Hindawi Publishing Corporation
VLSI Design
Volume 2011, Article ID 756561, 8 pages
doi:10.1155/2011/756561

Research Article

Weighted Transition Based Reordering, Columnwise Bit Filling, and Difference Vector: A Power-Aware Test Data Compression Method

Usha Mehta,[1] K. S. Dasgupta,[2] and N. M. Devashrayee[1]

[1] *PG (VLSI Design), Nirma University, Ahmedabad 382481, India*
[2] *Indian Institute of Space Science & Technology, Tiruvanthapuram, India*

Correspondence should be addressed to Usha Mehta, usha.mehta@nirmauni.ac.in

Received 29 March 2011; Revised 14 May 2011; Accepted 29 July 2011

Academic Editor: Yangdong Deng

Test data compression is the major issues for the external testing of IP core-based SoC. From a large pool of diverse available techniques for compression, run length-based schemes are most appropriate for IP cores. To improve the compression and to reduce the test power, the test data processing schemes like "don't care bit filling" and "reordering" which do not require any modification in internal structure and do not demand use of any test development tool can be used for SoC-containing IP cores with hidden structure. The proposed "Weighted Transition Based Reordering-Columnwise Bit Filling-Difference Vector (WTR-CBF-DV)" is a modification to earlier proposed "Hamming Distance based Reordering—Columnwise Bit Filling and Difference vector." This new method aims not only at very high compression but also aims at shift in test power reduction without any significant on-chip area overhead. The experiment results on ISCAS89 benchmark circuits show that the test data compression ratio has significantly improved for each case. It is also noteworthy that, in most of the case, this scheme does not involve any extra silicon area overhead compared to the base code with which it used. For few cases, it requires an extra XOR gate and feedback path only. As application of this scheme increases run length of zeroes in test set, as a result, the number of transitions during scan shifting is reduced. This may lower scan power. The proposed scheme can be easily integrated into the existing industrial flow.

1. Introduction

The testing cost and testing power are the two well-known issues of current generation IC testing [1].

The test cost is directly related to test data volume and hence test data transfer time [2]. Test data compression can solve the problem of test cost by reducing the test data transfer time. The dynamic test power plays a major role in overall test power. The switching activity during test has a large contribution in dynamic power and hence in overall test power. The extensive use of IP cores in SoC has further exaggerated the testing problem. Because of the hidden structure of IP cores, the SoCs containing large IP cores can use only those test data compression techniques and switching reduction technique which do not require any modification or insertion in architecture of IP core. These methods should not also demand the use of ATPG, scan insertion, or any such testing tools. They should be capable to use ready-to-use test data coming with IP core for data compression and power reduction. This test data may be partially specified or fully specified. Thus, the current research on IC testing can not be directly applied to the SoC because of the hidden structure of IP core.

So it can be inferred that the test data compression and switching reduction in context of hidden structure of IP core is the current need for SoC testing.

In literature, there are many test data compression techniques like linear decompression-based, broadcast scan-based, and code-based techniques. Considering to suitability to IP core-based SoC, code-based test data compression scheme is more appropriate. From the various code-based test data compression schemes like dictionary codes, constructive codes, statistical codes, and run length-based codes, the run length-based codes can be more suitable to IP cores

because of its simple on-chip decoder and better compression capacity. The do not care bit filling methods and test vector reordering further enhance the test data compression.

The switching activity reduction technique described in literature can be broadly classified in three categories: (1) techniques for built-in self-test, (2) techniques applied as design-for-test, and (3) techniques for external testing. Considering the suitability to IP cores, the techniques for external testing can be further explored. Out of various switching reduction techniques for external testing like low-power ATPG, input control, reordering, and do not care bit filling, the do not care bit filling and reordering are applicable for hidden structure of IP core.

To improve the compression ratio and to reduce the switching in the most famous run length-based data compression method, in this paper, a new scheme based on three techniques: Hamming distance and weighted transition-based reordering (WTR), columnwise bit filling (CBF), and difference vector (DV) is proposed. This scheme is applied to various test set prior to apply a variety of run based codes, and it gives better result in each of the case. The experiment results show that the test data compression ratio is significantly improved. Moreover, this scheme does not require any on-chip silicon area overhead compared to base run length code with which it is used. With the help of weighted transition-based reordering, the total number of transition during scan-in is also reduced. This method may reduce the overall scan power requirement during testing. Further, the proposed scheme can be easily integrated into the existing industrial flow.

The paper is organized as follows: the background for bit filling methods and test vector reordering used for test data compression and test power reduction is covered in Section 2. In Section 3, the Hamming distance-based reordering is explained with a motivational example. Section 3 introduces the concept of weighted transition-based reordering. Section 3.3 include the details of run length code used for compression. Experimental results and performance comparison are presented in Section 6 followed by concluding remarks in Section 8.

2. Background

2.1. Test Data Compression. In 1998, a scheme based on run-length codes that encoded runs of 0 s using fixed-length code words [3] was proposed. The Golomb code [4] encodes runs of 0 s with variable-length code words. The optimization of Golomb is achieved using frequency-directed run-length (FDR) codes [5]. Maleh and Abaji proposed an extension of FDR (EFDR) [6]. The alternating FDR uses runs of 0 s as well as 1 s in alternating fashion [7]. An evolution in alternate run length-based FDR is shifted alternate run length-based FDR [8]. The detailed description on each run length code with one example is given in [9] which also includes the compression, power, and area overhead in case of each run length-based compression code.

2.1.1. Do Not Care Bit Filling. Instead of simply filling all do not care bits with 0 s, if the do not care bits are filled considering the type of run used in particular compression scheme, the better compression can be achieved [10].

2.1.2. Reordering. Stuck at fault-based test patterns can be reordered without any loss of fault coverage. In literature, a number of test vector reordering techniques are proposed for test data compression. The run-based reordering approach [11] is based on reordering the test frames to give the bigger run lengths of 0 s. As this bigger run lengths are than coded with extended FDR, it gives better compression to normal extended FDR. As this approach uses scan frame reordering, it is not suitable to IP cores with hidden structure. The same thing is applicable to [12] which requires a large amount of area overhead to compensate the 2-D reordering. The Hamming distance-based reordering used in [13] is used as the basic scheme in this paper.

2.2. Test Power. For the reduction of switching activity in terms of number of transitions during scan operations, the do not care bit filling and reordering techniques are widely used.

2.3. Ordering Techniques. The greedy algorithm based reordering process using the minimum Hamming distance between them to reduce the scan power [14]. The concept of finding Hamiltonian cycle in a complete weighted graph is used in [15]. In [16], both scan latch reordering with test vector reordering is considered. Another work [17] has also considered the Hamming distance minimization between adjacent vectors to reduce the dynamic power dissipation during testing. Test vector reordering problem as TSP and genetic algorithm (GA) has been used to generate low-power test patterns in [18]. In [19], an evaluation of different heuristic approaches has been done in terms of execution time and quality. In [20], 2-opt heuristic and a GA-based approach with reduction in fault coverage is introduced. Roy et al. has proposed a test vector reordering technique switching activity reduction in case of combinational circuit with AI [21]. An ant colony optimization-based test vector reordering problem for power reduction is described in [22]. The particle swarm approach is used in [23]. There are few other approaches available in literature for test vector reordering, but, while considering the hidden structure of IP cores, these approaches are not found suitable. For capture power reduction in case of IP core-based SoC, the artificial intelligence-based reordering of scan vector is proposed in [24].

2.4. Do Not Care Bit Filling. An automatic test pattern generation (ATPG) scheme for low-power launch-off-capture (LOC) transition test based on do not care bit filling is proposed in [25]. A genetic algorithm based heuristic to fill the do not cares is proposed in [26]. This approach produces an average percentage improvement in dynamic power and leakage power over 0-fill, 1-fill, and minimum transition fill (MT-fill) algorithms for do not care filling. The work in [27] proposed segment-based X-filling to reduce test power and keep the defect coverage. The scan chain configuration tries to cluster the scan flip-flops with common successors into one scan chain, in order to distribute the specified bits

per pattern over a minimum number of chains. Based on the operation of a state machine, [28] elucidates a comprehensive frame for probability-based primary-input dominated X-filling methods to minimize the total weighted switching activity (WSA) during the scan capture operation. The work in [29] describes the effect of do not care filling of the patterns generated via automated test pattern generators, to make the patterns consume lesser power. It presents a tradeoff in the dynamic and static power consumption.

3. Weighted Transition-Based Reordering, Columnwise Bit Filling, and Difference Vector

The earlier proposed Hamming distance-based reordering, column-wise bit filling, and difference vector (HDR-CBF-DV) are taken as the basic scheme for the proposed method. This section includes the introduction to (HDR-CBF-DV) and the proposed modifications.

Before continuing the further explanation, the following two terms need to be defined.

Hamming Distance. The Hamming distance between two scan vectors is equal to the number of corresponding incompatible bits. This definition is similar to Hamming distance with extension of do not-care bits. For example, given two vectors V_1 = (10XX01) and V_2 = (001X11), the distance d(V_1,V_2) is 2 because the first and the fifth corresponding bits in the vectors are incompatible.

Weighted Transition. For a given test data set containing m vectors with n bits each, the weighted transition for each test vector is given by (1), and the total weighted transition during test is given by (2),

$$\text{Weighted Transitions for Scan-In vector } j = \sum_{i=1}^{n} (t(j,i) \oplus t(j,i+1)) * (n-i), \tag{1}$$

$$\text{Total Weighted Transitions during test} = \sum_{j=1}^{m} \sum_{i=1}^{n} (t(j,i) \oplus t(j,i+1)) * (n-i). \tag{2}$$

3.1. Weighted Transition-Based Reordering. If we take each test pattern as a vertex in a complete undirected graph G, and the distance between two patterns as the weight of an edge, then this problem is similar to Hamilton problem, which is NP-hard and solved by various greedy algorithms. The simplest pure greedy algorithm is choosing as the next pattern in a path the one that is closest to the current pattern, provided that it has not been visited yet. It seems that the Hamilton path of G is the solution to our reordering problem.

3.1.1. Selection of First Vector for Reordered Test Set

Hamming Distance-Based Selection. In [13], for the selection of first test pattern, the heuristic that is applied in this scheme is "Hardest Path First." The test pattern with minimum do not cares will be selected as the first test pattern of reorder list. The reason for selecting the test pattern with minimum do not care bits is that there is a minimum flexibility to stuff the bits later. If more than one test pattern have minimum do not care values than any one vector with the minimum do not care bits each will be selected.

Weighted Transition-Based Selection. In WTR-CBF-DV, if there are more than one test vectors with minimum number of do not care bits, each will be evaluated for its weighted transitions, and the vector with minimum weighted transition will be selected as the first test vector of the reordered test set. Further in earlier method, the first vector is kept unfilled until all the test vectors are reordered, but, in the proposed scheme, the selected first vector is MT filled before continuing reordering to make the overall testing and selection of remaining test vectors power aware.

3.1.2. Reordering and Bit Filling of Remaining Test Vector

Hamming Distance-Based Selection. For reordering of the remaining test patterns in [13], the pattern with minimum Hamming distance from first pattern of reordered set will be placed next to first pattern of reordered set. It is decided to take the next vector with minimum Hamming distance because when the further columnwise bit filling and difference vector will be done, this reordered vector sequence will generate maximum zeroes so run length, and hence the compression will increase. In HDR-CBF-DV [13], after completing the reordering of all the vectors, for the second test patterns and onwards, the do not care bit will be replaced with the same value which its upper vector has at the same position. The goal here is to get the maximum zeroes in difference vector.

Weighted Transition-Based Selection. During the reordering process for various circuits, it is found that generally the test set contains more than one test vectors with same Hamming distance. This happens because of the structural behavior of faults. This tie should be broken in favor of power reduction. So in the proposed scheme, while selecting the next vector of the reordered test set, if there are more than one vector with the same Hamming distance from the last selected vector of reordered set, then the weighted transition will be taken into consideration. All these equidistance vectors will be applied columnwise bit filling (explained in Section 5.2) and their weighted transitions are calculated. The vector with the minimum weighted transitions will be selected as next vector of reordered test set. The do not care bits in this vector will be replaced with the same value which its upper vector has at the same position.

3.2. Difference Vector. The next step is to take the difference vector of two consecutive vectors in reordered set. This will further increase the numbers of zeroes and hence data compression. Any run length code can be used to compress the difference vector sequence T_{diff}. Let $T_{\text{Dpt}} = \{t_1; t_2; \ldots; t_n\}$ be the reordered test set. T_{diff} is defined as follows: $T_{\text{diff}} = \{d_1; d_2; \ldots; d_n\} = \{t_1; t_1 \oplus t_2; t_2 \oplus t_3; \ldots; t_{n-1} \oplus t_n\}$, where a bit-wise exclusive or operation is carried out between

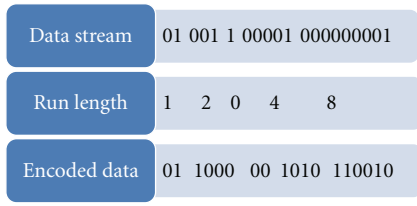

Figure 1: Example of frequency-directed run length coding.

Table 1: Frequency directed run length code.

Run length	k	Group prefix	Tail	Code word
0	1	0	0	00
1			1	01
2	2	10	00	1000
3			01	1001
4			10	1010
5			11	1011
6	3	110	000	110000
7			001	110001
-		---	--	------

patterns t_i and t_{i+1}. The successive test patterns in a test sequence often differ in only a small number of bits. Therefore, T_{diff} contains few 1 s, and it can be efficiently compressed using the FDR code.

3.3. Run Length Code for Compression. For the proposed method, the test data will be first preprocessed by WTR-CBF-DV scheme, and then the frequency-directed run length code (FDR) [5, 30] will be applied to preprocess data. The equation of % compression used very frequently in this paper is as follows:

$$\text{Test data compression in percentage} = \frac{\#\text{ of original bits} - \#\text{ of compressed bits}}{\#\text{ of original bits}} \times 100\%. \quad (3)$$

The examples in Figure 1 and Table 1 demonstrate this coding style.

4. Algorithm for WTR-CBF-DV

(1) Consider a digital circuit with n scan flip-flops, p inputs, and q outputs. The ATPG generated partially specified test set with m scan-in test vectors, and each of n bits is the input to this algorithm.

(2) Find test vector with minimum number of do not care bits in the given test set.

(3) If there are more than one vector with minimum do not care bits, then

(i) apply MT fill to each vector,

(ii) calculate weighted transition for each vector,

(iii) select the MT-filled vector with minimum WT as first vector of reordered set.

(4) Find the Hamming distance of remaining each vectors from the first vector of reordered set.

(5) Select the vector with minimum Hamming distance as next vector.

(6) If there are more than one vector with minimum Hamming distance, then

(i) apply columnwise bit fill to each vector, that is, replace the do not care bit of the vector with the same position bit value of last selected vector,

(ii) calculate weighted transition for each vector, and

(iii) select the columnwise bit filled vector with minimum WT as next vector of reordered set.

(7) Repeat step (6) until all the vectors are reordered.

(8) Apply difference vector mechanism.

(i) First vector of reordered set is kept unchanged.

(ii) From the second vector onward, if the same position bits in last vector and current vector are same, replace the bit with 0 else 1.

(9) Apply frequency-directed run length code.

5. Motivation Example

Considering the following test data, for example,

test vector

1	X	1	0	0	X	X	0	1	X	0	0	X	1
1	1	1	X	0	X	0	X	1	0	1	0	X	X
1	0	1	1	0	X	0	0	X	X	X	0	1	0
0	X	X	0	X	X	1	0	X	X	X	0	X	X
1	0	1	X	1	X	1	X	1	0	X	0	0	X
1	1	1	1	0	X	0	0	X	X	X	X	0	0

5.1. Selection of First Test Vector of Reordered Set. In the above test data, vector V_3 has the minimum number of do not care bits, that is, 4. Now this vector is minimum transition (MT) filled as shown below:

test vector

V_3	1	0	1	1	0	0	0	0	0	0	0	0	1	0

its corresponding weighted transition as per (1) is 38.

5.2. Reordering Remaining Test Vector with Columnwise Bit Filling. After placing the V_3 at first place and V_1 shifted to position of vector 3, the test set after first line selected and filled is as below:

test vectors after first vector reordered

R_1	1	0	1	1	0	0	0	0	0	0	0	0	1	0
V_2	1	**1**	1	X	0	X	0	X	**1**	0	**1**	0	X	X
V_3	1	X	1	**0**	0	X	X	0	**1**	X	0	0	X	**1**
V_4	**0**	X	X	**0**	X	X	**1**	0	X	X	X	0	X	X
V_5	1	0	1	X	**1**	X	**1**	X	**1**	0	X	0	**0**	X
V_6	1	1	**1**	1	0	X	0	0	X	X	X	X	**0**	0

Now the Hamming distance of remaining each test vector V_2, V_3, V_4, V_5, and V_6 from first reordered vector R_1 is 3, 3, 3, 4, and 2 as described in definition of Hamming distance and emphasized in above test set with bold letter. So V_6 is selected as next vector of reordered set. After placing V_6 as R_2, the columnwise bit filling is done.

Partially reordered test vectors

R_1	1	0	1	1	0	0	0	0	0	0	0	0	1	0
R_2	1	1	1	1	0	**0**	0	0	**0**	**0**	**0**	**0**	0	0
V_3	1	X	1	0	0	X	X	0	1	X	0	0	X	1
V_4	0	X	X	0	X	X	1	0	X	X	X	0	X	X
V_5	1	0	1	X	1	X	1	X	1	0	X	0	0	X
V_6	1	1	1	X	0	X	0	X	1	0	1	0	X	X

Now the Hamming distance of V_3, V_4, V_5, and V_6 from R_2 is calculated as 3, 3, 4, and 2. So V_6 is selected as R_3.

Partially reordered test vectors

R_1	1	0	1	1	0	0	0	0	0	0	0	0	1	0
R_2	1	1	1	1	0	0	0	0	0	0	0	0	0	0
R_3	1	1	1	1	0	0	0	0	1	0	1	0	0	0
V_4	0	X	X	0	X	X	1	0	X	X	X	0	X	X
V_5	1	0	1	X	1	X	1	X	1	0	X	0	0	X
V_6	1	X	1	0	0	X	X	0	1	X	0	0	X	1

Now the remaining all three V_4, V_5, and V_6 vectors have equal Hamming distance 3 from R_3. So each vector will be tried for its weighted transition as if columnwise bit filling is applied to it.

Test vector

R_3	1	1	1	1	0	0	0	0	1	0	1	0	0	0
V_4	0	**1**	**1**	0	**0**	**0**	1	0	**1**	**0**	**1**	0	**0**	**0**

Applying (1), the weighted transition is equal to 57 for this case.

Test vector

R_3	1	1	1	1	0	0	0	0	1	0	1	0	0	0
V_5	1	0	1	**1**	1	**0**	1	**0**	1	0	**1**	0	0	**0**

Test vector

R_3	1	1	1	1	0	0	0	0	1	0	1	0	0	0
V_6	1	**1**	1	0	0	**0**	**0**	0	1	**0**	0	0	**0**	1

The same way the possible weighted transitions for V_5 and V_6 are 67 and 23, respectively, as shown above. So V_6 is selected as the next test vector of reordered set. Repeating the reordering process until the last vector is reordered, the final reordered set is as shown bellow:

reordered test vectors

R_1	1	0	1	1	0	0	0	0	0	0	0	0	1	0
R_2	1	1	1	1	0	0	0	0	0	0	0	0	0	0
R_3	1	1	1	1	0	0	0	0	1	0	1	0	0	0
R_4	1	1	1	0	0	0	0	0	1	0	0	0	0	1
R_5	0	1	1	0	0	0	1	0	1	0	0	0	0	1
R_6	1	0	1	0	1	0	1	0	1	0	0	0	0	1

5.3. Difference Vector. The next step is to take the difference vector of two consecutive vectors in reordered set to increase the numbers of zeroes and hence data compression. The difference vector set is as shown below:

difference test vectors

R_1	1	0	1	1	0	0	0	0	0	0	0	0	1	0
D_2	0	1	0	0	0	0	0	0	0	0	0	0	1	0
D_3	0	0	0	0	0	0	0	0	1	0	1	0	0	0
D_4	0	0	0	1	0	0	0	0	0	0	1	0	0	1
D_5	1	0	0	0	0	0	1	0	0	0	0	0	0	0
D_6	1	1	0	0	1	0	0	0	0	0	0	0	0	0

5.4. Run Length Coding. The difference vector set applied the frequency directed run length coding as described in Section 3.3.

6. Comparison

In Table 2, the comparison of % compression, peak power, and average power for various test data processing scheme with FDR coding applied to test data of motivational example is shown in Table 2. Here the column 2 in Table 2 shows the results when test data is applied with MT filling and FDR coding. The peak power and average power is minimum in this case, but the compression is negative. Column 3 is for test data where do not care bits are filled with 0 s and without reordering, the difference vectors are created, and FDR is applied. The columns 4 and 5 represents the results of HDR-CBF-DV and WTR-CBF-DV. As it is seen from these results, the % compression is maximum in case of HDR-CBF-DV and WTR-CBF-DV, while average power is comparable in case of difference vector only, HDR-CBF-DV, and WTR-CBF-DV.

6.1. Experiment Results. For the WTR-CBF-DV, the working model is developed using MATLAB7.0 language and then for extensive experimental work, the C language is used. The experiments are conducted on a workstation with an Intel 2 GHz Core2Duo CPU T5750 with 3 GB of memory. The six largest ISCAS89 full-scan circuits have been considered for this experiment. For all ISCAS89 circuits, the test sets (with do not care) obtained from the Mintest ATPG program are used. Tables 3, 4, and 5 show the comparison of % compression, average power, and peak power for various ISCAS circuits' test data with FDR coding when test data applied the following processing prior to FDR coding.

(A) Do not care bits are MT filled, but no reordering is applied.

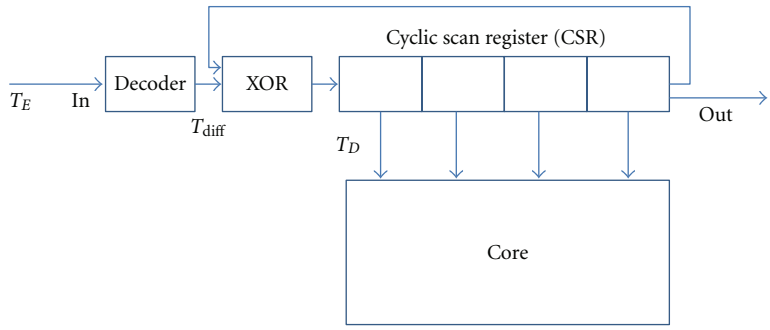

Figure 2: On-chip decoder for WTR-CBF-DV.

Table 2: Comparison of test data processing methods.

	MT Fill	Diff. Vector	HDR-CBF-DV	WTR-CBF-DV
% Compression	−2.381	7.1429	16.6667	16.6667
Peak power	38	81	82	82
Average power	23.8333	36.8333	42.1667	38.6667

Table 3: Comparison of % compression for various test data processing methods.

ISCAS circuit	Minimum transition fill	0 Filling + XOR [30]	Run based bit fill maximum limit	HDR-CBF-DV	2-D Reordering	WTR-CBF-DV
s5378	−12.31	48.02	52.36	62.33	59.66	62.15
s9234	−20.67	43.59	47.80	61.06	61.35	63.31
s13207	6.16	81.30	83.65	87.47	88.22	88.04
s15850	−17.91	66.22	68.18	72.84	73.96	73.38
s38417	−20.39	43.26	54.5	66.18	65.13	66.38
s38584	−8.90	60.91	62.49	64.79	66.08	65.21

Table 4: Comparison of average power for various test data processing methods.

ISCAS circuit	Minimum transition fill	0 Filling + XOR [30]	Run based bit fill maximum limit	HDR-CBF-DV	2-D Reordering	WTR-CBF-DV
s5378	3433	3526	3526	11133	7934	10344
s9234	3958	4022	4022	14382	13329	13492
s13207	7735	7887	7887	113890	78856	103400
s15850	13514	13659	13659	82421	71015	64275
s38417	117540	118080	118080	452860	486000	443030
s38584	85656	86305	86305	410240	423260	329110

Table 5: Comparison of peak power for various test data processing methods.

ISCAS circuit	Minimum transition fill	0 Filling + XOR [30]	Run based bit fill maximum limit	HDR-CBF-DV	2-D Reordering	WTR-CBF-DV
s5378	11519	12085	12085	13327	11769	12822
s9234	14092	15395	15395	17828	16106	17169
s13207	94879	110129	110129	128638	95541	125392
s15850	70875	84360	84360	96084	98903	96452
s38417	437884	514716	514716	644262	633561	660096
s38584	481158	530464	530464	550037	532809	551602

(B) Do not care bits are filled on the basis of run type, but no reordering is applied.

(C) Do not care bits are filled with 0 s, and difference vector is applied [30].

(D) HDR-CBF-DV applied.

(E) 2-D reordering is applied.

(F) WTR-CBF-DV applied.

7. On-Chip Decoder

Any of test data compression methods needs an on-chip decompressor, which loads compressed data from automatic test equipment (ATE) and restores the original test data. The decompressed test data will be transmitted to design under test. The WTR-CBF-DV is a test data processing method applied in conjunction with FDR coding. The same FDR decoder described in [5, 30] is used here. Figure 2 describes the decoder described in [30]. As in this approach, the difference vector is already used, the proposed WTR-CBF-DV does not require any extra on-chip area overhead.

8. Conclusion

In this paper, a scheme comprising of Hamming distance and weighted transition-based reordering (WTR), columnwise bit filling (CBF), and difference vector (DV) for test data compression is proposed. This scheme is applied to preprocess the test data before applying the FDR compression method. The proposed test data processing scheme improves the % compression compared to earlier methods described in literature. Moreover, this method increases the compression beyond the limit of maximum possible compression for run based bit filled data. The peak power and average power is a tradeoff with % compression, but still it is controlled using weighted transition-based reordering. The proposed scheme demands no extra on-chip area overhead compared to earlier methods in literature.

References

[1] M. Hirech, "Test cost and test power conflicts: EDA perspective," in *Proceedings of the 28th IEEE VLSI Test Symposium (VTS '10)*, p. 126, Santa Cruz, Calif, USA, January 2010.

[2] N. A. Touba, "Survey of test vector compression techniques," *IEEE Design and Test of Computers*, vol. 23, no. 4, pp. 294–303, 2006.

[3] A. Jas and N. A. Touba, "Test vector decompression via cyclical scan chains and its application to testing core-based designs," in *Proceedings of the IEEE International Test Conference (ITC '98)*, pp. 458–464, October 1998.

[4] A. Chandra and K. Chakrabarty, "Test data compression for system-on-a-chip using Golomb codes," in *Proceedings of the 18th IEEE VLSI Test Symposium (VTS '00)*, pp. 113–120, May 2000.

[5] 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*, vol. 52, no. 8, pp. 1076–1088, 2003.

[6] A. H. El-Maleh and R. H. Al-Abaji, "Extended frequency-directed run-length code with improved application to system-on-a-chip test data compression," in *Proceedings of the 9th IEEE International Conference on Electronics, Circuits and Systems (ICECS '02)*, vol. 2, pp. 449–452, September 2002.

[7] A. Chandra and K. Chakrabarty, "Reduction of SOC test data volume, scan power and testing time using alternating run-length codes," in *Proceedings of the 39th Conference on Design Automation*, pp. 673–678, June 2002.

[8] S. Hellebrand and A. Wrtenberger, "Alternating run length coding : a technique for improved test data compression," in *Proceedings of the 3rd IEEE International Workshop on Test Resource Partitioning*, Baltimore, MD, USA, October 2002.

[9] U. S. Mehta, K. S. Dasgupta, and N. M. Devashrayee, "Run-length-based test data compression techniques: how far from entropy and power bounds?—a survey," *VLSI Design*, vol. 2010, Article ID 670476, 9 pages, 2010.

[10] U. S. Mehta, K. S. Dasgupta, and N. M. Devashrayee, "Combining unspecified test data bit filling methods and run length based codes to estimate compression, power and area overhead," in *Proceedings of the IEEE Asia Pacific Conference on Circuits and Systems (APCCAS '10)*, pp. 40–43, Kuala-Lumpur, Malaysia, December, 2010.

[11] H. Fang, T. Chenguang, and C. Xu, "RunBasedReordering: a novel approach for test data compression and scan power," in *Proceedings of the IEEE International Conference on Asia and South Pacific Design Automation (ASP-DAC '07)*, pp. 732–737, Yokohama, Japan, January 2007.

[12] U. S. Mehta, K. S. Dasgupta, and N. M. Devashrayee, "Hamming distance based 2-D reordering with power efficient don't care bit filling: optimizing the test data compression method," in *Proceedings of the 9th International Symposium on System-on-Chip (SoC '10)*, pp. 1–7, Tampare, Finland, September 2010.

[13] U. S. Mehta, K. S. Dasgupta, and N. M. 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 *Proceedings of the 23rd International Conference on VLSI Design (VLSID '10)*, pp. 33–38, Bangalore, India, 2010.

[14] P. Girard, C. Landrault, S. Pravossoudovitch, and D. Severac, "Reducing power consumption during test application by test vector ordering," in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS '98)*, pp. 296–299, June 1998.

[15] K. Roy, R. K. Roy, and A. Chatterjee, "Stress testing of combinational VLSI circuits using existing test sets," in *Proceedings of the IEEE International Symposium on VLSI Technology, Systems, and Applications (ISVLSI '95)*, pp. 93–98, June 1995.

[16] I. P. V. Dabholkar, S. Chakravarty, and S. Reddy, "Techniques for minimizing power dissipation in scan and combinational circuits during test application," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 17, no. 12, pp. 1325–1333, 1998.

[17] H. N. J. M. P. Flores, J. Costa, H. Neto, J. Monteiro, and J. Marques-Silva, "Assignment and reordering of incompletely specified pattern sequences targetting minimum power dissipation," in *Proceedings of the 12th International Conference on VLSI Design*, pp. 37–41, January 1999.

[18] S. Chattopadhyay and N. Choudhary, "Genetic algorithm based approach for low power combinational circuit testing," in *Proceedings of the 16th International Conference on VLSI Design*, pp. 552–559, January 2003.

[19] H. Hashempour and F. Lombardi, "Evaluation of heuristic techniques for test vector ordering," in *Proceedings of the ACM*

Great lakes Symposium on VLSI (GLSVLSI '04), pp. 96–99, January 2004.

[20] A. S. D. Whitley, A. Sokolov, and Y. Malaiya, "Dynamic power minimization during combinational circuit testing as a traveling salesman problem," in *Proceedings of the IEEE Congress on Evolutionary Computation*, vol. 2, pp. 1088–1095, September 2005.

[21] S. Roy, I. S. Gupta, and A. Pal, "Artificial intelligence approach to test vector reordering for dynamic power reduction during VLSI testing," in *Proceedings of the IEEE Region 10 Conference (TENCON '08)*, pp. 1–6, Hyderabad, India, November 2008.

[22] Y. L. J. Wang, J. Shao, and Y. Huang, "Using ant colony optimization for test vector reordering," in *proceedings of the IEEE Symposium on Industrial Electronics and Applications (ISIEA '09)*, pp. 52–55, Kuala Lumpur, Malaysia, October 2009.

[23] S. K. Kumar, S. Kaundinya, and S. Chattopadhyay, "Particle swarm optimization based vector reordering for low power testing," in *proceedings of the 2nd International Conference on Computing, Communication and Networking Technologies (ICCCNT '10)*, pp. 1–5, Karur, India, July 2010.

[24] U. S. Mehta, K. S. Dasgupta, and N. M. Devashrayee, "Artificial intelligence based scan vector reordering for capture power minimization," in *Proceedings of the IEEE International Symposium on VLSI (ISVLSI '11)*, June 2011.

[25] S. J. Wang, Y. T. Chen, and K. S. M. Li, "Low capture power test generation for launch-off-capture transition test based on don't-care filling," in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS '07)*, pp. 3683–3686, May 2007.

[26] S. Kundu and S. Chattopadhyay, "Efficient don't care filling for power reduction during testing," in *Proceedings of the International Conference on Advances in Recent Technologies in Communication and Computing (ARTCom '09)*, pp. 319–323, October 2009.

[27] Z. Chen, J. Feng, D. Xiang, and B. Yin, "Scan chain configuration based X filling for low power and high quality testing," *IET Journal On Computers and Digital Techniques*, vol. 4, no. 1, pp. 1–13, 2009.

[28] J. L. Yang and Q. Xu, "State-sensitive X-filling scheme for scan capture power reduction," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 27, no. 7, Article ID 4544872, pp. 1338–1343, 2008.

[29] T. K. Maiti and S. Chattopadhyay, "Don't care filling for power minimization in VLSI circuit testing," in *Proceedings of THE IEEE International Symposium on Circuits and Systems (ISCAS '08)*, pp. 2637–2640, Seattle, Wash, May 2008.

[30] A. Chandra and K. Chakrabarty, "Frequency-directed run-length (FDR) codes with application to system-on-a-chip test data compression," in *Proceedings of the 19th IEEE VLSI Test Symposium*, pp. 42–47, May 2001.