Scan-Chain Intra-Cell Aware Testing

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TETC.2016.2624311, IEEE Transactions on Emerging Topics in Computing

Scan-Chain Intra-Cell Aware Testing

A. Touati, Student Member, IEEE, A. Bosio, Member, IEEE, P. Girard, Fellow, IEEE, A. Virazel, Member, IEEE, P. Bernardi, Member, IEEE, M. Sonza Reorda, Fellow, IEEE, and E. Auvray

Abstract—This paper first presents an evaluation of the effectiveness of different test pattern sets in terms of ability to detect possible intra-cell defects affecting the scan flip-flops. The analysis is then used to develop an effective test solution to improve the overall test quality. As a major result, the paper demonstrates that by combining test vectors generated by a commercial ATPG to detect stuck-at and delay faults, plus a fragment of extra test patterns generated to specifically target the escaped defects, we can obtain a higher intra-cell defect coverage (i.e., 6.46% on average) and a shorter test time (i.e., 42.20% on average) than by straightforwardly using an ATPG which directly targets these defects.

Index Terms— fault simulation, intra-cell defect, scan-chain testing, test quality.

1 Introduction

THE endless advance of semiconductor technologies results in an increasing complexity of digital circuits. Designing and manufacturing smaller, faster, cheaper and less power consuming devices are the main challenges in semiconductor industry. The reduction of transistor size and the latest packaging technology (i.e., System-On-a-Chip, System-In-Package, Through Silicon Via 3D Integrated Circuits) allow the semiconductor industry to satisfy the latest challenges. Although producing such advanced circuits can benefit users, the manufacturing process is becoming finer and denser, making chips more prone to defects. In modern deep submicron technologies, systematic defects are becoming more frequent than random defects [1].

Today, systematic defects appear not only in the cell interconnections, but also inside the cell itself (*intra-cell defects*). In the literature, many works prove that these defects can escape classical test solutions. In [2] a statistic carried out over 1 million tested devices showed that a significant number of defects appear inside standard cells (i.e., *intra-cell defects*). In [3][3][5], it is shown that these defects cannot be detected by using approaches based on classical fault models (i.e., stuck-at, transition, bridging). Despite the fact that previous works already proved that classical test sets lead to a low coverage of intra-cell defects, none of them deeply investigated the issues related to scan chain testing in the presence of intra-cell defects.

Usually, scan chain testing is performed by applying a so-called *shift test* [6]. A toggle sequence "00110011…" is shifted into the scan chain and values appearing at the other extreme of the chain are checked. The applied sequence produces all possible transitions at the scan-input of each scan flip-flop. In this way, the correctness of the shift operations is verified and the presence of possible stuck-at and transition faults in the scan flip-flop interconnections can be detected. Moreover, the work in [7] shows that the above sequence can only cover the intra-cell defects affecting the scan-path of each scan flip-flops.

Despite the fact that the shift test is widely used in practice, authors in [8] prove that some intra-cell defects can escape because the scan chain test is applied only when flip-flops are in test mode. The intra-cell defect coverage is indeed too low. It is thus mandatory to analyze and quantify the intra-cell defect escapes to eventually develop meaningful test solutions.

Authors in [9] presented a Cell-Aware (CA) production test results on a 130nm technology library composed of 216 cells. This work focused in the full application of the CA methodology, which includes slow-speed and at-speed CA tests for bridges, opens and transistor defects. Authors showed that during different tests, at different temperatures, CA tests detect unique failing parts.

In [3], authors evaluated the effectiveness and the quality of the CA test patterns, which are generated to reach the maximum achievable defect coverage, by comparing them with the state-of-the-art Stuck-at and Transitions fault patterns defect coverage. This evaluation is carried out on 10 industrial multi-million gate designs and the defect coverage gain was estimated for the complete design and for each standard cell type as well. In our previous work [10], we presented a preliminary analysis of intra-cell defects affecting the scan chain. In this paper, we first show that the percentage of intra-cell defects escaping the standard tests is indeed very high (up-to 60%). Then, conversely to [3] and [9] , we propose a high quality test solution (based on combining different test sets generated by a commercial ATPG targeting different fault models) achieving high intra-cell defect coverage. Finally, we compare the defect coverage of the proposed

- *Aymen Touati, Alberto Bosio, Patrick Girard, Arnaud Virazel are with the Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier (LIRMM), a research laboratory related to "Université de Montpellier" and the "Centre National de la Recherche Scientifique (CNRS)", Montpellier, France. E-mail: {atouati, bosio, girard, virazel}@lirmm.fr*
- *Matteo Sonza Reorda and Paolo Bernardi are with the Department of Control and Computer Engineering of Politecnico di Torino, Italy. E-mail: {matteo.sonzareorda, paolo.bernardi}@polito.it*
- *Etienne Auvray is with STMicroelectronics in Grenoble, France. E-mail: etienne.auvray@st.com*

xxxx-xxxx/0x/$xx.00 © 200x IEEE Published by the IEEE Computer Society

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TETC.2016.2624311, IEEE Transactions on Emerging Topics in Computing

test solution with the one produced by an available commercial tool when directly targeting intra-cell defects [3]. In fact, the novelty of our test solution is that the evaluation of Stuck-at and Transitions fault test patterns is provided with respect of the overall defect database and not only with respect to the Cell-Aware detectable defects as mentioned in [3]. Experimental results show that on average our solution outperforms the latter in terms of both defect coverage and test length. To the best of our knowledge, this is the first work which explicitly targets the intra-cell defects affecting the scan flip-flops. It aims at improving the defect coverage by combining together different test sets.

The paper is organized as follows. Section 2 presents the basics of scan chain design and test, while Section 3 sketches the overall flow. The defect characterization is briefly reviewed in Section 4. In Section 5 the methodology for scan chain intra-cell aware test is presented. Experimental results and defect coverage analysis are presented in Section 0. Conclusions are given in Section 7.

2 Scan Chain Test

This section presents the basics of scan chain design & test. Fig. 1 shows the well-known MUX-scan flip-flop library cell architecture. It is composed of a D Flip-Flop (DFF) plus a multiplexer. The input signal Scan Enable (SE) allows to select between the **scan mode** (SE = '1') and the **functional mode** (SE = '0'). In scan mode, the flip-flop stores the value coming from the Scan Input (SI), while in the functional mode it stores the value coming from D.

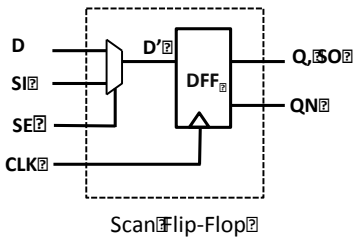

Fig. 1. MUX-scan flip-flop

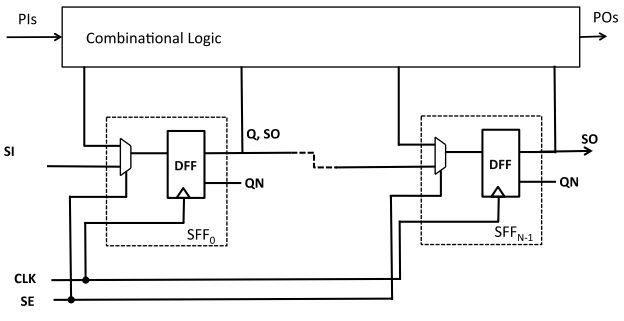

Fig. 2. Full Scan Architecture

Fig. 2 depicts the classical full scan architecture. Here, each scan flip-flop corresponds to one element of the scan chain (from SFF_0 to SFF_{N-1}). The scan chain is controlled by the SE signal and latched by the clock (CLK).

Scan designs are usually tested in two steps. First, a **Scan Chain Test** is applied. A single test pattern composed of a sequence of alternated 00 and 11 bits ("00110011…") is shifted in and out of the scan chain. This sequence produces all possible transitions on the input of each scan flip-flop. In this way, the ability of the scan chain to support the shift operations is verified and the presence of possible stuck-at and transition faults in the scan flip-flop interconnections is detected. If the scan chain test succeeds, the scan design test goes through the second step, corresponding to the **Logic Test**. It aims at testing the combinational logic in between the scan chains. It resorts to test patterns generated by an ATPG tool targeting several fault models (e.g., Stuck-at Faults, Transition Faults). Logic test is applied and the output of the combinational blocks is latched by switching scan flip-flops from test to functional mode [6].

As reported in [7], scan chain test is efficient in testing intra-cell defects affecting the D flip-flops. However, since the scan flip-flops always work in test mode during the scan chain testing, some defects in the scan flip-flops cannot be detected since they can be sensitized only in functional mode. Experimental evidence of the correctness of this hypothesis has been provided in [8], whose authors show that some intra-cell defects actually escape. Physical Failure Analysis proves that those defects affect the multiplexer controlling the scan flip-flop input.

The main contribution of this work is the analysis of the test quality w.r.t. intra-cell defects affecting the multiplexer of scan flip-flops. We first evaluate the percentage of intra-cell defects detected when scan chain test is performed. Secondly, we compute the intra-cell defect coverage when logic test is performed. The latter is done by applying three types of test sets, developed for the Stuck-at fault model and for the Transition Delay Fault model, adopting the Launch-Off-Capture (LOC) and the Launch-Off-Shift (LOS) schemes [6]. Finally, we propose to merge together the above test sets, thus achieving a high test quality for the scan chain (*intra-cell defect aware testing*).

3 Overall Flow Of The Proposed Approach

Fig. 3 sketches the overall flow of the proposed approach, which is composed of four steps. The first one is the **technology library characterization** (i.e., **the dashed line box**). In this step, an automatic tool extracts all possible defect locations for every library cell. Then, for each location, a defect injection campaign is executed. It exploits a transistor-level simulator to determine the faulty behavior of the injected defect. The result is the **Defect Database**. Please note that this step is applied only one time for a given technology library.

The second step is the **Logic Simulation Step**. Two inputs are required: (i) the circuit under evaluation and (ii) the test sets generated by the ATPG. For each simulated test set it stores the Flip-Flops activity.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TETC.2016.2624311, IEEE Transactions on Emerging Topics in Computing

The third step is the Defect Coverage Estimator. Again two inputs are used: (i) the previously computed defect database and (ii) the Flip-Flops activity. For each test set, it computes the defect coverage.

The last step is the **Analysis & Learning**. It aims at merging together the graded test sets by applying an in-depth analysis. The goal is to obtain a single high quality test set. Two inputs are required: the test sets and the defect coverage determined in the second step.

Please note that the main focus of this paper is on the intra-cell defect grading and on the defect coverage analysis steps. The latters can be applied on any technology library. However, for the sake of readability, we present in the next section the main guidelines about the considered defects and the location extraction performed during the technology library characterization.

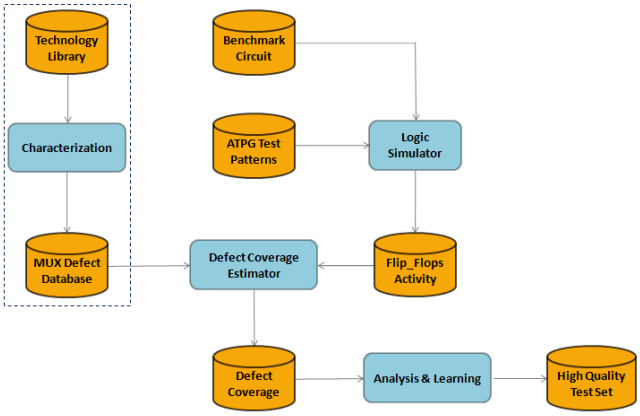

Fig. 3. Overall Flow

4 Defect Characterization

The defect characterization for a given technology library is done by means of a defect injection campaign. Several papers described this process [3][4][5]. In this work we exploit the approach published in [5]. In this section, we recall the main concepts of [5] for the sake of readability.

For each library cell, we must determine all the possible defect locations (i.e., where a defect can appear) and the type of defect. In our work the location can be any cell internal net. As already described in previous work [5], cell layout analysis can be used to identify the realistic defect locations. Then, for each realistic defect location, defect injection is performed to evaluate if the behavior induced by the injected defect is covered or not by the applied set of stimuli. Finally, the defect database is created. Note that any transistor-level simulator can be used to perform this analysis.

Injected defects are modeled in the transistor-level domain as: (i) an unexpected connection between two nets associated to a specific resistance value (*resistive-short*), (ii) an unexpected resistance value on a given net (*resistive-open*) [11]. Injected defects can lead to either static faults (i.e., stuck-at faults) or dynamic faults (i.e., delay faults). To sensitize the static faults test patterns are made with only one test vector, while for the dynamic ones we have to apply test patterns that contain a couple of test vectors in order to force a transition that actually sensitizes and propagates the fault.

Since we target library cells, we can resort to an exhaustive test pattern generation in order to ensure that all possible defects are sensitized and observed. As discussed above, we consider couples of test vectors in order to be sure that even dynamic faults are sensitized. We resort to the following notation, derived from [12]:

C0: this symbol corresponds to the couple “00”;

C1: this symbol corresponds to the couple “11”;

R1: this symbol corresponds to the couple “01”;

F0: this symbol corresponds to the couple “10”.

Four symbols are used, thus for a given library cell having n inputs, we have to simulate all possible 4^n inputs combinations. Once again, since we target library cells the number of input signals is low (i.e., four on average, up to 10 for the most complex cells) thus making the exhaustive generation and simulation feasible. The created database is a simple table having one row per input pattern. For each test pattern, the table stores the list of detected defects as shown in Table 1.

TABLE 1
Defect Database

IP	Detected Defects
Pattern1	Defects list
…	…
PatternN	Defects list

Let us resort to an example to show the process of library characterization. Fig. 4 gives the schematic view of a two-ways multiplexer (MUX21) from an industrial 90nm technology library.

Thanks to the layout analysis, we identify R10 as a realistic defect location (red connection in Fig. 4). Then, as a defect type we consider a resistive short, thus the resistance value is set to 1Ω. Finally, for each possible input pattern we run a SPICE simulation. Simulations are done by fixing the temperature at 25°C and the Vdd at 1V.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TETC.2016.2624311, IEEE Transactions on Emerging Topics in Computing

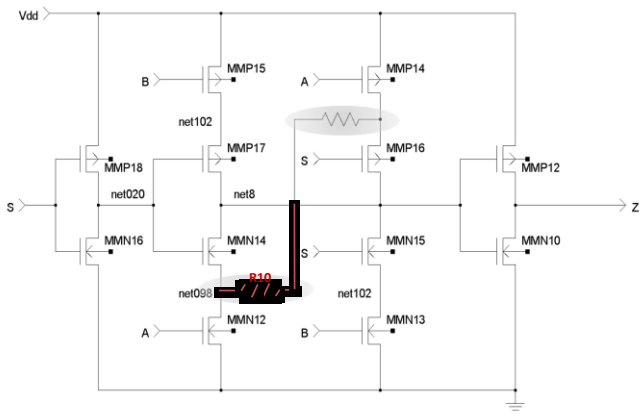

Fig. 4. Schematic view of a MUX21

TABLE 2
MULTIPLEXER DEFECT DATABASE EXAMPLE

Input A, input B, input S	Detected Defects
C1,C0,C1	R10, …
R1,F0,R1	R10, …
…	-

Results are collected in the defect database, as shown in Table 2. Among all the possible 64 patterns (i.e., 4^3) only two of them detect the injected defect: "C1,C0,C1" and "R1,F0,R1". This happens because the injected defect leads to a static fault, thus we can detect it either when we apply stable values ("C1,C0,C1") or when we apply transitions ("R1,F0,R1"). In this example, we omit to put in the table the remaining patterns and remaining defects only for the sake of readability. The same process should be repeated for all other defect types and defect locations.

Please note that the proposed methodology is general (scalable) and can be applied to any technology node.

5 INTRA-CELL DEFECT GRADING

This section describes the methodology followed during the intra-cell defect grading process. Once again, the target is the defect affecting the multiplexer of the scan flip-flop. The intra-cell defect grading is performed by exploiting a classical serial fault simulation technique [6]. After the library characterization, we identified 162 defects affecting the multiplexer of the scan flip-flop cell. Knowing the overall number of intra-cell defects affecting the multiplexers, we can define the Defect Coverage (*DC*) metric as follows:

$$DC = \frac{\#DD}{162 \times \#N} \quad (1)$$

where:

- *#DD*: is the number of detected defects;
- *#N*: is the number of flip-flops in the target circuit.

For each circuit we simulate two types of test: **scan chain** and **logic** tests (see Section 2). For each type of test, different test sets are considered. The details are given in the following subsections.

5.1 Scan Chain Test

The scan chain test set is composed of *load* and *unload* operations. All scan flip-flops are set into scan mode (SE = '1') and a single pattern is shifted-in (*load*) and shifted-out (*unload*) through the scan chain.

In our work, we consider several test sets. Each set is composed of a single test pattern to be applied through load and unload operations. Applied patterns are the ones generated by the commercial ATPG tool used for our experiments: "0011", "0101", "1000" and "0111". Please note that patterns are repeated several times depending on the scan chain length. Each pattern guarantees that each scan flip-flop is tested under the application of the same stimuli. For example, pattern "0011" produces the following stimuli: "00", "01", "11" in all flip-flops. Those values are applied through the Scan In (*SI*) input of each scan cell. On the other hand, the logic value applied to the input *D* depends on the combinational logic of the circuit. Thus, this value can be different for every scan cell. Since we are looking for the intra-cell defects affecting the multiplexer, the knowledge of the value of the input *D* is mandatory.

We define the activity of the scan flip-flop *i* ($0 \le i \le N$-1, where *N* is the length of the scan chain) at a given clock cycle *j* as the following 4-tuple:

$$Activity_{i,j} = \{D, SI, SE, Q\} \quad (2)$$

where *D*, *SI*, *SE* are the inputs and *Q* is the output of the flip-flop *i*. Please note that each element of the 4-tuple is encoded by using the symbols of [12], as described in Section 4. Thus, the activity at the clock cycle *j* encodes the couple of logic values applied during the clock cycles *j*-1, *j*.

In Fig. 5, we show the basic principle. Since we apply the scan chain test, all scan flip-flops are set to scan mode (red connections in Fig. 5. Then, for each scan flip-flop *i* and for each target defect we simulate a test pattern. During the simulation, at each clock cycle *j* we access the Defect Database by using the information stored in the related $activity_{i,j}$ tuple as defined by (2) in order to determine if the target defect is sensitized or not. If yes, the value stored in the flip-flop *i* is inverted. At the end of the simulation, we compare the unloaded values with the golden ones to determine if the target defect has been detected or not. The simulation length is equal to 2**N* clock cycles. However, in practice, *N*+4 clock cycles are sufficient to detect all possible detectable faults.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TETC.2016.2624311, IEEE Transactions on Emerging Topics in Computing

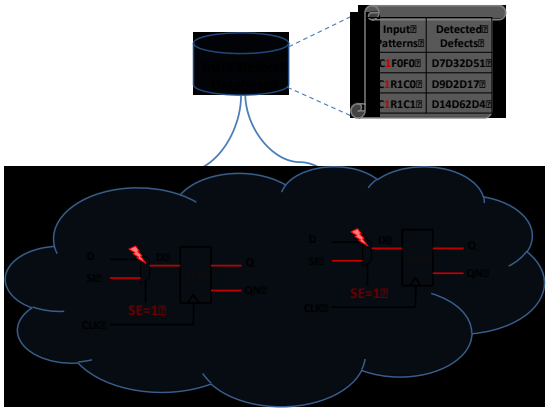

Fig. 5. Scan chain test intra-cell defect grading

5.2 Logic Test

During the logic test, several test patterns are applied by using the scan chain, which is repeatedly switched from scan to functional mode. Logic test patterns are composed of *load*, *launch*, *capture* and *unload* operations (thus testing basically the combinational logic). The *load* and *unload* operations correspsond to serial shift-in and -out of a test vector (as described in the previous section), while the *launch* and *capture* correspond to the application of the vector to the combinational logic and to the capture of the test response. Please note that the *launch* operation is only exploited during the test of delay faults (e.g., transition faults), while the *capture* is always used, independently on the target fault model.

Since intra-cell defects can lead to both static and dynamic faults (see Section 4), we resort to three widely used test sets: Stuck-at fault test set, Transition fault test set under the Launch-Off-Capture (LOC) scheme and Transition fault test set under the Launch-Off-Shift (LOS) scheme [6]. In this case, we aim at quantifying the intra-cell defects detected when the scan flip-flops are in scan mode (SE = '1') as well as in functional mode (SE = '0').

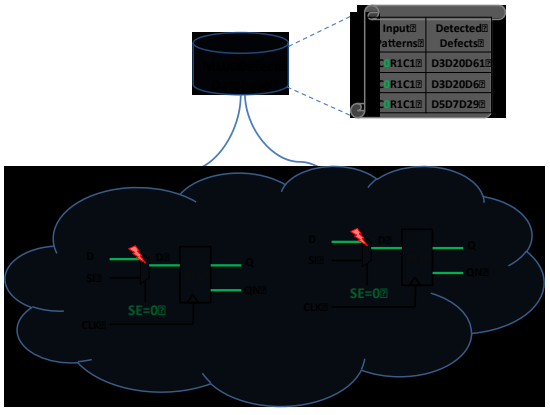

Fig. 6. Logic test intra-cell defect grading

In Fig. 6 we show the principle of the logic test intra-cell defect grading. In this case, we have to consider the activity$_{i,j}$ not only during the load and unload operations, but also during the launch and capture (i.e., in functional mode) phases. The latter depends on the simulated test set, as depicted in Fig. 7.

The figure reports waveforms related to the application of test sets: a) stuck-at test set, b) transition test set, further divided into the one adopting the LOC scheme, and the one adopting the LOS scheme. For each one we define the functional window as the time interval when SE = '0'. This includes the capture edge for stuck-at and LOS test sets, while it includes both launch and capture edges for the LOC test set. Except for the functional window, the defect grading for the logic test is exactly the same as described in the previous section. The simulation length is equal to [(N + 1)*T +N] clock cycles for the stuck-at and LOS test sets, where n is the number of flip-flops composing the scan chain and T is the number of patterns composing the simulated test set. The term '+ 1' is added to include the capture clock cycle during the functional window. For the LOC test set, the simulation length is equal to [(N + 2)*T+N] clock cycles because in this case the functional window contains two clock cycles, corresponding to the launch and the capture operations.

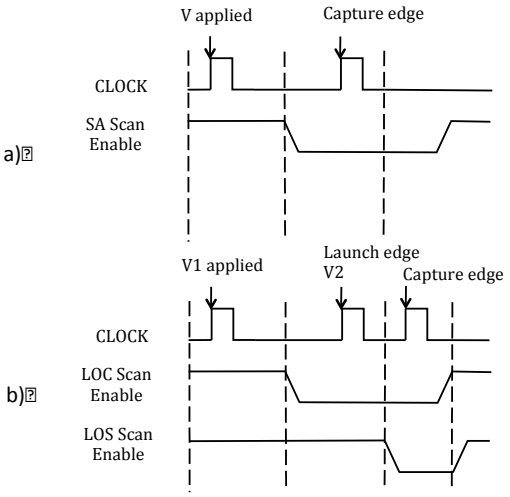

Fig. 7. Logic test "functional window", corresponding to the period when Scan Enable = 0

6 Results and Analysis

In this section, we present the results corresponding to the intra-cell defect grading process. Then, we present what it is possible to learn from the collected data in order to obtain a meaningful test solution. Finally, we compare the results achieved with the proposed test solution with the one from a commercial tool.

6.1 Defect grading results

We performed several experiments on the full-scan version of ITC'99 benchmark circuits. All circuits were synthesized using an industrial 90nm technology library, and for each circuit one scan chain was inserted during syn-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TETC.2016.2624311, IEEE Transactions on Emerging Topics in Computing

TABLE 3
DEFECT COVERAGE EVALUATION FOR LOGIC TEST

Ckt	#Flip-Flops	Logic Test								
		SAF			TF-LOC			TF-LOS		
		#p	FC%	DC%	#p	FC%	DC%	#p	FC%	DC%
b01	5	20	100	29.38	18	69.91	53.21	24	72.22	50.37
b02	4	14	100	19.60	13	82.00	42.75	16	90.00	46.45
b03	30	26	100	40.08	32	92.99	48.07	35	89.29	55.66
b04	66	46	100	23.18	23	65.85	62.69	57	79.23	65.29
b05	34	76	99.93	50.84	95	53.97	75.00	90	60.53	68.70
b06	8	14	100	25.54	12	69.82	49.00	14	88.29	28.78
b07	41	52	100	23.56	89	81.24	76.47	76	90.01	66.49
b08	21	51	100	24.37	33	72.73	67.40	53	78.33	64.10
b09	28	33	100	23.24	27	87.90	45.85	40	96.11	54.01
b10	17	47	100	25.75	32	66.63	66.41	56	91.57	65.00
b11	30	113	100	35.58	136	77.69	77.98	150	91.10	68.35
b12	119	93	100	23.49	260	88.27	80.47	157	92.04	68.66
b13	45	39	99.94	23.61	59	79.26	61.70	46	91.49	60.38
b14	215	305	99.99	39.66	284	84.87	81.53	336	98.27	68.96
b15	415	465	99.74	67.15	1,082	80.41	81.89	743	93.46	69.14
b17	1,311	433	97.83	51.13	1,062	83.84	81.88	861	92.22	69.08
b18	2,754	507	98.23	64.10	1,814	80.90	81.90	1230	91.03	69.09
b20	429	305	99.97	59.20	587	91.36	81.82	391	98.53	68.96
b21	429	315	99.92	65.85	634	91.80	81.89	397	98.53	68.94
b22	611	279	99.89	57.57	571	90.61	82.10	397	97.96	68.95
Average		-	97.77	38.63	-	79.60	69.45	-	89.00	62.27

thesis[1]. As mentioned in Section 5, during the library characterization we identified 162 defects affecting the multiplexer of the scan flip-flop cell. Thus, for each circuit the total number of defects is 162*N, where N is the number of flip-flops.

Table 3 summarizes the defect grading results when logic test patterns are evaluated: results related to SAF, TF-LOC and TF-LOS patterns are reported in columns 3, 4 and 5, respectively. The first column reports the name of the ITC'99 full-scan circuit. The corresponding number of scan flip-flops is reported in column 2. For each test set we report in the sub-columns the test length in terms of number of patterns (#p), the achieved fault coverage (FC%) and the intra-cell defect coverage (DC%).

As expected (and already mentioned and explained in a previous work [10]), we obtained a large gap between fault and defect coverage. In fact, the gap can reach up to 43% for the b09 circuit if we consider LOC patterns and 40% for b02 if we consider LOS patterns. More in detail, we obtained that on average SAF patterns achieved 38.63% of defect coverage, while TF-LOC patterns reached 69.45% and TF-LOS patterns reached 62.27%.

The difference in defect coverage between SAF and TF test patterns can be easily explained. TF test patterns (either LOC or LOS) lead to a greater number of signal transitions compared to SAF ones. Thus, the number of covered defects is higher because both static and dynamic faults are detected. Conversely, SAF test patterns mainly detect static faults, thus missing the dynamic ones. Please note that this depends not only on the applied test set, but also on the circuit structure and on how it propagates the fault effects. An example is the circuit b15, for which the defect coverage achieved by the SAF test patterns is comparable than the one obtained by TF- LOS patterns.

Table 4 summarizes defect grading results when scan chain test patterns are applied. The first column reports the name of the ITC'99 full-scan circuits. The corresponding number of flip-flops is reported in column 2. For each graded scan test pattern, we reported the achieved intra-ell defect coverage (DC%) in columns 3, 4, 5 and 6, respectively.

The first comment refers to the difference in defect coverage average between scan chain test (i.e., by considering the average of all the scan chain tests defect coverage) and logic test (i.e., by considering the average of all the logic test defect coverage), which is about 47%. The observed gap can be explained by the fact that in logic test the scan chain works in both functional and scan modes, thus leading to a higher defect coverage.

TABLE 4
DEFECT COVERAGE EVALUATION FOR SCAN CHAIN TEST

Ckt	#Flip-Flops	Scan Chain Test			
		"0011"	"0101"	"1000"	"0111"
b01	5	11.36	5.09	11.36	12.10

[1] Fault grading results would clearly be the same if multiple scan chains would have been adopted.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TETC.2016.2624311, IEEE Transactions on Emerging Topics in Computing

b02	4	9.88	11.11	9.88	11.11
b03	30	12.18	12.18	12.18	11.85
b04	66	3.22	3.25	3.17	3.25
b05	34	12.20	12.16	12.20	12.16
b06	8	10.34	9.88	10.80	9.72
b07	41	12.35	12.35	12.35	12.20
b08	21	12.35	12.35	12.35	12.22
b09	28	12.35	12.35	11.99	12.35
b10	17	12.06	12.35	11.18	11.76
b11	30	12.18	12.18	11.52	11.69
b12	119	3.16	3.18	3.13	3.18
b13	45	12.15	12.13	12.15	12.10
b14	215	3.13	3.14	3.11	3.14
b15	415	12.35	12.33	12.35	12.33
b17	1,311	12.34	12.34	12.34	12.34
b18	2,754	3.09	12.34	12.34	12.34
b20	429	3.11	3.11	3.10	3.11
b21	429	3.11	3.11	3.10	3.11
b22	611	3.10	3.10	3.10	3.10
Average		8.80	9.00	9.19	9.26

TABLE 5
LOGIC TEST OVERALL DEFECT COVERAGE

Ckt	LT[2] DC%	LT&ST[3] DC%
b01	68.52	68.52
b02	51.85	51.85
b03	71.91	71.91
b04	76.52	76.52
b05	81.70	81.70
b06	54.48	54.48
b07	79.66	79.66
b08	73.66	73.66
b09	67.95	67.99
b10	76.72	76.72
b11	80.21	80.21
b12	81.02	81.02
b13	68.68	68.70
b14	81.90	81.90
b15	85.49	85.49
b17	84.06	84.06
b18	85.54	85.54
b20	84.92	84.92
b21	85.64	85.64
b22	84.74	84.74
Average	76.26	76.26

[2] Logic Test
[3] Scan chain Test

The second comment is about the fact that the defect coverage varies depending on the circuit. This result is somewhat expected since during the shift in/out operations the output of each scan flip-flop is propagated through the combinational logic. In some cases, these propagations may prepare the right logic value on the D-input at the right moment to get the missed input combinations. Thus, for some circuits the above values can be effective for defect coverage (for example for the circuit b15) while this does not happen for some others (circuit b18).

Finally, results in Table 4 do not vary among the applied test patterns. Only few exceptions can be noted for the b01 and b18. Again, here it is a matter of circuit structure and how test patterns are propagated trough the combinational logic.

6.2 Learning

During the learning step, we aim at determining the capability of one test set to cover a specific set of defects that is actually uncovered by another test set. The goal of this analysis is to merge together different sets of test patterns to increase the overall defect coverage.

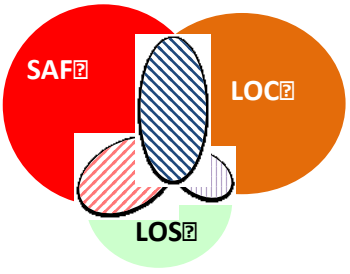

Fig. 5. Defect coverage capabilities of the different test sets

First of all, we compute the cumulative results for the three logic test sets: SAF, TF-LOC and TF-LOS as qualitatively reported in Fig. 8. Then, we further consider the contribution of the scan chain test to the overall defect coverage.

Table 5 quantifies the overall defect coverage by considering the union among the three logic test sets (second column). The obtained defect coverage increases for all the circuits; on average we detect about 26% more defects than considering a single test set. The third column reports the defect coverage achieved when considering the contribution of the scan chain test. From the reported data, it is easy to see that the scan chain test contribution is negligible (0.004% on average). This result is explained by the fact that in logic test, the scan chain works in both scan and functional modes. Therefore, most of defects covered by the scan chain test are indeed already covered by the logic test during the *load*, *unload* and *capture* operations.

After the learning phase, we are able to propose a test solution composed of different test sets. However, the maximum value of the defect coverage (achieved for the b21 circuit) is only 85.64%, which could not be enough. Thus, we further investigate the possibility to eventually

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TETC.2016.2624311, IEEE Transactions on Emerging Topics in Computing

increase the defect coverage by generating a dedicated test set. We exploit a commercial intra-cell aware ATPG tool to generate a test pattern set able to detect escaped defects. Each defect is defined with the following information:

- It is associated to a specific scan flip-flop i in the circuit under test;
- It requires the knowledge of the logic values to be applied to the scan flip-flop inputs to sensitize the faulty behavior;
- It requires the knowledge of the fault effect.

The above informations are extracted from the defect database created during the technology library characterization. The intra-cell aware ATPG exploits the above information to generate a test pattern able to sensitize the defect (i.e., it applies the required logic values to the scan flip-flop i inputs) and to propagate its fault effect to reach an observable point (i.e., either a primary output or a scan cell).

More in details, we opted for the User Defined Fault Model (UDFM) [13], which basically extends the natively supported faults models (e.g., stuck-at and transition faults) by adding constrainted pins/nets of a given library cell. One of the restrictions we could encounter when applying the UDFM is that we have to target static and delay faults separately since they cannot be handled together in one pattern generation step.

Let us refer to a small example that tests a cell internal bridge fault of the multiplexer considered in Fig. 1.

```
UDFM {
    version : 1;
    UdfmType ("Special_delay")
    {
        Cell ("MUX21")
      {
       Fault ("myFlt-D")
            {
            Test {
                    DelayFault {"Q" : 1;}
                    Conditions {"D" : 10; "SI" : 00; "SE" : 00;}
             }
      }
```

Fig. 6. Delay Fault UDFM Cell Example

Fig. 9 reports an example of UDFM modeling specifies the defect of Fig. 1. The defect behavior corresponds to a delay fault. The required activation conditions are specified on the input pins ("D, SI, SE"), the output pin ("Q") is used for observing the fault effect. It specifies the requirements for the test where the "D" input transitions from '1' to '0' while observing that the "Q" output is faulty if it remains at '1'.

```
UDFM {
    version : 1;
    UdfmType ("Special_delay")
    {
        Instance ("\stato_reg[1]")
        {
            Fault ("myFlt-0")
            {
                Test {
                    DelayFault {"Q" : 1;}
                    Conditions {"D" : 10; "SI" : 00; "SE" : 00;}
                }
            }
```

Fig. 7. Delay Fault UDFM Instance Example

We can later target a specific scan flip-flop instance in order to detect the same defect (referring to the same example) by writing an equivalent UDFM file as shown in Fig. 10. This example targets the scan flip-flop instance called "\stato_reg[1]". When the UDFM file is loaded, the defined delay faults are applied to the specified instance.

Table 6 reports the contribution to the defect coverage when adding extra patterns generated by the intra-cell aware ATPG tool. Column 2 reports the initial defect coverage by considering both logic tests & Scan chain Test. Column 3 reports the obtained final defect coverage. Again, we could actually increase the coverage for all the circuits. On average, the coverage improvement is about 13.93%. The fourth column reports the overhead in terms of the percentage of extra patterns that have to be applied to improve the coverage, which is about 42%. Therefore, a high quality test solution has to be composed of logic tests plus a fragment of extra test patterns generated to specifically target the escaped defects.

TABLE 6
INTRA-CELL AWARE ATPG CONTRIBUTION

Ckt	LT&ST DC%	DC%	Pattern Overhead (%)
b01	68.52	85.04	22.50
b02	51.85	72.36	29.50
b03	71.91	87.14	59.20
b04	76.52	88.79	67.70
b05	81.70	90.02	23.20
b06	54.48	84.51	50.60
b07	79.66	88.69	33.80
b08	73.66	86.93	35.10
b09	67.99	84.40	49.50
b10	76.72	86.17	23.70
b11	80.21	89.38	16.20
b12	81.02	88.92	36.70
b13	68.70	84.65	54.00
b14	81.90	91.88	41.50
b15	85.49	93.03	31.10
b17	84.06	92.36	59.20
b18	85.54	92.92	65.10

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TETC.2016.2624311, IEEE Transactions on Emerging Topics in Computing

b20	84.92	92.96	48.60
b21	85.64	92.93	40.60
b22	84.74	92.64	56.20
Average	76.26	88.29	42.20

We now compare the above results with those produced by a single test set targeting intra-cell defects. Basically, we exploit a commercial intra-cell aware ATPG tool to target the whole set of defects for each scan flip-flop. The comparison is done in terms of achieved defect coverage and test time (i.e., number of test patterns).

Table 7 reports the comparison results. Columns 6 and 7 summarize the efficiency of our proposed approach in terms of defect coverage increase and test length decrease as compared to the intra-cell aware ATPG. It can be noted that in most of the cases (17 out of 20 cases, i.e., column 6) the coverage achieved by the proposed solution is higher than the one achieved by the intra-cell aware ATPG tool. Even if for the smaller circuits the difference is not so high, for the larger circuits the difference is significantly high. For example, for b18 the proposed test solution achieves 11.3% higher defect coverage than the intra-cell ATPG. If we consider the test pattern length, we lead to similar conclusions. In most of the cases the number of patterns of the proposed solution is smaller than the one of the intra-cell aware ATPG tool.

TABLE 7
INTRA-CELL AWARE ATPG COMPARISON

Ckt	Proposed Test Solution		Intra-cell aware ATPG		Comparison	
	#DC (%)	*#p*	*#DC (%)*	*#p*	*% Incr in #DC*	*% Decr in #p*
b01	85.04	80	88.52	72	-4.10	-11.11
b02	72.36	61	73.77	45	-1.95	-35.56
b03	87.14	228	84.96	550	2.50	58.55
b04	88.79	390	84.89	1,199	4.40	67.47
b05	90.02	340	83.42	576	7.33	40.97
b06	84.51	81	84.03	127	0.57	36.22
b07	88.69	328	80.86	657	8.83	50.08
b08	86.93	211	80.25	343	7.69	38.48
b09	84.40	198	85.38	446	-1.16	55.61
b10	86.17	177	78.36	261	9.07	32.18
b11	89.38	476	80.41	484	10.03	1.65
b12	88.92	806	77.47	1,922	12.87	58.06
b13	84.65	313	75.51	639	10.80	51.02
b14	91.88	1,582	86.77	4,142	5.56	61.81
b15	93.03	3,322	82.24	7,471	11.60	55.53
b17	92.36	5,774	82.62	23,683	10.55	75.62
b18	92.92	10,170	82.45	49,369	11.27	79.40
b20	92.96	2,498	86.62	8,109	11.12	69.19
b21	92.93	2,267	86.83	8,192	6.56	72.33
b22	92.64	2,848	87.30	11,554	5.77	75.35
Average	88.29	1,607	82.63	5,992	6.46	46.64

Once again, for the largest circuits the difference is actually higher. If we consider b18, the proposed approach generates about 72% less patterns than the intra-cell aware ATPG (10,170 vs. 49,369 test patterns). It is important to mention that all the test sets have been generated by using the same commercial ATPG tool with the compaction option. The comparison results show that the proposed test solution definitely achieves better defect coverage with a lower test length.

7 CONCLUSION

Scan test is the most popular Design-for-Test technique. Assuring the correct behavior of the scan chain is therefore mandatory to ensure a high-test quality. Moreover, some of the defects affecting the scan cells may result in an uncorrect behavior of the circuit even during the normal mode. In this paper we presented an intra-cell aware testing approach for the scan chain. We first evaluated the effectiveness of different test sets w.r.t. the intra-cell defects affecting the scan flip-flops. This evaluation considered both scan chain test and functional test, and was based on test patterns generated by a commercial ATPG. Then, we analyzed the obtained results and we proposed to combine together the test sets to achieve higher intra-cell defect coverage. Finally, we compared the results produced by the combined test sets with the ones obtained by using a commercial ATPG tool directly targeting intra-cell defects. The comparison shows that considering different test sets is very effective in terms of test quality and of test length. Future works will mainly focus on the diagnosis of the addressed type of defects.

REFERENCES

[1] B. Kruseman, A. Majhi, C. Hora, S. Eichenberger and J. Meirlevede, “Systematic defects in deep sub-micron technologies,” *IEEE International Test Conference*, 2005, pp. 290-299.

[2] S. Eichenberger, J. Geuzebroek, C. Hora, B. Kruseman, and A. Majhi, “Towards a World without Test Escapes,” *IEEE International Test Conference*, 2008, paper 20.1.

[3] F. Hapke, W. Redemund, A. Glowatz, J. Rajski, M. Reese, M. Hustava, M. Keim, J. Schloeffel and A. Fast, “Cell-Aware Test,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* , vol.33, no.9, 2014, pp.1396-1409.

[4] F. Hapke, M. Reese, J. Rivers, A. Over, V. Ravikumar, W. Redemund, A. Glowatz, J. Schloeffel and J. Rajski, “Cell-aware Production test results from a 32-nm notebook processor,” *IEEE International Test Conference*, 2012, pp. 1-9.

[5] A. Bosio, L. Dilillo, P. Girard, A. Todri-Sanial, A. Virazel, S. Bernabovi and P. Bernardi, “An Intra-Cell Defect Grading Tool,” *IEEE International Symposium on Design and Diagnostics of Electronic Circuits & Systems*, 2014, pp. 298-301.

[6] M. Bushnell and V. Agrawal, "Essentials of Electronic Testing for Digital, Memory and Mixed-Signal VLSI Circuits," *Springer*, ISBN 978-0-306-47040-0, 2002.

[7] S.R. Makar and E.J. McCluskey, "Functional tests for scan chain latches," *IEEE International Test Conference*, 1995, pp. 606-615.

[8] Tang Huaxing, B. Benware, M. Reese, J. Caroselli, T. Herrmann, F. Hapke, R. Tao, Cheng Wu-Tung and M. Sharma, "Diagnosing Cell Internal Defects Using Analog Simulation-Based Fault Models," *IEEE Asian Test Symposium*, 2014, pp. 318-323.

[9] F. Hapke, R. Arnold, M. Beck, M. Baby, S. Straehle, J. F. Goncalves, A. Panait, R. Behr, G. Maugard, A. Prashanthi, J. Schloeffel, W. Redemund, A. Glowatz, A. Fast, J. Rajski, "Cell-aware Experiences in a High-Quality Automotive Test Suite," *IEEE European Test Symposium*, 2014, pp. 1-6.

[10] A. Touati, A. Bosio, L. Dilillo, P. Girard, A. Virazel, P. Bernardi and M. Sonza Reorda, "Scan-Chain Intra-Cell Defects Grading," *IEEE Design & Technology of Integrated Systems*, 2015, pp. 1-6.

[11] J. C.-M. Li, "Diagnosis of Resistive-Open and Stuck-Open Defects in Digital CMOS ICs," *IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 24, Issue. 11, 2005, pp. 1748-1759.

[12] A. Bosio, P. Girard, S. Pravossoudovich, P. Bernardi and M. Sonza Reorda, "An efficient fault simulation technique for transition faults in non-scan sequential circuits," *IEEE International Symposium on Design and Diagnostics of Electronic Circuits & Systems*, 2009, pp. 50-55.

[13] Mentor Graphics. Tessent Scan and ATPG User's Manual.

Patrick Girard received a PhD. degree in Microelectronics from the University of Montpellier, France, in 1992. He is currently Research Director at CNRS (French National Center for Scientific Research) and works in the Microelectronics Department of the Laboratory of Informatics, Robotics and Microelectronics of Montpellier (LIRMM) - France. He is co-Director of the International Associated Laboratory « LAFISI » (French-Italian Research Laboratory on Hardware-Software Integrated Systems) created in 2013 by the CNRS and the university of Montpellier with the Politecnico di Torino, Italy. His research interests include all aspects of digital testing and memory testing, with emphasis on critical constraints such as timing and power. Reliability and fault tolerance are also part of his research activities. He has served on numerous conference committees and is the founder and Editor-in-Chief of the ASP Journal of Low Power Electronics (JOLPE). He is also an Associate Editor of the IEEE Transactions on Computers, IEEE Transactions on CAD and the Journal of Electronic Testing – Theory and Applications (JETTA - Springer). He has supervised 33 PhD dissertations and has published 6 books or book chapters, 60 journal papers, and more than 200 conference and symposium papers on these fields. Patrick Girard is a Fellow of IEEE.

Arnaud VIRAZEL received the PhD. degree in Microelectronics from the University of Montpellier, France, in 2001. He is currently Assistant Professor at the University of Montpellier, and works in the Microelectronics Department of the LIRMM (Laboratory of Informatics, Robotics and Microelectronics of Montpellier France).
His has published 3 books or book chapters, 35 journal papers and more than 140 conference and symposium papers spanning diverse disciplines, including DfT, BIST, diagnosis, delay testing, power-aware testing and memory testing. He is an IEEE member.

Aymen Touati is a PhD student at Laboratory of Informatics, Robotics and Microelectronics of Montpellier (LIRMM)-University of Montpellier, France. His current research interests include Power-aware testing, At-speed testing, diagnosis and Computer-aided design techniques. Aymen was awarded MS degree in Telecommunications and Microelectronic Systems from the University of Nice Sophia Antipolis, France in 2012.

Matteo Sonza Reorda received his MS degree in Electronics (1986) and PhD degree in Computer Engineering (1990), both from Politecnico di Torino. He currently is a Full Professor at the Dept. of Control and Computer Engineering of the same University. His research interests include test of SoCs and fault tolerant electronic system design. He is an IEEE Fellow.

Alberto Bosio received the PhD. in Computer Engineering from Politecnico di Torino in Italy in 2006. From 2007 is an Associate Professor at the LIRMM-University of Montpellier in France. He has published articles spanning diverse disciplines including testing and reliability of digital circuits and systemes, approximate computing and emergent technologies. He is an IEEE member.

Paolo Bernardi obtained his MS degree in Computer Science Engeneering from the Politecnico di Torino in 2002. From May to December 2002, he worked in the Dipartimento di Automatica e Informatica (Prof. M. Sonza Reorda) of the Politecnico di Torino within the research project "Techniche di collaudo per Embedded Cores" oriented to the realization of a programmable architecture for the test and diagnosis of Flash memories.
In the period between 2003 and 2005, he has been in the Dipartimento di Automatica e Informatica as a PhD student (advisor Prof. M. Sonza Reorda). In March 2006, he obtained his PhD in Computer Science proposing a Doctoral thesis titled "Test Techniques for

Systems-on-Chip". From January 2006 to November 2007, he has been a post-doc fellow in the Dipartimento di Automatica e Informatica. He is an associate professor of the III faculty of the Politecnico di Torino.

Etienne Auvray obtained the engineering degree from "Ecole Supérieure d'Electricité" in 1979; he focused his activity in the military space division of Thomson and extended failure analysis techniques for CCD and IIIV devices. In 1995, he moved the Serma technology group and initiated FIB services in various domains, including device modification and TEM sample preparation on various products (i.e. IC, MEMS, magnetic heads, etc.). In 2001, he joined STMicroelectronics as lab manager where is currently in charge of failure analysis methodologies in the automotive group.