Using Multiple Expansion Ratios and Dependency Analysis to Improve Test Compression

Richard Putman[1,2] and Nur A. Touba[2]

[1]Cirrus Logic, Inc.
2901 Via Fortuna, Suite 100
Austin, TX 78746
Richard.Putman@cirrus.com

[2]Computer Engineering Research Center
Dept. of Electrical and Computer Engineering
University of Texas, Austin, TX 78712
touba@ece.utexas.edu

Abstract

A methodology is presented for improving the amount of compression achieved by continuous-flow decompressors by using multiple ratios of scan chains to tester channels (i.e., expansion ratios). The idea is to start with a higher expansion ratio than normal and then progressively reduce the expansion ratio to detect any faults that remain undetected. By detecting faults at the highest expansion ratio possible, the amount of compression can be significantly improved compared with conventional approaches. The expansion ratio is progressively reduced by concatenating scan chains together using MUXes to make fewer and longer scan chains. Selecting which scan chains to concatenate is done by using a dependency analysis procedure that takes into account structural dependencies among the scan chains as well as free-variable dependencies in the logic driving the scan chains to improve the probability of detecting faults. Results for applying the proposed approach to industrial designs using various types of decompressors indicate significant improvements in compression are possible.

1. Introduction

As technology continues to shrink and design complexity increases, the amount of test data volume needed to maintain a high quality of test is also increasing. This is due to the following factors: the designs are larger and have a larger number of gates, and, therefore, have a larger number of faults to test; and the smaller geometries require additional fault models, like the delay fault models, in order to detect different types of defects. These fault models go above and beyond the standard stuck-at fault model, and, in turn, require additional test patterns. For example, typical transition delay pattern sets are 3-5X the size of the stuck-at pattern set for the same design.

In conventional semiconductor manufacturing, chips are tested by external testers, referred to as automatic test equipment (ATE). These external testers are limited by their I/O channels, memory depth, and clock speed. As test data volume grows, test engineers are running out of memory to contain the increased number of test patterns, and since the device under test (DUT) can typically only be tested as fast as the data can be transferred from the tester, the more test data volume there is, the longer the test time, and thus the higher the manufacturing cost. Two undesirable solutions to the memory depth limitation are to truncate the pattern set, which sacrifices test quality, or to load a second set of test patterns into memory, which can be very expensive time-wise.

A more viable approach to reducing test data volume to avoid hitting the tester memory depth limitation, and, more importantly, to reduce test time is to use test data compression. A number of test compression schemes have been developed and are being used in industrial designs. A survey of the different techniques can be found in [Touba 06]. Most schemes are based on continuous-flow decompression, a process in which data is expanded from a smaller number of tester channels to fill a larger number of scan chains in each clock cycle. By increasing the number of scan chains, each scan chain becomes shorter, resulting in fewer clock cycles being needed to load and unload them and thereby reducing test time. The ratio of scan chains to tester channels is called the "expansion ratio". The higher the expansion ratio, the greater the potential amount of compression, but the smaller the space of encodable test vectors. Conventional continuous-flow decompression techniques use a fixed expansion ratio which is selected at a level where all or almost all faults can be detected by encodable test vectors. Generally, a "serial" or "bypass" mode which allows the application of uncompressed test vectors is also implemented to provide a way to apply "top-off" vectors to detect any faults that cannot be detected by encodable test vectors as well as to support diagnosis. Uncompressed vectors applied in serial mode require much more tester storage and test application time compared with compressed test vectors. Consequently, to maximize the overall compression, conventional approaches must select a low enough expansion ratio to ensure that very few vectors need to be applied in serial mode.

This paper proposes a methodology for using multiple expansion ratios to improve the overall amount

25th IEEE VLSI Test Symmposium (VTS'07)
0-7695-2812-0/07 $20.00 © 2007 IEEE

of compression. The idea is to start with a higher expansion ratio than normal and then progressively reduce the expansion ratio to detect any faults that could not be detected at higher expansion ratios. By detecting each fault at the highest expansion ratio that it can be detected, the amount of compression can be significantly improved compared with conventional approaches that use a single expansion ratio. The expansion ratio is progressively reduced by concatenating scan chains together using MUXes to make fewer and longer scan chains. Another key idea in the paper is to exploit the flexibility in choosing which scan chains to concatenate together. The scan chains that are concatenated together are selected in a way that maximizes the potential for detecting additional faults. This is done by using a dependency analysis procedure that takes into account structural dependencies among the scan chains as well as free-variable dependencies in the logic driving the scan chains. Each bit on the tester can be considered as a "free-variable" which can be assigned either a 0 or 1. Free-variable dependencies occur when the contents of two scan cells depend on the same free-variable on the tester. By analyzing all of the dependencies, the proposed procedure determines which scan cells to concatenate in order to minimize potential conflicts during ATPG (automatic test pattern generation). This increases the probability of detecting more faults at higher expansion ratios.

The proposed methodology does not have any impact on scan cell ordering/stitching. A conventional design flow can be used for obtaining the initial design with the maximum expansion ratio. The proposed methodology only involves inserting MUXes in the design to provide the ability to concatenate scan chains for different expansion ratios. Note that this is already conventionally done in most commercial test compression schemes to provide for a serial mode. In fact, regardless of the number of expansion ratios used, the proposed method does not use any additional MUXes over what is already used for serial mode. The only difference is that for serial mode, only one control signal is used, but for this method, c control signals are required, where c is the number of expansion ratios. Hence, implementing the proposed methodology is very simple and adds very little overhead. It can be used in conjunction with a number of existing test compression schemes including Illinois scan [Hamzaoglu 99], reconfigurable fanout networks [Pandey 02], [Samaranayake 03], [Sitchinava 04], [Tang 03], Virtualscan [Wang 04], combinational linear decompressors [Bayraktaroglu 03], [Mitra 06], and sequential linear decompressors [Könemann 01], [Rajski 04], [Dutta 06].

There has been some earlier work that also tries to minimize dependencies to reduce conflicts. In [Samaranayake 03], structural analysis is used during scan cell ordering to minimize potential conflicts for a fanout network based decompressor. The structural analysis used here is similar in nature, but is used for scan chain concatenation and not scan cell ordering. The scan cell order is assumed to be optimized for other criteria (e.g., layout area) and is not modified. Moreover, the proposed method also takes into account more complex free-variable dependence and can be used with decompressors containing XOR gates. In [Shah 04], scan chains are grouped together for Illinois scan with multiple inputs based on compatibility analysis on a set of test patterns. The proposed approach is also based on grouping together scan chains, but the dependency analysis is based on structural analysis and takes into accounts more complex free-variable dependence.

The remaining portion of this paper is organized as follows. Section 2 discusses multiple expansion ratios and how they are implemented. Section 3 covers dependency analysis and the cost function. Section 4 describes how the cost function is used. Section 5 reports the experimental results. Conclusions are given in section 6.

2. Multiple Expansion Ratios

To implement this multiple expansion methodology, the first step is to create the first expansion ratio by inserting a large number of scan chains into a given design, with lockup-latches at the end of each chain to avoid clock-skew problems. The number of scan chains inserted depends on the amount of compression desired. Here, the more scan chains that are inserted, the higher the overall compression ratio; however, the trade-off is the extra amount of area for the decompressor logic in front of each additional scan chain, along with the few MUXes needed to reconfigure the scan chains into different expansion ratios and the associated routing.

Then, after the scan chains have been created, some logic, depending on the decompressor that is used, is added in front of every scan chain. For example, if combinational linear decompression was being used with 2-input XOR gates, then one 2-input XOR gate would be placed in front of every scan chain. On the other hand, if Illinois scan was being used, then each scan chain would be driven by just a fanout network.

The second expansion ratio, and all subsequent follow-on expansion ratios, is created by concatenating pairs of scan chains together using MUXes. In this configuration, each follow-on expansion ratio is one-half of the previous expansion ratio. Figure 1 shows how this can be implemented by using MUXes. *X1-X4* are tester channels driving a large set of scan chains. Only the first four scan chains are shown in the figure. The control lines to the MUXes come from two registers (alternatively they could come from primary inputs).

When both control signals are low, each scan chain is driven by its own XOR gate. When the first control signal

25th IEEE VLSI Test Symmposium (VTS'07)
0-7695-2812-0/07 $20.00 © 2007 IEEE

goes high, then the XOR gates at the beginning of scan chains 2 and 4 are disconnected, and the beginning of scan chain 2 is connected (via a MUX) to the end of chain 1. Likewise, the beginning of chain 4 is connected to the end of chain 3. This reconfiguration reduces the expansion ratio by a factor of 2. Next, if the second control signal is also brought high, then in this example the XOR gate in front of chain 3 is disconnected, and the beginning of chain 3 is connected, to the end of chain 2. With this reconfiguration, the expansion ratio is reduced by another factor of 2.

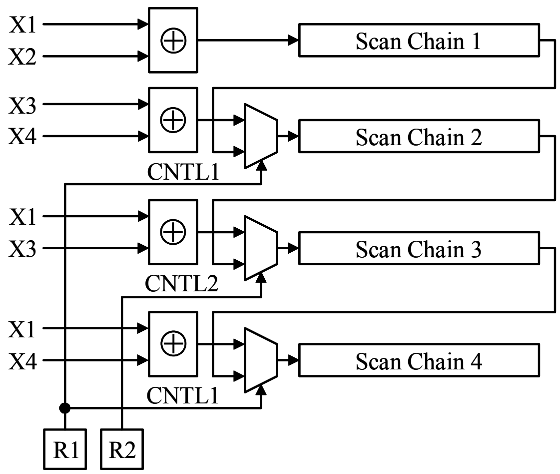

Figure 1. Multiple Expansion Ratio Implementation

Note that there are many different possible combinations of scan chain pairs to join together to form a new expansion ratio. Additionally, there are two possible orders in which a given pair of scan chains (*A* and *B*) can be concatenated (*A* could come first, or *B* could come first). Which scan chains are joined together and the order in which any pair is concatenated can have a significant impact on the overall test compression. In Fig. 1, if a random concatenation methodology was used, then scan chains 1 and 2 might get joined together as well as 3 and 4. In this case, the two scan chains created for the new expansion ratio are driven by $X1 \oplus X2$ for the first combined chain and $X1 \oplus X3$ for the second combined chain. Here, for each scan slice, both scan chains will be depending on the same free-variables coming from tester channel *X1*. To avoid this free-variable dependency, it would be better to concatenate scan chains 1 and 3 together and 2 and 4 together so that there is no free-variable dependency: in this case, the first combined chain will be driven by $X1 \oplus X2$, and the other will be driven by $X3 \oplus X4$. This illustrates how the degree of freedom in choosing which scan chains are concatenated together can be used to maximize the overall compression. A systematic approach for exploiting this is described in the following section.

3. Dependency Analysis

Conflicts that can prevent a fault from being detected at a particular expansion ratio arise when a fault requires a set of scan cells to have certain values, but the scan cells cannot be loaded with those values because some of the scan cells depend on the same free-variable. This type of conflict does not arise in serial (uncompressed) mode because in that case each scan cell depends on a unique free-variable.

There are two types of dependencies that both must be present in order to have a conflict. One is that detecting a fault must structurally depend on two scan cells, and the other is that the two scan cells must depend on the same free-variable. This is a necessary but not sufficient condition for a conflict. The dependency analysis proposed here is based on minimizing the number of potential conflicts by trying to minimize the number of scan cell pairs that have both structural and free-variable dependency present. A novel cost function is used which takes into account not only the existence of dependency, but also the degree of dependency. The cost function is then used as a greedy heuristic to guide the selection of which scan chains to concatenate together.

Linear Equations for Scan Cells

$SC_1 = X_1 \oplus X_2$ $\quad$ $SC_3 = X_3$

$SC_2 = X_2 \oplus X_3$ $\quad$ $SC_4 = X_3$

Degree of Free-Variable Overlap

$F_{1,2} = 1/3$ $\quad$ $F_{1,3} = F_{1,4} = 0$

$F_{2,3} = F_{2,4} = 1/2$ $\quad$ $F_{3,4} = 1$

Figure 2. Example of Calculating Degree of Free-variable Overlap

For a continuous-flow linear decompressor, each scan cell depends on a linear combination of the free-variables on the tester. Illinois scan is a degenerate case where each scan cell depends on a single free-variable. In the proposed cost function, the degree of free-variable dependency is measured by the number of free-variables the linear equations for two scan cells on two different scan chains have in common divided by the total number of distinct free-variables present across both of their linear equations. Consider the example shown in Fig. 2. Scan cells SC_1 and SC_2 have one free-variable in common (X_2) and they have a total of three distinct free-variables present across both of their linear equations (X_1, X_2, X_3), hence the degree of free-variable overlap is computed as 1/3. The maximum amount of free-variable overlap possible is 1 which occurs when two scan cells will always have the same value. For Illinois scan, free-variable overlap will always be either 0 or 1 since every scan cells depends on only a single free-variable.

However, for linear decompressors that contain XOR gates, the free-variable overlap can be a fraction. The higher the free-variable overlap is, the less flexibility there is to independently control the value of two scan cells.

The second part of the cost function is structural dependency. The degree of structural dependency is measured by looking at logic cone overlap. A one-time backwards trace from endpoints consisting of all primary outputs and the functional data input of all scan cells is performed, and then all of the scan cells driving the data cones of these endpoints are identified. Then a one-time record can be created to use with the cost function algorithm that keeps track of the number of times a given scan cell is in a logic cone with another scan cell. This one-time cone analysis is relatively quick to perform.

The dependency cost function for a pair of scan chains is computed as follows:

$$\text{Cost for Pair of Scan Chains} = \sum_{i,j} (F_{ij}^{\,e} * S_{ij}^{\,k})$$

where F_{ij} is the degree of free-variable overlap and S_{ij} is the amount of logic cone overlap between scan cell i in one scan chain and scan cell j in the other scan chain. These numbers are multiplied together so that if either value is zero, then the cost is zero since there is no chance for a conflict. The variables e and k are added as tuning parameters to adjust the relative weighting of the two parts of the cost function to optimize the results for different circuits and different test compression schemes. These values can be calibrated for a particular test compression scheme. In Fig. 3, the compression ratio that was obtained for different values of e and k were plotted for Design A (which is one of the circuits used in the experimental results reported in Sec. 5). Here it can be seen that the compression ratio varied some with different values of e and k but the results tended to be better where e was greater than or equal to 1 and k was less than or equal to 1. This held true for all three test cases over the three compression schemes that were used. The peak in Design A was 12.4X compression for the 2-input XOR combinational only decompressor.

4. Using Dependency Cost Function

The cost function can be used to build a new expansion ratio by selecting which pairs of scan chains to concatenate using a greedy procedure. The first task is to find the two scan chains that have the highest dependency cost when compared with each other. To eliminate their effect on each other, they are concatenated together to form one scan chain. For the initial highest cost scan chain pair, the second scan chain is just added to the end of the first scan chain. Then, the next highest cost scan chain pair is found. For this pair, and all subsequent pairs, both orders (one scan chain before the other and vise-versa) are considered. The cost function is applied for both orderings against each of the previously concatenated scan chains pairs for the new expansion ratio. The costs for both orders are summed up and compared. The order resulting in the lowest cost is selected. This continues until all scan chains of the previous expansion ratio have been concatenated. This procedure is repeated for each new expansion ratio.

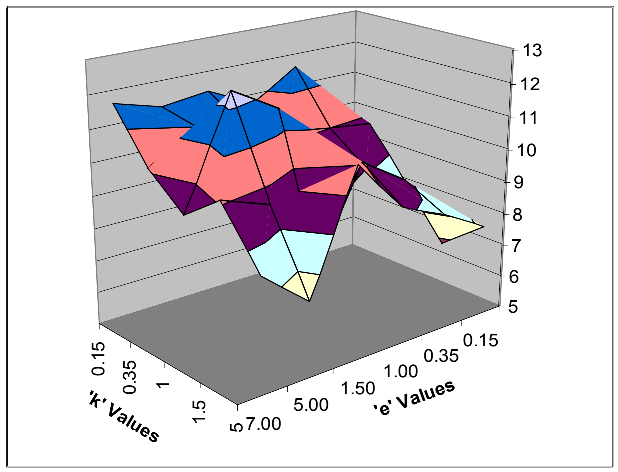

Figure 3. Plot of Compression for *Design A* using Different Values of e and k

5. Experimental Results

To demonstrate the viability of the proposed methodology, experiments were performed on three industrial designs. Table 1 shows the design details, including the number of scan cells and stuck-at faults in each design. The table also includes the number of full serial ATPG patterns required for complete coverage of the detectable faults. The three designs chosen range from a small mixed-signal chip with 4,019 scan cells up to a larger mostly digital chip with 30,776 scan cells. All of the pattern count results presented in this paper, including the ones in Table 1, were generated using a commercially available ATPG tool configured to use a high merge value to maximize dynamic pattern compaction and a limit for the minimum number of faults detected per pattern ranging from one to three.

Table 1. Design Details

Design	Scan Cells	Faults	Full Serial Pattern Count
Design A	4019	147,118	347
Design B	13557	696,514	633
Design C	30776	1,417,145	2743

5.1 Compression Ratio Results

Table 2 shows the compression results for the designs listed in Table 1. The designs are indicated in the first column, and the initial number of scan chains inserted into the design are listed in the third column. Three types of decompression schemes were tested, and they are noted in column 2. The first type is the Illinois, or broadcast, scan decompressor, and the second type is purely combinational decompression, using a single 2-input XOR gate at the beginning of every scan chain. The third type is the limited dependence sequential linear decompressor proposed in [Dutta 06] that uses two “tester slice registers” and a 2-input XOR gate at the beginning of every scan chain. For each decompressor type, 8 external tester channels were expanded to fill the scan chains.

Column four shows the results for a single expansion ratio and a single configuration of the logic driving the scan chains. Any faults not detected with this configuration are detected with top-off vectors in serial mode. Column five shows the results for a single expansion ratio and 4 configurations, where the logic driving the scan chains in each configuration is different. In this case, the ATPG is run on each configuration until it can’t detect any more faults. The leftover faults are passed on to the next configuration until all configurations have been used; then, top-off serial patterns are generated for any faults that still have not been detected. Columns six and seven both use a single fixed configuration, but use multiple expansion ratios. The difference between the two columns is that the technique represented by column six generates its second, and subsequent, expansion ratios, and the associated scan chains, by simply concatenating adjacent scan chains, whereas the method reported in column seven uses the dependency analysis procedure described in Sections 3 and 4 to determine which scan chain pairs are joined together to form the new expansion ratios.

As shown by the results in Table 2, in every case, for all three designs, going from a decompressor that uses a single expansion ratio with a single configuration to one that uses multiple expansion ratios (even one that uses a simple method for forming the follow-on expansion ratios) with a single configuration shows a significant improvement in test data volume compression. Also, using the simply created multiple expansion ratios was at least equal to, but most often significantly better than, using a single expansion ratio with multiple configurations. Additionally, as shown in the comparison of columns six and seven, the compression ratio for multiple expansion ratios created by dependency analysis was always greater, and usually, significantly greater, than the compression ratio resulting from the use of multiple expansion ratios, where the follow-on expansion ratios were created in a simple greedy fashion.

Table 2. Test Compression Results

Design	Decomp Type	Num of Scan Chains	Compression Ratio			
			1 Expan Ratio, 1 Config	1 Expan Ratio, 4 Config	4 Expan Ratios, 1 Config	4 Expan Ratios, Depend Analysis, 1 Config
A	Illinois	768	1.1	2.0	2.7	4.6
	xor2 comb	768	1.7	1.9	4.5	12.4
	xor2, 2reg	768	2.4	2.5	7.8	13.1
	Illinois	1024	1.2	1.6	5.7	6.4
	xor2 comb	1024	1.6	1.7	2.9	10.3
	xor2, 2reg	1024	1.9	2.9	2.9	14.8
B	Illinois	768	1.6	3.7	8.5	15.0
	xor2 comb	768	7.4	7.6	14.9	21.9
	xor2, 2reg	768	11.8	16.5	23.5	26.6
	Illinois	1024	1.5	2.2	9.8	16.1
	xor2 comb	1024	5.5	7.0	8.8	17.3
	xor2, 2reg	1024	13.3	20.3	26.9	28.2
C	Illinois	768	12.4	27.5	36.2	36.4
	xor2 comb	768	25.2	24.9	33.0	46.0
	xor2, 2reg	768	25.0	45.8	46.3	49.8
	Illinois	1024	17.0	31.6	29.0	33.7
	xor2 comb	1024	23.5	31.6	34.8	48.0
	xor2, 2reg	1024	25.9	27.9	30.3	54.5

5.2 Pattern Count Results

Tables 3 and 4 show some of the pattern count data results for Designs A and C. The first column shows the number of expansion ratios and configurations used for each experiment. The second column enumerates the configuration or expansion ratio for each reported pattern count. Columns 3, 4, and 5 show the number of ATPG patterns generated for three decompressor schemes. In all cases, the first expansion ratio (or configuration) has the most number of patterns, and each subsequent ATPG run with a new expansion ratio (or configuration) typically generates fewer and fewer patterns. The last run is always the full serial top-off run, and, depending on how many faults are left to detect, the pattern count will vary.

As can be seen in the pattern count results, the ATPG engine has a hard time benefiting from the cheaper first and second expansion ratios (or configuration) when using the Illinois scan decompression scheme. This is primarily due to the constraints it imposes on the ATPG engine, namely the increased free-variable dependencies, when compared to the two other compression schemes that use XOR gates. Likewise, the pattern count results for the 2-input XOR combinational decompressor scheme indicate more constraints on the ATPG tool than the 2-input XOR limited dependence sequential linear decompression scheme, which detects most of its faults with the first two expansion ratios (or configurations) and needs fewer full serial top-off vectors than the other two decompressor types.

0-7695-2812-0/07 $20.00 © 2007 IEEE

Table 3. Design A Pattern Count Results

Decompressor Type		Pattern Count: Illinois	xor2 comb	xor2, 2 reg
Num. of Scan chains		768	768	768
1 Ratio, 1 Config	1st Expan. Ratio	450	973	1220
	Serial Top-off	301	194	124
1 Ratio, 4 Configs	1st Config.	450	973	1220
	2nd Config.	457	205	75
	3rd Config.	215	78	34
	4th Config.	290	25	3
	Serial Top-off	160	166	120
4 Expan. Ratios, 1 Config	1st Expan. Ratio	450	973	1220
	2nd Expan. Ratio	300	218	205
	3rd Expan. Ratio	307	93	8
	4th Expan. Ratio	309	179	153
	Serial Top-off	74	39	4
4 Expan. Ratios, w/Depend. Ana., 1 Config	1st Expan. Ratio	450	973	1220
	2nd Expan. Ratio	586	485	244
	3rd Expan. Ratio	274	104	5
	4th Expan. Ratio	241	10	1
	Serial Top-off	22	1	0

Table 4. Design C Pattern Count Results

Decompressor Type		Pattern Count: Illinois	xor2 comb	xor2, 2 reg
Num. of Scan chains		1024	1024	1024
1 Ratio, 1 Config	1st Expan. Ratio	5029	5312	5139
	Serial Top-off	121	74	62
1 Ratio, 4 Configs	1st Config.	5029	5312	5139
	2nd Config.	450	94	136
	3rd Config.	157	9	18
	4th Config.	39	3	4
	Serial Top-off	41	43	53
4 Expan. Ratios, 1 Config	1st Expan. Ratio	5029	5312	5139
	2nd Expan. Ratio	324	93	111
	3rd Expan. Ratio	132	18	19
	4th Expan. Ratio	24	124	13
	Serial Top-off	43	26	43
4 Expan. Ratios, w/Depend. Ana., 1 Config	1st Expan. Ratio	5029	5312	5139
	2nd Expan. Ratio	248	182	123
	3rd Expan. Ratio	136	73	57
	4th Expan. Ratio	137	17	6
	Serial Top-off	24	8	2

6. Conclusion

The results of this paper demonstrate that by detecting faults with a higher than normal expansion ratio, and then using progressively smaller ones, the amount of compression can be significantly improved when compared with conventional approaches that use a single expansion ratio with one or more configurations. Furthermore, this paper also shows that by exploiting the flexibility of choosing how the scan chains are concatenated by using a dependency analysis procedure that takes into account structural dependencies among the scan chains as well as free-variable dependencies in the logic driving the scan chains, the detection of faults is significantly improved, as is the overall test compression.

Acknowledgements

This material is based on work supported in part by the National Science Foundation under Grant No. CCR-0306238.

References

[Bayraktaroglu 03] I. Bayraktaroglu and A. Orailoglu, "Concurrent Application of Compaction and Compression for Test Time and Data Volume Reduction in Scan Designs," *IEEE Trans. on Computers*, Vol. 52, No. 11, pp. 1480-1489, Nov. 2003.

[Dutta 06] A. Dutta and N.A. Touba, "Using Limited Dependence Sequential Expansion for Decompressing Test Vectors," *Proc. Int. Test Conf,* 2006.

[Hamzaoglu 99] I.Hamzaoglu and J.H.Patel, "Reducing Test Application Time for Full Scan Embedded Cores," *Proc. Int. Symp. on Fault-Tolerant Computing*, pp. 260-267, 1999.

[Könemann 01] B. Koenemann, C. Barnhart, B. Keller, T. Snethen, O. Farnsworth, and D. Wheater, "A SmartBIST Variant with Guaranteed Encoding," *Proc. Asian Test Symp.*, pp. 325-330, 2001.

[Mitra 06] S. Mitra and K.S. Kim, "XPAND: An Efficient Test Stimulus Compression Technique," *IEEE Trans. on Computers*, Vol. 55, No. 2, pp. 163-173, Feb. 2006.

[Pandey 02] A.R. Pandey and J.H. Patel, "Reconfiguration Technique for Reducing Test Time and Test Volume in Illinois Scan Architecture Based Designs," *Proc. VLSI Test Symposium*, pp. 9-15, 2002.

[Rajski 04] J. Rajski, J. Tyszer, M. Kassab, and N. Mukherje, "Embedded Deterministic Test," *IEEE Trans. on Computer-Aided Design*, Vol. 23, No. 5, pp. 776-792, May 2004.

[Samaranayake 03] S. Samaranayake, E. Gizdarski, N. Sitchinava, F. Neuveux, R. Kapur, and T. W. Williams, "A Reconfigurable Shared Scan-In Architecture," *Proc. VLSI Test Symposium*, pp. 9-14, 2003.

[Shah 04] M.A. Shah and J.H. Patel, "Enhancement of the Illinois Scan Architecture for Use with Multiple Scan Inputs," *Proc. of Annual Symp. on VLSI*, pp. 167-172, 2004.

[Sitchinava 04] N. Sitchinava, S. Samaranayake, R. Kapur, E. Gizdarski, F. Neuveux, and T.W. Williams, "Changing the Scan Enable During Shift," *Proc. VLSI Test Symposium*, pp. 73-78, 2004.

[Tang 03] H. Tang, S.M. Reddy, and I. Pomeranz, "On Reducing Test Data Volume and Test Application Time for Multiple Scan Designs," *Proc. Int. Test Conf.*, pp. 1079-1088, 2003.

[Touba 06] N.A. Touba, "Survey of Test Vector Compression Techniques," *IEEE Design & Test*, Vol. 23, Issue 4, pp. 294-303, Jul. 2006.

[Wang 04] L.-T. Wang, X. Wen, H. Furukawa, F.-S. Hsu, S.-H. Lin, S.-W. Tsai, K. S. Abdel-Hafez, and S. Wu, "VirtualScan: A New Compressed Scan Technology for Test Cost Reduction," *Proc. Int. Test Conf.*, pp. 916-925, 2004.