Reducing Launch and Capture Power Using Saboteur and Mutants Method

Jonisha Stanis.S[1] Maria Antony.S[2]
[1]M.E Student (VLSI Design) [2]Assistant Professor
[1,2]Department of Electronics & Communication Engineering
[1,2]Kalaignar Karunanidhi Institute of Technology

Abstract— Testing of VLSI circuit aims for high quality screening of the circuits by targeting on performance related faults. Excessive switching in launch and capture operation results in high yield loss. Power management is also a major issue in VLSI design technique. Scan chain reordering method is used to reduce the number of transitions and thus the power consumption gets reduced. Fault injection technique based on the use of hardware description language offer important advantages than other techniques. Saboteurs and mutants methods were important fault injection techniques. By which the number of test pattern generation can be reduced and thus the time for processing was also gets reduced. As this type of technique can be applied during the design phase of the system, which reduces the time-to-market. They provide high controllability and reachability. This technique improves the fault coverage and reduces pattern count. The proposed algorithm is coded in VHDL and simulated using ModelSim and Xilinx ISE 8.1isimulator. The results obtained are compared with the existing version of the technique.

Key words: Launch Off Capture (LOC), Launch off Shift (LOS), Automatic Test Pattern Generation (ATPG), Linear Feedback Shift Register (LFSR)

I. INTRODUCTION

Testing on VLSI circuits aims for high quality screening of VLSI circuits by targeting performance related faults. Scan based at speed testing necessitates load, launch and capture operation for all test pattern. Loading operation is done by using scan/shift operation, filling up all the scan chains with the pattern. There are mainly two different schemes for launching transitions off the serial loaded pattern. In the Launch Off Capture/broadside test, a functional capture operation launches transitions from the locations where the serially loaded pattern (v1) differs from the response of the combinational logic to v1, where the launch pattern (v2). In launch off shift (LOS)/skewed load test, a single cycle shift operation launches shift operation launches transition from the locations where serially loaded patterns(v1) differs from its one-bit shifted version, where the launch pattern is (v2). In both schemes, a subsequent fast functional capture operation, which is of a functional clock period apart from the launch event, sets a deadline for the transitions to arrive at their destinations, a timing-related defect that slows down the chip below its rated clock speed is thus exposed. Yield loss problems are experienced in at-speed testing schemes. Excessive switching activity during the launch cycle may result in elevated peak supply currents, leading to IR drop that increases the signal propagation delays in the combinational logic. The end effect cannot be differentiated from that of a timing-related defect, causing a functional chip to fail the at-speed test.

Peak power during the launch cycle of at-speed testing should therefore be reduced in order to avoid the yield loss induced by IR drop. Significant research efforts have been expended in reducing power dissipation during the launch and capture of at speed testing. Test pattern generation while accounting for the functional clock gating logic in order to produce patterns that disable parts of the design during launch and capture has been proposed to reduce peak power at the expense of pattern count inflation. Another approach that elevates pattern count while reducing peak launch power has been in the form of generating patterns under the constraint that only one chain launches transitions while all chains capture them. Another similar scan-segmented solution partitions the scan cells into three regions where only two out of three regions launch and capture any test pattern. A partitioning approach has been proposed where power wise costly patterns are further analyzed via fault simulation to identify the location of the care bits, which dictate the partitioning of the design during capture; with few problematic patterns, such an approach can deliver power savings. pattern count increase has been experienced even when the design is partitioned via ILP that minimizes capture violations, additional test patterns, possibly of a high sequential depth, need to be generated for the faults missed due to capture violations. Finally, low-power automatic test pattern generation (ATPG) solutions have also been proposed.

A. DFT Support and Implementation for LOC Testing:

To restore the load state of the interface register upon the launch and capture operations, one shadow register is inserted for each interface register. Throughout the shift operations, the shadow register copies the content of the interface register, and while processing the capture window, the shadow register is not clocked, ensuring that the copied value in the last shift cycle is retained.

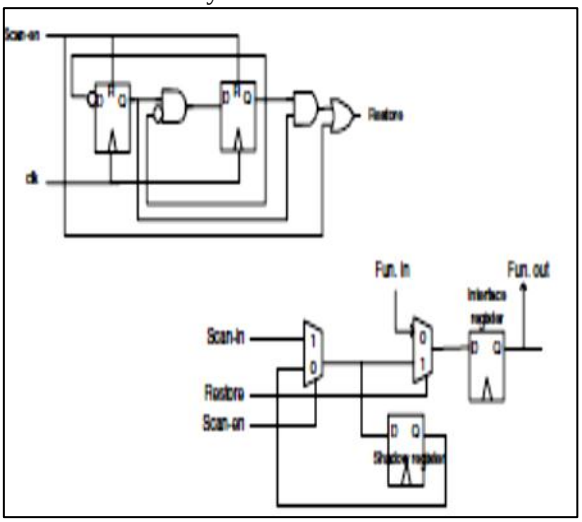

Fig. 1: DFT support for restore operation in LOC testing

Upon every launch and capture operation that the interface register gets involved in, its load state is restored by copying the content of the shadow register back into the interface register. Effectively, a multiplexer and a shadow flip-flops inserted for the interface register, doubling the size of the interface scan cell. The newly inserted logic falls on the test paths only, incurring to timing penalty whatsoever.. The total cost is Nint MUXes and Nint + 2 flip-flops for Nint interface registers.

B. DFT Support and Implementation for LOS Testing:

As the LOS scheme launches transitions via a shift operation, a set of test patterns is valid as long as the final scan cell ordering in the chain perfectly matches that during test generation. Therefore, LOS pattern generation should be done subsequent to scan stitching in conventional LOS. The only additional constraint imposed on scan stitching by the proposed partitioning scheme is that the interface registers of each region should be placed in consecutive positions on the scan chain and that they must be stitched in a bidirectional manner. Such a special stitching and the associated DFT support are required only for the interface registers in order to enable a proper rewind operation; minimization of the number of interface registers helps to minimize the area cost incurred. Finally, restoring the value of the rightmost bit of a group of interface registers subsequent to the launch operation necessitates an extra flip-flop, which holds the value of the rightmost interface bit upon launch; a subsequent rewind operation restores the value of the rightmost interface register from the value in this extra flip-flop. Bidirectional stitching of the interface registers in a region requires one additional multiplexer for each interface register; this multiplexer can be inserted on the scan path imposing no impact on the functional timing of the design whatsoever. The proposed scan architecture that supports design partitioning into two regions fig 2, illustrating the simple and cost-effective on-chip generation of the rewind signal out of the scan-enable and clock signals. The total cost is *N*int MUXes and *R*int + 1 flip-flops for *N*int interface registers and *R*int regions, which have at least one interface register.

II. EXISTING METHOD

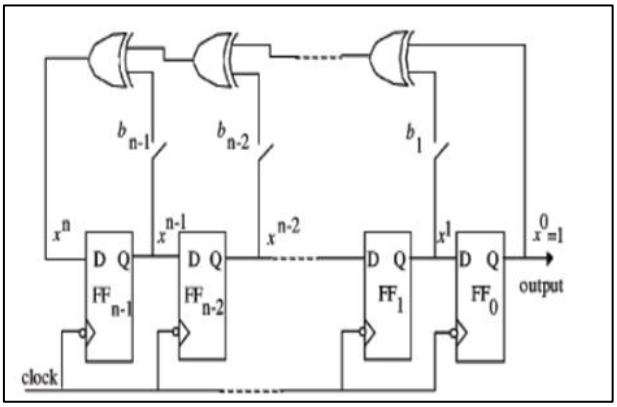

Fig. 2: Simplified circuit of a generic LFSR circuit

In the existing method the LFSR technique is used to generate the test patterns. They present the gated clock design approach for LFSRs which can lead to power reduction without unduly complicating the traditionally simple topology. The analysis demonstrated that the power reduction hardly depends on the technological characteristic of the gates employed. Finally, the real efficiency of the presented gated-clock design should be evaluated also considering the increased silicon area required to implement the gating circuit.

Linear Feedback Shift Register or LFSR is a shift register whose input is the result of XOR of some of its inputs. There are two ways to implement LFSRs: Internal feedback and External feedback. These techniques differ in the way feedback is applied. All the flip-flops that feed an XOR gate are known as 'taps'. These taps decide the patterns generated by the LFSR and hence define the characteristic polynomial of an LFSR. In case of an external feedback LFSR the XOR gates are in the feedback path and the input to the shift register is the XOR of all the taps.

For an internal feedback LFSR, the feedback from the last FF is the input to the first FF of the shift register and all the taps are XORed with the feedback to modify the input to the next FF in the shift register. An internal feed-back LFSR can also operate at higher speeds compared to an External feedback LFSR as there is maximum one XOR gate in any path between FFs, which is not the case for External feedback LFSR.

A. Disadvantage of Using LFSR Technique:

There are some undesirable properties of the pseudo-random patterns generated by LFSR, these properties of alternative ways to generate test patterns for BIST implementation. One of the biggest drawbacks of LFSR is the inability to efficiently generate test patterns which can detect stuck-open faults in CMOS and sequential faults such as delay faults in combinational circuits.

III. PROPOSED TECHNIQUE

A. Fault Injection with Saboteur:

A saboteur is a special VHDL component added to the original model. When activated, the mission of this component is to alter the value, or timing characteristics, of one or more signals, simulating the occurrence of a fault. During the normal operation of the system, instead, the component remains inactive. Saboteurs affect to the ports of the components in the model. Thus, this technique is applicable only to structural descriptions. Attending to how saboteurs are inserted in the model, two types can be distinguished: serial and parallel. A serial saboteur interposes between a component input port and its source signal whereas a parallel saboteur is added as an additional source of a given signal. Parallel saboteurs have two important drawbacks respect to serial. First, implementing them is noticeably more complex, because it is necessary to modify the data type of the signal affected, as well as the resolution function associated to the data type (a resolution function defines how values from multiple sources are resolved into a single value). Second, they allow to inject fewer fault models. For these reasons, their implementation has no special interest.

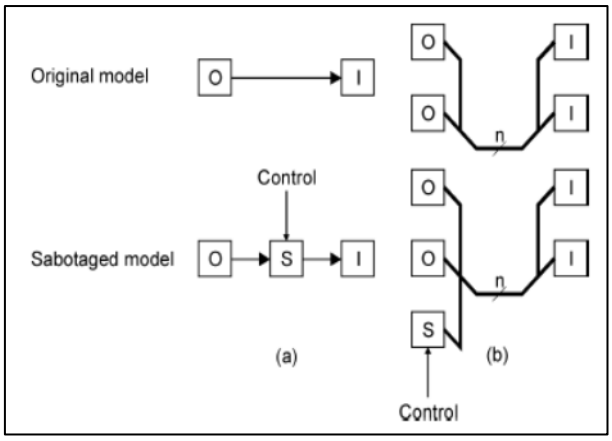

Fig. 3: Types of saboteurs. (a) Serial. (b) Parallel

B. Fault Injection with Mutants:

A mutant is a component that replaces another component. While inactive, it works like the original component, but when it is activated, it behaves like the component in presence of faults.

The mutation can be made in three ways:

- by adding saboteurs to structural model descriptions;
- by modifying structural descriptions replacing sub-components
- by modifying syntactical structures of behavioral descriptions.

There can exist lots of possible mutations in a VHDL model, so representative subsets of faults at logical and RT levels must be considered replacing the values of conditions in if and case statements, disturbing assignment statements, disturbing operators in expression.

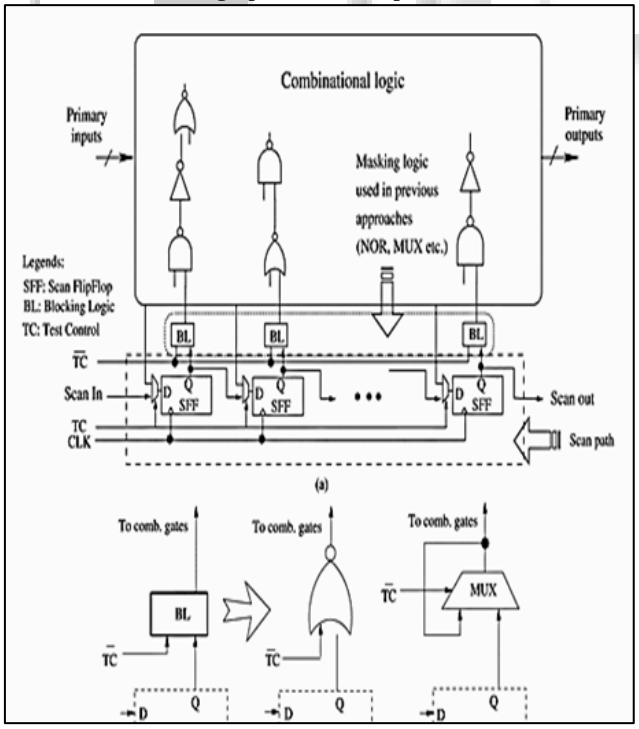

Fig. 4: Scan architecture with existing blocking circuitry to reduce power during scan operation and blocking logic.

Considered the following fault models

- Stuck-Then: Replacement of the if condition by true.
- Stuck-Else: Replacement of the if condition by false.
- Assignment Control: Disturbing an assignment operation.
- Dead Process: Elimination of the sensitivity list of a process.
- Dead Clause: Elimination of a clause in a case.
- Micro-Operation: Disturbing an operator.
- Local Stuck-Data: Disturbing the value of a variable, constant, or signal in an expression.
- Global Stuck-Data: Elimination of all value modifications of a variable or signal in architecture.

Many of these fault models do not have a direct correspondence with physical faults, but they can show somehow an erroneous internal operation This new proposal to implement mutants is so simple that automating the generation of mutants of a given model is not complicated at all. Assuming that an injection tool has a parser, locating in the code the target statements to be mutated and replacing them with new ones is very easy.

On the one hand, to declare the signal that selects the mutation to activate at injection time. On the other hand, to replace the original components with the mutated ones; this affects to both the component declaration and instantiation.

C. Scan Chain Reordering:

Scan chain reordering is a process used in the design and testing of computing devices that enables the optimization of placing and stitching flip flop registers with a scan chain. It is used to optimize and reorder the scan chain process if it gets detached, stopped or congested. Scan chainreordering is primarily used in the design of integrated circuits with a series of registers within them. The registers are placed using a design placement tool that stitches registers on a board under a systematic scan chain. However, during placement, the length of the overall wire/metal connecting these registers might extend, which creates congestion. Scan chain reordering is then applied by the design placement tool, which optimizes the stitching process and reduces the overall wire and metal required. We consider four test-cases and their corresponding responses.

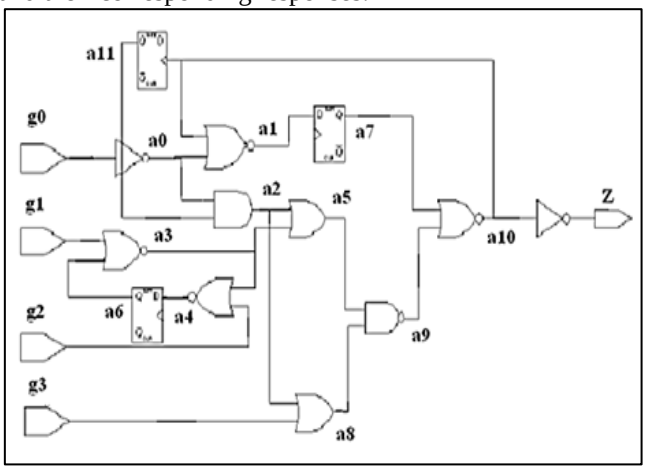

Fig. 5: S27 Benchmark Circuit

The vertices in the graph represent the flip-flops and the edges represent the transitions between each pair of flip-flops. The edge weights represent the total bit-difference between them as obtained from the table. Hence

the problem reduces to determination of a Hamiltonian cycle in the constructed acyclic graph. This problem is Traveling Salesman Problem, which is well known to be NP-hard (the number of possible solutions is (n-1)!/2 , n being the scan chain length). Among the number of polynomial-time approximation algorithms, we choose the greedy algorithm. The greedy algorithm starts from any scan cell which is ff1 in our case, as the choice of initial state is not so crucial when the number of scan cells in the graph is considerably high. From any given node the algorithm chooses the least weighted edge leading to an undiscovered flip-flop. The final series in this case is ff1-ff4-ff2-ff3-ff1. The complexity of the algorithm is O(n2), as for each node all its incident edges are considered.

IV. SIMULATION RESULT

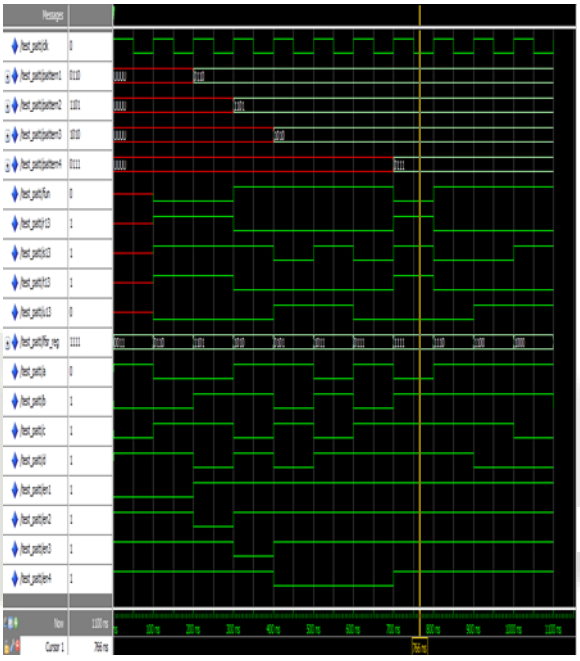

Fig. 6: Simulation result of scan chain reordering, saboteurs and mutants method.

	Using LOC-LOS	**After implementing Scan Chain reordering method**	**After pattern generation by Saboteur and mutants**
Time	6.910ns	6.710ns	3.104ns
Area	268	268	225
Power	141mw	138mw	54mw

Table 1: Comparison of existing and proposed technique

V. CONCLUSION

Launch Off Shift and Launch Off Capture techniques are used to reduce launch and capture power. In the existing method test patterns are generated by using LFSR technique. In the proposed technique we implement saboteur and mutants method to generate test patterns. Scan chain reordering method is used to reduce number of transitions between the flip-flops and thus the power consumption and area are reduced.

VI. FUTURE WORK

The future work is to use LOES (Launch Off Extra Shift) and using LOEC (Launch Off Extra Capture) for further launch and capture power reduction. The project will be implemented in tanner tool for exact power calculation.

REFERENCES

[1] Samah Mohamed Saeed, Ozgur Sinanoglu, "Design for Testability for Launch and Capture Power Reduction in Launch-Off-Shift and Launch-Off-Capture," in Proc. IEEE Transaction on Very Large Scale Integration,VOL.22,NO,3. Mar. 2014.

[2] K. Miyase, Y. Uchinodan, K. Enokimoto, Y. Yamato, X. Wen, S. Kaji- hara, F. Wu, L. Dilillo, A. Bosio, P. Girard, and A. Virazel, "Effective launch-to-capture power reduction for LOS scheme with adjacent- probability-based X-Filling," in Proc. 20th Asian Test Symp., Nov. 2011, pp. 90–95.

[3] Z. Chen, K. Chakrabarty, and D. Xiang, "MVP: Capture-power reduction with minimum-violations partitioning for delay testing," in Proc. IEEE ACM Int. Conf. Comput.-Aided Design, Nov. 2010, pp. 149–154.

[4] Juan-Carlos Baraza, JoaquínGracia, Sara Blanc, Daniel Gil, and Pedro-J. Gil, "Enhancement of Fault Injection Techniques Based On the Modification of VHDL Code," in Proc. IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS, VOL. 16, NO. 6, JUNE 2008.

[5] Miyase.K, Uchinodan.Y, Enokimoto.K, Yamato.Y, Wen.X, Kaji-hara.S, Wu.F, Dilillo.L, Bosio.A, Girard.P, and Virazel.A, "Effective launch-to-capture power reduction for LOS scheme with adjacent- probability-based X-Filling," in Proc.20th Asian Test Symp.,pp.90-95, JUNE 2011.

[6] PavelBartos, Juan A. Maestro, and Mark F. Flanagan"TESTTIMEREDUCTIONBYSCANCH AINREORDERING," in Proc. IEEE Trans. Very Large Scale Integration (VLSI) Systems, Vol. 21, No. 1, pp. 340-359, 2004.

[7] Praveen Kumar Aggarwal, VandanaYadav and Arti Noor"DFT (Design for Testability) Pattern Generation Task for Circuit Under Test," International Journal of Engineering Research and Applications (IJERA),ISSN:2248-9622, Vol. 1, Issue 2, pp. 190-193,2003.

[8] Samah Mohamed Saeed and Ozgur Sinanoglu"Design for Testability Support for Launch and Capture Power Reduction in Launch-Off-Shift and Launch-Off-Capture Testing," IEEE Transactions on Very Large Scale Integration (VLSI) systems,Vol 22 2014.

[9] Sridhar.D, AvinashKumar.K and Krishna Rao.P"VHDL Implementation of a low power fault tolerant system," International Journal of Modern Engineering Research (IJMER) Vol.2, Issue.3, pp-906-916, 2012.

[10] Subhramanyam.N, AmbavaramPoli Reddy, Rajpraveen.J"Fault Detection for ISCAS 89' S-27

Benchmark Circuit Using Low Power Lt-RTPG," International Journal of Engineering Research & Technology (IJERT) Vol. 2 Issue 1,ISSN:2278-0181,2013.

[11] SwarupBhuniaZ, Hamid Mahmoodi, DebjyotiGhosh, SaibalMukhopadhyay and Kaushik Roy (2005) "Low-Power Scan Design Using First-Level Supply Gating," in Proc. IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS,VOL.13, NO. 3.

