WAVEFORM APPROXIMATION TECHNIQUE IN THE SWITCH-LEVEL TIMING SIMULATOR BTS

Molin Chang, Wang-Jin Chen, Jyh-Herng Wang* and Wu-Shiung Feng

Department of Electrical Engineering National Taiwan University Taipei, Taiwan, R.O.C.

ABSTRACT

In this paper an accurate and efficient switch-level timing simulator is described. The high accuracy is attributed to a new waveform approximation technique, which includes delay estimation and slope estimation. Efficient delay and slope calculations are accomplished through a switch-level simulation instead of using a transistor-level simulation. A new approach for delay estimation is presented, and it models the delay behavior of an RC tree by two equations: a dominant delay equation and an offset delay equation. Both are derived by a special process to fit the surface built by experimental data measured from the actual delay behavior of a CMOS gate. The results show good agreement with SPICE.

1. INTRODUCTION

BTS (Binary-tree Timing Simulator), which is two to three orders faster than SPICE, is an event-driven switch-level timing simulator and performs more accurate waveform approximation during the transient state.

Most switch-level algorithms emphasized how to calculate the time constant of charging/discharging the load capacitance more accurately. There are many researches on this topic [1-3]. However, all of above can not offer us more accurate waveform information in transient state; we want to know not only whether the logic gate changes state or not, but also when the output voltage begins to change and how fast it will change. Therefore, the waveform approximation technique is divided into two parts that are the delay estimation and the slope estimation. The delay estimation tells us when the output begins to change and the slope estimation tells us how fast the output will change. An uncertain amount of overshoot, chiefly due to parasitic capacitors, will almost always be produced at the output node while an event is happening at the input. The width of overshoot is the keypoint; if it can be predicted well, and then the delay will be estimated accurately. Then, the slope relates closely to the RC time constant of the discharging/charging path. Lin and Mead [4] proposed an efficient method that can be implemented in a recursive way. Furthermore, another important feature of BTS is that the delay and slope calculations are considered with internal charges and charge sharing effects [5,6]. The Internal charges stored in the internal nodes of a MOS circuit will increase the

*National Center for High-Performance Computer 7, R&D Rd. VI, Science-Based Industry Park HsinChu, Taiwan, R.O.C.

delay time about 20% when the tested circuit is a five-input NAND gate with four fully charged internal nodes [7]. Therefore, the effect of internal charges should also be considered when the delay time is estimated.

The remainder of this paper is organized as follows. First, we describe the MOS model used in BTS in Section 2. Next, the method of waveform approximation is presented in Section 3. Then, the delay and the slope estimations are discussed in Sections 4 and 5, respectively. Finally, the simulation results are given in Section 6 and summary in Section 7.

2. MOS MODEL

The MOS model in BTS is composed of voltage-controlled switch, effective resistance R_{eff} and equivalent grounded capacitances. The transistor is *on* (the switch conducts) if and only if the gate voltage of the NMOS transistor is higher than its threshold voltage V_t. The turn-on effective resistor is distinguished by two cases: R_{on} (in steady state) and R_t (in transient state), because the MOS transistor (denoted by MOSt) has different response under different gate state. Therefore, the value of R_{eff} may be one of the three cases:

•	Infinity:	if V _{es} <	V

• R_{on} : if V_{gs} is high (in steady state)

• R_t : if V_{gs} changes from L to H (in transient state) The values of R_{on} and R_t depend on the physical parameters and the load capacitance, and R_t depends also on the slope of the signal at the gate.

3. WAVEFORM APPROXIMATION

The approximation work can be simplified if we cut off the overshoot and use a linear segment followed by an exponential tail to approach the falling (or rising) signal [8]. We use two equations as follows to plot the transient waveform.

•	for a rising signal			
<i>f</i> =	0.2t/T	for t<3T	or	
	1-0.4exp(-(t-3T)/2T)	for $t \ge 3T$		(1)
•	for a falling signal			
f=	I - (0.2t/T)	for t<3T	or	
	0.4 exp(-(t-3T)/2T)	for $t \ge 3T$		(2)

where T is half of the time spent by the signal between 90% (for a falling signal) or 10% (for a rising signal) and 50% of

the steady state. If the value of T can be obtained, the transient waveform will then be easily plotted.

The difference between the time when the output signal begins to change and the time when the input signal begins to change is defined as delay, which is denoted by D. The changing rate after the output begins to change is defined as slope, which is denoted by S.

4. DELAY ESTIMATION

4.1 Overshoot

Owing to the electrical characteristics of a MOS transistor, there are many parasitic capacitors existing inside a CMOS gate, e.g., Cgs, Cgd and so on. So the waveform of the drain of a MOSt depends not only the turn-on mechanism of MOSt but also the path formed by Cgd. The overshoot of output waveform, which can be treated as the excessive charge stored in the output node, is caused by the differential gate capacitor current. Observe that the amount of overshoot is determined by four factors as follows: (1) the slope S_i of input signal, (2) the size of C_{gd} , (3) the load capacitance Cl of output, and (4) the resistance R_p of discharging path in the N tree (or charging path in the P tree).

By analyzing some sample circuits using SPICE and varying the values of factors as mentioned above, we measure the data of delay time and then we can model the delay behaviors of CMOS gates by two equations.

(1) Dominant delay equation:

Cl is fixed, so this equation describes the relationship among delay, S_i , and R_p . It is easy to change S_i but R_p is not. Therefore, an alternative method is used. We increase the number of MOSt's in N tree circuit in order to change R_p discontinuously, and then R_p is replaced with N_p . In other words, we use the circuits such as inverter, two-input NAND gate, three-input NAND gate, and so on, as the primitive cases. The effect of internal charges that we probably meet in the actual circuits are extracted as an independent problem (see subsection 4.2).

For each primitive case, changing the input slope will produce a set of discrete two dimension curve, called NANDxcurve (x is the number of input). By collecting all the sets of data, we can plot a three-dimensional surface as shown in Fig. 1 and can use a hyperbolic surface (Eq. 3) to fit it.

$$D_p = (0.0292N_p + 0.369)(S_i + 0.3) + 0.12 \tag{3}$$

The deriving procedure is described as below:

Step 1: Use a straight line to fit a NANDx-curve in D_D -S_i plane, called curve α .

Step 2: Use a straight line to fit the curve, called *SLOPEy*curve (y is the value of input slope), in D_p - N_p plane, and then normalize this curve, called curve β , which is used to modulate the curve α in the direction of N_p -axis. Step 3 : D_p =(curve α)(curve β)+offset.

(2) Offset delay equation:

 N_p is fixed, so this equation describes the relationship among offset delay (an offset value with respect to D_p), S_i , and Cl. This equation is used for compensating the value of delay time calculated by the dominant delay equation, which does not consider the effect of the changing factor Cl. If N_p is adjusted, we obtain a set of surfaces. It means that we can obtain a discrete three-dimensional surface for each primitive case. The method for constructing this surface is the same as mentioned above. Similarly, we can also use a set of hyperbolic surfaces

$$D_0 = f(N_D)(0.293S_iC_l + 0.023) \tag{4}$$

to fit them, where $f(N_p)$ represents the coefficients that are the function of N_p . The surfaces when $N_p=1$ are shown in Fig. 2, which include the surfaces built by experimental data and derived approximate surface.



4.2 Internal Nodes

The delay due to the internal charges can be calculated approximately as

$$dt = Q/I_{av} = (Q/V_s)2R \tag{5}$$

where I_{av} is the average current, V_s is the voltage swing, R is the effective resistance of the conducting path and Q is the charge stored in the internal nodes. More than one internal nodes may be going to charge or discharge in the series-parallel tree, and these nodes must be taken into account when calculating the switching delay. Thus, Eq. 5 is rewritten as

$$D_l = \Sigma 2R_i(Q_l/V_s) \tag{6}$$

where R_i is the effective resistance of internal node v_i with respect to ground, Q_i is the charge stored in the internal node v_i .

After all, the total delay is summed up by the delay times caused by the effect of overshoot and the internal nodes, including the charge sharing effect [5,6].

$$D_{total} = D_D + D_0 + D_I \tag{7}$$

5. SLOPE ESTIMATION

If the output waveform can be treated as a simple RC waveform, then the parameter T in Eqs. 1 and 2 can be calculated by the equation: $T=(t_{50\%} - t_{10\%})/2=0.294$ RC. In BTS we defined slope as the time spent by the signal voltage dropping one volt, i.e. in units of time/volt, and then T=S when Vdd=5V. Therefore, T=S=0.294RC.

The equivalent RC time constant of active tree can be computed by the equations as described in reference 4 and implemented by a recursive algorithm while traversing the whole RC tree. Furthermore, two effects, called *non-active tree effect* and *bottle-neck effect*, should also be considered together.

5.1 Non-active tree effect

Because the output slope of falling signal is affected by not only the N tree but also some internal nodes connected to output node in the P tree, the algorithm for computing slope should consider both and then estimate the total effect. When estimating the slope, the internal nodes in the *non-active tree* should be considered, and its influence should be added into the component obtained from calculating the RC time constant of the active tree. For example, if the output state is changed from HIGH to LOW in the circuit as shown in Fig. 3(a), there is at least one discharging path existing in the *active tree* (the N tree in this case) and no charging path in the *non-active tree* (the P tree). In this case, the effect of the nodes v_5^* and v_6^* should be added when estimating the slope, and both internal nodes can be viewed as a charge supplier that can supplement the charge loss at the load capacitor.

5.2 Bottle-neck effect

A bottle-neck always exists in the discharging path and will form the highest barrier to prevent the discharging current flowing on it. Therefore, the discharging rate of the output node is dominated by the bottle-neck. However, it is a complicated work to find the bottle-neck of the charging/discharging (C/D) path because the C/D path is changed dynamically depending upon the input patterns. Furthermore, parallel connections in the C/D path will increase the difficulty of this problem. Finding the location of bottle-neck becomes a major work when estimate the bottle-neck effect. In general, we can not expect that the bottle-neck is a MOSt at most of time except that the C/D path is a pure series connection.

The transistors in the bottle-neck should be replaced with Rt and all others should be replaced with Ron while calculating the slope value. From simulation results by SPICE, we find that all MOSt's in transient state not located in the bottleneck can not affect the slope value of output waveform.



Fig. 3: A series-parallel circuit and its corresponding MTB tree.

6. RESULTS

This method has been tested extensively for basic modules such as counters, decoders, adders, and ALU's. The CPU time comparisons are summarized in Table 1. An one-cluster circuit, also using the circuit as shown in Fig. 3(a) as an example, is simulated by using SPICE and our timing simulator BTS. The results are compared as shown in Fig. 4. The bold solid lines are the results obtained from our simulator. Five input patterns are applied to this circuit, which are case 1: (/,0,0,0,1,1,1), case 2: (1,0,0,0,/,1,1), case 3: (1,0,0,0,1,/,1), case 4: (1,1,0,0,1,1/), and case 5: (1,1,/,0,1,/,1), and the associated output waveforms are labeled A, B, C, D and E, respectively. Note that '/' represents the ramp input signal with the rising time 1ns from LOW to HIGH. The waveform E is extracted independently for its different slope, and compared with the waveform C to distinguish its difference. There are small errors presented in the simulation results because the target circuit is simply an one-stage circuit.

7. SUMMARY

An accurate waveform approximation technique is proposed, which is achieved by the new approach of delay estimation and the modified slope estimation. The new approach of delay estimation improves the previous version of BTS that converted the overshoot effect to the turn-on-time of MOSt (the value of V_T was shifted to 3.1V), and can offer a better adaptability for a wide range of circuit and input specification. For each different fabrication process, the equations for delay estimation are derived only once. The deriving procedure is a simple and quick work because it can be achieved only by a few samples. Of course, this procedure can also be aided by a program on computer. The modified slope estimation also increases the accuracy of the transient waveform prediction, including under some special circumstances.

Circuit	MOS	CPU time on PC (DX4-100), secs		Speed ratio	Primary input
	no.	BTS	Pspice		event no.
complex gate (Fig. 1(a))	12	0.11	2.53	0.043	2
inverter chain (100 stages)	200	0.28	241.18	0.0012	1
74138	88	0.33	102.22	0.0032	11
7483	258	0.99	774.64	0.0013	13
74381	584	1.10	1670.98	0.00066	14

Table 1: Comparisons between BTS and Spice



Fig. 4: The simulated waveforms of the circuit as shown in Fig. 3(a). Bold line: BTS. Light line: SPICE

8. REFERENCES

R. E. Bryant, "A Switch level model and simulator for MOS digital systems," IEEE Computers, vol. C-33, pp. 160-177, 1984.
C. J. Terman, "RSIM - A Logic-Level Timing Simulator," Proceedings of the IEEE International Conference on Computer Design, New York, pp. 437-440, November, 1983.

[3] J. Rubinstein, P. Penfield, and M. A. Horowitz, "Signal delay in RC tree networks," IEEE Trans. on Computer-Aided Design, vol. CAD-2, NO. 3, pp.202-211, 1983.

[4] T. M. Lin, and C. A. Mead, "Signal delay in general RC networks," IEEE Trans. on Computer-Aided Design, vol. CAD-3, No.4, pp.331-349, 1984.

[5] Molin Chang, S,-J Yih and Wu-Shiung Feng, "Algorithm based on modified threaded binary tree for estimating delay affected by internal charges in CMOS gates", Electronics Letters, Vol. 32, No. 20, pp. 1877-1879, 26th September 1996.

[6] Molin Chang, S,-J Yih and Wu-Shiung Feng, "Recursive algorithm for calculating effective resistances in RC tree", Electronics Letters, Vol. 33, No. 2, pp. 131-133, 16th January 1997.

[7] J. H. Wang, Molin Chang, and W. S. Feng, "Binary-tree timing simulation with consideration of internal charges". IEE Proceed-ings-E, vol. 140, No.4, pp. 211-219, July 1993.

[8] F.C.Chang, C.F.Chen, and P.Subramaniam, "An accurate and efficient gate level delay calculator for MOS circuits," Proceedings of 25th ACM/IEEE conference on Design automation, Ana heim, CA, USA, pp.282-287, 1988.