ELSEVIER

Computer Standards & Interfaces 19 (1998) 189–203

COMPUTER STANDARDS & INTERFACES

Behavioral modeling of delta–sigma modulators

Alan J. Davis [a,*], Godi Fischer [b,1]

[a] *Weapon Systems Department, Naval Undersea Warfare Center, Newport, RI 02841-1708, USA*
[b] *Department of Electrical Computer Engineering, University of Rhode Island, Kingston, RI 02881-0805, USA*

Abstract

This paper presents an overview of DelSi, a MATLAB toolbox developed for the design and simulation of delta–sigma modulators. The DelSi toolbox can synthesize a stable noise-transfer function for complex network structures. The modelling tool consists of two c/c^{++} programs, sdMod.c, a single stage delta–sigma modulator and firfilt.c, a decimating FIR filter. Cascaded modulators can be constructed using the programs as building blocks. The simulation tools support the inclusion of nonideal effects caused by circuit imperfections, such as amplifier finite open-loop gain, incomplete settling and saturation as well as capacitor ratio mismatches. Two design examples are presented: one for a 1–1–1 cascade and the other for a single-stage fifth modulator.

Keywords: Delta–sigma modulation; Behavioral modelling; Simulation; Computer-aided design; Analog circuit imperfections

1. Introduction

Typically, delta–sigma A/D converters are characterized at the system level by specifying minimum sampling rate, desired resolution (in bits and/or in dB), linearity, maximum chip area, and power constraints. It is the job of the designer to translate these system-level specifications into circuit schematics and ultimately, a transistor-level description in a given device technology. The synthesis and analysis steps required for a complete design are outlined in Fig. 1 along with the number and cost of simulations. Before the circuits can be designed, the modulator loop filter must be synthesized at the transfer function level to obtain a stable coefficient set which optimizes SNR + THD. Subsequently, operational-amplifier (op-amp) performance requirements must be determined and, presuming a switched-capacitor based implementation, appropriate capacitor sizes must be chosen. All of this must be done prior to the actual design to be successful on first silicon.

Unfortunately, there are no closed-form solutions available which predict the exact performance of delta–sigma data converters [11–14]. Thus, the designer must select a suitable set of filter coefficients and adjust them based upon modulator output spectra computed by a high level circuit simulator. This is an iterative process which leads to incremental improvements in the noise-shaping transfer function (NTF).

* Corresponding author. Tel.: +1 401 8745482; fax: +1 401 7826422; e-mail: davis@ele.uri.edu
[1] E-mail: fischer@ele.uri.edu

0920-5489/98/$ - see front matter
PII: S0920-5489(98)00016-6

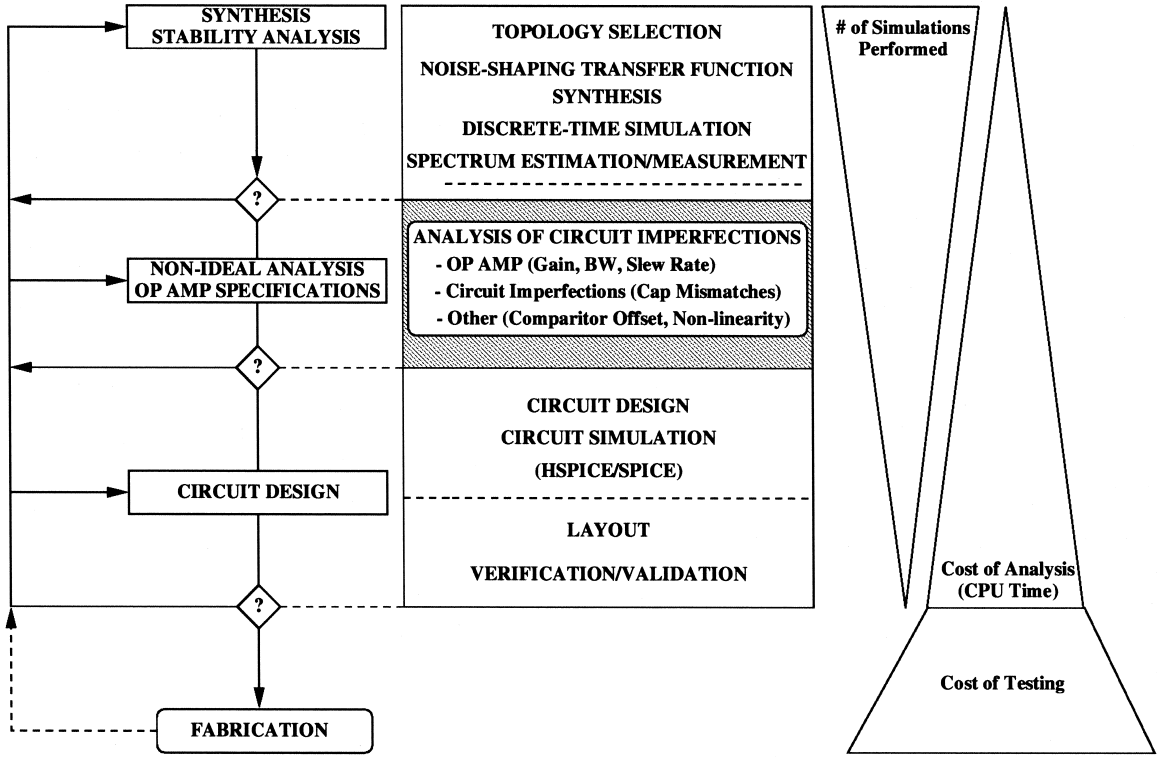


Fig. 1. Delta–sigma modulator design tasks in chronological order along with simulation-time and cost requirements.

The only reliable way to analyze stability and performance is to simulate the delta–sigma modulator in the discrete time domain. Traditional circuit simulators, such as HSPICE, which model devices at the semiconductor level, are ill-suited for this task since they consume an excess amount of CPU time. High spectral resolution requires many time samples (e.g., 10^{16} to 10^{20}) [11,13,19]. A device-level circuit simulator adjusts the time step in accordance with the fastest transients of the system and thus generates a high multiple of the actually required time samples. Consequently, simulating at the difference-equation level is orders of magnitude faster and more efficient than simulating at the device level. Due to the complexity of the device models, transistor-level circuit simulators may also suffer from rounding effects which can preclude an exact evaluation of the SNR + THD when the desired dynamic range exceeds 100 dB.

The nonideal analysis, identified by the shaded portion of Fig. 1 is perhaps the greatest obstacle to achieving a successful design. Once a stable NTF has been selected, circuit nonidealities can still significantly degrade performance. Therefore, it is important for a ‘complete’ modelling solution to aid the designer in making intelligent tradeoffs using nonideal components. This can be accomplished by modelling the op-amps and other effects, such as capacitor errors. The absence of simulators which include such nonideal effects prompted the development of DelSi.

2. The DelSi toolbox

DelSi offers a wide variety of modulator topologies from which the designer can choose. The selection includes both single- and multistage network structures. DelSi not only offers the speed advantages of a

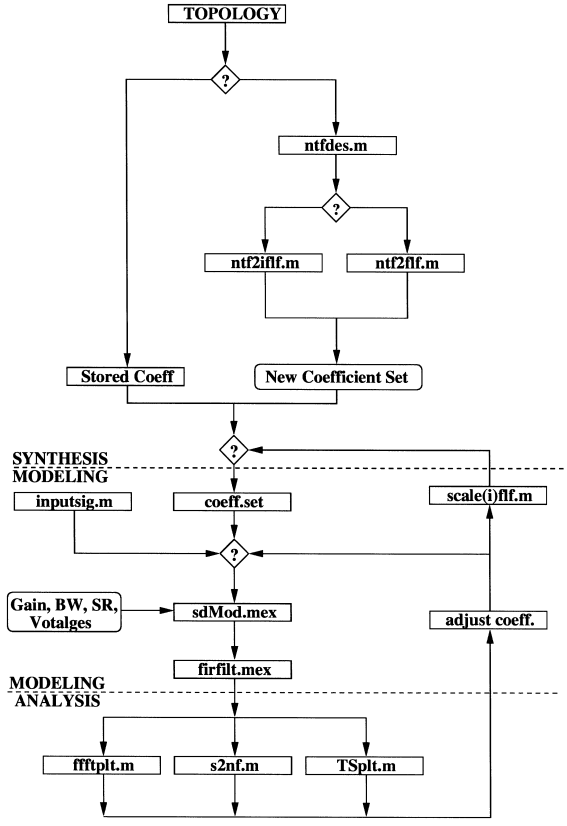


Fig. 2. Block diagram of general DelSi functions.

difference-equation based simulator, it also provides the designer with direct insight into how op-amp nonidealities affect the system performance.

The major components of the DelSi toolbox are depicted in Fig. 2. The tool suite, composed of a set of m-files and c-programs, contains the three essentials of any complete design package: synthesis, modelling, and analysis tools. The synthesis tools place the poles and zeros of the NTF. These tools also translate the NTF into the filter coefficients for a specific modulator network topology. Any of the common delta–sigma modulator network structures can be realized using either sdMod.mex (Fig. 3) alone or a combination of sdMod.mex, which models a single-stage modulator and firfilt.mex (Fig. 4), which is a decimating FIR filter. The analysis tools provide the designer with views of both the spectrum and time series at each integrator output and estimates of the maximum SNR + THD of the system.

DelSi is implemented as a toolbox in the MATLAB [2] environment. MATLAB was chosen since it has good cross platform support. Existing features have been amortized from its extensive library of functions [2,16].

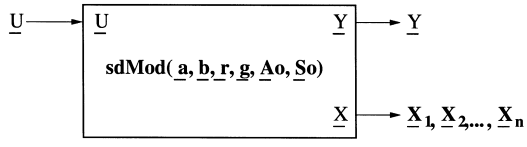


Fig. 3. sdMod.mex block diagram.

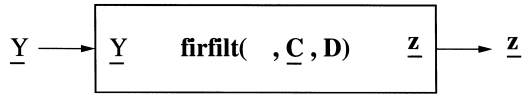


Fig. 4. firfilt.mex block diagram.

The discrete-time simulation of a nonlinear difference equation for a large number of samples requires at least the loop in the modeling tool to be implemented in c/c^{++} instead of an m-file (the common programming tool in MATLAB). This is readily accomplished by using a MATLAB EXecutable c program (mex file) [1]. The mex implementation provides a function call identical to the m-file and it reduces the execution time of the nonlinear loop by several orders of magnitude.

3. Architecture of the modeling tool

sdMod.mex is capable of modeling single-stage delta–sigma modulators of any practical order (≤ 8) with quantizer levels (single-quantizer) ranging from 2 (1-bit) to 16 (4-bit). This is accomplished by providing a discrete-time simulation of two general topologies, the inverse follow the leader feedback (iflf) and the follow the leader feedback (flf) network structures [3,15]. The iflf network structure used in DelSi (Fig. 5) consists of a chain of integrators which feeds the quantizer output back to the input nodes of each integrator [3,4]. The flf network structure consists (Fig. 6) of a chain of integrators with the outputs fed into a weighted summing node prior to quantization. The quantizer output is then fed back to the input node of the modulator [3,4,17]. Both general network structures contain resonator loops to place finite zeros and signal feed-in coefficients to provide some filtering for the input signal. From these two structures, specific modulator topologies can be derived through the choice of delays. sdMod.mex provides nonideal inputs, such as finite gain for each amplifier through the A_o vector. The other nonideal parameters, such as the normalized slew rate, normalized unity gain bandwidth, reference voltages, delay patterns, quantizer levels and others are lumped into the S_o vector (Fig. 3).

sdMod.mex allows one to model a delta–sigma modulator as an ideal system, with user-specified nonidealities (e.g., ideal settling finite gain) or with all nonideal effects present. In our experience, the ability to turn off the modelling of nonideal effects has proven to be an essential salient feature of this type of modelling tool. An ideal simulation is particularly useful in the early stages of a design for stability analysis or idle tone prediction.

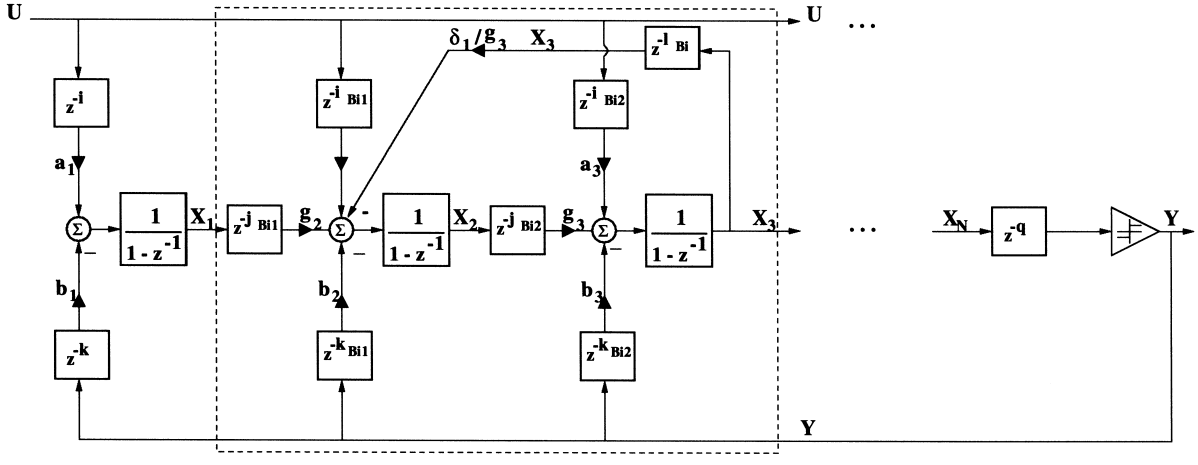


Fig. 5. The *n*th-order iflf network structure can be generated by repeating the BiQuadblock enclosed in the dashed lines.

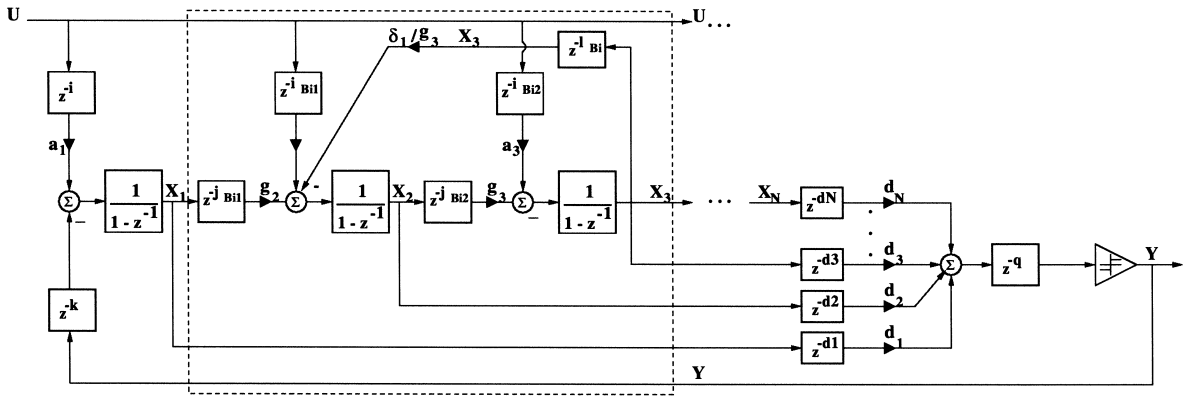


Fig. 6. The *n*th-order flf network structure can be generated by repeating BiQuad blocks in a similar manner to that of the iflf network structure.

Many network structures are more sensitive to one or two particular nonidealities. The designer may wish to focus exclusively on those particular nonideal effects.

Fig. 7 illustrates the range of cascaded network structures possible with the DelSi toolbox. By making multiple calls to sdMod.mex almost any reported multistage delta–sigma modulator can be simulated. The use of higher order stages or separate types of quantizers in each stage is easily accomplished. The noise-cancellation filter is created in a similar manner with an m-file which makes appropriate calls to firfilt.mex. M-files exist for the 1–1–1 cascade, also known as the MASH [10], the 2–1 cascade, the 3–2 cascade [7], and the 3–3 cascade [7]. By taking this approach, the creation of a new cascaded network structure or the modification of an existing one is dramatically simplified.

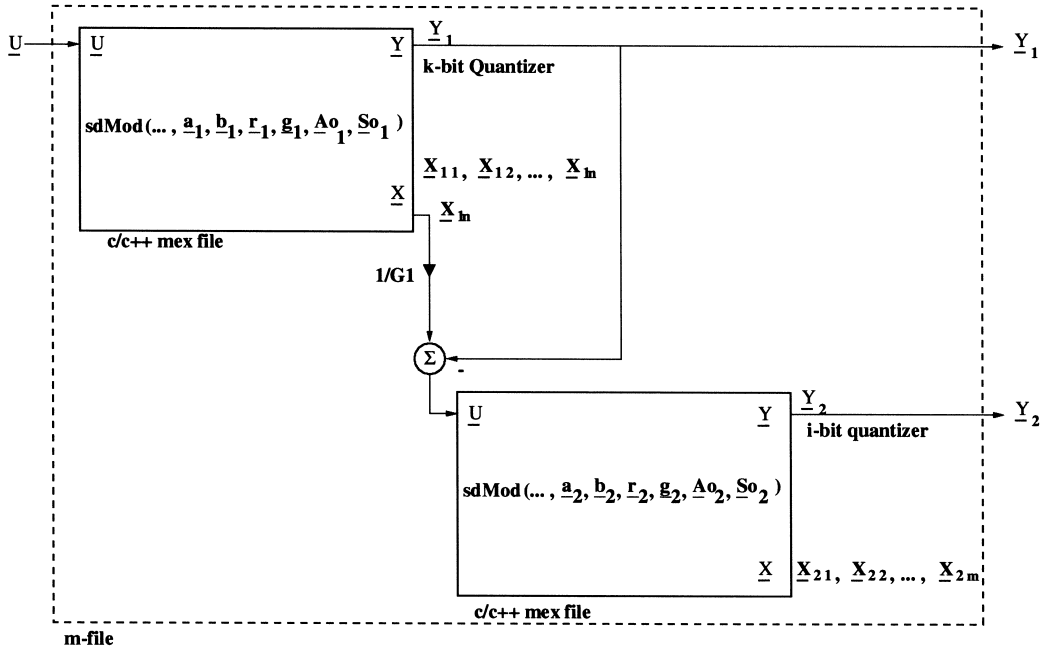


Fig. 7. Construction of a cascaded delta–sigma modulator using sdMod.mex.

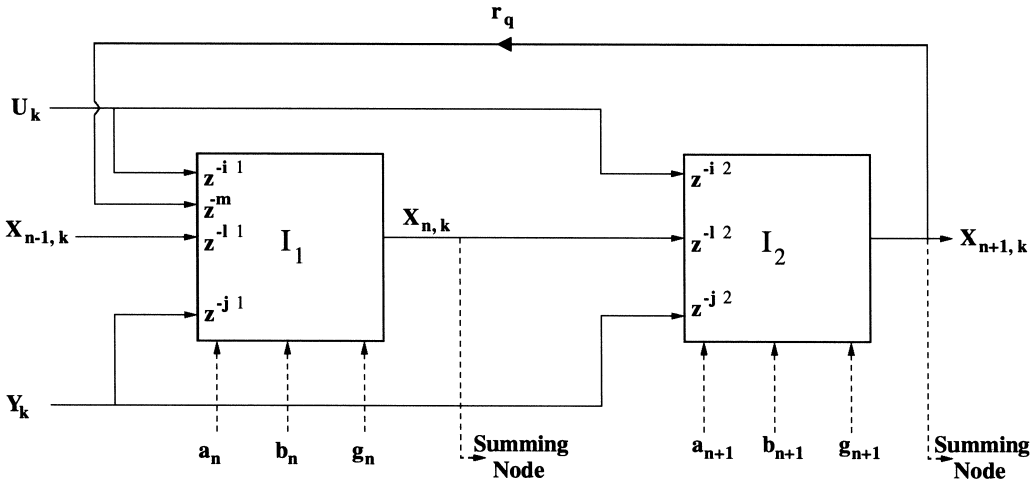


Fig. 8. BiQuad block primitive implemented in sdMod.mex.

The modelling tool generates all of the network structures from two primitives, a first-order integrator and a biquad integrator. The iflf and the flf network structures can be realized through appropriate combinations of the integrator block and BiQuad blocks. Fig. 8 provides a description of the BiQuad block. The delays, which distinguish different iflf and flf network structures, are broken down to the following functional units: the single integrator, the BiQuad, the prequantizer, and the postquantizer.

The discrete-time integrator model (Fig. 9) must provide an exact first-order model of the forward Euler stray-insensitive switched capacitor integrator employing a nonideal op-amp. Finite gain is modeled by including a gain term, G_o, at the integrator input and a leak term, P_o, in the feedback path. Eq. (1) shows the z-domain integrator transfer function resulting from an amplifier with finite gain. The influence of P_o and G_o on the amplifier open loop gain [8,9], A_o, is described by Eq. (2). Obviously, the gain and leak terms approach one when the amplifier gain is infinite.

The integrator output is passed through a nonlinear transfer function block to simulate linear setting, slewing and the amplifier saturation voltage [5,8]. The smooth curve in Fig. 9 represents linear settling while the dashed curve represents the nonlinear effects which include slewing and saturation. Notice that the transfer function in

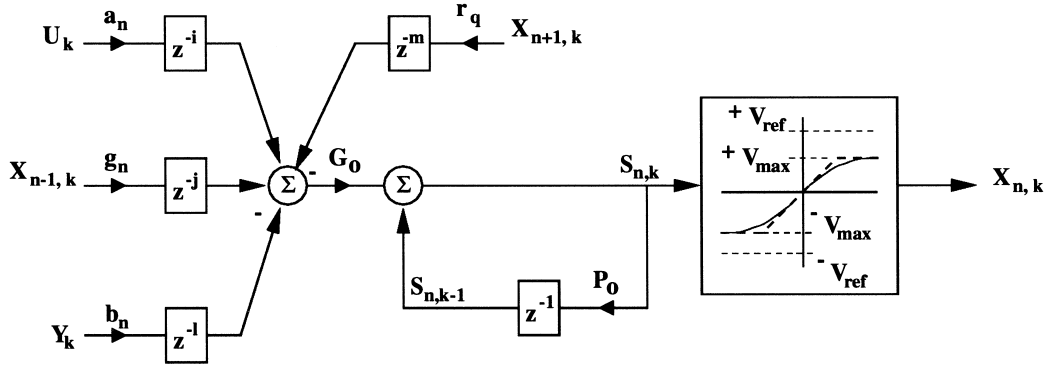


Fig. 9. Generic discrete-time integrator model used in sdMod.mex.

Fig. 9 indicates that the amplifier settling is linear, but saturation prevents the output voltage from reaching V_{ref}. Any combination of both linear and nonlinear effects due to the amplifier can be incorporated.

$$\frac{V_{out(z)}}{V_{in(z)}} = \frac{G_o}{1 - P_{o(z^{-1})}} \tag{1}$$

$$G_o = \frac{A_o}{A_o + 1 + \alpha}; \quad P_o = \frac{A_o + 1}{A_o + 1 + \alpha} \tag{2}$$

The parameter α represents the sum of the integrator input capacitors divided by the value of the feedback capacitance.

Settling behavior must be subdivided into three distinct cases: linear settling only, a combination of linear settling and nonlinear slewing, and nonlinear slewing only [5,8,18]. The computation of the resulting output voltage step is performed as follows:

$$\Delta_o = \begin{cases} \Delta_i\left[1 - e^{-\frac{T_0}{\tau}}\right] & \text{if } \Delta_i \le SR\tau \\ \Delta_i - \text{sign}(\Delta_i)SR\tau e^{-\frac{1}{\tau}\left[T_0 + \tau - \frac{\Delta_i}{SR}\right]} & \text{if } SR(T_0 + \tau) > \Delta_i > SR\tau \\ \text{sign}(\Delta_i)SRT_0 & \text{if } \Delta_i \ge SR(T_0 + \tau) \end{cases} \tag{3}$$

where Δ_i and Δ_o in Eq. (3) denote the integrator input and output voltage steps, respectively while T_0 and τ represent the available settling time (approximately half a clock cycle) and the linear settling time constant of each integrator stage. The latter is inversely related to the amplifier bandwidth GB. The parameter τ in Eq. (3) is unique for each integrator [8].

4. Examples

The advantages of the DelSi toolbox can best be illustrated through selected design examples. We have chosen the MASH as the first example and a single-stage modulator, the fifth-order iflf (iflf5) for the second design example. These examples underscore how features of the toolbox can be tailored to meet specific design needs.

4.1. Triple first-order cascade (MASH)

The NTF synthesis for the MASH [10] topology (see Fig. 10) is simple. The system consists of three first-order loops whereby each subsequent stage estimates the quantization noise of its predecessor. The two intermediate noise estimates, Y_2 and Y_3 are then subtracted from the primary output, Y_1, by a digital

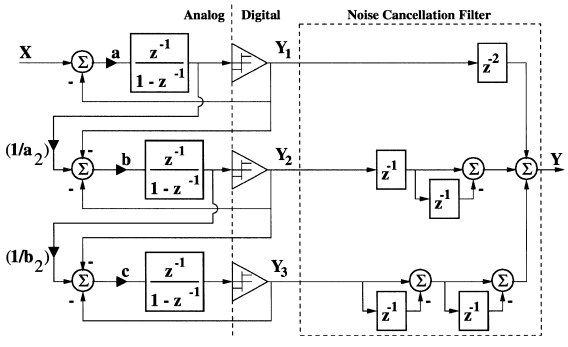


Fig. 10. Block diagram of triple first-order cascade (MASH).

noise-cancellation circuit [10]. Amplifier open-loop gains and capacitor ratio errors must be carefully analyzed since these will have the greatest impact on the overall performance.

With complete cancellation of the intermediate quantization noise terms, the performance of the MASH modulator will closely approach its theoretical maximum. Using a sinusoidal input voltage with a swing of V_{ref} the optimum value for the signal to noise ratio (SNR) of the MASH can be expressed as follows:

$$\text{SNR} = \frac{21}{2\pi^6}\text{OSR}^7 \tag{4}$$

where OSR denotes the oversampling ratio of the modulator. Thus, for an OSR of 64 a maximum SNR of 106.8 dB or 17.5 equivalent bits is predicted assuming ideal integrators. In practice, the performance typically falls short of the predicted value. This is due to incomplete cancellation of intermediate quantization noise terms caused by finite amplifier gain and C-ratio mismatch errors which are not third order noise shaped. These terms are completely cancelled only if the amplifier gain is infinite and the C-ratios are perfectly matched.

Capacitor mismatches between subsequent stages cause noise cancellation errors, which reduce the SNR + THD. For instance, the circuit element which is most critical for the cancellation of the quantization noise from the first stage is the loop gain $1/a$, from the capacitor ratio of the first stage which must be matched by $(1/a_2)$ in Fig. 10. If the value of a_2 does not exactly match the C-ratio, a, of the first stage, residual quantization noise will remain [8]. This uncancelled quantization noise, since it comes from a first order modulator, can have many unwanted spectral tones. The same is true for the loop gain of the second stage, $1/b$, the capacitor matched by $(1/b_2)$.

To illustrate how sensitive the MASH network structure is to amplifier gain and C-ratio mismatch errors, some simple expressions can be derived. If we rearrange the terms of the stray-insensitive switched-capacitor

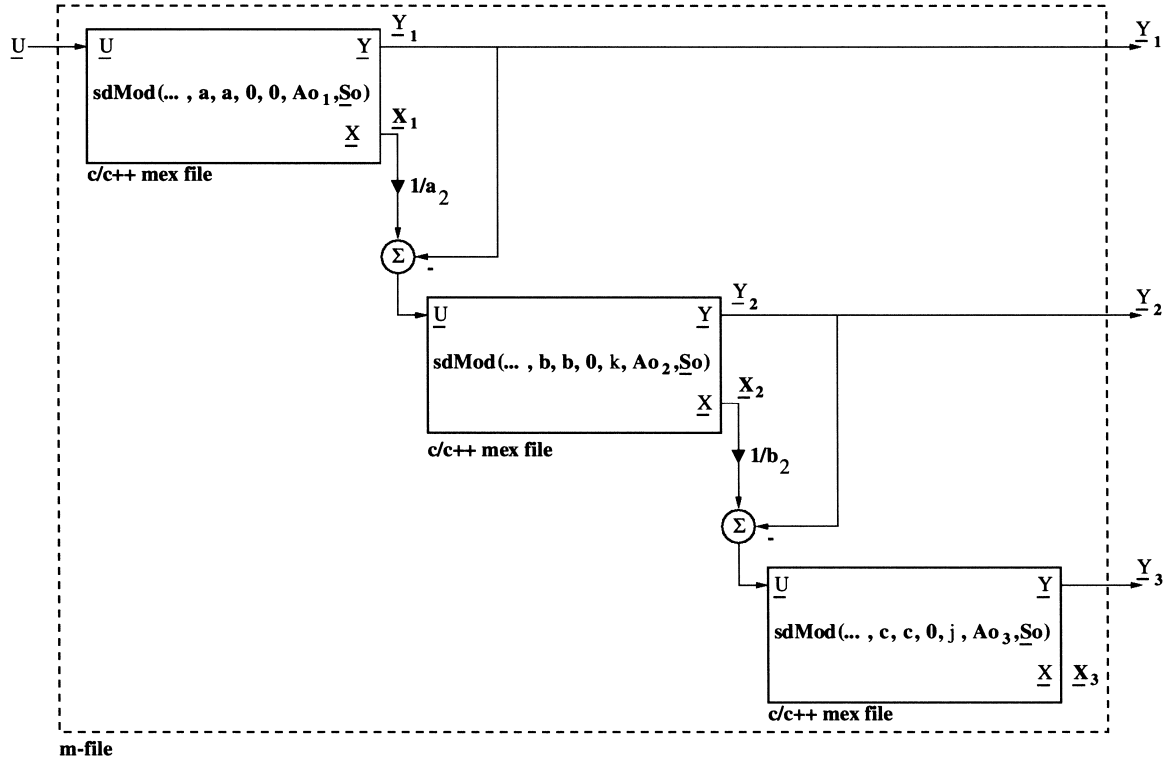


Fig. 11. Implementation of the MASH using sdMod.mex.

integrator (of Eqs. (1) and (2), the parasitic noise terms in of the MASH can be described by the following expression cite [6]:

$$\Delta N(z) = N_1\left[\epsilon_a z^{-2} + z^{-2}(1-z^{-1})(\epsilon_0 + \epsilon_{m1})\right] + N_2\left[z^{-1}(1-z^{-1})\epsilon_b + z^{-1}(1-z^{-1})^2(\epsilon_0 + \epsilon_{m2})\right] + N_3\left[(1-z^{-1})^2\epsilon_o + (1-z^{-1})^3\epsilon_0\right] \quad (5)$$

where $\epsilon_0 = \frac{1}{A_0\epsilon_a} = \frac{a}{A_0}$, $\epsilon_b = \frac{b}{A_0}$, and $\epsilon_c = \frac{c}{A_0}$, respectively.

Note that ϵ_{m1} and ϵ_{m2} denote the C-ratio mismatch errors between first and second and second and third stage, respectively. The coefficients ϵ_a, ϵ_b and ϵ_c represent the integrator phase errors of the three stages.

$$n_{ST1}^2 \approx e_{rms}^2\left[(\epsilon_{m1} + \epsilon_o)^2\frac{\pi^2}{3}\mathrm{OSR}^{-3} + \epsilon_o^2 a^2 \mathrm{OSR}^{-1}\right] \quad (6)$$

$$A_{0\,\mathrm{Mash}_{3\,dB}} \approx a\sqrt{7}\,\frac{\mathrm{OSR}^3}{\pi^3} \quad (7)$$

$$\epsilon_{\mathrm{Mash}_{3\,dB}} \approx \sqrt{\frac{3}{7}}\,\frac{\pi^2}{\mathrm{OSR}^2} \quad (8)$$

The result from Eq. (5) can be simplified since the dominant noise due to finite gain and capacitor matching errors is produced in the first integrator stage only [8]. This leads to the expression in Eq. (6), which relates the approximate quantization noise power of the MASH to the oversampling ratio (OSR) [8]. Recall that ϵ_{m1}

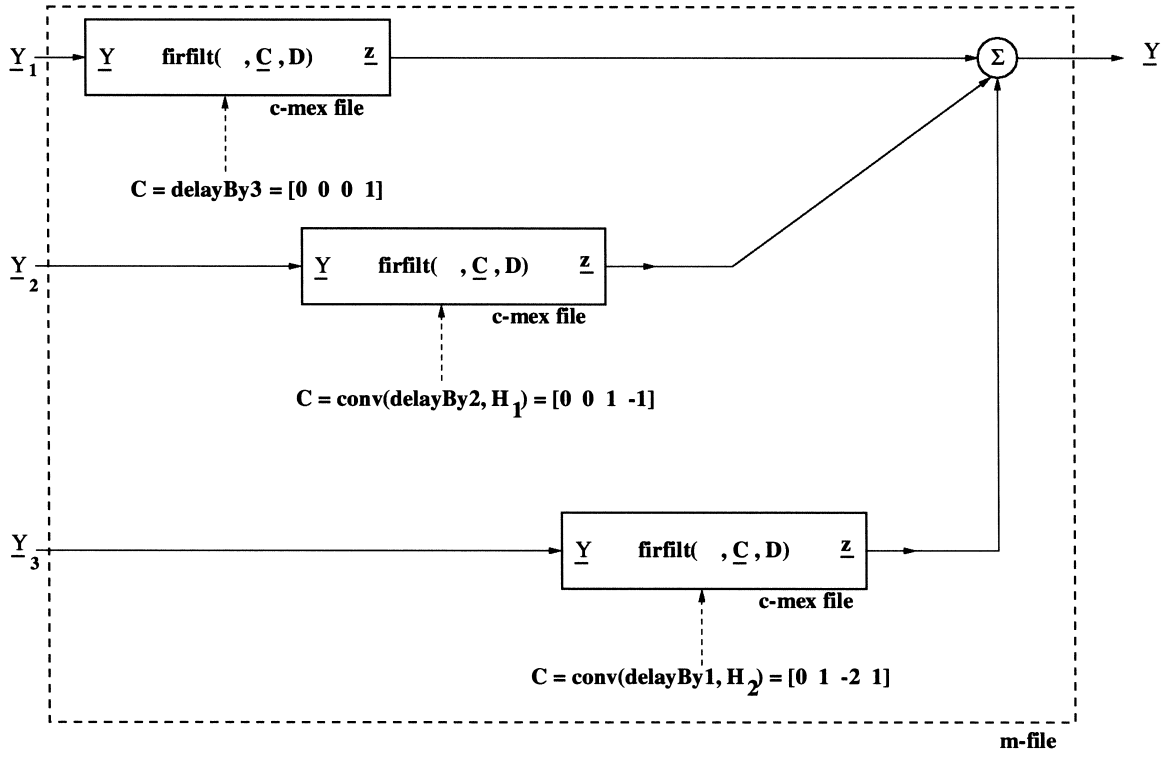


Fig. 12. Implementation of the MASH noise-cancellation filter using firfilt.mex.

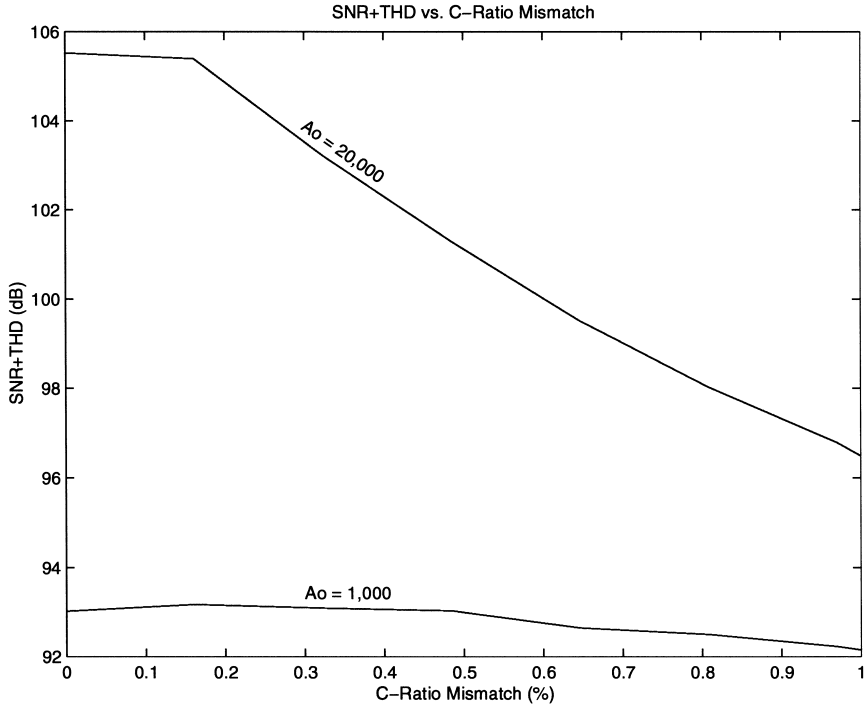


Fig. 13. SNR + THD VMS. Cap ratio mismatches between first and second MASH stages.

represents the relative cap matching error between a and a_2. The 3-dB value [2] for the amplifier gain and capacitor matching errors can then be expressed by Eqs. (7) and (8) [8]. Thus, in order for the capacitor matching errors to become negligible for an OSR of 64, $A_{0\,\mathrm{Mash}_{3\,\mathrm{dB}}}$ is $22\,000a$ from Eq. (7) [8]. Alternatively, if the amplifier gain is infinite, $\epsilon_{\mathrm{m3\ dB}}$ has to be less than 0.16% (e.g., Eq. (8)).

The major problem in the design of a MASH modulator is to quantitatively determine the actual SNR + THD for a nonideal amplifier and capacitor ratio matching errors. Although some expressions easily can be derived, they are based upon a linear model of the quantizer and are limited to specific cases. The only practical means of analyzing the performance is to simulate the MASH network over a range of amplifier gain value and C-ratio errors. The analog, switched capacitor portion of the MASH and the digital noise canceller are simulated by the m-files depicted in Figs. 11 and 12.

Notice from these two figures that the amplifier gain for each stage is controlled by the sdMod.mex input parameter, A_{0_i}, and the capacitor ratios of interest, a and b, are given values different from a_2 and b_2 within the m-file. Capacitor ratio mismatches are easy to simulate. Either a or $1/a_2$ can be varied to simulate a mismatch between the two capacitor ratios.

The plot in Fig. 13 demonstrates how DelSi is used in this analysis. The capacitor ratio error has been continuously increased from 0 to 1% while the op-amp open-loop gain has been maintained at 1000 and 20 000, respectively. The input signal amplitude is $0.94V_{\mathrm{ref}}$, almost full scale. The parameters employed in Fig. 13 cover a relatively wide range of gain values and C-ratio mismatch errors which characterize the possible trade-offs in conjunction with the MASH topology. Each curve in Fig. 13 requires about 50 to 100 simulations (and FFTs) to estimate the corresponding SNR + THD values.

[2] This value doubles the quantization noise (3 dB reduction in SNR).

4.2. Fifth-order iflf modulator

In the design of higher-order modulators, such as the fifth-order iflf topology (also known as a cascade of resonators [3,4]) shown in Fig. 14, the selection of coefficients will require a great deal of care since these network structures are only conditionally stable [4,12]. To further improve the noise shaping property, the modulator places two finite zeros realized by the additional feedback loops denoted as δ_1/g_3 and δ_2/g_5, respectively. The synthesis tools in DelSi allow the zero placement to be optimized for either the ∞-norm or the 2-norm. In most cases, we wish to minimize noise power [13]. Thus, the 2-norm is the method of choice.

Once the zeros are chosen, the NTF design tool, ntfdes.m, places the poles for a specified NTF gain. In the iflf5, as with most higher order delta–sigma modulators, the signal transfer function (STF) peaks outside the signal pass band. This occurs because the STF shares the poles of the NTF, which are optimized for quantization noise attenuation. Additional signal feed-in paths can be used to place zeros in the STF. In Fig. 14, one additional feed-in, i.e., a_3, is used to cancel the pole. Without the zero, there exists a risk of instability if too much signal energy exists outside the pass band, that is near the pole frequency where the peak occurs.

Figs. 15 and 16 show the NTF pole–zero plot and the magnitude response of the iflf5 modulator. The designer can optimize the SNR + THD figure by increasing the (high-frequency) noise gain of the NTF. However, the higher this gain, the higher the risk of instability. Optimization algorithms for pole placement and numerical simulations are thus indispensable to determine a reasonable compromise between performance and stability in a higher order modulator.

In order to properly scale the loop coefficients, the modelling tool has to be run at least once. Scaling prevents the integrator outputs from rising above or below a certain threshold value (e.g., the analog feedback voltage V_{ref}) while maintaining the NTF poles and zeros. DelSi can carry out an automatic scaling procedure whereby the swing at each integrator output is limited by V_{ref}. Another available alternative is to interactively determine the proper scaling coefficients by carefully adjusting all (time-domain) integrator outputs while viewing their respective time series. Figs. 17 and 18 show the DelSi plots of the time series and the resulting output spectrum. This particular modulator features an OSR of 32 and is sampled at 10.24 MHz. This corresponds to a signal bandwidth of 160 kHz.

When optimizing a higher-order modulator for wide bandwidth, it is necessary to fully explore the design space with knowledge of the acceptable ranges and combinations of amplifier gain, bandwidth and slewing. In the iflf5 case (and most other single-stage topologies) the degradation in performance due to op-amp gain and capacitor errors has virtually no impact on performance. Thus, one only needs to investigate the effects of bandwidth and slew rate [8]. Since its impractical to expect an op-amp to meet an exact settling time requirement, a range of acceptable combinations, hence a ‘profile’, of the op-amp requirements are provided. Fig. 19 shows such an amplifier profile plot displaying the SNR + THD vs. bandwidth for various slew rates values.

In order to obtain meaningful results regarding amplifier bandwidth and slew-rate values, one has to know the maximum signal swing and the effective load capacitance of each amplifier. By default, sdMod.mex

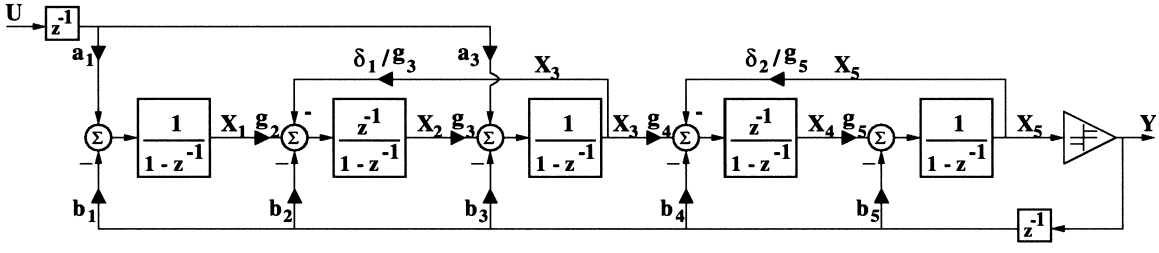


Fig. 14. Block diagram of fifth-order iflf modulator.

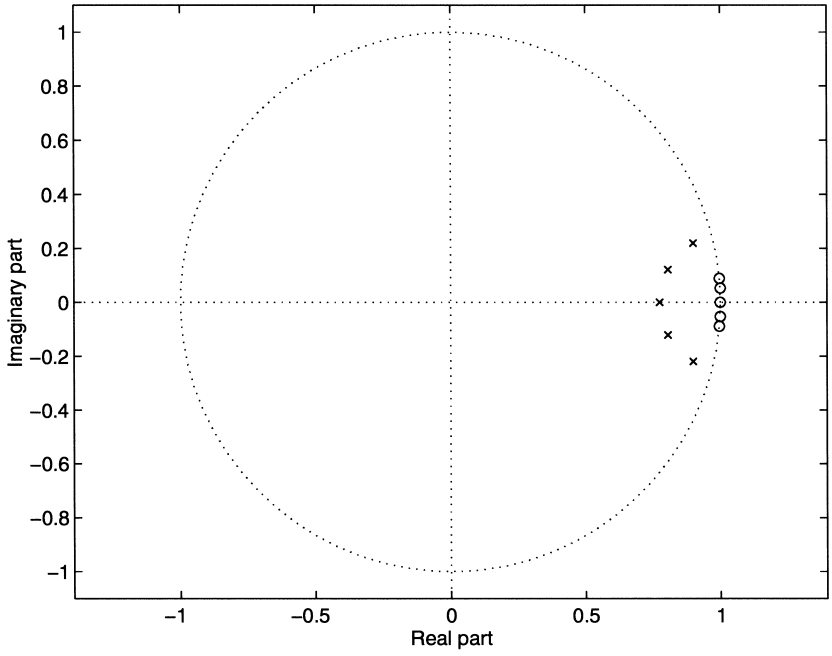


Fig. 15. Pole–Zero plot of fifth-order NTF.

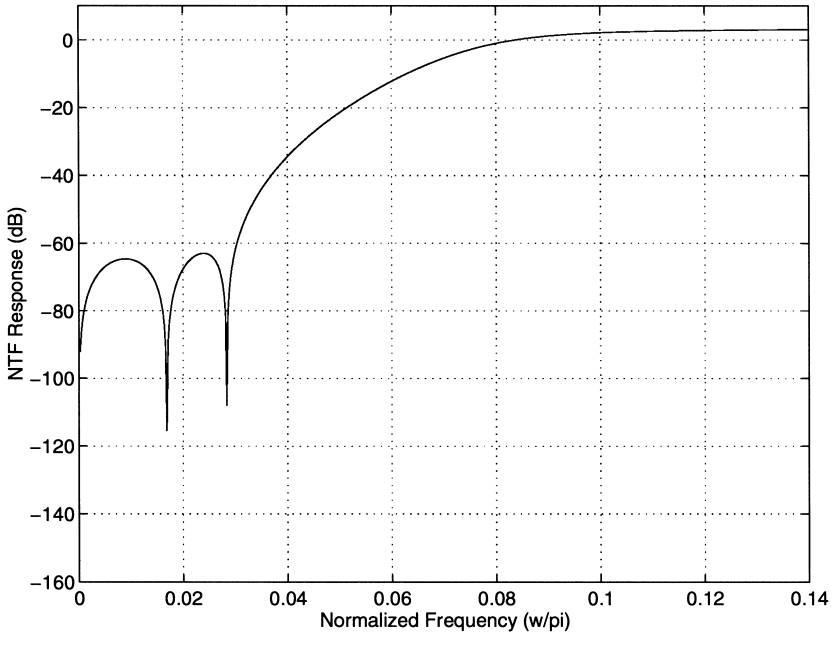


Fig. 16. Magnitude response of the fifth-order NTF.

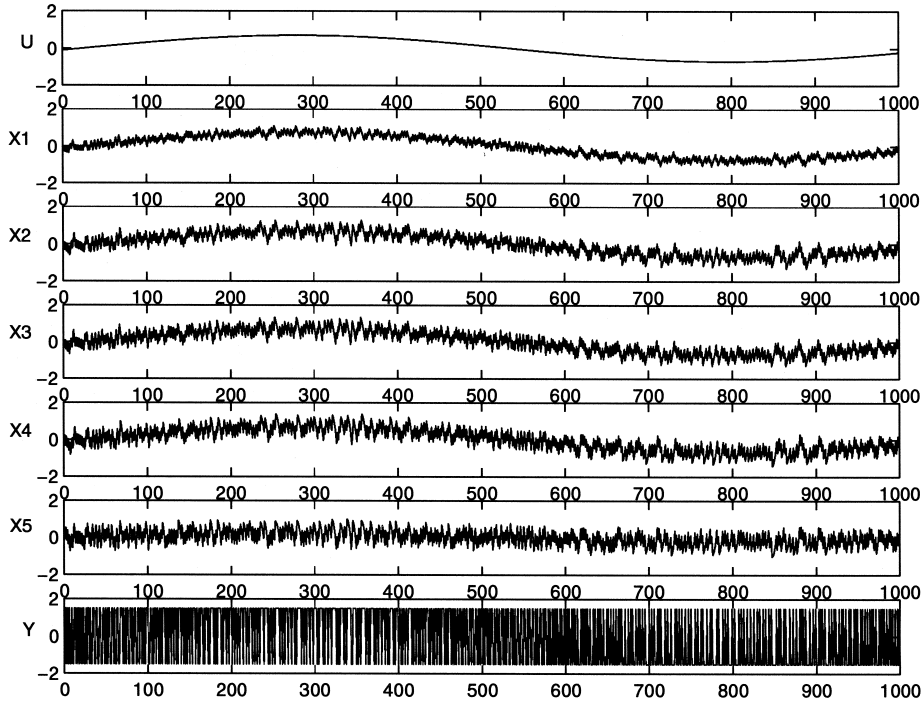


Fig. 17. Time series outputs for fifth-order iflf network.

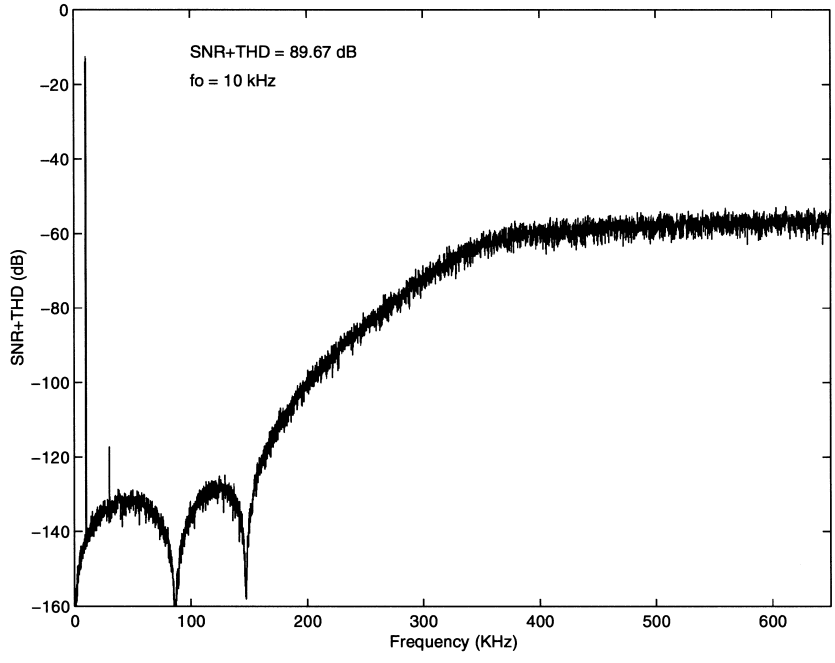


Fig. 18. Ideal modulator output spectrum for the fifth-order iflf modulator.

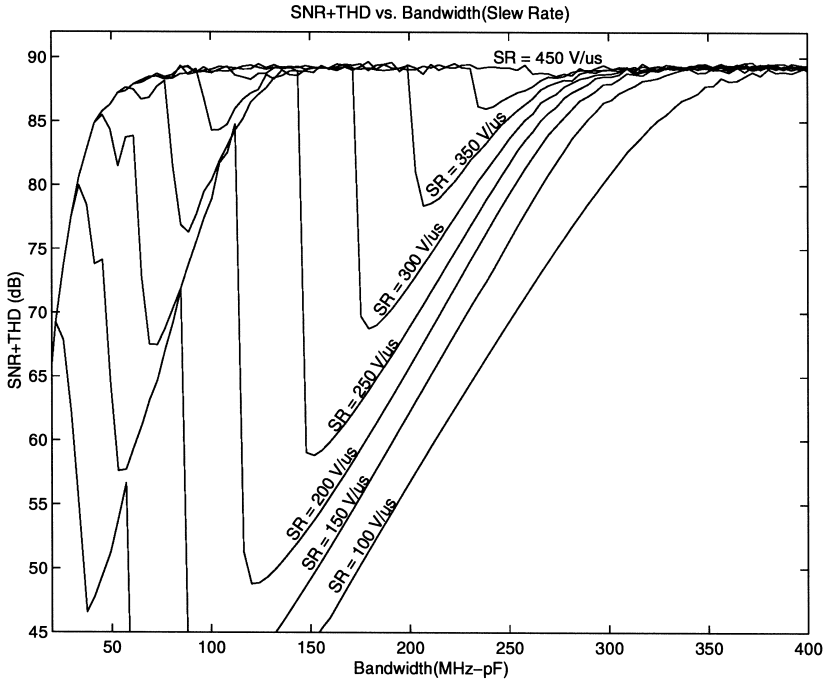


Fig. 19. Op-amp characterization based on DelSi analysis of fifth-order.

assumes a minimum integrator input capacitance of 1 pF. In the depicted example, the reference voltage, V_{ref}, has been chosen to be 1.5 V and the input signal swing has been fixed at 0.70 V, or 46% of V_{ref}.

5. Conclusions

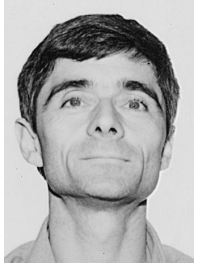
An overview of behavioral modelling and the design of delta–sigma modulators has been presented together with the DelSi toolbox. DelSi provides an experienced analog designer with a seamless path to move from the high-level network structure to a detailed circuit design. The simulation and analysis tools handle nonidealities in a very practical manner since profiling the behavior of the op-amps provides a range of acceptable values for gain, bandwidth and slew-rate requirements. By specifying a range for each parameter (e.g., an op-amp 'profile') rather than a single number, the first-order approximations provided by the behavioral model will yield information critical to obtaining a circuit which functions reliably in silicon.

The modelling tool is implemented in a unique, object oriented fashion. The implementation choices for the basic building blocks (e.g., as an m-file or mex-program) are especially important if one wishes to simulate any type of cascaded delta–sigma modulator without the burden of writing custom software for each structure. Although it is quite simple, firfilt.mex provides a very efficient primitive function which allows one to construct almost any noise-cancellation filter. This turns out to be useful for both simulation and measurement purposes. Typically, it is less troublesome to store the quantizer outputs from each stage vs. storing the multibit noise-cancelled outputs. We actually use the noise-canceller in DelSi in the spectrum analysis of cascaded modulator ICs.

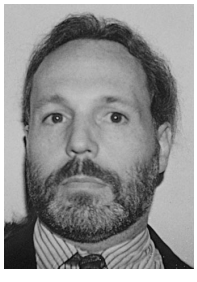
The work on DelSi is ongoing and the toolbox will continue to evolve. Future projects include a graphical user interface, a netlist output for schematic capture tools and more realistic models for the digital-to-analog converter.

References

[1] Application Program Interface Guide, Version 5, The Mathworks, 1996.
[2] Using MATLAB, The MathWorks, December 1996.
[3] R.W. Adams, The design of high-order single-bit Σ-Δ ADCs, in: Norsworthy et al. [12], Chap. 5, pp. 165–192.
[4] R.W. Adams, P.F. Fergusson Jr., A. Ganesan, S. Vincelette, A. Volpe, R.J. Libert, Theory and practical implementation of a fifth-order delta–sigma a/d converter, J. Audio Eng. Soc. 39 (1991) 515–528.
[5] C.T. Chuang, Analysis of the settling behavior of an operational amplifier, IEEE J. Solid-State Circuits SC 17 (1982) 74–80.
[6] A.J. Davis, G. Fischer, A mash modulator with digital correction for amplifier finite gain and C-ratio matching errors, in Proceedings of the 40th MWSCAS, Vol. I, Sacramento, CA, 3–6 August 1997, to be published.
[7] A.J. Davis, G. Fischer, Digital correction of non-ideal amplifier effects in the mash modulator, in Proceedings of the ISCAS'98, Monterey, CA., 31 May - 3 June 1998, pp. 600–603.
[8] G. Fischer, A.J. Davis, Alternative topologies for sigma–delta modulators: a comparative study, IEEE Transactions on Circuits and Systems: Part II. Analog and Digital Signal Processing, 44, 1997, pp. 789–797.
[9] K. Martin, A.S. Sedra, Effects of the op-amp finite gain and bandwidth on the performance of switched-capacitor filters, IEEE Trans. Circuits System CAS 28 (1981) 822–829.
[10] Y. Matsuya, K. Uchimura, A. Iwata, T. Kobayashi, M. Ishikawa, T. Yoshitome, A 16-bit oversampling a/d conversion technology using triple integration noise shaping, IEEE J. Solid-State Circuits SC 22 (1987) 921–929.
[11] S.R. Norsworthy, I.G. Post, H.S. Fetterman, A 14-bit 80-kHz sigma–delta A/D converter: modelling, design and performance evaluation, IEEE J. Solid-State Circuits SC 24 (1989) 256–266.
[12] S.R. Norsworthy, R. Schreier, G.C. Temes (Eds.), Delta–Sigma Data Converters: Theory, Design and Simulation, IEEE Press, 1997.
[13] R. Schreier, An empirical study of high-order single-bit delta–sigma modulators, IEEE Trans. Circuits System Part II CAS 40 (1993) 461–466.
[14] R. Schreier, M. Goodson, B. Zhang, An algorithm for computing convex positively invariant sets for delta–sigma modulators, IEEE Transactions on Circuits Systems: I. Fundamental Theory Applications 44 (1997) 38–44.
[15] S.K. Tewksbury, R.M. Hallock, Oversampled, linear predictive and noise-shaping coders of order $n > 1$, IEEE Trans. Circuits Systems CAS 25 (1978) 436–447.
[16] L.S. Thomas, P. Krauss, J.N. Little, Signal Processing Toolbox, The MathWorks, 1994.
[17] D.R. Welland, B.P. Del Signore, E.J. Swanson, T. Tanaka, K. Hamashita, S. Hara, K. Takasuka, A stereo 16-bit delta sigma a/d converter for digital audio, J. Audio Eng. Soc. 3 (1989) 476–486.
[18] L.A. Williams III, B.A. Wooley, A third-order sigma–delta modulator with extended dynamic range, IEEE J. Solid-State Circuits 29 (1994) 193–202.
[19] C. Wolff, J.G. Kenny, L.R. Carley, CAD for the Analysis and Design of $\Sigma - \Delta$ Converters, in: Norsworthy et al. [12], Chap. 14, pp. 447–467.



Alan J. Davis (S'94) was born in New Bedford, MA, on March 3, 1960. He received the BS degree in Physics and Electrical Engineering and the MS degree in Electrical Engineering from the University of Massachusetts, Dartmouth, in 1983 and 1985, respectively. He is currently completing his PhD in Electrical Engineering at the University of Rhode Island, Kingston, RI. Since 1983, he has been employed by the Naval Undersea Warfare Center (NUWC) in Newport, RI. Until 1993, he was a senior design engineer with the Weapon Technology and Undersea Physics Division, where he designed special-purpose computers for undersea vehicles. He is currently a lead design engineer with the Torpedo Systems Division developing prototype receivers for high resolution sonars. His research interests are sigma–delta modulation, advanced mathematics relating to discrete-time signal processing, and mixed-signal integrated circuits for high-frequency sonars. Mr. Davis is a member of Eta Kappa Nu and Sigma Pi Sigma.



Godi Fischer (M'85) was born in Romanshorn, Switzerland, on September 9, 1953. He received the Dipl. Ing. and the PhD degree in electrical engineering from the Swiss Federal Institute of Technology (ETH), Zürich, Switzerland, in 1978 and 1985, respectively. Between 1978 and 1981, he was a teaching assistant, and from 1981–1985 a research assistant and PhD candidate at the Institute of Signal and Information Processing, ETH Zürich. Since 1985, Dr. Fischer is a professor in the Department of Electrical Computer Engineering at the University of Rhode Island, Kingston, RI. His main research interests include analog and digital filter design, high-resolution data converters (delta–sigma modulators) and low-power CMOS circuits.