A Retiming-Based Test Pattern Generator Design for Built-In Self Test of Data Path Architectures

Aiman H. El-Maleh and Yahya E. Osais

Department of Computer Engineering
King Fahd University of Petroleum and Minerals, Dhahran 31261, Saudi Arabia
Email: {aimane,yosais}@ccse.kfupm.edu.sa

Abstract

Recently, a new Built-In Self Test (BIST) methodology based on balanced bistable sequential kernels has been proposed that reduces the area overhead and performance degradation associated with the conventional BILBO-oriented BIST methodology. This new methodology guarantees high fault coverage but requires special test sequences and test pattern generator (TPG) designs. In this paper, we demonstrate the use of the retiming technique in designing TPGs for balanced bistable sequential kernels. Experimental results on ISCAS benchmark circuits demonstrate the effectiveness of the designed TPGs in achieving higher fault coverage than the conventional maximal-length LFSR TPGs.

1. Introduction

In a conventional BILBO [1] test design methodology (TDM), each Combinational logic block in a circuit is tested by applying a test sequence generated by the BILBO registers at its inputs and capturing its response by the BILBO registers at its outputs. The area overhead and performance impact of such a technique can be excessive due to the need to replace each register in the original circuit by a BILBO register. In order to reduce the area overhead and performance degradation associated with the BILBO registers, the circuit needs to be partitioned into larger kernels, usually sequential structures. Such kernels need to be identified so that high fault coverage, short test time, and reduction in test resources can be achieved. The problem of designing bistable circuits that employ sequential kernels has been previously investigated and several BIST TDMs have been proposed to reduce the hardware overhead associated with these circuits [2,3,4,5]. The technique in [5], called BIBS (Built-In test for Balanced Structures) TDM, is an approach that achieves low area overhead, low performance degradation, and high fault coverage by allowing only balanced bistable sequential kernels. To achieve high fault coverage in testing balanced bistable sequential kernels, special TPGs need to be designed. Procedures for designing TPGs that guarantee high fault coverage for balanced bistable sequential kernels have been presented in [5].

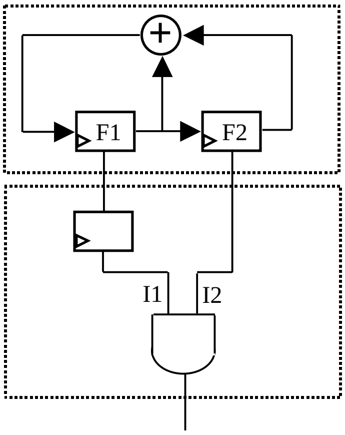
In this paper, we show that TPGs for balanced bistable sequential kernels can be efficiently designed based on the well-known retiming technique [6]. The paper begins by giving a brief description about balanced bistable sequential kernels, their properties, and test requirements. A motivation example illustrating the need for new TPG designs for such kernels is then demonstrated. After that, the application of the retiming technique in the design of the TPGs is described. Finally, experimental results are given to show that the TPGs designed using the retiming technique can achieve significantly higher fault coverage than the traditional LFSR TPGs.

2. Balanced Bistable Sequential Kernels

A synchronous sequential kernel *K* is said to be balanced BISTable [5] if it is (1) acyclic, (2) for any two vertices (say *v1* and *v2*) in its circuit graph, all directed paths (if any) from *v1* to *v2* are of equal sequential length, and (3) there does not exists a pair of input and output ports in *K* that is directly driven by and drives a common register. It has been shown in [7] that all detectable single stuck-at faults in a balanced circuit are single pattern testable. This single test pattern is applied to the circuit and the circuit is clocked for a number of clock cycles allowing the fault effect to propagate to a primary output. A fault is *1-step testable* [5] if it can be detected by a test sequence where the value for each bit position in the sequence is specified as either 0 or 1 in at most one vector in the sequence. Thus, a 1-step testable fault can be detected by applying a single test vector to the circuit for a number of clock cycles. In other words, such a fault is detectable by a sequence of test vectors formed from skewing some bits of this single test vector. Testing a circuit that is not 1-step testable is much harder than a 1-step testable circuit [7,8]. This is because testing such a circuit requires sequences of test vectors to be applied to the circuit to ensure high fault coverage. It has been shown in [5] that all single stuck-at faults in a balanced BISTable kernel are 1-step testable.

3. Motivation Example

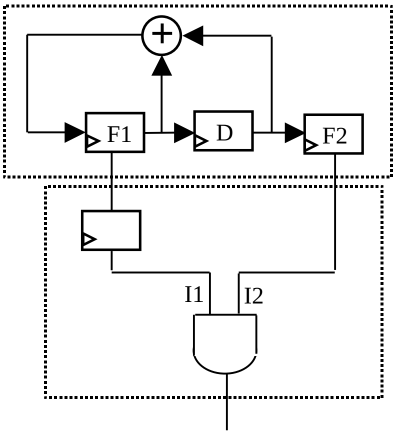
To illustrate the need for special TPGs for testing balanced bistable sequential circuits, let us consider the circuit shown in Figure 1(a). The circuit is a balanced



F1	F2	I1	I2
1	0	X	0
1	1	1	1
0	1	1	1
1	0	0	0

(a) (b)

Figure 1 (a) Testing a balanced sequential kernel using regular LFSR. (b) Test patterns generated and applied to kernel.



F1	D	F2	I1	I2
1	0	0	X	0
1	1	0	1	0
0	1	1	1	1
1	0	1	0	1

(a) (b)

Figure 2 (a) Testing a balanced sequential kernel using modified LFSR. (b) Test patterns generated and applied to kernel.

bistable circuit. In order to test this circuit under the BIBS TDM, a TPG is connected to its inputs, which is in this case a maximal-length LFSR. The sequences generated by the LFSR and the vectors applied to the inputs of the AND gate are shown in Figure1(b). As can be seen, regardless of the number of cycles applied to the LFSR, only the vectors {00, 11} will be applied to the inputs of the AND gate. This is due to the unequal sequential lengths of the paths from the LFSR inputs to the AND gate. Thus, only the stuck-at-0 faults on the AND gate will be detected and none of the stuck-at-1 faults can be detected by the LFSR. To gurantee full coverage of all the detectable faults in the kernel, it has to be ensured that all the paths from the TPG to every combinational block in the kernel have equal sequential length. A simple solution to achieve this is to insert registers in the kernels to make all paths of equal sequential length [5]. However, this would add significant area overhead and the circuit performance may be adversely affected. A simpler and much more efficient solution is to modify the TPG to achieve this goal. To compensate for the imbalance in the sequential lengths of the two paths from the LFSR to the AND gate, we need to add one sequential element along the path between *F2* and *I2*. The same balancing effect can be achieved by adding one D-type flip-flop (D-FF) before *F2,* as shown in Figure 2(a). Considering the test vectors generated by the modified LFSR and the test vectors applied to the inputs of the AND gate, shown in Figure 2(b), we can see that all the faults in the AND gate will be detected. The area overhead of this approach is much smaller than modifying the kernel itself and no additional performance degradation is introduced apart from the LFSR circuitry. The addition of this extra D-FF to the LFSR design to achieve the balancing effect can be easily achieved based on the retiming technique. Figure 3 illustrates the use of the retiming technique in designing the modified LFSR for this example. The TPG and the kernel to be tested are modeled by a weighted directed graph as shown. To make all the paths from the TPG to the AND gate have equal sequential length, we need to insert a sequential element along the path from *F2* to *I2*. This is indicated by assigning a weight of 1 on the edge representing the path in the graph, as shown in step1.

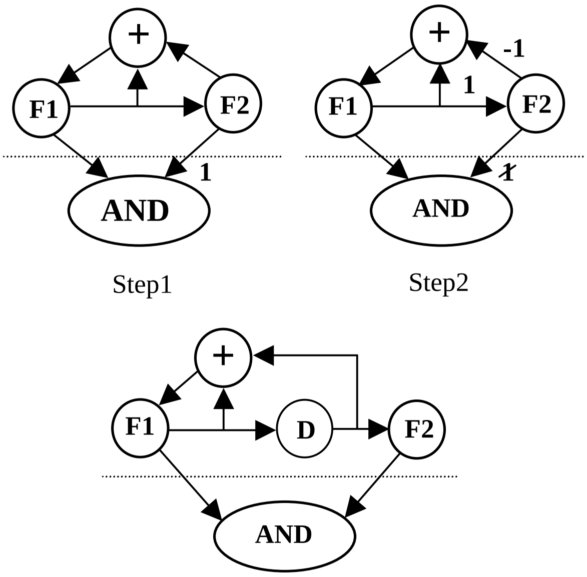


Figure 3 Steps illustrating TPG design based on retiming.

Then, we remove this sequential element from the kernel by retiming it across *F2*. This requires borrowing a sequential element from the connection of *F2* to the XOR gate and results in an extra sequential element in the input of *F2,* as shown in step2. The presence of a –1 weight on the edge connecting *F2* and the XOR gate

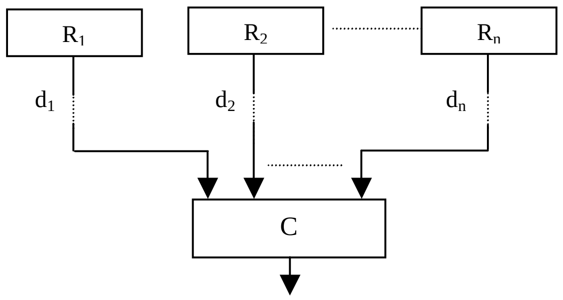


Figure 4 Ageneralized representation of a balanced bistable sequential kernel with a single logic cone.

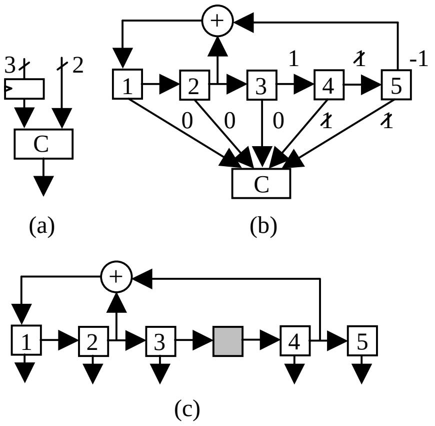
implies taking the connection to the XOR gate from the input of *F2* instead of its output, as shown in step3. This process results in the TPG design shown in Figure 2(a), which achieves full coverage of all the faults in the AND gate.

4. Design of TPGs Based on Retiming

A balanced bistable sequential kernel having a single cone of logic can be represented generally as shown in Figure 4 [5]. In this kernel, C represents the combinational logic block that feeds the output and d_1, d_2,…, and d_n are the sequential lengths of the paths from R_1, R_2, …, and R_n, respectively, to C. Each path in Figure 4 can be a data path between R_i and C, or a set of parallel data paths between R_i and C all having the same sequential length. In our work, we assume that the TPG will be designed based on a type 1 LFSR [9]. Let *G=(R, X, C, E, w)* be a weighted directed graph, where *R*, *X*, and *C* represent vertices in the graph, *E* represents edges, and *w* represents weights on edges. Each $r \in R$ represents a flip-flop in the LFSR TPG, X represents an XOR gate, and C represents the combinational logic cone. Each $e \in E$ connecting vertices *(v1, v2)* represents an interconnection in the TPG if v1 and v2 $\in \{R, X\}$. Otherwise, it represents a path between the TPG and the Combinational logic cone. The weight of an edge *w(e)* represents the number of sequential elements along the path or connection corresponding to this edge. The algorithm for TPG design for balanced bistable sequential kernels based on retiming is shown below.

Algorithm: *Retiming-Based TPG Design for Balanced Bistable Sequential Kernels*
Input: *G=(R, X, C, E, w)*
Output: *Designed TPG*

```
1. Let maxw= MAX{ w(e(r, C)) } for each $r \in R$.
2. For each $r \in R$
       w(r, C) = maxw – w(r, C).
3. For each $r \in R$
       Retime w(r, C) across r such that w(r, C) becomes 0.
4. Allocate a flip-flop for each $r \in R$.
5. Allocate an XOR gate to implement the characteristic polynomial of the LFSR.
6. For each edge e=(v1, v2) such that v1 and v2 $\in$ {R, X} and w(e)>0, allocate a number of FFs equal to w(e) connecting the nodes corresponding to v1 and v2 sequentially as shift registers.
7. For each edge e=(v1, v2) such that v1 and v2 $\in$ {R, X} and w(e)=0, make a direct connection between the nodes corresponding to v1 and v2.
8. For each edge e=(v1, v2) such that v1 and v2 $\in$ {R, X} and w(e)<0, make a direct connection between the node preceding v1 by |w(e)| sequential elements and v2.
```



(a)

(b)

(c)

Figure 5 A balanced bistable sequential kernel and its TPG design.

To illustrate the retiming-based TPG design algorithm, let us consider the example given in Figure 5. Given the kernel in Figure 5(a), a graph is constructed and the maximum weight on any path from the TPG to C is determined, which is 1 in this example. Then, the weights on the edges are computed as indicated in step2. The retiming of each node is then performed as indicated in step 3 and the result of this step is shown in Figure 5(b). Finally, the designed TPG is constructed, based on steps 4-8, as shown in Figure 5(c). Figure 6 demonstrates the application of the proposed algorithm on another balanced bistable sequential kernel. As can be seen the algorithm is simple and its correctness is guaranteed since it is based on the well-known retiming technique.

5. Experimental Results

In order to illustrate the effectiveness of the TPGs designed for balanced bistable sequential kernels, we have performed experiments on some of the ISCAS85 and ISCAS89 benchmark circuits. For each balanced bistable sequential kernel, we have applied a convential TPG based on a maximal-length LFSR, and a modified TPG designed based on the proposed algorithm. We have used a configuration similar to the one shown in

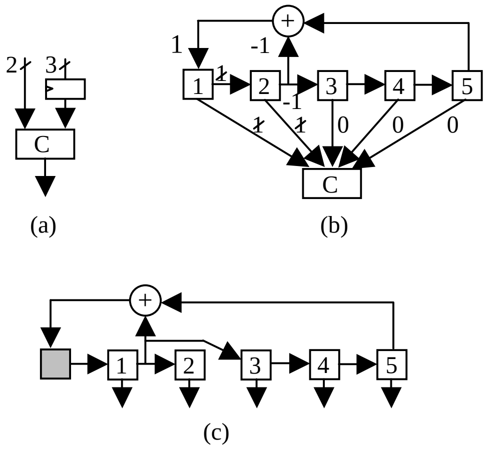


Figure 6 Another balanced bistable sequential kernel and its TPG design.

Figure 5(a). For the ISCAS85 cirucits, the first half of the primary inputs (rounded up) has a sequential length of 1 and the the second half has a sequential length of 0. For the full-scanned versions of the ISCAS89 circuits, the first set of primary inputs (obtained from making the circuit full-scanned) has a sequential length of 1. These inputs are equal to the number of FFs in the original circuit. The original primary inputs in the circuit have a sequential length of 0. Table 1 shows the fault coverage obtained for each case. As can be seen from the table, for all the circuits, the modified TPGs achieve higher fault coverage than the original TPGs based on the convential LFSR. In the results obtained in Table 1, we have used the primitive polynomials given in [10]. Furthermore, the LFSR seeds used are a 1 in the most significant position followed by 0's. We have also performed experiments on the benchmark circuits using a configuration similar to the one shown in Figure 6(a). We have observed that for this configuration the convential TPGs achieve fault coverages similar to the modified TPGs in most of the cases. The results of this experiment are not included due to space limitation. It is worth mentioning here that the modified TPGs guarantee exhaustive patterns applied to the kernel under test if a maximum-length LFSR is used and a maximum-length sequence is applied. However, this is not the case if the convential LFSR is used.

Table 1 A comparison between conventional and modified TPGs for testing balanced bistable sequential kernels.

		Fault Coverage				
	#vec.	*64*	*256*	*1024*	*4096*	*16000*
c17	*Conv.*	92.9	92.9	92.9	92.9	92.9
	Mod.	100	100	100	100	100
c3540	*Conv.*	54.9	84.7	91.8	92.3	92.5
	Mod.	57.4	86.5	95.1	95.8	96.1
c432	*Conv.*	57.1	87.0	98.2	98.2	98.2
	Mod.	63.9	80.4	99.3	99.3	99.3
s1488	*Conv.*	53.1	69.0	71.8	72.0	72.0
	Mod.	62.6	86.3	96.7	99.9	100
s1494	*Conv.*	52.6	68.3	71.0	71.2	71.2
	Mod.	62.1	85.6	95.9	99.1	99.2
s344	*Conv.*	92.7	97.3	97.6	97.6	97.6
	Mod.	94.6	99.7	100	100	100
s526n	*Conv.*	71.8	79.5	84.4	91.9	94.6
	Mod.	78.2	89.1	95.5	99.2	100
s820	*Conv.*	35.7	68.1	88.4	94.2	97.6
	Mod.	32.0	65.5	87.2	93.4	99.7

6. Conclusions

The BIBS TDM is a BIST methodology that can achieve high fault coverage with low area overhead and low performance degradation by restricting the circuits under test to be balanced bistable sequential kernels. In this paper, we have illustrated the efficient use of the retiming technique in designing TPGs for balanced bistable sequential kernels. Experimental results on ISCAS85 and ISCAS89 benchmark circuits demonstrate the effectiveness of the designed TPGs in achieving higher fault coverage than the convetional TPGs based on maximum-length LFSRs.

Acknowledgment

The authors would like to thank King Fahd University of Petroleum & Minerals for support.

References

[1] B. Konemann, J. Mucha, and G. Zwiehoff, "Built-In Logic Block Observation Technique," *Proc. of Int. Test Conference*, pp. 37-41, 1979.

[2] A.Krasniewski and A. Albicki, "Automatic Design of Exhaustively Self-Testing Chips with BILBO Modules," *Proc. of Int. Test Conference*, pp. 362-371, 1985.

[3] A.Krasniewski and S. Pilarski, "Circular Self-Test Path: A Low-Cost BIST Technique for VLSI Circuits," *IEEE Trans. Computer-Aided Design, pp. 46-55, Jan. 1989*

[4] M. S. Abadir, J. Newman, D. D'Souza, and S. Spencer, "Partitioning Hierarchical Designs for Testability," *Proc. of Int. Test Conference*, pp. 174-183, 1991.

[5] S.-P. Lin, S. K. Gupta, and M. A. Breuer, "A Low Cost BIST Methodology and Associated Novel Test Pattern Generator," *Proc. Eurobean Design and Test Conf.*, pp. 106-112, 1994.

[6] C. E. Lieserson and J. B. Saxe, "Retiming Synchronous Circuitry," *Algorithmica*, vol. 6, pp. 5-35, 1991.

[7] R. Gupta, R. Gupta, and M. A. Breuer, "BALLAST: A Methodology for Partial Scan Design," *Proc. of Int. Sympos. On Fault-Tolerant Computing.*, pp. 118-125, 1989.

[8] D. Vir Das, S. C. Seth, and V. D. Agrawal, "Estimating the Quality of Manufactured Digital Sequential Circuits," *Proc. of Int. Test Conference*, pp. 210-217, 1991.

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

[10] P. Bardell, W. H. Mcabbey, and J. Savir, *Built-In Test for VLSI: Pseudorandom Techniques.* John Wiley & Sons, 1987.