

# A SIMULINK-BASED APPROACH FOR FAST AND PRECISE SIMULATION OF SWITCHED-CAPACITOR, SWITCHED-CURRENT AND CONTINUOUS-TIME $\Sigma\Delta$ MODULATORS

Javier Moreno-Reina, José M. de la Rosa, Fernando Medeiro, Rafael Romay, Rocío del Río, Belén Pérez-Verdú  
and Angel Rodríguez-Vázquez

Instituto de Microelectrónica de Sevilla, IMSE-CNM (CSIC)  
Edif. CNM-CICA, Avda. Reina Mercedes s/n, 41012 Sevilla, SPAIN  
Phone: +34 95056666, FAX: +34 95056686, E-mail: jrosa@imse.cnm.es

## ABSTRACT

This paper describes how to extend the capabilities of SIMULINK for the time-domain simulation of  $\Sigma\Delta$  modulators implemented by using switched-capacitor, switched-current and continuous-time circuits, considering the most important error mechanisms. The behavioural models of these circuits are incorporated into the SIMULINK environment by using C-language *S*-function blocks, which leads to a drastic saving in the simulation time as compared to previous approaches based on MATLAB functions. The outcome is a complete SIMULINK block library that allows interactive, fast and accurate simulation of an arbitrary  $\Sigma\Delta$  topology (\*).

## 1. INTRODUCTION

Simulation is a critical part of both the top-down synthesis and the bottom-up verification of Integrated Circuits (ICs). Thus, the iterative use of simulators helps designers to explore the design space and to optimize critical trade-offs at different hierarchical levels [1]. In the case of  $\Sigma\Delta$  Modulators ( $\Sigma\Delta$ Ms), as a consequence of their sampled-data nature, simulation has to be done in the time-domain. However, transistor-level simulations with SPICE-like simulators yield to excessively long CPU times – typically several days, or even weeks. The reason is that several thousands clock cycles – with small numerical integration steps and complex models– are needed to obtain a realistic evaluation [2].

To overcome this problem, different alternatives for the simulation of  $\Sigma\Delta$ Ms have been proposed, which at the price of losing accuracy in their models, reduce the simulation time [2][3][4]. One of the best accuracy-speed trade-off is achieved by using the so-called *behavioural simulation* technique [1]. In this approach the modulator is broken up into a set of subcircuits, often called building blocks, which are described by explicit equations that relate the outputs in terms of the inputs and the internal state variables. Thus, the accuracy of the simulation depends on how precisely those equations describe the real behaviour of each block.

In case of Discrete-Time (DT)  $\Sigma\Delta$ Ms implemented with either *switched-capacitor* (SC) [3] or *switched-current* (SI) circuits [5], the value of signals is important only at specific time points. Therefore, each building block is defined by a set of finite difference equations which describe its functionality, and the simulation process consists of computing the node voltages and branch currents of the circuit consecutively in each clock phase. The outcome is a drastic saving in CPU time – only a few seconds to evaluate an output spectrum. Recently, behavioural simulation has been applied also to Continuous-Time (CT)  $\Sigma\Delta$ Ms [6]. In this case, model equations are computed analytically instead of numerically, leading to CPU times comparable with the DT case.

(\*)This work has been supported by the EU ESPRIT IST Project 2001-34283/TAMES-2 and the Spanish CICYT Project TIC2001-0929/ADAVERE.

In spite of their good trade-off between precision and CPU-time, previously reported behavioural simulators present several drawbacks. On the one hand, there is a limited number of  $\Sigma\Delta$  topologies that can be simulated, normally using only one circuit technique. On the other hand, except for [4], the user interface consists of an input netlist with a dedicated syntax, while postprocessing is performed by using commercial tools like MATLAB [7].

The above-mentioned problems can be overcome by implementing the behavioural models in the SIMULINK environment [8]. The benefits are a friendly Graphical User Interface (GUI), high flexibility for the extension of the block library and huge signal processing capabilities. Recently, a set of SIMULINK block models has been proposed for the behavioural simulation of SC  $\Sigma\Delta$ Ms [9]. However, it has two major constraints:

- The block library is limited to SC circuits, using simple models which do not include some important limitations like mismatch and the non-linearities associated to the open-loop opamp DC gain and capacitors. In addition, as models are implemented in the *Z*-domain, the circuit behaviour during different clock phases is not considered, thus leading to an imprecise modelling of some errors like the incomplete settling.
- Block models are realized by using MATLAB functions. This causes the MATLAB interpreter to be called at each time step, slowing down the simulation time drastically [8]. This problem is aggravated as the model complexity increases, yielding to excessive CPU times as compared to C-written simulators. This is true even using the SIMULINK accelerator [8].

This paper presents an interactive and flexible approach for a fast time-domain behavioural simulation of LowPass (LP) and Band-Pass (BP)  $\Sigma\Delta$ Ms implemented by using not only SC, but also SI and CT circuits. In order to speed up the simulation,  $\Sigma\Delta$ -blocks are incorporated in SIMULINK<sup>†1</sup> as C-coded *S*-functions [10]. As a result the CPU-time for one 65536-point simulation of a DT/CT  $\Sigma\Delta$  is typically less than 5 seconds<sup>†2</sup>, meaning only a few times slower than C-written simulators, but up to 2 orders of magnitude faster than using MATLAB functions as in [9].

## 2. DESCRIPTION OF THE $\Sigma\Delta$ -BLOCK LIBRARY

The proposed SIMULINK  $\Sigma\Delta$ -block library includes different sublibraries which are classified according to the modulator hierarchy level and the circuit technique. As an illustration, Fig.1 shows some of these sublibraries showing: SI memory cells, SC/CT integrators and resonators (used in BP- $\Sigma\Delta$ Ms), quantizers,

<sup>†1</sup>. SIMULINK 5 and MATLAB 6.5 (release 13) were used.

<sup>†2</sup>. All simulations shown in this paper were done using an Intel Pentium 4 CPU@1.7GHz @256MB RAM PC.

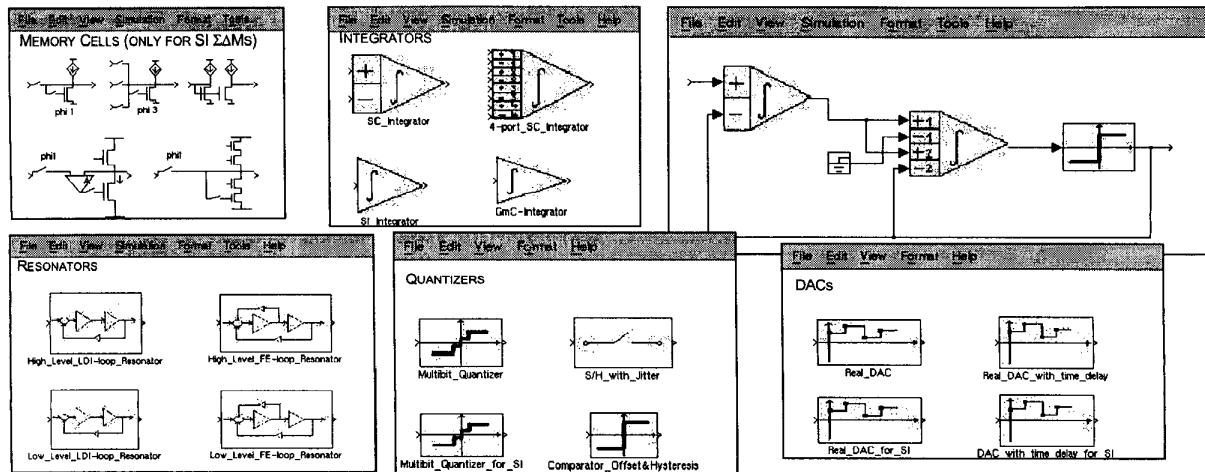


Figure 1. Illustrating some blocks of the proposed SIMULINK  $\Sigma\Delta$ -block library.

and both 1-bit and multi-bit (*mb*) Digital-to-Analog Converters (DACs). There is also a sublibrary including the most usual architectures of both LP- and BP- $\Sigma\Delta$ Ms using SC, SI and CT circuits. For each building block, the  $\Sigma\Delta$ -block library provides models with a different abstraction level. The purpose is twofold. First, high level models are suited for system level simulations and initial transmission of specifications. Second, low level accurate models, which takes into account main circuit parasitics, are suited for fine-tuning the specs transmission and circuit validation.

The main circuit non-idealities included in the integrators (and resonators) are:

- *SC circuits*: finite open-loop opamp DC gain, incomplete settling error, mismatch capacitor ratio error, thermal noise; and main non-linear effects, namely: non-linear sampling switch-on resistance, non-linear open-loop opamp DC gain, slew rate and non-linear capacitors.
- *CT circuits*: finite DC gain, integration time constant error, slew rate, finite non-linear transconductance and thermal noise.
- *SI circuits*: linear and non-linear gain error, finite output-input conductance ratio error, charge injection error, incomplete settling error, mismatch error and thermal noise.

Detailed descriptions of these errors as well as their behavioural models – beyond the scope of this paper – can be found in [3], [11] and [5] for SC, CT and SI circuits, respectively.

In addition to integrators and resonators, quantizer and DAC errors have to be considered, specially in SC/SI cascade *mb* architectures and CT single-loop topologies. For this purpose, the following circuit parasitics have been included:

- *quantizers*: offset, both deterministic and random hysteresis, and in *mb* realizations, gain error and integral non-linearity.
- *single-bit and multi-bit DACs*: offset, gain error and integral non-linearity. In case of CT  $\Sigma\Delta$ Ms, a time delay is also included in order to simulate the effect of *excess loop delay* [11].

The behavioural models of the above-mentioned errors have been coded in C language, and incorporated into the SIMULINK environment through the so-called *S*-functions [10]. These are special purpose C source files which allow us to add C algorithms to

SIMULINK models. The outcome is a notable saving of simulation time as compared to use MATLAB functions or M-files to code the models, even when the accelerator utility is used [8].

In order to create an *S*-function associated to building blocks like those shown in Fig.1, the following steps have to be followed:

- *Create a C-coded S-function containing the behavioural model.* For this purpose, SIMULINK provides different *S*-function templates which can accommodate the C-coded model of both DT and CT systems. These templates are composed of several routines that perform different tasks required at each simulation stage [10]. Among the others, these tasks include: variable initialization, computing output variables, updating state variables, etc. Thus, programmers' work simply consists in placing the C-coded behavioural model in the different parts of the template file. For illustration purposes, Fig. 2(a) shows the behavioural modeling and some significant sections of the *S*-function file associated to an SC integrator with non-linear opamp DC gain – not included in [9]. It includes model parameters, clock phase diagram, model code, etc.
- *Compiling the C MEX-file S-function.* This is done by using the *mex* utility provided by MATLAB [10]. The resulting object files are dynamically linked into SIMULINK when needed.
- *Incorporating the model into the SIMULINK environment.* This is done by using the *S*-function block of the SIMULINK libraries [8]. Fig.2(b) illustrates this process for the SC integrator of Fig.2(a). A block diagram containing the *S*-function block is created including the input/output pins. The dialogue box is used to specify the name of the underlying *S*-function – in this case *intfescavn1*. In addition, model parameters are also included in this box. In order to facilitate the use of building blocks, the user can insert the values of model parameters from a dialogue box associated to the block.

### 3. SIMULATION EXAMPLES

An arbitrary modulator architecture can be defined by connecting the building blocks available in the  $\Sigma\Delta$ -block library. This can be done by using the SIMULINK Library browser as usual. Alternatively, the  $\Sigma\Delta$ -block library can be browsed by using a dedi-

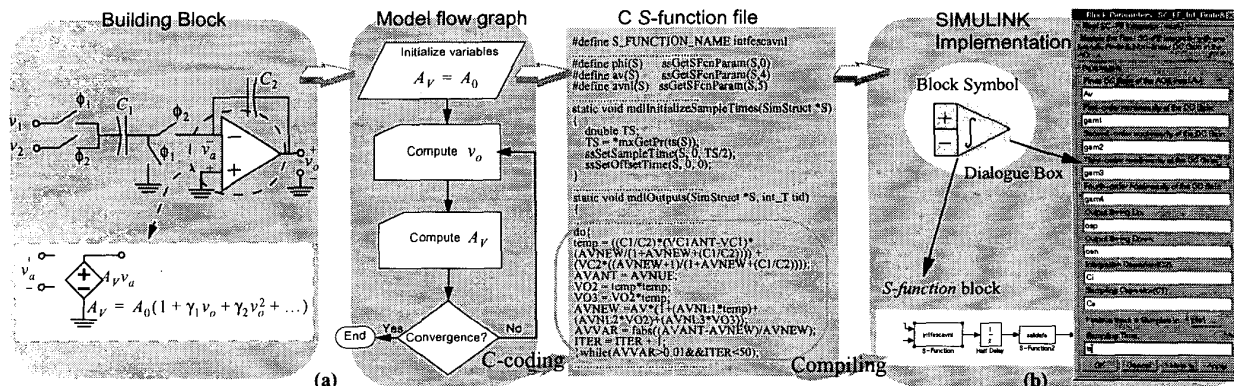


Figure 2. Basic steps to incorporate a behavioural model in the  $\Sigma\Delta$ M-block library: (a) S-function file. (b) S-function block.

cated GUI that allows the user to navigate in an easy way through all the steps of the simulation and post-processing of results.

As an illustration of the capabilities of the  $\Sigma\Delta$ M-block library, this section shows the impact of some circuit parasitics on the performance of the following modulator architectures:

- A CT (Gm-C) 2nd-order LP- $\Sigma\Delta$ M (CT 2nd-LP $\Sigma\Delta$ M).
- A SI 4th-order BP- $\Sigma\Delta$ M (SI 4th-BP $\Sigma\Delta$ M).
- A SC 2-1-1 cascade  $mb$  (3b)  $\Sigma\Delta$ M (SC 2-1<sup>2</sup> $mb$ ).

Fig.3 shows the block diagram of these architectures in the SIMULINK environment, including building blocks from the  $\Sigma\Delta$ M-block library.

### 3.1 CT 2nd-LP $\Sigma\Delta$ M example

In the example shown in Fig.3(a), an ideal 1-bit quantizer was used while the other blocks include the following parameters:

- *Gm-C integrators*: finite DC gain, time-constant error, unity

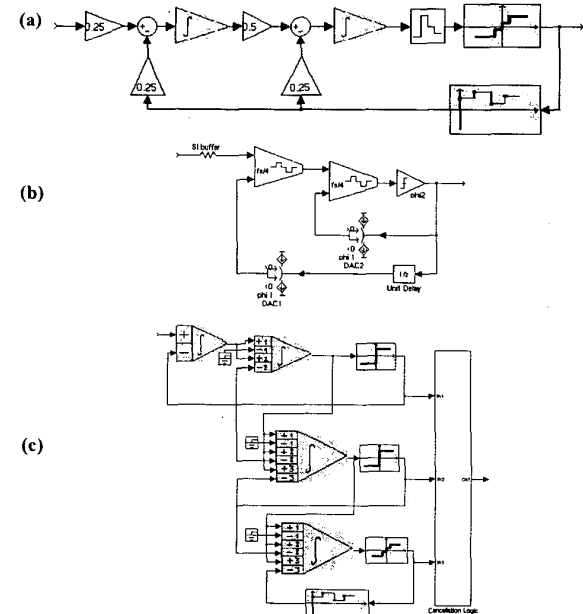


Figure 3. Block diagram of the  $\Sigma\Delta$ M-library examples. (a) CT 2nd-LP $\Sigma\Delta$ M. (b) SI 4th-BP $\Sigma\Delta$ M. (c) SC 2-1<sup>2</sup> $mb$ .

- gain frequency, slew rate, temperature and output-swing.
- *DAC*: reference voltage and time delay.

In high-speed applications, the performance of the modulator can be severely degraded by finite bandwidth and slew rate. These errors cause an increase of both the in-band noise power and the harmonic distortion. This is illustrated in the output spectrum of Fig.4(a), obtained by performing an Hanning-windowed 65536-point FFT to the output bit stream of Fig.3(a), with a half-scale@10-kHz input tone, when clocked at 20 MHz. This simulation takes 3 seconds when the accelerator is used.

In addition to integrator dynamics, one of the most important limiting factors arising in CT- $\Sigma\Delta$ Ms is the time delay between the quantizer clock edge and DAC response. This delay, referred to as *excess loop delay*, modifies the noise-shaping transfer functions, and may eventually make CT- $\Sigma\Delta$ Ms unstable [11]. Mathematically speaking, a complex analysis would be required to obtain the stability condition that relates the loop delay,  $\tau_d$ , with the clock

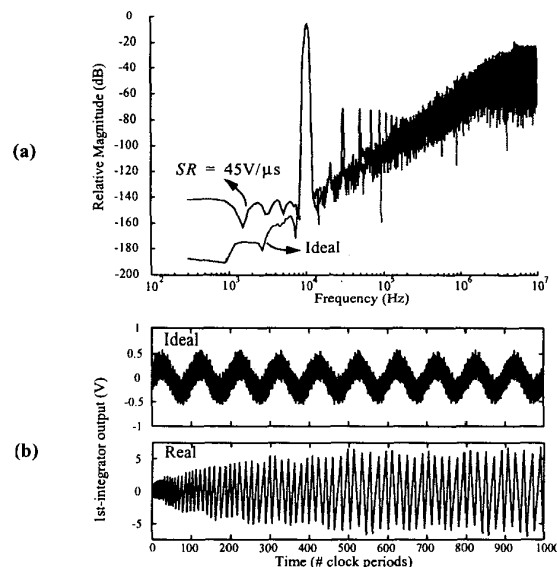


Figure 4. Performance degradation of a CT 2nd-LP $\Sigma\Delta$ M. (a) Harmonic distortion caused by slew rate. (b) Effect of excess loop delay on the transient response of the first integrator.

period,  $T_S$ . Instead of that, simulation-based analyses are normally used. As an illustration, Fig.4(b) shows the first integrator output waveform for different values of the DAC time delay, showing unstable behaviour for  $\tau_d \approx 3T_S/2$ .

### 3.2 SI 4th-BPΣΔM example

The SI 4th-BPΣΔM shown in Fig.3(b) has been obtained by applying a LP-to-BP transformation ( $z^{-1} \rightarrow -z^{-2}$ ) to a 2nd-LPΣΔM. As a consequence of this transformation, the zeroes of the noise transfer function shift from DC to a quarter of the sampling frequency,  $f_S$ . In addition, the integrators in the original LP-ΣΔM become resonators. In this example, resonators are based on lossless direct integrators. Note that a front-end block, named SI buffer, is used to model the voltage-to-current conversion.

One of the most important degrading factors in SI BP-ΣΔMs is the signal-dependent transconductance of memory cells,  $g_m$ , which force all errors to be non-linear. As a consequence, in addition to increase the in-band noise power, SI errors cause InterModulation Distortion (IMD). As an illustration, Fig.5 shows the impact of the non-linear settling on the performance of the modulator in Fig.3(b). In this case, the gate-source capacitance of memory transistors,  $C_{gs}$ , is varied showing three effects: increase of the in-band noise, third-order IMD and a shift of the quantization noise-filtering notch frequency,  $\delta f_n$ . These output spectra are obtained by running two 65536-point simulations of Fig.3(b), each one taking 4 seconds.

### 3.3 SC 2-1<sup>2</sup>mb example

In the case of SC ΣΔMs, the behavioural models of building blocks have been translated from ASIDES, a C-coded time-domain behavioural simulator for SC ΣΔMs [3]. As a consequence of this translation, simulation results obtained with both SIMULINK and ASIDES are practically identical. Compared to ASIDES, the proposed SIMULINK ΣΔ-block library offers a friendly user interface and a great flexibility to simulate an arbitrary SC filter topology, specially but not only, dedicated to ΣΔMs. However, there is a minor CPU-time penalty due to the SIMULINK interface. For instance, a 65536-point simulation of the modulator in Fig.3(c) including the most complex models for building blocks takes 2 seconds using ASIDES and 5 seconds using the ΣΔM-block library. This CPU-time increases up to 415 seconds if M-file building blocks are used – which means about 2 orders of magnitude slower than the approach in this paper.

As an illustration, Fig.6 shows the performance degradation of the 2-1<sup>2</sup>mb modulator in Fig.3(c) caused by two error mechanisms:

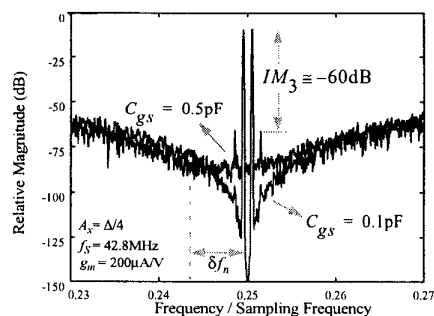


Figure 5. Effect of non-linear settling on SI BP-ΣΔMs.

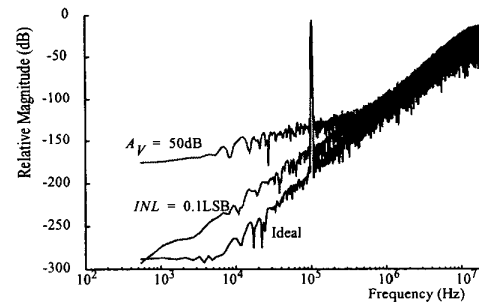


Figure 6. Degradation of an SC 2-1<sup>2</sup>mb ΣΔM with  $INL$  and  $A_V$ .

the Integral Non-Linearity ( $INL$ ) of the 3-bit DAC and the finite DC gain of the opamps,  $A_V$ . Both the isolated and the combined effects of these two error mechanisms on the output spectrum are shown in Fig. 6 when the modulator is clocked at  $f_S = 35.2\text{MHz}$  for  $INL = 0.1\text{LSB}$  and  $A_V = 50\text{dB}$ . In this case, as the  $INL$  error is shaped by the filtering performed by previous stages, the main degradation is caused by  $A_V$ , basically increasing the quantization noise power in the signal band.

## CONCLUSIONS

A complete SIMULINK block library intended for fast and interactive simulation of SC, SI and CT ΣΔMs has been described. The behavioural models of building blocks, including main circuit parasitics, have been incorporated as SIMULINK C-coded S-functions. The combination of high accuracy, short CPU-time and interoperability of different circuit models, make the block-library into a valuable instrument to optimize the design of ΣΔ analog-to-digital converters using MATLAB.

## REFERENCES

- [1] G.G. E. Gielen and R.A. Rutenbar: "Computer-Aided Design of Analog and Mixed-Signal Integrated Circuits", *Proceedings of the IEEE*, Vol. 88, pp. 1825-1852, December 2000.
- [2] V. F. Dias, V. Liberali and F. Maloberti: "Design Tools for Oversampling Data Converters: Needs and Solutions", *Microelectronics Journal*, Vol. 23, pp. 641-650, 1992.
- [3] F. Medeiro, B. Pérez-Verdú, and A. Rodríguez-Vázquez: *Top-Down Design of High-Performance Modulators*, Kluwer Academic Publishers, 1999.
- [4] K. Francken, P. Vancorenland and G. Gielen: "DAISY: A Simulation-Based High-Level Synthesis Tool for ΔΣ Modulators", *Proc. of 2000 IEEE International Conference on Computer Aided Design (ICCAD)*, pp. 188-192.
- [5] J. M. de la Rosa, B. Pérez-Verdú and A. Rodríguez-Vázquez: *Systematic Design of CMOS Switched-Current Bandpass Sigma-Delta Modulators for Digital Communications Chips*, Kluwer Academic Publisher, 2002.
- [6] K. Francken, M. Vogels, E. Martens and G. Gielen: "DAISY-CT: A High-Level Simulation Tool for Continuous-Time ΔΣ Modulators", *Proc. of 2002 Design, Automation and Test in Europe Conference (DATE)*, pp. 1110.
- [7] The MathWorks Inc.: "Using MATLAB Version 6.5", 2002.
- [8] The MathWorks Inc.: "Using Simulink Version 5", 2002.
- [9] S. Brigati, F. Francesconi, P. Malcovati, D. Tonieto, A. Baschirotto and F. Maloberti: "Modeling Sigma-Delta Modulator Non-idealities in SIMULINK", *Proc. of 1999 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. II.384-387.
- [10] The MathWorks Inc.: "Writing S-Functions Version 5", 2002.
- [11] J. A. Cherry and W. M. Snelgrove: *Continuous-Time Delta-Sigma Modulators for High-Speed A/D Conversion: Theory, Practice and Fundamental Performance Limits*, Kluwer Academic Publishers, 2000.