Effective Modeling of CT Functions for Fast Simulations Using MATLAB-Simulink and VHDL-AMS Applied to Sigma-Delta Architectures

Philippe Benabes, Catalin Tugui

Effective Modeling of CT Functions for Fast Simulations Using MATLAB-Simulink and VHDL-AMS Applied to Sigma-Delta Architectures

Philippe BENABES, *Member*, *IEEE*, Catalin-Adrian TUGUI,
Department of Signal Processing and Electronic Systems
SUPELEC
91192 GIF-SUR-YVETTE, FRANCE
philippe.benabes@supelec.fr

***Abstract*— The design, simulation and optimization of complex continuous-time (CT) circuits like Sigma-Delta modulators require large computation times when using only transistor-level analog simulators like CADENCE Spectre or PSpice. Effective high-level system modeling should be considered in order to reduce the conception effort. However, the closed-loop architecture characteristics and technology requirements should be strictly observed on the respective models. In this work, we present a design methodology and the resulted application tools implying the extraction of CADENCE schematics for analog elements into robust macro-models for MATLAB-SIMULINK and VHDL-AMS. Upon designer's choice, the resulted macro-models can be used to implement and optimize a whole modulator in the SIMULINK object-oriented environment or the code-based analog VHDL process. Using the proposed methodology, fast simulations of a sixth-order CT Sigma-Delta modulator have been performed.**

I. INTRODUCTION

Recent CT Sigma-Delta architectures adhered to the fast-growing world of submicron and deep-submicron IC technologies. The use of well-adapted technologies along with new processes like integrating CMOS circuits with micromechanical devices (e.g. Lamb Wave Resonators) offered the possibility to obtain very good bit resolutions for high levels of integration and low power consumption [1] [2]. Still, the conception effort requires very large computation times on transistor-level simulators; thus automatic optimization methods for an entire system are out of use. This situation raises the need of finding high-level modeling techniques which can simulate fast, with comparative accuracy to the analog simulators.

Different solutions have been proposed. One consists in the extraction of efficient semiconductor devices models and migration towards descriptive languages like Verilog-AMS or VHDL-AMS [3] [4]. Still this procedure is not mature enough and the large technology diversity limits its efficiency. As an alternative, for Sigma-Delta modulators, top-down design techniques using macro-models combined with optimization processes prior to transistor-level simulations were investigated [5]. This approach is well-suited for initial design steps, but cannot guarantee the optimization of the transistor-level structure. As long as the technological limitations and dispersion imply serious degradation of a Sigma-Delta modulator performance, the macro-models to be used should incorporate technology characteristics.

In our case, macro-models of amplification functions (op-amps, Gm, transconductances, resonators) are extracted by performing a complete set of analyses (DC, AC, transient, parametric) on the analog schematics implemented on specific processes. A MATLAB - CADENCE design environment connection was realized to automate simulations [6].

Once all the interesting macro-model characteristics (e.g. gain, DC and AC transfer functions, input and output impedances, non-linearities) are extracted or calculated, the MATLAB interface will export the modules as SIMULINK blocks or VHDL-AMS behavioral models.

We present in this paper the design methodology based on the interface previously described applied to a sixth-order CT Sigma-Delta modulator. The second section describes the interconnection MATLAB/SIMULINK – CADENCE – VHDL-AMS and the model extraction algorithms. In the third section the complete methodology flow is presented for a differential current-to-current converter used in the Sigma-Delta modulator structure. The fourth section implies the high-level system modeling and simulation for the complete modulator starting from the extracted components macro-models.

II. MACRO-MODEL EXTRACTION AND ALGORITHMS

A. Macro-model extraction framework

MATLAB is used as a master tool and the analog simulator as a slave tool. The CADENCE integrated SPECTRE simulator was used. MATLAB functions were implemented in order to automatically create batch command files that will be read by the Open Command Environment for Analysis (OCEAN) interface tool. Then OCEAN starts all required simulations. For each simulation, a different working

directory is used so that all results remain available. Then MATLAB can read the simulations results using the compiled functions delivered by CADENCE via their Virtuoso Multi-Mode Simulation (MMSIM) Spectre/RF toolbox. The MATLAB application will completely automate the analog simulation process and extract s-models of the transfer functions, combine offsets, gains, nonlinearities and other user-defined criteria (e.g. normalization of the frequency band) and will synthesize the macro-model. The simulation steps and interfacing process is shown in Figure 1.

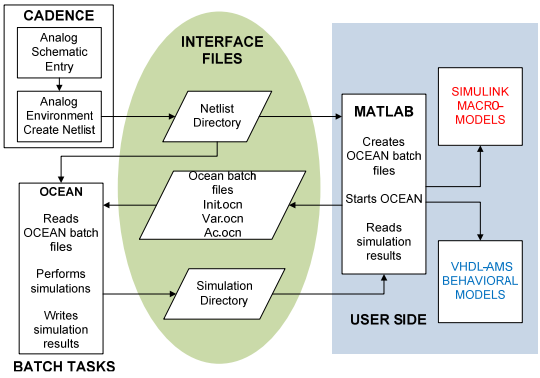

Figure 1 : Macro-model extraction framework

B. *Models types and algorithms*

As a general paradigm, each CT amplification function can be interpreted as a real dependent source of type:

- Voltage-controlled-voltage-source (VCVS);
- Voltage-controlled-current-source (VCCS);
- Current-controlled-voltage-source (CCVS);
- Current-controlled-current-source (CCCS);

This will generate 4 types of macro-models specifying the direct transfer function. Now, considering on both the input and output of the circuit the two quantities (voltage and current), a reverse transfer function, input and output impedances can be characterized. A sense convention was established for the model ports: on each side, the controlling quantity will be considered as an input port, while the controlled one as an output. Figure 2 gives an implementation of a SIMULINK macro-model of type VCVS.

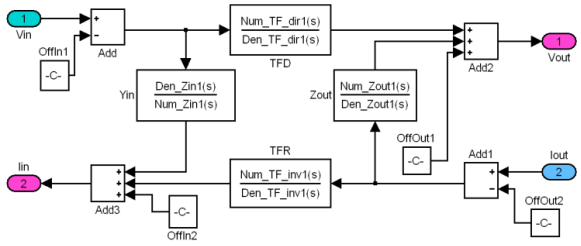

Figure 2 : VCVS SIMULINK macro-model

The direct transfer function (*TFD*), reverse transfer function (*TFR*), input admittance (*Yin*) and output impedance (*Zout*) are stable s-functions of a maximum specified order, extracted from AC analyses results using the damped Gauss-Newton method for iterative search. Considering H_n the *n*-vector of circuit responses at discrete frequencies w_n and wt_n a weighting vector with values in [0; 1], the *B* and *A* polynomials of the s-function will be extracted as:

- $B(s) = \overline{H_n}; A(s) = 1$ when $\left\langle\left\langle H_n \right\rangle\right\rangle = 0$; (1)
- *B* and *A* for $\left\langle\left\langle H_n \right\rangle\right\rangle \neq 0$ are found by minimizing the criteria:

$$c_{ord,b,a} = \min_{ord}\left(\min_{b,a}\sum_{k=1}^{n} wt[k]\left|H[k] - \frac{B(w[k])}{A(w[k])}\right|^2\right) \quad (2)$$

with *b*, *a* - the vectors of polynomial coefficients in *B*, *A* and *ord* the model order. An initial linear estimate is used.

On each input/output the offsets extracted from DC are added, while nonlinear behavior can be included on the outputs as:

$$V_{out} = V_{out\,0} + gV_{in} + Z_{out}I_{out} + c_{20}V_{in}^{\ 2} + c_{11}V_{in}I_{out} + c_{02}I_{out}^{\ 2} \quad (3)$$

$$I_{in} = I_{in\,0} + g^r I_{out} + Y_{in}V_{in} + c_{20}^r I_{out}^{\ 2} + c_{11}^r I_{out}V_{in} + c_{02}^r V_{in}^{\ 2} \quad (4)$$

The $c_{ij}^{(r)}$ factors in (3) and (4) are nonlinearity coefficients (maximum order = 3) extracted by a LMS approximation of the response area obtained from parametric DC analyses.

The same kind of macro-models can be extracted automatically as VHDL-AMS behavioral architectures.

The structures presented before are unipolar, but many designs, including Sigma-Delta architectures use differential amplifiers and filters. For this purpose, the differential input and output possibilities were added. Each of the 4 model types will define 4 sub-types corresponding to the combinations unipolar/differential on input and output. The differential structures are natural extensions of the basic models for which the differential and common-mode s-functions were converted into normal, path-associated functions. With the 16 macro-models structures one can extract all unipolar/differential circuits but also multiple-inputs-multiple-outputs designs (MIMOs), using a convenient decomposition method.

III. Differential Design Example

The model-extraction technique will be applied to the design of the sixth order CT Sigma-Delta modulator presented in [1]. The transistor-level implementation of the modulator was realized using the AMS 0.35µm BiCMOS technology.

As shown in Figure 3, its analog filter is a differential structure consisting of three transconductance amplifiers (Gm1-Gm3), four transimpedance amplifiers (Z1 to Z4), three current mirrors (M1 to M3), and LWR filters.

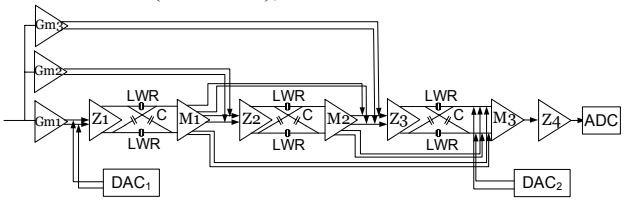

Figure 3 : Sixth-order Sigma-Delta modulator

The current mirrors are differential multi-output converters, but we will consider only one output for this example (Figure 4). Additional outputs are obtained by duplication with different sizes of the pairs M9-M10 and M4-M13. The main advantage of this structure is a good common-mode cancellation.

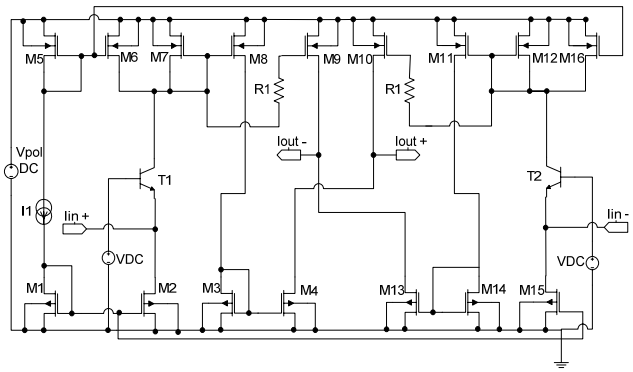

Figure 4 : Differential current mirror schematic

For macro-model extraction, the MATLAB-CADENCE interface starts common-mode and differential AC analyses on the circuit input and separate AC analyses for the outputs. This assures the minimum number of components per model (e.g. only 2 reverse transfer functions and separate output impedances), without degrading the differential behavior. The transistor-level differential/common responses will be used to recompose the 4 normal input-output responses (5). The resulted vectors are converted in the actual s-functions (6), using the MATLAB function Φ presented in II.B. (1), (2):

$$
\begin{aligned}
TFD_n^1[i] &= (TF_{comm}^{out\,1}[i] + TF_{diff}^{out\,1}[i])/2 \\
TFD_n^2[i] &= (TF_{comm}^{out\,2}[i] + TF_{diff}^{out\,2}[i])/2\,,\ i=1..n \\
TFD_n^3[i] &= (TF_{comm}^{out\,1}[i] - TF_{diff}^{out\,1}[i])/2 \\
TFD_n^4[i] &= (TF_{comm}^{out\,2}[i] - TF_{diff}^{out\,2}[i])/2
\end{aligned} \qquad (5)
$$

$$
TFD^j(s) = \Phi(TFD_n^j)\,,\ j=1..4 \qquad (6)
$$

Figure 5 presents the extracted s-functions (maximum order=2, frequency-normalized characteristics) on each path.

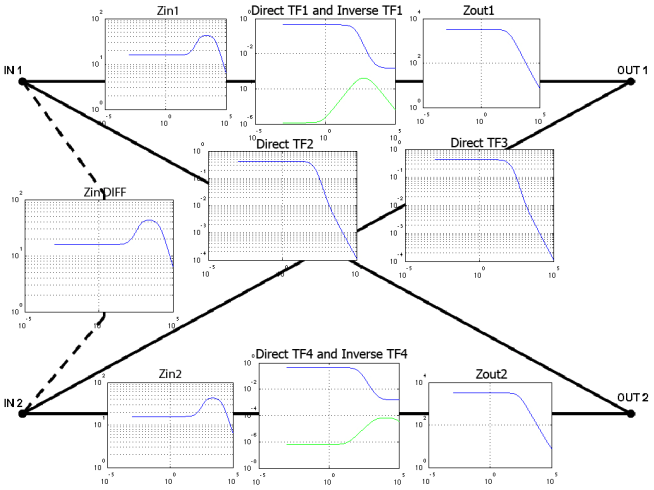

Figure 5 : Current mirror s-functions characterization

The extraction algorithm accuracy depends on the transistor-level function complexity in terms of poles and zeros versus the maximum s-model order selected. For the Sigma-Delta modulator components, we obtained a maximum total relative error normalized to the number of points in the interval [10^{-7}; 10^{-6}] for 2nd order models and [10^{-8}; 10^{-7}] for 6th order models (for the studied current mirror, 6.56e-7 for 2nd order and 2.98e-8 for 6th order).

Simple DC analyses on input and output will provide offsets extraction and DC direct/reverse gain verification, while parametric DC analyses will be used for studying the direct/reverse nonlinear effects. Also, transient impulse responses are performed to verify the design stability. Next step implies the actual SIMULINK macro-model extraction based on the previous characteristics (Figure 6).

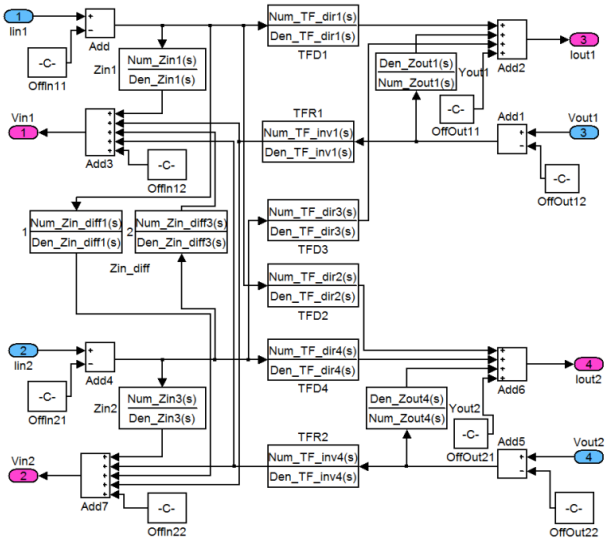

Figure 6 : Simulink macro-model for the differential design

The VHDL-AMS block for the differential current mirror is generated according to the same methodology. Due to limited space we will only exemplify the entity components and functions, while the entire model can be found at [6]:

- Input and output entity ports with the established sense convention are used for all the input/output quantities:

  PORT (QUANTITY Iin1: in current; QUANTITY Iin2: in current; QUANTITY Vin1: out voltage; QUANTITY Vin2: out voltage; QUANTITY Iout1: out current; QUANTITY Iout2: out current; QUANTITY Vout1: in voltage; QUANTITY Vout2: in voltage);

- S-functions are exported as vectors of real coefficients, e.g.

  CONSTANT Num_TF_dir1 : REAL_VECTOR := (-2.964843e-03, 1.815268e+09, -2.478997e+20);
  CONSTANT Den_TF_dir1 : REAL_VECTOR := (1.000000e+00, 4.619267e+10, 5.015811e+20);

- A linear model behavior will simply use the Laplace formalism to find output signals starting from inputs, e.g.

  Iout1 == deltaIin1'LTF(Num_TF_dir1, Den_TF_dir1) + deltaIin2'LTF(Num_TF_dir3, Den_TF_dir3) + deltaVout1'LTF(Den_Zout1, Num_Zout1) + OffOut11;
  Vin2 == deltaIin2'LTF(Num_Zin3, Den_Zin3) + deltaIin1'LTF(Num_Zin_diff1, Den_Zin_diff1) + deltaVout1'LTF(Num_TF_inv1, Den_TF_inv1) + deltaVout2'LTF(Num_TF_inv4, Den_TF_inv4) + OffIn22;

IV. HIGH-LEVEL MODELING OF THE SIGMA-DELTA MODULATOR

The sixth-order CT ΣΔ modulator considered is a 'pure-resonator' architecture [1], working at a central frequency $f_c=0.25f_s$ with an ADC sampling frequency $f_s=400MHz$.

A high-level implementation of the modulator was realized starting from the extracted macro-models of the components resulting, for example, the structure in Figure 7 for the first filter stage. The *gm1* is a unipolar to differential transconductance amplifier of gain=97µS, *TIA1a* and *TIA1b* form a differential inverted transimpedance structure, each having a gain of 1940Ω. The LWR circuit is a model for a differential resonator structure with $f_r=100MHz$. The current mirror (gain=1) has the design characteristics presented in III.

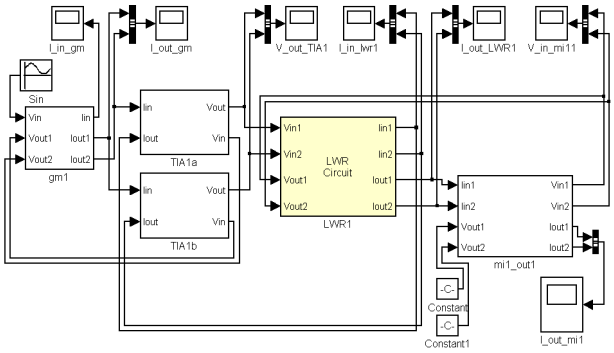

Figure 7 : First stage macro-model implementation

Parallel transient and frequency-domain simulations were conducted on the transistor-level schematic and the macro-model structure. Figure 8 gives a comparative representation of the first stage direct transfer function. The simulations results are coherent proving that even small-order block models (e.g. 2) can be used for very fast simulations with reasonable accuracy. Models of order 4-5 are very accurate for the large majority of applications but will increase the simulation time, while orders>6 will exceptionally justify the plus-performance while increasing the system complexity.

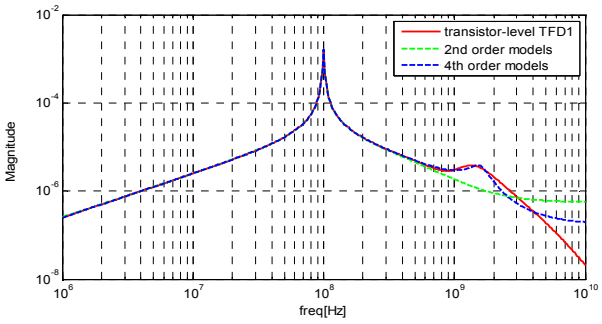

Figure 8 : First stage direct transfer function

The entire modulator was comparatively simulated with CADENCE Spectre at transistor-level and SIMULINK at macro-model level. In Figure 9, the Noise Transfer Function (NTF) of the modulator, represented as power spectral density (PSD) on the output is compared for the two cases (green: analog simulator, red: SIMULINK, order 2 macro-models). The two PSDs are coherent. Moreover, signal transfer functions (STFs) for different input signals and stability characteristics were validated.

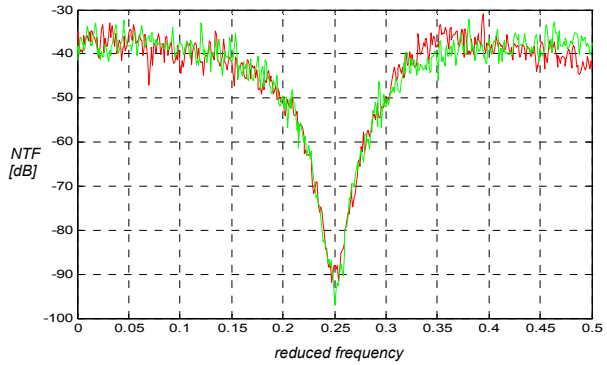

Figure 9 : NTFs of the Sigma-Delta modulator ($f_c=0.25f_s$)

Another aspect concerns the computation time needed to simulate the entire system. For 1000 output samples of the modulator, the simulation required 3h37m1s on the analog simulator, 7m8s on SIMULINK (variable-step ode15 solver used) and 6m41s on VHDL-AMS (under Dolphin Integration SMASH), implying a factor of 30x speed improvement.

V. CONCLUSIONS

A design methodology for macro-models extraction and high-level modeling of complex CT functions was proposed. In order to automate the analysis and synthesis processes, a MATLAB/SIMULINK – CADENCE – VHDL-AMS framework was developed. This application can start automatically all kinds of simulations (DC, AC, parametric, transient), and can, from these simulations, obtain macro-models of amplifying functions. The models can be exported as SIMULINK blocks or VHDL-AMS modules and used for system-level implementation. This methodology was applied to the design and simulation of a Sigma-Delta architecture, assuring a considerable speed improvement and consistent results. Even better accuracy is expected when using high-level non-linear macro-models and VHDL based simulations.

REFERENCES

[1] Javidan, M., Benabes, P., « Band-Pass Continuous-Time Delta-Sigma Modulators Employing LWR Resonators », *IEEE International Conference on Electronics, Circuits, and Systems (ICECS'08)*.

[2] Ouzounov, S. van Veldhoven, R. Bastiaansen, C. Vongehr, K. van Wegberg, R. Geelen, G. Breems, L. van Roermund, A., « A 1.2V 121-Mode CT Sigma-Delta Modulator for Wireless Receivers in 90nm CMOS », *IEEE International Solid-State Circuits Conference (ISSCC 2007)*.

[3] Mortezaee, R. Karimi, G.R. Mirzakuchaki, S., « Behavioral Modeling and Simulation of Semiconductor Devices and Circuits Using VHDL-AMS », *IEEE International Symposium on Intelligent Signal Processing, (WISP 2007)*.

[4] Kasulasrinivas, V.R. Carter, H.W., « Modeling and simulating semiconductor devices using VHDL-AMS », *Proceedings of the IEEE/ACM International Workshop on Behavioral Modeling and Simulation, 2000*.

[5] Chandra, N. Roberts, G.W., « Top-down analog design methodology using Matlab and Simulink », *IEEE International Symposium on Circuits and Systems (ISCAS 2001)*.

[6] http://www.mathworks.com/matlabcentral/fileexchange/26525