Enhancing Test Efficiency for Delay Fault Testing Using Multiple-Clocked Schemes

Jing-Jia Liou, Li-C. Wang and Kwang-Ting Cheng
Department of ECE, UC-Santa Barbara
jjliou@windcave.ece.ucsb.edu, {licwang, timcheng}@ece.ucsb.edu

Jennifer Dworak and M. Ray Mercer
Department of EE, Texas A&M University
jdworak@dropzone.tamu.edu, mercer@ee.tamu.edu

Rohit Kapur and Thomas W. Williams
Synopsys Inc., {rkapur, tww}@synopsys.com

ABSTRACT

In conventional delay testing, the test clock is a single pre-defined parameter that is often set to be the same as the system clock. This paper discusses the potential of enhancing test efficiency by using multiple clock frequencies. The intuition behind our work is that for a given set of AC delay patterns, a carefully-selected, tighter clock would result in higher effectiveness to screen out the potential defective chips. Then, by using a smarter test clock scheme and combining with a second set of AC delay patterns, the overall quality of AC delay test can be enhanced while the cost of including the second pattern set can be minimized. We demonstrate these concepts through analysis and experiments using a statistical timing analysis framework with defect-injected simulation.

Categories and Subject Descriptors

B.8.1 [**Hardware**]: Reliability, Testing, and Fault-Tolerance

General Terms

Experimentation, Measurement, Reliability

Keywords

Delay Testing, Statistical Timing Analysis, Transition Fault Model

1. INTRODUCTION

Suppose a circuit consists of n sites, numbered as $1 \ldots n$. Without loss of generality, assume that by applying a set A of 100% transition fault patterns, $l_1 \ldots l_n$ are the longest sensitized paths passing through site $1 \ldots n$, respectively. Let C be the test clock period and let M be a timing analysis model where $M(l_i)$ represents the delay time returned by the model for path l_i. Then, after the application of A, for each site i, any delay defect with a size greater than $C - M(l_i)$ can be captured. For the whole circuit, any single-site delay defect with a size greater than $C - \min\{M(l_1) \ldots M(l_n)\}$ can be captured. For enhancing the quality of transition fault testing, it is often suggested that we should maximize each $M(l_i)$.

In traditional timing analysis, model M is often based upon nominal or worst-case timing assumptions. In the deep sub-micron domain, factors that affect timing characteristics such as process variations, manufacturing defects and noise are often statistical in nature [1, 2]. These factors should be better captured and simulated using statistical models and methods [3, 4]. Hence, the outputs of $M(l_i)$ and $M(p_i)$ in the above analysis can be probability distributions rather than discrete delay values.

With M being a statistical timing analysis model, our focus does not lie in maximizing each $M(l_i)$ for transition fault patterns by selecting the longest l_i for explicit testing. Instead, we focus on achieving the following objective: We assume that two sets of transition fault patterns A and B are given (similarly, assume that A covers paths $T_1 = \{l_1^1 \ldots l_n^1\}$ and B covers paths $T_2 = \{l_1^2 \ldots l_n^2\}$). Test quality is enhanced by combining the two pattern sets. This is because for each site i, any delay defect with a size greater than $C - \max\{M(l_i^1), M(l_i^2)\}$ can be captured. Then, our goal is to find a clock scheme such that the test application cost is minimized relatively to the cost of applying only one set, and the quality is maximized relatively to the quality of applying both sets.

In our method, the application of tests is divided into two phases: *screening phase* and *confirmation phase*. In the screening phase, the goal is to use the first set of patterns with a tighter clock scheme to screen out *potential* defective chips. Then in the confirmation phase, both sets of patterns are applied with the normal clock to confirm which ones are actually bad. The key of this strategy is the selection of clock scheme in the screening phase in order to minimize the cost during the confirmation phase. The cost saving relies on the fact that majority of the chips are good and hence, only a few would be processed during the confirmation phase.

Several publications [5, 6, 7, 8] have tried to use variable or multiple clock schemes to improve the detection of delay defects. Unfortunately, there is one serious drawback of these schemes: the false negative outcomes. It is possible that most "bad chips" classified by these schemes are functionally and timingly correct, because most delay defects that increase the delays of some paths do not actually locate at timing critical paths. This problem will result in unwanted increase in the defective part level and decrease of the total yield (yield loss).

2. CLOCK SCHEMES

It is important to note that given the two sets A and B, alternating the clock itself will not improve the quality beyond that obtained using both A and B with the normal clock. Hence, $A + B$ combined

with the normal clock will represent an upper bound (in quality) of using any multiple-clocked scheme. This is because to avoid yield loss, it is necessary to apply again both A and B at the end with the normal clock to filter out good chips that were considered to be defective under a tighter clock. Hence, the total number of defective chips captured cannot be greater than that captured by all tests in $A + B$ combined with the normal clock.

2.1 Initial Clock Scheme

The above discussion implies a very simple 2-clock scheme. During the screening phase, a tighter clock $C - \delta$ is used with the test set A. During the confirmation phase, the normal clock is used with the combined test set $A + B$. It is important to note that path delay tests, if available, can be used for set B. However, set A has to target on transition faults. This is because a tighter clock in the screening phase does not help to capture a defect falling beyond the topological coverage of the paths sensitized by A. Hence, to ensure high quality, it is necessary to use a screening pattern set that can provide a complete topological (site) coverage.

Defect Miss A chip passing the screening phase may be defective and still can be captured if B is applied with the normal clock. However, it is guaranteed that the pass chip will also pass if A is applied with the normal clock. Based upon these points, it is possible that some defects are missed under the new test application scheme, but they can be captured by the original test method where $A + B$ is applied with the normal clock.

How likely will this happen? Consider a given site i. A true defect (which can affect circuit performance) on i will be missed if its size d falls into the range $[C - M(l_i^2), C - \delta - M(l_i^1)]$. This implies $M(l_i^1) + \delta < C - d < M(l_i^2)$. With a reasonably large δ, it is unlikely that $M(l_i^2) > M(l_i^1) + \delta$. Even with this happening, the chance of a defect occurrence with a size that falls exactly into the range is small. Hence, the problem of defect miss is not a serious concern.

Potential Yield Loss (Test Cost) Yield loss is defined as the percentage of chips which are considered to be defective by the tests but are actually good chips. To avoid yield loss, in the confirmation phase $A + B$ combined with the normal clock has to be applied again to test all potentially defective chips identified in the screening phase.

Again, how likely will potential yield loss happen? Since we have already use $A + B$ and the normal clock to prevent this from happening, the key here is that if a good chip is passed to the confirmation phase, it will increase our testing cost and hence, we want to minimize that as much as possible.

Similarly to the defect miss situation, a defect on site i will result in potential yield loss if its size d falls into the range $[C - \delta - M(l_i^1), C - \max(M(l_i^1), M(l_i^2))]$. This implies that $\max(M(l_i^1), M(l_i^2)) < C - d < \delta + M(l_i^1)$. Hence, to minimize potential yield loss after the screening phase (the test cost), δ should be as small as possible.

The Selection of δ

The above analysis suggests two contradictory objectives to select δ. On one hand, δ should be large to avoid defect miss. On the other hand, δ should be small to minimize the cost resulting from the effort to avoid yield loss during the confirmation phase. By combining the two objectives, it seems that the most logical solution is to set $\delta_i = D_i = | M(l_i^1) - M(l_i^2) |$ for each site i.

In practice, we can simply assume δ to be a small constant to begin with. Then, through statistical analysis and simulation, adjust δ to an acceptable level such that the cost to avoid yield loss during the confirmation phase is within the tolerance. From this perspective, δ is set based upon the cost we can afford. The larger the δ is, the better the quality will be and the higher the cost we will pay.

2.2 Revised Clock Scheme

If only one clock is used during the screening phase, the above iterative method provides a feasible way to approach the desired δ. Another alternative is to employ multiple clocks during the screening phase as well. Suppose we are allowed to use k clocks in the screening phase, what will be a good strategy to apply these k clocks? To answer this question, we first need to know how to partition the first test set. Then, we need to decide what clock to be used for each subset of patterns. Our overall strategy is illustrated in the following.

Test Set Partitioning Given a set of patterns, we first call the statistical timing analysis tool M [3] to characterize the delay distribution of the longest sensitized path passing through each site (again denote these results as $M(l_1^1) \ldots M(l_n^1)$). Next we compute the mean value in each distribution (assume that the order is $m(l_1^1) < \ldots < m(l_n^1)$). Then, the partition is based upon the difference $D = m(l_n^1) - m(l_1^1)$. The first partition is to divide $[0, D]$ into two ranges $D_1 = [0, \varepsilon D]$ and $D_2 = [\varepsilon D, D]$ where ε denotes the *ratio of partition* and $0 < \varepsilon < 1$ (normally, $0.5 < \varepsilon$ because short paths are less important than the long paths). For all paths whose m values falling into the first range will be grouped into the first set S_1. The rest will be grouped into the second set S_2. Then, the same idea can be applied recursively to partition S_2 further, and so on. In section 3.2, we will discuss the selection of ε so that the best performance can be achieved.

Clock Selection Given a subset S_i of paths, we extract the patterns to cover S_i. Then, for those patterns, we determine the path set S_i' being covered. Based upon the path set S_i', statistical timing analysis is performed to characterize the delay distribution of the longest path. This distribution is often normal and represented by its mean and standard deviation as (μ, σ). Then, the test clock for this subset of patterns will be set as $\mu + c\sigma$ where c is a small constant. Usually, it is reasonable to have $c = 3$.

2.3 Cost Evaluation

To enable us to compare different clock schemes, a cost evaluation method is required. Given the two sets of patterns A and B, intuitively we may define the cost as $A + \rho(A + B)$ where ρ represents the probability of a good chip being classified as being potentially defective during the screening phase.

As discussed before, this probability ρ depends on the clock(s) selected during the screening phase. However, the true cost is not entirely reflected in the simple equation $A + \rho(A + B)$. This is because we should also take defect size distribution into account. Without loss of generality, we assume defect distribution is a discrete function Γ. Then, for each defect size s, $\Gamma(s)$ gives the conditional probability that if a defect does occur, the chance of being with the size s. Then the true cost should be re-formulated as:

$$Cost = A + [\sum_{\forall s} \Gamma(s) \times \rho_s \times (A + B)] \qquad (1)$$

where s is the defect size and ρ_s is the probability of a good chip being classified as a defective chip during the screening phase due to a defect with size s. In essence, equation (1) calculates the cost by averaging across the defect distribution curve. If the defect

distribution is uniform, then the equation will be reduced to $A + \rho(A+B)$ again where $\rho = \sum_{\forall s} \rho_s$.

2.4 Cost Reduction

The cost can be further reduced if we allow an extremely small yield loss tolerance level. For example, suppose for large s where $s > t$, $\Gamma(s) < \varepsilon$ where ε is a very small number such as 10^{-4}. Then, all short paths of which timing lengths less than $C - t$ (with high probabilities) in A can be removed during the confirmation phase. This is because these paths will provide minimal help in deciding that a potential defective chip is actually a good chip. Accordingly, we can remove all patterns that cover only those removed short paths. In the next section, we will demonstrate that if Γ is an exponential distribution, a large numbers of paths in A can be removed in the confirmation phase with very little increase (almost zero) of the yield loss probability. As a result, the cost of applying our proposed test scheme can be dramatically reduced.

3. EXPERIMENTAL RESULTS

3.1 Evaluation Framework

In order to evaluate the cost and effectiveness of each clock scheme with different pattern sets, we implemented an evaluation framework to estimate the test coverage based on defect injection in the statistical timing analysis model.

Figure 1 illustrates the complete procedure of our evaluation scheme for a particular clock and pattern set. Note that the pattern set is characterized using the set of paths S sensitized by the patterns. In each Monte Carlo sampling run, first a circuit instance with cell/interconnect delays is generated according to the delay distributions characterized through Monte Carlo SPICE. This instance will then be evaluated by "statistical analysis of S". The "statistical analysis of S" is to check if there is any path in S (on the given instance) longer than the testing clock C. If there is, then this instance is said to be faulty and covered by S (*Covered*). At the end, our scheme will calculate the probability of a faulty path captured by S based upon all the instances statistically produced. This *capture probability* is defined as $Covered/(\# \ of \ instances \ generated)$.

Note that at the beginning of each Monte Carlo run, we will inject a delay defect randomly at a chosen signal node. This is to simulate the randomness of spot defects that cause chips to fail. Also note that we use a statistical cell/interconnect delay model to simulate the variation of each path delay. For multiple-clocked scheme, the evaluation process is similar to the above, but for those circuit instances classified as *Covered* during the screening phase, they will further be tested in the confirmation phase. Then, at the output of the process we will obtain the number of instances which have to be further tested, and the final number of *Covered* after the confirmation phase.

In our experiments, we utilize a Monte Carlo SPICE (ELDO) [9] to extract the statistical delays of cells for $0.25 \mu m$, 2.5V CMOS technology. The input transition time and output loading of the cells are used as indices for building/accessing these libraries.

3.2 Results

In the following, we will first explain in detail our results obtained for circuit s5378. In Figure 2, we show the effectiveness of combining two transition pattern sets (Note that since our objective here is to compare the relative strength among different methods, the absolute values of these probabilities are not important and hence, to avoid confusion they are not marked). The quality level of the combined set is improved (relatively to the individual set alone) as expected. However, test cost will be dramatically increased by applying the two test sets together.

Figure 3 shows how the test quality is improved by using multiple-clocked schemes. When using the initial clock scheme, δ is set to
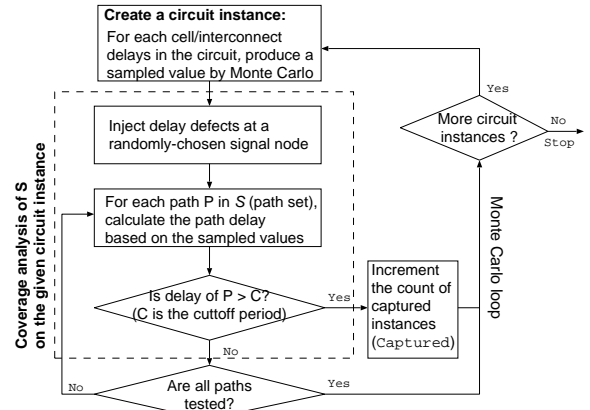

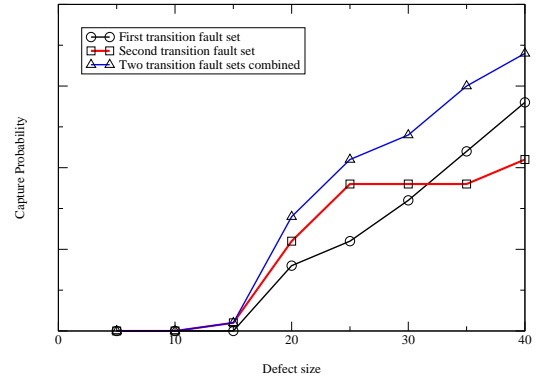**Figure 1: Flow Chart for Statistical Evaluation of** S

Figure 2: Defect Capturing Probabilities for Double Transition Pattern Sets

10 units. When using the revised clock scheme, the partition number is set to 3 (three different clocks are applied to three subsets during the screening phase). And the ratio of partition ε (as defined in section 2.2 above) is set to $1 - \frac{1}{e}$. Note that this particular ratio is obtained through experiments. We discovered that a smaller number such as 0.5 or a larger number such as 0.75 both would deliver inferior results.

In both figures, the first transition pattern set is used in the screening phase. As demonstrated in the figures, both the initial clock scheme and the revised clock scheme discussed in section 2 result in test quality levels that are bounded by the first transition pattern set and the *combined* transition set. However, the revised clock scheme provides a much better quality that is close to the combined set.
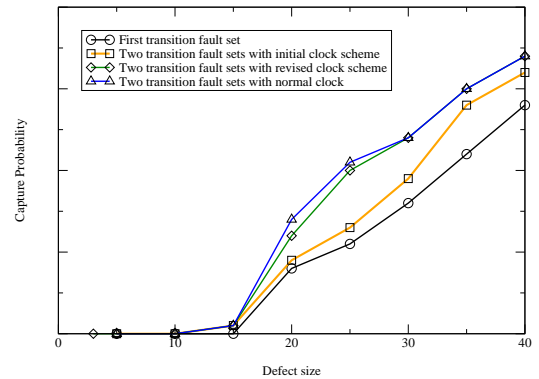

Figure 3: Defect Capturing Probabilities for Transition Pattern Sets Using Different Clock Schemes

Table 1 demonstrates the capture probabilities and costs for different testing schemes. The costs are calculated by the formula in section 2.3 and based on the assumption of a defect size distribution: $\lambda e^{-\lambda s}$ where s is the defect size and λ is a constant (we use 0.1 in the experiments). This exponential distribution for defect size (given that defects occur) has been studied in many publications [10, 11] and is a practical assumption to be used. Note that it is also possible to adopt other distributions. However, using other distributions in general does not invalidate the trends observed in our work.

To simplify the presentation, we will use the following notations in the table to represent different test schemes discussed above.

S1	first transition pattern set
S2	second transition pattern set
S3	first and second transition pattern sets with revised clock scheme
S4	first and second transition pattern sets combined with the normal clock

	S1	S2	S3	S4
capture prob.	0.60	0.77	**0.95**	1.0
cost (%)	1.000	1.006	1.274	2.006
reduced cost $t = 150$	1.000	1.006	1.113	2.006
reduced cost $t = 100$	1.000	1.006	1.027	2.006
reduced cost $t = 50$	1.000	1.006	**1.002**	2.006

Table 1: Test cost and capture probability comparison for different delay fault sets and clock schemes for s5378.

In the table, we also consider the cost reduction approach discussed in section 2.4. The t value in each case is shown in the table. Note that by assuming defect distribution as $\lambda e^{-\lambda s}$ ($\lambda = 0.1$), the probability of yield loss is close to 0.000676 for the worst case shown in the table (where $t = 50$ units). We emphasize that the absolute numbers in the table are less important than their trends.

Data in this table confirm various observations discussed above. For example, the quality of S3 (0.95) is close to that of S4 (1.0) and is greatly improved from S1 (0.60). Now given costs listed in the next three rows in the table, we can compare quantitatively the costs of different testing schemes. Here the capture probabilities are normalized with respect to that of S4, and the costs of schemes are normalized with respect to the cost of S1. Notice that in the case of cost reduction in S3, by linearly decreasing the t value, the cost can drop in terms of an order of magnitude.

The revised clock scheme S3 requires less testing effort than S4. Also S3 has the flexibility to reduce the cost by allowing a yield loss tolerance level. For the tolerance level of $t = 50$, the cost of S3 is only 1.002 which is almost the same as the cost from applying the first transition pattern set alone! We also have similar observations from Tables 2 and 3.

	S1	S2	S3	S4
capture prob.	0.75	0.8	**1.0**	1.0
cost (%)	1.000	0.990	1.274	1.990
reduced cost $t = 150$	1.000	0.990	1.221	1.990
reduced cost $t = 100$	1.000	0.990	1.061	1.990
reduced cost $t = 50$	1.000	0.990	**1.002**	1.990

Table 2: Test cost and capture probability comparison for different delay fault sets and clock schemes for s9234.

	S1	S2	S3	S4
capture prob. (%)	0.71	0.76	**0.95**	1.0
cost	1.000	1.000	1.779	2.000
reduced cost $t = 150$	1.000	1.000	1.406	2.000
reduced cost $t = 100$	1.000	1.000	1.233	2.000
reduced cost $t = 50$	1.000	1.000	**1.034**	2.000

Table 3: Test cost and capture probability comparison for different delay fault sets and clock schemes for s35932.

4. CONCLUSIONS

In this paper we discuss using multiple-clocked schemes to improve delay fault testing. We observed that applying a tighter clock alone does not help to improve delay test quality. Hence, we proposed a testing approach that consists of two test application phases: the screening phase and the confirmation phase. During the screening phase, the goal is to quickly identify those chips that are potentially defective. This goal is accomplished by using tighter clock(s) with an initial set of patterns that ensure a complete topological coverage. In the confirmation phase, a different pattern set is added to the original set for quality enhancement, and by testing with the normal clock, potential defective chips will further be confirmed as true defective chips.

Our two-phased multiple-clocked test application approach can deliver test quality close to that provided by combining the second pattern set and the original pattern set, with the cost close to that from the application of the original set alone. For the screening phase, we discuss two different clock schemes and conclude that the revised scheme is better than the initial scheme in quality. We studied our proposed methods based upon defect injection and simulation using a statistical timing analysis framework.

5. REFERENCES

[1] K. Baker, G. Gronthoud, M. Lousberg, I. Schanstra, , and C. Hawkins. Defect-Based Delay Testing of Resistive Vias-Contacts, A Critical Evaluation. *Proceedings of IEEE International Test Conference*, pages 467–476, September 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, IEEE VLSI Test Symposium*, April 1997.

[3] J.-J. Liou, K.-T. Cheng, and D. Mukherjee. Path Selection for Delay Testing of Deep Sub-Micron Devices Using Statistical Performance Sensitivity Analysis. *Proceedings of IEEE VLSI Test Symposium*, pages 97–104, April 2000.

[4] J.-J. Liou, A. Krstić, K.-T. Cheng, D. Mukherjee, and S. Kundu. Performance Sensitivity Analysis Using Statistical Methods and Its Applications to Delay Testing. *Proceedings of Asian South Pacific Design Automation Coneference*, pages 587–592, January 2000.

[5] W. W. Mao and M. D. Ciletti. A Variable Observation Time Method for Testing Dlay Faults. *Proceedings of Design Automation Conference*, pages 728–731, June 1990.

[6] V. S. Iyengar and G. Vijayan. Optimized Test Application Timing for AC Test. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 11(11):1439–1449, November 1992.

[7] D. Dumas, P. Girard, C. Landrault, and S. Pravossoudovitch. Effectiveness of a Variable Sampling Time Strategy for Delay Fault Diagnosis. *Proceedings of European Design and Test Conference*, pages 518–523, March 1994.

[8] W. B. Jone and Y. P. Ho. Delay Fault Coverage Enhancement Using Variable Observation Times. *Journal of Electronic Testing: Theory and Applications*, 11(2):131–146, October 1997.

[9] Anacad. Eldo v4.4.x User's Manual. 1996.

[10] N. N. Tendolkar. Analysis of Timing Failures Due to Random AC defects in VLSI moduels. *Proceedings of Design Automation Conference*, pages 709–714, June 1985.

[11] J. P. de Gyvez. *Integrated Circuits Defect-Sensitivity: Theory and Computational Models*. Kluwer Academic Publishers, Boston, MA, 1993.