TOSCA:
A Simulator for Oversampling Converters
with Behavioural Modeling

V.F.Dias, V. Librali, F. Maloberti
Università di Pavia
Dipartimento di Elettronica
Via Abbiatiereasso, 289
27100 Pavia - Italy

Abstract - Oversampling techniques are currently used in a variety of integrated signal processors. As the result of our research activities on the design of both Sigma-Delta (ΣΔ) A/D converters and non-linear oversampled based signal processors, we have developed a software tool (TOSCA) intended to help non-specialists in the design of such systems. Time-domain behavioural models were developed for a set of basic building blocks upon which generic circuit structures can be constructed. Very long transients can be simulated in reasonable CPU time, thus allowing the calculation of useful system parameters. The simulator is fully user-friendly in the sense that the circuit description is made by means of a netlist (like SWITCAP) and the results of the distinct possible analysis are stored in text files compatible with standard graphic software tools. Simulation results show the flexibility and the potentialities of this tool.

1. INTRODUCTION

Oversampled data converters have become very popular for implementing analog-to-digital interfaces in mixed analog-digital integrated circuits [1-4]. Their main advantage comes from the possibility of exchanging resolution with the oversampling ratio allowing the use of technologies that are optimized for digital processing. Recent results demonstrated the possibility of achieving up to 18 bits of resolution while using conventional CMOS technologies [5-8].

A great limitation in the design of oversampled circuits is the lack of general-purpose simulators suitable for their analysis. Basically a simulator of oversampled circuits should verify four main conditions:

(i) as the performance of oversampled data converters comes from a statistical average of low resolution signals (for sigma-delta converters the resolution is 1 bit), very long transient analysis should be able to be simulated in reasonable CPU time;

(ii) in order to take into account the non-ideal effects of the network elements, the simulator should be able to perform accurate and realistic evaluations of the waveforms during each clock cycle;

(iii) since the circuit performance is best characterized in terms of the Signal-to-Noise Ratio (SNR) parameter, rather than the usual non-linearity parameters, the simulator should contain a specific post-processor in order to accurately evaluate the overall circuit performance;

(iv) the simulator user-interface should be simple both because generic structures should be easily simulated by simply adding new blocks or rearranging the circuit interconnections (Netlist descriptions as in standard simulators such as SWITCAP [18]), and because distinct analysis output results should be easily interfaced with usual graphic software tools. Due to its intrinsic nature conventional electrical simulators are unable to perform this kind of analysis and up until now, no dedicated software tool exists that completely fulfills all four of these conditions [9].

Here we describe a software tool, TOSCA (Tool for Oversampling Switched-Capacitor Converter Analysis) that meets all of these four main conditions. A set of basic building blocks allows the construction of generic oversampled modulator structures and digital decimators. Furthermore, a series of commands permits intensive analysis of the circuits, namely the calculation of the SNR performance, the time and frequency-domain printing of circuit nodes as well as voltage amplitude histograms. The circuit description is like in SWITCAP, i.e. it is defined by a netlist that is divided in sections for separate specification of timing, modulation, decimation and analysis commands. Analysis results are stored in text files compatible with usual graphic tools. Details on the use of the simulator can be found in the report Tosca User’s Guide [10].
2. SIMULATOR SETUP

The usual test setup of an oversampled A/D conversion system is illustrated in Fig. 1. It is composed of four main blocks: (i) a timing block, responsible for the generation of clock signals, (ii) the modulator itself that generates a sequence of low-resolution digital words, (iii) a decimation block that filters out-of-band quantization noise components and reduces the sampling rate, and finally, (iv) a post-processor that performs all the analysis necessary to characterize the system.

Fig. 1 - Test setup for an oversampled A/D converter

As can be seen from Fig. 2, the input netlist for TOSCA is organized in a similar way. It is composed of five distinct sections, namely a Title, a Timing, a Modulator, a Decimator and an Analysis section. In the following paragraphs we have given details which concern each of these sections.

Timing Section

The Timing section contains the clocking specifications of the circuit. The simulator is limited to two-phase Switched-Capacitor (SC) circuits which have a sampling period T=1/Fs. For each phase the user can specify the corresponding duty cycle, information which is used when finite characteristics such as the gain-bandwidth and the slew-rate are considered for the amplifiers.

Modulator Section

The basic elements in oversampled circuits are operational amplifiers, switches and capacitors. These elements constitute the basis for the construction of SC integrators and comparators. The most relevant non-idealities of the components which must be taken into account are the following: the finite DC-gain, the bandwidth, the slew-rate, the swing limitation, the offset voltage and the gain non-linearity of the amplifiers, the offset voltage and settling time of the comparators. In order to take into account all these effects, suitable behavioural models were developed for both the integrator and the comparator [11,12]. Other than integrators and comparators, TOSCA provides a set of additional building blocks, namely:

- sinusoidal and user-defined Signal generators
- SC integrators
- Adders
- Multipliers
- Delays
- Quantizers (the comparator is the particular case of a 1-bit quantizer)

The basic two-phase SC integrator structure considered in the simulator is illustrated in Fig. 3. It can be built-up of a number of input SC branches, each one being characterized by two input nodes (n1 and n2), an integrating phase, \( \Phi_1 \), and an integration gain k.

Fig. 2.a - Schematic diagram of a 1-bit 2nd order sigma-delta A/D converter

Fig. 2.b - TOSCA input netlist of the sigma-delta A/D converter shown in Fig. 2.a
The adder block can be used to implement both the weighted sum of analog and digital signals, as well as the multiplication of a node voltage by a constant.

**Decimator Section**

The performance of oversampled circuits is intrinsically associated with the reduction of in-band quantization noise on account of increased out-of-band components. In this way, filtering and rate-reduction operations have to be realized on the modulated output sequence in order to reject such components, and accommodate the sampling frequency to the Nyquist-rate of the signal. With TOSCA decimator cascades can be defined either as having standard Sinc^k(f) frequency response (moving average filters) or user-specified finite-impulse response (FIR filters). Non-ideal characteristics such as round-off of filter coefficients can also be considered.

**Analysis Section**

The design of oversampled A/D converters requires exact knowledge of the internal states of modulator and decimator parts. For this purpose TOSCA gives the user the possibility of analysing circuits both in time and frequency domains, as well as calculating the overall SNR performance for different amplitudes of the input signal. Moreover, the user has the possibility to examine the amplitude histogram associated with any circuit node, which is useful when designing the dynamic range of the integrators.

**3. SIMULATOR STRUCTURE**

The numerical analysis of oversampled data converters requires the simulation of a very large number of clock periods. Conventional simulators are inadequate for this type of long analysis as they use iterative approaches and/or require the inversion of matrices. In order to limit the CPU time to reasonable levels, the only known method is to have explicit and independent solutions for each building block, i.e. associate each element with a behavioural model.

Let us consider for example the SC integrator in Fig. 3. We can say that this element is a one-way block in the sense that (i) the output has no effect on the input values and (ii) the internal state-variables can be updated simply by knowing the actual inputs and the previous integrator state. In general, oversampled modulators verify these two conditions, and so we can construct entire circuits by simply interconnecting a set of basic building blocks. Finally, if any loop in the circuit contains at least one delay, then the network is said to be computable, i.e. it is possible to reorder the network nodes in such a way that a sequential node updating can be performed [3]. This is the approach adopted in TOSCA. In the following part we describe in detail the simulator flux diagram—see Fig. 4.

**Input file parsing**

The input file containing the netlist is read, analyzed and the netlist information is transferred to internal data structures. Error and warning messages, if any, are displayed at this point.

**Computability check**

Network nodes are reordered in order to allow sequential updating of the state variables. The computability of the network is identified at this stage of the program.

---

Fig. 3: Basic Switched-Capacitor (SC) integrator considered in the simulator

![SC Integrator Diagram](image)

Fig. 4: Flux diagram of the program
**Time parameter adjustment**

The exact number of samples required in the modulator section is calculated. The simulation length depends on the number of samples required at the end of the processing chain, on the decimator factor and on the filter length.

**Dynamic memory allocation**

Memory is allocated dynamically. Data structures are created and removed according to the needs of each simulation step. Thanks to this feature no structural limitations are imposed to the circuit complexity nor to the number of clock periods to be analysed.

**Analysis**

Circuit analysis is performed in time domain. Firstly, the modulator is simulated for the entire analysis interval and the discrete-time signals are stored at the required nodes. Secondly, the modulator output is used as the input signal for the decimation block and, as for the modulator part, the required node voltages are stored in internal data vectors. At this point the required analysis commands are performed, namely a simple printing of the node voltages, the calculation of the amplitude histogram, the frequency spectra or the SNR evaluation.

Frequency analysis is realized with the Discrete Fourier Transform (DFT) or, when both possible and convenient, the Fast Fourier Transform (FFT) [14]. Windowing effects are reduced by using common windows [15].

Because of the very nature of oversampled A/D converters, the evaluation of the overall performance is better specified in terms of SNR parameter (code density test procedures could also be applied but at the expense of very time consuming simulations [16,17]). As illustrated in Fig.5, a specific algorithm was developed that calculates very accurate measures of the SNR parameter in time-domain. This algorithm is not limited to fixed frequency values. The dynamic nature of the sequence length allows the calculation of the DFT at a single frequency point, being subsequently subtracted from the signal in order to obtain the remaining noise components. The mean squared value $E(e^2(n))$ defines the noise power needed to calculate the SNR.

![Block diagram of the algorithm used to calculate the SNR performance parameter](image)

4. SIMULATION EXAMPLES

In this section we present some results concerning the use of TOSCA in the simulation of $\Sigma A$ A/D converters.

The simulations were performed for the circuit of Fig.2 with a $3 \text{ kHz}$ sinusoidal input.

Fig. 6.a illustrates the time-domain analysis of the second integrator. As can be seen, the signal at this point is a noise-modulated version of the input, were we can identify very sharp variations of the integrator output. This information can be useful, for example, to define the amplifier specifications in terms of settling-time. Fig.6.b shows the amplitude histogram at the same node. The curves which have been plotted correspond to two distinct cases of integration gains, namely $k=1/2$ and $k=1$ for both integrators. This type of analysis can be useful in the optimization of the dynamic range of the integrators. TOSCA can also calculate the frequency spectrum at any node of the network. For example, the frequency spectrum of $V2$ of the signal+noise at the modulator output is shown in Fig.7. Here, the shaping of the quantization noise is clear as well as the in-band noise level relating to the signal power. Coarse calculations of the in-band noise power can also be made if the power spectral density representation $V^2/\text{Hz}$ is asked for instead of the frequency spectrum. As a final example, the converter SNR performance for increasing amplitudes of the input signal is shown in Fig.8. The represented curves refer to distinct values of the output swing of the amplifiers. In this case the analysis performed at a histogram level can be complemented with an SNR analysis in order to define the minimum amplifier swing for which there is no significant degradation of the SNR characteristic.

5. CONCLUSIONS

In this paper we have presented a software tool intended to help designers of oversampled data converters. The tool is fully user-friendly in the sense that the circuit description is made by means of a netlist. Generic circuits can be constructed by interconnecting a series of basic building blocks. Specific analysis such as SNR, time and frequency domain evaluations are also available. Behavioural models were developed which take into account the most significant non-idealities associated with the components, allowing in this way the simulation of very long transients in reasonable CPU time.

**Acknowledgements**

The authors are grateful to the European Community (EEC, JRC-DG XII) and to the Consiglio Nazionale delle Ricerche - Progetto Finalizzato Telecomunicazioni for their financial support.
Fig. 6.a - Time domain analysis of the output voltage of the second integrator with integration gain k=1

Fig. 6.b - Amplitude histogram of the output voltage of the second integrator with integration gain k=1 (dashed line) and k=1 (continuous line)

Fig. 7 - Frequency Spectrum at the output of the modulator

Fig. 8 - SNR vs. input amplitude for different values of amplifier output swing

References