A Modeling Technique for CMOS Gates

Alexander Chatzigeorgiou, *Student Member, IEEE*, Spiridon Nikolaidis, *Member, IEEE*, and Ioannis Tsoukalas

***Abstract*— In this paper, a modeling technique for CMOS gates, based on the reduction of each gate to an equivalent inverter, is presented. The proposed method can be incorporated in existing timing simulators in order to improve their accuracy. The conducting and parasitic behavior of parallel and serially connected transistors is accurately analyzed and an equivalent transistor is extracted for each case, taking into account the actual operating conditions of each device in the structure. The proposed model incorporates short-channel effects, the influence of body effect and is developed for nonzero transition time inputs. The exact time point when the gate starts conducting is efficiently calculated improving significantly the accuracy of the method. A mapping algorithm for reducing every possible input pattern of a gate to an equivalent signal is introduced and the "weight" of each transistor position in the gate structure is extracted. Complex gates are treated by first mapping every possible structure to a NAND/NOR gate and then by collapsing this gate to an equivalent inverter. Results are validated by comparisons to SPICE and ILLIADS2 for three submicron technologies.**

***Index Terms*—CMOS gates, modeling, simulation, timing analysis.**

I. Introduction

It has been extensively pointed out that with shrinking device dimensions and increasing number of transistors on integrated circuits in the submicron era, the difficulty in performing accurate and fast simulations of these circuits is increasing. In contrast to numerical approaches for calculating propagation delay and power dissipation, analytical methods can offer a significant speed improvement, assuming that accurate models which can describe the behavior of transistor structures exist. Many efforts have been made for modeling the operation of the CMOS inverter which have resulted in accurate analytical expressions for output waveform, propagation delay, and power consumption [1]–[3]. However, the field of CMOS gates is still unexplored because of the intrinsic difficulties in the analysis of gates with multiple nodes and inputs.

Generally there are two approaches in modeling CMOS gates. The first one, which corresponds to the generalization of the inverter model, is based on a fully mathematical analysis of the gate structure. This approach leads to high mathematical complexity and, thus, the requirements for computational power are significantly increased. In addition, some approximations have to be applied in order to solve the system differential equations of complex structures such as the transistor chain, reducing the overall accuracy of the method. In this way, Kang and Chen [4] used linear approximations for the output voltage waveform of the transistor chain trying to model the propagation delay in domino gates and only step inputs and long-channel devices were considered. Cherkauer and Friedman [5] performed their analysis using a simplified long-channel model and applying step inputs in order to optimize channel widths for low power consumption. Each of the nonsaturated devices was replaced by an effective resistance which was calculated assuming negligible body effect.

Shih and Kang in [6] presented a fully mathematical solution of general MOS circuit primitives and in [7] a tool named ILLIADS has been developed on this solution. However, this method presents high complexity and is based on quadratic form models such as the Shichman-Hodges model and, therefore, depends on their deficiencies. Moreover, serially connected transistors are collapsed to an equivalent transistor using a simple transconductance reduction, resulting in limited accuracy. An improved method has been presented in [8] where current expressions for short-channel devices are transformed to the required quadratic form expressions. Their analysis requires the solution of nonlinear algebraic equations in order to obtain the time point of region crossings, thereby reducing the efficiency of the method.

The second approach in CMOS gate modeling utilizes the well-established theory of inverters and research has focused on the development of sophisticated methods for collapsing a CMOS gate to an effective equivalent inverter, whose output response will reflect accurately that of the gate. The accuracy of this approach is comparable to that of the previous one, while the mathematical complexity is kept low. According to this approach, Sakurai and Newton [9] developed their analysis for the CMOS inverter and extension to gates was made by a delay degradation factor. Nabavi-Lishi and Rumin [10] presented a semiempirical method for collapsing gates to an inverter model, however, only NAND/NOR structures were considered. The proposed model employs empirical constants which are difficult to extract and for practical cases ends up in the conventional n-times transconductance reduction for an n-transistor chain, resulting in limited accuracy. In the same way, Daga *et al.* [11] developed their analysis for an inverter macro-model and gates were treated by defining an equivalent drivability factor using simplified assumptions for the operation of the transistors in the chain.

Recently, in [12]–[14], many of the characteristics of complex structures, such as the transistor chain, have been pointed out from a macromodeling point of view. However, poor

Manuscript received February 23, 1998; revised August 13, 1998. This paper was recommended by Associate Editor E. Bracken.

A. Chatzigeorgiou is with the Computer Science Department, Aristotle University of Thessaloniki, 54006 Thessaloniki, Greece (e-mail: achat@skiathos.physics.auth.gr)

S. Nikolaidis is with the Department of Physics, Aristotle University of Thessaloniki, 54006 Thessaloniki, Greece.

I. Tsoukalas is with the Computer Science Department, Aristotle University of Thessaloniki, 54006 Thessaloniki, Greece.

Publisher Item Identifier S 0278-0070(99)03095-X.

theoretical support is offered for the calculation of the effective chain transconductance and the starting point of conduction of a gate, while the same ramp input is applied to all inputs. The parasitic behavior of the transistor chain is not discussed and only a subgroup of complex gates is examined.

In this paper, a method for modeling CMOS gates by an equivalent inverter is proposed. Key points in the behavior of CMOS gates are modeled analytically in order to improve the accuracy of the final equivalent inverter, whose transistor's widths are calculated efficiently taking into account the mode of operation of the transistors in the gate. Such key points are the starting point of the gate conduction, which has a significant impact on the output waveform, the form of the internal node voltages, the parasitic behavior of the chain, and the weight of each transistor position in the chain. Finally, a very efficient algorithm for reducing every possible input pattern of a gate to an effective single equivalent input that can be applied to an inverter model, is introduced.

In order to obtain the output voltage evolution of CMOS gates, the serially and parallel connected transistors have to be replaced by an equivalent one. The technique which is used for the replacement of serially connected transistors by an equivalent one is presented in Section II, while in Section III the time point when such a structure starts conducting is calculated. The parasitic behavior of the chain in gate operation is analyzed in Section IV and the input mapping algorithm is described in Section V. Parallel transistors are discussed in Section VI. The application of all results to complex gates is examined in Section VII and implementation details of the proposed technique are described in Section VIII. Finally we conclude in Section IX.

II. Transistor Chain Model

In order to analyze the operation of the transistor chain when it is conducting, let us consider the circuit in Fig. 1(a), where the parasitic drain/source node capacitances are also shown, and assume that an input ramp with transition time τ is applied to the gates of all transistors in the chain

$$V_{\text{in}} = \begin{cases} 0, & t \le 0 \\ \dfrac{V_{DD}}{\tau} \cdot t, & 0 < t \le \tau \\ V_{DD}, & t > \tau. \end{cases} \tag{1}$$

The parasitic behavior of the chain will be discussed later.

The α-power law model proposed in [2], which takes into account the carrier velocity saturation effect of short-channel devices, is used for the transistor currents

$$I_D = \begin{cases} 0, & V_{GS} \le V_{TN}\text{: cutoff region} \\ k_l(V_{GS} - V_{TN})^{a/2} V_{DS}, & V_{DS} < V_{D\text{-}SAT}\text{: linear region} \\ k_s(V_{GS} - V_{TN})^{a}, & V_{DS} \ge V_{D\text{-}SAT}\text{: saturation region.} \end{cases} \tag{2}$$

where $V_{D\text{-}SAT}$ is the drain saturation voltage [2], k_l, k_s are the transconductance parameters which depend on the width to length ratio of a transistor, α is the carrier velocity saturation index, and V_{TN} is the threshold voltage which is expressed by its first-order Taylor series approximation around $V_{SB} = 0.2V_{DD}$ as

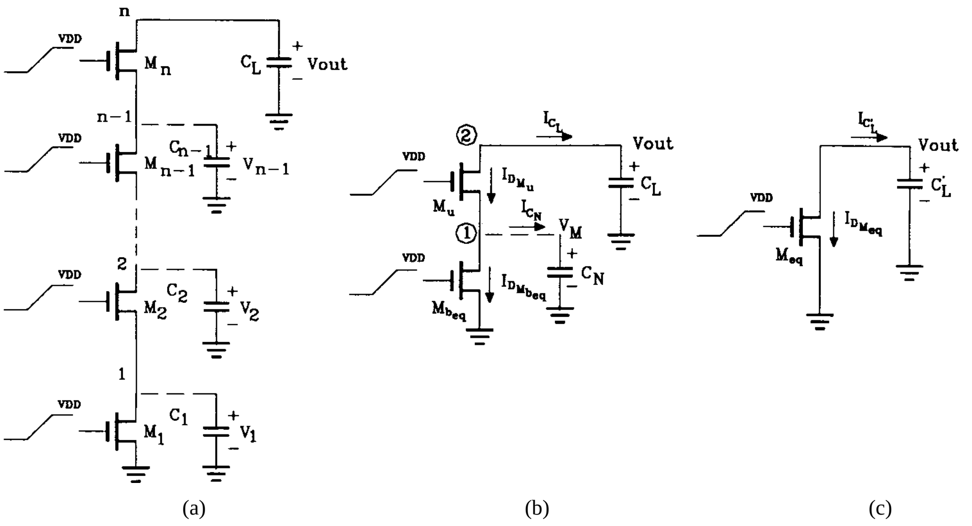
$$\begin{aligned} \tilde{V}_{TN} = V_{TN}|_{V_{SB}=0.2V_{DD}} + (V_{TN})'|_{V_{SB}=0.2V_{DD}} \\ \cdot (V_{SB} - 0.2V_{DD}) = \theta + \delta V_{SB} \end{aligned} \tag{3}$$

where V_{SB} is the source-to-substrate voltage. The Taylor series is calculated at $V_{SB} = 0.2V_{DD}$ since this voltage was found to lie close to the midpoint of the voltage swing of the source node of the top-most transistor which is of primary importance to the proposed analysis.

While the input is applied and assuming that the internal node capacitances are initially discharged, the top-most transistor in the chain (M_n), begins its operation in saturation mode and then enters the linear region when $V_{DS} = V_{D\text{-}SATN}$. The rest of the transistors operate in the linear region without ever leaving this region [5]. For the time interval during which the top-most transistor is in saturation and the input is rising, its current and, consequently, the voltages at the internal nodes are increasing. When the input reaches V_{DD} and until the top-most transistor exits saturation, its current and, therefore, the internal node voltages remain constant. That is because a further increase of the internal node voltages would decrease the gate-to-source voltage (V_{GS}) of the top-most transistor and, therefore, its current, leading in a decrease of the node voltages. On the other hand a decrease of the node voltages would increase the V_{GS} of the top-most transistor and, consequently, its current, leading in an increase of the internal node voltages. Therefore, these voltages remain constant until the top-most transistor exits saturation. During this time interval the parasitic currents due to drain/source node capacitances and gate-to-drain/source coupling capacitances are eliminated because the voltages at the corresponding nodes remain constant. Therefore, during this state, which is known as the "plateau" state [4], the same current flows through all transistors in the chain. According to the previous analysis, the plateau state is apparent only for fast input transitions (Fig. 2). Fast and slow inputs are determined according to the position of the time point t_2, when the top transistor in the chain exits saturation: in case $t_2 < \tau$, the input is slow, otherwise it should be considered fast. It should be mentioned that the previous analysis about the plateau voltage ignores channel-length modulation which forces the current of the top-most transistor to depend on its drain-to-source voltage, even when the top-most transistor is in saturation. However, its effect is insignificant and the validity of the above discussion has been proved by simulation results.

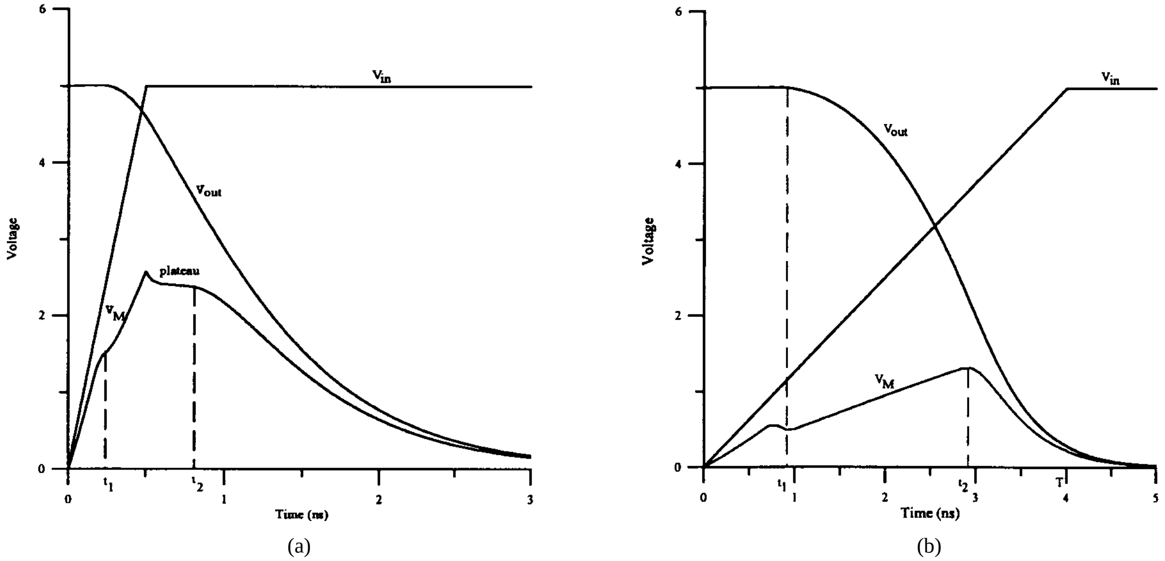
In order to calculate the plateau voltage at the source of the top-most transistor of the chain, V_p, let us consider the circuit of Fig. 1(a). Although the analysis here refers to fast input ramps where the plateau state appears, the derived results are also valid for slow inputs. A first approximation is used for the width $W_{b_{eq}}$ of the equivalent transistor $M_{b_{eq}}$ in Fig. 1(b), which replaces all nonsaturated transistors and is given by

$$\frac{1}{W_{b_{eq}}} = \frac{1}{W_1} + \frac{1}{W_2} + \cdots + \frac{1}{W_{n-1}}. \tag{4}$$



(a) (b) (c)

Fig. 1. (a) Complete transistor chain, (b) two transistor equivalent circuit, and (c) single equivalent transistor model.



(a) (b)

Fig. 2. Output and source voltage waveform of the top-most transistor of a transistor chain for (a) fast and (b) slow input ramp.

The plateau voltage V_p occurs at the end of the input ramp $(V_{\rm in} = V_{DD})$ when the current ceases to increase. Thus, V_p can be calculated by setting the saturation current of the top transistor (M_u) equal to the current of the bottom transistor $(M_{b_{\rm eq}})$ which operates in linear mode

$$k_{s_u}(V_{DD} - \theta - (1+\delta)V_p)^a = k_{l_{b_{\rm eq}}}(V_{DD} - V_{TO})^{a/2}V_p. \quad (5)$$

The above equation can be solved with very good accuracy using a second order Taylor series approximation.

In the following analysis all internal nodes of the chain are considered to be discharged at time $t = 0$. In case some of the internal nodes are initially charged (trapped charges), the output waveform which will result by the proposed method, should be appropriately shifted [4], since the charges in the internal nodes may cause an additional delay in the output response.

In addition, the source voltage of the top transistor in the chain V_M is considered linear for the interval between time t_1, where the chain starts conducting and time τ (fast inputs) or time t_2 (slow inputs) where the top transistor exits saturation. This observation is based on SPICE simulations and leads to highly accurate results (Fig. 2). Time points t_2, t_1 are calculated later in this section and in the Section III, respectively. Since time t_1 and $V_M[t_1]$ are known (see Section III) and for fast inputs the plateau voltage occurs at time τ, the slope of V_M is also known. For slow inputs the plateau voltage cannot be calculated as previously. However, it has been found that when the output load is sufficiently increased (which corresponds to the case of a fast input ramp), the slope of V_M remains almost the same (Fig. 3). Therefore, considering a larger load capacitance, V_p would occur at time $t = \tau$ and would be calculated as previously, by (5). The independence of V_p on the load capacitance which is required in order for the previous proposition to be valid, is obvious from (5). In this way the slope of V_M for the case of slow inputs can be obtained.

It is obvious that a single equivalent transistor will have the same output response with the complete chain if it successfully manages to reproduce the combined behavior of the

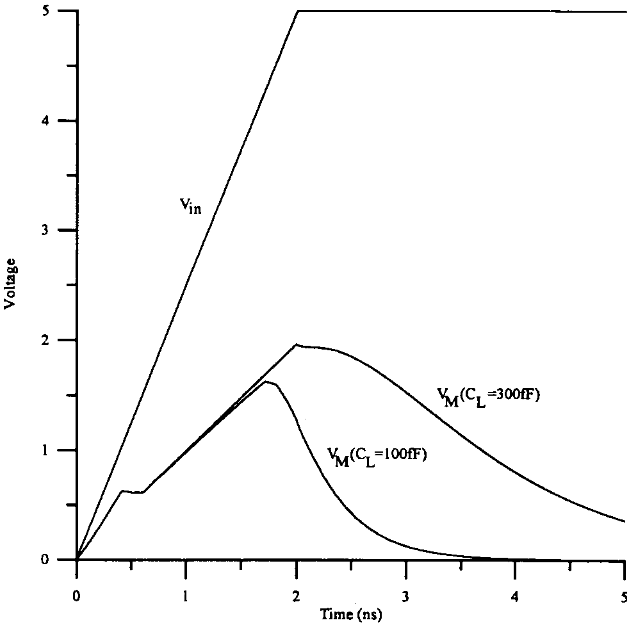


Fig. 3. Source voltage waveforms of the top-most transistor in a chain for the same input and different output loads.

nonsaturated devices with the dual operation of the top-most transistor, in saturation and the linear region. For the time interval where the top transistor in the chain is saturated, the current through that transistor is the bottleneck for the current that is flowing through the chain [5]. Therefore, in order to obtain the width $(W_{\rm eq})$ of the single equivalent transistor $(M_{\rm eq})$ [Fig. 1(c)] the currents through transistor M_n of the complete chain and transistor $M_{\rm eq}$ should be set equal

$$\begin{aligned} I_{M_n} =& I_{M_{\rm eq}} \Rightarrow P_s \frac{W_n}{L}(V_{\rm in} - \theta - (1+\delta)V_M)^a \\ =& P_s \frac{W_{\rm eq}}{L}(V_{\rm in} - V_{TO})^a. \end{aligned} \tag{6}$$

The above equation can be solved for several values of t yielding corresponding $W_{\rm eq}$ values. For the time interval $[t_1, t_2]$ during which the top transistor in the chain operates in saturation, $W_{\rm eq}$ plotted against time has the form of Fig. 4.

In order to find an average effective value for $W_{\rm eq}$ time point t_2 has to be calculated. This is achieved by solving the following differential equation at the output node:

$$C_L \frac{dV_{\rm out}}{dt} = -I_{M_n} = -k_s(V_{\rm in} - \theta - (1+\delta)V_M)^a. \tag{7}$$

The above equation can be solved since time point t_2 occurs in the region where V_M is linear (slow inputs) or has reached the plateau voltage (fast inputs).

Time t_2 is calculated by equating the drain saturation voltage to the actual drain-to-source voltage of transistor M_n

$$V_{D\text{-}SATN}[t_2] = V_{\rm out}[t_2] - V_M[t_2]. \tag{8}$$

It must be mentioned that in the estimation of t_2, the influence of the parasitic short-circuit current of a parallel pMOS transistor structure, which is the case of CMOS gates, has been neglected. This parasitic current acts as an additional charge which has to be removed through the chain, resulting in an extension of the saturation region and thus affecting the effective value of $W_{\rm eq}$ in this region. In order to obtain higher accuracy in the estimation of t_2, the expression of the pMOS current should be inserted in (7), increasing the mathematical complexity of the proposed method.

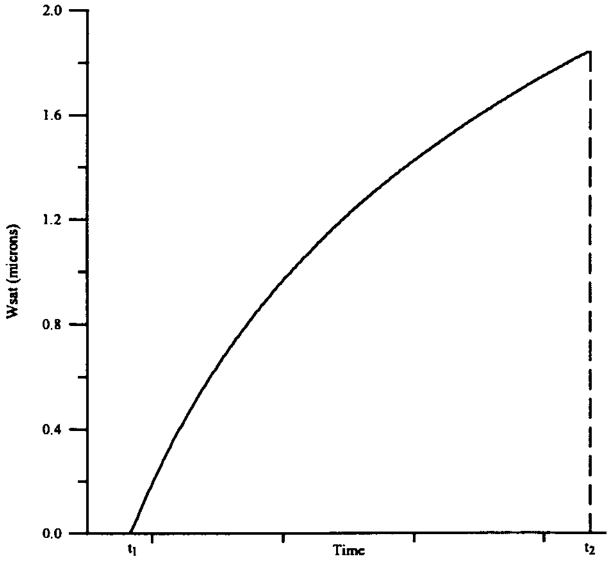


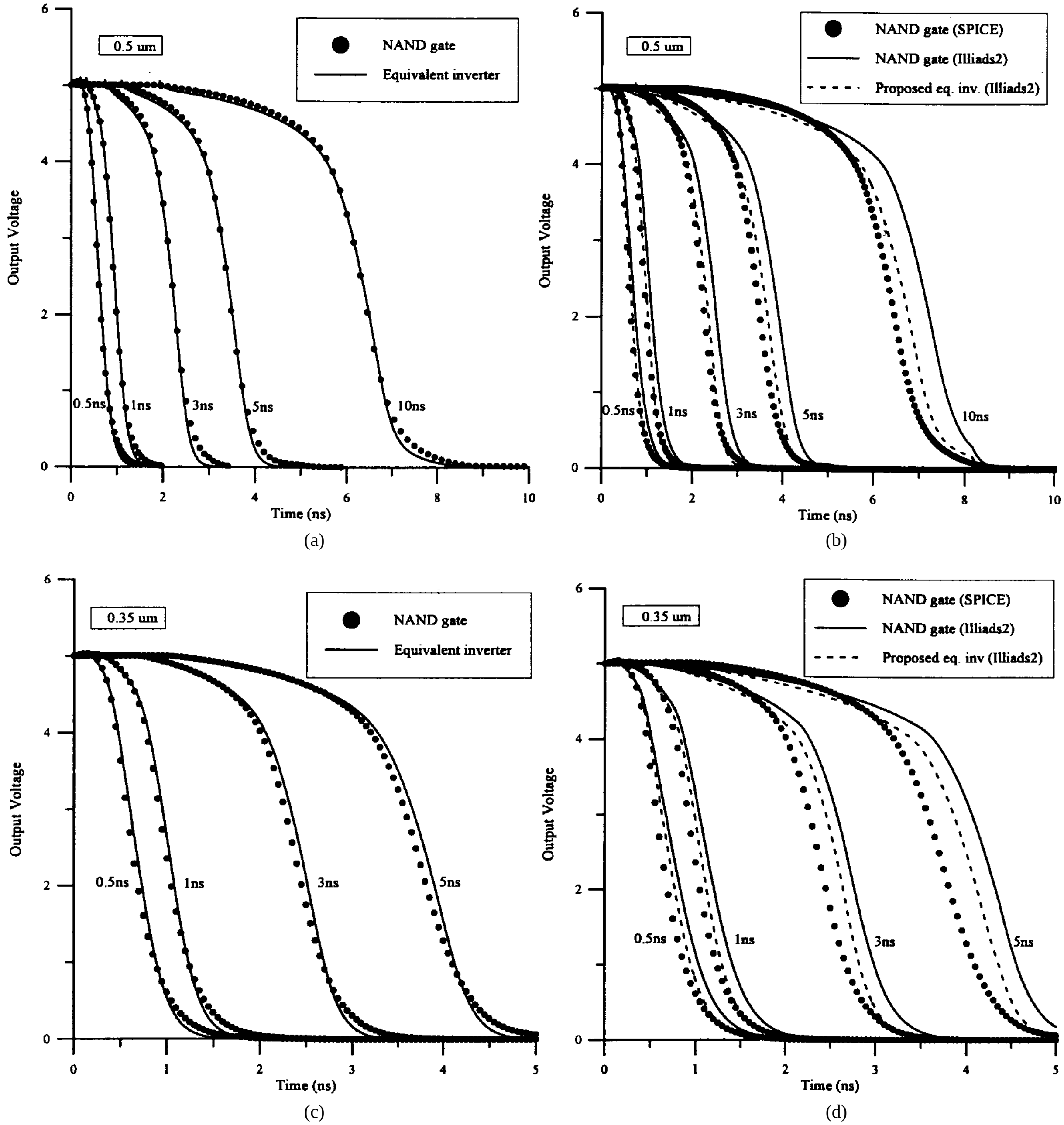
Fig. 4. Single transistor equivalent width for the saturation region.

However, for simplicity, t_2 is calculated by (7) and (8) and the effective value of $W_{\rm eq}$ is selected in such a way, so that the errors which are introduced by the underestimation of the bound of the saturation region be compensated. It has been observed by SPICE simulations that the appropriate value of $W_{\rm eq}$ depends on the input waveform slope and the load capacitance. This is reasonable since the short-circuit current increases as the input transition time increases and as the output load capacitance decreases [15]. A very good approximation for $W_{\rm eq}$, which has been found to be valid for a wide range of input slopes and load capacitances, is to calculate its value from (6) at $t = t_2$ for fast inputs and at $t = (t_1 + 3.3t_2)/4$ for slow inputs, where t_1 is the time when the transistor chain starts conducting and is calculated in Section III. The calculated value of $W_{\rm eq}$ in this region of operation will be referred to as $W_{\rm sat}$ and was found to be very accurate for the modeling of CMOS gates.

It should be noted that for gates which have no short-circuit current such as dynamic logic with nonoverlapping precharge and evaluation phases, $W_{\rm sat}$ should be calculated at $t = (t_1 + t_2)/2$ [16].

When all transistors operate in the linear region, the transistor chain can be considered as a voltage divider with a uniform distribution of the output voltage among all drain/source nodes. According to this, the width of the equivalent transistor for this region can be calculated as $W_{\rm lin} = (W/n)$ in case of nontapered transistor chains.

Consequently, since the effective transistor width for each operating condition is known, the chain can be modeled by a single equivalent transistor whose width from time t_1 to time t_2 is $W_{\rm sat}$ and for the rest of the time equal to $W_{\rm lin}$. However, since the aim was to provide an equivalent width



(a)

(b)

(c)

(d)

Fig. 5. (a) and (c). Output waveform comparison between the complete gate and the equivalent inverter for several technologies and input transition times. (b) and (d). Illiads2 improvement using the proposed equivalent inverter.

that would match the existing inverter models, the above two width values should be efficiently merged into one. This can be accomplished by calculating the fraction of charge (Q_{sat}) that is discharged to ground during the time in which the top transistor in the chain operates in saturation over the total charge $(Q_{\text{total}} = C_L V_{DD})$ that is stored initially in the output load and has to be discharged. The output voltage when the top transistor exits saturation, $V_{\text{out}}[t_2]$, is known from (7) and (8) and Q_{sat} can be calculated as

$$Q_{\text{sat}} = Q_{\text{total}} - Q[t_2] = C_L V_{DD} - C_L V_{\text{out}}[t_2]. \quad (9)$$

A saturation coefficient c_{sat} can now be calculated as

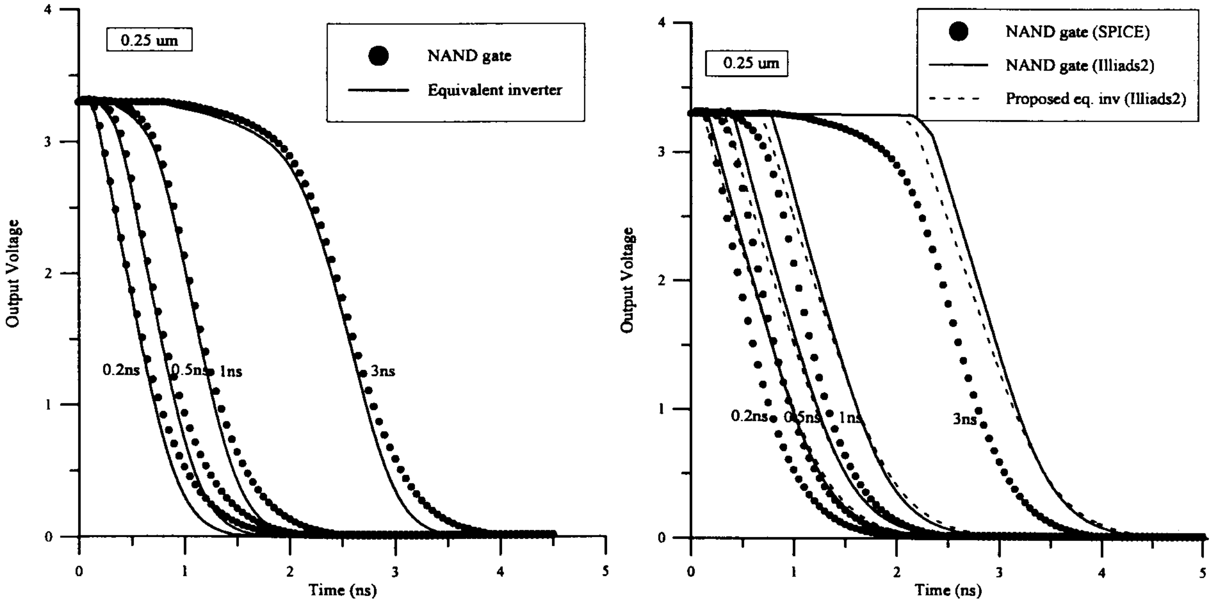
$$c_{\text{sat}} = \frac{Q_{\text{sat}}}{Q_{\text{total}}}. \quad (10)$$

Consequently, the corresponding coefficient when all transistors operate in linear mode, is equal to $c_{\text{lin}} = 1 - c_{\text{sat}}$.

Since the calculated coefficients act as the "weight" of each mode of operation on the overall output voltage temporal evolution, the width of the single equivalent transistor can be calculated as

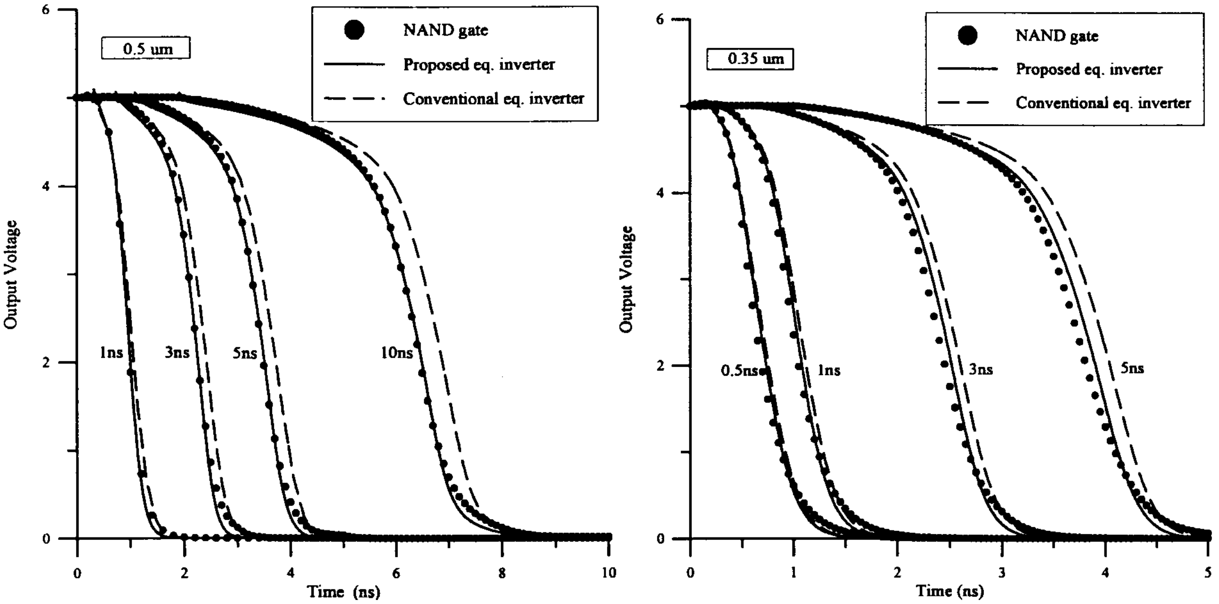
$$W_{\text{eq}} = c_{\text{sat}} \cdot W_{\text{sat}} + c_{\text{lin}} \cdot W_{\text{lin}}. \quad (11)$$

The output waveform of a four-input NAND gate for three submicron technologies (0.5-μm HP, 0.35-μm HP and 0.25-μm) and $C_L = 0.1$ pF is compared to that of the proposed equivalent inverter for several input transition times and is shown in Fig. 5(a), (c), and (e). The input transition time to which each output corresponds is also shown. The



(e) (f)

Fig. 5. (*Continued*). (e) Output waveform comparison between the complete gate and the equivalent inverter for several technologies and input transition times, (f) Illiads2 improvement using the proposed equivalent inverter.



(a) (b)

Fig. 6. Output waveform comparison between the complete gate and the equivalent inverter derived according to the proposed method and in the conventional way.

parallel transistors of the gate have been replaced by an equivalent one with its transconductance multiplied by the number of transistors. As it can be observed, the proposed method presents very good accuracy for all three submicron technologies. In Fig. 6 the output waveform of a four-input NAND gate is compared to that of an equivalent inverter whose width is calculated according to the proposed method and in the conventional way (where the transconductance of the equivalent transistor is reduced by the number of transistors in the chain) for two technologies (0.5 and 0.35 μm) and for several input transition times. The superiority of the proposed method compared to the conventional equivalent inverter is obvious.

In Fig. 5(b), (d), and (f), output waveform results from a widely used dynamic timing simulator, ILLIADS2 [17], are also shown and compared to the actual output waveform of the NAND gate which is obtained using SPICE. In the same plots the output waveform of an equivalent inverter whose transistor widths are calculated according to the proposed method and which is simulated using ILLIADS2 is also shown. It is clear that the improvement that is gained when the proposed additional step is added in order to calculate more

TABLE I
PROPAGATION DELAYS (ns) AND PERCENTAGE ERRORS FOR THE PROPOSED METHOD, ILLIADS2 (GATE) AND ILLIADS2 USING THE PROPOSED EQUIVALENT INVERTER AS COMPARED TO SPICE FOR THREE SUBMICRON TECHNOLOGIES

0.5 μm

τ	SPICE	Eq. Inv.	Error %	Illiads2	Error %	Illiads2 (inv)	Error %
0.5	0.338	0.357	5.62	0.425	25.74	0.366	8.28
1	0.423	0.439	3.78	0.540	27.66	0.458	8.27
3	0.680	0.675	0.74	0.935	37.50	0.763	12.21
5	0.882	0.896	1.59	1.282	45.35	1.038	17.69
10	1.308	1.321	0.99	2.041	56.04	1.585	21.18

0.35 μm

τ	SPICE	Eq. Inv.	Error %	Illiads2	Error %	Illiads2 (inv)	Error %
0.5	0.372	0.413	11.02	0.499	34.14	0.449	20.70
1	0.480	0.513	6.88	0.621	29.38	0.553	15.21
3	0.862	0.932	8.12	1.175	36.31	1.057	22.62
5	1.198	1.290	7.68	1.698	41.74	1.492	24.54

0.25 μm

τ	SPICE	Eq. Inv.	Error %	Illiads2	Error %	Illiads2 (inv)	Error %
0.2	0.459	0.441	3.92	0.629	37.04	0.621	35.29
0.5	0.512	0.490	4.30	0.728	42.19	0.689	34.57
1	0.628	0.609	3.03	0.860	36.94	0.838	33.44
3	1.058	1.019	3.69	1.419	34.12	1.345	27.13

efficiently the width of the equivalent transistor that replaces serially connected transistors is significant. The insufficiency that is present when the proposed analysis is incorporated in ILLIADS2 as we move to deep submicron cases is probably due to the inadequacy in modeling the equivalent inverter of the primitive macromodel that is being used by ILLIADS2.

Once the output waveform of a gate is obtained, propagation delay can be calculated as the time from the half-V_{DD} point of the input to the half-V_{DD} point of the output. Propagation delay results for NAND4 gates are also shown in Table I together with percentage errors of each method. Similar errors for ILLIADS have also been reported in [13].

III. Starting Point of Conduction

In a transistor chain with initially discharged internal nodes and the same input applied to the gates of all transistors, the closer to the output transistors start conducting later because of a gradual increase in their source and threshold voltage. In order to model efficiently the chain by an equivalent transistor, the starting point of conduction of the chain which is actually that of the top-most transistor, has to be estimated.

Let us consider the example of a six-transistor chain with all internal nodes initially discharged, where the same input is applied to all transistors. Fig. 7 shows a representation of the drain voltages of the five lower transistors together with the common input. Because of coupling capacitance between transistor gates and the drain/source nodes, drain voltages tend to follow the input ramp until all lower transistors start conducting. Initially the transistors are in the cutoff region and the coupling capacitance is calculated as the sum of the gate-to-source and gate-to-drain overlap capacitances of the upper and lower transistors respectively, in each node. These overlap capacitances are given by $C_{\text{overlap}} = W[C_{gdo} + C_{gso}]$ where W is the transistor width and C_{gdo}, C_{gso} are the gate-to-drain and gate-to-source overlap capacitances per micron which are determined by the process technology. Until the time where the transistor below a node starts conducting, the voltage waveform of that node, as it is isolated between two cut-off transistors, is derived by equating the current due to the coupling capacitance of the node $I_{C_{M_i}}$ with the charging current of the parasitic node capacitance I_{C_i}

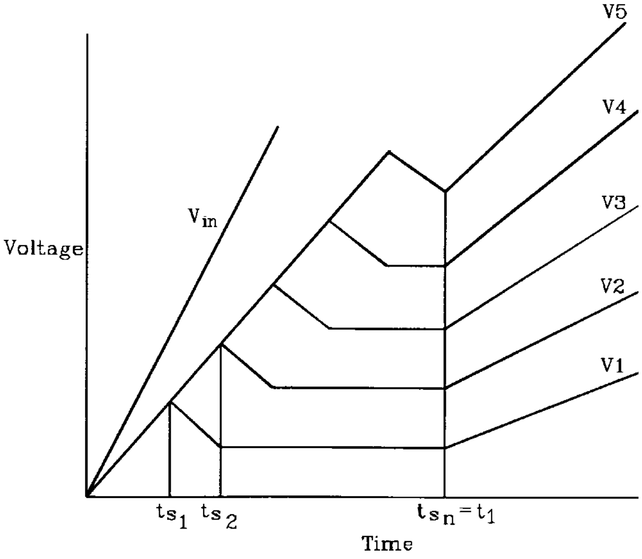


Fig. 7. Intermediate node voltage waveforms until the transistor chain starts conducting.

$$I_{C_{M_i}} = I_{C_i} \Rightarrow C_{M_i}\frac{dV_{\text{in}} - dV_i}{dt} = C_i\frac{dV_i}{dt} \Rightarrow V_i[t] = \frac{C_{M_i}}{C_{M_i} + C_i}V_{\text{in}}[t]. \tag{12}$$

After the time at which all transistors below the $\#i$ node start to conduct (t_{s_i}) and until the time at which the complete chain starts to conduct (t_1), this node is subject to two opposite trends. One tends to pull the voltage of the node high and is due

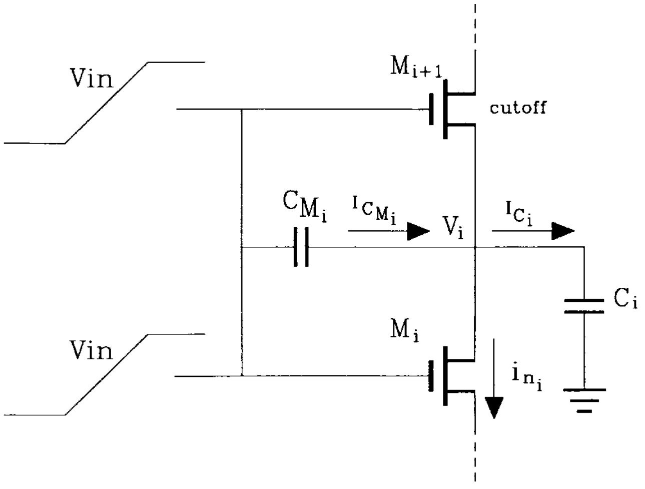


Fig. 8. Currents at the $\#i$ node of the transistor chain during $[t_{s_i}, t_{s_{i+1}}]$.

to the coupling capacitance between input and the node and is intense for fast inputs and high coupling to node capacitance ratio. The other tends to pull its voltage down because of the discharging currents through all lower transistors and is more intense for nodes closer to the ground.

When a transistor starts to conduct, e.g., transistor $\#i$, it operates initially in saturation. Therefore, since its gate-to-drain coupling capacitance is very small, the second (except for the case of very fast inputs) from the above mentioned trends dominates after time t_{s_i} and the voltage at node i decreases. This continues until time $t_{s_{i+1}}$ when transistor $i+1$ starts conducting and enters saturation. Transistor $\#i$, since its V_{GS} continues to increase after time t_{s_i} while its V_{DS} decreases, will enter the linear region close to $t_{s_{i+1}}$. From this point on, the gate-to-source coupling capacitance of transistor $\#i+1$ increases by $2/3C_{\text{ox}}$ and the gate-to-drain coupling capacitance of transistor $\#i$ increases by $1/2C_{\text{ox}}WL$. Because of this increased coupling capacitance at node $\#i$, the two previously mentioned trends are counterbalanced and for simplicity the node voltage is considered constant and equal to its value at $t_{s_{i+1}}$. This observation has also been verified by SPICE simulations. The node voltages start to rise again when the complete chain starts conducting at time t_1. Additionally, the slope of the voltage waveform during $[t_{s_i}, t_{s_{i+1}}]$ is considered the same for each node and the voltage expression of node 1 during this period can be calculated by solving the differential equation which results from the application of Kirchhoff's current law at node 1 (Fig. 8)

$$\begin{aligned} i_{n_1} &= i_{C_{M1}} - i_{C_1} \Rightarrow k_s \cdot (V_{\text{in}} - V_{TO}) \\ &= C_{M_1}\left(\frac{dV_{\text{in}}}{dt} - \frac{dV_1}{dt}\right) - C_1\frac{dV_1}{dt} \end{aligned} \tag{13}$$

where the transconductance k_s is measured on the I-V_{DS} characteristics for very low values of V_{GS} $(\approx V_{TO})$ and V_{DS} $(\approx (C_M/(C_M + C_n))V_{TO})$ and, for simplicity, the velocity saturation index α is considered one.

Since $t_{s_1} = (V_{TO} \cdot \tau / V_{DD})$ is known and $V_1[t_{s_1}]$ is given by (12), the expression of $V_1[t]$ during $[t_{s_1}, t_{s_2}]$ is derived and can be used in order to calculate the time when the next transistor further up starts conducting, by solving $V_{GS_2}[t_{s_2}] - V_{TN_2}[t_{s_2}] = 0$. Having time points t_{s_1}, t_{s_2}, and the corresponding drain voltage values of the bottom transistor, $V_1[t_{s_1}], V_1[t_{s_2}]$ the average slope r of each node voltage waveform during $[t_{s_i}, t_{s_{i+1}}]$ can be obtained.

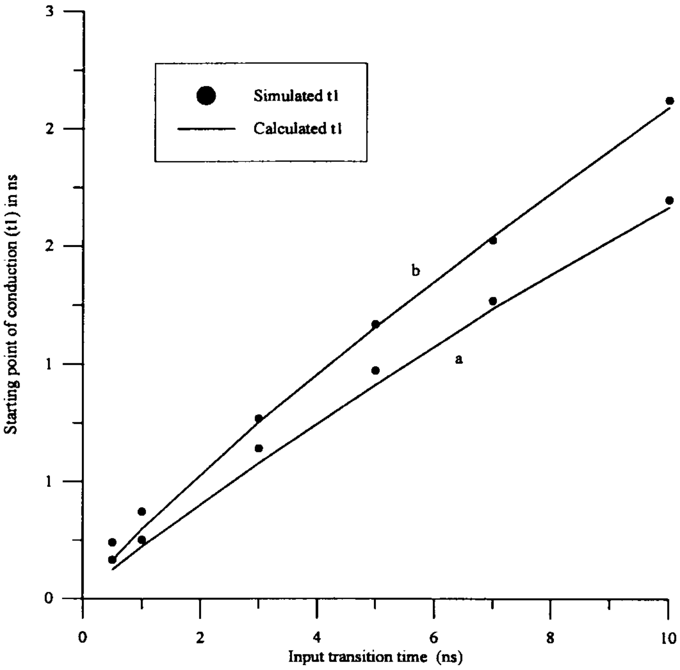


Fig. 9. Comparison between simulated and calculated starting point of conduction for several input transition times and for (a) four-transistor chain and (b) six-transistor chain ($L = 0.5$ μm).

According to the above analysis, the time point at which the $\#i$ transistor in the chain starts conducting can be found by solving

$$\begin{aligned} &V_{GS_i}[t_{s_i}] - V_{TN_i}[t_{s_i}] \\ &\quad = 0 \Rightarrow V_{\text{in}}[t_{s_i}] - \theta_0 - (1+\delta_0) \\ &\quad \cdot \left(\frac{C_{M_{i-1}}}{C_{M_{i-1}} + C_{i-1}} \cdot V_{\text{in}}[t_{s_{i-1}}] - r \cdot (t_{s_i} - t_{s_{i-1}})\right) = 0 \end{aligned} \tag{14}$$

which results in the recursive expression

$$t_{s_i} = \tau \frac{\theta_0 + (1+\delta_0)\left(\dfrac{C_{M_{i-1}}}{C_{M_{i-1}} + C_{i-1}} \dfrac{V_{DD}}{\tau} + r\right) \cdot t_{s_{i-1}}}{V_{DD} + (1+\delta_0) \cdot r \cdot \tau} \qquad i \geq 2. \tag{15}$$

From the above expression, the time at which the chain starts conducting $t_{s_n} = t_1$ can be easily obtained. Constants θ_0, δ_0 result from (3) by calculating the Taylor series approximation of the threshold voltage around $V_{SB} = V_{TO}$ for higher accuracy in this region. According to the previous analysis, the starting point of conduction can be calculated with very good accuracy as shown in Fig. 9 which is a comparison between the calculated and the actual time t_1 which is obtained from SPICE simulations.

IV. Modeling the Parasitic Behavior of the Transistor Chain

With the term parasitic behavior of the transistor chain during output switching of a CMOS gate, we refer to its parasitic effect on the output voltage evolution. The parasitic behavior results in a short-circuit current which reduces the rate of charging/discharging of the output load and increases the propagation delay.

Let us consider a NAND gate where all pMOS transistors have been replaced by an equivalent one with its transconductance multiplied by the number of the transistors. (Parallel transistors will be discussed in Section VI.) In this way the parasitic behavior of an nMOS transistor chain will be modeled. The case of a pMOS chain is symmetrical. A falling ramp input with transition time τ is considered to be applied to the gates of all transistors

$$V_{\text{in}} = \begin{cases} V_{DD}, & t < 0 \\ V_{DD} - \dfrac{V_{DD}}{\tau} \cdot t, & 0 \leq t \leq \tau \\ 0, & t > \tau. \end{cases} \tag{16}$$

The pMOS device starts conducting when the input reaches the threshold voltage $(V_{\text{in}} = V_{DD} - |V_{TPO}|)$ at time $t = t_p$. From this time on, current is flowing through the pMOS device and the load capacitance C_L charges. Since the nMOS devices are on when the pMOS transistor starts conducting, a short-circuit current is flowing through the gate from V_{DD} to the ground until time t_n $(V_{\text{in}} = V_{TN})$ when the nMOS transistors cease to conduct. First, because the output voltage is small while the gate-to-source voltage of the nMOS devices is large, all these transistors start their operation in linear mode. As the output voltage rises, the voltages at the internal nodes of the chain are also increasing. All nMOS transistors have almost equal V_{DS} (voltage divider) while the top is biased by the smallest V_{GS} (since its source voltage has the largest value from all internal nodes). This means that this transistor at some time point will enter saturation and after this time, the current in the chain will decrease and consequently the voltages at the internal nodes of the chain will also decrease, keeping all other devices in linear mode.

A significant amount of the parasitic current is also flowing through the coupling capacitances between the gates of the nMOS transistors and the corresponding drain/source diffusion areas. In order to perform an accurate modeling of the gate when the chain behaves parasitically, an equivalent capacitance that would draw the same current as the coupling capacitances at all nodes of the chain has to be inserted between the input terminal and the output node of the corresponding nMOS transistor in the equivalent inverter. Although the dual operation of the top-most transistor is also present during the parasitic operation of the chain, conventional estimation of the width of the equivalent transistor as $W_{\text{eq}} = (W/n)$ has been found to give sufficiently accurate results if the effect of the parasitic capacitances is modeled properly.

The coupling capacitance between gate and drain/source diffusion areas consists of the overlap capacitance (C_{gdo}, C_{gso}) of the transistor and a channel capacitance (C_{gd}, C_{gs}) whose value depends on the operation region [18]. Thus, the coupling capacitance for each node except for the output node in the transistor chain is calculated as the sum of the above four capacitances and denoted as C_M. For the output node only the contribution of the nMOS device is taken into account: when the top transistor of the chain operates in linear mode its gate-to-drain coupling capacitance is equal to $C_M/2$ while during saturation it can be neglected, because C_{gd} is almost equal to zero [18] and the overlap capacitance $C_{gdo}W$ is very small.

Since the pMOS transistor starts its operation in saturation, the output load will be charged by a current of the form $I_s = k_s \cdot (V_{GS} - |V_{TP}|)^a$. If we ignore the parasitic contribution of the nMOS transistor currents, the rate of the output voltage increase during $[t_p, t_{\text{sat}}]$ is given by

$$C_L \frac{dV_{\text{out}}}{dt} = I_s(t) \Rightarrow \frac{dV_{\text{out}}}{dt} = \frac{I_s(t)}{C_L} = I^c(t) \tag{17}$$

where t_{sat} is the time when the top transistor in the nMOS transistor chain enters saturation and is approximated by $(t_n + t_p)/2$, where t_n is the time when the nMOS transistors cease to conduct $(t_n \approx ((V_{DD} - V_{TN})/V_{DD}) \cdot \tau)$, which is a reasonable approximation according to SPICE simulations. It should be mentioned that the pMOS transistor until time t_{sat} operates in saturation, since the time point when it exits saturation for most of the cases is larger than t_{sat}.

Considering the whole chain as a voltage divider for the interval t_p to t_{sat}, the slope of each internal node voltage can also be found assuming a uniform distribution of the output voltage slope.

The current that each coupling capacitance is drawing during time interval $[t_p, t_{\text{sat}}]$ is equal to

$$I_i = C_{M_i} \frac{d(V_i - V_{\text{in}})}{dt} = C_{M_i} \cdot \left(\frac{i \cdot I^c}{n} + s \right) \tag{18}$$

where n is the number of the transistors in the chain, s is the slope of the input and $i \cdot I^c/n$ the slope of the voltage waveform at the internal node i.

By summing the currents through all coupling capacitances of the chain and equating the sum with the current that must flow through the equivalent coupling capacitance $(C_{M_{\text{eq}}})$ of the equivalent transistor, $C_{M_{\text{eq}}}$ is obtained

$$\sum_{i=1}^{n} I_i = C_{M_{\text{eq}}} \cdot (I^c + s). \tag{19}$$

A constant value for $C_{M_{\text{eq}}}$ can be obtained if an average value for I^c is calculated by integrating the pMOS current I_s in $[t_p, t_{\text{sat}}]$. This value corresponds to the average slope of the output voltage waveform until t_{sat}.

When the node voltages are decreasing during $[t_{\text{sat}}, t_n]$, the equivalent coupling capacitance can be found in a similar way. By symmetry, the same slope (with opposite sign) results for the voltage waveforms of the internal nodes.

Setting $c_r = \tilde{I}^c = (1/(t_{\text{sat}} - t_p)) \int_{t_p}^{t_{\text{sat}}} (I_s(t)/C_L)\, dt$, the equivalent coupling capacitance for the two time intervals can be written as

$$C_{M_{\text{eq}}} = C_M \cdot \frac{n \cdot c_r + (2n-1) \cdot s}{2 \cdot (c_r + s)}, \qquad [t_p, t_{\text{sat}}] \tag{20}$$

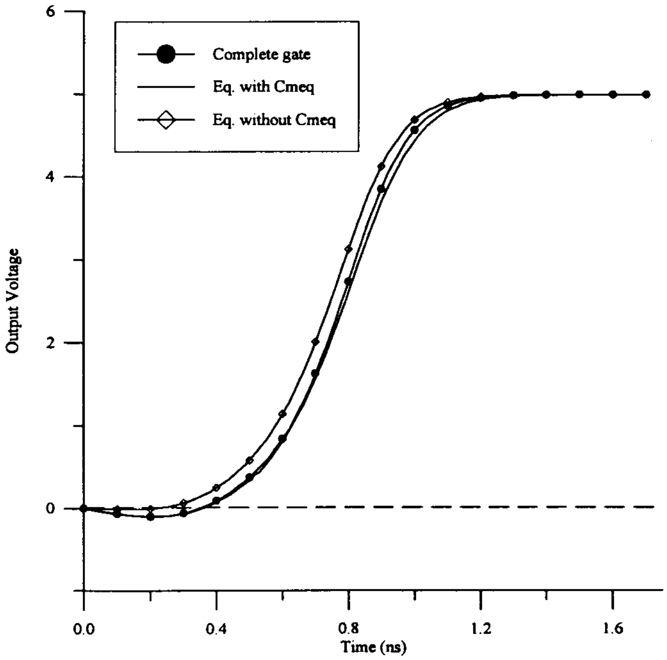


Fig. 10. Output waveform comparison between the complete gate and the equivalent inverter with and without the equivalent coupling capacitance ($L = 0.5$ μm).

$$C_{M_{\rm eq}} = C_M \cdot \frac{(n-1)\cdot\left(s-\dfrac{c_r}{2}\right)}{(c_r+s)}, \qquad [t_{\rm sat}, t_n]. \quad (21)$$

If the contribution of the gate-to-drain coupling capacitance of the top-most transistor in the chain is neglected during $[t_p, t_{\rm sat}]$ and the average slope of the output node is taken equal to the input waveform slope (with opposite sign) the above equivalent capacitances diminish to

$$C_{M_{\rm eq}} = C_M \frac{3\cdot(n-1)}{4}, \qquad [t_p, t_{\rm sat}] \quad (22)$$

$$C_{M_{\rm eq}} = C_M \frac{n-1}{4}, \qquad [t_{\rm sat}, t_n] \quad (23)$$

which express directly their dependency on the number of transistors in the chain.

The improvement that is gained by inserting this coupling capacitance to the equivalent inverter model of a gate is significant as shown in Fig. 10 which is a comparison of the output response of a four-input NAND gate and that of the corresponding equivalent inverter with and without the calculated coupling capacitance, when the transistor chain acts parasitically.

V. Weight Calculation and Input Mapping Algorithm

Since all transistors of a chain are collapsed to a single equivalent one, the problem that arises is how to map all input signals that are applied to all transistor gates and might have different starting points and transition times, to a single input signal that will be applied to the equivalent transistor gate. The problem of misaligned multiple transitions has also been identified in [19] for existing gate collapsing techniques at the gate level approach. As shown in the previous analysis, in case all inputs in the chain are the same, the single equivalent input will also be the same. From this point on, n input ramps which are applied to the n transistors of a chain and have the same transition time and starting time, will be referred to as *normalized inputs* and the single equivalent one as *normalized input*. Additionally, every set of input ramps (less than n) which have the same transition time and the same starting point, will be referred to as *equal ramps*. Therefore, the problem focuses on how to map any arbitrary number of different input ramps and DC voltages to normalized inputs, which when applied to the chain will cause the same output response.

One method for calculating a single effective input ramp was proposed in [10] and states that for all input ramps that are in transition after the starting time of the latest one, the equivalent ramp starts at the initial point of the ramp that starts first and ends at the last ending point of all ramps. This simplified algorithm introduces large errors for most of the cases, especially when some signals have a much smaller transition time than others or when the starting point of some signals differ significantly. In [20], the single equivalent input is taken as the one which reaches the threshold level last between all input ramps and even larger errors than in [10] are present.

An analytical method that can extract a single equivalent input is very difficult to develop, since the influence of each input ramp to the equivalent one depends on a number of factors. As it is obvious, the starting point of the last changing input plays an important role since the chain does not conduct before that point. Consequently, inputs which start at a time point which has a large distance in time from the last starting point, contribute less to the single equivalent input than signals which start close to the last changing input. A second factor that is important is the position of each input in the chain. Transistors further up in the chain (closer to the output node) have a lower gate-to-source voltage and a larger threshold voltage, thus their inputs result in a slower output response. The single equivalent input also depends on the slope of each ramp, the relation of its slope to the slope of other input ramps and its relative position in time to other signals.

According to the above, the proposed algorithm aims at the extraction of n normalized input ramps for each possible input pattern, so that when the normalized inputs are applied to the transistor gates of the chain, the chain will have the same output response with that of the actual inputs. Before the proposed algorithm can be applied, the “weight” of each transistor in the chain has to be calculated, i.e., a coefficient which corresponds to the position of each switching input (or combination of inputs). Using only one set of simulations for each technology, these weight coefficients can be obtained as follows: Equal ramp(s) are applied to one or more transistors in the chain whose weight coefficient is to be measured and the rest of the transistors receive a V_{DD} voltage (pattern CASE). In order for two different input patterns to be equivalent, the output response when these patterns are applied as inputs to a circuit must be identical. This means that for the chain of Fig. 1 the discharging rate of the load capacitance must be the same. Therefore, for each input pattern consisting of equal input ramps and DC voltages, the amount of charge that is discharged through the chain from time $t = t_1$ where the chain starts conducting until the input has reached its final value at time τ, is measured. If the same ramp input is applied to all transistors in the chain (pattern ALL) the rate of discharging slows down and the charge which is discharged in the same time interval will be a fraction of that of pattern CASE. Thus it is

$$\int_{t_1}^{\tau} I_{\rm all}[t]\cdot dt = g\cdot\int_{t_1}^{\tau} I_{\rm case}[t]\cdot dt \tag{24}$$

where $I_{\rm case}$ is the current that is flowing through the bottom transistor for each case and $I_{\rm all}$ is the corresponding current when all inputs are ramps. The fractional coefficient g $(g < 1)$ can be easily obtained, for example with SPICE, by calculating both integrals of the current through the bottom transistor.

For each case, the aim is to find the corresponding normalized inputs (pattern NORM) which start at the same point with the applied equal ramps of pattern CASE, but have a transition time $(\tau_{\rm norm})$, so that the system has the same output response with that of pattern CASE. It is obvious that is has to be $\tau_{\rm norm} < \tau_{\rm case}$. When two transistor chains have the same output response, the currents of the bottom transistors are equal at each time. Therefore, $I_{\rm case}[t] = I_{\rm norm}[t]$ where $I_{\rm norm}$ is the current through the bottom transistor when the normalized inputs are applied to all transistors in the chain. Integrating both sides of this equation from time $t = t_1$ to τ

$$\int_{t_1}^{\tau} I_{\rm case}[t]\,dt = \int_{t_1}^{\tau_{\rm norm}} I_{\rm norm}[t]\,dt + \int_{\tau_{\rm norm}}^{\tau} I_{\rm norm}[t]\,dt. \tag{25}$$

By equating (24) and (25) and replacing the transistor currents (bottom transistor operates always in linear mode) with their expressions, it is

$$\begin{aligned}\int_{t_1}^{\tau_{\rm norm}} & k_l\left(\frac{V_{DD}}{\tau_{\rm norm}}t - V_{TO}\right)^{a/2}\frac{\tilde{V}_p}{\tau_{\rm norm}}t\,dt \\ & + \int_{\tau_{\rm norm}}^{\tau} k_l(V_{DD}-V_{TO})^{a/2}\tilde{V}_p\,dt \\ & = \frac{1}{g}\int_{t_1}^{\tau} k_l\left(\frac{V_{DD}}{\tau}t - V_{TO}\right)^{a/2}\frac{\tilde{V}_p}{\tau}t\,dt.\end{aligned} \tag{26}$$

For the above equation to be valid, the input should be fast so that during the time interval $[\tau_{\rm norm}, \tau]$ V_{DS} of the bottom transistor remains constant and equal to $\tilde{V}_p = (V_p/(n-1))$ (for nontapered chains). The above equation can be solved for $\tau_{\rm norm}$ and the weight coefficient for each case is calculated as

$$c_{\rm weight} = \frac{\tau_{\rm norm}}{\tau}. \tag{27}$$

Although the above calculation is performed for a specific transition time of the inputs the obtained weight coefficients have been found to be valid for a wide range of input transition times. Consequently, these coefficients can be obtained running only one simulation for each combination of inputs for the calculation of the fractional coefficient g in (24). It should be mentioned that the calculated coefficients are according to (26) independent of the transistor widths. The "weight" coefficient $c_{\rm weight}$ for a four-transistor chain is given in Table II for a 0.5 μm HP technology. It should be noted that nonsimultaneous inputs might create a conducting path from the output to some internal nodes. The effect of the introduced charges on the output evolution is incorporated in the calculation of the weight coefficients.

TABLE II
"WEIGHT" COEFFICIENTS FOR A FOUR-TRANSISTOR CHAIN. THE INPUT NUMBERING STARTS FROM THE ONE CLOSEST TO THE GROUND ($L = 0.5$ μm)

Changing Inputs	c_{weight}
1,2,3,4	1
2,3,4	0.93
1,3,4	0.89
1,2,4	0.85
1,2,3	0.92
3,4	0.82
2,4	0.77
1,4	0.74
2,3	0.815
1,3	0.775
1,2	0.8
4	0.71
3	0.67
2	0.64
1	0.6

The next three steps of the mapping algorithm should be applied for every possible input pattern (here presented for the case of an nMOS chain).

Step 1: Inputs which efficiently act and should be treated as V_{DD} voltages have to be identified. In order to achieve this, every input ramp which at time $t = t_m$ has a value larger than $2/3V_{DD}$ should be considered V_{DD} for the following steps. Time t_m occurs when the last ending input ramp reaches $V_{DD}/2$. In case two or more inputs end at the same time, t_m is measured on the one that starts last.

Step 2: The m ramp inputs that remain from Step 1 have to be transformed to equal ramps. The starting point of these equal ramps (t_0) is taken as $t_0 = \max(t_1, t_2, \cdots, t_n)$ where $t_1, t_2, \cdots, t_n$ are the starting point of **all** input ramps in the chain. This is reasonable since the chain does not conduct current before the starting point of the last changing input. In order to take into account the slope as well as the time during which an input is in transition after time t_0, the transition time $(T_{\rm eq})$ of the the equal ramps is taken as

$$T_{\rm eq} = \frac{\displaystyle\sum_{i=1}^{m}\left[1 - \frac{V_i[t_0]}{V_{DD}}\right](t_{e_i} - t_0)}{m} \tag{28}$$

where $V_i[t_0]$ is the voltage that each input ramp has reached at the initial time and t_{e_i} is the time point at which each of the m input ramps reaches V_{DD}. Obviously, because of Step 1, $t_{e_i} > t_0$ for all of the m inputs. According to (28), inputs which start at the initial time t_0 will have the major influence on $T_{\rm eq}$ since the corresponding multiplication factor will be one.

Step 3: The resulted input pattern from Step 2 which consists of equal ramp inputs and V_{DD} inputs can be mapped to an equivalent normalized one consisting only of ramp inputs using the corresponding weight coefficients. Thus, the transition time, $T_{\rm eff}$ of the normalized inputs which are applied

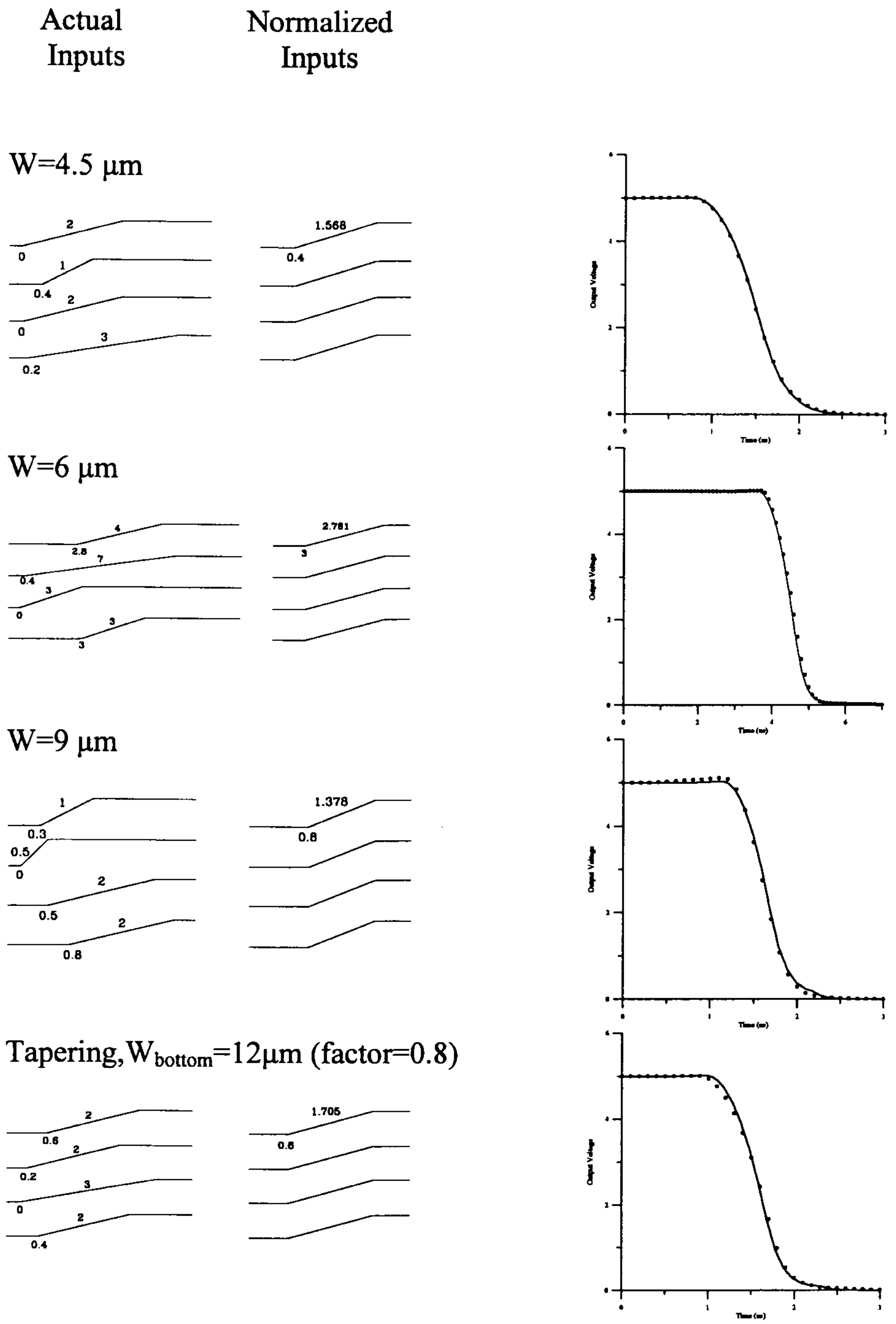


Fig. 11. Comparison between the output responses of the transistor chain ($L = 0.5$ μm) for actual inputs (dots) and for normalized ones. The starting point and the transition time of each input ramp is given in ns.

to the chain at time $t = t_0$ is

$$T_{\text{eff}} = c_{\text{weight}} \cdot T_{\text{eq}}. \tag{29}$$

The normalized ramp input is finally applied to the single equivalent transistor. It should be mentioned that the normalized input which has been derived for the conducting part of a gate will also be applied to the parasitic part of that gate.

The above algorithm presented very good accuracy for inputs with a wide range of transition times and relative distances in time of their starting points. In Fig. 11, a comparison of the output responses of a four transistor chain to the actual and normalized input patterns is presented, which shows the accuracy and efficiency of the proposed algorithm.

The implementation of the algorithm is explained in Fig. 12(a) for the case of an nMOS transistor chain. The algorithm is applied symmetrically for a pMOS transistor chain. Since at each stage of the circuit the inputs are known (they are either primary inputs or extracted from a previous stage) simulation can be performed in an event driven manner by defining phases instead of time points. A phase $P[t = 0, t = \tau]$ is defined by the starting and ending point of each input. An nMOS transistor chain starts conducting when

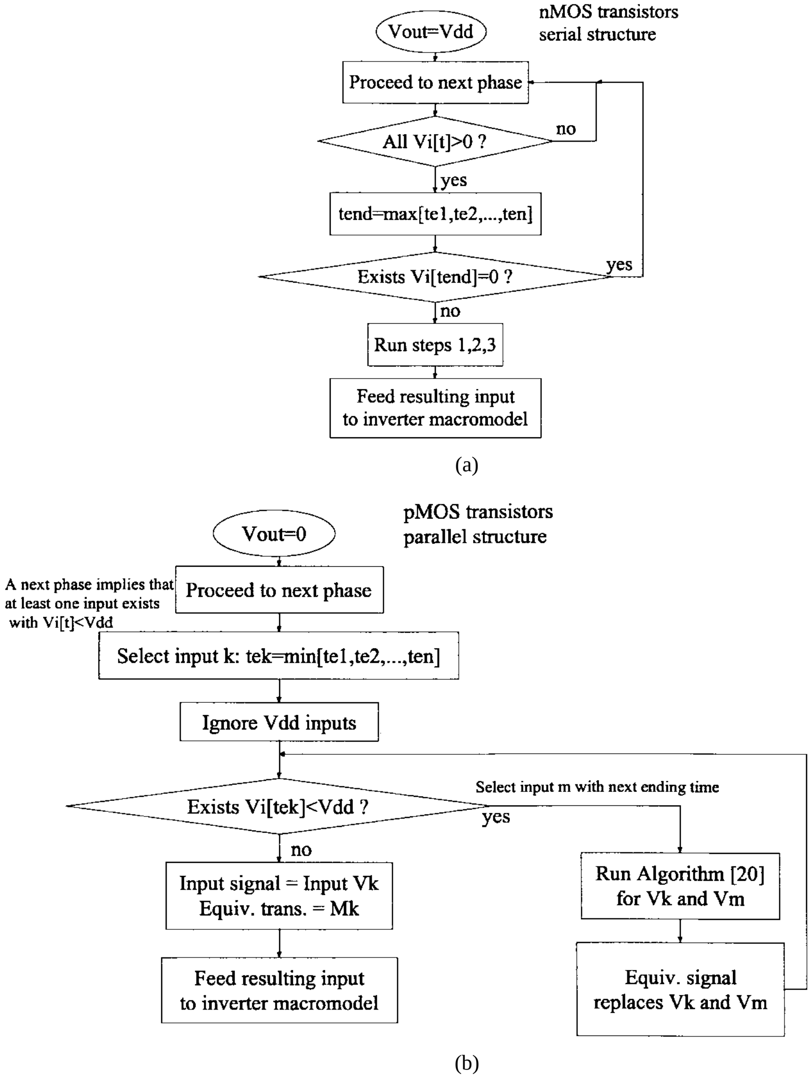


Fig. 12. Input mapping algorithm flowcharts.

the requirement that all transistors are conducting is satisfied. Then, at the end of all inputs which switch during this phase it is checked whether any input has returned to a low value. In that case a glitch may be formed but the output does not change state. Otherwise, the algorithm is performed on the inputs as explained previously and the resulting input is fed to the inverter macromodel.

VI. Parallel Transistor Structure

The parallel transistor structure is less complicated than the serial one, since if an equal transistor width is assumed (as in the case of a NAND/NOR gate) and the same input ramp is applied to the gates of all transistors, the currents flowing through each transistor will be identical. That is because all the devices operate under the same conditions, i.e., they have the same drain-to-source and gate-to-source voltage. Consequently, the parallel transistor structure can be replaced by an equivalent one with its width multiplied by the number of the transistors.

In case the applied input ramps are not normalized, the extraction of a single equivalent input ramp out of two inputs according to the algorithm presented in [20] leads to sufficiently accurate results. However, another step should be added in order to increase the accuracy. Therefore, before applying the algorithm, the inputs which act as DC voltages (voltages which keep the corresponding transistors in cutoff) should be identified and should not take part in the algorithm. For example, in the extraction of a single equivalent input out of two falling inputs for parallel pMOS transistors, if the last ending input at the time when the other one reaches $V_{DD}/2$ has a value larger than $2/3V_{DD}$ and its transition time is larger than twice the transition time of the other one, it should be considered as V_{DD} voltage. This extra step has been found to increase the overall accuracy of the algorithm significantly.

The application of the algorithm for the case of a group of parallel pMOS transistors is shown in Fig. 12(b). In that case the output can change state even when a single transistor is conducting. Whenever a new phase begins all inputs which are changing state during this phase are taken into account and the one which ends first is selected. In the next step inputs which effectively act as V_{DD} are ignored. In case only one input remains, this input is also selected as the input to the inverter macromodel and only the corresponding transistor will be taken into account. If more than one inputs are switching, the algorithm [20] is performed repeatedly until only one switching input (and transistor) is left as input to the inverter macromodel.

It is obvious that the starting point of conduction for a structure of parallel transistors is the same with the starting point of conduction for each one of them and is calculated, e.g., in the case of nMOS transistors as the time when $V_{GS} - V_{TN} = 0$.

The parasitic behavior of a parallel transistor structure is modeled accurately by a single equivalent transistor with multiplied width, since the effect of the coupling capacitances is captured by the increased coupling capacitance of the equivalent transistor. Thus, no further action is required for the modeling of the parasitic behavior of transistors connected in parallel.

VII. Complex Gate Modeling

The aforementioned model is valid for NAND/NOR gates and in order to perform an analysis for more complex gates, these gates have to be mapped to an equivalent NAND/NOR gate. An algorithm for such a reduction has been proposed in [13] and [14], which was based on simple conventional collapsing of transistors, i.e., the transconductance of the equivalent transistor of a transistor structure is a multiple or submultiple of the transistor transconductances, for parallel or serially connected transistors, accordingly. In this paper, the proposed algorithm takes into account the dual operation (saturation and linear region) of the transistors attached to the output node.

After finding the conducting path from the output node to the power supply, an efficient merging technique is applied reducing the complex gate to an equivalent NAND or NOR structure. Only transistors which belong to the conducting path are considered in the reduction algorithm. The rest are ignored but the contribution of their drain/source capacitances on the internal nodes is taken into account. According to the proposed merging technique, all parallel branches within a gate are reduced (starting from the inner branches) to an equivalent transistor by gradually collapsing the serially and parallel connected transistors within the branches to an equivalent one. This reduction is performed in the conventional way except for the case when the parallel branches are attached to the output node.

For branches which are attached to the output node, the transistors which are attached to that node will operate in saturation for some time and then enter the linear region. If the branches are identical, these transistors will operate under the same conditions and they will enter the linear region at the same time point. When the parallel branches are not identical, each transistor will exit saturation at a different time point. However, it has been observed from SPICE simulations that the time interval during which the transistors of nonidentical branches attached to the output node operate in saturation is almost the same. It has also been observed that for both cases (identical and nonidentical branches) and most of the practical cases, the two time intervals (saturation and linear region) for the transistors which are attached to the output node, are almost equal.

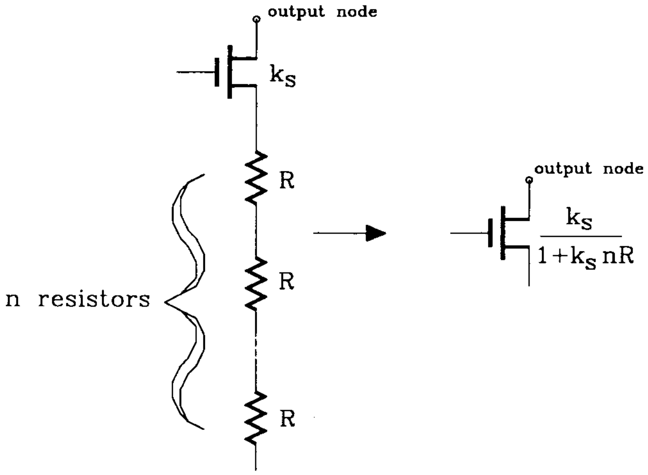


Fig. 13. Reduction of serially connected transistors when the top operates in saturation and the rest in linear mode.

Consequently, for each branch an equivalent transistor width is estimated for each of the above two operating regions and an average value is used for the single equivalent transistor which replaces each parallel branch. When the transistors which are attached to the output node operate in linear mode, the equivalent transistor is extracted in the conventional way. For the operation in saturation, the equivalent transistor of serially connected transistors within a branch where the top-most is attached to the output node, is estimated by the method shown in Fig. 13. According to this, all transistors which operate in linear mode are treated as resistors, the body effect is neglected and the velocity saturation index of the current model a is set equal to unity (good approximation for submicron devices). Ignoring the body effect is a compromise to enable further analysis in a simple and input-independent manner. However, the effect of this assumption at this level is limited as will be shown in output waveform results of complex gates. The tranconductance of the equivalent transistor results easily by equating the currents which flow through the top-most transistor and the equivalent one, taking into account the above assumptions. Finally, the width of the equivalent transistor of the parallel branches results by summing the width of the resulted transistor for each branch. The input waveform for each equivalent transistor during this reduction algorithm is extracted by applying the mapping algorithms mentioned in Sections V and VI.

For the transistor block of the complex gate which acts parasitically, the width of the equivalent transistor is estimated in the conventional way. It is obvious that only transistors which contribute to the parasitic behavior of the block are taken into account. In other words, transistors for which their counterpart in the conducting block belongs to a conducting path are taken into account; all others are considered as short-circuits. It should be noticed that the exact time point when the complex gate (t_{gate}) starts conducting has to be calculated. In order to determine t_{gate}, the shortest conducting path from power supply to the output node has to be identified. Then,

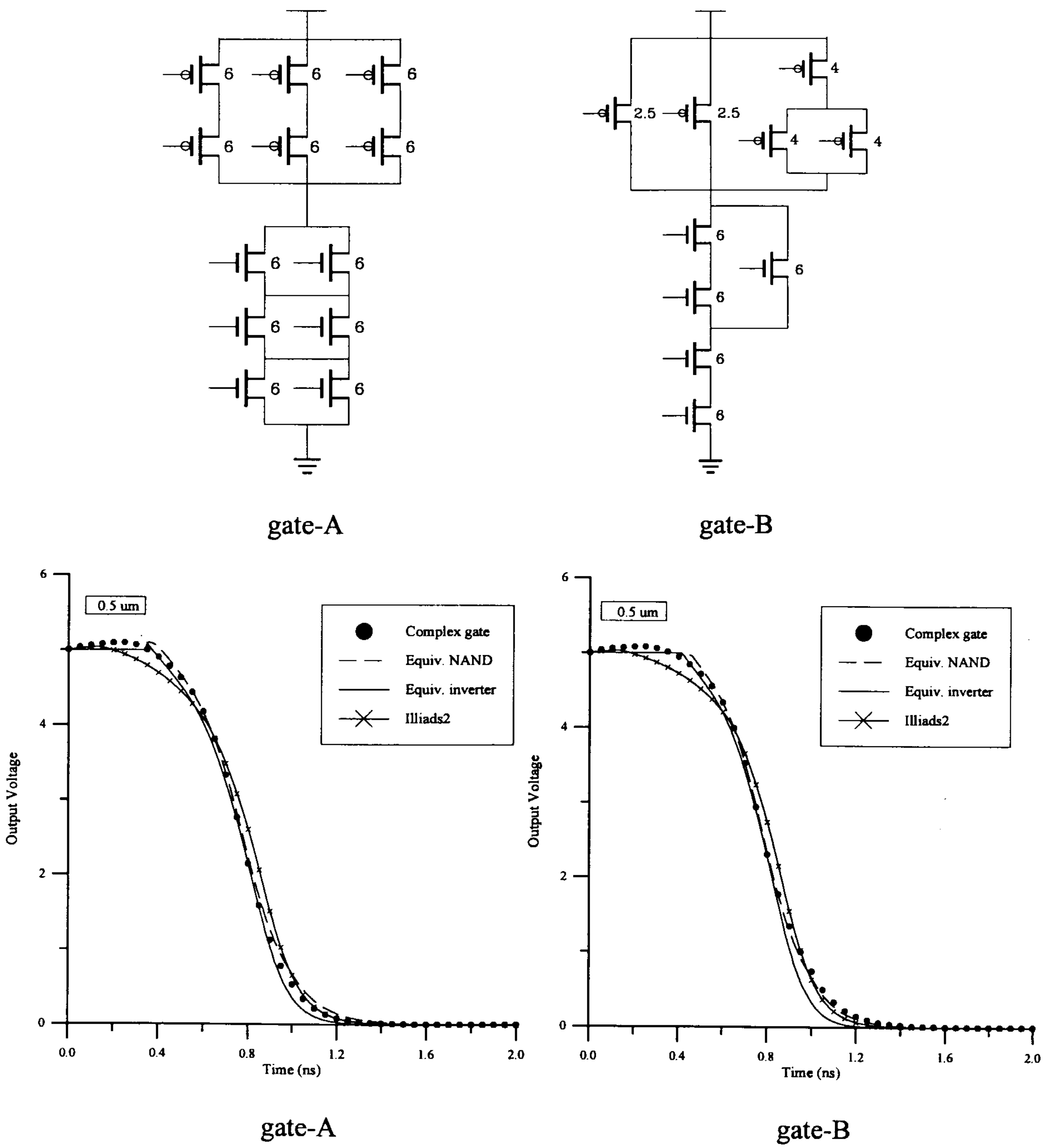


Fig. 14. Output waveform comparison between the complex gate, the equivalent NAND gate, the equivalent inverter and Illiads2 for the corresponding complex CMOS gates A and B for three submicron technologies. Transistor widths correspond only to the 0.5-μm technology.

the node capacitances on this path have to be estimated, and the starting point of conduction is calculated as described in Section III.

The proposed reduction algorithm has been applied in many complex gates with various input patterns and has been found to give sufficiently accurate results for most of the practical cases. Better results are obtained for the complex gates as the number of the serially connected transistors in the equivalent NAND/NOR gates, after the application of the reduction algorithm in the complex gate, approaches the number of the transistors in the longest path (from the output node to a power rail) of the complex gate. In Fig. 14 the output waveforms of two complex gates and their equivalent inverters are compared for three submicron technologies. The results prove the efficiency of the complete proposed modeling technique. In Table III propagation delay results for other complex gates (XOR, MUX, and AOI) are also given.

VIII. Implementation Issues

Macromodeling for timing simulation can be considered as a two-step process: In the first step complex gates are mapped to a primitive macromodel and in the next step analysis of the primitive macromodel is performed. It should be mentioned that the proposed method aims mainly at the improvement of existing gate modeling techniques which are

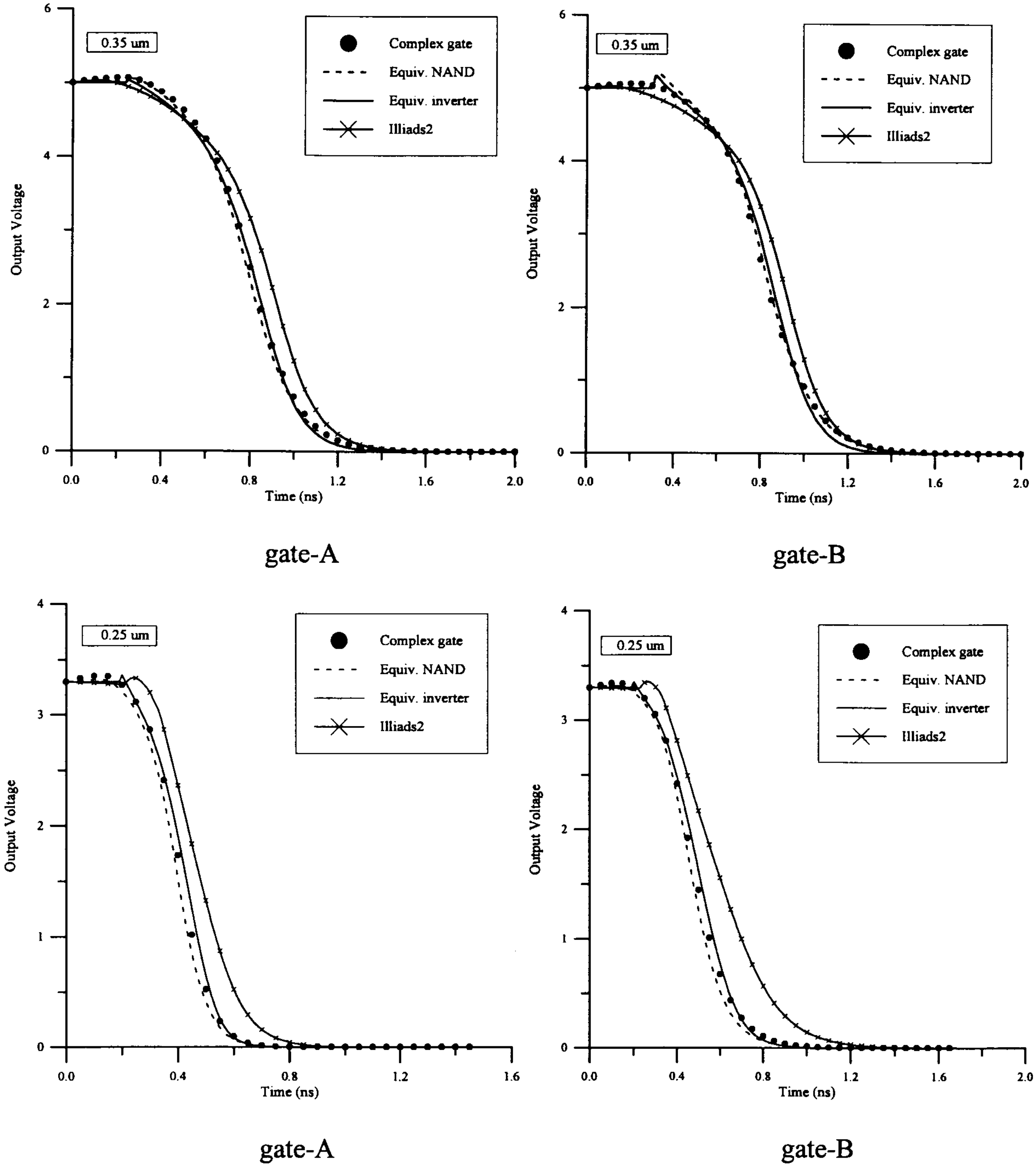


Fig. 14. (*Continued*). Output waveform comparison between the complex gate, the equivalent NAND gate, the equivalent inverter and Illiads2 for the corresponding complex CMOS gates A and B for three submicron technologies. Transistor widths correspond only to the 0.5-μm technology.

based on elementary primitives such as the inverter [13] or generic MOS circuit primitives [6], [7], [17], [21], [22]. As it has been explained in previous sections the main advantage of the proposed technique is the reduction of serially and parallel connected transistors of static and dynamic CMOS gates in an efficient and accurate way and the extraction of an accurate equivalent input signal. (Dynamic gates can be seen as a special case of complex static CMOS gates.) Consequently, this method does not introduce new ways of obtaining output or internal node waveforms of elementary primitives. Therefore, if the incorporation of the proposed method in a timing simulator is the case, simulation of basic primitives is performed exactly as in existing modeling techniques used in dynamic timing simulators (ILLIADS2 [17] and IDSIM2 [22]). The implementation of the proposed method is briefly outlined in the rest of the section.

Having a netlist description of a circuit the first step is circuit partitioning (Fig. 15) where individual gates have to be identified taking advantage of the unidirectional nature of MOS digital circuits. In other words, since signals applied to the gates of MOS transistors affect drain and source voltage but drain and source voltages have no effect on the transistor gate terminal voltage the circuit is first partitioned into individual MOS gates (similar to DC-connected blocks (DCCB's) in

TABLE III
PROPAGATION DELAY COMPARISONS (ns) BETWEEN SPICE AND THE PROPOSED METHOD FOR COMPLEX CMOS GATES (0.5 μm)

	τ = 0.5 ns			τ = 1 ns		
	Gate	Eq.Inverter	Error	Gate	Eq.Inverter	Error (%)
XOR	0.1689	0.1689	0.00 %	0.1621	0.1701	4.94 %
MUX2x1	0.2025	0.1982	2.12 %	0.2223	0.2272	2.20 %
AOI121	0.4519	0.4455	1.42 %	0.5722	0.5669	0.93 %
AOI322	0.3689	0.3488	5.45 %	0.4802	0.4587	4.48 %
GATE A	0.2351	0.2296	2.34 %	0.2715	0.2691	0.88 %
GATE B	0.2424	0.2346	3.22 %	0.2856	0.2821	1.23 %

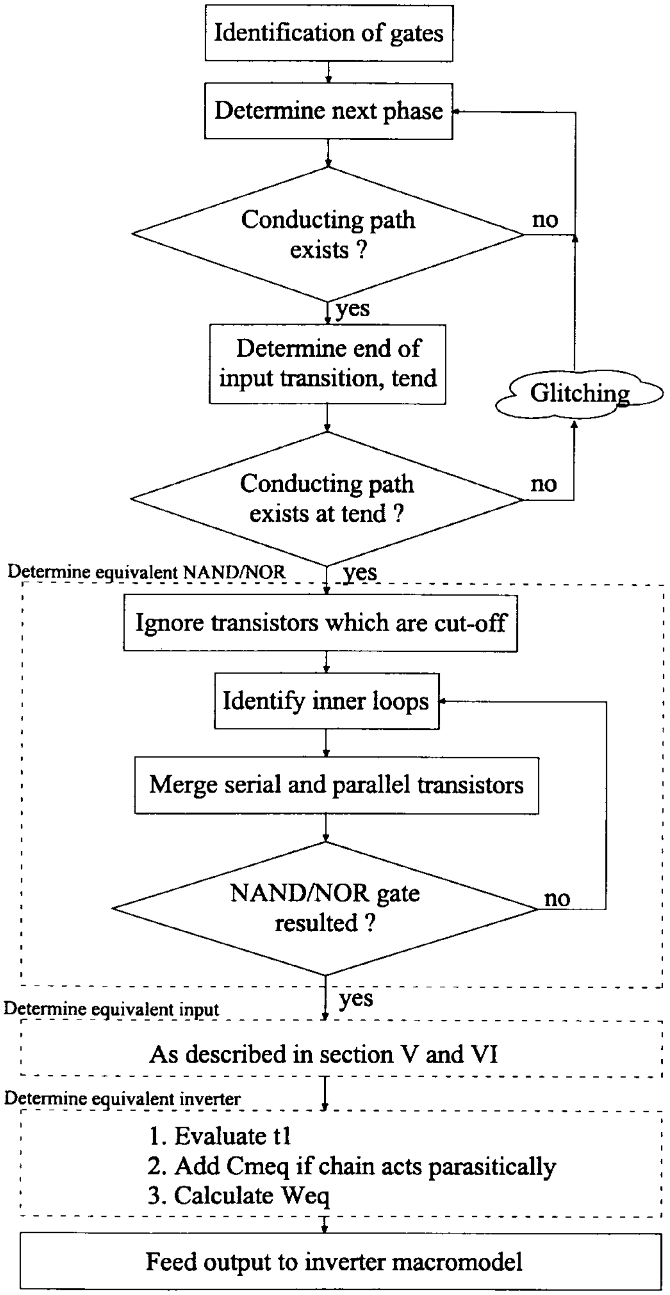

Identification of gates
Determine next phase
Conducting path exists ?
no
yes
Determine end of input transition, tend
Glitching
Conducting path exists at tend ?
no
Determine equivalent NAND/NOR
yes
Ignore transistors which are cut-off
Identify inner loops
Merge serial and parallel transistors
NAND/NOR gate resulted ?
no
Determine equivalent input
yes
As described in section V and VI
Determine equivalent inverter
1. Evaluate t1
2. Add Cmeq if chain acts parasitically
3. Calculate Weq
Feed output to inverter macromodel

Fig. 15. Flowchart indicating sequence of actions for handling a general transistor netlist.

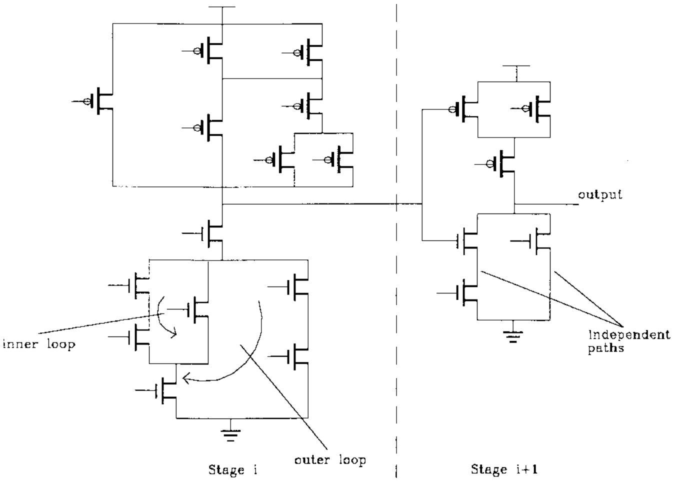


Fig. 16. Two successive logic stages with complex gates showing inner and outer loops.

[17]). In order to determine these individual MOS gates all circuit nodes which are connected by source-drain DC paths are grouped together. Whenever a "boundary" node is reached (i.e., a power supply node or a gate terminal) no other nodes are included in this group beyond this node. For each node, paths to power supply and ground are determined. Transistors which participate in both of these paths correspond to pass transistor logic and are not included in gates.

The flowchart in Fig. 15 explains the treatment of static and dynamic CMOS gates according to the analysis that has been presented. Circuits which contain feedback loops such as cross-coupled inverters can be modeled according to the methodology of strongly connected components (SCC's) [17], [22] and applying waveform relaxation (WR) [23]. Pass-transistor topologies can be handled by mapping them to an equivalent charge sharing primitive [17], [22].

The next step (Fig. 15) is the detection of the next phase where an input starts switching. Since all inputs at a given stage (Fig. 16) of a circuit are known, moving directly to the next phase improves simulation runtime. If the switching inputs lead to a conducting path from the output node to power supply or ground, the end of the input transition t_{end} is determined otherwise the program moves to the next phase. At time t_{end} the condition of a conducting path is checked again in order to determine whether the output will change state or a glitch will occur.

In case the output of a complex gate switches (i.e., conducting criteria are fulfilled at the beginning and the end of a phase), the gate has to be mapped to an equivalent NAND/NOR gate prior to transient simulation taking into account only transistors which belong to conducting paths. Merging of serial and parallel transistor structures begins from the inner loops (Fig. 16) of the complex gate and the process is

TABLE IV
EXECUTION TIMES (s) AND SPEEDUP FACTORS FOR FOUR CMOS GATES (ALL INPUTS SWITCHING)

	.TRAN 0.01ns 5ns					
	SPICE	Illiads2	Speedup	Proposed	Speedup	Penalty
NAND2	0.44	0.0646	6.81	0.0654	6.73	1.238 %
NAND4	0.74	0.0719	10.29	0.0732	10.11	1.808 %
GATE A	1.08	0.0695	15.54	0.0713	15.15	2.589 %
GATE B	0.88	0.0678	12.98	0.0694	12.68	2.359 %

```
1. Determine dc paths from output node to power rails
2. Identify independent paths (paths which have no common transistors)
     For all dependent paths
     {
3.     Find those that have the largest number of transistors in common
4.     For these paths, not common transistors form the most inner loops
5.     Merge the loop, update paths and go to step 3
     }
6. Merge independent paths
```

Fig. 17. Identification and merging of loops.

repeated until a NAND/NOR gate has been formed. The order in which loops are selected for merging is shown in Fig. 17. An equivalent input is obtained and finally the parameters of the equivalent inverter are calculated. This information is stored for the resulting transistors and is used when transient simulation of the elementary primitives begins.

It has to be noted that at each stage of the circuit the waveforms which are extracted as outputs are mapped to equivalent ramp inputs in order to feed them to the next stage. An equivalent ramp has a slope which is equal to 70% of the slope of the original waveform at the 50% point of the waveform [1]. The only timing information that is transferred to the transient analysis of the inverter macromodel (or analytical model) in addition to the inverter input is the starting point of conduction.

Simulation at elementary primitive level is performed exactly as in existing dynamic timing simulators [7], [17], [22]. An efficient and widely accepted timing simulator is ILLIADS2 which employs regionwise quadratic modeling (RWQ) for capturing of submicron MOS current models. The main shortcoming of fast timing simulators such as ILLIADS2 in the manipulation of complex gates are the large errors which are introduced by the merging of serially connected transistors. The proposed techniques can improve the accuracy of any dynamic timing simulator by handling more efficiently complex and NAND/NOR logic gates. Since the main algorithmic part of existing simulators will still be used and only a small overhead is added to improve merging of CMOS gates to an equivalent inverter the time penalty for this extra step is limited as shown in Table IV. In this table average execution times for SPICE, ILLIADS2 and the proposed method are given for two NAND gates and the two complex gate examples of Section VII. The time penalty indicates the percentage by which execution time is increased compared to ILLIADS2.

IX. CONCLUSIONS

In this paper, an efficient and accurate methodology for modeling CMOS gates is presented. The proposed method can be incorporated in existing timing simulators in order to improve their accuracy. The two basic structures of CMOS gates, serially and parallel connected transistors are examined. For each case an equivalent transistor is calculated so that each NAND/NOR gate can finally be replaced by an inverter whose output response matches that of the gate. The exact time point when each structure starts conducting is calculated, which has a significant effect on the accuracy of the model. Additionally, the parasitic behavior of parallel or serially connected transistors when they are conducting only the short-circuit current is modeled very efficiently. An input mapping algorithm which reduces every possible input pattern to an equivalent input signal is also introduced and the required weights of the transistor positions in a transistor chain are calculated. Finally, a method is presented for collapsing complex CMOS gates to equivalent NAND/NOR gates which then are further collapsed to an equivalent inverter. Comparisons with SPICE and ILLIADS2 have been presented to show the accuracy of the method and the improvement that can be gained by incorporating the reduction scheme in existing timing simulators. Results for deep submicron technologies (0.25 μm) show the effectiveness of the method for future processes.

REFERENCES

[1] N. Hedenstierna and K. O. Jeppson, "CMOS circuit speed and buffer optimization," *IEEE Trans. Computer-Aided Design*, vol. CAD-6, pp. 270–281, Mar. 1987.

[2] T. Sakurai and A. R. Newton, "Alpha-power law MOSFET model and its applications to CMOS inverter delay and other formulas," *IEEE J. Solid-State Circuits*, vol. 25, pp. 584–594, Apr. 1990.

[3] L. Bisdounis, S. Nikolaidis, and O. Koufopavlou, "Propagation delay and short-circuit power dissipation modeling of the CMOS inverter," *IEEE Trans. Circuits and Systems-I*, vol. 45, pp. 259–270, Mar. 1998.

[4] S. M. Kang and H. Y. Chen, "A global delay model for domino CMOS circuits with application to transistor sizing," *Int. J. Circuit Theory Applicat.*, vol. 18, pp. 289–306, 1990.

[5] B. S. Cherkauer and E. G. Friedman, "Channel width tapering of serially connected MOSFET's with emphasis on power dissipation," *IEEE Trans. Very Large Scale of Integration Syst.*, vol. 2, pp. 100–114, Mar. 1994.

[6] Y.-H. Shih and S. M. Kang, "Analytic transient solution of general MOS circuit primitives," *IEEE Trans. Computer-Aided Design*, vol. 11, pp. 719–731, June 1992.

[7] Y.-H. Shih, Y. Leblebici, and S. M. Kang, "ILLIADS: A fast timing and reliability simulator for digital MOS circuits," *IEEE Trans. Computer Aided Design of Integrated Circuits and Systems*, vol. 12, pp. 1387–1402, Sept. 1993.

[8] A. Dharchoudhury, S. M. Kang, K. H. Kim, and S. H. Lee, "Fast and accurate timing simulation with regionwise quadratic models of MOS I-V characteristics," *Proc. IEEE Int. Conf. on Computer-Aided Design (ICCAD)*, Nov. 1994, pp. 190–194.

[9] T. Sakurai and A. R. Newton, "Delay analysis of series-connected MOSFET circuits," *IEEE J. Solid-State Circuits*, vol. 26, pp. 122–131, Feb. 1991.

[10] A. Nabavi-Lishi and N. C. Rumin, "Inverter models of CMOS gates for supply current and delay evaluation," *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, vol. 13, pp. 1271–1279, Oct. 1994.

[11] J. M. Daga, S. Turgis, and D. Auvergne, "Design oriented standard cell delay modeling," in *Proc. Int. Workshop on Power and Timing Modeling, Optimization and Simulation (PATMOS)*, 1996, pp. 265–274.

[12] J.-T. Kong and D. Overhauser, "Methods to improve digital MOS macromodel accuracy," *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, vol. 14, pp. 868–881, July 1995.

[13] J.-T. Kong, S. Z. Hussain, and D. Overhauser, "Performance estimation of complex MOS gates," *IEEE Trans. Circuits Syst.-I: Fundamental Theory and Applications*, vol. 44, pp. 785–795, Sept. 1997.

[14] J.-T. Kong and D. Overhauser, "Digital timing macromodeling for VLSI design verification." Boston MA: Kluwer, 1995.

[15] H. J. M. Veendrick, "Short-circuit dissipation of static CMOS circuitry and its impact on the design of buffer circuits," *IEEE J. Solid-State Circuits*, vol. SC-19, pp. 468–473, Aug. 1984.

[16] A. Chatzigeorgiou and S. Nikolaidis, "Collapsing the transistor chain to an effective single equivalent transistor," *Proc. Design, Automation and Test in Europe Conference and Exhibition (DATE)*, Feb. 1998, pp. 2–6.

[17] A. Dharchoudhury, "Advanced techniques for fast timing simulation of MOS VLSI circuits," Ph.D. dissertation, Dept. Elec. Comput. Eng., Univ. Illinois, Urbana-Champaign, 1995.

[18] J. M. Rabaey, "Digital integrated circuits: A design perspective." Englewood Cliffs, NJ: Prentice-Hall, 1996.

[19] A. Bogliolo, L. Benini, G. De Micheli, and B. Ricco, "Gate-level power and current simulation of CMOS integrated circuits," *IEEE Trans. Very Large Scale Integration Syst.*, vol. 5, pp. 473–488, Dec. 1997.

[20] Y.-H. Jun, K. Jun, and S.-B. Park, "An accurate and efficient delay time modeling for MOS logic circuits using polynomial approximation," *IEEE Trans. Computer-Aided Design*, vol. 8, pp. 1027–1032, Sept. 1989.

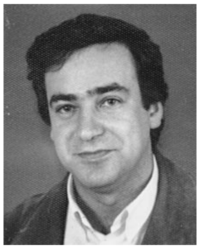
[21] Y. H. Shih, "Computationally efficient methods for accurate timing and reliability simulation of ultra-large MOS circuits," Ph.D. dissertation, Dept. Elec. Comput. Eng., Univ. Illinois, Urbana-Champaign, 1991.

[22] D. Overhauser, "Fast timing simulation of MOS VLSI circuits," Ph.D. dissertation, Dept. Elec. Comput. Eng., Univ. Illinois, Urbana-Champaign, 1989.

[23] J. K. White and A. Sangiovanni-Vincentelli, "Relaxation techniques for the simulation of VLSI circuits." Norwell, MA: Kluwer, 1987.

Alexander Chatzigeorgiou (S'95) was born in Thessaloniki, Greece, in 1973. He received the Diploma in electrical engineering from the Aristotle University of Thessaloniki, Thessaloniki, Greece, in 1996. He is currently pursuing the Ph.D. degree at the Computer Science Department of the same university working on timing and power modeling of digital integrated circuits.

He has held internship positions at Purdue University, West Lafayette, IN, the European Laboratory for Particle Physics (CERN), Geneva, Switzerland, and Imperial College, London, U.K. Since 1996 he has been with Intracom S.A. Greece as a Telecommunications Software Designer. His research interests include low-power VLSI design, computer architecture, and reconfigurable logic.



Spiridon Nikolaidis (S'89–M'93) was born in Kavala, Greece, in 1965. He received the Diploma and Ph.D. degrees in electrical engineering from Patras University, Greece, in 1988 and 1994, respectively.

Since September 1996 he has been with the Department of Physics of the Aristotle University of Thessaloniki, Thessaloniki, Greece, as a Lecturer in VLSI design. His research interests include CMOS gate propagation delay and power-consumption modeling, high-speed and low-power CMOS circuit techniques, power estimation of DSP architectures, and design of high-speed and low-power DSP architectures.

Ioannis Tsoukalas received the B.Sc., M.Sc., and Ph.D. degrees in physics from the Aristotle University of Thessaloniki, Thessaloniki, Greece, in 1968, 1975, and 1975, respectively.

He is currently Professor at the Computer Science Department of Aristotle University of Thessaloniki. His research interests include high-technology magnetic materials, electromagnetism, and electrodynamics.