A New Concept for Accurate Modeling of VLSI Interconnections and its Application for Timing Simulation

Bernhard Wunder, Gunther Lehmann, Klaus D. Müller-Glaser
Institut für Technik der Informationsverarbeitung (ITIV)
Universität Karlsruhe, D-76128 Karlsruhe, GERMANY
Email: wun@itiv.etec.uni-karlsruhe.de, WWW: http://www-itiv.etec.uni-karlsruhe.de/

Abstract

This paper presents a new concept for accurate modeling the timing behavior of VLSI interconnections using frequency domain methods and taking into consideration distributed parasitics as well as lumped elements and contact holes. A piecewise linear signal representation is used to catch the waveform dependencies of submicron structures. The models are applied in an analysis tool for clock trees and in a concept for accurate post-layout timing simulation.

1. Introduction

Taking a look at the progress in the field of semiconductor technology, we observe decreasing feature size and increasing chip dimensions.

As transistor dimensions decrease, the propagation delay of single logic cells does as well whereas the output slew rate increases, leading to higher operating frequencies. On the other hand, global on-chip interconnections (connecting functional blocks or clock trees) get longer but smaller in cross dimensions. The primary parasitic on on-chip interconnections is capacitance C. According to the mentioned trends, resistance R becomes more and more relevant (see table 1). With higher operating frequencies, enhanced circuit technologies (BiCMOS, GaAs) and faster switching transistors, inductivity L also gains importance, even on chip. These parasitics and faster switching transistors lead to contrary trends in the contribution of cells and interconnections to the overall path delay. Table 1 proves this by enumerating the results of scaling theory [1]. A simple RC constant is taken as a first estimation for the interconnection delay.

Besides propagation delay, smaller feature size and faster switching logic cells lead to waveform shaping (e.g. by resistive damping), reflexions and overshoots (by inductive parasitics), coupling effects to adjacent lines, and dI/dt-problems. These effects play a major role on the dynamic behavior of VLSI circuits. In many cases the performance is even limited by the passive interconnections. Therefore, they have to be taken into account in the timing verification process of digital high speed VLSI systems.

Dimensions and interconnection parasitics	
Transistor dimensions (W, L, t_{gox})	$1/S$
Intercon. cross dimensions (h, w, w_{sp}, t_{ox})	$1/S$
Local interconnection length (l_l)	$1/S$
Die size and global intercon. length (l_g)	S_c
Resistance per unit length $(R' \sim 1/(w \cdot h))$	S^2
Capacitance per unit length $(C' \sim w/t_{ox})$	1
RC constant per unit length $(R'C')$	S^2
Propagation delay	
Intrinsic cell delay $(t_{pd}(g))$	$1/S$
Delay of local intercon. $(t_{pd}(i) \sim R'C' \cdot l_l^2)$	1
Delay of const. intercon. $(t_{pd}(i) \sim R'C')$	S^2
Delay of global intercon. $(t_{pd}(i) \sim R'C' \cdot l_g^2)$	$S^2 S_c^2$

Table 1. Results of the scaling theory (scaling factors S, S_C) for cell and interconnection delay

2. State of the art

There is a wide range of accuracy in modeling the timing behavior of **lossy** on-chip interconnections.

RC equivalent circuits: The most inaccurate model for a single interconnection is created by reducing the whole interconnection into a single C or RC element. A further step was done by W. C. Elmore [3], who refined the resistive and capacitive modeling of interconnections by using several lumped RC elements and calculating an equivalent time constant. Different output ports cannot be distinguished by the resulting exponential load characteristics. Based on this work, upper and lower bounds for the load characteristics at various output ports of RC trees were calculated in [11]. These simple C or RC models are mostly used by conventional gate level simulators in cooperation with back-annotation methods. The load capacitance including interconnection capacitance is precalculated and propagated to the driving cell as output load factor. In the VITAL concept [6], interconnect delay is calculated,

EURO-DAC '96 with EURO-VHDL '96
0-89791-848-7/96 $4.00 © 1996 IEEE

backannotated to the driven cells via SDF and handled as an additional input delay. Lumped *RC* equivalent circuits are also used by so-called switch level simulators [4].

Circuit simulation: One of the most accurate ways to calculate the dynamic behavior of VLSI systems is to use a circuit simulator like SPICE [9]. At this level, active components are represented by their equivalent transistor structure, interconnections are usually modeled by lumped equivalent *RLCG* circuits or by lossy transmission line models. Numerical solution algorithms for the resulting system of differential equations consume huge amounts of computation time, making this method inapplicapable for VLSI systems.

Frequency domain methods: To capture all distributed parasitics as well as frequency dependent and coupling effects, frequency domain methods can be applied. The time domain response is calculated by transforming the frequency domain descriptions into the time domain and applying numerical convolution or by rational function approximations. An overview of frequency domain methods can be found in [2] and [14]. Frequency domain methods are the most promising techniques for accurate modeling VLSI interconnections with small simulation expense. But in all cases, simplifications and approximations have to be made to be capable of handling the complex structure and all the parasitic and nonlinear effects. High accuracy will always result in large numerical effort.

Most of the discussed models for interconnections do not suffice the needs for both accurate results and small simulation expense. Especially the mentioned *RC* approaches have limitations in modeling VLSI interconnections in submicron structures with high clock frequencies and fast switching transistors. Only some frequency domain approaches are promising. Our approach uses frequency domain methods. Together with models for nonlinear cells it can be applied to model high performance VLSI systems.

3. Signals and models

There are three main criteria for timing models: accuracy, integration capability and simulation performance. To fulfill all these needs, we developed a new modeling concept for interconnections that can interact with appropriate models for logic cells, so that it can be applied to timing simulation of single interconnections or even complete VLSI systems. The main solution concepts for this development are:

Accuracy: The most important aspect for accuracy are the waveform dependencies of the timing behavior of cells as well as of interconnections. Therefore, we use a new kind of representation for the connecting signals, which can propagate information about the shape of the signal transition. As we use separate models for cells and interconnections, we can apply different modeling techniques with their respective advantages. In the case of interconnections, the model includes capacitive, resistive and inductive parasitics and considers the real structure with different paths to all output ports. An appropriate backannotation method is provided, too. In the case of cells consisting of nonlinear MOS transistors, the load characteristics are as important as the input pattern dependencies. In static timing analysis, pattern dependencies are neglected. Contrary to this, our models for dynamic timing simulation are capable of handling these effects.

Integration and handling: The acceptance of new tools mainly depends on their integration in a typical design flow. Adaption or conversion effort should be as small as possible. Our models are written in the standard programming language *C*. Applications use public domain tools or standard VHDL simulators. The new concept requires only small conversion effort to use typical VHDL netlists generated by synthesis tools and standard layout formats. The conversion steps can automatically be performed by special modules.

Simulation performance: Tight development budgets require very short design cycles at simultaneously growing system complexity. Therefore, small computation time is an absolute must for VLSI simulation tools. Our accurate event driven simulation approach can be performed by VHDL simulators. Furthermore, a mixed mode simulation allows to spend the simulation time only where it is needed, the rest of the system can be treated by conventional and fast digital models.

3.1. Signal representation

It is well known, that the signal shape has a major impact on the timing behavior of cells and interconnections [7]. Modeling these “analog aspects” means to calculate and propagate the waveform for each signal, i.e. the shape of each individual event. In a typical design flow of digital VLSI systems today, gate level models and event driven algorithms with two valued (boolean) or multi valued logic systems are used for timing simulation [5]. Conventional digital signal representations (eg. type `bit` or `std_logic`) are not able to propagate any information about the shape of a signal transition. Propagating the rise or fall time by desribing the transition as one ramp is also insufficient, because overshoots or exponential transitions cannot be reproduced. Therefore, we use a piecewise linear (PWL) signal representation and an event driven approach. Each signal transition is described as a sum of ramps, resulting in a PWL shape.

3.2. Models for interconnections

Interconnections are not treated as a single, dimensionless node but as an independent component with several ports and different behavior at each output port.

Resistive, capacitive and inductive parasitics are taken into account. Since the behavior of interconnections is nearly linear, frequency domain methods can be used to compute it. The models consists of two major blocks, shown in figure 1: In a presimulation phase, the transfer functions for all relevant signal paths are calculated based on structural and technological data (middle and left hand side). The response of PWL input events are calculated at simulation time (right hand side).

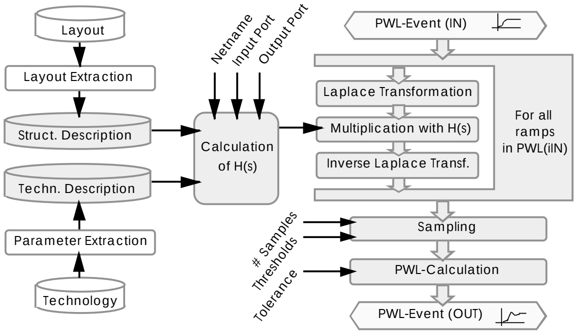

Figure 1. PWL model for interconnections

The calculation of the transfer function uses fourpole theory: The cascade matrix is build for each section of the interconnection net, each contact hole (CH) and each lumped element R, C or L (as representatives for resistance, cell input capacitance or bond wires). Interconnect sections are treated as homogenous twin wires (TW) with distributed parasitics. The exact cascade matrix for this fourpole element can be written as:

$$\mathbf{A}^{(TW)} = \begin{bmatrix} \cosh(\gamma \cdot l) & Z \cdot \sinh(\gamma \cdot l) \\ \frac{1}{Z} \cdot \sinh(\gamma \cdot l) & \cosh(\gamma \cdot l) \end{bmatrix} \qquad (1)$$

Because of the insulation properties of the dielectric in the frequency range of typical on chip signals, the conductance G can be neglected. Z and γ can then be written as functions of the parasitic interconnection parameters R', L' and C' [15]. Furthermore, the implemented algorithm uses an approximation for the hyperbolic functions.

Capacitive terminated branches are modeled by their input admittance, which is integrated into the direct path as a shunt conductance with the corresponding cascade matrix [15]. This way, the interconnection path from input i to output o is reduced to a series connected line of twin wires, lumped elements or shunt conductances. The final cascade matrix for this structure $\mathbf{A}_{(i,o)}$ can be built by matrix multiplication. The corresponding transfer function $H_{(i,o)}(s)$ can then be directly derived from matrix elements of $\mathbf{A}_{(i,o)}$ taking into account the terminating capacitance $C_{L,o}$:

$$H_{(i,o)}(s) = \frac{V_o(s)}{V_i(s)} = \frac{1}{A_{11_{(i,o)}}(s) + sC_{L,o} \cdot A_{12_{(i,o)}}(s)} \qquad (2)$$

As the input signal $v_i(t)$ is represented as a sum of ramps $v_{i,k}(t)$, each of these ramps contributes the portion $v_{o,k}(t)$ to the output signal. Each portion is computed using the rules of Laplace transformation ($v_{i,k}(t) \circ\!\!-\!\!\bullet V_{i,k}(s)$) and multiplication with $H_{(i,o)}(s)$. The continuous output signal $v_o(t)$ as the sum of these portions $v_{o,k}(t)$ is then sampled in the transition intervall and the resulting sample points are transformed into a piecewise linear form again by interpolation using least squares method (figure 1).

3.3. Models for logic cells

As discussed above, an appropriate model for logic cells has to consider the actual input situation (input pattern and the shape of the signal transitions) [7]. Even most of the macromodeling techniques for cells use simplified and linearized transistor equations. The complex cell structure is reduced into equivalent primitive blocks (e.g. inverters) [12], [8], so that pattern dependencies are neglected. Only few approaches reproduce the pattern dependencies. In the MOS cell primitive by Y.-H. Shih and S. M. Kang [13] we found an adequate model to represent the desired behavior. The circuit primitive is shown in figure 2. The transistors represent parallel connected NMOS- and PMOS-transistors. Series connected transistors are modeled as one equivalent transistor. Resistive and capacitive parasitics can be considered, too. Each substitute component in figure 2 and even the external feedback current I_{fb} contributes to the charging current of C_l. These current contributions lead to a differential equation for output voltage V. Assuming linear gate and drain voltages, a special form of I_{fb} and constant transistor states, this equation has the form of a so called quadratic Riccati differential equation. With an initial condition, this equation can be solved analytically for suitable time intervals. It is evaluated for discrete time points and the resulting sampled signal is transformed into an appropriate piecewise linear representation which is then propagated to the subsequent interconnection models. Although this model is one of the most accurate used for timing simulation, we made some enhancements in our implementation for submicron systems.

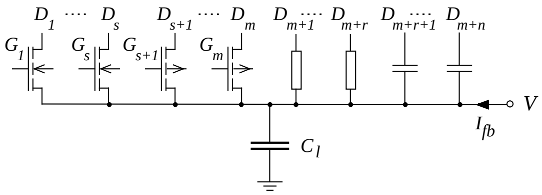

Figure 2. Circuit primitive by Shih/Kang

Pattern dependencies: The circuit primitive as proposed in [13] can only reflect pattern dependencies caused by parallel transistors. We enhanced the model to be capable of handling pattern dependencies caused by series connected transistors using a pattern and slope dependent calculation of the load capacitance as well as the size and the threshold voltage of the equivalent transistor.

Transistor equations: The Shih/Kang circuit primitive only uses the simplified level 0 transistor equations. Therefore, the output conductance λ (level 1 transistor equations) has been integrated by an operating point dependent recalculation of transistor parameters.

Influence of non capacitive loads: The node capacitance C_l in the Shih/Kang circuit primitive includes the output capacitance, the interconnection capacitance and input capacitances of the driven cells. Resistive properties of interconnections are neglected. By calculating a proper feedback current I_{fb}, we are now able to compute the output voltage considering the impedance of the load (interconnection plus driven cells).

4. Applications

The model for interconnections has been applied in tools for two typical design problems: CSKEW is a tool for analyzing and optimizing interconnection nets, e.g. for clock distribution (section 4.1), VAMP is a concept for accurate post layout timing simulation using the mentioned models for logic cells (section 4.2).

4.1. CSKEW: A tool for the design and analysis of clock trees

CSKEW consists of two main modules, NETEDIT and WAVES. Both modules have standalone functionality and together they control the simulation and visualize the results. An additional module coordinates NETEDIT and WAVES and provides dynamic help functions. The basic model routines are implemented in the programming language *C*, the graphical elements are realized using the public domain toolkit, Tcl & Tk [10].

The module NETEDIT, an intelligent clock tree editor, has the following features:

- visualizing interconnection structures with typical features (zoom, pan). There are graphical representations for all interconnection elements mentioned in section 3.2;

- providing graphical edit functionality for interconnection nets;

- checking of design rules for neighborhood relationships or intersections.

A typical session with NETEDIT is shown in figure 3.

WAVES, a combined stimuli editor and waveform display tool, offers the following features:

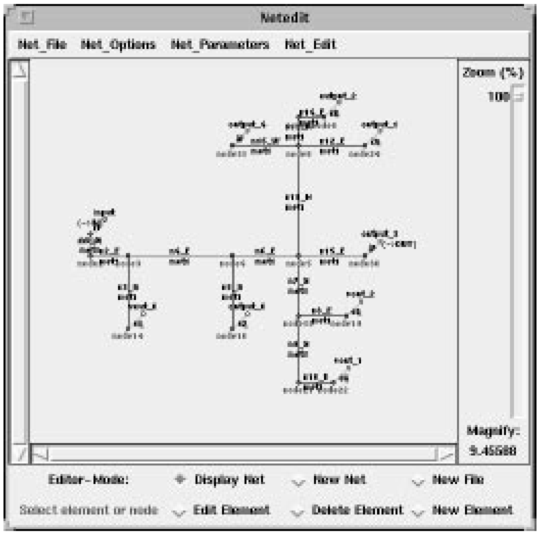

Figure 3. Module NETEDIT

- editing stimuli for simulation (manually or by automatic parametrizable clock signal creation);

- displaying simulation results (zoom, pan);

- performing measurement operations.

Figure 4 shows this module displaying a clock pulse as stimulus and the corresponding results.

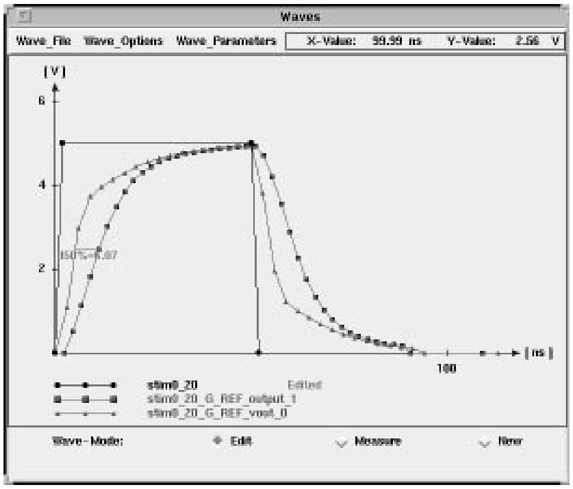

Figure 4. Module WAVES

Besides interconnections described by NETEDIT, layout extracted interconnections can be analyzed. After the stimuli to be applied are drawn or chosen by WAVES, the simulation can be started. Stimuli and the output waveform of a selected path are displayed in the same window, the delay is automatically annotated (single path simulation). Other simulation modes can compare two designated paths (compare simulation) or investigate all paths from a defined input (skew simulation). In the latter case, the waveforms of the output ports with maximum skew (minimum and maximum delay) are computed and displayed, and the skew is annotated, too. An example is shown in figure 4 for the net drawn in figure 3.

4.2. VAMP: A VHDL based concept for Accurate Modeling and Post layout timing simulation

To realize an accurate post layout timing simulation of VLSI systems integrated into a typical design environment using VHDL synthesis and simulation tools, a complete simulation environment was created. VAMP performs the necessary conversion steps of the design data. It creates the basis for both an efficient and accurate mixed mode simulation together with conventionally modeled components.

Figure 5 visualizes this concept. The PWL models for interconnections and cells are implemented in *C* and integrated into the VHDL simulator via a special interface architecture [16]. At the moment, we provide interface routines for *vss* (SYNOPSYS) and *Leapfrog* (CADENCE). As most of the VHDL simulators offer a *C* language interface, the concept is not restricted to these specific simulators.

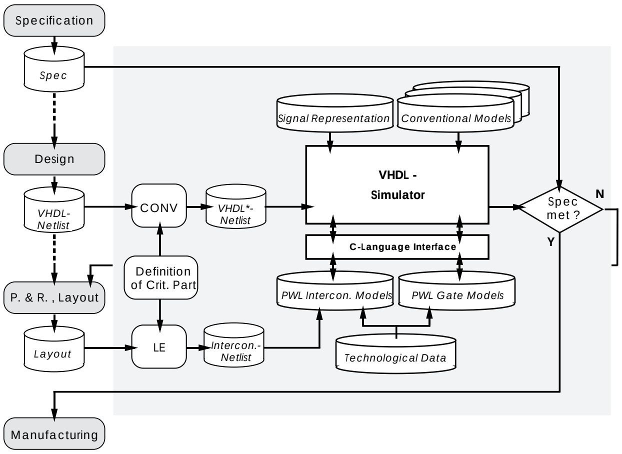

Figure 5. The VAMP simulation environment

If only critical portions have to be examined, these parts of the system have to be found and described. Based on this information, additional VAMP modules

- extract interconnection data for the relevant nets from layout (LE);
- modify the synthesized VHDL gate level netlist to integrate instances of the models for interconnections (CONV). Furthermore, conventional digital cell models are replaced by the new PWL models for the critical portions and proper signal conversion modules are provided at the boundaries of the PWL portion(s);
- provide a tool independent simulation control, stimuli editor and result viewing.

With all these modules, a user-friendly solution for an integration of the new modeling concept into a typical VHDL design flow and environment is realized.

To prove the usability of the VAMP concept and the proposed models for timing simulation, several examples were investigated [15], [16]. The results of a simple one bit full adder, built up of circuit primitives (Inverter, NAND, NOR, see figure 6), will be presented here. This circuit is stimulated by certain signal waveforms at the inputs *a*, *b* and *c_in*, shown in the upper part of figure 7. Two simulation runs were performed, the first one without interconnections, the second one with interconnection models integrated. The results for the first simulation show small spikes at $t = 47ns$ and $t = 57ns$ (not depicted here), resulting from the different paths from the input signals to the output *s*. If we integrate interconnection models into simulation, we get the results shown in the lower part of figure 7. The VAMP results (dashed line) are plotted together with

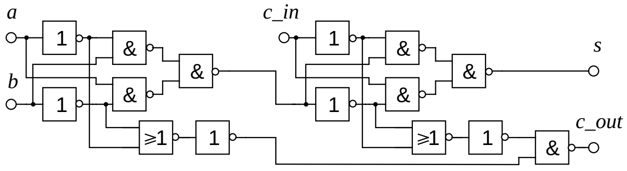

Figure 6. Structure of a one bit full adder

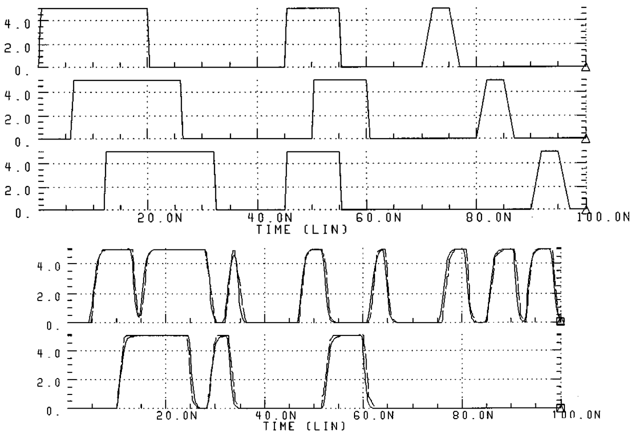

Figure 7. Full adder stimuli (a, b and c_in) and simulation results (sum bit (s) and carry bit (c_out)) with interconnections

the results of an equivalent circuit simulation (continuous line). The additional interconnection delays in the logic paths intensify the mentioned spikes, so that we can not speak of spikes any longer.

This example proves that additional interconnection delay not only affects the timing behavior, but even the logic behavior when integrated in critical paths. Even though the VAMP simulation needs a significant smaller CPU time for simulation than an equivalent circuit simulation (it is typically $50\ldots100$ times faster), the deviation is small ($\leq 5\%$).

5. Conclusion

We demonstrated the needs for accurate modeling the influences of interconnections as well as pattern and waveform dependencies for the timing simulation of VLSI systems with high performance, that can be easily integrated into an existing design flow. Therefore new models for interconnections were developed that use both an efficient event driven simulation algorithm and a voltage oriented piecewise linear form of the signal transition. The modeled behavior of the interconnections depends on the distributed parasitics, the structure and the loads of the interconnection branches, and the input signal shape. One application for this model is the analysis of clock trees. The tool CSKEW is an approach to combine both tasks: design and analysis. This way, the design of a complicated interconnection structure can immediately be optimized based on the simulation results.

Together with appropriate models for logic cells, a simulation of VLSI systems including an accurate representation of interconnection effects can be performed. The modeled behavior of the cells depends on the input signal shapes, the input pattern and the layout dependent load. The VAMP concept provides an environment that bases on data of a typical design flow using VHDL synthesis and simulation tools. Timing simulation using the VAMP concept offers remarkable improvements regarding accuracy compared to conventional methods and an advantage in CPU time up to the factor 100 at only 5% deviation compared to circuit simulation. The CPU time for VAMP simulations grows linear with the number of cells ($n^{1.0}$), while circuit simulation expense typically grows by $n^{1.2}$ up to $n^{2.0}$ [9].

Furthermore, a mixed mode simulation allows for spending the effort where it is needed, non critical portions are treated with digital modeling techniques.

6. Future work

To provide parametrizable models for a complete ASIC libraries, we have to implement models for busses and coupled interconnections as well as non basic functions of logic cells (e.g. complex gates) being capable of handling the PWL signal representation. The CSKEW application has to be extended to handle active clock trees including parametrizable buffers. For the VAMP concept and simulation environment a graphical user interface joining all the mentioned modules is still missing. With this interface a user-friendly push button solution for a mixed mode simulation of critical paths/portions together with conventionally modeled components will be realized.

References

[1] H. B. Bakoglu. *Circuits, Interconnections and Packaging for VLSI.* Addison-Wesley, Reading, MA, 1990.

[2] F. Y. Chang. Transient simulation of VLSI interconnects (A Tutorial). In *Proc. IEEE Multi Chip Module Conference*, 1992.

[3] W. C. Elmore. The transient response of damped linear networks with particular regard to wideband amplifiers. *Journal of Applied Physics*, 19, 1948.

[4] J. P. Hayes. Switch Level Modeling. *IEEE Design & Test*, August 1987.

[5] E. Hörbst. *Logic Design and Simulation*, volume 2. North Holland, Amsterdam, 1986.

[6] IEEE 1076.4 Working Group, FTP-address: ftp.vhdl.org. *VITAL V3.0 Specification*, 1995.

[7] G. Lehmann, P. Nagel, and K. Müller-Glaser. An approach to detailed modeling of digital CMOS gates for logic simulation using VHDL. In *Archiv für Elektronik und Übertragungstechnik*, pages 81–90, Hirzel-Verlag, Stuttgart, 1995.

[8] A. Nabavi-Lishi and N. C. Rumin. Inverter models of CMOS gates for supply current and delay evaluation. *IEEE Transactions on Computer Aided Design*, 13(10), 1994.

[9] L. W. Nagel. *SPICE2: A Computer Program to Simulate Semiconductor Circuits.* University of California, Berkeley, CA, 1975.

[10] J. K. Ousterhout. *Tcl and Tk Toolkit.* Addison-Wesley, Reading, MA, 1994.

[11] J. Rubinstein, P. Penfield, and M. A. Horowitz. Signal delays in RC tree networks. *IEEE Transactions on Computer Aided Design*, 2(3), 1983.

[12] T. Sakurai and A. R. Newton. Alpha-power law MOSFET model and its application to CMOS inverter delay and other formulas. *IEEE Journal of Solid State Circuits*, pages pp. 584–594, 1990.

[13] Y.-H. Shih and S.-M. Kang. Analytic transient solution of general MOS circuit primitives. *IEEE Transactions on Computer Aided Design*, 11(6), 1992.

[14] R. Wang. Transient analysis of interconnects (Tutorial on computer aided design for multi chip modules). In *International Conference on VLSI Design*, 1993.

[15] B. Wunder, G. Lehmann, and K. Müller-Glaser. Post layout timing simulation with accurate modeling of interconnections using a VHDL simulator. In *Proc. VHDL International Users Forum*, Boston, MA, October 1995.

[16] B. Wunder, G. Lehmann, and K. Müller-Glaser. VAMP: A VHDL based concept for accurate modeling and post layout timing simulation of electronic systems. In *Proc. ACM/IEEE Design Automation Conference (DAC)*, Las Vegas, NV, June 1996.