Delay Fault Localization in Test-Per-Scan BIST Using Built-In Delay Sensor

Swaroop Ghosh, Swarup Bhunia, Arijit Raychowdhury and Kaushik Roy
School of Electrical and Computer Engineering, Purdue University, IN
{*ghosh3, bhunias, araycho, kaushik*}@ecn.purdue.edu

***Abstract*— Delay failures are becoming a dominant failure mechanism in nanometer technologies. Diagnosis of such failures is important to ensure yield and robustness of the design. However, the increasing circuit size limits the granularity of diagnosis, resulting in large suspect fault list. In this paper, we present a methodology for improving delay fault localization in test-per-scan BIST using on-die delay sensing at selective test points. It is demonstrated that the proposed technique can improve the resolution of fault localization for both transition and segment delay fault models. Experimental results for a set of ISCAS89 benchmarks show upto 49% (82%) average improvement in fault localization for transition (segment) delay fault models. The area overhead due to delay sensing hardware have been limited to 4%.**

Keywords - Test-per-scan BIST, delay sensor, fault diagnosis, fault localization, test point insertion.

I. Introduction

Demand for performance and portability has given rise to extremely dense miniaturized devices. This is mainly due to the fact that smaller transistors are faster. Further, more transistors can be integrated in the same die area giving the designers opportunity to put more functionalities in the chip. However, large number of transistors also translates into equivalently large defect density. A small errant particle between two metal lines that was harmless earlier may ultimately kill the chip in scaled technologies. Similarly, the shrinking gap between parallel metal lines may lead to ringing and oscillations in the circuit. To make the matter worst, parametric defects (due to process variations) also play active role along with the physical defects in scaled devices. These defects may not cause functional failures but may result into violation of power or performance constraints of the circuit. A simple (but crucial) example of such defects is delay variations due to threshold voltage fluctuation of transistors. A logic stage designed to meet a certain delay constraint may violate it, leading to wrong computation of data by the following stages. Therefore, delay test is becoming mandatory along with stuck-at test in sub-100nm technology regime [1].

High fault coverage of delay faults has become essential to ensure the robustness of designed circuits. Transition-, segment- and path delay fault models [2] [3] have been proposed for delay test. Although path delay fault model is most accurate, transition fault model has emerged as the popular model. This is due to similarity of transition faults with stuck-at faults and linear nature of the number of faults. However, segment delay model is also important because it can capture the spot-defects as well as distributive nature of the delay fault defects. High coverage usually comes at the cost of large test set which converts into increased test time. Therefore, it is important to explore new techniques for test time reduction without affecting the coverage. One of the effective solutions is Test Point Insertion (TPI) for improving the controllability and observability of internal nodes of a circuit [2]. TPI for delay fault coverage in scan-based BIST has been explored very recently by [4] [5] [6]. In [4], a path delay fault testing methodology using TPI and successive test clock trimming is presented. However, note that controlling the clock in BIST environment could be difficult. In [5], flip-flops have been used as observation points for scan-based BIST. But since the timing constraint of different nodes are different depending on their positions in the circuit, the maximum allowable delay at an internal node will be $<=$ T, where T is the clock period. Therefore, if FFs are inserted at the observation nodes, it would require the FFs to sample the logic values at delayed clocks. To avoid the clock trimming and generation of multiple delayed clocks, a delay sensor has been proposed in [6] [7], which can be used as an observation point to detect delay failures. It has been shown that both transition and segment delay fault coverage could be improved significantly at the cost of small area/delay penalty.

High fault coverage certainly helps in improving the design confidence. However, measures must be taken to ensure that the design is robust enough to sustain parametric fluctuations and manufacturing errors (to some extent). This is possible only through fault analysis of *failed dies*. The fault analysis itself is a two-step process: (a) fault detection; and, (b) fault diagnosis/fault localization. In fault detection, a set of test stimulus is applied through primary inputs (PI's) and the responses of primary outputs (PO's) are compared with expected outputs to detect failures. Particularly in scan-BIST, a set of random patterns are applied through PI's and state inputs and the responses of PO's and state outputs are compared to generate a pass/fail signal. The other part of fault analysis process is fault diagnosis where the responses of the circuit-under-test are used to isolate the root cause of failure. The output of the diagnosis process is a *suspect fault list*. To get additional clues about the suspect faults, the state responses can also be used. After diagnosis, the design/layout/manufacturing process should be tuned at the suspect fault sites to avoid the failures in new design.

Due to growing circuit size, it is becoming difficult to

Proceedings of the 12th IEEE International On-Line Testing Symposium (IOLTS'06)
0-7695-2620-9/06 $20.00 © 2006 IEEE

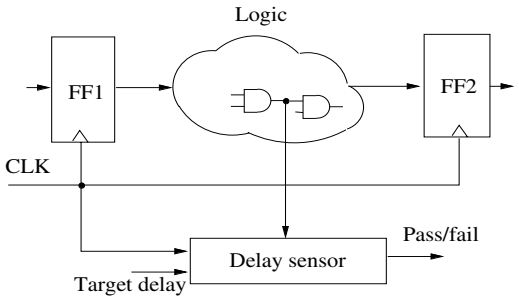

Fig. 1. A combinational logic block with delay sensor for fault detection and localization

isolate the suspect faults based only on the circuit's response at PO's. This results in a large suspect fault list, reducing the resolution of fault localization. Considering the size of delay faults and complexity of the present-day circuits, it is important to devise new techniques to reduce the size of suspect fault list. Observation points using delay sensors have been used to improve the fault detection of delay faults in scan-BIST environment in [6]. In this paper, we show that the response of the sensor can also be used along with the PO's and state responses to improve the delay fault localization (i.e., reducing size of suspect fault list). An example of delay sensor insertion in the logic circuit is illustrated in Fig. 1. The node of interest (or probe node) is fed to a delay sensor. A target delay is also provided against which the sensor detects a delay violation at this node. It can be noted that apart from detection, the sensor's response (pass/fail) also provides valuable insight about the location of the delay failure. Since the structure of the circuit and the position of the probe node are known, it becomes easy to track the origination of failure by analyzing the response of circuit and sensor. Further, multiple sensors inserted in the circuit improves the granularity of localization. In this paper, first we study the issues associated with delay sensors for delay test and diagnosis. Then we propose a fault dictionary-based approach to improve the fault localization ability of a scan-BIST by combining the scan test and delay sensor responses.

The remainder of the paper is organized as follows. In section II, we provide a comparative study of delay sensing hardwares. In section III, we discuss the modified test-per-scan BIST architecture for the delay testing and diagnosis. Section IV describes the proposed fault localization methodology. Section V presents simulation results and section VI concludes the paper.

II. Delay sensors and their applicability in delay test/diagnosis

Several delay sensors have been proposed in past for wide range of applications. A delay sensor can measure the delay of a signal at a specified node w.r.t the rising or falling edge of the reference clock. In this section, first we study three delay sensor designs proposed in past namely, time-to-voltage converter (TVC), digital-to-time converter (DTC) and built-in delay sensor (BIDS). Then we address the issues related

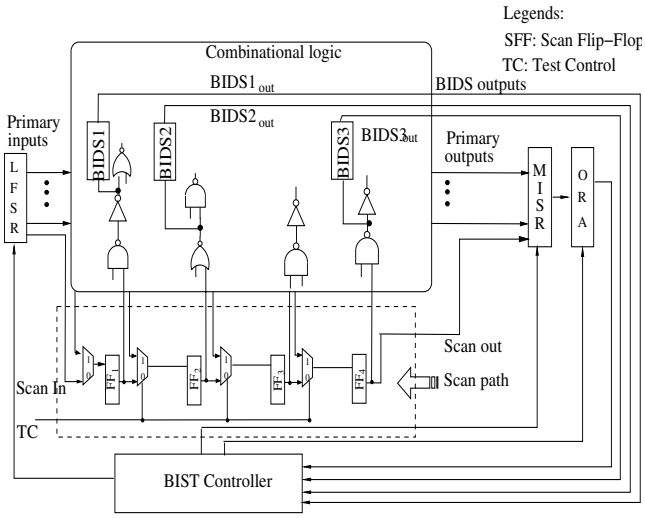

Fig. 2. Test-per-scan BIST architecture using delay sensing hardware

to applicability of delay sensors for delay testing and and subsequently for fault localization.

A. Comparison of delay sensors

Time-to-voltage converter: An analog time-to-voltage converter with a resolution of 0.5ns has been presented in [8] for applications in high-energy physics experiments. The TVC method works on the principle that the voltage of the capacitor is directly proportional to the charging time. The capacitor charges between the START and STOP pulse and thus the time interval between START and STOP is converted to the voltage of capacitor. For application in delay test/diagnosis a controller would be needed to generate START and STOP pulses at proper instant.

Digital-to-time converter: This technique was proposed in [9] as a part of an on-chip oscilloscope which is capable of performing high speed test in digital and mixed signal environment. It uses a pair of delay-locked-loop (DLL) to achieve high timing resolution. A single DLL locks the delay of an N stage voltage-controlled delay line (VCDL) to one clock period (T_{clk}). Thus the timing resolution obtained by one stage turns out to be T_{clk}/N. However, the second DLL has M stage VCDL producing a resolution of T_{clk}/M per stage. By cleverly using the sample clock from one DLL and trigger clock from another one, a resolution of $T_{clk}/N - T_{clk}/M$ (approx. 10.4ps) have been obtained. This sensor can be used for very fine measurement of delay. The only issue with this design may be the complexity and area overhead which could be very large if multiple delay sensors are inserted in the circuit for delay test.

Built-in delay sensor: In Built-in delay sensor [6] [7], the delay of a signal is first converted to a voltage and then compared against the voltage corresponding to the expected delay of that signal. The time-to-voltage conversion is performed as follows: A sawtooth pulse is generated at the clock edge which falls linearly throughout one clock cycle. The arriving signal

at the probed net holds the value of sawtooth voltage (say V_x) at that instant in a capacitor. The expected delay of the signal at the probed net w.r.t. the clock edge is known and converted to a corresponding voltage (V_{ref}). A delay failure is detected if $V_x < V_{ref}$. The resolution of this sensor is reported to be $T_{clk}/28$ where T_{clk} is the clock period.

Apart from the above mentioned delay sensors, other delay measuring hardwares have also been proposed in [10] and [11]. However, these designs are also complex and cannot be directly used as test points in delay testing. For the proposed test and diagnosis purpose we have used BIDS described in [7]. This is mainly due to its low area/delay overhead, process/glitch tolerance and reasonable resolution compared to other two sensors. However, *our delay fault localization method is generic and not limited by the choice of delay sensor.*

B. Delay sensor issues in delay test/diagnosis

Following points should be considered while using the delay sensors in test/diagnosis: (a) the sensor should be isolated as much as possible from the digital components (e.g., by placing dummy metals), (b) once the test points are determined, the sensors can be considered as *black boxes* during placement/routing, (c) the area/delay overhead due to sensors should be kept within limit (in our case, the area overhead is 4% while the delay overhead is 2%), (d) if the fabrication of metal capacitors are not supported by the process (but needed by the sensor), then MOS capacitors may be used at the expense of linearity of the circuit and, (e) the testing/calibration of sensors should be combined with the testing of other analog components present in the circuit to reduce the test cost.

TABLE I

Procedure *LocalizeFaults*
Input: Netlist, Faulty patterns T_1 for PO/SO, Faulty patterns T_2 for BIDS and, BIDS outputs for T_1 and T_2 **Output:** Suspect fault list
1. Read the netlist 2. Create dictionaries D_1 and D_2 by performing fault simulation 3. **for all** patterns in T_1 Compute S; 4. **for all** patterns in T_1 & T_2 Compute $SB_1 = \cap FB(bf_i)_{i=failing\ BIDS}$ 5. Compute $SB_2 = \cup FB(b_l)_{\forall\ BIDS}$ 6. Find the case number by observing the appearance of fault(s) at PO/SO and BIDS outputs for T_1 and/or T_2 7. Case 1: **return** $S - SB_2$ 8. Case 2: **return** SB_1 9. Case 3: **return** $S \cap SB_1$ 10. Case 4: **return** $S \cap SB_1$ 11. exit.

III. Test-per-scan bist architecture using delay sensors

In previous section, we presented a comparative study of sensors. In this section, we discuss the overall test and diagnosis scheme by using the sensors as test points in the circuit. First, we present the modified test-per-scan BIST-based test architecture with delay sensor circuits [6] inserted at internal nodes for efficient fault detection. Then we discuss how this architecture can facilitate the fault diagnosis process.

We illustrate the modified test-per-scan BIST architecture using three BIDS, namely, BIDS1, BIDS2 and BIDS3, as shown in Fig. 2. Only the nodes with better controllability and poor observability are chosen for sensor insertion [6]. The scan chain consists of four flip-flops, namely, FF1, FF2, FF3 and FF4. The test patterns for the PI and the scan latches are generated as weighted random patterns by a Linear Feedback Shift Register (LFSR). The PO and scan latch outputs are provided to a Multiple Input Signature Register (MISR). The signatures generated by MISR are fed to ORA (Output Response Analyzer). At the end of each test pattern application, ORA and BIDS responses (one bit for each BIDS) are collected by the BIST controller.

After performing the delay test and collecting the responses, it is essential to know the defective logic sections so that corrective actions can be taken to improve the yield/robustness of the design. In scan-BIST environment, the diagnosis process is carried out with the help of responses of the circuit and the state outputs for a passing or failing test pattern. The output of the diagnosis process is a suspect fault list. Considering the size of present day circuits, the suspect fault list may be very large unless, (a) a huge set of test patterns are applied or, (b) on-chip observable points are inserted. Note that our test methodology naturally provides observable test points (in form of $BIDS1_{out}$, $BIDS2_{out}$ and $BIDS3_{out}$ in Fig. 2) which gives an extra clue about the failures in the diagnosis process. It can be used to reduce the suspect fault list, thereby improving the diagnostic resolution. In next section, we present a methodology to utilize the BIDS responses in trimming down the suspect fault list.

Note that there may be an extra routing overhead associated with connecting several BIDS outputs to the BIST controller for diagnosis. If required by design constraints, the routing overhead can be reduced at the cost of resolution of fault localization. In such a case, some of the BIDS outputs should be OR-ed and provided to the BIST controller.

IV. Fault localization using a fault dictionary-based approach

In this section, we present a methodology to reduce the suspect fault list by using the delay sensor responses. We demonstrate the benefit of BIDS in improving the fault localization using a pass/fail dictionary-based approach.

In a pass/fail dictionary, for each test t_i and for each fault f_j, we store one-bit information indicating whether t_i detects f_j or not [12]. For example, suppose a circuit has three faults, say, f_1, f_2 and f_3 and two test patterns, say, t_1 and t_2. If t_1 detects fault f_2 and t_2 detects faults f_1 and f_3, then a pass/fail dictionary would contain entries 010 and 101 respectively for patterns t_1 and t_2.

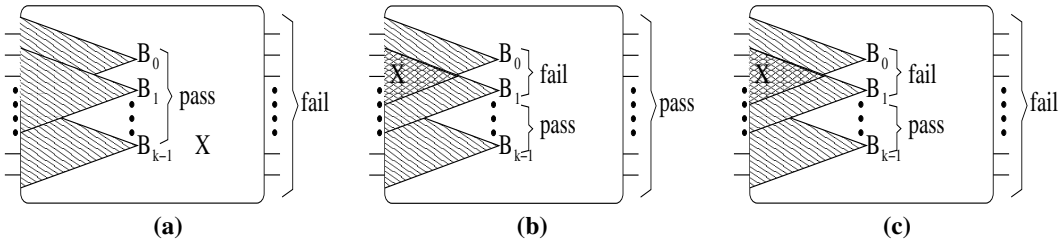

Fig. 3. Different cases arising during fault localization: (a) case 1, (b) case 2 and, (c) case 3.

A. Basic concept

Let us consider the case when there is no BIDS present in the circuit. We create a dictionary D_1 for all primary outputs (POs) and scan outputs (SOs). We denote the i^{th} failing patterns by tf_i and j^{th} passing patterns by tp_j. Now under the single fault assumption, it is evident that a single fault is responsible for all failing patterns. Consequently, the suspect fault list (S) must contain the faults lying at the intersection of the faults detected by all failing test patterns. Moreover, none of the faults covered by passing patterns are responsible for the failure [13]. Therefore, all faults detected by passing test patterns should be subtracted from S. In a nutshell, if $F(tf_i)$ denotes the set of faults covered by the i^{th} failing pattern and $F(tp_j)$ denotes the set of faults covered by the j^{th} passing pattern, then S is given by,

$$S = \{\cap F(tf_i)\} - \{\cup F(tp_j)\}, i = 1\ to\ p, j = 1\ to\ q \quad (1)$$

where p (q) is the total number of failing (passing) patterns.

The suspect faults calculated by equation 1 correspond to the circuit without BIDS. For analyzing fault localizing ability of the circuit containing BIDS as observation points, we create another two-dimensional dictionary D_2, which contains one-bit information indicating if the fault f_j is *covered* by BIDS B_l or not. For example, let us assume again that the circuit has faults, f_1, f_2 and f_3 and two BIDS, B_1 and B_2. If (f_1, f_2) are in the fanin cone of B_1 and (f_2, f_3) are in the fanin cone of B_2, then dictionary D_2 would contain entries 110 and 011 respectively for BIDS B_1 and B_2. During fault simulation process, we can have two sets of failing patterns, namely, (a) pattern set (say T_1) producing failure at PO/SO as well as BIDS and, (b) pattern set (say T_2) producing failure only at BIDS. Once we know the faulty patterns and BIDS outputs, we can encounter following four cases during fault localization procedure:

- **Case 1.** Outputs failing, BIDS passing: This situation is illustrated in Fig. 3(a). In this figure, B_0, B_1,...,B_{k-1} indicates the outputs of k BIDS. Since the PO and/or SO indicates failure whereas none of the BIDS indicates failure for all failing test patterns, it is evident that the failure (shown by 'X' in the figure) lies outside the region covered by BIDS. Therefore, the suspect fault list should not contain any fault that is covered by BIDS. In other words, if the $FB(b_l)$ denotes the set of faults covered by the BIDS B_l, then the new suspect fault list, SB, is given by,

$$SB = \{S\} - \{\cup FB(b_l)\}, l = 1\ to\ k \quad (2)$$

  where S is given by equation 1. Hence, by utilizing the BIDS outputs, it is possible to reduce the suspect fault list. An interesting observation is that, although the BIDS does not help in detecting the fault, it provides an indirect benefit by reducing the suspect fault list.
- **Case 2.** Outputs passing, BIDS failing: This situation is shown in Fig. 3(b). This case illustrates the situation where the failing test patterns are able to sensitize the hard-to-detect fault but the fault is not propagated to the SO or PO. Since one or more BIDS outputs indicate failure, it is evident that the failure lies inside the region covered by one or more BIDS. Therefore, the suspect fault list should contain the faults that are common between the failing BIDS. In other words, if $FB(bf_i)$ denotes the list of faults covered by the failing BIDS B_i then the new suspect fault list, SB, is given by,

$$SB = \{\cap FB(bf_i)\}, i = 1\ to\ u \quad (3)$$

  where u is the total number of failing BIDS. Note that in this case, it is not possible to detect/isolate any fault in a chip without BIDS circuitry whereas with BIDS, it is not only possible to detect but also localize the hard-to-detect faults.
- **Case 3.** Outputs failing, BIDS failing: This case is shown in Fig. 3(c). Here, we consider the situation where every failing pattern, showing a failure at PO/SO, also shows failures at the BIDS outputs. Since the PO/SO as well as one or more BIDS outputs indicate failure, it is evident that the failure lies inside the region covered by one or more number of BIDS. The initial suspect list contains the faults that are calculated by using fault dictionary D_1 (i.e. S). The suspect fault list is further reduced by observing that the final list should be the intersection of (a) the faults that are common between the failing BIDS and, (b) S. In other words, if the $FB(bf_i)$ denotes the list of faults covered by the failing BIDS B_i then the new suspect fault list, SB is given by,

$$SB = \{S\} \cap \{\cap FB(bf_i))\}, i = 1\ to\ u \quad (4)$$

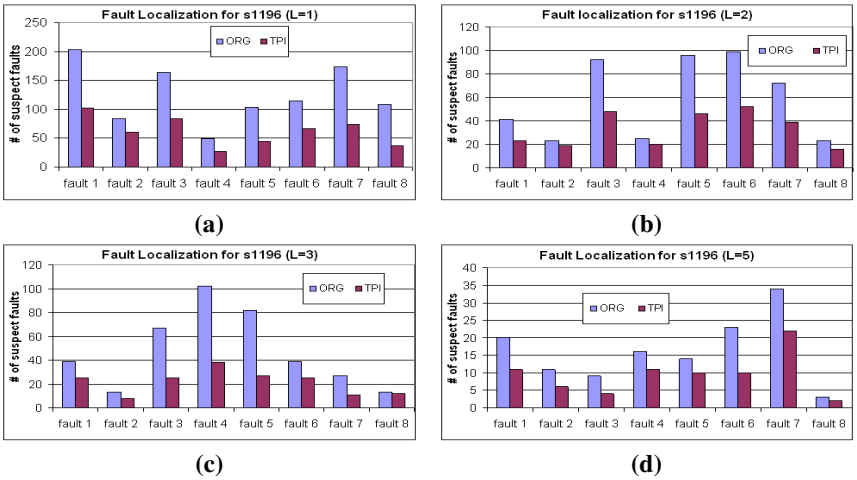

Fig. 4. Improvement in fault localization for s1196: (a) L=1, (b) L=2, (c) L=3, and (d) L=5.

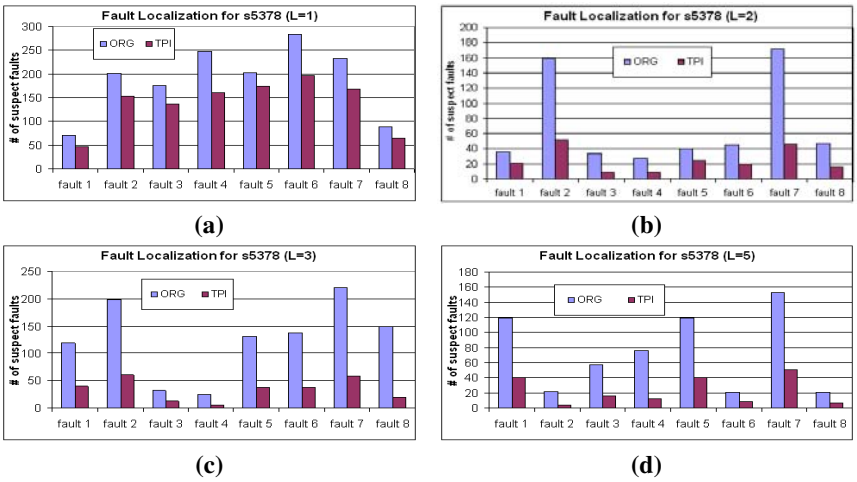

Fig. 5. Improvement in fault localization for s5378: (a) L=1, (b) L=2, (c) L=3, and (d) L=5.

where S is given by equation 1 and u is the total number of failing BIDS.

- **Case 4.** Combination of case 2 and 3: In this case, we consider the situation where we have both T_1 and T_2 set of failing patterns. It can be observed that if a fault is lying at the intersection of BIDS (case 2 and 3) is propagated to PO/SO for a test pattern, then it must also show failure on atleast one of the BIDS covering it.
  The suspect fault calculation for this case is similar to case 3. First of all, common faults between all BIDS that fail for T_1 and T_2 patterns are computed using D_2. The resulting suspect list is intersected with the suspect list of PO/SO found for T_1 patterns using D_1 and equation 1. This is mathematically expressed as,

$$SB = \{S\}_{\forall T_1} \cap \{\cap FB(bf_i))\}_{\forall T_1 and T_2} \qquad (5)$$

A situation where the fault does not produce failure at either PO/SO or BIDS are undetectable and therefore not considered for analysis. It should be noted that dictionary based diagnosis process consumes large storage for big circuits and researchers have proposed several techniques to compress the dictionary size [14]. Such compression techniques can also be applied in our technique if the circuit-under-test is large.

B. Fault localization procedure

The procedure for fault localization is shown in Table I. The netlist is read in step 1 and fault dictionaries D_1 and D_2 are created by performing fault simulation in step 2. For all faulty patterns T_1, that detect faults at PO/SO, the suspect fault list S is computed using equation 1 in step 3. Next, suspect fault list SB_1 is computed in step 4 for all failing BIDS for patterns in T_1 and T_2 using equation 3. Another fault list SB_2, that contains faults to be discarded for case 1, is created in step 5. In step 6, the procedure decides the case to be considered based on the faulty patterns in T_1 and T_2 and the corresponding BIDS outputs. This can be done easily in the following manner: (a) if T_1 is not NULL (i.e. faults are present at PO/SO); no faults are observed at BIDS outputs for T_1, and, T_2 is NULL, then case 1 is considered; (b) if T_1 is NULL (i.e. no faults observed at PO/SO) but T_2 is not NULL (i.e. faults are present at BIDS outputs) then it is case 2; (c) if

T_1 is not NULL and faults are present at the BIDS outputs for T_1 but T_2 is NULL, then case 3 is considered; (d) if T_1 and T_2 are both not NULL and faults are present at the BIDS outputs for T_1, then case 4 is considered. Once the cases are decided, steps 7 through 10 are performed to return appropriate values of suspect fault list using equations 2 through 5.

V. Results

The proposed test methodology is verified by finding a set of test points (where the BIDS will be inserted) for ISCAS89 benchmark circuits using the test point insertion algorithm [6] and treating them as pseudo primary outputs. The test circuits are mapped to the LEDA standard-cell library and transistors are modeled in TSMC 180nm technology. The area overhead due to test point insertion is limited to 4% by setting the maximum number of test points to 20.

We have used SIGMA [15] to compare the fault localization in the original (ORG) and the modified (TPI) circuit. The segment length L has been chosen to be 1, 2, 3 and 5. Note that, the results for $L = 1$ with non-robust fault propagation corresponds to the *transition delay fault* model. A pass/fail based fault dictionary (D_1) is created for the original circuit by application of 10,000 random test patterns. Now a set of 50 randomly chosen faults are inserted in the circuit and fault simulation is performed using SIGMA to identify the faulty test patterns. The suspect fault list (S) is obtained by using equation 1. For the modified circuit (BIDS inserted), we create two fault dictionaries: D_1 and D_2. The same set of random faults are once again inserted to this new circuit and the faulty patterns T_1 and T_2 are identified by fault simulation. The faulty patterns and the BIDS outputs are fed to the *LocalizeFaults* routine, which outputs SB as a set of new suspect fault list. The number of suspect faults, for original and modified benchmarks em s1196 and *s5378*, for a set of 8 random faults (out of 50), are provided in Fig. 4 and Fig. 5, respectively. The results for other benchmark circuits are not presented due to space limitation. It is evident from these plots that there is significant improvement in fault localization (i.e. reduction in the size of suspect fault list) for transition delay fault ($L = 1$) and segment delay faults of lengths 2, 3 and 5. Note that the reduction in suspect fault list is very small (e.g., fault-8 in Fig. 4(c)) for some faults. This may be due to the positioning of BIDS which is tuned for achieving better fault coverage. However, the test point selection algorithm [6] can also be tuned to achieve better fault diagnosis.

We provide average improvements (for the set of 50 randomly inserted faults) in fault localization for segment lengths of 2, 3 an 5 in Fig. 6. From this figure, it can be observed that as much as 82% average improvement can be obtained for benchmark *s838* for segment length of 2 ($L = 2$). Similarly, significant improvements can also be noted for the other segment lengths. Hence, inserting BIDS in the circuit not only improves the test time/coverage but also improves the fault localization capability by reduction of suspect fault list. From our simulations on segment delay fault model, it is also evident that the proposed methodology can be useful

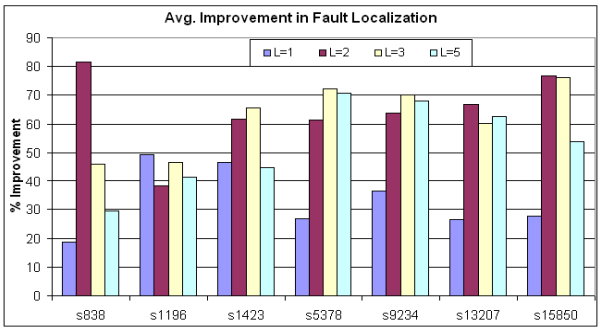

Fig. 6. Average improvement in fault localization for different segment lengths (L = 1, 2, 3 and 5)

in diagnosis of spot- as well as distributed delay defects. Further improvements can be obtained by tuning the test point selection procedure [7] to maximize localization. Note that by considering BIDS outputs for diagnosis, we increase the dictionary size. However, the number of BIDS are very limited ($\sim$10-20). Therefore, the increase in dictionary size is expected to be small.

VI. Conclusions

This paper presents a fault diagnosis methodology for test-per-scan BIST using on-die delay sensor. A fault dictionary based localization methodology has been presented to utilize the sensor outputs efficiently. Simulation results on transition and segment delay fault models show that the delay sensors can be inserted at strategic points to improve detection as well as efficient localization of delay faults in scan-based BIST.

References

[1] EEDesign Article, Delay-fault testing mandatory, Dec, 2002, http://www.eedesign.com/story/OEG20021204S0029.

[2] M. L. Bushnell et al., Essentials of electronic testing for digital, memory & mixed-signal VLSI circuits, Kluwer, 2000.

[3] K. Heragu et al., Segment delay faults: A new delay fault model, *VTS*, pp. 32-39, 1999.

[4] S. Tragoudas et al., Testing for path delay faults using test points, *DFT*, pp. 86-94, 1999.

[5] M. Nakao et al., Low overhead test point insertion for scan-based BIST, *ITC*, pp. 348-357, 1999.

[6] S. Ghosh et al., A novel delay fault testing methodology using low-overhead built-in delay sensor, *TCAD (accepted)*.

[7] A. Raychowdhury et al., A novel delay fault testing methodology using on-chip low-overhead delay measurement hardware at strategic probe points, *ETS*, pp. 108-113, 2005.

[8] A. E. Stevens et. al., A time-to-voltage converter and analog memory for colliding beam detectors, *JSSC*, pp. 1748-1752, 1989.

[9] Y. Zheng et. al., On-chip oscilloscopes for noninvasive time-domain measurement of waveforms in digital integrated circuits, *TVLSI*, pp. 336-344, 2003.

[10] R. Datta et al., On-chip delay measurement for silicon debug, *GLSVLSI*, pp. 145-148, 2004.

[11] C. Su et al., All digital built-in delay and crosstalk measurement for on-chip buses, *DATE*, pp. 527-531, 2000.

[12] I. Pomeranz et al., On dictionary-based fault location in digital logic circuits, *IEEE Tcomp*, pp. 48-59, 1997.

[13] I. Bayraktaroglu et al., Gate level fault diagnosis in scan-based BIST, *DATE*, pp. 376 - 381, 2002.

[14] C. Liu et al., Compact Dictionaries for Fault Diagnosis in Scan-BIST, *TCOMP*, pp. 775-780, 2004.

[15] K. Heragu et al., SIGMA: A simulator for segment delay faults, *ICCD*, pp. 502-507, 1996.