A Gate-Delay Model for High-Speed CMOS Circuits*

Florentin Dartu, Noel Menezes, Jessica Qian, and Lawrence T. Pillage

Department of Electrical and Computer Engineering

The University of Texas at Austin, Austin, Texas 78712

Abstract -- As signal speeds increase and gate delays decrease for high-performance digital integrated circuits, the gate delay modeling problem becomes increasingly more difficult. With scaling, increasing interconnect resistances and decreasing gate-output impedances make it more difficult to empirically characterize gate-delay models. Moreover, the single-input-switching assumption for the empirical models is incompatible with the inevitable simultaneous switching for today's high-speed logic paths.

In this paper a new empirical gate delay model is proposed. Instead of building the empirical equations in terms of capacitance loading and input-signal transition time, the models are generated in terms of parameters which combine the benefits of empirically derived k-factor models and switch-resistor models to efficiently: 1) handle capacitance shielding due to metal interconnect resistance, 2) model the RC interconnect delay, and 3) provide tighter bounds for simultaneous switching.

I. INTRODUCTION

As the minimum-feature sizes for integrated circuits scale downward, the portion of the delays attributable to the "gates" decreases, while the percentage of the delay due to the RC interconnect increases. It is reasonable to expect that a significant amount of the overall path delay is attributable to the RC interconnect delay. This relative increase in interconnect delay is mainly because the interconnect lengths do not generally scale due to increasing chip densities, while the RC per unit length product for the interconnect actually increases with scaling due to the fringing field effects. In addition, if the output impedances of the gates are decreased with scaling, the increased metal resistance will *shield* some of the load capacitance from the gate, thereby lowering the *gate delay*.

Unfortunately, this shift in delay dominance makes it significantly more difficult to efficiently model the delays with empirical equations, since the loading is not purely capacitive. Also, it is difficult for empirical delay models of the output signal delay and transition time to approximate the gate delay in the case of simultaneous switching, when more than one input switches, since these delays can be significantly different than the single input transition delays.

In this paper a new empirical delay model is proposed which accurately captures the gate delay while modeling the RC shielding and the RC interconnect delay. This is done in terms of a fixed dc resistance model for the gate's output impedance, along with a time-varying voltage source input. The resistance value is a function of the gate only, and there is a single resistance value for the pull-up path (there is a different one for the pull-down path) which is independent of the load and the input transition time. The voltage source parameters in this model are empirically generated in terms of the input waveform(s) and the "effective capacitance" loading. Extensions are proposed for the simultaneous switching problem and for estimating the gate and interconnect delay in terms of simple equations for higher level analyses in terms of the best- and worst-case switching.

* This work was supported in part by IBM Corporation, the Semiconductor Research Corporation under contract 94-DJ-343 and the National Science Foundation under contract MIP-9157263.

II. BACKGROUND

There are two approaches to gate delay modeling which have gained considerable acceptance: 1) a switch-resistor model comprised of a linear resistor and a step function of voltage, and 2) empirically derived expressions for delay and output-signal transition time as a function of load capacitance and input-signal transition time (*k*-factor equations). Both methods are empirically based, since even the second method requires empirical fitting to approximate the resistance value as a function of input transition time and output load.

A. k-factor models

When the load is purely capacitive, one can completely precharacterize a gate's delay and output signal behavior as a function of input signal transition time, t_{in}, and load capacitance, C_L [9]. The experimental data for the delay, t_d, and the gate-output waveform transition time, t_r or t_f, are generally fitted to *k*-factor equations:

$$t_d = k(t_{in}, C_L) \qquad and \qquad t_{r/f} = k'(t_{in}, C_L) \qquad (1)$$

For today's technologies the loads cannot be modeled as purely capacitive due to the RC interconnect, as shown in Fig. 1. Traditionally, the "gate delay" from A to B and output rise or fall time at B in this figure would be calculated from the equations in (1) using the total load capacitance (the sum of all of the interconnect capacitance and the gate input capacitances at C and D) and the signal transition time at A. The RC delay would then be analyzed separately using the waveform at B which is characterized by the result from (1).

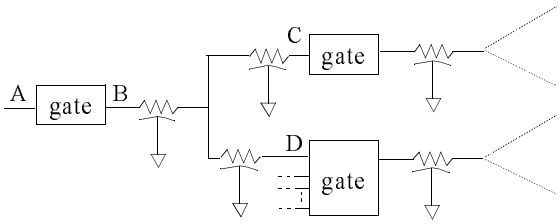


Fig. 1. A typical gate-delay problem.

This two-step delay approximation works well when the load "seen by the gate" is accurately approximated by the total capacitance of the net. That is, the procedure described for Fig. 1 assumes that the *driving point admittance* of the interconnect is equal to the total capacitance. It has been recognized, however, that this is an invalid assumption for today's high speed CMOS.

It was first recognized for ECL gates that the total capacitance was not a valid model of the load [6]. In [6], the 2nd order driving point admittance was modeled as a π-circuit, as shown in Fig. 2. Computing and storing *k*-factor equations for loads which are modeled by a π-circuit would be prohibitively expensive. In [6], the ECL gates are modeled by linear Thevenin equivalents which drive the π-circuit in Fig. 2.

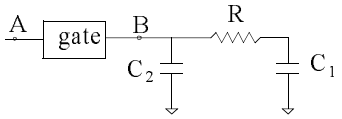


Fig. 2. A 2nd order driving point admittance model for the interconnect load at node B.

B. Linear resistance model

It would appear that the switch-resistor model is a more effective delay model when the load is not purely capacitive. That is, the resistance model is able to capture the interaction of the gate's output resistance and the RC load. Modeling the gate as a single resistance in series with a voltage step permits the use of computational efficient RC tree and mesh signal delay bounds [1,2] to approximate the complete gate and interconnect delay. In addition, it is recognized that the Elmore delay [3] facilitates the efficient incorporation of estimated RC circuit delays at the synthesis and floor planning levels of design. Higher-order RC analyses, such as the Asymptotic Waveform Evaluation (AWE) technique, can be used to generate an extremely accurate RC delay approximation; however, the result is only as accurate as the switch-resistor model for the gate.

Timing analysis tools such as TV [4] and Crystal [5] were developed using switch-resistor models to analyze the transistor level circuit descriptions. The main difficulty with these approaches is calculating a single linear resistor which captures the switching behavior of a CMOS gate. Recognizing that this resistance is a function of the gate's input signal transition time and output load, in [5], a single output resistance for the gate is empirically derived from something similar to a *k*-factor equation. That is, the resistance is calculated as the average output impedance as a function of the input signal transition time and the output load. This extension is advantageous from an accuracy standpoint; however, when the load is not purely capacitive, it has the same limitations as the *k*-factor model.

III. HIGH-PERFORMANCE GATE DELAY MODEL

A. The model

The proposed model is shown in Fig. 3. A ramp-like gate input signal is assumed throughout this paper as in (1); however, complex RC loading is considered here. The limitations of the ramp-like waveshape assumption are discussed in section VI.

The linear resistor is selected independent of the load and input waveform. It is computed in a pre-characterization step based upon the maximum allowable capacitive load. A complete description of the calculation of R_d is given in section IV.

To begin, we note the transfer function of the model in Fig. 3 is:

$$H(s) = \frac{\hat{V}_0(s)}{V(s)} = \frac{Z_L(s)}{Z_L(s) + R_d} \qquad (2)$$

The ideal waveshape for this voltage source model ($V_{id}(t)$) would be the one which produces the same output waveform as the gate itself.

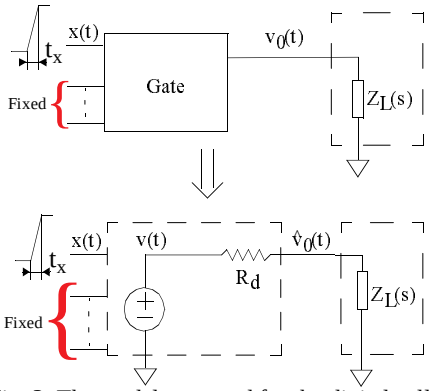


Fig. 3. The model proposed for the digital cells

This waveshape can be obtained by applying the actual gate output waveform, $v_0(t)$, to the inverse transfer function:

$$H^{-1}(s) = 1 + \frac{R_d}{Z_L(s)} \qquad (3)$$

Figures 4 and 5 show the $V_{id}(t)$ waveshapes obtained for this model given a capacitance load and an RC load, respectively. Also shown in Figures 4 and 5 are the saturated ramp approximations of

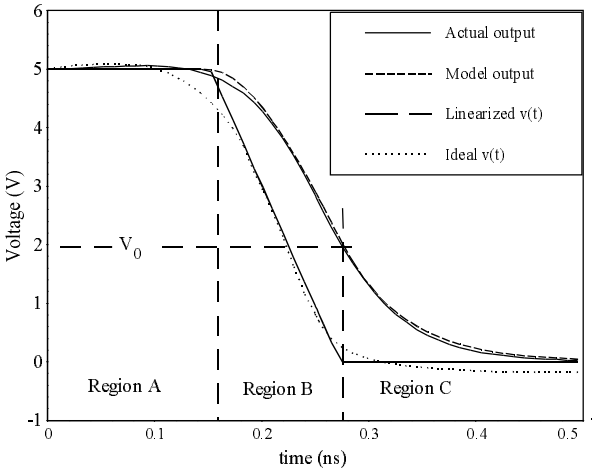


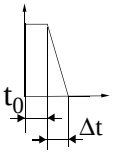
Fig. 4. Results obtained for an inverter with a 0.615pF load.

$V_{id}(t)$ and the response waveforms that they produce. From these examples it is apparent that the $V_{id}(t)$ waveshape is accurately modeled by a simple, linearized waveshape.

B. The model in use

Similar to the equations in (1), each gate is precharacterized by *k*-factor-like coefficients for the functions that define t_0 and Δt of the voltage source:

$$t_0 = f(t_{in}, C_L)$$
$$\Delta t = g(t_{in}, C_L) \qquad (4)$$



A single resistance value, R_d, for the pull-up or pull-down path of the gate is also calculated.

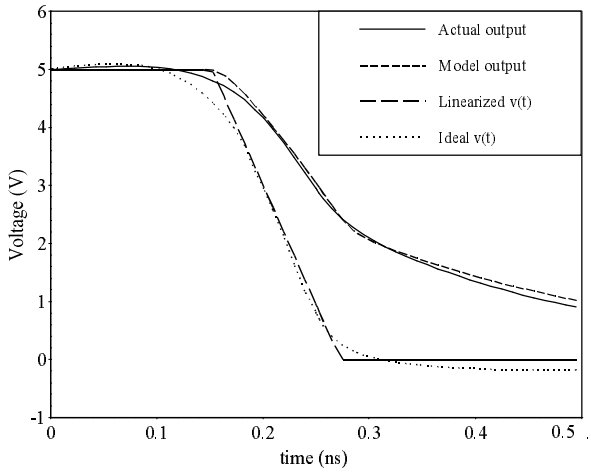


Fig. 5. Results obtained when an inverter drives a π-load with C_1=1.46pF, C_2=0.245pF, R=89.5Ω.

When the load is not purely capacitive, it was shown in [11] that a π-load represents a very accurate approximation of the CMOS RC-interconnect load. It was also shown that a 2nd order model is guaranteed to exist [11]. Since (4), as (1), is not valid for non-capacitive loads, we propose a method for mapping the π-circuit to a unique capacitance value which can then be used to determine (via (4)) the values of t_0 and Δt. This capacitance is similar to the *effective capacitance* (C_{eff}) described in [10]. Basically, we maintain the principle of equating the average current drawn by the effective capacitance and the π-circuit in the *active region*. We differ from the approach in [10] in our definition of the active region -- the time period over which the gate (the MOS transistors) can be naturally modeled as an average current. In [10], the active region was artificially terminated at the 50% point thereby ignoring the influence of the input waveform and load on the duration of the active region. Our model uses Region B from Fig. 4 as a more natural time period for the averaging process.

A procedure to compute the voltage source parameters (through C_{eff}) is:

1. Use min(C_{tot}, C_{max}) as an initial estimate for C_{eff}.
2. Compute Δt from (4) assuming $C_L = C_{eff}$.
3. Find the new C_{eff} by equating the average current drawn from the voltage source (Fig. 6) in Δt time (Region B) by C_{eff} and the π-load:

$$I_\pi = I_{C_{eff}} \tag{5}$$

These currents are given by:

$$I_\pi = \left[A\Delta t + \frac{B}{p_1}(1-e^{-p_1\Delta t}) + \frac{D}{p_2}(1-e^{-p_2\Delta t})\right]\frac{V_{dd}}{R_d(\Delta t)^2}$$
$$I_{C_{eff}} = \left[R_d C_{eff}\Delta t - (R_d C_{eff})^2\left(1-e^{-\frac{\Delta t}{R_d\cdot C_{eff}}}\right)\right]\frac{V_{dd}}{R_d(\Delta t)^2} \tag{6}$$

where

$$A = \frac{z}{p_1 p_2} \quad B = \frac{z-p_1}{p_1(p_1-p_2)} \quad D = \frac{z-p_2}{p_2(p_2-p_1)} \tag{7}$$

and z, p_1, p_2 are the zero and the two poles respectively of the transfer function (V_o(s)-V(s))/V(s).

4. Go to step 2 until convergence is reached.

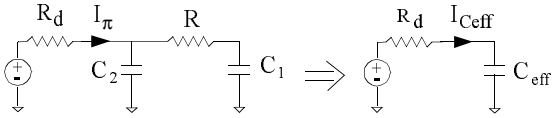


Fig. 6. The averaging of the load currents.

Steps 1 through 4 describe a simple fixed-point iteration procedure. Fig. 7 depicts a typical plot of the solutions of (5) for different values of Δt using $\Delta t = g(t_{in}, C_L)$ from (4). The intersection of these two curves represents the solution. From the graph we observe that a Newton-Raphson procedure to compute C_{eff} would converge within fewer iterations. For this reason we use N-R iteration in practice. We

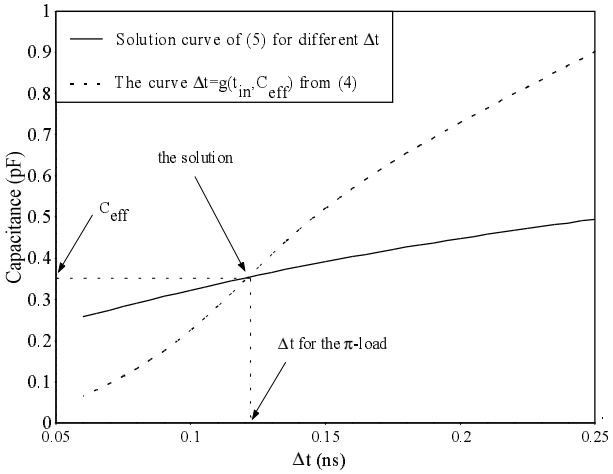


Fig. 7. Graphical representation of the iteration procedure for the result in Fig. 10

also note that by taking min(C_{tot}, C_{max}) as an initial estimate for C_{eff}, we never pass through an intermediate solution smaller than C_{eff}. We should point out that the C_{eff} from Fig. 5 is the same as C_L in Fig. 4. Therefore, the voltage source models in Figures 4 and 5 are identical, even though the response waveform are vastly different.

IV. COMPUTATION OF THE MODEL PARAMETERS

An implicit assumption in our model is that the behavior of a gate can be classified into three operating regions (shown explicitly in Fig. 4). Consequently, the model offers three degrees of freedom for a best fit of the experimental data.

A. The driver's resistance

Using the model in Fig. 3, we are modeling the last operating region (Region C in Fig. 4) as a linear resistor connected to ground (for a falling transition) or V_{dd} (for a rising transition). This is a refinement over switch-level simulators that use a linear resistor to model the gate over all operating regions. Our solution is more accurate due to the reduced swing in the output voltage over which we linearize the output resistance. We precharacterize R_d while making the following observations (exemplified for a falling transition):

- For purely capacitive loads, a larger load capacitance will drive the gate into Region C earlier (larger initial voltage for region C).
- The average output resistance increases with the increase in the initial voltage.

- The final portion of the "tail" (when the output voltage is less than some voltage, V_{min}) is of no importance for delay and output slope computation. Taking this portion of the waveform into consideration would artificially decrease the output resistance resulting in greater inaccuracy in the region of interest.
- Increasing R_d causes the output waveform to shift in a "pessimistic" direction.

We, therefore, fit the resistor value based on the largest load capacitance, C_{max}, encountered by the driver in an actual circuit.The model output voltage in region C is given by:

$$v_o(t) = V_0 \cdot e^{-\frac{t - t_0 - \Delta t}{R_d \cdot C_L}} \quad (8)$$

The two unknowns -- V_0 and R_d -- require us to provide two constraints. The first constraint is the Least-Mean-Squared (LMS) best fit of the actual data windowed between two time points (at which the output voltage equal V_0 and 10% of the V_{dd}, respectively). The other constraint is that there exists a set of model parameters that will fit the 20%, 50% points of the output waveform as well as V_0.

Finding a solution for the above system can be viewed as an iterative process (again, this procedure is described for a falling transition) during precharacterization:

1. Perform a SPICE analysis with the maximum allowable load capacitance to obtain the vector of the output voltage, v[n], and the corresponding vector of the discrete time points t[n] (n represents the n-th time point).
2. Choose an initial value for V_0 ($V_{dd}/2$ is a good choice).
3. Find the smallest k such that $v[k] < V_0$ and the smallest m such that $v[m] < 0.1V_{dd}$.
4. Compute R_d to obtain a least-squares fit of the data with (8).

$$R_d = \left(C_{L_{max}} \cdot \frac{\sum_{i \geq k}^{m} t[i] \cdot \ln \frac{V_0}{v[i]}}{\sum_{j \geq k}^{m} t^2[j]} \right)^{-1} \quad (9)$$

5. Using this value of R_d, compute t_0 and Δt in the manner described in the following subsection.
6. Find the smallest k such that $t[k] > t_0 + \Delta t$.
7. Go to step 4 until convergence is reached.

This procedure usually converges within two or three iterations. For the NAND gate in Fig. 7 an R_d of 201Ω was obtained. The response is compared with SPICE in Fig. 8.

We should point out that the delay approximation is fairly insensitive to the value of R_d as shown in Fig. 8. This is because the voltage source specifies the correct average current for Region B, for any R_d. Using a large or a small R_d will mainly impact the tail portion of the waveform (Region C).

B. The empirical characterization of the gate

One of the most obvious advantages of this model is the replacement of the gate output voltage (which grows further away from a digital shape with increasingly finer feature sizes) with a more digital waveform. To take full advantage of this feature of the model, we propose the replacement of the ubiquitous *k*-factor equations in (1)

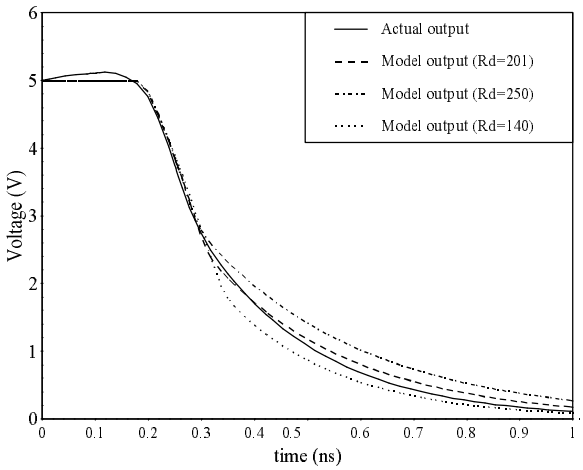


Fig. 8. Results obtained for a NAND gate driving a π-load with C_1=0.7pF, C_2=0.15pF and R=150Ω when the model is precharacterized using different resistance values.

with the empirical characterization of the voltage source parameters, t_0 and Δt, in equation (4).

The basic approach used to obtain the empirical equations remains unchanged -- perform SPICE runs for different input transition times and different load capacitances. However, instead of recording the output voltage parameters, t_d and t_r, we record parameters of the ideal voltage source voltage, $v_{id}(t)$. Unlike the output voltage, the ideal voltage source is not an observable waveform. Fortunately, it can be easily derived from the SPICE simulation itself. For a purely capacitive load:

$$V_{id}(s) = V_o(s) \cdot (1 + s \cdot R_d \cdot C_L) \quad (10)$$

which can be rewritten in the time domain as:

$$v_{id}(t) = v_o(t) + R_d C_L \frac{d}{dt} v_o(t) = v_o(t) + R_d i_{C_L}(t) \quad (11)$$

where $i_{C_L}(t)$ is the current through the load capacitance -- a quantity available from the SPICE simulation.

Although the ramp voltage can be characterized in the same manner as the output voltage in the *k*-factor equations in (1) (e.g. by the 50%-delay and the 10-90% slope) we prefer a slight modification. Firstly, we do not characterize the 50%-delay since it has no physical significance now. Secondly, we apply a correction to the t_0 and Δt obtained from the 10-90% levels of $V_{id}(t)$ so as to match the 20% and 50% levels of the output voltage. This correction, which provides an increased level of confidence in the new model precision, requires solving two nonlinear equations in two unknowns. This iterative computation during precharacterization is efficient since the t_0 and Δt of $V_{id}(t)$, which is a good approximation of the corrected waveform, serve as excellent initial values. Linearizing $V_{id}(t)$ by the straight line passing through the 20% and 80% points is not as accurate.

V. EXAMPLES

There are applications that require empirical models for complex cells (e.g. flip-flops, latches, etc.) and/or multiple inputs gates (e.g. n-inputs NAND gates). The complex cell case proved to be easily solvable by the method proposed in this paper. Fig. 9 presents the

result obtained using this model for a D-latch. The comparison is

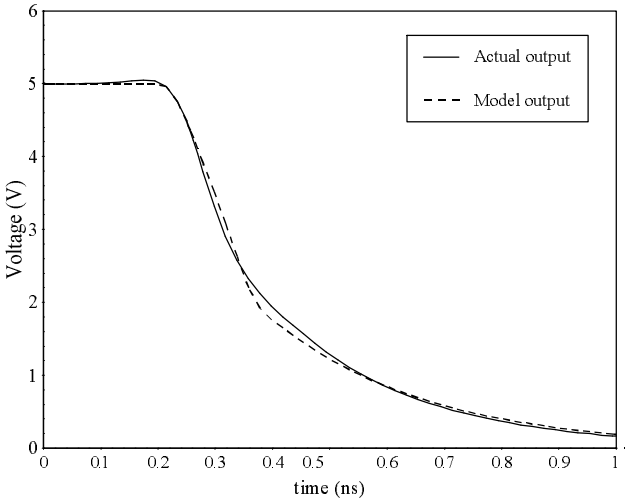


Fig. 9. Results obtained for a D-latch drving a π-load with C_1=1.46pF, C_2=0.245pF and R=89.5Ω.

between the actual SPICE result with an edge on CLK and that obtained with our model. Two C_{eff} iterations were required.

Accurate modeling of simultaneous switching is extremely difficult due to the ambiguity of the signal waveshapes and arrival times. With the model proposed here, however, we can easily bound the simultaneous switching response on both sides, and in the process demonstrate the sensitivity of the gate delay to the number of inputs switching. As an example, an empirical model was developed (following the steps outlined in Section IV) for a two-input NAND gate with both inputs tied together and for one input switching alone (with the other input set to the proper sensitization value). The voltage source waveshapes and the response waveforms, as they compare with SPICE, are shown in Fig. 10. Both models use the same pull-down resistance, but notice that for a falling transition the voltage source delay is smaller and the transition time is shorter for the single input transition as compared to the same input signal arriving at both inputs simultaneously. For the arrival of two signals with different waveshapes, the voltage source model for the two inputs tied together is a worst case response if the slower of the two edges is used in the empirical equation. Similarly, the single input transition response with the slower of the two transition times is an optimistic prediction for the gate delay.

VI. FUTURE WORK AND CONCLUSIONS

We have presented a delay modeling approach that works extremely well for the highly-resistive interconnect loads present in today's CMOS circuits. We observe that this approach accurately captures the non-digital behavior of the output waveforms. The parameters for this model can be precharacterized efficiently. We have also demonstrated the potential capabilities of this model for bounding the simultaneous switching delay. In addition, it should be noted that since the resistor for the gate model is fixed, one can completely precharacterize the RC loading and the gate in terms of dominant time constants following extraction. The only parameter which varies during timing analysis is the voltage source slope and delay. This facilitates the use of an extremely efficient delay equation for the gate and the interconnect. Using worst-case (or best-case) voltage source parameters one can also use this model to "estimate" the delay at a very high level of design such as floor planning.

This model provides the foundation for several directions of future work. First, we plan to provide more formal assertions regarding the bounds for simultaneous switching. Secondly, we plan to develop metrics and bounds so that this model can be used with

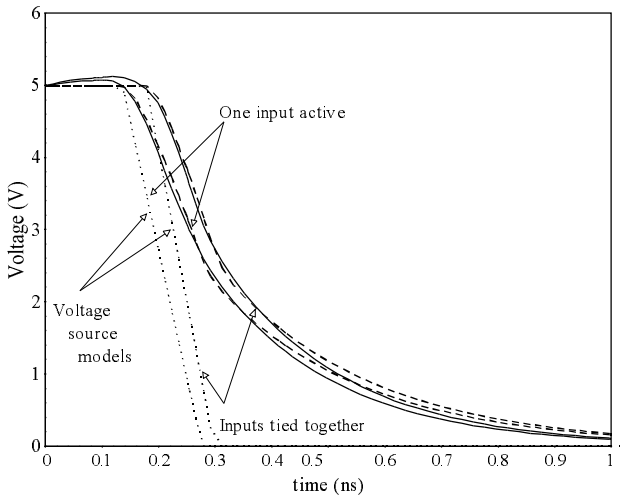


Fig. 10. Results obtained for a NAND gate driving a π-load with C_1=0.7pF, C_2=0.15pF and R=150Ω.

wiring estimates and loading estimates for high-level delay estimation. Finally, and perhaps most importantly, with this model the largest delay modeling error is now incurred when we attempt to apply the fan-out waveforms to subsequent stages. Since these waveshapes no longer appear digital, substantial error is incurred when we model them as linear. We will work on a more accurate, yet efficient, waveshape model which is compatible with this empirically-based delay methodology.

REFERENCES

[1] J. Rubenstein, P. Penfield, Jr., and M. A. Horowitz, "Signal delay in RC tree networks," *IEEE Trans. Computer-Aided Design*, vol. CAD-2, pp. 202-211, July 1983.

[2] J. L. Wyatt, Jr., "Signal delay in RC mesh networks," *IEEE Trans. on Circuits and Systems*, vol. CAS-32, no. 5, pp. 507-510, May 1985.

[3] W. C. Elmore, "The transient response of damped linear networks with particular regard to wide-band amplifiers," *J. Applied Physics*, vol. 19, no. 1, pp 55-63, Jan. 1948.

[4] N. Jouppi, "Timing analysis and performance improvement of MOS VLSI designs," *IEEE Trans. Computer-Aided Design*, vol. CAD-6, pp.650-665, July 1987.

[5] J. K. Ousterhout, "A switch-level timing verifier for digital MOS VLSI," *IEEE Trans. Computer-Aided Design*, vol. CAD-4, no. 3, pp. 336-349, July 1985.

[6] P.R. O'Brien and T.L. Savarino, "Modeling the driving-point characteristic of resistive interconnect for accurate delay estimation," *Proc. IEEE Intl. Conf. Computer-Aided Design*, November 1989.

[7] C.L. Ratzlaff, N. Gopal, and L.T. Pillage, "RICE: Rapid Interconnect Circuit Evaluator," *Proc. 28th ACM/IEEE Design Automation Conference*, June 1991.

[8] L.T. Pillage and R.A. Rohrer, "Asymptotic waveform evaluation for timing analysis," *IEEE Trans. Computer-Aided Design*, April, 1990.

[9] N. H. E. Weste and K. Eshraghian, *Principles of CMOS VLSI Design*, 2nd edition, Addison-Wesley Publishing Company, "Empirical Delay Models", pp. 213, 1992.

[10] C.L. Ratzlaff, S. Pullela, and L.T. Pillage, "Modeling the RC-interconnect effects in a hierarchical timing analysis," *IEEE Custom Integrated Circuits Conference*, May 1992.

[11] N. Gopal and L. T. Pillage, "Evaluation of on-chip interconnect using moment matching," *Proc. of the Intl. Conf. on Computer-Aided Design*, November, 1991.

[12] L.M. Brocco, S.P. Mccormick and J. Allen, "Macromodeling CMOS circuits for timing simulation," *IEEE Trans. Computer-Aided Design*, December 1988.