BIST Architectures for Dynamic Test

Silvia Andrea Teixeira Gomes

1IST/INESC-ID Lisboa, R. Alves Redol, 9, 3º, 1000-029 Lisboa, Portugal

Abstract

The increasing complexity and performance of integrated Systems on a Chip (SoC) implemented in Deep Sub-Micron (DSM) technologies do impose severe requirements in test setups. A significant number of physical defects in new node technologies require dynamic fault testing, eventually at-speed-testing. Delay Fault Testing, using transition and path delay fault models, can provide a good coverage for delay defects. The transition fault model is widely used in industry, since it leads to manageable test time (proportional to the number of test vectors composing the test pattern). For sequential digital modules, basically two transition fault test pattern generation methods can be applied with scan-based test: Launch-on-Shift (LOS) and Launch-on-Capture (LOC). In LOS, an at-speed transition of the Scan Enable signal is required; however, this feature is not supported in many designs. Most of them only implement a low speed solution and, thus, can only implement the LOC method. Even so, LOS achieves a higher fault coverage with fewer test vectors than the one obtained with LOC.

Digital SoC testing has become a significant problem. External test solutions can become prohibitive, and can only be applied in production test, not during lifetime testing, which may be required for many new products. Built-In Self-Test (BIST) is considered an attractive solution, as it can verify a failure-free status autonomously, without any external stimuli. The Scan BIST approach merges the benefits of conventional BIST and Scan-based design. Although leading to higher area overhead than scan design, scan BIST can lead to good fault coverage, with low cost test generation, and it can reduce maintenance costs. Moreover, it can provide a low-cost solution to implement at-speed test.

In this work, a new methodology for SoC dynamic test is proposed. The underlying principle is to combine conventional BIST with dynamic scan design (performing LOS or LOC). The proposed methodology introduces, using a single architecture, three solutions to implement LOC, LOS or both (LOS or LOC can be activated using only one control signal). The proposed methodology and the corresponding design flow are demonstrated using ITC’99 circuit benchmarks. Simulation results show that LOS-based scan BIST leads to higher fault coverage than the LOC-based scan BIST.

Keywords - Built-In Self-Test, Delay Test, Fault Coverage, Launch on Shift, Launch on Capture, Transition Faults

1. Introduction

Microelectronics industry profitability is based on Moore’s Law, which predicts an exponential decrease in feature size [1]. Hence, after four decades, integrated circuits became SoC (Systems on a Chip) with ever increasing complexity, density and performance. Such trend also enhanced power consumption, leading to the need of low-power design, and sophisticated power and thermal management solutions. In order to constraint internal electric fields (and constraint power consumption), for each node technology a reduction of the power supply voltage level, $V_{DD}$, is required. Today’s technologies (down to 65 nm, $V_{DD}$=1 V, and hundred million gates operating in the GHz range) use new processing materials and manufacturing processes, like low-K and high-K dielectric materials and OPC (Optical Proximity Correction) [2]. New materials and processing technology lead to new, emerging physical defects. Complexity, performance, power consumption and low pin count bring a difficult challenge: how to specify and run a cost-effective test process [1]. In fact, for digital systems, not only it is difficult to derive a cost-effective test to cover static faults (like those described e.g. by the classic single Line-Stuck-At (LSA) fault model), but also dynamic faults, as a significant set of emerging defects manifest themselves only as time-related defects. Hence, cost-effective solutions to uncover dynamic faults (namely, delay faults) became mandatory [3]. Delay test quests for two-vector sequences, the first vector to initialize the circuit, and the second vector to trigger a transition
and/or the activation of a Boolean difference through a
propagation path to an observable output.

The goal during new product development is, thus,
to provide high-quality test in a cost effective way. As
a matter of fact, testing is crucial in product life-cycle.
First, in the design environment, test is used in design
validation; second, in the manufacturing environment,
test must uncover manufacturing defects, discriminating
good from defective parts; third, during product lifetime,
for many products it is necessary to detect any defects
that can produce a failure operation of the system.

Test quality can be measured using various metrics,
describing different attributes. Usually, the Test
Effectiveness (TE), the Test Length (TL), the Test
Overhead (TO) and the Test Power (TP) are relevant.
The first one, TE, is the crucial one. Its measure is a
necessary condition to proceed with the test process. If
TE leads to unacceptably high values of the Defect
Level, the test pattern must be improved. Therefore,
in our work, the attention is focused on TE and on fault
coverage evaluation through fault simulation.

Device scaling down is not followed by a reduction
of Automatic Test Equipment (ATE) specifications and
cost. Consequently, some test cannot be externally
performed and it has to be made on-chip. Therefore,
Built-In Self-Test (BIST) [4] is an attractive alternative
and is becoming more popular as a complementary
technique of performing a device or system test. The
BIST methodology proposed in this work is intended as
a valuable contribution to self-test delay faults in a cost
effective way.

Delay Fault Test is mandatory due to the increasing
performance requirements and to the likely physical
defects occurring in DSM technologies. High
performance also makes the products more sensitive to
signal integrity loss, which may affect also the quality
of the test process [20]. Several delay fault models
have been used to describe the impact of defects on
circuit performance [1] [3] [6]. The most common are
the Transition Fault (TF) and Path Delay Fault (PDF)
Model. In both models, sequences of test patterns are
required: a vector pair (T1, T2), where T1 is the
initialization vector and T2 is the vector that launches
the appropriated transition to a primary output or to a
flip-flop.

For sequential circuits, typically with low
controllability and observability of the feedback
registers, a test mode is included, with scan design. [1]
[4]. Using this technique, the three most common TF
pattern generation methods are Enhanced Scan,
Launch-on-Capture (LOC) and Launch-on-Shift (LOS)
[3] [6]. Basically, they differ on the way of applying
the second vector in each sequence. In LOC, the second
vector is the Circuit under Test (CUT) response to the
first vector applied. In LOS, the second vector is a
shifted version of the first one. Yet, the most important
difference between to methods is associated with the
Scan Enable signal: in LOS it must switch at-speed.
Consequently, this method is difficult to implement.
However, it leads to higher fault coverage than LOC
with much less vectors applied and a reduced test
application cost. As a consequence, recently new
architectures have been proposed [10] [11] [12] [13].

In the literature, many solutions have been
presented to enhance of BIST solutions for static faults:
weighted pseudo-random TPG [21], LFSR reseeding
[16], and deterministic BIST [19]. The injection of
deterministic test vectors may be performed by bit-

Nevertheless, scan BIST solutions need to be
devised for covering dynamic faults, namely delay
defaults. As scan-based test using enhanced scan, LOS or
LOC have been applied widely in industry, it may be
rewarding to derive novel scan BIST solutions adapting
dynamic scan to BIST. In this work, we try to develop
a new methodology and several architectural solutions
for this problem, merging dynamic scan with scan
BIST and the results, as we will show, are promising.
TE of pseudo-random TPG to uncover transition faults,
and path delay faults, is also investigated.

The aim of this work is to propose a novel at-speed
Scan BIST methodology targeting the coverage of
delay faults. In the proposed methodology, two scan
design techniques for delay fault coverage (LOC and
LOS) are used in a self-test environment. During this
work three architectures have been derived: a Scan
BIST based on LOC, a scan BIST based on LOS and a
hybrid solution that merges the two techniques and
allow the implementation of LOC or LOS, by changing
only the value of one input variable. The Scan BIST
structure is the same for the three developed
approaches. The CUT, with scan insertion is
surrounded by an LFSR, MISR, multiplexers and a
BIST Controller. In this work, the CUT used to
demonstrate the usefulness of the methodology are
ITC’99 circuit benchmarks [14], with an inserted single
scan chain.

The fault coverage, based on transition fault
coverage, obtained with the dynamic scan BIST
architecture based on LOS is higher than the one that
the LOC approach can achieve.

The paper is organized as follows. The basic
definitions and terminology used are described in
Section 2. In Section 3, Dynamic BIST architecture is
proposed, as well as the details about the new
methodology. The benchmark circuits used and the
most relevant simulation results are described in
Section 4. Finally, Section 5 summarizes conclusions
and further research.
2. Design for Testability

Physical failures and fabrication defects are difficult to be modelled mathematically. What we really want is to model their impact on circuit behaviour. The presence of a Delay Fault results in a circuit whose operation is logically correct but does not perform at the required operating frequency [4]. The most common Delay Faults models are the Transition Fault (TF) and Path Delay Fault (PDF).

The Transition Fault Model is similar to the Stuck-at Fault Model in some aspects. The consequence of a transition fault at a given node in a circuit is that a rising or a falling transition at this node will not reach a scan flip-flop or a PO within the desired time (a clock period). As a result, the possible transition faults of a gate are slow-to-rise and slow-to-fall types [1]. In a circuit, the time slack is defined as the difference between the time allowed for signal propagation and the time required to propagate a transition through the critical path of the CUT [5]. For a transition fault to cause an incorrect value at a circuit output it is necessary that the size of the delay fault be such that it exceeds the slack of at least one path from the fault location to a PO or scan flip-flop.

Path Delay Faults causes the cumulative propagation delay of a combinational path to increase beyond some specified time duration [1]. A physical path is an interconnection of gates from an input to an observation point. This fault model is used to detect small delays along a path. For each path there are two path delay faults, which correspond to the rising and falling transitions, respectively [1].

Design for Testability (DFT) refers to the design techniques that make test generation and application cost-effective [8]. Scan Design is a structured method of DFT approach for digital circuits. The main idea in this technique is to obtain control and observability for flip-flops [1]. This is obtained by adding a test mode (also known as scan mode) such that when the circuit is in this mode the flip-flops are reconfigured into one or more shift registers. The inputs and outputs of these shift registers (known as scan registers) are made into primary inputs and primary outputs [1]. The mode of operation is controlled by the select input to the multiplexer: either system mode or scan mode.

Nowadays, with all the available Computer-aided Design (CAD) tools, scan design can be entirely automated, including scan flip-flops insertion, interconnection of the scan chain, generation of the test vectors and the analysis of the fault coverage. At the same time, with this test method, high fault coverage can be achieved, using a moderate increase in area and decrease in speed.

During its lifetime, a digital system is tested on numerous occasions. Test and diagnosis must be fast (low TL) and lead to high fault coverage. With self-test, test is specified as part of the system functionality. BIST methodology incorporates test pattern generation and output response analysis capabilities on-chip [4]. The goal of creating a BIST architecture based on the scan design is to incorporate a TPG in the form of an LFSR in the Scan In input of the scan chain and an ORA in the form of a MISR in the Scan Out output of the scan chain.

The architecture implemented in the present work is a test-per-scan, embedded BIST architecture where the scan chain is created from existing flip-flops.

The three most common approaches to apply deterministic at-speed test are the Enhanced Scan, Launch on Capture (LOC) and Launch on Shift (LOS) (Figure 1). Basically, the three approaches differ in the way of storing and applying the second vector of each vector pair.

Figure 1 – Classification of Delay Test.

In Enhanced Scan method both vectors (T1 and T2) are shifted in during the shift process to the scan flip-flops. During the shift of T2 it is assumed that the initialization of T1 is not destroyed. Therefore, special scan flip-flops are needed; these are referred as hold-scan flip-flops [9]. That has the benefits of achieves higher fault coverage and the test data volume is limited. However, it has the disadvantage of higher area overhead related to the need to implement the hold-scan flip-flops.

Launch on Capture is the most common form of delay fault application method [7] [12] [13]. That requires only one vector to be shifted in during the shift cycle. Vector T2 is the circuit response obtained by the application of the first vector. This approach does not require at-speed transition of the Scan Enable signal, because the scan flip-flops are changed into the normal mode of operation by lowering the Scan Enable signal before the Launch Cycle (LC). It is shown in Figure 2.

Figure 2 – Waveform for Clock and Scan Enable for LOC.
Hence, it can be implemented with low hardware overhead. This method is a cost-effective solution. However, it has the inconvenient of achieve lower fault coverage.

Launch on Shift method uses for each vector pair a shifted version of the first vector as its second vector. In that approach the Scan Enable signal is high during the last shift (or Launch Cycle) and must go low to enable response capture at the Capture Cycle [10]. That is presented in the diagram of Figure 3. The Scan Enable signal is high during LC and must go low to enable capture response during CC. This is the main disadvantage of this method. As a consequence, the design and implementation of that approach is too expensive and result in a longer design time.

The tool can achieve higher fault coverage within significantly fewer test vectors [12].

![Figure 3 – Waveform for Clock and Scan Enable for LOS.](image)

### 3. Dynamic BIST Methodology

In this section a new dynamic scan-BIST methodology is proposed and three architectures implementing it are introduced. The proposed methodology for scan-based dynamic BIST merge the LOS and/or LOC techniques for delay testing, used in scan design with external test, with BIST. The methodology is implemented with the architecture shown in Figure 4.

![Figure 4 – Dynamic BIST Architecture proposed in this work.](image)

The proposed architecture is composed of the CUT (reconfigured with scan flip-flops), a BIST Controller, a MISR, the input MUX and two LFSR. Due to fact that this new architecture will be used for at-speed test, some changes have to be made, especially in the BIST Controller (with different functionality) and in the TPG (using two LFSR). We use two linear LFSR, namely LFSR.PI and LFSR.Scan. The first one is used to generate pseudo-random (PR) test patterns to be applied to the CUT’s primary inputs and the second LFSR generates PR test patterns to be applied at the first flip-flop of the scan register. Although that solution introduces an area penalty, we decided to implement this configuration because it can reduce the polynomial correlation and, as a result, it can increase delay fault coverage.

The proposed methodology and its architecture can accommodate LOS, LOC or both. The difference among the three presented approaches is clustered in the BIST Controller functionality. Due to the fact that we have to implement different techniques to test delay faults and that the state of Scan Enable signal (defined as test_se signal in this work) is different in the Launch Cycle of LOS and LOC, we have to add a new state at the module definition (referred as LAUNCH). That decision implies an increase area in the BIST Controller, as compared to the traditional module. This area overhead may significant (of about 47%, in the examples presented in section 4). Consequently, the Dynamic BIST architecture introduces an additional overhead (about 17.5% area overhead, in the mentioned examples, for the overall structure). This cost occurs in the three approaches, and is similar in all, as they all exhibit similar total area.

Power consumption is another critical issue. Test power can be limited by reducing the test units within test sessions or reducing the clock frequency [15]. However, in this case that is not an option, as we are performing at-speed testing.

The proposed architecture has five modules: CUT, BIST Controller, LFSR, MISR, and MUX. The BIST Controller is the only that changes between approaches.

The BIST Controller controls all the other modules and operations. In the proposed architecture, two counters are used for the BIST Controller: one counter (referred as the scan counter, C_SCAN) to count the number of flip-flops in the scan chain for shifting in the scan vectors and another counter (referred as the vector counter, C_VEC) to count the number of scan vectors applied to the CUT.

The BIST Controller for this dynamic scan BIST with LOS is implemented based on the finite state machine whose state diagram is presented in Figure 5. This was designed to allow a credible implementation with an area optimization.

The state machine only leaves IDLE state if the BISTstart signal goes high. While this condition is false, the scan counter and the vector counter are disabled and the test in not performed. When BISTstart goes high and the state machine moves to the RES state, counters are immediately activated and the reset
to LFSR, MISR and CUT are done, to guarantee that they start their operation always at the same time and at the same way. Actually, as it can be seen by the name, in the RES state the modules are activated, preparing for the self-test. After all, the changing of test_se from 0 to 1 is the condition for the state machine to move to next state: the SCAN state.

Figure 5 – LOS testing state machine.

When test_se signal goes high, this means that the shifting in of the scan chain of the CUT is initialized. The input MUX is also enabling the local TPG to drive the CUT, and thus the LFSR_SCAN starts to generate the pseudo-random test patterns to be applied to the CUT. In this mode of operation, the SCAN mode, the test is performed and the state machine stays in this mode until the scan and vector counting are not completed. Therefore, the FSM only moves to the next mode (LAUNCH) when the scan chain is totally filled (scan_finished = 1) and the test vectors are totally applied to the CUT (vect_finished = 0). If the counting is completed, BISTdone goes high, which means that the self-test session is concluded. As a result, the state machine returns to the IDLE state.

When the FSM is in LAUNCH state, and because this is the LOS approach, test_se does not change and stays at the high level. The LFSR_PI generates the test vectors to be applied as primary inputs to the CUT (MUX is enable) and LFSR_SCAN stops generating test patterns to be applied to the scan chain. As a result the second vector is a shifted version of the first one applied.

Test_se must go low to allow the state machine to change state. This is the condition to move to the CAPTURE state. In this state, the self-test is stopped and the circuit’s response is captured, which is performed by enabling the MISR. Before continuing the test operation and returning to the SCAN state, the scan counter and the CUT are reset. The Test_se signal must go high again, which is the condition to return to the SCAN mode of operation.

Comparing the Scan based on LOS and the Scan based on LOC, the only difference between the state diagrams of the two FSM is in the test_se signal logic value between the SCAN and LAUNCH states.

In the Scan based on LOC architecture, to move from SCAN to the LAUNCH state, test_se signal must go low (that is the main difference between the approaches). LFSR_SCAN stops generating test patterns to be applied to the scan chain and the MUX is enabled, so the LFSR_PI generated test vectors can be applied as PI at the CUT.

In this third alternative (with the same architecture), LOS and LOC-based delay self-testing may be activated. This is a very interesting implementation because in the same structure we can perform two different delay tests, by changing the logical state of only one input signal, namely the BIST start signal. Usually, and as it will be confirmed in the next section, this new solution improves transition fault coverage with a similar area overhead of the other two presented solutions.

In Figure 6 the signal waveforms of LOS and LOC behaviour are presented. As can be observed, when the BIST_start goes low, the type of test performed changes. Thus, before the signal changes logical state, test_se is at low level only during CAPTURE mode and the LOS-based delay test is performed. After the changing, test_se is at low level at LAUNCH and CAPTURE states and, as a result, the LOC-based test is performed.

Figure 6 – Functional explanation of LOS and LOC implementation behaviour (ModelSim®).

4. Simulation Results

In order to perform the proposed dynamic scan BIST methodology for Transition Delay Fault testing, several software tools have been used. The corresponding design flow is presented in Figure 7. This design flow summarises all the performed steps starting with insertion, passing through the development of the Scan BIST architectures and logic synthesis, down to the transition fault simulation using pseudo-random test patterns.
The circuits that have been tested in this work (b06, b10 and b13) are ITC’99 Benchmarks [14], developed in the CAD Group at Polytechnic of Torino. The target IC technology is the CMOS 0.35 µm technology from Austria Micro Systems and the c35_CORELIB, c35_IOLIB_4M and c35_IOLIBV5_4M cell libraries.

The b06 benchmark circuit is an interrupt handler [14] with 56 gates, 2 PI, 6 PO and 9 feedback flip-flops. During the transition fault simulation process, 308 single transition faults were automatically injected in the circuit by the Synopsys TetraMax™ tool. To perform the test, TL06 = 130000 pseudo-random test vectors were generated and applied by the on-chip TPGs. The CUT estimated area is about 6912 µm². The implemented BIST Controller has an area overhead of 47% and the Scan BIST architecture has an area overhead of 17.5% as compared to the traditional BIST approach.

The transition fault coverage achieved by pseudo-random patterns, generated by the two used LFSR, is shown in Figure 8. As expected, the LOS and LOC approach is the one that leads to the best TF coverage results. Even with low fault coverage, the other two BIST solutions lead also to acceptable results. These need to be compared to those achieved by deterministic test patterns (see Table 1). According to the expectations, with deterministic test patterns, we can achieve higher transition fault coverage than with PR test patterns, with fewer test patterns applied. The same trend as in the pseudo-random test can be observed for deterministic test: the transition fault coverage is higher in the LOS and LOC architecture and has the lowest value in the Scan BIST based on LOC approach.

Comparing the results achieved by pseudo-random and deterministic patterns, the better results are reached by the latter ones (deterministic), as expected. Nevertheless, the TF coverage of the PR test pattern is also acceptable and shows that it is possible to guarantee good fault coverage with (at least some of) the developed solutions.

The second benchmark circuit, b10, is a voting system [14] with 206 gates, 11 PI, 6 PO and 17 scan flip-flops. In order to perform the transition fault simulation, 696 transition faults were automatically injected in the circuit by the TetraMax™ tool. In this case, TL10 = 300000 pseudo-random test patterns were applied. The CUT estimated area is about 14484.6 µm². The implemented BIST Controller has an area overhead of 48% and, as a consequence, the Scan BIST approach has an implementation overhead of 19% as compared to the traditional BIST approach.

The transition fault coverage values achieved by pseudo-random test patterns, generated by the two used LFSR, are shown in Figure 9. For performing those simulations 300000 pseudo-random test patterns were used. As it can be observed, the third BIST implementation (LOS and LOC) is the one that reaches the best results, although in this case the three of them exhibit low transition fault coverage. This can be justified by the circuit itself, being more complex and having a higher number of nodes that are not observable.

In Table 2 the TF coverage results for the deterministic test patterns are presented. As expected, the fault coverage achieved is much higher than with the pseudo-random test patterns and with fewer test patterns applied.
The b13 benchmark circuit is an interface to meteo sensors [14] with 362 gates, 10 PI, 10 PO and 53 flip-flops. During transition fault simulation, TetraMax™ automatically injects 1392 single transition faults in the CUT. For this more complex benchmark, $T_{13} = 650000$ pseudo-random test patterns generated by two LFSR were used. The CUT estimated area is about $36461.4 \mu\text{m}^2$. The implemented BIST Controller has an area overhead of about 48% and the Scan BIST approach has, in this case, an implementation overhead of 18.5%.

The transition fault coverage achieved by applying pseudo-random test patterns in this last case is presented in Figure 10. A test length of 650000 patterns was used. In the three cases the reached TF coverage value is interesting and can be considered as good fault coverage for using only PR stimuli. One more time, the LOS and LOC architecture allows the best results. In the opposite site is the Scan BIST based on LOC approach.

In Table 3 the transition fault coverage achieved for the three different proposed methods are shown, together with the number of deterministic test patterns applied. According to the expectations, the results achieved by applying deterministic patterns are better than those obtained with pseudo-random and with fewer applied test patterns.

For each circuit analysed, the transition fault coverage results achieved by applying to the CUT deterministic test patterns are presented in Figure 11. As it can be seen, LOC approach leads to the worst fault coverage, followed by LOS. The highest transition fault coverage is obtained by merging LOS and LOC solutions. These are the better results that we can achieve with deterministic test sets. The aim is, using LFSR test generators, to reach values as close as possible to those presented.

In Figure 12 the transition fault coverage results for the three proposed architectures and for each circuit are shown, applying the pseudo-random test patterns to the CUT. As in the last case, the LOC approach shows the worst results, followed by LOS. The LOS and LOC proposed architecture achieves the best transition fault coverage. The three circuits have worst results in comparison with those obtained by deterministic patterns. This is an expected situation since we used pseudo-random patterns. However, we achieved reasonably good transition fault coverage values and it is possible to say that those results may be acceptable for the presented proposed solutions. Anyway, it is possible to improve the TF coverage, using another LFSR test generator with low polynomial temporal correlation, or adding some deterministic test vectors. One way to solve this problem is to increase the degree of polynomial, despite the area increase caused by that.
As a summary, the transition fault coverage depends on the CUT complexity and sequential depth, of the type of delay test that is carried out (LOC, LOS or LOS and LOC) and on the used pseudo-random test generator, due to the polynomial correlation. In this manner, it is mandatory to have a good pseudo-random sequence generator.

In terms of CPU time and memory usage, the observable values are as expected, although the higher verified improvements. This has to do to the higher number of applied pseudo-random patterns.

5. Conclusions

In this work we addressed the problem of adapting scan BIST to uncover delay faults, which is mandatory for digital SoC implemented in nano-CMOS. The work presented is a preliminary step, and still requires further research. Anyway, a new methodology for dynamic scan BIST has been proposed. The underlying principle is to apply LOS and LOC techniques to scan BIST.

The proposed methodology uses a single architecture and if we modify the functionality of a single module – the BIST Controller - we are able to implement three new Scan BIST solutions that implement the transition fault pattern generation methods. They are referred as Scan BIST based on LOC, Scan BIST based on LOS and Scan BIST based on LOS and LOC (which merges the other two techniques).

The third proposed solution allows the two TF detection techniques, by switching only one input signal: if the BISTstart control signal is at high level, LOS is performed; if it is at low level, the LOC approach is performed. All three solutions have approximately the same total silicon area, performance degradation and pin count although the hardware changes. Performance degradation is similar to the one of classic scan BIST – basically, the additional propagation delays associated with the input MUX, and with the replacement of the CUT’s flip-flops by scan flip-flops.

The associated costs and benefits of the new approaches are similar to the traditional Scan BIST. The proposed solutions introduce pin overhead, because three pins must be added: (1) the BISTstart pin, to control whether BIST operates or not, (2) the BISTdone terminal and (3) the MISR_out pin. This is the same pin overhead that a traditional Scan BIST approach introduces. At the same time, the MUX (and scan insertion in the CUT) introduce performance penalty, similar to the one of classic scan BIST. The area overhead is other critical problem: due to the fact that we implemented a structure that requires one more state in the BIST Controller (LAUNCH), the part of the module that requires now 3 flip-flops (up to 8 states), instead of 2 flip-flops (the 4 states of the classic scan BIST) suffers an area enhancement of about 47% which provokes an increase of about 18% of area in the Dynamic Scan BIST architectures. Other costs are related to the increase in power consumption during test and the increased design effort and time overhead to perform self-test.

Associated benefits are the reduction of maintenance and test costs, the lower test pattern generation cost and the reduced need of external test equipment. At the same time, it has the advantage of perform at-speed test and lead to higher fault coverage.

In terms of transition fault coverage, the Scan BIST based on LOS and LOC architecture leads to the best coverage results, while the lower TF coverage is obtained by Scan BIST based on LOC. This confirms that similar results, obtained with scan design and external test, also apply to built-in self test. Despite of the area overhead related to the implemented BIST Controller and the two LFSR used, the proposed solutions, especially the Scan BIST based on LOS and LOC, are promising solutions, leading to good transition fault coverage results.

One relevant conclusion of the simulation results is that pseudo-random test patterns can do nicely to uncover a significant subset of the transition faults. In fact, with an area overhead of 18% we obtain, by applying pseudo-random test patterns, a transition fault coverage around 70% (depending on the used CUT and on the degree of polynomial of the used LFSR test generator). At the same time, with a small area increase of 1.2% we can obtain a better transition fault coverage with Scan BIST based on LOS and LOC implementation.

In the future work must be carried out to evaluate power consumption during BIST sessions, to improve test effectiveness and guarantee that the proposed methodology is fully automated, if it is to be used with complex designs.
References


