ISSN No. 0976-5697

Volume 3, No. 2, March-April 2012

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Fault Diagnosis In Vlsi Chips Using Different Strategies

Divya Sharma
FEI&CE
Shobhit University Meerut, India
divya2318@yahoo.com

Dr B. K. Kaushik
IITR,
Roorkee India
brajesh_k_k@yahoo.com

Dr R. P. Agarwal
Vice Chancellor
Shobhit University Meerut, India
prajanag@gmail.com

Abstract: In the recent years integrated circuit technology has evolved rapidly due to various innovations in computers, IT, electronics, medical, etc. The complex geometry of interconnects and high operational frequency introduce wire parasitic and inter-wire parasitic. These parasitic causes delay, power dissipation and crosstalk that may affect the signal integrity in VLSI system. Accurate analysis, sophisticated design, effective test methods and diagnostic algorithms are the requirement to ensure the proper functionality and reliability of VLSI circuits. New fault diagnostic techniques and algorithms are required which can deduct large number of faults in less time. Here is a discussion about various strategies for fault diagnosis.

Keywords: interconnects, faults, fault diagnosis, SoCs and NoCs

I. INTRODUCTION

Advances in VLSI technology have increased the density and speed of integrated circuits. Thus, the complexity and cost of testing digital integrated circuits, boards, and systems have also increased. By providing a simple means to access the periphery of digital circuits, boundary scan can greatly simplify the task of testing and maintaining systems which use these circuits. This advantage allows boundary scan to reduce the costs of wafer-level IC testing, board and system testing, and system field maintenance.

The complex geometry of interconnects and high operational frequency introduce wire parasitic and inter-wire parasitic. These parasitic causes delay, power dissipation and crosstalk that may affect the signal integrity in VLSI system. Accurate analysis, sophisticated design, effective test methods and diagnostic algorithms are the requirement to ensure the proper functionality and reliability of VLSI circuits. The testing of interconnect is becoming important and a challenge in the current technology. Testing of interconnect is important and emerging challenge in the nanotechnology era. Although some work has been done for testing of interconnect however, it is still an open area to test the effects of VLSI/ULSI interconnects. Efforts are required to analyze and develop test methods in current technology with solutions to minimize defects.

Here is the problem of fault diagnosis in a nutshell: a circuit has failed one or more tests applied to it; from this failing information, determine what has gone wrong. The evidence usually consists of a description of the tests applied, and the pass-fail results of those tests. In addition, more detailed per-test failing information may be provided. The purpose of fault diagnosis is to logically analyze whatever information exists about the failures and produce a list of likely fault candidates. These candidates may be logical nodes of the circuit, physical locations, defect scenarios (such as shorted or open signal lines), or some combination thereof.

II. DIAGNOSTIC ALGORITHMS

Diagnosis algorithms have traditionally been classified into two types, according to how they approach the problem. The first and by far the most popular approach is called cause-effect fault diagnosis. A cause-effect algorithm starts with a particular fault model (the “cause”), and compares the observed faulty behavior (the “effect”) to simulations of that fault in the circuit. A simulation of any fault instance produces a fault signature or a list of all the test vectors and circuit outputs by which a fault is detected, and which can be in one of the signature formats described earlier.

The opposite approach, and the second classification of diagnosis algorithms, is called effect-cause fault diagnosis [1, 2]. These algorithms attempt the common-sense approach of starting from what has gone wrong on the circuit (the fault “effect”) and reasoning back through the logic to infer possible sources of failure (the “cause”). Most commonly the cause suggested by these algorithms is a logical location or area of the circuit under test, not necessarily a failure mechanism.

The following section present algorithms for VLSI diagnosis proposed by previous researchers, from the early 2001 to the present day. In general, the earliest algorithms have targeted solely stuck-at faults and associated simple defects, while the later and more sophisticated algorithms have used more detailed fault models and targeted more complicated defects.

A. *Scan-Based BIST Fault Diagnosis:*

Here the authors give an overview of diagnostic algorithms in the connection of using in BIST [3, 4, 5, 6, 7]. Consider the BIST environment consisting of the Circuit under Test (CUT), the Pseudorandom Test Pattern Generator (TPG) and the Multiple Input Signature Register (MISR) as an Output Response Analyzer (ORA) as depicted in Fig. 1. Denote by N the length of the pseudorandom test sequence T generated by TPG, by F the set of possible faults in the CUT by $F(t) \subset F$ the set of faults detected by the test

© 2010, IJARCS All Rights Reserved

pattern t ε T, and by T(f) ⊆ T the set of test patterns failed because of the fault f ε F. Let us call by a test session (query) the procedure where a part of test sequence T is applied with the subsequent comparison of the signature in MISR with the expected reference value. The diagnosis problem can be formulated as follows: given a set F of faults, identify the subset of faults F* ⊆ F, where in general case, the number of faults to be localized d = |F*| is unknown, using the minimum number of queries. (The number of queries is directly proportional to the amount of time needed to diagnose the BIST system).

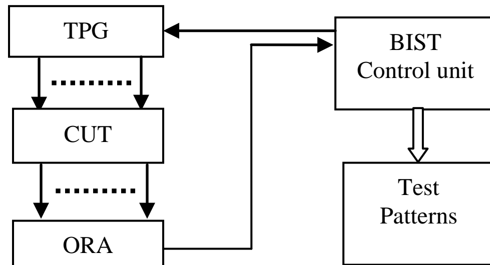

Figure. 1 BIST for Fault Diagnosis

B. Pseudo-Online Testing Methodologies for Various Components of Field Programmable Gate Arrays:

This paper by L. Kalyan Kumar, Aditya S. Ramani, Amol J. Mupid, V. Kamakoti, presented novel pseudo-online algorithms [8, 9, 10] for detecting and locating multiple faults in LUTs, interconnects and dedicated clock lines in FPGAs. The proposed algorithms used the pseudo-exhaustive BIST technique, which preserved the interconnect structure of the LUT network in-place. This is crucial for any online testing strategy as it ensures that the non-faulty parts of the system continue to function while the faulty sub circuit is diagnosed. Extending the proposed methods for sequential LUT networks with multiple faults is a very challenging open problem. A more elaborate fault model and efficient routing techniques for fault repair are interesting open issues.

C. Design and Implementation of Built-in-Self Test and Repair:

Survey of ITRS in 2001, the System-on-Chips (SoCs) is moving from logic dominant chips to memory dominant chips in order to deal with today's and future application requirements. The dominating logic (about 64% in 1999) is changing to dominating memory (approaching 90% by 2011). These shrinking technologies give rise to new defects and new fault models have to be defined to detect and eliminate these new defects. These new fault models are used to develop new high coverage test and diagnostic algorithms. The greater the fault detection and localization coverage, the higher the repair efficiency, hence higher the obtained yield. Memory repair is the necessary, since just detecting the faults is no longer sufficient for SoCs, hence both diagnosis and repair algorithms are required. March SS algorithm is a newly developed test algorithm that deals with detecting some recently developed static and dynamic fault models**.** A new micro coded BIST architecture in fig 2 is presented here which is capable of employing these new test algorithms. A word-oriented BISR array [11, 12, 13, 14] is used to detect the faulty memory locations and repair those faulty memory locations. As indicated by the BIST controller. The MBISR logic used here can function in two modes. Mode 1: Test & Repair Mode, Mode 2: Normal Mode. The BISR Control Circuitry consists of Clock Generator, Instruction Pointer, Microcode Instruction storage unit, Instruction Register. The Test Collar circuitry consists of Address Generator, RW Control and Data Control, Redundancy Logic array, Input multiplexer, Output multiplexer and Memory.

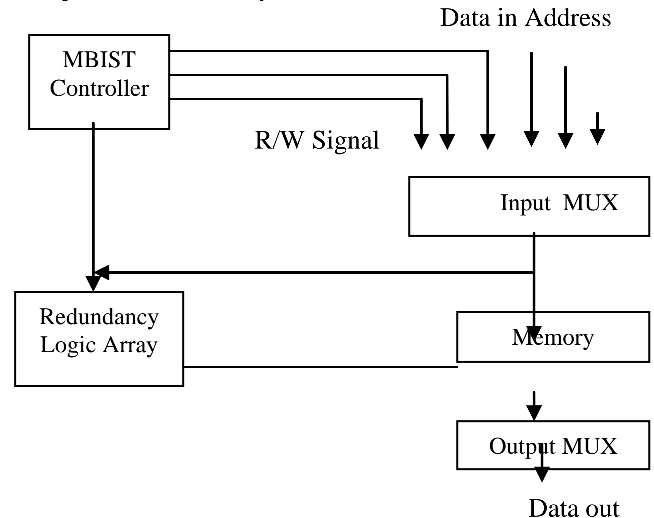

Figure 2 Block Diagram of BISR

D. Selection of a Fault Model for Fault Diagnosis Based on Unique Responses:

In this paper [15, 16] the authors described a procedure for selecting a fault model that is likely to be effective for diagnosing a given observed response of a faulty chip out of a given set of fault models. This procedure can be applied as a preprocessing step to fault diagnosis. After the appropriate fault model is selected, the fault diagnosis can proceed with the selected fault model. Such a preprocessing step is important since the accuracy of fault diagnosis can be improved significantly if the correct fault model is used. They described a specific implementation of this preprocessing step based on the unique output responses of a fault model. Considering two fault models, the unique output response of one model consists of the output vectors, which faults of this model can produce, while faults of the other model cannot produce. To select a fault model for an observed response, they found the number of output vectors in the observed response that match the unique response of each fault model. The fault model with the higher number of matches was selected to perform fault diagnosis. They applied this preprocessing step to the diagnosis of multiple stuck-at faults, selecting between single and double stuck-at faults as the fault model for diagnosis. They discussed the use of a subset of double stuck-at faults for diagnosis. Experimental results demonstrated that with a properly selected subset of double stuck-at faults, it is possible to improve the results of fault diagnosis compared to the case where only single stuck-at faults are considered. They also discussed the application of the proposed preprocessing step with single stuck-at, bridging, and transition faults.

E. Online Network-on-Chip Switch Fault Detection and Diagnosis Using Functional Switch Faults:

This paper [17] presented online fault detection and fault location method for NoC switches. Various forms of

© 2010, IJARCS All Rights Reserved

functional switch faults were considered in this research, including dropped and corrupted data faults, direction faults, and faults resulting in multiple copies of packets in time and space. For each of these faults an error detection and diagnosis method has been proposed. The proposed algorithms [18, 19, 20 21, 22] have been evaluated in terms of fault coverage and the necessary area overhead. The experimental results show that with a relatively low area overhead, a large number of NoC switch faults can be detected and diagnosed. The steps taken in this work are essential for design of reliable NoC structures.

III. CONCLUSION

A paper by Aitken and Maxwell identifies two main components to any fault diagnosis approach. The first is the choice of fault model, and the second is the algorithm used to apply the fault model to the diagnostic problem. As the authors explain, the effectiveness of a diagnostic technique will be compromised by the limitations of the fault model it employs. So, for example, a diagnosis tool that relies purely on the stuck-at fault model can never completely or correctly diagnose a signal-line short or open, simply because it is looking for one thing while another has occurred.

The authors go on to explain that the role of the diagnosis algorithm, then, has evolved to try to overcome the limitations of the chosen fault model. A common technique is to use the stuck-at model but adjust the algorithm to anticipate bridging-fault behaviors. But, the authors also opened a debate, which remains active to this day: is it better for a diagnosis technique to use more realistic fault models with a simple algorithm, or to use simple and abstract models with a more clever and robust algorithm?

As with any interesting debate, there are good arguments on both sides. The argument for simple fault models is that they are more practical to apply to large circuits and more flexible for a wide variety of defect behaviors. The argument for better models, taken by the authors in their original paper, is that good models are necessary for both diagnostic accuracy and precision. Simple models do not provide sufficient accuracy because defect behavior is often complex, more complex than even clever algorithms anticipate. They also do not result in sufficient precision because they do not provide enough specificity to guide effective physical failure analysis.

Research in nano-particles creates possibilities for both new materials and new processes. Fault diagnosis algorithms distinguish between modeling, uncertainties, normal process disturbances and real faults. The effectiveness of the presented methods mainly depends on the number of failing patterns and their location in the sequence of test patterns. By providing a simple means to access the periphery of digital circuits, boundary scan can greatly simplify the task of testing and maintaining systems which use these circuits. Thus in some cases one algorithm is better than other and conversely.

IV. REFERENCES

[1]. M. Abramovici and M. A. Breuer. Multiple fault diagnosis in combinational circuits based on an effect-cause analysis. IEEE Transactions on Computing, Vol. C-29, pages 451-460, June 1980.

[2]. V.M. Becerra, Advanced System Identification, "Fault Diagnosis using System Identification Techniques", CY4A2, Lecture 8, www.alexlab.net/mywiki/.../fetch.php

[3]. Y. Wu, S.M.I. Adham, Scan-Based BIST Fault Diagnosis, IEEE Trans. Computer-Aided Design 18 (2) (1999) 203–211.

[4]. K. De and A. Gunda. Failure analysis for full-scan circuits. Proceedings of the International Test Conference, pages 636-645, IEEE, 1995.

[5]. C. Liu, K. Chakrabarty, Failing Vector Identification Based on Overlapping Intervals of Test Vectors in a Scan-BIST Environment, IEEE Trans. CAD IC Syst. 22 (4) (2003) 593–604.

[6]. S. Pateras, Embedded Diagnosis IP, DATE (2002) 242–244.

[7]. P. Wohl et al., Effective Diagnostics through Interval Unloads in a BIST Environment, IEEE/ACM DAC (2002) 249–254.

[8]. L. Kalyan Kumar, Aditya S. Ramani, Amol J. Mupid, V. Kamakoti Pseudo-online testing methodologies for various components of field programmable gate arrays. Microprocessors and Microsystems 29 (2005) 99–119.

[9]. J. Jacob and N.N. Biswas. GTBD faults and lower bounds on multiple fault coverage of single fault test sets. Proceedings of the International Test Conference, pages 849-855, IEEE, 1986.

[10]. S. Hamdioui, A.J. van de Goor and M. Rodgers, "March SS: A Test for All Static Simple RAM Faults", In Proc. of IEEE International Workshop on Memory Technology, Design, and Testing, pp. 95-100, Bendor, France, 2002.

[11]. P. Ravinder and N. Uma Rani, "Design and Implementation of Built-in-Self Test and Repair", in International Journal of Engineering Research and Applications, Vol. 1,Issue 3, pp. 778-785.

[12]. H.-J. Wunderlich, From embedded test to embedded diagnosis, in: IEEE 10th European Test Symposium, Tallinn, 2005, pp. 22–25.

[13]. R. Aitken and P. Maxwell. Better models or better algorithms? On techniques to improve fault diagnosis. Hewlett-Packard Journal, February 1995.

[14]. S. Hamdioui, Z. Al-Ars, A.J. van de Goor, "Testing Static and Dynamic Faults in Random Access Memories", In Proc. of IEEE VLSI Test Symposium, pp. 395-400, 2002.

[15]. Irith Pomeranz, and Sudhakar M. Reddy, "Selection of a Fault Model for Fault Diagnosis Based on Unique Responses" IEEE transactions on very large scale integration (VLSI) systems, vol. 18, no. 11, November 2010.

[16]. T. Bartenstein, D. Heaberlin, L. Huisman, D. Sliwinski. Diagnosing Combinational Logic Designs Using the Single Location At-a-Time (SLAT) Paradigm. Proceedings of the International Test Conference, pages 287-296, IEEE, 2001.

[17]. N. Karimi, A. Alaghi, M. Sedghi, Z. Navabi, "Online Network-on-Chip Switch Fault Detection and Diagnosis Using Functional Switch Faults", Journal of Universal Computer Science, vol. 14, no. 22 pp 3716-3736 (2008).

© 2010, IJARCS All Rights Reserved

[18]. Marculescu, R., Ogras, U. Y., and Zamora, N. H.: Computation and Communication Refinement for Multiprocessor SoC Design: A System-Level Perspective, ACM Trans. on Design Automation of Electronic Systems, Special Issue on Novel Paradigms in System-Level Design, Vol. 11, No. 3, 2006, 564-592.

[19]. Mello, A., Tedesco, L., Calazans, N., and Moraes, F.: Virtual Channels in Networks on Chip: Implementation and Evaluation on Hermes NoC, In Proc. Integrated Circuits and Systems Design, 2005, 178-183.

[20]. Orgas, U. Y., and Marculescu, R.: Communication-Based Design for Nanoscale Socs, VLSI Handbook, Wai-Kai Chen(ed.), 2nd Edition, CRC Book Press, 2006, Chapter 16.

[21]. Jantsch, A., and Tenhunen, H., Networks on Chip, Kluwer Academic Publishers, 2003.

[22]. Hansson, A., Goossens, K., and Radulescu, A.: A Unified Approach to Constrained Mapping and Routing on Network-on-Chip Architectures, In Proc. 3rd Hardware/Software Codesign and System Synthesis (CODES+ISSS ’05), 2005, 75–80.

© 2010, IJARCS All Rights Reserved