An Industrial Case Study for X-Canceling MISR

Joon-Sung Yang, Nur A. Touba

Computer Engineering Research Center
University of Texas, Austin, TX 78712
{jsyang,touba}@ece.utexas.edu

Shih-Yu Yang, T.M. Mak

Intel Corporation
Portland, OR 97124
{shih-yu.yang,t.m.mak}@intel.com

Abstract

An X-tolerant multiple-input signature register (MISR) compaction methodology that compacts output streams containing unknown (X) values was described in [Touba 07]. Unlike conventional approaches, it does not use X-masking logic at the input of the MISR. Instead it uses symbolic simulation to express each bit of the MISR signature as a linear equation in terms of the X's. Linearly dependent combinations of the signature bits are identified with Gaussian elimination and XORed together to cancel out all X values and yield deterministic values. This new X-canceling approach was applied to some industrial designs under the constraints imposed by an industrial test environment. Practical issues for implementing X-canceling are discussed, and a new architecture for implementing X-canceling based on using a shadow register with multiple selective XORs is presented. Experimental results are shown for industrial designs comparing the performance of X-canceling with X-compact.

1. Introduction

Unknown *'X'* values cause issues in compacting output streams for test compression and BIST. Uninitialized memory elements, bus contention, floating tri-states, and other sources introduce unknown values. *X* values corrupt the final signature making it unknown. A number of schemes have been developed to deal with the problem of *X*'s in the output response.

One way of handling *X*'s is to modify the circuit-under-test (CUT) so that it does not generate *X* values. This approach is called *X-bounding* and requires adding design-for-testability (DFT) logic to prevent *X* value propagation to scan cells [Wang 06]. Another approach, which does not require modifying the CUT, is *X-masking* which masks out *X*'s at the input to the compactor. Mask control data is used to specify which scan chain outputs should be masked during which clock cycles. Many schemes for *X*-masking hardware design and mask control data compression have been developed [Barnhart 01], [Wohl 01, 03, 04], [Pomeranz 02], [Chickermane 04], [Volkerink 05], [Chao 05], [Tang 06], [Rajski 06a]. A third approach is to design an *X-tolerant* compactor which can compact an output stream that contains *X*'s without the need for *X*-masking. *X*-tolerant compactors have been developed based on linear combinational compactors [Mitra 04a], [Patel 03], [Sharma 05], convolutional compactors [Rajski 05], and circular registers [Rajski 06b]. While multiple-input signature registers (MISRs) are the most efficient for compacting output streams without *X*'s, they present difficulties when *X*'s are present because the *X*'s quickly spread and corrupt the signature bits [Mitra 04b].

In [Touba 07], the concept of canceling out *X*'s from MISR signatures was proposed. An *X*-canceling MISR methodology was described which can achieve arbitrarily high error coverage very efficiently where error coverage is the percentage of scan cells that are observed in the presence of *X*'s. Symbolic simulation is used to express each bit of the MISR signature as a linear equation in terms of the *X*'s. Linearly dependent combinations of MISR signature bits are identified with Gaussian elimination and are XORed together to cancel out all *X* values thereby yielding deterministic values that are invariant of what the final values of the *X*'s end up being during the test.

In this paper, a case study using an *X*-canceling MISR for two industrial designs is presented. The contributions of this paper include the following:

- A discussion of the practical issues in implementing an *X*-canceling method for industrial designs.
- A new architecture for implementing an *X*-canceling MISR using a shadow register with multiple selective XORs.
- Experimental results based on industrial test cases to compare the performance of *X*-canceling with *X*-compact (1*X* tolerance), and a comparison of the actual results obtained with the theoretical equations given in [Touba 07].

This paper is organized as follows: Sec. 2 gives a description of a concept of *X*-canceling and explains the symbolic simulation process to identify *X*-canceled combinations. Sec. 3 discusses some of the issues that arose in the case study. In Sec. 4, the proposed *X*-canceling MISR architectures are described and analyzed. Sec. 5 presents the industrial design details. The

978-1-4244-4867-8/09/$25.00 ©2009 IEEE

evaluation and the comparison with other techniques are shown in Sec. 6. Sec. 7 is a conclusion.

2. Overview of X-Canceling MISR

This section gives a brief overview of the operation of an *X*-canceling MISR. A more detailed explanation can be found in [Touba 07].

Assume the output response has been captured in the scan chains after applying a test vector. The value in each scan cell is represented with a symbol. An example is shown in Fig. 1. Once the output response has been shifted in to the MISR, the final MISR signature can be expressed in terms of the symbols through symbolic simulation. Each MISR bit is represented by a linear equation of the scan cell symbols. Fig. 1 illustrates this symbolic representation. The final value of the top bit of the MISR is $X_1 \oplus O_3 \oplus O_8 \oplus O_{13}$, where X_i denotes an *X* value and O_i indicates a non-*X* value.

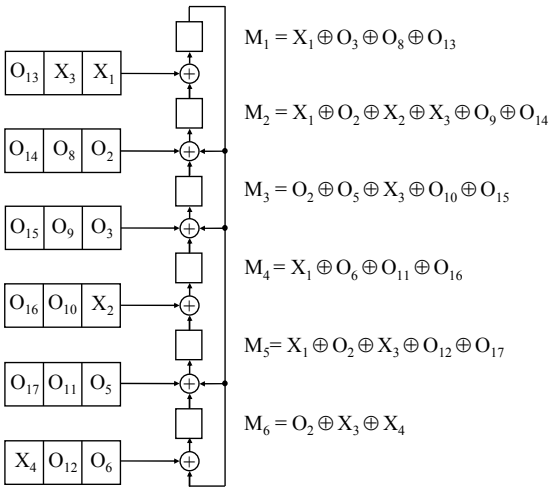

Figure 1. Example of Symbolic Simulation of MISR

The focus here is on the unknown values, so each MISR bit equation can be reduced to a linear combination of the *X* values by assigning *0* to each non-*X* values without loss of generality. These linear combinations can be expressed in the form of a matrix as shown in Fig. 2. Each entry in the matrix has a *1* if the MISR bit corresponding to the row depends of the *X* corresponding to the column.

$$\begin{array}{l} M_1 = X_1 \\ M_2 = X_1 \oplus X_2 \oplus X_3 \\ M_3 = X_3 \\ M_4 = X_1 \\ M_5 = X_1 \oplus X_3 \\ M_6 = X_3 \oplus X_4 \end{array} \rightarrow \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

Figure 2. Linear Equations for MISR in Fig. 1

If the number of columns is less than the number of rows, i.e., the number of *X*'s is less than the MISR size, then some row combinations will be linearly dependent. Gauss-Jordan elimination [Cullen 97] can be performed on the matrix in Fig. 2 to identify the linearly dependent combinations of rows as illustrated in Fig. 3. The last two rows in Fig. 3 have all *0*s and this indicates combinations of MISR bits in which all the *X*'s cancel out. The first all-*0* row corresponds to $M_1 \oplus M_3 \oplus M_5$. This implies that XORing MISR bits M_1, M_3, and M_5 generates an "*X*-canceled" signature bit which depends only on scan cells that captured non-*X* values as shown below:

$$M_1 \oplus M_3 \oplus M_5 = O_3 \oplus O_5 \oplus O_8 \oplus O_{10} \oplus O_{12} \oplus O_{13} \oplus O_{15} \oplus O_{17}$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \begin{matrix} M_1 \\ M_2 \\ M_3 \\ M_4 \\ M_5 \\ M_6 \end{matrix} \xrightarrow{\text{Gaussian Elimination}} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{matrix} M_1 \\ M_1 \oplus M_2 \oplus M_3 \\ M_3 \\ M_3 \oplus M_6 \\ M_1 \oplus M_3 \oplus M_5 \\ M_1 \oplus M_4 \end{matrix}$$

Figure 3. Gauss-Jordan Elimination of MISR Equations

The values of these *X*-canceled MISR bit combinations are deterministic and can be predicted through simulation. Therefore, during test, they can be compared with their fault-free values in order to detect errors.

The MISR is operated across many clock cycles and may span multiple test vectors until the MISR fills up with *X*'s. The MISR signature is then processed by selectively XORing linearly dependent combinations of MISR bits in terms of the *X*'s to generate *X*-free output response to send to the tester. The error coverage can be made arbitrarily high by generating and checking a sufficient number of *X*-canceled output responses. The probability of not detecting an error drops by a factor of 2 for each *X*-canceled combination that is checked. Note that the error coverage does not depend on the actual distribution of the *X*'s in the output response, i.e., it doesn't matter how many *X*'s there are in any particular scan slice.

3. Issues for Case Study

This case study involved investigating the application of an *X*-canceling MISR to two industrial designs. When using *X*-compact [Mitra 04a] for these two designs, the fault coverage dropped significantly from the case where the output response was not compressed. While *X*-compact is guaranteed to be able to tolerate one *X* per scan slice, the distribution of *X*'s in these designs was such that many scan slices had too many *X*'s to be efficiently compacted with *X*-compact. One way to

improve the fault coverage would be to partition the outputs to multiple smaller *X*-compact networks, however, that would result in less compaction and hence increase the number of tester channels needed for output response as well as the amount of test data. The idea of this study was to see whether an *X*-canceling MISR could provide better results since its error coverage does not depend on the distribution of *X*'s in each scan slice.

The *X*-canceling MISR architecture described in [Touba 07] requires only a single tester channel for the output response thus freeing up the remaining tester channels for providing input stimulus. Details of this architecture are given in Sec. 4.1. This architecture is very good for multi-site testing and other applications where it is desirable to have more tester channels for input stimulus and fewer channels for output response. However, in the application considered in this case study, there were some issues for using this architecture:

1. It was preferred to have more output response channels to aid in debug/diagnosis.
2. The implementation in Sec. 4.1 requires a scan architecture that is able to pause the scan load/unload operation during the processing of the MISR signature. This requires the ability to retain the values in the scan cells which requires some form of clock gating.
3. Since the cycle count of each load/unload procedure is different, it might be difficult to validate/debug patterns.

To address these issues, a new architecture for efficiently implementing an *X*-canceling MISR was developed which is based on having multiple selective XORs operating in parallel at the output. Details of this architecture are given in Sec. 4.2. It separates the control of the scan load/unload operation from the MISR signature processing operation which resolves the issues listed above. In this case study, experiments were performed for both architectures to see how the results compared.

4. *X*-Canceling MISR Architectures

The two X-canceling MISR architectures that were investigated in this case study are described in this section.

4.1 *X*-Canceling with Time Multiplexing

Fig. 4 shows the architecture for *X*-canceling with time multiplexing. The key idea is that two phases are alternated over time: a test vector application phase and a signature processing phase. During the test vector application phase, *m* tester channels are used to load the scan vectors through a decompressor. After the capture cycle, the output response is shifted into an *m*-bit MISR through a phase-shifter as the next test vector is loaded. This proceeds across multiple clock cycles and even multiple scan vectors until the MISR fills up with *X*'s. At that point, the scan shifting is stopped, and the signature processing phase begins. Linearly dependent combinations of MISR bits are computed via symbolic simulation as described in Sec. 2. The *X*-canceled combinations are generated using a selective XOR network. In the signature processing phase, the *m* tester channels are used to drive the control inputs to the selective XOR. The *m* tester channels are used to generate the *X*-canceled combinations by selecting which of the *m*-bits in the MISR should be XORed together. Once the MISR signature has been processed (i.e., a sufficient number of *X*-canceled combinations have been generated), then the MISR is reset and the test vector application phase resumes. Note that the *m* tester channels are fully utilized at all times to drive the scan vector decompressor during the test application phase and to drive the selective XOR during the signature processing phase.

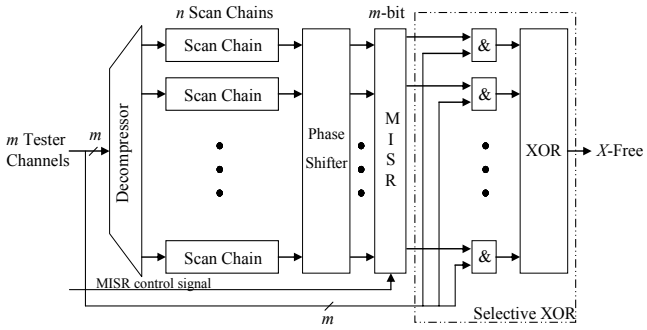

Figure 4. *X*-Canceling with Time Multiplexing

Table 1. Error Coverage versus Number of *X*-Canceled Combinations (*q*)

X-Canceled Combinations (*q*)	Error Coverage
1	50%
2	75%
3	87.5%
4	93.75%
5	96.88%
6	98.44%
7	99.2%
8	99.6%
9	99.8%
10	99.9%

The error coverage that is provided depends on the number of *X*-canceled combinations that are checked. Since the MISR with a primitive polynomial has a pseudo-random property, each *X*-canceled combination will depend on roughly half of the scan cells capturing

non-*X* values. Therefore, if *q X*-canceled combinations are checked, the error coverage will be theoretically equal to $1-2^{-q}$. If an *m*-bit MISR is used, it can store up to *m-q X*'s and obtain a $1-2^{-q}$ error coverage by checking *q* linearly dependent combinations of MISR signature bits obtained via Gauss-Jordan elimination. For example, if *7 X*-canceled combinations are checked, the error coverage is equal to $1-2^{-7}$ = 99.2%. Table 1 shows the theoretical error coverage with *q X*-canceled combinations.

Additional test time is required to stop the test vector application phase and perform the signature processing phase. The number signature processing phases that are required depends on the *X* density (percentage of output response bits that are *X*'s), MISR size, and target error coverage. The number of signature processing phases can be predicted. Assume that the *X* density is *x%*, there are *n* scan chains, and *q X*-canceled combinations are checked to get $1-2^{-q}$ target error coverage. Based on the given information, the theoretical test time can be calculated. In one scan slice, assuming a Gaussian *X* distribution, there would be *n*x X'*s. The MISR can tolerate up to *m-q X'*s to achieve the target test coverage. It takes *(m-q)/(n*x)* cycles to fill up the MISR with *m-q X*'s. Hence, the signature needs to be processed at every *(m-q)/(n*x)* cycles. In the signature processing phase, *q* cycles are needed to provide the control data for generating the *q X*-canceled combinations. Therefore, if the total number of cycles needed to apply the test patterns without stopping scan shifting is *c*, then the number of additional cycles added for canceling out the *X*'s is *[c / (m-q)/(n*x)]*q*. Hence, the total test time and normalized test time with respect to the test time with no compaction is equal to:

$$Total\ Test\ Time = c + [(c*n*x*q) / (m-q)] \quad Cycles$$

$$Normalized\ Total\ Test\ Time = 1 + [(n*x*q)/(m-q)]$$

While the test time goes up, note that only one tester channel is needed for the output response, so all the other tester channels could be used for providing test stimulus thereby permitting the use of more scan chains and thereby lowering *c*. This actually results in a lower overall test time.

The other benefit of this scheme is that the same tester channels are used for both test vector decompression and MISR signature processing via time multiplexing. Hence, no additional control tester channels are needed other than one channel to stop and resume MISR operation. For the output response, a single tester channel can be used for transferring the *X*-canceled bits. The requirements can thus be summarized as follows:

Input Tester Channels: Decompressor Channels + 1
Output Tester Channels: 1

4.2 *X*-Canceling with Shadow Register

If it is not desirable to halt scan shifting to process the intermediate MISR signatures, an alternate approach would be to use a shadow register. Fig. 5 shows the *X*-canceling with shadow register architecture. The shadow register is placed after the main MISR and retains the intermediate signature for further processing. This allows the MISR to continue to compress the scan data without interruptions. Additional control inputs from the tester are used to provide the control signals to one or multiple selective XOR networks.

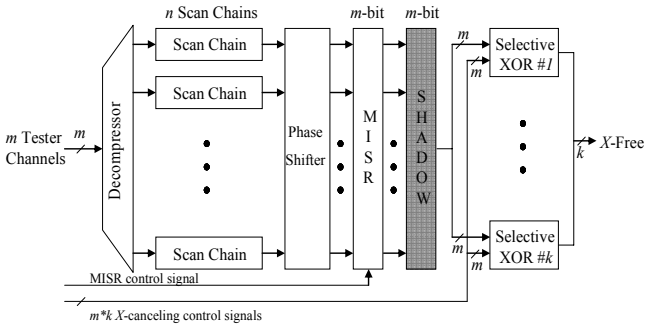

Figure 5. *X*-Canceling with Shadow Register

When the MISR fills up with *X*'s, the contents of the MISR are transferred to a shadow register, and the MISR is immediately reset so that scan shifting can continue uninterrupted. The saved intermediate signature in the shadow register is then processed to extract the *X*-canceled combinations as the next signature is being generated in a main MISR. Control signals need to be transferred while both the MISR and shadow register are operating. Therefore, extra tester channels are used to provide the control data that selects the *X*-canceled combinations.

In this scheme, because the shadow register gets rid of the additional test cycles for *X*-canceling, there is no additional test time penalty. As shown earlier, the error coverage depends on how many *X*-canceled combinations (*q*) are checked. *X*-canceling with time multiplexing requires *q* cycles to reach $1-2^{-q}$ error coverage during each signature processing phase. However, *X*-canceling with a shadow register only allows extracting *X*-canceled combinations before the next intermediate signature is transferred from the MISR to the shadow register. Calculating the theoretical error coverage is different in this case from what was done in Sec. 4.1. Fig. 5 shows *k* selective XOR gates after the shadow register. This allows *k* X-canceled combinations to be checked each clock cycle. However, the number of clock cycles over which the signature can be processed is limited by the time it takes for the MISR to fill up with *X*'s again. Let the "signature transfer period" be defined as the number of clock cycles from when one intermediate signature is

transferred from the MISR to the shadow register until the next one is transferred. The number of *X*-canceled combinations that are checked is determined by the number of selective XOR gates that are used times the number of cycles over which the signature is processed which is the signature transfer period. For *k* selective XOR gates, the error coverage is $1-2^{-k}$ after the first cycle. In the second cycle, the remaining errors that have not been covered yet are $(1-(1-2^{-k}))$, so the error coverage for them is again $1-2^{-k}$, hence the resulting error coverage after the second cycle is $(1-(1-2^{-k}))*(1-2^{-k})$ plus the error coverage after the first cycle. This is illustrated below:

$Cov_1 = 1 - 2^{-k}$ (Coverage at 1^{st} cycle)

$Cov_2 = Cov_1 + (1 - Cov_1) * (1-2^{-k})$ (Coverage at 2^{nd} cycle)

:

:

$Cov_s = Cov_{s-1} + (1 - Cov_{s-1}) * (1-2^{-k})$ (Coverage at s^{th} cycle)

Table 2 shows the error coverage for different values of *k* and the signature transfer period, *s*.

Table 2. Error Coverage for *X*-Canceling with Shadow Register Scheme

k XOR Gates (k Check/Cycle)	*s cycle (signature transfer cycle)*	Error Coverage
1	1	50.00%
	2	75.00%
	3	87.50%
	4	93.75%
2	1	75.00%
	2	93.75%
	3	98.43%
	4	99.60%
3	1	87.50%
	2	98.43%
	3	99.80%
	4	99.97%
4	1	93.75%
	2	99.60%
	3	99.97%
	4	99.99%

Unlike *X*-canceling with time multiplexing, *X*-canceling with a shadow register dedicates tester channels to provide control signals to the selective XORs. Hence, if *k* XOR gates (*k Checks/Cycle*) are used, *m*k* input tester channels are needed for driving them where *m* is the size of the MISR. And one input tester channel needs to be assigned to control when the MISR signature is transferred to the shadow register and reset. For the output response, *k* tester channels are required. The requirements can thus be summarized as follows:

Input Tester Channels: *Decompressor Channels + (MISR_size * Checks/Cycle) + 1*

Output Tester Channels: *Checks/Cycle*

5. Details of Industrial Designs

Two industrial designs from Intel were analyzed in detail for the experiments.

5.1 The First Test Case

The first test case (*Ckt1*) has 133 input and output tester channels respectively. *Ckt1* has a 10x compression ratio. 133 inputs are expanded into 1330 scan chains using Illinois scan [Hamzaoglu 99]. *Ckt1* has three sub-blocks (*A*, *B,* and *C*). 62, 38, and 31 output tester channels are assigned to *Ckt1-A*, *Ckt1-B,* and *Ckt1-C* respectively and 2 output channels are used for bypass mode. *Ckt1-A* has 1050 scan chains, *Ckt1-B* has 203 chains and *Ckt1-C* has 75 scan chains as shown in Fig. 6. The longest scan chain length is 481.

3000 automatic test pattern generation (ATPG) patterns were analyzed to determine the *X* density of each block in *Ckt1*. *Ckt1-A* has the least *X* density among three units. The *X* density is *0.07%* and the average number of *X*'s per each scan slice is *0.73* (*1050 * 0.07%*), i.e, *0.73 X*'s arrive at the compactor inputs every cycle. *Ckt1-B* has *3.35% X* density and the average number of *X*'s per scan slice is *6.8*. *3.28% X* density is found in *Ckt1-C* and this generates *2.46 X*'s per scan slice.

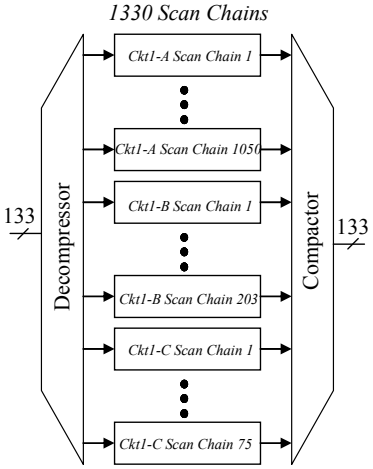

Figure 6. First Test Case

5.2 The Second Test case

The second test case (*Ckt2*) has relatively fewer test channels than the first test case in Sec. 5.1. *Ckt2* has 16 input and output tester channels. There are three partitions (*A*, *B* and *C*) in the design which are connected in a daisy chain manner. *Ckt2-A*, *Ckt2-B,* and *Ckt2-C* all

have 64 scan chains. *Ckt2* has a 4x compression ratio. 16 inputs are expanded to fill 64 scan chains.

Ckt2-A has a 2.01% *X* density and the average *X*'s per scan slice is 1.28. *Ckt2-B* has 1.05% *X* density which gives 0.67 average *X*'s in a scan slice. 2.74% *X* density is found in *Ckt2-C* and 1.76 *X*'s are in a scan slice, on average.

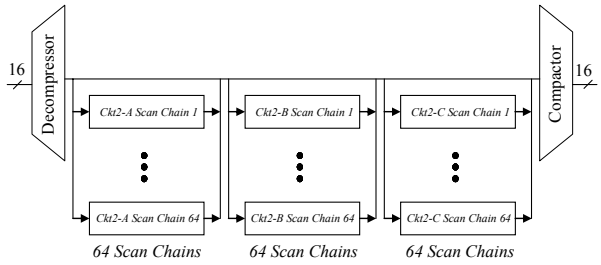

Figure 7. Second Test Case

6. Experimental Results

Experiments were performed for these two test cases described in Sec. 5. The *X*-canceling with time multiplexing and *X*-canceling with shadow register schemes are analyzed and compared with *X*-compact [Mitra 04a] which is widely used.

6.1 *X*-Canceling with Time Multiplexing

Table 3 shows the results for *X*-canceling with time multiplexing. A 32-bit MISR is used for each of the three blocks in *Ckt1* to compact the responses from the scan chains and to generate *X*-canceled combinations. The outputs of the scan chains are fed into a phase shifter before going to the MISR to reduce shift correlation [Touba 07]. The first column shows the circuits, and the second column shows the types of compactors. As shown in Sec. 4.1, the error coverage depends on how many *X*-canceled combinations (*q*) are checked. Results were generated for values of *q* ranging from 4 to 8. The third column shows the number of input and output tester channels used. The formula for the required number of input and output channels was given in Sec. 4.1. The number of two input XOR gates is shown in the fourth column. *X*-canceling with time multiplexing requires *(fanout * scan_chain + m) XOR gates* when an *m*-bit MISR is used. For *Ckt1-A*, a 32-bit MISR where each scan chain output fans out to 7 XOR gates in a phase shifter is used, so the number of two input XOR gates is 7,381 (*7*1050 + 31*) for *Ckt1-A*. The fifth column shows the test time for each scheme. The results are normalized with respect to the results for *X*-compact. The additional test time for control signal transfer (as described in Sec. 4.1) is also normalized and shown in the fifth column. The last column shows the error coverage. Unlike other schemes, the error coverage for an *X*-canceling MISR can be estimated based on the number of *X*-canceled combinations that are observed. The experimental results show what the theory would estimate the coverage and test time to be for purposes of comparison with the actual values. For *Ckt2*, a 64-bit MISR and a phase shifter with 5 fanouts per scan chain were used. Larger MISRs can hold more *X*'s before needing to be processed, however, they also require more data to process each signature, so the net effect is that test time and storage is relatively constant regardless of the MISR size. The main issue with the MISR size is the number of *X*'s in a single scan slice that it can handle. The MISR size should not be smaller than the maximum number of *X*'s in any scan slice.

As can be seen from Table 3, the proposed method achieves an error coverage and test time very close to that predicted by the theoretical formula. The reason for the slight deviation is that the formulas assume the MISR can stop when it takes exactly the full number of *X*'s values that it can hold. However, in practice, the *X*'s are entering the MISR in clusters scan slice by scan slice, so if the next scan slice puts the number of *X*'s over the limit, the MISR signature must first be processed before it can compact that scan slice. This results in some extra test time in comparison to that predicted by the theoretical formulas.

In comparing the results for *X*-canceling with *X*-compact, many fewer output tester channels are required while arbitrarily higher error coverage can be achieved to whatever the desired level is. For *Ckt1*, less overhead is required for *X*-canceling. For *Ckt2*, the overhead is very low for both methods. *X*-canceling with time multiplexing does have higher test time in this scenario because the output tester channels that have been reduced have not be used for providing test stimulus. Effectively, the tester bandwidth allocated for *X*-canceling here is less than that for *X*-compact.

Looking at the individual partitions, it can be seen that *Ckt1-A* has very low *X*-density, and both *X*-compact and *X*-canceling perform very well. *X*-canceling requires many fewer output tester channels and less overhead with a bit more test time. For *Ckt1-B* and *Ckt1-C*, the *X*-density is over 3% in both cases, and the error coverage provided by *X*-compact is low. This occurs because some scan slices have many *X*'s. Note that even though *Ckt1-B* and *Ckt1-C* have similar *X*-densities, the *X*-compact coverage for *Ckt1-B* is much lower. This is because the distribution of *X*'s in *Ckt1-B* is such that coverage is lost for a larger percentage of scan slices than in *Ckt1-C*. The *X*'s in *Ckt1-C* are more clustered in fewer scan slices, so the percentage of scan slices where coverage is lost is less. *X*-canceling can achieve high error coverage for any distribution of *X*'s, so it performs very well in terms of error coverage. The cost of achieving the higher error coverage is additional test time, but again fewer output

Table 3. Results for *X*-Canceling MISR with Time multiplexing Compared with *X*-Compact

Circuit	Compactor		Tester Channels		Num. XORs	Estimated Test Time (Normalized)	Actual Test Time (Normalized)	Estimated Error Coverage	Actual Error Coverage
			Input	Output					
Ckt1-A *X*-density = 0.07%	*X*-Compact		133	62	31,865	N/A	1	N/A	99.4%
	X-Canceling	q = 4	134	1	7,381	1.10	1.12	93.7%	93.7%
		q = 5	134	1	7,381	1.13	1.16	96.8%	96.8%
		q = 6	134	1	7,381	1.16	1.19	98.4%	98.4%
		q = 7	134	1	7,381	1.20	1.22	99.2%	99.2%
		q = 8	134	1	7,381	1.24	1.25	99.6%	99.6%
Ckt1-B *X*-density = 3.35%	*X*-Compact		133	38	4,135	N/A	1	N/A	36.9%
	X-Canceling	q = 4	134	1	1,452	1.97	2.42	93.7%	93.7%
		q = 5	134	1	1,452	2.25	2.78	96.8%	96.8%
		q = 6	134	1	1,452	2.56	3.13	98.4%	98.4%
		q = 7	134	1	1,452	2.90	3.49	99.2%	99.2%
		q = 8	134	1	1,452	3.26	3.85	99.6%	99.6%
Ckt1-C *X*-density = 3.28%	*X*-Compact		133	31	1,031	N/A	1	N/A	86.8%
	X-Canceling	q = 4	134	1	556	1.35	1.45	93.7%	93.5%
		q = 5	134	1	556	1.45	1.57	96.8%	96.6%
		q = 6	134	1	556	1.56	1.68	98.4%	98.2%
		q = 7	134	1	556	1.68	1.80	99.2%	99.0%
		q = 8	134	1	556	1.82	1.91	99.6%	99.3%
Ckt2-A *X*-density = 2.01%	*X*-Compact		16	16	192	N/A	1	N/A	95.4%
	X-Canceling	q = 4	17	1	447	1.34	1.35	93.7%	93.6%
		q = 5	17	1	447	1.43	1.44	96.8%	96.7%
		q = 6	17	1	447	1.52	1.54	98.4%	98.3%
		q = 7	17	1	447	1.62	1.64	99.2%	99.1%
		q = 8	17	1	447	1.73	1.74	99.6%	99.5%
Ckt2-B *X*-density = 0.67%	*X*-Compact		16	16	192	N/A	1	N/A	97.9%
	X-Canceling	q = 4	17	1	447	1.17	1.18	93.7%	93.6%
		q = 5	17	1	447	1.22	1.23	96.8%	96.7%
		q = 6	17	1	447	1.27	1.28	98.4%	98.3%
		q = 7	17	1	447	1.32	1.33	99.2%	99.1%
		q = 8	17	1	447	1.38	1.39	99.6%	99.5%
Ckt2-C *X*-density = 2.74%	*X*-Compact		16	16	192	N/A	1	N/A	92.7%
	X-Canceling	q = 4	17	1	447	1.46	1.48	93.7%	93.5%
		q = 5	17	1	447	1.59	1.60	96.8%	96.6%
		q = 6	17	1	447	1.72	1.74	98.4%	98.2%
		q = 7	17	1	447	1.86	1.87	99.2%	99.0%
		q = 8	17	1	447	2.00	2.01	99.6%	99.4%

tester channels are required. For *Ckt2,* *X*-compact is using 32 tester channels, while *X*-canceling is using only 18 tester channels. If the 14 tester channels that are reduced with *X*-canceling were to be employed in providing test stimulus, then *X*-canceling would have lower test time in all cases while providing greater error coverage.

6.2 *X*-Canceling with Shadow Register

Results for *X*-canceling with shadow registers are shown in Table 4. The control signals for generating the *X*-canceled combinations are provided by dedicated tester channels rather than through time multiplexing, so the test time is exactly the same for both *X*-compact and *X*-canceling. The first column shows the circuits and the second column shows the types of compactors with different numbers of checks/cycle. The number of required input and output tester channels and the number of two input XOR gates are shown in the third and fourth column respectively. The last column shows the error coverage.

As before, the error coverage for *X*-canceling can be made arbitrarily high. In this case, improving the error coverage comes at the cost of requiring more checks/cycle which requires more input tester channels, however, the test time remains constant.

Table 4. Results for *X*-Canceling MISR with Shadow Register Compared with *X*-Compact (for the Same Test Time)

Circuit	Compactor	Checks/Cycle	Tester Channels		Num. XORs	Estimated Error Coverage	Actual Error Coverage
			Input	Output			
Ckt1-A	*X*-Compact	N/A	133	62	31,865	N/A	99.4%
	X-Canceling 12-Bit MISR	1	146	1	5,261	93.7%	93.7%
		2	158	2	5,272	99.6%	98.2%
		3	170	3	5,283	99.9%	99.1%
		4	182	4	5,294	99.9%	99.4%
Ckt1-B	*X*-Compact	N/A	133	38	4,135	N/A	36.9%
	X-Canceling 19-Bit MISR	1	148	1	1,845	75.0%	74.4%
		2	162	2	1,863	93.7%	90.2%
		3	176	3	1,881	98.4%	97.9%
		4	190	4	1,899	99.6%	98.9%
Ckt1-C	*X*-Compact	N/A	133	31	1,031	N/A	86.8%
	X-Canceling 14-Bit MISR	1	143	1	1,028	87.5%	87.3%
		2	152	2	1,041	98.4%	95.8%
		3	161	3	1,054	99.8%	97.6%
		4	170	4	1,067	99.9%	98.8%
Ckt2-A	*X*-Compact	N/A	16	16	192	N/A	95.4%
	X-Canceling 16-Bit MISR	1	33	1	463	93.75%	93.60%
		2	49	2	478	99.60%	98.01%
		3	65	3	493	99.97%	98.92%
		4	81	4	508	99.99%	99.20%
Ckt2-B	*X*-Compact	N/A	16	16	192	N/A	97.9%
	X-Canceling 16-Bit MISR	1	33	1	463	93.75%	93.67%
		2	49	2	478	99.60%	98.07%
		3	65	3	493	99.97%	98.96%
		4	81	4	508	99.99%	99.24%
Ckt2-C	*X*-Compact	N/A	16	16	192	N/A	92.7%
	X-Canceling 16-Bit MISR	1	33	1	463	93.75%	93.63%
		2	49	2	478	99.60%	98.20%
		3	65	3	493	99.97%	99.10%
		4	81	4	508	99.99%	99.39%

6.3 Fault Coverage Results

Fault grading was performed on *Ckt1* to see the actual fault coverage that is achieved by the *X*-canceling methods and *X*-Compact. For each block in *Ckt1*, a 32-bit MISR with q = 8 configuration is used for *X*-canceling with time multiplexing. For *X*-canceling with shadow registers, a configuration was selected which has a similar number of tester channels to *X*-Compact for a fair comparison. The following is used: 12-bit MISR with 4 checks/cycle for *Ckt1-A*, 19-bit MISR with 2 checks/cycle for *Ckt1-B,* and 14-bit MISR with 2 checks/cycle for *Ckt1-C*. This configuration requires 268 tester channels and *X*-Compact needs 266 channels. The fault coverage for 3000 ATPG patterns is shown in Fig. 8. Without any compression, slightly over 90% fault coverage is obtained. As shown in Table 3 and Table 4, the *X*-canceling MISR schemes achieve high error coverage which translates to fault coverage which is very close to what is obtained without any compression. The fault coverage for *X*-Compact, however, is 2~3% lower.

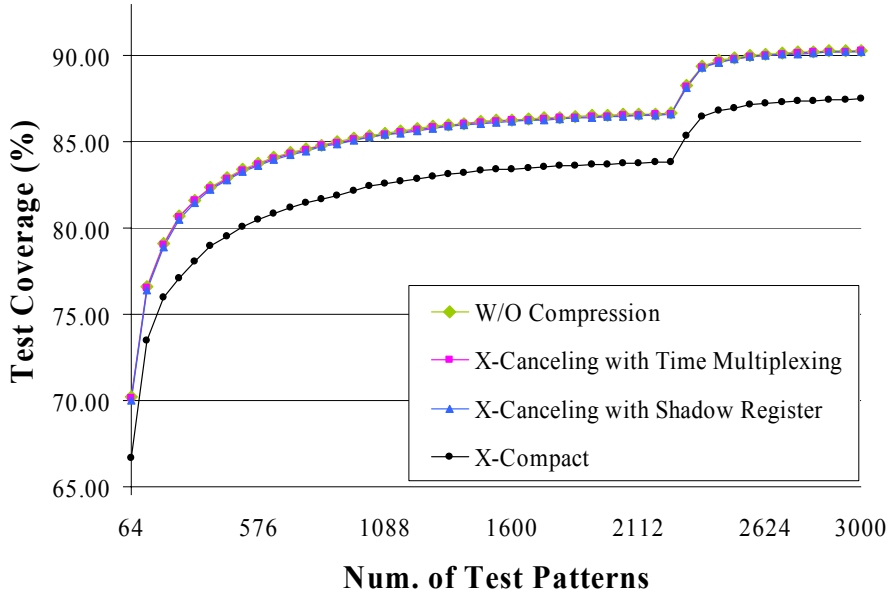

Figure 8. Fault Grading Results for *Ckt1* with Different Schemes

7. Conclusions

This industrial case study shows the benefits of *X*-canceling in terms of its scalability and ability to systematically achieve high fault coverage regardless of the distribution of *X*'s. Two different architectures were presented for *X*-canceling which can be used based on what the tester channel and test time requirements are for a particular design. It was also shown the theoretical equations for estimating the error coverage for *X*-canceling matched closely with the actual error coverage achieved in the experiments.

References

[Barnhart 01] Barnhart, C., V. Brunkhorst, F. Distler, O. Farnsworth, B. Keller, and B. Koenemann, "OPMISR: the Foundation for Compressed ATPG Vectors," *Proc. of International Test Conference*, pp. 748-757, 2001.

[Chao 05] Chao, M. C.-T., S. Wang, S.T. Chakradhar, and K.-T. Cheng, "Response Shaper: A Novel Technique to Enhance Unknown Tolerance for Output Response Compaction," *Proc. of International Conference on Computer-Aided Design*, pp. 80-87, 2005.

[Chickermane 04] Chickermane, V., B. Foutz, and B. Keller, "Channel Masking Synthesis for Efficient On-Chip Test Compression," *Proc. of International Test Conference*, pp. 452-461, 2004.

[Cullen 97] Cullen, C.G., *Linear Algebra with Applications*, Addison-Wesley, ISBN 0-673-99386-8, 1997.

[Garg 08] Garg, R., R. Putman, and N.A. Touba, "Increasing Output Compaction in Presence of Unknowns Using an X-Canceling MISR with Deterministic Observation," *Proc. of IEEE VLSI Test Symposium*, pp. 35-42, 2008.

[Hamzaoglu 99] Hamzaoglu, I., and J.H. Patel, "Reducing Test Application Time for Full Scan Embedded Cores," *Digest of Papers 29th Ann. Int'l Symp. Fault Tolerant Computing*, IEEE Press, pp. 260-267, 1999.

[Mitra 04a] Mitra, S., and K.S. Kim, "X-Compact: An Efficient Response Compaction Scheme," *IEEE Trans. on Computer-Aided Design*, Vol. 23, No. 3, pp. 421-432, Mar. 2004.

[Mitra 04b] Mitra, S., S.S. Lumetta, and M. Mitzenmacher, "X-Tolerant Signature Analysis," *Proc. of International Test Conference*, pp. 432-441, 2004.

[Patel 03] Patel, J.H., S.S. Lumetta, and S.M. Reddy, "Application of Saluja-Karpovsky Compactors to Test Responses with Many Unknowns," *Proc. of VLSI Test Symposium*, pp. 107-112, 2003.

[Pomeranz 02] Pomeranz, I., S. Kundu, and S.M. Reddy, "On Output Response Compression in the Presence of Unknown Output Values," *Proc. of Design Automation Conference*, pp. 255-258, 2002.

[Rajski 05] Rajiski, J., J. Tyszer, C. Wang and S.M. Reddy, "Finite Memory Test Response Compactors for Embedded Test Applications," *IEEE Trans. on Computer-Aided Design*, Vol. 24, No. 4, pp. 622-634, Apr. 2005.

[Rajski 06a] Rajski, J., J. Tyszer, G. Mrugalski, W.-T. Cheng, N. Mukherjee, and M. Kassab, "X-Press Compactor for 1000x Reduction of Test Data," *Proc. of International Test Conference*, Paper 18.1, 2006.

[Rajski 06b] Rajski, W., and J. Rajski, "Modular Compactor of Test Responses," *Proc. of VLSI Test Symposium*, pp. 242-251, 2006.

[Patel 03] Patel, J.H., S.S. Lumetta, and S.M. Reddy, "Application of Saluja-Karpovsky Compactors to Test Responses with Many Unknowns," *Proc. of VLSI Test Symposium*, pp. 107-112, 2003.

[Sharma 05] Sharma M. and W.-T. Cheng, "X-Filter: Filtering Unknowns from Compacted Test Responses," *Proc. of International Test Conference*, Paper 42.1, 2005.

[Tang 06] Tang, Y., H.-J. Wunderlich, P. Engelke, I. Polian, B. Becker, J. Scholöffel, F. Hapke, and M. Wittke, "X-Masking During Logic BIST and Its Impact on Defect Coverage," *IEEE Trans. on VLSI*, Vol. 14, No. 2, Feb. 2006.

[Touba 07] Touba, N.A., "X-Canceling MISR – An X-Tolerant Methodology for Compacting Output Responses with Unknowns Using a MISR," *Proc. of International Test Conference*, paper 6.2, 2007

[Volkerink 05] Volkerink, E.H., and S. Mitra, "Response Compaction with Any Number of Unknowns Using a New LFSR Architecture," *Proc. of Design Automation Conference*, pp. 117-122, 2005.

[Wang 06] L.T. Wang, C.-W. Wu, X. Wen, *VLSI Test Principles and Architectures*, Morgan Kaufmann, 2006.

[Wohl 01] Wohl, P., J.A. Waicukauski, and T.W. Williams, "Design of Compactors for Signature-Analyzers in Built-In Self-Test," *Proc. of International Test Conference*, pp. 54-63, 2001.

[Wohl 03] Wohl, P., J.A. Waicukauski, S. Patel, and M.B. Amin, "X-Tolerant Compression and Application of Scan-ATPG Patterns in a BIST Architecture," *Proc. of International Test Conference*, pp. 727-736, 2003.

[Wohl 04] Wohl, P., J.A. Waicukauski, and S. Patel, "Scalable Selector Architecture for X-Tolerant Deterministic BIST," *Proc. of Design Automation Conference*, pp. 934-939, 2004.