IARJSET

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

ICRAESIT - 2015

B V Raju Institute of Technology, Narsapur

Vol. 2, Special Issue 2, December 2015

A Novel Block Switch Logic For State-Skip Test Pattern Generation In Built In Self Test Scheme

Y.Sreenivasula Goud[1], Dr. B.K.Madhavi[2]

Professor, RCEW, Kurnool, AP, India[1]

Professor, SEWC, Hyderabad, Telangana, India[2]

Abstract: In the process of testing VLSI circuits, it is required to test the entire operation with minimum all patterns. However for complete fault analysis total test patterns derived were quite large. This large number of coefficient representation results in large time of testing and hence consumes more power for testing operation. It is required to optimize this power consumption, based on the pattern optimization. In the approach of pattern optimization, state skip modeling was proposed in recent. In such coding approach, the process of pattern optimization using LFSR optimization was developed. In this paper a optimization of test patterns based on block switch modeling is proposed. The approach of pattern alignment using a synonymous approach of genetic mutation is proposed. The test evaluation of the suggested work is carried over ISCAS 89 benchmark circuits. The fault tolerance, recursion overhead, and logical implementation are observed to be optimized for the proposed approach.

Keywords: Test Pattern optimization, Genetic coding, built in self test, Block mutation

I. INTRODUCTION

cent developments in the field of very-large-scale integration (VLSI) testing have led to a demand for fault diagnosis in large scale integration for logical and functional verification. A large number of failures have been observed during the manufacture of integrated circuits (ICs); as a result, it becomes very tedious in testing such failure individually. Thus, failures are evaluated based to the effect of logical faults on the functionality of a circuit, enabling the construction of logical fault models. High fault coverage is basically valuable during the production of ICs; such coverage is obtained using approaches such as Design for Test (DFT) and Automatic Test Pattern Generation (ATPG). Built-In-Self-Test (BIST), a DFT methodology, has large advantages for the testing schemes due of its testability, speed of testing and isolation of the automatic test equipment used. The Built-In Self-Test plays an important role in testing of complex and large circuits [2]. The BIST build a self-testing circuit by combining the Circuit Under Test (CUT), a Test Pattern Generator (TPG), a Test Response analyzer with a BIST controller on a chip. Low test performing time and hardware overhead with low performance degradation are required in many BIST applications. In the modeling of conventional system on chip (SoC) for testing operation, the power dissipation is of major concern. The overhead observed in the pattern generation is directly effective to test pattern generation process. In the controlling of power dissipation in [3] an approach of multiplier-accumulator pair model were used to compensate the power consumption in VLSI testing circuitry. An optimal test pattern development for a real time testing over a cellular automation system was defined in [4]. Towards the complex test circuitry logic testing, in [5] a BIST logic using LFSR logic was suggested. The process of simplifying the test logic based on the applied test vector was developed. The LFSR logic toward optimal test pattern generation was defined in [6]. The testing circuit was evaluated for circuit logic with high transition densities. The fault dependency for testing is the main limitation of such approach. Towards optimizing the power consumption in test vector generation, in [7] a state-skip model was suggested. This a recent approach for optimal test pattern generation based in the controlling of state transition controlling of the LFSR logic. The suggested approach defines a small test circuit for generating of test vector in addition to the polynomial units used in pattern generation logic. In the modeling of such unit is defined in 2 mode of operation, called the normal mode or the state skip mode. In the process of normal mode of operation, the polynomial feedback logic was used wherein in the state skip mode; a jump of constant length is made to skip states in the follow up. The approach of test pattern optimization based on the LFSR logic, were used for testing of the embedded units functioning in the state skip mode. In the approach of such coding large volume of test vectors is used indicating test vector compression, which reduces the test vector usage in test set embedding in the IP cores. The state skip logic links the gap between the data compression and the test embedding process in testing digital circuits. Towards the optimization of test vector generation, a built in test (BIT) was developed which, make use of multiple polynomial linear feedback logic in deriving the test pattern generation was suggested in [8]. In the approach of such multi-polynomial LFSR logic, the connecting seed values are of greater importance due to its specified memory logic. a two fold LFSR logic to minimize the test pattern generation overhead was suggested in [9].

IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology
ICRAESIT - 2015
B V Raju Institute of Technology, Narsapur

Vol. 2, Special Issue 2, December 2015

The pattern generation process minimizes the pattern generation overhead, however the sequence of test pattern generation are limited. In recent past intelligence logic were observed to be incorporated in test pattern generation. In [13] a genetic modeling for test pattern generation was defined, genetic based modeling were observed to be effective in test pattern generation, as it minimizes the search space point in the large varsity of test patterns. The optimization of the test pattern generation based on the pattern optimization was defied by the binary encoding string. The string optimization approach was effective in genetic based coding. However in the approach of test pattern generation, the approach of genetic mode is observed be optimal under string optimization. The approach of mutation and string alignment could lead to an optimization in test pattern generation. Wherein state skip logic reduces the number of test pattern generation states, genetic model can overcome the issue of randomness in test vector generation. In this paper an improvement in the test pattern generation called block switching approach is proposed. The model optimizes the code stream generation based on various block swapping formulating the process of mutation in the pattern generation approach. To present the suggested approach this paper is defined in 6 sections. Wherein section 2 defines the conventional model of test pattern generation, section 3 define the proposed approach of block switching logic for test pattern optimization, section 4 present the observed results for the developed system. Section 5 outlines the conclusion of the stated approach.

II. PATTERN GENERATION AND OPTIMIZATION USING STATE-SKIP LOGIC [7]

Automatic Test Pattern Generation and Automatic Test Pattern Generator (ATPG), is an electronic design automation technology that is used to obtain an input or test sequence. When the test sequences are applied to a digital circuit, they enable automatic test equipment to differentiate between the correct circuit behavior and the faulty circuit behavior caused by defects.

The test operation of a digital circuit is validated via testing its operational functionality. The testing of such devices required large test patterns to validate its functionality under all possible test condition. To generate these test patterns automated test pattern generator (ATPG) is used. These test patterns when applied over the test circuit under test develops test values are validated via a response analyzer. To declare the functionality of the test circuit, it is needed to test for all valid test patterns. However large test patterns gives a testing of same test values which then formulate a redundant testing. Hence such patterns need to be minimized so as testing overhead could be reduced. Towards test pattern optimization, test pattern generation or test pattern selection are used. Wherein generation is obtained via architectural designing of ATPG unit, test pattern selection are made based on fault testing algorithm. To generate a test pattern LFSR logics are basically used. The pseudorandom nature of the test pattern result in the robustness of the testing and validity under dynamic condition. For an optimal generation of test pattern in [9] a two fold state skip logic was presented for optimizing the operation of LFSR logic. the suggested state-skip model reduces the test patterns based o the functional mode operation of the CUT. The selected test vectors are stored as a seed counter. These seed counters are used to derive the optimal test pattern in twofold state-skip modeling. Wherein the conventional LFSR logics are operated based on the defining polynomial, the state-skip model is developed based on the instruction of jumping states. These jumping state leads to generation of the validated test data only, hence reducing the testing overhead. Based on the method used, the LFSR approach move from a standard mathematical model to a modular form. Moving from a modular form to a dual standard form is performed via inverting the orientation of all of the data flows between latches, and removing the XORs between the latches and performing an XOR on all of the feedbacks into the first latch. The resultant state sequence is presented in Table 1.

Table.1.: Lookup table with cycles for i=0,1,2 and 3.

Cycle (Y)	X0	X1	X2	X3
0	1000	0100	0010	0001
1	0100	0010	0001	1100
2	0010	0001	1100	0110
4	1100	0110	0011	1101
8	1010	0101	1110	0111

In the modular LFSR, the XOR gates are placed between adjacent latches. It is observed that the modular LFSR can process faster than the standard LFSR. In Table 1, X_0, X_1, X_2, and X_3 represent the bit positions (X); 0, 1, 2, 4, and 8 are the cycle counts (Y); 1000, 0100, 0010, 0001,. . ., 1010, 0101, 1110, and 0111 are shifted between the states.
A lookup table with cycles for i = 0, 1, 2, and 3 is derived from the sequences generated from the LFSR with the characteristic polynomial $P(X) = X^4 + X + 1$, as shown in Fig. 1.

IARJSET

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

ICRAESIT - 2015

B V Raju Institute of Technology, Narsapur

Vol. 2, Special Issue 2, December 2015

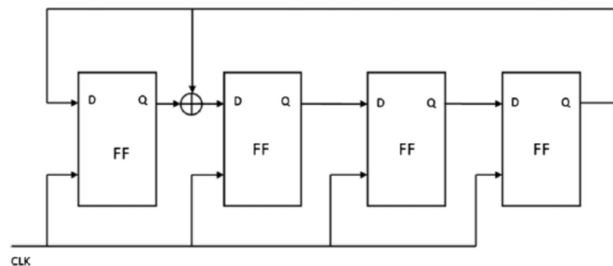

Fig.1. LFSR used in the TFSS logic [9]

For a skip-state (N0) estimation, all the cycle elements are in binary sequence and added to the remainder of the modulo (2^n-1) division. For example, the cycle rows 2 and 8 are selected from Table 1, which results in hex 10. Each bit set is identified in the skip state (N0). For individual bits in N0, the state S_{fc}-row (first cycle-row) for each single bit set in Si is calculated from the lookup table.

Similarly, for each bit set in the state model, the S_{fc}-row, state S_{nc}-row (next cycle-row) is computed. In the final step, all of the S_{lc}-row (last cycle-row) states are XORed to determine the nth state for bit N0. When all of the bits in the skip states are completed, a XOR operation is performed for all of the nth states of all of the bits to determine the machine's n^{th} state. The resultant states were then used for the LFSR architecture.

For example, a segment of Cycle 2 (Y_2) is obtained from the lookup table defined as; {0010, 0001, 1100, and 0110}.

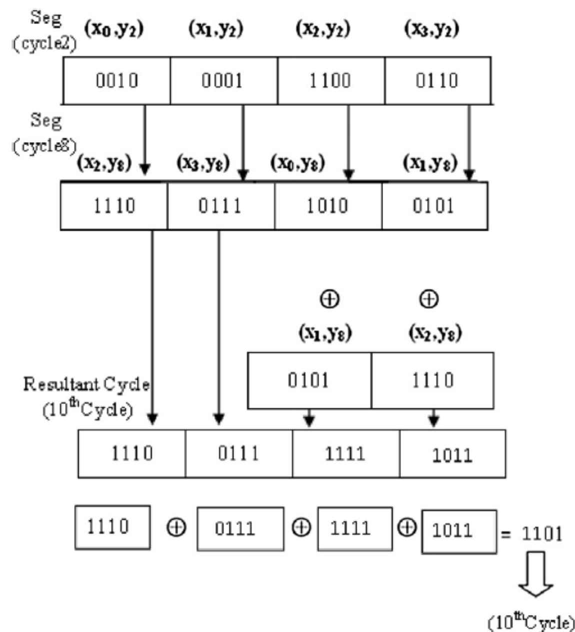

Fig.2. Illustration of the TFSS operation [9]

In X_0, Y_2, the second bit holds '1' and the other bits hold '0'. At this stage, the next cycle is estimated asX_2, Y_8. In the other case of X_2, Y_2, the first and second bits hold '1' and the others hold '0'. Using the same estimation routine,X_1, Y_8 and X_2, Y_8 are estimated from the lookup table. The resultant column vector represents the skip state N0 (Cycle Count N = 10). However, the main problem with the conventional TFSS approach, is the pattern generation time. The time taken for the LFSR based pattern generation time is considered to be larger for state transition model, which is to be minimized

III. OPTIMIZED BLOCK SWITCH LOGIC PATTERN GENERATION

For the optimization of the test pattern generation, a block switching logic is proposed. This logic is inspired by the functional logic of switching model in test pattern optimization. The approach of combining the patterns to derive new

.Copyright to IARJSET

IARJSET

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

ICRAESIT - 2015

B V Raju Institute of Technology, Narsapur

Vol. 2, Special Issue 2, December 2015

patterns termed as switching is referred for optimizing the sequence switching operation. In this approach, a new automatic test pattern generation and optimization was proposed based on the distribution of test patterns to reduce power dissipation and memory utilization. In this approach of block switching, a searches for a best sequence by applying combination of blocks for the available test patterns is developed. This approach re-align a set of pattern bits, and each realigned bit pattern is then computed for fitness value.

The fitness function is designed to reduce the switching pattern. Initially, a set of test patterns were generated and a combination of the test patterns were derived, the switching and new patterns are applied for generation of the optimal test patterns by evolving a fitness function. The fitness function provides a quantification of the quality of the pattern. It is the fitness of the pattern that determines whether the pattern will be selected to produce offspring and quantifies its chance for survival among the other pattern in the set to the next generation. The fitness function is problem specific. In this paper fault simulation with fault is with fault dropping is used in order to evaluate the test vectors. The score given to each individual is equal to the number of fault it detects, the fitness function forgiven test vector is calculated by eq.(1) given below:

$$f(x) = \frac{no.of\ deteced\ \ faults\ \ by\ test\ \ vector}{total\ \ no.of\ faults} \qquad (1)$$

The test pattern optimization based on the switch pattern alignment was defined in three step of operations, 1. Initialization, 2. Block switching and 3. Convergence of best patterns.

a. Initialization:

In this step, the test vectors are created randomly on feasible space using LFSR logic. The patterns are generated so as the size leads to large enough in order to ensure adequate diversity; however, it is a tradeoff between getting higher convergence rate with larger search space and less switching operation. The test pattern size increases with increasing test vector length under consideration, which intern leads to higher time for convergence. To obtain a Stabilization of block length and computation overhead a 16 bit pattern, with 4 bit pattern each as block is considered. Using these patterns then the block switching operation were carried out.

b. Creating a New block patterns:

New patterns were created by repeating the following steps until:

A. Selection:

The switching algorithm uses selection operator to simulate natural evolution of the test pattern. The test pattern with high fitness is inherited to the next generation with greater probability. Usually, test pattern with high fitness are selected for block switching to converge faster to best solution. patterns with high fitness is not been selected for switching to prevent the danger of diverting from good solutions in the search space. Therefore, patterns with low fitness are usually selected for switching. All selection methods are based the fitness of patterns. The disadvantage of selecting patterns with high fitness is the probability of less diversity in the search space. Therefore, pattern with low fitness is usually selected for switching. All Selection methods are based on the fitness of the pattern. In this approach, the wheel selection approach based on probability coding is used to select the individuals.

d. Wheel Selection coding:

Patterns are selected according to their fitness. The better the patterns are, the more chances to be selected they have. By using wheel where pattern are placed in all patterns in the set, every pattern has its place big accordingly to its fitness function.

Pattern with bigger fitness will be selected more times. This can be simulated by following algorithm.

1. **[Sum]** Calculate sum of all pattern finesses in set - sum ***S***.
2. **[Select]** Generate random number from interval ***(0,S)*** - ***r***.
3. **[Loop]** Go through the set and sum finesses from ***0*** - sum ***s***. When the sum ***s*** is greater than ***r***, stop and return the pattern.

B. Block switching:

Block switching is the key to switching algorithm, power that is to exchange corresponding switching properties from the two patterns, to allow useful genes on different patterns to combine in their offspring. Most common Block switching types are one-point, two-point, uniform Block switching. In this paper, as shown in fig (3), two-point Block switching is used.

IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology
ICRAESIT - 2015
B V Raju Institute of Technology, Narsapur

Vol. 2, Special Issue 2, December 2015

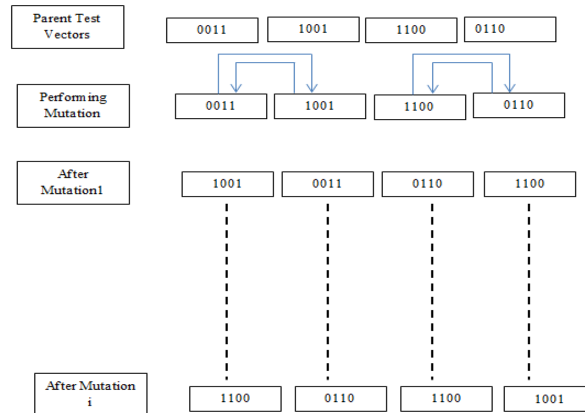

Fig.3. Two-point Block switching

C. Switching: After a Block switching is performed, switching takes place. This is to prevent falling all solutions in set into a local optimum of solved problem. Switching changes randomly the new offspring. For binary encoding we can switch a few randomly chosen bits from 1 to 0 or from 0 to 1. It says how often will be parts of pattern mutated. If there is no switching, offspring is taken after Block switching (or copy) without any change. If switching is performed, part of pattern is changed. If switching probability is 100%, whole pattern is changed, if it is 0%, nothing is changed. Switching is made to prevent falling the algorithm into local extreme, but it should not occur very often, because then the algorithm will in fact change to random search.

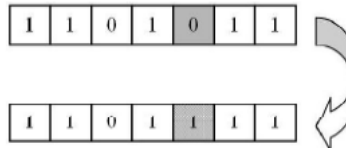

Fig.4. Switching in binary string

After obtaining new test vectors the TSFF was applied on the new test vectors to detect the faults. The implementation the algorithm before TFSS increases the fault detection accuracy and also reduces the power consumption.

IV. RESULT OBSERVATIONS

For the evaluation of the developed approach, a test evaluation on test circuit pattern is developed. The combination of the test pattern blocks are presented in table 1.

Table.2: pattern optimization using GA approach

Pattern	Number of Cycles
B1B2B3B4	8
B1B3B4B2	10
B1B4B2B3	12
B1B2B4B3	6
B2B3B4B1	9
B2B4B1B3	8
B2B4B3B1	11
B2B1B3B4	6
B3B4B1B2	7
B3B1B2B4	5
B3B2B1B4	10
B3B4B2B1	12
B4B1B2B3	9
B4B2B3B1	11
B4B2B1B3	7
B4B3B1B2	12

.Copyright to IARJSET

IARJSET

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

ICRAESIT - 2015

B V Raju Institute of Technology, Narsapur

Vol. 2, Special Issue 2, December 2015

The number of block combination and its obtained test cycles are presented in table 2. For each of the test pattern combination, a fitness function is derived, via fitness evaluation operation, defined by equation 1. The optimal value of the fitness is derived from the roulette wheel approach, and the best fitness block combination is used.

Table.3: Power Consumption Analysis

Circuit	**Power (μW)**		
	SSS-LFSR	**TFSS-LFSR**	**GSS-LFSR**
S27	287.36	281.53	275.23
S298	891.30	874.52	862.28
S420	608.32	604.25	600.84
S526	1572.56	1570.24	1566.28

The power consumption analysis is derived from the x-power analyzer of Xilinx tool. The developed approach is targeted on to SPATRAN-E series. The logical combination of the test pattern usage is focused. The power consumption for the testing of the test circuit is defined in table 3. Due to lower logical test pattern for optimization, the test factor for the test circuit is also less. This optimization result in faster and lower overhead in test circuitry.

Table.4: Synthesis Report

PERIOD	**Time Computation (ns)**		
	SSS-LFST	**TFSS-LFSR**	**GSS-LFSR**
Minimum Period	3.017	2.548	2.224
Maximum Period	7.352	4.040	3.852

The operational speed of process is also derived targeting on to the Xilinx device. The obtained process time is observed to be 1.8 ns faster for the developed approach as compared to the conventional GSS-LFSR logic. The test pattern optimization in such case is observed to be faster and optimal in testing operation.

V. CONCLUSION

This paper outlines an approach for test pattern optimization based on genetic approach. The developed approach illustrates an improvement in speed of operation and lower power consumption due to less power optimization process. The approach of the power consumption analysis is derived from the x-power analyzer of Xilinx tool. The developed approach is targeted on to SPATRAN-E series. The logical combination of the test pattern usage is observed to be more optimal in case on proposed GSS-LFSR logic. Operational speed of process is also derived targeting on to the Xilinx device. The obtained process time is observed to be 1.8 ns faster for the developed approach as compared to the conventional GSS-LFSR logic.

REFERENCES

[1] Abramovici M, Breuer MA, Friedman AD., Digital systems testing and testable design. New York (NY): Computer Science Press; 1990.
[2] Bardell PH, McAnney WH, Savir J., Built-in test for VLSI: pseudo-random techniques. New York (NY): Johh Wiley & Sons; 1987.
[3] Zorian, Psarakis, Paschalis, Kranitis, Gizopoulos., Low power/energy BIST scheme for datapaths. In: VLSI test symposium; 2000. p. 23–8.
[4] Corno F, Rebaudengo M, Reorda MS, Squillero G, Violante M., Low power BIST via non-linear hybrid cellular automata. In: 18th VLSI test symposium;2000. p. 29–34.
[5] Zorian Y, A distributed BIST control scheme for complex VLSI devices. In: VLSI test symposium. Digest of papers. Eleventh annual IEEE; 1993. p. 4–9.
[6] Wang S, Gupta SK., DS-LFSR: a new BIST TPG for low heat dissipation. In: Proceedings of the international test conference; 1997. p. 848–57.
[7] Tenentes V, Kavousianos X, Kalligeros E., Single and variable-state-skip LFSRs bridging the gap between test data compression and test set embeddingfor IP cores. IEEE Trans Comput-Aid Des IntegrCircSyst 2010;29(10):1640–4.
[8] Hellebrand S, Rajski J, Tarnick S, Venkataraman S, Courtois B. Built-in test for circuits with scan based on reseeding of multiple-polynomial linearfeedback shift registers. IEEE Trans Comput 1995;44(2):223–33.
[9] Savir J, McAnney W. A multiple seed linear feedback shift register. IEEE Trans Comput 1992;41(2):250–2.
[10] Koenemann B. LFSR-coded test patterns for scan design. In: Proceedings of European test conference, Germany; 1991. p. 237–42.
[11] Barnhart C, Brunkhorst V, Distler F, Farnsworth O, Keller B, Koenemann B. OPMISR: the foundation for compressed ATPG vectors. In: Proceedings ofinternational test conference; 2001. p. 748–57.
[12] Girard P, Landrault C, Moreda V, Pravossoudovitch S. An optimized IST test pattern generator for delay testing. In: Proceeding of very large scaleintegration test symposium; 1997. p. 94–100.
[13] MrsRachnasingh, Dr.ArvindRajawat, "Implementation of Genetic Algorithm forAutomatic Test Pattern Generation", International Journal of Scientific & Engineering Research Volume 3, Issue 4, April-2012.