ISSN (ONLINE): 2454-9762
ISSN (PRINT): 2454-9762
Available online at www.ijarmate.com

International Journal of Advanced Research in Management, Architecture, Technology and Engineering (IJARMATE)
Vol. 2, Issue 3, March 2016

Diagnosis of Multiple Scan Chain Failures Using Tester Architecture

Gowri.T[1], Kavitha.K[2]

PG Scholar, Department of ECE, Vandayar Engineering College, Thanjavur, India [1]

Assistant Professor, Department of ECE, Vandayar Engineering College, Thanjavur, India [2]

***Abstract*— Diagnosis is extremely important to ramp up the yield during the integrated circuit manufacturing process. It reduces the time to market and product cost. When multiple chains mapped to a single compactor, diagnosis becomes extremely difficult. The procedure is even more complicated because when a circuit fails the flush test, not all the patterns are applied. Only a few of the patterns are applied and the observed responses are used to diagnose the faulty chains. In this paper, an efficient masking strategy that will be very useful for diagnosis of scan chains when multiple scan chains fail. This strategy uses the redundancy in fault detection by the test patterns and masks scan chains. A new tester architecture that will select and apply only those patterns having enough information for diagnosis has also been proposed. Diagnostic resolution and first hit index achieved by our method are very close to their ideal values, which validate the applicability of our approach.**

***Index Terms*— Automatic test equipment (ATE), masking, scan chain diagnosis, test response compaction**

I. INTRODUCTION

When a circuit fails a test, diagnosis is the process of narrowing down the suspected list of possible defect locations. Fault diagnosis is extremely important to ramp up the manufacturing yield, especially for 90 nm and below technologies, where physical failure analysis is difficult due to reduced defect visibility by smaller feature sizes and larger leakage currents. Diagnosis helps to reduce the product debug time. By reducing the candidate locations down to possibly only a few, subsequent physical failure analysis becomes faster and easier when searching for the root causes of failure.

A failure can occur in a circuit due to the defects present in the logic circuit or in the scan chains [1], [2]. While many defects reside in the logic part of a chip, defects in scan chains are also quite common. Scan chains are the most important Design for Test mechanism used in today's very-large-scale integration (VLSI) industry. Thus, it is very important to test the integrity of scan chains. Scan chain failures are the cause for a substantial proportion of failing chips. As 30%–50% of logic gates of a typical chip impact the operation of scan chains [3], it is very likely that scan chain operations will be impacted by random and/or systematic defects. Chain failures often account for almost 30% of chip failures [4]. Therefore, scan chain failure diagnosis is very important for effective Scan-based testing strategies.

Scan chain diagnosis starts with a flush test application [5]. A flush pattern consists of shift-in and shift-out operations without pulsing capture clocks.

The scan cells situated between the scan chain's input and the input to a particular scan cell known as the upstream cells of that scan cell. The scan cells between a scan cell output and the scan chain's output terminal are called the downstream cells of that scan cell. Table I shows an example of identifying faulty chains and modeling chain defects by flush patterns. Suppose a scan chain with 12 scan cells is loaded with flush pattern 001100110011. The second column gives the unloaded faulty values for each type of permanent fault given in column 1. By using this table, the fault model to be used for diagnosis can be identified.

Different techniques have been proposed in the literature to diagnose failing scan cells effectively. The techniques can broadly be classified into three categories: 1) tester-based; 2) hardware-assisted; and 3) software-based methods. Each of these methods has been discussed in detail in Section II.

The manufacturing test cost of VLSI circuits using scan-based structural tests is determined by test application time and volume of tester memory requirement. Test data compression and test response compaction [1], [6] are the widely used techniques to reduce both. Test response compaction is typically done using multiple input signature register (MISR) or tree of Exclusive-OR (XOR) gates. MISRs are used for both space and time compaction, whereas XOR trees are used for space compaction alone. Fig. 1 shows a typical space compactor with three output channels where XOR based compaction has been used.

Test responses in scan chains go through the compactor and the compacted responses are observed through the output channels. Uninitialized memory cell, multicycle paths, and so on can lead to the presence of do not care (X) terms in the responses. For the compactor outputs to be definite, it is necessary that these Xs do not enter the compactors.

ISSN (ONLINE): 2454-9762
ISSN (PRINT): 2454-9762
Available online at www.ijarmate.com

International Journal of Advanced Research in Management, Architecture, Technology and Engineering (IJARMATE)
Vol. 2, Issue 3, March 2016

Otherwise, a single X in cell-*i* of scan chain *j* may corrupt outputs of all compactors taking this chain as input, for the time instant *i*. Solutions to handle Xs are listed in [8] – [10]. Normally, an AND-based masking circuitry is utilized for masking the Xs and prevents them from reaching the space compactors [11], [12]. This is achieved using scan chain selection logic block that consists of a mask register. The masking signals, determined separately for each pattern, are delivered through the test inputs to the mask register. The masking signals drive inputs to AND gates. Therefore, for the chains that have Xs, the scan selection logic sends a 0 to the corresponding AND gates, and a 1 to the rest. If for a pattern, response is observed in only one scan chain in each output channel while all the other chains are masked, it is called a 1-hot pattern. Generally flush tests are used in 1-hot pattern mode so that faulty scan chains can be identified [7].

Compaction of test responses negatively impacts fault diagnosis, due to reduced observability. The methods to improve fault diagnosis for circuits using test response compaction include: 1) bypassing the compaction circuitry; and 2) using additional tests [3], [13], [14]. Bypassing compaction requires additional on-chip circuitry and increased test data volume since no compaction is being performed. Using additional tests to improve diagnosis can be done in two ways. One is to augment production tests by including further patterns to enhance diagnosability. However, since this approach increases test application time, it is typically not used. The other approach is to use production tests first to detect defects and then to use additional tests for diagnosis purpose. However, in this case, multiple test sessions of the chip in the tester will increase the test cost. Moreover, with the use of space compactors, the number of scan chains is requirement. Test data compression and test response compaction [1], [6] are the widely used techniques to reduce both. Test response compaction is typically done using multiple input signature register (MISR) or tree of Exclusive-OR (XOR) gates. MISRs are used for both space and time compaction, whereas XOR trees are used for space compaction alone. Fig. 1 shows a typical space compactor with often much larger than that of traditional scan designs [7]. Thus, the probability of having multiple scan chain failures is even higher in modern compression-based scan design than in traditional scan design. In a recent survey [15], it has been reported that 32% of 529 units with scan chain failures contained multiple chain failures.

With this background, in this paper, we have proposed a technique to diagnose multiple chain failures in a response compaction environment. The basic problem in such an environment is that simultaneous failures in two/more scan chains feeding a compactor may cancel each other, making diagnosis difficult. AND-gate masking logic (generally used to block don't cares) at the compactor side can be used to aid in diagnosis [7]. In this paper, we have also used the masking logic. The basic motivation of the work is that if for some pattern, only one faulty chain is observed and others are masked off, the corresponding compacted response can only get affected by the non masked faulty chain. Such a test pattern will help to diagnose that chain. We call these patterns essential for that chain. If such essential patterns exist for all chains, the diagnosis problem will be benefited enormously. In this paper, the mask signals are generated in such a way to maximize the probability that there will always be some essential patterns for any combination of failed scan chains. Masking more chains is justified because of redundancy in fault detections by the test patterns. However, masking of scan chains lowers fault coverage also. An intelligent mask generation scheme has been proposed to minimize pattern count as well.

A simple way to ensure essential pattern for a chain is to mask all the remaining chains for a particular pattern. By this way, using patterns equal to the number of chain, an essential pattern for every chain can be generated. This method has the following problems: First of all, those patterns will not be a part of production test set. They are diagnostic test patterns. Second, one pattern may not be sufficient to pin point the faulty location. In experimental section, we have reported the results that prove the point. Finally, the increase in test pattern length due to addition of this pattern is significant. So, a proper formulation of the problem is necessary.

Our work has several fundamental contributions.

1) A novel modeling of the problem has been done. The masking scheme is independent of the diagnosis algorithm to be used.

2) A simple but efficient algorithm has been proposed to solve the formulated problem.

3) The procedure to diagnose scan chain is more complicated because when a circuit fails the flush test, not all the patterns are applied. Only a subset of patterns is applied and the observed responses are used to diagnose the faulty chains. Accordingly, we have proposed a new tester architecture that selectively applies only those patterns having enough information for diagnosis.

4) Unlike [7], we have considered both single as well as multiple faults at any failing chain. Multiple chains can have multiple faults simultaneously.

Rest of the paper is organized as follows: In Section II, literature on scan chain diagnosis has been reviewed. The details of the problem have been given in Section III. Section IV contains the test generation flow. In Section V, the modeling of the problem is described. The proposed

ISSN (ONLINE): 2454-9762
ISSN (PRINT): 2454-9762
Available online at www.ijarmate.com

International Journal of Advanced Research in Management, Architecture, Technology and Engineering (IJARMATE)
Vol. 2, Issue 3, March 2016

algorithm to enhance the diagnostic resolution is also given in Section V. Section VI contains the proposed tester architecture. In Section VII, the details of the multiple-fault diagnosis in multiple chains has been discussed. Experimental results are given in Section VIII. Finally, we conclude in Section IX.

TABLE I
SCAN CHAIN FAULT MODELS AND THEIR EFFECTS. (FAULT-FREE UNLOADED VALUES ARE 001100110011)

Fault Type	Unloaded values with one permanent fault
Stuck-at 0	000000000000
Stuck-at 1	111111111111
Slow-to-rise	00100010001X
Slow-to-fall	01110111011X
Fast-to-rise	X01110111011
Fast-to-fall	X00100010001

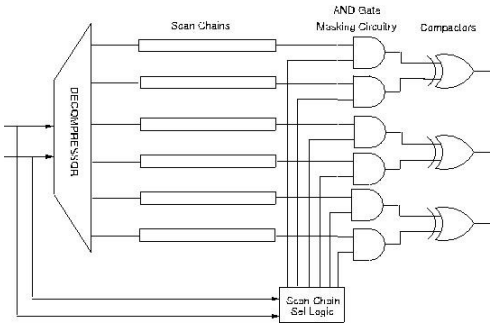

Fig. 1. Space compactor with three output channels [7].

II. PREVIOUS WORKS

Tester-based diagnosis techniques [16], [17] use tester to control scan chain shift operations and physical failure analysis equipment to observe defective responses at different locations to identify failing scan cells. These techniques normally provide good diagnosis resolution. However, they require expensive, time consuming, and often destructive sample preparation.

Hardware-assisted methods use special scan chain and scan cell designs to facilitate diagnosis. Schafer *et al.* [18] have proposed to connect each scan cell's output to a scan cell in another scan chain, so that its value could be observed from the other scan chain (partner chain) in diagnostic mode. Edirisooriya and Edirisooriya [19] have proposed to insert XOR gates between scan cells to enhance chain diagnosis. The proposed scheme will always identify the fault closest to the scan output if there are multiple faults. The scheme makes a tradeoff between the number of XOR gates added and the diagnostic resolution. Narayanan and Das [20] have proposed to add simple circuitry to a scan flip-flop to enable its scan-out port to be either set or reset. Tekumulla and Lee [21] have proposed a partitioning of scan chains into segments and bypassing segments that contain hold time violations. When a hold time violation is located on a scan chain segment, the flip-flops in those segments are bypassed and new test patterns are created. In [22], a special circuit has been proposed to flip, set, or reset scan cells to identify the defective ones. After shifting in a chain pattern, the circuit can invert, set, or reset each flip-flop's state. The faulty cell is located via the observed unloaded value. However, since these methods typically require extra hardware, they are not acceptable in many products. In addition, defects can occur in the extra control hardware, which make diagnosis more complicated.

Software-based techniques use diagnosis algorithms to identify faulty scan cells. Compared with hardware-based techniques, software-based techniques do not need modification of the conventional scan design and are more widely adopted in the industry. The inject-and-evaluate paradigm, commonly used for logic diagnosis, can also be applied to scan chain diagnosis with a specific fault model [23]. This technique is generally categorized as model based technique. Guo and Venkataraman [24] have proposed an algorithm that identifies an upper bound (UB) and a lower bound (LB) of scan cells within a chain to locate a faulty cell. A jump simulation technique has been proposed in [25] to diagnose a single chain fault. For each failing pattern, a simulator performs multiple simulations to quickly determine the UB and LB. After the range is finalized, a detailed simulator performs parallel pattern simulation for every fault in the final range. In [26], an effect-cause method has been proposed using dynamic learning. This method is based on several learning rules that analyze the circuit, patterns, and mismatched bits. It backtraces the logic cones to determine which cells should be simulated in the next iteration. As a result, only a few cells need to be simulated to find suspects instead of simulating every scan cell within a range. In [15], a method for multiple fault diagnosis has been proposed. The work is based on: 1) double candidate range calculation; 2) dynamic learning; and 3) 2-D linear search and can successfully identify the dominant fault pair in a chain.

Another set of software-based approaches, known as signal profiling based approach or data-driven chain diagnosis approach, have been proposed in [27]. These methods use special patterns for chain diagnosis. These patterns could be either functional test patterns that start from an initial state, or scan patterns that start with all 0s or all 1s. The main objective of such patterns is to avoid (or minimize) any faulty value introduced during loading of scan chains. Therefore, all (or most) of the failing bits are caused in the process of unloading scan chains. Diagnosis can then be performed by monitoring from which scan cell the signal probability has significantly changed. The algorithms select patterns to randomize signal probability of scan cells before unloading. The main

ISSN (ONLINE): 2454-9762
ISSN (PRINT): 2454-9762
Available online at www.ijarmate.com

International Journal of Advanced Research in Management, Architecture, Technology and Engineering (IJARMATE)
Vol. 2, Issue 3, March 2016

disadvantages of the data-driven approaches are: 1) manufacturing ATPG scan patterns cannot be used for diagnosis. This is because the faulty values during scan chain loading procedures could be propagated to faulty chain itself and will compromise signal profiling results; and 2) it cannot be applied to circuit with embedded compression logic without using bypass mode. In [28] and [29], an adaptive signal profiling algorithm using manufacturing ATPG patterns have been proposed.

Most of the proposed techniques work well in either compaction free environment or only a single chain failure cases. When multiple chains fail in a compaction environment, diagnosis becomes extremely difficult. In the next section, we have proposed a masking strategy that will aid in diagnosis of scan chains enormously.

III. Proposed Approach-An Example

Fig. 2 shows an example of how the use of space compactors can produce improper compacted responses at an output channel when multiple chains, observed through the same compactor fail. From Fig. 2, it can be noted that chain 1 has a stuck-at-1 fault at cell 1. Chain 3 has stuck-at-1 fault at cell 3. During unloading, all the upstream cells of cell 1 of chain 1 and cell 3 of chain 3 will be affected. After space compaction, some of them will cancel each other messing up the compacted response completely. Now, it is very difficult to predict the actual failures of cells from the compacted response. It may be noted that the flush tests run previously have identified the chains 1 and 3 to be faulty, out of the *n* chains feeding the compactor. However, we cannot tell which cells have actually failed. Using some strategy, if we can mask out chain 3, we will possibly be able to diagnose the failing cells of chain 1.

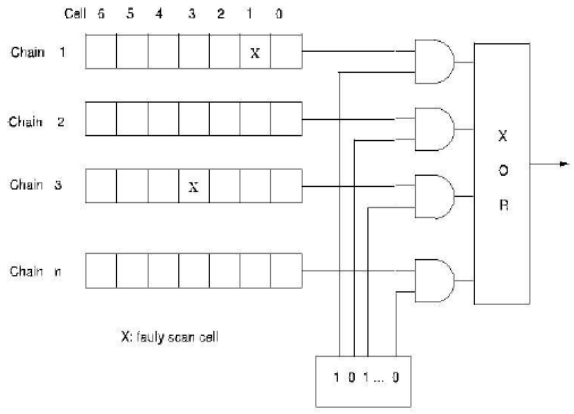

Fig. 2. Example of multiple chain failure [7].

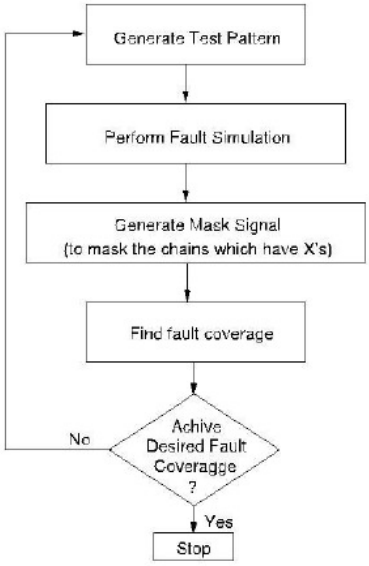

Fig. 3. Normal test generation flow.

Similarly, masking out chain 1 aids in the diagnosis of failures of chain 3.

The work proposed in [7] includes a simple heuristic to generate mask signals to help in diagnosis. It checks whether for a pattern, any of the chains for a compactor is masked during normal test flow. If no chains are masked for a pattern, the chains which have been masked the fewest number of times by previous patterns are masked. The diagnosis capability of multiple scan chain failures has been measured in terms of the frequency of a scan chain being masked/observed. It is obvious that the measure used is an indirect one. The method is simple but not very rigorous.

IV. Flow of Test Generation Process

The normal flow of test generation is shown in Fig. 3. After generating a test pattern, fault simulation is performed. One or more scan chains may capture Xs. Mask signals are generated to prevent these Xs from reaching the space compactor as they can corrupt the signature. So, scan chains that have Xs are masked off

In this paper, we take as input the test pattern set and the mask signals generated by this normal method. We modify the (or generate new) mask signals according to the proposed method. Along with the chains that contain Xs, we mask a few more chains to increase the chance of diagnosing scan chain failures. In general, a lot of redundancies are present in fault detection by the test patterns in the sense that a particular fault is detected by several patterns. Therefore, intelligently masking scan chains will not affect the fault coverage much. To keep the fault coverage unaltered (if reduced), more patterns may be added into the set. The modified flow is given in Fig. 4..

ISSN (ONLINE): 2454-9762
ISSN (PRINT): 2454-9762
Available online at www.ijarmate.com

International Journal of Advanced Research in Management, Architecture, Technology and Engineering (IJARMATE)
Vol. 2, Issue 3, March 2016

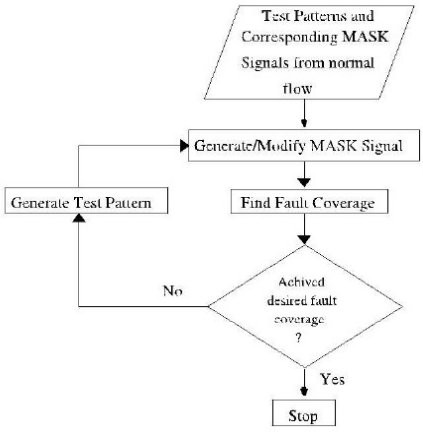

Fig. 4. Proposed modified test generation flow.

V. PROCEDURE TO IMPROVE DIAGNOSABILITY

In this section, we have presented the proposed strategy to mask additional scan chains to improve the diagnosis capabilities. The masking strategy is based on generating some essential patterns for any combination of failed scan chains. This will ensure that the compacted responses of the essential patterns will only be affected by one particular chain. Diagnosis of that chain will be extremely benefited by this. It may be noted that masking is done only at the compactor side, not at the decompressed side. Therefore, the fault from any faulty chain can still cause loading errors and these errors can be propagated to good chains as well as faulty chains. However, if for a pattern only one failed chain is observed (all remaining faulty chains being masked), the failure at a compacted response may come from: 1) the loading errors on the masked faulty chains; 2) the loading errors at the non masked faulty chains; and 3) the unloading errors on the non masked faulty chains. The unloading errors on the non masked faulty chain will affect almost the entire compactor response bit stream, whereas the loading errors on the masked faulty chain can only affect a few of the bits. Therefore, a scan chain diagnosis algorithm based on the partial matching response can successfully identify the failing chains even in the presence of loading errors on the masked faulty chains. Hence, the problem of generating mask signals (normally mask signals are generated to block Xs present in the responses) has been formulated for each pattern such that the compacted responses have enough information (essential patterns) for diagnosis of the failed scan chains.

However, masking signals are generated during the test generation phase (Fig. 4), much before the actual testing phase (when flush patterns are applied). Thus, at the time of mask generation, it is not known which chains are going to fail during test. This complicates the mask generation procedure to a great extent. The proposed approach should consider all possible situations.

First of all, an input, Number_of_faulty _chain, is taken from the user. It is an UB on the number of faulty chains that are expected to be observed using a single compactor. The value assumed in this paper for Number_of_faulty_chain is four. Since we are considering the maximum number of faulty chains observed at a single compactor, the assumption is reasonable. In general, let that number be k. Let the total number of scan chains coming into a particular compactor be n. Since it is not known which k (or $<k$) chains are faulty, it is required to consider all possible k-chain combinations while generating the mask signals. Next, a suitable metric to evaluate the masking procedure is required. This is because, the masking scheme can only be evaluated at the testing phase using a diagnosis algorithm. However, we should be able to evaluate the masking during mask signal generation phase itself. The metric should be able to model the actual effects accurately. In the following, it has been discussed with an example.

Consider a case with $n = 10$ and $k = 3$. The scan chains are numbered from 1 to 10. Let us assume that chains 1, 2, and 3 are faulty. Since the proposed method is based on the theory that if the number of times a faulty chain is observed alone (with no other faulty chains) is more, it is easy to diagnose that chain and the proposed metric is also based on this. Let X be a variable, defined as follows:

$$X = P + 0.25 * Q \qquad (1)$$

where P = Number of times chain 1 is observed (chains 2 & 3 are masked) and Q = Number of times chain 1 is observed with either chain 2 or chain 3 (the other one is masked).

The first part corresponds to the essential patterns, for which chain 1 is observed alone without any other faulty chain. They are the best suitable pattern for diagnosis of chain 1. The second part considers those test patterns where scan chain 1 is observed with one more faulty chain (2 or 3). These patterns may have information to diagnose chain 1. However, diagnosis information will be relatively less (that is why it has been multiplied by 0.25). The value 0.25 has been selected empirically. The assumption here is that the information for diagnosis is decreasing exponentially with more number of faulty chains being observed together. Beyond 2, we have assumed that the compacted response have almost no information that can be useful in diagnosis. Therefore, X is the measure of how well we can diagnose chain 1 (note that X is only applicable for faulty chain combination 1, 2, and 3. For any other combination, value of X will be different). Similarly, let variables Y and Z denote the measures of how well chains 2 and 3 can be diagnosed respectively. We define another variable W as follows:

$$W = X + Y + Z \qquad (2)$$

where W is the measure of ease with which chain combination 1, 2, and 3 can be diagnosed. For the i th combination, let it be denoted by W_i . The reason for choosing W as in the above equation is that W will be higher when X , Y , and Z are high.

In the above example, we assume that the chains 1, 2, and 3 are faulty. As discussed in the earlier section, we do not know which chains will fail during actual testing, we need to

ISSN (ONLINE): 2454-9762
ISSN (PRINT): 2454-9762
Available online at www.ijarmate.com

International Journal of Advanced Research in Management, Architecture, Technology and Engineering (IJARMATE)
Vol. 2, Issue 3, March 2016

consider all possible chain combinations. Thus, we need to calculate W for every ${}^{n}C_k$ chain combinations. Let SC_c denote the total number of k-chain combinations (${}^{n}C_k$). We define our problem of mask generation as follows.

Problem 1: Generate mask signals for test patterns suchthat the sum over W for all possible k-chain combinations (SC_c) gets maximized keeping pattern count as small as possible.

In precise form

$$\text{Maximize} \sum_{i=1}^{SC_c} W_i \qquad (3)$$

with the constrain of minimize T (T is the number of test patterns present in the test set).

This will ensure existence of some essential patterns for any combination of failed chains. However, masking arbitrary scan chains will lower fault coverage. Therefore, mask generation scheme should minimize pattern count as well.

A. Proposed Algorithm

A simple heuristic has been proposed to solve the problem. The input to the algorithm is the pattern set and masks signals generated by the normal test generation flow. We make a list of all possible scan chain combinations and keep track of their corresponding metrics. Note that, we do not have to explicitly consider the cases where actual number of faulty scan chains is less than the maximum number of faulty chains considered by the tool. They are automatically taken care of when bigger combinations are considered.

For a test pattern, top 15% scan chains in terms of their FDW are found out and they are always observed. Now, consider a fault that is not detected by any of the remaining patterns excepting the present one, and is also not detected by the current pattern due to masking of scan chains. Therefore, one more pattern is required to detect this fault—thereby increasing the test length. This step ensures that the scan chains with these types of faults are never masked leading to reduction in pattern count.

Next, α number of combinations with minimum W values are found out. α is defined as

$$\alpha = \frac{SC_c}{T} . \qquad (5)$$

Here T is the number of test patterns present in the test set and SC_c is the total number of k-chain combinations (${}^{n}C_k$).

All chains associated with these combinations are found. Chains for which the mask signals have already been generated are left out. The remaining chains are then sorted in descending order of number of times they appear in those combinations. Next, we continue to mask scan chains from the top of the list until the fault coverage drops below a certain limit or the list is exhausted. The idea to mask the chain that has appeared most of the times comes from the fact that masking that chain has high impact on cumulative W value. This step ensures that the chain combinations with least W values at each step are considered and ultimately the masking scheme is generated considering all possible combinations.

Algorithm 1 Mask Generation Procedure

Require: Test Pattern Set and corresponding Mask signals, $Init\_FC$, Compaction Ratio (n) and Maximum Number of faulty chains (k)
Ensure: Modified Test Patterns Set and corresponding Mask signals
1: Calculate α
2: Generate all possible combination ${}^{n}C_k$ and maintain a list, which contain the X value of each chain in the combination, and W value for each combination. Update the list according to the input masking signals
3: **for all** *remaining patterns in the test set* **do**
4: Calculate the FDW for each scan chain and observe the top 15% of them
5: Find the combinations with minimum W values. Find the chains associated with these combinations and sort them in descending order according to the number of times they appear in the combination. Maintain a list for these chains.
6: Set $FC = 0.0$
7: **while** $FC <$ *specified limit && list of chain is not exhausted* **do**
8: Mask the most appeared chain (if not observed by step 4)
9: Update

$$FC = FC + \frac{\#\ faults\ detected\ on\ the\ chain}{Total\ \#\ faults\ detected\ by\ the\ pattern}$$

10: Delete the chain from the list
11: **end while**
12: Update the X and W values in the list of combinations
13: **end for**
14: Find fault coverage. If the fault coverage is satisfactory, go to step 16. Otherwise go to step 15
15: Generate test patterns for the remaining undetected faults. Go to Step 3
16: Output the test pattern set and the corresponding mask signals
17: **return**

After generating the mask signals for these chains, the remaining chains are observed. W value is recalculated for each combination. This process is repeated for all the patterns generated by the normal method. Next, fault coverage is calculated with the modified masking signals, and if it is less than the desired value, new test patterns are generated. The same procedure of mask signal generation is applied to the new patterns also.

Let us consider an example with four chains (with chain indices 1, 2, 3, and 4). Let n be four and k be three. There are two test patterns (P_1 and P_2) generated by normal method. Let the mask signals for p_1 and p_2 be 0111 and 1111 (the leftmost bit is for chain 1 and rightmost is for chain 4) respectively.

For P_1, let chain 4 needs to be observed due to high FDW value. Now, the two combinations with minimum W values

ISSN (ONLINE): 2454-9762
ISSN (PRINT): 2454-9762
Available online at www.ijarmate.com

International Journal of Advanced Research in Management, Architecture, Technology and Engineering (IJARMATE)
Vol. 2, Issue 3, March 2016

are selected. They are combination nos. I and IV. Now chain 2 and 3 are in the list of chains as they are in both the combinations. Let chain 2 be masked and the situation be such that no further chains can be masked maintaining the desired fault coverage. So the mask signal for P_1 is 0011.

For P_2, let the chains 1 and 2 need to be observed. Now, the two combinations with minimum W values are combination nos. III and IV. Chains 3 and 4 are in the list. Let chain 3 be masked and no further chains can be masked maintaining the desired fault coverage. So the mask signal for P_2 is 1101.

Due to the extra masking of P_1 and P_2, we require one more pattern P_3. Suppose chains 1 and 2 are observed. The two combinations with minimum W values are II and III. The chain list contains three and four. Chain 4 will be masked as it appears in both the combinations. Chain 3 needs to be observed. So the mask signal for P_3 is 1110. W values at different stages are shown in Table II.

VI. Proposed Tester Architecture

It has been mentioned earlier that the scan chain diagnosis procedure is even more difficult because, when a circuit fails the flush test, instead of the entire test set, only a few test patterns are applied. These patterns are generally applied from the top of the pattern list. The applied patterns might not have enough information to diagnose the scan chains. Since our masking scheme works on the entire pattern set, the policy of applying only first few patterns in conventional tester would result in poor performance of the suggested strategy. In the following, we propose simple tester architecture to implement our scheme. The following are the components of the tester.

1) *Flush Pattern Memory:* Where the flush patterns are stored.
2) *Test Pattern Memory:* Where test patterns are stored.
3) *Mask Signal Memory:* Where the mask signals corresponding to each pattern are stored. Mask signals are also necessary for flush patterns.
4) *Response Memory:* Compacted responses are stored unit. It also has a comparator that compares the circuit under test (CUT) response with the golden response and generates Pass/Fail signal.
5) *Failed Chain Index Memory:* It contains the indices of the scan chains failing flush test.
6) *Control Logic:* The control logic checks the mask signal and the failed chain indices to decide whether to apply or skip the current pattern.

TABLE II
Example: W Values at Different Stages

		values for input mask signals				values after P_1				values after P_2				values after P_3			
No.	Comb.	X	Y	Z	W	X	Y	Z	W	X	Y	Z	W	X	Y	Z	W
I	1,2,3	0	0	0	0	0	0	1.0	1.0	0.25	0.25	1.0	1.5	0.25	0.25	1.0	1.5
II	1,2,4	0	0.25	0.25	0.5	0	0	1.0	1.0	0	0	1.0	1.0	0.25	0.25	1.0	1.5
III	1,3,4	0	0.25	0.25	0.5	0	0.25	0.25	0.5	0.25	0.25	0.5	1.0	0.5	0.5	0.5	1.5
IV	2,3,4	0	0	0	0	0	0.25	0.25	0.5	0.25	0.25	0.5	1.0	0.5	0.5	0.5	1.5

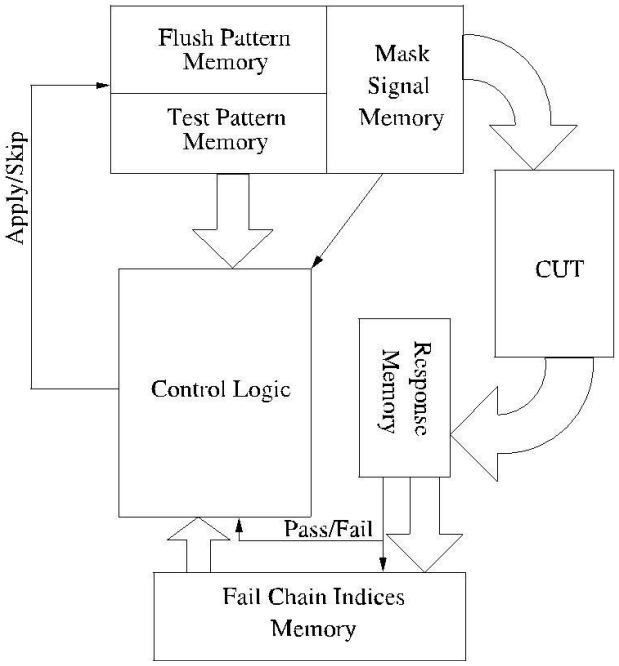

Fig. 5. Proposed tester architecture.

First the flush patterns are being applied to the CUT. If the circuit fails the flush test, the failed chain indices are stored in failed chain index memory. Then the control logic will check the mask signal and the failed chain indices, and decide whether to apply or skip the current pattern. Fig. 5 shows the block diagram of the tester.

Compared with the conventional external tester, the modified tester contains a control logic that decides whether to apply or skip a pattern. The rest of the tester architecture remains unaltered. For the proposed modified architecture, we have actually implemented two different approaches to select which of the patterns are going to be applied. In the first approach, the test patterns are selected on the fly. For a pattern next in the list, the tester generates a signal, apply/skip based on whether the pattern contains enough information for diagnosis. When the applied pattern count reaches a specified limit, the tester stops. The advantages of this approach are that the control logic for the tester is fairly simple and also the test time is less. The control logic of this approach can easily be generated by calculating the number of unmasked failed chains for a particular pattern. As given in (1), if more than 2 faulty chains are observed through the compactor, there is a very low chance to find any valuable information. Thus, if the number of observed failed chains is more than two, we skip the pattern. Otherwise, we apply it.

In the second proposed method, the tester checks all the patterns and finds out the suitable patterns which may have the most valuable information for diagnosis. The control logic of the tester is a bit complex, but a better diagnosis result could be achieved. The control logic assigns weights to every test pattern based on (1) and also considering the number of times a particular chain(s) has(have) already been observed.

ISSN (ONLINE): 2454-9762
ISSN (PRINT): 2454-9762
Available online at www.ijarmate.com

International Journal of Advanced Research in Management, Architecture, Technology and Engineering (IJARMATE)
Vol. 2, Issue 3, March 2016

Let us take an example. Consider the failed chain indices to be 1 and 2. Suppose patterns 1 and 2 observe chain 1 only while pattern 3 observes only chain 2. Since all the patterns observe only one failed chain, they should have similar weight (as per method 1, discussed earlier). However, as we have already taken a pattern which will help in diagnosis of chain 1, the weight of pattern 2 is less than that of pattern 3. Similarly, the chain combinations are also considered while generating the weight for a pattern. Once the patterns have been assigned weights, the best patterns are found out and applied.

VII. Methodology to Diagnose of Multiple Faults In Multiple Chains

When multiple chains fail due to the presence of multiple faults, diagnosis becomes even more difficult. For scan chains, we have considered at most two faults because the fault pair, consisting of a fault close to the scan-in and other fault close to the scan-out, will dominate the remaining faults [15]. The proposed masking scheme can be useful in this case also.

Let chains *A*, *B* , and *C* be faulty and each of them has single or double faults. From the masking signal, the essential patterns for *A*, *B* , and *C* can be found out. First, chain *A* is diagnosed with only the essential patterns for *A*. Top 10% suspected faults (may be single or double) for chain *A* are stored in a list. Similarly, suspected faults for chain *B* and *C* , have also been found out using their corresponding essential patterns. Now, patterns for which only chains *A* and *B* have been observed at the compactor, are found out. For each suspected fault for chain *A*, every fault in the suspected list for chain *B* are injected and simulated with these patterns. Then, a combined list, consisting of a suspected fault (or double faults) from chain *A* and a fault (or double faults) from chain *B* is prepared with the top 10% solutions. Similar combined lists of faults for chain combinations, (A, C) and (*B* ,*C*) have also been found out. Finally, for each chain, a ranked list of faults is presented. The rank for a fault is based on the scores obtained by the same in both the phases and the number of times it appeared on the combined list.

If a chain does not have an essential pattern, all the single and double faults are used in the second phase. If for a pair of chains, there does not exist a pattern for which only these chains are observed, ranking of their faults are based on other lists. For example, let there be no patterns for which chain *A* and chain *B* are observed (*C* is masked). Then, faults for chain *A* are ranked based on their rankings with essential patterns and ranking with (*A*,*C*) lists.

The methodology is independent of the diagnosis algorithm to be used to diagnose a failed chain. It only utilizes the masking strategy and finds the best patterns to narrow down the suspected faults for a chain. Then, combining suspected lists for different chains, the method gives a complete list of suspected faulty cells for each chain. Since it uses selective patterns for diagnosis, the time required to diagnosis is also reduced.

The algorithm of the proposed methodology is given in Algorithm 2.

Algorithm 2 Multiple Fault Diagnosis

Require: Test Pattern Set and corresponding Mask signals, list of failed chains
Ensure: Diagnosis of multiple faults in multiple chains
1: Find essential patterns for all the failed chains
2: **for** *each failed chain* FC_i **do**
3: **if** $\exists$ *essential patterns for* FC_i **then**
4: Use the essential patterns for FC_i to diagnose the chain
5: Take the top 10% best candidate faults and store it in $list_i$
6: **else**
7: Take all single and double fault candidates and store it in $list_i$
8: **end if**
9: **end for**
10: **for** *every pair of failed chain* FC_i *and* FC_j $(i \neq j)$ **do**
11: **if** $\exists$ *patterns for which only* FC_i *and* FC_j *are observed* **then**
12: **for all** *possible combinations of fault tuples from* $list_i$ *and* $list_j$ **do**
13: Simultaneously inject one fault from $list_i$ in FC_i and one fault from $list_j$ in FC_j
14: Use the patterns to diagnose FC_i and FC_j simultaneously
15: **end for**
16: Generate a combined ranked list of faults based on the diagnosis result
17: **end if**
18: **end for**
19: Combine all the results obtained from single and double chain diagnosis and prepare the final ranked list of faults for every chain
20: **return**

VIII. Experimental Results

The proposed algorithms have been implemented using C and results are obtained for ISCAS'89 and ITC'99 benchmark circuits. Since, a full scan version of each circuit has been assumed; no static sources of Xs are present in the responses. We have injected Xs in random scan cells after simulating a test pattern. Then, we have performed the normal mask signal generation procedure as given in Fig. 3. The number of Xs

Table III
Number of Test Patterns Obtained by Different Approaches

Circuits	Compression Ratio	No. of Test Patterns			% Extra Patterns	
		by NM	by [7]	by Our Approach	w.r.t. NM	w.r.t. [7]
s5378	20X	262	266	273	4.03	2.56
s9234	24X	383	386	392	2.30	1.53
s13207	25X	475	481	479	0.84	-0.42
s15850	25X	441	443	442	0.23	-0.23
s38417	30X	918	919	920	0.22	0.22
s38584	30X	699	706	700	0.14	-0.86
b14	10X	946	952	955	0.94	0.31
b15	25X	642	642	649	1.08	1.08
b20	16X	1189	1191	1221	2.62	2.46
b21	16X	1342	1346	1356	1.03	0.74
Average					1.34	0.74

ISSN (ONLINE): 2454-9762
ISSN (PRINT): 2454-9762
Available online at www.ijarmate.com

International Journal of Advanced Research in Management, Architecture, Technology and Engineering (IJARMATE)
Vol. 2, Issue 3, March 2016

inserted is about 1% of the total response bits. We have also implemented the method proposed in [7] for the purpose of comparison.

After generating the masking signals for normal method (as in Fig. 3), implemented method in [7], and the proposed method, comparisons are carried out to find the suitable one for diagnosis of multiple chain failures. For this, a tester emulator which will first apply flush tests (in 1-hot masking mode) to the circuit, and find out the faulty chains, has been made. In general, when a circuit fails during scan chain testing, not all the test patterns are applied to the circuit. Only a few of the patterns are applied and the obtained responses are used to diagnose the scan chains. Therefore, the tester emulator applies a few patterns (along with the mask signals) in presence of faults and produces the compacted faulty response. We have used top 20% of the total pattern present in the test set in our experiment. Using the normal tester emulator, patterns generated by normal method, implemented method [7] and the proposed method are simulated and compacted responses are used for diagnosis. The proposed modified tester architectures are also implemented. For them also, we have applied 20% of the total patterns, selected either via the first method or via the second method.

Table III shows the number of patterns generated by differ-ent approaches. The second column notifies the compaction ratio considered in this approach. A 30× compaction implies that 30 scan chains are observed through a single compactor output. We have used Atalanta [30] to generate test patterns. The third column, denoted as normal method (NM), shows the patters generated by the normal method. Due to intelligent selection of scan chains to be masked, our approach virtually uses the same test set as production test.

The single fault diagnosis algorithm, we have used here, is based on inject and evaluate paradigm. For each test pattern, faults are injected on every scan cell (only in the faulty chains) one at a time, and the responses are evaluated. The responses are matched with the actual responses and the final list is provided based on reward and penalty basis. The reward is given to a candidate fault based on number of test patterns it can explain; whereas penalty is given for the mismatched outputs it produces.

We define two variables, diagnosability (Dia.) and first hit rank (FHR) to measure the quality of the mask generation procedure.

TABLE IV
RESULTS FOR DIAGNOSABILITY FOR FAULTS IN TWO, THREE, AND FOUR CHAINS

Circuit	For 2 Faults					For 3 Faults					For 4 Faults				
	NM	by [7]	PM	PM(T1)	PM(T2)	NM	by [7]	PM	PM(T1)	PM(T2)	NM	by [7]	PM	PM(T1)	PM(T2)
s5378	0.69	0.80	1.00	1.00	1.00	0.54	0.54	0.88	0.90	0.95	0.46	0.46	0.75	0.91	0.93
s9234	0.75	0.92	1.00	0.98	1.00	0.55	0.62	0.95	1.00	1.00	0.52	0.59	0.94	0.94	0.95
s13207	0.95	1.00	1.00	1.00	1.00	0.65	0.74	0.98	1.00	1.00	0.61	0.68	0.83	0.96	0.99
s15850	0.83	0.93	1.00	1.00	1.00	0.67	0.68	0.97	1.00	1.00	0.69	0.71	0.91	1.00	1.00
s38417	0.93	0.98	1.00	1.00	1.00	0.57	0.62	0.95	0.95	0.97	0.35	0.38	0.87	0.95	0.95
s38584	0.95	0.98	0.98	1.00	1.00	0.73	0.75	0.87	0.98	1.00	0.55	0.61	0.74	1.00	1.00
b14	0.62	0.85	1.00	1.00	1.00	0.43	0.45	1.00	1.00	1.00	0.28	0.29	0.85	0.86	0.88
b15	0.80	0.95	1.00	1.00	1.00	0.53	0.52	0.98	0.98	0.99	0.38	0.39	0.92	0.92	0.94
b20	0.72	0.82	1.00	1.00	1.00	0.53	0.58	0.95	0.97	0.98	0.44	0.44	0.94	0.93	0.95
b21	0.88	0.90	1.00	1.00	1.00	0.55	0.57	0.98	0.98	0.98	0.43	0.40	0.87	0.88	0.89
Avg.	**0.81**	**0.91**	**0.99**	**0.99**	**1.00**	**0.57**	**0.61**	**0.95**	**0.98**	**0.99**	**0.47**	**0.50**	**0.86**	**0.94**	**0.95**

TABLE V
RESULTS FOR FHR FOR FAULTS IN TWO, THREE, AND FOUR CHAINS

Circuit	For 2 Faults					For 3 Faults					For 4 Faults				
	NM	by [7]	PM	PM(T1)	PM(T2)	NM	by [7]	PM	PM(T1)	PM(T2)	NM	by [7]	PM	PM(T1)	PM(T2)
s5378	3.90	2.45	1.07	1.07	1.03	4.23	3.92	1.69	1.6	1.28	4.49	4.36	2.42	1.53	1.48
s9234	2.50	1.55	1.30	1.28	1.08	3.60	3.25	1.27	1.28	1.30	4.10	4.10	1.75	1.46	1.40
s13207	1.68	1.38	1.10	1.05	1.03	3.30	3.08	1.35	1.18	1.13	3.86	3.82	2.46	1.41	1.28
s15850	2.53	1.43	1.15	1.33	1.08	3.32	3.05	1.43	1.13	1.15	3.56	3.56	1.84	1.13	1.09
s38417	1.55	1.23	1.07	1.10	1.05	5.87	4.75	1.08	1.57	1.13	9.09	7.93	2.00	1.75	1.75
s38584	1.43	1.35	1.23	1.13	1.00	3.00	2.57	1.70	1.28	1.17	4.25	3.74	2.88	1.39	1.34
b14	5.00	2.60	1.00	1.00	1.00	8.82	7.87	1.12	1.10	1.06	10.26	9.91	2.37	2.30	1.80
b15	2.20	1.65	1.00	1.00	1.00	5.98	5.42	1.30	1.30	1.20	6.58	6.11	1.81	1.80	1.78
b20	2.85	2.25	1.03	1.02	1.00	4.85	4.67	1.38	1.30	1.30	5.56	5.57	1.53	1.60	1.42
b21	2.00	1.83	1.10	1.05	1.02	4.58	4.65	1.10	1.10	1.10	5.41	5.42	2.11	2.10	2.08
Avg.	**2.56**	**1.77**	**1.11**	**1.10**	**1.03**	**4.76**	**4.32**	**1.34**	**1.28**	**1.18**	**5.72**	**5.45**	**2.11**	**1.65**	**1.64**

TABLE VI
RESULTS FOR DIA. AND FHR FOR MULTIPLE FAULTS IN TWO, THREE, AND FOUR CHAINS

Circuit	For 2 Faults				For 3 Faults				For 4 Faults			
	Proposed Technique		by generalized technique		Proposed Technique		by generalized technique		Proposed Technique		by generalized technique	
	Dia_r	FHR	Dia_r	FHR	Dia_r	FHR	Dia_r	FHR	Dia_r	FHR	Dia_r	FHR
s5378	0.99	1.03	0.88	2.53	0.92	1.22	0.80	2.33	0.88	1.41	0.68	2.89
s9234	1.00	1.03	0.89	2.03	0.94	1.23	0.87	2.07	0.93	1.39	0.73	2.64
s13207	1.00	1.03	0.83	2.68	0.93	1.28	0.72	2.73	0.93	1.25	0.75	2.65
s15850	1.00	1.03	0.86	2.13	0.98	1.05	0.84	2.27	0.98	1.25	0.79	2.55
s38417	0.96	1.03	0.85	2.05	0.91	1.32	0.82	2.10	0.82	1.55	0.66	2.44
s38584	1.00	1.03	0.89	2.18	0.99	1.08	0.74	2.2	0.96	1.23	0.73	2.29
b14	1.00	1.00	0.85	1.80	0.87	1.60	0.67	2.77	0.83	1.70	0.67	2.80
b15	1.00	1.00	0.98	1.75	0.93	1.35	0.91	2.10	0.83	1.63	0.73	2.45
b20	0.95	1.20	0.90	1.95	0.87	1.30	0.70	2.38	0.84	1.38	0.70	2.58
b21	0.90	1.00	0.85	2.30	0.93	1.50	0.70	2.45	0.89	1.40	0.70	2.45
Avg.	**0.98**	**1.04**	**0.88**	**2.12**	**0.93**	**1.29**	**0.78**	**2.36**	**0.89**	**1.42**	**0.71**	**2.57**

1) *Dia:* If the actual fault is among the top three resultsprovided by the diagnosis algorithm, we say that the fault is successfully diagnosed and Dia. is set as 1. Otherwise, the value of Dia. is 0.
2) *FHR:* FHR is measured as the position of the injectedfault in the ranked list of faults reported by the diag-nosis tool. The ideal values of both Dia. and FHR are 1.

Since we have assumed that at most four chains (connected to a comparator) can fail and for each chain, we have injected single faults randomly on two, three, and four scan chains (observed at a single compactor) and have found the results. We repeat the experiment 30 times and show the average results.

Tables IV and V contain the diagnosability and FHR results for two, three, and four faulty chains. For each circuit, results obtained by the NM, implemented method [7], and our proposed method (PM) are given. Under the heading PM(T1) and PM(T2), we have shown the results for the proposed tester architectures. PM(T1) gives the result for architecture 1 where patterns are applied from the top and depending on the information of the mask signal, a pattern is skipped or applied. PM(T2) denotes the result for architecture 2 where best patterns from the test set are chosen and applied. The result shows that our approach is much better than any of the existing approaches proposed so far. The PM(T1) and PM(T2) produce better results than the PM. This is due to the fact both the tester architectures help to find out suitable patterns for diagnosis of failed chains.

As discussed in Section I, a simply way to ensure essential pattern for a chain is to mask all the remaining chains for a particular pattern. For the circuit s38584, we have performed this masking and found the result. This technique requires 729 test patterns using the same initial test set used to generate patterns by the proposed method (Table III). The dia. obtained by the test set is 1.0, 0.9, and 0.78, respectively, for 2, 3, and 4 faulty chains. The FHR obtained are 1.1, 1.73, and 3.18,

ISSN (ONLINE): 2454-9762
ISSN (PRINT): 2454-9762
Available online at www.ijarmate.com

International Journal of Advanced Research in Management, Architecture, Technology and Engineering (IJARMATE)
Vol. 2, Issue 3, March 2016

respectively, for 2, 3, and 4 faulty chains. The results are obtained using the proposed tester architecture 2 (T2). The results obtained by this approach are not as good as expected, which conclude that one pattern may not be sufficient to pin point the faulty location. It justifies the proposed method of mask signal generation.

The results for multiple chain failures due to presence of multiple faults are shown in Table VI. In this experiment, we have assumed that multiple chains have failed and each chain can have up to two faulty cells. Diagnosis in this scenario is extremely difficult because of the messed up compacted responses. The patterns generated by the normal method and by [7] cannot diagnose in this case. Even the proposed method is not able to produce good dia. and FHR. Only the patterns used by the proposed tester architecture [PM(T2)], are able to diagnose the faulty chains. When these patterns are used according to the proposed multiple fault diagnosis methodology, then only the results are close to the optimum value. In Table VI, we have shown the diagnosis results for two techniques. The first one is the generalized diagnosis algorithm for multiple fault diagnosis in a single chain. Here, for each applied pattern, the algorithm injects all possible single as well as double faults in the chain and finds the compacted responses. The observed responses are then compared with these responses and a ranking of the faults are given based on the matching. The second technique is the proposed multiple fault diagnosis technique as described in Section VII. For both the techniques, number of faults in a chain is unknown.

IX. Conclusion

In this paper, we have proposed a technique to diagnose multiple chain failures in response compaction environment. The proposed technique encodes some information along with the test patterns, which can be used for diagnosis of scan chains. We have also proposed a new tester architecture which will select and apply only those patterns having enough information for diagnosis. The authors understand that an interactive ATE may not be easy to implement. However, to solve a difficult and relevant problem like this, it may be necessary. The proposed solution uses a simple logic, which the authors believe can be achieved by changing the ATE test program. But at the same time, it would be extremely useful to work with ATE Company to achieve the ultimate yield level. This may lead to new powerful testers from diagnosis angle. Though the proposed technique generates a few more patterns, our approach is able to generate almost ideal diagnostic resolution and FHR with very few patterns. The results obtained by our approach are much superior to any approach proposed so far. We have also considered multiple faults at multiple chains and the proposed multiple fault diagnosis strategy works fine for multiple faults. We are now trying to find a way out to implement the method for higher compaction ratios (more than 100×).

References

[1] L. Wang, C. Wu, and X. Wen, *VLSI Test Principles and Architectures:Design for Testability*. 1st ed. Amsterdam, The Netherlands: Elsevier,2006

[2] J. A. Waicukauski and E. Lindbloom, "Failure diagnosis of structured VLSI," *IEEE Design Test Comput.*, vol. 6, no. 4, pp. 49–60, Aug. 1989.

[3] S. Kundu, "Diagnosing scan chain faults," *IEEE Trans. Very LargeScale Integr. (VLSI) Syst.*, vol. 2, no. 4, pp. 512–516, Dec. 1994.

[4] J.-S. Yang and S.-Y. Huang, "Quick scan chain diagnosis using signal profiling," in *Proc. Int. Conf. Comput. Design, VLSI Comput.Processors*, Oct. 2005, pp. 157–160.

[5] K. Stanley, "High-accuracy flush-and-scan software diagnostic," *IEEEDesign Test Comput.*, vol. 18, no. 6, pp. 56–62, Nov./Dec. 2001.

[6] N. A. Touba, "Survey of test vector compression techniques," *IEEEDesign Test Comput.*, vol. 23, no. 4, pp. 294–303, Apr. 2006.

[7] X. Tang, R. Guo, W.-T. Cheng, and S. M. Reddy, "Improving compressed test pattern generation for multiple scan chain failure diagnosis," in *Proc. Design Autom. Test Eur. Conf. Exhibit.*, Apr. 2009, pp. 1000–1005.

[8] A. Chandra and R. Kapur, "Interval based X-masking for scan compression architectures," in *Proc. 9th Int. Symp. Qual. Electron.Design*, Mar. 2008, pp. 821–826.

[9] A. Chandra, Y. Kanzawa, and R. Kapur, "Proactive management of X's in scan chains for compression," in *Proc. 10th Int. Symp. Qual.Electron. Design*, Mar. 2009, pp. 260–265

[10] I. Pomeranz, S. Kundu, and S. M. Reddy, "On output response compression in the presence of unknown output values," in *Proc. 39thDesign Autom. Conf.*, 2002, pp. 255–258.

[11] V. Chickermane, B. Foutz, and B. Keller, "Channel masking synthesis for efficient on-chip test compression," in *Proc. Int. Test Conf.*, Oct. 2004, pp. 452–461.

[12] P. Wohl, J. A. Waicukauski, and S. Ramnath, "Fully X-tolerant combina-tional scan compression," in *Proc. Int. Test Conf.*, Oct. 2007, pp. 1–10

[13] J. C. Li, "Diagnosis of multiple hold-time and setup-time faults in scan chains," *IEEE Trans. Comput.*, vol. 54, no. 11, pp. 1467–1472, Nov. 2005.

[14] R. Guo, Y. Huang, and W.-T. Cheng, "A complete test set to diagnose scan chain failures," in *Proc. Int. Test Conf.*, Oct. 2007, pp. 1–10.

[15] Y. Huang, W.-T. Cheng, and R. Guo, "Diagnose multiple stuck-at scan chain faults," in *Proc. 13th Eur. Test Symp.*, May 2008, pp. 105–110.

[16] K. De and A. Gunda, "Failure analysis for full-scan circuits," in *Proc.Int. Test Conf.*, Oct. 1995, pp. 636–645.

[17] P. Song, F. Stellari, A. Weger, and T. J. Xia, "A novel scan chain diagnostics technique based on light emission from leakage current," in *Proc. Int. Test Conf.*, Oct. 2004, pp. 140–147.

[18] J. L. Schafer, F. A. Policastri, and R. J. McNulty, "Partner SRLs for improved shift register diagnostics," in *Proc. 10th IEEE VLSI TestSymp.*, Apr. 1992, pp. 198–201.

[19] S. Edirisooriya and G. Edirisooriya, "Diagnosis of scan path failures," in *Proc. 13th IEEE VLSI Test Symp.*, Apr./May 1995, pp. 250–255.

[20] S. Narayanan and A. Das, "An efficient scheme to diagnose scan chains," in *Proc. Int. Test Conf.*, Nov. 1997, pp. 704–713.

[21] R. C. Tekumulla and D. Lee, "On identifying and bypassing faulty scan segments," in *Proc. North Atlantic Test Workshop*, 2007, pp. 134–143

[22] W. Yuejian, "Diagnosis of scan chain failures," in *Proc. IEEE Int.Symp. Defect Fault Tolerance VLSI Syst.*, Nov. 1998, pp. 217–222.

[23] Y. Huang, W.-T. Cheng, C.-J. Hsieh, H.-Y. Tseng, A. Huang, and Y.-T. Hung, "Efficient diagnosis for multiple intermittent scan chain hold-time faults," in *Proc. Asian Test Symp.*, Nov. 2003, pp. 44–49.

[24] R. Guo and S. Venkataraman, "A technique for fault diagnosis of defects in scan chains," in *Proc. Int. Test Conf.*, 2001, pp. 268–277.

[25] Y.-L. Kao, W.-S. Chuang, and J. C. Li, "Jump simulation: A technique for fast and precise scan chain fault diagnosis," in *Proc. Int. Test Conf.*, Oct. 2006, pp. 1–9.

[26] Y. Huang, "Dynamic learning based scan chain diagnosis," in *Proc.Design, Autom. Test Eur. Conf. Exhibit.*, Apr. 2007, pp. 1–6.

ISSN (ONLINE): 2454-9762
ISSN (PRINT): 2454-9762
Available online at www.ijarmate.com

International Journal of Advanced Research in Management, Architecture, Technology and Engineering (IJARMATE)
Vol. 2, Issue 3, March 2016

[27] J.-S. Yang and S.-Y. Huang, "Quick scan chain diagnosis using signal profiling," in *Proc. IEEE Int. Conf. Int. Conf. Comput. Design*, Oct. 2005, pp. 157–160.
[28] W.-T. Cheng and Y. Huang, "Enhance profiling-based scan chain diagnosis by pattern masking," in *Proc. 19th IEEE Asian Test Symp.*, Dec. 2010, pp. 255–260.
[29] Y. Huang, W.-T. Cheng, R. Guo, T.-P. Tai, F.-M. Kuo, and Y.-S. Chen, "Scan chain diagnosis by adaptive signal profiling with manufacturing ATPG patterns," in *Proc. Asian Test Symp.*, Nov. 2009, pp. 35–40.
[30] H. Lee and D. Ha, "On the generation of test patterns for combinational circuits," Virginia Polytechnic Inst. State Univ., Dept. Electr. Eng., Blacksburg, VA, USA, Tech. Rep. 12-93, 1993.

AUTHORS BIOGRAPHY

T.Gowri has received her B.E Degree in the Department of Electronics and Communication Engineering from Mookambigai College of Engineering, tamilnadu, India in 2013 and the Master's Degree in VLSI Design from Vandayar Engineering College, Tamilnadu, India in 2016. Her area of interest is on VLSI Testing.

K.Kavitha has received her B.E Degree in the Department of Electronics and Communication Engineering from Annamalai University, Tamilnadu, India in the year 2010 and the Master's Degree in Applied Electronics from R.M.K Engineering College, Tamilnadu, India in 2013. She is currently working as an Assistant Professor in the Department of Electronics and Communication Engineering in Vandayar Engineering College, Tamilnadu,India. Her area of interest on Digital Image Processing and VLSI Testing.