BIST and Delay Fault Detection *

Slawomir Pilarski   Alicja Pierzyńska

School of Computing Science, Simon Fraser University
Burnaby, BC, Canada V5A 1S6

Abstract

We propose simple modifications to existing BIST schemes. These modifications significantly improve the path delay fault coverage. For example, a modified circular self-test path can detect a significant number of path delay faults within a reasonable test running time.

1 Introduction

The vast majority of Built-In Self-Test techniques aim at combinational faults, i.e., faults that can be always detected using a single test pattern. Little work, however, has been done on BIST for delay or stuck-open faults. To detect any of these faults a specific sequence of two test patterns has to be applied in two consecutive clock cycles [1, 2]. Unfortunately, simple LFSR-based structures do not detect such faults efficiently [3, 4].

A study of two-pattern test capabilities of autonomous test generators is presented in [3]. Test pattern generators for sequential faults have been discussed in [4, 5, 6]. In [5], the author suggests a high overhead BIST scheme generating a test sequence that covers a preselected set of test pairs. The structure presented in [6] generates all pairs of test patterns that differ at one binary position. In [4], the authors suggest that, to improve detectability of delay faults, the outputs of an LFSR or a cellular automaton should be permuted before they are connected to the inputs of a circuit under test, and they use the following permutation $(1, 3, 5, \ldots, 2, 4, 6 \ldots)$. The data presented in [4] shows some slight increase in delay fault coverage for gate delay faults [7].

It appears to us that, in general, the current work that analyzes delay fault detection by existing BIST structures focuses on maximizing the number of generated state transitions [2, 3, 4]. More precisely, the fraction of possible state transitions is of interest. The effectiveness of this strategy, however, seems to be limited to circuits with a relatively small number of inputs. For example, in a 40-input circuit, the fraction of state transitions is strongly limited by the length of test sequence that can be applied in practice. For this reason, we focus on "quality" of state transitions generated by BIST circuitry, rather than the actual numbers. Our considerations, based on a basic requirement for test application, suggest simple modifications to existing BIST techniques. We show that such modifications lead to a significant improvement in delay-fault coverage. In particular, we examine the path delay fault coverage [8] for patterns generated by a circular self-test path (CSTP) [9, 10, 11, 12, 13, 14, 15] and the path-delay fault coverage for test patterns generated by an LFSR. The experimental data that we present have been computed for ISCAS benchmark circuit c880 [16] using simulator introduced in [17]. The results demonstrate that a significant fraction of path delay faults can be detected by BIST circuitry.

2 Fault Model

In general, test procedures that aim at delay faults use circuit-dependent test generation algorithms, which are usually based on either the gate delay model [7, 18], or the path delay model [8]. Many of these algorithms assume further restrictions regarding hazards [19], robustness [20, 21, 22], size of faults [18, 23], actual gate delays [22, 24], etc. The variety of different restrictions reflects the complex nature of delay testing and makes it difficult to compare various test procedures in terms of test quality. Since the path fault model is considered to be more general than gate fault model [8, 22], in this paper, we report fault coverage

*This work was partially supported by grants from the Advanced Systems Institute of British Columbia and the Natural Sciences and Engineering Research Council of Canada.

0-7803-1429-8/93 $3.00 © 1993 IEEE

for the former model. We also present the fraction of all paths that are robustly tested [20, 21]. Such a fraction is often of a special interest, since a delay fault that affects a robustly tested path is detected regardless of the delays in other paths.

3 Idea for BIST Schemes

As already mentioned, in non-BIST solutions, to detect a delay fault, it is crucial to apply a specific pair of test patterns in two consecutive clock cycles. Unfortunately, a circuit that makes it possible may be expensive to implement. One solution is to organize all registers as a random access-memory. However, it may require a large address decoder, a data bus connected to every register, and expanded control logic; therefore, it may be unacceptable due to hardware overhead or performance degradation. Another, seemingly less expensive solution may be based on a scan-path. In this solution, however, two separate registers appear to be necessary to store a pair of test patterns in order to apply these patterns in two consecutive clock cycles. In general, such a solution requires twice as many flip-flops than a basic scan-path. To eliminate some of the extra flip-flops one may change the order of the flip-flops in a scan-path [2, 25]. If no two inputs in the same output cone are connected to adjacent flip-flops, two arbitrary test patterns can be applied to a circuit under test in two consecutive clock cycles [2]. Of course, there may be some penalties for changing the order of flip-flops. For example, the extra routing may consume substantial silicon area or degrade the circuit speed. Changes to the order of flip-flops may also complicate the test application schedule.

The discussion above implies that both test pattern generation and test pattern application for delay or stuck open faults are, in general, more expensive than test pattern generation and test pattern application for stuck-at faults. One could expect, therefore, that BIST that aims at delay faults may require more hardware overhead than BIST that aims at stuck-at faults. This expectation is confirmed by the fact that simple BIST structures based on LFSRs are not well suited to delay fault-testing [3, 4]. The reason for this is a strong correlation between two consecutive test patterns applied to circuits under test: some test sequences necessary to detect a fault cannot be generated. To see it more clearly, consider a group of adjacent flip-flops in an LFSR . These flip-flops constitute a shift register. The next state of this register is determined by its current state, the bit that is shifted in, and, in some cases, the most significant bit in the LFSR. Thus, there are at most only four possible transitions from the current state of the flip-flops.

A similar problem may occur when test patterns are generated by a CSTP. If a CSTP section (a group of adjacent flip-flops) does not compact test responses from a circuit under test (primary-input CSTP section), each state of this section section has exactly two different states that can occur in the next clock cycle (see Fig. 1).

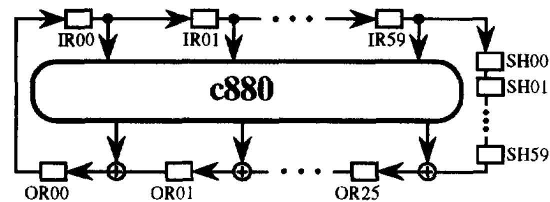

Figure 1: Primary-input CSTP.

The discussion on the test application and the discussion on the variety of test pairs produced by existing BIST schemes suggest the following: *a BIST structure that aims at delay faults should be able to produce all possible pairs of test patterns, even if not all such pairs are actually generated during the testing session.* Note that, for a circuit with a large number of inputs, the number of different test pairs produced by such a structure should be the very similar to the number of different test pairs produced by an LFSR. Nevertheless, the number of test pattern pairs produced at any small subset of the circuit inputs should increase. This should result in a more diversified collection of test pairs, and therefore, in an increased global path fault coverage.

Below, we present several relatively inexpensive modifications to existing BIST structures. These modifications are based on the idea that two consecutive test patterns should be independent. Our modifications significantly improve on the path delay-fault coverage offered by standard LFSR-based generators. We study two generators based on CSTP, and an LFSR structure. We compare these structures, in terms of the effectiveness of delay fault detection, with the original CSTP.

4 Double CSTP

The first of the proposed BIST circuits, referred to as a double CSTP (D-CSTP), consists of two circular self-test paths connected as presented in Fig. 2.

The two circular paths, both free of register adjacency [26, 12, 15], shift data in opposite directions; one of the paths compacts "test responses" from the other. The analysis of the CSTP presented in [15] implies that the state of register IR and the state of register SH (see Fig. 2) can be assumed independent. Since the states of register SH are all equally likely, the current IR state can be followed by any other state. A comparison of the number of path delay faults detected in benchmark circuit c880 by the double CSTP and the number of path delay faults detected by the original (primary-input) CSTP is presented in Fig. 3. The plots show a significant improvement in the total path-delay fault coverage. Moreover, the number of robust tests generated by the double CSTP is substantially larger than the number of robust tests generated by the original primary-input CSTP.

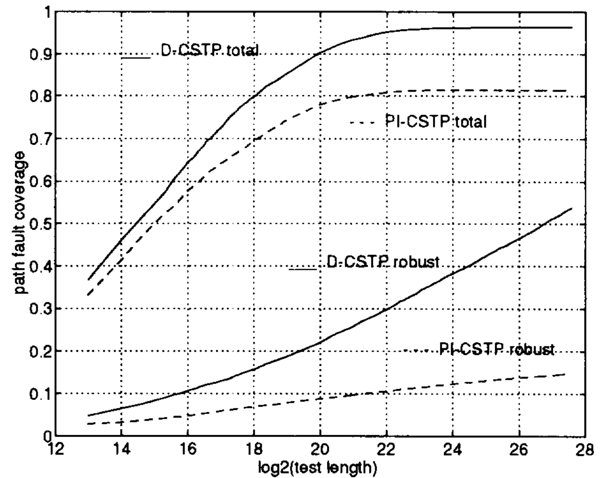

Figure 3: Fault Coverage of double CSTP (D-CSTP) and primary-input CSTP (PI-CSTP).

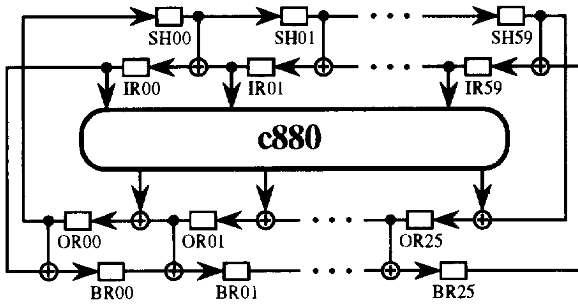

Figure 2: Double CSTP.

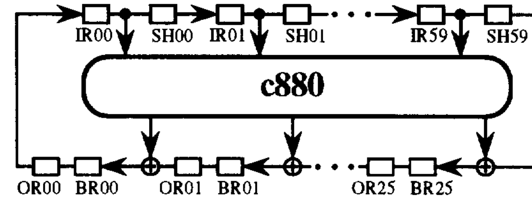

Figure 4: Double-flip-flop CSTP.

5 Double-Flip-Flop CSTP

A similar improvement in the path-delay fault coverage can be achieved if, in the original CSTP, every flip-flop is replaced with a two-bit shift register. This scheme, referred to as double-flip-flop CSTP (DF-CSTP), is presented in Fig. 4. Since the states of two adjacent CSTP flip-flops can be assumed independent [15], any two consecutive test patterns applied to c880 can be also assumed independent. It is interesting that the number of path delay faults detected in c880 by two-flip-flop CSTP and the number of path delay faults detected by double CSTP are practically indistinguishable (see Fig. 5). This observation applies not only to the total number of delay path tests, but also to the number of robust tests. It may suggest that the independence of two consecutive test patterns has more impact on the fault coverage than a particular structure of the test pattern generator.

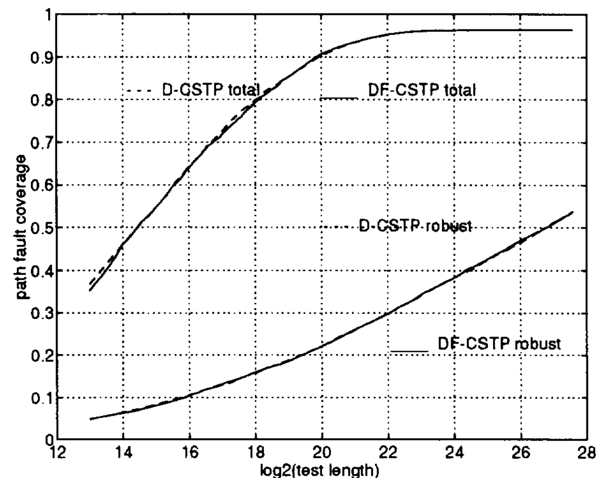

Figure 5: Fault Coverage of double-flip-flop CSTP (DF-CSTP) and double CSTP (D-CSTP).

6 Double-Length LFSR

The latest observation is confirmed by our experiments with an LFSR. Namely, we examined the number of path-delay faults detected by an LFSR characterized by a primitive polynomial of degree twice higher than the number of the CUT inputs. The inputs of c880 were connected to every second flip-flop in the LFSR (cf. [4]). This test generator, referred to as double-length LFSR (D-LFSR), detects as many path delay faults as the two previous structures. A comparison of the number of path delay faults detected by a double length LFSR and a single length LFSR is presented in Fig. 6. It is interesting to note that the number of delay faults detected by a single length LFSR and a primary-input CSTP are very similar. This confirms our observation that the ability to generate various state transition is more important that the actual test generator structure.

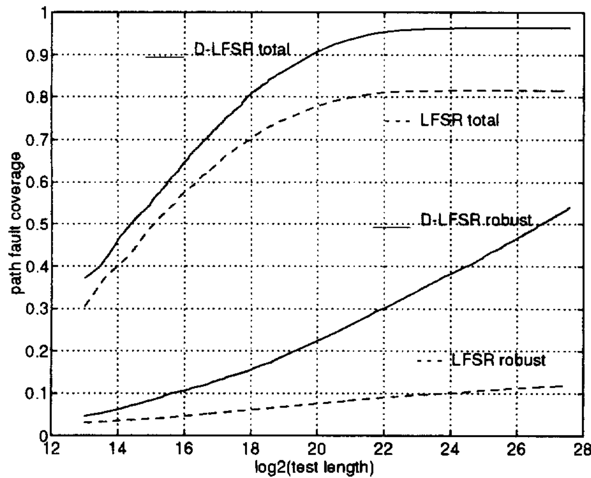

Figure 6: Fault coverage of double-length LFSR (D-LFSR) and LFSR.

7 Stimulated CSTP

The state sequence of a CSTP section that compacts test responses from a circuit under test (stimulated CSTP section) usually has many more possible state transitions than the state sequence of a primary-input CSTP section. This is because, for the same current state, compacting two different test responses usually results in two different states of the CSTP section. Thus, each test pattern produced by a CSTP section can be followed by at least as many patterns as there is different test responses from the CUT. The number of delay faults detected by the stimulated CSTP should be larger than the number of delay faults detected by the primary-input CSTP. To verify this conjecture, we examine the path delay fault coverage in c880 for test patterns generated by a stimulated CSTP section (see Fig. 7). It turns out that in this example the fault coverage practically does not differ from that attained by double-flip-flop-CSTP (see Fig. 8). This fact is probably due to a large variety of test responses produced by benchmark circuits c499 [16].

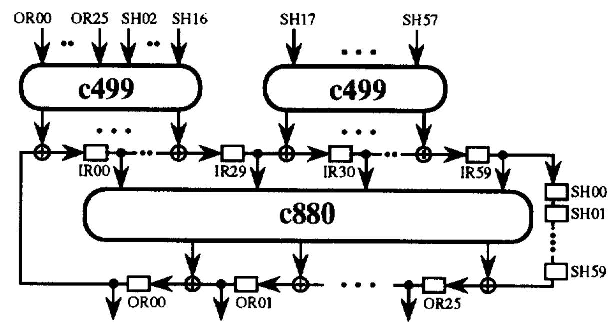

Figure 7: Stimulated CSTP.

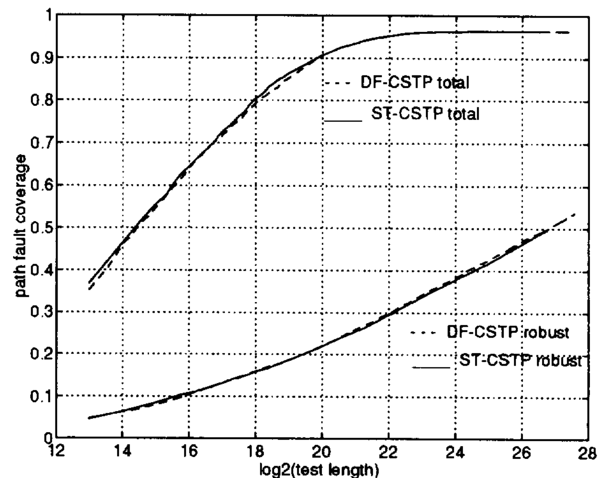

Figure 8: Fault coverage of stimulated CSTP (ST-CSTP) and double-flip-flop CSTP (DF-CSTP).

8 Relative Frequency of Test Pairs

To investigate the variety of state transitions in the proposed structures, we observed the relative frequency of consecutive states that occur at a 4-bit section of the test generators. In all double structures, after a sufficiently long time, all possible pairs of 4-bit vectors occurred with approximately the same frequency. The results of our experiments are summarized in Fig. 9, where the standard deviation of the relative frequency of 4-bit vector pairs is presented. One may conjecture that the similarity in the distributions of vector pairs is related to the similarity in the corresponding fault coverages. Such a conjecture is also confirmed by the fact that the distributions of 4-bit vector pairs in a single LFSR and in a primary-input CSTP look alike (recall that delay fault coverages for these two structures are almost identical).

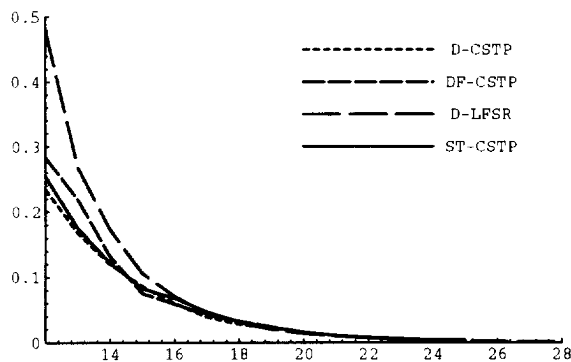

Figure 9: Standard deviation of the relative frequency of 4-bit vector pairs as a function of $\log_2$(test length).

9 Conclusions

The discussion presented in this paper and our simulation results for benchmark circuit c880 suggest the following:

- For a particular circuit, the path delay fault coverage strongly depends on the ability of the generator to produce pairs of uncorrelated test patterns; this ability appears to be more important than the actual generator structure.
- For a given test sequence length, BIST structures that can produce all pairs of consecutive test patterns detect significantly more path delay faults than earlier BIST schemes that aim at combinational faults. Moreover, the former generate significantly more robust tests than the latter.
- Since BIST structures that can produce all pairs of test patterns repeat the same patterns many times, to improve delay fault coverage, traditional exhaustive BIST techniques should be replaced with random BIST[1], e.g., circular self-test path. (Note that such a replacement has the potential to improve the overall test quality. This is because a CSTP offers nearly exhaustive testing if the test session length is 8 times longer than that required by an LFSR with a primitive characteristic polynomial [13, 15].)

Since random BIST techniques appear to offer a better overall test quality, and CSTP seems to be one of the least expensive, we would like to point out the following:

- A simple and relatively inexpensive modification to CSTP may significantly improve the path delay fault coverage.
- A modified CSTP offers much higher quality of delay testing than other BIST techniques targeting combinational faults.

It is worth stressing that the silicon area penalty for the CSTP modifications may be relatively low:

- Stimulated CSTP sections may not require modifications (see Section 7), and
- The independence of consecutive vectors can be obtained not only by adding flip-flops, but also by selecting the proper order of flip-flops in the path (as suggested in [2]).

Acknowledgements

We would like to acknowledge the contribution of Yuejian Wu, who performed the simulation experiments described in this paper. We are also greatful to Prof. Andre Ivanov for valuable discussions.

References

[1] Z. Barzilai and B. K. Rosen, "Comparison of AC self-testing procedures," in *Proc. International Test Conference*, pp. 89–94, 1983.

[1] By "random BIST" we mean techniques in which a test pattern may occur in many different subsequences of test patterns generated during a testing session.

[2] J. Savir and R. Berry, "At-speed test is not necessarily an AC test," in *Proc. International Test Conference*, pp. 722–728, 1991.

[3] K. Furuya and E. J. McCluskey, "Two-pattern test capabilities of autonomous TPG circuits," in *Proc. International Test Conference*, pp. 704–711, 1991.

[4] S. Zhang, R. Byrne, and D. M. Miller, "BIST generators for sequential faults," in *Proc. IEEE International Conference on Computer Design*, pp. 260–263, 1992.

[5] C. W. Starke, "Built-in test for CMOS circuits," in *Proc. International Test Conference*, pp. 309–314, 1984.

[6] G. L. Criag and C. R. Kime, "Pseudo-exhausive adjacency testing: a BIST approach for stuck-open faults," in *Proc. International Test Conference*, pp. 126–137, 1985.

[7] A. K. Pramanick and S. M. Reddy, "On detection of delay faults," in *Proc. International Test Conference*, pp. 845–856, 1988.

[8] G. L. Smith, "Model for delay faults based upon paths," in *Proc. International Test Conference*, pp. 342–349, 1985.

[9] A. Krasniewski and S. Pilarski, "Circular self-test path: A low cost BIST technique," in *Proc. ACM/IEEE Design Automation Conference*, pp. 407–415, June 1987.

[10] C. E. Stroud, "An automated BIST approach for general sequential logic synthesis," in *Proc. ACM/IEEE Design Automation Conference*, pp. 3–8, September 1988.

[11] M. M. Pradhan, E. J. O'Brien, S. L. Lam, and J. Beausang, "Circular BIST with partial scan," in *Proc. International Test Conference*, pp. 719–729, September 1988.

[12] C. E. Stroud, "Automated BIST for sequential logic synthesis," *IEEE Design and Test of Computers*, vol. 5, pp. 22–32, December 1988.

[13] A. Krasniewski and S. Pilarski, "Circular self-test path: A low cost BIST technique for VLSI circuits," *IEEE Trans. CAD*, vol. 8, pp. 46–55, January 1989.

[14] M. Abramovici, M. A. Breuer, and A. D. Friedman, *Digital Systems Testing and Testable Design*. New York: Computer Science Press, 1990.

[15] S. Pilarski, A. Krasniewski, and T. Kameda, "Estimating testing effectiveness of the circular self-test path technique," *IEEE Trans. CAD*, vol. 11, pp. 1301–1316, October 1992.

[16] F. Brglez and H. Fujiwara, "A neutral netlist of 10 combinational benchmark circuits and a target translator in FORTRAN." Special Session on Recent Algorithms for Gate-Level ATPG with Fault Simulation and Their Performance Assessment, *IEEE Int. Symp. Circuits and Systems*, June 1985.

[17] Y. Wu and A. Ivanov, "Accelerated path delay fault simulation," in *IEEE VLSI Test Symposium*, pp. 1–6, April 1992.

[18] V. S. Iyengar, B. K. Rosen, and J. A. Waicukauski, "On computing the sizes of detected delay faults," *IEEE Trans. CAD*, vol. 9, pp. 299–312, March 1990.

[19] P. C. McGeer, A. Saldanha, P. R. Stephan, R. K. Brayton, and A. L. Sangiovanni-Vincentelli, "Timing analysis and delay-fault test generation using path-recursive functions," in *Proc. International Conference on Computer-Aided Design*, pp. 180–183, 1991.

[20] C. J. Lin and S. M. Reddy, "On delay fault testing in logic circuits," in *Proc. International Conference on Computer-Aided Design*, pp. 148–151, 1986.

[21] C. J. Lin and S. M. Reddy, "On delay fault testing in logic circuits," *IEEE Trans. CAD*, vol. 6, pp. 694–703, September 1987.

[22] S. Patil and S. Reddy, "A test generation system for path delay faults," in *Proc. IEEE International Conference on Computer Design*, pp. 40–43, 1989.

[23] Y. Aizenbud, M. Leibowitz, P. Chang, D. Smith, B. Koenemann, V. Iyengar, and B. Rosen, "AC test quality: Beyond transition fault coverage," in *Proc. International Test Conference*, pp. 568–577, 1992.

[24] E. S. Park and M. R. Mercer, "An efficient delay test generation system for combinational logic circuits," *IEEE Trans. CAD*, vol. 11, pp. 926–938, July 1992.

[25] W. Mao and M. D. Ciletti, "Correlation-reduced scan-path design to improve delay fault coverage," in *Proc. ACM/IEEE Design Automation Conference*, pp. 73–79, 1991.

[26] C. Hudson and G. Peterson, "Parallel self-test with pseudo-random test patterns," in *Proc. International Test Conference*, pp. 954–963, September 1987.