

Mixed-Signal Modelling in VHDL for System-on-Chip Applications

Francis Pichon, Stéphane Blanc, Bernard Candaele
Thomson-CSF, RGS Division, 66 rue du Fossé Blanc, BP 156,
92231 Gennevilliers Cedex, France.

Abstract

Function transfer based modelling techniques are applied to the IEEE VHDL language to develop behavioral analogue models. This allows mixed-signal simulations integrated into a top-down ASIC design flow for system designers. An application to a clock module for a DSP-core ASIC, integrating a Phase-Locked Loop, demonstrates the usefulness of such an approach for the validation of large systems containing a small portion of analogue functionalities.

Key Words: Analogue VHDL modelling, Mixed analogue-digital simulation, Mixed signal ASIC design, Phase-Locked Loop.

1. Introduction

Modern telecommunication circuits integrate into a single chip a complete system, as for instance Modem, Voice-Coder, Telephone Answering Device, etc. A typical architecture of a system ASIC (figure 1) integrates a large digital core, and some mixed-signal cells. The digital core includes a Digital Signal Processor (DSP) core, and some custom interfaces. The type of mixed-signal cells is very limited, and usually includes only a Phase-Locked Loop (PLL) for in-circuit clock generation, and Sigma-Delta Analogue-to-Digital (AD) and Digital-to-Analogue (DA) converters.

In the point of view of system designers, there are two main requirements to use a mixed-signal simulation environment, providing capabilities both for analogue and digital behavioral modelling. The first one is due to the high complexity of the mixed-signal cells. For instance a clock generator module integrates an analogue PLL plus counters, programmable dividers, sequencers, etc. In the same order, a sigma-delta ADC is composed of an analogue modulator, and two digital filters. These filters may be realized either by digital logic, or by DSP software, or by a combination of both. Thus, the

validation of such modules requires a huge amount of mixed-signal simulations to validate the integration of the analogue cell into its digital environment. The use of a transistor level circuit simulator, as SPICE, is not appropriate for such validations.

The top-down design methodology used for System-on-Chip ASICs introduces the second requirement. The design engineer compiles his ASIC from different system level building blocks (DSP or μ P cores, memories, serial I/O, parallel ports, analogue cells, etc). There is a need of mixed-signal simulation at an early design stage, when analogue cells and models are not yet available at the transistor level. A behavioral model of the analogue cell, based on the specification, may be used to realize either quick investigations (choice of an architecture, design for test considerations, etc) or early design phase validations.

The choice of a mixed-simulation environment is not so obvious. There are solutions based on the use of an analogue simulator, and a proprietary behavioral modelling language, such as Saber from Analogy, or Eldo from Anacac. These environments are limited by the use of an analogue simulation kernel, and are not yet really integrated into a top-down ASIC design flow.

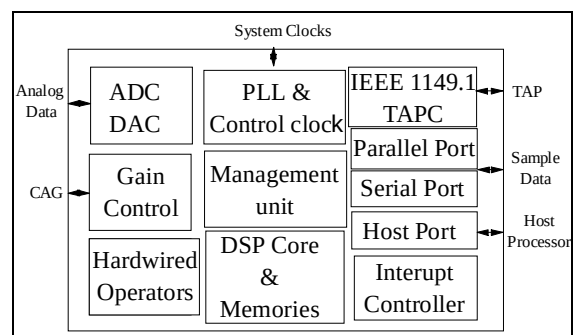


Figure 1: Architecture of a mixed-signal Telecommunication System ASIC

Up-to-date top-down ASIC design flow are based on VHDL modelling. Unfortunately, VHDL is dedicated to

digital modelling, and the analogue extension (IEEE P1076.1) is not yet available for industrial applications, even if some commercial products are announced (Mentor, Cadence, Anacac, etc). Thus, we have decided to use a methodology based on the modelling of the transfer functions compatible with event-driven simulators [1][2], to achieve mixed-signal simulations within a standard VHDL (IEEE Std1076-1992) environment [3]. This approach provides mixed-signal simulations fully integrated in the system top-down ASIC design flow.

The following chapters of this paper describe the analogue VHDL modelling techniques, and describe an industrial application of a mixed-signal clock generator for an ASIC module.

2. VHDL Behavioral Modelling Techniques

2.1 Representation of analogue signals

This section presents the process used to model analogue signals using the current IEEE Std 1076 VHDL language. The method uses a Piece Wise Linear (PWL) approximation to model the analogue signal.

First, the mathematic function representing the signal variations in time and value is developed using Taylor series :

$$U(t_0 + \Delta t) = u_0 + \Delta t * U'(t_0) + \Delta t^2 * \frac{U''(t_0)}{2}$$

This equation gives a linear representation of the signal during the time Δt with an error evaluated by the second order term of the development.

Leading to this principle, a type has been specified to represent analogue signals in VHDL, according to the implementation of the PWL approximation. This VHDL analogue signal type is a record of four fields (figure 2)[4] which contain the current value of the signal (Ampl), the gradient for the PWL segment (Grad), and the beginning (Start) and the ending (Stop) times of the current PWL approximation.

Those four fields are calculated at each event during the simulation in order to predict the next value of the signal for the forthcoming VHDL event. A segment of the PWL approximation is specified at each event.

TYPE	Ansigs	IS
	RECORD	
	Ampl	: real;
	Grad	: real;
	Start	: time;
	Stop	: time;
	END RECORD;	

Figure 2: Proposed VHDL analogue signal type

In the opposite of a simple sampling process, this VHDL record provides a continuous representation of an analogue signal. The value of the analogue signal can be evaluated between two events of the simulation by a linear interpolation. Furthermore, a realistic and dynamic description can be achieved by using this representation. For example, an rising edge of a square wave form generator can be described according a slew-rate value.

An other characteristic of this technique lies on the fact that the time step is a variable between consecutive simulator events. The step is a function of the accuracy required by the application. Figure 3 presents simulation results of a sine waveform (110 events per period), where the VHDL events are represented by vertical strokes. Its shows clearly that the event frequency changes during the simulation according to the gradient variations: an increase of the number of events depends on the non-linear variations of the signal.

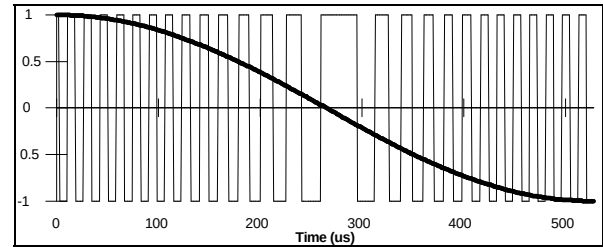


Figure 3: PWL modelling of a sine wave signal

This description provides an accurate and dynamic model. It allows to represent very easily the transient characteristics of an analogue cell, in the view of a high level behavioral description.

2.2 Modelling of basic analogue cells

The starting point of the PWL modelling method for basic analogue cells is the transfert function of the cell in the Laplace domain $F(p)$. This function allows to get the mathematical equation $Out(t)$ of the cell's output.

$$Out(t) \Leftrightarrow Out(p) ; Out(p) = F(p) * \epsilon(p)$$

For this calculation, the input cell signal $\epsilon(t)$ is considered as linear because of the PWL approximation.

$$\epsilon(t) \Leftrightarrow \frac{a}{p} + b;$$

For more complicated cells, the analogue cell model is built by assembling some basic analogue cells together.

The programming methodology to model these cells in VHDL is detailed in figure 4. Each cell is modelled by a VHDL process based on a "never ending" loop

which generates the output signal and controls the accuracy of the model. For each execution of the loop, a VHDL event is generated on the cell's output signal.

The first step of the VHDL process consists in scanning all the fields of the cell's inputs signal. These values are used for the evaluation of the cell's output signal.

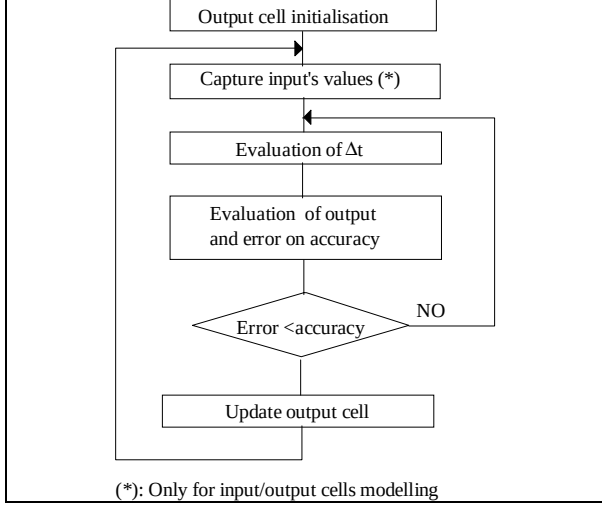


Figure 4: VHDL model structure for basic analogue cells

Then, the main task in this process is the evaluation of the time step Δt . This evaluation takes two steps. First, it is evaluated according to the error on the accuracy represented by the second order term of the Taylor development. Δt must verify the following condition:

$$\frac{\Delta t^2}{2} * \frac{\partial^2 V_{out}}{\partial t^2} \leq Accuracy$$

From this condition, the largest value of Δt is considered:

$$\Delta t = \sqrt{\frac{2 * Accuracy}{\frac{\partial^2 V_{out}}{\partial t^2}}}$$

According to this first value, the error between the real signal and its PWL approximation is calculated at time $t_0 + \Delta t/2$ (t_0 represents the current time of simulation). If the error is larger than the required accuracy, Δt is evaluated a second time. As the signal is linear during the time interval $[t_0, t_0 + \Delta t/2]$, Δt is adjusted by a combination in proportion of the accuracy, the error and the previous value of Δt :

$$\Delta t = \sqrt{\frac{2 * Accuracy}{\frac{\partial^2 V_{out}}{\partial t^2}} * \frac{Accuracy}{Error}}$$

Thus, the conditions for the VHDL process activation are either an event on one cell's input, or the end of the prediction time Δt . The principle of basic analogue cells activation is declared in the program by the following VHDL instructions :

WAIT ON Input_Event **FOR** DeltaT;

According to these activation conditions, when two analogues cells A and B are assembled together (figure 5), the input of Cell B provided by Cell A is therefore strictly linear. Then, each cell has its own time step and its own accuracy in order to minimize the number of events, according to the requirements of the designer.

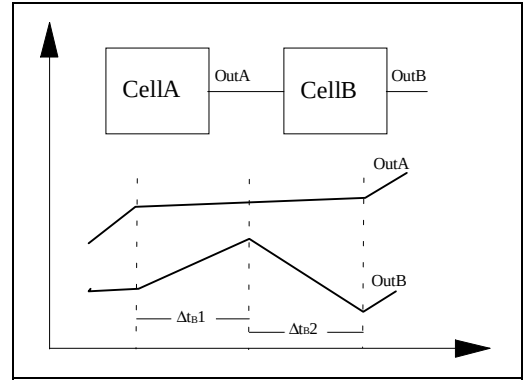


Figure 5: Process activation for VHDL analogue cells

For the generator cells, the signal is implemented from its equation in time (sinus) or its physical description (square wave). As this type of cell has no input the process activation is limited to the prediction time (Δt) condition:

WAIT FOR DeltaT;

Such a VHDL process allows to model all kind of analogues signals with the required accuracy for system simulations. In order to outline the capabilities of this modelling technique, a VHDL model of a PLL is presented in the next chapter.

3. Application To a Clock Generator

3.1 Circuit description

Figure 6 presents the schematic of a programmable clock generator. This ASIC module is part of Communication chips (such as Modem, Voice-Coder, etc) to generate the clock for the integrated DSP core. The cell provides the clock frequency to achieve the required processing performances of the DSP core, and

allows to reduce the power consumption of the communication system. It generates an output clock with a frequency between 20 to 40 Mhz, using an input clock frequency from 8 to 200 KHz.

This module is composed of two subcells. The analogue part (APLL) includes the phase comparator, the charge pump, and the Voltage-Controlled Oscillator (VCO). The digital part (DPLL) integrates the frequency dividers, control logic, and test logic. The low pass filter is realized using external devices.

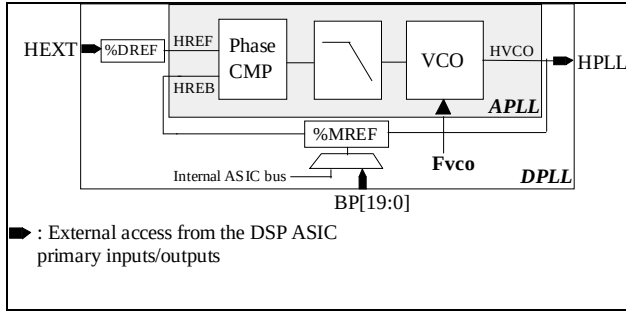


Figure 6: Mixed-signal clock generator module

In functional mode, the external clock, HEFT, is divided by DREF, to provide the reference clock, HREF, to the phase comparator. The output of the analogue PLL cell, HVCO, is divided by MREF to provide the second input of the phase comparator, HREB. Thus the frequency of the output clock, HPLL, is provided by the following relationship:

$$F_{HPLL} = F_{HREF} * MREF = F_{HEFT} * \frac{MREF}{DREF}$$

The coefficients MREF and DREF are programmed according to the DSP core's processing performances required by the application. It is also possible to chose to drive the DSP clock directly from the external clock, HEFT which is the default for the initialization of the chip and when the reset is enabled. The PLL may be disabled via the FVCO signal, when the cell is not used, to preserve the power consumption of the chip.

3.2 Analogue simulation results

A VHDL behavioral model of each device of the analogue part, APLL, (i.e the phase comparator, the charge pump, the first order filter, the VCO and its current sources) has been developed. The phase comparator model is implemented using a behavioral description of a digital implementation of such device (3-state machine). The Filter model uses the techniques described in chapter 2 of this paper, while the VCO model is a mathematical representation of the characteristics of this cell.

The VHDL simulation results of this part provides a good estimation of the main characteristics of the analogue cell, as for instance the lock time. Figure 7 presents the PLL locking transient results, for 800 periods of the HREF clock, with an input frequency of 40 MHz. The run-time corresponding to this simulation is 25 seconds on a SUN Sparc5 station. The number of simulation events is 20.000.

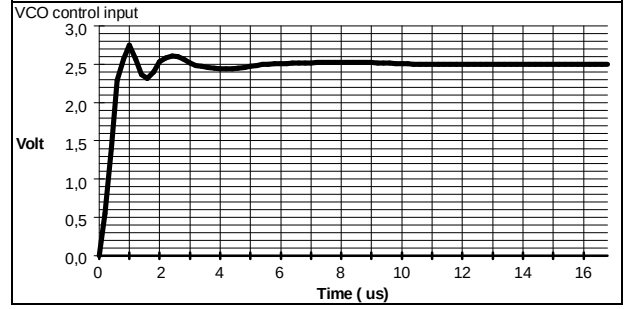


Figure 7: PLL locking transient result

3.3 Mixed-signal simulation

A VHDL model of the full digital part (DPLL) has been developed. The two parts, ie the digital one and the analogue one, have been simulated together in the early phases of the design of a Voice-Coder chip. These simulations have allowed the designers to validate the specifications of the complete mixed-signal module, as for instance the initial conditions, the ability to lock the PLL, the lock time, the output frequency, etc. All these validations have been realized at a time when the PLL cell was not yet available.

The same validation has also been realized using either a Register Transfer Level (RTL) VHDL model, or a gate level model of the digital part, instead of the behavioral one. Thus a validation is realized at each phase of the digital design, while the analogue cell is designed and validated by another design department.

3.4 Experimental results

This section presents results of the simulation of an voice-coder device using the VHDL model of the pll (DPLL) as a clock generator. The table 1 gives the performances of this PLL regarding the total simulation time.

According to the results of the table 1, the time of simulation in per cent of the analogue part of the circuit is 10% for a behavioral simulation, and 5% for a gate-level one.

	Behavioral ASIC model		Gate-level ASIC model	
	with PLL	without PLL	with PLL	without PLL
Simulation time (us)	100	100	100	100
nb of events	864.932	833.734	5.176.738	5.060.764
CPU time (s)	57	64	555	525

Table 1: PLL simulation results

4. Others applications

The modelling methodology presented in this paper have been used to implement many basic analogue cells at a behavioral level [5]. Those cells take part from a VHDL library which includes :

- analogue generators cells (sine-wave, square, etc)
- analogue filters, A/D and D/A converters
- PLL and Sigma-Delta modulators.

In order to validate all those VHDL models, a VHDL analogue package has been created with some post processing functions like a Fourier transform process, a VHDL interface for graphic edition softwares, and basic operators like adders, subtractors and multipliers.

5. Conclusions

Behavioral event-driven analogue simulation techniques have been implemented using the standard VHDL language. This approach is useful for the design and validation of complex system-on-chip integrating a small proportion of analogue functionalities. System ASIC designers have a simple way to investigate architectures, test techniques, etc, as well as to realize the validation of the complete mixed-signal chain. All these investigations and validations are not really possible with transistor level circuit simulators, due to the required run-time to perform them. These simulators are more dedicated to the validation of an analogue cell. Using the proposed VHDL modelling approach a designer may realize the validations in the early phases of the design, before the analogue cell layout is available, and also in a complete system simulation.

This VHDL behavioral analogue modelling technique has been demonstrated on a mixed-signal clock generator. It has also been applied to complex analogue-to-digital Sigma Delta converters, including an AD modulator, a first digital filter implemented using digital logic, and a second filter implemented into a DSP core software. This model has been used to validate the integrity of the conversion chain.

Future work will be realized to compare this methodology with the forthcoming analogue extension of VHDL (A-VHDL, IEEE P1076.1) when available. Especially fonctionnalités and performances will be studied for both analogue modelling approaches to define the interest and the application areas for each of them.

6. Acknowledgements

This study has been realized in the scope of the ARTEMIS ESPRIT 3 project, under CEC contract 6138.

7. References

- [1] "Mixed analogue-digital simulation of ASICs using transfer function models", R.A Cottrell, Journal of Semicustom ICs Vol. 7, No. 4, 1990, pp 21-25.
- [2] "Event-driven behavioral simulation of analogue transfer functions", R.A Cottrell, Proceedings of the EDAC-1990.
- [3] "Mixed-signal ASIC simulation via an analogue modelling package", U. Ko, S. Schenck, B. van Eerden, W. Locke, M. Runsey, J. Sackett, Proceedings of the IEEE ASIC Seminar 1 Exhibit, Sept 17-21, 1990, pp 2.1-2.4.
- [4] "High level simulation and hardware description languages", R.A Cottrell, Analogue-digital ASICs - Circuit techniques, design tools and applications -, IEE Circuits and Systems Series 3, 1991, chapter 13, pp 295-307.
- [5] "Mixed-signal Simulation in VHDL - The VHDL_MixedLib Modelling Guide and Evaluation Report", F. Pichon, S. Blanc, ARTEMIS Deliverable 2.2.L.