An efficient Self-test scheme for Random Logics Interconnects

Dongchul Kang*, Hyomoon Cho*, Dongkyun Park*, Sangbock Cho**

School of Electrical Engineering, University of Ulsan
San-29, Muger-Dong, Nam-Gu, Ulsan, Korea

kangdc05@yahoo.co.kr, hmcho67@ulsan.ac.kr

Abstract

The interconnections are critical determinants of performance, reliability and power, for deep sub-micron system on SoC (System on a Chip). Long interconnects being susceptible to crosstalk noise, may induce to functional and timing faults. The conventional at-speed interconnect crosstalk test algorithms are based on either inserting dedicated interconnection self test structures (leading to significant area overhead), or using existing logic BIST structures (e.g., LFSRs), which often result in poor defect coverage. Additionally, it has been shown that power consumed during testing can potentially become a significant concern. In this paper, we introduce the logic interconnect BIST, a wide-used self test solution for random logics. This method reuses existing LFSR structures but generates high quality test efficiency for inter-connection crosstalk, while minimizing area over-head and interconnection power consumption. This method achieved crosstalk defect coverage of 99.7% for the interconnections and single stuck-at-fault coverage of 91.36% for the random logics, while area overhead is increased of 4% for con-ventional methods.

I. Introduction

As SoC (System on a Chip) complexity grows with increasing integration and reducing feature sizes, the on-chip interconnect architecture per-forms a much more critical role because it domi-nates system performance and power consumption [8], [11]. Reliability of SoCs depends on the error-free operation of such interconnects. Testing of SoCs, hence, implies testing not only the logic cores but also the interconnect architecture. Several Hence, we need to address the crosstalk issue by means of testing techniques. Increased cross coupling capacitance between a pair of inter-connects can produce either glitches or delays depending upon the signal transitions at an interconnections. Also, previous studies have shown that crosstalk noise is more pronounced for long interconnects [5, 9].

In this paper, therefore, we employ self-testing techniques to address the problem of testing for crosstalk in SoCs to develop a comprehensive test solution for both the logic cores as well as buses and global interconnects of a SoC. The empirical studies have shown that the power dissipation associated with long interconnects accounts for a significant fraction of the overall system power [8]. This power consumption is dominated by the increasing inter-wire cross coupling capacitances in DSM technology [12]. The energy dissipated due to cross-coupling capacitances can vary depending on the type of transitions on the interconnections [12]. We, therefore, also focus on making our self-test scheme extremely power-efficient.

Bai et al, have proposed inserting dedicated interconnect self-test structures in the SoC to generate vectors which have 100% crosstalk defect coverage [2]. This scheme is based on the Maximal Aggressor Fault Model proposed by Cuviello et al [5]. However, this method has a prohibitively high area overhead. To reduce this overhead, existing logic BIST structures, like linear feedback shift registers (LFSRs), could be reused to generate interconnect tests. But, LFSR vectors have very poor crosstalk defect coverage. In this paper, we address the issue of how to generate high quality interconnect crosstalk tests, efficiently reusing existing test structures so as to minimize the area overhead. Our proposed algorithm produce high-

crosstalk defect coverage with low area overhead and low interconnect power consumption.

The interconnect fault model used in this paper is the Maximal Aggressor Fault Model that was reported and validated in [5]. Next, we briefly review this fault model, and the corresponding Maximal Aggressor test vectors.

II. New Test Generator Design and Logic Interconnect BIST Test Architecture

The test generator should be designed so as to maximize the reuse of existing self-test structures (LBIST) so that additional test circuitry area is minimized. The new test generator should achieve as high logic fault coverage as existing logic BIST schemes. Also, it should produce high quality interconnect crosstalk tests. Hence, the profile of the interconnect test vectors produced by the test generator should be such that the majority of the wires transition in the same direction (to act as aggressors) and relatively few wires remain 0/1 or transition in the opposite direction (to act as victims) so as to cause a high cross-coupling capacitive effect on the victim wires. The vectors generated by the test generator should be interconnected power efficient. Hence, we should minimize opposite direction transitions on adjacent wires, so that the total amount of coupling capacitances excited is low. Figure 1 shows the structure of a 4 bit logic interconnect BIST test generator.

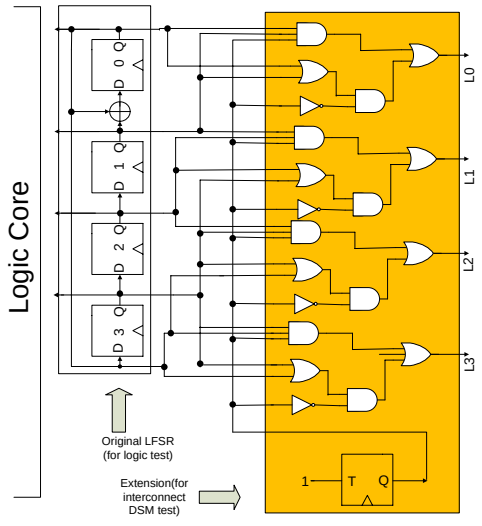

Figure 1. A 4-bit TG for logic interconnect BIST

It consists of an LFSR that produces vectors for testing the logic core (as in traditional logic BIST), and an extension circuit that modifies the LFSR vectors such that they have high interconnect crosstalk defect coverage. In the extension circuit, the adjacent bits of the LFSR are both ANDed and ORed. Figure 2 illustrates the test architecture of logic interconnect BIST on an example SoC. The core is surrounded by a Test Wrapper, which consists of self-test structures like a Test Pattern Generator (TPG) and a MISR. The TPG generates test vectors for both the logic core as well as the interconnections. MUXs are used to select between the normal core outputs and the interconnect test vectors. The MISR makes compact the output responses of both the core and interconnects; this is selected by MUXs. There is a centralized logic interconnect BIST controller which controls the whole test sequence. It gives control signals to all the test structures and is also responsible for seeding the TPG and unloading the MISR signature.

There are three operational modes of the SoC; normal mode, core test mode and interconnect test mode. In the normal mode, the SoC functions in its normal system operational mode. During the core test mode, the internal logic of the cores is tested via the TPG and the MISR, as in logic BIST. In the interconnect test mode, for each core-to-core transaction, the vectors generated by the TPG of the source core are routed onto interconnect and they are compacted by the MISR of the destination core.

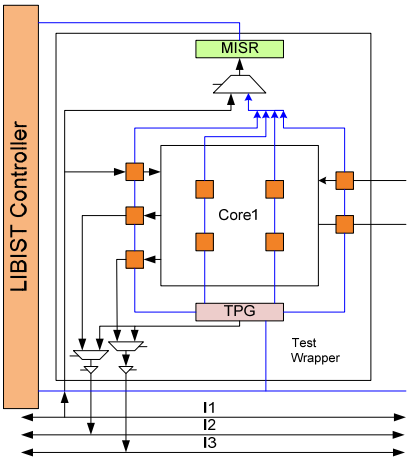

Figure 2. Test architecture of Logic interconnect BIST

III. Experimental Results and conclusion

Table 1 compares the three schemes in terms of the additional area overhead over LBIST, the interconnect crosstalk defect coverage, and the logic fault coverage. The values for the MA Test scheme are reported from [2]. We used Synopsys' Design Compiler [6] synthesis tool to synthesize CMUDSP along with the inserted test structures. The area overhead is in terms of the additional number of gates (over conventional LBIST) required to test interconnects of the chip. The defect coverage is measured using the high-level fault-simulation methodology described in the previous sub-section. The logic fault coverage is in terms of single stuck-at-fault coverage and is measured using Mentor Graphics' FastScan [7]. The logic interconnect BIST scheme, however, achieves very high defect coverage with very low area overhead. Table 1.

Table 1. Comparison of proposed with MA and LBIST

Test scheme	Area overhead	Defect coverage	Fault coverage	Power
MA	22%	100%	-	0.43mW
LBIST	1.22%	1%	91.36%	3.66mW
Proposed	4%	99.7%	91.36%	0.77mW

IV. Conclusion

In this paper, we presented logic interconnect BIST, a comprehensive SoC test methodology for both logic cores and interconnects. It efficiently reuses existing on-chip test structures to generate high quality interconnect crosstalk tests. We have compared our proposed BIST with existing solutions and validated it using a high level crosstalk error model. Experiments confirm that our proposed algorithm yields high crosstalk defect coverage, low area overhead and low interconnect power consumption.

Acknowledgement

This work was partly sponsored by NARC (Network-based Automation Research Center) and 2 level BK21 (EVERDEC(e-Vehicle Research & human Resource Development Center)) in MOE (Ministry of Education & Human Resources Development).

References

[1] X. Bai and S. Dey. High-level crosstalk defect simulation for system on-chip interconnects. In Proc. IEEE VLSI Test Symposium, Apr. 2001.

[2] X. Bai, S. Dey, and J. Rajski. Self-test methodology for at-speed test of crosstalk in chip interconnects. In Proc. Design Automation Conf., p 619 624, June 2000.

[3] Z. Chen and I. Koren. Crosstalk minimization in three-layer HVH channel routing. In Proc. IEEE Int. Symp. on Defect and Fault Tolerance in VLSI Systems, p 38-42, 1997.

[4] The Carnegie Mellon Synthesizable Digital Signal Processor Core (http://www.ece.cmu.edu/lowpower /benchmarks.html).

[5] M. Cuviello, S. Dey, X. Bai, and Y. Zhao. Fault modeling and simulation for crosstalk in system-on-chip interconnects. In Proc. Int. Conf. CAD, p 297-303, Nov. 1999.

[6] Design Compiler 2000.05, Synopsys Inc. (http://www.synopsys.com).

[7] FastScan v8.6 4.5, Mentor Graphics Corp. (http://www.mentor.com).

[8] R. Mehra, L. M. Guerra, and J. M. Rabaey. A partitioning scheme for optimizing interconnect power. IEEE J. Solid-State Circuits, 32(3):433 443, Mar. 1997.

[9] P. Nordholz, D. Treytnar, J. Otterstedt, H. Grabinski, D. Niggemeyer, and T.W.Williams. Signal integrity problems in deep submicron arising from interconnects between cores. In Proc. IEEE VLSI Test Symposium, p28-33, April 1998.

[10] K. Rahmat, J. Neves, and J. Lee. Methods for calculating coupling noise in early design: a comparative analysis. In Proc. Int. Conf. Computer Design VLSI in Computers and Processors, p76-81, 1998.

[11] Semiconductor Industry Association. Inter. Tech. roadmap for semiconductors, 1999.

[12] C. N. Taylor, S. Dey, and Y. Zhao. Modeling and minimization of interconnect energy dissipation in nanometer technologies. In Proc. Design Automation Conf., June 2001.