Gate Level Fault Diagnosis in Scan-Based BIST

Ismet Bayraktaroglu
Computer Science & Engineering Department
University of California, San Diego
La Jolla, CA 92093
ibayrakt@cs.ucsd.edu

Alex Orailoglu
Computer Science & Engineering Department
University of California, San Diego
La Jolla, CA 92093
alex@cs.ucsd.edu

Abstract

A gate level, automated fault diagnosis scheme is proposed for scan-based BIST designs. The proposed scheme utilizes both fault capturing scan chain information and failing test vector information and enables location identification of single stuck-at faults to a neighborhood of a few gates through set operations on small pass/fail dictionaries. The proposed scheme is applicable to multiple stuck-at faults and bridging faults as well. The practical applicability of the suggested ideas is confirmed through numerous experimental runs on all three fault models.

1. Introduction

Scan-based test methodologies provide an efficient way of test generation by reducing its complexity. Furthermore, cores with internal scan chains readily enable utilization of simple test reuse schemes. Scan-based designs not only confer test benefits but additionally provide simple means for fault diagnosis by enabling access to any point in the design. While scan-based designs provide sizable benefits, such benefits are offered at the expense of high test application time and data storage requirements. Furthermore, due to the limited speed of testers, test application is usually performed at lower clock speeds, resulting in low coverage of timing related faults.

Built-in self-test (BIST) provides a low-cost solution to both test generation and test application and can result in high fault coverage with DfT techniques. The benefits of both scan and BIST can be combined in a scheme denoted scan-based BIST [7], wherein the patterns are generated on chip, shifted through the scan chains, and compacted again on chip to generate a single test signature, resulting in a test application scheme with virtually no data volume, yet enabling at-speed test and reduced test application time. The downside in this scheme, however, is the loss of diagnostic capabilities of regular scan-based designs. A compacted test signature together with limited communication with the ATE obliterates diagnostic information almost completely.

Even though BIST drastically diminishes diagnostic capabilities of scan designs, it has been nonetheless widely utilized over the last decade. The problems related to diagnosis can easily be circumvented in the initial debugging stages of the design by bypassing the signature compaction stage, and directly observing all the scan chain outputs. However, in a manufacturing test environment, doing so will completely eliminate the advantages of BIST. Researchers, consequently, have shown interest in developing schemes to increase the information content attained during a BIST session without necessitating direct access to the outputs of the scan chains.

Information content can be increased in various ways, such as through utilization of configurable [10] and longer [9] signature registers or through repeated test applications [8, 2, 3]. Previous research has mostly focused on finding the failing scan cells [8, 2, 3, 10] or failing test vectors [9, 4]. Even though failing scan chain identification has been successfully performed, identification of failing test vectors has had no practical solution so far due to the high number of failing test vectors. As the latter problem is exceedingly difficult, identification of a small set of failing test vectors instead may be the only palatable option for locating faults within small neigbourhoods.

In this work, we propose a methodology that is capable of locating stuck-at faults within small neighborhoods through utilization of both fault embedding scan cells and failing test vectors. While for identification of fault embedding scan cells any of the previously suggested schemes [8, 2, 3, 10, 4] can be utilized, for failing test vector identification we propose to acquire additional signatures during test application. Additional signatures are to be captured for a small set of initial vectors individually and for the rest of the vectors in larger groups. Utilization of a small set of initial test vectors provides a sufficient number of failing test vectors for identification of easy-to-test faults. Signatures for groups of test vectors, which do include failing test vectors, as they cover the complete test set, are utilized to provide diagnostic resolution for hard-to-detect faults.

Proceedings of the 2002 Design, Automation and Test in Europe Conference and Exhibition (DATE'02)
1530-1591/02 $17.00 © 2002 IEEE

IEEE COMPUTER SOCIETY

Test Vectors \ Scan cells	S_1	S_2	S_3	$\cdots$	S_{n-2}	S_{n-1}	S_n
T_1	$O_{1,1}$	$O_{1,2}$	$O_{1,3}$	$\cdots$	$O_{1,n-2}$	$O_{1,n-1}$	$O_{1,n}$
T_2	$O_{2,1}$	$O_{2,2}$	$O_{2,3}$	$\cdots$	$O_{2,n-2}$	$O_{2,n-1}$	$O_{2,n}$
T_3	$O_{2,1}$	$O_{2,2}$	$O_{2,3}$	$\cdots$	$O_{2,n-2}$	$O_{2,n-1}$	$O_{2,n}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$	$\vdots$
T_{t-2}	$O_{t-2,1}$	$O_{t-2,2}$	$O_{t-2,3}$	$\cdots$	$O_{t-2,n-2}$	$O_{t-2,n-1}$	$O_{t-2,n}$
T_{t-1}	$O_{t-1,1}$	$O_{t-1,2}$	$O_{t-1,3}$	$\cdots$	$O_{t-1,n-2}$	$O_{t-1,n-1}$	$O_{t-1,n}$
T_t	$O_{t,1}$	$O_{t,2}$	$O_{t,3}$	$\cdots$	$O_{t,n-2}$	$O_{t,n-1}$	$O_{t,n}$

Figure 1. Test Response

The diagnostic procedures outlined in this work are based on small pass/fail stuck-at fault dictionaries for both failing test vectors and fault embedding scan cells. The diagnostic procedure is composed of simple set operations and in case of multiple stuck-at and bridging faults the procedure is followed up by a pruning step to improve diagnostic resolution levels.

Section 2 reviews the process of both fault embedding scan cell and test vector identification schemes in the literature and shows the difficulty of failing test vector identification from an information theoretic perspective. Section 3 proceeds with outlining the failing test vector identification scheme utilized in this work. Section 4 outlines the diagnostic procedures proposed in this work for three fault models. Experimental results on a set of benchmark circuits in section 5 are followed by conclusions in section 6.

2. Background

Research efforts on diagnosis of scan-based BIST designs have so far concentrated mostly on a reconstruction of the output responses of a circuit from a set of signatures. The output response of a scan-based circuit can be visualized as a two dimensional matrix as shown in figure 1. In this representation, rows correspond to the complete output sequence of each test vector, while columns correspond to the output responses of each scan cell in the scan chain. Previous research in this area has mostly concentrated on identification of either fault capturing scan cells, i.e., the columns with incorrect output responses, or fault detecting test vectors, i.e., the rows with incorrect output responses.

Identification of fault embedding scan cells has been widely analyzed and cost-effective solutions with reasonable tester times have been proposed in the literature [10, 8, 2, 3]. Even though identification of fault detecting test vectors has been similarly extensively analyzed [9, 4, 1], no practical techniques with reasonable accuracy have been proposed so far. While the two problems seem identical from a theoretical perspective, the relatively much larger number of test vectors typically imposes significant difficulties on failing test vector identification. For example, when half of the test vectors exhibit a failure in a scan chain, a not uncommon occurrence in the case of an easy-to-detect fault, the representation of all possible failure combinations would require $\log_2 \binom{N}{N/2}$ bits. Using Stirling's formula[1], the number of bits required can be shown to be approximately $N - 0.33 - \frac{1}{2}\log_2 N$; for N equal to 50, this expression computes to 46.85 bits, constituting a lower bound on the number of bits required, assuming perfect encoding of the failure combinations. Evidently, we would be better off scanning out all scan cell contents for identification of failing test vectors instead.

A quick look at the results of previous research confirms our observations. Savir *et al* [9] have proposed a diagnosis scheme for the case when only a couple of failing test vectors exists in the test set. In case of higher number of failing test vectors, aliasing problems with the proposed cyclic registers introduce a high number of non-failing vectors to the failing test vectors returned by the proposed scheme. Ghosh-Dastidar *et al* [4] have attempted to improve through a pruning algorithm upon the results of [9] in case of high number of failing test vectors. The proposed scheme effectively prunes the failing and non-failing vectors almost equally from the initial candidate set, resulting in no change in the ambiguity levels. Even more surprisingly, random selection of a set of test vectors as failing, in case of a high number of failing vectors, provides similar levels of ambiguity with no hardware or software overhead!

Identification of fault embedding scan cells may be utilized to point the cause of failures to a small neighborhood through cone analysis. In case of single stuck-at faults, such analysis can pinpoint the cause of failure to a handful of fault locations. However, multiple stuck-at faults, as shown later in this work, significantly deteriorate the level of achievable diagnostic resolution. Consequently, utilization of failing test vector information plays a more important role in the case of multiple stuck-at faults.

3 Failing Test Vector Identification

Subsequent to identification of the fault embedding scan cells, identification of failing test vectors can be performed either for each fault embedding scan cell individually [4], necessitating as many applications of the failing test vector identification procedure as the number of fault embedding scan cells, or for the whole scan chain only once. Successful identification of failing test vectors for fault embedding scan cells individually necessitates scanning out the contents of the complete scan chain, thus completely eliminating the benefits of BIST. Identification of failing test vectors, on the other hand, can be performed at significantly reduced cost by solely scanning out the signature for each test pattern. We consequently propose to identify failing test vectors for the whole scan chain but not for each individual fault embedding scan cell.

[1] $n! \approx \sqrt{2\pi n}\, n^n e^{-n}$

Proceedings of the 2002 Design, Automation and Test in Europe Conference and Exhibition (DATE'02)
1530-1591/02 $17.00 © 2002 IEEE

IEEE COMPUTER SOCIETY

While identification of failing test vectors for fault embedding scan cells individually enables reconstruction of the output sequences, $O_{t,n}$, which could be utilized with a full fault dictionary, the proposed approach can only be utilized with a pass/fail fault dictionary. Even though the diagnostic resolution of pass/fail dictionaries is lower than that of full dictionaries, they can provide comparable diagnostic resolution levels when they are coupled with cone analysis. Cone analysis is performed through utilization of fault embedding scan cells as mentioned in the previous section.

Once we have the signature for a test vector, determination of whether that vector is failing is a simple matter of comparison of the test signature with the predetermined correct signature. Yet scanning out the MISR and performing the comparison for every pattern imposes significant test time overhead. This exceedingly high overhead turns out to be largely unnecessary, as it can be questioned whether identifying all failing vectors, in the case of easy-to-detect faults, provides any appreciable incremental diagnostic information. Furthermore, capturing a small number of failing vectors capable of providing the bulk of diagnostic information in this case can be achieved in a rather small number of individually scanned signatures. For example, for scanned versions of the ISCAS89 benchmark circuits, within the first 20 test vectors, over 65% of the faults have at least 1 failing vector, while over 44% of the faults have at least 3 failing vectors, indicating that selection of the first 20 vectors for scanning out the signatures delivers sizable diagnostic information for most cases. In the manufacturing test environment, the tester needs to collect the signatures for a small number of vectors for subsequent off-line analysis.

While identification of failing vectors is a simple matter for easy-to-detect faults, the same identification problem for hard-to-detect faults is not as simple. Nonetheless, a set of test vectors that includes a failing test vector for a hard-to-detect fault can be identified by partitioning the complete test set into disjoint test vector groups. As each test vector is embedded in one of these groups, it is guaranteed that at least one of these test groups includes a failing test vector. Even though the exact failing test vector information would not immediately be known, the underlying group information constitutes sufficiently fine grounds for diagnosis of hard-to-detect faults.

4 Fault Location through Set Operations

In this section, we propose a diagnosis scheme primarily comprised of set operations on pass/fail dictionaries. The fault dictionaries utilized in this work are based on fault embedding scan cell and failing test vector information. The first two subsections discuss the diagnostic procedure for single stuck-at faults. The final subsection proceeds to discuss the modifications necessitated by multiple stuck-at and bridging fault models.

4.1 Fault Embedding Scan Cell Utilization

Failing scan cell information can help locate the cause of the failures. Through fault simulations, faults detected at each scan cell output can be determined. This information is employed to generate pass/fail dictionaries. If a fault is detected at the output of scan cell *i*, the fault is included in the list of faults detected by that scan cell. We denote the set of faults detectable by the test set at scan cell *i* as F_{s_i}. Let's assume that after an application of one of the fault embedding scan cell determination schemes, we have identified the failing scan cells. Under the single stuck-at fault assumption, a set of candidate faults, C_s, can, through utilization of fault embedding scan cell information, be identified by the following equation:

$$C_s = \left(\bigcap_{i\ failing} F_{s_i} \right) - \left(\bigcup_{i\ non-failing} F_{s_i} \right) \qquad (1)$$

As the single stuck-at fault has to assume responsibility for all the failures, it has to lie in the intersection of all the fault sets that are detected by the fault embedding scan cells. Conversely, the fault-free status of the scan cell *i* provides clear evidence that none of the faults detectable by it, F_{s_i}, can be the cause of the failure. Consequently, the union of the set of faults that are detectable by the non-failing scan cells needs to be subtracted from the candidate failure list. If the single stuck-at fault assumption holds, C_s is guaranteed to include the culprit fault that causes the observed behavior at the scan cell outputs.

4.2 Failing Test Vector Utilization

An analogous analysis can be performed for failing test vectors. The set of faults detected by a group of test vectors can be stored in a pass/fail dictionary. We denote the set of faults detected by a group of test vectors *i* as F_{t_i}. The individual test vectors utilized in this work can simply be represented by a test vector group of size one. The set of candidate faults can be determined by the following equations under the single stuck-at fault assumption.

$$C_t = \left(\bigcap_{i\ failing} F_{t_i} \right) - \left(\bigcup_{i\ non-failing} F_{t_i} \right) \qquad (2)$$

Since both of the two distinct sets of candidate faults generated so far include the culprit fault, the intersection of these two sets provides the final, smaller candidate set, C.

$$C = C_s \cap C_t \qquad (3)$$

The analysis outlined so far holds under the single stuck-at fault assumption. The following subsection discusses modifications necessitated to accommodate more general fault model assumptions, namely, multiple stuck-at and bridging fault models.

Proceedings of the 2002 Design, Automation and Test in Europe Conference and Exhibition (DATE'02)
1530-1591/02 $17.00 © 2002 IEEE

IEEE COMPUTER SOCIETY

4.3 Multiple Stuck-at Faults

Under the multiple stuck-at fault model, there can exist more than one cause for the observed failures. A simple intersection operation, consequently, cannot be utilized for candidate fault list generation, as each failure information can possibly be accounted for through a distinct fault! Instead, the union of the faults detected by the fault embedding scan cells needs to be assumed to be the cause of the failures, in the process resulting in a much larger candidate list. Every fault, nonetheless, that was detectable at non-failing scan cells continues to preserve its innocence, enabling the continued subtraction of all these faults from the candidate list, thus providing appreciable reductions in the otherwise highly enlarged candidate fault list. The list of candidate faults, C_s, in case fault embedding scan cell information is utilized, for multiple stuck-at faults becomes:

$$C_s = \left(\bigcup_{i\ failing} F_{s_i} \right) - \left(\bigcup_{i\ non-failing} F_{s_i} \right) \quad (4)$$

While the observation outlined in the derivation of equation 4 seems valid at first sight, the interaction between multiple faults may nonetheless invalidate this result. A scan cell output may display a failing (non-failing) result due to the interactions among multiple faults, while it would have displayed a non-failing (failing) result for all (at least one) of the faults, had the faults occurred individually. While an additional failure can only increase the size of the candidate list, thus slightly reducing diagnostic resolution, a missing failure can eliminate real faults from the candidate list, thus reducing diagnostic coverage. Even though the interaction among multiple stuck-at faults is of theoretical significance, we show in section 5 that one of the culprit faults is almost always included in the final candidate fault list nonetheless. Therefore, equation 4 can be utilized with no modifications, such theoretical grounds notwithstanding. Removal of the second term of equation 4 suffices to guarantee inclusion of all the culprit faults in C_s, though at the expense of significantly reduced diagnostic resolution.

In case of failing test vector utilization, the intersection operator in equation 2, due to the same reasoning given for the fault embedding scan cells, needs to be replaced with the union operator, analogously resulting in:

$$C_t = \left(\bigcup_{i\ failing} F_{t_i} \right) - \left(\bigcup_{i\ non-failing} F_{t_i} \right) \quad (5)$$

As for the failing scan chain utilization, intersections between multiple stuck-at faults may also invalidate this result. In case we want to guarantee inclusion of all the culprit faults in C_t, the second term of equation 5 needs to be removed.

Even though the diagnostic resolution of the proposed scheme cannot further be improved under a generic multiple fault assumption, under a restricted form of the multiple fault assumption, wherein a bound is placed on the maximum number of fault occurrences, further pruning of the candidate fault set is possible. If the maximum number of faults is limited to three for example, a fault which cannot account for all the failures in conjunction with any other two faults can be dropped from the fault list. The fault x can be dropped from the candidate fault list, if the following condition holds:

$$x_f \cup (y_f \cup z_f) \neq Failure \ \ \forall y,z \quad (6)$$

wherein x_f denotes the set of failures explainable by the existence of the fault x. While the condition in equation 6 effectively prunes the candidate list as shown through experiments in section 5, in some cases it causes the culprit faults to drop from the candidate list. This happens whenever multiple stuck-at faults, individually undetectable by a test vector, interact to become detectable by the test vector, resulting in failures that cannot be explained by any one of the culprit faults. Even though the suggested diagnostic resolution improvement results in a slight loss of diagnostic coverage, the benefits attained in diagnostic resolution outweigh the diagnostic coverage loss.

Diagnostic resolution for multiple stuck-at faults can be further improved if the aim of the diagnostic procedure is limited to identification of only one of the faults in the system. Inclusion of only one of the failing test vector groups in equation 5 results in inclusion of at least one of the culprit faults in C_t. As all culprit faults are included in C_s, the intersection of the two sets would still include at least one of the culprit faults in the final candidate list.

4.4 Bridging Faults

In case of AND or OR type bridging faults, observation of a bridging fault necessitates that a stuck-at fault at one of the nodes involved in the bridge be detectable and the other node be at the opposite structural value. This indicates that the stuck-at fault involved in the bridge will not be observable half of the times that it would have been detected, were it a single stuck-at fault. Consequently, utilization of the second term in equations 4 and 5 will most likely eliminate the bridging faults at the nodes involved in the bridge from the candidate list. Consequently, the following equation can be used for generating the candidate fault list.

$$C = \left(\bigcup_{i\ failing} F_{s_i} \right) \cup \left(\bigcup_{i\ failing} F_{t_i} \right) \quad (7)$$

In case of bridging faults, due to the elimination of the difference term from equations 4 and 5, the diagnostic resolution gets degraded significantly. However, a single bridging fault means that there are only 2 faults in the system. Therefore, the proposed pruning scheme for multiple stuck-at faults, outlined in equation 6, can also be utilized for the single bridging fault model with significant diagnostic resolution improvements. Furthermore, in case of AND or OR

Proceedings of the 2002 Design, Automation and Test in Europe Conference and Exhibition (DATE'02)
1530-1591/02 $17.00 © 2002 IEEE

IEEE COMPUTER SOCIETY

type of bridging faults, if we ignore faults that result in sequential or oscillatory behavior, only one of the faults involved in a bridging fault can be detected by a test vector at a time. This observation suggests that the two faults that are involved in the bridging fault cover all failing test vectors in a mutually exclusive way. Inclusion of this *mutual exclusion property* in the pruning algorithm further improves diagnostic resolution levels for bridging faults.

5. Results

In order to verify the performance of the proposed diagnostic procedure in this work, a set of scanned versions of ISCAS89 benchmark circuits have been utilized. In order to keep the experiments in a uniform framework, for each of these circuits, a set of 1,000 patterns are employed. The patterns include both deterministic patterns generated by Atalanta [5] and additional random patterns. The pattern set is later shuffled to eliminate any bias introduced due to deterministic patterns. Fault simulations and fault dictionary generation are conducted in this work with HOPE [6].

Table 1 provides a list of the circuits employed in this work. The total number of primary outputs, including the scan cell outputs, and the number of faults are also provided. While for smaller circuits, all faults are included in the diagnostic experiments, for larger circuits, a set of randomly selected 1,000 faults is utilized. The fourth column in the table indicates the number of fault equivalence groups under the given test set. The final three columns provide the number of equivalence classes attained when pass/fail dictionaries of the first 20 test vectors, 20 test groups of size 50, and cone information are utilized, respectively. While for easily testable benchmark circuits, such as s35932, individual test vectors provide a higher diagnostic resolution (larger number of equivalence classes), for hard-to-test circuits, such as s832, test vector groups instead provide a higher diagnostic resolution.

			Equivalence Groups			
Circuit	Outputs	Faults	Full Res.	20 Ps	20 TGs	Cone
s298	17	308	279	145	90	41
s344	26	342	337	133	55	52
s386	13	384	384	41	228	30
s444	27	460	377	138	96	66
s641	43	467	459	194	130	157
s832	24	856	814	97	472	50
s953	52	1079	1074	257	467	238
s1423	79	1501	1349	647	316	356
s5378	228	4563	4163	1329	1146	1054
s9234	250	6475	5292	1254	1811	1020
s13207	790	1000	977	466	281	608
s15850	684	1000	976	494	316	660
s35932	2048	1000	854	682	50	440
s38417	1742	1000	987	509	262	627

Table 1. Circuit parameters and number of equivalence groups for various dictionaries

Evaluation of the effectiveness of the proposed scheme necessitates a metric for diagnostic resolution. Diagnostic resolution can be measured as the number of faults in the candidate list averaged over all the faults in the circuit. However, such an approach underestimates the diagnostic resolution levels. For a given test set, the faults in a circuit can be grouped into equivalence groups as some of the faults in the circuit provide identical outputs for all the test vectors in the test set and can by no means be distinguished with this test set. A more realistic diagnostic resolution measure in this case can be given as the number of fault equivalence groups in the candidate list averaged over all faults. A diagnostic resolution of 1 would be the best achievable diagnostic resolution; the higher the number, the lower the diagnostic resolution.

Table 2a summarizes the results of diagnostic experiments under the single stuck-at fault model. In all the experiments performed, the culprit faults are invariably included in the final candidate sets, providing consistently 100% diagnostic coverage. The table provides experimental results for five different cases. The first two column pairs, for no use of fault embedding scan cell information and for no use of group information, provide the average diagnostic resolution (denoted as *Res*) and maximum cardinality of the candidate fault set (denoted as *Mx*). The results indicate that both the fault embedding scan cell information and test vector groups are important in providing high diagnostic resolution levels.

We have furthermore conducted experiments for multiple stuck-at faults. For each of the benchmark circuits, randomly selected 1,000 pairs of stuck-at faults are injected in the circuits and diagnostic resolution is computed. The diagnostic resolution for the basic scheme, with pruning, and with single fault targetting are summarized in table 2a. The columns denoted as *One*, *Both*, and *Res* correspond to the percentage of the cases where at least one of the faults is diagnosed, the percentage of the cases where both faults are diagnosed, and the average diagnostic resolution, respectively. In a small number of cases, the candidate list does not include any of the culprit faults due to the interactions between multiple faults as discussed in section 4.3. Under a restricted fault model, i.e., maximum two stuck-at faults, the additional pruning step discussed in section 4.3 improves diagnostic resolution. In case only one of the culprit faults is targeted, diagnostic resolution is further improved, indicating that relaxation of the diagnostic objectives improves the diagnostic resolution levels.

Additional diagnostic experiments are performed for bridging faults. In these experiments, an AND bridging fault model is assumed. As in the case of multiple stuck at faults, 1,000 single bridging faults are randomly injected to the benchmark circuits. The results summarized in table 2c indicate that diagnosis of bridging faults is more difficult compared to multiple stuck-at faults. Nevertheless, the improvement applicable in the case of multiple stuck-at faults is also

Proceedings of the 2002 Design, Automation and Test in Europe Conference and Exhibition (DATE'02)
1530-1591/02 $17.00 © 2002 IEEE

IEEE COMPUTER SOCIETY

Circuit	No Cone		No Group		All	
	Res	Mx	Res	Mx	Res	Mx
s298	2.11	11	2.17	10	1.23	4
s344	3.09	12	1.70	6	1.17	3
s386	2.41	11	13.28	38	1.15	5
s444	2.53	9	2.06	9	1.16	3
s641	3.12	22	4.48	37	1.08	2
s832	2.52	14	44.09	107	1.11	3
s953	3.11	28	6.16	32	1.06	3
s1423	2.21	19	2.30	24	1.13	3
s5378	4.67	64	11.52	181	1.17	6
s9234	5.74	52	33.81	249	1.40	9
s13207	2.24	19	2.75	20	1.05	3
s15850	1.67	8	1.45	9	1.02	2
s35932	1.70	15	1.03	2	1.03	2
s38417	3.05	26	4.73	30	1.15	5

a) Single stuck-at faults

Circuit	Basis Scheme			With Pruning			Single Fault		
	One	Both	Res	One	Both	Res	One	Both	Res
s298	100.0	97.5	27.5	98.7	97.2	8.1	99.1	8.1	2.4
s344	100.0	98.9	43.0	99.4	98.5	13.0	99.7	10.8	4.0
s386	100.0	97.9	67.8	98.5	96.3	41.5	99.1	11.1	7.0
s444	100.0	99.0	32.9	97.7	91.0	10.3	96.9	12.9	4.2
s641	100.0	98.8	53.9	99.1	98.4	13.6	99.1	9.3	3.0
s832	100.0	99.6	95.4	99.6	99.1	50.2	99.9	9.7	5.5
s953	100.0	99.2	120.2	99.5	98.5	35.0	99.4	8.5	5.0
s1423	100.0	99.6	53.8	99.7	99.5	8.3	99.8	10.0	3.0
s5378	99.7	99.1	119.8	99.3	98.8	21.8	99.7	11.1	3.6
s9234	100.0	99.6	194.4	98.3	95.0	50.3	97.4	5.8	4.9
s13207	100.0	99.9	22.2	95.0	94.0	4.5	96.8	8.6	1.2
s15850	100.0	99.6	21.1	95.5	94.5	3.5	97.4	10.0	1.3
s35932	99.6	98.7	8.9	86.5	82.3	2.6	94.3	14.3	2.0
s38417	100.0	99.5	36.3	97.7	97.4	9.3	98.6	9.7	2.0

b) Multiple stuck-at faults

Circuit	Basic Scheme		With Pruning		Single Fault	
	Both	Res	Both	Res	Both	Res
s298	90.8	91.9	90.8	48.8	2.1	20.2
s344	92.3	135.4	92.3	82.7	3.0	35.3
s386	79.8	153.3	79.8	150.7	17.2	64.1
s444	93.0	118.9	93.0	88.7	8.7	40.7
s641	96.2	230.3	96.2	116.2	3.9	52.5
s832	64.7	181.2	64.7	173.2	8.8	63.4
s953	73.3	420.9	73.3	354.3	9.4	151.3
s1423	95.4	445.9	95.4	295.5	2.8	106.1
s5378	85.9	64.5	85.9	63.6	3.6	42.1
s9234	82.8	363.6	82.8	280.8	13.0	221.8
s13207	87.4	102.8	87.4	33.2	3.3	11.2
s15850	90.8	159.3	90.8	67.7	3.0	29.4
s35932	99.1	54.5	99.1	9.2	0.1	12.6
s38417	87.8	83.8	87.8	35.7	2.3	12.3

c) Bridging faults

Table 2. Diagnostic resolution for various fault models

applicable in this case. Elimination of the faults that cannot explain the failures with any of the faults in the candidate set and utilization of the *mutual exclusion property* in failure explanation improves the diagnostic resolution.

Finally, diagnostic experiments are repeated with the aim of identification of only one of the faults involved in the bridge. The results provided under the last column pair indicate that diagnostic resolution improves significantly at the cost of losing one of the sites involved in the bridging fault. However, as the two sites of a bridging fault are electrically shorted, identification of a single site provides sufficient diagnostic specificity, thus alleviating the complexity of subsequent surface scan procedures.

6. Conclusion

In this work, an automated diagnosis scheme based on small pass/fail dictionaries for scan-based BIST designs is proposed. Diagnosis is performed off-line after test application through utilization of failing scan cell information and a set of test signatures that are attained during test application. Utilization of test vector groups in addition to the individual test vectors to provide failing test vectors helps improve diagnostic resolution significantly. Furthermore, utilization of failing scan cell information improves diagnostic resolution considerably as well.

The proposed methodology provides a rapid, automated diagnosis of various faults. Especially, in the case of single stuck-at faults, the proposed scheme provides very high diagnostic resolution levels with no diagnostic coverage loss. While multiple stuck-at faults and bridging faults have been traditionally understood to degrade the performance of diagnosis schemes, the pruning methods proposed in this work significantly improve the initial diagnostic resolution levels even for such faults.

The proposed scheme can be utilized in a manufacturing test environment to provide a fast diagnostic procedure to pinpoint correlated failures in the design. In case further diagnostic resolution is required, the proposed scheme provides an excellent starting point for subsequent debugging procedures, thus helping greatly reduce their cost.

References

[1] R. C. Aitken and V. K. Agarwal. A diagnosis method using pseudo-random vectors without intermediate signatures. In *International Conference on Computer-Aided Design*, pages 574–580, 1989.

[2] I. Bayraktaroglu and A. Orailoglu. Improved fault diagnosis in scan-based BIST via superposition. In *IEEE Design Automation Conference*, pages 55–58, June 2000.

[3] I. Bayraktaroglu and A. Orailoglu. Diagnosis for scan-based BIST: Reaching deep into the signatures. In *IEEE Design Automation and Test in Europe Conference*, pages 102–109, March 2001.

[4] J. Ghosh-Dastidar, D. Das, and N. A. Touba. Fault diagnosis in scan-based BIST using both time and space information. In *International Test Conference*, pages 95–102, 1999.

[5] H. K. Lee and D. S. Ha. On the generation of test patterns for combinational circuits. Technical Report 12_93, Department of Electrical Eng., Virginia Polytechnic Institute and State University.

[6] H. K. Lee and D. S. Ha. HOPE: An efficient parallel fault simulator. In *IEEE Design Automation Conference*, pages 336–340, June 1992.

[7] B. Nadeau-Dostie, D. Burek, and A. S. M. Hassan. ScanBist: a multifrequency scan-based BIST method. *IEEE Design & Test of Computers*, 11(1):7–17, Spring 1994.

[8] J. Rajski and J. Tyszer. Diagnosis of scan cells in BIST environment. *IEEE Transactions on Computers*, 48(7):724–731, July 1999.

[9] J. Savir and W. H. McAnney. Identification of failing tests with cycling registers. In *International Test Conference*, pages 322–328, 1988.

[10] Y. Wu and S. M. I. Adham. Scan-based BIST fault diagnosis. *IEEE Transactions on Computer-Aided Design*, 18(2):203–211, February 1999.

1530-1591/02 $17.00 © 2002 **IEEE**

IEEE COMPUTER SOCIETY