

Vector Generation using Spectral Methods

Ayoush M Dixit

Electrical and Computer Engineering Department

Auburn University, Auburn, AL-36849

Email: dixitam@auburn.edu

Abstract

Two new test generation algorithms for combinational and sequential circuits have been proposed. Test vectors are generated using characteristic faults and spectral information embedded in a circuit under test (CUT) in the form of Hadamard coefficients for the circuits. The Hadamard coefficients are extracted using input and output correlation for combinational circuits and using a test vectors targeted for a small set of faults in the circuits for sequential circuits often known as characteristic faults.

1. Introduction

Due to the advancement of the technology in terms of device density and clock speed, testing of the devices have posed a threat to design engineers. Market drive and feature requirements supported by devices have led designers to use complex designs that are hard to test. Researchers in the past have come up with different algorithms and innovative ways to test devices but the increment in device complexity has rendered those methods insufficient if not useless. Test equipments to test a device are falling short on keeping pace with the increasing operating speed of the device preventing “at speed” testing [2]. At-speed testing is crucial for reduction of test time which is the reason Built-in-self-test (BIST) solutions have become part of designs these days.

BIST solutions comprise of test pattern generator (TPG) that generates the vector sequence to be applied to the circuit under test (CUT), response compactor that compacts the responses obtained from the CUT into a signature and an output response analyzer (ORA) that compares the signature to the signature of the fault free circuit stored in the device.

In most of the BIST schemes, TPGs use linear feedback shift register (LFSR) to produce pseudo-random patterns. They have a simple structure and can also be used as output response analyzers, thereby serving dual

purpose [1]. The quality of the LFSR-generated test set depends on the CUT [3]. Random technique results in large test sets [2] and is useful for circuits without random-pattern -resistant faults [3].



Figure 1. Basic Architecture of BIST

Weighted random patterns have been found to yield better fault coverage in circuits that contain random-pattern-resistant faults [4][5]. The basic idea for weighted pseudo random test pattern generation method is to bias the probability at each input based on the information gathered on the circuit [6]. Weighting the pseudo-random patterns [7-8] is done using counter-based schemes [9-10] or performing bit-fixing (pattern mapping)[11]. However, the difficulty arises when no one set of weights may be suitable for all faults [12], and some structures in logic circuits are resistant even to weighted random testing [14]. Many hardware pattern generators often round-off optimal weights producing patterns that are sub-optimal for certain circuits [6].

Dynamic and static vector compaction methods have been in use to compact the generated test vectors for sequential circuits. The dynamic vector compaction uses heuristics aimed at producing a short test sequence whereas static vector compaction is a post-processing step which is independent of test generation process and uses reordering and overlapping of subset of test sequences [20-21], vector omission (omits vectors from a sequence without reducing coverage) [22], vector addition and removal of old vector [16], vector restoration (vector omission and then restoration to restore coverage) [23] to obtain shortest test sequence. Various other methods of extension of vectors have been proposed in literature. Some use both static compaction and sequence extension techniques that are achieved by repeating perturbed input

vectors or subsequences included in the compacted sequence [22], whereas some use spatial and temporal correlations among test vectors [24-26]. In [16-23], information about the functional properties and/or structure of circuit under test was not used. In [6][12][27][28] researchers have used spectrum based test generation algorithms. The main motivation of spectral methods is to use the circuit information in generating set of test vectors able to give better fault coverage.

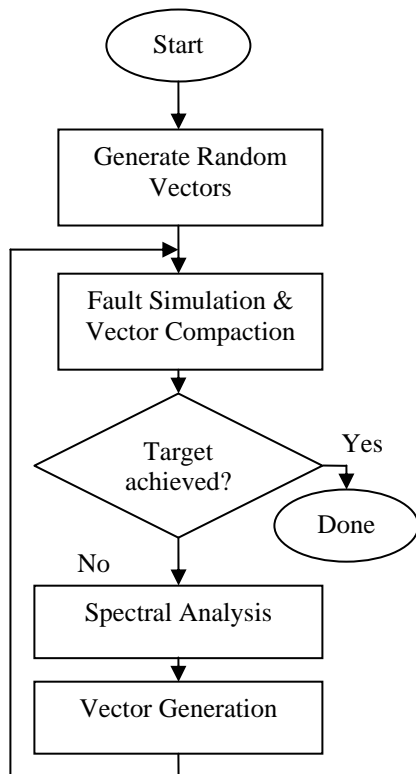


Figure 2. Test Generation Scheme

In this paper two new algorithms have been presented to generate test vectors using spectral information from CUT. These two algorithms will be useful for test generation for combinational circuits and sequential circuits as well. First algorithm is specifically targeted to combinational circuits where the relation between system inputs and outputs are used while generating the new sets of vectors. First algorithm finds its inspiration from [31]. In [31] circuit is labeled faulty after performing comparison of the Walsh coefficients generated after the inputs are applied with the stored Walsh coefficients of the good circuit. Second algorithm is a modification of the algorithm presented in [6] where the authors use state relaxation as this approach saves storage requirements [6]. Fault masking can occur when removing a recurrent subsequence [29]. It is also possible that removing a relaxed recurrent subsequence will mask

a fault [30]. For the stated reasons in [29-30] state masking presented in [6] is omitted in generating the vectors in this paper.

The remainder of the paper is organized as follows. Section 2 gives the overview and motivation for the new spectral methods of vector generation. Section 3 discusses the details of extraction of spectral characteristics and the two algorithms in light of their importance. Section 4 concludes the paper. Section 5 gives some insights on the future work and direction that would be done if given more time.

2. Overview and Motivation

Every signal can be represented in two domains, 1) time domain, and 2) frequency domain. The transformations from one domain to the other can be done using Fourier Transform. Depending on what we want to do, one is preferred to the other. Sometimes it is easier to perform calculations in time domain and sometimes it is easier to do so in frequency domain. Most of the time we have seen digital circuits be represented in time domain and it is always a good idea to try something new and thus the reason for trying the frequency domain for testability analysis of a circuit.

Frequency decomposition is the most commonly used technique in signal processing. A signal can be projected to set of independent waveforms that have different frequencies. This set of waveforms, each represented as a vector, forms a basis matrix. It is a known fact that if an orthogonal transform which has high energy compaction and de-correlation properties is used, most of the energy of the original signal will be compacted into a few high magnitude coefficients. If the input data is corrupted by additive white noise, components which correspond to noise will be distributed among low magnitude high frequency components. Hence, a thresholding algorithm in an orthogonal transform domain will remove most of the noise components.

Due to the properties of Hadamard matrix it is widely chosen as a projection matrix while extracting spectral characteristics of a signal. A Hadamard matrix is an $n \times n$ real matrix H which satisfies $HH^T = nI$ where I is an $n \times n$ identity matrix. They contain only 1s and -1s, and can be generated using the following recurrence relation:

$$H(x) = \begin{bmatrix} H(x-1) & H(x-1) \\ H(x-1) & -H(x-1) \end{bmatrix}$$

where $x=1, 2, \dots, n$,
 $H(0) = 1$,
 $n=\log_2 N$

For example, with $k=1$ and $k=2$, the above equation yields

$$H(1) = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

$$H(2) = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}$$

Since there are only 1s and -1s in the matrix, multiplication can be easily done using additions and subtractions. Moreover, the inverse transform of a Hadamard matrix is the same as forward transform so reconstruction of vectors becomes straight forward for our case.

Each row/column in a Hadamard matrix is a basis vector, carrying a distinct frequency component. Taking example of $H(1)$, the two basis vectors are $[1 \ 1]$ and $[1 \ -1]$. Any bit sequence of length of 2 can be represented as a linear combination of these basis vectors.

In this paper, the vector generation process starts with random vectors then static compaction for latest detected faults is performed to filter out noise unlike [23] where the all the detected faults are considered. Because of the promising results of using spectral methods in vector generation in [6, 12, 27-28] spectral methods are used in extraction of circuit and input sequence spectral characteristics. The spectral characteristics are extracted from the compacted test sequence using Hadamard transform and a cutoff is used as a threshold to eliminate noise (higher the cutoff, closer the new set to the original set) which produces newer test sets. This new set is added to the test set based on the spectrum.

3. Spectral Characteristic Extraction

a. Combinational Circuits

Input and output relation is targeted by this method in generation/construction of vectors based on the spectral information identified for the circuit. Algorithm 1 illustrates the construction of such vectors.

Algorithm 1:

Perform logic simulation with a set of random vectors and get the outputs.

*Let i denote the row of the set vectors.
 Let j denote the column of the set of vectors.
 Let k denote the column of the output for v_{ij} .
 Let v_{ij} denote each input.
 Let o_{ik} denote each output.*

```

For (each output  $o_{ik}$ )
  For (each input  $v_{ij}$  in the test set)
    Intermediary product  $IP_{ij} = v_{ij} \times o_{ik}$ 
  For (each  $IP_{ij}$ )
    Coefficient vector  $c_{ij} = H \times IP_{ij}$ 

For (each value in the coefficient matrix [ $c_{00}, \dots, c_{nn}$ ])
  If (absolute value of coefficient < cutoff)
    Set the coefficient to 0
  Else if (coefficient < 0)
    Set the coefficient to -1
  Else
    Set the coefficient to 1
For (each  $i$ )
  Extension vector  $e_{ij} = \text{modified } c_{ij} \times H$ 
If (weight > 0)
  Extend the vector set with value 1
Else
  Extend the vector set with value -1

```

The illustration of this algorithm is as follows:

Let $F(x_1, x_2, x_3) = x_1 * x_2 + x_2 * x_3 + x_1 * x_3$

x_1	x_2	x_3	F
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Replacing 0 with -1

x_1	x_2	x_3	F
-1	-1	-1	-1
-1	-1	1	-1
-1	1	-1	-1
-1	1	1	1
1	-1	-1	-1
1	-1	1	1
1	1	-1	1
1	1	1	1

Intermediary product,

$x_1 * F$	$x_2 * F$	$x_3 * F$
1	1	1
1	1	-1
1	-1	1
-1	-1	1
-1	1	1
1	1	1
1	-1	-1
1	-1	1

Obtaining transform coefficients using $x_1 * F$,

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ -1 \\ -1 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 4 \\ 0 \\ 0 \\ -4 \\ 0 \\ 4 \\ 4 \\ 0 \end{bmatrix}$$

Choosing cutoff of 4 the modified coefficient becomes [1 0 0 -1 0 1 1 0]. Now, multiplying the new coefficient with H(3) it yields [2 2 2 -2 -2 2 2 2]. Using weight of 2, and conversion the new vector becomes [1 1 1 0 0 1 1 1]. Similarly this is done for $x_2 * F$ and $x_3 * F$ and the new vectors are obtained.

b. Sequential Circuits

Instead of targeting the entire fault list to extract spectral information like in [12], “characteristic faults” is targeted by this algorithm. Because sequences that detect hard to detect faults generally detect many other faults as well, the aim of the second algorithm is on identifying the set of hard faults such that their detection will detect other hard to detect faults. Characteristic fault should have the property that a test set which detects them is also able to detect many other faults in the circuit. This algorithm is similar to the one discussed in [6] but it extracts spectral information without the use of state relaxation.

Algorithm 2:

Si set of faults that have similar spectral characteristics as K

$T =$ generate random vectors
 $i = 0$; $done = 0$; $S_{max} = 0$;

While (not *done*) and ($i < \text{max_iteration}$)
 Fault simulate T_i ;
 $k_j =$ k last-detected faults;
 $c_j = T_j$ filtered for k_j ;
 Perform Hadamard Transform on C_j ;
 Obtain spectral coefficients H_j ;
 $E_j =$ new vectors generated using coeff. H_j ;
 $S_j =$ faults detected by E_j ;
 if $S_j > S_{max}$
 Characteristic fault set = K_j ;
 $S_{max} = S_j$;
 $i++$;
 If no improvements for n iterations
 Done = 1;

Hadamard transform is done similarly as in previous algorithm.

4. Conclusion

Two new algorithms have been proposed for vector generation in testing combinational and sequential circuits. Extraction of input and output spectral characteristic of a circuit, which has been overlooked in the past, using Hadamard transform has been proposed. It is projected, through the literature reviews that the approaches presented in the paper will result in generation of test vectors able to give better coverage.

The vector generation methods discussed here can be very efficiently used in system-on-chip (SOC) environment. As discussed in [27], these algorithms can be implemented as BIST in SOC as follows:

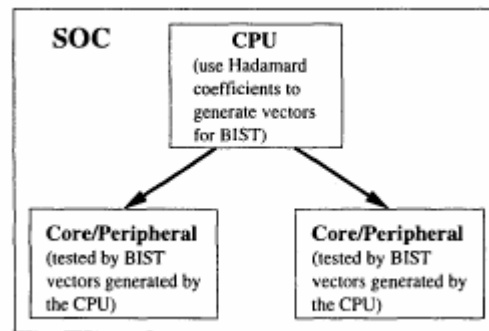


Figure 3. Use of Hadamard coefficients for BIST. [27]

The algorithms presented in this paper, in the light of previous literary and experimental works, looks promising. No matter how promising they sound or feel, the algorithms need to be validated and the future work should be targeted towards developing methods to test these algorithms on benchmark circuits.

5. Future Work and Direction

Due to limited time during the semester I was unable to produce some simulation results for the algorithms presented in this paper.

Algorithm 1 tries to characterize the circuit by using input and output relation to generate the spectral coefficients which then is used to generate test vectors for simulation purposes. This is a novel approach that was seen in earlier paper [31] but has not been in literature for generating the test vectors for simulation purposes. Simulation based test generation, which began with the advent of random test generation which soon gave way to weighted random patterns, is an efficient way to test designs before they are manufactured. Although the methods presented in this paper may be algorithms that will go un-noticed and un-cared for but the spectral methods of generating vectors looked promising after reviewing [6][12][27][31]. Since Algorithm 1 is proposed for combinational circuits and circuit simulation may be required many times while generating the vectors, I plan to write my own simulator which would save a whole lot of simulation time as that would mean I will not have to run the simulator and spectral coefficient extractor separately. Once that is done along with extraction of spectral coefficient and the new vectors are added to the randomly selected small set that started the process, and I would check for the results. It is clearly evident that for circuits with larger number of inputs and outputs this method will be time consuming if I was using two different simulators. If I have my own simulator and can do the manipulations in a structured way the simulation time will be reduced, probably orders of magnitude (humans work slower than machines!).

Algorithm 2 is very suitable for sequential circuits and is one of the algorithms that was proposed and experimented in [6] but without state relaxation. State relaxation means removal of subsequences (sequences that transfers a state machine from one state to another) from test vectors to reduce the number of vectors. This is one of the ways compaction of vectors are done in simulation based testing. If fault effects of on flip flops are unchanged due to application of the sequences then those sequences can be removed from the vector sets but sometimes the result may not change due to masking of a fault. In [30] authors predict that the fault coverage may

be slightly lower due to fault masking although these situations are rare. I did not want to let go of the idea of testing the circuits without state relaxation as at times the widely accepted facts without supporting data can be misleading. Implementation of this algorithm would also mean coming up with a simulator for sequential circuits as we need to operate on last detected fault on every cycle to find out a new set which will take up a long time (longer than the one for combinational circuits). Given more time I would work on a simulator that is able to handle both Combinational Circuits and Sequential Circuits as well with options like using state relaxation and without using state relaxation with my algorithms.

Doing so will give me some experimental results that can be publishable. Another thing that I had in mind was to use a FPGA board and implement these algorithms in it to try and see the effectiveness in SOC environment. Rapid prototyping is a very important aspect of design these days and FPGA can emulate a SOC environment in labs if not to the full extent.

I am pretty sure my algorithms will produce results that are equally good as the ones in [6][12][27][28] if not better. All these projections are based on the literature review of the algorithms and the experimental results made available by many authors.

References

- [1] B. Könenmann and J. Mucha and G. Zwiehoff, "Built In Test for Complex Digital Integrated Circuits," *IEEE J. Solid-State Circuits*, vol. SC-15, no. 3, June 1980, pp. 315-318.
- [2] M. Abramovici, M. A. Breuer and A. D. Friedman, *Digital System Testing and Testable Design*, New York, NY: Computer Science Press, 1990.
- [3] V. D. Agrawal, "When to use random testing", *IEEE Trans. on Computers*, vol. C-27, No. 11, pp. 1054-1055, Nov. 1978.
- [4] M. F. Alshaibi and C. R. Kime, "Fixed-biased pseudorandom built in self test for random pattern resistant circuits," *Proc. Intl. Test Conf.*, 1994, pp. 929-938.
- [5] F. Muradali, T. Nishada, and T. Shimizu, "Structure and technique for pseudo random-based testing of sequential circuits," *Journal of Electronic Testing: Theory and Applications*, pp. 107-115, Feb. 1995.
- [6] Xiaoding Chen and Michael S. Hsiao, "Characteristic Faults and Spectral Information for Logic BIST," *Proc. IEEE/ACM ICCAD*, November 2002.
- [7] H. J. Wunderlich, "Multiple distributions for biased random test patterns," *Proc. Fault-Tolerant Computing Symposium*, pp. 258-263, 1988.
- [8] F. Muradali, V. K. Agarwal and B. Nadeau-Dostie, "A new procedure for weighted random built-in self-test," *IEEE Int. Test Cons.*, pp. 660-669, 1990.
- [9] D. Kagaris and S. Tragoudas, "Generating deterministic unordered test patterns with counter," *IEEE VLSI Test Symp.*, pp. 374-379, 1996.
- [10] S. B. Akers and W. Jansz, "Test set embedding in a built-in self-test environment," *IEEE Int. Test Cons.*, pp. 257-263, 1989.

- [11] M. Chatterjee and D. K. Pradhan, "A novel pattern generator for near-perfect fault coverage," *IEEE VLSI Test Symp.*, pp. 417-425, 1995.
- [12] Ashish Giani, Shuo Sheng, Micheal S. Hsiao, and Vishwani D. Agrawal, "Efficient Spectral Techniques for Sequential ATPG", *Proc. Design, Automation and Test in Europe*, March 2001, pp. 204-208.
- [13] M. Bershteyn, "Calculation of multiple sets of weights for weighted random testing," *ITC*, pp. 1031-1040, 1993.
- [14] E. Eichelberger, E. Lindbloom, J. Waicukauski, T. Williams, *Structured Logic Testing*, Prentice Hall, Englewood Cliffs, New Jersey, 1991.
- [15] M. H. Schulz, E. Trischler, and T. M. Sarfert, "SOCRATES: A Highly Efficient Automatic Test Pattern Generation System," *IEEE Transactions on Computer-Aided Design*, vol. 7, pp. 126-136, January 1988.
- [16] S. Kajihara, I. Pomeranz, K. Kinoshita, and S. M. Reddy, "On Compacting Test Sets by Addition and Removal of Test Vectors," *VLSI Test Symposium*, April 1994, pp.202-207.
- [17] P. Goel and B. C. Rosales, "PODEM-X: An Automatic Test Generation System for VLSI Logic Structures," *Proceedings of the 18th ACM/IEEE Design Automation Conference*, pp. 260-268, June 1981.
- [18] S. Kajihara, I. Pomeranz, K. Kinoshita, and S. M. Reddy, "Cost-Effective Generation of Minimal Test Sets for Stuck-at Faults in Combinational Logic Circuits," in *Proceedings of the 30th ACM/IEEE Design Automation Conference*, pp. 102-106, June 1993.
- [19] S. T. Chakradhar and A. Raghunathan, "Bottleneck Removal Algorithm for Dynamic Compaction and Test Cycles Reduction," in *Proc. European Design Automation Conf.*, September 1995.
- [20] R. K. Roy, T. M. Niermann, J. H. Patel, J. A. Abraham and R. A. Saleh, "Compaction of ATPG-Generated Test Sequences for Sequential Circuits," *Proc. Intl. Conf. on Computer-Aided Design*, pp. 382-385, Nov. 1988.
- [21] B. So, "Time-Efficient Automatic Test Pattern Generation Systems", Ph.D. Thesis, EE Dept., Univ. of Wisconsin at Madison, 1994.
- [22] R. Guo, I. Pomeranz and S. M. Reddy, "A fault simulation based test pattern generator for synchronous sequential circuits," *Proc. VLSI Test Symposium*, pp. 260-267, April 1999.
- [23] I. Pomeranz, S. M. Reddy, "Vector restoration based static compaction of test sequences for synchronous sequential circuits", *Proc. ICCD*, pp. 360-365, Oct. 1997.
- [24] A. Jain, V. D. Agrawal, M. S. Hsiao, "On generating tests for sequential circuits using static compaction," *Intl. Test Synthesis Workshop*, March 1999.
- [25] S. Sheng, A. Jain, M. S. Hsiao, and V. D. Agrawal, "Correlation analysis of compacted test vectors and the use of correlated vectors for test generation," *IEEE Intl. Test Synthesis Workshop*, March, 2000.
- [26] A. Giani, S. Sheng, M. S. Hsiao, and V. D. Agrawal, "Correlation-based test generation for sequential circuits," *Proc. IEEE North Atlantic Test Workshop*, pp. 76-83, 2000.
- [27] A. Giani, S. Sheng, M. S. Hsiao, and V. D. Agrawal, "Novel spectral methods for built-in self-test in a system-on-a-chip environment", *Proc. IEEE VLSI Test Symposium*, April 2001, pp. 163-168.
- [28] G. Kasturirangan, M. S. Hsiao, "Spectrum-based BIST in complex SOCs," *Proc VTS 2002*, pp. 111-116, April 2002.
- [29] M. S. Hsiao, E. M. Rudnick, and J. H. Patel, "Fast algorithms for static compaction of sequential circuit test vectors," *Proc. IEEE VLSI Test Symp.*, pp. 188-195, Apr. 1995.
- [30] M. S. Hsiao, S. T. Chakradhar, "State relaxation based subsequence removal for fast static compaction in sequential circuits," *Proc. Design, Automation Test in Europe*, Feb. 1998 pp. 577-582.
- [31] A. K. Susskind, "Testing by Verifying Walsh Coefficients," *IEEE Trans. Computers*, vol. C-32, no. 2, Feb., 1983, pp. 198-201.