On Accommodating Particular Analog System Models With VHDL

Gabriel Stefan Popescu, Technical University of Iasi, Romania

Abstract

In this paper the problem of accommodating particular analog system models, with emphasis on interconnection's representation, with discrete event simulators, and particularly with VHDL circuit descriptions, is discussed and a solution is proposed. The results can be successfully used for VHDL modeling and simulation, but they are implementation independent.

1: Introduction

A key requirement of hardware description languages is to accurately represent the timing behavior of the actual modeled hardware. For the purpose of developing digital integrated circuits (IC's) and systems VHDL is a leading solution. The analog behavior of the actual components, mostly of the interconnections, is the limiting factor of the IC's overall performances. Consequently modeling and simulation of analog aspects of the digital components became increasingly important.

Reduced order approximations of complex linear networks using Padé like approximations such as AWE [1], [2], CFH [3], or Krylov subspace approximations such as PVL [4] or [5], were developed and successfully applied, to approximate distributed parameter systems such as interconnections. The purpose is to adjust interconnections' models with circuit simulators while losing accuracy.

To enhance the simulation speed of VLSI circuits there has been made efforts to adjust interconnection models into discrete event simulators [6], [7]. Analog simulation techniques are incompatible with VHDL. In literature there are a few results reported on the convenient interconnections models for VHDL [8], [9].

In this paper the problem of accommodating some particular analog models with discrete event simulators, and particularly with VHDL circuit descriptions, is discussed and a solution is proposed.

2: The modeling approach

The VHDL grammar proposes two syntactic constructs: inertial and transport delay. The VHDL semantics defines ways to accurately solve problems of event transactions using these timing specifications. But this set of events is unappropriated to describe analog behavior [8], [9]. To simulate analog behavior with a discrete event simulator an appropriate set of events must be used and an event transaction scheme must be developed.

The proposed solution to this problem developed particularly for VHDL is to incorporate into the models a supplementary layer which makes the translation from the set of adequate events for the analog part and the set of events used by VHDL simulation algorithms. This layer is also responsible for the synchronization between the analog and the digital stages of the simulated system.

The discrete event models for analog systems must be able to be activated by the input signals and, considering their nature, they must have a mechanism to self activate until they reach a temporary steady state corresponding to the input and output signals' state. The initial activation is a consequence of the activity of the drivers connected at the interconnection taps while all other events occur in order to update the state of model during the propagation of the signals.

Using a reduced order approximation of the circuit implies knowing the approximating pairs of poles/residues. Some of these pairs correspond to the dominant poles of the original system, and a few do not correspond to poles in the original system, but account for the effects of the remaining poles. Using y-parameter description for the N-port which represents the modeled system the in the frequency domain, its equations are:

$$I(s) = Y(s)V(s) \qquad \textbf{(1)}$$

where

$$Y(s) = \begin{bmatrix} Y_{11}(s) & Y_{12}(s) & \dots & Y_{1n}(s) \\ Y_{21}(s) & Y_{22}(s) & \dots & Y_{2n}(s) \\ \vdots & & & \\ Y_{n1}(s) & Y_{n2}(s) & \dots & Y_{nn}(s) \end{bmatrix} \qquad \textbf{(2)}$$

In (2) Y_{pq} , $p, q = 1, \dots, n$ are supposed to have the following shape:

$$Y_{pq} = K^{pq} + \sum_{i=1}^{m_1} \frac{k_i^{pq}}{(s + p_i)^{j_i^1}} + \\ + \sum_{i=m_1+1}^{m_1+m_2} \left(\frac{k_i^{pq}}{(s + p_i)^{j_i^2}} + \frac{k^*{}_i^{pq}}{(s + p^*{}_i)^{j_i^2}} \right) \quad \textbf{(3)}$$

All the transfer functions Y_{pq} have the same poles while their residues are different. Private pole factors can be easily added in Y_{ii} , $i = 1,...,n$ without affecting the pattern of the obtained set of equations. Further the significance of the elements involved in (3) are as follows:

$$\frac{k_i^{pq}}{(s + p_i)^{j_i^1}} \Rightarrow \begin{cases} j_i^1 = 1 : \textit{simple real pole}; \\ j_i^1 > 1 : \textit{multiple real pole}; \end{cases} \quad \textbf{(4)}$$

$$\frac{k_i^{pq}}{(s + p_i)^{j_i^2}} + \frac{k^*{}_i^{pq}}{(s + p^*{}_i)^{j_i^2}} \Rightarrow \\ \Rightarrow \begin{cases} j_i^2 = 1 : \textit{simple pair of complex poles}; \\ j_i^2 > 1 : \textit{multiple pair of complex poles}; \end{cases} \quad \textbf{(5)}$$

and K^{pq} is a factor representing any direct coupling between ports.

3: The model synthesis

The model is to be deduced from (3) taking into account the goal to obtain a good accommodation with the discrete event simulators. For the analog system the current at the port p can be expressed as the superposition of its components:

$$I_p = \sum_{q=1}^{n} I_{pq}(s) = \sum_{q=1}^{n} \sum_{i=1}^{1+d} I_{pqi}(s) \quad \textbf{(6)}$$

This approach enables the use of a multi rating integration method because of a good equation decoupling. This is of crucial importance to enforce a good compromise between the cost of the simulation and the accuracy of the results.

3.1: Real poles

Consider the term:

$$I_{pqi}(s) = \frac{k_i^{pq}}{(s + p_i)^{j_i^1}} V_q(s) \quad \textbf{(7)}$$

The case of a single pole is a particular one for which $j_i^1 = 1$. Without losing the generality the case of a single pole is further discussed. The time domain equation corresponding to (7) in this case is:

$$\frac{d}{dt} i_{pqi} = -p_i i_{pqi} + k_i^{pq} v_q \\ i(t_0) = i_0 \quad \textbf{(8)}$$

Using an implicit integration method, the order 2 Adams-Moulton algorithm in this particular case, will lead to the discretisation of (8):

$$i_{pqi\,n+1} = i_{pqi\,n} + \frac{t_{n+1} - t_n}{2} \cdot \\ \cdot \left[-p_i (i_{pqi\,n} + i_{pqi\,n+1}) + k_i^{pq} (v_{q\,n} + v_{q\,n+1}) \right] \quad \textbf{(9)}$$

A time domain step involves solving (9) with respect to $i_{pqi\,n+1}$:

$$i_{pqi\,n+1} = \left[i_{pqi\,n} \left(1 - \frac{t_{n+1} - t_n}{2} p_i \right) + \right. \\ \left. + \frac{t_{n+1} - t_n}{2} k_i^{pq} (v_{q\,n} + v_{q\,n+1}) \right] \left(1 + \frac{t_{n+1} - t_n}{2} p_i \right)^{-1} \quad \textbf{(10)}$$

An amplitude step involves solving (9) with respect to $t_{n+1} - t_n$ knowing the value of $i_{pqi\,n+1}$ when using an amplitude discretization step.

$$t_{n+1} - t_n = 2 (i_{pqi\,n+1} - i_{pqi\,n}) \cdot \\ \cdot \left[-p_i (i_{pqi\,n} + i_{pqi\,n+1}) + k_i^{pq} (v_{q\,n} + v_{q\,n+1}) \right]^{-1} \quad \textbf{(11)}$$

The VHDL implementation of this approach consists of a process which is sensitive on the input signal and on a wait statement based mechanism which schedules the activation of the process using amplitude steps based on (11) until a local steady state is reached. When the process is activated by the activity of the input signal, a rescheduling step begins with the purpose of synchronizing the value of the output for the respective moment of time. This is a time step based on (10).

3.2: Complex Poles

Consider the term:

$$I_{pq\,i}(s) = \left(\frac{k_i^{pq}}{(s + p_i)^{j_i^2}} + \frac{k^*{}_i^{pq}}{(s + p^*{}_i)^{j_i^2}} \right) V_q(s) \quad \textbf{(12)}$$

The case of a single pole is a particular one for which $j_i^2 = 1$. Without losing the generality the case of a single pole is further considered. The time domain equations corresponding to (12) in this case are:

$$\begin{cases} \frac{d}{dt} i_{pq\,i\,1R} = -p_{i\,R}\, i_{pq\,i\,1R} + p_{i\,I}\, i_{pq\,i\,1I} + k_{i\,R}^{pq} v_q \\ \frac{d}{dt} i_{pq\,i\,1I} = -p_{i\,I}\, i_{pq\,i\,1R} - p_{i\,R}\, i_{pq\,i\,1I} + k_{i\,I}^{pq} v_q \\ i_{pq\,i\,1R}(t_0) = i_{pq\,i\,1R\,0} \\ i_{pq\,i\,1I}(t_0) = i_{pq\,i\,1I\,0} \\ i_{pq\,i} = 2\, i_{pq\,i\,1R} \end{cases} \quad \textbf{(13)}$$

In (13) the R and I indexes were used to denote the real part and the imaginary part of the variables affected by them respectively. The discretization of (13) and the implementation of the VHDL models is similar, with minor changes, to the one described in the case of real poles.

3:3 Amplitude discretization

The determination of the appropriate value of the amplitude step given by the amplitude discretization used is of crucial importance because it decides in the end the number of steps to be made during a swing between two amplitude levels. Too many steps reduce the simulation efficiency, while too few affect the accuracy of the simulation results. Because the amplitude swing of the variables involved in the model can have unrealistic values, the amplitude step must be calculated using some assumptions. In this approach a uniform discretization of the amplitude domain is used, but others strategies might be taken into account. Also, an important assumption is that the amplitude swing of the driving signals at the ports is known.

For the case of a real pole, considering that the local steady state is given by the value zero of the derivative of the output when the input value is constant, being a signal in a digital circuit, from (8) results that the amplitude step $\Delta i_{pq\,i}$ is:

$$\Delta i_{pq\,i} = \left| \frac{k_i^{pq} V_q}{M p_i} \right| \quad \textbf{(14)}$$

where Vq is the amplitude swing of the signal vq, and M is the number of amplitude steps.

By a similar manner, using (13), the amplitude steps for the case of a pair of complex poles $\Delta i_{pq\,i\,1R}$ and $\Delta i_{pq\,i\,1I}$ are:

$$\begin{aligned} \Delta i_{pq\,i\,1R} &= \left| \frac{\left(k_{i\,R} p_{i\,R} + k_{i\,I} p_{i\,I}\right) V_q}{M\left(p_{i\,R}^2 + p_{i\,I}^2\right)} \right| \\ \Delta i_{pq\,i\,1I} &= \left| \frac{\left(k_{i\,R} p_{i\,R} - k_{i\,I} p_{i\,I}\right) V_q}{M\left(p_{i\,R}^2 + p_{i\,I}^2\right)} \right| \end{aligned} \quad \textbf{(15)}$$

Obviously, (14) and (15) imposes some inherent numerical limitations.

4: Experimental results

Some simple test structures are presented to prove in this summary the validity of the proposed approach. In all the following pictures the input signal is denoted vin, and the output signals are denoted vout followed by suggestive prefixes. The amplitude swing of the input signal is 3.0.

In fig. 1 the test structure contains system with one real pole. The signal denoted vout1 corresponds to the case when a 3 samples per amplitude unit amplitude sampling rate is used. The signal vout2 is obtained using a 10 sample per amplitude unit amplitude sampling rate. The signal vexact is obtained using a classical integration algorithm based on the same integration method.

In fig. 2 the test structure contains system with a pair of complex poles. The signal denoted vout1 corresponds to the case when a 3 samples per amplitude unit amplitude sampling rate is used. The signal vout2 is obtained using a 10 sample per amplitude unit amplitude sampling rate. The signal vexact is obtained using a classical integration algorithm based on the same integration method.

The next test structure represents an interconnection with 5 taps. At one tap the input signal vin is connected while at the other taps there are connected capacitive loads. The monitored signals are vload(0), vload(1), vload(2) and vload(3).

In fig. 3 simulation results using only real poles in the model of the interconnection are presented. The amplitude discretization step is 3 samples per amplitude unit.

In fig. 4 simulation results using a much more complex model involving both real and complex poles in the model of the interconnection are presented. The amplitude discretization step is 3 samples per amplitude unit.

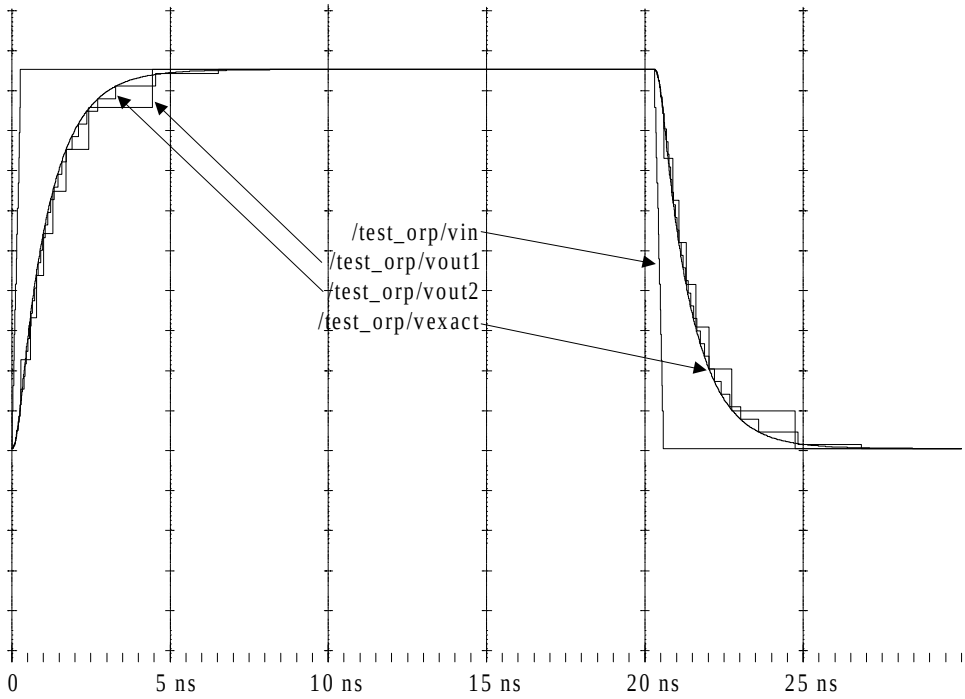

Figure 1 Simulation results for a system with one real pole

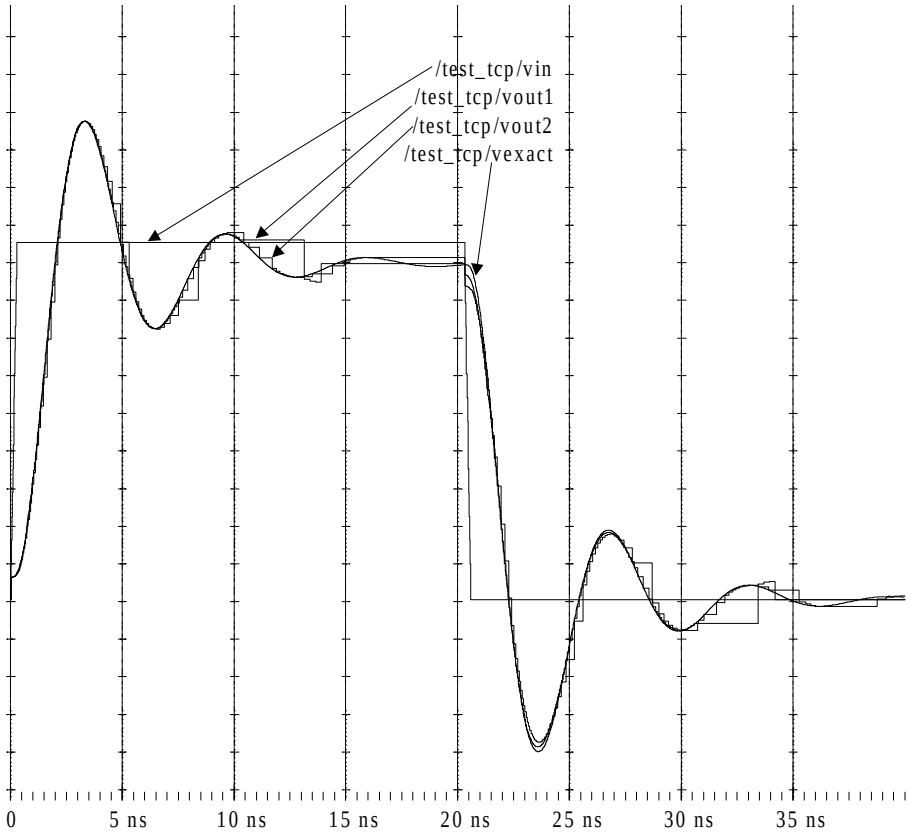

Figure 2 Simulation results for a system with one pair of complex poles

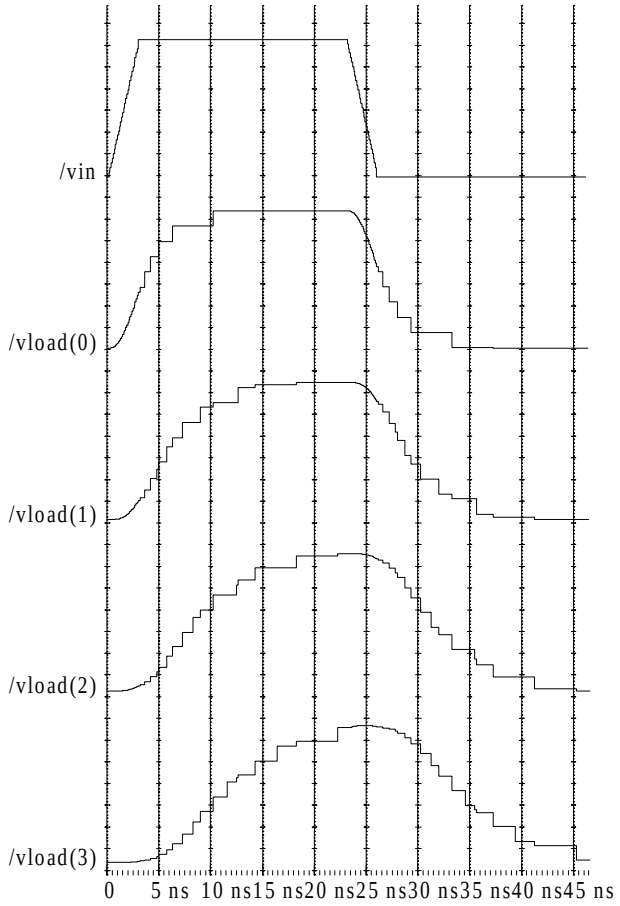

Figure 3 Simulation results for a interconnection modeled using real poles

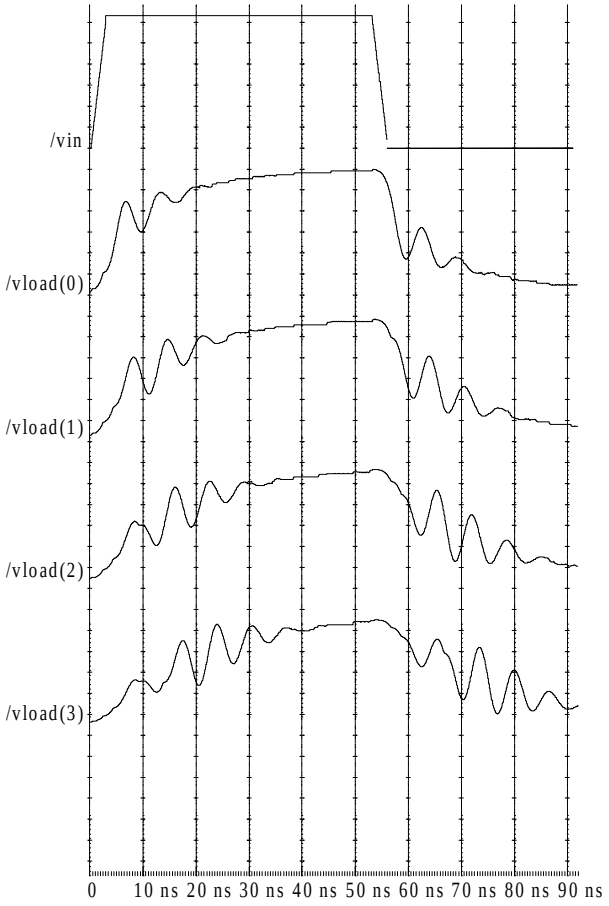

Figure 4 Simulation results for a interconnection modeled using both real and complex poles

4: Conclusions

The paper discusses the development of a VHDL analog system modeling approach with applications in interconnections' modeling. It uses a multi rate integration approach based on an implicit integration algorithm exploiting latency and superior stability properties of implicit integration methods. The modeling process is driven by the limitations of discrete event simulation algorithms, and particularly by VHDL limitations. The solution is to integrate an extra layer into the model in order to adapt the set of events and their transaction imposed by the analog stage to the discrete event approach used by the discrete event simulator. In this layer tow kind of iterations are used: amplitude step and time step. The former is needed for sincronization purposes.

The obtained modeling solution is able to represent the timing behavior of the complex on chip and off chip interconnections including crosstalk and non monotonic transitions using available VHDL constructions.

The presented approach is an alternative to a VHDL-AMS approach and remains usable for VHDL-AMS also. It is usable only into a particular simulation scenario mainly when the analog stage of the circuit is represented by nonideal interconnections.

References

[1] L.T. Pillage, R.A. Rohrer, "Asymptotic Waveform Evaluation for Timing Analysis", IEEE Trans. on CAD, v. 9, n. 4, 1990, pp. 352-366.

[2] S.Y. Kim, N. Gopal and L.T. Pillage, "Time Domain Macromodels for VLSI Interconnect", IEEE Trans. CAD, v. 13, n. 10, 1994, pp. 1257-1270.

[3] E. Chiprout, M.S. Nachla, "Analysis of Interconnect

Networks Using Complex Frequency Hopping (CHF)", IEEE Trans. on CAD, v. 14, n. 2, 1995, pp. 186-200.
[4] P. Feldmann, R.W. Freund, "Efficient Linear Circuit Analysis by Padé Approximation via the Lanczos Process", IEEE Trans. on CAD, v. 14, n. 5, 1995, pp. 639-649.
[5] L.M. Silveira, M. Kamon, J. White, "Efficient reduced-order modeling of frequency- dependent coupling inductances associated with 3-D interconnect structures", Proc. of 32nd ACM/IEEE DAC, 1995, pp. 376-380.
[6] G.S. Popescu, "On the Interconnect Macromodel Development for Source Based Event Driven Simulators", Proc. SCS'97, 1997, pp. 51-57.
[7] G.S. Popescu, I. Casian Botez, "Time Domain Response Computation for VLSI Interconnects", Proc. SCS'97, 1997, pp. 369-372.
[8] P.A. Walker, S. Gosh, "On the Nature and Inadequacies of Transport Timing Delay Constructs in VHDL Descriptions", IEEE Trans. on CAD, v. 16, n. 8, 1997, pp. 894-915.
[9] G.S. Popescu, "On VHDL Interconnections Analog Macromodels", Proceedings of the 1999 European Conference on Circuit Theory and Design, ECCTD'99, Stresa, Italy, Aug. 29-Sept 2, pp.900-904.