ISSN: 2278 – 909X
International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)
Volume 1, Issue 6, December 2012

A Method to Detect Uncovered Faults for Benchmark Circuits

Ch.Vinaykumar(Student,M.Tech.(VLSI),Dept.of ECE,Aditya Institut of Technology andManagment, Tekkali)
A.S.Srinivasa Rao(Professor and HOD,Dept of ECE,Aditya Institute of Technology and Management)

ABSTRACT: We demonstrate that undetectable single stuck-at faults in full-scan bench mark circuits tend to cluster in certain areas. This implies that certain areas may remain uncovered by a test set for single stuck-at faults. We describe an extension to the set of target faults aimed at providing a better coverage of the circuit in the presence of undetectable single stuck-at faults. The extended set of target faults consists of double stuck-at faults that include an undetectable fault as one of their components. The other component is a detectable fault adjacent to the undetectable fault. We present experimental results of fault simulation and test generation for the extended set of target faults.

Key Words : Full-scan, stuck-at faults, test generation, test quality, undetectable faults.

INTRODUCTION

Fault simulation plays a significant role in the testing of digital circuits. In fault simulation we use a fault model which models physical faults that may occur in actual circuits. The most commonly used model is a single stuck at fault model which has two faults per line, stuck-at-1 (s-a-1), stuck.at-0 (s-a-0). The basic assumptions that characterize single stuck-at fault model are:

1. Only one line is faulty
2. The faulty line is permanently set to either 0 or 1.
3. The fault can be at an input or output of a gate.

Unlike a single stuck at fault model, a multiple fault model represents a condition caused by the presence of a group of single faults. Research has shown that a large percentage of multiple faults is covered by the single-fault tests, specific design styles and test generation procedures assure the detection of most of the multiple stuck-at faults if the single faults are detected.

However the effects of multiple faults cannot be underestimated in VLSI circuits. If the number of single stuck at faults is O (n) for a circuit size n, the number of multiple faults is O(2m o(n)Cm) for a fault multitude m. The complexity of multiple fault simulation increases with the increase in the number of faults being handled together. But because of the increasing size of VLSI circuits it is important to analyze the properties and characteristics of the multiple faults in terms of both simulation and test generation. Some of the applications of Multiple fault analysis include:

1. improved circuit optimization.
2. Improved fault diagnosis
3. Better fault coverage

In this how an arbitrary multiple stuck-at fault in combinational or sequential circuits can be converted in to a logic level single stuck-at fault model which allows effective use of existing tools. In this how Test sets that contain several different tests for each fault (n-detection test sets) are expected to increase the likelihood of detecting defects associated with the sites of target faults.

MULTIPLE FAULT MODEL

A multiple stuck at fault of multiplicity n can be modeled as a single stuck at fault by using at most n+3 gates. Consider the circuit in the Figure 1. To convert the multiple stuck at fault in to a single stuck at fault model the following steps have to be performed:

1. A two input gate is inserted in each faulty line. An AND gate is inserted in a line with stuck at 0 fault and an OR gate is inserted in a line with a stuck at 1fault. These are called In-line gates.

2. The second input of the In-line gates is fed by an n input AND gate. The output of this AND gate feeds the In-line OR gates directly and feeds the In-line AND gates through an inverter. The n inputs to this gate are derived directly from the s-a-1 fault lines and through inversion from s-a-0 lines.

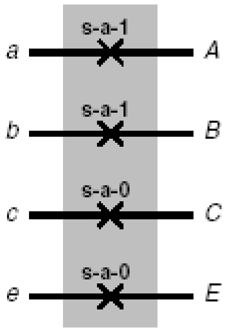


Fig1: Multiple stuck-at faults

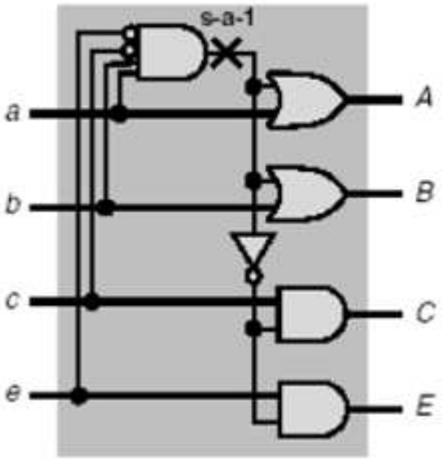


Fig2: An equivalent single stuck at fault

Through this conversion when a fault is not activated the output of the In-gate circuit is same as good circuit value. Hence circuit equivalence is preserved. This can be verified using Boolean algebra: In the modified circuit,

A= a+ab(ce)'=a
C=c(ab(ce)')'= c

which are same as fault free circuit. Similarly it can be proved for other signal lines also.

The multiple stuck at faults in Figure 1(a) should be same as single stuck at faults in Figure1 (b). By observing the signal values of lines A, B, C, D in both the models it can be verified.

1. Untraceable distinct Stuck-At Fault

Generation of effective tests for detecting defects is a challenge which can be addressed by stuck-at fault models which can act as targets for generating test sets. Defects in the periphery of a stuck –at fault can be exposed by executing a test set for a single stuck–at fault at that site. The probability of detecting defects in the vicinity of target faults increases when we execute unique tests for each fault. Uncovered sites in a circuit exist when a single stuck-at fault remains undetectable. Single stuck-at faults tend to bunch in certain areas in benchmark circuits, which means that single stuck-at test sets make some areas of the circuit uncovered or partially covered. This behavior can be observed from the gate level description of the circuit and it is independent of the layout parameters.

During the process of logic synthesis undetectable single stuck-at faults are introduced, the method proposed in this paper will be valid, independent of the layout of the circuit since the basis of clustering is at the gate level. In order to improve the predicted fault coverage of the circuit under consideration we shall consider a double stuck-at fault which includes an undetectable fault as one of their component. A detectable fault adjacent to the detectable fault will the other component. There is two fold incentive in including such double faults.

1) The type of double stuck-at fault that we are discussing here consists of an undetectable stuck-at fault f1 and an adjacent detectable stuck-at fault, the detection of such double stuck-at fault, gives indirect coverage for the region near f1. The double stuck at fault makes it possible for f1 to be detected, although it was not detectable when it was alone, it is detected when another adjacent fault is present. The defect coverage around the site of f1 is improved. In manufacturing tests a large test set is required for detecting bridging faults, for bridging faults occurring in areas of the circuit with undetectable faults, the probability of fault detection can be improved by introducing tests for double stuck-at fault as described in this paper.

2) If a fault fj occurs in a circuit and tests that detect it is also available, then if that circuit also has an undetectable fault fi, the test previously mentioned may not detect the double fault consisting of fi and fj. When the double fault is considered explicitly it is possible to ensure that fj will be detected when fi is present. It is a well known fact that test sets that detect single stuck-at faults detect also detect a good number of multiple stuck-at faults. Therefore the requirement to detect double stuck-at fault based on single stuck-at fault does not add to any overhead to the test set used for single stuck-at fault. The additional tests can be crucial in covering defects in the areas of the circuit with undetectable single stuck-at faults.

It should be noted that a test which uncovers a detectable fault fj is not guaranteed to detect a double fault consisting of fj and another undetectable fault fi unless fj is an adjacent detectable fault. The thesis is organized as follows.

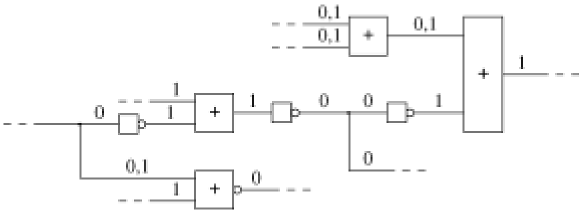
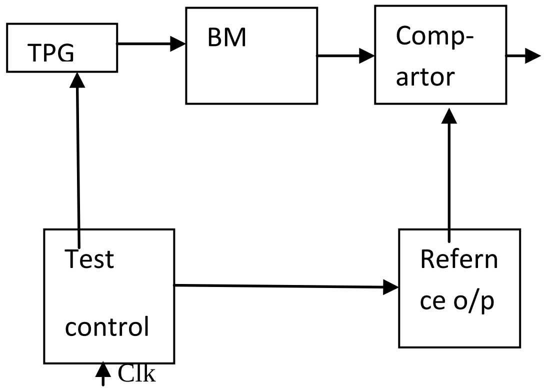


Fig. 3. Example sub-circuit 1.

It can be shown that clustering of undetectable single stuck-at faults occurs in certain areas for benchmark circuits. Faults that include double stuck-at faults based on undetectable single stuck-at faults are defined as the extended set of target faults. The fault simulation and test generation that we propose target the extended set of faults. The results of the fault simulation and test generation will be shown after suitable experiments are conducted. We present additional experimental results and discuss the case where information about detectable and undetectable faults is not available.

ARCHITECTURE



Architecture of testing circuit of bench Mark circuit

The essential Testing circuitry, shown in Figure 1 typically includes a test pattern generator (TPG), reference output analyzer, and test controller. The TPG

BENCH MARK CIRCUIT(S27)

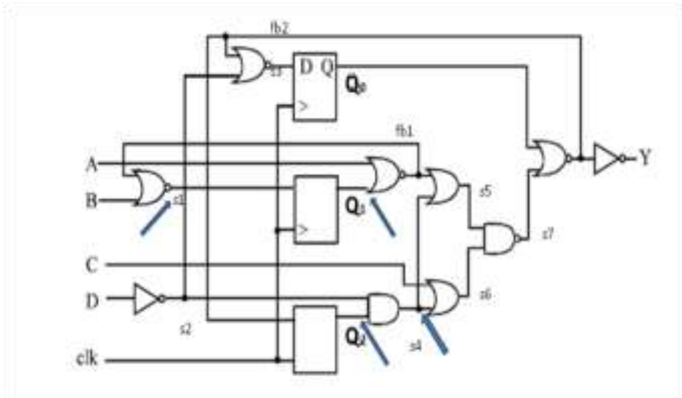


Fig7 : s27 Bench mark circuit

LFSR

The LFSRs represent the simplest and most commonly used pseudo-random TPG hardware, the efficiency of an LFSR is far from optimum in terms of fault coverage and testing runtimes. The test generated by an LFSR is usually up to several orders of magnitude longer than a similar ATPG test. In general, PRPG fault coverage trend is characterized by such peculiarities like fast initial growth and too long time to complete. Figure8 illustrates this fact clearly showing the corresponding sections of the curve. The slow growth section is mostly caused by existence of pseudo-random pattern resistant faults a.k.a. hard-to-test faults (HTTF) which are usually very difficult to handle by PRPG-based methods. In case of large numbers of HTTFs in the DUT, the maximum fault coverage cannot be achieved by LFSR sequence in realistic time at all. Due to this difficulty, as well as great importance of testing circuitry, there is a huge amount of works that target improvement of PRPG efficiency. Each of these works has certain advantages and disadvantages over the others.

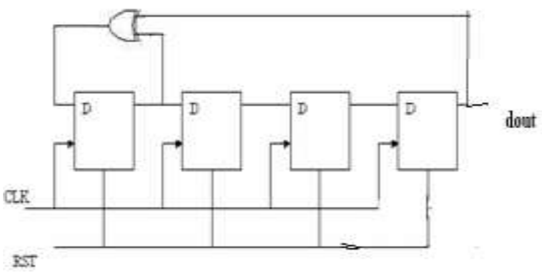


Fig 8 Block Diagram of Pn-sequence generator

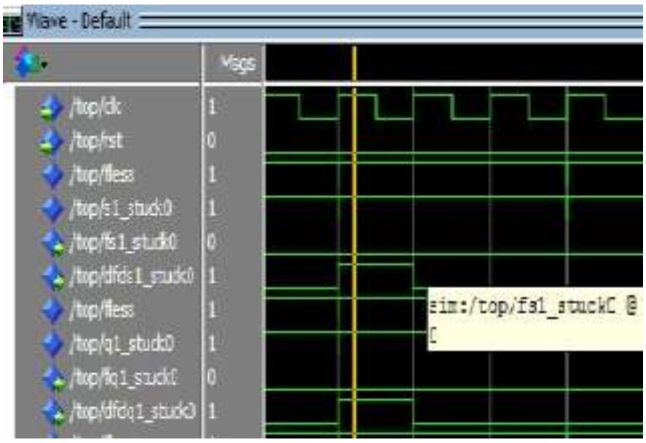
A big portion of research on testing circuitry is devoted to study of alternative PRPG types that have better saturation properties compared to the one of LFSR due to better fulfilling randomness criteria for generated sequences.These are, for instance, Cellular Automata and GLFSR. However, the randomness has only been empirically proven to improve the quality of testing. On the contrary, larger designs, especially those that contain random pattern resistant HTTFs, need special treatment. As the result, mixed-mode methods that combine PRPG and ATPG patterns gained a major attention.

From the implementation point of view, the available methods can be classified into two major categories accordingly to the way they handle ATPG patterns: a) memory-based methods and b)special encoding and embedding hardware based methods

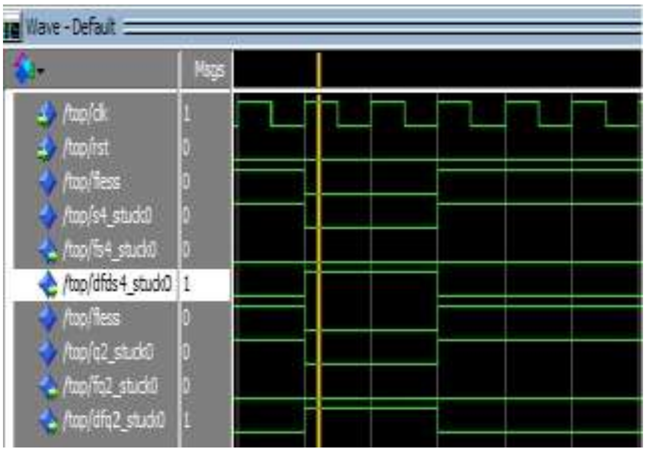
RESULTS:

circuit	Testset	tests	% of coverage
S1423 S1423	T1 T2	26 30	83.53 85.68
S5378 S5378	T1 T2	100 101	89.76 89.90
S9234 S9234	T1 T2	111 113	81.09 81.53
S13207 S13207	T1 T2	235 236	87.97 89.08
S15850 S15850	T1 T2	87 88	92.20 92.41
S38584 S38584	T1 T2	142 155	84.64 85.15

Simulation Results



Simulation results for s1 and q1



Simulation results for s4 and q2

CONCLUSION

We showed that in full scan benchmark circuits, undetectable single stuck at faults(T1) tend to cluster in certain areas. In order to provide a target for an extended set of faults we introduced double stuck-at fault(T2) to improve the coverage of these areas. Based on undetectable faults and detectable faults that are adjacent to them we defined the double stuck – at faults. Obtained better results for double stuck at faults method (T2)compared to single stuck at fault targets(T1) We showed through experiments consisting of test generation and fault simulation that the coverage areas with undetectable faults can be extended.

References

[1] M. Abramovici, M. A. Breuer, and A. D. Friedman, “Testing for single stuck faults,” *Digital Systems Testing and Testable Design*. Piscataway, NJ: IEEE, 1995, ch. 6, pp. 181–281.

[2] P. Goel and B. C. Rosales, “Test generation and dynamic compaction of tests,” in *Proc. Test Conf.*, 1979, pp. 189–192.

[3] I. Pomeranz, L. N. Reddy, and S. M. Reddy, “COMPACTEST: A method to generate compact test sets for combinational circuits,” in *Proc. Int. Test Conf.*, Oct. 1991, pp. 194–203.

[4] J.-S. Chang and C.-S. Lin, “Test set compaction for combinational circuits,” in *Proc. Asian Test Symp.*, 1992, pp. 20–25.

[5] Y. Matsunaga, “MINT: An exact algorithm for finding minimum test sets,” *IEICE Trans. Fundam.*, vol. E76-A, no. 10, pp. 1652–1658, Oct. 1993.

[6] S. Kajihara, I. Pomeranz, K. Kinoshita, and S. M. Reddy, “Cost-effective generation of minimal test sets for stuck-at faults in combinational logic circuits,” *IEEE Trans. Comput.-Aided Design*, vol. 14, no. 12, pp. 1496–1504, Dec. 1995.

[7] I. Hamazaoglu and J. H. Patel, “Test set compaction algorithms for combinational circuits,” in *Proc. Int. Conf. Comput.-Aided Design*, 1998, pp. 283–289.

[8] S. C. Ma, P. Franco, and E. J. McCluskey, “An experimental chip to evaluate test techniques experiment results,” in *Proc. Int. Test Conf.*, 1995, pp. 663–672.

[9] S. M. Reddy, I. Pomeranz, and S. Kajihara, “Compact test sets for high defect coverage,” *IEEE Trans. Comput.-Aided Design*, vol. 16, no. 8, pp. 923–930, Aug. 1997.

[10] J. T.-Y. Chang, C.-W. Tseng, C.-M. J. Li, M. Purtell, and E. J. McCluskey, “Analysis of pattern-dependent and timing-dependent failures in an experimental test chip,” in *Proc. Int. Test Conf.*, 1998, pp. 184–193.

[11] J. Dworak, M. R. Grimaila, S. Lee, L.-C. Wang, and M. R. Mercer, “Enhanced DO-RE-ME based defect level prediction using defect site aggregation-MPG-D,” in *Proc. Int. Test Conf.*, 2000, pp. 930–939.

[12] I. Pomeranz and S. M. Reddy, “Definitions of the numbers of detections of target faults and their effectiveness in guiding test generation for high defect coverage,” in *Proc. Conf. Design Autom. Test Eur.*, Munich,