

# Compressed Pattern Diagnosis For Scan Chain Failures

Yu Huang, Wu-Tung Cheng, Janusz Rajski

Mentor Graphics Corporation Wilsonville, OR, USA

## Abstract

*In scan based designs, 10%-30% defects are in scan chains. Hence scan chain fault diagnosis becomes an important process for silicon debug and yield ramp up. With embedded compression techniques getting popular, chain diagnosis on devices with the embedded compression techniques becomes a challenge. In this paper, we provide a general methodology that can be applied for performing chain diagnosis in the context of any embedded compression techniques with any existing chain diagnosis algorithms. The proposed methodology enables seamless reuse of the existing chain diagnosis infrastructure with compressed test data. Experimental results show that with compressed patterns, the chain diagnosis resolution can be enhanced up to one order of magnitude with only 25% of failure cycles collected from ATE, compared to the diagnosis results with uncompressed patterns.*

## 1. Introduction

With the increase of design size and the importance of at-speed test and multiple-detect test, the test data volume far exceed the capacities of the automatic test equipment (ATE). Due to the limited bandwidth between the IC and the ATE, this increased test data also results in increased test time and hence increased test costs.

Several compression techniques such as Embedded Deterministic Test (EDT) [RAJ02], OPMISR [BAR02], X-Compactor [MIT02] and X-DBIST [WOH03] have recently been proposed to contain the test costs while achieving the required test quality levels. These compression techniques involve two complimentary aspects – hardware that is placed on chip and software ATPG techniques that exploit the on-chip hardware to create highly compressed test sets that give dramatic reduction in test data volume and test application time. In EDT for example, a hardware block called *Decompressor* is placed along the scan path between the scan input channels and inputs of the scan chains. This *Decompressor* performs a transformation of the data that is supplied by the ATE on few scan input channels and feeds the large number of internal scan chains. A

hardware block called *Compactor* that is placed along the scan path on the output side transforms the data captured in the large number of internal scan chains into a data stream that is delivered on few scan output channels for comparison with the expected values by the ATE. EDT has successfully been adopted by many companies to achieve dramatic reduction in test data volume and test application time. EDT architecture is illustrated in Figure 1.

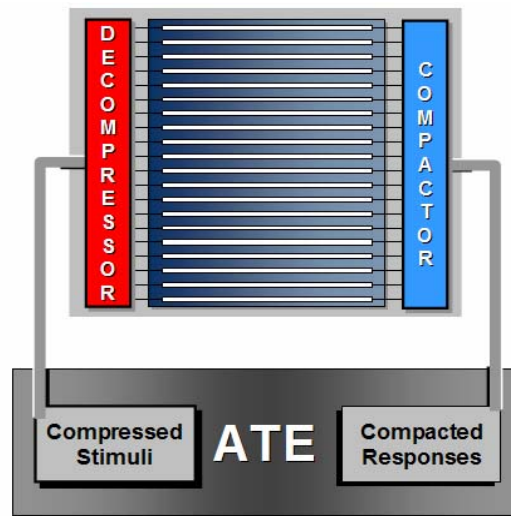


Figure 1: High Level EDT Architecture

IC manufacturing test typically includes two phases. The first test phase is chain test. The purpose of chain test is to check if the scan chains work as expected. With the compression techniques such as EDT, chain test also checks the embedded compression logic in addition to scan chains. If the chain test passed, it will go to the second test phase, which is called scan test. The purpose of scan test is to check if there are any faults in the system logic.

When a test failed, automatic diagnosis plays a vital role in the failure analysis process wherein the root cause of a failure is determined. Failure analysis has several applications – for example yield learning and improvement when a new technology is introduced or identifying systemic manufacturing issues that impact the yield. The diagnosis results report the suspect sites that

are used in probing the IC with devices such as scanning electron microscope (SEM).

Accordingly, there are two types of fault diagnosis. Chain diagnosis is performed to locate which scan chain(s) and scan cell(s) have defects. Scan diagnosis is performed to identify the suspect fault sites in the system logic. In this paper, we investigate the chain diagnosis in the context of embedded compression techniques.

Without embedded compression techniques, automated chain diagnosis has widely been studied [KUN94] [GUO01] [STA01] [HUA03] [HUA04]. With the compression techniques, however, the automated chain diagnosis needs additional efforts. This is due to the fact that the data that is applied to IC and the response that is captured undergo a transformation. Thus what is observed on the ATE from scan channel outputs is a compacted version of what is captured in the internal scan elements of the IC. Automated fault diagnosis has to be able to take failure log corresponding to the compacted response stream and determine the suspects. One way, to avoid having to deal with the failure log of compacted response stream would be to completely bypass the on-chip compression hardware. Doing so provides direct access to the scan cells enabling the application of the well-established standard ATPG based fault diagnosis. However, one of the major drawbacks of such an approach is the requirement for two separate test sets – a compressed test set for volume production testing purposes to contain the test costs and an uncompressed test set for fault diagnosis purposes. In addition, such an approach does not facilitate on-line diagnosis, which is fast diagnosis based on the volume production test results. Thus it is desirable to enhance the fault diagnosis techniques to be applicable in the context of compressed test sets.

In recent work [CHE04], a compactor-independent direct scan diagnosis methodology was proposed. For chain diagnosis, however, there is no prior work to solve the problem. In this paper, we propose a compactor-independent compressed pattern chain diagnosis methodology, which is based on circuit transformation technique that can be applied for performing chain diagnosis in the context of any compression method. The proposed technique enables seamless reuse of the existing chain diagnosis infrastructure.

The rest of this paper is outlined as follows. Section 2 reviews chain diagnosis methodologies without embedded compression techniques. Section 3 outlines a general flow and algorithm for chain diagnosis in the context of embedded compression techniques. Moreover, single faulty chain diagnosis by using the proposed methodology is illustrated by a popular industrial compactor – EDT compactor. Experimental results on single faulty chain diagnosis with EDT compactor are given in Section 4.

Multiple faulty chain diagnosis is discussed in Section 5 followed by the conclusions drawn in Section 6.

## 2. Prior Chain Diagnosis Algorithms

The existing chain diagnosis algorithms can be classified into two main categories. The first category is hardware-based chain diagnosis, which uses special scan cell and additional scan circuitry [SCH92] [WU98] [EDI95] [NAR97]. The second category is software-based chain diagnosis [KUN94] [GUO01] [STA01] [HUA03] [HUA04]. The hardware-based methods use some special scan chain designs to facilitate the scan chain diagnosis process. E.g., in [SCH92], it is proposed to connect the output of each scan cell to another scan cell on another scan chain such that its value can be observed by the other scan chain in diagnosis mode. In [EDI95], XOR gates are inserted between scan cells to enhance chain diagnosis. In [NAR97] special set/reset circuitry is used to enhance chain diagnosis. In [WU98], special circuitry is proposed to flip and set/reset scan cells to identify defective cells.

The software-based algorithms do not need any modification of the basic scan circuitry. In [KUN94], sequential ATPG algorithm is used to diagnose chain failures. In [GUO01] and [STA01] fault simulation and matching algorithms are applied to identify the defective scan cells. In [HUA03] and [HUA04], algorithms are proposed to diagnose realistic chain defects by targeting intermittent scan chain faults. Compared with the hardware-based methods, software-based techniques are more attractive due to no design modification.

Without embedded compressions, a chain diagnosis procedure typically includes two steps.

Step 1: Identify faulty chains and fault types by chain patterns [GUO01] [HUA03].

**Table 1: Various Scan Chain Fault Types**

Fault Types	Unloaded Values-Permanent Faults	Unloaded Values-Intermittent Faults
Slow-to-Rise	00100010001X	00110010001X
Slow-to-Fall	01110111011X	01110011011X
Slow	01100110011X	00100111011X
Fast-to-Rise	X01110111011	X01110110011
Fast-to-Fall	X00100010001	X00100110001
Fast	X00110011001	X00100111001
Stuck-at-0	00000000000	001000010000
Stuck-at-1	111111111111	101111111011
Stuck-at-X	—	100100111110

This step is illustrated by a chain test example given in Table 1. Suppose a scan chain is loaded with a chain pattern 001100110011, where the leftmost bit is loaded into the scan cell (cell 11) connected to scan chain input and the rightmost bit is loaded into the scan cell (cell 0) connected to scan chain output. Column 2 in Table 1 gives the unloaded faulty values for each type of permanent fault. Column 3 gives examples of the unloaded faulty values for each type of intermittent fault. Note that the “X”s depend on the previous or the next chain pattern.

Step 2: Locate the faulty scan cells by special scan circuitry (hardware-based diagnosis) or by simulation of scan patterns (software-based diagnosis).

In this step, suspect scan cells will be located. This is the step where hardware-based chain diagnosis is different from software-based chain diagnosis. In hardware-based chain diagnosis, some known values are loaded into the faulty chain by special chain circuitry (e.g. special set/reset). The defective scan cells will be located during the unloading procedure. For example, it is known that chain1 has a stuck-at-0 fault in the first step. Then the circuit is set to chain diagnosis mode and all the scan cells on chain1 are set to “1” by using special scan chain circuitry. During the unloading procedure, the first cell unloaded a “0” will tell where the defective scan cell is. In software-based diagnosis, this step is usually based on simulation with scan patterns. A fault is “injected” on a scan cell in software simulation context. Loading values in the downstream of this scan cell on this faulty chain will be modified for all scan patterns due to the fault. E.g., a scan pattern has good machine loading value 001110011010 on chain1. If we inject a stuck-at-0 fault on scan cell 3, the loading value will be modified as 001110010000. After pulsing the capture clock, the captured values in the upstream of the faulty scan cell on this faulty chain will be modified. E.g., if the simulated captured value is 101011101011, the unloading values will be 000000000011. This simulation results will be compared with the observed results from ATE. The best matching cell(s) will be reported as suspect(s). However, these chain diagnosis methodologies cannot be applied directly for designs with embedded compression techniques.

### 3. Proposed Scan Chain Diagnosis Methodology With Compressed Patterns

In the following sections, we focus on *single* faulty scan chain diagnosis methodology. Diagnosis for multiple faulty chains will be discussed in Section 5.

A general flow of diagnosing defects in scan chains with compressed patterns includes the following three steps.

(1) Model a compactor as a function  $\Phi$  such that  $\Phi(R) = r$ , where  $R$  is the test response before compaction and  $r$  is

the test response after compaction. The original circuit is then transformed into a circuit with pseudo-scan chains by incorporating compactor function  $\Phi$ . This step is same as proposed in [CHE04].

(2) Read the failure log file for chain test and identify faulty chains and fault types. In this step, masking patterns (explained below) are required to apply in chain test. It is straightforward to identify faulty chain(s) from pseudo-scan chains by using masking chain patterns.

(3) Read the failure log file for scan test. With scan test patterns, apply any previously proposed chain diagnosis algorithm by performing the fault simulation on faulty scan chain against the transformed circuit. Compare the simulation results with the compacted failure data collected from tester. Report the best matching scan cell(s) as suspect(s).

The advantages of the proposed chain diagnosis methodology include:

(1) It can be generally applied to a variety of embedded compression techniques.

(2) It can exploit a variety of existing scan chain diagnostic algorithms.

(3) As to be shown later in Section 4, chain diagnosis with embedded compressions can achieve better diagnosis resolution with less collected failure cycles.

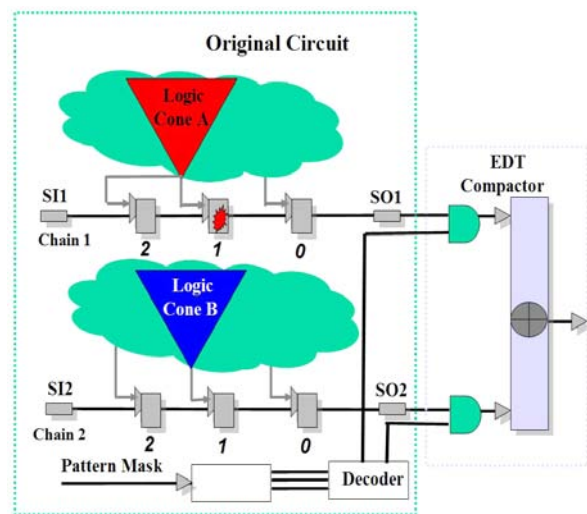


Figure 2: An Example of a Design with an EDT Compactor.

In the rest part of this section, we use EDT Compactor as an example to illustrate the proposed scan chain diagnostic methodology. EDT Compactor [RAJ02] is a linear space compactor. It is essentially an XOR tree with scan chain masking logic to select one chain per channel

to observe. A simple example circuit with EDT Compactor is shown in Figure 2. It has 2 scan chains that feed into an XOR gate.

Assuming a stuck-at-0 fault is at cell 1 of chain 1, we apply the 3 steps proposed above on this example.

Step (1): We transform the circuit into a circuit with an embedded compactor function and build a pseudo scan chain. We represent the test responses for each pattern before compaction and after compaction as matrixes  $R$  and  $r$  respectively. The compaction function is  $\Phi(R)=r$ . For the given example,  $\Phi=[1\ 1]$  if no scan chain is masked,  $\Phi=[0\ 1]$  if the first scan chain is masked and  $\Phi=[1\ 0]$  if the second scan chain is masked. Note that  $\Phi$  could be pattern dependent.

Step (2): For EDT compactor [RAJ03], a masking pattern is a pattern such that only one scan chain is observed from one compactor channel output while all the other scan chains connecting to the same channel output are masked (blocked). An example of EDT masking logic is illustrated in Figure 3. A masking code is shifted into the mask register. After decoding, only chain output 1 is observed from channel output 1 while all the other chains are masked for this specific masking pattern.

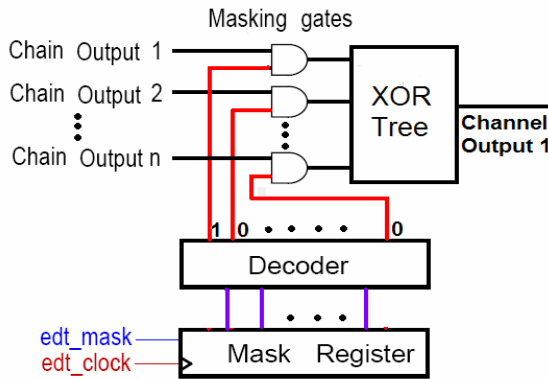


Figure 3: EDT Compactor Masking Logic

In EDT chain test set, each internal chain is observed uniquely from one channel output with one masking chain test pattern. Hence, for a  $NX$  compaction configuration, it must have at least  $N$  chain masking patterns. Using masking chain patterns, it is straightforward to identify the faulty chain and to model the chain fault. In the given example, assuming chain pattern 1 is a masking pattern to observe chain 1 and chain pattern 2 is a masking pattern to observe chain 2. Also suppose the patterns have loading values on internal chains and expected / observed values on channel output shown in Table 2.

Table 2. An Example of EDT Chain Patterns

	Chain1 Load	Chain2 Load	Compactor Output	
			Expected	Observed
Pattern 1	101	001	101	000
Pattern 2	110	100	100	100

For this example, chain pattern 1 failed and chain pattern 2 passed. We deduce that chain 1 has a stuck-at-0 fault.

Step (3): To narrow down the defect location on chain 1, regular scan patterns are used. One candidate fault is injected at a time. Fault simulation is then performed on the transformed circuit by taking consideration of compaction function  $\Phi$  on the test responses. Then the compacted simulation results on the channel outputs are compared with the data observed from the tester. Finally, the most matched suspect cell(s) is reported. The compressed pattern chain diagnosis is achieved.

#### 4. Experimental Results

By using EDT as an example, we perform two sets of experiments. The purpose of the first set of experiments is to investigate the diagnosis accuracy, resolution and performance of the proposed compressed pattern chain diagnosis algorithm. The purpose of the second set of experiments is to investigate how the diagnosis resolution varies with the number of collected failure patterns.

We use one small circuit (Ckt1) and one large industrial circuit (Ckt2) to perform the experiments, hoping the experimental results are valid for a large number of designs of different sizes. Information about these two circuits is given in Table 3. The numbers of gates, scan cells and scan chains are given for these two circuits. After incorporating EDT, the compaction ratios used for the experiments are set to  $2X$ ,  $4X$ ,  $8X$ ,  $16X$ ,  $32X$  and  $64X$ . The numbers of scan channels are fixed. Ckt1 has 2 scan channels and Ckt2 has 14 scan channels respectively. For a compaction ratio  $N$ , we break each original scan chain into  $N$  scan chains in a balanced manner such that the length of each internal scan chain is close to  $1/N$  of the original scan chain length. We connect every  $N$  scan chains to one scan channel output through EDT compactor. For chain defect, the number of failure cycles per pattern is typically proportion to the faulty scan chain length. To make a fair comparison of the diagnosis results, we have to collect the similar number of failure cycles no matter which  $N$  we select. Thus for a  $NX$  compaction scheme, the number of failing patterns we used for chain diagnosis is about  $N$  times of the patterns for the situation without EDT. E.g. we inject a stuck-at-1 fault at one scan cell in Ckt1. Without EDT, we apply 5 scan patterns on Ckt1 with 2 scan chains and collect about 500 failure cycles. If set  $N=16$ , Ckt1 has 32 internal scan chains with

2 EDT compactor channels. We now can apply 80 scan patterns and collect about 500 failure cycles.

**Table 3. Circuit Information**

	Ckt1	Ckt2
# Gates	9.4K	6.5M
# Scan Cells	495	529826
# Scan Chains	2	14

#### 4.1 Experiments with the Proposed Chain Diagnosis Methodology

To validate the proposed chain diagnosis methodology and compare the chain diagnosis accuracy, resolution and performance with and without EDT, we create chain failure test cases in the following way.

Since Ckt1 has only 495 scan cells, we make 495 test cases by injecting one permanent stuck-at-1 fault per test case. Each case has a fault at a different scan cell. With Ckt1, we also make 495 test cases by injecting one intermittent stuck-at-1 fault per test case. For each injected intermittent fault, we set fault triggering probability to 50%.

Similarly, we randomly select 100 scan cell locations in Ckt2 to inject one fault per case. We end up with 100 permanent stuck-at-1 failure cases ( $Prob=100$ ) and 100 intermittent stuck-at-1 ( $Prob=50$ ) failure cases by using Ckt2.

For these  $(495*2+100*2)$  chain failure cases, we first get one failure log file per case by simulating the patterns against the circuit with the injected fault. Note that the failure response is compacted if using EDT. We then perform chain diagnosis on each failure log file. We use EDT with 64X compaction for Ckt1 and 32X compaction for Ckt2 in the experiments. All the diagnosis reports include the injected failure scan cell. So diagnosis accuracy has no problem. The diagnosis resolution and performance are shown in Table 4 and Table 5 for Ckt1 and Ckt2 respectively.

In Tables 4 and 5, row 2 shows the number of failure patterns used for diagnosis. Row 3 shows the average number of failure cycles over  $(495*2)$  test cases for Ckt1 in Table 4 and  $(100*2)$  test cases for Ckt2 in Table 5.

As shown in Column 3 of Tables 4 and 5, we initially collect 5 failed patterns for Ckt1 and 8 failed patterns for Ckt2 without EDT. These numbers of patterns are selected randomly such that it leads to reasonable number of failure cycles in this scenario. As we pointed out above, for a design with  $NX$  EDT compaction,  $N$  times of scan patterns can be applied resulting in similar number of

failure cycles collected from ATE, compared to the same design without EDT. As illustrated in Table 4, applying 320 patterns to Ckt1 design with 64X EDT only results in 397 failure cycles in average while applying 5 patterns to Ckt1 design without EDT results in 473 failure cycles in average. As illustrated in Table 5, applying 256 patterns to Ckt2 design with 32X EDT results in 126K failure cycles in average while applying 8 patterns to Ckt2 design without EDT results in 139K failure cycles in average. This is because that EDT designs have much shorter faulty chains, which causes much less number of failure cycles and these failures are compacted at the channel output.

Therefore we collect 320 failed patterns for Ckt1 with 64X EDT and 256 failed patterns for Ckt2 with 32X EDT, which are shown in column 6 of Tables 4 and 5. To compare the diagnosis results with the same number of patterns, we also collect 320 failed patterns for Ckt1 without EDT and 5 failed patterns for Ckt1 with 64X EDT, as illustrated in Columns 4 and 5 in Table 4. Also, we collect 256 failed patterns for Ckt2 without EDT and 8 failed patterns for Ckt2 with 32X EDT, as illustrated in Columns 4 and 5 in Table 5.

**Table 4. Ckt1 Chain Diagnosis Results**

		No EDT	No EDT	EDT 64X	EDT 64X
	# Patterns	5	320	5	320
	Avg. # Failure Cycles	473	30126	6	397
Avg. # Susp.	$Prob=100$	10.38	3.01	2.57	1.01
	$Prob=50$	20.98	6.11	3.03	1.62
Susp  $\leq 2$	$Prob=100$	37.7%	80.7%	50.2%	100%
	$Prob=50$	13.6%	44.3%	38.1%	57%
Avg. Time (s)	$Prob=100$	0.54	2.32	0.15	1.21
	$Prob=50$	4.76	8.71	0.32	2.05

**Table 5. Ckt2 Chain Diagnosis Results**

		No EDT	No EDT	EDT 32X	EDT 32X
	# Patterns	8	256	8	256
	Avg. # Failure Cycles	139K	4535K	5.5K	126K
Avg. # Susp.	$Prob=100$	19.75	4.9	16.9	1.25
	$Prob=50$	87.44	19.2	78.8	13.9
Susp  $\leq 2$	$Prob=100$	50%	91%	50%	93%
	$Prob=50$	9%	25%	13%	35%
Avg. Time (s)	$Prob=100$	358	2468	296	1621
	$Prob=50$	607	3077	592	2674

Rows 4 and 5 in Tables 4 and 5 show the average number of the reported suspects after diagnosis. The diagnosis resolution is defined as  $1 / (\# \text{ Average Suspects})$ . When applying the same number of scan patterns (i.e., comparing column 3 vs. column 5 and column 4 vs. column 6), the diagnosis resolution is always enhanced by incorporating EDT for both Ckt1 and Ckt2. The enhancement ratios vary for different circuits, different number of patterns and different fault types. Note that when applying the same number of patterns, the number of failure cycles becomes much less if incorporating EDT.

When applying the similar number of failure cycles (i.e., comparing column 3 vs. column 6), for Ckt1, the diagnosis resolution is improved about 10 times and 13 times for permanent faults and intermittent faults respectively by incorporating EDT. The diagnosis resolution for Ckt2 is improved 15.8 times and 6.3 times for permanent faults and intermittent faults respectively by incorporating EDT.

Rows 6 and 7 in Tables 4 and 5 show the percentage of the test cases with only one or two suspects after chain diagnosis. If the number of suspects is small (e.g. less than 3), it will be quite helpful for failure analysis at physical level. The experimental results indicate that for permanent stuck-at-1 fault, we have 100% chances for Ckt1 and 93% of chances for Ckt2 to get useful diagnosis results for failure analysis if we use EDT with reasonable number of failure cycles.

The last two rows in Tables 4 and 5 show the average diagnosis run time for each set of test cases. The run time of chain diagnosis is determined by two factors: (1) the number of simulated scan patterns and (2) the number of candidates that need simulation.

Because for the similar number of failure cycles, the simulated patterns with  $NX$  EDT is typically  $N$  times of the patterns without EDT, the diagnosis run time with EDT designs is usually longer due to reason (1) list above. However, for EDT designs, because they have shorter chains, they usually have a smaller number of candidates to simulate than without using EDT, which may explain why the run time with EDT is even shorter than without EDT for Ckt1, when fault triggering probability is set to 50% and using similar number of failure cycles. When using the same number of patterns, diagnosis of EDT designs always runs faster.

#### 4.2 Diagnosis Resolution Vs. Number of Simulated Patterns

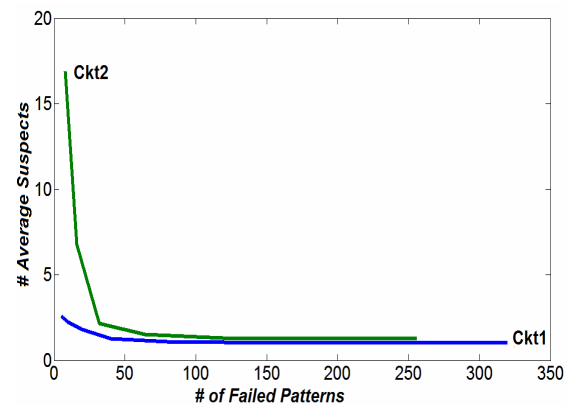
As we know, one of the factors limiting the resolution of the chain diagnostics is the amount of failing patterns / cycles that can be logged in reasonable time. In the experiments described in the previous subsections, for Ckt2, if we set the datalog limit to 150K failure cycles,

without compressions, a datalog of such size would result in only 8 failing patterns given that the average chain length is more than 37K. Collecting this magnitude of data on some types of ATE may take hours. In this subsection, we investigate how the diagnosis resolution varies with the number of collected failure patterns in the context of EDT.

**Table 6. Average # Suspects Vs. # Patterns**

Ckt1 with 64XEDT		Ckt2 with 32XEDT	
#Patterns	Avg. Suspects	#Patterns	Avg. Suspects
5	2.57	8	16.9
10	2.22	16	6.78
20	1.79	32	2.16
40	1.25	64	1.53
80	1.08	128	1.25
160	1.03	256	1.25
320	1.01	-	-

The average number of suspects vs. the number of failed patterns is illustrated in Table 6 and plotted in Figure 4 when using Ckt1 with 64X EDT and Ckt2 with 32X EDT. We calculate the average number of suspects by using 495 permanent stuck-at-1 test cases for Ckt1 and 100 permanent stuck-at-1 test cases for Ckt2.



**Figure 4: Avg. # Diag Suspects vs. # Patterns**

From Table 6, it shows that for Ckt1, after applying 80 scan patterns, the diagnosis resolution is almost saturated. Applying 80 scan patterns with 64X EDT for Ckt1 can improve the chain diagnosis resolution about 10 times with only 25% of the number of collected failure cycles compared to applying 5 patterns on Ckt1 without EDT. For Ckt2, after applying 64 scan patterns, the diagnosis resolution is almost saturated. Applying 64 scan patterns with 32X EDT for Ckt2 can improve the chain diagnosis

resolution about 13 times with only 25% of the number of collected failure cycles compared to applying 8 patterns on Ckt2 without EDT. The chain diagnosis run times for EDT designs are now close to the ones without EDT.

Note that:

(1) The experiments in Sections 4.1 and 4.2 use stuck-at-1 fault model as examples. Similar results are expected for other types of fault such as stuck-at-0 and timing-related scan chain fault.

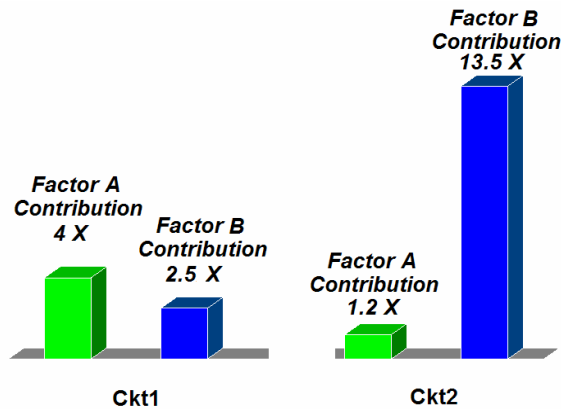
(2) All experiments in Section 4 use EDT. The proposed chain diagnosis methodology and experiments can also be applied to any other type of embedded compression techniques.

(3) We have illustrated with sampled test cases that chain diagnosis with EDT can achieve better diagnosis resolution. This would probably be also true for any type of compression techniques other than EDT. This is due to:

Factor A: with embedded compressions, the internal scan chains become much shorter. A defect on a shorter chain will have smaller impact on the scan patterns applied, which makes chain diagnosis easier.

Factor B: for  $NX$  compaction configuration, the compacted test responses from the channel output contain the information from  $N$  times of failed scan patterns without increasing the number of collected failure cycles, compared to non-compaction version.

To further analyze which factor has larger impact on diagnosis results, let's look at Figure 5.



**Figure 5: Break Down of Contributions of Different Factors**

The contribution ratios of Factor A are calculated from Columns 3 and 5 in Tables 4 and 5 for Ckt1 and Ckt2 respectively. The contribution ratios of Factor B are calculated from Columns 3 and 5 in Tables 4 and 5 for Ckt1 and Ckt2 respectively. We only consider permanent stuck-at-1 fault (row 4 in Tables 4 and 5) for this comparison.

From Figure 5, it shows that for Ckt1, Factor A has more contribution for the enhanced diagnosis resolution than Factor B, whereas for Ckt2, Factor B has much larger contribution for the enhanced diagnosis resolution.

This observation is probably due to the fact that Ckt1 has very short chains (only a few scan cells per chain) after incorporating EDT. Therefore Factor A leads to larger contribution to the enhanced diagnosis resolution. In contrast, Ckt2 still has long chains (>1K cells) even after incorporating EDT, which probably makes the Factor A has smaller contribution to the enhanced diagnosis resolution. In general, for large industrial designs, we would expect Factor B plays a more important role to enhance chain diagnosis resolution.

## 5. Multiple Faulty Chain Diagnosis

In the previous sections, we illustrated the single faulty chain diagnosis in the context of embedded compression techniques. A methodology with three steps was described in Sections 3. For multiple faulty chain diagnosis, we need distinguish two scenarios.

The first scenario is that there is only one faulty chain per channel output. Although there are multiple defective scan chains, the faulty chains are distributed at different scan channels such that their faulty responses are compacted independently at different channels. Therefore we can perform diagnosis one faulty chain at a time. The previous published multiple faulty chain diagnosis [GUO01] [HUA03] can be adopted in the context of compression techniques for this scenario.

The second scenario is that there are multiple faulty chains per channel output. In this case, Steps (1) and (2) of the proposed methodology are still the same as before. As mentioned earlier, Step (2) requires masking patterns in the chain pattern set such that each internal chain is observed uniquely from one channel output with one masking chain test pattern. Therefore, even it has multiple faulty chains per channel, it is straightforward to identify the faulty chains and the corresponding fault types by applying the masking chain patterns.

After we know the faulty chains and fault type for each faulty chain, the Step (3) of the proposed diagnosis algorithm can be implemented by two approaches:

(1) If masking scan patterns are applied and each failed scan chain is uniquely observed by at least one failed masking scan pattern (Ref. Figure 3), the diagnosis algorithm can be applied by using these masking scan patterns. In this case, for each failed masking scan pattern, only one defective chain is observed and diagnosed. Hence the multiple faulty chain diagnosis can be accomplished one chain at a time.

(2) Alternatively, if there are no appropriate masking scan patterns available, it is required to consider injecting faults on multiple faulty chains simultaneously to compute the compacted test responses. The simulation and suspect searching procedure is no different than diagnosing single chain except that the search space is expanded from one dimension for single chain to  $M$  dimensions, where  $M$  is the number of faulty chains per channel. The simulation time could be exponentially increased with the increase of  $M$ . In our future work, we need investigate new technology and algorithm in this step in order to contain the cost of searching by reducing the run time for simulation.

## 6. Conclusions

In this paper, we provide a general methodology that can be applied for performing scan chain diagnosis in the context of any compression schemes with any chain diagnosis algorithms, no matter it is hardware-based or software-based. The proposed diagnosis algorithm works on the “transformed” view of the circuit. Suspects are reported on the internal scan chains. The proposed methodology enables seamless reuse of the existing scan chain diagnosis infrastructure with compressed test data.

Experiments are performed by applying the proposed chain diagnosis algorithm on the simulated test cases using EDT technology. We show that chain diagnosis resolution can be enhanced up to one order of magnitude with only 25% of collected failure cycles compared to the diagnosis on the same design without EDT. Therefore using embedded compression techniques and the proposed chain diagnosis methodology can save cost and time for chain diagnosis and provide more accurate and meaningful information for failure analysis and yield learning.

The enhanced chain diagnosis resolution is caused by two factors: (A) The internal scan chains with embedded compressions are shorter. Hence a defect on a shorter chain will have smaller impact on the scan patterns applied. (B) For  $NX$  compaction configuration, the compacted test responses from the channel output contain the information from  $N$  times of failed patterns without increasing the number of the collected failure cycles. Due to these advantages, chain diagnosis for designs with embedded compressions can generally achieve better results than non-compression designs.

## REFERENCES

[BAR02] C. Barnhart, V. Brunkhorst, F. Distler, O. Farnsworth, A. Ferko, B. Keller, D. Scott, B. Koenemann, T. Onodera, “Extending OPMISR Beyond 10X Scan Test Efficiency”, IEEE

- Design and Test of Computers, Vol. 19, Issue 5, pp. 65-73, Sep.-Oct. 2002.
- [CHE04] W.-T. Cheng, K.-H. Tsai, Y. Huang, N. Tamarapalli and J. Rajski, “Compactor Independent Direct Diagnosis,” Proc. Asian Test Symp. 2004, pp. 204-209.
- [EDI95] S. Edirisooriya and G. Edirisooriya, “Diagnosis of Scan Failures,” Proc. VLSI Test Symposium 1995, pp. 250-255.
- [GUO01] R. Guo and S. Venkataraman, “A Technique for Fault Diagnosis of Defects in Scan Chains,” Proc. Int’l Test Conference, 2001, pp. 268-277.
- [HUA03] Y. Huang, W.-T. Cheng, S.M. Reddy, C.-J. Hsieh, Y.-T. Hung, “Statistical Diagnosis for Intermittent Scan Chain Hold-Time Fault,” Int’l Test Conference, 2003, pp.319-328.
- [HUA04] Y. Huang, W.-T. Cheng, C.-J. Hsieh, H.-Y. Tseng, A. Huang, Y.-T. Hung, “Intermittent Scan Chain Fault Diagnosis Based on Signal Probability Analysis,” DATE. 2004, pp.1072-1077.
- [KUN94] S. Kundu, “Diagnosing Scan Chain Faults,” IEEE Trans. On VLSI Systems, Vol. 2, No. 4, 1994, pp.512-516.
- [NAR97] S. Narayanan and A. Das, “An Efficient Scheme to Diagnose Scan Chains,” Proc. Int’l Test Conference, 1997, pp. 704-713.
- [MIT02] S. Mitra and K.S. Kim, “X-Compact An Efficient Response Compaction Technique for Test Cost Reduction,” Proc. ITC, pp.311-320, 2002.
- [RAJ02] J. Rajski, J. Tyszer, M. Kassab, N. Mukherjee, R. Thompson, H. Tsai, A. Hertwig, N. Tamarapalli, G. Eide, and J. Qian, “Embedded Deterministic Test for Low Cost Manufacturing Test,” Proc. ITC, pp.301-310, 2002.
- [RAJ03] J. Rajski, J. Tyszer, M. Kassab, N. Mukherjee, “Method and apparatus for selectively compacting test responses”, Patents 6,829,740 and 6,557,129.
- [SCH92] J. Schafer, F. Policastri and R. McNulty, “Partner SRLs for Improved Shift Register Diagnostics,” Proc. VLSI Test Symposium, 1992, pp.198-201.
- [STA01] K. Stanley, “High-Accuracy Flush-and-Scan Software Diagnostic,” IEEE Design & Test of Computers, Nov-Dec, 2001, pp.56-62.
- [WOH03] P. Wohl, J. Waicukauski, S. Patel, and M.B. Amin, “X-tolerant Compression and Application of Scan-ATPG Patterns in a BIST Architecture”, Proc. Int’l Test Conf., pp. 727-736, 2003.
- [WU98] Y. Wu, “Diagnosis of Scan Chain Failures,” Proc. Int’l Symp. on Defect and Fault Tolerance in VLSI Systems, 1998, pp.217-222.