Test Pattern Minimization Using Redundancy for Sequential Circuits

[1]Charliene Karunya. L, [2]Divya Bharathy. K, [3] Sheela Narayanan

[1,2] UG STUDENT(B.E., ECE),Sri Ramakrishna Engineering College, Coimbatore, India.
[3] ASSITANT PROFESSOR, Department of Electronics and Communication Engineering, Sri Ramakrishna Engineering College, Coimbatore, India.

Abstract--- **Detecting the faults in the digital circuit is a hard process. Hence, detecting a fault more than one time has been shown to result in high defect coverage. Previous works shows that n-detect test sets are of increased quality for a number of common defects. Method for multiple detect test generation usually produce fully specified test vectors. This becomes their disadvantage in most of the low power test applications. The paper proposes a step by step methodology for finding out a large number of bits that can be unspecified in a n-detect test set, while maintaining actual fault coverage. The experimental output shows that the number of specified bits can be gradually reduced. Subsequently, the size of the test set is reduced.**

Index Terms--- **ATPG, concatenation, LFSR, pseudo-random sequence, relaxation.**

I. INTRODUCTION

All manufactured VLSI chips are tested for defects. But it is not possible to generate or apply vectors to test all possible defects in a chip. So defects are modeled as faults to ease the test generation process. Among the various fault models proposed the single stuck-at fault model is widely accepted because of its closeness to actual defects and the algorithm impossibilities it offers for generating test vectors. However the smaller levels are desired for devices in most applications better fault models are needed which can accurately model the defects. Such fault models tend to be complex making test generation harder or even impossible. Therefore a practical idea that seems to work is to use the single stuck-at fault model and increase the probability of detecting unmodeled defects by increasing the number of times each single stuck-at fault is detected during a test. The combination of the large number of possible defect types together with the huge number of fault sites in a modern circuit implies that modeling these defects will give prohibitively large input for a systematic test generation methodology. For n-detect test generation and compaction produce tests that are full specified (i.e., all the test set bits have a fixed value of 0 or 1). This occurs since many of these techniques try to fix unspecified (don't care) bits to logic values such that the number of detected faults is increased. The method starts with an initial (given) test set which can be fully or partially specified. The total number of specified bits in the resulting test set is minimized, while maintaining its original -detect fault coverage. Furthermore, the test set size is guaranteed not to increase; actually, it is often decreased.

II. SHIFT REGISTER

In computing, a L**inear-Feedback Shift Register** (LFSR) is a shift register whose input bit is a linear function of its previous state. The most commonly used linear function of single bits is XOR. Thus, an LFSR is most often a shift register whose input bit is driven by the exclusive-or (XOR) of some bits of the overall shift register value. The initial value of the LFSR is called the seed, and because the operation of the register is deterministic, the stream of values produced by the register is completely determined by its current (or previous) state. Likewise, because the register has a finite number of possible states, it must eventually enter a repeating cycle. However, an LFSR with a well-chosen feedback function can produce a sequence of bits which appears random and which has a very long cycle. LFSRs are used in circuit testing, for test pattern generation

III. STUCK- AT FAULT

A **stuck-at fault** is a particular fault model used by fault simulators and automatic test pattern generation (ATPG) tools to mimic a manufacturing defect within an integrated circuit. Individual signals and pins are assumed to be *stuck* at Logical '1', '0' and 'X'. For example, an output is tied to a logical 1 state during test generation to assure that a manufacturing defect with that type of behavior can be found with a specific test pattern. Likewise the output could be tied to a logical 0 to model the behavior of a defective circuit that cannot switch its output pin. Not all faults can be analyzed using the stuck-at fault model. Compensation for static hazards, namely branching signals, can render a circuit untestable using this model. Also, redundant circuits cannot be tested using this model, since by design there is no change in any output as a result of a single fault.

ISSN: 2278 – 7798

IV. METHODOLOGY

Systematic methodology for identifying a large number of bits that can be unspecified in a multiple detect test set preserving the original fault coverage. The number of specified bits in even compact n-detect test sets can be significantly reduced without any impact on the n-detect property. Integer linear programming is used to find the smallest set. Additionally in many cases the size of the test set is reduced test set relaxation does not imply that the specified bits of the relaxed test set are a subset of the specified bits of the initial test set as it is the case with the existing relaxation methods. Rather relaxation refers to the process of increasing the total number of unspecified bits in order to make the test set more "flexible" for other applications. A novel systematic test replacement algorithm is proposed in which each test is replaced by a new one that detects a subset of the faults detected by the first one with fewer specified bits. In order to maintain the fault coverage each fault is guaranteed to be detected at least times where this is possible. The algorithm explicitly removes additional detections for each fault. The latter is possible since experimentation shows that in n-detect test sets the average detections for each fault is much greater than mainly due to the presence of many easy-to-detect faults. Specifically the methodology targets optimization problem it determines the most appropriate tests to detect a fault that give the maximum benefit in terms of specified bits savings in the entire test set. Thus it selects the "best" tests to detect the fault and drops the fault from the remaining tests in order to reduce the total number of specified bits in these tests. The obtained results indicate that the new method is very successful in reducing the total number of specified bits with often a decrease to the final test set size. A simple X-filling method for low power testing has been used in order to demonstrate how relaxed test sets can benefit low power testing

A. Fault Modelling

As it is not possible to enumerate all possible physical defects and develop tests for them the defects are modelled as faults. These abstract fault models emulate the behaviour of physical defects while simplifying the test generation process. Of the various fault models the single line stuck-at fault model is widely accepted because of its closeness to actual defects and the simplicity it allows in generating test vectors. Various efficient algorithms have been developed and programmed to efficiently generate tests for single stuck-at faults.

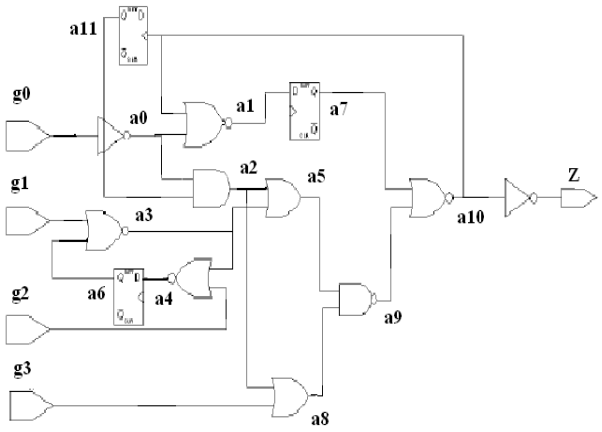
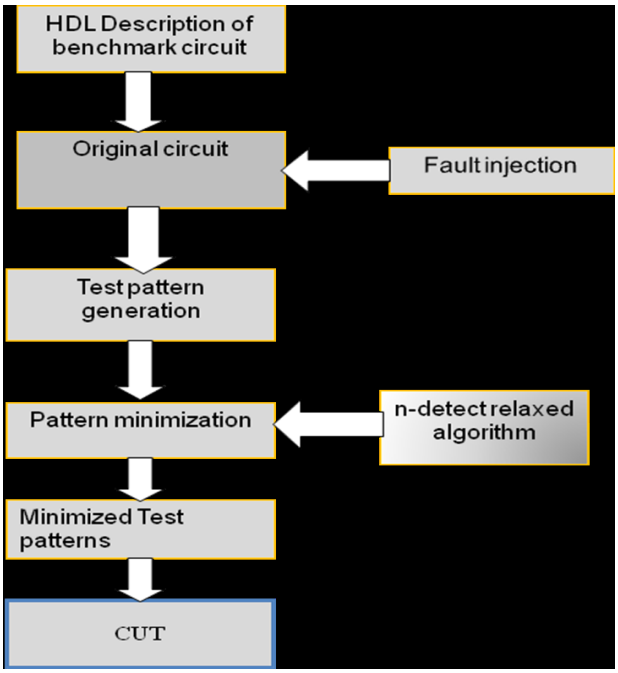


Fig 1. S27 A example circuit for testing

Fault Injection is a dependability validation technique based on the realization of the controlled experiments where the observation of the system behaviour in present of faults is explicitly induced by the deliberate introduction of faults into the system. Fig 1. is an example circuit, where both single and double faults are applied in few places. The LFSR which generates the test patterns find out these faults by comparing these generated patterns with the original circuit output before fault injection. Thus by using this concept we are able to describe the faults correlated to a component. These of the possible faults known for the component can therefore be described by a separated test pattern. For the conclusion that the mean time between faults and their duration should be an integral part of the description of each behavioural component of the model. In order to demonstrate the feasibility of this approach, we developed the VERIFY tool which allows the integrated description of the fault free behaviour as well as the component's behaviour after one of the faults correlated with this component has been activated. After the faults their parameters and their behaviour have been described for the behavioural components of the VHDL-model the simulator can inject the faults during simulation time. There are two basic alternatives to make these fault injection signals and their parameters visible for the simulator is include the signals in the entity declaration or keeping the FIS transparent to other components.

ISSN: 2278 – 7798

B. FLOW DIAGRAM



C. Pattern Generation

The proposed procedure is applied to the hard-to-detect faults remaining after a pseudo-random sequence of length and minimizes the storage requirements for the resulting deterministic test set. The proposed algorithm supports the concatenation of patterns and requires as input parameters the maximum number of patterns and the maximum number of specified bits accepted in a group of patterns to be concatenated. The basic idea to achieve an efficient encoding is to alternate test pattern generation and the encoding of test patterns. Test patterns are generated until the limits for a group of patterns to be concatenated are reached. All patterns in this group are concatenated to one pattern which is encoded as a seed for one of the available feedback polynomials and the first bits of the LFSR sequence resulting from this seed are determined. Since this subsequence corresponds to the fully specified patterns fed into the scan chain during test mode it is fault simulated against the remaining fault set. Faults which are detected in addition to the original target faults can be dropped immediately and the process is repeated.

D. n-Detect Test Set

An n-detect test set is a set that detects each stuck-at fault with at least n "different" test vectors. The more uniquely different the test vectors for a fault the better may be the defect coverage but harder will be the test generation. There is no general agreement on how the vector "difference" should be defined. The main problem that limits the use of n-detect tests is their size and there is need to minimize them. Existing methodologies for n-detect test generation and compaction produce tests that are fully specified. This occurs since many of these techniques try to fix unspecified bits to logic values such that the number of detected faults is increased. Actually even if bit fixing does not improve on the n-detect fault coverage it can improve on the coverage of non-targeted faults and defects even by randomly fixing the unspecified bits. As a result many existing test generation tools return fully specified test sets. The limits applicability of n–detect test sets in several currently important problems. For instance methods for compression schemes for on-chip or off-chip test set embedding as well as in compact test generation for high defect coverage can benefit when relaxed test sets are used. Such flexible test sets are also extremely crucial in low power test generation techniques like those in among many others where important power reduction may be obtained when appropriately fixing the unspecified bits. The work in this paper considers the problem of relaxing an n-detect test set. The proposed method also applies to multiple detect test set without any loss of generality present it here only for n-detect test sets. Whenever necessary elaborate on the trivial modifications that must be made for multiple detect test sets. The method starts with an initial test set which can be fully or partially specified. The total number of specified bits in the resulting test set is minimized while maintaining its original n-detect fault coverage. Furthermore the test set size is guaranteed not to increase actually it is often decreased. The motivation behind this problem is that a test bit needs to be initially fixed only if this helps the n-detect fault coverage otherwise it can be left unspecified. The generated relaxed test set can then be used in a variety of applications that fix the unspecified bits Relaxation of 1-detect test sets was studied in which proposed methods relying on various Automatic Test Pattern Generation (ATPG) concepts in order to identify specified bits in the test set that can be replaced by don't care values proposed a method for identifying don't care bits in a test pattern using ATPG concepts such as implication and justification used a similar rationale taking into consideration testability measures in the justification process. Extending these methods to n-detect test sets is not straightforward. Actually all of these methods benefit from identifying essential tests which do not exist in n-detect test sets. The recent method of also consider test set relaxation but in a dynamic manner. They do not consider an initial test set to be relaxed rather the ATPG process is restricted to consider the number of specified bits in the generated compact test set. As a result it cannot be easily extended to n-detect test sets especially for large values of the recent work of proposes a new static technique for test relaxation under the physically-aware n-detect model utilizing some existing concepts for 1-detect and n-detect test set relaxation. While the extension of both the static and dynamic methods to n-detect test sets could be investigated, in this work propose a new methodology that is optimized based on the characteristics and parameters of the n-detect test sets. Test set relaxation does not imply that the specified bits of the relaxed test set are a subset of the specified bits of

ISSN: 2278 – 7798

the initial test set as it is the case with the existing relaxation Rather relaxation refers to the process of increasing the total number of unspecified bits in order to make the test set more "flexible" for other applications. A novel systematic test replacement algorithm is proposed in which each test is replaced by a new one that detects a subset of the faults detected by the first one with fewer specified bits. In order to maintain the fault coverage each fault is guaranteed to be detected at least times where this is possible. The algorithm explicitly removes additional detections for each fault. The latter is possible since experimentation shows that in n-detect test sets the average detections for each fault is much greater than mainly due to the presence of many easy-to-detect faults. Specifically the methodology targets an optimization problem it determines the most appropriate tests to detect a fault all of that give the maximum benefit in terms of specified bits savings in the entire test set. Thus it selects the "best" tests to detect the fault and drops the fault from the remaining tests in order to reduce the total number of specified bits in these tests. The obtained results indicate that the new method is very successful in reducing the total number of specified bits with often a decrease to the final test set size. A simple X-filling method for low power testing has been used in order to demonstrate how relaxed test sets can benefit low power testing. The rest of this paper is organized as follow elaborates on our motivation and presents supportive data gives necessary notation and the problem formulation and Section describes the proposed technique. A comprehensive example illustrating the proposed method's execution is presented gives the obtained experimental results for the specified bits reduction as well as the impact of the proposed method in X-filling for low power test. This experimentation was done using the popular stuck-at model however fault models can be used such as the transition fault model.

E. Pattern Minimization

Several patterns detect the same faults. This increase the complexity of the patterns. To avoid this and to enhance the process of detection of faults, it is necessary for us to go in for *"PATTERN MINIZATION"*. The term implies that we compare the patterns with one another which detects the same fault, and if there is a one bit change, we replace it with don't care(the same for multiple fault) or else we will ignore it. By this way, in this paper sixteen patterns are reduced into eight patterns as they are shown in table 1 and table 2

Table 1
INITIAL TEST SET

Initial Test set T after fault simulation	
Test Pattern	
V0	1011
	1011
	1011
	1011
V1	0011
	0110
	1101
	1101
V2	0011
	0110
	1101
	1101
V3	0011
	0110
	1101
	1101

The above table represents the various patterns that has been generated by the Linear Feedback Shift Register, in order to find out the faults in the circuit. Among the patterns generated only few patterns detect the faults and one or more patterns detect the same fault. In the above tabular column each fault has been detected by four patterns, taken into consideration that a same pattern shouldn't be repeated within the same subset. These patterns find out each STUCK-AT fault that has been injected in the S27 circuit.

Table 2
RELAXED TEST SETS

Relaxed Test set T after fault simulation					
Test Pattern					**Detects**
n1	1	**0**	1	1	R1
n2	1	**0**	1	1	R1
n3	0	X	1	x	S9
n4	1	1	0	1	S9
n5	0	**X**	1	x	T10
n6	1	**1**	0	1	T10
n7	0	X	1	x	U10,u11
n8	1	1	0	1	U10, u11

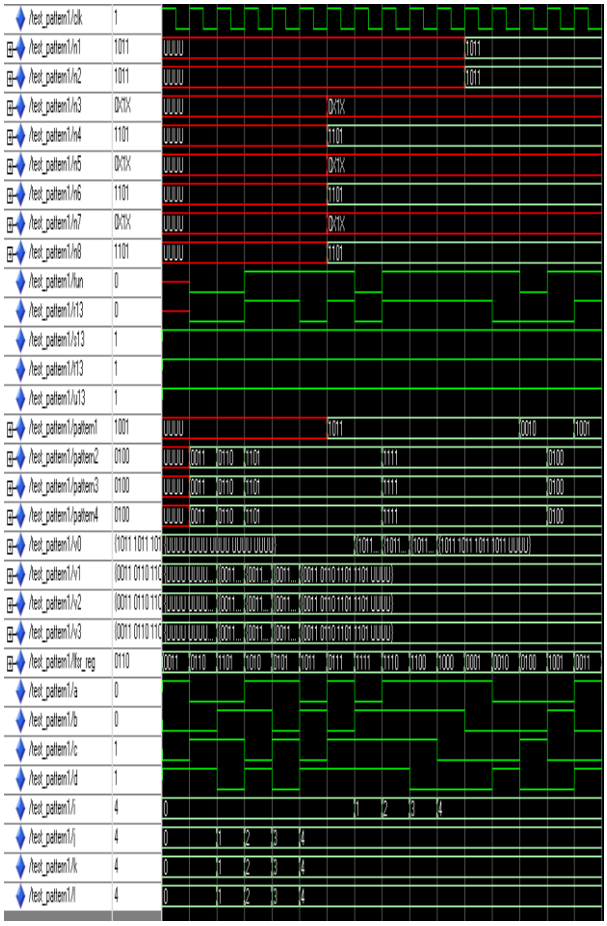
The above gives us the minimized test patterns. This is obtained by comparing the test patterns that are within the each subset respectively. These patterns are grouped in two's in their specific orders to find out their change in value and they are replaced with 'X' (i.e redundancy). Here comparison of 3rd and 4th patterns results in 4-bit don't care sequence, which can be removed from the list, because at least there should be one specified bit in the sequence. by

ISSN: 2278 – 7798

this way 16 patterns that has been generated have been reduced to 6 test patterns.

V. RESULTS AND DISCUSSION

The fault detection in the circuit have been analyzed and coded with the help of VHDL, and it is synthesized and simulated using MODEL SIM 6.3f software, which is simulation software used to design and testing of VLSI sequential circuits. The experimental results represent the patterns that have detected the faults in the S27 circuit. The patterns are randomly generated by the LFSR. Among the patterns generated, only a few patterns detect the fault **R_1, S_9, T_{10}, U_{10} and U_{11}.** These patterns are grouped in two's in their specific orders to find out their change in value and they are replaced with 'X'(i.e redundancy). While considering the faults at U_{10} and $U_{11,}$ the 3rd and 4th pattern results in 4-bit don't care pattern, which can be removed from the list, because at least there should be one specified bit in the sequence. When the patterns are minimized there is a moderate decrease in the processing time. Thus it gives a clear comparison that the processing time taken by the CPU is more during the implementation of initial test sets, when compared to benefits of the proposed technique after it has been introduced i.e by 0.25µs.



VI. CONCLUSION

The test pattern minimization is investigated the impact of test set relaxation in n-detect test sets. A systematic methodology for decreasing the number of specified bits in a given n-detect test set or a multiple detect test set. The experimental results reported demonstrate the effectiveness of the proposed method in achieving high specified bit reduction rates in n–detect test sets while maintaining the n-detect fault coverage. Provided discussion and experimentation data also explain how the defect coverage and non-targeted fault coverage of the relaxed test sets will be similar to that of the initial test sets when the relaxed test sets are fully-specified before test application. Thus by using MODEL SIM 6.3f software we are able to prove that the compactness of the test patterns that has been applied to the S_{27} circuit have been reduced i. e 16 patterns have been reduced to 6 patterns by using 'X' filling method. This 'x' can be either zero or one. And even we are able to clarify that there is a slight difference in the CPU processing time of about o.25µs, were the CPU processes in a faster rate after inducing the relaxed test technique.

VII. REFERENCES

[1] Stelios N. Neophytou and Maria K. Michael,"Test Pattern Generation of Relaxed n-Detect Test Sets", vol. 20, no. 3, march 2012.

[2] Antonis Paschalis, Ioannis Voyiatzis, and Dimitris Gizopoulos "Accumulator Based 3-Weight Pattern Generation", Vol. 20, No. 2, February 2012

[3] H. Lee and D. Ha, "Atalanta: An efficient ATPG for combinational circuits," Dept. Electr. Eng., Virginia Polytechnic Inst. State Univ., Blackburg,1993.

[4] I. Pomeranz and S. M. Reddy, "Test enrichment for path delay faults using multiple sets of target faults," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 22, no. 1, pp. 82–90, 2003.

[5] S.Wang and S. Gupta, "ATPG for heat dissipation minimization during test application," IEEE Trans. Comput., vol. 47, no. 2, pp. 256–262, 1998.

[6] K. C. Y. Mei, "Bridging and stuck-at faults," IEEE Trans. Comput., vol. C-23, no. 7, pp. 720–727, 1974.

[7] G. Mrugalski, J. Rajski, D. Czysz, and J. Tyszer, "New test data decompressor for low power applications," in Proc. DAC, pp. 539–544, 2007

[8] S. Neophytou and M. Michael, "Two new methods for accurate test set relaxation via test set replacement," in Proc. ISQED, pp.827–831, 2008

[9] El-Maleh and A. Al-Suwaiyan, "An efficient test relaxation technique for combinational & full-scan sequential circuits," in Proc. VTS, pp. 53–59, 2002.

[10] E. Kalligeros, X. Kavousianos, and D. Nikolos, "Multiphase BIST: A new reseeding technique for high

ISSN: 2278 – 7798

test-data compression," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 23, no. 10, pp. 1429–1446, 2004.

[11] K. Butler, J. Saxena, T. Fryars, and G. Hetherington, "Minimizing power consumption in scan testing: Pattern generation and DFT techniques," in Proc. ITC, 2004, pp. 355–364, IEEE Computer Society.

[12] J. Dworak, J. D. Wicker, S. Lee, M. R. Grimaila, M. R. Mercer, K. M. Butler, B. Stewart, and L.-C. Wang, "Defect-oriented testing and defective-part-level prediction," IEEE Design Test Comput., vol. 18, no. 1, pp. 31–41, Jan. 2001.

[13] B. Vaidya and M. Tahoori, "Delay test generation with all reachable output propagation and multiple excitation proc," in Proc. DFTS, 2005, pp. 380–388.

[14] S. Biswas, P. Srikanth, R. Jha, S. Mukhopadhyay, A. Patra, and D. Sarkar, "On-line testing of digital circuits for n-detect and bridging fault models," in Proc. ATS, 2005, pp. 88–93.

[15] J. Geuzebroek, E. J. Marinissen, A. Majhi, A. Glowatz, and F. Hapke, "Embedded multi-detect ATPG and its effect on the detection of unmodeled defects," in Proc. ITC, 2007.

Charliene Karunya. L completed my Under Graduate In Sri. Ramakrishna Engineering College, Coimbatore under the domain Electronics and Communication, during the year 2010 to 2014.

Divya Bharathy. k completed my Under Graduate In Sri. Ramakrishna Engineering College, Coimbatore under the domain Electronics and Communication, during the year 2010 to 2014.

Sheela Narayanan is working as Assistant Professor in Ramakrishna Engineering College, Coimbatore, Tamilnadu. She has received her M.E. degree in VLSI Design from Karpagam University, Coimbatore, Tamilnadu. She has completed her B.E. (Electronics & Communication Engg.) from VLB Janakiammal College of Engg & Tech, Coimbatore, Tamilnadu.

ISSN: 2278 – 7798