Low Power Mixed-Mode BIST Based on Mask Pattern Generation Using Dual LFSR Re-seeding

Paul M. Rosinger* and Bashir M. Al-Hashimi
Dept. of ECS
University of Southampton, United Kingdom
{p.rosinger,bmah}@ecs.soton.ac.uk

Nicola Nicolici†
Dept. of ECE
McMaster University, Canada
nicola@ece.mcmaster.ca

Abstract

Low power design techniques have been employed for more than two decades, however an emerging problem is satisfying the test power constraints for avoiding destructive test and improving the yield. Our research addresses this problem by proposing a new method which maintains the benefits of mixed-mode built-in self-test (BIST) (low test application time and high fault coverage), and reduces the excessive power dissipation associated with scan-based test. This is achieved by employing dual linear feedback shift register (LFSR) re-seeding and generating mask *patterns to reduce the switching activity. Theoretical analysis and experimental results show that the proposed method consistently reduces the switching activity by 25% when compared to the traditional approaches, at the expense of a limited increase in storage requirements.*

1. Introduction

Scan-based built-in self-test (BIST) represents an attractive solution to the challenges of testing complex integrated circuits. Using simple built-in structures for pattern generation and test response analysis eliminates the need for expensive external test equipment as well as the problem of external access to internal components (cores) of complex integrated circuits. The general approach uses a simple random-pattern generator, for example a linear feedback shift register (LFSR), which minimizes both the hardware overhead and the impact on system's performance [1]. Several techniques have been proposed for alleviating the problem of random pattern resistant (RPR) faults. These solutions range from test point insertion to weighted random pattern testing and mixed-mode testing [1] and offer different trade-offs between fault coverage, area, performance and testing time.

In mixed-mode BIST a limited number of pseudo-random vectors are used to cover the easy-to-detect faults, while the few remaining RPR (or hard-to-detect) faults are covered with a small number of deterministic vectors. The deterministic vectors are stored on-chip in a compressed format and expanded during test (i.e. "store and generate" architectures) [5, 8, 6], or directly embedded into an LFSR sequence by "bit-fixing" [11] or "bit-flipping" [13] techniques. Although "bit-fixing" and "bit-flipping" techniques provide high quality tests, the corresponding BIST hardware is very dependent on the test set and the circuit under test (CUT), thus any change in the test set or CUT requires a complete re-synthesis of the BIST hardware. "Store and generate" BIST architectures represent a more flexible solution with comparable associated costs. In these approaches, deterministic patterns are encoded as seeds of simple test pattern generators (TPGs) such as LFSRs [6] or multiple-polynomial LFSRs (MP-LFSRs) [5, 8]. Koenemann [6] proposed an interesting method of encoding test data based on controlled re-seeding of single polynomial LFSRs. This technique is suitable for scan designs and delivers, at the cost of a small storage requirement, shorter test application time when compared to weighted random pattern generators. Hellebrand et al. [5] extended the LFSR re-seeding technique to multiple-polynomial LFSRs which reduces the storage requirements and the LFSR length when compared to the re-seeding of single-polynomial LFSRs. Rajski et. al. [8] adapted the MP-LFSR TPG to a test environment with multiple scan-chains and boundary scan chain.

While mixed-mode BIST solves the test application time and fault coverage problems associated with pseudorandom test [2], its main drawback is the excessive power dissipation caused by the uncorrelated sequences in the scan chain. The excessive power dissipation may lead to destructive test or manufacturing yield loss. Many techniques for reducing switching activity during scan or BIST have been proposed recently and summarized in [4, 7]. The TPG schemes for generating correlated vectors [12] or non-detecting pattern suppression [3] require very little hardware overhead, however they do not provide high fault coverage for circuits with RPR faults, which is the very purpose of mixed-mode BIST. To the best of our knowledge no previous approaches have addressed the problem of high power dissipation associated with mixed-mode BIST, which is the aim of this paper.

This paper presents a new TPG architecture based on re-seeding MP-LFSR structures which achieves high fault coverage in short test application time while consistently re-

*Work carried out in part at McMaster University

†Research supported in part by Micronet (C.6.MM2) and NSERC (RGP239-003-01)

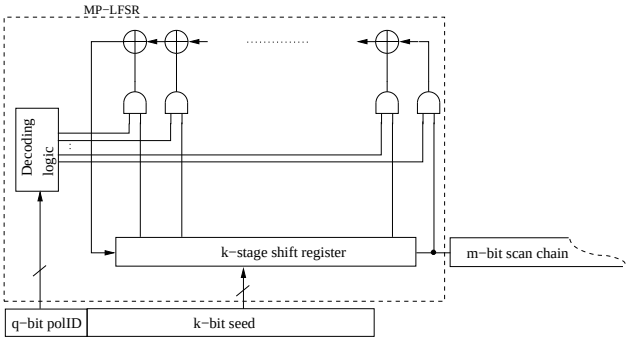

Figure 1. MP-LFSR approach for mixed-mode BIST [5, 8]

ducing the number of transitions in the scan chain by 25%. The rest of the paper is organized as follows. Section 2 reviews the basic concepts behind the use of re-seedable MP-LFSRs and their high switching activity. Section 3 presents our method for reducing the number of transitions in the scan chain, and consequently the power dissipation in the circuit under test, without affecting the test quality. Our method exploits the "masking" properties of AND and OR logic functions and the randomness of patterns generated by LFSRs. Section 4 describes the TPG architecture which implements the method presented in section 3. Section 5 reports the results of the experiments performed to assess the efficiency of our method.

2. Deterministic test generation by re-seeding multiple-polynomial LFSRs

In this section we summarize the theory for encoding deterministic test cubes using MP-LFSRs and, using a motivational example, we show that this approach can lead to unnecessarily high switching activity.

While most faults in a circuit can be detected with high probability with any pseudo-random pattern sequence of a certain, yet reasonable, length, detecting the few RPR faults of a circuit requires prohibitively long pseudo-random sequences. Hence, in order to keep the test application time within acceptable limits, in mixed mode BIST the easy-to-detect faults are covered with a limited number of LFSR-generated pseudo-random patterns, and the remaining RPR faults are detected using a few deterministic test cubes stored on chip. The sparseness of "care bits" (or specified bits) in deterministic test cubes (typically less than 10% of the length of the scan chain) suggested the idea of encoding them into a compressed form in order to avoid large and unjustified test data storage requirements.

Figure 1 shows the basic structure of a MP-LFSR pattern generator. The decoding logic is used to select one of the several possible LFSR feedback configurations, and the seeds are loaded serially or in parallel into the shift register. Hence, in this approach, a test cube C is encoded as an identifier for the LFSR feedback polynomial (the q-bit polID from Figure 1) and the initial seed (the k-bit seed from Figure 1) which will generate a test vector covered by C. Let $h(x) = x^k + \sum_{i=0}^{k-1} h_i x^i$ be the feedback polynomial of the LFSR and

$$A(t) = \begin{bmatrix} a_0(t) \\ \vdots \\ a_{k-1}(t) \end{bmatrix}$$

the state of the shift register at clock t. The system can be described as

$$A(t+1) = T_S A(t)$$

where

$$T_S = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & 0 \\ 0 & 0 & 0 & \dots & 0 & 1 \\ 1 & h_1 & h_2 & \dots & h_{k-2} & h_{k-1} \end{bmatrix}$$

represents the state transition matrix for the given LFSR. Let $C = (c_0, ..., c_{m-1}) \in \{0, 1, X\}^m$ be a test cube and $S(C) = \{i \mid c_i \neq X\}$ the set of specified bits of C. C can be generated using the LFSR described by T_S if the following system of linear equations:

$$c_i = a_i = \left[{T_S}^i A(0)\right]_1, \quad for\ all\ i \in S(C) \tag{1}$$

is consistent, where $\left[{T_S}^i A(0)\right]_r$ denotes the r-th component of $\left[{T_S}^i A(0)\right]$. The solution of this system of equations, if it exists, represents the initial seed from which the LFSR described by $h(x)$ will generate a test vector covered by cube C. It was shown in [5] that the probability of not finding a seed for a test cube with $s = |S(C)|$ specified bits using a LFSR with $s + 4$ stages and 16 possible feedback polynomials is less than 10^{-6}. Thus, by using this technique the storage requirements for encoding a test cube are determined only by the number of specified bits in the test cube. The procedure for computing the initial seed for a given test cube and feedback polynomial is illustrated in the following example.

Example 1 Consider test cube $C = (X, 1, 1, X, X, 0, X, X, 1, X)$ and a 4-stage LFSR with the characteristic polynomial given by $h(x) = x^4 + x^3 + 1$. The corresponding transition matrix will be

$$T_S = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix}$$

The equations which need to be solved in order to find the initial seed are

$$\begin{bmatrix} c_0 = X \\ c_1 = 1 \\ c_2 = 1 \\ c_3 = X \end{bmatrix} = {T_S}^0 \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix}$$

and

$$\begin{bmatrix} c_5 = 0 \\ c_6 = X \\ c_7 = X \\ c_8 = 1 \end{bmatrix} = {T_S}^5 \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix}$$

Solving the equations above will produce the following solution, i.e. initial seed

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

The test pattern generated by the given LFSR starting from the computed initial seed will be $P = (0, \underline{1}, \underline{1}, 1, 1, \underline{0}, 1, 0, \underline{1}, 1)$, where the underlined positions represent the specified bits from the original test cube. This example has shown how a 10-bit test cube can be encoded as a 4-bit LFSR seed. The pattern P generated using this initial seed contains 5 transitions between successive bits.

Patterns generated by LFSRs, with or without re-seeding, contain a large number of transitions between consecutive bits due to the pseudo-random behavior of LFSRs. In the following section we are proposing a method for reducing the number of transitions from LFSR-generated patterns by exploiting properties of AND and OR logic functions. Our approach, tailored for a mixed-mode BIST environment, overcomes the shortcomings of previous solutions for low power scan testing by providing high fault coverage within short test application time and with reduced power dissipation during test.

3. Reducing the number of transitions in the scan chain

This section explains the method we are proposing in order to reduce the number of transitions in the scan chain when using a MP-LFSR based TPG. First, we give some definitions and basic concepts.

Given a logic signal S, the signal probability $P_1(S)$ represents the average fraction of clock cycles when signal S is 1. Analogously, $P_0(S)$ represents the average fraction of clock cycles when signal S is 0. Obviously, $P_0(S) + P_1(S) = 1$. If signal S is generated by a random source, its signal probabilities are equal $P_0(S) = P_1(S) = 0.5$ The transition probability of signal S, $P_{tr}(S)$ represents the average fraction of clock cycles when the current value of S is different than its previous value. Assuming temporal independence between consecutive values of S, the transition probability can be computed as $P_{tr}(S) = P_0(S) \times P_1(S) + P_1(S) \times P_0(S)$. The transition probability of a random logic signal S is

$$P_{tr}(S) = 0.5 \times 0.5 + 0.5 \times 0.5 = 0.5 \quad (2)$$

Assume two mutually independent random signals S_A and S_B, and let $S_{AND} = S_A \, AND \, S_B$. Given the mutual independence of S_A and S_B, the signal probabilities of S_{AND} can be computed as follows:

$$\begin{aligned} P_1(S_{AND}) &= P_1(S_A) \times P_1(S_B) = 0.25 \\ P_0(S_{AND}) &= 1 - P_1(S_{AND}) = 0.75 \end{aligned}$$

Hence, the transition probability of S_{AND} is given by

$$\begin{aligned} P_{tr}(S_{AND}) &= 2 \times P_1(S_{AND}) \times P_0(S_{AND}) \\ &= 0.375 \end{aligned} \quad (3)$$

In a similar fashion we can compute the transition probability of an OR composition of S_A and S_B, $S_{OR} = S_A \, OR \, S_B$:

$$P_{tr}(S_{OR}) = 0.375 \quad (4)$$

From equations (2), (3) and (4) we conclude that both AND and OR compositions of two mutually independent random signals produce signals with transition probabilities 25% lower than those of the original signals. The fact that LFSR-generated bit-sequences exhibit a high degree of randomness together with the previous observation motivated the use of AND/OR composition for reducing the number of transitions in the scan chain, and consequently the power dissipation in the CUT. In the following we will explain how this method can be used in a mixed mode BIST environment. There are two problems which need to be addressed in order to guarantee the fault coverage of patterns generated by AND/OR composition:

- The first problem is to ensure that by AND/OR composition we can produce patterns with good random properties for covering the easy-to-detect faults within a reasonable amount of time. This is achieved by performing the AND/OR composition on mutually independent patterns.
- The second problem is to employ AND/OR composition for generating patterns corresponding to deterministic test cubes for RPR faults. This is addressed next.

The previous section has shown how a pattern P covering the specified bits in a test cube C can be generated by re-seeding an LFSR. What we need now is an algorithmic method for deriving a "mask pattern" MP which, by AND/OR composition with P, will produce a pattern P' which covers the specified bits from C, $S(C) = \{i \in C | c_i \neq X\}$. The procedure we are proposing for generating the suitable"mask pattern" consists of the following steps:

1. chose a composition function $f_{comp} \in \{AND, OR\}$
2. from C compute the "mask cube" MC comprising the bits in $M(C) = \{i \in S(C) \mid c_i = NCV(f_{comp})\}$, where $NCV(f_{comp})$ is the non-controlling value of f_{comp}, i.e. 1 for AND and 0 for OR;
3. generate a "mask pattern" MP which covers the specified bits in $M(C)$.

The MP constructed using the above procedure guarantees that the pattern $P' = f_{comp}(P, MP)$ will cover all the specified bits from C. This is because the bits in $M(C)$ are covered both by P and MP, and hence they are preserved through f_{comp}, while the bits in $S(C) - M(C)$ are covered by P and having the controlling value of f_{comp} they are not altered by f_{comp}. Thus, both P and MP can be generated by seeding LFSR structures. The seeds for P and MP can be computed as described in the previous section. The following example illustrates the procedure for generating the "mask pattern" for a given test cube.

Example 2 Assume we want to generate the "mask pattern" MP corresponding to the AND composition for the deterministic cube $C = (X, 1, 1, X, X, 0, X, X, 1, X)$ from Example 1. The "mask cube" for AND composition is $MC = (X, 1, 1, X, X, X, X, X, 1, X)$. Consider $h_m(x) = x^3 + x + 1$ as the characteristic polynomial LFSR we will use to generate MP. We compute the initial seed for MC as described in the previous section. The state transition matrix associated to the LFSR is:

$$T_{Sm} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

The equations needed to compute the initial seed are:

$$\begin{bmatrix} c_0 = X \\ c_1 = 1 \\ c_2 = 1 \end{bmatrix} = {T_{Sm}}^0 \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix}$$

and

$$\begin{bmatrix} c_6 = X \\ c_7 = X \\ c_8 = 1 \end{bmatrix} = {T_{Sm}}^6 \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix}$$

Solving these equations will lead to the following seed:

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

The "mask pattern" which will be generated using this seed is $MP = (1, \underline{1}, \underline{1}, 0, 0, 1, 0, 1, \underline{1}, 1)$, where the underlined positions represent the specified bits from MC. The pattern resulted by AND composition of P and MP is $P' = (0, \underline{1}, \underline{1}, 0, 0, \underline{0}, 0, 0, \underline{1}, 1)$, where the underlined positions show the specified bits from the original test cube C, preserved through AND composition. The pattern P from Example 1 generated by the traditional MP-LFSR TPG has 5 transitions between consecutive bits, while the pattern P' produced by our method for the same deterministic cube contains only 3 transitions between consecutive bits.

Thus we have shown how suitable "mask patterns" which reduce the number of transitions in the scan chain by AND (OR) composition can be generated in a similar manner with patterns for deterministic test cubes. By using our method, a deterministic cube is encoded as two (polynomial identifier, initial seed) pairs: one pair encodes the deterministic test cube while the second pair encodes the "mask cube", corresponding to the test cube and to the selected composition function, which ensures that the specified bits in the test cube are preserved through composition. The following section describes a new TPG architecture which implements this method.

4. Test pattern generator for low power mixed-mode BIST

In the previous section we have shown how AND(OR) composition of MP-LFSR generated pseudo-random sequences can be used to reduce the number of transitions in the scan chain during pseudo-random test. We have also provided a method of generating "mask patterns" for reducing the number of transitions during the deterministic testing mode without affecting the fault coverage. In this section we will describe the architecture of our TPG based on AND(OR) composition and re-seeding of MP-LFSRs.

The basic idea is to have two different MP-LFSRs operated in parallel and apply to the input of the scan chain the AND/OR composition of the bit-sequences generated by the two MP-LFSRs. The main MP-LFSR will act as a traditional MP-LFSR TPG [5, 8], while the secondary MP-LFSR will produce "mask patterns" to reduce the number of transitions in the scan chain. We have implemented this solution with the TPG shown in Figure 2. The two MP-LFSRs share the decoding logic for selecting the feedback configurations in order to minimize the hardware overhead and the memory required for storing the polynomial identifiers for the test and mask cubes. The randomness of the test patterns generated by such an architecture, which ensures rapid coverage of easy-to-detect faults, can be easily achieved by using different primitive characteristic polynomials for the two MP-LFSRs. Generation of deterministic patterns for covering the RPR faults is achieved by re-seeding the two MP-LFSRs with stored seeds computed as explained in sections 2 and 3.

The minimum length of the MP-LFSR, and consequently the size of the memory for storing the initial seeds, is given by the number of specified bits in the cube to be encoded, as mentioned in section 2. The size of the main MP-LFSR, k from Figure 2, is determined by the maximum number of specified bits per cube in the precomputed test cubes. However in the case of the secondary MP-LFSR, for each test cube we have two possible mask cubes, one for AND composition and one for OR composition. We can exploit this fact by selecting for each test cube the composition function which leads to the mask cube with less specified vectors. The MS bit from Figure 2 is used to select the appropriate composition function for each test cube. In the following we will determine an upper bound for the maximum number of specified bits in mask cubes when we can choose for each cube between AND and OR composition. Given a test cube C, let $M_{AND}(C)$ and $M_{OR}(C)$ be the sets of specified bits of the mask cubes corresponding to AND and OR composition respectively. The mask cube which will be selected in order to minimize the MP-LFSR size will have

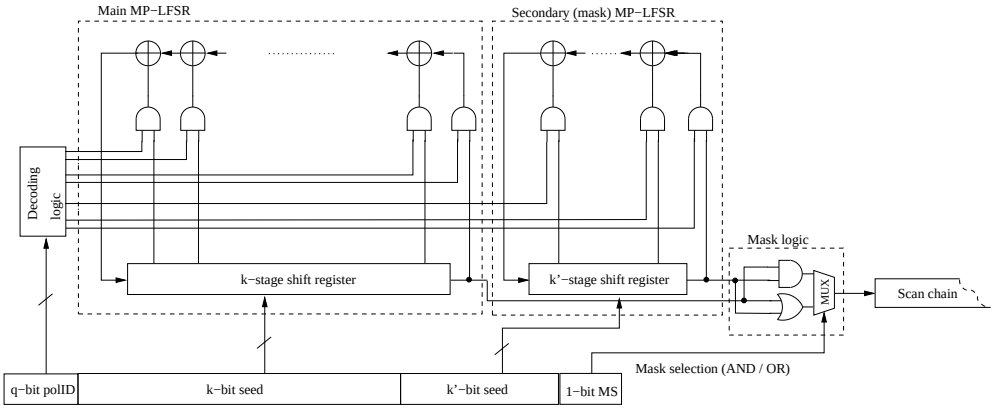

Figure 2. Proposed dual MP-LFSR TPG architecture for low power mixed-mode BIST

$s' = min(|M_{AND}(C)|, |M_{OR}(C)|)$ specified bits. As the sets of specified bits of $M_{AND}(C)$ and $M_{OR}(C)$ are disjoint, and their union is equal to $S(C)$, the set of specified bits of C, we have

$$|M_{AND}(C)| + |M_{OR}(C)| = |S(C)| \quad (5)$$

$$s' \leq |M_{AND}(C)| \quad (6)$$

$$s' \leq |M_{OR}(C)| \quad (7)$$

By adding inequalities (6) and (7) and using equation (5) we obtain:

$$s' \leq \frac{|S(C)|}{2} \quad (8)$$

Thus the upper bound for the length of the secondary MP-LFSR, k' from Figure 2, is given by $\frac{|S(C)|}{2} + l$, where l is a small constant, 2 in our experiments, which ensures that the probability of finding an initial seed for each mask cube is high, and consequently the total number of feedback polynomials is small. Although intuitively it seems that we need to double the storage requirements, inequality (8) has shown that the length of the secondary (mask) MP-LFSR from Figure 2 is only approximately 50% of the length of the main MP-LFSR, whose length is dependent on $|S(C)|$ (see section 2).

5. Experimental results

We have performed several experiments using the full scan versions of the ISCAS-89 circuits in order to asses the efficiency and the cost of the proposed method. We have used ATALANTA [9] for automatic generation of the deterministic test cubes and FSIM [10] for fault simulation. The method for computing initial seeds and feedback polynomials described in section 2, and our method for deriving mask patterns described in section 3 were implemented using C++.

In our experiments shown in Table 1, we have first assigned some default initial seeds and feedback polynomials to the two MP-LFSRs. We have also selected a default composition function, AND for example. Using this configuration we have generated 1k pseudo-random patterns and

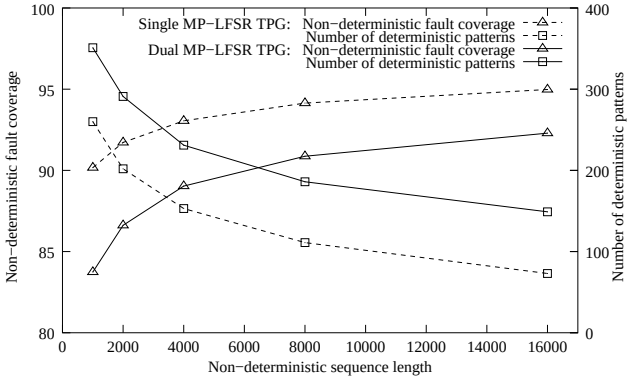

Figure 3. Circuit s38584: Non-deterministic fault coverage vs. the number of deterministic patterns for single and dual MP-LFSR TPG

fault simulated them on the target circuit. For the remaining undetected faults we have used ATALANTA to derive the deterministic test cubes. We have set the length of the main MP-LFSR to $PDm = s_{max} + 2$ where s_{max} is the maximum number of specified bits s in all the test cubes in the precomputed test cube set. We have set the length of the secondary MP-LFSR to $PDs = s'_{max} + 2$ where s'_{max} is the maximum number of specified bits s' in all the mask cubes. Having set the length of both MP-LFSRs, we have computed for each test cube and corresponding mask cube the initial seeds, feedback polynomials and corresponding output patterns for the two MP-LFSRs as well as the pattern resulted from the composition of the latter. Column **NC** from Table 1 gives the number of deterministic test cubes computed to achieve maximal fault coverage. Columns **NPm** and **NPs** show the number of polynomials used by the main and secondary MP-LFSRs to cover all test and mask cubes. Columns **PDm** and **PDs** give the length of the polynomials used by the two MP-LFSRs (**m**ain and **s**econdary). Columns **TC** and **origTC** give the number of transitions in the scan chain obtained by using our TPG and respectively a traditional TPG with a single MP-LFSR [5, 8]. Column **redTC%** shows the reduction in transition count, which is

Circuit	NC	NPm	PDm	NPs	PDs	TC	origTC	redTC%	xStorage%	origFC%	mFC%
s420	44	2	22	1	8	231378	306721	24.56	36.36	76.92	72.3
s526	32	2	15	1	8	116369	152027	23.46	53.33	95.49	86.99
s641	23	1	24	2	10	577057	753730	23.44	41.67	96.32	88.98
s713	22	2	24	2	12	573862	753044	23.79	50	90.53	84.3
s820	53	5	15	1	8	108245	143475	24.55	53.33	91.41	81.52
s832	52	4	15	1	8	108090	143284	24.56	53.33	89.88	79.66
s838	131	1	38	1	8	904106	1202210	24.8	21.05	58.75	59.39
s953	39	1	17	3	9	404086	532909	24.17	52.94	83.31	91.47
s1196	124	1	19	4	9	227156	294182	22.78	47.37	88.64	73.91
s1238	130	1	19	4	9	228405	295494	22.7	47.37	83.1	68.48
s1423	28	1	28	2	13	1611456	2137027	24.59	46.43	96.96	93.33
s5378	199	2	29	5	15	10188661	13592997	25.04	51.72	93.78	86.17
s9234	243	2	53	2	27	14152373	18707043	24.35	50.94	70.72	63.7
s13207	319	3	24	3	13	120663103	161446001	25.26	54.17	79.82	81.14
s15850	238	3	40	2	21	86268594	115238803	25.14	52.5	86.23	84.65
s38417	498	3	88	2	45	771889958	1027848936	24.9	51.14	86.19	84.89
s38584	351	4	56	2	29	543841106	722223634	24.7	51.79	90.32	83.74

Table 1. Experimental results for 1k pseudo-random patterns and deterministic patterns for maximal fault coverage

consistent with the probabilistic estimation of 25% reduction from section 3. Column **xStorage%** shows the amount of additional storage required for storing the mask seeds, relative to the storage size for the test seeds, result which is consistent with the theoretical upper bound of nearly 50% determined in section 4. Finally, columns **origFC** and **mFC** compare the fault coverage of 1k pseudo-random pattern sequences and 1k pattern sequences generates using our TPG in non-deterministic mode. The lower fault coverage of the non-deterministic sequences generated by our TPG, a side effect of the masking process, is compensated by a higher number of deterministic patterns, needed to achieve maximal fault coverage. Figure 3 shows, for both the traditional TPG with a single MP-LFSR as well as for our dual MP-LFSR TPG, the relation between the number of non-deterministic test patterns, their fault coverage and the number of deterministic test patterns needed to achieve maximal fault coverage. The number of deterministic patterns for full fault coverage decreases as the length of the non-deteterministic sequence increases. Hence, the storage requirements can be controlled by varying the length of the non-deterministic sequence.

6. Conclusions

In this paper we have proposed a new TPG for mixed-mode BIST based on mask pattern generation and reseeding of dual MP-LFSRs. By employing AND and OR masking, our TPG, at the cost of additional, yet limited, storage requirements, reduces the number of transitions in the scan chain by 25% while preserving the fault coverage and test application time when compared with a traditional TPG with a single MP-LFSR [5, 8]. Future work will investigate the trade-offs in power dissipation and storage requirements when using AND/OR composition functions which have an increased fan-in and hence additional mask MP-LFSRs that will further lower power dissipation during test.

References

[1] M. Abramovici, M. A. Breuer, and A. D. Friedman. *Digital Systems Testing and Testable Design*. IEEE Press, 1990.

[2] P. H. Bardell, W. H. McAnney, and J. Savir. *Built-In Self Test - Pseudorandom Techniques*. John Wiley & Sons, 1986.

[3] S. Gerstendorfer and H. Wunderlich. Minimized power consumption for scan-based BIST. In *Proc. IEEE International Test Conference*, pages 77–84, 1999.

[4] P. Girard. Low power testing of VLSI circuits: Problems and solutions. In *First International Symposium on Quality of Electronic Design (ISQED)*, pages 173–180, 2000.

[5] S. Hellebrand, J. Rajski, S. Tarnick, S. Venkataraman, and B. Courtois. Built-in test for circuits with scan based on reseeding of multiple-polynomial linear feedback shift registers. *IEEE Transactions on Computers*, 44(2):223–233, February 1995.

[6] B. Koenemann. LFSR-coded test patterns for scan designs. In *European Test Conference*, pages 237–242, 1991.

[7] N. Nicolici. *Power Minimisation Techniques for Testing Low Power VLSI Circuits*. PhD thesis, University of Southampton, UK, http://www.bib.ecs.soton.ac.uk/records/4937/, October 2000.

[8] J. Rajski, J. Tyszer, and N. Zacharia. Test data decompression for multiple scan designs with boundary scan. *IEEE Transactions on Computers*, 47(11):1188–1200, November 1998.

[9] V. Tech. ATALANTA. In *http://www.ee.vt.edu/ "ha/ cadtools/ cadtools.html*.

[10] V. Tech. FSIM. In *http://www.ee.vt.edu/ "ha/ cadtools/ cadtools.html*.

[11] N. A. Touba and E. J. McCluskey. Altering a pseudo-random bit sequence for scan-based BIST. In *IEEE Internatinal Test Conference (ITC)*, pages 167–175, 1996.

[12] S. Wang and S. K. Gupta. LT-RTPG: A new test-per-scan BIST TPG for low heat dissipation. In *Proc. IEEE International Test Conference*, pages 85–94, 1999.

[13] H. J. Wunderlich and G. Kiefer. Bit-flipping BIST. In *International Conference on Computer Aided Design (ICCAD)*, 1996.