ISSN (Print) : 0974-6846
Indian Journal of Science and Technology, *Vol 9(29), DOI: 10.17485/ijst/2016/v9i29/90874, August 2016* ISSN (Online) : 0974-5645

Efficient Test Sequence Generator for Area Optimization in LFSR Reseeding

D. S. Nivetha Joice* **and S. Saravanan**

School of Computing, SASTRA University, Thanjavur - 613401, Tamil Nadu, India;
joiceece@gmail.com, saran@core.sastra.edu

Abstract

In this paper, a new concept of test pattern generator is used with Seed Initialization Method (SIM) for area optimization in LFSR reseeding. LFSR Reseeding is the method which is mainly used in logic BIST. Some of the reseeding mechanism needs some sort of memory to store all seeds. But this issue is overcome by present method. By this method, the proposed test pattern generator employs the output response of the CUT to the LFSR as the controlling signal to transform the LFSR state. It contains a net providing cell and LFSR with reversal logic to conform that there is no storage of seeds. When compared to previous methods, by using this approach (SIM), the test sequence that will have been given to LFSR is much reducing. The empirical outcome on ISCAS circuits shows that the conferred seed initialization method has brought reduced area overhead.

Keywords: BIST, LFSR Reseeding, Seed Initialization Method, Test Sequence Generator (TSG)

1. Introduction

Build-in self-test is the efficient method for testing complicated circuits. It incorporates some test module to circuit-under-test creating the way of testing the circuit. Some methods say about different ways to generate test pattern. Those techniques are Cellular Automata (CA), Linear Feedback Shift Register (LFSR), and Single Bit Change Sequence Generator (SBCSG). In this method, LFSR is the prominent device for generating pseudo random patterns in BIST, because of its simplicity and less hardware[1]. But this scheme, does not give satisfactory result. So to resolve this problem, further move onto several techniques. Test pattern compression is determined by appropriate clustering technique and equivalent decompression technique. This technique includes compression and decompression achieved using LFSR reseeding[2].

The weighted random pattern testing[3], here some methodology is used for producing deterministic patterns and then it need to be modified us weight sets. And this may require some memory to store all weights. Here the drawback is each deterministic pattern can result in various weight sets. This leads to large number of weights. The mixed mode technique[4] uses some other logic to transform pseudo random pattern to deterministic pattern. This leads to additional hardware overhead, due to this additional hardware, it leads to increase number of gates. In order to overcome this storage problem, paper[5] used the circuit response as the test patterns, but not all the states of LFSR can be obtained. LFSR reseeding scheme is efficient and it is decrease of test pattern. Proposed encoding scheme determine to decrease the size of test facts[6].

In circular self test scheme[7], this scheme includes some additional logic that leads to circular chain, which

* *Author for correspondence*

cause some states to skip and particular pattern will be generated. But this additional logic may lead to area increase. In mapping logic[8], is used to get required pattern, with increased area. The next BIST scheme[9], says that the output of the CUT is used as control signal for producing both pseudo and deterministic patterns. Various tap connection is used in LFSR technique. This work related by common LFSR and customized LFSR by seed selection method. Various input patterns are produced by this scheme and it decreases the switching bits[10]. The number of nets which that get from CUT can be easily reduced by using net sharing procedure[11]. In case more number of nets is used or outputs of the CUT, area overhead will be more and also it is necessary to use one multiplexer for each input port in order to use several nets. In recent paper, this reseeding scheme[12] includes some inversion logic that is added with the LFSR architecture, then by giving some outputs of CUT to this inversion logic the LFSR states will be changed for reseeding purpose. In the proposed system, seed initialization method is used, with this approach the input is given to some part of CUT and the random patterns which are generated from this, is given as input to next part of CUT. So by this method the area overhead will be reduced.

2. Proposed Test Sequence Generator (TSG)

This method consists of Test Sequence Generator (TSG) and control block as shown in Figure 1. The Test Sequence Generator (TSG) consists of net providing cell and the LFSR with XOR gates and the CUT output response will have given as the input to net providing logic and depending on the output of the control block the Test Sequence Generator (TSG) will get vary. The main purpose of the XOR gates is to act as reversal logic for modifying LFSR states. By giving appropriate signal to the reversal logic, any particular bits in the pattern can be altered in order to produce a new seed. Suppose the 5 bit input pattern is given to LFSR, it will produce only 32 combinations and repeat it and rebound to the first pattern and will follow the same. If the reversal logic gets some input from the control block, the continuous sequences of patterns will get destructed and the new pattern will be generated according to the input of reversal logic, so the outcome pattern is called as next new seed. In order to provide all mandatory seeds, different output response from the CUT is to be used. The individual set of net is called configuration.

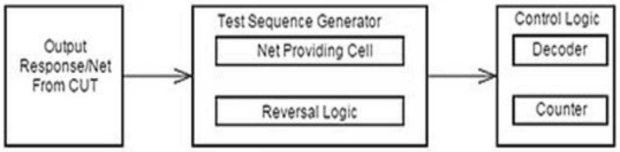

Figure 1. Basic architecture.

The control block consists of counter and the decoder. The counter is used to carry the present pattern index and the decoder is used to give all necessary control signals to transform between various configurations with respect to the pattern index. The main advantage here is that, all the nets from the CUT can be collected into the net providing cell, and also all the nets can be shared among the different inputs. When compared with the previous work, all the inputs need one multiplexer for to select different net, in this multiplexer based BIST the area overhead will be very large. This problem is eliminated by the introduction of reversal logic into the LFSR.

In Figure 2, proposed Test Sequence Generator (TSG) is used. Here main disadvantage is, the pre-computed seeds can be generated, in order to include some extra seed, multiplexer can be included in front of flip-flop. The multiplexer included architecture is new Test Sequence Generator (TSG). This include some extra pin seed_EN is added to one input of multiplexer and an AND gate is added to each output of net providing cell and another new pin is added, that is control_EN this input is used to control all added AND gates. The beneficial part is that only two additional pin is needed for new Test Sequence Generator (TSG). The working condition for this is, if both seed_EN and control_EN is 0, then it will execute pseudo-random testing. Then by setting control_EN to 1, the BIST scheme will be performed, and in order to shift new seed into the BIST, the seed_EN should be set to 1. And the further output response can be obtained by setting seed_EN to 0. By the introduction of multiplexer to this proposed test sequence generator will results in area overhead.

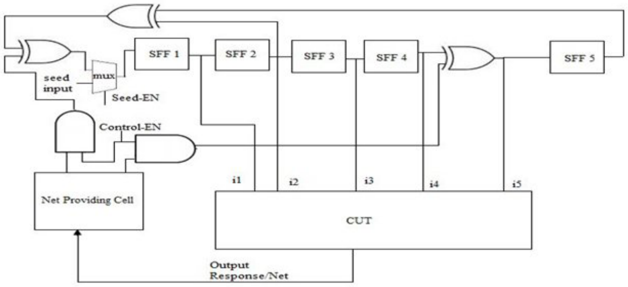

Figure 2. Proposed test sequence generator.

2.1 Seed Determination Method

In 2010, Lien and Lee[9] developed seed determination procedure that is the seed is determined by using specified algorithm that will get polynomial function f (x), and the fault list F, and some input parameter L as the input and provides set of seeds as output. This algorithm will find the seed one after another. First, the user defined pattern is to be described, with X bits and include this pattern to TS it will save all patterns produced from initial seed. And all the faults generated from this particular sequence will be removed. The count value used to indicate the number of patterns produced from first seed.

Let name the user defined pattern as P1, according to the polynomial of LFSR the next pattern will be generated name it as P2, and next as P3 let's assume the patterns goes up to P10. All the patterns which are generated will contain some X bits, because user defined pattern may contain x bits, so do X-filling process for better fault coverage, and the update the P1 and also update all the patterns generated from P1 and certainly drop all the detected faults. And the counter is used here because the X filling process is not the successive one, the count value will get increased. Once the count value will be equal to input parameter L, next step is to stop the further generation of random patterns and designate a pattern for next seed. The significant point is that, any one pattern generated from the used defined seed and that is not used for better fault coverage can be altered to generate a new seed by use of reversal logic. The particular pattern which is modified to new seed and further patterns generated from the particular pattern is getting removed from the TS. And all the X bits are to be modified according to the seeds.

3. Seed Initialization Method (SIM)

The previous algorithm mainly focused on generation of seeds, due to more seed generation more number of nets need to be used, so there will be area overhead. In order to eradicate this problem, seed initialization method is used. Let's consider this example, the 40 bit input CUT, it obviously need 40 bit LFSR for Random Pattern Generation (RPG). Instead of using 40 bit LFSR, this algorithm says to use only 5 individual 8 bit LFSR. An advantage is that, the previous methods uses 40 bit input sequence for 40 bit LFSR, here input sequence length is too high, but by using Seed Initialization Method, it is possible to give only 8 bit input sequence to LFSR and the random patterns which are generated by this 8 bit input sequence can be given as input for further CUT inputs. By doing this, the register storage size will be less and area overhead also reduced. The pseudo code is given below for this method.

Pseudo Code for Seed Initialization Method

```
        {* algorithm
        N               : no of seed is generated
        Sequence        : user defined input
        Reseeding       : LFSR reseeding generated by portioning
        in              : clk, reset, seed sel, seed in
        *}
        {* Random Generation D1,D2,D3 . . . . *}
                Patterns are updated by seed generation
                LFSR_in <= (D1(user defined input))
                Rising edge of clk
                D2 <=Random patterns from D1
                Assume D1 to Dn Generated
                Output response get from CUT
                Repeat --------> until all seeds generated
endwhile
```

Here, the LFSR reseeding is generated by portioning method and numbers of patterns are updated by seed generation technique. Output response is get from Circuit Under Test (CUT).

4. Experimental Result

The Table 1 shows the results for proposed Test Sequence Generator on ISCAS 89benchmark circuits. The experimental result shows, the comparison of area for both seed determination method and Seed initialization method. Table 2 shows that C3540 benchmark circuit is compared with GE method[4] and area is reduced by 13%. Area is compared with seed determination and seed initialization method. C3540 benchmark circuit is reduced 24.2% of area, S5378 benchmark circuit is reduced up to 37.80% of area and S1512 benchmark circuit is reduced 12.52% of area with comparing seed initialization method.

Table 1. Area comparison for various benchmark circuit

Benchmark circuit	**Area**	
	Seed determination algorithm	**Seed initialization method**
C3540	2070	1554
S5378	5160	3239
S1512	3078	2693

Table 2. Area comparison of C3540

Benchmark circuit	Comparison of Area	
	GE	Proposed work
	Method[4]	(Proposed)
	(existing)	
C3540	1804.5	1554

5. Conclusion

The LFSR with reversal logic and Seed initialization Method are combined together for significant reduction of area for BIST architecture. And also it is noted that seed generation complexity is very much reduced. The performance criteria for proposed test sequence generator have been proved by experimental result. Area is reduced by 13% in C3540 Benchmark circuit by comparing with existing method.

6. References

1. Jamal K, Srihari P. Analysis of test sequence generators for built-in self-test implementation. IEEE International Conference on Advanced Computing and Communication Systems; 2015 Jan. p. 1–4.
2. Saravanan S, Charaphani K, Selvakumar P. Cluster based LFSR reseeding for test data compression. Research Journal of Applied Sciences, Engineering and Technology. 2012 Jul; 4(14):2120–5.
3. Reeb B, Wunderlich HJ. Deterministic pattern generation for weighted random pattern testing. VLSI European Design and IEEE Test Conference; 1996 Mar. p. 30–6.
4. Fagot C, Gascuel O, Girard P, Landrault C. A ring architecture strategy for BIST test pattern Generation. ATS '98. Proceedings. Seventh Asian Test Symposium, 1998; 2003 Dec. p. 418–23.
5. Kalligeros E, Kavousianos X, Bakalis D, Nikolos D. On-the-fly reseeding: a new reseeding technique for test-per-clock BIST. Journal of Electronic Testing. 2002 Jun; 18(3):315–32.
6. Saravanan S, Upadhyay HN. Effective LFSR reseeding technique for achieving reduced test pattern. Research Journal of Applied Sciences, Engineering and Technology. 2012 Nov; 4(22):4783–6.
7. Touba NA. Circular BIST with state skipping. IEEE Transactions Very Large Scale Integration Systems. 2002 Oct; 10(5):668–72.
8. Krasniewski A, Pilarski S. Circuilar self-test path: a low-cost BIST technique for VLSI circuits. IEEE Transactions Computer Aided Design for Integrated Circuits and System. 1989 Jan; 8(1):46–55.
9. Lien WC, Lee KJ. A complete logic BIST technology with no storage requirement. IEEE Asian Test Symposium; 2010 Dec. p. 129–34.
10. Saravanan S, Sowmiya G, Premalatha P, Rajaram A, Sai RV. Design and analysis of scan power reduction based on liner feedback shift register reseeding. International Conference on Information and Communication Technology; 2013 Apr. p. 638–41.
11. Lien WC, Hsieh TY, Lee KJ. Routing-efficient implementation of an internal-response-based BIST architecture. Proceedings VLSI Design, Automation and Test Symposium; 2012 Apr. p. 1–4.
12. Lien WC, Lee KJ, Hsieh TY. Efficient LFSR reseeding based on internal-response feedback. Journal of Electronic Testing. 2014 Dec; 30(16):673–85