Methods of Critical Paths Delay Faults Testing in Digital Systems

Miroslav Siebert[*]
Institute of Computer Systems and Networks
Faculty of Informatics and Information Technologies
Slovak University of Technology in Bratislava
Ilkovičova 2, 842 16 Bratislava, Slovakia
miroslav.siebert@stuba.sk

Abstract

The dissertation thesis is aimed at automatic delay faults test generation methods for digital systems. Path delay faults are tested via selected critical paths in a tested digital circuit. The critical paths can be specified e.g. by static timing analysis (STA), statistical static timing analysis (SSTA) and others. Signal delay propagation is also affected by many factors such as power supply noise, multiple input switching, temperature and others during test. The impact of each factor to the path delay faults has been individually solved and published in some papers but their joint effects should be also investigated.

A new method for evaluation of critical paths and a general system for critical paths selection, named PaCGen, are presented in the dissertation thesis. The method is based on STA and influences of multiple factors used in the new technique for path criticality calculation. Some critical paths can be found as untestable based on structure of the circuit. The proposed system PaCGen includes also more effective DFT (design-for-testability) technique to change untestable critical paths to testable. Evaluation of the proposed method, system and used techniques has been done over selected benchmark circuits and compared with published results.

Categories and Subject Descriptors

B.6.2 [**Logic design**]: Reliability and Testing—*Test generation, Testability*; B.7.3 [**Integrated circuits**]: Reliability and Testing—*Test generation, Testability*

Keywords

digital circuits, test generation, delay faults, path delay faults, critical path, path criticality, design-for-testability

1. Introduction

Delay fault testing is more and more important due to huge number of gates and lines integrated on a chip. The significance of delay defects increases with the shrinking features sizes of today's designs. Therefore, delay testing is a major issue in the post-production test to filter out defective devices. Delay tests are widely used to check whether a manufactured chip is free of delay faults and meets its performance specification. Right timing is very important in each design process of complex digital circuits. A digital circuit has a delay fault if its output fails to reach the right values within the pre-defined timing specifications. Different basic delay fault models (gate delay, path delay, segment delay, line delay faults etc.) have been developed which require inclusion of both logical value transitions. A test for delay faults consists of vector-pairs: the first vector is needed for set up an initial logic value at a tested location and the second one is for launching the value transition [1]. Delay fault test can be robust or non-robust. Robust are supposed to detect delay faults they target regardless of the presence of other delay faults in the circuit under test. A non-robust test is guaranteed to detect the fault it targets only if no other delay faults affect the circuit. When robust test generation is not possible non-robust test is generated.

Besides of transition delay faults, testing path delay faults is very important in the digital circuit design implemented in deep submicron and nanometer technologies. The path delay faults testing is very important mostly in nanotechnologies where digital circuits are more complex and signal lines length is too long. The path delay fault model is a cumulative delay of timing faults from primary inputs (or flip-flops outputs) to primary outputs (or flip-flops inputs). The number of paths is in exponential complexity depending on the number of gates. Therefore the delays are tested over a selected subset of paths specified as critical paths. The critical paths are functional paths in a circuit structure with the calculated least slack. A slack is a time margin (in picoseconds) which is left on a tested path for the proper propagation of logical values transition. If the slack is equal to zero, then there is no timing margin left on that path (it is the most critical path). Different tools exist for automatic timing analysis with slack calculations for digital circuits which also give crit-

ical paths for them. Thus a set of critical paths can be created based on slack lengths and automatic test pattern generator (ATPG) can generate test vector pairs over the critical paths [3].

Once the paths are determined for testing by one of existing algorithms, they are fed into an ATPG tool for generating the test patterns for sensitizing the paths. A path is said to be sensitized when its off-path inputs carry non-controlling values. Path sensitization is a necessary condition for the detection of a path delay fault under any type of tests (robust, non-robust etc.). Some paths turn out to be unsensitizable and thus untestable for delay faults during the ATPG process resulted in test fault coverage loss. The main reason is that value transition propagation is stopped inside the circuit and it cannot be observable at the primary outputs. Many physical paths can be recognized as untestable thus coverage loss is a severe problem.

The paper is organized as follows. The next sections II and III describe existed DFT techniques used to delay fault improvement and related work for usability of some factors with influence to delay in digital circuits. Section IV introduces motivation and objectives of Thesis and section V describes a new DFT techniques. A new technique for path criticality calculation and the system PaCGen are presented in section VII supported by defined basics and notation introduced in section VI. Experimental results are in section VIII. The last section concludes the paper.

2. DFT Techniques for Increasing Fault Coverage

A circuit structure can be modified for increasing the number of testable critical paths. Several DFT (design-for-testability) techniques [13, 7, 17, 10] have been proposed for solving activation of untestable paths to testable paths in the tested circuit. The techniques presented in [7] and [17] insert test points to disconnect circuit lines for test application purposes and eliminate dependencies of logical values based on the circuit structure. The DFT technique published in [7] inserts test-points in order to ensure that the maximum number of paths in every set has to be two. The DFT technique described in [17] connects disconnected branches whose stems are inputs of the tested circuit to new added inputs of the circuit. Both published DFT techniques remove dependencies between signal lines driven by the different fanout branches of the same inputs. Thus dependencies between lines that are driven by different fanout branches of the same inputs are removed. Authors in [13] and [10] propose partitioning circuit paths into sub-paths by using test-points. Each subpath can be tested independently. Thus each sub-path is tested using a fast clock cycle that corresponds to the expected delay of the subpath. The subpath testing requires using several different fast clock cycles.

The untestable path delay fault can be caused by an off-path input that is driven from a fanout stem on the path through one or more gates. The technique presented in [11] uses disconnection of the fanout branch that is not on a critical path from its stem and driven from a new input in order to reduce dependencies between off-path inputs of a target path delay fault. The fanout branch that is disconnected from its fanout stem requires a multiplexer. The fanout branch during functional operation is driven

without changing when the control input of multiplexer is set to logical value 0. During test application it has to be driven from a new added circuit input and the control input of multiplexer is set to logical value 1.

All the published DFT techniques for transforming untestable paths to testable paths need new inputs or additional overhead for modification of circuits. This paper presents a new developed DFT technique for the mentioned paths transformation with lower overhead of inputs in comparison with published techniques till now. Signal line selection is more flexible, it removes the need usability of several different fast clock cycles and the circuit paths have not to be divided.

3. Path Criticality Related Work

Variety factors can cause signal delay besides of the supposed timing analysis. These factors cannot be considered during timing analysis because the tools lack information about test vectors, layout, test applications, off-path switching activity etc.

Factors influence to the signal delay in digital circuits can be up to 36 % [16]. Then it is important to integrate them into critical paths selection. The described factors below have been individually investigated and published in the context of path delay faults in digital circuits and critical paths selection till now.

Multiple input switching (MIS). When multiple inputs switch simultaneously or close to each other, the gate delay can increase or decrease significantly compared to that of single input switching. The problem arises when multiple inputs of a gate switch in temporal proximity, in which case the max delay is more than the corresponding pin-to-pin delays, or the min delay is lesser than the pin-to-pin delays. Considering MIS makes the timing analysis conservative, as the probability of having an MIS event on every multi-input gate along a critical path may be very small. MIS on off-path inputs increases the delay on gates and can affect path delay significantly. Some experimental results were presented for 40 nm (delay increasing about 36 %) and 65 nm technologies [16].

Test robustness (TR). Different kinds of path delay tests have been defined: hazard free robust, robust and non-robust. Non-robust test vector pairs could excite longer path delay than robust test [2]. The delay testing quality can be significantly increased by combining timing-aware ATPG (automatic test patter generation) and robust test vector pair generation [5].

Asymmetric transition delay (ATD). Authors in [15] show that delays of rising and falling edges in gates can differ considerably. Again some experimental results for 40 nm and 65 nm technologies were presented such phenomena.

Number of don't care values (NoX). It is the number of undefined logical value in test vectors, marked as X and $X \in (0,1)$. When test vectors have a large number of values X thus there is possible to apply test compression techniques [4]. The compressed test can decrease MIS influence in dependence on logical values 0 and 1 assigned to X. The paths activated by such test vectors should be less critical and test should be robust and can produce less power consumption.

Paths used in the function mode (FNC). The paths which are never sensitized during functional operation do not need to be optimized for timing and their delays may be higher than the clock period [9].

Power supply noise (PSN). This parameter is caused by considerable switching activity in the circuit. The current switching activity can increase path delay caused by power-supply voltage variations [12]. Power supply and switching activity depend on different factors from circuit structure.

Through silicon vias (TSV). TSV is identified in three and two and half dimensional circuits (3D, 2.5D). TSVs are created by etching holes in silicon and filling the void with metal. The process of electroplating the metal can result in partial or porous metal fill meaning that the TSV channel is not completely filled or partly broken thus, creating an open defect (it can be also done by aging). The open defects can be categorized into resistive open (weak open) and open (strong open) defects. [8].

Temperature (TMP). In [14] it can be seen that, path delay increases nearly linearly with temperature. Temperature increase by $20°C$ cause delay increase about 5 %.

Others as power consumption, aging factor, area overhead added in redesign of circuits by DFT methods etc.

The next two sections present a new developed technique for path criticality calculation and a new system, named PaCGen (**Pa**rameterized **C**ritical paths **Gen**erator), for critical paths selection in digital circuits using joint effects of multiple factors described in this section. Thus critical paths selection mirrors better the paths for delay faults testing.

4. Motivation and Objectives

The key idea of the dissertation thesis is development of a new technique for path criticality calculation using multiple factors together. Results are known for using only one factor till now but in practice more factors have influence concurrently to path delay faults. A new method and the software system PaCGen have been developed for critical paths selection in digital circuits considering impact of multiple weighted factors. The method and the PaCGen system are based on STA results and new defined expressions for path criticality calculation.

This critical path selection method is original and uses also a new DFT technique for circuit structure modification for decreasing area overhead and number of added inputs. The new DFT method is based on usability of basic gates instead of multiplexers published till now [11]. This DFT technique reduces two transistors or three logic gates used as test points involved on disconnected critical paths as previously published works. The technique also reduces using of new circuit input per every untestable path.

5. New DFT Technique

The new DFT technique has been developed for modification of untestable paths in the circuit and its main steps are:

1. Selection of a set of untestable path delay faults per-

forming by ATPG for the set of target path delay faults in a circuit. An untestable path can be found from defined set of all critical paths. Path in the circuit will be untestable if it is impossible to assign right values on off-path inputs by one vector from vector-pair test.

2. Find line l_i as off-path input of first untestable path from set of untestable paths which can't be set to required transition value, and specify gate U, which causing untestable.

3. Disconnect line l_i from gate U and connect it as input of new added two-input gate G. A new logical two-input gate G is needed to insert before gate U. Output of gate G is input of gate U. This new gate G guarantees non-controlling value on off-path input of gate U. The non-controlling value marked as n where n is logical value 0 or 1 depending on type of gate U.

4. Selection of a suitable logical basic gate G for activating an untestable path to the testable path can be realized simply at design phase. The non-controlling value n for OR, NOR or XOR gates is logical 0 and for gates AND, NAND and XNOR will be logical 1. For gates XOR and XNOR, both logical values are non-controlling. With this restriction to a non-controlling value, new gate G can be formed by two types of gates. Gate U and type of new logical gate G is based on Table 1.

   Output of gate U depends only on the input from the critical path after insertion of the new gate G using previous rules.

5. Second input t of the new two-input gate G indicates path delay fault test mode. When path delay is tested thus $t = 1$ and n will have non-controlling value. In case of the normal functional mode $t = 0$ and gate G behaves as a conductor. Control input t is common to all new gates in the circuit structure and needs to be available in direct and negated forms. It is possible to use it only at direct form when gate G is AND with inverted input of this signal.

6. Run a test generator and check if selected path from step 2 is testable. If set of untestable paths from step 1 is not empty repeat steps 2 – 5 with other specified untestable path. If the set of untestable paths is empty, the DFT procedure is stopped.

New control signal t can be driven as combination of global test mode signal and new input value. A hazard-free robust test is suggested for gate U which causing untestability by this modification step. This type of test is most desirable. The hazard-free robust test on gate U may be combined with other type of test (robust test, non-robust test) on other gates.

Some limitations exist for this new DFT technique. New gate G can't be placed on an existing critical path because the new gate increases the delay of this path. Timing analysis has to be rerun after adding the new gates to the circuit. A non-critical path can become the critical by placing this new gate G on it. Therefore the new gate can be placed only on non-critical paths which no become critical after the small increasing delay.

Table 1: Type of New Gates U and G

original gate U	new gate U	new gate G
NAND		2-1 OAI
NOR		2-1 AOI
AND	NOR	NOR (negated from tested path)
OR	NAND	NAND (negated from tested path)
XOR	-	AND
XNOR	-	OR
AOI	-	OR if input joint to AND
		AND in others case
OAI	-	AND if input to joint OR
		OR in others case

6. Basic Definitions and Notation

Basic terms used in the proposed technique for path criticality calculation and critical path selection method are described in the section.

Path delay faults are tested through selected or defined critical paths. Test quality depends also on the right critical paths set selection. Obviously tester memories and test time for testing are limited. Therefore it is desirable to have an optimum number of paths in a tested circuit. Critical paths selection based only on slacks from STA is not sufficient for current and future technologies. Using influence of described or other factors has to be formalized for enriching critical path selection process. A list of the ordered critical paths is received from STA for a tested circuit. Testability of found critical paths can be verified by SAT solver with division into testable and untestable paths. The DFT technique described in section V can be applied to untestable critical paths, if necessary.

The list of all critical paths should be reordered according a critical variable. Therefore each critical path received from STA will be evaluated by the path criticality c_p. The expression for c_p calculation is involved in definition 1.

Let index i_{jp} is calculated for each parameter j on path p with ranking $i_{jp} \in <0,1>$ (higher number represents higher impact to path criticality). Each i_{jp} is evaluated by parameter weight w_j also from range $<0,1>$ where the higher number represents higher impact of the parameter j to path criticality c_p.

Definition 1. Path criticality in digital circuit represents quantified rate at which the path is critical.

Definition 2. Path criticality on path p is noted as c_p. It is assumed that $c_p \in <0,1>$.

Definition 3. Path criticality c_p based on multiple parameters is expressed by formula:

$$c_p = \left(1 - \frac{s_p}{t}\right) \cdot \prod_{j=1}^{k}[1 - w_j(1 - i_{jp})],$$

where s_p is the slack for path p from STA, k represents the number of accepted parameters on which criticality depends, t represents length of time interval, w_j is weight of parameter j and i_{jp} represents calculated index of parameter j impact on path p.

Definition 4. Critical path with criticality calculation is noted as qualified critical path.

Maximal optional impact *(I)* of all parameters to path criticality depends on the used technology, number of considerable parameters *(k)*, designer experiences etc.

Definition 5. If I is the maximal value of parameters impact thus the following formula is valid for their weights:

$$\sum_{j=1}^{k} w_j < I.$$

For example I can be set on 0.2 based on information and results from published works. Evaluation of paths in a tested digital circuit using STA results and formulas from definitions 3 and 5 has to be realized automatically therefore a new parameterized system for critical path selection has been developed and it is described in next section.

7. PaCGen System for Critical Path Selection

The proposed method of critical paths selection based on timing analysis, DFT techniques, path criticality calculation according definitions 2, 3, 5 has been implemented into a new software system PaCGen in C++ language. The PaCGen system architecture is shown in figure 1. It consists of seven main blocks: critical paths finding, testability verification, DFT method, path ordering, test generator, fault coverage and blocks to calculate indexes for parameters. Functionality of them is described below. The path ordering uses seven blocks for calculating indexes for individual parameters with impact to path delays. Each block runs over an internal structure of a tested circuit. Functionality of main blocks is follows.

Critical paths – it runs over the combinational part of a tested circuit structure, its STA data with timing information of logical gates and random timing data for wires. This block calculates slacks for every existing path in the circuit by modified Dijkstra's algorithm [6].

Testability verification – this block checks if critical paths are sensitive because the list of critical paths can contain also untestable paths. SAT solver is used for checking path sensitivity. The outputs of this block are lists of testable and untestable critical paths.

DFT method – this block ensures the new DFT method application over the untestable critical paths. Each untestable critical path is changed to testable by adding only one new logical gate into circuit structure. Delay fault coverage can be increased to 100 % by this method.
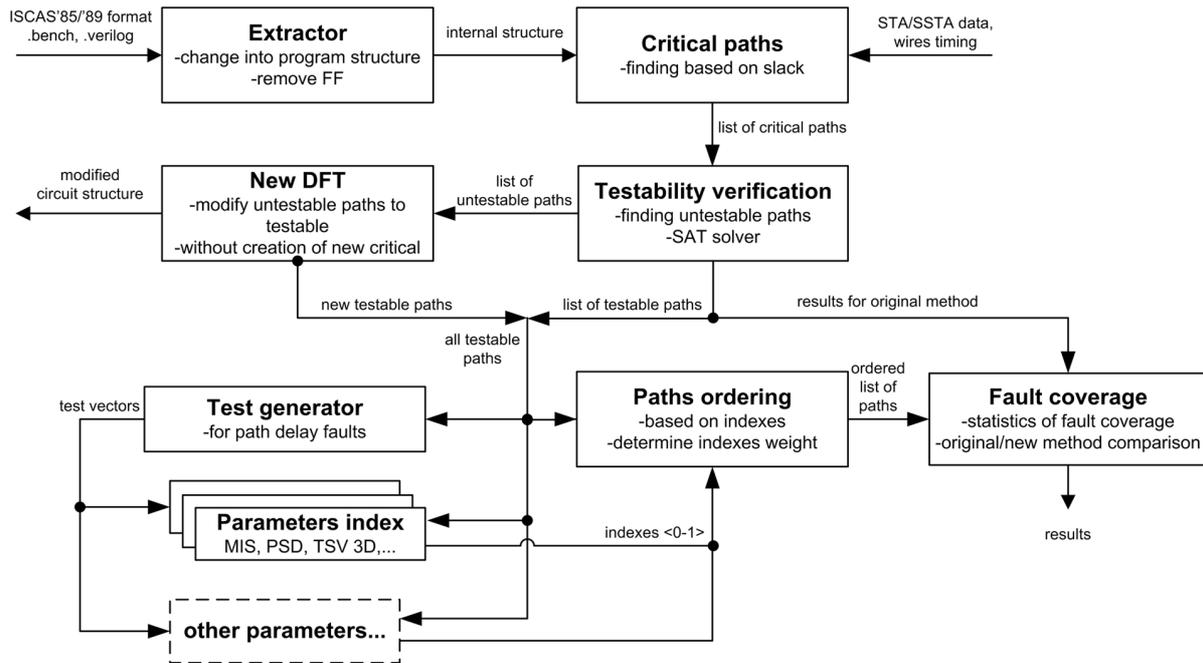
Figure 1: The PaCGen system architecture.

Paths ordering – in this block criticality c_p is calculated according definition 3 for each testable critical path. The list of testable critical paths ordered by criticality is output of this block.

Test generator – it generates a test set for critical paths. A hazard-free robust test set is generated, if exist. Otherwise a robust or non-robust test set is generated. For each critical path is possible to generate test sets ensured by the testability verification block.

Fault coverage – it calculates statistics and fault coverage for both types of test sets for their comparison based on fault simulation data. It compares fault coverage for unordered and ordered list of critical paths.

The other blocks are developed for indexes i_{jp} calculation. Next blocks are represented by Parameters index block in figure 1. The following parameters are involved in the PaCGen system for indexes calculation for a testable critical path.

Area index – this block calculates the area overhead if DFT technique is used on the critical path.

Index 3D, 2.5D – this index is based on number of TSV over the critical paths from netlist of a tested circuit. If the path contains TSVs then it is more critical.

MIS index – this block ensures calculation of multiple switching logical values at off-path inputs of logical gates. The logical values are assigned for propagating transition on the critical path. Maximum value of the index is 1 if each off-path input of logical gates changes its logical value.

PSD index – assigned logical values changing at gates inputs for propagating transition on the critical path. The logical values changing on-path inputs are also involved.

Type of edge index – the index depends on propagation of transition type and its value depends on used technology and type of logical gates on the critical path.

X–filling index – number of don't care logical values (X) in a test vector for the critical path.

Consumption index – the index is based on data about power consumption during testing the critical path.

The system PaCGen is flexible for adding or removing other parameters for path criticality calculation and for specification of weight values of parameters indexes based on used technology, circuit structure and its application.

8.   Implementation and Experimental Results

Experiments and results using the PaCGen system are presented in this section. The STA data was received by synthesis using Cadence Encounter RTL Compiler with 45nm NanGate FreePDK45 Generic Open Cell Library. Weights of parameters have been chosen on the basis of its published research. Specifically 3D - 2.4 %, edge type 2.8 %, MIS - 4.5 %, PSD - 4.8 %, X-filling - 3.8 %, consumption 0.7 % and area 1.0 %. Fault coverage is the same for original and reordered list of critical paths using the path delay fault model because PaCGen chooses the same number of paths for testing. Fault coverage is higher by using the transition fault model over the same number of tested original and reordered paths. Only 20 % of all critical paths have been selected because we simulate limited test time size.

Table 2 reports fault coverage for the transition fault model over selected ISCAS-89 benchmarks circuits. The second column shows initial fault coverage, the third column shows fault coverage after selection of the most critical paths ordered by the PaCGen (tested the same number of paths with column two). Column four shows improvement of fault coverage over transition delay fault model. Last column shows the number of untestable paths

Table 2: Delay Fault Coverage on ISCAS-89

Circuit	Fault coverage [%]			# paths changed
	Original	PaCGen	Impr.	
s386	87.32	87.54	0.22	3
s510	87.38	87.45	0.06	73
s641	52.22	55.27	0.06	73
s820	47.56	51.4	3.85	52
s832	46.29	51	4.7	71
s953	31.33	33.68	2.34	14
s1196	33.32	43.6	10.28	2160
s1238	17.8	33.53	15.72	1809
s1488	23	24.91	1.91	45
s5378	22.44	30.06	7.62	2068

changed to testable by application of the new DFT method.

Significant improvements of delay fault coverage exist in circuits greater and equal to s820 (approximately more than 280 logic gates). The method and the system for critical path selection have to be modified in implementation on parallel or cloud computing for achieving more real results on bigger circuits than circuits reported in Table 2.

9. Conclusions

The main contribution of the dissertation thesis is development of the new complex and adaptive method for critical path selection to delay faults testing and improving path delay fault coverage based on multiple parameters influence. The main idea is sorting a list of critical paths founded by STA using the new defined requirement – path criticality for increasing delay fault coverage. Higher fault coverage can be achieved using the transition delay faults with the same number of tested critical paths. Thus test set can be shorter for transition delay faults. Experimental results show increasing up to 16 % coverage in transition delay faults. This method reduces the test length and cost when test set is combination of path and transition delay fault models.

The partial contribution is in the new more effective DFT method which significantly increases delay fault coverage and test quality and decrease area overhead and number of added inputs.

Theoretical contributions of the dissertation thesis are in application of multiple factors with impact to delay faults based on qualified critical path definition, path criticality and its calculation. Practical contributions are in test length shortening, cost decreasing and test quality improving.

Future works can be in identifying more new factors which also have impact to signal delay (e.g. aging effects, temperature etc.) and impact of DFT applications to STA results in comparison with original circuits.

References

[1] M. Balaz, R. Dobai, and E. Gramatova. Delay faults testing. In *Design and test technology for dependable Systems-on-Chip*, pages 377–394. IGI Global, 2010.

[2] L. Chen, S. Gupta, and M. Breuer. High quality robust tests for path delay faults. In *Proc. of VLSI Test Symposium*, pages 88–93, 1997.

[3] J. Chung, J.Xiong, V. Zolotov, and J. A. Abraham. Testability-driven statistical path selection. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 31(8):1275–1287, August 2012.

[4] R. Dobai and M. Balaz. Compressed skewed-load delay test generation based on evolution and deterministic initialization of populations. *Computing and informatics*, 32(2):251–272, 2013.

[5] S. Eggersgluss, M. Yilmaz, and K. Chakrabaty. Robust timing-aware test generation using pseudo-boolean optimization. In *Proc. of Asian Test Symposium*, pages 290–295, 2012.

[6] H. N. Gabow and S. Nie. Finding long paths, cycles and circuits. In *Proc. of 19th International Symposium on Algorithms and Computation*, pages 752–763, 2008.

[7] A. Krstic, K. T. Cheng, and S. T. Chakradhar. Primitive delay faults: identification, testing and design for testability. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 18(6):669–684, Jun 1999.

[8] C. Metzler, A. Todri-Sanial, A. Bosio, and L. Dilillo. Computing detection probability of delay defects in signal line tsvs. In *Proc. of European Test Symposium*, pages 1–6, 2013.

[9] I. Pomeranz. On the detection of path delay faults by functional broadside tests. In *Proc. of European Test Symposium*, pages 1–6, 2012.

[10] I. Pomeranz and S. M. Reddy. Design-for-testability for path delay faults in large combinational circuits using test points. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 17(4):333–343, April 1998.

[11] I. Pomeranz and S. M. Reddy. Design-for-testability for improved path delay fault coverage of critical paths. In *Proc. of 21st International Conference on VLSI Design*, pages 175–180, 2008.

[12] S. K. Rao, R. Robucci, and C. Patel. Scalable dynamic technique for accurately predicting power-supply noise and path delay. In *Proc. of VLSI Test Symposium*, pages 1–6, 2013.

[13] S. Tragoudas and N. Denny. Path delay fault testing using test points. *ACM Transactions on Design Automation of Electronic Systems*, 8(1):1–10, January 2003.

[14] X. Wang, M. Tehranipoor, and R. Datta. A novel architecture for on-chip path delay measurement. In *Proc. of International Test Conference*, pages 1–10, 2009.

[15] S. H. Wu, S. Chakravarty, A. Tetelbaum, and L. C. Wang. Refining delay test methodology using konwledge of asymmetric transition delay. In *Proc. of Asian Test Symposium*, pages 137–142, 2008.

[16] S. H. Wu, S. Chakravarty, and L. Wang. Impact of multiple input switching on delay test under process variation. In *Proc. of VLSI Test Symposium*, pages 87–92, 2010.

[17] X. Xie and A. Albicki. Bit-splitting for testability enhancement in scan-based design. In *Proc. of International Conference on Computer Design*, pages 155–158, 1998.

Selected Papers by the Author

M. Siebert, E. Gramatová. Parameterized critical path selection for delay fault testing. *Proceedings of DDECS 2015 – International Symposium on Design and Diagnostics of Electronic Circuit & Systems* , pages 153–156, Belgrade, Serbia, 2015. CS IEEE Press.

M. Siebert, E. Gramatová, L. Nagy. PaCGEN: Automatic system for critical path selection based on multiple parameters. *Proceedings of BEC 2014 – 14th Biennial Baltic Electronics Conference* , pages 97–100, Tallinn, Estonia, 2014. CS IEEE Press.

M. Siebert, E. Gramatová. Delay Fault Coverage Increasing in Digital Circuits. *Proceedings of DSD 2013 – Euromicro Conference on Digital System Design* , pages 475–478, Santander, Spain, 2013. CS IEEE Press.