NTNU – Trondheim
Norwegian University of
Science and Technology

An energy efficient noise-shaping SAR ADC in 28 nm FDSOI

Harald Garvik

Master of Science in Electronics
Submission date: June 2015
Supervisor: Trond Ytterdal, IET
Co-supervisor: Carsten Wulff, IET

Norwegian University of Science and Technology
Department of Electronics and Telecommunications

Problem description

The main task of this thesis is to design an energy-efficient, noise-shaping SAR ADC at the transistor level in 28 nm FDSOI technology. The intended application is medical ultrasound, and the required specifications are:

- Accuracy: At least 11.0 bit ENOB
- Bandwidth: At least 2 MHz
- Sample rate: At least 32 MHz (i.e. minimum OSR of 4)
- The ADC must be as energy efficient as possible.

This master thesis will be a continuation of the specialization project that was carried out last semester. In that project, noise-shaping SAR ADCs were studied at the architectural level, and behavioral simulations were performed. The results from this work will therefore be used to choose the top level architecture of the ADC to be designed. Results from the master thesis will in turn hopefully verify results found in the specialization project.

The goal is to make a finished ADC that is ready for tape-out. If this turns out to be too time consuming, most effort should be put into the loop-filter part of the ADC such that this can be finished. It may also be a good idea to design the ADC around an already existing SAR implementation.

Abstract

In a noise-shaping SAR ADC, oversampling and noise shaping are used to increase the conversion accuracy beyond that the SAR exhibits alone. To implement the noise shaping, the residue voltage present at the SAR DAC plates after each conversion is exploited, and fed into a loop filter connected to an extra input of the SAR comparator.

In this thesis, an energy efficient noise-shaping SAR ADC for medical ultrasound applications is designed in 28 nm FDSOI. The design specification is minimum 11.0 bit ENOB of accuracy, signal bandwidth of minimum 2 MHz, and sample rate of minimum 32 MHz. According to post-layout Monte Carlo simulations, the designed ADC has an accuracy of 11.1 bit ENOB, and thus satisfies the accuracy requirement. The signal bandwidth and sample rate are the same as in the design specification. Specifically, the topics of this thesis are the design of the loop filter and its interfacing towards the SAR, as well as the overall high level design. The 9-bit SAR used in the system is an already existing implementation.

A cascaded FIR-IIR filter topology is used for the loop filter. In this work, the circuit implementation of this topology is improved, most importantly through the introduction of chopped buffers at the filter input. This eliminates signal attenuation due to charge sharing, and a DAC capacitance that is smaller than the sampling capacitance in the loop filter can therefore be used. Also, auto-zeroed, cascoded inverters rather than a standard OTA are used as gain elements in the switched-capacitor filter structure, and this leads to better energy efficiency.

The designed ADC achieves a figure of merit (FOM) of 7.5fJ/conv-step in post-layout Monte Carlo simulations, and to the best of the author's knowledge, this is better than the current state-of-the-art of noise-shaping ADCs. When all kinds of ADCs are taken into consideration, the achieved FOM seems to be similar to the current state-of-the-art in the same specification range.

Sammendrag

I en støyformende SAR ADC blir oversampling og støyforming brukt til å øke konverteringsnøyaktigheten forbi den SARen har alene. For å implementere støyformingen blir restspenningen som ligger på kondensatorplatene til SAR DACen etter hver konvertering utnyttet, og sendt inn i et loopfilter som er koblet til en ekstra inngang på SAR-komparatoren.

I denne oppgaven blir en energieffiktiv støyformende SAR ADC designet i 28 nm FDSOI for bruk innen medisinsk ultralyd. Designspesifikasjonen er minst 11.0 bit ENOB nøyaktighet, signalbåndbredde på minst 2 MHz, og samplingsrate på minst 32MHz. I følge post-layout Monte Carlo-simuleringer har ADCen som er designet en nøyaktiget på 11.1 bit ENOB, og oppfyller derfor nøyaktighetskravet. Signalbåndbredden og samplingsraten som brukes er den samme som i designspesifikasjonen. Spesifikt er hovedtemaene i denne oppgaven designet av loopfilteret og tilhørende grensesnitt mot SARen, samt det generelle høynivådesignet. SARen på ni bit som blir brukt er en implementasjon som allerede finnes.

Det brukes en kaskade FIR-IIR filtertopologi i loopfilteret. Denne topologien blir forbedret i denne oppgaven, blant annet ved å inkludere buffere med chopping på filterinngangen. Dette eliminerer signaldempning på grunn av ladningsrefordeling, og dette muliggjør bruk av DAC-kapasitans som er mindre enn samplingskapasitansen til loopfilteret. I tillegg til dette brukes kaskodeinvertere med auto-zeroing som forsterkningselement i svitsjet-kapasitans-strukturen til filteret i stedet for en vanlig OTA. Dette fører til bedre energieffektivitet.

ADCen som er designet oppnår et ytelsestall (FOM) på 7.5 fJ/conv-step i post-layout Monte Carlo-simuleringer, og dette er så langt undertegnede vet bedre enn nåværende state-of-the-art på støyformende ADCer. Hvis alle typer ADCer tas med i betraktningen ser oppnådd FOM ut til å være omtrent lik som nåværende state-of-the-art i samme spesifikasjonsområde.

Preface

First, I would like to thank my supervisors Professor Trond Ytterdal and postdoc. Carsten Wulff. Thank you very much for good help and counselling, lots of laughs, tons of coffee, and for helping me realize that analog design is the way to go. I really look forward to continue as a Ph.D. student under your supervision.

I will also thank my fellow analog design students Erlend Strandvik, Endre Larsen and Thomas Nornes for a memorable year and good moments in our small office. I will definitely miss it.

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

Energy efficiency is one of the most important requirements when electronic devices are designed today, and this is also one of the main reasons for the continuously ongoing downscaling of CMOS processes. Digital circuitry benefit greatly from this downscaling due to decreased parasitic capacitance in the logic gates.

To achieve low power consumption for entire chips or systems, energy efficiency is also essential for the analog blocks inside the integrated circuits. One class of such blocks are analog-to-digital converters (ADCs), which are present in almost every electronic device to facilitate interfacing between the analog world and the digital system. There exists several studies of the energy efficiency of ADCs, and among them are [1–3]. In such studies, it is found that ADCs having low accuracy (less than 60 dB SNDR approx. [1]) have their energy efficiency limited by the minimum feature sizes of the process, and thus benefit from downscaling. Also, the energy consumption of such ADCs is found to approximately double every time the accuracy is increased by one bit ENOB (effective number of bits).

On the other hand, medium-to-high accuracy ADCs are usually found to have their accuracy limited by the thermal noise of the circuitry. When this is true, the thermal noise power of the converter must be decreased by a fourfold, i.e. 6 dB if the one wants to increase the accuracy of the ADC by one bit ENOB. Since it is generally needed to increase the power consumption by a factor of two every time thermal noise is halved, the power consumption of the converter then increases by

four. This drastic increase in power consumption per bit ENOB impacts the energy efficiency of medium-to-high accuracy ADCs.

Roughly, we can thus divide ADCs into a process limited regime where the energy consumption increases by factor two per bit, and one thermally limited regime where the energy consumption quadruples per bit. It also makes sense to state that accuracy and energy only trades fair in the first regime (i.e. a twofold energy increase for a twofold accuracy increase), and that the energy efficiency of thermally limited ADCs therefore deteriorates. Moreover, process limited ADCs benefit from scaling, while thermally limited ADCs will generally not do that [1]. This in turn means that more and more of the process limited ADCs will hit thermal noise limitations as the scaling goes on, and thus enter the thermal regime. This development is also shown in [1].

The current trend seems to be that the most energy efficient ADCs reported in the literature are of the SAR (successive approximation register) type [4, 5]. The circuitry of this kind of ADC is relatively simple, and in addition to digital circuitry, the SAR only consists of a capacitive DAC, a comparator, and a sample-and-hold switch. This makes it well suited for energy efficient operation in the process limited regime. In recent years, however, different techniques aiming to make the SAR better suited also at high accuracy levels have been presented in the literature. In [6, 7] for example, a data driven noise reduction scheme is employed for the comparator to alleviate thermal noise, and the comparator accuracy is in this way increased in an energy efficient manner. Another interesting technique is pipelining of SARs, such as in [8, 9]. These papers presents two-step pipeline SAR structures, where the requirements for the last SAR is relaxed by the pipeline itself, and the requirements for the first SAR is relaxed by the redundancy between the stages. The residue from the first stage still has to be generated and propagated to the last stage at the full ADC accuracy though. According to [10], all the ADCs mentioned in this paragraph exhibit state-of-the-art energy efficiency, and some of them do so in the 70 dB SNDR accuracy region.

A noise-shaping SAR (NS-SAR) is another improvement to the SAR structure, aiming to extend its region of energy efficient operation. Here, oversampling is used together with noise-shaping, and like in a delta-sigma converter, this lowers the quantization noise of the ADC. Additionally, the noise of the SAR ADC comparator is also shaped, and its accuracy can therefore be considerably relaxed [11]. In a noise-shaping SAR, it is the residual charge left on the DAC array after a conversion that is exploited for noise shaping. This charge is extracted and fed into a filter (which can be as simple as a unit delay), which subsequently connects into the ADC signal chain at some other point to realize a noise-shaping loop. The use of the SAR DAC for residue generation eliminates the need for an extra DAC, like in a delta-sigma converter.

The noise-shaping SAR concept seems to first be described in [12], and thereafter [13]. In these papers, noise-shaping according to the error-feedback scheme [14, p. 81] is proposed and shown in simulations. That is, the residue is extracted and

fed into a passive FIR filter, which injects its output into the the ADC input. In this way, n-th order noise shaping can be achieved, depending on the filter. However, the error feedback scheme is in general very sensitive to parameter variations [14, p. 81], and the circuits must therefore be designed carefully, with low losses in the noise-shaping loop. A handful of other papers also show transistor schematic level simulations of error-feedback NS-SARs [15–17]. Some of these circuits have low losses in the noise-shaping loop, but the unit capacitor of the SAR DAC has to be sized according to the kT/C noise requirement associated with the input sampling. This is in contrast to normal SARs, where the kT/C noise requirement sets the size of the whole DAC array, and not the unit capacitor. This means that the small unit capacitors that can be realized in deep submicron CMOS technologies may be impossible to use. There also exists a taped-out error-feedback NSSAR [18], but due to large losses, the ENOB improvement due to the noise-shaping is modest.

Another NS-SAR approach is presented in [11], and also proven by measurements. Here, the filter output is fed into an extra comparator input rather than to the ADC input. Also, both a passive FIR filter and an IIR filter consisting of an active integrator is used. This yields a noise-shaping system more similar to delta-sigma converters, where one or more integrators in the loop are used to store the filter states. The extraction of the residue is done passively by charge sharing, and this introduces some loss. However, the structure is not very sensitive to this loss, as will be shown in chapter 3.

1.1 Goal of this thesis

In the specialization project preceding this thesis [19], a comprehensive high-level study of NS-SARs was carried out. The structure presented in [11] was used as a starting point, and possible alternatives to the filter topology used there were investigated. All filters were connected to an extra comparator input, likewise in [11]. Also, the optimum (i.e. most energy efficient) choice of high level design variables like SAR bit count, and amount of oversampling was studied. This resulted in a behavioral simulation framework which is capable of selecting the optimum design variables for a specific design specification, given a set of assumptions about energy consumption. The framework was also used to compare the filter topologies.

Building upon the work carried out in the specialization project, the goal of this master thesis has been to move on to the transistor level and implement a noise-shaping SAR in 28 nm FDSOI CMOS technology. The intended application of the ADC is medical ultrasound, and the specification is given in table 1.1. Given this specification, it was found in the project thesis that the same topology as in [11] is most optimal, and it is hence used in the implementation.

The actual SAR block used in the implementation has not been developed as a part of this thesis, but is the work of postdoc. Carsten Wulff at NTNU [20]. Carsten's

Table 1.1: Design specification for the NS-SAR implemented in this thesis:

Accuracy	> 11 bit ENOB
Bandwidth, B_w	> 2 MHz
Sample rate, f_s	> 32 MHz

current research topic is energy efficient SARs in 28 nm FDSOI, and it was therefore considered wise to use one of his SARs. The reason for this is both that the implementation of a whole NS-SAR during the time frame of a master's thesis can be challenging, and because it was considered more important to focus on the blocks not already developed in this technology. The work of this master's thesis has thus consisted of the design of the loop filter and the related interfacing towards the SAR, high level design of the entire system, in addition to verification of the NS-SAR as a whole.

Post-layout simulations show the ADC designed in this thesis exhibit an accuracy of 11.1 bit ENOB, and the design specification is thus met. Also, a figure of merit (FOM) of 7.5 fJ/conv-step is achieved, and this is to the best of the author's knowledge better than the current state-of-the-art of noise-shaping ADCs, and similar to the current state-of-the-art in the same specifiaction range if any ADC type is considered.

1.2 Main contributions

More systematically, the main contributions of this thesis are

- The high level design, schematic design, layout design, and post-layout verification of a noise-shaping SAR ADC in 28 nm FDSOI. In this design, the actual SAR block is from [20], and the rest is carried out in this thesis.
- The implementation of the loop filter topology found in [11]. The circuit implementation of the topology has in this thesis been improved by the use of input buffers, such that an arbitrary DAC capacitance can be used without introducing loop gain attenuation due to charge sharing. Chopping has also been implemented around these buffers to mitigate flicker noise, and a switching scheme that merges the chopping and sampling switches in the filter has been developed.
- Modification of the sampling capacitor scheme used to implement the FIR part of the loop filter in [11]. The result is the removal of a capacitor, and saved area.
- The use of auto-zeroed inverter amplifiers in the loop filter implementation, likewise as in [26]. This makes the loop filter more energy efficient, and is to

the best of the author's knowledge not done in any NS-SAR before.

- Implementation of the improved source-follower topology found in [29] for the use as loop filter buffers. The buffers are turned on only when they are needed, and thus only constitute a small share of the ADC power consumption. Also, equations for the bandwidth and noise power of the buffer are derived in this thesis.
- An analysis of the optimum division of the noise budget in a NS-SAR. Specifically, some equations developed can be used numerically to find the optimum share between shaped and unshaped noise in a given design.
- An investigation of how incomplete linear settling in the loop filter affects the ADC performance. The results have been utilized during the circuit design, with the result that the loop filter is not overdesigned.

1.3 Thesis outline

The rest of this thesis is organized as follows

Chapter 2 – Background theory: Important background theory for the rest of the thesis is given. A part of it is a summarization of the theory and NS-SAR overview chapters in [19].

Chapter 3 – High level design: The filter topology as well as high level design parameters like SAR bit count and oversampling rate (OSR) are presented and analyzed. Also, some important requirements for the filter implementation, like settling times are discussed.

Chapter 4 – Implementation: A detailed description of the schematic design of the loop filter is given. Finally, the layout is presented.

Chapter 5 – Results: Post-layout simulation results for the entire NS-SAR.

Chapter 6 – Discussion: A comparison to the current state-of-the-art is done, and the results and the design are discussed. Also, the validity of the assumptions done in [19] is considered.

Chapter 7 – Conclusion: The thesis is concluded, and the way ahead is described.

CHAPTER 2

Background theory

In this chapter, some background theory topics relevant to the NS-SAR design are presented. To cover all the relevant background material would not be feasible in this thesis, so the emphasis has been put on theory that is more or less directly used in the subsequent chapters.

The goal of the first part of the chapter is to give the reader a basic understanding of how a noise-shaping SAR works. This is done by first covering the principles of oversampling and noise-shaping, before the noise-shaping SAR itself is covered. All these topics were extensively covered in the project thesis [19], and the presentation given in this chapter is a summarization of how the topics were treated there. The project thesis chapter that explained the principle of NS-SARs is also included in this thesis as appendix A.

Most of the topics presented can be regarded as commonly known, and explicit citations are thus not given through the chapter unless this is not the case. The main sources used as theoretical background are [21–23], as well as the project thesis.

2.1 Oversampling and noise shaping

When an ADC converts an analog voltage V_{in} to a digital word D_{out}, there will be a quantization error due to the finite number of digital output codes available. This limits the accuracy of the conversion. However, if we increase the sample rate of the ADC beyond the Nyquist rate, we obtain redundant samples which can be used to decrease the effective quantization error through the use of averaging/low pass filtering. This technique is called oversampling and is described in the following.

If we express D_{out} in the unit of volts, the quantization error can be written as

$$V_Q = D_{out} - V_{in} \tag{2.1}$$

From this, we see that V_Q is deterministic and correlated to V_{in}. However, if V_{in} varies quite rapidly, the distribution of the V_Q-values tends to resemble that of uniformly distributed white noise. It is therefore possible to treat the quantization error as an independent, linear noise source $e(n)$, and this greatly simplifies the analysis of ADCs.

If we then denote the spacing between two adjacent output levels of the ADC as Δ, it can be shown that the corresponding quantization noise source will have a noise power of

$$P_E = \frac{\Delta^2}{12} \tag{2.2}$$

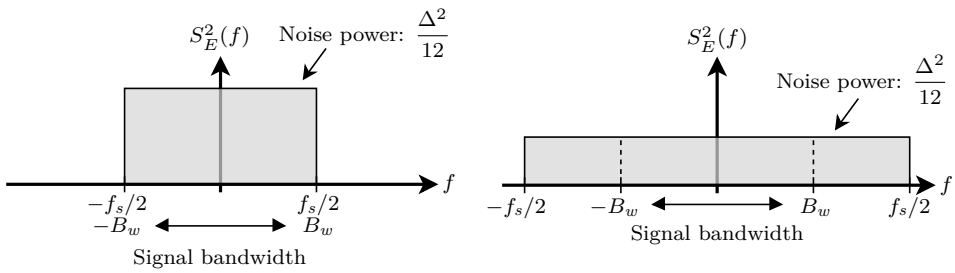
This relationship is derived by integration of the quantization noise probability density function, and is for instance carried out in [19].

Since we assume that the quantization noise is white, it is easy to find its power spectral density when we know the noise power. This can be done by realizing that no power can lie outside $[-f_s/2, f_s/2]$ due to the sampling, and that the integral of the power spectral density have to equal P_E. The power spectral density is thus

$$S_E^2(f) = \frac{\Delta^2}{12 f_s} \tag{2.3}$$

In figure 2.1a, the power spectral density is shown for an ADC operating at the Nyquist rate. The signal bandwidth B_w thus corresponds to $f_s/2$. If we then double the sample rate, but retain the signal bandwidth, we get the situation in figure 2.1b. Here, the noise power (the integral of $S_E^2(f)$) is still the same. However, the power spectral density is halved due to the f_s increase, and from the figure we can readily see that this leads to halved noise power inside the signal band. If we finally assume that we can remove the out-of-band noise with a digital filter, the conversion accuracy of the ADC has been increased through the use of oversampling.

To quantitatively find the in-band noise power, we integrate the power spectral



(a) OSR = 1 **(b)** OSR = 2

Figure 2.1: Noise spectral densities with and without oversampling [19].

density in the signal band. That is

$$P_E = \int_{-B_w}^{B_w} S_E^2(f)\,df = \frac{\Delta^2}{12 f_s} 2B_w = \frac{\Delta^2}{12\,\mathrm{OSR}} \tag{2.4}$$

where the *oversampling rate* (OSR) denotes how many times faster than the Nyquist rate the converter operates, and is given as

$$\mathrm{OSR} = \frac{f_s}{2B_w} \tag{2.5}$$

We then take the signal to noise ratio between the in-band quantization noise and a full range sinusoid to establish the accuracy of the oversampling ADC. If the spacing between adjacent ADC outputs is Δ, and the resolution is B bits, the signal range of the ADC will be $2^B\Delta$. The power of a full range sinusoid is thus

$$P_S = \left(\frac{2^B\Delta}{2}\frac{1}{\sqrt{2}}\right)^2 = \frac{\Delta^2 2^{2B}}{8} \tag{2.6}$$

The signal to quantization noise ratio (called SQNR) is then

$$\mathrm{SQNR} = 10\log\left(\frac{P_S}{P_E}\right) = 10\log\left(\frac{3}{2}\mathrm{OSR}\,2^{2B}\right) = 6.02B + 1.76 + 10\log(\mathrm{OSR}) \tag{2.7}$$

We can also express the SQNR as effective number of bits (ENOB) through the commonly used relation $\mathrm{SNR}_{\mathrm{dB}} = 6.02\,\mathrm{ENOB} + 1.76$. This yields

$$\mathrm{ENOB} = B + \frac{1}{2}\log_2(\mathrm{OSR}) \tag{2.8}$$

From this we see that we gain 0.5 bit of extra accuracy for every doubling of OSR, i.e. for every octave of f_s increase.

Although oversampling increases the ADC accuracy, it is not very attractive on its own in terms of energy efficiency. This is because the power consumption of the

ADC generally doubles when the sampling rate is doubled. The energy used per bit in a conversion will therefore increase if we only get 0.5 bit ENOB per doubling of f_s, and the ADC becomes less energy efficient. It is therefore common to increase the number of bits achieved per octave of oversampling through the use of *noise-shaping*.

In noise shaping ADCs, which are often called delta sigma modulators, the shape of the noise power spectrum is altered through the use of a feedback loop and a loop filter. This is shown in figure 2.2. If a suitable transfer function $H(z)$ is chosen for the filter, it is possible to lower the in-band part of the noise spectrum significantly at the cost of increased noise outside the band. The accuracy of the ADC is thus further increased. Note that a DAC is needed in the feedback loop because the feedback goes from the digital to the analog domain.

To analyze delta sigma modulators, it is easiest to use a linear model of the system. This is given in figure 2.3, and was obtained by swapping the ADC block with its linear counterpart, i.e. the quantization noise source. Also, there is no need for the DAC in the model when we express $D_{out}(n)$ in volts.

We can now derive transfer functions from the input signal $u(n)$ and the quantization noise $e(n)$ respectively, to the ADC output. This is done by direct analysis of the signal schematic in figure 2.3, and gives us the *signal transfer function* (STF) and the *noise transfer function* (NTF) as

$$\mathrm{STF}(z) = \frac{D_{out}(z)}{U(z)} = \frac{H(z)}{1 + H(z)} \tag{2.9}$$

and

$$\mathrm{NTF}(z) = \frac{D_{out}(z)}{E(z)} = \frac{1}{1 + H(z)} \tag{2.10}$$

From these equations, we see that if the magnitude of $H(z)$ is large in the signal band, the system passes the signal quite unaffected through to the output while the noise is attenuated. The principle becomes quite analogous to an op-amp in unity gain feedback, where the output have to track the input accurately due to the high loop gain.

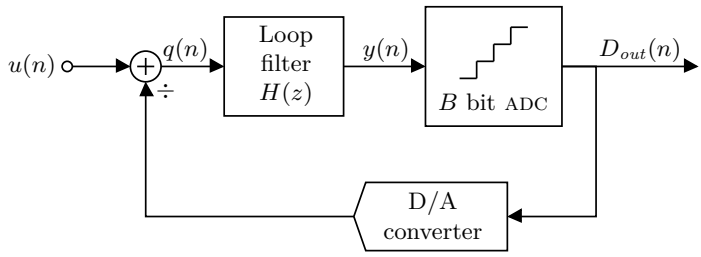


Figure 2.2: Delta sigma modulator [19].

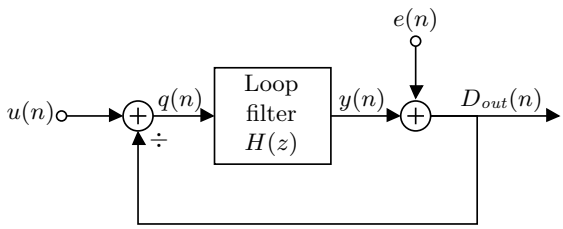


Figure 2.3: Delta sigma modulator linear model [19].

Equation (2.4) can now be altered to give the in-band noise power when noise shaping is taken into account. This gives

$$P_E = \int_{-B_w}^{B_w} S_E^2(f)|\mathrm{NTF}(f)|^2\,df = \frac{\Delta^2}{12f_s}\int_{-B_w}^{B_w}|\mathrm{NTF}(f)|^2\,df \tag{2.11}$$

The actual in-band noise power thus depends on the loop filter, but will clearly be much smaller than with pure oversampling if $\mathrm{NTF}(f)$ is chosen small in the signal band. The choice of transfer functions for the loop filter was one of the main topics of the specialization project [19], and is hence not treated extensively here. The choice of transfer function/loop filter topology for the ADC designed in this master's thesis is a direct result from [19], and will be presented in chapter 3.

2.2 Noise-shaping SAR

In a noise-shaping SAR, the principles of oversampling and noise-shaping is used in a SAR ADC to improve its accuracy. This is done by using the SAR as the ADC block in a delta-sigma modulator (see figure 2.2). In addition to this, the internal DAC that the SAR uses to conduct binary searches is also used as the feedback DAC in the delta-sigma loop. This means that the only block that is needed in addition to the SAR itself is the loop filter, and the NS-SAR is therefore a noise-shaping ADC where blocks are reused for different purposes. This is advantageous when it comes to energy efficiency.

The workings of NS-SARs are best illustrated by an example (a similar example/derivation is also carried out in [11]): Consider figure 2.4, which shows a simple NS-SAR circuit. The input is first sampled at the bottom plates of the capacitive DAC, while the comparator is held in reset. After this, the reset switch is opened, and the bottom plates are switched to ground. Due to charge conservation at the top plates, we will then get

$$V_{res} = -V_{in} \tag{2.12}$$

The digital SAR logic will then conduct a binary search to find the digital code closest to V_{in}. This is done by switching the DAC switches to V_{ref} in a certain

pattern. This alters the top plate voltage to

$$V_{res} = -V_{in} + V_{D/A} \tag{2.13}$$

where the additional term represents the current state of the DAC. As V_{res} is fed into the comparator, the SAR logic can find the correct digital word by observation of the comparator results as the DAC is switched into different states. Finally, we get the output

$$D_{out}(n) = V_{in}(n) + V_Q(n) \tag{2.14}$$

if we assume that the voltage at the comparator positive terminal was zero during the conversion. Also note that we will have $V_{res}(n) = D_{out}(n) - V_{in}(n)$ after a completed conversion.

So far, the operation is similar to a normal SAR. However, if we sample the negative value of the SAR top plates V_{res} after each conversion, and then apply it to the positive comparator input during the next conversion, we will introduce noise shaping. It can be shown (see appendix A) that this will change the output to

$$D_{out}(n) = V_{in}(n) - V_{res}(n-1) + V_Q(n) \tag{2.15}$$

Then, by using $V_{res}(n) = D_{out}(n) - V_{in}(n)$, taking the z-transform and rearranging, we arrive at

$$D_{out}(z) = V_{in}(z) + \frac{1}{1+z^{-1}} V_Q(z) \tag{2.16}$$

This means that the input is let straight through to the output, while the quantization error V_Q is shaped by a factor $\frac{1}{1+z^{-1}}$. This is a high-pass filter, and the quantization noise is thus lowered at low frequencies, as it should in a noise-shaping ADC. However, the maximum attenuation of this filter is only -6 dB and will not lead to attractive bandwidth-accuracy trade-offs when energy efficiency is considered [11].

The noise-shaping can be improved if the sampling capacitor at the positive comparator terminal in figure 2.4 is exchanged with a more general loop filter. To

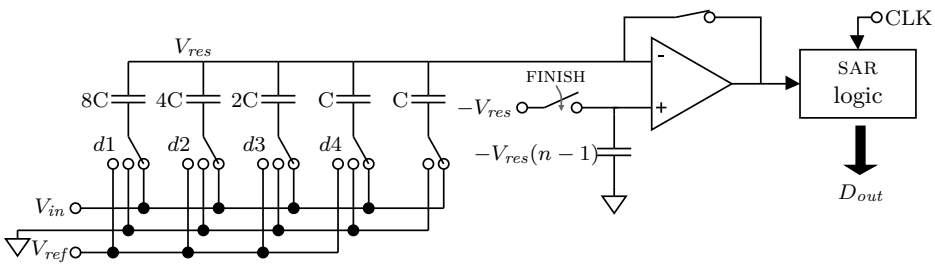


Figure 2.4: A simple noise shaping SAR adding the previous residue to the current output [19].

easier analyze a system like this, a linear model of the NS-SAR with general loop filter $H(z)$ was developed in the project thesis (and also included in appendix A). This model is depicted in figure 2.5, and closely resembles that for a general delta-sigma modulator in figure 2.3. The main difference is the feed forward path from the input to the output. This is present because it is only V_{res} that propagates through the loop filter, and not the signal. This fact greatly eases the loop filter design, because it will only need signal swing to accommodate V_{res}, which is small if the SAR has many bits.

The STF and NTF can be directly derived from the model, and this yields

$$\mathrm{STF}(z) = \frac{D_{out}(z)}{U(z)} = 1 \tag{2.17}$$

and

$$\mathrm{NTF}(z) = \frac{D_{out}(z)}{E(z)} = \frac{1}{1 + H(z)} \tag{2.18}$$

That is, the NTF is identical as for a delta-sigma converter, while the STF is unity. The last mentioned is due to the extra feed forward path.

Another thing that it is important to realize, is that the feedback path from the output does not exist in the physical NS-SAR circuit. This is because $-q(n) = V_{res}$ is generated directly by the SAR during each conversion (to flip the sign of $q(n)$ is easily done in a differential implementation). This is how the need for an extra feedback DAC is eliminated.

When it comes to circuit noise, the NS-SAR will shape the comparator noise, because it enters the circuit at the same place as the quantization noise $e(n)$. However, $\frac{kT}{C}$-noise in the DAC as well as input referred noise in the filter itself will not be shaped. These noise contributions will still be lowered by oversampling though.

Note that the error feedback NS-SARs mentioned in the introduction does not feed the loop filter output into an extra input of the comparator. Due to this, the theory presented here is not totally valid for such noise-shaping SARs.

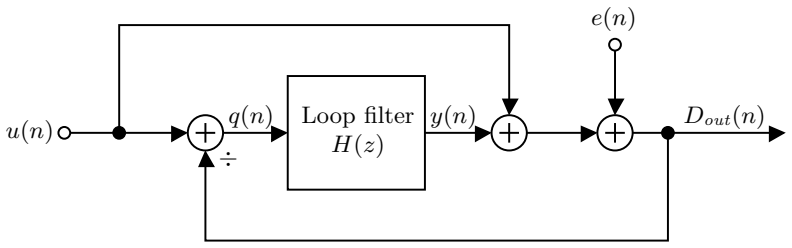


Figure 2.5: General NS-SAR linear model [19].

2.3 Power consumption of amplifiers

The main cause of power consumption in the loop filter designed in this thesis, as well as in many other circuits, is amplifiers. It is therefore important to understand how much power an amplifier have to use, and if it is sized for optimum power efficiency. These topics are explored in this section.

A key amplifier property is small signal speed, often expressed as a unity gain frequency f_t. To understand what governs this quantity, consider the amplifier with capacitive load in figure 2.6. Assume that this amplifier has a high output impedance, as is commonly the situation for integrated amplifiers. Also, assume that the amplifier has a single stage topology, such that its dominant pole is determined by its load capacitance.

If we gradually increase the frequency of v_{in}, we will eventually enter a region where the load capacitor C_L dominates the impedance of the output node, and the internal amplifier output impedance can be neglected. This region can be called the mid-band region of the amplifier. We can here regard the amplifier as a transconductor with transconductance G_m, where the small signal current that arises due to v_{in} is entirely delivered to the capacitor as an output current

$$i_{out} = G_m v_{in} \tag{2.19}$$

We can now find the transfer function valid under these assumptions. At the output node, we have

$$v_{out} = \frac{i_{out}}{sC_L} = \frac{G_m}{sC_L} v_{in} \tag{2.20}$$

which yields

$$H(s) = \frac{v_{out}}{v_{in}} = \frac{G_m}{sC_L} \tag{2.21}$$

Furthermore, we can find f_t by taking the magnitude of the frequency response and equate to one. That is

$$|H(f_t)| = \frac{G_m}{2\pi f_t C_L} = 1 \tag{2.22}$$

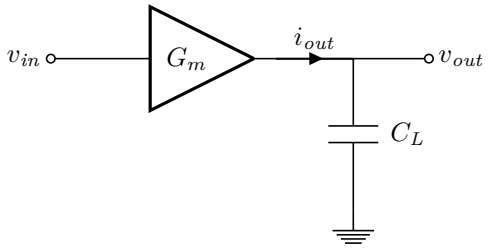


Figure 2.6: Amplifier with capacitive load

which finally gives the unity gain frequency as

$$f_t = \frac{G_m}{2\pi C_L} \tag{2.23}$$

This relationship reveals that for a given unity gain frequency, the transconductance is proportional to the load capacitance. That is, an increase in C_L, for example to mitigate noise or mismatch problems, calls for a corresponding increase in G_m.

The next question is how G_m of the amplifier is related to the power consumption. To answer this, we assume that G_m of the amplifier is proportional to g_m to one of the active transistors, and that we therefore can consider g_m without loss of generality. It is then possible to consider the so-called g_m/I_D ratio of transistors to relate g_m to the drain current I_D. This quantity should in turn be directly proportional to power for a given supply voltage and amplifier topology.

The g_m/I_D ratio is a quantity that turns out to be independent of the transistor sizing in a given technology [24]. Furthermore, it is dependent on the inversion level of the transistor. The largest g_m/I_D level is achieved in weak inversion, where it has the constant level of $\frac{q}{nkT}$, where n is called the sub-threshold slope. When the operating point is increased towards strong inversion, the g_m/I_D ratio decreases. In the used 28 nm FDSOI technology, maximum g_m/I_D values of around 33 has been observed during simulation in weak inversion, which yields $n \approx 1.15$.

If we now regard g_m/I_D as a parameter of the transistor, and substitute it into equation (2.23), we can write

$$I_D = \frac{2\pi f_t}{(g_m/I_D)} C_L \tag{2.24}$$

From this, we conclude that the power consumption of an amplifier scales with load capacitance, and that it is minimized if g_m/I_D is maximized. That is, the active transistors should have as small overdrive as possible. Note that we have only considered small signal speed in this discussion. Slewing may also impact the performance, but this is not critical in the designed loop filter since the magnitude of the residue signal is small.

2.4 Switched-capacitor circuits

Switched-capacitor circuits (SC circuits) are discrete time analog signal processing circuits where the signals are represented as charges stored on different capacitors. As the circuit operates, the charges are moved between the capacitors in a certain pattern through the use of amplifiers and clocked switches, and a signal transfer function is in this manner realized. Since a switched capacitor circuit is a discrete time system, it can only represent discrete time signals, i.e. series of samples. It

is thus common to denote signals propagating in such circuits as $v(n)$, rather than $v(t)$. n is here the sample number corresponding to the time instant $t = nT$ where T is the sampling period.

As an example, consider the switched capacitor integrator in figure 2.7. The circuit consists of two capacitors, an op-amp which provides a virtual ground, and a set of clocked switches that are realized as either single transistors or transmission gates, depending on the signal levels. The clock signals ϕ_1 and ϕ_2 run with opposite phase, and have to be non-overlapping to prevent unintended charge leakage. Say, if ϕ_1 first goes high, it will have to go low before ϕ_2 can rise. This means that the clocks typically will have a duty cycle of slightly less than 50 %, and the rest of the time in each clock cycle will be "non-overlapping intervals" where both clocks are low. A special non-overlapping clock generator is needed to realize a clocking scheme like this. This can typically be a circuit consisting of logic gates that derive ϕ_1 and ϕ_2 from a single incoming clock signal.

To illustrate how the circuit in figure 2.7 operates as an integrator, its transfer function will now be derived. Assume that the op-amp is ideal during this derivation, and that the circuit settles fully in each clock phase. Further assume that the discrete output signal $v_{out}(n)$ is sampled by the next circuit block at the end of the ϕ_1 phase. To find an expression for this signal, we consider the charge present at the capacitors at the end of the *previous* ϕ_1 phase, which we then can denote as $(n-1)$. We see that the sampling capacitor is connected to the input at this time, and its charge is thus

$$Q_{s,1} = C_s v_{in}(n-1) \tag{2.25}$$

Due to the virtual ground created by the op-amp, the charge at C_i is

$$Q_{i,1} = C_i v_{out}(n-1) \tag{2.26}$$

Then, in the subsequent ϕ_2 phase, C_s is discharged. However, since the virtual ground has high impedance, the charge $Q_{s,1}$ that have to flow into the C_s negative plate have to be drawn through C_i. The charge $Q_{s,1}$ is thus transferred to C_i

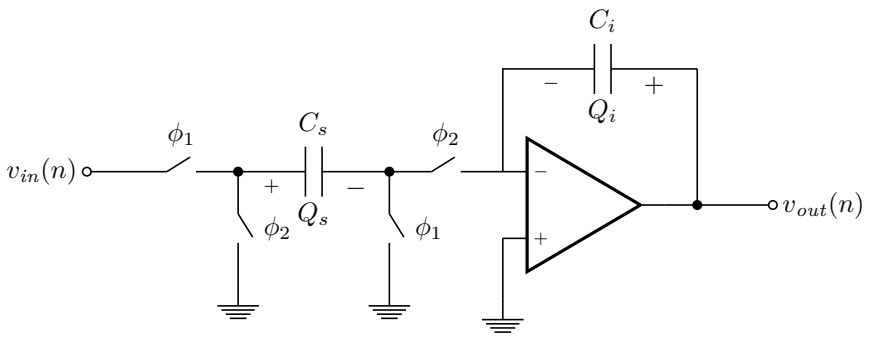


Figure 2.7: A switched-capacitor delaying integrator.

during ϕ_2, and its new charge is now

$$Q_{i,2} = Q_{i,1} + Q_{s,1} = C_i v_{out}(n-1) + C_s v_{in}(n-1) \tag{2.27}$$

We finally notice that the C_i charge does not change as the clock phase now changes to ϕ_1, meaning that the charge when $v_{out}(n)$ is sampled is still $Q_{i,2}$. The output voltage is thus

$$v_{out}(n) = \frac{Q_{i,2}}{C_i} = v_{out}(n-1) + \frac{C_s}{C_i} v_{in}(n-1) \tag{2.28}$$

Then, by taking the z-transform and rearranging, we obtain the discrete time transfer function

$$H(z) = \frac{V_{out}(z)}{V_{in}(z)} = \frac{C_s}{C_i} \frac{z^{-1}}{1 - z^{-1}} \tag{2.29}$$

We can recognize this transfer function as the transfer function of a discrete time integrator with a unit delay, and a gain of C_s/C_i. Capacitor ratios like this can be realized very accurately in integrated circuits, and is one of the main reasons why switched-capacitor circuits are very common in IC design. That is, switched-capacitor transfer functions are usually determined by capacitor ratios rather than by RC time constants, which are difficult to realize accurately in integrated processes.

2.4.1 Charge injection

As mentioned earlier, the switches in a SC circuit are implemented as either single MOSFETs, or as transmission gates. When these switches turn on and off, charge will have to enter and exit the capacitor that exists between the channel and the gate of each transistor. That is, when a switch turns on and off, charge have to enter and leave the channel, correspondingly. Since the channel is a part of the signal path of the circuit, and since nodes often are high impedance when switches turn off, the charge from the channels can end up on the capacitors in the circuit, affecting its performance.

The common way to mitigate this problem, is to ensure that the charge injection that occurs in the circuit is signal independent. Charge injection of this kind will only generate offsets, and if the circuit is differential, often only common mode offsets. On the other hand, charge injection correlated with signals will generate distortion.

To see when switches generates signal dependent charge injection, we need look at the charge stored in the switch MOSFETs when they are on. This is for an NMOS given as

$$Q_{CH} = -WLC_{ox}V_{eff} \tag{2.30}$$

This means that if V_{eff} is signal independent, the charge injection will also be charge independent. This is the case for switches that are connected to nodes with

a constant voltage, such as ground and virtual ground nodes. Say, if a switch is connected to ground at one side, it will have $V_{eff} = V_{DD} - V_{th}$ in its on state if the clock equals V_{DD} when it is high.

It is possible to let switches like this dominate the charge injection if they are turned off slightly before other switches. This can be explained by considering the two ϕ_1 switches in figure 2.7. If the switch connected to ground is turned off first, the other switch will see a very high impedance when looking into the positive C_s terminal. Because of this, only a small part of the charge in the channel of the switch will enter it.

To use this technique to mitigate charge injection, more clock signals are needed, and the clock generator must therefore be altered slightly. The clocks that turn off first are usually denoted something like ϕ_{1ad} and ϕ_{2ad}.

2.5 Noise

In all electronic circuits, stochastic signals called noise will arise due to various physical phenomena. Moreover, if the noise is due to properties of the circuit itself, and not outside disturbances, it is called inherent noise. Noise can potentially corrupt the signals in the circuit, and careful design must be carried out to prevent this. How well a circuit is designed in terms of low noise, is often expressed as a signal-to-noise ratio (SNR), which gives the ratio between the powers of the signal and the noise.

As noise is random processes, it is most easily analyzed by looking at its statistical properties. Frequency domain analysis is most common, and the noise is then expressed by its *normalized power spectral density* $V_n^2(f)$, often just called the noise spectrum. This is the stochastic counterpart of the squared magnitude spectrum of a deterministic signal, and gives information of how the noise is distributed in the frequency domain. The overall normalized noise power P_n can be found through integration of the noise spectrum.

In CMOS circuits, there is primarily two kinds of inherent noise:

Thermal noise arises because of random motions of the charge carriers due to their temperature. Thermal noise occurs in all resistors, including the resistive channel of MOSFETs, and is proportional to absolute temperature. The noise is white, which means that it has a flat noise spectrum. That is, $V_{n,thermal}^2(f) = S_0$ (where S_0 is a constant) for thermal noise.[1] Also, thermal noise has a Gaussian probability density function.

[1] More precisely, the noise spectrum starts to roll off in the THz region (otherwise, the noise power would not have been finite). However, normal electronic circuits does not have bandwidths like this, and the noise can therefore be considered white for normal applications.

Flicker noise occurs in active devices, and is due to various phenomena. One of them is that charge carriers randomly enter and leave trapping states in the semiconductor bandgap, but flicker noise is not fully understood. Flicker noise arises only when there is a DC current flowing through the device, and has a noise spectrum that is inversely proportional to the frequency. For a MOSFET in the active region, the flicker noise can be modeled as a noise voltage source in series with the gate, having noise spectrum of

$$V_{n,flicker}(f) = \frac{K}{WLC_{ox}f} \tag{2.31}$$

where K is some constant, WL is the transistor area, and C_{ox} is the gate oxide capacitance. From this, we see that flicker noise can be lowered by increasing the device area.

It is also common to compare the flicker noise to the co-existing thermal noise through the use of a flicker noise corner frequency f_k, where the flicker noise equals the thermal noise. If the thermal noise spectrum has a height of S_0, it is then possible to express the flicker noise as

$$V_{n,flicker}(f) = S_0\frac{f_k}{f} \tag{2.32}$$

In the following, we will investigate what determines the amount of noise in circuits, and how it can be controlled. We will only consider thermal noise, and the reason for this is that the flicker noise can usually be alleviated such that thermal noise dominates. This can either be done by increase of device area, or by the use of circuit techniques like auto-zeroing or chopping, where the last mentioned will be described in section 2.6.

2.5.1 Resistor noise

The thermal noise contributed by both resistors and MOSFETs in triode region can be modeled as a noise voltage source in series with a noiseless version of the actual device. The one-sided power spectral density of the noise source is $V_{nr}(f) = 4kTR$, and the arrangement is shown in figure 2.8. In this figure, the resistor is also connected to a capacitive node, having the capacitance C. This is a very common situation in integrated circuits, and it is therefore important to know the noise spectrum $V_{no}^2(f)$ in the node, as well as the total noise power P_{no}.

The key to solving this problem, is to realize that the resistor and the capacitor constitutes a first order passive low pass filter, which filters the noise coming from the noise source. The filter has the transfer function

$$H(f) = \frac{1}{1 + j\frac{f}{f_0}} \tag{2.33}$$

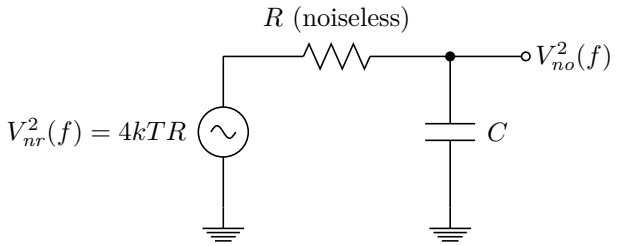


Figure 2.8: A resistor contributing noise to a capacitive node.

Furthermore, linear system theory tells us that we can obtain the output noise spectrum by multiplying the input by the squared magnitude of the transfer function. That is

$$V_{no}^2(f) = |H(f)|^2 V_{nr}^2(f) = \frac{4kTR}{1 + \left(\frac{f}{f_0}\right)^2} = \frac{4kTR}{1 + (2\pi RCf)^2} \tag{2.34}$$

where $f_0 = \frac{1}{2\pi RC}$ was used. This result can be useful on its own, but it is often more important to know the total noise power. This is for instance what is needed to compute SNR. This is found by integrating $V_{no}^2(f)$, which now represents *filtered* thermal noise. We get

$$P_{no} = \int_0^\infty \frac{4kTR}{1 + (2\pi RCf)^2} = \frac{4kTR}{2\pi RC}\frac{\pi}{2} = \frac{kT}{C} \tag{2.35}$$

That is, the noise power in the node is determined solely by capacitance and absolute temperature. Interestingly, the resistance is eliminated from the equation, despite its influence on the input noise spectrum. This is because changes in R also affect the bandwidth of the filter, such that the net effect is zero.

It turns out that this result is also valid for networks consisting of resistors/triode MOSFETs and capacitors. In this arrangement, the noise power on capacitor C_n is $\frac{kT}{C_n}$ [21]. This result often sets the lower constraint for capacitance in circuits. For instance, the sampling capacitance in switched-capacitor circuits and ADCs must be chosen such that the $\frac{kT}{C}$ noise is acceptable.

2.5.2 Noise in active circuits

The amount of noise coming from active circuits such as amplifiers, depends on both topology and device sizing. Also, such circuits generally consists of many transistors which each contribute noise, complicating the analysis. Because of this, it is common to transform all the noise contributions into a single equivalent noise source, for example at the circuit input terminal. Thereafter, the noise spectrum at for instance the output can be computed by multiplication with the corresponding $|H(f)|^2$. Finally, the noise power can be found by integration of the filtered noise.

To get an impression of what governs the noise in active circuits, we shall analyze the common source stage depicted in figure 2.9. The MOSFETs used here operates in the active region, and do therefore have non-homogeneous channels due to the pinch-off. Because of this, their noise spectrum is no longer the same as in the triode region. Instead, the noise is modeled as a noise current in the channel, having the one-sided noise spectrum

$$I_n^2(f) = 4kT\gamma g_m \tag{2.36}$$

where the γ parameter equals 2/3 for long channel devices, but can be higher for short channel lengths.

The noise sources depicted in figure 2.9 can be converted to a single noise voltage source at the input by dividing the noise spectrums by the squared transconductance of the M_1 device, and then adding the contributions. The resulting input noise voltage will give rise to currents equal to $I_{n,1}^2(f)$ and $I_{n,2}^2(f)$ at the output node, and the noise representations are thus equivalent. Specifically, we get

$$V_{ni}^2(f) = \frac{I_{n,1}^2(f)}{g_{m,1}^2} + \frac{I_{n,2}^2(f)}{g_{m,1}^2} = 4kT\gamma\frac{1}{g_{m,1}} + 4kT\gamma\frac{g_{m,2}}{(g_{m,1})^2} \tag{2.37}$$

Note that this noise source transformation assumes that the input impedance at the M_1 gate is infinite. We assume that this is true until the gain of the circuit is very low, and therefore neglect the effect. Also, note that the noise contributions has been added squared. This is because they represent uncorrelated stochastic processes.

To find the output noise spectrum, we multiply $V_{ni}^2(f)$ with the squared transfer function of the circuit. We assume that the amplifier exhibits a first order frequency response. The transfer function can thus be assumed to be

$$H(f) = \frac{A_0}{1 + j\frac{f}{f_0}} \tag{2.38}$$

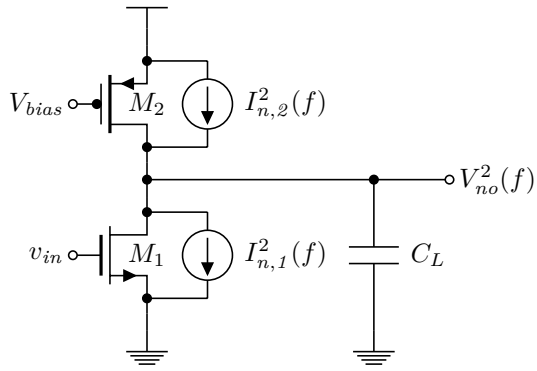


Figure 2.9: A common source amplifier with noise sources.

We correspondingly get

$$V_{no}^2(f) = |H(f)|^2 V_{no}^2(f) = \frac{A_0^2}{1+\left(\frac{f}{f_0}\right)^2}\left(4kT\gamma\frac{1}{g_{m,1}} + 4kT\gamma\frac{g_{m,2}}{(g_{m,1})^2}\right) \tag{2.39}$$

Then we integrate to find the noise power. This yields

$$P_{no} = \int_0^\infty V_{no}^2(f) = A_0^2 f_0 \frac{\pi}{2}\left(4kT\gamma\frac{1}{g_{m,1}} + 4kT\gamma\frac{g_{m,2}}{(g_{m,1})^2}\right) \tag{2.40}$$

We then utilize that the corner frequency of an amplifier with first order frequency response equals $f_0 = \frac{f_t}{A_o}$. Furthermore, we assume that the unity gain frequency is given by $f_t = \frac{g_{m,1}}{2\pi C_L}$. Substitution into equation (2.40) then yields

$$P_{no} = A_0\frac{kT\gamma}{C_L}\left(1+\frac{g_{m,2}}{g_{m,1}}\right) \tag{2.41}$$

This expression is dependent on the amplifier gain A_0, and thus suggests that the gain should be taken as low as possible to lower the noise. However, this is not the case since the signal is also amplified by A_0 (if we assume that it resides in the amplifier bandwidth), and the SNR at the output is therefore unaffected. To mitigate this inconvenience in equation (2.41), we can compute the input referred noise power. To do this, we divide P_{no} by the gain we assume that the signal will experience from the input to the output, in this case A_0. This yields

$$P_{ni} = \frac{kT\gamma}{C_L}\left(1+\frac{g_{m,2}}{g_{m,1}}\right) \tag{2.42}$$

As for passive networks, we see that the noise is also now inversely proportional to the load capacitance. Additionally, one of the terms are dependent on the transconductances of the devices, and this means that the noise is now also dependent on the amplifier design. For the common source amplifier, we see that $g_{m,1}$ should be taken much larger than $g_{m,2}$ for a low noise design. This implies high g_m/I_D for M_1, and low g_m/I_D for M_2. High g_m/I_D for M_1 also gives an energy efficient circuit, so these design goals are not in conflict here.

It is also important to note that the first and second terms in equation (2.42) corresponds to the noise contributions from M_1 and M_2, respectively. This means that the noise from M_1, which is the active device, is not dependent on the transconductances under the assumptions done during the derivation. The reason for this is the same as for the resistor discussed in the previous section: $g_{m,1}$ adjusts both the input referred noise spectral density and the circuit bandwidth in the opposite directions, such that the effects cancel.

Since the noise in active circuits depends on the circuit design, it is not as easy as for the passive networks to draw universal conclusions. Nevertheless, it should be

possible from the preceding derivations to conclude that the dominant capacitance in the circuit governs the noise to a large extent. Additionally, the active circuit itself must be designed as noise efficient as possible. How this is done depends on the topology, but it turns out that high g_m/I_D is often beneficial for the input transistors, since all the noise contributions is divided by their g_m^2 when they are input referred. Also, we observe that the input referred noise power of an active circuit can easily exceed $\frac{kT}{C}$.

2.5.3 Noise in discrete time systems

When signals are sampled at a sampling frequency f_s, aliasing will occur if the continuous time signal contains frequency content above the Nyquist frequency $f_s/2$. This also holds true for the noise in circuits that employ sampling, for instance switched-capacitor circuits and ADCs.

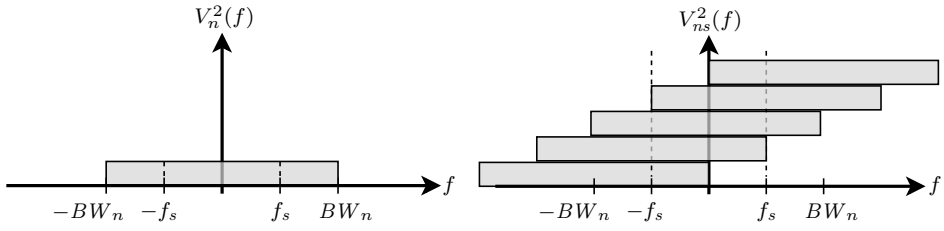
To investigate how this affects circuit performance, we consider an ideal sampler, which samples the input noise signal $v_n(t)$ into a discrete time signal $v_{ns}(n)$, at a sample rate of f_s. Assume that the input noise is white, having a noise spectrum $V_n^2(f)$, and a bandwidth of

$$BW_n = nf_s \tag{2.43}$$

The spectrum is shown in figure 2.10a. Note that the noise outside BW_n is assumed to be zero as this simplifies the following analysis. Also note that two-sided power spectral densities are used in this derivation, as opposed to one-sided spectral densities earlier in the text.

If $n > 0.5$ in equation (2.43), noise aliasing will occur due to the existence of noise outside the baseband, i.e outside the $[-f_s/2, f_s/2]$ range. This will lead to increased noise spectral density in the baseband, shown in [25] to be specifically

$$V_{ns}^2(f) = 2V_n^2(f)\frac{BW_n}{f_s} \tag{2.44}$$



(a) Before sampling. **(b)** After sampling.

Figure 2.10: Stacking of noise spectrums due to sampling. $n = 2$.

This expression can be better understood by considering $V_{ns}^2(f)$ for $n = 2$ in figure 2.10b. Because sampling is equivalent to periodic repetition in the frequency domain, replicas of $V_n^2(f)$ has appeared with a spacing equal to f_s between their centers (only the replicas that reach into the baseband are shown). Also, since the the noise is considered white, the replicas are uncorrelated and can be added directly in their squared form [25]. The result is that the noise spectrums just "stack up", and increase the power spectral density. Considering the figure, we see directly that $V_{ns}^2(f) = 4V_n^2(f)$. Equation (2.44) also yields the same answer.

The noise power before and after sampling are obtained by integrating the noise spectrums. Due to the rectangular spectrums used in this section, this can merely be done by multiplying the spectrums with the bandwidth of interest. For continuous time, we integrate over the noise bandwidth and get

$$P_n = 2BW_n V_n^2(f) \tag{2.45}$$

For the discrete time case, we are only interested in the power in the baseband. This yields

$$P_{ns} = f_s V_{ns}^2(f) = f_s 2V_n^2(f)\frac{BW_n}{f_s} = 2BW_n V_n^2(f) \tag{2.46}$$

This reveals that $P_n = P_{ns}$, despite the increased power spectral density after the sampling. This shows us that sampling "compresses" all the continuous time noise noise into the baseband, but will not add extra noise power. Consider for instance figure 2.8, and assume the the resistor represents the on-resistance of a sampling switch that subsequently closes. The noise power then sampled onto the capacitor as a discrete time signal will in this case equal the noise power when the switch conducts, i.e. $\frac{kT}{C}$.

2.6 Chopper stabilization

Chopper stabilization is a technique where flicker noise and DC offset problems are mitigated by applying a modulation scheme to the circuit. The conceptual working principle is shown in figure 2.11. Here, the amplifier, which suffers from flicker and DC offset, is put in the middle of two mixers which receive a modulating signal $m(t)$. We assume that this signal is an ideal unit square wave with 50 % duty cycle, no DC offset, and a frequency f_{chop}. That is, $m(t)$ alternates between the values 1 and -1.

We first consider what happens to the signal: When $v_{in}(t)$ enters the first mixer, it is multiplied by $m(t)$. This operation is equivalent to convolution in the frequency domain, and the result is that the frequency content of $v_{in}(t)$ is shifted up to all the harmonics of $m(t)$. Under the assumptions done about $m(t)$, these harmonics lie at nf_{chop} where $n = 1, 3, 5, ...$, and have an amplitude proportional to $1/n$. After this, the modulated signal is amplified, before it multiplied by $m(t)$ once more. This

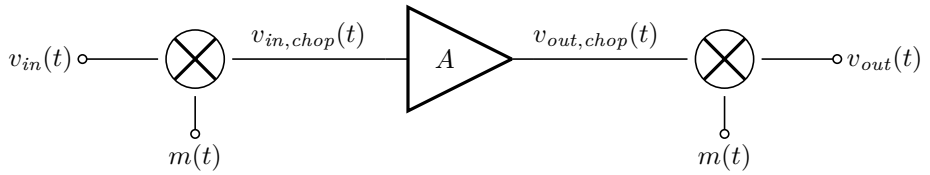


Figure 2.11: Amplifier with chopper stabilization.

operation will ideally demodulate the signal back to the baseband and remove all harmonics introduced by the modulation process, giving $v_{out}(t) = Av_{in}(t)$. This is only true if the amplifier is assumed to have infinite bandwidth and no delay. This is most easily understood by realizing that under these assumptions, the net effect on $v_{in}(t)$ due to the modulation is just a multiplication by $m^2(t)$. If $m(t)$ just alternates between 1 and -1, the squared value equals just one, and the modulation has no net effect on $v_{in}(t)$. When the amplifier has finite bandwidth, there will be some residual harmonic content around the even harmonics of f_{chop} after the demodulation [23].

When it comes to the noise and offset of the amplifier, this will only pass through the last mixer, and is therefore modulated up to the $m(t)$ harmonics, without being demodulated again. If we assume that the flicker has a corner frequency of f_k, it will now reside in the bands $f_{chop} \pm f_k, 3f_{chop} \pm f_k, ...$ rather than in the baseband. Correspondingly, the DC offset will be transformed into harmonics at $f_{chop}, 3f_{chop}, ...$ This means that if an appropriate f_{chop} is chosen, flicker and offset can be separated from the frequency content of the signal and thus easier be dealt width by some kind of filtering.

For a quantitative analysis of the chopped noise, we consider [23]. Here, the chopped noise spectrum is given (with our notation) as

$$V_{n,chop}^2(f) = \left(\frac{2}{\pi}\right)^2 \sum_{\substack{n=-\infty \\ n \text{ odd}}}^{+\infty} \frac{1}{n^2} V_n^2(f - nf_{chop}) \tag{2.47}$$

That is, the original noise spectrum $V_n^2(f)$ is shifted to the odd harmonics of f_{chop}. Also, each shifted version is scaled by $1/n^2$, and not $1/n$ since we work with squared, i.e. power spectrums.

Furthermore, if we assume that we have both white and thermal noise, such that $V_n^2(f) = S_0(1 + \frac{f_k}{f})$, the summation of equation (2.47) is shown in [23] to yield

$$V_{n,chop}^2(f) = S_0(1 + 0.8525\frac{f_k}{f_{chop}}) \tag{2.48}$$

if $|\frac{f}{f_{chop}}| \leq 0.5$ (i.e we consider the baseband), and the bandwidth of the amplifier is much larger than f_{chop}.

The first term in this equation represents the white noise, which we see are unaffected by the chopping. The second term represents the residual flicker noise, which is due to the "tails" of the flicker noise spectrum being modulated into the baseband. We see that this component now also looks white, and is dependent on the ratio between f_k and f_{chop}. If the chopping frequency is chosen such that the ratio is unity, white and residual flicker noise are comparable. From this we conclude that the flicker noise effectively can be alleviated in the baseband by chopping. However, it is only moved to other frequencies, and filtering of some kind is thus necessary to actually remove the flicker noise.

In practice, chopper stabilization is implemented in differential circuits by using switches between the positive and negative branches. These switches are clocked such that the branches alternates between being cross coupled and "normally" coupled. The cross coupling will then be equivalent to a multiplication by -1.

CHAPTER 3

High level design

The aim of this chapter is to describe the NS-SAR and loop filter design at a high level, before the actual circuit implementation is described in chapter 4. This will hopefully help bridging the gap between the specialization project [19] and this thesis, and also make it easier to describe the actual circuitry afterwards.

First, the loop filter topology will be presented, and its high level design parameters will be given and discussed. Then, the overall noise budget of the NS-SAR will be considered, including noise contributions not investigated in [19]. After this, the break-up of the total noise budget will be analyzed in terms of energy efficiency, before considerations regarding settling time in the loop filter concludes this chapter.

3.1 Filter topology

In the project thesis, the NS-SAR filter topology used in [11], as well as three other topologies were compared. Under the assumptions done in the comparison, all the topologies were found to give the about the same energy efficiency for the specification presented in chapter 1 of this thesis, and a definite conclusion was therefore not drawn. However, the topology presented in [11] was found to be marginally better, and was additionally the simplest topology considered. Specifically, this

topology uses only one integrator, while the others uses two or three integrators. The topology is therefore chosen for implementation in this thesis.

A linear model of the NS-SAR with loop filter is shown in figure 3.1. Everything inside the dashed box is the loop filter, while all the signal flow on the outside is inherent to the NS-SAR structure, as discussed in section 2.2 and appendix A. In the physical circuit, the signal $q(n)$ feeding the loop filter circuit is extracted from the SAR DAC plates after each conversion, while the filter output $y(n)$ is connected to an extra input pair of the SAR comparator. The loop filter itself consists of a delaying switched-capacitance integrator (the right half inside the dashed box), and an extra delay path (to the left), which is realized passively by using some extra capacitors to feed the integrator. Further implementation details are the topic of chapter 4.

The loop filter has the transfer function

$$H(z) = \frac{b_1 z^{-1} + b_2 z^{-2}}{1 - z^{-1}} \tag{3.1}$$

which gives the whole NS-SAR the NTF

$$\mathrm{NTF}(z) = \frac{1 - z^{-1}}{1 + (b_1 - 1)z^{-1} + b_2 z^{-2}} \tag{3.2}$$

That is, the ideal NTF has a zero at DC, and a pole pair that can be freely chosen by the coefficients.

3.2 High level design parameters

The most important high level design parameters for the NS-SAR are the number of bits B in the SAR, the oversampling ratio (OSR), as well as the coefficients b_1 and b_2

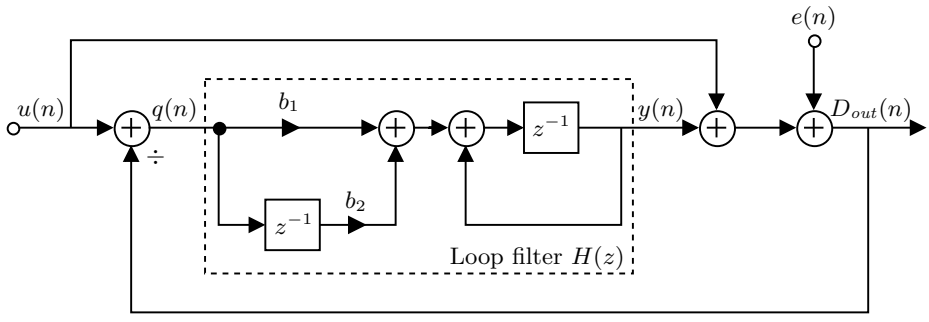


Figure 3.1: Linear model of the NS-SAR, including the chosen loop filter.

in the filter. When it comes to the two first mentioned, the behavioral simulation framework developed in [19] is able to pick the pair of OSR and B values that yields a given accuracy (i.e. a specified ENOB value) in the most energy efficient manner. Using this framework, $B = 9$ bit is found when the ENOB specification of 11 is entered, and a corresponding OSR of around 2.6. This is true both for the energy consumption estimates done in [19], and if the simulated energy consumption from the ADC designed in this thesis is used as input to the simulation framework.

However, since only quantization noise and comparator noise were considered in [19], a practical implementation using these values for B and OSR will fail. First and foremost, this is because the loop filter itself as well the SAR sample-and-hold also contributes noise to the system, and most of it is not noise-shaped. Therefore, these contributions need to be given a significant part of the total noise budget. In addition to this, second order effects and distortion will also degrade the performance, and this must be made room for. Due to this, $B = 9$ bit and OSR $= 8$ is chosen for the implementation.

This is a much higher OSR value than proposed by the behavioral simulation framework, and the quantization noise as well as the comparator noise will thus be much smaller and not dominate the total noise of the ADC. It makes sense to do it this way, since the noise-shaping action can lower the quantization and comparator noise very efficiently. Specifically, the chosen loop filter ideally increases the ENOB with 1.5 bit or more per oversampling octave. That means that half the bandwidth can be traded for around an eightfold decrease of the noise power. In comparison, for not noise-shaped noise contributors, the capacitance needs to be doubled to halve the noise. That is, a twofold increase in energy consumption is traded for a twofold noise power decrease. Therefore, it makes sense to use the noise-shaping "for what it's worth", and let the unshaped noise contributors dominate. Both the noise budget as well as the choice of OSR $= 8$ will be quantitatively analyzed later in this chapter.

The filter coefficients are chosen to be $b_1 = 2.9$ and $b_2 = 0.9$. These values are somewhat larger than those found by the behavioral simulation framework (given $B = 9$ and OSR $= 8$), and this yields a slightly more aggressive filter (the simulation framework chooses on the contrary conservative values). The coefficients have nevertheless been found stable for input amplitudes close to full scale when the loop filter gain is limited by the practical circuit design, i.e. by finite amplifier gain.

The NTF magnitude response and pole/zero chart computed for the NS-SAR in Matlab, given all the chosen parameters, are shown in figure 3.2. The DC gain of the filter, which is also modeled in the behavioral simulations, is set to 40 dB in this figure. This is the same filter gain as achieved in the finished circuit, and this yields a peak ENOB value of 14.4 in behavioral time domain simulations with only quantization noise enabled. It is further observed in these simulations that higher gain values do not increase the performance further, and this is probably due to the low OSR, which makes the gain requirement relaxed (was discussed in [19]). On

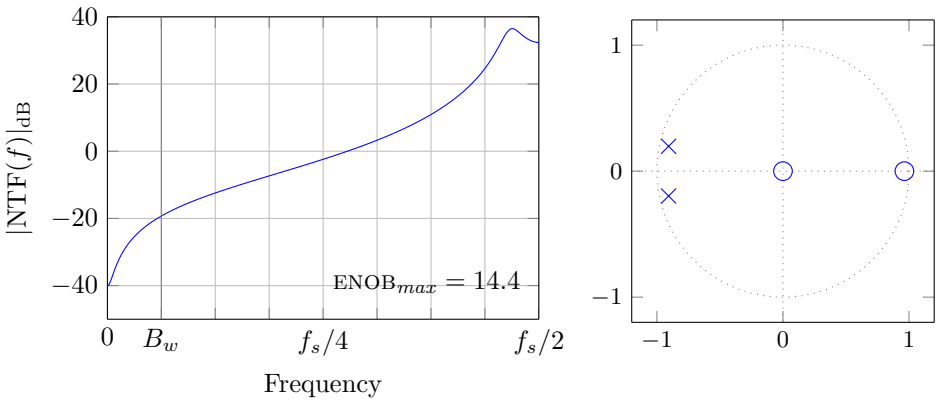


Figure 3.2: NTF amplitude response and pole/zero chart for the loop filter with DC gain equal to 40 dB, and only quantization noise included.

the contrary, significantly higher values are found to decrease the maximum stable input amplitude due to the coefficients used.

3.3 Noise budget

The noise of the ADC will now be analyzed in more detail. Specifically, it will be shown how much of the noise budget that is available to the unshaped noise sources as a consequence of the increased OSR value chosen for the ADC.

The main noise contributors and their place in the NS-SAR linear model are shown in figure 3.3. Here, all the noise enters through the gray adders, which there are three of. The noise contributions are specifically:

- To the right, the quantization noise $V^2_{n,Q}(f)$ enters together with the comparator noise $V^2_{n,cmp}(f)$. The entering point is the same as that earlier used for the quantization error $e(n)$. As discussed in section 2.2 and appendix A, the noise that enters here is noise-shaped by the noise transfer function.
- To the far left, a noise contribution $V^2_{n,S/H}(f)$ enters due to the noise from the sample and hold switch of the SAR. When the switch samples, a sample from the noise source is added to the SAR DAC in the same manner as the input signal sample. The noise contributor can therefore be viewed as connected to the input, and the noise will therefore propagate to the output according to the signal transfer function, which is unity. The noise is thus not noise-shaped.
- There exists various noise contributors inside the loop filter. To treat these as one quantity, they are referred to the filter input as the noise source $V^2_{n,LF}(f)$.

The transfer function from this entering point to the output is $\frac{H(z)}{1+H(z)} \approx 1$ when $H(z)$ is large, as it is in the signal band. It is therefore a good approximation to regard also this noise contribution as entering through the input terminal and propagating straight to the output. The input referred loop filter noise is thus not noise-shaped.

From this, we can conclude that in-band output noise power of the ADC will consist of a contribution $P_{n,sh}$ which is noise shaped, in addition to a contribution $P_{n,us}$ which is not affected by the noise shaping. If we also take distortion into consideration through a contribution P_{HD}, the overall SNDR of the ADC can be expressed as

$$\text{SNDR} = \frac{P_{s,max}}{P_{n,us} + P_{n,sh} + P_{HD}} = \frac{P_{s,max}}{P_{nd,tot}} \tag{3.3}$$

where $P_{s,max}$ is the power of the maximum input signal (i.e. the one that yields the highest SNDR in simulations), which we assume is a sinusoid. Also, the SNDR is regarded as a power ratio rather than a dB value in this equation.

From the above relation, it is seen that the unshaped noise power is given as

$$P_{n,us} = P_{nd,tot} - (P_{n,sh} + P_{HD}) \tag{3.4}$$

Furthermore, if the maximum signal power is known, the maximum allowed $P_{nd,tot}$ can also be found from equation (3.3) if SNDR is set according to the design specification.

We can now put some numbers into these equations to gain insight into how the noise budget is divided in the designed ADC. First, we obtain the maximum signal power, which comes directly from the maximum input amplitude through $P_{s,max} = \frac{V_{s,max}^2}{2}$. In the way the used SAR is designed, this maximum input amplitude shall in theory equal a value close to its reference voltage, which is 0.8 V. However, the maximum input amplitude of the designed NS-SAR is found to be approx. $V_{s,max} = 550\,\text{mV}$, which is much smaller than the reference voltage. The reason for

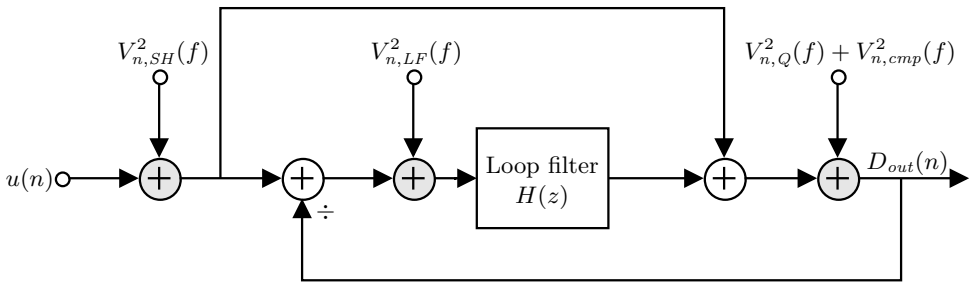


Figure 3.3: The main noise contributors in the NS-SAR shown in the linear model.

this is parasitic capacitance in the DAC. This decreases the magnitude of all the DAC coefficients, which in turn is equivalent to decreasing V_{ref}. The designed ADC thus experiences a lower equivalent reference voltage, which is found to be slightly less than 600 mV. This is much closer to the maximum input amplitude.

Having $V_{s,max}$, we obtain the total allowed noise power from equation (3.3) if the accuracy specification of ENOB = 11 ⇒ SNDR = 68 dB is also inserted. We get

$$P_{nd,tot} = \frac{P_{s,max}}{\text{SNDR}} = 24\ \text{nV}^2 \tag{3.5}$$

where the SNDR was inserted as a power ratio.

Now, the shaped noise power and distortion power is most easily found from a simulation. Either behavioral, or more realistically, from a SPICE simulation of the circuit. The last mentioned is done by doing a transient noise simulation with only the comparator enabled as noise contributor, and the result for the designed ADC is $P_{n,sh} + P_{HD} = 3.8\ \text{nV}^2$. The noise power available for the unshaped noise then follows from equation (3.4) as

$$P_{n,us} = 24\ \text{nV}^2 - 3.8\ \text{nV}^2 = 20.2\ \text{nV}^2 \tag{3.6}$$

This is 84 % of the total noise budget, and this share is a consequence of the increase of OSR, discussed in section 3.2.

Finally, we want to estimate the noise power available for the loop filter. To do this, the noise power used by the sample-and-hold first have to be found. This can be estimated quite easily through a calculation, if we realize that the situation where the ADC input signal is sampled onto the DAC array is identical to the passive sampling situation discussed at the end of section 2.5. It was here concluded that the sampled noise power in cases like this equals kT/C, and this is therefore also the case for the sample-and-hold. However, some of the kT/C noise power (which is distributed from DC to $f_s/2$) will be outside the signal band due to the oversampling. Additionally, the implementation of the SAR is differential, and there will thus arise noise power from each of the branches. These two power contributions add squared since they are uncorrelated. Taking all this into account, the sample-and-hold noise power is estimated as

$$P_{n,S/H} = \frac{1}{\text{OSR}} \frac{2kT}{C_{DAC}} \tag{3.7}$$

where C_{DAC} is the total DAC capacitance in one of the branches, including parasitic capacitance. In the designed ADC, this is about $C_{DAC} = 280\ \text{fF}$, and this yields $P_{n,SH} = 3.7\ \text{nV}^2$ at $T = 300\ \text{K}$.

We can now obtain the noise power theoretically available for the filter as

$$P_{n,LF} = 20.2\ \text{nV}^2 - 3.7\ \text{nV}^2 = 16.5\ \text{nV}^2 \tag{3.8}$$

Estimates like this have been used through the design process.

3.4 Optimum division of the noise budget

It should now be clear that the largest part of the total noise budget is allocated to the unshaped noise sources, and that this is probably a good approach. Still, a more quantitative analysis is needed to in detail understand how energy efficient the noise budget break-up is. This will be performed in the following.

Assume that the circuitry that gives rise to the unshaped noise uses the power P_{us}. In the NS-SAR, this power consumption term will be constituted by the loop filter and the DAC. The last mentioned have to be included since it is the DAC capacitance that have to be increased to decrease the sample-and-hold noise, and this in turn increases the power consumption used during DAC switching. Further assume that unshaped noise have to be adjusted by capacitance adjustment, and that the power consumption therefore is inversely proportional to the unshaped noise budget share. Additionally, it will be inversely proportional to the OSR, since unshaped noise is also oversampled. We can then write P_{us} on the form

$$P_{us} = \frac{B_{ref}}{\mathrm{OSR} \cdot \alpha} \tag{3.9}$$

where α is the allocated noise share given as factor relative to one, and B_{ref} is a constant that is chosen such that the equation yields correct absolute values. We see that $\alpha = 0$ implies infinite power since the unshaped noise then have to equal zero.

To estimate α the following equation has been developed:

$$\alpha = 1 - 10^{-\gamma \log_2(\mathrm{OSR}_I)/10} \tag{3.10}$$

α is here given as the difference between the total relative noise share, i.e. unity, and the noise share used by the shaped noise contributions. This is computed from a parameter γ, which tells how effective the noise shaping is, in dB per OSR octave. Also the parameter OSR_I is used. This is a factor that tells how much the OSR is increased from the point OSR_0, which is the point where the whole noise budget is used by shaped noise. That is

$$\mathrm{OSR} = \mathrm{OSR}_0 \cdot \mathrm{OSR}_I \tag{3.11}$$

As an example, assume that the noise shaping gives us 6 dB per octave (approx. 1 bit). Furthermore, assume that the entire noise budget is used by shaped noise when the OSR is 4, and that the OSR is now increased to 8. This means that $\mathrm{OSR}_i = 2$. Equation (3.10) then yields $\alpha = 0.75$, which makes sense since the shaped noise is decreased by a fourfold with the numbers given. Note that $\mathrm{OSR}_I = 1$ yields $\alpha = 0$, since all the noise is then allocated to the shaped contributors.

To evaluate the energy efficiency of the noise budget break-up, we can then make use of the commonly used Walden figure-of-merit, defined as

$$\mathrm{FOM} = \frac{P}{2^{\mathrm{ENOB}} \cdot 2B_w} \tag{3.12}$$

where P is the power consumption of the whole ADC, ENOB is the accuracy given as effective-number-of-bits, and B_w is the width of the signal band. The computed FOM will have the units of Joule per conversion-step, and smaller values thus mean better energy efficiency.

For the ongoing analysis, we can modify the FOM formula to

$$\text{FOM} = \frac{P_{us} + P_{rest}}{2^{\text{ENOB}} \cdot 2\frac{B_w}{\text{OSR}_I}} \tag{3.13}$$

where P_{rest} is the part of the ADC power consumption which is independent of the unshaped noise share. Also, the extra increase in OSR is reflected as a reduction in signal bandwidth.

If the expression for P_{us} is now inserted in the above formula, FOM can be swept as a function of OSR_I to find the optimum noise break-up. For this to be possible in practice, B_{ref} as well as γ have to be estimated from circuit level simulations or by other means. Specifically, B_{ref} must be found on the form

$$B_{ref} = P_{ref} \cdot \text{OSR}_{I,ref} \cdot \alpha_{ref} \tag{3.14}$$

where P_{ref} is the power at the design point given by $\text{OSR}_{I,ref}$ and α_{ref}. By choosing B_{ref} like this, P_{us} will be normalized to P_{ref} in that point. Finally note that the 2^{ENOB} term in the FOM formula is a constant during the sweep, since it is assumed that the extra accuracy that is obtained through OSR adjustment is used to relax the unshaped noise requirements.

For the designed ADC simulated post-layout, the result of a FOM-sweep is shown in figure 3.4. Through the use of equation (3.10), α is used as the x-axis variable rather than OSR_I. We see that under the used assumptions, optimal energy efficiency is obtained if slightly more than 80 % of the noise budget is allocated to the unshaped noise contributors. This percentage is close to the 84 % that was found in section 3.3, and the choice of an OSR of 8 is therefore proved to yield an energy efficient system.

3.5 Settling time requirements

As mentioned in section 3.1, the circuit implementation of the loop filter is built around a switched-capacitance integrator. For the ideal discrete time transfer function for such a circuit to be valid, the signals have to settle fully in each clock phase. Otherwise the performance will be impacted. This requirement is in general not fulfilled, since the elements constituting the circuit have finite bandwidths and slew rates. If the settling errors that occur are signal signal dependent due to slewing behavior, distortion will occur. In the designed loop filter, however, this does not seem to be a problem because of the low signal swing of the SAR residue. The

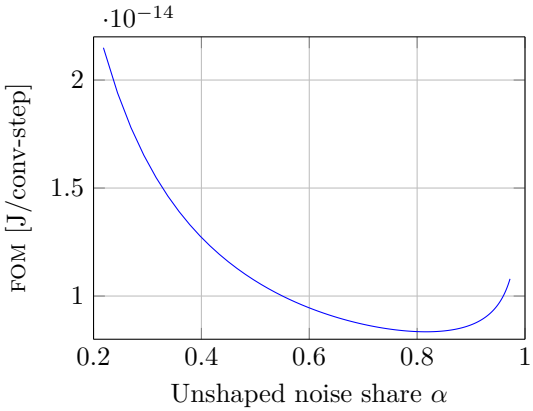


Figure 3.4: Energy efficiency analysis for the noise budget break-up in the designed ADC.

limiting factor is instead linear settling errors, and their effect on the system will thus be analyzed in the following.

What happens to the filter if settling operations do not finish, is conceptually the same as what would happen to the simple SC integrator analyzed in section 2.4. Here, we first have a phase where an input signal have to settle over some sampling capacitance, followed by a phase where the stored charge due to this signal is transferred to some integration capacitance. We realize that if a linear settling error occurs in any of the phases (i.e. the relative settling error is independent of the input signal), this is equivalent to multiplying the input signal of the ideal circuit by some gain $\beta < 1$, which represents the attenuation due to the settling error. Since we now have the input-output relation $V_{out}(z) = H(z) \cdot \beta V_{in}(z)$, we can use a new transfer function $H'(z) = \beta H(z)$ to describe the system with settling error.

If we use this technique on the transfer function for the filter, given in equation (3.1), and then find the equivalent NTF, we get

$$\text{NTF}(z) = \frac{1}{1 + \beta H(z)} = \frac{1 - z^{-1}}{1 + (\beta b_1 - 1)z^{-1} + \beta b_2 z^{-2}} \tag{3.15}$$

That is, only the pole polynomial is affected, but not the zero at DC.

The impact of β on the SNDR is most easily studied in behavioral simulations. This has been carried out by applying β to the transfer function that was shown in figure 3.2, and then performing time domain simulations for different β-values. The resulting SNDR-values are plotted in figure 3.5, normalized to the SNDR without settling error. This plot reveals that the SNDR is degraded by about the same amount as the value of β in dB.

This relation makes the settling time requirements for the circuit rather relaxed.

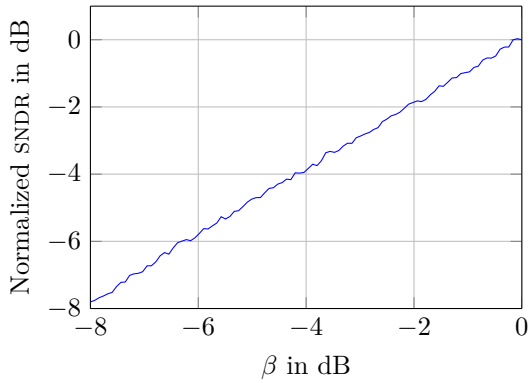


Figure 3.5: The impact of linear settling errors on the SNDR, given as a function of the settling attenuation parameter β.

To show this, we assume that the settling in the circuit obeys equations like

$$v(t) = V_0(1 - e^{-t/\tau}) \tag{3.16}$$

Here, the voltage $v(t)$ is about to settle to the steady state value V_0, and this happens according to the settling behavior specified by the factor inside the parenthesis. At a certain point in time, this factor can also be interpreted as the instantaneous settling error, and it will therefore make sense to equate it to β. By doing this and changing the unit of time from seconds to numbers of passed time constants, we get

$$1 - e^{-k} = \beta \quad \Rightarrow \quad k = -\ln(1 - \beta) = -\ln(1 - 10^{\beta_{\text{dB}}/20}) \tag{3.17}$$

So, say that we choose to tolerate -0.5 dB SNDR degradation. We then use the same value for β_{dB} and obtain a settling time requirement of $k = 2.88$ time constants. Correspondingly, we obtain $k = 3.56$ time constants if we choose a stricter β_{dB} of -0.25 dB. From this, we conclude that three time constants of settling yields a pretty reasonable performance. This is a fairly modest requirement compared to, say, if it was necessary to settle within the equivalent $V_{LSB}/2$ of the whole system.

Finally, it should be pointed out that the simulations performed by the behavioral simulation framework lacks unshaped noise contributors, and the analysis above therefore only shows how the settling time impacts the shaped noise sources of the system. As we have seen, these should not dominate the noise budget, and we should thus expect different results when unshaped noise is also taken into account. However, the input referred noise of the filter will also be affected when the settling error is increased. This is because all the noise sources inside the filter are in general not affected by the settling time, so when the gain of $H(z)$ decreases due to β, the input referred noise increases. So if we assume that the noise seen at the filter output is unchanged when the settling time is changed, the input referred noise power will increase by an amount $1/\beta^2$. This assumption is probably not entirely

true, and will for example depend on how the settling time is changed. Still, we choose to use it as a worst case estimate.

If this estimate is true, it leads to the reasonable approximation that the input referred filter noise source as well as the shaped noise at the ADC output increases by the same amount when β is changed. Since these contributors together uses most of the noise budget (only the sample-and-hold noise is excluded), it is concluded that the overall SNDR degrades by an amount approximately equal to β_{dB}.

CHAPTER 4

Implementation

The purpose of this chapter is to give a detailed description of the loop filter implementation, as well as its interfacing towards the SAR.

First, the loop filter will be presented at the switched-capacitor (SC) abstraction level, where the schematic consists of amplifiers, switches and capacitors. Thereafter, the focus will be directed towards the transistor level implementations of the amplifiers used in the circuit. That is an inverter used as main amplifier in the SC circuit, and a buffer used to extract the residue from the SAR. After this, some considerations regarding the noise of the filter is discussed, before the non-overlapping clock generator is presented. Finally, the physical layout of the whole NS-SAR will be presented.

4.1 Loop filter circuit

The switched-capacitor circuit implementation of the loop filter is depicted in figure 4.1. As a reference, the supply voltage and the capacitor sizes (as measured in the layout) are given in table 4.1. Most of the capacitors are sized according to noise requirements, and all of them will be mentioned more specifically throughout this chapter. When it comes to the switches in the schematic, all are implemented as NMOS switches except the inverter reset switches, which are transmission gates.

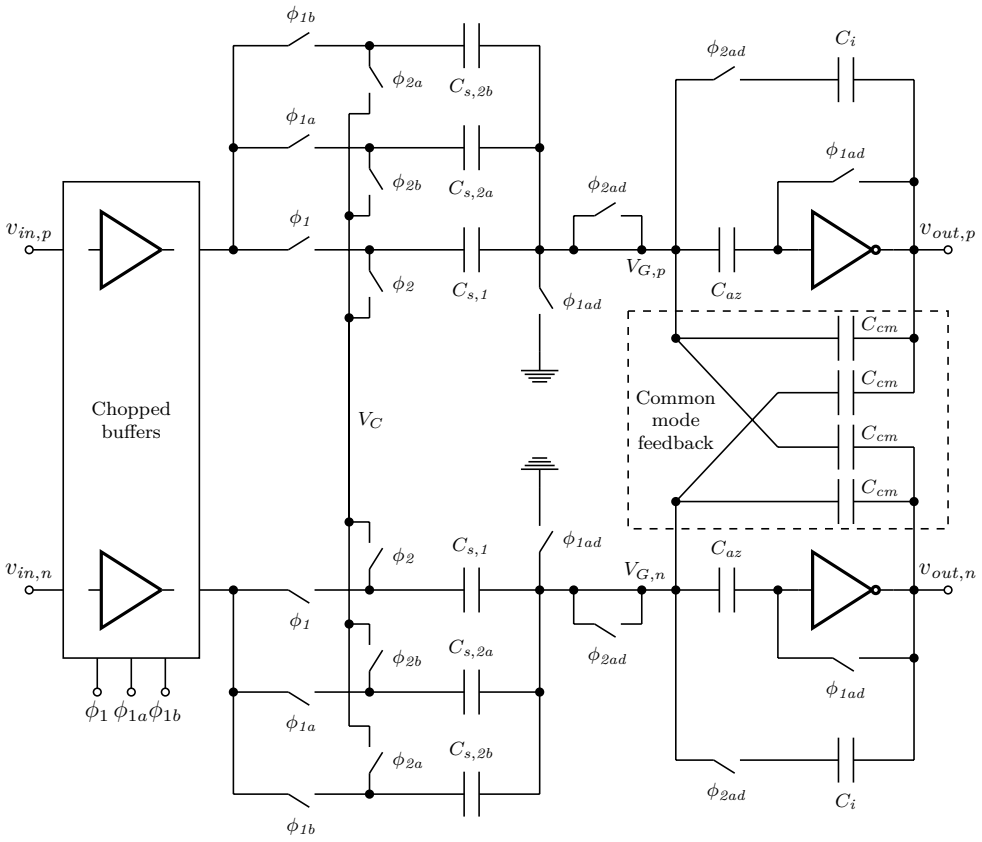


Figure 4.1: Loop filter circuit schematic.

Basically, the operating principle of the loop filter is similar to the simple integrator described in section 2.4, where a sampling capacitor C_s is charged during a phase ϕ_1, before the charge is delivered to an integration capacitor during a phase ϕ_2. This is also the case in the loop filter circuit, but more clocks and a bank of sampling capacitors are now used to realize two input paths, having different delay. These input paths were also shown in the signal flow schematic in figure 3.1, and exactly how the SC circuit realizes the signal flow depicted there (i.e. realizes $H(z)$) will be shown later in this section. Also, the circuit is pseudo-differential. Compared to single-ended implementations, this improves the performance in many ways, for instance by lowering even order harmonics and by giving protection against external interference noise.

The timing diagram for the loop filter and SAR interfacing is shown in figure 4.2, and the tasks going on in the filter and SAR at different times are also indicated. First and foremost, we have a standard pair of non-overlapping ϕ_1 and ϕ_2 clocks, denoting sampling and integration respectively. More specifically, these clocks con-

Table 4.1: Design parameters for the loop filter.

Parameter	Value
V_{DD}	0.8 V
$C_{s,1}$	765 fF
$C_{s,2}$	239 fF
C_i	265 fF
C_{az}	852 fF
C_{cm}	11 fF

trols the charge transfer to and from the capacitor pair $C_{s,1}$ directly. The charge flow through this capacitor pair constitutes the normal path into the integrator, giving the signal a unit delay from input to output. Additionally, we have a clock pair ϕ_{1a}, ϕ_{2a}, and a clock pair ϕ_{1b}, ϕ_{2b} that have every other pulse nulled, compared to ϕ_1, ϕ_2. These clocks controls the charge transfer to and from the pairs $C_{s,2a}$ and $C_{s,2b}$, which together works as an interleaved capacitor pair implementing the input path having extra delay. Consider for example $C_{s,2a}$ being charged at phase ϕ_{1a}. The stored charge is then not transferred to C_i before at ϕ_{2b}, one and a half clock cycle later. The signal path therefore have one extra cycle of delay compared to the path through $C_{s,1}$. To realize this path, two capacitor pairs operated in an interleaved manner are needed because one pair is "kept busy" for two clock cycles. $C_{s,1}$ however, is emptied in the phase after it is charged, and therefore not needs interleaving. This is different from the implementation done in [11], which uses interleaved capacitors for both input paths. This is not necessary, and area is therefore saved in the implementation carried out in this thesis, compared to [11].

The clocks ϕ_1 and ϕ_2 also exists in advanced versions named ϕ_{1ad} and ϕ_{2ad}, falling down before their counterparts. As discussed in section 2.4.1, these clocks are used to minimize signal dependent charge injection by letting switches connected to DC nodes turn off first. For the dummy switch and inverter reset switch, the advanced clocks are just used to ease the routing in the layout, since the whole "right part" of the circuit, laying physically at one place in the layout, then only needs access to the advanced clocks.

The timing towards the SAR is also shown in figure 4.2, and works as follows: When the SAR has finished a conversion, it rises the signal DONE, which means that the SAR is finished and idle, and that the DAC plates hold a valid residue. This signal triggers a transition from ϕ_2 to ϕ_1 in the filter, which thus enters sampling mode to sample the residue onto the currently used C_s capacitors. The system stays in this state until the external sample clock called SMP rises to denote that a new conversion should be started. This causes the filter to change to ϕ_2 to enter integration mode, and the SAR sample-and-hold will also start to conduct. In this transition, it is important that the ϕ_1 clocks fall before the SAR starts to sample and corrupts the residue on the DAC plates. Therefore, the sample clock connected to the SAR is gated until ϕ_1 has fallen, and is then sent to the SAR sample-and-hold

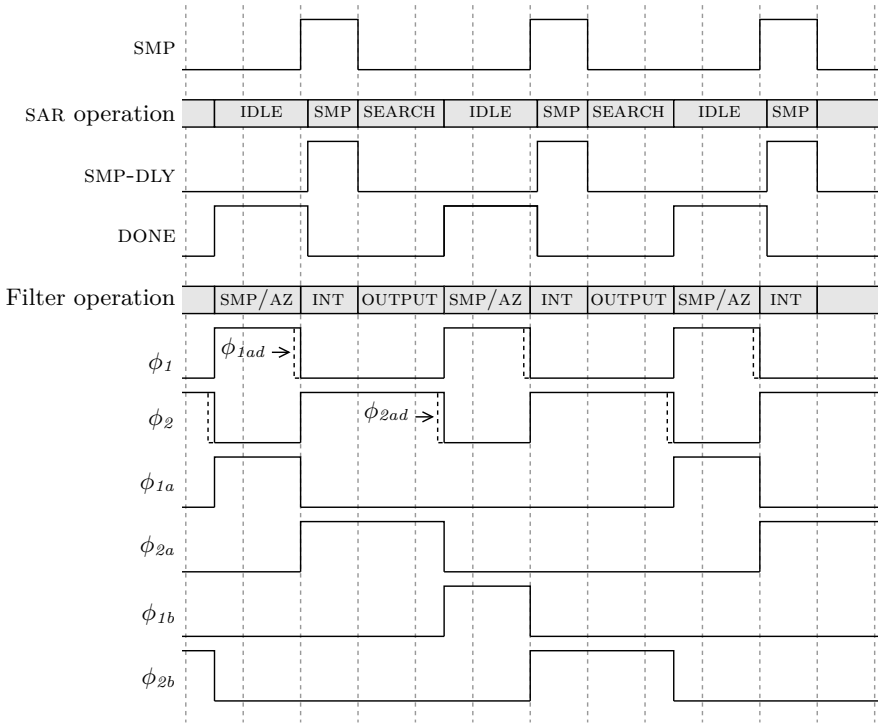


Figure 4.2: Timing diagram.

as the signal SMP-DLY. When the SAR samples, the filter has time to do charge transfer until the binary search starts at the falling sample clock. The output from the filter then has to be valid since it is used during the binary search comparisons.

4.1.1 Auto-zeroed inverters as amplifiers

Instead of a standard OTA, inverters are employed as amplifiers in the loop filter. The use of inverters as amplifiers in integrators/loop filters is elaborately presented and analyzed in [26], and the use of inverters together with auto-zeroing (soon to be explained) in this thesis is conceptually similar to what is presented there. One of the reasons to use inverters instead of OTAs, is that inverters are better suited for low voltage operation [26]. Also, inverters are energy efficient since they operate in a push-pull manner such that both the main NMOS and PMOS work as transconductive devices.

An inverter can be viewed as an amplifier having only an inverting input. Because of the lacking positive input, it will not provide a virtual ground on its own when connected in negative feedback. Instead, the inverter will settle to a voltage near

its offset voltage V_{off} (typically around $V_{DD}/2$) when the output is fed back to the input, or more specifically to $\frac{A}{1+A}V_{off} \approx V_{off}$ (A is here the open loop gain of the inverter) [26]. This offset voltage is not suited for use as a virtual ground, because it is not a well defined quantity. To mitigate this problem, an auto-zeroing scheme is used to cancel V_{off} when the inverter is used for charge transfer in the ϕ_2 phase.

The C_{az} capacitors shown in the main schematic are used for the auto-zeroing. In ϕ_1, these are connected to ground on their left side and to the inverters being held in reset on their right side. Since the inverters provide V_{off} in this configuration, the offset voltages are sampled on C_{az} at the end of ϕ_1 due to the unity gain feedback. In ϕ_2, the inverter inputs are still held at V_{off} because negative feedback is now present around the outer loop. Also, the right side of C_{az} are high impedance, and the charge from the ϕ_1 phase are thus ideally unchanged. This means that the C_{az} capacitors now work as floating voltage sources of magnitude V_{off}, and cancel the inverter offsets when looking into C_{az} from the nodes $V_{G,p}$ and $V_{G,n}$. The inverters together with the C_{az} capacitors are therefore equivalent to offset free amplifiers in the ϕ_2 phase, and the V_G nodes therefore work as virtual grounds.

In addition to V_{off}, the instantaneous input referred noise voltages of the inverters are also sampled onto C_{az} during ϕ_1. Similarly to the offset, this means that the equivalent input referred noise seen from $V_{G,p}$ and $V_{G,n}$ during ϕ_2 will equal the difference between the real input referred noise of the inverter, and the sample of the noise at C_{az}. However, since the instantaneous noise voltages are changing all the time, we will not cancel the noise likewise as the offset. Nevertheless, auto-zeroing will reduce low-frequency noise, and especially flicker noise, which dominates at low frequencies [23]. That this is true can be realized by thinking of the auto-zeroing operation as a differentiator, which subtracts a recent sample of the noise from its instantaneous value. The noise is therefore high-pass filtered by the differentiation operation. The detailed frequency response of auto-zero circuits is given in [23]. As an approximation, it is also given that if $d\pi\frac{f}{f_s} \ll 1$, where d is the duty cycle of ϕ_2, the frequency response can be given as

$$|H_0(f)| \approx d\pi\frac{f}{f_s} \tag{4.1}$$

This transfer function has a zero at DC and increases linearly. To estimate its effectiveness on noise attenuation, we can compute its squared value at our signal band edge of 2 MHz. This yields a value of 0.019, and we thus conclude that the auto-zeroing removes low frequency noise from the amplifier very efficiently. However, noise from frequencies over $f_s/2$ will be aliased down into the baseband due to the sampling in the circuit. In practice, it is therefore only the flicker noise, which we assume originally resides in the baseband, that is efficiently removed. The white noise will still be present as fold-over components due to heavy aliasing. This is elaborately explained in [23].

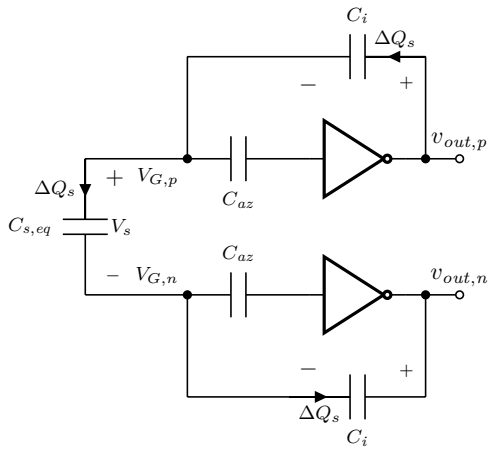


Figure 4.3: Charge transfer during ϕ_2.

4.1.2 Loop filter transfer function

Having established that the V_G nodes works as virtual grounds during ϕ_2, it is now possible to derive the transfer function of the loop filter. To do this, we first consider figure 4.1 and realize that every ϕ_2, the $C_{s,1}$ pair and one of the $C_{s,2}$ pairs will be connected to the V_C node, and charge transfer to the C_i pair will take place. When this happens and before any current flows, the $C_{s,1}$ pair will hold a residue sample from the previous conversion, and the $C_{s,2}$ pair will hold a residue sample from the conversion before that. Specifically, the charge on each of the four capacitors is

$$\begin{aligned} Q_{s1,p} &= C_{s,1}\left[v_{in}^{+}(n-1) + v_{in,CM}(n-1)\right] \\ Q_{s1,n} &= C_{s,1}\left[v_{in}^{-}(n-1) + v_{in,CM}(n-1)\right] \\ Q_{s2,p} &= C_{s,2}\left[v_{in}^{+}(n-2) + v_{in,CM}(n-2)\right] \\ Q_{s2,n} &= C_{s,2}\left[v_{in}^{-}(n-2) + v_{in,CM}(n-2)\right] \end{aligned} \tag{4.2}$$

where the input is split into differential and common mode components, and the capacitor negative plates are the ones connected to the virtual grounds. When these four capacitors are connected together through the V_C node, we get an equivalent capacitor $C_{s,eq}$ between the virtual grounds. This is shown in figure 4.3. The capacitance of $C_{s,eq}$ is

$$C_{s,eq} = \frac{C_{s,1} + C_{s,2}}{2} \tag{4.3}$$

and before any current flows through the virtual grounds, it will hold the voltage

$$V_{s,eq} = \frac{1}{C_{s,1} + C_{s,2}}\left[-C_{s,1}\,v_{in}(n-1) - C_{s,2}\,v_{in}(n-2)\right] \tag{4.4}$$

This relation is found by combining the charge on the parallel capacitors, and then adding the voltages in the resulting series combination. Also, $v_{in} = v_{in}^{+} - v_{in}^{-}$, which

means that all the common mode components have canceled and will not result in any charge transfer.

What now happens, is that the inverters will empty $C_{s,eq}$ to restore the virtual grounds, due to the negative feedback. The only path for the current to flow is through both C_i capacitors, and they will therefore receive a charge ΔQ_s according to figure 4.3. Since this charge empties $C_{s,eq}$, it is given by

$$\Delta Q_s = -C_{s,eq} V_{s,eq} \tag{4.5}$$

After this charge transfer is finished, the differential output of the filter is given by the difference between the voltages across the C_i capacitors. Due to ΔQ_s, this output has now received an update

$$\Delta v_{out}(n) = \frac{\Delta Q_s}{C_i} - \frac{-\Delta Q_s}{C_i} = \frac{C_{s,1}}{C_i} v_{in}(n-1) + \frac{C_{s,2}}{C_i} v_{in}(n-2) \tag{4.6}$$

Finally, we need to take the old output voltage into account, as the C_i capacitors are never reset. We can then write the output voltage during conversion n as

$$v_{out}(n) = v_{out}(n-1) + \Delta v_{out}(n) = v_{out}(n-1) + \frac{C_{s,1}}{C_i} v_{in}(n-1) + \frac{C_{s,2}}{C_i} v_{in}(n-2) \tag{4.7}$$

The transfer function is now found by taking the z-transform and rearranging. This yields

$$H(z) = \frac{V_{out}(z)}{V_{in}(z)} = \frac{\frac{C_{s,1}}{C_i} z^{-1} + \frac{C_{s,2}}{C_i} z^{-2}}{1 - z^{-1}} \tag{4.8}$$

By comparing to the high level transfer function given in equation 3.1, we see that these equations are equal, and that the coefficients are given as

$$\begin{aligned} b_1 &= \frac{C_{s,1}}{C_i} \\ b_2 &= \frac{C_{s,2}}{C_i} \end{aligned} \tag{4.9}$$

4.1.3 Common mode feedback

As seen in the previous subsection, the output voltage is regulated in a differential manner when the circuit operates. That is, the main feedback loops regulate the difference between the charges stored on the C_i capacitors, according to the input. As the output common mode voltage is not adjusted at all by this mechanism, an own common mode feedback circuit (CMFB) is needed to ensure a well defined common mode output voltage.

The common mode feedback circuit is shown inside the dashed box in figure 4.1, and it consists of four equally sized, permanently connected capacitors. This circuit

adjusts the common mode voltage towards the inverter reset voltage V_{off}. This voltage is close to mid-rail, and is a convenient common mode voltage for the comparator following the loop filter. Also, it keeps the output voltage changes between ϕ_1 and ϕ_2 to a minimum.

The common mode feedback circuit will now be analyzed. We do this for the positive input branch, and therefore consider the two C_{cm} capacitors connected to the $V_{G,p}$ node. During ϕ_1, these capacitors will hold charges proportional to the inverter reset voltages (which we assume assume are constant between clock cycles). That is

$$\begin{aligned} Q_{p,1} &= C_{cm} V_{off,p} \\ Q_{n,1} &= C_{cm} V_{off,n} \end{aligned} \tag{4.10}$$

where p and n denotes the positive and negative output nodes respectively. Then, after the output voltages have settled in ϕ_2, we have

$$\begin{aligned} Q_{p,2} &= C_{cm} v_{out,p}(n) \\ Q_{n,2} &= C_{cm} v_{out,n}(n) \end{aligned} \tag{4.11}$$

Since the charge has changed during ϕ_2, some charge ΔQ has been transferred from the common mode feedback circuit. This charge must flow through the virtual ground and into the C_i negative plate. By taking the difference between the charge stored on each of the two CMFB capacitors in ϕ_2 and ϕ_1 and adding the contributions, we find ΔQ as

$$\Delta Q = 2C_{cm} \left[\frac{v_{out,p}(n) + v_{out,n}(n)}{2} - \frac{V_{off,p} + V_{off,n}}{2} \right] = 2C_{cm} \left[v_{cm}(n) - V_{off,avg} \right] \tag{4.12}$$

We see that a charge proportional to the difference between the current output common mode and the average reset voltage enters the C_i plate. By inspection, we see that the exact same happens in the negative branch, and the change to the output voltage due to ΔQ is therefore only common mode.

We can now find the common mode voltage in clock cycle n by taking the old common mode voltage and ΔQ into consideration. We get

$$v_{cm}(n) = v_{cm}(n-1) + \frac{-\Delta Q}{C_i} = v_{cm}(n-1) - \frac{2C_{cm}}{C_i} v_{cm}(n) + \frac{2C_{cm}}{C_i} V_{off,avg} \tag{4.13}$$

From this, the common mode voltage can be expressed in the z-plane as

$$V_{cm}(z) = \frac{2C_{cm}}{C_i} \frac{1}{1 + \frac{2C_{cm}}{C_i} - z^{-1}} V_{off,avg} \tag{4.14}$$

This is a low pass filter having a DC gain of unity. Thus ideally, the common mode voltage is gradually adjusted to the average of the inverter reset voltages.

This common mode feedback circuit is conceptually the same as the one with eight switches used in [26], but when the target common mode voltage ($V_{cm,avg}$) and

the actual common mode voltage ($v_{cm}(n)$) happen to be collected from the same nodes, all the switches used there become redundant.

Since the size of the correction term (ΔQ term) added to v_{cm} every clock cycle is proportional to $\frac{2C_{cm}}{C_i}$, this ratio determines how fast the common mode feedback loop can correct errors that arise. Also, there are common mode errors that occur every clock cycle, for example due to charge injection. If the common mode capacitors are too small (i.e. the circuit too slow), they fail to correct such errors, and an offset between $V_{off,avg}$ and the actual common mode voltage arises. On the other hand, the capacitors should be as small as possible to not contribute extra load to the inverters. In the designed loop filter, it was found that dummy switches clocked at ϕ_2 connected to the virtual ground nodes counteracts a large part of the charge injection into C_i every clock cycle. The use of these dummys has therefore allowed small CMFB capacitors of $C_{cm} = 11\,\text{fF}$ to be used when still getting a low common mode offset (around 5 mV observed). This capacitor value yields a fairly slow common mode adjustment, but the observed common mode drift during operation has been found negligible (around 2 mV observed).

4.1.4 Chopped input buffers

If the SAR residue is sampled onto the C_s capacitors passively, attenuation occurs due to charge sharing between the DAC capacitance and C_s. This attenuation is equivalent to the incomplete linear settling situation discussed in section 3.5 (i.e. a factor $\beta < 1$ occurs in the transfer function), and similar performance degradation can therefore be expected. Passive sampling is the used solution in [11], and their DAC capacitance therefore have to be much larger than the sampling capacitance to keep the attenuation insignificant. The sampling capacitance is in turn sized according to kT/C requirements, which means that the DAC needs to be larger than its own kT/C requirement dictates. This might be a good solution when the DAC size is limited by mismatch requirements long before it hits kT/C requirements anyway, but at high target accuracy and in heavily scaled technologies, like in this thesis, this is not the case.

To mitigate this problem, input buffers are employed in the designed loop filter, and there is therefore no need to size the DAC according to C_s. These buffers are only needed during ϕ_1, and are therefore shut down during ϕ_2 to save power. The buffer design is an improved source follower circuit (presented in section 4.3), and also steps down the common mode voltage by a V_{gs} between its input and output. This is an advantage since it allows almost all the switches in the loop filter to be implemented as single NMOS switches rather than transmission gates due to signal levels close to the ground potential.

A drawback of using active buffers is that they contribute extra noise to the circuit. In the used buffers, the noise is dominated by flicker, and this have to be handled to achieve usable performance. This problem has been solved by chopping the buffers

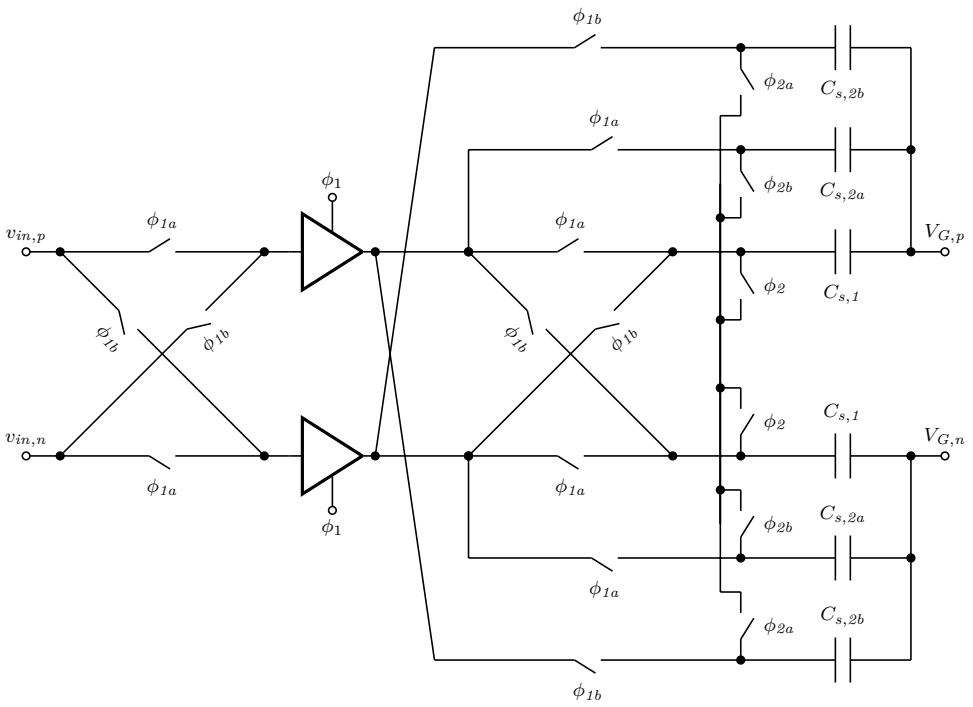


Figure 4.4: Chopping implementation for the buffers.

using a chopping frequency $f_{chop} = f_s/2$. According to the theory presented in section 2.6, this modulates the flicker noise (and also the offset) up to $f_s/2$, which is outside the signal band. In simulations, this has been found to reduce the flicker noise corner frequency to about 50 Hz, with the result that the flicker contribution is insignificant compared to the white noise.

To implement the chopping, switches are needed both at the input and the output of the buffers. In the loop filter, the consequence of this is that the sampling switches that was shown in figure 4.1 become redundant, and can instead be integrated into the chopping system. This leads to the chopping implementation shown in figure 4.4. Here, the "left part" of the loop filter is shown with the switching scheme actually used for chopping and sampling. The first thing to note, is that no new clocks are needed, and the chopping implementation is therefore done particularly easy, with a minimum of new circuitry. Also, due to the common mode voltage step-down in the buffers, only the switches in front of them need to be implemented with transmission gates.

At the input of the circuit, the ϕ_{1a} and ϕ_{1b} clocks are used to flip the inputs for every other sample. This is equivalent to multiplying the input by a square wave alternating between +1 and -1, and modulates the signal as discussed in

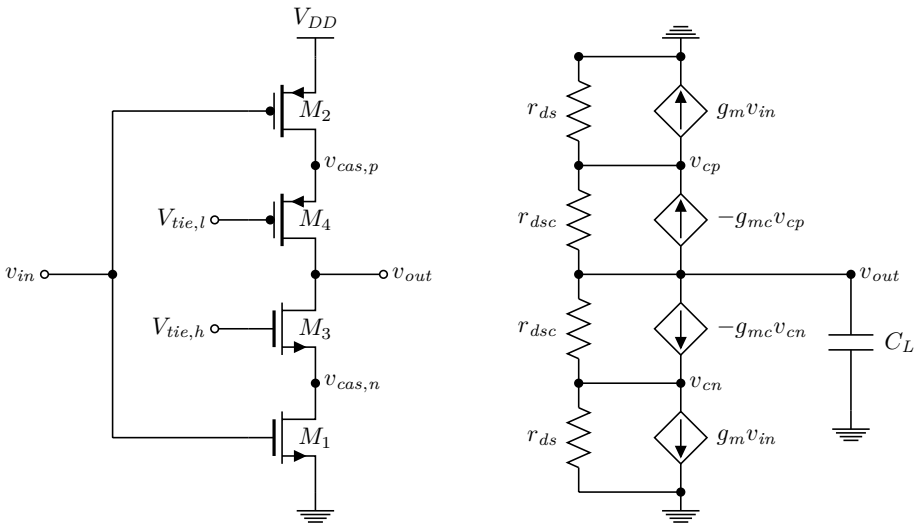
section 2.6. Likewise, the same is done at the input of the $C_{s,1}$ pair, which thus receives a sample of the demodulated input signal in addition to modulated noise and offset every clock cycle. When it comes the $C_{s,2}$ capacitors, each pair is only used for sampling every other cycle due to the interleaving. This leads to a situation where the same buffers always drive the same $C_{s,2}$ capacitors, as seen in the schematic. At first glance, it is easy to think that this destroys the chopping, since each $C_{s,2}$ capacitor is always connected to the same noise source (i.e. same buffer). However, charge is only transferred from each capacitor pair every second clock cycle due to the interleaving. Seen from the virtual grounds, this interleaving makes both $C_{s,2}$ pairs look like a single capacitor pair delivering charge every single clock cycle. Realizing this, it follows that this equivalent pair will be connected to flipped buffers every second clock cycle (every time the $C_{s,2b}$ pair is used), and the chopping thus works.

4.2 Inverter

The inverter used as amplifier in the loop filter is shown in figure 4.5a. Instead of a standard inverter topology using two transistors, a cascoded topology is chosen for this design. The gates of the cascode NMOS and PMOS are biased to V_{DD} and ground respectively, and this is done through tie cells due to ESD considerations. As a reference, transistor dimensions as well as some important simulation results are given in table 4.2.

As soon will be shown, the cascoding increases the DC gain of the inverter. As discussed in section 3.2, however, it is not necessary to have more than around 40 dB DC gain in the entire loop filter, so high gain is not the motivation for cascoding in this design. The more important advantage of cascoding in this work is instead that it decreases the parasitic capacitance at the amplifier input. As will be discussed in section 4.4, this parasitic capacitance impacts the noise performance of the filter when it becomes significant compared to the auto-zero capacitance, and it thus have to be minimized.

The parasitic capacitance to AC ground seen at the inverter input can become quite large due to the Miller effect, explained in detail in for example [27]. Told shortly, what happens is that when the input changes, the drains of $M_{1,2}$ change in the opposite direction, and according to the gain from the input to the nodes $v_{cas,p}$ and $v_{cas,n}$. The C_{gd} parasitic capacitances of these devices therefore get their voltage changed from both sides, in opposite directions, and at one of the sides proportional to the gain. The result of this is that much more current have to flow from the input and into these capacitances than the input voltage change alone should indicate, and the equivalent capacitance seen is therefore much larger than the physical C_{gd} capacitances. If a normal inverter was used, the Miller effect would have occurred between the input and output node, and therefore been proportional to the total inverter gain. Now, it is only the gain into the cascode nodes that count, and this



(a) Circuit schematic. **(b)** Small signal model.

Figure 4.5: Inverter schematics.

can be kept lower.

4.2.1 Small signal analysis

To discuss how the inverter should be sized to meet a given specification with minimum parasitic capacitance, equations for DC gain to the output and cascode nodes, in addition to a unity gain frequency expression is needed. This can be derived from the small signal model in figure 4.5b, where body effect and device capacitances have been neglected. We also assume that the main devices and the cascode devices respectively have the same transconductance and channel length modulation resistance. Finally, we realize that $v_{cp} = v_{cn}$ due to the symmetry of the small signal circuit. This will be used in the following, and this voltage will just be denoted as v_c.

To find the DC gains, we ignore the load capacitance and write KCL equations in the output node, and in one of the cascode nodes. After some simplification, this yields

$$\begin{aligned} \frac{v_{out} - v_c}{r_{dsc}} &= g_{mc} v_c \\ \frac{v_{out} - v_c}{r_{dsc}} - \frac{v_c}{r_{ds}} &= g_m v_{in} + g_{mc} v_c \end{aligned} \tag{4.15}$$

When solving this equation set simultaneously for v_{out} and v_c, we obtain the DC

Table 4.2: Inverter specifications.

Post-layout simulation results:	
Unity gain freq.	225 MHz @ $C_L = 277$ fF
DC gain	60.5 dB
DC gain to cascode nodes	23 dB
Steady state I_{DD}	8 µA
Reset voltage V_{off}	400.7 mV

Transistor dimensions:			
Device	Width	Length	Fingers
M_1	7 µm	60 nm	4
M_2	21 µm	40 nm	12
M_3	0.98 µm	400 nm	2
M_4	2.94 µm	400 nm	2

gain to the output as

$$\frac{v_{out}}{v_{in}} = -(g_m r_{ds} \cdot g_{mc} r_{dsc} + g_m r_{ds}) \approx -g_m r_{ds} \cdot g_{mc} r_{dsc} \tag{4.16}$$

For the gain to the cascode nodes, we get

$$\frac{v_c}{v_{in}} = -\frac{g_m r_{ds} \cdot g_{mc} r_{dsc} + g_m r_{ds}}{1 + g_{mc} r_{dsc}} \approx -g_m r_{ds} \tag{4.17}$$

An expression for the unity gain frequency can be obtained by ignoring all the channel length modulation resistances and taking the load capacitance into account. KCL at the output and cascode nodes then gives the equations

$$\begin{aligned} sC_L v_{out} &= 2g_{mc} v_c \\ g_{mc} v_c &= -g_m v_{in} \end{aligned} \tag{4.18}$$

From this, we find the transfer function valid around the unity gain frequency as

$$\frac{v_{out}}{v_{in}}(s) = -\frac{2g_m}{sC_L} \tag{4.19}$$

Then, by substituting $s = j2\pi f$, taking the magnitude, and evaluating at unity, we find the unity gain frequency as

$$f_t = \frac{2g_m}{2\pi C_L} \tag{4.20}$$

By comparing this equation to the more general unity gain equation (2.23), we see that the transconductance of the amplifier stage is twice the main device transconductance. This is because both M_1 and M_2 work as transconductive devices.

4.2.2 Sizing considerations

These equations derived in the previous section give good insight into how the inverter should be sized. First and foremost, we see from equation (4.20) that the transconductance of the main devices has to be set according to the load capacitance seen at the inverter output, and the needed unity gain frequency. This means that $M_{1,2}$ should have high g_m/I_D to give an energy efficient design. From section 2.3, we know that g_m/I_D almost only varies with the inversion level, which can be changed by adjusting V_{gs} or the threshold voltage V_{th}. In the inverter, the first mentioned is not easy to adjust if V_{DD} is not changed. Therefore, V_{th} has been increased by using regular threshold voltage devices (RVT devices) rather than low threshold voltage devices. Additionally, V_{th} increases with the device length in the used technology, and this can also be exploited. If too high g_m/I_D is chosen, the g_m achieved per W/L can get very low, with the result that the transistor widths must be increased excessively to get the needed transconductance. This results in large area and large parasitic capacitances, so a good balance have to be found. In the designed inverter, the main devices have g_m/I_D values of 29 and 28, and this was achieved by increasing the transistor lengths slightly up from minimum. This results in reported threshold voltages of 434 mV and 424 mV, and it is therefore reasonable to say that the transistors operate in moderate inversion when the input operating point is around 400 mV.

When it comes to the gain into the cascode nodes, we see from equation (4.17) that it is only controlled by the transconductance and r_{ds} of the main devices. As the first mentioned is locked by the needed unity gain and bandwidth, it is important to keep r_{ds} quite small by not increasing the length of the main transistors more than needed. This is in contrast to a normal inverter, where it is probable that the r_{ds} of these devices have to be increased actively to get acceptable gain in the whole inverter.

As pointed out earlier, the cascode devices are biased by the ground and supply rails. Due to this, the overdrive of these transistors can easily get pretty large, resulting in problems of getting them into saturation. To mitigate this problem, the threshold voltages of these devices are increased quite a lot by selecting RVT devices and by using long transistor lengths. This also increases both g_m/I_D and r_{dsc}, and from equation (4.16), we see that both these things contribute to higher DC gain in the inverter, which is an advantage.

Since the inverter is connected directly to the supply rail and thus biases itself, all the W/L ratios have to be adjusted until the wanted transconductance is achieved, and all devices have sensible V_{ds} voltages. In the design, the reset voltage V_{off} has also been adjusted to $V_{DD}/2$, and this is the reason for the slightly different lengths of M_1 and M_2.

The closed loop bandwidth of the inverter is generally given by f_t/β, where β is the feedback factor the inverter output sees when connected in the loop filter. In

practice, the bandwidth has been regulated by observing the time domain performance in transient simulations of the whole NS-SAR. For the chosen design point, time constants of around 2.6 ns are observed during ϕ_2, and this gives slightly more than three time constants of settling before the SAR binary search starts. This is sufficient according to the considerations in section 3.5. When simulated in open loop width a load capacitance of 277 fF, which corresponds to what the inverter sees in the loop filter, a unity gain frequency $f_t = 225\,\text{MHz}$ is achieved, as reported in table 4.2. The DC gain of 60.5 dB also reported in this table is not due to a definitive requirement, but followed when the circuit was otherwise sensibly sized.

The input referred white noise spectrum of a standard inverter is according to [26]

$$V_{n,inv}^2 = \frac{2kT\gamma}{g_m} \tag{4.21}$$

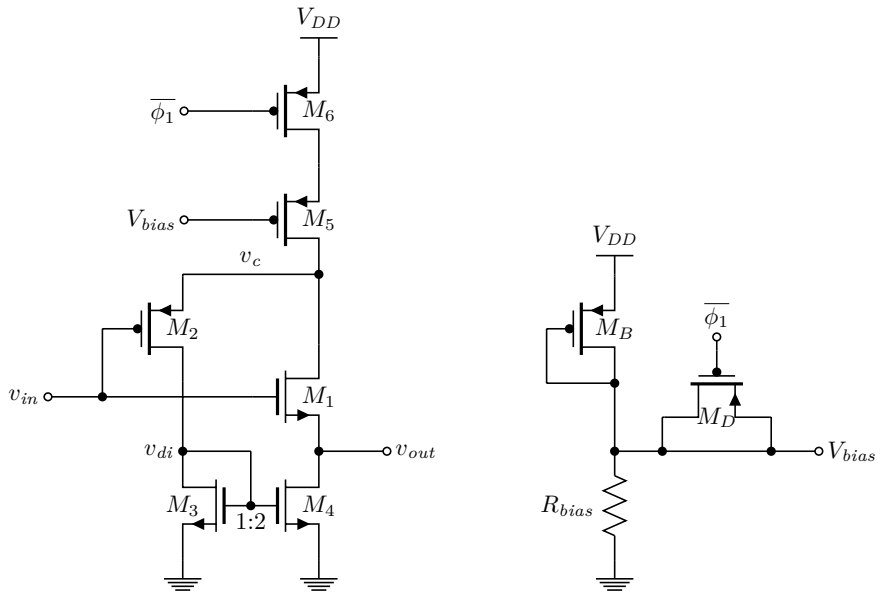
with $\gamma = 2/3$ in strong inversion and $\gamma = 1/2$ in weak inversion (it is possible that these γ values are technology dependent). Furthermore, cascode devices do not contribute significantly to the noise as long as r_{ds} is reasonably high [28, p. 233]. Therefore, equation (4.21) should be a good approximation also for cascoded inverters.

4.3 Buffer

The input buffer is shown in figure 4.6a, and device dimensions and important simulation results are found in table 4.3. The bias circuit shown in figure 4.6b is common for both buffers in the loop filter. Also, all the transistors are LVT devices.

The buffers are only needed during the ϕ_1 phase, and are therefore otherwise shut down to save power. This is done through the use of transistor M_6, which acts as a switch. Related to this is also the MOSCAP M_D in the bias circuit, which counteracts charge kicks coming through M_5 when the buffer is turned on. This is needed since the bias circuit has a low bandwidth to save power, and will thus use too long time to settle after the kicks without this transistor.

The buffer topology is found in [29], and is based around a source follower constituted by M_1 and its load M_4. Additionally, an extra source follower is made up by M_2, with load/bias transistor M_5. The current through this source follower is further used to make a feedback loop which controls the load M_4 of the first follower. The result of these modifications is lower attenuation (i.e. gain closer to unity) and less distortion [29]. The first reason why this is achieved is because the extra source follower M_2 makes the v_c node track the input voltage, such that V_{ds} of M_1 is kept nearly constant. Due to this, the finite and non-linear output resistance (i.e. r_{ds} resistance) is nearly canceled, with less attenuation and distortion as result. Secondly, the M_3 transistor in the feedback loop will "measure" current changes through M_1 since the bias current through M_5 is nearly constant. This



(a) Buffer circuit. **(b)** Bias circuit. Common for both buffers.

Figure 4.6: Buffer schematics.

measurement is fed to the M_4 gate through the voltage v_{di}, with the result that the current change through M_1 is counteracted. The output resistance of M_4 will therefore also appear to be larger, with decreased attenuation and distortion as result. All this is explained in [29].

Since it is only residue voltages that propagate through the buffers in the loop filter, linearity is not the biggest challenge due to the low swing. However, signal attenuation affects the loop filter transfer function in the same way as linear incomplete settling (discussed in section 3.5), and should thus be minimized. It is therefore advantageous to use the chosen buffer instead of a standard source follower, for instance. Additionally, it will be revealed in the following that the feedback loop also increases the bandwidth of the buffer.

The bandwidth and the noise power of the buffer will now be derived since these things are not given in [29]. After this, the sizing of the buffer will be discussed in light of these analyses.

4.3.1 Bandwidth

The small signal model in figure 4.7 is used to find the bandwidth of the buffer. In this model, channel length modulation resistances and device capacitances have

Table 4.3: Buffer parameters

Post-layout simulation measurements	
Bandwidth	74.25 MHz @ $C_L = 1$ pF
DC gain	-0.8 dB
Steady state I_{DD}	8.15 µA

Transistor dimensions			
Device	Width	Length	Fingers
M_1	21.6 µm	47 nm	8
M_2	6.48 µm	30 nm	4
M_3	0.36 µm	250 nm	2
M_4	0.72 µm	250 nm	2
M_5	0.822 µm	160 nm	2
M_6	4.5 µm	30 nm	2
M_B	0.18 µm	600 nm	1
M_D	2 µm	600 nm	1
R_{bias}	500 kΩ		6

been neglected. Also note that the current source due to bias transistor M_5 disappears from the model since it never conducts small signal current.

By using KCL in all the nodes, we obtain the equation set

$$\begin{aligned} g_{m4}v_{di} + sC_L v_{out} &= g_{m1}(v_{in} - v_{out}) \\ g_{m3}v_{di} &= -g_{m2}(v_{in} - v_c) \\ g_{m1}(v_{in} - v_{out}) &= g_{m2}(v_{in} - v_c) \end{aligned} \tag{4.22}$$

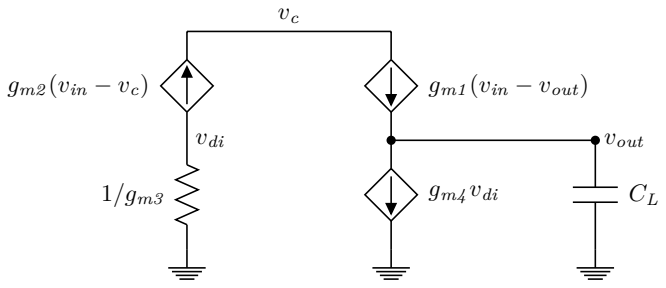


Figure 4.7: Buffer small signal model used to find the bandwidth.

This can be solved with respect to the transfer function, and this yields

$$\frac{v_{out}}{v_{in}}(s) = \frac{g_{m1}\left(1+\frac{g_{m4}}{g_{m3}}\right)}{g_{m1}\left(1+\frac{g_{m4}}{g_{m3}}\right)+C_L s} = \frac{1}{1+\tau s} \tag{4.23}$$

where the transfer function has been equated to the transfer function of a general all-pole, first order low-pass filter with unity gain. By comparison, we find the time constant τ to be

$$\tau = \frac{C_L}{g_{m1}\left(1+\frac{g_{m4}}{g_{m3}}\right)} \tag{4.24}$$

Finally, we use general knowledge about such a low-pass filter, and find the $-3\,\mathrm{dB}$ bandwidth through the relation

$$f_c = \frac{1}{2\pi\tau} = \frac{g_{m1}\left(1+\frac{g_{m4}}{g_{m3}}\right)}{2\pi C_L} \tag{4.25}$$

We see that in addition to g_{m1}, the bandwidth is also proportional to a factor $\left(1+\frac{g_{m4}}{g_{m3}}\right)$ due to the feedback loop. In comparison, the bandwidth of a normal source follower under the same assumptions (i.e. load capacitance dominates) is $\frac{g_{m1}}{2\pi C_L}$ [21]. The bandwidth of the used buffer is thus boosted by the feedback loop, and with the 1:2 ratio between M_3 and M_4 used in this design, this boosting factor equals 3. We thus get triple bandwidth when using 1.5 times more current than a normal source follower (the current in both branches add up to 1.5). This is equivalent to say that a given bandwidth can be realized by using half the current as in a normal source follower.

4.3.2 Noise

The small signal model in figure 4.8 is used for the noise analysis of the buffer. In this model, all capacitances are neglected, which means that it is the low frequency value of the noise spectrum that is found. Furthermore, channel length modulation is still neglected, and v_{in} is set to zero. The white noise of transistor M_{1-5} is modeled by the use of independent current sources where $I_n^2 = 4kT\gamma g_m$, such as in section 2.5.2 (the direction of the current sources was chosen arbitrarily). Flicker noise is not taken into consideration since it is handled well by the chopping, and the noise of the switch transistor M_6 is neither taken into consideration.

In the small signal model, super position can be used to find the transfer function from each of the noise current sources to v_{out}. This yields, when not using squared

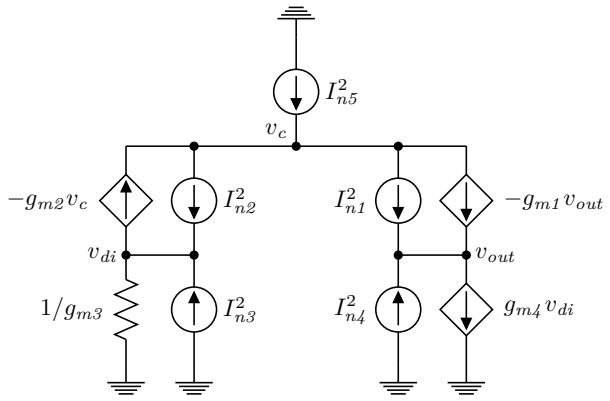


Figure 4.8: Buffer small signal model used for noise analysis.

currents yet

$$\begin{aligned}
\frac{v_{out}}{I_{n1}} &= \frac{1}{g_{m1}} \\
\frac{v_{out}}{I_{n2}} &= 0 \\
\frac{v_{out}}{I_{n3}} &= -\frac{1}{g_{m1}\left(1+\frac{g_{m3}}{g_{m4}}\right)} \\
\frac{v_{out}}{I_{n4}} &= \frac{1}{g_{m1}\left(1+\frac{g_{m4}}{g_{m3}}\right)} \\
\frac{v_{out}}{I_{n5}} &= -\frac{1}{g_{m1}\left(1+\frac{g_{m3}}{g_{m4}}\right)}
\end{aligned} \tag{4.26}$$

Using these transfer functions, we can compute the value of the output noise spectral density before it starts to roll off due to the buffer bandwidth. Moreover, this value will correspond to the unfiltered input referred noise spectral density if we assume that the gain is unity at low frequencies. Using this assumption we find the input referred noise spectral density as

$$\begin{aligned}
V_{ni}^2 &= I_{n1}^2\left|\frac{v_{out}}{I_{n1}}\right|^2 + I_{n3}^2\left|\frac{v_{out}}{I_{n3}}\right|^2 + I_{n4}^2\left|\frac{v_{out}}{I_{n4}}\right|^2 + I_{n5}^2\left|\frac{v_{out}}{I_{n5}}\right|^2 \\
&= \frac{4kT\gamma}{g_{m1}}\left[1 + \frac{g_{m3}}{g_{m1}\left(1+\frac{g_{m3}}{g_{m4}}\right)^2} + \frac{g_{m4}}{g_{m1}\left(1+\frac{g_{m4}}{g_{m3}}\right)^2} + \frac{g_{m5}}{g_{m1}\left(1+\frac{g_{m3}}{g_{m4}}\right)^2}\right]
\end{aligned} \tag{4.27}$$

Now we can find the noise power at the output by multiplying with the squared frequency response of the buffer transfer function and integrating. We assume DC gain of unity and first order roll-off, and can thus use equation (4.23) as the transfer

function. This, together with the relation in equation (4.25) yields

$$P_{no} = \int_0^\infty \frac{1}{1+\left(\frac{f}{f_c}\right)^2} V_{ni}^2 \, df = \frac{\pi}{2} f_c V_{ni}^2$$

$$= \frac{\gamma kT}{C_L} \left[\underbrace{\left(1+\frac{g_{m4}}{g_{m3}}\right)}_{\text{Due to } M_1} + \underbrace{\frac{g_{m3}+g_{m4}}{g_{m1}\left(1+\frac{g_{m3}}{g_{m4}}\right)^2}}_{\text{Due to } M_3} + \underbrace{\frac{g_{m3}}{1+g_{m1}\frac{g_{m3}}{g_{m4}}}}_{\text{Due to } M_4} + \underbrace{\frac{g_{m5}\left(1+\frac{g_{m4}}{g_{m3}}\right)}{g_{m1}\left(1+\frac{g_{m3}}{g_{m4}}\right)^2}}_{\text{Due to } M_5} \right] \tag{4.28}$$

This is also the same as the input referred noise power under the assumption of DC gain equal to unity.

4.3.3 Sizing considerations

By inspection of equation (4.28), it is possible to see that the noise gets minimized when g_{m1} is high, g_{m3}, g_{m4} and g_{m5} are low, and the $\frac{g_{m4}}{g_{m3}}$ ratio (i.e. the ratio between M_4 and M_3) is low. Equation (4.25) on the other hand, suggests that the $\frac{g_{m4}}{g_{m3}}$ should be taken high to boost the bandwidth. There thus exists a trade-off between bandwidth per bias current and noise performance in this circuit. However, the impact on the noise performance is larger than seen by first glance, because if large ratios are chosen, g_{m1} will probably be taken lower than before to keep the bandwidth constant. This implies that if the bandwidth is kept, g_{m1} will go down when $\frac{g_{m4}}{g_{m3}}$ goes up in equation (4.28), and the noise performance is thus worsened quickly. Due to this situation, the ratio between M_4 and M_3 is kept at 2:1, which is the ratio used in [29].

Having this established, we see from equation (4.25) that the bandwidth have to be regulated by g_{m1} for a given load capacitance. Because of this, a large g_m/I_D should be chosen for M_1 such that the bias current can be minimized. This can be accomplished by using a large W/L ratio, and can additionally be influenced by changing V_{th} through L adjustments. Large W/L ratios will also keep M_4 in saturation for a larger signal swing due to lower V_{gs1}. As for the inverter, too high g_m/I_D will result in excessively large area and more parasitic capacitance.

g_m/I_D must be taken low for both M_3, M_4 and M_5 to minimize the noise. For M_3 and M_4, this can be done by using small W/L ratios. For M_5, V_{bias} have to be low. This voltage is generated by the bias circuit in figure 4.6b, and here M_B will need a low W/L ratio to yield a low V_{bias}. R_{bias} is in practice tuned until a satisfying voltage and bias current through the resistor is obtained. M_5 is then sized such that the wanted bias current in the buffer is achieved.

The sizing of all the transistors affect the swing of the buffer in some manner.

When M_3, M_4 and M_5 are sized for low g_m/I_D , their overdrive is increased, with the consequence that higher V_{ds} voltages are needed to put them in saturation. This effect sets the lower bound for g_m/I_D for these transistors. Since the signal swing is quite low, much swing can be sacrificed, but it is nevertheless challenging to achieve very low g_m/I_D values due to the low supply voltage. Transistor M_2 neither affects the noise or the bandwidth notably, but it should have a fairly high g_m/I_D ratio such that its V_{gs} is not too high. Otherwise, high v_c values will put M_5 quickly into triode.

4.4 Noise

As pointed out in section 3.3, the noise contribution of the loop filter can be modeled as a single, input referred noise source. Although no analytical solution for the power of this noise source has been derived in this work, it is possible to discuss the overall noise performance of the loop filter qualitatively. Consider the main circuit schematic in figure 4.1. During ϕ_1, noise from the sampling/chopping switches and the buffers will be present over the sampling capacitors and stored there at the falling clock edge. As discussed in section 2.5.3, this sampling will lead to aliasing, with the result that the continuous time noise spectral density is compressed into the baseband. During ϕ_2, this stored noise is transferred to C_i likewise as the input. Also, new noise will arise over C_i due to the inverters and the switches that conducts during ϕ_2, and this contribution is also stored at C_i at the end of ϕ_2. As these things happen every clock cycle, and the resulting noise is integrated over C_i likewise as the input, the described noise contribution mechanisms can be viewed as input referred. Correspondingly, the output referred noise will be the integrated version of the noise sampled every clock cycle.

From the paragraph above, and from what is written about the loop filter and about noise so far in this thesis, it should be understandable that the input referred noise is inversely proportional to C_s and C_i, and this is also confirmed by simulations. In practice, the loop filter can therefore be sized to meet noise requirements by ensuring that the buffers and inverters are designed as noise efficient as possible, and then pick large enough capacitors (the amplifiers may need to be resized for the chosen capacitance though). In practice, this has been carried out by using *pnoise* simulations from the Spectre RF simulator. These simulations complete reasonably quickly, and it is therefore possible to explore the design space in terms of noise performance.

Another important thing to point out, is that the size of C_{az} compared to the size of the parasitic capacitance C_{pi} of the inverter input can seriously degrade the noise performance, and this is not easy to see without conducting a full noise analysis of the circuit. An analysis of an almost similar integrator structure is carried out in [30], with the input referred noise of the OTA (or inverters in this thesis) as the only noise contributor. It is found that the noise performance degrades

significantly when the $\frac{C_{az}}{C_{pi}}$ is decreased, and that C_{az} has to be chosen much larger than C_{ci} to approach the noise performance of an ideal auto-zeroed integrator. The performance degradation also affect the ability of the auto-zeroing system to remove flicker noise.

Similar effects as in [30] have also been found in the loop filter during simulations, and large C_{az} capacitors are therefore needed. In practice, the value of C_{az} has been swept in *pnoise* simulations to find a value that makes the noise performance impact insignificant. As seen in table 4.1, this has resulted in a large $C_{\text{AZ}} = 852\,\text{fF}$, and this is because C_{pi} is over 100 fF, even with cascoded inverter. Large C_{AZ} values will nevertheless not lead to significantly increased power consumption, since the voltage across these capacitors are nearly constant. The main impact is therefore only increased area.

4.5 Non-overlapping clock generator

The non-overlapping clock generator designed for the loop filter is shown in figure 4.2. This circuit takes in the external clock signal SMP as well as the DONE signal from the SAR, and derives all the other clocks that was shown in the timing diagram (figure 4.2) from these signals. The logic standard cells used in the design are from [20].

In the upper part of the schematic, two inverter delay chains with feedback are used to make ϕ_1 and ϕ_2 non-overlapping. This works because the input signals to these chains (derived from DONE and SMP, and supplied to the two NOR-gates) can turn ϕ_1 and ϕ_2 off on their own, since a NOR-gate yields a zero when one of their inputs are high. On the other hand, both inputs of one of the NOR-gates have to be low to turn either ϕ_1 or ϕ_2 on, and due to the feedback, this can not happen to the one of them if the other is high. The advanced clocks ϕ_{1ad} and ϕ_{2ad} are generated by some extra NOR-ports that taps the same delay chain, such that the signals turn on with the normal clocks, but off earlier.

The circuitry that generates the a- and b-clocks are shown in the bottom of the schematic. Here, a D-flip-flop are clocked by a signal in the delay-chain, such that it changes state every clock cycle. Both the non-inverting and inverting outputs of this flip-flop are fed to NOR-gates together with the (inverted) ϕ_1 and ϕ_2 signals, and connected such that the flip-flop-states always masks two of the outputs. To prevent spikes, the flip-flop have to change states during the non-overlapping interval, and its clock signal is therefore taken from a point in the delay chain that facilitates this. This solution has been verified to be spike-free across corners.

Also, the circuitry that gates the SAR sample clock is shown in the lower left of the schematic. Here, the sample clock is fed through a NOR-gate also connected to ϕ_1, and this ensures that the SAR will never get a high clock when ϕ_1 is high.

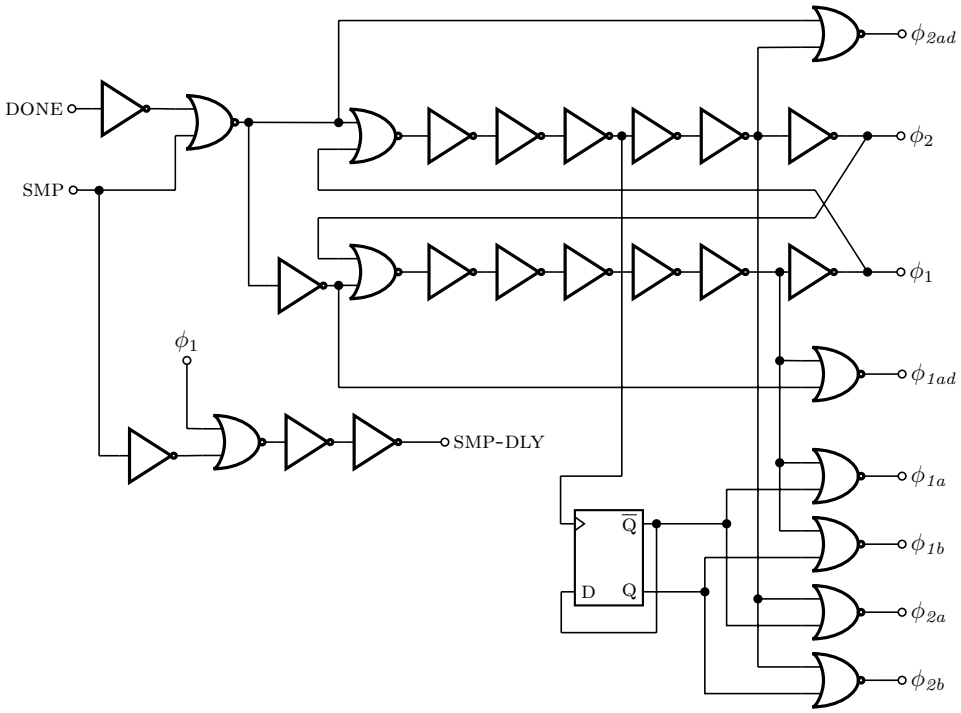


Figure 4.9: Non-overlapping clock generator circuit schematic.

In addition to the circuitry shown in figure 4.9, some inverters that generate inverted clock signals to PMOS transistors, and some gates that derive an enable signal to the SAR are also a part of the clock generator.

4.6 Layout

An annotated picture of the NS-SAR layout is shown in figure 4.10 (also, some more magnified pictures of different sections of the layout can be found in appendix B). Everything inside the block labeled SAR is from [20], but all the rest has been carried out in this work. The whole layout measures 133.5 µm × 119 µm (when regarded as a rectangular cell), and this corresponds to an area of approximately 0.016 mm^2.

In general, the loop filter is laid out symmetrically. Parasitic effects introduced by the layout will thus to a big extent appear as common mode errors. It was chosen to put the SAR and loop filter next to each other, in a way that facilitates a reasonably short routing path between the DAC plates and the buffer inputs. This connection is in general susceptible to interference and parasitic capacitance, since a small

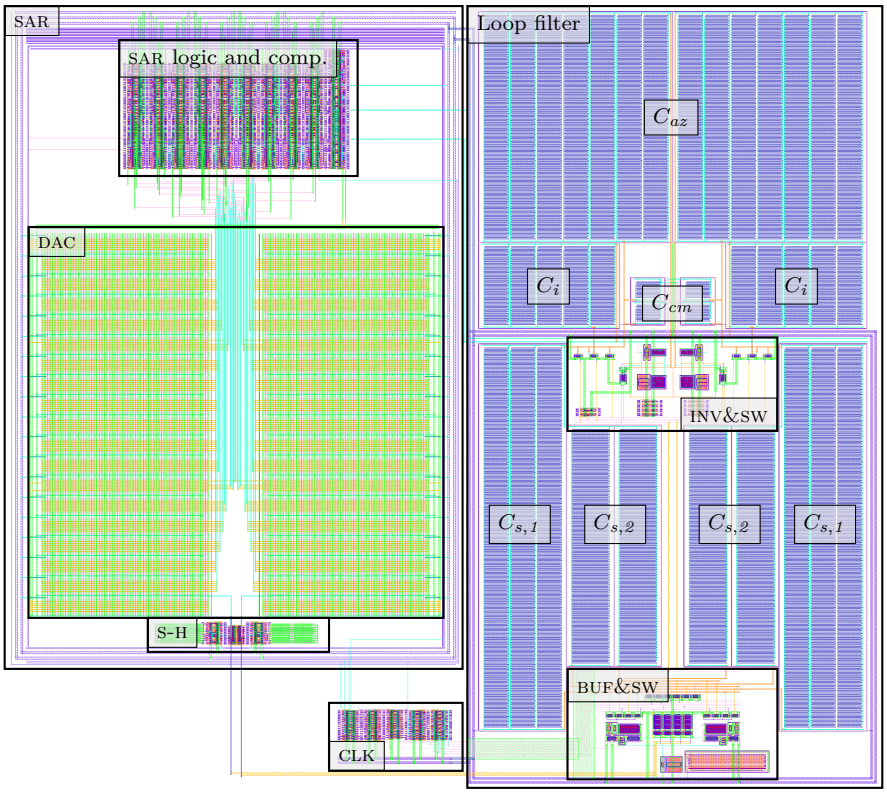


Figure 4.10: The layout of the entire NS-SAR

residue voltage have to be extracted from a high impedance capacitor array, with an accuracy of the same order as the whole ADC accuracy. The connection can be seen exiting the SAR through the S-H switch and progressing towards the BUF&SW block in the loop filter, which contains the buffers and everything else that was shown in the schematic in figure 4.4. Since most of the clocks are also used in this block, there is a lot of clock routing nearby, and the clock generator (CLK) is also located in the vicinity. Care has therefore been taken to separate input and clock lines as much as possible, and ground shields have also been utilized at critical points.

All the capacitors used in the circuit are of the MOMCAP type (metal-on-metal capacitors), and are generated by a script system from [20]. As an example, one of the C_{CM} capacitors of 11 fF are shown in figure 4.11. This capacitor is made of three unit MOMCAPs, placed after each other in the y-direction (you can not actually see that it is three unit capacitors due to overlapping). Similarly, an arbitrary number of repetitions of the unit capacitor in the y-direction as well as the x-direction can

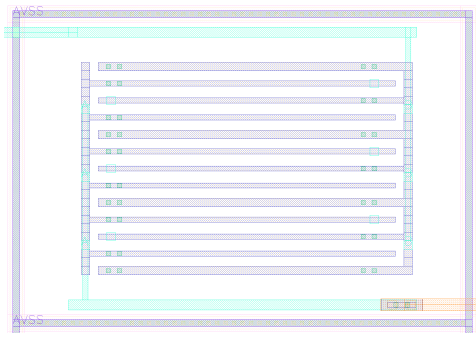


Figure 4.11: The layout of one of the common mode feedback capacitors.

be specified in the script, such that capacitors having different physical size and capacitance can be made. Also, the number of metal layers to use can be specified. In the capacitors used in the loop filter, all the thin metal layers in the technology have been used, i.e. M1 to M6. The exception is for C_{az}, where M1 has been left out to decrease the parasitic capacitance to ground. C_{az} has thus a bit less capacitance per area. Guard rings has also been generated around the capacitors to provide a more silent substrate underneath them.

The transistors in the loop filter have initially been generated from the schematic as parametric cells. After this, minor modifications have usually been carried out to make the transistors fully compliant with the *recommended* DRC (design rule check) rule set, and not just the *standard* one. Multiple fingers have been used for all transistors where it is possible (i.e. for almost all), and the number of fingers has mainly been chosen from geometrical considerations. All transistors have their own bulk connections in the form of full or partial guard rings. Dummy poly has also been used for all transistors to improve the regularity of the environment for each device. In the clock generator, standard cells from [20] have been used. This has made it possible to do a highly regular and compact layout for this block.

CHAPTER 5

Results

This chapter contains post-layout simulation results of the entire NS-SAR. Considerations directly regarding the results will be discussed when they are presented in this chapter, whereas more general and elaborate discussions of the work carried out in this thesis are located in chapter 6. After describing the testing methodology, this chapter will start by presenting sweeps conducted to find the input amplitude and the sample rate which yield best ENOB. After this, results from nominal simulation runs at different process corners will be presented, before mismatch performance will be considered in the form of Monte Carlo simulations. Then, a breakdown of the power consumption of the ADC will end the chapter.

5.1 Testing methodology

All the results presented in this chapter originate from transient simulations with noise enabled (unless stated otherwise), performed by the Cadence Spectre simulator with the *conservative* accuracy preset chosen. The maximum noise frequency is 10 GHz. The design is represented in the simulations as a single layout parasitic extracted (LPE) view, generated by Cadence QRC. Due to long simulation times, a LPE view generated with only parasitic capacitance extraction enabled has been used. Additionally, one single nominal simulation run with both resistance and capacitance extraction has been performed, and no significant differences in the

results where found, compared to simulations with only parasitic capacitance.

In the test-bench used for simulations, a differential sinusoid with common mode of $V_{DD}/2$ is applied to the ADC input when the system has reached steady state after power-up, and this results in an output signal in the form of a digital, parallel bit stream. This bit stream is fed into an ideal DAC implemented in Verilog-A, and which makes an analog version of the digitized signal. This results in a "staircase" waveform, which is suitable for post-processing by FFT. Other important stimuli from the test-bench are the main sample clock, and the supplies V_{DD} and V_{REF}. All these signals are generated using ideal sources in the performed simulations, and no resistance is inserted in series with the supplies. The actual test-bench schematics are included in appendix C.

The main quantity to extract from ADC simulations or measurements is accuracy, expressed either as ENOB, or equivalently as SNDR. This is most easily done in the frequency domain, and an FFT therefore needs to be performed on the staircase signal generated by the ideal DAC. This is not straight-forward because the time domain sequence is of finite length, while we actually want information about the infinitely long version of the signal (i.e. the steady state sinusoid coming from the ADC). Explained shortly, the effect of the finite data length is that the frequency spectrum corresponding to the infinite data is convoluted by a sinc-signal, such that the frequency components are smeared together. However, if a couple of precautions are taken, the spectrum corresponding to the infinite signal can be estimated with good quality from a finite data record. Such techniques, called *spectrum estimation*, are a fairly comprehensive field, and can not be covered in detail here. Instead, it can be referred to [14, appendix 2], which covers spectrum estimation for noise-shaped ADCs in detail. Additionally, the aspects that are important in practice for the performed simulations will be covered briefly in the following.

Firstly, one needs to ensure that the applied signal, which we assume is a sinusoid, gets its power confined into a single or a few FFT bins (bins are "samples" of the Fourier transform in the frequency domain), rather than getting smeared out in a larger area. This is accomplished by selecting an input frequency such that the sinusoid completes an integer number of cycles during the length of the data record. The result of this is that all the signal power gets confined in one single signal bin, centered at the input frequency. This technique is often called *coherent sampling*. Secondly, an extra concern arises for oversampled ADCs, since only the noise in the in-band portion of the spectrum is included when the ENOB is computed. Out-of-band noise getting smeared into the in-band area will then degrade the result, often seriously. This can be resolved by multiplying the time domain data by a *window function* which, explained shortly, is formed such that signal power will gradually rise in the start of the data record, and gradually fall in the end of the data record such that discontinuities in the start and end are removed. The result in the frequency domain is less smearing such that the noise leakage into the signal band can be held at tolerable levels if the FFT contains enough points. By using

these techniques, the accuracy of the ADC can be estimated by taking the ratio between the power of the signal bins (which there are three of due to the chosen window function, not one), and the power of all the other bins in the signal band. It is also important to use enough points in the FFT to get a good enough estimate of the noise statistics.

In practice, the ENOB of the ADC has been estimated through the use of Cadence Virtuoso's FFT capabilities. Specifically, 2048 points FFTs are taken (where one point in the time domain corresponds to one ADC conversion), a *Hann* window function is used, and the input frequency is set to 359.375 kHz. FFT lengths of 2048 points have been found to yield a good compromise between uncertainty and simulation times. At this length, a single simulation run completes in about a day in the used computer setup. When it comes to uncertainty, this has been investigated by checking the difference between 2048 points and 4096 points simulations, testing the effect of different FFT lengths in behavioral simulations, and by observing differences between simulations that supposedly should have yielded nearly identical results. The uncertainty has not been established quantitatively, but is probably around ± 0.2 bit. This means that when ENOB differences of this order are reported for example across corners, and between adjacent values in sweeps in this chapter, it does probably not have very good confidence. However, such differences can probably be taken as a reliable sign of that the changes are small.

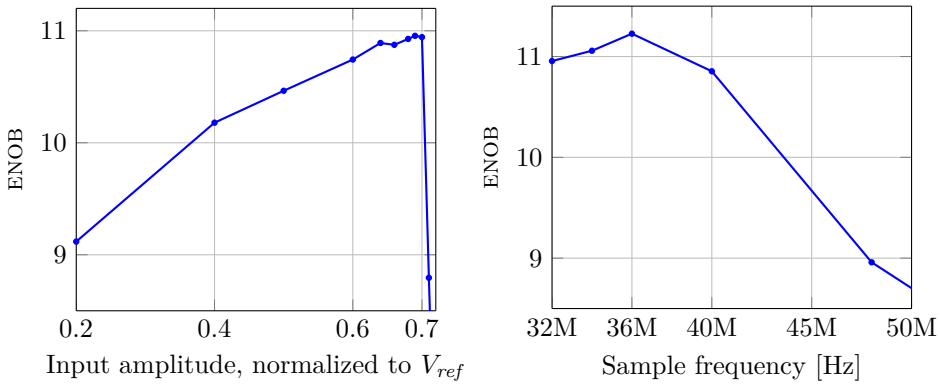
The second most interesting quantity to extract from the simulations is power consumption. This is measured by logging the current flowing through the V_{DD} pin of interest throughout the simulation, and then multiplying the average of this current with V_{DD} (which is ideal in these simulations). This corresponds to solving the integral

$$P = \frac{1}{T} \int_0^T v(t) i(t)\, dt \tag{5.1}$$

when $v(t)$ is a constant, and yields the active power.

5.2 Sweeps of amplitude and sample rate

Since ENOB is derived from the ratio between signal power and the noise and distortion power, it is clearly dependent on the input amplitude. That is, the ENOB will increase as the amplitude is increased. Then, the ADC will saturate at some point, either in the quantizer of for example the integrator, and the ENOB will decrease if the amplitude is increased further. We want to conduct all simulations at this peak point that yields best ENOB, and it is thus necessary to perform an amplitude sweep to find it. The result of such a sweep, carried out at the typical corner, is shown in figure 5.1a. Here, each dot corresponds to an entire transient simulation. The peak point is located at $V_{in,max} = 0.69 V_{ref} = 552$ mV, and that amplitude is thus used for all other simulations.



(a) Amplitude sweep

(b) Sample rate sweep. B_w held constant.

Figure 5.1: ENOB plotted versus input amplitude and sample rate.

One should believe that the peak point should be located significantly closer to V_{ref}, since the SAR ideally reaches full scale when the input has an amplitude of V_{ref}. The reason why this does not happen is because, as also discussed in section 3.3, that the parasitic capacitance introduces attenuation in the DAC, such that its coefficients and equivalent V_{ref} are decreased. From inspections of the DAC time domain waveforms, this equivalent reference voltage can be estimated as $V_{ref,eq} = 562\text{mV}$. From this $V_{in,max} = 0.98V_{ref,eq}$ follows, which must be considered as good performance. The parasitic capacitance clearly decreases the ENOB of the ADC since the input amplitude is limited, but the effect is expected.

It is also interesting to increase f_s from the design value of 32 MHz while the signal bandwidth is retained. This means that the OSR is increased, and better performance can be achieved if the ADC handles higher sample rates than the design target. The result is shown in figure 5.1, and reveals that the performance improves marginally if the sample rate is increased to 36 MHz, which corresponds to an OSR of 9. The simulations also yielded best FOM at this sample rate. Simulation data from a run at this point will be included in the next section, but since the improvement is modest, the design value of 32 MHz will be kept as the sample rate in the rest of the simulations. We also see that the accuracy quickly falls if the sample rate is increased further, and this shows that the ADC is not overdesigned.

5.3 Nominal runs across corners

To verify the performance of the ADC over process variations, nominal simulations (i.e. without mismatch) are performed at the FF, SS, FS, and SF process corners, at 0°C and 60°C temperature. The typical corner at room temperature is also included. The results for the corners that yielded best and worst ENOB and FOM,

as well as for the typical corner, are given in table 5.1.

First we see that the ADC achieves an ENOB equal to the specification of 11.0 at the typical corner. The FOM shows that the ADC is energy efficient, but this is more easily discussed in section 6.1, where comparisons to other ADCs are performed. The performance variations over corners are generally small, and the ADC thus shows good tolerance to process variations. $V_{DD} = 0.8\,\text{V}$ was also used at all corners, and it is possible that the performance at the worst corners can be improved by manual supply voltage adjustment. This has not been tried since the results are still good at the nominal supply voltage. A simulation with OSR = 9 at the typical corner has also been included, and shows slightly better ENOB and FOM, as mentioned.

As a reference, a typical run without noise enabled has also been included in the table. The only things that limit the performance in this run are quantization noise and distortion, and this results in a very high ENOB of 12.9. That confirms that the accuracy is normally limited by thermal noise (flicker is removed efficiently by chopping and auto-zeroing), and is in accordance to the discussions of the noise budget in section 3.3. We also note that the ENOB of 12.9 is lower than the theoretical 14.4 presented in section 3.2. One of the main reasons for this seems to be systematic mismatch in the DAC, as better results has been achieved using an ideal DAC.

Output spectrums for the typical runs with and without noise are shown in figure 5.2. Although it is difficult to compare the shape of the spectrums directly to the ideal transfer function in figure 3.2 due to the logarithmic x-axis, the peak close to f_s due to the poles in the NTF can be recognized. Also, we see that the spectrum is flat in the signal band when noise is included in the simulations. This is due to the unshaped thermal noise, which is the dominating noise contribution in the signal band. When the noise is disabled in figure 5.2b though, the noise looks a bit more shaped also in the signal band, but the sharp roll-off towards DC seen in figure 3.2 is still missing. This is again due to the systematic DAC mismatch, and

Table 5.1: Nominal results for typical, worst and best corners.

	Typical	FF @ 0 °C	SS @ 0 °C	FS @ 0 °C	FF @ 60 °C	Typ. @ best f_s	Typical wo. noise
Comment		Best ENOB	Worst ENOB	Best FOM	Worst FOM		
B_w	2 M	2 M	2 M	2 M	2 M	2 M	2 M
f_s	32 M	32 M	32 M	32 M	32 M	36 M	32 M
OSR	8	8	8	8	8	9	8
ENOB	11.0	11.1	10.7	11.0	10.9	11.2	12.9
Power	67.6 µ	64.8 µ	54.5 µ	58.5 µ	96.3 µ	73.0 µ	67.5 µ
FOM	8.5 f	7.6 f	8.2 f	7.0 f	12.5 f	7.6 f	2.1 f

this missing roll-off is probably the main reason for the discrepancy between ideal and measured ENOB without noise. Finally, it should be noted that it is difficult to spot distortion in either plot. The fifth harmonic can barely be seen close to B_w when the noise is turned off, but is not visible else. Almost equal SNDR and SNR values have also been observed in the simulation data both with and without noise enabled, and distortion thus limits the ADC performance only marginally.

5.4 Mismatch simulation

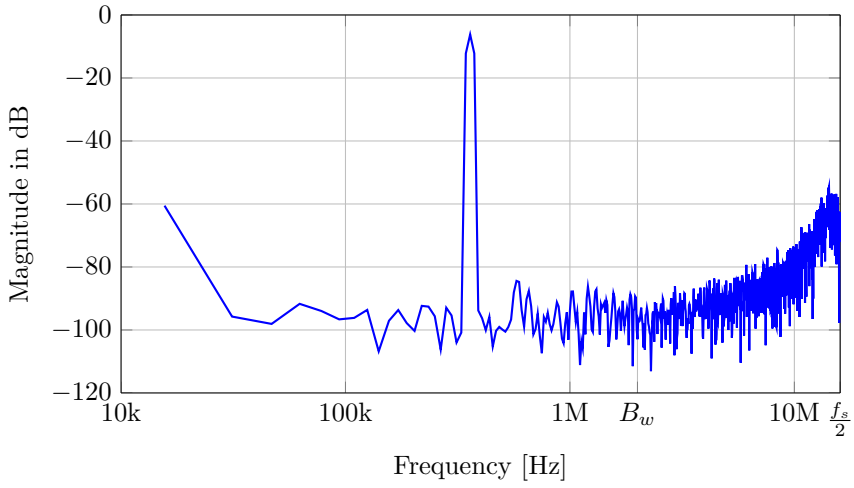
To verify how the ADC performs when mismatch is taken into consideration, Monte Carlo simulations have been carried out. One important thing to know regarding this, is that stochastic capacitor mismatch is not modeled. This is because the capacitor structures that are used are fully custom, and has not been taped-out yet. There does therefore not exist mismatch models for these capacitors.

Simulation results for the ADC after 36 Monte Carlo runs are shown in the upper half of table 5.2, and a histogram for the ENOB distribution is given in figure 5.3a. These data reveal that problems arise in the ADC when mismatch is taken into consideration. From time domain waveforms, it is possible to see that what happens is that about one third of the runs yield an unstable ADC, where the integrator and the quantizer run into saturation. The ADC does thus not work in these runs, and the result is low, supposedly random, ENOB results as seen in figure 5.3a. On the other hand, it is seen that the runs that are stable yield good mismatch performance.

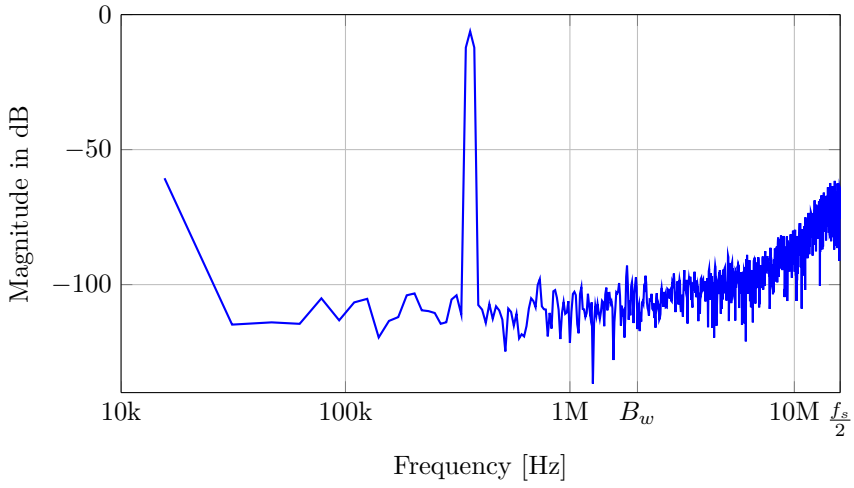
The instability problems are probably due to increased loop filter gain, and this is equivalent to increasing the coefficient β in front of $H(z)$, discussed in section 3.5. With the loop filter coefficients used, it has been found through behavioral simulations in Matlab that the loop stability is very sensitive to increase in this parameter. It was also mentioned in section 3.2 that the chosen coefficients are only stable when the loop gain is limited, but the high sensitivity to loop gain variations was not discovered before the final verification.

Variations in the gain of the loop filter circuit due to mismatch has been checked by running Monte Carlo *pac* simulations in Spectre, and found insignificant. It is therefore probable that the gain variations is due to mismatch in the comparator. In this circuit, which is a part of the SAR, two differential pairs are connected to the DAC plates and to the loop filter output respectively, and are responsible for summing the signals and driving the comparator latch in the correct direction. If mismatch arise between these pairs, the loop filter can get more control over the comparator than the SAR DAC input, and this is equivalent to increasing the loop gain.

The maybe easiest way to solve this problem, is to decrease the filter coefficients such that more variations in the loop gain can be tolerated. During the verification

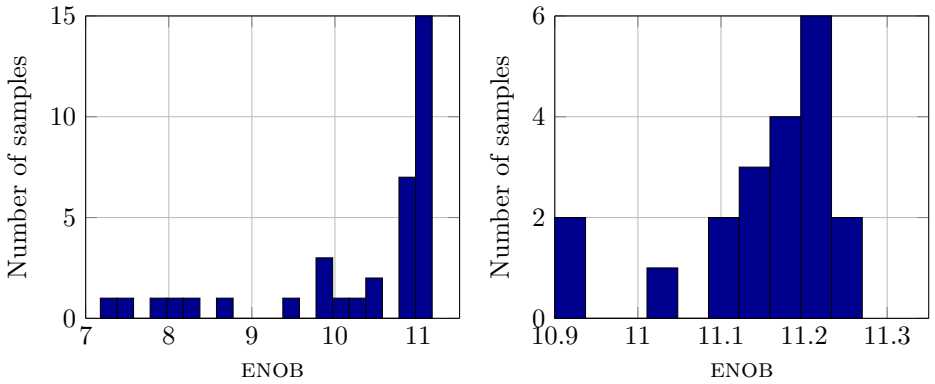


(a) With noise.



(b) Without noise.

Figure 5.2: Output spectra for the ADC at the typical corner.



(a) Original circuit, 36 runs.

(b) Circuit with increased C_i capacitance, 20 runs.

Figure 5.3: ENOB histograms from Monte Carlo runs.

Table 5.2: Monte Carlo results for the ADC.

Original circuit, 36 runs:				
	Mean	Std. dev.	Min.	Max.
ENOB	10.3	1.18	7.2	11.2
Power	67.8 µ	0.43 µ	66.9 µ	68.7 µ
FOM	21.1 f	27.1 f	7.3 f	119 f

Circuit with increased C_i capacitance, 20 runs:				
	Mean	Std. dev.	Min.	Max.
ENOB	11.1	0.10	10.9	11.3
Power	67.8 µ	0.42 µ	67 µ	68.5 µ
FOM	7.5 f	0.54 f	6.9 f	8.9 f

phase, a revised version of the ADC was made where this was done. Specifically, the C_i capacitors was increased by about 14 fF by adding capacitors the holes in the layout seen above the C_{cm} capacitors in figure 4.10. This results in coefficients of about 2.75 and 0.85, rather than 2.9 and 0.9.

Results for this revised circuit after 20 Monte Carlo runs are shown in the bottom half of table 5.2, and in figure 5.3b, and we see that the mismatch performance is now very good. A nominal simulation at the typical corner has also been run on this circuit, and yields an ENOB of 11.0, which is the same as before. The new coefficients therefore seems to have corrected the problem, without impacting the nominal performance.

5.5 Power consumption breakdown

A detailed breakdown of the ADC power consumption is presented in table 5.3, based on a nominal run at the typical corner. When comparing the power consumption of the SAR versus the loop filter, we see that the SAR dominates the power consumption, but not excessively. This suggests that the noise budget is divided sensibly, such that no blocks need to use very much power to cope with too tight noise requirements.

When considering the power consumption in the SAR, it is important to note that the comparator does not dominate the power consumption. This is because its noise is shaped, such that it not needs to be resized when the ENOB is increased by noise shaping. In the loop filter, we see that the buffers use only a small part of the power. This is good, since the buffer solution was introduced as an extra element when compared to the passive sampling solution in [11].

The power consumption of the ADC will be discussed more elaborately in the general discussions in chapter 6, and also in the context of the project thesis [19].

Table 5.3: Power consumption breakdown for the ADC.

Block	Power consumption	% of total
SAR:	44.4 µ	65.7 %
DAC	21.2 µ	31.4 %
Comparator	12.4 µ	18.3 %
SAR logic/rest	10.8 µ	16.0 %
Loop filter:	23.2 µ	34.3 %
Inverters	12.9 µ	19.1 %
Buffers	3.73 µ	5.5 %
Buffer bias circuit	0.330 µ	0.5 %
Clock generator	6.24 µ	9.2 %
Total	67.6 µ	100 %

CHAPTER 6

Discussion

The results chapter shows that the designed ADC meets the design specification in simulations, and that it after a minor modification also is robust against mismatch. Discussions directly regarding the results was also carried out in that chapter. The aim of this chapter is to take one step back and discuss the work of this thesis more in general, and to point out things that could have been done better, are still uncertain, or that might need further investigation in the future.

We start by comparing the ADC to existing state-of-the-art found in the academic literature. After that, a handful of aspects that were not analyzed fully during the design chapters, or that will be unknown variables until after a tape-out will be discussed. Finally, the validity of the work carried out in the project thesis [19] have to be discussed, since it was used to both choose filter topology and the number of bits in the SAR.

6.1 Comparison to the current state-of-the-art

To find published works to compare the designed ADC against, the survey in [10] has been used. Here, the performance of ADCs presented at the ISSCC and VLSI conferences between 1997 and 2015 are tabulated, and because of the high reputation of these conferences, these statistics give a good impression of the current

state-of-the-art when it comes to ADCs in the academic world.

The comparison between the designed NS-SAR and five ADCs found in [10] is given in table 6.1. The results used for the work of this thesis are the mean Monte Carlo results for the ADC version with new coefficients. Also note that the term "best" used in the table comments and in this section always means best among the ADCs found in [10], and best in terms of energy efficiency, i.e. lowest FOM. Correspondingly, the term "all ADCs" should be interpreted as all ADCs found in [10].

From the table, we see that the designed ADC is more energy efficient than all other ADCs that uses oversampling, and among them also [11], which uses the same filter topology as this work. Much of the reason for the better performance compared to this ADC is probably that [11] reports that 75 % of the power consumption is used for the digital part of the ADC. In comparison, it is seen from table 5.3 that the work in this thesis only uses among 25 % for digital parts, if we also count the clock generator. This can partly be explained by the larger technology node used in [11].

When comparing to the other ADCs, we see that the most energy efficient ADC in [10] is almost ten times as energy efficient as this work. However, this ADC has quite low both ENOB and B_w, and if we take this into account and consider the best ADC over 10.5 ENOB, and the best ADC over 10.5 ENOB and 1 MHz B_w, the differences are smaller. Specifically, the energy efficiency of this work and [9], which has quite similar specifications, are almost equal.

From the comparison, we can therefore conclude that the designed ADC is the best NS-SAR and best noise-shaping ADC if the simulation results are correct. Also, the energy efficiency is similar to the best ADC having similar specifications. An important point is also that [7] and [9] uses data driven noise reduction and pipelining respectively, to be able to achieve energy efficient operation in the thermal noise limited accuracy region, as discussed in chapter 1. That this work achieves energy efficiency of the same order as these ADCs thus suggests that noise-shaping can have the same success to extend the energy efficient operating region for ADCs. This has also been the main motivation for the noise-shaping SAR research conducted in this thesis and in [19].

6.2 DAC mismatch

As noted in section 5.4, stochastic capacitor mismatch cannot be modeled due to the custom capacitors. This also means that stochastic DAC mismatch is not modeled. This introduces uncertainty, since the DAC needs to have accuracy on the same order as the entire ADC, as mentioned in for example [19]. Systematic capacitor mismatch is on the other hand modeled through the parasitic extraction.

Table 6.1: Comparison to prior art.

	[4]	[7]	[9]	[31]	[11]	This work
Comment	Best ADC	Best over 10.5 ENOB	Best over 10.5 ENOB and 1M B_w	Best noise-shaping ADC	Best NS-SAR	Mean Monte Carlo results
Type	SAR	SAR	Pipel. SAR	$\Delta\Sigma$	NS-SAR	NS-SAR
Tech.	40 nm	65 nm	65 nm	28 nm	65 nm	28 nm
f_s	200 kHz	32 kHz	50 MHz	640 MHz	88 MHz	32 MHz
B_w	100 kHz	16 kHz	25 MHz	18 MHz	11 MHz	2 MHz
OSR	1	1	1	18	4	8
Area	0.0065 mm^2	0.18 mm^2	0.054 mm^2	0.08 mm^2	0.03 mm^2	0.016 mm^2
ENOB	8.95	11.3	11.5	11.9	10.0	11.1
Power	0.084 µW	0.352 µW	1000 µW	3900 µW	806 µW	67.8 µW
FOM	0.85 f	4.4 f	6.9 f	27.7 f	35.8 f	7.5 f

The first DAC that was used in the NS-SAR had too much systematic mismatch to work properly with noise-shaping, and the DAC finally used has therefore been designed by [20] in parallel with this thesis work. The layout of this new DAC has been done even more carefully than the previous one, and the DAC unit capacitance is also a bit larger. The result has been very good performance when it comes to systematic mismatch, and this is also seen from the ENOB of 12.9 without noise in table 5.1.

Hopefully, the changes done to improve the systematic mismatch can help on stochastic mismatch as well, but the effect will remain unknown until a tape-out is performed. If the performance turns out to be inadequate, circuit techniques to alleviate DAC mismatch can be put into use in future designs. For example, chopping and dithering are used in [7] to mitigate mismatch, and an extra DAC that is switched more energy efficiently is used in [9] to deal with both mismatch and DAC noise in the first pipeline stage.

6.3 The use of loop filter input buffers

One of the main differences between the loop filter used in the designed ADC, and the loop filter in [11], is that input buffers rather than passive sampling are used in this thesis. As described in section 4.1.4, this removes the need to size the DAC capacitance much larger than the sampling capacitance C_s to mitigate signal attenuation.

If we compare the value of $C_s = C_{s,1} + C_{s,2} = 1004$ fF, and the value of $C_{dac} = 280$ fF, it gets clear that the DAC would have needed to be very much larger without

the buffers. However, the size of C_s is partly as large as it is due to the extra noise introduced by the buffers, so if passive sampling was used, C_s could have been decreased. This in turn relaxes the requirement of large DAC capacitance somewhat.

The interesting question if buffers really are better than passive sampling therefore arises. To answer this properly, mathematical formulas for the input referred noise of loop filters with and without buffers are needed. The increase in C_s due to the buffers will then be known, and this can serve as a starting point for an analysis of the two residue sampling methods against each other. Also, much of the same knowledge can be gathered from simulations if the buffers are removed and the ADC is then resized to meet the same specifications.

Despite the lack of of such an analysis, it is still possible to reason a bit around the energy efficiency of the buffer solution. Consider the power consumption breakdown that was given in table 5.3. Here we see that the buffers plus the bias circuit only use 5 % of the overall power, and this means that power consumption impact due to the buffers themselves are small. This is mostly due to powering them down when not used. However, the inverters that uses 19 % of the power are probably also sized up due to the buffers, since increased C_s capacitance also leads to increased inverter load. Although this increase is unknown, the inverters and buffers in sum still uses less power that the DAC, which is at over 31 %. This means that if the buffers are removed and the inverters are sized down, the power consumption in the circuit will most certainly still increase since the DAC size must now be increased due to passive sampling. The amount of increase is unknown in lack of a proper analysis, but the power consumption of the ADC will nevertheless increase fast since the DAC already uses much power.

Although the reasoning above do not prove that buffers are the best design choice, it makes it quite evident that they are not a bad design choice neither. This statement is also supported by the good energy efficiency achieved in the ADC when comparing to previous state-of-the-art. Nonetheless, a definitive analysis would clearly have been interesting to fully conclude on the buffer idea.

6.4 Division of the noise budget

In section 3.4, an analysis of how the noise budget should be divided between shaped and unshaped noise was performed, and the actual division of noise in the ADC was found to be close to optimal. In addition to this analysis, it is still possible to go a step further and also analyze how the unshaped noise should be divided between the DAC and the loop filter. Such an analysis should not be too difficult to perform when one both knows how much power the loop filter and the DAC uses, and how much noise they generate when using this power. These data can be used as input to an analysis that finds the noise contribution combination that

minimizes the power.

Such an analysis has not been carried out since the DAC has nevertheless been sized out from systematic mismatch requirements. Still, it was shown in 3.3 that the DAC noise power is at 3.7 nV^2, which is a significant part of the total noise budget of 24 nV^2. One could therefore argue that the DAC is sized from both mismatch and noise requirements at the same time. Also, since it is seen from table 5.3 that the DAC also is the block that uses most power, it would be interesting to know the exact capacitance value that minimizes the power consumption. However, from the current relative power consumption, it is most probable that this value is lower than the one currently used, and since the situation regarding stochastic mismatch is unknown, it is probably best to not try to decrease the DAC capacitance until after a successful tape-out.

6.5 Choices made according to the project thesis

In chapter 3, the behavioral simulation framework from the specialization project [19] was used to choose both topology and the number of bits in the SAR. Since these choices are important when it comes to energy efficiency, an important question is, how valid are the assumptions and models that this simulation framework is built upon?

In section 3.2 it is already made clear that the high level design parameters that come from the behavioral simulations cannot be used "out of the box", since only shaped noise is taken into consideration. Due to this, an OSR value had to be chosen manually, and although an exhaustive analysis of this choice was carried out, the high level design could not be done in accordance to the behavioral simulations and the confidence they were supposed to give.

In addition to the lack of unshaped noise in the behavioral simulations, there are also shortcomings in the power consumption models that was introduced in [19], and that influences all results regarding energy efficiency. In the power consumption model for the SAR, it was assumed that power consumption of the whole SAR, including the DAC, only scales according to the number of bits and the sample rate. It was mentioned that the DAC in the reality has to be sized according to the target ENOB, but the impact on the power consumption was assumed to be insignificant. When looking at the relative DAC power consumption of 31.4 % in table 5.3, we can conclude that this assumption does not hold in the designed ADC. The SAR power consumption estimates in the behavioral simulation framework are thus too optimistic.

When it comes to the loop filter power consumption model, it was assumed that the loop filter power scales linearly with filter order, i.e. with the numbers of integrators. When the circuit noise of the loop filter itself is taken into account, this assumption is wrong. This is because if we consider a filter of order two or

more as a cascade of integrator stages, it is only the input referred noise of the first stage that appear directly as input referred noise for the whole ADC. The noise sources on the input of the other stages have to be divided by the squared transfer functions of the stages in front of them to be referred to the input of the ADC, and the noise requirements thus fall sharply after the first stage. This means that the first stage will use considerably more power than the other stages, and this is not modeled. The impact of this modeling error may be quite severe when it comes to the evaluation of the different topologies, since the filter power consumption is greatly overestimated when the filter order is increased past one.

The conclusion from the considerations above is that although an energy efficient ADC is designed in this thesis, it is possible that its high level design can be optimized even more. Especially interesting is the error regarding the filter power consumption, since it means that filters of higher order than the one used in this thesis may give even better energy efficiency. To investigate this fully, the behavioral simulation framework can be updated in light of this thesis to produce more reliable results. It should not be too difficult to do this, since it is mainly the models that need a revision, and not the simulation/optimization approach that was used. Also, unshaped noise can be taken into account by adding more noise sources to the behavioral simulations.

6.6 Energy efficiency of NS-SARs versus standard SARs

In [19], it was found that the an NS-SAR operating in the thermally limited region should have a FOM comparable to the SAR alone operated in the process limited region. The accuracy of this conclusion may also be impacted by the shortcomings in the project thesis pointed out in the preceding section, but it is still interesting to see how this assertion turns out for the designed NS-SAR.

The FOM of the SAR with the new DAC used in this thesis has still not been measured on its own since the focus has been to get it working in the noise-shaping setup. We therefore choose to compare the NS-SAR against a previous 9-bit SAR made by [20] in 28 nm FDSOI. This SAR is also a part of a prototype tape-out currently in production, and is thus fully finished. Moreover, the DAC of this SAR has not been improved for noise-shaping operation, and thus uses less power. This makes the comparison even more strict.

The mentioned SAR exhibits a FOM of 3.1 fJ/conv-step, width an ENOB of 8.1 bit. The FOM is thus better than the 7.5 fJ/conv-step for the NS-SAR, but only by a factor 2.4. The assumption done in [19] is therefore not totally correct for the designed ADC.

However, if we assume that the pure SAR operates at the edge between the process

and thermally limited regimes (which is not far from the truth), and consider the FOM it will have at 11.1 bit ENOB after pure thermal scaling, we can better see the how energy efficient the noise-shaping is. Specifically, we assume that the energy quadruple per bit, and will thus get a FOM of 24.8 fJ/conv-step for the pure SAR. Even if we round this a bit down to say 20 fJ/conv-step to make the approximation a bit more conservative, it seems clear that it has been more energy efficient to increase the accuracy through the use of noise-shaping in this thesis.

CHAPTER 7

Conclusion

In this thesis, an energy efficient noise-shaping SAR ADC for medical ultrasound applications has been designed in 28 nm FDSOI. According to post-layout Monte Carlo simulations, the ADC has an accuracy of 11.1 bit ENOB, and this is in compliance with the design specifications. The signal bandwidth is 2 MHz, and an OSR of 8 is used.

The 9-bit SAR used as quantizer in the ADC is designed by [20], whereas the loop filter design and its interfacing towards the SAR, as well as the general high-level design have been carried out in this thesis. The final design used during the verification is in the form of a single, top-level circuit layout for the whole NS-SAR.

The used loop filter topology is the same as the one used in [11], and was chosen based on the specialization project [19] preceding this thesis. Compared to [11], some improvements have been done in the switched-capacitor circuit implementation of the topology, and the most important is the use of chopped input buffers to extract residue voltages from the DAC. As opposed to the passive residue extraction solution in [11], this facilitates the use of a DAC capacitance that is smaller or of the same size as the sampling capacitance in the loop filter. Also, auto-zeroed, cascoded inverters rather than a standard OTA have been used as gain elements in the switched-capacitor structure, and this facilitates better energy efficiency.

When compared the current state-of-the-art, the designed ADC is found to have

good energy efficiency. Specifically, a FOM of 7.5 fJ/conv-step is achieved in post-layout Monte Carlo simulations, and to the best of the author's knowledge, this is better than the current state-of-the-art of noise-shaping ADCs. When all kinds of ADCs are taken into consideration, the designed NS-SAR seems to have a FOM similar to the current state-of-the-art in the same specification range.

As mentioned, the loop filter topology was chosen based on the specialization project [19]. During the work on this master thesis, however, some of the assumptions done in [19] have been found invalid. This means that despite the good energy efficiency achieved with the current loop filter topology, it is still possible that even better energy efficiency can be achieved with one of the other topologies considered in the specialization project.

7.1 Further work

As the author of this thesis will continue the research on noise-shaping SARs as a Ph.D. student at NTNU, it is planned that the designed ADC will be a part of a tape-out this fall. Hopefully, no major changes in the design need to be carried out before this, but more extensive verification performed at the chip top-level will be conducted. It is also planned to revise the behavioral simulation framework from [19] based on the knowledge from this thesis, such that it can predict optimal topologies and high level design parameters more accurately.

The path after this will clearly depend on the measurement results of the taped-out circuit, but if they are satisfying, it can be interesting to make a new NS-SAR based on updated information from the simulation framework. Also, to optimize the ADC even more at the circuit level, more elaborate noise analyses than carried out in this work can be conducted. Another thing that will be interesting in a new NS-SAR is to increase the target ENOB, and try to push the energy efficient operating region of ADCs further.

If the measurement results are not satisfying, more research can be directed towards the found problems. The DAC might have too much mismatch for example, and both new and existing ways to mitigate such problems can then be researched. Moreover, the DAC is the block that uses most power in the current version of the ADC, so more energy efficient DACs is also an interesting research topic either way.

Bibliography

[1] B. Murmann, "Energy limits in A/D converters," in *2013 IEEE Faible Tension Faible Consommation (FTFC)*, Jun. 2013, pp. 1–4. DOI: 10.1109/FTFC.2013.6577781.

[2] B. Murmann, "A/D converter trends: power dissipation, scaling and digitally assisted architectures," in *IEEE Custom Integrated Circuits Conference, 2008. CICC 2008*, Sep. 2008, pp. 105–112. DOI: 10.1109/CICC.2008.4672032.

[3] T. Sundstrom, B. Murmann, and C. Svensson, "Power dissipation bounds for High-Speed nyquist Analog-to-Digital converters," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 56, no. 3, pp. 509–518, Mar. 2009, ISSN: 1549-8328. DOI: 10.1109/TCSI.2008.2002548.

[4] H. Tai, Y. Hu, H. Chen, and H. Chen, "11.2 a 0.85fJ/conversion-step 10b 200kS/s subranging SAR ADC in 40nm CMOS," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2014 IEEE International*, Feb. 2014, pp. 196–197. DOI: 10.1109/ISSCC.2014.6757397.

[5] P. Harpe, H. Gao, R. van Dommele, E. Cantatore, and A. van Roermund, "21.2 a 3nW signal-acquisition IC integrating an amplifier with 2.1 NEF and a 1.5fJ/conv-step ADC," in *Solid- State Circuits Conference - (ISSCC), 2015 IEEE International*, Feb. 2015, pp. 1–3. DOI: 10.1109/ISSCC.2015.7063086.

[6] P. Harpe, E. Cantatore, and A. van Roermund, "A 10b/12b 40 kS/s SAR ADC with Data-Driven noise reduction achieving up to 10.1b ENOB at 2.2 fJ/Conversion-Step," *IEEE Journal of Solid-State Circuits*, vol. 48, no. 12, pp. 3011–3018, Dec. 2013, ISSN: 0018-9200. DOI: 10.1109/JSSC.2013.2278471.

[7] P. Harpe, E. Cantatore, and A. van Roermund, "11.1 an oversampled 12/14b SAR ADC with noise reduction and linearity enhancements achieving up to 79.1dB SNDR," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2014 IEEE International*, Feb. 2014, pp. 194–195. DOI: 10.1109/ISSCC.2014.6757396.

[8] B. Verbruggen, K. Deguchi, B. Malki, and J. Craninckx, "A 70 dB SNDR 200 MS/s 2.3 mW dynamic pipelined SAR ADC in 28nm digital CMOS," in *2014 Symposium on VLSI Circuits Digest of Technical Papers*, Jun. 2014, pp. 1–2. DOI: 10.1109/VLSIC.2014.6858451.

[9] Y. Lim and M. Flynn, "26.1 a 1mW 71.5dB SNDR 50MS/S 13b fully differential ring-amplifier-based SAR-assisted pipeline ADC," in *Solid- State Circuits Conference - (ISSCC), 2015 IEEE International*, Feb. 2015, pp. 1–3. DOI: 10.1109/ISSCC.2015.7063124.

[10] B. Murmann. (). ADC performance survey 1997-2015, [Online]. Available: http://web.stanford.edu/~murmann/adcsurvey.html.

[11] J. Fredenburg and M. Flynn, "A 90-MS/s 11-MHz-Bandwidth 62-dB SNDR Noise-Shaping SAR ADC," *IEEE Journal of Solid-State Circuits*, vol. 47, no. 12, pp. 2898–2904, Dec. 2012, ISSN: 0018-9200. DOI: 10.1109/JSSC.2012.2217874.

[12] K. Kim, J. Kim, and S. Cho, "Nth-order multi-bit sigma-delta ADC using SAR quantiser," *Electronics Letters*, vol. 46, no. 19, pp. 1315–1316, Sep. 2010, ISSN: 0013-5194. DOI: 10.1049/el.2010.1554.

[13] Z. Chen, P. Zhang, H. Wei, S. Sin, S. U, R. Martins, and Z. Wang, "Noise shaping implementation in two-step/SAR ADC architectures based on delayed quantization error," in *2011 IEEE 54th International Midwest Symposium on Circuits and Systems (MWSCAS)*, Aug. 2011, pp. 1–4. DOI: 10.1109/MWSCAS.2011.6026454.

[14] R. Schreier and G. C. Temes, *Understanding Delta-Sigma Data Converters*. Wiley, Nov. 2004, ISBN: 9780471465850.

[15] R. Inanlou, M. Shahghasemi, and M. Yavari, "A noise-shaping SAR ADC for energy limited applications in 90 nm CMOS technology," *Analog Integrated Circuits and Signal Processing*, vol. 77, no. 2, pp. 257–269, Sep. 2013, ISSN: 0925-1030, 1573-1979. DOI: 10.1007/s10470-013-0147-2. [Online]. Available: http://link.springer.com/article/10.1007/s10470-013-0147-2.

[16] M. Shahghasemi, R. Inanlou, and M. Yavari, "An error-feedback noise-shaping SAR ADC in 90 nm CMOS," *Analog Integrated Circuits and Signal Processing*, vol. 81, no. 3, pp. 805–814, Oct. 2014, ISSN: 0925-1030, 1573-1979. DOI: 10.1007/s10470-014-0434-6. [Online]. Available: http://link.springer.com/article/10.1007/s10470-014-0434-6.

[17] R. Inanlou and M. Yavari, "A simple structure for noise-shaping SAR ADC in 90 nm CMOS technology," *AEU - International Journal of Electronics and Communications*, 2015, ISSN: 1434-8411. DOI: 10.1016/j.aeue.2015.04.006. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1434841115001077.

[18] H. Park and M. Ghovanloo, "A 13-bit noise shaping SAR–ADC with dual-polarity digital calibration," *Analog Integrated Circuits and Signal Processing*, vol. 75, no. 3, pp. 459–465, Feb. 2013, ISSN: 0925-1030, 1573-1979. DOI: 10.1007/s10470-013-0050-x. [Online]. Available: http://link.springer.com/article/10.1007/s10470-013-0050-x.

[19] H. Garvik, "Noise-shaping SAR ADCs: a systematic study, investigation of energy-efficiency, and behavioral simulation," Specialization project thesis, Norwegian University of Science and Technology, Trondheim, Dec. 2014.

[20] C. Wulff, private communication, 2015.

[21] T. C. Carusone, D. Johns, and K. W. Martin, *Analog Integrated Circuit Design*. Wiley, 2012, ISBN: 9781118092330.

[22] J. de la Rosa, "Sigma-Delta modulators: tutorial overview, design guide, and State-of-the-Art survey," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 58, no. 1, pp. 1–21, Jan. 2011, ISSN: 1549-8328. DOI: 10.1109/TCSI.2010.2097652.

[23] C. Enz and G. Temes, "Circuit techniques for reducing the effects of op-amp imperfections: autozeroing, correlated double sampling, and chopper stabilization," *Proceedings of the IEEE*, vol. 84, no. 11, pp. 1584–1614, Nov. 1996, ISSN: 0018-9219. DOI: 10.1109/5.542410.

[24] F. Silveira, D. Flandre, and P. Jespers, "A gm/ID based methodology for the design of CMOS analog circuits and its application to the synthesis of a silicon-on-insulator micropower OTA," *IEEE Journal of Solid-State Circuits*, vol. 31, no. 9, pp. 1314–1319, Sep. 1996, ISSN: 0018-9200. DOI: 10.1109/4.535416.

[25] J. Fischer, "Noise sources and calculation techniques for switched capacitor filters," *IEEE Journal of Solid-State Circuits*, vol. 17, no. 4, pp. 742–752, Aug. 1982, ISSN: 0018-9200. DOI: 10.1109/JSSC.1982.1051806.

[26] Y. Chae and G. Han, "Low voltage, low power, Inverter-Based Switched-Capacitor Delta-Sigma modulator," *IEEE Journal of Solid-State Circuits*, vol. 44, no. 2, pp. 458–472, Feb. 2009, ISSN: 0018-9200. DOI: 10.1109/JSSC.2008.2010973.

[27] B. Razavi, *Fundamentals of Microelectronics, 2nd Edition*. Wiley Global Education, Mar. 2013, ISBN: 9781118559574.

[28] B. Razavi, *Design of Analog CMOS Integrated Circuits*. Tata McGraw-Hill, Oct. 2002, ISBN: 9780070529038.

[29] B. Hernes, J. Bjornsen, T. Andersen, A. Vinje, H. Korsvoll, F. Telsto, A. Briskemyr, C. Holdo, and O. Moldsvor, "A 92.5mW 205MS/s 10b pipeline IF ADC implemented in 1.2V/3.3V 0.13 #x003bc;m CMOS," in *Solid-State Circuits Conference, 2007. ISSCC 2007. Digest of Technical Papers. IEEE International*, Feb. 2007, pp. 462–615. DOI: 10.1109/ISSCC.2007.373494.

[30] O. Oliaei, "Noise analysis of correlated double sampling SC integrators with a hold capacitor," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 50, no. 9, pp. 1198–1202, Sep. 2003, ISSN: 1057-7122. DOI: 10.1109/TCSI.2003.816314.

[31] Y. Shu, J. Tsai, P. Chen, T. Lo, and P. Chiu, "A 28fJ/conv-step CT delta-sigma modulator with 78dB DR and 18MHz BW in 28nm CMOS using a highly digital multibit quantizer," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2013 IEEE International*, Feb. 2013, pp. 268–269. DOI: 10.1109/ISSCC.2013.6487729.

[32] J. Silva, U. Moon, J. Steensgaard, and G. Temes, "Wideband low-distortion delta-sigma ADC topology," *Electronics Letters*, vol. 37, no. 12, pp. 737–738, Jun. 2001, ISSN: 0013-5194. DOI: 10.1049/el:20010542.

[33] D. Zhang, C. Svensson, and A. Alvandpour, "Power consumption bounds for SAR ADCs," in *2011 20th European Conference on Circuit Theory and Design (ECCTD)*, Aug. 2011, pp. 556–559. DOI: 10.1109/ECCTD.2011.6043594.

APPENDIX A

Noise-shaping SAR overview

This is a chapter from the project thesis included here to give extra coverage of the noise-shaping SAR *principle.*

The actual ADC performing all the quantizations in a delta-sigma modulator can in principle be of any type, including SAR. The most straight-forward way to use a SAR to do quantizations in a delta-sigma modulator would be to just swap the ADC block in the general delta-sigma structure with a SAR, as shown in figure A.1. Considering the figure, we realize that when doing this, we still need a separate DAC outside the SAR to implement the feedback from the digital to the analog domain. Taking this into account, we see that the overall amount of circuitry in the modulator will probably be substantially larger than for the SAR alone. This makes energy efficient design more demanding.

It is, however, possible to use a SAR in a delta-sigma modulator in a much more efficient way, by realizing that the signal $q(n)$ that enters the loop filter in figure A.1 is actually the negative of the final SAR residue voltage V_{res}, which can be obtained from the SAR DAC top plates after a completed conversion. If we utilize this fact, it is not necessary to have an extra DAC in the feedback loop, and the system complexity immediately comes closer to that of the SAR alone. In simple terms, we can say that this approach is more like adding delta-sigma techniques to the inside of the SAR structure, rather than to just include a SAR in a delta-sigma modulator. It is thus natural to call delta-sigma modulators of this form for noise-shaping SARs

(NS-SAR).

To the best of the author's knowledge, the first and only NS-SAR designed and fabricated to this date is a 10.0 ENOB one presented by Fredenburg and Flynn in [11]. Before this, the general idea of exploiting the top plate residue voltage of the SAR was presented by Kim, Kim and Cho in [12].

In [11], the most simple way to do noise-shaping in a SAR is presented initially, and this is also done in this overview as an instructive example. After this, we will generalize the NS-SAR structure and establish a general linear model in addition to general expressions for the NTF and STF. This generalization is not done in [11], because this paper focuses more on a concrete NS-SAR implementation rather than on general discussions. Lastly, we will look into what happens to the thermal noise in the different blocks of the SAR when noise shaping is introduced. This topic is also extensively treated in [11].

A.1 Simple noise shaping in a SAR

In figure A.2, simple noise shaping is added to a 4-bit charge-redistribution SAR. Specifically, the negative of the final value of the residue voltage $-V_{res}$ is sampled onto a capacitor after a completed conversion $n-1$, and held there during the consecutive conversion n. The capacitor is connected to the comparator positive terminal, meaning that comparisons during the binary search are now on the form

$$-V_{in}(n) + V_{D/A} < -V_{res}(n-1) \tag{A.1}$$

where $V_{D/A}$ is just the current DAC voltage at some point of the binary search. If we rearrange (A.1) into

$$-[V_{in}(n) - V_{res}(n-1)] + V_{D/A} < 0 \tag{A.2}$$

we see that the value $-V_{res}(n-1)$ on the capacitor is added to the input, and therefore also to the output. The output can then be written down as

$$D_{out}(n) = V_{in}(n) - V_{res}(n-1) + V_Q(n) \tag{A.3}$$

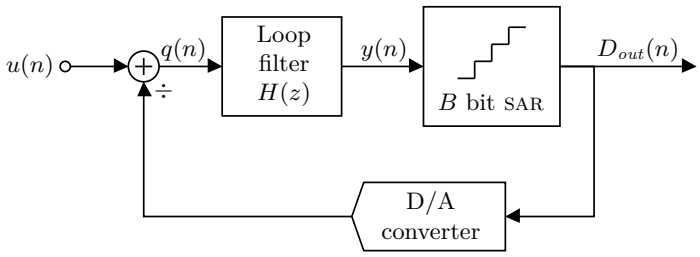


Figure A.1: Straight-forward way to use a SAR in a delta-sigma modulator.

where $V_Q(n)$ is the quantization error of the current sample n. If we then write the residue voltage as

$$V_{res}(n) = D_{out}(n) - V_{in}(n) \tag{A.4}$$

and put this into equation (A.3), we get

$$D_{out}(n) = V_{in}(n) - D_{out}(n-1) + V_{in}(n-1) + V_Q(n) \tag{A.5}$$

Finally, we take take the z-transform and rearrange to get

$$D_{out}(z) = V_{in}(z) + \frac{1}{1+z^{-1}} V_Q(z) \tag{A.6}$$

From this we see that the input still passes straight through to the output, while the quantization error is shaped by the term $\frac{1}{1+z^{-1}}$, which turns out to be a simple high pass filter. Noise shaping is thus achieved by modifying the SAR in this way. The maximum noise attenuation in the signal band is in this case only -6 dB, and this quite poor performance will not lead to attractive bandwidth-accuracy trade-offs when it comes to energy efficiency [11].

A.2 Generalization of the noise-shaping SAR

If the $-V_{res}$ sample capacitor (see figure A.2) in the simple NS-SAR is exchanged for some kind of loop filter, more effective noise shaping will be achieved. We can formalize this idea for an arbitrary loop filter $H(z)$ by deriving a general NS-SAR linear model, and a general NS-SAR NTF and STF. When having this at hand, further exploration of possible filters $H(z)$ is possible.

To obtain the model, NTF and STF, we first consider figure A.3a, which shows a linear model for the simple noise-shaping SAR. The model is quite simple; The sum of $-V_{res}(n-1)$ and $V_{in}(n)$ is digitized into the signal $D_{out}(n)$, and a white noise

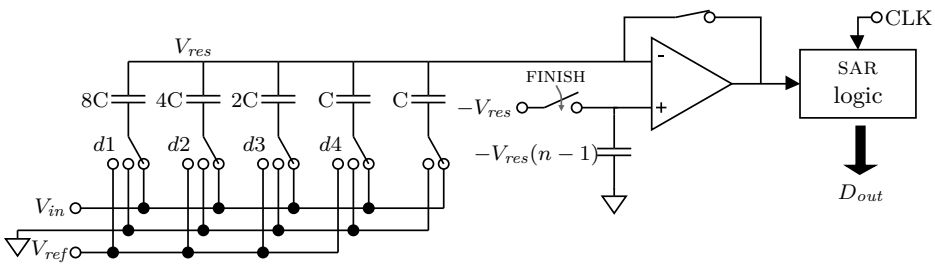
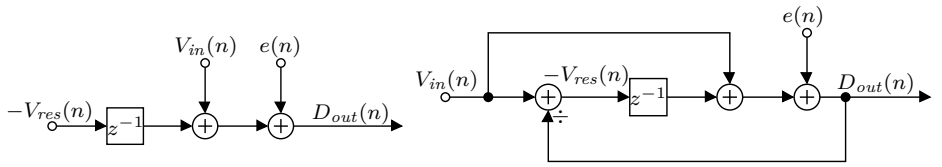


Figure A.2: A simple noise shaping SAR adding the previous residue to the current output.



(a) Shown as consistent as possible with respect to the physical circuit.

(b) Logical signal flow paths for V_{res} added.

Figure A.3: Linear models for the simple NS-SAR.

source $e(n)$ is used to represent the quantization errors. This is in accordance to what was derived for the NS-SAR circuit in section A.1.

If we then express $-V_{res}(n)$ in this model in terms of $V_{in}(n)$ and $D_{out}(n)$ by drawing some extra arrows, we arrive at the model in figure A.3b. This model starts to resemble that of a delta-sigma modulator (was given in figure 2.3), but with a specific loop filter z^{-1} rather than a general loop filter $H(z)$, and an extra feed forward from $V_{in}(n)$ to the quantizer input. We now realize that a generalization of the model can be made by just changing the loop filter function z^{-1}, realized by the $-V_{res}$ sampling capacitor, with a general loop filter $H(z)$. This gives the model in figure A.4, which is the general NS-SAR linear model used in the rest of this work. Here, the signal names are also changed to more general ones.

We can now obtain the general NS-SAR NTF and STF. By using the model, the output can be written down in the z-plane as

$$D_{out}(z) = U(z) + \frac{1}{1 + H(z)} E(z) \tag{A.7}$$

which gives

$$\mathrm{STF}(z) = \frac{D_{out}(z)}{U(z)} = 1 \tag{A.8}$$

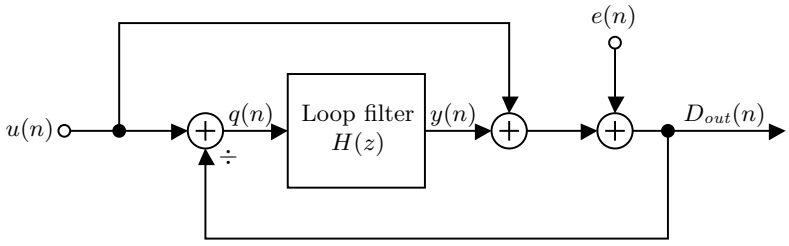


Figure A.4: General NS-SAR linear model.

and

$$\mathrm{NTF}(z) = \frac{D_{out}(z)}{E(z)} = \frac{1}{1+H(z)} \tag{A.9}$$

This means that the NTF of the NS-SAR structure is equal to the general delta-sigma NTF, while the STF is always unity because of the feed-forward in the model. This is a very desirable property, because the loop filter input and output then takes the form

$$Q(z) = -\frac{1}{1+H(z)}E(z) \tag{A.10}$$

$$Y(z) = -\frac{H(z)}{1+H(z)}E(z) \tag{A.11}$$

which is not dependent on the modulator input $U(z)$. This means that the signal swing inside $H(z)$ can be kept small if $E(z)$ is made small by the choice of a reasonable high bit count B in the SAR. The circuitry complexity of $H(z)$ can thus be held simpler. This advantage of an unity NTF was probably first pointed out in [32], and is also commonly utilized in standard delta-sigma modulators by the introduction of the same feed-forward path as in the NS-SAR model.

One more thing to mention about the NS-SAR model is that the feedback from D_{out} is not a physical signal inside the NS-SAR, because the residue signal $q(n)$ is generated directly by the internal SAR DAC. This means that if further generalizations of the model should be done, then $D_{out}(z)$ should probably not be allowed to enter into nodes of the loop filter on its own, as this will imply that an analog version of $D_{out}(n)$ has to be created explicitly.

A.3 Thermal noise in the noise-shaping SAR

As stated in the introduction, one of the main aims of this work is to decouple the ADC accuracy from thermal noise constraints. How thermal noise in the NS-SAR affects the accuracy is hence of primary concern. This topic is treated extensively in [11], which is thus the basis for the discussions in this section.

The first and most important thing to point out is that the NS-SAR also noise-shapes the comparator noise. This can be seen by realizing that the comparator noise will enter into the linear model in figure A.4 through the same terminal as the quantization noise $e(n)$. The comparator noise will thus be treated similarly as the quantization noise by the modulator, and thus noise-shaped. This means that the comparator noise does not need to be lowered when noise-shaping is introduced to a SAR.

This noise-shaping of the comparator noise is more or less pointed out as the main advantage of the NS-SAR in [11], and this is probably indeed the case since the comparator often dominate the power consumption at medium to high ADC

resolutions [33]. This means that if the comparator had to be made to the accuracy of the whole NS-SAR, then any energy-efficiency increase would probably not have been possible compared to a pure SAR.

Noise in the DAC is also present in a SAR, and its noise power is given by the common formula $\frac{kT}{C}$, where C is the total DAC capacitance (k is Boltzmann's constant and T is absolute temperature). This noise enters into the NS-SAR model at the same terminal as the input, and is thus not noise-shaped. The oversampling will, however, attenuate the DAC noise, but only by 3 dB per OSR octave.[1]. This means that it might be necessary to increase the DAC capacitance to achieve the wanted accuracy, and this will thus lead to higher power consumption. Whether this is a deal-breaker for energy efficiency improvements or not, depends on if the DAC dominates the overall SAR power consumption. If we turn to [33], this is at least not the case in the SAR power consumption model presented there.

[1] Equivalent to the quantization noise in a oversampling converter

APPENDIX B

Layout

In this appendix, some more pictures of the loop filter and clock generator layout are provided.

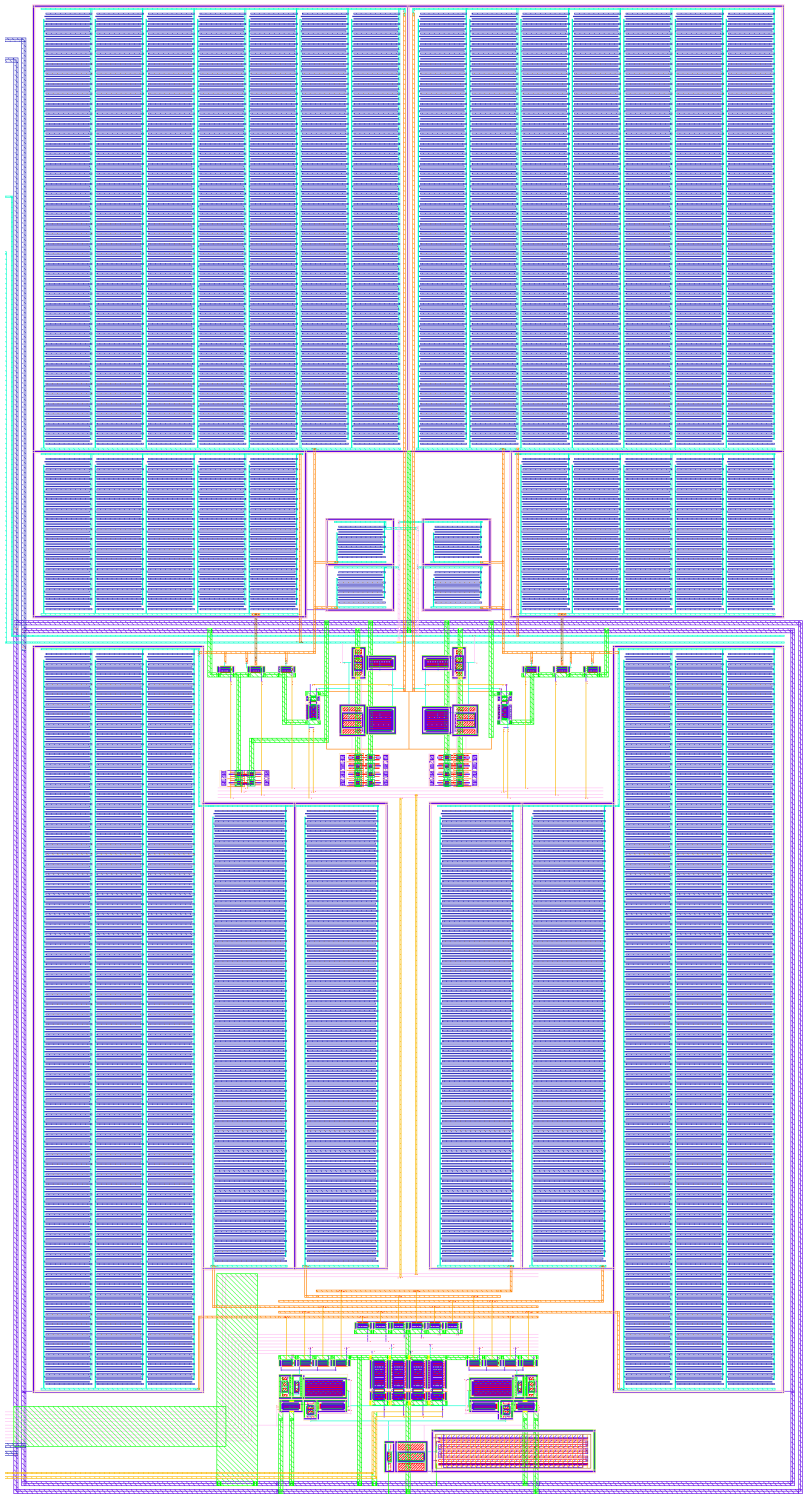


Figure B.1: The layout of the loop filter.

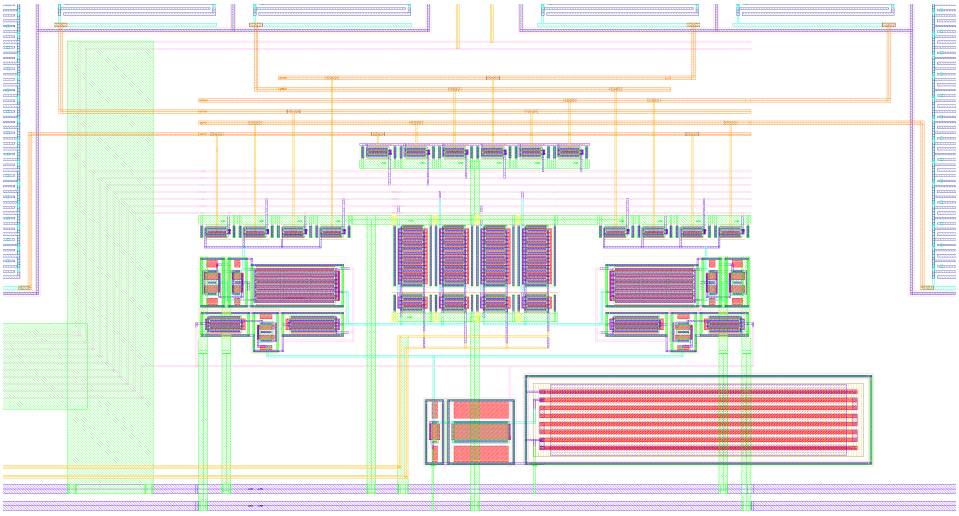


Figure B.2: The layout of the buffers, bias circuit and chopping/sampling switches.

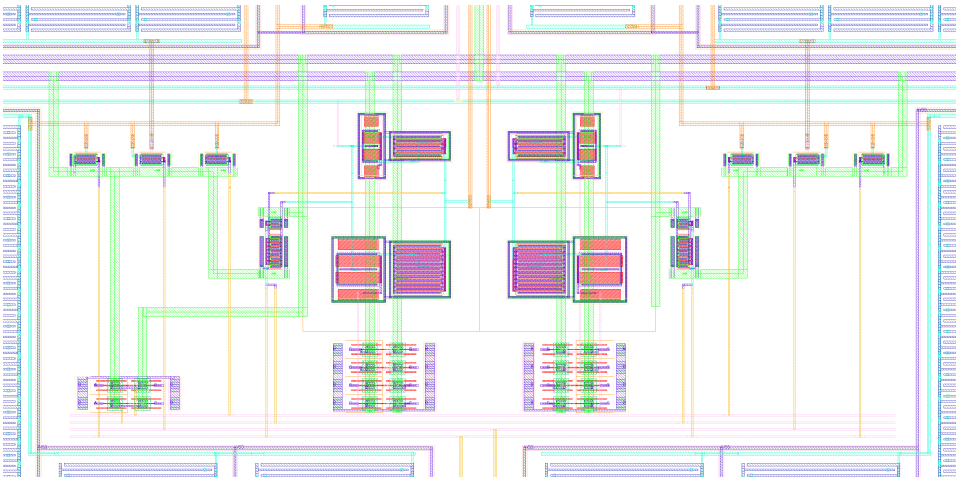


Figure B.3: The layout of the inverters and related switches.

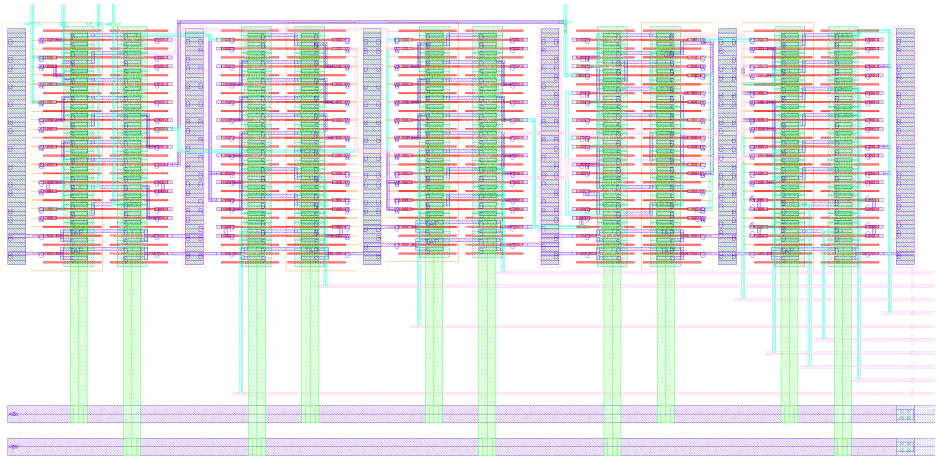
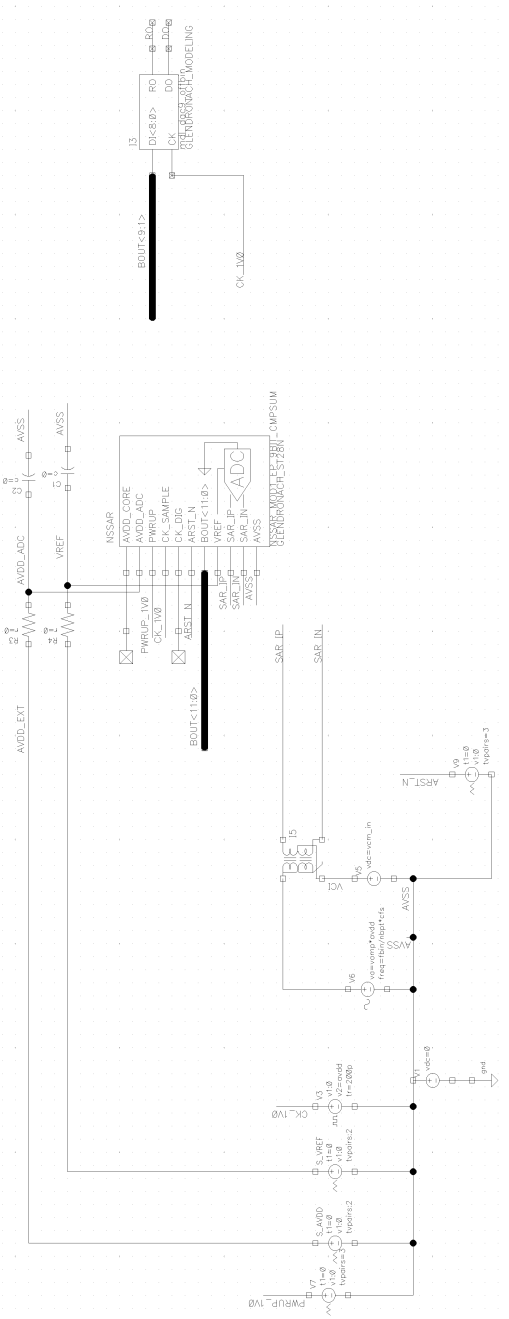


Figure B.4: The layout of the non-overlapping clock generator.

APPENDIX C

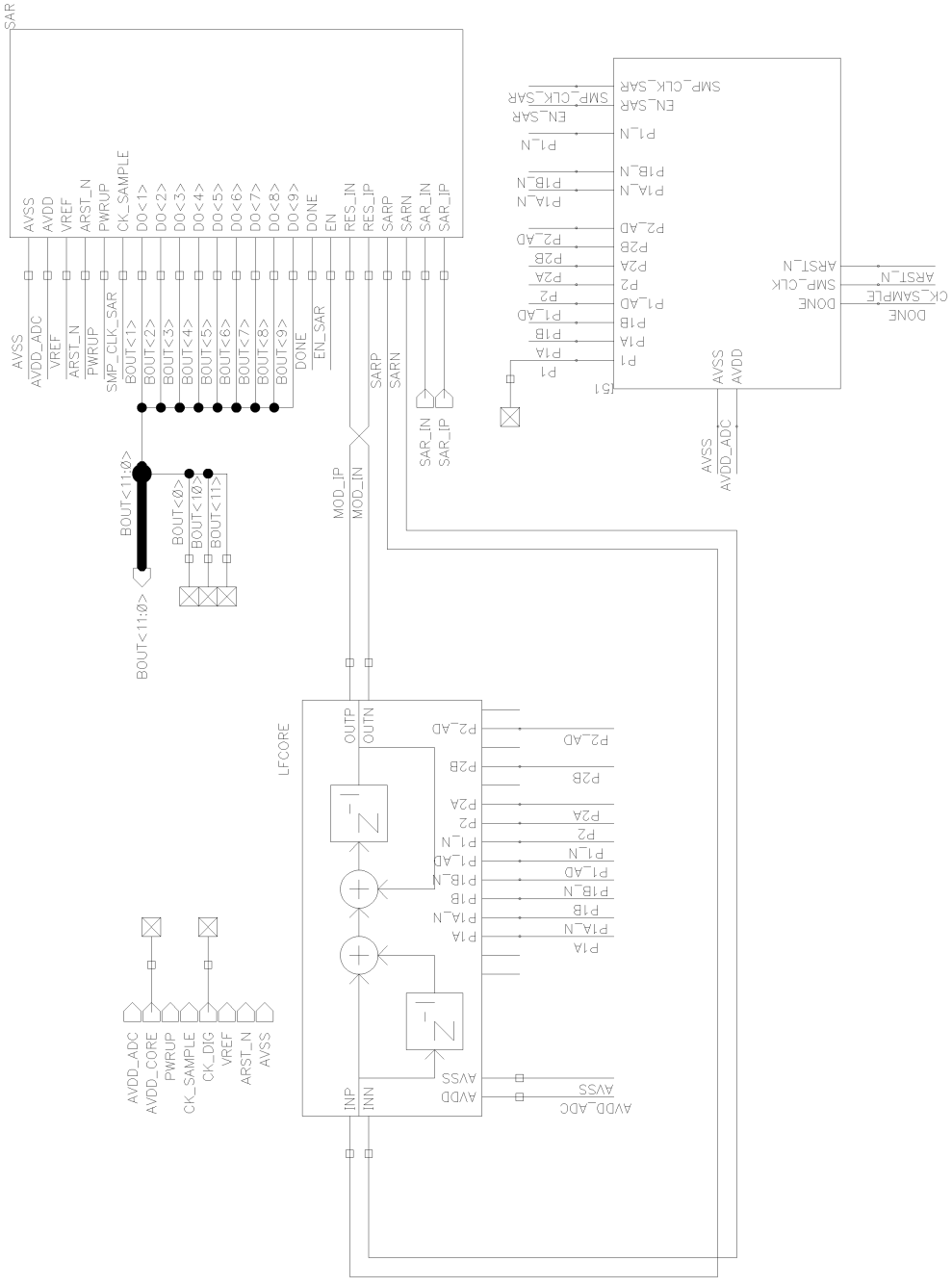
Schematics

In this appendix, all design schematics as well as the test-bench schematic used for the final verification are given as they appear in the Cadence Virtuoso software.



100

Figure C.1: Test-bench used for the final verification



101

Figure C.2: Top level schematic for the NS-SAR.

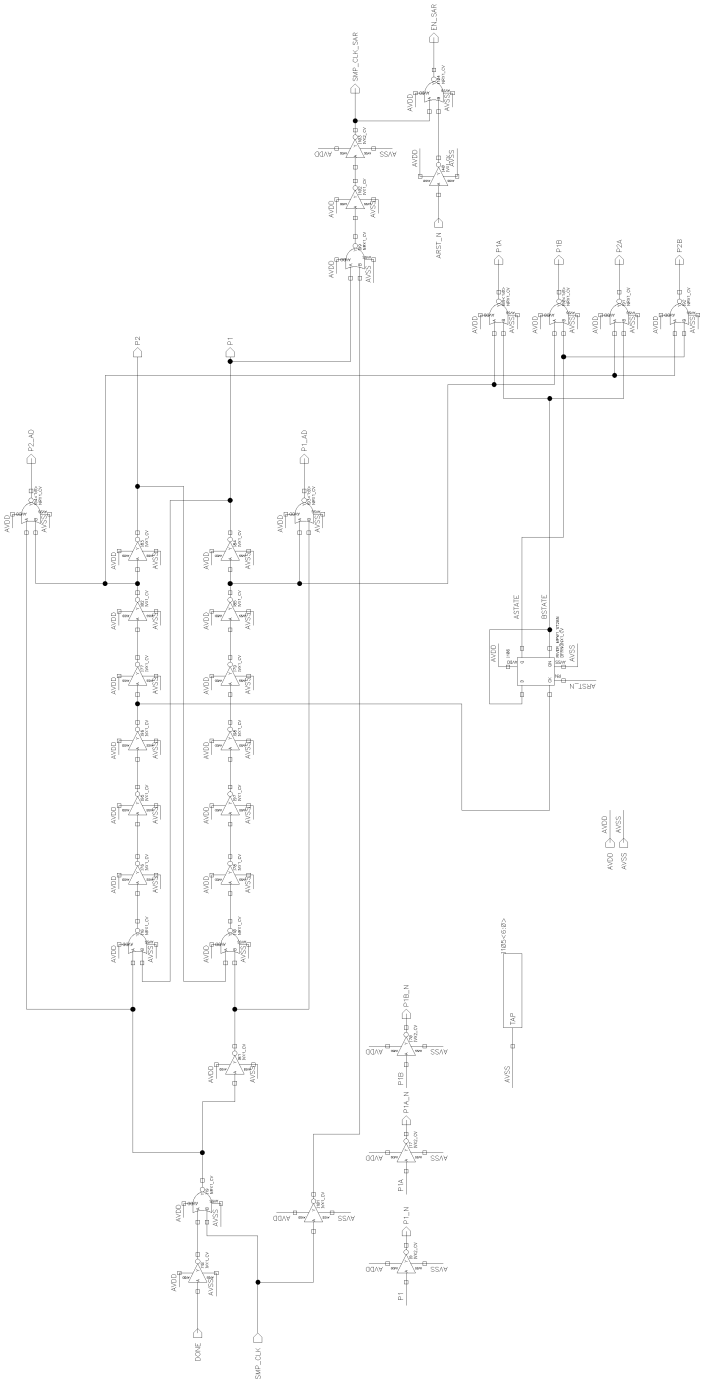


Figure C.3: Non-overlapping clock generator.

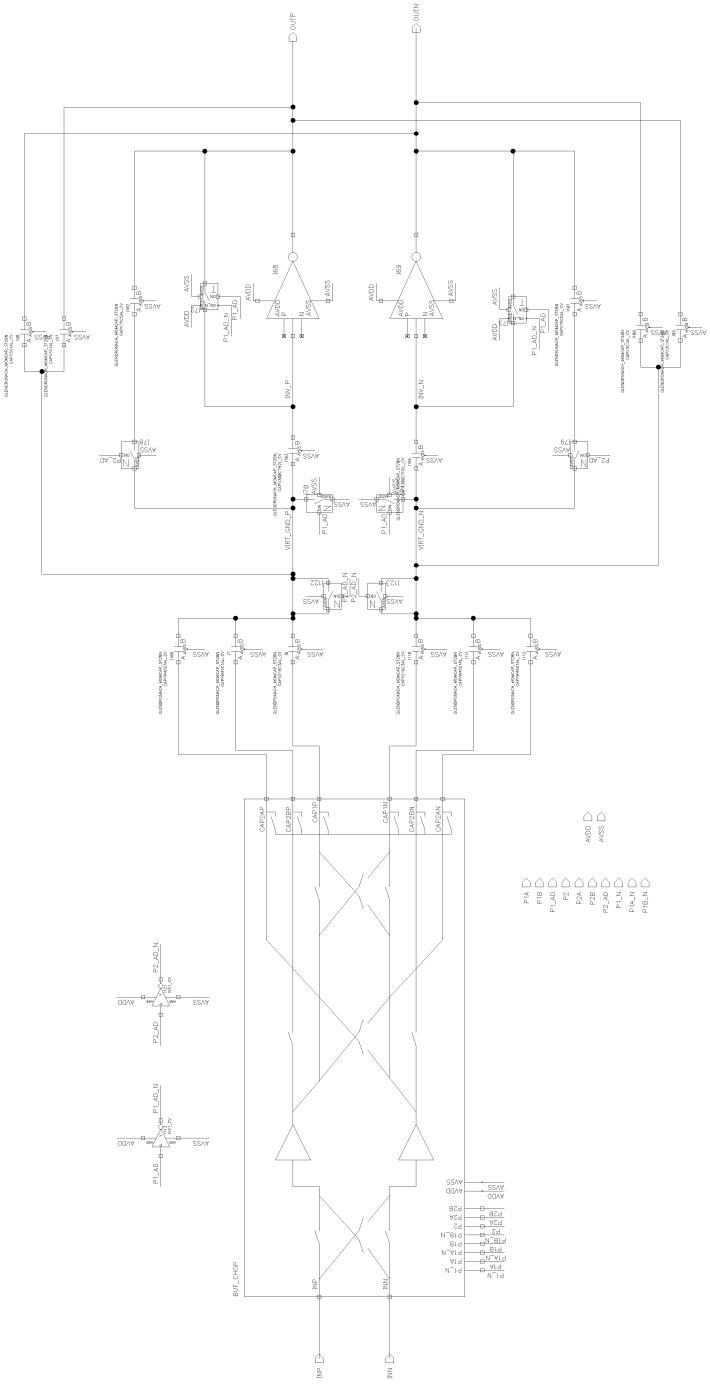


Figure C.4: Loop filter.

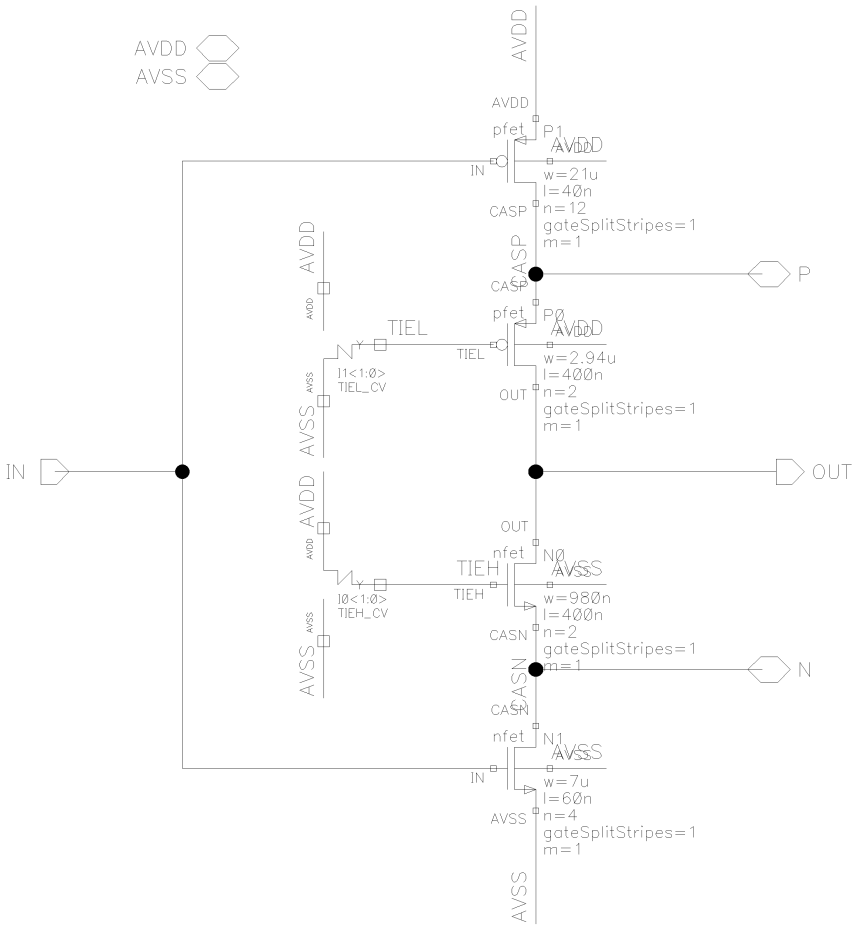


Figure C.5: Inverter.

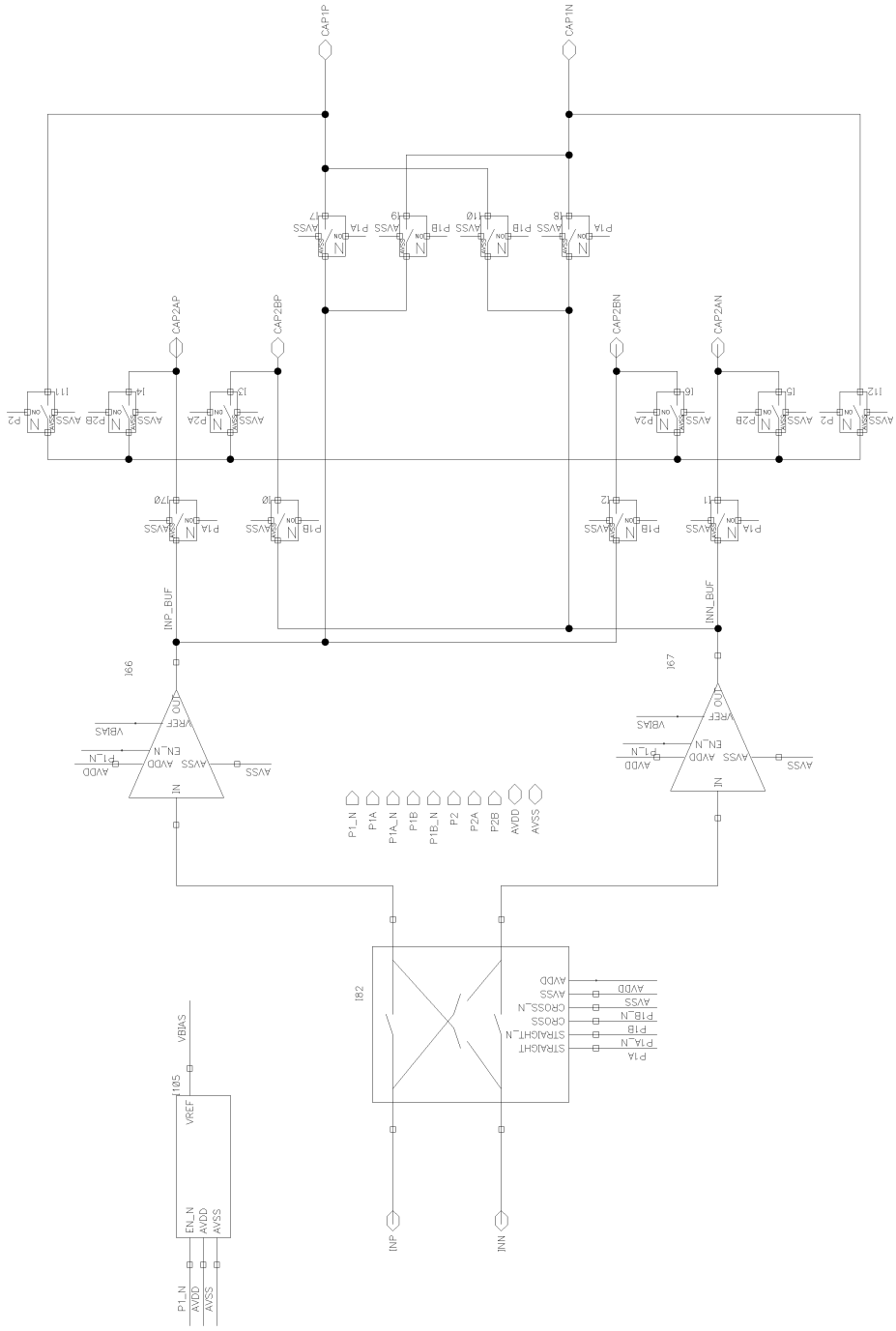


Figure C.6: Chopped buffers and related switches.

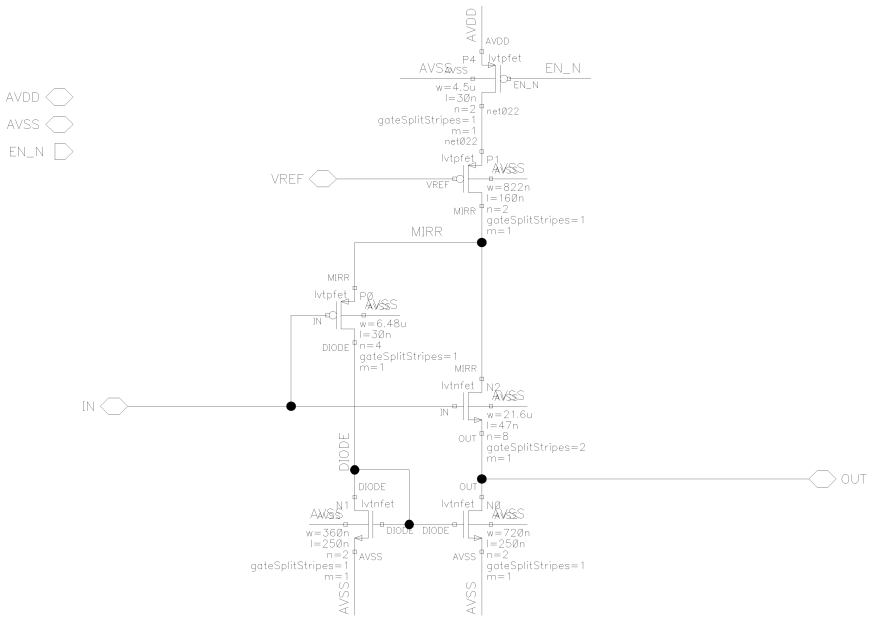


Figure C.7: Buffer.

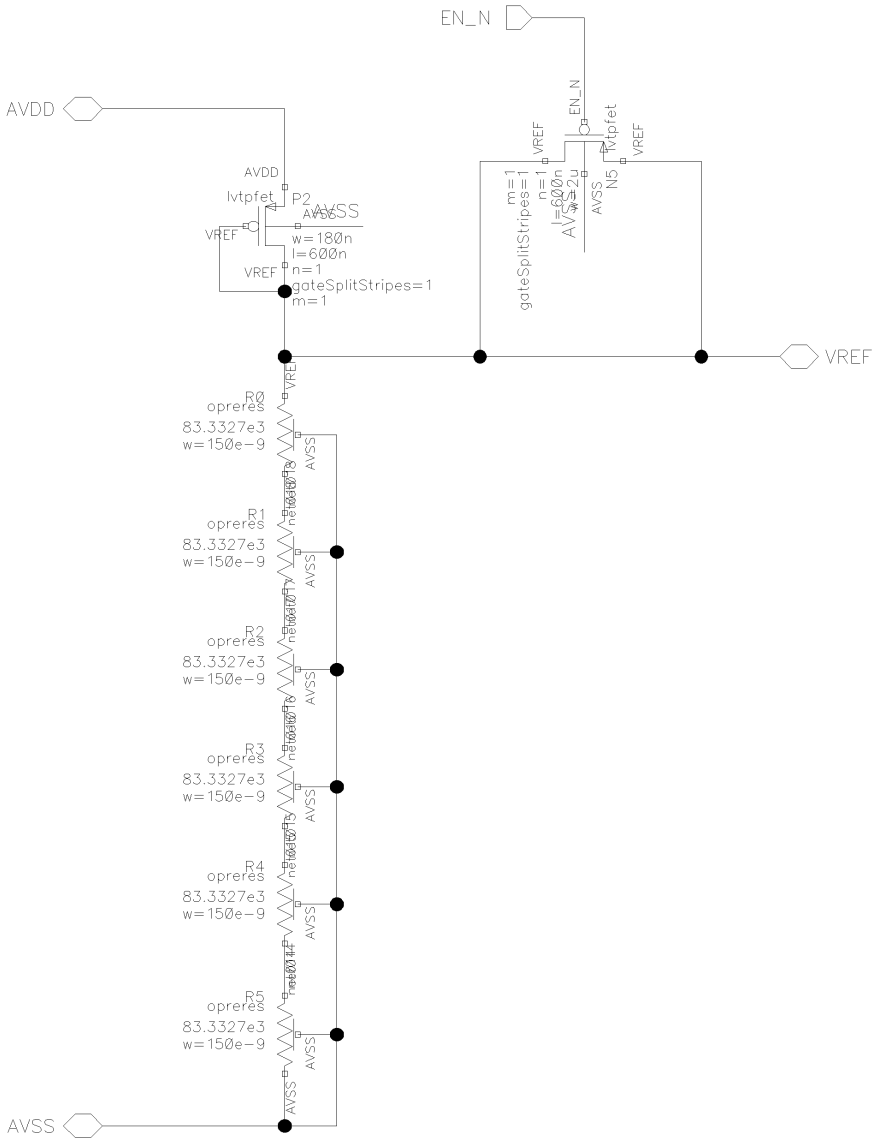


Figure C.8: Buffer bias circuit.

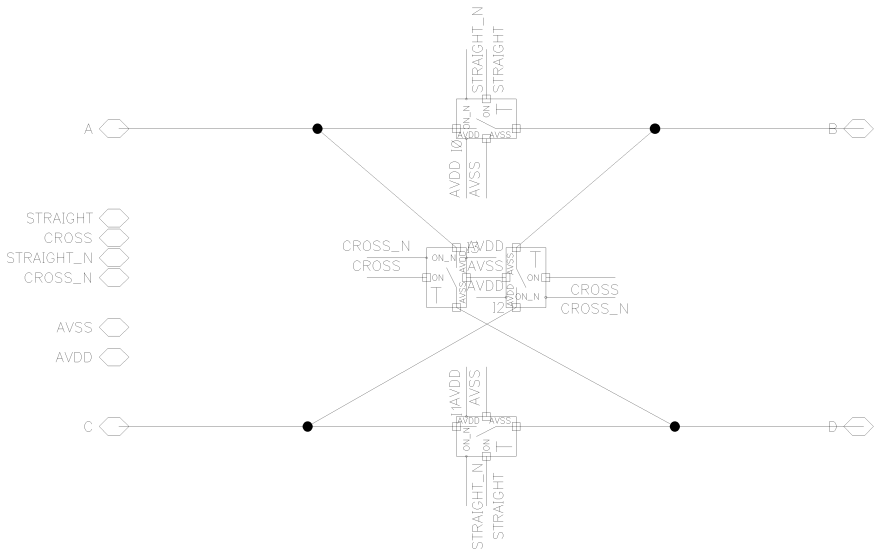


Figure C.9: Butterfly switch used in front of the buffers.

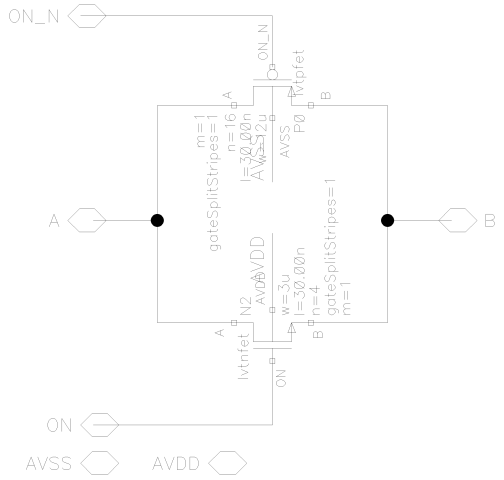


Figure C.10: Transmission gate used in the butterfly switch in front of the buffers.

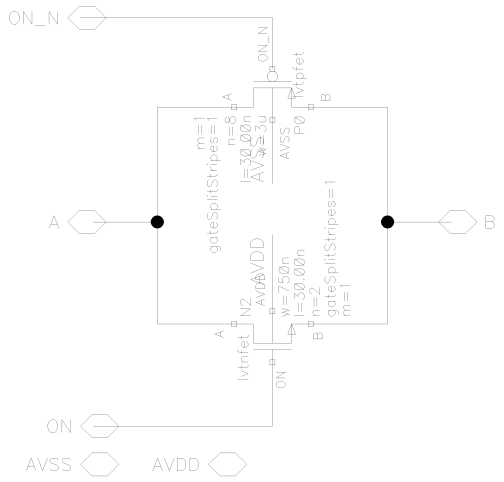


Figure C.11: Transmission gate used as inverter reset switch.

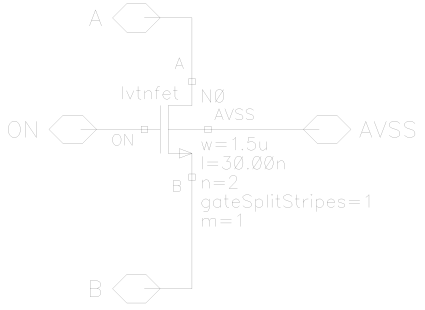


Figure C.12: NMOS switch used everywhere in the loop filter where transmission gates are not used.