National Library of Canada
Acquisitions and Bibliographic Services Branch
395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale du Canada
Direction des acquisitions et des services bibliographiques
395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file Votre référence

Our file Notre référence

NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

Canada

A Study of Delta-Sigma Oscillator Circuits

Benoît R. Veillette,

(B.Eng 1993)

Department of Electrical Engineering

McGill University, Montréal

August 1995

A Thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of Master of Engineering.

ISBN 0-612-07987-2

Canada

Je doute que toute la philosophie du monde parvienne à supprimer l'esclavage: on en changera tout au plus le nom. Je suis capable d'imaginer des formes de servitude pires que les nôtres, parce que plus insidieuses: soit qu'on réussisse à transformer les hommes en machines stupides et satisfaites, qui se croient libres alors qu'elles sont asservies, soit qu'on développe chez eux, à l'exclusion des loisirs et des plaisirs humains, un goût du travail aussi forcené que la passion de la guerre chez les races barbares. À cette servitude de l'esprit, ou de l'imagination humaine, je préfère encore notre esclavage de fait.

Mémoires d'Hadrien, Marguerite Yourcenar

Abstract

The generation of spectrally pure analog sinewave with predictable characteristics is an important issue of mixed-signal testing. Digital frequency synthesizers exhibit many good features but current implementations are area expensive. This dissertation will study a novel digital frequency synthesizer approach based on digital resonators and delta-sigma modulation. Except for a 1-bit digital-to-analog converter, the analog signal generator implementation is entirely digital allowing precise control over the amplitude and frequency of oscillation. It may therefore be tested using digital methods, making this signal generator attractive for analog built-in self-test (BIST) implementations. Furthermore, this signal generation method is area efficient as it does not require a ROM or a multiplier. However the presence of a delta-sigma modulator in a feedback loop make these circuits non-linear. The limitations of the linear model will therefore will be addressed. We envision that this signal generator can used in communication or for analog testing. One such application, wireless communication system BIST will be presented.

Résumé

La génération d'ondes sinusoïdales de haute qualité spectrale avec des caractéristiques prévisibles est un problème important de la vérification de circuits hybrides analogiques-numériques. Les synthétiseurs de fréquences digitaux comportent beaucoup d'avantages mais leur fabrication requiert un espace très important sur silicium. Cette dissertation présente une nouvelle classe de générateurs d'ondes analogiques utilisant des circuits digitaux résonnants et des concepts de modulation delta-sigma. À l'exception d'un convertisseur analogique-numérique 1-bit, ces générateurs d'ondes analogiques sont entièrement numériques permettant un contrôle précis de l'amplitude et de la fréquence d'oscillation. Ils peuvent donc être vérifiés à l'aide de méthodes digitales, devenant ainsi attrayants pour des applications d'auto-vérification analogique. De plus cette méthode de génération d'onde est économique d'un point de vue d'intégration puisqu'elle ne requiert pas de mémoire morte et ne nécessite pas de multiplication. Par contre, ce circuit n'est pas linéaire de par la présence d'un modulateur delta-sigma dans une boucle de rétroaction. Les limites du modèle linéaire seront donc étudiées. Nous envisageons l'utilisation de ces générateurs de signaux pour des applications en communication et en vérification de circuits analogiques. L'auto-vérification de système de communication sans fil sera donnée en exemple.

Acknowledgements

The appearance of a single name on the cover of this thesis obviously hide the contribution of a number of people, in numerous ways, to the success of this work. While it is common to acknowledge the help of the supervisor, the contribution of Gordon Roberts went well beyond. This research is the result of daily discussions between the two of us and I must point to his enthusiasm and constant availability as a factor of success.

Je remercie énormément mes parents pour leur support moral et financier dont l'origine est de beaucoup antérieure à ces travaux. Aussi plusieurs traits de caractère nécessaires à la recherche m'ont été inculqués par leur exemple.

Je suis de plus reconnaissant envers le Conseil de Recherches en Sciences Naturelles et Génie (CRSNG) qui m'a offert une bourse me permettant de ne concentrer mes soucis que sur la modulation delta-sigma.

Finalement, j'aimerais remercier aussi mes collègues étudiants pour les discussions enrichissantes, Jacek Slaboszewicz et Charles Arsenault pour avoir minimisé les problème logistiques et les membres du *club de café* pour avoir contribué à maintenir la motivation.

Contents

List of Tables

List of Figures

Chapter 1 : Introduction

This thesis is concerned with the development, simulation and experimental characterization of a class of analog signal generator circuits that include a delta-sigma ($\Delta\Sigma$) modulator in a digital resonator. They are generalizations of the low frequency analog oscillator based on a lowpass $\Delta\Sigma$ modulator presented by Lu, Roberts and Johns [1]. These designs are entirely digital except for a 1-bit digital-to-analog converter. They therefore exhibit many digital circuit qualities such as programmability, accuracy, low sensitivity to process variations and mature testing methods. However, because of the use of $\Delta\Sigma$ modulation, the signal frequency band is surrounded by noise. Removing this noise requires processing of the output. These signal generators are therefore best used in applications that have a bandpass frequency response since out-of-band noise is naturally rejected in these systems. We believe they are suitable for testing wireless communication systems and phase-lock loops.

1.1 Motivation

As device dimensions shrink and manufacturing costs decrease, a larger part of the product cost of mixed-signal integrated circuits (IC) is attributed to the test cost. Indeed mixed-signal testers are very expensive and a number of them might be required if the test time is significant. To tackle this concern, researchers are attempting to adapt built-in self-test (BIST) schemes that have been developed for digital circuits to mixed-signal devices. By moving most of the test functions on the die, the complexity of the external tester may be drastically reduced.

The first step toward any such system is to devise an analog signal to stimulate the device under test (DUT). This test stimulus should have a high level of spectral purity and allow a wide range of signal frequencies and amplitudes. It should also be immune to drift caused by temperature or aging if testing in the field is contemplated. Ideally, it should also have

a small area overhead. Finally testing of the analog stimulus source should involve a minimal effort.

A BIST strategy for testing low frequency mixed-signals systems called mixed analog-digital built-in self-test (MADBIST) was presented recently [2]. The MADBIST stimulus is an analog oscillator based on a lowpass (LP) $\Delta\Sigma$ modulator [1]. The analog signal generators presented in this dissertation are generalizations of this circuit that will be referred to as LP $\Delta\Sigma$ oscillator. Specifically, the use of other types of $\Delta\Sigma$ modulation such as bandpass (BP) and highpass (HP) will be explored in this thesis. While the LP $\Delta\Sigma$ modulator limits the operating range to 1% of the clock frequency, a BP $\Delta\Sigma$ modulator may operate at up to almost 50% of the clock frequency. On the other hand, HP $\Delta\Sigma$ modulators operate at half the clock rate. An oscillator using a BP $\Delta\Sigma$ modulator [3] has been presented at the same time as the design contained here [4], but we will demonstrate that this circuit is prone to instability. To our knowledge, the use of a HP $\Delta\Sigma$ modulator in signal generation applications has not yet been reported. When placed in a digital resonator, this $\Delta\Sigma$ modulator allows interesting circuits to be designed for multi-tone signal generation because the basic signal appears very close to its image. The effect of $\Delta\Sigma$ modulators in feedback loops are currently not well understood. Therefore we will lay the foundation of a theory for digital resonators that incorporate $\Delta\Sigma$ modulators.

1.2 Bandpass Analog Circuits Testing

Circuits are labelled bandpass if they respond to sinusoids in a small frequency band and reject signals outside this band. The best example of an analog bandpass circuit is a radio receiver. Indeed this device must demodulate the signal around the carrier frequency while rejecting adjacent channels. Radio receivers include BP filters or tuned circuits to implement this frequency selectivity. Another well known bandpass analog device is the phase-lock loop (PLL). The output of this circuit will track the input signal frequency but only if this frequency is in the PLL locking range.

Currently, signal generators used in the testing of these analog bandpass circuits do not exploit the out-of-band frequency rejection feature of the device under test (DUT). The test sinusoids are spectrally pure over a very wide spectrum. We believe more efficient signal generators may be designed if the bandpass nature of the DUT is taken into account as it has been demonstrated for low frequencies mixed-signal circuits in MADBIST. $\Delta\Sigma$ modulation may in fact be viewed as a trade-off between bandwidth, performance and complexity. By focussing on a smaller frequency band while maintaining the same clock frequency, the performance of a signal generator may be increased and its complexity reduced.

1.3 High-Frequency Sinusoidal Generation

The issue of high frequency sinusoids need not be addressed only with respect to the testing problem. Indeed a critical building block in communication is the mixer. It requires a signal generator, usually identified as local oscillator, which must be capable of generating frequencies up to the gigahertz range with high precision [5] [6]. Considerable research effort is underway in this area, fueled by the increase in wireless communication needs. There is an attempt to reduce the size and the power consumption of the mixer/local oscillator pair. At present, two signal generation schemes are commonly used. The first one is based on the use of a phase-lock loop (PLL) to multiply the frequency of a reference low frequency crystal generated sinusoid. The other possibility offered to designers is to generate a digital sinusoid and then convert it to the analog domain. This second method is however limited to intermediate frequencies.

1.3.1 Phase-Lock Loop Based High Frequency Synthesis

The principle of frequency multiplication using a phase-lock loop (PLL) [7] is shown in Figure 1.1. The phase detector compares the input signal and a representation of the output of the voltage control oscillator (VCO) and generates a signal proportional to the difference in phase. The phase detector output is lowpass filtered to remove short term variations. This signal is then used to adjust the frequency of the VCO. Negative feedback insures that the VCO output remains in phase lock with the input. A counter may be added in the feedback

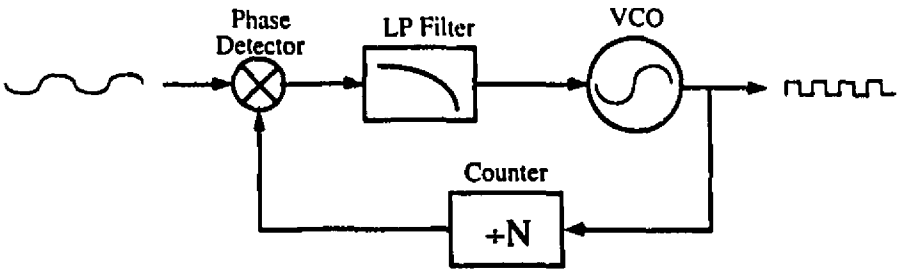


Figure 1.1: Phase-lock loop frequency multiplication.

loop to force the output signal frequency to be a multiple of the input signal frequency. PLLs are used extensively for frequency synthesis in personal communication systems [8]. They are not suitable however as a testing stimulus as they are rather difficult to integrate and depend on precision components.

1.3.2 Direct Digital Frequency Synthesis

The key components of PLLs, the VCO and the LP filter, are analog circuits and as such are sensitive to process variation and are difficult to test. Reducing the proportion of analog circuitry and relying on digital methods is therefore desirable. Direct digital frequency synthesizers (DDFS) generate analog signals using a combination of digital circuity and digital-to-analog converters (DAC) [9] [10]. A common DDFS architecture is shown in Figure 1.2. A counter increments the phase variable ($\phi(n)$) by a frequency-control word F. This value is used to reference the 2^L-sized ROM which contains sine samples. The output is a digital sinewave which is then converted to an analog signal.

The frequency resolution of this device is a function of the number of samples stored in the ROM while the signal-to-noise ratio depends on the word length of each sample and the characteristics of the DAC. This circuit may easily be adapted to generate communication

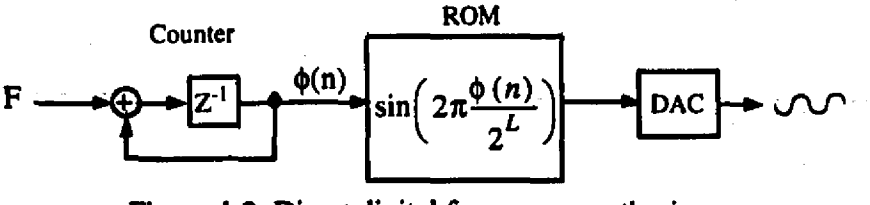


Figure 1.2: Direct digital frequency synthesis.

signals. Indeed frequency modulation is implemented by modifying the frequency-control word F while phase modulation involves incrementing the phase counter. The drawback of the ROM-based DDFS is the large silicon area required for state-of-the-art implementation [11]. Some DDFS schemes do not use ROM but are based on an elaborate signal processing scheme. For example, a circuit has been reported that uses an angle-rotation algorithm implemented as a multiplierless feedforward datapath [12].

1.4 Thesis Overview

Chapter 2 will review the basics of $\Delta\Sigma$ modulation as these notions are essential to the understanding of the new circuits to be presented in later chapters. Chapter 3 presents the low frequency oscillator of Lu *et al.* [1]. The modifications required to adapt the resonator to bandpass $\Delta\Sigma$ modulation and thus generate high frequency sinusoids, will be explained. The issues introduced by the presence of $\Delta\Sigma$ modulators in IIR filters, especially digital resonating circuit will be explored. The presentation of an interesting circuit based on HP $\Delta\Sigma$ modulation will complete the chapter. Experimental results obtained using a field programmable gate array (FPGA) are disclosed in Chapter 4. Finally, in chapter 5, we will discuss a BIST scheme for bandpass type systems of which wireless communication devices are example of. As one will see, the bandpass $\Delta\Sigma$ oscillator plays a central role in that scheme. In Chapter 6 we will summarize this work and take a look at future research topics. In Appendix A, the characteristics of the raised cosine window used throughout this work are described. In Appendix B, the FPGA based prototyping environment used to characterize the circuits is presented.

Chapter 2 : Delta-Sigma Modulation

Delta-sigma ($\Delta\Sigma$) modulation is at the heart of the oscillator circuits to be presented in Chapter 3. The theory of $\Delta\Sigma$ modulation will therefore be explained in this chapter with emphasis on bandpass and highpass $\Delta\Sigma$ modulation. A number of $\Delta\Sigma$ modulator designs will be presented and analyzed.

2.1 Delta-Sigma Modulation Basics

$\Delta\Sigma$ modulators, or oversampling converters, are extensively used in data conversion circuits. They have replaced other schemes, labelled Nyquist converters, in most applications where the clock frequency may be made significantly larger than the signal bandwidth. Even though $\Delta\Sigma$ modulators used in digital-to-analog converters (DACs) and in analog-to-digital converters (ADCs) share many features, we will be mainly concerned with the former as our goal is analog signal generation.

2.1.1 Quantization Noise

The purpose of a $\Delta\Sigma$ modulator in the context of digital-to-analog conversion is to encode the input such that there is a reduced number of quantization levels at the output. Conversion to the analog domain is thus simplified since DACs are easier to design for smaller numbers of quantization levels. The output signal may even be encoded into a single bit stream and thus require a very simple DAC. Indeed, 1-bit DACs are inherently linear because they only have two possible output levels, thereby lowering the amount of distortion in the output signal. However, in any digital system, reducing the number of quantization

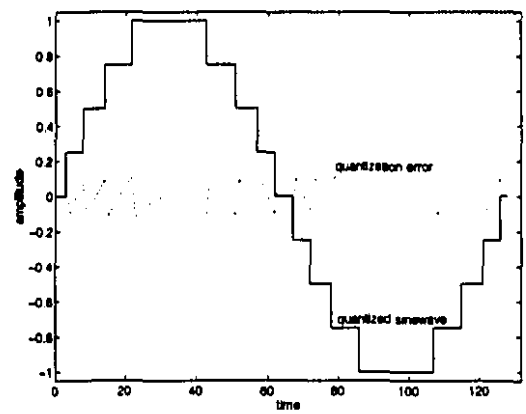


Figure 2.1: Quantized sinusoid and quantization noise.

levels involves adding an error signal corresponding to the difference between the quantized signal and the original signal. This error signal is usually labelled quantization error. This is illustrated in Figure 2.1 for a sinusoid of amplitude equal to one with nine quantization levels. The dotted line is the quantization error.The characteristics of this quantization error depend on the quantizer input signal and therefore, linear analysis may not be used. However it has been observed that in most instances, the quantization error, for a signal below the saturation level of the quantizer, approximates additive white noise and is rightfully labelled quantization noise. It should be emphasized that this assumption is not always valid since the quantization error is sometimes strongly correlated with the input signal. However, it is standard practice to use this assumption in analyzing circuits involving quantizers, most notably delta-sigma modulators. Results obtained from simulation agree to a large extend with predictions done using the linear model in most situations. For linear analysis, the quantizer shown in Figure 2.2(a) is thus replaced by a two-input summer circuit with one input coming from the original source and the other attached to a signal source Q(z) representing the quantization noise, as shown in Figure 2.2(b).

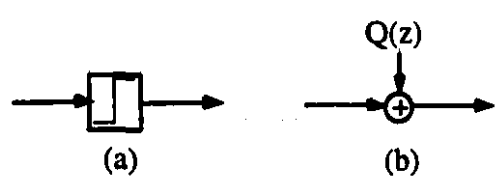


Figure 2.2: (a) One-bit quantizer (b) Linear model of quantizer.

2.1.2 Oversampling and Noise Shaping

The quantization noise power for a 1-bit encoding is so large that it would not be practical with Nyquist conversion. Instead oversampling and noise shaping techniques must be used. The difference between Nyquist conversion and oversampling can be highlighted by observing their operation in the frequency domain. Consider a single tone input to each converter. The power spectral density appearing at the output are those seen in Figure 2.3 (a) and (b). In Nyquist conversion, the sampling frequency (f_S) is equal to twice the bandwidth of the signal (f_b) and therefore, the quantization noise power, illustrated by the shaded area, is distributed between DC and f_b. Consequently, the noise floor, represented by a broken line, is high. When sampling at a larger frequency, the quantization noise power is the same, but is spread over a larger frequency band. Therefore the in-band noise (defined as the noise falling between 0 a f_b) is reduced and the noise floor is lower. The ratio of the Nyquist frequency (f_N) to the bandwidth of the signal frequency band is called the oversampling ratio (OSR):

$$OSR = \frac{f_N}{f_b} = \frac{f_S/2}{f_b}. \qquad (2.1)$$

$\Delta\Sigma$ modulation further filters out most of the quantization noise in a signal band by using noise shaping techniques as shown in Figure 2.3(c). To achieve noise shaping, a feedback loop is built around the quantizer such that the transfer function of the signal to the converter output is unaffected, at least in the signal band. Whereas the transfer function from the quantizer to the output represents a filter with significant attenuation in the signal band.

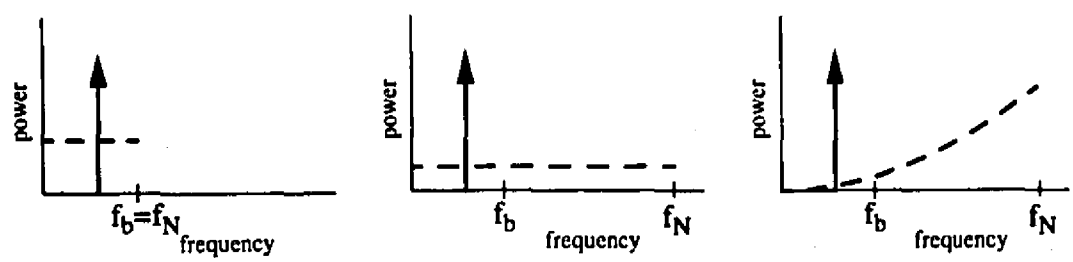


Figure 2.3: Frequency operation: (a) Nyquist conversion. (b) Oversampling conversion (c) Noise shaping.

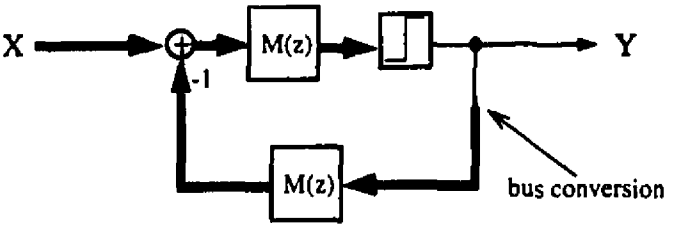


Figure 2.4: Generic delta-sigma modulator.

This topic will be further developed in the following section. A generic ΔΣ modulator is shown in Figure 2.4. M(z) and N(z) are digital filters that implement noise shaping. The output bus width, shown with a thin line, is smaller than the input bus width, represented as a thick line, with the quantizer block doing the conversion. In the feedback path, there is a bus size conversion that is accomplished simply by padding with zero digits. ΔΣ modulators shaping the quantization noise to high frequencies are labelled lowpass (LP) and were introduced first [13] [14]. They are widely used in low frequency data conversion applications such as telephony and digital audio. Bandpass (BP) ΔΣ modulation [15] however is attracting an increasing amount of attention. These circuits shape the noise to both the low and high ends of the Nyquist interval. The signal band can therefore be located at any ratio, up to one-half, of the clock frequency. Finally, a relatively new type of ΔΣ modulator, which we refer to as a highpass (HP) ΔΣ modulator [16], has a signal band centered at the Nyquist frequency.

2.2 Mathematical Treatment of Delta-Sigma Modulation

ΔΣ modulators are highly non-linear circuits because of the presence of a quantizer in a feedback loop. Modelling the quantizer of the network of Figure 2.4 as a source of additive white noise, we arrive at the linear model of the generic ΔΣ modulator shown in Figure 2.5. The transfer function from the quantization noise source (Q(z)) to the output (Y(z)) is called the noise transfer function (NTF), while the response of the output to the input (X(z)) is designated the signal transfer function (STF). ΔΣ modulators are therefore characterized by these two functions, according to:

$$Y(z) = STF(z) \cdot X(z) + NTF(z) \cdot Q(z). \qquad (2.2)$$

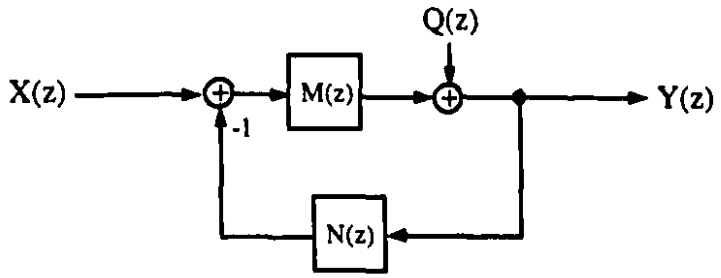


Figure 2.5: Linear model of generic delta-sigma modulator.

For the generic ΔΣ modulator of Figure 2.4, the signal transfer function is

$$STF(z) = \frac{M(z)}{1 + M(z)N(z)} \tag{2.3}$$

while the noise transfer function is

$$NTF(z) = \frac{1}{1 + M(z)N(z)}. \tag{2.4}$$

The order of the NTF function indicates the order of the ΔΣ modulator. NTFs of higher order usually translate to superior noise shaping performance. However not any transfer function may be used for noise shaping. Indeed, ΔΣ modulators may be unstable, even though their linear model predicts otherwise, because of their highly non-linear behavior. Instability is observed when the internal values exceed an arbitrary large bound. Thus far, there has been no absolute method for determining stability of a ΔΣ modulator other than extensive simulation. However, designers have developed rules of thumb to assess stability of ΔΣ modulators. A popular one is named the maximum gain criterion [17] and claims that the probability of stable operation is significantly increased if the magnitude of the NTF is always less than 2 over the Nyquist interval.

Lowpass ΔΣ modulators are usually designed using a zero-pole placement approach. The problem is to design a filter that will yield the largest attenuation in the signal band while leading to a stable ΔΣ modulator implementation. Two methods may be used to design bandpass and highpass ΔΣ modulators. The first one is to use the zero-pole placement approach like for the lowpass case. The second one is to take an optimized lowpass ΔΣ modulator design and use a mapping to convert it to a bandpass or highpass ΔΣ modulator.

2.3 Delta-Sigma Modulator Design using Zero-Pole Placement

Designing a ΔΣ modulator consists in finding suitable transfer functions for the NTF and the STF. It should be noted that not all functions are realizable since delay free close-loops, for example, must be avoided. Stability is also an issue, especially for higher order designs. Also, before designing a bandpass ΔΣ modulator for inclusion inside a digital resonator, one must be aware that, as will be demonstrated later, the STF of this ΔΣ modulator, because it is placed in a feedback loop, must be equal to unity or to z^{-1}. To illustrate the design procedure, a second-order LP ΔΣ modulator and a fourth-order BP ΔΣ modulator will be created. A design that guarantees a unity STF is shown in Figure 2.6 [18]. The NTF of this structure is found to be

$$NTF(z) = 1 + H(z). \tag{2.5}$$

The problem then is to find a suitable transfer function $H(z)$ and a realization of this transfer function with the least amount of hardware. Multipliers should be avoided altogether because of their large silicon area requirements and their significant latency. Software has recently been written to address this problem [19]. However, because the designs presented in this section are very simple, they have been created without the use of this software.

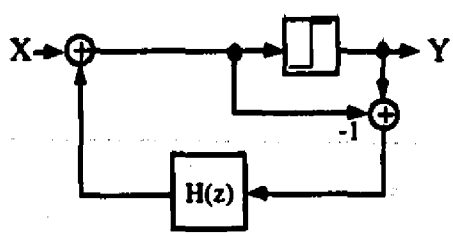


Figure 2.6: Delta-sigma modulator structure with unity signal transfer function.

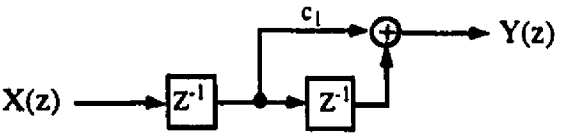


Figure 2.7: Second-order digital filter.

In creating a ΔΣ modulator, the first step is to select a topology for *H(z)*. A second-order network suitable for design of lowpass ΔΣ modulators is the biquad of Figure 2.7. It can be described with the following equation:

$$H(z) = \frac{Y(z)}{X(z)} = z^{-2} - c_1 z^{-1}. \tag{2.6}$$

This biquad, when used in the circuit of Figure 2.6, will lead to an NTF of the form (see Eq. (2.5))

$$NTF(z) = z^{-2} - c_1 z^{-1} + 1. \tag{2.7}$$

It has a pair of zeros that are guaranteed to be on the unit circleand two poles at z=0. The position of the zeros on the unit circle is set by the parameter c_1. Since multipliers should be avoided, this parameter should be a sum of a small number of powers-of-two. A value of $2\text{-}2^{-10}$ was selected for c_1 and it leads to the implementation of Figure 2.8. The NTF has zeros on the unit circle at $+0.01\pi$ and -0.01π radians. The zero-pole plot and the frequency response plot of the NTF are shown in Figure 2.9 (a) and (b) respectively. Frequency is normalized with respect to the Nyquist frequency. The dotted line represent the 6 dB line used for the maximum gain criterion. Note that the magnitude of the NTF at high frequencies exceeds this threshold. If, as we will show, the modulator is stable, then the maximum gain criterion is not an absolute measure of stability. A transmission zero may be seen at 0.01π

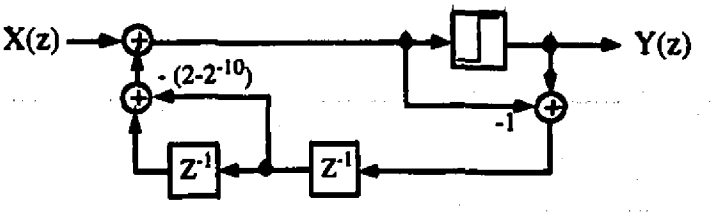
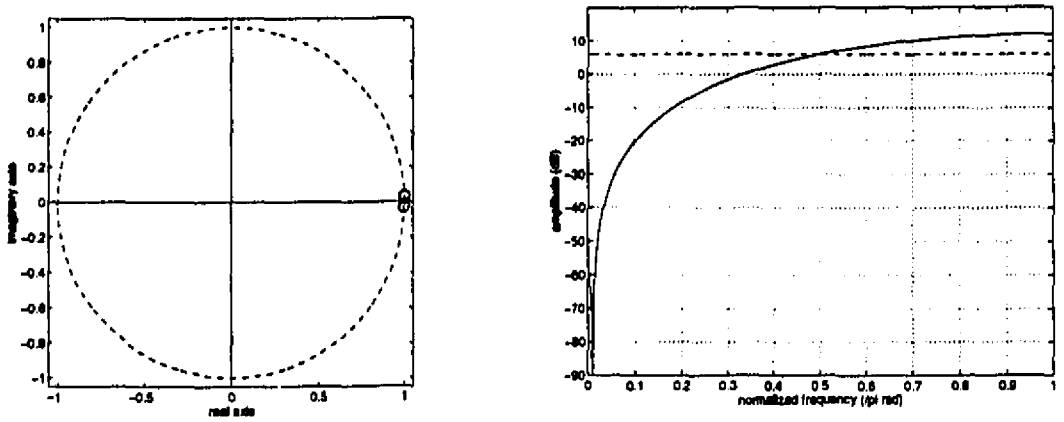


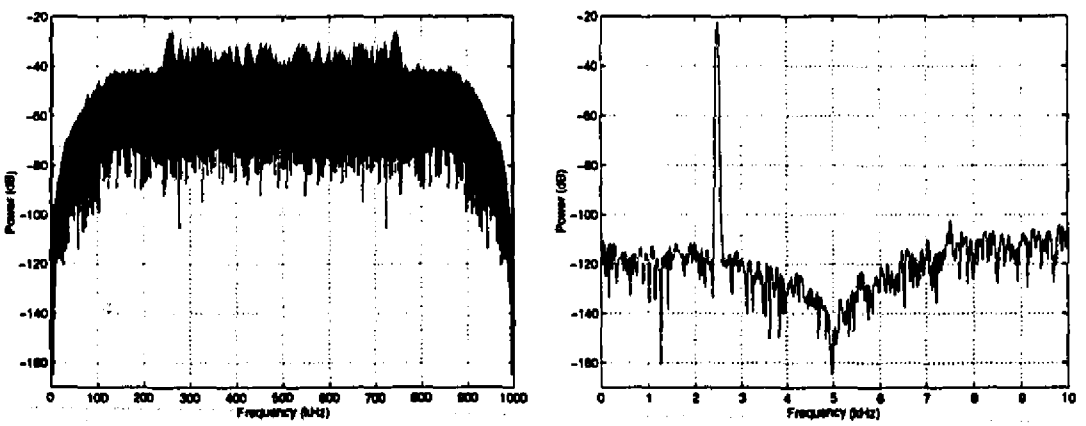
Figure 2.8: Second-order lowpass delta-sigma modulator.



(a) (b)

Figure 2.9: Lowpass delta-sigma modulator NTF (a) Zero-pole plot
(b) Magnitude frequency response.

both on the zero-pole plot and on the frequency response plot. To verify stability, a 2.5 kHz sinewave was applied at the input of the $\Delta\Sigma$ modulator simulated with a 1 MHz clock. Simulation results are shown in Figure 2.10. The power density spectrum was obtained by applying a Fast Fourier Transform (FFT) on 64 K consecutive samples of the output. A raised cosine window was first applied on the samples [20]. The characteristics of this window are described in Appendix A. The sinusoid is visible at the prescribed frequency on the power density spectrum of the signal band in (b). A noise shaping zero is visible at 5 kHz



(a) (b)

Figure 2.10: Simulation of second-order lowpass delta-sigma modulator
(a) Wideband (b) Signal band.

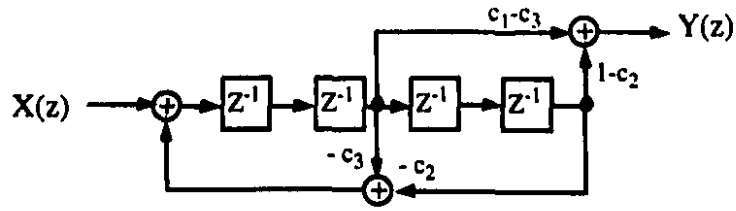


Figure 2.11: Fourth-order digital filter.

corresponding to its desired location of 0.01π radian. Since the simulation results agree closely to the prediction using the linear model, this $\Delta\Sigma$ modulator seems to be stable. A larger number of samples is required to guarantee stability and this is why the experimental setups of Chapter 4 have been left operating for hours.

The design of the bandpass $\Delta\Sigma$ modulator is similar except for the topology of H(z). A fourth-order digital filter that leads to an NTF with interesting properties is shown in Figure 2.11. It contains three parameters and is of the form

$$H(z) = \frac{Y(z)}{X(z)} = \frac{(1-c_2)z^{-4} + (c_1 - c_3)z^{-2}}{c_2 z^{-4} + c_3 z^{-2} + 1}. \tag{2.8}$$

It yields the following NTF when placed in the $\Delta\Sigma$ modulator structure of Figure 2.6 (see Eq. (2.5)),

$$NTF(z) = \frac{z^{-4} + c_1 z^{-2} + 1}{c_2 z^{-4} + c_3 z^{-2} + 1}. \tag{2.9}$$

The zeros are guaranteed to be on the unit circle and the zeros and poles will exhibit not only symmetry along the horizontal axis but also along the vertical axis. It is therefore well suited for bandpass $\Delta\Sigma$ modulators with signal bands centered at $\pi/2$. The coefficients are then selected, keeping in mind that each power-of-two term in a coefficient will require an adder. Coefficient c_1 is selected first since it positions the zeros on the unit circle. The coefficients c_2 and c_3 are set such as to stabilize the BP $\Delta\Sigma$ modulator; the maximum gain cri-

c_1	$2-2^{-8}$
c_2	2^{-1}
c_3	$1-2^{-8}$

Table 2.1: Bandpass delta-sigma modulator coefficients.

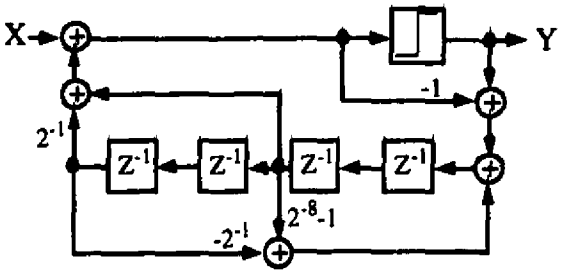
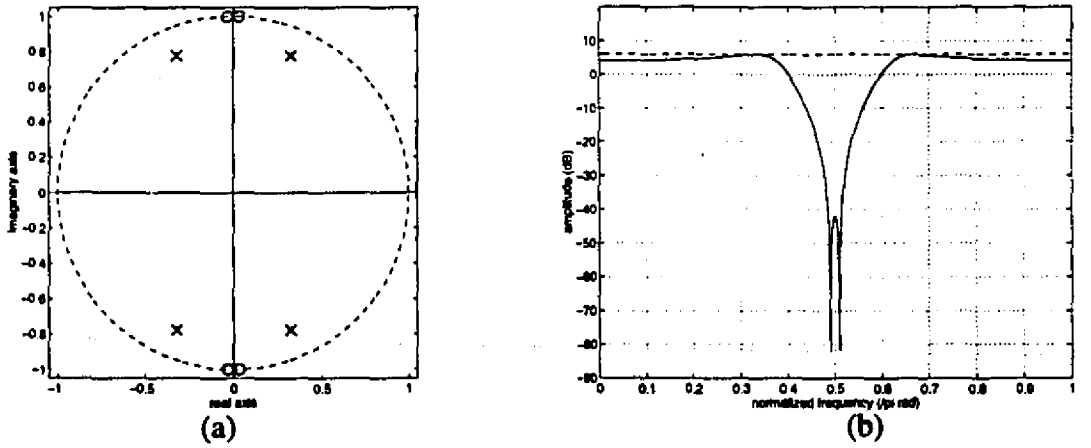


Figure 2.12: A fourth-order bandpass delta-sigma modulator.

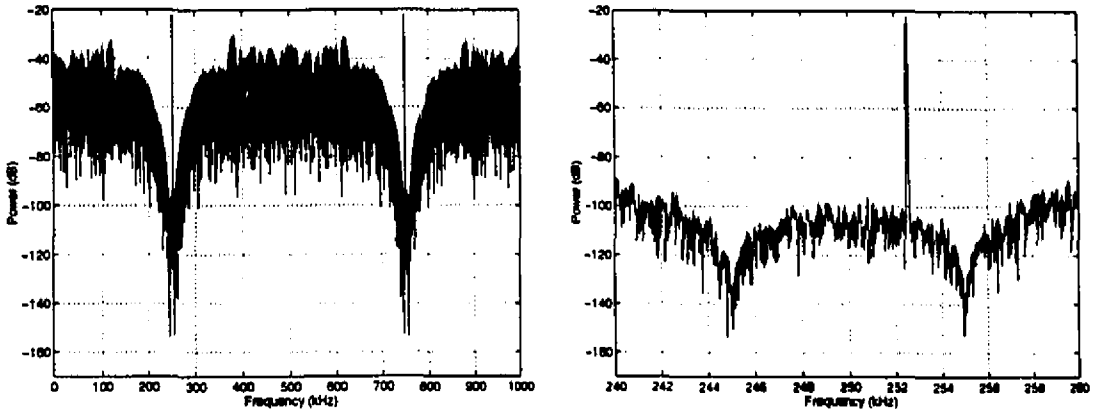
terion is used for this task. The magnitude of the NTF is not allowed to be larger than two. A 4^{th} order bandpass $\Delta\Sigma$ modulator designed with this topology is displayed in Figure 2.12. The selected coefficients are summarized in Table 2.1. It should be noted that all scaling operations use powers-of-two or sums of two powers-of-two and thus no multipliers are required to implement this modulator. Figure 2.13 (a) shows the locations, in the z-plane, of the zeros and the poles of the NTF. The unit circle is represented by a dashed line.The zeros



(a) (b)

Figure 2.13: Bandpass delta-sigma modulator noise transfer function

(a) zero-pole plot (b) magnitude frequency response.



(a) (b)

Figure 2.14: Simulation of fourth-order bandpass delta-sigma modulator
(a) Wideband (b) Signal band.

in the upper half-plane are located on the unit circle at frequencies 0.49π and 0.51π and therefore the signal band is centered around $\pi/2$. The poles in the same half-plane are situated at a radius 0.84 from the origin at angles 0.37π and 0.63π. These poles are required to stabilize the $\Delta\Sigma$ modulator [18]. This 4^{th} order bandpass $\Delta\Sigma$ modulator is in fact equivalent to a second order LP $\Delta\Sigma$ modulator in terms of noise shaping. This is due to the fact that in the LP case all the zeros, both above and below the real axis of the z-plane are located close to the signal band and thus provide a better attenuation of the quantization noise. Figure 2.13 (b) shows the magnitude response of the noise transfer function. Zeros are clearly visible in the signal band. It should be noted that the magnitude of the NTF never crosses the 6 dB line (dashed line) in accordance with the maximum gain criterion. However stability of the $\Delta\Sigma$ modulator must still be verified through by simulation. A 252.5 kHz sinusoid is applied at the input and the $\Delta\Sigma$ modulator is clocked at 1 MHz. Figure 2.14(a) displays the power density spectrum up to the sampling frequency while the $\Delta\Sigma$ modulator passband, between 240 kHz and 260 kHz, is magnified in (b). The tone is present at the desired frequency and two noise shaping zeros are visible at 245 and 255 kHz.

2.4 Delta-Sigma Modulator Design using Mappings

Another method of designing a bandpass ΔΣ modulator centered around π/2 is to perform a mapping on a lowpass ΔΣ modulator. The signal band, where noise attenuation is maximal, may therefore be displaced from low frequencies to around a quarter of the clock frequency. A simple lowpass to bandpass mapping is $z \rightarrow -z^2$ [21]. It will take a zero on the unit circle at $z=e^{\Omega j}$ and move it to both $e^{(\pi/2+\Omega/2)j}$ and its complex conjugate. In the process however, the bandwidth is reduced by a factor of two.

Another mapping for bandpass ΔΣ modulators involves rotating the zeros and poles of the NTF of a LP ΔΣ modulator without creating complex conjugate pairs. This does not seem possible since the zeros and poles of a digital biquad exhibit a horizontal axis of symmetry when there are only real coefficients. However a vertical axis of symmetry is possible with complex coefficients. The benefit of this method is that a bandpass ΔΣ modulator of order N centered at π/2 may be realized by designing a lowpass ΔΣ modulator of order N/2. ΔΣ modulators of lower order are in general easier to stabilize. Using the mapping $z \rightarrow e^{j\pi/2}z$, the LP ΔΣ modulator of Figure 2.8 is turned into the complex BP ΔΣ modulator of Figure 2.15. The NTF has two zeros located on the unit circle on each side of the vertical axis in the upper half-plane at 0.49π and 0.51π. They are placed very close to the upper half-plane zeros of the NTF of the ΔΣ modulator in Figure 2.12. Two poles are placed at the origin. However, because most signal processor use only real arithmetic and since registers in silicon only hold real values, the network must be modified to a realizable form. A set of two registers, R_k and I_k, holding respectively the real and imaginary components is

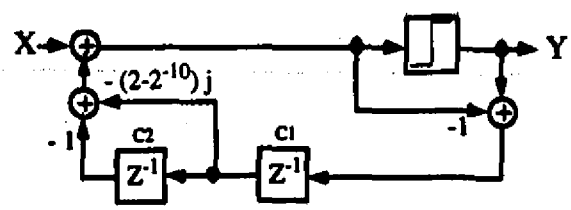
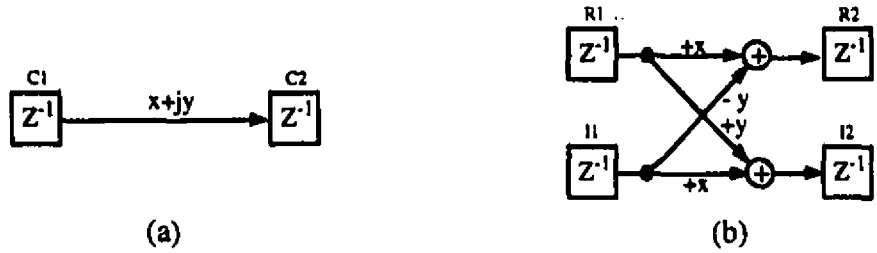


Figure 2.15: Second order complex bandpass delta-sigma modulator.



(a) (b)

Figure 2.16: (a) Complex network (b) Realization of the complex network.

substituted for each complex register (C_k). Complex scaling is handled by decoupling the real and imaginary components according to the following equation:

$$R_2 + jI_2 = (x + jy)(R_1 + jI_1) = (xR_1 - yI_1) + j(yR_1 + xI_1). \tag{2.10}$$

Therefore, grouping the real and imaginary parts, the following two equations are obtained:

$$R_2 = xR_1 - yI_1 \text{ and} \tag{2.11}$$

$$I_2 = yR_1 + xI_1. \tag{2.12}$$

Figure 2.16 illustrates these transformations. The realization of the complex flowgraph of Figure 2.15 is shown in Figure 2.17. The registers C_1 and C_2 are implemented by the register pairs R_1,I_1 and R_2,I_2. There is only one input injection as well as one quantizer since

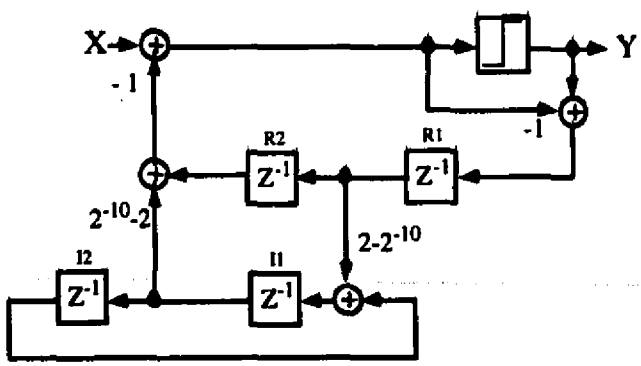
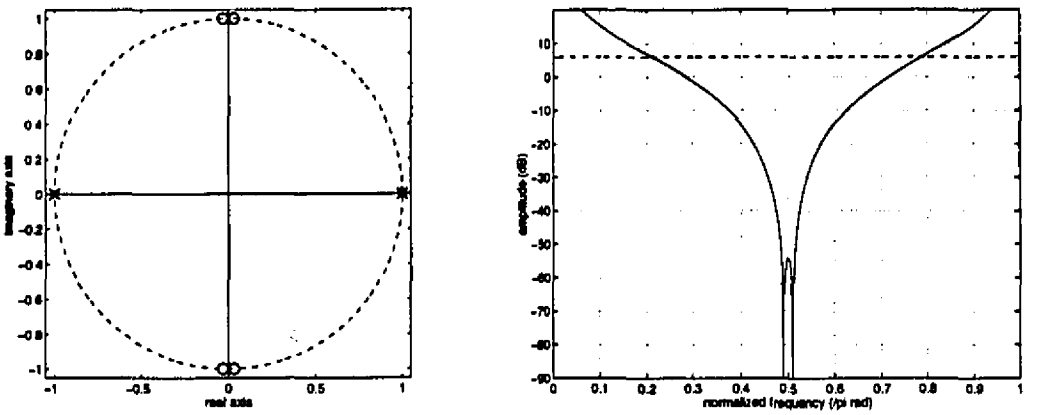


Figure 2.17: Realization of second order complex bandpass delta-sigma modulator.



(a) (b)

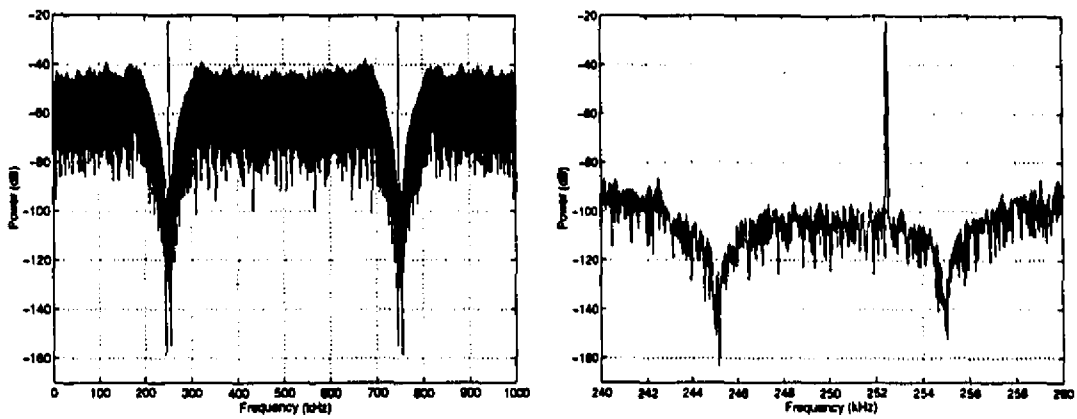
Figure 2.18: Realization of complex bandpass delta-sigma modulator NTF

(a) Zero-pole plot (b) Magnitude frequency response.

the input and output do not have imaginary components. Analyzing the network, we find that the NTF of this realization is

$$NTF = \frac{z^{-4} + \left[\left(2 - 2^{-10}\right)^2 - 2\right] z^{-2} + 1}{-z^{-2} + 1}. \tag{2.13}$$

Since this is a real transfer function, the complex conjugate of the zeros have appeared. More surprising is the creation of a pair of poles. The zero/pole plot and the frequency response plot of the NTF are shown in Figure 2.18 (a) and (b) respectively. The transmission zeros are visible in the signal band. It is interesting to note that because the NTF has two poles on the unit circle it violates blatantly the maximum gain criterion. In simulation the device was clocked at 1 MHz and a 252.5 kHz sinewave was applied at the input. Simula-



(a) (b)

Figure 2.19: Simulation of realization of complex bandpass delta-sigma modulator (a) Wideband (b) Signal band.

tions results shown in Figure 2.19 indicate that the ΔΣ modulator is probably stable. The tone is present at the expected frequency and a dynamic range of about 80 dB may be measured.

Like bandpass ΔΣ modulators, highpass ΔΣ modulators may also be designed from a lowpass ΔΣ modulators using a mapping. The simplest one is to replace each register by a negated register ($z^{-1} \rightarrow -z^{-1}$). A highpass ΔΣ modulator obtained from the classical 2nd order lowpass ΔΣ modulator [22] is shown in Figure 2.20. The reader should note that the inversion at the output is caused by the mapping. The noise is shaped according to the following equation:

$$NTF(z) = \left(z^{-1} + 1\right)^2. \tag{2.14}$$

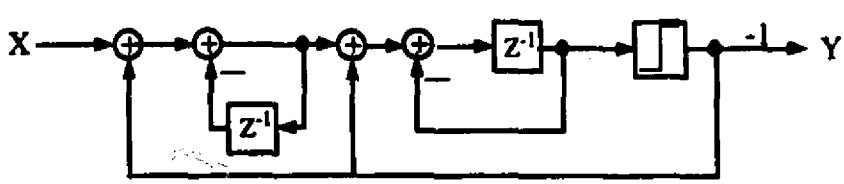


Figure 2.20: Second order highpass delta-sigma modulator.

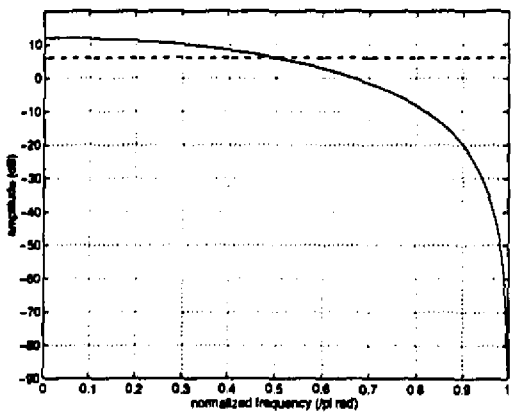
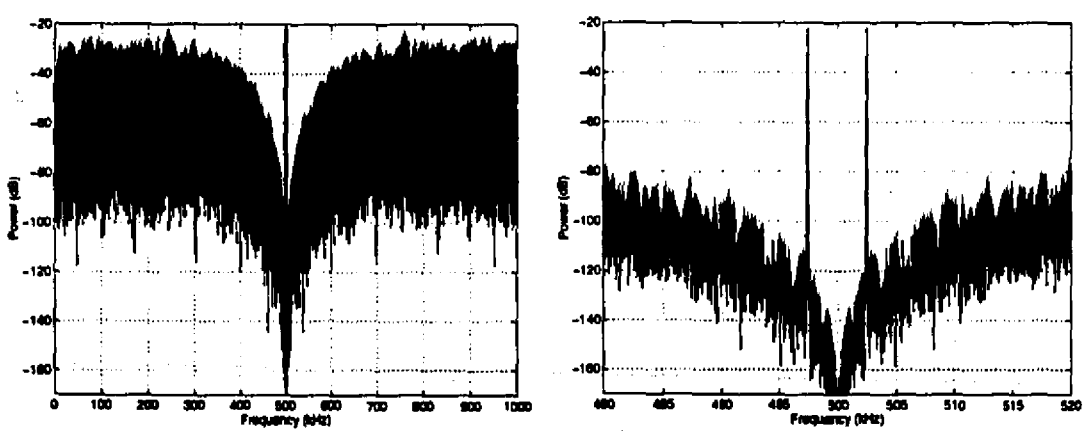


Figure 2.21: Highpass delta-sigma modulator NTF magnitude frequency response.

This NTF has two zeros located at z=-1 and the frequency response is shown in Figure 2.21. Again, stability is verified through simulation. A 497.5 kHz sinusoid was used to stimulate the ΔΣ modulator which was clocked at 1 MHz. Figure 2.22 shows the power density spectrum of the output bit stream. The signal appears at the correct frequency. The tone seen at 502.5 kHz on the plot is the image of the first tone.



(a) (b)

Figure 2.22: Simulation of second-order highpass delta-sigma modulator

(a) Wideband (b) Signal band.

2.5 Summary

The theory of delta-sigma ($\Delta\Sigma$) modulation was introduced at the beginning of this chapter. The concept of quantization noise and noise shaping were first described qualitatively and then a linear model of $\Delta\Sigma$ modulators was presented. Two methods were then explained for the design of $\Delta\Sigma$ modulators. Using the zero-pole placement method, a lowpass $\Delta\Sigma$ modulator and a bandpass $\Delta\Sigma$ modulator were devised. As an alternative method, we show how a lowpass $\Delta\Sigma$ modulator can be mapped to either a bandpass or a highpass form using several different types of transformations. The stability and performance of the $\Delta\Sigma$ modulators were evaluated using simulations. The one-bit encoding feature of $\Delta\Sigma$ modulators is the key to the oscillators that will be presented in the following chapter.

Chapter 3 : Delta-Sigma Based Oscillators

An analog oscillator circuit has recently been presented that mimics the behavior of an LC tank circuit using a low-pass delta-sigma modulator inside a digital resonator circuit. However it is limited in frequency, partly because of the small bandwidth of the lowpass ΔΣ modulator. This device will thus be reviewed and then the modifications required to adapt it to bandpass ΔΣ modulation will be explained. The concept will then be generalized to analog oscillators based on digital resonators that use any flavor of delta-sigma modulation. Finally a three-tone analog signal generator making use of a highpass ΔΣ modulator in a third-order digital resonating circuit will be revealed to illustrate the versatility of the concept.

3.1 Lowpass Delta-Sigma Oscillator

The lowpass ΔΣ oscillator was introduced by Lu *et al.* [1] as a candidate for an analog built-in self-test (BIST) stimulus. The basic idea behind this design is shown in Figure 3.1. On the left side of the digital-to-analog converter (DAC), consisting of a lowpass (LP) ΔΣ mod-

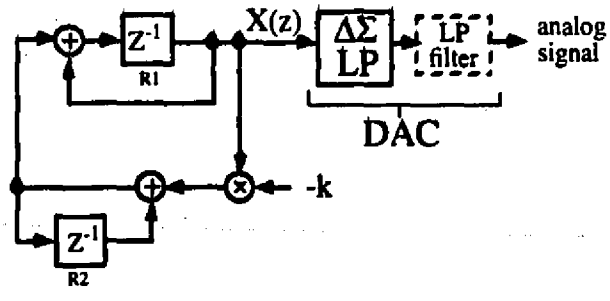


Figure 3.1: Analog signal generation using a digital resonator.

ulator followed by a lowpass filter, is a second-order digital resonator circuit [23] consisting of one digital multiplier, two adders and two delay elements. The register composing the resonator are labeled R1 and R2. A coefficient (k) is used to modify the loop gain. For discussion purposes, the output of the digital resonator is labelled $X(z)$.

3.1.1 Digital Resonator

Examining the circuit Figure 3.1 we find that the characteristic equation of the digital resonator network can be shown to be

$$z^{-2} - (2-k)z^{-1} + 1. \tag{3.1}$$

With $0 < k < 4$, one can show that the roots of this equation remain on the unit circle. Interesting enough, the network will oscillate regardless of the presence of quantization error introduced by the multiplier, as it affects k only. The frequency of oscillation (which we denote as F_o) can be obtained directly from the roots of Eq. (3.1) and satisfies the following equation

$$\cos\left(\frac{2\pi F_o}{F_s}\right) = 1 - \frac{k}{2} \tag{3.2}$$

where F_s is the clock frequency. For mathematical convenience, we shall denote $\Omega_o = \frac{2\pi F_o}{F_S}$ as the normalized angular frequency expressed in radians.

The selectivity of the signal frequency is set by the bus width, as this sets the granularity of k. The amplitude and phase of the sinewave depend on the initial value of register R1 and R2. To obtain the equations relating the sinewave parameters to the initial values, each register is replaced by a combination of a zero initial condition register and an impulse signal

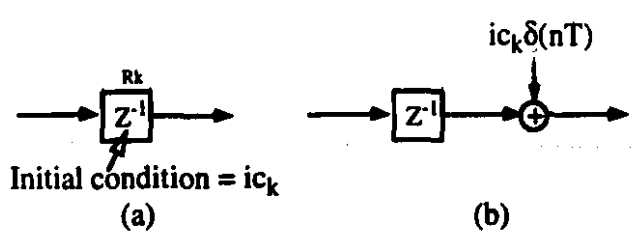


Figure 3.2: (a) Register with initial condition (b) Impulse model.

as shown in Figure 3.2. The output is then expressed as function of the initial conditions in R_1 and R_2 (labelled ic_1 and ic_2 respectively):

$$X(z) = \frac{\left(1-z^{-1}\right)ic_1 + z^{-1}ic_2}{1-(2-k)z^{-1}+z^{-2}}. \tag{3.3}$$

Substituting $\cos\Omega_o = 1-\frac{k}{2}$ and rearranging, we can find an expression for X(z) in terms of the initial conditions and the normalized angular frequency (Ω_0) the result is

$$X(z) = \left[\frac{\left(1-\cos\Omega_o z^{-1}\right)+\frac{\cos\Omega_o-1}{\sin\Omega_o}\sin\Omega_o z^{-1}}{1-2\cos\Omega_o z^{-1}+z^{-2}}\right]ic_1 + \frac{1}{\sin\Omega_o}\frac{\sin\Omega_o z^{-1}}{1-2\cos\Omega_o z^{-1}+z^{-2}}ic_2 \tag{3.4}$$

X(z) in Eq. (3.4) is now in a suitable form to apply the inverse z-transform and get an expression for the impulse response (*x(nT)*) in time steps *nT* as

$$x(nT) = \left[\cos(\Omega_o nT) + \frac{\cos\Omega_o-1}{\sin\Omega_o}\sin(\Omega_o nT)\right]ic_1 + \frac{1}{\sin\Omega_o}\sin(\Omega_o nT)\,ic_2. \tag{3.5}$$

Rearranging Eq. (3.5), the following equation is obtained:

$$x(nT) = ic_1\frac{[\sin(\Omega_o nT+\Omega_o)-\sin(\Omega_o nT)]}{\sin\Omega_o} + ic_2\frac{\sin(\Omega_o nT)}{\sin\Omega_o}. \tag{3.6}$$

Using the cosine law, Eq. (3.6) may further be simplified to its final form as

$$x(nT) = ic_1\frac{\sqrt{2(1-\cos\Omega_o)}}{\sin\Omega_o}\sin\left(\Omega_o nT+\frac{\pi+\Omega_o}{2}\right) + ic_2\frac{1}{\sin\Omega_o}\sin(\Omega_o nT). \tag{3.7}$$

The two terms in Eq. (3.7) are sinusoids of the same frequency and therefore the effect of the initial conditions of the registers may be described as phasors. Phasors are vectors representing the amplitude and phase of an oscillation of a fixed frequency. The characteristics of the oscillation resulting from a combination of sinewave may be inferred from the vector sum of the phasor of each sinewave. The initial condition phasors are shown in graphical

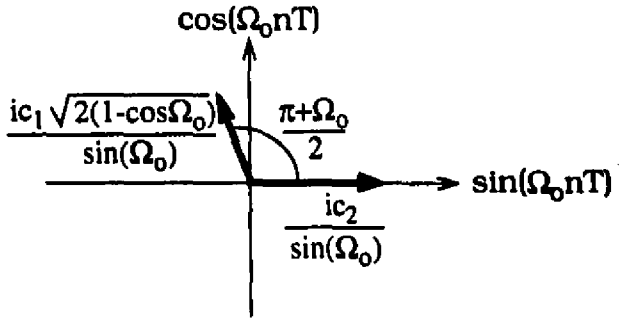


Figure 3.3: Lowpass delta-sigma oscillator initial condition phasors.

form in Figure 3.3. The amplitude and phase of oscillation due to the initial conditions of register 1 and 2 may thus be obtained from the vector addition of each respective phasor. The amplitude and phase of the resulting vector will determine the characteristics of the sinewave. An example is shown in Figure 3.4 for $k = 0.001$ ($\Omega_o = \pi/100$ rad), $ic_1 = 0.5$ and $ic_2 = 0.01$. The resulting oscillation will have an amplitude of 0.591 and a phase of 0.32π ($0.591\sin(\Omega_o nT + 0.32\pi)$).

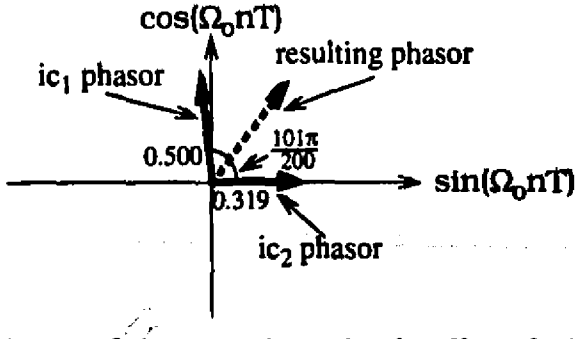


Figure 3.4: Example of the use of phasors to determine the effect of initial conditions.

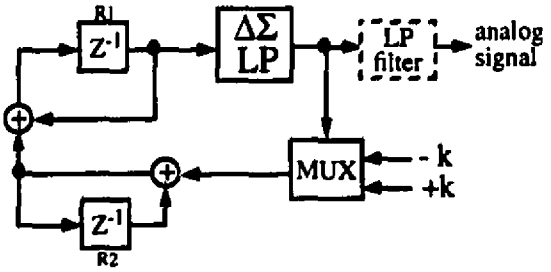


Figure 3.5: Lowpass delta-sigma oscillator.

3.1.2 Delta-Sigma Modulator-Based Low-Frequency Oscillator

The network of Figure 3.1 contains a multiplier unit which is usually expensive to implement and significantly slower than the other blocks. In Figure 3.5, the ΔΣ modulator is thus inserted in the loop so that scaling by coefficient *k* becomes a 1-bit by n-bit multiplication and can be performed by a simple 2:1 multiplexer. The resulting circuit is called a ΔΣ oscillator. For this operation to succeed, the LP ΔΣ modulator must have a unity signal transfer function (STF). The circuit may also accommodate a LP ΔΣ modulator that has an STF equal to z^{-1} such as the classical second-order LP ΔΣ modulator shown in Figure 3.6 but the resonator must be modified by changing the delayed integrator by a non-delayed one. The output of the ΔΣ oscillator is a Pulse Density Modulated (PDM) binary stream so called because the input level is encoded in the local pulse average. A frequency analysis reveals that this PDM stream contains a very high-quality low-frequency analog sinewave and high frequency noise. The analog sinewave signal can then be isolated by linear filtering. Since the ΔΣ modulator has a unity STF, the network is functionally the same as the one in Figure 3.1 and the equations developed for the frequency, phase and amplitude of oscillation in the digital resonator are good approximation for the ΔΣ oscillator in most situation.

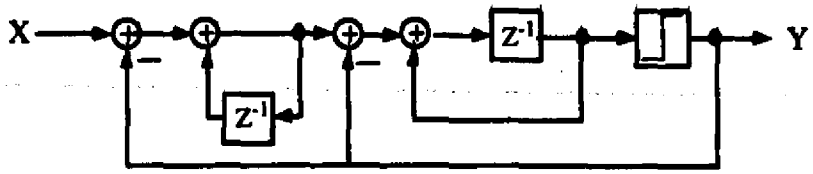


Figure 3.6: Second-order lowpass delta-sigma modulator.

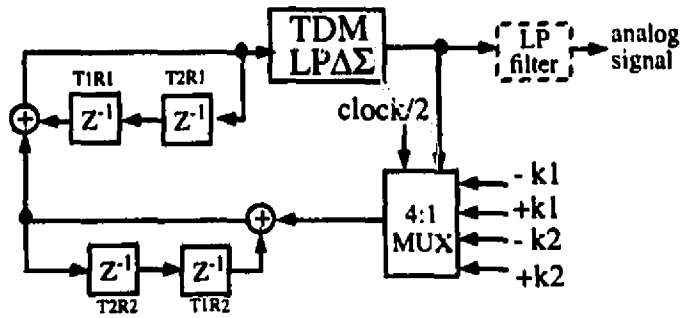


Figure 3.7: Two-tone lowpass delta-sigma oscillator.

3.1.3 Multi-Tone Signal Generation

Multi-tone signals are essential for the measurement of analog circuit parameters such as inter-modulation distortion. The basic design was therefore modified to allow multi-tone signal generation. A hardware efficient technique to realize this has been demonstrated [24]. Time division multiplexing is used and the resulting bit streams are interleaved at the output producing a single bit stream. The price to pay however is a reduction in the effective clock frequency by a factor equal to the number of tones. Figure 3.7 shows the circuit of Figure 3.5 modified for two-tones signal generation using time-division multiplexing. The combinational part of the circuit remains the same except for the 2:1 multiplexer exchanged for a 4:1 multiplexer. All registers have been replaced by a string of two registers. However, on any clock cycle only the content of the first of these two registers has an effect on the combinational circuit. The content of the second one is just passed on to the first to be used in the following clock cycle. Figure 3.8 shows the 2^{nd} order LP $\Delta\Sigma$ modulator of Figure 3.6 adapted to handle a two-tone time-division multiplexed input signal.

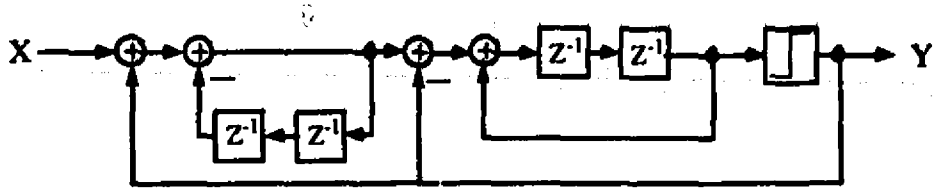


Figure 3.8: Time-division multiplexed second-order lowpass delta-sigma modulator.

3.1.4 Generating Arbitrary Waveforms

Using sinewave features such as zero crossing, arbitrary waveforms may be generated with very accurate frequency control [25]. Figure 3.9 illustrates the basic principle. A sinewave is produced by the ΔΣ oscillator in the lower part and is used to generate another waveform through a mapping function *f(x)*. This digital waveform is then converted to the analog domain by the combination of a ΔΣ modulator and a lowpass filter. Square waves for example are generated with *f(x)* implementing a quantizer function. Triangular waves and sawtooth waves circuits have also been built using this scheme. A time-division multiplexed ΔΣ modulator such as the one shown in Figure 3.8 may be used to save hardware by replacing the two ordinary LP ΔΣ modulator.

3.1.5 Limitations

One of the drawbacks of the LP ΔΣ modulator-based design is that the range of signal frequencies that can be generated is limited to low ratios of the clock frequency. This is because coefficient *k* must be constrained to low values in order for the circuit to remain stable (section 3.3 will cover this in greater details). Experiments have shown that the signal frequency is limited to about 1% of the clock frequency. Furthermore, images of the baseband cannot be used for single tone applications because tones of odd and even images appear

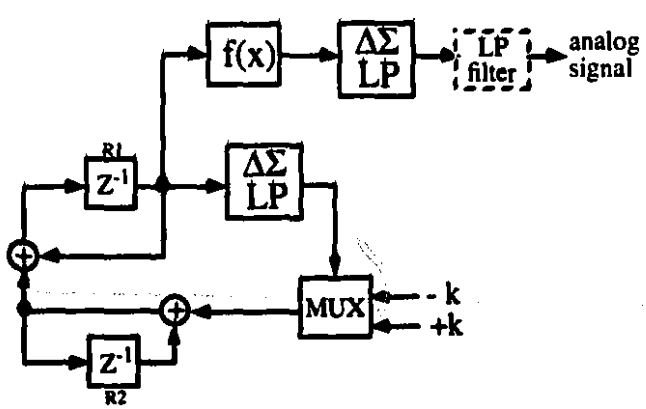


Figure 3.9: Arbitrary waveform generation using lowpass delta-sigma oscillator.

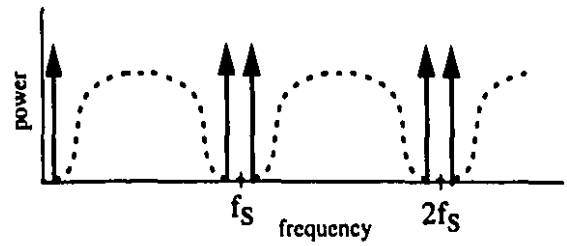


Figure 3.10: Images of the lowpass delta-sigma oscillator.

too close to one another to be separated easily by filtering as illustrated in Figure 3.10. The first and second images are located around F_S with a frequency difference of only $2F_o$ between the two tones.

3.2 Bandpass Delta-Sigma Oscillator

A natural operation to increase the functionality of the LP ΔΣ modulator based circuit of Figure 3.5 is to trade the LP ΔΣ modulator for a BP one. A group claims to have succeeded in obtaining meaningful results with this modification [3] but we have not been able to replicate their simulation. As explained in the previous section, k must be kept to a small value for stability reasons. With large values of k, the amplitude of oscillation may no longer be predicted using the analysis of section 3.1.1 and it may lead to amplitude larger than the saturation level of the ΔΣ modulator, thereby causing instability. The lossless discrete integrator (LDI) resonator of Figure 3.1 must therefore be adapted for bandpass ΔΣ modulation.

3.2.1 Bandpass Digital Resonator

Noting that for $k = 0$ the digital resonator network of Figure 3.1 has 2 poles at DC, it is possible to position these poles at other frequency locations along the unit circle by re-arranging the topology of the resonator. A digital resonator circuit that achieves this is presented

in Figure 3.11. Unlike the previous design of Lu *et al.* [1] this design does not depend on LDI integrators. It consists of two multipliers with coefficients K_c and k_f, two delay elements and two adders. Coefficient K_c is used for coarse tuning and it determines the center frequency of the available oscillation range. Coefficient k_f is responsible for fine tuning the oscillation around this center frequency. To reduce hardware requirements, K_c should be a power-of-two or a sum of a small number of power-of-two terms to avoid the need for a multiplier. From analysis of the block diagram of Figure 3.11, the characteristic equation of this circuit is found to be

$$z^{-2} - (K_c + k_f) z^{-1} + 1 . \tag{3.8}$$

It is very important to note that as long as $| K_c + k_f | \leq 2$, the roots lie on the unit circle and the circuit oscillates. The frequency of oscillation F_o can be obtained directly from the roots of Eq. (3.8), and satisfies the following equation:

$$\cos\left(\frac{2\pi F_o}{F_s}\right) = \frac{K_c + k_f}{2}. \tag{3.9}$$

The amplitude and phase of the sinewave depend on the initial value of register R_1 and R_2. They are obtained by following the procedure outlined in section 3.1.1. The transfer function from the registers to the resonator output (X(z)) is

$$X(z) = \frac{ic_1}{1 + (K_C + k_f) z^{-1} + z^{-2}} - \frac{ic_2 z^{-1}}{1 + (K_C + k_f) z^{-1} + z^{-2}}. \tag{3.10}$$

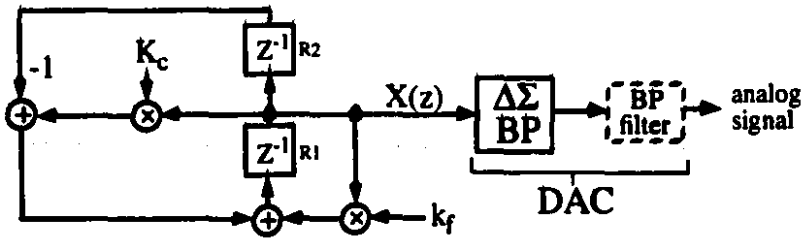


Figure 3.11: Analog sinewave generator based on digital oscillator.

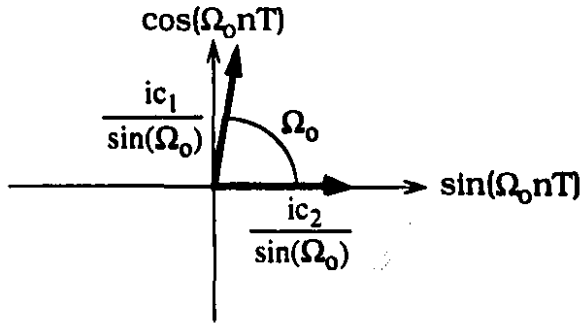


Figure 3.12: Bandpass resonator initial condition phasors.

Substituting $\cos\Omega_o = -\frac{K_C + k_f}{2}$, where Ω_o is the normalized angular frequency, and rearranging terms yield the following equation:

$$X(z) = \left[\frac{1-\cos\Omega_o z^{-1}}{1-2\cos\Omega_o z^{-1}+z^{-2}} + \frac{\cos\Omega_o}{\sin\Omega_o}\frac{\sin\Omega_o z^{-1}}{1-2\cos\Omega_o z^{-1}+z^{-2}}\right] ic_1 + \frac{1}{\sin\Omega_o}\frac{\sin\Omega_o z^{-1}}{1-2\cos\Omega_o z^{-1}+z^{-2}} ic_2 \quad . \tag{3.11}$$

Eq. (3.11) is now in a suitable form to apply the inverse z-transform to find the impulse response. After simplification, a final expression for the impulse response is obtained as

$$x(nT) = ic_1 \frac{1}{\sin\Omega_o}\sin(\Omega_o nT + \Omega_o) + ic_2 \frac{1}{\sin\Omega_o}\sin(\Omega_o nT) \, . \tag{3.12}$$

Again, since the sinusoid triggered by the initial conditions of the registers share the same frequency, they may be represented by phasors. The effects of the initial conditions of the registers are shown in phasor form in Figure 3.12. These phasors may be used to determine the amplitude and phase of oscillation of the network as explained in section 3.1.1.

3.2.2 Delta-Sigma Modulation-Based Bandpass Oscillator

It is desirable to remove the multiplication operation in the network of Figure 3.11 because of its significant latency and large hardware requirements. Therefore, instead of using an N-by-N multiplier for coefficient k_f, a $\Delta\Sigma$ modulator with a signal band located around the center frequency of the digital oscillator, together with a 2:1 multiplexer, is inserted in the

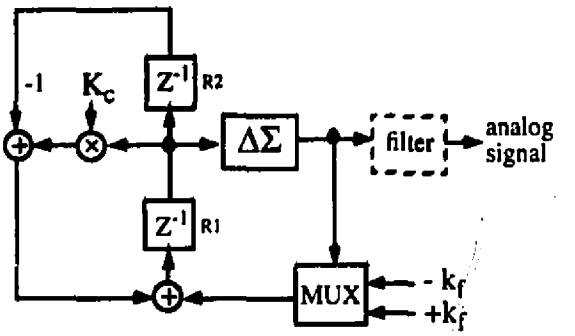


Figure 3.13: Bandpass delta-sigma oscillator.

loop. Since the output of the ΔΣ modulator is 1-bit wide, the operation is reduced to a N-by-1 multiplication, much like that in the lowpass ΔΣ oscillator of Figure 3.5. The binary stream at the output of the ΔΣ modulator is composed of the signal plus out-of-band quantization noise. The analog signal is extracted by bandpass filtering this bit stream. Signals at frequencies higher than the clock frequency can even be obtained from images in the frequency spectrum as the signals are isolated from one another. Figure 3.13 shows the resulting circuit. The reader may notice that this circuit is functionally equivalent to the oscillator of Figure 3.5 for $K_c = 2$ and therefore the circuit of Figure 3.13 is more general. It should be noted that the ΔΣ modulator must have a signal transfer function (STF) with unity gain and zero phase shift, at least in the signal band. Also experiments have shown that a bandpass ΔΣ oscillator making use of a 2^{nd} order bandpass ΔΣ modulator is not stable. We believe the problem arise because the quantization error is distinctly colored for these ΔΣ modulators [18]. When this is the case, the noise power, instead of being spread at a constant level for all frequencies now depends on the input signal. Therefore the STF of the ΔΣ modulator is no longer unity and the characteristic equation is modified, most probably with the poles moving out of the stable region of the z-plane (unit disc).

3.2.3 Multi-tone Bandpass Signal Generation

Just like the LP ΔΣ oscillator, the BP ΔΣ oscillator may be economically adapted for multi-tone generation using time-division multiplexing. Figure 3.14 shows the resulting circuit

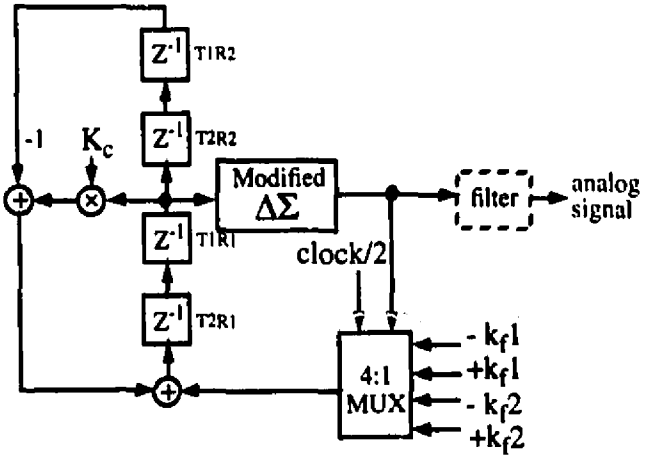


Figure 3.14: Two-tone bandpass delta-sigma oscillator.

when this technique is applied to the circuit of Figure 3.13. The bandpass ΔΣ modulator of Figure 2.12 must also be modified as shown in Figure 3.15. The details of this modification were outlined in Section 3.1.3.

3.2.4 Simulation of the Bandpass Oscillator

The behavior of the circuit of Figure 3.13, including the bandpass ΔΣ modulator of Figure 2.12, was verified using *Simulink*, a floating-point dynamic simulator found in the software called *Matlab*. The simulation conditions are summarized in Table 3.1. Since K_c is set to zero, the center frequency will be a quarter of the clock frequency. From Eqs. (3.9) and (3.12) the sinewave should have a frequency of 255 kHz and an amplitude of 0.5 (signal

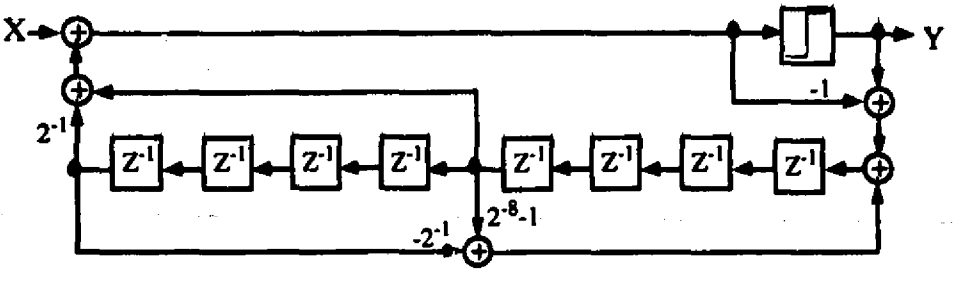
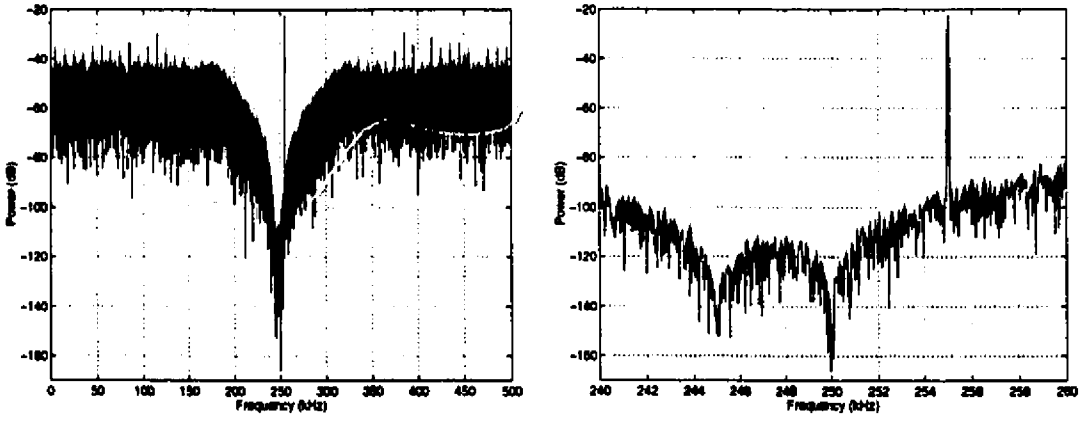


Figure 3.15: Fourth-order time-multiplexed bandpass delta-sigma modulator.

Clock Frequency (F_s)	1 MHz
Coarse tuning feedback coefficient (K_c)	0
Fine tuning feedback coefficient (k_f)	- 0.0628
Initial value of register 1	0.5000
Initial value of register 2	0.0000

Table 3.1: Bandpass signal generator simulation parameters

power -18 dB below the bit stream power). Figure 3.16 (a) and (b) show the simulation results after the output bit stream was processed by a 64K FFT using a raised cosine window respectively in the Nyquist interval and the signal band. Clearly the results in (b) show that the sinewave is present at the desired frequency and that the signal power is close to that predicted. The small difference of approximately 4 dB is due to windowing. Transmission zeros are visible in the signal band located at 245 kHz and 250 kHz. These originate from BP ΔΣ modulation.



(a) (b)

Figure 3.16: Single tone simulated power density spectrum (a) Nyquist interval (b) Signal band.

3.3 Delta-Sigma Modulators Inside Resonating Circuits

In the previous two sections dealing with the lowpass and bandpass delta-sigma oscillators, the same equations developed for the digital resonators were used also for the delta-sigma based oscillators. However simulation and experiments have shown that this model breaks down in some instances, notably when the feedback coefficient k is large. In this section, the concept of delta-sigma oscillation will be generalized and differences with the digital resonator plus DAC model will be examined. The theory developed will then be used to design a $\Delta\Sigma$ oscillator using a third-order digital resonator.

3.3.1 Digital Resonator

The low frequency analog oscillator as well as the bandpass $\Delta\Sigma$ oscillator are based on a tunable digital resonator plus DAC design that may be reduced to the general form shown in Figure 3.17. The resonator is composed of two digital blocks: one denoted B(z) surrounded by unity feedback and the other one denoted as A(z) in cascade with a multiplier with a controlling coefficient k. The digital output of the resonator is fed to a digital-to-analog converter (DAC) composed of a $\Delta\Sigma$ modulator and an analog filter. In general, any type of DAC can be used but a $\Delta\Sigma$ modulator-based DAC will allow for an important circuit simplification. The expressions for A(z) and B(z) for the lowpass and bandpass delta-sigma

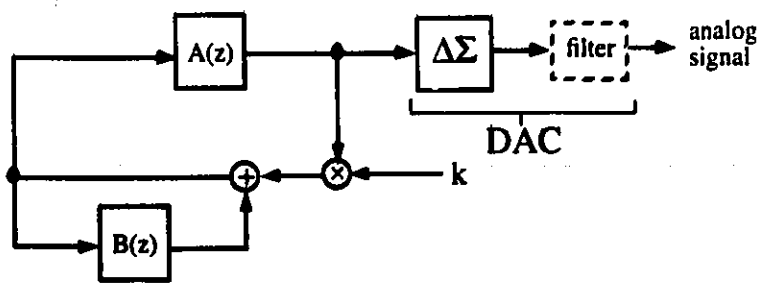


Figure 3.17: General form of a digital resonator feeding an analog-to-digital converter.

lowpass ΔΣ oscillator		bandpass ΔΣ oscillator	
A(z)	$\frac{-z^{-1}}{z^{-1}-1}$	A(z)	z^{-1}
B(z)	z^{-1}	B(z)	$-z^{-2}+K_c z^{-1}$

Table 3.2: Mapping of delta-sigma oscillators to general form

oscillators seen previously are listed in Table 3.2. The characteristic equation of the digital resonator may be obtained by simple analysis of the network and is found to be

$$1 - B(z) - kA(z) . \tag{3.13}$$

The digital network will oscillate if the roots of this characteristic equation lie on the unit circle. The blocks *A(z)* and *B(z)* must therefore be selected such that for any value of *k*, the roots of the characteristic equation remain on the unit circle. The coefficient *k* is thus used to modify the position of the roots on the unit circle and therefore the frequency of oscillation.

3.3.2 Delta-Sigma Attenuator and its Effects on the Resonator

The circuit of Figure 3.17 can be simplified by including the unity signal transfer function (STF) ΔΣ modulator, which is part of the DAC, inside the resonating loop. It should be noted that both the amplitude and the phase are constrained by the unity STF condition. Since the output of the ΔΣ modulator is 1-bit wide, the N-bit by N-bit multiplication which is area expensive and slow, becomes a N-bit by 1-bit multiplication. It can thus be implemented

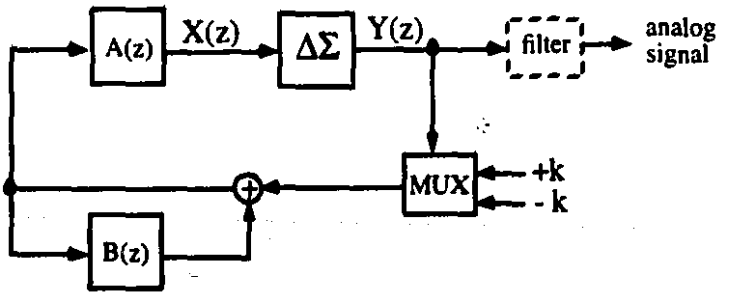


Figure 3.18: General form of a delta-sigma oscillator.

by a 2-to-1 multiplexer as seen in Figure 3.18. The input and output of the ΔΣ modulator have been labeled *X(z)* and *Y(z)* respectively. The association of a ΔΣ modulator and a 2-to-1 multiplexer has been dubbed a *delta-sigma attenuator* [26]. The advantages of the delta-sigma attenuator in the digital resonator, whose combination we now refer to as a delta-sigma oscillator, are many fold. Indeed, such an approach leads to a very area efficient silicon implementation, as no multiplier is required. Moreover, except for a 1-bit DAC, the entire circuit is digital and thus can be implemented in a standard digital CMOS process. It may be tested using well established digital BIST methods and is therefore a valid candidate for the generation of an analog BIST stimulus. The absence of analog components makes it immune to process variations and therefore very predictable. Because the ΔΣ modulator has a unity STF, the operation of the circuit of Figure 3.18 is similar to that of the circuit of Figure 3.17. Equations relating frequency, amplitude and phase to the feedback coefficient (*k*) and the initial values of the registers in the network of Figure 3.17 may be used for the network of Figure 3.18. However the behavior of the delta-sigma oscillator will differ from that of the digital resonator plus DAC counterpart in three aspects. First there is a maximum amplitude of operation due to the saturation of the ΔΣ modulator. Secondly quantization noise is injected in the resonator loop by the quantizer in the ΔΣ modulator. Finally the transient of the ΔΣ modulator will affect the operation of the digital resonator. These effects will be described next and then a mathematical expression linking them together will be presented.

Signal Amplitude

As opposed to the digital resonator, the amplitude of the signal in the ΔΣ resonator is limited. Indeed, the quantization noise added by the quantizer in a ΔΣ modulator will be uncorrelated (white noise) with the input signal as long as the input signal is below the saturation level of the ΔΣ modulator. This saturation level depends on the topology and, to date, no analytic solution has been found so instead extensive simulation must be performed to find it. The initial conditions of the oscillator must therefore be set such that the maximum amplitude is below the saturation level.

Quantization Noise in the Resonator

The ΔΣ modulator will inject quantization noise in the digital resonator. Since the quantization noise power at the output is much larger than the signal power, there is a concern that this noise might affect the operation of the resonator by reducing the dynamic range. Also the quantization noise present in the network may drive the signal at the input of the ΔΣ modulator out of its operating range. However detailed analysis proves that these effects are small when the feedback coefficient (k) is limited to small values. Looking at the circuit of Figure 3.18, we can see that the input of the ΔΣ modulator is related to its output in the following way

$$X(z) = \frac{kA(z)}{1 - B(z)} Y(z) . \tag{3.14}$$

It should be noted that, for simplicity, the expression describing the oscillation triggered by the initial conditions of the registers has not been included in this equation. In addition, we can also relate the output and input of the ΔΣ modulator according to Eq. (2.2), repeated here for convenience, as

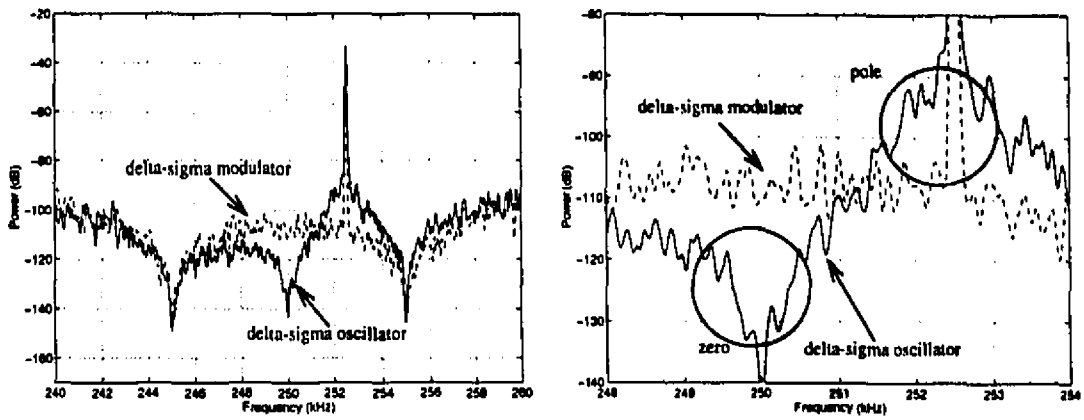
$$Y(z) = STF(z) \cdot X(z) + NTF(z) \cdot Q(z) . \tag{2.2}$$

Combining these relations and eliminating $X(z)$, we obtain the following result:

$$Y(z) = \frac{1 - B(z)}{1 - B(z) - kA(z)} NTF(z) \cdot Q(z) . \tag{3.15}$$

Here we see that the effective noise shaping is equal to the NTF of the ΔΣ modulator multiplied by a zero-pole pair created by the digital resonator blocks. Noise shaping by the ΔΣ modulator is thus modified by the presence of the resonating circuit. In essence, we can view the digital resonator blocks as increasing the order of ΔΣ modulation.

Although the effect of the noise shaping poles positioned on the unit circle is not yet fully understood, they are believed to cause a skirt seen at the base of signals on the power density spectrum. Figure 3.19 illustrates these points by displaying the output of the bandpass ΔΣ oscillator (solid line) and the output of the same bandpass ΔΣ modulator fed by a sinusoid of the same amplitude and frequency (dashed line). The two curves have been



(a) (b)

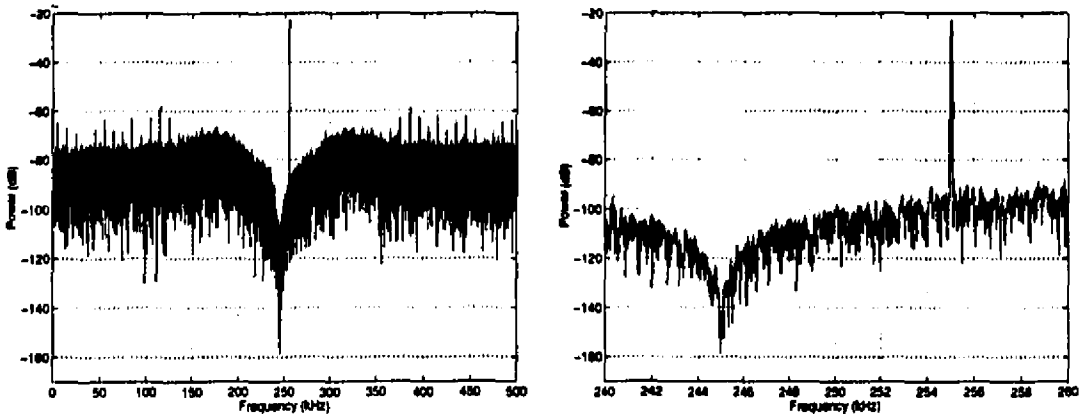
Figure 3.19: Noise shaping difference between delta-sigma modulator and delta-sigma oscillator (a) Signal band (b) Magnification.

smoothed to increase readability. While the ΔΣ modulator output shows only two noise shaping zeros located at 245 and 255 kHz, the ΔΣ oscillator output exhibits an extra noise shaping zero at 250 kHz and a noise shaping pole at 252.5 kHz highlighted by the skirt at the base of the signal. It is interesting to contrast this result with our previous resonator example in Figure 3.16 (b) where noise shaping zeros are seen at 245 and 250 kHz only and that the tone has a very narrow base. This favorable condition occurred because by coincidence the pole formed by the resonator blocks cancelled with a noise shaping zero from the ΔΣ modulator. Therefore a good design strategy is to place the ΔΣ modulator noise shaping zeros at the frequencies of oscillation. Refer to Chapter 2, Section 2.3 for a discussion of zero-pole placement ΔΣ modulator design. This approach is of course limited by the number of noise-shaping zeros in the ΔΣ modulator.

From Eq. (3.15) the effect of quantization noise at each node may also be evaluated. Of particular interest is the amount of quantization noise that appear at the input of the ΔΣ modulator. It may be determined using Eq. (3.14) and (3.15) to be

$$X(z) = k\frac{A(z)}{1 - B(z) - kA(z)}NTF(z) \cdot Q(z) . \tag{3.16}$$

From this expression, it can be seen that as long as the feedback factor k is small, the noise at the input of the ΔΣ modulator will be negligible. Figure 3.20 shows the frequency con-



(a) (b)

Figure 3.20: Power density spectrum of the signal at the input of the delta-sigma modulator (a) Wideband (b) Signal band.

tent of the signal present at the input of the ΔΣ modulator for the bandpass ΔΣ oscillator previously described and whose parameters are listed in Table 3.1. The out of band noise has been considerably reduced by a factor equal to k when compared with the output signal of the resonator (see Figure 3.16). In the passband, the noise power remains approximately at the same level because the reduction due to k is offset by the loss of a zero formed by the resonating circuit ($1-B(z)$) at 250 kHz. The wideband signal to noise ratio (SNR) at this node is close to 20 dB. This oscillator may therefore also be used for digital signal generation. This digital signal has the same dynamic range and frequency tuning capabilities as the analog signal.

Delta-Sigma Modulator Transient

An effect which may significantly reduce the accuracy of the signal amplitude and phase is the manner in which the ΔΣ modulator begins or starts up. On account of the ΔΣ modulator dependency on its initial conditions, its output does not settle immediately into steady-state. As a result, its transient behavior will affect the initial operation of the digital resonator - an effect which can be viewed as altering the initial conditions of the digital resonator. Although the transient will be significant for a few clock cycles, we can lump the time-distributed effect and model it as an impulse applied on the first clock period. While an expression for the magnitude of the impulse that represents the transient may not be ob-

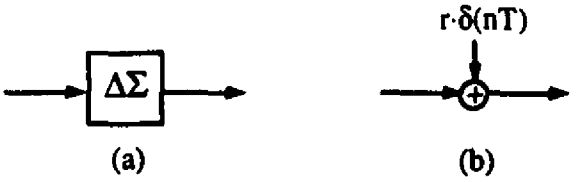


Figure 3.21: (a) Delta-sigma modulator (b) Worst case model of the delta-sigma modulator transient.

tained, as ΔΣ modulators are non-linear, a conservative estimate is to assign it an upper bound of unity. Indeed, the very first bit at the output of the ΔΣ modulator may carry an error of magnitude at most equal to unity. However negative feedback insures that the following bit will make a smaller error and be of opposite polarity. The overall effect should therefore have a magnitude less than unity and an arbitrary phase. The response of the network to the ΔΣ modulator transient may then be obtained using a method similar to the analysis of the effect of initial conditions. Here we replace the ΔΣ modulator block by a two-input summer circuit with one input coming from the original source and the other attached to an impulse signal source δ(nT) as shown in Figure 3.21. Using linear analysis we can therefore assess the effect of the initial conditions on the ΔΣ resonator by determining the transfer function (S(z)) from this impulse signal to the output. Since the actual ΔΣ modulator transient impulse is smaller than unity the effect calculated will yield the maximum amplitude of the sinusoid generated by the transient. However the calculated phase is meaningless as a result of lumping the time-distributed effect of the transient in a single impulse. The actual effect can therefore by written as

$$Y(z) = re^{j\phi} \cdot S(z) \tag{3.17}$$

where the coefficient of $S(z)$ is a complex number of magnitude (r) smaller than unity and unknown phase (ϕ). The value of r may be inferred from experimental results as will be demonstrated latter.

The frequency of this sinusoid being of the same frequency as the signal generated by the initial conditions of the registers, the effect of the ΔΣ modulator transient may be represented by a phasor. But because of the unknown amplitude and phase, the ΔΣ modulator transient phasor is seen as uncertainty added to the prediction of the signal amplitude and phase

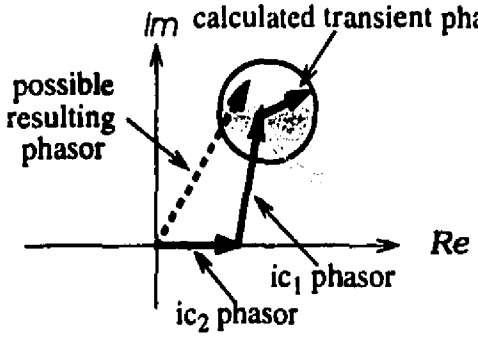


Figure 3.22: The effect of the delta-sigma modulator transient on oscillation.

from the initial condition. Figure 3.22 illustrates the effect of the ΔΣ modulator transient phasor. The initial condition phasors are first summed. Then the transient phasor is added but because the only reliable information about this phasor is a bound on its magnitude, the phasor representing the resulting oscillation will lie anywhere in the shaded area. Evidences supporting this analysis will be provided in Section 4.5.

One must be careful that the ΔΣ modulator transient, coupled with the initial conditions, does not force an amplitude of oscillation larger than the saturation level of the ΔΣ modulator. Since the response of the network to the ΔΣ modulator transient is always proportional to coefficient k, the most efficient way to guarantee accuracy and stability is to restrict this coefficient to small values. This is the main reason why the digital resonator of Figure 3.11 is superior over the LDI resonator of Figure 3.1 for signal generation at high ratios of the clock frequency.

Mathematical Description of Delta-Sigma Oscillators

In summary, ΔΣ oscillators operation is affected by three different stimulus: the initial conditions of the registers in the resonating blocks ($A(z)$ and $B(z)$), the quantization noise introduced by the quantizer in the ΔΣ modulator and the ΔΣ modulator transient. Expressions for each of these have been previously developed. They can be grouped together to yield a complete description of the operation of ΔΣ oscillators. It should be clear however that these equations were obtained using the assumption that the quantization noise approxi-

mates white noise. All the factors affecting the output signal of a resonator of order N may therefore be described in a single equation in the z-domain:

$$Y(z) = \sum_{k=1}^{N} ic_k R_k(z) + re^{j\phi} \cdot S(z) + \frac{1 - B(z)}{1 - B(z) - kA(z)} NTF(z) \cdot Q(z) . \quad (3.18)$$

The first term is the oscillation triggered by the initial conditions (ic_k) of the registers. $R_k(z)$ are the transfer functions from the register to the output. The second term is the oscillation generated because of the $\Delta\Sigma$ modulator transient. The last term is an expression for the effect of quantization noise ($Q(z)$). The first two terms may easily be converted to the time-domain using the inverse z-transform. However the time-domain representation of the last term may not be obtained except by simulation because there is no close form equation for the quantization error. Eq. (3.18) is a general form and may be rewritten for the $\Delta\Sigma$ oscillators presented previously. Using Eq. (3.3) and Table 3.2, the equation governing the behavior of the lowpass $\Delta\Sigma$ oscillator of Figure 3.5 is found to be

$$Y(z) = ic_1 \frac{1 - z^{-1}}{1 - (2-k)z^{-1} + z^{-2}} + ic_2 \frac{z^{-1}}{1 - (2-k)z^{-1} + z^{-2}} + re^{j\phi} \frac{kz^{-1}}{1 - (2-k)z^{-1} + z^{-2}} + \frac{\left(z^{-1} - 1\right)^2}{z^{-2} - kz^{-1} + 1} NTF(z)\, Q(z) \quad (3.19)$$

From Eq. (3.10) and Table 3.2, the equation characterizing the bandpass $\Delta\Sigma$ oscillator of Figure 3.13 is the following:

$$Y(z) = ic_1 \frac{1}{1 + (K_C + k_f)z^{-1} + z^{-2}} - ic_2 \frac{z^{-1}}{1 + (K_C + k_f)z^{-1} + z^{-2}} + re^{j\phi} \frac{1}{1 + (K_C + k_f)z^{-1} + z^{-2}} + \frac{z^{-2} + (Kc + k_f)z^{-1} + 1}{z^{-2} + K_c z^{-1} + 1} NTF(z)\, Q(z) \quad (3.20)$$

These equations can then be used to predict the behavior of the $\Delta\Sigma$ oscillators under realistic conditions.

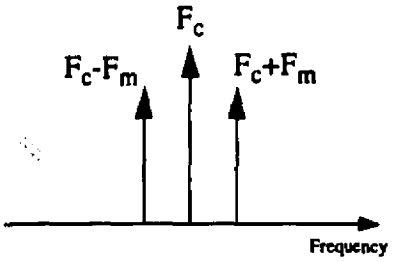


Figure 3.23: Amplitude modulated signal.

3.4 Amplitude Modulation Signal Generation

The principles of section 3.3, that were demonstrated with lowpass and bandpass ΔΣ oscillators, may be used in any oscillating circuit. An interesting case of amplitude modulation is studied in this section. An amplitude modulated (AM) signal consists of three tones as seen in Figure 3.23. A sinewave appears at the carrier frequency (F_c) and two sinewaves at the sum and difference of the carrier frequency and the modulated signal frequency (F_m).

3.4.1 AM Digital Resonator

A third-order digital resonator to implement AM signal generation is shown in Figure 3.24. The characteristic equation of the circuit is

$$z^{-3} + (3-k)z^{-2} + (3-k)z^{-1} + 1 . \qquad (3.21)$$

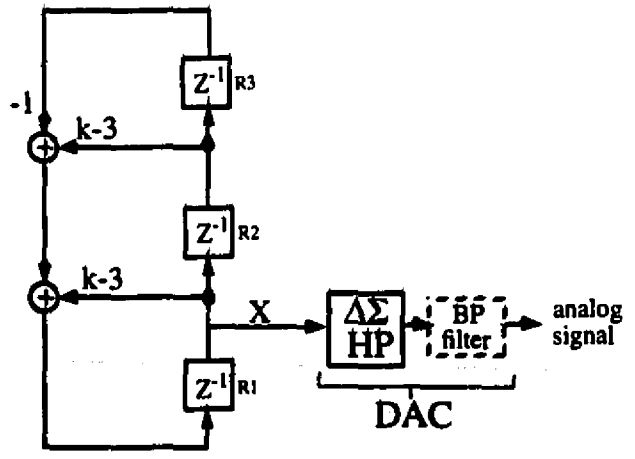


Figure 3.24: Amplitude modulation signal generator based on digital resonator.

For k=0, the three characteristic roots will be located at -1 in the z-plane. Changing the value of the feedback factor k, will force two of the roots to move on the unit circle away from the fixed one. The frequencies of oscillation associated with this pair of complex conjugate roots satisfy this equation:

$$\cos\left(\frac{2\pi F_o}{F_s}\right) = \frac{k}{2} - 1. \tag{3.22}$$

The root at z=-1 will give rise to the carrier sinewave with a frequency equal to half the clock frequency. The other two roots will generate the sidebands where the modulated signal frequency satisfies

$$\cos\left(\frac{2\pi F_m}{F_s}\right) = 1 - \frac{k}{2}. \tag{3.23}$$

The effect of initial conditions of registers on the amplitude and phase may be resolved by analyzing the digital resonator of Figure 3.24 in the same way as for the lowpass and bandpass oscillator. The transfer function from the registers to the output is first determined:

$$Y(z) = \frac{ic_1 + \left[(k-3)z^{-1} - z^{-2}\right]ic_2 - z^{-1}ic_3}{z^{-3} + (3-k)z^{-2} + (3-k)z^{-1} + 1}. \tag{3.24}$$

Using partial fraction expansion,

$$Y(z) = \frac{1}{k}\frac{ic_1\left[(k-1)-z^{-1}\right] + ic_2\left[(k-2)-2z^{-1}\right] + ic_3\left[-1-z^{-1}\right]}{z^{-2} + (2-k)z^{-1} + 1} + \frac{1}{k}\frac{ic_1 + ic_2(2-k) + ic_3}{z^{-1} + 1} \tag{3.25}$$

The inverse z-transform of the first term may be obtained easily. We can substitute $\cos\Omega_o = \frac{k}{2} - 1$ in the second term of Eq. (3.25) and obtain

$$Y(z) = \frac{1}{k}\frac{ic_1 + ic_2(2-k) + ic_3}{z^{-1} + 1} + \frac{1}{k}\frac{ic_1\left[k-1-z^{-1}\right] + ic_2\left[k-2-z^{-1}\right] + ic_3\left[-1-z^{-1}\right]}{z^{-2} - 2\cos\Omega_o z^{-1} + 1} \tag{3.26}$$

The inverse z-transform may then be applied. The effects of the initial conditions of the three registers have been decoupled for clarity:

$$y(nT) = y_1(nT) + y_2(nT) + y_3(nT), \tag{3.27}$$

where

$$y_1(nT) = ic_1\left[\frac{1}{k}\sin(\pi nT) + \frac{k-1}{k\sin\Omega_o}\sin(\Omega_o nT + \Omega_o) - \frac{1}{k\sin\Omega_o}\sin(\Omega_o nT)\right], \tag{3.28}$$

$$y_2(nT) = ic_2\left[\frac{2-k}{k}\sin(\pi nT) + \frac{k-2}{k\sin\Omega_o}\sin(\Omega_o nT + \Omega_o) - \frac{2}{k\sin\Omega_o}\sin(\Omega_o nT)\right], \tag{3.29}$$

$$y_3(nT) = ic_3\left[\frac{1}{k}\sin(\pi nT) - \frac{1}{k\sin\Omega_o}\sin(\Omega_o nT + \Omega_o) - \frac{1}{k\sin\Omega_o}\sin(\Omega_o nT)\right]. \tag{3.30}$$

3.4.2 Delta-Sigma Modulator Based AM Generator

Scaling by the expression *3-k* in the AM generator of Figure 3.24 requires a multiplier which is expensive both in silicon area and in latency. The ΔΣ attenuation concept is therefore applied to this network. One possible way is to rearrange the network of Figure 3.24 by decoupling the multiplicand *3-k* in two paths as shown in Figure 3.25. However the

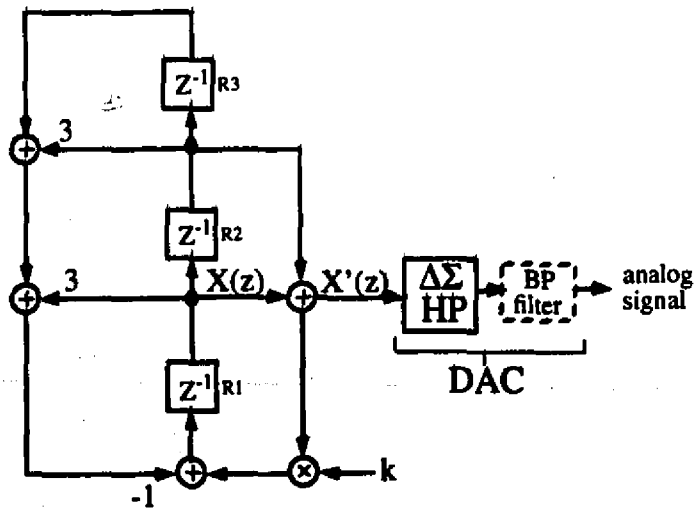


Figure 3.25: Modified amplitude modulation signal generator, first topology.

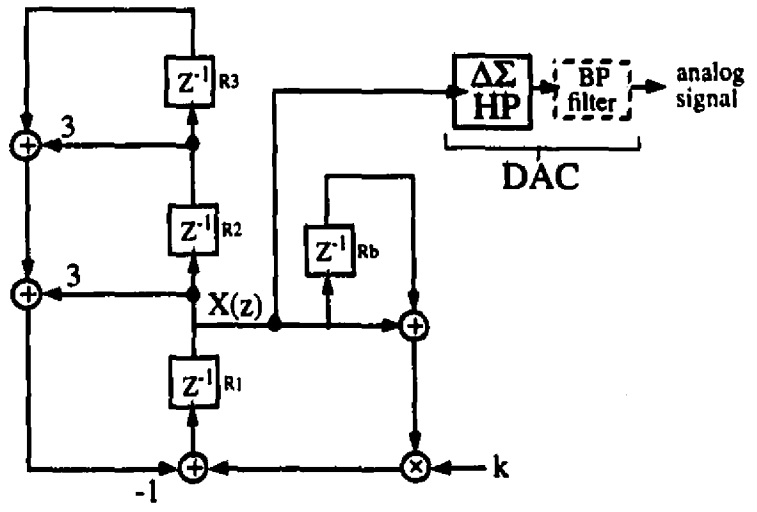


Figure 3.26: Modified amplitude modulation signal generator, second topology.

DAC no longer taps the network at the node labelled X(z) but rather at the X'(z) node related to the first one as:

$$X'(z) = \left(1 + z^{-1}\right)X(z) . \tag{3.31}$$

When replacing $z=e^{j\omega}$ in the $1+z^{-1}$ factor to determine the frequency response, we find that this factor becomes very small around z=1, where our signal band is located. This means that the signal at the input of the ΔΣ modulator will have a much smaller dynamic range than the rest of the circuit. This effect will make the circuit more difficult to stabilize when the ΔΣ modulator is inserted in the loop.

The second topology solves this problem by connecting the input of the ΔΣ modulator to node X(z) and adding a register (R_b) as shown in Figure 3.26. The highpass ΔΣ modulator can then be inserted in the loop as shown in Figure 3.27. Register R_b then becomes a single bit register. However because of the 1-bit by 1-bit adder after the ΔΣ modulator which output is 2-bit wide, the multiplication is now 2-bit by N-bit. It can thus be implemented by a

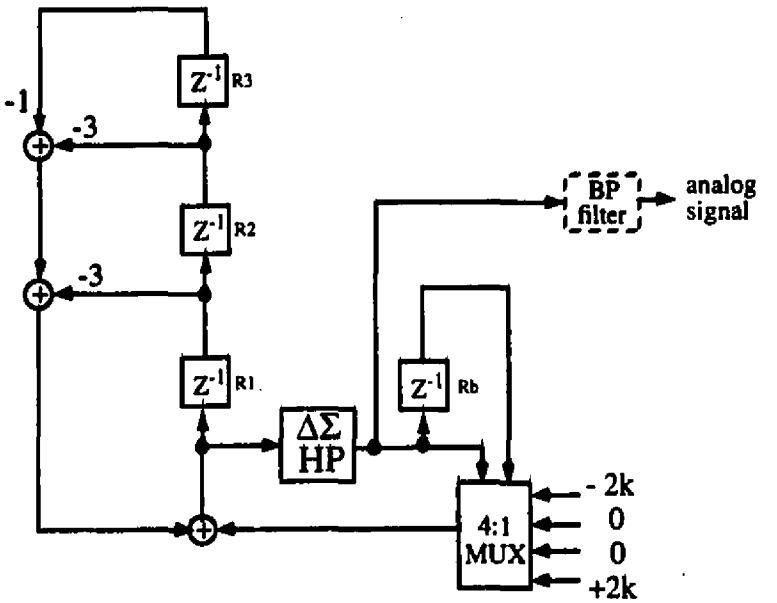


Figure 3.27: AM signal generation using highpass delta-sigma modulation.

4:1 multiplexer. The feedback network taps the resonator at the input of R_1 because the highpass ΔΣ modulator has a $-z^{-1}$ STF. There is however a problem with the initial values of the registers with respect to stability. Indeed, according to Eq. (3.28), (3.29) and (3.30), the initial values of the registers must be smaller than the feedback coefficient (k) for an amplitude of oscillation smaller than unity. While this is not a problem for the digital resonator, the transient behavior of the ΔΣ modulator, discussed in Section 3.3.2, will modify these initial conditions and most likely move the signal at the input of the ΔΣ modulator out of the stability zone. Therefore the initial conditions that will lead to stable oscillation may only be confirmed using simulation.

Clock frequency (F_s)	1 MHz
Feedback coefficient (k)	-2^{-14}
Initial value of register 1	2^{-16}
Initial value of register 2	0
Initial value of register 3	2^{-14}
Initial value of bit register	-1

Table 3.3: AM signal generator simulation parameters

3.4.3 Simulation of the AM Signal Generator

The AM signal generator design was verified using *Simulink*. The initial conditions used for the simulation are shown in Table 3.3. The carrier frequency is 500 kHz and from Eq. (3.23) the modulated signal frequency is 1.24 kHz. Figure 3.28 (a) and (b) show the simulation results after the output bit stream was processed by a 64K fast Fourier transform (FFT) using a raised cosine window respectively up to the clock frequency (0 to 1 MHz) and in the signal band (490 kHz to 510 kHz).

The results in (b) show that the three sinewaves are present at the desired frequencies. A fifth order transmission zero is visible in the signal band at 500kHz. Two zeros are due to

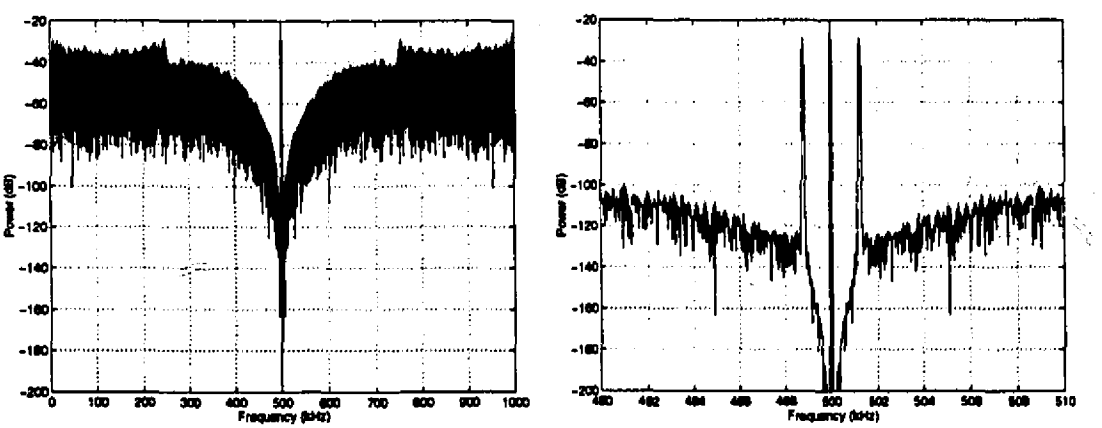


Figure 3.28: Amplitude modulated signal simulated power density spectrum
(a) Wideband (b) Signal band.

the ΔΣ modulator while the other three are generated because of the presence of the resonating circuit around the ΔΣ modulator as explained in Section 3.3.2.

3.5 Summary

A high quality analog sinewave generator based on a digital resonator and ΔΣ modulation techniques was first reviewed. Because of the frequency limitation of the circuit, it is beneficial to replace the lowpass ΔΣ modulator by a bandpass ΔΣ modulator. However, the circuit must be modified such that k is small for the generation of the sinewaves in the frequency band. Then the concept was extended to any digital resonating network and the non-ideal behaviors of this class of analog resonator, called ΔΣ oscillators, were quantified. Finally, a third order ΔΣ oscillator was designed to further illustrate these concepts.

Chapter 4 : Experimental Results

4.1 Prototyping Environment

A field programmable gate array (FPGA) based development system [27] has been devised to allow rapid prototyping of the delta-sigma oscillators. The digital signal processing flow graph is first translated to VHDL, a hardware description language and, after validation using a VHDL simulator, the design is compiled for the Xilinx XC4000 FPGA family using Synopsys synthesis tools. The resulting netlist is then partitioned, placed and routed for an XC4010 by the Xilinx software. As a result, a configuration data stream is generated that will be loaded on to the FPGA. Full control of the operational parameters such as clock frequency, coefficients and initial conditions of registers is provided by digital VXI modules. The output of the FPGA is fed to a 1-bit DAC, shown in Figure 4.1, and the analog output is examined on an HP 3588A spectrum analyzer or filtered and then viewed on an oscilloscope. Measurement accuracy is limited by these instruments. A Sun workstation is used for design processing and to provide high-level control of the digital modules. It is also used to retrieve experimental results from the analog test equipment and provide post-process-

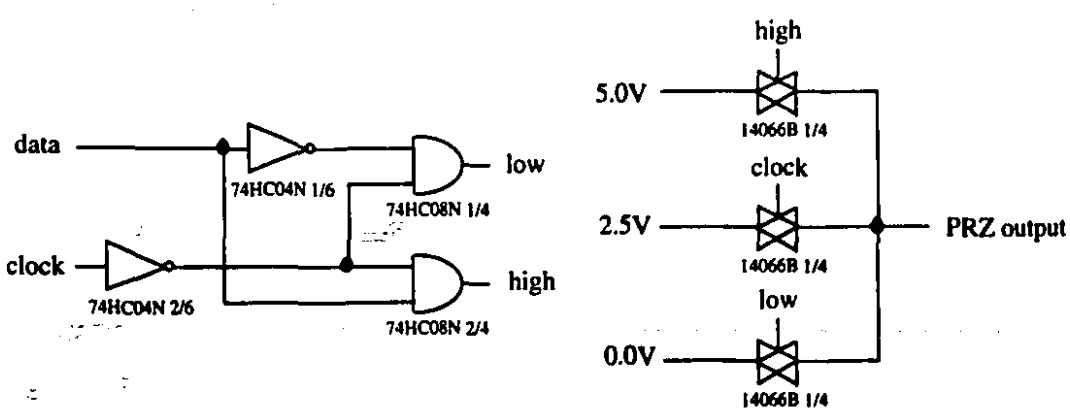


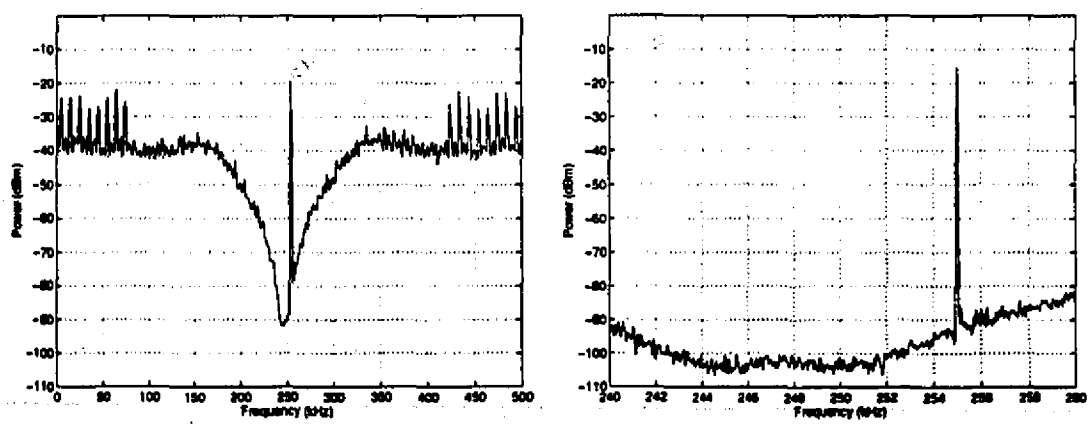
Figure 4.1: One-bit polar-return-to-zero digital-to-analog converter.

ing. A thorough description of the prototyping environment and the design process is given in Appendix B.

4.2 Single Tone Bandpass Delta-Sigma Oscillator

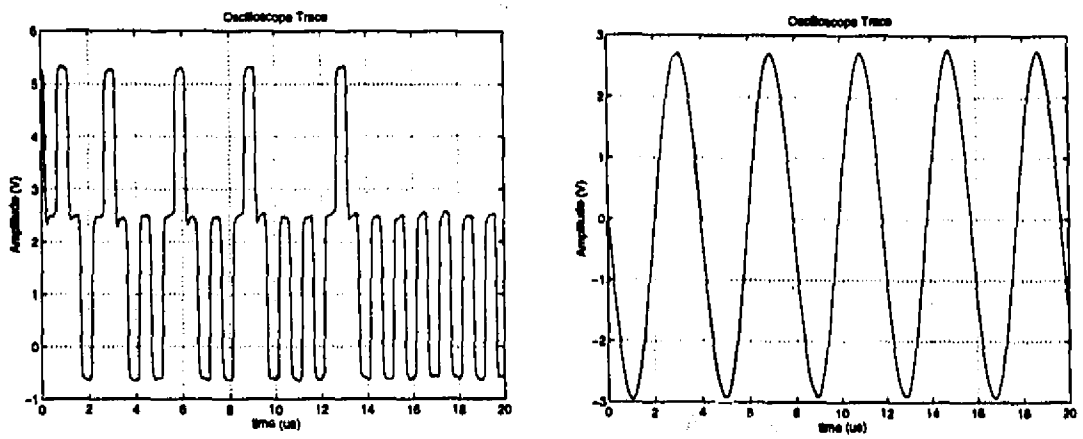
The digital portion of the oscillator of Figure 3.13, including the $\Delta\Sigma$ modulator of Figure 2.12, was described using VHDL. This code has been reproduced in Appendix C. The resonator bus width was set to 16 bits, while the delta-sigma modulator had 27-bit internal buses. The VHDL description was compiled with *Synopsys Design Compiler*. The netlist was then routed with *Xilinx XACT tools*. The circuit uses 48% of the function generator resources of a Xilinx XC4010 and 21% of the flip-flops. The output is fed to a 1-bit DAC that converts the digital sequence to a Polar-Return-to-Zero coded signal.

Under the same conditions as for the simulations, the output of the circuit was examined on the spectrum analyzer. Figure 4.2 (a) shows the power density spectrum in the Nyquist interval (0 to 500 kHz) while (b) displays the signal band (240 kHz to 260 kHz) in more detail. In comparison with the previous simulation results of Figure 3.16, the experimental results seem to agree. As expected the signal generated is located at 255 kHz and the quantization noise is pushed out of the region surrounding the tone. The measured signal-to-



(a) (b)

Figure 4.2: Single tone measured power density spectrum (a) Nyquist interval (b) Signal band.



(a) (b)

Figure 4.3: (a) Signal in the time domain. (b) Bandpass filtered signal in the time domain.

noise ratio (SNR) is 50.9 dB for an oversampling ratio (OSR) of 25 (20 kHz bandwidth). Transmission zeros similar to those seen in simulation are visible at 245 kHz and 250 kHz. The oscillator was left running for over 12 hours with no change in the output signal, indicating good stability.

The PDM signal was then filtered by a 4^{th} order bandpass filter. The result is shown in Figure 4.3 (b). The frequency of the sinusoid was measured to be exactly 255 kHz. To illustrate the relationship between the generated sinusoid and the clock frequency, the bit stream output is shown in (a). Approximately 4 pulses of the sequence make up a single period of the sinusoidal signal. Thus illustrating the capability of this circuit to generate sinusoids at frequencies very near the clock frequency.

4.3 Images of the Baseband Signal

This signal generator could be used for high frequency sinusoidal generation. However it is clearly not possible to operate the previously presented oscillator at 4 times the carrier frequency of a wireless system for example, since that would require a clock frequency in the gigahertz range. However signal generation is not limited to the Nyquist frequency. In-

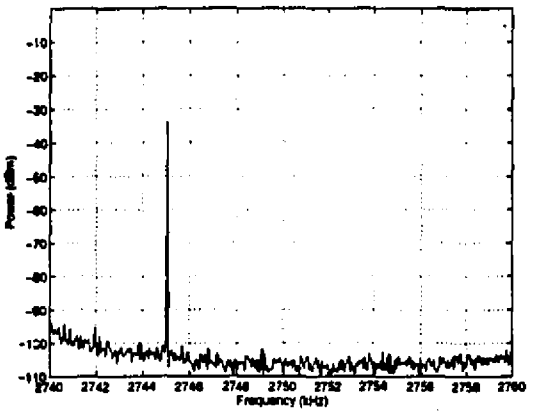


Figure 4.4: 6th image measured power density spectrum.

deed, because the output is sampled data, images of the baseband will be created that can be used for signal generation. The signal power of these images depends on the type of the 1-bit DAC. For a zero-order hold function, the signal power drops like the familiar sinc function. Similarly the quantization noise from the $\Delta\Sigma$ modulator will also be reduced by the same level. However the dynamic range might degrade if the electronic noise floor becomes larger than the quantization noise. Figure 4.4 shows the sixth image of the baseband tone. The signal power has been reduced by 19 dBm when compared with the first image (Figure 4.2 (b)). The quantization noise is also reduced by a similar factor but the test bench does no show this effect. Therefore the SNR decreases to 43.8 dB over the signal band. Figure 4.5 shows the measured image power with respect to frequency for a 1 MHz clock

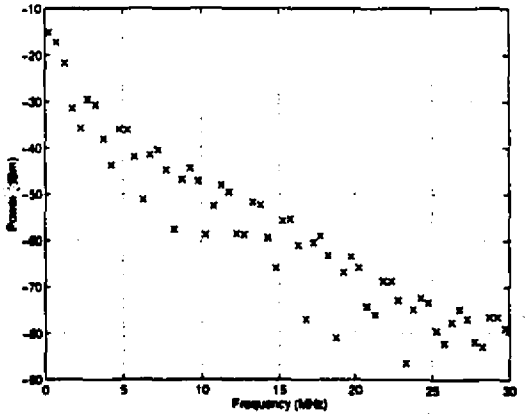


Figure 4.5: Measured signal power of images.

frequency, polar return-to-zero (PRZ) coding and a 5 V power supply. It should be noted that sinusoids almost 30 times the clock frequency have signal power in excess of -80 dBm.

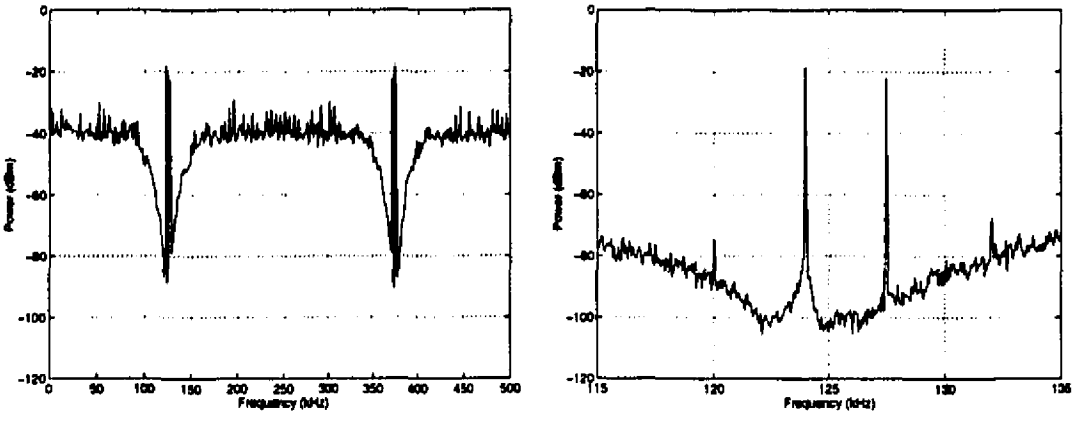
4.4 Multi-Tone Bandpass Delta-Sigma Oscillator

A two-tone high-frequency sinusoidal generator as shown in Figure 3.14 together with the time multiplexed $\Delta\Sigma$ modulator of Figure 3.15 was implemented on a single XC4010 FPGA. The circuit uses 74% of the function generator resources of a Xilinx XC4010 and 42% of the flip-flops. The simulation parameters are shown in Table 4.1.

Figure 4.6 (a) shows the spectral density up to half the 1 MHz system clock. Because time-multiplexing is used, the effective clock is reduced to 500 kHz and two images are visible.

Clock Frequency (F_s)	1 MHz
Coarse tuning feedback coefficient (K_c)	0
Tone 1 Fine tuning feedback coef. (k_{f1})	- 0.0628
Tone 1 initial value of register 1	0.5000
Tone 1 initial value of register 2	0.0000
Tone 2 Fine tuning feedback coef. (k_{f2})	0.0314
Tone 2 initial value of register 1	0.5000
Tone 2 initial value of register 2	0.0000

Table 4.1: Two-tone bandpass delta-sigma oscillator simulation parameters.



(a) (b)

Figure 4.6: Two-tone measured power density spectrum (a) Nyquist interval (b) Signal band.

Figure 4.6 (b) shows the first signal band. The signals appear at 124 kHz and 127.5 kHz. The spurious tones at 120 kHz and 132 kHz were not present in the simulation and are probably caused by the non-ideal behavior of the 1-bit DAC.

4.5 Effect of Transient on the Delta-Sigma Oscillator

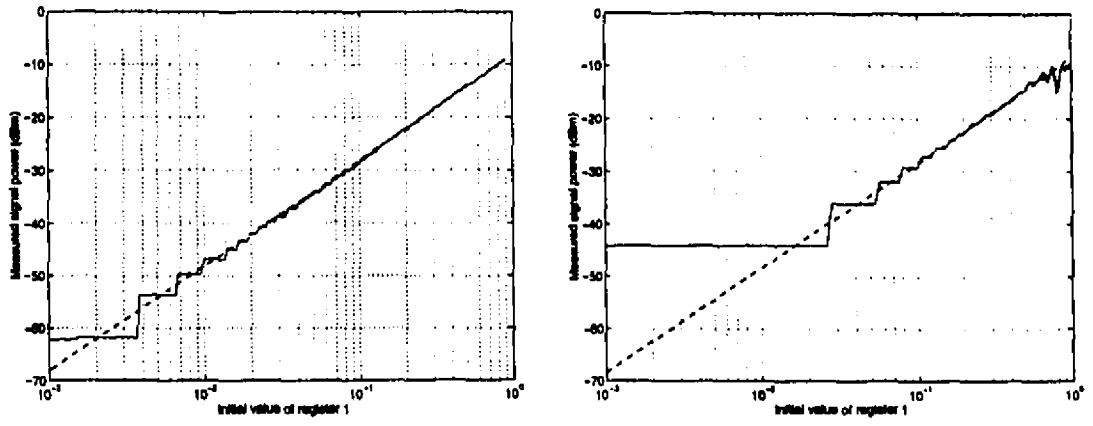
To assess the impact of the ΔΣ modulator transient on the amplitude of oscillation, the FPGA implementation of the lowpass ΔΣ oscillator was operated with different initial values for register 1. In this network, the oscillation amplitude is predicted by the first three terms of Eq. (3.19) reproduced here without the last one pertaining to noise shaping:

$$Y(z) = ic_1 \frac{1-z^{-1}}{1-(2-k)z^{-1}+z^{-2}} + ic_2 \frac{z^{-1}}{1-(2-k)z^{-1}+z^{-2}} + \frac{re^{j\omega_k} z^{-1}}{1-(2-k)z^{-1}+z^{-2}} \quad (3.19)$$

The time-domain equivalent to this equation yields more insight:

$$y(nT) = ic_1 \frac{\sqrt{2(1-\cos\Omega_o)}}{\sin\Omega_o} \sin\left(\Omega_o nT + \frac{\pi+\Omega_o}{2}\right) + ic_2 \frac{1}{\sin\Omega_o} \sin(\Omega_o nT) \quad . \quad (4.1)$$

$$+ r\frac{k}{\sin\Omega_o} \sin(\Omega_o nT + \omega)$$



(a) (b)

Figure 4.7: Measured signal power versus initial value of register 1

(a) k=9.9x10^{-6} (b) k=6.3x10^{-4}.

Figure 4.7 (a) shows a plot of the signal power versus initial value of register 1 for a a signal frequency set to 500 Hz (k=9.9x10^{-6}). The dotted line is the theoretical prediction without the effect of the ΔΣ modulator transient (Eq. (3.7)). An amplitude of unity corresponds to -8 dBm. With ic_2 equal to zero and very low values of ic_1, the oscillation is due almost entirely to the ΔΣ modulator transient and Eq. (4.1) is reduced to

$$y(nT) = A\sin(\Omega_o nT + \omega) = r\frac{k}{\sin\Omega_o}\sin(\Omega_o nT + \omega) \tag{4.2}$$

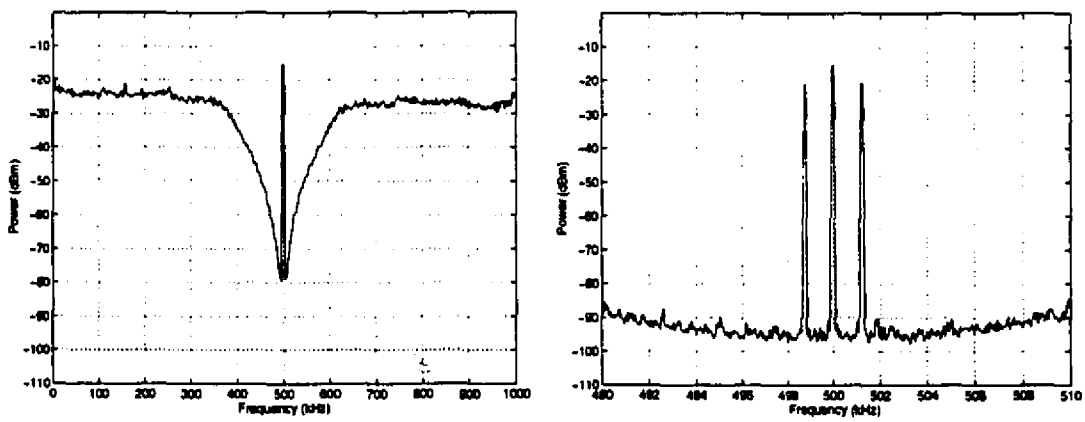
where A is the measured amplitude. The ΔΣ modulator transient error magnitude (r) may then be estimated by simplifying and rearranging Eq. (4.3) to

$$r = \frac{A \cdot \sin\Omega_o}{k}. \tag{4.3}$$

For the experiment of Figure 4.7 (a), A is measured to be -62 dBm (0.002) and the value of r is calculated to be 0.63. Figure 4.7 (b) shows the result of the same experiment for a signal frequency of 4 kHz. The oscillation for small values of ic_1 is larger because the oscillation due to the ΔΣ modulator transient is proportional to k. A has increased to -44 dBm (0.016) and the magnitude of the ΔΣ modulator transient error is evaluated again to 0.63.

4.6 Amplitude Modulated Signal Generator

The AM signal generator of Figure 3.27 was implemented on an FPGA using 19-bit buses inside the HP ΔΣ modulator and 16-bits buses everywhere else. The circuit was operated using the same parameters as in the simulations. Figure 4.8 (a) shows the power density spectrum up to the clock frequency. The noise shaping effect of the HP ΔΣ modulator is clearly seen, with a signal band centered at 500 kHz. Figure 4.8 (b) displays this signal band (490 kHz to 510 kHz) in more details. The three tones appear at the same frequencies as in the simulation. The fifth order noise shaping zero at 500 kHz is not visible because of the limited accuracy of the test bench.



(a) (b)

Figure 4.8: AM signal measured power density spectrum (a) Wideband (b) Signal band.

Chapter 5 : Application

Wireless communication is a rapidly expanding field and considerable research effort is underway. Electronic system manufacturers are actively trying to incorporate all of the RF, IF and baseband functions on the fewest possible ICs so as to reduce system size and cost, and to improve overall system performance. Unfortunately, without a coherent test strategy, any cost reduction gained by miniaturization will be offset by increased testing costs. In this chapter we propose a built-in self-test scheme for bandpass (BP) type systems such as those used in wireless communication devices [28]. The scheme is centered around the bandpass delta-sigma ($\Delta\Sigma$) oscillator presented in Chapter 3. Measures meaningful to the mixed-signal test engineer such as signal-to-noise ratio, frequency response and intermodulation distortion are obtainable with this method.

5.1 Introduction

Over the past decade we have observed the emergence of a class of integrated circuits (IC) including both digital and analog function on a single die. These mixed-signals ICs have been steadily replacing boards and chipsets, performing various low-frequency analog functions. Although there are many different types of mixed-signal ICs, a typical device is one that contains an analog-to-digital converter (ADC), a digital-to-analog converter (DAC) and a digital processing unit (DPU) as illustrated in Figure 5.1. A similar trend is beginning to appear in wireless communication systems. A larger number of mobile communication device functions are being integrated onto a single die to reduce the overall chip count. Unfortunately, without consideration at the design stage of the testing difficulties, the cost savings of system miniaturization will be written off by increased test costs.

The testing issue of low-frequency mixed-signals IC has been tackled by the recent introduction of a mixed analog-digital built-in self-test (MADBIST) scheme [2]. It features the

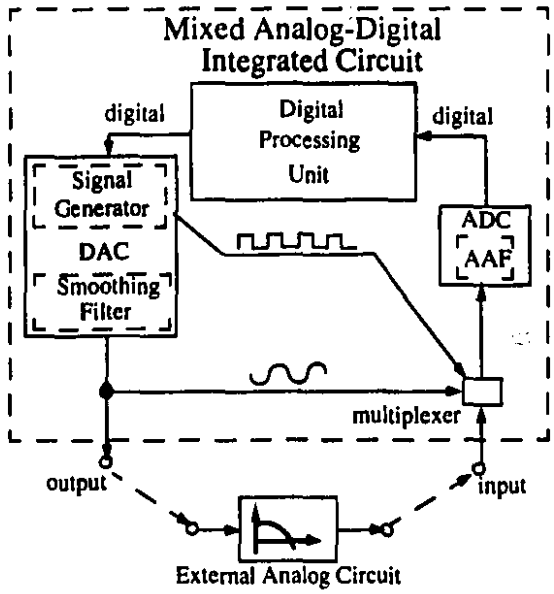


Figure 5.1: A mixed-signals IC adapted to MADBIST.

LP ΔΣ oscillator of Chapter 3 as the stimulus source. The extraction of circuit parameters is performed efficiently by digital filtering, making use of available computing power on-chip or in the system. MADBIST may now be extended to bandpass type devices such as those found in wireless communication systems with the help of the bandpass ΔΣ oscillator presented in Chapter 3. This is a unique opportunity, as wireless communication systems are still being designed, to incorporate the test strategy at the design stage and thus significantly reduce the testing cost and even allow testing in the field.

5.2 Low-Frequency MADBIST

An obvious solution to the testing problem of mixed-signal ICs is to connect the output of the mixed-signals IC in Figure 5.1 to the input so that the loop is closed and then use the DPU to test both the ADC and the DAC at the same time [29]. However this approach is not sound as an error, for example gain tracking, in one of the blocks could be masked by a corresponding error in the other block. Therefore in MADBIST the ADC is first verified alone before closing the loop. To implement the BIST capability in the mixed-signals IC of Figure 5.1, the lowpass ΔΣ oscillator and the multiplexer have been added. A diagram of

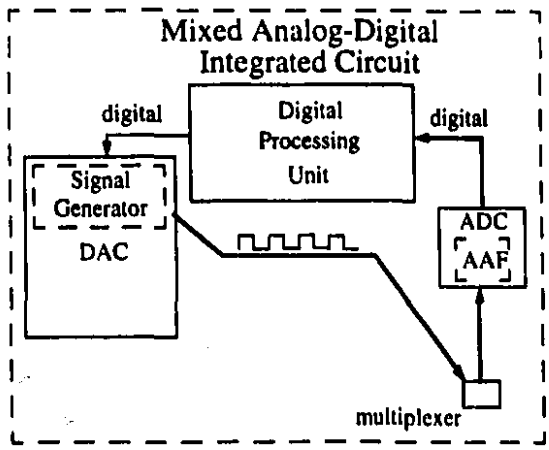


Figure 5.2: Analog-to-digital converter test setup.

the test setup is shown in Figure 5.2. A pulse density modulated (PDM) signal from a low-pass $\Delta\Sigma$ oscillator is applied to the input of the ADC. In the PDM bit stream is a high-quality low-frequency sinewave and high-frequency quantization noise. This noise is removed by the anti-aliasing filter (AAF) leaving a pure sinusoid to excite the remaining circuitry. The digital output of the ADC is then analyzed by the DPU to separate the signal and the noise using a digital filter as shown in Figure 5.3. A Fast Fourier Transform (FFT) could also be used to yield more accurate results albeit at the price of more computing power. Analog measures such as signal-to-noise ratio (SNR), gain tracking and frequency response may then be obtained. Since the lowpass $\Delta\Sigma$ oscillator may further be adapted to provide multi-tone signals [24], intermodulation distortion measurements [30] and rapid frequency response tests are also possible.

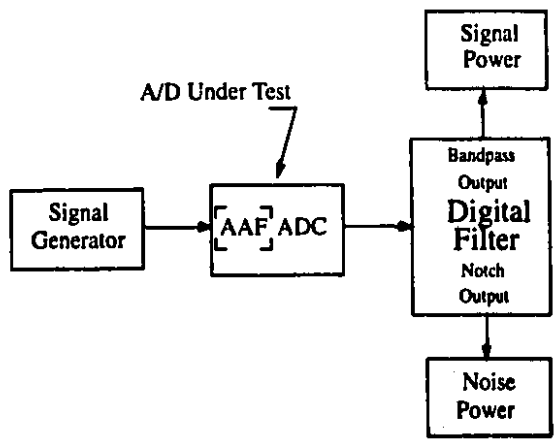


Figure 5.3: Output signal processing in MADBIST.

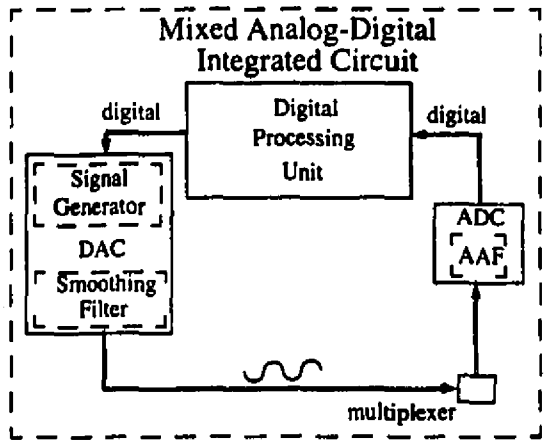


Figure 5.4: Digital-to-analog converter test setup.

After the ADC has passed all the tests, the DAC may now be verified by placing it in the loop as shown in Figure 5.4. A digital signal, generated by the ΔΣ oscillator or the DPU, excites the DAC. The analog output is sent to the ADC which converts it back into digital form. A digital filter or FFT implemented by the DPU may then separate the noise from the signal. After the DAC is characterized, other analog circuits may be tested by placing them between the DAC and the ADC. We intend to demonstrate how the MADBIST scheme can be extended to include bandpass devices such as those found in wireless communication systems.

5.3 High-Frequency MADBIST

A block diagram of a wireless communication system is shown in Figure 5.5. On the left side are the RF and IF stages mostly composed of analog circuits. The baseband section is

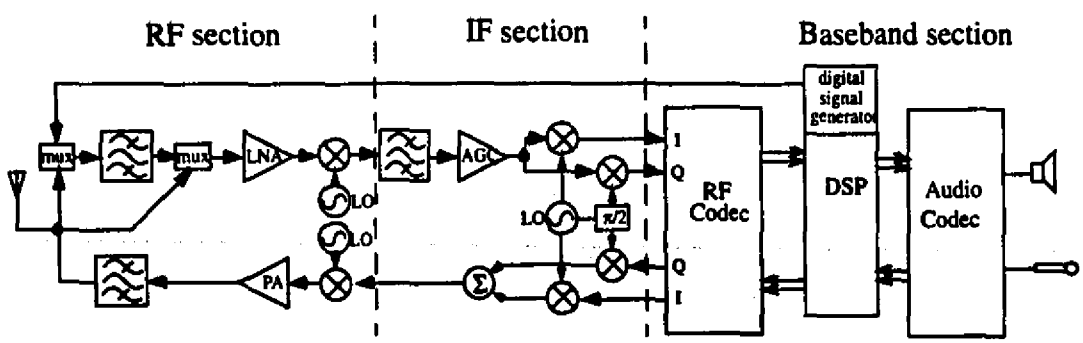


Figure 5.5: Wireless communication system.

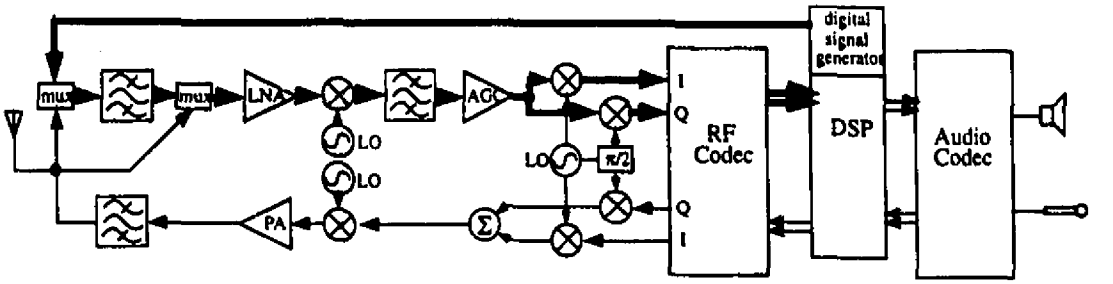


Figure 5.6: Wireless receiver test setup.

made of a voice-band codec, a digital signal processor (DSP) and an RF codec which modulate/demodulates low-frequency signals coming from the analog front-end. The MAD-BIST scheme may be applied to the audio codec and one can now also adapt it to the remaining analog circuitry, the RF and IF stages and the RF codec, using the bandpass ΔΣ oscillator. The only change in the circuit that is required to make it self-testable is the addition of two multiplexers at the input of the receiver and after the RF bandpass filter as illustrated in Figure 5.5. It should be noted that although a single IF receiver is illustrated, the test scheme may be adapted to other down-conversion methods.

The diagram for the receiver test setup is shown in Figure 5.6. The test signal path is illustrated using bold lines. The BP ΔΣ oscillator feeds the high-frequency analog signal imbedded in a 1-bit stream to the RF front end. The out-of-band noise is removed by the RF and IF filters. The RF codec demodulates the down-converted signal and the DSP unit then analyzes the output and decides if the device meets the necessary specifications. The stimulus may also be applied at the IF stage if the precise location of the defect is of interest.

After validation of the receiving side, the transmitter may now be tested. A diagram of the test setup is shown in Figure 5.7. A digital signal, either from the BP ΔΣ oscillator or from the DSP, excites the transmitter which generates an RF signal. This RF signal is then rerouted and applied to the receiver input via the multiplexer. The receiver down-converts and demodulates the output to a voice-band digital signal. The DSP may then analyze the result and either accept or reject the circuit. For this scheme to work, the transmitter and the receiver must use the same carrier frequency. Unfortunately this is usually not the case. Therefore the receiver local oscillator should be set to the transmitter carrier frequency or

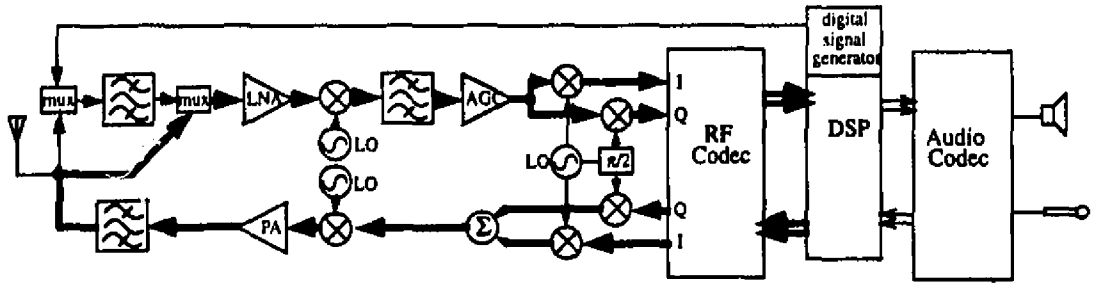


Figure 5.7: Wireless transmitter test setup.

the receiver mixer should use the transmitter local oscillator for the duration of the test. The bandpass filter at the input of the receiver must be bypassed because it is tuned to a different frequency than the transmitter carrier frequency.

The digital signal generator is currently capable of generating only sinusoids. While this stimulus is sufficient to test the RF and IF analog front-end, it leaves a portion of the RF codec circuitry unverified, as the RF codec implements various modulation/demodulation schemes. The circuitry involved in the decoding simply sees a steady signal and no time-varying information. Research is currently underway addressing these needs by attempting to modify the BP $\Delta\Sigma$ oscillator for the generation of analog and digital modulated signals, thus providing a complete stimulus toolset [31].

5.4 Extraction of Test Results

Two methods may be used to separate the signal from the noise. The industry standard is the Fast Fourier Transform (FFT) but it requires a large amount of resources and is therefore not suitable for on-chip implementation. MADBIST proposes an alternative method based on filters.

5.4.1 Extraction using Fast Fourier Transform

Separation of signal and noise using an M-point FFT yields very accurate results. However, it necessitate M memory cells to store the samples and requires on the order of $M \cdot \log_2 M$ operations. Since measurement accuracy increases with the number of samples, in many systems the overhead to implement the FFT extraction procedure may be prohibitive.

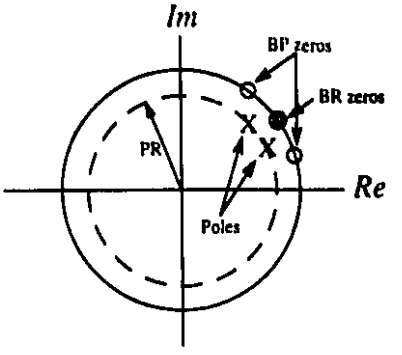


Figure 5.8: Position of the poles and zeros of fourth order bandpass and band-reject filters.

5.4.2 Extraction using Filters

To reduce hardware requirements, a circuit implementing both the bandpass function and the band-reject function was used in [2]. It is based on a time-recursive implementation of an arbitrary transform [32]. The signal power is computed by squaring and adding the output of the bandpass filter while the noise power is obtained by processing the band-reject output in a similar manner. Using this scheme, only 4 memory cells are required for sequences of any size. While the second order filter in [2] is very efficient and may be easily extended to multi-tone extraction, it suffers from an extremely narrow notch function. A very small error in the clock frequency of the signal generator will cause the stimulus to shift frequency and pass through to the band-reject output. The problem is exacerbated by down-conversion because the error is increased relative to the clock frequency. However a more tolerant extraction circuit may be obtained using fourth-order filters. The positions in the z-plane of the poles and zeros of the filters used in the experiments of the following section are shown in Figure 5.8. The bandpass and band-reject functions share the same poles to reduce complexity. The pole radius (*PR*) parameter sets the accuracy as well as the convergence rate. The choice of second-order filters, fourth-order filters or FFT to implement parameter extraction depends on the availability of computing resources in the system and the required accuracy. Two of these methods will be compared in our experiments.

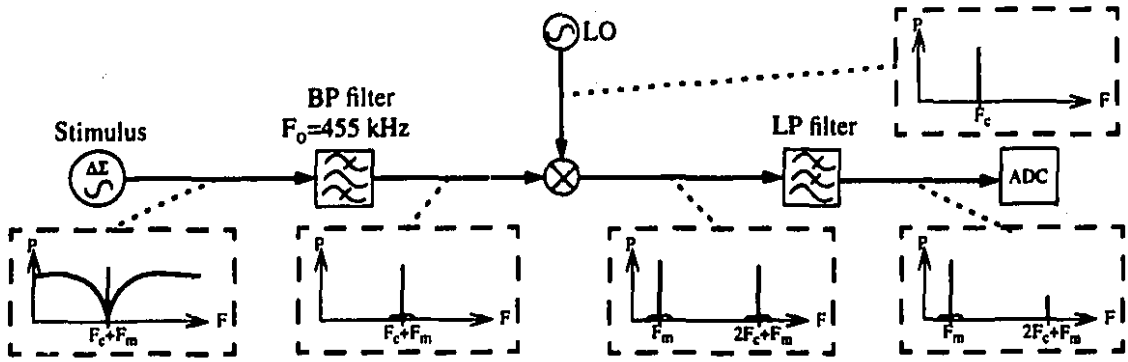


Figure 5.9: Down conversion experimental setup.

5.5 Experiments

To illustrate the feasibility of this scheme, the experimental setup of Figure 5.9 was built. A BP ΔΣ oscillator is implemented using an XC4010 FPGA. The stimulus is passed through a BP filter to simulate the frequency selectivity of wireless receivers. The filtered signal is then down-converted by a balance modulator with the second input being a pure sinusoid. The output is filtered to remove the second modulation product and then sampled at 40 kHz by an HP 1430A ADC. The carrier frequency (F_c) was selected to be 448 kHz while the modulated signal frequency (F_m) was set to 7 kHz. The ΔΣ oscillator thus generated a 455 kHz sinewave while being clocked at 1818 kHz. An FFT of the baseband sampled signal is shown in Figure 5.10 (a). A raised cosine window was applied to the samples

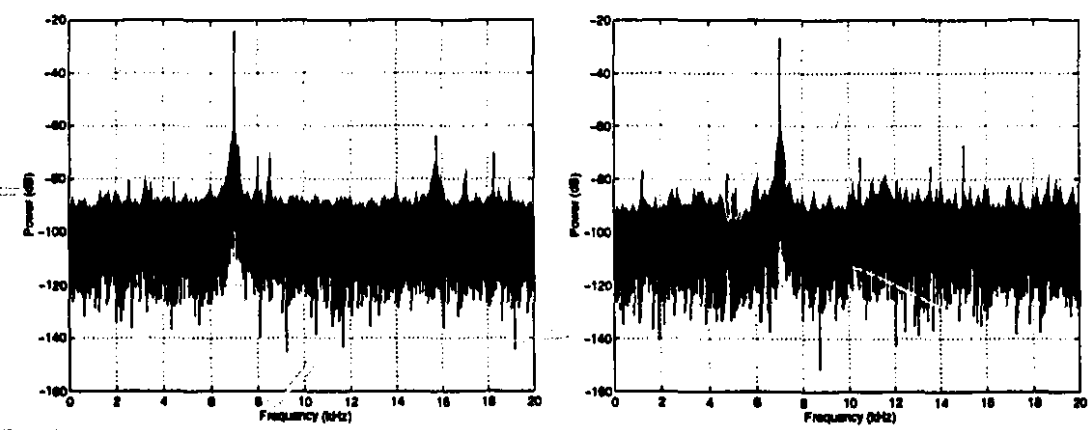


Figure 5.10: Power density spectrum of sampled down-converted signal
(a) Delta-sigma oscillator (b) HP 3314A signal source.

to eliminate leakage effects caused by non-coherent sampling. If the filtering scheme is used, then the application of a window on the samples prior to signal extraction is not necessary. Returning to the results displayed in Figure 5.10 (a), we see that a tone is present at 7 kHz as expected. For comparison, the experiment was repeated with a sinewave from an HP 3314A signal generator. The spectral content of the sampled baseband signal is shown in Figure 5.10 (b). The results are very similar to those in (a) except for some spurious tones that are a result of our noisy experimental environment.

Since the validity of the BP ΔΣ oscillator as a test stimulus for the wireless receiver has been established, we may now perform various tests to obtain meaningful analog metrics.

5.5.1 Signal-to-Noise Ratio

A very important measure of the quality of an analog system is the SNR. It is obtained by applying a sinewave of fixed amplitude while varying the power at the input of the circuit-under-test. An FFT is applied on the output and a ratio of the signal power over the total noise power is computed. This experiment was performed with both the BP ΔΣ oscillator and a sinewave from an HP 3314A signal generator. The results are shown in Figure 5.11. The two sources yield similar measures especially at low signal levels. At the high end, the discrepancy between the two curves illustrate the limitations of filter method for separation of signal and noise. Indeed in this experiment a significant amount of noise is located at

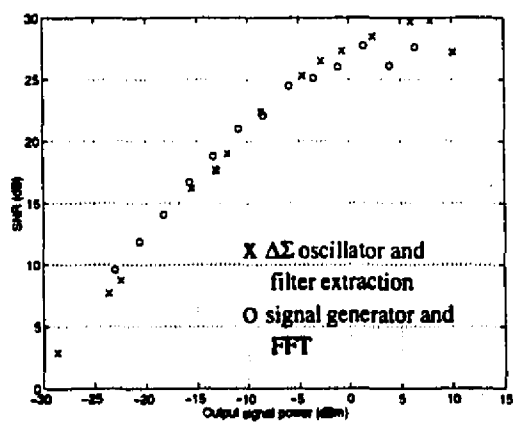


Figure 5.11: SNR of the sampled down-converted signal with delta-sigma oscillator and HP 3314A signal generator.

frequencies close to the signal frequency. Since a brick-wall filter is not possible, the band-reject filter, while eliminating the signal, will attenuate this noise, thus causing the observed increase in the SNR measure.

5.5.2 Frequency Response

Another characteristic of an analog circuit is its frequency response. This measure is realized by sweeping a sinusoid of fixed amplitude over a range of frequencies and measuring the signal power at the output. Again the experiment was done with the BP $\Delta\Sigma$ oscillator as well as a sinewave from a signal generator for reference. On accounts that the amplitude of the signal generated by the BP $\Delta\Sigma$ oscillator is not exactly governed by Eq. (3.12), it is difficult to find the initial values for the two registers that will maintain a constant amplitude for a wide range of test frequencies; variation in signal amplitude of 5 dB over a frequency range of 20 kHz has been observed. So to avoid a situation where the input signal is not what is expected, we first run a calibration cycle where we determine the signal amplitude received by the extraction circuitry and use these values to correct any future measurement. Fortunately, this entire process involves deterministic digital circuits and therefore can be performed off-chip using a digital simulator that is capable of accounting for finite-length register effects. An example of the calibration correction factors used in this experiment are shown in Figure 5.12, The frequency response of the receiver is shown in Figure 5.13 for the two different types of excitation. Clearly, they correspond quite closely. Also seen is the frequency response behavior of the BP filter used in this experiment. This plot was in-

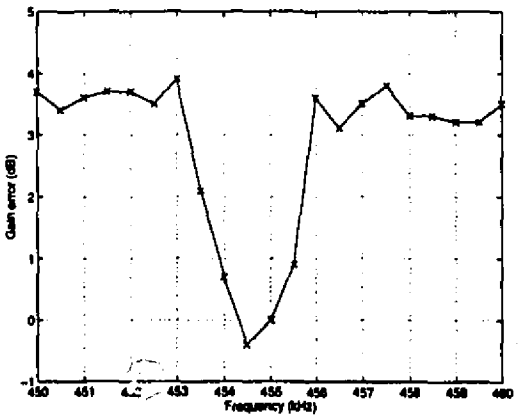


Figure 5.12: Calibration correction factors of bandpass delta-sigma oscillator.

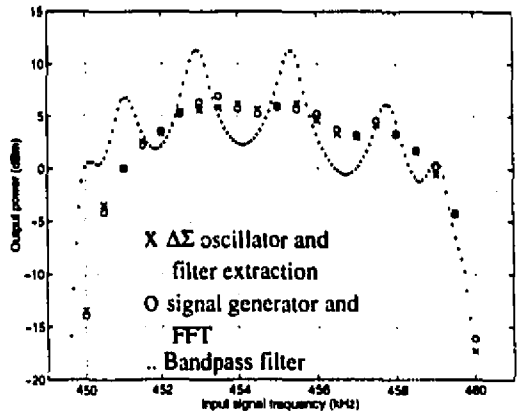


Figure 5.13: Frequency response of receiver for delta-sigma oscillator and HP 3314A signal generator.

cluded to provide a sense of the expected channel characteristics. By no means do we intend to imply that the other two measurement sets should correspond with these measurements, only come close.

5.5.3 Intermodulation Distortion

A significant analog metric of bandpass systems, intermodulation distortion, may not obtained using a single tone as stimulus. The two-tone BP ΔΣ oscillator described in Chapter 3, Section 3.2.3, was therefore implemented for the following experiment. The circuit was clocked at 1212 kHz and the second signal band, centered around 454.5 kHz, was utilized

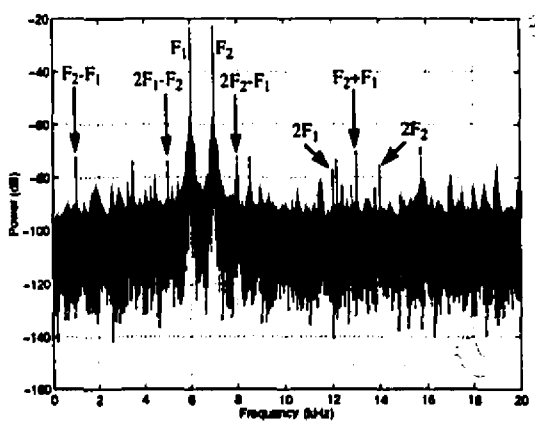


Figure 5.14: Power spectral density of sampled down-converted two-tone delta-sigma oscillator signal.

for this test. By generating two sinewaves at 151 and 152 kHz, images at 454 and 455 kHz were created. Since the carrier frequency was 448 kHz, two sinusoids appeared in the base-band at 6 and 7 kHz. The power spectral density (PSD) of the sampled output is shown in Figure 5.14. Intermodulation distortion products (F_2-F_1, $2F_1$-F_2,...) are clearly visible. The other signals are spurious tones introduced by our measuring set-up. Similar results would be obtained with commercial waveform generators.

5.6 Implementation Issues

Implementing this wireless BIST scheme in actual systems involves design trade-offs as well as some technological challenges. Some of these issues will be discussed below.

5.6.1 Digital Circuit

The BP $\Delta\Sigma$ oscillator may be implemented using signal processing resources available in the system. However in most applications the speed performance will not be sufficient. This is particularly true when the output processing task taps the same resources. Therefore a fully parallel dedicated circuit might be required. Hopefully, the proposed signal generator is area efficient enough. For example, the implementation used in the previous experiments only contains 7 adders. It is desirable to operate the digital circuit at the maximum clock frequency so that a low order image of the baseband can be used thus reducing the constraints on the pulse shape of the 1-bit DAC. Also signal frequency error caused by clock skew or clock frequency offset will be more important for higher order images.

5.6.2 1-Bit DAC

Being the only analog component of the BIST circuitry, the 1-bit DAC is a critical block. Even though images are guaranteed to appear because of the sampled data nature of the output signal, the power of these images will depend on the shape of the pulses generated by the 1-bit DAC. Also distortion and noise may be created if the pulses have varying shapes over time.

5.6.3 RF Multiplexer

The proposed wireless BIST scheme requires two multiplexer in the receiver signal path at the RF stage. Depending on the technology used at this stage, this may degrade the performance of the circuit.

5.7 Summary

A Built-In Self-Test scheme for bandpass type systems such as those used in today's wireless communication systems has been presented. The scheme is capable of measuring the SNR, frequency response and intermodulation distortion behavior of both the receiver and transmitter sections of an RF CODEC. At the heart of the scheme is a bandpass delta-sigma oscillator capable of generating high-frequency analog sinusoidal signals using only digital electronics. Such an approach will allow for high-frequency sinusoidal signal generation without the need for external trimming, essential for a BIST scheme. Also, measurements are made exclusively in the digital domain, further minimizing the dependency on any analog electronics. Experimental work using an FPGA for the signal generator and some discrete modulators and filters verify the test principles. The results corresponded quite closely with those obtained using a high-quality sinusoidal generator.

Chapter 6 : Conclusions and Future Work

6.1 Thesis Contributions

A class of digital frequency synthesizers based on digital resonator and delta-sigma modulation has been presented and characterized. This technique is much more efficient than other digital frequency synthesis scheme in terms of silicon area required for implementation. The extension of the lowpass delta-sigma oscillator to bandpass delta-sigma modulation was first demonstrated. Sinewave generation at any ratio of the clock frequency is therefore possible with this circuit. Even though another design was proposed at the same time, we demonstrated that our circuit is more stable. The technique was then generalized to any resonating digital network including a delta-sigma modulator. For the first time, issues introduced by the presence of a delta-sigma modulator in an infinite impulse response filter or resonator with respect to accuracy and stability of the output signals were addressed. Finally a novel third order digital resonator embedding a highpass delta-sigma modulator was presented. This is the first reported use of highpass delta-sigma modulation in a signal generation application. FPGA implementations have proved that these circuits are hardware efficient and have shown that it is possible to generate sinusoidal signals having a dynamic range in excess of 70 dB with this technique. An interesting application in which the bandpass delta-sigma oscillator plays a central role, wireless built-in self-test, was presented and its feasibility demonstrated through experiments. Analog measures such as signal-to-noise ratio, frequency response and intermodulation distortion may be obtained on bandpass systems such as those used in wireless communication circuits.

6.2 Directions for Future Work

A integrated circuit implementation of some of the devices mentioned in this thesis will be fabricated to assess the chip area required as well as the maximum operating frequency.

On the theoretical side, the signal amplitude predictability issue should be studied with respect to different factors such as the feedback coefficient magnitude, the initial conditions of the registers, the digital resonator topology and the delta-sigma modulator order. We believe this problem is the only hurdle preventing a widespread use of this signal generation technique. However, its effects may be accounted for in many applications by first simulating the devices before its use in a system.

Also the application of a sinewave stimulus is not sufficient to fully characterize many analog circuits. Indeed communication circuits for example process analog or digital signals that have been modulated first. The presented signal generator should be modified to allow analog and digital modulation such as frequency modulation (FM) or phase shift keying (PSK).

References

[1] A.K. Lu, G.W. Roberts and D.A. Johns, "A High-Quality Analog Oscillator Using Oversampling D/A Conversion Techniques," *IEEE Trans. on Circuits and Systems-II*, vol. 41, no. 7, pp. 437-444, July, 1994.

[2] M.F. Toner and G.W. Roberts, "A BIST Scheme for an SNR, Gain Tracking, and Frequency Response Test of a Sigma-Delta ADC," *IEEE Trans. on Circuits and Systems-II*, Vol. 41, No. 12, pp. 1-15, Jan., 1995.

[3] K.A. Kozma, D.A. Johns and A.S. Sedra, "An approach for Tuning High-Q Continuous-Time Bandpass Filters," *Proc. IEEE International Symposium on Circuits and Systems*, Seattle, pp. 1037-1040, May, 1995.

[4] B.R. Veillette and G.W. Roberts, "High Frequency Sinusoidal Generation Using Delta-Sigma Modulation Techniques", *Proc. IEEE International Symposium on Circuits and Systems*, Seattle, pp. 637-640, May, 1995.

[5] J.A. Crawford, *Frequency Synthesizer Design Handbook,* Artech House, Norwood, MA, 1994.

[6] V. Manassewitsch, *Frequency Synthesizers, Theory and Design*, 2nd ed., Wiley, NewYork, 1986.

[7] F.M. Gardner, *Phaselock Techniques*, John Wiley & Sons, New York, NY, 1979.

[8] T. Aytur and B. Razavi, "A 2GHz, 6mW BICMOS Frequency Synthesizer," *Proc. IEEE International Solid-State Circuits Conference*, pp. 264-265, Feb., 1995.

[9] J. Tierney, C.M. Rader and B. Gold, "A Digital Frequency synthesizer," *IEEE*

Trans. on Audio Electroacoustic, vol. 19, pp. 48-57, 1971.

[10] *Direct Digital Synthesizer Handbook*, Stanford Telecom, Santa Clara, CA, Mar. 1990.

[11] L.K. Tan, E. Roth, G.E. Yee and H. Samuelli, "An 800MHz Quadrature Digital Synthesizer with ECL-Compatible Output Drivers in 0.8μm CMOS," *Proc. IEEE International Solid-State Circuits Conference*, pp. 258-259, Feb., 1995.

[12] A Madisetti, A. Kwentus and A.N. Wilson, Jr., "A Sine/Cosine Direct Digital Frequency Synthesizer using an Angle Rotation Algorithm," *Proc. IEEE International Solid-State Circuits Conference*, pp. 262-263, Feb., 1995.

[13] H. Inose, Y. Yasuda and J. Marakami, "A telemetering system by code modulation - Δ–Σ modulation," *IRE Trans. Space, Electronic and Telemetry*, vol. SET-8, pp. 204-209, Sep., 1962.

[14] H. Inose and Y. Yasuda, "A unity bit coding method by negative feedback," *Proc. of the IEEE*, vol. 51, pp. 1524-1535, Nov., 1963.

[15] R. Schreier and M. Snelgrove, "Bandpass Sigma-Delta Modulation," *Electronics Letters*, vol. 25, no. 23, Nov., 1989, pp. 1560-1561.

[16] S.R. Norsworthy, "Optimal Nonrecursive Noise Shaping Filters for Oversampling Data Converters, Part 2: Applications," *Proc. IEEE International Symposium on Circuits and Systems*, Chicago, pp.1357-1360, May, 1993.

[17] W.L.Lee, *A novel higher order interpolative modulator topology for high resolution oversampling A/D converters*, Master's Thesis, Massachusetts Institute of Technology, Cambridge, pp. 34-36, June, 1987.

[18] R.Schreier, *Noise-Shaped Coding*, Ph.D.'s Thesis, University of Toronto, Toronto, 1991.

[19] X. Haurie and G.W. Roberts, "A Multiplier-Free Structure for 1-Bit High-Order Digital Delta-Sigma Modulators," *Proc. IEEE Midwest Symposium on Circuits and Systems*, Brazil, Aug., 1995.

[20] A.H. Nutall, "Some Windows with Very Good Sidelobe Behavior," *IEEE Trans. Acoustics Speech and Signal Processing*, vol. ASSP-29, no. 1, Feb., 1981.

[21] L. Longo, "A 15b 30kHz Bandpass Sigma-Delta Modulator," *Proc. IEEE International Solid-State Circuits Conference*, San Francisco, pp. 226-227, Feb. 1993.

[22] J.C. Candy, "A Use of Double Integration in Sigma-Delta Modulation," *IEEE Trans. on Communications*, vol COM-33, no. 3, pp. 249-258, Mar., 1985.

[23] K. Furuno, "Design of Digital Sinusoidal Oscillators With Absolute Periodicity," *IEEE Trans. on Aerospace and Electronic Systems*, pp. 1286-1298, Vol. AES-11, Nov., 1975.

[24] A.K. Lu and G.W. Roberts, "An analog multi-tone signal generator for built-in self-test applications," *Proc. IEEE International Test Conference*, Washington, pp. 650-659, Oct., 1994.

[25] G.W. Roberts and A.K. Lu, *Analog Signal Generation For Built-In Self-Test Of Mixed-Signal Integrated Circuits*, Kluwer Academic Publishers, Norwell, MA, USA, 1994.

[26] D.A. Johns and D.M. Lewis, "Design and Analysis of Delta-Sigma Based IIR Filters," *IEEE Trans. on Circuit and Systems-II*, vol. 40, no. 4, pp. 233-240, Apr., 1993.

[27] B.R. Veillette and G.W. Roberts, "Evaluating DSP Systems in Real-Time using an FPGA-Based Test Station," *Proc. Canadian Workshop on Field Programmable Devices*, pp. 207-211, Jun., 1995.

[28] B.R. Veillette and G.W. Roberts, "A Built-In Self-Test Strategy for Wireless Com-

munication Systems", *Proc. IEEE International Test Conference*, Oct., 1995.

[29] M.J. Ohletz, "Hybrid built-in self-test (HBIST) for mixed analogue/digital integrated circuits," *Proc. European Test Conference*, pp. 307-316, 1991.

[30] M.F. Toner and G.W. Roberts, "A BIST Technique for a Frequency Response and Intermodulation Distortion Test of a Sigma-Delta ADC," *IEEE VLSI Test Symposium*, Cherry Hill, pp. 60-65, Apr., 1994.

[31] D. Picken, "Testing GSM Mobile Phones," *Proc. of the Second Annual Wireless Symposium*, pp. 189-204, Santa-Clara, Feb., 1994.

[32] M. Padmanabhan and K. Martin, "Filter banks for time-recursive implementations of transforms," *IEEE Trans. on Circuits and Systems*, vol. 40, no. 1, pp. 41-50, Jan. 1993.

[33] P.J. Ashenden, *The VHDL Cookbook*, July 1990.

[34] *The Programmable Logic Data Book*, Xilinx, 1993

[35] *Design Compiler™ Reference Manual v3.0*, Synopsys, December 1992

[36] *Using Xilinx XC4000 with FPGA Compiler and Xilinx XACT5.0*, Synopsys, Feb., 1995.

[37] *XACT Reference Guide*, Xilinx, 1992,

Appendix A Raised Cosine Window

The mathematical operation used to obtain the spectral representation of a discrete-time signal is the discrete Fourier transform (DFT). The DFT is a decomposition of a finite length sequence into sinusoids of discrete frequencies. The Fast Fourier Transform (FFT) is an algorithm that implements efficiently the DFT. In this thesis, a raised cosine window is applied on digital samples before they are processed with an FFT. This appendix explains the reason for using windowing and exposes the characteristics of the raised cosine window.

Discrete signals are theoretically infinite in time. However the FFT may only be applied to a finite length sequence. In the time-domain, this operation is the multiplication of the infinite duration signal by a rectangular window. In the frequency domain, it corresponds to a convolution of the signal by the frequency spectrum of the rectangular window, shown in Figure A.1. The spectral power at integer index values is null except at index zero meaning that sinewaves in the input signal which periods are a divisor of the window length will have non-zero value at a only one index. If a sinewave is present in the signal but its frequency is not one of the discrete frequencies (non-coherent), then a phenomenon called fre-

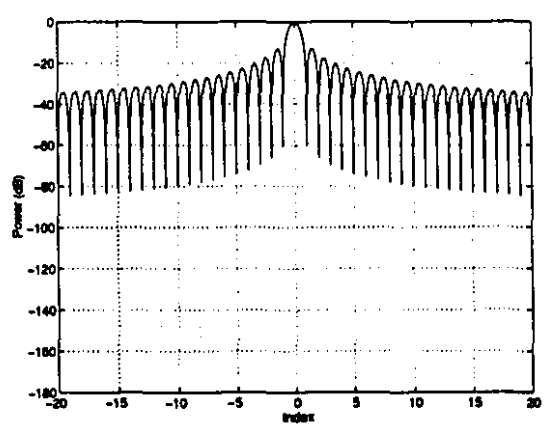


Figure A.1: Rectangular window frequency spectrum.

Window	Width of main lobe	First sidelobe decrease	Sidelobe decay
Rectangular	1	- 13 dB	6 dB/octave
Blackman	5	- 58 dB	18 dB/octave
Raised cosine	7	- 61 dB	42 dB/octave

Table A.1: Characteristics of windows.

quency leakage will take place. This sinewave will be decomposed in a number of sinewaves of decreasing power. The use of a different window helps reduce the frequency leakage for the non-coherent sinewaves. The characteristics of a window may be summarized by three parameters: the main lobe width, the largest sidelobe power relative to the main lobe and the sidelobe asymptotic decay. These parameters are summarized for the rectangular window, the Blackman window and the raised cosine window in Table A.1. The Blackman window, which frequency spectrum is shown in Figure A.2, has smaller sidelobes than the rectangular window but the main lobe is wider. The window used in this document is a raised cosine window [20] of the fourth-order. Its time-domain characteristics are described by the following equation:

$$w(t) = \frac{1}{L}\sum_{k=0}^{3} a_k \cos\left(\frac{2\pi k t}{L}\right) \tag{A.1}$$

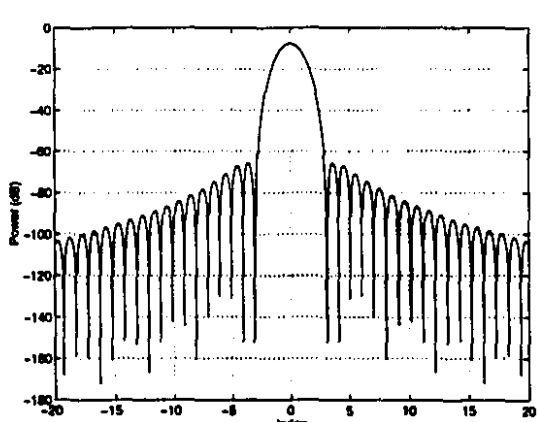


Figure A.2: Blackman window frequency spectrum.

a_0	10/32
a_1	15/32
a_2	6/32
a_3	1/32

Table A.2: Fourth-order raised cosine window coefficients.

where $|t| \leq L/2$. The coefficients a_k are listed in Table A.2. Its frequency content is displayed in Figure A.3 The most important feature of the raised cosine window is that the sidelobes created by non-coherent sampling fall-off sharply. This is very important when the dynamic range is large or when the signal band is surrounded by noise as is the case for $\Delta\Sigma$ modulator simulations. With the default rectangular window, the sidelobes from low-frequency and high-frequency noise will have significant power in the signal band to mask features

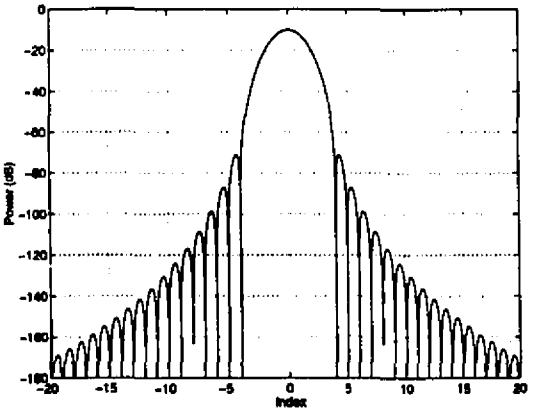


Figure A.3: Raised cosine window frequency spectrum.

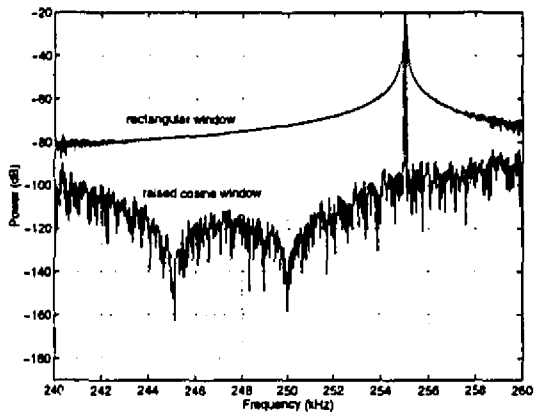


Figure A.4: Comparison of rectangular and raised cosine window.

such as noise shaping zeros. The simulation results of Figure 3.16 (b) are shown in Figure A.4 for both a rectangular window and the raised cosine window. It is obvious that the use of the rectangular window reduces the observed dynamic range and obscure important details such as noise shaping zeros.

Appendix B Prototyping Environment

This appendix presents the development system for real-time digital signal processing (DSP) applications that was used to obtain the experimental results of Chapter 4. It is centered around a Xilinx Field-Programmable Gate Array (FPGA) but may be extended to make use of any number of Field Programmable Devices (FPD). The development system features fast turn-around time, great flexibility, increased throughput and is at the heart of an integrated environment for automated mixed-signal testing. Furthermore, after validation, the design can be used in a simplified, stand-alone configuration of the development system or can be re-targeted to an Application Specific Integrated Circuit (ASIC) because it is described using a hardware description language (HDL).

B.1 Motivation

Because of the non-linear behavior of $\Delta\Sigma$ modulation, a large number of simulation must be performed in order to properly characterize $\Delta\Sigma$ oscillator designs. Computer simulation are relatively slow and the output must be post-processed with a fast Fourier transform (FFT) to obtain frequency-domain parameters. An FPGA-based development system has therefore been devised to allow rapid prototyping of new design ideas. It allows much faster throughput than computer simulations and simultaneous frequency information with the help of a spectrum analyzer. Since the digital and analog test equipment is connected to a workstation through an IEEE-488 bus, experimental results are collected automatically and may be redirected quickly to a post-processing software package. Another advantage is that the results of the real-time experiments can be compared directly with simulation results.

The turnaround time from a flow graph to an FPGA implementation is usually less than half a day. The digital signal processing flow graph is first translated to VHDL [33], a hardware description language (HDL). After validation using a VHDL simulator, the design is com-

piled for the Xilinx XC4000 FPGA family using Synopsys synthesis tools. The resulting netlist is then physically organized by the Xilinx software. As a result, a configuration data stream is generated that will be loaded on to the FPGA.

The development system hardware is centered around a single Xilinx XC4010 but it may be adapted to house a larger number of FPGAs. Digital stimulus VXI modules provide the clock and other control signals and selects the parameters for circuit operation. The output of the FPGA is fed to a DAC and the analog output can be examined on a spectrum analyzer or filtered and then viewed on an oscilloscope. $\Delta\Sigma$ modulation-based circuits are convenient in that they only require a 1-bit DAC. A Sun workstation is used for design processing and to provide high-level control of the digital modules. It may also be used to retrieve experimental results from the test equipment and provide post-processing.

A stand-alone configuration has also been developed. It allows the use of a circuit in a system when the corresponding design has been validated and characterized. Again based on a XC4010 FPGA, it now relies on an Electrically Erasable Programmable Read Only Memory (EEPROM) to supply the programming bit stream. Operating parameters are selected by a control unit in the system or by DIP switches. The only off-board requirements are the power supply and the clock signal. Should a finer version of a given design become available, then only the EEPROM needs to be replaced in the system.

B.2 Development System Hardware

A block diagram of the development system is shown in Figure B.1. The heart of the system is a Xilinx XC4010 FPGA which implements the digital signal processing network and provides an interface for external control of operating parameters such as coefficients and initial values of registers. The system may include more than one PLD to accommodate larger designs. The basic processing unit of this FPGA family is called a Combinational Logic Block (CLB). The selected device (XC4010) contains 400 CLBs for an estimated total capacity of 10,000 gates [34]. A Xilinx XChecker cable constitutes the link to the workstation for downloading the programming bits.

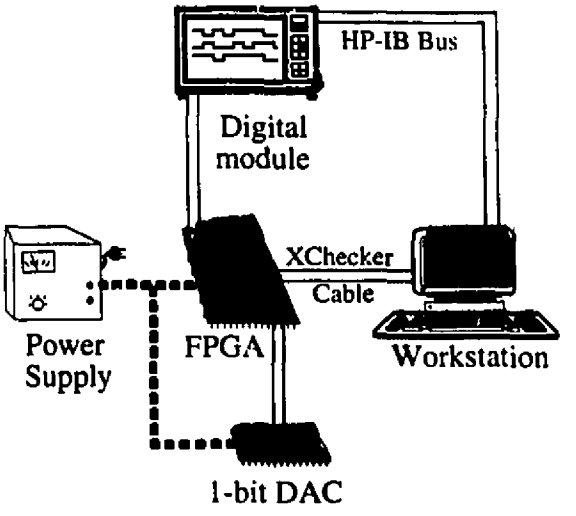


Figure B.1: Development system hardware.

A small board has been assembled with the FPGA, a few discrete transistors to implement the 1-bit D/A and connectors for the Xchecker cable, the digital modules and power supply. XC4000 devices are very easy to set up, requiring only a power supply and a small number of pull-up resistors. These FPGAs are electrically robust and can sustain a fair amount of abuse.

Digital stimuli are provided by an HP-E1450A digital timing module and an HP-E1452A digital I/O module from Hewlett-Packard. Both modules reside in a VXI frame. Four control lines and 2 eight-bit wide ports are used. The control lines are assigned to timing and control signals such as 'clock', 'reset' and 'load'. One port is allocated to address selection while the other is used for data so that 256 bits may be accessed. Other programming schemes are also possible and probably as easy to implement in VHDL. All 20 lines from the digital modules are each assigned to a pin. Therefore the place and route software must map the circuit I/O ports to these specific pins and this may lead to problems as explained in the following section. Experimental results may be retrieved digitally through the digital modules or by using the appropriate analog test equipment.

B.3 Stand-Alone Board

Once a design has been validated and characterized, there are two ways to incorporate it in a system. An application specific integrated circuit (ASIC) may be fabricated but it takes a few months to receive back the IC. Furthermore a better design may become available in the mean time and the ASIC then has to be redrawn. The use of FPGAs is thus preferable, at least in the short term, but it must be autonomous from the workstation and the digital modules. This sparked the creation of the stand-alone board which is illustrated in Figure B.3.

This stand-alone board is composed of a Xilinx XC4010 FPGA, an EEPROM, DIP switches, and a one-bit DAC. Power supply and a clock signal come from off-board although a clock signal could easily be generated on-board. The EEPROM contains the FPGA programming bit stream and upon power up the FPGA automatically retrieves this program. The DIP switches along with pull-up resistors allow user input for the operating parameters if a controlling unit is not available.

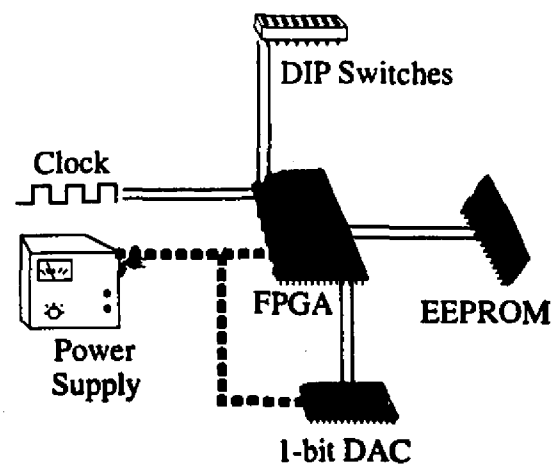


Figure B.2: Stand-alone board.

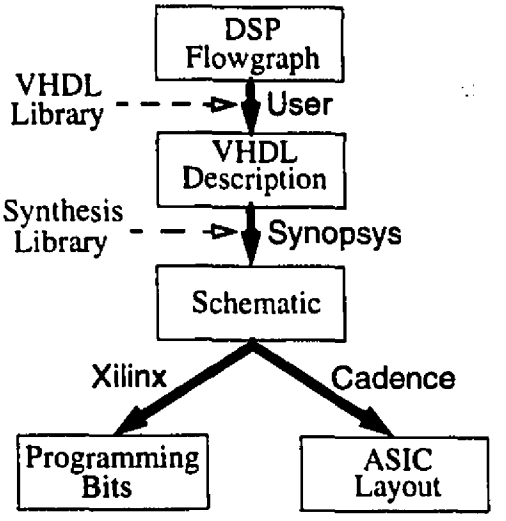


Figure B.3: Design flow.

B.4 Synthesis

Figure B.3 presents an overview of the design flow from a DSP flowgraph to an FPGA or an ASIC implementation. Digital signal processing flowgraphs are composed of a small number of basic blocks such as adders, registers and multiplexers. These blocks have been programmed in VHDL using parameterized behavioral description. They can therefore be instantiated with adaptable characteristics such as varying bus width. This is the reason why HDL behavioral description was preferred over schematic capture. However great care must be taken when using behavioral description as the code may not synthesize correctly even when it simulates correctly. Higher level entities such as $\Delta\Sigma$ modulators are translated from their flowgraphs using structural description. The DSP blocks are instantiated and the data flow between these blocks is detailed. An extensive library of parameterized entities has been developed and is now available for use by designers. The VHDL netlist of the top module is then simulated for functionality with Synopsys VHDL System Simulator (VSS). After the code has been debugged, the design is synthesized using Synopsys' Design Compiler [35] [36]. This operation takes up to a few hours to complete. Various compiling directives are available for the designer to influence the result. For example,

setting area and speed constraints will force the software to select either ripple-through or carry look-ahead implementation for adders.

It should be noted that ASIC design involves the same operations up to this point, only a different synthesis library being required. The final step for an ASIC would be to turn the schematic obtained from the compiler into a layout using a place and route software such as Cadence *Cell Ensemble*. Therefore, a design validated by an FPGA implementation can be quickly retargeted toward a standard cell IC implementation.

The netlist in Xilinx format (xnf) obtained from Synopsys is passed on to the XACT tools package for device mapping [37]. This is usually the bottleneck as big designs or designs that have many pins tied down may not be mapped completely. However FPGA vendors awareness to the problem is increasing and more efforts are placed into developing powerful routing tools. At this point, the placed netlist could be simulated for back-annotation but timing is not critical for the type of design being implemented here so this operation is bypassed. Finally a bit stream is generated that can be sent to the FPGA for programming.

B.5 Control and Test Software

The FPGA versatility and the programmability of the digital I/O modules make it effortless to interface them. A C program was written to allow users to modify any operating parameter from the workstation. This means that instead of entering coefficients and initial values of registers, the user may simply type in operational parameters such as frequency and amplitude and let the computer find the matching values from the equations. Experimental data collection from any test equipment is also available from the same program and results are organized in a format acceptable for post-processing software such as *Matlab*. Finally, vectored tests may be automated allowing parameter sweeps, such as frequency response, without further assistance from an operator.

Other environments have been explored for the development system control. Using Comdisco's *Signal Processing Worksystem (SPW)* turns the development system into an accelerator board while working in an environment that provides a number of DSP tools. Code

has been written to allow experimental data collection from *SPW*. Comdisco claims that *SPW* includes a tool for automatic VHDL code generation from a DSP flowgraph but this feature has not yet been investigated.

B.6 Summary

A FPGA based development system for DSP application has been presented. It is a very quick way to implement in hardware a digital signal processing design. It has even proven faster in some instances than a workstation for extended simulations. FPGAs incorporate the advantages of both electronic circuits and programming: they are very fast and they can link with other electronic circuits but they induce low stress because errors are not fatal and are easily corrected. Furthermore, in-system upgrading is very easy since only a memory chip needs to be replaced.

Even though the turn-around time for a design is longer for an FPGA than for a DSP processor, the resulting circuit is closer to the silicon implementation and as such is likely to draw more attention from the research community and industry. Furthermore, because of its abundant I/O ports, it is easier to interface with a controlling module, be it a computer or a set of DIP switches.

An FPGA implementation can be easily retargeted to any silicon process since the first steps of development up to and including synthesis are done with the same set of tools.

The most important drawback of FPGA is their limited capacity. This effect is further enhanced by the fact that these devices are not well suited for datapath applications.

Appendix C VHDL Source Code

C.1 Single Tone Bandpass Oscillator

```
--============================================================
--      Progammed by B. Veillette
--      September 1994
--============================================================

------------------------------------------------------------
-- Single tone (pi/2) bandpass oscillator core (4th order ds)
--
--      init1   Register 1 initial value
--      init2   Register 2 initial value
--      coefficient
--      clk     main clock, circuit uses rising edge
--      reset   load registers with default values
--              active high
--      sine    pulse density modulated sine wave
--      ow      overflow error
------------------------------------------------------------

library synopsys;
use synopsys.bv_arithmetic.all;

entity core_bandpass is
        generic (bus_size: positive := 16);
        port    (init1,
                 init2,
                 coef: in bit_vector (bus_size-1 downto 0);
                 clk,
                 reset: in bit;
                  sine,
                  ow: out bit);
end core_bandpass;

architecture structural of core_bandpass is

        component ds_4_bp
        generic (ds_size: positive := 1;
                 intbits: positive := 1;
                 fracbits: positive := 1);
                 port    (wide: in bit_vector(ds_size-1 downto 0);
                          pdm: out bit;
                          clk: in bit;
                          reset: in bit;
                          ow: out bit);
```

```
        end component;

        component bv_flipflop_sd
                generic (size: positive := 16);
                port    (value: out bit_vector(size-1 downto 0);
                        next_value: in bit_vector(size-1 downto 0);
                         default: in bit_vector(size-1 downto 0);
                        clk: in bit;
                        reset: in bit);
        end component;

        component adder
                generic (size: positive := 16);
                port    (a, b: in bit_vector(size-1 downto 0);
                        x: out bit_vector(size-1 downto 0);
                         ow: out bit);
        end component;

        component coef_mux2
                generic (size: positive := 16);
                port    (selector: in bit;
                        choice: out bit_vector(size-1 downto 0);
                         coef: in bit_vector(size-1 downto 0));
        end component;

        component bit_flipflop_sr
                port    (value: out bit;
                        next_value: in bit;
                        clk: in bit;
                        reset: in bit);
        end component;

        signal  mux_out,
                reg1_in,
                reg1_out,
                reg2_out,
                reg2_out_neg: bit_vector(bus_size-1 downto 0);
        signal  ow_ds,
                ow_a1,
                b_stream: bit;

begin
        register1: bv_flipflop_sd
                generic map (bus_size)
                port map (reg1_out, reg1_in, init1, clk, reset);

        modulator: ds_4_bp
                generic map (bus_size, 3, 8)
                port map (reg1_out, b_stream, clk, reset, ow_ds);
```

```
        mult1: coef_mux2
                generic map (bus_size)
                port map (b_stream, mux_out, coef);

        add1: adder
                generic map (bus_size)
                port map (mux_out, reg2_out_neg, reg1_in, ow_a1);

        register2: bv_flipflop_sd
                generic map (bus_size)
                port map (reg2_out, reg1_out, init2, clk, reset);

        reg2_out_neg <= - reg2_out;
        ow <= ow_ds or ow_a1;

        buf_sine: bit_flipflop_sr
                port map (sine, b_stream, clk, reset);

end structural;

------------------------------------------------------------------
-- Fourth order Bandpass delta-Sigma centered at pi/2
--
--      ds_size         bus size of the input
--      intbits         extra integer bits
--      fracbitsextra fractional bits
--
--      wide    modulator input
--      pdm     single bit output
--      clk     main clock, circuit uses rising edge
--      reset   load registers with null value (synchronous)
--      ow      overflow error
------------------------------------------------------------------

library synopsys;
use synopsys.bv_arithmetic.all;

entity ds_4_bp is
        generic (ds_size: positive := 1;
                intbits: positive := 1;
                fracbits: positive := 1);
        port    (wide: in bit_vector(ds_size-1 downto 0);
                pdm: out bit;
                clk: in bit;
                reset: in bit;
                ow: out bit);
end ds_4_bp;

architecture structural of ds_4_bp is
```

```
        component bv_flipflop_sr
              generic (size: positive := 1);
              port    (value: out bit_vector(size-1 downto 0);
                next_value: in bit_vector(size-1 downto 0);
                clk: in bit;
                reset: in bit);
        end component;

        component adder
              generic (size: positive := 16);
              port    (a, b: in bit_vector(size-1 downto 0);
                      x: out bit_vector(size-1 downto 0);
                       ow: out bit);
        end component;

        component shift_right
              generic (size: positive := 16;
                      shift: positive := 8);
              port    (a: in bit_vector(size-1 downto 0);
                      x: out bit_vector(size-1 downto 0));
        end component;

        signal  widex,
                regnet1,
                regnet2,
                regnet3,
                regnet4,
                shnet1,
                shnet2,
                adnet1,
                adnet2,
                adnet3,
                adnet4,
                adnet5,
                adnet6,
                regnet2_n,
                shnet2_n,
                adnet2_n,
                feedback: bit_vector(ds_size+intbits+fracbits-1
                                     downto 0);
        signal  ow1,ow3,ow5,ow6: bit;

Begin
        widex(fracbits-1 downto 0) <= ext("0", fracbits);
        widex(ds_size+intbits+fracbits-1 downto fracbits
                <= sxt(wide, ds_size+intbits);

-- Direct form registers

        reg1: bv_flipflop_sr
```

```
                generic map (ds_size+intbits+fracbits)
                port map (regnet1, adnet5, clk, reset);

        reg2: bv_flipflop_sr
                generic map (ds_size+intbits+fracbits)
                port map (regnet2, regnet1, clk, reset);

        reg3: bv_flipflop_sr
                generic map (ds_size+intbits+fracbits)
                port map (regnet3, regnet2, clk, reset);

        reg4: bv_flipflop_sr
                generic map (ds_size+intbits+fracbits)
                port map (regnet4, regnet3, clk, reset);

-- Input

        add1: adder
                generic map (ds_size+intbits+fracbits)
                port map (widex, adnet6, adnet1, ow1);

-- Comparator

        pdm <= adnet1(ds_size+intbits+fracbits-1);

-- Error

        adnet2(ds_size+fracbits-2 downto 0) <=
                adnet1(ds_size+fracbits-2 downto 0);

        process (adnet1)
        begin
                if (adnet1(ds_size+intbits+fracbits-1) = '1') then
                        adnet2(ds_size+intbits+fracbits-1
                                downto ds_size+fracbits-1) <=
                        adnet1(ds_size+intbits+fracbits-1 downto
                                ds_size+fracbits-1)+ext("1",int-
bits+1);
                else
                        adnet2(ds_size+intbits+fracbits-1 downto
                                ds_size+fracbits-1) <=
                        adnet1(ds_size+intbits+fracbits-1 downto
                                ds_size+fracbits-1) + sxt("1",int-
bits+1);
                end if;
        end process;

        adnet2_n <= - adnet2;

-- Feedback
```

```
        add5: adder
                generic map (ds_size+intbits+fracbits)
                port map (adnet2_n, adnet3, adnet5, ow5);

        mult1: shift_right
                generic map (ds_size+intbits+fracbits, 8)
                port map (regnet2, shnet1);

        regnet2_n <= - regnet2;

        add4: adder
                generic map (ds_size+intbits+fracbits)
                port map (shnet1, regnet2_n, adnet4, OPEN);

        mult2: shift_right
                generic map (ds_size+intbits+fracbits, 1)
                port map (regnet4, shnet2);

        shnet2_n <= - shnet2;

        add3: adder
                generic map (ds_size+intbits+fracbits)
                port map (adnet4, shnet2_n, adnet3, ow3);

-- Feedforward

        add6: adder
                generic map (ds_size+intbits+fracbits)
                port map (regnet2, shnet2, adnet6, ow6);

        ow <= ow1 or ow3 or ow5 or ow6;

end structural;

-------------------------------------------------------------------
-- Bit vector flip-flop with synchronous default
--
--      value           output (Q)
--      next_value      input (d)
--      default         reset value
--      clk             main clock, circuit uses rising edge
--      reset           load default value (synchronous)
-------------------------------------------------------------------

entity bv_flipflop_sd is
        generic (size: positive := 1);
        port    (value: out bit_vector(size-1 downto 0);
                next_value: in bit_vector(size-1 downto 0);
                default: in bit_vector(size-1 downto 0);
```

```
                clk: in bit;
                reset: in bit);
end bv_flipflop_sd;

architecture behavioural of bv_flipflop_sd is
begin
        process
        begin
                wait until clk'event and clk = '1';
                if reset = '1' then
                        value <= default;
                else
                        value <= next_value;
                end if;
        end process;
end behavioural;

-------------------------------------------------------------------
-- Bit vector flip-flop with synchronous reset
--
--      value           output (Q)
--      next_value      input (d)
--      clk             main clock, circuit uses rising edge
--      reset           loads zero (synchronous)
-------------------------------------------------------------------

library synopsys;
use synopsys.bv_arithmetic.all;

entity bv_flipflop_sr is
              generic (size: positive := 1);
              port    (value: out bit_vector(size-1 downto 0);
                       next_value: in bit_vector(size-1 downto 0);
                       clk: in bit;
                       reset: in bit);
end bv_flipflop_sr;

architecture behavioural of bv_flipflop_sr is
begin
      process (reset, clk, next_value)
      begin
        if (clk'event and clk = '1') then
                       if (reset = '1') then
                              value <= sxt("0", size);
                else
                       value <= next_value;
                end if;
                end if;
      end process;
end behavioural;
```

```
------------------------------------------------------------------
-- Bit flip-flop with synchronous reset to 0
--
--      value           output (Q)
--      next_value      input (d)
--      clk             main clock, circuit uses rising edge
--      reset           loads zero (synchronous)
------------------------------------------------------------------

entity bit_flipflop_sr is
        port    (value: out bit;
                next_value: in bit;
                clk: in bit;
                reset: in bit);
end bit_flipflop_sr;

architecture behavioural of bit_flipflop_sr is
begin
        process
        begin
                wait until clk'event and clk = '1';
                if reset = '1' then
                        value <= '0';
                else
                        value <= next_value;
                end if;
        end process;
end behavioural;

------------------------------------------------------------------
-- Adder
--
--      a       input 1
--      b       input 2
--      x       output
--      ow      overflow error
------------------------------------------------------------------

library synopsys;
use synopsys.bv_arithmetic.all;

entity adder is
        generic (size: positive := 1);
        port    (a, b: in bit_vector(size-1 downto 0);
                x: out bit_vector(size-1 downto 0);
                ow: out bit);
end adder;

architecture behavioural of adder is
```

```
begin
        process (a, b)
                variable y: bit_vector(size-1 downto 0);
                variable owp, own: bit;
        begin
                y := (a + b);
                x <= y;
                own := a(size-1) and b(size-1) and not y(size-1);
                owp := not a(size-1) and not b(size-1) and y(size-
1);
                ow <= own or owp;
        end process;
end behavioural;

-------------------------------------------------------------------
-- Coefficient multiplexer 2x1
--
--      selector        selects negative (0) or positive (1)
--      choice          output = +/- input
--      coef            input
-------------------------------------------------------------------

library synopsys;
use synopsys.bv_arithmetic.all;

entity coef_mux2 is
        generic (size: positive := 1);
        port    (selector: in bit;
                choice: out bit_vector(size-1 downto 0);
                coef: in bit_vector(size-1 downto 0));
end coef_mux2;

architecture behavioural of coef_mux2 is
begin
        process (selector, coef)
        begin
                if selector = '1' then
                        choice <= coef;
                else
                        choice <= -coef;
                end if;
        end process;
end behavioural;

-------------------------------------------------------------------
-- Fixed shift right
--
--      shift   number of digits to the right
--
--      a       input
```

```
--      x       output
-------------------------------------------------------------------

library synopsys;
use synopsys.bv_arithmetic.all;

entity shift_right is
        generic (size: positive := 1;
                 shift: positive := 1);
        port    (a: in bit_vector(size-1 downto 0);
                 x: out bit_vector(size-1 downto 0));
end shift_right;

architecture behavioural of shift_right is
begin
        process (a)
        begin
                 x (size-1 downto size-shift) <= sxt(a(size-1 downto
size-1), shift);
                 x  (size-1-shift  downto  0)  <=  a(size-1  downto
shift);
        end process;
end behavioural;
```

C.2 Test Bench For Single Tone Bandpass Oscillator

```
--=============================================================
--      Programmed by B. Veillette
--      September 1994
--=============================================================

use work.core_bandpass;

entity test_bandpass is
end test_bandpass;

architecture structure of test_bandpass is

        component core_bandpass
                 generic (bus_size: positive := 16);
                 port    (init1,
                          init2,
                          coef: in bit_vector (bus_size-1 downto 0);
                          clk,
                          reset: in bit;
                  sine,
                  ow: out bit);
        end component;
```

```
        signal  init1,
                init2,
                coef:   bit_vector (15 downto 0);

        signal  clk,
                reset,
                sine,
                ow:     bit;

begin
        init1 <= X"4000";
        init2 <= X"0000";
        coef  <= X"080B";

        osc: core_bandpass
                generic map (16)
                port map (init1, init2, coef, clk, reset, sine, ow);

        reset <= '0', '1' after 1000 ns, '0' after 3000 ns;

        clock_driver: process
        begin
                clk <= '0', '1' after 500 ns;
                wait for 1000 ns;
        end process clock_driver;

end structure;

configuration tb_bp of test_bandpass is
        for structure
        end for;
end tb_bp;
```