NTNU

Innovation and Creativity

Automated Self-Test of an Analog Delta-Sigma Modulator

Trond Jarle Pedersen

Master of Science in Electronics

Submission date: June 2007

Supervisor: Trond Sæther, IET

Co-supervisor: Are Hellandsvik, Atmel Norway

Norwegian University of Science and Technology

Department of Electronics and Telecommunications

Problem Description

Background:
Time spent in semiconductor factories on testing manufactured integrated circuits is a large part of the total cost of production. The AVR Microcontroller contains an increasing amount of analog circuitry that pose a great challenge in design for test, both in terms of time spent on testing and the accuracy of test equipment.

Delta-Sigma Modulators are popular in ADCs due to the high accuracy obtainable in low-cost standard CMOS technology. DS-ADCs are however difficult to test efficiently, especially in low-bandwidth, high-oversampling implementations.

Description:
We want to investigate the possibility of implementing a Built In Self-Test of an analog delta-sigma modulator using circuit components available on the AVR Microcontroller. The project objective is to investigate the state-of-the-art in Automated Built-In Self Test, ABIST, for DS ADCs and implementing one on the AVR Microcontroller.

The following topics are of special interest:
- Implementation cost (area, circuit complexity, speed of test)
- Test coverage
- Test equipment requirements

Analysis and detection of output, accuracy of the characterization

Assignment given: 18. January 2007
Supervisor: Trond Sæther, IET

Preface

This thesis is the result from work done as part of a master's degree from Department of Electronics and Telecommunications (IET), The Norwegian University of Science and Technology.

The project title is "Automated Self-Test of an Analog Delta-Sigma Modulator" and was proposed by Are Hellandsvik at Atmel Norway. The objective was to investigate different self-test schemes for a ΔΣ-modulator and suggest a feasible test for integration on Atmel's AVR® 8-bit microcontrollers.

The work was carried out during the spring 2007 at Atmel Norway AS and the Department of Electronics and Telecommunications (IET), The Norwegian University of Science and Technology.

I would like to thank my supervisor Are Hellandsvik at Atmel Norway for his invaluable contribution and continuous guidance throughout the project.

Trondheim 12 June 2007

Trond Jarle Pedersen

® Atmel and AVR are registered trademarks of Atmel Corporation.

Abstract

This project investigates the feasibility of automating the test of ΔΣ-modulators using circuit components available on 8-bit microcontrollers, and by doing so reducing test costs.

A Built-In-Self-Test (BIST) scheme, using a binary stream as stimuli and two different solutions for signal analysis is suggested and simulated in SPICE to investigate its suitability.

The test can not lead to a large area increase, increasing area leads to an increase in production cost. The test has to reduce testing time. The extra area occupied by the test architecture has to be paid in shorter testing time and therefore a lower unit price. The test has to remove or lower the requirements of the off-chip tester, and by doing so reducing cost.

The proposed BIST requires a very small area and is capable of calculating offset, gain and Signal to Noise Ratio with a high degree of accuracy. The proposed solution enables on-chip testing without the need for expensive external stimuli and signal analyzers, making testing on wafer possible thus improving production yield.

The proposed test will not reduce test time by itself, however by integrating the test on-chip and allowing this to run in the background while other on-chip modules are tested total test time can be reduced to the time required to shift the stimuli into the chip

.

Table of contents

1 Introduction 1
2 Background Theory 2
2.1 Anti-aliasing Filter 2
2.2 Sample and Hold 3
2.3 ΔΣ-Modulator 3
2.3.1 ΔΣ-Modulator with Delaying Integrators 4
2.3.2 Quantization Noise and the White Noise Assumption 5
2.3.3 The Oversampling Advantage 6
2.3.4 Noise Shaping 6
2.4 Decimation Filter 8
2.4.1 Hybrid Decimation Filter 8
3 ΔΣ-Modulator Limitations and Non-Idealities 9
3.1 Clock Jitter 9
3.2 Integrator Noise and Non-Idealities 10
3.2.1 Switching Thermal Noise 11
3.2.2 Op-amp Thermal Noise 12
3.2.3 Flicker Noise 12
3.2.4 Gain and Pole-error 13
3.2.5 Slew Rate and Finite Settling Time 13
3.2.6 Saturation and Integrator Clipping 14
3.3 Comparator Non-Idealities 14
3.4 Feedback Non-Idealities 14
4 Failure Modes in Physical Circuits 15
4.1 Failure Mechanisms in CMOS materials 16
4.1.1 Metal Failure modes 16
4.1.2 Gate Oxide Failures 17
4.2 Bridging defects 18
4.2.1 Resistance in Bridges 18
4.2.2 Gate Oxide shorts 19
4.3 Open Defects 20
4.4 Parametric Failures 20
4.4.1 Intrinsic Parametric Failures 22
4.4.2 Extrinsic Parametric Failures 24
4.5 Capacitor failure mechanisms 25
5 ADC Specifications, Terminology and Testing 26
5.1 Static Specifications and Testing 26
5.1.1 Offset and Gain 26
5.1.2 Integrated nonlinearity error 27
5.1.3 Differential nonlinearity error 28
5.1.4 Histogram Testing 28
5.1.5 Coherent Sampling 29
5.1.6 Missing codes 30
5.2 Dynamic specifications 30
5.2.1 Signal to Noise Ratio 30
5.2.2 Signal to Noise and Distortion Ratio 30
5.2.3 Sine-Wave Fitting 30
5.2.4 Effective Number Of Bits 32
6 BIST Implementation Techniques 33

6.1 Existing BIST Proposals ... 34
6.1.1 Integrator Created Ramp Stimulus ... 34
6.1.2 ΔΣ-DAC Created Stimulus ... 35
6.1.3 ΔΣ-Modulator As a Resonator ... 36
6.1.4 Periodic Recorded Stimulus ... 36
6.1.5 Software ΔΣ-Modulator ... 37
6.1.6 Recorded Binary Stimulus ... 38
7 Binary Signals as Stimuli ... 39
7.1 Test bench and Test Setup ... 40
7.1.1 Signal Generation ... 40
7.1.2 Test bench ... 42
7.2 Response and Results ... 44
7.2.1 Detectability in integrator 1 ... 45
7.2.2 Detectability in Bias Circuit ... 48
7.2.3 Detectability in the Common Mode Voltage generator ... 50
7.2.4 Detectability in the Comparator ... 52
7.3 Summary ... 53
8 BIST Proposal ... 54
8.1 BIST Structure ... 54
8.2 BIST Signal Generation ... 55
8.2.1 Dynamic Signal Generation ... 55
8.2.2 Static Signal Generation ... 57
8.3 Signal Quality ... 57
8.4 Signal Analysis ... 58
8.5 BIST Performance ... 61
8.5.1 Dynamic Testing ... 61
8.5.2 Static Testing ... 63
9 Discussion ... 64
9.1 BIST coverage ... 64
9.2 BIST Arithmetic Requirements ... 65
9.3 BIST Implementation Cost ... 66
9.3.1 BIST with External Signal Analysis ... 66
9.3.2 BIST with Internal Signal Analysis ... 66
9.3.3 Test time ... 67
9.3.4 Implementation Summary ... 67
9.4 External Test Equipment Requirements ... 68
10 Conclusion ... 69
11 Future work ... 70
A References ... A-1
B ΔΣ-ADC used in testing ... B-3
B.1 ΔΣ-modulator schematics and VHDL-AMS mapping ... B-5
B.1.1 Op-amp 1 ... B-10
B.1.2 Integrator 1 ... B-11
B.1.3 Bias Circuit ... B-13
B.1.4 Common mode voltage generator ... B-14
B.1.5 Comparator ... B-15
C Pre-study ... C-16
C.1.1 Dynamic Testing ... C-16
C.1.2 Static Testing ... C-17
C.1.3 Integrator Signal Swing ... C-18

D TestBench D-19
D.1 Testbench Code D-20
D.2 DAC Code D-27
D.3 Octave Scripts D-28
D.3.1 Four Parameter Sine Wave Fitting D-29
D.3.2 BIST octave code D-33
E Arithmetic Calculations E-35
F Simulation results Binary signals as stimuli F-37
F.1 Simulation results Integrator 1 .... F-37
F.1.1 Simulation results Op-amp F-37
F.1.2 Simulation results Integrator F-39
F.2 Simulation Results Common Mode Voltage Generator F-41
F.3 Simulation Results Bias Circuit F-41
F.4 Simulation Results Comparator F-42
G BIST Proposal. Dynamic Simulation results G-44
G.1 BIST Simulation results Integrator 1 .... G-44
G.1.1 BIST Simulation results Op-amp, Integrator 1 .... G-44
G.1.2 BIST Simulation results Integrator G-45
G.2 BIST Simulation results Common Mode Voltage Generator G-46
G.3 BIST Simulation results Bias circuit G-46
G.4 BIST Simulation results Comparator G-47

List of figures

Figure 2-1 a ΔΣ-ADC prinsipal schematic 2
Figure 2-2 Boser-Wooley modulator 4
Figure 2-3 Modified modulator structure 4
Figure 2-4 Input vs. output and quantizer error 5
Figure 2-5 ΔΣ-modulator model 7
Figure 2-6 ΔΣ-modulator linear model 7
Figure 3-1 Single-ended SC integrator 10
Figure 3-2 Sampled Capacitor 11
Figure 4-1 Bathtub Failure Rate Curve (Hawkins and Segura, 1999) 15
Figure 4-2 Void and Hillock due to electromigration (Lienig, 2006) 16
Figure 4-3 Metal Stress Void (Segura and Hawkins, 2004) 17
Figure 4-4 Metal Sliver and Metal Blob (Segura and Hawkins, 2004) 18
Figure 4-5 pMOS and nMOS cross section 19
Figure 4-6 Electrical Models For Gate Oxide Shorts (Segura and Hawkins, 2004) 19
Figure 4-7 Resistive Open and missing Vias (Segura and Hawkins, 2004) 20
Figure 4-8 Mousebites 24
Figure 5-1 Offset and Gain errors (Johns and Martin, 1997) 26
Figure 5-2 INL error (Johns and Martin, 1997) 27
Figure 5-3 Histogram testing (Hoeschele, 1994) 29
Figure 6-1 Traditional test 33
Figure 6-2 Static test with ramp generation (Yun-Che and Kuen-Jong, 2000) 34
Figure 6-3 Static test with triangular signal generation (Wang et al., 2005) 35
Figure 6-4 ΔΣ-DAC stimulus generator (Wang et al., 2005) 35
Figure 6-5 ΔΣ-modulator as resonator (Kuen-Jong et al., 2003) 36
Figure 6-6 Stimulus generated from a periodic stream 36
Figure 6-7 Stimulus generated by a software ΔΣ-modulator (Chee-Kian et al., 2004) 37

Figure 6-8 Stimulus generated by recorded binary stream (Rolindez et al., 2006).................. 38
Figure 7-1 Stimulus Spectrum.. 41
Figure 7-2 Test Bench, Binary stimulus... 42
Figure 7-3 Analog Top Level 44
Figure 7-4 Op-amp schematic 45
Figure 7-5 Integrator 1 schematic 47
Figure 7-6 Bias Circuit Schematic 49
Figure 7-7 Common voltage schematic 50
Figure 7-8 Comparator schematic 52
Figure 8-1 BIST architecture.. 54
Figure 8-2 Non-periodic ΔΣ-modulator output.. 55
Figure 8-3 Periodic ΔΣ-modulator output.. 56
Figure 8-4 Optimized ΔΣ-modulator output 56
Figure 8-5 Modulator output spectrum. High-resolution and Periodic binary stimulus 57
Figure 8-6 Modified BIST architecture.. 59
Figure 8-7 Static vs Test Stimuli.. 63
Figure B-1 Boser-Wooley modulator.. B-3
Figure B-2 Modified modulator structure B-4
Figure B-3 Analog Top Level B-5
Figure B-4 Op-amp 1 schematic B-10
Figure B-5 Integrator 1 schematic.. B-11
Figure B-6 Bias schematic B-13
Figure B-7 Common voltage schematic... B-14
Figure B-8 Comparator schematic B-15
Figure C-1 Matlab Filtered Output, DC-stimuli... C-17
Figure C-2 Integrator swing first integrator C-18
Figure C-3 Integrator swing second integrator C-18
Figure D-1 Test Bench, Binary stimulus.. D-19

List of symbols and acronyms

$\Delta\Sigma$	Delta Sigma
Δf	Bandwidth
ADC	Analog-to-Digital Converter
AMS	Analog Mixed Signal
BIST	Build-In-Self-Test
DAC	Digital-to-Analog Converter
DLP	Digital Low-Pass filter
DNL	Differential Non-Linear error
DSP	Digital Signal Processor
DUT	Device Under Test
e	Noise/Error
e_t	Thermal noise voltage
E_{off}	Offset error
E_{gain}	Gain error
ENOB	Effective Number Of Bit
f_0	Signal frequency
f_1	Unity gain frequency
f_s	Sampling frequency

f_{sn}	Nyquist frequency
FFT	Fast Fourier Transformation
g_m	Transconductance
INL	Integral Non-linearity
G	Gain
k	Boltzmann's constant
k_x	Height of the quantization noise spectral density, see section 2.3.2
L	Gate length
LSB	Least Significant Bit
LPF	Low-Pass Filter
M	Sequence length
MUT	Modulator Under Test
N	Number of digitized bit
OSR	Over-Sampling Rate
P_e	mean square quantization noise power
P_s	mean square quantization signal power
S&H	Sample And hold
SC	Switched-capacitance
SINAD	Signal-To-Noise-and-Distortion Ratio
SNR	Signal-to-Noise Ratio
SR	Slew Rate in V/s
T	Absolute Temperature in Kelvin
Vdd	Supply Voltage
W	Gate width

1 Introduction

Time spent in semiconductor factories on testing manufactured integrated circuits is a large part of the total cost of production. Microcontrollers contains an increasing amount of analog circuitry, that pose a great challenge in design for test, both in terms of time spent on testing and the accuracy of test equipment.

ΔΣ-Modulators are popular in ADCs due to the high accuracy obtainable in low-cost standard CMOS technology. ΔΣ-ADCs are however difficult to test efficiently due to their requirement for high-resolution test-stimulus. And due to their oversampling implementation they do not have a direct input to output relationship.

One possible solution to reduce test time and thereby reducing cost associated with testing is to embed an automated Built-In Self Test into the microcontroller. To justify the development and implementation of a BIST, increase in area must be very small and the time required to characterize the modulator can not be larger than the time a traditional test setup uses.

2 Background Theory

Before going into ΔΣ-ADC non-linearity's and their impact on performance a brief introduction to the ΔΣ-ADC is in place. An oversampling ADC works by trading resolution in time with resolution in amplitude, running at a sampling rate much higher than the Nyquist rate they are able to extract extra resolution out of a relatively low resolution ADC. By utilizing negative feedback and thereby shaping the quantization noise-spectrum it is possible to increase the resolution even further. Figure 2-1 shows a principal schematic of a ΔΣ-ADC.

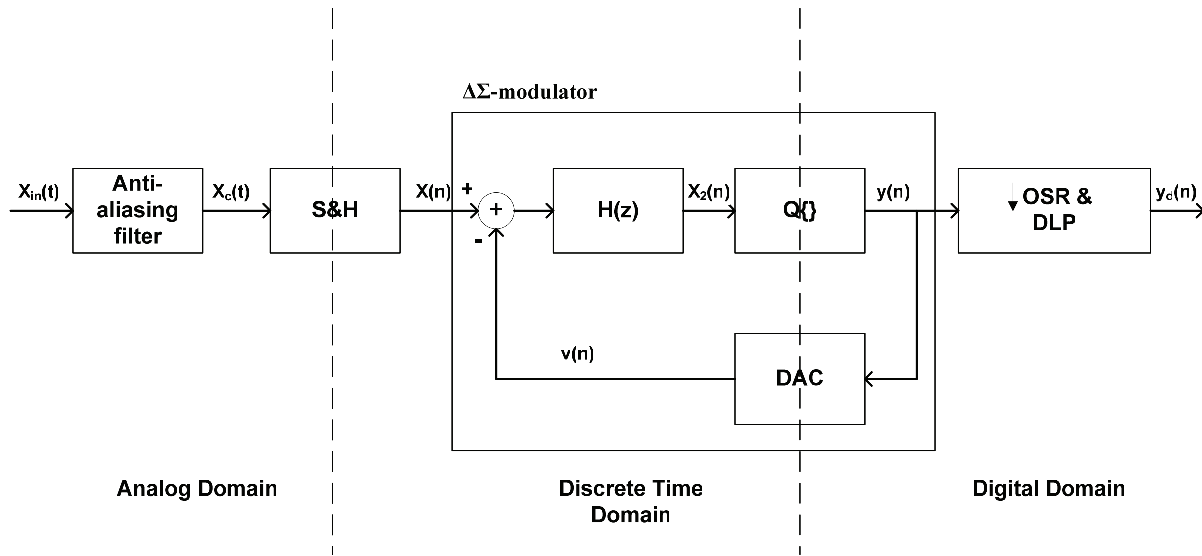

Figure 2-1 a ΔΣ-ADC prinsipal schematic

An analog filer is used as an anti-aliasing filter to remove out-of-band components and thereby preventing aliasing of the signal. The input signal is sampled in the sample and hold circuit at a much higher sampling-rate than the Nyquist-rate. The ΔΣ-modulator acts as a low-pass filter for the input signal, and as a high-pass for the quantization noise provided a proper choice of *H(z)*, this concept is further explained in section 2.3.4. And a quantizer that digitizes the sample, using a 1-bit quantizer is favorable due to its inherent linearity. By utilizing a single bit DAC as feedback source the signal spectrum is shaped and the noise is moved to higher frequencies where a digital low-pass filter can easily remove it at the same time as the signal is down-sampled to the wanted samplerate.(Johns and Martin, 1997)

2.1 Anti-aliasing Filter

This is a analog low pass filter that band-limits the input signal prior to sampling thus preventing aliasing the high-frequency components into the base band. Due to the oversampling nature of ΔΣ-modulators the anti-aliasing filter requirements are not as strict as in a Nyquist rate ADC.

2.2 Sample and Hold

The Sample and hold circuit samples the input signal at a much higher rate than the Nyquist rate. The Nyquist rate f_{sn} is defined as twice the bandwidth of the continuous-time signal. (Proakis and Manolakis, 1996)

$$f_{sn} = 2f_0 \tag{2-1}$$

It should be noted that the Sample and hold circuit is usually implemented in the first integrator of the ΔΣ-modulator along with the feedback and not as a separate component.

2.3 ΔΣ-Modulator

The ΔΣ-modulator consists of an filter *H(z)* and a coarse quantizer *QTODO* enclosed in a feedback loop. The feedback loop and filter, constitute a low-pass filter when seen from the signal input, but as a high pass filter as seen from the quantizer input, where the quantization noise is injected. Thus the over sampled system will leave the signal band unaltered while the quantization noise is pushed towards higher frequencies. Since the ΔΣ-modulator samples the signal at a much higher rate than the Nyquist rate these high frequency noise components can be removed without affecting the wanted frequencies. The simplest ΔΣ-modulators is a first order loop where the filter is realized using a single integrator, however the quantization noise from a first order modulator is highly correlated and the required oversampling rate needed to achieve a resolution over 12 bit is unreasonable large . Throughout this project a second order ΔΣ-modulator with delaying integrators is used.

2.3.1 ΔΣ-Modulator with Delaying Integrators

All testing and development in this thesis is performed on a second order ΔΣ-modulator used to monitor charge and discharge current flowing through an external sense resistor (Atmel, 2005). The modulator is of a modified Boser-Wooley design utilizing delaying integrators. Using delaying integrators is favorable because it allows the op-amps to settle independently, leading to greater signal independency, thus reducing the speed requirements (Boser and Wooley, 1988).

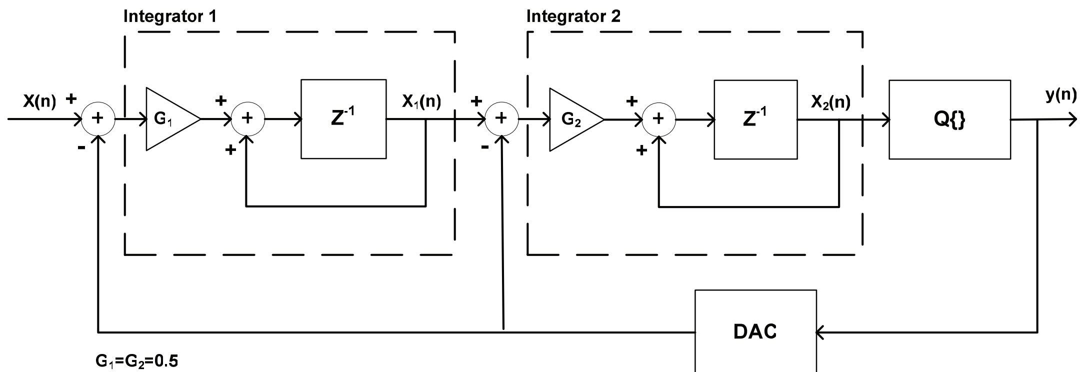

Figure 2-2 Boser-Wooley modulator

In the original design (Figure 2-2) an integrator attenuation of 0.5 was used, however given the maximum input swing of ±220mV specified such a modulator structure would require large sampling capacitors. (Atmel, 2006) The modulator was therefore redesigned with a larger first integrator gain resulting in the block diagram shown in Figure 2-3

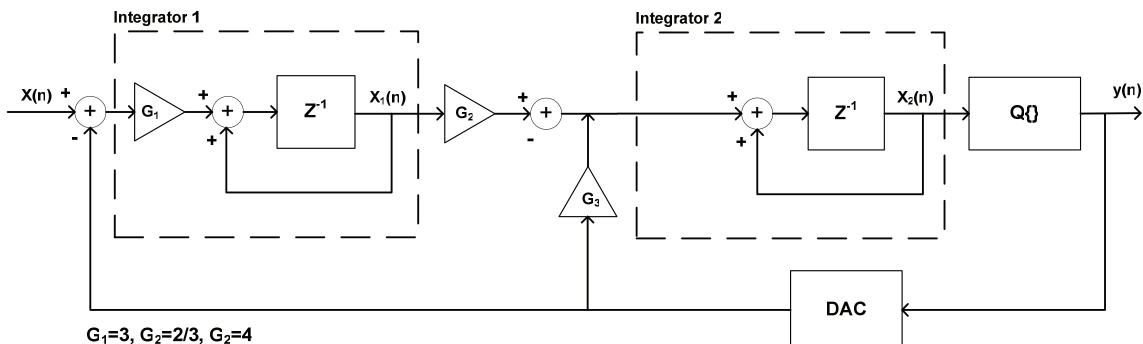

Figure 2-3 Modified modulator structure

The modulator is thoroughly described in appendix B.

2.3.2 Quantization Noise and the White Noise Assumption

To convert a continuous analog signal into a digital one, the signal is sampled and then its amplitude is rounded into a finite value, this quantization is (usually) uniform so that two adjacent quantized values are spaced by a fixed spacing of Δ. The size of Δ is determined by the number of bits used to represent the signal. The deviation *e* from these fixed values to the real analog value is a nonlinear process that never will exceed:

$$-\frac{\Delta}{2} \le e \le \frac{\Delta}{2} \tag{2-2}$$

provided that the analog input is within the signal range.

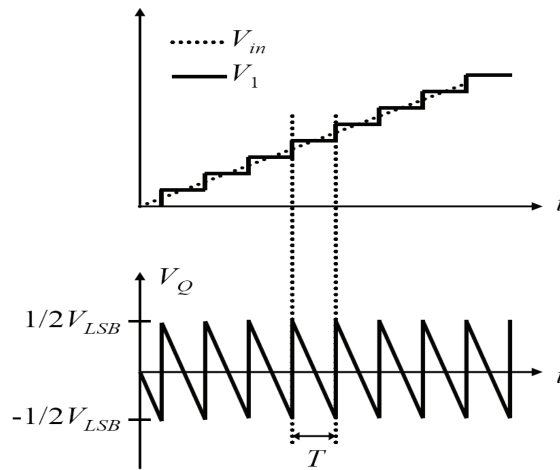

Figure 2-4 Input vs. output and quantizer error

The quantization error *e* is completely defined by the input. If the input signal is very active, one may assume that *e* is uncorrelated from sample to sample. Under this assumption the quantization error can be viewed as a white noise source with samples uniformly distributed between ±Δ/2. The mean-square power of the quantization noise will then be given by: (Johns and Martin, 1997)

$$P_e = \int_{-f_s/2}^{f_s/2} S_e{}^2(f)df = \int_{-f_s/2}^{f_s/2} k_x^2 df = k_x^2 f_s = \frac{\Delta^2}{12} \tag{2-3}$$

where

$$k_x = \left(\frac{\Delta}{\sqrt{12}}\right)\sqrt{\frac{1}{f_s}} \tag{2-4}$$

Where $S_e(f)$ is the spectral density of $e(n)$, and k_x is the height of $S_e(f)$

2.3.3 The Oversampling Advantage

Oversampling is the process of sampling a signal band-limited to f_0 at a rate of $f_s > f_{sn}$, the oversampling rate OSR is defined as:

$$OSR = \frac{f_s}{2f_0} \quad \text{(2-5)}$$

When the sampled signal is filtered through a low-pass filter all frequency components higher than f_0 is attenuated. This filtering leads to a reduction of the quantization noise power to:

$$P_e = \int_{-f_s/2}^{f_s/2} S_e^2(f)|H(f)|df = \int_{-f_0}^{f_0} k_x^2 df = \frac{2f_0}{f_s}\frac{\Delta^2}{12} = \frac{\Delta^2}{12}\left(\frac{1}{OSR}\right) \quad \text{(2-6)}$$

From equation **(2-6)** one can se that a doubling of OSR decreases the quantization noise power by a factor of one half, or equivalent 3 dB.

If the input signal is a sinusoidal wave where the maximum peak amplitude without clipping is $2^N(\Delta/2)$, the signal power P_s will be:

$$P_s = \left(\frac{\Delta 2^N}{2\sqrt{2}}\right)^2 = \frac{\Delta^2 2^{2N}}{8} \quad \text{(2-7)}$$

Where the term N is the quantizers number of bit. The maximum theoretic SNR defined as the ratio between the sinusoidal input signal and the quantization noise (in dB) can be found combining equations **(2-6)** and **(2-7)**

$$SNR_{\max} = 10\log\left(\frac{P_s}{P_e}\right) = 10\log\left(\frac{3}{2}2^{2N}\right) + 10\log(OSR) \quad \text{(2-8)}$$

This translates to:

$$SNR_{\max} = 6.02N + 1.76 + 10\log(OSR) \quad \text{(2-9)}$$

Where the term containing *OSR* represents the gain in SNR obtained from oversampling. As expected from equation **(2-6)** a doubling of the oversampling rate gives a SNR improvement of 3dB/octave (Johns and Martin, 1997).

2.3.4 Noise Shaping

By utilizing feedback a noise shaping effect can be obtained. When feedback is introduced into the modulator as seen in Figure 2-5 and the feedback signal *v(n)* is subtracted from the input signal, quantization noise can be suppressed in the signal band while the signal itself remains largely unaffected. The best way to illustrate this effect is to employ the ΔΣ-modulators linear model.

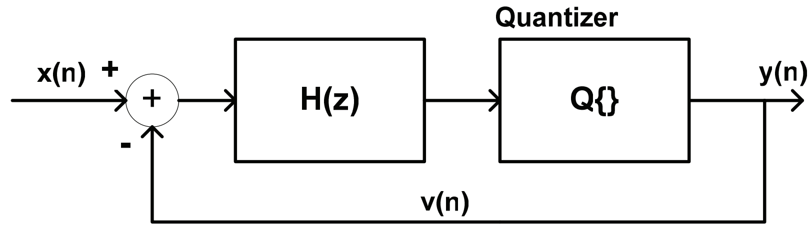

Figure 2-5 ΔΣ-modulator model

Figure 2-5 shows a ΔΣ-modulator and by using the white noise assumption from chapter 2.3.2 the model can be transformed to Figure 2-6.

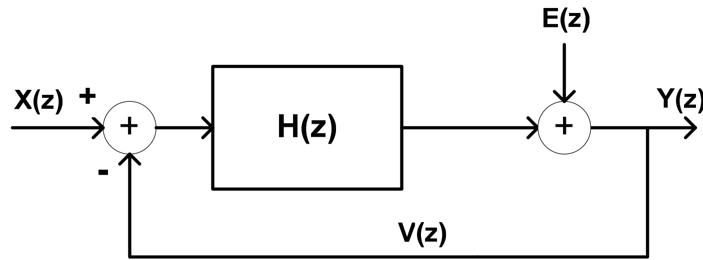

Figure 2-6 ΔΣ-modulator linear model

Treating the linear model as having two independent inputs, the signal transfer function $S_{TF}(z)$ and a noise transfer function $N_{TF}(z)$ can be derived.

$$S_{TF}(z) = \frac{Y(z)}{X(z)} = \frac{H(z)}{1 + H(z)} \tag{2-10}$$

$$N_{TF}(z) = \frac{Y(z)}{E(z)} = \frac{1}{1 + H(z)} \tag{2-11}$$

By writing the output $Y(z)$ as a combination of the noise and signal inputs where each is filtered by its corresponding transfer function as shown in equation **(2-12)** and choosing the loop filter $H(z)$ so that it has a high gain in the signal band, the in-band quantization noise can be attenuated while leaving the input signal intact. (Johns and Martin, 1997)

$$Y(z) = S_{TF}(z)X(z) + N_{TF}(z)E(z) \tag{2-12}$$

Chapter 14.2 in (Johns and Martin, 1997) shows that the theoretic maximum SNR for a second order modulator is

$$SNR_{MAX} = 6.02N + 1.76 - 12.9 + 50\log(OSR) \tag{2-13}$$

where N represents the number of bits in the quantizer.

2.4 Decimation Filter

The high samplerate one-bit output from the ΔΣ-modulator is filtered in a low pass digital filter before being down-sampled into a multi-bit low-samplerate output. This filter has to remove all frequency components above the Nyquist rate $2f_0$, or the remaining high frequency components can be aliased down into the signal band. This requirement is usually fulfilled if a filter with one higher order than the ΔΣ-modulator is used (Norsworthy, 1997).

2.4.1 Hybrid Decimation Filter

Throughout this thesis a hybrid decimation filter consisting of a third order Sinc filter (Proakis and Manolakis, 1996) is cascaded with a first order Sinc filter. The third order Sinc filter decimates the signal and outputs a 13-bit sample every 128th clock sample. The following first order filter decimates the data with a programmable decimation-ratio of 32, 64, 128 or 256 and outputs an accumulated 18-bit signal. Internal registers are 22-bit long.

3 ΔΣ-Modulator Limitations and Non-Idealities

One of the main advantages of ΔΣ-modulators their high tolerance for analog component mismatch and non-idealities but at the same time they call for increased digital complexity. The oversampling nature allows a low amplitude resolution to be traded for high resolution over time. However all circuits are subjected to non-idealities that affect performance by limiting speed and accuracy, and even though ΔΣ-modulators have high insensitivity their performance is affected. This section will investigate important non-idealities, how they affect SC ΔΣ-modulators and how the non-idealities can be detected.

3.1 Clock Jitter

When a signal is moved it from the analog to the discrete domain it is sampled at fixed intervals determined by the sampling frequency f_s, if this interval changes from sample to sample, clock jitter or sampling time uncertainty rises and causes distortion.
Once the signal is sampled the system is a discrete time system thus variations in clock period have no direct effect on the circuit performance. Therefore the effect of clock jitter can be completely described by computing its effect on the input signal sampling. This also means that clock jitter degradation on ΔΣ-modulators is independent of modulator structure and order.

The non-uniform sampling causes an increase in output noise power. The jitter error magnitude is a function of the jitters statistical properties and the modulators input signal. If the input is a sinusoidal signal $x(t)$ with amplitude A and frequency f_{in} that is sampled with a deviation δ from the ideal sampling spacing, (Malcovati et al., 2003) shows that the error introduced will be given by:

$$x(t+\delta) - x(t) \approx 2\pi f_0 \delta A \cos(2\pi f_0 t) = \delta \frac{d}{dt} x(t) \qquad \textbf{(3-1)}$$

By assuming that the sampling time uncertainty δ is a Gaussian random process with a standard deviation of $\Delta\tau$, the resultant error has uniform power-spectral density from 0 to $f_s/2$ with a total power of:

$$e_\delta^2 = \frac{1}{2}(2\pi f_0 \Delta\tau A)^2 \qquad \textbf{(3-2)}$$

By using this assumption and equation **(3-1)** clock jitter can easily be simulated by taking the derivative of the continuous time-signal, multiplying it with δ and adding the result to the acquired sample.

3.2 Integrator Noise and Non-Idealities

When isolating and analyzing noise sources in SC integrators it is convenient to analyze the single ended version shown in Figure 3-1

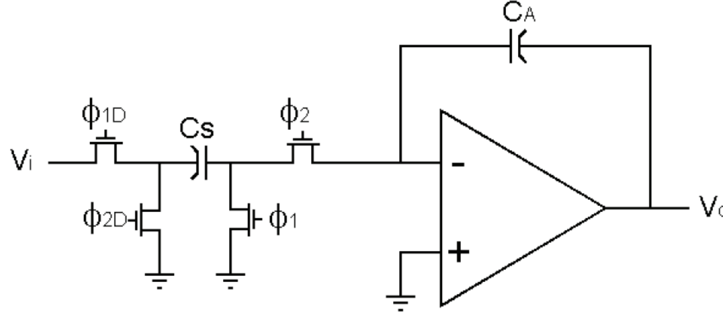

Figure 3-1 Single-ended SC integrator

Due to the noise shaping the input integrator is the one that affects overall performance most, since all subsequent noise contributions are divided by the integrator gain when referred back to the input. The easiest way to estimate the effect a particular non-ideality has on the modulator is to refer it back to the modulator input.

The integrators ideal transfer function is

$$H(z) = \frac{z^{-1}}{1 - z^{-1}} \qquad (3\text{-}3)$$

And the unity gain frequency f_1 will then be:

$$\left|\frac{z^{-1}}{1 - z^{-1}}\right| = \left|\frac{e^{-j2\pi\frac{f_1}{f_s}}}{1 - e^{-j2\pi\frac{f_1}{f_s}}}\right| = 1 \Rightarrow \left|1 - e^{-j2\pi\frac{f_1}{f_s}}\right| = 1 \Rightarrow f_1 = \frac{f_s}{6} \qquad (3\text{-}4)$$

Referring the signal back to its input is the equivalent to dividing it by the transfer function of the integrator. Since ΔΣ-modulators usually employ a large amount of oversampling causing

$$OSR = \frac{f_s}{2f_0} \Rightarrow f_s = 2f_0 OSR \qquad (3\text{-}5)$$

f_s to be large the integrator have considerable gain in the base band, causing noise and distortion due to circuit non-idealities to be greatly attenuated when they are referred back through the integrator. Noise and non-idealities in succeeding integrators will be further attenuated by being referred back through all the integrators. (Norsworthy, 1997)

3.2.1 Switching Thermal Noise

Thermal noise is caused by thermal excitation of carriers in all conductors. (Johns and Martin, 1997) When the sampling transistors on the integrator input are turned on, their equivalent on-resistance acts as a source for thermal noise, a white wideband noise that is proportional to temperature. The effect of thermal noise in SC circuits can be illustrated as shown in Figure 3-2 as a resistor R_{on} in series with an ideal switch that periodically opens, to sample a noise voltage onto the capacitor C.

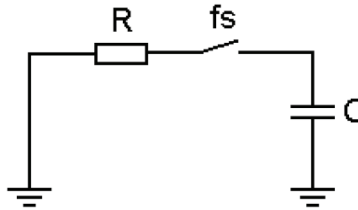

Figure 3-2 Sampled Capacitor

To calculate the total noise power, the resistor can be modeled as a noise source in series with a power source equal to $4kTR\Delta f$ where k is Boltzmann's constant, T absolute temperature and Δf is the signal bandwidth. (Norsworthy, 1997) The total noise power can then be found by evaluating the integral:

$$e_T^2 = \int_0^{\infty} \frac{4kTR_{on}}{1+\left(2\pi f R_{on} C\right)^2} df = \frac{kT}{C} \qquad (3\text{-}6)$$

An interesting property is that although it is the on-resistance that generates the thermal noise, the total noise power is only dependant of the capacitor C.

Assuming that the pole created by the RC time constant is at much higher a frequency than the sampling frequency, the noise power will be aliased down into the frequency band from 0 to $f_s/2$ giving a white final spectrum with a spectral density:

$$S_T(f) = \frac{2kT}{f_s C} \qquad (3\text{-}7)$$

However this is the noise spectral density when the thermal noise is sampled once every clock cycle, while most SC circuits samples twice every clock cycle. The noise spectral density will then be doubled, leading to:

$$S_T(f) = \frac{2kT}{f_s C} \qquad (3\text{-}8)$$

An important note here is that integrators usually include more than one SC branch, leading to the general expression:

$$S_T(f) = \frac{2kT}{f_s} \sum_1^i \frac{1}{C_i} \qquad (3\text{-}9)$$

Where C_i is the *i'th* input capacitor.

Superimposing the thermal noise voltage e_T (also called *kT/C* noise) to the input voltage *x(t)* the following equation gives a simulation model:

$$\begin{aligned} y(t) &= [x(t) + e_T(t)]b \\ &= \left[x(t) + \sqrt{\frac{kT}{C_s}} n(t) \right] b \\ &= \left[x(t) + \sqrt{\frac{kT}{bC_A}} n(t) \right] b \end{aligned} \quad (3\text{-}10)$$

Where *n(t)* is a Gaussian random process with unity standard deviation while $b=C_s/C_A$ is the integrator coefficient for each SC branch. (Malcovati et al., 2003)

3.2.2 Op-amp Thermal Noise

As SC switches, op-amps excite broadband thermal noise, the total noise power in op-amps depends on the transconductance of the selected op-amp structure. The noise can usually be represented as a source with power spectral density $2kTR_{eq}$ at the input terminal where

$$R_{eq} = \frac{4}{3} \frac{1}{g_m} \left(1 + \frac{\alpha}{g_m} \right) \quad (3\text{-}11)$$

represent the equivalent thermal noise resistance. The parameter α will be dependant on the op-amp structure and the process used in fabrication. (Dias et al., 1992)

3.2.3 Flicker Noise

Flicker noise, also called *1/f* noise, has a spectral density inversely proportional to frequency. The noise arises by variations in channel charge. In MOSFET devices flicker noise can be modeled as a voltage source in series with the gate. The source has a spectral power density given by:

$$S_{1/f}(f) = \frac{K}{WL} \frac{1}{f} \quad (3\text{-}12)$$

Where K is an empirical determined constant, W is gate-width and L is gate-length. Flicker noise can be substantially reduced by raising OSR, increasing gate area, correlated double sampling, chopper stabilization and employing auto zeroing integrators. (Norsworthy, 1997)

3.2.4 Gain and Pole-error

The integrator transfer function *H(z)* given in section 3.2, and repeated her for convenience,

$$H(Z) = \frac{z^{-1}}{1 - z^{-1}} \tag{3-13}$$

show an ideal transfer function, Analog circuit implementations will deviate from this due to non-ideal effects. Both pole and gain errors influences the integrators transfer function. The consequence is that only a fraction of the previous output is added to the new input sample. In a single ended integrator such as the one shown in Figure 3-1 the gain and pole-error will lead to the following transfer function:

$$H(Z) = \frac{C_S}{C_A} \frac{(1-\alpha) z^{-1/2}}{1 - (1-\beta) z^{-1}} \tag{3-14}$$

Where α represents gain error and β represents pole-error. The dc gain of the integrator therefore becomes:

$$H_0 = H(1) = \frac{C_S}{C_A} \frac{1-\alpha}{\beta} \tag{3-15}$$

However (Norsworthy, 1997) states that gain and pole error does not have a large effect in single-loop modulators. If integrators have a dc-gain equal to or larger than the OSR increase in base band noise will be less than 0.3 dB.

3.2.5 Slew Rate and Finite Settling Time

Speed constraints given by the RC-time constant in the op-amp limits maximum sampling frequency by introducing a finite settling time. The RC-time constant follows a slope that is called slew rate. Settling time can be described as the minimum time required for the output of the op-amp to reach its new value between each sample. Sampling at shorter intervals than the settling time or if the settling time is changed by circuit imperfections harmonics distortion is introduced into the circuit. This noise may be interpreted as a nonlinear gain and can be described by two separate cases.
Case one is when the sought change in output is able to settle in time, in this case the output reaches its target within one finite sample and no noise is introduced.
Case two appears when the change in output is higher than the op-amps slew rate, the integrator is not able to settle fully and the output *v(n)* will then be given by:

$$v(n) = v(n-1) + (\alpha V_{dd} - SRn)\left(1 - e^{-\frac{n_{\max} - n}{\tau}}\right) \tag{3-16}$$

Where *v(n-1)* is the previous output, α is op-amp gain, SR is slew rate in V/s, n_{max} is the sample time and τ is the time constant.

3.2.6 Saturation and Integrator Clipping

Physical SC integrators have an upper and lower boundary that is determined by the output range of the op-amp. When the sought integrator output exceeds the op-amps maximum range the integrator will saturate. This indicates that even if the input rise the output will be fixed at the maximum output. This behavior is called clipping and introduces nonlinear noise into the modulator that degrades its performance. Integrator clipping can be simulated by setting a maximum signal swing.

3.3 Comparator Non-Idealities

The comparator or quantizer quantizes the signal in the modulator loop and by so doing it generates the modulator output. This output is also fed to the one-bit DAC where it is converted back to an analog signal to provide feedback. Since the quantizer is placed in the feedback loop after the loop gain blocks and before the output terminal distortion generated by this is noise-shaped and effectively attenuated in the base band.

3.4 Feedback Non-Idealities

The feedback loop consists of a DAC that reconvert the digital output back into the analog feedback signal. Due to its placement in the feedback loop it has to have high accuracy and linearity, using a one-bit DAC in the feedback guaranties differential linearity by having only one step. But the output is the product of the digital input and the analog voltage reference and inaccurate unstable or skewed references can cause distortions in the modulators transfer function and by doing so creating integral nonlinearity and harmonic distortions.

4 Failure Modes in Physical Circuits

Modern integrated circuits contains several million transistors, miles of metal interconnect lines and several millions vias and contact that all work together to form a high performing entity that contains both digital and analog circuits.
One common used fault classification in analog circuits is to divide the faults into two categories:

- Catastrophic Faults or Hard Faults
- Deviation Faults or Soft Faults.

Hard faults describe defects that cause shorts, open circuits or large deviations from the expected performance. Soft Faults represent parametric deviations from the nominal value that causes the circuit to perform outside its intended performance band.

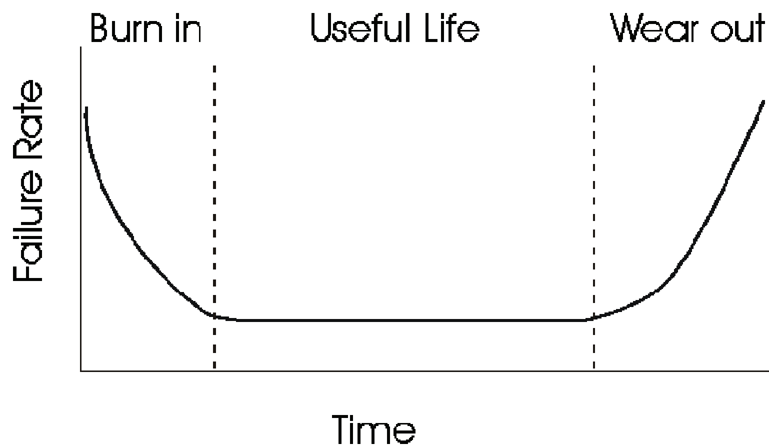

Figure 4-1 Bathtub Failure Rate Curve (Hawkins and Segura, 1999)

Figure 4-1 shows the bathtub curve expressing product failure rate as a function of product life. The early large failure rate is attributed to defective (extrinsic) material and is often referred to as the infant mortality phase. The lower, nearly constant failure rate is the stable useful life of the product that typically declines with time for modern ICs. The stable life period has a finite failure rate due to a low level of residual defects or due to electrical overstress/electrostatic discharge events. The final increase in failure rate occurs in intrinsic material and is due to wear out mechanisms.(Hawkins and Segura, 1999)

This chapter will try to highlight some of the failure mechanisms that exist in integrated circuits. It is deliberately placed at a high abstraction level, and does not deal with the underlying effects in detail.

4.1 Failure Mechanisms in CMOS materials

Metal and oxide materials failure modes have always been with us, but they are now more significant in the submicron technologies. This chapter tries to highlight the failure modes and effects that material defects might cause.

4.1.1 Metal Failure modes

Metal structures, such as interconnect lines, pads and vias is essential to modern integrated circuits. Metal must be made without defects and must not fail over time.

4.1.1.1 Electromigration

Electromigration is the movement of metal under the influence of electron flow and temperature. Any metal line will fail if sufficient current density and/or high temperature are applied. Electrons are believed to transfer a small but sufficient momentum to thermally active metal atoms forcing those atoms out of their lattice sites, and moving them in the same direction as the electrons.
To start the electromigration process an imperfection or a defect in the metal structure must exist. Unfortunately, all metals have, to some extent, flaws and defects that can initiate Electromigration. The non-uniform atomic flow rates, caused by these flaws, through different sections of the conductor result in mass depletion, and mass accumulation.(Lienig, 2006) (Segura and Hawkins, 2004)

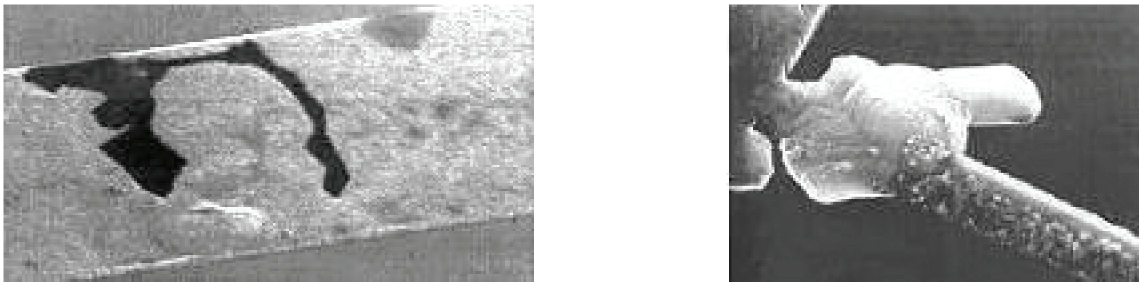

Figure 4-2 Void and Hillock due to electromigration (Lienig, 2006)

The mass depletion leads to resistive bridges and eventually opens in the circuits, while the mass accumulation can lead to shorts.

4.1.1.2 Metal Stress Voiding

The effect when metal lines is pulled apart due to the difference in the thermal coefficient of expansion between the metal line and the pacification materials surrounding it is called stress voiding, (or stress induced voiding), it might cause open faults and/or fertile conditions for electromigration.

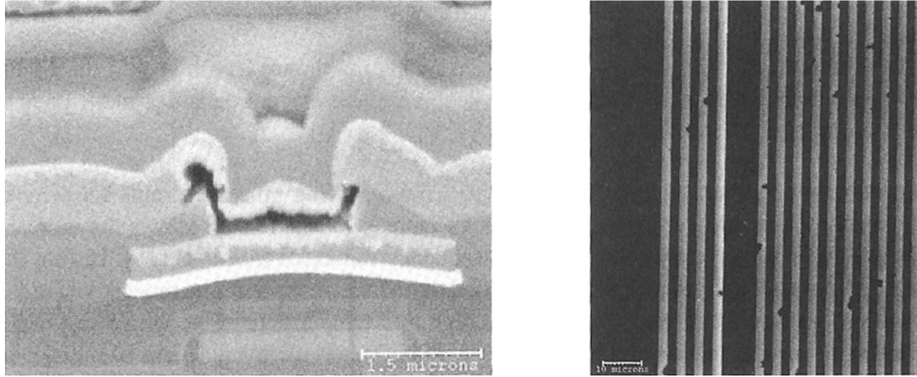

Figure 4-3 Metal Stress Void (Segura and Hawkins, 2004)

4.1.2 Gate Oxide Failures

In MOS transistors the gate is separated from the channel by a thin dielectric silicone-dioxide (SiO_2) layer. Gate oxide failure is the destruction of this dielectric layer. Gate oxide failure is also sometimes referred to as gate oxide rupture or gate oxide breakdown, and often manifests as a short or leakage path from the gate to the channel or substrate.

4.1.2.1 Oxide Wearout

All gate oxides wear out and rupture if a charge is continuously injected. This effect has nothing to do with defects from fabrication. Each time a transistor has a voltage put across its gate oxide, a small amount of charge is injected into the oxide. This charge slowly generates defects within the oxide until a defect path links the gate terminal to the substrate. When this path is established a soft breakdown increases the gate current and noise. Eventually a hard breakdown appears where the gate current exponentially increases. (Hawkins et al., 1999)

4.1.2.2 Hot Carrier Injection

When the transistors electric field at the drain-to-channel depletion is too high an electron can gain sufficient kinetic energy to overcome the barrier that the oxide layer poses and degrades this, causing electron and hole traps to form and by so doing increase leakage current, shift in threshold voltage, and change in transconductance. It will eventually cause the transistor to fail.(Segura et al., 2002)

4.1.2.3 Defect-induced Oxide Breakdown

Foreign particles or poor quality oxides can lead to premature breakdown of the oxide for lower voltages than those caused by Hot Carriers. These are called oxide short and are discussed in section 4.2.2

4.2 Bridging defects

A bridge or shorting failure is an unintentional connection between to or more circuit nodes. Bridges induce abnormal behavior depending on the resulting circuit topology. Bridging defects usually can be placed in the following categories: (Segura and Hawkins, 2004):

- Ohmic or nonlinear bridge defects
- Intragate-connections across transistors internal nodes.
- Connections across the I/O nodes of logic gates
- Power to ground rail bridges
- Combinational or sequence resulting circuits topology
- Interconnect material types – metal, poly and diffusion region
- Critical resistance- transistor drive strength and W/L ratios

Ohmic bridge defects may be metal slivers that connect to interconnection lines, large amount of material shorting more than one interconnect or certain transistor gate oxide shorts as seen in Figure 4-4. (Segura and Hawkins, 2004)

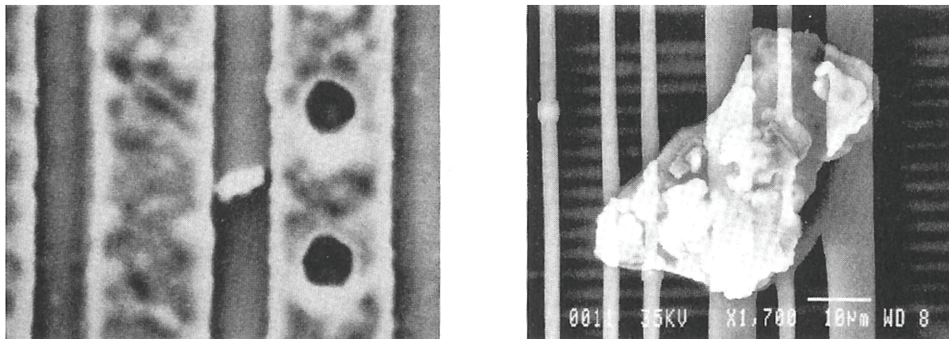

Figure 4-4 Metal Sliver and Metal Blob (Segura and Hawkins, 2004)

4.2.1 Resistance in Bridges

If a circuit defect shorts two logic nodes the resulting resistance between these two nodes, called bridging resistance, may or may not cause the surrounding circuit to fail. Based on this, one can define a critical resistance, which is defined as the boundary resistance where the circuit will not functionally fail.

4.2.2 Gate Oxide shorts

Gate oxide Shorts (GOS) describes defects caused by hard oxide breakdown from particles or oxide imperfections.

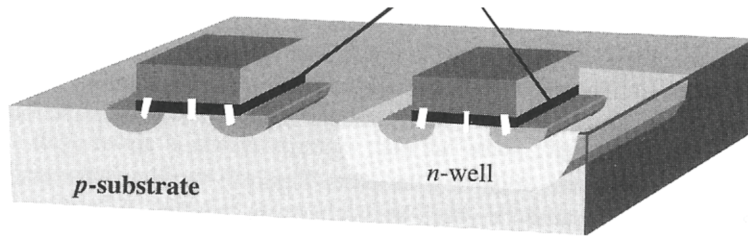

Figure 4-5 pMOS and nMOS cross section

Figure 4-5 show a cross section of pMOS and nMOS transistors in CMOS technology. Gate to drain/source oxide shorts have simple electrical models. There are six places where a gate short can acquire a distinct parasitic connection when the gate material merges with the substrate material. Since gate oxide shorts connect the gate polysilicon with the drain, source or bulk of the device, the electrical properties of the contact depend on the doping type of the terminals being shorted. If the gate and diffusion are of the same doping type, then the electrical model is a resistor between both terminals. If the shorted region has the opposite doping, the electrical model is a *pn* junction diode. (Segura and Hawkins, 2004). This leads to the simulation models shown in Figure 4-6

n- type polysilicon			*p*-type polysilicon		
GS	GB	GD	GS	GB	GD

Figure 4-6 Electrical Models For Gate Oxide Shorts (Segura and Hawkins, 2004)

4.3 Open Defects

Open circuit defects are unintentional electrical discontinuities. They cause behavior that may vary greatly and be difficult to predict. The defects include open contacts (missing metal or unopened oxide), metallization opens (patterning, improper etching, electromigration, or stress voiding), or opens in diffusion or polysilicon (mask or fabrication errors). (Hawkins et al., 1994)

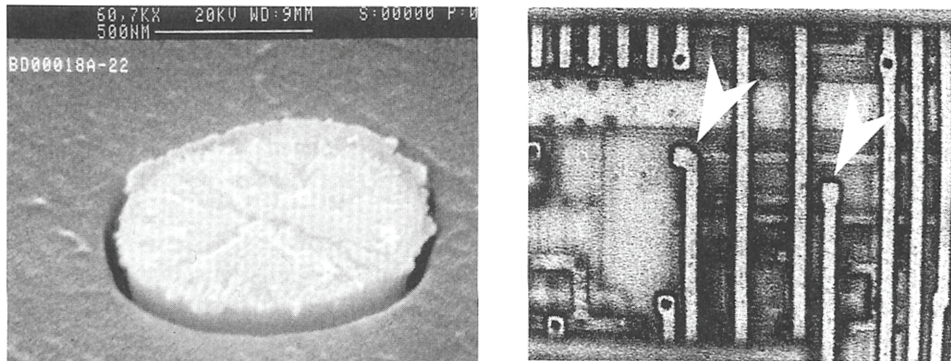

Figure 4-7 Resistive Open and missing Vias (Segura and Hawkins, 2004)

As transistors decreases in size, behavior modeling of open defects becomes more and more complex. A modern IC contains form millions to several billions transistors, and a total metal length of several kilometers. As deep-submicron CMOS technology comes into use with metal line width of 130 µm or less and via height-to-width ratios of more than 5:1 open defects are more or less inevitable. (Ohletz, 1996b)

The main effect of an open IC signal line is that one circuit node is no longer driven by any gate, but might be left in a floating or high impedance state. The floating node does not have a direct path to V_{dd} or ground and the voltage on the floating node depends on the properties and topology of the surrounding circuitry. The size of the crack or open determines if electrons can tunnel across the open, thereby controlling the amount of charge injected from the original driver toward the floating node, in addition the charge of the floating node depend on the capacitive coupling to the surrounding nodes and the charge at the gate and drain terminals of the connected transistors (Segura and Hawkins, 2004).

4.4 Parametric Failures

Parametric failures are failures due to variations in one or a set of circuit parameters so that the circuits' performance falls outside its specifications. Parametric failures have always been present in integrated circuits, but their impact on circuit performance grows spectacularly in deep-submicron technologies. Parametric failures shows in two general forms: one failure form is caused by defect-free (intrinsic) parameter shifts, the second failure form affects functionality through environmentally sensitive defects on the die (extrinsic). This failure class is called parametric as the intrinsic or extrinsic failures are either caused by variation of the IC process parameters, or due to sensitivity to environmental parameters such as power supply, temperature, clock frequency and/or radiation.

Table 4-1 Parametric Failure Sources and Effects (Segura and Hawkins, 2004)

	Failure mechanism	Physical Effect
Intrinsic	Process Fabrication Parameter Variation	Threshold Voltage Shift ILD variation Interconnect Ω shift, metal width, spacing, thickness, granularity L_{eff} shift W_{eff} shift nMOS-to-pMOS length ratio shift Diffusion resistance
Extrinsic or intrinsic	Metal: Via-interconnect defect; Electromigration Stress Void Oxide: Defects or wear out; Hot Carriers	Resistive metal Gate oxide short Hot Carrier injection

Table 4-1 lists some of the different forms of intrinsic and extrinsic parametric failure mechanisms and the physical effects that may significantly alter a circuit's performance. Parametric failures are typically insensitive to many test methods such as I_{DDQ}, stuck-at, delay and logic functional tests. Most parametric failures are speed, current and amplitude related and subsequently not conditions that causes DC parametric failures.

4.4.1 Intrinsic Parametric Failures

Two factors cause intrinsic parameter variation, environmental and physical. Environmental factors include variation of the power supply levels within the die or on the board, or switching activity and temperature variations across the circuit. Physical variation comes from the inherent weakness in circuit manufacturing that allow transistor and interconnect structural variations. These deviations from the targeted values are limitations or imperfections in process and mask steps.

4.4.1.1 Transistor Parameter Variation

Individual transistor parameters vary within a die, die-to-die, and lot-to-lot, making transistor speed and drive strength difficult to predict. The main parameters that determine the transistor drive properties are: (Segura and Hawkins, 2004)

- Channel length variation
- Channel width variation
- nMOS-to-pMOS length ratio variation
- Effective gate oxide thickness variation
- Doping variation- threshold voltage and diffusion resistance

These parameters directly modulate the drive and behavior of the device. Both drain current, threshold voltage and signal amplitude are interrelated and a function of these, and other, parameters.

Channel Length Variation: As seen equation **(4-1)** for a MOS in active region, channel length lies in the denominator and as L_{eff} gets smaller, drain current raises, and by so doing alters the characteristics of the device, and the speed characteristics of the circuits. Channel length variations are due to a combination of photolithography, gate etching, ion-implant, spacer formation, and thermal processing effects. It has been showed that L_{eff} variation has the greatest affect on IC performance compared to other process parameters (Segura and Hawkins, 2004).

$$I_D = \frac{\mu_n C_{ox}}{2}\frac{W}{L}\left(V_{GS} - V_{tn}\right)^2\left[1 + \lambda\left(V_{DS} - V_{EFF}\right)\right] \quad \textbf{(4-1)}$$

Channel Width Variations: As with length variations, width variations affect the drain current, but another important effect is the narrow-channel which rises in minimum width devices. The main narrow-channel effect is that the transistor effective threshold voltage varies across the transistor width, changing from the nominal value at the center region to an altered value toward the device edge. The narrow-width effect is typical seen for channel widths less than 0.4 µm. (Segura and Hawkins, 2004)

***nMOS* to *pMOS* length ratio variation:** this variation is important in ICs that use ratioed logic styles. The width-to-length aspect ratio between *nMOS* and *pMOS* transistor determines the noise margin and the switching point of the logic gates. The channel length variations attributed to the two types of devices may come from implant dose, energy and diffusion tolerances of the dopants associated with the different MOSFET types.

Effective Gate Oxide Thickness Variation: oxide thickness has a first-order impact on device performance, directly affecting transconductance, threshold voltage, and device drive current. Gate oxide physical thickness variance is related to the tolerance of the thermal growing process. In modern processes this variation is kept below 0.5% of the effective oxide thickness. (Hawkins et al., 2003)

Random Doping Fluctuations: Doping variations may differ for devices in the same die and are due to variations in the implant dose, angle and energy. These change the junction dept and doping profiles impacting the effective channel length and threshold voltage. Another performance noise source that impacts the threshold voltage is related to the distributing of dopant atoms. A variation in doping density beneath the gate makes local threshold voltage uneven under the gate.

Intrinsic variation impact on transistor properties
The parameters listed over cause fluctuations of drain saturation current, threshold voltage or both in devices from the same circuit. The transistor effective length and threshold voltages are the main parameters that determine current drive and therefore the circuits’ maximum clock speed.

4.4.2 Extrinsic Parametric Failures

There are five major extrinsic IC mechanisms associated with parametric failures:
- Weak interconnect opens
- Resistive vias and contacts
- Metal mousebites
- Metal slivers
- Gate oxide shorts

Weak opens and resistive vias are major defect-related parametric failure mechanisms. Mousebites occur when sections of metal are missing from an interconnect line. Slivers are common defects in which a metal particle lays between two metal conductors and barely contacts the signal lines. Gate shorts may show timing and power-supply-dependent failures. (Segura et al., 2002)

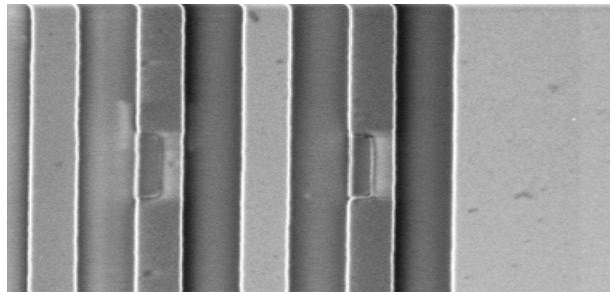

Figure 4-8 Mousebites

Weak interconnects opens: In section 4.3 open defects were discussed, weak opens are included here since they cause a relatively small increase in interconnect resistance, and does not prevent current through the line. Depending of the location and the resistance of the defect the impact will vary from none, through decreased speed/amplitude to a hard circuit defect.

Resistive Vias and Contacts: connecting the different metal layers, and physical parts of each transistor are contacts and vias. Several fabrication-related mechanisms can cause resistive failures, and although this is a different failure mechanisms that cause resistive failures the way they fail is similar to that of weak interconnect opens.

Metal Mousebites: missing regions of interconnect metal are called mousebites. These can be caused by particle defects, electromigration or stress voids. Mousebites have a minor electrical effect, but are a major reliability risk, and may cause electromigration, or local hotspots that cause interconnects to fail.

Metal Slivers: a small metal slivers that lie between two interconnect lines, barely or not even touching them, when the temperature rises, the metal expand, and the sliver will now touch the signal lines, causing bridges and circuit failures.

4.5 Capacitor failure mechanisms

Capacitors are much like transistors susceptible to faults such as bridges, shorts and opens. The large use of unit size capacitors reduce the effect of capacitance size deviations thus cause the most common fault in on-die capacitors to be oxide breakdown caused by a rupture in the thin oxide layer between the capacitors top and bottom plate. Oxide rupture causes a short and a represents itself as a short between the two input nodes, the fault simulation model will then be a resistor in parallel with the capacitor.

5 ADC Specifications, Terminology and Testing

Before studying different test architectures and how they are implemented on-chip it is important to have a clear perception of key specifications, the terminology used and how different tests reveals different non-idealities.
ADC testing can roughly be divided into two separate parts, static and dynamic testing. In static testing the converter under test is subjected to a series of dc voltage levels and the output values are monitored to determine the converters accuracy. One major limitation in static testing is that nonlinearities related to the input signal bandwidth may pass undetected. Dynamic testing is when the converter is stimulated by periodic waveforms instead of dc levels. This type of testing is usually better suited for production testing due to easier signal generation and less time consuming behavior. The signal bandwidth can be higher than in static testing and thereby resembling the actual applications signal. However one disadvantage is that dynamic tests are usually not deterministic in nature, the analog input is not compared to the resulting digital code, instead the converter transfer function is interpreted from the resulting out data. To ensure sufficient resolution under test the signal source needs a resolution at least 3-bit greater than device under test. (IEEE_Std_1241, 2001)

The following sections are based on the IEEE standards (IEEE_Std_1057, 1994) and (IEEE_Std_1241, 2001)

5.1 Static Specifications and Testing

5.1.1 Offset and Gain

Offset and gain are endpoint errors. They characterize the deviation from the ideal transition level for the first and last code.

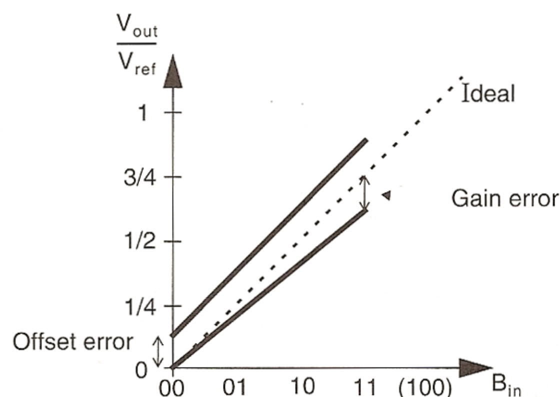

Figure 5-1 Offset and Gain errors (Johns and Martin, 1997)

Offset error E_{off} is defined as the deviation from $V_{0..01}$ from ½ LSB.

$$E_{off} = \frac{V_{0...01}}{V_{LSB}} - \frac{1}{2}LSB \tag{5-1}$$

Where V_{LSB} is the ideal width of a code bin, *LSB* is the step size and *N* is the number of digitized bits.

An effective method of estimation offset is to start at the lowest input voltage possible and raise the voltage slowly until the transition level has been found.

Gain error E_{gain} is defined as the deviation from $V_{1\ldots11}$ corrected for E_{off}.

$$E_{gain} = \left(\frac{V_{1\ldots11}}{V_{LSB}} - \frac{V_{0\ldots01}}{V_{LSB}} \right) - \left(2^N - 2 \right) \quad (5\text{-}2)$$

An effective method of finding the transition level is to start at the highest input voltage possible and lower the voltage slowly until the transition level has been found.

5.1.2 Integrated nonlinearity error

Integrated nonlinearity error is the difference between the ideal and the measured code transition levels after both gain and offset errors have been removed.

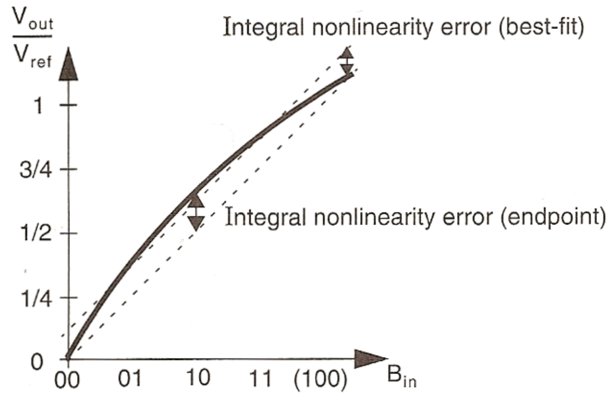

Figure 5-2 INL error (Johns and Martin, 1997)

INL can be expressed in either a percentage of full scale or in unit of LSB. If INL is given as a single number without code bin specification it specifies the maximum INL over the entire code range. INL can be calculated using:

$$INL(k) = \frac{T(k) - V_{LSB}(k-1) - T_{ideal}}{V_{LSB} 2^N} \quad (5\text{-}3)$$

Where *T(k)* is the code transition level for the *k*th transition, V_{LSB} is the ideal width of a code bin, T_{ideal} is the ideal input value corresponding to the code transition level and *N* is the number of digitized bits.

5.1.3 Differential nonlinearity error

Differential Nonlinearity error is the difference between a specified code bin width and the ideal code bin width divided by the ideal code bin width e.g. the variation in analog step sizes away from 1 LSB. When this is given as a single number without code bin specification it is the maximum DNL of the entire range. DNL can be calculated using:

$$DNL(k) = \frac{W(k) - V_{LSB}}{V_{LSB}} \tag{5-4}$$

Where $W(k)$ is the width of code bin k. Note that neither $W(o)$ nor $W(2^N-1)$ is defined.

5.1.4 Histogram Testing

Histogram testing is based on the fact that any periodic waveform has a predictable percentage of time in each code bin, after sampling a periodic signal a statistical large number of periods the occurrence of each code can be compared to the amount it should have if the ADC was ideal and the device non-linearity can be calculated.

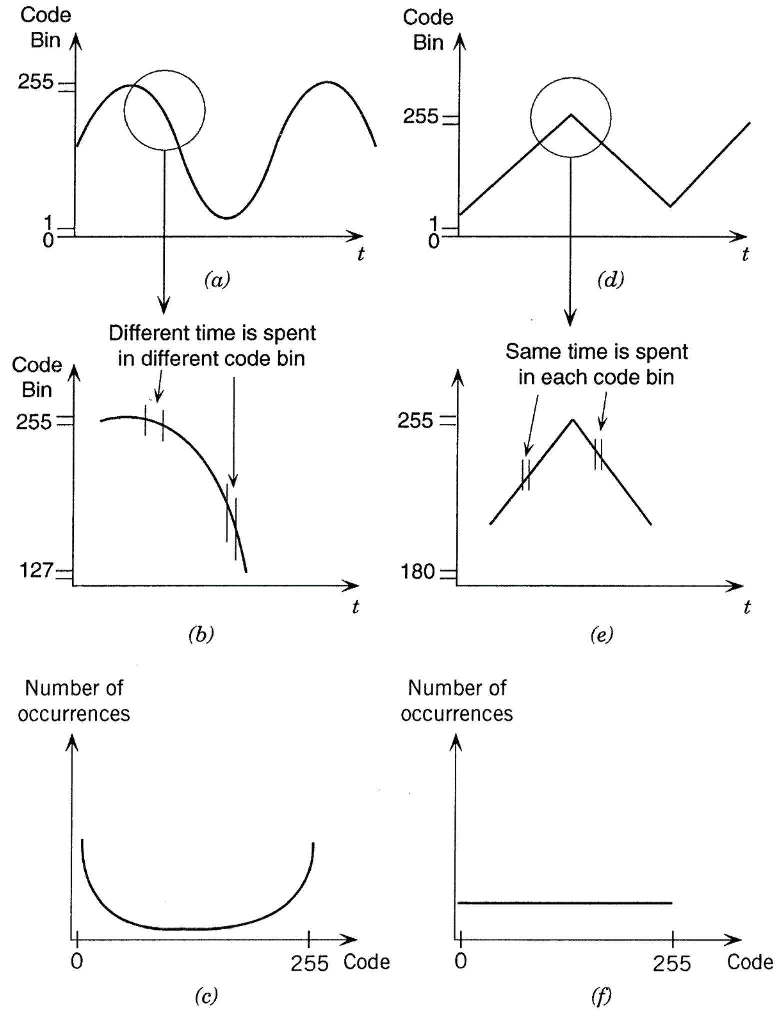

Figure 5-3 Histogram testing (Hoeschele, 1994)

As shown in Figure 5-3 both sine- and triangular-waveforms can be used in histogram testing although using a sine-wave will cause a predictable skew in the number of occurrences due to the fact that a sine-wave changes value faster when it is at its midpoint than it does when it is close to its extreme values.
Two important consideration when using a histogram test is that an integer number of periods is not used the number of occurrences will be skewed, and the input amplitude has to be exactly controlled. If the input exceeds the converters range the zero and full scale occurrence will be excessively high and if the input range is to low the zero and full scale values will never appear.

5.1.5 Coherent Sampling

Coherent sampling is when an integer number of cycles of the test signal are sampled
In other words coherent sampling occurs when the following equation is fulfilled:

$$f_s \cdot M_c = f_0 \cdot M \qquad (5\text{-}5)$$

Where M_c is the integer number of cycles in the test signal, and M is the number of samples in the output.

5.1.6 Missing codes

An ADC is guarantied to have no missing codes if the maximum INL is less than 0.5 LSB or maximum DNL is less than 1 LSB.

5.2 Dynamic specifications

5.2.1 Signal to Noise Ratio

Signal to noise ratio is the ratio of the signal root-mean-square (*RMS*) value of the fundamental frequency to the *RMS* of all other frequencies excluding harmonics.

$$SNR = \frac{Signal_{RMS}}{Noise_{RMS}} \quad (5\text{-}6)$$

Note that depending of the context and application it is often used in different ways and therefore is ambiguous, as an example (IEEE_Std_1057, 1994) uses the same definition of SNR as (IEEE_Std_1241, 2001) does for SINAD,

5.2.2 Signal to Noise and Distortion Ratio

SINAD is the ratio of the *RMS* value of the signal to the *RMS* value of all other frequencies including harmonics. Section 4.5.1.1 in (IEEE_Std_1241, 2001) claims that SINAD test results are affected by almost all error sources, with the exception of dc offset and gain accuracy thus making SINAD testing well suited for production testing.
SINAD is defined as:

$$SINAD = \frac{Signal_{RMS}}{Noise_{RMS} + Distortion_{RMS}} \quad (5\text{-}7)$$

A SINAD test can be executed in the time or frequency domain. In the frequency domain SINAD can be determined from conducting a FFT of the digital out signal. In the time domain it is possible to calculate SINAD by performing a sine wave fitting to calculate the noise *RMS* value.

5.2.3 Sine-Wave Fitting

Sine-wave fitting or curve fitting is finding a wave form which matches a series of data points, in this case the output from an ADC when a sine wave is introduced at the input. By calculating the phase, amplitude, DC-offset and frequency of the data points and using these it is possible to generate a reference sine-wave that minimizes the squared difference between the calculated sine-wave and that observed in the data points.

$$\Pi = d_1^2 + d_2^2 + \ldots + d_n^2 = \sum_{i=1}^{m} d_i^2 = \sum_{i=1}^{m} \left[y_i - f(x_i) \right]^2 = small \qquad \textbf{(5-8)}$$

Equation **(5-14)** describes the general problem where *Π* is the square sum of difference (residue) , d_i is the *i'*th coefficient used to minimize the difference, y_i is the *i'*th data point and $f(x_i)$ is the *i'*th estimated data point. There is no exact solution to finding the least square difference *Π*, but algorithms such as the Gauss-Newton algorithm is developed to minimize the least square difference. In the special case where the output is a sine wave this problem can be expressed in the following form:

$$\Pi = \sum_{i=1}^{m} \left[y_i - A_0 \cos(\omega_0 t_i) - B_0 \sin(\omega_0 t_i) - C_0 \right]^2 \qquad \textbf{(5-9)}$$

By calculating the coefficients A_0, B_0, C_0 and ω_0 it is possible minimize the least sum of squared difference.
Performing a Sine-wave fit requires the estimation of four parameters (three if frequency is known) which can be found by performing complex matrix operations or large summating algorithms, this is complicated and time consuming to execute in a microcontroller. Chapter 4.1.4 in (IEEE_Std_1241, 2001) gives different algorithms for estimating the coefficients and is highly recommended as further studies in this topic.

When best fit coefficients have been found C_0 represents DC-offset and the signal amplitude A_{signal} and phase θ can be found:

$$A_{signal} = \sqrt{A_0^2 + B_0^2} \qquad \textbf{(5-10)}$$

$$\theta = \tan^{-1}\left(-\frac{B_0}{A_0} \right) \quad \text{if } A_0 \geq 0$$
$$\theta = \tan^{-1}\left(-\frac{B_0}{A_0} \right) + \pi \quad \text{if } A_0 < 0 \qquad \textbf{(5-11)}$$

The residuals r_i of the fit will be given by the following equation:

$$r_i = y_i - A_0 \cos(\omega_0 t_i) - B_0 \sin(\omega_0 t_i) - C_0 \qquad \textbf{(5-12)}$$

And the *RMS* error will be given by:

$$e_{RMS} = \sqrt{\frac{1}{M} \sum_{i=1}^{M} r_i^2} \qquad \textbf{(5-13)}$$

When the total *RMS* error in the data point has been established it is easy to calculate SINAD:

$$SINAD = \frac{Signal_{RMS}}{Error_{RMS}} = \frac{\frac{A_{signal}}{\sqrt{2}}}{e_{RMS}} = \frac{\sqrt{2} \cdot A_{signal}}{2 \cdot e_{RMS}} \qquad \textbf{(5-14)}$$

5.2.3.1 Test Length, Amplitude and Frequency

When conducting a dynamic SINAD test a large input signal is required to ensure that the ADC is tested over its complete range without overloading the ADC. By choosing the signals amplitude to be at least 90% of the full-scale range this can be achieved.
By choosing an input frequency relatively prime to the sampling frequency the samples are spread in the sine-waves phase over 0 to 2π.
In a dynamic test the number of cycles a test has to be performed will vary depending on the amount of noise present in the modulator. However section 4.1.3.5 in (IEEE_Std_1057, 1994) states that as a general rule of thumb a test length larger than five cycles has to be used to ensure a sufficient accuracy.

5.2.4 Effective Number Of Bits

Given that the SNR of an ideal ADC without any noise except quantization noise was shown in section 2.3.4 to be:

$$SNR_{MAX} = 6.02N + 1.76 \quad \textbf{(5-15)}$$

Where N is number of bit, it is possible to calculate ENOB directly out of this to:

$$ENOB = \frac{SNR - 1.76}{6.02} \quad \textbf{(5-16)}$$

6 BIST Implementation Techniques

Testing high resolution ΔΣ-modulators is a costly task due to the high resolution and large number of samples required. To successfully test an N-bit ADC the stimuli need a resolution greater than N+2 bit (IEEE_Std_1241, 2001). Due to the ΔΣ-modulators oversampling nature it does not have a direct input to output relationship, making characterization and testing even more challenging.

A traditional setup required for testing high resolution ΔΣ-modulators consist of a high resolution external signal generator that stimulates the modulator under test.

As mentioned in section 5 the stimuli depend on the test performed. If the tester performs static testing, to find offset and linearity errors, input is usually a set of DC values or a saw tooth waveform. To estimate dynamic performance such as SNR or SINAD a sine waveform is used as stimulus. The digital response is loaded from the MUT and into a signal analyzer that evaluates performance.

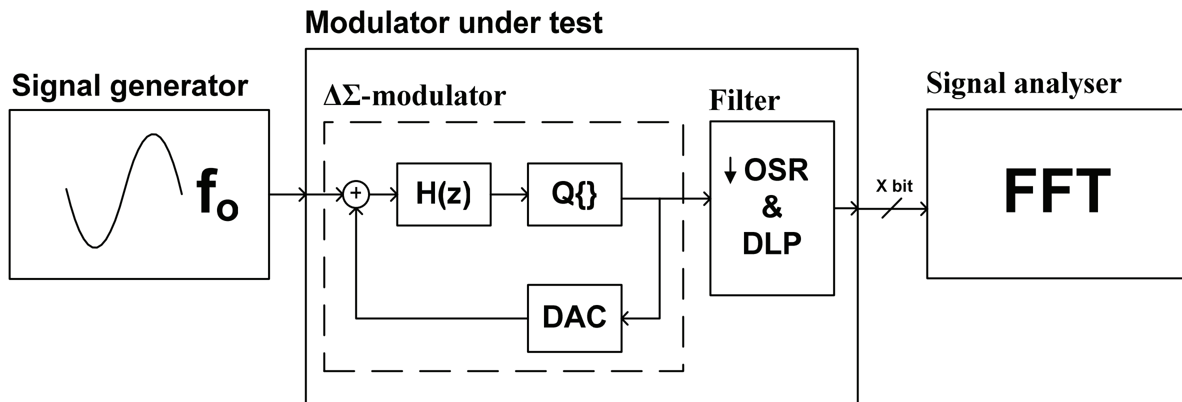

Figure 6-1 Traditional test

Using this setup the chip/system is usually prevented from running any other testing in parallel since shifting data into the chip can contribute to noise on the ADC input pin. This leads to long and ineffective test time that could be better utilized.

By implementing an on-chip stimulus the likelihood for digital noise affecting the input signal will be reduced since the analog input pin can be removed. However other challenges then appear. On-chip signal generation is susceptible to supply voltage drift and substrate noise and the signal linearity and resolution can not be guarantied since production can alter or degrade the signal generator.

Generating stable high resolution signal generators on-chip is quite possible but usually this signal generation requires large area, and some sort of on-chip calibration to guaranty signal quality, thus leading to even greater area consumption.

Response analysis can be performed in the digital domain by implementing a Digital Signal Processor (DSP) core or using existing architecture. However if the ΔΣ-ADC is implemented as a stand-alone device the area required to implement a DSP core can be too large to be cost-effective.

6.1 Existing BIST Proposals

Several systems have been developed for testing oversampling converters. These can be divided into static and dynamic test structures.

6.1.1 Integrator Created Ramp Stimulus

One frequently suggested method is using high precision integrators to develop an analog ramp or a saw tooth waveform and analyzing the digital output to find linearity errors and offset. (Yun-Che and Kuen-Jong, 2000) proposes the structure shown in Figure 6-2 where a differential integrator creates a slow stepped ramp stimulus and a counter keep track of the time passed. When a change in digital output is detected the INL and DNL detectors store time and the digital value.

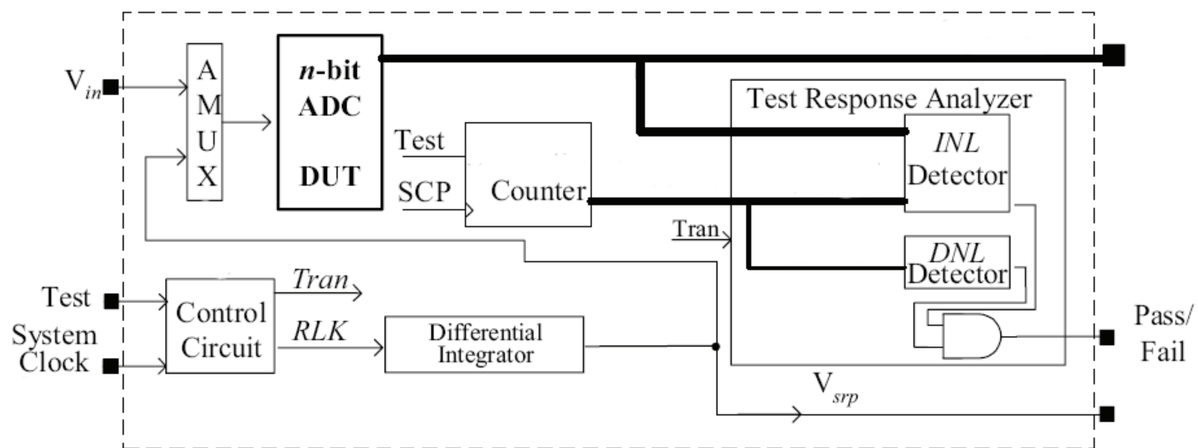

Figure 6-2 Static test with ramp generation (Yun-Che and Kuen-Jong, 2000)

After the test the INL and DNL detectors calculates INL and DNL and decides if data is acceptable.
This is a relative simple way to test ADC performance however it has some design challenges. Due to integrator leakage and production variances the integrator requires a large area and/or a complex calibration circuit to ensure linearity and accuracy. In addition the slow changing stimuli can mask out and hide dynamic nonlinearities, thereby allowing flawed devices to pass the test.

6.1.2 ΔΣ-DAC Created Stimulus

Another static test scheme is proposed in (Wang et al., 2005) where a on-chip analog generator creates a high accuracy triangular waveform.

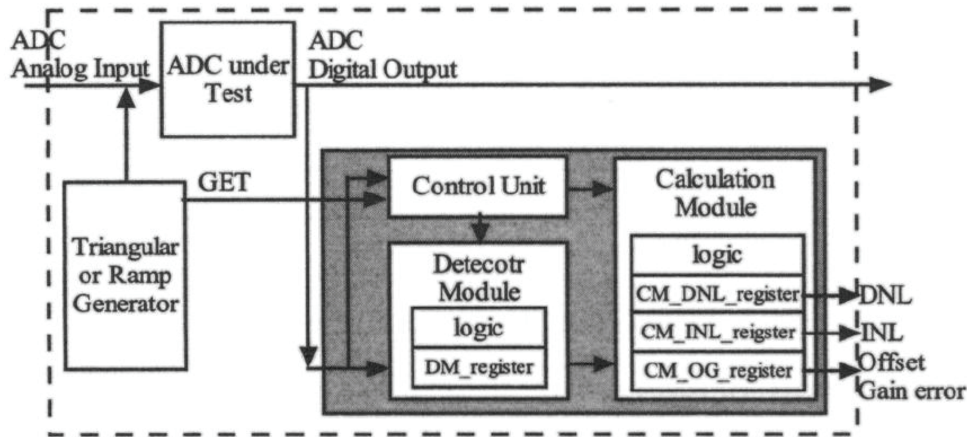

Figure 6-3 Static test with triangular signal generation (Wang et al., 2005)

The test setup consists of a control unit that for calculate DNL, INL, offset and gain errors at the same time as it controls the signal generator. In an attempt to circumvent the request for a high precision integrator the signal generator is implemented with a ΔΣ-DAC that converts a digital ramp signal into an analog stimulus.

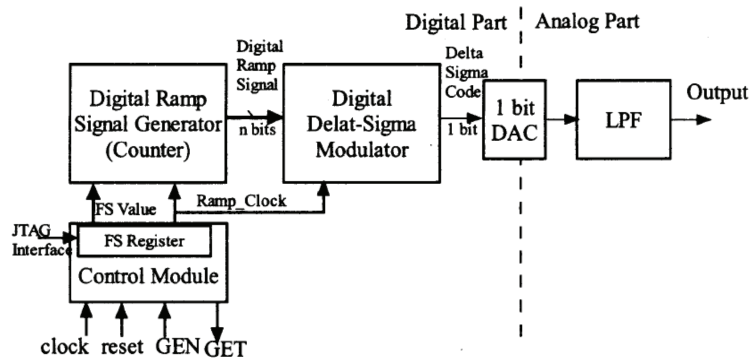

Figure 6-4 ΔΣ-DAC stimulus generator (Wang et al., 2005)

This design uses the ΔΣ-modulators noise-shaping properties to achieve a high resolution and linearity, a one-bit DAC to convert the digital ΔΣ-modulator output into an analog signal and a Low pass filter to remove the high frequency quantization noise.
Although this design bypasses the troublesome integrator it has the same problems detecting dynamic nonlinearities in the MUT.
The proposed BIST requires a large area due to the implementation of a complete ΔΣ-DAC and a control module, if the target chip already contains a DAC the increase in area would be fairly low, otherwise the area required to implement this BIST would be large and therefore lead to an inappropriate increase in production cost.

6.1.3 ΔΣ-Modulator As a Resonator

In (Kuen-Jong et al., 2003) a second order ΔΣ-modulator, modified to work as a digital resonator, is used as stimulus. By setting coefficients the frequency, phase and amplitude of the output sine wave can be altered to fit different test scenarios.

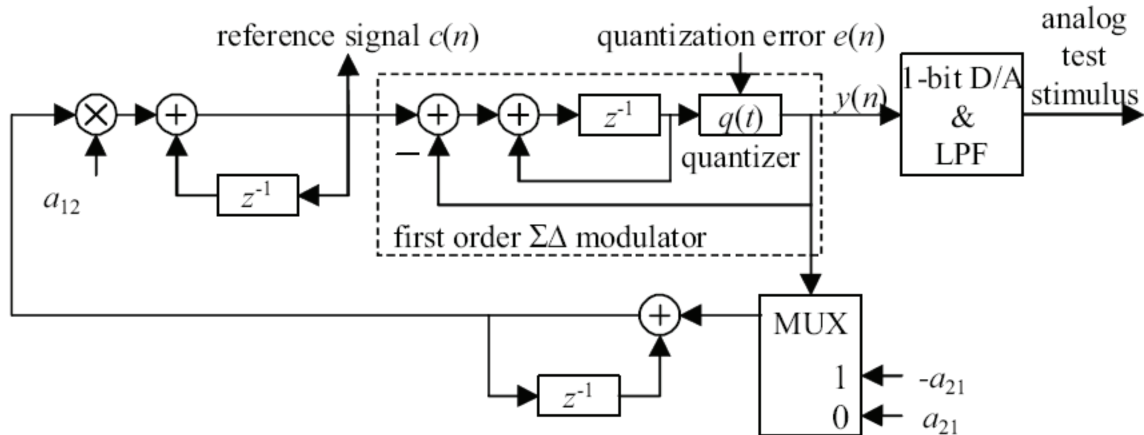

Figure 6-5 ΔΣ-modulator as resonator (Kuen-Jong et al., 2003)

This test setup allows static testing using a low frequency sine wave to estimate INL and DNL using histogram testing. By setting different coefficients a high frequency sine wave can be created and used in dynamic testing. The signal *c(n)* can be used as an reference to perform a sine-wave fitting and by doing so measuring SNR and SINAD.
As in section 6.1.2 this test method uses a ΔΣ-modulator to create an analog stimulus, but in this proposal the digital signal is created on-chip and there is no need to shift digital data into the chip. Nevertheless this method has the same problem as the one proposed in section 6.1.2 by requiring a large extra area.

6.1.4 Periodic Recorded Stimulus

(Dufort and Roberts, 1997) and (Dufort and Roberts, 1999) propose a test where the ΔΣ-modulator is replaced by a circular shift register. Stimulus is generated by simulating an ideal ΔΣ-modulator and recording its binary output stream. The output is shifted into the register where it serves as stimulus to a one-bit DAC and a low-pass filter to generate an analog signal.

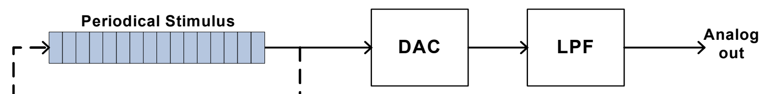

Figure 6-6 Stimulus generated from a periodic stream

This setup replaces the on-chip ΔΣ-modulator with a shift register and by doing so decreases test circuit complexity and area. Generation and quality of this binary stream will be discussed in section 8.2. However as in section 6.1.3 this setup requires a DAC and an analog low pass filter that either has to be tested to ensure operation or presumed to be functional and by doing so introducing uncertainty into the test.

6.1.5 Software ΔΣ-Modulator

In (Chee-Kian et al., 2004) a self test using a software ΔΣ-modulator is used to stimulate the modulator under test.

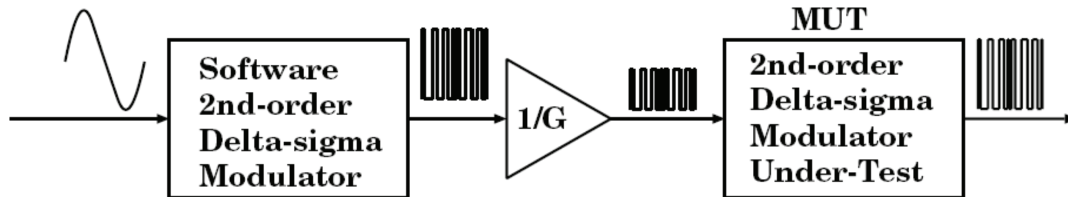

Figure 6-7 Stimulus generated by a software ΔΣ-modulator (Chee-Kian et al., 2004)

As shown in Figure 6-7 the proposed test uses a digital sine wave as input into an on-chip software ΔΣ-modulator to generate a binary pulse-train. The binary stream is the converted in a combined DAC and attenuator to a scaled digital signal and used to stimulate the modulator under test. Theory and background of the attenuation and its effect on the signal is discussed in section 7.1.1. Using this scheme it is possible to perform both static and dynamic testing however generating input wave form to the software ΔΣ-modulator and running this modulator with the desired accuracy requires a fast on-chip CPU/DSP, thus preventing this setup to be cost effective.

6.1.6 Recorded Binary Stimulus

(Rolindez et al., 2006) further develop the method suggested in (Chee-Kian et al., 2004) by removing the software ΔΣ-modulator and replacing it with the recorded binary stream suggested in (Dufort and Roberts, 1997) and (Dufort and Roberts, 1999) By using the stream as a reference one can perform a sine-wave fitting to evaluate the MUTs performance. To generate the reference some modifications on the digital filter has to be made, this is further discussed in section 8.4

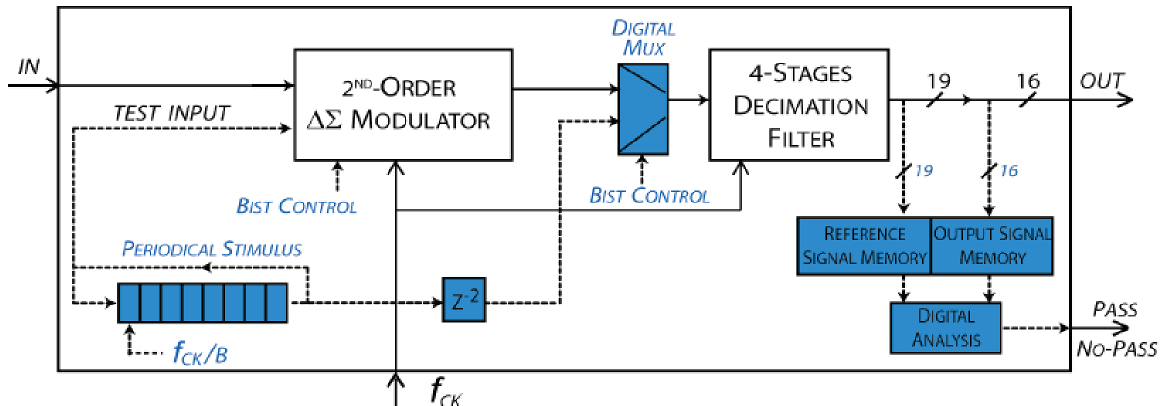

Figure 6-8 Stimulus generated by recorded binary stream (Rolindez et al., 2006)

By implementing a stored binary string and removing the analog filter used in previous suggestions, this setup is better suited for on-chip implementation due to the elimination of an analog signal generation which need to be tested to ensure signal resolution and linearity. This method has some drawbacks, the suggested shift-register length is 1126 bit long, the proposed digital analysis is hardwired and can not be changed after implementation and there is no static testing to find offset and gain errors that is not detectable in dynamic SNR/SINAD testing.

7 Binary Signals as Stimuli

A pre-study was conducted by the author where the BIST setup proposed in section 8.1 (with some modifications) was simulated in Matlab (MATLAB, 2006) and showed great promise towards implementation (see appendix C).

However to validate if one by using binary stimuli can detect defects and parametric faults in a physical circuit a test bench using a SPICE model of the delta sigma converter used in Atmel's battery monitoring circuits and described in appendix B was developed.

By changing individual transistor parameters in the SPICE model it is possible to estimate if using binary signals as stimuli gives sufficient fault coverage. The single fault hypothesis is used, this hypothesis assumes that that one and only one fault condition exists in the circuit.

The main scope in this test is to ensure that soft faults are detected and to what extent they are discovered when a binary stimulus is used as test input. In order to restrict the simulation scope some limitations were placed on the kind of faults simulated. One or more of the following tests are conducted on chosen transistors:

- L_{eff} change by –20%
- L_{eff} change by +20%
- W_{eff} change by –20%
- Gate-Source short, Short resistance = 2 kΩ
- Gate-Drain Short, Short resistance = 2 kΩ
- Stuck open
- Stuck close

In *pMOS* transistors a diode in series with the shorting resistance were used, as described in section 4.2.2 and shown in Figure 4-6.

Capacitors were simulated with an oxide short between the nodes with a resistance of 5 kΩ.

7.1 Test bench and Test Setup

The test bench is written in VHDL-AMS, and use the ΔΣ-ADCs SPICE implementation to achieve as accurate simulations as possible. All simulations are conducted in ADVance MS (Mentor_Graphics_Corporation, 2006).

7.1.1 Signal Generation

To create a binary stream as mentioned in section 6.1.4 a simulation model of a ΔΣ-modulator has to be stimulated by an appropriate stimulus and the output recorded. This method was first proposed in (Hawrysh and Roberts, 1996).

Replacing the high-resolution analog input stimulus with a binary stream degrades the signal by embedding it in shaped quantization noise. Usually the binary stream is converted to an analog signal in a one-bit DAC and a low-pass filter removes the out-of-band quantization noise. However in a ΔΣ-ADC the high frequency quantization noise will not affect the analog modulators functionality and the decimation filter filters out most of the in-band-noise. This property makes it possible to use the unfiltered DAC output as a direct stimulus. On the other hand, the fast rising and falling edges may cause saturation in the integrators. To reduce the risk of integrator saturation the unfiltered signal must be attenuated. The attenuation required depends on the input signal swing and the modulator bandwidth.(Chee-Kian and Kwang-Ting, 2002)

Stimulus is generated with the Delta Sigma toolbox, by (Schreier, 2004), using an ideal second order delta sigma modulator. The binary response is stored to an ASCII file and used in the test bench. With an oversampling ratio of 512 the stimulus was measured to have a SINAD of:

$$SINAD_{stimulus} = 129.9dB$$

which is very close to what equation **(7-1)** stipulates as maximum achievable in an second order delta sigma modulator:

$$SINAD_{MAX} = 6.02 \cdot 1bit + 1.76 - 12.9 + 50 \cdot \log(512) = 130.343dB \qquad \textbf{(7-1)}$$

And more than sufficient for the modulator under tests maximum of:

$$SINAD_{Modulator} = 6.02 \cdot 13bit + 1.76dB = 80.02dB \qquad \textbf{(7-2)}$$

The stimulus spectrum is shown in Figure 7-1.

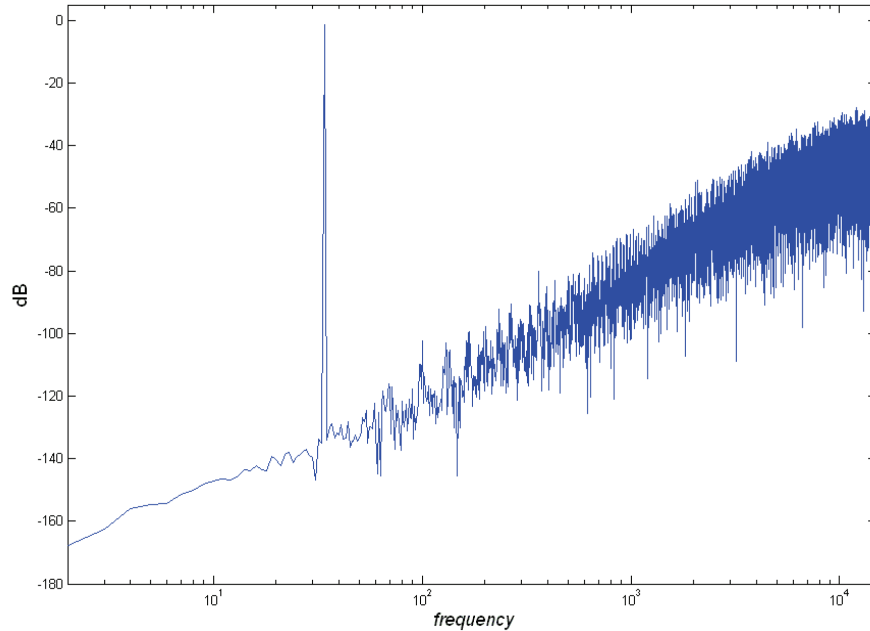

Figure 7-1 Stimulus Spectrum

.

The MUTs input bandwidth of 32 Hz and sampling frequency of 32.768 kHz, dictates that a low test frequency has to be used. To validate if using an unfiltered stream as test stimulus is able to reveal soft and hard faults a signal with an input frequency f_0 of:

$$f_0 = 9\pi \approx 28.27 Hz$$

was selected and generated. This frequency was chosen due to relatively prime relation with the sampling frequency. To ensure that the test was carried out in accordance to the guidelines given in section 5.2.3.1 a stimulus length of 13107 samples were chosen, allowing the test signal to describe six complete periods, one to ensure the internal state of the filter and five used to calculate the modulators performance.

7.1.2 Test bench

The test bench uses the existing SPICE implementation of the modulator and links it together with the existing decimation filter and the ideal DAC used to generate test stimulus on the top level as shown in Figure 7-2

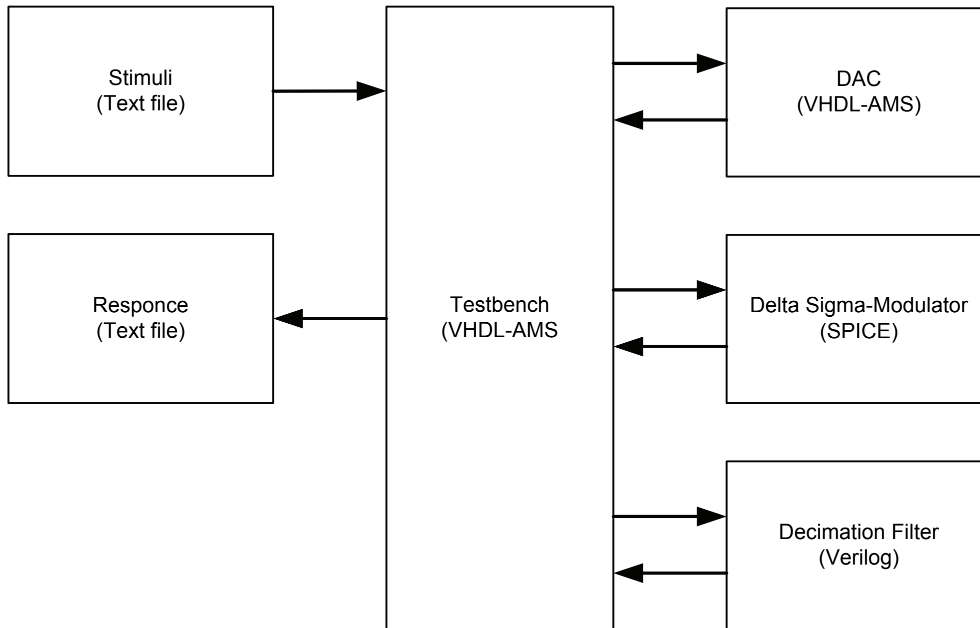

Figure 7-2 Test Bench, Binary stimulus

The modulator is stimulated by an ideal DAC, the modulators output bit-stream is down sampled and filtered in the decimation filter and the result is stored in a text file for analysis in GNU Octave (Eaton, 2004).

Table 7-1 Key data ΔΣ-ADC

	Name		Unit
Digital Supply Voltage	DVDD	3.3	Volt
Analog Supply Voltage	AVDD	3.3	Volt
Sampling frequency	fclk	32.768	kHz
Signal Bandwidth		32	Hz
Signal range		± 220	mV
Filtered output 1	CADIC	13	bit
output 1 frequency		256	Hz
Filtered output 2	CADAC	18	bit
output 2 frequency		1-8	Hz

ENOB, SINAD, Phase, Amplitude and Offset are calculated in octave using a four parameter fit algorithm. The results are also compared to those of the fault free modulator. Each test runs two simulations:

1. Test stimulus
2. Analog stimulus 440 mV peak-to-peak

Simulations showed that to avoid overloading the modulator when using binary stimulus the amplitude needed to be reduced to ± 40 mV instead of the designed ± 220 mV. (Rolindez et al., 2006) and (Chee-Kian et al., 2004) found that a reduction in input signal by 4 would be sufficient but the special scaling of the modulator under test causes overloading earlier then that of a traditional Boser-Wooley modulator.
To compensate for this attenuation the results from binary testing is adjusted with:

$$SINAD_{LOSS} = 20 \cdot \log\left(\frac{0.220mV}{0.040mV}\right) = 14.8dB$$

Table 7-2 MUT. Analog vs Test Stimulus

	Analog		Binary	
SINAD	65,64	dB	65,78	dB
ENOB	11,03	Bit	11,22	Bit
Phase	-44,96	Deg	-56,77	Deg
Amplitude[1]	0,1879	Volt	0,0259	Volt
Offset	0,0001	Volt	0,0004	Volt

As one can see using the test stimulus produce results that are close to those achieved by using an analog input signal. Selecting the DACs output amplitude as high as possible without overloading the modulators, cause the offset to be slightly higher than when using regular analog stimulus.

[1] Amplitude is converted from binary representation to the corresponding voltage.

7.2 Response and Results

Figure 7-3 shows the modulators analog top level. This is also repeated in a larger version in appendix B.1. To validate the test stimulus ability to detect flaws and parametric deviations imperfections were injected into the circuit.

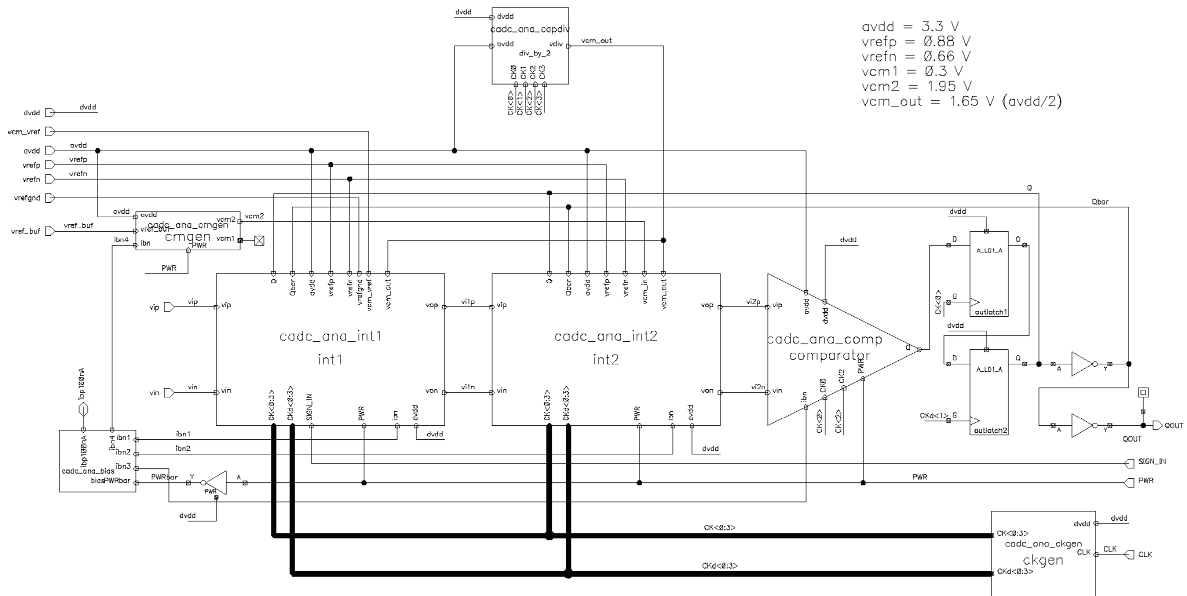

Figure 7-3 Analog Top Level

This modulator is as the figure shows fully differential and the placements of flaws have been limited to one of the two differential branches, however it should be valid for both.

7.2.1 Detectability in integrator 1

Due to the ΔΣ-modulators second integrators insensibility to noise and non-idealities as mentioned in section 3.2,the fact that it uses the same architecture as the first integrator, and to reduce simulation time only the first integrator was subjected to deviations. The integrator consists of an operational amplifier and a SC-network that determines feedback, gain and amplification. The testing of the integrator was therefore divided into two sections.

7.2.1.1 Detectability in the Operational amplifier

Figure 7-4 shows the single-stage folded cascade op-amp used in integrator one.
This is repeated in a larger version in appendix B.1.1

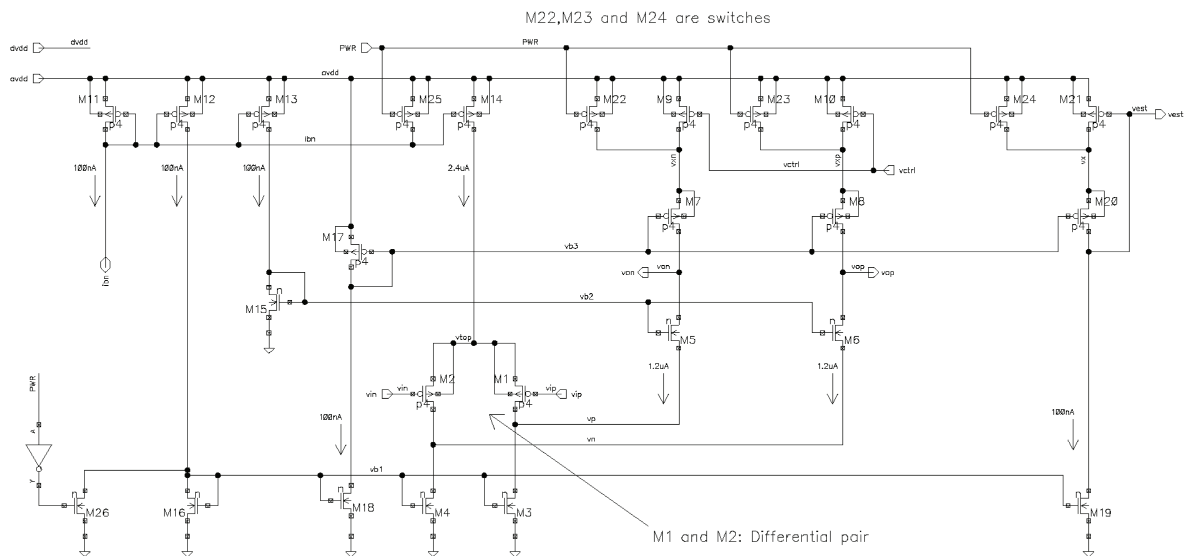

Figure 7-4 Op-amp schematic

To verify the test stimulus suitability deviations were introduced in the following transistors: M2, M6, M8, M14 and M18.

Table 7-3 Performance with deviations introduced in Op-amp 1 transistor M2

Deviation Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
L_{eff} +20%	Test Analog	65,07 63,72	-0,71 -0,92	0,0259 0,1879	0,0 0,0
L_{eff} -20%	Test Analog	66,51 64,62	0,73 -0,02	0,0259 0,1879	0,0 0,0
W_{eff} -20%	Test Analog	66,43 64,74	0,65 0,10	0,0259 0,1879	0,0 0,0
GS-short	Test Analog	10,33 11,67	-55,45 -52,97	0,0001 0,0003	-99,6 -99,8
GD-short	Test Analog	33,00 31,83	-32,78 -32,81	0,0002 0,0004	-99,2 -99,8
Stuck Open	Test Analog	-1,40 -2,32	-67,18 -66,96	0,0000 0,0000	-100,0 -100,0
Stuck Close	Test Analog	-7,02 -2,16	-72,80 -66,80	0,0000 0,0000	-100,0 -100,0

M2 is one of the two transistors that make up the input differential pair, and as the table show, it is surprisingly robust to parametric changes. As expected shorts and opens causes hard faults which is easily detected. However increasing the length of M2, and thereby reducing current, causes an reduction in analog SINAD of 0.92 dB while the test stimuli causes a reduction of only 0.71 dB. Even though the test stimulus shows less reduction than that of the analog stimuli it is clearly detected.

To increase readability only the most noteworthy performance changes for transistor M6, M8, M14 and M18 are shown, complete simulation results can be found in appendix F.1.1.

Table 7-4 Performance with deviations introduced in Op-amp 1 Transistor M6, M8, M14 & M18

Transistor tested	Deviation Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
M2	L_{eff} +20%	Test Analog	65,07 63,72	-0,71 -0,92	0,0259 0,1879	-0,1 -0,0
M6	Stuck Open	Test Analog	63,00 62,12	-2,78 -2,52	0,0231 0,1467	-10,8 -21,9
M8	L_{eff} +20%	Test Analog	66,02 64,88	0,24 0,24	0,0259 0,1879	0,0 0,0
	GD-short	Test Analog	65,20 63,61	-0,58 -1,03	0,0259 0,1879	-1,7 -1,9
	Stuck Open	Test Analog	65,69 64,57	-0,09 -0,07	0,0259 0,1879	0,0 0,0
M14	W_{eff} -20%	Test Analog	65,25 64,37	-0,53 -0,27	0,0259 0,1879	0,0 0,0
	GD-short	Test Analog	21,22 22,09	-44,56 -42,55	0,0062 0,0212	-76,0 -88,7
M18	Stuck Open	Test Analog	66,12 64,95	0,34 0,31	0,0259 0,1879	0,0 0,0

When simulating transistor M6 stuck open the modulators SINAD is reduced with 2.78 and 2.52 dB respectively.

Transistor M8 stuck open causes only a minute reduction in performance. As in transistor M2 performance reduction is higher when using analog stimulus than using the test stimulus. Nevertheless the test stimulus is able to detect the existence of a flaw. The same tendency appears when a short exists between gate and drain.

An interesting property shown in this table is that change in transistor M14s width cause larger performance degradation when using the test signal than that of the analog signal.

When simulating transistor M18 stuck open SINAD increases with 0.34 and 0.31 dB while there is no change in amplitude.

Section 3.2.4 states that errors caused by pole and gain error do not have a large affect modulator performance as long as the dc-gain is sufficient. The simulations performed support this, and as seen in Table 7-4 and appendix F.1.1 using the test stimuli is capable to detect faults in the op-amp.

7.2.1.2 Detectability in The integrator

Figure 7-5 show integrator 1 including the op-amp tested in section 7.2.1.1. The Integrator is of fully differential SC-design. The schematic is repeated in a larger version in appendix B.1.2

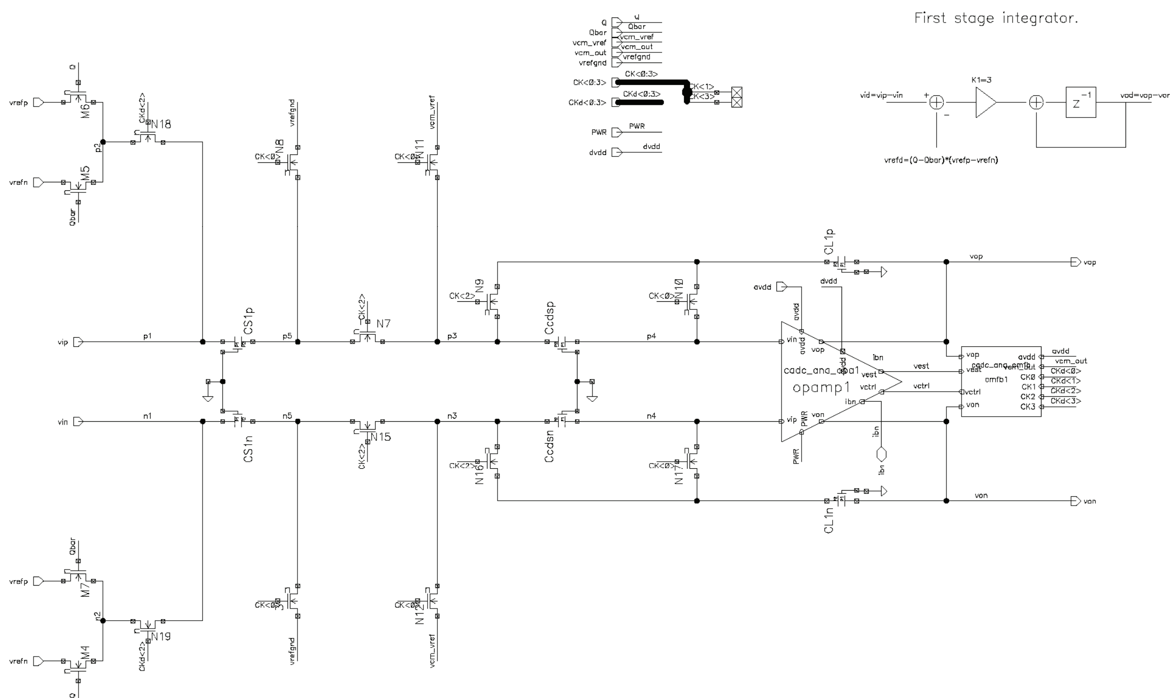

Figure 7-5 Integrator 1 schematic

To verify the test stimulus suitability deviances were introduced in the following transistors:
N7, N9, N10 and N18
And the following capacitors:
CS1p, Ccdsp and CL1p.

The transistors were simulated with the following parametric deviances:

- L_{eff} -20 %
- L_{eff} +20 %
- W_{eff} -20 %
- Gate-Source Short
- Gate-Drain Short

To increase readability only the most noteworthy performance changes are shown, complete simulation results can be found in appendix F.1.2

Table 7-5 Performance with deviations introduced in integrator 1

Device	Deviance Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
CS1p	Short	Test Analog	51,61 46,60	-14,17 -18,04	0,0031 0,0167	-88,03 -91,11
CL1p	Short	Test Analog	3,60 5,28	-62,18 -59,36	0,0103 0,0890	-60,23 -52,63
Ccdsp	Short	Test Analog	65,63 64,27	-0,15 -0,37	0,0259 0,1879	0,00 0,00
N9	L_{eff} +20%	Test Analog	65,15 63,92	-0,63 -0,72	0,0259 0,1879	0,00 0,00
N10	L_{eff} -20%	Test Analog	65,34 64,17	-0,44 -0,47	0,0259 0,1878	0,00 -0,05
N18	L_{eff} +20%	Test Analog	66,22 64,96	0,44 0,32	0,0259 0,1879	0,00 0,00
	W_{eff} -20%	Test Analog	65,30 64,39	-0,48 -0,25	0,0260 0,1879	0,39 0,00

As expected a short in the capacitors cause both amplitude and SINAD degradation. Testing the op-amp showed that SINAD and amplitude resulting from the test signal followed that of the analog signal closely, testing the complete integrator emphasize this as seen in the table.

7.2.2 Detectability in Bias Circuit

Figure 7-6 Show the bias circuit, it consist of a current mirror that distributes and scales bias current for the modulator. The schematic is repeated in a larger version in appendix B.1.3.

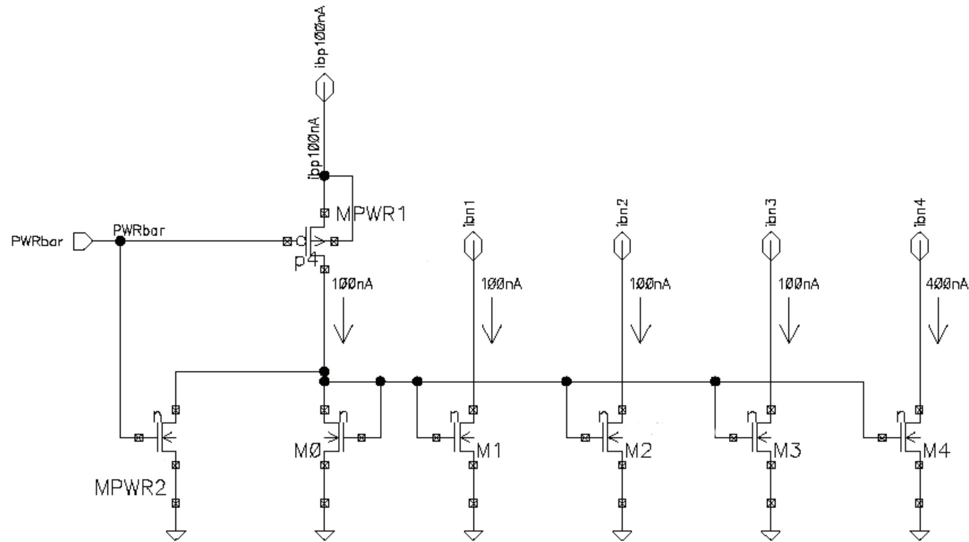

Figure 7-6 Bias Circuit Schematic

The transistors scale bias current to the following modulator parts:

M1 → Integrator 1
M2 → Integrator 2
M3 → Comparator
M4 → Common mode Voltage generator

These transistors have been tested for the following parametric deviances:

- L_{eff} -20 %
- L_{eff} +20 %
- W_{eff} -20 %

To increase readability only the most noteworthy performance changes for the transistors are shown, complete simulation results can be found in appendix F.3

Table 7-6 Performance with deviations introduced in the Bias Circuit

Placement	Deviance Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
M1	L_{eff} +20%	Test	65,76	-0,02	0,0259	0,00
		Analog	65,50	0,86	0,1879	0,00
M2	L_{eff} +20%	Test	39,50	-26,28	0,0035	-86,49
		Analog	31,45	-33,19	0,0169	-91,01
M3	L_{eff} +20%	Test	66,29	0,51	0,0259	0,00
		Analog	64,72	0,08	0,1879	0,00
M4	L_{eff} -20%	Test	66,29	0,51	0,0259	0,00
		Analog	65,09	0,45	0,1879	0,00

When the current to integrator 1 is reduced, by increasing transistor M1's length, there is only a minute change in SINAD, while reducing current to integrator 2 cause the amplitude and SINAD to plunge. Root cause for this might be that the reduced bias current causes the integrators transistors to move out of their active working range.
As seen when testing the integrator the test stimulus is able to detect these changes.

7.2.3 Detectability in the Common Mode Voltage generator

Figure 7-7 show the schematic for the common mode voltage generator responsible for generating and supplying the common voltage level used in the integrators. The schematic is repeated in a larger version in appendix B.1.4

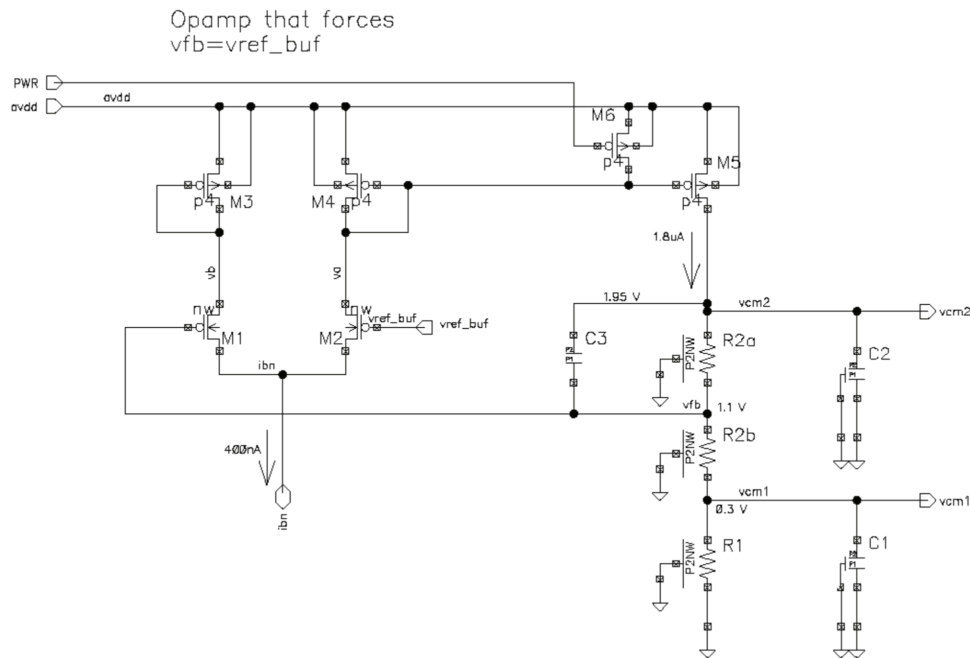

Figure 7-7 Common voltage schematic

To verify the test stimulus suitability deviations were introduced in transistor M2, and the following capacitors: C2 and C3.

The transistor was simulated with the following parametric deviances:

- L_{eff} -20 %
- L_{eff} +20 %
- W_{eff} -20 %

To increase readability only the most noteworthy performance changes for the transistors are shown, complete simulation results can be found in appendix F.2

Table 7-7 Performance with deviations introduced in the Common mode Voltage Generator

Placement	Deviance Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
C2	Short	Test Analog	66,03 64,78	0,25 0,14	0,0259 0,1879	0,00 0,00
C3	Short	Test Analog	64,15 62,49	-1,63 -2,15	0,0258 0,1870	-0,39 -0,48
M2	L_{eff} -20%	Test Analog	65,30 64,28	-0,48 -0,36	0,0259 0,1879	0,00 0,00
	W_{eff} -20%	Test Analog	65,55 64,27	-0,23 -0,37	0,0259 0,1878	0,00 -0,05

When testing a short in capacitor C3 the test signal show a reduced capability to reveal flaws, there is a reduction of 0.5 dB between the performance given from a regular analog input signal and the test signal, although the soft fault is indicated in test mode this reduction could lead to problems when implemented in a chip.

7.2.4 Detectability in the Comparator

Figure 7-8 Show the comparator responsible for sampling the output from integrator 2 and converting it to a digital value. The schematic is repeated in a larger version in appendix B.1.5

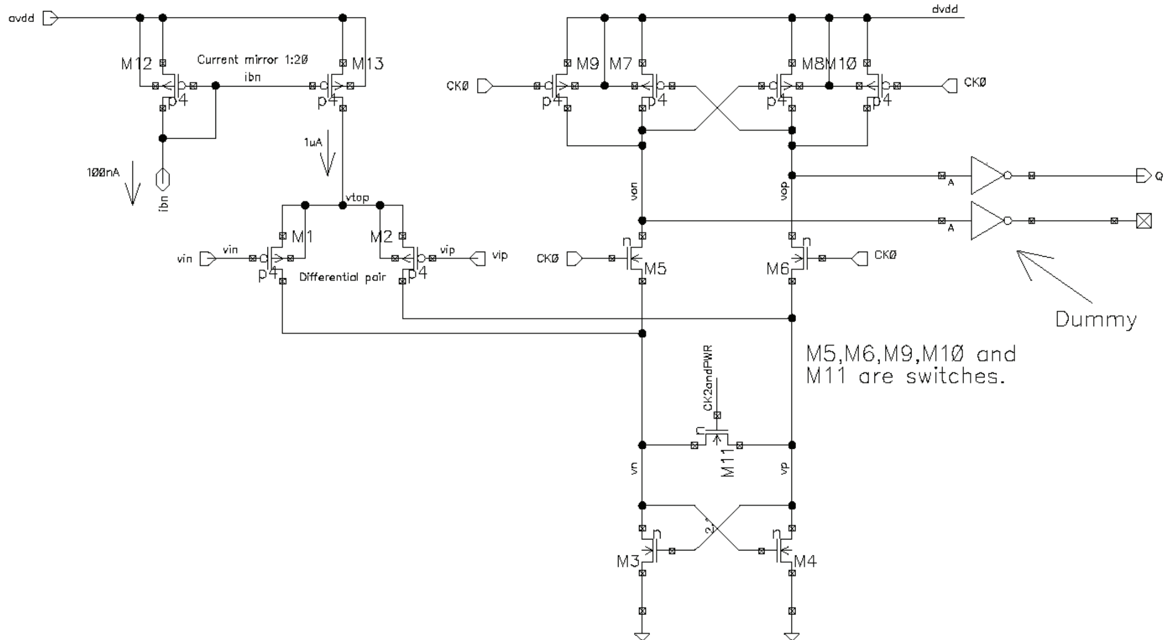

Figure 7-8 Comparator schematic

To verify the test stimulus suitability deviations were introduced in the following transistors: M2, M4, M8 and M13

The transistors were simulated with the following parametric deviances:

- L_{eff} -20 %
- L_{eff} +20 %
- W_{eff} -20 %

To increase readability only the most noteworthy performance changes for the transistors are shown, complete simulation results can be found in appendix F.4

Table 7-8 Performance with deviations introduced in the Comparator

Placement	Deviation Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
M2	W_{eff} -20%	Test	65,44	-0,34	0,0259	0,00
		Analog	64,49	-0,15	0,1879	0,00
M4	L_{eff} +20%	Test	65,39	-0,39	0,0259	0,00
		Analog	64,45	-0,19	0,1879	0,00
M8	W_{eff} -20%	Test	65,13	-0,65	0,0253	-2,32
		Analog	64,12	-0,52	0,1852	-1,44

As Table 7-8 show the test stimulus introduce a larger degradation of the signal quality than that of the analog stimulus. Testing with an increased length in transistor M4 the test signal deviates with 0.3 dB from that of the analog signal, this could cause converters that are within the performance parameters to be falsely indicated as defect ones.

7.3 Summary

Simulations run show that it is possible to detect soft and hard faults by using a stream replicating a ΔΣ-DAC without an analog filter as test stimulus. This methods suitability for testing and validating an ADC will however depend on the requirements and specifications of the converter. The test method deviates from the real results that an analog test signal produce. Maximum deviation found was 0.5 dB, and for most ADCs this level of test uncertainty may be acceptable.

8 BIST Proposal

Before integrating a self-test into a commercial available product some reflections have to be made. The test must not lead to a large area increase, increasing area leads to an increase in production cost. The test have to reduce testing time, extra area occupied by the test architecture has to be paid in shorter testing time and therefore a lower unit price. The test has to remove or lower the requirements of the off-chip tester, and by doing so reducing cost.

Michale J. Ohletz shows that between 75% and 84% of all failures observed in analog circuits are hard faults. (Ohletz, 1996a) These can be found and observed as early as in wafer tests, however due to the noisy environment that exists in wafer test a simple and robust test to sort out these faults would be favorable.

As seen in section 7 using a binary stream vector as DAC stimulus is capable to reveal both soft and hard faults in a ΔΣ-ADC. However the proposed test by (Rolindez et al., 2006) uses a very long bit-stream which would require a large amount of area.

By further developing the BIST scheme proposed in (Rolindez et al., 2006) by implementing offset and gain-error detection, reducing the shift register length and adapting it so that it uses the onboard resources available on 8-bit microcontrollers this can be a viable test scheme.

8.1 BIST Structure

By utilizing the available CPU and RAM the BIST architecture can be modeled as shown in Figure 8-1

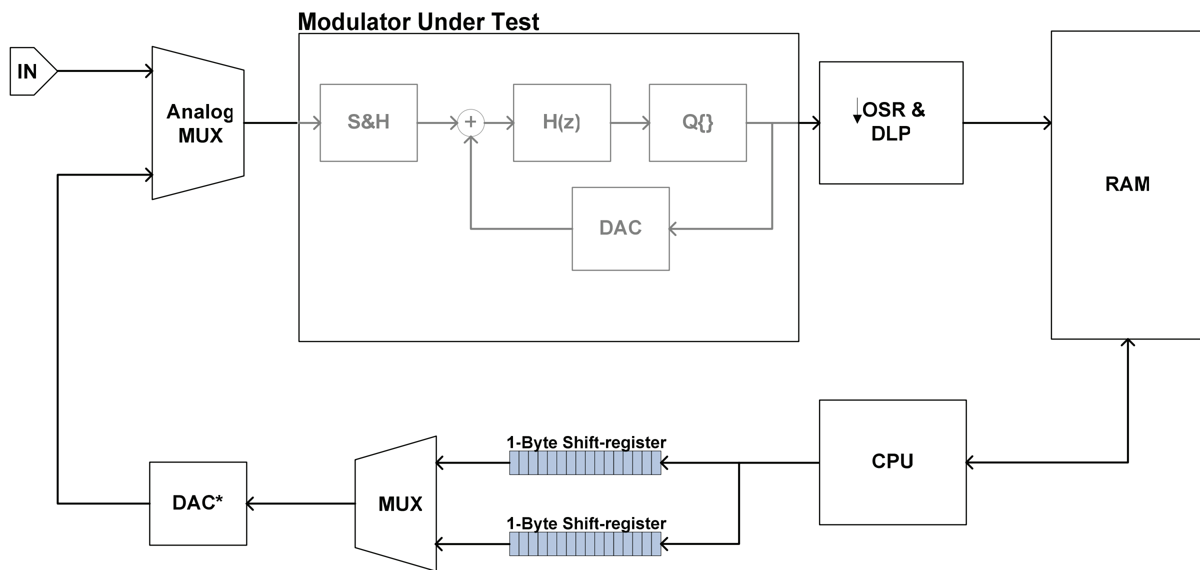

Figure 8-1 BIST architecture

BIST sequence:

- A binary string is loaded into RAM
- One byte is shifted into each register and test input is selected
- Test starts. First shift-register shifts binary data to DAC
- DAC converts this into analog signal and it is used to stimulate the MUT
- After shifting the first byte, the second register is chosen and used as stimulus
- CPU collects third byte and shift it into the first register
- Repeated until the whole binary string has been used, then start at the first byte again.
- Repeated until desired test length is reached.

The modulator output is down-sampled, filtered and stored in RAM. Depending on the type of test carried out and resources available in the microcontroller the output is either analyzed on or off-chip.

8.2 BIST Signal Generation

8.2.1 Dynamic Signal Generation

In order to reduce the binary stimulus length on can try to utilize the periodic behavior inherent to a sine wave. As shown in Figure 8-2 modulator output (blue waveform) is non-periodic even though a periodic stimulus (red waveform) is used.

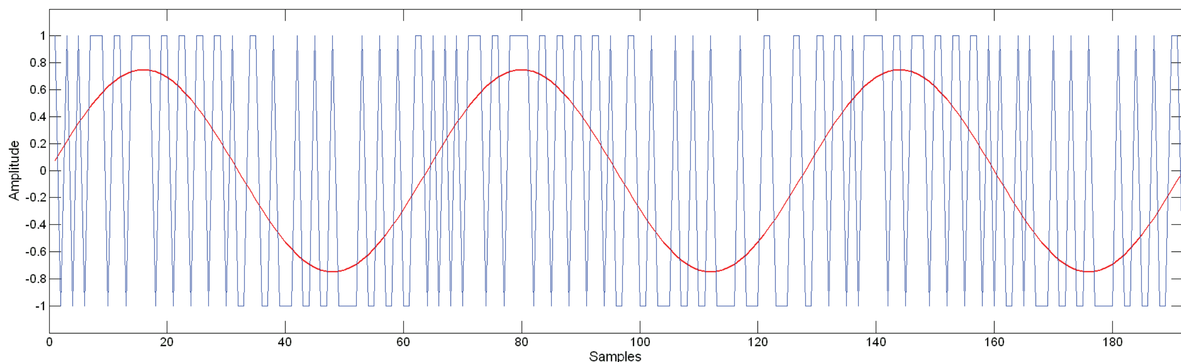

Figure 8-2 Non-periodic ΔΣ-modulator output

However by ensuring that the input stimulus maintains the condition set by the coherent sampling theorem. An approximation of the original modulator output is possible to generate by repeating the output generated in one single input period.

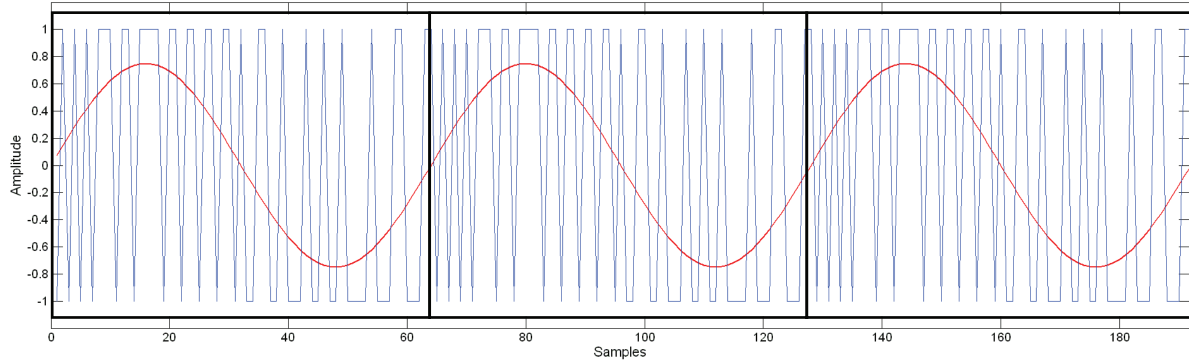

Figure 8-3 Periodic ΔΣ-modulator output

Using this technique a periodic signal is easily generated. Figure 8-3 shows how an example of this approximation. Each of the bounding boxes is identical. Length of the periodic signal will depend on the sampling frequency, input frequency and if it is desirable to test with multi-tone stimulus, all of these have to fulfill the coherent sampling theorem as explained in section 5.1.5.
Nevertheless the absolute minimum length is determined by the Nyquist rate given in section 2.2, and repeated here for convenience,

$$f_{sn} = 2f_0 \quad (8\text{-}1)$$

This indicates that the absolute minimum pattern length would be *2*OSR*, but since a sine-wave is not just periodic but also symmetrical it is possible to utilize the symmetrical behavior to shorten the stimulus length even further.

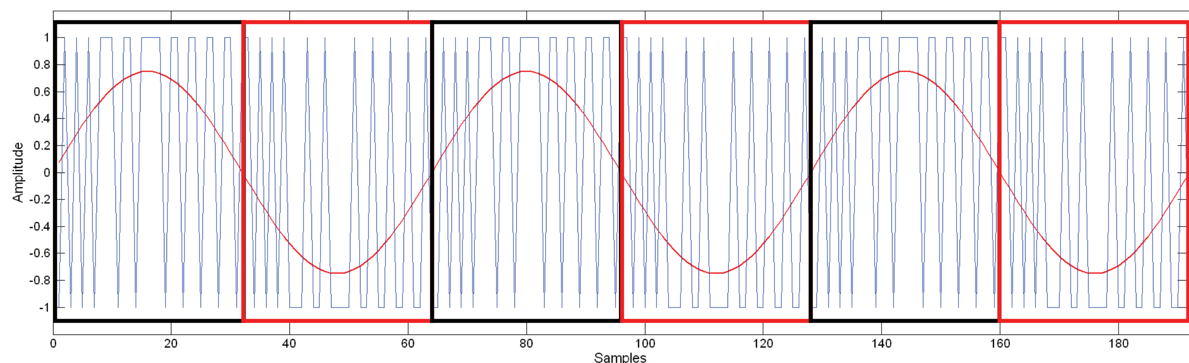

Figure 8-4 Optimized ΔΣ-modulator output

Figure 8-4 shows the resulting bit pattern. The red squares is the inverted form of the black squares thus utilizing a sine-waves symmetric behavior, this gives a minimum pattern length of *OSR*.

8.2.2 Static Signal Generation

To facilitate static testing, the binary stimulus needs to represent a stationary DC signal. Construction of such a signal is much simpler than a waveform and it can be created on the fly during testing. By using an alternate pattern of zeros and ones, and adjusting the ratio between these, it is possible to represent different static input values. The testable range will be limited by the output amplitude from the DAC. By implementing a DAC with four levels, where the new levels are placed at the ΔΣ-modulators maximum and minimum amplitude it is possible to represent the entire input range. However when using signals with such large amplitudes a special care has to de made to assure that the modulator is not overloaded by the rapid falling edges of the signal. By allowing the stimulus to be held stationary for several samples it is possible to minimize the influence these have on the modulators internal voltages.

Linearity of the ΔΣ-modulator can be found by using an input containing zeroes, the filtered output will then be at its minimum value. By replacing one of the samples with a sample containing a one after each time OSR samples has been placed at the modulator. The input will create a slowly rising ramp that can be used to estimate the transition level in accordance to (IEEE_Std_1241, 2001).

8.3 Signal Quality

Shortening the binary test vector from a long and correct one down to a shorter approximation will degrade the signal quality thus reducing the signal to noise ratio of the test signal. Figure 8-5 show the output signal spectrum of an ideal ΔΣ-modulator when stimulated with the high resolution non-periodic signal and when it is stimulated with the periodic optimized test stream.

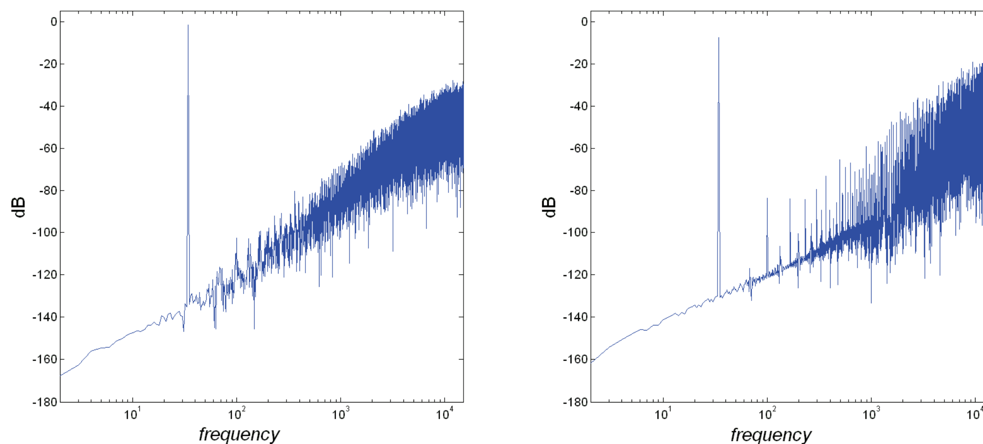

Figure 8-5 Modulator output spectrum. High-resolution and Periodic binary stimulus

As shown in Figure 8-5 the remaining in-band-noise and the loss of resolution caused by shortening the stream lead to a raised noise floor and appearance of harmonic peaks.

Table 8-1 SINAD different input stimuli

Stimulus	SINAD		Loss	
Regular	129.9	dB	-	-
Periodic	129.6	dB	0.3	dB
Optimized	128.8	dB	1.1	dB

Table 8-1 clearly show that the unfiltered stream does not cause a large degradation of the signal, however using the *OSR* long stream causes a loss of 0.8 dB when compared to the *2*OSR* long periodic stream.

The ΔΣ-modulator used in this project was simulated with both the optimized periodic binary stimulus and a high resolution sine-wave.

8.4 Signal Analysis

Signal analysis will depend on the test performed and the sought accuracy. A linearity test will open for digital compensation and thereby increasing linearity in low bandwidth converters. On the other hand nonlinearities and circuit errors may be masked out and remain undetected in a static linearity test. SINAD testing detects almost all circuit errors and deficiencies, but will not detect offset- and gain-errors. Running offset and gain-error test in addition to a SINAD test will detect and characterize almost all errors and thereby provide very good fault coverage.

Estimating offset and gain errors can easily be carried out on-chip. This can be done by comparing the expected output value with the actual output value. This means that the input value must be known, if the input is an on-chip generated ramp this value can be found by using the time passed before a code transition is achieved, and calculating the input value from this.

When testing a circuit dynamically, SINAD analysis can be performed in the time domain or in the frequency domain. Analysis in the frequency domain using DFT requires a large number of multiplications on the complete data set, which can be time consuming and difficult to execute in an 8-bit microcontroller core.

The algorithm used to perform a time domain analysis SINAD using sine-wave fitting was discussed in section 5.2.3 and as shown performing this can be a challenging task when the resources are as limited, as they generally are in a microcontroller. But for a known frequency, and if it is possible to synchronize the phase between the reference and out-signal a sine wave fitting may be feasible.

The reference signal can either be obtained by simulation off-chip, and loaded into RAM at the same time as the stimulus or it can be calculated by modifying the BIST architecture to enable generation on-chip (Rolindez et al., 2006).

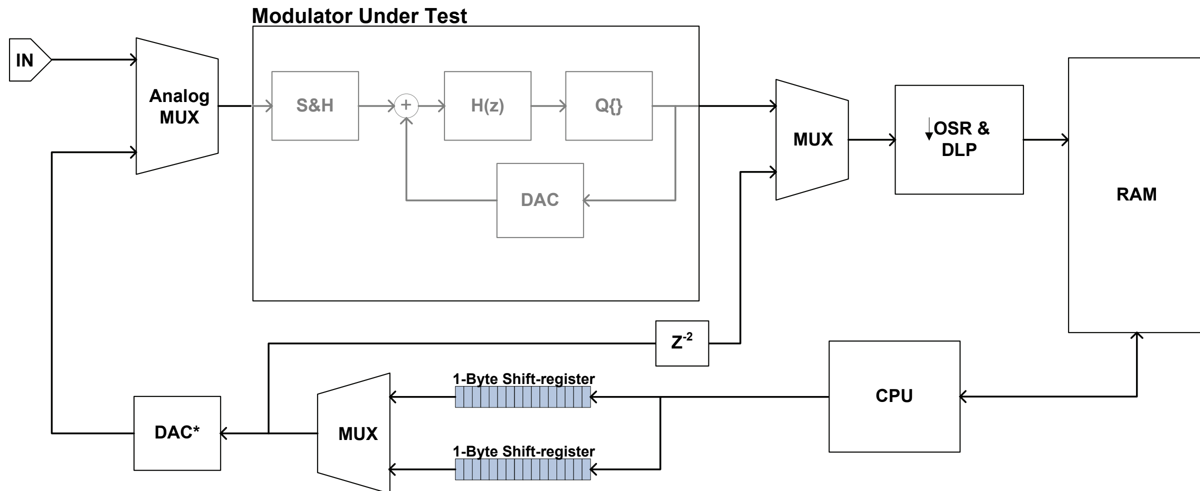

Figure 8-6 Modified BIST architecture

Implementing a MUX between the modulator and the filter allows the reference to be filtered. The delay element ensures that the two signals have synchronized phase.
Depending on the total RAM available the test can either do two runs, one with the reference as filter input and then one with the MUT as filter input, or utilize time interleaving to generate both signals in one single run. To employ time interleaving the filters internal frequency has to be doubled and all the internal registers has to be duplicated thus increasing area.

To achieve the necessary resolution of two bit more than the modulator output various actions can be taken. Since the digital filter has finite-size internal registers increasing this register length will gain some resolution. Simulating the sequence off-chip utilizing higher order filters than realizable in hardware will also give some resolution, but doing so would remove the possibility to run the test fully on-chip if the on-chip RAM is of an insufficient size. Nevertheless it is one implementation that needs to be considered. One way to ensure resolution is to generate stimulus by simulating a ΔΣ-modulator with a higher order than the modulator under test. The higher out-of-band quantization noise will unfortunately require a higher order in the internal decimation and low-pass filter to suppress this noise and by doing so increasing area.
If adequate resolution can be achieved a complete test sequence can be executed through the following proposal (Rolindez et al., 2006):

BIST Test sequence, known frequency and synchronized phase

- Calculate offset by using a static test or by using equation **(8-2)**

$$Offset = \frac{1}{M}\sum_{i=1}^{M} y(i) \qquad \textbf{(8-2)}$$

- Calculate gain by using a static test or by using equation **(8-3)**

$$Gain = \frac{2}{M \cdot Gain_{ref}}\sum_{i=1}^{M} y(i) \cdot y_{ref}(i) \qquad \textbf{(8-3)}$$

- Compensate reference to fit the reference:

$$y_{ref,fitted}(i) = \frac{Gain}{Gain_{ref}} y_{ref}(i) + Offset \qquad \textbf{(8-4)}$$

- Subtract reference from out-signal:

$$N_{error} = \sum_{i=1}^{M} y(i) - y_{ref,fitted}(i) \qquad \textbf{(8-5)}$$

- Sum all samples and divide by the number of samples to get $Noise_{RMS}$:

$$Noise_{RMS} = \sqrt{\frac{N_{error}}{M}} \qquad \textbf{(8-6)}$$

- Divide $Signal_{RMS}$ with $Noise_{RMS}$ to get SINAD:

$$SINAD = \frac{Signal_{RMS}}{Noise_{RMS}} \qquad \textbf{(8-7)}$$

Where M is the number of samples, y is the output signal, y_{ref} is the reference signal $Gain$ is the fitted gain, $Gain_{ref}$ is the gain of the reference signal and N_{error} is noise error

As equation **(8-4)** and **(8-5)** indicate it is possible to start the signal analysis before the complete out sequence is completed and by so doing reducing the computation time required to attain a the final SINAD result.

8.5 BIST Performance

8.5.1 Dynamic Testing

To validate the proposed BIST's ability to detect modulator errors and non-idealities the ΔΣ-ADC was simulated using the optimized stimulus with the shortest possible length. The resulting response was analyzed in GNU Octave using the algorithm proposed in section 8.4, and a four parameter fit algorithm. To ensure that the ADCs response to test stimulus was comparable an additional simulation with analog input stimulus was conducted.

To ensure sufficient resolution for on-chip analysis a reference signal was created by filtering the output stream in a modified version of the decimation filter featuring a 15-bit output instead of the usual 13-bit. None of the internal filter registers were modified since their length of 22 bit is more than adequate. This modification proved to gain 1.7 bit thus giving the reference signal a resolution of:

$$RESOLUTION_{REF} = 14.7\,Bit \qquad \textbf{(8-8)}$$

Increasing the filter output by 2-bit did not result in a 2-bit increase in resolution but this has to be attributed to the close proximity to the cut-of frequency.

The test was run for six sine wave periods, or 190 *ms.* Were the first period was used to ensure the filters internal state and the five succeeding periods was used to calculate SINAD and amplitude.

As in section 7.2 deviations were placed in the modulator, the most noteworthy results are shown in Table 8-2. Complete results can be found in appendix G.

Table 8-2 BIST performance

Device tested	Deviation Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
Op-Amp M2	Leff +20%	BIST	64,74	-0,77	0,0235	0,01
		TEST	65,02	-0,82	0,0254	0,00
		Analog	65,31	-0,95	0,1847	0,01
M6	Stuck Open	BIST	43,31	-22,20	0,0209	-11,22
		TEST	45,37	-20,47	0,0224	-11,81
		Analog	31,99	-34,27	0,1446	-21,71
M14	Weff -20%	BIST	64,97	-0,54	0,0239	1,72
		TEST	65,37	-0,47	0,0259	1,97
		Analog	65,32	-0,94	0,1879	1,73
Integrator 1 M7	Leff +20%	BIST	65,13	-0,38	0,0239	1,56
		TEST	65,54	-0,30	0,0259	1,97
		Analog	65,46	-0,80	0,1879	1,73
M10	Leff -20%	BIST	64,68	-0,83	0,0234	-0,21
		TEST	64,68	-1,16	0,0254	-0,12
		Analog	64,53	-1,73	0,1846	-0,05
M18	Weff -20%	BIST	65,19	-0,32	0,0235	-0,02
		TEST	65,93	0,09	0,0254	0,03
		Analog	66,21	-0,05	0,1847	0,01
Cmgen M2	Weff -20%	BIST	65,06	-0,45	0,0237	0,79
		TEST	65,52	-0,32	0,0254	0,08
		Analog	65,99	-0,27	0,1857	0,54
Bias M2	Leff +20%	BIST	65,04	-0,47	0,0235	-0,11
		TEST	65,63	-0,21	0,0253	-0,24
		Analog	65,36	-0,90	0,1846	-0,05
Comp M4	Leff +20%	BIST	65,12	-0,39	0,0235	-0,02
		TEST	65,48	-0,36	0,0254	-0,04
		Analog	66,08	-0,18	0,1847	-0,03

As the table show the BIST proposal responds to parametric deviances in a similar manner to the simulations performed in section 7.2. However the shorter stimuli length, the increased noise and not performing the test in compliance with the guidelines given in section 5.2.3.1 leads to a decrease in resolution. Maximum deviation for soft faults is found in the integrators M19 transistor where the proposed test only show a reduction of 0.83 dB versus the analog stimulus reduction of 1.73 dB. This deviation of 0.93 dB is large and a faulty device might be reported to be within tolerance limits and consequently pass testing.
Deviance between the response when using test and analog stimulus is greater when it comes to hard faults, however this is of less importance since the test clearly detects hard faults.

8.5.2 Static Testing

To investigate if it is possible to represent a static DC input with a binary stream the MUT was stimulated with a stationary DC-input and with a binary stream corresponding to this DC input. To ensure accuracy the decimation filters 18-bit output and a test length of 10 second (2560 18-bit samples) were used.

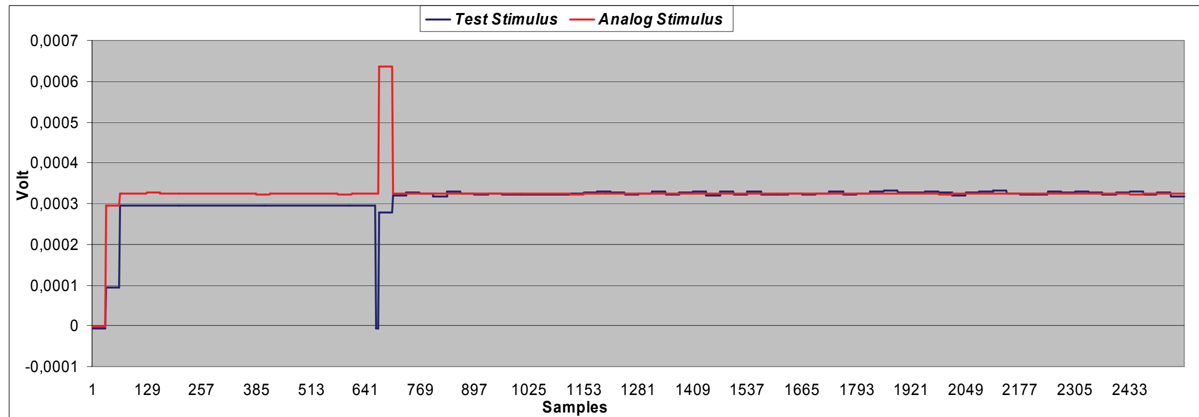

Figure 8-7 Static vs Test Stimuli

Figure 8-7 show the filtered output from the MUT, with an analog input stimulus of zero volts was used. As test stimulus an alternating pattern of ±40 mV is used.
As the figure indicates it is possible to use an alternating pattern to represent DC values. One can clearly see that the modulator under test has an offset of 3 mV using both DC and test stimulus. The figure also clearly show that to the high oversampling and accumulation in the decimation filter require a long test time, in this case samples prior to sample 759 are invalid due to the internal state of the filter.

9 Discussion

The proposed test shows promise of a relative standalone test that may be able to cut cost associated with testing ΔΣ-ADC embedded in commercially available microcontrollers. There are however some limitations to the test.

9.1 BIST coverage

The proposed test scheme seems to be able to detect all hard faults and all soft faults caused by parametric deviances that have been tested. Using a long test vector containing at least five cycles of a sine wave with a relative prime relationship with the sampling frequency an estimate of SINAD within 0.5 dB seen to be possible and a test carried out in this manner will be in compliance with section 4.1.3.5 in (IEEE_Std_1057, 1994) while the proposed test scheme with a shortened test vector seems to be capable of estimating SINAD within 1 dB.

Shortening the test vector to the proposed minimum length of *OSR* will lead to a reduction in sensitivity. If one sample of the sine wave is off by one *LSB* when using the long test vector the error will usually not be repeated since the sine is sampled at different places each cycle. A single error using the parameters used in section 7 will then lead to a noise increase by:

$$Noise_{RMS} = \sqrt{\frac{1}{L} \cdot \left(\frac{A_{signal}}{2^N}\right)^2} = \sqrt{\frac{1}{64} \cdot \left(\frac{0.440mV}{2^{13}}\right)^2} = 7 \cdot 10^{-6} V_{RMS} \qquad \textbf{(9-1)}$$

However if the same situation rises in the proposed test scheme the deviation may be repeated each cycle due to the fixed sampling point and therefore lead to an increase in noise by:

$$Noise_{RMS} = \sqrt{\frac{Cycles}{L} \cdot \cdot \left(\frac{A_{signal}}{2^N}\right)^2} = \sqrt{\frac{1}{40} \cdot 5 \cdot \left(\frac{0.440mV}{2^{13}}\right)^2} = 19 \cdot 10^{-6} V_{RMS} \qquad \textbf{(9-2)}$$

Where

$$L = Cycles \cdot Samples_{Cycle} \qquad \textbf{(9-3)}$$

A_{signal} is the signals amplitude and N is the number of bit.

Which is an increase in noise by a factor of three, in addition formula **(9-2)** and **(9-3)** show that increasing test length L will increase the number of cycles thus keeping the error constant since L is a multiple of the number of cycles and the number of samples per cycle.

Changing the BIST vectors frequency slightly would reduce or remove this problem at the cost of a small increase in test vector length. By increasing the test vector to 548 samples the resulting frequency would become:

$$f_{test} = \frac{f_s}{2 \cdot Vector_length} = \frac{32768Hz}{2 * 548} = \frac{4096}{137} Hz \approx 29.9Hz \qquad \textbf{(9-4)}$$

a frequency that is relatively prime to the sampling frequency and it is therefore possible to perform the test in compliance with the IEEE standard.

The proposed BIST is unable to detect clock jitter due to the digital nature of the test, this is a major limitation since clock jitter is a major contribute to the degradation in ADC performance.

Clock jitter error power was in section 3.1 found to be:

$$e_\delta^2 = \frac{1}{2}\left(2\pi f_0 \Delta\tau A\right)^2 \qquad \textbf{(9-5)}$$

Since this error power is caused by the sampling time uncertainty $\Delta\tau$, calibrating the clock externally will reduce the error power and at the same time allow for an estimate of the clock jitter, therefore this lack of detection may be accepted.

To ensure that the test results are valid a calibration of the internal bandgap references need to be performed. The tests DAC and the DUT employ a fully differential architecture that ensures a high degree of offset cancellation, gain would be severely influenced by reduced supply voltages.

9.2 BIST Arithmetic Requirements

Performing the proposed test scheme on the fly requires that the microcontroller is capable to perform simple arithmetic operations between each filtered output.

Calculating gain and offset on the fly and simultaneously in this case requires one 13*15-bit multiplication and two 16-bit summations which in an Atmel AVR 8-bit microcontroller would call for 23 clock cycles. The last step will require one 32-bit division for offset calculation and one 32-bit division for gain calculation, each call for up to 640 clock cycles.

Sine wave fitting requires one multiplication, one summation and one subtraction, requiring up to 19 clock cycles. In addition the final noise calculation requires one 32-bit division, requiring up to 640 clock cycles and one square root which is next to impossible to perform in an 8 bit microcontroller.
By squaring the Signal amplitude instead of extracting the square root of the noise SINAD can be computed in less than 640 clock cycles.

These requirements make signal analysis impossible to perform on the fly if the microcontroller runs synchronous with the ΔΣ-ADC. Complete calculations can be found in appendix E

9.3 BIST Implementation Cost

The implementation cost assumes that this BIST is to be implemented on a microcontroller with available memory, CPU, and means to load data onto the device.

This report proposes two possible methods of implementing the BIST onto a microcontroller.

The common requirements are:

- Available RAM to store the complete binary string.
    - The memory requirement will be dependant of string-length, if a OSR long string as proposed is to be used the requirement will be 64 Byte with an oversampling rate of 512.
- Two shift registers
    - Length of these will depend on the RAM access time and CPU/ADC frequency ratio. One byte should be sufficient if the RAM access time is four cycles or less.
- One Digital MUX
    - To shift between the registers.
- One one-bit DAC
    - This DAC has to have to modes, one where an attenuated voltage is used as output and one where $\pm V_{dd}$ is used as output.
- One Analog MUX
    - To enable switching between the regular input and the test input.
- One extra order in the digital low-pass and decimation filter.
    - This requirement will only be necessary if the binary stimulus has to be created in a higher order modulator to assure sufficient resolution.

Depending on whether the signal analysis is to be performed on-chip or off-chip the implementation will have different requirements when it comes to output from the modulator under test. These two cases will be discussed separately.

9.3.1 BIST with External Signal Analysis

In addition to the common requirements an off-chip analysis will require either RAM to store data in or a fast way to load data off-chip. The modulator used in this test will require storage of at least 40 samples or 80 byte.

9.3.2 BIST with Internal Signal Analysis

If the signal analysis is to be performed on-chip the requirements will be dependant of the RAM available. If it is insufficient RAM to store one complete test, the filter has to be modified to use time-interleaving. This would require doubling all internal registers and doubling the filers operating frequency.

If there is a sufficient RAM available two possible solutions exists.
The first is to load the signal reference at the same time as the binary stream is loaded, this will not require any extra area compared to the off-chip analysis.

The other solution is to perform a separate run where the output from the shift registers is run directly to the filter without going through the modulator under test and by so doing creating the reference signal on-chip.
This would require another digital MUX and one delay element consisting of a two bit shift-register to ensure synchronization between the reference and the test output. To achieve the necessary extra resolution extending the filters internal registers with three bit and in worst case increasing the filter order with one will be required.

9.3.3 Test time

Total test time will be a sum consisting of:

- Loading the stimulus vector to the chip.
- Ensuring the internal filters state (one period of the stimulus)
- At least five periods with test stimulus.
- Loading data off-chip or analyzing in on-chip.

During testing of the proposed BIST scheme the test was run for six sine wave periods, or 190 *ms*. If sufficient RAM is available or off-chip signal analysis is to be performed required test time would then be 190 *ms* plus the overhead required for loading data. Lack of RAM and on-chip analysis would require running a static test.

9.3.4 Implementation Summary

If the test proposal is to be implemented without on-chip analysis, the area required is very small. The cost associated with the increased area is more than outweigh by removing the demand for an external high precision analog signal generator.
When on-chip analysis is to be implemented the extra area required will vary, if the sought reference resolution can be achieved by analyzing the reference signal off-chip and loading it at the same time as the binary string is loaded, the implementation will not require much more area than the off-chip analysis. If the reference is to be generated on-chip the area increase will depend on the total RAM available, if the necessary quality can be achieved by increasing the internal filter registers length or if a higher order is necessitated. In low-bandwidth high-oversampling applications with very high accuracy the medium implementation cost this would require should be outweighed by the extreme requirements needed by the external signal generator and the possibility to run the test in parallel with other testing.

9.4 External Test Equipment Requirements

Depending on type of signal analysis the external test equipment requirements will be much reduced from a regular test. The demand for a high precision analog source is completely removed and if internal signal analysis is to be used the only requirement is a method of loading the binary stream into the microcontroller.
Removing the analog input requirement open for the possibility to test the ADC on wafer, before packing thus improving production yield and by so doing cutting production costs.

If the Signal analysis is to be performed off-chip, an FFT-analyzer or sine-wave fitting has to be performed which even the simplest production tester is able to execute.

10 Conclusion

A Built-in Self Test have been proposed and simulated to explore its ability to detect and characterize non-idealities in a ΔΣ-modulator. The test is capable of detecting soft and hard faults subjected to the modulator under test and is capable of estimating offset, gain and SINAD with a high degree of accuracy.

With the exception of one 1-bit DAC, with attenuation, the test is fully digital, and as a result the need for expensive AMS-testers is eliminated. By removing the requirement for a high precision analog input stimulus, wafer testing is rendered possible and production yield may be improved. The digital nature of the test leads to clock jitter insensibility, necessitating external clock calibration.

Due to the extensive number of clock cycles required to perform 32-bit divisions, an 8-bit microcontroller running on the same frequency as the sampling frequency is not capable of performing on the fly signal analysis. However clocking the microcontroller at 350 kHz or more makes on-chip signal analysis feasible. An external signal analysis will only be required if the maximum microcontroller frequency is lower than this and the available RAM is of insufficient size to store data.

The proposed test will not reduce test time by itself, however by integrating the test on-chip and allowing this to run in the background while other on-chip modules are tested, total test time can be reduced to the time required to shift the stimuli into the chip.

11 Future work

Although the proposed test setup is quite straight forward, and simulations show that the proposed BIST is capable to detect soft and hard faults, testing on a physical ADC would be required before full scale integration in Atmel's AVR 8-bit microcontrollers.

A modified version of Atmel's battery monitoring microcontrollers is in production where the sampling clock, the ΔΣ-ADC's input and output are available on external I/O pins making testing is possible.

A References

ATMEL 2002 AVR201: Using the AVR Hardware Multiplier Date Last Accessed: 01.may 2007 http://www.atmel.com/dyn/resources/prod_documents/doc1631.pdf

ATMEL 2005 Smart Battery and Battery Protection Functionality Date Last Accessed: 18 may 2005 http://www.atmel.com/dyn/resources/prod_documents/doc4083_mega406.pdf

ATMEL 2006 ATmega406 Data Sheet Date Last Accessed: 18 may 2006 http://www.atmel.com/dyn/resources/prod_documents/doc2548.pdf

BOSER, B. E. & WOOLEY, B. A. (1988) The design of sigma-delta modulation analog-to-digital converters. *IEEE Journal of Solid-State Circuits,* 23**,** 1298-1308.

CHEE-KIAN, O. & KWANG-TING, C. (2002) Self-testing second-order delta-sigma modulators using digital stimulus.

CHEE-KIAN, O., KWANG-TING, C. & WANG, L. C. (2004) A new sigma-delta modulator architecture for testing using digital stimulus. *IEEE Transactions on Circuits and Systems,* 51**,** 206-213.

DIAS, V. F., PALMISANO, G., O'LEARY, P. & MALOBERTI, F. (1992) Fundamental limitations of switched-capacitor sigma-delta modulators. *IEE Journal of Circuits, Devices and Systems,* 139**,** 27-32.

DUFORT, B. & ROBERTS, G. W. (1997) Optimized periodic sigma-delta bitstreams for analog signal generation. *Proceedings of the 40th Midwest Symposium on Circuits and Systems.*

DUFORT, B. & ROBERTS, G. W. (1999) On-chip analog signal generation for mixed-signal built-in self-test. *IEEE Journal of Solid-State Circuits,* 34**,** 318-330.

EATON, J. W. (2004) GNU Octave. 2.1.63 ed.

HAWKINS, C., KESHAVARZI, A. & SEGURA, J. (2003) View from the bottom: nanometer technology AC parametric failures - why, where, and how to detect.

HAWKINS, C. F. & SEGURA, J. (1999) Test and reliability: partners in IC manufacturing. I. *Design & Test of Computers, IEEE,* 16**,** 64-71.

HAWKINS, C. F., SEGURA, J., SODEN, J. & DELLIN, T. (1999) Test and reliability: partners in IC manufacturing. 2. *Design & Test of Computers, IEEE,* 16**,** 66-73.

HAWKINS, C. F., SODEN, J. M., RIGHTER, A. W. & FERGUSON, F. J. (1994) Defect classes-an overdue paradigm for CMOS IC testing.

HAWRYSH, E. M. & ROBERTS, G. W. (1996) An integration of memory-based analog signal generation into current DFT architectures. *Proceedings of the International Test Conference.*

HOESCHELE, D., F. (1994) *Analog-to-Digital and Digital-to-Analog Convercion Techniques*, John Wiley & Sons, Inc.

IEEE_STD_1057 (1994) IEEE standard for digitizing waveform recorders. *IEEE Std 1057-1994.*

IEEE_STD_1241 (2001) IEEE standard for terminology and test methods for analog-to-digital converters. *IEEE Std 1241-2000.*

JOHNS, D. A. & MARTIN, K. (1997) *Analog Integrated Circuit Design*, John Wiley & Sons, Inc.

KUEN-JONG, L., SOON-JYH, C. & RUEI-SHIUAN, T. (2003) A sigma-delta modulation based BIST scheme for A/D converters. *12th Asian Test Symposium.*

LIENIG, J. (2006) introduction to electromigration-aware physical design. *International Symposium on Physical Design.* San Jose, California, USA, ACM Press New York, NY, USA.

MALCOVATI, P., BRIGATI, S., FRANCESCONI, F., MALOBERTI, F., CUSINATO, P. & BASCHIROTTO, A. (2003) Behavioral modeling of switched-capacitor sigma-delta modulators. *IEEE Transactions on Circuits and Systems,* 50**,** 352-364.

MATLAB (2006) 7.2.0.232, Release 2006a ed., The MathWorks, Inc.

MENTOR_GRAPHICS_CORPORATION (2006) ADVance MS. 4.5_1.1 ed.

NORSWORTHY, S. R. (1997) Delta-Sigma Data Converters. Theory, Design and Simulation. IN SCHREIER, R. & TEMES, G. C. (Eds.), IEEE Press.

OHLETZ, M. J. (1996a) L^2RFM-local layout realistic faults mapping scheme for analogue integrated circuits.

OHLETZ, M. J. (1996b) Realistic faults mapping scheme for the fault simulation of integrated analogue CMOS circuits.

PROAKIS, J., G. & MANOLAKIS, D. G. (1996) *Digital Signal Processing*, Prentice-Hall International, Inc.

ROLINDEZ, L., MIR, S., BOUNCEUR, A. & CARBONERO, J. L. (2006) A SNDR BIST for Sigma Delta analogue-to-digital converters. *Proceedings of the 24th IEEE VLSI Test Symposium.*

SCHREIER, R. (2004) The Delta-Sigma Toolbox. 7.1 ed., Mathworks.

SEGURA, J. & HAWKINS, C. F. (2004) *CMOS Electronics, How It Works, How It Fails*, IEEE Press.

SEGURA, J., KESHAVARZI, A., SODEN, J. & HAWKINS, C. (2002) Parametric failures in CMOS ICs - a defect-based analysis.

WANG, Y.-S., WANG, J.-X., LAI, F.-C. & YE, Y.-Z. (2005) A low-cost BIST scheme for ADC testing. *6th International Conference On ASIC.*

YUN-CHE, W. & KUEN-JONG, L. (2000) An on chip ADC test structure. *Proceedings of Design, Automation and Test in Europe.*

B ΔΣ-ADC used in testing

All testing and development in this thesis was performed on a second order ΔΣ-modulator used to monitor charge and discharge current flowing through an external sense resistor (Atmel, 2005). The modulator is of a modified Boser-Wooley design utilizing delaying integrators. Using delaying integrators is favorable because it allows the op-amps to settle independently, leading to greater signal independency, thus reducing the speed requirements (Boser and Wooley, 1988).

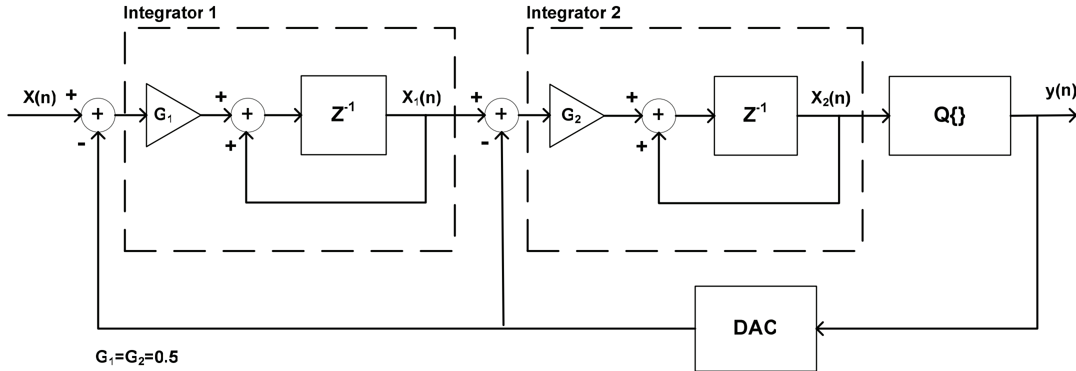

Figure B-1 Boser-Wooley modulator

In the original design Figure B-1 an integrator attenuation of 0.5 was used, however given the maximum input swing of ±220mV specified such a modulator structure would utilize a very small part of the integrators dynamic range. (Atmel, 2006) A modulator utilizing Boser-Wooleys original design as shown in Figure B-1 will, assuming that the quantization noise is modeled as a white noise, have a noise transfer function as following:

$$STF(z) = \frac{G_1 G_2 \cdot z^{-2}}{1 + (G_2 - 2) \cdot z^{-1} + (1 + G_1 G_2 - G_2) z^{-2}}$$

$$NTF(z) = \frac{1 - 2 \cdot z^{-1} + z^{-2}}{1 + (G_2 - 2) z^{-1} + (1 + G_1 G_2 - G_2) \cdot z^{-2}} \quad \text{(B-1)}$$

And the modulators output is given by:

$$Y(z) = STF(z)X(z) + NTF(z)E(z) \quad \text{(B-2)}$$

A maximum signal range of ±0.22 V is specified, while the integrators can swing between close to the voltage supply and zero. The penalty for not using a larger portion of the available swing is that the integrator capacitors are larger than necessary, knowing that G_1 and G_2 are given as the ratio between the sampling and integrator capacitor. By increasing G_1 by a factor of say, 4, the size of the integrator capacitor is reduced by the same amount.

However, by changing the integrator gain in the first stage the transfer function of the modulator is also changed. To compensate for the integrator gain in of the second stage was changed as well. This is shown in Figure B-2 where a modified signal flow-graph is shown.

The integrator gain of the first stage is increased, thus the integrator capacitances size are decreased.

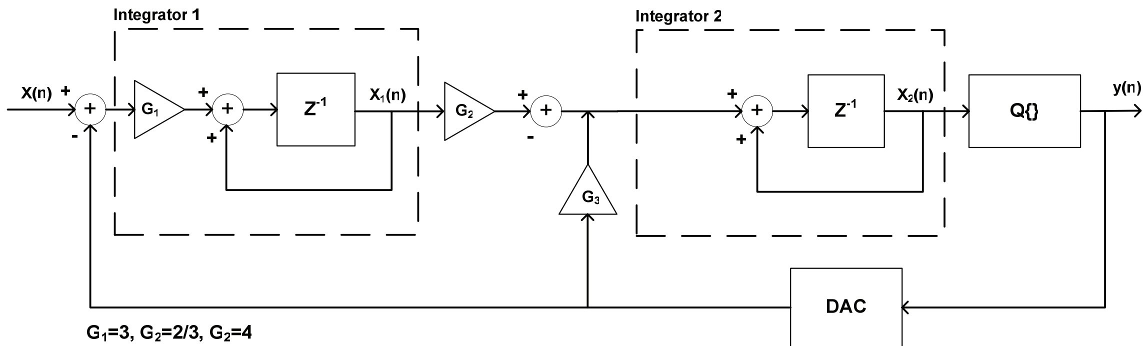

Figure B-2 Modified modulator structure

This modification gives the following relationship for the gain blocks in the modulator:

$$
\begin{aligned}
G_1 &= \alpha \cdot G_1 \\
G_2 &= \frac{G_1 \cdot G_2}{G_1} \cdot \beta = \frac{\beta}{\alpha} G_2 \\
G_3 &= \beta \cdot G_2
\end{aligned}
\quad \text{(B-3)}
$$

Where

$\alpha = 6$

$\beta = 8$

Table B-1 Key data ΔΣ- show key data for the ΔΣ-Modulator.

Table B-1 Key data ΔΣ-Modulator

	Name		Unit
Digital Supply Voltage	DVDD	3.3	Volt
Analog Supply Voltage	AVDD	3.3	Volt
Sampling frequency	fclk	32.768	kHz
Signal Bandwidth		32	Hz
Signal range		± 220	mV

B.1 ΔΣ-modulator schematics and VHDL-AMS mapping

Only the schematics and VHDL-AMS entities for those components subjected to parametric deviances and flaws are shown.

Figure B-3 show the modulators analog top level.

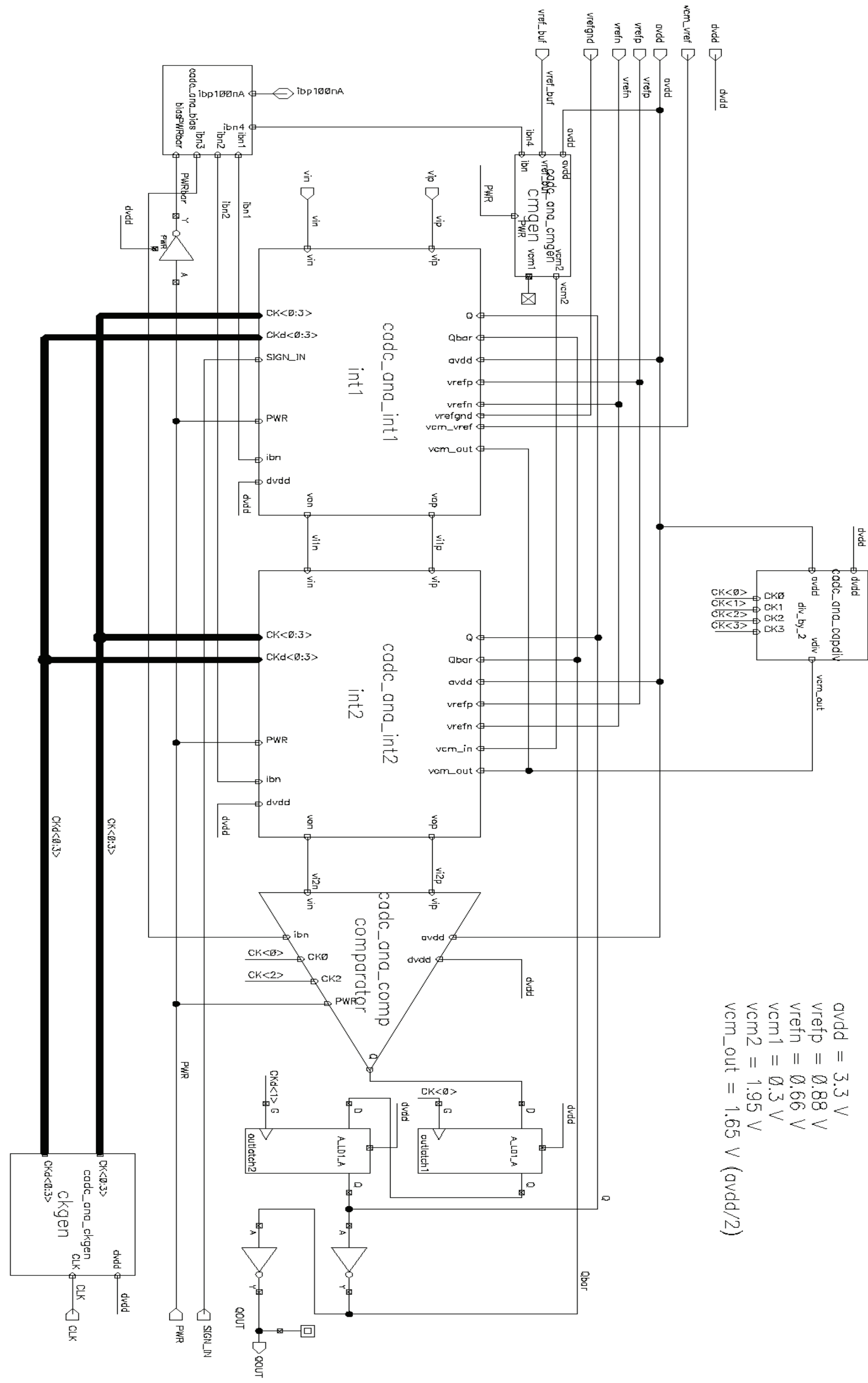

Figure B-3 Analog Top Level

A VHDL-AMS file maps this analog layout and create an interface to the test bench.

Top level entity and interface:

cadc_ana_int1-ent.vhd 09.06.2007

```
 1  -------------------------------------------------------------------------
 2  -- Filename:      cadc ana int1-ent.vhd
 3  -- Description:   Integrator 1 maps cadc ana int1-ckt
 4  -- Author:        Trond Jarle Pedersen <tjpedersen@atmel.com>
 5  -- Modified by:   Trond Jarle Pedersen <tjpedersen@atmel.com>
 6  --
 7  -------------------------------------------------------------------------
 8  library IEEE;
 9  use IEEE.MATH REAL.all;
10  use IEEE.std logic 1164.all;
11  library IEEE proposed;
12  use IEEE_proposed.electrical_systems.all;
13
14  entity CADC_ANA_INT1 is
15
16    port (
17      signal   CK 0      : in std logic;
18      signal   CK 1      : in std logic;
19      signal   CK 2      : in std logic;
20      signal   CK 3      : in std logic;
21      signal   CKd 0     : in std logic;
22      signal   CKd 1     : in std logic;
23      signal   CKd 2     : in std logic;
24      signal   CKd_3     : in std logic;
25      terminal PWR       :    electrical;
26      terminal Q         :    electrical;
27      terminal Qbar      :    electrical;
28      signal   SIGN_IN   : in std logic;
29      terminal AVDD      :    electrical;
30      terminal DVDD      :    electrical;
31      terminal IBN       :    electrical;
32      terminal VCM OUT   :    electrical;
33      terminal VCM_VREF  :    electrical;
34      terminal VIN       :    electrical;
35      terminal VIP       :    electrical;
36      terminal VON       :    electrical;
37      terminal VOP       :    electrical;
38      terminal VREFGND   :    electrical;
39      terminal VREFN     :    electrical;
40      terminal VREFP     :    electrical);
41
42  end entity CADC_ANA_INT1;
43
44  -------------------------------------------------------------------------
    --------
45
```

Analog top level structural mapping:

cadc_ana-str.vhd 09.06.2007

```
1  ----------------------------------------------------------------------
2  -- Filename:       cadc ana-str.vhd
3  -- Description:    Analog top Level,
4  --                 Maps SPICE implementation
5  -- Author:         Trond Jarle Pedersen <tjpedersen@atmel.com>
6  -- Modified by:    Trond Jarle Pedersen <tjpedersen@atmel.com>
7  ----------------------------------------------------------------------
8
9  architecture structural of cadc_ana is
10
11   terminal NET0103 :    electrical;
12   terminal NET045  :    electrical;
13   terminal NET050  :    electrical;
14   terminal Q       :    electrical;
15   terminal QBAR    :    electrical;
16   terminal PWRBAR  :    electrical;
17   terminal VCM2    :    electrical;
18   terminal IBN1    :    electrical;
19   terminal IBN2    :    electrical;
20   terminal IBN3    :    electrical;
21   terminal IBN4    :    electrical;
22   terminal VCM OUT :    electrical;
23   terminal VI1N    :    electrical;
24   terminal VI1P    :    electrical;
25   terminal VI2N    :    electrical;
26   terminal VI2P    :    electrical;
27   signal CK         : std logic vector(0 to 3);
28   signal CKD         : std_logic_vector(0 to 3);
29
30
31 begin
32
33 -- Submodule instantiation --------------------------------
34
35   Latch1 : entity work.A LD1_A(A_LD1_A)
36     port map(D    => NET0103,              -- D
37              G    => CK(0),                -- clk
38              PWR  => DVDD,
39              Q    => NET045                -- Q
40              );
41
42   Latch2 : entity work.A LD1_A(A_LD1_A)
43     port map(D    => NET045,               -- D
44              G    => CK(1),                -- clk
45              PWR  => DVDD,
46              Q    => Q                     -- Q
47              );
48   Inv1  : entity work.A_INV 1(A_INV_1)
49     port map(A          => PWR,
50              PWR        => DVDD,
51              Y          => PWRBAR
52              );
53   Inv2  : entity work.A_INV 1(A_INV_1)
54     port map(A          => Q,
55              PWR        => DVDD,
56              Y          => QBAR
57              );
58   Inv3  : entity work.A_INV 1(A INV_1)
59     port map(A          => QBAR,
60              PWR        => DVDD,
61              Y          => QOUT
```

```
cadc_ana-str.vhd                                                    09.06.2007

 62                 );
 63
 64       Cmgen : entity work.CADC_ANA CMGEN(CADC_ANA_CMGEN)
 65         port map(PWR          => PWR,
 66                  AVDD         => AVDD,
 67                  IBN          => IBN4,
 68                  VCM1         => NET050,
 69                  VCM2         => VCM2,
 70                  VREF_BUF     => VREF_BUF
 71                  );
 72
 73       capdiv : entity work.CADC ANA CAPDIV(CADC_ANA_CAPDIV)
 74         port map(CK0          => CK(0),
 75                  CK1          => CK(1),
 76                  CK2          => CK(2),
 77                  CK3          => CK(3),
 78                  AVDD         => AVDD,
 79                  DVDD         => DVDD,
 80                  VDIV         => VCM_OUT
 81                  );
 82
 83       bias   : entity work.CADC_ANA BIAS(CADC_ANA_BIAS)
 84         port map(PWRBAR       => PWRBAR,
 85                  IBN1         => IBN1,
 86                  IBN2         => IBN2,
 87                  IBN3         => IBN3,
 88                  IBN4         => IBN4,
 89                  IBP100NA     => IBP100NA
 90                  );
 91
 92       ckgen : entity work.CADC ANA_CKGEN(behavioral2)
 93         port map(CK          => CK,
 94                  CKD         => CKD,
 95                  CLK          => CLK
 96                  );
 97
 98       int2 : entity work.CADC ANA INT2(CADC_ANA_INT2)
 99         port map(CK 0         => CK(0),
100                  CK 1         => CK(1),
101                  CK 2         => CK(2),
102                  CK 3         => CK(3),
103                  CKD 0        => CKD(0),
104                  CKD 1        => CKD(1),
105                  CKD 2        => CKD(2),
106                  CKD_3        => CKD(3),
107                  PWR          => PWR,
108                  Q            => Q,
109                  QBAR         => QBAR,
110                  AVDD         => AVDD,
111                  DVDD         => DVDD,
112                  IBN          => IBN2,
113                  VCM IN       => VCM2,
114                  VCM_OUT      => VCM OUT,
115                  VIN          => VI1N,
116                  VIP          => VI1P,
117                  VON          => VI2N,
118                  VOP          => VI2P,
119                  VREFN        => VREFN,
120                  VREFP        => VREFP
121                  );
122
```

cadc_ana-str.vhd 09.06.2007

```
123     int1 : entity work.CADC ANA INT1(CADC_ANA_INT1)
124       port map(CK 0        => CK(0),
125                CK 1        => CK(1),
126                CK 2        => CK(2),
127                CK 3        => CK(3),
128                CKD 0       => CKD(0),
129                CKD 1       => CKD(1),
130                CKD 2       => CKD(2),
131                CKD_3       => CKD(3),
132                PWR         => PWR,
133                Q           => Q,
134                QBAR        => QBAR,
135                SIGN_IN     => SIGN IN,
136                AVDD        => AVDD,
137                DVDD        => DVDD,
138                IBN         => IBN1,
139                VCM OUT     => VCM OUT,
140                VCM_VREF    => VCM VREF,
141                VIN         => VIN,
142                VIP         => VIP,
143                VON         => VI1N,
144                VOP         => VI1P,
145                VREFGND     => VREFGND,
146                VREFN       => VREFN,
147                VREFP       => VREFP
148
149                );
150
151     comp : entity work.CADC_ANA COMP(CADC_ANA_COMP)
152       port map(CK0        => CK(0),
153                CK2        => CK(2),
154                PWR         => PWR,
155                Q           => NET0103,
156                AVDD        => AVDD,
157                DVDD        => DVDD,
158                IBN         => IBN3,
159                VIN         => VI2N,
160                VIP         => VI2P
161                );
162
163   end architecture structural;
164
165   ----------------------------------------------------------------------
166
```

B.1.1 Op-amp 1

Figure B-4 show the folded cascade op-amp used in integrator one.

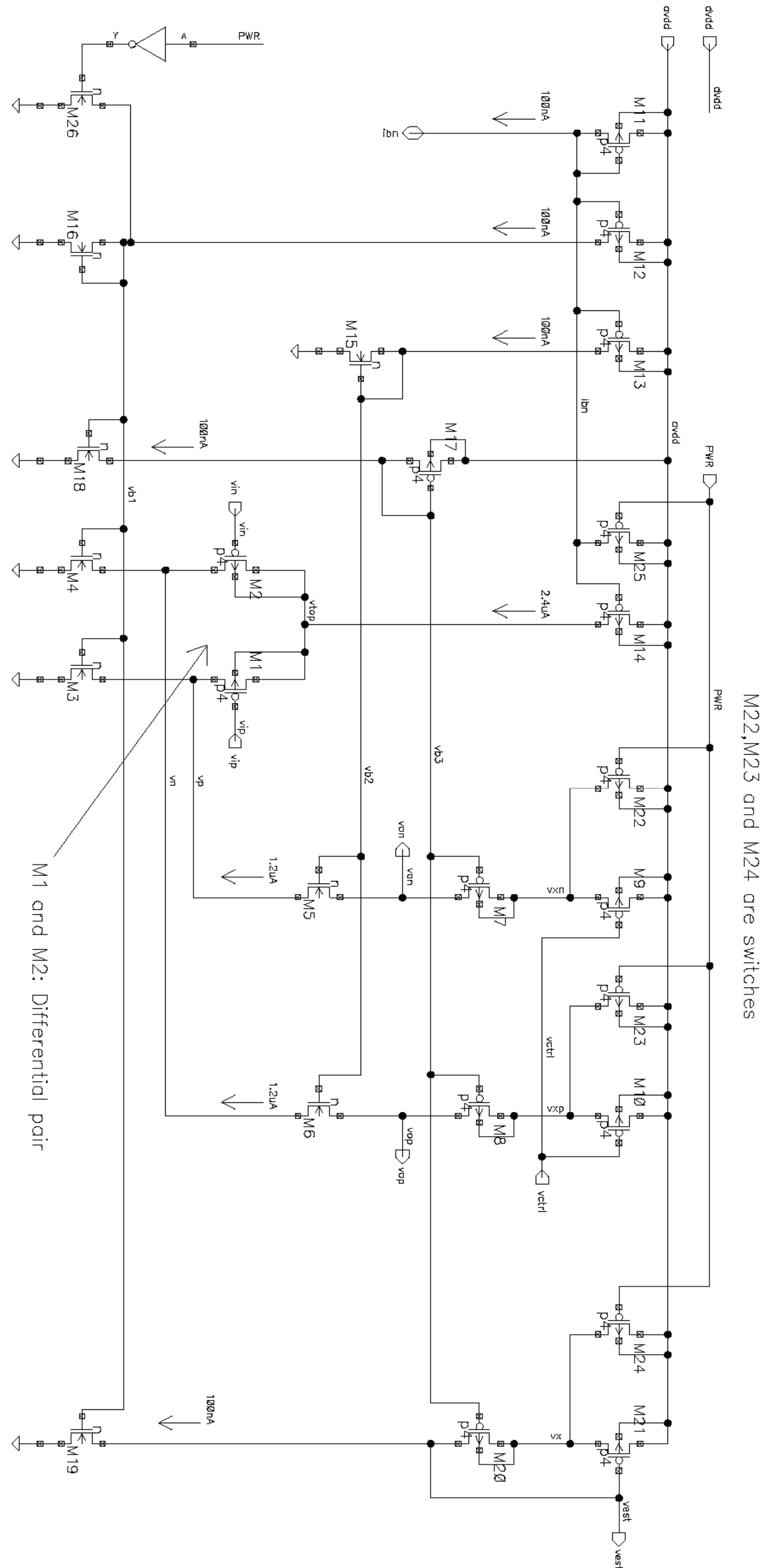

Figure B-4 Op-amp 1 schematic

The Op-amp is mapped directly as a sub circuit in the integrators SPICE model.

B.1.2 Integrator 1

Figure B-5 show integrator 1 including the op-amp shown in Figure B-4.

Figure B-5 Integrator 1 schematic

Entity to map SPICE file to VHDL-AMS:

cadc_ana_int1-ent.vhd 09.06.2007

```
 1  ----------------------------------------------------------------------
 2  -- Filename:       cadc ana int1-ent.vhd
 3  -- Description:    Integrator 1 maps cadc ana int1-ckt
 4  -- Author:         Trond Jarle Pedersen <tjpedersen@atmel.com>
 5  -- Modified by:    Trond Jarle Pedersen <tjpedersen@atmel.com>
 6  --
 7  ----------------------------------------------------------------------
 8  library IEEE;
 9  use IEEE.MATH REAL.all;
10  use IEEE.std logic 1164.all;
11  library IEEE proposed;
12  use IEEE_proposed.electrical_systems.all;
13
14  entity CADC_ANA_INT1 is
15
16    port (
17      signal   CK 0      : in std logic;
18      signal   CK 1      : in std logic;
19      signal   CK 2      : in std logic;
20      signal   CK 3      : in std logic;
21      signal   CKd 0     : in std logic;
22      signal   CKd 1     : in std logic;
23      signal   CKd 2     : in std logic;
24      signal   CKd_3     : in std logic;
25      terminal PWR       :    electrical;
26      terminal Q         :    electrical;
27      terminal Qbar      :    electrical;
28      signal   SIGN_IN   : in std logic;
29      terminal AVDD      :    electrical;
30      terminal DVDD      :    electrical;
31      terminal IBN       :    electrical;
32      terminal VCM OUT   :    electrical;
33      terminal VCM_VREF  :    electrical;
34      terminal VIN       :    electrical;
35      terminal VIP       :    electrical;
36      terminal VON       :    electrical;
37      terminal VOP       :    electrical;
38      terminal VREFGND   :    electrical;
39      terminal VREFN     :    electrical;
40      terminal VREFP     :    electrical);
41
42  end entity CADC_ANA_INT1;
43
44  -----------------------------------------------------------------------
    --------
45
```

B.1.3 Bias Circuit

Figure B-6 Show the bias circuit, it consist of a current mirror that distributes and scales bias current for the modulator.

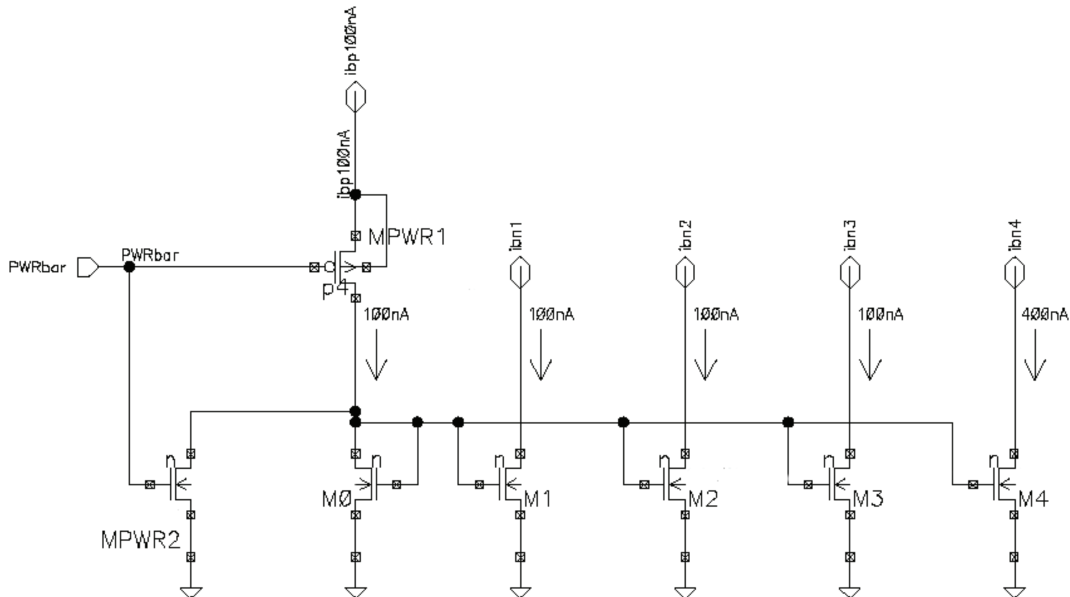

Figure B-6 Bias schematic

The transistors scale bias current to the following modulator parts:

- M1 → Integrator 1
- M2 → Integrator 2
- M3 → Comparator
- M4 → Common mode Voltage generator

Entity to map SPICE file to VHDL-AMS:

```
cadc_ana_bias-ent.vhd                                                    10.06.2007

 1  ----------------------------------------------------------------------
 2  -- Filename:       cadc ana bias-ent.vhd
 3  -- Description:    Bias generator, maps cadc ana bias.ckt
 4  -- Modified by:    Trond Jarle Pedersen <tjpedersen@atmel.com>
 5  --
 6  ----------------------------------------------------------------------
 7
 8  library IEEE;
 9  use IEEE.std logic 1164.all;
10  use IEEE.MATH REAL.all;
11  library IEEE proposed;
12  use IEEE_proposed.electrical_systems.all;
13
14  entity CADC_ANA_BIAS is
15
16    port (
17      terminal PWRBAR    :    electrical;
18      terminal IBN1      :    electrical;
19      terminal IBN2      :    electrical;
20      terminal IBN3      :    electrical;
21      terminal IBN4      :    electrical;
22      terminal IBP100NA  :    electrical
23
24      );
25
26  end entity CADC_ANA_BIAS;
27
28  ----------------------------------------------------------------------
```

B.1.4 Common mode voltage generator

Figure B-7 show the schematic for the common mode voltage generator responsible for generating and supplying the common voltage level used in the integrators.

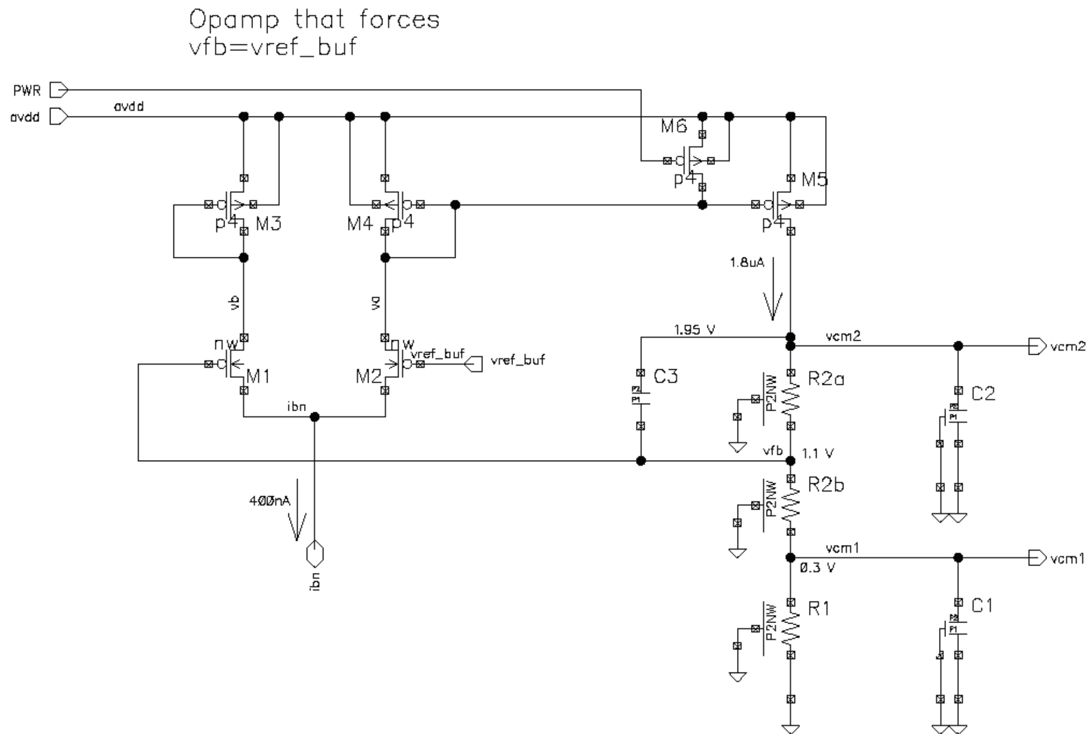

Figure B-7 Common voltage schematic

Entity to map SPICE file to VHDL-AMS:

cadc_ana_cmgen-ent.vhd 10.06.2007

```
 1  -------------------------------------------------------------------
 2  -- Filename:      cadc ana cmgen-ent.vhd
 3  -- Description:   Common mode voltage generator
 4  --                maps cadc ana cmgen.ckt
 5  -- Modified by:   Trond Jarle Pedersen <tjpedersen@atmel.com>
 6  --
 7  -------------------------------------------------------------------
 8
 9  library IEEE;
10  use IEEE.std logic_1164.all;
11  library IEEE;
12  use IEEE.MATH REAL.all;
13  library IEEE proposed;
14  use IEEE_proposed.electrical_systems.all;
15
16  entity CADC ANA_CMGEN is
17    port(terminal PWR      : electrical;
18         terminal AVDD     : electrical;
19         terminal IBN      : electrical;
20         terminal VCM1     : electrical;
21         terminal VCM2     : electrical;
22         terminal VREF_BUF : electrical
23         );
24  end entity CADC_ANA_CMGEN;
25
26  -------------------------------------------------------------------
```

B.1.5 Comparator

Figure B-8 Show the comparator responsible for sampling the output from integrator 2 and converting it to a digital value

Figure B-8 Comparator schematic

Entity to map SPICE file to VHDL-AMS:

cadc_ana_comp-ent.vhd 09.06.2007

```
 1  ----------------------------------------------------------------------
 2  -- Filename:      cadc ana comp-ent.vhd
 3  -- Description:   comparator. maps cadc ana comp.ckt
 4  -- Modified by:   Trond Jarle Pedersen <tjpedersen@atmel.com>
 5  --
 6  ----------------------------------------------------------------------
 7
 8  library IEEE;
 9  use IEEE.MATH REAL.all;
10  use IEEE.std logic 1164.all;
11  library IEEE proposed;
12  use IEEE_proposed.electrical_systems.all;
13
14  entity CADC_ANA_COMP is
15
16    port(
17      signal   CK0  : in  std logic;
18      signal   CK2  : in  std logic;
19      terminal PWR  :     electrical;
20      terminal Q    :     electrical;
21      terminal AVDD :     electrical;
22      terminal DVDD :     electrical;
23      terminal IBN  :     electrical;
24      terminal VIN  :     electrical;
25      terminal VIP  :     electrical
26      );
27
28  end entity CADC_ANA_COMP;
29
30  ----------------------------------------------------------------------
```

C Pre-study

To validate if a binary stream is able to detect modulator errors and non-idealities a SC second order ΔΣ-ADC with delaying integrators was simulated in MATLAB® (MATLAB, 2006) with both analog and digital stimuli and subjected to relevant non-ideal effects.
Key data for the modulator used during this testing is found in Table C-1

Table C-1 Matlab Modulator Key Parameters

Supply voltage V_{dd}	1.00	V
Gain block G_1	0.30	
Gain block G_2	0.30	
First integrator saturation limit	0.9	V
Second integrator saturation limit	0.9	V
Max input Voltage	± 0.5	V
Voltage reference comparator V_{ref}	0.0	v

C.1.1 Dynamic Testing

The converter was subjected to *kT/C* noise, input-referred thermal noise, op-amp noise, integrator-leakage and integrator clipping caused by changed saturation limits. The quality of the proposed test can be determined by the difference between the SNR when an analog signal is used as stimulus and the SNR when the binary stream is used as stimulus.
The modulator was simulated with the parameters shown in Table C-2 using an input amplitude of 1 Volt peak to peak when using analog stimulus and 0.5 Volt peak to peak when using binary stimulus.

Table C-2 Matlab Parameters for Simulation

Sampling frequency	3.2768	MHz
Input frequency	3.2000	kHz
Over sampling rate	512	
Sequence length	2^{16}	Samples
Supply voltage	1.0	V
Integrator saturation limit	0.9	V
Integrator gain	0.3	

The modulators output was analyzed and the output SNR was calculated. The resulting data is showed in Table C-3

Table C-3 SNRADC(simulated) vs. SNRMUT(Calculated by BIST method)

ΔΣ-Modulator Non-Ideality	SNR_{ADC} (dB)	SNR_{MUT} [2] (dB)	Deviation (dB)
Ideal Modulator	100.10	101.00	0.90
kT/C noise C= 1.0 pF	82.95	82.92	-0.03
kT/C noise C= 0.5 pF	70.00	69.95	-0.05
Input Op-amp noise 10 µV	94.60	95.89	1.29
Input Op-amp noise 100 µV	76.56	76.48	-0.08
First Integrator Leakage 1%	100.26	95.26	-5.00
First Integrator Leakage 2%	88.60	93.39	4.79
First Integrator Saturation 0.7 V (instead of 0.9 V)	52.00	65.85	13.85
First Integrator Saturation 0.6 V (instead of 0.9 V)	40.66	41.28	0.62
All of the above	41.07	34.96	-6.11

As the table shows the BIST proposal detects all degradations it was subjected to but the ability to detect distortion induced by change in saturation voltages is to some extent reduced.

This reduced sensitivity is mainly caused by the attenuation of the input signal.

C.1.2 Static Testing

To investigate if it is possible to represent a static DC input with a binary stream a test where the MUT is stimulated with a stationary DC-input and with a binary stream representing this DC-input has been developed. The Modulator was simulated and the output was filtered through a third order SINC filter

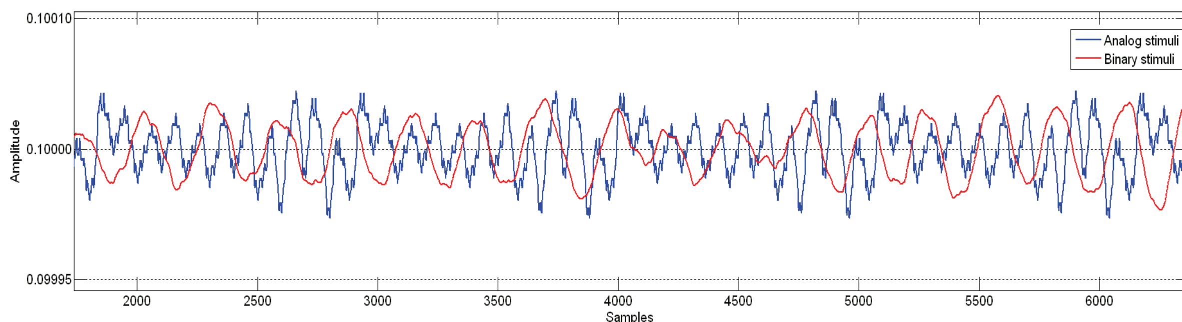

Figure C-1 Matlab Filtered Output, DC-stimuli

Figure C-1 show the filtered output from the MUT when a binary stimuli representing a DC value is used as stimuli and when the DC-stimuli is used as stimulus, And as seen in the figure it indicates that it is possible to use binary stimuli as a mean to measure offset and gain-errors.

[2] The SNR numbers for the MUT is compensated by 6 dB due to the attenuation.

C.1.3 Integrator Signal Swing

To check if the Integrators are tested over their full signal range the output from the integrators during dynamic testing was measured.

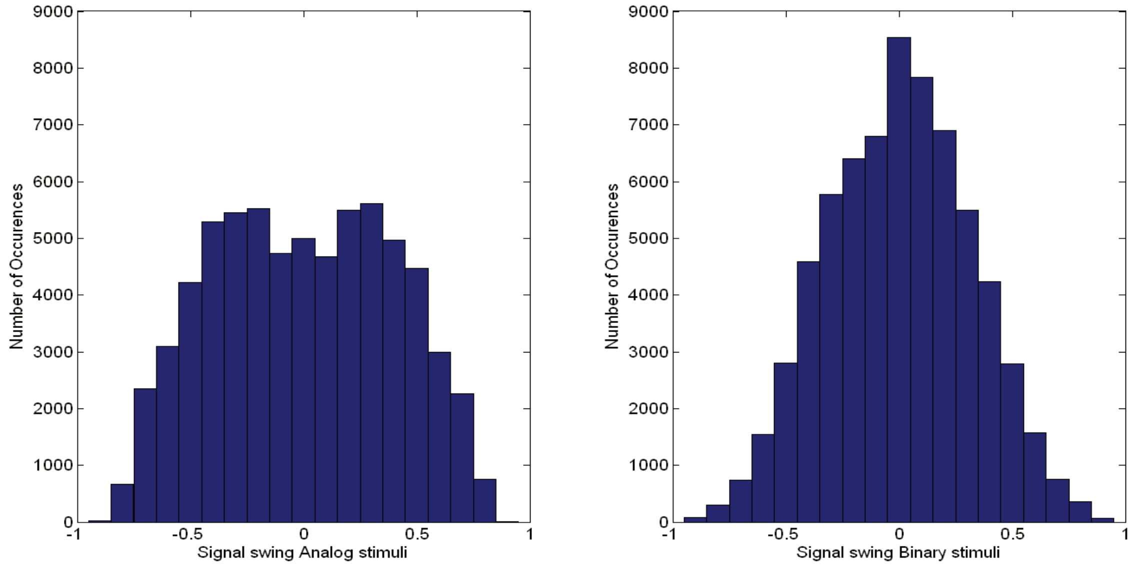

Figure C-2 Integrator swing first integrator

As Figure C-2 and Figure C-3 shows' using binary stimuli excites the integrators over the same signal range as an analog stimuli does.

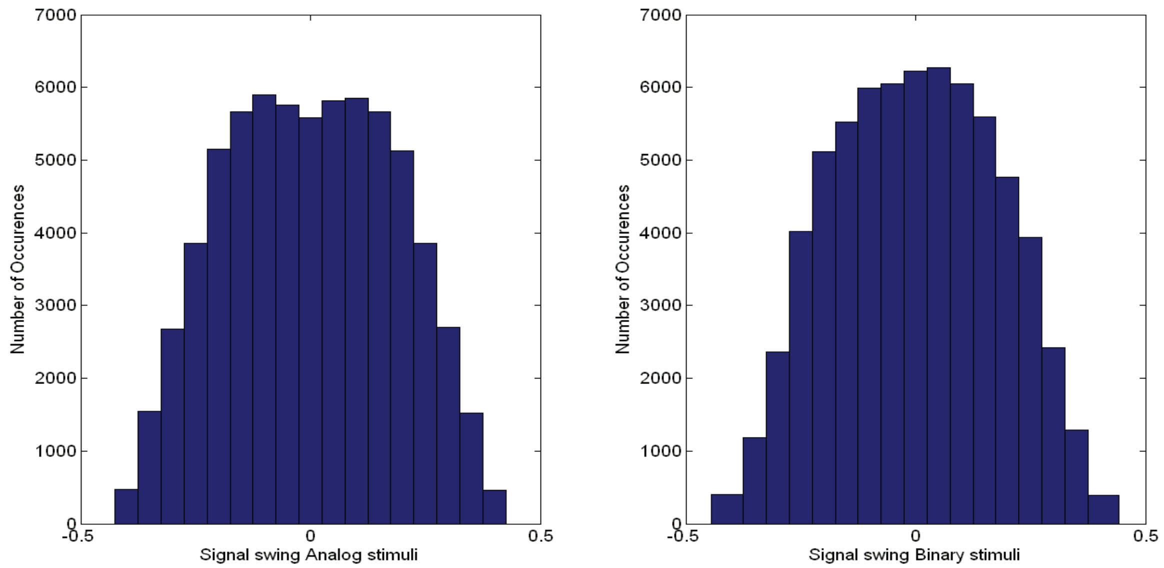

Figure C-3 Integrator swing second integrator

D TestBench

The test bench takes inn the ΔΣ-modulator described appendix B and links it together with the decimation filter and the ideal DAC used to generate test stimulus on the top level as shown in Figure D-1

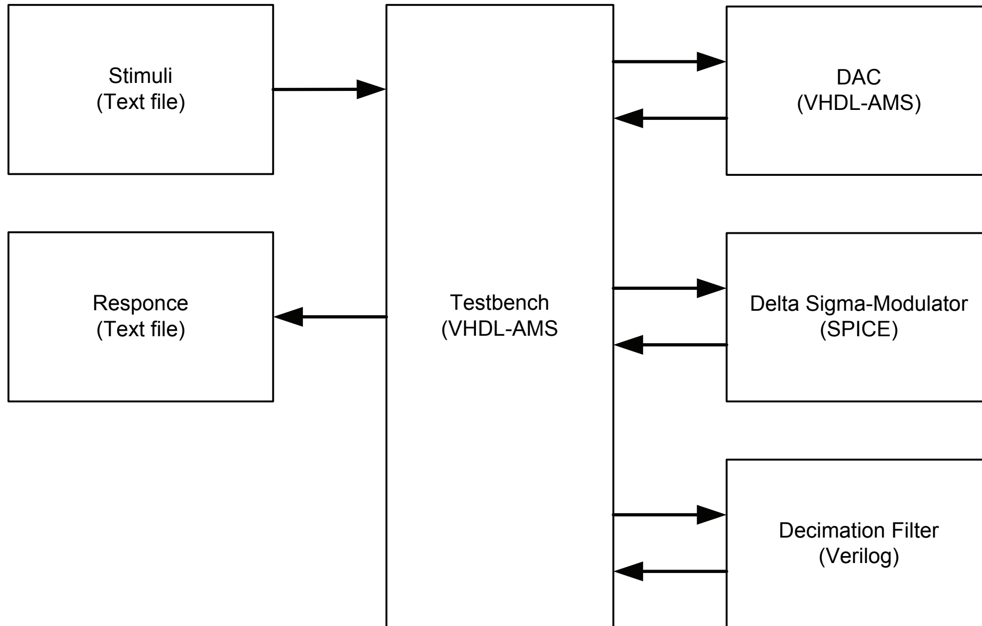

Figure D-1 Test Bench, Binary stimulus

The modulator is stimulated by an ideal DAC, the modulators output bit-stream is down sampled and filtered in the decimation filter and the result is stored in a text file for analysis in octave.

Table D-1 Key data ΔΣ-ADC

	Name		Unit
Digital Supply Voltage	DVDD	3.3	Volt
Analog Supply Voltage	AVDD	3.3	Volt
Sampling frequency	fclk	32.768	kHz
Signal Bandwidth		32	Hz
Signal range		± 220	mV
Filtered output 1	CADIC	13	bit
output 1 frequency		256	Hz
Filtered output 2	CADAC	18	bit
output 2 frequency		1-8	Hz

ENOB, SINAD, Phase, Amplitude and Offset are calculated in octave using a four parameter fit algorithm. The results are also compared to those of the fault free modulator. Each test runs two simulations:

3. Test stimulus
4. Analog stimulus 440 mV peak-to-peak

Simulations showed that to avoid overloading the modulator when using binary stimulus the amplitude needed to be reduced to ± 40 mV instead of the designed ± 220 mV. (Rolindez et al., 2006) and (Chee-Kian et al., 2004) found that a reduction in input signal by 4 would be sufficient but the special scaling of the modulator under test causes overloading earlier then that of a traditional Boser-Wooley modulator.
To compensate for this attenuation the results from binary testing is adjusted with:

$$SINAD_{LOSS} = 20 \cdot \log\left(\frac{0.220mV}{0.040mV}\right) = 14.8dB$$

Table D-2 Analog vs Test Stimulus

	Analog		Binary	
SINAD	65,64	dB	65,78	dB
ENOB	11,03	Bit	11,22	Bit
Phase	-44,96	Deg	-56,77	Deg
Amplitude[3]	0,1879	Volt	0,0259	Volt
Offset	0,0001	Volt	0,0004	Volt

As one can see using the test stimulus produce results that are close to those achieved by using an analog input signal. Selecting the DACs output amplitude as high as possible without overloading the modulators, cause the offset to be slightly higher than when using regular analog stimulus.

D.1 Testbench Code

[3] Amplitude is converted from binary representation to the corresponding voltage.

Test bench that read a digital vector from file, uses the vector as stimulus in an ideal 1-bit DAC.

```
tb_cadc_ana_s.vhd                                                  10.06.2007

  1  ----------------------------------------------------------------------
  2  -- Filename:      tb cadc ana s.vhd
  3  -- Description:   BIST testbench
  4  -- Author:        Trond Jarle Pedersen <tjpedersen@atmel.com>
  5  -- Modified by:   Trond Jarle Pedersen <tjpedersen@atmel.com>
  6  --
  7  ----------------------------------------------------------------------
  8
  9  library IEEE;
 10    use IEEE.MATH REAL.all;
 11    use IEEE.std logic 1164.all;
 12    use IEEE.Numeric STD.all;
 13    use std.textio.all;
 14  library IEEE proposed;
 15    use IEEE proposed.electrical_systems.all;
 16  library work;
 17    use work.global vdd.all;
 18    use work.txt_util.all;
 19
 20
 21
 22  entity tb cadc ana s is
 23  end entity tb_cadc_ana_s;
 24
 25  ----------------------------------------------------------------------
 26
 27  architecture testbench of tb_cadc_ana_s is
 28
 29  -- Constants
 30    constant stimuli_length : integer :=  1024;-- Test vector length
       (OSR)
 31    constant supply     : voltage := 3.3;  -- Supply voltagte
 32    constant fclk       : real    := 32.768e3; -- Sample frequency
 33
 34
 35    constant Ts         : time    :=
       time'val(integer(round(1.0e15/fclk)));
 36    constant tr         : real    := 1.0e-9; -- used to
 37    constant tf         : real    := 1.0e-9;
 38    constant Q limit    : voltage := 2.0;
 39    constant vrefp val  : voltage := 0.88;
 40    constant vrefn_val  : voltage := 0.66;
 41
 42  -- Terminals
 43    terminal vrefp         :       electrical; --Ref voltages
 44    terminal vrefn         :       electrical;
 45    terminal vrefp half    :       electrical;
 46    terminal vrefn half    :       electrical;
 47    terminal ibp100nA_1    :       electrical;
 48    terminal vref          :       electrical;
 49    terminal vref buf      :       electrical;
 50    terminal vcm vref      :       electrical;
 51    terminal Vrefp dac     :       electrical;
 52    terminal Vrefn dac     :       electrical;
 53    terminal Vrefp dac at  :       electrical;
 54    terminal Vrefn_dac_at  :       electrical;
 55    terminal avdd          :       electrical; -- Digital supply
 56    terminal dvdd          :       electrical; -- analog supply
 57    terminal Stim ana      :       electrical; -- Stimulus to ADC
 58    terminal QOUTtmp       :       electrical; -- Temp
 59  -- Decimation filter signals
```

tb_cadc_ana_s.vhd 10.06.2007

```
 60    signal CADC RESET     : std logic;
 61    signal CADAC          : std logic vector(17 downto 0);
 62    signal CADAC_COMPLETE : std logic;
 63    signal CADIC          : std logic vector(12 downto 0);
 64    signal CADIC_COMPLETE : std_logic;
 65
 66  -- Post-processing signals
 67    signal vout13bit : real := 0.0;
 68    signal vout18bit : real := 0.0;
 69
 70  -- Toplevel signals
 71    signal CLK MUX  : std logic; -- Main clock
 72    signal QOUT     : std logic; -- ADC output
 73    signal PWR CLK  : std logic; -- Ennable signal
 74    signal Stim sel : std logic; -- DAC sel
 75    signal Stim dig : bit;       -- Digital stimulus signal
 76    signal stimuli  : bit vector(Stimuli length-1 downto 0);
 77    signal Packet   : std logic_vector(Stimuli_length-1 downto 0);
 78    signal Stim init: std logic   := '0';
 79    signal vavdd s  : real := 0.0;
 80    signal vdvdd s  : real := 0.0;
 81    signal vref_buff_s :real := 0.0;
 82
 83    quantity vref buff across ireff buff  through vref buf;
 84    quantity vavdd       across iavddd      through avdd;
 85    quantity vdvdd     across idvddd      through dvdd;
 86    quantity vrefpp dac    across ivreffp dac    through vrefp dac ;
 87    quantity vrefnn dac    across ivreffn dac    through vrefn dac ;
 88    quantity vrefpp dac at across ivreffp dac at through vrefp dac at ;
 89    quantity vrefnn dac at across ivreffn_dac_at through vrefn_dac_at ;
 90    quantity vQOUTtmp across QOUTtmp;
 91    alias vss is electrical_ref;
 92
 93  begin
 94
 95  -- Stimuli-generating processe
 96   stimulus : process is
 97    begin
 98
 99      PWR CLK    <= '1';
100      Stim sel   <= '1';
101      Stim init  <= '0';
102      vavdd s    <=  0.0, supply after 1 ns;
103      vdvdd s    <=  0.0, supply after 1 ns;
104      vref buff s <= 0.0, 1.1 after 1 ns;
105      CADC_RESET <= '1'; --'0' after 1 ns;
106      wait until CLK MUX'event and CLK_MUX='0';
107      Stim_init  <= '1';
108      wait until CLK MUX'event and CLK_MUX='1';
109      Stim init  <= '1';
110      CADEN      <= '0', '1' after 1 ns;
111      CADAS      <= ('0', '0');
112      wait for 0.5 ms;
113      wait until CLK MUX'event and CLK_MUX='1';
114      Stim init  <= '0';
115      CADC_RESET <= '0';
116
117      wait;
118    end process stimulus;
119
120
```

tb_cadc_ana_s.vhd 10.06.2007

```
121  -- Clock and IO processes
122   clock : process
123      variable Ts2 : time := Ts/2.0;
124    begin
125      loop
126        if PWR_CLK = '1' then
127          loop
128            CLK MUX <= '0';
129            wait for Ts2;
130            exit when PWR CLK = '0';
131            CLK MUX <= '1';
132            wait for Ts2;
133          end loop;
134        else
135          CLK MUX <= '0';
136        end if;
137        wait on PWR_CLK;
138      end loop;
139    end process clock;
140
141
142   Read file : process is
143  -- Reads stimulus vector from file.
144    file stimulus file :text open read_mode is "stimuli_sine.dat";
145    variable input line :line;
146    variable s : string(stimuli_length downto 1);
147    begin
148      while not endfile(stimulus file) loop
149        readline(stimulus file, input_line);
150        read(input line,s);
151        Packet <= to_std_logic_vector(s);
152      end loop;
153      wait;
154    end process;
155
156
157
158   write file : process is
159   -- writes simulation results to file
160      variable v13bit : real :=0.0;
161      variable v18bit : real :=0.0;
162      variable cadic13bit : bit vector(12 downto 0);
163      variable cadac18bit : bit vector(17 downto 0);
164      variable trace line : line;
165      variable line count : natural := 0;
166      constant header : string(1 to 76)
167        := " CADIC Vector RealResult   CADAC Vector RealResult  TIME    ";
168      file log : text open write_mode is "Sim_results_dynamic.mat";
169
170    begin
171      if line_count = -1 then
172        write ( trace line, header );
173        writeline ( log, trace line );
174        writeline ( log, trace_line );     -- empty line
175      end if;
176
177      wait until CADIC_COMPLETE'event and CADIC_COMPLETE='1';
178      wait for 1ns;
179      v13bit := vout13bit;
180      v18bit := vout18bit;
181      cadic13bit := to_bitvector(CADIC);
```

tb_cadc_ana_s.vhd 10.06.2007

```
182       cadac18bit := to bitvector(CADAC);
183       write ( trace_line,to_bit(CADIC_COMPLETE),justified => left, field
          => 2);
184       write ( trace line, string'("      ") );
185       write ( trace line, cadic13bit );
186       write ( trace line, string'("      ") );
187       write ( trace_line, v13bit, field => 17,justified => left, digits
          => 15);
188       write ( trace line, string'("      ") );
189       write ( trace_line, to_bit(CADAC_COMPLETE),justified => left,
          field => 2);
190       write ( trace line, string'("       ") );
191       write ( trace line, cadac18bit );
192       write ( trace line, string'("       ") );
193       write ( trace_line, v18bit, field => 17,justified => left, digits
          => 15);
194       write ( trace line, string'("       ") );
195       writeline ( log, trace line );
196         line_count := line_count + 1;
197
198    end process write_file;
199
200    input stimulus : process(CLK MUX) is
201    -- Uses test vector read from file to generate DAC input signals.
202      begin
203          if (rising edge(CLK MUX)) then
204            if Stim init = '1' then
205              stimuli <= to_bitvector(Packet);
206            else
207              Stim dig <= stimuli(stimuli_length-1);
208              stimuli  <= stimuli rol 1;
209            end if;
210          end if;
211      end process input_stimulus;
212
213    QOUT A2D : process(CLK MUX) is
214    -- One bit comparator.
215      begin
216        if (CLK MUX'event and CLK MUX = '1') then
217          if (vQOUTtmp'above(Q_limit) ) then
218            QOUT <= '1';
219          else
220            QOUT <= '0';
221          end if;
222        end if;
223     end process QOUT_A2D;
224
225
226
227   -- Component instantiation
228     bandgap reference : entity work.topbg(behavioral)
229       port map (
230         vreg buf    => avdd,
231         ibp100nA_1 => ibp100nA_1);
232
233     resdiv : entity work.resdiv_vref(structural)
234       port map (
235         vref        => vref buf,
236         vrefp       => vrefp,
237         vrefn       => vrefn,
238         vrefp_half => vrefp_half,
```

tb_cadc_ana_s.vhd 10.06.2007

```
239         vrefn half  => vrefn half,
240         vcm vref    => vcm_vref,
241         vrefgnd     => vss );
242
243     dac :  entity work.cadc_ana_dac(behavioral)
244       port map (
245         CLK        =>      CLK MUX,
246         Stim sel   =>      Stim sel,
247         Stim dig   =>      Stim dig,
248         Vrefp      =>      Vrefp dac,
249         Vrefn      =>      Vrefn dac,
250         Vrefp at   =>      Vrefp dac at,
251         Vrefn_at   =>      Vrefn_dac_at,
252         vss        =>      vss,
253         Stim_ana   =>      Stim_ana);
254
255     dut : entity work.cadc_ana(structural)
256       port map(
257         vip       => Stim_ana,
258         vin       => vss,
259         vrefp     => vrefp,
260         vrefn     => vrefn,
261         vref buf  => vref_buf,
262         vrefgnd   => vss,
263         vcm vref  => vcm vref,
264         ibp100nA  => ibp100nA_1,
265         avdd      => avdd,
266         sign_in   => PWR CLK,
267         CLK       => CLK MUX,
268         PWR       => AVDD,
269         QOUT      => QOUTtmp,
270         DVDD       => AVDD );
271
272
273     dec filter : entity work.cadc dfilter
274       port map(cadac           => CADAC,
275               cadac_complete => CADAC COMPLETE,
276               cadic          => CADIC,
277               cadic_complete => CADIC COMPLETE,
278               caden          => CADEN,
279               cadas          => CADAS,
280               cadc di        => QOUT,
281               cp2cadc        => CLK MUX,
282               ireset         => CADC_RESET );
283
284
285   -- Post-processing
286    dac1 : process (CADIC COMPLETE) is
287    -- Generates a real number representation of the 13 bit signal
288       variable vout     : voltage;
289       variable delta v  : voltage;
290     begin  -- process dac1
291       if CADIC COMPLETE = '1' then
292         delta_v := vrefp_val-vrefn_val;
293         vout    := 0.0;
294         for index in CADIC'range loop
295           if index = CADIC'high and CADIC(CADIC'high) = '1' then
296             vout := -1.0*delta_v;
297           elsif CADIC(index) = '1' then
298             vout := vout + delta_v;
299           end if;
```

tb_cadc_ana_s.vhd 10.06.2007

```
300          delta_v := delta_v/2.0;
301        end loop;  -- index
302        vout13bit <= vout;
303      end if;
304    end process dac1;
305
306   dac2 : process (CADAC_COMPLETE) is
307   -- Generates a real number representation of the 17 bit signal
308      variable vout    : voltage;
309      variable delta_v : voltage;
310    begin  -- process dac2
311      if CADAC_COMPLETE = '1' then
312        delta_v := vrefp_val-vrefn_val;
313        vout    := 0.0;
314        for index in CADAC'range loop
315          if index = CADAC'high and CADAC(CADAC'high) = '1' then
316            vout := -1.0*delta_v;
317          elsif CADAC(index) = '1' then
318            vout := vout + delta_v;
319          end if;
320          delta_v := delta_v/2.0;
321        end loop;  -- index
322        vout18bit <= vout;
323      end if;
324    end process dac2;
325
326  -- Simultaneous equations
327    vavdd       == vavdd_s'ramp(10.0e-9,5.0e-9);
328    vdvdd       == vdvdd_s'ramp(10.0e-9,5.0e-9);
329    vref_buff   == vref_buff_s'ramp(10.0e-9,5.0e-9);
330    vrefpp_dac     == 0.22;
331    vrefnn_dac     == -0.22;
332    vrefpp_dac_at == 0.040;
333    vrefnn_dac_at == 0.040;
334
335
336  end architecture testbench;
337
338  -----------------------------------------------------------------------
     ---------
339
```

D.2 DAC Code

cadc_ana_dac-ent.vhd   09.06.2007

```
 1  ----------------------------------------------------------------------
 2  -- Filename:      cadc ana dac.vhd
 3  -- Description:   1-bit DAC with 2 reference voltage levels.
 4  -- Author      : Trond Jarle Pedersen  <tjpedersen@atmel.com>
 5  -- Modified by:   Trond Jarle Pedersen <tjpedersen@atmel.com>
 6  --
 7  ----------------------------------------------------------------------
 8
 9
10  library IEEE;
11  use IEEE.std logic 1164.all;
12  library IEEE proposed;
13    use IEEE_proposed.electrical_systems.all;
14
15  entity cadc_ana_dac is
16    port (
17      signal   CLK       : in std logic;
18      signal   Stim sel  : in std logic;
19      signal   Stim dig  : in bit;
20      terminal Vrefp     :    electrical;
21      terminal Vrefn     :    electrical;
22      terminal Vrefp at  :    electrical;
23      terminal Vrefn_at  :    electrical;
24      terminal vss       :    electrical;
25      terminal Stim_ana  :    electrical
26      );
27
28  end entity cadc_ana_dac;
29
30
```

```
cadc_ana_dac-beh.vhd                                                09.06.2007

 1  ----------------------------------------------------------------------
 2  -- File       : cadc_ana_dac-beh.vhd
 3  -- Description: Behaviour of a 1-bit DAC with two voltage reflevels.
 4  -- Author     : Trond Jarle Pedersen  <tjpedersen@atmel.com>
 5  -- Modified by:   Trond Jarle Pedersen <tjpedersen@atmel.com>
 6  --                .
 7  ----------------------------------------------------------------------
 8
 9
10  library IEEE;
11  use IEEE.std_logic_1164.all;
12  library IEEE_proposed;
13  use IEEE_proposed.electrical_systems.all;
14
15
16  architecture behavioral of cadc_ana_dac is
17    signal   vStim_tmp : real := 0.0;
18    quantity vStim_ana  across iStim_ana through Stim_ana;
19    quantity Vrefpp     across Vrefp    to vss;
20    quantity Vrefnn     across Vrefn    to vss;
21    quantity Vrefpp_at  across Vrefp_at to vss;
22    quantity Vrefnn_at  across Vrefn_at to vss;
23
24  begin
25
26    dac : process(CLK)
27      begin
28      if (CLK'event and CLK = '0') then
29        if (Stim_sel = '1') then
30          if (Stim_dig = '1') then
31            vStim_tmp <= Vrefpp;
32          else
33            vStim_tmp <= Vrefnn;
34          end if;
35        else
36          if (Stim_dig = '0') then
37            vStim_tmp <= Vrefpp_at;
38          else
39            vStim_tmp <= Vrefnn_at;
40          end if;
41        end if;
42      end if;
43
44    end process dac;
45
46    vStim_ana == vStim_tmp;
47
48  end architecture behavioral;
49
```

D.3 Octave Scripts

D.3.1 Four Parameter Sine Wave Fitting

```
1   % Process simulation data and calculate SINAD and Resolution
2
3   % Load reference
4   load -force oct/ref.txt;
5
6   %  Load simulation results
7   Sim results dynamic       = load Sim results dynamic.mat;
8   Sim_results_sine          = load Sim_results_sine.mat ;
9
10
11  L = length(Sim_results dynamic)-5;
12  cadic(1:L,1)  = Sim results dynamic(5:L,2);
13  cadic(1:L,2)  = Sim_results_sine(5:L,2);
14
15  % Perform a four param fitting.
16  for h = 1:2
17    tmp = cadic(1:L,h);
18    out data = fourParam fit(tmp);
19    ENOB(h)          = out data(1);
20    SINAD(h)         = out data(2);
21    Resolution(h)    = out data(3);
22    phi(h)           = out data(4);
23    eps er(h)        = out data(5);
24    amp(h )          = out data(6);
25    DC(h)            = out_data(7);
26  end
27
28  % Compansate the test for loss in amplitude.
29  SINAD(1)= SINAD(1) + 14.807253789885;
30
31  for h=1:2
32    ENOB diff(h)           = ((ENOB(h)- ENOB ref(h))/ENOB ref(h)) *100;
33    SINAD diff(h)          = ((SINAD(h)-SINAD ref(h))/SINAD ref(h)) *100;
34    Resolution diff(h)     = ((Resolution(h)-Resolution_ref(h))/
      Resolution ref(h)) *100;
35    phi diff(h)            = ((phi(h)-phi ref(h))/phi ref(h))*100;
36    eps er diff(h)         = ((eps_er(h)-eps_er_ref(h))/eps_er_ref(h)) *100;
37    if h==1
38      amp_diff(h)          = ((amp(h)-amp_ref(h))/amp_ref(h)) *100;
39    else
40      amp_diff(h)          = ((amp(h)-amp_ref(h))/amp_ref(h)) *100;
41    end
42    DC_diff(h)             = ((DC(h)-DC_ref(h))/DC_ref(h)) *100;
43  end
44
45  filename = ('Simulation results.txt');
46  fid = fopen(filename, 'w');
47  fprintf(fid,
    '-----------------------------------------------------------------------
    ----------\n');
48  fprintf(fid, 'Simulation results binary vs. analog stimuli\n');
49  fprintf(fid,
    '-----------------------------------------------------------------------
    ----------\n \n');
50  fprintf(fid, 'Binary stimuli \n')
51  fprintf(fid,
    '-----------------------------------------------------------------------
    ----------\n');
52  fprintf(fid,
    'ENOB:          %2.2f\tvs  %2.2f \tBIT   diff: %2.2f\t%2.2f\tpercent
    \n', ENOB(1), ENOB_ref(1),(ENOB(1)-ENOB_ref(1)), ENOB_diff(1));
53  fprintf(fid,
    'SINAD:         %2.2f\tvs  %2.2f \tdB    diff: %2.2f\t%2.2f\tpercent
    \n', SINAD(1), SINAD_ref(1),(SINAD(1)-SINAD_ref(1)), SINAD_diff(1));
54  fprintf(fid,
    'Resolution:    %2.2f\tvs  %2.2f \t      diff: %2.2f\t%2.2f\tpercent
    \n', Resolution(1),
    Resolution ref(1),(Resolution(1)-Resolution_ref(1)),
    Resolution diff(1));
55  fprintf(fid,
    'Phase:         %2.2f\tvs  %2.2f \tDeg   diff: %2.2f\t%2.2f\tpercent
    \n', phi(1), phi_ref(1),(phi(1)-phi_ref(1)), phi_diff(1));
56  fprintf(fid,
    'Amplitude:     %2.4f\tvs  %2.4f \tV     diff: %2.4f\t%2.1f\tpercent
    \n', amp(1), amp_ref(1),(amp(1)-amp_ref(1)), amp_diff(1));
57  fprintf(fid,
    'DC offset:     %2.4f\tvs  %2.4f \tV     diff: %2.4f\t%2.1f\tpercent
```

```
57  'DC offset:      %2.4f\tvs  %2.4f \tV     diff: %2.4f\t%2.1f\tpercent
    \n', DC(1), DC_ref(1),(DC(1)-DC_ref(1)),DC_diff(1));
58
59  fprintf(fid, '\nSine wave stimuli 440mv p-2-p ')
60  fprintf(fid,
    '\n------------------------------------------------------------------
    ------------\n');
61  fprintf(fid,
    'ENOB:          %2.2f\tvs %2.2f \tBIT   diff: %2.2f\t%2.2f\tpercent
    \n', ENOB(2), ENOB_ref(2),(ENOB(2)-ENOB_ref(2)), ENOB_diff(2));
62  fprintf(fid,
    'SINAD:         %2.2f\tvs %2.2f \tdB    diff: %2.2f\t%2.2f\tpercent
    \n', SINAD(2), SINAD_ref(2),(SINAD(2)-SINAD_ref(2)), SINAD_diff(2));
63  fprintf(fid,
    'Resolution:    %2.2f\tvs %2.2f \t      diff: %2.2f\t%2.2f\tpercent
    \n', Resolution(2),
    Resolution ref(2),Resolution(2)-(Resolution_ref(2)),
    Resolution diff(2));
64  fprintf(fid,
    'Phase:         %2.2f\tvs %2.2f \tDeg   diff: %2.2f\t%2.2f\tpercent
    \n', phi(2), phi_ref(2),(phi(2)-phi_ref(2)), phi_diff(2));
65  fprintf(fid,
    'Amplitude:     %2.4f\tvs %2.4f \tV     diff: %2.4f\t%2.1f\tpercent
    \n', amp(2), amp_ref(2),(amp(2)-amp_ref(2)), amp_diff(2));
66  fprintf(fid,
    'DC offset:     %2.4f\tvs %2.4f \tV     diff: %2.4f\t%2.1f\tpercent
    \n', DC(2), DC_ref(2),(DC(2)-DC_ref(2)),DC_diff(2));
67
68  fclose(fid);
69
70  save Simulation_variables.txt cadic ENOB SINAD Resolution phi eps_er
    amp DC;
71
72
73
```

D.3.1.1 Four Parameter Function

```
 1  %  Load and process simulation results
 2  function out_data = fourParam_fit(data);
 3
 4  % Function to calculate a 4-param sine wave fitting.
 5  % In accordance to the IEEE standard.
 6  % Hardcoded to fit BIST procjet.
 7
 8
 9  % Declear variables.
10    L = length(data);
11    Sample time = 0;
12    Sample rate = 0;
13    fullscale = 0.44;
14
15  %          % Parse 13bit
16  % Reads a 13-bit twos' complement vector, parses it to analog and
17  % Scales it to the repective input voltages
18    for i = 1:L
19      tmp = int2str(data(i));
20      sign = 1;
21      if length(tmp) == 13
22        if tmp(1) == '1'
23          data(i)= -4096 + bin2dec(tmp(2:13));
24        end
25      else
26        data(i)= bin2dec(tmp);
27      end
28    end
29
30    for i=1:L
31      data(i) = (data(i)/4096)*0.22;
32    end
33
34  % Starts by findin intial frequency by FFT
35    Fw = fft(data(1:81,1));
36    Fw = abs(Fw);
37    L = length(Fw);
38    [Fw est,w] = max(Fw(2:round(L/2)));
39    freq = pi*w;
40    w = 2*pi*w/(L*Sample rate);
41    t= (0:L-1)*1/Sample_time;
42
43  % Create fit matrix.
44    fit = [cos(w*t);sin(w*t);ones(1,L)]';
45    x0 = fit\data;
46    x1= fit\data;
47    x0(4)=0;
48
49    i=0;
50  % Loop conditions,
51  % Breaks loop if fit or maximum itterations is acheived.
52    Max err = 2*pi*1e-6;
53    Max cycles = 100;
54  % Perform fitting
55    while 1
56      i=i+1;
57      fit=[cos(w*t);sin(w*t);ones(1,L);-x0(1)*t.*sin(w*t)+x0(2)*t.*cos(w*
        t)]';
58      x0=fit\data;
59
60      if abs(x0(4))<0.05*w,
61        w=w+x0(4);
62      else
63        w=w+.05*x0(4);
64      end
65      if abs(x0(4))<Max_err/Sample_rate
66        break;
67      end;
68      if i>=Max_cycles
69        break;
70      end
71    end
72
73  % Calculate the Residue vector, RMS error, Amplitude
74  % DC-offset, Phase.ENOB SINAD and Resolution
75
```

```
76     Residue = data - (x0(1)*cos(w*t)+x0(2)*sin(w*t)+x0(3))';
77     eps_er=((1/L)*Residue'*Residue)**(1/2);
78     amp = (x0(1)**2 +x0(2)**2)**(1/2);
79     DC = x0(3);
80
81     if x0(1) >= 0
82       phi = atan(-x0(2)/x0(1));
83     else
84       phi = atan(-x0(2)/x0(1))+ pi;
85     end
86     phi = phi*180/pi;
87     ENOB  = log2(fullscale/(eps_er*(12**(1/2))));
88     SINAD = 1/100*(100*20*log10((amp/(2**(1/2)))/eps_er));
89     Resolution= 2**ENOB;
90
91     out_data = [ENOB, SINAD, Resolution, phi, eps_er, amp, DC];
92
93   end
```

D.3.2 BIST octave code

```
1   % Code to calculate SINAD by a simplified sine wave fitt.
2   % by comparing to a reference signal.
3   %
4
5   %  Load simulation results and the signature it is to be compared to.
6   Sim results dynamic = load Sim results dynamic.mat;
7   Sim_results_signature = load Sim_results_signature.mat ;
8
9
10  L= length(Sim results dynamic)-5;
11  period = 5; % number of periods used in calculation.
12  samples = 8; % number of samples pr period.
13
14  cadic(1:L,1) = Sim results signature(1:L,2);
15  cadic(1:L,2) = Sim_results_dynamic(1:L,2);
16
17
18  %          % Parse 13bit
19  for h=1:2
20    for i = 1:L
21      tmp = int2str(cadic(i,h));
22      sign = 1;
23      if length(tmp) == 13
24        if tmp(1) == '1'
25          cadic(i,h)= -4096 + bin2dec(tmp(2:13));
26        end
27      else
28        cadic(i,h)= bin2dec(tmp);
29      end
30    end
31  end
32
33
34
35
36  offset = 0;
37  for i=5:L       % % find start of valid signals.
38      if cadic(i,2) >=0
39          for j=i:L
40              if cadic(j,2) <=0
41                  offset = j;
42                  break;
43              end
44          end
45          if offset ~= 0
46              break
47          end
48      end
49      if offset ~= 0
50          break
51      end
52  end
53
54
55  % set new length to record 5 periods.
56  L = period * samples;
57  cadic= cadic(offset:L+offset-1,1:6);
58
59  % Calculate Offset
60  % Possible to use this simplification due to the coherent sampling.
61  DC = zeros(1,6);
62  for h=1:2
63      for i=1:L
64          DC(1,h) = DC(1,h)+cadic(i,h);
65      end
66      DC(1,h) = DC(1,h) /L;
67  end
68
69  % Calculate Amplitude
70  % Possible to use this simplification due to the coherent sampling.
71  Amp = zeros(1,2);
72  for i=1:2
73    Amp(1,i) = (max(cadic(1:L,i)) - min(cadic(1:L,i))) /2;
74  end
75
76  for i=1:L
```

```
77       cadic fit(i,2)  = (Amp(1,2)/Amp(1,2)) * cadic(i,2) + DC(1,2)-
         DC(1,2) ;
78   end
79
80
81   Error power = zeros(1,3);
82   Error_dB    = zeros(1,3);
83
84   % Calculate error vector
85   for i=1:L
86       cadic error(i,1) = (cadic(i,2)-cadic fit(i,1))**2;
87       Error_power(1,1) = Error_power(1,2) + cadic_error(i,1);
88     end
89   end
90
91
92   % calculate RMS error
93   Error_power(1,i) = (Error_power(1,i)/L)**(1/2);
94
95   % Calculate ENOB
96   ENOB(1,i) = 12 - log2(Error_power(1,i)/Error_q);
97
98   % Calculate SIAND
99   SINAD(1,i) =  1/100*(100*20*log10((Amp(1,i+3)/(2**(1/2)))/Error_power(
     1,i)));
100
101  % Compensate SIAND for attenuation
102  SINAD(1,1)=SINAD(1,1)+14.807253;
103
104  save process results.txt Error power DC Amp ENOB Error_q SINAD;
105  save process_variables.txt cadic cadic_fit cadic_error Error_power DC
     Amp;
106
```

E Arithmetic Calculations

Performing the proposed test scheme requires that the microcontroller is capable to perform simple arithmetic operations between each filtered output. All calculations are based on an Atmel AVR© 8-bit microcontroller using the hardware 8-bit multiplier. Calculations are based on (Atmel, 2002)

Calculating gain and offset on the fly and simultaneously in this case requires one 13*15-bit signed multiplication and two 16-bit summations which would call for 23 clock cycles.	$Y_{DC,ACU} = \sum_{i=1}^{M} y(i)$ $Y_{AMP,ACU} = \sum_{i=1}^{M} y(i) \cdot y_{ref}(i)$
The last step will require one 32-bit signed division for offset calculation and one 32-bit signed division for gain calculation, each call for up to 640 clock cycles. Note that $M*Gain_{ref}$ is known and can be calculated earlier.	$Offset = \frac{Y_{DC,ACU}}{M}$ $Gain = \frac{2 \cdot Y_{AMP,ACU}}{M \cdot Gain_{ref}}$
Creating the fitted reference sine wave requires one unsigned multiplication and on signed summation. Requiring up to 16 cycles.	$y_{ref,fitted}(i) = \frac{Gain}{Gain_{ref}} y_{ref}(i) + Offset$
Calculating the noise demands one signed multiplication and one signed subtraction Requiring up to 4 clock cycles.	$N_{error} = \sum_{i=1}^{M} y(i) - y_{ref,fitted}(i)$
To calculate the *RMS* noise value an 32-bit division requiring up to 640 clock cycles. And a square root However performing square root calculations is next to impossible in microcontrollers.	$Noise_{RMS} = \sqrt{\frac{N_{error}}{M}}$

Calculating the Signals *RMS* value requires a division and a square root.	$Signal_{RMS} = \frac{S_A}{\sqrt{2}}$
By squaring the Signal amplitude instead of extracting the square root of the noise a possible solution disclose it self.	$SINAD = \frac{\left(\frac{S_A}{\sqrt{2}}\right)^2}{\left(\sqrt{\frac{N_{error}}{M}}\right)^2} = \frac{S_A^2 \cdot M}{2 \cdot N_{error}}$
By calculating the *tmp* variable in advance,	$tmp = \frac{S_A^2 \cdot M}{2}$
The final SINAD calculation would only require a signed 32-bit division.	$SINAD = \frac{tmp}{N_{error}}$

These requirements make signal analysis impossible to perform on the fly if the microcontroller runs synchronous with the ΔΣ-ADC. However a CPU frequency of 350 kHz or more allows on-chip signal analysis.

F Simulation results Binary signals as stimuli

F.1 Simulation results Integrator 1

F.1.1 Simulation results Op-amp

ΔΣ-ADC performance with deviations in transistor M2:

Table F-1 Performance with deviations introduced in Op-amp 1 transistor M2

Deviation Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
Leff +20%	Test	65,07	-0,71	0,0259	0,0
	Analog	63,72	-0,92	0,1879	0,0
Leff -20%	Test	66,51	0,73	0,0259	0,0
	Analog	64,62	-0,02	0,1879	0,0
Weff -20%	Test	66,43	0,65	0,0259	0,0
	Analog	64,74	0,10	0,1879	0,0
GS-short	Test	10,33	-55,45	0,0001	-99,6
	Analog	11,67	-52,97	0,0003	-99,8
GD-short	Test	33,00	-32,78	0,0002	-99,2
	Analog	31,83	-32,81	0,0004	-99,8
Stuck Open	Test	-1,40	-67,18	0,0000	-100,0
	Analog	-2,32	-66,96	0,0000	-100,0
Stuck Close	Test	-7,02	-72,80	0,0000	-100,0
	Analog	-2,16	-66,80	0,0000	-100,0

ΔΣ-ADC performance with deviations in transistor M6:

Table F-2 Performance with deviations introduced in Op-amp 1 transistor M6

Deviation Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
Leff +20%	Test	65,82	0,04	0,0259	0,0
	Analog	64,69	0,05	0,1879	0,0
Leff -20%	Test	47,59	-18,19	0,0056	-78,4
	Analog	39,69	-24,95	0,0272	-85,5
Weff -20%	Test	45,53	-20,25	0,0055	-78,8
	Analog	32,48	-32,16	0,0268	-85,7
GS-short	Test	65,72	-0,06	0,0259	0,0
	Analog	64,36	-0,28	0,2058	9,5
GD-short	Test	21,73	-44,05	0,0001	-99,6
	Analog	15,94	-48,70	0,0001	-99,9
Stuck Open	Test	63,00	-2,78	0,0231	-10,8
	Analog	62,12	-2,52	0,1467	-21,9
Stuck Close	Test	-2,93	-68,71	0,0000	-100,0
	Analog	-0,04	-64,68	0,0000	-100,0

ΔΣ-ADC performance with deviations in transistor M8:

Table F-3 Performance with deviations introduced in Op-amp 1 transistor M8

Deviation Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
Leff +20%	Test	65,90	0,12	0,0259	0,0
	Analog	65,05	0,41	0,1879	0,0
Leff -20%	Test	65,88	0,10	0,0259	0,0
	Analog	64,85	0,21	0,1880	0,1
Weff -20%	Test	65,73	-0,05	0,0259	0,0
	Analog	64,65	0,01	0,1879	0,0
GS-short	Test	65,48	-0,30	0,0259	0,0
	Analog	64,46	-0,18	0,1881	0,1
GD-short	Test	-0,34	-66,12	0,0000	-100,0
	Analog	-4,36	-69,00	0,0002	-99,9
Stuck Open	Test	66,12	0,34	0,0259	0,0
	Analog	64,95	0,31	0,1879	0,0
Stuck Close	Test	42,98	-22,80	0,0046	-82,2
	Analog	35,84	-28,80	0,0271	-85,6

ΔΣ-ADC performance with deviations in transistor M14:

Table F-4 Performance with deviations introduced in Op-amp 1 transistor M14

Flaw Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
Leff +20%	Test	65,67	-0,11	0,0259	0,0
	Analog	64,30	-0,34	0,1879	0,0
Leff -20%	Test	0,46	-65,32	0,0000	-100,0
	Analog	-3,64	-68,28	0,0003	-99,8
Weff -20%	Test	65,25	-0,53	0,0259	0,0
	Analog	64,37	-0,27	0,1879	0,0
GS-short	Test	0,07	-65,71	0,0000	-100,0
	Analog	-4,49	-69,13	0,0002	-99,9
GD-short	Test	21,22	-44,56	0,0062	-76,0
	Analog	22,09	-42,55	0,0212	-88,7
Stuck Open	Test	1,47	-64,31	0,0000	-100,0
	Analog	1,47	-63,17	0,0006	-99,7
Stuck Close	Test	133,13	67,35	0,0000	-100,0
	Analog	126,35	61,71	0,0007	-99,6

ΔΣ-ADC performance with deviations in transistor M18:

Table F-5 Performance with deviations introduced in Op-amp 1 transistor M18

Deviation Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
Leff +20%	Test	65,90	0,12	0,0259	0,0
	Analog	65,05	0,41	0,1879	0,0
Leff -20%	Test	65,88	0,10	0,0259	0,0
	Analog	64,85	0,21	0,1880	0,1
Weff -20%	Test	65,73	-0,05	0,0259	0,0
	Analog	64,65	0,01	0,1879	0,0
GS-short	Test	65,48	-0,30	0,0259	0,0
	Analog	64,46	-0,18	0,1881	0,1
GD-short	Test	-0,34	-66,12	0,0000	-100,0
	Analog	-4,36	-69,00	0,0002	-99,9
Stuck Open	Test	66,12	0,34	0,0259	0,0
	Analog	64,95	0,31	0,1879	0,0
Stuck Close	Test	42,98	-22,80	0,0046	-82,2
	Analog	35,84	-28,80	0,0271	-85,6

F.1.2 Simulation results Integrator

ΔΣ-ADC performance with deviations in transistor M7:

Table F-6 Performance with deviations introduced in Integrator 1 transistor M7

Deviation Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
Leff +20%	Test	65,58	-0,20	0,0259	0,0
	Analog	64,52	-0,12	0,1880	0,1
Leff -20%	Test	65,59	-0,19	0,0259	0,0
	Analog	64,36	-0,28	0,1879	0,0
Weff -20%	Test	65,64	-0,14	0,0259	0,0
	Analog	64,51	-0,13	0,1879	0,0
GS-short	Test	-1,45	-67,23	0,0000	-100,0
	Analog	0,13	-64,51	0,0000	-100,0
GD-short	Test	-1,20	-66,98	0,0000	-100,0
	Analog	-3,87	-68,51	0,0000	-100,0

ΔΣ-ADC performance with deviations in transistor N9:

Table F-7 Performance with deviations introduced in Integrator 1 transistor N9

Deviation Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
Leff +20%	Test	65,15	-0,63	0,0259	0,0
	Analog	63,92	-0,72	0,1879	0,0
Leff -20%	Test	65,68	-0,10	0,0259	0,0
	Analog	64,55	-0,09	0,1880	0,1
Weff -20%	Test	65,85	0,07	0,0259	0,0
	Analog	64,78	0,14	0,1879	0,0
GS-short	Test	4,93	-60,85	0,0052	-80,0
	Analog	5,64	-59,00	0,0161	-91,4
GD-short	Test	32,91	-32,87	0,0002	-99,2
	Analog	32,46	-32,18	0,0004	-99,8

ΔΣ-ADC performance with deviations in transistor N10:

Table F-8 Performance with deviations introduced in Integrator 1 transistor N10

Deviation Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
Leff +20%	Test	65,67	-0,11	0,0259	0,0
	Analog	64,65	0,01	0,1879	0,0
Leff -20%	Test	65,34	-0,44	0,0259	0,0
	Analog	64,17	-0,47	0,1878	-0,1
Weff -20%	Test	65,49	-0,29	0,0260	0,4
	Analog	64,42	-0,22	0,1879	0,0
GS-short	Test	0,33	-65,45	0,0000	-100,0
	Analog	1,67	-62,97	0,0000	-100,0
GD-short	Test	6,93	-58,85	0,0052	-80,0
	Analog	5,64	-59,00	0,0161	-91,4

ΔΣ-ADC performance with deviations in transistor N18:

Table F-9 Performance with deviations introduced in Integrator 1 transistor N18

Deviation Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
Leff +20%	Test	66,22	0,44	0,0259	0,0
	Analog	64,96	0,32	0,1879	0,0
Leff -20%	Test	65,85	0,07	0,0259	0,0
	Analog	64,92	0,28	0,1878	-0,1
Weff -20%	Test	65,30	-0,48	0,0260	0,4
	Analog	64,39	-0,25	0,1879	0,0
GS-short	Test	-0,07	-65,85	0,0000	-100,0
	Analog	0,13	-64,51	0,0000	-100,0
GD-short	Test	-0,54	-66,32	0,0000	-100,0
	Analog	-0,16	-64,80	0,0000	-100,0

F.2 Simulation Results Common Mode Voltage Generator

ΔΣ-ADC performance with deviations in transistor M2:

Table F-10 Performance with deviations introduced in CMgen transistor M2

Deviation Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
Leff +20%	Test	65,69	-0,09	0,0259	0,0
	Analog	64,86	0,22	0,1879	0,0
Leff -20%	Test	65,30	-0,48	0,0259	0,0
	Analog	64,28	-0,36	0,1879	0,0
Weff -20%	Test	65,55	-0,23	0,0259	0,0
	Analog	64,27	-0,37	0,1878	-0,1

F.3 Simulation Results Bias Circuit

ΔΣ-ADC performance with deviations in transistor M1:

Table F-11 Performance with deviations introduced in Bias transistor M1

Deviation Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
Leff +20%	Test	65,76	-0,02	0,0259	0,0
	Analog	65,50	0,86	0,1879	0,0
Leff -20%	Test	65,61	-0,17	0,0259	0,0
	Analog	64,50	-0,14	0,1879	0,0
Weff -20%	Test	65,74	-0,04	0,0259	0,0
	Analog	64,65	0,01	0,1879	0,0

ΔΣ-ADC performance with deviations in transistor M2:

Table F-12 Performance with deviations introduced in Bias transistor M2

Deviation Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
Leff +20%	Test	39,50	-26,28	0,0035	-86,5
	Analog	31,45	-33,19	0,0169	-91,0
Leff -20%	Test	65,61	-0,17	0,0259	0,0
	Analog	64,42	-0,22	0,1879	0,0
Weff -20%	Test	65,93	0,15	0,0259	0,0
	Analog	64,82	0,18	0,1879	0,0

ΔΣ-ADC performance with deviations in transistor M3:

Table F-13 Performance with deviations introduced in Bias transistor M3

Deviation Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
Leff +20%	Test	66,29	0,51	0,0259	0,0
	Analog	64,72	0,08	0,1879	0,0
Leff -20%	Test	66,12	0,34	0,0259	0,0
	Analog	64,92	0,28	0,1879	0,0
Weff -20%	Test	66,14	0,36	0,0259	0,0
	Analog	64,74	0,10	0,1879	0,0

ΔΣ-ADC performance with deviations in transistor M4:

Table F-14 Performance with deviations introduced in Bias transistor M4

Deviation Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
Leff +20%	Test	65,75	-0,03	0,0259	0,0
	Analog	64,46	-0,18	0,1879	0,0
Leff -20%	Test	66,29	0,51	0,0259	0,0
	Analog	65,09	0,45	0,1879	0,0
Weff -20%	Test	65,32	-0,46	0,0257	-0,8
	Analog	64,11	-0,53	0,1871	-0,4

F.4 Simulation Results Comparator

ΔΣ-ADC performance with deviations in transistor M2:

Table F-15 Performance with deviations introduced in comparator transistor M2

Deviation Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
Leff +20%	Test	65,97	0,19	0,0259	0,0
	Analog	64,77	0,13	0,1879	0,0
Leff -20%	Test	65,68	-0,10	0,0259	0,0
	Analog	64,56	-0,08	0,1879	0,0
Weff -20%	Test	65,44	-0,34	0,0259	0,0
	Analog	64,49	-0,15	0,1879	0,0

ΔΣ-ADC performance with deviations in transistor M4:

Table F-16 Performance with deviations introduced in comparator transistor M4

Deviation Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
Leff +20%	Test	65,39	-0,39	0,0259	0,0
	Analog	64,45	-0,19	0,1879	0,0
Leff -20%	Test	65,78	0,00	0,0259	0,0
	Analog	64,63	-0,01	0,1879	0,0
Weff -20%	Test	65,82	0,04	0,0259	0,0
	Analog	64,68	0,04	0,1879	0,0

ΔΣ-ADC performance with deviations in transistor M8:

Table F-17 Performance with deviations introduced in comparator transistor M8

Deviation Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
Leff +20%	Test	65,74	-0,04	0,0259	0,0
	Analog	64,88	0,24	0,1879	0,0
Leff -20%	Test	65,80	0,02	0,0259	0,0
	Analog	64,65	0,01	0,1879	0,0
Weff -20%	Test	65,13	-0,65	0,0253	-2,3
	Analog	64,12	-0,52	0,1852	-1,4

ΔΣ-ADC performance with deviations in transistor M8:

Table F-18 Performance with deviations introduced in comparator transistor M13

Deviation Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
Leff +20%	Test	65,98	0,20	0,0259	0,0
	Analog	64,70	0,06	0,1879	0,0
Leff -20%	Test	66,04	0,26	0,0259	0,0
	Analog	65,03	0,39	0,1879	0,0
Weff -20%	Test	66,02	0,24	0,0259	0,0
	Analog	64,73	0,09	0,1879	0,0

G BIST Proposal. Dynamic Simulation results

G.1 BIST Simulation results Integrator 1

G.1.1 BIST Simulation results Op-amp, Integrator 1

Table G-1 BIST Performance with deviations introduced in Op-amp, Integrator 1

Transistor tested	Deviation Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
M2	Leff +20%	BIST	64,74	-0,77	0,0235	0,01
		TEST	65,02	-0,82	0,0254	0,00
		Analog	65,31	-0,95	0,1847	0,01
	GD-short	BIST	7,12	-58,39	0,0001	-99,57
		TEST	6,70	-59,14	0,0001	-99,72
		Analog	8,54	-57,72	0,0008	-99,57
M6	Leff -20%	BIST	49,21	-16,30	0,0053	-77,44
		TEST	51,12	-14,72	0,0051	-80,04
		Analog	55,17	-11,09	0,0273	-85,22
	Weff -20%	BIST	42,52	-22,99	0,0057	-75,74
		TEST	43,11	-22,73	0,0061	-75,98
		Analog	31,12	-35,14	0,0267	-85,54
	GS-short	BIST	65,11	-0,40	0,0212	-9,77
		TEST	65,66	-0,18	0,0224	-11,81
		Analog	65,42	-0,84	0,1446	-21,71
	Stuck Open	BIST	43,31	-22,20	0,0209	-11,22
		TEST	45,37	-20,47	0,0224	-11,81
		Analog	31,99	-34,27	0,1446	-21,71
M8	Leff -20%	BIST	65,53	0,02	0,0249	5,98
		TEST	66,45	0,61	0,0259	1,97
		Analog	66,23	-0,03	0,1874	1,46
	GS-short	BIST	65,20	-0,31	0,0234	-0,22
		TEST	65,57	-0,27	0,0253	-0,39
		Analog	66,01	-0,25	0,1842	-0,27
	GD-short	BIST	-5,07	-70,58	0,0046	-80,50
		TEST	0,15	-65,69	0,0046	-81,89
		Analog	12,42	-53,84	0,0256	-86,14
	Stuck Open	BIST	65,44	-0,07	0,0235	-0,11
		TEST	65,45	-0,39	0,0254	0,04
		Analog	66,01	-0,25	0,1847	0,01
M14	Leff +20%	BIST	65,61	0,10	0,0239	1,64
		TEST	65,90	0,06	0,0259	1,97
		Analog	66,01	-0,25	0,1879	1,73
	Weff -20%	BIST	64,97	-0,54	0,0239	1,72
		TEST	65,37	-0,47	0,0259	1,97
		Analog	65,32	-0,94	0,1879	1,73
M18	GS-short	BIST	65,01	-0,50	0,0235	0,07
		TEST	65,00	-0,84	0,0254	0,01
		Analog	65,81	-0,45	0,1849	0,11
	Stuck Close	BIST	41,64	-23,87	0,0042	-82,05
		TEST	43,04	-22,80	0,0045	-82,28
		Analog	32,08	-34,18	0,0251	-86,41

G.1.2 BIST Simulation results Integrator

Table G-2 BIST Performance with deviations introduced in Integrator 1

Unit tested	Deviation Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
CS1p	Short	BIST	46,68	-18,84	0,0027	-88,51
		TEST	50,21	-15,63	0,0031	-87,80
		Analog	46,53	-19,73	0,0164	-91,12
CL1p	Short	BIST	1,58	-63,93	0,0007	-97,02
		TEST	1,87	-63,97	0,0012	-95,28
		Analog	5,36	-60,90	0,0120	-93,50
Ccdsp	Short	BIST	65,50	-0,01	0,0239	1,55
		TEST	65,84	0,00	0,0259	1,97
		Analog	65,63	-0,63	0,1878	1,68
M7	Leff +20%	BIST	65,13	-0,38	0,0239	1,56
		TEST	65,54	-0,30	0,0259	1,97
		Analog	65,46	-0,80	0,1879	1,73
	Leff -20%	BIST	65,53	0,02	0,0235	-0,10
		TEST	65,83	-0,01	0,0254	0,00
		Analog	66,26	0,00	0,1847	0,00
	Weff -20%	BIST	65,06	-0,45	0,0239	1,66
		TEST	65,25	-0,59	0,0259	1,97
		Analog	65,72	-0,54	0,1879	1,73
M9	Leff +20%	BIST	65,64	0,13	0,0235	0,01
		TEST	65,79	-0,05	0,0254	0,00
		Analog	66,25	-0,01	0,1848	0,05
	Leff -20%	BIST	65,06	-0,45	0,0239	1,62
		TEST	65,00	-0,84	0,0258	1,42
		Analog	65,65	-0,61	0,1878	1,70
M10	Leff -20%	BIST	64,68	-0,83	0,0234	-0,21
		TEST	64,68	-1,16	0,0254	-0,12
		Analog	64,53	-1,73	0,1846	-0,05
	Weff -20%	BIST	65,06	-0,45	0,0238	1,45
		TEST	65,83	-0,01	0,0258	1,58
		Analog	65,67	-0,59	0,1877	1,62
M18	Weff -20%	BIST	65,19	-0,32	0,0235	-0,02
		TEST	65,93	0,09	0,0254	0,03
		Analog	66,21	-0,05	0,1847	0,01

G.2 BIST Simulation results Common Mode Voltage Generator

Table G-3 BIST Performance with deviations introduced in Common Mode Voltage Generator

Unit tested	Deviation Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
C2	Short	BIST	65,06	-0,45	0,0236	0,62
		TEST	65,75	-0,09	0,0254	0,16
		Analog	66,09	-0,17	0,1862	0,81
C3	Short	BIST	64,41	-1,11	0,0234	-0,23
		TEST	63,38	-2,46	0,0253	-0,39
		Analog	65,19	-1,07	0,1845	-0,11
M2	Leff -20%	BIST	65,06	-0,45	0,0239	1,60
		TEST	65,77	-0,07	0,0257	1,30
		Analog	65,78	-0,48	0,1877	1,62
	Weff -20%	BIST	65,06	-0,45	0,0237	0,79
		TEST	65,52	-0,32	0,0254	0,08
		Analog	65,99	-0,27	0,1857	0,54

G.3 BIST Simulation results Bias circuit

Table G-4 BIST Performance with deviations introduced in Bias circuit

Transistor tested	Deviation Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
M1	Leff -20%	BIST	65,06	-0,45	0,0235	-0,02
		TEST	65,45	-0,39	0,0254	0,04
		Analog	66,48	0,22	0,1848	0,05
M2	Leff +20%	BIST	65,04	-0,47	0,0235	-0,11
		TEST	65,63	-0,21	0,0253	-0,24
		Analog	65,36	-0,90	0,1846	-0,05
	Leff -20%	BIST	65,25	-0,26	0,0235	0,01
		TEST	65,57	-0,27	0,0254	0,02
		Analog	66,20	-0,06	0,1847	0,01
M4	Leff +20%	BIST	65,46	-0,05	0,0234	-0,49
		TEST	65,78	-0,06	0,0254	-0,08
		Analog	66,28	0,02	0,1846	-0,05
	Weff -20%	BIST	65,35	-0,16	0,0235	-0,14
		TEST	65,74	-0,10	0,0253	-0,39
		Analog	65,75	-0,51	0,1845	-0,11

G.4 BIST Simulation results Comparator

Table G-5 BIST Performance with deviations introduced in Comparator

Transistor tested	Deviation Introduced	Stimulus	SINAD dB	Change dB	Amplitude V	Change %
M2	Weff -20%	BIST	65,17	-0,34	0,0235	-0,02
		TEST	65,68	-0,16	0,0254	-0,02
		Analog	66,14	-0,12	0,1847	-0,01
M4	Leff +20%	BIST	65,12	-0,39	0,0235	-0,02
		TEST	65,48	-0,36	0,0254	-0,04
		Analog	66,08	-0,18	0,1847	-0,03
M8	Weff -20%	BIST	65,63	0,12	0,0235	0,11
		TEST	66,37	0,53	0,0255	0,39
		Analog	66,30	0,04	0,1853	0,32