Survey of Scan Chain Diagnosis

Yu Huang, Ruifeng Guo, and Wu-Tung Cheng
Mentor Graphics

James Chien-Mo Li
National Taiwan University

Editor's note:
What happens when the diagnostic infrastructure itself fails? How does diagnosis proceed? This article provides a survey of the available techniques to meet the challenge.

—Rob Aitken, ARM

SCAN-BASED TESTING HAS proven to be a cost-effective method for achieving good test coverage in digital circuits. The Achilles heel in the application of scan-based testing is the integrity of the scan chains. The amount of die area consumed by scan elements, chain connections, and control circuitry varies with different designs. Scan elements and clocking can occupy nearly 30% of a chip's area.[1] The percentage of scan chain defects also varies with different designs. From 10% to 30% of all defects cause scan chains to fail,[2] and chain failures account for almost 50% of chip failures.[3] Therefore, scan chain failure diagnosis is important to effective scan-based testing.

Typically, each scan cell in a scan chain has an index number. The cells in the chain are sequentially numbered from scan output to scan input, starting with 0. A *chain pattern* (sometimes called a *flush pattern*[4]) is a pattern consisting of shift-in and shift-out operations without pulsing capture clocks. The purpose of chain patterns is to test scan chain integrity. A *scan pattern* (also known as a *logic test pattern*) is a pattern consisting of a shift-in operation, one or multiple capture clock cycles, and a shift-out operation. The purpose of scan patterns is to test system logic. The scan cells between the scan chain input and a scan cell's scan input terminal are called the *upstream cells* of that scan cell. The scan cells between the scan chain output and a scan cell's scan output terminal are called the *downstream cells* of that scan cell.

Scan chain fault models include *stuck-at faults* (stuck-at-0 and stuck-at-1), *slow faults* (slow-to-rise, slow-to-fall, and slow), and *fast faults* (fast-to-rise, fast-to-fall, and fast).[2] Slow faults result from setup time violations, and fast faults from hold-time violations (slow and fast faults are also called *timing faults*). Using a specific fault model, it's also possible to model a scan chain defect as a *permanent fault*, which occurs in all shift cycles, or an *intermittent fault*, which occurs in a subset of shift cycles.[5]

Table 1 shows an example of identifying faulty chains and modeling chain defects by chain patterns. Suppose a scan chain with 12 scan cells is loaded with chain pattern 001100110011, in which the leftmost bit is loaded into cell 11 and the rightmost bit is loaded into cell 0. The second column gives the unloaded faulty values for each type of permanent fault. The third column gives examples of unloaded faulty values for each type of intermittent fault. Underlines show differences between expected unloaded values and observed values. By using this table, the best chain fault model to be used for diagnosis can be identified.

Chain patterns alone are sufficient to determine the fault type, but insufficient to pinpoint the index of a failing flip-flop. This is the fundamental motivation for chain failure diagnosis, which is the process of identifying one or multiple defective scan cells in a scan chain or defective scan-enable or clock signals. In this article, we survey chain fault diagnosis techniques, which we classify into three categories: tester based, hardware based, and software based.

Tester-based chain diagnosis

Tester-based diagnosis techniques use a tester to control scan chain shift operations, and physical

0740-7475/08/$25.00 © 2008 IEEE Copublished by the IEEE CS and the IEEE CASS

Table 1. Scan chain fault models and their effects. (Fault-free unloaded values are 001100110011; underlines indicate different expected and observed values.)

Fault models	Unloaded values with one permanent fault	Unloaded values with one intermittent fault (examples)
Slow-to-rise	00100010001X	00110010001X
Slow-to-fall	01110111011X	01110011011X
Slow	01100110011X	00100111011X
Fast-to-rise	X0111011011	X011101100 11
Fast-to-fall	X0010001000 1	X0010011000 1
Fast	X0011001100 1	X0010011001
Stuck-at-0	000000000000	001000010000
Stuck-at-1	111111111111	101111111011

failure analysis (PFA) equipment to observe defective responses at different locations and identify failing scan cells. These techniques normally provide very good diagnosis resolution. However, they require expensive, time-consuming, and often destructive sample preparation, and they provide visibility only through a small peephole. Hence, you must know where to look with your PFA equipment.

De and Gunda propose a tester-based technique in which they apply a chain pattern of alternating 0 s and 1 s and use electron-beam probing to detect the toggles.[6] They apply a binary-search scheme to detect a stuck-at fault at a cell where the toggles start to disappear.

Song et al. propose a diagnostic method based on light emission due to off-state leakage current (LEOSLC).[7] They apply two chain patterns: all 0 s and all 1 s. They compare two emission images of a cell for both chain patterns. If there is no difference, a stuck-at fault could be on this cell or its upstream cells. This procedure is repeated until reaching the first cell that shows a different emission image for all 0 s than for all 1 s. Applying a binary search can speed up the process. Stellari et al. combine LEOSLC with picosecond imaging circuit analysis technology to enhance the efficiency and effectiveness of chain diagnosis.[8]

If passing or failing of scan shift operating conditions such as power supply, reference voltages, or clock speed can be identified, then passing or failing conditions can be used to shift in a chain pattern and change the test environment to the opposite condition for shift out. The location where failures start to appear (or disappear) is the defect location. Three groups of researchers have proposed techniques of this type. Motika et al. identify the passing or failing shift speed to diagnose slow faults.[9] By varying operating parameters, Motika, Nigh, and Song trigger one or more latches downstream from the fault location to change state from the stuck-at fault value.[10] Kong and Islam perform a shmoo plot logging the result of the chain test results with respect to voltage, frequency, and temperature to identify passing and failing test conditions.[11]

Hirase, Shindou, and Akahori use I_{DDQ} testing for chain diagnosis.[12] Taking the stuck-at-1 fault for example, if 0111… was shifted in, when the 0 was shifted to the cell with a stuck-at-l fault, I_{DDQ} would have an abnormally high value.

Hardware-based chain diagnosis

Hardware-based methods use special scan chain and scan cell designs to facilitate diagnosis. These techniques are effective in isolating scan chain defects. However, because they typically require extra hardware overhead, they are not acceptable in many products. In addition, if defects occur in the extra control hardware, diagnosis becomes more complicated.

Schafer, Policastri, and McNulty proposed connecting each scan cell's output to a scan cell (called the partner shift register) in another scan chain so that its value could be observed from the other scan chain in diagnostic mode.[13] For example, assume there is one stuck-at-0 at the output of cell 2 of chain 1, and chain 1 has four cells. After shifting in 1111, chain 1 should have 1100. Then the circuit is turned into diagnostic mode, and the data in chain 1 is transferred to its partner chain. Assuming the partner chain is a good chain, 1100 is observed from this chain, and it can be deduced that the defect must be in the middle of chain 1.

In another hardware-based method, S. and G. Edirisooriya insert XOR gates between scan cells to

enhance chain diagnosis.[14] In case of multiple faults, the proposed scheme will always identify the fault closest to the scan output. The scheme makes a trade-off between the number of XOR gates added and the diagnostic resolution. The same authors also proposed a dictionary-based chain failure diagnosis technique using the special scan chain design.[15] They create a fault dictionary for each scan cell fault and analyze the responses with XOR gates along the scan chain to identify the failing scan cell.

Narayanan and Das proposed adding simple circuitry to a scan flip-flop to enable its scan-out port to be either set or reset.[16,17] The authors presented a global strategy based on the set/reset feature to account for disparities in defect probabilities and controllability and observability attributes of flip-flops in a scan chain. They also presented an algorithm to optimally modify a subset of the flip-flops to maximize diagnostic resolution. One solution is that each adjacent pair of flip-flops consists of a flip-flop whose scan output can be reset to 0, and a flip-flop whose scan output can be set to 1. Hence, any single stuck-at fault can be diagnosed down to a pair of flip-flops.

Wu proposed a special circuit to flip, set, or reset scan cells to identify defective cells.[18] After shifting in a chain pattern, the circuit can invert, set, or reset each flip-flop's state. The faulty cell is located via the observed unloading value. Song proposed a bidirectional scan chain architecture in which the scan chain performs both forward and backward scan shift to diagnose scan faults.[19]

Motika, Nigh, and Tran apply an on-chip controller for scan chain diagnosis.[20] Each chain is divided into multiple shorter subchains through multiplexers. The controller controls each subchain's inputs and outputs independently. The multiple-input signature register (MISR) observes each subchain while the controller masks the other subchains.

Tekumulla and Lee propose partitioning scan chains into segments, and bypassing segments that contain hold-time violations.[21] When a hold-time violation is located on a scan chain segment, the flip-flop in that segment is bypassed and new test patterns are derived.

Software-based chain diagnosis

Software-based techniques use algorithmic diagnosis to identify failing scan cells. Compared with hardware-based methods, software-based techniques are more widely applied in industry for general designs, because no design modification is required. Software-based techniques fall into two categories: using production scan patterns and generating special diagnostic chain patterns.

Production scan patterns

We further classify production scan methods as simulation based, probability based, and dictionary based.

Simulation-based methods. Stanley uses fault injection and simulation to find faulty scan cells, injecting one fault in a cell for each run.[4] Because all scan cells on a faulty chain are candidates, this method is time-consuming for a scan chain with many cells. To speed up the diagnosis procedure, researchers have proposed several techniques.

For example, Guo and Venkataraman proposed an algorithm that identifies an upper bound (UB) and lower bound (LB) for a faulty cell. Figure 1a shows an example to explain this algorithm. First, the faulty chain's simulated loading values are changed to all Xs. After the capture clock pulses, assume the simulated captured values on this faulty chain are XX10XXX0XX1X. That means cells 8 and 4 will capture 0 s no matter what values were actually loaded to the faulty chain. Suppose the observed values on the ATE are actually 111111001010. Because the observed value at scan cell 8 is 1, a stuck-at-1 fault must be downstream of cell 8. So, cell 8 is the UB. Meanwhile, because the observed value at cell 4 matches the simulated value, the stuck-at-1 fault must be upstream of cell 4. So, cell 4 is the LB. Ranking the suspect cells within the bounded range further improves the diagnosis resolution. The same authors provide experimental results of applying the technique to industrial designs.[22] They also give more details of this diagnosis method and its application to production test fallouts, using several case studies.[23]

Figure 1b illustrates another method for speeding up simulation-based diagnosis. Kao, Chuang, and Li propose jump simulation to diagnose a single chain fault.[24] For each failing pattern, a simulator performs multiple simulations to quickly search multiple UB or LB fault segments. After the range is finalized, a detailed simulator performs parallel pattern simulation for every fault in the final range. Suppose there is a stuck-at-1 fault on a scan chain and the current UB = 27 and the current LB = 20. The scan cells from the UB

to the LB are evenly divided into three parts, and the boundary scan cells (22, 24, and 26) are chosen as jump bits. In searching for a new UB, the algorithm assumes the fault is upstream from the jump bit. It changes all 0 s downstream from the jump bit to 1 s, and all 0 s between the jump bit and the UB to Xs. If a simulation mismatch occurs in the second jump bit (24), the algorithm deduces that the stuck-at-1 fault is actually downstream from the jump bit. It therefore moves the new UB to scan cell 23. It searches for the LB in a similar way.

Huang proposed a simulation-based method using dynamic learning.[25] This algorithm was based on several learning rules. These rules analyzed the circuit, patterns, and mismatched bits and back-traced the logic cones to determine which cells should be simulated in the next iteration. As a result, rather than simulating every cell within a range, only a few cells need to be simulated to find suspects. Figure 2a shows an example of a technique that updates the LB. Here, a fault is injected at the current LB at cell 1. If there is a simulation mismatch on the cell of a good chain (the shaded box in Figure 2a), we can back-trace the fault from the mismatched cell. Assuming this cell is driven by cells 4 and 3 on the faulty chain, we learn that either cell 4 or cell 3 or both carried wrong loading values in the previous simulation. Therefore, the new LB is updated to scan cell 3. This process can be iterated several times until the actual defective cell is found.

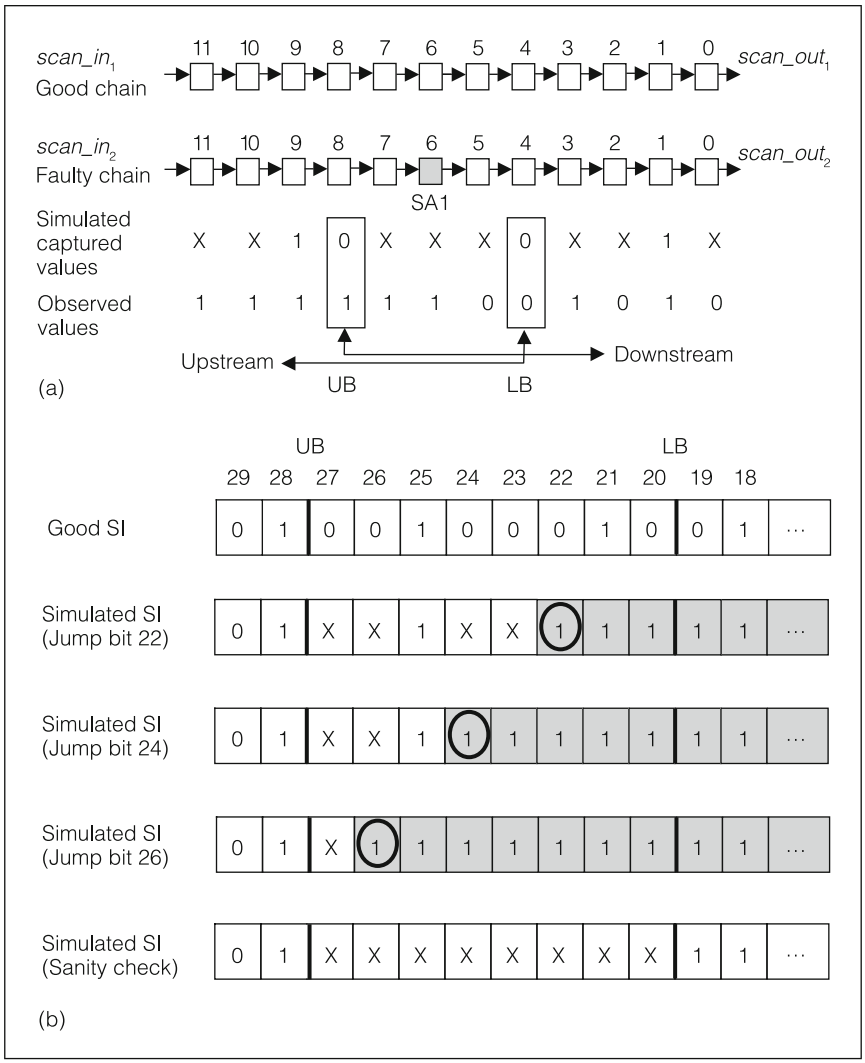

Figure 1. Techniques for speeding up simulation-based chain diagnosis: upper bound (UB) and lower bound (LB) identification (a), and jump simulation (b). Circled numbers represent jump bits. Shaded boxes represent the affected scan cells between the LB and the jump bit. (SA: stuck-at, SI: scan input.)

Huang et al. discuss diagnosis of intermittent hold-time faults and propose an algorithm based on X simulation in which intermittent loading and unloading behavior is modeled with Xs.[26] Huang, Cheng, and Crowell present case studies to illustrate the problems of using a fault model to diagnose real chain defects.[27] They propose a fault model relaxation flow in which chain fault models are adaptively selected according to fault model relaxation rules and simulation results.

Chain diagnosis on devices with embedded compression techniques is a challenge. Huang, Cheng, and Rajski proposed a methodology that enables seamless reuse of existing chain diagnosis algorithms with compressed test data.[28] Huang and Gallie proposed an algorithm that locates the defects on the scan-enable tree for a multiplexed data flip-flop (mux-DFF) scan architecture.[29] The algorithm uses simulation and postprocessing of diagnosis results by tracing the scan-enable tree. The authors extended the algorithm to diagnose clock tree defects.[30] Sarrica and Kessler proposed an algorithm for diagnosing scan clock defects in the level-sensitive scan design (LSSD) architecture.[31]

The ATE's fail buffer capacity and test time restrict the total number of failing bits that can be logged, negatively affecting the diagnosis resolution. Huang et

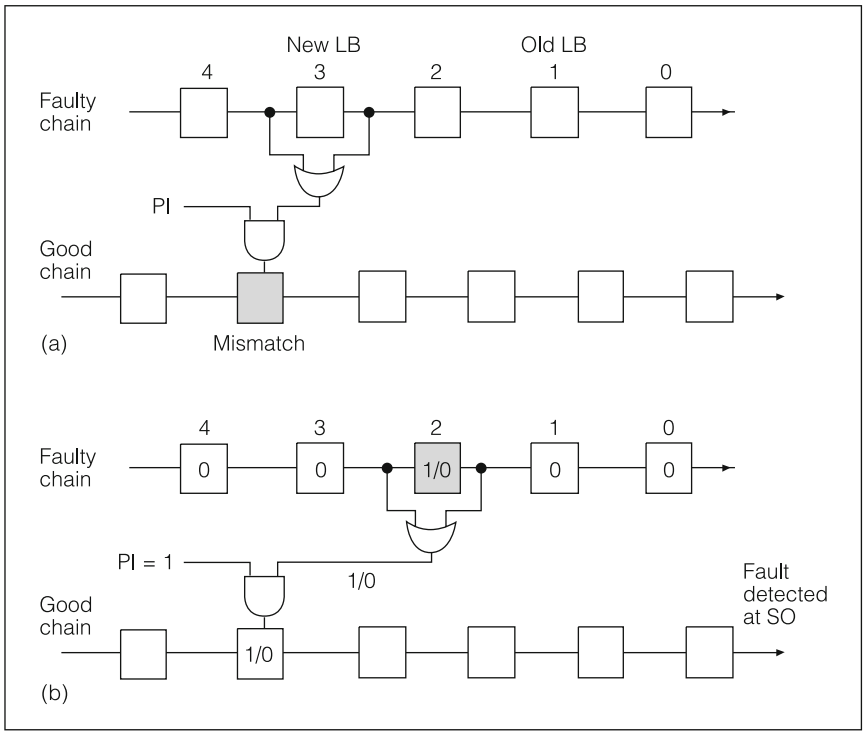

Figure 2. Dynamic learning (a) and single-excitation ATPG (b). (SO: scan output.)

al. proposed three methods of running chain diagnosis with limited failures: static pattern reordering, dynamic pattern reordering, and per-pin-based diagnosis.[32]

Compound defects—scan chain defects and system logic defects that coexist on the same die—make diagnosing real defects challenging.[33] Huang et al. discussed a special compound defect that can impact both chain and system logic simultaneously, and they proposed using per-shift-cycle simulation to identify defect locations.[34] Huang et al. introduced an algorithm for diagnosing more-general compound defects. It first separates failures caused by faulty chains from those caused by faulty system logic. It then masks the faulty scan chains to diagnose system logic defects, and vice versa. Ahmed et al. presented a case study of yield enhancement due to successful simultaneous diagnosis of scan chain hold-time faults and system logic faults.[35]

Probability-based methods. Probability-based chain diagnosis algorithms primarily target intermittent chain faults. Huang et al. proposed a statistical-diagnosis algorithm based on Bayes' theorem to calculate a cell's probability of being faulty.[5] Huang et al. proposed an algorithm that incorporates signal probability calculation.[36] It injects one fault at a time to the faulty scan chain and searches for the best-matching candidate on the basis of probabilities.

Dictionary-based method. Guo, Huang, and Cheng proposed a dictionary-based technique for scan chain failure diagnosis.[37] In this technique, differential signatures are stored in fault dictionaries to reduce the fault signature redundancy of adjacent scan cell faults. The differential signatures serve to diagnose single stuck-at faults, timing faults, and some multiple stuck-at faults in a single scan chain.

Chain diagnostic pattern generation

When production scan patterns cannot provide good diagnosis resolution, special diagnostic patterns are necessary to achieve better diagnosis resolution. Researchers have proposed several techniques for generating patterns.

Kundu proposed a scan chain diagnosis algorithm that focuses on generating test patterns for stuck-at faults.[1,38] It creates test patterns either to capture desired values in target scan cells or to propagate fault effects to good scan chains for failure observation. Several other researchers use similar methods.[39–41]

Yang and Huang proposed using functional test patterns for scan chain failure diagnosis.[3] Their procedure selected patterns to randomize the signal probability of scan cells. By comparing the observed signal profile on a tester and the expected signal profile along a faulty scan chain, test engineers can identify the failing scan cell's position.

Several researchers proposed chain algorithms that include two parts:

- use diagnostic ATPG to obtain scan patterns that don't use chain-loading procedures so that the impacts of chain defects come only from chain-unloading procedures, and
- apply heuristics to analyze test failures and identify defective cells.[42–44] The heuristics include signal profiling, best alignment, delay insertion, and image recovering.

Li proposed a single-excitation technique to generate diagnostic patterns.[45,46] Single-excitation pat-

Table 2. Classification of chain diagnosis techniques (by reference number).

		Software based			
		Production scan patterns			
Tester based	Hardware based	Simulation based	Probability based	Dictionary based	Diagnostic ATPG
6–12	13–21	2, 4, 22–35	5, 36	37	1, 3, 38–49

terns have only one sensitive bit that can be flipped by the fault. This technique converts the diagnosis problem into a single-stuck-at-fault ATPG problem, which existing tools can easily solve. Figure 2b shows an example. Suppose that a stuck-at-0 chain fault exists. The single-excitation pattern 00100 shifts into the faulty chain, making cell 2 the sensitive bit. Hence, this technique detects a fault in the same way as it would detect a stuck-at-0 fault in combinational logic.

Crouch suggested propagating fault effects to as many primary outputs and good scan chains as possible.[47] He also proposed adding shift cycles between capture clocks, which can be helpful for diagnosing multiple chain faults. Sinanoglu and Schremmer proposed generating test stimuli (such as all 0 s or all 1 s) that are immune to hold-time violations on the faulty chain and randomly changing stimuli on the good chains.[48] Guo, Huang, and Cheng proposed a complete test set generation technique for single-chain fault diagnosis.[49] This technique attempts to create test patterns that uniquely identify any faulty scan cell. The authors extended the algorithm to handle multiple failing scan chains and designs containing test compression logic. During test generation, the algorithm carefully analyzes constraints on scan cell controllability and observability if there are logic correlations between scan cells of the same scan chain.

New directions

Several aspects of current chain diagnosis tools and techniques still need improvement:

- Diagnosing multiple faults per chain is important for diagnosing chain failures caused by systematic defects, library cell reliability problems, or process variations.
- Because of a gap between fault models and real defects, modeled faults show up only under certain situations. Diagnosis resolution must be enhanced for intermittent faults.
- A reliable solution for diagnosis of defects on clocks, scan-enable signals, and embedded-compactor logic is needed.
- Runtime needs improvement to speed up volume diagnosis of large quantities of chips in production for yield learning.
- Chain defects produce many failure cycles, but tester memory capacity is limited. Performing chain diagnosis with central-buffer-based testers is still challenging.
- All currently used chain fault models are cell-based, so diagnosis resolution is at best one cell. Normally, a scan cell and its connections spread over a large area in silicon. Therefore, enhancement of resolution down to a specific signal or pin would be more helpful for physical failure analysis.

Table 2 classifies the chain diagnosis techniques we have presented. The various techniques have their own application scenarios, advantages, and disadvantages. Tester-based diagnosis techniques are very effective but are time-consuming and costly. Special scan designs for chain diagnosis are useful but are not available in most real designs. Software-based diagnosis can be easily automated for quick fault diagnosis but still needs enhancements of diagnosis resolution and runtime. ■

References

1. S. Kundu, "On Diagnosis of Faults in a Scan-Chain," *Proc. 11th Ann. IEEE VLSI Test Symp.* (VTS 93), IEEE Press, 1993, pp. 303-308.
2. R. Guo and S. Venkataranman, "A Technique for Fault Diagnosis of Defects in Scan Chains," *Proc. Int'l Test Conf.* (ITC 01), IEEE CS Press, 2001, pp. 268-277.
3. J.-S. Yang and S.-Y. Huang, "Quick Scan Chain Diagnosis Using Signal Profiling," *Proc. Int'l Conf. Computer Design* (ICCD 05), IEEE CS Press, 2005, pp. 157-160.
4. K. Stanley, "High Accuracy Flush-and-Scan Software Diagnostic," *IEEE Design & Test*, vol. 18, no. 6, Nov.-Dec. 2001, pp. 56-62.

5. Y. Huang et al., "Statistical Diagnosis for Intermittent Scan Chain Hold-Time Fault," *Proc. Int'l Test Conf.* (ITC 03), IEEE CS Press, 2003, pp. 319-328.
6. K. De and A. Gunda, "Failure Analysis for Full-Scan Circuits," *Proc. Int'l Test Conf.* (ITC 95), IEEE Press, 1995, pp. 636-645.
7. P. Song et al., "A Novel Scan Chain Diagnostics Technique Based on Light Emission from Leakage Current," *Proc. Int'l Test Conf.* (ITC 04), IEEE CS Press, 2004, pp. 140-147.
8. F. Stellari et al., "Broken Scan Chain Diagnostics Based on Time-Integrated and Time-Dependent Emission Measurements," *Proc. 30th Int'l Symp. Testing and Failure Analysis* (ISTFA 04), ASM Int'l, 2004, pp. 52-57.
9. F. Motika et al., *AC Scan Diagnostic Method*, US patent 6516432, Patent and Trademark Office, 2003.
10. F. Motika, P.J. Nigh, and P. Song, *Stuck-At Fault Scan Chain Diagnostic Method*, US patent 7010735, Patent and Trademark Office, 2006.
11. C.L. Kong and M.R. Islam, "Diagnosis of Multiple Scan Chain Faults," *Proc. Int'l Symp. Testing and Failure Analysis* (ISTFA 05), ASM Int'l, 2005, pp. 510-516.
12. J. Hirase, N. Shindou, and K. Akahori, "Scan Chain Diagnosis Using IDDQ Current Measurement," *Proc. Asian Test Symp.* (ATS 99), IEEE CS Press, 1999, pp. 153-157.
13. J.L. Schafer, F.A. Policastri, and R.J. McNulty, "Partner SRLs for Improved Shift Register Diagnostics," *Proc. 10th IEEE VLSI Test Symp.* (VTS 92), IEEE Press, 1992, pp. 198-201.
14. S. Edirisooriya and G. Edirisooriya, "Diagnosis of Scan Path Failures," *Proc. 13th IEEE VLSI Test Symp.* (VTS 95), IEEE Press, 1995, pp. 250-255.
15. G. Edirisooriya and S. Edirisooriya, "Scan Chain Fault Diagnosis with Fault Dictionaries," *Proc. Int'l Symp. Circuits and Systems* (ISCAS 95), IEEE Press, 1995, pp. 1912-1915.
16. S. Narayanan and A. Das, "An Efficient Scheme to Diagnose Scan Chains," *Proc. Int'l Test Conf.* (ITC 97), IEEE CS Press, 1997, pp. 704-713.
17. S. Narayanan and A. Das, *Flip-Flop Design and Technique for Scan Chain Diagnosis*, US patent 5881067, Patent and Trademark Office, 1999.
18. Y. Wu, "Diagnosis of Scan Chain Failures," *Proc. Int'l Symp. Defect and Fault Tolerance in VLSI Systems* (DFT 98), IEEE Press, 1998, pp. 217-222.
19. P. Song, "A New Scan Structure for Improving Scan Chain Diagnosis and Delay Fault Coverage," *Proc. 9th IEEE North Atlantic Test Workshop* (NATW 00), 2000, pp. 14-18; http://www.ele.uri.edu/natw00.
20. F. Motika, P.J. Nigh, and P.T. Tran, *Diagnostic Method for Structural Scan Chain Designs*, US patent 6961886, Patent and Trademark Office, 2005.
21. R.C. Tekumulla and D. Lee, "On Identifying and Bypassing Faulty Scan Segments," *Proc. 16th IEEE North Atlantic Test Workshop* (NATW 07), 2007, pp. 134-143; http://www.ee.duke.edu/NATW/2007.
22. R. Guo and S. Venkataraman, "A New Technique for Scan Chain Failure Diagnosis," *Proc. Int'l Symp. Testing and Failure Analysis* (ISTFA 02), ASM Int'l, 2002, pp. 723-732.
23. R. Guo and S. Venkataraman, "An Algorithmic Technique for Diagnosis of Faulty Scan Chains," *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, vol. 25, no. 9, Sept. 2006, pp. 1861-1867.
24. Y.-L. Kao, W.-S. Chuang, and J.C.-M. Li, "Jump Simulation: A Technique for Fast and Precise Scan Chain Fault Diagnosis," *Proc. Int'l Test Conf.* (ITC 06), IEEE CS Press, 2006, paper 297659 (9 pp.).
25. Y. Huang, "Dynamic Learning Based Scan Chain Diagnosis," *Proc. Design, Automation and Test in Europe Conf.* (DATE 07), IEEE CS Press, 2007, pp. 510-515.
26. Y. Huang et al., "Efficient Diagnosis for Multiple Intermittent Scan Chain Hold-Time Faults," *Proc. 12th Asian Test Symp.* (ATS 03), 2003, pp. 44-49.
27. Y. Huang, W.-T. Cheng, and G. Crowell, "Using Fault Model Relaxation to Diagnose Real Scan Chain Defects," *Proc. Asia and South Pacific Design Automation Conf.* (ASP-DAC 05), IEEE Press, 2005, pp. 1176-1179.
28. Y. Huang, W.-T. Cheng, and J. Rajski, "Compressed Pattern Diagnosis for Scan Chain Failures," *Proc. Int'l Test Conf.* (ITC 05), IEEE CS Press, 2005, pp. 751-759.
29. Y. Huang and K. Gallie, "Diagnosis of Defect on Scan Enable Tree," *Proc. 2nd Int'l Workshop Silicon Debug and Diagnosis* (SDD 05), 2005; http://evia.ucsd.edu/conferences/sdd/05/index.html.
30. Y. Huang and K. Gallie, "Diagnosis of Defects on Scan Enable and Clock Trees," *Proc. Design, Automation and Test in Europe Conf.* (DATE 06), IEEE CS Press, 2006, vol. 1, pp. 436-437.
31. G.A. Sarrica and B.R. Kessler, "Theory and Implementation of LSSD Scan Ring & STUMPS Channel Test and Diagnosis," *Proc. 13th IEEE/CHMT Int'l Electronics Manufacturing Technology Symp.*, IEEE Press, 1992, pp. 195-200.
32. Y. Huang et al., "Diagnosis with Limited Failure Information," *Proc. Int'l Test Conf.* (ITC 06), IEEE CS Press, 2006, paper 297660 (10 pp.).

33. Y. Huang et al., "Diagnose Compound Scan Chain and System Logic Defects," *Proc. Int'l Test Conf.* (ITC 07), IEEE CS Press, 2007, paper 4437578 (10 pp.).
34. Y. Huang et al., "Diagnosing DACS (Defects That Affect Scan Chain and System Logic)," *Proc. 30th Int'l Symp. Testing and Failure Analysis* (ISTFA 04), ASM Int'l, 2004, pp. 191-196.
35. I. Ahmed et al., "Yield Improvement with Compressed Pattern Diagnosis," *Proc. 3rd IEEE Int'l Workshop Silicon Debug and Diagnosis* (SDD 06), 2006; http://evia.ucsd.edu/conferences/sdd/06/index.html.
36. Y. Huang et al., "Intermittent Scan Chain Fault Diagnosis Based on Signal Probability Analysis," *Proc. Design, Automation and Test in Europe Conf.* (DATE 04), IEEE CS Press, 2004, vol. 2, pp. 1072-1077.
37. R. Guo, R. Huang, and W.-T. Cheng, "Fault Dictionary Based Scan Chain Failure Diagnosis," *Proc. 16th Asian Test Symp.* (ATS 07), IEEE CS Press, 2007, pp. 45-50.
38. S. Kundu, "Diagnosing Scan Chain Faults," *IEEE Trans. Very Large Scale Integration (VLSI) Systems*, vol. 2, no. 4, Dec. 1994, 512-516.
39. O.P. Forlenza et al., *Look Ahead Scan Chain Diagnostic Method*, US patent 6308290, Patent and Trademark Office, 2001.
40. A.C. Anderson et al., *Method, Apparatus, and Computer Program Product for Implementing Deterministic Based Broken Scan Chain Diagnostics*, US patent application 20050229057, Patent and Trademark Office, 2005.
41. V. Brunkhorst et al., *Method for Optimizing a Set of Scan Diagnostic Patterns*, US patent 6996791, Patent and Trademark Office, 2006.
42. E. Hsu, S.-Y. Huang, and C.-W. Tzeng, "A New Robust Paradigm for Diagnosing Hold-Time Faults in Scan Chains," *Proc. IEEE Int'l Symp. VLSI Design, Automation and Test* (VLSI-DAT 06), IEEE Press, 2006, pp. 171-174.
43. C.-W. Tzeng and S.-Y. Huang, "Diagnosis by Image Recovery: Finding Mixed Multiple Timing Faults in a Scan Chain," *IEEE Trans. Circuits and Systems II*, vol. 54, no. 8, Aug. 2007, pp. 690-694.
44. C.-W. Tzeng, J.-J. Hsu, and S.-Y. Huang, "A Robust Paradigm for Diagnosing Hold-Time Faults in Scan Chains," *IET Proc. Computers and Digital Techniques*, vol. 1, no. 6, 2007, pp. 706-715.
45. J.C.-M. Li, "Diagnosis of Single Stuck-At Faults and Multiple Timing Faults in Scan Chains," *IEEE Trans. Very Large Scale Integration (VLSI) Systems*, vol. 13, no. 6, June 2005, pp. 708-718.
46. J.C.-M. Li, "Diagnosis of Multiple Hold-Time and Setup-Time Faults in Scan Chains," *IEEE Trans. Computers*, vol. 54, no. 11, Nov. 2005, pp. 1467-1472.
47. A. Crouch, "Debugging and Diagnosing Scan Chains," *Electronic Device Failure Analysis*, vol. 7, no. 1, Feb. 2005, pp. 16-24.
48. O. Sinanoglu and P. Schremmer, "Diagnosis, Modeling and Tolerance of Scan Chain Hold-Time Violations," *Proc. Design, Automation and Test in Europe Conf.* (DATE 07), IEEE CS Press, 2007, pp. 516-521.
49. R. Guo, Y. Huang, and W.-T. Cheng, "A Complete Test Set to Diagnose Scan Chain Failures," *Proc. Int'l Test Conf.* (ITC 07), IEEE Press, 2007, paper 4437579 (10 pp.).

Yu Huang is a senior staff member in the Advanced Research Group in the DFT Division of Mentor Graphics. His research interests include VLSI testing and diagnosis. He has a BS in electronic science and an MS in photo electronic thin film devices and technology, both from Nankai University, China; and a PhD in electrical and computer engineering from the University of Iowa. He is a member of the IEEE.

Ruifeng Guo is an R&D engineer at Mentor Graphics. His research interests include VLSI testing, diagnosis, and yield improvement. He has a BS in electronic science and technology from Nankai University, Tianjin, China, an MS in electronics engineering and computer science from Peking University, Beijing, and a PhD in electrical and computer engineering from the University of Iowa. He is a member of the IEEE and the IEEE Computer Society.

Wu-Tung Cheng is a chief scientist and an advanced test research director at Mentor Graphics. His research interests include developing new DFT solutions for future semiconductor quality and yield issues. He has a BS and an MS in electrical engineering from National Taiwan University, and a PhD in computer science from the University

of Illinois at Urbana-Champaign. He is an IEEE Fellow.

James Chien-Mo Li is an associate professor in the Graduate Institute of Electronics Engineering at National Taiwan University, Taipei. His research interests include DFT, BIST, low-power testing, and fault diagnosis. He has a BS in electrical engineering from National Taiwan University, and an MS and a PhD in electrical engineering from Stanford University. He is a member of the IEEE.

■ Direct questions and comments about this article to Yu Huang, Mentor Graphics, 300 Nickerson Rd., Marlborough, MA 01752; Yu_Huang@mentor.com.