LUND UNIVERSITY

Floating-Point Analog-to-Digital Converter

Piper, Johan

2004

Link to publication

Citation for published version (APA):
Piper, J. (2004). *Floating-Point Analog-to-Digital Converter*. [Doctoral Thesis (monograph), Department of Electrical and Information Technology]. Department of Electroscience, Lund University.

Total number of authors:
1

General rights
Unless other specific re-use rights are stated the following general rights apply:
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.
• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal

Read more about Creative commons licenses: https://creativecommons.org/licenses/

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

LUND UNIVERSITY

PO Box 117
221 00 Lund
+46 46-222 00 00

Floating-Point Analog-to-Digital Converter

Johan Piper

Lund
19 November 2004

Copyright © 2004 Johan Piper

Competence Center for Circuit Design
Department of Electroscience
Lund University
P.O. Box 118
S-221 00 Lund
Sweden

This thesis was produced with LyX, a LaTeX front-end.

ISSN 1402-8662, no. 47

Till Diana och Helen

Abstract

To deal with the wide dynamic rage necessary for a radio receiver or corresponding applications, but to avoid impractically high resolution at high data rates, the approach of using a floating-point analog-to-digital converter (FP-ADC) has been investigated. This approach de-links the dynamic range with the resolution, and a very wide dynamic range can be achieved by an ADC with a moderate resolution.

The solution for an FP-ADC presented in this thesis is to amplify the input signal in several channels using binary weighted gains. The channel output with the largest amplitude, but still within the ADC input range, will be selected for conversion. The binary weighting is obtained by dividing the input signal with passive divider and amplifying the divider outputs by identical amplifiers. The outputs from the amplifiers are sampled individually. The selected sampled signal is then converted by a pipelined ADC. The result from the selection gives the exponent and the ADC output the mantissa of the floating-point number.

Issues on the specific problems of designing the proposed FP-ADC have been addressed, including a general discussion about a pipelined ADC along with its sub-blocks.

A thorough investigation of the distortion in a pipelined ADC due to static mismatches and systematic errors is also presented. The result of the investigation is a general approach on how to calculate the distortion in a pipelined ADC. The distortion analyses can be performed by both analytical methods and computer simulations.

A chip has been manufactured in a standard analog 0.35 μm CMOS process giving 10 bits of resolution, and a dynamic range corresponding to a 15-bit ADC. The sampling rate is 54 Ms/s using 330 mW of power.

Acknowledgments

First I would like to thank my lovely wife Helen who got me started to finalize my PhD studies. By showing me enormous patience and giving me great support I was actually able to conclude my research and finish this thesis.

This project started when Pietro Andreani talked me into pursuing PhD studies within the Electroscience department, and to investigate an interesting problem Jiren Yuan, my supervisor, presented. While being with the department my knowledge has been deepened and broadened within the area of mixed signal design, and especially data converters.

I have also had lots of fruitful discussions with my colleagues at the department. By keeping up a friendly atmosphere and a high academic ceiling, it has been fun and well worth spending so many years at the same place. Especially I would like to mention Roland Strandberg, Martin Lantz, Martin Andersson and Gang Xu who have been there to discuss everything from transistors physics to amplifier design to intricate problems in data converters. I would like to thank them for their enthusiasm and help.

I would also thank Anders "J" Johansson and Lars Olsson for taking their time to check my spelling and grammar and helping me make this thesis more readable.

No thesis would have been possible without the work of Erik Jonsson or Stefan Molund, who are constantly keeping the computers, network and tools up running day and night.

In addition to this, I would also like to thank all other parties who have been involved in this process. I cannot mention you all by name, but I will always remember your patience and help. Thank you.

The work has been sponsored by the Competence Center for Circuit

Design (CCCD) at the department of Electroscience, Lund University.

Preface

This thesis is the result of my research within the Competence Center for Circuit Design at the Department of Electroscience, Lund University. My main contributions to the field of analog-to-digital converters are the design of a 10+5-bit floating-point analog-to-digital converter and my findings during the design process. I have also contributed with a new concept for analyzing the distortion in pipelined analog-to-digital converters.

The outline of this thesis is as follows: In chapter 1 a popular science introduction to the area of floating-point analog-to-digital conversion is given. In chapter 2 the fundamental processes of analog-to-digital conversion, sampling and quantizing, are discussed. In chapter 3 the proposed FP-ADC is presented, along with an orientation of ADC architectures. In chapter 4 the functionality and performance of the analog blocks in the FP-ADC are discussed. In chapter 5 the new concept for analyzing the distortion in a pipelined ADC is presented. In chapter 6 the design tradeoffs that have been made in the design of the FP-ADC chip. In chapter 7 the actual design of the FP-ADC chip is presented. The measurements of the two FP-ADC chips produced and a resistor matching test chip are presented in chapter 8. My thesis is concluded in chapter 9.

These articles has been published and are is directly related to the thesis:

- J. Piper and J. Yuan, *A delay-balanced binary-weighted CMOS amplifier tree for a floating-point A/D converter*, Proc. of 16th NORCHIP Conference, November 1998, pp. 131-138
- J.Yuan and J.Piper, *Floating-Point Analog-to-Digital Converter*, Proc. of 6th International Conference of Electronics, Circuits and Systems, September 1999, vol. 3, pp. 1663-1666
- J.Piper and J.Yuan, *Realization of a Floating-Point A/D Converter*,

ISCAS, May 2001

- J.Piper and J.Yuan, *Distortion in Pipelined Analog-to-Digital Converters*, 2004 IEEJ International Analog VLSI Workshop, October 2004
- J. Piper and J. Yuan, *Design Considerations of a Floating-Point ADC with Embedded S/H*, submitted to the 2005 ISCAS Conference, May 2005

I have also published other articles, not directly related to the thesis:

- J. Piper and P.Andreani, *A CMOS Current Amplifier for Biological Sensors*, Proc. of 16th NORCHIP Conference, November 1998, pp. 296-301
- J. Piper and J. Yuan, *A Simulation Model for Embedding the Transistor Bias*, Proc. of the NORCHIP Conference, November 2004
- R. Strandberg and J. Piper, *Analytical Expression of the Efficiency of Phantom-Zero Compensation Applied on Negative-Feedback*, Proc. of the NORCHIP Conference, November 2004

Contents

Abstract **v**

Acknowledgements **vii**

Preface **ix**

1 Introduction **1**
1.1 Relative accuracy 2

2 ADC Fundamentals **5**
2.1 Sampling 6
2.1.1 Aliasing 7
2.1.2 Sample noise 8
2.1.3 Input bandwidth and settling time 10
2.1.4 Timing 11
2.2 Quantization 13
2.2.1 Fixed-point quantization 13
2.2.2 Floating-point quantization 15
2.2.3 Blocking 16
2.2.4 Dithering 17

3 Non-linear ADC Architectures **19**
3.1 Flash ADC 20
3.2 Pipelined ADC 21
3.2.1 The combiner 23
3.2.2 Redundant sign digit, RSD 25
3.3 Logarithmic ADC 26
3.3.1 Logarithmic ADC architectures 29

3.4 Floating-point ADC 33
3.4.1 Floating-point ADC architectures 35
3.4.2 Normalizing amplifiers for FP-ADC 38
3.5 The proposed FP-ADC architecture 39

4 Analog Building Blocks 43
4.1 Switched-capacitor circuits 43
4.1.1 Noise and correlated double-sampling 45
4.2 Switches 47
4.2.1 The basic NMOS switch 47
4.2.2 Non-ideal effects in the NMOS switch 48
4.2.3 More complex switches 52
4.3 Amplifiers in a time-discrete environment 56
4.3.1 Amplifier models 56
4.3.2 Required gain 59
4.3.3 Settling time 60
4.3.4 DC gain and settling 66
4.4 Voltage dividers 67
4.4.1 Resistive and capacitive dividers 67
4.4.2 R-2R ladder divider 68

5 Distortions in Pipelined ADC 71
5.1 The general error function 72
5.1.1 Conclusions 74
5.1.2 Multi-bit stages 76
5.2 The error power 77
5.2.1 Sinusoids 77
5.3 Circuit example with simulations 81
5.3.1 Simulation of the non-linearity 83
5.3.2 Harmonic power of the static error 85
5.3.3 Harmonic power of the total error 85
5.4 Differential and integral non-linearity 88

6 Design Considerations for the Proposed FP-ADC 91
6.1 Overall design tradeoffs 91
6.2 Matching 93
6.2.1 Delay matching 93
6.2.2 Gain matching 95

6.2.3 Offset cancellation 96
6.3 Noise 97
6.4 The control algorithm 99
6.4.1 Comparators of the controller 100
6.4.2 Gain stage saturation 101

7 The Design of the Floating-Point ADC 105
7.1 Noise calculation 107
7.2 Sampling capacitors 108
7.3 The phase generator 109
7.4 Resistive divider 111
7.5 The gain stage 113
7.6 The sample and pipeline stages 119
7.6.1 The sample-and-hold stage 119
7.6.2 The pipeline stage 119
7.6.3 Sample and pipeline stage amplifier 120
7.7 The flash ADC 123
7.8 The comparator 124
7.9 The controller 126
7.10 Synchronizing and digital error correction 126
7.11 Reduction of the switching noise 129

8 Chip Measurements 133
8.1 Test setup 133
8.1.1 INL and DNL measurements 133
8.1.2 Harmonic distortion measurements 134
8.2 8+4-bit FP-ADC measurements 135
8.3 10+5-bit FP-ADC measurements 136
8.3.1 Measurement results 139
8.3.2 Discussion 143

9 Conclusions 149
9.1 Suggestions for future work 150

Bibliography 153

A List of Symbols 163

B The CMOS Transistor Model **169**
B.1 The cutoff region 170
B.2 The triode region 170
B.3 The saturation region 171
B.4 The capacitances 172
B.5 The noise model 173

C Non-Ideal Effects in Dividers **175**
C.1 Resistive feedback 175
C.1.1 The mismatch in resistive feedback 175
C.1.2 Mismatch in a capacitive feedback 176
C.1.3 Noise in a resistive feedback 176
C.2 R-2R ladder 176
C.2.1 The dividing factor in an R-2R ladder 177
C.2.2 The mismatch in an R-2R ladder 178
C.2.3 The delay in an R-2R ladder 180
C.2.4 The noise at the terminals in an R-2R ladder 181

Chapter 1

Introduction

Today the general trend goes towards making machines, toys and communication tools digital. By the development of powerful and complex digital signal processing methods we are able to make our gadgets more intelligent thus making our lives easier, more productive or just more pleasurable. However there is a catch as humans cannot interpret digital signals, nor can digital systems interpret our world, which is analogue. To be of any use for us there has to be some interface between the digital world and the analog (natural) world surrounding it. Therefore the components that makes the interface is called analog-to-digital converter (ADC) or digital-to-analog converter (DAC), depending the direction on the information flow. These converters translate an electrical quantity (a voltage or a current) from a sensor (thermometer, radio receiver, camera, microphone, etc.) into digital value, or vice versa.

Figure 1.1: A signal processing system.

A system using digital converters is shown in Figure 1.1. The system could be exemplified by a digital camera. The image in front of the camera is projected onto a CCD-chip in the camera, the sensor. The CCD-chip converts the light that hit the sensor by subdividing the image into a large number of pixels, where the light that hit each pixel are converted into electrical charges. The charges form a flow of current that is converted into a value by an ADC. This is done for every pixel and the result is a digital image where the color and intensity of every pixel is described by a number. The computer can store, manipulate or send the image to another computer. When someone wants to look at the image, the value stored in each pixel is converted by a DAC into a current that lights up the corresponding pixel, with the correct color and intensity, on the computer screen.

1.1 Relative accuracy

You want the analog-to-digital conversion to be accurate, fast and have low power consumption. There is a cost involved in this desire of perfection. High accuracy, speed and low power consumption put high demands on the ADC design, not only for the analog parts but also for the digital parts [1]. You have to define your minimum requirements to fulfill the transport of information without a significant loss of information.

Quite often it is more relevant to talk about the relative accuracy in a system than the absolute accuracy. If we want a variable to describe a very large and a very small value with the same absolute accuracy, many digits in the number are needed. In many cases it is not necessary to have full accuracy at all magnitudes. It is enough to keep the relative accuracy constant. Then a floating-point number comes in hand. An ADC that can perform a floating-point number conversion is called a floating-point ADC, abbreviated as FP-ADC. The accuracy of an FP-ADC is called the resolution.

Take a distance for example. The driving distance from Norregatan 21, Malmö to Oskarsgatan 49B, Lidingö to is 621.6 km (in Sweden) [2]. Here the accuracy is 100 m. When someone asks you how long the trip is you are more likely to say 620 km. In this case you rounded the number so only two significant digits remains. To tell all the digits would be silly as the distance would vary with the road-map and who asked (in [3] the distance is 630 km). The distance to the sun is approximately 149 600 000 km [4]. The accuracy

is about 100 000 km and it would not matter much if you are in Stockholm or in Malmö. By using a floating-point notation we can describe the distances with two significant numbers as $6.2 \cdot 10^5$ and $1.5 \cdot 10^{11}$ m respectively. This way we are able to describe both distances with the same number if digits.

To measure physical parameters with these differences in magnitude is common for many physical processes. A few examples are audio, video, seismology, and mobile communication. If the signal is continuous it is not much of a problem as we can adjust the range of the ADC with automatic gain control (AGC). This is like adjusting the light intensity depending whether it is light or dark outside. The AGC solution is robust but slow. For transient and non-repetitive signals we have to adjust the range instantly. A solution is to build an ADC with a resolution that incorporates the whole dynamic range. To build such an ADC can be very expensive. Think about making a ruler from the earth to the sun with a tick mark every 100 m. Also the ruler is definitely over-sized if it is used most of the time to measure driving distances. Then a floating-point solution might be a better choice. Then the size of the ruler and its tick marks are adjusted according to the magnitude of what is currently measured.

Chapter 2

ADC Fundamentals

A signal in an analog system is continuous, both in time and in amplitude. The signal has infinite precision but its information handling capacity is limited by the noise, the distortion and the bandwidth of the system handling the signal [5].

A time-discrete signal is a signal that only changes its value at specific time instants defined by a sample clock. The system that handles the signal will show a transient behavior at the start of every clock cycle, before it settles to a stationary point. This way the frequency behavior of the system does not affect the signal, as long as the sample period is long enough. Instead the limit on the signal bandwidth is set by the sampling frequency. The signal bandwidth of a time discrete signal is limited to the Nyquist frequency, i.e. half the sampling frequency. Still the noise and distortion of the system itself limit the performance. The process of making an analog signal into a time discrete signal is called sampling.

An amplitude discrete signal is a signal that is continuous in time, but can only take on discrete values in amplitude. An amplitude discrete signal is insensitive to amplitude noise and also distortion, as a small perturbation can be corrected by rounding to the nearest discrete value. The process of converting an analog signal into an amplitude discrete signal is called quantization.

A digital signal is a signal that is both sampled and quantized. That means that both the time and the amplitude of the signal are discrete. The signal changes at known time instants and only into known discrete values. A digital system is very insensitive to noise, distortion and bandwidth of

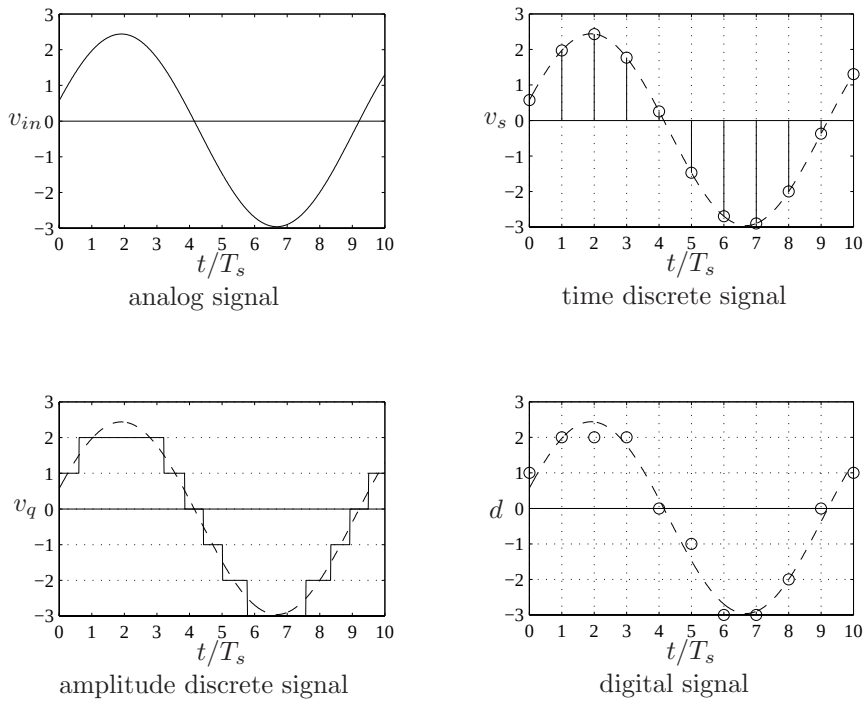

Figure 2.1: The four signal domains, exemplified by a sinusoid waveform.

the system that handles the signals. However, there can be infinite accuracy neither in time nor in amplitude of the signal.

In Figure 2.1 the four signal domains are visualized with a sinusoid signal. Normally the signal is sampled before it is quantized when converted from the analog to the digital domain. The reason is that a sampled signal has a low signal slope which is beneficial when designing an ADC.

2.1 Sampling

When we sample we take the instant value of a continuous time signal and storing it on a memory. The electrical component used in an ADC as memory

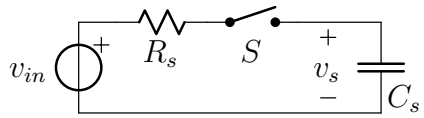

Figure 2.2: A passive sampler.

is the capacitor. Then the signal is stored as a voltage. Inductors can also be used for memories but are not of sufficient quality in a standard IC-process technology.

The simplest sampling device is a passive sampler which is shown in Figure 2.2. Here the signal is passing through to the sampling capacitor when the switch is closed. When the switch is opened the voltage is preserved on the capacitor. The signal $v_s(t)$ becomes a track-and-hold signal version of $v_{in}(t)$. It is the held value that is used for the time discrete calculation, not the actual waveform of $v_s(t)$. The sampled voltage is therefore the track-and-held voltage taken in the holding phase

$$v_s(kT_s) = v_{in}(kT_s) \quad , k \in \{\ldots -1, 0, 1 \ldots\} \tag{2.1}$$

where T_s is the sampling period. This way the signal is sampled.

One must, of course, pay attention to the low-pass filter formed by R_s and C_s so the signal can pass.

2.1.1 Aliasing

An effect of sampling is aliasing. Aliasing is the effect that the spectrum of a sampled signal is repeated every f_s, the sampling frequency $f_s = 1/T_s$. This means that the spectrum is effectively folded at $\pm f_s/2$. This leads to the sampling theorem which states that the input signal can be reconstructed exactly from the sampled signal if and only if [6]

$$|f_{in}| < f_s/2 \tag{2.2}$$

Mathematically, sampling is the multiplication of two signals, the analog signal $v_{in}(t)$and the pulse train $p(t)$

$$v_s(t) = v_{in}(t) \cdot p(t) \tag{2.3}$$

where

$$p(t) = \sum_{k=-\infty}^{\infty} \delta(t - kT_s) \tag{2.4}$$

The Fourier transform of $p(t)$is

$$\mathcal{F}p(t) = P(f) = \begin{cases} 1, & f = kf_s \\ 0, & \text{otherwise} \end{cases} \tag{2.5}$$

Multiplication in the time domain means convolution in the frequency domain. Thus the spectrum of the sampled signal is

$$V_s(\omega) = V_{in}(f) \star P(f) = \sum_{k=-\infty}^{\infty} V_{in}(f - kf_s) \tag{2.6}$$

The sampling process is visualized in Figure 2.3, both in the time and frequency domain.

A consequence of the folding is that the power of v_{in} is the same as v_s, between $\pm f_s/2$,

$$\int_{-\infty}^{\infty} V_{in}(f)df = \int_{-\frac{f_s}{2}}^{\frac{f_s}{2}} V_s(f)df \tag{2.7}$$

2.1.2 Sample noise

There is always noise associated with sampling. This noise arises from the fact that there is a finite resistance in series with the switch (including the resistance of the switch itself). This resistance gives rise to thermal noise. The resistor noise is filtered by the low-pass filter formed by R_s, C_s and is sampled when the switch opens as seen in Figure 2.2.

The noise power on the capacitor is

$$\bar{v}_n^2 = 4kTR_s \int_0^{\infty} \left| \frac{1}{1 + j\omega R_s C_s} \right|^2 d\omega \tag{2.8}$$

where $k = 1.3807 \cdot 10^{-23}$ is Boltzmann's constant and T the absolute temperature. This integral yields the classical expression of the sample noise voltage

$$\bar{v}_n^2 = \frac{kT}{C_s} \tag{2.9}$$

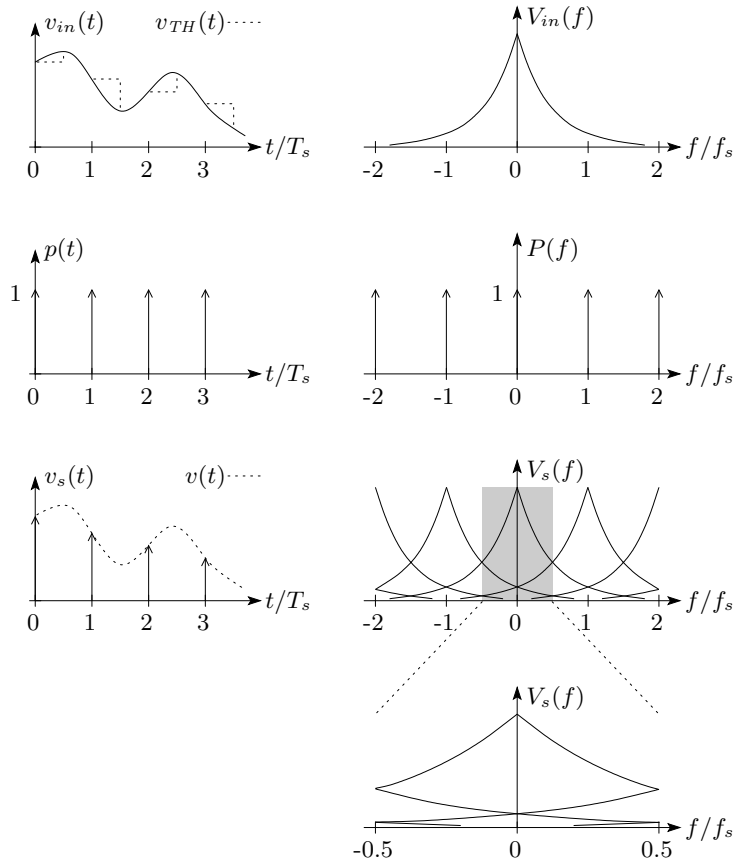

Figure 2.3: The input signal, the pulse train and the sampled signal are shown in both time and frequency domain. The gray area is $|f| < f_{s/2}$.

What is interesting here is that the resistance is unimportant. It is the temperature and the capacitance of the sampling capacitor that set the lower limit of the noise.

Let us calculate the energy of a sinusoid signal expressed in voltage

$$\bar{v}_s^2 = \frac{V_{FS}^2}{8} \tag{2.10}$$

where V_{FS} is the full scale input voltage and also the peak-to-peak voltage of the sinusoid. Then it is possible to calculate the maximal signal-to-noise (SNR) of the time discrete signal by using equation 2.9

$$\text{SNR} = \frac{\bar{v}_s^2}{\bar{v}_n^2} = \frac{V_{FS}^2 C_s}{8kT} \tag{2.11}$$

This equation shows that the SNR for a sinusoid signal is proportional to the size of the sampling capacitor, the square of the input range and inversely proportional to the temperature.

2.1.3 Input bandwidth and settling time

The input bandwidth is, in the case of the basic passive sampler, set by the low-pass filter formed by R_s and C_s. Let us define the box bandwidth as the bandwidth of a box filter that gives the same noise as the low-pass filter. The box bandwidth can be calculated by comparing equation 2.9 to the thermal noise energy of a resistor

$$\frac{kT}{C_s} = 4kTR_sB \tag{2.12}$$

Solving for B yields the box bandwidth

$$B = \frac{1}{4R_sC_s} \tag{2.13}$$

The box bandwidth is the bandwidth of a box filter that gives the same noise energy as the actual low-pass filter. The box bandwidth is larger that the -3 dB bandwidth of the low-pass filter, which equals $1/2\pi R_sC_s$. The box bandwidth can be used when calculating the noise from other noise sources.

Sometimes it is interesting to calculate the minimal sampling frequency to minimize broadband noise from other noise sources than R_s. Then the box bandwidth should be minimized with respect to the sampling frequency.

When the switch changes state the transients needs to die out. Then an accurate sample of the input signal can be taken. The time needed is called the settling time t_s. The relation between the settling time and the time constant τ_s of the low-pass filter is

$$t_s = \theta(n)\tau_s = \theta(n)R_sC_s \tag{2.14}$$

for an n-bit ADC. The traditional settling time is the time it takes for the signal to settle within a window of $\pm\frac{1}{2}$ LSB[1]. Then according to equation 4.34 ($\epsilon = 2^{-n-1}$) the relation between the settling time and τ_s can be resolved

$$\theta(n) = \frac{t_s}{\tau_s} = (n+1)\ln 2 \tag{2.15}$$

The relation between the sampling frequency and the settling time is

$$t_s \leq \eta T_s = \frac{\eta}{f_s} \tag{2.16}$$

where η is the duty-cycle. The duty-cycle is the relation the switch open time and the sample period. Using equation 2.17 the relation between the sampling frequency and the box bandwidth is solved

$$f_s \leq \frac{\eta}{t_s} = \frac{\eta}{\theta(n)R_sC_s} = \frac{4\eta}{\theta(n)}B \tag{2.17}$$

To minimize the noise from other noise sources than R_s the box bandwidth needs to be as small as possible with respect to the sampling frequency. For a duty-cycle of 50% the optimal box bandwidth is

$$B = \frac{\theta(n)}{2}f_s = \frac{(n+1)\ln 2}{2}f_s \tag{2.18}$$

2.1.4 Timing

One of the major problems with high-speed converters is the timing errors. An accurate time reference is needed for sampling. The sample time is defined by a physical clock, distributed in the ADC. All timing errors in the clock give rise to errors in the sampling time instants

$$t_k = kT_s + t_{j,k}$$

[1] Least significant bit, see section 2.2.

where $t_{j,k}$ is the time jitter at the sampling instant k. The sampling time instant is incorrect, which can be attributed to an error in the amplitude. A random jitter will give a random error mainly visible as an increased noise, while a signal-dependent jitter will give rise to harmonics of the signal.

When sampling a sinusoid signal the jitter error power becomes [7]

$$\bar{v}_j^2 \approx 2(\pi a_1 f_{in})^2 \cdot \bar{t}_j^2 \tag{2.19}$$

for a random jitter. $\bar{t}_j^2$ is the time variance of the jitter t_j. The jitter error is dependent on the amplitude and frequency of the input signal, a_1 and f_{in} respectively. By comparing the jitter energy to the signal energy the SNR can be calculated

$$\text{SNR} = \frac{\bar{v}_s^2}{\bar{v}_j^2} = \frac{1}{(2\pi f_{in}\bar{t}_j)^2} \tag{2.20}$$

Interesting is the factor $1/f_s^2$; the SNR in inversely proportional to the square of the signal frequency.

The timing errors can be attributed to four main error sources [8].

1. Sampling clock jitter. The sampling jitter originates from noise processes in the clock signal generation inside and outside the converter. This gives rise to phase-noise on the sampling clock. i.e. an uncertainty in time and also referred to as clock jitter.

2. Finite slew-rate clock edges. The clock edged have a finite slew-rate and if the instant when the sampling switch turns off is dependent of the amplitude of the input signal, then the sampling instant will be dependent of the signal.

3. Skew in clock and signal distributions. In many cases, the sampling of the signal is made at many points at the same time. If the clock and the signal do not appear at the same time at all the points a signal dependent error might be introduced.

4. Signal-dependent delay. If an amplitude-limiting circuit is followed by a frequency-limiting circuit a delay might be introduced that it dependent of the slope of the signal. An example is a comparator.

The sampling clock jitter cannot be removed even though it can be reduced by careful design. Finite slew-rate of the clock edges can be remedied by making the turn-off of the sampling switch independent of the signal. To

remove the skew in clock and signal, it is important to synchronize the wire delay of the signal and the clock. The clock and the signal should appear at the same time at all the sample points.

The timing errors become unimportant as soon as the signal is sampled and held. This is the case for pipeline stages, where the residue from the preceding pipeline stage is re-sampled. Then the signal is time discrete the signal slope approaches zero thus timing errors does not introduce errors in time discrete systems.

2.2 Quantization

When we quantize a continuous signal we want to map the real valued signal on to predefined discrete values. The process is called quantizing.

The analog signal level v_{in} can be expressed as the quantized value d plus the quantization error ϵ

$$v_{in} = d + \epsilon \tag{2.21}$$

The quantized value chosen is depending on the rounding. For an ADC the rounding is chosen to minimize the quantization error; that means rounding to the nearest quantized value.

2.2.1 Fixed-point quantization

For a fixed point number representation we use an array of bits, the binary word w, to represent a number. If the word length is n bits then we can express the number as

$$w = \sum_{k=0}^{n-1} b_k 2^k \tag{2.22}$$

where b_k is the bit at position k in the binary word. The number of unique values we can represent is 2^n. The resolution of the number representation is the smallest step between the unique values. It equals the value change when toggling the bit b_0, also called the least significant bit (LSB). It is common to use the LSB as a reference when measuring the quality of converters, like e.g. non-linearity.

The quantized value d can be represented by a binary word using the transformation

$$d = w \cdot \frac{V_{FS}}{2^n} \tag{2.23}$$

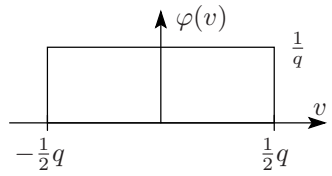

Figure 2.4: PDF of quantization error

where V_{FS} is the full scale input. The smallest step d can take is denoted by q and equals

$$q = \frac{V_{FS}}{2^n} \tag{2.24}$$

q is equivalent to one LSB. The largest absolute error from a quantization process is half the quantization step hence the quantization error is in the interval from $-\frac{q}{2}$ to $\frac{q}{2}$. Under the assumption that the input is a random signal with the standard deviation $\sigma_s \gg q$ then the quantization error has a white spectral density with a rectangular probability density function (PDF) [9, 10]. Therefore the quantization error is often called quantization noise and this noise plays an important role in designing a system using ADC. The PDF of the quantization error is illustrated in Figure 2.4.

It is a simple matter to calculate the energy of this noise and compare it to the signal power to obtain the SNR. The power of the quantization error translates to the variance of the error and equals

$$\bar{\epsilon}^2 = \frac{1}{q}\int_{-\frac{q}{2}}^{\frac{q}{2}} v^2 dv = \frac{1}{12}q^2 = \frac{V_{FS}^2}{12 \cdot 2^{2n}} \tag{2.25}$$

Comparing the power of the error to the power of a full scale sinusoid ($V_{FS}^2/8$) the well known expression for the SNR of a quantized signal is obtained [8]

$$\text{SNR} = \frac{3}{2}2^{2n} = 6.02n + 1.8\,\text{dB} \tag{2.26}$$

For non-sinusoid input signals the above expression is not exactly true. The input signal is then modeled as a stochastic process with some PDF. Two popular distributions for the input signal are the Laplacian and the Gaussian distributions. The Laplacian distribution should be used for a single signal input, like music, and the Gaussian for multiple signals, like a multi-carrier signal, where the central limit theorem can be used. The SNR

for fixed-point quantization then becomes [11]

$$\mathrm{SNR} = 6.02n + 10.8 - 20\log_{10}\left(\frac{V_{FS}}{\sigma(v_{in})}\right)\ \mathrm{dB} \tag{2.27}$$

under the assumption that no clipping occurs. $\sigma(v_{in})$ is the standard deviation of the input signal distribution.

Generally the SNR is increasing by 6 dB for every bit added.

2.2.2 Floating-point quantization

A floating-point value is described by two digital words, the mantissa w_M and the exponent w_E

$$d = w_M 2^{w_E} \cdot \frac{V_{FS}}{2^{n+2^{n_e}-1}} + \frac{V_{FS}}{2} \tag{2.28}$$

where n and n_e are the word lengths of the mantissa and the exponent respectively. $V_{FS}/2$ is the input voltage corresponding to a zero digital signal. The mantissa needs to represent both negative and positive values. Therefore the most significant bit in the mantissa is a sign bit. The binary word representation used in this thesis is a variant of the offset binary representation

$$w_M = \sum_{k=0}^{n-1} b_k 2^k - 2^{n-1} + 0.5 \tag{2.29}$$

where the term 0.5 is added to remove the offset between the negative and the positive numbers.

The floating-point number representation is redundant. The number of unique values we can represent with floating-point representation is 2^{n+n_e-1}.

Example These floating-point numbers ($n = 5$, $n_e = 3$) are the same

$$(1)0011 \cdot 2^{011} = (1)0110 \cdot 2^{010} = (1)1100 \cdot 2^{001}$$

The digit inside the parenthesis is the sign bit of the mantissa.

According to equation 2.28 the smallest increment that can be made by a floating-point number is

$$q_{min} = \frac{V_{FS}}{2^{n+2^{n_e}-1}} \tag{2.30}$$

For floating-point quantization q, the smallest step between two quantization levels, varies with the signal amplitude so the probability density function will not be uniform. Instead one can use the PDF of the relative quantization error.

The SNR for a sinusoid signal will be roughly constant with the input amplitude. The maximum number of mantissa bits used for the quantization is n. Therefore the maximum SNR of a floating-point signal will be equal to a fixed point quantization with a bit width of n

$$\mathrm{SNR} = 6.02 \cdot n + 1.8\,\mathrm{dB} \tag{2.31}$$

as expressed in equation 2.26. For other signals than sinusoids the power of the relative quantization error becomes

$$\bar{\epsilon}_r^2 = \frac{2^{-2(n-1)}}{8 \ln 2} \tag{2.32}$$

for most distributions of the input signal [10, 11]. The relative error will have a white spectral density and be uncorrelated to the input signal for all practical cases [10]. Then the SNR for a floating-point quantization for most signal distributions is

$$\mathrm{SNR} = 6.02n + 1.4\,\mathrm{dB} \tag{2.33}$$

The dynamic range[2] (DR) of a floating-point quantization will be different from its SNR. The smallest increment is much smaller compared to a fixed-point quantization why much smaller signals can be distinguished. Therefore the dynamic range becomes the power of a full-scale input sinusoid compared to $q_{min}^2/12$

$$\mathrm{DR} = \frac{3}{2} 2^{2(n+2^{n_e}-1)} = 6.02 \cdot (n + 2^{n_e} - 1) + 1.8\,\mathrm{dB} \tag{2.34}$$

2.2.3 Blocking

Floating-point quantization is especially destructive when the input sees two input signals with different magnitudes. The minimum quantization step q value defines the quality of the quantization and it changes with the instantaneous amplitude. The problem when adding a weak and a strong signal is that the quantization step corresponds to the sum of the signals. Then

[2] The range where the signal is stronger than the quantization noise

the weak signal will suffer a large relative quantization. If the signals differ a lot in magnitude the small signal might be smaller than q and effectively disappear. This is called blocking and is a fundamental disadvantage for systems using compression.

2.2.4 Dithering

If the sample clock and signal frequency are correlated in any way, then the assumption that the input is a random signal does not hold. The quantization error is still rectangular but correlated to the signal. This manifests itself in a colored quantization noise. The noise will show peaks at the harmonics of the input signal.

This is the case when synthesizing frequencies using a DAC. Then the frequency at the output is always a rational number times the reference (sampling) frequency and thus the quantization noise will be colored.

The remedy is to use dithering [10]. By introducing a noise to the signal the correlation between the sampling frequency and the signal is removed and the quantization noise will become white. But dithering will increase the overall noise as well. For the interested reader, the articles [12, 13, 14] give more information on this topic

Chapter 3

Non-linear ADC Architectures

To perform an analog-to-digital conversion we need to compare the input signal, or a processed version of the input signal with threshold values generated from a voltage reference.

The ADC is shown as a black box in Figure 3.1. The ADC has one input, the analog signal v_{in} and two references, the amplitude reference V_R and the time reference here represented by the sampling rate f_s. The quality of the digital signal d is dependent on the quality of the amplitude and the time references.

The ADCs presented in this chapter are only a subset of the many various types of ADC existing today. The ADCs described in this chapter are only

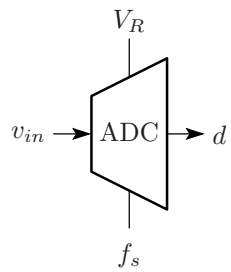

Figure 3.1: The ADC as a black box.

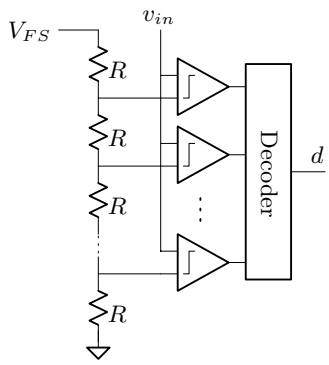

Figure 3.2: A flash converter.

to those related to the FP-ADC solution presented in this thesis.

3.1 Flash ADC

The flash converter is the basic type of ADC. It is also considered to be the fastest. The speed of the flash is mainly set by the speed of the comparators. Thus the flash ADC is inherently very fast. The three basic functional blocks of the flash converter are the references, the comparators and the decoder.

The function of the flash ADC, as seen in Figure 3.2, is as follows. The input signal v_{in} is compared simultaneously to all the references. The references are generated by a resistor ladder and the voltage reference V_{FS}. Where the signal is larger than the reference the comparator will give a logical high to the decoder. Thus the output from the comparators is a temperature code[1] and needs to be recoded e.g. into a binary code. This is done in the decoder.

The basic functionality of the flash ADC is very simple but its size grows exponentially with the number of bits, n. The flash converter needs a reference and a comparator for every quantized value. This means that the number of comparisons (and references) is

$$N_{comp} = 2^n - 1 \tag{3.1}$$

[1]A temperature code is a code where number of '1' in the code word indicates the value, e.g. the value 0011 1111 of a temperature code equals the decimal value of 6.

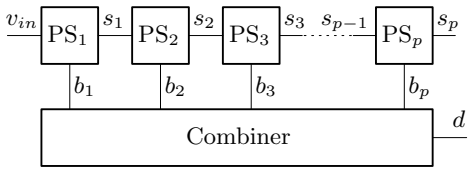

Figure 3.3: A pipelined ADC. p is the number of stages in the pipelined ADC.

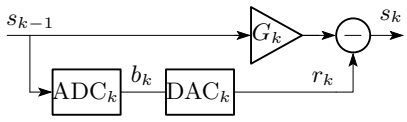

Figure 3.4: Block diagram of a pipeline stage.

Thus for every additional bit the size of the flash converter is doubled. Also the accuracy of the comparators increases with the same amount. This means that the offset voltages of the comparators need to be

$$|v_{off}| < \frac{V_{FS}}{2^{n+1}} \tag{3.2}$$

The practical resolution of a flash converter is limited to $n \leq 8$. However, there exist clever solutions to reduce the number of comparators and references, like interpolation and folding [15, 16, 17].

3.2 Pipelined ADC

A divide-and-conquer technique can be used to decrease the complexity of an ADC. The conversion is then done in steps. Such an ADC, called a pipelined ADC, is shown in Figure 3.3. The most significant bit(s) b_1 are converted in the first stage (PS_1). The residue from the coarse quantization s_1 is fed to the next pipeline stage (PS_2) to obtain the finer bit(s) b_2. This is repeated until all bits are converted. The digital outputs from the pipeline stages are combined in the combiner. The result is the digital output d.

The stage operations in a pipelined ADC can be assigned to three categories. They differ in the way the references and the analog input signal are processed:

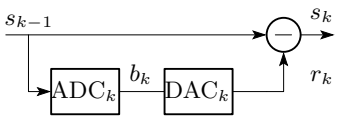

Figure 3.5: Block diagram of a residue stage.

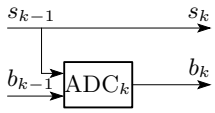

Figure 3.6: Block diagram of a sub-ranging stage.

1. The pipeline stage (presented in Figure 3.4); The quantized signal is subtracted from the analog input signal. The residue is amplified in such a way that the same references can be used in the next stage.

2. The residue stage (presented in Figure 3.5); The quantized signal is subtracted from the analog input signal. The residue of the subtraction is quantized in the next stage using a finer reference.

3. The sub-ranging stage (presented in Figure 3.6); A sub-range of the references, selected by the coarse bits, is used for fine conversion of the input signal in the next stage.

The pipeline stage has the benefit that only a few references have to be generated since the references can be reused. The drawback is that a high accuracy amplifier is needed. This is the type of stages used in the FP-ADC proposed in this thesis.

A pipeline stage is shown in Figure 3.4. The transfer function of the pipeline stage is

$$s_k = G_k \cdot s_{k-1} - r_k \tag{3.3}$$

where s_{k-1} and s_k are the signals at the input and the output of the pipeline stage respectively. The gain G_k is the gain of the pipeline stage and r_k is a reference that is subtracted from the signal.

The gain G_k equals the radix of the pipeline stage. The radix is the

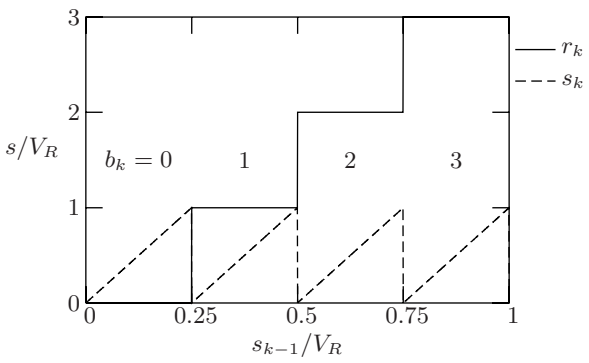

Figure 3.7: The signals in a 2-bit pipeline stage.

number of digits the pipeline stage provides. For a binary converter

$$G_k = 2^{n_k} \tag{3.4}$$

where n_k is the number of bits from pipeline stage k. The reference r_k is a function of the input signal. It is chosen in such a way that the signal is always within the range of the following pipeline stage. Due to the sub-ADC-DAC conversion r_k is a piecewise linear function of s_k that takes predefined discrete values. An example of the signals in a 2-bit pipelined stage can be found in Figure 3.7. Here $G_k = 4$ since 2 bits give 4 digits.

The number of comparisons needed in the pipelined ADC is dependent of the radix on the pipeline stages

$$N_{comp} = \sum_{k=1}^{p} G_k - p \tag{3.5}$$

where p is the number of stages in the pipelined ADC. The number of comparisons can be greatly reduced compared to a flash converter. On the other hand each pipeline stage needs an accurate amplifier.

3.2.1 The combiner

When the pipeline stages are chained in the pipelined ADC the input-output relation becomes an iteration of equation 3.3. The flow-graph in Figure 3.8 shows the input-output relation of the pipelined ADC.

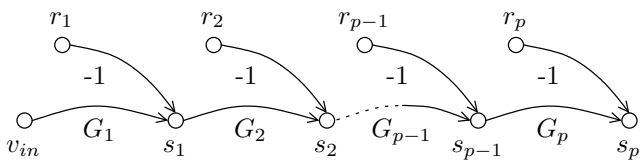

Figure 3.8: Flow-graph of a pipelined ADC.

From the flow-graph the signal at the output of the last pipeline stage is found as

$$s_p = v_{in} \prod_{l=1}^{p} G_l - \sum_{k=1}^{p} \left[r_k \prod_{l=k+1}^{p} G_l \right] \tag{3.6}$$

How do we make a digital signal out of this? If we solve equation 3.6 for v_{in} we see that the input signal is

$$v_{in} = \sum_{k=1}^{p} \frac{r_k}{\prod_{l=1}^{k} G_l} + \frac{s_p}{\prod_{l=1}^{p} G_l} \tag{3.7}$$

The answer to the question is that we know exactly the output r_k from the sub-DACs through the digital output b_k

$$r_k = \mathrm{DAC}_k(b_k) \tag{3.8}$$

An approximation of the signal would be the first sum in equation 3.7. This approximation gives the digital output,

$$d = \sum_{k=1}^{p} \frac{r_k}{\prod_{l=1}^{k} G_l} \tag{3.9}$$

The error we make by this approximation is the quantization error (the right term in equation 3.7). If the gain is two for all pipeline stages this equation reduces to the well known expression for a digital word, $\sum_{k=1}^{p} r_k/2^k$.

So effectively, the combiner can be implemented by adding the weighted digital outputs of the pipeline stages.

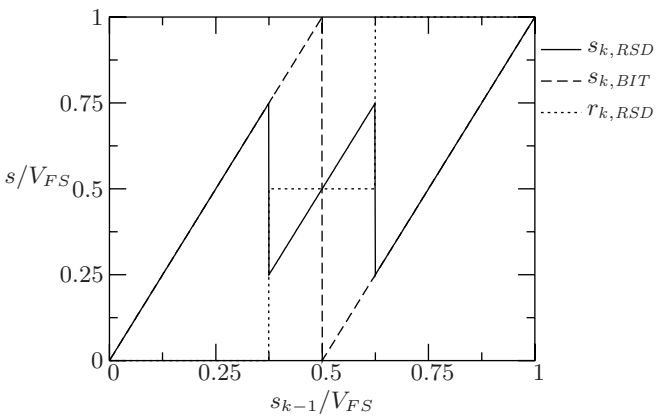

Figure 3.9: The residues from a 1-bit RSD and a 1-bit pipeline stage.

3.2.2 Redundant sign digit, RSD

The residue of the 1-bit pipeline stage is the dashed line shown in Figure 3.9. If the reference value of $V_{FS}/2$ is not exact, then s_k will go outside the range $0 \rightarrow V_{FS}$. The result is that the following stages will be saturated and the output of the pipelined ADC signal would show missing codes. That means that the comparators in the sub-ADC need to be very accurate. For an over/undershoot of less than $\frac{1}{2}$ LSB in the first pipeline stage the offsets in the sub-ADC need to be

$$|v_{off}| < \frac{V_{FS}}{2^{n+1}} \tag{3.10}$$

This is the same requirement as for the flash ADC, but instead of $2^n - 1$ comparators we only need $G_1 - 1$. Also, for every pipeline stage k the requirement on the comparators is relaxed by a factor of G_{k-1}.

There exist several techniques to correct the offset errors in the sub-ADC. The correction is made in such a way that an offset in the sub-ADC never saturates the following pipeline stages. One solution is to increase the input range of the following pipeline stages. Another solution is to decrease the gain in the pipeline stages and adjust G_k accordingly in the combiner.

The third solution is to use redundant sign digit (RSD). The idea is to split the step of a bit-transition in the pipeline stage into two halves.

The residue function $s_{k,BIT}$ of a 1-bit pipeline stages is shown in Figure

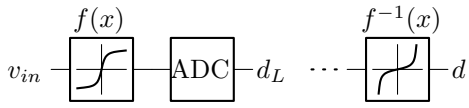

Figure 3.10: A block diagram of non-uniform quantization using a compressor and an expander to generate a linear transfer of the system.

3.9. The step of $r_{k,BIT}$ is split into two halves at the voltages $\frac{1}{2} \pm \frac{1}{8} V_{FS}$. The result is the waveform of the RSD reference $r_{k,RSD}$. A small variation on the threshold voltages does not force the residue of the 1-bit RSD stage[2] outside the input range of the following pipeline stage. This way the 1-bit RSD pipeline stage can accept sub-ADC offset errors up to

$$|v_{off}| < \frac{V_{FS}}{8} \tag{3.11}$$

without any clipping in the following stages. Further the sub-ADC offset errors, within the tolerance, are completely cancelled in the combiner. With such a high tolerance it is possible to use low-power comparators [18].

3.3 Logarithmic ADC

For systems that require a very large dynamic range the needed resolution becomes very high for a linear ADC. However many systems accept that the resolution is proportional to the magnitude of the signal. A way to achieve this is to use a logarithmic-like function to compress the signal. Finally, to retrieve the linear signal, an expander is used that is the inverse function of the compressing function. This way a logarithmic ADC (L-ADC), with a non-uniform quantization, can be implemented. The process of compressing and expanding is called companding, and the pair of expanding and compressing function is called the codec. A block diagram of a companding system is shown in Figure 3.10, where $f(x)$ is the compressing function and $f^{-1}(x)$ the expanding function.

For the codecs described below, the maximum signal amplitude has been normalized to 1. A first approach would be to use a logarithm as the codec,

[2] Also referred to as a 1.5-bit pipeline stage

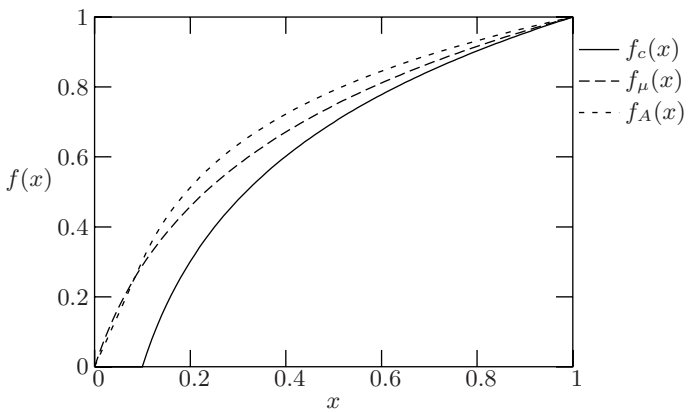

Figure 3.11: The codec in the positive half-plane according to the c-law, the μ-law, and the A-law, $c = \mu = A = 10$

the c-law

$$f_c(x) = \text{sign}(x)\frac{\ln(c|x|)}{\ln c} \tag{3.12}$$

where c is the compression coefficient. The logarithm function goes to $-\infty$ when $x \to 0$. Thus the c-law can be used for input signals, $1/c \leq |x| \leq 1$.

There exist two standards of codecs that can handle signals close to zero, the μ-law and the A-law. The μ-law was suggested by Bell Laboratories and the codec is

$$f_\mu(x) = \text{sign}(x)\frac{\ln(1+\mu|x|)}{\ln(1+\mu)} \tag{3.13}$$

where μ is the compression coefficient. The A-law was instigated by the European Committee of Post and Telegraph Administrations (CEPT) and is composed of two parts; A linear part for small signals and a logarithmic part for larger signals. The A-law codec is

$$f_A(x) = \begin{cases} \frac{Ax}{1+\ln A} & , |x| \leq \frac{1}{A} \\ \text{sign}(x)\frac{1+\ln(A|x|)}{1+\ln A} & , |x| > \frac{1}{A} \end{cases} \tag{3.14}$$

where A is the compression coefficient. The three codecs are plotted in Figure 3.11 with the compression coefficients set to 10.

The highest slope of the compression function gives the gain of the codec. Very small signals will be amplified by this gain thus giving a higher relative

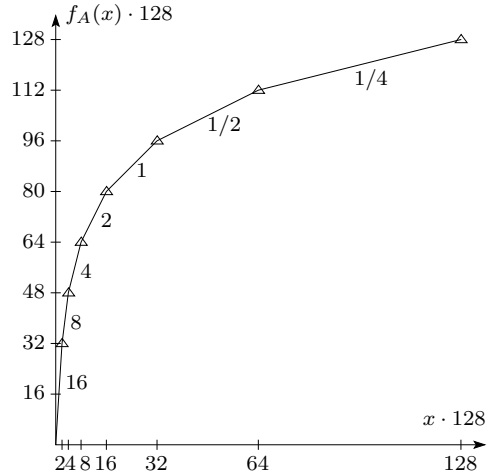

Figure 3.12: Piecewise linear approximation of the A-law scaled by a factor of 128. Only the positive half-plane is shown.

precision and extending the dynamic range. The smallest step the logarithmic ADC can take is $q/y'(0)$ ($q/y'(1/c)$ for the c-law). The standard compression coefficient for the μ-law is $\mu = 255$ and $A = 87.6$ for the A-law which gives the coding gains

$$f'_A(0) = 16 \text{ (24 dB)} \tag{3.15}$$

$$f'_\mu(0) = 47 \text{ (33 dB)} \tag{3.16}$$

The codecs can be difficult to implement directly with electronic circuits. To facilitate the physical implementation of the logarithmic functions, piecewise linear approximations of the codecs can be used. Then the codec is approximated using linear segments (chords). The length of the segments is successively halved while the slope of the segment is doubled. Each segment is divided into linear intervals (steps).

For pulse code modulation (PCM) of speech signals the piecewise-linear approximation of the μ-law and the A-law have been standardized. The piecewise-linear approximation of the A-law is shown in Figure 3.12. A more extensive formulation and synthesis of the segment companding of the μ-law and the A-law can be found in [19].

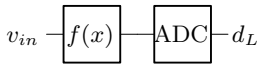

Figure 3.13: Analog logarithmic compression followed by an ADC.

3.3.1 Logarithmic ADC architectures

Various implementations of logarithmic ADCs have been published. Most of the implementations can be assigned to three categories [20]:

1. Analog compression ADC, an analog logarithmic circuit followed by a linear ADC.
2. Digital compression ADC, a high resolution ADC, followed by a digital compression.
3. A logarithmic ADC, where the ADC and the logarithmic function are merged. The logarithmic ADCs can be assigned to three sub-categories:
   (a) Successive approximation ADC based on a non-linear DAC.
   (b) $\Delta\Sigma$ ADC based on a non-linear DAC.
   (c) Pipelined ADC based on non-linear sub-DACs.

Analog compression ADC

The logarithmic ADC shown in Figure 3.13 uses an analog compression circuit in front of the ADC to implement the codec. The logarithmic compression is performed by the logarithmic relation between the base voltage and the collector current of a bipolar transistor. Care has to be taken in designing the analog compressor since it is very sensitive to parameter variations, like the temperature [21, 22].

Digital compression ADC

The principle of the high-resolution ADC, followed by a digital compression is shown in Figure 3.14. Here the signal is converted by a high-resolution ADC ($\Delta\Sigma$ [23, 24] and dual slope [25, 26]) and then compressed using a digital compressor. This solution is flexible since the codec can be implemented in digital domain but does not ease the accuracy requirements on the ADC.

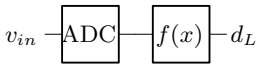

Figure 3.14: High resolution ADC followed by digital compression.

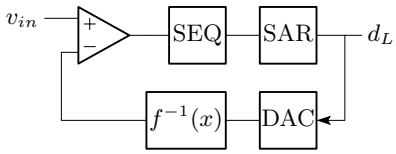

Figure 3.15: Successive approximation ADC based on a non-linear DAC.

Logarithmic successive approximation ADC

The principle of the successive approximation (SA) is to approximate the input signal with a digital signal. By making more and more accurate approximations of the input signal the value of the input signal can be obtained. The concept of a logarithmic SA is shown in Figure 3.15.

The approximation is controlled by the sequencer (SEQ). The sequencer controls the successive approximation register (SAR). The SAR contains the approximation of the input signal. The digital signal is compared to the input signal with help of the DAC. The digital sequence is made in such a way that the number of comparisons is minimized. The algorithm is called binary-search and is presented in table 3.1.

If the DAC is exponential then the digital output is the logarithm of the input. If the bit number $\rightarrow \infty$ then the approximation is exact, $v_{in} =$

1	The most significant bit is set in the SAR.
2	The ADC output, d_L, is compared to the signal.
3	If $\lvert v_{in} \rvert > v_{DAC}$ then the SEQ sets the actual bit to 'one', otherwise to 'zero'.
4	The next significant bit is set.
5	Repeat from point 2 until all bits are resolved.

Table 3.1: The binary-search algorithm for a successive approximation ADC.

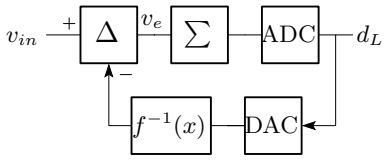

Figure 3.16: ΔΣ ADC based on a non-linear DAC.

$f^{-1}(d_L)$, thus

$$d_L = f(v_{in}) \tag{3.17}$$

The DAC in the SA-ADC can be implemented using several techniques.

1. Exponential DAC [21]. The expander function is implemented using the exponential relation of a bipolar transistor.

2. High resolution DAC [27]. The expander function is implemented in the SAR.

3. 2-step voltage-DAC. The longitudinal resistor in an R-2R-ladder is tapped to generate the linear segments [28]. First one longitudinal resistor is chosen (the segment). Then the step voltage is taken from a tap in the chosen longitudinal resistor.

4. 2-step current-DAC [29, 30, 31]. The current from binary weighted current sources are switched to a current-steering DAC. The segment determines how many of the current sources are directed to the output. The linear current-steering DAC determines the fraction of the current from the most significant current source (the step) to be used. The remaining current is dumped into a dummy load.

5. 2-step charge-redistribution-DAC [32, 33]. An array of binary weighted capacitors is used with a linear V-DAC. First the segment is chosen by adding binary weighted charges to the capacitor array (the segment). Then the output from a linear DAC is connected to one of the capacitors to add a linear charge to the capacitor array (the steps).

Logarithmic ΔΣ ADC

The principle of a ΔΣ ADC [34] is to make the quantization error v_e of a low-resolution ADC as small as possible by the use of negative-feedback. An

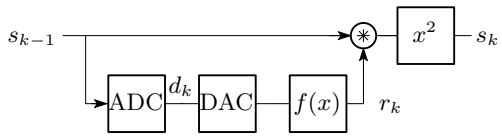

Figure 3.17: A logarithmic pipeline stage.

$\Delta\Sigma$ ADC is shown in Figure 3.16. The feedback in the circuit strives to set the quantization error v_e as close to zero as possible. The higher the gain is, the smaller the error becomes. The gain is provided by the integrating function that has an infinite gain when the frequency goes to zero. This way the quantization error spectrum is very small for low frequencies. This effect is called noise-shaping and if the digital signal is filtered a low noise (high accuracy) is obtained.

A $\Delta\Sigma$-ADC can be changed into a logarithmic ADC by changing the quantization step sizes in the DAC dynamically with respect to the magnitude of the signal [35, 36, 37].

Logarithmic pipelined ADC

A logarithmic ADC can also be realized by a pipelined ADC [38, 39, 40]. Instead of amplifying the signal and subtracting a reference, the logarithmic pipelined ADC performs the equivalent logarithmic functions, power and multiplication respectively. The meaning of this is described below.

The logarithmic pipeline stage can be seen in Figure 3.17. The transfer function of the logarithmic pipeline stage is

$$s_k = (s_{k-1} r_k)^2 \tag{3.18}$$

For p pipeline stages the output of the p:th logarithmic pipeline stage becomes

$$s_p = |v_{in}|^{2^p} \prod_{k=1}^{p} r_k^{2^{(p-k+1)}} \tag{3.19}$$

The logarithm of s_p is

$$\ln s_p = 2^p \ln |v_{in}| + \sum_{k=1}^{p} 2^{p-k+1} \ln r_k \tag{3.20}$$

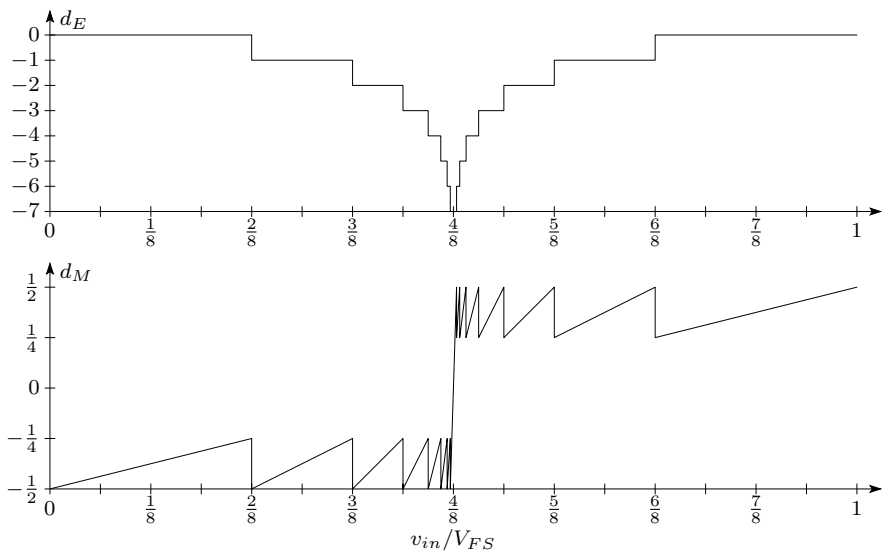

Figure 3.18: The exponent and the mantissa for a floating point ADC ($n_e = 3$, $n \to \infty$)

Solving for $\ln|v_{in}|$ and approximating gives

$$\ln|v_{in}| \approx \sum_{k=1}^{p} \frac{r'_k}{2^k} = d_L, \quad r'_k = 2\ln r_k \tag{3.21}$$

in the same way as in a linear pipelined ADC. This way the digital signal becomes the logarithm of the input signal.

3.4 Floating-point ADC

The equation for the digital value of a floating-point quantization (2.28) can be rewritten into

$$d = d_M 2^{d_E} + \frac{V_{FS}}{2} \tag{3.22}$$

where

$$d_E = w_E - 2^{n_e} + 1 \tag{3.23}$$

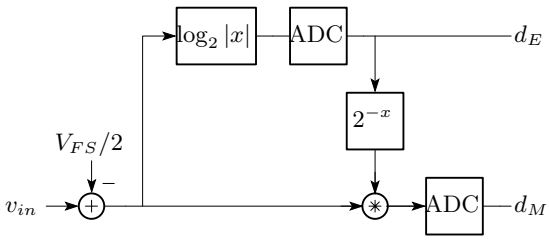

Figure 3.19: Block diagram of floating-point quantization.

$$d_M = w_M \cdot \frac{V_{FS}}{2^n} \tag{3.24}$$

A plot of the transfer functions from the input voltage v_{in} to the exponent and the mantissa is shown in Figure 3.18. Here the number of bits in the exponent is $n_e = 3$ and the number of bits in the mantissa is high, $n \to \infty$. The mantissa uses a binary representation that is able to represent negative numbers.

The mantissa is scaled by the exponent and therefore limited to $1/4 \leq |d_M| < 1/2$. According to [11] d_E and d_M in equation 3.22 can be solved

$$d_E = \left\lfloor \log_2 |d - \frac{V_{FS}}{2}| \right\rfloor + 1 \tag{3.25}$$

$$d_M = \frac{d}{2^{d_E}} \tag{3.26}$$

where $\lfloor \cdot \rfloor$ is the floor function.

Thus, one way to implement a floating-point ADC would be to resolve the exponent and then obtain the mantissa by normalizing the signal by a factor of 2^{d_E}. A block diagram of such a floating-point ADC is shown in Figure 3.19.

The notation used for the accuracy and dynamic range is $n+m$ FP-ADC. Where n is the number of bits in the linear ADC and m is the bit-equivalent of the extra dynamic range provided by the floating-point architecture

$$m = \max w_E \leq 2^{n_e} - 1 \tag{3.27}$$

3.4.1 Floating-point ADC architectures

The floating-point ADCs presented in literature can be assigned to four categories.

1. Logarithmic FP-ADC, the floating-point digital value is obtained by converting the output from a logarithmic ADC into a floating-point number.

2. 2-Step FP-ADC, the exponent and mantissa are determined in two steps. Fist step determines the exponent and the gain and the second the mantissa.

3. Distributed FP-ADC, the signal is amplified by a distributed amplifier, with outputs with different gains, and the exponent determines which amplifier output to be used for the mantissa.

4. Pipelined FP-ADC, the exponent and mantissa are determined in steps by pipeline stages.

Logarithmic FP-ADC

Another way to implement a floating-point ADC would be to use a logarithmic ADC and convert the compressed digital number into a floating-point notation [41].

The logarithm of the signal can, according to the c-law (equation 3.12), be expressed as

$$d_L = f_c(d) = \frac{\ln 2}{\ln c}\frac{\ln(|d|)}{\ln 2} + 1 = \frac{\ln 2}{\ln c}\log_2 |d| + 1 \tag{3.28}$$

By replacing the logarithm of the digital signal with the logarithmic digital signal the equations 3.25 and 3.26 can be rewritten as

$$d_E = \lfloor d_L \rfloor \tag{3.29}$$

$$d_M = \text{sign}(d - V_{FS}/2) \cdot 2^{d_L - d_E} \tag{3.30}$$

That is, the exponent is the integer part of d_L while the mantissa is two to the power of the fractional part of d_L. The sign of the mantissa has to be treated separately.

The procedure is equivalent for other codecs.

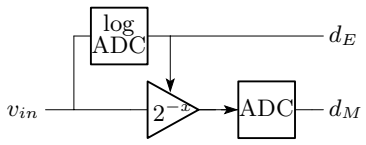

Figure 3.20: Block diagram of a 2-step FP-ADC.

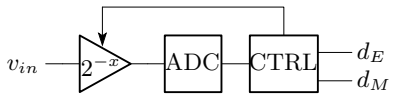

Figure 3.21: Block diagram of a 2-step FP-ADC using one linear ADC.

2-step FP-ADC

In a 2-step FP-ADC the floating-point number is determined as described by the equations 3.25 and 3.26. First the exponent is determined in the logarithmic ADC. The exponent controls the gain of an amplifier that normalizes the signal. The amplified signal is converted in the linear ADC which produces the mantissa [42, 43, 44]. The block diagram of a direct FP-ADC is shown in Figure 3.20.

In a variant of the 2-step FP-ADC both the exponent and the mantissa are determined by the same linear ADC [45, 46]. The functionality of the 2-step FP-ADC only using one linear ADC is shown by the block diagram in Figure 3.21.

First the gain of the amplifier is set to its lowest value and the signal is converted by the ADC. The controller determines the exponent from the ADC output and sets the amplifier gain accordingly. A consecutive conversion is performed by the ADC to resolve the mantissa. The effect of using the same ADC is that it requires two conversions for one sample, and the number of bits in the exponent is limited to

$$n_e \leq \log_2 n \tag{3.31}$$

Also reported [47, 48] are 2-step ADCs, where the exponent (thus the gain) and mantissa are determined by successive approximation much in the same way as logarithmic successive-approximation ADC.

Attempts have been made to predict the exponent in the next sample

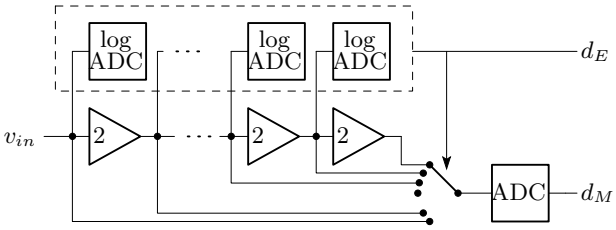

Figure 3.22: Block diagram of a distributed FP-ADC.

by an algorithm [49]. This way the conversion step of when the exponent is determined can be removed.

Distributed FP-ADC

In the distributed FP-ADC the mantissa gain is provided by a distributed amplifier. The gain of the input signal at the outputs of the distributed amplifier are binary weighted [50, 51, 52, 53, 54]

$$v_k = 2^k v_{in} \quad k = 0 \ldots 2^{n_e} \tag{3.32}$$

In the block diagram in Figure 3.22 the distributed amplifier is exemplified by a chain of amplifiers each having a gain of 2.

The functionality is as follows. The input signal is amplified by the distributed amplifier. The outputs of the distributed amplifier are measured by a logarithmic ADC. The ADC determines the mantissa and controls the switch to select the correct gain. The linear ADC converts the mantissa. The sub-ADCs in the distributed logarithmic ADC can be made identical and are very simple. They just have to determine whether the signal is

$$|v_k - \frac{V_{FS}}{2}| > \frac{V_{FS}}{4} \tag{3.33}$$

It is possible to lump the distributed logarithmic ADC and use it at the input, as in the direct FP-ADC, to determine the exponent and the switch position.

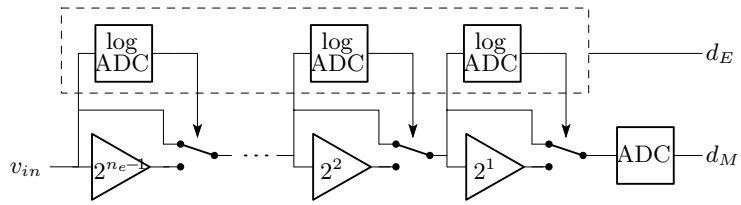

Figure 3.23: Block diagram of a pipelined FP-ADC.

Pipelined FP-ADC

In the pipelined FP-ADC the exponent and the gain of the amplifier are determined consecutively in pipeline stages [55].

The block diagram in Figure 3.23 shows the functionality of the pipelined FP-ADC. In the first pipeline stage the most significant bit (MSB) of the exponent is determined by the logarithmic ADC. If the MSB is set then the signal is amplified by 1, otherwise by$2^{2^{n_e-1}}$. In the next pipeline stage the second MSB is determined. This way the exponent and the gain are determined in binary steps until the complete exponent is resolved. The gain has been set, when the exponent was resolved, so the mantissa can be converted in the linear ADC.

3.4.2 Normalizing amplifiers for FP-ADC

In the FP-ADC architectures (except for logarithmic FP-ADC) an accurate amplifier is needed which can produce binary weighted gains. The amplifier can be assigned to three different categories:

1. Switched feedback amplifier [45, 43, 56]. The gain is set by switching the feedback ratio of a feedback amplifier.

2. Charge redistribution amplifier [48]. The signal is sampled on a set of unit capacitors. Depending on the selected gain a binary weighted number of capacitors are discharged into an integrator.

3. Distributed amplifier. The gain is set by several amplifiers with a static gain. By selecting the appropriate output the wanted gain is set.

   (a) Chained amplifier [50]. Several amplifiers with a gain of 2 are chained to produce the binary weighted gain.

(b) Tree amplifier. The gain is provided by an amplifier tree. An example is a binary tree, where each amplifier produces two gains 2^{n_e-k} or 1. $k = 0 \ldots 2^{n_e} - 1$ is the depth in the amplifier tree. A pipelined version is used in [55].

(c) Parallel amplifier [57, 52, 46]. The binary weighted gain is provided by a set of parallel amplifiers. A variant is to use binary weighted current mirrors to integrate the signal on several capacitors [54].

3.5 The proposed FP-ADC architecture

So far the placement of the sampling interface has not been addressed. If the sample-and-hold (SH) circuit is situated in the front of the FP-ADC it has to cope with the full dynamic range of the input signal. This means that the sampling capacitor has to be large (see equation 7.8, $n \leftarrow n+m$), which will have a large impact on both the speed and the power consumption. Associated with the sampling is not only the sample noise, but other errors are introduced as well, such as clock feed-through and charge injection from the sampling switch (see section 4.2).

To reduce the accuracy requirements of the sampling the SH circuit is placed after the normalizing amplifier and before the linear ADC. Here disturbances from the sample-and-hold will have a much smaller impact. The drawback is that now $m + 1$ instances of the SH are needed. The only floating-point ADC that can allow the SH into the architecture is the distributed FP-ADC.

The architecture for the FP-ADC proposed in this thesis, a distributed FP-ADC with embedded SH, is shown in Figure 3.24.

The distributed amplifier produces m amplified output signals, each with a gain of 2^k, where $k = 0 \ldots m$ denotes the row in the architecture. The distributed amplifier outputs are sampled simultaneously by a column of sample-and-hold stages. The absolute values of the sampled voltages $v_{s,k}$ are compared to a reference $V_R/2$ to determine which of the sampled voltages should be used for the mantissa conversion. The controller is made up by digital logic, represented here by XOR-gates, and operates as follows; The comparator k generates the blocking signals b_k if $|v_{in}2^k| > V_R/2$. The blocking signal sets the address signal a_k if the blocking signal b_{k-1} is unset

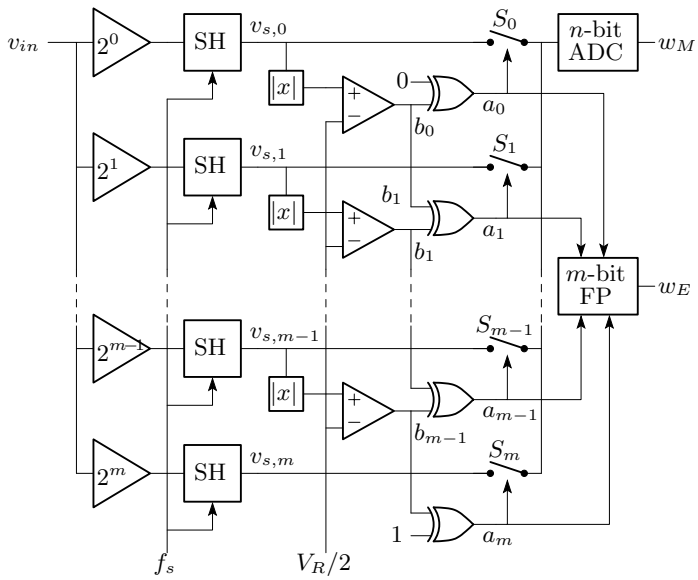

Figure 3.24: The proposed architecture for the FP-ADC, a distributed FP-ADC with embedded SH.

($|v_{in}2^{k-1}| < V_R/2$). The Boolean expression of the control function is

$$a_k = b_k \wedge \overline{b_{k-1}} \tag{3.34}$$

The control function is terminated by $b_m = 1$ and $b_0 = 0$ making sure that the signal is always switched to the n-bit ADC. The control function is

$$\begin{aligned} \frac{V_R}{2} \leq |v_{s,0}| &\Rightarrow a_0 = 1 \\ \frac{V_R}{2} \leq |v_{s,k}| < V_R &\Rightarrow a_k = 1, \quad \forall k = 2 \ldots m-1 \\ 0 \leq |v_{s,m}| < V_R &\Rightarrow a_m = 1 \end{aligned} \tag{3.35}$$

The controller uses the address codes to close the appropriate switch S_k. The mantissa w_M is determined by the n-bit linear ADC. The exponent is determined by converting the address codes $a_{0\ldots k}$ to the binary number w_E in the m-bit FP block.

Chapter 4

Analog Building Blocks

In this chapter some of the analog building blocks of the FP-ADC are presented and analyzed.

4.1 Switched-capacitor circuits

The pipeline stages in the pipelined ADC have been implemented using the switched-capacitor (SC) technique. The SC-technique exploits some of the strengths of CMOS technology, the ability to make good voltage switches and capacitors. The field of SC-techniques is vast [58, 59, 60] and in this section we only present the multiply-and-accumulate circuits that are needed in the pipeline stage [61].

An SC-circuit effectively works with charge redistribution. Multiplying a voltage by a constant factor can be implemented by transferring a charge from one capacitor C_1 to a smaller capacitor C_o. An addition can be made by transferring a charge from a third capacitor C_2 to C_o. A subtraction can be made by reversing the polarity of the charge transfer. This way the functions needed for the pipeline stage can be implemented.

For the charge redistribution switches are needed to connect the capacitors and for voltage gain an amplifier is needed as well. All-in-all there is a need for switches, capacitors and amplifiers in an SC-circuit.

An SC-circuit that can perform multiplication and addition is shown in Figure 4.1. The functionality is as follows. In the sampling phase ϕ_s the capacitors C_1 and C_2 are charged with the voltages $v_{n,s} - v_{1,s}$ and $v_{n,s} - v_{2,s}$

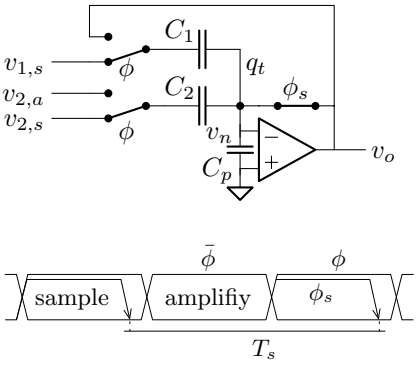

Figure 4.1: Multiply-and-add SC-circuit with the corresponding clock phases below. The arrows indicate the sampling instants.

and the parasitic capacitance C_p is charged with $v_{n,s}$, the voltage on the negative input of the amplifier in the sampling phase. The total charge on the capacitors in the sampling phase is

$$q_{t,s} = (v_{n,s} - v_{1,s})C_1 + (v_{n,s} - v_{2,s})C_1 + v_{n,s}C_p \tag{4.1}$$

When the clock phase ϕ_s goes low, the switch in the feedback path opens and now the node v_n floats. This means that the total charge on the capacitors q_t is preserved until ϕ_s goes high again since the net current (charge) through the node is zero. Then the other switches are changed, when ϕ goes low. Now in the amplifying phase the total charge on the capacitors is

$$q_{t,a} = (v_{n,a} - v_o)C_1 + (v_{n,a} - v_{2,a})C_1 + v_{n,a}C_p \tag{4.2}$$

As stated above, the charge is preserved between the sampling and the amplifying phase $q_{t,s} = q_{t,a}$. Solving v_o gives the transfer function from the input to the output

$$v_0 = v_{1,s} + \frac{C_2}{C_1}(v_{2,s} - v_{2,a}) + (v_{n,a} - v_{n,s})\frac{C_1 + C_2 + C_p}{C_1} \tag{4.3}$$

By setting $C_1 = C_2 = C_s/2$, $v_{1,s} = v_{2,s} = s_{k-1}$, $v_0 = s_k$ and $v_{2,a} = r_k$ a radix-2 pipeline stage can be implemented. If the amplifier is ideal ($v_{n,a} =$

$v_{n,s} = 0$, infinite gain and no noise) then the transfer function becomes

$$v_k = 2s_{k-1} - r_k \tag{4.4}$$

as per equation 3.3.

An SC-circuit can be seen as a sampling circuit. The major difference between the passive sampler and the SC-circuit in terms of sampling is that it is the complete SC-circuit, including the amplifier, which sets the bandwidth and the settling time.

4.1.1 Noise and correlated double-sampling

Simple techniques exist for the cancellation of the DC offset in the amplifiers, auto-zero (AZ) techniques. These operations are equivalent to subtracting the instantaneous values of the input noise of the amplifier at the times kT_s and $kT_s - T_s/2$ (see $v_{n,a} - v_{n,s}$ in equation 4.3). This operation is named correlated double sampling (CDS). The transfer function of the noise becomes [62] $H_{CDS}(z) = 1 - z^{-1/2}$ in the z-domain or, in the frequency domain

$$H_{CDS}(j\omega T) = 1 - e^{-j\omega T/2} = 2je^{-j\omega T/4} \sin\left(\frac{\omega T_s}{4}\right) \tag{4.5}$$

where $T_s = 1/f_s$. The spectrum of the noise will be multiplied by the absolute value squared

$$\left|H_{CDS}(j\frac{2\pi f}{f_s})\right|^2 = 4\sin^2\left(\frac{\pi f}{2f_s}\right) \tag{4.6}$$

Now let us investigate the effect of CDS on the spectrum for the thermal and the flicker noise. The plots of the spectrum are shown in Figure 4.2. It is evident that the noise is eliminated at DC and every $2f_s, 4f_s, \ldots$ CDS is very effective in eliminating the noise at low frequencies. The flicker noise is $S_{fl}(f) \sim 1/f$ while $|H_{CDS}(f)|^2 \sim f^2$ for low frequencies, why the noise spectrum at DC, $S_{fl}|H_{CDS}|^2(0) \to 0$. However when the signal is sampled, the noise spectrum is folded at $f_s/2$, meaning that some noise will be folded to DC. But compared to the original flicker noise this DC noise energy is limited.

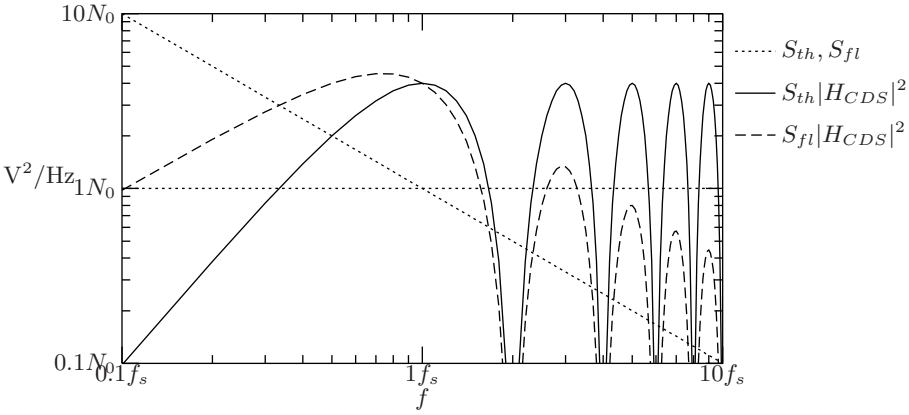

Figure 4.2: The effect of CDS on the noise spectrum, $S_{th} = N_0$ and $S_{fl} = N_0/f$. The plot is normalized to N_0 and f_s.

The integrated thermal noise over the box bandwidth B is

$$\bar{v}_{n,th}^2 = N_0 \int_0^B \left| H_{CDS}(j\frac{2\pi f}{f_s}) \right|^2 df = 2N_0 B + 2N_0 \frac{f_s}{\pi} \sin\left(\frac{B\pi}{f_s}\right) \tag{4.7}$$

where N_0 is the spectral density of the noise source. The thermal noise energy can be approximated to

$$\bar{v}_{n,th}^2 \to 2N_0 B, \quad B \to \infty$$

For the case of sampling circuits this is a good assumption since the bandwidth needs to be much larger than the sampling frequency. This means that effectively the energy of the noise from the amplifier is doubled for CDS compared to if no AZ was employed.

The integrated flicker noise over the bandwidth B is

$$\bar{v}_{n,fl}^2 = N_0 \int_0^B \frac{f_f}{f} \left| H_{CDS}(j\frac{2\pi f}{f_s}) \right|^2 df = 2N_0 f_f \left[\gamma_{\text{euler}} - \text{ci}\left(\frac{2\pi B}{f_s}\right) + \ln\left(\frac{2\pi B}{f_s}\right) \right] \tag{4.8}$$

where $\gamma_{\text{euler}} \approx 0.577$ and $\text{ci}(x) = \int_x^\infty \frac{\cos t}{t} dt$.

A plot of the integrated CDS noise, equations 4.7 and 4.8, can be found in Figure 4.3.

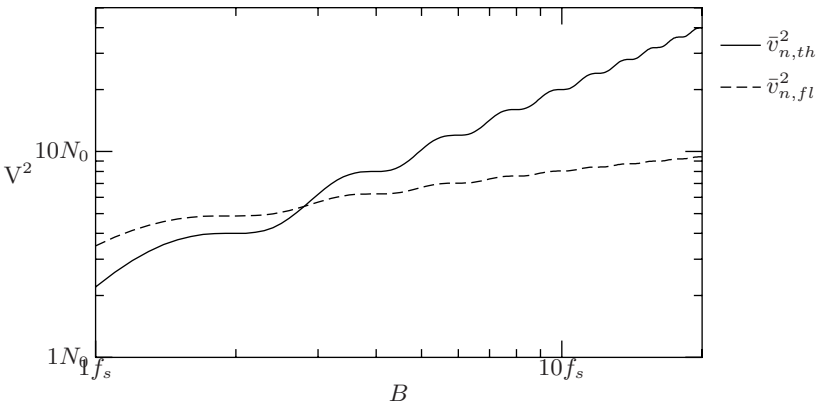

Figure 4.3: The integrated thermal and flicker noise for CDS, $S_{th} = N_0$ and $S_{fl} = N_0/f$. The plots are normalized to N_0 and f_s respectively.

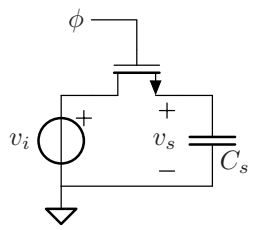

Figure 4.4: The NMOS switch.

4.2 Switches

The SC-circuits are depending on switching charges to and from capacitors. In CMOS technology these switches are implemented by transistors. The properties of the switches are examined below.

4.2.1 The basic NMOS switch

The simplest switch is the NMOS switch which is shown in Figure 4.4. In the sampling phase when ϕ is high the NMOS transistor is in the triode region and therefore behaves like a resistor and a closed switch. When ϕ is low the transistor is in the cutoff region and behaves like an open circuit.

4.2.2 Non-ideal effects in the NMOS switch

The NMOS switch uses an NMOS transistor for switching. However the NMOS transistor is not an ideal switch. It will show a capacitive load to the clock line, both when the switch is open and closed. When the clock voltage is changing a charge will be pulled from the sampling node so a charge error in the sampled voltage will be created. Also the resistance of the NMOS switch is non-linear and will change with the input voltage.

All of these effects can be derived from the transistor model (see appendix B) and are described below. A general recommendation to designers is to use minimal length of the transistors to minimize the non-ideal effects.

Capacitive load

The transistor switch operates in two regions, the triode region when it is turned on and the cutoff region when turned off. The capacitive load from the switches will load the circuit and produce a parasitic path from the clock into the signal path.

The on-capacitance for the switch is

$$C_{ON} = C_S + C_D = 2WC_{ov} + WLC_{ox} + (A_S + A_D)C_j(v_{SB}) \tag{4.9}$$

The off-capacitance is

$$C_{OFF} = C_S = WC_{ov} + A_S C_j(v_{SB}) \tag{4.10}$$

The on-capacitance includes the gate oxide that gives a high parasitic capacitance to the clock line.

Charge error

When the clock voltage is falling, charge is transferred via capacitive dividing from the clock line and from the inversion layer (gate capacitance) of the transistor itself onto the sampling node. This charge will appear as an offset voltage and a gain error on the input signal.

The charge error can be divided into two parts of the falling clock edge depending on the operating region of the transistor; 1) the inversion region $\phi > v_i + V_T$ and 2) cutoff region $\phi < v_i + V_T$.

1. In the first part when the clock signal is larger than the threshold voltage the capacitance is the sum of the overlapping capacitance and

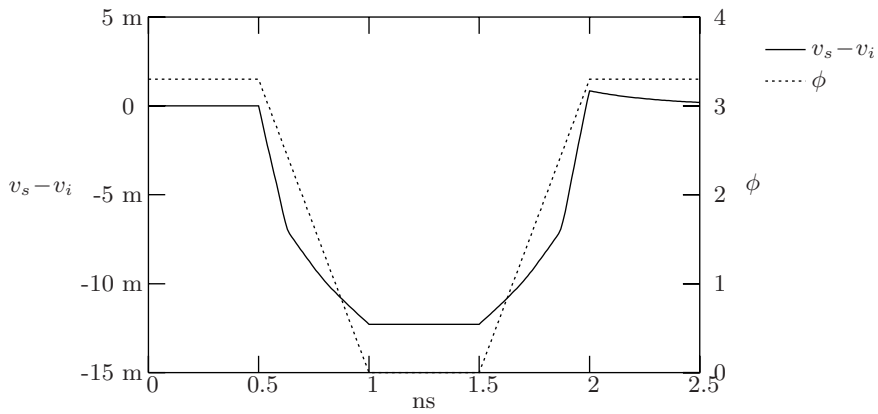

Figure 4.5: Simulation of the charge error of an NMOS switch. The unit on the left and right y-axis is V.

half the gate capacitance. The charge error contribution will be

$$q_1 = -(WC_{ov} + \frac{WL}{2}C_{ox})(V_{DD} - v_i - V_T) \tag{4.11}$$

2. In the second part the inversion layer is gone and the capacitance only consists of the overlapping capacitance. The charge injected in the cutoff region is

$$q_2 = -WC_{ov}(v_i + V_T) \tag{4.12}$$

In this model we assume that the total error charge is evenly distributed between source and drain. This is true under the assumption that the clock edge is steep [63, 64]. The assumption will overestimate the charge error. Further the gate capacitance is not really linear, but for our purposes a linear approximation will be satisfactory [64].

The total charge error on the sampling capacitor becomes

$$q_s = -\frac{WLC_{ox}}{2}(V_{DD} - v_i - V_T) - WC_{ov}V_{DD} \tag{4.13}$$

where the first term is the charge from the channel and the second term is the charge from the overlapping capacitance. They are referred to as the charge injection and the clock feed-through respectively.

Looking at equation 4.13 we can see a linear dependence on v_i plus constant terms giving a gain and a DC error in the sampled voltage.

Example An NMOS transistor is used as a sampling switch for a 500 fF capacitor. The input voltage is 1.6 V. What will be the error voltage on the capacitor? The clock voltage is falling from 3.3 V to 0 V. The transistor parameters are as following: V_T=0.8 V, W=5 μm, L=0.35 μm, C_{OX}=4.3 fF/μm^2, C_{OV} = 0.22 fF/μm

1. 3.3 V > ϕ > 2.4 V:

$$\Delta v_{S1} \approx \frac{q_1}{C_S} = \frac{WC_{OV} + \frac{WL}{2}C_{OX}}{C_S}(V_{DD} - v_i - V_T) = \frac{1.1 + 3.8}{500}0.9 = 8.8\,\text{mV}$$

2. 2.4 V > ϕ > 0 V:

$$\Delta v_{S2} \approx \frac{q_2}{C_s} = \frac{WC_{OV}}{C_S}(v_i - V_T) = \frac{1.1}{500}2.4 = 5.3\,\text{mV}$$

The charge feed-through error voltage will be $\Delta v_{S1} + \Delta v_{S2}$= 14 mV. A plot of the simulated waveforms for the example is shown in Figure 4.5. The simulator gives an offset error of 12 mV which is 14% lower than calculated.

Non-linear switch conductance

The conductance of the switch is depending on the input voltage, the clock voltage and the threshold voltage. The conductance for a switch is mainly linearly dependent on the gate voltage in the triode region and zero when the gate voltage is below the threshold voltage. The conductance of a single switch can be approximated [60] to

$$g_{on} = \begin{cases} \mu C_{ox}\frac{W}{L}(V_{DD} - v_i - V_T) & , v_i < V_{DD} - V_T \\ 0 & , v_i > V_{DD} - V_T \end{cases} \tag{4.14}$$

$$V_T = V_{T0} + \gamma(\sqrt{2\phi_F + v_i} - \sqrt{2\phi_F}) \tag{4.15}$$

If $v_i > V_{DD} - V_T$ for an NMOS switch the conductance will be zero and the switch is open. By using a complementary transistor this can be avoided. This means that a transmission gate allows rail-to-rail input voltages.

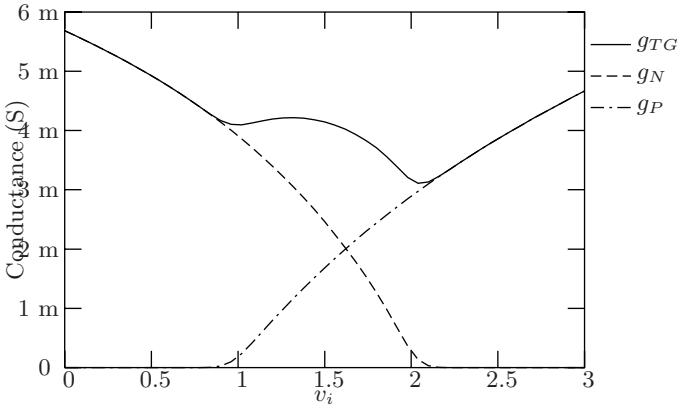

Figure 4.6: Simulation of conductance of switches. The PMOS transistor is three times wider than the NMOS transistor.

Further the back-gate voltage adds to the non-linear behavior. The bulk connections are connected to the supply lines, thus there is an increase in the threshold voltage from the nominal V_{T0}. A bigger problem arises when the supply voltage goes below $V_{Tn} - V_{Tp}$. Then neither the NMOS nor the PMOS will conduct in the midrange so the transmission gate is always open. A solution might be to use a higher clock voltage for the switches, clock-boosted switches, or to reduce the input voltage of the switch, bottom-plate sampling.

A simulation of the switch conductance for both an NMOS and a PMOS transistor is shown in Figure 4.6, where the conductance is plotted against the input voltage. As seen in Figure 4.6, the conductance is nonlinear and strongly dependent on the input voltage. For large signals this will introduce harmonics to the system, but in most cases in SC-circuits this effect is small and can be neglected, since the resistance of the switch is of minor importance as soon as the signal has settled.

Input dependent jitter

Another issue connected to the nonlinear conductance is the input dependent jitter. Jitter is a major problem when sampling high frequency signals. The input dependent jitter arises from the fact that the clock edge has a finite

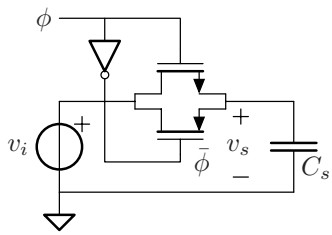

Figure 4.7: A transmission gate.

slew-rate. The switch turns off at a voltage dependent of the input voltage

$$\phi = v_{OFF} = v_i + V_T \tag{4.16}$$

Because of the finite slew-rate of the clock signal the sampling instant shifts with the input voltage. This jitter will show up as harmonic distortion.

4.2.3 More complex switches

In this section more complex switches are introduced. They are used to reduce the non-ideal effects of the NMOS switch.

The transmission gate

To improve the conductance from the NMOS switch, especially for high input voltages, where the NMOS transistor will eventually switch off, one can place a complementary (PMOS) transistor in parallel. Then we have the complementary switch, called a transmission gate. The transmission gate is the workhorse of switches and transmits rail-to-rail voltages. A transmission gate is shown in Figure 4.7.

If the width of the transistors are the same then the charge errors from both transistors will cancel each other and there will be a small residue remaining from the charge injection – mainly from the fact that the threshold voltages are not equal. Unfortunately, the PMOS transistor needs to be around three times wider than the NMOS switch to give the same conductance.

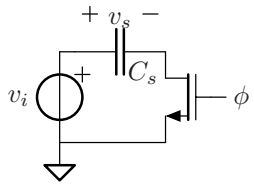

Figure 4.8: Bottom plate sampling for a passive sampler.

Bottom-plate sampling

Both the charge error and the non-linear conductance are dependent on the voltages across the transistor switch. By using bottom-plate sampling the voltages on the transistor terminals are independent of the input signal, thus the charge errors and the conductance are constant as well.

For the passive sampler bottom-plate sampling is about changing the order of the switch and the sampling capacitor. Bottom-plate sampling is illustrated in Figure 4.8. Here the sampling capacitor is connected to the source while the switch is connected to the signal ground. It does not matter for the sampled voltage whether the switch is connected to the input voltage or to the ground, but it is important considering the errors from the switch. Both the charge error and the conductance are affected by the order of the switch and the sampling capacitor.

Since the source and drain voltages of the transistor are always at ground voltage the charge feed-through is constant from sample to sample. The charge errors will only appear as a DC error. The constant voltage on the source also gives a constant conductance of the transistor, independent of the input voltage.

When looking at SC-circuits the bottom-plate sampling can be used as well. Bottom-plate sampling in an SC-circuit is not a special way to connect the switch and the sampling switch. It is about in what order you switch off the switches in the sampling phase.

The trick is to switch off the switch connected to the sampling node — the negative input of the amplifier. This switch is the actual sampling switch. in Figure 4.1 it is the switch in the feedback path driven by ϕ_s. The sampling is made at that instant the sampling switch is opened. After the sampling switch is opened the charge is preserved on the capacitors at the negative input of the amplifier. Any other switching activity will not affect

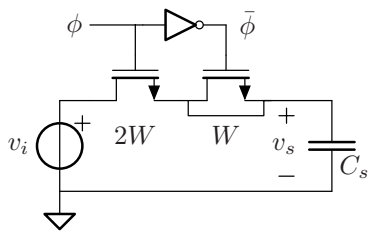

Figure 4.9: A dummy switch.

the amount of charge stored at the sampling instant. They just redistribute the charge on the capacitors connected to the sampling node.

Further, the sampling switch is located at virtual ground during the sampling phase. There is a constant voltage at the drain and the source of the switch transistor which is independent of the input signal. This is very important as we want the non-ideal behavior of the switch to be independent of the input signal.

Dummy switch

The main idea to use a dummy switch, is to cancel the charge injected from the main switch with a half size transistor driven by the inverse clock. If the transistors match there will be only a fraction of the charge error left on the sampling capacitor. The fraction left depends on the amount of second order effects, like the uneven distribution of charge from the main switch. The concept of the dummy switch is shown in Figure 4.9.

The dummy switch is on after the sampling instant so there will be significant increase of the parasitic coupling between the sampling node and the clock. This effect will have a negative effect on the PSRR.

Differential switch

In differential switches all even order distortion will be cancelled [65]. This means that the DC error will appear as a common-mode voltage and will cancel each other. However the linear and the odd order terms of the errors remain.

When differential sampling switches are used it is important to minimize the charge errors. One might be tempted to use dummy switches. Contrarily

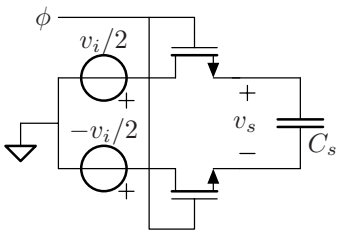

Figure 4.10: Differential switch

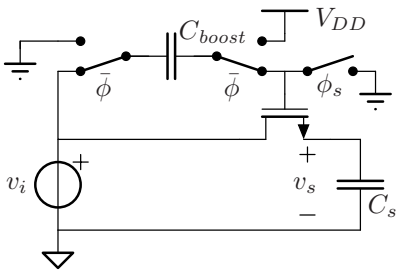

Figure 4.11: Schematic of a clock-boosted switch

the charge feed-through mismatch will actually increase. Because then the mismatch is between two pairs of transistors instead two transistors. Also the dummy switch adds parasitic capacitance to the sample node.

Clock-boosted switch

One way to remedy both the nonlinear conductance and low supply voltages is to use clock-boosted switches. Here the gate voltage is kept constant by a capacitor C_{boost} charged with the supply voltage. The clock-boosted switch is shown schematically in Figure 4.11. In the holding phase C_{boost} is charged by switching it between V_{DD} and ground. Then, in the sampling phase C_{boost} is coupled across the drain (or the source) and the gate of the switching transistor. Now v_{gs} is kept steady, regardless of the input voltage. This way the linearity and conductance are improved a lot. However the back-gate voltage (v_{bs}) will still vary with the input voltage which introduces a nonlinear behavior.

Another drawback is that the voltage between some nodes can become

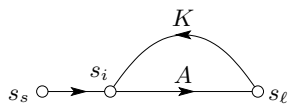

Figure 4.12: Black's feedback model.

very high and exceed the breakdown voltages of the transistors, thus destroying them. Great care has to be taken to construct clock-boosted switches. Some solutions to the problems described above have been proposed in [66].

4.3 Amplifiers in a time-discrete environment

The design of negative-feedback amplifiers is a wide engineering area. The amplifier is the heart of analog circuits and its design needs a lot of care. This thesis has no intentions of being an exhaustive source for amplifier design. There exists lots of literature for the interested reader [67, 62, 68].

Instead the focus has been set upon specific parts of amplifier design targeted for time-discrete environments like SC-circuits. Therefore, in this section a brief summary of amplifier modeling is given, followed by an investigation of the accuracy and the step response of feedback amplifiers.

4.3.1 Amplifier models

In this section amplifier models according to systematized amplifier design are presented [69, 70, 71, 65, 72].

The most elementary feedback model is the Black's feedback model. This model is not accurate enough so the superposition and asymptotic gain model are used instead.

Black's model

The elementary feedback model, Black's feedback model [73], is depicted in Figure 4.12. The input signal s_s is compared to the output signal s_ℓ damped by the feedback factor K. The input difference s_i is amplified by the gain factor A to the output. A and K form a feedback loop, where the transfer function is

$$G = \frac{s_\ell}{s_s} = \frac{A}{1 - AK} \tag{4.17}$$

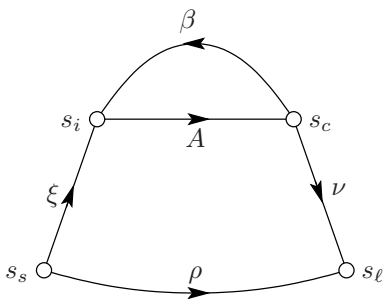

Figure 4.13: The superposition model

If the gain is high then the transfer of the feedback amplifier becomes the inverse of the feedback factor

$$G_\infty = \lim_{A\to\infty} A_t = -\frac{1}{K} \tag{4.18}$$

where G_∞ is the asymptotic gain. The loop gain AK must be negative for negative feedback. An amplifier with positive feedback will be unstable.

Black's model is too simple. It assumes that the feedback circuit is unilateral. But if the feedback circuit consists of passive (bilateral) components this assumption does not hold. A more suitable amplifier model is the superposition/asymptotic gain model.

The superposition model

Another way to model a feedback amplifier is to use the superposition model [69]. The flow-graph of the superposition model is shown in Figure 4.13. Here we divide the amplifier into the unilateral gain factor, denoted by A, and the feedback factor, denoted by β. The boundaries between the feedback factor and the gain factor are the input signal s_i and the controlled signal s_c.

The transfer from the source to the input is ξ and the transfer from the controlled signal to the load is ν. We also need to define a direct transfer ρ from the source to the load due to the bilateral nature of the feedback network. We can write the superposition model as an equation

$$\begin{cases} s_\ell = \rho s_s + \nu s_c \\ s_i = \xi s_s + \beta s_c \end{cases} \tag{4.19}$$

where the relation between s_i and s_c is the gain factor A

$$s_c = As_i \tag{4.20}$$

Now the transfer function of the amplifier can be found by solving (4.19) and (4.20) for $G = s_\ell/s_i$

$$G = \frac{\nu\xi A}{1 - A\beta} + \rho \tag{4.21}$$

The factor $A\beta$ is the loop-gain.

The asymptotic gain model

The asymptotic gain model [69] is an extension of the superposition model.

Let $A \to \infty$, then the transfer of the amplifier is

$$G_\infty = \lim_{A\to\infty} G = -\frac{\xi\nu}{\beta} + \rho \tag{4.22}$$

The asymptotic gain is the ideal transfer function of the amplifier[1]. It is easily computed by using a circuit element with infinite gain and bandwidth, like the nullor, as the amplifying device. Later when the amplifying device is realized by real components neither the gain nor the bandwidth will be infinite, thus there will be a discrepancy from the ideal transfer function.

By substituting (4.22) in (4.21) we find the transfer function

$$G = G_\infty \frac{-A\beta}{1 - A\beta} + \frac{\rho}{1 - A\beta} \tag{4.23}$$

The factor $-A\beta/(1 - A\beta)$ is called the discrepancy factor. The discrepancy factor describes all the non-ideal effects like finite gain and speed of the amplifier implementation. Thus the discrepancy factor is important when designing feedback amplifiers. The second term $-\rho/(1 - A\beta)$ is the direct transfer term. For most practical cases (high loop-gain) this term is insignificant and can be neglected.

A benefit of using the asymptotic gain model before the superposition model is that only G_∞, A, and β needs to be calculated. Therefore the asymptotic gain model is simple to use when analyzing a feedback amplifier.

[1] If $A \to \infty$ then Black's and the asymptotic gain model are converging, meaning that $1/K = \xi\nu/\beta - \rho$. Now it is apparent that $K \neq \beta$, to clear away a common misunderstanding.

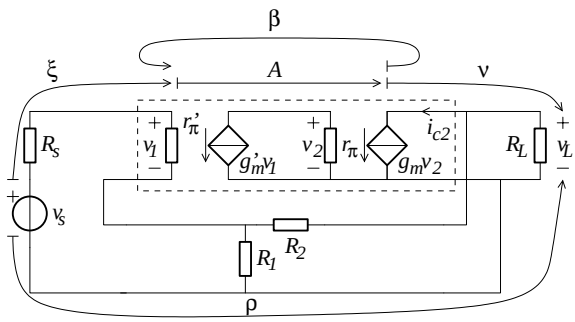

Figure 4.14: The superposition model applied on a two-stage voltage amplifier.

Example in Figure 4.14 the small signal schematic of a two-stage voltage amplifier is shown. The input stage is an anti-series common-emitter[2] (ASCS) stage followed by a common-emitter (CS) stage. The input signal is the voltages source $s_s \leftarrow v_s$ and the output signal is the voltage across the load resistor $s_\ell \leftarrow v_L$. One is free to choose the interfaces of the input and the controlled signal, but the natural positions are the voltage at the input stage $s_i \leftarrow v_1$ and the current of the output stage of the amplifier $s_c \leftarrow i_{c2}$. Then G_∞, A, and β are

$$G_\infty = 1 + \frac{R_2}{R_1} \tag{4.24}$$

$$A = g'_m r_\pi g_m \tag{4.25}$$

$$\beta = -\frac{R_L}{R_L + R_2 + R_1 \| (r'_\pi + R_s)} \frac{R_1 r'_\pi}{R_1 + r'_\pi + R_s} \tag{4.26}$$

4.3.2 Required gain

In chapter 5 the harmonic distortion was calculated for a specific gain error in the pipeline stages of a pipelined ADC. From equation 5.2 the non-ideal gain in a pipeline stage can be expressed as $G(1+e)$, where G is the wanted gain and e is the gain error. In the asymptotic gain model the asymptotic gain is the wanted gain while the discrepancy factor is the gain error. The

[2]Also known as a differential stage

two different expressions of the gain error are the same

$$1 + e = \frac{-A\beta(0)}{1 - A\beta(0)} \tag{4.27}$$

The loop gain corresponding to a specific gain can be solved in the above equation

$$A\beta(0) = 1 + \frac{1}{e} \tag{4.28}$$

Example You are designing 1-bit RSD pipeline stages for a 10-bit pipelined ADC. You want to know what loop gain to your amplifier needs to get a total harmonic distortion (THD) lower than -60 dB.
If the pipeline stages are identical then equation 5.21 can be used to express the loop gain in terms of the THD

$$A\beta(0) = 1 - \frac{0.34}{\text{THD}} \tag{4.29}$$

$$A\beta(0) = 1 - \frac{0.34}{10^{-3}} = -340\,(51\,\text{dB}) \tag{4.30}$$

4.3.3 Settling time

The settling time is the time required for a signal to settle to a specific accuracy. The settling time for a passive sampler is of first order and trivial to solve. However for an SC-circuit the whole circuit needs to settle before a sample can be taken. then we need to know the transfer function of the amplifier.

The step response is modeled as either a first or a second order system since many amplifiers are designed to have one or two dominant poles. The poles are normalized in such a way that the -3 dB bandwidth ω_0 of the amplifier is the same for all cases.

If we use the asymptotic gain model then the ideal behavior is described by the asymptotic gain while the discrepancy factor describes all the non-ideal effects emerging from the finite gain and the finite speed of the amplifier implementation.

The discrepancy factor gives us the frequency response of the amplifier. For simplicity the frequency response of the discrepancy factor is mapped

on the transfer function

$$H(\omega) = \frac{1}{(1 + \omega/p_1)} \tag{4.31}$$

for a first order system, or mapped on

$$H(\omega) = \frac{1}{(1 + \omega/p_1)(1 + \omega/p_2)} \tag{4.32}$$

for a second order system. Notice that the discrepancy factor at DC is normalized to 1.

First order system

The step response for a first order system is

$$v_s(t) = 1 - \mathrm{e}^{p_1 t} \tag{4.33}$$

Solving the settling time for a first order system is trivial

$$t_{s1} = -\frac{\ln \epsilon}{\omega_1} \tag{4.34}$$

Here, $\omega_1 = -p_1$ and ϵ is the residual error.

Second order system

The step response of a second order system is

$$v_s(t) = \begin{cases} 1 - \frac{p_1 \mathrm{e}^{p_2 t} - p_2 \mathrm{e}^{p_1 t}}{p_1 - p_2} & , \quad p_1 \neq p_2 \\ 1 - \mathrm{e}^{p_1 t} + p_1 t \mathrm{e}^{p_1 t} & , \quad p_1 = p_2 \end{cases} \tag{4.35}$$

For a second order system we can distinguish three solutions depending on the pole positions. The poles are arranged in such a way that the -3 dB bandwidth $\omega_2 = \sqrt{p_1 p_2}$ is the same for the three solutions.

1. For real poles, $p_1 = -\omega_{2r}/k$, $p_2 = -\omega_{2r} \cdot k$, the step response is

$$v_s(t) = 1 - \frac{k^2 \mathrm{e}^{-\frac{\omega_{2r}}{k} t} - \mathrm{e}^{-k\omega_{2r} t}}{k^2 - 1} \tag{4.36}$$

When some time has elapsed the second term in the numerator in

(4.36) will be negligible, if $k > 1$. Then it is possible to solve the settling time

$$t_{s2r} = \frac{-\ln\epsilon + \ln\frac{k^2}{k^2-1}}{\omega_{2r}/k},\ k > 1 \tag{4.37}$$

2. If the poles are complex, $p_{1,2} = -\omega_{2c}\mathrm{e}^{\pm\mathrm{i}\varphi}$, the step response is

$$v_s(t) = 1 - \mathrm{e}^{-\omega_{2c}t\cos\theta}\sqrt{1+\cot^2\varphi}\cdot\sin(\omega_{2c}t\sin\varphi + \varphi) \tag{4.38}$$

The problem with solving the step response is that the signal is ringing. Then the signal may pass the residual window several times. The settling time can be approximated by using the envelope of the decay factor in (4.38)

$$t_{s2c} \leq \frac{-\ln\epsilon + \ln\sqrt{1+\cot^2\varphi}}{\omega_{2c}\cos\varphi} \tag{4.39}$$

If the maximum overshoot $1 + \mathrm{e}^{-\pi\tan\varphi} = 1 + \epsilon$ then the settling time is the shortest. The optimal φ can be calculated to

$$\varphi = -\arctan(\pi/\ln\epsilon) \tag{4.40}$$

The time when (4.38) first crosses 1 is larger than t_s. This happens when $\omega_{2c}t\sin\varphi + \varphi = \pi$. then it is possible to bound the settling time to

$$t_{s2c} < \frac{\pi - \varphi}{\omega_{2c}\cdot\sin\varphi} \tag{4.41}$$

The settling time of a complex double pole system is shown in Figure 4.15.

3. For a double-pole system, $p_1 = p_2 = -\omega_{2d}$, the step response is

$$v_s = 1 - (1 + \omega_{2d}t)\mathrm{e}^{-\omega_{2dp}t} \tag{4.42}$$

The settling time of a double-pole system cannot be solved analytically why an approximation is proposed. To solve this equation iteration is used. Equation 4.42 is rearranged into

$$t_{s2d,k+1} = \frac{-\ln(\epsilon) + \ln(1 + \omega_{2dp}t_{s,k})}{\omega_{2d}} \tag{4.43}$$

After inspecting the step response equations 4.37 and 4.39 an approx-

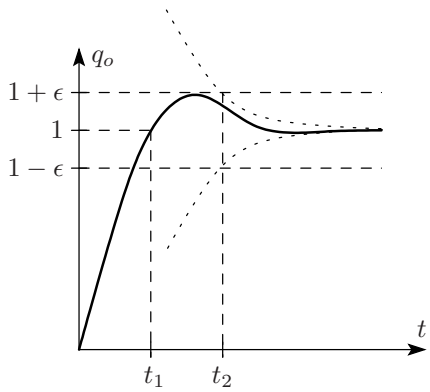

Figure 4.15: Illustration of the step response complex poles step response, t_1 corresponds to equation 4.41 and t_2 to equation 4.39.

imation of the initial value is attempted

$$t_{s2d,0} = \frac{-v \cdot \ln(\epsilon) + w}{\omega_{2d}} \tag{4.44}$$

Simulation and least square error fitting gave the parameters v and w to be 1.10 and 1.56. The fitting was made in the region $\epsilon \in [5 \cdot 10^{-7}, 10^{-1}]$, which corresponds to an error of one half LSB, for $n \in [2, 20]$.

4. Now the first iteration in (4.43) gives

$$t_{s2d,1} = \frac{-\ln(\epsilon) + \ln(1 - v \cdot \ln(\epsilon) + w)}{\omega_{2d}} \tag{4.45}$$

Simulations and conclusions

Simulations in MATLAB were made to check the validity and behavior of the equations. The plot in Figure 4.16 displays the settling time for various residual errors versus the sum of the poles[3]. in Figure 4.17 a plot is shown for the settling time versus ϵ. In both figures the simulations are compared with the equations (dashed lines). The correspondence between the equations and the simulations is high, in some cases the differences are hardly visible.

Now we can draw the conclusions:

[3]Note that the quality factor of a second-order system is $Q = -\omega_2/(p_1 + p_2)$

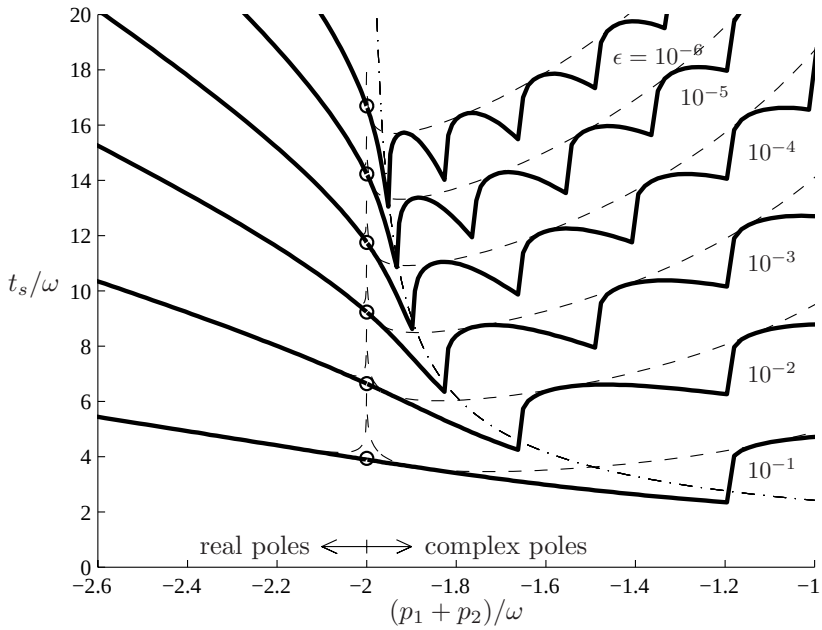

Figure 4.16: Plot of t_s versus $(p_1+p_2)/\omega_2$ for different values of ϵ. The solid line is the settling time from simulations of the step response. The dashed lines represent equation 4.37 left of $(p_1+p_2)/\omega_2 = -2$ and 4.39 to the right. The dash-dot line is equation 4.41 and the circles equation 4.45

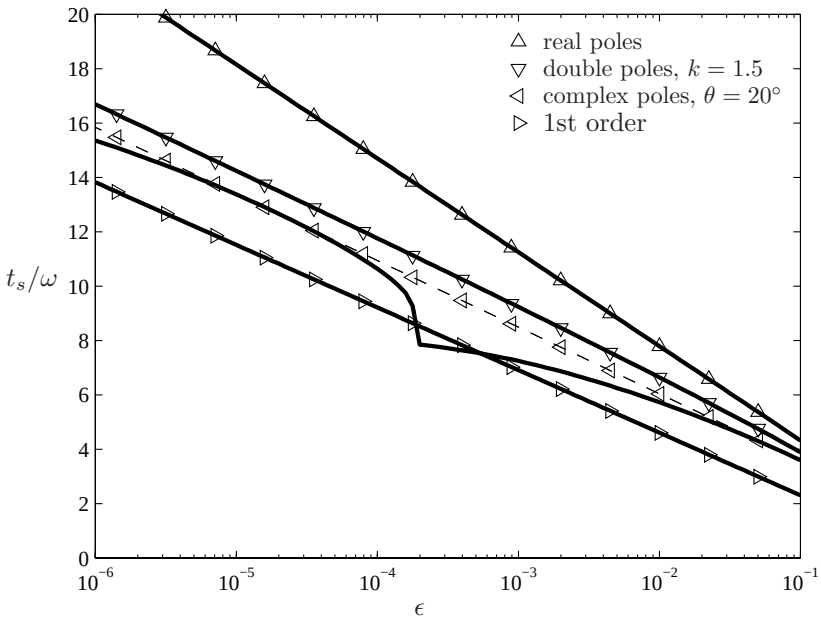

Figure 4.17: Plot of t_s versus ϵ for systems with same $\omega = \omega_1 = \omega_2$. The solid lines are the settling times for a first order system and three second order systems with the poles arranged as; double pole, real poles and complex poles. The corresponding dashed lines are the equations 4.34, 4.45, 4.37 and 4.39 respectively.

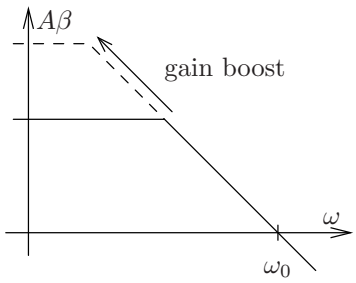

Figure 4.18: Bode plot of gain boosting

1. A first order system step response is generally faster than a second order system when the two systems have the same -3 dB bandwidth.

2. The optimal settling time for a second order system is comparable to or better than a first order system. The optimal settling time is predicted by equation 4.41. However a small increase in the pole sum will immediately destroy the performance. This is the case if some parasitics make the amplifier more unstable. To use this optimum in a design is hazardous, especially when ϵ is small.

3. The equations 4.37 and 4.39 are accurate except close to $(p_1+p_2)/\omega_2 = -2$ (a double pole). Then equation 4.45 should be used.

4. For a second order system the settling time is approximately equal to $\ln\epsilon/\max(\Re(p_1), \Re(p_2))$, where $\Re(p)$ is the real part of p

5. When designing a second order amplifier a double pole is recommended due to robustness. Then small parasitic capacitors actually make the amplifier more unstable thus making the settling faster.

4.3.4 DC gain and settling

A popular way to improve the DC performance of an amplifier is to use some kind of gain boosting. Most often this is employed by using cascode stages. If the load is capacitive the increase in impedance in the node corresponds to an equal increase in gain of the amplifier. This is excellent for low frequencies but what happens with the settling time? Well the pole in the same node where the impedance was boosted is also affected. The pole is shifted down

Figure 4.19: Simulation of settling time for first order settling into a 0.1% accuracy window where $\omega_0 = 5$ Grad/s

in frequency by the same amount as the impedance is boosted. Because the unity-gain frequency is constant and therefore the speed actually remains the same. Figure 4.18 illustrates how the pole is shifted lower in frequency while the DC loop gain increases.

However there is a small improvement in the settling time if the gain is increased, at least for low values of $A\beta$. This is illustrated in Figure 4.19 which shows a simulation, where the loop gain was swept from 1.5k to 1M. But shouldn't the settling be the same as ω_0 is constant? The answer is no. Because for low gain the final value is far from the ideal value why the effective settling window becomes smaller and the amplifier needs more time to settle. Fortunately this effect is small due to the exponential settling of the amplifier. According to the simulation in Figure 4.19 the benefit of increasing the loop gain ten times is about 10%! Efforts to improve the settling time should be spent on other design parameters than increasing the loop gain beyond $1/e$.

4.4 Voltage dividers

Division is easily made and is accurate using passive components. In the FP-ADC three kinds of passive voltage dividers are used. An R-2R-ladder for the binary weighting, a resistive divider for the gain stage and a capacitive divider for the pipeline stages. These dividers will be examined by looking at the accuracy, the noise and for the R-2R ladder — the speed.

The details of the derivation of the equations below can be found in appendix C.

4.4.1 Resistive and capacitive dividers

The simple voltage divider used for setting the gain of voltage amplifiers. The standard deviation for the gain error in a voltage amplifier using a

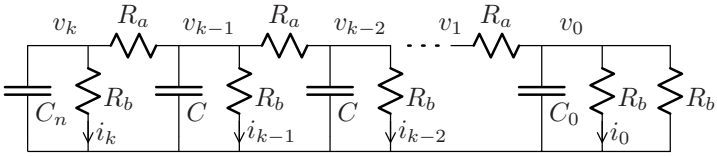

Figure 4.20: A R-2R divider

resistor voltage divider for feedback is

$$\sigma(e) = (1 - \frac{1}{G_\infty})\frac{A_R}{\sqrt{2}}\sqrt{\frac{1}{(WL)_1} + \frac{1}{(WL)_2}} \tag{4.46}$$

where e is the relative gain error of the amplifier, A_R is the process dependent matching parameter for resistors and WL the area of each resistor. The standard deviation of the relative gain error for a capacitive divider is similar to the voltage divider

$$\sigma(e) = (1 - \frac{1}{G_\infty})\frac{A_C}{\sqrt{2}}\sqrt{\frac{1}{(WL)_1} + \frac{1}{(WL)_2}} \tag{4.47}$$

where A_C is the process-dependent matching parameter for capacitors and WL the area of each capacitor.

The equivalent input noise contribution from the resistive feedback divider is

$$S_{FB} = 4kTR_1 \| R_2 \tag{4.48}$$

A capacitor does not add noise why the noise contribution from the capacitive divider is naught.

4.4.2 R-2R ladder divider

The R-2R ladder divider is shown in Figure 4.20. By choosing the ratio $R_b = 2R_a$ a voltage ratio of 2 is established between the taps. This means that the R-2R ladder is good from a matching point of view as it is possible to use unit resistors. The standard deviation of the relative gain error (relative

voltage division) between two taps, k and $k-1$, is

$$\sigma(e_k) = \frac{A_R}{\sqrt{10WL}}\sqrt{1+4/16^k}, \quad k \geq 1 \tag{4.49}$$

where WL is the area of the unit resistor. It is also assumed that R_a is realized by two parallel unit resistors. If R_b is realized by two series resistors the standard deviation of the relative gain error will increase roughly by $\sqrt{3/2}$.

The equivalent noise spectrum at each tap v_k is

$$S_{R2R} = 4kT\frac{R_b}{3} \tag{4.50}$$

Another interesting property of the ladder is the signal propagation delay through the ladder structure. Since a high accuracy is needed the resistor area tend to be quite large thus giving a significant parasitic capacitive load C. A first order approximation of the delay time between two taps is

$$\tau_k - \tau_{k-1} = \frac{2}{3}R_aC + 4^{-k}\left(2R_aC_0 - \frac{8}{3}R_aC\right) \tag{4.51}$$

Chapter 5

Distortions in Pipelined ADC

The standard way of describing the static non-linearity is integral and differential non-linearity (INL and DNL). The non-linearity is measured is the length of the quantization step compared to the ideal step. The INL and DNL measures are well suited for monotonous ADCs like a flash ADC. However these measures cannot accurately describe the non-linear behavior of a standard pipelined ADC.

In this thesis the definition of the static error of the ADC is the difference between d, the digital output signal referred to the input, and v_{in}, the input signal

$$v_e = d - v_{in} \tag{5.1}$$

This definition gives a direct measure of the ADC error in the amplitude domain.

By looking at the errors in a pipelined ADC from an architectural perspective it is possible to get the non-linearity as a linear combination of the individual errors in the pipeline stages. This way the origin of the non-linearity can be tracked and controlled [74].

The errors from the pipeline stages closely resemble the residues of the pipeline stages. Since the residue is a piecewise-linear function the non-linearity also becomes piecewise-linear. This distortion will show infinite derivatives so Volterra [75] and power series theory [76, 71] will fail.

The non-linearity can be used for calculating the harmonic distortion.

In [77] a power function is used to calculate low order harmonics while the method presented in [78] calculates the whole spectrum. These methods call for exact knowledge of the INL and cannot take into account stochastic variations. Other methods use a stochastic approach to predict the non-linearity [79, 80]. These methods do not directly indicate what contributes to the non-linearity.

However when knowing the shape of the non-linearity it is a simple matter to get the distortion from the ADC. The approach used is an amplitude-domain method [79] to calculate the energy directly from the distortion waveform, thus obtaining the THD without using frequency transforms.

5.1 The general error function

The nonlinear errors from the pipeline stages are considered to be of the first order (only gain and offset errors) and originate from the sub-DAC and the amplifier in the pipeline stages (see figure 3.4). The contribution to the non-linearity from higher order effects is considered to be small and all transient behaviors are considered to have died out. The errors of the sub-ADC are left out because with digital correction the effects of the errors in the sub-ADC can be reduced or eliminated [81].

Let us introduce errors in the pipeline stage model. A first order approximation is a gain error plus an offset. The gain errors of the signal and the reference are in many cases not equal. Therefore the expression for the transfer in the non-ideal pipeline stage can be expressed as

$$s_k = G_k(1 + e_k) \cdot s_{k-1} - (1 + f_k) \cdot r_k + g_k \tag{5.2}$$

where e_k and f_k are the gain errors of the radix and the reference respectively, whereas g_k is the offset error.

The flow-graph of the non-ideal input-output relation of a pipelined ADC, according to equation 5.2, is shown in Figure 5.1. The output at the last pipeline stage is

$$s_p = v_{in} \prod_{l=1}^{p} G_l(1 + e_l) - \sum_{k=1}^{p} \left[(r_k + r_k f_k - g_k) \prod_{l=k+1}^{p} G_l(1+e_l) \right] \tag{5.3}$$

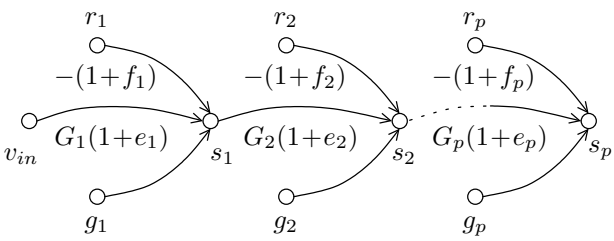

Figure 5.1: Flow-graph of a non-ideal pipelined ADC.

Solving v_{in} in equation 5.3 gives

$$v_{in} = \sum_{k=1}^{p} \frac{r_k + r_k f_k - g_k}{\prod_{l=1}^{k} G_l(1+e_l)} + \frac{s_p}{\prod_{l=1}^{p} G_l(1+e_l)} \tag{5.4}$$

When we recreate the input signal we are use equation 3.9 as if there were neither gain nor offset errors. According to the definition in equation 5.1 the input-referred error we get with this approximation is

$$d - v_{in} = \sum_{k=1}^{p} \frac{1}{\prod_{l=1}^{k} G_l} \left(r_k - \frac{r_k + r_k f_k - g_k}{\prod_{l=1}^{k}(1+e_l)} \right) - \epsilon \tag{5.5}$$

where ϵ is the quantization error. To simplify the equation of the input-referred error let us assume that the errors are so small that a multiplication of two errors becomes insignificant. It is also assumed that $\sum e_l \ll 1$ so we can use series expansion. Then the input-referred error is

$$d - v_{in} \approx \sum_{k=1}^{p} \frac{r_k \left(\sum_{l=1}^{k} e_l - f_k \right) + g_k}{\prod_{l=1}^{k} G_l} - \epsilon \tag{5.6}$$

This is consistent with results from other articles [82, 83]. This equation shows the error contribution from every reference. Still the contribution from the individual errors in the pipeline stages are concealed.

Now let us rearrange the input-referred error in such a way that the contribution from each error is separated. The quantization error is also removed so only non-ideal effects remain. The result is called the general

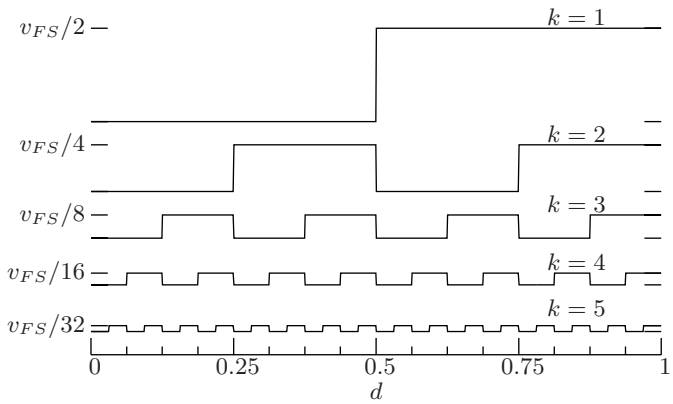

Figure 5.2: The first five digits for a pipelined ADC with 1-bit pipeline stages with respect to the digital signal d.

error function

$$v_e(d) = \sum_{k=1}^{p} e_k \left[d - \sum_{l=1}^{k-1} d_l \right] - \sum_{k=1}^{p} f_k d_k + \sum_{k=1}^{p} \frac{g_k}{\prod_{l=1}^{k} G_l} \tag{5.7}$$

where d_k is the digit from stage number k

$$d_k = \frac{r_k}{\prod_{l=1}^{k} G_l} \tag{5.8}$$

The sum of the digits form the digital output d (see equation 3.9). The relations between the digits and the digital signal are shown in the figures 5.2 and 5.3. in Figure 5.2 the digits for 1-bit pipeline stages are shown while in Figure 5.3 the digits for 1-bit RSD pipeline stages are shown.

The general error function holds under the assumption that no clipping occurs in the ADC. Clipping will happen if the errors of the first stages forces the signal outside the input range of the following stages. Clipping can be avoided by a redundant design.

5.1.1 Conclusions

The conclusion that we draw from equation 5.7 is that the error from a non-ideal pipelined ADC is a linear combination of the following:

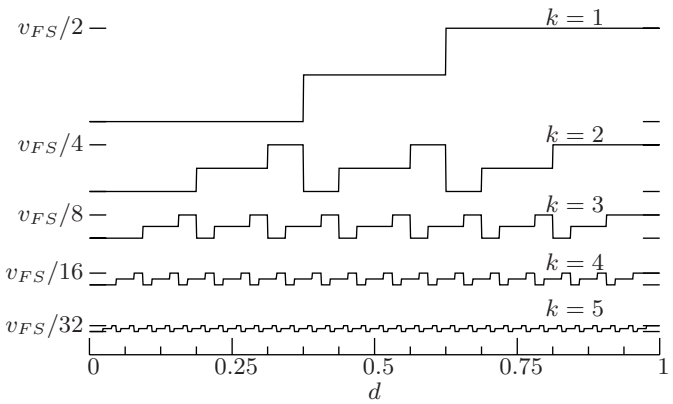

Figure 5.3: The first five digits for a pipelined ADC with 1-bit RSD pipeline stages with respect to the output signal d.

1. A gain error that is the digital signal times the sum of the radix gain errors, $d \sum_{k=1}^{p} e_k$;

2. A non-linear error for every stage due to the radix gain error. This error is the radix gain error times the sum of the digits from the previous stages, i.e. the digital signal prior to the pipeline stage. The sum of these errors is $-\sum_{k=1}^{p} \left(e_k \sum_{l=1}^{k-1} d_l \right)$;

3. A non-linear error for every stage due to the reference gain error. This error is the reference gain error times the digit from the same stage. The sum of these errors is $-\sum_{k=1}^{p} f_k d_k$;

4. An offset error that is the sum of the stage offsets referred to the input, $\sum_{k=1}^{p} \left(g_k / \prod_{l=0}^{k} G_l \right)$;

A common case is when $e_k = f_k$. Then the errors from point 2 and 3 can be combined into $-\sum_{k=1}^{p} \left(e_k \sum_{l=1}^{k} d_l \right)$. That is, each stage contributes with an error that is the radix /reference error times the digital signal at the stage.

The error from each stage can be seen as a fraction of a coarse quantization of the signal. Let us sum up the error contributions from point 1–3 for the stage k and assume that $e_k = f_k$. Then the error contribution from the stage resembles the quantization error that the ADC would give if it was

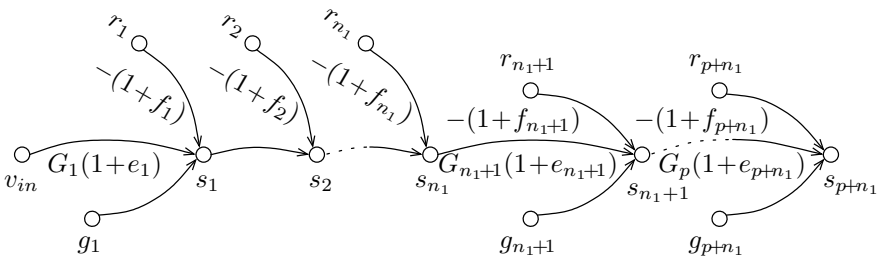

Figure 5.4: Flow-graph of a converter with an n_1-bit pipeline stage at the input.

truncated right after stage k [84]. The difference is a factor of e_k.

$$e_k \sum_{l=k+1}^{p} d_l \approx e_k s_k \tag{5.9}$$

5.1.2 Multi-bit stages

The general error function so far has only one common reference error per stage. This reference error works on a single reference that can take multiple values. Sometimes the reference in a multi-bit pipeline stage is the sum of multiple references each with its own reference error

$$s_k = G_k(1+e_k) \cdot s_{k-1} - \sum_{l=0}^{n_k} (1+f_{k,l}) \cdot r_{k,l} \tag{5.10}$$

n_k is the number of references in stage number k.

With a small rearrangement the general error function can be applied to a general multi-bit pipeline stage. By viewing each reference as its own pipeline stage with a radix gain of one, the general function does not need to be rewritten. Then the indexes are renumbered according to

$$k \leftarrow \mathrm{f}(k,l) = \sum_{\lambda=1}^{k-1} n_\lambda + l \tag{5.11}$$

Refer to the flow-graph in Figure 5.4. This figure shows a pipelined ADC with an n_1-bit pipeline stage at the input. Here the references in the first pipeline stage have been renumbered from $r_{1,1} \ldots r_{1,n_1}$ to $r_1 \ldots r_{n_1}$. The indexes of the following pipeline stages have been increased by n_1.

5.2 The error power

The power of the error now can be calculated using an amplitude-domain method [79] to calculate the energy of the distortion. The power of the error is calculated by the integration of the error power times the probability density function of the signal φ_d. Quite a large portion of the error energy comes from the offset and the linear term of the error. In many cases the gain and the offset errors are not considered when measuring the distortion. For harmonic distortion the energy of higher order distortion is compared to the energy of the signal [7]. Therefore the linear and offset terms should be removed, otherwise the calculated error power will be incorrect. The error power is calculated as the distance from the line $b_1 d + b_0$

$$\bar{v}_e^2 = \int_0^{v_{FS}} \left(v_e(x) - b_1 x - b_0\right)^2 \varphi_d(x) dx \tag{5.12}$$

For a periodic and odd input signal the coefficients b_1 and b_0 are the Fourier coefficients

$$b_m(s) = \frac{1}{T} \int_P \sin(k \frac{2\pi}{T} t) s(t) dt \tag{5.13}$$

where $b_m(s)$ is the Fourier coefficient number m of the signal $s(t)$.

The Fourier transform is linear. Applying equation 5.13 on the general error function, equation 5.7, yields the Fourier coefficients as follows

$$b_m(v_e) = \sum_{k=1}^{p} \left[e_k \left(b_m(d) - \sum_{l=1}^{k-1} b_m(d_l) \right) - f_k b_m(d_k) + \frac{g_k}{\prod_{l=1}^{k} G_l} b_m(1) \right] \tag{5.14}$$

The whole spectrum can be obtained this way, but we are only interested in the coefficients b_1 and b_0, the linear and the offset term respectively.

5.2.1 Sinusoids

The power of the error can be solved for a sinusoid signal

$$d(t) = a_1 \sin(2\pi \cdot t) + a_0 \tag{5.15}$$

where a_1 and a_0 are the amplitude and offset of the sinus respectively. Because the static error is independent of the frequency, the frequency has been normalized to 1 Hz. Now the Fourier coefficients can be calculated since the

s	$b_1(s)/V_{FS}$ BIT	$b_1(s)/V_{FS}$ RSD	$b_0(s)/V_{FS}$
d	1		1/2
d_1	0.6366	0.6164	1/4
d_2	0.2330	0.2430	1/8
d_3	0.0839	0.0900	1/16
d_4	0.0300	0.0325	1/32
d_5	0.0106	0.0116	1/64
d_6	0.0038	0.0041	1/128
$\sum_7^\infty d_k$	0.0021	0.0024	1/128

Table 5.1: Table of the Fourier coefficients b_1 and b_0 for different signals in the error function. The input signal is a full scale sinus input $a_0 = a_1 = V_{FS}/2$.

input signal is known and thus also the waveforms of the digits.

The coefficients $b_1(s)$ and $b_0(s)$ have been calculated for a full-scale sinus signal, $a_0 = a_1 = V_{FS}/2$. The result is shown in table 5.1. The coefficients used together with equation 5.14 give the linear coefficients for equation 5.12.

The coefficients are varying with the amplitude of the sinusoid signal. The variation in the linear term $b_1(s)$ for 1-bit pipeline and 1-bit RSD pipeline stages are shown on figure 5.5 and 5.6 respectively.

The probability density function of a sinus signal, with the amplitude a_1 and the offset a_0, is

$$\varphi_d(x) = \frac{1}{a_1\pi}\frac{1}{\sqrt{1-(x-a_0)^2/a_1^2}}, \quad \frac{|x-a_0|}{a_1} < 1 \tag{5.16}$$

RSD example

Let us assume that we have a pipelined ADC with 1-bit RSD stages, where the errors in the six first 1-bit RSD pipeline stages are $e_k = f_k = e$, $g_k = 0$. The input signal is a full-scale sinus and $V_{FS} = 1$.

Then equation 5.14 can be rewritten as

$$b_m(v_e) = e\left[6b_m(d) - \sum_{k=0}^{5}(6-k)b_m(d_k)\right] \tag{5.17}$$

The static and linear coefficients of the errors are calculated by inserting the

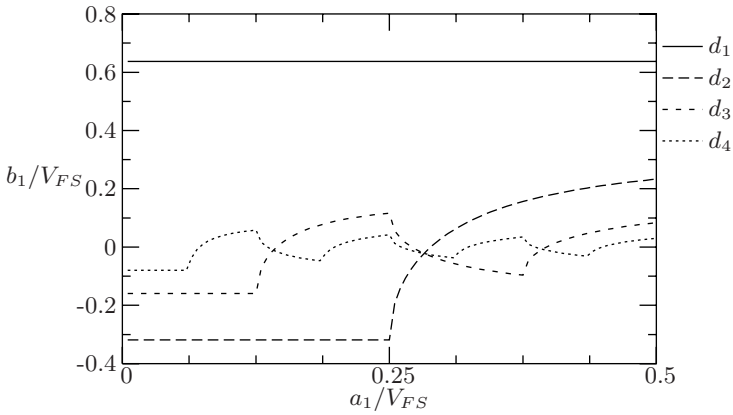

Figure 5.5: The linear coefficient b_1 versus the sinus amplitude a_1 for 1-bit pipeline stages.($a_0 = V_{FS}/2$).

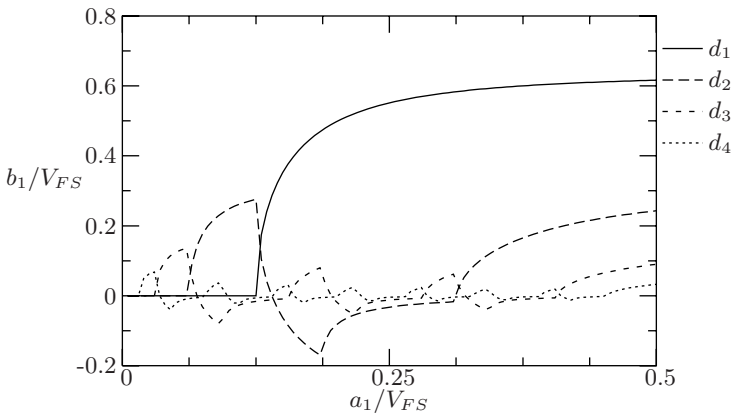

Figure 5.6: The linear coefficient b_1 versus the sinus amplitude a_1 for 1-bit RSD pipeline stages.($a_0 = V_{FS}/2$).

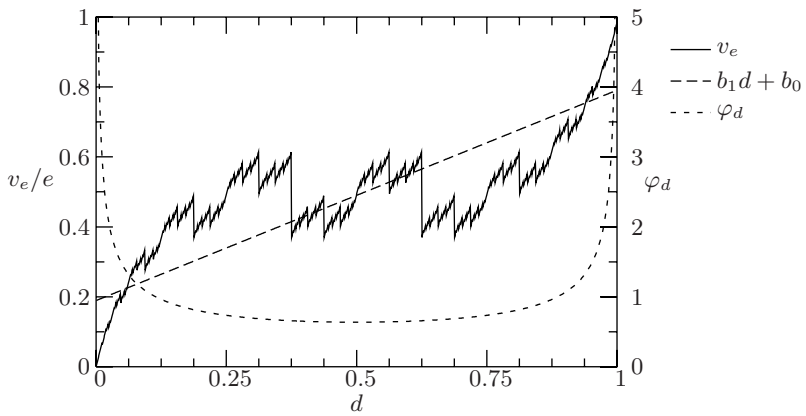

Figure 5.7: Plot of the functions in equation 5.12. $e_k = f_k = e,\ g_k = 0$

numbers from table 5.1

$$b_1 = 0.60\, V_{FS} \tag{5.18}$$

$$b_0 = 0.49\, V_{FS} \tag{5.19}$$

The coefficients and the PDF of a sinus signal given in equation 5.16 are inserted in equation 5.12. The waveforms of the integration are plotted in Figure 5.7. The voltage power of the error is

$$\bar{v}_e = 0.12|e|V_{FS} \tag{5.20}$$

which corresponds to a THD of

$$\text{THD} = \frac{\bar{v}_e}{V_{FS}/2\sqrt{2}} = 0.34|e| \tag{5.21}$$

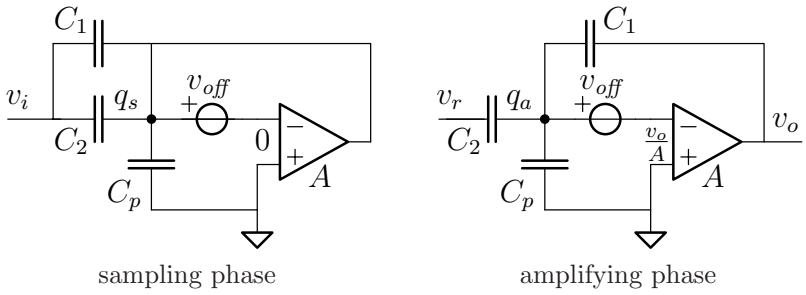

Figure 5.8: The pipeline stage's two phases.

5.3 Circuit example with simulations

Let us look more into details of the operation of a standard RSD 1-bit pipeline stage. The transfer function of such a stage is

$$v_o = \begin{cases} 2v_i + 1 & , v_i < -\frac{1}{4} \\ 2v_i & , -\frac{1}{4} \leq v_i \leq \frac{1}{4} \\ 2v_i - 1 & , v_i > \frac{1}{4} \end{cases} \tag{5.22}$$

Here the signal range has been normalized to ± 1 V.

A commonly used switched-capacitor (SC) stage that can implement this function [18] is shown in Figure 5.8. Non-ideal effects included in the amplifier model are a finite gain and an offset voltage. The dominant gain errors in an SC-circuit are capacitor mismatch, finite amplifier gain, parasitic input capacitance (see section 4.1). The offset error is determined by the offset of the amplifier and charge error of the sampling switch (see section 4.2.2).

In the sampling phase the two sampling capacitors C_1 and C_2 are charged to the input voltage. Then the capacitors are rearranged in the amplifying phase; C_1 is coupled in feedback over the amplifier and C_2 is connected to a reference voltage. Now the output voltage is the input voltage times two and minus the reference voltage. In this way equation 5.22 is implemented.

Due to the finite gain of the amplifier there is a slight change in the voltage of the common node. The charge on C_1, C_2 and the parasitic capacitance C_p changes and this manifest itself as an error on the output.

To get an exact value of the output a charge calculation is made. The total charge on the common node is the same for the sampling and amplifying

phase, thus

$$q_s = v_{off}(C_1 + C_2 + C_p) - v_i(C_1 + C_2) \tag{5.23}$$

$$q_a = v_{off}(C_1 + C_2 + C_p) - v_r C_2 - v_o C_1 - \frac{v_o}{A}(C_1 + C_2 + C_p) \tag{5.24}$$

Solving this equation gives the output voltage

$$v_o = \frac{v_i(C_1 + C_2) - v_r C_2}{\frac{C_1+C_2+C_p}{A} + C_1} \tag{5.25}$$

It is worth noting that the offset voltage v_{off} is completely cancelled.

Now we assume that C_1 and C_2 are two matched capacitors with a small mismatch of $\pm\frac{\Delta C}{2}$ from the nominal value because of random variations in the manufacturing

$$\begin{cases} C_1 = C + \frac{\Delta C}{2} \\ C_2 = C - \frac{\Delta C}{2} \end{cases} \tag{5.26}$$

Then output from the pipeline stage becomes

$$v_o = \frac{2v_i - v_r + \frac{\Delta C}{2C} v_r}{1 - \frac{\Delta C}{2C} + \frac{2}{A}\left(1 + \frac{C_p}{2C}\right)} \tag{5.27}$$

Identification between equation 5.2 and 5.27 give the error parameters

$$\begin{cases} e = -\frac{\Delta C}{2C} - \frac{2}{A}\left(1 + \frac{C_p}{2C}\right) \\ f = -\frac{\Delta C}{C} - \frac{2}{A}\left(1 + \frac{C_p}{2C}\right) \\ g = 0 \end{cases} \tag{5.28}$$

Let us call the term $\frac{\Delta C}{C}$ the capacitor mismatch and the term

$$\frac{\Delta G}{G} = \frac{2}{A}\left(1 + \frac{C_p}{2C}\right) \tag{5.29}$$

the gain mismatch.

If the ADC is made of identical stages then A, C, and C_p are constant and therefore also the gain mismatch. However the capacitor mismatch is a random process and is different for every pipeline stage.

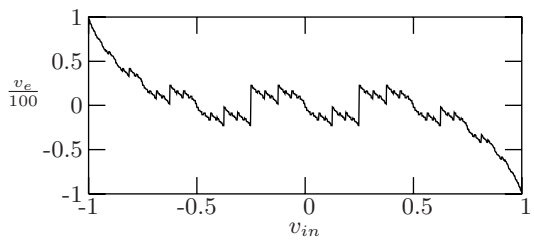

Figure 5.9: Simulation of the non-linearity v_e for a 10-bit pipelined RSD ADC. Here $\frac{\Delta G}{G} = 0.01$ and $\frac{\Delta C}{C} = 0$.

Now the input-referred nonlinear error of the ADC is, using equation 5.7

$$v_e = \sum_{k=1}^{p} \left[\frac{\Delta G}{G} \left(\sum_{l=1}^{k} d_l - d \right) + \frac{\Delta C_k}{2C} \left(d_k + \sum_{l=1}^{k} d_l - d \right) \right] \tag{5.30}$$

The quantization error has been omitted.

The error from an ADC is a non-linear error which is a linear combination of the individual errors from each stage, as expected. The static and the random errors in each stage are independent as well. Therefore it is possible to treat each effect individually.

5.3.1 Simulation of the non-linearity

The simulation of a 10-bit RSD pipelined ADC confirms the results from equation 5.30. First the gain mismatch in each stage was set to 0.01. The resulting total non-linearity due to the finite amplifier gain is shown in Figure 5.9. Shown in Figure 5.10a are the error contributions from the three first stages. From the figures one can see that the static error is a linear combination of the individual errors from each stage, as described in equation 5.30.

To show the shapes of the error function due to the capacitor mismatch a simulation of a 10-bit pipelined ADC with a capacitor mismatch of 0.01 was made. The error contributions from the first three stages are shown in Figure 5.10b.

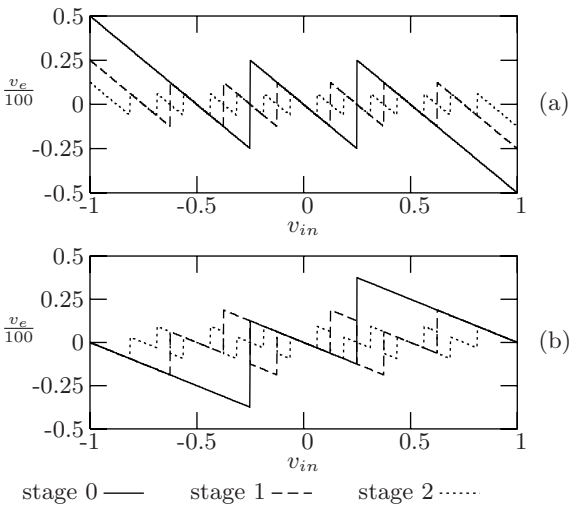

Figure 5.10: Simulation of the individual contribution to the non-linearity from the three first stages. v_e is shown for (a) $\frac{\Delta G}{G} = 0.01$ and (b) $\frac{\Delta C}{C} = 0.01$

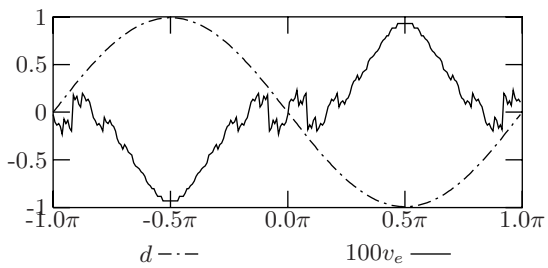

Figure 5.11: Simulation of the output and the error of a 10-bit pipelined ADC when $\frac{\Delta G}{G} = 0.01$. The input signal is a full scale sinus.

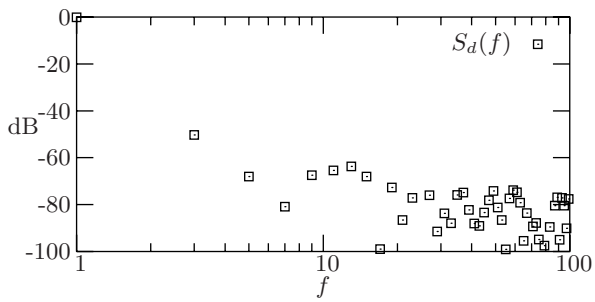

Figure 5.12: Plot of the spectrum of d when $\frac{\Delta G}{G} = 0.01$ for a full scale sinus.

5.3.2 Harmonic power of the static error

Now we want to obtain the energy of the harmonics in the output signal. The figure of merit discussed in this article is the THD. The test signal is a full scale sinus, where the frequency has been normalized to 1.

The digital output of a 10-bit RSD pipelined ADC where the input signal is a full-scale sinus is shown in Figure 5.11. Every pipeline stage has a gain mismatch of 0.01. Since the distortion on the digital output is not visible for the naked eye the normalized error is plotted in the same figure. The spectrum of the digital output is shown in Figure 5.12. The harmonic distortion for a full-scale sinus input is largely dominated by the third order harmonic. From the spectrum the THD can be calculated to $3.36 \cdot 10^{-3}$.

According to equation 5.30 the distortion is scaled linearly with $\frac{\Delta G}{G}$. Therefore the THD can be calculated directly from the gain mismatch

$$\text{THD} \simeq 0.34 \left| \frac{\Delta G}{G} \right| \tag{5.31}$$

5.3.3 Harmonic power of the total error

Now let us examine what happens if the gain error in each pipeline stage is not only static but random due to mismatch in the sampling capacitors. In this thesis the mismatch $\frac{\Delta C}{C}$ of the sampling capacitors is modeled as a random process with a normal distribution. If the capacitors are large (which is the case for high accuracy matching) oxide effects dominate the mismatch [85]. If the layout is carefully made only local oxide effects persist [86]. Then the standard deviation of the capacitor mismatch is inversely

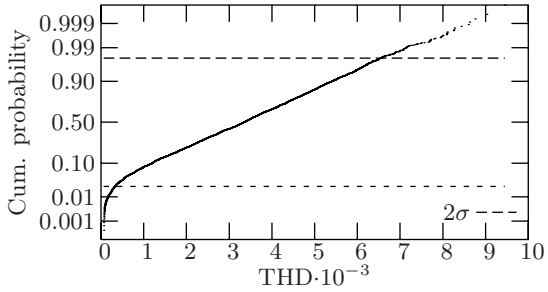

Figure 5.13: Cumulative probability plot for 4000 pipelined ADC samples, when $\frac{\Delta G}{G} = \sigma(\frac{\Delta C}{C}) = 0.01$

proportional to the square root of the size of the capacitors

$$\sigma(\frac{\Delta C}{C}) = \frac{A_C}{\sqrt{WL}} \tag{5.32}$$

In the equation, A_C is a process dependent constant, and WL the area of the unit capacitor.

Monte-Carlo simulations show that the THD from the ADC closely resembles a normal distribution with a standard deviation that is proportional to $\sigma(\frac{\Delta C}{C})$. The cumulative probability plot of the THD from such a simulation when $\frac{\Delta G}{G} = \sigma(\frac{\Delta C}{C})$ is shown in Figure 5.13. The plot shows that at least within 2σ the THD is normally distributed. The deviation from the normal distribution close to zero will be discussed shortly.

The simulations show that if $\frac{\Delta G}{G} \geq \sigma(\frac{\Delta C}{C})$ then the THD from the ADC can be approximated by a normal distribution. The mean of the THD can be approximated to

$$E(\text{THD}) \simeq 0.34 \cdot \frac{\Delta G}{G} \tag{5.33}$$

and the standard deviation to

$$\sigma(\text{THD}) \simeq 0.16 \cdot \sigma(\frac{\Delta C}{C}) \tag{5.34}$$

where 0.34 and 0.16 are fitting parameters. Now it is possible to use statistics to calculate the yield for the ADC. For example, the probability α that the

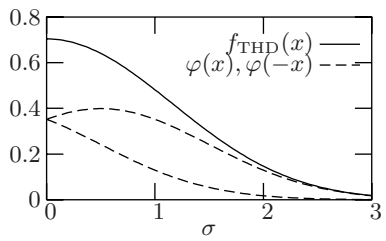

Figure 5.14: Plot of the PDF of the THD when $\frac{\Delta G}{G} = 0.5 \cdot \sigma(\frac{\Delta C}{C})$. $\varphi(x)$ is the PDF of the normal distribution.

α	λ_α	λ'_α
0.5	0	0.68
0.1	1.28	1.65
0.05	1.64	1.96
0.01	2.33	2.58
0.005	2.58	2.81
0.001	3.09	3.29

Table 5.2: λ_α and λ'_α versus α.

ADC fails a specification of THD_α is

$$\alpha = P(\text{THD} > \text{THD}_\alpha) = 1 - \phi(\text{THD}_\alpha) \tag{5.35}$$

where $\phi(x)$ is the CDF[1] for the normal distribution. The inverse is

$$\text{THD}_\alpha = 0.34 \left| \frac{\Delta G}{G} \right| + \lambda_\alpha \cdot 0.16 \cdot \sigma \left(\frac{\Delta C}{C} \right) \tag{5.36}$$

where λ_α is the α-quantile for the normal distribution, which can be found in table 5.2.

However if the capacitor mismatch starts to dominate over the gain mismatch, the normal distribution becomes distorted. Since the THD never can be negative, the tail of the PDF for the normal distribution is mirrored at the origin. This effect is illustrated in Figure 5.14. The result is that α is getting higher than expected. In situations where $\frac{\Delta G}{G} \ll \sigma(\frac{\Delta C}{C})$ the modified quantile λ'_α, presented in table 5.2, should be used instead of λ_α.

[1] Cumulative distribution function

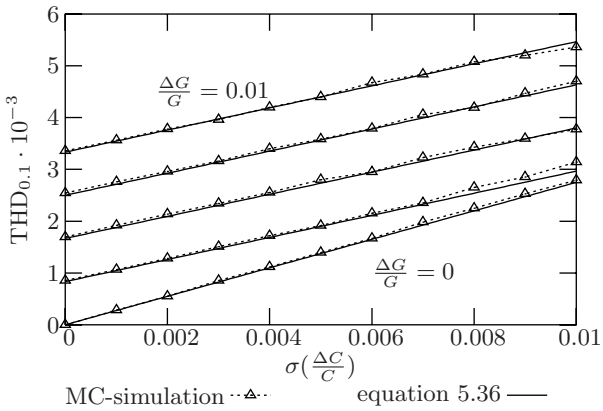

Figure 5.15: Plot of $\mathrm{THD}_{0.1}$ versus $\sigma(\frac{\Delta C}{C})$ for $\frac{\Delta G}{G}$ in steps $2.5 \cdot 10^{-3}$.

This is the case for the Monte-Carlo simulations of the $\mathrm{THD}_{0.1}$, which are shown in Figure 5.15. For the case when $\frac{\Delta G}{G} = 0$ then $\lambda'_{0.1}$ is used instead of $\lambda_{0.1}$. In this figure every point is the result of a 4000 pipelined ADC samples Monte-Carlo simulation.

5.4 Differential and integral non-linearity

The most commonly used measure of the non-linearity in an ADC is the differential non-linearity (DNL) and the integral non-linearity (INL). The unit of the DNL and INL is one LSB.

The DNL is defined [8] as the input referred step size of a specific digital code, weighted by the ideal step size q (LSB weighting)

$$\mathrm{DNL}(d) = \frac{v_{th}(d + 1\ \mathrm{LSB}) - v_{th}(d)}{q} - 1 \tag{5.37}$$

For an ideal linear ADC the step size is always q, which corresponds to a DNL of zero. The step size cannot be smaller than zero why the lower limit of the DNL is -1 LSB.

The INL is defined as the distance from centers of the ideal step and the actual step, weighted by the step size q. The INL can be calculated from

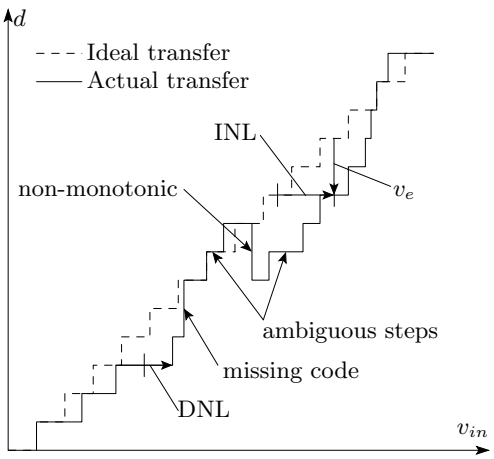

Figure 5.16: Illustration of the differential and integrated non-linearity.

the sum of the DNL according to

$$\mathrm{INL}(d) = \sum_{k=0}^{d-1\mathrm{LSB}} \mathrm{DNL}(k) + \frac{1}{2}\mathrm{DNL}(d) \tag{5.38}$$

The INL can also be approximated by the general error function (equation 5.7)

$$\mathrm{INL} = -\frac{v_e}{q} \tag{5.39}$$

The DNL and INL measures are well suited for monotonous ADCs, like a flash ADC. However the DNL and INL cannot describe all the non-linear behaviors of a standard pipelined ADC. If the transfer function is non-monotonic then two steps belong to a specific code. Then the DNL and INL become ambiguous. Refer to the illustration in Figure 5.16. In the figure the transfers for an ideal and an actual ADC are shown. The DNL, INL and the input referred error v_e are indicated by arrows in the figure. Also shown are a missing code and a non-monotonicity. It is clear from the figure that the INL and DNL can describe missing codes but not non-monotonicities.

Chapter 6

Design Considerations for the Proposed FP-ADC

The target is to create an $n + m$-bit FP-ADC chip according to the architecture presented in section 3.5. Then we need a distributed amplifier with binary weighted gains of 2^k, where $k = 0 \ldots m$. The gain has to be chosen instantly every sample by a controller. The exponent corresponds to the chosen gain and the mantissa is converted using an n-bit linear ADC.

The parts of the architecture that are the most important for high performance are the ones that generate the binary weighted gain. In particular the divider, the gain stage, and the controller need consideration.

The implementation of the row (including a part of an R-2R divider, a gain stage, a SH stage, and an RSD 1-bit pipeline stage) is shown in Figure 6.1. Many different design considerations have led to this row configuration. The design considerations follow below.

6.1 Overall design tradeoffs

Since the gain stages need to be continuous in time we can not use capacitors neither in the divider nor in the feedback of the gain stage. Capacitive networks need some way of setting the DC voltage in the floating nodes. This means SC-circuits with AZ. One of the main ideas of this FP-ADC is to push the sample-and-hold (SC-operation) later in the architecture. That is why capacitors in the gain stages are not allowed.

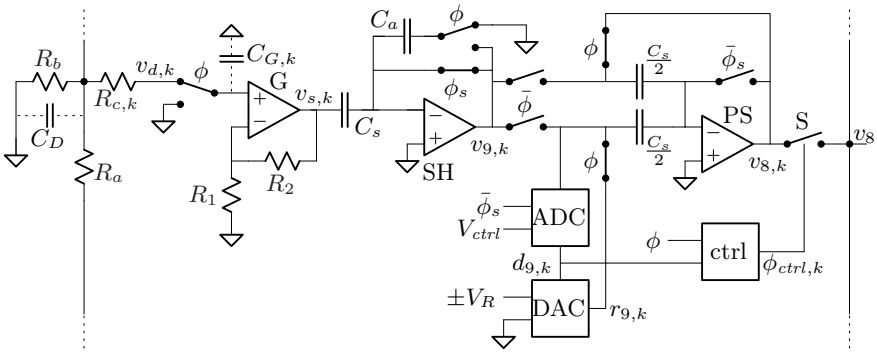

Figure 6.1: Overview of row number k in the FP-ADC. The interfaces between the divider, the gain stage, the sample-and-hold stage and the pipeline stage are the nodes of $v_{d,k}$, $v_{s,k}$, and $v_{9,k}$

The FP-ADC solution used in this thesis suffers from the same problem as time-interleaved converters [87, 88]. The phase delay, gain and offset of each row have to be identical. The absolute gain in the rows is not a problem as long as they are identical. In order to minimize the mismatch between the rows a voltage divider is used in front of identical gain stages.

For a differential design only shunt-feedback amplifier topologies can be used. Then the divider needs to be tapped of current. These currents can be very high due to the low resistance of the ladder. Also to remove the offsets a chopper switch is needed. Otherwise matching between the rows would become a problem. The resistance of the chopper switch needs to be very small in order not to introduce harmonics due to the non-linear resistance of the transmission gate. Then the transmission gates used in the chopper need to be very large. These are the reasons why a differential design was rejected even though it has many benefits. A single-ended design was chosen. This means that special care on PSRR and charge injection have to be taken.

The n-bit ADC is implemented using RSD 1-bit pipeline stages terminated by a small flash ADC.

The type of switches used for sampling (all switches using ϕ_s or $\bar{\phi}_s$) is the dummy switch. The dummy switch is used since charge errors should be minimized at the sampling instant. For all other switches in the FP-ADC design, the transmission gate is used since it is simple and the voltage supply is high enough. The size ratio of the NMOS and PMOS transistors in the

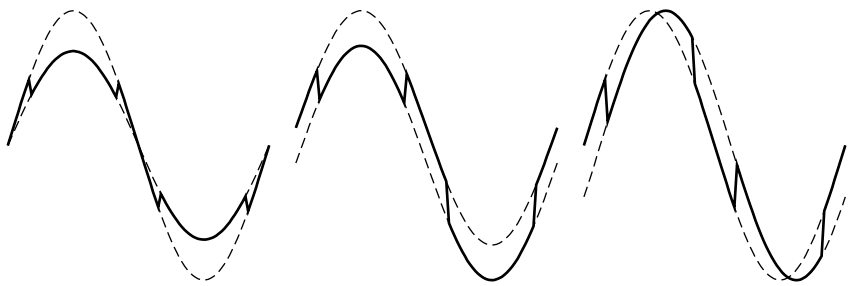

Figure 6.2: Illustration of the effect on the digital waveform from the mismatch between two rows. From left to right: gain, offset and delay mismatch. The errors have been greatly exaggerated for visibility.

transmission is 1:3, to get the most linear switch conductance. But for the chopper switches the currents and voltages are small so a linear conductance is of minor importance. Instead a ratio of 1:1 was used to minimize the charge errors.

6.2 Matching

When a parallel structure is used it is very important that all the rows match each other. Between two successive rows the amplitudes should differ exactly by a factor of two. The gain, the offset and the delay should be identical for all the rows to guarantee that no spurious signals appear when the signal is reconstructed. If the rows are not identical mismatch errors, illustrated in Figure 6.2, will come into effect. This is similar to time-interleaved converter arrays [87]. But unlike in time-interleaved converters the distortion of the matching errors in an FP-ADC appears as harmonic distortion.

The efforts to match the FP-ADC rows with regards to gain, delay and offset are discussed in the following sections.

6.2.1 Delay matching

The delay of the signal through the gain stage equals the group delay, $d\varphi(\omega)/d\omega$, of the gain stage. Let us assume that the amplifier in the gain stage has one dominant pole situated at p_1 and a DC loop gain of $A\beta_0$. Then

the closed loop pole is situated at

$$p_1' = p_1(1 - A\beta_0) = \omega_0 \tag{6.1}$$

where ω_0 is the -3 dB bandwidth of the gain stage. Now we can calculate the delay of the gain stage k for low frequencies

$$t_{d,k} = \frac{d\varphi(\omega)}{d\omega} = \left.\frac{1/p_1'}{1 + (\omega/p_1')^2}\right|_{\omega \ll p_1'} = \frac{1}{p_1(1 - A\beta_{0,k})} = \frac{1}{\omega_{0,k}} \tag{6.2}$$

where $k = 0 \ldots m$ is the row number. If we want to create a row of gain stages with binary weighted gains a first try would be to make them with amplifiers having the same gain A, but different feedback factors

$$\beta_{0,k} = \beta_{0,0} \cdot 2^k \tag{6.3}$$

where $\beta_{0,0}$ is the feedback factor of the gain stage number 0. The maximal delay error between the gain stages with the highest and the lowest gain is

$$\Delta t_d = t_{d,0} - t_{d,m} \approx t_{d,0} = \frac{1}{\omega_{0,0}} \tag{6.4}$$

If we want the delay time to generate an error smaller than $\frac{1}{2}$ LSB for a sinusoid,

$$\frac{V_{FS}}{2}\omega_{in}\Delta t_d < \frac{V_{FS}}{2^{n+1}} \tag{6.5}$$

then the requirement on the bandwidth of the slowest gain stage is

$$\omega_{0,0} > \omega_{in} \cdot 2^n \tag{6.6}$$

Example What is the required closed loop bandwidth for the gain stage if you want less than $\frac{1}{2}$ LSB error at Nyquist frequency for a 10-bit accuracy at a sampling frequency of 100 MHz?
Answer: $f_0 \geq \frac{f_s}{2} \cdot 2^n = 51\,\text{GHz}$

A better solution would be to use identical amplifiers with identical feedback factors in the gain stage. Then the systematic delay error will disappear and only random effects will introduce a delay mismatch. To create the binary weighted gains we can use a passive voltage divider and identical gain stages. The divider must be placed in front of the gain stages. Otherwise the voltage supply will limit the output of the amplifiers before the division.

The divider itself generates a systematic delay. The limited speed of the signal through the divider generates a phase delay between the outputs of the divider. The delay of the divider (R-2R ladder) is expressed in equation 4.51 and can be approximated to

$$t_{d,k} = \frac{2}{3} k R_b C_D \tag{6.7}$$

where $C_D$is the parasitic capacitance at the R-2R ladder terminals. The delay in the divider can be equalized using a resistor $R_{c,k}$ in series with the outputs. This resistor will interact with the input capacitance of the gain stage C_G and produce an additional delay. The value of the resistor becomes

$$R_{c,k} = \frac{t_{d,k}}{C_G} = \frac{2}{3}(m-k) R_b \frac{C_D}{C_G} \tag{6.8}$$

6.2.2 Gain matching

In a pipelined ADC it is important that the absolute gain is accurate so the linearity is not destroyed [74]. However in the proposed FP-ADC architecture the absolute gain in the rows are of no importance, just as long as the gain is binary weighted between the rows. By using a high loop-gain negative-feedback the gain matching is determined predominantly by the passive components in the feedback, in our case R_1 and R_2. By careful layout of the circuits systematic errors of the passive components can be removed so only small random variations from the manufacturing remain.

In order to reduce random variations in passive components the device area should be increased. The standard deviation is inversely proportional to the square root of the area of the passive component

$$\sigma \sim \frac{1}{\sqrt{WL}} \tag{6.9}$$

In the rows there are three major sources to the gain mismatch. The ladder mismatch σ_D, the gain stage mismatch σ_G, and the sample-and-hold stage gain mismatch σ_{SH}. The total gain mismatch is the geometrical sum of the stages involved

$$\sigma^2(\frac{\Delta G}{G}) = \sigma_D^2 + \sigma_G^2 + \sigma_{SH}^2 \tag{6.10}$$

The sample-and-hold stage is an SC-circuit. Because the matching of capacitors is approximately 10 times better than the matching of resistors the term

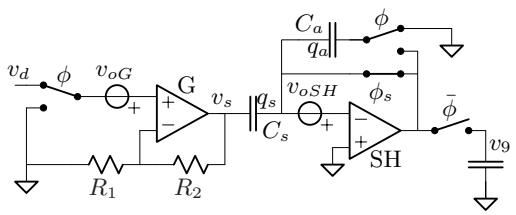

Figure 6.3: The offsets from the gain and sample-and-hold stages are cancelled by chopping.

σ^2_{SH} can be considered insignificant. The area of the resistors in the gain stage needs to be small to maximize the speed of the gain stage. Therefore these resistors were allowed to dominate the gain mismatch.

6.2.3 Offset cancellation

Offset in the gain stage is a killer. The input offset error v_{off} of a CMOS amplifier can easily reach 10 mV. The offset error at the output of each row is $v_{off} \cdot 2^m$, generating an offset between two different rows in the range of >100 mV! This is clearly unacceptable. The offset of the gain stages have to be cancelled. The problem is solved by chopping the input signal in synchronization with the AZ of the SH-stage. The proposed solution where the chopping switch is placed after the divider is presented in Figure 6.3. With this arrangement the size of the chopping switches can be made smaller compared to placing a single switch at the input.

A charge preservation calculation shows that the offsets of both the gain and sample-and-hold stages are removed

$$q_s + q_a|_{\phi} = -v_{o,SH}C_s - v_d G_G C_s - v_{o,G} G_g C_s - v_{o,SH} C_a \quad (6.11)$$

$$q_s + q_a|_{\bar{\phi}} = -v_{o,SH}C_s - v_{o,G} G_G C_s - v_{o,SH} C_a - v_9 C_a \quad (6.12)$$

where $G_G = 1 + R_2/R_1$ is the gain of the gain stage. Solving for v_9 gives

$$v_9 = v_d G_G \frac{C_s}{C_a} \quad (6.13)$$

This way both offsets of the gain stage and the sample-and-hold stage amplifier are cancelled, and still the operation is completely analog, the input

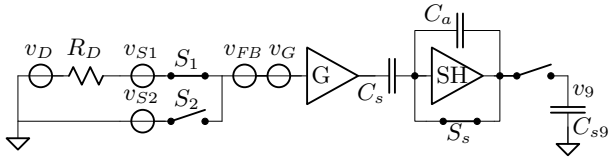

Figure 6.4: Block level figure of the noise from the divider (V_D), the chopper switches (v_{S1} and v_{S2}) and the gain stage (v_{FB} and v_G) .

signal is chopped (not time discrete). However even though the offset is cancelled by AZ, it is important that it is limited, otherwise the positive or the negative swing of the gain stage output can effectively disappear.

Another source of offset is the charge error from the sampling switch. This offset cannot be removed by AZ so a dummy switch has to be introduced to cancel the charge errors. However this canceling is never perfect and a residue will remain. When using a dummy switch the parasitic capacitance between the sampling node and the clock supply increases. This will affect the PSRR negatively.

6.3 Noise

The noise from the divider and the gain stage will dominate the total noise performance of the FP-ADC. The model used for calculating the noise is shown in Figure 6.4. Here are v_D the voltage noise from the divider, v_{FB} the voltage noise from the gain stage feedback resistors, and v_G the voltage noise of the gain stage amplifier. It is evident that the noise contribution is dominant from the circuits prior to and within the gain stage. After the gain stage the noise power contributions are a factor $1/G_G^2$ smaller when referred to the input of the FP-ADC.

The noise energies, from both the amplifying and the sample phase, appear at v_9 due to the AZ. The noise at the input of the n-bit ADC is

$$v_9 = 2^m \left(v_D + v_{S1} + v_{S2} + \sqrt{2} v_{FB} + \sqrt{2} v_G \right) \tag{6.14}$$

where $G \cdot C_a / C_s = 2^m$. The noise energies of the divider and the two chopper switches just appear once during a clock cycle. The noise of v_{FB} and v_G appear both in the amplifying and sampling phase (CDS) which doubles the

thermal noise energy, hence the $\sqrt{2}$ factor.

By using Wiener-Kintchine's theorem [70] and $S_{S1} = S_{S2}$ the noise spectrum becomes

$$S_9 = 2^{2m}(S_D + 2S_{S1} + 2S_{FB} + 2S_G) \tag{6.15}$$

By assuming that every spectrum $S_x = 4kTr_x$ in the above equation the equivalent noise resistors are obtained

$$r_9 = 2^{2m} \cdot (r_D + 2r_{S1} + 2r_{FB} + 2r_G) = 2^{2m} r_{eq} \tag{6.16}$$

where r_{eq} is the equivalent noise resistor of the noise sources at the input of the gain stage.

If we want the thermal noise to be lower than the quantization noise

$$4kTr_9 \cdot B < \frac{V_{FS}^2}{12 \cdot 2^{2n}} \tag{6.17}$$

then the maximal r_{eq} can be solved. By inserting the optimal box-bandwidth from equation 2.18 into the expression and solving for r_{eq} yields

$$r_{eq} < \frac{V_{FS}^2}{4kT\frac{(n+1)\ln 2}{2} f_s \cdot 12 \cdot 2^{2n+2m}} \tag{6.18}$$

The noise requirement on the $n+m$ bit FP-ADC is more relaxed compared to a linear $(n+m)$-bit ADC, especially for high values of m. This is because of the required settling time.

Example For a 10+5 bit FP-ADC sampled at 100 MHz with an input range of 2.8 V the equivalent noise resistance is 92 Ω (a 15-bit linear ADC would yield 67 Ω).

The equivalent resistances of the R-2R divider, the feedback, the chopper switches, and the gain stage amplifier are as follows. r_D is given from equation 4.50 plus the equalization resistor

$$r_D \approx R_b/3 + R_c \tag{6.19}$$

From the equations 7.16 and 7.17, r_D and r_{FB} are found to be

$$r_G = 2\gamma_{noise}\frac{1}{g_m} = \frac{2}{3}\frac{V_{GS} - V_T}{I_D} \tag{6.20}$$

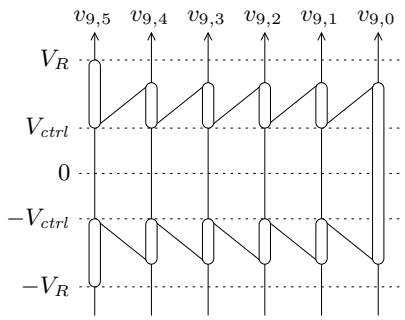

Figure 6.5: Illustration on how the controller selects the appropriate row for conversion ($v_{9,k} = v_{in}2^{5-k}$).

$$r_{FB} = R_1 \parallel R_2 \approx R_1 \tag{6.21}$$

The noise of the switches are approximated to the inverse switch conductance

$$r_{S1} \approx \frac{1}{g_{d,S1}} \tag{6.22}$$

6.4 The control algorithm

The purpose of the controller is to select the row where the signal has been amplified just within the range of the ADC. The inputs to the controller are the data from the comparators in each row, and the outputs are the clock phases to the CMOS switches. The timing is controlled by the system clock phases.

The selection is done in such a way that the row with the highest index k, which satisfies the equation

$$V_{ctrl} = \frac{V_R}{2} - V_{safe} \leq |v_{9,k}| \tag{6.23}$$

, will be selected for conversion. V_{ctrl} is half the reference voltage, V_R, minus a safety margin, V_{safe}, to ensure that no clipping occurs due to offsets in the comparators. This way the controller can reject gain stages that are saturated. The selection algorithm of the controller is illustrated in Figure 6.5.

6.4.1 Comparators of the controller

It is not recommended to use comparators before the signal has been sampled. The signals are continuous time and are likely to have a fast slope. The comparators need to be very fast and accurate timing-wise. If they are not fast enough the delay time in the comparator will introduce a large offset error for high frequency signals.

Example What is the maximal offset error when sampling a 50 MHz sinusoid, amplitude of 1V, using a comparator with t_d=0.5 ns?
Answer: The offset error will be $\approx 2\pi \cdot a_1 \cdot f_{in} \cdot t_d$= 157 mV

It is also troublesome that the number of comparators is multiplied by the number of rows, so the use of high power and fast comparators is not feasible.

The solution is to have one pipeline stage before the CMOS-switch. In this way half a clock period is gained and the signals are discrete time. Then the rate of change is almost zero. This has the benefit that comparators with a long delay can be used. Still the comparators need to be very fast but now in terms of latency, i.e. the comparators need to decide quickly.

The increase in power consumption will not be dramatic as the pipeline stages are low-power compared to the gain and sample-and-hold stages. Also the comparators inside the first pipeline stage can be merged with the controller's comparators. Since we accept a $\pm V_{FS}/8$ error to the threshold value in an RSD 1-bit pipeline stages, this operation is permitted. The same thresholds are used for the first pipeline stage and for the controller. In this way the number of comparators is reduced by half.

The timing of the input and control signals in a row is shown in Figure 6.6. First the chopping switch switches the gain stage from the AZ phase to the amplifying phase when ϕ goes high. The gain stage starts to track and amplify the signal $v_{d,k}$ from the resistive divider. When ϕ_s goes low, the charge on the sampling capacitor C_s is preserved. The non-overlap time t_{no} makes sure that no other switching occurs at the sampling instant. Then ϕ goes low and the chopper switch sets the gain stage input to zero in the AZ-phase. By chopping the input this way of the gain stage the offset of the gain stage is cancelled as described in section 6.2.3.

When the gain stage is in AZ the sample-and-hold stage is in its amplifying phase. The charge of the sampling capacitor C_s is transferred to the amplifying capacitor C_a. The output from the sample-and-hold stage, $v_{9,k}$, is sampled by the first pipeline stage of the n-bit ADC. The decision of

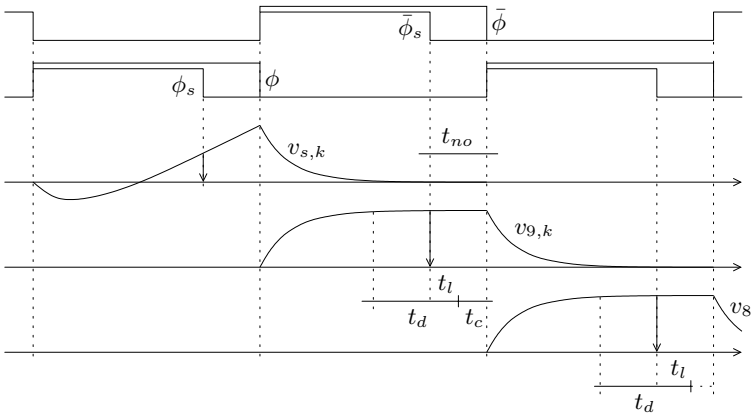

Figure 6.6: The settling of the stages in a row and the timing for the control logic.

the controller must be ready when ϕ goes high once again and the pipeline stage starts to amplify. This means that the latency time of the comparator, t_l, plus the decision time of the controller, t_c, need to be smaller than the non-overlap time

$$t_l + t_c < t_{no} \tag{6.24}$$

The actual decision time point referred to the output of the gain stage, $v_{9,k}$, is actually earlier, due to the delay t_d in the comparator. However this does not introduce errors as the slope of $v_{9,k}$ is low and the control logic accepts an offset error. The decision of the controller drives the CMOS switch and the output from the selected row is fed to the next pipeline stage.

6.4.2 Gain stage saturation

The fact that the gain stages are always in operation, regardless of the input amplitude, means that at some point one or more of the gain stages will be saturated to the supply voltage or the ground. This is not a problem as long as the saturated signal is definitely outside the input range of the ADC. These rows will be rejected by the controller and their outputs will not be used. The gain stages are reset to zero during the AZ phase. The AZ makes sure that no gain stage is saturated at the beginning of each amplifying phase.

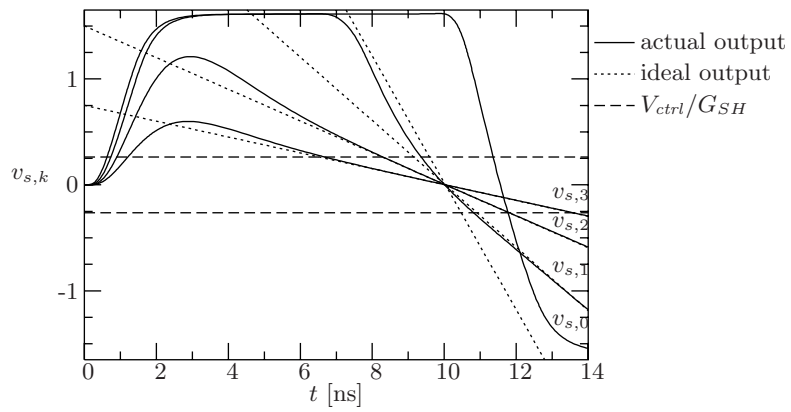

Figure 6.7: A simulation of saturation in the gain stages. The slope corresponds to the zero-crossing of a 2 MHz full-scale signal. The corresponding V_{ctrl} at the input of the sample-and-hold stage is plotted as well.

However this scheme will not alleviate the problem of the dynamic settling behavior when the gain stage goes out of saturation. The problem arises when the input amplitude goes from a high amplitude and returns to zero, so the gain stages come out of saturation. The recovery from saturation is not instantaneous. It takes some time for the output of the gain stage to settle to its correct value. During the recovery time (which can be long) the signal from the gain stage may be sampled. The control algorithm should reject that sample.

A simulation of the transient response of the gain stages, where the input signal is a slope corresponding to a zero-crossing of a 2 MHz signal, is presented in Figure 6.7. Initially the gain stages are in the AZ phase. At $t = 0$ they are switched to the amplifying phase and start to amplify the signal, $v_k = v_i G_G 2^{-k}$. Initially the output of gain stage number 0 goes into saturation but recovers from the saturation when the signal returns to zero. If the signal is sampled between 9–11 ns then the controller will select row 0 which will give rise to a large error as the output of gain stage 0 is still saturated.

A solution to this problem is to detect whether a gain stage has been saturated during the amplifying phase with a mono-stable latch. The latch should be active during the whole amplifying phase and be probed at the sampling instant. If $v_{s,k} > V_{ctrl}$ during the amplifying phase the latch will

detect that the gain stage has, at one point before the sampling instant, been saturated. Just to be sure this row will be rejected.

The solution reported in [50], is to insert a memory (low-pass filter) in the control logic to ensure that if a sample from a row is saturated, the row will not be used for some time. This ensures that the row has enough time to recover before it is used again.

Chapter 7

The Design of the Floating-Point ADC

In the design of an FP-ADC as an integrated circuit, we set out to make a 100 MS/s 10+5 bit FP-ADC with an input bandwidth of 50 MHz. In this chapter the design of this FP-ADC chip is described; The functionality and the simulated performance of different functional blocks are given.

The block level architecture is given in Figure 7.1. The signal v_{in} enters the voltage divider from the left in the figure. In the divider the signal is attenuated successively by a factor of two for every divider output

$$v_{d,k} = v_{in} 2^{-k} \tag{7.1}$$

where k ranges from 0 to 5. Each divider output is amplified by a factor of 17 in the gain stage column and then sampled in the sample-and-hold stage column. The sample-and-hold stages have a gain of 2 which means that the output of the k-th sampling stage equals

$$v_{9,k} = \frac{17}{16} v_{in} 2^{5-k} \tag{7.2}$$

These signals are sampled by the pipeline stage column. The column corresponds to the first pipeline stage in the 10-bit ADC. At the same time the controller decides which of the rows shall be used. The controller also calculates the exponent and forwards the digital signals from the selected row to the synchronizer. The CMOS switch is embedded in the pipeline stage

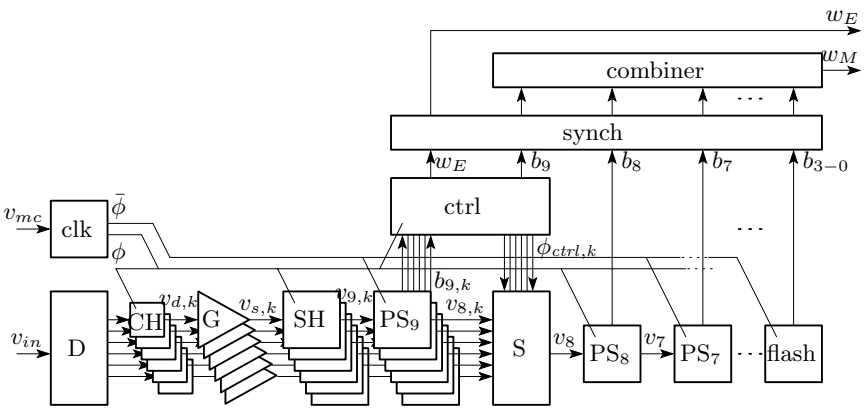

Figure 7.1: The architecture of the 10+5 FP-ADC chip. The pipeline stages 6–4 are not shown in the figure.

column.

After the pipeline stage column, there is only one row and the signal is converted through the pipeline stages 8–4. The four LSBs are converted in the terminating 4-bit flash converter stage. All the digital outputs from the pipeline stages are synchronized and then fed to the combiner. The combiner executes the digital error correction and produces the mantissa value.

The comparators of the pipeline stage column and the controller have been merged. So the comparators from the first pipeline stages are actually used for the control logic. Since we accept a $\pm V_{FS}/8$ error to the threshold value in an RSD 1-bit pipeline stage, this operation is permitted. The same thresholds are used for the pipeline stage column and for the controller. In this way the number of comparators is reduced by half.

All the references to the pipeline stages and the 4-bit flash are generated by means of a resistor ladder.

The clock phases are generated in the phase generator and distributed throughout the chip. The phases used for the pipeline stages are toggled when going down the pipeline. It ensures that when one pipeline stage is amplifying, the next is sampling.

7.1 Noise calculation

The noise is important in the design of the FP-ADC. Equation 6.18 gives a requirement on the equivalent noise resistor r_{eq}. r_{eq} is given in equation 6.16.

A reasonably small value of R_b in the divider would be 100 Ω, not to get a too large current in the divider. For an sinusoid with an amplitude of input voltage of 1.4 V, the peak input current would be 14 mA, corresponding to a power of 10 mW. This gives

$$r_D = 33\,\Omega \tag{7.3}$$

The gain stage feedback resistors cannot be too small, otherwise the current through the feedback will be too large

$$I_{FB} = \frac{V_{FS}}{2}/2^m/R_1 \tag{7.4}$$

If $I_{FB} \leq 100\,\mu$A, then $R_1 \geq 440\,\Omega$. The decision was taken to let this noise dominate the noise performance. The value chosen is

$$R_1 = 440\,\Omega \tag{7.5}$$

The drain current of the ASCS input in the gain stage was decided to be $I_D = 1$ mA giving a gain stage equivalent noise resistance of

$$r_G = 133\,\Omega \tag{7.6}$$

The sizes of the chopper switches were chosen such that the noise resistance,

$$r_{S1} = 50\,\Omega \tag{7.7}$$

, was smaller than other noise resistances.

All adds up to an equivalent noise resistance of

$$r_{eq} = 33 + 2 \cdot 50 + 2 \cdot 440 + 2 \cdot 133 = 1300\,\Omega$$

which is 14 times higher than the required noise resistance for a 10+5 bit FP-ADC. According to equation 6.18 this noise corresponds to the quantization noise of an 8+5-bit FP-ADC. This means that the noise will not meet the requirements of a 10+5-bit FP-ADC.

7.2 Sampling capacitors

The size of the sampling capacitor was chosen based on three criteria; thermal noise, charge errors, and mismatch.

We want the quantization noise (equation 2.25) to dominate over the thermal noise (equation 2.9) by a factor of F_n

$$\frac{V_{FS}^2}{12 \cdot 2^{2n}} \geq F_n \cdot \frac{kT}{C_s}$$

This gives the size of the capacitor to

$$C_s \geq \frac{12 \cdot F_n \cdot 2^{2n} \cdot kT}{V_{FS}^2} \tag{7.8}$$

The charge error from the sampling switch is transferred to the sampling capacitor. The charge error will be cancelled by a dummy switch, leaving only a residual error. Let us assume the residual charge error is $1/F_{\text{sw}}$ of the total charge error q_{sw}. So if we want the residual charge error to be less than $\frac{1}{2}$LSB then

$$\frac{1}{F_{\text{sw}}} |q_{\text{sw}}| \leq \frac{C_s V_{FS}}{2^{n+1}} \tag{7.9}$$

Inserting equation 4.13 and solving for C_s gives a sample capacitor size of

$$C_s \geq \frac{2^{n+1}}{M} \frac{V_{DD}}{V_{FS}} W_{\text{sw}} \left(\frac{L_{\text{sw}} C_{ox}}{4} (1 - \frac{2V_{TH}}{V_{DD}}) + C_{ov} \right) \tag{7.10}$$

Here it is assumed that bottom-plate sampling is used with $v_I = V_{DD}/2$.

The matching of the input capacitor with the amplifying capacitor in the sampling stage has to be accurate not to introduce distortion when switching among the rows. According to equation C.8 the gain mismatch of the sampling stage becomes

$$\sigma(\frac{\Delta C}{C}) = \frac{A_C}{2\sqrt{C_s/C_{poly}}} \frac{\sqrt{3}}{\sqrt{2}} \tag{7.11}$$

when using physical capacitor sizes of $(WL)_s = C_s/C_{poly}$ and $(WL)_a = C_s/C_{poly}/2$. Now it is possible to calculate the requirement of the capacitor

Noise	130 fF
Charge	300 fF
Matching	50 fF

Table 7.1: Minimal C_s according to equations 7.8, 7.10 and 7.12.

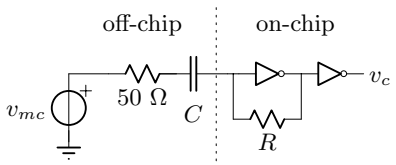

Figure 7.2: Clock input buffer.

size in terms of matching

$$C_s \geq C_{poly} \frac{3}{8} \left(\frac{A_C}{\sigma(\frac{\Delta C}{C})} \right)^2 \tag{7.12}$$

By entering the physical parameters of the CMOS process into the equations and setting $F_n = F_{sw} = 10$ the minimal sampling capacitor size was obtained. The sampling capacitor size was chosen to be 300 fF. The results from the equations 7.8, 7.10 and 7.12 can be seen in table 7.1.

7.3 The phase generator

The phase generator is the component that drives all the switches in the design. The phase generator is made up by two components; the clock buffer and the phase generator. The clock buffer's task is to make sure that the duty cycle of the on-chip clock v_c is 50% regardless of the DC-level of the off-chip master clock v_{mc}. It also makes sure that the on-chip clock has sharp edges even though the master clock is a sinusoid. The phase generator generates the four clock phases needed in the FP-ADC.

The clock input buffer is shown in Figure 7.2. The buffer contains two inverters, where the N- and P-transistors are sized in such a way that the delay is equal for the falling and the rising edges. This preserves the duty cycle. To make sure that the clock buffer triggers exactly at the middle of

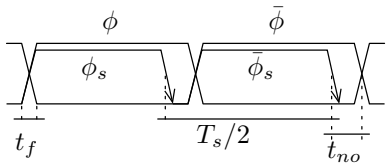

Figure 7.3: The clock phases used in the FP-ADC. The arrows indicate the sampling instants.

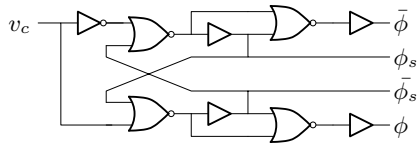

Figure 7.4: The phase generator.

the input signal a DC-blocking capacitor has to be inserted in series with the clock input. The DC level is provided by the feedback resistor over the first inverter. This way the clock input becomes insensitive to the DC-level of the clock source. The resistor and the DC-blocking capacitor form a high-pass filter for the clock input. Therefore the pole, $-1/RC$, of the high-pass filter must be lower than the lowest sampling frequency. The capacitor was placed off-chip to make sure that the pole could take an arbitrarily low frequency.

Normally the clock phases in an SC-circuit are non-overlapped [60]. However this scheme eats up some precious nanoseconds. To speed things up let us investigate what is important for the phase pattern. The most important for the phase pattern is that the falling edge of the sampling phase is ahead of all other phases. This ensures that bottom-plate sampling is maintained, as described in section 4.2.3. All other phase transitions can happen at the same time instant. The principle is illustrated in Figure 7.3. It also has the benefit of reducing the number of clock phases from six to four.

The circuit that implements the clock phases is shown in Figure 7.4. In the figure the on-chip clock enters from the left and the inverse clock is generated in the inverter. The NOR-gate pair and the first buffer-pair form a bi-stable latch that generates the non-overlapping complementary sample phases, ϕ_s and $\bar{\phi}_s$. The phases $\bar{\phi}$ and ϕ are essentially the delayed and inversed versions of ϕ_s and $\bar{\phi}_s$ respectively. By bridging the buffer in the

$t_f(\phi_s)$	110 ps
$t_f(\phi)$	430 ps
t_{no}	350 ps

Table 7.2: The simulated characteristics of the phase generator, the chip load included.

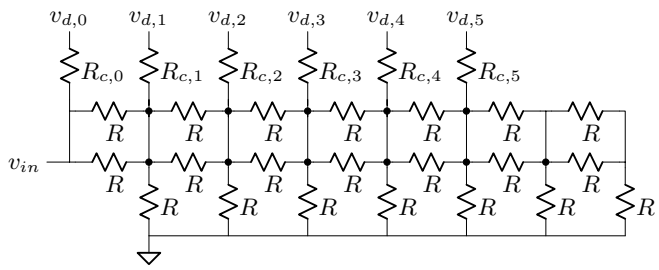

Figure 7.5: The resistive divider.

bi-stable latch with a NOR-gate the rising edges of ϕ and $\bar{\phi}$ are advanced in time and synchronized with the rising edges of ϕ_s or $\bar{\phi}_s$ respectively.

The simulated time constants of this circuit are presented in table 7.2. For a sampling rate of 100 MS/s the sampling period is 10 ns. The blocks in the FP-ADC need to settle when ϕ_s is high, thus the settling time need to be faster than

$$t_s < \frac{T_s}{2} - t_{no} \approx 4.5 \text{ ns} \tag{7.13}$$

7.4 Resistive divider

The resistive divider was implemented as an R-2R ladder. The reason why we chose this topology is that it is highly regular, has low impedance and is fast. The resistive divider used is shown in Figure 7.5. The R-2R ladder has seven terminals but the first is used as a dummy structure to improve the gain and delay matching. Resistors have been added in series with the terminals to equalize the delay through the R-2R ladder.

According to simulations, shown in Figure 7.6, the requirement on the

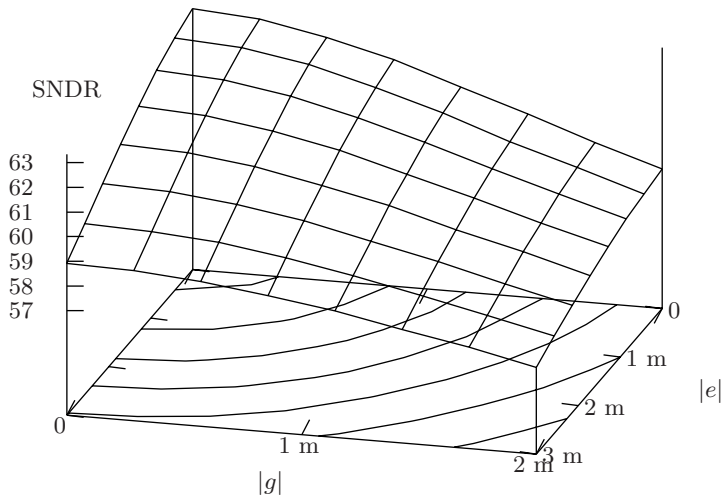

Figure 7.6: A plot of the effect on the distortion from the relative gain error e and the offset error g in the 10+5-bit FP-ADC rows. At the base of the plot are the surface contours that correspond to the tick-marks on the SNDR axis.

	Simulated	Theoretical
t_d	11.5 ps	11.5 ps
$\sigma(\frac{\Delta G}{G})$	0.09 %	0.11 %
$f_{-3\text{dB}}$	4.8 GHz	

Table 7.3: The simulated characteristics of the divider, compared to the theoretical estimates (equation 4.51 and 4.49). The given delay time and the gain mismatch are between two adjacent terminals.

standard deviation of the gain error, to get an SNDR > 60 dB, is

$$|e| < 0.25\,\% \tag{7.14}$$

A reasonable value for the matching error of the ladder was chosen to be 0.1%, leaving a mismatch requirement for the gain stage to be 0.23%, according to equation 6.10.

The unit resistance was chosen to be 100 Ω, the lowest possible with respect to the input current and also to the parasitic wire resistances. The area of the resistors was calculated from equation 4.49 to give a relative error of around 0.1 %. The matching of resistors is poor, which means that the area of the resistors must be large, hence the delay between the terminals becomes significant (11.5 ps according to equation 4.51).

The simulated and theoretical characteristics of the divider are presented in table 7.3. The delay between two adjacent terminals is 11.5 ps, giving a timing mismatch between the first and the last terminal to be 60 ps. Therefore the delay equalizing resistors R_c was used to eliminate this systematic delay error.

7.5 The gain stage

The gain stage is the most sensitive part of the FP-ADC. The signals here are weak and in general very sensitive to disturbances why special care must be taken. The most important parameters are the gain matching and the noise. Also important are the speed, for settling, and the PSRR, due to the single ended nature of the design.

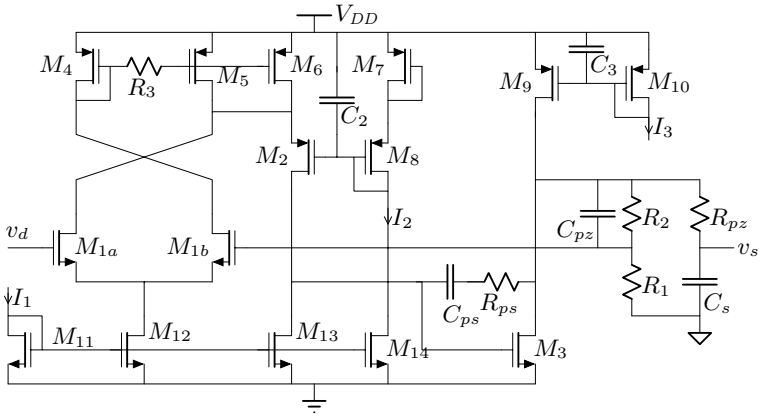

Figure 7.7: The gain stage.

Amplifier

The three-stage amplifier shown in Figure 7.7 was used as the gain stage. The input stage of the amplifier is an ASCS-stage (the identical transistors M_{1a} and M_{1b}) followed by a CG-stage[1] (M_2). The output stage is a CS-stage (M_3). The feedback is the resistive voltage divider of R_1 and R_2.

Matching

The gain matching is determined by the area of the feedback resistors. The larger area is the better the matching becomes. However, when the area of the feedback resistors are getting larger the signal is getting delayed through the feedback due to the parasitic capacitance of the resistors. Ultimately this delay sets the speed of the gain stage. To minimize the area of the feedback resistors, consequently also the parasitic capacitance, the mismatch of the divider was chosen in such a way that the gain stage dominated the total mismatch. By choosing the area of the feedback resistors R_1 and R_2 to be the same the total area was minimized. The gain of the stage was set to 17 to facilitate the use of unit resistors for good matching. The realization of the feedback is shown in Figure 7.8. According to equation 4.47 the mismatch

[1] Also referred to as a differential pair with a folded cascode.

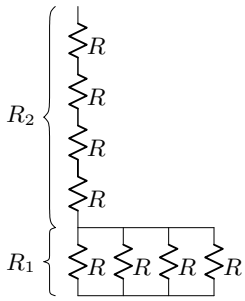

Figure 7.8: Realization of the feedback divider in the gain stage with unit resistors. This realization gives a gain of 17 and the same total area for R_1 and R_2.

is

$$\sigma(\frac{\Delta G}{G}) = \frac{16}{17}\frac{A_R}{\sqrt{WL}} \tag{7.15}$$

where WL is the area of R_1 (and R_2).

Noise

The noise from the amplifier comes from the feedback resistors and the input stage. The noise spectrum, generated from inside the amplifier, referred to the input of the amplifier is

$$S_G = 4kT\left(\frac{2}{3}\frac{2}{g_{m1a}} + \frac{2}{3}\frac{g_{m4}+g_{m5}+g_{m6}}{g_{m1a}^2} + \left(\frac{g_{m5}+g_{m6}}{g_{m1a}}\right)^2 R_3\right) \tag{7.16}$$

The noise spectrum, generated from the amplifier feedback, referred to the input of the amplifier is

$$S_{FB} = 4kT\,(R_1 \parallel R_2) \tag{7.17}$$

The flicker noise has been neglected due to the AZ i.e. the CDS will render the flicker noise to be insignificant with respect to the thermal noise.

The first term in equation 7.16 and 7.17 were considered the dominant noise sources in the gain stage, the noise of the ASCS-stage noise and the feedback resistors. These two dominant noise contributions have been used in the design considerations of the FP-ADC, section 6.3.

In addition to the dominant noise sources the current mirror $M_4 \rightarrow$

(M_5, M_6) contributes to the input referred noise. According to equation 7.16 the noise contribution will be small as long as the $\sum g_m$ of the current mirror is less than g_{m1a}. The thermal noise of the resistor R_3 also contributes to the input noise. This noise contribution is likely to become dominant. In order to avoid this, the resistor R_3 should be removed. Why R_3 was included anyway is explained in the bias section below.

Frequency compensation

The amplifier is compensated mainly by a phantom-zero compensation. The phantom-zero capacitor at the feedback, C_{pz}, is especially effective as it bypasses the phase delay of R_2. The phantom zero compensation is backed up by a moderate pole-split compensation over the CS-stage (C_{ps}). The right half plane zero of the pole-split is shifted to infinity by R_{ps}. The resistor R_{pz} in series with the sampling capacitor is also a phantom zero compensation but not as effective. This resistor reduces the noise bandwidth and also improves the stability of the circuit.

Bias

The bias currents for the signal transistors (M_{1-3}) in the amplifier are set by the two current mirrors; $M_{11} \rightarrow (M_{12}, M_{13})$ and $M_{10} \rightarrow M_9$. The bias current in the ASCS-stage was determined by the noise requirement in section 7.1. To maximize the speed the bias current at the output was chosen large enough to prevent slewing in the output stage [65]

$$I_3 > \frac{V_{FS}}{2}\left(\frac{1}{2^m R_1} + \frac{\pi f_s C_s}{G_{SH}}\right) \tag{7.18}$$

where G_G and G_{SH} are the gain of the gain stage and SH stage respectively. A low-pass filter is formed by R_3 and the combined gate-source capacitances of M_5 and M_6. This way the gate voltages of M_5 and M_6 are kept constant and no signal travels through the current mirror, thus the pole-zero pair of the current mirror is removed. The bias voltage to the gate of M_2 is provided by the two diode-coupled transistors M_7 and M_8.

Systematic offset

The major source of systematic offset in the amplifier is mismatch in the drain currents and the drain-source voltages of the ASCS-stage [67].

The currents from the current sources M_6 and M_{13} must be equal, otherwise a residual current will be injected into the ASCS-stage. A ratio of the current mirrors $M_4 \rightarrow (M_5, M_6)$ and $M_{11} \rightarrow (M_{12}, M_{13})$

$$\frac{\left(\frac{W}{L}\right)_{12}}{2\left(\frac{W}{L}\right)_{13}} = \frac{\left(\frac{W}{L}\right)_4}{\left(\frac{W}{L}\right)_6} \tag{7.19}$$

establishes the same currents from M_6 and M_{13}. Then no DC current will be injected into the ASCS-stage.

The drain voltage of M_{1a} is set by the gate voltages of M_2 and the two diode-coupled transistors M_7 and M_8

$$V_{D1a} = V_{DD} + V_{GS7} + V_{GS8} - V_{GS2} \tag{7.20}$$

The diode coupled transistors M_7 and M_8 are replicas of the transistors M_6 and M_2 respectively; the same size and the same drain currents (provided by the current mirror $M_{11} \rightarrow (M_{13}, M_{14})$). This way V_{GS7} and V_{GS8} track V_{GS6} and V_{GS2} respectively. Then

$$V_{D1a} = V_{DD} + V_{GS6} = V_{DD} + V_{GS4} = V_{D1b} \tag{7.21}$$

In this way the systematic offset in the amplifier was reduced. The residual offset comes from the finite output resistance and the non-identical drain-source voltages in the current mirror $M_{11} \rightarrow (M_{12}, M_{13}, M_{14})$. This means that the current mirroring is not exact, thus also the matching of the drain currents in the ASCS-stage.

PSRR

A variation in the supply voltage can influence the currents and voltages in the amplifier and hence the output voltage [67]. The amplifier is single-ended why a change in the output voltage will influence the signal directly. The PSRR can be seen as an offset dependence of the supply voltage. Then it is clear that it is important to remove the systematic offset and the dependence of the bias point from the supply voltage.

First the bias currents I_1 and I_3 are generated from an independent and noise-free power supply. Also the current mirrors were made as long as possible to maximize their output resistance of the current mirrors. This way the bias currents are kept as constant as possible. Also the de-coupling

G_G	17
$\sigma(\Delta G/G)$	0.24 %
$\bar{v}_{n,in}$	0.12 mV
$\lvert\Delta t_d\rvert$	$<$ 13 ps
t_s	5.6 ns
f_{-3dB}	330 MHz
v_{off}	$<$ 3.5 mV
PSRR	65 dB
THD	$<$ -54 dB
Current	3.5 mA

Table 7.4: The simulated characteristics of the gain stage.

capacitors C_2 and C_3 were introduced to stabilize the bias voltages at high frequencies.

When the systematic offset was minimized as described in the previous section, the biggest contribution to the PSRR comes from the finite output resistance and the non-identical drain-source voltages in the current mirror $M_{11} \rightarrow (M_{12}, M_{13}, M_{14})$.

Simulation results and comments

The characteristics of the gain stage are presented in table 7.4. The values for the gain mismatch, the offset, and the delay time have been acquired by means of Monte-Carlo simulations. The gain was chosen to be 17 since the absolute gain is not important. The gain mismatch is dominated by the feedback resistors as expected, and is right on target. The noise turned out to be much higher than expected, which will be discussed in the next chapter. The delay time mismatch between gain stages will be the dominant gain error source in the FP-ADC. Using this figure in equation 6.5 gives a maximum input frequency of 75 MHz. The settling time gives, together with the non-overlap time of the clock phases, a sampling frequency of 80 MHz. The input offset gives DC offset at the output of $<$ 60 mV, which is acceptable from a signal swing point of view. The 65 dB PSRR value means that the supply voltage must vary $>$100 mV for a 1 mV variation on the output. The THD was measured in a simulation with a 50 MHz full swing input signal. For lower frequencies the THD is much less.

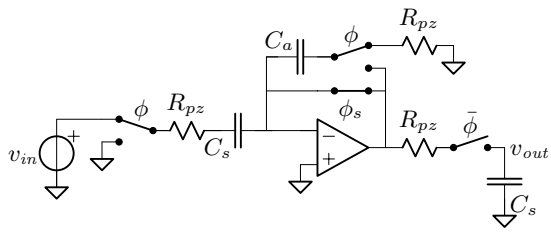

Figure 7.9: The sample-and-hold stage.

7.6 The sample and pipeline stages

In this section the sample-and-hold stage and the RSD 1-bit pipeline stages are described. Since the two types of stages use the same type of amplifier, the functionality of the stages are described individually while the amplifier type is described afterwards.

7.6.1 The sample-and-hold stage

The sample-and-hold stage is shown in Figure 7.9. The operation of the sample-and-hold stage is as follows. During the sampling phase, when ϕ and ϕ_s are high, the input signal v_{in} charges the sampling capacitor C_s. At the same time the amplifying capacitor C_a is discharged. When ϕ_s goes low the sampling switch opens and the charge at the negative input of the amplifier is preserved. When ϕ goes low the sample-and-hold stage enters its amplifying phase and C_s is discharged while C_a is connected to the output. In the amplifying phase the total charge at the negative input of the amplifier is constant. Therefore the charge of C_s is transferred to C_a. The ratio of C_s and C_a determines the gain of stage

$$v_{out} = \frac{C_s}{C_a} v_{in} \tag{7.22}$$

and in our case $C_s = 2C_a$.

7.6.2 The pipeline stage

The pipeline stage is shown in Figure 7.10. The operation of the pipeline stage is as follows. During the sampling phase, when ϕ and ϕ_s are high, the

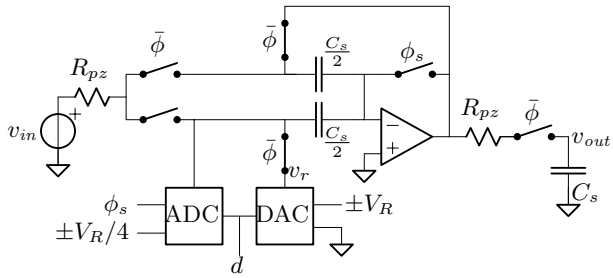

Figure 7.10: The RSD 1-bit pipeline stage.

input signal v_{in} charges the two sampling capacitors, $C_s/2$. When ϕ_s goes low the sampling switch opens. From this moment the charge at the negative input of the amplifier is preserved. At the same time the sub-ADC samples the input signal. The digital output is used in the sub-DAC to produce v_r. When ϕ goes low the sample-and-hold stage enters its amplifying phase. The upper $C_s/2$ is connected to the output while the lower is connected to v_r. The sampled charge of the lower $C_s/2$ is then transferred to the upper, thus doubles the input voltage. The output voltage of the DAC is subtracted at the output, giving the transfer function

$$v_{out} = 2v_{in} - v_r \tag{7.23}$$

as per equation 3.3.

7.6.3 Sample and pipeline stage amplifier

The same topology has been used for the sample and pipeline stage amplifier. The difference between the two amplifiers is that the transistor sizes and bias currents have been scaled down in the pipeline stage.

The requirements on the sample-and-hold stage are much more relaxed compared to the gain stage. It is the benefit of the FP-ADC architecture that from here on the requirements are relaxed. However, some effort have been made regarding the speed, offset and PSRR. The random mismatch of the capacitors is so small that they can be neglected. This has already been discussed in section 7.2.

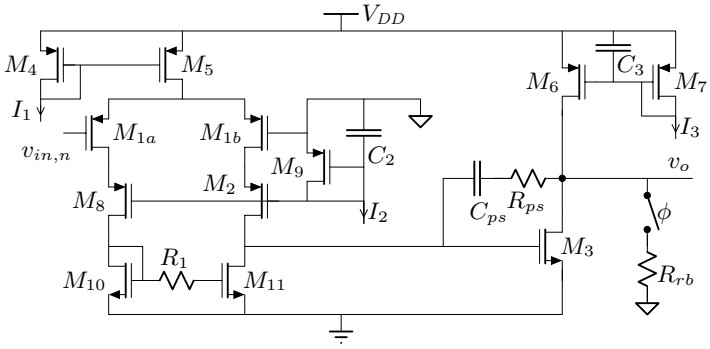

Figure 7.11: The sample/pipeline stage amplifier.

Amplifier

The three-stage amplifier used is different from the gain stage, see figure 7.11. Here the second stage, the CG-stage, M_2, is not folded but uses the same current as the ASCS-stage. This can be done because the variation of the input voltages is very small. This way the current is reused and, more important, the CMRR is improved.

The amplifier current bias is set by the two current mirrors, $M_4 \rightarrow M_5$ and $M_7 \rightarrow M_6$. The gate voltages of M_2 and M_8 are set by the diode coupled transistor M_9 and I_2 ($I_2 = I_1/2$) in such a way that $V_{DS1} = V_{GS1}$. This ensures that the ASCS-stage transistors are always biased in the saturation region. The bias currents in the ASCS-stage are set equal by the current mirror $M_{10} \rightarrow M_{11}$.

Compensation

The amplifier is compensated during the amplifying phase using a pole-split capacitor C_{ps}. The right half-plane zero is moved to infinity by means of the resistor R_{ps}. During the sample phase the amplifier is coupled as a follower why some extra compensation is needed. This extra compensation is given by the switch S which enables the resistive broad-banding compensation by shunting the output with R_{rb} to ground. The shunt resistor R_{rb} lowers the loop gain and makes the amplifier more stable.

Also the resistor R_{pz} has been included in series with the sampling capacitor to work as a phantom zero compensation. The resistors improve the

G_{SH}	2
$\Delta G/G$	0.1 %
$\sigma(\Delta C/C)$	0.08 %
$\bar{v}_{n,in}$	0.55 mV
v_{off}	< 7 mV
PSRR	38 dB
t_s	5.2 ns
Current	2.1/1.5 mA

Table 7.5: The simulated characteristics of the sample/pipeline stage.

stability of the circuit and also reduce the sample noise bandwidth.

CMRR and offset

With the SC-operation another issue of CMRR, other than minimizing the systematic offset, is important. It is the fact that when the ground supply is moving then also the drain voltage of M_{1a} is moving as well. Then the charge across C_{gs1a} is changed as well. This is no problem in the sampling phase as the charge is provided from the output of the amplifier. But when the amplifier is in the amplifying phase this charge is taken from C_a. In that case the CMRR is

$$\text{CMRR} = \frac{C_a}{C_{gs1a}} \frac{C_s}{C_a} = \frac{C_s}{C_{gs1a}} \tag{7.24}$$

This effect can be removed by making V_{D1a} constant with the CS-stage M_8. M_8 is tracking M_2 and therefore the systematic offset error is minimized as well.

The systematic offset of the amplifier itself is minimized when improving the CMRR. Further both the systematic and random offsets are cancelled by AZ. What is not removed is the offset generated by charge errors from the sampling switch. This is discussed in section 7.2. The CMRR will then be dominated by C_{OFF} of the sampling switch

$$\text{CMRR} = \frac{C_s}{C_{OFF}} \tag{7.25}$$

The characteristics of the sample and pipeline stage are presented in table 7.5. The values for the gain mismatch, the offset, and the delay time have been acquired by Monte-Carlo simulations. Only the current consumption

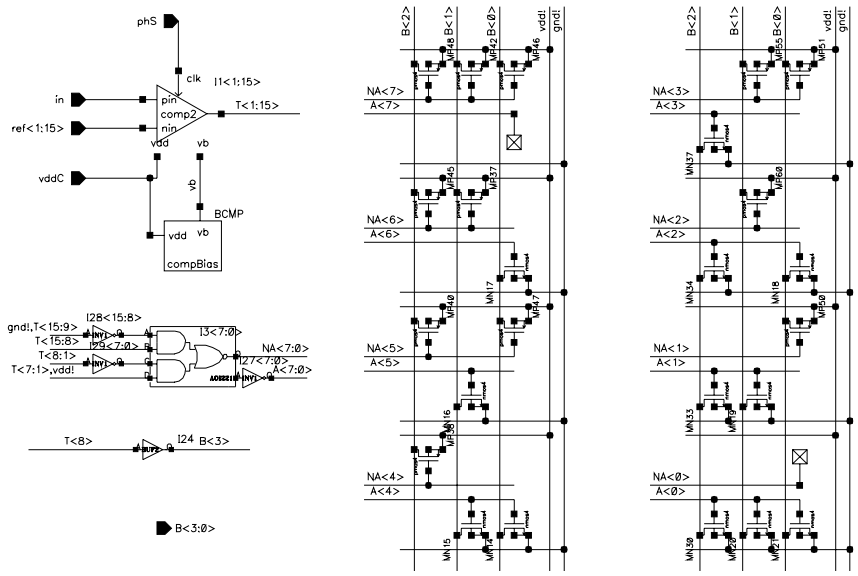

Figure 7.12: The 4-bit flash ADC. The resistor ladder is omitted.

differs significantly between the sample and pipeline stage. A difference in the input noise was expected, but was alleviated by the noise contribution from R_1.

7.7 The flash ADC

A 4-bit flash is used at the end of the pipelined ADC. The size of the flash was chosen to minimize the current consumption according to

$$(2^{n_F} - 1)I_C < I_P \tag{7.26}$$

where I_C and I_P are the current consumptions of a comparator and a pipeline stage respectively. n_F is the number of bits of the flash. Solving n_F gives the number of bits in the flash converter

$$n_F < \ln\left(\frac{I_P}{I_C} + 1\right) / \ln 2 \tag{7.27}$$

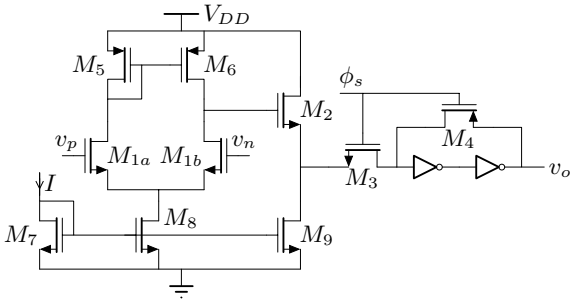

Figure 7.13: The comparator.

With the numbers given in the tables 7.5 and 7.6 this gives a 4-bit flash.

The design of this flash is straightforward. There are 15 comparators comparing the input signal to 15 voltage references generated from a resistor ladder. The outputs from the comparators forms a thermometer code. This code is converted into an address code that addresses a small ROM. Finally the ROM converts the address into a binary code. To reduce the ROM size by a factor of two the MSB is taken directly from the thermometer code and the address must therefore be folded. The design is shown in Figure 7.12.

There is no bubble correction [89] used in this flash as the 3σ (offset plus noise) variation of the comparators is less than the threshold intervals

$$3 \cdot \sqrt{2} \cdot \sqrt{16^2 + 22^2}\,\mathrm{mV} = 115\,\mathrm{mV} < 125\,\mathrm{mV} = \frac{2\,\mathrm{V}}{2^4} \tag{7.28}$$

according to the simulated comparator data in table 7.6. The variations in the reference voltages are considered negligible.

7.8 The comparator

The comparator used throughout in the FP-ADC is a combination of an amplifier and a latch. For the fastest operation using only a latch would be the best. But as other things are also important, e.g. to minimize offset, noise and impulse leakage, the latch is preceded by an amplifier.

As is shown in Figure 7.13, the comparator input stage is a plain ASCS-stage (M_{1a} and M_{1b}) loaded with a current mirror ($M_5 \rightarrow M_6$). The output

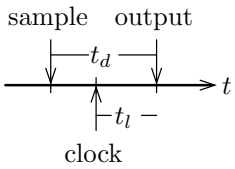

Figure 7.14: Illustration of the delay and the latency time of the comparator.

DC current	100 μA
t_d	770 ps
t_l	130 ps
$\overline{v_{off}}$	16 mV
$\overline{v_n}$	22 mV

Table 7.6: The simulated characteristics of the comparator.

voltage of the input stage is conveyed to the output stage via a CD-stage[2] (M_2) and an NMOS switch (M_3). The output stage is the two chained inverters.

When the clock ϕ_s is high the comparator is in its amplifying phase. In the amplifying phase the comparator is an amplifier with no latching. When the clock falls the NMOS (M_3) switch turns off and output stage is latched with the PMOS switch (M_4). This is the sampling instant. Now, in the sampling phase when the clock is low, the latch output goes quickly to the supply rails and holds its state due to the positive feedback.

The comparator is fairly slow during its amplifying phase, thus giving a long delay. But the latching of two inverters is very fast, meaning that the latency is short[3]. The delay and the latency time of the comparator are illustrated in Figure 7.14. In this way the comparator exploits the combination of the low slope of the exponential step response and the fact that the actual threshold level does not need to be very accurate due to the RSD 1-bit operation.

The simulated characteristics of the comparator are summarized in table 7.6

[2]common-drain stage or voltage follower

[3]The delay is the time it takes for the signal at the input to arrive at the output while the latency is the time from the sampling clock edge until the output is valid.

7.9 The controller

The controller consists of five parts; the thermometer code logic, the blocking code logic, the CMOS switch drivers, the exponent demultiplexer, and the data selection logic. The controller schematic is shown in Figure 7.15.

The thermometer code $\mathbf{A}$ is generated from the vectors $\mathbf{H}$ and $\mathbf{L}$, the column of sub-ADC comparator outputs in first pipeline stage,

$$\mathbf{A} = \mathbf{H} \vee \overline{\mathbf{L}} \tag{7.29}$$

The thermometer code represents the logarithmic magnitude of the input signal. The row that is to be selected corresponds to the highest index k where $A_k = 1$. To ensure this a blocking code $\mathbf{B}$ is generated from the thermometer code. The blocking code is generated in such a way that all indexes in the blocking code below the highest index k where $A_k = 1$ are set to 1

$$B_k = A_{k+1} \vee B_{k+1}, \quad B_5 = 0 \tag{7.30}$$

A_k and B_k are then used in the CMOS switch drivers for gating the system phases to the CMOS switch clock phases $\phi_{\text{ctrl},k}$ and $\bar{\phi}_{\text{ctrl},k}$

$$\phi_{\text{ctrl},k} = \bar{\phi} \wedge A_k \wedge \overline{B_k} \tag{7.31}$$

$\mathbf{A}$ and $\mathbf{B}$ are also used to calculate the exponent and to forward H and L from the correct row to the combiner.

7.10 Synchronizing and digital error correction

Before the outputs from the pipeline stages can be combined they need to be synchronized. Each sample is delayed by a half clock cycle for every pipeline stage. Therefore the outputs from the early pipeline stages need to be delayed so that the data arrives at the combiner at the same time. The timing of the data is shown in Figure 7.16. The corresponding digital circuit is shown in Figure 7.17.

The data arriving from the sub-ADC in the pipeline stages are the direct output from the comparators. The data needs to be converted into RSD 1-bit numbers. This is done by the small circuit in the upper right hand corner in

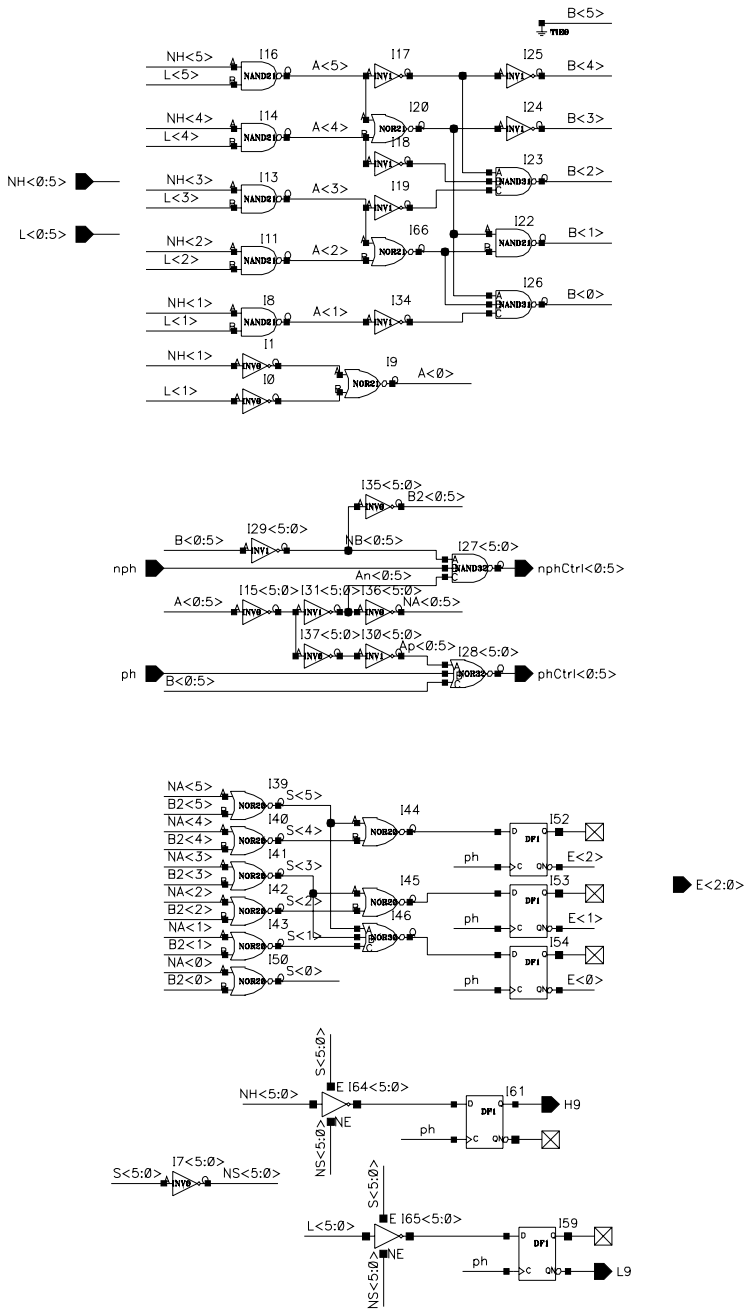

Figure 7.15: The controller.

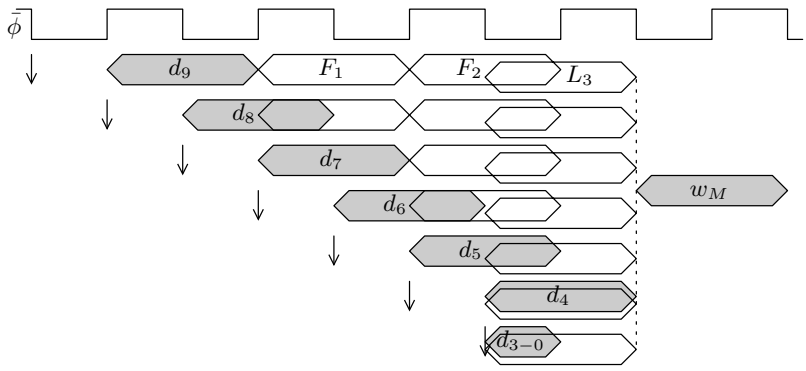

Figure 7.16: Timing diagram of valid data from the blocks of the FP-ADC. The gray boxes represent the output and the bits from the pipelines stages. The white boxes represent the delays of the flip-flops and the latches in the synchronizer. The arrows show the sampling instant for the pipeline stages.

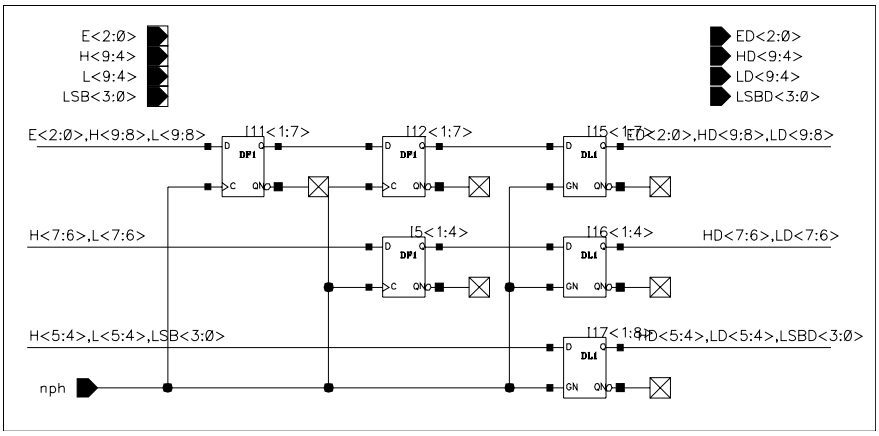

Figure 7.17: Synchronizer to de-skew the output from the pipeline stages.

H_k	L_k	d_k
0	0	0.0 (0)
0	1	0.1 (0.5)
1	1	1.0 (1)
1	0	N/A

Table 7.7: The comparator outputs and its corresponding binary (decimal) value

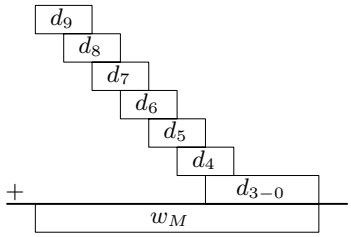

Figure 7.18: The principle of the combiner.

Figure 7.19. The truth table of this conversion is found in table 7.7, together with the corresponding decimal number. Then an addition is performed to combine the data and do error correction. The bit-wise addition is

$$(c_k, w_{M,k}) = d_{k,0} + d_{k+1,-1} + c_{k-1} \tag{7.32}$$

where c_k is the carry and $w_{M,k}$ is the bit k in the mantissa. The principle of the addition is illustrated in Figure 7.18, where the data from the pipeline stages are shifted and added.

The complete combiner schematic is shown in Figure 7.19.

7.11 Reduction of the switching noise

Since the chip design is single ended and highly active in terms of switching, it is of great importance to reduce the switching noise on the power supply. The following strategies [90] were used to reduce switching noise:

- Reduce the bond wire inductance by using many pins for the voltage supply and the voltage references.

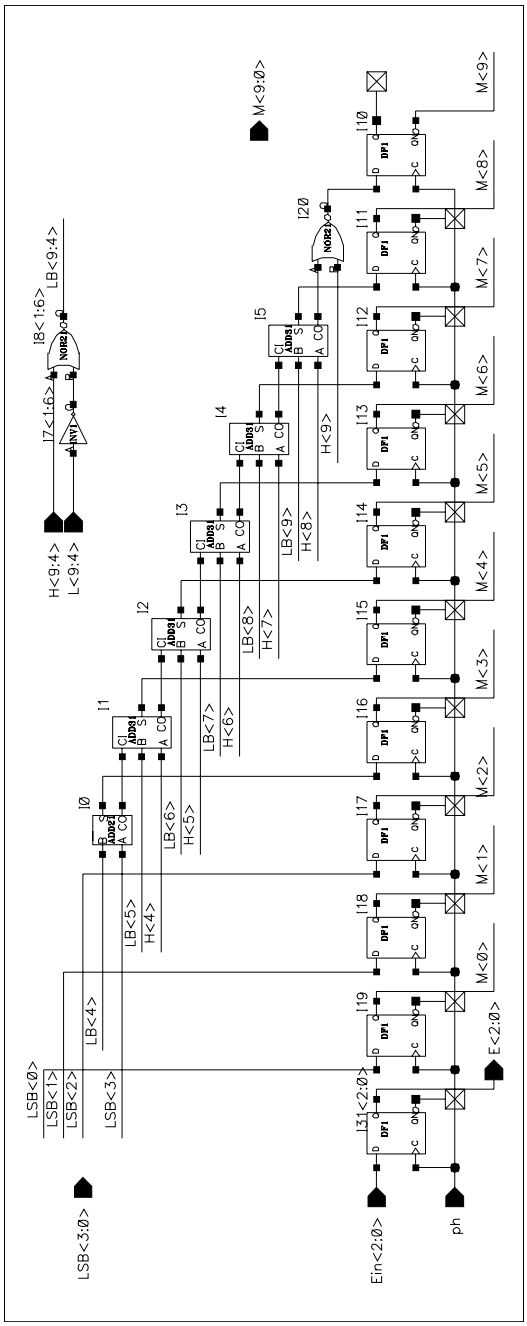

Figure 7.19: The combiner and digital error correction.

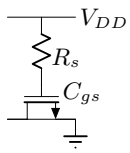

Figure 7.20: The unit decoupling capacitor, a gate capacitor $C_{gs} = 1.4\,\text{pF}$ and a series resistance $R_s = 1.5\,\text{k}\Omega$.

- Reduce the ground loop inductance by interleaving the V_{DD} and the ground supply pins.
- Separate the digital and the analog circuits, both by physical distance and galvanic isolation on the chip.
- The use of decoupling capacitors.

The use of decoupling capacitor can be hazardous. The bond-wires and the decoupling capacitor form a resonating circuit and in the case of low damping, ringing can go on for a long time. To counter power supply ringing a series resistance was put in series with the decoupling capacitor. Then the resonating energy is dissipated in the resistance and damps off the ringing. Also the series resistance is not visible for low frequencies thus having no impact on the voltage drop on the power lines.

Since the linearity of the decoupling capacitor is of little importance, the gate oxide of a transistor was used to implement the decoupling capacitor. The unit decoupling capacitor in Figure 7.20 was implemented and placed under the supply wires. This way the silicon area underneath the metal wires was exploited. The total number of unit decoupling capacitors was approximately 1500, which gives a total decoupling capacitance of 2 nF.

Chapter 8

Chip Measurements

Two FP-ADC chips have been manufactured, according to the proposed FP-ADC architecture, and measured.

The first version was an 8+4-bit FP-ADC to test the feasibility of the architecture. The actual design of this chip is different from what is presented in chapter 6 and chapter 7. The experience from the design and the measurements of that chip made it possible to pin-point the critical design issues when implementing a distributed FP-ADC with embedded SH. The results from this chip are presented in section 8.2. The second version was a 10+5-bit FP-ADC based on the knowledge from the previous chip and the measurement results are presented in section 8.3.

8.1 Test setup

The FP-ADC has been measured to acquire the performance in terms of dynamic range, nonlinearity and speed.

8.1.1 INL and DNL measurements

The static input referred error was measured by feeding a voltage ramp to the FP-ADC. The digital output is compared to a line drawn from end-to-end of the digital output ramp to remove the offset and gain error.

The INL and DNL were measured using a statistical method. By feeding the FP-ADC with a triangular wave the amplitudes of the samples are evenly distributed over the input range. The DNL then can be calculated from the

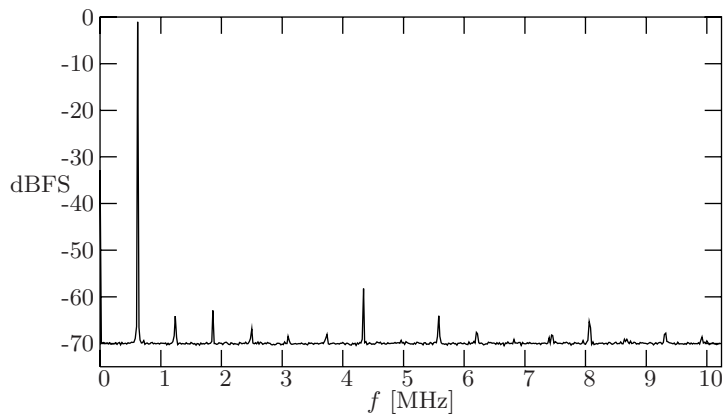

Figure 8.1: The average spectrum of the digital output signal for a sinus input, $f_{in} = 620$ kHz, $f_s = 20.5$ MHz, $A = -1$ dBFS.

dBFS	dB full scale
SNDR	signal-to-noise-plus-distortion ratio
SNR	signal-to-noise ratio
SDR	signal-to-distortion ratio, SDR=1/THD
SFDR	spurious-free dynamic range

Table 8.1: Explanation of the abbreviations used in the harmonic distortion plots.

histogram, hist(d) [7]

$$\mathrm{DNL} = \frac{\mathrm{hist}(d)}{\mathrm{mean}\left(\mathrm{hist}(d)\right)} - 1 \tag{8.1}$$

The INL is found from the integration of the DNL.

8.1.2 Harmonic distortion measurements

Three different sweeps of the harmonic distortion were performed; a sampling frequency sweep, two input frequency sweeps, and an input amplitude sweep.

The harmonic distortion was also measured. A single sinus was fed to the FP-ADC of various input frequencies/amplitudes and sampling frequencies. For every measurement point, 1000 fast Fourier transforms (FFTs) were calculated from the digital signal d. Every FFT has 1024 points. The power

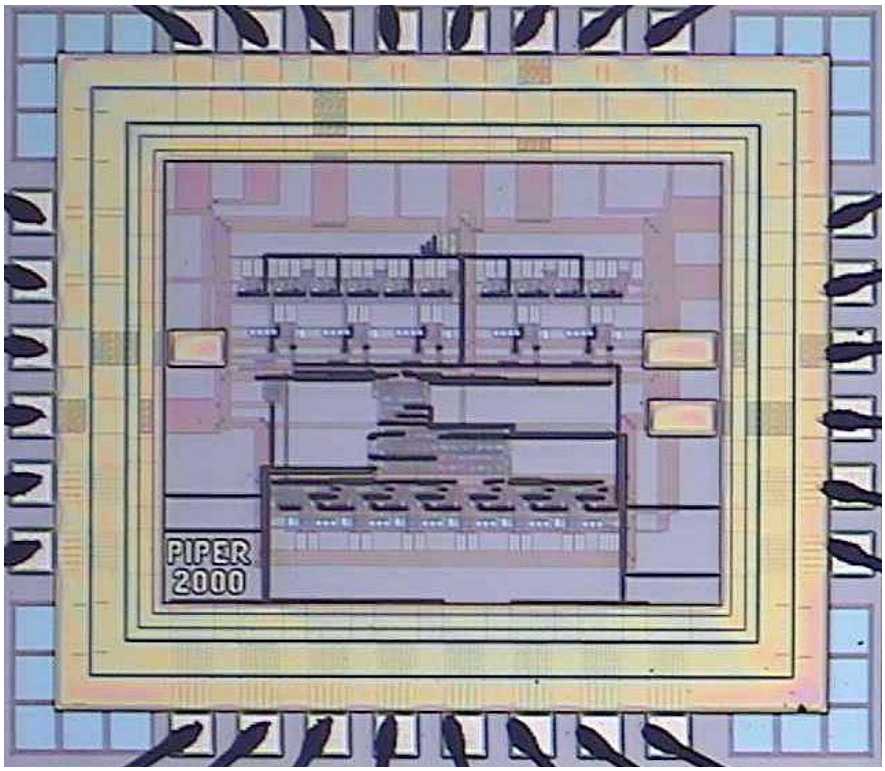

Figure 8.2: Chip photo of the 8+4-bit FP-ADC.

average of the 1000 FFTs was calculated to get an accurate spectrum. Such a spectrum from one of the measurements can be seen in Figure 8.1. In table 8.1 the abbreviations used in the plots are explained.

8.2 8+4-bit FP-ADC measurements

The first version, the 8+4 bit FP-ADC, was manufactured in a 0.35μm CMOS process with 3 metal layers and 2 poly-silicon layers. A chip photo of the 8+4-bit FP-ADC is presented in Figure 8.2.

To get the floating-point value the mantissa and exponent are combined according to

$$d = (w_M - 127.5) \cdot 2^{w_E} \tag{8.2}$$

Dynamic range	67 dB
SNDR	42 dB
SDR	< 45 dB
SFDR	< 55 dB
8-bit INL	2 LSB
8-bit DNL	0.7 LSB
Sampling rate	25 MHz
Input bandwidth	300 kHz
Input range	2.5 V_{FS}
Voltage supply	3.3 V
Current supply	10 mA
Core size	1×0.8 mm

Table 8.2: The measured overall characteristics of the 8+4 bit FP-ADC chip.

The measurement results are summarized in table 8.2. The measurements show that the dynamic range is 4 bits higher than the resolution.

The measurement of the DNL and INL can be seen in Figure 8.3 and 8.11 respectively.

Three different sweeps of the harmonic distortion were performed; a sampling frequency sweep, two input frequency sweeps, and an input amplitude sweep. The results can be seen in Figure 8.6, 8.5, and 8.4 respectively.

The measurements show that the maximal SNDR is 42 dB – corresponding to 6.7 bits. The amplitude sweep in Figure 8.4 shows that the total dynamic range is 67 dB – corresponding to 10.9 bits. Thus the dynamic range has been extended by 25 dB – corresponding to 4.2 bits.

According to the plot in Figure 8.6 the sampling rate $f_{s,-3\,\mathrm{dB}} = 25$ MHz and in Figure 8.5 the input signal bandwidth $f_{in,-3\,\mathrm{dB}} = 300$ kHz.

8.3 10+5-bit FP-ADC measurements

The second version of the 10+5 bit FP-ADC was manufactured in a 0.35μm CMOS process with 4 metal layers and 2 poly-silicon layers. A chip photo of the 10+5-bit FP-ADC is presented in Figure 8.7.

The input-output behavior can be seen in Figure 8.8. To get the floating-point value the mantissa and exponent are combined according to

$$d = (w_M - 511.5) \cdot 2^{w_E} \tag{8.3}$$

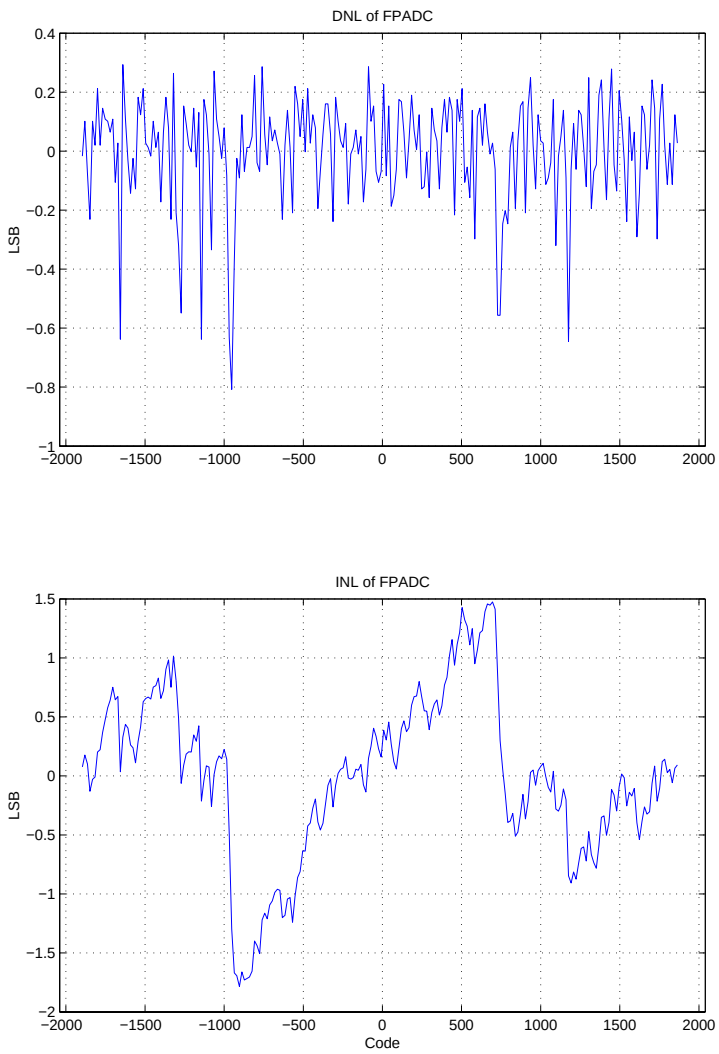

Figure 8.3: Plots of the DNL and the INL of the 8+4-bit FP-ADC. The equivalent 8-bit LSB scales are shown.

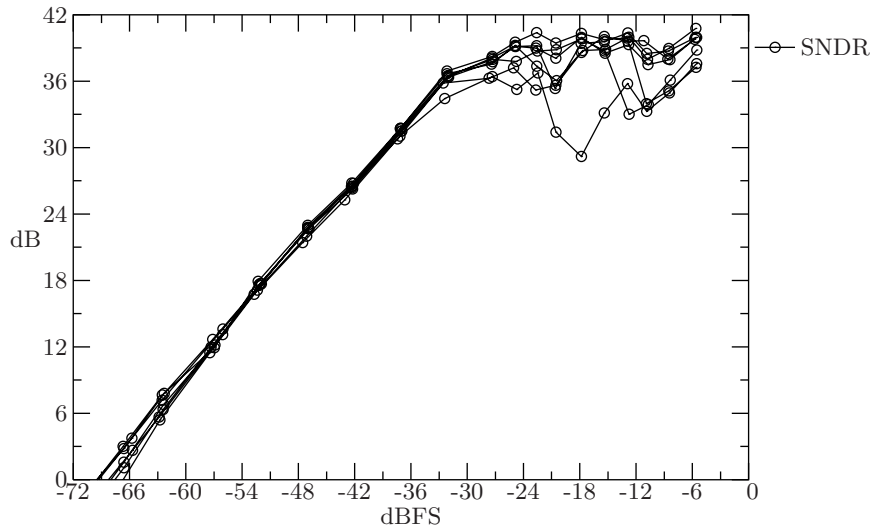

Figure 8.4: The signal to noise and distortion ratio of the 8+4-bit FP-ADC, when sweeping the input amplitude, $f_s = 10$ MHz, $f_{in} = 103$ kHz.

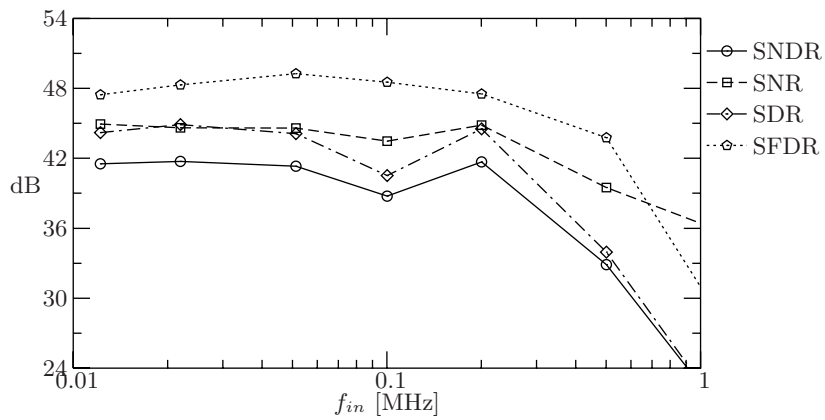

Figure 8.5: The signal to noise and distortion ratio of the 8+4-bit FP-ADC, when sweeping the sampling frequency, $f_s = 10$ MHz, $A = -3$ dBFS.

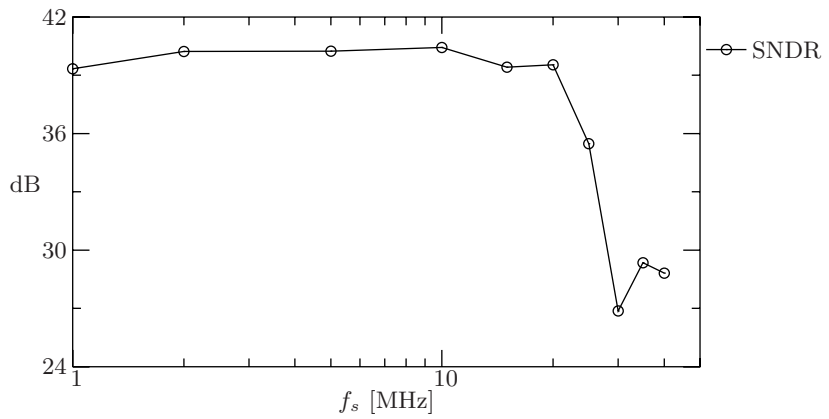

Figure 8.6: The signal to noise and distortion ratio of the 8+4-bit FP-ADC, when sweeping the sampling frequency, $f_{in} \approx 100$ kHz, $A = -3$ dBFS.

The FP-ADC has been measured to acquire the performance in terms of dynamic range, nonlinearity and speed. The results from the measurements are presented in section 8.3.1. The measurement results are summarized in table 8.3. The measurements show that the dynamic range is 5 bits higher than the resolution. The measurements also reveal three design limits; the noise and the distortion limits the ENOB to 6.8 bits and analog input bandwidth is 4.2 MHz. The causes behind the limits of the performance are discussed in section 8.3.2.

8.3.1 Measurement results

The static error of the 10+5-bit FP-ADC is shown in Figure 8.9. The static error is plotted against the input voltage (bottom axis) and the corresponding digital output (top axis). On the left axis the error is expressed in mV and on the right axis as LSBs.

The measurement of the DNL and INL can be seen in Figure 8.10 and 8.11 respectively.

In the figures a large jump is clearly visible at the codes of ±6100. Smaller jumps are also visible at half and quarter of this value. These jumps correspond to the row shifting levels of the controller. Interesting is the fact that all chips show the same behavior. This indicates that the gain mismatch between the rows is not random but deterministic and stems from parasitic

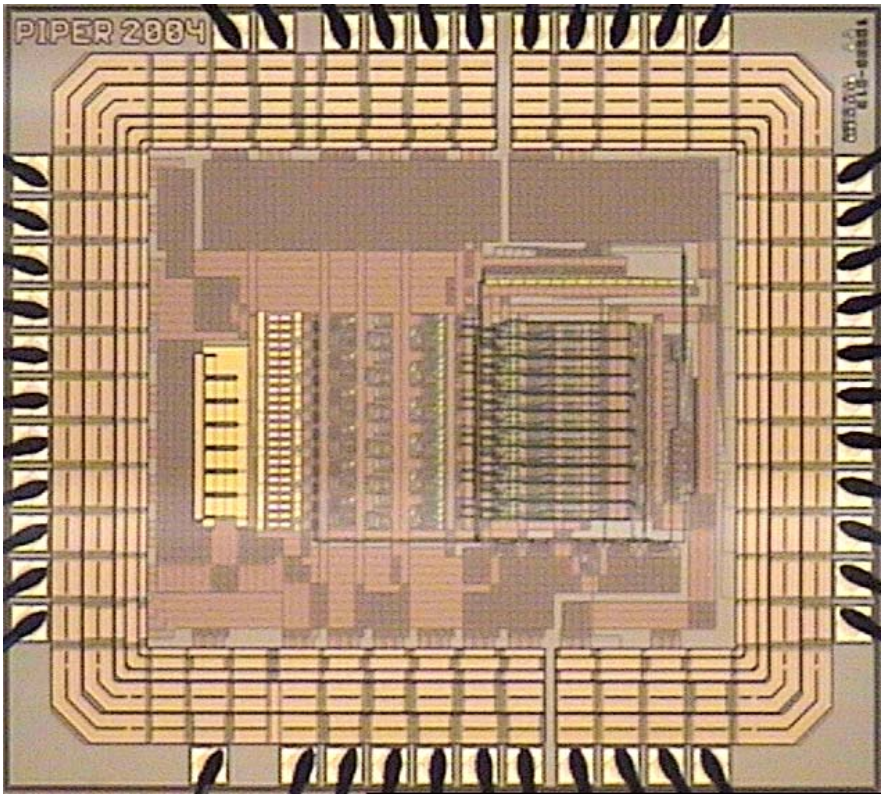

Figure 8.7: Chip photo of the 10+5-bit FP-ADC.

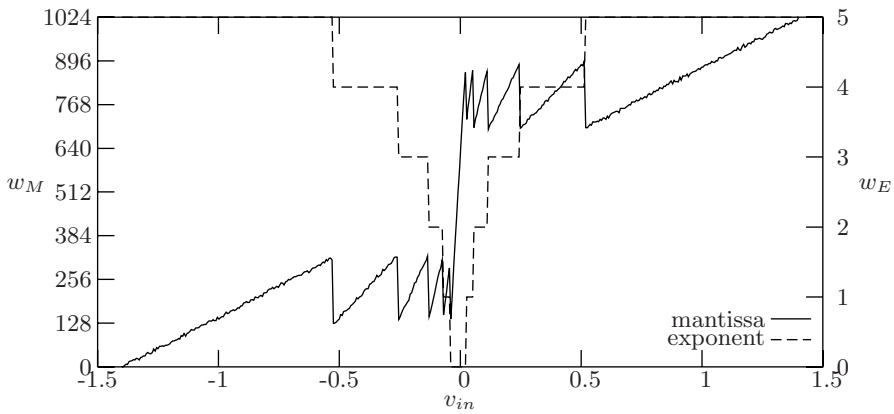

Figure 8.8: The input-output relation for the 10+5-bit FP-ADC.

Dynamic range	71.5 dB
SNDR	42.5 dB
SDR	51.7 dB
SFDR	57.4 dB
10-bit INL	3.7 LSB
10-bit DNL	0.5 LSB
Sampling rate	53 MHz
Input bandwidth	16.7 MHz
Input range	2.8 V_{FS}
Voltage supply	3.3 V
Current supply	100 mA
Core size	1.4×1.2 mm

Table 8.3: The measured overall characteristics of the 10+5 bit FP-ADC chip.

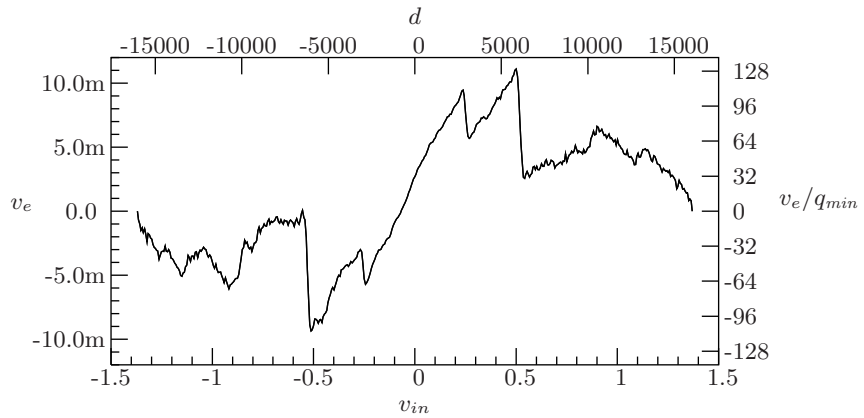

Figure 8.9: The static error v_e of the 10+5-bit FP-ADC.

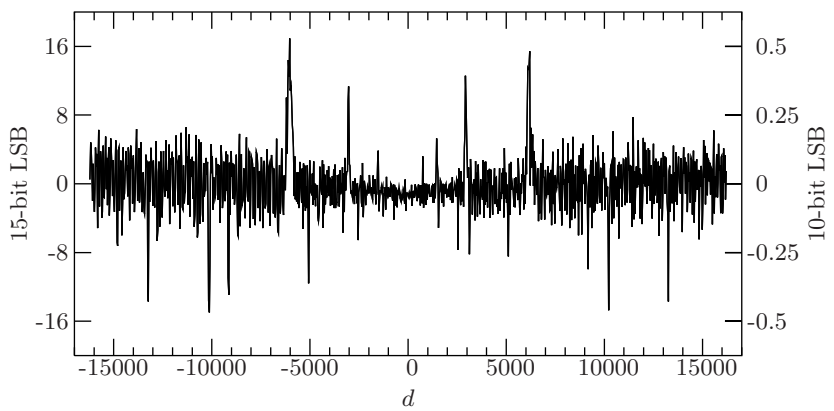

Figure 8.10: The DNL of the 10+5-bit FP-ADC is on the left axis. The equivalent 10-bit DNL is on the right axis.

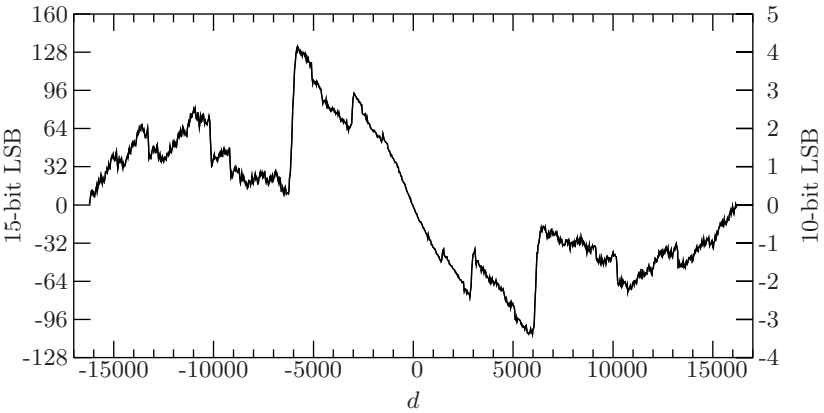

Figure 8.11: The INL of the 10+5-bit FP-ADC is on the left axis. The equivalent 10-bit INL is on the right axis.

resistances in the resistive divider.

The INL of the underlying 10-bit ADC is also visible in the INL curves for the complete FP-ADC. The codes, $|d| > 6100$, in Figure 8.11 indicate that the linearity of the 10-bit ADC is approximately ±1LSB.

Three different sweeps of the harmonic distortion were performed; a sampling frequency sweep, two input frequency sweeps, and an input amplitude sweep. The results can be seen in Figure 8.13, 8.15, 8.14, and 8.12 respectively.

The measurements show that the maximal SNDR is 42.5 dB – corresponding to 6.8 bits. The amplitude sweep in Figure 8.12 shows that the total dynamic range is 71.5 dB – corresponding to 11.6 bits. Thus the dynamic range has been extended by 29 dB – corresponding to 4.8 bits.

According to the plot in Figure 8.13 the sampling rate $f_{s,-3\,\mathrm{dB}} = 53$ MHz and in Figure 8.14 the input signal bandwidth $f_{in,-3\,\mathrm{dB}} = 16.7$ MHz (4.2 MHz for -1 dBFS in Figure 8.15).

8.3.2 Discussion

The design suffers from three different error sources that degrade the performance. The main cause of the degrading of the dynamic performance is excess noise. The excess noise is mainly due to resistor R_3 in the gain stage

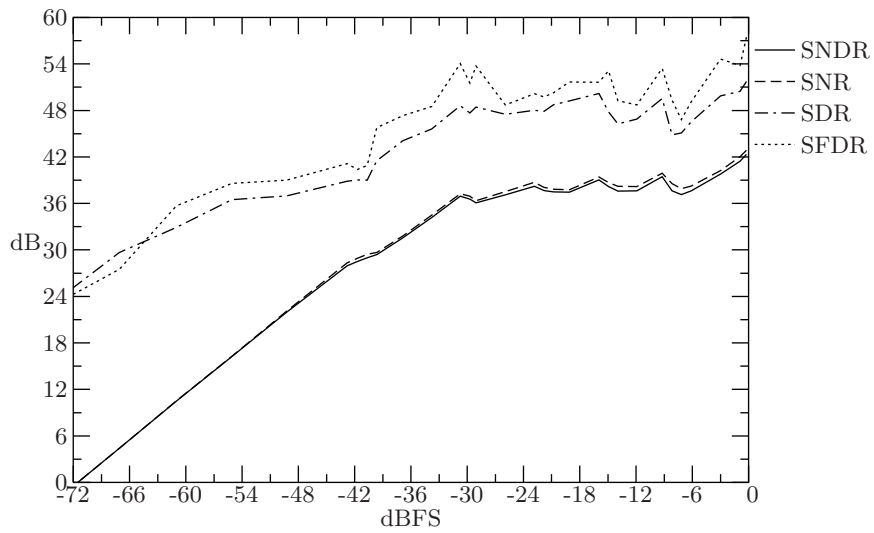

Figure 8.12: The signal to noise and distortion ratio of the 10+5-bit FP-ADC, when sweeping the input amplitude, $f_s = 10$ MHz, $f_{in} = 610$ kHz.

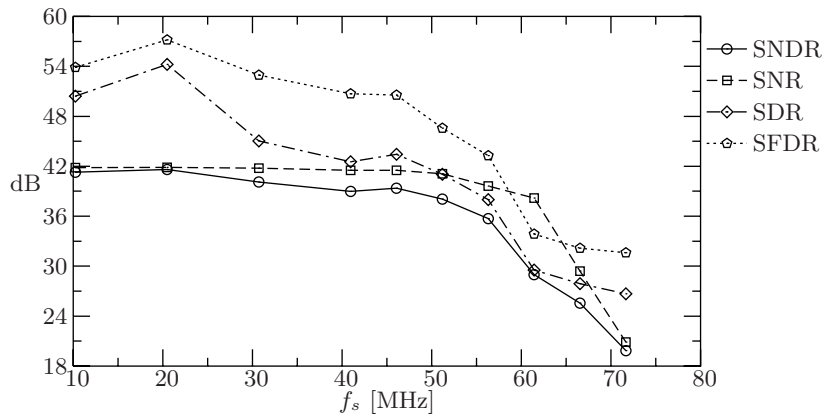

Figure 8.13: The signal to noise and distortion ratio of the 10+5-bit FP-ADC, when sweeping the sampling frequency, $f_{in} \approx 600$ kHz, $A = -1$ dBFS.

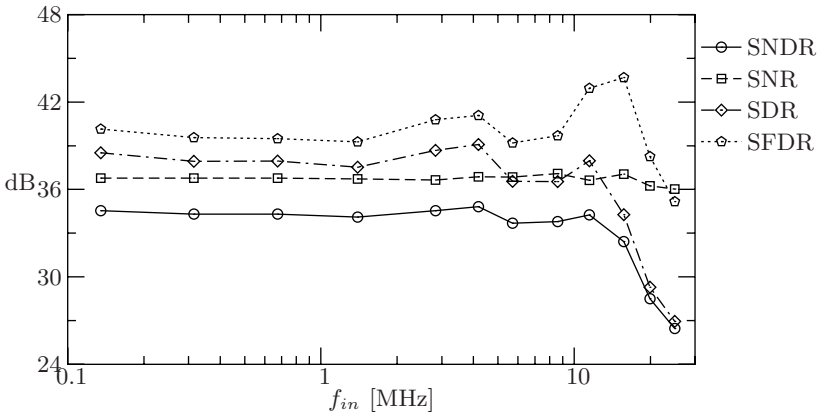

Figure 8.14: The signal to noise and distortion ratio of the 10+5-bit FP-ADC, when sweeping the input frequency, $f_s = 46$ MHz, $A = -20$ dBFS.

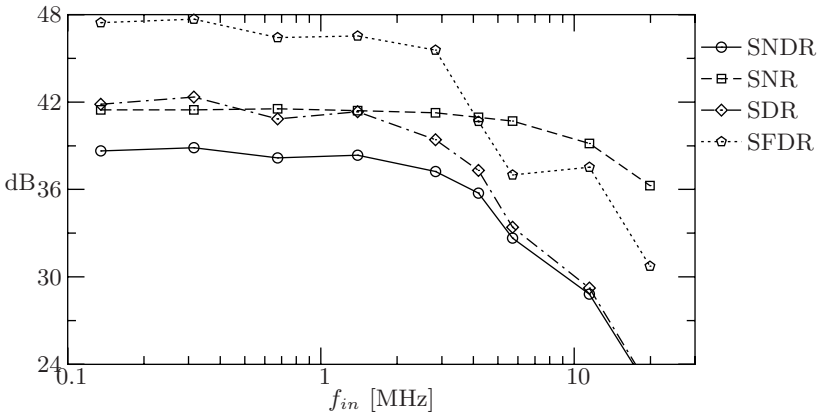

Figure 8.15: The signal to noise and distortion ratio of the 10+5-bit FP-ADC, when sweeping the input frequency, $f_s = 46$ MHz, $A = -0.6$ dBFS.

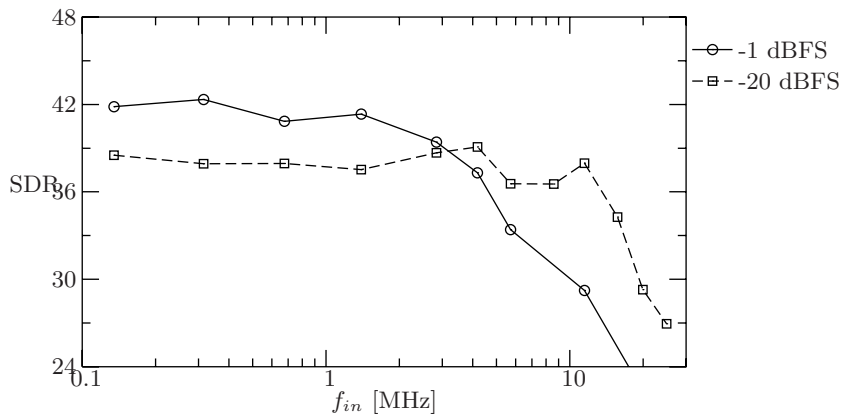

Figure 8.16: The SDR for two different input amplitudes when f_{in} is swept, $f_s = 46$ MHz. The curves are taken from the plots in Figure 8.15 and 8.14.

amplifier (see figure 7.7). The purpose of this resistor was to improve the stability by reducing the loop gain and move the current mirror pole from the band edge to lower frequencies. The benefit on the stability is small so this kind of resistor never should have been used. The resistor R_3 in the gain stage amplifier (see figure 7.7) increases the noise energy by three times more than expected. A post-chip noise simulation confirms this.

Another error is the low analog input frequency, for a specified SDR, at high input amplitudes. The cause is that the controller fails to detect when the gain stages have returned from saturation but have not yet settled. This effect introduces a significant distortion. This is apparent when comparing the distortion for two different amplitudes, as in Figure 8.16. For the same level of SDR the -20 dBFS curve is three times higher in frequency. This is an indication that the bottleneck for high input frequencies with high amplitudes is the control algorithm.

The third error is the non-linearity. As mentioned before, all chips show the same behavior. This indicates that the deterministic effects dominate the non-linearity. A probable cause is temperature gradients in the resistors in the resistive divider. The maximum input power is around 10 mW and 3/4 of the power is dissipated in the two first resistors, R_a and R_b, in the resistor ladder. With a temperature coefficient of 0.07 %/K a 10 degree difference in temperature will give a mismatch of 0.7% which is severe for

the 10+5-bit ADC.

The sampling rate is 54 MHz and this is close to what was expected. Since the gain stage, the SH stage and the first RSD 1-bit pipeline stage are interacting in the amplifying phase the settling time can be approximated to square root of the geometric sum of the settling times of the three stages [91]

$$t_s = \sqrt{t_{s,G}^2 + t_{s,SH}^2 + t_{s,PS}^2} = 9.2 \text{ ns} \tag{8.4}$$

which corresponds to a sampling frequency of

$$f_s = \frac{1}{2(t_s + t_{no})} = 52 \text{ MHz} \tag{8.5}$$

Chapter 9

Conclusions

In my research I set out to make a high-speed distributed floating-point analog-to-digital converter. The idea of this FP-ADC was to move the SH-circuit to the position after the distributed amplifier. This way the requirements on the SH would become much lower. Also the distributed amplifier would operate in the analog domain, hence the speed requirements would be lower. The focus then came to be on the distributed amplifier, to make it accurate in delay, gain and offset in such a way that no calibration would be needed. My research has proved that this idea is technically feasible and a distributed FP-ADC with embedded SH can well extend the dynamic range by 5 bits.

Design experiences were gained. Below are some practical issues and design considerations:

1. The amplifier saturation proved to be a major issue when the chip had been manufactured. Even though the distributed amplifier was reset every sample, saturation at the zero-crossings of the signal deteriorated the signal quality.

2. To remove the phase mismatch, an array of identical amplifiers was used with a preceding R-2R divider. Since all amplifiers were identical they needed to have the same noise requirement, i.e. the noise requirement for the full dynamic range. This was not properly addressed when the chip was designed, and the noise degraded the performance.

3. The gain mismatch was addressed by increasing the area of the resis-

tors. This was effective against the random variation of the mismatch. However, a systematic mismatch remained and dominated the distortion performance. The probable cause is temperature gradients on the chip. Also the large resistor sizes limited the speed and thus increased the signal delay skew in the R-2R ladder.

4. The offset in the CMOS amplifiers has to be removed. This was done effectively by chopping. The drawback of chopping is that the distributed amplifier needs to settle every sample, forcing a higher bandwidth of the amplifier. Also the current consumption was driven up because of the higher bandwidth. Thus the benefit of moving the SH-circuit was partially lost.

9.1 Suggestions for future work

If you want to build a distributed FP-ADC with the SH after the distributed amplifier there are three issues that have to be addressed:

1. Amplifier saturation cannot be avoided and will have a large impact on the signal quality if not properly dealt with. Thus the controller must have a conservative algorithm that alleviates the possibility of selecting a saturated amplifier before it has settled to a correct value. Some solutions to the problem are presented in section 6.4.2.

2. There is a tradeoff between the bandwidth/delay and the gain. The bandwidth and the delay matching set the maximum area of the resistors, while the gain matching sets the minimum area of the resistors. If the maximum and minimum requirements do not overlap, then we recommend using the requirements from the bandwidth and the delay matching to set the resistor area, and make use of digital correction to remove the gain mismatch errors. Digital correction of gain mismatch errors is much easier to implement than digital correction of delay mismatch.

3. Gradients in the voltage divider should be minimized in the layout . The effect of heating and parameter gradients on the chip should not affect the relative resistances in the voltage divider. This can be done by a common-centroid layout [58, 92].

4. The noise requirement of the gain stage gives the current consumption of the gain stage (times $m + 1$ for the gain stage column). Therefore the noise of the gain stage should dominate; The voltage divider and chopper switch resistances should be designed to be non-dominant from a noise perspective. Then the noise of the FP-ADC is predominately set by the noise of the active part plus the resistive feedback of the gain stage.

Bibliography

[1] P. B. Kenington and L. Astier, "Power consumption of A/D converters for software radio applications," *VT*, vol. 49, pp. 643–650, March 2000. 2

[2] www.eniro.se, September 2004. 2

[3] www.shellgeostar.com, September 2004. 2

[4] E. Ingelstam, R. Rönngren, and S. Sjöberg, *TEFYMA, Handbok för teknisk fysik, fysik och matematik*. Sjöbergs Bokförlag AB, 1993. 2

[5] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, pp. 379–432, 623–656, July, Oct. 1948. 5

[6] H. Nyquist, "Certain topics in telegraph transmission theory," *Trans. of the AIEE*, pp. 617–644, February 1928. 7

[7] M. Burns and G. W. Roberts, *An introduction to mixed-signal IC test and measurement*. Oxford university press, 2001. 12, 77, 134

[8] R. van de Plassche, *Integrated analog-to-digital and digital-to-analog converters*. Kluwer, 1994. 12, 14, 88

[9] A. B. Sripad and D. L. Snyder, "A necessary and sufficient condition for quantization errors to be uniform and white," *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. 25, pp. 442–448, Oct. 1977. 14

[10] B. Widrow, I. Kollár, and M.-C. Liu, "Statistical theory of quantization," *IEEE Trans. Instr. and Meas.*, vol. 45, pp. 353–361, April 1996. 14, 16, 17

[11] J. Kontro, K. Kalliojärvi, and Y. Neuvo, "Floating-point arithmetic in signal processing," in *IEEE International Symposium on Circuits and Systems*, vol. 4, pp. 1784–1791, may 1992. 15, 16, 34

[12] R. Gray and T. S. Jr., "Dithered quantizers," *IEEE Trans. Information Theory*, vol. 39, pp. 805–812, May 1993. 17

[13] P. Carbone and D. Petri, "Effect of dither on the resolution of ideal quantizers," *IEEE Trans. Instr. and Meas.*, vol. 49, pp. 337–340, April 1994. 17

[14] R. Wannamaker, S. Lipshitz, and J. Vanderkooy, "A theory of non-subtractive dither," *IEEE Trans. Signal Processing*, vol. 48, pp. 499–516, Feb. 2000. 17

[15] K. Bult and A. Buchwald, "An embedded 240-mW 10-b 50-MS/s CMOS ADC in 1mm^2," *IEEE J. Solid-State Circuits*, vol. 32, pp. 1887–1895, December 1997. 21

[16] H. Pan, *A 3.3-V 12-b 50-MS/s A/D converter in 0.6-μm CMOS with 80-dB SFDR*. PhD thesis, University of California at Los Angeles, USA, 1999. 21

[17] M. Choi and A. A. Abidi, "A 6-b 1.3-Gsample/s A/D converter in 0.35-μm cmos," *IEEE J. Solid-State Circuits*, vol. 36, pp. 1847–1858, December 2001. 21

[18] T. B. Cho and P. R. Gray, "A 10 b, 20 Msample/s, 35mW pipeline A/D converter," *IEEE J. Solid-State Circuits*, vol. 30, pp. 166–177, March 1995. 26, 81

[19] H. Haneko, "A unified formulation of segment companding laws and synthesis of codecs and digital compandors," *Bell System Technical J.*, vol. 49, pp. 1555–1588, September 1970. 28

[20] S. Cantarano and G. Pallottino, "Logarithmic analog-to-digital converters: A survey," *IEEE Trans. Instr. and Meas.*, vol. 22, pp. 201–213, September 1973. 29

[21] H. O. Kuntz, "Expinential D/A converter with a dynamic range of eight decades," *IEEE Trans. Circuits and Systems*, vol. 25, pp. 522–526, July 1978. 29, 31

[22] D. Seitzer, G. Pretzl, and N. Hamdy, *Electronic analog-to-digital converters*. John Wiley & Sons, 1983. 29

[23] J. D. Everard, "A single-channel PCM codec," *IEEE J. Solid-State Circuits*, vol. 14, pp. 25–37, February 1979. 29

[24] K. N. Leung and P. K. T. Mok, "Analysis of multistage amplifier-frequency compensation," *IEEE Trans. Circuits Syst. I*, vol. 48, pp. 1041–1056, Sept. 2001. 29

[25] A. Kayanuma *et al.*, "An integrated 16b A/D converter for PCM audio systems," in *ISSCC*, pp. 56–57, February 1981. 29

[26] T. Sugawara *et al.*, "A trimless 14b/20μs dual-channel ADC for PCM audio," in *Proc. IEEE ISSCC*, pp. 184–185, February 1983. 29

[27] K. B. Ohri and M. J. Callahan, "Integrated PCM codec," *IEEE J. Solid-State Circuits*, vol. 14, pp. 38–46, February 1979. 31

[28] J. B. Cecil, E. M. W. Chow, J. A. Flink, and J. E. Solomon, "Per channel codecs for PCM telecommunications," in *Proc. IEEE ISSCC*, pp. 176–177, February 1979. 31

[29] E. Pfrenger, P. Picard, and F. van Sichart, "A companding D/A converter for a dual-channel PCM codec," in *Proc. IEEE ISSCC*, pp. 186–187, February 1997. 31

[30] S. Kelley and D. Ulmer, "A single-chip PCM codec," *IEEE J. Solid-State Circuits*, vol. 14, pp. 54–59, February 1979. 31

[31] R. A. Blauschild *et al.*, "A single-chip I^2L PCM codec," *IEEE J. Solid-State Circuits*, vol. 14, pp. 59–64, February 1979. 31

[32] G. Smarandoiu, D. A. H. P. R. Gray, and G. Landsburg, "CMOS pulse-code-modulation voice codec," *IEEE J. Solid-State Circuits*, vol. 13, pp. 504–509, August 1978. 31

[33] J. T. Caves, C. Chan, S. Rosenbaum, L. P. Sellars, and J. Terry, "A PCM voice codec with on-chip filters," *IEEE J. Solid-State Circuits*, vol. 14, pp. 65–73, February 1979. 31

[34] J. C. Candy and G. C. Temes, eds., *Oversampling delta-sigma data converters*. IEEE Press, 1991. 31

[35] J. C. Candy, W. H. Ninke, and B. A. Wooley, "A per-channel A/D converter having 15-segment μ-255 companding," *IEEE J. Solid-State Circuits*, vol. 24, pp. 33–42, January 1976. 32

[36] B. A. Wooley, "An integrated per-channel PCM encoder based on interpolation," *IEEE J. Solid-State Circuits*, vol. 14, pp. 14–20, February 1979. 32

[37] B. A. Wooley, D. C. Fowlis, J. L. Henry, and C. Williams,Jr, "An integrated interpolative PCM decoder," *IEEE J. Solid-State Circuits*, vol. 14, pp. 20–25, February 1979. 32

[38] J. Guilherme and J. Franca, "New CMOS logarithmic A/D converters employing pipeline and algorithmic architectures," in *IEEE International Symposium on Circuits and Systems*, vol. 1, pp. 529–532, April 1995. 32

[39] J. Guilherme, J. Vital, and J. Franca, "A true logarithmic analog-to-digital pipeline converter with 1.5 bit/stage and digital correction," in *IEEE International Conference on Electronics, Circuits and Systems*, vol. 1, pp. 393–396, September 2001. 32

[40] J. Guilherme, J. Vital, and J. Franca, "A CMOS logarithmic pipeline A/D converter with a dynamic range of 80 dB," in *IEEE International Conference on Electronics, Circuits and Systems*, vol. 1, pp. 193–196, September 2002. 32

[41] K. Kalliojärvi, J. Kontro, and Y. Neuvo, "Novel floating-point A/D- and D/A-conversion methods," in *IEEE International Symposium on Circuits and Systems*, vol. 2, pp. 1–4, May 1994. 35

[42] F. Chen and C. S. Chen, "A 20-b dynamic-range floating-point data acquisition system," *Trans. Industrial Electronics*, vol. 38, pp. 10–14, February 1991. 36

[43] L. Grisoni, A. Heubi, P. Balsinger, and F. Pellandini, "Implementation of a micro-power 15-bit 'floating-point' A/D converter," in *Int. Symp. Low Power Electronic and Design*, pp. 247–252, August 1996. 36, 38

[44] V. Groza, "Floating-point ADC optimized for acqusition of deterministic signals," in *IEE Instr. and Meas. Tech. Conference*, vol. 1, pp. 707–711, May 2002. 36

[45] S. Sharma, G. Otomo, K. Tsukamoto, and T. Miyata, "A floating-point A/D converter uses low resolution DAC to get wide dynamic range," *Int. J. of Electronics*, vol. 64, pp. 787–794, May 1988. 36, 38

[46] V. Z. Groza, "High-resolution floating-point ADC," *IEEE Trans. Instr. and Meas.*, vol. 50, pp. 1822–1829, December 2001. 36, 39

[47] J. Heath and T. Nagle, "Design a floating-point A/D converter," *Electronic design*, vol. 22, pp. 80–84, may 1974. 36

[48] G. Ootomo, K. Tsukamoto, T. Watahiki, and T. Miyata, "A floating-point A/D converter with self-calibration," in *Trans. of the inst. of Electronics, Information and Communication Eng.*, vol. E71, pp. 1303–1308, 1988. 36, 38

[49] V. Groza, "Floating-point analog-to-digital converters with predictive auto-ranging," in *IEE Instr. and Meas. Tech. Conference*, vol. 2, pp. 759–762, May 2000. 37

[50] K. E. Prada, K. von der Heydt, and T. F. O'Brien, "A versatile multichannel data acquisition system for seismic and acoustic applications," *OCEANS*, vol. 13, pp. 43–47, September 1981. 37, 38, 103

[51] G. M. Haller and D. R. Freytag, "Analog floating-point BiCMOS sampling chip and architecture of the BaBar CsI caliometer front-end electronics system at the SLAC B-factory," *IEEE Trans. Nuclear Science*, vol. 43, pp. 1610–1614, June 1996. 37

[52] J. Yuan and J. Piper, "Floating-point analog-to-digital converter," in *IEEE International Conference on Electronics, Circuits and Systems*, vol. 3, pp. 1385–1388, 1999. 37, 39

[53] J. Piper and J. Yuan, "Realization of a floating-point A/D converter," in *IEEE International Symposium on Circuits and Systems*, vol. 1, pp. 404–407, 2001. 37

[54] T. Zimmerman and J. R. Hoff, "The design of a charge-integrating modified floating-point ADC chip," *IEEE J. Solid-State Circuits*, vol. 39, pp. 895–905, May 2004. 37, 39

[55] D. U. Thompson and B. A. Wooley, "A 15-b pipelined CMOS floating-point A/D converter," *IEEE J. Solid-State Circuits*, vol. 36, pp. 299–303, Feb. 2001. 38, 39

[56] D. H. K. Hoe and D. B. Ribner, "An auto-ranging photodiode preamplifier with 114 db dynamic range," *IEEE J. Solid-State Circuits*, vol. 31, pp. 187–193, February 1996. 38

[57] J. Piper and J. Yuan, "A delay-balanced binary-weighted CMOS amplifier tree for a floating point A/D converter," in *Proc. IEEE NORCHIP Conference*, pp. 131–138, Nov. 1998. 39

[58] J. L. McReary and P. R. Gray, "All-MOS charge redistribution analog-to-digital conversion techniques—part I," *IEEE J. Solid-State Circuits*, vol. 10, pp. 371–379, December 1975. 43, 150

[59] R. E. Suárez, P. R. Gray, and D. A. Hodges, "All-MOS charge redistribution analog-to-digital conversion techniques—part II," *SC*, vol. 10, pp. 379–385, December 1975. 43

[60] P. E. Allen and A. Sánchez-Sinencio, *Switched Capacitor Circuits*. Van Nostrand Reinhold, 1984. 43, 50, 110

[61] T. Cho, *Low-Power Low-voltage Analog-to-digital conversion techniques using pipelined architecture*. PhD thesis, University of California at Berkeley, USA, 1995. 43

[62] R. Gregorian and G. Temes, *Analog MOS Integrated Circuits for signal processing*. John Wiley & Sons, 1986. 45, 56, 172

[63] E. A. Vittoz, "The design of high performance analog circuits on digital CMOS chips," *IEEE J. Solid-State Circuits*, vol. 20, pp. 657–665, June 1985. 49

[64] W. Wilson, H. Massoud, E. Swanson, R. George, and R. Fair, "Measurement and modeling of charge feedthrough in n-channel MOS analog switches," *IEEE J. Solid-State Circuits*, vol. 20, pp. 1206–1213, Dec. 1985. 49

[65] J. Piper, R. Strandberg, and F. Tillman, "Advanced analog design." course lecture notes, Dept. of Electroscience, Lund University, Box 118, SE-211 00 Sweden, 2004. 54, 56, 116

[66] J. Steensgaard, "Bootstrapped low-voltage analog switches," in *IEEE International Symposium on Circuits and Systems*, vol. 2, pp. 29–32, 1999. 56

[67] P. R. Gray and R. G. Meyer, "MOS operational amplifier design—a tutorial overview," *IEEE J. Solid-State Circuits*, vol. SC-17, pp. 969–982, Dec. 1982. 56, 116, 117

[68] P. R. Gray and R. G. Meyer, *Analysis and Design of Analog Integrated Circuits*. New York: Wiley, 3rd ed., 1993. 56

[69] E. H. Nordholt, *Design of High-Performance Negative-Feedback Amplifiers*. Delft University Press, Delft, the Netherlands, 1993. 56, 57, 58

[70] C. J. M. Verhoeven, A. van Staveren, G. L. E. Monna, M. H. L. Kouwenhoven, and E. Yildiz, *Structured Electronic Design: Negative-feedback amplifiers*. Delft University of Technology, Delft, the Netherlands, 2002 ed., July 2002. 56, 98, 176

[71] M. Lantz, *Systematic Design of Linear Feedback Amplifiers*. PhD thesis, Lund Institute of Technology, department of Electroscience, Sweden, Nov. 2002. 56, 71

[72] J. Piper, R. Strandberg, and F. Tillman, "Advanced analog design." course exercises, Dept. of Electroscience, Lund University, Box 118, SE-211 00 Sweden, 2004. 56

[73] H. S. Black, "Stabilized feedback amplifiers," *Bell Syst. Tech. J.*, vol. 13, pp. 1–18, Jan. 1934. 56

[74] J. Piper and J. Yuan, "Distortion in pipelined analog-to-digital converters," in *Proc. IEEJ AVLSI workshop*, October 2004. 71, 95

[75] P. Wambacq and W. Sansen, *Distortion Analysis of Analog Integrated Circuits*. Dordrecht: Kluwer, 1998. 71

[76] M. L. Lantz and S. Mattisson, "Local feedback and nonlinearity of multistage feedback amplifiers," in *Proc. NDES'02*, (Çeşme, Izmir, Turkey), pp. 2:29–32, June 2002. 71

[77] K. Kim, "Analog-to-digital conversion and harmonic noises due to the integral nonlinearity," *IEEE Trans. Instr. and Meas.*, vol. 43, pp. 151–156, April 1994. 72

[78] J. Bruce and P. Stubberud, "An anlyses of analog to digital conversion and harmonic distortion," in *IEEE Midwest symp. on Curcuits and Systems*, vol. 2, pp. 656–659, August 2000. 72

[79] E. Liu, G. Gielen, H. Chang, A. Sangiovanni-Vincentelli, and P. R. Gray, "Behavioral modeling and simulation of data converters," in *IEEE International Symposium on Circuits and Systems*, vol. 5, pp. 2144–2147, May 1992. 72, 77

[80] D. Goren, E. Shamsaev, and I. A. Wagner, "A novel method for stochastic nonlinearity analysis of a CMOS pipeline ADC," in *Design automation conf.*, pp. 127–132, June 2001. 72

[81] S. H. Lewis and P. R. Gray, "A pipleined 5-Msamples/s 9-bit analog-to-digital converter," *IEEE J. Solid-State Circuits*, vol. 22, pp. 954–961, Dec. 1987. 72

[82] O. E. Erdoğan, P. J. Hurst, and S. H. Lewis, "A 12-b digital-background-calibrated algorithmic ADC with –90-db THD," *IEEE J. Solid-State Circuits*, vol. 34, pp. 1812–1820, December 1999. 73

[83] E. G. Soenen and R. L. Geiger, "An architecture and an algorithm for fully digital correction of monolithic pipelined ADC's," *IEEE Trans. Circuits and Systems-II*, vol. 42, pp. 143–153, March 1995. 73

[84] H. Pan, M. Segami, J. Cao, and A. A. Abidi, "A 3.3-V 12-b 50-MS/s A/D converter in 0.6-μm CMOS with over 80-db SFDR," *IEEE J. Solid-State Circuits*, vol. 35, pp. 1769–1780, Dec. 2000. 76

[85] J.-B. Shyu, G. C. Temes, and K. Yao, "Random errors in MOS capacitors," *IEEE J. Solid-State Circuits*, vol. 17, pp. 1070–1076, Dec. 1982. 85

[86] J.-B. Shyu, G. C. Temes, and F. Krummenacher, "Random error effects in matched MOS capacitors and current sources," *IEEE J. Solid-State Circuits*, vol. 19, pp. 948–955, Dec. 1984. 85

[87] W. C. Black, JR and D. A. Hodges, "Time interleaved converter arrays," *IEEE J. Solid-State Circuits*, vol. SC-15, pp. 1022–1029, Dec. 1980. 92, 93

[88] J. Yuan and C. Svensson, "A 10-bit 5-MS/s successive approximation ADC cell used in a 70 MS/s ADC array in 1.5μm CMOS," *IEEE J. Solid-State Circuits*, vol. 29, pp. 866–872, August 1994. 92

[89] G. Xu, *Charge samping circuits and A/D converters*. PhD thesis, Lund University, Sweden, 2004. 124

[90] X. Aragonès, J. L. Ganzález, and A. Rubio, *Analysis and solutions for switching noise coupling in mixed-signal ICs*. Kluwer, 1999. 129

[91] T. H. Lee, *The design of a CMOS radio-frequency integrated circuits*. Cambridge university press, 1998. 147, 171

[92] G. Gielen, "Systematic design of data converters," in *IEEE International Symposium on Circuits and Systems*, vol. Tutorial guide, pp. 6.3.1–6.3.14, May 2001. 150

[93] D. P. Foty, *MOSFET Modeling with SPICE: Principles and Practice*. New Jersey: Prentice Hall, 1997. 169

Appendix A

List of Symbols

List of acronyms

ADC	analog-to-digital converter
AGC	automatic gain control
ASCS	anti-series CS (differential stage)
AZ	auto-zero
CDF	cumulative distribution function
CD	common-drain (voltage follower)
CDS	correlated double-sampling
CG	common-gate (current follower)
CMOS	complementary metal-oxide semiconductor
CMRR	common-mode rejection ration
codec	code and decode
compand	compress and expand
CS	common-source
DAC	digital-to-analog converter
DC	direct current (zero frequency)
DNL	differential non-linearity
DR	dynamic range
ENOB	effective number of bits
FFT	fast Fourier transform
FP-ADC	floating-point ADC
INL	integral non-linearity

L-ADC	logarithmic ADC
LSB	least significant bit
MSB	most significant bit
NMOS	N-type MOS metal-oxide semiconductor
PCM	pulse-code modulation
PDF	probability density function
PMOS	P-type metal-oxide semiconductor
PSRR	power-source rejection-ratio
RMS	root-mean-square
R-2R	ladder resistor network
SA	successive approximation
SH	sample-and-hold
SDR	signal-to-distortion ratio, SDR=1/THD
SFDR	spurious-free dynamic range
SNDR	signal-to-noise-plus-distortion ratio
SNR	signal-to-noise ratio
THD	Total harmonic distortion
$\Delta\Sigma$	delta-sigma

List of parameters

a_0, a_1	constant and amplitude of a sinusoid, $a_1 \sin(\cdot) + a_0$
A	gain factor
$A\beta$	loop-gain
A_C	capacitor matching parameter
A_R	resistor matching parameter
b	bit(s)
b_k	Fourier coefficient
B	box bandwidth
c, μ, A	compression coefficients for f_c, f_μ and f_A
C	capacitance
C_s	sample capacitor
d	digital signal
e, f, g	relative radix, reference and offset error
f	frequency in Hertz
f_{in}	input frequency
f_s	sample frequency, $f_s = 1/T_s$

F_n	quantization versus sample noise factor
F_{sw}	residual charge error factor
G	gain of amplifier
G_G	gain of gain stage
G_{SH}	gain of SH stage
G_∞	asymptotic gain
i	current
j	imaginary, $j = \sqrt{-1}$
k	Boltzmann's constant, $k = 1.3807 \cdot 10^{-23}$
k, l, m	integers
K	feedback factor in Black's model
m	the extra number of bits provided by an FP-ADC
n	number of bits
n_e	n in the exponent
M	CMOS transistor
N_{comp}	number of comparators
p	pole
p	number of pipeline stages
q	charge
r	reference
r_D	equivalent divider noise resistance
r_{eq}, r_9	equivalent input noise resistance
r_{FB}	equivalent feedback noise resistance
r_G	equivalent gain amplifier noise resistance
r_{SW}	equivalent switch noise resistance
R	resistance
R_a	longitudinal resistor in an R-2R
R_b	transversal resistor in an R-2R
R_c	delay equalizing resistor in R-2R
R_s	sample resistor
R_1	series feedback resistor
R_2	shunt feedback resistor
s	signal
s_s, s_i, s_c, s_ℓ	signal interfaces in the superposition model
S	switch
S	spectrum
t	time
t_d	delay time

$t_j, \bar{t}_j^2$	time jitter and variance of t_j
t_{no}	non-overlap time
t_s	settling time
t_f	clock phase fall time
T	absolute temperature
T_s	sample period, $T_s = 1/f_s$
v	voltage
$\bar{v}$	RMS voltage,$\bar{v} = \sqrt{\bar{v}^2}$
$\bar{v}^2$	power of voltage, $\bar{v}^2 = \frac{1}{T}\int_0^T v^2(t)dt$
V_{DD}	positive supply voltage
v_e	input-referred non-linear error, $v_e = d - v_{in}$
V_{FS}	full-scale voltage
v_{in}	analog input voltage
v_n	noise voltage
v_{off}	offset voltage
v_q	quantized voltage
V_R	reference voltage
v_s	sample voltage, $v_s = v_{in}(kT_s)$
v_{TH}	track-and-hold voltage
w	binary word, $w = \sum_{k=0}^{n-1} b_k 2^k$
w_E, w_M	w of the exponent and mantissa
β	feedback factor in the superposition model
γ_{euler}	mathematical constant, $\gamma_{\text{euler}} \approx 0.577$
$\Delta G/G$	gain mismatch
$\Delta C/C$	capacitor mismatch
ε	residual error
ϵ	quantization error
$\theta(n)$	the ratio between the t_s and τ_s
λ_α	the α quantile
ξ, ν, ρ	transfers in the superposition model
τ	time constant
ϕ	clock phase
$\phi_s, \bar{\phi}$	sample ϕ and inverse ϕ
φ	angle
ω	angular frequency
ω_1, ω_2	-3 dB bandwidths

List of transistor parameters

C_{ov}	gate overlap capacitance, per length
C_{ox}	oxide capacitance, per area
C_{db}	drain-bulk capacitance
C_{gs}	gate-source capacitance
C_{gd}	gate-drain capacitance
C_{gb}	gate-bulk capacitance
C_{poly}	poly-silicon capacitance, per area
C_{sb}	source-bulk capacitance
g_{ds}	output conductance
g_m	gate transconductance
g_{mb}	bulk conductance
V_T	transistor threshold voltage
V_{T0}	nominal threshold voltage
W, L	physical width and length
γ	body effect factor
γ_{noise}	noise fitting parameter
λ	channel length modulation
μ	channel charge mobility
ϕ_F	Fermi potential

List of functions

$\mathrm{ci}(\cdot)$	$\mathrm{ci}(x) = \int_x^\infty \frac{\cos t}{t} dt$
$E(\cdot)$	mean (expected) function
$\cdot \star \cdot$	convolution
$f(\cdot), f^{-1}(\cdot)$	codec
$f_c(\cdot)$	c-law
$f_\mu(\cdot)$	μ-law
$f_A(\cdot)$	A-law
$H(\cdot)$	frequency response
$\ln(\cdot)$	natural logarithm
$\log_2(\cdot)$	radix-2 logarithm
$v_e(\cdot)$	general error function
$\mathcal{F}(\cdot)$	Fourier transform
$\Re(\cdot)$	real part transform

$\phi(\cdot)$	CDF
$\varphi(\cdot)$	PDF
$\sigma(\cdot)$	standard deviation
$\lfloor \cdot \rfloor$	floor function
$\lvert \cdot \rvert$	absolute function
$\cdot \vee \cdot$	logical OR
$\cdot \wedge \cdot$	logical AND
$\bar{\cdot}$	logical NOT

Appendix B

The CMOS Transistor Model

The MOSFET long channel model used for analyzing the CMOS circuits in this thesis is a simple model, the SPICE level 1 model [93]. Even though it is not very accurate it is a good model for humans to understand the basic behavior of a circuit using CMOS transistors. More details are left to computers and real life circuits.

The small-signal model of a CMOS transistor is shown in Figure B.1. The meaning of the capacitor indexes are; g–gate, d–drain, s–source, and b–bulk. g_m is the gate transconductance and g_{mb} the bulk (back-gate) transconductance. g_{ds} is the output conductance.

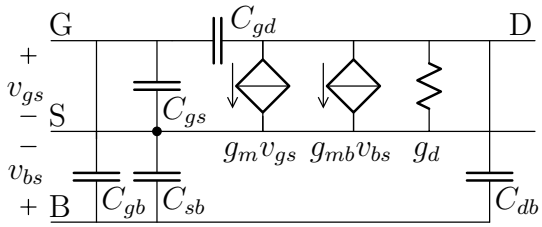

Figure B.1: High frequency small-signal model of the CMOS transistor.

B.1 The cutoff region

In the cutoff region v_{GS} is so small that the inversion layer in the channel does not form

$$v_{GS} < V_T \tag{B.1}$$

In the cutoff region there is no current flowing through the transistor. However the capacitances are still present. This is the case when the CMOS transistor is used as an open switch.

$$C_{gs} = C_{gd} = WC_{ov} \tag{B.2}$$

$$C_{gb} = WLC_{ox} \tag{B.3}$$

where C_{ov} is the overlapping capacitance, W and L are the width and length of the transistor and C_{ox} the oxide capacitance per unit area.

B.2 The triode region

In the triode region the inversion layer is well developed between source and drain

$$v_{DS} < v_{GS} - V_T \tag{B.4}$$

The drain current in the triode region is

$$i_D = \mu C_{ox} \frac{W}{L} \left[(v_{GS} - V_T) v_{DS} + \frac{v_{DS}^2}{2} \right] \tag{B.5}$$

where μ is the mobility of electrons or holes in the channel and $\frac{W}{L}$ is the aspect ratio of the transistor. The threshold voltage V_T of the transistor is

$$V_T = V_{T0} + \gamma(\sqrt{2\phi_F + v_{SB}} - \sqrt{2\phi_F}) \tag{B.6}$$

whereV_{T0} is the nominal threshold voltage, γ is the body effect factor and ϕ_F is the Fermi potential.

In the triode region the output conductance is

$$g_d = \frac{\partial i_D}{\partial v_{DS}} = \mu C_{ox} \frac{W}{L} \left[v_{GS} - V_T + v_{DS} \right] \tag{B.7}$$

B.3 The saturation region

In the saturation region the inversion layer is formed but is cut off just before the drain area

$$v_{DS} \geq v_{GS} - V_T$$

The small signal model in saturation region for long channel devices is

$$i_D = \frac{\mu C_{ox}}{2}\frac{W}{L}(v_{GS} - V_T)^2(1 + \frac{\lambda' v_{DS}}{L}) \tag{B.8}$$

where $\lambda'/L = \lambda$ is the channel length modulation. Now we can derive the transconductance and output conductance

$$g_m = \frac{\partial i_D}{\partial v_{GS}} = \frac{2I_D}{V_{GS} - V_T} = \sqrt{2\mu C_{ox}\frac{W}{L}I_D} = \mu C_{ox}\frac{W}{L}(V_{GS} - V_T)(1 + \frac{\lambda' V_{DS}}{L}) \tag{B.9}$$

The transconductance has three dominant parameters; the bias current I_D, the gate over-drive $V_{GS} - V_T$, and the aspect ratio W/L. One of the parameters can be eliminated meaning that the transconductance has two degrees of freedom.

The output conductance is

$$g_d = \frac{\partial i_D}{\partial v_{DS}} = \frac{I_D}{L/\lambda + V_{DS}} = \frac{\lambda\prime\mu C_{ox}}{2}\frac{W}{L^2}(V_{GS} - V_T)^2 \tag{B.10}$$

If we are interested in the intrinsic voltage gain, it is

$$\mu = \frac{g_m}{g_d} = \frac{2(L/\lambda' + V_{DS})}{V_{GS} - V_T} = \sqrt{\frac{2\mu C_{ox}WL}{I_D}}\frac{(1 + \frac{\lambda' v_{DS}}{L})}{\lambda} \tag{B.11}$$

Interesting is also the transit frequency. The gate-source capacitance in the saturation region is

$$C_{gs} = \frac{2}{3}C_{ox}WL \tag{B.12}$$

This gives the transit frequency approximation [91]

$$\omega_T \approx \frac{g_m}{C_{gs}} = \frac{3\mu(V_{GS} - V_T)}{2L^2} = \sqrt{\frac{\mu I_D}{2C_{ox}W}} \cdot \frac{3}{L^{3/2}} \tag{B.13}$$

	Cutoff region	Triode region	Saturation region
C_{gs}	WC_{ov}	$WC_{ov} + \frac{1}{2}WLC_{ox}$	$WC_{ov} + \frac{2}{3}WLC_{ox}$
C_{gd}	WC_{ov}	$WC_{ov} + \frac{1}{2}WLC_{ox}$	WC_{ov}
C_{gb}	WLC_{ox}	0	$\frac{\frac{1}{3}WLC_{ox} \cdot C_j(v_{DB})}{\frac{1}{3}WLC_{ox} + C_j(v_{DB})}$
C_{bs}	$A_S \cdot C_j(v_{DB})$	$(A_S + \frac{1}{2}WL) \cdot C_j(v_{SB})$	$(A_S + \frac{2}{3}WL) \cdot C_j(v_{SB})$
C_{bd}	$A_D \cdot C_j(v_{DB})$	$(A_D + \frac{1}{2}WL) \cdot C_j(v_{DB})$	$A_D \cdot C_j(v_{DB})$

Table B.1: Capacitors parameters for the different operating regions.

B.4 The capacitances

The capacitances of the MOSFET are

- the gate oxide capacitance WLC_{ox}. C_{ox} is the oxide capacitance per unit area.
- the overlapping capacitances between the gate and source/drain WC_{ov}. C_{ov} is the overlap capacitance per unit length.
- the PN-junction capacitances between source/drain and the bulk AC_j. C_j is the PN-junction capacitance per unit area.
- the PN-junction capacitance between the channel and the bulk (WLC_j)

Depending on the operation region these capacitances are distributed differently among the MOSFET terminals. The PN-junction has an area capacitance which is voltage dependent

$$C_j(v_R) = \frac{C_{j0}}{\sqrt{1 + v_R/2\phi_F}} \tag{B.14}$$

The values of the capacitances [62] are shown in table B.1.

B.5 The noise model

The spectrum noise model for the CMOS transistor consists of two independent current sources; the thermal noise in the channel S_{Id} and the flicker noise in the channel $S_{Id,f}$. The thermal noise in the saturation region is resistive

$$S_{Id} = 4kT\gamma_{noise}g_m \tag{B.15}$$

$$S_{Id,f} = 4kT\gamma_{noise}g_m/f_f \tag{B.16}$$

where k is Boltzmann's constant, T the absolute temperature, g_m the transconductance of the transistor, γ_{noise} a fitting parameter ($\frac{2}{3}$ for long and 2 for short channel transistors, due to the incremental channel resistance) and f_f the frequency where $S_{Id}(f) = S_{Id,f}(f)$.

Appendix C

Non-Ideal Effects in Dividers

C.1 Resistive feedback

The non-ideal effects analyzed in this section are for a divider used as a feedback network for a voltage amplifier.

C.1.1 The mismatch in resistive feedback

The standard deviation of the relative gain error for an amplifier using resistive feedback is calculated below.

The gain of the amplifier is

$$A_t = 1 + \frac{R_2}{R_1} \tag{C.1}$$

and the relative gain error

$$\Delta A = \frac{A_t}{A} - 1 \tag{C.2}$$

Differentiating the relative gain error

$$\frac{\delta \Delta A}{\delta R_2} = \frac{1}{AR_1}, \quad \frac{\delta \Delta A}{\delta R_1} = -\frac{R_2}{AR_1^2} \tag{C.3}$$

gives the standard deviation

$$\sigma_{\Delta A}^2 = (\frac{1}{AR_1})^2 \sigma_{R_2}^2 + (\frac{R_2}{AR_1^2})^2 \sigma_{R_1}^2 \tag{C.4}$$

We know that

$$\sigma_R = \frac{RA_R}{\sqrt{2WL}} \tag{C.5}$$

why the standard deviation of the relative gain error is

$$\sigma_{\Delta A}^2 = (\frac{R_2 A_R}{\sqrt{2}AR_1})^2 (\frac{1}{(WL)_1} + \frac{1}{(WL)_2}) \tag{C.6}$$

using $A_t = 1 + \frac{R_2}{R_1}$ gives

$$\sigma_{\Delta A} = (1 - \frac{1}{A_t}) \frac{A_R}{\sqrt{2}} \sqrt{\frac{1}{(WL)_1} + \frac{1}{(WL)_2}} \tag{C.7}$$

C.1.2 Mismatch in a capacitive feedback

The calculation of the mismatch in a capacitive divider is analogue to the mismatch in a resistive divider. A simple exchange of variables ($R_1 \leftarrow C_2$, $R_2 \leftarrow C_2$, $A_R \leftarrow A_C$, $\sigma_R \leftarrow \sigma_C$, $A_R \leftarrow A_C$) gives

$$\sigma_{\Delta A} = (1 - \frac{1}{A_t}) \frac{A_C}{\sqrt{2}} \sqrt{\frac{1}{(WL)_1} + \frac{1}{(WL)_2}} \tag{C.8}$$

C.1.3 Noise in a resistive feedback

If the noise from the feedback resistors is referred to the input the equivalent noise spectrum is [70]

$$S_{FB} = 4kT \cdot R_1 \| R_2 \tag{C.9}$$

C.2 R-2R ladder

The analysis in this section focuses on the non-ideal effects in an R-2R ladder divider.

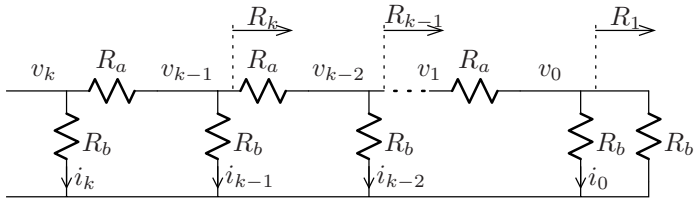

Figure C.1: Ladder divider

C.2.1 The dividing factor in an R-2R ladder

Here we look at the voltage factors between the voltage taps in an R-2R ladder when the voltage in the last tap is normalized to 1. The ladder can be seen in Figure C.1.

Nodal analyses in the node $k-1$ gives

$$v_k - v_{k-1}\left(2 + \frac{R_a}{R_b}\right) + v_{k-2} = 0 \tag{C.10}$$

To solve this equation we look at the characteristic polynomial $p(r)$ of equation C.10. The roots of the characteristic polynomial is

$$r = 1 + \frac{R_a}{2R_b} \pm \sqrt{\frac{R_a}{R_b} + \left(\frac{R_a}{R_b}\right)^2} \tag{C.11}$$

The general solution to equation C.10 is

$$v_k = \alpha r_1^k + \beta r_2^k \tag{C.12}$$

with the initial values

$$\begin{cases} v_0 = 1 \\ v_1 = 1 + \frac{2R_a}{R_b} \end{cases} \tag{C.13}$$

then α and β can be solved

$$\alpha = \frac{1 + \frac{2R_a}{R_b} - r_2}{r_1 - r_2} \tag{C.14}$$

$$\beta = \frac{1 + \frac{2R_a}{R_b} - r_1}{r_2 - r_1} \tag{C.15}$$

C.2.2 The mismatch in an R-2R ladder

The standard deviation of the relative gain error in an R-2R ladder is calculated below.

The gain between two taps, k and $k-1$, is

$$A_k = \frac{v_k}{v_{k-1}} = 1 + \frac{R_a}{R_k} + \frac{R_a}{R_b} \tag{C.16}$$

where R_k is the equivalent resistance defined in Figure C.1. The value of this resistor is recursive

$$R_k = R_a + \frac{R_{k-1}R_b}{R_{k-1}+R_b} \tag{C.17}$$

Let us start by calculating the standard deviation of the equivalent resistor R_k by differentiation

$$\tfrac{\partial R_k}{\partial R_a} = 1, \quad \tfrac{\partial R_k}{\partial R_b} = \tfrac{R_b^2}{(R_{k-1}+R_b)^2} = \tfrac{1}{4}, \quad \tfrac{\partial R_k}{\partial R_{k-1}} = \tfrac{R_{k-1}^2}{(R_{k-1}+R_b)^2} = \tfrac{1}{4} \tag{C.18}$$

Here we have used a priori knowledge ($R_a = R$, $R_b = R_k = 2R$) to calculate the differentials. Now the standard deviation becomes

$$\sigma_{R_k}^2 = \sigma_{R_a}^2 + \frac{1}{16}\sigma_{R_b}^2 + \frac{1}{16}\sigma_{R_{k-1}}^2 \tag{C.19}$$

which is also a recursive equation. The standard deviation of R_a and R_b are known to be

$$\sigma_{R_a} = \tfrac{RA_R}{\sqrt{2u_aWL}}, \quad \sigma_{R_b} = \tfrac{2RA_R}{\sqrt{2u_bWL}} \tag{C.20}$$

where u is the number of unit resistors realizing the resistor value. The expression for standard deviation of R_k is chosen to be

$$\sigma_{R_k} = \frac{2RA_R}{\sqrt{2u_kWL}} \tag{C.21}$$

where u_k is a function of k. Entering these values into equation C.19 gives an equation only containing u

$$\frac{4}{u_k} = \frac{1}{u_a} + \frac{1}{u_b} + \frac{1}{4u_{k-1}} \tag{C.22}$$

Now let us do the variable substitution $h(k) = 1/u_k$ and state the general

solution

$$h(k) = \alpha + \beta \frac{1}{16^k} \tag{C.23}$$

From figure C.1 we know that $\sigma_{R_1} = \sigma_{R_b}$ so the initial value $h(1) = \frac{1}{u_b}$. $h(2)$ is found using equation C.22. Then the equation system for obtaining α and β is

$$\begin{cases} \alpha + \frac{1}{16}\beta = \frac{1}{u_b} \\ \alpha + \frac{1}{16^2}\beta = \frac{1}{4u_a} + \frac{1}{8u_b} \end{cases} \tag{C.24}$$

Solving the equation system gives

$$\frac{1}{u_k} = h(k) = \frac{1}{15}\left(\frac{4}{u_a} + \frac{1}{u_b} + \frac{\frac{14}{u_b} - \frac{4}{u_a}}{16^{k-1}}\right) \tag{C.25}$$

We are interested in the relative gain error

$$\Delta A_k = \frac{A_k}{2} - 1 \tag{C.26}$$

by differentiating the above equation

$$\frac{\partial \Delta A_k}{\partial R_a} = \frac{R_k + R_b}{2R_k R_b} = \frac{1}{2R}, \quad \frac{\partial \Delta A_k}{\partial R_b} = \frac{R_a}{2R_b} = \frac{1}{8R}, \quad \frac{\partial \Delta A_k}{\partial R_k} = \frac{R_a}{2R_k^2} = \frac{1}{8R} \tag{C.27}$$

the standard deviation of the relative gain error can be resolved

$$\sigma^2_{\Delta A_k} = \frac{A_R^2}{2WL}\left(\frac{1}{4u_a} + \frac{1}{16u_b} + \frac{1}{16u_k}\right) \tag{C.28}$$

By replacing $1/u_k$ by equation C.25, we have an expression for the standard deviation of the relative gain error

$$\sigma^2_{\Delta A_k} = \frac{A_R^2}{30WL}\left(\frac{4}{u_a} + \frac{1}{u_b} + (\frac{14}{u_b} - \frac{4}{u_a})16^{-k}\right) \tag{C.29}$$

To minimize the error u_a and u_b are chosen to be 2 and 1. The result is

$$\sigma_{\Delta A_k} = \frac{A_R}{\sqrt{WL}}\frac{1}{\sqrt{10}}\sqrt{1 + 4/16^k} \tag{C.30}$$

The different ΔA_k will be correlated. This has not been investigated.

C.2.3 The delay in an R-2R ladder

This section contains a derivation of the first order approximation of the delay between the taps in an R-2R ladder with parasitic capacitances. The structure investigated is shown in Figure 4.20. The approach is to map the transfer function on to the general form

$$v_k = a_0(k) \cdot v_0(1 + a_1(k) \cdot sR_aC + a_2(k) \cdot (sR_aC)^2 + \ldots) \tag{C.31}$$

and then approximate the delay with the first order time constant

$$\tau_k = a_1(k) \cdot R_aC \tag{C.32}$$

which will give an accurate estimate of the delay for low frequencies.

Nodal analysis in node k gives

$$v_k = \frac{5}{2}v_{k-1}(1 + sR_aC) - v_{k-2} \tag{C.33}$$

The start values of the recursion are

$$v_0 = 1 \tag{C.34}$$

$$v_1 = 2(1 + \frac{1}{2}sR_aC_0) \tag{C.35}$$

By using the general form of v_k in equation C.33 recursive equations for a_0 and a_1 are found

$$a_0(k) = \frac{5}{2}a_o(k-1) - a_0(k-2) \tag{C.36}$$

$$a_1(k) = \frac{a_0(k-1)}{a_0(k)}\left(\frac{5}{2}a_1(k-1) + 1\right) - \frac{a_0(k-2)}{a_0(k)}a_1(k-2) \tag{C.37}$$

The solution to equation C.36 is trivial and equals

$$a_0(k) = 2^k \tag{C.38}$$

Then the equation C.37 becomes

$$a_1(k) = \frac{5}{4}a_1(k-1) + \frac{1}{2} - \frac{1}{4}a_1(k-2) \tag{C.39}$$

The general solution of equation C.39 is

$$a_1(k) = \alpha k + \beta + \gamma 4^{-k} \tag{C.40}$$

Using the start values and equation C.39 α, β and γ can be solved

$$\begin{cases} \beta + \gamma = 0 \\ \alpha + \beta + \frac{\gamma}{4} = \frac{1}{2}\frac{C_0}{C} \\ 2\alpha + \beta + \frac{\gamma}{16} = \frac{5}{8}\frac{C_0}{C} + \frac{1}{2} \end{cases} \tag{C.41}$$

The solution gives the expression for a_1

$$a_1(k) = \frac{2}{3}k + \frac{6\frac{C_0}{C} - 8}{9}(1 - 4^{-k}) \tag{C.42}$$

Thus, the delay between two taps is

$$\tau_k - \tau_{k-1} = \frac{2}{3}R_aC + 4^{-k}(2R_aC_0 - \frac{8}{3}R_aC) \tag{C.43}$$

C.2.4 The noise at the terminals in an R-2R ladder

Looking into the tap v_k the resistance to the right is $2R_a = R_b$, to the bottom is R_b and to the left is R_b (under the assumption that the source resistance $R_i = R_b$). Then the equivalent resistance is $R_b/3$ thus

$$S_D = 4kT\frac{R_b}{3} \tag{C.44}$$