Improving Test Compression with Scan Feedforward Techniques

Sreenivaas S. Muthyala and Nur A. Touba

Computer Engineering Research Center
University of Texas, Austin, TX 78712
sreenivaas@utexas.edu, touba@utexas.edu

Abstract

Scan feedforward techniques are proposed for improving test compression for both sequential linear decompressors based on solving linear equations and for broadcast scan based decompressors in which the decompressor constraints are included in the automatic test pattern generation (ATPG) backtrace. The conventional approach is to first pre-load a sequential linear decompressor with free variables from the tester for some number of clock cycles before beginning to shift the scan chain. A scan feedforward scheme is proposed here which can shorten or eliminate the need for pre-loading the decompressor. More importantly, even if the pre-load phase is not changed, the proposed scheme significantly expands the free-variable dependence of each scan cell, which increases the probability of encoding a test cube. This permits more faults to be targeted by each test cube generated during ATPG (i.e., improved dynamic compaction) thereby leading to fewer test cubes for the same fault coverage. For broadcast scan based schemes, a methodology is presented for using both feedforward and feedback in the scan chains to increase encoding flexibility. The proposed scheme is effective even when narrow TAMs (even 1-bit TAMs) are feeding the decompressor. There is no need to pre-load pipeline flip-flops as is conventionally required in existing solutions thereby reducing test time and improving compression. Experimental results are presented demonstrating the effectiveness of the proposed schemes.

1. Introduction

Test data compression is widely used to compress the amount of data stored on the tester. This helps to reduce tester storage requirements and improve test time, as less data has to be transferred over the limited test data bandwidth between the tester and circuit-under-test (CUT) [Touba 06]. The data brought in from the tester will be referred to here as *free variables,* which can be assigned any value. In test compression, each *test cube* (scan vector in which unassigned inputs are left as don't cares, *X*'s) is encoded by making assignments to free variables such that after decompression, the values loaded into the scan cells match the test cubes in all the care bit locations.

The trend towards increasingly hierarchical designs with cores having their own local decompressor has led to small bandwidth test access mechanisms (TAMs) being used to deliver free variables to the decompressor (in some cases the TAM may be only one bit wide). Having only a narrow TAM feeding a decompressor causes issues for both sequential linear decompressors (e.g., [Rajski 04]) and broadcast scan based decompressors (e.g., [Lee 98], [Hamzaoglu 99], [Samaranayake 03], [Wang 04]) which are the two commonly used approaches in commercial test compression tools. To illustrate this, first consider a sequential linear decompressor. The conventional approach is to "pre-load" the sequential linear decompressor for some number of cycles to fill it with a sufficient number of free-variables to encode the first scan slices [Rajski 04]. During the pre-load phase, the decompressor receives free variables from the TAM with scan shift disabled. If the TAM bringing in free variables is only one bit wide, the number of cycles for pre-loading the decompressor can become long thereby increasing the test time. Moreover, the test cube with the worst-case largest number of specified bits in the early scan slices will dictate the length of the pre-load phase, which is then typically used for decompressing all test cubes. Thus, a long pre-load phase increases the number of free variables used to encode each test cube thereby degrading the encoding efficiency and reducing the amount of compression achieved. For reconfigurable broadcast scan based decompressors, having a narrow TAM means that there are fewer ways that the connections between the TAM lines and the scan chains can be reconfigured thereby reducing the flexibility and effectiveness of the decompressor. If the TAM is only one bit wide, then the connections cannot be reconfigured. In [Chandra 09], a scheme was proposed for addressing this problem by having *n* "pipeline" flip-flops store a window of *n* bits arriving on the TAM and then using these *n* bits to perform reconfigurable fanout. This solution requires *n* clock cycles to effectively "pre-load" the *n* pipeline flip-flops for each test cube, so again this has the drawback of increased test time and reduced test compression.

This paper proposes an approach that eliminates the need for pre-loading a decompressor, and more importantly increases the encoding flexibility, by feeding forward data in the scan chain after the scan chain is loaded halfway. This process will be described in detail in Sec. 2. Feedforward schemes are described for both

978-1-4799-4722-5/14/$31.00

sequential linear decompressors which solve linear equations to encode test cubes as well as for broadcast scan based schemes which incorporate the decompressor constraints in the automatic test pattern generation (ATPG) backtrace.

For sequential linear decompressors, the proposed approach can be used to shorten or eliminate the time required for pre-loading the decompressor. Even if the pre-load phase is not shortened, the proposed approach provides more diverse linear equations with greatly expanded free variable dependence. This increases the probability of encoding a test cube and can permit more faults to be targeted by each test cube generated during ATPG thereby leading to fewer test cubes for the same fault coverage.

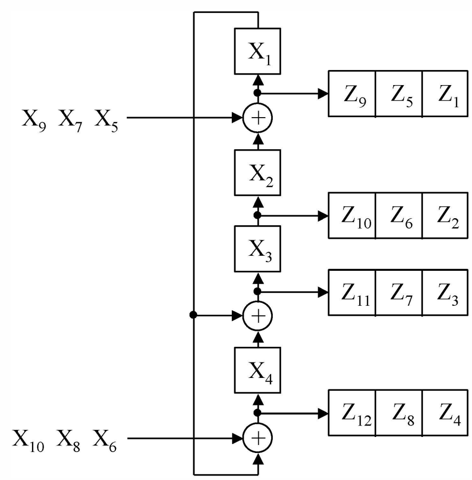
For broadcast scan based schemes, a methodology is described for using both feedforward and feedback in the scan chains to increase encoding flexibility compared to [Dutta 06] and [Chandra 09]. Moreover, the proposed method does not need the extra pipeline flip-flops that [Dutta 06] and [Chandra 09] do, and thereby eliminates the test time and test data required to pre-load those flip-flops thereby increasing test compression.

The paper is organized as follows: Sec. 2 describes the feedfoward scheme for use with sequential linear decompressors and explains how it improves the utilization of free variables. Sec. 3 presents a scheme that uses both feedforward and feedback in the scan chain to keep linear dependence small so that the constraints can be incorporated into the ATPG backtrace. Sec. 4 shows experimental results, and Sec. 5 is a conclusion.

2. Feedforward Scheme for Sequential Linear Decompressors

In sequential linear decompression, the free variables coming from the TAM are fed into a linear finite state machine (LFSM), e.g., a linear feedback shift register (LFSR) or ring generator [Mrugalski 04]. Symbolic simulation of the free variables coming from the TAM is performed to obtain the linear equations corresponding to the value of each bit in the scan cells after decompression in terms of the free variables. This process is illustrated in Fig. 1. These linear equations can be represented as a Boolean matrix as shown in Fig. 2. To encode a particular test cube, the system of linear equations in Fig. 2 needs to be solved for each care bit in the scan cells (i.e., each care bit in the *Z* vector) to obtain the assignment of values to the free variables (i.e., the *X* vector).

The example in Fig. 1 assumed the decompressor was initially loaded with free variables. In reality, the decompressor is reset to all 0's between each test cube to decouple the linear equations for each test cube so that each test cube is encoded with its own set of free variables. After the decompressor is reset, the conventional approach is to have a pre-load phase where the decompressor is loaded with free variables from the TAM for a few clock cycles before scan shifting begins. Consider the case where the TAM width is only one bit. In this case, to preload the decompressor with *n* free variables, the pre-load phase is *n* clock cycles long. The form of the system of linear equations in this case is shown in Fig. 3 where the shaded areas indicate free variable dependence. Note that the columns correspond to free-variables and the rows correspond to scan cells. The linear equations for the scan cells in the first scan slice will depend on the *n* free variables that were pre-loaded plus the free variables arriving on the TAM inputs during the first scan shift cycle. In each subsequent scan shift cycle, the TAM will bring additional free-variables equal to the width of the TAM. The scan cells in each subsequent scan slice will depend on all the free variables that have arrived up to that point in time (which increases by a number equal to the width of the TAM in each shift cycle). The very last scan slice loaded into the scan chains depends on all the free variables (used to encode that particular test cube).



$Z_9 = X_1 \oplus X_4 \oplus X_9$	$Z_5 = X_3 \oplus X_7$	$Z_1 = X_2 \oplus X_5$
$Z_{10} = X_1 \oplus X_2 \oplus X_5 \oplus X_6$	$Z_6 = X_1 \oplus X_4$	$Z_2 = X_3$
$Z_{11} = X_2 \oplus X_3 \oplus X_5 \oplus X_7 \oplus X_8$	$Z_7 = X_1 \oplus X_2 \oplus X_5 \oplus X_6$	$Z_3 = X_1 \oplus X_4$
$Z_{12} = X_3 \oplus X_7 \oplus X_{10}$	$Z_8 = X_2 \oplus X_5 \oplus X_8$	$Z_4 = X_1 \oplus X_6$

Figure 1. Example of Symbolic Simulation of Linear Decompressor

$$\begin{pmatrix} 0&1&0&0&1&0&0&0&0&0 \\ 0&0&1&0&0&0&0&0&0&0 \\ 1&0&0&1&0&0&0&0&0&0 \\ 1&0&0&0&0&1&0&0&0&0 \\ 0&0&1&0&0&0&1&0&0&0 \\ 1&0&0&1&0&0&0&0&0&0 \\ 1&1&0&0&1&1&0&0&0&0 \\ 0&1&0&0&1&0&0&1&0&0 \\ 1&0&0&1&0&0&0&0&1&0 \\ 1&1&0&0&1&1&0&0&0&0 \\ 0&1&1&0&1&0&1&1&0&0 \\ 0&0&1&0&0&0&1&0&0&1 \end{pmatrix} \begin{pmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \\ X_5 \\ X_6 \\ X_7 \\ X_8 \\ X_9 \\ X_{10} \end{pmatrix} = \begin{pmatrix} Z_1 \\ Z_2 \\ Z_3 \\ Z_4 \\ Z_5 \\ Z_6 \\ Z_7 \\ Z_8 \\ Z_9 \\ Z_{10} \\ Z_{11} \\ Z_{12} \end{pmatrix}$$

Figure 2. System of Linear Equations for Fig. 1

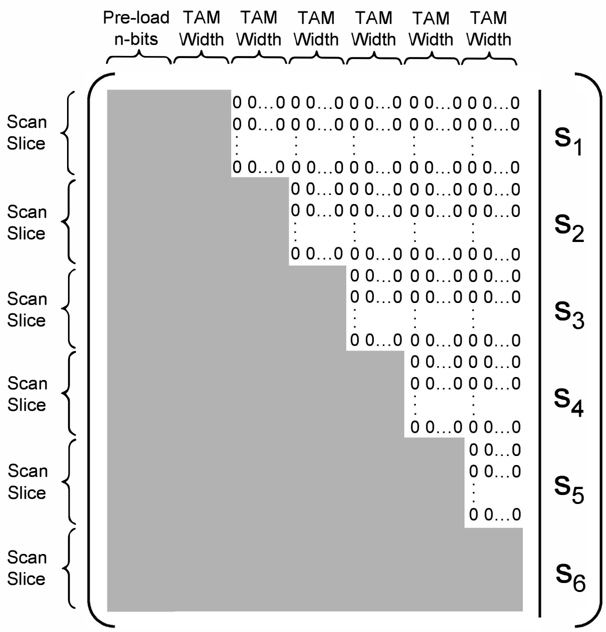


Figure 3. Form of Linear Equations for Conventional Sequential Linear Decompressor

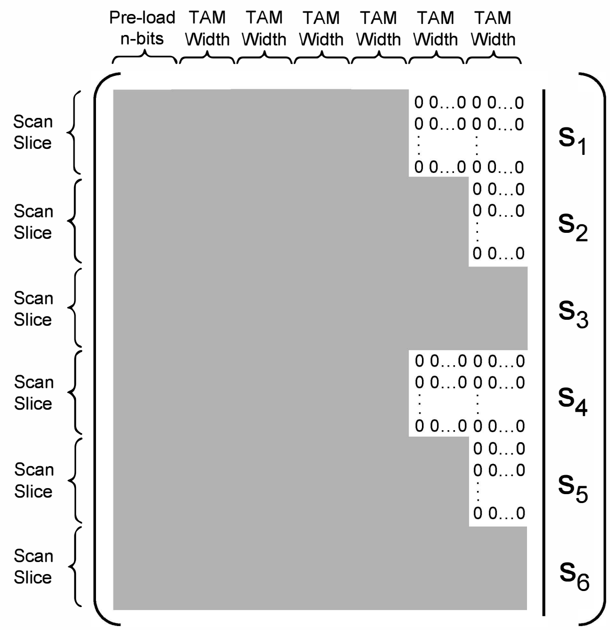


Figure 4. Form of Linear Equations for Sequential Linear Decompressor with Proposed Feedfoward Scheme

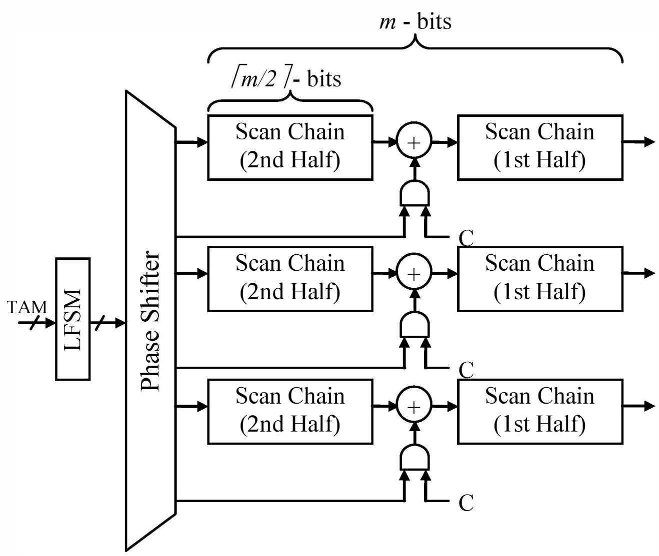


Figure 5. Feedforward Scheme for Sequential Linear Decompressor where each scan chain is *m* bits long. *C=0* for first $\lceil m/2 \rceil$shift cycles after capture cycle, and *C=1* for remaining shift cycles until scan chain fully loaded.

The proposed feedforward technique is illustrated in Fig. 5. An additional output from the phase shifter is fed forward and XORed into the middle of each scan chain. These feedforward inputs are activated only after the output response has shifted past the XOR gate. There is a control signal, *C*, that is ANDed with all feedforward inputs and holds them at 0 until the output response can shift past the XOR gates without being modified. If each scan chain is *m*-bit long, then after the capture cycle the control signal, *C*, deactivated for the first $\lceil m/2 \rceil$ clock cycles after which it is activated for the remaining clock cycles until the scan chain is fully loaded. The phase shifter should be designed to ensure appropriate channel separation between all outputs of the phase shifter [Rajski 00] including those used as feedforward inputs.

The advantage of using the feedforward technique shown in Fig. 5 is that it significantly increases the free variable dependence of the early scan slices. Consider the first scan slice loaded in the scan chains. Normally it only depends on the number of free variables that are pre-loaded into the decompressor plus the first set of free variables on the TAM. With the feedforward technique shown in Fig. 5, after $\lceil m/2 \rceil$ clock cycles, the first scan slice is getting XORed with additional free variables being fed forward from the decompressor. At this point in time, the decompressor has received many more additional free variables from the TAM. So the linear equations for the first scan slice after the feedforward input is XORed in will depend on not only the pre-loaded free variables, but also on all the free variables received by the decompressor over the first $\lceil m/2 \rceil$ scan shift cycles plus the current shift cycle. The form of the system of linear equations with the proposed feedforward scheme is shown in Fig. 4. In comparing Fig. 4 with Fig. 3, it can be seen that the linear equations with the feedforward scheme are more diverse with increased free variable dependence providing more degrees of freedom for solving them. If there is a cluster of care bits in the early scan slices, there may not be enough free variables in the system of linear equations in Fig. 3 to solve for it, but with the more diverse and considerably expanded free variable dependence in Fig. 5, it is much easier to solve.

Consider the case where the linear decompressor is not pre-loaded at all before beginning to load the scan chain. With the proposed feedforward technique, the linear equations for the early scan slices will still depend on all the free variables received by the decompressor from the TAM over the first $\lceil m/2 \rceil$ clock cycles plus the current shift cycle. These equations will still have much greater free variable dependence than the conventional approach of pre-loading the decompressor, which will typically be much shorter than $\lceil m/2 \rceil$ clock cycles. Consequently, with the proposed feedforward scheme, it is possible to shorten or omit the pre-load phase altogether thereby saving test time and improving test compression.

Even if the pre-load phase is not shortened, the increased diversity and expanded free variable dependence in the linear equations for the feedforward scheme increases the probability of encoding a test cube. Typically for sequential linear decompressors, ATPG will continue to make input assignments to target additional faults (i.e., perform dynamic compaction) as long as the test cube remains encodable (i.e., the system of linear equations can be solved). With the proposed feedforward scheme, there is more flexibility to make additional input assignments while still being able to solve the linear equations thereby improving the amount of dynamic compaction that can be performed during ATPG. This results in fewer test cubes being generated to achieve the same fault coverage. Fewer test cubes means less test data stored on the tester and less test time to apply the tests.

3. Feedforward Scheme for Including Decompressor Constraints in Backtrace

The other major class of test compression schemes used in commercial tools are based on having simple decompressor constraints that can be incorporated into the ATPG backtrace. This focuses the ATPGs search so that it only considers encodable input assignments when targeting faults and can allow more aggressive dynamic compaction without the risk of the test cube becoming unencodable. One way to keep the decompressor constraints simple is to use broadcast scan in which the same value is loaded into multiple scan chains. In this case, the decompressor constraints can be incorporated into the ATPG by simply tying dependent inputs together in the circuit description given to the ATPG. The drawback of broadcast scan is that the encoding flexibility is limited because some scan cells can never take on opposite values, which can make it impossible to detect some faults. The faults that cannot be detected through the decompressor have to be tested in "serial mode" in which the decompressor is bypassed and all the scan cells are tied to form one long scan chain [Hamzaoglu 99]. These uncompressed serial vectors degrade the amount of compression that can be achieved. One approach to reduce the number of faults for which serial vectors are needed is to provide the ability to reconfigure the decompressor to change the constraints. This can be done statically (on a per scan basis) by either reconfiguring the scan chains [Pandey 02] or reconfiguring the fanout network [Samaranayake 03], [Tang 03], [Mitra 06], or using Multimode Illinois Scan [Chandra 07]. Or, it can be done dynamically (on a per shift basis) [Sitchinava 04] where MUXes are placed in front of each scan chain and the control signals for the MUXes are driven by tester channels. One issue for reconfiguration based approaches is that for small width TAMs, the number of different ways that the constraints can be reconfigured becomes limited, and for a one-bit TAM, there is only one broadcast configuration which is to load all scan chains from the single TAM input. The scheme proposed in [Chandra 09] addresses this issue by having n pipeline flip-flops store a window of n bits arriving on the TAM and then using these n bits to perform reconfigurable fanout. These pipeline flip-flops have to be pre-loaded before scan shifting similar to sequential linear decompressors as discussed earlier.

An alternative to reconfiguration is to have limited dependence XOR constraints as proposed in [Wang 04] and [Dutta 06] where the value of each scan cell depends on the XOR of a small number of free variables (e.g., 2 or 3). This provides more flexibility for encoding test cubes compared with fanout based constraints, but is simple enough that it can be incorporated into the ATPG backtrace and still allow efficient ATPG. Unlike fanout based constraints where certain scan cells must always have the same value as other scan cells, with XOR constraints, there is more than one way to justify a particular value on a scan cell. The impact of the improved encoding flexibility provided by XOR gates on the overall probability of encoding a test cube was quantitatively analyzed in [Dutta 06] and shown to be substantial. In [Dutta 06], the number of unique combinations of limited dependence XORs was increased by using pipeline flip-flops, which again requires a pre-load phase.

The proposed scheme using both feedforward and feedback in the scan chain to create limited dependence XOR constraints without the need for pre-loading is illustrated in Fig. 6. There is a control signal, C, which when set to 0 disables the feedforward and feedback allowing normal scan shifting. After the capture cycle, C is held at 0 until the output response shifts past the feedback tap. In the example in Fig. 6, there is a two-bit TAM with its channels labeled X and Y which are used to load 6 scan chains (3 loaded from X and 3 loaded from Y). Each scan chain is 8 bits long. The feedback tap comes after the 5th scan element in each scan chain. So after the capture cycle, C is held to 0 for 5 shift cycles thereby disabling the feedback and feedforward lines so that the scan chain shifts normally. This allows the output response to be shifted past the feedback line. The free variable dependence of each scan cell after the first 5 shift cycles are indicated in Fig. 6. The free variables coming in from the X and Y channels are labeled as X_1-X_8 and Y_1-Y_8, respectively. For the last three shift cycles (shift cycles 6 through 8), C is set to 1 which activates the feedforward and feedback lines. The free variable dependence of each scan cell when the scan chain is fully loaded is shown in Fig. 7. As can be seen, no scan cell depends on the XOR of more than 3 free variables and no two scan cells have identical XOR dependence. No pipeline flip-flops are used. Even those flip-flops which are used to store the window of data required to drive the feedforward inputs are part of the scan chains and hence no pre-load phase is required. The number of cycles used to load the scan chains is exactly equal to the length of the scan chain. This reduces the test time and amount of test data required to load the scan chains in comparison to [Dutta 06] and [Chandra 09]. Similar to what is done in [Dutta 06], the XOR constraints for encoding a test cube imposed by the proposed scheme can be incorporated into the circuit description during ATPG so that it is included in the backtrace when generating test cubes.

Given any TAM width (including 1 bit) and any number of scan chains, the proposed scheme can be implemented. An algorithm for selecting the feedback and feedforward locations and their inputs is described as follows.

First some definitions. The *expansion ratio (ER)* is defined as the number of scan chains driven by each TAM channel (i.e., *ER = Chains/TAM_Width*). One of the *ER* scan chains fed by the TAM channel is the primary scan chain for the TAM channel and is fed directly by the TAM without additional inputs XORed at the input of the scan chain (refer to Fig. 8), and the others are secondary scan chains for the TAM channel, which have inputs from primary scan chain of other channels XORed at the input of the scan chain. These additional feedforward inputs are controlled by the control signal *C*. Thus, a primary scan chain is the reference scan chain, which has unshifted scan data from the TAM channel. The outputs of the various scan cells of the primary scan chains are tapped and fed to other scan chains. The distribution of scan data from the primary scan chains of various TAM channels to the other scan chains is done in such a way that at the end of the scan shift, no two scan cells depend on the same set of free variables. The form of the primary and secondary scan chains is shown in Fig. 8 where the scan length is *m* bits. The scan chains are divided into 3 segments where the first scan segment contains $\lfloor [m-(ER/TAM\_Width)]/2 \rfloor$ scan cells, the second scan segment contains $\lceil [m-(ER/TAM\_Width))/2 \rceil$ scan cells, and the third scan segment contains $\lceil ER/TAM\_Width \rceil$ scan cells. This type of segmentation ensures maximum compression while ensuring sufficient flexibility to ensure no two scan cells have XOR dependance on the same set of free variables. Note that the third segment contains the "pipeline" flip-flops, which are also a part of the scan chain. Hence extra flip-flops are not required to store the required window of scan data, which are then used to drive the feedforward taps of various scan chains. Each primary scan chain for the *i*-th TAM channel (TAM_i) has one feedforward input labeled as $IN_{i,1,1}$ in Fig. 8. Each *j*-th secondary scan chain for TAM_i has three feedforward inputs labeled as $IN_{i,j,1}$, $IN_{i,j,2}$, and $IN_{i,j,3}$. Each feedforward input is driven from the output of a scan cell in a primary scan chain. The outputs of the scan cells in the primary scan chain will be denoted as $OUT_{x,y,z}$ where *x* is the TAM channel that the primary scan chain is associated with, *y* is the scan segment number (1, 2, or 3), and *z* is the scan cell position in the scan segment numbered starting from 1 (where position 1 is the rightmost scan cell in the scan segment). The pseudo-code for connecting the feedforward inputs in each scan chain is as follows:

```
for TAM = 1 to TAM_Width {
  t = TAM; p = 1;
  for chain = 1 to ER {
    if (TAM_Width > 1) {
      t = (t % TAM_Width) + 1;
      if (t == TAM)  {
        p++;
        t = (t % TAM_Width) + 1;
      }
    }
    else p++;
    IN_TAM,chain,1 = OUT_t,2,p;
  }
  t = TAM; p = 1;
  for chain = 2 to ER {
    IN_TAM,chain,2 = OUT_t,3,p;
    IN_TAM,chain,3 = OUT_t,3,p;
    t = (t % TAM_Width) + 1;
    if (t == TAM)  p++;
  }
}
```

The connections described by the pseudo-code above ensure that no two scan cells depend on an identical set of free variables, and no scan cell depends on the XOR of more than 3 free variables. Moreover, only one control signal *C* is required to activate the feedback and feedforward taps, which is very small control overhead for the advantages obtained using the proposed method.

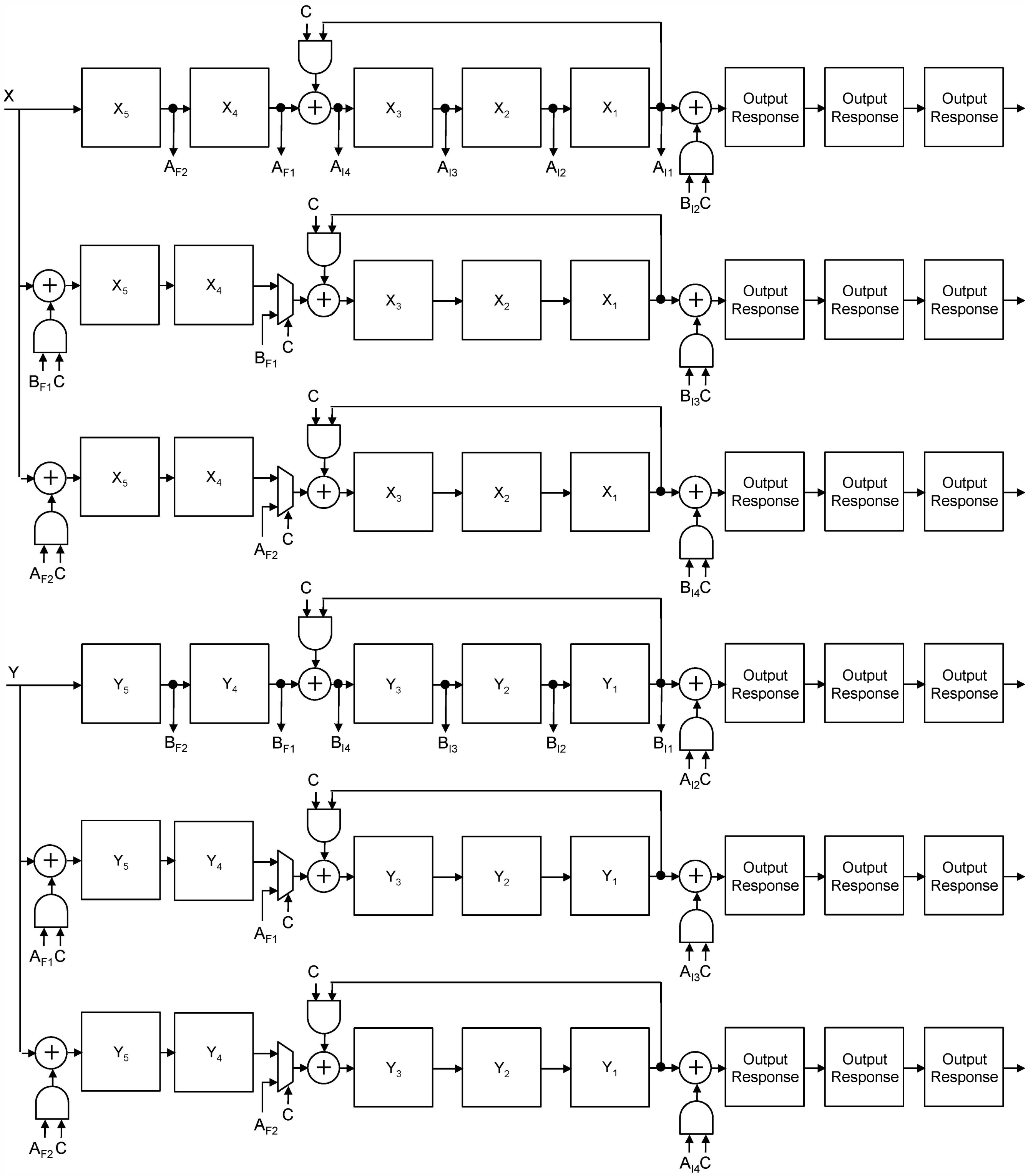


Figure 6. Example of proposed scheme with 2-bit TAM driving 6 scan chains. C=0 for first 5 shift cycles. Contents of each scan cell shown after 5 shift cycles.

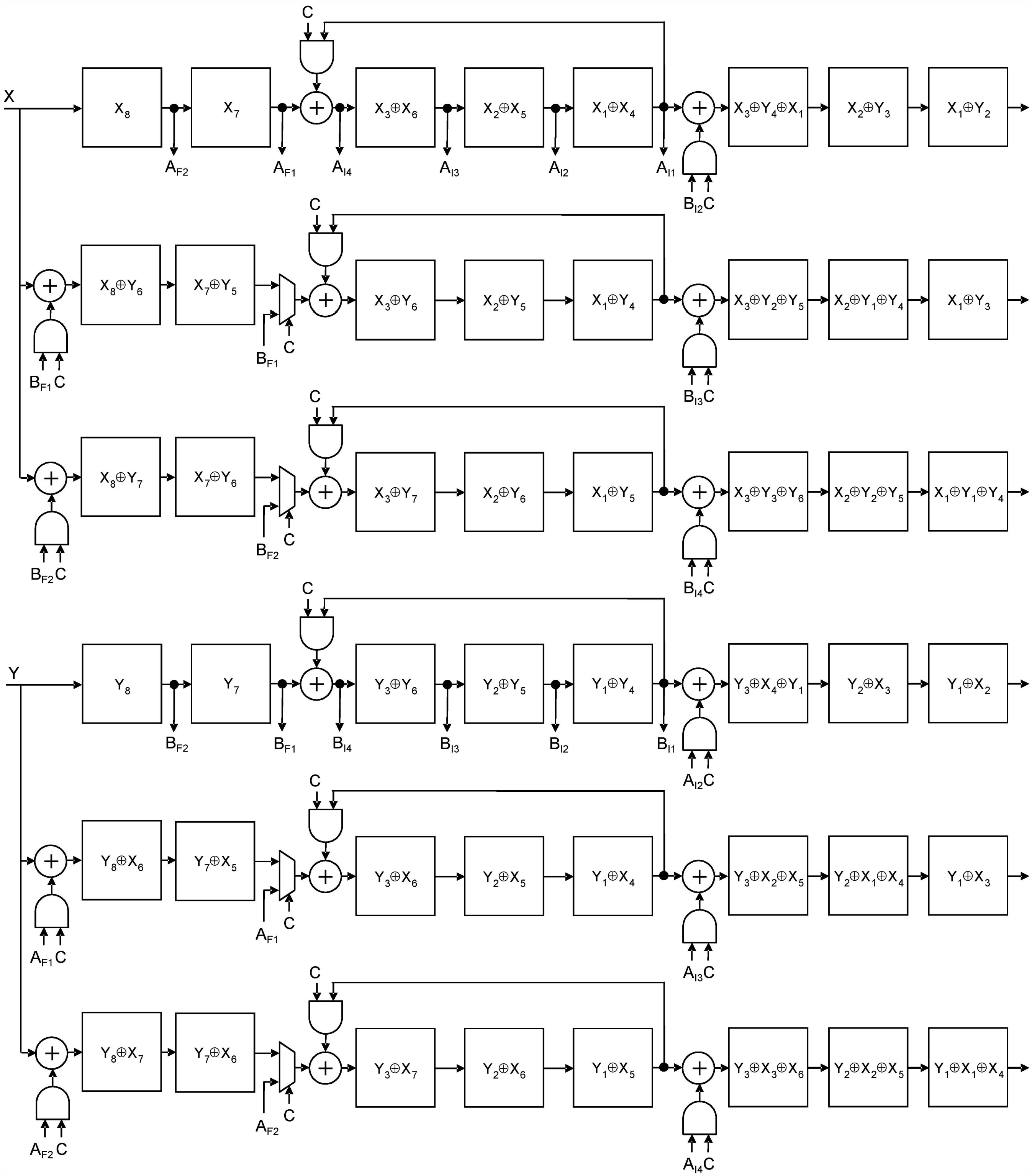


Figure 7. Example from Fig. 6 after scan chains are fully loaded.
C=0 for first 5 shift cycles, and C=1 for last 3 shift cycles.

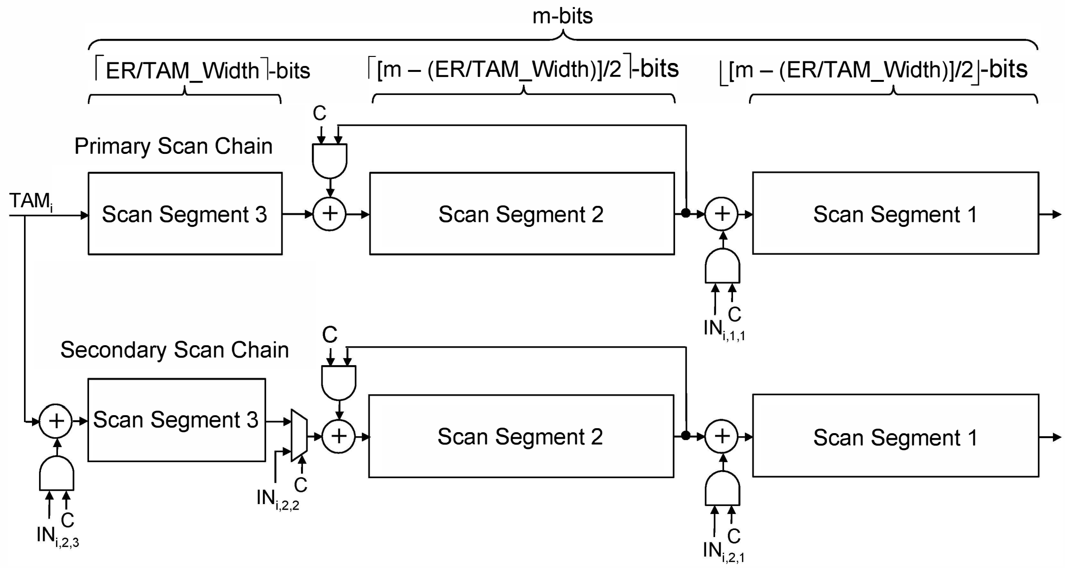


Figure 8. Form of Primary and Secondary Scan Chains and Nomenclature of Feedfoward and Feedback Taps

4. Experimental Results

To evaluate the impact of the proposed feedforward scheme for sequential linear decompressors (described in Sec. 2), experiments were performed to measure the probability of encoding test cubes. This was done by generating a statistically significant number of random test cubes with different numbers of specified bits and seeing how often they can be encoded with a sequential linear decompressor with the conventional scheme and with the feedforward scheme. Figs. 9 and 10 show graphs plotting the probability of encoding a test cube versus the number of specified bits in the test cube. In Fig. 9, results are shown for a scan architecture with 100 scan chains each 100 bits long, and the decompressor is an LFSR driven by a 4-bit TAM. In Fig. 10, results are shown for a scan architecture with 50 scan chains each 50 bits long, and the decompressor is an LFSR driven by a 2-bit TAM. In both cases, 8 pre-load cycles were used before the scan shifting begins. While the proposed feedforward scheme could be effective without any pre-loading, the conventional approach would not be effective without pre-loading because there would not be enough free-variables to encode the early scan slices. So to have a fair comparison, 8 pre-load cycles are used for both the conventional method and the feedforward scheme. As can be seen from the results, the feedfoward scheme is able to effectively encode test cubes with more specified bits than the conventional approach, which means that more dynamic compaction can be performed during ATPG resulting in fewer test vectors, or alternatively a more aggressive expansion ratio can be used to achieve greater compression.

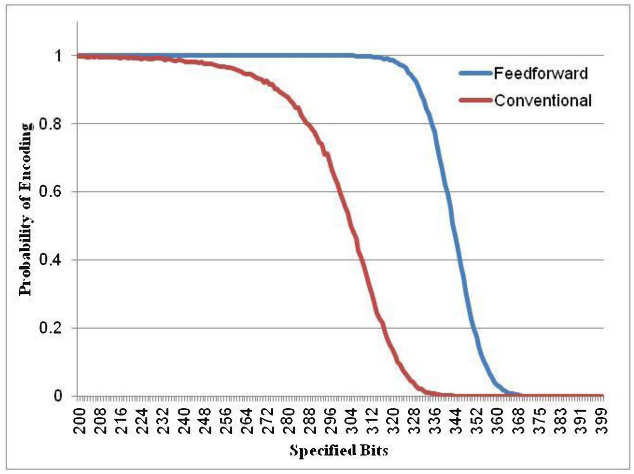


Figure 9. Probability of encoding test cubes using an LFSR decompressor with a 4-bit TAM driving 100 scan chains with a scan length of 100 with 8 pre-load cycles

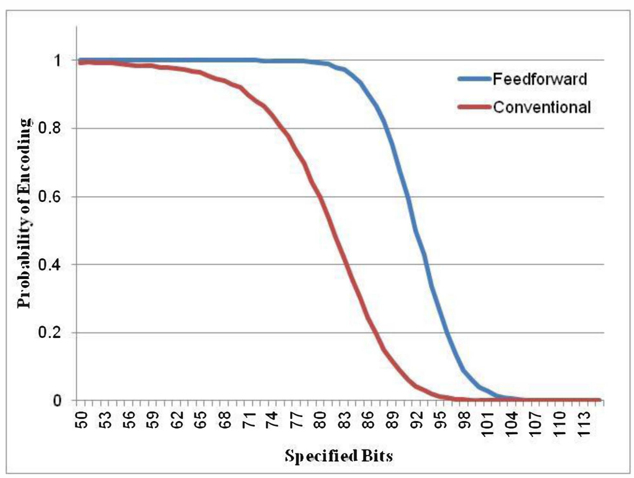


Figure 10. Probability of encoding test cubes using an LFSR decompressor with a 2-bit TAM driving 50 scan chains with a scan length of 50 with 8 pre-load cycles

Table 1. Results for Decompressor that Incorporates Constraints in ATPG Backtrace

Circuit	Scan Cells	TAM Width	Expansion Ratio	Illinois Scan			Proposed Feedforward/Feedback Scheme		
				Parallel Vectors	Serial Vectors	Tester Data	Parallel Vectors	Serial Vectors	Tester Data
Ckt-A	3065	1	8	242	23	163,423	239	2	97,906
			16	267	72	271,944	272	4	64,484
		4	8	223	41	211,297	236	3	99,819
			16	256	76	282,092	365	4	82,340
		8	8	236	32	188,704	235	3	99,435
			16	288	64	251,456	293	4	68,516
Ckt-B	1960	1	8	192	49	143,080	220	1	55,860
			16	236	91	207,388	287	3	41,181
		4	8	197	54	154,696	219	2	58,232
			16	243	74	175,172	286	3	41,344
		8	8	216	38	128,048	205	2	54,760
			16	258	72	174,144	276	4	43,168

The next set of experiments was performed to evaluate the effectiveness of the proposed feedforward/feedback scheme for incorporating the decompressor constraints in the ATPG (described in Sec. 3). Table 1 shows results comparing implementing compression using Illinois Scan [Hamzaoglu 99] versus using the proposed scheme. For each circuit in Table 1, the number of scan cells is shown. Results were generated when the TAM width is 1, 4, and 8 bits, and in each case, results are shown for an expansion ratio of 8 and 16. For Illinois scan, each TAM channel was fanned out to a number of scan chains equal to the expansion ratio. For the proposed scheme, the feedforward and feedback connections were made using the algorithm given in Sec. 3. For both Illinois scan and the proposed scheme, the decompressor constraints were incorporated into the circuit description used for ATPG. ATPG was performed to generate a set of vectors that can be applied through the decompressor, which are referred to as "parallel" vectors in Table 1. Serial vectors were generated to detect the remaining faults that could not be detected with the decompressor constraints. In Table 1, the number of parallel and serial vectors and the corresponding total amount of tester data is shown for both Illinois Scan and the proposed scheme. As can be seen, the proposed scheme provides greater encoding flexibility, which allows more faults to be detected with parallel vectors even with a TAM width of 1 and without the need for any pre-loading.

5. Conclusions

The proposed feedfoward schemes can be implemented by inserting only a small amount of logic in the scan chains. For sequential linear decompressors, it was shown that the probability of encoding a test cube can be significantly improved allowing more dynamic compaction during ATPG to reduce the number of test cubes, or alternatively, to obtain a more aggressive expansion ratio. It can also be used to reduce or eliminate the need for pre-loading the decompressor, which helps reduce test time. For decompressors in which the constraints are incorporated into the ATPG backtrace, a method was described for increasing encoding flexibility. It was shown to be effective even for one-bit wide TAMs without the need for pre-loading pipeline flip-flops as is required in previous schemes (e.g., [Dutta 06] and [Chandra 09]). The proposed approach can be applied in hierarchical designs where cores may have local decompressors fed by narrow TAMs.

Acknowledgement

This research was supported in part by the National Science Foundation under Grant No. CCF-1217750.

References

[Chandra 07] A. Chandra, H. Yan and R. Kapur, "Multimode Illinois Scan Architecture for Test Application Time and Test Data Volume Reduction," *Proc. of VLSI Test Symposium*, pp. 84-92, 2007.

[Chandra 09] A. Chandra, R. Kapur, and Y. Kanzawa, "Scalable Adaptive Scan (SAS)," *Proc. of Design Automation and Test in Europe (DATE) Conference*, pp. 1476-1481, 2009.

[Dutta 06] A. Dutta and N.A. Touba, "Using Limited Dependence Sequential Expansion for Decompressing Test Vectors," *Proc. of International Test Conference*, Paper 23.1, 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.

[Lee 98] K.-J. Lee, J.J. Chen, and C.H. Huang, "Using a Single Input to Support Multiple Scan Chains," *Proc. Int. Conf. on Computer-Aided Design*, pp. 74-78, 1998.

[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.

[Mrugalski 04] G. Mruglaski, J. Rajski, and J. Tyszer, "Ring Generators – New Devices for Embedded Test Applications," *IEEE Trans. on Computer-Aided Design*, Vol. 23, No. 9, pp. 1306-1320, Sept. 2004.

[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 00] J. Rajski, N. Tamarapalli, and J. Tyszer, "Automated Synthesis of Phase Shifters for Built-In Self-Test Applications," *IEEE Trans. on Computer-Aided Design*, Vol. 19, No. 10, pp. 1175-1188, Oct. 2000.

[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.

[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 Magazine,* 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.