

Institutionen för systemteknik
Department of Electrical Engineering

Examensarbete

**A High Speed $\Sigma\Delta$ A/D-Converter for a General
Purpose RF Front End in 90nm-Technology**

Examensarbete utfört i Elektronik
vid Tekniska högskolan i Linköping
av

Per Öresjö

LiTH-ISY-EX--07/4017--SE

Linköping 2007



Linköpings universitet
TEKNISKA HÖGSKOLAN

Department of Electrical Engineering
Linköpings universitet
SE-581 83 Linköping, Sweden

Linköpings tekniska högskola
Linköpings universitet
581 83 Linköping

A High Speed $\Sigma\Delta$ A/D-Converter for a General Purpose RF Front End in 90nm-Technology

Examensarbete utfört i Elektronik
vid Tekniska högskolan i Linköping
av


Per Öresjö

LiTH-ISY-EX--07/4017--SE

Handledare: **Stefan Andersson**
isy, Linköpings universitet

Examinator: **Christer Svensson**
isy, Linköpings universitet

Linköping, 6 March, 2007

	Avdelning, Institution Division, Department Electronic Devices Department of Electrical Engineering Linköpings universitet SE-581 83 Linköping, Sweden		Datum Date 2007-03-06
	Språk Language <input type="checkbox"/> Svenska/Swedish <input checked="" type="checkbox"/> Engelska/English <input type="checkbox"/> _____	Rapporttyp Report category <input type="checkbox"/> Licentiatavhandling <input checked="" type="checkbox"/> Examensarbete <input type="checkbox"/> C-uppsats <input type="checkbox"/> D-uppsats <input type="checkbox"/> Övrig rapport <input type="checkbox"/> _____	ISBN _____ ISRN LiTH-isy-ex--07/4017--SE Serietitel och serienummer ISSN Title of series, numbering _____
URL för elektronisk version http://www.ek.isy.liu.se http://urn.kb.se/resolve?urn=nbn:se:liu:diva-8706			
Titel Title	En höghastighets $\Sigma\Delta$ A/D konverterare för en generell radiomottagare i 90nm teknologi A High Speed $\Sigma\Delta$ A/D-Converter for a General Purpose RF Front End in 90nm-Technology		
Författare Author	Per Öresjö		
Sammanfattning Abstract	<p>In this report a transistor-level design of a GHz $\Sigma\Delta$ analog-to-digital converter for an RF front end is proposed. The design is current driven, where the integration is done directly over two capacitances and it contains no operational amplifiers.</p> <p>The clock frequency used for verification was 2.5 GHz and the output bandwidth was 10 MHz. The system is flexible in that the number of internal bits can be scaled easily and in this report a three-bit system yielding an <i>SNR</i> of 76.5 dB as well as a four-bit system yielding an <i>SNR</i> of 82.5 dB are analyzed.</p>		
Nyckelord Keywords	ADC, A/D Converter, analog-to-digital converter, Sigma-Delta		

Abstract

In this report a transistor-level design of a GHz $\Sigma\Delta$ analog-to-digital converter for an RF front end is proposed. The design is current driven, where the integration is done directly over two capacitances and it contains no operational amplifiers.

The clock frequency used for verification was 2.5 GHz and the output bandwidth was 10 MHz. The system is flexible in that the number of internal bits can be scaled easily and in this report a three-bit system yielding an *SNR* of 76.5 dB as well as a four-bit system yielding an *SNR* of 82.5 dB are analyzed.

Acknowledgments

I would like to thank a few people for helping me along the way.

Professor Christer Svensson.

Dr. Stefan Andersson for helping me along the way.

M.Sc. Anton Blad for Matlab code for ADC evaluation.

Laura Reece for proofreading the text.

Abbreviations

AC/DC = A classic metal band

A/D = Analog-to-Digital

ADC = Analog-to-Digital Converter

AHDL = Altera Hardware Description Language

DAC = Digital to Analog Converter

DR = Dynamic Range

ENOB = Effective Number of Bits

FET = Field Effect Transistor

IEEE = Institute of Electrical and Electronics Engineers

LNA = Low Noise Amplifier

MOS = Metal-Oxide-Semiconductor transistor

OSR = Oversampling Ratio = $\frac{f_{sample}}{2f_{bw}}$

RF = Radio Frequency

RMS = Root Mean Square

SFDR = Spurious-Free Dynamic Range

SINAD = Signal-to-Noise-and-Distortion

SNR = Signal-to-Noise-Ratio

TA = Transconductance Amplifier

VLSI = Very-Large-Scale Integration

Contents

1	Introduction	1
1.1	Background	1
1.2	This Report	1
2	Basic Theory	3
2.1	ADC Characterization	3
2.2	ADC:s	3
2.3	$\Sigma\Delta$ -ADC:s	4
2.4	Harmonics and Intermodulation Products	5
3	Hardware Design	7
3.1	Integrator and charge dump	8
3.1.1	Integrator	8
3.1.2	Charge Dump	8
3.1.3	Transmission Gates	12
3.1.4	Schematic of the entire integrator and charge dump	14
3.2	DAC	14
3.2.1	Current Sources and Sinks	14
3.2.2	Switching	15
3.3	Flash-ADC	17
4	Verification	19
4.1	Two tone test	19
4.2	Out-of-band blocker	21
4.3	In-band blocker	22
4.4	Resolution	22
5	Noise	25
5.1	Parasitic capacitances	25
5.2	Thermal noise	25
6	Conclusions and future work	27
6.1	Conclusions	27
6.2	Future work	28

Bibliography	29
A 3-Bit ADC	31
B 4-Bit ADC	33
C Matlab Code for DAC evaluation	36
D Matlab Code for the Low-pass filter	40
E Matlab Code for Measure_sinad	41
F Matlab Code for Sinefit	43
G Graphs and Tables	45

Chapter 1

Introduction

1.1 Background

A software defined radio is a communication system that can be used for a wide range of frequency bands and modulation techniques by changing the software that runs on the system but not changing the hardware. An important part in this system is a wide-band analog-to-digital converter (ADC) that produces a digital signal that can be processed by the software. One approach for this ADC is a $\Sigma\Delta$ -design that uses oversampling with a much higher clock frequency than the output frequency in order to reduce the quantization noise of the ADC.

1.2 This Report

The main part of this report consist of chapter 3, where i describe the hardware design of the project and chapter 4, where i show the verification results from simulations.

The appendices consist of AHDL code used in the design in appendices A and B and Matlab code to verify the design in appendices C, D, E and F and finally in appendix G, graphs and tables with simulation results not included elsewhere in the report.

Chapter 2

Basic Theory

2.1 ADC Characterization

There are several different parameters that are important when analyzing the noise contribution of an analog-to-digital converter.

SNR = Signal-to-noise-ratio describes the relation between the signal power and the noise and is defined as $SNR = 10 \log_{10}(\frac{P_s}{P_n})$ (dB).

SFDR = Spurious-free dynamic range. Harmonics of input frequencies and intermodulation products of two or more input frequencies can cause large unwanted signals at other frequencies. These signals are called spurious signals and the relation between the largest spurious signal and the input signal is called *SFDR* and is defined as $SFDR = 10 \log_{10}(\frac{P_s}{P_{d,max}})$ (dB).

SINAD = Signal-to-noise-and-distortions describes the relation between the signal power and the power of all other frequencies within the channel, both from noise and spurious signals. *SINAD* is defined as $SINAD = 10 \log_{10}(\frac{P_s}{P_{n+d}})$ (dB). *SINAD* is sometimes referred to as *SNDR*.

ENOB = Effective number of bits and stands for the resolution in bits that is possible to achieve after a signal has been digitalized. *ENOB* is directly related to *SINAD* since both noise and spurious signals can affect the output. *ENOB* is defined as $ENOB = \frac{SINAD-1.76}{6.02}$ (dB).

2.2 ADC:s

When a signal is quantized, an error is introduced. The average of this error is half that of one quantization step and a *N*-bit quantization leads to an *SNR* of:

$$SNR_{quantization} = 6.02N + 1.76 \text{ (dB)} \quad (2.1)$$

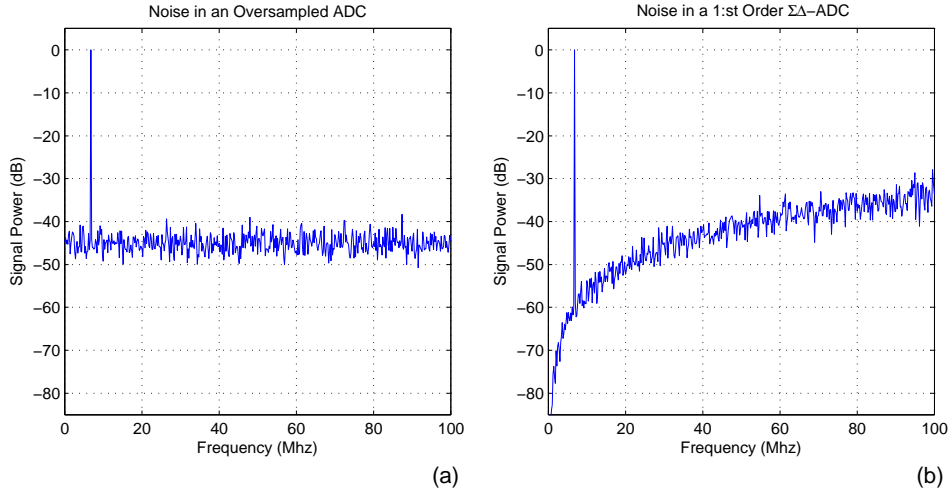


Figure 2.1. Power spectrum of a 10 MHz signal sampled at 200 MHz in an ADC (a) and in a $\Sigma\Delta$ -ADC (b).

This noise is spread evenly over the frequency spectrum from 0 to the Nyquist frequency of the sampling frequency, $f_s/2$ (see figure 2.1 (a)). If the signal is sampled at a much higher frequency than the signal frequency and subsequently low-pass filtered, a large proportion of the noise can be filtered out. The resulting SNR with an oversampling factor of $OSR = f_s/(2f_{bw})$ is [1]:

$$SNR_{oversampling} = 6.02N + 1.76 + 10 \log_{10} OSR \text{ (dB)} \quad (2.2)$$

2.3 $\Sigma\Delta$ -ADC:s

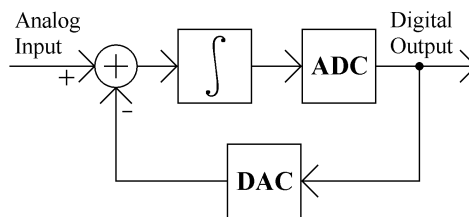


Figure 2.2. In a $\Sigma\Delta$ -ADC, the output is the quantization of the integrated error of the output compared to the input signal. Even if the error for each individual output can be large the average error is small.

The quantization noise of the ADC can be significantly reduced by different types of feedback. In a $\Sigma\Delta$ structure or as it's also referred to $\Delta\Sigma$, the output

error is integrated and subsequently quantized for the next output (see figure 2.2). With this structure each output error will affect the subsequent outputs and as a result the average error will be reduced. In other words, even if most outputs will have a large error, the average of several outputs will have a smaller error.

In a $\Delta\Sigma$ -ADC the total noise in the system is increased compared to the case of non-feedback ADC but the noise is lower at lower frequencies and higher at higher frequencies. The resulting output from a first order $\Sigma\Delta$ is shown in figure 2.1 (b). With this noise-shaping a larger proportion of the quantization noise can be filtered out in a low-pass filter and with the same OSR result in a higher SNR than in a non-feedback system. The SNR of a first order $\Sigma\Delta$ is [1]:

$$SNR_{\Sigma\Delta} = 6.02N + 1.76 - 5.17 + 30 \log_{10} OSR \text{ (dB)} \quad (2.3)$$

2.4 Harmonics and Intermodulation Products

Non-linear properties of a system cause distortion to the output signal. These distortions cause unwanted harmonics to appear at integer multiples of the input frequency ($2f_{in}, 3f_{in}, 4f_{in} \dots$). In a differential system the even-order harmonics of the positive branch and the negative branch of the system will cancel each other out so that only the odd multiples of f_{in} will appear at the output.

When two signals are fed through a system at the same time, they and their harmonics will mix with each other causing intermodulation products at the output [3]. The largest of these intermodulation products will appear at $f_1 + f_2$ and $f_1 - f_2$ called second-order intermodulation products or I_{D2} . Further intermodulation products will appear at $2f_1 - f_2$ and $2f_2 - f_1$ called third-order intermodulation products, I_{D3} and to a lesser extent also I_{D4} , I_{D5} and so on.

If two signals with frequencies close together, in the middle of a frequency band are passed through a system, the third-order will also occur within the band. Because of this I_{D3} can be a big problem and a common way to test a system's linearity is through a two-tone test where two signals with frequencies close to each other are fed to the system to measure I_{D3} .

Chapter 3

Hardware Design

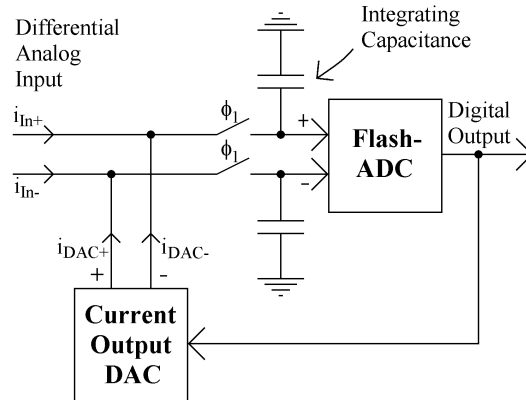


Figure 3.1. Principle schematic of the proposed design.

The proposed structure is a differential first-order $\Sigma\Delta$ -loop with three or four internal bits. The integration in the loop is done over two capacitances, one for each path of the differential system (see figure 3.1).

The input to the system is two equally large but opposite currents, either from a transconductance amplifier or as a current output from a mixer. The peak currents of these inputs are not allowed to be larger than 1.584 mA differentially, the same as the maximum feedback from the DAC. If the inputs are larger than this the DAC can no longer compensate for the input signals and the voltage levels of the integrating capacitances will grow out of control. The signal power of this maximum input is used as a reference for the unit dBfs so that the maximum allowed input is 0 dBfs.

The design is made in a 90nm technology and all transistors used in the design are of low threshold voltage type in order to reduce the on-resistance of switches

and to improve the allowed output voltage swing of the DAC. The supply voltage, V_{dd} of the system is 1.2 V.

To avoid leak-through from the dump capacitances to the integrating capacitances (see section 3.1.2), non-overlapping clocks, ϕ_1 and ϕ_2 have been used (see figure 3.2).

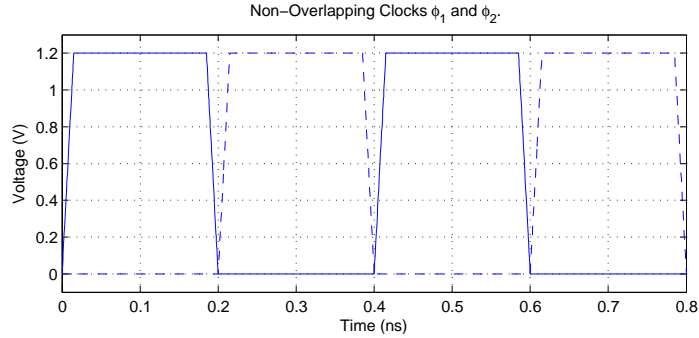


Figure 3.2. The non-overlapping clocks, ϕ_1 and ϕ_2 (dashed). Because the transmission gates contain both n-MOS and p-MOS transistors the complements $\overline{\phi_1}$ and $\overline{\phi_2}$ are also needed.

3.1 Integrator and charge dump

3.1.1 Integrator

The integrating part of the loop is done over two capacitances connected to Gnd , one for the positive and one for the negative side of the differential system. During the positive phase of the ϕ_1 clock cycle, the output of the DAC and the current inputs to the system are connected to the integrating capacitances. During one such integration phase, the differential voltage of the integrating capacitances will change according to:

$$\Delta v_{diff} = \frac{(i_{In,diff} - i_{DAC,diff})}{C_{integration}} \cdot \frac{T_{clock}}{2} \quad (3.1)$$

Note that if the clock frequency is changed, the integrating capacitances have to be changed because of the resulting change in integration time. One way to address this could be to use a structure with variable capacitances and increase $C_{integration}$ when the clock frequency is reduced.

3.1.2 Charge Dump

During the negative phase of the ϕ_1 clock, the integrating capacitances are disconnected from the input and from the DAC. During this time, the currents i_a and i_b in figure 3.3 have to be taken care of. If the currents are not taken care

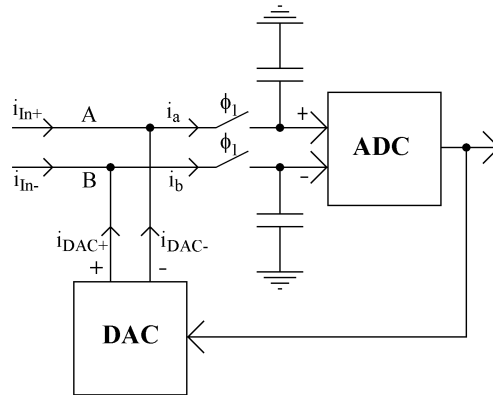


Figure 3.3. If the currents i_a and i_b are not taken care of during the negative phase of the ϕ_1 clock cycle, they would quickly cause the voltage levels of nodes A and B to approach Gnd and V_{dd} .

of the voltages of nodes A and B in figure 3.3 will quickly approach Gnd or V_{dd} and the nodes would have to be recharged to the voltage levels of the integrating capacitances at the start of each integration phase. This would introduce a noise in the system and different strategies could be employed to minimize this noise:

1. The input and the DAC could be turned off.
2. Since $i_a = -i_b$ in figure 3.3, nodes A and B could be short-circuited so the currents would cancel each other out during the negative phase of the ϕ_1 clock.
3. The currents i_a and i_b could be fed to dump capacitances of the same size as the integrating capacitances.

The first strategy has two drawbacks. The first is that for the current sources and current sinks to be turned off, their connections with V_{dd} and Gnd respectively have to be switched off. This adds one transistor in each path from V_{dd} and Gnd to the outputs of the DAC and a similar problem regarding the input currents. This would reduce their allowed voltage swing. The second drawback is that since the transistors at the outputs of the DAC are relatively large, it would take some time for them to be able to deliver their correct output currents after each rising edge of the ϕ_1 clock. On the other hand, one benefit of this strategy would be that since i_a and i_b are turned off when they are not needed, no power would be wasted during the negative phase of the ϕ_1 clock and the overall power consumption and heat generation of the circuit would decrease.

The second strategy also suffers from a few drawbacks. During the positive phase of the ϕ_2 clock, a transmission gate would short circuit nodes A and B in figure 3.5 and the currents i_a and i_b would cancel each other out.

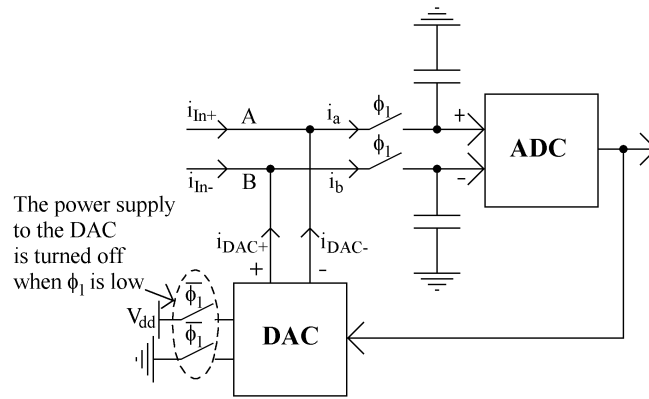


Figure 3.4. Strategy 1: When the power supply to the input current and the DAC is turned off they will stop driving currents and the voltage levels of nodes A and B will remain unchanged during the negative phase of the ϕ_1 clock.

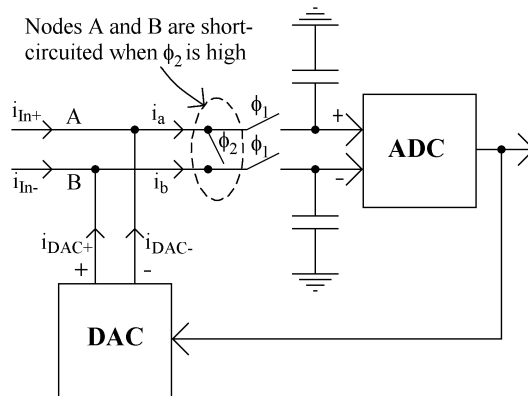


Figure 3.5. Strategy 2: During the positive phase of the ϕ_2 clock, nodes A and B are short-circuited.

One problem with this strategy is that at every rising edge of the ϕ_1 clock, the voltage levels of nodes A and B would have to be recharged from $V_{dd}/2$ to the voltage levels of the integrating capacitances. Another problem is that the non-zero on-resistance of the short-circuit switch would lead to a difference between the voltage levels of node A and node B equal to $i_a \cdot R_{switch}$. Since the switch-resistance can be in the order of 50Ω and the current i_a in the order of 0.2 mA the voltage difference could be as large as 10 mV and not negligible compared with the overall maximum voltage swing of 200 mV .

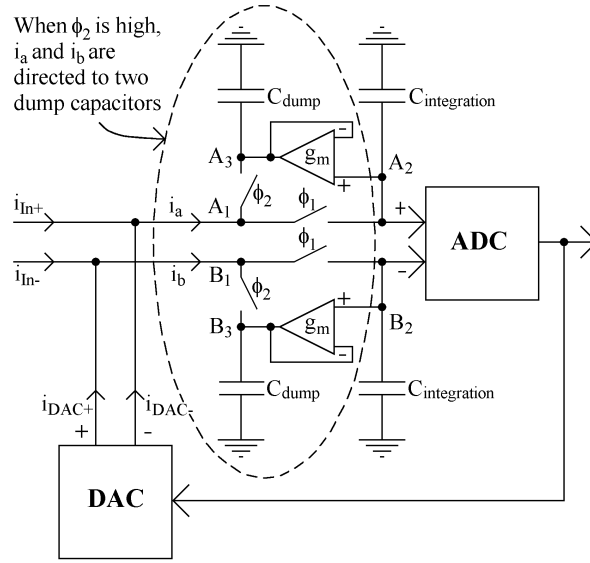


Figure 3.6. Strategy 3: During the positive phase of the ϕ_2 clock, the currents i_a and i_b are directed to two dump capacitances of the same size as the integrating capacitances. A small transconductance amplifier is also added at either side of the differential system to further help the voltage levels of nodes A_3 and B_3 to track those of nodes A_2 and B_2 .

The third strategy is the one implemented in this design. The currents i_a and i_b in figure 3.6 are directed to two dump capacitances during the positive phase of the ϕ_2 clock.

Since i_a and i_b on average are approximately as large during a positive phase of the ϕ_1 clock as in the following positive phase of the ϕ_2 clock, the voltage levels of the dump capacitances (A_3 and B_3 in figure 3.6) will trace those of the integrating capacitances (A_2 and B_2 in figure 3.6) with only a small error. At a rising edge of the ϕ_1 clock, the output of the DAC, nodes A_1 and B_1 will have to be recharged to the voltage levels of A_2 and B_2 . At this time, the voltage levels of nodes A_1 and B_1 are the same as the voltage levels of A_3 and B_3 and therefore it is $\frac{(v_{A3}-v_{A2}) \cdot C_{A1}}{C_{integration}}$ and $\frac{(v_{B3}-v_{B2}) \cdot C_{B1}}{C_{integration}}$, at the rising edge of the ϕ_1 clock that determines the amount of added noise during the transition.

To reduce the noise, either the capacitances C_{A1} and C_{B1} or the voltage differences could be lowered. In order to reduce $v_{A3} - v_{A2}$ and $v_{B3} - v_{B2}$, a transconductance amplifier has been introduced at either side of the differential system (see figure 3.6). When v_{A3} and v_{B3} starts to differ from v_{A2} and v_{B2} , the transconductance amplifiers will start to drive currents to and from the dump capacitances and thereby reduce the difference. These extra transconductance amplifiers will increase the power consumption of the circuit, but since v_{A3} and v_{B3} track v_{A2} and v_{B2} relatively closely, even without these amplifiers, their transconductance does not have to be large in order for the overall performance to be improved.

It is important that the two clocks ϕ_1 and ϕ_2 do not overlap in order to avoid leak-through from the dump capacitances to the integrating capacitances.

The drawbacks with this approach are the need for two extra space-consuming capacitances and the added power consumption of the amplifiers. The schematics of the transconductance amplifiers are shown in figure 3.7.

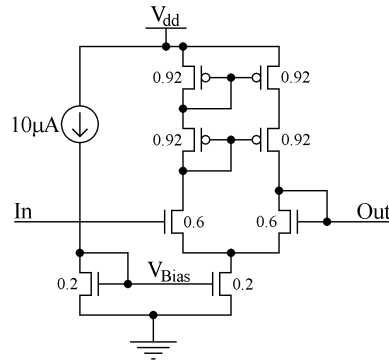


Figure 3.7. Schematic view of the transconductance amplifiers. The numbers by the transistors are their widths in μm . The bottom left transistor together with the current driver is used to set the bias voltage of the amplifier. This voltage can be used for both the amplifier on the positive and on the negative branch of the differential system.

3.1.3 Transmission Gates

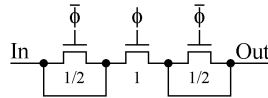


Figure 3.8. Schematic of a n-MOS switch with compensating transistors to reduce clock feed-through.

When a MOS-transistor is used as a switch and the transistor switches, the change in gate voltage affects the source and drain of the transistor through the

gate-drain and gate-source capacitances. To avoid this leak-through, compensating transistors, half the size of the switch transistor can be connected to each side of the switch transistor (see figure 3.8). The drains and sources of these compensating transistors should be short-circuited because it is only the capacitive property of the transistors that is used and the gate connections should be connected to the complement of the gate connection of the switch transistors. With this set-up each time charge leaks through from or to the gate of the switching transistor, an equal amount of charge will leak in the opposite direction to or from the compensating transistors. This design greatly reduces the clock feed-through.

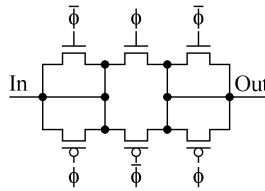


Figure 3.9. Schematic of a complete transmission gate.

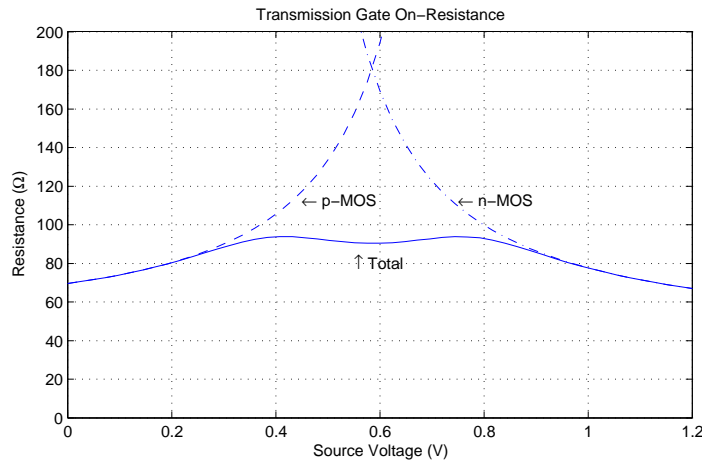


Figure 3.10. On-resistance for $5\ \mu\text{m}$ n-MOS transistor and of a $17.25\ \mu\text{m}$ p-MOS transistor as a function of their source voltages and the on-resistance for both transistors connected in parallel to form a transmission gate.

Because the on-resistance of a transistor depends on $V_{GS} - V_T$, a switch with only a n-MOS transistor would have an undesirably high resistance when the source voltage is close to $V_{dd} - V_T$. To combat this problem it is often desirable to connect a p-MOS transistor in parallel with the n-MOS transistor to create a transmission gate (see figure 3.9). With a single transistor switch the resistance would be

strongly related to the source voltage and therefore be a source of nonlinearities in the system. In a transmission gate on the other hand, at least one of the transistors will be in a low resistive region regardless of the voltages at the input and output. The on-resistance of the transmission gates used in this design is shown in figure 3.10.

3.1.4 Schematic of the entire integrator and charge dump

As you can see in figure 3.11, no compensating transistors have been added to the input node in order to keep the input capacitance low. In order to keep the voltage levels of the integrating capacitances from drifting due to offset errors of the current inputs, or from mismatch errors between the pull-up and pull-down elements of the DAC, two resistors have been added, one from each integrating capacitance to $V_{dd}/2$. The common-mode amplification of the transconductance amplifiers compensate in a similar way for drift in voltage levels of the dump capacitances.

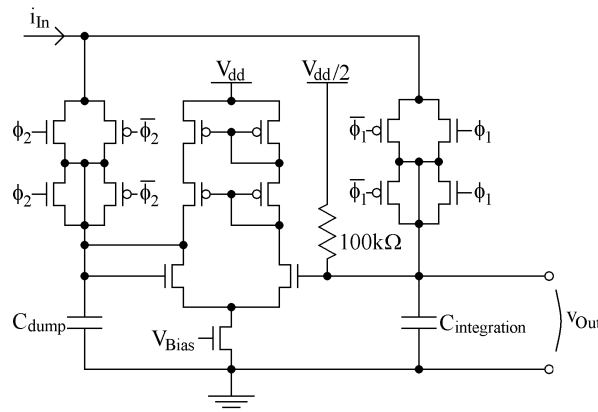


Figure 3.11. Schematic of one of the branches of the integrator and charge dump. Because the two transmission gates operate with opposite clocks, ϕ_1 and ϕ_2 , no compensating transistors are connected at the input node.

3.2 DAC

3.2.1 Current Sources and Sinks

To achieve high enough linearity at the output of the current sources and current sinks regardless of the voltage levels at the output nodes, a cascode structure has been chosen. Because a high output impedance is desired but at the same time a low output capacitance (see section 3.2.2), the first transistor in the source or sink is chosen both wide and long to get a high impedance and the transistor

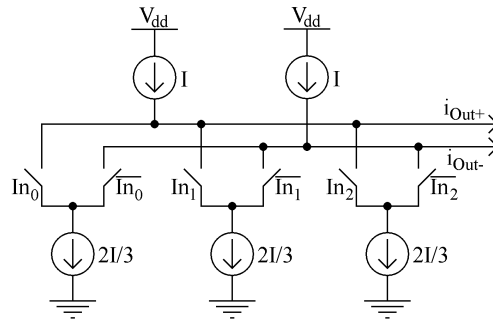


Figure 3.12. Principle structure of a two-bit DAC. The current of the current sinks draws $2/3$ times the current of the current sources so that when all three current sinks are connected to the same output ($3 \cdot \frac{2}{3}I - I = I$) is drawn from the output and at the same time I is added to the other output. In a three-bit DAC, there would be seven current sinks, each with $2I/7$ current and in a four-bit DAC, fifteen current sinks, each with $2I/15$ current.

closest to the output, the cascode transistor is chosen relatively small to get a low capacitance at the output node. Because both transistors in the current sources and sinks must be in saturation for the output current to be linear a cascode structure reduces the allowed output voltage swing.

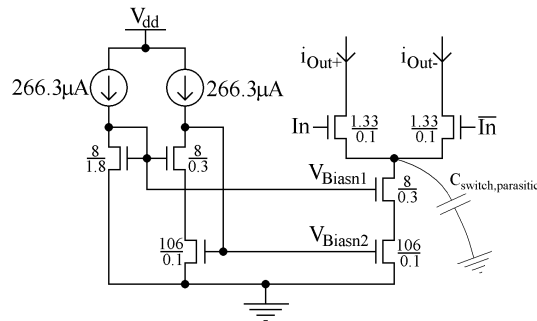


Figure 3.13. Schematic view of a pull-down element of a three-bit DAC. The numbers by the transistors are their widths and lengths in μm . In a four-bit DAC the reference currents and all widths should be multiplied by $7/15$ as there are 7 pull-down elements in a three-bit DAC and 15 pull-down elements in a four-bit DAC. $C_{\text{switch,parasitic}}$ denotes the parasitic capacitance of that node.

3.2.2 Switching

If the thermometer coded input to the switches (see section 3.3) in the pull-down elements of the DAC and their complements do not switch at the same time,

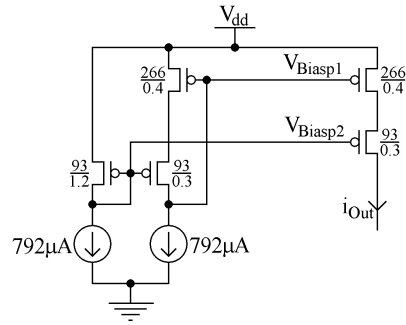


Figure 3.14. Schematic view of a pull-up element of the DAC. The numbers by the transistors are their widths and lengths in μm .

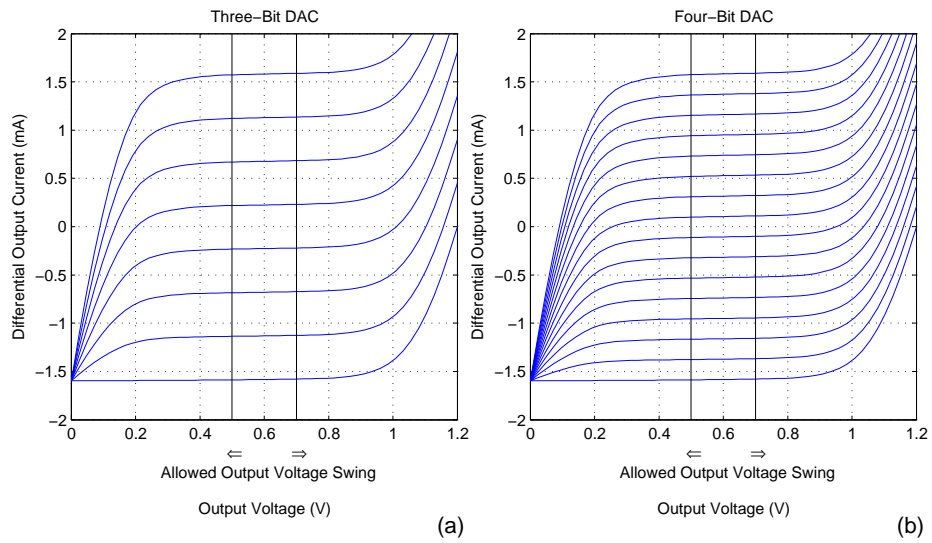


Figure 3.15. Simulated current output as a function of the output voltage for a three-bit DAC (a) and a four-bit DAC (b).

both switches of a pull-down element could be open at the same time or both switches could be closed at the same time. The first case would cause current to leak through from one dump capacitance to the other and thereby increase the noise in the system. The second case would cause the current driver to try to draw current from a low capacitive node that quickly would approach Gnd . It is therefore important that In and \overline{In} in figure 3.13 switch simultaneously.

Even if the input switches correctly, charge will be added to the node that the pull-down element switches in to and the input nodes voltage will change according to (see figure 3.13):

$$\Delta v_{dump,to} = \frac{(v_{dump,from} - v_{dump,to}) \cdot C_{switch,parasitic}}{C_{dump}} \text{ (V)} \quad (3.2)$$

If $C_{switch,parasitic}$ would recharge instantaneously, this effect would only affect the dump capacitances but if the recharge time stretches into the integration phase it will also affect the integration capacitances.

3.3 Flash-ADC

A flash-ADC is an ADC with (quantization levels - 1) comparators that compares the input voltage to different reference voltages to decide the digital output. This is a fast design but since the number of comparators is $2^{number\ of\ bits} - 1$, it is not practical to use when high resolution is needed.

The Flash-ADC in this design has not been realized on a schematic level but programmed in AHDL, Altera Hardware Description Language. The code for a three-bit flash-ADC can be seen in appendix A and the code for a four-bit flash-ADC in appendix B. During the positive phase of the ϕ_2 clock the flash-ADC evaluates the differential analog input and at the falling edge of the clock delivers a digital thermometer coded output, coded according to table 3.1 as well as the complement of the output. The transfer function of a three-bit and a four-bit flash-ADC can be seen in figure 3.11.

Value	Three-bit	Four-bit
15		1111111111111111
13		0111111111111111
11		0011111111111111
9		0001111111111111
7	1111111	0000111111111111
5	0111111	0000011111111111
3	0011111	0000001111111111
1	0001111	0000000111111111
-1	0000111	0000000011111111
-3	0000011	0000000001111111
-5	0000001	0000000000111111
-7	0000000	0000000000011111
-9		0000000000001111
-11		0000000000000111
-13		0000000000000011
-15		0000000000000001

Table 3.1. Thermometer code for a three- and for a four-bit flash-ADC.

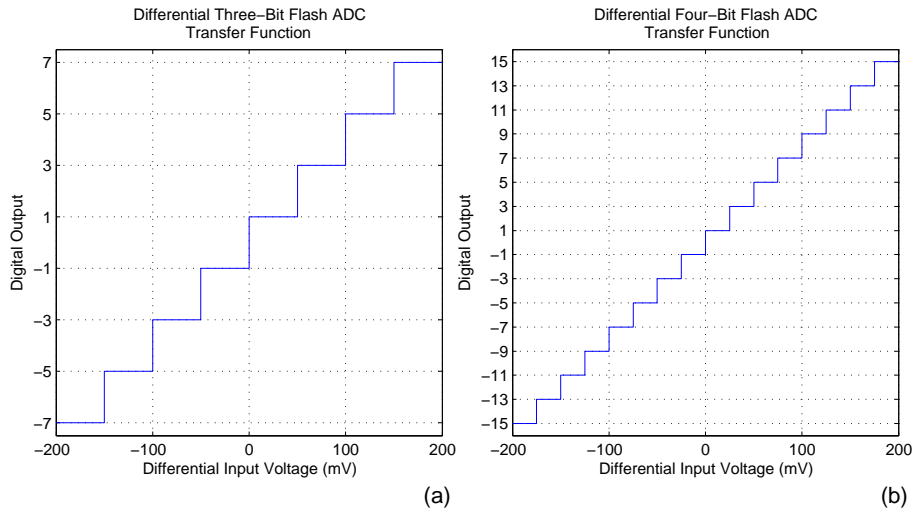


Figure 3.16. Transfer functions for a three-bit ADC (a) and four-bit ADC (b).

Chapter 4

Verification

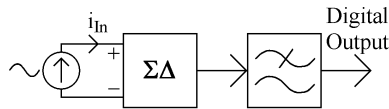


Figure 4.1. Test setup.

The circuit evaluation was done according to the IEEE 1241 – 2000 standard for ADC evaluation [2]. First the digital signal was low-pass filtered through a 15:th order Chebyshev I filter with passband edge at 10 MHz and 0.1 dB ripple in the pass-band (see figure 4.2). After low-pass filtering the input signal was removed by finding the phase and amplitude with least-squares method and then subtracted from the filtered signal. After that the signal power of the input signal was compared to the power of the residual in-band noise.

Each *SINAD* and *SNR* calculation in this report was made with a 20 μ s simulation with a 2.5 GHz clock frequency resulting in 50,000 samples. In order to remove any residues of the initial state and to let the low pass-filter stabilize, the first 17,232 samples were discarded, leaving 32,768 samples to be analyzed.

According to equation 2.2, the theoretical *SNR* for a three-bit first-order $\Sigma\Delta$ with an oversampling ratio of $\frac{f_{clk}}{2f_{bw}} = \frac{2.5 \text{ GHz}}{2 \cdot 10 \text{ MHz}} = 125$ is 77.56 dB leading to an effective number of bits of 12.6, the same values for a four-bit system is *SNR* = 83.58 dB and *ENOB* = 13.6.

4.1 Two tone test

Figure 4.3 shows the output power spectrum from the DAC with two input signals each with -6 dBfs signal power. Since it is the maximum amplitude of the input signal and not the total signal power that decides when the DAC can no longer compensate for the input signal, this is the maximum input allowed to the $\Sigma\Delta$.

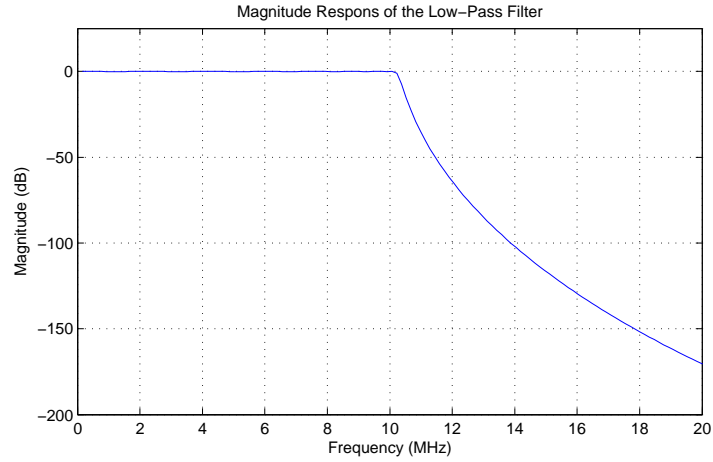


Figure 4.2. Magnitude response of the low-pass filter. The filter is a 15:th order Chebyshev I low-pass filter with a 10 MHz passband and 0.1 dB passband ripple. The attenuation in signal power of the filter is twice as large as the magnitude attenuation, shown in this figure.

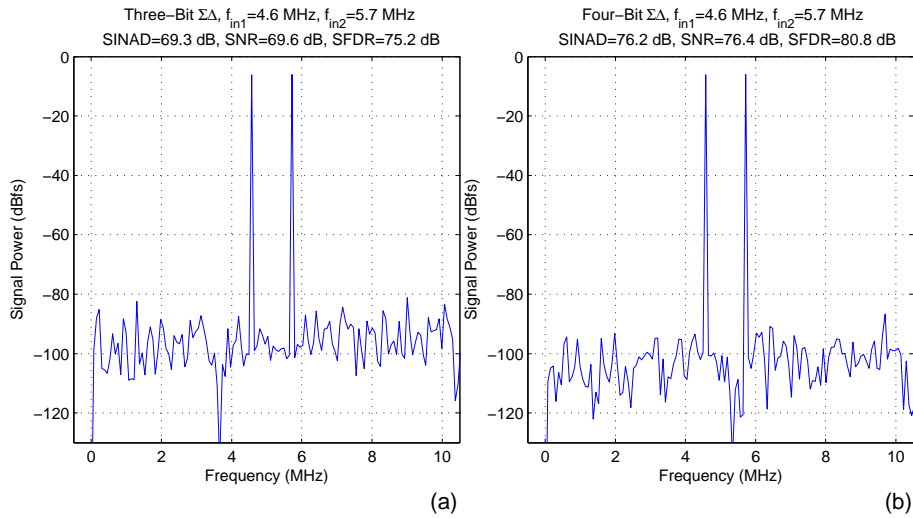


Figure 4.3. Power spectrum of the digital output of a three-bit (a) and a four-bit (b) system with a 4.6 MHz and a 5.7 MHz, -6 dBfs input signal. As can be seen in this figure, the intermodulation products at $2f_{in1} - f_{in2} = 3.5$ MHz and $2f_{in2} - f_{in1} = 6.8$ MHz do not rise above the noise floor.

In the figure it can be seen that the third order intermodulation products at $2f_{in1} - f_{in2} = 3.5\text{MHz}$ and $2f_{in2} - f_{in1} = 6.8\text{ MHz}$ (see section 2.4) do not rise above the noise floor.

4.2 Out-of-band blocker

In figure 4.4, simulated $SINAD$ for a 1.1 MHz input signal with an out-of-band blocker signal of -3 dBfs at 20 MHz is shown for a three-bit and for a four-bit ADC.

To find where the 1.1 MHz signal would be too corrupted by the blocker signal to be detected, we first find at what input signal power level $SINAD$ would fall below 0 dB by linear extrapolation of the simulated data (see figure 4.4).

If the gain of the low noise amplifier before the ADC in the receiver chain can be varied, the input signal to the ADC can be kept at a level for optimum operation. In this case the maximum allowed blocker-to-signal relationship for an out-of-band blocker can be expected to be $P_{blocker} - P_{SINAD=0dB} - SNR_{min}$ (dB) where SNR_{min} is the minimum SNR according specifications of the standard used and $P_{SINAD=0dB}$ is the 0 dB intersection shown in figure 4.4.

For a three-bit ADC the maximum blocker-to-signal relationship is $-3\text{ dBfs} - (-78.2\text{ dBfs}) - SNR_{min} = 75.2 - SNR_{min}$ dB and for a four-bit ADC the relationship is $79.9 - SNR_{min}$ dB.

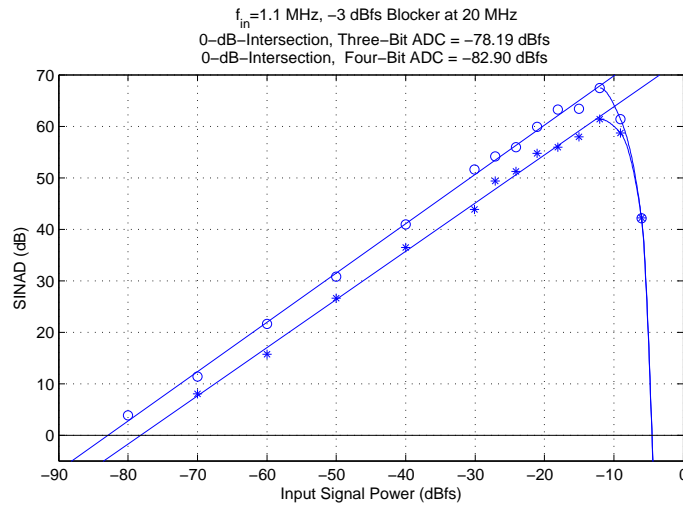


Figure 4.4. Simulated $SINAD$ as a function of input signal power with a -3 dBfs blocker signal at 20 MHz for a three- and a four-bit ADC. The results are extrapolated to find what input power would yield a $SINAD$ of 0 dB.

4.3 In-band blocker

Because some harmonics of a blocker signal with a frequency lower than $f_{bw}/3$ appear within the band (see figure 4.5 (a)), the maximum blocker-to-signal allowed in the system for an in-band blocker will be lower than for a out-of-band blocker.

In figure 4.5 (b) you can see the extrapolation of $SINAD$ as a function of different input signal power and after similar calculations as in 4.2, the maximum blocker-to-signal relationship for an in-band blocker in a three-bit ADC is $64.1 - SNR_{min}$ dB and for a four-bit ADC the relationship is $71.8 - SNR_{min}$ dB.

Nonlinearities in the mixer and in the LNA before the ADC will further degrade the maximum in-band blocker-to-signal relationship.

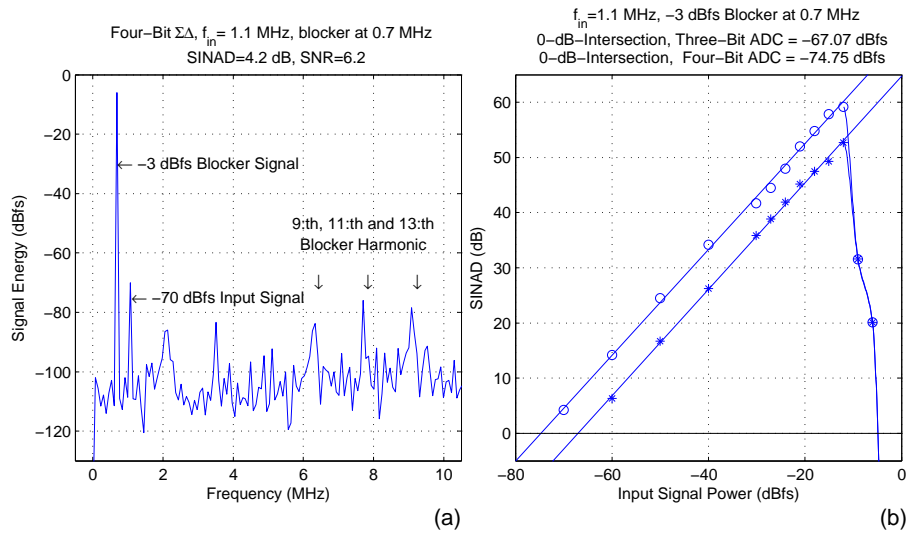


Figure 4.5.

a. Output power spectrum from a 1.1 MHz, -70 dBfs input signal and a 0.7 MHz, -3 dBfs blocker signal from a four-bit ADC. Since the harmonics of the blocker signal occur within the band, they degrade $SINAD$ and SNR for the input signal. The blocker signal was removed before $SINAD$ and SNR was calculated.

b. Simulated $SINAD$ as a function of input signal power with a -3 dBfs blocker signal at 0.7 MHz for a three- and a four-bit ADC.

4.4 Resolution

In figure 4.6 (a) we see that $SINAD$ for a 9.1 MHz input signal is 76.4 dB, allowing a resolution of 12.4 effective number of bits for a three-bit ADC and for a four-bit ADC, $SINAD$ is 83.2 dB, allowing a resolution of 13.5 effective number of bits. These values come close to the theoretical values of 12.6 and 13.6 effective number of bits calculated for an ideal first order $\Sigma\Delta$ -ADC at the beginning of this chapter.

When a signal with a frequency lower than $f_{bw}/3$ is sent to the ADC, the harmonics of that signal will appear inside the output band and $SINAD$ will be reduced (see figure 4.6 (b)). In the case of a 1.1 MHz input signal, $SINAD$ is reduced to 62.0 dB for a three-bit ADC, allowing a resolution of 10.3 effective number of bits and to 75.0 dB for a four-bit ADC, allowing a resolution of 12.1 effective number of bits. If we compare SNR instead of $SINAD$ for a 1.1 MHz input and a 9.1 MHz input (see tables G.1 and G.2), we can see that the SNR is roughly the same for both input signals and it is therefore apparent that this reduction comes from the harmonics of the input signal.

Nonlinearities in the mixer and in the LNA before the ADC will further degrade $SINAD$.

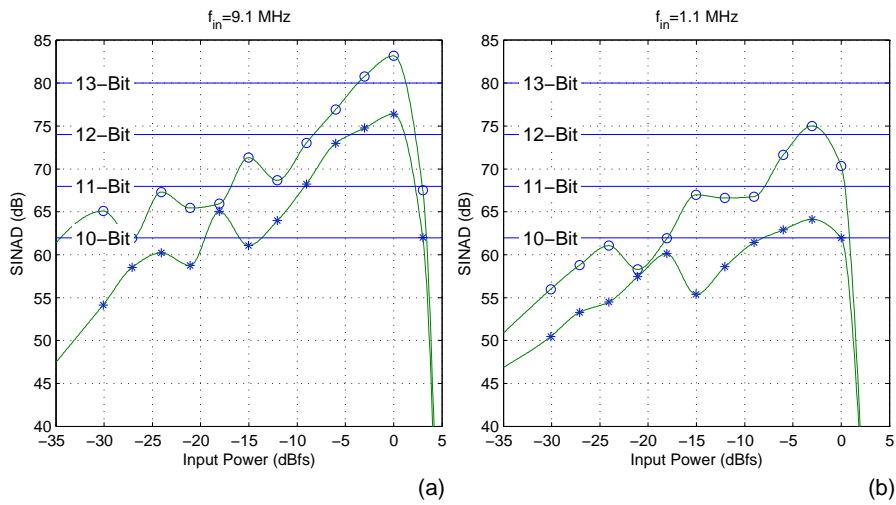


Figure 4.6.

a. Simulated $SINAD$ for a 9.1 MHz input as a function of input signal power for a three- and a four-bit ADC.

b. Simulated $SINAD$ for a 1.1 MHz input as a function of input signal power for a three- and for a four-bit ADC. $SINAD$ is lower than in the case of a 9.1 MHz input signal because several of the harmonics occur within the output band.

Chapter 5

Noise

5.1 Parasitic capacitances

In VLSI design parasitic capacitances are usually a big problem. In this design the part that is most sensitive to parasitic capacitances is the output node of the DAC combined with the output node of the element that generates the input signal to the system. In particular it is very important to keep the capacitance of the pull-down elements low because each time one of these switches, the voltage level of this node has to change from that of one dump capacitance to that of the other (see equation 3.2).

5.2 Thermal noise

The voltage levels of an integrating capacitance will have an uncertainty with a variance of $kT/C_{integration}$ where T is the absolute temperature and k is the Boltzmann constant [3] after each integration period because of the resistive properties of the transmission gate and of thermal noise. Because the flash-ADC has two inputs and each of these is subject to thermal noise, the total noise variance is twice that of the noise variance of one capacitance. The thermal noise is spread evenly over the entire output spectrum from 0 to the Nyquist frequency of the sampling frequency, $f_{sample}/2$. The in-band noise after low-pass filtering will therefore be reduced by the oversampling-factor and the standard deviation of the in-band noise will be $\sqrt{\frac{2kT}{OSR \cdot C_{integration}}}$.

The rms-value of a sinusoidal signal with an amplitude of the maximum differential voltage swing is $V_{FS}/\sqrt{2}$. This value divided by the standard deviation of the thermal noise gives us our maximum dynamic range due to thermal noise:

$$DR_{thermal} = 20 \log_{10} \left(\frac{V_{FS}}{2} \sqrt{C_{integration} \cdot \frac{OSR}{kT}} \right) \text{ (dB)} \quad (5.1)$$

In this system we have $OSR = 125$ and $V_{FS} = 200$ mV and with $T = 300^\circ$ K

and $k = 1.38 \cdot 10^{-23}$ J/K inserted in the equation 5.1, solved for $C_{integration}$ we get the minimum value for the integration capacitance due to thermal noise as:

$$C_{integration} = 1.656 \cdot 10^{-21} \cdot 10^{DR/10} \text{ (F)} \quad (5.2)$$

In this system the integrating capacitances are dimensioned to $C_{integration} = 1.33$ pF to keep the thermal noise 6 dB below the quantization noise of a 13-bit output from a four-bit $\Sigma\Delta$ -ADC. In a three-bit $\Sigma\Delta$ -ADC with a 12-bit output, the capacitances as well as all transistors and bias currents in the system could be reduced to 1/4 size and still keep the thermal noise 6 dB below the quantization noise.

<i>Bits</i>	<i>C_{integration}</i>
10	0.0052 pF
11	0.0208 pF
12	0.0933 pF
13	0.3334 pF
14	1.3336 pF

Table 5.1. Integration capacitances needed to keep the thermal noise at the same level as the quantization noise.

Chapter 6

Conclusions and future work

6.1 Conclusions

In this report I have shown that it is possible to design a GHz $\Sigma\Delta$ -ADC suitable for an RF front-end. This design has an SNR of 76.5 dB for a first order $\Sigma\Delta$ -loop with three internal bits capable of handling a $75.2 - SNR_{min}$ dB out-of-band blocker-to-signal relationship and a $64.1 - SNR_{min}$ dB in-band blocker-to-signal relationship.

For a four-bit system the SNR is 82.5 dB and the system is capable of handling a $79.9 - SNR_{min}$ dB out-of-band blocker-to-signal relationship and a $71.8 - SNR_{min}$ dB in-band blocker-to-signal relationship.

Design characteristics		
	three-bit ADC	four-bit ADC
SNR	76.5 dB	82.5 dB
$SINAD$	64.1 dB	75.0 dB
Maximum OB blocker	$75.2 - SNR_{min}$ dB	$79.9 - SNR_{min}$ dB
Maximum IB blocker	$64.1 - SNR_{min}$ dB	$71.8 - SNR_{min}$ dB
Theoretical $ENOB$	12.6 bits	13.6 bits
$ENOB$ from SNR	12.4 bits	13.5 bits
$ENOB$ from $SINAD$	10.3 bits	12.1 bits

Table 6.1. Characteristics of the two designs analyzed in this report. "Maximum OB blocker" is the maximum allowed out-of-band blocker-to-signal relationship and "Maximum IB blocker" is the maximum allowed in-band blocker-to-signal relationship.

6.2 Future work

Future work could include:

Increase the output swing of the DAC to increase V_{fs} and thus reduce the capacitance sizes needed due to thermal noise (see equation 5.1).

Reduce the output capacitances of the pull-down elements of the DAC to reduce the non-linear properties of the $\Sigma\Delta$.

Introduce a current output mixer as the source of input signal to the $\Sigma\Delta$. It is important to note that due to its design, the $\Sigma\Delta$ -loop mixes the input signal with the clock signal and care has to be taken to avoid noise around the same frequency as the clock frequency.

Bibliography

- [1] Anton Blad, Christer Svensson, Håkan Johansson, and Stefan Andersson. An RF Sampling Radio Frontend Based on $\Sigma\Delta$ -conversion. 2006.
- [2] IEEE-SA Standards Board. IEEE Std 1241-2000, IEEE Standard for Terminology and Test Methods for Analog-to-Digital Converters. 2000.
- [3] Bosco Leung. *VLSI for Wireless Communication*. Prentice Hall, 2002.

Appendix A

3-Bit ADC

```
// Spectre AHDL for ADC3Bit, ahdl
// Analog to Digital Converter with differential input
// and temperature coded output.

module ADC3Bit (
Vtemp0, Vtemp1, Vtemp2, Vtemp3, Vtemp4, Vtemp5, Vtemp6,
VinNeg, VinPos, clk, Vout)
(maxdiff, Vtrans_clk, Vdd, tdel, tfall, trise, filename)
node [V,I] Vtemp0;
node [V,I] Vtemp1;
node [V,I] Vtemp2;
node [V,I] Vtemp3;
node [V,I] Vtemp4;
node [V,I] Vtemp5;
node [V,I] Vtemp6;
node [V,I] VinNeg;
node [V,I] VinPos;
node [V,I] clk;
node [V,I] Vout;
parameter string filename = "simout.dat";
parameter real maxdiff = 0.2 from (0:inf);
parameter real Vtrans_clk = 0.6 from [0:inf];
parameter real Vdd = 1.2 from (0:inf);
parameter real tdel = 0 from [0:inf];
parameter real tfall = 10p from [0:inf];
parameter real trise = 10p from [0:inf];
{
real Vneg, Vpos, Vsens;
integer t[0:6], n, out;
stream outfile;
initial {
Vsens = maxdiff / 4;
outfile = $fopen(filename, "w");
t[0] = 1;
```

```

        t[1]    = 1;
        t[2]    = 1;
        t[3]    = 1;
        t[4]    = 0;
        t[5]    = 0;
        t[6]    = 0;
        out     = 0;
    }
    analog {
// Falling Clock Edge
        if ($threshold(V(clk) - Vtrans_clk, -1)) {
            Vneg = V(VinNeg);
Vpos = V(VinPos);
            out = -7;
                for (n=0; n<=6; n=n+1){
                    if (Vpos-Vneg > (-3+n)*Vsens){
                        out = 2*n-5;
                        t[n] = 1;
                    }else{
                        t[n] = 0;
                    }
                }
    };
    $fstrobe(outfile, "%d\n", out);
    }
    V(Vtemp0) <- $transition( t[0] * Vdd, tdel, trise, tfall);
    V(Vtemp1) <- $transition( t[1] * Vdd, tdel, trise, tfall);
    V(Vtemp2) <- $transition( t[2] * Vdd, tdel, trise, tfall);
    V(Vtemp3) <- $transition( t[3] * Vdd, tdel, trise, tfall);
    V(Vtemp4) <- $transition( t[4] * Vdd, tdel, trise, tfall);
    V(Vtemp5) <- $transition( t[5] * Vdd, tdel, trise, tfall);
    V(Vtemp6) <- $transition( t[6] * Vdd, tdel, trise, tfall);
    V(Vout)   <- $transition( out*0.05, tdel, trise, tfall);
    }
    final {
        $fclose(outfile);
    }
}

```

Appendix B

4-Bit ADC

```
// Spectre AHDL for ADC4Bit, ahdl
// Analog to Digital Converter with differential input
// and temperature coded output.

module ADC4Bit (
Vtemp0, Vtemp1, Vtemp2, Vtemp3, Vtemp4, Vtemp5, Vtemp6, Vtemp7,
Vtemp8, Vtemp9, Vtemp10, Vtemp11, Vtemp12, Vtemp13, Vtemp14,
VinNeg, VinPos, clk, Vout)
(maxdiff, Vtrans_clk, Vdd, tdel, tfall, trise, filename)
node [V,I] Vtemp0;
node [V,I] Vtemp1;
node [V,I] Vtemp2;
node [V,I] Vtemp3;
node [V,I] Vtemp4;
node [V,I] Vtemp5;
node [V,I] Vtemp6;
node [V,I] Vtemp7;
node [V,I] Vtemp8;
node [V,I] Vtemp9;
node [V,I] Vtemp10;
node [V,I] Vtemp11;
node [V,I] Vtemp12;
node [V,I] Vtemp13;
node [V,I] Vtemp14;
node [V,I] VinNeg;
node [V,I] VinPos;
node [V,I] clk;
node [V,I] Vout;
parameter string filename = "simout.dat";
parameter real maxdiff = 0.2 from (0:inf);
parameter real Vtrans_clk = 0.6 from [0:inf];
parameter real Vdd = 1.2 from (0:inf);
parameter real tdel = 0 from [0:inf];
parameter real tfall = 10p from [0:inf];
```

```

parameter real trise      = 10p  from [0:inf];
{
  real      Vneg, Vpos, Vsens;
  integer  t[0:14], n, out;
  stream   outfile;
  initial {
    Vsens   = maxdiff / 8;
    outfile = $fopen(filename, "w");
    t[ 0]   = 1;
    t[ 1]   = 1;
    t[ 2]   = 1;
    t[ 3]   = 1;
    t[ 4]   = 1;
    t[ 5]   = 1;
    t[ 6]   = 1;
    t[ 7]   = 1;
    t[ 8]   = 0;
    t[ 9]   = 0;
    t[10]   = 0;
    t[11]   = 0;
    t[12]   = 0;
    t[13]   = 0;
    t[14]   = 0;
    out     = 0;
  }
  analog {
// Falling Clock Edge
    if ($threshold(V(clk) - Vtrans_clk, -1)) {
      Vneg = V(VinNeg);
Vpos = V(VinPos);
out  = -15;
      for (n=0; n<=14;n= n+1){
if (Vpos-Vneg > (-7+n)*Vsens){
  out = -13+2*n;
  t[n] = 1;
}else{
  t[n] = 0;
}
}
};
$fstrobe(outfile, "%d\n", out);
}
V(Vtemp0) <- $transition( t[0] * Vdd, tdel, trise, tfall);
V(Vtemp1) <- $transition( t[1] * Vdd, tdel, trise, tfall);
V(Vtemp2) <- $transition( t[2] * Vdd, tdel, trise, tfall);
V(Vtemp3) <- $transition( t[3] * Vdd, tdel, trise, tfall);
V(Vtemp4) <- $transition( t[4] * Vdd, tdel, trise, tfall);
V(Vtemp5) <- $transition( t[5] * Vdd, tdel, trise, tfall);
V(Vtemp6) <- $transition( t[6] * Vdd, tdel, trise, tfall);
V(Vtemp7) <- $transition( t[7] * Vdd, tdel, trise, tfall);
V(Vtemp8) <- $transition( t[8] * Vdd, tdel, trise, tfall);

```

```
V(Vtemp9) <- $transition( t[9] * Vdd, tdel, trise, tfall);
V(Vtemp10) <- $transition( t[10] * Vdd, tdel, trise, tfall);
V(Vtemp11) <- $transition( t[11] * Vdd, tdel, trise, tfall);
V(Vtemp12) <- $transition( t[12] * Vdd, tdel, trise, tfall);
V(Vtemp13) <- $transition( t[13] * Vdd, tdel, trise, tfall);
V(Vtemp14) <- $transition( t[14] * Vdd, tdel, trise, tfall);
V(Vout) <- $transition( out*0.025, tdel, trise, tfall);
}
final {
    $fclose(outfile);
}
}
```

Appendix C

Matlab Code for DAC evaluation

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% Matlab code to evaluate an ADC, the program uses the m-files sinefit.m,
% measure_sinad.m and lowpass.m.
% The data to be evaluated has to be at least 16,000 samples long and be
% loaded from a file into the variable InData.
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

InData = load('simout.dat');           % Load the sequence to analyze
fsample = 2500;                        % Sampling frequency in Mhz
fmaxband = 10;                         % Bandwidth to be analyzed
fIn = [1.1];                           % Input frequencies, row vector
lp = lowpass(fsample, fmaxband);       % Create the low-pass filter

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Section to find possible in-band harmonics and intermodulation. What is
% considered in-band in this section is actually 1.1 times the maximum
% frequency of the band in order to remove any spurious with low
% attenuation in the low-pass filter

% Find in-band harmonics
Dist = [];
for n=1:1:5,
    for m=1:size(fIn,1),
        if fIn(m)*(2*n+1)<fmaxband*1.1,
            Dist = [Dist, fIn(m)*(2*n+1)];
        end;
    end;
end;
```

```

% Set maximum harmonic to be evaluate for intermodulation products
IMMax = 3;

% Find in-band intermodulation products
for n=1:1:length(fIn), for m=n+1:1:length(fIn),...
for k1=1:1:IMMax, for k2=1:1:IMMax,
    if ((k1*fIn(n)-k2*fIn(m))<fmaxband*1.1) && ((k1*fIn(n)-k2*fIn(m))>0),
        Dist = [Dist, k1*fIn(n)-k2*fIn(m)];
    end;
    if ((k1*fIn(m)-k2*fIn(n))<fmaxband*1.1) && ((k1*fIn(m)-k2*fIn(n))>0),
        Dist = [Dist, k1*fIn(m)-k2*fIn(n)];
    end;
end; end;
end; end;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Low-pass filtering the input signal
InData = filter(lp,InData);

% Determine the number of fft points possible after at least 16,000 samples
% have been discarded
FFT_Points = 2^(floor(log2(length(InData)-16000)));

% Determine how many initial samples to discard
wait = length(InData)-FFT_Points;

% Determine SINAD and signal power
[SINAD,ND,rmsfIn,rmsND] = ...
    measure_sinad(InData(wait:length(InData)),(2*pi/fsample) *fIn);

% Fouriertransform the input including the signal to determine the signal
% power
fftTot = fft(InData(wait:length(InData)),FFT_Points);
fftTot = fftTot(1:round(fmaxband*FFT_Points/fsample));
fftTot = 10*log10((fftTot.*conj(fftTot)));

% Fouriertransform the remaining signal after the input signals have been
% removed to be able to visualize the inband noise spectrum
fftND = fft(ND,FFT_Points);
fftND = fftND(1:round(fmaxband*FFT_Points/fsample));
fftND = 10*log10((fftND.*conj(fftND)))-max(fftTot);

% Create a variable for the x-data of the plot
f = fsample*(0:round(fmaxband*FFT_Points/fsample)-1)/(FFT_Points);

% If there are no spiruouses, set rmsD to 0 and rmsND to rmsN, otherwise
% determine their rms values. After this rmsIn, rmsN, rmsD and rmsND
% contains the rms values for the insignal, in-band noise, in-band
% distorsions and in-band noise + distorsions respectively

```

```

if isempty(Dist),
    rmsD = 0;
    rmsN = rmsND;
else % Remove the spirouses to determine SNR
    [SNR,Noise,rmsD,rmsN] = measure_sinad(ND,(2*pi/fsample)*Dist);
end;

% Set SNR to rms of the signal compared to the rms of the noise without
% spirouses
SNR = 20*log10(sqrt(sum(rmsfIn.^2))/rmsN);

% Effective number of bits is derived from SINAD
ENOB = (SINAD-1.76)/6.02;

% SFDR is the rms of the input signal compared to the rms of the in-band
% domenating frequency
SFDR = -max(fftND);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Reset the rms value of the signal in the fouriertransformed noise
for n=1:1:length(fIn),
    fftND(round(fIn(n)*FFT_Points/fsample)+1) = ...
        fftTot(round(fIn(n)*FFT_Points/fsample)+1)-max(fftTot);
end;

% Print the calculated values on the screan
fprintf('SINAD = %1.3f dB\n', SINAD);
fprintf('SNR = %1.3f dB\n', SNR);
fprintf('SFDR = %1.3f dB\n', SFDR);
fprintf('ENOB = %1.2f \n' , ENOB);

% Print what harmonics have been removed to determine SNR on the screan
fprintf('Harmonics removed at: ');
fprintf('%1.1f, ',Dist);
fprintf('\n\n');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Section to create the title of the plot with relevant information

titel = ['\Sigma\Delta'];
if length(fIn)==1, titel = [titel ', f_{in}= ', num2str(fIn), ' MHz'];
else
    titel = [titel sprintf(', f_{in%1.0f}=%1.1f MHz',1:1:length(fIn),fIn)];
end;

titel = strvcat(titel, ['SINAD=',num2str(SINAD,'%6.1f'),...
    ' dB, SNR=',num2str(SNR,'%6.1f'),...
    ' dB, SFDR=',num2str(SFDR,'%6.1f'), ' dB']);

```

```
%%%%%%%%%%  
% Plot the spectrum of the insignal  
  
plot(f,fftND);  
  
title(titel);  
  
set(get(gca,'XLabel'),'String','Frequency (MHz)');  
set(get(gca,'YLabel'),'String','Signal Energy (dB)');  
xlim([-0.5 fmaxband+0.5]);  
ylim([-130 0]);  
grid on
```

Appendix D

Matlab Code for the Low-pass filter

```
function Hd = lowpass(Fs,Fpass)

%LOWPASS Returns a discrete-time filter object.
%Fs is the Sampling Frequency and Fpass the Passband Frequency.
% Chebyshev Type I Lowpass filter designed using the CHEBY1 function.

Apass = 0.1;          % Passband Ripple (dB)
N      = 15;          % Filter order

% Calculate the zpk values using the CHEBY1 function.
[z,p,k] = cheby1(N, Apass, Fpass/(Fs/2));

% To avoid round-off errors, do not use the transfer function. Instead
% get the zpk representation and convert it to second-order sections.
[sos_var,g] = zp2sos(z, p, k);
Hd          = dfilt.df2sos(sos_var, g);
```

Appendix E

Matlab Code for Measure_sinad

```
function [sinad,noise,rmswk,rmsn] = measure_sinad(y, wk)
%[sinad,noise,rmswk,rmsn] = measure_sinad(y, wk)
%Computes SINAD (signal to noise and distortion ratio) according to
%IEEE1241 using a three-parameter fit on the angular frequencies
%specified in wk.

N = length(y);
n = 1:N;
K = length(wk);

%Find the sinusoidal components with angular frequencies wk.
[m,p,o] = sinefit(y, n, wk);

%Compute the residual (noise).
xr = zeros(K, N);
for k = 1:K
    xr(k, :) = m(k)*cos(wk(k)*n+p(k));
end
noise = y - sum(xr, 1) - o;

%Compute the total rms signal power.
rmss = sqrt(sum(m.^2/2));

%Compute rms signal power for each frequency.
rmswk = m./sqrt(2);

%Compute the rms noise power.
rmsn = sqrt(1/N*sum(noise.^2));

%Compute the SINAD in dB.
```

```
sinad = 20*log10(rmss/rmsn);
```

Appendix F

Matlab Code for Sinefit

```
function [magnitudes,phases,offset]=sinefit(y,n,wk)
%[magnitudes,phases,offset]=sinefit(y,n,wk)
%Uses a least squares fit to find the amplitudes and phases
%of the sinusoidal components of angular frequencies given
%by wk. A three-parameter fit according to IEEE STD1241 Sec.
%4.1.4.1 is performed.
%
%y is a row vector containing data samples
%n is a row vector containing the sample indeces
%wk is the set of angular frequencies of the sinusoids
%
%The component k is given by
% $x_k = \text{magnitudes}(k) * \cos(\text{wk}(k)*n + \text{phases}(k))$ 
%
%The residual of the fit is given by
% $x = y - \text{sum of } x_k - \text{offset}$ 
%
%[magnitudes,phases,wk,offset]=sinefit(y,n)
%Performs a four-parameter fit according to IEEE STD1241 Sec.
%4.1.4.3. Not implemented.

N = length(n);
K = length(wk);

%Build DO
DO = zeros(N, 2*K+1);
for k = 1:K
    DO(:, 2*k-1) = cos(wk(k)*n)';
    DO(:, 2*k) = sin(wk(k)*n)';
end
DO(:, 2*K+1) = ones(N, 1);

%Compute the least squares fit x0
x0 = (DO'*DO)\(DO'*y');
```

```
%Extract the magnitudes
for k = 1:K
    magnitudes(k) = sqrt(x0(2*k-1)^2 + x0(2*k)^2);
end

%Extract the phases
for k = 1:K
    if x0(2*k-1) < 0
        phases(k) = atan(-x0(2*k)/x0(2*k-1)) + pi;
    else
        phases(k) = atan(-x0(2*k)/x0(2*k-1));
    end
end

%Extract the offset
offset = x0(2*K+1);
```

Appendix G

Graphs and Tables

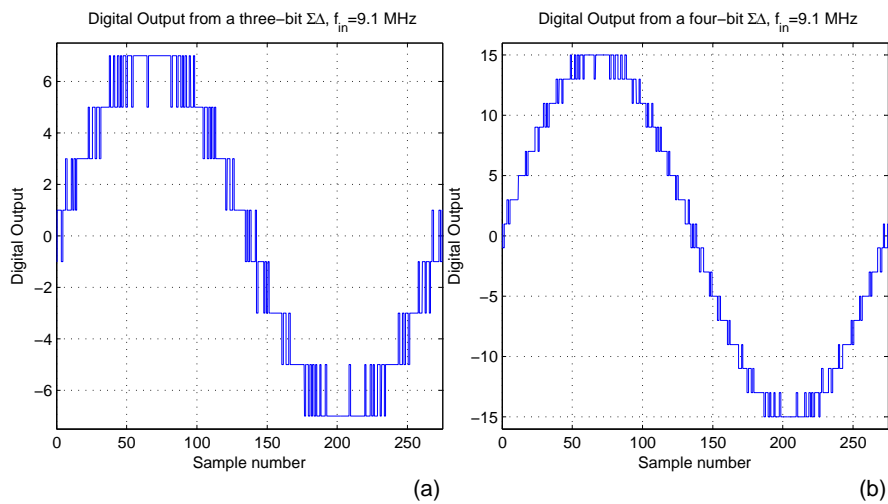


Figure G.1. The digital output with a 9.1 MHz, 0 dBfs input signal for a three-bit (a) and a four-bit (b) ADC.

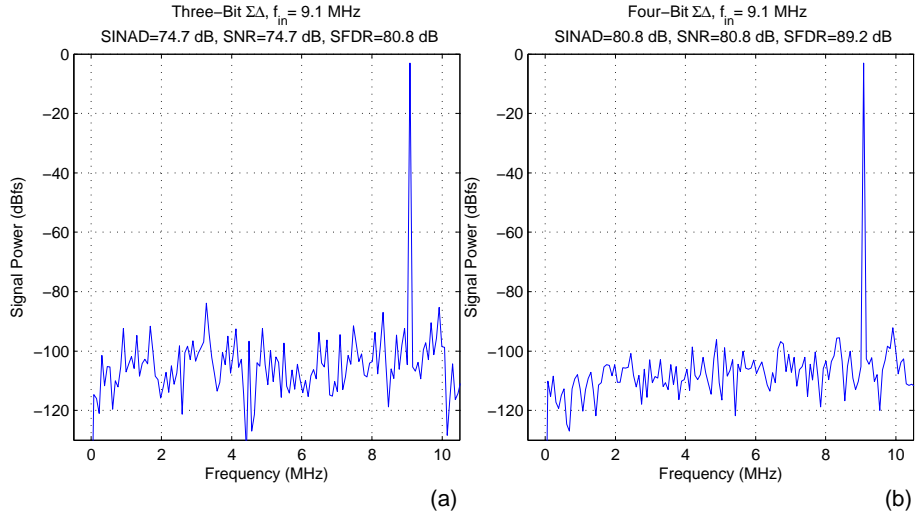


Figure G.2. Power spectrum of the digital output from a three-bit (a) and from a four-bit (b) ADC with a 9.1 MHz, -3 dBfs input signal.

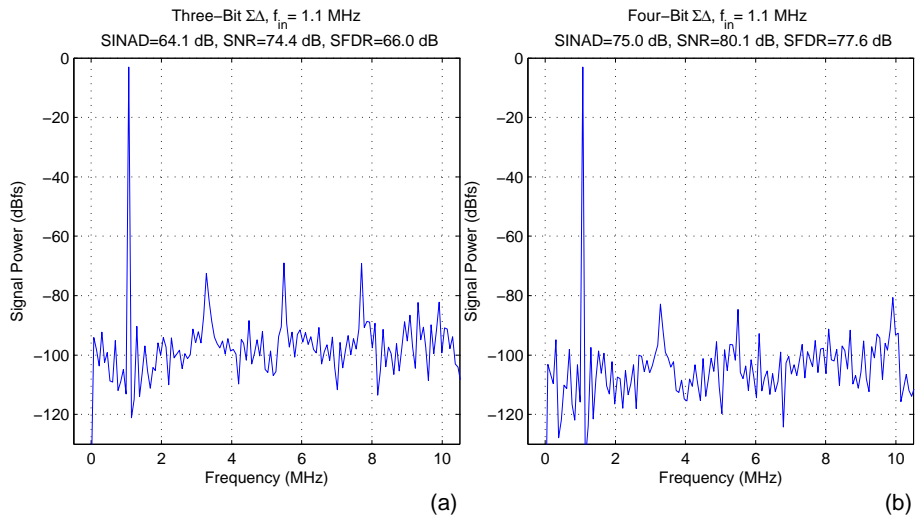


Figure G.3. Power spectrum of the digital output from a three-bit (a) and from a four-bit (b) ADC with a 1.1 MHz, -3 dBfs input signal. Note the harmonics at $3 \cdot f_{in}$, $5 \cdot f_{in}$, $7 \cdot f_{in}$ and $9 \cdot f_{in}$.

Three-bit $\Sigma\Delta$ -ADC						
Input	1.1 MHz		9.1 MHz		4.6 and 5.7 MHz	
	<i>SINAD</i>	<i>SNR</i>	<i>SINAD</i>	<i>SNR</i>	<i>SINAD</i>	<i>SNR</i>
+3 dBfs	17.3 dB	62.0 dB	62.0 dB	62.0 dB	-	-
0 dBfs	62.0 dB	77.0 dB	76.4 dB	76.4 dB	22.8 dB	41.3 dB
-3 dBfs	64.1 dB	74.4 dB	74.8 dB	74.8 dB	69.3 dB	69.6 dB
-6 dBfs	62.9 dB	70.5 dB	73.0 dB	73.0 dB	67.3 dB	67.6 dB
-9 dBfs	61.4 dB	68.7 dB	68.3 dB	68.3 dB	64.5 dB	65.7 dB
-12 dBfs	58.6 dB	65.0 dB	64.0 dB	64.0 dB	63.0 dB	64.0 dB
-15 dBfs	55.4 dB	59.7 dB	61.1 dB	61.1 dB	59.7 dB	61.6 dB
-18 dBfs	60.1 dB	66.1 dB	65.0 dB	65.0 dB	55.6 dB	58.1 dB
-21 dBfs	57.4 dB	59.3 dB	58.7 dB	58.7 dB	56.2 dB	58.1 dB
-24 dBfs	54.5 dB	54.5 dB	60.2 dB	60.2 dB	54.9 dB	55.8 dB
-27 dBfs	53.3 dB	56.9 dB	58.5 dB	58.5 dB	50.0 dB	50.9 dB
-30 dBfs	50.5 dB	52.8 dB	54.1 dB	54.1 dB	46.7 dB	47.2 dB

Table G.1. Simulated *SINAD* and *SNR* for different inputs to a three-bit ADC.

Four-bit $\Sigma\Delta$ -ADC						
Input	1.1 MHz		9.1 MHz		4.6 and 5.7 MHz	
	<i>SINAD</i>	<i>SNR</i>	<i>SINAD</i>	<i>SNR</i>	<i>SINAD</i>	<i>SNR</i>
+3 dBfs	17.3 dB	62.5 dB	67.5 dB	67.5 dB	-	-
0 dBfs	70.3 dB	82.6 dB	83.2 dB	83.2 dB	23.0 dB	41.6 dB
-3 dBfs	75.0 dB	80.8 dB	80.8 dB	80.8 dB	76.2 dB	76.3 dB
-6 dBfs	71.5 dB	76.5 dB	76.9 dB	76.9 dB	74.1 dB	74.6 dB
-9 dBfs	66.8 dB	72.6 dB	73.0 dB	73.0 dB	72.0 dB	73.4 dB
-12 dBfs	67.0 dB	71.3 dB	68.7 dB	68.7 dB	69.2 dB	70.0 dB
-15 dBfs	67.0 dB	70.6 dB	71.3 dB	71.3 dB	67.9 dB	68.1 dB
-18 dBfs	62.0 dB	66.5 dB	65.9 dB	65.9 dB	64.7 dB	65.2 dB
-21 dBfs	58.3 dB	60.4 dB	65.4 dB	65.4 dB	61.3 dB	61.7 dB
-24 dBfs	61.6 dB	66.5 dB	67.3 dB	67.3 dB	57.4 dB	58.2 dB
-27 dBfs	58.8 dB	62.0 dB	61.9 dB	61.9 dB	57.7 dB	58.8 dB
-30 dBfs	56.0 dB	59.2 dB	65.1 dB	65.1 dB	53.0 dB	54.6 dB

Table G.2. Simulated *SINAD* and *SNR* for different inputs to a four-bit ADC.

Three-bit $\Sigma\Delta$ -ADC				
Input	1.1 Mhz, 0.7 MHz Blocker		1.1 Mhz, 20 MHz Blocker	
	<i>SINAD</i>	<i>SNR</i>	<i>SINAD</i>	<i>SNR</i>
-6 dBfs	20.1 dB	20.4 dB	42.2 dB	62.0 dB
-9 dBfs	31.6 dB	32.0 dB	58.7 dB	63.3 dB
-12 dBfs	52.7 dB	53.3 dB	61.4 dB	63.4 dB
-15 dBfs	49.3 dB	50.5 dB	58.0 dB	60.4 dB
-18 dBfs	47.5 dB	48.1 dB	55.9 dB	57.0 dB
-21 dBfs	45.2 dB	45.1 dB	54.7 dB	55.6 dB
-24 dBfs	41.9 dB	42.4 dB	51.2 dB	51.4 dB
-27 dBfs	38.8 dB	39.7 dB	49.3 dB	49.6 dB
-30 dBfs	35.8 dB	37.5 dB	43.9 dB	44.3 dB
-40 dBfs	26.2 dB	27.5 dB	36.2 dB	36.5 dB
-50 dBfs	16.7 dB	18.3 dB	26.6 dB	26.8 dB
-60 dBfs	6.3 dB	7.6 dB	15.7 dB	17.1 dB
-70 dBfs	-	-	8.1 dB	8.4 dB

Table G.3. Simulated *SINAD* and *SNR* for different input signal powers with a -3 dBfs blocker present for a three-bit ADC.

Four-bit $\Sigma\Delta$ -ADC				
Input	1.1 Mhz, 0.7 MHz Blocker		1.1 Mhz, 20 MHz Blocker	
	<i>SINAD</i>	<i>SNR</i>	<i>SINAD</i>	<i>SNR</i>
-6 dBfs	20.1 dB	20.4 dB	42.2 dB	65.8 dB
-9 dBfs	31.6 dB	32.0 dB	61.3 dB	68.6 dB
-12 dBfs	59.2 dB	59.7 dB	67.4 dB	68.4 dB
-15 dBfs	57.8 dB	58.3 dB	63.4 dB	64.5 dB
-18 dBfs	54.8 dB	55.4 dB	63.2 dB	65.4 dB
-21 dBfs	51.1 dB	51.4 dB	59.9 dB	62.1 dB
-24 dBfs	48.0 dB	48.5 dB	55.9 dB	56.4 dB
-27 dBfs	44.5 dB	44.8 dB	54.2 dB	54.3 dB
-30 dBfs	41.7 dB	42.3 dB	51.6 dB	52.0 dB
-40 dBfs	34.2 dB	35.8 dB	40.9 dB	41.1 dB
-50 dBfs	24.5 dB	26.1 dB	30.8 dB	31.1 dB
-60 dBfs	14.2 dB	16.0 dB	21.6 dB	22.3 dB
-70 dBfs	4.2 dB	6.2 dB	11.4 dB	11.8 dB
-80 dBfs	-	-	3.8 dB	4.3 dB

Table G.4. Simulated *SINAD* and *SNR* for different input signal powers with a -3 dBfs blocker present for a four-bit ADC.

Upphovsrätt

Detta dokument hålls tillgängligt på Internet — eller dess framtida ersättare — under 25 år från publiceringsdatum under förutsättning att inga extraordinära omständigheter uppstår.

Tillgång till dokumentet innebär tillstånd för var och en att läsa, ladda ner, skriva ut enstaka kopior för enskilt bruk och att använda det oförändrat för icke-kommersiell forskning och för undervisning. Överföring av upphovsrätten vid en senare tidpunkt kan inte upphäva detta tillstånd. All annan användning av dokumentet kräver upphovsmannens medgivande. För att garantera äktheten, säkerheten och tillgängligheten finns det lösningar av teknisk och administrativ art.

Upphovsmannens ideella rätt innefattar rätt att bli nämnd som upphovsman i den omfattning som god sed kräver vid användning av dokumentet på ovan beskrivna sätt samt skydd mot att dokumentet ändras eller presenteras i sådan form eller i sådant sammanhang som är kränkande för upphovsmannens litterära eller konstnärliga anseende eller egenart.

För ytterligare information om Linköping University Electronic Press se förlagets hemsida <http://www.ep.liu.se/>

Copyright

The publishers will keep this document online on the Internet — or its possible replacement — for a period of 25 years from the date of publication barring exceptional circumstances.

The online availability of the document implies a permanent permission for anyone to read, to download, to print out single copies for his/her own use and to use it unchanged for any non-commercial research and educational purpose. Subsequent transfers of copyright cannot revoke this permission. All other uses of the document are conditional on the consent of the copyright owner. The publisher has taken technical and administrative measures to assure authenticity, security and accessibility.

According to intellectual property law the author has the right to be mentioned when his/her work is accessed as described above and to be protected against infringement.

For additional information about the Linköping University Electronic Press and its procedures for publication and for assurance of document integrity, please refer to its www home page: <http://www.ep.liu.se/>