

# Path Delay Test Generation for Domino Logic Circuits in the Presence of Crosstalk\*

Rahul Kundu and R. D. (Shawn) Blanton  
Center for Silicon System Implementation  
ECE Department

Carnegie Mellon University  
Pittsburgh, PA 15213-3890

e-mail: {rkundu, blanton}@ece.cmu.edu

## Abstract

*A technique to derive test vectors that exercise the worst-case delay effects in a domino circuit in the presence of crosstalk is described. A model for characterizing the delay of a domino gate in the presence of crosstalk is developed and exploited by a new efficient timing analysis algorithm. The algorithm uses a single, breadth-first traversal to compute delays in the presence of crosstalk. Thus, it avoids the iterative methods commonly employed for static CMOS circuits. The timing analysis technique is used to generate test input vectors that exercise the worst-case delays of a multiplier circuit implemented using domino logic. Hspice simulation results demonstrate that the technique identifies test vectors that produce circuit delay that satisfy the targeted value in the presence of crosstalk.*

## 1 Introduction

Crosstalk affects domino circuit operation in two ways. First, excessive noise due to crosstalk can alter the logic value of an otherwise non-switching gate [1]. Second, crosstalk may affect the delay of a domino logic gate since transitions on neighboring lines may cause an increase or decrease in the effective gate delay. The effect of crosstalk on the delay of domino circuits is the subject of this paper.

Research work pertaining to the effect of crosstalk on circuit delay can be roughly classified into two categories. The first category [2, 3, 4, 5] describes how static timing analysis techniques for static CMOS circuits can be modified to cope with crosstalk effects. However, static timing analysis is inherently circuit functionality independent. The work in [6] is an exception, but it only uses circuit functionality to prune aggressors that cannot transition opposite to the victim. It does not consider circuit functionality for the purposes of path sensitization. To generate manufacturing tests for path delay faults, test vectors that exercise maximal delay in the circuit must be identified.

---

\*This work was supported by the Semiconductor Research Corporation under contract no. 982.001.

A second category of work [7, 8, 9] attempts to generate delay tests for a circuit considering crosstalk effects. A major limitation of these techniques is that the effect of crosstalk is analyzed for only one targeted victim line. However, in reality, the worst-case delay of a circuit may be exhibited when crosstalk effects are distributed along a sensitized path. Hence, a delay test generation methodology should consider crosstalk effect on the delay of all signal lines along a sensitized path, rather than a single victim line. This is a complicated problem for static CMOS circuits because the delay of a victim line is dependent upon the transition of lines physically adjacent to the victim line. Similarly, the transition of adjacent lines are also dependent on the transition of lines that are adjacent to them, thus creating a chicken-and egg problem. Most proposed solutions use an iterative timing analysis technique [3, 4, 5]. In addition, analyzing crosstalk is further complicated since static CMOS circuits can have static hazards, a fact considered in [6] but not accounted for in other work [7, 8, 9].

Finally, there is work [10, 11, 12] that has investigated the timing analysis of domino logic circuits. However, they have not addressed the influence of crosstalk on the delay of domino gates.

In this paper, we confine ourselves to “full-domino” circuits, that is, circuits composed solely of domino gates. Since full-domino circuits are unate, they offer several modeling simplifications: (1) all transitions are monotone (*i.e.* 0→1 transitions) and (2) there are no static hazards. Since transitions occur only in one direction, a line can only be sped up by switching adjacent lines. Also, our modeling and analysis of delay reduction reveals that speedup can only occur if adjacent lines transition prior to the victim gate. We show that these two properties can be exploited to compute the delays in a domino circuit in a single, breadth-first traversal of the circuit, implying that the iterative procedures used for static CMOS circuits are unnecessary. We use our delay computation scheme in a timed ATPG framework to identify vectors that cause the worst-case delay of a domino circuit.

The subsequent sections of the paper are organized as follows. Section 2 formulates the problem of analyzing

crosstalk on the delay of domino circuits. Specifically, Section 2.1 describes the assumptions adopted concerning the analyzed circuit. Section 2.2 presents our model of domino logic delay, while Section 2.3 describes how crosstalk impacts the delay of a domino logic gate. Based on the analysis in Section 2.3, Section 3 presents a breadth-first analysis strategy to perform timing analysis of a domino circuit in the context of crosstalk. Section 4 describes the results obtained from the application of our method to a multiplier circuit and their validation using Hspice simulations. Finally, Section 5 summarizes our current work and presents directions for future work.

## 2 Domino Gate Delay

This section describes a model for domino circuit delay. Our delay analysis is limited to circuits consisting of only domino gates with characteristics that are elaborated upon in the upcoming subsections.

### 2.1 Preliminaries

The following assumptions are adopted concerning the domino circuit under analysis. It is assumed that the circuit to be analyzed is a full-domino circuit, the only static gates used are the inverters that are pushed to the primary inputs. We also assume that significant cross-coupling capacitance is present only at the outputs of the domino gate and the internal dynamic nodes of a gate have negligible cross-coupling capacitance. The cross-coupling capacitance to the dynamic node is assumed insignificant since domino circuits are realized using standard cells, where a single domino gate equates to one standard cell. Correspondingly, all transistors within a cell are sized and placed with short intra-cell connections. Hence, the transistors within a cell are typically placed close together, ensuring that the dynamic node cannot have significant coupling capacitance. Another assumption is that the effect of noise induced by external (static CMOS gates outside the domino netlist of concern) signals or signals from the other phase of domino operation are neglected. The input-to-output delay<sup>1</sup> of a domino gate is defined as the interval of time between the time instant when the input begins to transition to the time instant when the output begins to rise. It is also assumed that the capacitive and resistive parasitics of the circuit have been extracted from the layout resulting in a SPICE netlist; this netlist serves as the starting point of our analysis.

### 2.2 Coupling-Free Delay Analysis

Consider a domino AND gate that transitions from 0→1 due to a transition at its input A. Figure 1a shows line A driven

<sup>1</sup>This definition of delay is different from the conventional definition that equates gate delay to the time between the 50% transition points of the input and output; however our definition greatly simplifies analysis and can be easily mapped to the conventional definition.

by a domino gate DR, where MN1 and MP1 are the transistors of the output inverter of DR. For simplification, assume that the input (DR\_D) to the driver inverter transitions from  $V_{DD}$  to 0 at  $t=0$ , implying MP1 is in saturation since its drain-to-source voltage  $v_{ds}$  is high for  $t>0$ . The effect of MN1 can be neglected since it is off due to its input being at ground for  $t>0$ . For this scenario, the capacitance at line A ( $C_A$ ) is being charged through the PMOS transistor MP1 which acts as a constant current source as shown in Figure 1b.  $C_A$  is the total capacitance associated with line A and includes the coupling capacitances of the adjacent signal lines, the substrate capacitance of line A, and the input capacitance of transistor MNA. Thus, the voltage  $v_A$  at line A initially increases linearly with time as shown in Figure 2 and its rate of change is given by

$$\frac{dv_A}{dt} = \frac{I_C}{C_A} \quad (1)$$

where  $I_C$  is the drain current of the transistor MP1 in saturation. The voltage at line A causes a discharge current  $I_d$  to flow through MNA, which discharges the dynamic node D causing the eventual transition of the gate output. The discharge current  $I_d$  can be expressed as a function of the voltage  $v_A$  [13]:

$$I_d = 0 \quad \text{if } v_A < V_T \quad (2)$$

$$I_d = K(v_A - V_T) \quad \text{if } v_A \geq V_T \quad (3)$$

The parameters  $K$  and  $V_T$  depend on several characteristics of the evaluate chain. For example, if the victim is the only transistor in the evaluate chain,  $K$  is large and  $V_T$  is small.  $K$  is reduced and  $V_T$  increases if there are other transistors in series with the victim transistor. Also, if other transistors in series transition simultaneously with the victim,  $K$  is further reduced and  $V_T$  increases even more. To address this variation of  $K$  and  $V_T$ , the maximum and minimum values of  $K$  and  $V_T$  are derived for every evaluate chain of every standard cell through Hspice simulations. The  $K$  and  $V_T$  extrema are then used in the analysis of maximum and minimum delays. From equations 2 and 3, it is evident that the discharge current  $I_d$  becomes non-zero only after  $v_A$  reaches  $V_T$ . This occurs for  $t > t_a = \frac{V_T}{\frac{dv_A}{dt}}$  as illustrated in Figure 2.

For  $t > t_a$ , the dynamic node begins discharging due to  $I_d$ . When the integral of  $I_d$  reaches  $C_{inv} \times V_{thp}$  ( $V_{thp}$  is the threshold voltage of MP2 and  $C_{inv}$  is the input capacitance of the output inverter), the output OUT of the domino gate increases and becomes non-zero. Assume OUT becomes non-zero at  $t = t_d$  as shown in Figure 2. Then  $t_d$  has to satisfy the relation,

$$\int_0^{t_d} I_d dt = C_{inv} \times V_{thp} \quad (4)$$

therefore, using equations 2 and 3

$$\int_{t_a}^{t_d} K \times (v_A - V_T) dt = C_{inv} \times V_{thp} \quad (5)$$

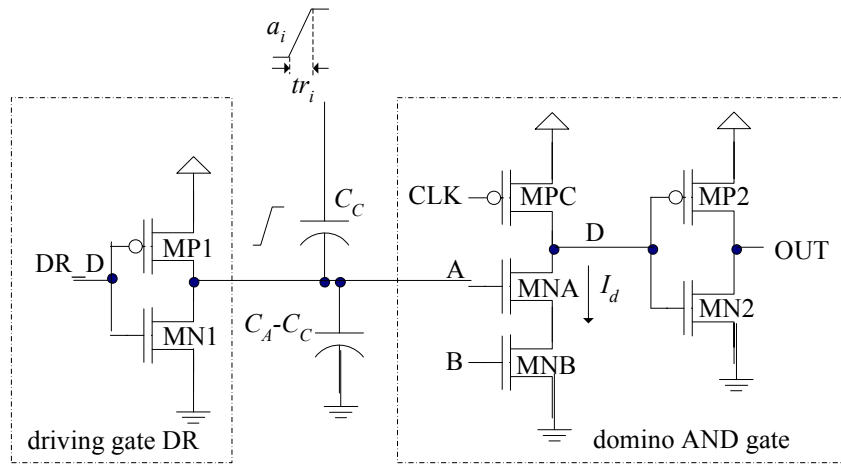


Figure 1: The domino AND gate transitions after the input to the inverter of the driver domino gate (DR.D) transitions from  $V_{DD}$  to 0. Line A charges through the PMOS transistor MP1 which acts as a constant current source equal to  $I_C$ .

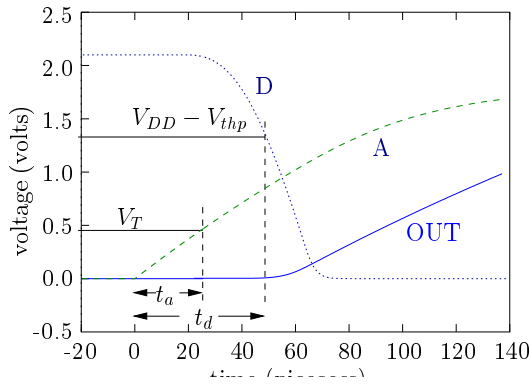


Figure 2: The signal waveforms for various lines when the domino AND gate in Figure 1 transitions.  $v_A$  reaches  $V_T$  at time  $t_a$  and OUT transitions at time  $t_d$ .

since  $v_A = V_T$  at  $t = t_a$  and  $v_A$  increases linearly with slope  $\frac{dv_A}{dt}$ . Equation 5 then reduces to

$$\int_{t_a}^{t_d} K \left( \frac{dv_A}{dt} \times (t) \right) dt = C_{inv} \times V_{thp} \quad (6)$$

where  $t_d$  is given by

$$t_d = t_a + \sqrt{\left( \frac{2 \times C_{inv} \times V_{thp}}{K \times \frac{dv_A}{dt}} \right)} \quad (7)$$

Thus, the total delay ( $t_d$ ) of the domino gate without any coupling can be estimated using equation 7.

### 2.3 Effect of Coupling on Delay

When one or more adjacent lines coupled to the victim can transition, the problem of determining gate delay becomes more complicated since the exact worst-case times for aggressor transitions must be determined.

Consider the effect of one aggressor  $a_i$  on the transition of the domino gate shown in Figure 3a. Assume  $a_i$  transitions with a slope of  $\frac{V_{DD}}{tr_i}$  at  $t = -t_b$  and there is a coupling capacitance of  $C_c$  between  $a_i$  and the victim. The glitch waveform due to  $a_i$  can be superimposed with the waveform due to DR. The glitch waveform due to the aggressor is given by

$$v_{coup}(t) = \frac{V_{DD} C_c R_{drv}}{tr_i} (1 - e^{-\frac{t-t_b}{R_{drv} \times C_A}}) \quad t \leq 0 \quad (8)$$

$$v_{coup}(t) = v_{coup}(0) + \frac{V_{DD} \times C_c \times t}{C_A \times tr_i} \quad t > 0 \quad (9)$$

where  $R_{drv}$  is the linear-region resistance of the transistor MN1.  $v_{coup}$  is shown in Figure 3b. Before  $t=0$ , the glitch waveform increases slowly since the driving NMOS transistor MN1 drains charge from line A. After  $t=0$ , the glitch waveform increases at a greater rate since the driver MN1 is off. (Although the PMOS transistor MP1 is on, it does not appear in equation 9 since the additional voltage due to crosstalk depends only on the small-signal resistance of the driver transistor [14], which is significant when the PMOS is in saturation. This point is further illustrated using Figure 4. The model of the overall circuit is shown in Figure 4a. The response of this circuit can be obtained through the superposition of two responses. First, the waveform that appears due to the driver PMOS without coupling is shown in Figure 4b. Second, the glitch induced at line A by the switching aggressor with the PMOS transistor replaced by its small-signal resistance is shown in Figure 4c.) Thus, the glitch waveform due to the aggressor is greater, if the aggressor transitions after  $t=0$  since more of the glitch obeys equation 9 instead of equation 8. However, a late aggressor transition means less time to participate in the discharge of the dynamic node D. Thus, a tradeoff exists in determining worst-case transition time of an aggressor. Determining the alignment of aggressor transitions so that the complete discharge of the dynamic node occurs as early as possible is analyzed in the Appendix and is used in the next section to develop a domino-centric timing analysis algorithm.

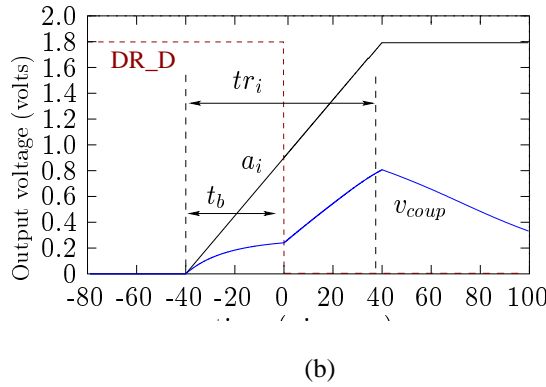
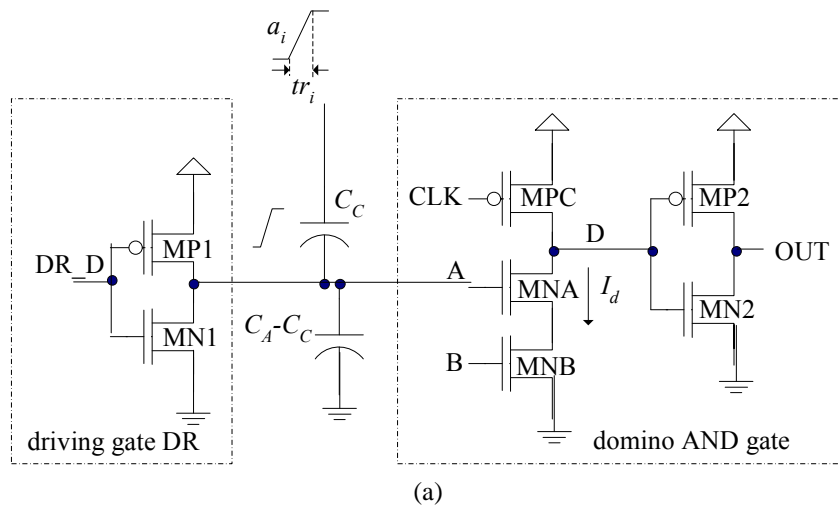


Figure 3: (a) An aggressor  $a_i$  coupled to the victim line A through capacitance  $C_c$ . (b) The waveforms involved in the discharge of dynamic node D are shown.  $a_i$  and DR\_D are the waveforms at line  $a_i$  and DR\_D, respectively, and  $v_{coup}$  is the voltage glitch at line A due to the transition on aggressor  $a_i$ .

### 3 Timing Analysis

From the arguments made in Section 2, two observations can be made. First, since the glitch waveform due to an aggressor switching from 0 to 1 is always positive, aggressor transitions can only hasten the transition of the victim gate. Thus, the maximum delay of a domino gate occurs when all the adjacent lines coupled to its inputs do not transition. We can therefore calculate the maximum possible delay of a gate by replacing all the adjacent lines with ground (assuming input patterns exist which prevent the aggressors from switching while the victim transitions). Second, if an aggressor line is to decrease the delay of a victim domino gate, it must transition before the victim output transitions. In fact, it is proven in the Appendix that if an aggressor  $a_i$  in Figure 3a is aligned to cause minimum input-to-output delay, it must transition even before the gate inputs transition. If  $a_i$  transitions significantly after the transition of the input, its effect on the input-to-output delay will be negligible. So it can be concluded that if  $a_i$  is to aid the minimization of the input-to-output delay significantly, its transition must begin before the transition of the input. Thus, if the mini-

imum gate-to-gate delays are evaluated from least to greatest (*i.e.* from smallest  $t_{min}$  to largest  $t_{min}$ ), the effect of coupling on delay can be estimated based on the delay of gates already evaluated.

The procedure for calculating the minimum delays and hence the minimum timing windows is given in Figure 5. The algorithm begins with the primary inputs, each having a  $t_{min}=0$ . The algorithm in Figure 5 maintains a linked list of signal lines called *Signal\_wvfrnt*. *Signal\_wvfrnt* contains gate output lines whose  $t_{min}$  have been calculated (arranged in ascending order of their  $t_{min}$  value). *Signal\_wvfrnt* is akin to a wavefront of signal lines; it propagates through the circuit from the primary inputs to the primary outputs. *Signal\_wvfrnt* initially contains the primary input lines (line 1 of Figure 5). A line  $l_A$  with minimum  $t_{min}$  from *Signal\_wvfrnt* is selected and the delay of all fanout gates of  $l_A$  are calculated. The minimum delay between  $l_A$  and  $n_A$  calculated in line 6 is based on the aggressor alignment criteria described in Section 2 and the Appendix. It is important to note that each aggressor  $a_i$  that could influence  $l_A$ -to- $n_A$  delay has a corresponding  $t_{min}(a_i) \leq t_{min}(l_A)$  and therefore has been already

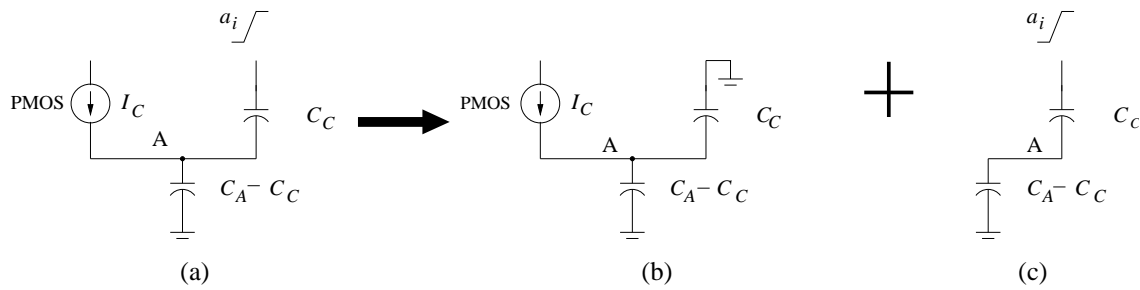


Figure 4: (a) There are two drivers that include the PMOS (in saturation, acting as a current source) and the switching aggressor  $a_i$ . The resulting waveform at line A is the superposition of the responses in Figures 4b and 4c. (b) The circuit is driven by the PMOS with a static aggressor. (c) The circuit is driven by aggressor  $a_i$  with the driver PMOS replaced by an open.

```

1 Initialize Signal_wvfrnt to contain the PIs.
2 while(Signal_wvfrnt is not empty)
3   { $l_A = \text{pop}(\textit{Signal\_wvfrnt})$ }
4   for every fanout gate  $G_A$  of  $l_A$ 
5     { $n_A = \text{output of gate } G_A$ }
6      $\text{delay}_{\text{new}} = \text{delay}(l_A \rightarrow n_A)$ 
7     if  $G_A$  is an OR gate
8       {if ( $t_{\min}(n_A) \geq \text{delay}_{\text{new}} + t_{\min}(l_A)$ )
9          $t_{\min}(n_A) = \text{delay}_{\text{new}} + t_{\min}(l_A)$ }
10    }
11    if  $G_A$  is an AND gate
12      {if ( $t_{\min}(n_A) \leq \text{delay}_{\text{new}} + t_{\min}(l_A)$ )
13         $t_{\min}(n_A) = \text{delay}_{\text{new}} + t_{\min}(l_A)$ }
14      }
15    Insert ( $n_A, t_{\min}(n_A)$ ) in Signal_wvfrnt
16  }
17 }

```

Figure 5: Pseudocode for static timing analysis.

analyzed. Once the delays of all the fanout gates from  $l_A$  are calculated, the minimum transition time ( $t_{\min}$ ) of each gate ( $n_A$ ) is updated based on its logic function (lines 7 to 14). Finally, the newly evaluated lines with their updated minimum transition times are added to the sorted list *Signal\_wvfrnt* in line 15.

## 4 Simulation Results

Validation of our ideas utilize a dual-rail, domino Wallace multiplier circuit [15] implemented in a 2-metal, 0.18 $\mu\text{m}$ , 1.8V technology. A layout of the multiplier was generated automatically using an industrial place and route tool. A netlist containing parasitic capacitances was extracted from the layout using the tool Space [16].

Static timing analysis was applied using our method and the iterative methods described in [4]. For the iterative method, five iterations were required for the timing windows to converge. Our method, on the other hand, requires only a single iteration and consumed 0.23 seconds of CPU time. Column 2 of Table 1 shows the percentage mismatch

Iteration no.	%Mismatch with our method	Iterative method CPU time (secs)
1	21	0.21
2	12	0.42
3	6	0.63
4	3	0.84
5	3	1.05

Table 1: Comparison of our method with the iterative method in terms of accuracy and time.

of the timing windows obtained using our method and those obtained using the traditional iterative methods, after every iteration. Column 3 of Table 1 shows the cumulative CPU time after each iteration. The conventional iterative method requires five iterations to converge and hence 5X the CPU time to reach the final values, which our method closely approximates in just a single iteration. Although just one example, we believe this illustrates the gains achievable by using a timing analysis approach developed to exploit the properties of domino logic. Also, it can be observed that the percentage mismatch between the timing windows obtained by our method with those obtained using the iterative method is about 3%. Hence our method sacrifices very little, if any, accuracy.

Our method was next extended to a dynamic timing analysis procedure to generate test vectors. The motivation for deriving the vectors using dynamic timing analysis is based on the following. Designers use static timing analysis to ensure that under nominal conditions all vectors in the circuit produce delay less than some delay value  $t_{stat}$ , thus allowing the circuit to be clocked with a time period equal to  $t_{stat}$ . However, in the presence of extreme manufacturing variations or defects that are likely difficult to model or predict, the circuit may exhibit a delay greater than  $t_{stat}$  causing incorrect circuit operation. To detect such failures, we would like to apply vectors that can detect any circuit delay that surpasses  $t_{stat}$ . The most likely circuit paths to exceed  $t_{stat}$  under manufacturing perturbations are those which (1) are sensitizable and (2) when sensitized, exhibit delay nearly equal to  $t_{stat}$  under nominal conditions (*i.e.* the critical paths). Thus, it is necessary to identify vectors that exercise the circuit to exhibit delays close to  $t_{stat}$  under nom-

Test vector no.	Predicted max delay	Delay from Hspice	Test vector generation time (secs)
1	278	269	1.2
2	267	250	1.4
3	254	234	1.8
4	252	220	1.2
5	248	227	1.4
6	244	229	1.2

Table 2: Test vectors obtained using dynamic timing analysis.

inal conditions because under manufacturing perturbations those vectors may cause the circuit delay to exceed  $t_{stat}$ . Generating the aforementioned vectors requires a generalized methodology to identify test vectors that cause circuit delay within a small delta of  $t_{stat}$ . These requirements are satisfied by our dynamic timing analysis technique which is described next.

For dynamic timing analysis, a timed-PODEM approach similar to the one proposed in [17] is used, where both logic values and min-max transition times ( $t_{min}, t_{max}$ ) are maintained for every signal line. Timed ATPG takes a gate-level netlist, a corresponding Hspice-level netlist, and a target circuit delay  $t_{target}$  as input. It then assigns logic values to the inputs of the netlist to justify the required time ( $t_{target}$ ) at some primary output. For every new logic value assignment at the primary inputs, all the logic values and timing windows are recalculated using a method similar to that described in Figure 5. Whenever the maximum possible time at the primary outputs falls short of  $t_{target}$  due to a new primary input assignment, a decision reversal (back-track) is performed. Whenever the minimum switching time at any primary output exceeds  $t_{target}$ , a test vector is obtained. Several runs of dynamic timing analysis, each with a larger value of  $t_{target}$  than the last and terminating with  $t_{stat}$ , leads to all possible vectors that exhibit delay near or greater than  $t_{stat}$ .

Using static timing analysis, the maximum and minimum circuit delay were found to be 282 ps and 70ps, respectively. Table 2 shows the results for various test vectors obtained from dynamic timing analysis. Each row in Table 2 provides information for a particular test vector. Column 1 shows the test vector number. Column 2 shows the maximum delay ( $t_{target}$ ) that was predicted by our dynamic timing analysis tool for that test vector. Column 3 shows the delay obtained from Hspice simulation of the netlist for that particular test vector. Column 4 shows the time taken by timed PODEM to generate the test vector. The Hspice simulation delays agree fairly well (to about 6 percent average mismatch) with our predicted maximum delays. The variability that exists between the maximum delays predicted by our tool and the delays obtained using Hspice is attributed to sources modeled in our methodology in only a min-max fashion (e.g. the effect of multiple gate inputs changing simultaneously is captured only at the min-max corners).

## 5 Summary and Future Work

In this paper, the effect of crosstalk on the delay of domino CMOS circuits was investigated. It was shown that the monotone nature of signal transitions in domino CMOS circuits can only cause crosstalk effects that reduce gate delay. In addition, significant delay reduction only occurs when coupled lines transition earlier than the victim gate inputs. We exploited these properties of domino circuits to develop a breadth-first timing analysis algorithm to compute signal delays in the presence of crosstalk. The timing analysis technique is more efficient than standard iterative techniques and at the same time sacrifices very little accuracy compared to existing iterative methods. We applied our novel timing analysis strategy in a timed ATPG framework to generate delay tests for a domino multiplier circuit under test. The delay test generation strategy was validated by comparing our delay prediction for a domino multiplier with Hspice simulations.

Future work will focus on extension of the methodology to (1) accommodate mixed static-domino circuits and (2) inclusion of the effect of noise due to external circuit lines that pass near or through the domino circuit under analysis. The extension to mixed static-domino circuits can be performed in two stages. The first stage is to generalize the output inverter of the domino gate to any static CMOS gate. This generalization will extend the methodology to any mixed static-domino circuit that has no significant coupling to the dynamic node. The second stage must examine coupling to the dynamic node. This will allow the application of the methodology to any mixed static-domino circuit. However, the second stage will be difficult since aggressors that have  $1 \rightarrow 0$  transitions can increase delay. For this more realistic circuit, the simplifications adopted in our analysis are likely invalidated implying that a judicious combination of the iterative methods and our methodology must be employed.

## Appendix

In this appendix, the problem of determining the alignment of aggressor transitions to minimize the victim domino gate delay is addressed. Consider an aggressor  $a_i$  coupled to a switching gate input line A through a coupling capacitance  $C_c$  as shown in Figure 6a. In Figure 6b, the aggressor begins its transition at some time instance  $t_i < 0$  and causes a glitch waveform  $v_{coup}(t)$  at A. As in Section 2.2, we assume that the driver inverter input DR\_D transitions at time  $t=0$ . Thus, the waveform  $v_A$  at line A is the superposition of two voltages:  $v_{coup}$  due to the switching aggressor and  $v_{DR}$  due to the transition of the driver inverter.  $v_{DR}$  increases linearly with time, starting from  $t=0$ , with a slope  $\frac{dv_A}{dt}$  given by equation 10, where  $I_C$  is the drain current of the aggressor's PMOS transistor MP1 and  $C_A$  is the total capacitance of the victim line.

$$\frac{dv_A}{dt} = \frac{I_C}{C_A} \quad (10)$$

In general, the aggressor can transition at any time. Therefore, the equations governing  $v_{coup}$  must be partitioned into several time intervals. For  $t \leq 0$ ,  $v_{coup}$  can be modeled using equations 11 and 12 for  $t < t_i + tr_i$  and  $t > t_i + tr_i$ , respectively,

$$v_{coup}(t) = \frac{V_{DD}C_c R_{drv}}{tr_i} (1 - e^{-\frac{t-t_i}{R_{drv} \times C_A}}) \quad (11)$$

$$v_{coup}(t) = \frac{V_{DD}C_c R_{drv}}{tr_i} (1 - e^{-\frac{tr_i}{R_{drv} \times C_A}}) (e^{-\frac{-(t-t_i-tr_i)}{R_{drv} \times C_A}}) \quad (12)$$

where  $R_{drv}$  is the linear-region resistance of the transistor driving the victim line,  $C_c$  is the coupling capacitance, and  $tr_i$  is the rise time of the aggressor. For  $t > 0$ ,  $v_{coup}$  increases according to equations 13 and 14, where  $tr_i^-$  is the time instance just before the aggressor completes its transition.

$$v_{coup}(t) = v_{coup}(0) + \frac{V_{DD} \times C_c \times t}{C_A \times tr_i} \quad t < t_i + tr_i \quad (13)$$

$$v_{coup}(t) = v_{coup}(tr_i^-) \quad t > t_i + tr_i \quad (14)$$

Assuming  $v_A$  reaches  $V_T$  at  $t = t_a$ , discharge of the dynamic node occurs for  $t > t_a$  according to equations 2 and 3. At  $t = t_d$  (see Figure 6b), the discharge from the dynamic node ( $\Delta Q = C_{inv} \times V_{thp}$ ) causes the output to transition. Our objective is to determine the value of  $t_i$  that minimizes  $t_d$ .

A decrease in  $t_i$  means more of  $v_{coup}$  will follow equation 11 and 12 which means the glitch waveform will decrease. On the contrary, if  $t_i$  is significantly increased, the domino gate is discharged due to  $v_{DR}$  alone and the glitch waveform does not contribute to the discharge. An illustrative example of this point is presented in Figure 7. In Figure 7a, the glitch waveform has  $t_i = -50$  ps. Although the glitch waveform  $v_{coup}$  is small, it partially discharges the dynamic node D as shown. In Figure 7b,  $t_i = -20$  ps, so the glitch waveform  $v_{coup}$  is large, but the glitch does not participate in the discharge since it begins after the output transitions. Given such tradeoffs, our purpose is to determine the optimal  $t_i = t_{opt}$  so that  $v_{coup} + v_{DR}$  discharges the dynamic node (*i.e.* causes a discharge of  $\Delta Q = V_{thp} \times C_{inv}$ , where  $V_{thp}$  is the threshold voltage of MP2 and  $C_{inv}$  is the capacitance of the dynamic node) as early as possible.

Our definition of optimal alignment  $t_{opt}$  is as follows: At  $t_i = t_{opt}$ , the aggressor transition is assumed to be optimally aligned causing the dynamic node to discharge at  $t = t_d$ . Then, if the aggressor transition time  $t_i$  is moved by an infinitesimally small time interval  $\Delta t$  in either direction (*i.e.* the past or the future), there cannot be any increase in the total discharge from  $t = -\infty$  to  $t = t_d$ .

**Theorem 1** *Domino gate delay  $t_d$  is minimal if and only if for each aggressor  $a_i$  coupled to the victim, the following relation is satisfied:*

$$\Delta s \times (t_d - t_a) = v_{coup}(t_d) - v_{coup}(t_a) \quad (15)$$

where  $v_{coup}$  is the glitch waveform due to a transition on  $a_i$ ,  $t_a$  is the time instant where the victim waveform  $v_A =$

$V_T$  (the voltage parameter of the victim transistor evaluate chain), and  $\Delta s = \frac{dv_{coup}(0^+)}{dt} - \frac{dv_{coup}(0^-)}{dt}$  is the change in slope of  $v_{coup}$  at  $t = 0$ .

**Proof:** We will show both necessity and sufficiency by contradiction. Assume that the aggressor transition in Figure 6 is aligned to produce minimum  $t_d$ , that is, it is not possible to increase  $\int_{-\infty}^{t_d} I_d dt$  by changing  $t_i$ , the transition time for aggressor  $a_i$ .

Now assume the aggressor transition is pushed to the future by an infinitesimally small interval of time  $\Delta t$  causing a new coupling waveform  $v'_{coup}$  that starts at  $t_i + \Delta t$ .  $v'_{coup}(t)$  can be related to  $v_{coup}(t)$  using the following equations:

$$v'_{coup}(t) = v_{coup}(t - \Delta t) \quad t < 0 \quad (16)$$

Also, due to the abrupt discontinuity of slope at  $t = 0$ ,

$$v'_{coup}(t) = v_{coup}(t - \Delta t) + \Delta s \times \Delta t \quad t \geq 0 \quad (17)$$

It is assumed that the contribution due to  $v_{DR}$  remains unchanged for either  $v_{coup}$  or  $v'_{coup}$ . The discharge contribution of  $v_{coup}$  between  $-\infty$  and  $t_d$  is

$$\Delta Q = \int_{-\infty}^{t_d} I_d(v_{coup}(t)) dt \quad (18)$$

Using equation 3 from Section 2.2, equation 18 can be rewritten as

$$\Delta Q = \int_{t_a}^{t_d} K(v_{coup}(t) - V_T) dt \quad (19)$$

The discharge caused by ( $v'_{coup}$ ) between the times  $-\infty$  and  $t_d$  is

$$\begin{aligned} \Delta Q' &= \int_{-\infty}^{t_d} I_d(v'_{coup}(t)) dt \\ &= \int_{t_a}^{t_d} K(v'_{coup}(t) - V_T) dt \\ &= \int_{t_a}^{t_d} K(v_{coup}(t - \Delta t) - V_T + \Delta s \times \Delta t) dt \\ &= \int_{t_a - \Delta t}^{t_d - \Delta t} K(v_{coup}(t) - V_T) dt + \Delta s \Delta t (t_d - t_a) \\ &= \int_{t_a}^{t_d} K(v_{coup}(t) - V_T) dt - (v_{coup}(t_d) - v_{coup}(t_a)) \Delta t + \Delta s \Delta t (t_d - t_a) \\ &= \Delta Q - (v_{coup}(t_d) - v_{coup}(t_a)) \Delta t + \Delta s (t_d - t_a) \Delta t \end{aligned}$$

therefore

$$\Delta Q' - \Delta Q = -(v_{coup}(t_d) - v_{coup}(t_a)) + \Delta s (t_d - t_a).$$

If  $\Delta Q' - \Delta Q > 0$ , movement of the aggressor transition into the future will increase discharge. If  $\Delta Q' - \Delta Q < 0$ , movement of the aggressor into the past will increase discharge. Therefore, for optimal alignment,  $\Delta Q' - \Delta Q = 0$ , implying that  $v_{coup}(t_d) - v_{coup}(t_a) = \Delta s (t_d - t_a)$ .  $\square$

It can be shown that for any arbitrary waveforms, the condition can be generalized to

$\Delta s (t_d - t_a) = v_{coup}(t_d) - v_{coup}(t_a) + \sum (v_{coup}(t_b) - v_{coup}(t_a))$  where  $t_a$  and  $t_b$  refer to the time instants when the total waveform at line A crosses  $V_T$  ( $t_a$  refers to a time when  $v_A$  is increasing and  $t_b$  refers to a time when  $v_A$  is decreasing). The criteria in equation 15 is used to derive the optimal alignment of aggressor waveforms for minimum delay.

From equation 15, we can conclude that  $t_i$  must be less than 0 to minimize delay, that is, the aggressor transition

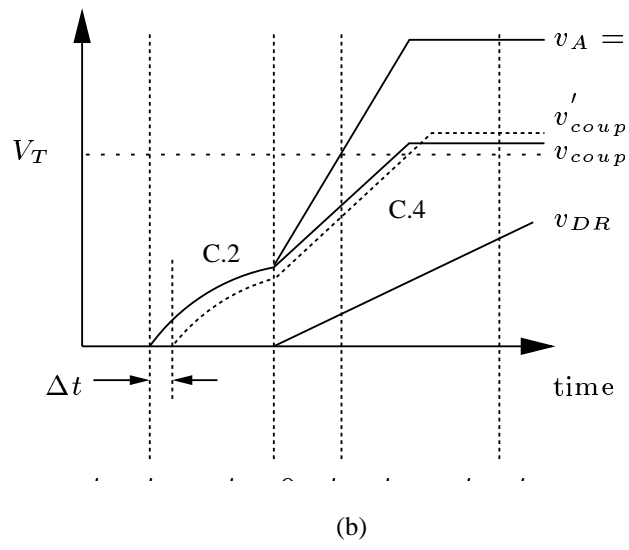
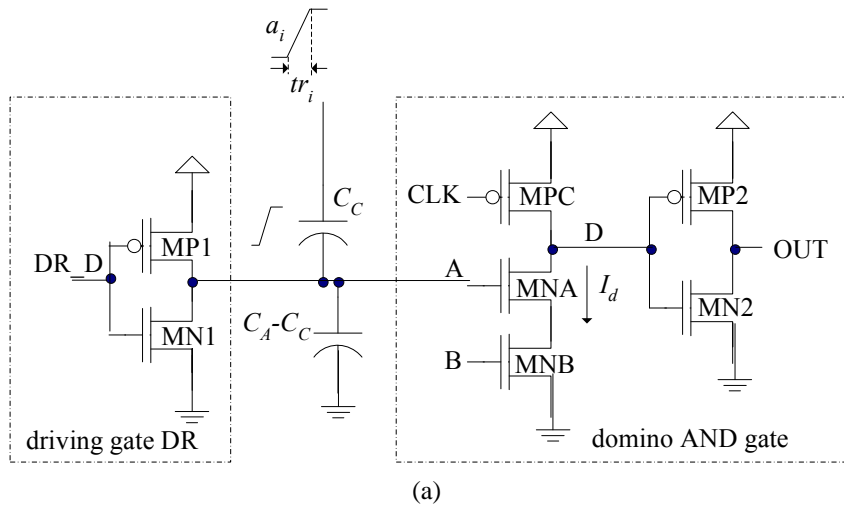


Figure 6: (a) A domino gate switches due to a transition on its input A and crosstalk due to aggressor  $a_i$ . (b)  $v_{DR}$  is the voltage waveform due to the driver gate DR.  $v_{coup}$  (equations 11, 12, 13 and 14) is the contribution due to the aggressor.  $v'_{coup}$  is another version of  $v_{coup}$  displaced by  $\Delta t$  to the right.  $v_A$  is the total waveform at line A.

should begin before the driving gate DR.D begins to transition. Specifically,  $v_{coup}(t_d) \geq v_{coup}(t_a)$  since  $v_{coup}$  (equations 13 and 14) and  $v_{DR}$  are non-decreasing functions. From equation 15,  $v_{coup}(t_d) - v_{coup}(t_a)$  is positive and since  $t_d - t_a$  is also positive due to  $t_d \geq t_a$ ,  $\Delta s$  is positive.  $\Delta s$  can only be positive if  $v_{coup}$  starts before  $t = 0$ , otherwise  $\Delta s = 0$ . Thus, if  $v_{coup}$  starts prior to  $t = 0$ ,  $t_i$  should be negative, implying the aggressor transition should be earlier than the driving gate. This conclusion is used in the timing analysis algorithm described in Section 3.

## References

- [1] R. Kundu and R. D. Blanton, "Timed Test Generation for Crosstalk Switch Failures in Domino CMOS Circuits," in *VLSI Test Symposium*, pp. 379–385, April 2002.
- [2] A. B. Kahng, S. Muddu, and E. Sarto, "On Switch-factor Based Analysis of Coupled RC Interconnects," in *Design Automation Conference*, pp. 79–84, June 2000.
- [3] R. Arunachalam, K. Rajagopal, and L. T. Pileggi, "Timing Analysis with Coupling," in *Design Automation Conference*, pp. 266–269, June 2000.
- [4] T. Xiao and M. Marek-Sadowska, "Worst Delay Estimation in Crosstalk Aware Static Timing Analysis," in *International Conference on Computer Design*, pp. 115–120, Sept. 2000.
- [5] S. S. Sapatnekar, "On the Chicken-and-Egg Problem of Determining the Effect of Crosstalk on Delay in Integrated Circuits," in *Electrical Performance of Electronic Packaging*, pp. 245–248, 1999.
- [6] R. Arunachalam, R. D. Blanton, and L. T. Pileggi, "False Coupling Interactions in Static Timing Analysis," in *Design Automation Conference*, pp. 726–731, June 2001.

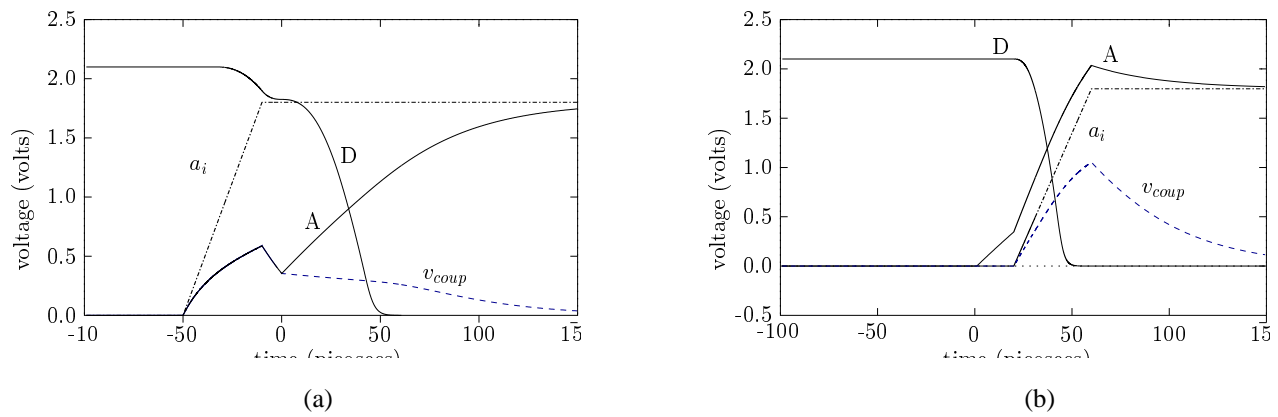


Figure 7: Example waveforms of various signal lines in the domino gate of Figure 6a when aggressor  $a_i$  transitions (a) early (50ps before the driver switches) and (b) late (20ps after the driver switches).

- [7] S. Nazarian and H. Huang and S. Natarajan and S. K. Gupta and M. A. Breuer, "XIDEN: Crosstalk Target Identification Framework," in *International Test Conference*, pp. 365–374, Sept. 2002.
- [8] W. Chen and S. K. Gupta and M. A. Breuer, "Test Generation for Crosstalk Induced Delay in Integrated Circuits," in *International Test Conference*, pp. 191–200, Sept. 1999.
- [9] L. Chen and T. Mak and S. K. Gupta and M. A. Breuer, "Crosstalk Test Generation on Pseudo Industrial Circuits: A Case Study," in *International Test Conference*, pp. 548–557, Sept. 2001.
- [10] D. Van Campenhout, T. Mudge, and K.A. Sakallah, "Timing Verification of Sequential Domino Circuits," in *International Conference on Computer-Aided Design*, pp. 127–132, Oct. 1996.
- [11] H. Cheng and J. A. Abraham, "Timing Analysis of Dynamic Logic Circuits," in *International Test Conference*, pp. 628–631, June 2000.
- [12] S. Natarajan, S. K. Gupta, and M. A. Breuer, "Switch-level Delay Test of Domino Logic Circuits," in *International Test Conference*, pp. 367–376, Oct. 2001.
- [13] D. Somasekhar, S. H. Choi, K. Roy, Y. Ye, and V. De, "Dynamic Noise Analysis in Precharge-Evaluate Circuits," in *Design Automation Conference*, pp. 243–246, June 2000.
- [14] S. Sirichotiyakul, D. Blaauw, C. Oh, R. Levy, V. Zolotov, and J. Zuo, "Driver Modeling and Alignment for Worst-Case Delay Noise," in *Design Automation Conference*, pp. 720–725, June 2001.
- [15] B. Ramasubramanian, H. Schmit, and L. R. Carley, "Mixed-Swing Quadrail for Low Power Dual-Rail Domino Logic," in *International Symposium on Low Power Electronics and Design*, pp. 82–84, Aug. 1999.
- [16] N. P. Van der Meijs and A. J. Van Genderen, "An Efficient Finite Element Method for Submicron IC Capacitance Extraction," in *26th Design Automation Conference*, pp. 678–681, June 1989.
- [17] S. Devadas, K. Keutzer, and S. Malik, "Computation of Floating Mode Delay in Combinational Circuits: Theory and Algorithms," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 12, No. 12, pp. 1913–1923, Dec. 1993.