# Delay Analysis of neuron-MOS and Capacitive Threshold-Logic

Peter Celinski, *Student Member, IEEE*, Said Al-Sarawi, *Member, IEEE* and Derek Abbott, *Senior Member, IEEE*

*Abstract*— **A model for the delay of neuron-MOS (neu-MOS) and Capacitive Threshold-Logic (CTL) based logic circuits is presented for the first time. It is based on the analysis of the basic neuron-MOS [1] and CTL gate structures [2]. A closed form analytic expression for the delay of the threshold gate is derived. A relation for the delay in terms of an ordinary CMOS inverter delay expressed as a function of the number of inputs to the threshold gate is presented. This relation is shown to be useful in comparing the delay of logic circuit designs based on neu-MOS or CTL and ordinary CMOS.**

*Keywords*— **neuron-MOS, capacitive-threshold logic, floating gate transistor**

## I. Introduction

In recent years, there has been renewed interest in Threshold Logic (TL), mainly as a result of the development of a number of successful inplementations of TL gates in silicon, including the neu-MOS and Capacitive Threshold-Logic architectures.

A threshold gate operates on binary variables and produces a binary output, and is functionally very similar to a hard limiting neuron, where a linear weighted sum is generated followed by a thresholding operation. The operation of a threshold gate is described by the following relations [3],

$$Y = 1 \quad \text{if} \quad \sum_{i=1}^{n} W_i X_i \geq T \tag{1}$$

$$Y = 0 \quad \text{if} \quad \sum_{i=1}^{n} W_i X_i < T \tag{2}$$

where $X_i \in \{0, 1\}$, $i = 1,\ldots,n$ are the binary input variables, $Y \in \{0, 1\}$ is the Boolean function realized by the threshold gate, and $W_i$ is the weight corresponding to the $i$th input variable $X_i$. $T$ represents the gate threshold and is generally a real number satisfying

$$0 \leq T \leq \sum_{i=1}^{n} W_i. \tag{3}$$

A Boolean function realized by such a threshold gate is called a threshold function. Threshold functions have some rather interesting and useful properties. All threshold functions are Boolean functions and all Boolean functions can be realized by a threshold gate network of depth at most two [3]. A TL gate can be programmed to realize many distinct Boolean functions by adjusting the threshold $T$. For example, an $n$ input TL gate with $W_i = 1, \forall i$ and $T = 1/2$ will realize an $n$ input *OR* gate. The same gate with $T$ set to $n$ - $1/2$ will realize an $n$ input *AND* gate. Generally, a threshold gate realizes a majority function, where by setting the threshold $T$ to an appropriate value, the output of the gate $Y$ is 1 if $k$ or more input variables are equal to 1, where $k \in \{0,\ldots,n\}$.

The threshold gate thus offers an increased computational capability over conventional Boolean gates, and it is possible to realize Boolean functions in TL using fewer gates with a reduced logic depth [3]. Threshold logic offers improved area density and higher speed logic circuits, particularly in applications involving a large number of input variables. It is the aim of this paper to quantify the delay of threshold gates.

In Section II of this paper, the basic structure and operation of the neu-MOS and Capacitive Threshold-Logic gates are reviewed. Section III introduces the delay model and Section IV presents the analysis of the delay of the threshold gates. In Section VI, simulation results of test circuits are presented and discussed. The results of this work are summarized in Section VII.

## II. Overview of Current TL Architectures

Threshold Logic gates can be realized using neuron-MOS [1] or the Capacitive Threshold-Logic Gate (CTL) method described in [2], both of which may be fabricated in standard CMOS processes. The circuit schematics of the CTL and neuron-MOS gates are shown in Fig. 1 and Fig. 2. The operation of the neu-MOS and CTL gates is discussed in detail in [1] and [2] respectively, and will be briefly summarized here. An $n$-input neu-MOS or CTL gate comprises $n$ weight-implementing capacitors ($C_i$) followed by one or more inverters which function as voltage comparators to generate the binary output. The first inverter in the chain will be referred to as the ***primary inverter***. The main difference between CTL and neu-MOS lies in the way the value of the threshold is set.

The CTL gate operates in a two-phase non-overlapping clock scheme consisting of a reset phase $\phi_R$ and an evaluate phase $\phi_E$. During the reset phase the row voltage $V_R$ is reset to the threshold voltage $V_{th}$ of the primary inverter, while the capacitor bottom plates are set to the reference voltage $V_{ref}$. During the evaluation phase, the row voltage is perturbed from $V_{th}$ by the inputs $X_i$ which now become

P. Celinski, S. Al-Sarawi and D. Abbott are with the Centre for High Performance Integrated Technologies & Systems (CHiPTec), The Department of Electrical and Electronic Engineering, The University of Adelaide, SA 5005, Australia, email: celinski@eleceng.adelaide.edu.au, alsarawi@eleceng.adelaide.edu.au, dabbott@eleceng.adelaide.edu.au.

0-7803-6542-9/00/$10.00 © 2000 IEEE

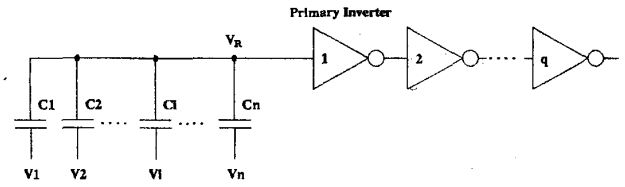


Fig. 1. The neu-MOS gate

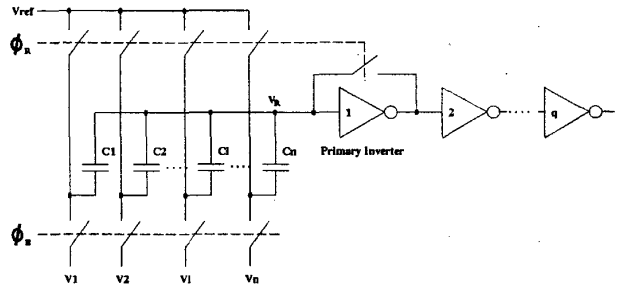


Fig. 2. The Capacitive Threshold Logic (CTL) gate

capacitively coupled onto the effectively floating input to the primary inverter. The magnitude of this perturbation is a function of $V_{ref}$, which effectively controls the threshold of the gate. The floating gate voltage, during the evaluation phase, is given by

$$V_R = V_{th} + \frac{\sum_{i=1}^{n} C_i(V_i - V_{ref})}{C_{tot}}. \quad (4)$$

In the case of the simpler, static neu-MOS gate, the gate threshold is adjusted via two threshold setting capacitors $C_{T1}$ connected to $Gnd$ and $C_{T2}$ connected to a threshold programming voltage $V_{ref}$. The floating gate voltage, is given by

$$V_R = \frac{\sum_{i=1}^{n} C_i V_i}{C_{tot}}. \quad (5)$$

In both (4) and (5), $C_{tot}$ is the sum of all capacitances in the gate, including parasitic capacitances. Although the operation of the neu-MOS gate is simpler, the maximum attainable fan-in is an order of magnitude less than that of the CTL gate because the CTL gate is not limited by the process variability which contributes to uncertainty in the inverter threshold voltage [2].

A number of applications of both gates have been proposed and developed, including a high-speed (31,5) parallel counter [4], a full adder cell with an area of approximately one half of the conventional CMOS design [5], [6], a high-speed, high-density multiplier cell [7]. An example of a commercially developed product, which uses neu-MOS, is the fingerprint sensing and encoding chip developed at Siemens [8]. Both neu-MOS and CTL based designs have illustrated the potential of threshold logic to significantly reduce the transistor count [9] and to increase the effective integration density.

## III. Developing the Delay Model

The importance of having a delay model for threshold gates lies in being able to predict the delay of circuits designed in neu-MOS or CTL, and to compare the speed performance of a threshold logic versus conventional CMOS design. Although both neuron-MOS and CTL have been shown to yield area efficient designs as compared to conventional CMOS, the dynamic performance of the threshold logic paradigm remains, until now, largely unexplored.

Despite its fundamental nature, the development of a delay model for neu-MOS or CTL circuits has received little attention in the literature. To the best knowledge of the authors, the only attempt at characterising the delay, of neu-MOS gates, has been a brief, qualitative description in [10]. The model developed here can be applied to structures such as neu-MOS, CTL and other structures where the input does not swing from rail to rail.

Although many articles have been written on the subject of CMOS circuit delay in response to a full rail to rail swing step (and other pulse shapes) [11], [12], none of this work is applicable to the threshold logic type structures discussed here.

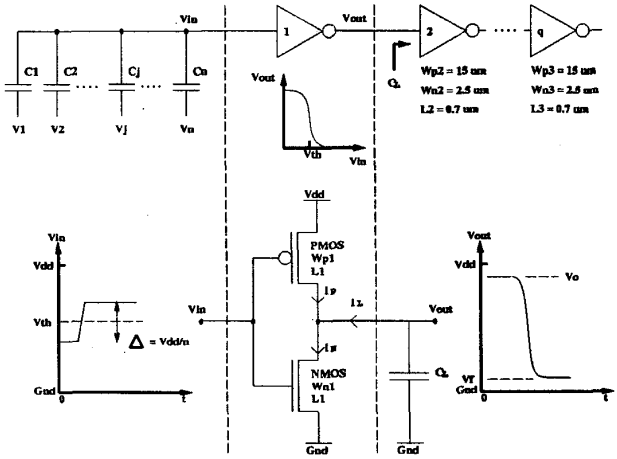


Fig. 3. Equivalent circuit of the neu-MOS gate and equivalent input voltage generated by capacitive input network

## IV. Analysis

In the case of both the CTL and neu-MOS gates, the floating gate voltage of the primary inverter in the comparator chain may be expressed as $V_{in} = V_{th} + \frac{\Delta}{2}$, where $V_{th}$ is the threshold of the primary inverter. As illustrated in Fig. 3, the worst case (ie. smallest) deviation of the input from the threshold $V_{th}$ is given by $\frac{\Delta}{2} = \frac{V_{DD}}{2n}$, where $n$ is the number of input weights. We will assume that the threshold $V_{th}$ is set to $V_{th} = \frac{V_{DD}}{2}$ which results in equal output rise and fall times. The derivations developed here are for a falling output response, ie. for a positive input voltage deviation about the threshold of the primary inverter. The analysis for a rising output response is similar. The input voltage may therefore be expressed

as $V_{in} = \frac{V_{DD}}{2} + \frac{V_{DD}}{2n}$. For n≥2, the input voltage causes both the PMOS and NMOS transistors to turn on and a DC current to flow in the inverter from $V_{DD}$ to *Gnd*. This occurs when the common gate is biased in the transition region $V_{TN} \leq V_{in} \leq \mid V_{DD} - V_{TP} \mid$, where $V_{TN}$ and $V_{TP}$ are the NMOS and PMOS threshold voltages, respectively. We will assume that $V_T = V_{TN} = V_{TP}$. Note that since the inverter threshold $V_{th}$ is chosen to be $V_{DD}/2$, the current gain, $\beta$, of both transistors is also the same. Referring to Fig. 3, the load current is given by $I_L = I_N - I_P$, where $I_L$ is the capacitive load discharge current, and $I_N$ and $I_P$ are the NMOS and PMOS currents, respectively. This is in contrast to the situation in conventional CMOS gates where most of $I_N$ is available to discharge the load.

From Fig. 3, the NMOS and PMOS transistor operating regions vary depending on the instantaneous value of the output voltage $V_{out}$. Immediately after the arrival of the input pulse, when $V_{in}+V_T \leq V_{out} \leq V_o$, the PMOS transistor is in the linear region and the NMOS is in saturation. Then as $V_{out}$ falls to $V_{in}-V_T \leq V_{out} \leq V_{in}+V_T$, both transistors enter into saturation. Finally, for $V_f \leq V_{out} \leq V_{in}-V_T$, the PMOS transistor operates in saturation and the NMOS in the linear region, where $V_o$ and $V_f$ are the initial and final values of the output voltage before and after the arrival of the input pulse, respectively.

The drain currents in the saturation region of operation for the NMOS and PMOS transistors are given by $I_{NS} = \frac{1}{2}\beta(V_{in} - V_T)^2$ and $I_{PS} = \frac{1}{2}\beta(V_{DD} - V_{in} - V_T)^2$, respectively, and the drain currents in the linear region are given by $I_{NL} = \beta\left((V_{in} - V_T)\, V_{out} - \frac{V_{out}^2}{2}\right)$ and $I_{PL} = \beta\left((V_{DD} - V_{in} - V_T)(V_{DD} - V_{out}) - \frac{(V_{DD}-V_{out})^2}{2}\right)$. The discharge of the effective load capacitance can be expressed as $C_L \frac{dV_{out}}{dt} = -I_L$. From this the gate delay may be written as

$$\Delta t = -C_L \int_{V_{out,i}}^{V_{out,f}} \frac{1}{I_L}\, dV_{out}, \tag{6}$$

where $V_{out,i}$ and $V_{out,f}$ are the 90% and 10% falling output voltage points. If we use the output voltage fall time ($\Delta t_f$) as a measure of gate delay, and if the output swings from $V_o$ to $V_f$ as shown in Fig. 3 then $V_{out,i} = 0.9(V_o - V_f) + V_f$ and $V_{out,f} = 0.1(V_o - V_f) + V_f$, corresponding to the 90% and 10% points on the output waveform. It is common to define gate delay as the average of the gate output fall and rise times, but since they have been made equal, the output fall time will be used. Defining $I_{L1} = I_{NS} - I_{PL}$, $I_{L2} = I_{NS} - I_{PS}$ and $I_{L3} = I_{NL} - I_{PS}$, we obtain

$$\Delta t_f = -C_L\left(\int_{V_{out,i}}^{V_{in}+V_T} \frac{1}{I_{L1}}\, dV_{out} + \int_{V_{in}+V_T}^{V_{in}-V_T} \frac{1}{I_{L2}}\, dV_{out} + \int_{V_{in}-V_T}^{V_{out,f}} \frac{1}{I_{L3}}\, dV_{out}\right) \tag{7}$$

which may be evaluated as

$$\Delta t_f = \frac{2C_L}{\beta}\left(\frac{2V_T}{(2V_{in}-V_{DD})(V_{DD}-2V_T)} - \frac{\arctan\left(\frac{-V_{out,i}+V_{in}+V_t}{\sqrt{(2V_{in}-V_{DD})(V_{DD}-2V_T)}}\right)}{\sqrt{(2V_{in}-V_{DD})(V_{DD}-2V_T)}} - \frac{\arctan\left(\frac{-V_{out,f}+V_{in}-V_t}{\sqrt{(V_{DD}-2V_{in})(V_{DD}-2V_T)}}\right)}{\sqrt{(V_{DD}-2V_{in})(V_{DD}-2V_T)}}\right). \tag{8}$$

As expected the delay of the TL gate is found to be proportional to the load capacitance $C_L$, inversely proportional to the current gain $\beta$, and inversely related to the supply voltage $V_{DD}$. A similar expression may be derived for a conventional CMOS inverter driven by a rising step input. The steps are similar to those followed above, except that now $V_{in} = V_{DD}$ and the PMOS transistor remains turned off over the entire range of $V_{out}$ during switching (ie. $I_L = I_N$) and the output voltage fall time is given by

$$\Delta t_{f,CMOS} = \frac{2C_L}{\beta}\left(\frac{V_T - 0.1V_{DD}}{(V_{DD}-V_T)^2} + \frac{\ln(-2\,V_T + 1.9V_{DD}) - \ln(0.1V_{DD})}{2(V_{DD}-V_T)}\right). \tag{9}$$

## V. The Normalized Delay

To compare the delay of gates we will use a method similar to that described in [13]. The delay of a TL gate is expressed in terms of the number of conventional CMOS inverter delays, of the same transistor size as the primary inverter in the TL gate comparator chain, and driving the same load $C_L$, by dividing expression (8) by (9). This removes the process and environmental dependencies, and allows sensible speed comparisons to be made.

Using typical 0.35$\mu$m, 3.3 V CMOS process values of $V_{DD}$=3.3V and $V_T$=$V_{DD}$/5, and setting $V_{out,i}$=0.8$V_{DD}$ and $V_{out,f}$=0.2$V_{DD}$, we can plot the analytically derived normalized fall time versus the number of inputs for $n$ = 1 to 20, as indicated by the line labelled *Analytic Model* in Fig. 4. These precess values also give $\Delta t_{f,CMOS}$=0.56($2C_L/\beta$). The values of 0.8$V_{DD}$ and 0.2$V_{DD}$ are chosen as the limits because they correspond to average values of the 10% and 90% points on the falling edge of $V_{out}$ over the range of $n$. For example, from Fig. 4, it can be seen that for a value of $n$ = 10, the analytically derived TL gate delay, contributed by the primary inverter, constitutes approximately 6 standard CMOS inverter delays of the same size as that primary inverter, and driving the same capacitive load.

## VI. Simulation Results

To verify the analytically developed delay model, the circuit shown in Fig. 3 was simulated using HSPICE Level 1 parameters for a 0.35 $\mu$m precess. All transistor gate lengths were set to 0.35 $\mu$m, all PMOS gate widths to 10 $\mu$m and all NMOS gate widths to 3 $\mu$m. The Level 1 simulated normalized delay plot is shown in Fig. 4, labelled *Level 1*. Comparing the Level 1 simulation result, and the analytically derived first order normalized delay, it can be seen that the derived model is within 7% of the Level 1 simulation result over the entire range of $n$.

To obtain a more realistic set of results for TL normalized delay, the circuit shown in Fig. 3 was simulated using

HSPICE Level 49 parameters for a number of different input voltage pulse magnitudes and primary inverter transistor sizes with a constant load. Again, the capacitive input network has been replaced by the equivalent input voltage step of variable amplitude, and a rise time of 100 ps. The size of the primary inverter was varied, and to maintain a constant load on the primary inverter, the size of the second and third inverters in the chain were kept constant. The lengths of the PMOS and NMOS transistors were made equal in each inverter and their width ratio was maintained such that the threshold voltage of each inverter was 1.65 V.

The normalized delay of the primary inverter loaded by the second inverter was then plotted as is also shown in Fig. 4, with the transistor sizes shown next to each of three lines line. The normalized delay contribution to the overall chain of three inverters by the second and third inverter was then also measured, and as expected it was found to be approximately equal to unity, for both. This means that these inverters contribute their normal delay, as in any standard CMOS circuit. The three lines with labelled

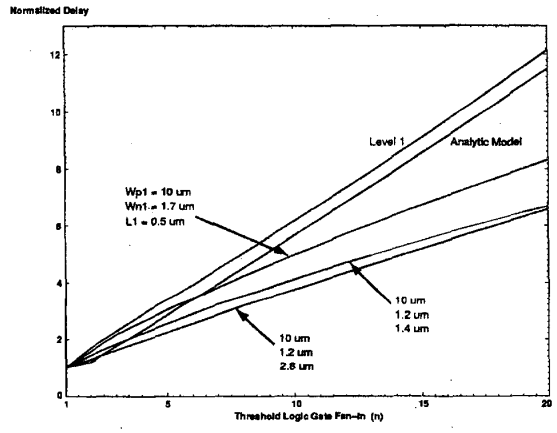


Fig. 4. Analytically derived normalized delay, simulated HSPICE Level 1 normalized delay and simulated HSPICE Level 49 normalized delay vs. fan-in for three different sizes of the primary inverter in the threshold gate

with transistor sizes in Fig. 4, correspond to three significantly different drive strengths of the primary inverter. The normalized delay for the three drive strengths differs by less than 2 normalized delay units over the entire range of fan-in shown, which indicates that the normalized delay of a threshold gate is only marginally dependent on the transistor sizes used. Hence an approximate, technology independent model may be derived. We conclude that the approximate normalized delay of a threshold gate is given by

$$Normalized\ Delay \;=\; 1 + 0.35n\,, \qquad (10)$$

where $n$ is the fan-in of the gate, and the factor 0.35 is the approximate average of the slopes of the three lines in Fig. 4 corresponding to the three Level 49 HSPICE simulations.

Comparing the slopes of the analytically derived model and and the Level 49 simulation results of Fig. 4, it can be seen that the analytic model predicts a higher than actual delay. Both, however, show an approximately linear relation between normalized delay and the gate fan-in.

## VII. Conclusions

In this paper, the basic structure and operation of the neu-MOS and Capacitive Threshold-Logic gates was reviewed and an analytically derived first order model for the delay of threshold logic gates was presented. The normalized analytical model was compared against HSPICE Level 1 and Level 49 simulations and an approximate normalized delay expression as a function of threshold gate fan-in was developed.


## Acknowledgements

The support the Australian Research Council and the Sir Ross and Sir Keith Smith Fund is gratefully acknowledged.


## References


[1] T. Shibata and T. Ohmi, "An intelligent MOS transistor featuring gate-level weighted sum and threshold operations," in *IEDM, Technical Digest*, New York, NY, USA, Dec 1991, IEEE.

[2] H. Özdemir, A. Kepkep, B. Pamir, Y. Leblebici, and U. Çiliniroğlu, "A capacitive threshold-logic gate," *IEEE JSSC*, vol. 31, no. 8, pp. 1141–1149, August 1996.

[3] S. Muroga, *Threshold Logic and Its Applications*, Wiley, New York, 1971.

[4] Y. Leblebici, H. Özdemir, A. Kepkep, and U. Çiliniroğlu, "A compact high-speed (31-5) parallel counter circuit based on capacitive threshold-logic gates," *IEEE JSSC*, vol. 31, no. 8, pp. 1177–1183, August 1996.

[5] K. Hirose and H. Yasuura, "A comparison of parallel multipliers with neuron MOS and CMOS technologies," in *Proceedings of IEEE Asia Pacific Conference on Circuits and Systems 96.* IEEE, November 1996, pp. 488–491.

[6] P. Celinski, J. F. Lopez, S. Al-Sarawi, and D. Abbott, "Complementary neu-GaAs structure," *IEE Electronics Letters*, vol. 36, no. 5, pp. 424–425, March 2000.

[7] W. Weber, S. J. Prange, R. Thewes, E. Wohlrab, and A. Luck, "On the application of the neuron MOS transistor principle for modern VLSI design," *IEEE Transactions on Electron Devices*, vol. 43, no. 10, pp. 1700–1708, October 1996.

[8] S. Jung, R. Thewes, T. Scheiter, K. F. Goser, and W. Weber, "A low power and high-performance CMOS fingerprint sensisng and encoding architecture," *IEEE JSSC*, vol. 34, no. 7, pp. 978–984, July 1999.

[9] B. Gonzalez, D. Abbott, S. Al-Sarawi, A. Hernandez, J. Garcia, and J.F. Lopez, "Efficient transistor count reduction for a low power GaAs A/D converter," in *Proceedings XIII Design of Circuits and Integrated Systems conference (DCIS'98)*, Madrid, Spain, November 1998, pp. 62–66.

[10] Ho-Yup Kwon, K. Kotani, T. Shibata, and T. Ohmi, "Low power neuron-MOS technology for high-functionality logic gate synthesis," *IEICE Trans. Electron.*, vol. E80-C, no. 7, pp. 924–930, July 1997.

[11] T. Sakurai and R. Newton, "Alpha power law MOSFET model and its application to CMOS inverter delay and other formulas," *IEEE JSSC*, vol. 25, no. 2, pp. 584–593, April 1990.

[12] A. I. Kayssi, K. A. Sakallah, and T. M. Burks, "Analytical transient response of CMOS inverters," *IEEE Transactions on Circuits and Systems*, vol. 39, no. 1, pp. 42–45, January 1992.

[13] S. Knowles, "A family of adders," in *Proceedings 14th IEEE Symposium on Computer Arithmetic*, 1999, pp. 30–34.