Pak. J. Biotechnol. Vol. 13 special issue II **(International Conference on Engineering and Technology Systems (ICET'16) Pp. 1-6 (2016)**

POWER AWARE ENTROPIC HIDDEN MARKOV CHAIN ALGORITHM FOR CODE BASED TEST DATA COMPRESSION

[1]*S. Rooban and* [2]R. Manimegalai

[1]Department of Electronics & Communication Engineering, Sri Krishna Institute of Technology, Chennai, India.
[2]Department of Computer Science & Engineering, Park College of Engineering and Technology, Coimbatore, India

ABSTRACT

Even though the scan architectures generally utilize advanced designs for testing reason, most of them remain an expensive design in test data volume and power consumption. A novel software based test data compression technique for testing System on Chip (SoC) is proposed in this paper. The proposed technique concurrently addresses the problem of reducing large test data volume and reduction of power consumption for scan testing on embedded Intellectual Property (IP) cores. In comparison to the aim of reducing only test data volume by recognizing the appearance of vector patterns and thereby eliminating don't-care bits using entropic Hidden Markov Chain (HMC) algorithm, here we address the task of decreasing power consumption. The proposed power proficient test data compression method is tested using the Verilog model of ISCAS'89 and ITC'99 benchmarks.

Index Terms: Automated Test Equipment, Circuit under Test, System on Chip, Built-in self-test, low power test, Test data compression

I. INTRODUCTION

Intellectual Property (IP) cores are currently commonly utilized as a part of large System on Chip designs. In general, IP cores posture two difficult test challenges. First, recent rise in design intricacy and the assimilation of embedded cores in SoC has prompted a massive growth in test data volume. Industry specialists predicted that this tendency will continue throughout the next couple of years [1-2]. Owing to the increase in volume of test data, design for testability and testability analysis plays a major role in testing of digital circuits and devices [3-4]. Second, the switching activity of IP cores is substantially higher amid test than that of functional mode. The resulting inordinate power consumption might bring about structural harm to IP cores or serious decline in reliability of the circuit under test [5-6]. High test data volume prompts an expansion in testing time. Additionally, vast test data volume might likewise surpass the limited memory depth of Automated Test Equipment (ATE). Multiple ATE reloads are tedious because data transfers from a computing terminal to the ATE memory or from the ATE memory to ATE channels are relatively moderate; the transfer time spans from minutes to hours [7]. Test application time can be decreased by using countless scan chains. The quantity of ATE channels which can candidly drive scan chains is restricted because of pin count constraints and ATE capabilities. To effectively test these systems, every intellectual property centre must be adequately practiced with a set of pre-computed test patterns given by the core vendor.

Test data compression presents assuring solution to the issue of expanding test data volume. A test set represented by T_D for the Circuit Under Test (CUT) is packed to a much lesser data set T_E, which is stored in the ATE memory. An on-chip decoder is utilized to regenerate T_D from T_E amid test application. Various test data compression techniques have been proposed for decreasing SoC test data volume [1, 8-11]. Results demonstrate that reductions up to 23% in test data volume and 20% in test application time are accomplished while applying this solution. Nonetheless, although it decreases drastically test data volume and test time, it doesn't consider test power usage when cramming doesn't care bits.

As a starting point, we have picked the solution presented in [1] as it offers a high-test data volume reduction factor. This technique is adjusted so that the don't-care bit assignment is done in an approach to decrease not only test data volume but additionally test power consumption. We demonstrate that we can significantly diminish power usage by adjusting the initial compression method while maintaining the same request of magnitude in test data volume reduction. The rest of the paper is sorted out as takes after. In the next section, we present the literature review of code based test data compression technique. Section 3 discusses the proposed methodology. Section 4 presents simulation based experimentation results obtained for ISCAS' 89 and ITC'99 benchmarked Verilog model [12] and for various industrial circuits to tackle issue of power consumption with this compression technique. Section 5 gives concluding comments.

II. RELATED WORK

In the context of scan testing, the issue of unwarranted power amid test is a great deal more serious as the application of every test pattern requires countless operations that contribute to superfluously expanding the switching activity [6]. High power consumption amid test might affect the circuit reliability but might likewise prompt its damage. Elevated test power consumption might be responsible for yielding noises and thus causing damages such as IR-Drop or Ground Bounce [5, 13].

Various studies have been conducted for the issues arise while reducing the testing time and test data volume for the core based system on chip. Recently, there has been a developing trend of hybrid SoC design i.e., SoC contains an embedded processor [14, 15]. The utilization of embedded processor has extended SoC design flexibility, diminished design cycle, and lengthened SoC product lifetime significantly by permitting gadgets to adapt with the changing standards and extra features [16]. Another real advantage of this type of SoC is that the embedded processor can be utilized to run software for decompressing the pre-compacted test data rather than utilizing exceptional decompression hardware circuitry. The compression process is utilized for compact the test data volume before downloading the data into the processor memory along with the decompression process. The processor executes small decompression software, and after that the test vectors are connected to individual cores in the SoC. It should be noted that compression of test data does not have any direct effect on the testing time. Compression of test data should be done well before downloading the test set into the embedded processor for testing. Notwithstanding, the compression technique must be sufficiently efficient not only to decrease the total test data volume, but additionally allows in avoiding unnecessary power usage.

Some other methods have been proposed to simultaneously address the issue of test data compression and power consumption. The technique presented in [17] depends on Golomb code and a particular assignment of don't-care bits to decrease test data volume and power consumption. The segmented addressable scan architecture presented in [18] adjusts the Illinois Scan Architecture [19,20] to decrease the power consumption. Hamzaoglu et al., have proposed the test data compaction technique [21], which reduces the volume of test data transferred from the workstation to the ATE. While these methodologies effectively diminish the volume of test data, they don't decrease the ATE bandwidth requirements. Embedded Deterministic Test (EDT) which is a novel test-data volume-compression methodology is proposed in [22]. It diminishes manufacturing test cost by reducing scan test data volume. But these methodologies might experience the ill effects of inefficient tester channel utilization. Iyengar et al., [9] have proposed another built-in plan for testing sequential circuits in view of statistical coding but this method is suitable only for circuits with little number of essential inputs.

The selective Huffman coding [23] eliminates the disadvantage of statistical coding. This method splits the test vectors into altered length input patterns of size *b* (where *b* is a square size), and applies Huffman coding. But these methods expand the computational complexity and large zone overhead. Following that the selective Huffman coding is not optimal, Kavousianos et al., have proposed the optimal Selective Huffman coding [24] with low hardware overhead and better compression. The Golomb coding allocates a variable-length Golomb code, size of group, to a keep running of '0's as depicted in [25]. But these Golomb codes are optimum only for a particular pattern distribution. A methodology called, Frequency directed run-length (FDR) coding is proposed in [26] form two parts, a prefix and a tail, where both parts have the same length. But the FDR code requires more complicated decoder with altered region overhead. Gonciari et al., [27] have proposed the variable-length Huffman coding. This method diminishes the ATE memory and channel capacity requirements by obtaining great compression ratios. Nourani et al., proposed RL-Huffman coding in [28].This method blends two encoding techniques to diminish power dissipation, test data volume and test pattern conveyance time in scan test applications. Eventhough this method is pertinent to the condition, the number of don't care bits is high. Nine code words used for test data compression technique is presented in [29]. Stuffing the bit column wise and reordering of vectors based on hamming distances proposes better solution in run length based data compression codes in [30]. One more technique which capitalizes on the fact that numerous consecutive squares of the test data can be combined together is presented in [31]. Compression is accomplished in this plan by storing the consolidated piece and the quantity of squares combined. These techniques offer a high reduction in terms of power consumption, i.e., from 70% to 80%. Unfortunately they offer only a little reduction of test data volume, i.e., from 2x to 5x. This paper focuses on reducing the power consumption and test data volume simultaneously by utilizing code based compression technique.

III. METHODOLOGY

A. Experimental Setup

Keeping in mind the end goal to exhibit the viability of the proposed test compression technique, independent simulations were conducted on different ISCAS 85 and ISCAS 89 HDL benchmark model of Verilog. An automatic test generation process is first utilized for obtaining a set of test vectors giving 100% fault scope [15]. The proposed compression algorithm is implemented in MATLAB programming environment on a Windows machine having Quad-center 2.5-GHz Ultra processor with 8GBs of RAM. On account of space constraints, only some partial results on ISCAS'89 and ITC'99benchmark circuits are given. All the test vectors required for testing the SOC are first compacted in software mode. The packed test vectors and an efficient decompression system are then stacked into the simulation of embedded handling core of the SOC. The processor executes the decompression process, and afterward applies all the uncompressed unique test vectors to every last center of SOC for generating and breaking down the output responses.

B. Model for Test Data Compression

In the proposed compression technique all the test vectors are divided into several blocks comprising of equal size, where the size of this blocks depends on the proportionality of bits in each vector. Here, the idea of compression is that primarily passed test vectors are considered as reference test vectors whereas the consequent vectors are generated from the weighted evaluation of the referenced one by storing those blocks/bits that exhibits difference with it. This will allow us to compress and recover the whole set of test vectors during the decompression process. The entropy of the test data is compressed using the estimated weighted factors obtained using the proposed algorithm. It is applied for compression on test data set as a matrix of size m x n. Here, each of the rows belonging to C is divided into blocks of equal size. Owing to the structural relationships among the faults there tends to be lots of similarities between test vectors. Hence, this test vectors can be ordered optimally such that the successive test vectors experience differences by just a fewer number of blocks. Thereby, reducing the amount of information is requisite for storing these differences will comparatively fall beyond the size required for storing the entire test vectors.

Let C_k, be the test vectors where k=1,2,3,…n of size m x n and N is the block size of data. Thus, the entropy of this test vector can be represented in equation 1.

$$E(C_k) = -\sum_{i=1}^{n} p_i * \log_2 p_i \qquad (1)$$

Where, $C_k = \{C_1, C_2, \dots C_n\}$ and $p_i = p_{i-1} + b_k$ is the index of bits where b_k is the weighted indices distances corresponding between the adjacent blocks of the test vectors. Since, the distribution changes with time, therefore it is necessary to use an encoding scheme with respect to the changing time. Thus, the distance frequency for the distribution of indexes needs optimum parameters for each C_k&b_k respectively. This correspond to the hard problem because of the increase in size relegates the necessity of general solution. Thus, for a given k distance of good vectors, the problem is that how to encode each section of the scan vectors and what is the best parameter for reducing the size of encoded data that is derived using the proposed Entropic Hidden Markov Chain Algorithm EHMCA. The data needs to be pre-processed or divided into scan lines of test vectors using equation 2.

$$C_k = \sum_{i=1}^{m} C_0(K_i) + \sum_{i=1}^{K_1} C_0(b_i) + \cdots + \sum_{i=K_{m-1}+1}^{K} C_m(b_i) \qquad (2)$$

In equation 2 $C_i(.)$ is the required size to encode the indices value at each section. Here the optimum weighted parameter is determined using the proposed EHMCA algorithm.

It is assumed that the P($C_K|C_{K-1}$) is independent of K, which leads to the definition of the stochastic transition matrix $K = \{K_{ij}\} = P(C_K = j|C_{K-1} = i)$

The initial state distribution of the primary test vector (i.e. when K=1) is given in equation 3.

$$\tau_i = P(C_1 = i) \qquad (3)$$

The observation variables $O_K$can take one of K possible values. The probability of a certain observation at K index for state j is given in equation 4.

$$b_i(o_K) = P(O_K = o_t|C_t = j) \qquad (4)$$

Taking into account all the possible values of O_K and C_K we obtain the weighted b_k i.e., indices distances corresponding between the adjacent blocks by N matrix $b_k = \sum_{j=1}^{K} b_j(o_K)$

An observation sequence is given by $O = (O_1 = o_1, O_2 = o_2, \dots, O_K = o_K)$

Thus we can describe a hidden Markov chain by initializing the computing with $\theta = (C, O, \tau)$ iteratively updating for local weighted maximum for $\theta^* = \arg\max P(O|\theta)$.

Algorithm: Entropic Hidden Markov Chain Algorithm (EHMCA) for Test Data Compression

Input:$C_k = \{C_1, C_2, \dots C_n\}$ entropy based divided test vectors

Output: $O_k = \{O_1, O_2, \dots O_n\}$ compressed test vectors &b_K^* is the expected weighted indices distance corresponding between the adjacent blocks of the test vectors.

Step 1: Set $\theta = (C, O, \tau)$ with random initial conditions by using prior information of primary test vector.

Step 2: Recursively determine the probability for the continued manifestation of bits. Let $\alpha_i(K) = P(O_1 = o_1, O_2 = o_2, \dots, O_T = o_T, C_K = i|\theta)$, the probability of appearing the $o_1, o_2, \dots, o_K$and being in state i at time K and j at time K+1 is represented in equations 5 and 6.

$$\alpha_i(1) = \tau_i b_i(o_1) \qquad (5)$$

$$\alpha_j(K+1) = b_i(o_{t+1}) \sum_{i=1}^{N} \alpha_i(K)\alpha_{ij} \qquad (6)$$

Step 3: Calculate the temporary variables from the test vectors, by evaluating the transition probability of bit being in state i at time K given the spotted sequence O and the parameters θ (using the Bayes' theorem) is given in equation 7.

$$T_i(K) = P(C_K = i|O, \theta) \qquad (7)$$

Step 4: Calculate the probability of being in state i and j at times K and K+1 respectively given the observed sequence O and parameters θ by the equation8.

$$\varphi_{ij}(K) = P(C_K = i, C_{K+1} = j|O, \theta) \qquad (8)$$

Step 5: Update θ in order to resolvethe expected number of transitions from state i to state j in comparison to the expected total number of shifts away from state i using equation 9.

$\tau_i^* = T_i(1)$*Expected frequency spent in state i at time 1*

$$t_{ij}^* = \frac{\sum_{K=1}^{K-1} \varphi_{ij}(t)}{\sum_{K=1}^{K-1} T_i(t)} \qquad (9)$$

Step 6: Compress the bits by replacing 1 or 0 with r and simultaneously determining the final weighted vectors for the number of shifts away from state i that do not mean transitions to a different state j, but to any state including itself using equation 10. This is corresponding to the no. of times state i is detected in the sequence from K=1 to K-1.

$$b_K^*(r) = \frac{\sum_{K=1}^{K} 1_{o_K=r} T_i(K)}{\sum_{t=1}^{T-1} T_i(K)} \qquad (10)$$

Where, b_K^* is the expected weighted indices distance corresponding between the adjacent blocks of the test vectors; such that the number of times the output observations have been equal to r while in state i over the expected total number of times in state i by equation 11.

$$1_{o_K=r} = \begin{cases} 1, if\ o_K = r \\ 0, otherwise \end{cases} \qquad (11)$$

Step 7: Repeat Steps 4-6 iteratively until $P(O|\theta_{final}) > P(O|\theta)$ is reached.

Step 8: Recalculate the entropy of the compressed test vector using equation 12.

$$E(O_k) = -\sum_{i=1}^{n} p_i * \log_2 p_i$$

$$//for\ O_k = \{O_1, O_2, \dots O_n\} \qquad (12)$$

Where $p_i = p_{i-1} + b_K^*$

Step 9: Stop

IV. EXPERIMENTAL RESULTS

A series of experiments were performed on ISCAS'89 and ITC'99 benchmark circuits. The experimental outcome of these benchmark testing are presented in Table 1 and 2 below. We've listed the data volume, compressed bits, the compression ratio and the power reduction. T_D alludes to the initial test data volume and T_E to the packed test data volume. These results are given without repetition of ATE pattern. The results demonstrate that compression percentage obtained with ISCAS is most of the time higher and consistent than that obtained with ITC. The comparison plot of test data reduction between the proposed method and that of several other techniques namely, Golomb, FDR, ALT-FDR, AVR, MAVR, EFDR for ISCAS and ITC circuits are represented in figure 1 below [25,26,30, 32, 33].

TABLE 1: Compression& Power Reduction Percentage of Experimented ISCAS'89Benchmark.

Circuit	Total Original Bits	Output Compressed Bits	Compression (in %)	Power Reduction (in%)
C7552	2934	573	80.45	64.5
C6288	2342	366	84.34	65.03
C5315	3032	505	83.32	65.4
C3540	2786	411	85.22	64.9
C2670	3789	587	84.49	65.4

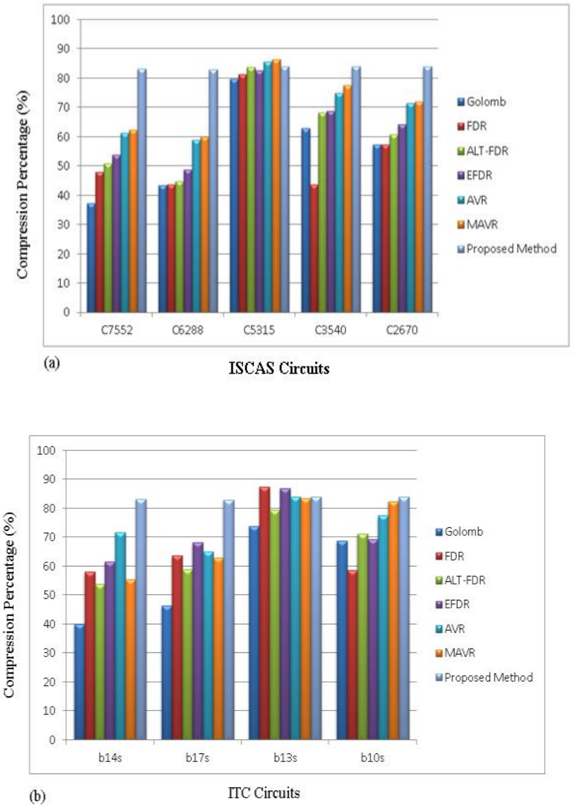

Figure 1: Comparison plot between the proposed method and other techniques for various run length based codes on (a) ISCAS & (b) ICT.

Test power consumption has additionally been evaluated with ISCAS and ITC. The reduction in power is estimated by utilizing the total number of weighted transition metric $b_K^*(r)$. We have evaluated the power reduction with initial arrangement generated with an irregular assignment of don't-care bits and afterward contrasted it with ISCAS and ITC. These results demonstrate the effectiveness of the proposed power reduction while compressing test data. With the presented technique we have achieved power reduction of up to 65% with ITC when differentiated with ISCAS; a little fluctuation is noted in the results. For a majority of benchmarking circuits, the heuristic performs better than the others as it guarantees less activity in the scan affix contrasted with ITC. On the other hand, the little reductions obtained for a few circuits can be clarified by the high percentage of power reduction obtained with ISCAS contrasted with an irregular fill solution. In other words, when ITC accomplishes a decent reduction in power consumption, there is not much opportunity for additional reductions with ISCAS.

TABLE 2: Compression & Power Reduction Percentage of Experimentation on ITC'99Benchmark

Circuit	Total Original Bits	Output Compressed Bits	Compression (in%)	Power Reduction (in%)
b14s	3324	559	83.16	65.00
b17s	2245	382	82.94	65.03
b13s	4782	772	83.85	65.04
b10s	5348	861	83.90	65.10

V. CONCLUSION

In this study, we have proposed a software based test data compression technique. The technique is completely lossless and efficient in terms of both time and space on account of its higher compression percentage (83%) and quick decompression process. Experimental results demonstrate that contrasted with an irregular filling solution, up to or above 83% reduction in test data volume and up to 65% of power reduction is achieved with the proposed technique. Thus, it decreases both the amount of test storage and testing time, thereby diminishing the tester memory and channel capacity requirements. As the proposed method is predominantly software based, the hardware requirements and cost of ATE are minimized. In addition, this method does not require detailed structural information about the IP circuit under test and utilizes a non-specific on-chip decoder that is independent of the IP center and the test set. Future work of the study will be concerned with automatic application of test vectors for investigating test fault scope.

REFERENCES

[1] Y. Zorian, E. J. Marinissen, and S. Dey, Testing Embedded-Core based System Chips, in Proc. International Test Conference pp.130-143 (1998).

[2] P. S. Zuchowski, C.B. Reynolds, R. J. Grupp, S. G. Davis, B. Cremen, and B. Troxel, A Hybrid ASIC and FPGA Architecture International Conference Computer Aided Design Pp. 187-194 (2002).

[3] S.C. Seth, and V.D. Agarwal, A Review of Testing of Digital VLSI devices. IETE Technical Review 2: 363-374 (1985}.

[4] A. Ahmed, and A. H. AI-Habsi, Design of a Built in Multi-Mode ICs Tester with Higher Testability Fatures – A Most Suitabe Testing Tool for BIST Environment. IETE Technical Review 15: 283-288 (1998).

[5] A. Jas, J. Dastidar and N. A. Touba, Scan Vector Compression/Decompression using Statistical Coding, in Proc. VLSI Test Symposium Pp. 114-120 (1999).

[6] A. Jas and N. A. Touba, Deterministic Test Vector Compression/Decompression for Systems on a Chip using an Embedded Processor. Journal of Electronic Testing: Theory and Applications 18: 503-514 (2002).

[7] A.Jas and N.A.Touba, Test Vector Decompression via Cyclical Scan Chains and its Application to Testing Core-based Designs, in Proc. International Test Conference Pp. 458-464 (998).

[8] M. Ishida, D. S. Ha and T. Yamaguchi, COMPACT: A Hybrid Method for Compressing Test Data, in Proc. VLSI Test Symposium Pp. 62-69 (1998).

[9] V. Iyengar, K. Chakraborty and B. T. Murray, Built-in Self Testing of Sequential Circuits using Precomputed Test Sets, in Proc. VLSI Test Symposium Pp. 418-423 (1998).

[10] A. Chandra and K. Chakrabarty, Test Data Compression for System-on-a-Chip using Golomb Codes, in Proc. VLSI Test Symposium Pp. 113-120 (2000).

[11] A. Chandra and K. Chakrabarty, Efficient Test Data Compression and Decompression for System-on-a-Chip using Internal Scan Chains and Golomb Coding, in Proc. Design Automation and Test in Europe Pp. 145-149 (2001).

[12] ISCAS'89 and ITC'99 benchmark Verilog model, http://ddd.fit.cvut.cz/prj/Benchmarks/

[13] A. Chandra and K. Chakrabarty, Frequency-Directed Run-Length Codes with Application to System-on-a-Chip Test Data Compression, in Proc. VLSI Test Symposium Pp. 42-47 (2001).

[14] Semiconductor Industry Association, International Technology Roadmap for Semiconductors (ITRS) (2005).

[15] A. Khoche, and J. Rivoir, I/O Bandwidth Bottle neck for Test: is it Real? Test Resource Partitioning Workshop (2000).

[16] M. L. Bushnell and V. D. Agrawal, Essentials of Electronic Testing, Kluwer Academic Publishers, (2000).

[17] M. R. Nelson, Data Compression with the Burrows Wheeler Transform, Dr. Dobb's Journal 22: 46-50 (1996).

[18] M. Burrows and D. J. Wheeler, A Block Sorting Lossless Data Compression Algorithm, Digital Equipment Corporation. Palo Alto, CA, Res. Rep. 124 (1994).

[19] I. Hamzaoglu, and J. H. Patel, Test Set Compaction Algorithms for Combinational Circuits, in Proc.International Conference on Computer-Aided Design Pp. 283-289 (1998).

[20] L. Chen and S. Dey, Software-based Self-Testing Methodology for Processor Cores. IEEE Transactions on Computer-Aided Design 20: 369–380, (2001).

[21] I. Hamzaoglu and J.H. Patel, Test Set Compaction Algorithms for Combinational Circuits. IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems 19(8): 957–963 (2000).

[22]. J. Rajski, J. Tyszer, M. Kassab and N. Mukherjee, Embedded Deterministic Test, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 23(5): 776–792 (2004).

[23] A. Jas, J.G. Dastidar, J.Mom-Eng and N.A. Touba, An Efficient Test Vector Compression Scheme using Selective Huffman Coding. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 22(6): 797– 806 (2003).

[24] S. Mitra and K. S. Kim, X-Compact: An Efficient Response Compaction Technique. IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems 23(3): 421–432 (2004).

[25] A. Chandra and K. Chakrabarthy, System on a Chip Data Compression and Architecture based

on Golomb Codes. IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems 20(3): 335-368 (2001).

[26] A. Chandra and K. Chakrabarty, Test Data Compression and Test Resource Partitioning for System on a Chip using Frequency Directed Run-Length (FDR) Codes. IEEE Transactions on Computers 52(8): 1076– 1088 (2003).

[27] P. T. Gonciari, B. Al-Hashimi and N. Nicolici, Variable Length Input Huffman Coding for System on a Chip Test. IEEE Transaction on Computer Aided Design of Integrated Circuits and Systems 22(6): 783–796 (2003).

[28] M. Nourani and M. Tehranipour, RL-Huffman Encoding for Test Compression and Power Reduction in Scan Application, ACM Transaction on Design Automation Electronic Systems 10(1): 91–115 (2005).

29] M. Tehranipoor, M. Nourani and K. Chakrabarty, Nine-Coded Compression Technique for Testing Embedded cores in SoCs. IEEE Transcriptions on Very Large Scale Integration(VLSI) Systems 13(6): 719–731 (2005).

[30] S. Usha Mehta, S. Kankar Dasupta, M. Niranjan and Devashrayee, Hamming Distance Based Reordering and Column Wise Bit Stuffing with Difference Vector: A Better Scheme for Test Data Compression with Run Length Based Codes, in Proc. 23rd International Conference on VLSI Design Pp. 33-38 (2010).

[31] A.H. El-Maleh, Efficient Test Compression Technique based on Block Merging. IET Computers and Digital Techniques 2(5): 327–335 (2008).

[32] J. Lee and N.A Touba, LFSR-Reseeding Scheme Achieving low-Power Dissipation during Test. IEEE Transaction on Computer-Aided Design of Integrated Circuits and System 26(2): 396-401 (2007).

[33] A. H. El-Maleh, Test Data Compression for System on a chip using Extended Frequency Directed Run-Length Code. IET Computers and Digital Techniques 2(3): 155-163 (2008).