UNIVERSITY OF TWENTE.

Faculty of Electrical Engineering,
Mathematics & Computer Science

Analysis and design of an ADC for a spectrum analyzer

P.l.Bicker

MSc. Thesis
September 2011

Supervisors
Prof. Ir. A.J.M. van Tuijl
Dr. Ing. E.A.M. Klumperink
M.S. Oude Alink MSc
Dr.Ir. A.B.J. Kokkeler

Report number: 067.3419

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

Abstract

Cognitive radio requires a system that is capable of quickly sensing the spectrum for available frequencies. In order for such a system to work a spectrum analyzer is required. The spectrum analyzer must be capable of detecting strong and weak signals simultaneously. This requires a high linearity in combination with low noise. A design for a spectrum analyzer was proposed in [1]. The spectrum analyzer increases linearity by attenuating the received signal. However, attenuating the signal reduces the SNR. Cross-correlation is used to reduce the noise in the digital domain.

The requirements of the ADC in the spectrum analyzer are different from a typical situation. The linearity requirement is an SFDR of at least 70dB, and in order to measure a wide spectrum at once, the bandwidth requirement is 20MHz. The input voltage swing is low as result of attenuation. The quantization error can be reduced by cross-correlation. By reducing the resolution to 1 bit, the calculations during cross-correlation can be reduced, as multiplications become single gate operations.

In order to reduce the resolution of the ADC, oversampling is required. A well known oversampling architecture is the sigma delta, which makes use of noise shaping to push noise outside the band of interest. It was found that an oversampling rate of at least 12 times is required in order to meet system requirements. A higher order loop filter reduces the amount of noise in the band of interest, but instability introduces distortion components, making a first order architecture the preferable choice. In order to achieve a high bandwidth, a continuous time converter is the preferred choice. However, the required bandwidth is hard to achieve. In current literature only implementations that reach only halve the required bandwidth or lower are found. Another issue with sigma delta converters are idle tones. It was found that these tones can mostly be reduced by dithering, at the cost of additional noise. Because of a limited bandwidth and occurrence of tones and distortion, the sigma delta architecture is found to be less suited for a spectral analyzer.

Nyquist converters require dithering in order to reduce non-linearities as result of quantization. When adding white noise to the input, a resolution of at least 7 bit is required in order to achieve the system requirements for the spectrum analyzer. Several architectures are considered. Flash requires amplification of the input signal. Pipelined and successive approximation register converters do not required high voltage swings on the input. Pipelined converters require amplification in each stage, consuming both power and reducing the linearity. High speed SAR architectures are possible in current CMOS technology, and have both a very high efficiency and have sufficient linearity when the components are correctly dimensioned.

A SAR ADC implementation is proposed, based on an existing design. By adjusting capacitors and transistor sizes the requirements for the SA are met. Linearity issues in the original design are resolved by a lower input voltage swing. The final solution achieves the required 70dB SFDR. The bandwidth requirement of 20MHz can be achieved by redesigning the control logic. The DAC, sample and hold and comparator are already capable of reaching the required speed.

Contents

Chapter 1

Introduction

Currently radio frequencies are dedicated to specific services, such as GSM and FM. Each service is only allowed to broadcast in a specific frequency band, so that interference to other services is avoided. Both commercial organizations and governments have exclusive rights to use certain frequencies. Laws prevent the use of these dedicated frequencies without a licence, whether they are actually in use or not. Since some of these frequencies are rarely used, the spectrum is utilized inefficiently. A study by Swisscom in Bern [2] shows a measurement of the spectrum between 1GHz and 3GHz during one day. In figure 1.1 it can be seen that measured over a whole day only a small portion of the spectrum is used.

In order to make use of unused parts of the spectrum, a concept called cognitive radio has been presented. Cognitive radio is a method of making use of the spectrum without interfering with other (for example licensed) users of that same spectrum. To achieve this, the system must be able to know what is going on in the spectrum, as well as being able to quickly switch to another frequency when another signal is detected. This requires a Spectrum Analyzer (SA) that is constantly sensing the spectrum and the system must have a very flexible transmitter and receiver so it can switch frequencies without interruptions.

Traditionally, narrow-band transmitters use filters (such as external LC, Ceracic or SAW filters) to filter out all unwanted frequencies. These filters

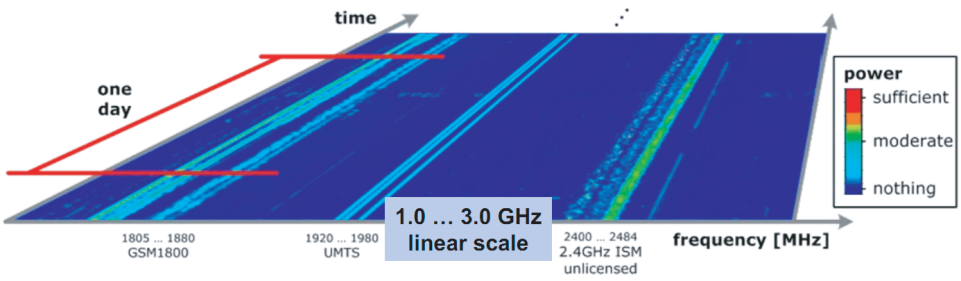

Figure 1.1: Usage of the spectrum between a band of 1GHz to 3GHz (Swisscom)

are tuned to a specific fixed frequency. Since the cognitive radio needs to be able to transmit over a wide range of frequencies, this is not a viable solution. To solve this issue, cognitive radio relies on software defined radio. Software defined radio requires a very flexible hardware frontend. Signal processing is performed by the software or reconfigurable hardware. Techniques such as polyphone multi-path [3] can be used to create the required flexibility.

1.1 The spectrum analyzer

The function of the spectrum analyzer is to monitor the spectrum, both to determine if the frequency that is in use is free, as well as to find free frequencies that can be used in case another signal is detected on the current frequency. The spectrum analyzer consists of a receiver, an Analog to Digital Converter (ADC) and digital hardware or software to measure the frequencies. Since only the strength of the signal is measured and the actual data itself is not relevant, the requirements of the receiver are different from a standard receiver. The goal of sensing the spectrum is to find an unoccupied frequency. To obtain this information, the received signal has to be transformed to the frequency domain. In order to speed up the detection time, it is of benefit to measure a wide spectrum at once, containing several channels. The spectrum can exist of both weak signals and strong signals at the same time at different frequencies. Strong signals can result in large harmonic distortion components. The spectrum analyzer can not distinguish between harmonic distortion or an actual signal, which results in false-positive detections. This creates a design challenge, as the system must be able to detect both strong and weak signals simultaneously for the spectrum analyzer to be useful.

Harmonic distortion is a result of non-linear system components. A method to decrease the harmonic distortion is to decrease the signal amplitude. However, decreasing the signal amplitude also decreases the signal to noise ratio, making the detection of weak signals harder, as those could be obscured by noise.

Figure 1.2 illustrates the problem when sensing both strong and weak signals. The figure shows two situations. In both situations, there are two signal to be detected, a strong signal and a weak signal. In situation (a) the harmonic distortion of the strong signal is larger than the weak signal. This makes it impossible to discriminate between a weak signal or distortion. In situation (b) the input is attenuated. The harmonic distortion is reduced to a level below the weak signal. However, the weak signal is also attenuated and is now below the noise floor, making it yet again impossible to detect.

There are two approaches to solve this problem, by decreasing the harmonic distortion in situation 1.2 (a), or by lowering the noise floor in situation 1.2 (b). Designing linear and efficient system components (such as mixers and amplifiers) can be hard. On the other hand, decreasing the noise can be done digitally. A technique to do this is cross-correlation. Correlation is a technique often used in GPS receivers, which receive weak signals from satellites. When taking the cross-correlation between two signals, information that correlates is conserved, while unwanted noise is eliminated. Since the data itself is not required, cross-correlation can be performed over a long period of time. Increasing the measurement time results in the signal to be measured to be

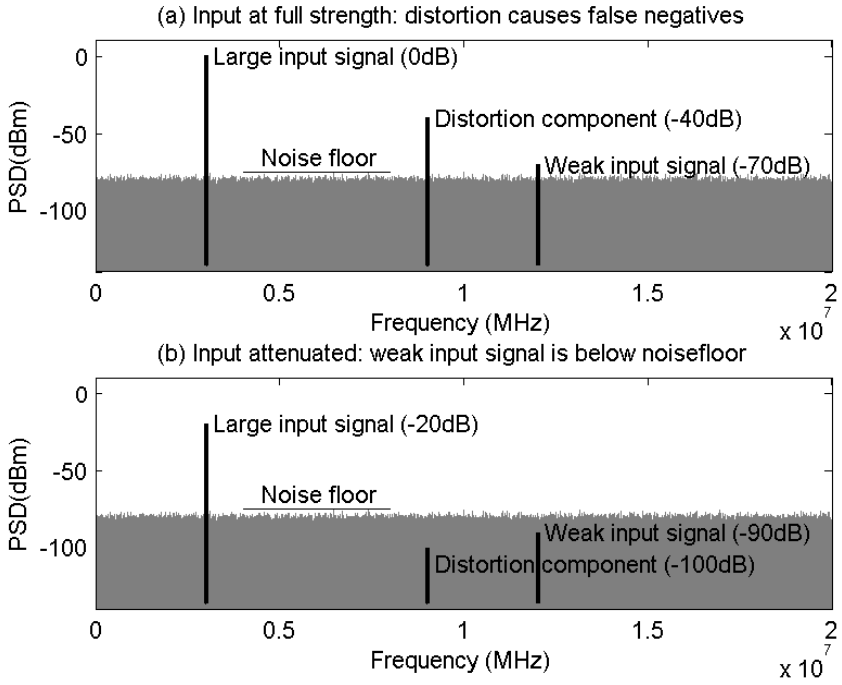

Figure 1.2: Illustration of problem of (a) too much distortion and (b) too much noise

accumulated, while the noise is reduced. When using cross-correlation as a method to reduce noise, the frontend is allowed to be more noisy and noise can be traded for linearity, which is basically what happens when the input signal is attenuated.

Commercial spectrum analyzers currently available are able offer a high performance but that is accompanied with high prices and often high power consumption. The spectrum analyzer in this work is targeted for use in mobile applications. This means the production cost and power consumption must be low. CMOS is a widely used technology that has proven itself over the years, and offers both low production costs and low power consumption. CMOS technology can be used for both analog and digital design, integrated in a single chip, making the production costs considerably lower. These advantages make CMOS the preferred choice over other technologies. Disadvantages are relative high noise levels and mismatch in the production process. These issues have to be considered but can be overcome by making a proper design choices. The challenges of creating a spectrum analyzer in CMOS is to reach both high linearity and low noise simultaneously, while power consumption remains low. For spectral sensing in cognitive radio, the accuracy of the measurement of the signal strength is less important, more important is to detect whether there is a signal at all and at which frequency.

1.2 Goal

The goal of this master thesis is to find the best solution for the ADC in the spectrum analyzer for cognitive radio. The requirements for the ADC are different from a typical application. Because cross-correlation is used, the noise is less important than the linearity, so the main focus is on linearity. Another difference is that because the signal is attenuated, the input signal of the ADC is low. Other points of consideration is optimization of the output of the ADC for easy signal processing in the digital domain, and the interface between the Analog frontend and the ADC.

In chapter 2 the spectrum analyzer is examined in further detail, and more detailed requirements for the ADC are determined. In chapter 3 and 4 the theoretical performance of two different ADC types is analyzed. In chapter 5 the advantages and disadvantage of several architectures for use in the spectrum analyzer are examined. Based on the analysis of chapter 3, 4 and 5, an architecture is chosen, and in chapter 6 a design proposal is made. Simulations are performed to confirm if the ADC meets the requirements set in chapter 2.

Chapter 2

System level analysis of the spectrum analyzer

In this chapter, a system level analysis is made of the spectrum analyzer. Each component is analyzed individually in order to determine its effect on the requirements of the Analog to Digital Converter (ADC). The requirements of the ADC are summarized in the last section of this chapter.

2.1 system overview

The complete system overview of the spectrum analyzer this work is based on [1] can be seen in figure 2.1. The proposed system consists of two signal paths both with their own antenna. A Tayloe mixer is used for its high linearity and good noise figure to down-mix a wide spectrum directly to the baseband frequency. In order to improve linearity, the signal is attenuated before the mixer. After the mixer the signal is digitized. After digitization cross-correlation is performed. This work focusses on finding the best solution for the part after the mixer and before the cross-correlation. This includes amplification, filtering and ADC.

2.1.1 Mixer and attenuation

The first components after the antenna are an attenuation circuit and a mixer (see figure 2.1). The mixer shifts the spectrum that is being analyzed directly to DC. The signal on the output of the mixer can directly be converted to a digital signal, if possible without any additional processing. The mixer contributes to the non-linearity of the spectrum analyzer and adds noise. The used mixer, the Tayloe mixer, has an exceptionally high IIP3 and low noise figure (NF) compared to other designs. The mixer is a type of sampling mixer with an additional resistor that limits the bandwidth of the mixer. This additional resistance can be the R_{on} of the transistor that makes up the switch. The IIP3 of a passive implementation of the mixer is larger than 26dBm and the NF is smaller than 6.5dB [4]. Simulations including the attenuation frontend show that the IIP3 drops at higher frequencies [1], so a more conservative value of 22dBm is used here. The bandwidth of the mixer is approximately 20MHz [1], which forms the limitation for the bandwidth of the spectrum analyzer. The

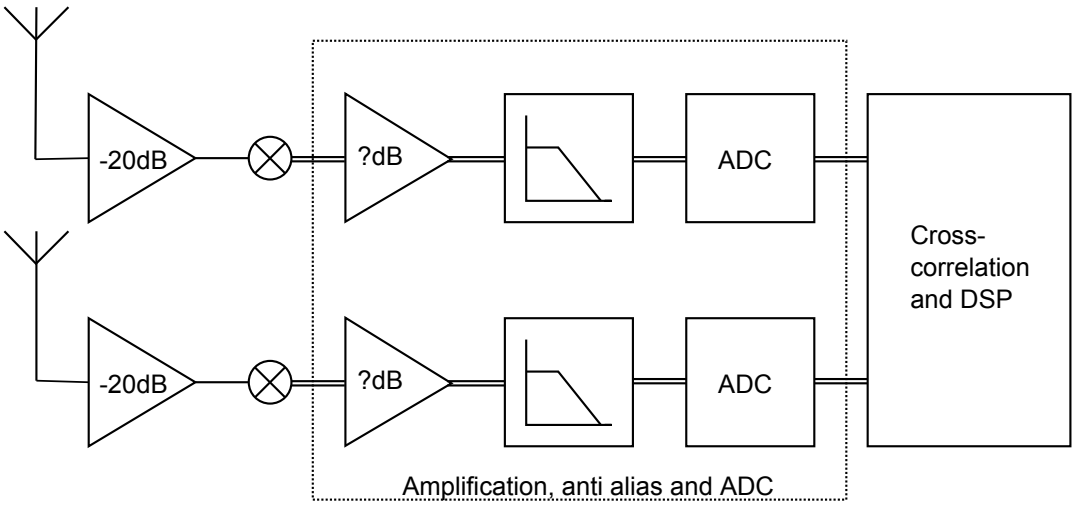

Figure 2.1: Block scheme of complete spectral sensing system

specification for the maximal signal power on the input of the SA is 0dBm, this is estimated to be the strongest signal a mobile device will receive. With a signal of 0dBm on the input, the SFDR on the output of the mixer is:

$$\Delta P = 2(IIP3 - P_{fund,in}) = 2(22dBm - 0dBm) = 44dB \tag{2.1}$$

In order to achieve an SFDR (ΔP) of at least 70dB, an attenuation is required of at least:

$$P_{fund,in} = IIP3 - \frac{\Delta P}{2} = 22dBm - \frac{70dB}{2} = -13dBm \tag{2.2}$$

In the proposed design, the attenuator is a R2-R ladder, that is capable of attenuating the input signal in steps of 6dB. The complete RF frontend (attenuation and mixer) has a noise figure of 11.2dB without attenuation and an input impedance of 50Ω [1]. When attenuation is used, the noise figure increases with steps of approximately 6dB (17.1, 23.1, 29.2 and 35.5dB respectively). With a bandwidth of 20MHz the noise power is:

$$P_{n,dBm} = -174 + 10 \times log_{10}(\Delta f) = -101dBm \tag{2.3}$$

For this work, an attenuation of 20dB is assumed. With a signal strength of $P_s = 0dBm$ the SNR can be approximated by:

$$SNR = P_s - P_n - NF = 0dBm - -101dBm - 20dB - 11.2 = 69.8dB \tag{2.4}$$

With an attenuation of 20dB, the signal power on the output of the mixer is $P_{dBm} = -20dBm$ (or $P = 0.01mW$). The maximal output voltage of the mixer with $R = 50\Omega$ is equal to $V_{out,RMS} = \sqrt{0.01mW \cdot 50\Omega} \approx 22.4mV$, or a peak value of $V_{out,p} = 31.6mV$. The mixer is implemented in a Quadrature Sampling Mixer (QSM) configuration, which consists of 4 similar branches,

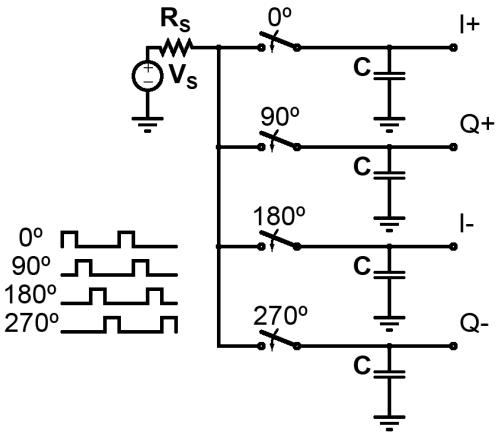

Figure 2.2: Tayloe mixer schematic [4]

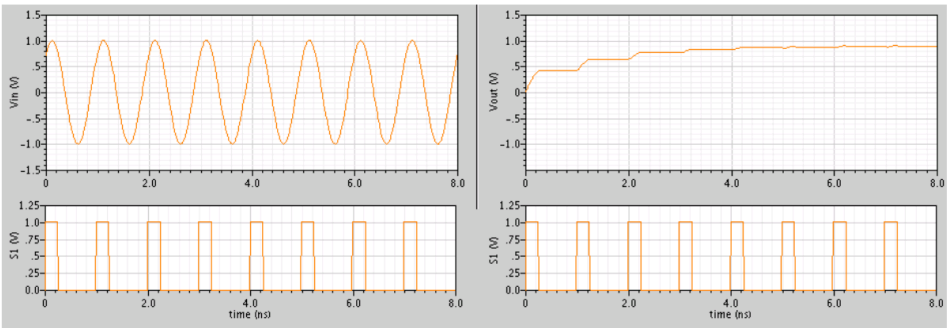

Figure 2.3: Tayloe mixer waveforms [5]

each with a 90 degrees phase shift (figure 2.2). QSM provides image rejection as well as even-order cancelation [4]. Each branch consists of a switch and a capacitor. The switch and the capacitor make up a circuit that is similar to a sample and hold circuit, which results in a sampled output on the mixer. Figure 2.3 shows an example of the output waveform of the mixer. The switch has a duty-cycle of 25%, in this period the output tracks the input. The rest of the time the output is being held. A sampled output can be directly read by the ADC, which may be beneficial. In chapter 6 this is further examined.

2.1.2 Anti-aliasing

After the mixer, the signal can be digitized. When digitizing an analog signal, aliasing can occur. Aliasing is the effect that input signals of different frequencies can result in the same sampled series. For a baseband system, the frequencies that cause aliasing are all frequencies that are above the Nyquist frequency. These frequencies are folded back in the Nyquist band, and are interpreted as lower frequencies during sampling. Figure 2.4 illustrates this effect.

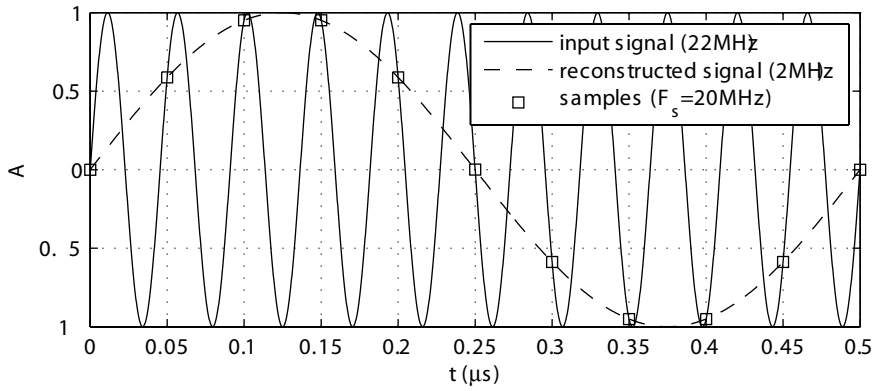

Figure 2.4: Illustration of the aliasing effect

To prevent aliasing, the higher frequencies need to be filtered out. This is achieved by an anti-aliasing filter. An anti-aliasing filter is a low-pass filter with a cut-off frequency equal to the maximal baseband frequency of the converter. The type of filter and the number of poles determine the amount of attenuation in the stopband and the roll-off.

Filter types such as the Chebyshev and elliptic filter have a steep roll-off, but this is at the cost of a ripple in the baseband. These type of filters can also have stability issues. A Butterworth filter does not have these issues, at the cost of a less steep roll-off. The Butterworth filter has a 6dB per octave attenuation of unwanted signal for each additional pole. A sharp roll-off requires a lot of poles and results in a filter that requires a large area or a filter that is external. In order to relax the filter requirements a higher sample frequency can be chosen, which is often done in practice. The resulting spacing between the baseband frequency and the alias can be used for the roll-off of the filter.

In oversampling converters this concept can be exploited greatly, as most of the alias frequencies can be filtered out in the digital domain, making the requirement on the analog alias filter much lower. Since digital filters are easier to implement and can be more efficient, a high oversampling rate is sometimes chosen for the purpose of alias filtering alone. The Tayloe mixer used in the SA design has an inherent first order low pass filter characteristic. This relaxes the requirements on the anti-alias filter by approximately one order.

2.1.3 Amplification

Amplification may be required if the ADC requires high input amplitudes in order to operate properly. Amplification consumes additional power and adds distortion components. In chapter 5 several ADC architectures are examined to determine if amplification can be avoided. An architecture that does not require any amplification is preferable as it saves power and chip area. From a system point of view, the power consumption saved by avoiding amplification can be reallocated to improve the performance of the ADC itself. As long as the power consumption of the complete system remains lower, an overall decrease

in power consumption can be achieved. This makes an ADC architecture that does not require amplification highly preferable.

2.2 Cross-correlation

After the ADC the signal can be processed digitally. Digital signal processing can be done both with hardware and software. The function of the digital frontend is to determine the spectrum of the input signal. In this section, the mechanics and types of cross-correlation are briefly described. The different types of cross-correlation require different types of digital input signals. So the requirements of the ADC depend on the type of cross-correlation that is used.

Cross-correlation is a measure of similarity between two signals, which can both be performed in the analog domain and in the digital domain. The operations consist of delays, multiplications and additions. In the analog domain these operations will add noise to the signal which can not be reduced by the cross-correlations process itself, as the two signals are already combined. This makes cross-correlation in the digital domain the preferable solution.

Cross-correlation can be used to reduce noise added to the signal by the receiver and the ADC. In order to use this technique in the SA, two separate signal paths are required. The noise in the two signals paths require to be uncorrelated with each other. Although noise is by definition a random variation, it is essential that this variation is different in both signals. An example where this may not be the case is the quantization error of an ADC. The quantization error is sometimes considered as quantization noise. In chapter 3 the properties of the quantization error is further examined.

2.2.1 Correlation function

The cross-correlation between two functions is mathematically very similar to the convolution between two signals. A convolution consists of reversing a signal, and then shift and multiply it by another signal. The difference between cross-correlation and convolution is that with cross-correlation the signal is not reversed. The cross-correlation function in discrete time is given by equation 2.5.

$$(f \star g)[n] \stackrel{\text{def}}{=} \sum_{m=-\infty}^{\infty} f[m]^* g[n+m] \tag{2.5}$$

where $f[m]^*$ denotes the complex conjugate of $f[m]$.

2.2.2 Spectral estimation methods

In order to analyze the spectrum, at some point the signal must be transformed from the time domain to the frequency domain. To do this, a spectral estimation method is required. There are three main techniques for spectral estimation: classic non-parametric estimation, parametric estimation and subspace spectral estimation. In statistics, non-parametric means that no predictions about the properties of the data is made. Parametric statistics on the other hand makes use of characteristics of the data that is known before-hand. The

advantage is that more precise predictions can be made, depending on the assumptions that are made. However, if the assumptions are not correct, the method can result in misleading data. The flexibility of the system is determined by the made assumptions. Subspace spectral estimation is a method that works best on detecting sines. For an SA it is not known beforehand what the signals that are to be detected look like, so the non-parametric approach is the best solution.

A well known non-parametric estimation of the spectrum is a Digital Fourier Transform (DFT). An efficient implementation is the Fast Fourier Transform (FFT). Disadvantages are a relative low frequency resolution and spectral leakage. The low frequency resolution makes it impossible to distinguish between two signals with frequencies very close to each other. In case of the SA used for cognitive radio, this is not a very big problem, as only the total power inside a channel that is to be measured as free or occupied is of relevance, the exact frequencies inside a single channel are not important. The spectral leakage however can cause major problems. In order the reduce the effect of spectral leaking, a window function can be applied. Section 2.2.4 goes into further detail on windows.

2.2.3 Correlation methods and computational complexity

Equation 2.5 shows the theoretical expression for cross-correlation over an infinite time frame. In practice the signal can only be measured for a limited time, and the result is dependent on the chosen time frame. There are two approaches to determine the spectrum of the cross-correlation between two signals. The first one is to first calculate the cross-correlation in the time domain, and take the FFT of the result. This method is called the XF-Correlator, where X stands for cross-correlation and F stands for Fourier transform. This method is described by equation 2.6. The other approach is to first take the FFT, and do the cross-correlation in the frequency domain (equation 2.7). This method is called FX-Correlator. From a mathematical perspective, the Fourier transformation of a convolution (2.6) results in a multiplication (2.7), so mathematically the methods are the same when the window size is infinite.

$$c_{XY}[k] = \frac{1}{N}\sum_{n=1}^{N}\overline{x[n]}y[n+k] \tag{2.6}$$

$$c_{XY}[k] = \frac{1}{N}\overline{X(f)}Y(F) \tag{2.7}$$

where N is the number of data points and, $\overline{x[n]}$ is the complex conjugate of one input signal and y is the other signal. Figure 2.5 shows the block-diagram of the two methods.

An overview of the computational complexity of both methods is shown in appendix A.4. In [1] it was found that for a large number of samples (M) and large number of lags (K) the computational complexity of the FX-correlator can be approximated by 2KML where $M = 2^L$. For the XF-correlator it can be approximated by $2KM^2$. Choosing a large M has the benefit that the noise is distributed over more bins in the FFT result.

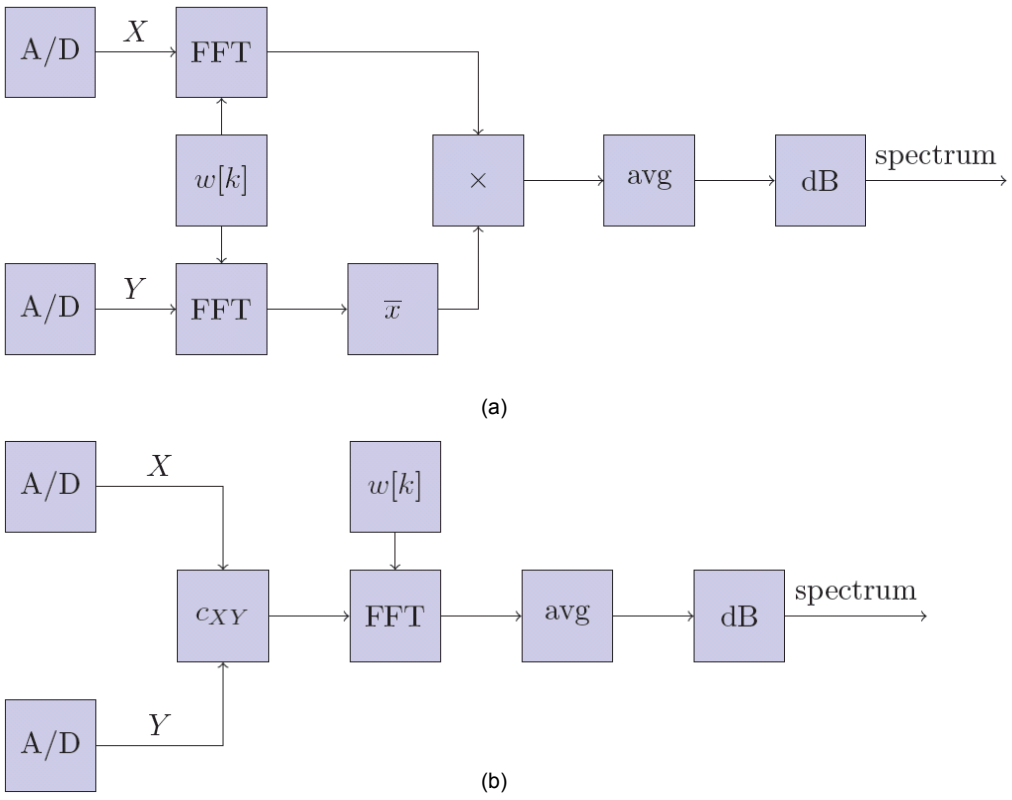

Figure 2.5: Block diagram of (a) FXC and (b) XFC [1]

The square in the approximation of the XF-correlator suggests it is more computational intensive. However, the complexity of multiplications is dependent on the number of bits of the values that are multiplied. An interesting observation can be made when the input signal is 1 bit. Multiplications of two 1 bit signals can be performed by XOR gates, making multiplications operations consist of only one logical gate. When the output signal of the ADC is one bit, the XF-correlator architecture can benefit greatly from this, as the real multiplications in the X-part of the XF-correlator are all replaced by XOR operations. When the signal is first transformed into the frequency domain the benefit of a 1 bit signal is lost, so the FX-correlator can not benefit from a 1 bit input.

To benefit from this, an ADC architecture with a 1 bit output is required. 1 bit ADCs will be examined in further detail in chapter 4. By reducing the computational complexity of the cross-correlation, power consumption can be reduced. However, a 1 bit ADC architecture may consume more power than an architecture with a higher resolution. Since overall power consumption is of relevance, a 1 bit architecture can consume more power than a multi bit architecture while overall power consumption is still lower, as the computational complexity in the digital domain is reduced. Since the exact implementation

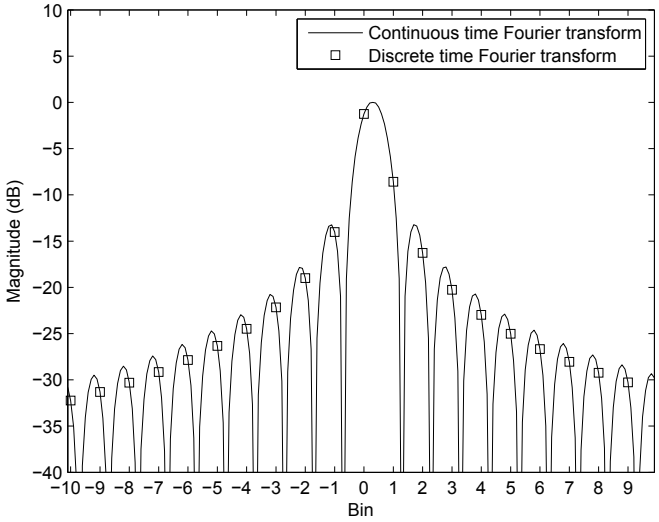

Figure 2.6: Example of spectral leakage

of the cross-correlation method is not known, the exact power consumption requirements of the ADC are also not known. Still, the power consumption should of course be as low as possible, and very large differences between different architectures still give a good indication which solution will result in an overall lowest power consumption.

2.2.4 Window functions

Spectral leakage is the occurrence of unexpected non-zero values when calculating the FFT of a signal, which are usually concentrated around the signal frequencies. Figure 2.6 shows an example of spectral leakage. The effect of spectral leakage on an SA is that it limits its dynamic range and its frequency resolution. In order to reduce the effects of spectral leakage a window functions can be used. Spectral leakage always occurs when calculating the FFT of a signal, but with certain combinations of window size and number of periods in the signal, the leakage will be exactly zero. In figure 2.6 this would be the case when the all the FFT points fall exactly in the valleys, except for the signal itself. This gives the illusion that no leakage is happening, but this is only interesting in a theoretical situation, such as simulations. The SA in this work must be capable to detect any frequency.

A window function is a set of coefficients that has the same length as the FFT that is being calculated. Applying a window is simply an element by element multiplication. Depending on the coefficients, the window function will have a different effect on the resulting spectrum. There are two important properties, the frequency resolution and the dynamic range. The frequency resolution determines the how far two signals must be apart in order to distinguish between them. When a single frequency occupies a single bin, the resolution is optimal. The dynamic range is the difference in magnitude be-

tween a signal and its leakage components. When choosing a window function, the tradeoff is between these two properties. Appendix A.6 shows an overview of the frequency characteristic of a number of available window function. The horizontal axis shows units of FFT bins. For example, a rectangular window gives the best resolution, but has a very limited dynamic range. A window like the Nuttall window has great dynamic range but low resolution.

In case of the SA used for cognitive radio, the goal is to sense free channels with certain bandwidths. The dynamic range is determined by the amount suppression of the side slopes by the window function. The leakage error is correlated with the signal to be measured, so it is not possible to use cross-correlation to attenuate it. With a requirement to achieve an SFDR of 70dB, the leakage must be suppressed at least by this amount. There are several window functions that offer this kind of suppression. In figure 2.6 is can be seen that window functions with a dynamic range of at least 70dB have the lowest resolution (about 7 bins). This has consequences for the window size when used in an SA. For example, when two channels need to be detected which are next to each other, the resolution of each channel must at least be 7 bins wide. With a lower amount of bins, it will be impossible to distinguish between the two channels, as leakage components of one channel will overlap the channels next to it, resulting in false-positive signal detection.

An SA with a total bandwidth of 20MHz capable of measuring channels of 1MHz wide will require a minimal amount of bins of $7 \times 20 = 140$. The number of bins in the PSD is equal to the window size. The most common implementation of the FFT algorithm requires an internal window that is a power of 2, but algorithms for any window size exist. For compatibility the window size is rounded to a power of 2, so a window size of 256 chosen. A slightly larger window size also reduces the effect of leakage components interfering with nearby channels. By increasing the window size the amount of noise per bin is decreased, but at the cost of more computations in the FFT. For the SA a Nuttall window is a good choice (see appendix A.6). It has a leakage attenuation of over 80dB, which is more than sufficient to achieve an SFDR of 70dB.

2.3 Requirements of the ADC

2.3.1 Summary

For cognitive radio a draft exists [6] to define a standard which states that signals as low as -116dBm need to be detected in a 6 MHz bandwidth. No transmission is allowed when a signal stronger than this value is detected. This is a rather extreme requirement, since professional spectral sensing equipment currently available on the market is capable of an SFDR of around 70-80dB in 1MHz [1]. For this reason the goal of the spectrum analyzer in this project is to achieve an SFDR of around 70dB. Another requirement of the SA is that it must be capable of detecting signals as strong as 0dBm. The problem that arises as a result of this requirement in combination with a limited SFDR is that it becomes impossible to achieve the -116dBm detection requirement.

A possible solution to this is to use two different modes of operation during the sensing period. First a large bandwidth is scanned to make a rough estimate of possible free channels, going as accurate as 70dB SFDR. After that a

candidate is picked and a more detailed scan is performed only on that specific channel. In this second scan the bandwidth is much smaller, only that of a single channel. Because the signal is much weaker, and no strong components are present in this small band, the received signal can be amplified instead of attenuated. Since the exact implementation is not yet known, the requirement on the ADC is that there has to be a flexible way to change the sample frequency. There is also a time limit to how long it is allowed to keep a frequency occupied after another transmission starts. This puts a restraint on the time available to sense the spectrum.

The amount of reduction in noise as a result of cross-correlation is determined by the amount averaging between the cross-correlations over a fixed time frame. The maximal number of time frames that can be averaged is determined by the length the time frame (and thus the number of samples) and the maximal allowed measurement time. For the SA used for cognitive radio the maximal allowed measurement time has a limit of 1 second. In section 2.2.4 is was found that the number of samples in each time frame is 256 and in section 2.1.1 it was found the bandwidth of the mixer is approximately 20MHz, so to measure an as large as possible bandwidth at once, the bandwidth of the ADC should also be 20MHz. As a result the maximal number of averaged time frames in one second is $20^6/256 = 250000$. Simulations show that for each doubling of the number of time frames, the noise is reduced by about 1.5dB [1]. With 250000 being approximately 2^{18}, the amount of noise that can be reduced in about 1 second time is $18 * 1.5dB = 27dB$ (1.5dB is the approximate attenuation of noise with each doubling of the measurement time).

As explained in section 2.2.4 using FFT to calculate the spectrum requires the signal to be windowed. In order to reach at least an SFDR of 70dB, a window with a high dynamic range must be used. The downside of these windows is that they have a lower frequency resolution. As a consequence it was found that a minimal required number of bins in the FFT is 7 for a channel, resulting in a window size of 256.

From cross-correlation alone, the noise can be reduced about 27dB in a period of about 1 second. The noise power is divided over $2^8 = 256$ bins in the FFT. The reduction of noise power per bin by increasing the windows size of the FFT is called the Noise Improvement Factor (NIF). The NIF is defined by NIF = 10log(K), where K is a number between N/4 and N/2 [7] and N is the number of bins. So the NIF of a window of 256 bins is between NIF=10log(256/2)=21dB and NIF=10log(256/4)=18dB. The advantage of the NIF depends on the measured signal. When the signal is concentrated on one frequency, the power of the signal is concentrated in an single peak in the FFT. But when the signal is uniformly spread over a larger bandwidth, the signal power is spread over the bins that span the frequency range of the signal. So in this case, when the number of bins in the FFT is increase, both the noise as well as the signal power is divided over more bins. The exact benefit of the NIF in an SA depends on the spectrum of the signal that is measured. For this work it is assumed that the signal is concentrated on a small frequency band and the NIF is of benefit. The most conservative estimate of 18dB is used.

From this information, an estimate can be made on the minimal allowed SNR of the system. To detect a signal of at least 70dB under maximal magnitude, the minimal allowed SNR equals the wanted dynamic range (70dB) minus the reduction as a result of cross-correlation (27dB) minus the noise reduction

Table 2.1: Requirements and specifications for the ADC in a spectrum analyzer

Input voltage swing	$63mV_{pp}$
SFDR	$> 70dB$
SNR	$> 25dB$
Bandwidth	$20MHz$

in the FFT (18db):

$$SNR_{min} = 70dB - 27dB - 18dB = 25dB \tag{2.8}$$

Such low SNR requirement gives much freedom in circuit design, and can be exploited to significantly improve linearity. The above figure for SNR applies to a system where FX-Correlation is used. It is assumed that for a XF-Correlation system a similar performance can be achieved. The choice between XF-Correlation and FX-Correlation depends on the resolution of the digitized signal. Two 1 bit signals can be multiplied by XOR-ports, reducing the overall complexity of the XF-Correlation design significantly. Therefore a 1 bit converter is the preferably choice from the perspective of the cross-correlation.

2.3.2 Overview of the requirements

The requirements of the ADC that can be derived from the previous sections are summarized in table 2.1.

These requirements on the ADC make the design choices different from an ADC used in a standard situation. The goal of this work is to find out which architecture is most suited for use in the SA, with an extra focus on 1 bit converters. The next two chapters are about the theoretical performance of two ADC types: Nyquist rate converters and oversampling converters.

Chapter 3

Analysis of Nyquist rate ADCs

In the previous chapter it was concluded that the type of ADC depends on the type of cross-correlation that is used. When an FX-Correlator is used in the spectrum analyzer, the number of bits of the digital signal is not of much relevance to the computational complexity of the cross-correlation. When resolution is not an important factor, a Nyquist rate converter can be used, and the required performance can be achieved by choosing the appropriate resolution. In this chapter an analysis of the quantization error and the linearity of an idealized ADC model are made. The ADC is treated as a black box, where an analog signal goes in, and a sampled and quantized signal comes out. The purpose of this analysis is to determine the minimal requirements of the resolution and linearity for an ADC used in the spectrum analyzer.

3.1 Nyquist rate ADC's

ADC's can be categorized in two groups: Nyquist converters and oversampling converters. The NyquistShannon sampling theorem states that in order to reproduce a signal accurately it has to be sampled at least at twice its maximal frequency. This means that a converter has to operate at least at twice the maximal baseband frequency, which is called the Nyquist frequency. Oversampling converters operate at frequencies that are even higher. The benefit of oversampling is that more quantization noise is pushed outside the baseband. Chapter 4 goes into further detail on oversampling converters.

The performance of an ADC is often characterized by the Signal to Noise Ratio (SNR) and Spurious Free Dynamic Range (SFDR). The SNR is the signal strength compared to the total noise, while the SFDR is the signal strength compared to the largest spurious signal. In this work the focus is on linearity of the ADC, as the noise can later be reduced by cross-correlation. This means that the SFDR is more important than the SNR.

The non-linearity of Nyquist converters can be expressed in terms of INL (Integral Non-Linearity) and DNL (Differential Non-Linearity). DNL is the error between each conversion step, and gives insight in monotonic behavior [7]. The INL is the error between the ideal conversion curve and the actual conversion curve. INL is directly related to harmonic distortion. The shape of the INL determines the magnitude of the distortion components [7]. Therefore the INL is the most important characteristic to get insight in the linearity of

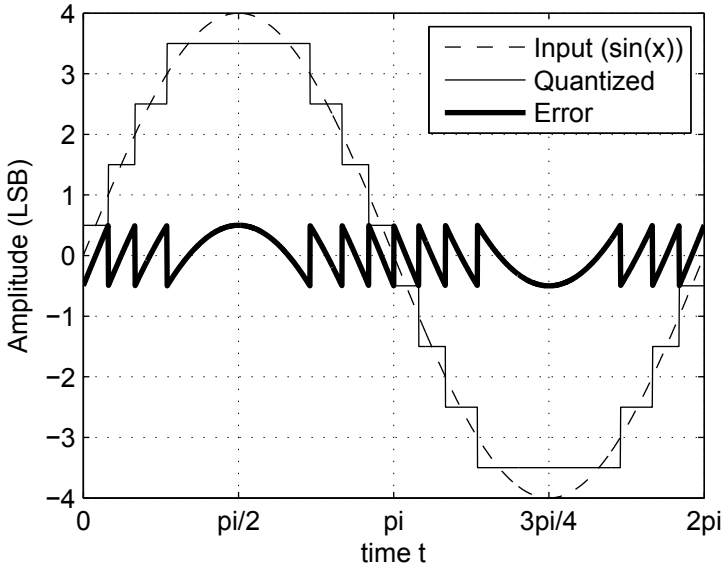

Figure 3.1: Quantization of a sine and the resulting error

an ADC converter. Since the distortion components are harmonics of the input signal, they are on a fixed position in the spectrum. They are correlated and cannot be attenuated using cross-correlation. In section 3.7 the relation of INL and distortion components is explained in further detail.

The quantization error itself also consists of harmonics. These harmonics are spread over a very wide spectrum and behave a lot like noise. The position of these harmonics are deterministic, so they can not be attenuated using cross-correlation. In the next section these harmonics are further analyzed, and a method is explained how the distortion components can be reduced at the cost of noise, which can be attenuated by cross-correlation.

3.2 Quantization error

The quantization error arises as result of representing a signal with infinite variation with a finite set of values and a finite interval. Figure 3.1 illustrates this process and shows the resulting error. The quantization error is sometimes regarded as noise with a white spectrum. In many situations this is a sufficient way to model the error. However, when using cross-correlation the exact shape of the spectrum is relevant, as correlated cannot be removed. When assuming quantization noise can be treated as white noise, it can be characterized in terms of SNR. The SNR for a Nyquist converter can be determined by equation (3.1). See appendix A.5 for more details.

$$SNR = 10log_{10}\frac{3}{2}2^{2b} = 6.02b + 1.76[dB] \tag{3.1}$$

From this equation, the minimal required number of bits can be determined for the SA. In the previous chapter (see table 2.1) it was determined the SNR of

the system should be at least 25dB. Using equation (3.1), the minimal required amount of bits can be determined.

$$b = \frac{SNR - 1.76dB}{6.02dB} = \frac{22dB - 1.76dB}{6.02dB} \approx 3.86 \tag{3.2}$$

Rounding this value up results in 4 bit. However, this figure is only valid under the assumption that the quantization error is white noise, which is not the case. In the next section the deterministic nature of the quantization error is investigated.

3.3 Spurs in ideal ADCs

The quantization error causes spurs on the output as a result of the quantization process. The quantization error is deterministic and its shape is dependent on the input signal. So even though the spectrum of the quantization error looks like white noise, it can not be attenuated using cross-correlation when the inputs are identical. In this section the impact of the quantization error for a SA is analyzed. After that a solution to overcome the deterministic nature of these quantization spurs is presented. A sine wave is used as input signal for for analysis of the quantization error.

When quantizing a sine, two types of errors occur. One error has a bell shape, which is the result of quantization of the peaks of the sine. The other error has a sawtooth shape, which is the result of quantizing the slope of the sine around the zero crossing [8]. Figure 3.1 shows the quantization error for a 3 bit converter. The error is the original sine minus the quantized signal. The bell shaped error appears around pi/2 and 3pi/4 and the sawtooth shaped error appears around 0, pi and 2pi. In between is a transition region. The bell shaped error results in low frequency harmonics, the sawtooth error results in high frequency harmonics. The spectrum of the quantization error is shown in figure 3.2. Because of aliasing all spurs fold back into the baseband, so both the spurs as a result of the bell shaped error and sawtooth shaped error show up in the baseband as distortion components.

3.4 Simulation and approximation of spurs as a result of quantization

The quantization noise can be characterized by a mathematical approximation. To do this, the Fourier transform of the quantization error is taken to determine its spectrum. This subject has been researched before in [9] and [1], so this section will contain a summary of the results. In order to find the highest harmonic for all quantization levels, these calculations are divided in two parts. One part for the 3rd harmonic caused by the bell shape, and one part for the harmonics caused by the sawtooth shape.

The bell shape error results in low harmonics (3rd, 5th, 7th etc). Using the method from Blachman [9] the harmonics as a result of the error can be approximated. This approximation is valid for harmonics in the region $p << 2\pi A$, where p is the harmonic and A is the amplitude (expressed in LSB). The 3rd harmonic is the strongest and can be approximated using equation 3.3.

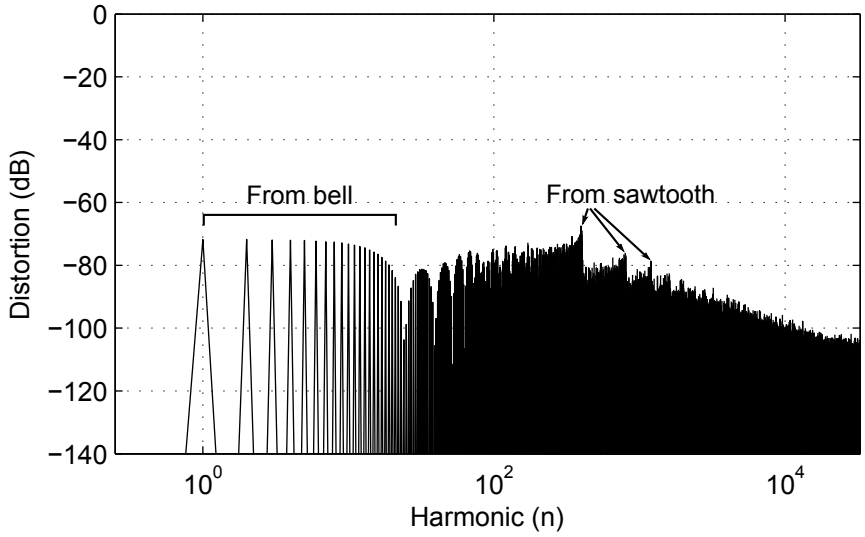

Figure 3.2: Output spectrum of an 8 bit ideal ADC with a full scale sine as input, simulated in Matlab with a window size of 2^{17}.

In this equation [A] is the largest integer not exceeding A. For more details on deriving this equation see Appendix A.2.1.

$$A_3(A) = -\frac{2}{3\pi} \sum_{k=-[A]}^{[A]} \left(4 \left(\frac{k}{A} \right)^2 - 1 \right) \left(1 - \frac{k^2}{A^2} \right)^{1/2} \tag{3.3}$$

The approximation of the strongest harmonic in the sawtooth shaped error was analyzed in [1]. The approximation is valid for the harmonics in the region $p \approx 2\pi A >> 1$. This error becomes dominant for a higher number of quantization levels. See appendix A.2.3 for the derivation of the approximation. The results of the analysis show that the strongest harmonic as a result of the sawtooth error is proportional to:

$$A_p(m = 1) \propto \frac{1}{p^{\frac{1}{3}}} \tag{3.4}$$

This is equivalent to a decrease of 2.01dB per bit. In figure 3.3 the approximation of the bell shape (equation 3.3) and the linear fit found in [1] are plotted relative to the fundamental sine for a range of quantization levels. The figure also shows simulated values of the SFDR. The simulations are done with Matlab using an ideal ADC model. In the figure it can be seen that for a low number of quantization levels the simulation results follow the 3rd harmonic estimation, and for a higher number of quantization levels the simulation results show a 2dB increase each bit. The SFDR can be approximated by a linear fit for $n < 4$: [8]:

$$SFDR_{N<4} = 9.03n + 0.91dB \tag{3.5}$$

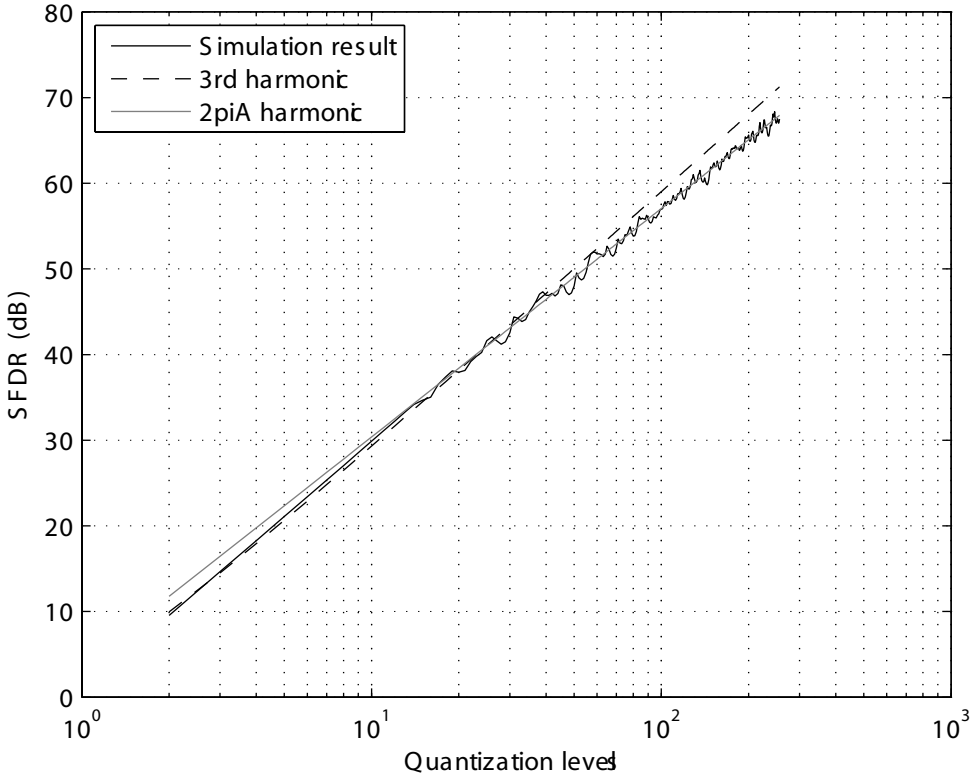

Figure 3.3: SFDR with relation to the number of quantization levels, both calculated and simulated

For $n > 4$ the SFDR can be approximated by [1]:

$$SFDR_{N>4} = 8.07n + 3.29dB \tag{3.6}$$

These approximations only apply to an ideal ADC without any noise and INL. In practice this is never the case as there will always be some noise. Since noise causes random variations during quantization, the resulting spectrum will be different. The next section will go into further detail of the effect of noise.

3.5 Effect of noise

Noise on the input can significantly affect the shape of the quantization error. Because noise causes random variations of the quantization level decision, the resulting quantization error is randomized. This effect can be used to reduce distortion components. The technique is called dithering, and has successfully been used in application such as reducing undesired colors in tv signals, resolving signals smaller than the quantization step, or in general improvement of resolution and linearity. White noise has a gaussian distribution that is

described by $f(x) = (2\pi\sigma^2)^{-1/2}e^{-x^2/2\sigma^2}$, were σ is the variance. Adding the noise to the input signal results in:

$$\overline{q}(t) = \int q(t+d) \cdot p(d)dd \tag{3.7}$$

where $p(d)$ is PDF of the noise, q(t) is the quantization error and $\overline{q}(t)$ is the error after dithering. This equation shows the convolution between the original input signal with the probability density function of the noise. A convolution in the time domain corresponds to a multiplication in the frequency domain, which simplifies the calculations. Since it is the spectrum that is of interest in the first place, the Fourier transform is used. The Fourier transform of (3.7) is:

$$\overline{Q}(f) = Q(f) \cdot P(f) \tag{3.8}$$

The Fourier transform of the gaussian noise is $P(f) = \exp(-2\pi^2\sigma^2 f^2)$ (appendix A.1.2), where σ is the RMS-noise and f is the frequency n (cycles per quantization step). The equation for $Q(f)$ can be found in the appendix, equation A.20. Filling in P(f) and Q(f) results in:

$$A_p = \delta_{p1}A + \sum_{n=1}^{\infty} \frac{2}{n\pi} \exp(-2\pi^2\sigma^2 n^2) J_p(2n\pi A) \tag{3.9}$$

Where δ_{pq} is 0 for $p \neq q$ and 1 for $p = q$. This equation describes the quantization error including dithering. From this equation the required noise level can directly be calculated. For example, for a 7 bit converter the strongest harmonic is equal to 59.5dB (from eq. 3.6). In order to achieve an SFDR of at least 70dB the strongest harmonic needs to be attenuated by 10.5dB. This means that

$$20log_{10} \sum_{n=1}^{\infty} \exp(-2\pi^2\sigma^2 n^2) = -10.5dB \tag{3.10}$$

For $n > 1$ the exponential diminishes so fast that only n=1 is significant [9]. This results in:

$$20log_{10}(\exp(-2\pi^2\sigma^2) = -10.5dB \tag{3.11}$$

from this equation the RMS noise can be determined:

$$\sigma_n = \sqrt{-\frac{ln10^{-\frac{10.5}{20}}}{2\pi^2}} \approx 0.248[LSB] \tag{3.12}$$

or expressed in dBFS

$$\sigma_n = 10log_{10}\frac{sigma_n[LSB]}{2^{N-1}} = 10log_{10}\frac{0.248}{64} \approx -48dBFS \tag{3.13}$$

In a converter operating at Nyquist rate the higher harmonics from the quantization error will fold back into the band of interest. At some frequencies two or more of the stronger harmonics are added up, resulting in higher distortion. Therefore in practice a higher noise level is required than the -48dB calculated here.

3.6 Simulation with noise

The balance between number of bits and amount of additional noise depends on the required SFDR. The ADC in the SA is required to achieve an SFDR of at least 70dB. The minimal amount of noise required for dithering can be calculated as shown above. The upper limit is determined by the combination of the quantization noise and the dither noise.

In order to get insight in the optimal resolution of the ADC, simulations are done in Matlab and shown in figure (3.4). In the simulation an ideal Nyquist converter model is used with noise added to the input. The horizontal axis shows the RMS noise level with respect to the maximal input amplitude. A sweep is made of the input noise and the resulting ratio between strongest unwanted bin to signal is determined, which is depicted on the vertical axis. The strongest unwanted bin in the FFT can either be a distortion component or noise (hence the curve in the figure can not be called SFDR). The SNDR is also determined and depicted in figure 3.4. The input signal is a full scale sine.

In chapter 1 it was found that the maximal attenuation of noise in 1 second as a result of both cross-correlation and FFT window size is about 45dB. By choosing a window size of 2^{17} samples, the noise improvement factor is about $NIF = 10log(2^{17}/4) \approx 45dB$ compared to a sinusoidal input with a fixed frequency. This is about equal to the maximal attenuation of noise achievable by the cross-correlation system. Figure 3.4 shows the simulation results and the calculated values, which match correctly. The point where the simulation results stop following the calculated curves is where the input noise is starting to dominate in the FFT result.

From figure 3.4 the minimal resolution to achieve an SFDR of 70dB can be determined. The simulation shows that the minimal resolution is 6 bit. However, the required noise for dithering is high and the resulting SNDR is low, making 6 bit not the best option. An ADC with a resolution of 9 bit does not require any dithering in order to attenuate the distortion from quantization, as the distortion peaks are, as expected from equation 3.6, already more than 70dB below the maximal signal power. When the requirement of 70dB SFDR is met, the next important property is the SNR. Attenuating noise with cross-correlation takes both time and consumes power. To choose an optimal resolution, parameters like power consumption of cross-correlation and the amount of noise on the input signal and noise generated by the ADC internally must be known. When for example the noise is below approximately 60dB compared the the signal, a 9 bit converter can be the best choice. When the noise is higher, the additional bits of a 9 bit converter have no or little advantage over an 8 or 7 bit converter. The conclusion that can be drawn from figure 3.4 is that the resolution of the ADC should be either 7,8 or 9 bit, depending on the amount of noise.

In section 2.2.3 it was concluded that a 1 bit converter could result in a significant reduction in complexity of the cross-correlation. However, the amount of noise that needs to be added to a single bit, or even 2 or 3 bit, Nyquist converter is so high that it is no longer realistic to attenuate it to acceptable levels using cross-correlation. The time it would take to detect a signal 70dB below the maximal signal range would be magnitudes greater than 1 second. For a 1 bit converter, the noise that is required to achieve and SFDR of 70dB is about -10dB compared to the signal. This means that the noise needs to be

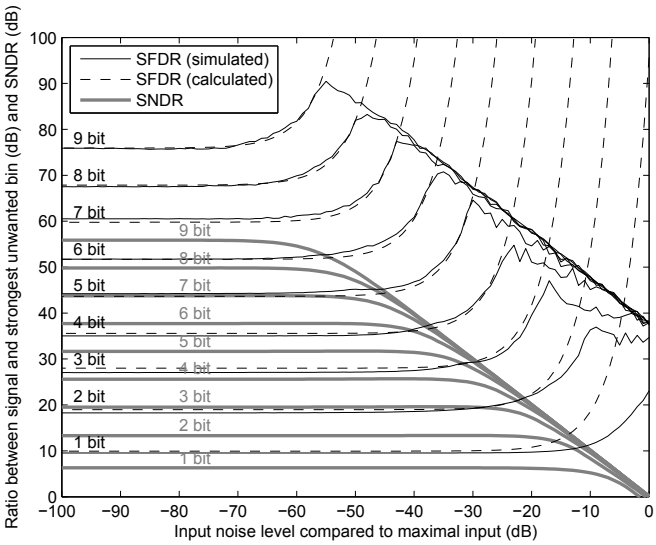

Figure 3.4: Effect of dithering on resolutions of 1 bit to 9 bit

attenuated 60dB. The attenuation of the noise in the FFT is about 18dB (due to NIF), so 42dB has to be attenuated by cross-correlation. When using cross-correlation, with each doubling of the measurement time, the noise is reduced approximately 1.5dB. In other words, the measurement time has to be doubled 28 times. A window of 256 samples takes $6.4\mu s$ to measure. Multiplying this with 2^{28} results in a measurement time of approximately 1717 seconds. It can be concluded that a 1 bit Nyquist converter is not suitable for usage in the SA. Another implementation of a 1 bit converter is the oversampling converter. This type of converter is the subject of chapter 4.

3.7 Effect of INL and DNL

In the previous sections only ideal ADC's are considered. In reality, the performance of an ADC will suffer from all kind of non-idealities caused by mismatch and non-linearities between and in the components. The non-ideality's result in deviation from the ideal quantization staircase. There are two ways of expressing this error, as Differential Non-Linearities (DNL) or as Integral Non-Linearities (INL). DNL is the error between every subsequential step in the staircase, and the INL is the error between the ideal and the actual step. The INL and DNL are mathematically described by (3.14) and (3.15) [7].

$$INL = \frac{A(i) - i \times A_{LSB}}{A_{LSB}}, \forall i = 0...(2^N - 1) \tag{3.14}$$

$$DNL = \frac{A(i+1) - A(i)}{A_{LSB}}, \forall i = 0...(2^N - 2) \tag{3.15}$$

Where A(i) is the analog value where the digital code trips from i to i+1 and A_{LSB} is the amplitude of the least significant bit. The DNL only shows

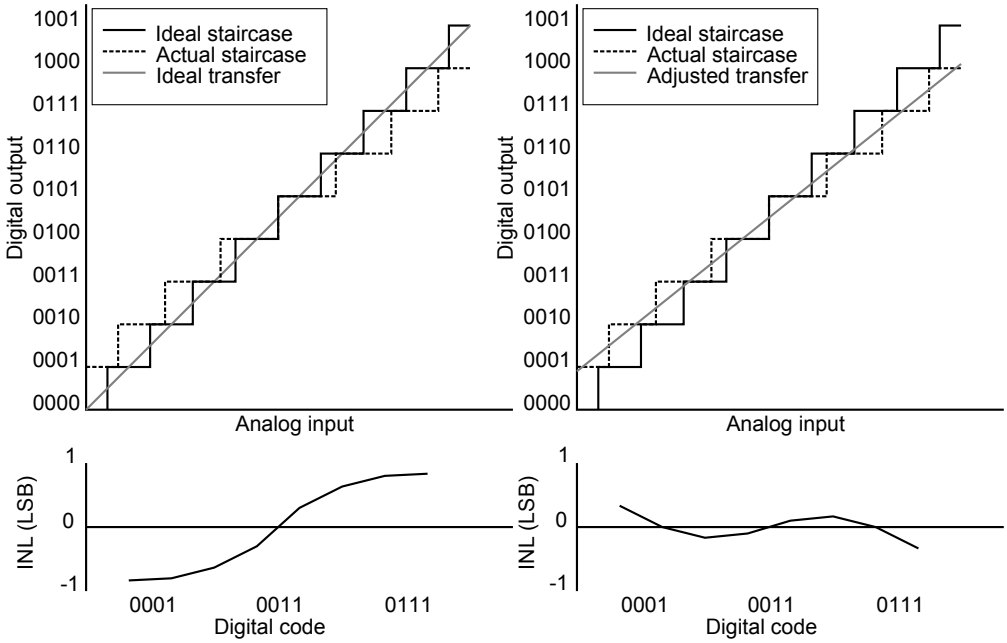

Figure 3.5: Two methods of expressing INL

the error between single steps, and does not give a good indication of harmonic distortion. Instead, the DNL gives insight in the the SNR. Equation 3.16 shows the relation between DNL and reduction in SNR based on equation 3.1 [7].

$$SNR = 6.02b - 9.03 - 10log_{10}\left(\frac{1}{12} + \frac{DNL^2}{2\alpha^2}\right)[dB] \tag{3.16}$$

Where α is the threshold value for the stochastic variable in a normal distribution describing the offsets between subsequential steps [7]. The DNL can show an important effect called non-monotonicity, which means that an increasing input magnitude results in a decreased output magnitude. This behavior is especially significant in control loops, where corrections must always results in zero or a positive output. For an SA this is not of significant relevance.

The shape of the INL determines the harmonic distortion. For example, an INL that has a second order shape will subsequently cause a 2nd order distortion component, an INL with a third order shape will subsequently result in a third order distortion component. There are two methods to define the INL. One is to take the ideal full scale and compare each level with it. The deviation between the two is the INL. The other method is to measure the derivation of each step against a best fitting straight line. This can result in an amplification error and a DC offset, but the maximal INL is then smaller. In differential designs this is often not an issue, and is therefore the preferred way of expressing the INL. In the next section the harmonics in the output as a result of INL are calculated and simulated in order to determine the maximal allowed INL while still achieving the reguirements given in table 2.1.

3.7.1 Calculation of harmonics on the output as a results of INL

The type of harmonic distortion depends on the actual implementation of the ADC. In many situations, the third order harmonic is the largest problem. Second order distortion can often be prevented by making use of differential designs [7].

In order to determine the maximal allowed third order deviation in the INL curve, a pure third order deviation is added to the quantization staircase. The magnitude of this deviation can directly be translated in to a third order harmonic distortion at the output. The output can be approximated by a Taylor expansion:

$$y(t) = \alpha_1 x(t) + \alpha_2 x^2(t) + \alpha_3 x^3(t) + ... \tag{3.17}$$

using $x(t) = Acos\omega t$

$$y(t) = \alpha_1 Acos\omega t + \alpha_2 A^2 cos^2\omega t + \alpha_3 A^3 cos^3\omega t + ... \tag{3.18}$$

$$y(t) = \alpha_1 Acos\omega t + \frac{\alpha_2 A^2}{2}[1 + cos(2\omega t)] + \frac{\alpha_3 A^3}{4}[3cos\omega t + cos(3\omega t)] + ... \tag{3.19}$$

Since only the third order term is of interest, the second and higher order terms are ignored. The harmonic distortion as a result of the third harmonic can be calculated by:

$$HD_{3,\%} = \frac{(\alpha_3 A^3/4)^2}{(\alpha_1 A + 3\alpha_3 A^3/4)^2} \tag{3.20}$$

To simplify the equation an amplitude of $A = 1$ and the a gain of 1 ($\alpha = 1$) is chosen. The third order term in the denominator can be ignored as its contribution is very small and makes the calculation unnecessary complex. So $HD_{3,\%}$ becomes:

$$HD_{3,\%} \approx (\alpha_3/4)^2 \tag{3.21}$$

Or in terms of α_3 the equation becomes:

$$\alpha_3 \approx 4\sqrt{10^{HD_{3,dB}/10}} \tag{3.22}$$

In order to calculate the maximal allowed deviation of the third order non-linearity compared to the ideal transfer, first the linear component is determined:

$$y_1 = \frac{\alpha_1 x_{max} + \alpha_3 x_{max}^3}{x_{max}} x = (\alpha_1 + \alpha_3 x_{max}^2)x \tag{3.23}$$

The difference between this straight line and y(t) is the deviation caused by the third order non-linearity. When $x_{max} = 1$ and $\alpha_1 = 1$:

$$y - y_1 = \Delta y = (\alpha_1 x + \alpha_3 x^3) - (\alpha_1 x + \alpha_3 x) = \alpha_3 x^3 - \alpha_3 x \tag{3.24}$$

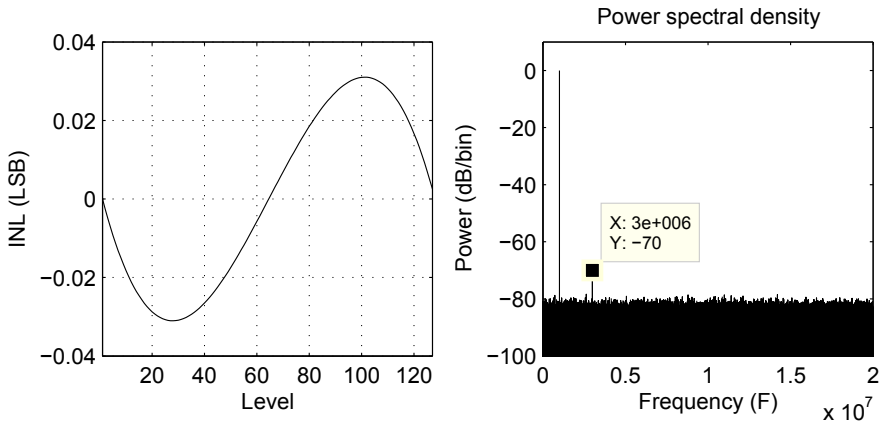

Figure 3.6: (a) 3rd order INL and (b) the resulting PSD

To find the maximal deviation, the point were the derivative is zero is calculated. With $\alpha_3(3x^2 - 1) = 0, x = 1/\sqrt{3}$. The normalized peak-to-peak output swing is equal to $y_{max} = 2A(\alpha_1 + \alpha_3)$. α_3 is small compared to α_1 and can be ignored, which results in $y_{max} = 2$. The deviation with respect to a full output swing can now be calculated:

$$\frac{\Delta y}{y_{max}} = \frac{\alpha_3}{3\sqrt{3}} \tag{3.25}$$

The above calculations result in a general equation to calculate the INL in LSB for the third order distortion:

$$INL_3 = \frac{N4\sqrt{10^{HD_3/10)}}}{3\sqrt{3}} \tag{3.26}$$

where N is the number of steps of the ADC and HD_3 is the third harmonic in dBFs. The resulting INL is in LSB. For example, for a 7 bit converter and $HD_3 = -70dB$ the maximal third order INL becomes:

$$INL_3 = \frac{N4\sqrt{10^{HD_3/10)}}}{3\sqrt{3}} = \frac{2^7 4\sqrt{10^{-70/10)}}}{3\sqrt{3}} \approx 0.031[LSB] \tag{3.27}$$

Simulation confirm these calculations. Using Matlab, a third order non-linearity with the maximal deviation found in equation 3.27 is added to the quantization staircase, resulting in an INL shown in figure 3.6 (a). The resulting spectrum is shown in figure 3.6 (b), and shows an SFDR of 70dB.

3.7.2 The effect of INL on cross-correlation

When two ADC's have the same INL, it is not possible to attenuate the distortion when taking the cross-correlation of the output of these converters, as the distortion components are related to the input signal, and therefore have the same shape in both converters. However, when the INL is different, the

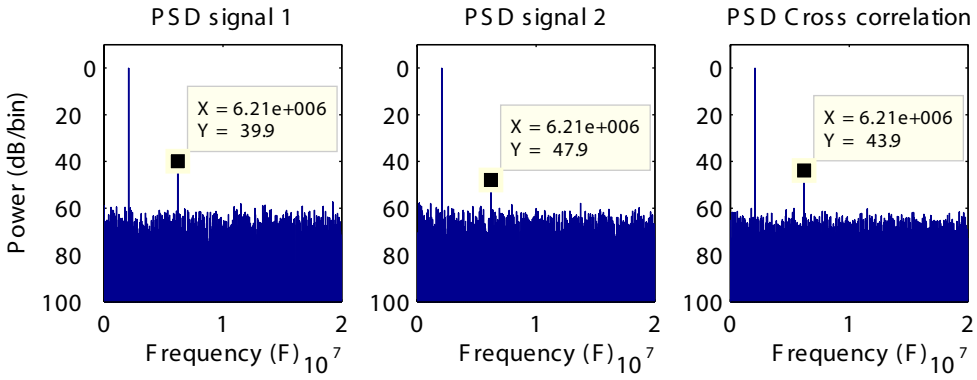

Figure 3.7: Spectrum (c) of the Cross-correlation between a 3rd order distortion component of -40dB (left) and -48dB (middle) resulting in 44dB (right) after cross-correlation.

resulting distortion can be attenuated. When only one ADC causes harmonic distortion, all distortion peaks are attenuated at the same rate as the noise by cross-correlation, as the distortion peaks are completely uncorrelated to the other signal path. In practice, both ADC's will have an INL greater than zero. However, the amount of distortion can differ as a result of non-ideal process effects such as mismatch. Since cross-correlation is a multiplication of two signals, the resulting power of a signal after cross-correlation can be determined by equation (3.28)

$$P_{CC} = \sqrt{P_{path1} \times P_{path2}} \tag{3.28}$$

For example, if in path one there is a harmonic of -40dB and in path two there is a harmonic of -48dB, the resulting amplitude after cross-correlation is:

$$A_{CC} = \sqrt{A_{path1} \times A_{path2}} = 10log_{10}\left(\sqrt{10^{(40/10)} \times 10^{(48/10)}}\right) = -44dB \tag{3.29}$$

Simulation confirms these values. Figure 3.7 shows the cross-correlation spectrum where the magnitude of the distortion in path 1 is -40 dB and in path 2 -48dB. The resulting magnitude of the harmonic after cross-correlation is equal to the calculated value of -44dB.

3.8 Conclusions

Although the deterministic quantization error can not be attenuated using cross-correlation, this problem can be solved by adding noise. Added noise will be in the baseband and needs to be removed again using cross-correlation or other methods after conversion. The minimal requirement of the ADC is to reach an SFDR of 70dB (table 2.1). This results in a limit to the amount of noise that can be added. Both calculations and simulations show that in order keep the signal to noise bin ratio below 70dB, at least 6 bit is required.

This however requires a very high amount of noise attenuation, making the measurement time close to 1 second, which is set as the upper limit for the SA. Choosing a resolution of 7 bit or higher significantly decreases the measurement time.

Distortion as a result of INL can not be attenuated using cross-correlation. However, the magnitude of the distortion is averaged between the two signal paths, so there is still a benefit. To achieve a 70dB SFDR, the maximal allowed deviation from the ideal transfer function is about 0.031 LSB for a 7 bit converter.

Chapter 4

Analysis of oversampling ADCs

In chapter 2 it was found that when XF-Correlation is used in the spectrum analyzer, the computational complexity of the cross-correlation is reduced when the resolution of the signal is reduced. With a 1 bit resolution, the multiplication between the two signals become a XOR operation, which reduces the computational complexity significantly. However, in the previous chapter is was found that for a Nyquist rate converter a resolution of at least 7 bit is requires to achieve the required performance. To solve this problem, oversampling can be used. In contrast to Nyquist converters, oversampling converters operate at higher sample frequencies than two times the bandwidth of the input signal. By doing so, more noise is pushed outside the band of interest. In this chapter an analysis is made of the performance of oversampling converters for a spectrum analyzer in order to determine the minimal required oversampling rate and to see if linearity requirements are met.

4.1 Noise in oversampling converters

The oversampling rate (OSR) is equal to $OSR = f_s/2f_b$, where f_s is the sample frequency and f_b is the bandwidth or highest frequency of the signal. The mean squared quantization noise power of a Nyquist converter is determined by (see appendix A.5):

$$\varepsilon_{RMS}^2 = \frac{1}{12} LSB^2 \tag{4.1}$$

This is the noise inside the baseband, between 0Hz and $0.5f_s$. When oversampling is used, this noise is spread over a wider spectrum. For each doubling of the sample rate, the noise is halved:

$$P_n = \frac{\varepsilon_{RMS}^2}{OSR} \tag{4.2}$$

The effect is illustrated in figure 4.1. Although there is an improvement in performance, the main downside of oversampling is the increased power consumption as a result of high clock frequencies. The benefits may not outweigh this drawback. There is however a method which makes oversampling converters a lot more attractive. The idea is to shape the noise into a non uniform spectrum. This is called noise shaping. The oversampling architecture that

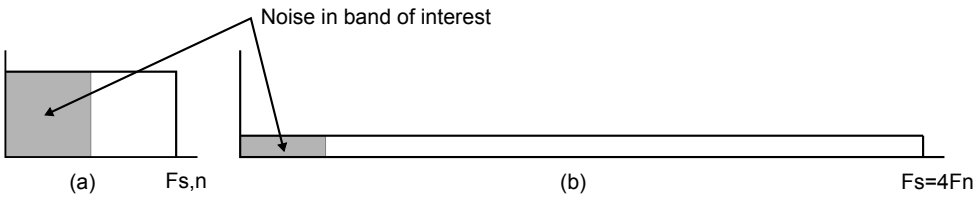

Figure 4.1: Noise in the band of interest of (a) a Nyquist converter and (b) a 4 times oversampling converter

makes use of noise-shaping is the sigma delta converter. This type of converter works on the principle of feeding back the error and correcting for it. The noise is shaped by a loop filter, and is pushed to higher frequencies, outside the band of interest. The following two sections go into further detail on noise shaping and the sigma delta converter.

4.2 Noise shaping

Noise shaping is a procedure that shapes the uniform spectrum of the noise into a non uniform shape. This can be beneficial when noise is pushed to frequencies outside the band of interest, with the result that the noise inside the band of interest is reduced. Figure 4.2 shows the block diagram of a noise shaper. Although the total noise power does not decrease, the noise at higher frequencies can be filtered out with a digital filter. This results in an improved SNR. In the noise shaper in figure 4.2, first the quantization error is measured, then this error is filtered by a loop filter, and finally it is fed back to the input. This concept is used in Sigma Delta converters. The order of the filter determines the shape of the noise. Higher order filters push more noise to higher frequencies. For a spectrum analyzer, the total noise in the band of interest is of less importance. Since the noise is no longer uniformly distributed over the spectrum, more important is the frequency where the noise power is largest, as this will be the noise that determines the performance of the SA. Since the noise is pushed to higher frequencies, the noise power is highest at f_b. To find this value, first the transfer function of the input + noise to output is determined in the z domain:

$$Y(z) = X(x) + [1 - J(z)]N_Q(z) \tag{4.3}$$

The filter J(z) can simply consist of unit delays, the expression $1 - J(z)$ then becomes $(1 - z^{-1})^n$. The input-output relation becomes:

$$Y(z) = X(z) + [1 - z^{-1}]^n N_Q(z) \tag{4.4}$$

where n is the order of the filter. The input-output relation in the frequency domain is obtained by a domain transformation, substituting z by $e^{-j\omega T_s}$:

$$Y(\omega) = X(\omega) + (1 - e^{-j\omega T_s})N_Q(\omega) \tag{4.5}$$

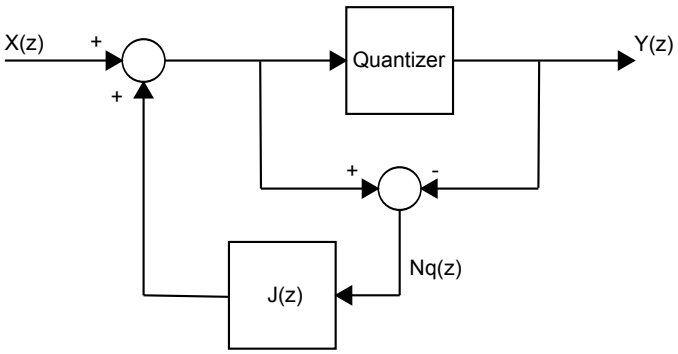

Figure 4.2: Block scheme of a noise shaper

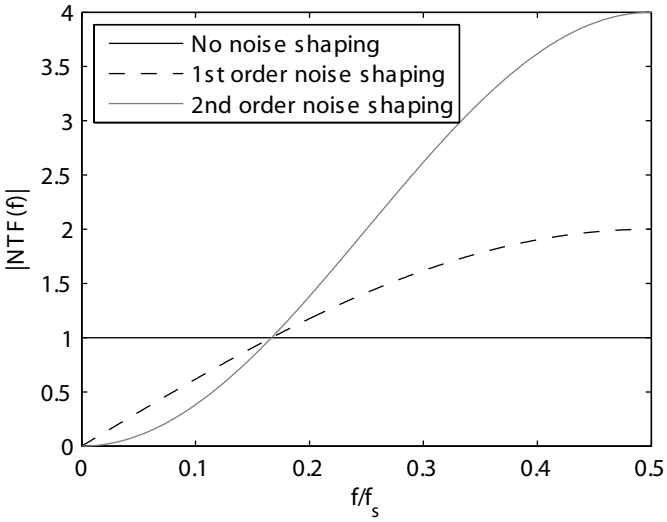

Figure 4.3: Ideal noise transfer function

And the Noise Transfer Function (NTF):

$$|NTF(\omega)|^2 = \left|\frac{Y(\omega)}{N_Q(\omega)}\right|^2 = |1 - e^{-j\omega T_s}|^{2n} \tag{4.6}$$

Using the NTF, the noise power on every frequency can be determined. The noise at f_b is of interest, as this will determine the performance of the SA. The NTF is plotted for filters of orders 1 and 2 in figure 4.3. It can be seen that noise shaping starts to become effective at an OSR larger than approximately 3 (note that f/fs=0.5 corresponds to OSR=1). At lower OSR a converter with uniform noise performs better.

4.3 Sigma Delta

The basic block scheme of a sigma delta modulator is shown in figure (4.4). The operation principle of a Sigma Delta converter is to feed back the quantization

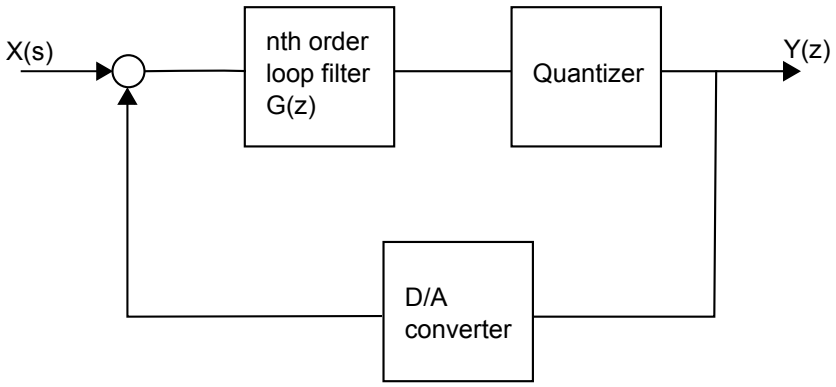

Figure 4.4: Block scheme of a sigma delta modulator

error to the input and correct for it. In contrast to the noise shaper in the previous section, the loop-filter is in the forward-path, and the feedback path contains both the signal and the error. There are three significant design parameters when designing a sigma delta converter; the number of quantization levels, the oversampling rate and the order of the loop filter. They will be discussed in the next three paragraphs.

4.3.1 Number of quantization levels

The number of quantization levels of the quantizer has significant impact on the performance. In general, increasing one bit results in about 6dB increase in SNR. Increasing the number of bits from 1 to 2 bit even results in an increase in SNR of approximately 7dB [7]. Although the performance benefits seem significant, the main drawback is linearity of the feedback DAC. Single bit designs have intrinsic linearity. Dynamic Element Matching (DEM) techniques can be used to increase the linearity of multi-bit architectures by switching between the DAC elements following certain algorithms [10]. Although this resolves much of the linearity issues, a DAC with DEM has a more complex design than a regular DAC.

4.3.2 Sample rate

Because of the noise shaping, the relation between OSR and SNR is non-linear. Doubling the OSR results in an increase in SNR of approximately $(2n+1)\times 3dB$, where n is the oversampling rate. However, high OSR results in relative high power consumption, so a low OSR is desirable. Especially for bandwidths as high as 20MHz, the clock frequency can get in the GHz range. Besides power consumption, bandwidth of the loop filter and other components also limit the maximal OSR.

4.3.3 Order of the loop filter

Increasing the order of the loop filter results in more aggressive noise shaping. More noise is pushed to higher frequencies, resulting in less noise in the band of interest. A major downside of higher orders is stability. In order to

prevent instability, less aggressive filter coefficients can to be used, resulting in lower performance than what might be expected. Adding extra orders to the filter also increases the complexity of the design, as an additional integrator is required for each additional order. Still, in combination with increasing the OSR, significant performance benefits can be achieved.

4.4 Design and modeling

In order to meet the requirements from section 2.3.2, design choices need to be made. The SA using XF-Correlation can benefit from a 1 bit architecture, therefore a resolution of 1 bit is the most attractive choice. Since linearity is important for the SA, the intrinsic linearity of a 1 bit DAC is also an advantage. By choosing 1 bit the use of often complex DEM techniques can be avoided. The main disadvantage of a 1 bit architecture is that it requires a higher oversampling rate or high order loop filter compared to a multi-bit converter in order to achieve similar performance. Based on a resolution of 1 bit, the order of the loop filter and the oversampling rate can be determined.

In order to get insight in the performance, the maximal noise level at f_b for a 1st, 2nd and 3rd order 1 bit sigma delta is plotted against the OSR rate in figure 4.5. The figure shows both the calculated performance based on equation 4.6 and the performance found by simulations. For the simulations a Matlab model is used. In order to simplify the simulations a time discrete model is used. Although this is an idealized model, it can be used to get an initial impression on the noise and distortion. The figure shows the "equivalent" SNR. This is not a real SNR figure, but the SNR based on the noise power at f_b. The SNR improvement as a result of noise shaping at frequencies below f_b is not considered, as this improvement in SNR does not improve the detection range of the SA. From figure 4.5 it can be determined that a first order sigma delta requires an OSR of around 12 and a second order sigma delta requires an OSR of about 10 in order to achieve an SNR of 25dB (requirement from table 2.1).

In the figure it can be seen that increasing the order can reduce the noise significantly. However, this is at the cost of additional complexity of the design and can also cause stability issues. The stability issues are further analyzed in the sections below. The improvement of SNR with each order depends on the OSR, and increases when the OSR is higher. However, high oversampling rates can result in other issues such as limited integrator bandwidth. This will be examined in more detail in chapter 5.

4.5 Linearity

Most literature on the performance of Sigma Delta architectures is about the SNR. The SFDR is usually left out from analysis, as the random, non-linear nature of the architecture makes both analytical approach as well as simulations very hard. Small changes in the conditions can have large and unpredictable effects on the output. The model can perform very well under certain specific conditions, but can perform bad when the conditions slightly change. To get an impression on the over-all linearity thorough simulations are required. Still this will only give insight in the linearity of the ideal sigma delta converter.

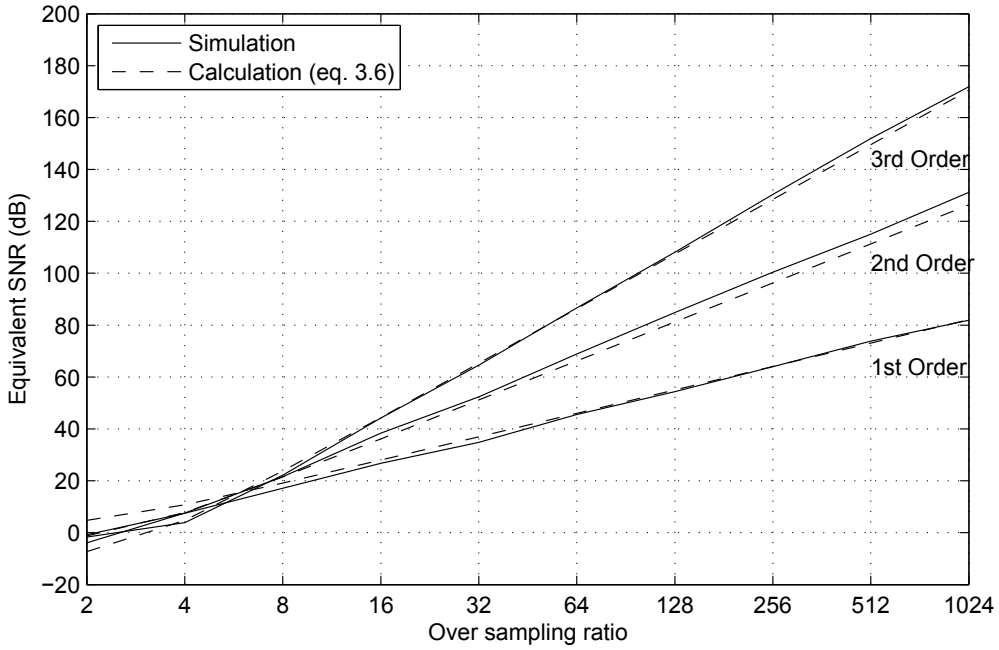

Figure 4.5: Noise spectrum of shaped quantization noise

In an actual implementation on a chip, unpredictable effects can occur that did not show up during simulations. According to [11] the only way to obtain realistic figures for the SFDR is to measure them from the actual circuit on chip. For an ideal Sigma Delta converter, there are two important effects that can cause harmonic distortion. These are pattern noise and instability.

4.6 Pattern noise

Pattern noise in sigma delta converters can occur when there is a repetitive input. This causes unwanted tones on the output. The most common input to cause pattern noise is a DC input. This type of pattern noise is also referred to as idle tones. The occurrence of these patterns is not always predictable due to random components, and as a result it is not possible to analyze every possible situation where these tones may occur. Pattern noise can be prevented using dithering. Dithering can be achieved by adding (pseudo) random noise to the least significant bit of the quantizer, which in case of a 1 bit converter is a comparator. With a properly chosen random bit sequence, the pattern noise can be eliminated completely according to [12]. The order of the loop filter influences the probability that a tone occurs. Higher order architectures have a lower probability for idle tones to occur, because higher order loop filters have a more aggressive filter, resulting in a more random output signal.

In an SA the most obvious situation where idle tones can occur is when the complete spectrum is empty, which is in effect a DC input. One can argue

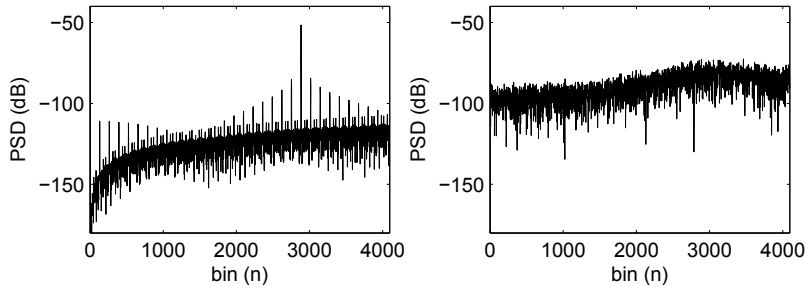

Figure 4.6: Idle tones in a 1st order Sigma Delta converter with a -39dB DC input before and after dithering

that in this case the occurrence of idle tones does not matter that much, as any channel is available. In that case it does not matter when some channels are falsely detected as occupied due to idle tones, the system can simply switch to a channel that is not affected by idle tones and is free. On the other hand, pattern noise can also occur with non-DC inputs. A property of the SA that makes pattern noise less of a problem is the measurement time. Since the spectrum is measured over a longer period of time, pattern noise that only occurs on a certain frequency for a very short time is still largely filtered out as a result of cross-correlation.

In order to examine the influence of noise on the pattern noise, simulations in Matlab are performed. In figure 4.6 (left) the spectrum of an ideal 1st order Sigma Delta converter is shown. Spurs as a result of patterns are clearly visible. Internal components of the ADC will also contribute to noise, however this noise may not be enough to reduce the pattern noise to acceptable levels. Figure 4.6 shows a simulation including noise on the input. As can be seen the spurs are now gone at the cost of a lower SNR.

In order to find an optimal value for the input noise, a sweep of the input noise is made using ideal Sigma Delta models in Matlab. As in section 3.6 the window size is chosen in such a way that the NIF in the FFT is similar to that of the SA system. At each input noise level, the input DC level is also swept, and the maximal distortion component is determined. Figure 4.7 shows this maximal distortion component as function of the input noise. In order to meet the requirement of an SFDR of 70dB, the distortion peak must be at least 70dB below the maximal input signal. A first order sigma delta meets this requirement at an input noise of around -40dB compared to maximal input. As predicted, a second order sigma delta converter is less affected by pattern noise and shows much weaker distortion components. Even at an input noise of -80dBFs the distortion is well within requirements. Depending on the design, the thermal noise of internal components can already be sufficient to reduce idle tones in a 2nd order sigma delta converter to an acceptable level. At an input noise of about 25dBFs the input noise starts to dominate the FFT result, so the values between 25dBFs and 0dBFs are due to input noise and not quantization noise or idle tones.

Numerical analysis of pattern noise, such as in [13] and [14], show similar

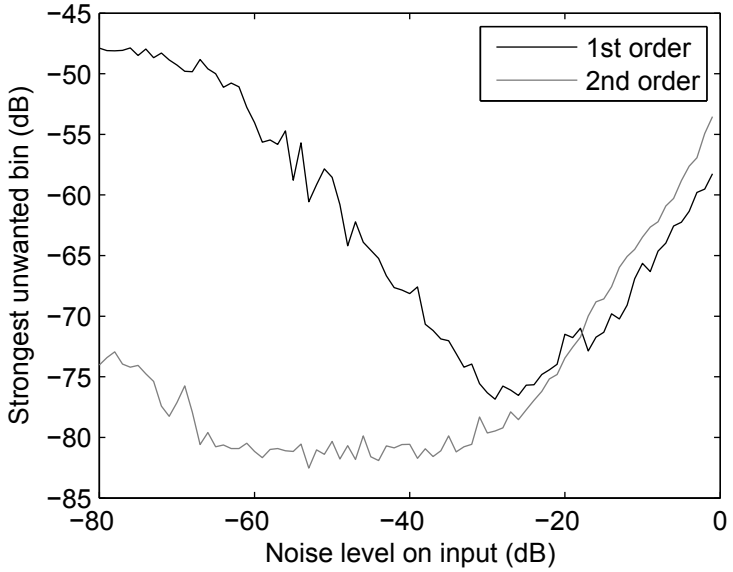

Figure 4.7: Effect of noise on distortion components caused by idle tones

results. These studies show that adding a RMS noise of 16uV (-96dBFs) reduces the number of identified patterns in the signal band from 108 to 12 for a 128 OSR second order converter. When only taking pattern noise in consideration a higher order architecture is preferable over a 1st order architecture. However, even with a first order architecture the pattern noise can be reduced to acceptable levels without too much effort.

4.7 Instability

Instability occurs when the buffers of a sigma delta are overloaded. This happens with large input amplitudes. Once a buffer is overloaded, it can take some time for the buffers to reach acceptable levels again. Overloading results in clipping, causing harmonic distortion on the output, reducing the SFDR of a the converter. In practice, the maximal value of a buffer is limited by external factors, such as the supply voltage. Limiting the input level also prevents overloading from occurring. The level by which the input is limited is called the overload level (OL), which is about -2dB to -4dB below the full scale amplitude, depending on the order of the loop filter and the oversampling rate. Higher order converters require a lower overload level. The overload level is defined as the highest input level where the SNR is 6dB below the maximal SNR level. This means that the converter still suffers from overloading at high input levels, meaning distortion can still occur. The frequency of the input signal also influences the overloading.

In order to determine the maximal input level where all distortion components are below 70dB, both a frequency sweep and amplitude sweep is performed for a 1st and 2nd order 1 bit sigma delta converter at 32 times oversampling. The resulting signal to largest unwanted bin ratio (in the figure called

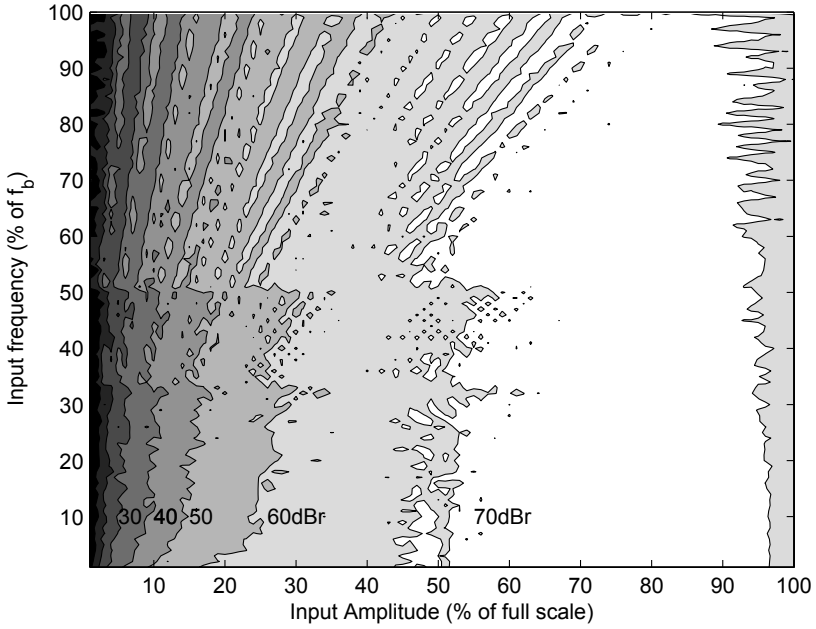

Figure 4.8: SFDR as a result of amplitude and frequency sweep for a 1st order SD converter

dBr) is plotted in figure 4.8. The third harmonic is the first strong one, and it is outside the baseband when the input frequency is at 1/3 of the base band frequency. At 1/3 of the baseband frequency itself, the harmonic distortion is the highest, as can be seen in figure 4.9. From the simulations results (figure 4.8 and 4.9) it can be determined that in order to achieve 70dB SFDR the input level of a 1st order converter has to be 0.9 (-0.5dB) and of a second order 0.15 (-8.2dB). This results in a direct impact on the performance of the converters. The 2nd order converter performs a lot worse than expected based on these simulations. A 7.7dB performance hit compared to a first order sigma delta makes the benefits of a second order converter small. Higher order converters even have larger performance hits from instability issues.

4.8 Conclusion

Because of inherent linearity, simple DAC design and benefits in the SA when using a XF-Correlator, a 1 bit resolution is the most interesting implementation from a theoretical point of view. Choosing the order and sample frequency is a less trivial choice. High sample frequencies result in higher power consumption, while higher filter orders result in more complex designs and stability issues. Considering an SNR of at least 25dB (section 2.3.2), the minimal required OSR can be found in figure 4.5. For a 1st order sigma delta converter an OSR of approximately 12 and for 2nd and 3rd order around 10 is required.

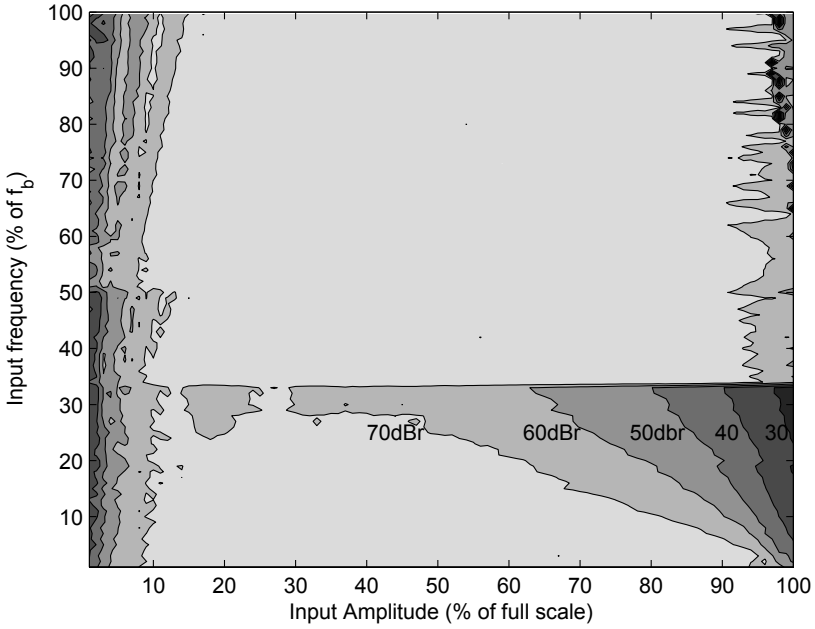

Figure 4.9: SFDR as a result of amplitude and frequency sweep for a 2nd order SD converter

Higher order architectures suffer from stability issues, causing distortion components and spurious tones. Simulations show that this distortion has a large impact on the SFDR, so the benefit of higher order converters is a lot lower than expected, in some situations performance is even worse. A first order converter does not suffer from instability issues and has a less complex design, and is therefore the preferable choice for the SA. A drawback of the first order sigma delta is that it suffers more from distortion as a result of limit cycles than a sigma delta with a higher order loop filter. However, this distortion can largely be prevented by dithering.

The sample rate depends on the desired measurement time of the SA and the power consumption. The lower limit of an OSR of 12 will result in a measurement time of around 1 second. Increasing the OSR will result in a higher power consumption. The required sample frequency for a converter with a bandwidth of 20MHz is high compared to existing CMOS sigma delta architectures. The highest speed 1 bit CMOS sigma delta at the time of this writing is [15], which reaches a bandwidth of 10MHz at OSR=32 and SFDR=72dB. In the next chapter different implementations of the sigma delta are considered, and a conclusion on the feasibility of a sigma delta converter is made.

Chapter 5

Analysis of ADC Architectures for spectrum analyzers

In chapter 3 and 4 both the Nyquis rate converter and the oversampling converter were examined in an idealized abstract model. Although some non-idealities were considered, the implications that come with actual implementations were left out. In this chapter the non-idealities that are encountered during design are analyzed, with a special focus on linearity. The goal of this chapter is the determine which architecture is best suited for the SA, and find out which non-idealities are important in the design.

5.1 ADC architecture types

At its core an ADC consists of a sample and hold circuit to sample the analog input signal, and a quantizer to digitize the sampled input signal. These are the basic building components found in any ADC. However, there are different methods to do the conversion, each has its specific advantages and disadvantages.

The sigma delta converter can be divided in two main categories, time discrete or time continuous. As the name suggests, the time discrete implementation directly samples the input signal, and does all processing in discrete time, while in the time continuous implementation sampling happens after processing the signal.

For Nyquist converters there are globally 3 different categories; the "parallel search" converter, the "sequential search" converter and the "linear search" converters. The "parallel search" converter does the complete conversion in one step. Every reference level is available at any moment. In theory this type of converter does not need to sample the signal. Main advantages of this type of converter is speed, main disadvantage is accuracy. Parallel converters include flash converters and folding converters.

Another category of Nyquist converters is the "sequential search" converter. With this type of converter the conversion takes place in several steps, where in each step a specific set of reference levels is chosen. In each step the result becomes more accurate. Each step can either have its own hardware, or the same hardware can be reused each step. This has impact on both the speed and the accuracy. In general this type of converter is able to achieve relative high

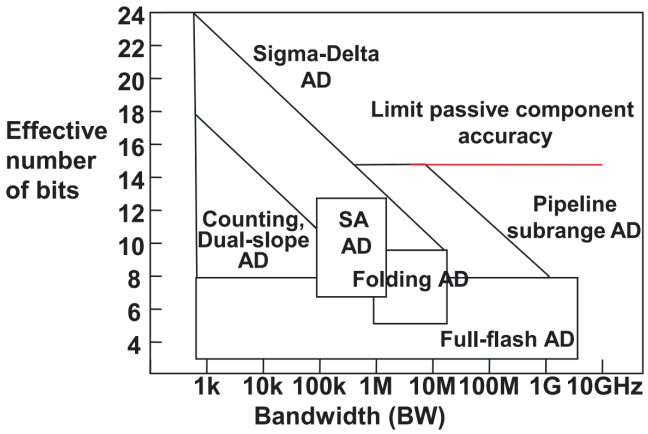

Figure 5.1: Performance overview of common ADC architectures [7]

speeds with high accuracy, which makes them suited for communication applications. The subset of converters include successive approximation conversion, pipelined conversion and multi-step conversion.

The last category of Nyquist converters are "linear search" converters. In this type of converter all possible conversion levels are compared to the input signal one by one. This results in very slow conversion but with a high accuracy. This type of converter however is not suited for speeds in the range that are needed for the SA in this work, so will not be further examined.

Figure 5.1 shows an overview of the basic types of ADC architectures and their approximate region of operation. The overview is taken from [7], written in 2010, but there is active development in ADCs and bandwidth and number of effective bits is constantly increasing as the technology improves. In the rest of this chapter the different implementations will be further analyzed for use in an SA. Properties that are important are the linearity and the input level. The ADC in the SA should be able to handle a low input level (around $63mV_{pp}$) and have high linearity (at least 70dB SFDR), while noise requirements are relaxed (at least 25dB SNR), see section 2.3.2.

5.2 Sample and Hold

In order to convert a time continuous signal to a time discrete signal, the signal must be sampled. Sampling means that between a certain time interval the value of the signal is determined. In order to further process the sampled signal, the value must be stored somehow. This can be done by a Track and Hold (TH) circuit. The most basic design of a TH circuit is a switch and a capacitor, see figure 5.2. During the tracking phase, the switch is on, and the capacitor follows the input voltage. When the switch is turned off, the capacitor will hold the input value for the duration of the hold phase. After that the switch is turned on again and the track phase is initiated again. When two TH circuits are placed in cascade, the result will be a Sample and Hold circuit. The difference is that the sampled value is available for the full period,

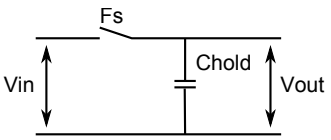

Figure 5.2: Basic sample and hold cicuit

so there is more time to process the value. This is at the cost of a delay of one cycle, although for an SA this is not an issue.

When designing a sample an hold circuit, there is a trade off between the distortion and the noise. In a typical ADC, the noise and distortion is chosen in such a way that the signal to noise and distortion ratio (SNDR) is optimal. In the SA the linearity is more important, so a different approach is required. The performance of a SH circuit is determined by both the switch and the capacitor.

In analog CMOS designs the capacitors are fabricated in "poly-diffusion", "poly-poly" or "metal-poly" structures. The poly-diffusion capacitor consists of an oxide layer between the substrate and a poly layer. The poly-diffusion capacitor is the most common, but suffers from non-linearity as the width of the depletion regions changes as the applied voltage changes. The non-linearity of the capacitance can be expressed by $C \approx C_0(1 + \alpha_1 V + \alpha_2 V^2)$ with α_1 and α_2 in the order of $5 \times 10^{-4} V^{-1}$ and $5 \times 10^{-5} V^{-2}$ [16]. The poly-poly capacitor is often used in double poly processes, and has about the same linearity as a poly-diffusion capacitor. The metal poly-capacitor consists of a poly-oxide-metal combination that is placed on a SiO_2 layer after the transistors are formed. As a result, no depletion layer is formed, and the linearity of the capacitor is greatly improved. Non-linearity coefficients in the order of a few parts per million can be achieved, so effectively the non-linearity of this type of capacitor can be neglected. A disadvantage of this type of capacitor is that the used process has to support it.

The switch consists of a MOS device with the gate connected to a clock. In the MOS switch the most important property for the linearity is the resistance R_{on}, which is dependent on the input voltage. The relation between the input voltage and R_{on} is given by equation (5.1) for an NMOS and by (5.2) for an PMOS [16].

$$R_{on,NMOS} = [\mu_n C_{ox}(W/L)V_{DD} - V_{in} - V_{THN}]^{-1} \tag{5.1}$$

$$R_{on,PMOS} = [\mu_p C_{ox}(W/L)V_{in} - V_{THP}]^{-1} \tag{5.2}$$

Figure 5.3 shows the on-resistance characteristic of an NMOS transistor. When the switch is on the equivalent circuit is an RC circuit with R dependent on the input signal. The variations in RC time result in distortion components as these variations are dependent on the input signal. As can be seen in figure 5.3, R_{on} can cause major problems with high input amplitudes. When the maximal input amplitude is small, the transistor can be configured in such a way that R_{on} variations are limited. When large voltage swings are required, a complementary switch can be used. This type of switch consists of both an

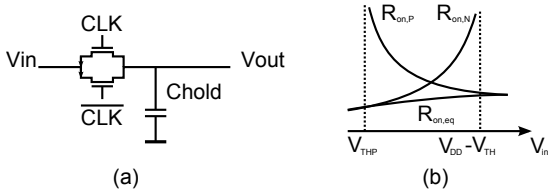

Figure 5.3: (a) Sample and Hold circuit with complementary switch (b) On resistance of the NMOS, PMOS and the complementary of the two.

NMOS and PMOS. The R_{on} of an NMOS increases with increasing input voltage, while R_{on} of a PMOS decreases. The resulting characteristic is an almost flat resistance characteristic for a full voltage swing. A method to further improve the performance of is to make use of bootstrapping [7]. Bootstrapping is a method to drive the gate voltage of the switch with voltages that are above the power supply voltage. This decreases the switch resistance variation and increases the maximal input voltage swing.

Considering a sample and hold circuit for the ADC in the SA, the output voltage swing of the mixer is low. When the ADC architecture allows a low input voltage, no amplification is required. With low voltage swings the input voltage can be biased on a level where the on-resistance variation of the MOS is minimal. The need for a complementary switch can be avoided without great impact on linearity.

5.3 Comparator

Every type of ADC has at least one comparator. This section contains a short analysis of important non-idealities of a comparator in order to get insight in the requirements for a comparator used in the ADC for a spectrum analyzer.

A comparator usually consists of a differential pair with a high gain. Any difference is to be amplified to either maximal or minimal voltage, effectively quantizing the signal. The defining attributes of a comparator are its accuracy, bandwidth and power consumption. The bandwidth is limited by the slew-rate of the amplifier. Changes of the input signal that are faster than the comparator can follow result in a higher distortion. Another effect that increases distortion is when the transistors in the amplifier enter the saturation region as a result of large input signals. The time it takes to return the transistor to its linear region can exceed the available time, with the result that harmonics can occur. This can be avoided by having a low input voltage swing.

Comparators suffer from a couple of effects that causes errors in the conversion. Some important effects include offset, metastability and memory [7]. The effect of comparator offset depends on the type of converter. A parallel search converter uses a unique comparator for every conversion level, so comparator offset has a direct effect on the INL. In sequential search converters, comparator offset results in a DC shift of the entire output, so only has a negative effect on the range and not on the linearity.

Metastability occurs when the inputs of the comparator are close to identical. When the difference is too small, a decision point is not reached within the available time. This causes erroneous conversions, which can be expressed as

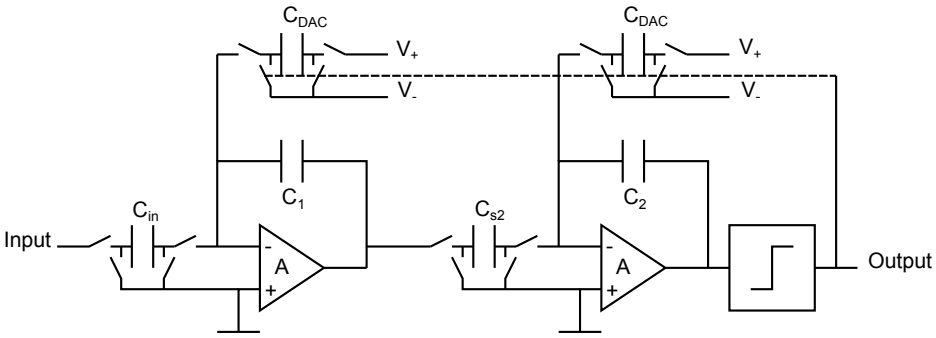

Figure 5.4: Discrete time sigma delta architecture

the Bit Error Rate of the converter (BER). Since these errors occur randomly it can be interpreted as added random noise. It does not cause harmonic distortion and is not very important for an SA, as noise can later be removed using cross-correlation.

To summarize, the most important non-idealities for a comparator used in a ADC in a spectrum analyzer are the amplifier slew-rate, the operation region of the transistors, and the offset.

5.4 Oversampling converters

As discussed in the previous chapter, the most interesting oversampling architecture is the Sigma Delta, which uses noise shaping to push noise outside the band of interest. Its error correction mechanism opens up the possibility of a 1 bit architecture, as the output becomes a pulse density modulated signal. There are two types of implementation for the sigma delta architecture; continuous time and discrete time. Both have their advantages and disadvantages. In order to get insight in which architecture would be preferable for an SA, both variants are examined further.

5.4.1 Discrete time

A discrete time sigma delta typically uses switched capacitors to do the processing in discrete time. Figure 5.4 shows a switched capacitor sigma delta implementation. The input signal is sampled at the input of the converter by the switch and capacitor, which make up a sample and hold circuit. Because the signal is sampled at the input, the loop filter will have no effect on filtering out high frequencies at the input. As a result the converter is sensitive to aliasing, and an anti aliasing filter is required. In contrast with Nyquist rate converters, filtering can, at least for a large part, be done in the digital domain. Since digital filters can be implemented more efficiently, this gives a benefit over Nyquist rate converters.

The operation principle of the switched capacitor sigma delta is as following: the sampled input and feedback value from the DAC are both stored in capacitors, after which these capacitors are discharged into the first integrator stage. As a result, the converter has a fixed settling time, determined by the

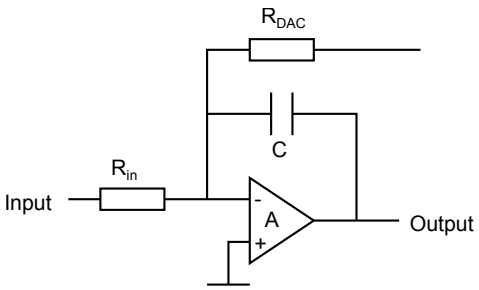

Figure 5.5: Continuous time sigma delta stage

time it take to discharge the capacitors. This settling time puts a limit on the maximal speed of the converter. The RC time as a result of the resistance of the switches and the size of the capacitors has influence on the settling time, but by choosing the right dimensions this doesn't offer a huge design problem. More significant is the amplifier gain of the integrator. When either the amplifier gain is too low or the sample frequency is too high, the result will be an incomplete discharge of the capacitors. This can result in harmonic distortion in the output signal. Other design properties that cause settling errors are the bandwidth of the amplifier and the slew rate. In CMOS technology the switches in figure 5.4 suffer from clock feedthrough at high frequencies. Asymmetrical edges are integrated, and show up as harmonic distortion. The rapid switching also causes large glitches on the virtual ground nodes.

Currently existing SC Sigma Delta converters are capable of reaching sample speeds of up to 420MHz. Typically the sample speed of high performance SC Sigma Delta converters in the range of 100MHz to 200MHz [17]. With a bandwidth requirement of 20MHz this means the OSR is limited to around 5. With such a low oversampling ratio, it is impossible to meet the system requirement of 70dB SFDR for the SA using a 1 bit resolution. Thus a multi-bit architecture is required. [18] and [19] are examples of SC sigma delta design achieving a bandwidth of 20MHz or more. [18] uses a combination of a 1.5 bit stage and a 4 bit stage in order to achieve high linearity. [19] uses a 13 level feedback stage. Interesting to note is that this converter is targeted at spectral sensing specifically. An SFDR between 62 and 79 is claimed, depending on the mode of operation. It uses DEM techniques in order to increase linearity in the feedback DAC. Key advantages of this converter are its flexibility.

The idea behind using a sigma delta architecture was its 1 bit output. When multi-bit is required, the complexity of the design increases. The inherent linearity of a 1 bit design is lost, and the non-linearity in the feedback DAC becomes significant. Techniques to prevent harmonics from element mismatch in the feedback DAC, such as Dynamic Element Matching, increase the complexity of the design. Since a multi-bit architecture is required to achieve the wanted performance for a discrete time implementation, the main benefit for cross-correlation is diminished.

5.4.2 Continuous time

A continuous time sigma delta has a similar design as a discrete time architecture. The integrator stage consists of an amplifier, capacitor and resistor. The voltage from the input and DAC are directly added and connected to the input of the integrator. Figure 5.5 show a basic continuous time sigma delta stage. In contrast to a discrete time sigma delta, in a continuous time Sigma Delta ADC the sampling happens after the integration stages. Because of this, aliasing is reduced significantly. The attenuation of unwanted aliasing effect is both determined by the order and cutoff frequency of the loop filter and the oversampling rate. Each order results in a 6db attenuation per octave. In combination with a high oversampling rate, the attenuation of aliasing effect can be significant, even for a first order filter.

Under similar conditions, the required anti-alias filter can be of much lower order than in the case of a discrete time sigma delta converter. In some cases a filter may not be required at all, which gives a continuous time implementation a major benefit over a discrete time implementation. This is also a major advantage compared to Nyquist converters, which also require an alias filter with high order. Another advantage of a continuous time architecture over discrete time is that the required bandwidth of the integrator is relaxed. According to literature, a CT modulator can be clocked up to an order of magnitude higher than a DT modulator [11].

The main cause for non-idealities of the continuous time sigma delta are the opamps in the integrators. These non-idealities include finite opamp gain, finite bandwidth, finite slew rate, a limited output swing and gain non-linearity. Finite opamp gain causes leaky integration with the result that the zeros in the NTF are moved toward z=0. This reduces the attenuation of the quantization noise in the baseband, which results in a lower SNR. A finite bandwidth affects the settling time of the converter. However, high performance can still be achieved with a high error in settling time, as long as the settling error is linear [11].

For the SA these are not significant problems as additional noise can be attenuated. The gain non-linearity however does cause problems. When the gain of the opamp is influenced in a non-linear manner by the input voltage, harmonic distortion which can not be attenuated can occur, especially with large input signals. Other non-idealities, such as clock jitter and component mismatch, are of less significance, as they cause white noise like distortion. Clock jitter causes small random variations of the charge that is fed back into the integrator. This can be seen as random phase modulation on the output bitstream, which causes noise to fold back into the baseband. Errors from process mismatch can influence the coefficients of the NTF, which results in more noise in the baseband. The effect of mismatch is very dependent on the specific circuit architecture, and cannot be generalized. Still, for example [20] found that errors up to 30% in the coefficients are tolerable, which suggests that mismatch is not a major issue. The noise caused by clock-jitter or mismatch can be attenuated using cross-correlation, and can therefore be considered less significant.

The highest speed CMOS sigma delta that was found in the "ADC Performance Survey" [17] at the time of this writing is a design by Crombez [15], which has a bandwidth of 10MHz. This still does not meet the 20MHz require-

ment which is the target in this work, so if a sigma delta converter is to be used in the SA, the bandwidth will be lower. An SFDR of 72dB is claimed, which would be enough for the requirements of the SA.

5.4.3 Conclusions

A sigma delta converter is a viable candidate for use in the SA. However, there are some significant issues which do not make them the preferable choice. First of all, the switched capacitor implementation requires a multi-bit architecture in order to achieve the required performance. This negates the benefit for choosing the sigma delta architecture in the first place, which was the 1 bit output. The continuous time architecture offers some benefits over a switched capacitor architecture. Firstly, it allows for higher sample rates. Still, the fastest 1 bit sigma delta architecture currently found in papers is only capable of a bandwidth of 10MHz, which does not meet the design requirements. Secondly, the continuous time sigma delta has inherent alias filtering which relaxes the requirements on an additional anti-alias filter. A disadvantage both continuous time and switch capacitor converter suffer from are idle tones and instability. As concluded in chapter 4 these effects make the sigma delta architecture less suited for an SA.

5.5 Nyquist Converters

5.5.1 Flash

The flash converter can either be used as a stand alone converter, or is used as conversion stage in other architectures, such as a subrange converter. The converter has an individual comparator for each conversion level. The complete conversion can be achieved in a single step. This allows this type of converter to achieve very high speeds, but the accuracy is limited by the comparator offset. This type of converter can typically achieve a maximal resolution of about 7 bit. An common implementation consists of a resistor ladder to bias the comparators. Figure 5.6 shows such a implementation. Because of process variation the resistor values will have small differences. This results in a random INL with no distinct shape, so no major distortion components are expected. Another cause of distortion in this type of converter is non-linearities of the sample and hold circuit, but the main bottleneck is the limited accuracy of the comparators as a result of offset.

In the SA, the voltage at the input of the ADC is low (about $62mV_{pp}$, section 2.3.2. In order to overcome the comparator off-setts, the input voltage needs to be amplified as much as possible. This is a major drawback of the flash converter, as amplification consumes both additional power, and adds additional distortion components. The main advantage of the full-flash converter is its high speed as the complete conversion is done in a single step.

5.5.2 Successive approximation

The principle behind Successive Approximation Register (SAR) ADCs is to systematically try different words and compare them with the input signal. The number of compare operations is equal to the number of bits. Since the

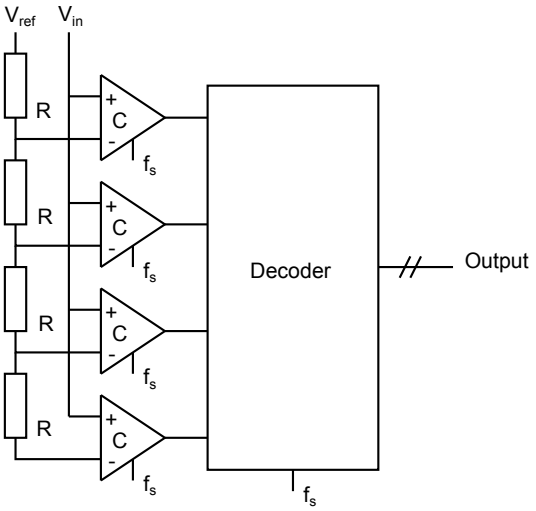

Figure 5.6: Full-flash ADC

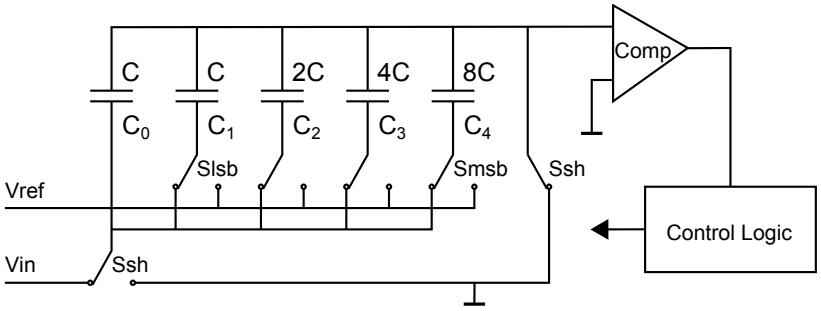

Figure 5.7: Successive approximation ADC

input is hold during this period, the comparators require to operate at a speed that is N*fs, where N is the number of bits and fs the sample frequency. This makes this type of ADC less suitable for high speed conversion. However, current day CMOS technology can offer high switching speed, making this architecture more attractive. Bandwidths up to 45MHz are achieved [17]. Liu [21] shows a design that is able to achieve 91.79 SFDR at a power consumption of 3mW with a bandwidth of 22.5MHz.

Figure 5.7 shows the block diagram of a SAR converter. The linearity of this type of converter is often limited by the distortion caused by the sample and hold circuit. Other causes for nonlinearity is mismatch of the capacitors in the capacitor bank and parasitic capacitances in the comparator. Offsets in the comparator result in a DC shift in the ADC conversion range, but the shift is equal for every conversion level, so no distortion components are added due to comparator offset. An advantage of this is that the input signal can be of low magnitude, at the cost of a lower SNR noise. For use in the SA, no additional amplification is required as long as the noise and the mismatch between the capacitors in the capacitor bank is not to high.

Charge redistribution

Charge redistribution SAR is a well known architecture in CMOS technology, and exploits the fact that good switches and capacitors are available in this technology. The basic principle is shown in figure 5.7. During sampling, the input voltage is stored in the capacitors C_1 to C_4. The additional C_0 is added to make the total capacitance 16C The capacitors are connected to the ground. When the sampling period is over, the conversion cycle is started. The switches are switched one at a time between a reference voltage and ground. When a switch is switched, the charge of the connected capacitor it redistributed over the other capacitors. This will cause a rise or drop in the voltage on the comparator input. When for example C4 is switched, the voltage on the comparator can be expressed by equation 5.3.

$$V_{comp} = -V_{in} + \frac{8C}{8C + 4C + 2C + C + C} V_{ref} = -V_{in} + \frac{8}{16} V_{ref} \tag{5.3}$$

This voltage corresponds exactly to a rise of 8 LSB. When the voltage is higher than the reference voltage of the comparator, the switch is turned off again and when it is lower it will stay on. Each capacitor is connected to the reference voltage successively. The final position of the switches determines the digital code that represents the input voltage in the digital domain. In figure 5.8 these conversion steps are illustrated.

A major advantage of this conversion method is that the total charge in the capacitors is conserved during one complete conversion, as it is only redistributed each conversion step. This means that during conversion no energy is consumed, except for the switching logic. Distortion components are caused by the switches, the comparator and capacitor mismatch. The major contributor to the thermal noise is the equivalent capacitance of the total capacitance bank.

5.5.3 Pipelined

Pipelined converters are a type of sub ranging converters that are popular because of their high speed. Figure 5.9 shows the block diagram of a pipelined converter. The converter consists of a number of equal stages. The principle behind the converter is to make a course estimation of the input in every stage and subtract this estimation from the input. The resulting signal is amplified and send to the next stage. This is repeated until the desired resolution is obtained. This principle is very similar to the SAR ADC, but the major difference is that each estimation stage has its own hardware implementation. Pipelined converters often make two level estimations, making the conversion perfectly linear, preventing distortion components from originating. However, other components in the converter can still cause non-linearity. The most significant components are the switches in the sample and hold circuitry and the amplifiers. Errors in every sequential stage are reduced by a factor two, making the requirements for later stages more relaxed. Random errors reduce by root-mean-square. This leads to expression 5.4, which can be used to determine the equivalent noise.

$$v_{in,noise} = \sqrt{v_{1,noise}^2 + 2^{-2} v_{2,noise}^2 + 2^{-4} v_{3,noise}^2 + ...} \tag{5.4}$$

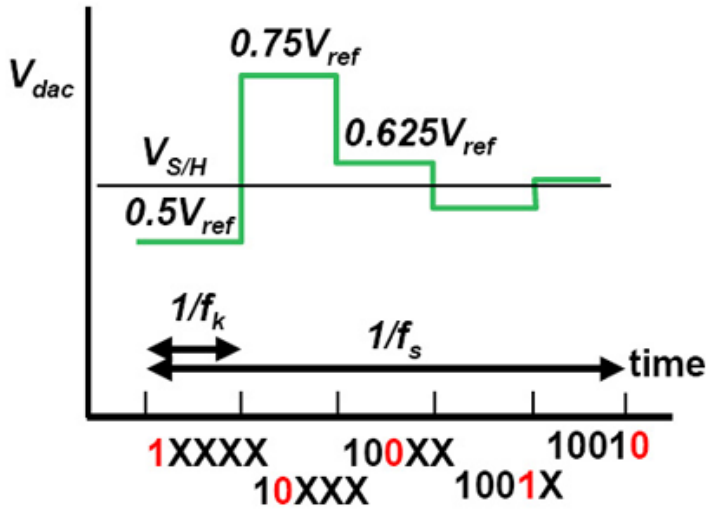

Figure 5.8: Successive approximation conversion steps [7]

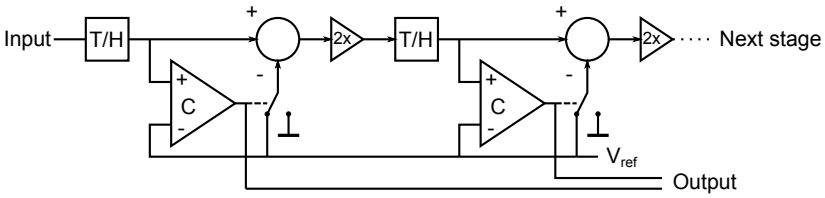

Figure 5.9: Two stages of a Pipline ADC

This type of converter is capable of handling a low input voltage swing, as long as the noise does not become too large. The main reason this architecture is interesting for use in the SA is because it is capable of reaching high sample rates, easily achieving the 20MHz bandwidth required for the SA. The main downside it that each stage consumes power, making it a less efficient architecture than the SAR ADC.

5.5.4 Conclusion

Nyquist converters are available in three main categories; flash, pipelined and successive approximation register. Flash requires a high voltage swing to overcome comparator offset, which requires the input to be amplified. Pipeline and SAR can both handle a low input voltage. The pipeline converter requires amplification in each stage, making it less efficient than a SAR converter. The dynamic power consumption of the SAR is further reduced by making use of charge redistribution. In section 5.4.3 it was concluded that although the 1 bit sigma delta architecture has some promising features, the limited bandwidth and unpredictable occurrence of tones make it less suitable for the SA. Overall, the SAR ADC has the best specifications for the SA. The only drawback is the

limited bandwidth as a result of multiple conversion steps. The next chapter is about the design of a SAR ADC for the SA.

Chapter 6

Implementation of a SAR-ADC for a spectrum analyzer

In this chapter a proposal for a design is made based on the findings in the previous chapters. An architecture is chosen and circuit level design choices are made in order to meet the requirements from table 2.1. The design choices are calculated and simulated where applicable.

6.1 Choice of architecture

Designing a complete ADC is a rather large undertaking, and a large part of this master thesis consists of system level analysis. So instead of designing a completely new ADC from scratch, an existing design is used as basis, the key issues are identified, and suggestions are made for improvement for the application in an SA. In chapter 3 it was found that in order to achieve an SFDR of at least -70dB, at least 7 bit is required. The other requirements of the ADC are found in section 2.3.2.

In the previous chapter, several ADC architectures were considered for a SA. With a 1 bit sigma delta architecture it was found that it is very hard to reach the required bandwidth of 20MHz. Unpredictable occurrence of tones also make the architecture less interesting. A flash converter may seem a viable candidate, as suggested by the diagram in figure 5.1. It is able to achieve high bandwidths at resolutions up to 7 bit. However, because of comparator offset, a full scale input is required. A Pipeline ADC can operate at both the resolution and bandwidth without problems, and it can handle low input swings. This type of converter also benefits from linear 1 bit conversion steps, resulting in relative low distortion. The main disadvantage is the required amplification after each stage, which both has a negative impact on linearity and power consumption. With continuously improved CMOS scaling, higher switching speed can be achieved. The higher switching speed makes higher bandwidths possible using the SAR architecture. Currently efficient SAR architectures with bandwidths up to 45MHz exist [17]. A major advantage of SAR is the very low power consumption as a result of charge redistribution. The architecture can handle low input amplitudes, so amplification may not be required, depending on the amount of noise.

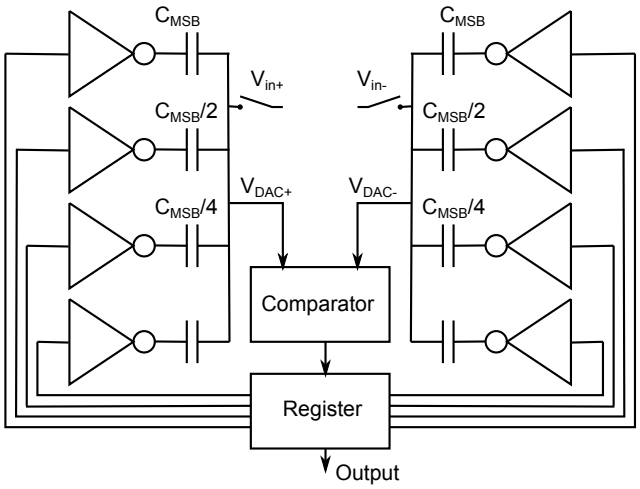

Figure 6.1: Simi-Differential charge redistribution ADC [22]

For the basis of the suggested design a 10-bit Charge-Redistribution ADC is chosen [22]. The reasons for this choice is the exceptional efficiency of the design and the availability of the schematics. The FOM of the converter is as low as 4.4 fJ/conversion-step, which is better than any published ADC with similar performance at the time of its publishing (2009). The ADC has been designed in an 65nm CMOS process, and test chips has been made for to perform measurements. The areas where the design does not meet the requirements for the SA is the bandwidth of 1MHz compared to 20MHz as required, and the linearity. Both issues are further examined, and suggestions for improvement are made.

6.1.1 Design

Figure 6.1 shows the schematic of the converter. The converter has a differential input and directly samples the input signal into the capacitor bank. This way an additional sample and hold circuit is not required. The register drives the inverters which act as the switches as seen in figure 5.7.

6.1.2 Resolution and bandwidth

The resolution of the reference design is 10 bit. For the SA a converter at least 7 bit is required. Reducing the number of bits is simply a matter of reducing the number of capacitors in the capacitor bank, and adjusting the control logic accordingly. The sizes of the capacitors in the capacitor bank needs to be recalculated to meet the requirements of the SA. Reducing the number of bits also reduces the conversion time, as instead of 11 only 8 conversion steps are made during one conversion (1 step is required for sampling). This is of benefit to the bandwidth of the converter.

In the current implementation, the bandwidth of the converter is mainly determined by the delay in the controller. The controller makes use of an internal delay line in order to progress through the conversion steps. An external

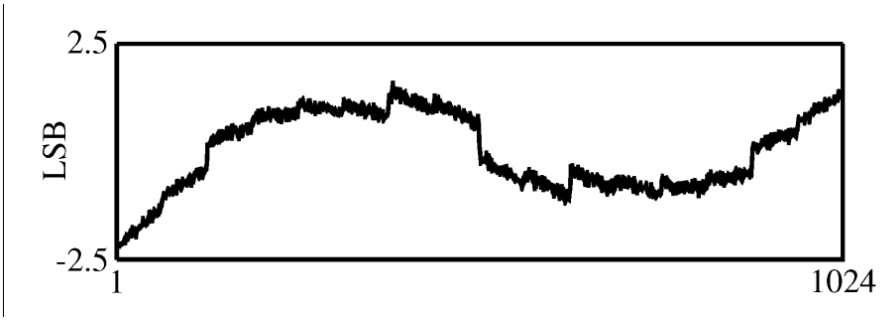

Figure 6.2: INL of the original ADC over the input range [22]

clock is therefore not needed. A disadvantage is that the delay interval of the subsequent steps is not very accurate as a result of process variation and other factors. The minimal time requirement for one step is dependent on the settling time it takes for the charge to redistribute and for the comparator to settle. Because of this process mismatch there needs to be a large margin of error in order to prevent that the next step is already initiated before the comparator is settled. The original design uses transistors with a high threshold voltage, and benefit from a high supply voltage. With a supply voltage of 1.3V, a speed of 4.9MS/s can be achieved. When the number of bits is reduced to 7 the bandwidth is increased by a factor of approximately 11/8=1.38 as a result of a reduction in conversion steps.

In order to achieve the required 20MS/s bandwidth, the control logic will need to be redesigned to handle such speeds. One way to achieve this is by making use of an external clock, or use a clock divider based on the sample clock. Since there are no major practical obstacles in achieving a higher speed for the control logic, no further investigation is done in the control logic and it is assumed that the control logic with a sufficient speed is achievable. According to [22] the used comparator is already capable of operating at the required speed. Other components that determine the bandwidth (mainly the RC time of the sample and hold and DAC) are examined in the next section.

6.1.3 Linearity

The linearity of the reference design is not sufficient for the SA. Figure 6.2 shows the INL of a test chip of the ADC over an input range of $2V_{pp}$. The INL has a maximal deviation of about 2LSB and a 3rd order shape distortion is recognizable. Using equation 3.26 from chapter 3, an $SFDR \approx 52dB$ is found, which does not meet the requirement of 70dB SFDR. In [22] a THD of 61.1dB is claimed for an input voltage swing of $1.25V_{pp}$. This suggests that lowering the input voltage swing considerably increases linearity. Important causes for non-linearities are the mismatch of the capacitors in the capacitor bank, the RC time of the sample and hold switch and the inverters in the DAC and nonlinear parasitic capacitances on node V_{DAC}. These causes for nonlinearity are analyzed in the following sections.

Mismatch

The mismatch of the capacitors in the capacitor bank result in binary-weighted artifacts. In figure 6.2 an example of these artifacts are visible in terms of INL. They appear as steps, where the most significant bit has the highest probability of having the largest step, which shows up in the middle. In the original design the value of C_{DAC} is determined by the mismatch. The value of C_{DAC} is chosen so that the average INL is not greater than 0.5 bit. For the SA this is not adequate. For the SFDR, the shape of the INL is relevant, not just the average. In chapter 2 it was found that in order to achieve 70dB SFDR the maximal allowed third order error is 0.031 [LSB].

Another factor that should be considered is that there are 2 separate ADCs to convert the two signal paths of the SA (section 2.1), each having their own set of capacitors with their own mismatch. This means that the resulting binary-weighted error will also be different between the two converters. Therefore the errors are not correlated and are averaged during cross-correlation as shown in chapter 3. The largest capacitor has the largest error in terms of LSB, so will have the most impact on the distortion. Since this error has a square wave shape of a single period, the largest distortion component is a 3rd order harmonic. The second capacitor will have a smaller error in terms of LSB, so will have a smaller contribution to the distortion.

Since the shape of the INL as result of the mismatch is a sum of all the random mismatch errors of the capacitors, the maximal allowed mismatch in C_{MSB} is hard to determine, although a good estimate to start with is the maximal allowed INL for a perfect 3rd order distortion. It is reasonable to assume that, although the shape of the distortion is different, the power of the third order component will be similar. The average of the two gaussian distributions of the capacitor banks in the two ADCs result in a lower over all error. This means the maximal allowed error in each ADC is $\alpha_{LSB} = \sqrt{2 \times 0.031^2} \approx 0.044[LSB]$. In order to achieve industry standard yield of 99.7 the 3σ design specification is used. Monte carlo simulations are performed to confirm if the desired yield is achieved. Only the capacitor mismatch is modeled, the rest of the ADC is ideal. Figure 6.3 shows the histogram after 10000 simulations. The SFDR does not drop below 70dB at all, which shows that higher mismatch is allowed.

Based on a mismatch of 0.044 LSB the size of the capacitors can be determined. The mismatch of a $C_0 = 1fF$ metal plate capacitor is $\alpha_0 = 0.9\%$ [22]. The maximal error in the largest capacitor is:

$$\alpha_{MSB} = \frac{\alpha_{LSB}}{2^{N-1}} = \frac{0.044}{64} \approx 6.88 \cdot 10^{-4} \tag{6.1}$$

with $M\sigma = 3\sigma$. The capacitance becomes:

$$C_{MSB} = \frac{C_0 M^2 \alpha_0^2}{\alpha_{MSB}^2} = \frac{1fF \cdot 3^2 \cdot 0.009^2}{(6.88 \cdot 10^-4)^2} \approx 1540fF \tag{6.2}$$

The required capacitance of 1540fF is larger than the original design. In order to find the maximal allowed mismatch while still achieving a 99.7% yield, Monte Carlo simulations are used to to determine the minimal sigma in LSB.

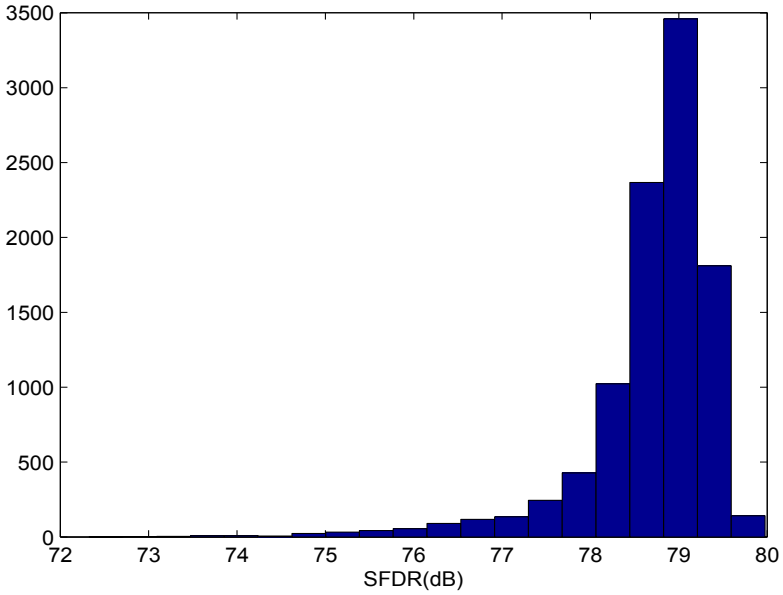

Figure 6.3: histogram of SFDR after 10000 simulations

It was found that a mismatch of $\alpha_{LSB} = 0.027[LSB]$ in the largest capacitor results in the desired yield. The capacitor size can again be calculated:

$$\alpha_{MSB} = \frac{\alpha_{LSB}}{2^{N-1}} = \frac{0.027}{64} \approx 4.22 \cdot 10^{-4} \tag{6.3}$$

Since the allowed mismatch already includes the 3 sigma standard, M = 1. C_{MSB} can be calculated:

$$C_{MSB} = \frac{C_0 M^2 \alpha_0^2}{\alpha_{MSB}^2} = \frac{1fF \cdot 1^2 \cdot 0.009^2}{(4.22 \cdot 10^-4)^2} \approx 454fF \tag{6.4}$$

The overall capacitance is still about two times larger than in the original design but offers no problems with implementation. An advantage of a larger capacitance is that the parasitic capacitances of the comparator have a smaller influence on the linearity.

RC time of the sample and hold switch and the inverters in the DAC

The RC time in the the sample and hold circuit causes harmonic distortion. In a differential design the third order harmonic is dominant. Using the maximal allowed error of 0.031 LSB found in chapter 3, the maximal R_{on} of the switch in the sample and hold circuit can be calculated. The maximal INL caused by the RC time is an exponential function described by:

$$V_{out} = V_{in}(1 - e^{-t/\tau}) \tag{6.5}$$

where $e^{-t/\tau}$ is the maximal allowed error in relative to full scale, so

$$e^{-t/\tau} = \frac{1}{2^N} INL_{max} = \frac{1}{2^7[LSB]} 0.031[LSB] \approx 2.42e-4 \tag{6.6}$$

From this equation, the minimal settling time can be determined:

$$t = -ln(2.42e-4)\tau \approx 8.3\tau \tag{6.7}$$

So in order to achieve an error of at most 0.031 LSB, the minimal required settling time of the sample and hold circuit is approximately 8.3RC. With $C \approx 1pF$, R_{on} can be determined:

$$R_{on} = \frac{t}{8.3C} \tag{6.8}$$

where t is the time available for sampling.

When sampling, the mixer is already offering a sampled signal at the frequency of the mixer (see section 2.1.1. When a band in the 2.5GHz region is mixed to baseband, the sampled output signal of the mixer is around 2.5Ghz. The mixer has a duty cycle of 25%, so 75% of one period is available for sampling. Adding a safety margin and some overhead, a settling time of at least $t = 1/5GHz = 200ps$ is available. This results in an R_{on} of:

$$R_{on} = \frac{200ps}{8.3 \cdot 1pF} \approx 24\Omega \tag{6.9}$$

In the used 65nm process the width of the switch can roughly be approximated by:

$$R_{on,nmos} \approx \frac{500}{W[\mu]}[\Omega] \approx 24\Omega \tag{6.10}$$

This results in a width of approximately $W = 21\mu m$.

For each conversion step, an inverter in the DAC (figure 6.1) changes its state. Similarly to sampling, the change in state requires a time to settle. Again the settling time is dependent on R_{on} of the transistors in the inverters. In this case R_{on} can be much higher, as the allowed settling time depends on the operation frequency of the ADC. With an operation frequency of 400MHz the settling time is about 3ns. This results in $R_{on} = \frac{3e-9s}{10.41e-12} \approx 288\Omega$. Using equation 6.10 a width is found of around $1.7\mu m$. From this analysis it can be concluded that he required transistor sizes of both the sample and hold circuit and the inverters in the DAC are realistically achievable in the 65nm process and do not offer any design problems.

Parasitic capacitances

In [22] it was suggested that most of the non-linearity was caused by parasitic capacitances on node V_{DAC} (figure 6.1). In order to decrease the effect of the parasitic capacitances, an alternative switching scheme has been proposed in [23]. Using this switching scheme, the non-linearities as a result of the parasitic capacitances cancel each other out. Since the voltage swing in the ADC for the SA is much lower than in the original design, linearity requirements may already be met. The nonlinear capacitance on the node V_{DAC} is caused by

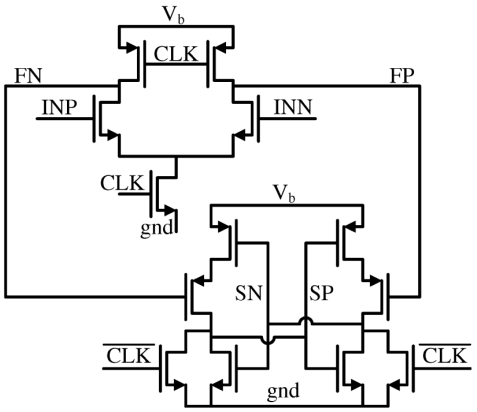

Figure 6.4: Two-stage comparator [22]

the input capacitance of the comparator. In [22] is was also suggested that parasitics between the capacitors and the substrate can be a cause of non-linear capacitances. These parasitics can be minimized by using a metal-insulator-metal type capacitor, were an additional metal plate is dividing the capacitor from the substrate.

The non linear capacitance on the input of the comparator can be determined by simulations. Figure 6.4 shows the layout of the comparator. In a test setup in Cadence a sweep of the input voltage of the comparator is made. The integral of the current that goes into the comparator is taken to determine the capacitance. The original design has an input range of $2V_{pp}$. The input capacitances found from simulation is shown in figure 6.5 (a). The variation in capacitance is up to 6.3fF. In the reference design, C_{LSB} is only about 0.5fF. The non-linearity in the INL (figure 6.2 (b)) can easily be accounted to the parasitic capacitance. Since the adjusted design only has an input voltage swing of $63mV_{pp}$, this voltage region is examined in more detail in figure 6.5 (b). It can be seen that the variation in $C_{parasitic}$ is about 142aF. If this variation is smaller than 0.031 LSB, it can be concluded that linearity requirements of the comparator are met without further analysis. With a C_{MSB} of 500fF, $C_{LSB} = 500/64 \approx 7.8fF$. This means that in terms of LSB the variation in $C_{parasitic}$ is $142aF/7.8fF \approx 0.018[LSB]$, which is well within the requirement of 0.031 [LSB].

A possible explanation for the shape of the non-linearity in figure 6.5 is the input resistance of the transistor. Since the values in the graph are an integral of the current, the current as a result of the input resistance of the MOS is also integrated. Even though this is a very small current, it can affect the results when there is a very small voltage change. The value found above includes this error, so the linearity is probably better than calculated there. Because the requirements are met even with this error, no further analysis is done. The conclusion from the simulations is that the linearity of the comparator is sufficient to meet the requirements for the SA.

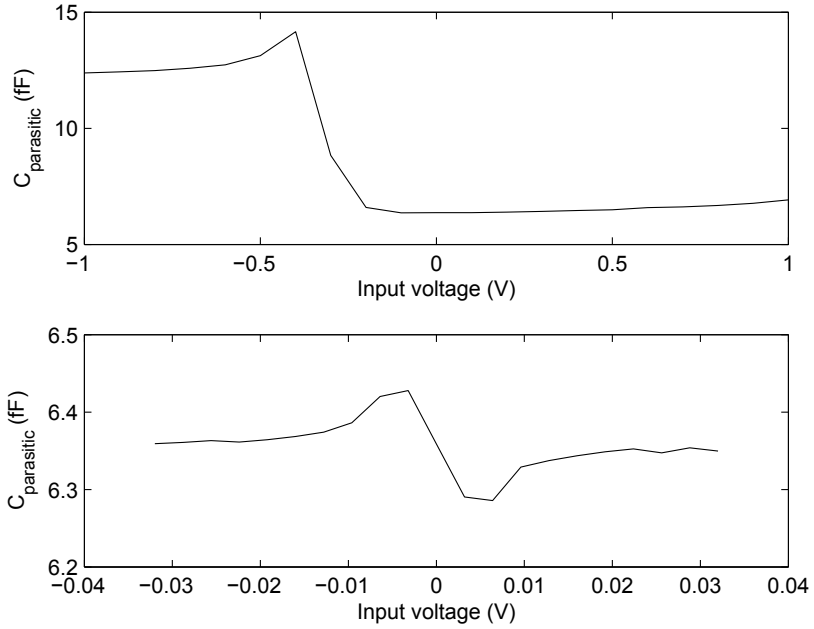

Figure 6.5: Parasitic input capacitance of the comparator

6.1.4 Noise

In section 3.5 it was found that a dithering noise level of approximately 48dB below the input signal is required to achieve an SFDR of 70dB for a 7 bit converter. This was however without taking in account the folding back of harmonics, which means the value should be a bit higher. In order to assure that all spurs are below 70dB, a value of -40dBFs is used. This equals to about 0.64 [LSB] for a 7 bit converter. With $v_{in,max} = 31.6mV$, the required RMS noise level is:

$$n_{RMS} = \frac{v_{in,max} \times 10^{40/20}}{\sqrt{2}} = 223\mu V \tag{6.11}$$

Capacitor bank

In order to achieve this kind of noise level, the capacitors in the capacitor bank could be chosen in such a way that the thermal noise on node V_{DAC} is equal to $223\mu V$ RMS. During the complete conversion process, there are two moments the noise will affect the result. The first is the moment the signal is sampled, the second is during comparing. Since the noise is independent, the variance can be added. In order to get a total standard deviation of $\sigma = 223\mu V$, the noise resulting from the capacitors needs to be:

$$\sigma_{total} = 223\mu V = \sqrt{2\sigma_C}[V] \tag{6.12}$$

so:

$$\sigma_C = 158\mu V \tag{6.13}$$

The noise on V_{DAC} as result of the capacitance is determined by equation:

$$\sigma_v = \sqrt{\frac{kT}{C_{DAC}}}[V] \tag{6.14}$$

where C_{DAC} is the total charge of all capacitors in the capacitor bank, k is the Boltzman constant and T is the temperature. C_{DAC} equals

$$C_{DAC} = \frac{kT}{\sigma_v^2} = \frac{1.38 \times 10^{-23} J/K^{-1} \times 300° K}{158\mu V} \approx 166fF \tag{6.15}$$

In the previous section it was found that in order to meet linearity requirements the minimal value for C_{MSB} is 454fF. The noise from the capacitors is lower than required for dithering, and additional noise has to be added.

Comparator

The equivalent input noise of the comparator was calculated in [22]. In the original design a noise level was of about $500\mu V$ was chosen in order to approximately match the quantization noise. This is larger than the optimal $223\mu V$ for dithering. A higher noise level results in a longer required time for cross-correlation in order to achieve similar performance. However, the noise level of 500mV in [22] is based on a noise integration time determined by a parasitic capacitance in the comparator. As noted in [24], the integration time of the comparator can be made longer, resulting in lower noise. The expression for the equivalent noise is [22]:

$$\sigma_v \approx \sqrt{\frac{kT}{C_{FP}}}\sqrt{8\frac{V_T}{V_{th}}} \tag{6.16}$$

where C_{FP} is the parasitic capacitance on node FP (figure 6.4), $V_T = kT/q$ and $V_{TH} = 0.4V$ [22]. With a requirement of $\sigma_v = 223V$, the wanted parasitic capacitance becomes:

$$C_{FP} \approx \frac{kT8\frac{V_T}{V_{th}}}{\sigma_v^2} \approx \frac{2.153e-21J}{223mV^2} \approx 43fF \tag{6.17}$$

In the original design the parasitic capacitance on node FP is a bit more than 10fF. The new value for C_{FP} can be achieved by changing dimensions of transistors. Further analysis is required to determine if the speed requirement is still met.

6.1.5 Anti-Alias filtering

In section 2.1.2 it was already suggested to use oversampling so that anti-alias filtering can for a large part be done in the digital domain. With a requirement of 70dB SFDR, any alias must be attenuated below this level when it folds back into the baseband. The amount of required oversampling can easily be calculated for each order, assuming a 6dB/octave attenuation for each order.

This means that for each time the sample frequency is doubled, the aliases are attenuated by 6dB for each order. When a first order anti-alias filter is used, the required oversampling rate is $OSR = 2^{70dB/6dB} \approx 3251$. With a second order filter, this is reduced to $OSR = 2^{70dB/12dB} \approx 57$ and with a third order filter $OSR = 2^{70db/18db} \approx 15$.

The limited settling time of the capacitor bank in the SAR ADC puts a limit on the sampling rate. A solution is to use time interleaving. The concept behind time interleaving is to use N ADC's in parallel to each other with a multiplexer to the input signal. The required area on chip of the resulting architecture is about N times the size of a single ADC. This puts a limit on the amount of ADC's that can by used in parallel while maintaining a reasonable chip size. The 2 capacitor banks in the SAR ADC are about 1pF in total each. The requirement when a 3rd order analog filter is used is 15 times oversampling. 15 of these ADC's in parallel results in a total capacitance of 30pF. The required area in this scenario is still reasonable. An additional benefit of oversampling is that the noise as result of quantization and dithering is spread over a larger spectrum, making the system rely less on cross-correlation to decrease the noise. The relation between SNR improvement and OSR is given by $\Delta SNR = 10log(OSR)$ [7], so when 16 times oversampling us used, the SNR improvement is 12dB.

A disadvantage of time interleaving is a fixed pattern noise as a result of different DC offsets between the converters. One of the benefits of a SAR converter is that its performance is little affected by comparator offset. When using a time-interleaved architecture, this benefit is lost, as each comparator has its own offset. There are 3 types of error that cause unwanted components in the output spectrum when time-interleaving is used; time errors (static jitter), amplitude offset errors and gain errors. These errors have impact on the linearity and may require a more accurate comparator in the ADC. Further study is required when time-interleaving is being used in the SA.

Even when oversampling is used, there still needs to be a filter. Since the mixer already has a time discrete output, a possible solution for filtering is a time discrete filter between the mixer and the ADC. Such filters are studied in more detail in for example [25]. The output of a filter like this is also time discrete. Further study is required in the implementation of this filter.

6.1.6 Interface with mixer

In section 2.1.1, figure 2.2 the full schematic of the mixer is shown. For the interface with the ADC, only one branch of the Quadruple Sampling Mixer (QSM) architecture needs to be analyzed, as the interface to all 4 branches are similar. The output of the mixer is a sampled signal for the duration of that the switch in the mixer is off. To directly reuse the sampled output of the mixer, the sample and hold switch of the ADC must run on a frequency that is synchronized with the frequency of the mixer. As found in section 6.1.3, the switch in the sample and hold can be dimensioned in such a way so that the linearity requirement is met at frequencies up to 2.5GHz, so synchronization is not a problem, although some additional logic is required to control the sample moment.

One issue with the Tayloe mixer is that the input impedance requires tuning [26]. In [26] a method is proposed to tune the impedance of the mixer so that

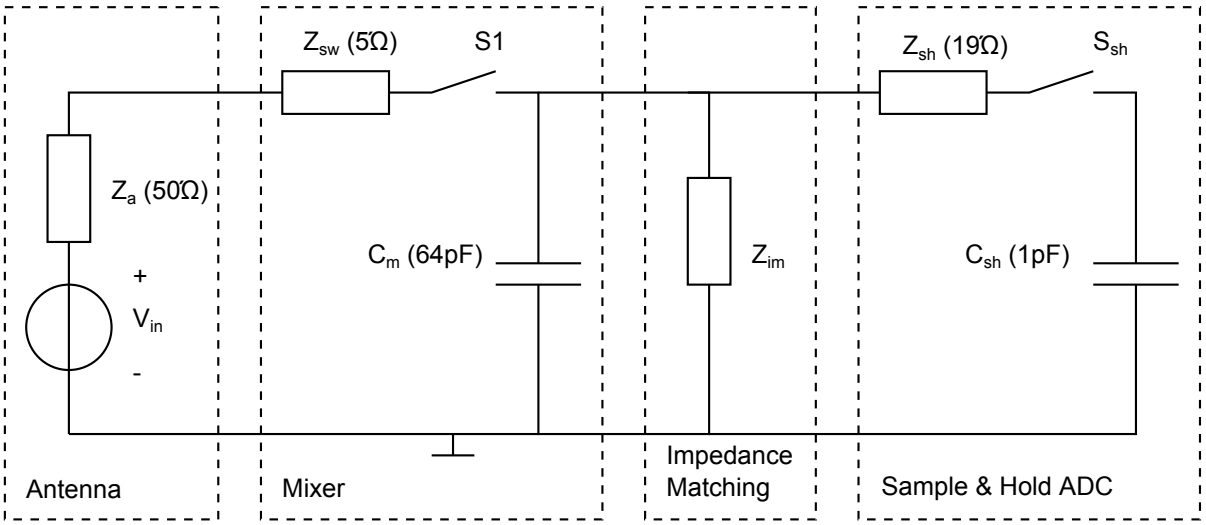

Figure 6.6: Interface between mixer and ADC

an impedance of 50Ω is presented to the antenna. By doing this, reflections are avoided and linearity is improved. This is achieved by adding a resistor Z_{im} between the output of the mixer and the ground. See figure 6.6 for the equivalent circuit. At the moment the mixer is sampling, which happens during 25% of the period ($D = 25\%$), the antenna is connected to the output through the switch. The rest of the time the switch is closed, and the input is disconnected from the output. During this time the capacitor discharges through the resistor. This means there is constantly a current flowing through Z_{im}. To determine the input impedance, it is assumed that the voltage drop over C can be ignored as C is relatively large. This simplifies the calculations. When the switch is closed the equivalent input impedance is $Z_{in,c} = Z_{sw} + Z_{im}$, and when the switch is open $Z_{in,o} = Z_{im}$. With a duty-cycle of 25% the resulting input impedance is:

$$Z_{in} \approx D \cdot Z_{in,c} + (1 - D) \cdot Z_{in,o} = 0.25 \cdot Z_{sw} + Z_{sh} \tag{6.18}$$

In case of a QSM architecture 4 of these impedances are in parallel, resulting in a total impedance equal to:

$$Z_{in} \approx \frac{0.25 \cdot Z_{sw} + Z_{sh}}{4} \tag{6.19}$$

resulting in $Z_{im} = 198.75\Omega$ for $Z_{in} = 50\Omega$. Simulations in LTSpice show an input impedance of $Z_{in} \approx 54\Omega$ with an $Z_{im} = 198.5\Omega$. The small difference can be accounted to ignoring the voltage drop over C and other non-idealities.

When the output of the mixer is sampled by the ADC, a part of the charge from the capacitor of the mixer is transferred to the capacitor bank in the ADC. The sample and hold circuit of the ADC can be interpreted as a switched capacitor resistor. The resistance is frequency dependent and can be expressed by $R = 1/(Cf)$. The ADC runs at a sample frequency of 40MHz and has a hold capacity of $C \approx 1pF$. The resulting equivalent resistance is $R_{eq} =$

$1/(1pF40MHz) = 25K\Omega$. This resistance is rather insignificant compared to Z_{sh}, The total effect is a decrease in equivalent resistance of about 1.5Ω. In effect the ADC has no meaningful effect on the impedance matching of the mixer. However, when time-interleaving is used, the capacitance increases. With 15 converters in parallel, the overall capacitance is 30pF. The equivalent resistance is $R_{eq} = 1/(30pF40MHz) \approx 833K\Omega$. When an equivalent resistance of around 200Ω is wanted, the required total capacitance is 125pF.

In this section a direct connection between the mixer and ADC was described. However, an anti alias filter may be required. It is assumed that a time discrete solution is possible, as described in the previous section. When such a filter is added between the mixer and ADC, the same principles are still valid for matching and sampling.

6.1.7 Conclusion

The reference design does not meet the requirements for the SA. The bandwidth can be increased by improving the control logic, which needs to be redesigned. Reducing the resolution of the ADC from 10 to 7 bit increases the bandwidth by a factor of about 1.38. Mismatch of the capacitors in the capacitance bank result in a binary weighted shape in the INL. The main effect is a third order harmonic. Based on calculations and simulations, a capacitance of 1pF was found to be sufficient to meet the linearity requirements. The RC time of the sample and hold switch and the transistors in the inverters and the capacitance on the capacitor bank put a limit on the linearity. By choosing a small enough R_{on} the RC time can be made small enough to ensure the linearity requirements are met. The resulting transistor sizes are reasonable and do not offer any problems. The non-linearity as a result of parasitic capacitances is greatly reduced as a result of a lower input voltage swing. Simulations show that the non-linearity problems that were found in [3] are solved. Analysis shows that noise requirements are met, and additional noise is required for dithering in order to reduce the distortion from quantization. The interface between the mixer and ADC do not offer major problems. The ADC has little influence on impedance matching, and the sampled signal on the output of the mixer can directly be sampled by the ADC. Synchronizing the sampling required some additional digital hardware. For impedance matching, an additional resistance is required, the load of the ADC is too small.

Chapter 7

Summery and Conclusions

The goal of this master thesis was to examine which ADC implementation best meets the requirements for usage in an SA in a cognitive radio system, and propose a design based on these findings. The SA is designed to measure a wide spectrum at once and achieve a high linearity. Several converter types and architectures have been examined. The key requirements of the ADC are an SFDR of at least 70dB, 20MHz bandwidth, an SNR of 25MHz and a low input amplitude of about $V_{pp} = 63mV$. Special attention is given to 1 bit architectures as a 1 bit resolution can be of benefit to the digital implementation of cross-correlation.

7.1 Nyquist converters

In chapter 3 it was found that the minimal required resolution of a Nyquist converter that meets the requirements is 7 bit. Dithering is required to reduce the deterministic behavior of the quantization error so that it can be attenuated by cross-correlation. The amount of dithering required for a 7 bit converter is in the order of 0.5 to 1 LSB. In order to meet the linearity requirement and achieve an SFDR of 70dB, the maximal calculated 3rd order shaped INL is about 0.031 LSB, which is confirmed by simulations. Several architectures were considered. The architectures that are capable of speeds in the range of a 20MHz bandwidth are the flash converter, the pipelined converter and the SAR converter. The flash converter has a major disadvantage that it required the input voltage swing to be as large as possible, resulting in the requirement of an additional amplifier stage before the ADC. The pipelined converter requires amplification in each stage, reducing the efficiency and linearity of the overall design. The SAR was found to be the most efficient solution, as no additional amplification is required, and charge redistribution results in a very low power consumption.

7.2 Oversampling converters

The benefit of oversampling converters compared to Nyquist converters is that as a result of oversampling, the resolution of the converter can be reduced. Cross-correlation can benefit from a 1 bit input as multiplications become single logic gate operations. Oversampling can be achieved by increasing the

sample frequency of a Nyquist converter. A sigma delta is a special type of oversampling converter and is capable of pushing noise outside the band of interest by shaping the noise to higher frequencies. This results in significant better performance, making the sigma delta architecture preferable over other architectures. The amount of noise that is pushed outside the band of interest depends on the order of the noise shaping filter and the oversampling ratio. Both noise and linearity have been analyzed of the sigma delta architecture. The noise from a sigma delta converter is not uniform, but because of the random nature of the feed back circuit it is not deterministic, and can be attenuated by cross-correlation.

The main issue with sigma delta converters are idle tones and stability issues. Idle tones are caused by a repetitive input, and can be prevented by dithering. Instability issues occur mostly for higher order converters, and can for a large part be prevented by reducing the maximal swing of the input signal. This however also has an impact on the performance, making a higher order converter less attractive than it initially seems. A sigma delta converter has two types of implementations, discrete time and continuous time. A discrete time converter can achieve a lower bandwidth than a continuous time converter, and it requires a higher resolution than 1 bit in order to achieve the specification for the SA. An additional benefit of the continuous time converter is that the signal is sampled after the loop filter, making the loop filter itself work like an anti-alias filter. Even though higher bandwidths are achievable with a continuous time sigma delta, at the time of this writing the highest bandwidth achieved by a 1 bit continuous time sigma delta converter is around 10MHz according to available literature [17].

7.3 ADC implementation in a spectrum analyzer

The first choice to be made is between the Nyquist converter and the oversampling converter. Because of the tones and stability issues, and the limited bandwidth of 1 bit oversampling converters are not the best choice. Still, the architecture has potential if the issues with speed and harmonics are resolved. This is however not an easy task as the unpredictable behavior of the architecture makes simulation results unreliable and accurate information about its performance can only be acquired from measurements from a test chip. The limited bandwith of a 1 bit architecture can be resolved by increasing the resolution. Increasing the resolution decreases the benefit of a low resolution with cross-correlation. This effectively negates the main reason to use an oversampling converter in the first place.

All points considered makes a Nyquist converter the better choice, as it behaves more predictable and is capable of achieving the required bandwidth without much problems. The most efficient architecture is found to be the SAR converter, as its charge redistribution algorithm minimizes the power consumption. It does not require any amplification and analysis showed that there are no major issues that would prevent reaching the required linearity. Speed can be an issue, but SAR converters with bandwidths of up to 40MHz already exist. The ADC design proposed in this work is based on an existing ADC which is chosen for its very high efficiency. The lower input voltage swing solves a major non-linearity issue as result of parasitic capacitances that existed

in the original design. Other causes of non-linearities, such as mismatch of components and RC times of switches, are solved by adjusting the dimensions of the capacitors and transistor of the original design. The linearity requirement of SFDR=70dB is met. The resulting components sizes are still of very reasonable size, and power consumption is little affected.

7.4 Conclusions

It was estimated that the cross-correlation implementation is capable of reducing the noise floor by about 45dB. This results in a minimal required SNR of about 25dB for the ADC. Two types of ADCs were analyzed, the sigma delta and the Nyqist rate converter. Even though cross-correlation can benefit from a low resolution ADC, a 1 bit sigma delta ADC was not found to be very suitable for an SA because of unpredictable occurrence of tones and a limited bandwidth. A Nyquist rate converter requires at least 7 bit in order to achieve a SFDR¿70dB and a SNR¿25dB. The bandwidth requirements are easily met. Distortion components as a result of quantization can be reduced to acceptable levels by adding noise.

The proposed ADC architecture is a charge redistribution SAR ADC, based on [22]. In order to meet the linearity requirers of 70dB SFDR, the values of the capacitors and the widths of the switches and the transistors in the inverters were calculated. The resulting values are very reasonable and do not offer any problems. The linearity of the comparator was determined using simulations. It was found that the linearity requirement is met when the input voltage swing is around 64mV. Although a noise requirement of 25dB is met, less noise reduces the work for the cross-correlator. In order to reduce the noise of the comparator, the transistors have to be resized, at the cost of either a lower speed or higher power consumption.

The ADC proposed in this work meets most requirements but does require more work. Overall it looks like a very suitable solution for the ADC in a spectrum analyzer for mobile usage.

7.5 Future research

In this master thesis an implementation for the ADC in a SA have been presented. The main focus has been on the linearity, for which the requirements are met. Other aspects such as speed and noise have been given less attention, and require further research. The digital control logic needs to be redesigned in order to achieve the required clock speeds. Also further investigation in the speed of the comparator is required. Although no drastic changes were made to the original design, the effect of the changes on the power consumption has to yet to be determined. Because of the lower noise requirement of the comparator and the larger capacitors in the capacitor bank, it is expected that the power consumption will be higher. Additionally, power consumption of the new control logic is also expected to be higher.

Besides the ADC, further work has to be done in the implementation of the anti-alias filter. A solution suggested in this work is a discrete time filter in combination with oversampling. A discrete time filter may offer an interesting solution because on the output of the mixer there is already a discrete time

signal available. Time-interleaving increase the sample frequency and makes it possible to use the ADC architecture presented here with over sampling. A large part of the alias filtering can then be done in the digital domain. However, time-interleaving introduces new issues, such as comparator offsets between the ADCs, which required further investigation.

Appendix A

Appendix

A.1 Fourier transform definitions

A.1.1 Convolution

$$timedomain : (f * g)(x) \tag{A.1}$$
$$frequencydomain : F(\omega)G(\omega) \tag{A.2}$$

A.1.2 gaussian ditribution

$$f(x) = e^{-\frac{x^2}{2\sigma^2}} \tag{A.3}$$

$$F(f) = \int_{-\infty}^{\infty} e^{-\frac{x^2}{2\sigma^2}} e^{-2\pi i f x} \tag{A.4}$$

$$\frac{dF(f)}{df} = \int_{-\infty}^{\infty} e^{-\frac{x^2}{2\sigma^2}} (-2\pi i f) e^{-2\pi i f x} = -4\pi^2\sigma^2 f F(f) \tag{A.5}$$

$$F(f) = F(0)e^{-2\pi^2\sigma^2 f^2} \tag{A.6}$$

$$F(0) = \int_{-\infty}^{\infty} e^{-\frac{\pi x^2}{2\pi\sigma^2}} = \sqrt{2\pi\sigma^2} \tag{A.7}$$

$$F(f) = (\sqrt{2\pi\sigma^2})e^{-2\pi^2\sigma^2 f^2} \tag{A.8}$$

A.2 Bessel function

A.2.1 Describing the Fourier transfer of the quantization staircase using the Bessel function

This applies to a sinusoidal signal that is digitized (sampled and quantized)
A sinusoidal input is expressed by:

$$x(t) = A(t)sin\phi(t) \tag{A.9}$$

with $\phi(t) = 2\pi ft + \Gamma(t)$. The output can be expressed by the Fourier series:

$$y = Asin\phi + Im\left\{\sum_{n=1}^{\infty}\frac{1}{n\pi}e^{j2n\pi Asin\phi}\right\} \tag{A.10}$$

Making use of the fact that:

$$e^{jzsin\theta} = \sum_{p=\infty}^{\infty} J_p(z)e^{jp\theta} \tag{A.11}$$

and:

$$J_p(x) = \frac{1}{\pi}\int_0^{\pi}\cos(n\tau - x\sin\tau)\,\mathrm{d}\tau \tag{A.12}$$

equation A.10 becomes:

$$y = Asin\phi + Im\left\{\sum_{n=1}^{\infty}\sum_{p=-\infty}^{\infty}\frac{1}{n\pi}J_p(2n\pi A)e^{jp\phi}\right\} \tag{A.13}$$

where J_p is the Bessel function. The output can be expressed as:

$$y = \sum_{p=1}^{\infty} A_p sinp\phi \tag{A.14}$$

Where A_p is

$$A_p = \delta_{p1}A + \sum_{n=1}^{\infty}\frac{2}{n\pi}J_p(2n\pi A) \tag{A.15}$$

Where δ_{pq} is 0 for $p \neq q$ and 1 for $p = q$. In this expression the first term is the original sine and the second term is the error.

A.2.2 Approximation of the 3rd harmonic using Chebyshev transformation

from [9]

$$A_p = \delta_{p1}A + \sum_{n=1}^{\infty}\frac{2}{n\pi}J_p(2n\pi A) \tag{A.16}$$

For the lower harmonics in the region $p << 2n\pi A$, this equation can be approximated by:

$$A_p(A) = (-1)^{(p-1)/2}\frac{2}{p\pi}\int_{-\pi/2}^{\pi/2} y(A\cos\theta)cos\,p\theta\,d\theta \tag{A.17}$$

$A\cos\theta = x$ and integration by parts results in [1]:

$$A_p(A) = (-1)^{(p-1)/2}\frac{2}{p\pi}\sum_{k=-[A]}^{[A]} U_{p-1}\left(\frac{k}{A}\right)\left(1 - \frac{k^2}{A^2}\right)^{1/2} \tag{A.18}$$

where U_{p-1} is the Chebyshev polynomial of the second kind [9]. Since the third harmonic is the strongest it is only required to calculate A_3. For A_3 the Chebyshev polynomial is $U_2(z) = 4z^2 - 1$ (appendix A.3), so A_3 is equal to:

$$A_3(A) = -\frac{2}{p\pi} \sum_{k=-[A]}^{[A]} \left(4\left(\frac{k}{A}\right)^2 - 1\right)\left(1 - \frac{k^2}{A^2}\right)^{1/2} \tag{A.19}$$

A.2.3 Approximation using Airy function

$$A_p = \delta_{p1}A + \sum_{n=1}^{\infty} \frac{2}{n\pi} J_p(2n\pi A) \tag{A.20}$$

As suggested in [9], J_p can be approximated by an Airy function.

$$J_p(pz) = \left(\frac{4\zeta(z)}{1-z^2}\right)^{1/4} \frac{Ai(p^{\frac{2}{3}}\zeta)}{p^{\frac{1}{3}}} \tag{A.21}$$

with:

$$\zeta = -\left(\frac{3}{2}\sqrt{z_2 - 1} - \frac{3}{2} arccos\frac{1}{z}\right)^{2/3} \tag{A.22}$$

Because of the odd symmetry of the quantization staircase, only the odd harmonics need to be considered in order to find the strongest harmonic. Substituting J_p in A.20 with the Airy approximation results in:

$$A_p = \sum_{n=1}^{\infty} \frac{2}{n\pi} J_p(2n\pi A) = \sum_{n=1}^{\infty} \frac{2}{n\pi} \left(\frac{4\zeta(z)}{1-z^2}\right)^{1/4} \frac{Ai(p^{\frac{2}{3}}\zeta)}{p^{\frac{1}{3}}}, for\, p\, is\, odd \tag{A.23}$$

The strongest harmonic p is close to $2\pi A$ and $z = 2\pi mA/p$. For the case where m = 1, z is also close to 1. Numerical analysis [1] of $p^{\frac{2}{3}}\zeta$ shows the value to be around -1, making the Airy function $A(p^{\frac{2}{3}}\zeta)$ behaves much like a constant. In $4\zeta(z)/1 - z^2$ the value is determined by z so this factor can also be considered a constant. This results in the strongest harmonic being proportional to $A_p(m = 1) \propto \frac{1}{p^{\frac{1}{3}}}$. This is equivalent to a decrease of 2.01dB per bit. For $m > 1$ the function $A(x)$ can be approximated by:

$$Ai(-x) \approx \frac{sin(\frac{2}{3}x^{\frac{3}{2}} + \frac{\pi}{4})}{\sqrt{\pi}x^{\frac{1}{4}}} \tag{A.24}$$

Evaluation [1] of this equation shows that this only results in random variations that are added to the trend found for m=1.

A.3 Chebyshev polynomials of the second kind

used in A.2.2

Table A.1: Computational complexity of FXC and XFC cross correlation

Part	Operation	FXC	XFC
Windowing	Real multiplication	$2KM$	$2M-1$
F-part	Complex multiplication	$KM(\frac{1}{2}L-2)+4K$	$M(\frac{1}{4}L-1)+2$
F-part	Real multiplication	$K(M-4)$	$\frac{1}{2}M-2$
F-part	Addition	$2K(ML-2)$	$LM-2$
X-part	Complex multiplication	KM	
X-part	Real multiplication		$M^2(2K-1)+M(1-K)$
X-part	Addition		$M^2(2K-1)+M(1-K)$
X-part	Devision		$2M$
Averaging	Addition	$2M(K-1)$	
Averiging	Devision	2M	

A.3.1 Recurrence relation

$$U_0(x) = 1 \tag{A.25}$$
$$U_1(x) = 2x \tag{A.26}$$
$$U_{n+1}(x) = 2xU_n(x) - U_{n-1}(x). \tag{A.27}$$

A.3.2 conventional generating function

$$\sum_{n=0}^{\infty} U_n(x)t^n = \frac{1}{1-2tx+t^2} \tag{A.28}$$

A.4 computational complexity

See table A.1 [1]

A.5 SNR

$$\epsilon = \frac{1}{LSB}\int_{-1/2LSB}^{1/2LSB} x^2 dx = \frac{LSB^2}{12} \tag{A.29}$$

$$SNR = \frac{\frac{1}{2}(LSB \times 2^{b-1})^2}{\frac{LSB^2}{12}} = \frac{3}{2}2^{2b} \tag{A.30}$$

A.6 Window functions

Figure A.1 the stop-band attenuation of several window functions.

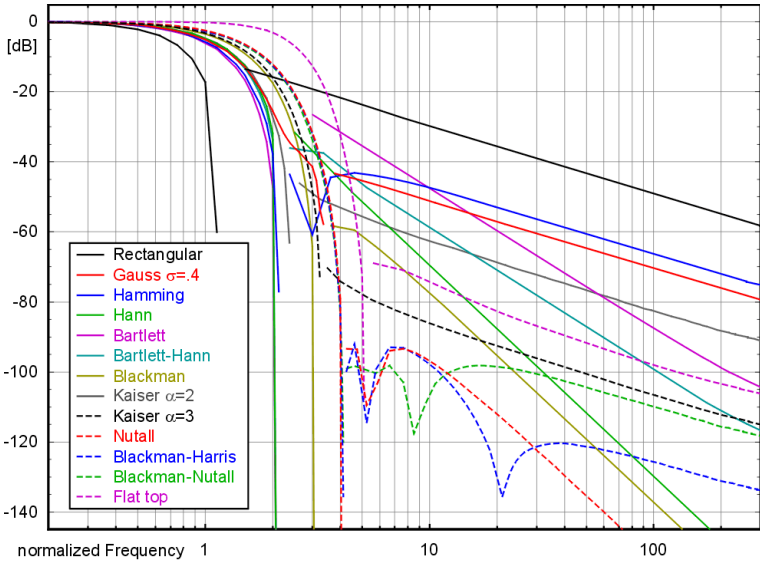

Figure A.1: Stopband attenuation of several different windows functions

Bibliography

[1] M.S Oude Alink. Increasing the spurious-free dynamic range of an integrated spectrum analyzer. Master thesis, University of Twente.

[2] Berlemann L. and Mangold S. *Cognitive Radio and Dynamic Spectrum Assignment*. John Wiley and Sons, 2009.

[3] Rameswor Shrestha, Eric A.M. Klumperink, Eisse Mensink, Gerard J.M. Wienk, and Bram Nauta. A polyphase multipath technique for software-defined radio transmitters. *IEEE Journal of Solid-State Circuits*, 41(12): 2681–2692, December 2006. URL http://doc.utwente.nl/58139/.

[4] M.C.M. Soer, E.A.M. Klumperink, Z. Ru, F.E. Vliet van, and B. Nauta. A 0.2-to-2.0ghz 65nm cmos receiver without lna achieving >11dbm iip3 and <6.5 db nf. In *IEEE International Solid-State Circuits Conference - Digest of Technical Papers, ISSCC 2009*, pages 222–223, Piscataway, January 2009. IEEE Computer Society Press. URL http://doc.utwente.nl/65507/.

[5] M. Soer. Analysis and comparison of switch-based frequency converters. Master thesis, University of Twente.

[6] D. Birru C. Cordeiro, K. Challapali and S. Shankar N. Ieee 802.22: An introduction to the first wireless standard based on cognitive radios. *Journal of Communications*, 1:3847, 2006-04.

[7] Marcel J.M. Pelgrom. *Analog-to-Digital Conversion*. Springer Science+Business Media B.V.

[8] Hui Pan and A. A. Abidi. Spectral spurs due to quantization in nyquist adcs. 51(8):1422–1439, 2004. doi: 10.1109/TCSI.2004.832755.

[9] N. Blachman. The intermodulation and distortion due to quantization of sinusoids. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 33(6):1417–1426, 1985.

[10] Willy Sansen Yves Geerts, Michiel Steyaert. *Design of multi-bit delta-sigma AD Converters*. Kluwer academic publishers.

[11] James A. Cherry; W. Martin Snelgrove. *Continuous-time delta-sigma modulators for high-speed A/D conversion*. Kluwer Academic Publishers.

[12] I.Galton. One-bit dithering in delta-sigma modulator-based da conversion. *Circuits and Systems, 1993., ISCAS '93, 1993 IEEE International Symposium on*, 2:1310–1313, 1993.

[13] G. Deokhwan Hyun; Fischer. Limit cycles and pattern noise in single-stage single-bit delta-sigma modulators. *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, 49(5):646–656, 2002.

[14] D. Fischer, G.; Hyun. Quantization noise and limit cycle patterns in single-bit delta-sigma modulators. *Instrumentation and Measurement Technology Conference, 2001. IMTC 2001. Proceedings of the 18th IEEE*, 2:729–732, 2001.

[15] G.; Steyaert M.; Craninckx J. Crombez, P.; Van der Plas. A single bit 6.8mw 10mhz power-optimized continuous-time sd with 67db dr in 90nm cmos. *ESSCIRC, 2009. ESSCIRC '09. Proceedings of*, pages 336–339, 2009.

[16] Razavi. B. *Design of Analog CMOS Integrated Circuirs*. Mc-Graw-Hill.

[17] B. Murmann. Adc performance survey 1997-2010.

[18] Masoud Zargari Hirad Samavati David K. Su Ali Tabatabaei, Keith Onodera. A dual channel $\sigma\delta$ adc with 40mhz aggregate signal bandwidth. *IEEE International Solid-State Circuits Conference*, 2003.

[19] H.; Kornegay K.; Soumyanath K. Malla, P.; Lakdawala. A 28mw spectrum-sensing reconfigurable 20mhz 72db-snr 70db-sndr dt sd adc for 802.11n/wimax receivers. *Solid-State Circuits Conference, 2008. ISSCC 2008. Digest of Technical Papers. IEEE International*, pages 496–631, 2008.

[20] K.C.-H. Chao, S. Nadeem, W.L. Lee, and C.G. Sodini. A higher order topology for interpolative modulators for oversampling a/d converters. *Circuits and Systems, IEEE Transactions on*, 37(3):309 –318, mar 1990. ISSN 0098-4094. doi: 10.1109/31.52724.

[21] Wenbo Liu ; Pingli Huang ; Yun Chiu. A $12b22.5/45ms/s3.0mw0.059mm2$ cmos sar adc achieving over 90db sfdr. *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2010 IEEE International*, page 380, 2010.

[22] M. van Elzakker, E. van Tuijl, P. Geraedts, D. Schinkel, E. Klumperink, and B. Nauta. A 10-bit charge-redistribution adc consuming 1.9 uw at 1 ms/s. 45(5):1007–1015, 2010. doi: .1109/JSSC.2010.2043893.

[23] J. L. McCreary and P. R. Gray. All-mos charge redistribution analog-to-digital conversion techniques. i. 10(6):371–379, 1975.

[24] P.; Sodini C.G.; Lee H.-S.; Sepke, T.; Holloway. Noise analysis for comparator-based circuits. *Circuits and Systems I: Regular Papers, IEEE Transactions on*.

[25] R.BogdanStaszewski. Discrete-time receiver. *IMSWorkshopWSB*.

[26] A.C. ;Cornell Univ. Ithaca-NY USA Andrews, C. ; Molnar. A passive-mixer-first receiver with baseband-controlled rf impedance matching, <6db nf, and >27dbm wideband iip3. *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2010 IEEE International*.