ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 4, April 2015

Test data compression and optimal power seed selection for Scan Power Reduction

G.Pavithra Devi[1], K.S.Neelukumari [2]

PG Scholar, VLSI Design, P.A. College of Engineering and Technology, Coimbatore, India[1]

Assistant Professor, ECE, P.A. College of Engineering and Technology, Coimbatore, India[2]

Abstract: XOR network-based on-chip test compression schemes have been widely employed in large industrial scan designs due to their high compression ratio and efficient decompression mechanism. Due to the highly divergent power impact of distinct seeds appreciable power reductions in the decompressed test data can be attained through the pinpointing of the power-optimal seeds during the compression phase. This work explores the test data compression technique which searches for the power optimal seeds in the compression phase in order to reduce the scan power consumption. The seeds which are identified as power optimal alone will be decompressed in the decompression phase and the power optimization is based on the number of transitions between the seeds. The LFSR is used to generate the sequence of seeds in order to identify the power optimal one from the sequence of seeds which are fed into the XOR based decompression network. Experimental results confirm that the proposed technique delivers significant scan power reduction with negligible impact on the compression ratio.The simulation results are obtained using MODELSIM 6.3f and the power is analysed using XILINX 8.1 software.

Keywords: on-chip test compression, XOR decompression network, scan power reduction, seed selection.

I. INTRODUCTION

The increase in the integration level of today's circuits results in a rapidly growing volume of test data required to attain acceptable fault coverage. Higher test data volumes result in prolonged test application times and necessitate external automatic test equipments (ATEs) with larger test storage and higher bandwidth, significantly boosting test costs. Test compression techniques assume significant importance in ameliorating test costs, as they provide a means for squashing the test data volume and possibly reducing the tester bandwidth and the test application time, yet with no adverse impact on test quality.

The aforementioned benefits lead to a wide application of test compression techniques in industrial scan designs. Nonetheless, circuits with a large number of scan cells are confronted with the challenge of excessive scan power induced by the high transition density of scan cells. It has been widely reported that uncontrolled scan power dissipation significantly degrades the scan test quality by incurring chip damage and over screening-induced yield loss [13].

While reducing power for non-compressed test vectors constitutes a formidable research challenge as exemplified by the number of research works published in the area [12], the possibly conflicting compression goal elevates the challenge of power reduction to a highly challenging plane, as shown by the limited number of works on this topic. The attainment of either target necessitates the appropriate utilization of the unspecified bits (don't cares) in the test cubes. The identification of a don't care utilization strategy that maximally fullfills both constraints is thus essential for delivering a power-aware compression scheme.

On-chip test compression schemes based on XOR networks [1]-[2] have been widely employed in large industrial scan designs, due to their utmost compression ratio and simple decompression mechanism. Drastic power reduction yields can nonetheless be attained through a judicious seed selection strategy, if one considers that multiple legal seeds with highly divergent consequent power impact may typically exist in a linear compression scheme. Motivated by this observation, a power-aware test compression technique has been proposed in [5], which efficiently identifies seeds that result in a low level of toggling activities between scan slices of the test cubes.

Memory is considered to be one of the key factor in embedded systems. Because a larger memory influences the increased chip area, power dissipation and cost. Test data compression plays a crucial role reducing the testing time and memory requirements and also overcomes the automatic test equipment bandwidth limitations. However, BIST is not appropriate for logic testing because of its inadequate test coverage. There are also other techniques for test coverage with the disadvantage that structural information has to be provided. Test data can be reduced using the structural information by modifying the design. Test data compression algorithms can reduce the test data efficiently without facing any problems mentioned above. The original test data is compressed and stored in memory thus the memory size is reduced. The test data is compressed using the compression techniques with more efficiency. An on chip decoder is used for decoding the compressed test data from the memory and delivers the original uncompressed set of test vectors to the design under test. From the previous methods, it is observed that the switching activity is large and the power dissipates to a

DOI 10.17148/IJARCCE.2015.4421

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 4, April 2015

large extent. This paper is proposed to reduce the power dissipation by identifying the power optimal seeds during the compression phase.

II. PROPOSED METHOD

The proposed work includes the technique for the compression of test data, seed generation through the LFSR and the decompression technique after identifying the power optimal seeds. The proposed methodology is given in the Fig.1.

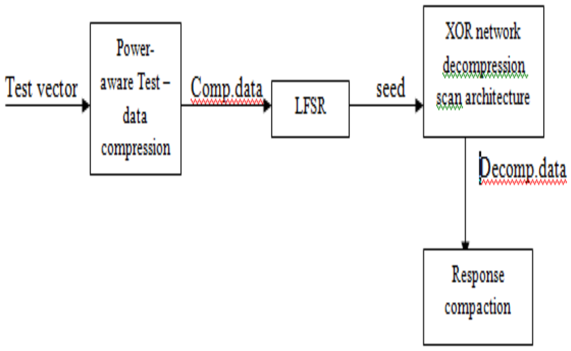
Test vector
Power-aware Test-data compression
Comp.data
LFSR
seed
XOR network decompression scan architecture
Decomp.data
Response compaction

Fig.1 Proposed methodology

A. Test data compression

Test data can be reduced using the structural information, by modifying the design. Test data compression algorithms can reduce the test data efficiently. The over view of a test compression methodology is shown in Fig.2

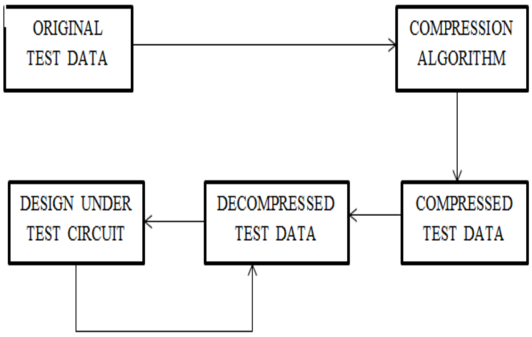

Fig.2 Test data compression.

The proposed scheme is a modified Golomb coding method, in which both 0* and 1*are encoded. The goal is to reduce both *TD* and *TE* with this encoding scheme.

1)Basic idea:

Most of the research results on test data compression focus on reducing the amount of encoded test data (*TE*). However, this approach may actually lead to a larger volume of applied test data (*TD*), and thus increases test time. In order to reduce both *TD* and *TE*, the proposed scheme starts from a compact test set, in which the number of test vectors is near minimum. The proposed method tries to change each vector so that the new vector has longer runs of 0's and 1's but the fault coverage is not sacrificed. Instead, the proposed scheme tries to exploit consecutive 0's and 1's inside test patterns for compression. There are two possible types of test set: either every bit in a test vector is fully specified, or some bit are not specified. A test vector with unspecified bits is usually referred to as a test cube. If the initial test set is not fully specified, it will be easier to compress. However, the size of recovered test data *TD* will be larger, and thus the test application time is longer. An example of test set with test cubes is shown in Fig.3. Since the don't- care bits in test cubes can be randomly assigned to 0 or 1, it is possible to produce longer runs of 0's and 1's. As a result, the compression rate can be greatly increased.

Test cubes		Test vectors
xxx00xxx1xxxxxxxx		00000000111111111
xxx10xxx1xxxxxxxx	⇨	11110000111111111
xxxx00100xxxxxx1x		00000010000000011
xxxx10100xxxxxx1x		11111010000000011

Fig. 3. Assign 0 or 1 to don't-care bits in test cubes.

In this paper, we adopt a greedy approach to assign don't-care bits from left to right, in which either '0' or '1' is selected to make the current run longer. According to this heuristic, the test cubes in Fig. 3 are assigned to the test vectors as shown. However, some test cubes cannot be efficiently handled with this heuristic. Consider the third and forth test vectors in Fig. 3, in which there is a 1 sitting between two runs of 0's. In these cases, there are short runs that are difficult to be compressed.

In order to push the compression rate to an even higher level, the idea used in Forced Pair-Merging (FPM) algorithm is applied here. This algorithm was first presented for static test set compaction. It used the concept of "raise bit" to change only one bit each time without affecting essential faults. When a 1 is changed to 0, or vice versa, it is necessary to check if the new test vector can detect the same number of essential faults. Whenever the coverage of essential faults is not affected, the bit is raised to *x* (*i.e.*, don't-care), so that the new test cube may be merged with other test cubes. In the proposed method, however, it is not necessary to merge a test cube with other patterns to reduce the size of a test set. Instead, we look for those bits whose inversions (*i.e.*, 0 →1 or 1 →0) can create a longer run of 0* or 1*.

For example, consider test vectors 3 and 4 in Fig. 3. The single 1's locating between two runs of 0* are good candidates for bit inversion. For each candidate bit inversion, we need to check if the new test vector created by this bit inversion still detects the same set of essential faults. If the fault coverage is reduced, the bit inversion will not be accepted. This procedure is referred to as the Forced Bit-Inversion (FBI) henceforth. With a good don't-care bits assignment heuristic and the FBI algorithm, we can achieve a very high compression rate. In the proposed encoding scheme, a test vector is first changed to alternate runs of 0* and 1*, and then the length of each run is encoded in the same way as Golomb code. Once the value of the first run is known, the other runs are also known. The encoding mechanism of the two-value (2-V) Golomb

DOI 10.17148/IJARCCE.2015.4421

code is similar to the alternating run-length coding scheme. An example of the encoding scheme is shown in TABLE I. In this example, we have m = 4, n = 45, CR = 10/45 = 0.22. R is the value of current run.

TABLE I A 2-VALUE GOLOMB CODING EXAMPLE

T_D	000	111111	0	1111	00000	1111111	00
	L_1=3	L_2=6	L_3=1	L_4=4	L_5=5	L_6=7	L_7=2
T_E	011	1010	001	1000	1001	1011	010
R	0	1	0	1	0	1	0

2) Compression procedure and FBI algorithm:

The proposed compression method works as follows.

Step 1**:** Generate an initial test set $T_{initial.}$

Step 2**:** If all test patterns in $T_{initial}$ are deterministic (*i.e.*, no test cubes), go to step 4.

Step 3**:** Assigned 0’s and 1’s into don’t-care bits to create longer runs of alternate 0* and 1*.

Step 4**:** Eliminate redundant test patterns, and the test set is *T*.

Step 5**:** Apply FBI algorithm to maximum run lengths in each test pattern. The result is a new test set *T*’.

Step 6**:** Reorder patterns in *T*’ to generate a sequence *TD* with maximum 0* and 1*.

Step 7**:** Encode *TD* with two-value Golomb codes. Step 5 is the FBI algorithm, which inverts some bits in the test patterns.

3) FBI Algorithm:

```
FBI(T) {
T’ = ∅;
swi = the number of switching in pattern pi;
while (T != ∅) {
select pattern pk ∈□T with maximum swk;
change_pattern(pk); // Similar to raise bit
update_essential_fault_list();
T’ = T’ ∪□{pk}; T = T □□{pk};
}
check_essential_fault_list();
}
return T’;
```

The FBI algorithm is outlined above. In each iteration of the **while** loop, the algorithm picks the pattern with the maximum number of bit transitions from the test set. The algorithm tries to invert some bits to get longer runs of 0’s and 1’s. For example, 00100 can be changed to 00000 or 00111, while 11011 can be changed to 11111 or 11000. After the bit inversion, we must update the essential fault list for this test pattern again. The function update_ essential_fault() is for that purpose. When the above process is finished, the reduction of essential faults has to be checked. If there is no reduction, the move is accepted; otherwise, it is discarded.

B. Power optimal seed identification

The compressed test data will be the initial seed for the LFSR to generate the sequence of seeds to be identified. In this seed sequence, the seed that involves less switching activity will be considered as the power-friendly seeds. This identification is carried out for all the seeds and the power friendly seeds are sent to the decompression network. The LFSR generated seed sequence will be sent into the decompression network to produce the decompressed data which will be the inputs to the scan chains. Each seed will have their corresponding decompressed data which is shifted to next bit of the scan chain when the next seed generates the decompressed data. The transitions of the decompressed data of seed1 and seed2 are compared with each bit.

The seed which has the maximum transitions will be considered as the seed that consumes maximum power. Hence that particular seed will be eliminated in the sequence and the seed that consumes less power with minimum number of transitions will be considered as the power-optimal seed. The scan power impact of distinct seeds exhibits a high variation; therefore, it is possible to attain power reductions by selecting the power-optimal seed during the compression process.

As illustrated in Fig.4, assuming the XOR network is employed as the decompression hardware; two sets of seeds are identified for the original test cube. The test cube generated from Seed 1 contains a total of 17 transition patterns in adjacent scan flip-flops, whereas the one generated from Seed 2 has only 9 transition patterns. Selecting Seed 2 for compression delivers a 50% reduction almost in scan power with no impact on the compression ratio whatsoever.

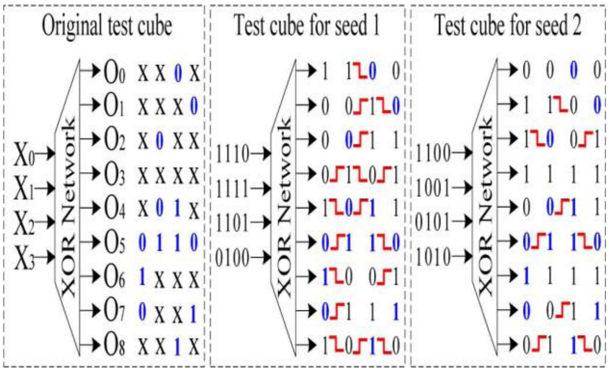

Fig.4 Identification of power optimal seeds

C. Decompression Architecture

The power-aware test compression gives the compressed data as the output which provides the seed to LFSR to generate the sequence. This generated seeds are sent into the XOR based decompression network. The decompression architecture describes the operation of retaining the original data from the compressed data. Fig.5 illustrates the general structure of the decompression hardware through the scan cells in the XOR networks. It also depicts a possible implementation of the decompression scheme, in which the S bit data provided by the tester is decompressed to generate Si bit data to the internal scan chains at every clock cycle.

DOI 10.17148/IJARCCE.2015.4421

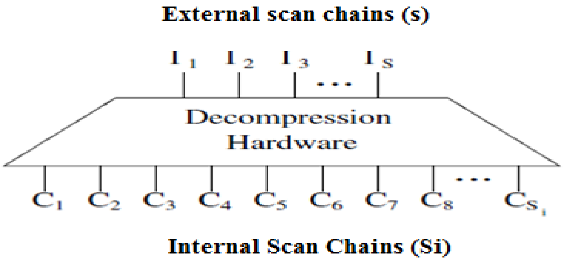

Fig.5 Decompression Architecture

This mechanism of feeding the seed value into the decompression network is illustrated in the Fig.6.As shown in the Fig.6, once the seed is generated in the LFSR according to the compression algorithm, the seed is fed into the decompression network.

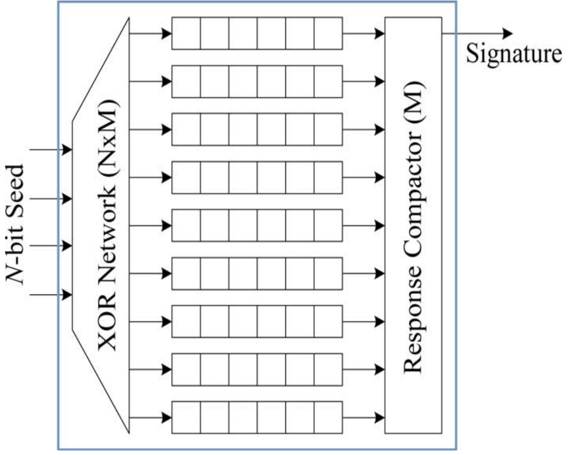

Fig.6 Decompression network

In each scan cycle, the XOR network takes as inputs an N-bit seed from the ATE and decodes it to an N-bit test slice. All bits of a test slice are generated in parallel and are applied to the actual scan chains simultaneously. The scan-out values of internal scan chains are compacted in parallel using an XOR-based response compactor or a multi-input signature register (MISR). The decompression process is independent of the response-compaction mechanism, thus enabling the designers to choose the most appropriate compaction hardware according to the needs. The fixed-length to fixed-length decompression mechanism offered by the XOR network-based decompression scheme eliminates the need for complicated synchronization methodologies between ATE and the decompression hardware, making it highly practicable for a variety of industrial applications.

The decoding mechanism of the XOR network is essentially a linear transformation from the seed to the test slice. Each bit of the test slice is a linear combination of multiple seed bits.

Thus the compression process consists of the identification of an appropriate seed vector that can successfully generate all the specified bits of the original test slice through the linear transformation. Test slices with few specified bits can be reconstructed from very short seeds, thus delivering appreciable compression ratios.

III. EXPERIMENTAL RESULTS

The proposed scheme has been implemented for effectiveness validation. Our implementation uses MODELSIM for fault simulation and Xilinx tool for the power report analysis. The linear compression technique in [2] is used as a comparison baseline. The power-aware compression technique and the proposed technique are compared to the baseline approach, and their impact on scan power and compression ratio are reported. The comparison illustrated in the TABLE II gives the clear scenario that the scan power is reduced drastically by decompressing the power-friendly seeds rather than all the generated seeds.

TABLE II. COMPARISON OF SCAN POWER REDUCTION

Circuits	#PI	#FF	Power Analysis	
			With all seeds	With power-friendly seeds
s13207	62	638	627	217
s15850	77	534	584	224
s35932	35	1732	1535	312

IV. CONCLUSION

The compression scheme employed in this process offers more efficiency than the other compressed techniques. In this paper, a two-value Golomb coding method is proposed to encode test patterns. This method, when combined with the FBI algorithm, achieves better compression rate than other compression methods besides the volume of decompressed data is always smaller. As a result, test application time can be minimized too. On-chip test compression schemes based on XOR networks have been widely employed in large industrial scan designs, due to their utmost compression ratio and simple decompression mechanism. Thus the decompression network is employed using the XOR networks. The proposed scheme extracts and embeds the compression constraints imposed by the XOR network. It can be observed that significant power reduction can be delivered by both power-aware compression schemes compared to the baseline.

A power aware linear compression scheme is proposed in this paper, which exploits the flexibility in the seed space and identifies power-optimal seeds during the compression process. Thus the scan power is reduced by decompressing the seeds with minimum number of transitions in the decompression network. The testing of sequential circuits with the power optimal seeds in the compression and the decompression phase will reduce the power consumption of the scan chain rather than decompressing all the seeds generated by the LFSR after the compression phase. Thus identifying the power-optimal seeds during the compression phase and decompressing those power optimal seeds alone will reduce the power consumption drastically.

DOI 10.17148/IJARCCE.2015.4421

REFERENCES

[1]. Balakrishnan K. and Touba N.(2006),'Improving linear test data compression',IEEE Trans. Very Large Scale Integr.(VLSI)Syst.,Vol. 14, no.11, pp. 1227–1237.

[2]. Bayraktaroglu I. and Orailoglu A. (2003),'Concurrent application of compaction and compression for test time and data volume reduction in scan designs', IEEE Trans.Comput.,Vol. 52, no. 11, pp.1480–1489.

[3]. Butler K.M. and Hetherington G.(2004), 'Minimizing power consumption in scan testing: Pattern generation and DFT techniques',in Proc.ITC,pp. 355–364.

[4]. Chandra and Chakrabarty K.(2001),'System-on-a-Chip test-data compression and decompression architectures based on golomb codes',IEEE Trans. Comput.-Aided Design Integr.Circuits Syst.,Vol. 20, no.3, pp. 335–368.

[5]. Chandra A. and Chakrabarty K., "Test data compression for system-on-a-chip using Golomb codes," IEEE VLSI Test Symposium, 2000, pp. 113-120.

[6]. Chen M. and Orailoglu A.(2009),'Scan power reduction in linear test datacompression scheme', in Proc. ICCAD,pp. 78–82.

[7]. Dabholkar V. and Reddy S.M.(1998),'Techniques for minimizing power dissipation in scan and combinational circuits during test application',IEEE.Trans.Comput.-Aided Design Integr. Circuits Syst.,Vol. 17, no. 12, pp. 1325–1333.

[8]. Girard P. (2002), 'Survey of low-power testing of VLSI circuits', IEEE Design Test Comput., Vol. 19, no. 3, pp. 80–90.

[9]. Gonciari P.T. and Al-Hashimi B.M.(2003), 'Variable-length input huffman coding for system-on-a-chip test',IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.,Vol.22,no.6,pp.783–796.

[10]. Jas and Ghosh-Dastida J.(1999),'Scan vector compression/decompression using statistical coding', in Proc. DAC, pp.25-29.

[11]. Krishna and Touba N.(2001),'Test vector encoding using partial LFSR reseeding' in Proc. ITC , pp. 885–893.

[12]. Rajski J. and Mukherjee N.(2004),'Embedded deterministic test,"IEEE Trans.Comput.-Aided Design Integr. Circuits',Syst.,Vol. 23, no. 5,pp. 776–792.

[13]. Sankaralingam R. and Oruganti R.(2000),'Adapting scan architecture for low power operation',in Proc. VTS,pp. 35–40.

[14]. Sinanoglu O. and Orailoglu A. (2003),'Modeling scan chain modifications for scan-in test power minimization' in Proc. ITC, pp. 602–611.

[15]. Sun J. and Fujiwara H. (2007), 'Reconfigured scan forest for test application cost, test data volume and test power reduction', IEEE Trans. Comput., Vol. 56, no. 4, pp. 557–562.

DOI 10.17148/IJARCCE.2015.4421