ISSN 0280–5316
ISRN LUTFD2/TFRT--5602--SE

Methods for Tone-Suppression in $\Sigma\Delta$ Modulators

Tommy Isaksson

Department of Automatic Control
Lund Institute of Technology
September 1998

Department of Automatic Control
Lund Institute of Technology
Box 118
S-221 00 Lund Sweden

Document name
MASTER THESIS

Date of issue
September 1998

Document Number
ISRN LUTFD2/TFRT--5602--SE

Author(s)
Tommy Isaksson

Supervisor
Bo Bernhardsson, Karl Johan Åström,
Håkan Eriksson

Sponsoring organisation
Ericsson Mobile Communications, Lund

Title and subtitle
Methods for Tone-Suppression in Sigma-Delta Modulators

Abstract

Unwanted tones in the output spectrum is a well-known problem in Sigma-Delta modulation. The actual tones, however, are just one aspect of a complex problem. A more comprehensive picture is given by considering tones as the evidence of a modulator diverging from the desired way of function. The desired, or ideal, modulator function is described by a linearized model, for which performance and behaviour is easily predicted. A key concept in the analysis of the tone-problem is the close relationship between limit cycles in state-space, repeated patterns in the time domain and tones in the frequency domain.

A commonly used method to linearize the Sigma-Delta modulator is to use additive dither, i.e., to add an independent, random signal to the modulation. This approach has proved capable of tone suppression. However, the knowledge of dither signals has been rather empirical of nature. On that account, a modulator model which utilizes Signal Dependent Dithering (SDD) is introduced in an attempt to gain a better understanding of the consequences of different dither implementations. One objective is to find an optimal dither signal, which is capable of suppressing tones with minimal impact on the overall quantization noise of the modulation. As a starting point, the quantization entropy is used as a measure of tone suppressing ability, assuming that a high level of uncertaincy will help to break up limit cycles and dissolve unwanted tones. Unfortunately, tone suppression ability proves a difficult quality to measure and, in particular, the entropy of the modulation proves inadequate. However, the impact of dither have another interesting interpretation: For proper choices of dither signals, the actual quantization is linear in the mean. Moreover, such dither signals also seem capable of tone suppression.

The use of State-Vector Dependent Dither is discussed as a natural extension of the Signal Dependent Dither model. Finally, the investigated methods for tone suppression are evaluated for one specific application. The application is Sigma-Delta modulation in fractional-N frequency synthesis, for which it is demonstrated that proper dithering can entail favourable results.

Key words
Sigma-Delta Modulation

Classification system and/or index terms (if any)

Supplementary bibliographical information

ISSN and key title
0280–5316

ISBN

Language
English

Number of pages
56

Recipient's notes

Security classification

Contents

Preface

The present thesis is the result of a joint project between Ericsson Mobile Communications AB and the Department of Automatic Control at the Lund Institute of Technology. It has been carried out in fulfilment for the M.Sc. degree in Electrical Engineering at the University of Lund, Sweden.

Advisors for the project have been Bo Bernhardsson (the Department of Automatic Control), Håkan Eriksson (Ericsson Mobile Communications) and Prof. Karl Johan Åström (the Department of Automatic Control). I would like to thank them for all their help and support. Thanks also to Leif Andersson for helping me with valuable LaTeX- expertise.

Abstract

Unwanted tones in the output spectrum is a well-known problem in ΣΔ modulation. The actual tones, however, are just one aspect of a complex problem. A more comprehensive picture is given by considering tones as the evidence of a modulator diverging from the desired way of function. The desired, or ideal, modulator function is described by a linearized model, for which performance and behaviour is easily predicted. A key concept in the analysis of the tone-problem is the close relationship between limit cycles in state-space, repeated patterns in the time domain and tones in the frequency domain.

A commonly used method to linearize the ΣΔ modulator is to use additive dither, i.e., to add an independent, random signal to the modulation. This approach has proved capable of tone suppression. However, the knowledge of dither signals has been rather empirical of nature. On that account, a modulator model which utilizes Signal Dependent Dithering (SDD) is introduced in an attempt to gain a better understanding of the consequences of different dither implementations. One objective is to find an optimal dither signal, which is capable of suppressing tones with minimal impact on the overall quantization noise of the modulation. As a starting point, the quantization entropy is used as a measure of tone suppressing ability, assuming that a high level of uncertaincy will help to break up limit cycles and dissolve unwanted tones. Unfortunately, tone suppression ability proves a difficult quality to measure and, in particular, the entropy of the modulation proves inadequate. However, the impact of dither have another interesting interpretation: For proper choices of dither signals, the actual quantization is linear in the mean. Moreover, such dither signals also seem capable of tone suppression.

The use of State-Vector Dependent Dither is discussed as a natural extension of the Signal Dependent Dither model. Finally, the investigated methods for tone suppression are evaluated for one specific application. The application is ΣΔ modulation in fractional-N frequency synthesis, for which it is demonstrated that proper dithering can entail favourable results.

1. Introduction

1.1 ΣΔ Modulation

Sigma-Delta modulation is a widespread technique with applications in several fields. One example is analogue-to-digital conversion: The modulation process attains high resolution for relatively low signal bandwidths, which makes it especially suitable for speech and audio applications [1]. Another example is fractional-N frequency synthesis, where ΣΔ modulators can be used to improve frequency resolution with simplicity and reduced cost [2, 3].

A block diagram of a ΣΔ modulator system is presented in Fig. 1.1. The system contains an interpolator and a ΣΔ modulator followed by a lowpass filter and a decimator. The input signal to the modulator, $u(n)$, is usually oversampled, i.e. the sampling frequency is much higher than the Nyquist rate. The purpose of the modulation is to produce an output, $y(n)$, which has low resolution (typically only one bit), yet upon lowpass filtering and decimation approximates the input, $x(n)$. As the modulator output has low resolution, oversampling is vital in order to retain the signal information.

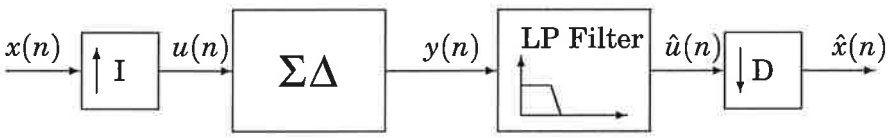

Figure 1.1 ΣΔ Modulation

It is well known that ΣΔ modulators may suffer from repeated patterns or "tones" in the modulator output. This phenomenon is particularly common in the presence of low or moderate amplitude constant input [4, 5, 6]. Tones are basically sinusoidal oscillations caused by limit cycles due to the circuit nonlinearity and can be extremely disturbing in many applications. For some time it was generally believed that higher order modulators did not suffer from unwanted tones. However, later investigations have shown that this is not the case [6]. Suggested techniques for eliminating tones in ΣΔ modulation include the use of dither signals or using chaotic modulators. The discussion in the present thesis will be limited to non-chaotic methods only. The purpose of dither signals, is to introduce a certain degree of uncertainty into the modulation and thereby randomize the output sequence. However, there is a trade-off between system stability, performance and tone persistence.

1.2 Overview

The objective of this paper is to investigate the tone problem in ΣΔ modulation. In particular, the use of non-chaotic methods for tone elimination will be examined.

Ch.2 presents the ΣΔ modulator as a non-linear dynamical system. Basic properties of ΣΔ modulation are reviewed, supplying necessary tools and a starting point for the following analysis of tones and tone suppression. The chapter includes a discussion on the linearized modulator model, its utilities and limitations. Other key concepts are limit cycles and, in particular, their connection to tones in the modulator output spectrum.

Ch.3 discusses methods for tone-suppression. The starting point is a ΣΔ modulator model with Signal Dependent Dither (SDD). It is shown that tone suppression in general brings about an increased noise-floor level and that it is necessary to find methods that randomizes the output sequence with minimal impact on the overall quantization noise. The chapter also includes a discussion on State-Vector Dependent Dither (SVDD).

Ch.4 is an application example and deals with ΣΔ modulators in connection to fractional-N frequency synthesis. The purpose is to evaluate the methods for tone-suppression discussed in Ch.3.

2. Basic Analysis

2.1 The ΣΔ Modulator

A ΣΔ modulator is shown in Fig. 2.1. The modulator consists of a quantizer embedded in a negative feedback loop together with a linear filter, $G_R(z)$. Although multi-bit quantizers may very well be used, this report deals with 1-bit quantizers exclusively. This means that the quantizer output, $y(n)$, is restricted to two values only (typically ± 1).

A ΣΔ modulator can be viewed upon as a control system where the modulator input, $u(n)$, is the desired signal. The controller, $G_R(z)$, is fed with an error signal, which is the difference between the desired signal, $u(n)$, and the actual modulator output, $y(n)$. The output from the controller, $e(n)$, is the input to the quantizer. Basically, the controller acts to keep the low-pass part of the modulator output equal to the low-pass part of the input. Clearly, a binary quantizer gives poor resolution as the input may have an infinite number of amplitudes. However, taken over a large set of output elements, the modulator will track the input in the mean. The larger the set, the better the approximation, which explains the trade-off between speed and resolution.

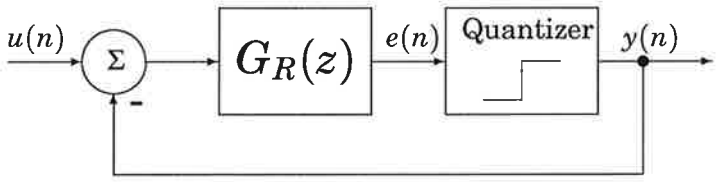

Figure 2.1 ΣΔ Modulator

A more general modulator topology is shown in Fig. 2.2. This model has two distinct transfer functions, namely the feedback transfer function, $H(z)$, and the input transfer function, $G(z)$. If the two transfer functions are equal, this model is equivalent to the one in Fig. 2.1 with $G_R(z) = G(z) = H(z)$. However, this is not necessarily the case.

Example 2.1 A first order ΣΔ modulator is shown in Fig. 2.3. The linear filter is simply a discrete time integrator, accumulating the difference between the input signal, $x(n)$, and the binary output, $y(n)$. This modulator is equal to the one in Fig. 2.1 if $G_R(z) = \frac{z^{-1}}{1-z^{-1}}$. Fig. 2.4 shows a simulation of this particular modulator. □

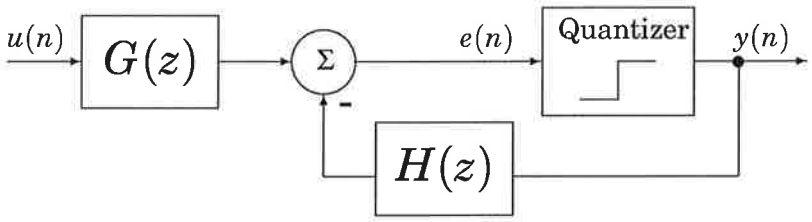

Figure 2.2 Generic ΣΔ Modulator

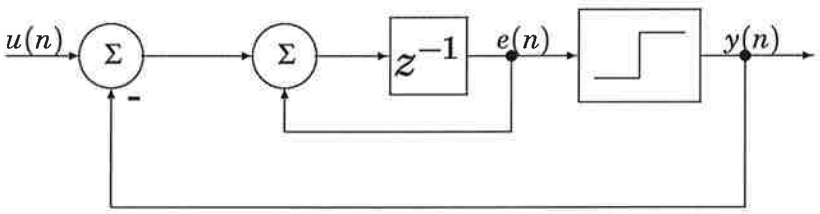

Figure 2.3 First-order, 1-b ΣΔ modulator

State-Space Description of the ΣΔ Modulator

Any ΣΔ modulator can be characterized by a state-space, $\mathcal{S}$, and two mappings, $\mathcal{F}$ and O. The state-space is the set of possible states of the systems and the mapping $\mathcal{F}$ describes the next state as a function of the current state and the system input. The output of the system is determined by the current state and the system input.

$$\begin{aligned} \boldsymbol{x}(k+1) &= \mathcal{F}(\boldsymbol{x}(k), u(k)) = \Phi\boldsymbol{x}(k) + Au(k) - By(k) \\ y(k) &= O(\boldsymbol{x}(k), u(k)) = \mathrm{sgn}(C\boldsymbol{x}(k) + Du(k)) \end{aligned} \tag{2.1}$$

The mapping $\mathcal{F}$ is linear while O is not.

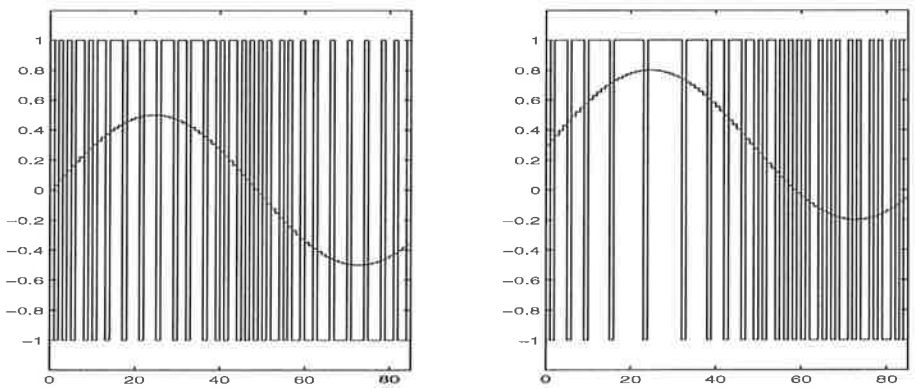

Figure 2.4 Input and output from the first-order Sigma-Delta modulator of Example 2.1. a) The input signal is a sine with amplitude 0.5 and frequency 0.0104. b) The same sinusoid with a DC-offset of 0.3.

Example 2.2 Consider the second order modulator in Fig. 2.5 with multiple feedback and feedforward topology to minimize delay. The state vector is chosen as $\boldsymbol{x}(k) = \begin{bmatrix} x_1(k) & x_2(k) \end{bmatrix}^T$, which results in the following state-space description:

$$\begin{aligned} \boldsymbol{x}(k+1) &= \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \boldsymbol{x}(k) + \begin{bmatrix} 1 \\ 1 \end{bmatrix} u(k) - \begin{bmatrix} 1 \\ 2 \end{bmatrix} y(k) \\ y(k) &= \operatorname{sgn}\left(\begin{bmatrix} 0 & 1 \end{bmatrix} \boldsymbol{x}(k) \right) \end{aligned}$$

The feedback transfer function (according to the generic ΣΔ modulator model of Fig. 2.2) is

$$H(z) = \frac{2z^{-1} - z^{-2}}{1 - 2z^{-1} + z^{-2}}$$

and the input transfer function is

$$G(z) = \frac{z^{-1}}{1 - 2z^{-1} + z^{-2}}$$

□

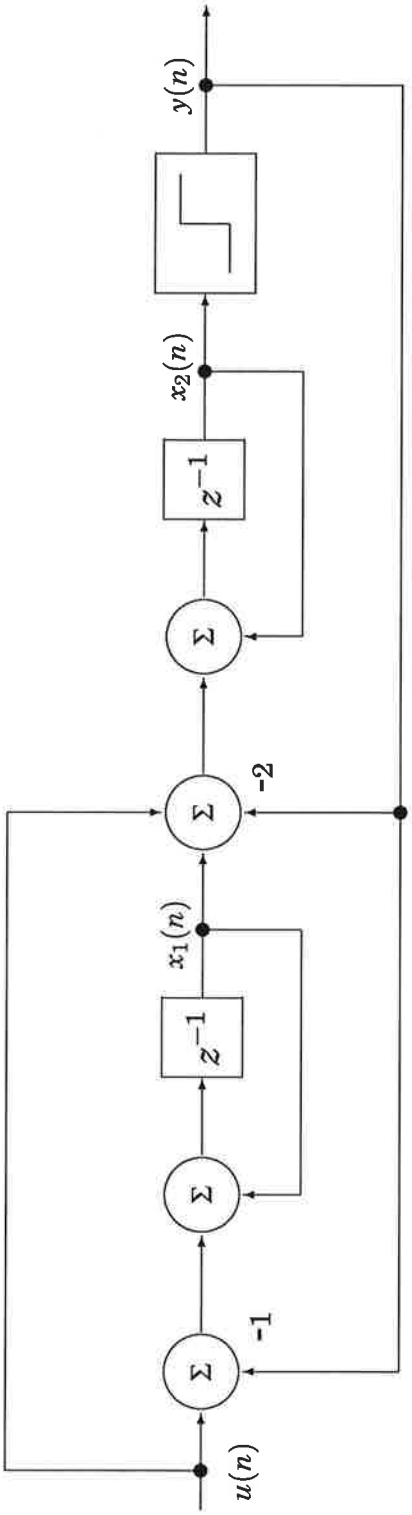

Figure 2.5 Second-order, feedforward ΣΔ modulator of Example 2.2

The Linearized Modulator Model

A popular method to analyze the ΣΔ modulator of Fig. 2.2 is to model the highly non-linear quantizer as a quantizer gain together with additive quantizer noise as seen in Fig. 2.6 [1, 4, 5]. This model is generally referred to as the linearized modulator model. The output of the linearized modulator can be split into two parts:

$$Y(z) = Y_u(z) + Y_q(z) = STF_K(z)U(z) + NTF_K(z)Q(z)$$

where

$$STF_K(z) = \frac{KG(z)}{1 + KH(z)}$$

and

$$NTF_K(z) = \frac{1}{1 + KH(z)}$$

are the Signal Transfer Function and the Noise Transfer Function respectively [5]. The indices K for the STF and NTF indicate a model with variable quantizer gain. In the present thesis it will be assumed that $K = 1$. For a discussion on this property, refer to [5].

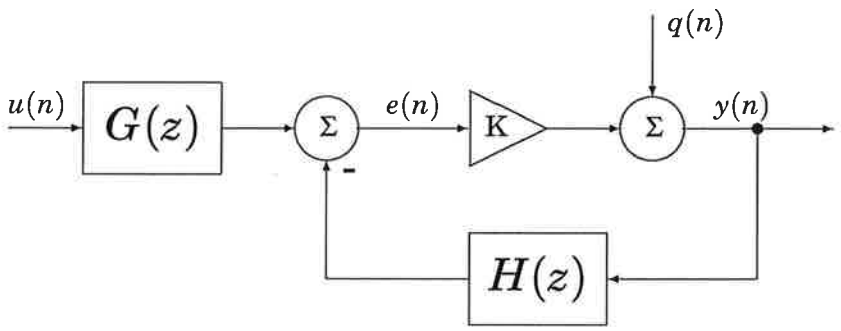

Figure 2.6 Linearized model of a general Sigma-Delta modulator

A common approximation is that the quantization noise is independent of the linear filter output, $e(n)$, and uniformly distributed in $[\frac{-\Delta}{2}, \frac{\Delta}{2}]$, where Δ is the quantizer step size. This yields a quantization noise variance of $\sigma_q^2 = \frac{\Delta^2}{12}$ and the resulting quantization noise in the modulator output power spectrum is:

$$R_{Y_q}(e^{j\omega}) = |NTF_1(e^{j\omega})|^2 \cdot \frac{\Delta^2}{12} \tag{2.2}$$

In general, $H(z)$ is designed as a low-pass filter, which implies that $NTF_1(z)$ is a high-pass filter. This is generally known as noise-shaping and acts to push as much as possible of the quantizer noise up in frequency and out of the signal band, where it can be removed by low-pass filtering. Fig. 2.7 displays the (linear model) power spectrum of the modulator output for $U(z) = 0$ for the first and second order modulators of Fig. 2.3 and Fig. 2.5 repectively. In the discussion to follow, this linear behaviour will be refered

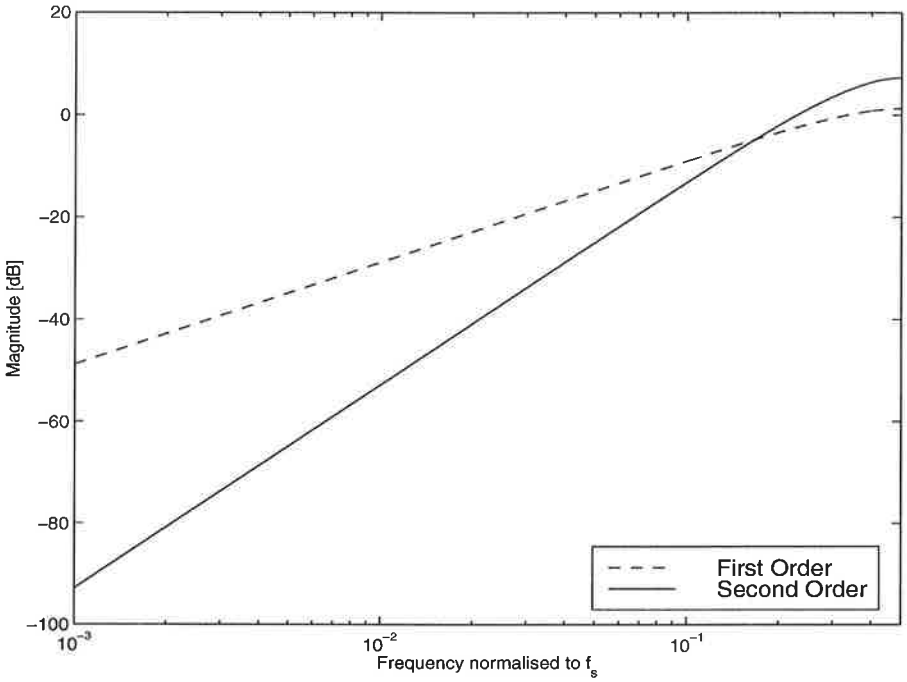

Figure 2.7 Linearized model output power spectrum for $U(z) = 0$, for first order (dashed) and second order (solid) $\Sigma\Delta$ modulators.

to as the ideal, or desired, behaviour. The reason for this is simply that a linear modulator will not exhibit unexpected tones and its behaviour and performance are easily predictable. The quantization noise in the modulator output is well-defined and, moreover, does not depend on the input signal.

For the first-order modulator of Fig. 2.3, every doubling of the sampling rate improves the signal-to-noise ratio by 9 dB, providing 1.5 extra bits of resolution [4]. Higher order modulators provide more quantization noise suppression over the signal band and more amplification of the noise outside the signal band.

The Classical Dither Approach

The linearized modulator model has proved valuable for predicting performance and to understand basic properties of $\Sigma\Delta$ modulators. However, the approach has some drawbacks. For instance, the assumption that the quantization noise is independent of the quantizer input does not capture the non-linear behaviour of the actual modulator. In fact, the quantization noise is usually non-white and the modulator may behave rather unpredictable.

An often used method to linearize the modulator, i.e., force the modulator to behave according to the linearized model, is to add a random dither signal to the signal to be quantized as seen in Fig. 2.8. In the linearized model, the dither signal, $d(n)$, is added to the signal together with the quantization noise, $q(n)$, yielding a total quantization noise variance of $\sigma_q^2 + \sigma_d^2$, where σ_d^2 is the variance of the random dither signal. Thus, the

total noise power spectrum for the dithered modulator is

$$R_{Y_q}(e^{j\omega}) = |NTF_1(e^{j\omega})|^2 \cdot \left(\frac{\Delta^2}{12} + \sigma_d^2\right) \tag{2.3}$$

to be compared with the undithered noise power spectrum of Eq. (2.2). The idea is that the sum of the dither signal and the actual quantization noise will be sufficiently white for the linear approximation to hold. As seen in Eq. (2.3), the price to pay for linearizing the modulator is an increase in noise power. For higher order modulators (modulator order > 2), an increased amount of noise in the modulator loop tend to reduce system stability, which in turn requires a reduction in loop gain (K) and a corresponding increase in baseband quantization noise [5, 6].

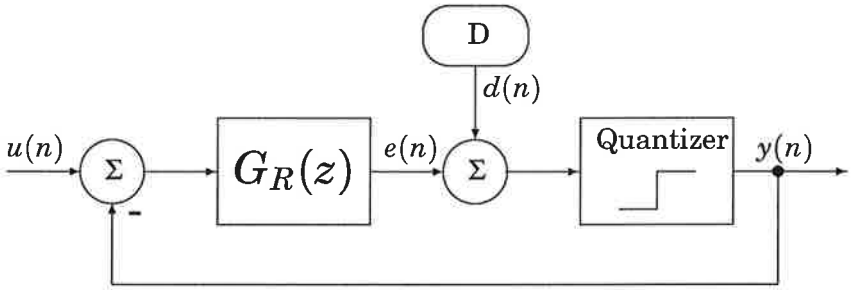

Figure 2.8 ΣΔ modulator with additive dither source

2.2 Properties of Non-Linear Dynamical Systems

In Sec. 2.1, the linearized modulator model was used to describe the desired behaviour of ΣΔ modulators. In order to explain the origin of tones, however, the linearized model is not sufficient. The purpose of this section is to provide a starting point for the investigation of tones in ΣΔ modulators. Certain characteristics of the non-linear dynamical systems are of particular interest in this respect. The definitions to follow are reviewed from Parker and Chua [9] and Risbo [5].

Autonomous Systems

A certain subset of the dynamical systems are the autonomous systems. Autonomous systems do not depend on time, i.e., they operate without an external input signal. For instance: Consider the nonautonomous dynamical system representation of the ΣΔ modulator of Eq. (2.1). If the modulator input is constantly zero ($u(k) = 0, k = 0, 1, 2, ...$) the modulator can be characterized as an autonomous system:

$$\begin{aligned} \boldsymbol{x}(k+1) &= \mathcal{F}(\boldsymbol{x}(k)) = \Phi\boldsymbol{x}(k) - By(k) \\ y(k) &= O(\boldsymbol{x}(k)) = \mathrm{sgn}(C\boldsymbol{x}(k)) \end{aligned}$$

The states depend on previous states only. Consequently, the states at a given time k can be found by iteratively applying the mapping $\mathcal{F}$ on the initial states of the modulator, $\boldsymbol{x}(0) \in \mathcal{S}$. Likewise, the modulator output, $y(k)$, is simply the output mapping of the kth iterative mapping of $\mathcal{F}$ on $\boldsymbol{x}(0)$:

$$\begin{aligned} \boldsymbol{x}(k+1) &= \mathcal{F}^k(\boldsymbol{x}(0)) \\ y(k) &= O(\mathcal{F}^k(\boldsymbol{x}(0))) \end{aligned} \tag{2.4}$$

Definition 2.1 *The solution to* (2.4)*, starting at an initial condition* $\boldsymbol{x}(0)$*, is the sequence* $\{\boldsymbol{x}(k)\}_{k=0}^{\infty} = \{\mathcal{F}^k(\boldsymbol{x}(0))\}_{k=0}^{\infty}$*. This solution is called the* ***orbit*** *corresponding to the initial state.*

The orbit defines all the states the system visits for a given initial condition.

If the input signal to a nonautonomous system can be generated by an autonomous system, the nonautonomous system and the input generating system can be merged into a single autonomous system [5]. Especially, systems with constant input can generally be described as autonomous. According to [9], an nth-order time-periodic nonautonomous system can always be converted into an $(n+1)$th order autonomous system. Thus, a $\Sigma\Delta$ modulator with constant or periodic input can generally be characterized as autonomous.

Limit Sets

This section aims to characterize the steady state behaviour of non-linear autonomous dynamical systems from a state-space point of view. The definitions of limit sets and limit cycles apply on autonomous systems only. Steady state refers to the asymptotic behaviour as $k \to \infty$.

Definition 2.2 *A point* $\boldsymbol{y}$ *of* $\boldsymbol{x}(0)$ *is called a* ***limit point*** *if, for every neighborhood* U *of* $\boldsymbol{y}$*, the orbit repeatedly enters* U *as* $k \to \infty$*. The set* $L(\boldsymbol{x}(0))$*, containing all limit points of* $\boldsymbol{x}(0)$*, is called the* ***limit set****.*

The limit set L of an initial condition is the set in $\mathcal{S}$ the orbit visits frequently in steady-state.

Definition 2.3 *A limit set* L *is* ***attracting*** *if there exists an open neighborhood* U *of* L *such that* $L(\boldsymbol{x})=L$ *for all* $\boldsymbol{x} \in U$*. The* ***basin of attraction*** B_L *of an attracting limit set* L *is the union of all such neighborhoods* U*.*

B_L is the set of all initial conditions that are asymptotically attracted to the limit set.

Definition 2.4 *A* ***periodic point*** $\boldsymbol{x}_p$ *of* $\mathcal{F}$ *is a point for which* $\mathcal{F}^k(\boldsymbol{x}_p) = \boldsymbol{x}_p$ *for some period* k*. The least number* K *for which* $\mathcal{F}^K(\boldsymbol{x}_p) = \boldsymbol{x}_p$ *is called the prime period of the periodic point and a periodic point with prime period* K *is called a period-*K *point. The closed orbit* $\{\boldsymbol{x}_p, \mathcal{F}(\boldsymbol{x}_p), ..., \mathcal{F}^K(\boldsymbol{x}_p)\}$ *is called a* ***limit cycle****, which is the limit set of the period-*K *point.*

Example 2.3 Consider again the first order modulator in Fig. 2.3 with the following dynamical system description:

$$\begin{aligned} \boldsymbol{x}(k+1) &= \boldsymbol{x}(k) + u(k) - y(k) \\ y(k) &= \operatorname{sgn}(\boldsymbol{x}(k)) \end{aligned}$$

Assuming zero input ($u(k) = 0, k = 0, 1, 2, ...$) and introducing polar coordinates:

$$\begin{aligned} r &= |\boldsymbol{x}(k)| \\ \theta &= \frac{\pi}{2}\operatorname{sgn}(\boldsymbol{x}(k)) \\ \Delta r &= |\boldsymbol{x}(k) - \operatorname{sgn}(\boldsymbol{x}(k))| - |\boldsymbol{x}(k)| \\ &= \sqrt{(\boldsymbol{x}(k) - \operatorname{sgn}(\boldsymbol{x}(k)))^2} - |\boldsymbol{x}(k)| \\ &= \sqrt{\boldsymbol{x}^2(k) - 2\boldsymbol{x}(k)\operatorname{sgn}(\boldsymbol{x}(k)) + \operatorname{sgn}^2(\boldsymbol{x}(k))} - r \\ &= \sqrt{r^2 - 2r + 1} - r = \sqrt{(r-1)^2} - r \\ &= -1 \\ \Delta\theta &= \frac{\pi}{2}(\operatorname{sgn}(\boldsymbol{x}(k) - \operatorname{sgn}(\boldsymbol{x}(k)))) - \frac{\pi}{2}\operatorname{sgn}(\boldsymbol{x}(k)) \\ &= \begin{cases} -\pi, & if \quad r < 1 \\ 0 & if \quad r \geq 1 \end{cases} \end{aligned}$$

If $r > 1$, then $\Delta r = -1$ and $\Delta\theta = 0$ and the magnitude of r will decrease with 1. If $0 < r < 1$ then Δr is still -1 while $\Delta\theta = -\pi$. Eventually, this will force the modulator into a limit cycle behaviour, i.e., there exists a steady state periodic solution. In fact, there are infinitely many solutions; the steady-state depends on the initial condition of the modulator. However, all solutions have the prime period 2 and are oscillating between a and $(a-1)$, where the constant a is in the interval [0,1[. Thus; for practical purposes they are equivalent as they all produce the same output. The basin of attraction is the whole state space, i.e., any starting condition will converge towards the limit set. Fig. 2.9 shows the limit cycle for the modulator simulated with zero input and $\boldsymbol{x}(0) = 0.5$. □

Limit Cycle Identification

Exact analysis of higher order modulators is usually extremely difficult. There may be several coexisting limit cycles and the basin of attraction can in general not be determined analytically. However, the existence of a specific limit cycle can be tested by opening the feedback loop. According to [5], a periodic sequence, $y(n)$, with period k exists as a limit cycle if the condition $y(n) = \operatorname{sgn}(e(n))$, where $\boldsymbol{e}(n)$ is the steady-state filter output, holds for all n. Furthermore, $e(n)$ can be found from the linear set of

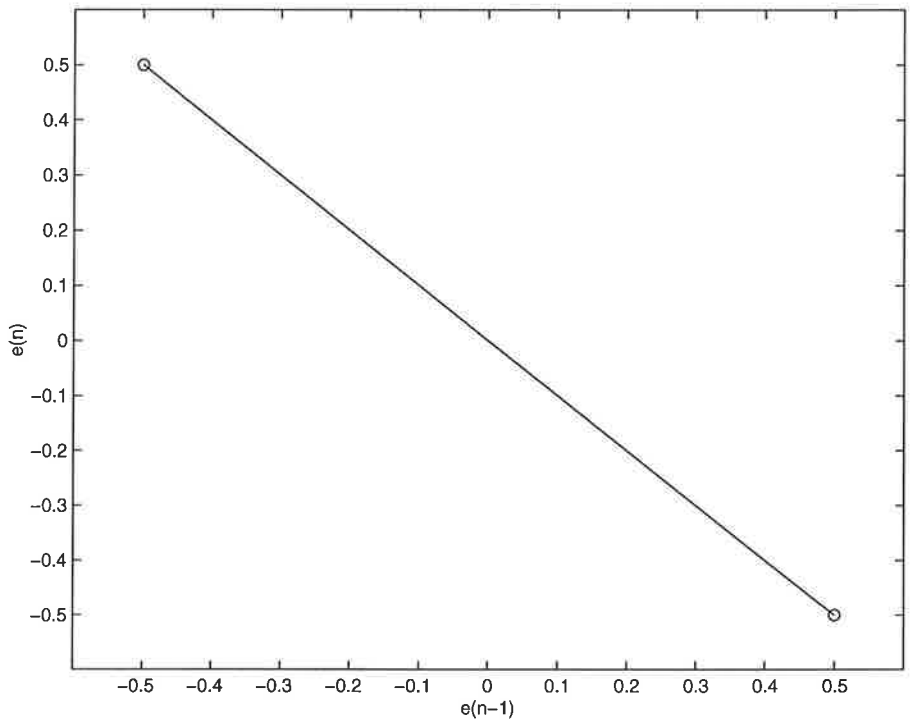

Figure 2.9 Limit cycle of first order ΣΔ modulator.

equations:

$$\begin{bmatrix} 1 & 0 & \cdots & d_N & \cdots & d_2 & d_1 \\ d_1 & 1 & 0 & \cdots & d_N & \cdots & d_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & d_N & \cdots & d_3 & d_2 & d_1 \end{bmatrix} \begin{bmatrix} e(0) \\ e(1) \\ \vdots \\ e(k-1) \end{bmatrix} = \begin{bmatrix} 0 & \cdots & c_N & \cdots & c_2 & c_1 \\ c_1 & 0 & \cdots & c_N & \cdots & c_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & c_N & \cdots & c_2 & c_1 & 0 \end{bmatrix} \begin{bmatrix} v(0) \\ v(1) \\ \vdots \\ v(k-1) \end{bmatrix} \tag{2.5}$$

Where d_n and c_n are the coefficents of the loop filter:

$$H(z) = \frac{C(z)}{D(z)} = \frac{c_1 z^{-1} + c_2 z^{-2} + \cdots + c_N z^{-N}}{1 + d_1 z^{-1} + d_2 z^{-2} + \cdots + d_N z^{-N}}$$

and $v(n) = u(n) - y(n)$ is the loop filter input. Using symbolic notation, the solution to (2.5) is:

$$\boldsymbol{e} = \mathbf{D}_k^{-1} \mathbf{C}_k \boldsymbol{v} \tag{2.6}$$

Following Risbo [5], to simplify notation, code sequences will be written as sequences of the symbols '1' and '0' corresponding to quantizer outputs 1 and -1 respectively. A periodic repetition of a code sequence is indicated by overlining. For instance, the limit cycle observed in Ex. 2.2 is $\overline{10}$. This limit cycle is very persistent, especially for low-order modulators and causes a strong tone at half the sample rate of the modulator.

Example 2.4 The second order modulator of Fig. 2.5 will be tested for the existence of the $\overline{10} = \overline{1010}$ periodic sequence as a limit cycle with zero input. The sequence gives $\mathbf{v} = [-1 \quad 1 \quad -1 \quad 1]^T$. For this particular modulator, $H(z)$ has a pole at z=1, which means that the filter has infinite dc-gain. The matrix $\mathbf{D}_k$ is non-invertible and the direct solution, Eq. (2.6), can not be used because the set of equations has infinitely many solutions. However, using Eq. (2.5) yields that $\mathbf{e} = [\frac{3}{2}+a \quad a \quad \frac{3}{2}+a \quad a]^T$ is a solution for any $a \in \mathbf{R}$. The limit cycle thereby exists for zero input, since $y(n) = \text{sgn}(e(n))$ for $-\frac{3}{2} \leq a < 0$. This corresponds to any starting condition $\boldsymbol{x}(0) = [\pm 0.5 \quad x_2]^T$, where the choice of x_2 is arbitrary. A test for the $\overline{1001}$ limit cycle also proved positive for all initial conditions with $x_1 \in \{-1, 0, 1\}$. Fig 2.10 displays the limit cycles of the modulator for $\boldsymbol{x}(0) = [0 \quad 0.5]^T$ and $\boldsymbol{x}(0) = [0.25 \quad 0.5]^T$ respectively. □

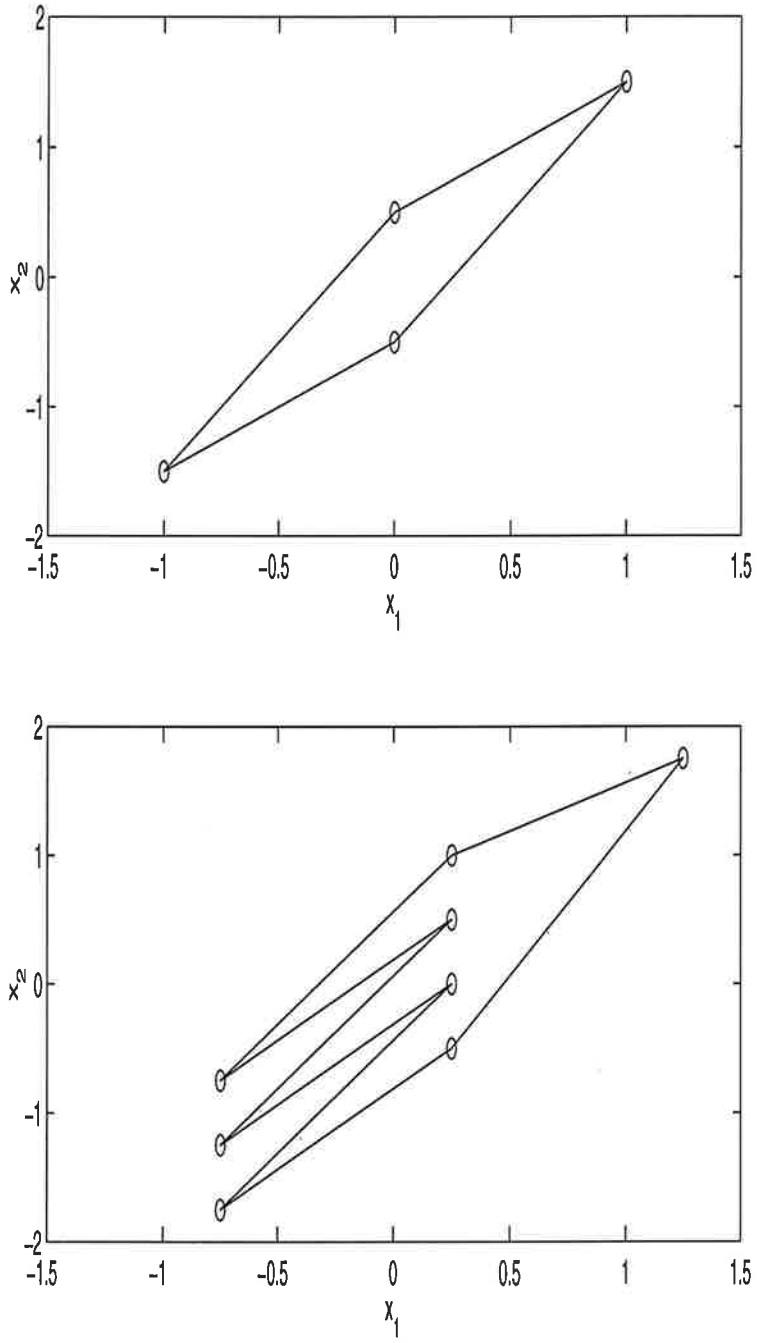

Figure 2.10 Limit cycles of second order feedforward ΣΔ modulator, a) $\overline{1001}$ limit cycle b) $\overline{11010100}$ limit cycle.

The Tone Problem

For practical purposes, the tone problem is the presence of unwanted tones in the modulator output spectrum. That is, instead of the ideal noise spectrum of Fig. 2.7, the quantization noise power is concentrated at specific frequencies. However, there are more aspects to the problem: Recall that the desired way of function of the modulator is described by the linearized model of Sec. 2.1. In reality, the modulator is not linear, because of the non-linear feedback of the 1-b quantizer output. This means that the modulator risks getting trapped in a limit cycle behaviour. In that case, the quantization noise is not white and the result is repeated patterns in the time-domain output and tones in the output spectrum. To sum up: As a more comprehensive picture of the problem, the origin of tones can be seen as the result of a modulator diverging from the linearized model, where the non-linear behaviour is characterized by:

- Limit cycles in state-space
- Non-white quantization noise
- Repeated patterns in the modulator output
- Tones in the output spectrum

As for the actual output spectrum tones, there are several aspects to take into consideration, such as:

- The magnitude of the first tone, i.e. the lowest frequency tone.
- The magnitude of the highest tone.
- The location of the tones, e.g., the distance from a low-frequency signal to the first tone.

For this reason, it can be troublesome to determine whether a method to eliminate tones is successful or not. The grade of success depends not only on the method itself, but on the actual application the modulator is being used for.

> **Example 2.5** The simple first-order modulator corresponding to Fig. 2.3 was simulated over 2^{16} samples with a constant input of $1/256$ and the quantization levels were ± 1. Fig. 2.12 shows that the modulator output spectrum contains several tones. When the modulator has a small constant input m_x, the output will alter between $+1$ and -1. From time to time, in order to keep the mean value of the modulator equal to the dc-bias, the modulator will typically generate two identical code segments. This pattern will repeat approximately every $1/m_x$ sample, as the difference between the modulator input and output accumulates in the integrator loop, and cause tones.
>
> In the frequency domain, each component of a limit cycle contains a spike at frequency 0 and spikes spaced at integer multiples of the fundamental frequency $f = 1/K$ [9]. Fig. 2.12 indeed contains such spikes or tones which indicates a limit cycle behaviour. The spectrum consists of tones located at every 128th bin. This corresponds to a fundamental frequency of $f = \frac{128}{N} = \frac{2^7}{2^{16}} = \frac{1}{512}$, suggesting a periodic solution with a prime

period of 512. However, the strong emphasis on tones at intervals of 256 bins indicates a strong correlation between adjacent groups of 256 samples. A possible interpretation is that the solution is *almost* 256-periodic. Fig. 2.13 displays the steady-state space of the modulator, confirming a periodic solution (although it is difficult to make out that the solution is 512-periodic.) The emergence of tones can also be seen from a time-domain perspective. In Fig. 2.11 the output of the modulator, fed with a constant input of 1/256, is shown together with the output corresponding to zero input. In comparison to the zero-input case, the modulator with non-zero input displays two distinct patterns: The basic $\{+1, -1, +1, -1, \ldots\}$ sequence is still obvious, but when the output is arranged in groups of 256 samples it is clear that adjacent groups are inverted. Secondly, in every other group, the last element is inverted. As expected, the period is 512 samples and there is indeed strong correlation between adjacent groups of 256 samples. □

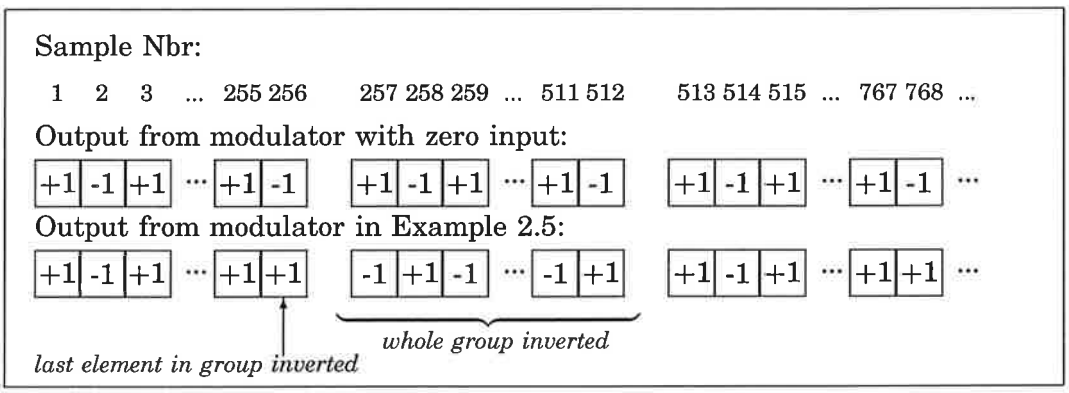

Figure 2.11 Output for the modulator in Example 2.5

For a thorough investigation of quantization noise in single-loop ΣΔ modulators with dc input, refer to [8].

Example 2.6 The second-order modulator of Fig. 2.5 was simulated under the same conditions as in Ex. 2.5 and the resulting spectrum is showed in Fig. 2.14. Since the space between adjacent tones is 64 bins, the prime period of the solution is $K = \frac{2^{16}}{2^6} = 1024$. In comparison to the first-order example of Fig. 2.12, the magnitude of the tones are significantly lower. However, the distance to the first tone is only half of that of the first order modulator. Fig. 2.15 displays the orbit of the second-order modulator plotted in a grey-scale: Points on the orbit that are visited frequently are lighter than points that are rarely visited. Points that are never visited are plotted in black. It is difficult to identify any limit cycle behaviour from the plot. However, the orbit is seemingly periodic. As for the time-domain perspectives, second- or higher-order modulators are rather difficult to analyse; the output patterns are usually quite complex. □

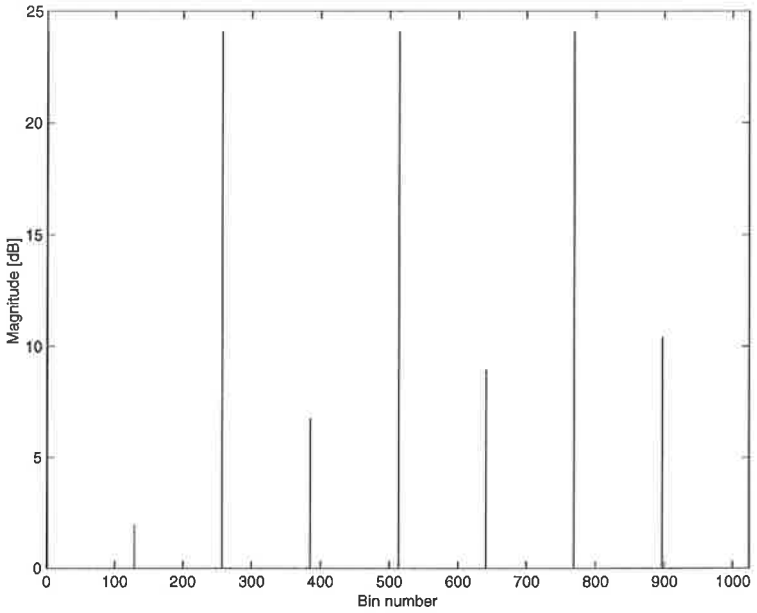

Figure 2.12 Magnitude of FFT of first order modulator output of Ex. 2.5. The spectrum has tones at every 128th bin and the tones of greatest magnitude recur every 256th bin.

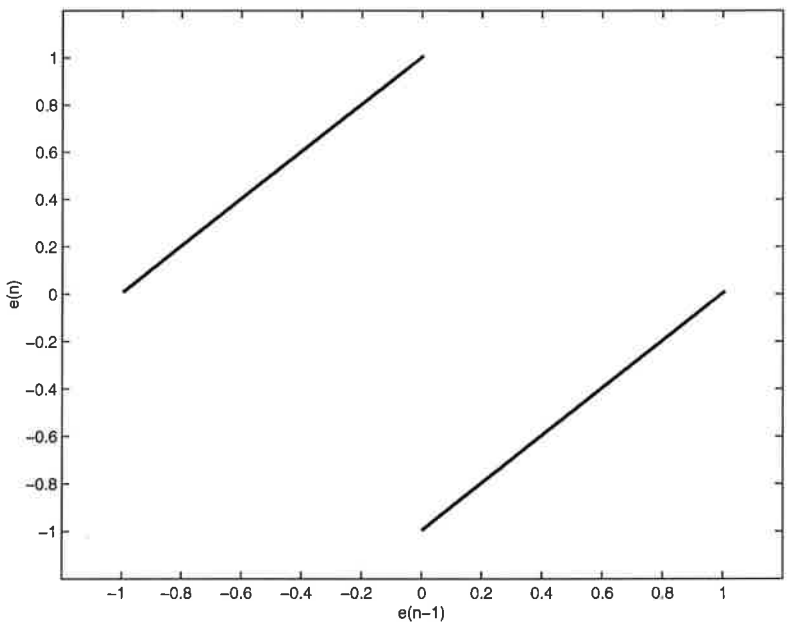

Figure 2.13 Limit Cycle of the first-order modulator of Ex. 2.5.

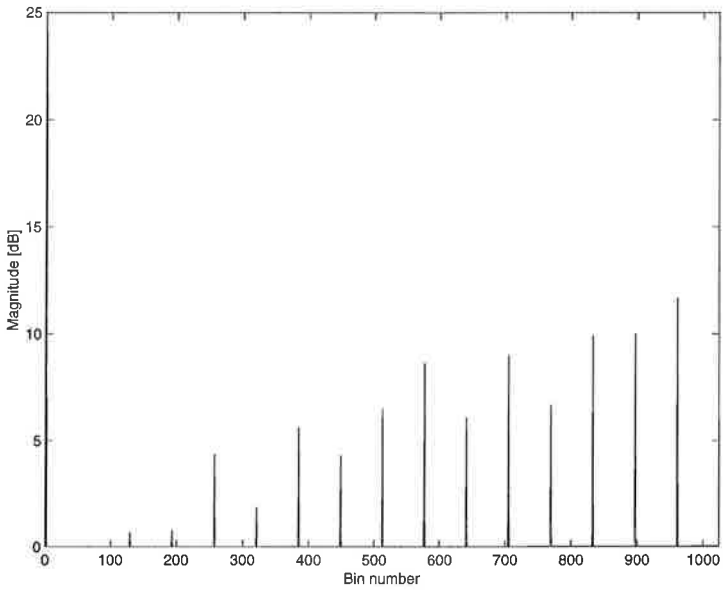

Figure 2.14 Magnitude of FFT of second order modulator output of Ex. 2.6. There are tones at every 64th bin.

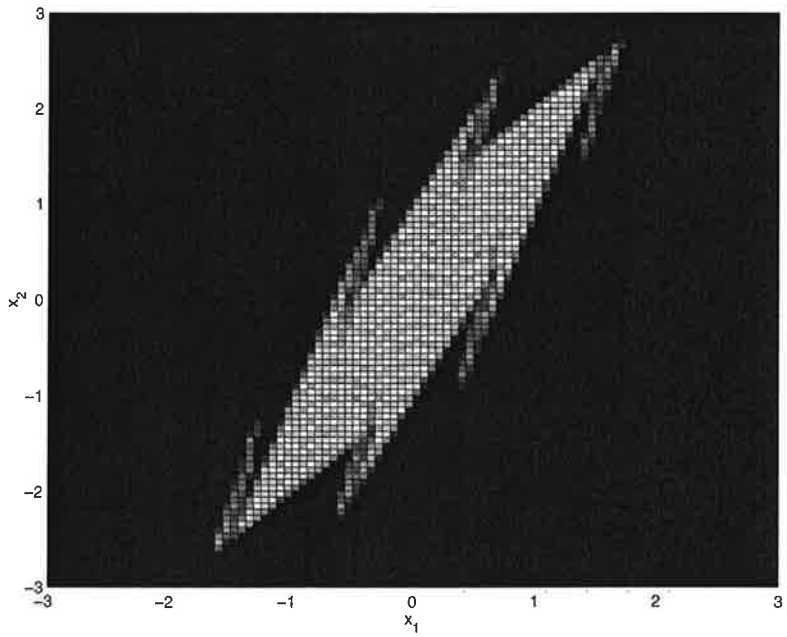

Figure 2.15 Orbit of the second order modulator of Ex. 2.6.

3. Methods for Tone-Suppression

Unwanted tones are produced when the modulator is trapped in a limit cycle behaviour. This is often characterized by repeated patterns in the output signal, i.e., there is a close relationship between the periodic output and the limit cycles in state-space [5]. In order to eliminate tones it is therefore important to consider how to break up, or randomize, these patterns. This can also be seen as an attempt to linearize the modulation, i.e., force the modulator to behave according to the linearized model.

The starting point for the analysis is a modulator model, which utilizes Signal Dependent Dithering (SDD) to linearize the modulation and suppress tones. This model is displayed in Fig. 3.1, where the modulator is equipped with a source D that generates a dither signal, $d(n) \in \{+1, -1\}$. As the dither signal affects the filter output, $e(n)$, before quantization, a -1 bit will alter the output signal in sign. The inversion probability, i.e., the probability that a -1 bit is generated, depends on the filter output and will be used to characterize different SDD methods:

$$p_d(e(n)) = \text{Prob}\{(d(n) = -1)|e(n)\}$$

There are certain advantages with the SDD model:

- A general feature of a $\Sigma\Delta$ modulator is the low-pass characteristics of the loop filter, $H(z)$. As a consequence, the (Quantization) Noise Transfer Function is high-pass and acts to push most of the quantization noise out of the signal band. Moreover, *any* modification made on the signal between $H(z)$ and the quantizer will be shaped accordingly, that is, have minimal impact on the modulated signal from a low-frequency perspective. In other words: Noise-shaping in the modulator ensures that the dither signal, $d(n)$, has minimal impact on the low-pass part of the output signal.
- Sign inversions may have a direct influence on limit cycles in state-space.
- Commonly used methods for tone-suppression, e.g., classical additive dither, are comprised in the SDD model.
- The model helps to understand the effect of different dither methods.
- The expected value of the probability $p_d(e(n))$ is a measure of the uncertainty of the quantization: Let $x = E[p_d(e(n))]$ be the expected value of the inversion probability and let $Y = \text{sgn}(y(n))$ and $E = \text{sgn}(e(n))$ be stochastic variables. The conditional entropy of the quantization output given the sign of the linear filter output is then:

  $$H(Y|E] = -x \cdot log_2(x) - (1-x) \cdot log_2(1-x),$$

  If $x = 0$, the uncertainty is 0 bits and for $x = 0.5$ the uncertainty is 1 bit.

- The probability $p_d(e(n))$ can be used to estimate the additional noise power introduced by the dither signal. Moreover, making $d(n)$ signal dependent can help to reduce the amount of additional noise power introduced.

It should be pointed out that the following analysis is conducted with a starting point in the need to randomize repeated patterns in the modulator output. This might seem surprising, since Ch. 2 recommends a more comprehensive approach to the problem. However, that position is, in many respects, motivated by the problems encountered in the following analysis.

For the simulations in the present work, the oversampling rate is set to 32, yielding a signal bandwidth of $0 \leq f < \frac{1}{64}$.

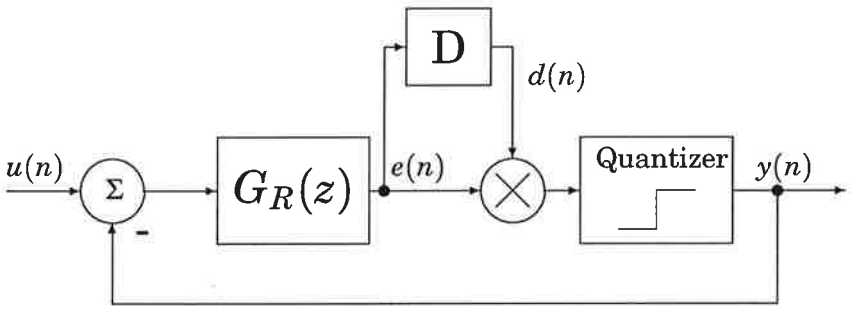

Figure 3.1 Sigma-Delta Modulator with Signal Dependent Dithering.

3.1 Signal Dependent Dither

Example 3.1 Consider the case where $p_d(e(n)) = p_1$, where p_1 is a constant (which implies that the dither signal is in fact signal independent). The first order modulator was simulated for different values of p_1 and the resulting output spectrums are shown in Fig. 3.2. Apparently, the noise floor drowns the signal before the tones are suppressed. In other words: this method is not capable of linearizing the modulator. □

The signal independent method of Ex. 3.1 has a major drawback: To randomize the output sequence the probability for sign alteration, p_1, needs to be large, and with that, the noise floor rises concurrently. Thus: in order to successfully suppress tones, it seems necessary to find measures with minimal impact on the overall quantization noise in the modulator.

Consider a 1-bit modulator, i.e. the quantizer is basically a sign detector. The quantizer output, $y(n)$, is typically chosen from the set $\{+Q, -Q\}$, where Q is the quantization level and the output is chosen in order to minimize the quantization error, $q(n) = y(n) - e(n)$. If the magnitude of $e(n)$ is close to a quantization level the choice of $y(n)$ seems natural since a different choice of $y(n)$ would introduce a considerable error. However, if

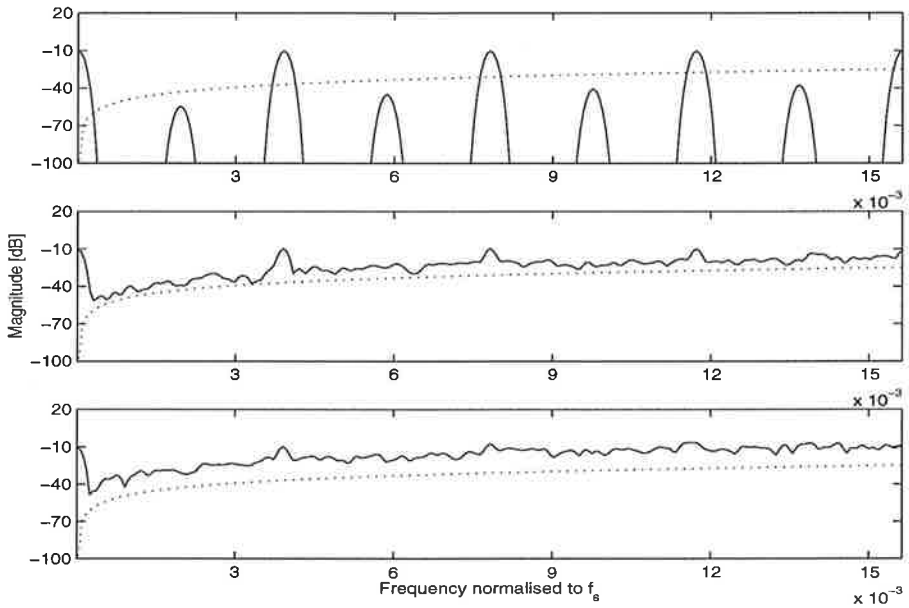

Figure 3.2 PSD of modulator output of Ex. 3.1. The values of p_1 are a) 0, b) 0.15 and c) 0.25. The dashed line is the corresponding linear approximation of the quantization noise.

$|e(n)| \ll Q$ the choice appears more arbitrary. For example, if $e(n)$ is small and positive, the additional error introduced by choosing $y(n) = -Q$ is relatively benign. The relationship between inverted codes and additional quantization error is illustrated in Fig 3.3. In Ex. 3.1, no attention was paid to the actual quantizer input when the output code was inverted: It was equally likely that elements corresponding to low magnitude input were inverted as elements corresponding to inputs of magnitudes close to Q. One might guess that this would lead to unnecessary quantization noise and that a better approach would be to take advantage of the existent filter output signals when deciding which samples to invert. This is also the basic idea of the Signal Dependent Dither model of Fig 3.1, which utilizes the information of the filter output, $e(n)$, when determining the instantaneous value of p_d.

Inverted Bits and Quantization Noise Power

To explain the benefit of the SDD model, consider Fig. 3.3: The quantization error can be written as $q(n) = q_0(n) + q_d(n)$, where $|q_0(n)| = |y(n)| - |e(n)|$ is the error corresponding to quantization without dither and $q_d(n)$ is the additional quantization error corresponding to an inverted bit. In the interval $0 \leq |e(n)| \leq 1$, the magnitude of the additional quantization error is $|q_d(n)| = 2|e(n)|$ and if $|e(n)| > 1$ then $|q_d(n)| = 2$. Naturally, inverted bits contribute to the overall quantization noise power and an intuitive counter-move is to let p_d decrease with $|e(n)|$. This can be achieved in the SDD model.

> **Example 3.2** Simulations were made on the first-order modulator for different $p_d(e(n))$. For each choice of $p_d(e(n))$, the modulator was simulated over 2^{12} input samples for 100 randomly chosen constant input signals in [-1,1]. For $p_d(e(n)) = 0$,

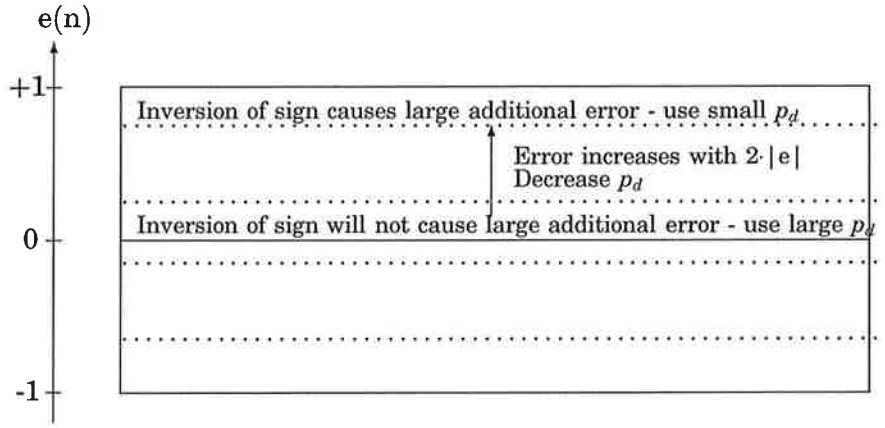

Figure 3.3 Impact of sign inversion on additional quantization error.

the mean squared quantization noise was -4.77 dB. Secondly when $p_d(e(n)) = 0.15$, the mean squared quantization noise was as high as 38 dB[1] and the fraction of the number of inverted samples to the total number of samples was 15%. Finally, $p_d(e(n)) = 0.45(1 - |e(n)|)$ also resulted in 15% inverted bits whereas the mean squared quantization noise was -2.81dB. □

The remainder of this section will be devoted to an analytical analysis of different choices of the inversion probability function, $p_d(e(n))$. Two aspect are of particular interest. One is the expected value of $p_d(e(n))$, which is used as a measure of the ratio of inverted samples to the total number of samples. This quality will be put in relation to the additional noise power introduced by the method and the basic idea is to find a choice of $p_d(e(n))$, which introduces a minimum of additional noise power, yet with large $E[p_d(e(n))]$. This objective needs to be commented on: First, the additional noise power is not the most eligible measure. In reality, only the low-frequency components of the quantization noise are of interest and it would be desirable to utilize some sort of weighting function when evaluating the impact of this quantity. This, however, requires knowledge of the quantization noise correlation function, which depends on both the choice of modulator and the actual input signal. Another obvious question is whether or not $E[p_d(e(n))]$ is a good measure of tone-suppression ability. This will be investigated in Sec. 3.1.

Recall the linearized modulator model in section 2.1. Let the quantization noise consist of q_0 and q_d so that:

$$Y(z) = STF_k(z)U(z) + NTF_k(z)(Q_0(z) + Q_d(z))$$

For quantization without dither, i.e., no inverted bits, the quantization

[1]The reason for this very poor result is that the input signals are allowed to be large.

noise is:

$$q_0(n) = \begin{cases} 1-e(n) & if \quad e(n) \geq 0 \\ -1-e(n) & if \quad e(n) < 0 \end{cases}$$

and the quantization noise power is:

$$V[q_0(n)] = \frac{\Delta^2}{12} = \frac{1}{3} \tag{3.1}$$

Now lets analyse two cases when the signs of some bits are inverted with probability $p_d(e(n))$. To simplify the analysis, it is assumed that the quantizer input is uniformly distributed in [-1,1]. The assumption is not entirely correct. However, it is reasonably accurate for the first-order modulator with constant input. The assumption will be used throughout the rest of this chapter. The quantization noise is:

$$q(n) = \begin{cases} 1-e(n), & \text{with probability} \quad 1-p_d(e(n)), & \text{if} \quad e(n) \geq 0 \\ -1-e(n), & \text{with probability} \quad p_d(e(n)), & \text{if} \quad e(n) \geq 0 \\ 1-e(n), & \text{with probability} \quad p_d(e(n)), & \text{if} \quad e(n) < 0 \\ -1-e(n), & \text{with probability} \quad 1-p_d(e(n)) & \text{if} \quad e(n) < 0 \end{cases}$$

First, consider the signal independent model with $p_d(e(n)) = p_1$, where p_1 is a constant in [0,1]. The total quantization noise power in this case is:

$$\begin{aligned} V[q_1(n)] &= \int_{-\infty}^{0} ((-1-e_1)^2(1-p_1) + (1-e_1)^2 p_1) f_{E_1}(e_1) de_1 + \\ &+ \int_{0}^{\infty} ((1-e_1)^2(1-p_1) + (-1-e_1)^2 p_1) f_{E_1}(e_1) de_1 \\ &= 1 - 2(1-2p_1)E[|e_1|] + E[e_1^2] \end{aligned} \tag{3.2}$$

Observe that the pdf of the quantizer input is not the same as in the case where no bits were inverted. In fact, the pdf is shown in Fig. 3.4. The heights $f_1, f_2 \ldots$ are determined by using a $M/M/1$ model with $\lambda = p_1$ and $\mu = 1-p_1$ [12]:[2]

$$f_i = \frac{1}{2} \cdot p(i)$$

where

$$p(i) = \varphi^i p(0) = \left(\frac{p_1}{1-p_1}\right)^i p(0)$$

is the probability that $i \leq |e_1(n)| < (i+1)$ and

$$p(0) = \frac{1-2p_1}{1-p_1}$$

[2]The time between sign inversions is assumed to be exponentially distributed with an expected value of $\frac{1}{\lambda}$. Likewise, the time between two uninverted bits is also assumed exponentially distributed with an expected value of $\frac{1}{\mu}$.

It is now possible to determine $E[|e_1|]$ and $E[e_1^2]$:

$$\begin{aligned} E[|e_1|] &= \sum_{i=0}^{\infty} p(i) \cdot \frac{1}{2}(2i+1) \\ &= \frac{1}{2}p(0) \sum_{i=0}^{\infty} \varphi^i (2i+1) \\ &= \frac{1}{2}p(0) \left(\frac{1}{1-\varphi} + 2\varphi \frac{d}{d\varphi} \frac{1}{1-\varphi} \right) \\ &= \frac{1}{2} + \frac{p_1}{1-2p_1} \end{aligned} \tag{3.3}$$

$$\begin{aligned} E[e_1^2] &= \sum_{i=0}^{\infty} p(i) \cdot \frac{1}{3} \left((i+1)^3 - i^3 \right) \\ &= \frac{1}{3}p(0) \sum_{i=0}^{\infty} \varphi^i (3i^2 + 3i + 1) \\ &= \frac{1}{3}p(0) \left(\frac{1}{1-\varphi} + 3\varphi \frac{d}{d\varphi} \frac{1}{1-\varphi} + 3\varphi \frac{d}{d\varphi} \left(\varphi \frac{d}{d\varphi} \frac{1}{1-\varphi} \right) \right) \\ &= \frac{1}{3} + \frac{p_1}{1-2p_1} + \frac{p_1}{(1-2p_1)^2} \end{aligned} \tag{3.4}$$

Now, Eqns. (3.2) to (3.4) give the variance of the simple method:

$$V[q_1(n)] = V[q_0] + V[q_{d_1}] = \frac{1}{3} + \frac{2p_1}{(1-2p_1)^2}(1-p_1) \tag{3.5}$$

where the second term is the additional noise power, caused by the inverted bits. Also, the expected value of $p_d(e(n))$ is p_1.

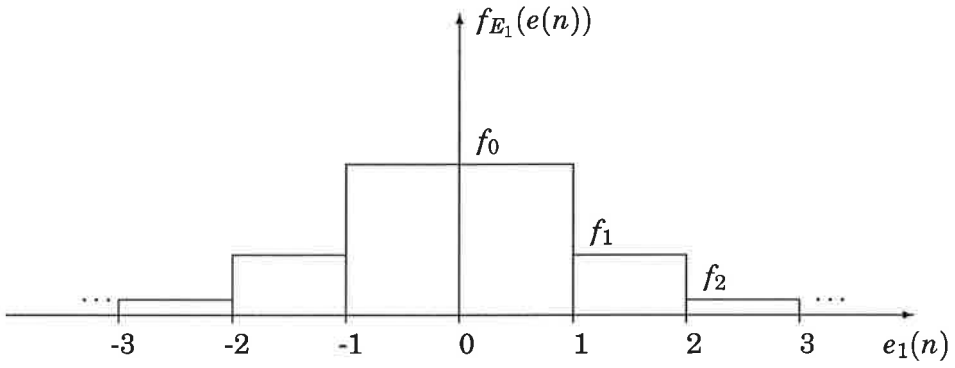

Figure 3.4 PDF for the quantizer input for the signal independent method, where $p_d(e(n)) = p_1$

Next, consider a refined, signal dependent implementation. For instance:

$$p_d(e(n)) = \begin{cases} p_2(1-|e(n)|), & \text{if } |e(n)| \leq 1 \\ 0, & \text{if } |e(n)| > 1 \end{cases} \tag{3.6}$$

for some constant $0 \leq p_2 \leq 1$. In this case, the probability of sign-inversion decreases with the magnitude of the quantizer input, which is expected to entail favourable results. The total noise power of the implementation (3.6) is determined by using the pdf of the quantizer input shown in Fig. 3.5:

$$\begin{aligned} V[q_2(n)] &= 2\int_0^1 \left((1-e_2)^2(1-p_2(1-e_2)) + (-1-e_2)^2 p_2(1-e_2)\right) f_{E_2}(e_2)de_2 + \\ &+ 2\int_1^2 (1-e_2)^2 f_{E_2} de_2 \\ &= 2\int_0^1 \left((1-e_2)^2(1-p_2(1-e_2)) + (-1-e_2)^2 p_2(1-e_2)\right)\frac{1}{2}(1-p_2e_2)de_2 + \\ &+ 2\int_1^2 (1-e_2)^2 \frac{1}{2} p_2(2-e_2)de_2 \\ &= \frac{1}{3} + \frac{p_2}{3}(2-p_2) \end{aligned} \tag{3.7}$$

Furthermore, the expected value of $p_d(e(n))$ is:

$$\begin{aligned} E[p_{d_2}(e(n))] &= 2\int_0^1 p_2(1-e_2) f_{E_2} de_2 \\ &= 2\int_0^1 p_2(1-e_2)\frac{1}{2}(1-p_2e_2)de_2 \\ &= \frac{p_2}{2}(1-\frac{p_2}{3}) \end{aligned} \tag{3.8}$$

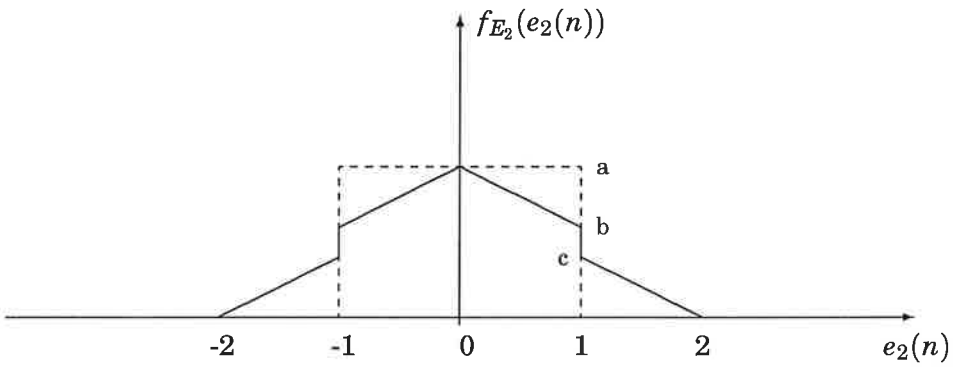

Figure 3.5 PDF for the quantizer input for the refined method of Eq. (3.6), where $p_d(e(n)) = p_2(1-|e|)$. The levels in the figure are $a = \frac{1}{2}$, $b = \frac{1}{2}(1-p_2)$ and $c = \frac{1}{2}p_2$

In Table 3.1, the estimated values of the quantizer noise variance and expected value of $p_d(e(n))$ are compared to simulated values. The simulated values are the mean values of 50 simulations of the first-order modulator over 2^{12} samples with random constant inputs in the range [-0.1,0.1]. In this input range, the estimated values seem reasonable accurate. Fig. 3.6 shows

a plot of the estimated additional noise power against the estimated value of $E[p_d(e(n))]$ for the signal independent and signal dependent methods. Apparently, the signal dependent method introduces less noise power to the modulation than the independent method. In other words; it can invert more bits than the signal independent one for a certain noise floor level. Intuitively, a large number of inverted bits increases the chance to dissolve a limit cycle behaviour. This theory is in agreement with Fig. 3.7 and 3.8, where the PSD:s of the modulator outputs are plotted for the two methods.

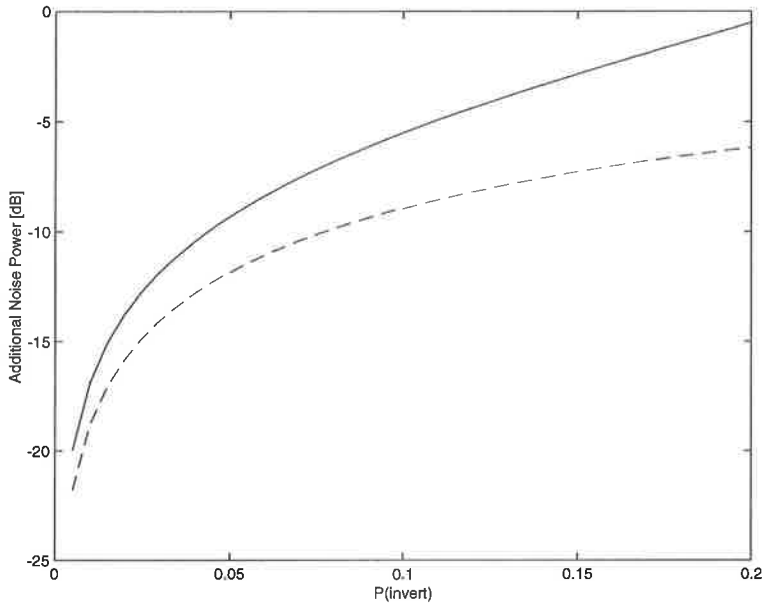

Figure 3.6 Additional noise power vs. $E[p_d(e(n))]$ for the signal independent method (solid line) and the signal dependent method (dashed line).

Signal Independent Method(1)	$p_1 = 0.05$	$p_1 = 0.10$	$p_1 = 0.15$	$p_1 = 0.20$	$p_1 = 0.25$
Estimated $V[q_1]$ [dB]	-3.522	-2.114	-0.687	0.871	2.632
Simulated $V[q_1]$ [dB]	-3.480	-2.114	-0.639	1.004	2.923
Estimated p_{d_1}	0.0500	0.1000	0.1500	0.2000	0.2500
Simulated p_{d_1}	0.0490	0.0990	0.1510	0.2000	0.2501
Signal Dependent Method(2)	$p_2 = 0.10$	$p_2 = 0.20$	$p_2 = 0.30$	$p_2 = 0.40$	$p_2 = 0.50$
Estimated $V[q_2]$ [dB]	-4.015	-3.436	-2.982	-2.623	-2.341
Simulated $V[q_2]$ [dB]	-4.021	-3.410	-2.955	-2.588	-2.255
Estimated p_{d_2}	0.0483	0.0933	0.1350	0.1733	0.2083
Simulated p_{d_2}	0.0480	0.0990	0.1370	0.1731	0.2160

Table 3.1 Comparison between estimated and simulated values of noise power and inversion probability. The simulated values are the mean of 50 simulations with random constant inputs in [-0.1,0.1]

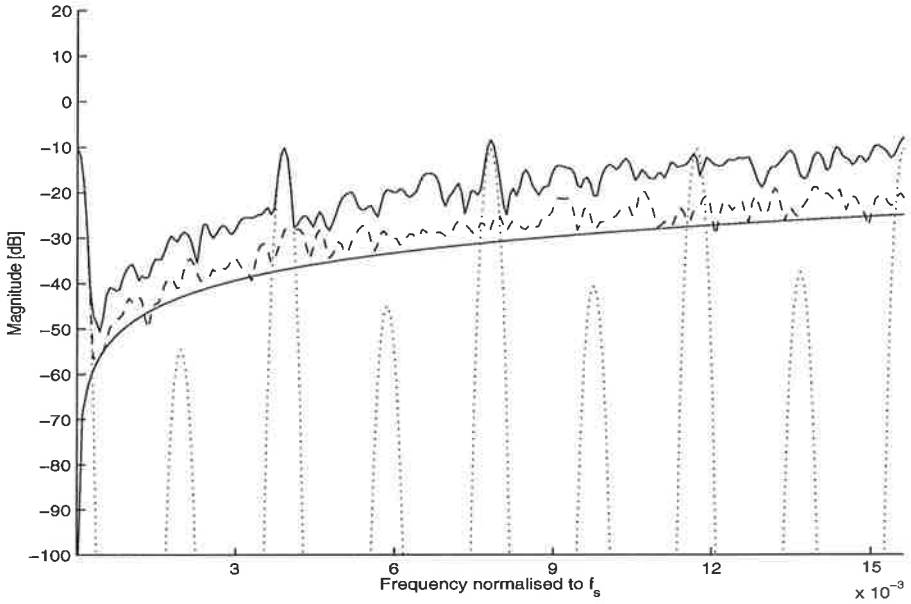

Figure 3.7 PSD of modulator output for $p_d(e(n)) = 0$ (dotted line), $p_d(e(n)) = 0.20$ (solid) and $p_d(e(n)) = 0.5(1 - |e|)$ (dashed), for constant input 1/256. The total quantization noise power for the methods are -4.8 [dB], 0.8 [dB] and -2.2 [dB] respectively and the ratio of inverted bits are 0, 19.9% and 21.7%. The smooth solid line is the corresponding linear approximation of the quantization noise.

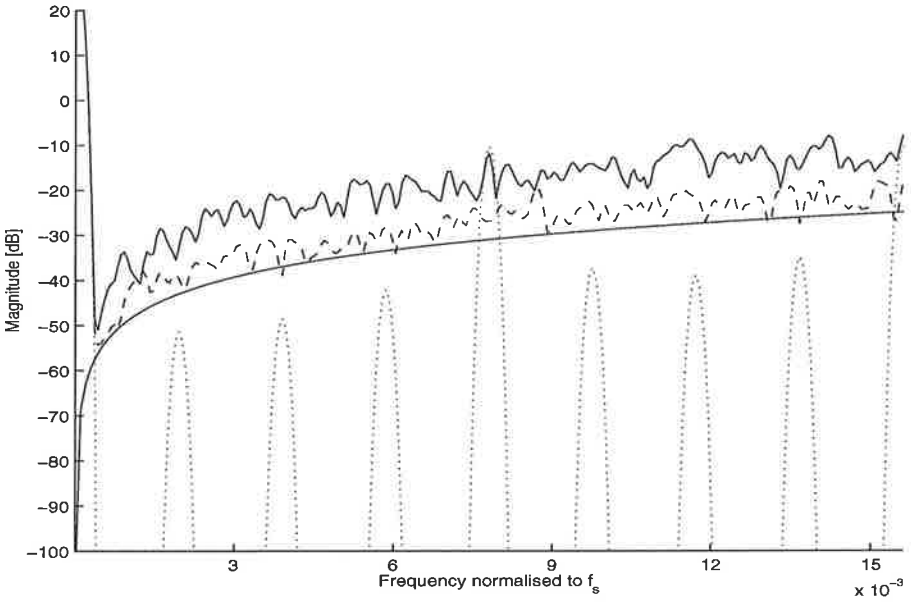

Figure 3.8 PSD of modulator output for $p_d(e(n)) = 0$ (dotted line), $p_d(e(n)) = 0.10$ (solid) and $p_d(e(n)) = 0.5(1 - |e|)$ (dashed), for constant input 1/2-1/256. The total quantization noise power for the methods are -4.8 [dB], 0.6 [dB] and -2.6 [dB] respectively and the ratio of inverted bits are 0, 9.9% and 17.56%.

Minimizing the Quantization Noise Power

It is easy to see that the inversion probability function that has the lowest level of additional quantization noise for some fixed $E[p_d(e(n))]$ is:

$$p_d(e(n)) = \begin{cases} p & for \quad |e(n)| \leq b \\ 0 & for \quad |e(n)| > b \end{cases} \tag{3.9}$$

Obviously, p=1 minimizes the noise power: Cheap sign changes are always made and costly ones are always avoided. However, it is not desirable to always invert codes, even if they are cheap. The purpose is to randomize the output sequence and hence, the natural choice is p=0.5. The quantizer input pdf for this implementation is shown in Fig. 3.9. Furthermore, the quantization noise power and the expected value of $p_d(e(n))$ are:

$$\begin{aligned} V[q_3(n)] &= 2\int_0^b (\frac{1}{2}(1-e_3)^2 + \frac{1}{2}(-1-e_3)^2)\frac{1}{2}de_3 + \\ &+ 2\int_b^{1-b}(1-e_3)^2\frac{1}{2}de_3 + 2\int_{1-b}^{1+b}(1-e_3)^2\frac{1}{4}de_3 \\ &= \frac{1}{3} + b^2 \end{aligned} \tag{3.10}$$

$$E[p_{d_3}(e(n))] = \frac{b}{2} \tag{3.11}$$

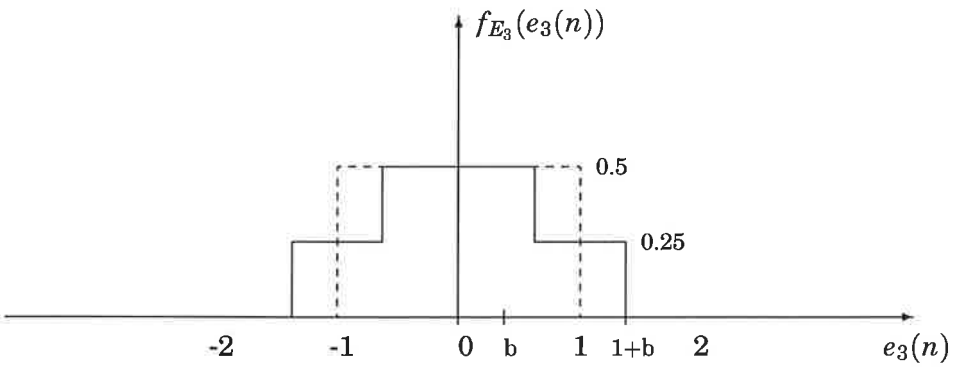

Figure 3.9 Approximate pdf for the quantizer input for the method of Eq. (3.9) with $p = 0.5$.

Fig. 3.10 shows the estimated amount of additional noise power that this - seemingly optimal - method produces in comparison with the two methods investigated previously. Obviously, the method allows the largest number of inverted bits for any fixed level of quantization noise and is, in fact, optimal in this sense. However, this does not necessarily mean that the method is optimal from a tone-suppression point of view.

In Table 3.2, the estimated values of the quantizer noise variance and $p_d(e(n))$ are compared to simulated values. The simulated values are the

mean values of 50 simulations of the first-order modulator over 2^{12} samples with random constant inputs, this time in the range [-0.5,0.5]. There are some interesting things to notice:

- The estimates for the signal independent method do not hold - the simulated values of the noise power are considerably higher.
- The estimates for the signal dependent methods are still reasonable accurate. However; the simulated values of $p_d(e(n))$ are somewhat lower than the estimated, resulting in less quantization noise power.

In explanation, recall that the quantizer input - and the quantization noise in consequence - is not white: The quantizer input, $e(n)$, is the sum of the preceding values of the quantizer input, $e(n-1)$, the negated quantizer output, $y(n-1)$ and the modulator input, $u(n-1)$. When the quantizer input is inverted, the number of time steps needed to "recover" (return to the interval $-1 \leq e(n) \leq 1$) depends on both $p_d(e(n))$ and the magnitude of the modulator input. If the modulator input has opposite sign to the quantizer input this will speed up the recovery. Inversely, if the modulator input has the same sign, recovery will take more time-steps. For instance, consider the extreme case, where $u(n) = 1$ and, consequently, the sum of the modulator input and the negated modulator output is either 0 or 2, depending on whether or not the quantizer input is altered in sign. If the quantizer input is altered in sign, the modulator can never recover and in return, the quantizer noise will increase dramatically. To sum up: Large modulator inputs may slow down the recovery rate of the modulator. As a consequence, the quantizer input pdf tends to decrease for small $e(n)$:s and increase for large $e(n)$:s, resulting in increased quantizer noise for the signal independent method and lower $E[p_d(e(n))]$ for the signal dependent methods.

The line of arguments above suggests that the inversion probability should depend on the actual modulator input. That is, $p_d(e(n))$ should increase with the magnitude of the modulator input, $u(n)$, if $u(n)$ has opposite sign to the filter output, $e(n)$. Conversely, the inversion probability should decrease with $u(n)$, if $u(n)$ has the same sign as $e(n)$. This will be investigated further in Sec. 3.4.

Signal Independent Method(1)	$p_1 = 0.05$	$p_1 = 0.10$	$p_1 = 0.15$	$p_1 = 0.20$	$p_1 = 0.25$
Estimated $V[q_1]$ [dB]	-3.522	-2.114	-0.687	0.871	2.632
Simulated $V[q_1]$ [dB]	-3.143	-1.077	1.308	4.948	8.942
Estimated p_{d_1}	0.0500	0.1000	0.1500	0.2000	0.2500
Simulated p_{d_1}	0.0505	0.1011	0.1508	0.1983	0.2522
Signal Dependent Method(2)	$p_2 = 0.10$	$p_2 = 0.20$	$p_2 = 0.30$	$p_2 = 0.40$	$p_2 = 0.50$
Estimated $V[q_2]$ [dB]	-4.015	-3.436	-2.982	-2.623	-2.341
Simulated $V[q_2]$ [dB]	-4.054	-3.491	-3.016	-2.651	-2.355
Estimated p_{d_2}	0.0483	0.0933	0.1350	0.1733	0.2083
Simulated p_{d_2}	0.0468	0.0903	0.1327	0.1684	0.2031
Signal Dependent Method(3)	$b = 0.10$	$b = 0.20$	$b = 0.30$	$b = 0.40$	$b = 0.50$
Estimated $V[q_3]$ [dB]	-4.643	-4.279	-3.734	-3.069	-2.341
Simulated $V[q_3]$ [dB]	-4.641	-4.283	-3.753	-3.174	-2.718
Estimated p_{d_3}	0.0500	0.1000	0.1500	0.2000	0.2500
Simulated p_{d_3}	0.0497	0.1001	0.1485	0.1916	0.2169

Table 3.2 Comparison between estimated and simulated values of noise power and inversion probability. The simulated values are the mean of 50 simulations with random constant inputs in [-0.5,0.5]

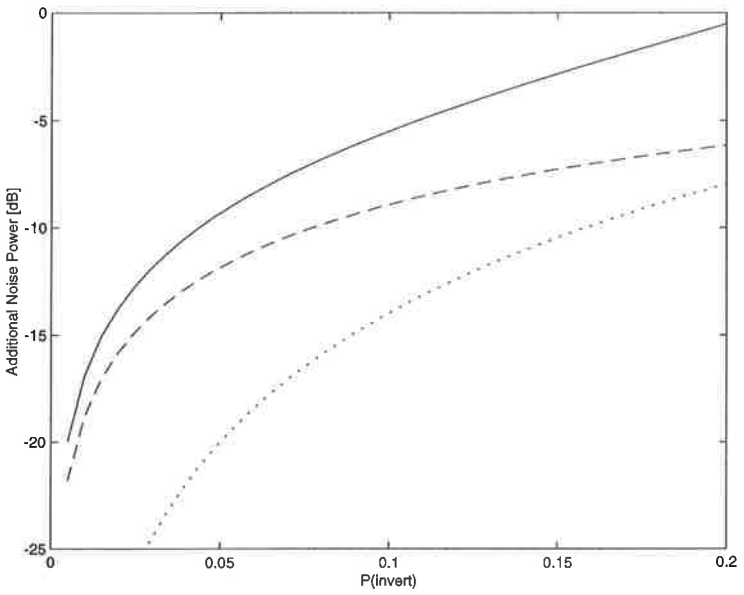

Figure 3.10 Additional noise power vs. $p_d(e(n))$ for the signal independent method (solid line) and the signal dependent methods of Eq. (3.6) (dashed line) and Eq. (3.9) (dotted line).

The Inversion Probability vs. Tone-Suppression

The next question to answer is whether or not the expected value of $p_d(e(n))$ is a good measure of tone suppressing ability. As seen, it is possible to obtain a large number of inverted bits by avoiding to invert bits corresponding to large magnitudes of $e(n)$. However, it may very well be so that these costly sign changes have a greater influence on limit cycle behaviour than the inversion of bits corresponding to small values of $|e(n)|$. Consider for example Fig. 3.11, which shows the PSD of the method of Eq. (3.9). In comparison to the method of Eq. (3.6) (which is seen in Fig. 3.7), the quantization noise power is less and the ratio of inverted bits is higher. Still, the method does not perform well and is, in fact, not capable of tone-suppression at all. To all appearances, this means that the expected value of $p_d(e(n))$ is not a good measure of tone-suppression ability.

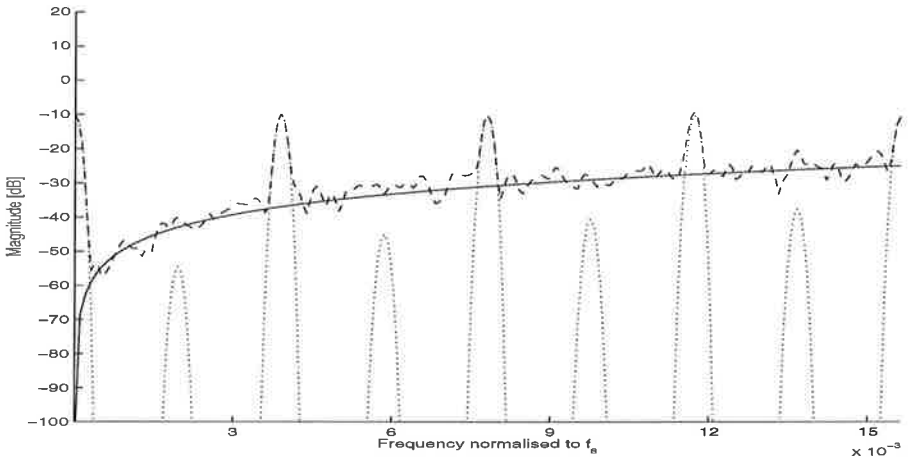

Figure 3.11 PSD of first order modulator output for constant input 1/256. The dotted line is $p_d(e(n)) = 0$. The dashed line is $p_{d_3}(e(n))$ with b=0.5, with a quantization noise power of -2.4 [dB] and a ratio of inverted bits of 25%. The solid line is the linear approximation of the quantization noise.

3.2 A Comparison with the Classical (Additive) Dither Approach

The use of additive dither to eliminate unwanted tones in ΣΔ modulators was described in Sec. 2.1: The dither source generates a random sequence, which is added to the quantizer input. If the dither signal has opposite sign to the quantizer input and exceeds it in magnitude, it will alter the output in sign. It is common to use dither signal with Rectangular Probability Distribution (RPD) or Triangular Probability Distribution (TPD), where it is claimed in [6] that RPD dither is to be preferred in ΣΔ modulation. Examples of simulations of the usual first order modulator with additive RPD dither are displayed in Fig. 3.12, together with the linear approximation. In this case, RPD dither in $[-\frac{1}{2}, \frac{1}{2}]$ is sufficient to linearize the modulator. According to Eq. (2.3), the addition of dither should cause the noise power to increase, whereas in Fig. 3.12, the dithered modulator quantization noise spectrum seems to be equal to the linear approximation. One possible explaination is that the additional noise power can be found at higher frequencies, possibly around $\frac{f_s}{2}$ and that this is caused by the very persistent $\overline{10}$ limit cycle of the first-order modulator. The total noise power, however, is in accordance with Eq. (2.3).

RPD and TPD Dither

An obvious question is how the model of Fig. 2.8 with additive dither is related to the SDD model of Fig. 3.1. To analyse that relationship, consider a general rectangular probability density function

$$f_R(d(n)) = \begin{cases} \frac{1}{b-a}, & a \leq d(n) \leq b \\ 0, & \text{otherwise} \end{cases}$$

The dither signal, $d(n)$, alters the sign of an output element if $d(n)$ has

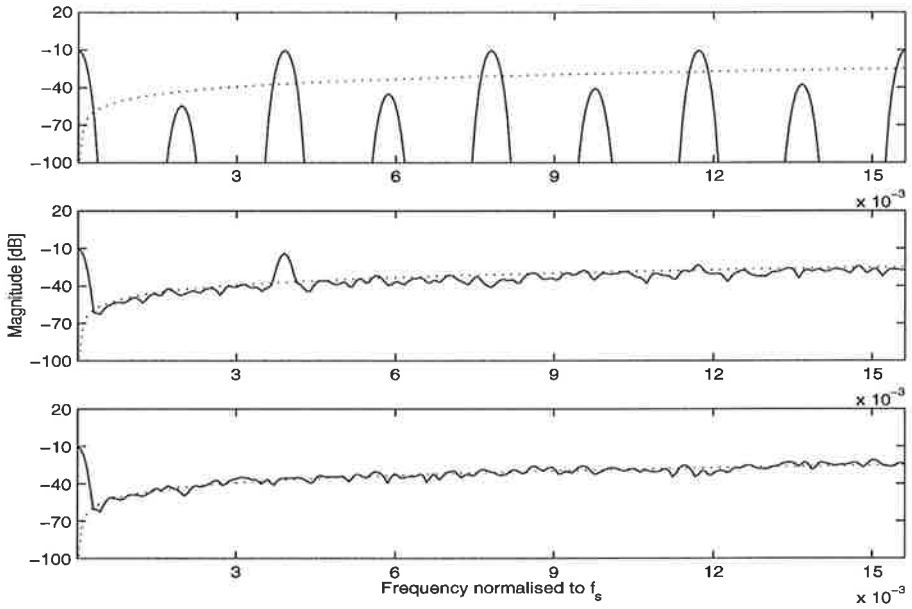

Figure 3.12 PSD of first order modulator output. a) no dither. b) RPD dither with magnitude 1/4. c) RPD dither with magnitude 1/2.

opposite sign to $e(n)$ and exceeds it in magnitude. Hence:

$$p_d(e(n)) = \mathrm{P}\Big((|d(n)| > |e(n)|) \wedge (\mathrm{sgn}(d(n)) \neq \mathrm{sgn}(e(n)))\Big) \quad (3.12)$$

In order not to corrupt the mean of the modulated signal, the RPD should be centered around zero, i.e. $a = -b$. This means that the probability that the dither signal has opposite sign to $e(n)$ is always 0.5 and Eq. (3.12) becomes:

$$\begin{aligned} p_d(e(n)) &= \mathrm{P}\big(|d(n)| > |e(n)|\big) \cdot \mathrm{P}\big(\mathrm{sgn}(d(n)) \neq \mathrm{sgn}(e(n))\big) \\ &= \frac{1}{2}\mathrm{P}\big(|d(n)| > |e(n)|\big) \end{aligned} \quad (3.13)$$

If $-b < e(n) < 0$ then (refer to Fig. 3.13)

$$p_d(e(n)) = \int_{-e(n)}^{b} \frac{1}{2b}\, dx = \frac{1}{2b}(b + e(n)) \quad (3.14)$$

and if $0 < e(n) < b$ then

$$p_d(e(n)) = \int_{-b}^{-e(n)} \frac{1}{2b}\, dx = \frac{1}{2b}(b - e(n)) \quad (3.15)$$

Combining Eq. (3.14) and (3.15) yields

$$p_{d_R}(e(n)) = \begin{cases} \frac{1}{2b}(b - |e(n)|), & |e(n)| \leq b \\ 0, & |e(n)| > b \end{cases} \quad (3.16)$$

for RPD dither in [-b,b]. A similar analysis of additive dither with a trian-

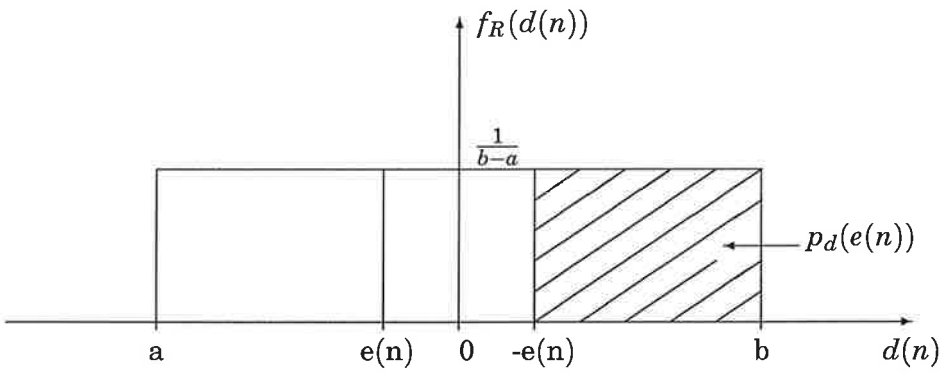

Figure 3.13 $p_d(e(n))$ for RPD

gular probability density function

$$f_T(d(n)) = \begin{cases} \frac{1-\frac{|d(n)|}{b}}{b}, & -b \leq d(n) \leq b \\ 0, & \text{otherwise} \end{cases}$$

yields

$$p_{d_T}(e(n)) = \begin{cases} \frac{1}{2b^2}(b - |e(n)|)^2, & |e(n)| \leq b \\ 0, & |e(n)| > b \end{cases} \tag{3.17}$$

Example 3.3 The uniform distribution assumption for the quantizer input will be used to compare additive RPD dither to the signal dependent implementation of Eq. (3.6). To simplify the analysis, the additive dither is assumed to be in the interval [-b,b], where $0 \leq b \leq 0.5$. In Fig. 3.12 it was also seen that $b = 0.5$ was sufficient to dissolve tones. The estimated values of the noise power and the expected value of the inversion probability are:

$$V[q_R] = \frac{1}{3} + \frac{b^2}{3}$$

$$E[p_{d_R}] = \frac{b}{4}$$

Fig. 3.14 shows the additional noise power plotted against the expected value of $p_d(e(n))$ for the two methods, together with the, in this sense, optimal method of Eq. (3.9). The implementation with additive RPD dither is able to invert a larger amount of bits for a fixed level of quantization noise power and is indeed close to the optimal method. In Fig.3.15, the PSD:s for the two methods are plotted. The SDD implementation of Eq. 3.6

is using $p_2 = 0.5$ while the RPD dither is in [-0.5,0.5]. The performance of the RPD method is clearly better than the SDD method. This is not surprising: The SDD method of Eq. (3.6) with $p_2 = 0.5$ is in fact equal to additive RPD dither in [-1,1], which can be seen by comparing Eq. (3.6) to Eq. (3.16). To all appearances, the RPD method is the most suitable of the two when it comes to linearizing the first-order modulator. □

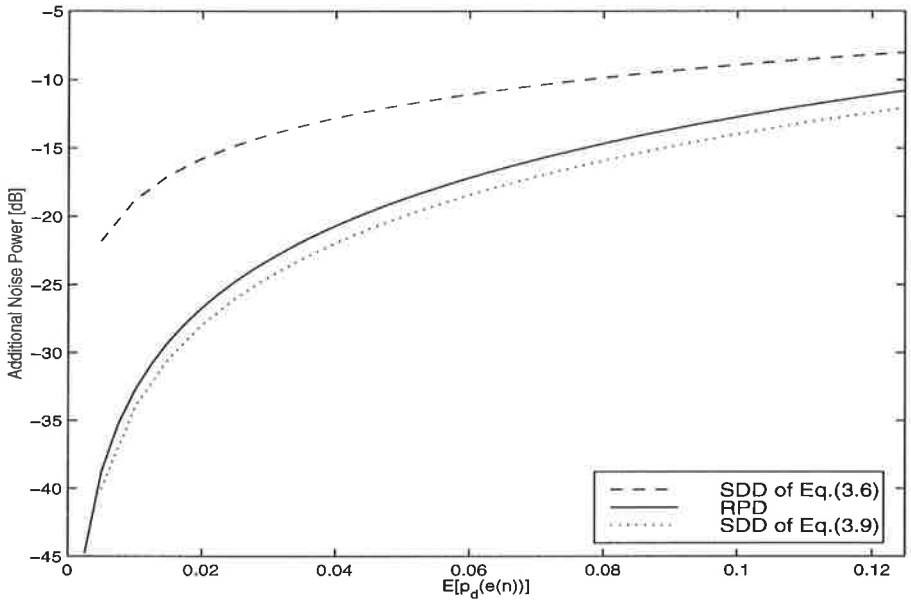

Figure 3.14 Additional noise power vs. $E[p_d(e(n))]$ for additive RPD dither and the SDD implementations of Eq. (3.6) and Eq. (3.9).

Example 3.4 The second-order modulator of Fig. 2.5 was simulated using RPD dither and the SDD method of Eq. (3.6). In comparison to Ex. 4.5, the RPD dither magnitude needed to be larger in order to linearize the modulation. Inversely, the SDD method required approximately $p_2 = 0.3$ to dissolve tones. The reason for this altered behaviour is basically that the distribution of the linear filter output is not the same for the second-order modulator. For instance, the orbit of Fig. 2.15 contains many points corresponding to a filter output larger than 1, whereas the first-order modulator filter output is roughly in [-1,1]. The PSD:s for the two methods are displayed in Fig. 3.16: The two methods have the same quantization noise power and seem equally capable of linearizing the modulator. Fig. 3.17 shows the orbit of the modulator with RPD dither in [-0.78,0.78] to compare with the undithered case of Fig. 2.15. □

Dither with Arbitrary PDF

An additive dither signal with an given, arbitrary, PDF, $f_D(d(n))$ with

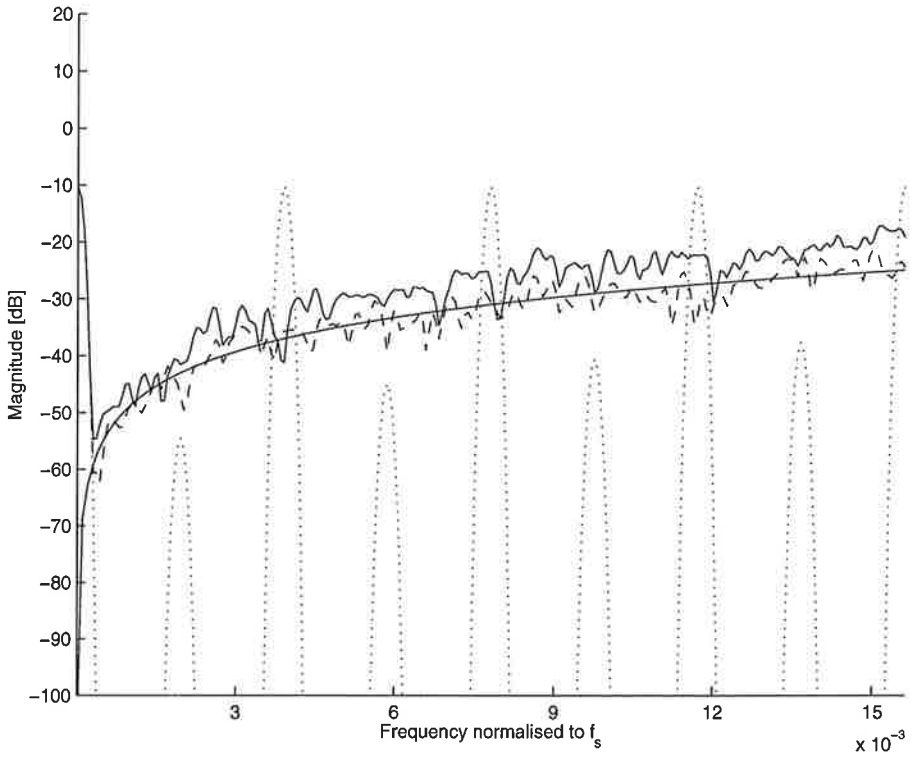

Figure 3.15 PSD of first-order modulator output for additive RPD dither (dashed line) and the SDD implementation of Eq. (3.6) (solid line).

$E[d(n)] = 0$ and $a \leq d(n) \leq b$ corresponds to:

$$p_d(e(n))(e(n)) = \begin{cases} 0, & b < -e(n) \\ \int_{-e(n)}^{b} f_D(x)dx, & 0 \leq -e(n) \leq b \\ \int_{a}^{-e(n)} f_D(x)dx, & a \leq -e(n) \leq 0 \\ 0, & -e(n) < a \end{cases} \tag{3.18}$$

Thus, the SDD model comprises all classes of additive dither pdf:s - it is all a matter of matching $p_d(e(n))$ to a given pdf, using Eq. (3.18). The opposite is, however, not true: Taking the derivative of Eq. (3.18) with respect to e(n) yields:

$$\frac{dp_d(e(n))}{de(n)} = \begin{cases} 0, & b < -e(n) \\ f_D(-e(n)), & 0 \leq -e(n) \leq b \\ -f_D(-e(n)), & a \leq -e(n) \leq 0 \\ 0, & -e(n) < a \end{cases} \tag{3.19}$$

Since $f_D(x) \geq 0$ for all x, the function $p_d(x)$ is always increasing for $x \leq 0$ and decreasing for $x \geq 0$ for all additive dither implementations. This gives a restriction on the inversion probability functions that can be implemented by additive dither.

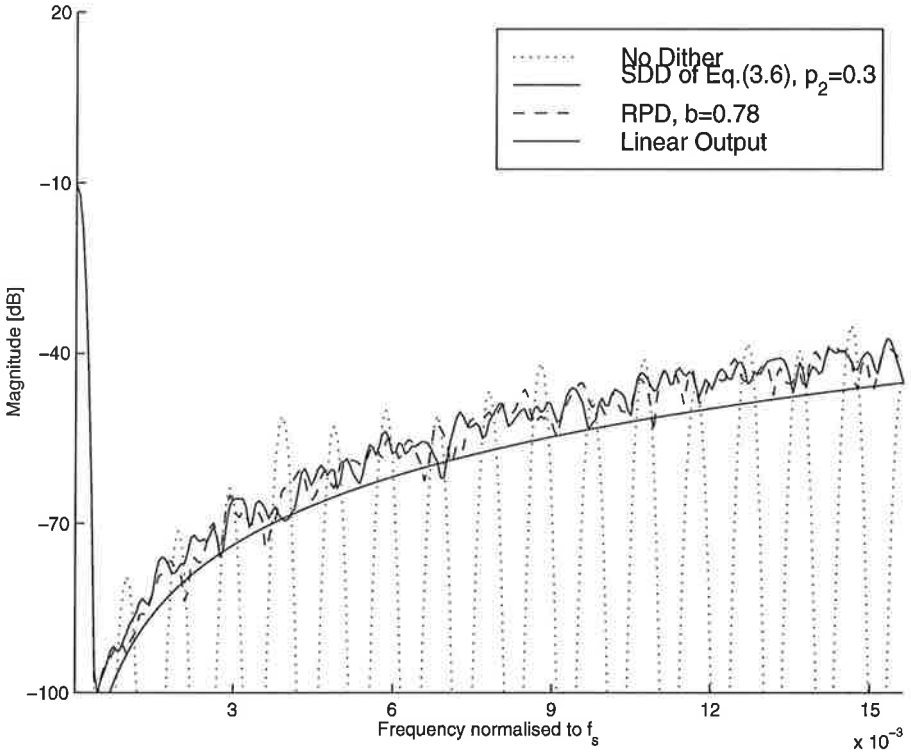

Figure 3.16 PSD:s of second-order modulator output for additive RPD dither and the SDD implementation of Eq. (3.6).

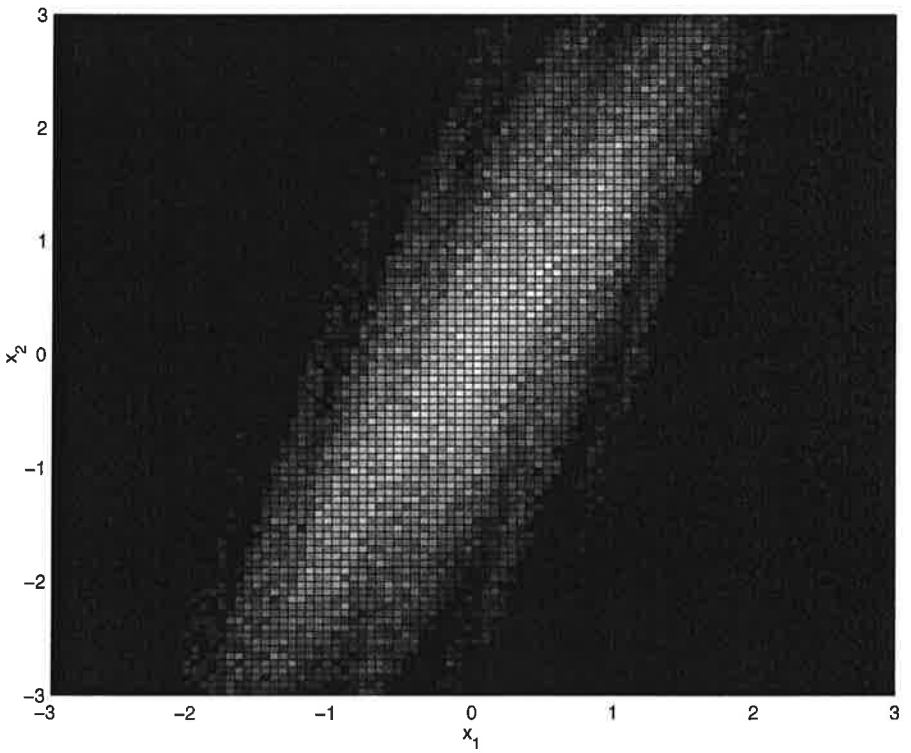

Figure 3.17 Orbit of second-order modulator with RPD dither in [-0.78,0.78].

3.3 Linearized One-Bit Quantization

In Sec. 3.1 it was seen that even if a method introduces a high ratio of inverted bits, it does not necessarily suppress tones. In fact, neither of the methods that proved capable of ton-suppression was the one with the largest $E[p_d(e(n))]$ for a fixed level of noise power. However, the methods in question have another thing in common; they both have an inversion probability function that decreases with $|e(n)|$. Moreover, the inversion probabilities for the successful methods have had an infinite number of possible values, whereas the unsuccessful methods have had only one or two. This aspect of the inversion probability function has an interesting interpretation: The actual 1-bit quantizer takes no consideration to the magnitude of $e(n)$. A positive value of the quantizer input results in an output of +1, no matter how big or small the input is. However, the quantizer together with a proper inversion probability function, may behave linear *in the mean.*

Assume that the quantizer input is constant, $e(n) = k$ and that the inversion probability function is $p_d(k)$. The expected value of the quantizer output, $y(n)$, is then:

$$\begin{aligned} E[y(n)] &= E\left[\text{sgn}((1-p_d(k))\cdot k + p_d(k)\cdot(-k))\right] \\ &= \text{sgn}(k)\cdot(1-2p_d(k)) \end{aligned} \tag{3.20}$$

Using Eq. (3.20), the expected values of the quantizer outputs for the methods previously examined are:

$$E[y_1(n)] = \text{sgn}(k)\cdot(1-2p_1) \tag{3.21}$$

$$\begin{aligned} E[y_2(n)] &= \text{sgn}(k)\cdot(1-2p_2(1-|k|) \\ &= \text{sgn}(k)\cdot(1-2p_2+2p_2|k|) \\ &= 2p_2k + (1-2p_2)\text{sgn}(k) \end{aligned} \tag{3.22}$$

$$E[y_3(n)] = \begin{cases} 1, & e(n) > b \\ 0, & |e(n)| < b \\ -1, & e(n) < -b \end{cases} \tag{3.23}$$

$$\begin{aligned} E[y_R(n)] &= \text{sgn}(k)\cdot(1-2\frac{1}{2b}(b-|k|)) \\ &= \text{sgn}(k)\cdot(1-1+\frac{|k|}{b}) \\ &= \frac{k}{b} \end{aligned} \tag{3.24}$$

Fig. 3.18 displays the impact the different methods have on the mean of the quantizer output. There are some interesting things to notice:

- The method in c) is the method that allows the greatest ratio of inverted bits. However, from the figure it is clear that what the method basically achieves is to introduce - in the mean - a third quantization level. Since modulators with two bit quantizers exhibit tones as well, this can explain the failure of the method.
- The signal independent method in a) is even worse. In the mean, the quantizer output is only scaled.

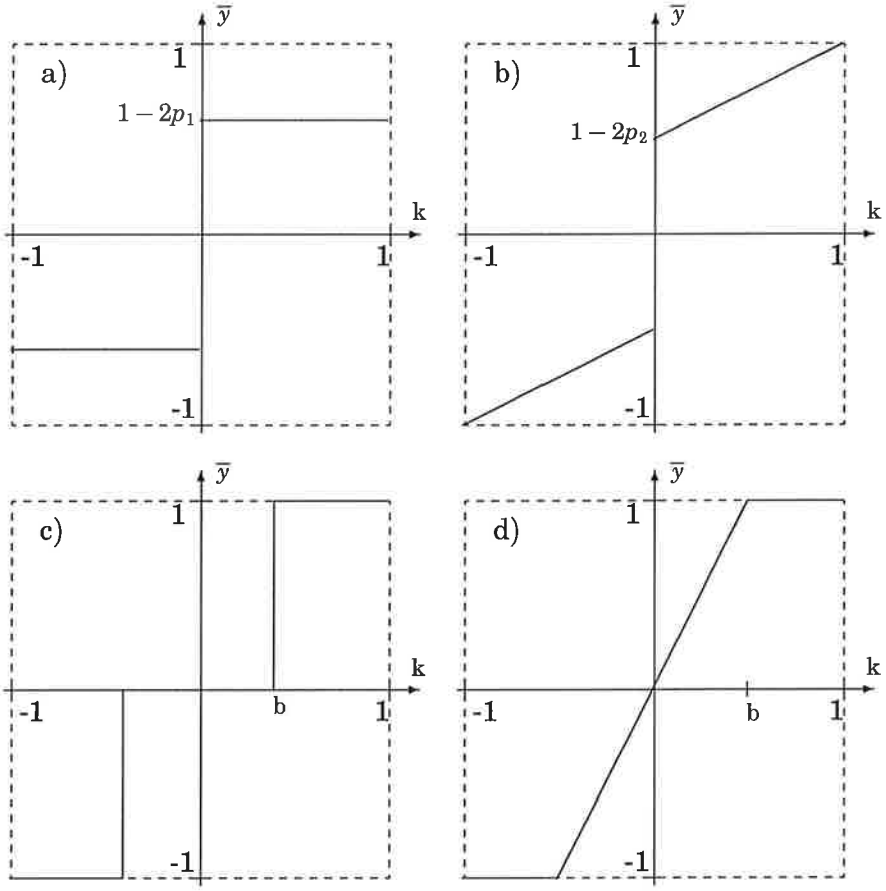

Figure 3.18 Expected value of quantizer output, $\overline{y}$, plotted against constant quantizer input, k, for different $p_d(e(n))$. a) p_1 b) $p_2(1-|e(n)|)$ c) 0.5 if $|e(n)| \leq b$ and otherwise 0 d) additive RPD dither ($\frac{1}{2b}(b-|e(n)|)$).

- The methods in b) and d) are the ones that proved successful in dissolving tones. They also seem to act towards making the quantization linear in the mean. For instance, in the case when $p_2 = 0.5$ and $b = 1$ the two methods are equal and the expected value of the quantizer output is k, i.e., the same as the quantizer input.

Unwanted tones are basically caused by the highly non-linear one-bit quantizer. If the number of quantization levels is large, tones are less likely to occur in the modulator output. Bearing this in mind, the use inversion probability distributions making the quantizer linear in the mean is no doubt appealing.

A possible criterion for linearizing a ΣΔ modulator is that a sufficient ratio of the linear filter output samples are within the interval where the quantization is linear in the mean. For instance; the RPD method allows linear quantization in the interval [-b,b]. For the first-order modulator, the filter output is roughly in [-1,1] and $b = 0.5$ is sufficient for tone-suppression. In the second-order case, there are several points on the orbit outside [-1,1] and, consequently, a smaller ratio of samples within the linear interval. Hence; the linear interval needs to be increased in order to sup-

press tones. This theory is in agreement with the behaviour of the SDD method of Eq. (3.6) as well: In this case, there are two linear intervals, namely [-1,0[and [0,1]. However, the quantization is not linear close to the origin. The first-order modulator has a very persistent $\overline{10}$ limit cycle, which means that the orbit frequently moves across the quantization nonlinearity. In order to linearize the modulation, the method requires $p_2 = 0.5$, for which the origin is contained in the linear interval. As for the second-order modulator, there are several possible limit cycles and the $\overline{10}$ pattern is not as pronounced. This means less crossings of the origin and, in turn, a smaller value of p_2 is required.

3.4 State-Vector Dependent Dither

The signal dependent dither model has been introduced as a method to suppress tones in $\Sigma\Delta$ modulation. It was seen that the result was improved when the filter output, $e(n)$, was taken into consideration when determining $p_d(e(n))$. The attendant question is: Could the result be improved by using the information of the whole state-space as basis for the decision? This will be addressed in the following section.

The vector quantizer model of Fig. 3.19 was proposed in [5]. Modulators which utilize vector quantization is naturally a superset of the SDD model. The basic idea is to map the entire state space into the binary output alphabet. It is pointed out in [5] that the dither signal should preferably be a non-linear projection of the filter states. If a linear projection is used it will be equivalent to a traditional modulator topology with a modified $H(z)$ transfer function, i.e., a linear state-space projection can only change the zeros of $H(z)$. Therefore, the basic ideas from previous sections will still be used, with the modification that $p_d(e(n))$ may now depend on all states.

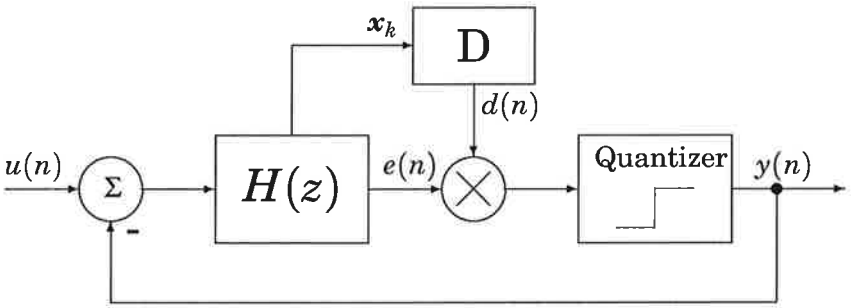

Figure 3.19 ΣΔ modulator with vector quantization

In the following, the basic idea is to use the information to predict the quantization noise in the following time-step, and take this under consideration when deciding upon which bits to invert. Recall again that the quantization noise is not white; an inverted bit at some time step will affect the quantization noise at following time steps as well. This effect was,

by way of example, seen in Table. 3.2, where the performance of the signal independent method was dramatically deteriorated when the constant input signal was large. However, this non-whiteness can also be used to predict the impact an inverted bit at a cerain time-step will have on the quantization noise at following time-steps. It should be pointed out that the following line of arguments is rather intuitive of nature.

Consider the dynamical system model of a ΣΔ modulator:

$$\begin{aligned} \boldsymbol{x}(k+1) &= \Phi\boldsymbol{x}(k) + Au(k) - By(k) \\ e(k) &= C\boldsymbol{x}(k) + Du(k) \\ y(k) &= \mathrm{sgn}(e(k)) \end{aligned}$$

When a bit is inverted at time-step k, the quantizer input at time-step $k+1$ is

$$\begin{aligned} e(k+1) &= C(\Phi\boldsymbol{x}(k) + Au(k) - By(k)) + Du(k+1) \\ &= C(\Phi\boldsymbol{x}(k) + Au(k) + B\,\mathrm{sgn}(e(k))) + Du(k+1) \end{aligned} \quad (3.25)$$

To simplify the analysis it is assumed that if the sign of the quantization input is altered, then $\mathrm{sgn}(e(k+1)) = \mathrm{sgn}(e(k))$ (where $e(k)$ refers to the true value of the quantizer input, i.e., before inversion). Moreover, it is assumed that the sign of the quantizer input at time-step $k+1$ is not inverted. These assumptions are based on the action of the negative feedback of the modulator: For instance, if the quantizer input at time-step k is positive and its sign is altered, the negative feedback will most likely cause $e(k+1)$ not only to have the same sign but also to be greater in magnitude. The second assumption is, however, only applicable on implementations whose inversion probabilities decreases with the magnitude of $e(k)$.

Assuming constant input ($u(k+1) = u(k)$), the quantization error at time-step $k+1$ is:

$$\begin{aligned} q(k+1) &= y(k+1) - e(k+1) \\ &= \begin{cases} (1 - CB) - (C\Phi * \boldsymbol{x}(k) + (CA + D) * u(k)), & e(k) \geq 0 \\ -(1 - CB) - (C\Phi * \boldsymbol{x}(k) + (CA + D) * u(k)), & e(k) < 0 \end{cases} \end{aligned} \quad (3.26)$$

That is, given the assumptions above, the quantization error at time-step $k+1$ is known at time-step k and it is possible to take this information into consideration when deciding upon which bits to invert. In fact, the quantization error can be estimated irrespective of the assumptions, but the expression will be far more complicated and also depend on $p_d(e(n))$.

Example 3.5 The first order modulator of Fig. 2.3 has only one state, and the state is equal to the quantizer input. However, Eq. (3.26) yields:

$$q(k+1) = -(x(k) + u(k))$$

That is, the quantization noise at time-step $k+1$ is small if $x(k)$ and $u(k)$ have opposite signs and are close in magnitude. This

suggests a modified inversion probability function. An example is:

$$p_d(e(n)) = \begin{cases} p(1-|e(k)|)(1-e(k)u(k))), & |e(k)| \leq 1, e(k)u(k) \leq 1 \\ 0, & \text{otherwise} \end{cases} \tag{3.27}$$

where the last factor causes the inversion probability to increase or decrease, depending on the signs of $e(k)$ and $u(k)$. Simulations using this particular inversion probability function unfortunately show little improvement: The quantization noise variance is marginally less, but the difference is negligible. This is especially the case when the modulator input is small, but even if the input signal is quite large there are no significant differences. Modifications on the last factor of Eq. (3.27), such as $\sqrt{1-e(k)u(k)}$ or $(1-\text{sgn}(e(k)) \cdot u(k))$, gave similar results. □

Example 3.6 Consider the second-order feedforward modulator of Fig. 2.5. In this case:

$$q(k+1) = \begin{cases} -(1+x_2(k)) - (x_1(k)+u(k)), & e(k) \geq 0 \\ (1-x_2(k)) - (x_1(k)+u(k)), & e(k) < 0 \end{cases}$$

where $e(k) = x_2(k)$. Apparently, the quantization noise is less if $\text{sgn}(x_1(k)+u(k)) \neq \text{sgn}(x_2(k))$. This fact suggests that the inversion probability function is modified so that the probability increases in case the quantization error at the following time-step is small. For this particular modulator, an example of a modified inversion probability function is:

$$p_d(e(n)) = \begin{cases} p(1-|x_2(k)|)(1-x_2(k)(x_1(k)+u(k))), \\ \text{if } x_2(k)(x_1(k)+u(k)) \leq 1 \text{ and } |x_2(k)| \leq 1 \\ 0, & \text{otherwise} \end{cases}$$

When $\text{sgn}(x_1(k)+u(k)) \neq \text{sgn}(x_2(k))$, the probability of inversion increases with $|x_1(k)+u(k)|$ and otherwise, the probability decreases. Again, however, simulations show little improvement. □

The analysis of SVD dither in this section is in many respects incomplete. There are many possible ways to incorporate the state-vector information in the inversion probability function that were never examined. It is also difficult to estimate the impact of the assumptions made to simplify the analysis. Another problem is that the reduction of quantization noise is quite marginal. For that reason it is difficult to judge if there are any true improvements or if the noise reduction is only accompanied by a reduced ability to suppress tones. However, the use of SVD dither seems intuitively sound and should be object for future research.

4. Application Example: Fractional-N Frequency Synthesis

4.1 Introduction

As mentioned in Sec. 2.2, the grade of success of a method for tone suppressing is highly dependent on the actual application the modulator is being used for. The purpose of this chapter is to put the methods described in previous chapters into practice, i.e., investigate if the suppresion of tones can enhance the performance of a particular system. In this case, the application is digital frequency synthesis. It should be pointed out that the system description will be quite brief; the aim is to define a quality measure and a simulation model, rather than presenting an analysis of the system.

A model of a frequency synthesizer using a PLL is shown in Fig. 4.1. Basically, the action of the PLL is to drive the frequency f_d to be equal to the input reference frequency, f_{ref}, and the output frequency is thus $f_{out} = Nf_i$ [2]. The reference frequency is fixed, which means that the output frequency is controlled by N. It is desirable that N is an integer, which would yield a frequency resolution of f_{ref}. However, by dividing by n sometimes and by $n+1$ at other times, it is possible to, on average, divide by a fractional N such that $n < N < n+1$ and by that improve the frequency resolution. A possible way to control this frequency division is to use a $\Sigma\Delta$ modulator, where the modulator input is the desired fractional offset. However, tones in the modulator output may deteriorate the performance of the system and the objective is thus to investigate if proper dithering can entail a more favourable result.

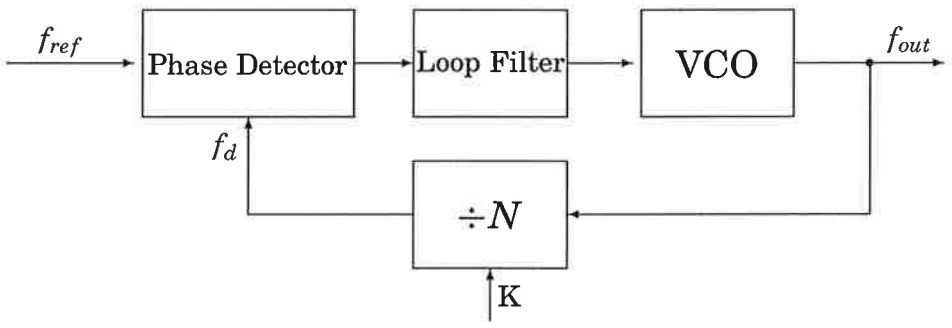

Figure 4.1 Use of a phase-locked loop for digital frequency synthesis

The simulations to follow are utilizing a linearized model of the frequency synthesizer, showed in Fig. 4.2. The input, x, to the $\Sigma\Delta$ modulator is the desired fractional offset and the reference frequency is $f_{ref} = 13MHz$,

yielding an output signal with frequency $f_{out} = (x + 1) \cdot 13MHz$. $G(s)$ is a Butterworth low-pass filter of order 3. The cutoff frequency, f_c, of the low-pass filter is important for the performance of the system as the choice of f_c entails a tradeoff between noise suppression and system speed. It is desirable to have a fast system, i.e., high cutoff frequency. However, f_c must be sufficiently low, in order to meet system requirements on noise suppression. The frequency-domain performance requirements on this particular system are shown in Fig. 4.3, putting an upper limit for the cutoff frequency. This maximum cutoff frequency may vary for different choices of dither and is a natural quality measure of system performance.

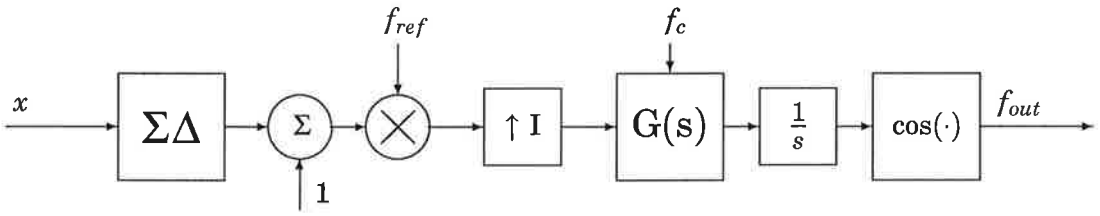

Figure 4.2 The linearized simulation model

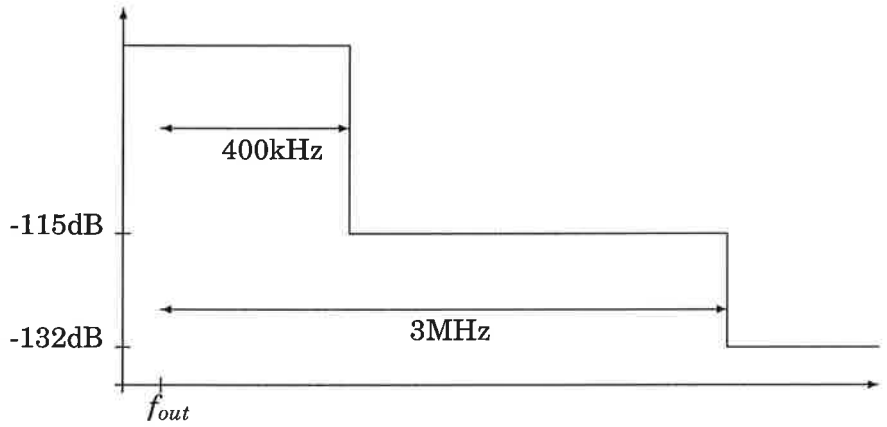

Figure 4.3 Performance specification for the frequency synthesizer

4.2 Simulation Results

In the simulations to follow, an output frequency of f_{out} = 13.2 MHz was generated. Simulations were also made for a few different output frequencies with similar results. In an actual application, however, all possible

output frequencies need to be examined in order to establish the upper cutoff-frequency limit.

Simulations of the first-order modulator are displayed in Fig. 4.4 and Fig 4.5. The undithered system is corrupted by tones and fails to meet system requirements when the Butterworth filter cutoff-frequency is $f_c = 60$ kHz. However, simulated with RPD dither in [-0.5,0.5], the system requirements are met.

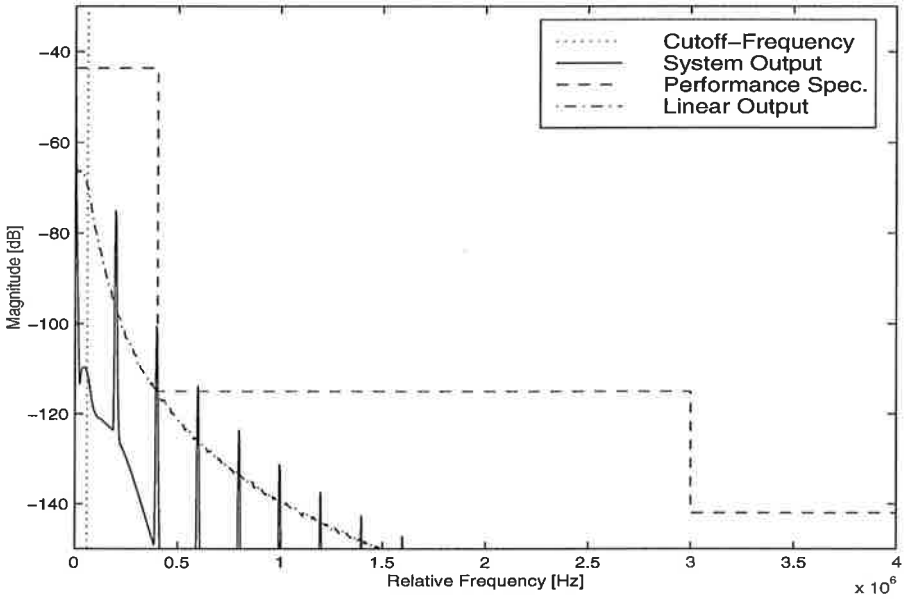

Figure 4.4 System output for first-order, undithered modulator. The Butterworth filter cutoff-frequency is $f_c = 60$ kHz.

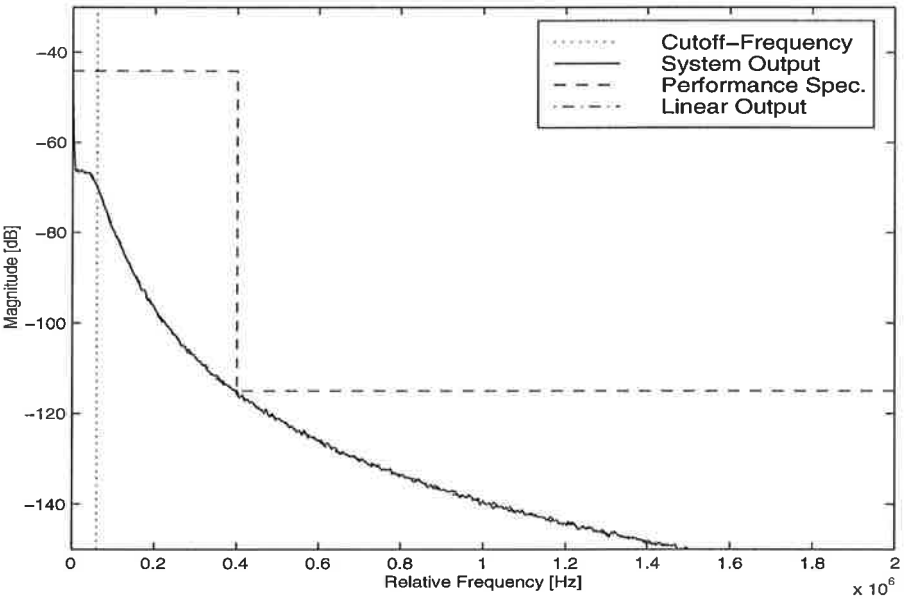

Figure 4.5 System output for first-order modulator with RPD-dither (b=0.5). The Butterworth filter cutoff-frequency is $f_c = 60$ kHz.

Fig. 4.6 displays the output for the second-order modulator system simulated without dither and a cutoff-frequency of 84 kHz. Again, the performance of the system is corrupted by modulator tones. The use of dither (Fig. 4.7) proves capable of sufficient tone-suppression. However, it is clear that the dither signal increase the general noise-floor level in comparison to the linear approximation.

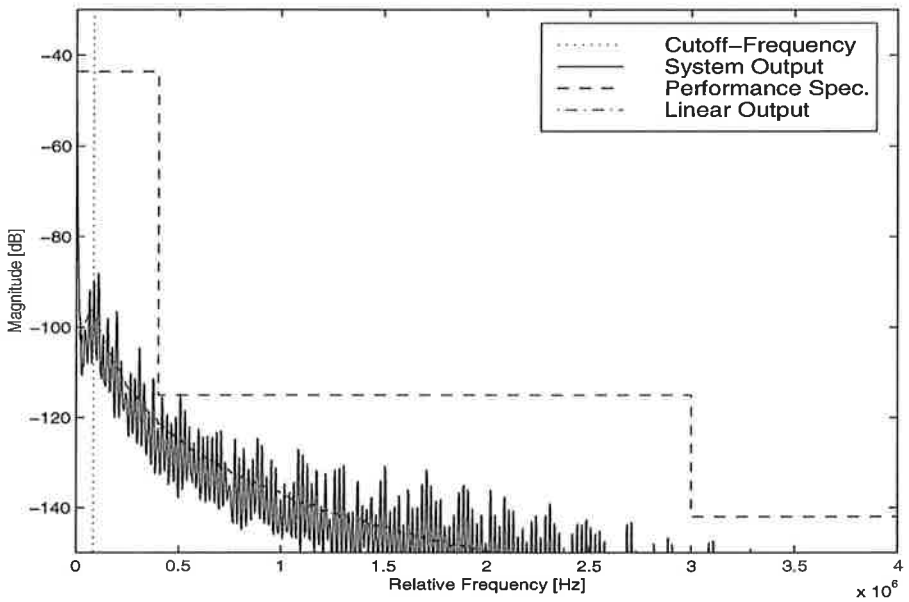

Figure 4.6 System output for second-order, undithered modulator. The Butterworth filter cutoff-frequency is $f_c = 84$ kHz.

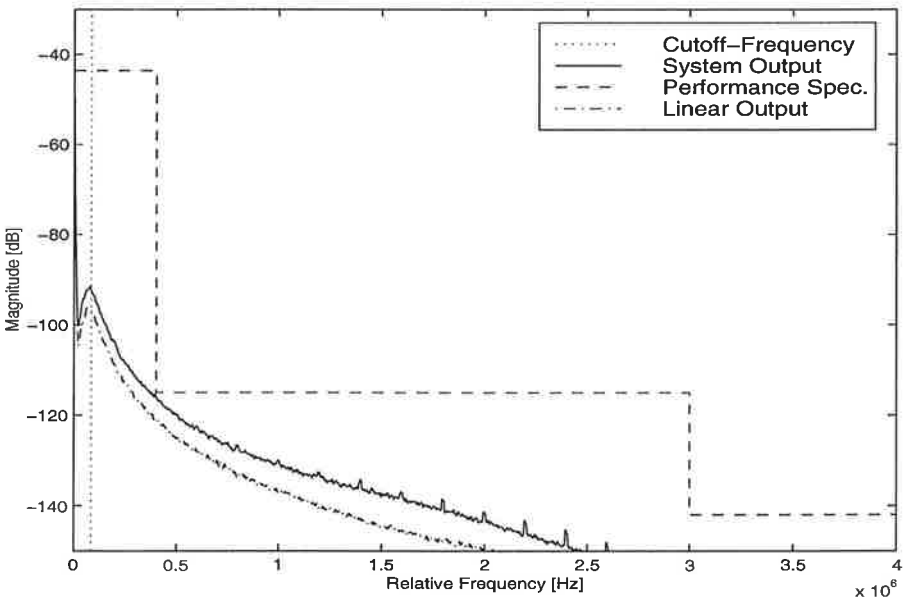

Figure 4.7 System output for second-order modulator with SVD dither based on RPD-dither (b=0.7). The Butterworth filter cutoff-frequency is $f_c = 84$ kHz.

5. Conclusions

The present thesis has investigated the tone problem in ΣΔ modulators. In particular, the use of dither signals to suppress tones was examined.

In Ch. 2, basic properties of ΣΔ modulators were reviewed and a comprehensive picture of the tone-problem was proposed: The behaviour of the linearized modulator was described as the ideal way of function and the tone-problem was considered to be the result of a modulator diverging from that ideal behaviour.

The Signal Dependent Dither model was introduced in Ch. 3. The objectives were to increase the understanding of the effects of different dither signals and to find a dither signal capable of tone-suppression, yet with minimal impact on the overall quantization noise of the modulation. Unfortunately, the assumption that the entropy of the modulation was a good measure of tone-suppression ability proved wrong and an optimal dither signal could not be found. Of the different dither signals examined, the classical approach with additive RPD dither was the most promising for the first-order modulator whereas the results were more ambiguous for the second-order modulator. This change of behaviour was partly explained when the impact of sign-inversions was put in relation to the actual quantization. In fact, this perspective gives an important understanding to the effect of different dither signals: Successful dither signals seem to make the quantization linear in the mean.

A natural extension of the SDD approach was the State-Vector Dependent Dither model. This model allowed dither signals that utilized the information of the entire state-space. Some promising tendencies were seen, however, no significant improvements could be established.

There are several approaches to the tone-problem: The classical dither approach can be seen as an attempt to whiten the quantization noise and by that linearize the modulator. In the present work, the investigation was conducted from a starting point in the need to randomize repeated output patterns. A third possible approach is to act to dissolve the limit cycle behaviour of the modulator. The different approaches are naturally closely related, e.g. when the quantization noise is white there are no limit cycles and no repeated patterns in the modulator output. However, different perspectives on the problem naturally lead to different measures. An important conclusion of the present work is that a comprehensive picture of the tone-problem is necessary for finding successful methods for tone-suppression.

Bibliography

[1] S.R. Norsworthy, R. Screier, G.C. Temes, *Delta-Sigma Data Converters. Theory, design and simulation*, IEEE Press (1997)

[2] T.A.D. Reily *et. al.*, "Delta-Sigma Modulation in Fractional-*N* Frequency Synthesis," *IEEE Journal of solid-state circuits*,Vol 28, No 5, (May 1993)

[3] B. Miller, "Technique Enhances the Performance of PLL Synthesizers," *Microwaves & RF*, (Jan 1993)

[4] P.M. Aziz, H.V. Sorensen, J. Van Der Spiegel, "An Overview of Sigma-Delta Converters," *IEEE Signal Processing Magazine*, (Jan 1996)

[5] L. Risbo, ΣΔ *Modulators - Stability Analysis and Optimization*, Ph. D. Thesis, Technical University of Denmark (1994)

[6] C. Dunn, M. Sandler, "A Comparison of Dithered and Chaotic Sigma-Delta Modulators," *J. Audio Eng. Soc.*, Vol 44, No 4, (April 1996)

[7] S. Hein, "Exploiting Chaos to Suppress Spurious Tones in General Double-Loop ΣΔ Modulators," *IEEE Transactions on Circuits and Systems - II: Analog and Digital Signal Processing.* Vol 40, No 10 (Oct 1993)

[8] R.M. Gray, "Spectral Analysis of Quanyization Noise in a Single-Loop Sigma-Delta Modulator with DC Input," *IEEE Transactions on Communications*, Vol 37, No 6, (June 1989)

[9] T.S. Parker, L.O. Chua, *Practical Numerical Algorithms for Chaotic Systems*, pp. 1-56, Springer-Verlag (1989)

[10] H.B. Griffiths, A. Oldknow, *Mathematics of Models: Continuous and Discrete Dynamical Systems*, Ellis Horwood Ltd (1993)

[11] J.E. Slotine, W. Li, *Applied Non-linear Control*, Prentice-Hall, Inc. (1991)

[12] U. Körner, *Köteori och Tillförlitlighetsteori*, pp. 161-170, Studentlitteratur (1994)