages. It only shows fluctuations at 30°C and lower. Having said that, it should be noted that, due to the uncertainty introduced by the limited resolution of input voltage steps, the significance of these deviations for higher supply voltages is low.

Let's continue evaluating the behavior of the 8-bit mode. Figure 5.13 illustrates

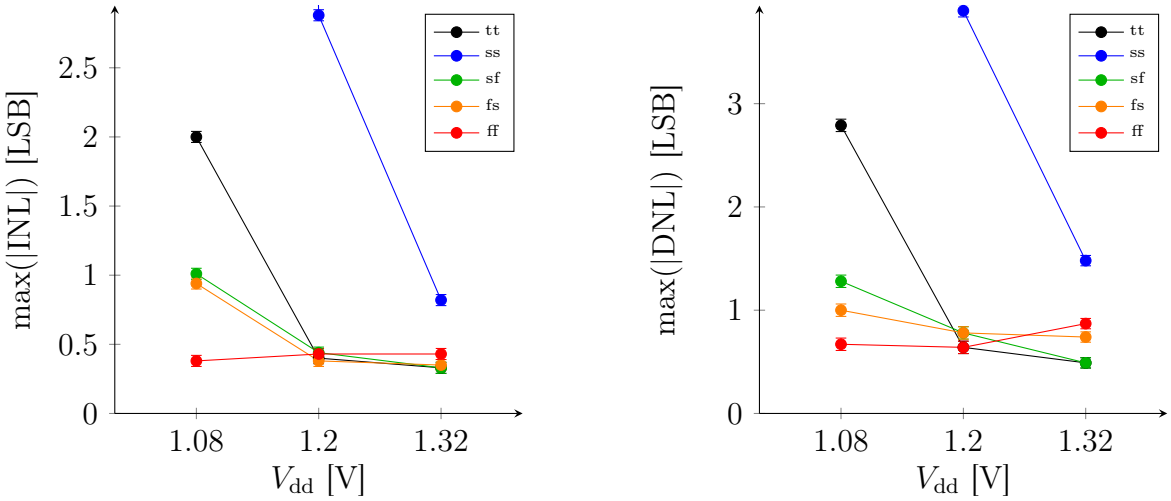


Figure 5.13.: Non-linearities in the 8-bit mode for different process corners and supply voltages. The effect of a decreased supply voltage and slow transistors is more severe than in the 7-bit mode, since not only the sampling is affected, but the voltages also require more time to settle at both DACs now.

the non-linearities at a constant temperature of 50°C for varying supply voltages and process corners. Like before, the behavior at a supply voltage of 1.2 V and 1.32 V is very similar. At 1.08 V, however, a difference to the 7-bit mode is the magnitude of the errors. The rise of the INL is higher already in the tt-configuration. Consequently, this effect aggravates in the ss-corner. However, the maximum INL and DNL now are present at the second bit decision, as shown in figure 5.14. Actually, this is not surprising, remembering the working principle of the 8-bit mode: The MSB decision is made directly after sampling. Then, the largest capacitor of one of the two 7-bit

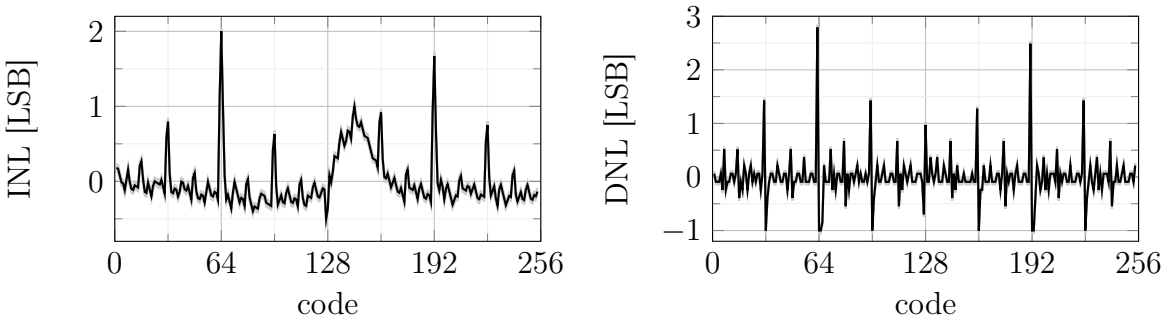


Figure 5.14.: Non-linearities in the typical case for a reduced supply voltage of 1.08 V with uncertainties in gray. Besides the rise in the center, indicating an insufficient sampling time, the linearity is affect at the code transition, where the largest capacitor is switched. Here, the voltages between both capacitive arrays and the comparator doesn't settle quick enough, before the following decision.

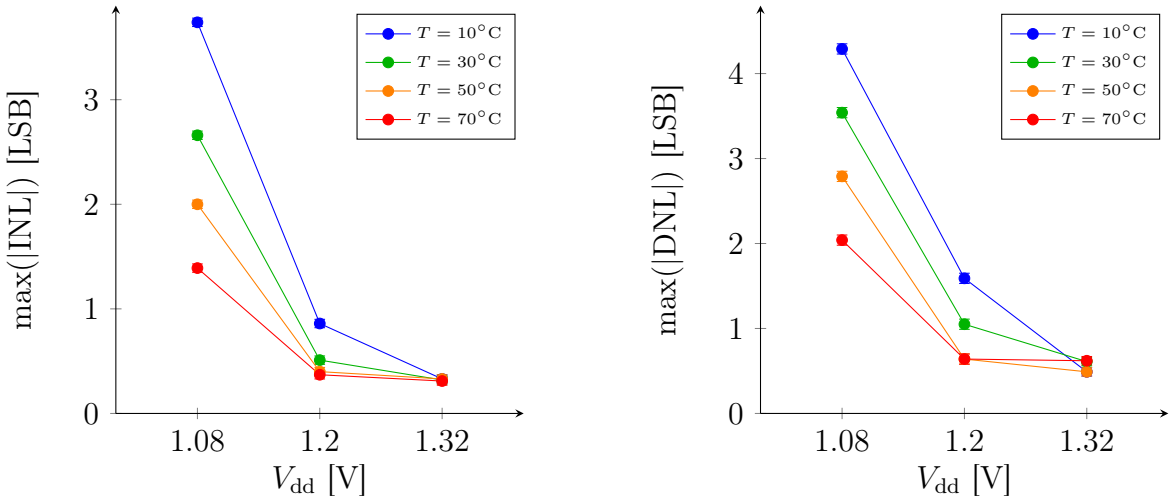
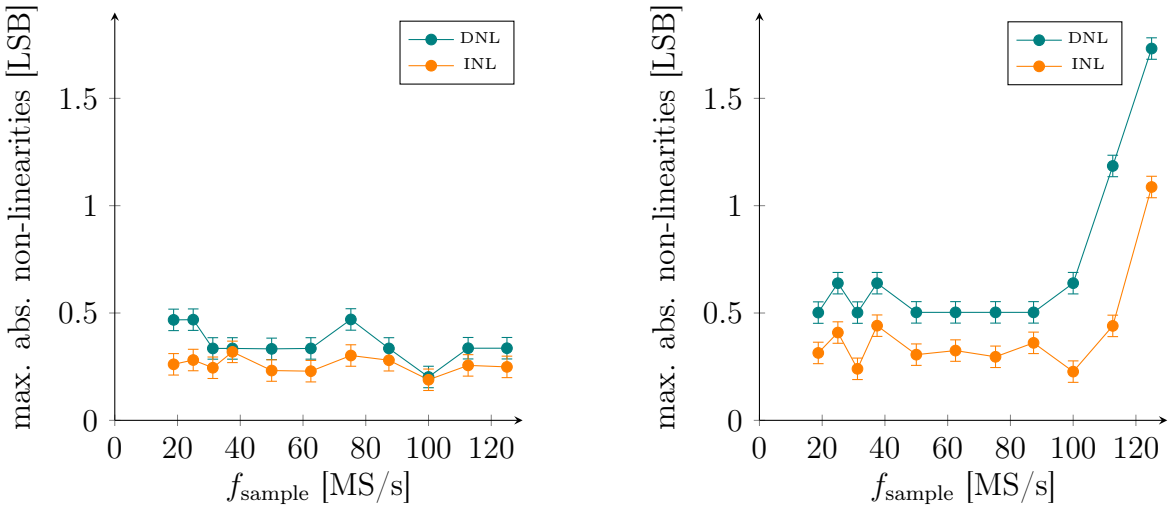


Figure 5.15.: Temperature dependence of the non-linearities in the 8-bit mode. Again, the reduced mobility at lower temperatures has an aggravated effect compared to the 7-bit mode.

arrays is switched. The newly configured potential needs to settle at both DACs. This is achieved by the two transmission gates, connecting the arrays to the comparator, and an additional one, shorting the capacitive arrays directly. Even though the latter was added with the purpose of achieving faster settling times, it is not sufficient in the case with reduced supply voltage. Hence, at the second bit decision the voltage at the comparator doesn't have sufficient time to settle. Consequently, this leads to a high DNL, INL and even missing codes. Thus, in the 8-bit case choosing the appropriate speed is even more critical than in the 7-bit case.

Lastly, the operation temperature is varied again. Figure 5.15 depicts clearly the inverse relationship between temperature and non-linearities. Furthermore, in contrast to 7-bit case, an increased INL and DNL is already visible for a supply voltage of 1.2 V at 30°C and lower.

Concluding, we find that the chosen implementation doesn't perform ideally given all possible operation conditions at the target frequency. Thus, we analyze how the choice of clock speed influences the conversion. Figure 5.16 visualizes the maximum non-linearities for different sampling frequencies. In the 7-bit mode, the maximum INL is not affected significantly and also the DNL always stays below 0.5 LSB. However, it seems to have an optimal operation frequency of 100 MS/s, since both linearity measures are at their minimum value there. Having said that, the significance of this deviation is not high, due to the limited input resolution. Furthermore, the effect of insufficient settling times leading to non-monotonicities can be observed in the output codes above 100 MS/s. In addition, the number of non-monotonic transitions increases



(a) 7-bit mode

(b) 8-bit mode

Figure 5.16.: Non-linearity dependence on the sampling frequency. If the frequency is chosen too high, the voltages won't have sufficient time to settle. If it is chosen too low, however, the reference voltage needs to be refreshed more often, in order to avoid errors.

again at 50 MS/s and lower. This is explained by the voltage drop of the reference DAC. In order to keep a nearly constant voltage, it is necessary to refresh the reference DAC regularly. In the current implementation, this is done after a fixed number of clock cycles. If the frequency is reduced, however, the time between refreshings increases. This causes the reference voltage to decrease by a larger amount, during the same number of conversions. Hence, if the ADC is operated at lower frequencies, the period between refreshings of the reference DAC needs to be adjusted, too. Then, these non-monotonic effects can be mitigated.

In the 8-bit mode a clear rise of both DNL and INL is observable for a sampling frequency of 100 MS/s and higher. Starting at approximately 90 MS/s, the voltage at the DACs and comparator input has enough time to settle, after the first capacitor is switched. Here, the maximum non-linearities occur at the MSB transition. Again, at lower frequencies the decrease of the reference voltage can be observed in the monotonicity. Concluding, like for the 7-bit mode the configuration of the reference DAC needs to be adjusted to the clock frequency.

This insight is taken into consideration for the following simulations, where parasitic effects introduced by the layout are examined. More thorough deliberations, about what needs to be adjusted in order to enhance the performance, will be discussed afterwards.

5.3.2. Parasitic extraction simulations

So far, we have performed all simulations using the schematic transistor-level representation of our design. For the physical implementation, however, it is necessary to determine a placement for all components and connections, such that it can be fabricated. This is realized in the layout. However, the placement introduces additional non-idealities, e.g. long wires have non-negligible resistances or closely routed nets couple to each other. Hence, it is necessary to perform further simulations considering the placement information, in order to examine the effects introduced by the layout. To do so, a parasitic extraction is conducted, providing an equivalent schematic circuit, that also represents the resistances and capacitances introduced by the chosen placement. From this we find, that the parasitic coupling from the DAC capacitors to ground amounts to approximately 23%. This is close to the value we considered before in chapter 3 to investigate the effect of parasitic capacitance on different capacitive arrays. Thus, we expect a similar behavior as observed in chapter 3 regarding non-linearities, as well as a narrowed conversion range.

The equivalent circuit obtained by the parasitic extraction, now, is used to evaluate the layout-aware behavior of the ADC in the typical case at 50°C. Since the number of components that are simulated increases significantly, so does the time to perform the simulation, which is why not all operation conditions are investigated individually. Nevertheless, we expect qualitatively similar results compared to the previous section, if those circumstances are applied to the layout-implementation. In preliminary tests, we find that as expected the voltages require more time to settle, taking additional resistances and capacitances into account. Therefore, we choose operation frequencies lower as the optima determined in figure 5.16. Namely, for the 7-bit mode a 500 MHz clock is used, corresponding to a sampling frequency of 62.5 MS/s, and for the 8-bit mode the clock runs at 250 MHz, which is equivalent to a sampling frequency of approximately 15.6 MS/s.

The performance of the 7-bit mode is shown in figure 5.17. In the transfer curve a few non-monotonicities can be seen at irregular places. Mostly, they are directly at code transitions. However, the MSB transition is unaffected, indicating sufficient settling times. Having said that, we remember that the 7-bit mode works with two time-interleaved digital-to-analog converters. As depicted in figure 5.5, the components underneath the capacitive array are not completely symmetrical. Thus, it is possible that in spite of the shielding between the capacitors and the underlying components, the behavior of both DACs slightly varies. Then, at a transition point, one of the DACs already determines the higher code, while the other is still at the previous decision,

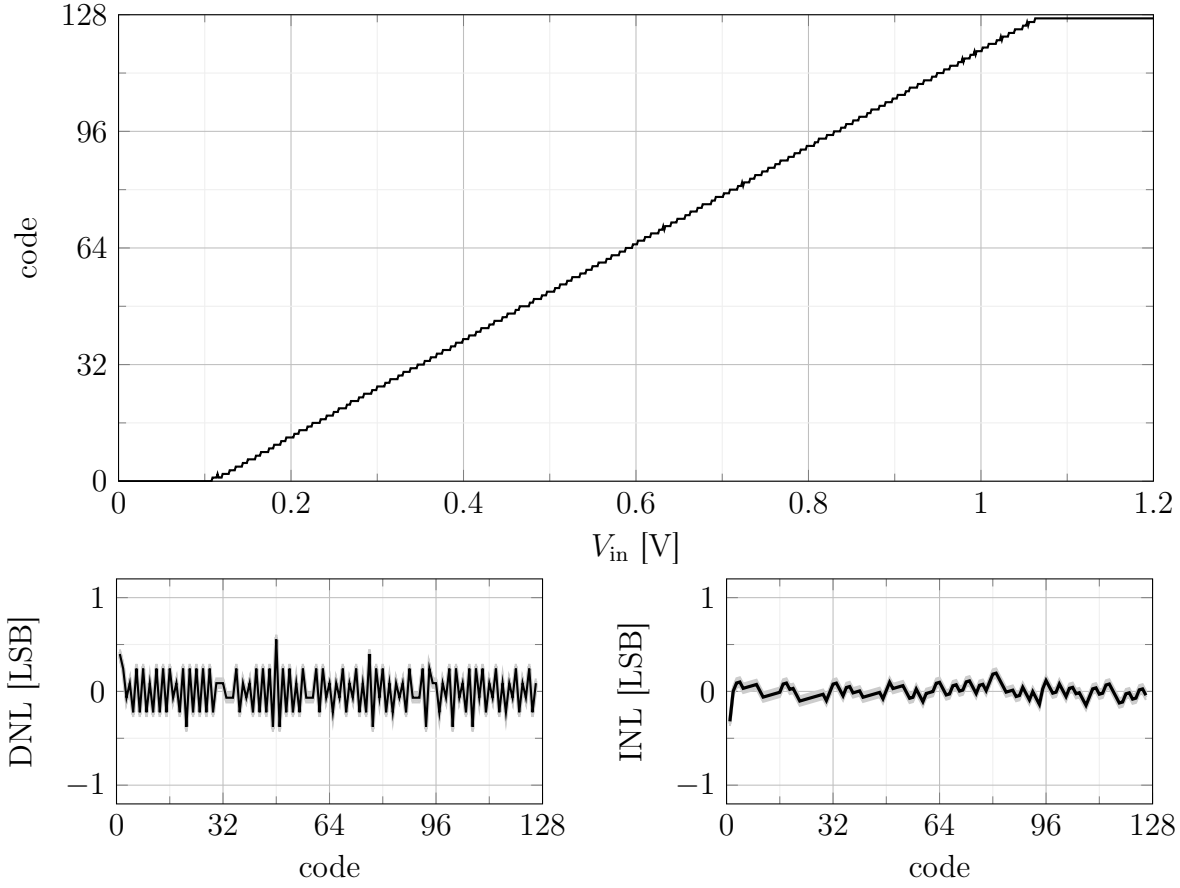


Figure 5.17.: Characteristic of the 7-bit mode with parasitic extraction and uncertainty in gray. Some non-monotonicities occur, due to time-interleaving and an insufficient refreshing of the reference voltage, which also affects the DNL. However, deviations of 0.2 LSB are expected there, since the nodes between the split LSB capacitors also carries a parasitic capacitance.

leading to a non-monotonicity. Indeed, looking at the arrays individually, all non-monotonicities at the transition points disappear.

Only the first one in the middle of the step of output code two remains. Analyzing the analog traces, we find that it occurs at the point in time, where the capacitive array of the reference DAC connected to the inverting input of the comparator is switched. Due to the kick-back of the comparator, some charge gets lost over time, leading to a decrease of the potential of the reference DAC. Thus, it is necessary to refresh its voltage after a certain amount of time, as discussed before. To do so without interrupting the conversion, the reference DAC changes between two capacitive arrays. One of them provides the voltage at the input of the comparator, while the other is given time to refresh. Hence, at the moment of switching, the newly connected array has a slightly higher potential. In figure A.4 and A.5 the difference can be determined to 6 mV. Since this is more than half of the voltage range of an LSB, $\frac{1}{2}V_{\text{LSB}} = 3.75\,\text{mV}$,

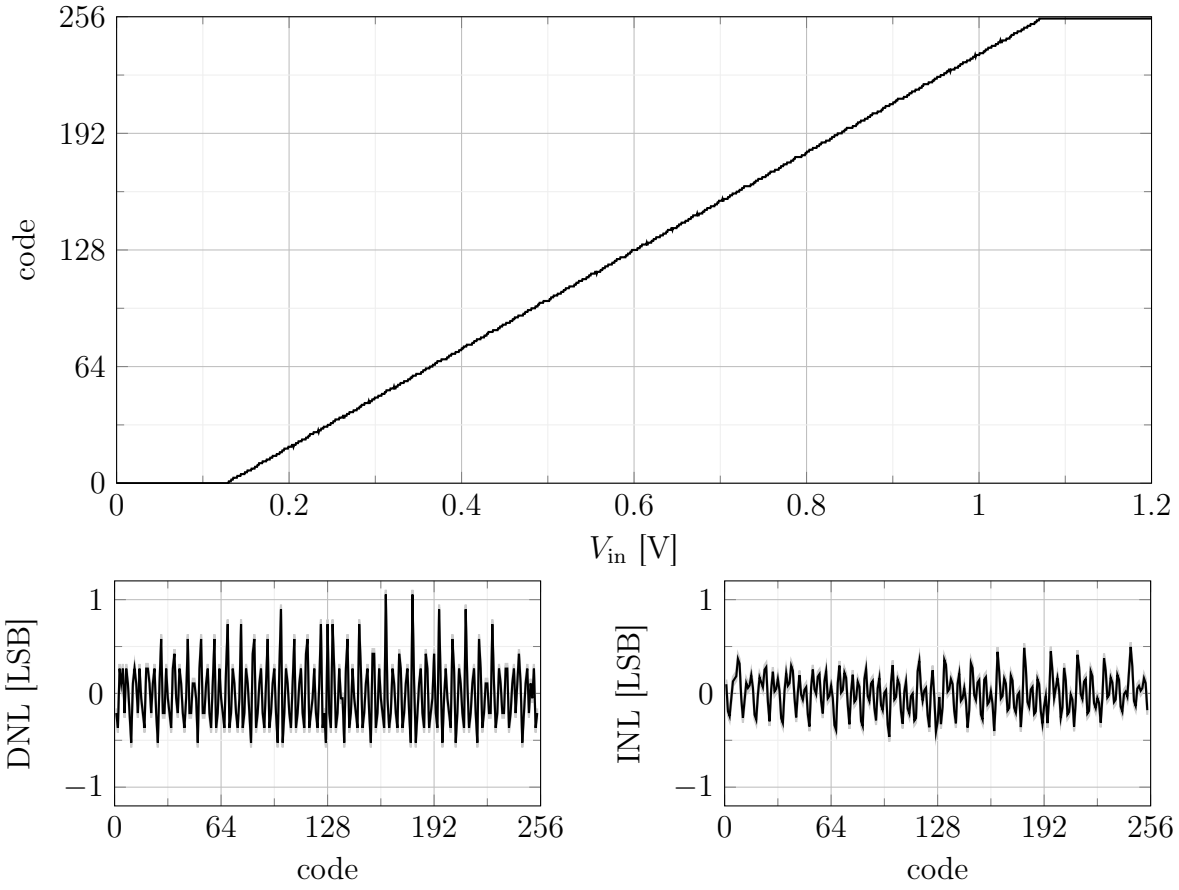


Figure 5.18.: Characteristic of the 8-bit mode with parasitic extraction and uncertainty in gray. The reference voltage drop has an even larger effect, since the time between refreshings is longer. This affects the DNL in addition to the parasitic capacitance of the split LSB capacitors.

the switching at the reference DAC can lead to non-monotonicities. Consequently, it is necessary to refresh the reference DAC sooner, such that the voltage drop stays below $\frac{1}{2}V_{\text{LSB}}$ and can't intrude the conversion.

This refreshing of the reference voltage also affects the step size, leading to deviations in the differential non-linearity. The regular jags around 0.2 LSB, however, are known to us from the previous analysis of the split LSB capacitor in section 3. Apart from that, the integral non-linearity doesn't show significant deviations. Since no rise after the MSB transition can be observed, the sampling time is sufficient, meaning the frequency is chosen adequately.

The performance of the 8-bit mode is shown in figure 5.18. Again, a few non-monotonicities can be seen at code transitions, however only in certain regions, reflecting asymmetries in the layout. The differential non-linearity mainly shows two components: On the one hand, large deviations can be observed in regular spacings.

Even though they remind of insufficient settling times between the DACs and the comparator, this isn't the predominant effect. Otherwise the DNL error would be the highest at output code 64 and 192. However, the spikes can be seen at closer spacings with varying heights. The reason, again, is the refreshing of the reference voltage, which happens at regular time steps. Since the frequency in this simulation is lower, but the number of clock cycles between refreshings is not adjusted, the effect is more severe than before. Nevertheless, again this is solved by adjusting the time between refreshings. On the other hand, the differential non-linearity shows smaller components again around 0.2 LSB. They occur like in the 7-bit mode, due to the parasitic capacitances between the split LSB capacitors and can't be mitigated by adjusting the frequency or other operation conditions. The integral non-linearity reflects both these effects. Again, no rise at the MSB transition is visible, indicating sufficient settling times.

5.3.3. Energy consumption

Lastly, we take a look at the energy consumption of the analog-to-digital converter. For this, we observe the current flow during the conversion of different input voltages. In figure 5.19, inputs from 0 V to 1.2 V are converted subsequently with a 7-bit precision and the currents of the different components are plotted. We see that for the first decisions of the comparator its current is lower, but when both input sides reach similar potentials, the current for every triggering has an almost constant amount. For the digital-to-analog converter the situation is different: All capacitors are switched once during the conversion in descending order for bit testing. Depending on the decision of the comparator, they are either switched back directly or at the end of the conversion. An exception presents the MSB capacitor, due to its different configuration during the sampling phase, as discussed in section 4.2.2. It is only switched for input voltages higher than $\frac{1}{2}V_{\text{ref}}$. Concluding, the largest amount of current flows in the moment all switches need to be reset, i.e. after a conversion of 0.6 V in the 7-bit mode or at the end of a conversion in the 8-bit mode. Since this happens in a short period of time, the transient current flow needs to be watched.

Changing a large amount of current quickly might cause a voltage drop of the supply. The amount by which the supply voltage decreases is coupled to the current via the capacitance that is applied to it. Approximating the transient to be constant over a short period of time Δt, the change in the supply voltage can be determined by

$$\Delta U = \frac{1}{2C_{\text{tot}}} \cdot \Delta t^2 \cdot \frac{dI}{dt}. \tag{5.1}$$

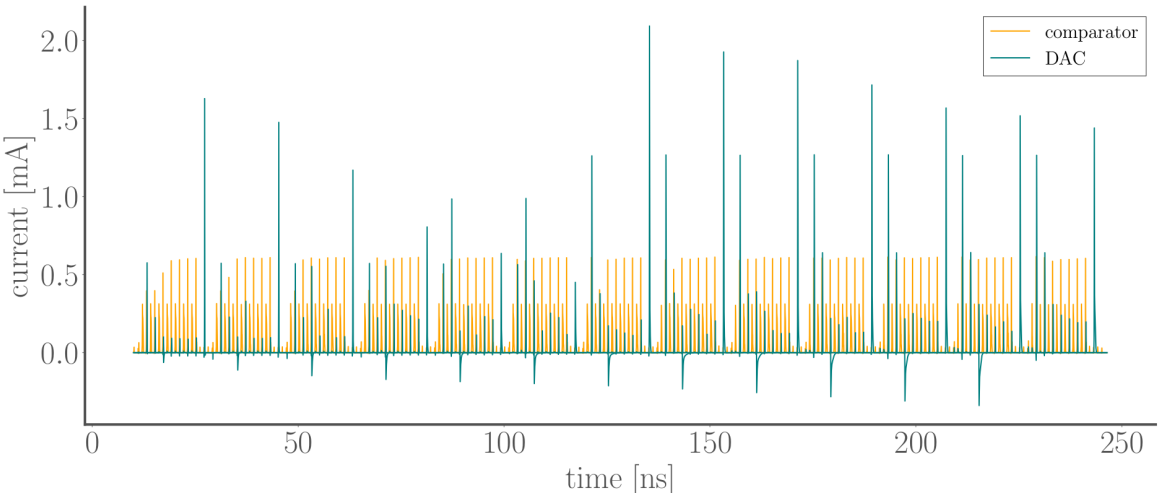


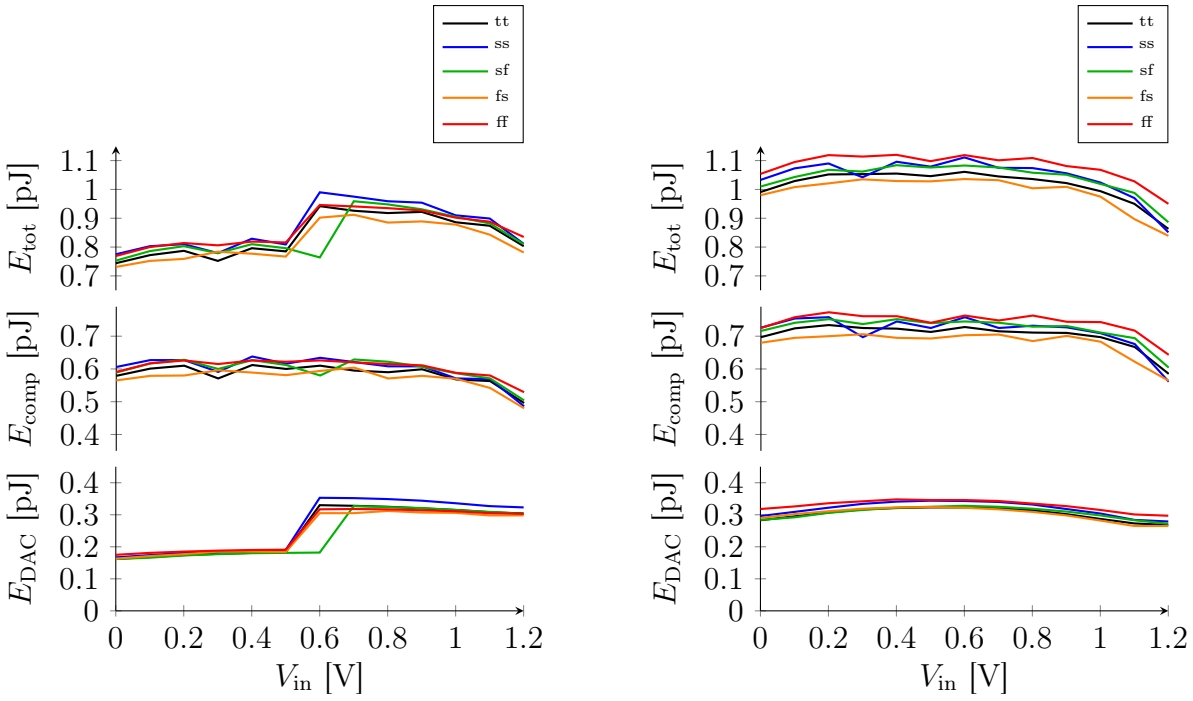
Figure 5.19.: Current flow for the 7-bit conversion of input voltages from 0 V to 1.2 V. The currents of the comparator reach an almost constant level, as soon as its inputs are close to each other. For the DAC the times at which current flows depend on the decision of the comparator. The highest amount can be observed, when all capacitors are switched simultaneously after converting 0.6 V.

Using simulation we find that if all 512 channels of a chip convert simultaneously, in the worst case the supply voltage might drop by approximately 250 mV. In order to reduce this to a negligible amount, i.e. below $\frac{1}{2}V_{\mathrm{LSB}}$, a supplementary capacitance $C = \mathcal{O}(10^{-9})\,\mathrm{F}$ needs to be added. This can either be realized by other components on chip, e.g. inherent parasitic capacitors or explicitly added MOS capacitors, or by external capacitors on the circuit board. In the latter case, however, the inductance of the bond wires needs to be taken into account, too. This also causes changes in the supply not to happen instantaneously, but they are delayed by its transient response, leading to impairments of the performance. In addition, if large capacitors are added on the chip, the bond wires need to be dimensioned, such that they don't burn during the activation.

Taking a look at the energy consumption again, this now can be determined by integrating the power over the conversion time.

$$E = \int_{t_{\mathrm{start}}}^{t_{\mathrm{end}}} V_{\mathrm{dd}} \cdot I \; dt \tag{5.2}$$

Figure 5.20 shows the results in dependence of the input voltage. In the 7-bit mode, an increased energy consumption for input voltages greater than 0.6 V is apparent. This is caused by the switching of the MSB capacitor as explained before, which is



(a) 7-bit mode

(b) 8-bit mode

Figure 5.20.: Energy consumption for both modes. In the 7-bit mode, the energy of the DAC depends significantly on the MSB decision. The sf-curve varies, since unlike in the other corners, at 0.6 V the MSB is still determined to be zero. On average, 30% of the total energy is consumed by the DAC.

in accordance with equation 4.7. The energy consumption of the comparator varies slightly for different input voltages, since it is dependent on the voltage difference at its inputs (Dauer [2022]). However, as shown in figure 5.19, after a few decisions it quickly settles at a constant amount. On average, the total energy for a conversion with 7-bit resolution determined like this amounts to 0.84 pJ.

In the 8-bit mode the behavior looks different. Here, the MSB decision doesn't affect the energy consumption of the DAC, according to equation 4.8. The comparator shows qualitatively the same behavior as before, only its absolute value now is slightly increased, since it must perform an additional decision. This leads on average to a total energy per conversion of 1.01 pJ.

In general, the DAC makes up approximately 30% of the conversion energy and the comparator 70%. The contribution of the reference generation for the comparator can be neglected, since it only amounts to 0.2 pJ per refreshing, which compared to the conversions happens very sparsely. Including the effect of parasitic capacitances, we find that the energy consumption increases by 60 - 70%. Thus, in the physical

implementation we expect an average conversion energy of 1.39 pJ and 1.64 pJ for the analog part in the 7- and 8-bit mode, respectively. In addition, the implementation of the digital control unit expectedly increases these values by a factor of 2.5 - 3 (Dauer [2022]).

5.4. Summary

To sum up, taking into account the effect of resistivities and parasitic capacitances, it is necessary to reduce the conversion speed of the analog-to-digital converter, in order to still obtain reasonable results. Hence, the 7-bit mode was simulated with a sampling frequency of 62.5 MS/s and the 8-bit mode with 15.6 MS/s. Their behavior regarding non-linearities and gain is summarized in table 5.2. Depending on the processing and operation conditions, further adjustments regarding the speed might need to be applied.

	7-bit mode	8-bit mode
max(DNL) [LSB]	0.55 ± 0.05	1.05 ± 0.05
min(DNL) [LSB]	-0.38 ± 0.05	-0.53 ± 0.05
max(INL) [LSB]	0.19 ± 0.05	0.50 ± 0.05
min(INL) [LSB]	-0.32 ± 0.05	-0.46 ± 0.05
gain [$\frac{\text{LSB}}{\text{V}}$]	132.6	269.7

Table 5.2.: Performance of the fast 7-bit and precise 8-bit mode. The results were obtained performing a layout-aware simulation with $V_{\text{dd}} = 1.2\,\text{V}$ at a temperature of 50°C, assuming typical process conditions. The uncertainties are caused by the limited resolution of the input voltage steps.

Furthermore, it needs to be mentioned that in both modes non-monotonicities are present, due to impairments of the comparator and its reference voltage. For compensating mismatch, it is possible to configure the offset of the comparator, by the means of the reference DAC. Thus, a settable input voltage range of nearly 1 V can be achieved in both modes. The analog energy consumption per conversion was determined to 1.39 pJ and 1.64 pJ in the fast 7-bit and precise 8-bit mode, respectively. In the former, however, it strongly depends on the MSB decision. Considering the determined sampling frequencies, this yields a power of 86.9 µW in the time-interleaved mode and 25.6 µW in the mode with increased resolution. Since both modes make use of the same components, their area is unvaried. Table 5.3 comprises an overview of the results found by simulation with respect to the initial target values.

	fast mode		precise mode	
	targets	results	targets	results
resolution [bits]	6	7	8	8
sampling frequency [MS/s]	125	62.5	62.5	15.6
input range [V]	0.6 - 1.2	0.97 (settable)	0 - 1.2	0.95 (settable)
energy per conversion [pJ]	1.56	1.39	3.12	1.64
power per channel [µW]	500	86.9	500	25.6
area per channel [μm^2]	1176	1317	1176	1317

Table 5.3.: Overview of the targets and results obtained by a layout-aware simulation. The goals regarding resolution and input range of the fast mode are actually overfulfilled. The energy per conversion and area demand are close to their desired value. Note, however, that so far only the analog part is considered. Due to performance impairments, the sampling frequencies stay below their aspired levels.

6. Discussion

In this work a successive approximation register analog-to-digital converter was implemented, providing two modes of operation. On the one hand, a fast mode with 7-bit resolution and a sampling frequency of 62.5 MS/s was designed, that uses two time-interleaved DACs. On the other hand, both of them can be operated jointly, achieving an 8-bit precision at a reduced sampling frequency of 15.6 MS/s. Both modes cover an input range of approximately 1 V, which can be configured using a reference DAC. Thus, the initially set targets for the fast mode concerning the resolution and input range are overfulfilled.

Taking a look at the area and energy consumption, we are close to the aspired values. However, so far only the analog part was considered. Synthesizing the digital control logic, an additional area demand of nearly 500 μm^2 and an increase in energy of a factor 2.5 - 3 is expected, depending on the mode (Dauer [2022]).

Furthermore, most importantly, the ADC must operate at a reduced sampling frequency than desired, due to impairments of the performance. If the sampling period or, moreover for the 8-bit mode the settling time between the DACs and comparator after a decision, is insufficient, errors in the differential and integral non-linearity will be visible. But, even if these effects are remedied by reducing the clock frequency, there are non-monotonicities that remain. These must be eliminated, in order to avoid errors and ensure a reliable conversion. From analyzing the analog traces, we found that they appear, on the one hand, at transition points, due to time-interleaving. Taking into account parasitic capacitances introduced by the layout, there are input voltages, at which the conversion with one DAC already determines the higher output code, whereas the following conversion with the other DAC yields the previous result. In order to avoid this effect, it is necessary to not only make the layout of the digital-to-analog converters completely symmetrical, but also the underlying components. Since the capacitors are shielded by a layer with static potentials, it was assumed that slight variations in the structures underneath wouldn't have a significant effect. However, this was found not to be the case. Hence, it is necessary to adjust the layout accordingly.

On the other hand, a reason leading to non-monotonicities is the reference DAC. It provides half of the supply voltage for comparison. However, some of its charge gets

lost over time, due to the kick-back of the comparator. Consequently, the reference voltage decreases slightly. To avoid this effect from intruding the conversion, it needs to be refreshed regularly. The intervals for this must be chosen, such that the voltage drop stays beneath the half of an LSB step and, thus, is undetectable for the ADC. Expectedly, doubling the refreshing rate already mitigates this effect significantly.

After the monotonicity of the analog-to-digital converter is ensured, adjustments can be applied improving the maximum sampling frequencies of both modes. In the 7-bit mode, the limiting factor was found to be an insufficient sampling time, causing a rise of the INL at input voltages, that resemble the maximum on-resistance range of the implemented transmission gate. One approach to mitigate this effect is to increase the relative sampling period. In the 8-bit mode, it can be observed that a single clock cycle is sufficient for the reset. This can be transferred to the fast mode to gain an additional clock period for sampling, where currently two cycles are used for the reset. However, this is only a slight improvement and neither makes up for a factor of two in the sampling frequency, nor relieves the 8-bit mode. Hence, an adjustment of the transmission gate is necessary. It was dimensioned to perform well in simulations under typical conditions. However, also unfavorable process and operation conditions need to be taken into account, as well as additional effects obtained by the parasitic extraction, in order to achieve a satisfactory performance in the physical implementation. Thus, the width of the transmission gate should be increased accordingly, decreasing its resistance. This needs to be done with all components, where an insufficient settling time can be observed, i.e. especially between the DACs in the 8-bit mode, too.

If these changes are implemented, the speed limiting factor will be the kick-back of the comparator. At every flank of its trigger signal, the transient responses show a short, steep rise of the input potentials, requiring a certain amount of time to regain their initial values. To mitigate this effect, either the total capacitance of the reference DAC can be increased or the sizes of the input transistors of the comparator can be adjusted. For this, the influence of the kick-back should be evaluated with respect to the available area, after the previous mentioned modifications have been applied.

In addition to the effects observed so far, further impairments will be introduced by the physical implementation of the digital control logic. This is, since the flanks of the control signals won't arrive instantaneously, as currently assumed in the performed simulations, but they'll show some wire delay. This could lead to a negative impact on the linearity of the conversion. In order to keep the delay low, interconnections should

be kept as short as possible. Furthermore, to avoid settling times from being negatively affected, an accurate timing between the control signals of the different components is necessary.

Lastly, all simulations performed in this work used the same procedure to evaluate the behavior of the ADC. Namely, the input voltage was gradually increased in small steps from the ground potential up to the full scale voltage. From this, the offset, gain and non-linearities were calculated. However, it might be possible that effects evolving over the operation time of the ADC are absorbed into the gain error or that the linearities behave differently for high-frequency inputs. Thus, it is necessary to analyze the behavior with regard to different input signals, too. Moreover, the operation conditions were assumed to be constant over time. Presumably, however, also dynamic variations will have an impact on the conversion. Therefore, further analyses given time-varying conditions should be conducted, in addition.

7. Outlook

We have seen from the first design iteration that the developed dual-mode conversion method in principle works, however, some adjustments need to be made, in order to enhance its performance. Mainly, the layout needs to be modified, striving for complete symmetry. Currently, the components underneath the MOM capacitors of both 7-bit DACs are not arranged completely identical. This leads to even-odd effects, operating the analog-to-digital converter in a time-interleaved fashion. To avoid this, the layout needs to be rearranged. Before, the lower metal layers were separated in vertical regions for implementing different components. Having completed a first version of the full ADC, now it is possible to optimize the placement of all elements jointly.

An approach to do so could be locating the comparator in the center, since there is only one instance of it. Furthermore, components occurring only once should be located there, too, like the transmission gate connecting both digital-to-analog converters in the 8-bit mode. Of course, the dimensions of all transmission gates need to be adjusted, such that they provide settling times short enough not to constrain the conversion. In case an asymmetric region is introduced by placing these components in the center, it should be covered by dummy MOM capacitors on the upper metal layers. Thus, no capacitors in operation are influenced, but those adjacently still see an unvaried environment.

Underneath the two 7-bit DACs the components occurring twice can be implemented, like the capacitive arrays of the reference DAC and the switches. This symmetry should also avoid possible differences between the two parts of the reference DAC, ensuring a reliable reference voltage, given an adequate refreshing period. By modifying the placement of the switches connected to the bottom terminals of the MOM capacitors, a more efficient use of area can be realized. Currently, large spaces are left blank, that were initially reserved for the control of the DACs, however, not required in the end. Rearranging the regions all together, the switches can be placed more compactly. Doing so, however, the lengths of the control wires need to be considered for equally steep flanks. Probably, a total symmetry can't be achieved here, since the digital control unit is placed at one side of the ADC column, resulting in different distances for both DACs. Then, this should be counteracted by adjusting the driver strengths accordingly.

Performing these adjustments, presumably a smaller area can be reached on the lower metal layers. To benefit from it, also the capacitive array on the upper layers should be modified. The spacing between capacitors was chosen with a large safety margin, to avoid parasitic coupling. However, also a denser placement still shows low disturbances, as found by a parasitic extraction. Hence, it is possible to move together the capacitors, such that an additional fifth one fits in the row of a channel. Then, the current 4×17 grid of a single DAC could be replaced by a 5×14 grid. Thus, depending on possible dummy capacitors in the center, up to six rows could be saved like this. Consequently, it might be feasible to reach the endeavored total area. Of course, the synthesized digital control unit needs to be take into account here, too.

As a further step, adjustments of the unit capacitor itself could be considered. The implemented woven MOM capacitor has six fingers with three belonging to each terminal. A smaller version, e.g. containing four fingers can be designed oneself and tested. If the effect of noise and parasitic coupling doesn't disturb the conversion significantly, this way even more area and, in addition, energy can be saved.

Performing these adjustments, of course the outcomes of further analyses need to be taken into account. Possibly, applying different input patterns reveals systematic errors that were absorbed into the gain before. Furthermore, the linearity might behave differently for inputs in other frequency ranges, as previously discussed. Hence, before optimizing the layout, it is necessary to try detecting and explaining all effects that might occur, such that fixes for them can be integrated directly.

Applying the discussed modifications in a second design iteration, it can be tested, if they do yield the desired improvements. If so, as a next step adjacent ADC channels can be simulated, too, since in the application later 512 channels ought to function simultaneously. Before they are integrated in BrainScaleS-2, however, they are tested in a prototype chip. Fabricating the designed ADC on a mini@sic allows to measure its behavior given physical circumstances and to compare it with simulation results. Like this, it is possible to further identify potential shortcomings and to evaluate the actual performance of the analog-to-digital converter.

A. Appendix

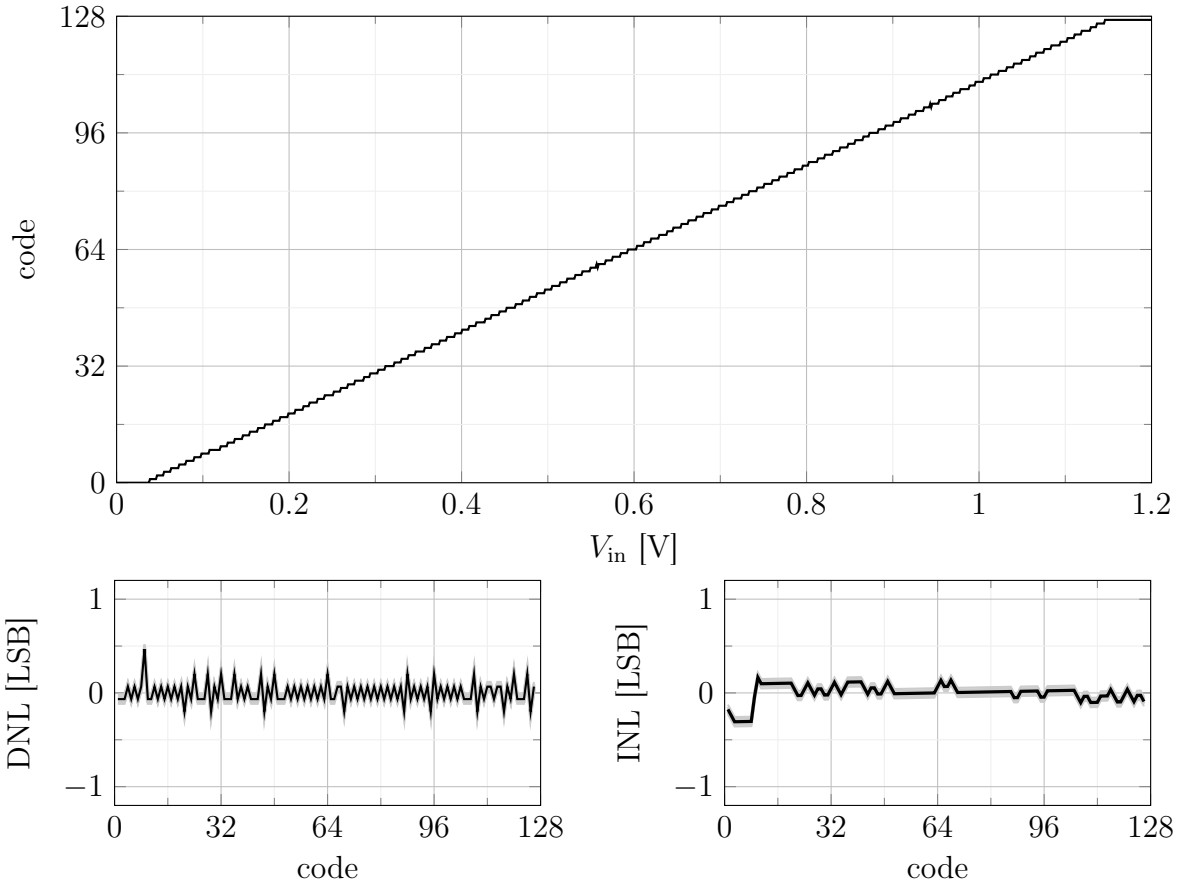


Figure A.1.: Characteristic of the 7-bit mode in the typical case with $V_{\text{dd}} = 1.2\,\text{V}$ and at $T = 50°\text{C}$. The uncertainty is shown in gray. Two non-monotonicities occur, due to an incorrect comparator decision.

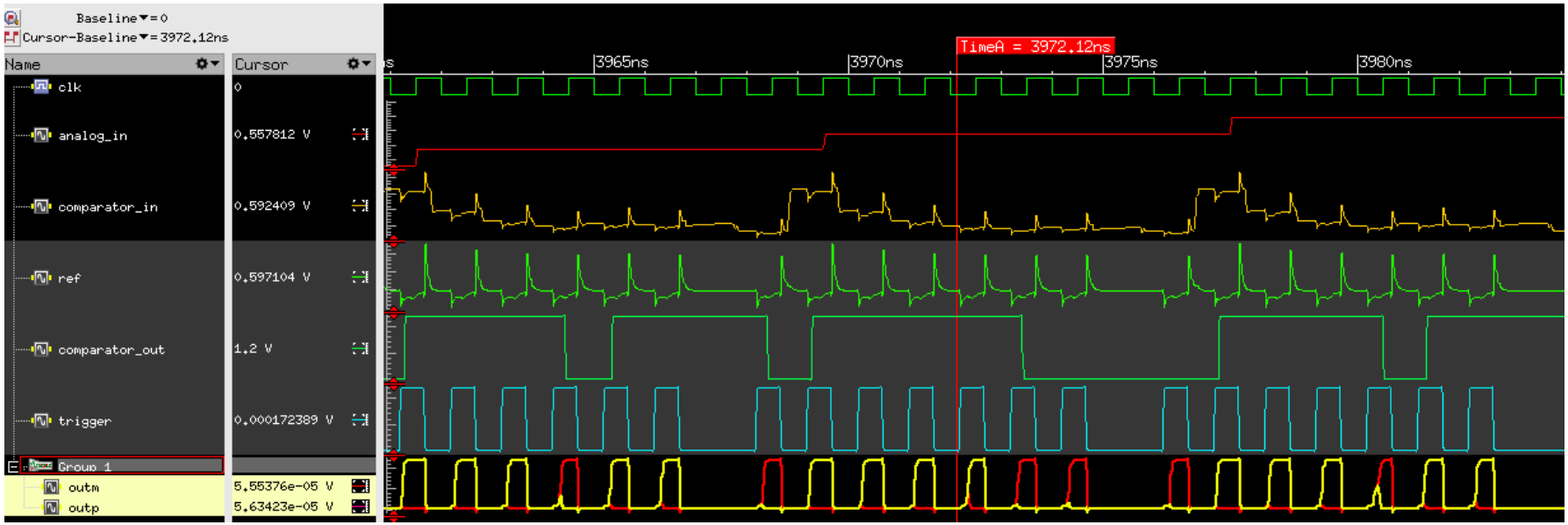


Figure A.2.: Analog traces showing a non-monotonicity. At the contemplated point in time, the comparator decides for an increased code, namely `7'b0111100`. Observing a reference voltage `ref` at the inverting input of the comparator, that is higher than the voltage `comparator_in` at the non-inverting input, reveals an offset of the comparator.

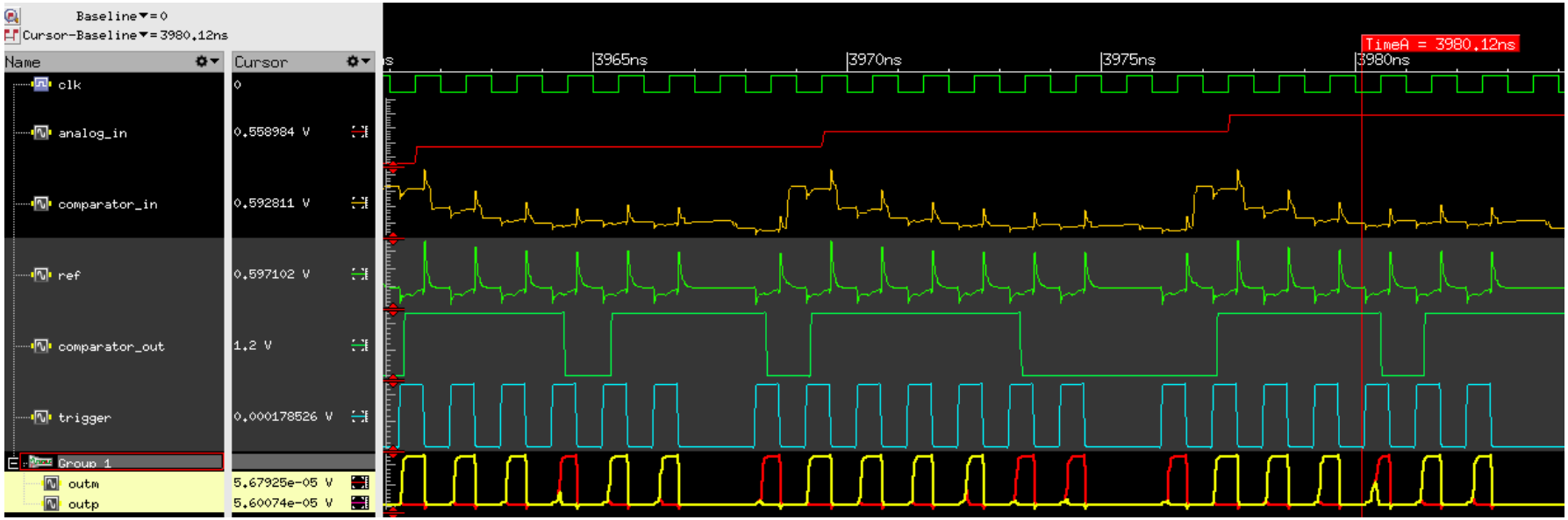


Figure A.3.: Analog traces showing a non-monotonicity. After the comparator decided for an increased output code, in the subsequent sample it returns to its previous decision. Comparing the input voltages `comparator_in` and `ref` with their values from the previous decision, as shown in figure A.2, however, indicates correct voltages.

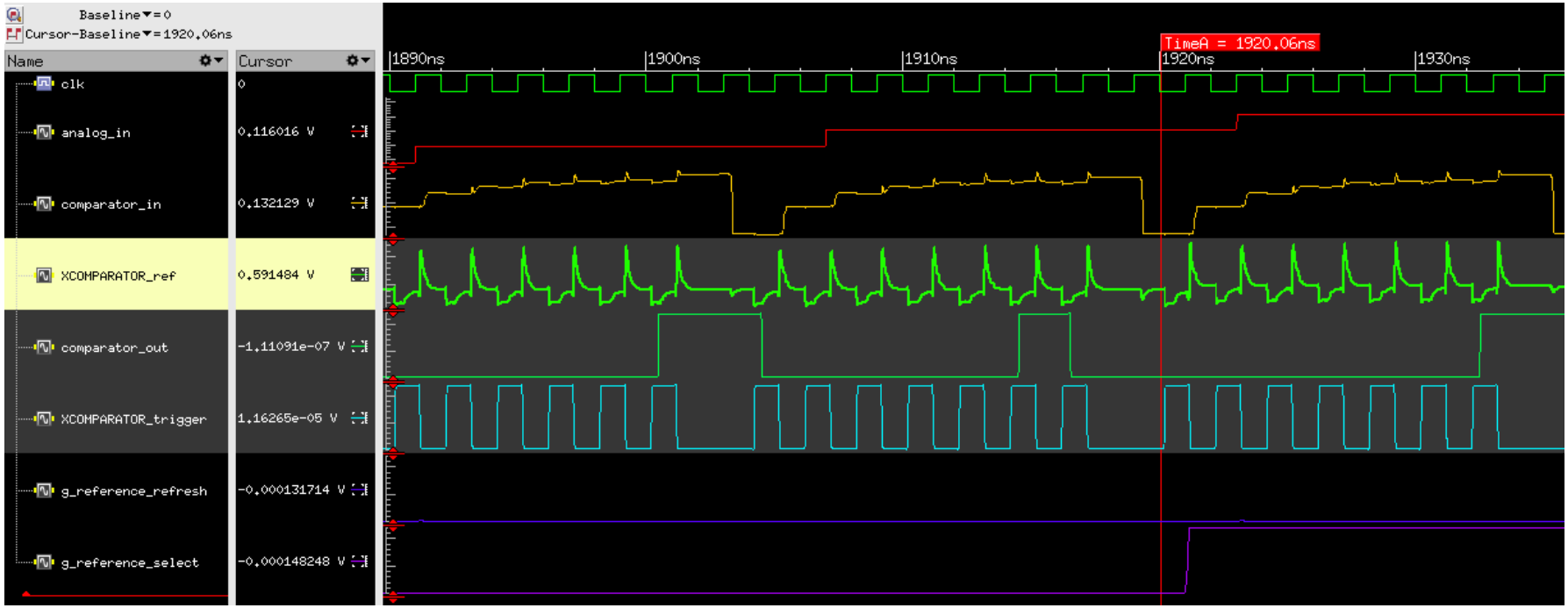


Figure A.4.: Analog traces showing the change of the reference DAC array. Due to the kick-back from triggering the comparator, the voltage at the inverting input of the comparator `XCOMPARATOR_ref` decreases over time. Hence, it needs to be refreshed in regular intervals. To do so, the comparator input is switched to the other array of the reference DAC by `g_reference_select`.

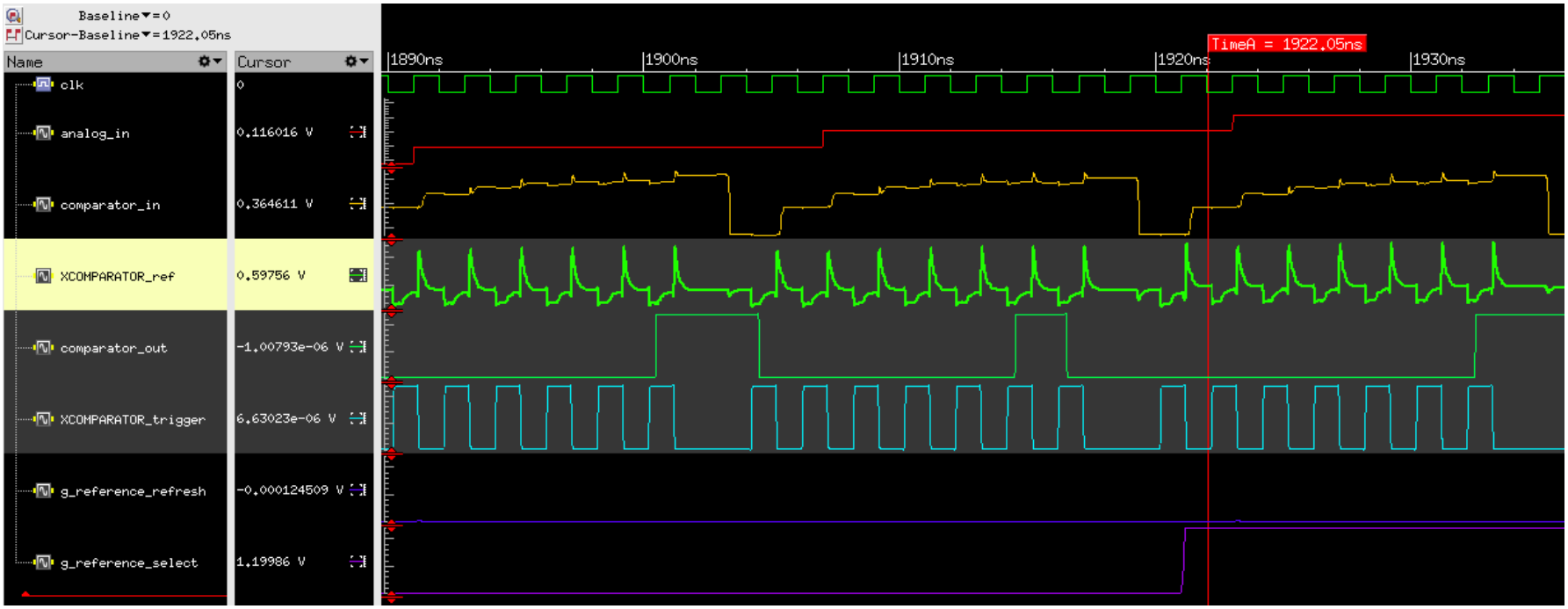


Figure A.5.: Analog traces showing the change of the reference DAC array. After the change, the voltage at the inverting input of the comparator `XCOMPARATOR_ref` is approximately 6 mV higher than before (see figure A.4), leading to a non-monotonicity in the output code.

B. Bibliography

P. E. Allen and D. R. Holberg. *CMOS Analog Circuit Design*. Oxford University Press, 2 edition, 2002.

R. J. Baker. *CMOS: Circuit Design, Layout, and Simulation*. Wiley, IEEE Press, 3 edition, 2010.

S. Bashir, S. Ali, S. Ahmed, and V. Kakkar. Analog-to-digital converters: A comparative study and performance analysis. *2016 International Conference on Computing, Communication and Automation (ICCCA)*, pages 999–1001, 2016.

S. Billaudelle, J. Weis, P. Dauer, and J. Schemmel. An accurate and flexible analog emulation of AdEx neuron dynamics in silicon, 2022. URL `https://arxiv.org/abs/2209.09280`.

T. C. Carusone, D. A. Johns, and K. W. Martin. *Analog integrated circuit design*. Wiley, 2 edition, 2013.

N.-C. Chen, P.-Y. Chou, H. Graeb, and M. P.-H. Lin. High-density MOM capacitor array with novel mortise-tenon structure for low-power SAR ADC. In *Design, Automation & Test in Europe Conference & Exhibition*, pages 1757–1762, 2017. doi: 10.23919/DATE.2017.7927277.

B. Cramer, S. Billaudelle, S. Kanya, A. Leibfried, A. Grübl, V. Karasenko, C. Pehle, K. Schreiber, Y. Stradmann, J. Weis, J. Schemmel, and F. Zenke. Surrogate gradients for analog neuromorphic computing. *Proceedings of the National Academy of Sciences*, 119:e2109194119, 01 2022. doi: 10.1073/pnas.2109194119.

J. Craninckx and G. van der Plas. A 65fJ/conversion-step 0-to-50ms/s 0-to-0.7mW 9b charge-sharing SAR ADC in 90nm digital CMOS. In *2007 IEEE International Solid-State Circuits Conference. Digest of Technical Papers*, pages 246–600, 2007. doi: 10.1109/ISSCC.2007.373386.

F. Cremonesi and F. Schürmann. Understanding computational costs of cellular-level brain tissue simulations through analytical performance models. *Neuroinformatics*,

2020. ISSN 1559-0089. doi: 0.1007/s12021-019-09451-w. URL https://doi.org/10.1007/s12021-019-09451-w.

P. Dauer. Components for a successive approximation ADC on BSS2. Masterarbeit, Universität Heidelberg, 2022.

W. Gerstner, W. Kistler, R. Naud, and L. Paninski. *Neuronal Dynamics*. Cambridge University Press, 2014.

M.-O. Gewaltig and M. Diesmann. NEST (NEural Simulation Tool). *Scholarpedia*, 2 (4):1430, 2007.

J. Göltz, L. Kriener, A. Baumbach, S. Billaudelle, O. Breitwieser, B. Cramer, D. Dold, A. F. Kungl, W. Senn, J. Schemmel, K. Meier, and M. A. Petrovici. Fast and energy-efficient neuromorphic deep learning with first-spike times. pages 823–835, 2021. doi: 10.1038/s42256-021-00388-x. URL https://www.nature.com/articles/s42256-021-00388-x.

A. R. Hamade. A single chip all-MOS 8-bit A/D converter. *IEEE Journal of Solid-State Circuits*, 13(6):785–791, 1978. doi: 10.1109/JSSC.1978.1052051.

M. L. Hines and N. T. Carnevale. *The NEURON Book*. Cambridge University Press, Cambridge, UK, 2006. ISBN 978-0521843218.

J. Jordan, T. Ippen, M. Helias, I. Kitayama, M. Sato, J. Igarashi, M. Diesmann, and S. Kunkel. Extremely scalable spiking neuronal network simulation code: From laptops to exascale computers. *Frontiers in Neuroinformatics*, 12:2, 2018. ISSN 1662-5196. doi: 10.3389/fninf.2018.00002. URL https://www.frontiersin.org/article/10.3389/fninf.2018.00002.

T. Kugelstadt. The operation of the SAR-ADC based on charge redistribution, 2005.

C.-C. Liu, S.-J. Chang, G.-Y. Huang, and Y.-Z. Lin. A 0.92mW 10-bit 50-Ms/s SAR ADC in 0.13µm CMOS process. In *2009 Symposium on VLSI Circuits*, pages 236–237, 2009.

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

Microelectronic Systems Laboratory. MADC documentary, 2015.

H. Nyquist. Certain topics in telegraph transmission theory. *Transactions of the American Institute of Electrical Engineers*, 47(2):617–644, 1928. doi: 10.1109/T-AIEE.1928.5055024.

C. Pehle, S. Billaudelle, B. Cramer, J. Kaiser, K. Schreiber, Y. Stradmann, J. Weis, A. Leibfried, E. Müller, and J. Schemmel. The BrainScaleS-2 accelerated neuromorphic system with hybrid plasticity. *Frontiers in Neuroscience*, 16, 2022. ISSN 1662-453X. doi: 10.3389/fnins.2022.795876. URL `https://www.frontiersin.org/articles/10.3389/fnins.2022.795876`.

M. Pelgrom. *Analog-to-Digital Conversion*. Springer, 3 edition, 2017. doi: 10.1007/978-3-319-44971-5_1.

M. A. Petrovici. *Form Versus Function: Theory and Models for Neuronal Substrates*. Springer, 2016. doi: 10.1007/978-3-319-39552-4.

S. Rapuano, P. Daponte, E. Balestrieri, L. De Vito, S. Tilden, S. Max, and J. Blair. ADC parameters and characteristics. *IEEE Instrumentation and Measurement Magazine*, 8(5):44–54, 2005. doi: 10.1109/MIM.2005.1578617.

J. Schemmel, S. Billaudelle, P. Dauer, and J. Weis. Accelerated analog neuromorphic computing, 2020. URL `https://arxiv.org/abs/2003.11996`.

S. Schmitt, J. Klähn, G. Bellec, A. Grübl, M. Güttler, A. Hartel, S. Hartmann, D. Husmann, K. Husmann, S. Jeltsch, V. Karasenko, M. Kleider, C. Koke, A. Kononov, C. Mauch, E. Müller, P. Müller, J. Partzsch, M. A. Petrovici, S. Schiefer, S. Scholze, V. Thanasoulis, B. Vogginger, R. Legenstein, W. Maass, C. Mayr, R. Schüffny, J. Schemmel, and K. Meier. Neuromorphic hardware in the loop: Training a deep spiking network on the BrainScaleS wafer-scale system. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 2227–2234, 2017.

K. Schreiber. *Accelerated neuromorphic cybernetics*. PhD thesis, Universität Heidelberg, January 2021.

C. Shannon. Communication in the presence of noise. *Proceedings of the IRE*, 37(1):10–21, 1949. doi: 10.1109/JRPROC.1949.232969.

M. Stimberg, R. Brette, and D. F. Goodman. Brian 2, an intuitive and efficient neural simulator. *eLife*, 8:e47314, Aug. 2019. ISSN 2050-084X. doi: 10.7554/eLife.47314.

Y. Stradmann, S. Billaudelle, O. Breitwieser, F. L. Ebert, A. Emmel, D. Husmann, J. Ilmberger, E. Muller, P. Spilger, J. Weis, and J. Schemmel. Demonstrating analog

inference on the BrainScaleS-2 mobile system. *IEEE Open Journal of Circuits and Systems*, 3:252–262, 2022. doi: 10.1109/ojcas.2022.3208413. URL https://doi.org/10.1109%2Fojcas.2022.3208413.

TSMC. CMN65LP MOSFET v1.4 statistical models release notes, 2011.

M. van Elzakker, E. van Tuijl, P. Geraedts, D. Schinkel, E. Klumperink, and B. Nauta. A 1.9μW 4.4fJ/conversion-step 10b 1Ms/s charge-redistribution ADC. In *2008 IEEE International Solid-State Circuits Conference - Digest of Technical Papers*, pages 244–610, 2008. doi: 10.1109/ISSCC.2008.4523148.

P. A. H. Vardhini and M. L. Makkena. Design and comparative analysis of on-chip sigma delta ADC for signal processing applications. *International Journal of Speech Technology*, (24):401–407, 2021. doi: 10.1007/s10772-021-09800-8.

H. Wei, C.-H. Chan, U.-F. Chio, S.-W. Sin, S.-P. U, R. P. Martins, and F. Maloberti. An 8-b 400-Ms/s 2-b-per-cycle SAR ADC with resistive DAC. *IEEE Journal of Solid-State Circuits*, 47(11):2763–2772, 2012. doi: 10.1109/JSSC.2012.2214181.

J. Weis, P. Spilger, S. Billaudelle, Y. Stradmann, A. Emmel, E. Müller, O. Breitwieser, A. Grübl, J. Ilmberger, V. Karasenko, M. Kleider, C. Mauch, K. Schreiber, and J. Schemmel. Inference with artificial neural networks on analog neuromorphic hardware. In *Communications in Computer and Information Science*, pages 201–212. Springer International Publishing, 2020. doi: 10.1007/978-3-030-66770-2_15. URL https://doi.org/10.1007%2F978-3-030-66770-2_15.

F. Zenke and S. Ganguli. Superspike: Supervised learning in multilayer spiking neural networks. *Neural computation*, 30(6):1514–1541, 2018.

J. Zhang. Basic neural units of the brain: Neurons, synapses and action potential, 2019. URL https://arxiv.org/abs/1906.01703.

The work carried out in this Master Thesis will be incorporated in future BrainScaleS systems. The BrainScaleS system received funding from the European Union's Horizon 2020 Framework Programme for Research and Innovation under the Specific Grant Agreements Nos. 720270, 785907 and 945539 (Human Brain Project, HBP)

Acknowledgments

I would like to thank

- Dr. Johannes Schemmel for giving me the opportunity to write my thesis within the Electronic Vision(s) group and agreeing to be my supervisor.
- Prof. Dr. Peter Fischer for examining this thesis as a second reviewer.
- Philipp Dauer for working on this project together with me, for the constructive work-related and personal discussions and his friendship.
- Johannes Schemmel, Sebastian Billaudelle and Andreas Grübl for their impact on my work.
- Johannes Weis, Joscha Ilmberger, Yannik Stradmann and Sebastian Billaudelle for proofreading.
- All members of the Electronic Vision(s) group, who contributed to making my time in the group enjoyable.
- My friends and my family and everyone, who supported me during the course of my studies.

Statement of originality (Erklärung)

I certify that this thesis, and the research to which it refers, are the product of my own work. Any ideas or quotations from the work of other people, published or otherwise, are fully acknowledged in accordance with the standard referencing practices of the discipline.

Ich versichere, dass ich diese Arbeit selbstständig verfasst habe und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Heidelberg, December 9th 2022 .....