

Research Article

A Seed Selection Method for LFSR Reseeding Based Test Data Compression

¹K.S. Neelukumari and ²Dr. K.B. Jayanthi

¹Department of Electronics and Communication Engineering, P.A. College of Engineering and Technology, Pollachi,

²Department of Electronics and Communication Engineering, K.S. Rangasamy College of Technology, Namakkal, Tamilnadu, India

Abstract: Power dissipation is a significant problem as complexity of the circuit increases, which also increases during testing the VLSI circuits. So test data volume and test application time are major concerns for large industrial circuits. Test set selection is necessary to ensure that the most effective patterns are chosen from large test set in a high volume testing environment. LFSR reseeding forms the basis for many test compression solution, which increase the encoding efficiency of test compression based on LFSR. The most important criteria is to optimize the patterns generated for Built In Self-Test (BIST) which maximize the fault coverage and reduce the number of transition in the scan chains. Efficient techniques for test generations are essential in order to reduce the test generation time and size. In this study the output deviation method is used to select the effective test pattern from a large n-detect test set for test data compression scheme. The experiments are performed on ISCAS '85 benchmark circuits.

Keywords: BIST, encoding algorithm, Linear Feedback Shift Register (LFSR)

INTRODUCTION

An enormous number of various failures have been experienced during the manufacture of ICs and it is totally infeasible to analyze them individually. Thus, failures are clustered according to the logical fault effect on the functionality of the circuit which in turn leads to the edifice of logical fault models. Faults present in the IC can be classified into three types, namely, permanent fault, temporary (transient or intermittent) fault and delay fault. Both permanent and temporary faults purely time-dependant, the permanent faults exist long enough in the circuit which can be observed at test time. On the other hand the temporary faults appear and disappear in short interval of time. The delay fault will de-perform the operating speed of the circuit. High fault coverage is particularly valuable during the manufacturing of IC and it is accomplished by techniques such as Design for Test (DFT) and Automatic Test Pattern Generation (ATPG). BIST, a DFT methodology is advantageous than other testing scheme because of its testability, testing speed and independent of automatic test equipment.

Built-In-Self-Test (BIST) plays an important role in testing the complex and large circuits. The BIST will form a self testable circuit by accompanying Circuit under Test (CUT) in the chip, Test Pattern Generator (TPG), Test Response Verifier and BIST controller. In

conventional BIST architectures, the Linear Feedback Shift Register (LFSR) is commonly used in the Test Pattern Generators (TPGs) and output response analyzers. A major drawback of these architectures is that the pseudorandom patterns generated by the LFSR lead to significantly high switching activities in the CUT, which can cause excessive power dissipation. They can also damage the circuit and reduce product yield and lifetime. In addition, the LFSR usually needs to generate very long pseudorandom sequences in order to achieve the target fault coverage in nanometer technology. Least test application time and hardware overhead as well as least performance degradation are necessary in many BIST applications. The conventional SoC's and IP-cores are limited to a power dissipation factor. Complex block for test vector and pattern generation rises as a burden in VLSI testing in terms of power dissipations.

MATERIALS AND METHODS

The ordering technique, first presented in Al-Yamani *et al.* (2003), exploits the algebraic properties of the PRPG and the don't care bits in the patterns are to be encode the maximum number of patterns per seed. The encoding technique, presented in Al-Yamani and McCluskey (2003a) tries to encode seed by the number of additional clock cycles needed to reach it. In this

Corresponding Author: K.S. Neelukumari, Department of Electronics and Communication Engineering, P.A. College of Engineering and Technology, Pollachi, Tamilnadu, India

This work is licensed under a Creative Commons Attribution 4.0 International License (URL: <http://creativecommons.org/licenses/by/4.0/>).

study a technique for built-in reseeding (encoding the seeds in hardware) in Al-Yamani and McCluskey (2003b) is which combines mapping logic and reseeding and is based on running the LFSR in pseudorandom mode after every seed is load. In Krishna *et al.* (2001), the authors presented a scheme where the contents of the LFSR are incrementally modified instead of modifying them all at once. Both techniques achieve higher encoding efficiency than regular reseeding. The study (Volkerink and Mitra, 2003) presented a scheme for eliminating the boundaries between test patterns to improve the encoding efficiency using a simple modification to the BIST architecture.

In these studies, a new encoding algorithm that can be used in combining with any LFSR-reseeding scheme to significantly reduce the power consumption during testing mode in Jinkyu and Touba (2007) and Lee and Touba (2004). A key feature of the proposed approach is that it reduces the number of specified bits and the number of transitions at the same time. Since the amount of compression for LFSR reseeding depends on the number of specified bits, the proposed approach exploits this property. In Wang *et al.* (2008) and Chandra and Chakrabarty (2000) output deviation method is used to select the effective test pattern from large n -detect set. The confidence level of each gate is used to check the output for various input combination. The effective pattern is selected as seed for LFSR which is used to pre-computed set of test cubes. Vector ordering is an essential task that is needed to extract correlation among test data for compression. The seed selection is done by solving systems of linear equation based on feedback structure using LFSR (Kalligeros *et al.*, 2004). In the study (Xrysovalantis Kavousianos *et al.*, 2011) a technique is proposed based on new output-deviation metric for stuck-at pattern from LFSR seed.

In Chandra and Chakrabarty (2001) and Wang and Chakrabarty (2008a, 2008b) proposes an approach that encodes the slices of test data that are fed to the scan chains. It uses control code for the interpretation of data code. This encodes a subset of specified bits in a slice. The scan inversion is used to invert the subset of scan chain which improves the compression in Balakrishnan and Touba (2006).

Algorithmic TPG is mostly used for testing regular structures such as RAMs. Exhaustive TPG is a counter-based approach that is not able to generate specific sequence of test vectors. With some modifications, counter-based solutions are able to generate deterministic test patterns. Pseudo-random TPG is the most commonly applied technique in practice; here Linear Feedback Shift Register (LFSR) or Cellular Automata (CA) is employed to generate pseudo-random test patterns. In order to decrease the complexity of a TPG designers usually try to embed deterministic test patterns into the vector sequence generated by some linear register. Such embedding can be done either by re-seeding a TPG or modifying its feedback function.

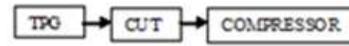


Fig. 1: BIST setup

Some solutions also modify or transform the vector sequence produced by a LFSR in such a way that it contains deterministic test patterns. In the Fig. 1 a test pattern generator (TPG) is used to apply a test sequence $\bar{T} = (T_1, T_2, \dots, T_n)$, represented by the polynomial $\bar{T}(x)$, into the circuit under test (CUT), is shown where n is the test length.

The test response sequence $R = (R_1, R_2, \dots, R_n)$, represented by the polynomial $R(x)$, is applied into the compressor that compresses $R(x)$ into a compact signature, i.e., a few bit word. Signature analysis is a method based on polynomial-division based linear compression, for TPG the test pattern are given from the LFSR.

In this study (Chandra and Chakrabarty, 2000), an encoding algorithm was presented that reduces both test storage and test power. Golomb code was used to encode the Partial Reseeding Scheme. An LFSR seed is the starting state of an LFSR when this is used to fill a set of scan chain with test vectors as shown in Fig. 2. Different LFSR seed will produce different test vectors. LFSR reseeding is used to compute a set of seeds that when expanded by LFSR will produce the test cubes. So instead of storing each full test vectors a much smaller seed is stored on the tester. The encoding efficiency for a set of test cube is defined as the total number of specified bits in the test cube divided by the total number of bits required to encode it. But the encoding efficiency based on reseeding algorithm is limited by large dependencies in LFSR and variance in number of specified bits in test cubes.

A single set of seed is transferred to the LFSR one at a time and expanded into the corresponding full test vectors in the scan chain. Since the seeds are much smaller than the full test vectors, the test data storage and bandwidth can be reduced for the test.

In the partial reseeding method, an extra XOR gate is included in the feedback of the LFSR. Let us consider the LFSR length r , is atleast $s_{\max} + 20$, where s_{\max} is the maximum number of specified bit in any test cube. The r -bit LFSR is initialized with starting r -bit seed. The initial seed is used to generate the first test cube by running the LFSR for m -clock cycles, to fill the scan chain in m -scan length. For second test cube, the LFSR is run LFSR is initialized for another m -clock cycle as shown in Fig. 3.

During each of first n -clock cycle a bit is shifted from the tester and xored with the feedback of the LFSR. These n -bits coming from the tester alter the state of the LFSR and in effect dynamically reseed the LFSR. After the first n -clock cycles, the tester stops shifting in data and the LFSR simply cycles through its normal sequence of state. The total number of bits required to encode a set of L test cubes using the proposed approach with an r -bit LFSR is $n(L-1)+r$. So the number of bits required for encoding is not

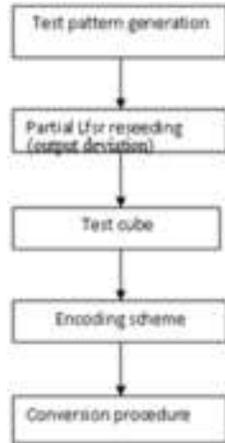


Fig. 2: Partial reseeding method using output deviation

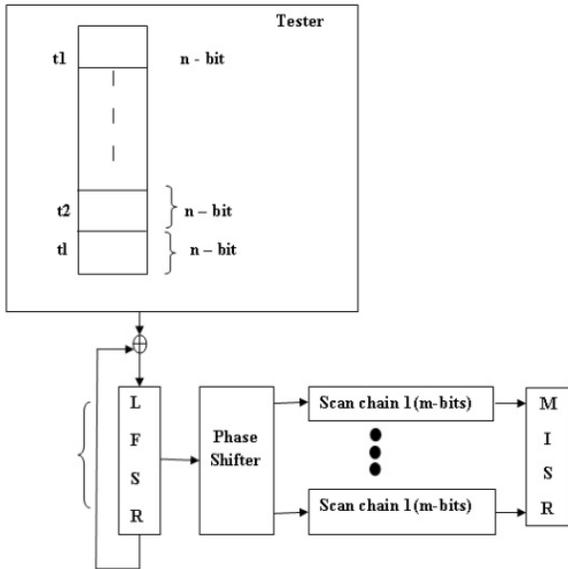


Fig. 3: Partial reseeding scheme

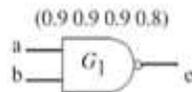


Fig. 4: An illustrative example

proportional to S_{max} , by which the length of LFSR size can be varied.

Seed selection based on output deviations: A technique is proposed to select seeds for the LFSR-based test pattern generators that are used in VLSI BISTs. By setting the computed seed as an initial value, target fault coverage, for example 100%, can be accomplished with minimum test length. It can also maximize fault coverage for a given test length. The output deviation as coverage metric and a test pattern grading method is used for pattern generation. The confidence level of a single output gate encompasses all the different input combination of the gate and for a

Table 1: Signal probabilities for different input combinations

Input pattern, z	$P_{e,0}$	$P_{e,1}$
0000, 0	0.1	0.9
0101, 0	0.1	0.9
1111, 1	0.8	0.2

Table 2: Example for encoding data

Block	Block 1
Original	0xx1
Encoded	00xx1

Table 3: Example for encoding data

Block	Block 1
Original	X111
Encoded	1----

given input combination. This is represented as (CL) of a gate with m-input and a single output which is defined as $R_i = (r_i^{(00\dots00)} r_i^{(00\dots01)} \dots r_i^{(11\dots11)})$ where the component of R_i denotes the probability that the gate output is correct for the corresponding input combination.

The signal probabilities can be computed for NAND and XNOR gates, let i be the output of a two-input gate g and let j and k be its input lines. If g is a NAND gate:

$$P_{i,0} = p_{j,1} p_{k,1} r_i^{(11)} + p_{j,0} p_{k,0} (1 - r_i^{(00)}) + p_{j,0} p_{k,1} (1 - r_i^{(01)}) + p_{j,1} p_{k,0} (1 - r_i^{(10)})$$

$$P_{i,1} = p_{j,0} p_{k,0} r_i^{(00)} + p_{j,0} p_{k,1} r_i^{(01)} + p_{j,1} p_{k,0} r_i^{(10)} + p_{j,1} p_{k,1} (1 - r_i^{(11)})$$

Let i be the output of a two-input gate g and let j and k be its input lines. If g is a XNOR gate, we have:

$$P_{i,0} = p_{j,0} p_{k,0} r_i^{(00)} + p_{j,0} p_{k,1} (1 - r_i^{(01)}) + p_{j,1} p_{k,0} (1 - r_i^{(10)}) + p_{j,1} p_{k,1} r_i^{(11)}$$

$$P_{i,1} = p_{j,0} p_{k,0} (1 - r_i^{(00)}) + p_{j,0} p_{k,1} r_i^{(01)} + p_{j,1} p_{k,0} r_i^{(10)} + p_{j,1} p_{k,1} (1 - r_i^{(11)})$$

Consider a two input NAND gate as shown in Fig. 4. Suppose each transition can be stuck-open due to defect i.e., it cannot be switched on, similarly suppose each transistor can be stuck on due to a defect i.e., it cannot be switched off. Next let us consider input combination $X_1 X_2 = 00$, if only stuck-open fault are considered. The NAND gate produces the correct output for this combination. The signal probability for different input combination is shown in the Table 1.

The patterns are sorted according to the output deviation such that the pattern with high deviation is considered as seed. A matrix is formed with all patterns in descending based on output deviation to select as seed by considering the fault coverage also. This seed is loaded in LFSR through tester. The encoding efficiency is compared between the normal reseeding and output based method. When the seed are loaded to LFSR, each n-bit LFSR can be divided into blocks. Each block is compressed based on encoding algorithm and if the seed have more number of zeros then the fault coverage may be reduced. So the probability value is compared

Table 4: Comparison result with block encoding scheme

Circuit name	Partial reseeding method		Partial reseeding using seed selection based on output deviation with block Encoding Scheme				
	No. of specified bits	Test storage (bits)	No. of specified bits	Test storage (bits)	% Reduction of specified bits	% Change in test storage	% Reduction in transition
C880	50	60	31	35	62.0	-41.67	15
C432	30	36	7	33	23.3	8.4	30
C499	39	41	20	35	51.2	-14.6	22
C1908	17	33	10	25	58.8	-24.2	18

Table 5: Comparison result with the block encoding scheme for s-circuit

Circuit name	No. of blocks	Encoding algorithm (Kalligeros <i>et al.</i> , 2004)			With output deviation method		
		No. of specified bits	No. of transition	% Reduction in transition	No. of specified bits	No. of transition	% Reduction in transition
S5378	5	4613	361552	31.6	4013	321551	30.6
	10	4312	154111	38.7	3998	150002	36.7
S9234	5	9840	524899	26.5	9002	524169	23.5
	10	8486	208477	34.3	7992	200266	30.5
S13207	5	8958	5542237	34.0	8052	5041125	31.4
	10	8536	2114606	36.3	8005	2014004	33.3

to check the high deviated output. This high deviated value is considered as seed.

Encoding algorithm: The encoding algorithm exploits the property that the number of transition in a test cube is smaller than the number of specified bits. The compatibility of blocks across the different test cube is exploited by reducing control information. A transition in a test cube be defined as a specified 0(1), followed by ‘0’s or more number of xs and then specified by 1(0). By using the encoding algorithm, the number of transition in a test cube is always less than the number of specified bits in an each test cube. In the proposed encoding algorithm the cube is divided into blocks that contain transition.

Each test cube will have two kinds of data, they are hold flag and data bit. In each block there is 1-bit hold flag which indicate whether a transition occur in a block. There are three types of blocks:

Transition block: In Table 2 the hold flag = 0, one or more transition occurs in the block. Both 0 and 1 are present in the block and then the data bits are loaded directly from the LFSR.

Non transition block: In Table 3, if hold flag = 1, only 0 or 1 is present in the block and then the data bits in the block are simply held constant.

Don’t care block: The hold flag = X there is no specified bits, then it is treated either or a non-transition block with all X data. By using this encoded data, the number specified bits that need to be generated using LFSR reseeding.

RESULTS AND DISCUSSION

The program is written to compute output deviation for test vector. The simulation is performed in Xilinx ISE and modelsim. The simulation is performed for the benchmark circuit C880.

Table 4 shows the comparison between partial reseeding and encoding algorithm for c-circuits, the number of specified bit is comparatively less by using the proposed encoding scheme. The number of transition is also less when compared to the existing system.

Table 5 proposes simulation result for total number of specified bits and total number of transition for the proposed encoding are compared with the previous encoding algorithm scheme. In most cases the total number of specified bits are reduced which results in less test-data.

CONCLUSION

The output deviation is an effective metric to model the quality of test pattern. Pattern ranking based on output deviation offers a useful method for test-set selection during high volume. Effective reseeding algorithm has been combined to reduce the test storage and the transition. The proposed encoding scheme provides a way to reduce the test power in LFSR and also improve the test compression. By using output deviation in partial LFSR reseeding an appropriate seed can be selected so the transition activity will be reduced compare to all other reseeding techniques.

REFERENCES

- Al-Yamani, A.A. and E.J. McCluskey, 2003a. Seed encoding with LFSRs and cellular automata. Proceeding of the Design Automation Conference, pp: 560-565.
- Al-Yamani, A.A. and E.J. McCluskey, 2003b. Built-in reseeding for serial BIST. Proceeding of the VLSI Test Symposium, pp: 63-68.
- Al-Yamani, A.A., S. Mitra and E.J. McCluskey, 2003. BIST reseeding with very few seeds. Proceeding of the 21st VLSI Test Symposium, IEEE Computer Society, pp: 69-74.

- Balakrishnan, K.J. and N.A. Touba, 2006. Improving linear test data compression. *IEEE T. VLSI Syst.*, 14(11): 1227.
- Chandra, A. and K. Chakrabarty, 2000. Test data compression for system on- a-chip using Golomb codes. *Proceeding of the VLSI Test Symposium*, pp: 113-120.
- Chandra, A. and K. Chakrabarty, 2001. Combining low-power scan testing and test data compression for system-on-a-chip. *Proceeding of the 38th Annual Design Automation Conference (DAC '01)*, pp: 166-169.
- Jinkyu, L. and N.A. Touba, 2007. LFSR-reseeding scheme achieving low-power dissipation during test. *IEEE T. Comput. Aid. D.*, 26(2): 396-401.
- Kalligeros, E., X. Kavousianos and D. Nikolos, 2004. Multiphase BIST: A new reseeding technique for high test-data compression. *IEEE T. Comput. Aid. D.*, 23(10): 1429-1446.
- Krishna, C.V., A. Jas and N.A. Touba, 2001. Test vector encoding using partial LFSR reseeding. *Proceeding of the International Test Conference*, pp: 885-893.
- Lee, J. and N.A. Touba, 2004. Low power test data compression based on LFSR reseeding. *Proceeding of the International Conference on Computing Design: VLSI in Computers and Processors*, pp: 180-185.
- Volkerink, E.H. and S. Mitra, 2003. Efficient seed utilization for reseeding based compression. *Proceeding of the 21st VLSI Test Symposium*, pp: 232-237.
- Wang, Z. and K. Chakrabarty, 2008a. Test-quality/cost optimization using output-deviation-based reordering of test patterns. *IEEE T. Comput. Aid. D.*, 27(2): 352-365.
- Wang, Z. and K. Chakrabarty, 2008b. Test data compression using selective encoding of scan slices. *IEEE T. VLSI Syst.*, 16(11): 1429-1440.
- Wang Z., F. Hongxia, K. Chakrabarty and M. Bienek, 2008. Deviation-based LFSR reseeding for test-data compression. *IEEE T. Comput. Aid. D.*, 28(2): 259-271.