ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 2, March 2016

TEST DATA COMPRESSION USING MULTISTAGE ENCODING

P.Ashika[1],P.Ramya[1],P.Ramya[1],S.Sivasankari[1],M.Vishnupriya[2]

B.E Electronics and Communication Engineering

Assistant professor[2] Bharathiyar Institute of Engineering for Women

(pashika2@gmail.com)[1] (ramyakpece1995@gmail.com)[1] (er.ramya95@gmail.com)[1]
(sankarisiva720@gmail.com)[1] (visnupriyam@yahoo.com)[2]

Abstract- Developments in process technology have lead to the propose of systems millions of transistors on a single chip and it has resulted in an increase of test data essential to test the circuits. conventional peripheral testing processes involve store all test vectors and test responses on the automatic test equipment (ATE) memory. The test data amount for the scan-based test is generally very large due to its single pattern length generated using a combinational persistent test prototype generation (ATPG) tool. The test submission time depends on the sum of test data stored on ATE, the time essential to recommend the test data from ATE to the core and duration of the scan chain. But these testers have restricted memory, speed and I/O channels. resourceful test data diminution techniques can reduce the testing time, test power and ATE memory supplies. Three multistage compression techniques are introduced to decrease the test data ability in scan-test applications. The three encoding schemes mainly equal run-length coding (ERLC), extended frequency directed run length (EFDR) coding, alternating variable run length (AVR) is used for computing the data. These encoding scheme together with nine coded (9C) technique improve the test compression ratio. In the first stage, pre-generated test cubes with undetermined bits are encoded using nine-coded (9C) method. Later the three encoding schemes utilize the properties of compacted data to enhance the test compression. This multistage compression is effective particularly when the portion of don't worry in a test set is very high. The experimental result obtained from ISCAS'85 benchmark circuit confirms the average compression ratio of 40%, 42%, 47% with the projected 9C-ERLC, C-EFDR, 9C-AVR codes respectively.

RELATED WORK

The test pattern contains zeros and ones. By compressing the runs of zeros Chandra (2001) proposed Golomb codes [1]. Another method of considering run length of zeros proposed frequency intended for run-length (FDR) code [2]. It is a variable-to-variable run-length code which converts variable length symbols into variable length codeword. It is not suitable when there are more number of 1's in test cube. The compression methods considering both the run length of zeros and ones such as Huffman coding [3], alternating FDR (ALT-FDR) coding, unlimited FDR (EFDR) coding [4], alternating variable-run-length (AVR) coding [5], equal run-length (ERLC) coding [6]. The minimum transition count (MTC) filling approach is used for simultaneous reduction of test data volume and power dissipation. There are several multi-stage compression are available to improve the compression ratio [7]. The run length encoding and Huffman encoding are mixed in order to reduce the test volume, test power, test application time. Multi-level Huffman coding is obtainable to improve the compression ratio [8].

I. INTRODUCTION

The technology today makes it possible for designers to increase the difficulty of their designs such that an entire electronic scheme can be implemented on a single chip. Testing is essential to identify chips with failure and to diagnose the reason for a failure in order to improve the manufacturing process. The conventional external testing contains

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 2, March 2016

storing of all the test patterns and test responses of the automatic test equipment (ATE). The amount ofdata stored represents the test data volume which is very large due to its single pattern length generated by automatic test pattern generator (ATPG) tool. The test application time depends on the amount of test data stored in ATE.

These testers have limited memory, speed, I/O channels. The usage of more power in test mode than in normal mode leads to decrease in reliability. Test compression is one of the method to decrease the size of the ATE memory. It reduces the number of bits in test vector by using compression coding schemes and the reduced patterns are stored in automatic test equipment (ATE). If unspecified bits are present in the test cubes then it leads to large amount of compression. There are several types of code-based compression techniques are available and are explained as follows. Run-length code, golomb code are used to partition the original data into symbols. These symbols are assigned with a codeword to form encoded data. These methods don't require any structural information about the circuit under test (CUT). We present three multi-stage compression techniques called 9C-EFDR, 9C-ERLC and 9C-AVR respectively. The data obtained after the 9C compression is further compressed to improve the compression ratio. First the test set with undetermined bits is compressed with the nine coded compression (9C) technique [9] and the resulted test set is further encoded with extended FDR (EFDR) coding, equal run-length (ERLC) coding and alternating variable run-length (AVR) coding. The obtained results are compared with other compression coding techniques. From that it is analyzed that 9C-AVR provides better compression ratio than the other compression coding techniques.

II. COMBINED CODING

The test patterns obtained from automatic test pattern generator (ATPG) contains more number of unspecified bits. By using the nine-coded compression (9C) coding technique, better compression is achieved. It provides another important application that it reduces the test power. The encoded test set obtained from 9C coding consists of extended runs of 0's and 1's, as well as a large number of repeated codeword can also be noted. Because of the reduction in number of bits to be tested, it is used as a first stage of compression. The second compression uses the encoded data obtained from 9C as its test data. The first technique is extended FDR (EFDR) which considers both the runs of 0's and runs of 1's. The second technique is equal run length (ERLC) coding which considers both the runs of 0's and runs of 1's. If there are any repeat runs then it replaces it with the codeword 000 or 100 respectively. The third technique is alternative variable run-length (AVR) coding which considers both the runs of 0's and runs of 1's respectively.

The nine coded compression technique (9C) consists of nine codeword in which the input containing a fixed block size. This block size is user-defined. The input data is partitioned into blocks, say k, called symbols. These symbols are replaced with the corresponding codeword.The size of the block should be selected as even. So that each block can be partitioned equally and these partitioned blocks may be all 0's, 1's or a set of unequal bits, that is, a varied group of 0's, 1's and X bits. Table 1 shows the coded scheme containing block size k=8. Each half of the k-bits contains either all 0's or all 1's for cases 1-4, but for cases 5-8, there is one half with either all 0's or all 1's and other half with all mismatched bits, indicated as uuuu. The case 9 has entire k-bits to be mismatched. The test data volume is further reduced

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 2, March 2016

using EFDR,ERLC or AVR technique.

Table 1 9C coding scheme having block size k=8

Case	Input block	Symbol	Description	Code word
1	0000 0000	00	All 0's	0
2	1111 1111	11	All 1's	10
3	0000 1111	01	Left half 0's, right half 1's	11000
4	1111 0000	10	Left half 1's, right half 1's	11001
5	1111 uuuu	1U	Left half 1's, right half mismatched bits	11010
6	uuuu 1111	U1	Left half mismatched bits, right half 1's	11011
7	0000 uuuu	0U	Left half 0's, right half mismatched bits	11100
8	uuuu 0000	U0	Left half mismatched bits, right half 0's	11101
9	uuuu uuuu	UU	All mismatched bits	1111

B. Extended frequency directed run-length (EFDR) code.

It is a variable-to-variable coding (ie., both the symbol and codeword have variable length). It has group prefix and tail. It is similar to FDR code but 0 is prefixed for codeword runs of 0's and 1 is prefixed for codeword runs of 1's. Table 2 shows the coding scheme of extended FDR technique. It considers the runs of 0's as strings of 0's followed by bit 1 and runs of 1's as strings of 1 followed by bit 0.

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 2, March 2016

Table 2 Extended FDR coding technique

Group	Run-length	Group prefix	Tail	Code word Runs of 0's	Code word Runs of 1's
A1	1	0	0	000	100
	2		1	001	101
A2	3	10	00	01000	11000
	4		01	01001	11001
	5		10	01010	11010
	6		11	01011	11011
A3	7	110	000	0110000	1110000
	8		001	0110001	1110001
	9		010	0110010	1110010
	10		011	0110011	1110011
	11		100	0110100	1110100
	12		101	0110101	1110101
	13		110	0110110	1110110
	14		111	0110111	1110111
....	....	....	... .	....	....

C. Equal run-length (ERLC) code

It is a variable- to-variable coding (ie.,both the symbol and codeword have variable length) which considers both runs of 0's and 1's. It is very similar to the EFDR code except that the ERLC code for run length i is the EFDR code for run-length i+1.For repeated runs a 3-bit codeword (000 or 100) is used, that is, if the length of a consecutive run is the same as that of the former, then the whole second run is replaced with the shorter codeword. Table 3 shows the coding scheme of the equal run-length coding technique.

Table 3 Equal run-length coding technique

Group	Run-length	Prefix	Tail	Code word Runs of 0's	Code word Runs of 1's
A1	1	0	1	001	101
A2	2	10	00	01000	11000
	3		01	01001	11001
	4		10	01010	11010
	5		11	01011	11011
A3	6	110	000	0110000	1110000
	7		001	0110001	1110001
	8		010	0110010	1110010
	9		011	0110011	1110011
	10		100	0110100	1110100
	11		101	0110101	1110101
	12		110	0110110	1110110
	13		111	0110111	1110111
....	....	....	... .	....	....

D. *Alternative variable run-length (AVR) code*

Table 4 Alternative variable run-length (AVR) coding

Group	Run-length	Group prefix	Tail	Code word
A1	1	01	0	010
	2		1	011
	3	10	0	100
	4		1	101
A2	5	001	00	00100
	6		01	00101
	7		10	00110
	8		11	00111
	9	110	00	11000
	10		01	11001
	11		10	11010
	12		11	11011
....	....	....	....	....

It is a variable-to-variable coding (i.e., both the symbol and codeword have variable length). It

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 2, March 2016

encodes both the runs of 0's and 1's. Here, the run of 0s is considered as strings of 0's followed by bit 1 and the run of 1s is considered as strings of 1's followed by bit 0. It contains same codeword for both the runs of 0's and 1's having same run-length. Table 4 shows the alternative variable run-length coding (AVR) technique.

III. EXPERIMENTAL RESULTS

The experiments were conducted on five ISCAS '89 benchmark circuits. C language was used for the implementation of the work done. The obtained results were compared with the already presented codes.

A. Compression result with 9C-EFDR, 9C-ERLC and 9C-AVR techniques

The compression obtained from the 9C-EFDR, 9CERLC and 9C-AVR codes for ISCAS '89 benchmark circuits were compared with the other run-length codes which is shown in table 5. The other run-length codes, such as, golomb, FDR, EFDR, ERLC, AVR, AFDER, Huffman and 9C-AFDER were used for relationship. It represents the total amount of data that has been reduced in terms of number of bits.

Table 5 Compressed bits for various compression codes

Circuit	S27	S298	S344	S400	S526
Original	112	1819	2424	2832	3504
Golomb	74	999	1103	1306	1727
FDR	80	1054	1162	1296	1860
Huffman	67	1053	1047	1255	1643
AFDER	76	1118	1305	1477	1859
AVR	81	1094	1055	1318	1682
EFDR	65	1330	1365	1598	2143
ERLC	91	1278	1234	1522	1950
9C-AFDER	72	1017	1070	1255	1781
9C-ERLC	74	1021	1127	1330	1913
9C-EFDR	66	899	1007	1178	1671
9C-AVR	62	805	909	1014	1513

The compression ratio is calculated by considering the size of the bits. It is obtained as, Compression gain (%) = (original bits –compressed bits) / original bits x 100 Where, the original bits are the total amount of bits obtained from the test pattern. The compressed bits are the total amount of bits obtained after various compression techniques. Table 6 represents the compression ratio in terms of percentage for five benchmark circuits. From table 6, it is found that the 9CEFDR and 9C-AVR provides better compression ratio.

Table 6 Compression ratio in terms of percentage

Circuit	S27	S298	S344	S400	S526
Golomb	33.9	45	54.4	53.8	50.7
Huffman	40.1	42.1	56.8	55.6	51.1
FDR	28.5	42	52	54.2	46.9
EFDR	39	26.8	43.6	43.5	38.8
AVR	27.6	39.8	56.4	53.4	50.1
ERLC	18.7	29.7	49	46.2	44.3
AFDER	32.1	38.5	46.1	47.8	46.9
9C-AFDER	35.7	44	55.8	55.6	49.1
9C-ERLC	33.9	43.8	53.5	53	45.4
9C-EFDR	41	50.5	58.4	58.4	52.3
9C-AVR	44.6	55.7	62.5	64.1	56.8

IV. CONCLUSION AND FUTURE WORK

The compressed data reduces the size of the external memory is reduced. The number of bits to transfer from the external memory to device under test is reduced. Because of this, the bandwidth is also being reduced. Further the decompression has to be done. This is achieved with the help of the decoder architecture in which the structure depends on the coding technique. The decompression process is carried out in order to check the testability of the circuit.

REFERENCES

[1] Ye, Bo, Qian Zhao, Duo Zhou, Xiaohua Wang, Min Luo, (2011), 'Test data compression using alternating variable run-length code', Integration, the VLSI Journal 44, No. 2, pp. 103-110.

[2] Zhan,Wenfa, Aiman El-Maleh, (2012). 'A new scheme of test data compression based on equal-run-length coding (ERLC)', INTEGRATION, the VLSI journal45, No. 1, pp. 91-98.

[3] Sivanantham, S., M. Padmavathy, Ganga Gopakumar, P. S. Mallick, J. Perinbam ,(2013), 'Enhancement of test data compression with multistage encoding', Integration, the VLSI Journal.

[4] Mehta, Usha Sandeep, Kankar S. Dasgupta, Nirnjan M. Devashrayee, (2010), 'Modified selective Huffman coding for optimization of test data compression, test application time and area overhead', Journal of Electronic Testing26, No. 6, pp. 679-688.