Stochastic Resonance Analog-to-Digital Conversion

1-Bit Signal Acquisition Employing Noise

D.P.N. Mul

Technische Universiteit Delft

TU Delft Delft University of Technology

Challenge the future

Stochastic Resonance Analog-to-Digital Conversion

1-Bit Signal Acquisition Employing Noise

by

D.P.N. Mul

in partial fulfillment of the requirements for the degree of

Master of Science
in Electrical Engineering

at the Delft University of Technology,
to be defended publicly on Wednesday August 22, 2018 at 1:30 PM.

Supervisor:	Prof. dr. ir. W.A. Serdijn	
Thesis committee:	Dr. ir. J. Weber,	TU Delft
	Dr. M. Alavi,	TU Delft
	Dr. O.C. Akgun,	TU Delft

An electronic version of this thesis is available at http://repository.tudelft.nl/.

TU Delft Delft University of Technology

"I wanted to make noise, not study theory"

– James Hetfield

Abstract

Stochastic resonance (SR) is a phenomenon in which the presence of noise increases the performance of the system. The phenomenon has first been discovered in a climate change model and is later observed in neuronal systems. In artificial, electronic, systems, stochastic resonance is observed in systems based on Schmitt-Triggers and comparators. The common property of all these systems is the threshold, which, when reached, causes a large transition in the system.

The presence of noise in small signals can cause the system to reach the threshold or can increase the number of state transitions, increasing the quality of the output signal. Oversampling and integration are applied to reconstruct the original signal.

Biomedical signals are typically affected relative high noise levels. The observations of stochastic resonance in nature, such as biological neural systems, combined with the observations of stochastic resonance in comparator-based circuits formed the inspiration and fundamental of this thesis research.

The goal of this project is to investigate the potentials for using stochastic resonance in biomedical signal acquisition. In this thesis, an explorative study on the behavior, the performance, and the design of an analog-to-digital (ADC) converter fully based on stochastic resonance in a 1-bit quantizer is presented. The design and application focus on processing ECG measurements.

A comprehensive analysis of the behavior, and an analytical method to determine the performance of 1-bit stochastic resonance analog-to-digital conversion in a comparator based circuit is presented. A novel technique using a negative hysteresis is found, showing a potential increase in SNDR up to 6.4 dB.

Based on this analysis, a system level design is presented which implements a closed-loop operation of the stochastic resonance ADC. This design comprises a feedback loop to control the noise level, realizing the maximum performance over an input amplitude range from 1-10 mV, independent from noise present in the system.

Furthermore, an offset compensation scheme is presented, which controls the threshold of the comparator, and a digital multi-rate filter is implemented to filter the high-frequency noise, and to apply downsampling the highspeed bitstream.

IC implementation of the comparator and the noise source is studied. A low-offset comparator, based on a strongARM latch, is proposed, with an offset calibration technique, reducing the offset to below 50 μV. The noise source creates a flat noise spectrum from 20 kHz to 4 MHz, using amplified thermal noise of a resistor combined with the amplifier noise.

The proposed system can deliver a 27 dB SNDR in a signal bandwidth of 216 Hz, and an input amplitude range of 10×, using a sampling frequency of 2 MHz.

Acknowledgements

Here I would like to show my gratitude to the people who have helped me during my thesis project, and the road leading to this point. First of all, I want to thank Wouter Serdijn, who has been an inspiration to me as a teacher during my Bachelor and my Master. I am thankful for the opportunity to perform my master thesis research under his supervision, and for the great project, I have been able to do. I want to thank him for the freedom I had to give shape to the project. It allowed me to combine my interest in electronics with stochastic processes and signal processing in a biomedical environment.

I would like to thank everybody in the Bioelectronics group. It has been an inspiring environment to work in. I have always appreciated the interest in each other's work, the preparedness to help each other and to share the knowledge. My thanks go out to Can, for the interesting discussions. Your open-minded way of finding solutions has been inspiring and very helpful. Furthermore, I would like to thank Samprajani for her help and the time she took to look at my designs, Ali for his technical support throughout the project, and Marion her support in the administrative work.

Finally, I want to show my gratitude to my family and friends. The road has not always been easy, and I want to sincerely thank everybody who supported me on the way. I want to thank my fiancee, Anna Bernard, for her unconditional love and support. My parents, brother, and sister, who have always supported and believed in me. I would not have reached this point without your support.

Dieuwert Mul

Delft, The Netherlands

30 July 2018

Contents

1

Introduction

In signal acquisition and analog-to-digital conversion, noise is typically avoided as much as possible. Traditionally analog circuits are designed for a highly linear, deterministic behavior. Noise will, in this case, always have a negative influence on the resolution of a signal.

When non-linear systems are considered, noise can contribute to signal detection. This phenomenon is called stochastic resonance. It is observed in several natural systems, among which climate change, but also neuronal signals and its detection in the brain seem to benefit from this phenomenon. Stochastic resonance is also observed in electronic systems based on comparators and Schmitt-triggers.

This work is an explorative study on using stochastic resonance in analog-to-digital conversion for biomedical signal acquisition. The goal is to design an ADC fully based on stochastic resonance.

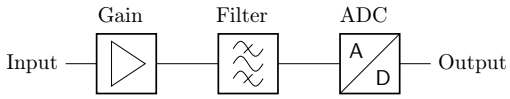


Figure 1.1.1: Typical analog front-end for signal detection

1.1 From Current Solutions to Bio-inspired Design

Current analog front-ends of sensing systems typically consist of three building blocks, each of which has a specific function. A block diagram of a typical analog front-end is shown in Figure 1.1.1. The signal sensed by the electrode first gets amplified, secondly, frequency filtering is applied to select the frequency bands of interest. Finally, analog-to-digital conversion is applied.

This approach has the advantage of orthogonal design. Each block can be designed and optimized individually, for each specific function. Although all blocks are optimized, separation of all function in different blocks does not fundamentally lead to the best or most power-efficient solution.

The electrochemical systems in the human body, on the other hand, cannot be decomposed into well-defined orthogonal building blocks. The system is a result of evolution rather than engineering, and breaking down the system in different building blocks with a specific functionality would be extremely hard, if not impossible. A more accurate model would be one which consists of a vast amount of interconnected blocks which all work together to fulfill one, or more, functions.

The efficiency and computational power of the human brain is, as well as its complexity, on many levels superior to engineered systems. Although biological systems are built up and behave fundamentally different, engineered systems can benefit from bio-inspired design.

A Stochastic resonance ADC (SR-ADC) is an interesting example of incorporating a bio-inspired phenomenon in electronic design. The phenomenon is first discovered in natural systems[1] and later observed in comparator-based electronic circuits. Biomedical systems often contain high noise levels, and stochastic resonance is observed in many biological neural systems.

The nervous system, using stochastic resonance, is a highly power efficient system, reaching efficiencies which are not met by electronic circuits. By using stochastic resonance in an electronic solution to sense biomedical signals, the efficiency of electronic signal acquisition can potentially be increased.

1.2 Stochastic Resonance Analog-to-Digital Conversion

As already mentioned above, stochastic resonance is a phenomenon in which the performance of a system is improved in the presence of noise. It is observed in systems that show a large change in the output when a certain threshold is reached. Electronic circuits such as a comparator or a Schmitt-trigger can be used in a stochastic resonance system[2]. Since the output of these threshold-based systems is binary, using stochastic resonance, the signal can be directly converted into a digital signal.

A basic stochastic resonance implementation is shown in Figure 1.2.1. A signal x is summed with noise η. The noisy signal is applied to a comparator with some threshold θ. In the case of a Schmitt-trigger, this threshold depends on the state of the system. The

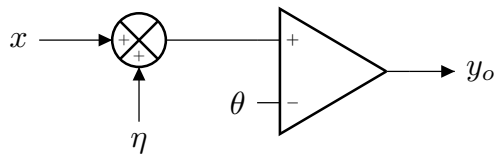


Figure 1.2.1: Basic stochastic resonance analog-to-digital conversion block diagram.

system creates the 1-bit quantized output bitstream y_o. In this way, a very small noisy signal can be directly converted into a binary signal.

Quite some literature can be found describing the phenomenon, and some electrical circuits which use stochastic resonance have been presented. However, a completely integrated stochastic resonance amplifier and an analog-to-digital converter is, to the knowledge of the author, not yet presented.

1.3 ECG Measurement

The SR-ADC is designed electrocardiograph (ECG) signal acquisition. An ECG records the electrical activity of the heart using electrodes placed on the skin. Typically the peak-to-peak amplitude of the signal from the electrodes is 1-10 mV. The bandwidth of an ECG signal recording is typically around 0.5-200 Hz. Based on the guidelines of the IEC (International Electrotechnical Commission), and the specifications used in literature, the SR-ADC is designed for the above-mentioned amplitude and bandwidth [3, 4].

This application is chosen for several reasons. First of all, the maximum signal frequency is low. The SR-ADC requires a high oversampling ratio, which will limit the signal bandwidth. Secondly, the ECG signal is relatively easy to measure and well-described. This, for example, in contrast to EEG signals which often needs more knowledge and experience to read and understand the data.

The ECG signal should be considered a sample application of the SR-ADC. It is used to create a framework for the design in this thesis project. The goal of the thesis is to explore the potential of stochastic resonance in analog-to-digital conversion, rather than to address an issue regarding ECG measurements, or to compete with the current state-of-the-art solutions for ECG recording.

For this reason, not all requirements regarding ECG measurement are taken into account in each step of developing the SR-ADC. At the early stage which the research of stochastic resonance analog-to-digital conversion is in, it is believed that studying the fundamental potentials, and developing a methodology to design a SR-ADC is of more importance than designing for a specific application.

1.4 Goal

The goal of this thesis is to proof the concept of the SR-ADC and to propose a design towards implementation of an ADC fully based on stochastic resonance. It is an explorative research project which aims to determine potentials of using stochastic resonance in analog-to-digital conversion. The study is divided into 3 subgoals, which are treated separately.

The first goal is to determine the behavior and to analyze the performance of a SR-ADC. To achieve this goal, several questions are formulated:

- What types of stochastic resonance occur, and how can they be translated into an analog-to-digital converter based on 1-bit quantizing?
- Which stochastic resonance mechanism shows the most promising behavior for implementation an SR-ADC?
- How is the behavior, and the performance of stochastic resonance in 1-bit quantized systems mathematically described?
- What are the degrees of freedom in designing a SR-ADC, and how does the design parameters affect the performance?
- What performance can be achieved in a SR-ADC, and what is the optimal noise level in the system?

Several publications are found in which stochastic resonance is used to perform analog-to-digital conversion. However, a closed-loop system in which the system is controlled such that operation at the optimal noise level is assured has not been presented yet. The second goal is therefore to develop a system level design, which enables implementation of the SR-ADC in a closed-loop system, whereby the system is controlled towards its stochastic resonance peak. In this design, the two main questions that need to be answered are:

- How can the noise level be controlled such that system always operate at the optimal noise level, independent of the input signal amplitude?
- How can the threshold level be controlled such that the desired threshold is always maintained?

The third goal is to develop an electronic implementation of the SR-ADC and to address the challenges involving the design. The main questions to be answered in the implementation study are:

- Is it feasible to implement the SR-ADC such that (precise) pre-amplification of the signal can be avoided?
- How can the unique properties of an SR-ADC, such as the beneficial influence of the noise, be employed to improve the (power) efficiency of the system?

1.5 Thesis Outline

To accomplish these goals, this thesis first gives background information on stochastic resonance in Chapter 2. Based on the different observations of SR in literature, a classification of stochastic resonance is presented in Chapter 3.

While designing an electrical stochastic resonance circuit is still a largely unexplored field, the behavior of a stochastic resonance analog-to-digital converter is examined, and analytically described, in Chapter 4. The chapter comprises a comprehensive analysis of analog-to-digital conversion using stochastic resonance. An analytical method is proposed to examine and determine the performance of the SR-ADC, and a novel technique to enhance the performance is presented.

In Chapter 5, the system level design is presented. A digital multi-rate low-pass filter is presented which filters the high-frequency noise and downsamples the comparator output to the Nyquist rate. Furthermore, two control loops are proposed to compensate the threshold and to adjust the noise level to the optimal operating point.

An IC-implementation of the comparator and the noise source are presented in Chapter 6. Both blocks are simulated and the behavior in the SR-ADC is derived.

Finally, the conclusions and the recommendations for future work are presented in Chapter 7 and 8 respectively.

2

Background Information on Stochastic Resonance

As already stated in the introduction, stochastic resonance (SR) is a phenomenon in which noise has a beneficial influence on the performance of the system. The phenomenon was first described in 1982 and has been a research topic for several decades. However, stochastic resonance is mainly investigated as a phenomenon which occurs in natural systems and is thereby relatively unknown in the field of engineering.

Through the years, stochastic resonance has been observed in a wider range of systems. The first observation occurred in a climate change model, which used a double-well model [5]. Research in SR has long been limited to systems with a hysteresis. Not until 1995, SR was also observed in excitable systems, which lead to a broader definition of the phenomenon [6, 7]. Until 2001 SR was described as a phenomenon occurring in subthreshold systems. A beneficial effect of noise was shown in suprathreshold systems, therewith a new class of SR, suprathreshold SR (SSR), was introduced in [8, 9].

This chapter follows the historical expansion in the field of SR. Observations in hysteresis systems are presented in Section 2.1. SR in excitable systems is discussed in Section 2.2 and suprathreshold SR is introduced in section 2.3. Finally, Section 2.4 comprises a brief comment on the definition of SR.

2.1 Stochastic Resonance in Hysteresis Systems

The first publication which mentions stochastic resonance is a paper by Benzi et.al. in 1982[5]. In this paper, stochastic resonance is used to explain a strong fluctuation of the earth's temperature with a period of 10^5 years. The model discussed in the paper is based on a double well system as shown in Figure 2.1.1. Without any disturbance, the system will stay in one of the two wells and the output only fluctuates over a small temperature range. However, by random external fluctuations, the system is pushed away from the local minimum. The distance between a stable point and an unstable point changes periodically. When the distance between the stable point and the unstable point is small, random fluctuations can push the system into the other state. This results in a large, periodic temperature shift, induced by a relatively small input signal. The output, created by a computer simulation, is shown in Figure 2.1.2. The plot shows the strong periodic temperature shift discussed above.

An electronic implementation is proposed in [2], where the phenomenon is reproduced using a Schmitt-trigger. Although the behavior of a Schmitt-trigger is not exactly equal to a double-well system, the mechanism and simulation results correspond to the double-well system. The main results of this paper are shown in Figure 2.1.3. The power spectral density plots on the left (2.1.3a) show strong enhancement of the input signal frequency compared to the noise values, for a higher input noise variance (0.5 V_{rms} compared to 0.3 V_{rms}).

The plot on the right (2.1.3b). shows an optimum for the amount of noise added to the system. These peaks are often referred to as the 'stochastic resonance peak'.

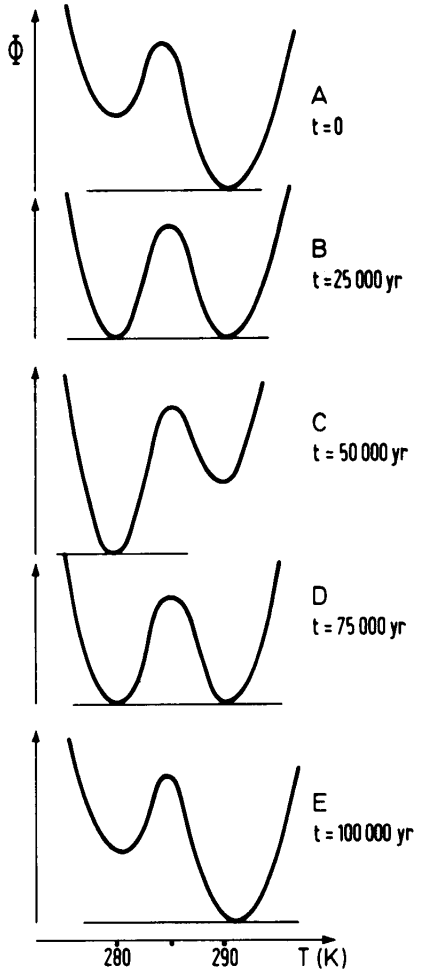


Figure 2.1.1: Bi-stable climate model used by Benzi e.a. (1982)[5]

It should be noted that the use of the term 'resonance' is not trivial. The use of the term originates from the notion that a small input force leads to a large signal at the output that contains a single frequency. The first observation occurred in a periodic system, which led to some misinterpretation of the term resonance. Discussion on the use of the term resonance arose already in correspondence regarding [5] before it was published ([10, 11]). In [2], it is shown that the signal-to-noise ratio is maximized for a certain frequency. However, in contradiction to conventional resonance, stochastic resonance is not driven by a natural frequency of the system but is driven by (wideband) noise. The frequency at which the resonance occurs is not defined by the system itself, but by the frequency of the (small) input signal.

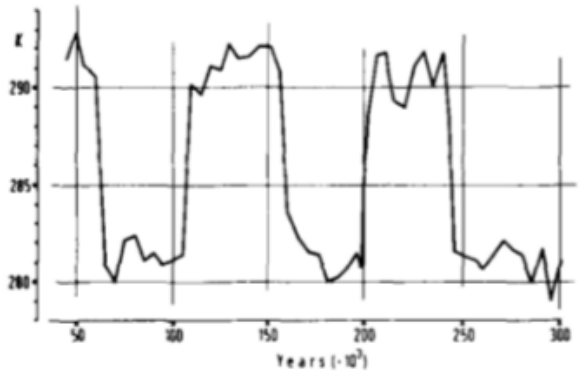
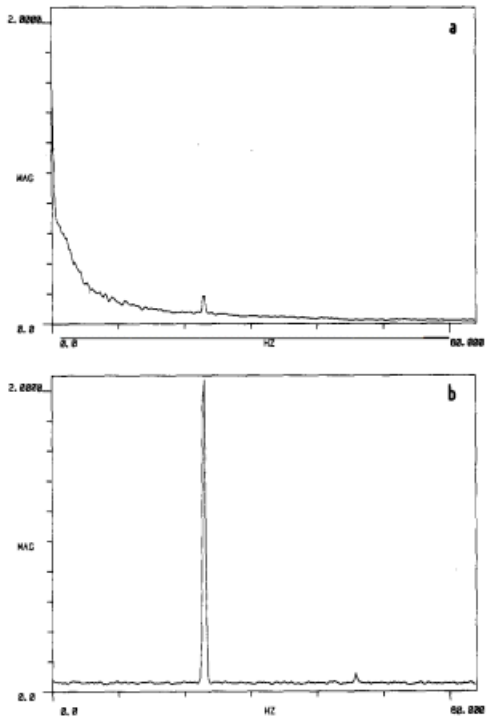
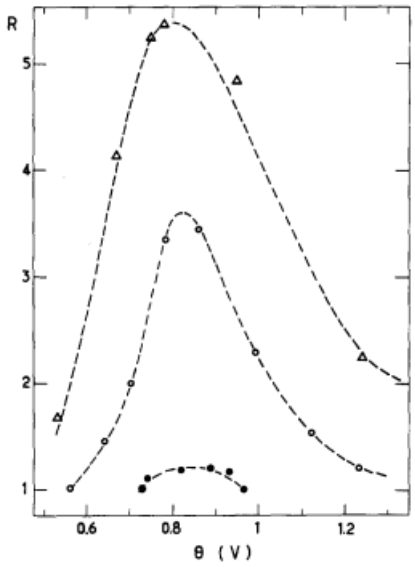


Figure 2.1.2: Output of a computer simulation of the model presented in [5]. The plot shows large periodic temperature shifts on top of smaller, more noise-like variations. Figure is taken from [5].



(a) Frequency power spectra of the output voltage measured in [2]. The input signal has an amplitude of 0.5 V at a frequency of 23 Hz. The noise variance is 0.3 V_{RMS} above and 0.5 V_{RMS} below. The lower plot shows suppression of noise and strong enhancement of the signal frequency for a non-zero noise level.



(b) Signal-to-noise ratio, in this plot indicated with R, measured in [2]. The SNR is plotted as a function of the noise variance θ. The SNR is plotted for input signal amplitudes $V = 0.05$ (•), $V = 0.25$ (○) and $V = 0.5$ (△). Around $\theta = 0.8$, the stochastic resonance peak is shown.

Figure 2.1.3: Results presented in [2]. Stochastic resonance is observed in the power spectral density (PSD) and in SNR plots. Both figures are taken from [2].

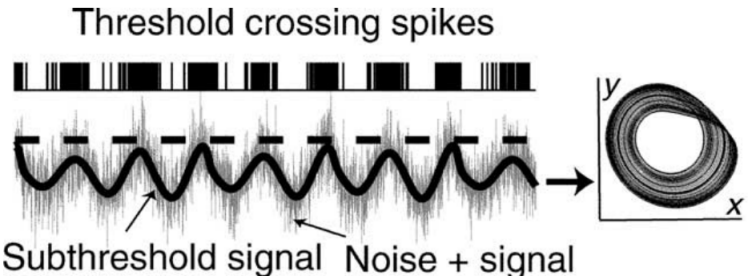


Figure 2.2.1: Illustration of SR in an excitable system. The noiseless input signal does not reach the threshold. Due to added noise, the threshold is reached and the signal is detected. The x-y-plot in the right represents the stimulus, which was generated from a Rössler system. Figure is taken from [17]

The first neuronal models based on SR, presented in 1991 [12, 13, 14], led to a large increase in interest in stochastic resonance. The interest increased, when the occurrence of stochastic resonance in physiological neural systems was shown in [15, 16] in 1993 and 1996 respectively.

2.2 Stochastic Resonance in Excitable Systems

Further research in stochastic resonance in neuronal models led to the observation of noise enhancement in excitable systems. Stochastic resonance in aperiodic excitable systems was first observed in [7], and stochastic resonance has been shown in an integrate-and-fire neuronal model in [6]. A great part of the literature treating stochastic resonance has since this period been in the field of neurobiology. Several neur(on)al models and experiments have shown that the brain highly relies on stochastic resonance [17, 18]. In contrast to the first models used in stochastic resonance, excitable systems will return to the first state after excitement, if no further input is applied. The systems mentioned above can roughly be modeled by a simple comparator. An example of SR in an excitable system is shown in Figure 2.2.1.

2.3 Suprathreshold Stochastic Resonance

The work discussed above only considered subthreshold systems, i.e. no signal appears without noise added to the system. The beneficial effect of SR in suprathreshold signals has been shown in [8]. In this paper, an array of comparators is used, as shown in Figure 2.3.1. The signal at the input of each comparator is the same, while each comparator is subjected to an independent noise source. The output of all comparators is summed. The summed output contains information about the amplitude of the signal. If the signal is higher, the output has a higher probability to be positive. Thereby, the expected value of the summation of multiple samples will be higher. A lower signal will lead to a lower expected output value. Furthermore, the paper shows that the ideal threshold value is at the DC-level of the signal.

The beneficial effect of noise in suprathreshold signals in a neuronal array model is shown in [9]. This paper confirms that the system performance is optimal for a threshold which is set to the DC-level of the signal. The distance between the signal level and the threshold is minimized and the threshold crossings are maximized.

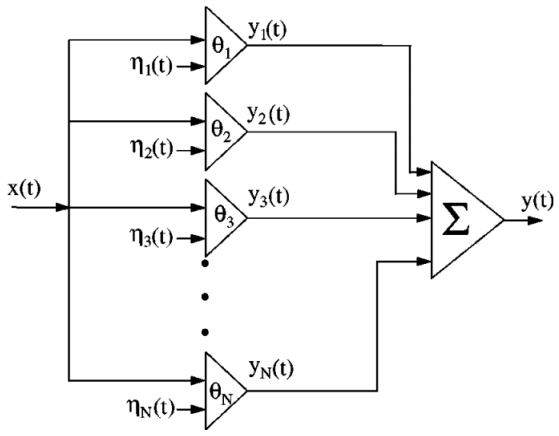


Figure 2.3.1: Suprathreshold stochastic resonance system which uses N comparators and sums the output of all comparators. Taken from [8].

In [19], an extensive research on stochastic resonance in 1-bit signal processing is presented. An analog-to-digital converter based on stochastic resonance is proposed. This SR-ADC is based on the same system as shown in Figure 2.3.1. As an alternative to multiple comparators and summing of the output, integration in time is also examined in [19]. This system will be discussed in detail in Chapter 4.

2.4 Definition

The definition of stochastic resonance has shifted significantly over the years. The phenomenon was first discovered for systems in which the (noiseless) signal cannot reach the threshold, and the term was only applied to periodic signals.

The extension of research in stochastic resonance to excitable systems and the introduction of aperiodic SR lead to more unclarity regarding the use of the term resonance. The resonance peak in SR is defined by an output performance peak for a given noise value, rather than a peak in the power spectral density for a given frequency, as shown in Figure 2.4.1.

Further expansion of stochastic resonance to non-linear systems in which the input signal can reach the threshold to switch from one state to another resulted in an even broader definition of SR. Careful notion of the used definition should be taken. There is not one, unambiguous, definition of stochastic resonance, and the narrower, original criteria are still in use [18]. Current literature often proposes to use a broad definition of the term stochastic resonance, which includes all systems in which noise has a beneficial influence on the performance [18, 20, 19, 21]. This does not take into account the underlying mechanism. According to this definition, e.g. dithering can be considered a technique which uses stochastic resonance [21]. This broad definition of stochastic resonance is used in this thesis.

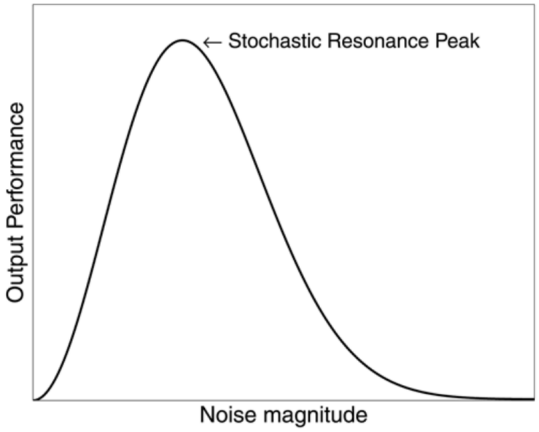


Figure 2.4.1: Qualitative illustration of the stochastic resonance peak. The resonance is not defined for a specific natural frequency, but for a specific noise magnitude. Where conventional resonance shows a peak in the power density, stochastic resonance shows a performance peak at a certain noise level. The output performance can be defined using different measures such as SNDR, mutual information and correlation. Figure is taken from [18].

2.5 Conclusions

In this chapter, an overview of previous studies in stochastic resonance is presented. While stochastic resonance is mostly studied as a phenomenon which occurs in nature, in recent publications it is more and more considered as a phenomenon which can be used in engineering. This thesis project builds further on these recent publications. Hereby, especially the observations of stochastic resonance in a 1-bit quantized system are of interest, since the small input signal is directly converted into a digital signal. Meanwhile, the basic building block to implement this function is a simple comparator.

3

Classification

In the previous chapter, several systems that use SR found in literature have been discussed. This chapter focuses on a classification of SR. It should be considered a study to extract and classify fundamental SR behavior from an engineering perspective. This differs from most of the literature, which mainly examines and describes the stochastic resonance phenomena as they are found in nature.

The goal of this chapter is to examine and classify different SR phenomena, in order to determine the fundamental degrees of freedom for engineering SR-based systems. The qualitative analysis presented in this chapter is used to determine the most promising stochastic resonance mechanism to implement the SR-ADC.

First, the stochastic resonance definition used in this thesis is formulated, and the fundamental criteria of the system are defined in Section 3.1. Secondly, a general transfer of a stochastic resonance system is derived in Section 3.2. Based on this generalized SR transfer, different SR phenomena are classified in Section 3.3, 3.4, 3.5, and 3.6. The chapter is concluded and the design space of this project is determined in Section 3.7.

3.1 Definition and Criteria

Since there is no single unambiguous definition of Stochastic Resonance, before classification, the definition used in this work is determined. Based on this definition, the main criteria for SR are defined.

Current literature proposes to use a broad definition of SR. This definition will be used in this thesis project since a narrower definition will unnecessarily limit the design space.

The definition is formulated as follows [18, 20, 19, 21]:

Stochastic resonance is the phenomenon where the performance of a system is optimal for a non-zero noise level.

Based on this definition and the literature study in Chapter 2, three fundamental criteria of a SR-system are derived:

1. The input-output relation of the system is non-linear
2. The system has two or more (locally stable) states
3. The input contains, besides the signal, random noise

Non-Linearity

The non-linearity criterion can be illustrated by considering the linear superposition principle, as shown in Equation 3.1.1.

$$\begin{aligned} F(x+\eta) &= F(x) + F(\eta) \\ F(\alpha\eta) &= \alpha F(\eta) \end{aligned} \tag{3.1.1}$$

The output of the signal input (x) can be calculated independently from the noise input (η). Since both are independent, increasing the noise does not alter the signal transfer. The signal transfer is equal to the noise transfer, and thereby, the signal-to-noise ratio can not be increased in a linear system.

Multiple states

The second criterion is more specific than the first. In all systems in which stochastic resonance is observed multiple states can be distinguished. For example the climate change model of Benzi, in which the states can be described by either ice-age or no ice-age, or a neuronal model which can be either excited or not. A threshold or unstable point should exist, which lead switching from one state to another. The output of the system should show a large change when the system switches state. Fluctuations in the output within one state should be relatively small. The output of the system should be dominated by the state in which the system is, not by output fluctuations within one state.

The transition between the two states should be sharp. A gradual transfer between the states results in output values between the output values determined by the state. The probability that the state fully determines the output hereby decreases.

Finally, literature mainly focuses on bi-stable systems and it is often seen as a requirement of the system. However, there is no falsification for stochastic resonance in systems with more stable states. The notion that stochastic resonance can occur in tri-stable systems is shown in [22].

Noise

Stochastic resonance is based on the principle that noise enables, or increases the number of threshold-crossings. When noise enables state switchings which would not occur at all without noise (subthreshold SR), SR enables detection of the signal. In suprathreshold systems, the noise increases the number of state switchings, and SR can enhance the accuracy of the system [20].

3.2 Generalized SR-system

To classify stochastic resonance, first, a generalized system is proposed. From this generalized system, the parameters which play a key role in SR are extracted. Based on different values for each the parameters, the classification is performed.

The generalized transfer is shown in Figure 3.2.1. The plot shows the input-output relation. The system has two states, A and B. The state of the system is dominant in determining the output value.

The switching thresholds of the system are determined by a hysteresis with magnitude H and static threshold θ. Finally, the gains within one state and in the transition between to states are defined as $\delta y_s/\delta x$ and $\delta y_t/\delta x$ respectively.

This model is simplified, and only fulfills the purpose of qualitative classification. The system is not designed to model the exact behavior, but to show the qualitative SR mechanism that can be observed. The classification is performed without considering influence $\delta y_s/\delta x$ and $\delta y_t/\delta x$. Adding these parameters would not change the SR mechanism, but would only add (piecewise) linear behavior to the system. This linear behavior cannot contribute to the stochastic resonance since the superposition principle holds for this behavior.

A great advantage of using this model is that its transfer is a 1-bit quantizer, and can thus be implemented to perform analog-to-digital conversion.

As shown in the previous chapter, the performance of a SR-system can, in some cases, be improved by integration of the output. Integration can be applied by parallelization or integration over time. For parallelization, multiple parallel systems are used, each with an independent noise source. All outputs are summed or averaged. Integration over time can be implemented by means of oversampling and a low-pass filter. In this classification, only integration over time is considered. The observations however also hold for parallelization.

To compare and evaluate the output performance, signal-to-noise ratio (SNR) and signal-to-noise-and-distortion ratio (SNDR) is used. The SNR is defined as the power in the fundamental, i.e. the first harmonic, divided by the total noise power. The higher harmonics are not taken into account. The SNDR does take the higher harmonics into account and is defined as the power in the fundamental divided by the total power of the noise and the higher harmonics combined, defined as the total error.

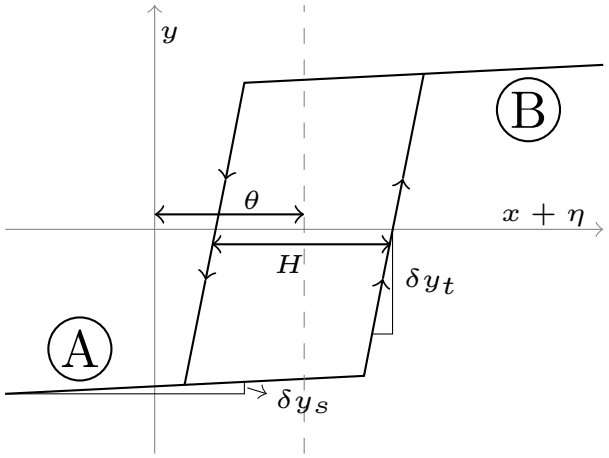


Figure 3.2.1: A generalized SR system transfer, with two states (A and B), a threshold θ, hysteresis H and the slopes in steady state Δy_s and in the transition Δy_t.

3.3 Subthreshold SR – Schmitt-trigger

In this model, shown in Figure 3.3.1, the hysteresis is implemented by Schmitt-trigger. Often a more extensive model, such as a double-well pseudo-potential is used. Although the implementation differs, the SR-mechanism is the same. The climate change system of Benzi ([5]) is an example of this class of SR-systems.

In subthreshold SR, the signal, without noise, never causes a state change, so $H/2 > |x|$. Noise added to the signal causes the input to reach the threshold. The largest probability of a state transition occurs when the signal x is the closest to the threshold. After a transition, the threshold changes due to the hysteresis. The distance between the signal and the threshold becomes large and the probability of a transition is small. This mechanism results in a periodic transition between the states.

Simulations of the system are shown in Figure 3.3.2. The system creates a square wave output. The exact moment of switching varies due to the stochastic properties of the input. Furthermore, a phase shift can be observed. The probability of a state transition increases when the signal approaches the threshold and is, therefore, more likely to occur when the signal approaches the threshold.

Frequency Behavior

The frequency spectra of the output are shown in Figure 3.3.3, for a noise variance of 0.5, 1 and 2. The output shows a low-pass behavior. As already described in [2], the cut-off frequency (f_{-3dB}) depends on the noise variance.

The largest output signal is achieved for $\sigma_\eta = 1$. Since the total output power is 1, this input noise value also gives the highest signal-to-noise ratio. In this case, the fundamental frequency is slightly higher than the cut-off frequency. At this point, the highest passband gain is achieved, while the signal is not affected by the attenuation in the stop-band, such as the case for $\sigma_\eta = 0.5$.

The optimal noise value is determined by the signal and noise at the output. The influence of harmonics is relatively small.

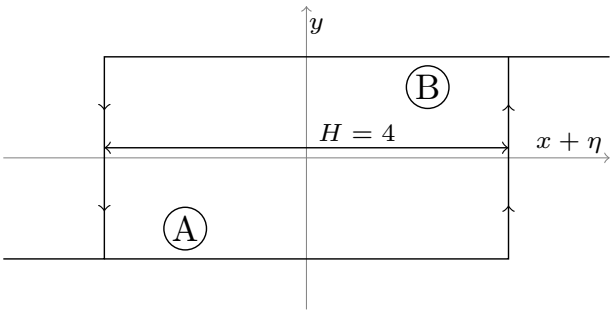
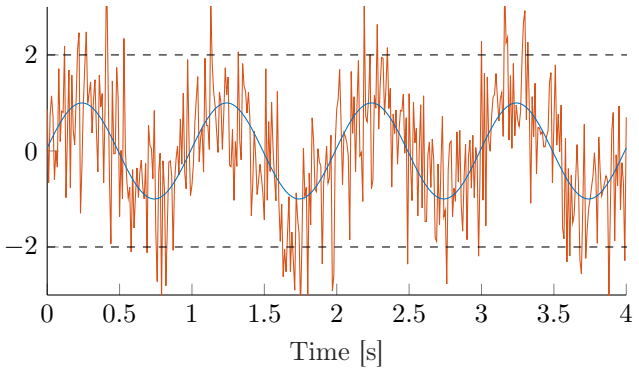
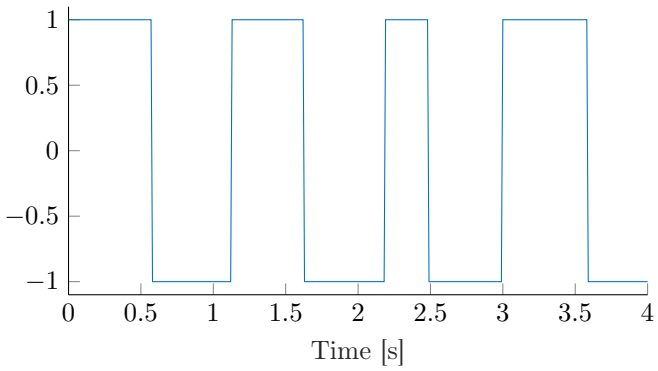


Figure 3.3.1: System transfer of dynamical subthreshold SR with Schmitt-trigger, $H = 4$



(a) Input signal, with and without noise



(b) Output

Figure 3.3.2: Simulations of the system. The input comprises a sine wave of the form $x = \sin(2\pi t)$ and white noise with standard deviation $\sigma = 1$. The hysteresis $H = 4$. The noise is simulated using 50 white noise samples per period.

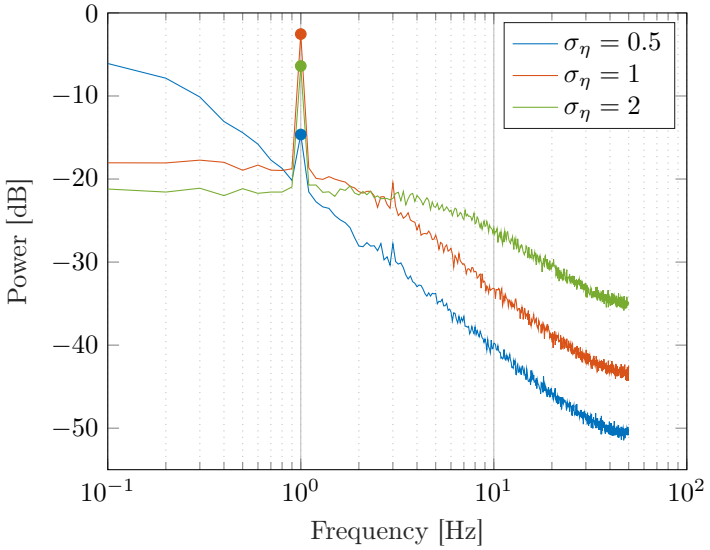


Figure 3.3.3: The power spectral density of the input and output of the simulated subthreshold, Schmitt-triggered system. The input comprises a sine wave of the form $x = \sin(2\pi t)$ and white noise with standard deviation $\sigma = 0.5$, $\sigma = 1$ and $\sigma = 2$. The noise bandwith is 50 Hz, and the hysteresis $H = 4$.

3.4 Subthreshold SR – Comparator

While in dynamical SR the threshold is determined by the hysteresis, in non-dynamical SR a static threshold θ is used. The signal is below the threshold, but additional noise can cause state transitions. Since the threshold does not change, the system will easily fall back to the 'resting' state. The mechanism is observed in excitable systems, such as excitable neuron models ([17]).

In Figure 3.4.1, the input-output relation of the system is shown. The model shows an excitable system, with a maximum signal input of 0.4 and a threshold $\theta = 0.5$. White noise, with $\sigma_\eta = 0.2$, is added to the signal.

Simulation results of the systems are shown in Figure 3.4.2. The output is only determined by the probability of the noisy input to be above the threshold. Since the system has no memory, periodicity of the input signal is no requirement. This class is therefore also referred to as non-periodic SR.

The output can be considered a pulse density modulated signal. The expected pulse density is determined by the probability of $x + \eta > \theta$. By integrating or averaging multiple samples, the pulse density can be computed. In the plot, this is achieved by averaging over 20 samples. The averaged output shows that the amplitude information is, to some extent, preserved.

Frequency Behavior

The power spectral density of the simulation, for noise levels of $\sigma_\eta = [0.1,\ 0.2,\ 0.5]$, is shown in Figure 3.4.3. In contradiction to the Schmitt-trigger, this system does not affect the noise spectrum at the output of the system. Since the system has no memory, and all noise samples at the input are independent, the output noise contribution is also independent,

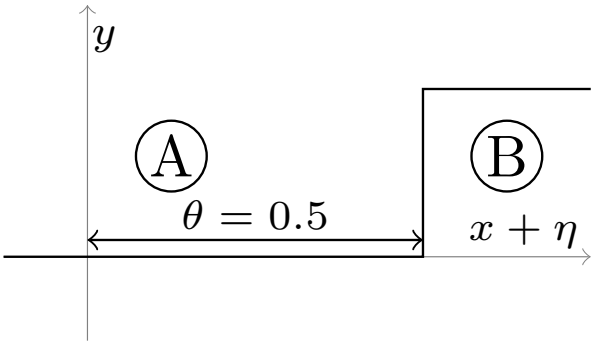


Figure 3.4.1: System transfer of subthreshold SR with comparator, $\theta = 0.5$

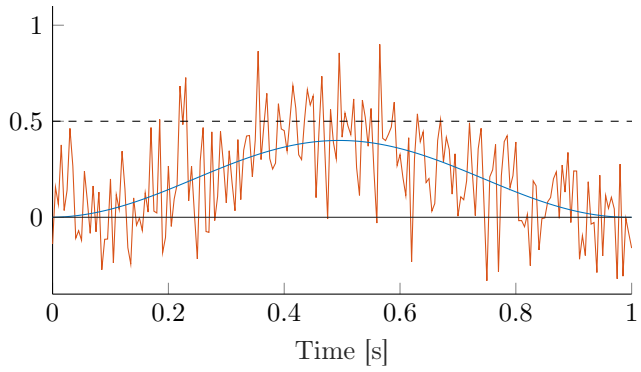
Noise	$\sigma_\eta = 0.1$	$\sigma_\eta = 0.2$	$\sigma_\eta = 0.5$
SNDR	$-12.7\ dB$	$-10.3\ dB$	$-16.0\ dB$
SNR	$-12.6\ dB$	$-10.2\ dB$	$-16.0\ dB$

Table 3.1: SNDR for different noise values in subthreshold SR without hysteresis

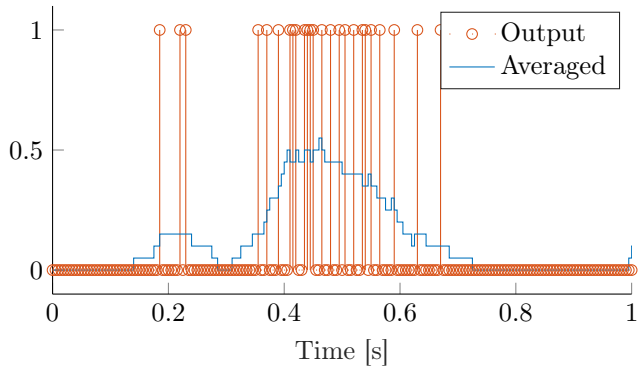
and thus white.

For low noise values, only a small output signal is detected. Increasing the noise to $\sigma_\eta = 0.2$ results in a higher output signal and more noise. However, the distance between the noise floor and the signal increased, thus the SNR is higher. When the input noise is further increased to $\sigma_\eta = 0.5$, the signal power decreases, while the noise power increases, thus the SNR decreases. The SNR of each noise value is shown in Table 3.1.

The input noise power also affects the power in the harmonics. For higher noise values, the (relative) power in the harmonics decrease. In Table 3.1 the SNDR is compared to the SNR. Since SNDR $\approx$ SNR, the effect of the harmonics on the performance is small.



(a) Input signal, with and without noise



(b) Output of the comparator, and the moving average of 20 output samples

Figure 3.4.2: Simulations of the comparator based subthreshold SR system. The input signal is: $x = 0.2\sin(2\pi t + \frac{1}{2}\pi) + 0.2$. White noise with standard deviation $\sigma = 0.2$ is added to the signal. The threshold $\theta = 0.5$. The noise is simulated using 200 white noise samples per period.

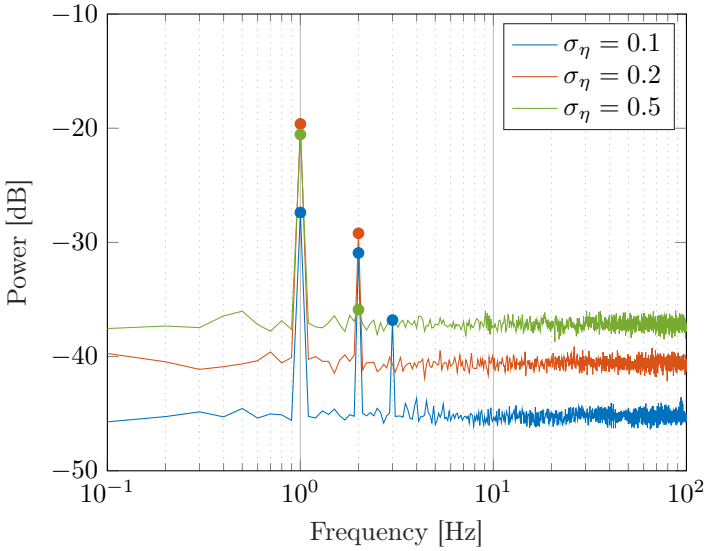


Figure 3.4.3: The power spectral density of the input and output of the simulated of the comparator based subthreshold SR system. The input signal is: $x = 0.2\sin(2\pi t + \frac{1}{2}\pi) + 0.2$ and the noise bandwith is 100 Hz.

3.5 Suprathreshold SR – Schmitt-trigger

In the previous mentioned SR systems, the threshold was larger than the input signal magnitude. This section discusses suprathreshold SR in a system with hysteresis. The system that is used, is equal to the system used in Section 3.3. The H parameter is reduced from 4 to 1, smaller than two times the signal magnitude of 1.

Figure 3.5.1 shows the input-output relation of the system. The system is simulated with input signal $x = \sin(2\pi t)$. White noise is added with standard deviation $\sigma_\eta = 1$. The sampling frequency is set to $f_s = 200$ Hz.

The simulation results are shown in Figure 3.5.2. Without noise, the output would be a perfect square wave. The addition of noise introduces extra state transitions. The power spectral density of the output for noise levels $\sigma_\eta = [0.5,\ 1,\ 2]$ is shown in Figure 3.5.3. The cut-off frequency of the Schmitt-trigger is shifted to a higher frequency compared to the subthreshold example, due to the lower hysteresis distance. Around the zero-crossings, the state is randomized due to the noise. Since the probability of each state depends on the signal input value, integration of the output can be applied to reconstruct the input signal.

Unlike subthreshold SR, the power at the signal frequency decreases when the noise increases. Since the output power is 1, the total error power increases with increasing noise levels.

Noise filtering is thereby necessary for SR to occur. The beneficial effect of the noise is observed in the reduced harmonics for higher noise values. By filtering the noise, the beneficial effect of decreased harmonics can be larger than the negative effect of the higher output noise. The output plot in Figure 3.5.2b shows a filtered output. The filtering is again performed by averaging over 20 samples.

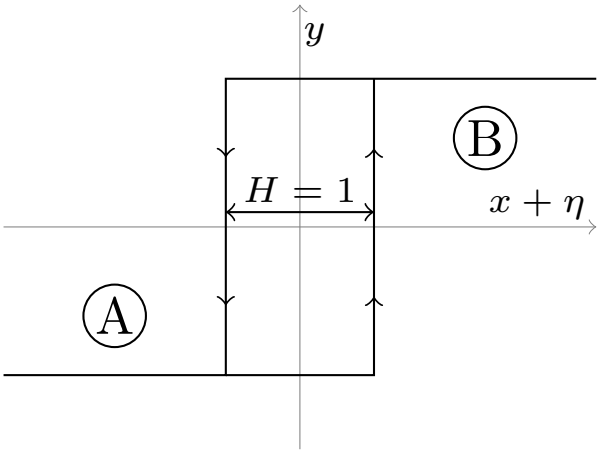
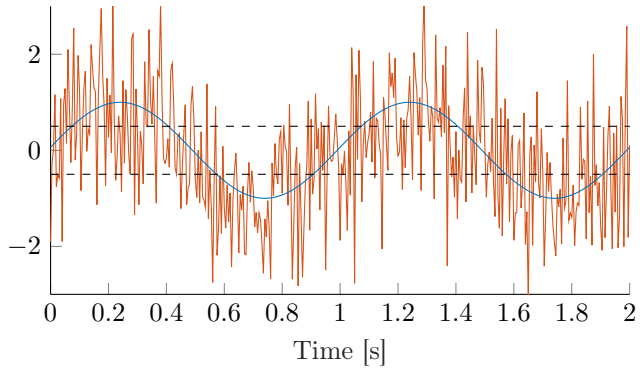
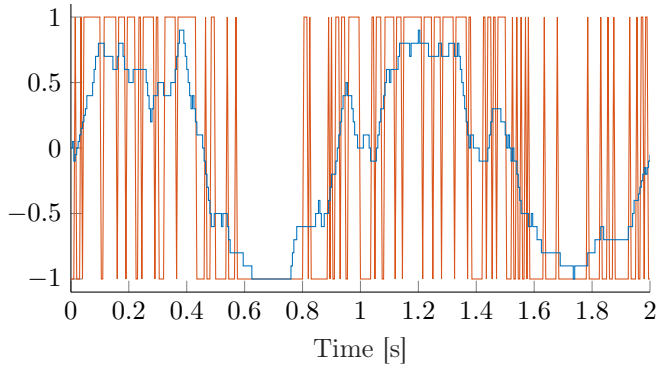


Figure 3.5.1: System transfer of suprathreshold SR with hysteresis, $H = 1$.



(a) Input signal, with and without noise



(b) Output of the comparator, and the moving average of 20 output samples

Figure 3.5.2: An example of suprathreshold stochastic resonance with hysteresis.

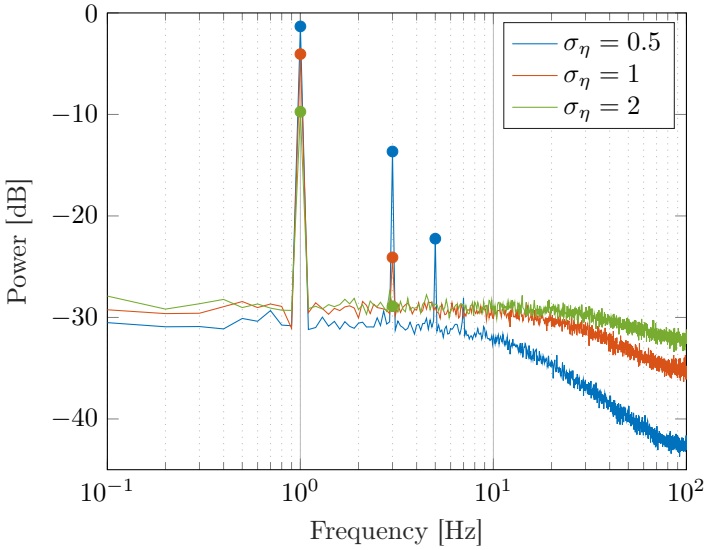


Figure 3.5.3: The power spectral density of the input and output of the simulated suprathreshold SR system with Schmitt-trigger. The input signal input amplitude is 1, while the noise value varies from 0.5 to 2. The noise bandwith is 100 Hz.

3.6 Suprathreshold SR – Comparator

The final class distinguished in this classification is suprathreshold SR with a static threshold. Figure 3.6.1 shows the input-output transfer of this system. The threshold is set to DC-level of the signal, the optimal value according to the literature [8, 19].

Figure 3.6.2 shows the output of a simulation of the system. The simulation is performed using input signal: $x = \sin(2\pi t)$. White noise with standard deviation $\sigma_\eta = 1$ is added to the signal, and the sampling frequency f_s is 200 Hz.

The output is again considered a pulse-density modulated signal. The expected output is determined by the probability of a positive input $P(y = 1|x+\eta)$ minus by the probability of a negative input $P(y = -1|x + \eta)$. By averaging over 20 samples, the pulse density is computed, and shown in Figure 3.6.2b.

Frequency Behavior

The power spectral density of the simulated outputs is shown in Figure 3.6.3. The simulation is performed using the input signal: $x = \sin(2\pi t)$, with a varying noise level with a bandwidth of 100 Hz. Similar to SSR with a variable threshold, SR does not occur without noise filtering. Increasing the noise leads to a lower output value, and thereby a lower SNDR. Only after low-pass filtering noise has a beneficial effect on the performance.

The plot clearly shows a decreasing third harmonic for larger noise values. Since the threshold is set at DC-level the output is symmetrical around zero and, unlike subthreshold SR without hysteresis, does not contain second harmonics. The noise floor increases when the input noise increases, showing the trade-off between output noise and harmonics.

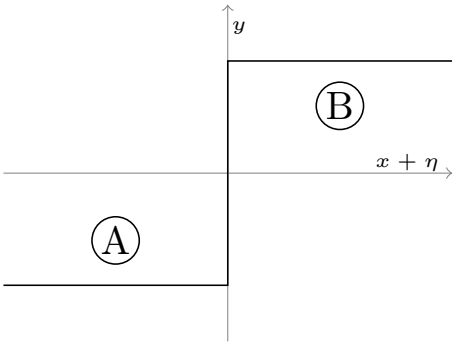
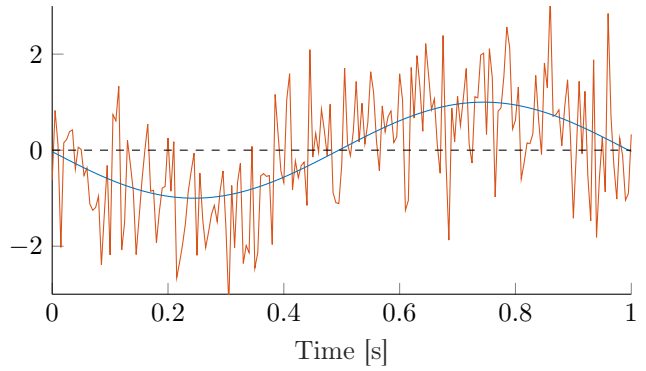


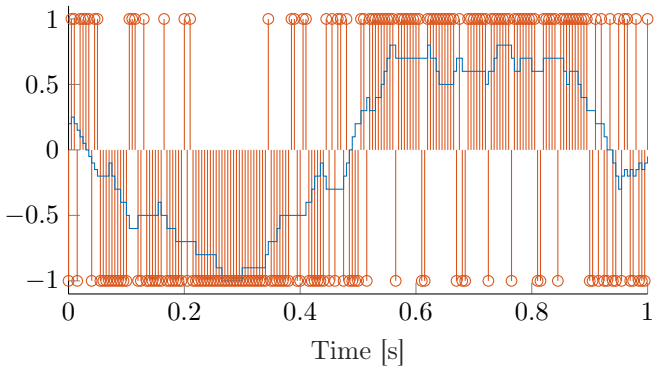
Figure 3.6.1: System transfer of suprathreshold SR with comparator

This class is also referred to as dithering SR [21]. The fundamental mechanism used in dithering is also observed in this system. The quantization error, in this system observed as higher harmonics, is randomized by the addition of noise.

Dithering and SR, however, should not be used interchangeably. First of all, the phenomenon SR is broader than dithering. Furthermore, SR is a natural phenomenon which occurs in systems with naturally existing noise. Dithering, on the other hand, is a technique where noise is artificially added to the system, to achieve the same effect. Dithering can be considered a *technique* which uses the *phenomenon* SR.



(a) Input signal, with and without noise



(b) Output of the comparator, and the moving average of 20 output samples

Figure 3.6.2: Simulations of the system. The input comprises a sine wave with amplitude $X = 1$ and white noise with standard deviation $\sigma_\eta = 1$. The threshold θ and hysteresis H are zero, a perfect comparator is simulated. The noise is simulated using 200 white noise samples per period.

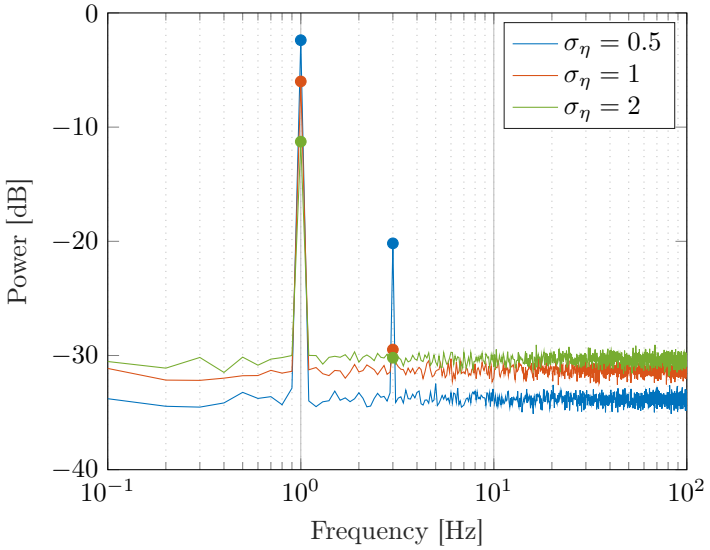


Figure 3.6.3: The power spectral density of the input and output of the simulated comparator system with $\theta = 0$. The input is again a sine wave with amplitude $X = 1$.

3.7 Conclusions

Stochastic resonance is classified into four classes, based on 2 fundamental characteristics. The classification is performed based on the fundamental parameters that define the design space and is thereby optimized for engineering purposes, rather than for descriptive work of natural phenomena.

The two main classes comply with literature and are subthreshold and suprathreshold. In both classes, two subclasses are defined: with hysteresis and without hysteresis.

Subthreshold versus Suprathreshold

In subthreshold SR, the signal is not detected without noise. Adding noise increases the SNDR of the output signal until an optimum is achieved. In suprathreshold SR, on the other hand, the signal is detected without noise. The addition of noise to the non-linear block only does not lead to SR. The added noise at the output is, based on SNDR, larger than the beneficial influence of the noise. To achieve stochastic resonance, filtering of the noise is required. Since the noise is wideband, this can be done by a low-pass filter.

The optimum noise value in subthreshold SR is determined by the signal strength and the noise at the output. In suprathreshold SR, the optimum noise value is determined by a trade-off between the suppression of harmonics and noise.

Integration

When the non-linear block has a hysteresis, such as a Schmitt-trigger, the system has a memory. This leads to a low-pass behavior. The cut-off frequency is determined by the sampling frequency, the size of the hysteresis and the standard deviation of the noise and the input signal. In subthreshold, this is exploited by minimizing the passband of the

Schmitt-trigger, which lead to a higher gain in the passband. Since the output noise is already filtered, integration over time has no significant advantage.

In suprathreshold, the cut-off frequency of the Schmitt-trigger is higher. The high-frequency noise appears as extra state transitions and the input signal amplitude can be reconstructed using a low-pass filter. The expected output value is determined by the state probabilities.

When the non-linear block is implemented by a comparator, white noise at the input results in a white noise spectrum at the output. The broad noise spectrum can be filtered by a low-pass filter.

Only in subthreshold SR, integration, or low-pass filtering, has no significant beneficial effect. Stochastic resonance based on comparators and suprathreshold SR in a Schmitt-trigger benefit from integration. By integrating, the input signal amplitude is reconstructed. The expected output is determined by the state probabilities.

Design space in this project

Based on the observations in different SR classes, the design space is determined. In subthreshold SR with hysteresis, the power in the fundamental is enhanced, but the signal amplitude is lost. When a comparator is applied in subthreshold, the signal amplitude is, to some extent preserved. However, a large second harmonic is observed.

In the suprathreshold SR simulations, the threshold was set to zero. No second harmonic is observed and the distance between the noise floor and the signal power is higher. These observations comply with the observations in the literature that the highest accuracy is achieved when the threshold is zero.

By applying a hysteresis to suprathreshold SR, the output signal increases slightly. On the other hand, the relative power in the harmonics, and the noise at low frequencies is increased.

To achieve the highest accuracy of signal reconstruction, suprathreshold SR is the most promising option. A simple comparator is expected to outperform an implementation with hysteresis, while the low-pass behavior of the Schmitt-trigger decreases the efficiency of integration over time.

The design space in this project will be limited to SSR, using an ideal comparator and a low-pass filter. A comprehensive analysis of the behavior of this system, with and without hysteresis, is presented in Chapter 4.

4

Analysis of Stochastic Resonance in Single-bit ADC

In the previous chapter, four classes of SR systems have been presented, and their behavior is qualitatively described for each class. Based on this qualitative description, the design space of this project is limited to supra-threshold SR. The analysis first focusses on using a simple comparator without hysteresis. Based on this analysis, the performance of the system with hysteresis is analyzed. This analysis led to a novel technique to increase the performance of a SR-ADC, by using a negative hysteresis in the comparator.

In this chapter, an elaborate analysis of stochastic resonance based on a single comparator is presented. The analysis allows to calculate the performance of the SR system, based on fundamental design parameters. Different signals are applied to the system to show the dependency of the system, and its performance, on the input signal.

The behavior of the system is highly dependent on the signal, as well as the noise at the input. To design a reliable stochastic resonance front-end, control over the input is crucial. Since the input signal cannot be controlled, the system should be controlled via the input noise.

The goal of this analysis is to create in-depth insight into the mechanism behind this application of stochastic resonance and quantify the performance of the system. This insight allows to determine the measures and output characteristics that can be used to control the input noise and to ensure the optimal performance of the system. Furthermore, the analysis will be used to determine the specifications of the final system design.

The analysis is performed in several steps. First, the system is introduced in Section 4.1. Secondly, the expected transfer is derived in Section 4.2. The expected transfer is used to derive the linear gain and the harmonic distortion of the system in Section 4.3. The noise at the output of the system is analyzed in Section 4.4. The analysis of the gain, harmonic distortion, and noise are combined to determine the signal-to-noise-and-distortion ratio (SNDR) in Section 4.5. In Section 4.6 a static analysis is performed, and a more detailed analysis of the non-linearity errors is presented. Finally, the analysis is expanded

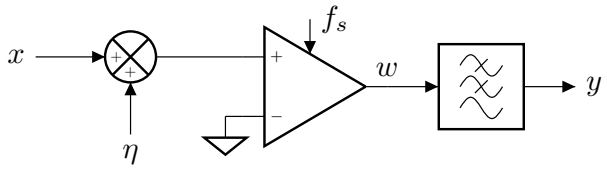


Figure 4.1.1: System level diagram of the analyzed system.

to systems with hysteresis in Section 4.7 and Section 4.8, presenting an improved SR block based on negative hysteresis.

4.1 System Introduction

The system that will be analyzed is shown in Figure 4.1.1. It consists of a summation of signal input x and noise η, and an ideal, clocked, comparator with sampling frequency f_s. The second input of the comparator is set to ground. The comparator output, w, is filtered by a low-pass filter, with output y.

Integration

While integration can be done by parallelization, implementation would need high numbers of parallel comparators and high numbers of independent noise sources. All comparators and noise sources will use precious resources such as area and power.

Instead of implementing the integration in space, the integration will be implemented in the time domain. Since the system targets low-frequency input signals, a high oversampling ratio can be achieved, thus integration, or averaging, in time can be performed using a low-pass filter. This means only one comparator has to be implemented, and instead of multiple independent noise sources, the independence of multiple samples in time can be exploited.

Linear Transfer in State and State Transition

In the previous chapter, the linear transfer in the state δy_s, and the transfer in the state transition δy_t was not taken into account, since this would add a linear behavior rather than showing the SR-mechanism. While it does not add to the SR, it is not said that the performance will be less when δy_s and δy_t are not equal to zero and infinite, respectively.

However, the goal of the thesis is to design a fully integrated analog-front-end, a SR-ADC. When an ideal comparator is used, the signal is directly converted into the binary domain. A non-zero $\delta y_s/\delta x$ or a finite $\delta y_t/\delta x$ result in a non-discrete output, and further analog-to-digital conversion would be necessary. To avoid the necessity of adding a separate ADC, an ideal comparator is used to perform the stochastic resonance.

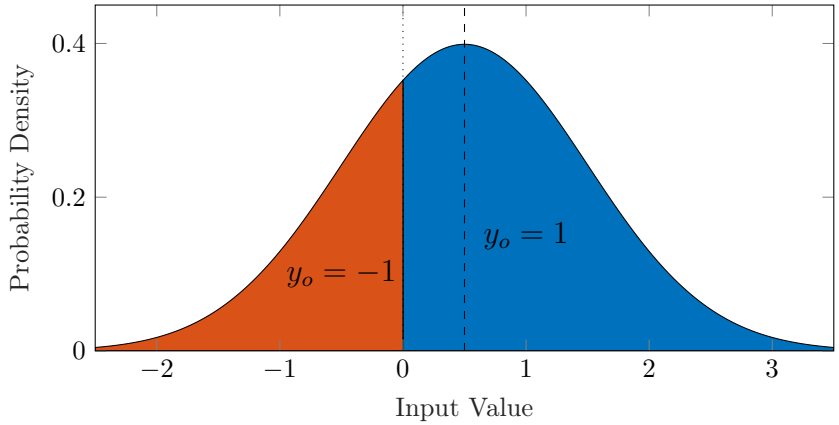


Figure 4.2.1: The probability density function of an input consisting of a value $x = 0.5$ and white Gaussian noise with $\sigma_\eta = 1$. The orange area represents the probability that $w = -1$, and the blue area represents the probability that $w = 1$.

4.2 Expected Transfer

The output of the comparator is considered a stochastic process. There is no direct linear transfer from the input to the output of the comparator, and the state of the comparator at each moment in time cannot be predicted. The expected comparator output value can be determined by the probability of each of the two states, either $w = 1$ or $w = -1$. Since the low-pass filter is used to filter out noise, and should not alter the signal in the passband, the expected output $\hat{y} = \hat{w}$.

Figure 4.2.1 shows the probability density function (PDF) a noisy input. Gaussian white noise, with $\sigma_\eta = 1$, is added to an input value 0.5. The total probability of $w = 1$ is equal to the blue area, while the probability of $w = -1$ is equal to the orange area. The expected comparator output is derived as follows:

$$\begin{aligned} \hat{w}\left(x, \sigma_\eta\right) &= \mathbf{E}\left[w|x\right] \\ &= P\left(w = 1|x\right) - P\left(w = -1|x\right) \\ &= P(\eta > -x|x) - P(\eta < -x|x) \\ &= 1 - 2 \cdot P(\eta < -x|x) \end{aligned} \tag{4.2.1}$$

The probability $P(\eta < -x|x)$ can be calculated by integrating over the probability density function. The mean value μ of the input noise is zero.

$$P(\eta < -x|x) = \int_{-\infty}^{-x} \frac{1}{\sigma_\eta \sqrt{2 \cdot \pi}} \cdot e^{-\frac{\eta^2}{2\sigma_\eta^2}} d\eta \tag{4.2.2}$$

By using the error function defined as [23]:

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-u^2} du$$

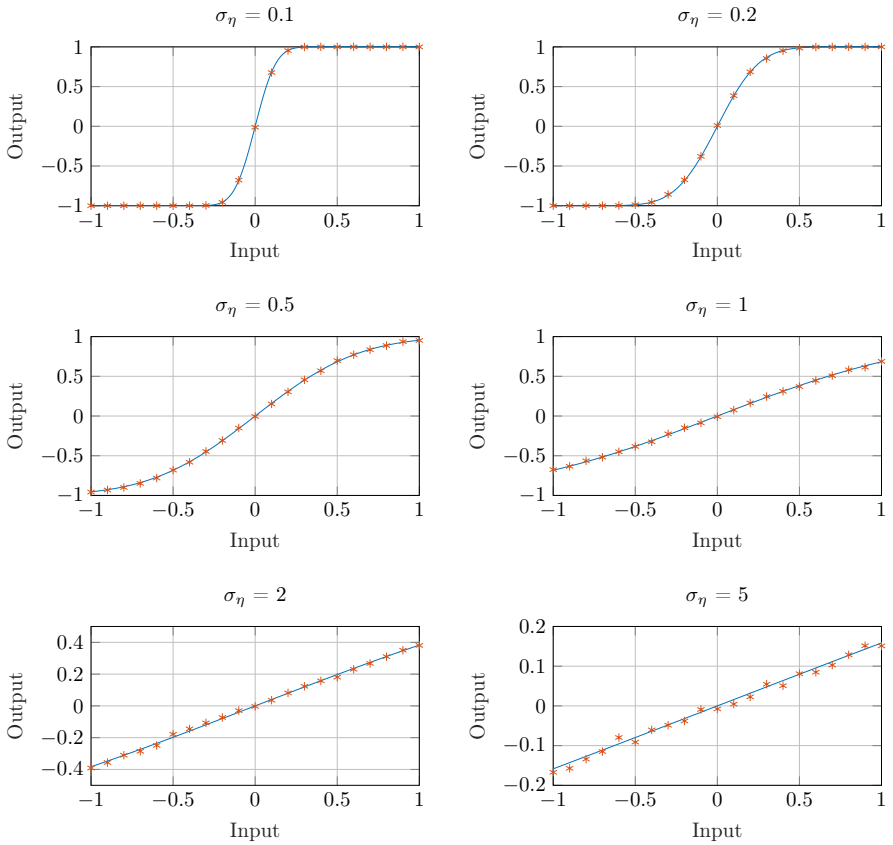


Figure 4.2.2: Expected input-output relations for different noise levels. The calculated and simulated outputs are plotted. The calculated output is represented by the blue line. The simulated output is obtained by averaging 10,000 comparator output samples, and is represented by the orange asterixes.

The expected comparator output can be written as:

$$\hat{w}(x, \sigma_\eta) = \text{erf}\left(\frac{x}{\sqrt{2}\sigma_\eta}\right) \tag{4.2.3}$$

A full derivation is shown in Appendix A. Figure 4.2.2 shows plots of the calculated and the simulated expected transfer for different noise levels. The expected comparator output, $\hat{w}(x, \sigma_\eta)$, is approximated in the simulations by averaging the comparator output w over $10,000$ samples.

The figure shows a more linear input-output relation, but also a larger deviation from the calculated expected output, for a higher σ_η. This complies with the observations in the previous chapter regarding the trade-off between harmonic distortion and noise.

The harmonic distortion and the noise will be discussed separately. In the next section, the gain and harmonic distortion will be covered. The noise behavior will be discussed in Section 4.4.

4.3 Gain and Harmonic Distortion

Although the system is built using highly non-linear blocks, the desired transfer should be as linear as possible. The resonance peak, as described in Chapter 2, is a trade-off between noise and harmonic distortion. Therefore an accurate quantization of the harmonic distortion is crucial to analyze the performance of the system.

To evaluate the harmonic distortion in the time domain, the expected output is split into a signal part and the harmonics as shown in Equation 4.3.1. The comparator output signal is defined as the input signal ($x(t)$) multiplied with gain $\hat{H}\left(f(t), \sigma_\eta\right)$.

$$\hat{w}(x(t), \sigma_\eta, t) = \hat{H}(x(t), \sigma_\eta) \cdot x(t) + HD(x(t), \sigma_\eta, t) \tag{4.3.1}$$

The HD term should only contain power in the higher harmonics, while all power of the fundamentals should be identified as the expected output signal, $\hat{H}(x(t), \sigma_\eta) \cdot x(t)$. Ideal splitting of the fundamental frequency and the harmonic distortion is obtained by choosing $\hat{H}(x(t), \sigma_\eta)$ such that the power in HD is minimized. The power at the input frequency in the HD term is in that case zero.

4.3.1 Least Square Error

The linear gain $\hat{H}(x(t), \sigma_\eta)$ is derived using the least square error (LSE) method. This method scales the gain $\hat{H}(x(t), \sigma_\eta)$ such that the squared error, and thus the power of the error, the harmonic distortion, is minimized. The distortion is highly dependent on the shape of the input signal. To show the effect of a different input signal, the analysis of the gain and the harmonic distortion is applied to a single-tone, as well a two-tone signal.

Single-tone

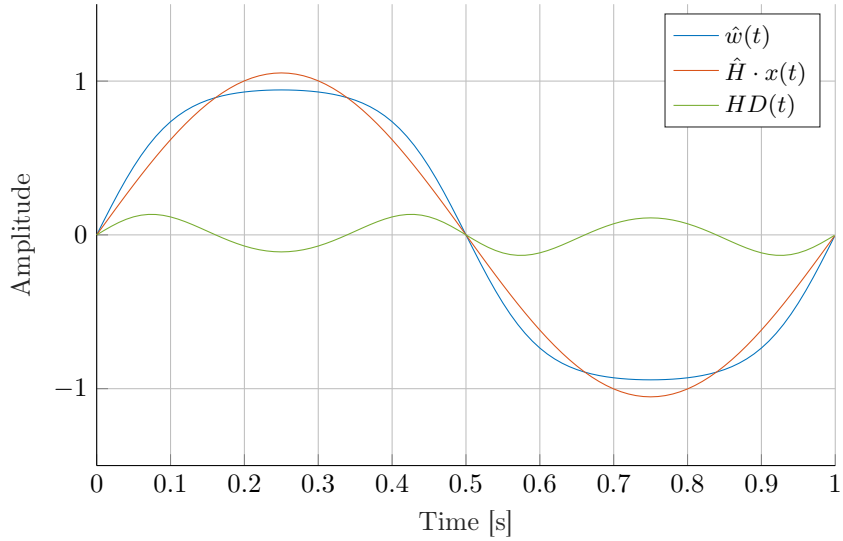
Figure 4.3.1a shows the expected output $\hat{w}(x(t), \sigma_\eta, t)$, the signal part of the output, $\hat{H}(x(t), \sigma_\eta) \cdot x(t)$, and the harmonic distortion for an input signal $x(t) = 2 \cdot \sin(2\pi \cdot t)$, and white noise with $\sigma_\eta = 1$. The frequency spectra, plotted in Figure 4.3.1b, show that an accurate split of the signal and the harmonic distortion is obtained.

Two-tone

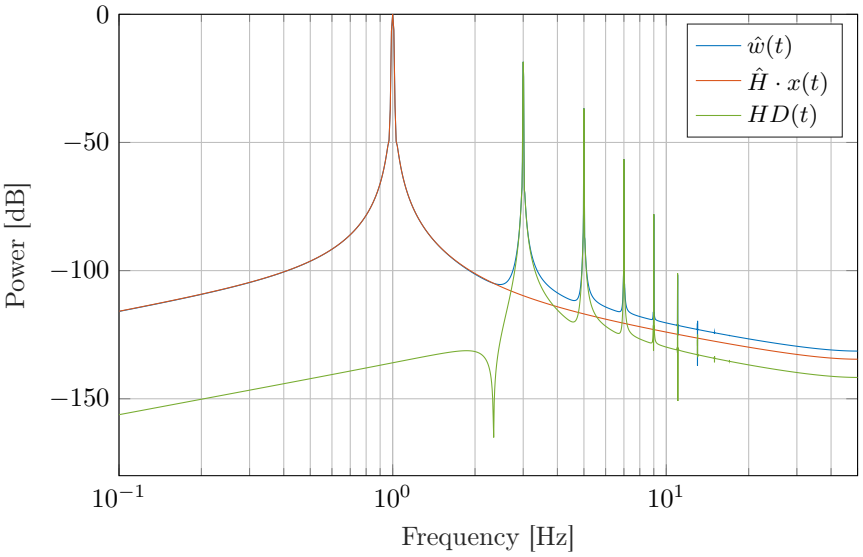
A two-tone signal consists of two sine waves with two frequencies close to each other. This results in intermodulation distortion, which appear at frequencies given by:

$$\begin{array}{c} f_1 + \sum_{n=1}^{\infty} n \cdot (f_1 - f_2) \text{ and} \\ f_2 + \sum_{n=1}^{\infty} n \cdot (f_2 - f_1) \end{array} \tag{4.3.2}$$

The expected output of a two-tone signal is, together with the linear transfer and the harmonics, shown in Figure 4.3.2a. The analysis is performed using signal $x(t) = \sin(2\pi \cdot t) + \sin(1.1 \cdot 2\pi \cdot t)$ and noise with $\sigma_\eta = 1$. The frequency spectra resulting from this input are shown in Figure 4.3.2b. Both in the time domain and the frequency domain, the intermodulation distortion is clearly shown.

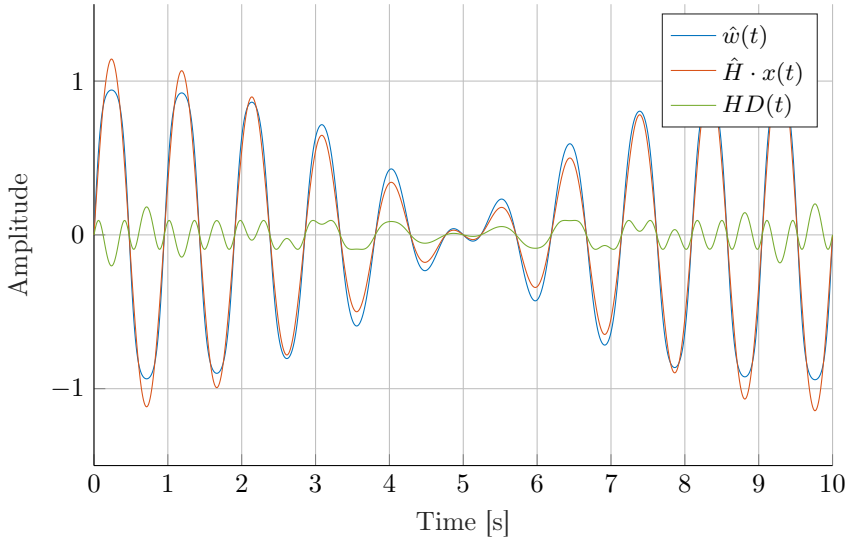


(a) Time domain plot

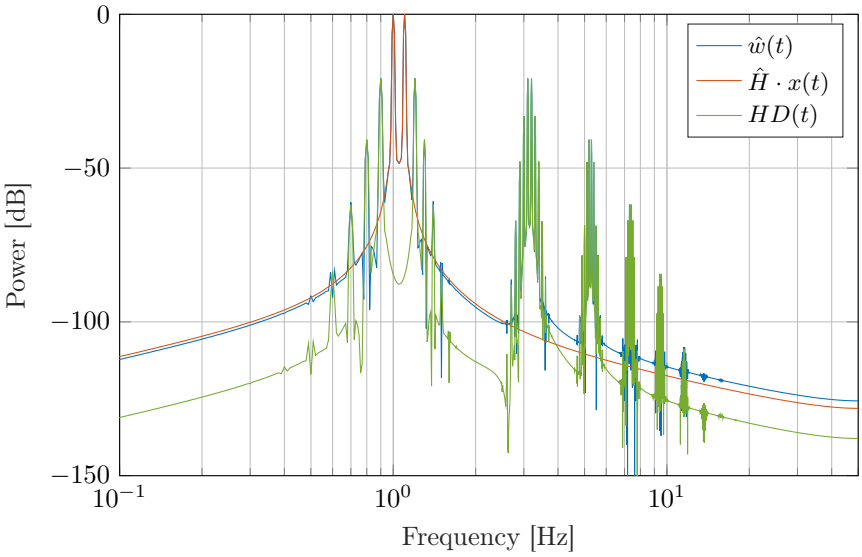


(b) Frequency spectrum

Figure 4.3.1: Expected comparator output, $\hat{w}(x(t), \sigma_\eta, t)$, split into signal, $\hat{H}(x(t), \sigma_\eta) \cdot x(t)$, and harmonic distortion, $HD(x(t), \sigma_\eta, t)$. The input signal is $x(t) = 2 \cdot \sin(2\pi \cdot t)$, with added white noise of $\sigma_\eta = 1$.



(a) Time domain plot



(b) Frequency spectrum

Figure 4.3.2: Expected output, $\hat{w}(x(t), \sigma_\eta, t)$, split into signal, $\hat{H}(x(t), \sigma_\eta) \cdot x(t)$, and harmonic distortion, $HD(x(t), \sigma_\eta, t)$. The input signal is $x(t) = \sin(2\pi \cdot t) + \sin(1.1 \cdot 2\pi \cdot t)$, with added white noise of $\sigma_\eta = 1$.

4.3.2 Gain

The first property that can be derived from the calculations is the gain of the system. Figure 4.3.3 shows the gain plotted against input amplitudes (X_{max}), for the sine wave and the two-tone input. The maximum gain, for both the single-tone and two-tone input, is achieved for:

$$\lim_{\frac{X_{max}}{\sigma_\eta}\to 0} \hat{H}\left(x(t),\sigma_\eta\right) = \frac{d}{dx}\mathrm{erf}\left(\frac{X_{\max}}{\sqrt{2}\sigma_\eta}\right) = \sqrt{\frac{2}{\pi}}. \tag{4.3.3}$$

By increasing the input amplitude, the output swing uses a larger part of the input-output transfer, as shown in Figure 4.2.2. This saturation of the output results not only in more harmonic distortion but also decreases the linear gain.

For a relative high input signal amplitude, $\frac{X_{max}}{\sigma_\eta}$, the output approximates a square wave. In case of the single-tone input, the square wave can be represented by a Fourier series, in which the amplitude of the fundamental tone is $\frac{4}{\pi}$. The gain for a high input signal amplitude is thereby given:

$$\text{if } X_{\max} \gg \sigma_\eta, \text{ then } \hat{H}\left(x(t),\sigma_\eta\right) \approx \frac{4}{\pi X_{\max}}.$$

The two-tone input signal gives a higher gain for the same maximum input amplitude. Analysis shows that the gain for a large relative input amplitude, the gain is equal to $\frac{1.621}{X_{\max}} = \left(\frac{4}{\pi}\right)^2 \cdot \frac{1}{X_{\max}}$. The maximum linearized output is for the two-tone is squared the value for the sine wave.

The error due to the harmonic distortion varies over time when a signal is applied. Since the gain is optimized for a specific input signal, the probability density function of the output signal gives a weight to the error at a certain output. For example, a sine wave is a relatively large part of the time close to the maximum amplitude and crosses zero relatively fast. The error occurring at higher amplitudes gets thereby a higher weight than the error around zero.

This weighting is observed in Figure 4.3.1a and 4.3.2a. In the former figure, the harmonic distortion peaks around 0.08s, 0.42s, 0.58s, and 0.92s are higher than the peaks at 0.25s and 0.75s. The highest harmonic distortion occurs for a lower input amplitude. The latter figure, on the other hand, shows the highest harmonic distortion peaks when the signal is at the maximum amplitude. This weighting causes the difference in gain between the sine wave and the two-tone input.

4.3.3 Harmonic Distortion

The harmonic distortion is further analyzed using the two plots in Figure 4.3.4. In Figure 4.3.4a, the RMS value of the harmonic distortion is plotted. For low input amplitudes, the harmonics produced in the two tone analysis are lower than the harmonics in the sine wave analysis: The harmonics are introduced when the signal is at the maximum amplitude, where the PDF of the sine wave is high and the PDF of the two-tone signal is low. When the input amplitude increases, the harmonic distortion level converges to a constant value. The expected output of the comparator starts approximating a square wave, which results in a higher harmonic distortion for the two-tone signal. For extremely large input amplitudes, the harmonic distortion levels converge to:

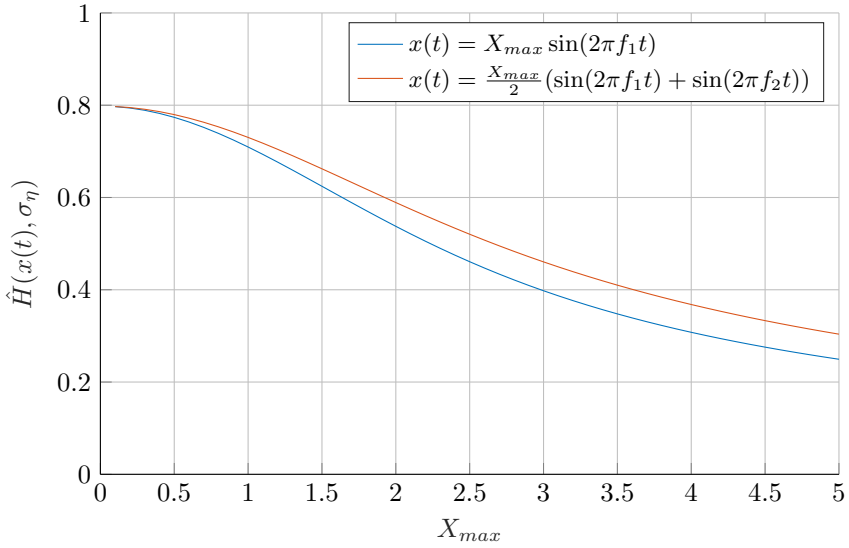


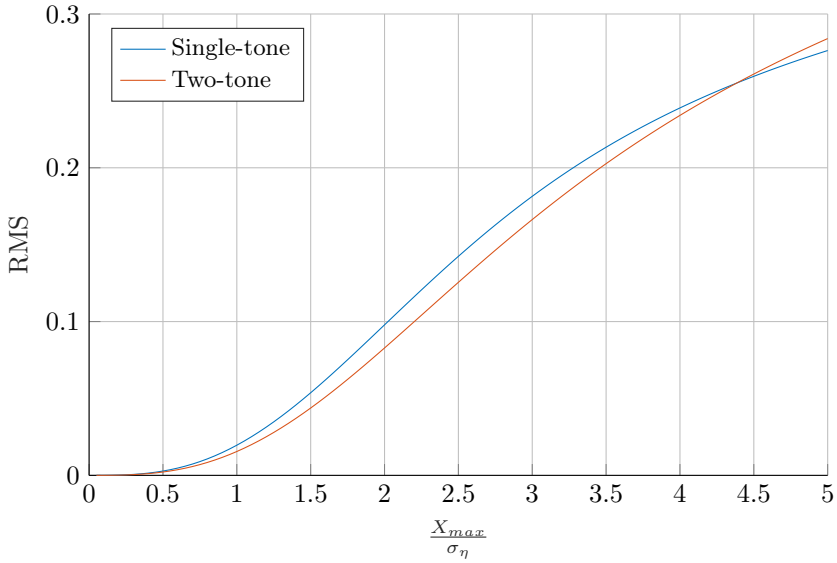
Figure 4.3.3: Linear voltage gain for different input amplitudes of a single-tone, and a two-tone input signal.

$$\begin{aligned} HD_{\text{RMS,sine}} &\approx \sqrt{1 - \left(\frac{4}{\pi X_{\max}}\right)^2 \frac{X_{\max}^2}{2}} \quad for \ \frac{X_{\max}}{\sigma_\eta} \to \infty \\ &= \sqrt{1 - \frac{8}{\pi^2}} \approx 0.44 \end{aligned} \tag{4.3.4}$$

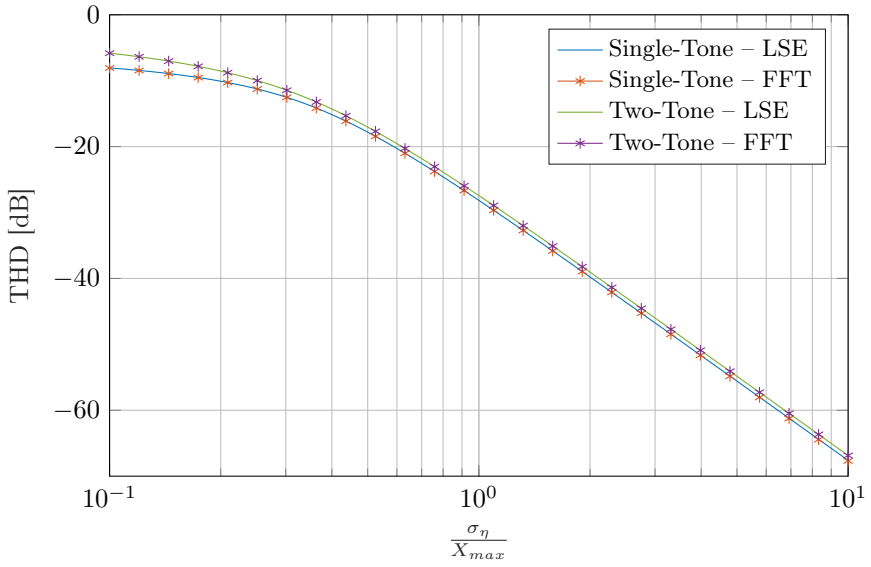
and

$$\begin{aligned} HD_{\text{RMS,two-tone}} &\approx \sqrt{1 - \left(\left(\frac{4}{\pi}\right)^2 \frac{1}{X_{\max}}\right)^2 \frac{X_{\max}^2}{4}} \quad for \ \frac{X_{\max}}{\sigma_\eta} \to \infty. \\ &= \sqrt{1 - \left(\frac{8}{\pi^2}\right)^2} \approx 0.56 \end{aligned} \tag{4.3.5}$$

The plot of the total harmonic distortion (THD) in Figure 4.3.4b gives more insight into the behavior for lower input amplitudes. The THD is defined as the total power in the harmonics divided by the output signal power. The THD is plotted against the standard deviation of the noise relative to the input amplitude. For large noise values, the THD shows an approximately second-order decrease with respect to the input noise level. Both the sine and the two-tone analysis show a decrease of 39 dB in harmonic distortion for a tenfold input noise level. Despite de higher gain and lower RMS value of the harmonic distortion, the THD of the two-tone signal is lower than a sine wave input, due to the higher peak-to-average-power ratio of the two-tone input signal for the same input amplitude. For $\frac{\sigma_\eta}{X_{max}} > 0.5$, the difference in THD is approximately constant at 0.8 dB.



(a) The RMS value of the harmonics, plotted against the input amplitude relative to the standard deviation of the input noise.



(b) Total harmonic distortion of the sine wave and the two-tone calculations. The results of the least square error method are shown, as well as results based on the fast fourier transform.

Figure 4.3.4: Harmonic distortion results for single-tone and two-tone analysis

4.4 Noise

The previous section described the expected behavior of the system. The state of the system at each exact moment in time is unknown. The deviation of the exact output from the expected output appears as noise. In this chapter, an analysis of the noise transfer through the system is presented. First the noise at the output of the comparator is described, secondly the effect of the comparator on the frequency behavior is shown and finally, the noise filtering is examined.

4.4.1 Comparator Output Noise Level

First, the unfiltered noise at the output of the comparator is considered. The noise level depends on the expected output value of the comparator, and thereby depends on the input noise and value of the comparator. The power of the comparator is a constant value of 1, since both $(-1)^2 = 1$ and $1^2 = 1$. The noise is defined as the difference between the expected output and the actual output. Thereby the noise power can be calculated by using the variance of w ([23]:

$$\begin{aligned} \mathrm{Var}\left[w\right] &= E\left[\left(w - \mu_w\right)^2\right] \\ &= E\left[w^2\right] - \mu_w^2 \end{aligned} \tag{4.4.1}$$

The mean value $\mu_w = \hat{w}$, and since the comparator's output is either 1 or -1: $w^2 = 1$. The variance can thereby be rewritten to:

$$\mathrm{Var}\left[w\right] = 1 - \hat{w}^2 \tag{4.4.2}$$

Since the variance of the noise equals the average noise power, and $\hat{w}$ is known:

$$P_{\eta,w} = \mathrm{Var}\left[w\right] = 1 - P_{\hat{y}} = 1 - \mathrm{erf}\left(\frac{x}{\sqrt{2}\sigma_\eta}\right)^2 \tag{4.4.3}$$

The standard deviation is thus:

$$\sigma_w = \sigma_y = \sqrt{1 - \hat{w}^2} = \sqrt{1 - \mathrm{erf}\left(\frac{x}{\sqrt{2}\sigma_\eta}\right)^2} \tag{4.4.4}$$

Figure 4.4.1 shows the output noise standard deviation plotted against the input value, for Gaussian white noise with $\sigma_\eta = 1$. A simulation is performed by deriving the standard deviation of the comparator output of 100,000 samples, to validate the calculations.

4.4.2 Frequency behavior of the noise

The noise behavior of the system is first derived assuming white noise at the input. The power of white noise is flat over the frequency spectrum.

From the definition of white Gaussian noise, a stationary Gaussian stochastic process is white if, and only if both the following requirements are met:

$$\begin{aligned} \mu_\eta &= 0 \\ R_\eta(\tau) &= \sigma_\eta^2 \delta(\tau), \end{aligned} \tag{4.4.5}$$

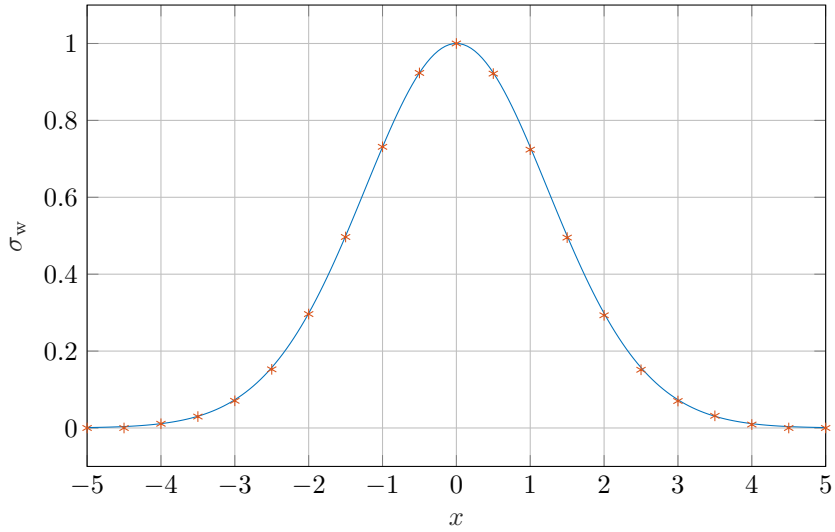


Figure 4.4.1: Standard deviation of the comparator output, y, for different input levels. The blue line represent the calculations, which are validated by means of a simulation, represented by the orange asterix.

in which R_η is the autocorrelation function, and $\delta(t)$ is the delta-dirac function. If the auto-correlation function is zero for all non-zero values of τ, all samples are independent [23].

While the values of each sample are independent, there is also no dependency on whether or not a sample is below or above a certain threshold. In other words: the stochastic process has no memory, and the comparator does not add a memory to the system. Thereby, the independence is preserved through the comparator, and the comparator output noise (σ_w) is white.

4.4.3 Filtered Noise Level

As mentioned in Chapter 3, SR will not occur without reducing the noise by averaging or filtering. If averaging is applied by parellalization, the output noise power can be calculated using the sample mean estimator [23]:

$$e_n = E\left[\left(M_N(w) - E\left[w\right]\right)^2\right] = \mathrm{Var}\left[M_N(w)\right] = \frac{\mathrm{Var}\left[w\right]}{N}, \tag{4.4.6}$$

where e_n is the mean square error, and $M_n(w)$ is the mean estimate. The noise power after averaging is thereby:

$$P_{noise,y} = \frac{P_{noise,w}}{N} \tag{4.4.7}$$

The standard deviation, and thereby the RMS value of the noise becomes:

$$\sigma_y = \frac{\sigma_w}{\sqrt{N}} \tag{4.4.8}$$

The noise power is reduced by the factor $\frac{1}{N}$. This reduction factor is defined as the integration factor IF, where $IF = \frac{1}{N}$. For filtering in time, the same integration factor can be derived, based on the bandwidth of the noise, the passband of the filter and the filter type.

The integration factor of a low-pass filter, with the passband gain set to 0 dB is derived below. The output noise power is determined using the equivalent noise bandwith (f_{ENBW}), defined as:

$$2\pi f_{\text{ENBW}} = \omega_{\text{ENBW}} = \int_0^\infty \left| \frac{H(j\omega)}{H_{max}} \right|^2 d\omega. \tag{4.4.9}$$

The noise power at the output of the filter can be calculated using the ratio between f_{ENBW} and the input (noise) bandwith ($\frac{1}{2}f_s$) :

$$\begin{aligned} P_{\text{noise},y} &= \frac{f_{\text{ENBW}}}{\frac{1}{2}f_s} P_{\text{noise},w} \\ P_{\text{noise},y} &= \frac{f_{\text{ENBW}}}{\frac{1}{2}f_s} \left(1 - \text{erf}\left(\frac{x}{\sqrt{2}\sigma_\eta} \right)^2 \right) \end{aligned} \tag{4.4.10}$$

The standard deviation of the filtered noise, at y, is:

$$\sigma_y = \sqrt{\frac{f_{\text{ENBW}}}{\frac{1}{2}f_s} \left(1 - \text{erf}\left(\frac{x}{\sqrt{2}\sigma_\eta} \right)^2 \right)} \tag{4.4.11}$$

The output noise power is linearly dependend on the amount of integration, defined as the integration factor IF. The integration factor is defined both for integration by parallelization and integration in time:

$$IF = \begin{cases} \frac{1}{N} & \text{for parallelization} \\ \frac{f_{\text{ENBW}}}{\frac{1}{2}f_s} & \text{for integration in time} \end{cases} \tag{4.4.12}$$

In this derivation, a first order butterworth filter is used. The integration factor is in this case:

$$\sqrt{\frac{f_{\text{ENBW}}}{\frac{1}{2}f_s}} = \sqrt{\frac{\frac{\pi}{2}f_{-3\text{dB}}}{\frac{1}{2}f_s}} = \sqrt{\frac{\pi f_{-3\text{dB}}}{f_s}} \tag{4.4.13}$$

Signal input

As the standard deviation of the output depends on the output value, the noise power changes over time when a signal is applied to the input. The instantaneous noise power as a function of time for input signal $x(t)$ is defined as:

$$P_{\text{noise},y}(x(t), \sigma_\eta) = \frac{f_{\text{ENBW}}}{\frac{1}{2}f_s} \left(1 - \text{erf}\left(\frac{x(t)}{\sqrt{2}\sigma_\eta} \right)^2 \right). \tag{4.4.14}$$

The average output noise power is computed by the integral over the expected noise power during one period, divided by the duration of one period. The average noise power and RMS value are thereby:

$$P_{\text{noise},y}(x(t), \sigma_\eta) = \frac{f_{\text{ENBW}}}{\frac{1}{2}f_s} \left(1 - \frac{1}{\tau} \int_0^\tau \text{erf}\left(\frac{x(t)}{\sqrt{2}\sigma_\eta} \right)^2 dt \right), \tag{4.4.15}$$

and

$$\sigma_y(x(t),\sigma_\eta) = \sqrt{\frac{f_{\text{ENBW}}}{\frac{1}{2}f_s}\left(1-\frac{1}{\tau}\int_0^\tau \text{erf}\left(\frac{x(t)}{\sqrt{2}\sigma_\eta}\right)^2 dt\right)}, \tag{4.4.16}$$

in which τ is equal to one period.

The equation above is validated by a simulation. The input signal $x(t) = \sin(2\pi t)$, the used filter is a first-order Butterworth, with a cut-off frequency of $f_c = 10\ Hz$. The equivalent noise bandwidth is thereby $\frac{\pi}{2}f_c$. Figure 4.4.2a shows the standard deviation of the output noise plotted against the input signal amplitude over the standard deviation of the input noise $\frac{X_{max}}{\sigma_\eta}$.

The signal-to-noise ratio (SNR) is shown in Figure 4.4.2b. The output signal power is in this case defined as the power of the expected output, $\hat{w}(x(t),\sigma_\eta)$. The split of the expected output into signal and harmonic distortion is not applied.

For $\lim \frac{X_{max}}{\sigma_\eta}\to 0$, the gain is, as defined in Equation 4.3.3, $\sqrt{\frac{2}{\pi}}$. Furthermore, the harmonic distortion goes to zero and the output noise power becomes $\frac{f_{ENBW}}{\frac{1}{2}f_s}$. The output signal-to-noise ratio can be approximated by:

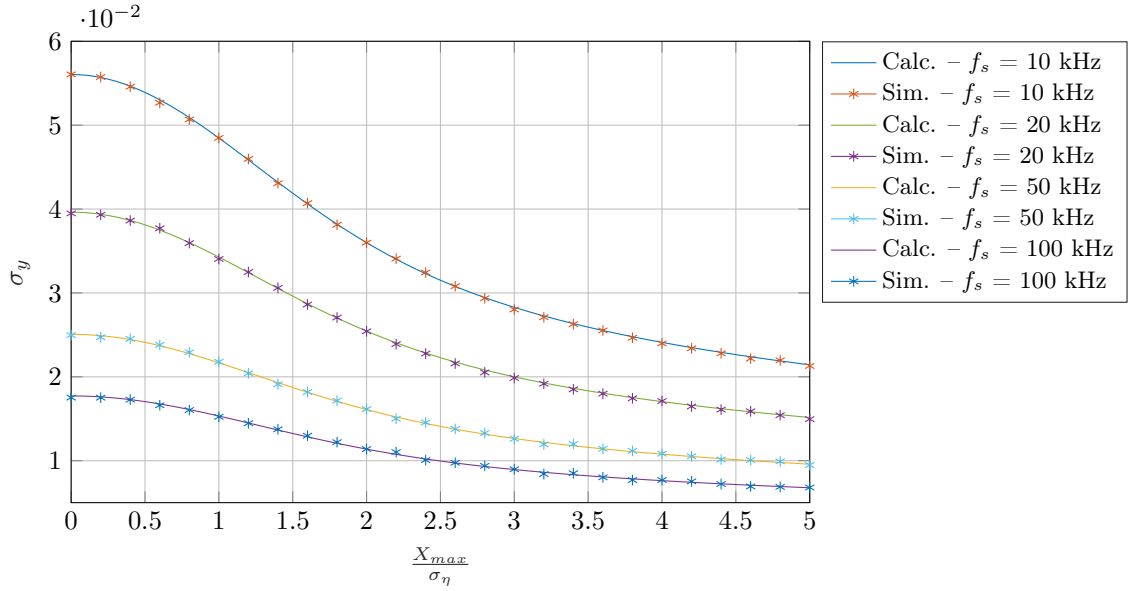
$$\lim_{\frac{X_{max}}{\sigma_\eta}\to 0} \frac{P_{\text{noise},y}(x(t),\sigma_\eta)}{P_{\hat{y}}(x(t),\sigma_\eta)} = \frac{f_{\text{ENBW}}}{\frac{1}{2}f_s}\cdot\frac{2}{\left(\hat{H}(x(t),\sigma_\eta)\cdot X_{\max}\right)^2} = \frac{f_{\text{ENBW}}}{\frac{1}{2}f_s}\cdot\frac{2}{\frac{2}{\pi}X_{\max}^2} \tag{4.4.17}$$

Where $x(t)$ is the sinusodal input, with amplitude X_{max}. The difference between this approximation and the ideal linear system, is only the term $\hat{H}(x(t),\sigma_\eta)^2$. The signal-to-noise ratio loss is thus $\frac{2}{\pi}$, which is -1.96 dB. Figure 4.4.3 shows the SNR-loss as a result of SR, for a single tone and two tone input. For high input noise levels, SNR-loss approaches its final value of -1.96 dB.

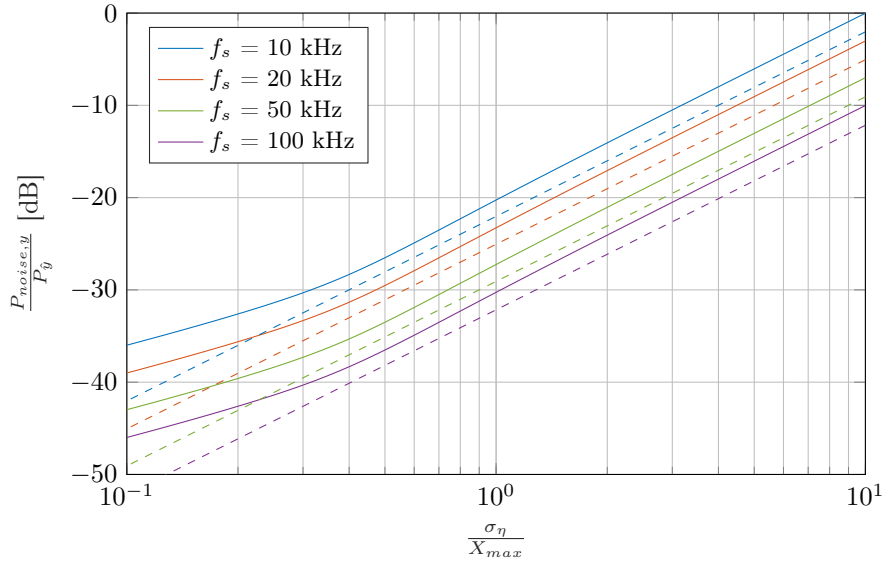
In case of a single tone input , the SNR-loss decreases to -1.52 dB around at $\frac{\sigma_\eta}{X_{max}} \approx 0.5$. By decreasing the noise level, the expected output power ($P_{\hat{y}}$) increases, and thereby the noise power ($P_{noise,y}$) decreases. Meanwhile, the harmonic distortion increases, leading to a lower expected output power gain. At $\frac{\sigma_\eta}{X_{max}} \approx 0.5$, the noise power reduction dominates over the reduction of expected output power gain, which results in the lower SNR loss. The expected output power gain is hereby defined as the expected output power divided by the input signal power.

When the input noise is reduced further, the expected output power gain decreases faster and dominates over the noise reduction. The SNR loss thereby increases.

The peak observed for a single tone input does not appear when a two-tone signal is applied. The signal power is lower, and the noise reduction due to the increased output signal power is always smaller than the decrease in output power gain. This results in a higher SNR loss.



(a) Standard deviation of the output noise for a sine wave input plotted against the amplitude of the sine wave, relative to the noise level σ_η.



(b) Output noise power relative to the expected output power, plotted against the ratio between σ_η and X_{max}, for different sample frequencies. The dashed lines represent the results for a ideal linear system.

Figure 4.4.2: Output Noise for a single-tone input signal.

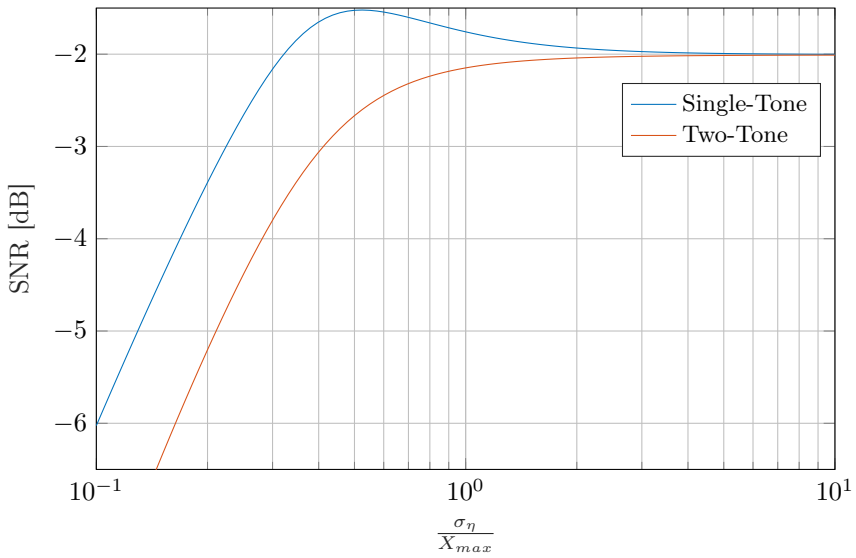


Figure 4.4.3: Difference of Signal-to-Noise Ratio between an ideal linear system and the employed system.

4.5 Signal-to-Noise-and-Distortion Ratio

The noise and harmonic distortion behavior are combined to derive a final output performance. To quantify the performance, the signal-to-noise-and-distortion ratio (SNDR) is used. Using this metric, the error power due to the noise and the harmonic distortion are summed to a total error power. The SNDR is defined as the ratio between the signal power and the total error power.

The performance of a sine wave input for different oversampling ratios is shown in Figure 4.5.1. The plot shows the output SNDR plotted against the standard deviation of the input noise over the input amplitude for different sample frequencies.

Besides the calculation and simulations of the SNDR, also the calculated harmonic distortion and output noise are plotted relative to the output signal power, as the signal-to-distortion ratio (SDR) and the signal-to-noise ratio (SNR). The simulation results are acquired by running all simulations 100 times, to minimize the deviation of the simulation results. The signal frequency is again 1 Hz, the cut-off frequency is set to 10 Hz, and again a first-order Butterworth filter is used. The mean error between the simulation and calculation is 0.01%. The analytic derivation of the SNDR presented in this chapter is thereby an accurate method to predict the performance.

In Tabel 4.1 and Tabel 4.2, the input and the output signal and performance are described for the single-tone and two-tone analysis. Several conclusions can be drawn from this data, which are discussed below.

SNDR As expected, the SNDR increases for higher oversampling ratios. The optimal input noise increases for an increased sampling frequency since not only the noise but also the harmonic distortion should be reduced. The increase is thereby approximately 6.5 dB for a tenfold oversampling ratio, while an ideal linear system would achieve a 10 dB increase of SNR. The rate of increase of 6.5 dB per tenfolding of the sampling frequency is also observed in calculations over a f_s range from 1 KHz to 1 MHz. This value coincides with

f_s	SNDR	$\frac{\sigma_\eta}{X_{\max}}$	Output Amplitude $\hat{H}(x(t), \sigma_\eta) \cdot x_{max}(t)$	σ_y	HD_{RMS}	SNDR loss
10 kHz	20.45 dB	0.79	0.84	$4.49 \cdot 10^{-2}$	$3.43 \cdot 10^{-2}$	3.61 dB
20 KHz	22.36 dB	0.90	0.77	$3.32 \cdot 10^{-2}$	$2.48 \cdot 10^{-2}$	3.58 dB
50 KHz	24.91 dB	1.06	0.68	$2.20 \cdot 10^{-2}$	$1.60 \cdot 10^{-2}$	3.59 dB
100 KHz	26.86 dB	1.19	0.62	$1.59 \cdot 10^{-2}$	$1.17 \cdot 10^{-2}$	3.66 dB

Table 4.1: Properties of the input and output signal at the stochastic resonance peak of a single-tone input

f_s	SNDR	$\frac{\sigma_\eta}{X_{\max}}$	Output Amplitude $\hat{H}(x(t), \sigma_\eta) \cdot x_{max}(t)$	σ_y	HD_{RMS}	SNDR loss
10 kHz	17.8 dB	0.72	0.94	$4.93 \cdot 10^{-2}$	$3.54 \cdot 10^{-2}$	4.07 dB
20 KHz	19.78 dB	0.82	0.86	$3.58 \cdot 10^{-2}$	$2.55 \cdot 10^{-2}$	3.96 dB
50 KHz	22.40 dB	0.96	0.76	$2.32 \cdot 10^{-2}$	$1.69 \cdot 10^{-2}$	3.97 dB
100 KHz	24.39 dB	1.09	0.68	$1.66 \cdot 10^{-2}$	$1.19 \cdot 10^{-2}$	3.88 dB

Table 4.2: Properties of the input and output signal at the stochastic resonance peak of a two-tone input

the observation that the harmonic distortion decreases approximately 39 dB for a tenfold increase of noise level, while the output noise only increases 20 dB.

Noise vs. Harmonic Distortion The SR-peak occurs when the derivative of the output noise power and the derivative of the harmonic distortion power with respect to the relative input noise level are equal. This holds for a harmonic distortion power of approximately 1.9-2.0 times smaller than the noise power. The total error power thereby increases approximately 1.51×.

SNDR loss The SNDR loss is defined as the difference between the SNDR in the SR-system and an ideal linear system with the same input, and bandwidth. The SNR loss due to the output noise varies from around 1.7 dB to 1.8 dB, as shown in Figure 4.4.3. Since the total error power is about 1.5x higher compared to the output noise power, the total SNDR loss increases with approximately 1.8 dB. This approximation is based on the first and second order behavior of the noise and the harmonic distortion respectively.

Single-tone vs. Two-tone The two-tone analysis shows a significantly lower performance since the peak-to-average-power ratio is two times smaller compared to a sine wave. The SR-peak, however, occurs at a slightly lower input noise level. The final SNDR is thereby only 2.47-2.65 dB lower. The SNDR loss compared to the ideal linear system is higher for the two-tone input, which can be explained by the SNR loss as shown in Figure 4.4.3.

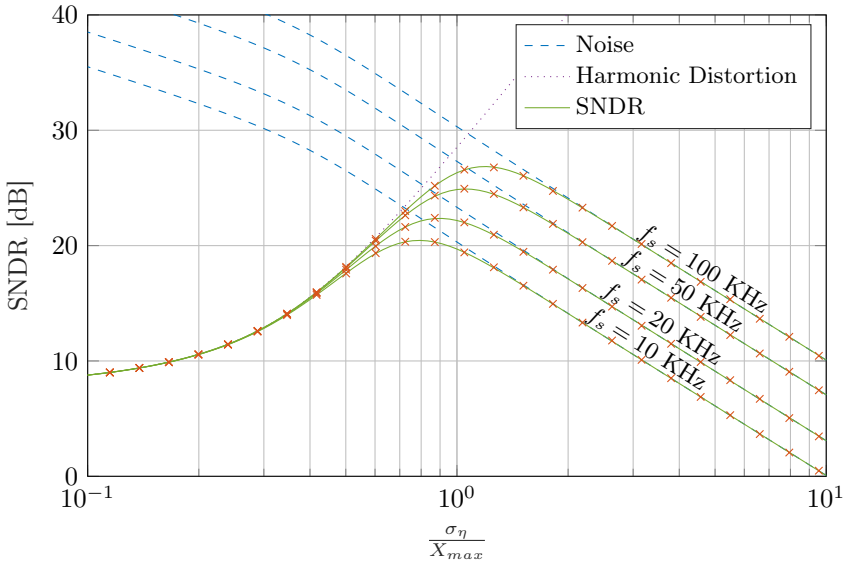


Figure 4.5.1: Signal-to-noise-and-distortion ratio of a sine wave input, plotted against the input amplitude relative to the input noise level,σ_η, for different sample frequencies. The calculated SNDR, as well as simulated results are shown. The contribution of the noise and the harmonic distortion to the SNDR are also plotted.

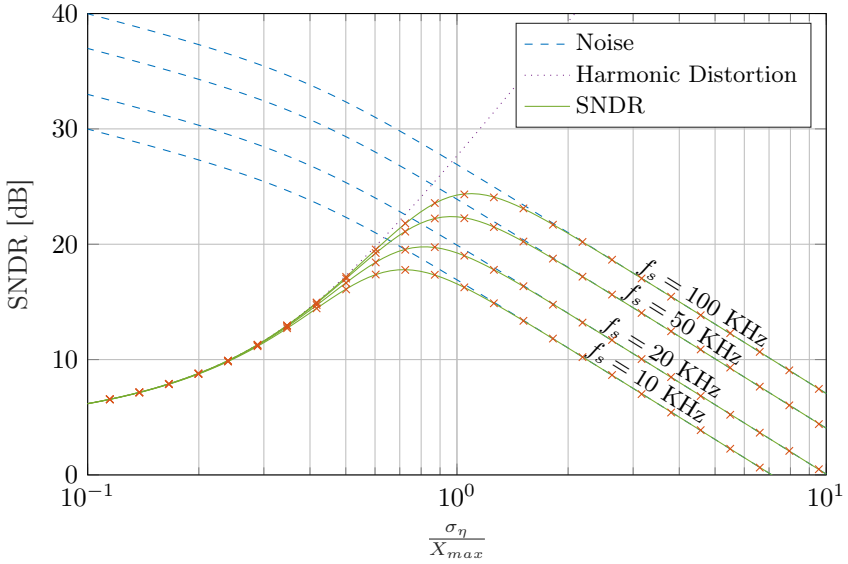


Figure 4.5.2: Signal-to-noise-and-distortion ratio of a two tone input, plotted against the input amplitude relative to the input noise level,σ_η, for different sample frequencies. The calculated SNDR, as well as simulated results are shown. The contribution of the noise and the harmonic distortion to the SNDR are also plotted.

4.6 Static Performance

In the former derivations, it is necessary to assume a certain input signal, and the performance depends on the input signal. This is a fundamental characteristic of this SR-system and should be considered in the evaluation of the performance. In the derivations above, only a sine-wave input and a two-tone input are considered.

Since the error power is not constant for different input values, the error varies over time when a signal is applied. The probability density function (PDF) of the input signal gives a weight to the error at a certain output. The system is optimized for the best SNDR and only takes into account the total expected error.

The expected error as a function of the input is plotted in Figure 4.6.1. This plot shows the error, as a percentage of the dynamic range of the output, for the single-tone and two-tone analysis at the SR peak found in Section 4.5 for $f_s = 100\ textkHz$. The dynamic range is defined as $2 \cdot \hat{H}(x(t), \sigma_\eta) \cdot X_{\max}$.

To eliminate the influence of the non-constant probability density function, a third, static analysis is added to the plot. No signal is assumed, only a dynamic range. The gain of this optimization is determined by linearizing the expected output over the dynamic range. The input noise level is chosen such that the average error power over the dynamic range is minimized. Using this method, all input values, and the corresponding errors, are weighted equally. The minimal error, relative to the dynamic range at the output, is observed for $\frac{\sigma_\eta}{X_{\max}} = 1.1416$. The expected output amplitude is 0.65.

The influence of the input signal on the error optimization is clearly visible. The PDF of the sine wave input is high close to the maximum amplitude and low for values close to zero. The linearization by the LSE minimizes the total average error power. The weighting due to the specific input signal leads thereby to a small error around -1 and 1, and a relatively large error at ± 0.5. The two-tone signal results in a high error for large values of X, and a small error for low input amplitudes. The expected error based on the static analysis performs between the single-tone and the two-tone analysis.

The differences in weighting are reflected in the peak-to-average-power ratio. For a single-tone and a two-tone, this is respectively 2 and 4. The static analysis gives equal weight for each input value. This corresponds with a sawtooth input signal, which has a peak-to-average-power ratio of 3.

To determine the optimal input noise level, not only the average error power should be taken into account, but the maximum error over the dynamic range should be considered as well. Shifting to a slightly higher input noise level can reduce the maximum error significantly, for a low cost of SNDR, as illustrated in the following example: The maximum error for the two-tone input is 3.0%. The SNDR is, as shown in Table 4.2, 24.4 dB, which corresponds to an average expected error of 1.5%. When the noise value is increased to $\frac{\sigma_\eta}{X_{\max}} = 1.19$, the ideal value for a single-tone input, the maximum error of a two-tone input decreases significant, to 2.6%, while the SNDR only decreases 0.14 dB.

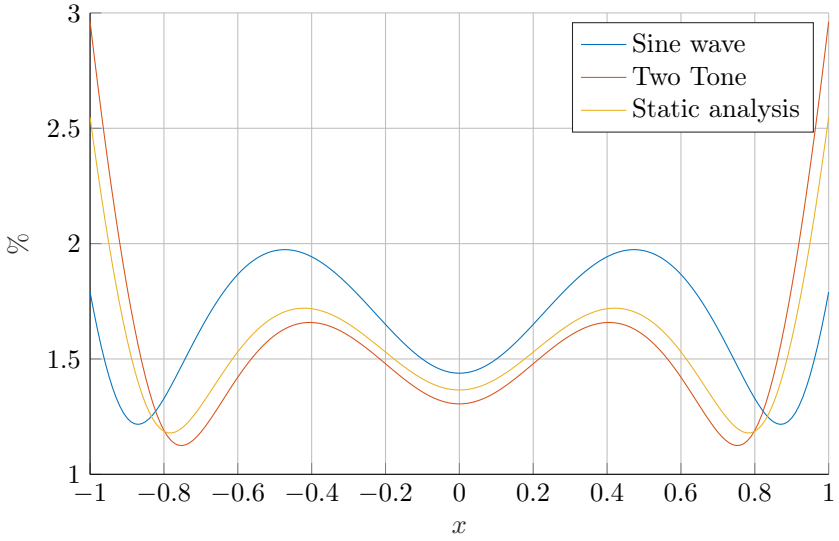


Figure 4.6.1: The expected error as a percentage of the dynamic output range, plotted against the dynamic input range. The expected error of 3 optimizations is shown: Optimization to maximum SNDR of the single-tone and the two-tone input, and an optimization to a minimal average error power of a static analysis. The integration factor is based on a first-order Butterworth filter with $f_{-3\text{ dB}} = 10$ Hz, and $f_s = 100$ kHz.

4.7 Schmitt-trigger Analysis

In chapter 3, it is concluded that a Schmitt-trigger will perform worse than a comparator without hysteresis. This conclusion is based on literature and qualitative insight in the fundamental mechanism behind stochastic resonance. A mathematical derivation of the effect of the hysteresis on the performance, however, is not found in the literature. In this section, the performance of a Schmitt-trigger is quantitatively analyzed, and the effect of a hysteresis on the performance is analytically derived.

4.7.1 Expected Output and Harmonic Distortion

The output of the Schmitt-trigger adds a memory to the system. The probability of the states (either +1 or -1) at sample n depends not only on the input but also on the state of the previous sample $(n-1)$. The Schmitt-triggered system can be analyzed using a Markov Chain [23]), as shown in Figure 4.7.1.

The system has two states, which are defined by the output value (w). From one state, the output of the quantizer can either change state, or stay in the same state. The transition probabilities can be written in a state transition matrix:

$$\begin{bmatrix} P_{LL} & P_{LH} \\ P_{HL} & P_{HH} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} - \frac{1}{2}\text{erf}\left(\frac{x-|\theta|}{\sqrt{2}\sigma}\right) & \frac{1}{2} + \frac{1}{2}\text{erf}\left(\frac{x-|\theta|}{\sqrt{2}\sigma}\right) \\ \frac{1}{2} - \frac{1}{2}\text{erf}\left(\frac{x+|\theta|}{\sqrt{2}\sigma}\right) & \frac{1}{2} + \frac{1}{2}\text{erf}\left(\frac{x+|\theta|}{\sqrt{2}\sigma}\right) \end{bmatrix} \tag{4.7.1}$$

The threshold θ is the hysteresis distance H divided by 2 (see Figure 3.5.1). Since the sample frequency is much larger than the signal frequency, the transition probabilities are assumed to be stationary. The state probabilities at time n can be found using the following equation[23]:

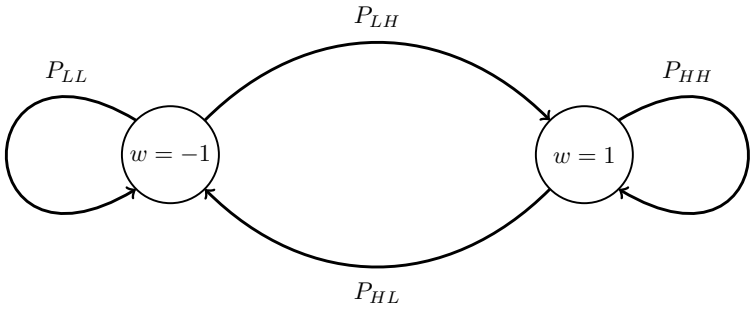


Figure 4.7.1: Markov Chain of Schmitt-triggered SR-ADC

$$p(n) = p(0) \cdot P^n, \tag{4.7.2}$$

Which can be rewritten to:

$$p(n) = \begin{bmatrix} p_0(n) & p_1(n) \end{bmatrix} = \begin{bmatrix} \frac{P_{HL}}{P_{LH}+P_{HL}} & \frac{P_{LH}}{P_{LH}+P_{HL}} \end{bmatrix} + \lambda^n \begin{bmatrix} \frac{p_0 P_{LH} - p_1 P_{HL}}{P_{LH}+P_{HL}} & \frac{-p_0 P_{LH} + p_1 P_{HL}}{P_{LH}+P_{HL}} \end{bmatrix}, \tag{4.7.3}$$

where λ is the eigenvalue unequal to one of the transition matrix:

$$\lambda = 1 - (P_{LH} + P_{HL})$$

Since $|\lambda| < 1$:

$$\lim_{n\to\infty} p(n) = \begin{bmatrix} \frac{P_{HL}}{P_{LH}+P_{HL}} & \frac{P_{LH}}{P_{LH}+P_{HL}} \end{bmatrix} \tag{4.7.4}$$

While high oversampling ratios are employed, the expected output $\hat{w}$ can be calculated using Equation 4.7.4:

$$\begin{aligned} \hat{w}(x,\sigma_\eta) =& \frac{P_{LH}}{P_{LH}+P_{HL}} - \frac{P_{HL}}{P_{LH}+P_{HL}} \\ =& \frac{P_{LH}-P_{HL}}{P_{LH}+P_{HL}} \\ =& \frac{\frac{1}{2} + \frac{1}{2}\mathrm{erf}\left(\frac{x-|\theta|}{\sqrt{2}\sigma_\eta}\right) - \left(\frac{1}{2} - \frac{1}{2}\mathrm{erf}\left(\frac{x+|\theta|}{\sqrt{2}\sigma_\eta}\right)\right)}{\frac{1}{2} + \frac{1}{2}\mathrm{erf}\left(\frac{x-|\theta|}{\sqrt{2}\sigma_\eta}\right) + \frac{1}{2} - \frac{1}{2}\mathrm{erf}\left(\frac{x+|\theta|}{\sqrt{2}\sigma_\eta}\right)} \\ =& \frac{\frac{1}{2}\mathrm{erf}\left(\frac{x-|\theta|}{\sqrt{2}\sigma_\eta}\right) + \frac{1}{2}\mathrm{erf}\left(\frac{x+|\theta|}{\sqrt{2}\sigma_\eta}\right)}{1 + \frac{1}{2}\mathrm{erf}\left(\frac{x-|\theta|}{\sqrt{2}\sigma_\eta}\right) - \frac{1}{2}\mathrm{erf}\left(\frac{x+|\theta|}{\sqrt{2}\sigma_\eta}\right)} \end{aligned} \tag{4.7.5}$$

The gain of this transfer is higher for the same input noise level. Besides the gain, also the linearity differs from the transfer of a comparator without Schmitt Trigger. In Figure 4.7.2, the derivative of the expected transfer is shown for a comparator without hysteresis and for a Schmitt Trigger with $H = 2$. The noise levels are respectively $\sigma_\eta = 1$ and $\sigma_\eta = 1.56$. The derivative in the origin is hereby equal. The plot shows that the gain of the Schmitt-trigger has a faster roll-off near to the origin. This results in higher harmonic distortion since the gain is ideally constant over a wide input range.

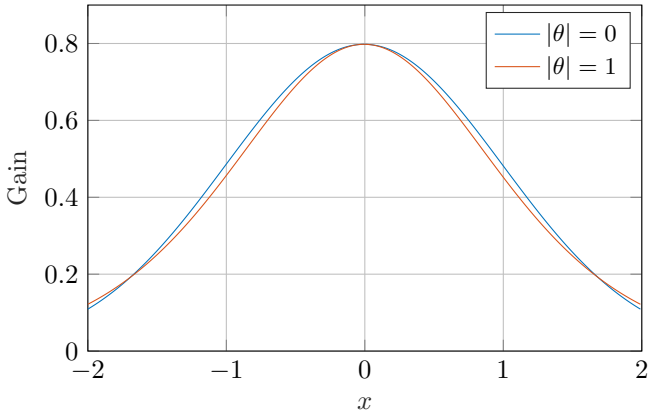


Figure 4.7.2: Derivative of the expected transfer for the system without hysteresis, and with a hysteresis of $H = 2$. The input noise is $\sigma_\eta = 1$, and $\sigma_\eta = 1.56$ respectively. The plot shows the increased non-linearity due to the Schmitt-trigger.

4.7.2 Noise

Figure 4.7.3 shows the spectral power density of the comparator output. The simulated thresholds are: $|\theta| = [0,\ 0.2,\ 0.5]$ and the input only consists of noise, with a standard deviation: $\sigma_\eta = 1$. The employed sampling frequency is 10 kHz.

Due to the low-pass behavior of the Schmitt-trigger, the noise power spectral density increases at low frequency with a factor:

$$\frac{P_{HH}}{P_{LH}} \cdot \frac{P_{LH}}{P_{LH} + P_{HL}} + \frac{P_{LL}}{P_{HL}} \cdot \frac{P_{HL}}{P_{LH} + P_{HL}}. \tag{4.7.6}$$

The bandwith of the Schmitt-trigger, which is directly related to the speed of the converge described above, is larger than the bandwith of the low-pass filter. Hereby, the noise power at the output of the low-pass filter is described by:

$$\sigma_y^2 = \left(\frac{P_{HH} + P_{LL}}{P_{LH} + P_{HL}}\right) \cdot \frac{f_{ENBW}}{\frac{1}{2} f_s} \cdot \left(1 - \left(\frac{P_{LH} - P_{HL}}{P_{LH} + P_{HL}}\right)^2\right). \tag{4.7.7}$$

4.7.3 SNDR

Since the expected transfer and the noise is described, the SNDR can be derived using the same method presented in chapter 4. The simulation results, as well as the calculated SNDR, is shown in Figure 4.7.4. The input signal frequency is 1 Hz, and the filter is a first-order Butterworth filter, with a cut-off frequency of 10 Hz. The sampling frequency is 100 kHz.

The calculations and simulations deviate, in case of $|\theta| = 1$ and $|\theta| = 0.5$ for small values of σ_η. The assumptions made above, regarding the speed of the convergence and the bandwidth of the Schmitt-trigger, do not hold for these values. The bandwidth of the Schmitt-trigger is smaller than the bandwidth of the low-pass filter.

The conclusions based on literature are confirmed. Increasing the hysteresis results a lower stochastic resonance peak. The harmonic distortion increases significantly for larger hysteresis distances. For high noise values, the SNDR is not affected significantly. It is

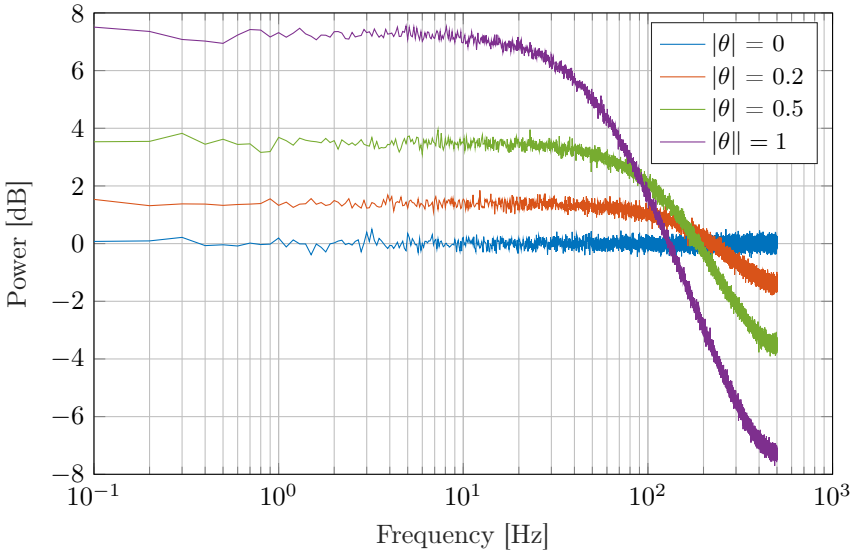


Figure 4.7.3: Frequency spectrum of the comparator output, for different hysteresis distances. The input noise level $\sigma_\eta = 1$, and the sampling frequency is 1,000 Hz. The solid line shows the calculated values, the simulations are denoted by the asterixes

shown that the noise level increases when the hysteresis is increased. However, the gain of the system increases as well. Both effects cancel each other, resulting in an equal SNDR when the SNDR is limited by the noise.

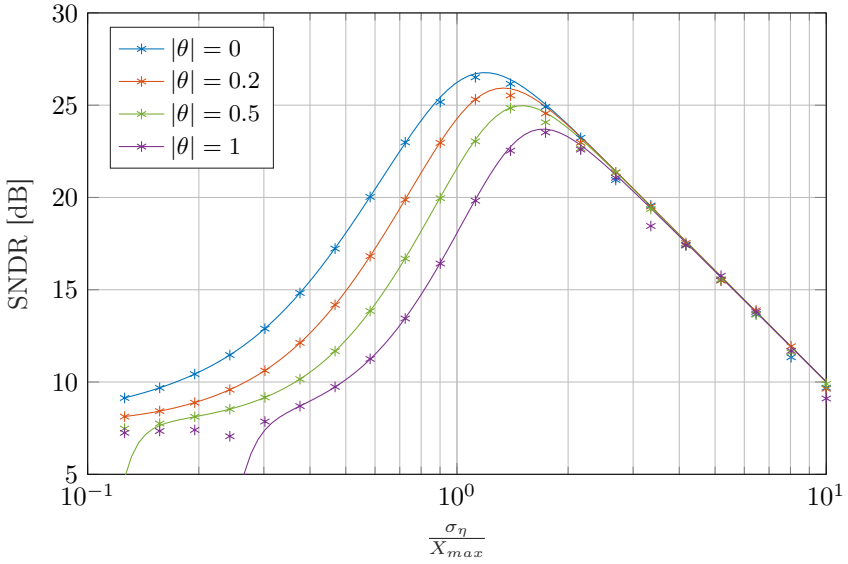


Figure 4.7.4: The results are derived for threshold levels: $|\theta| = [0,\ 0.2,\ 0.5,\ 1]$ and a sampling rate of $f_s = 100\ KHz$. A first order butterworth filter with $f_c = 10\ Hz$ is applied to filter the output. The input signal is: $x(t) = \sin(2\pi t)$.

4.8 Negative Hysteresis Comparator

Where a Schmitt-trigger stabilizes the system from switchings due to noise around zero, a negative hysteresis introduces input range for which the system will be unstable. The threshold value is increased when the comparator output is high and decreased when it's low, as shown in Figure 4.8.1. When the absolute input value is smaller than H, the output will switch every clock cycle.

In a noise-less system, this comparator could be seen as a 1.5-bit system, the third state being the unstable state. The oscillations at $\frac{1}{2}f_s$ can be filtered out, resulting in a reduced quantization error.

In a stochastic resonance system, however, the hysteresis changes the behavior in more ways. The analysis of a negative hysteresis can be done using the same approach as the Schmitt-trigger analysis.

4.8.1 Expected Output and Harmonic Distortion

The expected output of the negative hysteresis system is calculated in the exact same way as the positive hysteresis. The transfer function only changes in the signs before the absolute value of the threshold, and becomes:

$$\hat{w}(x, \sigma_\eta) = \frac{\frac{1}{2}\text{erf}\left(\frac{x+|\theta|}{\sqrt{2}\sigma_\eta}\right) + \frac{1}{2}\text{erf}\left(\frac{x-|\theta|}{\sqrt{2}\sigma_\eta}\right)}{1 + \frac{1}{2}\text{erf}\left(\frac{x+|\theta|}{\sqrt{2}\sigma_\eta}\right) - \frac{1}{2}\text{erf}\left(\frac{x-|\theta|}{\sqrt{2}\sigma_\eta}\right)}. \tag{4.8.1}$$

The gain of the system is reduced for a fixed noise level. The nominator in Equation 4.8.1 is not changed compared to Equation 4.7.5, but the denominator is increased by the threshold shifts. In Figure 4.8.1, the derivative of the expected transfer is shown for a

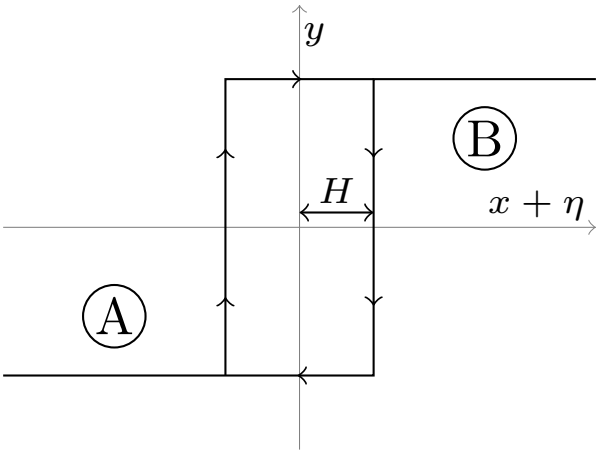


Figure 4.8.1: System transfer for a negative hysteresis comparator

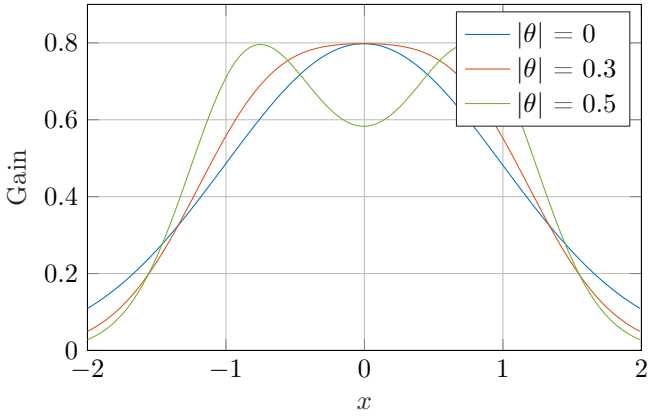


Figure 4.8.2: Derivative of the expected transfer for the system without hysteresis, and with a negative hysteresis of $H = 0.6$ and $H = 1$. The input noise is $\sigma_\eta = 1$, $\sigma_\eta = 0.674$, and $\sigma_\eta = 0.53$ respectively. The plot shows the increased linearity due to the negative hysteresis, and a dip in the gain around zero if the hysteresis is very large.

comparator without hysteresis, and with a negative hysteresis. The noise level is adjusted such that the gain in $x = 0$, is equal for both cases. The plot clearly shows an increased linearity when the negative hysteresis is added to the comparator.

4.8.2 Noise

Just as in the Schmitt-trigger, the noise spectrum is shaped by the hysteresis. While the Schmitt-trigger behaves as a low-pass filter for the noise, the negative hysteresis will show a high-pass behavior. Figure 4.8.3 shows the frequency spectrum of the comparator output when only noise is applied to the input. As expected the low-frequency noise decreases for a larger hysteresis and the oscillations around the Nyquist frequency increase.

The reduction in low-frequency noise is described, again in the same way as the Schmitt-trigger, by Equation 4.7.7.

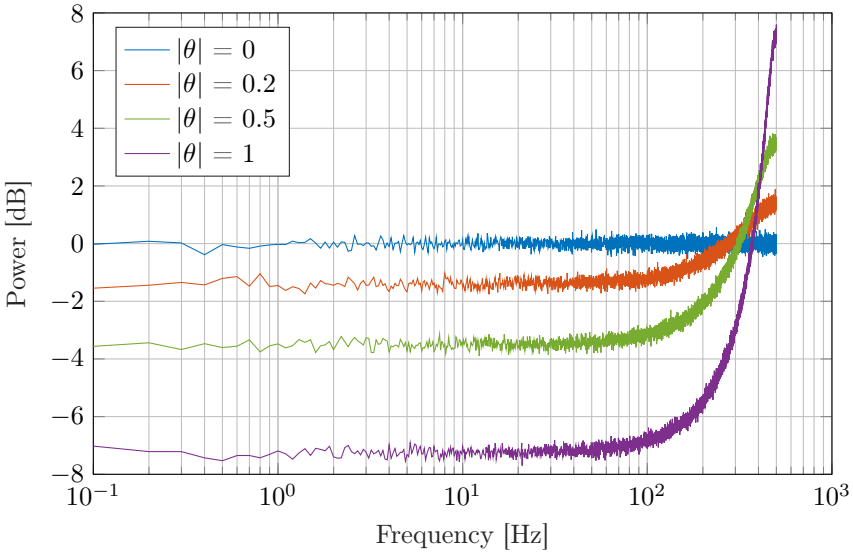


Figure 4.8.3: Figure 0.1.3: Frequency spectrum of the comparator output, for different negative hysteresis distances. The input noise level $\sigma_\eta = 1$, and the sampling frequency is 1,000 Hz.

4.8.3 SNDR

The expected output, the harmonic distortion and the noise level of the negative hysteresis comparator in stochastic resonance are described by the same equation as used for the Schmitt-trigger analysis. The SNDR can thus be calculated using exactly the same method.

In Figure 4.8.4, the SNDR is shown for different negative hysteresis distances. The performance of the system increases drastically by adding this hysteresis to the comparator. The SNDR peaks for $|\theta| = 0.4$ at 33.3 dB, while the standard deviation of the noise is only 0.52. The expected maximum output for this value is 0.78. This clearly shows the increased linearity of the system, since harmonic distortion is significantly lower, while the output is larger.

For high noise levels, the decreased gain and the noise shaping due to the hysteresis cancel each other. The resulting SNDR does not change (significantly) for different hysteresis levels.

In Figure 4.8.5, the power spectral density of both the system with and without the hysteresis are shown. The plot clearly shows the increased output signal amplitude, the low-frequency noise reduction and the increased linearity of the transfer function. A high peak in fifth harmonics does appear. Although the highest SNDR is observed for $|\theta| = 0.4$, the hysteresis is large enough to create a dip in the gain around 0, causing this strong fifth harmonic. The fifth harmonic can be reduced by increasing the noise, this is however at the cost of a higher third harmonic. By shifting from the SR-peak based on the SNDR, to an input noise with $\sigma_\eta = 0.64$, a trade-off between the third and the fifth harmonic can be made such that the spurious free dynamic range (SFDR) is maximized, as shown in the plot. The SFDR is increased from 39.4 dB to 45.4 dB. The price for this higher SFDR is paid in extra noise. The SNDR drops to 32.6 dB, for a maximum expected output of 0.70.

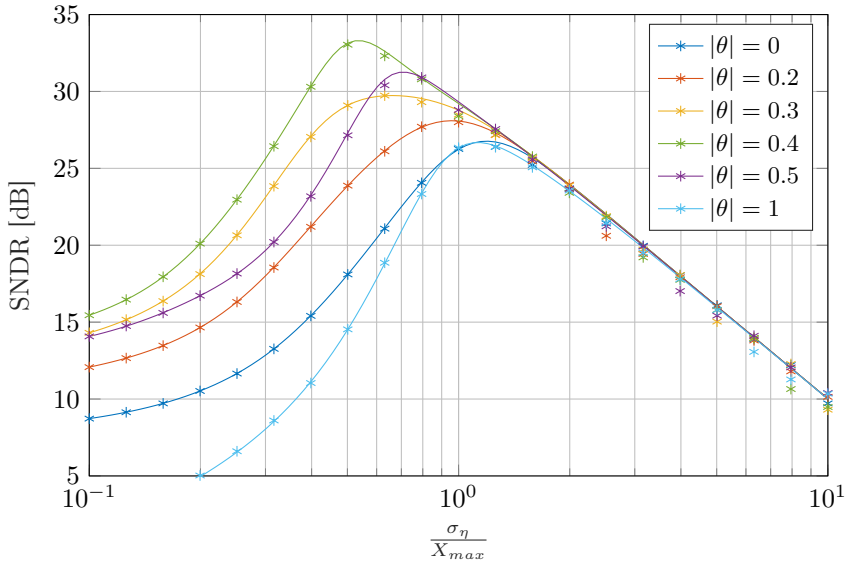


Figure 4.8.4: Signal-to-noise-and-distortion ratio of the system with negative hysteresis. The results are derived for threshold levels: $|\theta| = [0,\ 0.2,\ 0.5,\ 1]$ and a sampling rate of $f_s = 100$ KHz. A first order butterworth filter with $f_c = 10$ Hz is applied to filter the output. The input signal is: $x(t) = \sin(2\pi t)$.

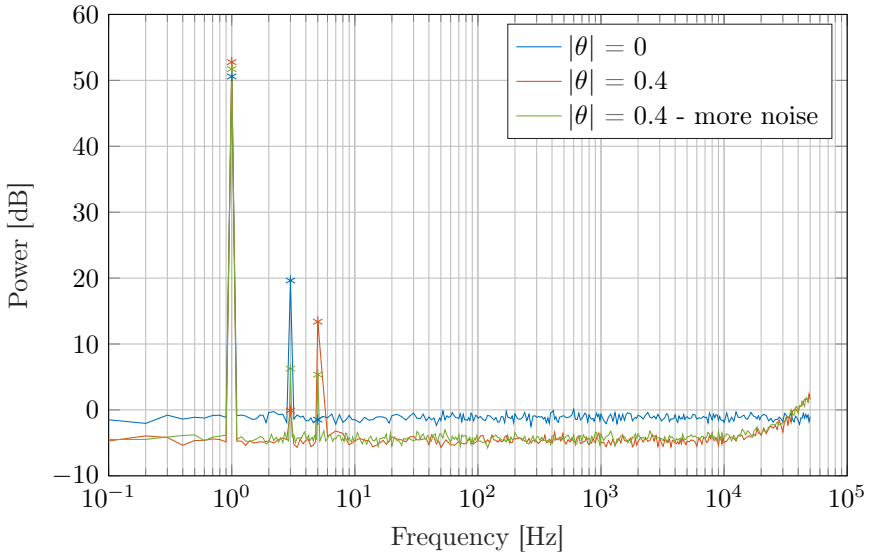


Figure 4.8.5: The output power spectral density of in the SR-peak with and without negative hysteresis. The plot shows an significant.

4.9 Conclusion

In this chapter, the behavior of the proposed SR system is analyzed in detail. An analytic derivation of the expected transfer, linear gain, harmonic distortion, and the noise is presented. The method applied to structurally derive all properties is applied to, and verified by, a single-tone and a two-tone input. However, the method can be applied to all periodic input signals.

The expected output signal is described by:

$$\hat{w}(x, \sigma_\eta) = \operatorname{erf}\left(\frac{x}{\sqrt{2}\sigma_\eta}\right).$$

Secondly, the harmonic distortion is derived by fitting the expected output to the input signal using the least-square-error method. The difference between the fitted signal and the expected output is identified as the harmonic distortion.

The noise is calculated based on two observations: the instantanuous power of the quantizer output is constant, and white noise at the input of the comparator results in white noise at the output of the comparator. The output noise is thereby described by:

$$\sigma_y = \sqrt{\frac{f_{\mathrm{ENBW}}}{\frac{1}{2}f_s}\left(1 - \hat{w}(x, \sigma_\eta)\right)}.$$

The expected output, harmonic distortion, and noise are combined to derive the SNDR of the system. The SNDR results, presented in Section 4.5, show that an SNDR of 26.9 dB and 24.4 dB can be achieved for a single-tone and a two-tone input respectively. This result is achieved for an integration factor $IF = \frac{\pi}{10^4}$. Compared to an ideal linear system, an SNDR loss of 3.6 to 4.1 dB occurs.

The SNDR only represents the error power in respect to the output signal power. The expected output error varies over the input range, which can lead to a maximum error significantly higher than the average error power. To ensure accurate reconstruction of the signal over the complete dynamic range, the expected error over the full dynamic range should be taken into account.

Negative Hysteresis Comparator

The addition of a hysteresis to the comparator is mostly considered to decrease the performance of the system. However, by implementing a novel technique using negative hysteresis in stochastic resonance, a significant improvement in performance is observed.

This new stochastic resonance block exceeds the performance of the comparator in multiple ways. The transfer is much more linear, and thus the dynamic output range in the SR-peak is increased significantly: from 0.60 to 0.78. While the output signal amplitude is increased, the unfiltered output noise power decreases. This effect works therefore in two way to increase the SNDR. Secondly, the noise is shaped by the comparator and the low-frequency noise is reduced significantly.

By using this novel building block to implement stochastic resonance, the final SNDR can be increased by 6.4 dB, from 26.9 to 33.3 dB. By increasing the noise, the harmonic distortion can be distributed over the third and the fifth harmonic. The SNDR drops to 32.6 dB, while the SFDR increases to 45.4 dB.

A large drawback of implementing the negative hysteresis is the required knowledge of the input amplitude (or input noise power). The optimal hysteresis distance is directly related to the input amplitude and input noise. The system itself does not measure the input amplitudes since the transfer is determined by the noise power and input value only.

5

System Level Design

Based on the observations in Chapter 3 and 4, a system level design for the SR-ADC is proposed in this chapter. In the previous chapter, an elaborate analysis of stochastic resonance in comparators is presented.

To implement this principle in a practical system, several challenges should be addressed. The performance of the system depends heavily on the (total) noise level of the system. Meanwhile, it is hard to control external noise in a real-world situation. Moreover, one of the challenges addressed in this thesis is employing the existing noise. To achieve this goal, the noise level should be adjusted dynamically. By implementing the noise dynamically, the signal-to-noise ratio is maximized, independent of its input signal. In this way, the system not only uses existing noise, but the noise can also be adjusted to the input signal amplitude.

The second goal of this thesis is to build an analog-to-digital converter without the need of a pre-amplifier. This means that the system needs to be capable of handling very small input signals. One of the issues arising from this specification is the offset. A small offset in the comparator will lead to a large DC-error at the output. Since the linear part of the input-output transfer is completely occupied, a DC-error should be avoided as much as possible, to prevent harmonic distortion. Besides static offset due to mismatches in the comparator, other very low-frequency signals such as drift from the electrodes can be expected. In this chapter, a dynamic offset compensation technique and its effect on the performance is analyzed. A dynamic offset compensation can, in contrast to an offline offset calibration filter these other, unwanted, low-frequency signals.

Besides the implementation of these control loops, a digital filter is presented. This filter implements the low-pass filter which replaces the ideal Butterworth filter used in the previous chapter.

The chapter starts with a system overview in Section 5.1, after which the system level implementation of the different blocks is presented. In Section 5.2, the digital low-pass filter is presented. A noise control feedback path is presented in Section 5.3, followed by an offset compensation feedback in Section 5.4. In Section 5.5 the simulation results of the system are presented, focusing on the effect of the control loops.

5.1 System Overview

The system examined in this chapter is shown in Figure 5.1.1. The signal processing is, just as in the previous chapter fully performed by the comparator and the digital filter. However, several additions are made to enable implementation in a practical system. The most important additions are the two added feedback paths, to control the dynamic range and the offset, or DC level.

The main block of the system is the comparator, followed by a low-pass filter. While in the previous chapter an ideal Butterworth filter was used, in this chapter a digital filter is implemented. The Butterworth filter was only chosen to accommodate the analysis. The digital filter is implemented using a rectangular window function and converts the high-frequency binary signal into a 488 Hz, 7-bit output signal. The output signal is used in two feedback paths.

The first path controls the noise level based on the output signal. Independent of the input signal amplitude and the already existing noise, the output dynamic range is set such that the system operates in, or close to, the SR peak. The feedback path detects the absolute output signal peak and compares this value with the desired dynamic range. The difference between the real and the desired dynamic range is used as input of the noise adjustment algorithm (NAA). This algorithm decides, based on the measured deviation, how much the noise level is adjusted. The algorithm controls the adjustment such that the sensitivity to small random deviations is minimized. The output of the algorithm is fed to an up-down counter, which adds the new input to its previous value. This digital value is used to control the gain of the variable gain amplifier after the noise source.

The second feedback loop sets the comparator threshold to the correct value. A digital integrator is implemented in the feedback path. This creates a high-pass behavior in the total system, and thus the DC error. The cut-off frequency of this filter is determined by the gain in the feedback path and the time constant of the integrator. Implementing the offset compensation in this manner, not only the comparator offset is compensated, but all DC errors and low-frequency signals (such as drift) are filtered out.

A link from the noise control path to the threshold compensation path is added to adjust the gain of the feedback loop. The noise level is used to estimate the input amplitude. Since the noise level determines the expected transfer function of the comparator, the loop gain depends on the noise level in the system. The loop gain influences the cut-off frequency of the filter implemented by the offset compensation. By multiplying the feedback path with the noise level, the loop gain remains constant.

The implementation and an analysis of the filter and the control loops are discussed in detail in the following sections.

Negative Hysteresis The design of these blocks is based on the system with a zero threshold, so without negative hysteresis. While the system with hysteresis shows promising results in the previous chapter and increases the performance by 6.4 dB, controlling the hysteresis introduces some difficulties. The goal of this thesis is to design a system in which the noise present in the signal can be employed, which is especially interesting in high noise environments. While noise is added to the system to on purpose, the exact total noise level is not known. The control loop which controls the dynamic range only takes care of the ratio between the signal level and the noise level. A second goal is to implement the system without (accurate) pre-amplification. The input of the system can range from 1 to 10 mV. Neither the exact noise level nor the input level is known. The output of the digital filter does not contain any information about the Therefore, the ideal size of the hysteresis can therefore

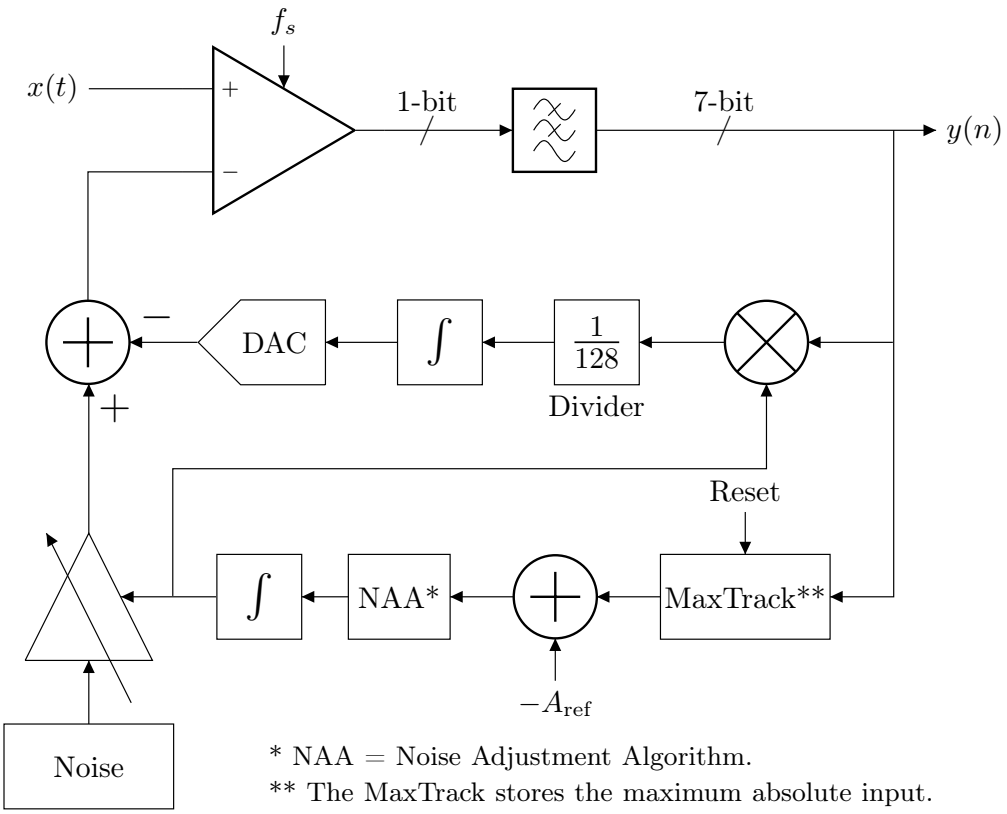


Figure 5.1.1: Top level view of the proposed system. The basic SR-ADC blocks, the noise and the comparator, are complemented with two digitally implemented feedback loops to set the system to the right operating point (the SR peak) and to compensate the threshold. The control loops are fully digital.

not be determined from this output, neither based on the added noise level. Furthermore, it is shown in the previous chapter that an incorrect size of the hysteresis can lead to either strong third harmonics when the hysteresis is too small, or strong fifth harmonic when the hysteresis is too large.

5.2 Digital Filter

The binary digital signal at the output of the comparator is filtered to reconstruct the input signal. In the derivations presented in Chapter 4, the binary output is filtered by an ideal first-order Butterworth filter. The (digital) implementation was not taken into account in the choice for this filter.

The filter specifications are not very strictly defined or elaborate. Unlike (digital) filters in e.g. telecommunication, there are no strict specifications of the attenuation in the stopband. The main purpose of the filter is to remove noise. It does not have to eliminate specific signals close to the passband. Besides the passband behavior, the main requirement is a low equivalent noise bandwidth (ENBW). To avoid an over-complicated filter design, the choice of the filter topology is largely based on simplicity, while achieving a good ENBW.

In this section, the implemented filter is presented. First, the chosen impulse response is discussed in subsection 5.2.1. In Subsection 5.2.2, the filter window and its transfer function are presented, and in Subsection 5.2.3, the implemented topology is presented. Subsection 5.2.4 discusses two techniques for reducing the complexity of the filter.

5.2.1 IIR/FIR

In digital filter design, two main filter classes exist: finite impulse response (FIR) and infinite impulse response (IIR). Both are briefly discussed below.

IIR

Infinite impulse response filters are based on analog filters and can be implemented by transforming the transfer function to the digital domain using e.g. the bilinear transform.

IIR filters can be implemented using several structures, such as direct form, cascade, parallel and lattice structures. Although the sensitivity depends on the structure, all implementations will involve multiplications, which introduce quantization errors due to parameter quantization and round-off errors.

In this project, the passband is, relative to the sampling frequency, very small. As a result, the filter coefficients necessary to implement this filter are often very close to, but not exactly an integer. Accurate implementation of these coefficients need a very high bit-length and is necessary to avoid quantization noise, parameter errors and even instability [24].

FIR

Finite impulse response filters can be designed based on the desired filter window. Depending on the window, the filter structure can be very simple. A drawback of FIR filters is a large amount of memory elements that are needed. Especially to create a low cut-off frequency, the window length should be long, which means a lot of memory is necessary. For this reason, FIR filters typically need more arithmetic operations per unit time. However, by using a multi-rate implementation, the necessary computation power can be reduced drastically. This will be discussed further in Section 5.2.3.

Implemented structure

An efficient FIR filter structure is used to implement the filter. By using a rectangular window there is no need for multiplication. Furthermore, using this window, the required memory can be reduced by employing a multi-rate implementation. Hereby, the disadvantages of the IIR filter are avoided, while the main disadvantage of the FIR filter, the large amount of memory elements, is overcome.

5.2.2 Filter Window and Transfer Function

As described above, the binary signal is filtered by a rectangular window. The rectangular window, as shown in Figure 5.2.1a, creates an equally weighted summation of multiple samples. The window is described by the following function:

$$h(n) = \begin{cases} 1 & 0 \leq n < L \\ 0 & n \geq L \end{cases}, \tag{5.2.1}$$

where, in Figure 5.2.1, $L = 16$. The response of the filter is a convolution of the window and binary input signal: $w(n) * h(n) = y(n)$. Since convolution in the time domain is equal to a multiplication in the frequency domain, the frequency response of the filter is described by the Fourier transform of the applied window. The Fourier transform of a rectangular window is the sinc function shown in Equation 5.2.2, and illustrated in Figure 5.2.1b. The plot shows the frequency response of a rectangular function of length 16. The frequency response is described by:

$$H(\omega) = \frac{\sin(0.5L \cdot \omega)}{0.5L \cdot \omega} = \text{sinc}(0.5L \cdot \omega), \tag{5.2.2}$$

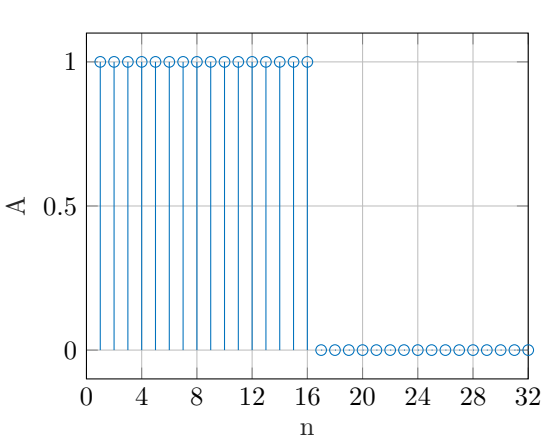
where ω is the normalized frequency ranging from $-\pi$ to π.

The phase of the output decreases linearly for increasing frequencies. As shown in Figure 5.2.1b, the frequency spectrum shows periodic behavior of $t = 32$ samples. The phase thereby decreases by 360° every $2L$ samples. The -3dB point is found for $\omega_{-3dB} \approx \frac{2.78}{L}$. Since ω is the normalized frequency the -3dB frequency is given by:

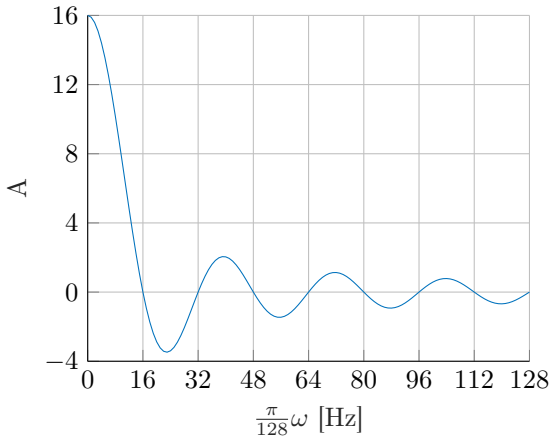
$$\begin{aligned} f_{-3db} &= \tfrac{\omega_{-3dB}}{\pi} \cdot \tfrac{f_s}{2} \\ &\approx \tfrac{2.78}{L} \cdot \tfrac{f_s}{2\pi}. \end{aligned} \tag{5.2.3}$$

Based on the required bandwidth of 200 Hz, with a sampling frequency of 2 MHz, the calculated rectangle width is 4425 samples. The implemented filter has a width of 4096 samples. The bandwidth is thereby increased to 216 Hz. The filter is implemented using a window of 4096 samples while this simple and efficient to implement. This will be discussed in the next section. The filter transfer of the implemented rectangle is shown in Figure 5.2.2. The transfer of a first-order Butterworth filter is added to the plot for comparison. The plot shows that the gain of the implemented filter just below the cut-off frequency is slightly higher, while the attenuation in the stopband is higher. The sinc-filter shows a sharper transition from passband to stopband.

In Chapter 4, it is shown that the equivalent noise bandwidth of the filter is a crucial parameter in the performance of the system. The equivalent noise bandwidth of the sinc



(a) A rectangular window of length n=16.



(b) The Fourier transform of the rectangular window from 0 to π.

Figure 5.2.1: A rectangular window and its Fourier transform.

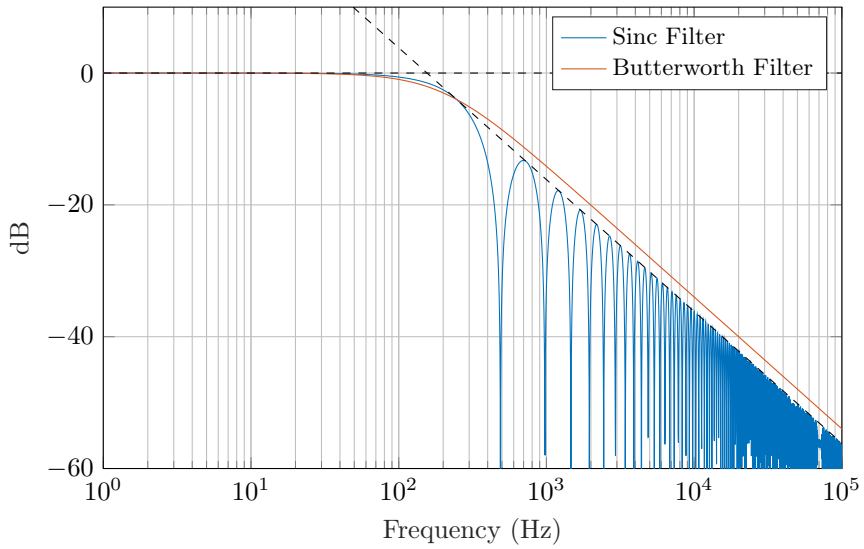


Figure 5.2.2: The frequency response of the implemented filter. The response of an ideal first order Butterworth filter with $f_{-3\text{ dB}} = 200$ Hz is added for comparison.

filter is given by the integral over the squared sinc function:

$$\int_0^\infty H(\omega)^2 d\omega = \int_0^\infty \mathrm{sinc}(0.5L \cdot \omega)^2 \mathrm{d}\omega. \tag{5.2.4}$$

The integral of the squared sinc function is given by:

$$\int \mathrm{sinc}(0.5L \cdot \omega)^2 d\omega = \frac{L\omega \mathrm{Si}(L\omega) + \cos(L\omega) - 1}{\frac{L^2}{2}\omega}, \tag{5.2.5}$$

where the function $\mathrm{Si}(L\omega)$ is defined as:

$$\mathrm{Si}(L\omega) = \int \mathrm{sinc}(L\omega)\mathrm{d}L\omega. \tag{5.2.6}$$

Evaluating the integral gives:

$$\begin{aligned} \lim_{\omega \to \infty} \frac{L\omega \mathrm{Si}(L\omega) + \cos(L\omega) - 1}{\frac{L^2}{2}\omega} &= \frac{L\mathrm{Si}(L\omega)}{\frac{L^2}{2}} \\ &= \frac{2\int_0^\infty \mathrm{sinc}(L\omega)\mathrm{d}L\omega}{L}, \end{aligned} \tag{5.2.7}$$

whereby the integral of the sinc filter is:

$$\int_0^\infty \mathrm{sinc}(L\omega)\mathrm{d}L\omega = \frac{\pi}{2}. \tag{5.2.8}$$

The equivalent noise bandwidth is thereby:

$$\begin{aligned} \omega_{ENBW} &= \frac{\pi}{L} \\ f_{ENBW} &= \frac{\pi}{L} \cdot \frac{f_s}{2\pi} \\ &\approx \frac{\pi}{2.78} f_{-3dB} \\ &\approx 1.13 \cdot f_{-3dB}. \end{aligned} \tag{5.2.9}$$

The designed filter, with a rectangle width of 4096 samples thereby has an equivalent noise bandwidth of 244.1 Hz. The bandwidth of the sinc filter is 0.777× the bandwidth of the first order Butterworth applied in Chapter 4. The performance therewith increases, but also the calculated SR optimum changes. Using the derivations in Chapter 4, the SNDR in the SR peak is increased to 27.57 dB.

5.2.3 Topology

In the time domain, the filter has a transfer of:

$$y(n) = \sum_{i=0}^{L-1} x(n-i) \tag{5.2.10}$$

Direct implementation of this function would require 4096 delay elements. The output of all elements should then be summed. By implementing the filter using a series of multiple blocks of one delay element, and a downsampler, the rectangular window is build more efficiently [24].

The basic filter structure of three stages is shown in Figure 5.2.3. In this implementation of the rectangular window, downsampling is applied within the filter structure. The output of the each stage is given by:

$$\begin{aligned} w_1[k] &= x[2k-1] + x[2k-2] = \sum_{r=1}^{2} x[2k-r] \\ w_2[l] &= w_1[2l-1] + w_1[2l-2] = \sum_{p=1}^{2} w_1[2l-p] \\ y[m] &= w_2[2m-1] + w_2[2m-2] = \sum_{q=1}^{2} w_2[2m-q] \end{aligned}$$

The full impulse response is thereby:

$$\begin{aligned} y[m] &= \sum_{q=1}^{2} w_2[2m-q] \\ &= \sum_{q=1}^{2}\sum_{p=1}^{2} w_1[2(2m-q)-p] \\ &= \sum_{q=1}^{2}\sum_{p=1}^{2}\sum_{r=1}^{2} x[2(2(2m-q)-p)-r] \\ &= \sum_{r=1}^{8} x[8m-r]. \end{aligned} \tag{5.2.11}$$

In general form, the impulse response is:

$$y[m] = \sum_{o=1}^{L} x[L \cdot m - r], \tag{5.2.12}$$

where the window length L is given by $2^{\#\text{stages}}$. Equation 5.2.12 shows that, as a result of the downsampling, for every L input samples, only one output sample is created. To create a window length of 4096 samples, the filter is implemented using 12 stages. The sampling frequency at the output of the filter is thereby 488.28 Hz.

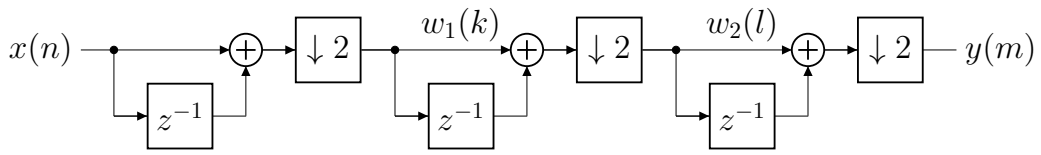


Figure 5.2.3: Three stages of the implemented digital filter.

5.2.4 Dataflow

The input of the filter is binary. However, each stage is a summation of two states. Thereby the required dynamic range doubles every stage and one bit should be added to the signal. Since the filter contains 12 stages, the output would become a 13-bits wide. The accuracy of a 13-bit signal is orders of magnitude larger than the accuracy of the system. Therefore the bit-length can be reduced in the last stages of the filter. Two techniques are applied to reduce the bit-length of the output signal.

Division by 2

In the last five stages of the filter, a divider is implemented in the adder. Since the divider rounds the LSB, an error is introduced at each divider. The expected error introduced by each divider is when the divider is implemented by throwing away the LSB:

$$E(\epsilon_{\text{div}}) = -\frac{1}{2}\text{LSB}. \tag{5.2.13}$$

In each successive stage, the error of each rounding doubles. The expected error introduced in each divider thereby is:

$$E(\epsilon_{\text{div},n}) = 2^{n-1} \cdot \left(-\frac{1}{2}\right) \text{ LSB} \tag{5.2.14}$$

One output filter sample consists of 16 output samples of the 8th stage, 8 output samples of the 9th stage, 4 of the 10th stage, 2 of the 11th stage and 1 of the 12th stage. The total error introduced by the divisions is thereby:

$$\begin{aligned} E(\epsilon_{\text{div,tot}}) &= \sum_{n=1}^{5} 2^{5-n} \cdot E(\epsilon_{\text{div},n}) \text{ LSB} \\ E(\epsilon_{\text{div,tot}}) &= -40 \text{ LSB}. \end{aligned} \tag{5.2.15}$$

The error of 40 LSB is the expected DC-error, and can thus be compensated. The standard deviation of the introduced error appears at noise added to the system. This can be calculated using the variance, which is defined as:

$$\text{Var}(\epsilon_{\text{div}}) = E(\epsilon_{\text{div}}^2) - E^2(\epsilon_{\text{div}}). \tag{5.2.16}$$

The variances in each stage are:

$$\begin{aligned}\mathrm{Var}(\epsilon_{\mathrm{div},n}) &= E(\epsilon_{\mathrm{div},n}^2) - E^2(\epsilon_{\mathrm{div},n}) \text{ LSB} \\ &= \frac{\left(2^{n-1}\right)^2}{2} - \left(2^{n-1}\cdot\frac{-1}{2}\right)^2 \text{ LSB} \\ &= \frac{1}{4}\cdot 2^{2n-2} \text{ LSB}.\end{aligned} \tag{5.2.17}$$

The total variance is the sum of all contributions of each stage to the output:

$$\begin{aligned}\mathrm{Var}(\epsilon_{\mathrm{div,tot}}) &= \sum_{n=1}^{5} 2^{5-n}\cdot\mathrm{Var}(\epsilon_{\mathrm{div},n}) \text{ LSB} \\ \mathrm{Var}(\epsilon_{\mathrm{div,tot}}) &= 124 \text{ LSB}.\end{aligned} \tag{5.2.18}$$

The standard deviation of the expected error is thereby 11.1 LSB, which corresponds to 0.271% of the total dynamic range. From Chapter 4, the expected error is 0.02 for a total dynamic range of 2, which corresponds to 1% of the dynamic range. The quantization error due to rounding is thereby small compared to the already existing error.

Limiting the Word Length

After the 5th stage, the word length is increased to 6 bits, since the word length increases by one bit in every stage. The output of the fifth stage is equal to the sum of $2^5 = 32$ successive samples of the comparator output, giving a maximum value of 32. The MSB of the 6-bit word is only used when the output of the filter is 32. Higher values are not possible and for lower values, a 5-bit representation suffices.

The stochastic nature of the bit-stream at the output of the comparator can be used to reduce the word length. The maximum output is 0.6 for a dynamic range of -1 to 1, as shown in the previous chapter. The probability of a high output is in that case 0.8, since $0.8\cdot 1 + 0.2\cdot -1 = 0.6$. The probability of the maximum value of signal w for different word lengths can be calculated by:

$$P(w = 2^n) = 0.80^{2^n}$$

The probability that the MSB is equal to 1 drops after every stage. The probability that the output of the fifth stage is 32 is only 0.079%. The integer range of 0-32 can therefore be limited to 0-31 in the fifth stage. When the output is 32, this value is reduced to 31. This introduces an error of 3.1%. The total expected error caused by removing the MSB in this way is $0.00079\cdot 0.031 = 0.00249\%$ of the full dynamic range.

5.2.5 Conclusion

The full filter design is shown in Figure 5.2.4. By implementing the filter in a multi-rate fashion, the total number of memory elements is reduced from 4095 to only 78. Further optimization is implemented to further reduce the required number of memory elements. By rounding and saturating the signal, only 61 memory elements are required to build the filter. Since rounding is implemented by throwing away one bit, no extra logic is required. The saturation algorithm can be implemented by simple logic elements. Apart from the

savings of required memory, also the summing blocks are smaller since the bit length is reduced. In Appendix D, the implementation of the filter in Verilog is presented.

The output signal of the filter is a 7-bit signal with a sample frequency of 488.28 Hz. The 7-bit resolution is high enough to display the calculated 27.3 dB SNDR.

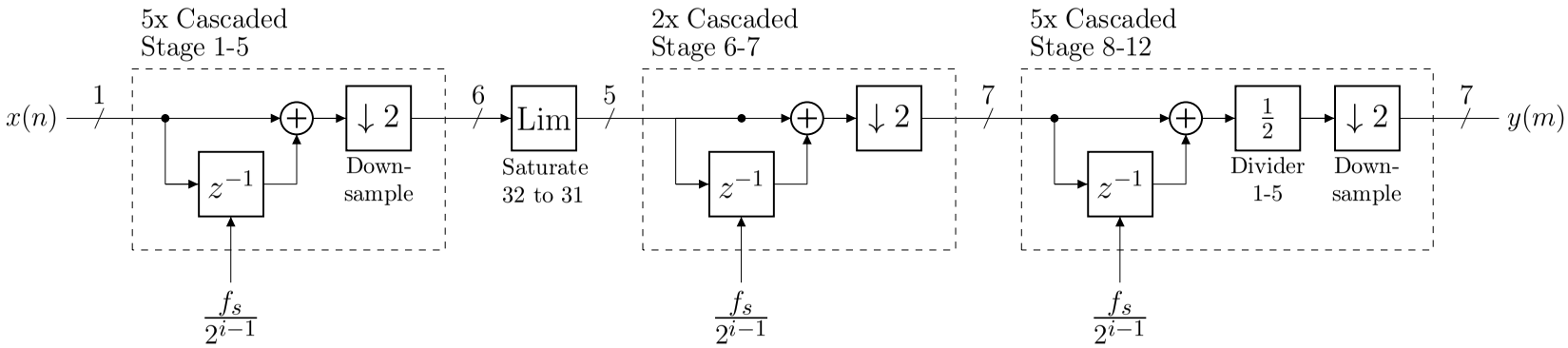


Figure 5.2.4: Implemented 12-stage sinc filter. In each state, the sample frequency is divided by 2. a limiter is applied after the fifth state which saturates the input value 32 to 31. Furthermore, in the final five stages, a divider is is applied to round the LSB. The sampling frequency of each successive stage is halved. The sampling rate of stage i is described by $f_s/(2^{i-1})$, whereby f_s is the sampling frequency of the comparator.

5.3 Dynamic Range Control

The dynamic range settling circuit is designed to ensure that the SR-ADC operates in the SR peak. Dynamically adjusting the noise improves the system in two ways. First of all, the noise that is already existing in the system and the signal can be used beneficially. By using feedback, the total noise is measured and controlled, and not only the added noise. Secondly, the system can handle a wider input amplitude range. Since, the amplitude of the input signal can vary significantly, analog front-ends for ECG applications are typically designed for an input amplitude ranging from 1 $mV_{p\text{-}p}$to 10 $mV_{p\text{-}p}$. By adjusting the noise level based on the output signal, the system is also kept in the SR-peak independent of the input amplitude.

This section is built up as follows. First, the basic design of the noise control signal is examined, followed by an analysis of the loop gain and stability. Based on the observations in the first sections, the design is optimized by adjusting the loop gain and introducing different attack and release times. Finally, the simulation results of the implemented feedback loop are presented.

5.3.1 Basic Design - Design Approach

This subsection presents the functional design of the feedback path. The basic building blocks are discussed and the functionality is implemented. Furthermore, the simulation set-up and a quick calculation of the required resolution are discussed.

Overview

The feedback path, shown in Figure 5.1.1, consists of 3 main building blocks. The first is a maximum value tracker. This block saves the maximum absolute output of the digital filter and compares this value to a desired value.

This tracker is periodically reset. The period of this reset is greater than the period of the minimum signal frequency, 0.5 Hz for this project. However, during the design, a test signal of 5 Hz is used. To speed up the simulations, the reset frequency is set to 3.81 Hz.

Figure 5.3.1 shows a simplified example of the feedback signals in time. The MaxTrack signal saves the peak value. The difference between the measured peak and a reference value (E_n) is integrated to create the noise control signal. The second building block is the digital integrator. The integrator is implemented by an up-down counter, which adds the measured error (E_n) to the previous value of the counter. When the noise increases, the expected output signal decreases, such that a negative feedback is created and the output amplitude will settle at the reference value.

The third block is the Noise Adjustment Algorithm, as shown in Figure 5.1.1. This algorithm is not shown in the simplified example above.

In the design of this loop, several issues have to be taken into account. First of all, the accuracy. The resolution of the integrator which controls the noise source is limited. The real value will always differ slightly from the ideal value, causing a small shift from the SR peak.

Besides a small shift from the SR peak, a feedback loop should always be designed with the loop gain and stability in mind. If the gain of the loop is too high, stability issues can arise. The output of the counter can start to show some ringing, caused by overcompensation. This results in periodic variations in the gain of the amplifier. On the other hand, a low

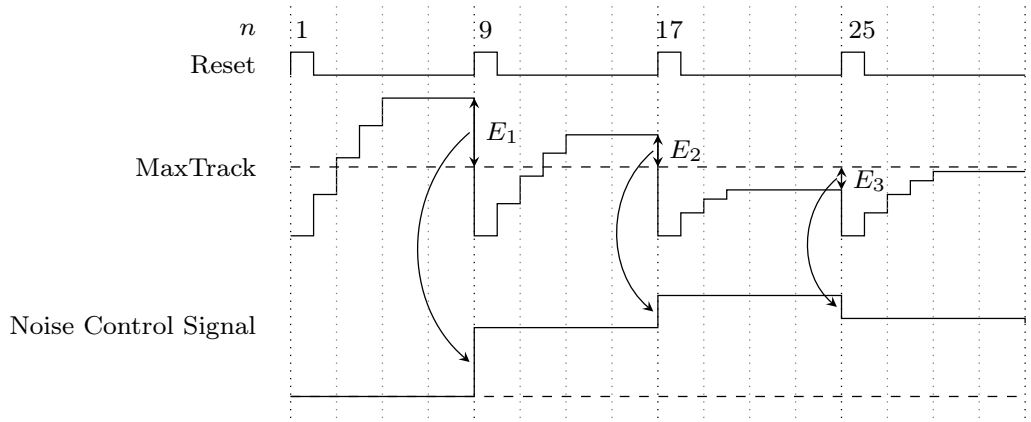


Figure 5.3.1: An illustration of the feedback path signal.

loop gain results in a slow convergence of the loop. The Noise Adjustment Algorithm is implemented for reducing the two latter issues.

Output Signal Feature

The noise level can only be controlled based on (a feature of) the output signal. In Chapter 4, it is shown that input signal shape influences the optimum noise level.

Various features of the output signal can be used to control the feedback loop. Considered output features are the average (absolute) output value, maximum and minimum values, and the output power. For robust design, it is important to choose the output feature such that optimal performance is ensured for all input signals.

The SR peak is determined for a single-tone and a two-tone input signal in Chapter 4. The values of considered output features at the SR peak of both signals are shown below. The output amplitude shows the smallest change for the two different inputs. Even when considering that the output power changes quadratically with respect to a change in x/σ_η, the dependency of the ideal output amplitude to the shape of the signal is the smallest of the considered output features. Therefore, the output amplitude is used as the output feature to control the noise.

Although the SNDR optimum shifts to a higher output amplitude for a two-tone input, the error at the peaks increases rapidly due to the increased non-linearity. By controlling the input noise level based on the output amplitude, the maximum error due to the non-linearity in the transfer is limited. The SNR is in this case slightly lower than the calculated optimum, but large errors at high output amplitudes are avoided.

The maximum output amplitude is set to 0.60. It should be noted that this is slightly smaller than the 0.62 mentioned in the table. This difference is caused by the harmonic distortion. The table shows the output amplitude of the linearized output, $\hat{H}(x(t), \sigma_\eta) \cdot x(t)$. The actual maximum output is smaller due to harmonic distortion. The expected output value for the optimal noise value is calculated by:

$$\hat{w}_{\max}(x, \sigma_\eta) = \operatorname{erf}\left(\frac{X_{\max}}{\sqrt{2} \cdot \sigma_\eta}\right) = \operatorname{erf}\left(\frac{1}{\sqrt{2} \cdot 1.19}\right) = 0.60.$$

	Output Amplitude	Output Power	Average Absolute Output
Single- tone	0.62	0.192	0.395
Two-tone	0.68	0.116	0.276
Difference (in %)	9.7%	65.5%	43.1%

Table 5.1: Different output features of the sine wave input and the two-tone input at the SR peak. The bottom row compares the value of each output feature of the single-tone to the two-tone input signal.

Simulation Set-Up

A Simulink simulation is performed with an input signal of 5 Hz. The cut-off frequency is, as designed, 216 Hz at a sample rate of 2 MHz. The output of the digital filter is a positive unsigned bit-value. The output is transposed to a signed integer by subtracting the offset, 64, from the filter output. The absolute value of this signed signal is used as input for the next block. This block compares the input to its output, stores the highest value and sets the output to this value.

The reference amplitude A_{ref} is subtracted from the stored maximum output. A_{ref} is set to 38, based on a scale from 0 to 127. This corresponds to an amplitude of 0.60 in a range from -1 to 1. The result of this subtraction (E_n) is sampled right before the reset of the maximum value tracker. The signal E_n, from this point referred to as output amplitude error, is send to the up-down counter. First, the system is simulated without NAA. The output of the up-down counter is used as input of the variable gain amplifier. The variable gain amplifier is simulated by a multiplier with the analog noise value and the noise control signal as inputs. A screen-shot of the Simulink model is presented in Appendix B.

First, the simulation is performed with an input amplitude of 1. A reference simulation, without feedback path, shows an SNDR of 27.35 dB, which corresponds with the SNDR predicted in the previous section. Adding the feedback path does not reduce the SNDR significantly. After settling, the SNDR is 27.30 dB. In Figure 5.3.2, the output as well as the noise control signal are plotted. The system is settled to 95% after 10 iterations. This shows that the basic functionality of the feedback loop and the simulation set-up works.

Resolution

The finite resolution of the noise source control causes a deviation from ideal SR-peak. While the input amplitude range can vary from 1 mV to 10 mV, the minimum step of the noise adjustment relative to the input amplitude increases when the input amplitude is small. For a 7-bit resolution, each LSB is equal to $\frac{1}{128}$ of the total noise range. If the input signal amplitude is 1 mV, the noise level settles at 1/10 of the total range, thus around 12 or 13. An adjustment in the noise level of 1 bit results changes the gain by 8%. This large relative step size limits the accuracy of the noise control.

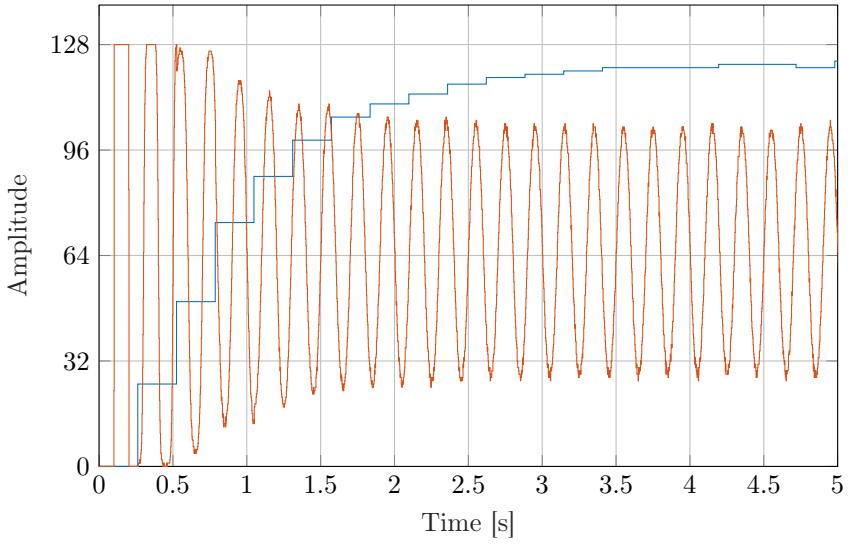


Figure 5.3.2: Output of noise control simulations. The output signal as well as the control signal are shown. The input signal is 5 Hz, and has an amplitude of 1. The control signal is updated every 3.81 Hz.

5.3.2 Loop Gain and Stability

Based on a measured output amplitude error E_n, the noise level, and thus the output amplitude is changed (ΔA_{in}). For stable operation, ΔA_{in} may not be too large. The amplitude error in sample n consists of the error in the previous sample plus change in output amplitude:

$$E_n(n) = E_n(n-1) + \Delta A_{\text{in}}.$$

One sample, n, is hereby one cycle of the noise control loop (in this case 3.81 Hz). When ΔA_{in} is larger than the measured error $E_n(n-1)$, the adjustment creates a new error $E_n(n)$ of opposite sign. This can cause instability in the system and ringing of the noise control level. If $\Delta A_{\text{in}} > 2 \cdot E_n$, the output amplitude error increases every cycle and the system oscillates.

The stability of the system is analyzed by determining the loop gain of the noise control loop. The loop gain is determined by the change in amplitude divided by the measured error: $\Delta A_{\text{in}}/E_n$.

The multiplier that controls the noise level can be translated into a gain as well. In a linearized model, increasing the noise with a factor X, will decrease the output amplitude with the same factor X. This means that:

$$\frac{E_n}{\text{NC}} = \frac{\Delta\text{NC}}{\text{NC}} = \frac{\Delta\sigma_\eta}{\sigma_\eta} = \frac{-\Delta A_{\text{in}}}{A_{\text{in}}} = \frac{-\Delta A_{\text{out}}}{A_{\text{out}}}, \tag{5.3.1}$$

whereby NC digital signal controlling the noise level and A_{out} is the amplitude at the output of the filter. This equation can be rewritten to determine the loop gain:

$$\frac{\Delta A_{\text{out}}}{E_n} = \frac{-A_{\text{out}}}{\text{NC}}. \tag{5.3.2}$$

Input amplitude A_{in}	$\Delta A_{\text{out}}/E_n$
1	$-38/127 = -0.30$
1/4	$-38/31.5 = -1.2$
1/7	$-38/18.1 = -2.1$
1/10	$-38/12.7 = -3.0$

Table 5.2: Loop gain for different input amplitudes.

Amplitude A_{in}	SNDR
1	27.30 dB
1/4	26.94 dB
1/7	24.56 dB
1/10	7.43 dB

Table 5.3: SNDR for different input amplitudes. Due to spurious tones caused by the feedback network, the SNDR drops for lower input amplitudes.

When the system is in the right operating point, the output amplitude is equal to 38. The noise control level for which this occurs is determined by the input signal level. The noise source is designed to deliver the right amount of noise for an input amplitude of 1 when the multiplier is set to 127. The noise control level decreases proportionally to the input amplitude. In Table 5.2, the loop gain for different input amplitudes is shown. The loop gain increases for lower input amplitudes. For $A_{\text{in}} = 1/4$, the noise control is expected to show a small overshoot, after which the system will stabilize. For an input amplitude of $1/7$, the loop gain is approximately -2. This would mean that every cycle the sign of E_n will reverse, but the error will not decrease. Reducing the input amplitude to $1/10$ results in a loop gain of -3. The system will not be stable and large ringing is expected in the noise control signal.

The effect of the loop gain is shown in Figure 5.3.3. The noise control signal is shown for simulations with different input amplitudes. The plot confirms the calculations. A clear stable behavior occurs for $A_{\text{in}} = 1$ and $A_{\text{in}} = 1/4$. For $A_{\text{in}} = 1/7$ the noise control signal has a clear overshoot, followed by some ringing. However, around 4 seconds it stabilizes. The real loop gain is thus slightly smaller than calculated. The same holds for $A_{\text{in}} = 1/4$, where the expected overshoot does not occur. For$A_{\text{in}} = 1/10$, the system is clearly unstable. The noise control signal keeps ringing without convergence.

The calculated approximations overestimate the loop gain. The loop gain is based on a linearization of the system in the desired operating point, and the non-linearities in the system reduce the loop gain when the system is not at its correct operating point.

The behavior of the noise control is reflected clearly in the simulated SNDRs, which are shown in Table 5.3. The SNDR loss is mainly caused by spurious tones as a result of the periodic behavior in the feedback path. These tones are observed at $f_{in} + f_s/2$ and $f_{in} - f_s/2$, where f_s is the sample frequency of the up/down-counter. The periodic movements of the feedback loop cause a periodic difference in gain, similar to the effect of aliasing in sampled systems.

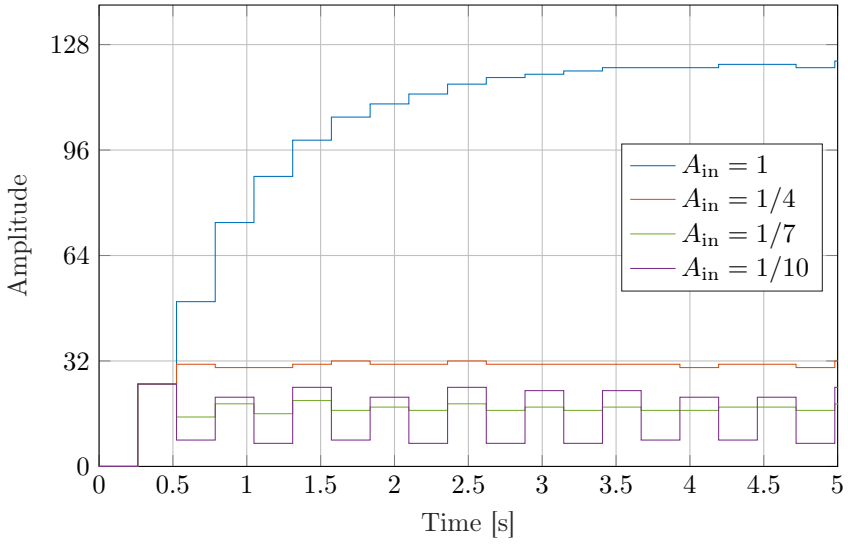


Figure 5.3.3: Simulated noise control signal, for different input amplitudes (A_{in}). The input signal is 5 Hz. A large input results in slower settling. For small inputs the loop becomes unstable.

5.3.3 Exponential Noise Level Control

In Equation 5.3.2 the loop gain is defined as the ratio between the input amplitude and the noise control signal. The loop gain depends on the input amplitude. Ideally, the loop gain is independent of the the input amplitude. To achieve this, the control of the variable gain amplifier is implemented exponentially. The noise level thereby depends on the noise control level as follows:

$$A_{\text{in}} \propto \frac{1}{\sigma_\eta} \propto \frac{1}{e^{\text{NC}}}.$$

If the noise control signal changes with ΔNC, the

$$\frac{\Delta\sigma_\eta}{\sigma_\eta} \propto e^{\Delta\text{NC}}$$

A certain ΔNC, will always lead to the same percentage difference in noise change, independent of the noise control level. Therefore the change in the output of the digital filter is proportional to ΔNC. The exponential function is implemented by $e^{x/26}$. The factor 26 is chosen since $e^{127/26} = 132$. The output range of the noise multiplier is approximately the same as in the previous simulations for the same bit-length of the noise control signal. The noise level increases by 3.9% per step in the noise control signal, since $e^{1/26} = 1.039$. A measured output amplitude error of 1 bit results in an output change of 3.9%. The measured error, however, is $1/38 = 2.6\%$. The loop gain is thereby approximately 1.5.

The noise control signals for different input amplitudes are shown in Figure 5.3.4. The plot shows that the settling speed is at first limited by saturation of the output. All noise control signals start with the maximum E_n. This is determined by the maximum output minus the reference, $64 - 38 = 26$. When the system reaches its linear region, the system settles fast to the correct noise value. This corresponds to the relatively high loop gain. The simulated SNDRs of the different input amplitudes are shown in Table 5.4. This table shows

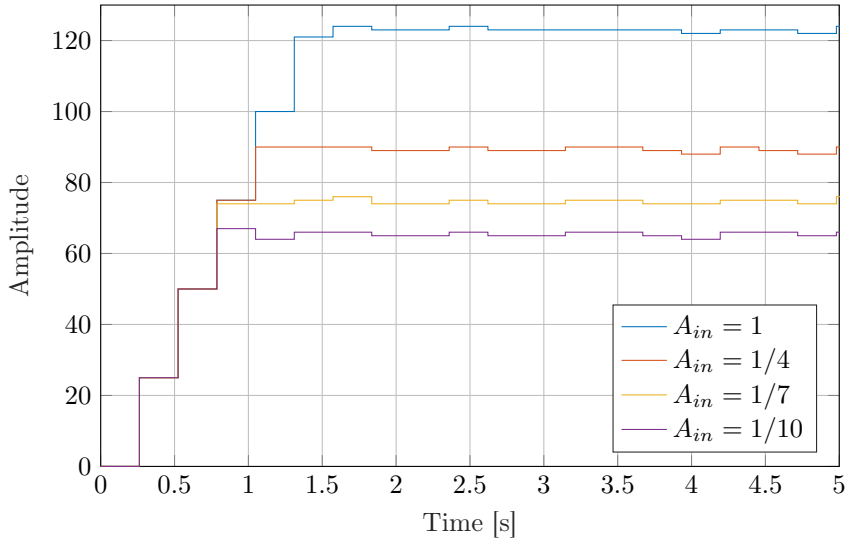


Figure 5.3.4: Noise control signal for different input amplitudes. The noise level control is implemented exponentially. The input signal is 5 Hz, the control signal is updated at 3.81 Hz.

that performance is independent of the input amplitude. However, a significant SNDR loss is observed when comparing the results with the SNDR of 27.35 dB obtained without the feedback path. This error is mainly caused by (small) random fluctuations in the maximum amplitude, which can also be observed in the plot in Figure 5.3.4.

5.3.4 Noise Adjustment Algorithm

The simulation results in the previous subsection show some small random fluctuations in the noise control signal. To avoid these small fluctuations, an algorithm is implemented which reduces the sensitivity when the measured output amplitude shows a small deviation from the reference amplitude A_{ref}.

The algorithm works as follows. When the absolute measured output amplitude error $|E_n| > 3$, the output of the algorithm (ΔNC) is equal to E_n. If E_n is equal to either 3 or -3, ΔNC is 1 or -1 respectively. For $|E_n| < 3$, the noise level is not adjusted, so $\Delta\text{NC} = 0$.

The simulation results of the control loop with the added noise adjustment algorithm are shown in Table 5.4. The SNDR is increased significantly compared to the system without this algorithm. The settling accuracy is slightly compromised since small output amplitude errors are ignored. However, the reduction of random fluctuations in the noise control signal has a more important effect on the SNDR. The noise control signals are again plotted, in Figure 5.3.5. The plot shows that the settling self is not affected by the algorithm, but when the system is settled, the noise control signal is much more stable.

Amplitude A_{in}	SNDR linear sized noise control	SNDR exponentially sized noise control	SNDR with NAA
1	27.30 dB	26.68 dB	27.20 dB
1/4	26.94 dB	26.66 dB	27.21 dB
1/7	24.56 dB	26.49 dB	26.96 dB
1/10	7.43 dB	26.46 dB	27.14 dB

Table 5.4: SNDR for different input amplitudes. The addition of the divider improves the dynamic range of the system drastically.

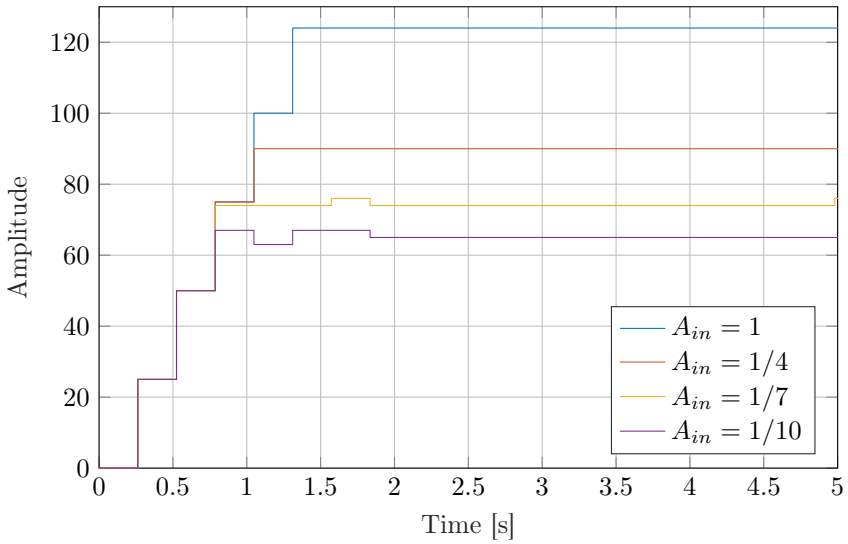


Figure 5.3.5: Noise control signal for different amplitudes. Simulations of the system with algorithm to reduce the sensitivity for low variations of the output amplitude.

5.4 Offset Compensation

Since the comparator input signal is very small, the threshold level should be set accurately. The performance of the SR-ADC is maximized when the threshold is set to the DC level of the input signal. Current state-of-the-art comparators easily have offset errors in the order of several millivolts, while the input signal of the implemented system can be as small as 1 mV.

To address this issue, the offset needs to be compensated. Two techniques are considered to implement the offset. First of all, an offline calibration can be performed. The offset is in this case calibrated when no input is applied. Once the calibration is performed, the compensation is held at the same value.

A second method would be implementing an integrating negative feedback loop. This creates a highpass filter, and the DC is filtered out. This technique would not only compensate static comparator threshold but compensates for all (differential) low-frequency signals in the system.

In this section, the design of the offset compensation is presented. First, the effect of a non-zero threshold on the SNDR is discussed in Subsection 5.4.1, to determine the minimum required accuracy of the threshold calibration. Secondly, a linear model of the system is derived to calculate the frequency behavior of the highpass filter in Subsection 5.4.2. A digital implementation of the feedback path, addressing the limited resolution is presented in Subsection 5.4.2. In Subsection 5.4.2, the dependency of the offset compensation on the input amplitude is addressed. Finally, simulations results are presented in Subsection 5.4.2.

5.4.1 Offline Offset Calibration Accuracy

The resolution of the calibration determines the maximum expected residual offset. To determine the minimum required resolution, the effect of a non-zero threshold on the SNDR is calculated.

The analysis is performed using the method presented in Chapter 4, implementing the offset by adding a constant to the input signal. The effect on the performance is shown in Figure 5.4.1 and in Table 5.5. The SNDR is calculated using the method in presented in Chapter 4. The SNDR is thereby slightly lower than in the previous sections since the implemented filter is a first-order Butterworth filter. Alongside the SNDR, the ideal noise value, and the maximum expected output is given. For an offset of 5% of the peak-to-peak input amplitude, the SNDR is decreased 0.65 dB. In the implemented SR-ADC, this would translate to an accuracy of 50 μV, since the minimum peak-to-peak amplitude is 1 mV.

5.4.2 Dynamic Offset Compensation

For dynamic offset calibration, a more comprehensive analysis is performed. The offset compensation signal will couple into the signal path directly. Therefore a small signal analysis is performed to determine the behavior of the system.

Linear model

The integrator in the feedback path introduces a high-pass characteristic in the system. To determine the frequency behavior of the system, a linearized, continuous time, small-signal model is created. The model is shown in Figure 5.4.2.

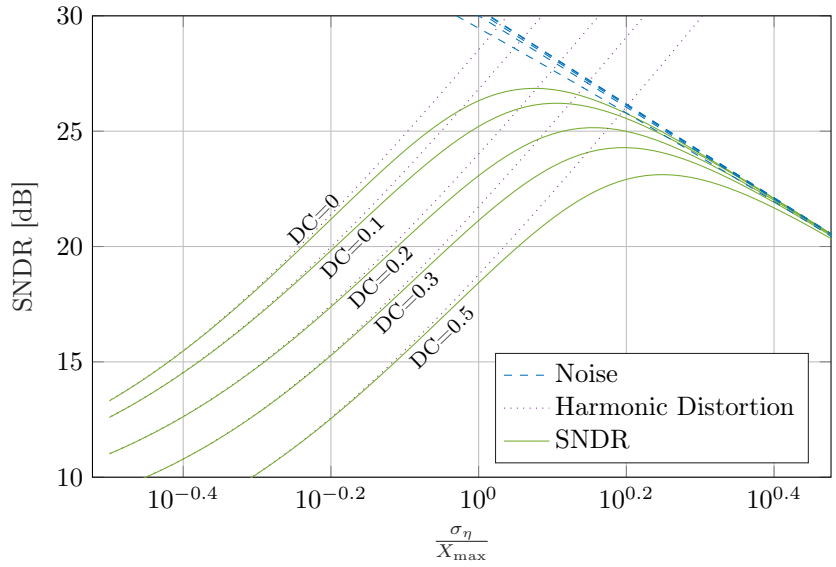


Figure 5.4.1: SNDR for different DC bias levels. The calculations are based on a a sinusoidal input with $A_{\text{in}} = 1$.

Offset	SNDR	Noise level, σ_η	$\hat{y}_{\text{out,max}}$
0	26.86 dB	1.19	0.60
0.01	26.85 dB	1.19	0.60
0.02	26.82 dB	1.19	0.61
0.05	26.67 dB	1.22	0.61
0.1	26.21 dB	1.28	0.61
0.15	25.67 dB	1.35	0.61
0.2	25.15 dB	1.43	0.60
0.3	24.29 dB	1.58	0.59
0.5	23.12 dB	1.78	0.60

Table 5.5: The SNDR, noise level and expected output amplitude for different offset levels in the SR-peak. Calculations are based on a sinusoidal input with $A_{\text{in}} = 1$.

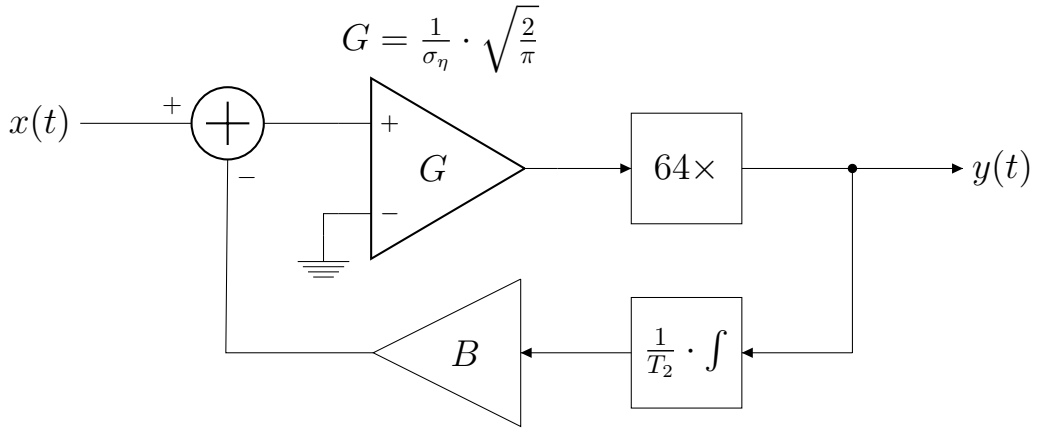


Figure 5.4.2: Linearized, analog equivalent, system model with offset compensation loop. A low-pass feedback loop is added to obtain a high-pass system behavior.

A small-signal analysis, performed on a linearized model of the system, shows the frequency characteristics of the filter. The comparator is modeled as an amplifier with gain $G = \sqrt{2/(\pi\sigma_\eta^2)}$, based on the derivative of the transfer function at $x = 0$. The digital filter is modeled as a gain of 64 since the transfer in the passband is considered 1 and the range shifts from $< -1, 1 >$ to $< 0, 127 >$. The digital filter also adds an offset of ~ 64. Apart from the step response calculations, the digital filter output is mapped to $< -64, 63 >$, to compensate for this offset. The up-down counter is modeled as an integrator, with a time constant of $1/T_2$, to model the sampling time of the up-down counter. Hereby is $T_2 \approx 1/488$ Hz. A (digital) gain block, B, is added to the loop to tune the cut-off frequency.

The transfer function of this model is:

$$\frac{Y(s)}{X(s)} = \frac{64G}{1 + 64G \cdot B \cdot \frac{1}{T_2} s^{-1}} \tag{5.4.1}$$

For an input amplitude of 1, the noise level $\sigma_\eta = 1.2$ and the gain $G = 0.66$. Since $s = j\omega$, the gain B can be calculated for a cut-off frequency of $f_{-3\text{dB}} = 0.5$ Hz:

$$\frac{1}{B} = 64G \cdot \frac{1}{T_2} \cdot \frac{1}{2\pi f_{-3\text{dB}}} \approx 6560 \tag{5.4.2}$$

Sinusoidal Input To confirm the model, a system simulation is performed with $x(t) = \sin(2\pi \cdot 5t)$ and $x(t) = \sin(2\pi \cdot 0.5t)$. The simulations are performed using a fixed (optimal) noise level.

The results are plotted in Figure 5.4.3. For $f_{\text{in}} = 5$, the RMS output is 26.7, the amplitude is 38 and the SNDR is 27.11 dB. The THD is -32.14 dB and the SNR is equal to 28.75 dB. The threshold compensation does not show a significant deterioration of the performance. The output amplitude is reduced by approximately 5% due to the high-pass filter, since $1/\sqrt{1 + 0.1} = 0.95$.

The second simulation, with $f_{\text{in}} = 0.5$, shows the transfer at the cut-off frequency. The simulated RMS is 19.9, which is close to the calculated value: at the cut-off frequency, the RMS value is expected to be $39/\sqrt{2}/\sqrt{2} = 19.5$.

Since the output power decreases, the relative output noise increases. The harmonic distortion however decreases, resulting in the following performance: SNDR = 25.56 dB,

THD = -36.82 dB, SNR = 25.90 dB.

Step Response Figure 5.4.4 shows the step response of the system. The output is described by:

$$y(t) \approx 64G \cdot e^{-(t-0.1)/\tau} + 64 \tag{5.4.3}$$

The output is plotted from the uncapped output, with a range of $< 0, 127 >$. The DC level of 64 is therefore added to the output. The time constant τ can be derived for $t - 0.1 = \tau$:

$$\begin{aligned} y(t) &\approx 64G \cdot e^{-1} + 64 \\ y(t) &\approx 80 \end{aligned} \tag{5.4.4}$$

From the plot, can be read that $y(t) = 80$ at $t \approx 0.43$, and thus $\tau = 0.33$. Using the time constant, the cut-off frequency of the filter is derived:

$$f_{\text{-3dB}} = \frac{1}{2\pi\tau} \approx 0.48 \text{ Hz}. \tag{5.4.5}$$

Compensation Accuracy and Divider Implementation

The system as proposed above does not take into account the limited resolution of the digital computations, nor the resolution of the digital-to-analog conversion. In the simulation, the gain block B is implemented using a double data type.

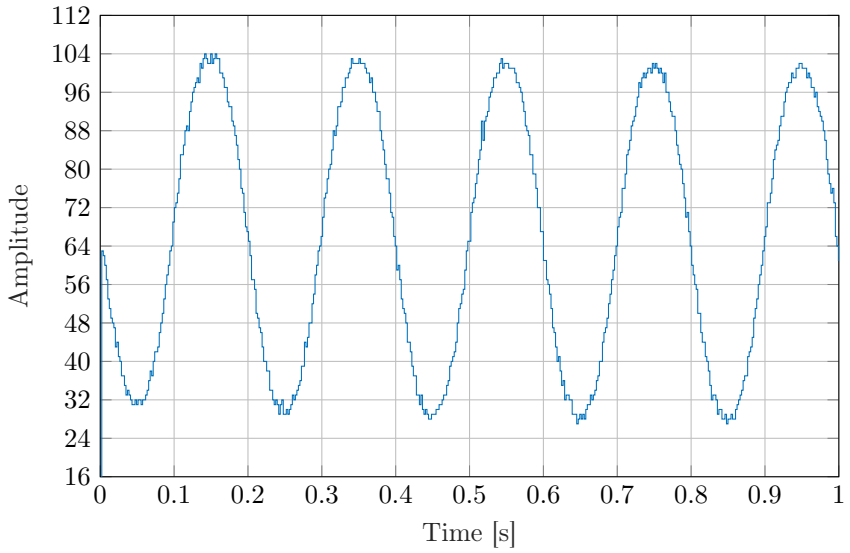
To create a precise system-level model and to determine the exact implementation specifications, implementation limitations are included in the system level design. The first and most fundamental limitation is the digital-to-analog conversion step, whereby the digital feedback signal is translated into an adjustment in threshold voltage. The analysis is performed using the same resolution as presented in Section 5.4.1. An 8-bit DAC is implemented, the maximum compensation is thereby ±6.4 mV. The minimum step is 50 μV, which is 0.5% of the maximum peak-to-peak amplitude.

The step size complies with modeling the A/D conversion by a gain block of 1/100 linearized continuous time simulation. The simulated input signal amplitude ranges from 0.1 to 1. The maximum peak-to-peak input amplitude is thus 2. A 1-bit step should thereby create a change in the threshold of $2 \cdot 0.5\% = 0.01$.

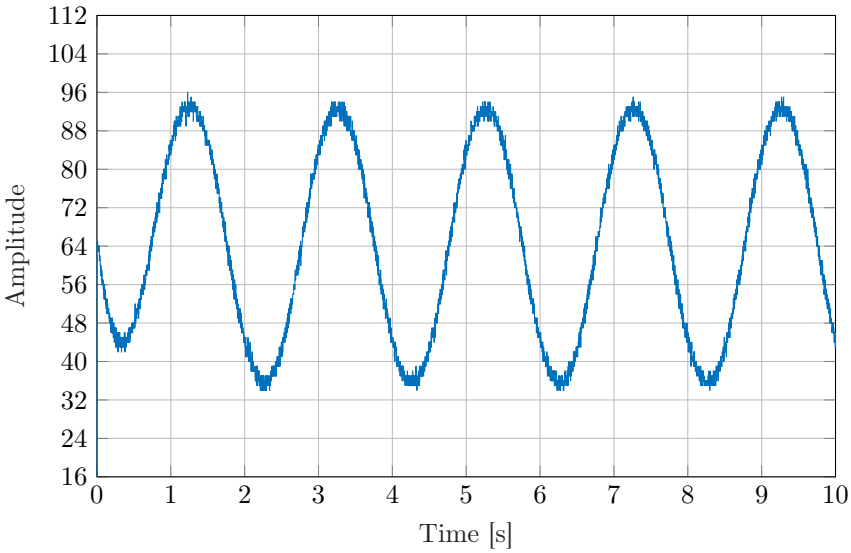
To implement the total division of 6560, the residual division of 65.6 should still be implemented. This value is approximated by a 64× divider. The divider is implemented by rounding 6 LSBs of the digital value and is placed behind the up-down counter.

Input Amplitude Sensitivity

The analysis performed above only considers an input amplitude of 1. When the amplitude is decreased, the gain G of the system increases. This alters the transfer function shown in Equation 5.4.1 in two ways. First of all, the numerator, and thus the passband gain, increases. Secondly, the denominator, and thereby the cut-off frequency, changes inversely proportional to G. To compensate for this, the gain B can be controlled by the input amplitude. While the absolute value of the input amplitude is unknown, an approximation can be made based on the added noise power.



(a) $f_{\text{in}} = 5$ Hz



(b) $f_{\text{in}} = 0.5$ Hz

Figure 5.4.3: Simulated output of the system with threshold compensation for a sinusoidal input. The input amplitude is 1, and the noise level $\sigma_\eta = 1.19$. The plots clearly show the effect of the implemented high-pass filter. The reduced output amplitude corresponds with the linearized model.

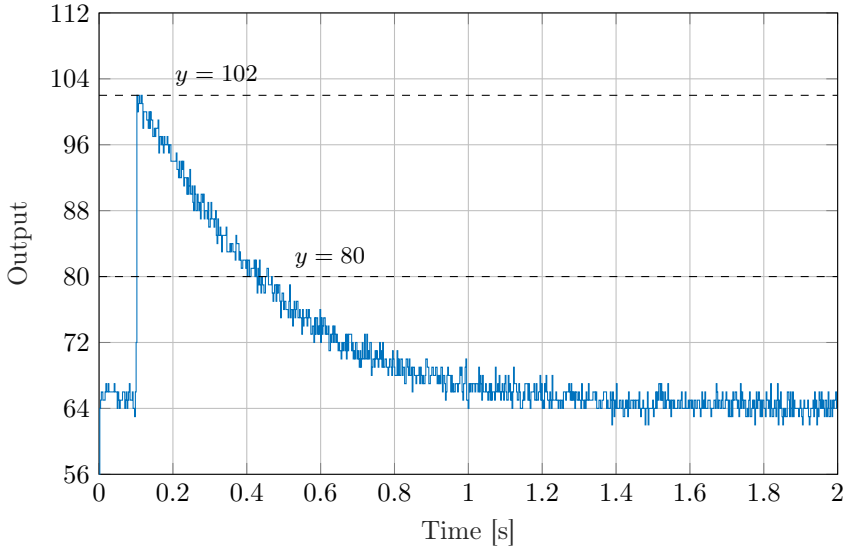


Figure 5.4.4: Step response of the system with threshold compensation. The plot shows that the RC-time of the system is 0.33 s, corresponding with a cut-off frequency of 0.48 Hz.

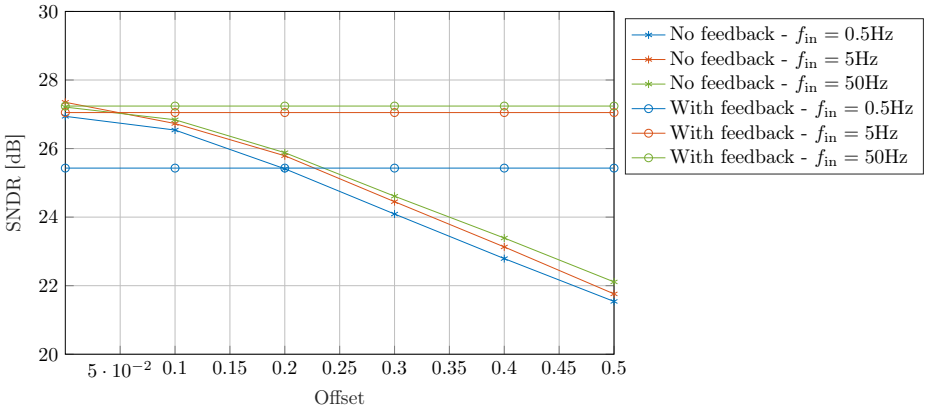
The gain of the feedback loop is controlled by multiplying the output with the normalized noise control level. The multiplier is placed between the divider and the integrator.

This method assumes a predictable linear relationship between the input amplitude and the noise control signal. Mathematically this relationship exists. However, the noise introduced by other components in the system and the noise at the signal input are not reflected in the noise control signal. The true noise level will be higher than the noise level solely based on the noise control signal. This will result in an error in the cut-off frequency.

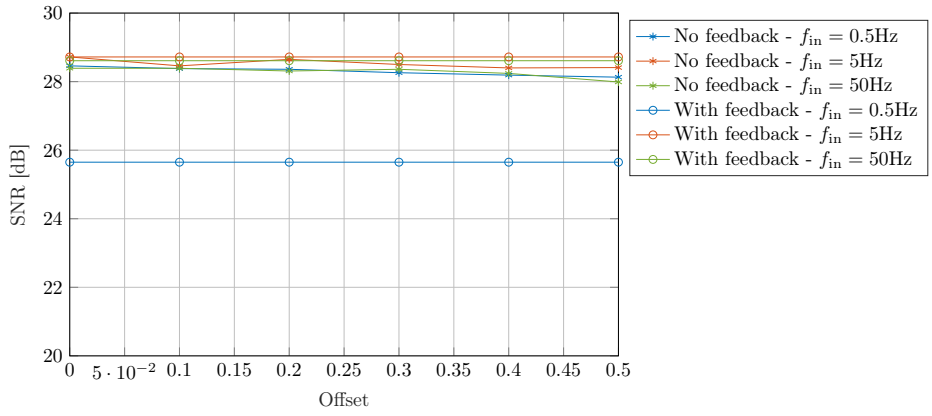
Results

The offset compensated system is compared to the system without offset compensation, for different input sinusoidal input signals, with $A_{\text{in}} = 1$. The algorithm to compensate for the input amplitude as presented above is not simulated. This simulation needs an implementation of the full system. The simulation results of the system with both feedback loops implemented are discussed in the next section.

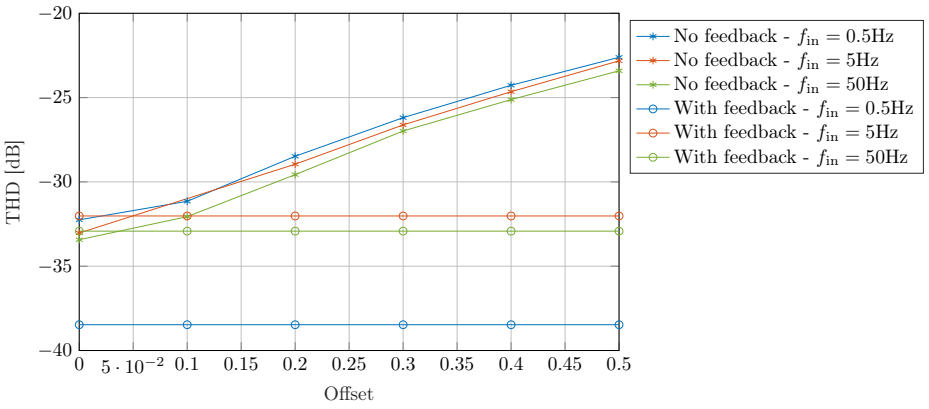
The simulation results are plotted in Figure 5.4.5. The performance of the system is independent of the offset level. When the input frequency is high, the feedback signal is very small and does not alter the output, nor the performance significantly. When f_{in} is reduced to 0.5 Hz, the output is reduced by 2.8 dB, as expected based on the cut-off frequency of the filter. The THD distortion decreases by approximately 6 dB, since the slope harmonic distortion slope is 40 dB/decade, as shown in Chapter 4.



(a) Signal-to-Noise-and-Distortion Ratio



(b) Signal-to-Noise Ratio



(c) Total Harmonic Distortion

Figure 5.4.5: The SNDR, SNR and THD versus the input offset with and without offset compensation. The performance is simulated for sinusoidal inputs with $A_{\text{in}} = 1$ and $f_{\text{in}} = [0.5,\ 5,\ 50]$ Hz

5.5 Simulation Results of Complete Control System

This section presents the simulation results of the complete proposed system. An analysis is performed on the performance of the system, focusing on the errors introduced by the control loops.

The performance of the two control loops is shown in the previous sections. However, the control loops will interact with each other, since an offset affects the maximum absolute amplitude. Furthermore, the threshold compensation is only simulated for an input amplitude of 1. By simulating the system with both loops active, the performance of the threshold compensation for lower input amplitudes is determined.

5.5.1 Results

The system is simulated while sweeping the input amplitude, input frequency, and the DC offset. The full results are presented in tables in Appendix C. This section will discuss the most important results and observations.

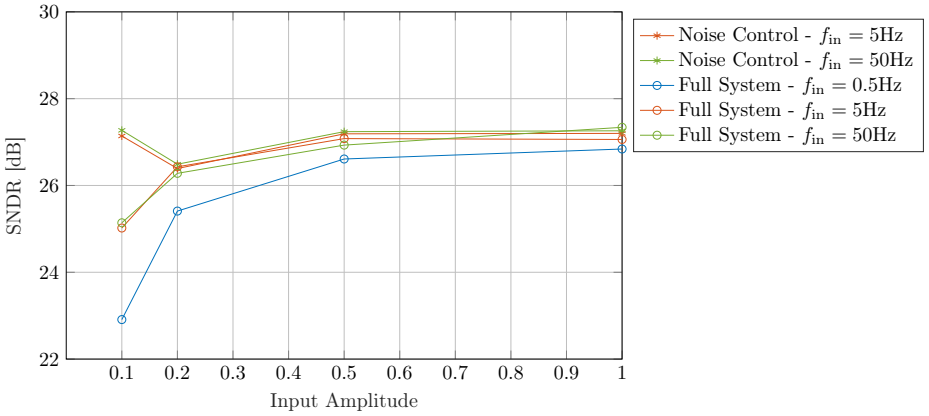
A summary of the simulations results of the full systems is given in Figure 5.5.1. The results of the full system are compared to the results with only noise control, to give more insight into the performance losses due to the offset compensation for different input amplitudes. A significant loss in performance is shown when A_{in} is small, especially for lower frequencies. This is due to inaccuracies in the offset compensation signal and will be discussed in the next paragraph.

Noise Control with Offset Compensation

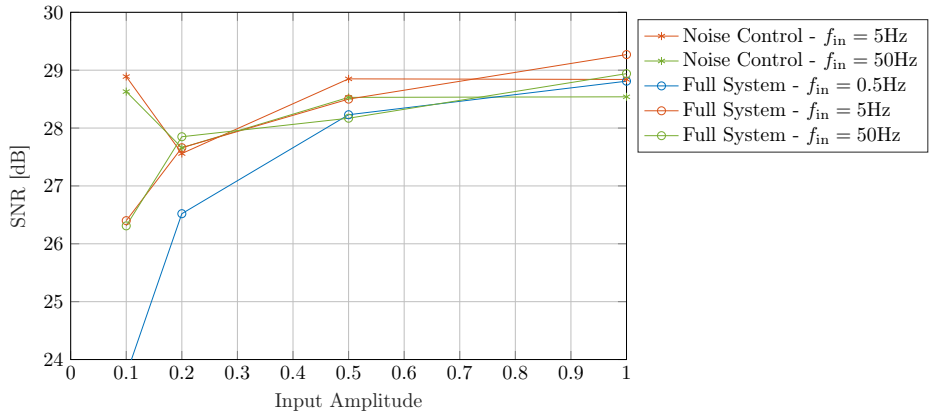
In Section 5.3, the noise control signal is shown for different input amplitudes. However, the effect of an offset to the noise control loop is not considered. When the offset is not (yet) compensated, a large overshoot can occur, as shown in Figure 5.5.2b. The plots show the output and the noise control signal for a sinusoidal input signal with $A_{\text{in}} = 0.1$, $f_{\text{in}} = 0.5$ Hz, and an offset of 0.5.

Initially, the noise level is zero. Since the offset is larger than the signal, the output is always -1. Therefore the noise control level takes the maximum step. Since the threshold compensation depends on the noise control signal, it only starts to compensate when noise is added. In the very first sample after the noise control level is adjusted the first time, the offset is not compensated. The output stays -64, as shown in Figure 5.5.2a. This high absolute output is stored as a peak output value and causes the noise control level to increase further at 4.2 seconds. Only after 6.4 seconds, the offset is fully compensated. Up until the offset is compensated, the maximum absolute output is increased by the offset. The noise level increases, which lead to an overshoot. The noise level is reduced to its settling point at 59 after 8.4 seconds.

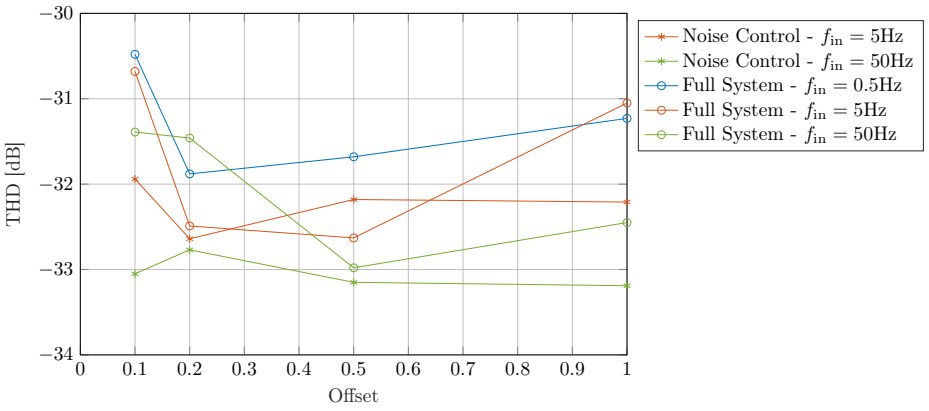
This simulation shows that the noise control signal is affected by an uncompensated offset. The result is that the noise level increases to a value which is too high. Meanwhile, the threshold compensation starts slow, while at the start of the simulation, the noise level is low. The settling speed is reduced by these effects, but will always converge to the correct settling point.



(a) Signal-to-Noise-and-Distortion Ratio

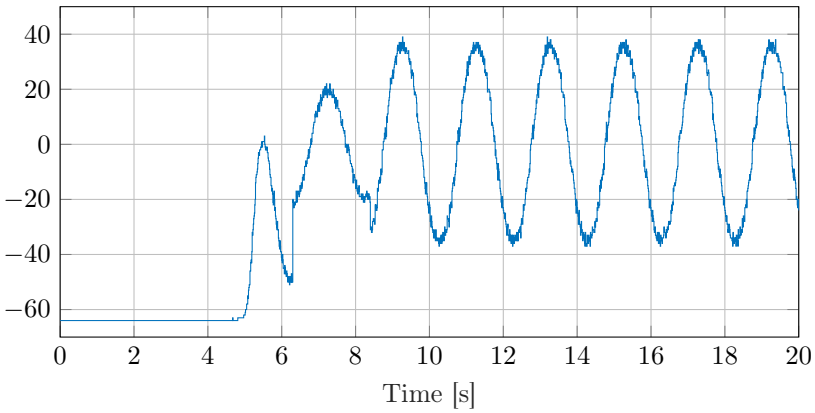


(b) Signal-to-Noise Ratio

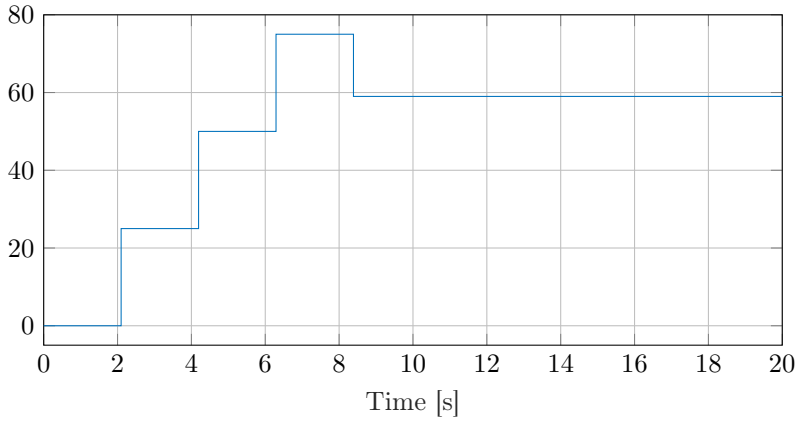


(c) Total Harmonic Distortion

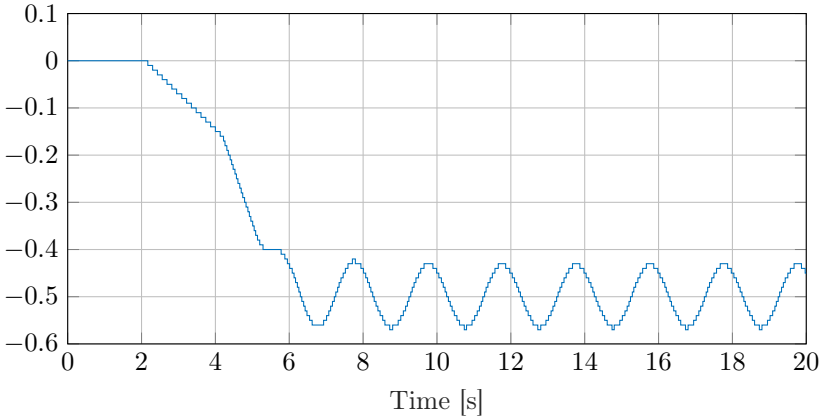
Figure 5.5.1: The SNDR, SNR and THD versus the input amplitude of the system with only noise control and the full system. The performance is simulated for sinusoidal inputs with $A_{\text{in}} = 1$ and $f_{\text{in}} = [0.5,\ 5,\ 50]$ Hz



(a) Output signal



(b) Noise Control Signal



(c) Noise Control Signal

Figure 5.5.2: Output signal, noise control signal, and offset compensation signal for an sinusoidal input signal with $A_{\text{in}} = 0.1$, $f_{\text{in}} = 0.5$ Hz, and an offset of 0.5. The plots show the large overshoot and the ringing of the noise control signal.

Offset Compensation

The limited resolution of the offset compensation introduces a quantization error. This is not significant for large input values, but if the input values drop, the performance loss increases to -2 dB for $f_{\text{in}} = 5$ Hz and even -4 dB for $f_{\text{in}} = 0.5$ Hz.

In the former section, only the error for a constant offset level is taken into account to determine the step size of threshold compensation. The expected quantization error of the threshold compensation, based on the LSB step is:

$$\sigma_Q = \sqrt{\frac{\text{LSB}^2}{12}} = \frac{0.01}{3.46} = 2.89 \cdot 10^{-3} \tag{5.5.1}$$

For an input amplitude of 0.1, the SN_QR is calculated:

$$20 \cdot \log^{10}\left(\frac{A_{\text{in}}/\sqrt{2}}{2.89 \cdot 10^{-3}}\right) = 20 \cdot \log^{10}\left(\frac{\left(0.1/\sqrt{2}\right)/\sqrt{2}}{2.89 \cdot 10^{-3}}\right) = 24.8 \text{ dB},$$

where N_Q is the quantization error only. The signal input amplitude $A_{\text{in}} = 0.1/\sqrt{2}$. The amplitude is reduced by a factor $\sqrt{2}$, while the threshold compensation filters the signal. The input frequency is after all equal to the cut-off frequency of the filter. The SN_QR is therefore higher when the frequency is increased. For a input signal with $f = 5$ Hz, the $SN_QR = 27.8$ dB

5.5.2 Conclusion

In this chapter, a system level design is presented which controls the noise level and the offset of the system. To employ the existing noise, it is necessary to control the amount of noise in the system. While the noise added by the electronic components can be predicted to a relatively high accuracy, the noise already existing in the signal is not reliable. By implementing a controllable noise source, the system can not only adjust the noise level to the noise from the environment, but also the dynamic range is extended. Since the system only measures the input signal relative to the noise level, by controlling the noise source, also a smaller input amplitude can be compensated. This design allows for input amplitudes ranging from 0.1 to 1, while the SNDR decrease is limited to 0.4 dB.

An offset compensation path is examined. A dynamic offset compensation feedback path is considered. This compensation does not only compensate for the offset of the comparator. It filters out all signals below 0.5 Hz. While this method showed good results for high input amplitudes, the quantization noise reduces the SNDR significantly when the input signal amplitude is small.

The optimized system without any control has shown to perform at 27.3 dB SNDR. After implementation of the offset compensation, the performance of the system is independent of the applied offset. The feedback signal itself, however, reduces the performance of the system. This occurs especially when the input amplitude is low. The largest decrease in performance is observed when both the input amplitude and the input frequency are small. The peak SNDR of 27.34 dB is observed for $A_{\text{in}} = 1$ and $f_{\text{in}} = 50$ Hz. For $A_{\text{in}} = 0.1$ and $f_{\text{in}} = 0.5$ Hz, de SNDR is only 22.9 dB, due to quantization error introduced by the offset compensation loop. To limit the SNDR reduction to 0.15 dB, an LSB step-size of the DAC smaller than $0.02 \cdot A_{\text{in}}$ is required.

The filter and feedback loops are implemented completely digital. By doing so, the reconfigurability of the system is maximized. Properties of the digital blocks can easily be changed on chip to control, for example, the bandwith of the system. The output amplitude is set by one digital value, and could therefore also be adjusted easily.

6

IC Implementation

This chapter comprises the implementation of the comparator and the noise source in 180nm AMS technology. While the chip will not be fabricated, the results are based on schematic simulations in Cadence. The noise control loop, as well as, the digital filter are not implemented in Cadence. The noise control loop, as proposed in the previous chapter only converges after a long time, due to the low minimum frequency of the system. Meanwhile, the oversampling ratio is very high and without noise simulation, the system does not perform. Simulation of the complete system can only be performed using a transient noise simulation. The required computing power to perform a transient noise simulation is too high to run a full system simulation in Cadence.

By focusing on the conversion of the input with added noise to a bit-stream, the concept of stochastic resonance analog-to-digital conversion is shown. The main challenges in implementing the system on an integrated circuit lie in these components.

In contrast to the system level design, the electronic implementation is based on a differential input. The performance, nor the behavior of the SR-ADC changes on system-level. However, by implementing the comparator in a differential fashion, many artifacts and non-idealities are reduced. To achieve a low power consumption, the supply voltage is 1.2 V. The system does not need a high dynamic range since the input is directly transformed into a bit-stream. Especially in the digital components, the power will be reduced significantly by using 1.2 V instead of 1.8 V.

The design and simulations of the comparator are presented in Section 6.1. In Section 6.2 a controlled noise source implementation is proposed. The limitations of the simulator are discussed in Section 6.3. Finally, in Section 6.4 the conclusions and discussion are presented.

Sampling frequency f_s	2 MHz
Differential Input Amplitude	$1 - 10$ $\mathrm{mV_{p\text{-}p}}$
Gain	∞
Differential offset, input referred	100 μV
Total maximum input referred noise	< 0.6 $\mathrm{mV_{RMS,dif}}$
Power	As low as possible
V_{DD}	1.2 V

Table 6.1: Requirements for the comparator.

6.1 Comparator

Based on the system level design, the requirements of the comparator are determined. The most important requirements are shown in Table 6.1. The sampling frequency and the maximum offset are directly taken from the system level design. As mentioned above, the V_{DD} is chosen to be 1.2 V. Since the input is not pre-amplified, the comparator needs to be able to distinguish very small voltage differences. The total required gain of the comparator is thus chosen to be infinite. The maximum noise level is based on the total desired maximum input referred noise in the system at the minimum input amplitude, 0.6 $\mathrm{mV_{RMS,dif}}$. To ensure that the total input referred noise in the system is below this value, the contribution of the comparator should be well below this value.

This section is build up as follows: First, the topology is chosen in Subsection 6.1.1, the offset compensation is discussed in Subsection 6.1.2. The noise, including the kickback noise, simulations are presented in Subsection 6.1.3 and the effect of a bias voltage mismatch is discussed in Subsection 6.1.4.

6.1.1 Topology

A large number of comparator topologies exist. Generally, two main categories can be distinguished: Open loop comparators and latch comparators. The open loop comparator is based on a saturating amplifier. The latch-comparator is clocked and has a positive feedback loop. Combinations of both principles are used as well, where mostly the latch is combined with a pre-amplification stage. This last topology is often used to reduce the input referred offset, a drawback is that traditional pre-amplification requires a wideband amplifier with continuous power consumption.

The pre-amplification can also be performed in a dynamic fashion, which means the power is still shut down after regeneration. An example is a double-tail latch as presented in [25]. This topology targets especially very low voltage supplies since the number of stacked transistors is reduced from 4 to 3. Furthermore, the paper shows improved insensitivity for the common mode voltage. Many alternative implementations of the 2-stage regenerative latch can be found in literature, typically targetting high speed and low power conversion, while minimizing the noise and offset.

In Table 6.2 some general advantages of the different topologies are summarized. Based on this comparison a regenerative latch is chosen, although it suffers from relatively large offsets. Various techniques exist to compensate the offset, including compensation by adjustment of the current through one of the branches and compensation by adjusting the output capacitances of the comparator.

	Advantages	Disadvantages
Multi-stage open loop (e.g. [26, 27, 28]	• Precize	• Static power consumption • Gain is limited
Regenerative latch [29]	• High speed • Efficient - no static current • Compact	• Large offset
2-stage regenarative latch(e.g. [25, 30])	• High precision • High speed • No static current required	• Increased complexity, higher demands on transistor sizing

Table 6.2: Comparison of different basic comparator topologies

Regenerative Latch

The regenerative latch, better known as the StrongARM latch is introduced in 1993 [29]. An improved architecture, with two added switches which pull up the sources of the input-transistors is proposed in [31]. The implemented circuit is shown in Figure 6.1.1. In addition to the basic regenerative latch, two adjustments have been made. The capacitors $C_1 - C_4$ are added, to implement the offset compensation presented in Subsection 6.1.2. Secondly, two extra input transistors are added (M_{1b} and M_{2b}).

The StrongARM latch is analyzed as follows. In this analysis, M_{1a} and M_{1b}, as well as M_{2a} and M_{2b}, are considered one transistor. During one comparison, the operation of the StrongARM latch can be divided into 4 phases. The first phase is the reset phase. In this phase, the switches $S_2 - S_5$ are turned on and S_1 is turned off. The capacitances $C_1 - C_4$ are precharged to V_{DD}.

The second phase, the sampling or amplification phase, starts when the clock becomes high. The switches pulling the nodes to V_{DD} are turned off, and current is drawn through S_1. The capacitors C_1 and C_2 discharge through M_1 and M_2. The input voltage diffence result in a current difference between the two input transistors. The voltage drop over each of the capacitances C_1 and C_2 is described by:

$$V_C(t) = \int_0^t \frac{i_D}{C} dt = \frac{i_D \cdot t}{C}, \tag{6.1.1}$$

where i_D is the current through the corresponding transistor (I_{M_1} or I_{M_2}). This phase ends when the voltage V_C drops below $V_{th,M_{3,4}}$, and the transistors $M_{3,4}$ start to conduct. The duration is given by:

$$T_s = \frac{C_{1,2} \cdot V_{th,M_{3,4}}}{\frac{1}{2} I_{CM}}. \tag{6.1.2}$$

In the third phase, the propagation phase, M_3 and M_4 start to conduct as the voltages over the input capacitors drop. The charge at C_1 and C_2 is shared with C_3 and C_4 respectively. Meanwhile, the current imbalance through M_1 and M_2 also creates a voltage

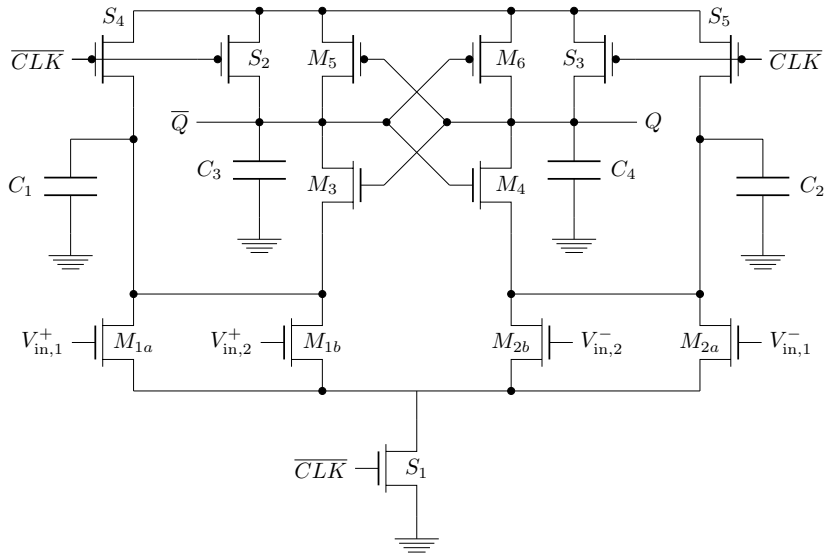


Figure 6.1.1: Regenerative latch, as proposed by Kobyashi et al. with added switches S_4 and S_5. Two input transistors (M_{1b} and M_{2b}) are added to allow differential comparison.[29, 31]

difference between C_3 and C_4. The contribution of both effects to the output voltage gain depends on the ratio between C_{1-2} and C_{3-4}.

Both output voltages will drop, until either M_5 or M_6 start to conduct, which announces the final phase: regeneration. The conducting transistor pulls up one of the two nodes, while the other node will be pulled down further.

Differential Implementation

The comparator is implemented in a differential fashion. The four input comparator is built by splitting the transistors $M_1$and M_2, as shown in Figure 6.1.1. Hereby, the width of the transistors is halved.

The total current through the transistors M_1 and M_2, are $I_{M_{1a}} + I_{M_{1b}}$ and $I_{M_{2a}} + I_{M_{2b}}$ respectively. Assuming equal common mode voltages at each input, the voltage transfer is found in Equation 6.1.3. Hereby is $g_{m,a} = g_{m,b} = g_m/2$, since the current through the transistors is halved.

$$\begin{aligned} I_{M_1} &= I_{M_{1a}} + I_{M_{1b}} \\ g_m \cdot V^+_{\text{in}} &= g_{m,a} \cdot V^+_{\text{in},1} + g_{m,b} \cdot V^+_{\text{in},2} \\ V^+_{\text{in}} &= \frac{1}{2} V^+_{\text{in},1} + \frac{1}{2} V^+_{\text{in},2} \end{aligned} \tag{6.1.3}$$

The differential implementation is necessary for two reasons. The noise input and the signal input originate from different sources. The DC-mismatch between the signal input and the noise input is not well-controlled. By implementing the comparator differentially, a common mode mismatch between the signal and the noise input appears in both branches of the comparator, and thus in the common mode. While in a non-differential implementation the mismatch in DC-level between the two inputs is directly added to the differential input, in a differential implementation it will only create a change in the g_m of the input transistors.

The mismatch in g_m translates to a gain mismatch between the two inputs. Since the noise level is controlled by a feedback loop, this feedback loop can take care of this mismatch.

The second advantage of a differential implementation is the reduced effect of the kick-back noise. During the reset phase, the parasitic capacitances of M_1 and M_2 are charged. After the reset these capacitances are (partially) discharged through the signal and the noise input, resulting in a voltage change at the input. When the source impedances are not matching, the voltage due to this charge injection is not matched either. The impedance of the signal source and the noise source are not matched. However, by using a four input comparator, the source impedance at each branch of the comparator are better matched.

6.1.2 Offset Compensation

The offset requirements are a limiting factor in this comparator design. In general, the regenerative latch does not have a very low offset. However, by scaling up the transistors, the offset due to device mismatch can be reduced. To meet the specification of a differential offset < 50 μV controlled offset cancellation needs to be implemented.

Two techniques to compensate the comparator offset are considered [32, 33]. The first technique is to compensate for the offset by adjusting the voltage in one branch of the comparator. Simulations of the current adjustment showed a limited accuracy of the compensation. To increase the accuracy, it is concluded that a precise, low-current, current source needed to be implemented. Compensating the threshold using adjustable capacitances, on the other hand, relies fully on the ratio between the smallest adjustable capacitance and the total capacitance. The ratio can be designed such that the compensation can be implemented very accurately.

Uncompensated Offset

The comparator is designed such that the initial, uncompensated offset is relatively low. To achieve a low uncompensated offset, large transistors are used in the comparator. The transistors, except for the input transistors, are equally sized at $W = 5\mu$ and $L = 1.08\mu$. The width of the four input transistors ($M_{1a,1b,2a,2b}$), is 2.5 μm. The capacitors C_{1-4} are all 80 fF.

A Monte-Carlo simulation is performed to determine the uncompensated offset of the comparator. The simulation is performed keeping 3 inputs at a constant voltage of 600 mV, while slowly sweeping the third input. The offset is defined as the differential input on one of the two input pairs. The results of 250 Monte-Carlo runs are shown in Figure 6.1.2. The simulated standard deviation is: $\sigma_{\text{offset}} = 3.93$ mV.

Adjustable Capacitor Implementation

The offset is compensated by adjusting the capacitors C_1 and C_2. The capacitances are adjusted by adding small, adjustable capacitances parallel to C_1 and C_2. The adjustable capacitors are implemented using the parasitic capacitance of a transistor, connected as shown in Figure 6.1.3.

The source and the drain are shorted, and the gate is either connected to ground or to V_{DD}. The voltage on the drain and the source is, at the start of the comparison 1.2 V. The bulk is connected to the substrate.

The gate-source and gate-drain capacitance changes slightly when the control voltage

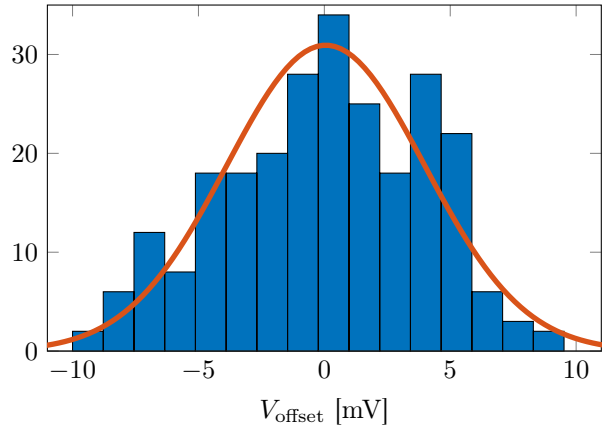


Figure 6.1.2: Monte-Carlo simulation results of the comparator without offset compensation. The graph shows a histogram the differential input-reffered offset of 250 runs and a fitted normal distribution.

	$V_{\text{control}} = 0$ V	$V_{\text{control}} = 1.2$ V
C_{bb}	38.58 aF	24.22 aF
C_{bs}	-4.178 zF	-2.622 zF
C_{gd}	4.178 zF	2.627 zF
$\boldsymbol{C_{gdol}}$	**51.45 aF**	**66.66 aF**
C_{gg}	38.59 aF	24.23 aF
C_{gs}	4.178 zF	2.627 zF
$\boldsymbol{C_{gsol}}$	**51.45 aF**	**66.66 aF**
C_{jd}	48.81 aF	48.81 aF
C_{js}	48.81 aF	48.81 aF

Table 6.3: Simulated parasitic capacitances of the transistors used to implement the controllable capacitors. The values are simulated for $V_S = V_D = 1.2$ V, the value at the start of the sampling phase. The gate-source and gate-drain capacitance are bold, while the shift in these capacitances implement the controlled capacitor.

switches from 0 V to 1.2 V. When the gate voltage is 1.2 V, the gate-source and gate-drain voltages are 0 V. By switching the gate voltage to 0 V, the transistor gets negatively biased with respect to the drain and the source. The overlapping capacitances decrease hereby slightly. For a minimum-sized transistor, the change in capacitance is approximately 30 aF. In Table 6.3 the simulated parasitic capacitances of a minimum sized transistor are shown for both possible control voltages.

This implementation differs from conventional implementation in which the ΔC is implemented by creating an inversion layer. Using this technique the minimum ΔC is reduced by approximately factor 3.

The change in capacitance is proportional to the with of the transistor. By doubling the transistor size, the parasitic capacitance is doubled as well and the LSB to MSB can be implemented.

The offset change due to the added capacitance is given by:

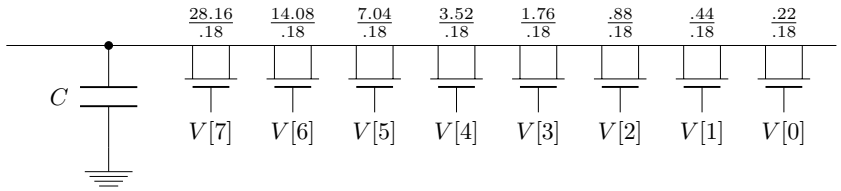


Figure 6.1.3: Implementation and sizing of the contollable capacitances. The control is binary coded, whereby $V[7]$ is the MSB and $V[0]$ is the LSB.

$$\Delta V_{\text{offset}} = \frac{2\Delta C_{1,2}}{C_{3,4} + 2C_{1,2}} \cdot \frac{V_{P1,2}}{2} = \frac{2 \cdot 30 \cdot 10^{-18}}{3 \cdot 80 \cdot 10^{-15}} \cdot \frac{0.45}{2} \approx 56\ \mu\text{V}, \tag{6.1.4}$$

where $C_{1-4} = 80$ fF and $V_{P1,2} \approx 0.45$ V.

Figure 6.1.4a shows simulations of the offset change created by the controllable capacitances for different binary input values. The step size varies between 36 μV and 50 μV, as shown in Figure 6.1.4b. The simulated ΔV_{offset} is smaller than calculated. The total capacitance is assumed to be equal to C_{1-4}, and the parasitic capacitances are neglected.

The total range over which the threshold can be compensated is 10 mV. The standard deviation of the uncompensated offset is 3.93 mV, thereby the probability that the initial offset can be compensated is 98.9%.

6.1.3 Comparator Noise and Errors

The strongARM latch comparator is typically not the lowest noise comparator which can be used. However, by using large transistors and caused by the added capacitors, the noise is reduced significantly.

Sampled Noise

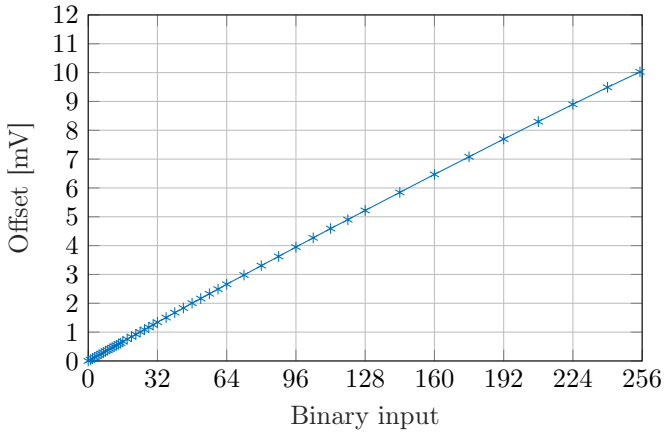
The comparator works in a sampled regime. While the comparator is sampled at 2 MHz, the actual bandwidth of the comparator is determined by the aperture time and can be orders of magnitude higher than the sampling frequency [34]. Since the actual sampling of the comparator is performed faster than the sampling time, high-frequency noise is sampled by the comparator.

Since the comparator works in a sampled regime, all signals (and noise) above the Nyquist frequency will fold back into the Nyquist bandwidth due to aliasing. The noise folding is illustrated in Figure 6.1.5. Since white noise is dominating at high frequencies, noise folding adds white noise to the input referred comparator noise.

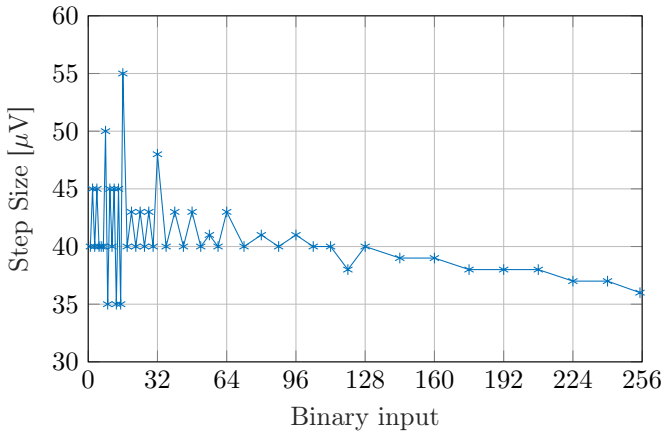
To realize the offset compensation, four relatively large capacitors were added to the comparator. This capacitors reduced the speed of the comparator significantly, and thereby decreased the noise folding.

Noise Simulation

The noise of the comparator is simulated by performing a transient noise simulation. A small input voltage is applied to the comparator, and the probability of each of the two



(a) The change in offset created by the controllable capacitance. The offset is shown for different binary input values.



(b) The step size for different binary inputs. The step size is calculated from the offset difference between to successive simulation points shown in Figure 6.1.4a.

Figure 6.1.4: Simulation results for the comparator offset compensation. The simulation is performed by measuring the offset change for different binary input values.

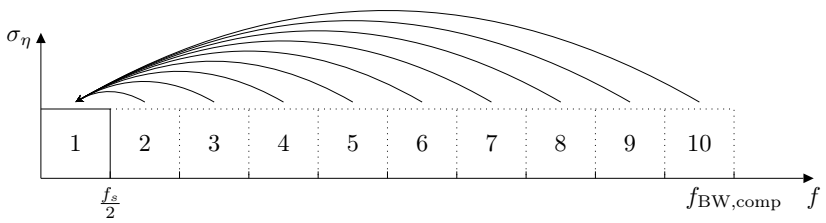


Figure 6.1.5: Illustration of noise folding due to undersampling of the noise. High frequency noise is sampled and appears in the in the bandwith from 0 to $f_s/2$ Hz.

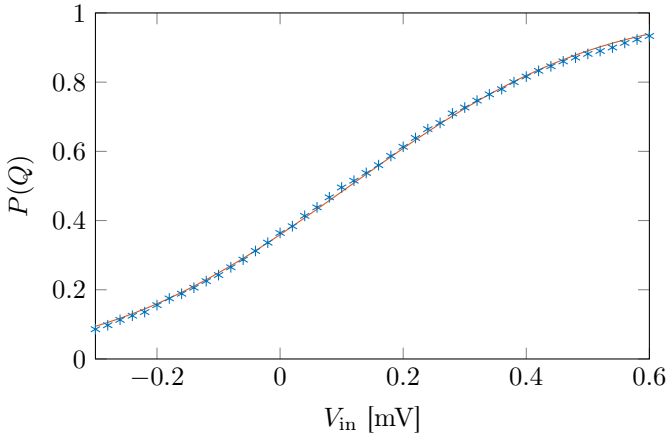


Figure 6.1.6: Cumulative distribution function of the comparator noise. The CDF is fitted on transient noise simulations. For different input voltages the probability of a high comparator output is calculated, based on 2000 samples per run.

output states is measured. By repeating this simulation for different input voltages, the cumulative distribution function is built.

The comparator noise is simulated using a transient noise simulation with a small input voltage applied to the comparator. Both negative inputs are set to 600 mV, while both positive inputs shifted from 599.7 mV to 600.6 mV in steps of 0.02 mV. For each input voltage, the transient noise simulation is run for 1 ms. The probability of a high output for each simulation is shown in Figure 6.1.6. The simulations are fitted to the Gaussian cumulative distribution function. From this result, the standard deviation of the input referred noise is calculated at 0.32 mV. Furthermore, an offset of 0.11 mV is simulated. This offset is not observed in a transient simulation without noise. While both the positive inputs are shifted, the noise referred to one input pair is equal to 0.64 mV differential.

Kickback Noise

When switching to the sampling phase, the source voltage of the input transistors drops quickly. The parasitic gate-source capacitance creates a charge injection in the sources.

The voltage peak due to the charge injection can be calculated using the equivalent circuit shown in Figure 6.1.7. Left the section of the comparator is shown which is involved in the kickback noise. Right, an equivalent circuit is shown which is used to calculate the voltage peak.

During the reset phase, $V_{D,S_1,\text{reset}} \approx 528$ mV. On a falling edge of the clock, the reset phase ends and S_1 starts to conduct, causing V_{D,S_1} to drop. While $V_{GS,M_1} < V_{th}$, the current $I_{M_1} = 0$. The equivalent circuit on the right in Figure 6.1.7 is valid during the voltage drop of V_{D,S_1}, from $V_{D,S_1,\text{reset}}$ to V_{th,M_1}.

To create the voltage drop over the capacitor the current source I_{S_1} needs to charge the parasitic capacitance. The gate-drain voltage is described by:

$$V_C = \frac{1}{C} \int I_{S_1} \mathrm{d}t \tag{6.1.5}$$

The current I_{S_1} creates a voltage drop over the source resistance of the input. The

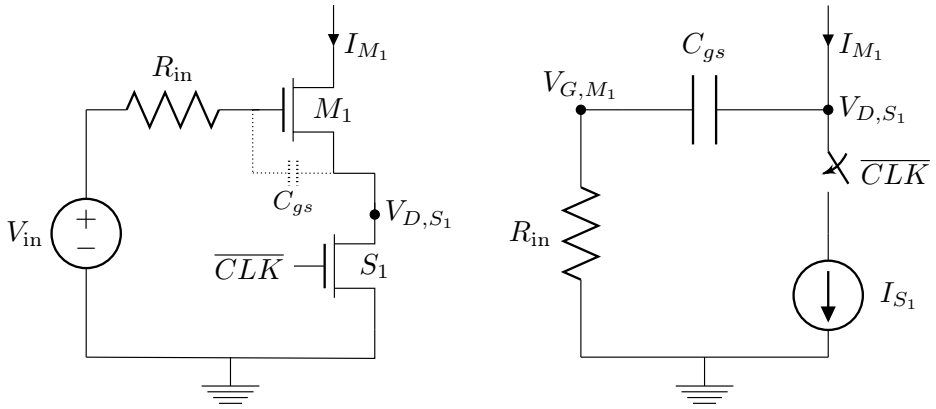


Figure 6.1.7: Equivalent circuit used to determine the kickback noise.

kickback noise is minimized by designing the current I_{S_1} to be small, thus the W/L ratio of transistor S_1 is chosen to be small.

The kickback noise is simulated by adding a resistance to one of the two differential inputs. The input impedance of the electrodes can not accurately be predicted, since it depends on the electrodes themselves, but also heavily depend on the electrode-tissue interface. To illustrate and quantify the effect of the kickback noise, a resistor of 50 $K\Omega$ and 45 $K\Omega$ are added to the positive and the negative signal inputs respectively. The simulation results are shown in Figure 6.1.8. The differential input of the first input pair is 0.75 mV, equal to the simulated offset. The second input pair is set to 600 mV.

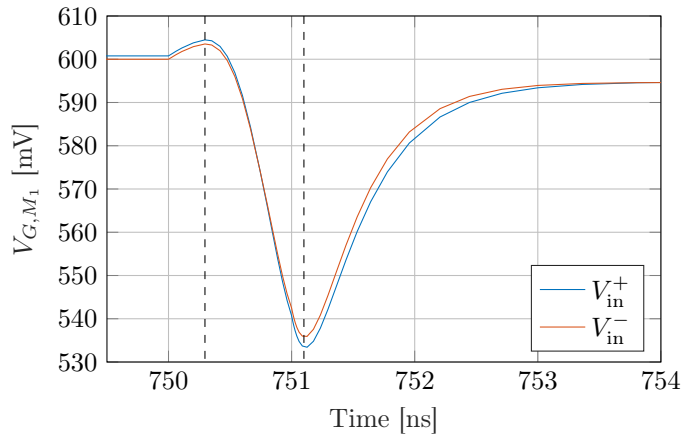
The voltage V_{D,S_1} drops till M_1 starts to conduct, creating a voltage drop from 545 mV to 200 mV. The maximum current through the gate of M_1 is 1.3 μA. The voltage drop at the gate, $\Delta V_{G,M_1} = 70$ mV.

The current through the gate of each input transistor depends on the input source impedance. A smaller source impedance results in a larger current through the gate, reducing the voltage mismatch between the gates of the two input transistors.

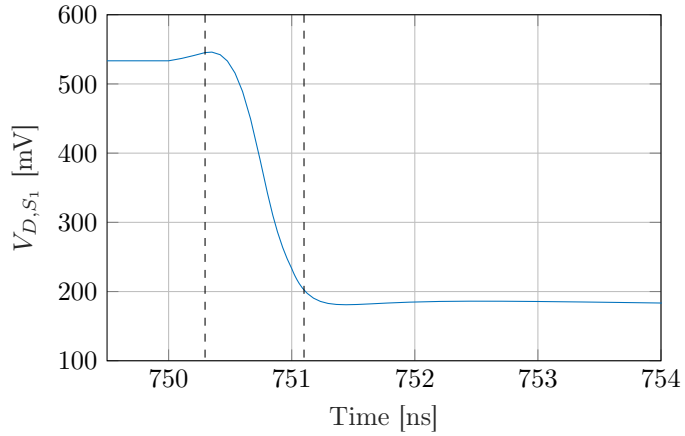
The offset depends on the absolute impedance mismatch between the positive and the negative input source. The simulated mismatch for 245 $K\Omega$ and 250 $K\Omega$ is 0.73 mV. Increasing the offset to 50 $K\Omega$ gives an offset of 7.2 mV, both for input impedances of 200 $K\Omega$ and 250 $K\Omega$ and 0 Ω and 50 $K\Omega$.

The second effect of the kickback noise is a gain mismatch between the signal and the noise input since the kickback voltage changes the g_m of the transistor. The input with a larger source resistance suffers from a larger voltage drop. Due to this voltage drop, the other input pair will start to conduct relatively quicker.

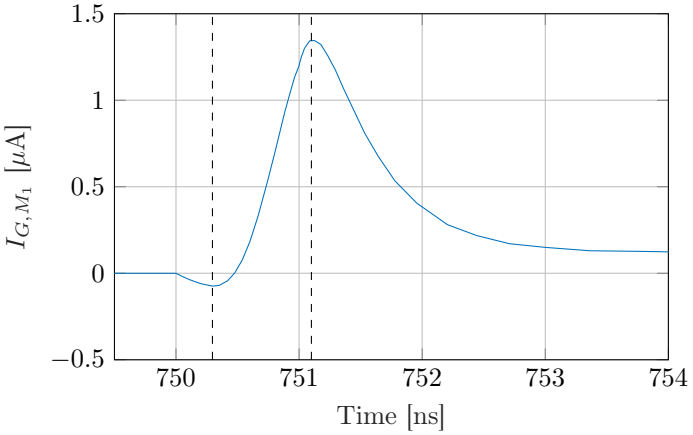
This effect is simulated by applying an equal impedance to both terminals of the signal input. A differential voltage varying from 0.1 to 10 mV is applied to the noise input. The simulation results, presented in Table 6.4, show that for a large source impedance, the gain of the differential inputs is not matched. For a 50 $K\Omega$ source impedance, the gain mismatch is approximately 8%, while for a source impedance of 250 $K\Omega$, this increases to 50%.



(a) Kickback noise voltage at the gate of the input transistors.



(b) Drain-source voltage of S_1 when the kickback occurs. The input impedance is 50 $K\Omega$ and 45 $K\Omega$ for V_{in}^+ and V_{in}^- respectively.



(c) Current throug the gate of M_1 to charge the gate-source capacitance of M_1, the input transistor of V_{in}^+.

Figure 6.1.8: Simulation results of the kickback noise for an input impedance of 50 $K\Omega$ and 45 $K\Omega$ for input V_{in}^+ and V_{in}^- respectively. The input $V_{\text{in}}^+ = 600.75$ mV and $V_{\text{in}}^- = 600$ mV.

	Source impedance 50 $K\Omega$		Source impedance 250 $K\Omega$	
Differential Noise Input	Differential Signal Input	Offset	Differential Signal Input	Offset
10 mV	-10.78 mV	-780 uV	- 14.71 mV	-4.71 mV
5 mV	- 5.40 mV	-400 uV	- 7.43 mV	- 2.43 mV
2 mV	- 2.16 mV	-160 uV	-2.99 mV	-990 uV
1 mV	-1.08 mV	-80 uV	-1.50 mV	-500 uV
500 uV	-541 uV	-41 uV	-749 uV	-249 uV
200 uV	-216 uV	-16 uV	-300 uV	-100 uV
100 uV	-108 uV	-8 uV	-150 uV	-50 uV

Table 6.4: Simulation results of the gain mismatch caused by kickback noise. A source impedance is added to the signal input pair. A differential voltage is applied to the noise input pair. The table shows the differential voltage on the signal input for which the comparator switches state.

	Differential Voltage At Noise Input = 1 mV		Differential Voltage At Noise Input = 5 mV	
Bias Voltage Noise Input	Differential Voltage Signal Input	Error	Differential Voltage Signal Input	Error
580 mV	-830 uV	170 uV	-4.16 mV	840 uV
590 mV	-910 uV	90 uV	-4.57 mV	430 uV
595 mV	-955 uV	45 uV	-4.78 mV	220 uV
600 mV	-1000 uV	0 uV	-5.00 mV	0 uV
605 mV	-1050 uV	-50 uV	-5.23 mV	-230 uV
610 mV	-1090uV	-90 uV	-5.47 mV	-470 uV
620 mV	-1200 uV	-200 uV	-5.98 mV	-980 uV

Table 6.5: Simulated error for a common mode mismatch

6.1.4 Common Mode Mismatch

A mismatch in common mode voltage between the two differential inputs results in a gain mismatch between the two differential inputs. The effect of a bias mismatch is simulated as follows: The noise input is biased at 590 mV to 610 mV, while a differential input of 1 mV is applied. The signal input is biased at 600 mV. The differential input for which the comparator switches is measured. The results are shown in Table 6.5.

The measured error depends on the difference in bias voltage, as well as the differential input. A first-order approximation gives an error of: $10 \cdot (V_{\mathrm{CM1}} - V_{\mathrm{CM2}}) \cdot V_{\mathrm{diff}}$. The gain mismatch is approximately 1% per mV mismatch, based on a bias voltage of 600 mV. The input transistors of the differential pair with the highest bias voltage start to conduct earlier than the differential pair with the lower bias voltage. When all four input transistors are in saturation, the (common mode) current through the transistors with a higher bias voltage is larger, leading to a higher gain with respect to the other input.

6.2 Noise Source

In this section, several implementations of the noise source are considered. First, different random fluctuations appearing in the electronic systems are briefly discussed. Some noise mechanisms are normally not dominant in IC-design in 180 nm technology, such as shot noise and burst noise.

The second subsection focusses on the implementation of thermal, resistive noise as a source of the noise. To achieve the required noise power a noise amplifier is implemented in Subsection 6.2.3. A high-pass filter to reduce the flicker noise is presented in Subsection 6.2.4. Finally, simulations of the generated noise are presented in Subsection 6.2.5.

6.2.1 Noise Source Selection

Several natural random fluctuations in electrical circuits can be used to generate noise or random numbers. The most common noise observed in every analog circuit are pink noise and Johnson-Nyquist noise, or thermal noise. While in most applications noise is avoided as much as possible, in cryptography, a need for random number generators exists. The systems creating random numbers found in the literature are based on meta-stability or random telegraph noise (RTM).

Pink Noise

The frequency spectrum of pink noise is inversely dependent on the frequency and is therefore often referred to as $1/f$-noise. Pink noise appears in all active analog circuits. The pink noise introduced by a transistor is:

$$\bar{V}^2_{\mathrm{eq},1/f} = \frac{K_{1/f}}{2\mu C^2_{ox} W L} \cdot \frac{\Delta f}{f} \tag{6.2.1}$$

Relatively good results are obtained in SR systems using pink noise when it is compared to an analog system with the same pink noise [19]. The SNR-loss of an SR-system using pink noise is lower than the SNR-loss for white noise. However, the SNR-loss is a comparison to an ideal analog system. Since a large part of the noise power appears in the passband of the system, only a small part of the noise can be filtered out in an analog system.

The non-linearity introduced in the SR-system shapes the frequency spectrum of the noise. The noise at the output of the comparator contains relatively more high-frequencies. However, the noise power density still decreases when the frequency is increased. Thereby, the amount of noise in the passband of the filter is higher for pink noise than for white noise.

Thermal Noise

Thermal noise, or Johnson-Nyquist noise is caused by random fluctuations of the charge carriers due to thermal activity. The power spectral density of the noise is described by:

$$\bar{v}^2 = 4k_B T R \left[\frac{\mathrm{V}^2}{\mathrm{Hz}}\right], \tag{6.2.2}$$

where the Boltzmann constant, $k_B = 1.38 \cdot 10^{-23}$ J/K, and the temperature is assumed to be at room temperature: $T = 300$ °K.

The noise spectral density is independent on the frequency, and thereby a resistor can be used as a white noise source. Thermal noise does not only appear in resistors but appears in all impedances at of a circuit, such as in the output resistance of a transistor. The theoretical power of thermal noise is infinite since it is not fundamentally bounded to a maximum frequency. The practical bandwidth of the noise is limited by the capacitive behavior of the system.

Shot Noise

Another source of noise in analog circuits is shot noise. It is caused by the discrete nature of charge carriers, electrons. Shot noise is observed when electrons cross a pn-junction. Shot noise is typically large in diodes in avalanche breakdown. The electrons crossing the p-n junction cross a large potential difference, leading to a high noise energy. However, to create avalanche breakdown, a large reverse bias voltage has to be applied to the p-n junction. In discrete implementations, where higher voltages are available, avalanche noise is often used to generate white noise.

In ultrathin gate-oxide devices, the leakage current increases, giving rise to gate shot noise. However, in 180 nm technology, the gate-oxide thickness is too large to create a high amount of shot noise with low voltages.

While higher amounts of shot-noise occur in the situations described above, shot noise occurs in all channels in which a current flows. The amount of shot noise is described by:

$$\begin{aligned} \bar{\sigma}_{i,\eta}^2 &= 2qI_d\Delta f \\ \bar{\sigma}_{v,\eta}^2 &= 2qI_d\Delta f \cdot r_o^2 \\ r_o &= \frac{v_T}{I_d}, \\ \bar{\sigma}_{v,\eta}^2 &= \frac{2q\Delta f \cdot v_T^2}{I_d} \end{aligned} \tag{6.2.3}$$

where $q = 1.602 \cdot 10^{-19}$, and $v_T = 26$ mV. Based on the equations above, the current I_D should be approximately 6 pA. The goal is to implement an adjustable noise source. To dynamically adjust the noise, this current source has to be adjustable. Adjustable noise sources in this range have not been found in literature, and would put high demands on the noise source design.

Burst Noise/Random Telegraph Noise

In cryptography, a need of random number generation exists. Since noise is fundamentally not different than a sequence of random values, techniques to create random numbers are investigated. Several papers are found in which burst noise or random telegraph noise (RTN) is used to design a random number generator.

Burst noise, also referred to as RTN, popcorn noise, bi-stable noise or impulse noise, is caused by metallic impurities in the pn-junctions. The effect of these impurities increases for ultra-thin metal-oxide gates. From the observations presented in [35], several conclusions can be drawn. The paper shows that the PSD of the RTN in 180 nm technology shows a $1/f$ behavior. Only in smaller technologies, the noise shows a flat behavior for in low frequencies. Furthermore, the PDF of RTN noise follows the Poisson distribution [36]. The Poisson distribution is a discrete, counting process. Its PDF is described by:

$$f_X(x) = \frac{\lambda^x}{x!} e^{-\lambda}, \tag{6.2.4}$$

where $x \geq 0$. The PDF is asymmetric in nature. For larger values of λ the distribution shows a more symmetric, Gaussian-like behavior. However, this only occurs is smaller technologies than the employed 180nm technology.

Conclusions

Based on the behavior of the different noise sources, a thermal noise source seems to be the most promising source for a controllable white noise source. While Shot noise and RTN is used in Gaussian white noise sources, and random number generators, both are not suitable to apply in the used technology.

In shot noise implementations, high voltages are used. Realizing those voltages in this design will introduce the need for a DC-DC converter, leading to higher power consumption and complicate the design significantly.

Burst noise, or RTN, does not show the desired behavior in 180nm technology. The PSD is not flat over a wide frequency range, and the PDF is asymmetric. The latter will result in an asymmetric transfer.

The spectral density of thermal noise is flat, has a Gaussian distribution, and the noise level can be predicted relatively precise. Therefore thermal noise complies with the desired behavior and is further examined.

6.2.2 Passive Thermal Noise

Based on Equation 6.2.2, the required impedance can be determined. The required noise RMS value, based on an input amplitude of 10 $\mathrm{mV_{p\text{-}p}}$, is 6.0 mV. Bandwith of the noise should be at least 1 MHz, and thus the required power spectral density of the noise is $3.6 \cdot 10^{-11}$ $\mathrm{V^2/Hz}$. To achieve this power spectral density, an impedance of 2.2 GΩ is required.

The bandwidth of the noise is limited by the total capacitances seen by the resistor, which create an RC-filter. The cut-off frequency of the RC-filter is given by:

$$f_{\text{-3dB}} = \frac{1}{2\pi RC}. \tag{6.2.5}$$

The equivalent noise bandwidth is:

$$f_{\mathrm{ENBW}} = \frac{\pi}{2} f_{\text{-3dB}} = \frac{1}{4RC}. \tag{6.2.6}$$

Thereby the total noise power is can be calculated:

$$\bar{v}^2 = 4k_B T R \cdot f_{\mathrm{ENBW}} = \frac{k_B T}{C}. \tag{6.2.7}$$

The total noise power is limited by the capacitance rather than by the resistance. To achieve the required noise power, the capacitance should be 115 aF.

The minimum capacitance related to the thermal noise source is determined by the parasitic capacitance of the resistor, the wire and the transistors it is connected to. Implementing this resistance, with such small parasitic capacitances is not feasible. To further use the resistor noise, it will need to be connected to a transistor.

The gate capacitance is defined by:

$$C_{gs} = C_{ox}WL. \tag{6.2.8}$$

The used technology is the AMS 0.18 um technology, with minimum dimensions: $W =$ 220 nm, and $L = 180$ nm and $C_{ox} \approx 250\mu A/V^2$. The minimum gate-source capacitance is thereby calculated at 66 aF.

Added to the parasitic capacitances of the transistors, also the implemented resistor, as well as the path between the resistor and the transistor will add to the total parasitic capacitance. The noise power of only the passive resistor, will not meet the specifications.

6.2.3 Noise Amplifier

The thermal noise is amplified to achieve the required noise power using a smaller resistor. A simple single-stage amplifier is used in open loop configuration. The output of the amplifier consists of both the noise of the resistor in front of the amplifier and the noise added by the amplifier. The amplifier is shown in Figure 6.2.1. The PMOS is biased at a current of 2.3 μA. The $g_m = 42$ μS, and $r_{o,M1} = 2.2$ MΩ, $r_{o,M1} = 3.1$ MΩ, resulting in a DC-gain of 54$\times$. M2 is sized 6u/0.5u and M1 is sized 2u/0.5u. The noise resistor is 2 Mohm. Figure 6.2.2 shows the output noise of the amplifier without load. The noise has an approximately flat spectrum from 20 kHz to 1 MHz. For lower frequencies, the $1/f$-noise dominates, while for higher frequencies the bandwidth of the amplifier limits the noise. From 20 KHz to 1 MHz, the resistive noise is dominating.

Pink Noise

Reduction of the pink noise can be achieved in two ways: increasing the transistor size or increasing the power. By increasing the transistor size (length), the bandwidth of the amplifier is reduced. When the $1/f$-noise of the amplifier is reduced without reducing the bandwidth, the thermal, white noise level of the amplifier decreases with the same amount.

The ratio between the bandwidth of the amplifier and the knee frequency only depends on technology parameters, and can not be changed by design parameters. The bandwidth requirement of the noise amplifier is minimal 1 MHz. This bandwidth determines the minimum knee frequency. The contribution of $1/f$-noise relative to the white noise of the amplifier can thereby not be reduced further.

This means that, to reduce the Flicker noise contribution, either the amplifier itself should have a low noise contribution to the output noise, the output noise has to be filtered, or noise shaping, for example by chopping, has to be applied.

To create a more flat noise frequency spectrum, three options are considered. First of all, reduction of the noise contribution of the amplifier. This approach will increase the power consumption of the amplifier significantly.

Secondly, the low-frequency noise can also be filtered out. This does not result in a white noise source. However, eliminating low-frequency noise is actually beneficial for performance, since the output of the comparator will also contain less low-frequency noise.

The third option is to apply chopping to the noise source. This will shift the frequency of the noise peak to the chopping frequency. However, it is observed in Appendix E that the performance is negatively affected by chopped pink noise. This is caused by the low-frequency variations of the instantaneous noise power, which affect the gain of the system.

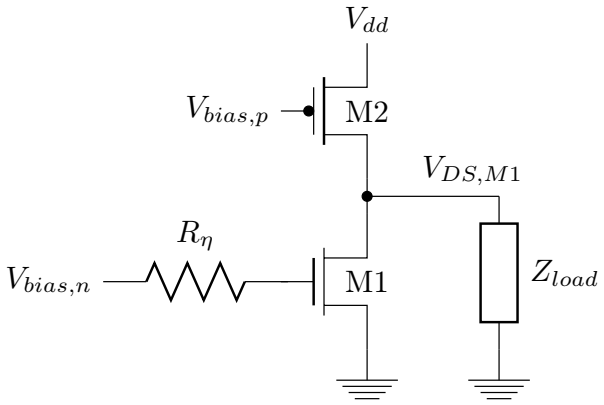


Figure 6.2.1: Single stage noise amplifier, with R_η to create the noise input.

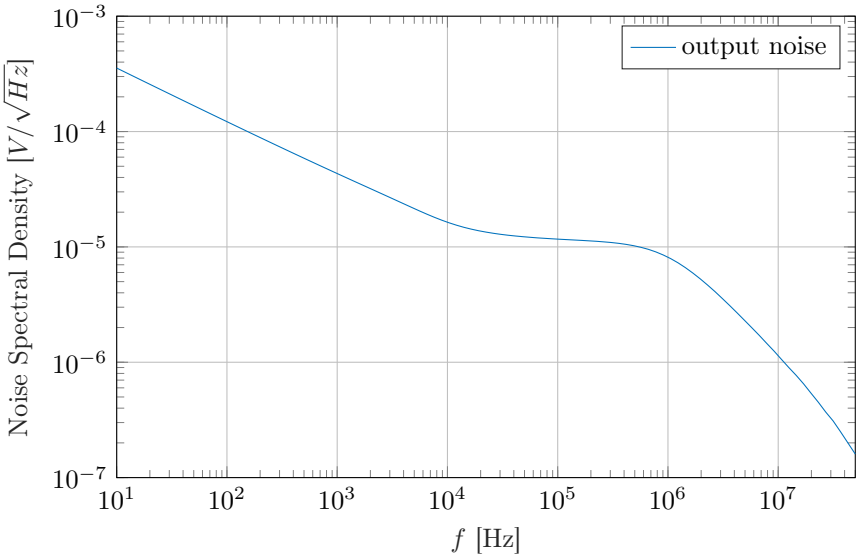


Figure 6.2.2: Output noise of the noise amplifier without load.

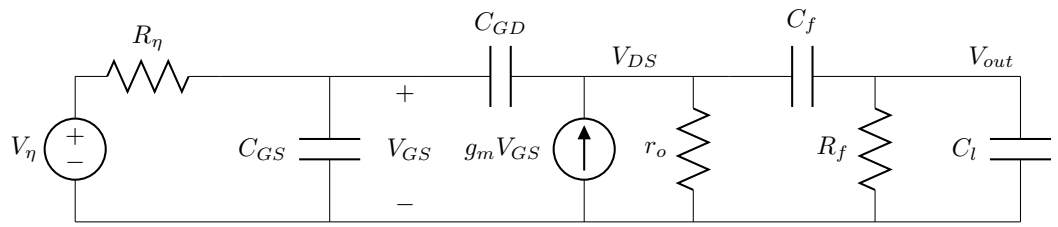


Figure 6.2.3: Small signal model of the noise amplifier, including high-pass filter. The output resistance $r_o = 1.29$ MΩ. The filter values are $C_f = 10$ pF and $R_f = 400$ KΩ. The input capacitance of the comparator, C_l, is 33 fF.

6.2.4 High-Pass Filter

The noise source is therefore implemented such that the amplifier drives a high-pass filter. Figure 6.2.3 shows the high-pass filter, including the small signal model of the amplifier. At low frequencies, the current flows through r_o. The filter capacitor $C_f = 10$ pF blocks the current, creating a highpass filter with a cut-off frequency:

$$f = \frac{1}{2\pi RC} = \frac{1}{2\pi(1.29 + 0.4) \cdot 10^6 \cdot 10 \cdot 10^{-12}} \approx 9.4 \text{ kHz} \tag{6.2.9}$$

In the passband, the gain is determined by the ratio between R_f and r_o:

$$\frac{V_{\text{out}}}{V_{G,M1}} = g_m \left(\frac{R_f \cdot r_0}{R_f + r_o} \right) \approx 12.8\times \tag{6.2.10}$$

The AC-response of the noise amplifier, including the filter, is shown in Figure 6.2.4. The high-frequency cut-off is approximately 3.9 MHz. The equivalent noise bandwidth is 6.1 MHz when assuming a first-order low-pass filter. The thermal noise of R_η over this bandwidth can be approximated by:

$$P_{\text{noise},R_\eta} = (4k_B T R \Delta f) \cdot 12.8^2 = 4.83 \cdot 10^{-5} \text{ V}^2 \tag{6.2.11}$$

$$V_{\text{noise,RMS}} = 6.95 \text{ mV} \tag{6.2.12}$$

The noise power filtered by the high-pass filter is hereby neglected since its bandwidth is only 10 kHz. The bandwidth of the noise is larger than the Nyquist rate of the sampler, the comparator. However, due to noise folding, the noise above 1 MHz will contribute to the total noise power.

The output power of the noise can be controlled by changing R_f. The noise level is not linearly proportional to the resistance, as shown in Equation 6.2.10. Moreover, the bandwidth will be increased when the resistance is reduced. To accurately implement the noise control, each noise level has to be converted into a resistance one-by-one to accurately implement the noise level control. Meanwhile, adding the switches to implement the control circuit adds a significant amount of parasitic capacitance which reduces the noise bandwidth.

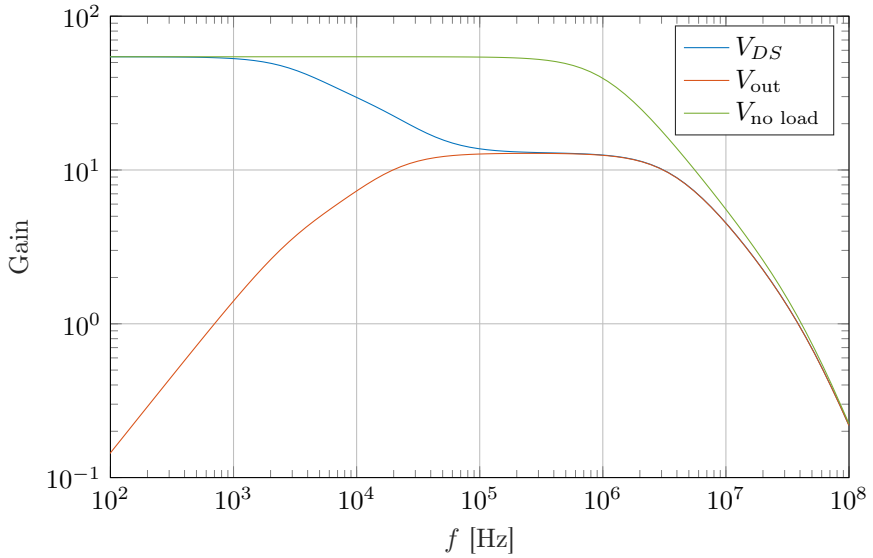


Figure 6.2.4: AC simulation of the noise amplifier with high-pass filter.

	Bandwith = 50 MHz		Bandwith = 216 Hz	
Total	6.24 mV		4.88 μV	
R_η	6.16 mV	97.11%	0.517 μV	1.1 %
R_f	0.173 uV	0.08 %	1.19 μV	5.99 %
M2	0.534 mV	0.73%	3.11 μV	40.52 %
M1	0.892 mV	2.04%	1.19 μV	5.94 %
M3	0.118 mV	0.04%	3.11 μV	40.51 %
M4	0.032 mV	0.00%	1.19 μV	5.94 %

Table 6.6: Output noise levels in the full simulated bandwith and in the signal bandwith.

6.2.5 Simulated Output Noise

The full simulated circuit, including the biasing, is shown in Figure 6.2.5. The differential noise output is measured between the $V^+_{\eta,out}$ and the bias voltage $V_{b,n} = V^-_{\eta,out}$. The bias current I_{bias} is 2.3 μA. The M1 and M3 are sized $W/L = 2\mu/0.5\mu$ and M2, M4 and M5 are sized $W/L = 6\mu/0.5\mu$.

The total simulated output noise is 6.24 mV, of which 97.1% originates from R_η. In Table 6.6, the noise contributions of each component are shown. While the total noise is dominated by the thermal noise of R_η, the noise contributions change significantly when only the signal bandwidth up to 216 Hz is considered. For low frequencies, the noise is dominated by the transistors M1 and M3. The total noise in the signal bandwidth is 4.88 μV, which corresponds with $6.1 \cdot 10^{-7}$ of the total noise power. The noise spectra of the significant components are shown in Figure 6.2.6.

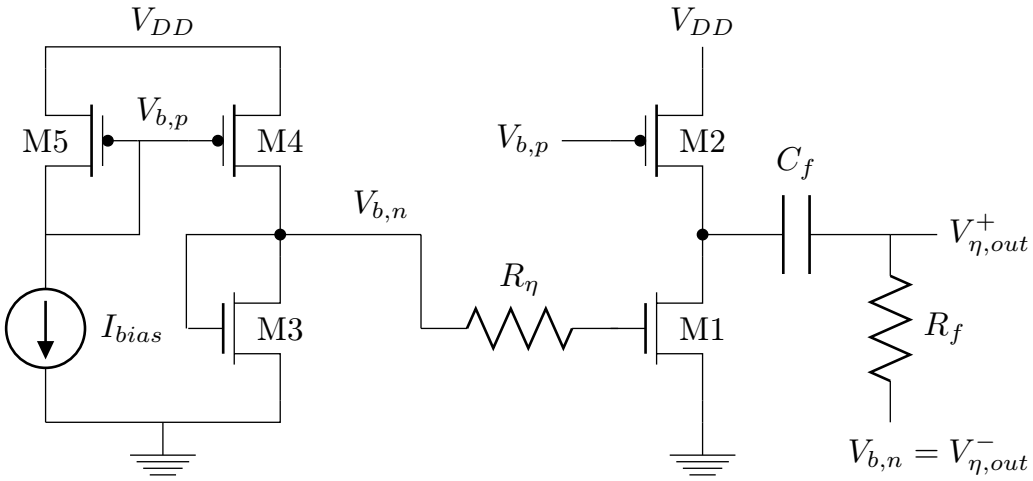


Figure 6.2.5: Schematic of the noise source.

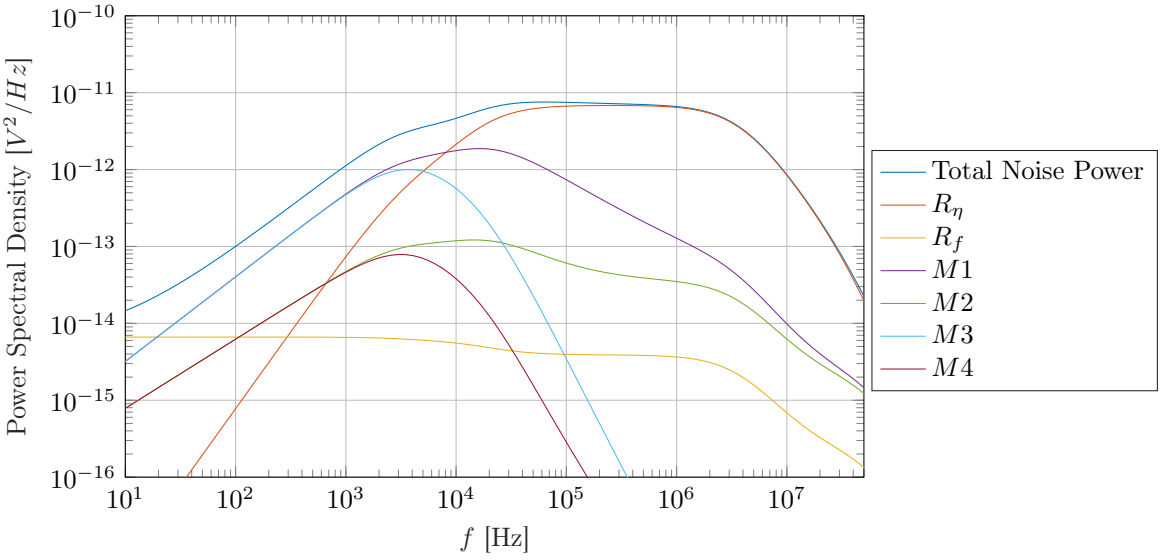


Figure 6.2.6: Output noise spectra of different components of the system.

6.3 Simulation Limitations

The proposed implementations are only simulated separately. Simulations of the full, operating system on circuit level could not be performed. The cause for this is limitations within the simulator and the specific properties of this system.

6.3.1 Periodic Analysis

Clocked systems can normally be simulated using periodic analysis. The AC and noise behavior is simulated in 3 steps. First, the periodic steady state is simulated. Clocked systems do not have a true steady state since switches will always introduce non-linear behavior. A steady state can be derived based on the periodic behavior of the system. Transient simulations are performed to derive a periodic operating point. Based on the linearization at this periodic operating point, the AC behavior and the noise behavior can be derived.

A SR-ADC cannot be simulated using this approach. The (linear) transfer of the system is based on stochastic properties and not on periodically linear behavior. The (linear) small signal model which forms the basis of this simulation is no useful model for simulating stochastic resonance.

Moreover, the periodic analysis tools allow to perform a periodic AC and periodic noise analysis separately. It is not possible to incorporate both signal and noise in one simulation, which is required to observe stochastic resonance.

6.3.2 Transient Noise Analysis

The only available simulation which can model stochastic resonance on circuit level is the transient noise analysis. This analysis is a transient simulation, in which the noise models are incorporated. The noise levels are calculated based on the power spectral density of the noise sources. By calculating the inverse Fourier transform, a noise value is derived for each simulation point.

Noise Bandwith

To perform this simulation, the simulator needs the upper and lower boundaries of the simulated noise spectrum. The system is based on a sampled comparator, so noise sampling occurs. The maximum noise bandwidth should, therefore, be much higher than the sampling frequency since the maximum frequency noise frequency which is sampled is determined by the aperture time. The lower boundary of the simulated noise needs to be very low. Since the input noise inside the frequency bandwidth is of importance to the performance of the system, the minimum frequency is ideally close to 0.5 Hz.

Maximum Time Steps and Simulation Time

To simulate the given maximum noise frequency, the maximum time step of the simulator is given by one over the Nyquist rate. The maximum time-step of the simulator is hereby very small.

In a normal transient simulation of a comparator, the time steps vary a lot over time. Around the switching moments, and when the comparator is taking a decision, the system

behaves highly nonlinear, and thus small time-steps are taken. However, after the decision has been made, the time steps can be large, since the currents and voltages remain constant until the next edge of the clock. Especially simulations of clocked comparators can speed up a lot by this adjustment of the time steps. During the state transitions, the steps are very small due to fast, non-linear variations, while the time steps can be very large when the system is settled. In a transient noise analysis, the time step is constantly at the fast rate which is necessary to calculate the transitions.

The total required simulation time is determined by the minimum noise frequency which needs to be simulated. To simulate the noise in the signal bandwidth, the required simulation time is several milliseconds, let alone simulating the effects of the control loop in the circuit simulation.

The large time constants in the system combined with the sampling of high-frequency noise by the comparator lead to extremely large simulation times. As a reference, the simulation time required to determine the noise level of the comparator is taken. To simulate in total 60 ms, the simulator had to run several days to only simulate the comparator.

Based on these observations it is concluded that full system simulations could not be performed within the scope of this project.

6.4 Conclusions and Discussion

In this chapter, an implementation of the comparator and the noise source are presented. This section is build up as follows. First, the results of the comparator and the noise source implementation are discussed separately. The results of the comparator design and the noise source designed are combined in the discussion.

6.4.1 Comparator

A StrongARM latch is proposed to implement the comparator. The main focus in the comparator design has been the threshold compensation. To compensate for the offset, adjustable capacitances are added to the several nodes in the comparator. This technique allows adjusting the offset voltage to an accuracy of 50 μV. The initial, uncompensated, offset is reduced by choosing large transistors to implement the comparator. Based on the uncompensated offset, the offset is designed such that compensation can be applied over a range of $\pm$10 mV. The capacitors, added to implement the offset compensation, reduce the speed, and thereby the input referred noise of the comparator. The simulated comparator noise is 0.64 mV.

The kickback effects are shown to be critical to the performance of the comparator and are inherent to the fast switching of the comparator. For a 50 KΩ source impedance, on one of the two inputs, 8% gain mismatch is simulated. Moreover, a mismatch between the two branches of one differential input results in 145 μV/KΩ. While electrodes typically have a large source impedance, which is not very well matched, measures will have to be taken to reduce this error. To effectively reduce the kickback noise, the source impedance of the inputs should be reduced, since the design freedom in the other parameters (current and gate-source capacitance) is limited and affect other aspects of the comparator design.

The four input implementation of the comparator reduces the kickback effects when the impedances of the noise source and the signal source are unmatched. However, it also introduces an added complexity to the system and new inaccuracies in terms of gain mismatch, which still needs to be taken into account.

6.4.2 Noise Source

The noise source delivers 6.4 mV wideband, Gaussian noise, based on the amplified thermal noise of a resistor. The contribution of the flicker noise is reduced by a high-pass filter, and the output can be controlled by changing the output impedance of the filter.

While the basic specifications of the noise source are met, on several points, the noise source can be improved significantly. The single-ended implementation which uses a reference voltage as second input has an asymmetric output impedance. Since it is shown that the comparator is sensitive to source impedance mismatches, a fully differential implementation is more promising if this noise source is implemented in the full system.

Moreover, the simulated power consumption of the noise source is 2 times the power of the comparator, excluding the biasing circuit. This is a fundamental drawback of the chosen implementation. Alternative noise sources which are considered show the same fundamental limitation. The maximum required noise level requires power consuming, active elements.

6.4.3 Conclusions on the Combining the Comparator and Noise Source

In Section 6.3, it is shown that simulating the stochastic resonance in a full system simulation is not feasible. However, based on the simulation results of the separate blocks, a number of conclusions can be drawn.

The proposed comparator shows a significant sensitivity to the source impedances and the bias mismatches. These non-idealities are inherent to a (fast) switching comparator. While optimization is certainly possible, a more fundamental solution is expected to realize better results in a more efficient way.

Meanwhile, the implementation of the noise source can not meet the requirements of the comparator regarding the source impedances. Especially the proposed noise control, implemented by changing the load resistance of the amplifier, will affect the performance of the comparator. The gain of the noise input pair will reduce significantly if the noise level is high.

Significant improvements can be made regarding the design of the noise source. For example, a full differential implementation will eliminate inequality in the output impedance of the two output voltages, and other design choices can be made to specifically address the kickback effects in the comparator.

However, it can be concluded that the requirements and the complexity of the system will increase in order to reduce the errors sufficiently. Hereby, an increase in power consumption is expected. An alternative approach would be to use the insight created in this chapter to critically evaluate the system level design and to find a more fundamental solution.

6.4.4 Discussion

One of the research questions was, whether it is possible to implement a SR-ADC without pre-amplification. Based on the observations above, it is concluded that, in this system, an implementation with pre-amplification is more promising. The main argument for doing this research towards an implementation with no pre-amplification was the notion that the absolute value of the input signal is of no importance, as long as the optimal noise level in the system is achieved. However, secondary properties of the input, such as source impedance, and the secondary effects of the comparator, such as kickback noise, showed to have a crucial effect on the performance. Therefore, the consequences of adding a pre-amplification stage are considered.

When a (controllable) pre-amplifier is designed, the requirements on the comparator and the noise source can be relaxed significantly. A large advantage is found regarding the noise source. When the signal amplitude is adjusted to the right level, the noise source needs no control. When the noise does not need to be controlled, it does not necessarily need to be implemented in a separate block.

Furthermore, an argument not to use transistor noise as the main noise source is the flicker noise which dominates the low-frequency noise. The realization of noise folding in the sampled comparator opens possibilities to look towards an implementation which uses comparator noise as the main noise source.

Another advantage of a pre-amplifier is that it can take care of correct biasing and eliminate or reduce the effects of drift and the common mode signal. It is shown that a common mode mismatch translates into a considerable gain mismatch. ECG measurements are typically prone to relatively large common mode signals and drift. The implemented system should be insensitive to these properties of the input signal.

7

Conclusions

In this thesis, an explorative study in stochastic resonance for analog-to-digital conversion is presented. The study comprises a mathematical analysis of stochastic resonance in a 1-bit quantizer, a system level design of an analog-to-digital converter and an implementation of the comparator and the noise source.

The main novelties found in this project are:

- Development of an analytical method to describe the behavior, and to determine the performance of SR in a 1-bit quantized system.
- Introduction of stochastic resonance using a negative hysteresis.
- Design of a closed-loop system which controls the ratio between the input signal and noise such that the system always operates in the stochastic resonance peak, independent of signal input amplitude and the noise in the system.

Finally, a study into the implementation has been performed. While, this implementation study did not lead to concrete results, in terms of (simulations of) a fully operating design, several steps have been taken and various lessons have been learned regarding the challenges in the implementation of a SR-ADC. With this new insight, possible solutions to the challenges are discussed in the Recommendations (Chapter 8).

The conclusions are presented as follows. First, the different steps taken in this study are concluded separately in an overview presented in Section 7.1. The results of the different steps are combined in Section 7.2. Finally, the scientific contributions presented in this work are summarized in Section 7.3.

7.1 Overview

The first goal was to determine the behavior and to analyze the performance of a SR-ADC. First, the phenomenon stochastic resonance is classified into 4 different classes, based on two characteristics: Subthreshold vs. suprathreshold, and with hysteresis vs. without hysteresis. The classification has been performed by translating the observed mechanisms to a comparator based system and shows the most promising results for suprathreshold SR without hysteresis.

Analytical Derivation

In Chapter 4, a comprehensive analysis is presented of comparator based stochastic resonance. The system consists of a summation of a signal input and noise, which is applied to an ideal, clocked, comparator, with the reference input set to zero.

The expected output is determined, and based on this expected output, the harmonic distortion and output noise are calculated. For an oversampling ratio of 5000× (i.e. $f_s = 10000 f_{-3\text{dB}}$) and a first-order Butterworth output filter, the SNDR of the system without threshold is 26.9 dB in the SR peak.

Although the classification suggests, and literature reports, that the best performance is obtained without hysteresis, suprathreshold stochastic resonance with hysteresis is analyzed as well. It is shown that introducing a negative hysteresis increases the linearity and decreases the noise power in the passband relative to the total noise power. This novel technique shows a potential SNDR increase of 6.4 dB, up to 33.3 dB.

System Level Design

The second goal is to design a system, which enables closed-loop operation of the SR-ADC. A multi-rate digital filter with a rectangular window is designed. The filter converts the 1-bit 2 MHz signal to a 7-bit 488 Hz signal. The resulting SNDR is 27.3 dB.

To guarantee that the system always operates at the right noise level and at zero-offset, a closed-loop system is designed. Two feedback loops are designed. The first feedback loop ensures that the ratio between the input amplitude and the noise is always such that the system operates at the stochastic resonance peak. The loop controls the noise level based on the measured output amplitude of the system. An algorithm is implemented to reduce the sensitivity to random fluctuations in the maximum output amplitude. Finally, by implementing the noise level adjustments in a logarithmic fashion, the control loop keeps the system close to the optimal noise value. An SNDR of 27.0-27.2 dB is simulated for an input amplitude range of 10×, showing that the noise level is automatically adjusted to the input amplitude.

Secondly, the threshold level is controlled by a negative feedback loop with a digital integrator, creating a high-pass filter. It is shown that the quantization noise, caused by the limited resolution of the DAC is critical to the performance. The proposed threshold compensation technique requires LSB step-size of the DAC smaller than $0.02 \cdot A_{\text{in}}$ to limit the SNDR reduction to 0.15 dB.

The proposed system level design keeps the SR-ADC in the right operation point, independent of the input amplitude and other noise sources. The threshold is kept at the desired DC-level of the signal input, compensating for any source of offset.

Implementation

Finally, the implementation of the noise source and the comparator is studied.

Comparator The comparator is implemented using a 4 input StrongARM latch. The standard deviation of the threshold of the comparator is $\sigma_\theta = 3.9\text{mV}_{\text{RMS}}$, and its input referred noise is 0.64 mV_{RMS}. A threshold compensation technique is implemented which adjusts the threshold in steps of $35 - 50$ μV, over a range of $+/-10$ mV, using adjustable capacitors in both branches of the comparator.

The kickback effects introduce a critical sensitivity to the source impedances of the inputs and the common mode voltage mismatches. Connecting the comparator directly to the electrodes will result in significant inaccuracies. In ECG signals typically significant variation in source impedances, and large common mode signals and drift are observed. For this reason, it is concluded that the most promising step towards an implemented SR-ADC is to step away from the implementation without pre-processing of the electrode signals. This creates several opportunities which will be discussed further in the Recommendations in Chapter 8.

Noise Source To implement a controllable noise source which delivers enough noise power to keep the system in its stochastic resonance peak, an active implementation is required. The noise source is implemented using the thermal noise of a resistor, amplified by a single gain stage, and filtered by a high-pass filter. By adjusting the resistor value in this filter, the output noise power of the amplifier can be controlled. The noise source delivers maximal 6.24 mV noise over a bandwidth of 10 kHz to 4 MHz, while using 2.8 μW of power. This excludes the power used by the biasing circuit.

The power required to deliver the required noise exceeds the power consumption of the comparator significantly. The need for an amplifier has a significant impact on the advantages of SR analog-to-digital conversion. One of the goals, after all, was to perform the conversion without precise pre-amplification. The advantage of not having a pre-amplifier largely disappears when the amplifier needs to be implemented at another point in the system.

7.2 Conclusions on the SR-ADC

Overall it can be concluded that potentials have been found in fully stochastic resonance based analog-to-digital conversion. In applications where relatively low resolution is required, but power requirements are strict, an SR-ADC is an interesting alternative to conventional ADCs. The research into stochastic resonance analog-to-digital conversion is still in a very early stage. More extensive research is required to fully exploit all the potentials of SR. Therefore, the value comparison of the result with other, existing, ADCs is little. At this point, the well studied, and highly optimized ADC structures and implementations presented in literature outperform system presented in this thesis. Meanwhile, many steps can be taken to improve on this system.

While most goals of this thesis project have been met, and most research question are confirmed, the goal to design a system which operates without any pre-amplification has shown not to be the optimal solution. Difficulties regarding directly connecting the input signal to the comparator where hard to overcome and the controllable noise source is relatively power hungry. It is concluded that an implementation in which a pre-amplifier is

added to the system will result in a better performance. The potentials created by the implementation with a pre-amplifier are further discussed in the Recommendations (Chapter 8).

7.3 Contributions

First of all, a comprehensive analysis of the behavior of stochastic resonance in a 1-bit quantizer to perform analog-to-digital conversion had not been found in the literature. By developing a mathematical description the performance of the system can be predicted. This derivation is published at the International Symposium on Circuits and Systems Conference 2018 in Florence [37]. Furthermore, in addition to this derivation, model to predict the performance of a SR-ADC using a quantizer with hysteresis is presented. This model shows that a negative hysteresis can boost the performance significantly. In the literature, no reports have been found describing stochastic resonance in a quantizer with negative hysteresis.

A closed-loop system-level design is proposed. By adjusting the noise level and the threshold via feedback loops, a controlled system is created which ensures operation in the correct operating point. While exact noise levels and the input amplitude, as well as the ideal threshold level, cannot always be predicted accurately, with this closed loop system a step is taken towards the practical implementation of an analog-to-digital converter fully based on stochastic resonance.

8

Recommendations and Future Work

A number of interesting results have been found during the project. However, not all interesting observations could fully be worked out, or incorporated in the system. Several observations interesting for further research are discussed in this chapter.

Moreover, the study towards the electronic implementation revealed difficulties which raised the question whether the solution would be on implementation level or if solving the issues on system level would lead to a more efficient solution.

8.1 Recommendations for Implementation

Several issues regarding the implementation are mentioned in the Conclusions that could not directly be solved on the implementation level. By reconsidering the use of a pre-amplifier, some interesting potentials are observed.

8.1.1 Noise Source and Pre-Amplification

The noise source requires an active, amplifying element to achieve the necessary noise levels. This introduces a fundamental limitation to the power efficiency of the system. The largest drawback in this solution is the required bandwidth of the noise amplifier.

A potential alternative to implementing the noise would be by using the comparator noise. During the study on the noise source implementation, the focus was on the implementation of a controllable, and thus separate, noise source. An interesting observation is the effect of noise folding, which creates more white-like noise.

Using the comparator as the noise source will complicate the control of the noise. However, the current solution needs a noise amplifier. The power saved by removing the separate noise source can be invested in a pre-amplifier. The requirements of this pre-amplifier regarding the output swing and noise power can still be relaxed compared to a conventional LNA solution, and the required bandwidth to amplify the signal very small compared to the bandwidth of the noise amplifier. Control over the ratio between the signal amplitude and the noise level can be implemented in the pre-amplifier.

Another advantage of this implementation is that the remaining issues such as the input impedance (matching) and common mode matching are reduced by a well-designed the pre-amplifier. Moreover, if the noise of the comparator is employed, no physical noise input is required, which simplifies the design.

8.1.2 Comparator

Currently, the offset compensation is implemented using controllable capacitors at internal and the output nodes. This was found to be the most promising technique to realize a precise offset compensation. Meanwhile, this technique reduces the comparator noise via the added capacitances, which is in the current solution required to be able to handle the smallest input amplitudes.

By shifting towards an implementation whereby the signal is pre-amplified, and rerouting the noise control loop such that it controls the gain of the pre-amplifier, the comparator (noise) can be designed for a specific, higher, input amplitude. This means that for the purpose of noise reduction, the added capacitances are not required anymore.

The current offset compensation technique does not allow compensation over a large range and relies on changing the added capacitances. However, a compensation technique has been found that adjusts the voltage on one leg of a comparator by means of a charge pump [38]. This technique allows adjustments over a wider range and does not require added capacitances.

The total energy required to perform the comparison is determined by the total capacitance that needs to be charged. By removing the capacitances, the power consumption can be reduced significantly (to 570 nW by only removing the added capacitors), and the transistors can be sized smaller since a larger initial offset can be compensated. The latter reduces the power consumption further. The comparator design can be focussed on design

for the right noise value.

8.1.3 Hypothesis

By implementing a pre-amplifier, and by controlling the gain of the amplifier towards a certain input signal amplitude, the noise of the comparator can be used as a noise source. Furthermore, the threshold compensation can be implemented by adjusting the voltage on one leg using a charge pump. The comparator can then be implemented with only two legs, one being the signal input, the other being the threshold voltage. Kickback errors are reduced since in this implementation they only lead to a DC-error which can be compensated by the charge pump offset compensation. The requirements on the comparator are relaxed and the energy per conversion step decreases significantly.

8.2 Negative Hysteresis Implementation

To limit the scope of the project, the negative hysteresis is not further implemented in the system level design or the implementation. However, a significant increase in performance is observed. As mentioned earlier in the thesis, controlling the size of the hysteresis might not be straightforward. This holds especially if the input signal amplitude is not known.

In the implementation suggested above, the noise level of the comparator is known and constant, and thus the optimal hysteresis size is constant as well. The hysteresis size can be determined in the design phase, without the need for an extra control loop.

However, for precise implementation of the hysteresis, a control loop could be beneficial to the performance. For example to reduce the effect of PVT variations. An interesting option to study would be controlling the hysteresis size by calculating the switch rate of the comparator. The switch rate increases for larger hysteresis sizes. This control loop will need to run without a signal applied to the system since the input amplitude will change the switch rate of the comparator as well.

When the noise level is adjusted, the hysteresis size needs to be adjusted accordingly. Therefore implementing this technique is most feasible in a system whereby the input amplitude is controlled rather than the noise level.

8.3 Digital Post-Processing

Digital post-processing is not treated in the thesis, but can potentially result in a large increase of the SNDR. The thesis project focussed on achieving a direct linear transfer from input to output. It is not considered that the non-linearity in the transfer can be compensated by digital post-processing. When the inverse error function is applied to the output of the digital filter, the error due to harmonic distortion is compensated. The output error is in this case only determined by the noise. First simulations of this system show a maximum SNDR of 38.5 dB, an increase of almost 12 dB, for a system with $f_s = 10,000 \cdot f_{-3\text{db}}$, using a first-order Butterworth filter in a zero-threshold system.

8.4 Discrete Component Implementation

Simulating the system required a high amount computation power. Because of this, full system simulations were not possible within the scope of this project. However, further characterization of the system full system, either with accurate simulations or by measurements will be necessary to completely verify the system. An alternative is to develop a first prototype using discrete components. This proof of concept will be limited by the characteristics of the commercially available components. However, the system itself is relatively easy to scale and the system level design is, while the implentation is fully digital, relatively easily transferred into a discrete component design.

Bibliography

[1] Tara Julia Hamilton, Saeed Afshar, AndrÃ© van Schaik, and Jonathan Tapson. Stochastic electronics: A neuro-inspired design paradigm for integrated circuits. *Proceedings of the IEEE*, 2014.

[2] S. Fauve and F. Heslot. Stochastic resonance in a bistable system. *Physics Letters*, 1983.

[3] Iec060601-2-47 - medical electrical equipment - part 2-47: Particular requirements for the basic safety and essential performance of ambulatory electrocardiographic systems.

[4] Jaakko Malmivuo and Robert Plonsey. *Bioelectromagnetism - Principles and Applications of Bioelectric and Biomagnetic Fields*. Oxford University Press, 1995.

[5] Roberto Benzi. Stochastic resonance in climate change. *Tellus*, 1982.

[6] J.J. Collins, Carson C. Chow, Ann C. Capela, and Thomas T. Imhoff. Aperiodic stochastic resonance. *Physical Review E: Statistical, Nonlinear, and Soft Matter Physics*, 1996.

[7] J.J. Collins, C.C. Chow, and T.T. Imhoff. Aperiodic stochastic resonance in excitable systems. *Physical Review E: Statistical, Nonlinear, and Soft Matter Physics*, 1995.

[8] N.G. Stocks. Information transmission in parallel threshold arrays: Suprathreshold stochastic resonance. *Physical Review E: Statistical, Nonlinear, and Soft Matter Physics*, 2001.

[9] N.G. Stocks and R. Manella. Generic noise-enhanced coding in neuronal arrays. *PHYSICAL REVIEW E*, 2001.

[10] Roberto Benzi, Alfonso Sutera, and Angelo Vulpiani. The mechanism of stochastic resonance. *Journal of Physics A: Mathematical and General*, 1981.

[11] J.P. Eckmann and L.E. Thomas. Remarks on stochastic resonances. *Journal of Physics A: Mathematical and General*, 1982.

[12] A.R. Bulsara and F.E. Moss. Single neuron dynamics: Noise-enhanced signal processing. *IEEE International Joint Conference on Neural Networks*, 1991.

[13] A. Bulsara, E.W. Jacobs, T. Zhou, F. Moss, and L. Kiss. Stochastic resonance in a single neuron model: Theory and analog simulation. *Journal of Theoretical Biology*, 1991.

[14] A. Longtin, A. Bulsara, and F. Moss. Time-interval sequences in bistable systems and the noise-induced transmission of information by sensory neurons. *The American Physical Society*, 1991.

[15] J.K. Douglass, L. Wilkens, E.Pantazelou, and F. Moss. Noise enhancement of information transfer in crayfish mechanoreceptors by stochastic resonance. *Nature*, 1993.

[16] J.E. Levin and J.P. Miller. Broadband neural encoding in the cricket cercal sensory system enhanced by stochastic resonance. *Nature*, 1996.

[17] F. Moss, L.M. Ward, and W.G. Sannita. Stochastic resonance and sensory information processing: a tutorial and review of application. *Clinical Neurophysiology*, 2004.

[18] Mark D. McDonnell and Derek Abott. What is stochastic resonance? definitions, misconceptions, debates, and its relevance to biology. *PLoS Computational Biology*, 2009.

[19] Hakon A. Hjortland. *Sampled Continuous-Time 1-Bit Signal Processing in CMOS for Wireless Sensor Networks*. PhD thesis, University of Oslo, 2016.

[20] Gregory P. Harmer, Bruce R. Davis, and Darek Abott. A review of stochastic resonance: Circuits and measurement. *IEEE Transactions on Instrumentation and Measurement*, 2002.

[21] Mark D. McDonnell, Nigel G. Stocks, Charles E.M. Pearce, and Derek Abott. *Stochastic Resonance: From Suprathreshold Stochastic Resonance to Stochastic Signal Quantization*. Cambridge University Press, 2008.

[22] Insani Abdi Bangsa. System building blocks for methematical operators using stochastic resonance: Application in an action potential detection system. Master's thesis, Delft University of Technology, 2017.

[23] Roy D. Yates and David J. Goodman. *Probability and Stochastic Processes*. John Wiley & Sons, Inc., 2005.

[24] Dimitris G. Manolakis John G. Proakis. *Digital Signal Processing - Principles, Algorithms and Applications*. Pearson, 1996.

[25] Daniel Schinkel, Eisse Mensink, Eric Klumperink, Ed van Tuijl, and Bram Nauta. A double-tail latch-type voltage sense amplifier with 18ps setup+hold time. *International Conference on Solid State Circuits*, 2007.

[26] Ili Shairah Abdul Halim, Siti Lailatul Mohd Hassan, Nurul Dalila binti Mohd Akbar, and A. Afhzan Ab. Rahim. Comparative study of comparator and encoder in a 4-bit flash adc using 0.18ÎŒm cmos technology. *International Symposium on Computer Applications and Industrial Electronics*, 2012.

[27] Vladimir Milovanovic and Horst Zimmermann. A two-differential-input / differential-output fully complementary self-biased open-loop analog voltage comparator in 40 nm lp cmos. *International Conference on Microelectronics*, 2014.

[28] Titu Mary Ignatius, Jobin.K.Antony, and Silpa Rose Mary. Implementation of high performance dynamic flash adc. *International Conference on Magnetics, Machines and Drives*, 2014.

[29] T. Kobayashi, K. Nogami, T. Shirotori, and Yukihiro Fujimoto. A current-controlled latch sense amplifier and a static power-saving input buffer for low-power architecture. *IEEE Journal of Solid-State Circuits*, 1993.

[30] Michiel van Elzakker, Ed van Tuijl, Paul Geraedts, Daniel Schinkel, Eric Klumperink, and Bram Nauta. A 1.9ÎŒw 4.4fj/conversion-step 10b 1ms/s charge-redistribution adc. *IEEE International Solid State Circuits Conference*, 2008.

[31] Yun-Ti Wang and Behzad Razavi. An 8-bit 150-mhz cmos a/d converter. *IEEE Journal of Solid-State Circuits*, 2000.

[32] A.A. Abidi. Analysis and design of regenerative comparators for low offset and noise. 2018.

[33] Koon Lun Jackie Wong. Comparison of digital offsetcompensation in comparators. Master's thesis, University of California Los Angeles, 2002.

[34] M. Jeeradit, J. Kim, B. Leibowitz, P. Nikaeen, V. Wang, B. Garlepp, and C. Werner. Characterizing sampling aperture of clocked comparators. *Symposium on VLSI Circuits Digest of Technical Papers*, 2008.

[35] Abinash Mohanty, Ketul B. Sutaria, Hiromitsu Awano, Takashi Sato, and Yu Cao. Rtn in scaled transistors for on-chip random seed generation. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 2017.

[36] E. Simoen, B. Kaczer, M. Toledano-Luque, and C. Claeys. Random telegraph noise: From a device physicist's dream to a designer's nightmare. *ECS Transactions*, 2011.

[37] I.A. Bangsa, D.P.N. Mul, and W.A. Serdijn. Stochastic resonance mixed-signal processing: Analog-to-digital conversion and signal processing employing noise. *International Symposium on Circuits and Systems*, 2018.

[38] S. Liu, T. Rabuske, and J. Paramesh. Analysis and background self-calibration of comparator offset in loop-unrolled sar adcs. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 2017.

A

Derivation of Signal Transfer of a 1-Bit Quantizer in SR

The expected output $\hat{y}$ can be written in probabilities of the states multiplied with the value of each state:

$$\begin{aligned}\hat{y}\left(x,\eta\right) &= \mathbf{E}\left[y|x\right] \\ &= P\left(y_o = 1|x\right) - P\left(y_o = -1|x\right) \\ &= P(\eta > -x|x) - P(\eta < -x|x)\end{aligned} \tag{A.0.1}$$

Since the quantizer is always in one of the two states the following equation holds:

$$\begin{aligned}P(\eta > -x|x) + P(\eta < -x|x) &= 1 \\ P(\eta > -x|x) &= 1 - P(\eta < -x|x)\end{aligned} \tag{A.0.2}$$

Substituting A.0.1 into A.0.2, $\hat{y}$ can be rewritten:

$$\hat{y} = 1 - 2 \cdot P(\eta < -x|x) \tag{A.0.3}$$

The probability density function of Gaussian noise is defined as:

$$p_G(x) = \frac{1}{\sigma\sqrt{2 \cdot \pi}} \cdot e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \tag{A.0.4}$$

where σ is the standard deviation and μ is the mean value of the noise [23], page 119.

The probability $P(\eta < -x|x)$ can be calculated by integrating the cumulative distribution function shown. The mean value μ of the considered noise is zero.

$$P(\eta < -x|x) = \int_{-\infty}^{-x} \frac{1}{\sigma_\eta \sqrt{2 \cdot \pi}} \cdot e^{-\frac{\eta^2}{2\sigma_\eta^2}} d\eta \tag{A.0.5}$$

The integral can be written as the error function, defined as [23], page 142:

$$erf(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-u^2} du$$

Equation A.0.5 can hereby be rewritten to:

$$\begin{aligned} P(\eta < -x|x) &= \int_{-\infty}^{-x} \frac{1}{\sigma_\eta \sqrt{2 \cdot \pi}} \cdot e^{-\frac{\eta^2}{2\sigma_\eta^2}} d\eta \\ &= \int_{-\infty}^{0} \frac{1}{\sigma_\eta \sqrt{2 \cdot \pi}} \cdot e^{-\frac{\eta^2}{2\sigma_\eta^2}} d\eta - \int_{-x}^{0} \frac{1}{\sigma_\eta \sqrt{2 \cdot \pi}} \cdot e^{-\frac{\eta^2}{2\sigma_\eta^2}} d\eta \end{aligned} \tag{A.0.6}$$

Since:

$$\int_{-\infty}^{0} \frac{1}{\sigma_\eta \sqrt{2 \cdot \pi}} \cdot e^{-\frac{\eta^2}{2\sigma_\eta^2}} d\eta = \frac{1}{2} \tag{A.0.7}$$

The probability $P(\eta < -x|x)$ can be written into:

$$P(\eta < -x|x) = \frac{1}{2} - \int_{-x}^{0} \frac{1}{\sigma_\eta \sqrt{2 \cdot \pi}} \cdot e^{-\frac{\eta^2}{2\sigma_\eta^2}} d\eta \tag{A.0.8}$$

The probability density function is symmetrical around zero, therefore:

$$P(\eta < -x|x) = \frac{1}{2} - \int_{0}^{x} \frac{1}{\sigma_\eta \sqrt{2 \cdot \pi}} \cdot e^{-\frac{\eta^2}{2\sigma_\eta^2}} d\eta \tag{A.0.9}$$

The integral can be rewritten and substituted for the error function:

$$\begin{aligned} P(\eta < -x|x) &= \frac{1}{2} - \frac{1}{\sigma_\eta \sqrt{2 \cdot \pi}} \cdot \int_{0}^{x} e^{-\frac{\eta^2}{2\sigma_\eta^2}} d\eta \\ &= \frac{1}{2} - \frac{1}{\sigma_\eta \sqrt{2 \cdot \pi}} \int_{0}^{x} e^{-u^2} d\left(u \cdot \sqrt{2}\sigma_\eta\right), \text{ where } u = \frac{\eta}{\sqrt{2}\sigma_\eta} \\ &= \frac{1}{2} - \frac{1}{\sigma_\eta \sqrt{2 \cdot \pi}} \cdot \frac{\sqrt{\pi}}{2} \text{erf}\left(\frac{x}{\sqrt{2}\sigma_\eta}\right) \cdot \sqrt{2}\sigma_\eta \\ &= \frac{1}{2} - \frac{1}{2} \text{erf}\left(\frac{x}{\sqrt{2}\sigma_\eta}\right) \end{aligned} \tag{A.0.10}$$

By substituting Equation A.0.6 in A.0.3, the expected output of the system is derived:

$$\begin{aligned}\hat{y}(x,\sigma) &= 1 - 2 \cdot P(\eta < -x|x) \\ &= 1 - 2 \cdot \left(\frac{1}{2} - \frac{1}{2}\text{erf}\left(\frac{x}{\sqrt{2}\sigma_\eta}\right)\right) \\ &= erf\left(\frac{x}{\sqrt{2}\sigma_\eta}\right)\end{aligned} \tag{A.0.11}$$

B

Simulink Simulation Models of the Control Loops

This appendix comprises figures showing the Simulink simulation models used for the system simulations in Chapter 5. A short description of the functionallity of each block is added below each figure.

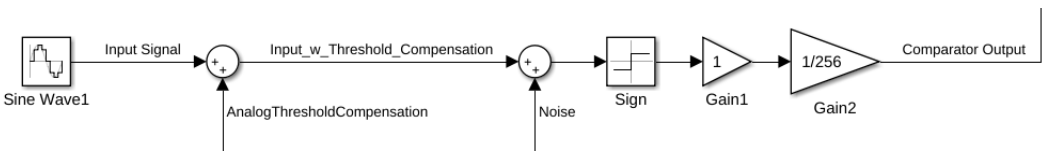


Figure B.0.1: Input Section of the Simulink Model. The offset compensation and the noise are added to the signal input. This signal is applied to a sign function. The gain block Gain1 is used to transform the signal to an unsigned 8 bit integer, which is either 0 or 1.

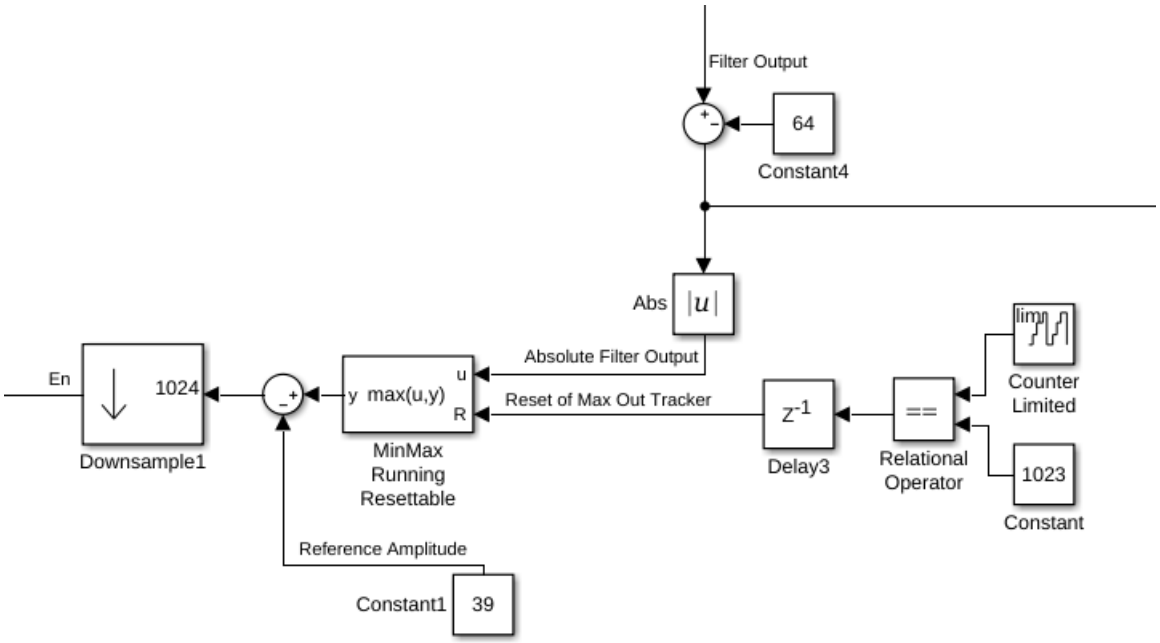


Figure B.0.2: The Maximum Output Amplitude Tracker. First the offset is subtracted from the filter output, after which the absolute value is calculated. A reset signal is generated which resets every 1024 samples. The reset signal is designed such that the MinMax Running Resettable block is resetted directly after the Downsample1 samples. The output of the MinMax Running Resettable is compared to the Reference Amplitude before it is downsampled. The output of this block is the signal E_n, the output amplitude error.

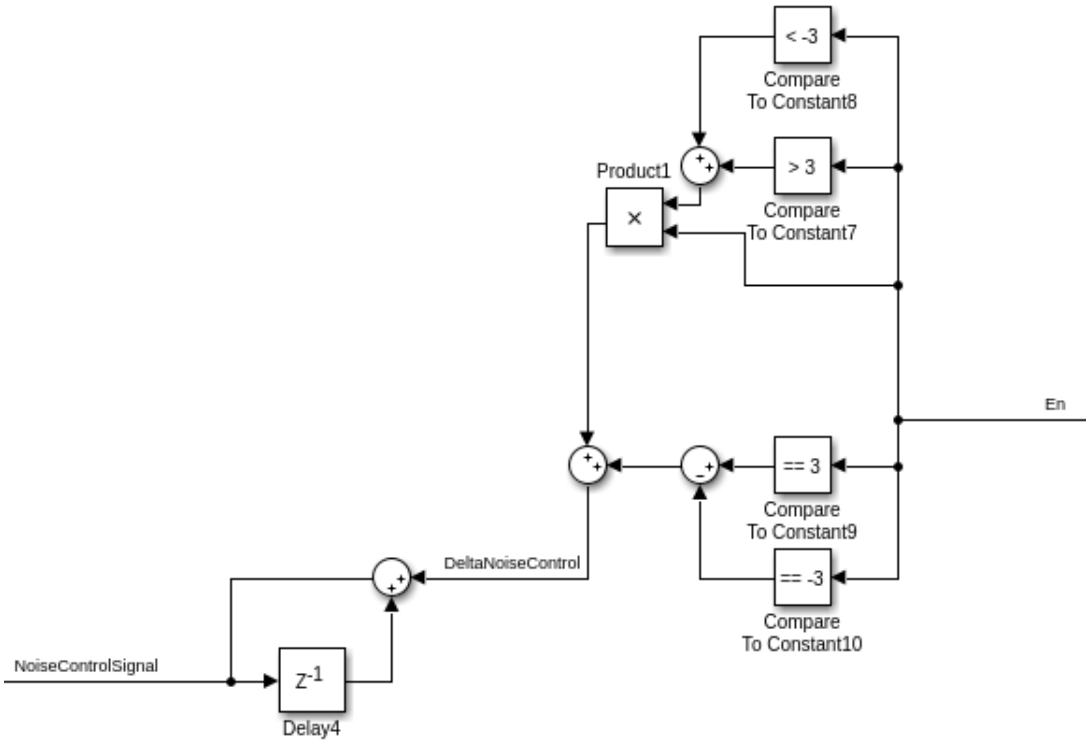


Figure B.0.3: The Noise Adjustment Algorithm with up-down counter. The input of this block is the output amplitude error, E_n. If $|E_n| > 3$, than $\Delta NC = E_n$. For $|E_n| = 3$, the DeltaNoiseControl signal is either 1 or -1. If $|E_n| > 3$, the DeltaNoiseControl signal is 0.

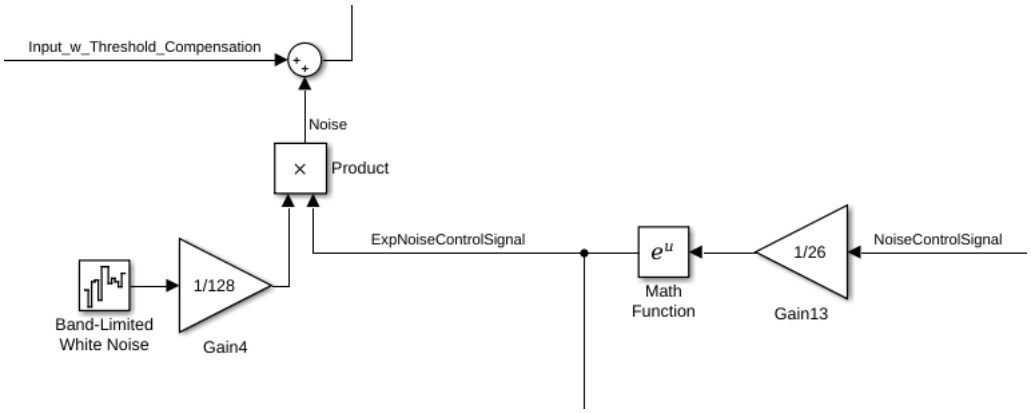


Figure B.0.4: The implementation of the exponential function and the noise multiplier which adjusts the noise level. An exponential function is applied to the NoiseControl Signal. The noise signal is multiplied byt this signal, ExpNoiseControlSignal. Finally, the output of this product block is added to the signal input.

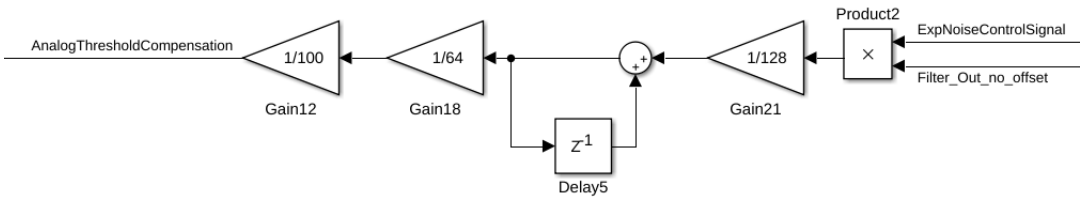


Figure B.0.5: The Threshold Compensation Block. The inputs are the filter output, mapped to $\langle -64, 63 \rangle$. This signal is multiplied with the ExpNoiseControlSignal and divided by 128. The input is thereby normalized to the noise level. The output of this gain block is integrated by the up-down counter implemented by the delay block in positive feedback. The output of the up-down counter is divided by 64. Hereby is the bit length reduced by 6 bits. The DAC is implemented by the 1/100 gain block. This block translates the the digital bit values to an analog signal.

C

System Level Simulation Results

This appendix comprises simulation results of the system level as proposed in Chapter 5.

C.1 Noise Control

Signal Amplitude	$f = 5$ Hz	$f = 50$ Hz
0.1	27.32 dB	27.16 dB
0.2	26.39 dB	26.49 dB
0.5	27.19 dB	27.24 dB
0.7	27.23 dB	27.32 dB
1	27.20 dB	27.26 dB

(a) SNDR

Signal Amplitude	$f = 5$ Hz	$f = 50$ Hz
0.1	28.75 dB	28.24 dB
0.2	27.56 dB	27.66 dB
0.5	28.85 dB	28.53 dB
0.7	28.86 dB	28.50 dB
1	28.84 dB	28.54 dB

(b) SNR

Signal Amplitude	$f = 5$ Hz	$f = 50$ Hz
0.1	-32.90 dB	-33.72 dB
0.2	-32.64 dB	-32.77 dB
0.5	-32.18 dB	-33.12 dB
0.7	-32.27 dB	-33.58 dB
1	-32.21 dB	-33.16 dB

(c) THD

Table C.1: Simulation results of the system with dynamic range control loop, without offset control loop.

C.2 Offset Compensation

Offset	f_{in} = 0.5 Hz	f_{in} = 5 Hz	f = 50 Hz
0	25.43 dB	27.05 dB	27.24 dB
0.1	25.43 dB	27.05 dB	27.24 dB
0.2	25.43 dB	27.05 dB	27.24 dB
0.3	25.43 dB	27.05 dB	27.24 dB
0.4	25.43 dB	27.05 dB	27.24 dB
0.5	25.43 dB	27.05 dB	27.24 dB

(a) SNDR

Offset	f_{in} = 0.5 Hz	f_{in} = 5 Hz	f = 50 Hz
0	25.65 dB	28.72 dB	28.61 dB
0.1	25.65 dB	28.72 dB	28.61 dB
0.2	25.65 dB	28.72 dB	28.61 dB
0.3	25.65 dB	28.72 dB	28.61 dB
0.4	25.65 dB	28.72 dB	28.61 dB
0.5	25.65 dB	28.72 dB	28.61 dB

(b) SNR

Offset	f_{in} = 0.5 Hz	f_{in} = 5 Hz	f = 50 Hz
0	-38.47 dB	-32.02 dB	-32.92 dB
0.1	-38.47 dB	-32.02 dB	-32.92 dB
0.2	-38.47 dB	-32.02 dB	-32.92 dB
0.3	-38.47 dB	-32.02 dB	-32.92 dB
0.4	-38.47 dB	-32.02 dB	-32.92 dB
0.5	-38.47 dB	-32.02 dB	-32.92 dB

(c) THD

Table C.2: Simulation results of the system with offset control loop, without dynamic range control loop. The input signal is a sine wave with an amplitude of 1. The offset is varied from 0 to 0.5. The tables show that the performance of the system is independent on the applied offset.

C.3 Full System

Signal Amplitude	Offset = 0	Offset = 0.1	Offset = 0.2	Offset = 0.5
0.1	22.91 dB	22.91 dB	22.80 dB	22.79 dB
0.2	25.41 dB	25.41 dB	25.42 dB	25.58 dB
0.5	26.61 dB	26.61 dB	26.61 dB	26.47 dB
1	26.84 dB	26.84 dB	26.84 dB	26.74 dB

(a) $f = 0.5$ Hz

Signal Amplitude	Offset = 0	Offset = 0.1	Offset = 0.2	Offset = 0.5
0.1	25.02 dB	25.54 dB	25.56 dB	25.54
0.2	26.43 dB	26.44 dB	26.44 dB	26.34
0.5	27.08 dB	27.08 dB	27.09 dB	27.09
1	27.06 dB	27.06 dB	27.06 dB	27.06

(b) $f_{\text{in}} = 5$ Hz

Signal Amplitude	Offset = 0	Offset = 0.1	Offset = 0.2	Offset = 0.5
0.1	25.14 dB	25.20 dB	25.20 dB	25.20 dB
0.2	26.28 dB	26.06 dB	26.06 dB	26.26 dB
0.5	26.93 dB	26.94 dB	26.94 dB	26.94 dB
1	27.34 dB	27.34 dB	27.34 dB	27.34 dB

(c) $f = 50$ Hz

Table C.3: Signal-to-noise-and-distortion ratios of the total system

Signal Amplitude	Offset = 0	Offset = 0.1	Offset = 0.2	Offset = 0.5
0.1	23.74 dB	23.75 dB	23.57 dB	23.56 dB
0.2	26.52 dB	26.53 dB	26.54 dB	26.99 dB
0.5	28.23 dB	28.23 dB	28.23 dB	27.77 dB
1	28.81 dB	28.81 dB	28.81 dB	29.11 dB

(a) $f_{\text{in}} = 0.5$ Hz

Signal Amplitude	Offset = 0	Offset = 0.1	Offset = 0.2	Offset = 0.5
0.1	26.40 dB	27.30 dB	27.32 dB	27.30 dB
0.2	27.66 dB	27.49 dB	27.69 dB	28.02 dB
0.5	28.50 dB	28.50 dB	28.51 dB	28.51 dB
1	29.27 dB	29.27 dB	29.27 dB	29.27 dB

(b) $f_{\text{in}} = 5$ Hz

Signal Amplitude	Offset = 0	Offset = 0.1	Offset = 0.2	Offset = 0.5
0.1	26.31 dB	26.61 dB	26.61 dB	26.42 dB
0.2	27.85 dB	27.46 dB	27.46 dB	27.83 dB
0.5	28.17 dB	28.18 dB	28.18 dB	28.18 dB
1	28.94 dB	28.94 dB	28.94 dB	28.95 dB

(c) $f = 50$ Hz

Table C.4: Signal-to-noise ratios of the total system

Signal Amplitude	Offset = 0	Offset = 0.1	Offset = 0.2	Offset = 0.5
0.1	-30.48 dB	-30.47 dB	-30.68 dB	-30.71 dB
0.2	-31.88 dB	-31.85 dB	-31.86 dB	-31.18 dB
0.5	-31.68 dB	-31.68 dB	-31.68 dB	-32.35 dB
1	-31.23 dB	-31.23 dB	-31.23 dB	-30.50 dB

(a) $f_{\text{in}} = 0.5$ Hz

Signal Amplitude	Offset = 0	Offset = 0.1	Offset = 0.2	Offset = 0.5
0.1	-30.68 dB	-30.32 dB	-30.32 dB	-30.33 dB
0.2	-32.49 dB	-32.47 dB	-32.47 dB	-31.27 dB
0.5	-32.63 dB	-32.63 dB	-32.64 dB	-32.64 dB
1	-31.05 dB	-31.05 dB	-31.05 dB	-31.05 dB

(b) $f_{\text{in}} = 5$ Hz

Signal Amplitude	Offset = 0	Offset = 0.1	Offset = 0.2	Offset = 0.5
0.1	-31.39 dB	-30.77 dB	-30.77 dB	-31.32 dB
0.2	-31.46 dB	-31.65 dB	-31.66 dB	-31.43 dB
0.5	-32.98 dB	-32.98 dB	-32.98 dB	-32.99 dB
1	-32.45 dB	-32.44 dB	-32.44 dB	-32.44 dB

(c) $f = 50$ Hz

Table C.5: Total harmonic distortion of the total system

D

Digital Low-pass Filter

The filter consists of 4 basic building blocks. First of all, a clock divider is implemented to deliver the right sampling frequencies to all stages. Secondly a delay block of 1 clock sample is implemented. The summation block comprises not only the summation of two values, but also performs the downsampling, and the division in the last 5 stages.

D.1 Clock Divider

The clock divider is implemented by means of D-Flipflops, as shown in Figure D.5.1a. The output $\overline{Q}$ is connected to the data input D. When the flipflop is triggered, the output $\overline{Q}$ is transfered to the output Q, flipping the outputs. The outputs flip once every clock cycle, resulting in new clock signal with $\frac{f_s}{2}$. In Figure D.5.1b, the output waves of the clock divider are shown.

The divider is cascaded 12 times, to create $Q[1]$ to $Q[12]$ and $\overline{Q}[1]$ to $\overline{Q}[12]$. The signals $Q[0]$ and $\overline{Q}[0]$ are the input clock and inverse input clock respectively.

D.2 Delay Block

The delay block is implemented by a edge triggered D-flipflop, on the positive clock edge. The clock signal used in stage i is $Q[i-1]$, and multibit signal delays are implemented by multiple, parallel flipflops.

D.3 Summation and Downsampling

A multi-bit adder is implemented using a half-adder, and full-adders, as shown in Figure D.5.2. In the final five stages, the LSB S_0 is not calculated. The halfadder is in these cases implemented by just an AND-port.

The output of the adder is latched to implement the downsampling. The flipflops are triggered by the negative edge of the clock. The sample time is twice the sample time of the delay in the same stage: in stage i, the flipflops are triggered by $\overline{Q}[i-1]$.

.

D.4 Filter Delay

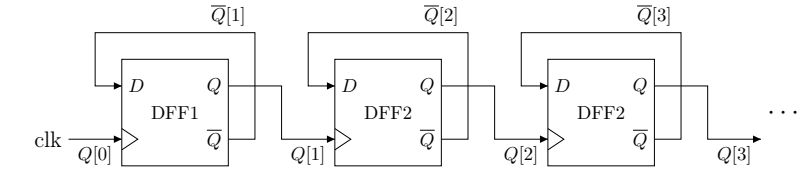
The filter topology introduces a delay of maximum one cycle of $Q[12]$, as shown in Chapter 5. The implementation using negative edge triggered flipflops to perform the downsampling introduces another delay of half a clock cycle. By triggering the downsamplers on the negative edge instead of the positive edge, the added delay is reduced from one cycle to a half cycle.

The total delay can be reduced further by a more advanced sampling structure of the downsamplers. The sampling time of the clocks is orders of magnitude larger than the co mputation time necessary to perform the summation. One could, for example, think of a timing schedule in which each successive downsampler is triggered with only a small delay compared to the previous downsampler, reducing the total delay from the input to the output.

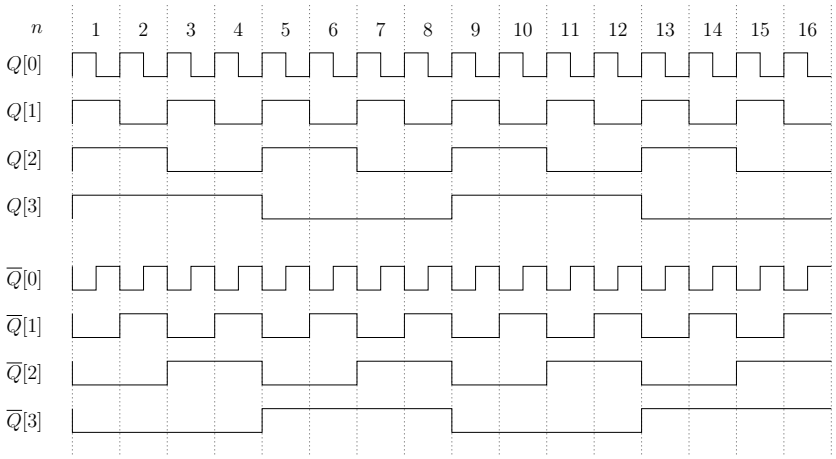
In this design however, the delay introduced by the filter is not restricted. For this reason, the negative edge triggered downsamplers are implemented.

D.5 Verilog

The filter is implemented in Verilog, to simulate the timing of the multi-rate design. The code is provided in the Algorithms 1, 2, 3, 4, and 5.



(a) Clock Divider



(b) Clock Divider Timing Diagram

Figure D.5.1: First three stages of the clock divider, and the corresponding output waves. The implemented divider consists of 12 stages.

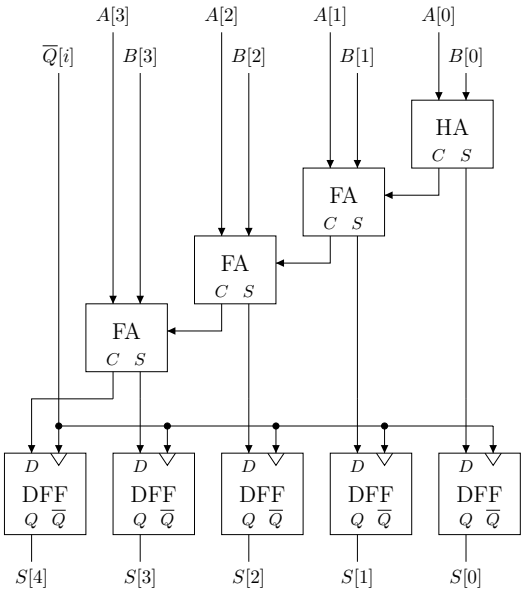


Figure D.5.2: Summation block example for two 4 bits input.

Algorithm 1 Verilog Code for the Clock Divider

```
module clk_div(
output div_clk,div_clk_inv,
input clk,reset );

DFF DFF_clk_div(
        .Q      (div_clk),
        .nQ     (div_clk_inv),
        .data   (div_clk_inv),
        .clk    (clk),
        .reset  (reset) );
endmodule
```

```
module clk_div_array2(
output [12:0] clk_array,
output [12:0] clk_array_inv,
input clk,reset
);

assign clk_array[0] = clk;
not not1(clk_array_inv[0],clk);

genvar index;
generate for (index = 0; index < 12; index=index+1)
begin: gen_clk_array
        clk_div clk_div1(
                .div_clk                (clk_array[index+1]),
                .div_clk_inv    (clk_array_inv[index+1]),
                .clk                    (clk_array[index]),
                .reset            (reset)
        );
end
endgenerate
endmodule
```

Algorithm 2 Verilog Code of the Latch

```
module DFF(
output Q,nQ,
input data,clk,reset
);

wire nandA1_wire, nandA2_wire, QA_wire, nQA_wire,nandB1_wire,nandB2_wire;
wire nQ_wire;
wire clk_inv,clk2;
wire reset_inv;

not not1(clk_inv,clk);
not not2(clk2,clk_inv);
not not3(reset_inv,reset);

and and1(Q,Q_wire,reset_inv);
or or1(nQ,nQ_wire,reset);

nand nandA1(nandA1_wire,data,clk_inv);
nand nandA2(nandA2_wire,nandA1_wire,clk_inv);
nand nandA3(QA_wire,nQA_wire,nandA1_wire);
nand nandA4(nQA_wire,QA_wire,nandA2_wire);
nand nandB1(nandB1_wire,QA_wire,clk2);
nand nandB2(nandB2_wire,nandB1_wire,clk2);
nand nandB3(Q_wire,nandB1_wire,nQ);
nand nandB4(nQ_wire,Q,nandB2_wire);

endmodule
```

Algorithm 3 Verilog Code of the Half-Adder and Full-Adder

```
module HA(
output sum,carry,
input a,b );

xor xor1(sum,a,b);
and and1(carry,a,b);

endmodule
```

```
module FA(
output sum,carry,
input a,b,c
);

wire s1,s2,s3;

xor xor1(s1,a,b);
and and1(s2,c,s1);
and and2(s3,a,b);
xor xor2(sum,s1,c);
or or1(carry,s2,s3);

endmodule
```

Algorithm 4 Verilog Code of the Summation

```
module sum2_7 #(parameter nr_bits=2)(
output [nr_bits:0] out,
input  [nr_bits-1:0] a,b,
input  clk, reset
);

wire [nr_bits-1:0] sum_wire,carry_wire;
wire [nr_bits:0] out_DFF1;
wire clk_wire,clk_wire_inv;

not not1(clk_wire_inv,clk2);
not not2(clk_wire,clk_wire_inv);

HA HA1(
        .sum(sum_wire[0]),
        .carry(carry_wire[0]),
        .a(a[0]),
        .b(b[0])
);
DFF DFF_1(
        .Q(out[0]),
        .nQ(),
        .data(sum_wire[0]),
        .clk(clk),
        .reset(reset)
);

genvar index;
generate
for (index = 1; index < nr_bits; index=index+1)
begin: gen_fulladders_and_out_latches
        FA FA1(
                .sum(sum_wire[index]),
                .carry(carry_wire[index]),
                .a(a[index]),
                .b(b[index]),
                .c(carry_wire[index-1])
        );

        DFF DFF_2(
                .Q(out[index]),
                .nQ(),
                .data(sum_wire[index]),
                .clk(clk),
                .reset(reset)
        );
end
endgenerate

DFF DFF_C1(
        .Q(out[nr_bits]),
        .nQ(),
        .data(carry_wire[nr_bits-1]),
        .clk(clk),
        .reset(reset)
);
endmodule
```

Algorithm 5 Verilog Code of the Limiter

```
module limiter(
output [4:0]out,
input [5:0] in
);
genvar index;
generate
for (index = 0; index < 5; index=index+1)
begin: gen_limiter
        or or1(out[index],in[index],in[5]);
end
endgenerate
endmodule
```

E

Noise Chopping

Chopping can be applied to the noise source to reduce the negative effect of the pink noise originating from the amplifier. In this case, chopping has to be performed only once, since no signal is applied at the input of the amplifier. The effect of chopping the noise is simulated using Matlab, multiplying every other noise sample with -1. The input noise is a combination of white and pink noise. The simulation results are shown in Table E.1. The results show that the negative effect of the added pink noise is, reduced significantly.

It should be noted that in $1/f$-noise simulations, the frequency resolution has a significant influence on the simulation results. The results shown in Table E.1, are obtained using a resolution of 2 Hz, a sampling frequency of 2 MHz, and a signal bandwidth of 200 Hz. Increasing this resolution results in a larger portion of the $1/f$-noise at lower frequencies.

Although the chopper removes the $1/f$-noise from the passband, the performance of the system is reduced for higher pink noise levels. This is caused by fluctuations of the noise power in the time domain. The chopper removes the peak in the frequency domain but does not change the instantaneous power within the system. The instantaneous noise power shows low-frequency fluctuations over time due to the pink noise. These fluctuations appear in the output of the system since the output is determined by the input value and the noise power. The low-frequency fluctuations in the noise power alter the transfer function, resulting in a decreased performance.

Even when noise chopping is applied, the total contribution of $1/f$-noise power should be kept below 10% of the total noise.

pink noise power	SNDR no chopper	SNDR with chopper
0	26.9 dB	26.9 dB
0.01	21.7 dB	26.9 dB
0.02	20.2 dB	26.9 dB
0.05	17.9 dB	26.7 dB
0.1	16.3 dB	26.5 dB
0.2	14.8 dB	25.2 dB
0.5	12.9 dB	22.2 dB
1	11.7 dB	19.9 dB

Table E.1: Simulation results of the system when pink noise is added to the system. The maximum SNDR is shown for different amount of pink noise added to the white noise source of power 1. The simulations are performed using the set-up introduced in Chapter 4.

F

Conference Paper International Symposium on Circuits and Systems

Stochastic Resonance Mixed-Signal Processing: Analog-to-Digital Conversion and Signal Processing Employing Noise

Insani Abdi Bangsa[*,‡,§,†], Dieuwert P. N. Mul[*,§], and Wouter A. Serdijn[*]
iabdi.bangsa@ft.unsika.ac.id, dpnmul@gmail.com, and w.a.serdijn@tudelft.nl
[*]Bioelectronics Section, Department of Microelectronics
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology, the Netherlands
[‡]Faculty of Engineering, State University of Singaperbangsa Karawang, Indonesia

***Abstract*—Stochastic resonance (SR) is a phenomenon in which noise can be employed to increase the performance of a system. It can e.g. be used to improve the performance of comparator-based circuits. This paper presents the analytical derivation of input-output relation, harmonic distortion, and noise behaviour of a 1-bit ADC using SR. Furthermore, the design of a new signal multiplier based on SR-ADCs is presented. The predicted behaviours are demonstrated by means of simulations. The work presented in this paper shows the potential for analog to digital conversion and integrated signal processing fully based on stochastic resonance.**

***Index Terms*—stochastic resonance, analog-to-digital converter, multiplier, 1-bit processing.**

I. Introduction

Noise has been mostly considered as something undesirable in the world of signal processing. When mixed with a signal, it is thought to obstruct the extraction of information contained in the signal. However, the occurrence of a phenomenon called stochastic resonance shows that the presence of noise can improve the performance of a system.

Stochastic resonance (SR) is a phenomenon in which the performance of a nonlinear system is better than it is without noise [1]. By adding noise to, e.g., a comparator-based circuit, a system based on SR can be built [2], [3]. The addition of noise will increase the frequency of changes of the output states, which, when averaged, will lead to an input-output relation that is more linear than that of a noiseless system. An SR system has a performance peak for a specific noise level.

This paper focuses on using SR to design a 1-bit analog-to-digital converter (ADC) and to integrate mathematical operations with the ADC. First, the SR-ADC (Fig. 1) is introduced in Section II and analytic derivations of the behaviour of the proposed system are presented in Section III. This derivation allows designers to predict the performance of the SR-ADC, and thereby enables them to determine the optimal values of the design parameters. Furthermore, 1-bit stochastic resonance signal processing operators are discussed and the design of a SR-multiplier is presented in Section IV. Finally, the conclusions are presented in Section V.

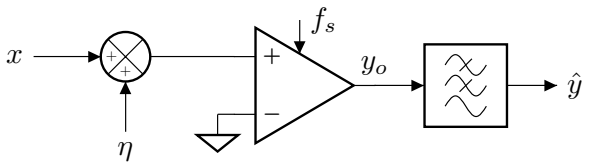


Fig. 1. Stochastic resonance system.

II. Stochastic Resonance System

A stochastic resonance system has a non-zero optimal noise value. To achieve this property, the system needs to be nonlinear, and should have two (or more) stable states [4]. In sub-threshold SR, the signal without noise cannot reach the threshold to switch from one state to another [1]. In this case, the noise is necessary to reconstruct any information from the signal. A second class of SR systems is supra-threshold SR. The signal without noise can reach the threshold to switch from one stable state to another. However, adding noise can cause the system to switch from one state to another more often, which increases the performance [5], [6]. For higher noise levels, the system behaves (close to) linearly.

Fig. 1 shows the proposed system, where x, η, y_o, and $\hat{y}$ are the input signal, input noise, 1-bit output, and averaged output, respectively. The output of the quantiser is a pulse density modulated (PDM) signal. By averaging the PDM signal using a low-pass filter, the output is transferred back to the amplitude domain. The quantiser amplifies the signal, since it can be driven by a weak noisy input signal. Besides performing the amplification, the signal is directly transformed to the binary domain, and can thus be defined as a stochastic resonance analog-to-digital converter (SR-ADC). The mechanism employed in this system shows large similarities to dithering [3], [7].

[§]I. A. Bangsa and D. P. N. Mul contributed equally to this work.
[†]I. A. Bangsa is funded by The Indonesia Endowment Fund for Education (LPDP RI).

978-1-5386-4881-0/18/$31.00 ©2018 IEEE

III. Analytical Derivation

The behaviour of stochastic resonance systems can be found in literature [3], [7]. However, although a similar idea has been mentioned in [8], to the knowledge of the authors, an analytical derivation targeting a designers guide for a SR-ADC is not reported on yet. This section presents a method to predict the performance of the considered SR-ADC. To validate the method, an example is given using white Gaussian noise and a sine wave input. However, the method can be used for any input signal, and any type of white noise.

A. Input-Output Relation

The output of the quantiser is a stochastic signal. By averaging this signal, the average of multiple samples is calculated. As a result of the averaging, the expected transfer is defined by the probability of each of the two states, either Q_+ or Q_-. The expected output $\hat{y}$ is described by (1), where x is the input signal value, η is the noise value. The quantiser outputs are normalized to $Q_+ = 1$ and $Q_- = -1$.

$$\begin{aligned}\hat{y}(x,\eta) &= \mathbf{E}[y \,|\, x] \\ &= P\big((x+\eta) > 0 \,|\, x\big) - P\big((x+\eta) < 0 \,|\, x\big) \\ &= P(\eta > -x \,|\, x) - P(\eta < -x \,|\, x) \\ &= 1 - 2 \cdot P(\eta < -x \,|\, x)\end{aligned} \tag{1}$$

The probability $P(\eta < -x \,|\, x)$, and thus the expected transfer, can be calculated using the cumulative distribution function of the noise. The noise is assumed to be white Gaussian noise, which gives:

$$\begin{aligned}P(\eta < -x \,|\, x) &= \int_{-\infty}^{-x} \frac{1}{\sigma\sqrt{2\pi}} \cdot \exp\left(\frac{-\eta^2}{2\sigma^2}\right) \mathrm{d}\eta \\ &= \frac{1}{2} + \frac{1}{2}\mathrm{erf}\left(\frac{-x}{\sqrt{2}\sigma}\right).\end{aligned} \tag{2}$$

Thus the expected output for a given input value x is

$$\hat{y}(x,\sigma) = \mathrm{erf}\left(\frac{x}{\sqrt{2}\sigma}\right). \tag{3}$$

The expected transfer of the system for different noise levels is shown in Fig. 2. The expected output is compared with simulations of the system. Averaging is performed over $10,000$ samples.

B. Gain and Harmonic Distortion

The performance of the SR-ADC is a trade-off between noise and harmonic distortion. Thus, an accurate quantisation of the harmonic distortion is crucial in the performance analysis. The total harmonic distortion (THD) is derived in the time domain. The expected output $\hat{y}(x,\sigma,t)$ is split into a linear component $\hat{H}(x,\sigma) \cdot x(t)$ and harmonic distortion $HD(t)$, which leads to

$$\hat{y}(x,\sigma,t) = \hat{H}(x,\sigma) \cdot x(t) + HD(t). \tag{4}$$

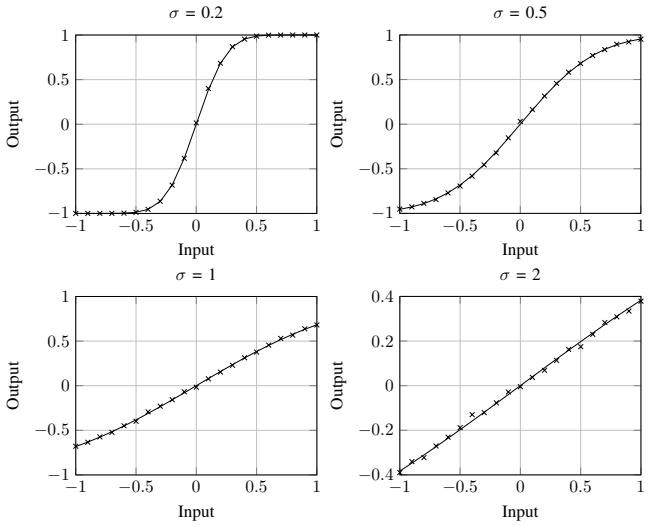


Fig. 2. Expected transfer for different noise levels. The simulation results are marked with 'x'.

The linear gain $\hat{H}(x,\sigma)$ is derived by performing a least square error fit. The gain $\hat{H}(x,\sigma)$ is scaled such that it produces the least mean squared error, and thus the power of the harmonic distortion is minimized. Since the fitted signal only affects the fundamental frequency, minimizing the error power ensures that the error only contains harmonics.

C. Noise

The noise power at the output of the quantiser can be calculated directly from the expected output ($\hat{y}(x,\sigma,t)$). Since the output of the quantiser is either 1 or -1, the power at the output of the quantiser is 1. The noise power is thus at

$$\begin{aligned}P_{\eta,Q} &= 1 - P_{\hat{y}(x,\sigma)} \\ &= 1 - \frac{1}{T}\int_0^T \hat{y}(x,\sigma)^2 \,\mathrm{d}t\,,\end{aligned} \tag{5}$$

where $x(t)$ is the input signal and T is the signal period.

Since every sample of white noise is independent of each other, the noise at the output of the quantiser is also independent, and thus white. The noise at the output of the system is averaged by a filter, which leads to an output noise power formulated as:

$$P_{\eta,out} = \frac{f_{ENBW}}{\frac{1}{2}f_s} P_{\eta,Q}, \tag{6}$$

where f_{ENBW} is the equivalent noise bandwidth of the filter and f_s is the sampling frequency.

D. Signal-to-Noise-and-Distortion Ratio

The performance of the system can be defined using the signal-to-noise-and-distortion ratio (SNDR). This measure gives the ratio between the output signal power and the combined harmonic distortion and noise power. The SNDR is defined by

$$\mathrm{SNDR} = \frac{P_{out,signal}}{P_{HD} + P_{\eta,out}}. \tag{7}$$

978-1-5386-4881-0/18/$31.00 ©2018 IEEE

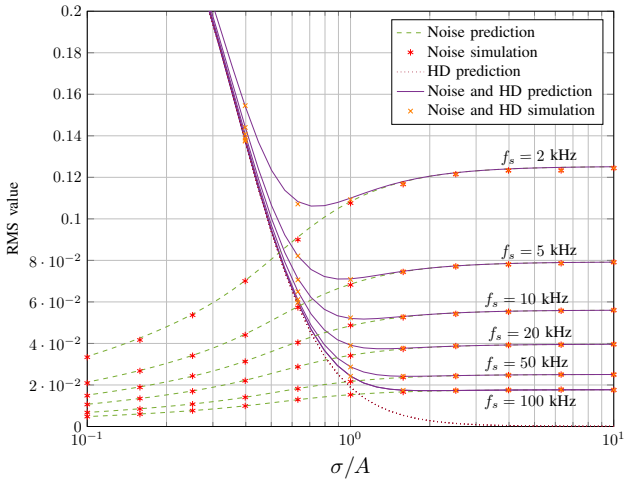


Fig. 3. RMS output error of the SR-ADC for a sine-wave input with a frequency of 1 Hz and $f_s = [2, 5, 10, 20, 50, 100]$ kHz. The output is filtered by a first-order Butterworth LPF with a cut-off frequency of 10 Hz. The RMS input noise σ is set to 1, while the input signal amplitude A is shifted from 0.1 to 10.

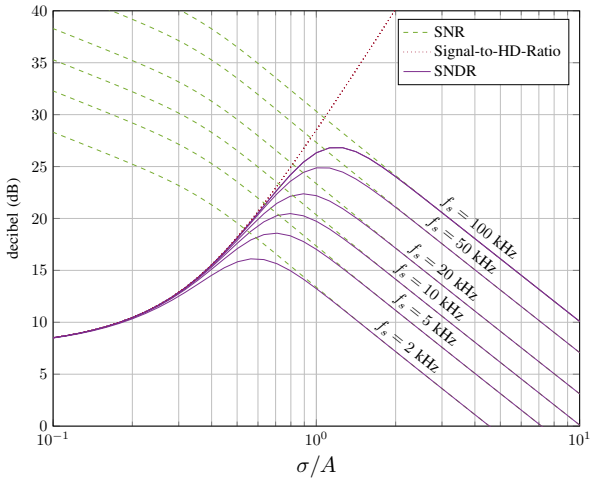


Fig. 4. Signal-to-noise-and-distortion ratio of the SR-ADC for a sine-wave input with a frequency of 1 Hz and $f_s = [2, 5, 10, 20, 50, 100]$ kHz. The output is filtered by a first-order Butterworth LPF with a cut-off frequency of 10 Hz. The RMS input noise σ is set to 1, while the input signal amplitude A is shifted from 1 to 10.

The output signal and harmonic distortion power are given by

$$P_{out,signal} = \hat{H}(x,\sigma)^2 P_{in,signal} \tag{8}$$

and

$$P_{HD} = \frac{1}{T}\int_0^T \left(\hat{y}(x,\sigma,t) - \hat{H}(x,\sigma)\cdot x(t)\right)^2 \mathrm{d}t\,, \tag{9}$$

respectively.

The results of the proposed method are verified for the case of a sine-wave input. In this example, the signal frequency is set to 1 Hz, the cut-off frequency is 10 Hz and the filter is a first-order Butterworth LPF. The input noise is white Gaussian noise with $\sigma = 1$, while the input signal amplitude (A) is shifted from 0.1 to 10. Fig. 3 shows the output error for $f_s = [0.2, 0.5, 1, 2, 5, 10]$ kHz. Alongside the analytical results, simulation results of the system are shown to validate the method. The final SNDR results for the same sample ratios are shown in Fig. 4. The figure clearly shows the performance peak. A higher sampling ratio gives a higher SNDR. For low noise levels, the distortion limits the behaviour, while for high noise values, the noise becomes the dominant source of error. It should be noted that these results only hold for a single sine-wave input. The waveform of the signal affects the noise as well as the harmonic distortion behaviour.

IV. SR-Based Mathematical Operations

Not only limited to analog-to-digital conversion and amplification, SR can also be used to do mathematical operations on signals. There are four fundamental mathematical operations: addition, subtraction, multiplication, and division. An SR-adder can be built by implementing a half-adder after the SR-ADCs. Consequently, a subtractor can be implemented by using a half-adder while inverting one of the SR-ADC outputs.

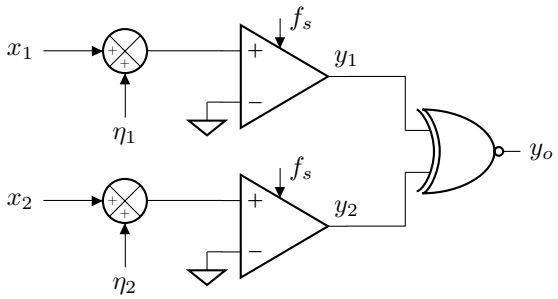


Fig. 5. Proposed SR-multiplier using an XNOR gate in combination with two SR-ADCs.

While a division might be difficult to implement due to the large value of $1/x$ when x is close to zero, a multiplication is feasible. In the next part, we demonstrate the design of a signal multiplier using two SR-ADCs.

A. SR-Multiplier

The output of the SR-ADCs is either $+V$ or $-V$ which corresponds to a logic "1" or "0", respectively. Therefore, the output of the proposed SR-multiplier should be "1", i.e. positive, when both outputs of the SR-ADCs are the same, and "0", i.e. negative, when they are different. Therefore, an SR-multiplier can be implemented with a combination of two SR-ADCs and an XNOR gate. The block diagram of the SR-multiplier is shown in Fig. 5. x_1 and x_2 denote the input signals while η_1 and η_2 are the input noise signals corresponding to each SR-ADC. η_1 and η_2 are independent of each other.

Using the probability of the SR-ADC output derived in

978-1-5386-4881-0/18/$31.00 ©2018 IEEE

Section III, the probability of $y_o = 0$ is

$$\begin{aligned} P(y_o = 0 \,|\, x_1, x_2) &= P(y_1 = 1 \,|\, x_1) P(y_2 = 0 \,|\, x_2) \\ &\quad + P(y_1 = 0 \,|\, x_1) P(y_2 = 1 \,|\, x_2) \\ &= -2P(\eta_1 < -x_1 \,|\, x_1) P(\eta_2 < -x_2 \,|\, x_2) \\ &\quad + P(\eta_1 < -x_1 \,|\, x_1) + P(\eta_2 < -x_2 \,|\, x_2). \end{aligned} \tag{10}$$

From (1) and (10), the expected output $\hat{y}(x_1, x_2, \sigma_1, \sigma_2)$ can be calculated by

$$\begin{aligned} \hat{y}\,(x_1, x_2, \sigma_1, \sigma_2) &= V\big(1 - 2P(y_o = 0 \,|\, x_1, x_2)\big) \\ &= V\Big(1 + 4P(\eta_1 < -x_1 \,|\, x_1) P(\eta_2 < -x_2 \,|\, x_2) \\ &\quad - 2\big(P(\eta_1 < -x_1 \,|\, x_1) + P(\eta_2 < -x_2 \,|\, x_2)\big)\Big). \end{aligned} \tag{11}$$

By substituting the $P(\eta_n < -x_n \,|\, x_n)$ with $\hat{y}_n$ using (1), as long as η_1 and η_2 are independent of each other, (11) can be rewritten as

$$\hat{y}\,(x_1, x_2, \sigma_1, \sigma_2) = \frac{\hat{y}_1(x_1, \sigma_1) \cdot \hat{y}_2(x_2, \sigma_2)}{V}, \tag{12}$$

which shows that the proposed operator is a multiplier. For an SR-multiplier using white Gaussian noise sources, (12) becomes

$$\hat{y}(x_1, x_2, \sigma_1, \sigma_2) = V \mathrm{erf}\left(\frac{x_1}{\sqrt{2}\sigma_1}\right) \mathrm{erf}\left(\frac{x_2}{\sqrt{2}\sigma_2}\right). \tag{13}$$

B. Signal-to-Noise-and-Distortion Ratio

To determine the gain, distortion, and output noise power, the methods presented in Section III can be used by substituting (13) for $\hat{y}$ in (5). The gains $\hat{G}$ in case of two sinusoidal inputs $\sin(2\pi t)$ and $\sin(2\pi t + \phi)$ are shown in Fig. 6. The SR-multiplier is subjected to independent white Gaussian noise sources with equal power, denoted by σ^2. V is set to 1 and $\phi = \left[-\frac{\pi}{2}, -\frac{\pi}{2} \pm \frac{\pi}{6}, -\frac{\pi}{2} \pm \frac{\pi}{2}\right]$, such that the correlation coefficient of the input signals, $\rho = [0, \pm 0.5, \pm 1]$. The signals are sampled with $f_s = 100$ kHz and the outputs are filtered by a second-order Butterworth LPF with a cut-off frequency of 10 Hz. The case with zero correlation ($\phi = -\pi/2$) has the biggest gain due to its output having zero mean, which leads to a lower distortion power. As the noise RMS value σ goes up, the gains go down approaching $2V/\left(\pi\sigma^2\right)$, shown by the solid line.

The simulated and predicted SNDRs for the same cases are presented in Fig. 7. SNDR peaks can be observed for every case where noise is present due to the stochastic resonance.

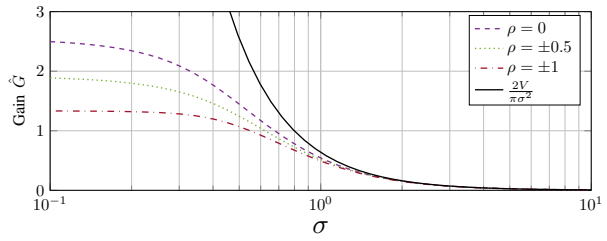


Fig. 6. Gain of the SR-multiplier for $x_1 = \sin(2\pi t)$ and $x_2 = \sin(2\pi t + \phi)$. V is set to 1 and $\phi = \left[-\frac{\pi}{2}, -\frac{\pi}{2} \pm \frac{\pi}{6}, -\frac{\pi}{2} \pm \frac{\pi}{2}\right]$ such that $\rho = [0, \pm 0.5, \pm 1]$. η_1 and η_2 are white, independent, and have equal RMS values σ. As σ goes up, the gain approaches $2V/\left(\pi\sigma^2\right)$.

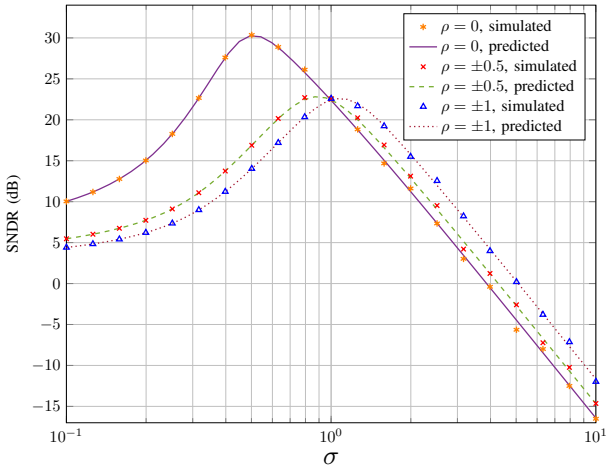


Fig. 7. Signal-to-noise-and-distortion ratio of the SR-multiplier for $x_1 = \sin(2\pi t)$ and $x_2 = \sin(2\pi t + \phi)$, sampled with $f_s = 100$ kHz and filtered by a second-order Butterworth LPF with a cut-off frequency of 10 Hz. V is set to 1 and $\phi = \left[-\frac{\pi}{2}, -\frac{\pi}{2} \pm \frac{\pi}{6}, -\frac{\pi}{2} \pm \frac{\pi}{2}\right]$ such that $\rho = [0, \pm 0.5, \pm 1]$. η_1 and η_2 are white, independent, and have equal RMS values σ. The marks indicate the simulated results and the lines are the predicted results based on the formula. SNDR peaks can be found at different σ for different cases.

V. Conclusions

In this paper, a novel signal conversion and signal processing technique has been discussed. As two examples, an SR-ADC and an SR-multiplier have been introduced.

In Section III, an analytical method to determine the performance of a fully stochastic resonance ADC is proposed. Based on the expected output of the quantiser, the signal power, distortion power and noise power are derived. Since the spectrum of white noise is preserved by the quantisation step, the filtered output noise can be calculated. The method can be used for all (periodic) input signals combined with Gaussian, as well as non-Gaussian white noise. The method is developed to allow structured analysis of SR-ADC systems for design purposes and can also be applied to SR-ADCs with integrated signal processing.

In Section IV, we have shown that SR can be used to do mathematical operations. Using the stochastic properties of the 1-bit signal acquired by the SR-ADC, a compact and efficient signal multiplier using an XNOR logic gate can be designed. It is proven that the proposed operator works as a multiplier as long as the noise sources are independent of each other.

The observations presented in this paper show a potential for analog to digital conversion and integrated signal processing fully based on stochastic resonance. For low-frequency signal reconstruction in high-noise environments, the SR-ADC can be a simple and efficient alternative for conventional analog front-ends and signal processing.

978-1-5386-4881-0/18/$31.00 ©2018 IEEE

REFERENCES

[1] M. D. McDonnell and D. Abbott, "What is stochastic resonance? definitions, misconceptions, debates, and its relevance to biology," *PLoS Computational Biology*, vol. 5, pp. 1–9, May 2009.

[2] S. Fauve and F. Heslot, "Stochastic resonance in a bistable system," *Physics Letters*, 1983.

[3] H. A. Hjortland, *Sampled and continuous-time 1-bit signal processing in CMOS for wireless sensor networks*. PhD thesis, University of Oslo, 2016.

[4] G. P. Harmer, B. R. Davis, and D. Abbott, "A review of stochastic resonance: circuits and measurement," *IEEE Transactions on Instrumentation and Measurement*, vol. 51, pp. 299–309, Apr 2002.

[5] N. G. Stocks, "Suprathreshold stochastic resonance in multilevel threshold systems," *Physical Review Letters*, vol. 84, pp. 2310–2313, Mar. 2000.

[6] G. P. Harmer, B. R. Davis, and D. Abott, "A review of stochastic resonance: Circuits and measurement," *IEEE Transactions on Instrumentation and Measurement*, 2002.

[7] M. D. McDonnell, N. G. Stocks, C. E. M. Pearce, and D. Abbott, *Stochastic Resonance: from Suprathreshold Stochastic Resonance to Stochastic Signal Quantization*. Cambridge, UK: Cambridge University Press, 2008.

[8] A. Gelb and W. E. V. Velde, *Multiple-Input Describing Functions and Nonlinear System Design*. New York, USA: McGraw-Hill Book Company, 1968.

978-1-5386-4881-0/18/$31.00 ©2018 IEEE