A Bridging Fault Model for Line Coverage in the Presence of Undetected Transition Faults

Irith Pomeranz
School of Electrical and Computer Engineering
Purdue University
West Lafayette, IN 47907, U.S.A.
E-mail: pomeranz@ecn.purdue.edu

***Abstract*—A variety of fault models have been defined to capture the behaviors of commonly occurring defects and ensure a high quality of testing. When several fault models are used for test generation, it is advantageous if the existence of an undetectable fault in one model does not imply that a fault in the same component but from a different model is also undetectable. This allows a test set to cover the circuit more thoroughly when additional fault models are used. This paper studies the possibility of defining such fault models by considering transition faults as the first fault model, and bridging faults as the second fault model. The bridging faults are defined to cover lines for which transition faults are not detected. A test compaction procedure is used for demonstrating the bridging fault coverage that can be achieved, and the effect on the number of tests.**

I. Introduction

A variety of fault models have been defined to capture the behaviors of commonly occurring defects and ensure a high quality of testing [1]-[24]. In addition to using different fault models, generating different tests for the same set of faults was shown to improve the quality of a test set [25]-[26].

When fault models with sets of faults M_0 and M_1 are used for a circuit, it is advantageous if the existence of an undetectable fault in M_0 does not imply the existence of undetectable faults in the same component of the circuit in M_1 (a component may be a line, gate or cell, depending on the fault models under consideration). For this purpose, it should be possible to satisfy the detection conditions for faults in M_1 even if it is not possible to satisfy the detection conditions for faults in M_0 that affect the same components. This contributes to the quality of testing by ensuring that tests can be generated to cover the circuit components better.

For illustration, let us consider a stuck-at fault g_i/a_i where line g_i is stuck-at the value a_i. Let us consider a transition fault $g_i : a_i \rightarrow a_i'$ where the $a_i \rightarrow a_i'$ transition on line g_i is delayed. The transition fault is detected by a two-cycle test that assigns $g_i = a_i$ in the first cycle, and $g_i = a_i'$ in the second cycle, creating the fault effect $g_i = a_i'/a_i$ in the second cycle. In addition, the test detects the single stuck-at fault g_i/a_i during the second cycle (the fault is not activated in the first cycle with $g_i = a_i$). If the single stuck-at fault g_i/a_i is undetectable, the transition fault $g_i : a_i \rightarrow a_i'$ is also undetectable. Thus, the use of transition faults does not provide a better coverage of the circuit for lines where single stuck-at faults are undetectable.

As another example, let us consider the four-way bridging fault $g_i/a_i/h_i$ where the value a_i on line h_i dominates the value of line g_i [10], [15]. A test that detects the fault assigns $g_i = a_i'$ and $h_i = a_i$ in the fault free circuit in order to create the fault effect $g_i = a_i'/a_i$. In addition, the test detects the single stuck-at fault g_i/a_i. Again, if g_i/a_i is undetectable, the four-way bridging fault $g_i/a_i/h_i$ is also undetectable.

Given a fault model with a set of faults M_0, the goal of this paper is to study the possibility of defining a fault model M_1 such that an undetectable fault in a circuit component under M_0 does not imply that the fault in the same component under M_1 is also undetectable. Because of the importance of transition faults in modeling delay defects, the study is performed by considering transition faults as the fault model M_0. In this case, the goal of M_1 is to provide coverage for lines with undetectable transition faults. The lines are logical lines (they are associated with a transition). Additional coverage is achieved by a carefully selected bridging fault model that also requires two-cycle tests.

A test compaction procedure for this approach was developed based on the procedure from [27]. The procedure accepts a test set T_0 for M_0. Considering the transition faults that are not detected by T_0, the procedure selects the set of bridging faults M_1. The procedure increases the fault coverage of M_1 while maintaining the fault coverage of M_0 and attempting to produce as few tests as possible. Accidental detection of additional faults from M_0 is also accommodated by the procedure, and causes it to update M_1. The procedure is applied iteratively to produce test sets T_1, T_2, ... that are more compact, or detect more faults from M_0 or M_1. The results of this procedure demonstrate that it is possible to detect bridging faults from M_1 when transition faults on the same lines are not detected, and thus increase the line coverage for a circuit using two-cycle tests.

The discussion in this paper does not attempt to claim that the bridging faults in M_1 are accurate in modeling the effects of bridge defects, or that they model delay defects. The study focuses on providing line coverage in cases where transition faults are not detected. The justification for this approach comes from the fact that fault models typically represent a simplified defect behavior in order to enable test generation. They are effective in spite of the simplifications because the tests that are produced based on them are effective in detecting

978-3-9815370-8-6/17/$31.00 ©2017 IEEE

defects.

The paper is organized as follows. Section II describes the bridging fault model used for defining M_1. Section III presents experimental results. The details of the test compaction procedure are omitted because of space constraints.

II. Bridging Faults

This section defines a set M_1 of bridging faults to complement the use of the set M_0 of transition faults, focusing on undetectable transition faults. Since the identification of undetectable faults is complex, bridging faults are defined based on transition faults that are not detected by a given test set, T_0. This is appropriate when the bridging faults in M_1 are used as targets for generating tests that will complement the test set T_0.

A. Definition

A logical bridging fault model that models the behavior of resistive bridges was defined in [21]. A bridging fault from [21] is represented as $g_i/a_i \rightarrow a'_i/h_i$. The fault represents the case where the $a_i \rightarrow a'_i$ transition on h_i dominates the transition on g_i. If g_i makes an $a'_i \rightarrow a_i$ transition in the fault free circuit, the fault delays it, resulting in an $a'_i \rightarrow a_i$ transition fault on g_i. The fault is defined in [21] as follows.

Definition 1: A bridging fault $g_i/a_i \rightarrow a'_i/h_i$ is detected by a test that satisfies the following conditions. (1) The test assigns the $a_i \rightarrow a'_i$ transition to h_i. (2) The test assigns the $a'_i \rightarrow a_i$ transition to g_i. (3) When the first two conditions are satisfied, and the fault is present, the transition on g_i is delayed. The test needs to detect the single stuck-at fault g_i/a'_i during the second cycle in order to detect the $a'_i \rightarrow a_i$ transition fault on g_i.

With these detection conditions, if the transition fault $g : a'_i \rightarrow a_i$ is undetectable, the bridging fault $g_i/a_i \rightarrow a'_i/h_i$ is also undetectable. The detection conditions of a bridging fault are modified as follows to ensure that this relationship between the models is eliminated. The definition is written such that the two fault models will have conflicting detection conditions for the same fault. Thus, detection by Definition 1 does not imply detection by Definition 2, and vice versa.

Definition 2: A bridging fault $g_i/a_i \rightarrow a'_i/h_i$ is detected by a test that satisfies the following conditions. (1) The test assigns the $a_i \rightarrow a'_i$ transition to h_i (the same condition exists in Definition 1). (2) The test assigns $g_i = a_i$ under both cycles of the test (Definition 1 requires a transition). In the presence of the fault, the $a_i \rightarrow a'_i$ transition on h_i dominates the value of g_i, and causes g_i to make an $a_i \rightarrow a'_i$ transition. (3) The first two conditions create a fault effect a_i/a'_i on g_i during the second cycle when the fault is present. The test needs to detect the single stuck-at fault g_i/a'_i during the second cycle (the same condition exists in Definition 1).

The two definitions differ in the second condition that requires a transition on g_i in Definition 1, and a constant value in Definition 2 (hazards are allowed on g_i). As a result of this difference, the bridging fault $g_i/a_i \rightarrow a'_i/h_i$ may be detectable under Definition 2 even if the transition fault $g_i : a'_i \rightarrow a_i$ is

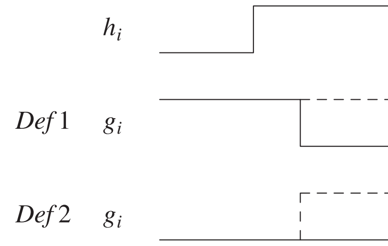

Fig. 1. Difference between Definitions 1 and 2

undetectable. In this case, the bridging fault $g_i/a_i \rightarrow a'_i/h_i$ is undetectable under Definition 1.

The difference between Definitions 1 and 2 is illustrated by Figure 1. Figure 1 considers the bridging fault $g_i/0 \rightarrow 1/h_i$. To activate the fault, a $0 \rightarrow 1$ transition must occur on h_i as shown in Figure 1. This holds under both definitions.

According to Definition 1, g_i needs to make a $1 \rightarrow 0$ transition that will be delayed by the transition on h_i. The delay in the faulty circuit is shown by dashes. Line g_i is labeled with $Def1$ in this case.

Definition 2 is important when the transition fault on g_i cannot be detected. In this case, Definition 2 allows a fault to be detected when g_i assumes a constant value in the fault free circuit. In the faulty circuit, g_i will make a transition that is shown by dashes. Line g_i is labeled with $Def2$ in this case.

While Definition 1 considers a bridging fault that delays a transition, Definition 2 considers a bridging fault that induces a transition. This leads to two different fault models. Only Definition 2 is considered in the next sections.

The bridging fault under Definition 2 is undetectable if the stuck-at fault g_i/a'_i is undetectable when it affects only the second cycle of the test. This is discussed next.

A true single stuck-at fault affects both cycles of the test. Definition 2 does not create fault effects in the first cycle. Therefore, the detectability of a true single stuck-at fault is not relevant to it. Single stuck-at faults that affect only the second cycle are not physical faults, but it is convenient to use them in the selection of sets of target bridging faults based on Definition 2. A single stuck-at fault that affects only the second cycle of a test is referred to as a second-cycle stuck-at fault. To check whether a second-cycle stuck-at fault is detected by a test, the circuit is considered to be fault free in the first cycle, and the fault is simulated only under the second cycle of the test.

B. Set of Target Faults

A set of bridging faults M_1 based on Definition 2 is selected with the goal of providing coverage for lines with transition faults that are not detected by a given test set, T_0. The transition faults may be undetectable or hard-to-detect, causing a test generation procedure to abort. Accordingly, the set M_1 is selected as follows.

Let M_0 be the set of transition faults. Fault simulation with fault dropping of M_0 under T_0 yields the subset of transition

faults that are not detected by T_0. This subset is denoted by U_0.

For a transition fault $g_i : a'_i \to a_i \in U_0$, the corresponding second-cycle stuck-at fault is g_i/a'_i. To be able to cover g_i with the $a'_i \to a_i$ transition using bridging faults, the second-cycle stuck-at fault g_i/a'_i needs to be detectable. The procedure for defining M_1 simulates the second-cycle stuck-at faults corresponding to the faults in U_0 under T_0. It includes in a set that is denoted by S_0 the undetected transition faults whose corresponding second-cycle stuck-at faults are detected. For every transition fault $g : a'_i \to a_i \in S_0$, the procedure adds to M_1 bridging faults of the form $g_i/a_i \to a'_i/h_i$, as follows.

For a parameter n_{BR}, if n_{BR} bridging faults are selected for every transition fault in S_0, the number of bridging faults in M_1 will be $n_{BR}|S_0|$. For the experiments in this paper, n_{BR} is selected such that $n_{BR}|S_0| \approx |M_0|$, or $n_{BR} = \lceil |M_0|/|S_0| \rceil$. The faults are selected randomly as follows.

The procedure finds the set H of lines such that there is no feedback from g_i to any line $h_i \in H$. If H contains more than n_{BR} lines, the procedure removes lines randomly from H until it contains n_{BR} lines. For every line $h_i \in H$, the procedure adds to M_1 the bridging fault $g_i/a_i \to a'_i/h_i$.

Starting from T_0, the test compaction procedure developed for this study produces a sequence of test sets T_1, T_2, ... with non-decreasing bridging fault coverages and non-increasing numbers of tests. Accidentally, the procedure may increase the transition fault coverage or detect a new second-cycle stuck-at fault. This leads to the following updates of M_1. (1) If a fault $g_i : a'_i \to a_i \in U_0$ is detected by a test set T_j, for $j \geq 1$, the procedure removes from M_1 all the bridging faults of the form $g_i/a_i \to a'_i/h_i$. (2) If the transition fault $g_i : a'_i \to a_i$ is still undetected, but the second-cycle stuck-at fault g_i/a'_i is detected accidentally by T_j, the procedure adds to M_1 n_{BR} bridging faults of the form $g_i/a_i \to a'_i/h_i$. The faults are selected randomly as described earlier.

Considering the test set T_j, for $j \geq 0$, after updating M_1, the bridging fault coverage is computed as the number of detected faults in the updated M_1, divided by the total number of faults in the updated M_1.

III. Experimental Results

The iterative test compaction procedure for M_0 and M_1 was applied to benchmark circuits as described in this section.

The test set T_0 consists of broadside and skewed-load tests for transition faults. The use of both test types allows more transition faults to be detected than using one of the test types alone. The test set was compacted by the test compaction procedure from [27].

The test compaction procedure is applied iteratively up to $J_{MAX} = 10$ times. Experimental results indicate that additional calls do not have a significant effect on the quality of the test set.

The results for several benchmark circuits are shown in Table I as follows. For every circuit, Table I contain a row for T_0, and for every test set T_j, where $1 \leq j \leq 10$, with an

TABLE I
Experimental Results

				faults			f.c.		
circuit	iter	tests	frac	undet	2sa	bridg	trans	bridg	ntime
s38417	0	503	1.10	337	85	76755	99.56	60.77	1.00
s38417	1	700	1.23	336	86	77658	99.56	95.86	903.67
s38417	2	792	1.41	329	81	73143	99.57	96.72	1636.78
s38417	3	788	1.47	325	77	69531	99.58	99.09	2219.44
s38417	4	776	1.59	321	73	65919	99.58	99.26	2730.63
s38417	5	744	1.68	320	72	65016	99.58	99.32	3289.43
s38417	6	733	1.82	318	71	64113	99.59	98.26	3816.58
s38417	7	712	1.81	318	71	64113	99.59	99.36	4336.53
s38417	8	669	1.99	318	71	64113	99.59	99.37	4817.34
s38417	9	670	2.10	318	71	64113	99.59	99.39	5317.73
s38417	10	663	2.14	318	72	65016	99.59	98.25	5828.79
b04	0	44	1.20	4	1	1138	99.82	34.97	1.00
b04	1	49	1.04	4	1	1138	99.82	93.50	74.14
b04	2	59	1.19	4	1	1138	99.82	94.38	150.57
b04	3	57	1.19	4	1	1138	99.82	94.38	227.57
b04	4	56	1.07	4	1	1138	99.82	94.38	294.14
b04	5	53	1.12	4	1	1138	99.82	94.38	368.71
b04	8	51	1.22	4	1	1138	99.82	94.38	595.43
b04	9	49	1.23	4	1	1138	99.82	94.38	665.29
b15	0	423	2.41	949	150	35550	97.32	18.52	1.00
b15	1	794	1.92	773	166	39342	97.82	38.18	630.19
b15	2	1024	1.74	634	93	22041	98.21	48.56	1252.91
b15	3	930	2.11	544	41	9717	98.46	57.49	1602.60
b15	4	766	2.70	509	37	8769	98.56	62.37	1817.72
b15	5	773	2.87	483	36	8532	98.64	67.58	2020.98
b15	6	792	2.44	480	33	7821	98.64	72.09	2214.51
b15	7	783	2.26	468	21	4977	98.68	79.73	2405.90
b15	8	764	2.08	466	23	5451	98.68	72.61	2577.43
b15	9	767	1.98	457	21	4977	98.71	71.63	2763.64
b15	10	773	1.88	423	22	5214	98.81	74.78	2946.73
i2c	0	84	2.23	51	46	4324	98.81	31.04	1.00
i2c	1	203	3.06	46	43	4042	98.93	80.38	204.23
i2c	2	204	3.25	44	43	4042	98.97	82.68	357.75
i2c	3	222	3.35	43	43	4042	99.00	84.81	507.24
i2c	4	234	2.90	43	43	4042	99.00	85.33	649.07
i2c	5	237	2.59	43	43	4042	99.00	85.70	802.07
i2c	6	228	2.56	38	38	3572	99.11	84.77	949.85
i2c	7	218	2.76	37	37	3478	99.14	84.68	1087.51
i2c	8	220	2.67	37	37	3478	99.14	84.96	1219.97
i2c	9	220	2.61	37	37	3478	99.14	85.08	1351.75
i2c	10	222	2.47	37	37	3478	99.14	85.14	1484.10
spi	0	460	6.67	36	34	11968	99.70	49.77	1.00
spi	1	648	4.49	35	33	11616	99.71	90.85	197.93
spi	2	675	4.00	32	31	10912	99.73	93.91	355.99
spi	3	676	3.60	31	30	10560	99.74	94.36	494.61
spi	4	644	3.50	23	22	7744	99.81	93.36	623.61
spi	5	663	3.54	21	20	7040	99.82	93.96	758.33
spi	6	644	3.63	21	20	7040	99.82	94.01	887.85
spi	7	629	3.46	21	20	7040	99.82	94.02	1011.76
spi	8	645	3.51	21	20	7040	99.82	94.03	1139.26
spi	9	632	3.58	21	20	7040	99.82	94.06	1262.69
spi	10	628	3.39	21	20	7040	99.82	94.08	1389.12
wb_dma	0	138	2.14	53	43	16598	99.68	65.26	1.00
wb_dma	1	238	2.26	50	45	17370	99.70	88.88	646.85
wb_dma	2	279	2.93	48	45	17370	99.71	90.87	1248.42
wb_dma	3	286	3.33	45	43	16598	99.73	92.62	1783.82
wb_dma	4	302	3.95	44	43	16598	99.73	92.81	2346.62
wb_dma	5	311	4.36	43	42	16212	99.74	94.81	2917.81
wb_dma	6	315	3.92	43	42	16212	99.74	94.97	3390.17
wb_dma	7	315	4.73	42	42	16212	99.75	95.05	3882.35
wb_dma	8	316	4.64	42	42	16212	99.75	95.16	4398.43
wb_dma	9	317	4.56	42	42	16212	99.75	95.22	4901.75
wb_dma	10	312	4.67	42	42	16212	99.75	95.23	5408.88

improved quality (either the fault coverage with respect to M_0 or M_1 is increased, or the number of tests is reduced).

For every test set T_j, column $iter$ shows the iteration, j. Column $tests$ shows the number of tests in T_j. Column $frac$ shows the ratio of skewed-load to broadside tests in T_j.

Column $faults$ subcolumn $undet$ shows the number of transition faults that are not detected by T_j. Subcolumn $2sa$ shows the number of undetected transition faults for which

the corresponding second-cycle stuck-at faults are detected. Subcolumn $bridg$ shows the number of bridging faults in M_1. As discussed earlier, the number of bridging faults is selected such that it is approximately equal to the number of transition faults. The number of bridging faults decreases if new transition faults are detected accidentally. It increases if new second-cycle stuck-at faults corresponding to undetected transition faults are detected.

Column $f.c.$ shows the transition fault coverage, and the bridging fault coverage of T_j. Column $ntime$ shows the normalized run time. This is the cumulative run time, divided by the run time for fault simulation with fault dropping of M_0 and M_1 under T_0.

From Table I it can be observed that T_0 achieves a high transition fault coverage but a low bridging fault coverage. The test compaction procedure is able to increase the bridging fault coverage significantly for all the circuits. It thus allows lines to be covered that are not covered by transition faults.

An iterative application of the test compaction procedure is important since the set of bridging faults M_1 is adjusted based on a test set T_j after it is generated. However, the procedure does not have to be run for a large number of iterations. Two applications of the procedure are typically sufficient.

Overall, these results show that it is possible to provide additional coverage for components with undetectable faults of one fault model by using a second fault model.

IV. Concluding Remarks

This paper studied the possibility of defining fault models where the existence of an undetectable fault in the first model does not imply that a fault in the same component from the second model is undetectable. This allows the second fault model to cover components that are not covered by the first fault model, thus covering the circuit more thoroughly. The paper considered transition faults as the first fault model, and a special type of bridging faults as the second fault model. The bridging faults were defined to cover lines for which transition faults are not detected by a given test set. A test compaction procedure was developed that considers the two fault models. Experimental results demonstrated the ability to increase the bridging fault coverage for benchmark circuits.

References

[1] Z. Barzilai and B. Rosen, "Comparison of AC Self-Testing Procedures", in Proc. Intl. Test Conf., 1983, pp. 89-94.

[2] G. L. Smith, "Model for Delay Faults Based Upon Paths", in Proc. Intl. Test Conf., 1985, pp. 342-349.

[3] J. L. Carter, V. S. Iyengar and B. K. Rosen, "Efficient Test Coverage Determination for Delay Faults", in Proc. Intl. Test Conf., 1987, pp. 418-427.

[4] W. Maly, P. K. Nag and P. Nigh, "Testing Oriented Analysis of CMOS ICs with Opens", IEEE Trans. on Computer-Aided Design, Nov. 1988, pp. 344-347.

[5] M. Renovell and G. M. Cambon, "Electrical Analysis of Floating-Gate Fault", IEEE Trans. on Computer-Aided Design, Nov. 1992, pp. 1450-1458.

[6] M. Dalpasso, M. Favalli, P. Olivo and B. Ricco, "Fault Simulation of Parametric Bridging Faults in CMOS IC's", IEEE Trans. Computer-Aided Design, Sept. 1993, pp. 1403-1410.

[7] P. C. Maxwell and R. C. Aitken, "Biased Voting: A Method for Simulating CMOS Bridging Faults in the Presence of Variable Gate Logic Thresholds", in Proc. Intl. Test Conf., 1993, pp. 63-72.

[8] H. Konuk and F. J. Ferguson, "An Unexpected Factor in Testing for CMOS Opens: The Die Surface", in Proc. VLSI Test Symp., 1996, pp. 422-429.

[9] S. Rafiq, A. Ivanov, Y. Bertrand, F. Azias and M. Renovell, "Testing for Floating Gates Defects in CMOS Circuits", in Proc. Asian Test Symp., 1998, pp. 228-236.

[10] S. Sengupta, S. Kundu, S. Chakravarty, P. Parvathala, R. Galivanche, G. Kosonocky, M. Rodgers and T. M. Mak, "Defect-Based Tests: A Key Enabler for Successful Migration to Structural Test", Intel Technology Journal, Q.1, 1999.

[11] H. Konuk, "Voltage and Current based Fault Simulation for Interconnect Open Defects", IEEE Trans. on Computer-Aided Design, Dec. 1999, pp. 1768-1779.

[12] S. Ma, I. Shaik and R. S. Fetherston, "A Comparison of Bridging Fault Simulation Methods", in Proc. Intl. Test Conf., 1999, pp. 587-595.

[13] V. R. Sar-Dessai and D. M. H. Walker, "Resistive Bridge Fault Modeling, Simulation and Test Generation", in Proc. Intl. Test Conf., 1999, pp. 596-605.

[14] T. Maeda and K. Kinoshita, "Precise Test Generation for Resistive Bridging Faults of CMOS Combinational Circuits", in Proc. Intl. Test Conf., 2000, pp. 510-519.

[15] V. Krishnaswamy, A. B. Ma, P. Vishakantaiah, "A Study of Bridging Defect Probabilities on a Pentium (TM) 4 CPU", Intl. Test Conf., 2001, pp. 688-695.

[16] V. H. Champac and A. Zenteno, "Detectability Conditions for Interconnect Open Defects", in Proc. VLSI Test Symp., 2005, pp. 305-311.

[17] I. Polian, S. Kundu, J.-M. Galliere, P. Engelke, M. Renovell and B. Becker, "Resistive Bridge Fault Model Evolution from Conventional to Ultra Deep Submicron Technologies", in Proc. VLSI Test Symp., 2005, pp. 343-348.

[18] D. Arumi, R. Rodriguez-Montanes and J. Figueras, "Defective Behaviours of Resistive Opens in Interconnect Lines", in Proc. European Test Symp., 2005, pp. 28-33.

[19] R. Gomez, A. Giron and V. Champac, "Test of Interconnect Opens Considering Coupling Signals" in Proc. Defect and Fault Tolerant VLSI Systems Symp., 2005, pp. 247-255.

[20] R. Rodriguez-Montanes and J. Figueras, "Electrical and Topological Characterization of Interconnect Open Defects", IEEE Intl. Workshop on Current and Defect Based Testing, 2005.

[21] I. Pomeranz and S. M. Reddy, "Generation of Broadside Transition Fault Test Sets that Detect Four-Way Bridging Faults", IEEE Trans. on Computer-Aided Design, July 2007, pp. 1311-1319.

[22] H. Takahashi, Y. Higami, S. Kadoyama, T. Aikyo, Y. Takamatsu, K. Yamazaki, H. Yotsuyanagi and M. Hashizume, "Clues for Modeling and Diagnosing Open Faults with Considering Adjacent Lines", in Proc. Asian Test Symp., 2007, pp. 39-44.

[23] A. Sreedhar, A. Sanyal and S. Kundu, "On Modeling and Testing of Lithography Related Open Faults in Nano-CMOS Circuits", in Proc. Design, Autom. and Test in Europe Conf., 2008, pp. 616-621.

[24] S. Spinner, I. Polian, P. Engelke, B. Becker, M. Keim and W.-T. Cheng, "Automatic Test Pattern Generation for Interconnect Open Defects", in Proc. VLSI Test Symp., 2008, pp. 181-186.

[25] B. Benware, C. Schuermyer, N. Tamarapalli, K.-H. Tsai, S. Ranganathan, R. Madge, J. Rajski and P. Krishnamurthy, "Impact of multiple-detect test patterns on product quality", in Proc. Intl. Test Conf., 2003, pp. 1031-1040.

[26] S. Venkataraman, S. Sivaraj, E. Amyeen, S. Lee, A Ojha and R. Guo, "An Experimental Study of n-Detect Scan ATPG Patterns on a Processor", in Proc. VLSI Test Symp., 2004, pp. 23-28.

[27] I. Pomeranz, "Static Test Compaction for Mixed Broadside and Skewed-Load Transition Fault Test Sets", IET Computers & Digital Techniques, Jan. 2013, pp. 21-28.