

Delay Variation Tolerance for Domino Circuits

Kai-Chiang Wu, Cheng-Tao Hsieh, Shih-Chieh Chang

Department of CS, National Tsing Hua University, Hsinchu, Taiwan

Alexe@nthucad.cs.nthu.edu.tw, jdshieh@nthucad.cs.nthu.edu.tw, scchang@cs.nthu.edu.tw

ABSTRACT

Factors of delay variation, such as process variation and noise effects, may cause a manufactured chip to violate the pre-specified timing constraint. In this paper, we propose a novel re-synthesis technique to tolerate delay variation for domino circuits. Note that the slacks of nodes along critical paths are zero; any delay addition to those zero-slack nodes will worsen the final performance of a circuit. Our basic idea is to increase the slacks of nodes in the critical region by appending a redundant auxiliary sub-circuit to the original circuit. The auxiliary sub-circuit can cause critical paths to become false paths or imperceptible paths [7] so as to improve the capability of delay variation tolerance. Experimental results are very encouraging.

1. Introduction

Circuit delay in advanced technologies becomes increasingly sensitive to process variation and noise [1][2]. Those factors would cause a circuit's performance to fluctuate and, in the worst case, a timing violation may occur. Especially for high performance designs, timing critical regions are often implemented in domino circuit style. As a result, the delay variation problem becomes a critical issue for domino circuits. In this paper, we propose a re-synthesis method to tolerate delay variation for a domino circuit.

The slacks of nodes along critical paths are zero; any delay addition to those zero-slack nodes will worsen the final performance of a circuit. The degree of delay variation tolerance in [3] is formulated by the concept of slacks. The authors proposed a re-synthesis technique which incorporates Triple Module Redundancy (TMR) like structure into a circuit so that the circuit can tolerate a given range of delay uncertainty. Their results show that by adding 40% area overhead, a certain degree of variation tolerance can be achieved for static CMOS circuits.

Unlike static circuits, dynamic domino circuits operate in two phases: the pre-charge phase and the evaluation phase. To avoid the racing problem, domino circuits require all

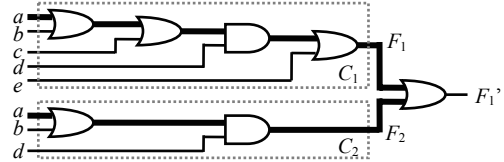


Figure 1: An example demonstrating delay variation tolerance for a domino circuit.

signals, except primary inputs, to have only rising transitions during the evaluation phase. Due to the rising-transition-only property, delay variation tolerance in domino circuits is much easier to accomplish than that in static circuits. In other words, directly applying the same method as [3] to domino circuits may have an unnecessarily large area.

Our basic idea of delay variation tolerance is to combine a (target) domino circuit with a redundant auxiliary sub-circuit, which is illustrated in the following example. A domino circuit C_1 in Figure 1 implements logic function $F_1 = (a+b+c)d+e = ad+bd+cd+e$. For delay tolerance on C_1 , we construct an auxiliary circuit C_2 implementing $F_2 = (a+b)d = ad+bd$ and generate a new output $F_1' = F_1+F_2$. Because the on-set of F_1 apparently covers that of F_2 (i.e., $F_1 \supseteq F_2$), the new function $F_1' = F_1+F_2$ is identical to the original function F_1 . We will show that appending circuit C_2 does not change the functionality of the original circuit but has the effect of delay tolerance. Consider an input pattern $(a, b, c, d, e) = (1, 0, 0, 1, 0)$ which induces transitions propagating along critical (highlighted) paths in both C_1 and C_2 . Since both F_1 and F_2 have only rising transitions and feed into an OR gate, whichever rising signal arrives earlier will dominate the OR gate's output. In other words, the earlier arrival of either F_1 or F_2 determines the output value instead of the late one. Delay variation tolerance is consequently achieved because late arriving signals will not influence the whole circuit's delay.

In addition to proposing a new structure for domino circuits, this paper also derives novel theorems which allow us to carry out a smaller C_2 than those from [3]. We also show that delay variation tolerance can be applied to both internal nodes and primary outputs to achieve better results.

The experiments show that the area overhead of our method is less than a half of that from [3]. Also, we have performed Monte-Carlo experiments. The delay of a gate is given as a probability density function similar to [6]. The results show that about 77% of samples of the original circuit C6288 can meet a certain delay requirement; however, about 99% of samples of the corresponding re-synthesized circuit can satisfy the same requirement.

We would like to mention that there have been several studies [4][5][7] attempting to optimize the statistical timing results by gate sizing techniques. These methods require statistical models to be given for all gates. We think that if statistical distributions of gates and statistical correlations among gates are given precisely, methods [4][5][7] can be more efficient than redundant structures as described by this paper. On the other hand, this paper does not assume any statistical timing models. Therefore, our method will be better when accurate statistical models are not available. For example, very few chips are manufactured for new technologies, such as 90nm and 65nm, so it is difficult to gather or to verify statistical models for new technologies. In addition to process variation, delay variation due to noise issues, such as IR drop, is not easy to model.

2. Delay Variation Tolerance in Duplex Domino Systems

Let a domino circuit C_1 be under consideration for delay variation tolerance. A duplex system is shown in Figure 2, which consists of the original circuit C_1 , its duplication C_2 , and an OR gate taking the outputs of C_1 and C_2 as its inputs. There are two properties for a duplex domino system. First, it performs the same function as the original circuit. Secondly, a transition traveling along a path in C_1 also travels along the mirror path in C_2 . Since a domino circuit has only 0-to-1 transitions, the OR gate's output transits from 0 to 1 when the earlier transition arrives. Any delay increase which is postponing either C_1 or C_2 will not affect the eventual timing result. Therefore, each node (except the OR gate) in the

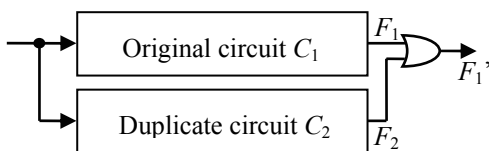


Figure 2: A duplex system.

duplex system has an infinite slack. Still, there is more than 100% area overhead for a duplex system, making the duplex system impractical. In the remainder of this paper, we will explain how to reduce C_2 while maintaining a given degree of delay variation tolerance.

Let us first define the degree of delay variation tolerance. The degree of delay variation tolerance can be quantified with the smallest slack of gates/wires in a circuit. A circuit is defined as having d_t delay tolerance [3] if the slack of each gate/wire is at least d_t . Given a delay tolerance value d_t and a circuit, our objective is to re-synthesize the circuit so that every gate/wire in the new circuit can tolerate at least delay variation d_t . The slack of each gate/wire in a duplex domino system is infinite (i.e., $d_t = \infty$), which is over-protective for the delay variation problems. In general, delay tolerance of 10%~20% of the original circuit delay is sufficient for our consideration of process variation and noise effects.

3. Re-synthesis for Delay Variation Tolerance

To accomplish a given delay tolerance value d_t without adding too much area overhead, a practical scheme originated from a duplex system is proposed. Our re-synthesis steps are as follows. (1) Begin with a duplex system in Figure 2. (2) Remove and modify some wires in C_2 for area reduction while maintaining the tolerance value. We now discuss how to perform wire removal and modification in C_2 .

Assume the original circuit C_1 implements the logic function F_1 , the redundant auxiliary circuit C_2 implements F_2 , and the combinative output is $F_1' = F_1 + F_2$. If the on-set of F_2 is a sub-set of that of F_1 (i.e., $F_2 \subseteq F_1$), we can preserve the original functionality of F_1' ($= F_1 + F_2 = F_1$). While there are many possible Boolean functions which satisfy $F_2 \subseteq F_1$, we only consider wire removal in C_2 to reduce the on-set of F_2 . Before presenting our theorems, we describe what wires in C_2 can be removed to maintain $F_2 \subseteq F_1$ in the following lemma.

Lemma 1: All direct input wires to OR gates in C_2 are redundant and can be **simultaneously** removed.

Proof: Omitted.

Let the delay tolerance value be d_t . A path is called a d_t -critical path if its delay is greater than the timing requirement minus d_t . In other words, d_t delay increment on a d_t -critical path will cause the path's delay to exceed the

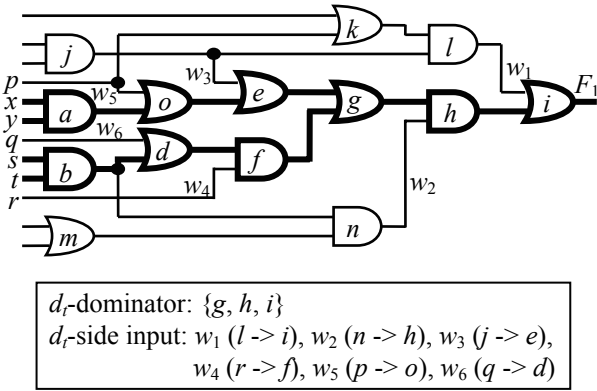


Figure 3: The original circuit.

timing requirement. In addition, we say that a node is a d_r -critical node if it is along a d_r -critical path. One can also find that the slack of a d_r -critical node is smaller than d_r . Consider the example in Figure 3, where the delay of each gate is 1. Suppose the delay tolerance d_r is 1 and the timing requirement d_r is 6. In this example, the timing requirement is equal to the length of the longest path. Path $\{s-b-d-f-g-h-i\}$ is a d_r -critical path because its delay is 6, greater than d_r-d_i ($6-1=5$). In fact, all highlighted paths in Figure 3 are d_r -critical paths. Node g is a d_r -critical node because it is along a d_r -critical path. Besides, all highlighted nodes $\{a, b, o, d, e, f, g, h, i\}$ are d_r -critical nodes. Note that the slacks of these d_r -critical nodes are 0, smaller than d_r .

A node is defined to be a d_r -dominator if it is a d_r -critical node and all d_r -critical paths to a primary output must pass through the node. A wire $n_1 \rightarrow n_2$ is a d_r -side input if node n_1 is not a d_r -critical node but node n_2 is a d_r -critical node. In the same example, nodes $\{g, h, i\}$ are d_r -dominators because all d_r -critical paths to the primary output must pass through these nodes. Wire w_1 ($l \rightarrow i$) is a d_r -side input because node l is not a d_r -critical node but node i is. Similarly, wires $\{w_2, w_3, w_4, w_5, w_6\}$ are also d_r -side inputs.

Theorem 1 [3]: A d_r -side input wire w to an OR d_r -dominator in C_2 can be removed (replaced by a non-controlling value, i.e. a logic 0) without violating the requirement of d_r delay tolerance.

Proof: Omitted.

Take the original circuit in Figure 3 as an example. A duplex system can be constructed by two duplicates (C_1 and C_2) of the original circuit. According to Theorem 1, we can remove wire w_1 in C_2 by replacing it with a logic 0 since it is a d_r -side input to OR d_r -dominator i .

We say that a node is a *transitive fanout* of wire w if

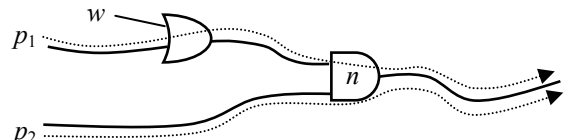


Figure 4: Node n is the AND-converging node of d_r -critical paths p_1 and p_2 .

there is a path from wire w to the node. In addition, in Figure 4, two paths may “converge” on a node, which is called the *converging node* of the two paths. We define node n to be the *AND-converging node* of two d_r -critical paths if these two paths converge on node n and node n is an AND gate. For example in Figure 4, d_r -critical paths p_1 and p_2 converge on an AND gate n so node n is the AND-converging node of p_1 and p_2 . We have the following theorem.

Theorem 2: Let wire w be a d_r -side input to an OR gate in C_2 . If there is no AND-converging node of d_r -critical paths in wire w 's transitive fanout, wire w can be removed without violating d_r delay tolerance.

Proof: Omitted.

For example, wire w_3 ($j \rightarrow e$) can be removed according to Theorem 2. First, wire w_3 is a d_r -side input to OR gate e . The transitive fanout nodes of w_3 consist of $\{e, g, h, i\}$, among which OR gate g is the only converging node of d_r -critical paths. Since there is no AND-converging node in w_3 's transitive fanout, wire w_3 in C_2 can be removed. In fact, wires $\{w_3, w_5, w_6\}$ all satisfy the condition in Theorem 2 so wires $\{w_3, w_5, w_6\}$ in C_2 are removable. The resultant circuit after removing wires w_1 (by Theorem 1), $w_3, w_5,$ and w_6 in C_2 is shown in Figure 5.

We can also adopt signal sharing as in [3] to further reduce the area overhead. Two signals which implement the

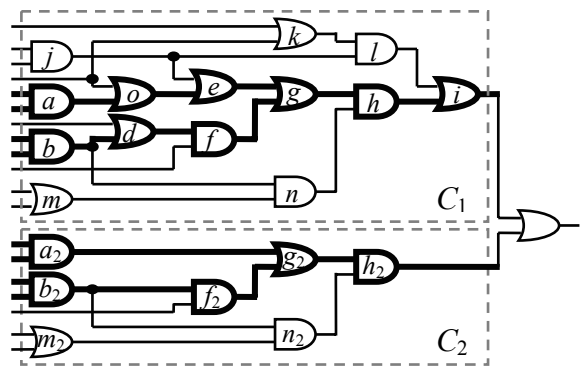


Figure 5: The re-synthesized circuit after wire removal.

equivalent Boolean function but do not belong to d_t -critical paths can be shared. For example in Figure 5, the output of node n in C_1 and that of node n_2 in C_2 have the same functionality. We can share the output signals of n and n_2 , as demonstrated in Figure 6. Suppose all equivalent signals are allowed to be shared without violating the requirement of d_t delay tolerance. The final circuit is shown in Figure 6.

A path is said to be an *imperceptible* path if any change (increase or decrease) on the path's delay can never affect the circuit delay [7].

Theorem 3: After wire removal according to Theorem 1 and Theorem 2, all d_t -critical paths are either false paths or imperceptible paths [7].

Proof: Omitted.

The intuition for this theorem is as follows. Our wire removal theorems guarantee that after wire removal, circuits C_1 and C_2 still have the same output value whenever d_t -critical paths in C_1 or C_2 are activated. Therefore, any delay increment on a d_t -critical path will not affect a circuit's delay. In Figure 6, all highlighted paths are either false paths or imperceptible paths. We now discuss slacks of nodes after re-synthesis. Consider node e in Figure 6. The longest true path passing through node e is $\{j-e-g-h-i\}$ whose path length is 5. Assume the timing requirement is 6. Therefore, node e has the slack of 1 in the re-synthesized circuit in Figure 6 while it has the slack of 0 in the original circuit in Figure 3. In another example, since all paths passing through node a are false paths, the slack of node a is infinite. Generally, after re-synthesis, paths whose delays are greater than 5 become either false paths or imperceptible paths; that is, the longest true path in the re-synthesized circuit has the delay of 5. As a result, the slack of each node in the re-synthesized circuit is at least 1 ($=d_t$).

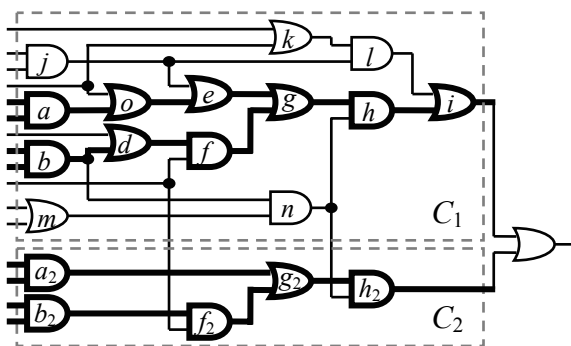


Figure 6: The re-synthesized circuit after signal sharing.

4. Delay Tolerance on Internal Signals

The re-synthesis methodology described previously is applied to primary outputs whose arrival time is susceptible to delay variation. We can also employ an identical approach to protect the arrival time of internal signals from delay variation. Delay tolerance on internal signals can have the advantage of area reduction.

In Figure 7, consider the circuit which is almost the same as that in Figure 3 except node g in Figure 7 is an AND gate. Bold lines in Figure 7 represent d_t -critical nodes and d_t -critical paths. If the re-synthesis technique is applied to only the primary output, only d_t -side input wire w_1 in C_2 can be removed and the resultant circuit is shown in Figure 8. In this example, the area overhead is large because there is only one removable d_t -side input wire. We will show that by employing delay tolerance on internal signal w_f , the area penalty can be reduced. The result after performing re-synthesis on w_f is shown in the gray blocks in Figure 9. According to Theorem 3, path segment $\{s-b-d-f\}$ becomes either a false path (segment) or an imperceptible path (segment). Thus, path $\{s-b-d-f-x-g-h-i\}$ passing through a false path (segment) $\{s-b-d-f\}$ is also a false path. Similarly, all other paths passing through $\{t-b-d-f\}$, $\{s_2-b_2-f_2\}$, and $\{t_2-b_2-f_2\}$ are either false paths or imperceptible paths. Those

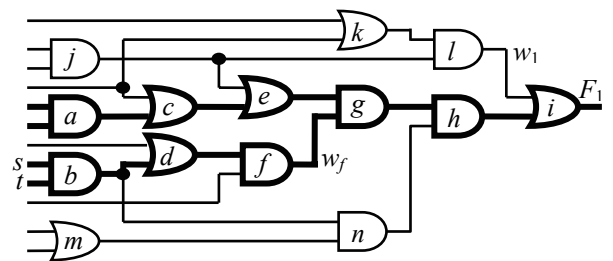


Figure 7: An example demonstrating delay tolerance on internal signals.

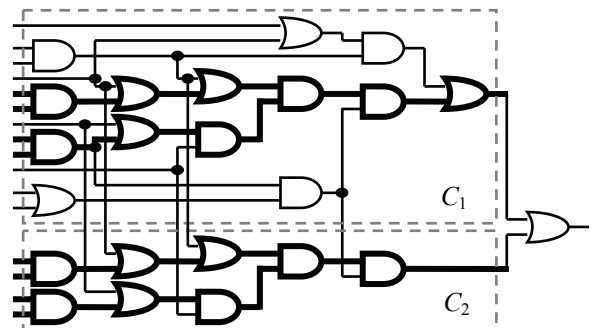


Figure 8: The re-synthesized circuit by applying delay tolerance on the primary output in Figure 7.

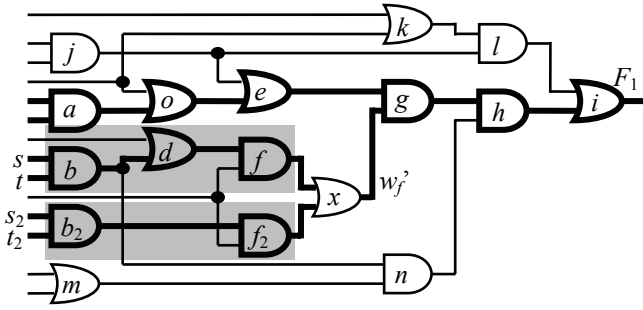


Figure 9: The re-synthesized circuit by applying delay tolerance on internal signal w_f in Figure 7.

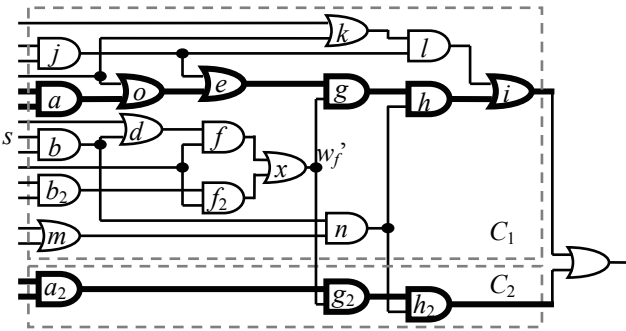


Figure 10: The re-synthesized circuit by applying delay tolerance on the primary output in Figure 9.

false paths or imperceptible paths no longer belong to d_r -critical paths. Consequently, nodes $\{a, o, e, g, h, i\}$ in Figure 9 become d_r -dominators. When we continue to perform delay tolerance on the primary output, d_r -side input wires to three OR d_r -dominators $\{o, e, i\}$ in C_2 become removable. The re-synthesized circuit is shown in Figure 10. The total area overhead of the re-synthesized circuit in Figure 10 is 7 (gates), whereas that of the re-synthesized circuit in Figure 8 without delay tolerance on internal signals is 9 (gates).

5. Experimental Results

We have implemented the re-synthesis method [3] and ours in SIS environment, and also experimented on a set of MCNC and ISCAS benchmarks. Table 1 provides the comparison of the area overhead between the method in [3] and ours. Table 2 demonstrates the advantages of delay variation tolerance for domino circuits. We first optimized a circuit with “*script.delay*,” and then used delay tolerance values of 10% and 15% of the original circuit delay to re-synthesize a circuit. The timing requirement for each

re-synthesized circuit is set to be the delay of the corresponding original circuit.

In Table 1, column one lists the name of the original circuit. Columns two and three provide the area and delay of the original circuit, respectively. Columns four and five show the results for 10% delay tolerance and columns six to seven for 15% delay tolerance. For a given delay tolerance requirement, we re-synthesized the original circuit by the method in [3] and ours individually. Then we compared the area overhead between the re-synthesized circuits. For example, circuit *C880* has the area of 3569 and the delay of 33.6. When the delay tolerance is 3.36 ($= 10\% \cdot 33.6$), the re-synthesized circuit by [3] has the area overhead of 31.2% of the original circuit’s area. On the other hand, the re-synthesized circuit by our method has the area overhead of 14.9% of the original circuit’s area. When the delay tolerance is 5.04 ($= 15\% \cdot 33.6$), the re-synthesized circuit by [3] has the area overhead of 53.1% while by our method has the area overhead of 25.4%. On the average, for 10% (15%) delay tolerance, the area overhead from [3] and ours is 29% (48%) and 12% (21%), respectively. Both the methods usually spend few seconds for each circuit.

We have also performed Monte-Carlo experiments to demonstrate the effect of delay variation tolerance in Table 2. In the experiment, the delay of a gate is given as a probability density function similar to [6]. We obtained 1000 samples of an original circuit and 1000 samples of its re-synthesized circuit with 10% delay tolerance from Monte-Carlo experiments. After that, the delay of each sample is calculated. In Table 2, column one lists the name of the original circuit. Let d_r be the delay of the circuit. Column two shows the number of circuit samples whose delays are smaller than $\{0.8 \cdot d_r\}$ for the original circuit, and column three shows the number for its re-synthesized circuit. Similarly, columns four to eleven report the numbers of samples whose delays are smaller than $\{0.9 \cdot d_r\}$, $\{1.0 \cdot d_r\}$, $\{1.1 \cdot d_r\}$, and $\{1.2 \cdot d_r\}$ for both the original circuit and its re-synthesized circuit. Take circuit *t481* as an example. Let the timing requirement be $1.1 \cdot d_r$ ($1.1 \cdot 34.1 = 37.51$), 815 samples of the original circuit meet the timing requirement while 963 samples of the re-synthesized circuit satisfy the same requirement.

In Figure 11, we drew the distribution curves of all Monte-Carlo samples of circuit *t481* and its re-synthesized circuit with 10% delay tolerance. The standard deviation of

Table 1: Comparison between [3] and our method

Circuit	Original circuit		10% delay tolerance		15% delay tolerance	
	Area	Delay	[3]	Ours	[3]	Ours
			Area overhead (%)	Area overhead (%)	Area overhead (%)	Area overhead (%)
C432	1568	35.9	6.0	2.5	6.0	2.5
C880	3569	33.6	31.2	14.9	53.1	25.4
C1355	5854	35.8	29.3	10.4	47.1	20.4
C1908	6080	48.2	35.0	16.7	54.5	26.3
C2670	11281	62.2	23.2	11.4	34.2	16.8
C3540	19170	85.4	30.2	15.0	45.0	22.4
C5315	29937	90.0	38.1	18.6	59.5	29.4
C6288	46379	161.7	39.2	19.3	59.2	29.2
pair	12941	28.6	33.1	14.0	51.3	22.3
rot	5915	35.7	30.8	11.8	50.9	19.2
t481	4927	34.1	6.9	1.1	11.4	2.4
9symml	1317	21.4	45.3	10.9	74.1	21.9
alu2	3855	50.7	39.8	19.0	57.0	27.2
apex6	5220	56.2	18.5	7.2	63.8	24.4
apex7	1836	26.9	32.9	12.5	48.0	17.4
Avg.			29.3	12.3	47.7	20.5

the original circuit is 2.39 while that of the re-synthesized circuit is 1.92. The curves also reveal that the re-synthesized circuit has more stable timing behavior than the original circuit.

6. Conclusions

We have proposed a framework to re-synthesize a given domino circuit for d_r delay tolerance. Our method begins with a duplex system; we then adopt wire removal and signal sharing to reduce the area overhead of the delay tolerance structure. Two novel theorems for further area reduction are presented. Experimental results demonstrate the advantages of delay variation tolerance for a re-synthesized domino circuit.

References

- [1] K. Baker, G. Gronthoud, M. Lousberg, I. Schanstra, and C. Hawkins, "Defect-based delay testing of resistive vias-contacts, a critical evaluation," *Proc. of International Test Conference*, pp. 467-476, Sept. 1999.
- [2] M. A. Breuer, C. Gleason, and S. Gupta, "New validation and test problems for high performance deep sub-micron VLSI circuits," *Tutorial Notes, VLSI Test Symposium*, April 1997.
- [3] Shih-Chieh Chang, Cheng-Tao Hsieh, and Kai-Chiang Wu, "Re-synthesis for delay variation tolerance," *Proc. of Design Automation Conference*, pp. 814-819, June 2004.
- [4] Seung Hoon Choi, Bipul C. Paul, and Kaushik Roy, "Novel sizing algorithm for yield improvement under

Table 2: Statistical results of Monte-Carlo experiments

Circuit	$0.8*d_r$		$0.9*d_r$		$1.0*d_r$		$1.1*d_r$		$1.2*d_r$	
	O	R	O	R	O	R	O	R	O	R
C432	0	0	1	0	125	280	663	852	977	1000
C880	0	0	2	117	265	819	907	996	999	1000
C1355	0	0	0	0	0	7	68	445	583	862
C1908	0	0	0	0	1	0	248	389	964	976
C2670	0	0	8	14	150	186	515	680	893	965
C3540	0	0	5	2	66	50	361	386	855	910
C5315	0	0	7	1	96	62	346	570	796	992
C6288	0	0	0	0	35	553	769	999	1000	1000
pair	0	0	0	0	1	13	162	386	753	912
rot	0	0	0	17	75	299	486	706	877	976
t481	1	0	29	19	373	463	815	963	984	1000
9symml	0	0	0	0	3	3	138	249	639	926
alu2	0	0	0	5	75	259	551	825	940	1000
apex6	248	398	368	504	470	601	564	718	693	836
apex7	4	1	71	47	264	298	622	711	780	890
Avg.	17	27	33	48	133	260	481	658	849	950

O: Original circuit

R: Re-synthesized circuit

- process variation in nanometer technology," *Proc. of Design Automation Conf.*, pp. 454-459, June 7-11, 2004.
- [5] E.T.A.F. Jacobs and M.R.C.M. Berkelaar, "Gate sizing using a statistical delay model," *Proc. of DATE*, pp. 27-30, 2000.
 - [6] Jing-Jia Liou, A. Krstic, Li-C. Wang, and Kwang-Ting Cheng, "False-path-aware statistical timing analysis and efficient path selection for delay testing and timing validation," *Proc. of Design Automation Conference*, pp. 566-569, June 2002.
 - [7] Sreeja Raj, Sarma B. K. Vrudhula, Janet Wang, "A methodology to improve timing yield in the presence of process variations," *Proc. of Design Automation Conf.*, pp. 448-453, June 7-11, 2004.
 - [8] Alexander Saldanha, "Functional timing optimization," *Proc. of International Conference on Computer-Aided Design*, pp. 539-543, Nov. 1999.

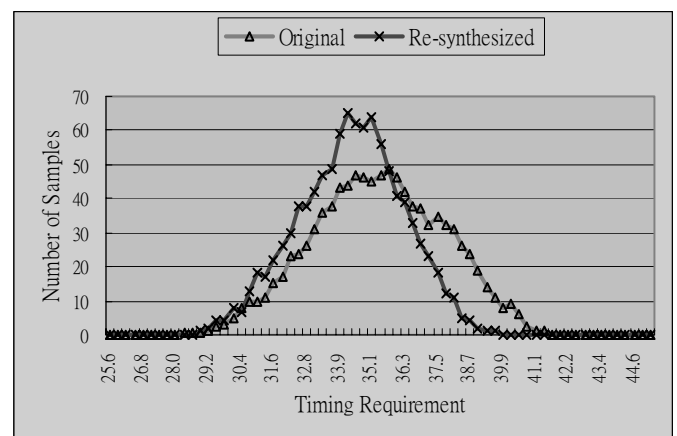


Figure 11: The distribution curves of Monte-Carlo samples of circuit *t481* and its re-synthesized circuit.