International Journal of Engineering Research-Online
A Peer Reviewed International Journal
Email:editorijoer@gmail.com http://www.ijoer.in

Vol.4., Issue.2., 2016
(Mar-Apr)

RESEARCH ARTICLE

ISSN: 2321-7758

EFFECTIVE TEST DATA COMPRESSION USING REDUCED CONTROL CODE

K. S. NEELU KUMARI[1], RANJITHA . M[2], SANDHIYA . G[3], SARANYA AMUTHA . C[4]
[1]Assistant professor, [2,3,4,]Student,
Dept. of ECE,P.A. College of Engineering and Technology, Pollachi.

ABSTRACT

Test data compression is an efficient method to reduce the large test data volume for a system on chip designs. A new compression technique is proposed in this paper to achieve high compression for test data having large number of specified values (0's and 1's) called as reduced control code. In Multi Dimensional Pattern Run Length Coding (MDPRC) are made respectively from one dimensional PRC to three dimensional PRC. In this technique, test data are divided into a number of blocks, then each block is compared with its consecutive block as bit by bit, then it has to be merged. Depends upon the number of blocks are merged, it generates the control code. The compression can be achieved by using reduced control code technique. To demonstrate the effectiveness of the proposed method, experiments are conducted on ISCAS'89 benchmarks. The results shown in this method, can achieve significant compression in a test data volume.

Key words: Test data compression, MDPRC, Control code, Specified bits.

INTRODUCTION

In modern devices several modules are integrated on a single chip. Thus the increasing integration of transistors on a single chip produces robust design, more defects are produced accordingly. In this scenario there is a need to test those designs. As the technology advances huge volume of test data is needed to be tested. But the huge volume of test data will increase the testing time and area, this is one of the major issues for the SOC vendors. Because the huge test data volume not only exceeds the commercial ATE memory and I/O channel capacity, it also increases the testing time and test power. Increase in test time, memory and power cause a direct impact on test cost and time to market. So the vendors are in a situation to reduce the volume of test data. Hence the test data compression is mandatory for such cases.

Data compression is broadly divided into two categories such as lossy and lossless compression. In lossless compression the original data can be recovered without any data loss. It is usually used to compress text or binary files. The other is lossy compression, some loss of data quality is so acceptable. It usually used to compress image data files.

Many test data compression techniques have been proposed so far to reduce the test data volume and improve the transmission efficiency between the automatic test equipment and system on chip. The compression technique is used to compress the test data and is stored in ATE memory. Through the test channels the compressed data are

transferred to SOC. An on chip decoder is used to retrieve the original test data without any loss and is scanned regularly.

Previous works: As a result of many researches on test data compression many techniques were evolved. Besides all the test data compression techniques fall into any one of this category such as code-based schemes, linear decompression based schemes and broadcast-scan-based schemes. Code based scheme compression appears as dividing the original test data into patterns and replacing it by a code-word according to its specific property to encode the data. In decompression area the decoder simply replace the codeword by the specific patterns.

A block merging technique is used to merge many successive test blocks to reduce the test data volume. Both the test data volume and dictionary volume are reduced by having smaller number of codeword for larger block size. Based on encoding runs, a variable-to-variable length compression technique is called Multi-Dimensional Pattern run length codes (MDPRC), which divides the test vector into number of blocks. The compatible patterns can be given by a code word where number of pattern runs and pattern length is encoded.

A block merging technique is used to merge many consecutive test blocks to reduce the test data volume. To minimize the test time a high speed and throughput method is proposed in Pattern Run Length Coding (PRL).It gives the data independent methods which compress the consecutive patterns in an innovative manner with simple decompression architecture. A shortest average code word length can be given by Huffman coding with a disadvantage of larger decoder size. A cyclic shift compression with multiple scan chains will completely reduce the test time. On the other hand, it gives very less compression ratio with higher hardware overhead. By using MDPRC compression, by skipping the switching activity for longer scan chains it is possible to reduce the test power and test data volume.

Multi Dimensional Pattern compression is a run-length-based compression technique. In this method the encoder encodes the test set by runs of compatible pattern. The test pattern can be generated for compression using LFSR. The generated pattern must have small number of don't-care bits (X's). In this technique the test pattern can be divided into number of bit-patterns (Blocks). Each bit-pattern is compared with the adjacent one and the number of runs can be denoted by a control code. The final codeword has control code followed by encoded pattern. The Control code (C) denotes the number of pattern runs of the Encoded pattern (E) and the merge of both is denoted by codeword.

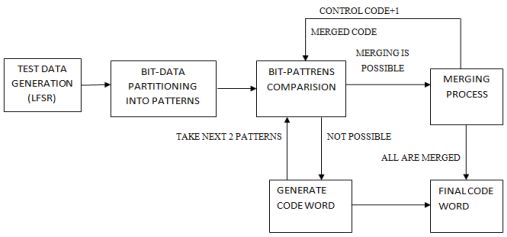


Figure .1 Existing Block diagram.

Figure 1, shows the block diagram of test pattern compression. The block diagram shows the flow of the compression process. The test pattern is partitioned into number of bit-patterns of specified length. First two bit-patterns has to be taken and have to compare bit-by-bit. Depending upon the possibility of merging, successive bit-patterns has to be merged with the previous outputs and final codeword has to be generated. If there is no possibility of merging two bit strings, a codeword has to be generated for the previous outputs and new process of merging has to be started from the next string.

The test vectors generated by ATE can be divided into number of patterns; each pattern can be compared and merged with adjacent one. Depending upon the number of runs a control code can be encoded. This method gives more compression when test vectors have less number of specified bits. If the specified bits are more and more we cannot able to merge those patterns. Similarly, if none of the pattern can be merged then it has to be represented by a separate control code for each pattern.

TEST PATTERN	PARTITIONED BLOCKS (B=4) CONTROL CODE=(00,01,10,11)				CODEWORD(C+E) (C=CONTROL CODE, E=ENCODED PATTERN)							
	B1	B2	B3	B4	C	E	C	E	C	E	C	E
1XXXX0XX1X11XX1X	1XXX	X0XX	1X11	XX1X							11	1011
X1XX1X001XXXX001	X1XX	1X00	1XXX	X001					01	1100	01	1001
XX011X00X0111100	XX01	1X00	XO11	1100	00	XX01	00	1X00	00	X011	00	1100

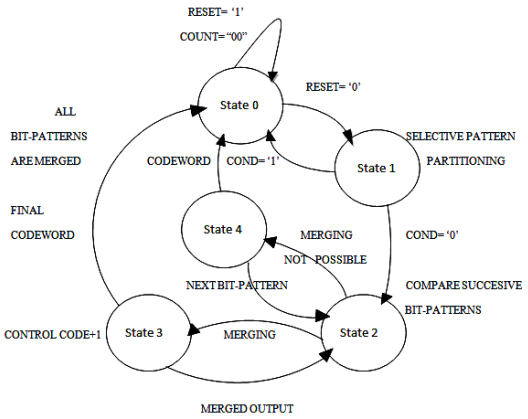
Table – 1 Actual compression in existing method of MDPRC

From the above table Test pattern1, Test pattern2, Test pattern3 is considered as test patterns and divided into four blocks as B1, B2, B3 and B4.Final code considered as combination of Codeword(C) with Encoded pattern (E). The first two blocks can be compared bit-by-bit. After comparing, the two blocks can be merged when there is no transition between the 2-bits during comparison. The merged output is then compared with next block and the process of merging is going on till any transition occurs. At last a Control code can be encoded which denotes the number of blocks merged. If a transition occurs the merging process should be stopped before the block where the transition occur and the corresponding codeword should be generated. Then a new process of merging is started from the next two blocks.

State Diagram

Figure 2, shows the state diagram of MDPRC compression. When RESET='1', the system will be in state 0, which initialize the system. When RESET='0', the test data is divided into number of bit patterns in state 1. Again the system goes to state 0, when condition (COND)='1'. If COND='0', it goes to state 2 where comparison of bit patterns are done to check the possibility of merging two bit patterns. If possible go to state 3 and if not goes to state 4.

State 3, the two bit patterns are merged into one with the increment of control code and the merged output going to state 2 for comparison with ext bit pattern for merging. The process continues till the last bit pattern was merged. When all the bit pattern was merged, the present Control code (C) with a encoded data (merged pattern) gives the code word. The final code word is the compressed data. At state 4, a code word for the present pattern continues its process from the state2 by taking next two bit patterns for merging. Finally the process resets when all the patterns were merged.



Proposed work

The proposed method is Reduced Control Code. The test vectors having larger number of unspecified values (X's) can be merged easily resulting in more compression. If the known values (0's and 1's) are more the possibility of merging is comparatively less. At certain instances there is no possibility for merging of sequences. At this situation for every sequence, a separate Control code has to be generated. As a result, the size of the compressed sequence is larger than the original input sequence. So the overall compression ratio is getting reduced in existing method.

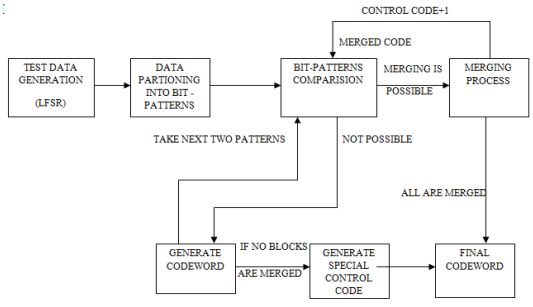


Figure 3 Proposed block diagram

In Table-2, the test pattern (TP3) has more specified bits and it cannot able to compress. The size of TP3 is 16-bits and its compressed codeword as the size of 24 bits. This is because none of the block is merged. So each block encodes a 2-bit control code. So the compressed sequence has 8-bits more than the original data. In this situation it is possible to reduce the 8-bit control code into 3-bit control code. A special control code (SC) "000" is encoded before the original sequence which indicates none of the blocks are merged. So the final codeword for the uncompressed pattern has a SC followed by original sequence (TP3 or E) is given in table 2.The size of the codeword using normal control code has 24 bits. But the size of the codeword using special control code has the uncompressed size of 19 bits. Totally 5 bits can be reduced in the below example.

TEST PATTERN	CODEWORD USING NORMAL CONTROL CODE CODEWORD=C+E								FINAL CODEWORD SPECIAL CONTROL CODE(SC)	
	C	E	C	E	C	E	C	E	SC	E
XX011X00X0111100	00	XX01	00	1X00	00	X011	00	1100	000	XX011X00X0111100

Table -2 Proposed compression technique

Experiment Results

This proposed compression technique has been experimented in ISCAS'89 benchmark circuits. Here 2-bit control code and 4-blocks are taken, which means that the test patterns can be divided into 4-blocks and the process of comparison and merging can be done. It is possible to modify or reduce the size of the control code when none of the blocks are compressed. By an average of 3% to 9% of compression can be obtained for the various benchmark circuits when no blocks are compressed.

The compression ratio (CR) for each and every method can be calculated by the equation as,

$$\text{Compression Ratio (CR) \%} = \frac{\text{((Total number of test data)–(Compressed data))}}{\text{(Total number of test data)}}$$

Bench mark Circuits	Compression Ratio for Existing system	Compression Ratio for Proposed system
S953	10%	23.75%
S9234	26.6%	32.77%
S13207	32.5%	35.93%

Table-3 Compression ratio for the proposed method

n Table-3 the compression ratio of the proposed method is compared with existing technique.

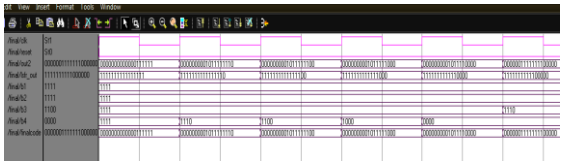


Figure .4, The Output signal of Final Test data Compression

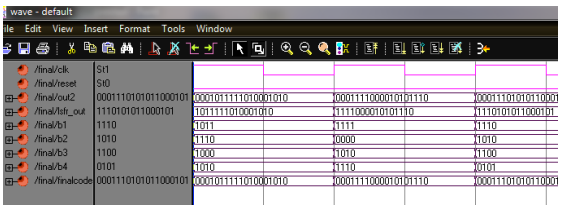


Figure .5, The Output signal for none of the blocks matched

Figure 4 and 5 shows the output signal for test data compression using reduced control code.

Conclusion

Most of the compression techniques give more compression ratio when the unspecified bits are higher. It is not possible to achieve better compression when known bits are higher, because the codeword size is comparatively larger than the size of the original sequence due to larger number of control codes. This method is intensively proposed to reduce the size of the control code to achieve better compression by using proposed special control code method. This proposed method is experimented on ISCAS'89 benchmark circuits and its results are verified. The results clearly proved that the proposed method reduces the size of the

control code by the factor of 3% to 9% and a better compression is achieved.

References

[1]. Touba NA, "Survey of Test Vector Compression Techniques", IEEE Design and Test of Computers, 2006, 23(4):294– 303.

[2]. Chandra A, Chakrabarty K, "System-on-a-chip Data Compression and Decompression Architecture Based on Golomb Codes", IEEE Transactions Computer-Aided Design of Integrated Circuits & System., 2001, 20(3):355–368.

[3]. Chandra A, Chakrabarty K, "Test Data Compression and Test Resource Partitioning for System-on-a-chip using Frequency-directed run-length (FDR) codes", IEEE Transactions on Computer., 2003, Vol-52 (8), pp:1076–1088.

[4]. Gonciari PT, Al-Hashimi B and Nicolici N, "Improving Compression Ratio, Area Overhead, and Test Application Time for System-on-a-chip Test Data Compression/Decompression", In Proc design automation test in Europe, Paris., 2002, pp:604–611.

[5]. Nourani M and Tehranipour M, "RLHuffman Encoding for Test Compression and Power Reduction in Scan Application", ACM Transactions on Design Automation of Electronic Systems (TODAES)., 2005, Vol-10(1), pp:91–115.

[6]. El-Maleh AH, "An Effcient Test Vector Compression Technique Based on Block Merging", IET Computer Digit Tech., 2008, 2(5):327–335.

[7]. Terumine Hayashi, Naohiro Hiraiwa, Tsuyoshi Shinogi, Haruhiko Takase, and Hidehiko Kita, "Test Modification and Compression Technique for Reducing Total Test Volume with Dictionary Data", ASICON 2005, 6th International Conference, IEEE Transaction.

[8]. Julien Dalmosso, Marie-Lise Flotters and Bruno Rouzeyre, "Systems-on-Chip: Use of Test Data Compression Technique for Reducing Test Time"., Research in Microelectronics and Electronics Conference, PRIME 2007, IEEE Transaction.

[9]. Keun-Soo Lee, Hyuntae Park, Hyeonuk Son and Sungho Kang, "A New Scan Slice Encoding Scheme with Flexible Code for Test Data Compression", SoC Design Conference (ISOCC), 2010 International Conference, IEEE Transaction.

[10]. Usha S. Mehta, Kankar S. Dasgupta and Niranjan M. Devashrayee, "Hamming Distance Based Reordering and Columnwise Bit Stuffing with Difference Vector: A Better Scheme for Test Data Compression with Run Length Based Codes", VLSI Design, 2010. VLSID'10, 23rd International Conference.

[11]. S.Saravanan and Har Narayan Upadhyay, "Transition Vector Reduction using Segmentation method based on Compression Technique", Australian Journal of Basic and Applied Sciences, 5(9): 2147-2151, 2011.

[12]. S.Saravanan, P.Selvakumar, A. Balasubramaniyan and R. Silambamuthan, "Achieving Higher Test Data Compression using Pattern ", IEEE International Conference on Computational Intelligence and Computing Research, ICCIC-2011.

[13]. Chia-Yi Lin and Hung-Ming Chen, "A Selective Pattern –Compression Scheme for Power and Test-data Reduction", Computer-Aided Design (ICCAD), IEEE/ACM International Conference, 2007.

[14]. Chia-Yi Lin, Hsiu-Chuan Lin and Hung- Min Chen, "On Reducing Test Power and Test Volume by Selective Pattern Compression Schemes", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 18(8) : 1220- 1224, 2010.