Built in Self Diagnosis for Logic Blocks in Application Specific FPGA

Kavya V

***Abstract*- Field Programmable Gate Array (FPGAs) are widely used in applications such as networking, storage systems, communication, and adaptive computing. This is due to the reprogrammability, flexibility and reduced time to market of FPGAs. But the recent sub micron technology results in low yield FPGA with increased number of faults in it. To improve the yield, faulty FPGAs can be utilized for application specific designs. An application, 8-bit multiplier is tested using optimized BIST by mapping on SPARTAN III FPGA.**

***Index term:* Built in self test (BIST), Fault Tolerance (FT), Fault Diagnosis(FD), Linear Feedback Shift registers (LFSR).**

I. INTRODUCTION

Application Specific Integrated Circuits (ASIC) architecture depends on the application while FPGA design is independent of application. Because of this flexibility FPGAs have become most popular among integrated circuits. Since speed is the key factor in most of the applications, FPGAs have advanced towards sub-micron technology. Reduction in size of the device usually leads to different kinds of fault.

Faulty FPGAs are unusable and are hence thrown away. This leads to increase in the production of FPGAs and in turn the cost. In order to reduce the cost of FPGA, faulty one's can be utilized for application specific designs by avoiding the faulty locations instead of throwing.

The available FPGA testing methodologies are categorized into Application Independent Testing (AIT) and Application Dependent Testing (ADT). AIT also called manufacturing test exhaustively tests all the configuration blocks which consumes more time. ADT tests only logic and routing resources used in the particular application. This results in less time consumption.

In this paper ADT technique is used for testing FPGA and BISD is used for detecting and diagnosing the fault location in the FPGA. Addition is taken as application and fault in the mapped location is detected, diagnosed.

II. PREVIOUS WORK

In [1], [2] ADT is used where logic and interconnect resources are tested. For logic diagnosis, a *built-in self diagnosis* (BISD) method is adapted in which the configuration of used logic blocks remains unchanged while the configurations of the interconnect resources and unused logic blocks are modified. Unlike [2] which was limited to single fault diagnosis, [1] is extended to achieve high diagnosis resolution using fault dictionary. [3] provides insight into working of BIST

III. FAULT DIAGNOSIS

Built in self test (BIST) method is used for fault diagnosis of Application specific FPGA. BIST consists of Test Pattern Generator (TPG), Circuit under Test (CUT) and Output Response Analyzer (ORA). TPG generates test vectors for detecting faults in the device. ORA compare the desired output signal with the output of circuit under test giving a unique pass/fail signal as output.

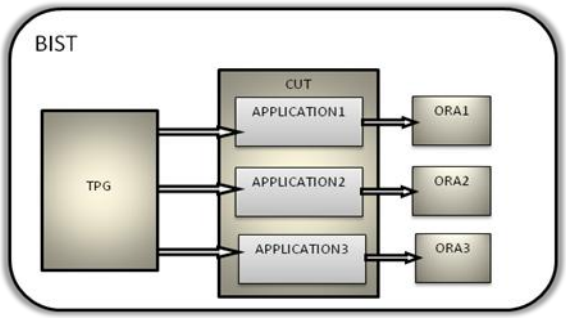

Fig. 1. BIST Block Diagram

In this paper, logic block diagnosis is done. For logic diagnosis, configuration of used logic blocks remains unchanged while the configurations of the interconnect resources and unused logic blocks are modified.

In Fig.2 the original design, with used logic blocks F1 to F9 with original interconnections is shown. In the BIST configuration, the original interconnections are modified such that LFSR outputs, implemented in unused blocks, are connected to the inputs of all used blocks F1-F9 in parallel. The outputs of used blocks along with the parity predictor block are connected to the response compactor, which is also implemented in the available unused resources.

Linear Feedback Shift register (LFSR) acts as TPG that provides all possible test patterns randomly. Parity Predictor provides the desired output to be compared in ORA. Response Compactor is an XOR tree that produces parity of the logic blocks being tested. Since exhaustive test patterns are applied to the inputs of each logic block, the exact faulty resource inside the failing logic block can be uniquely diagnosed based on a fault dictionary.

ISSN: 2278 – 1323

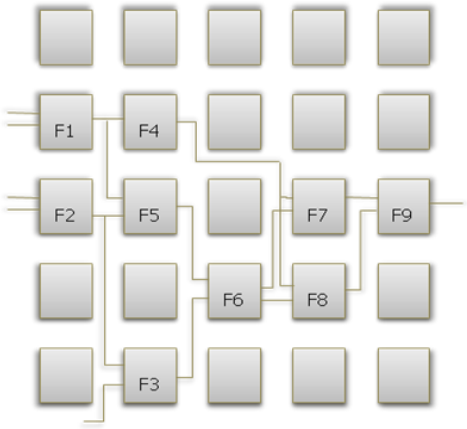

Fig. 2.Original design with used logic blocks F1 to F9 and interconnections

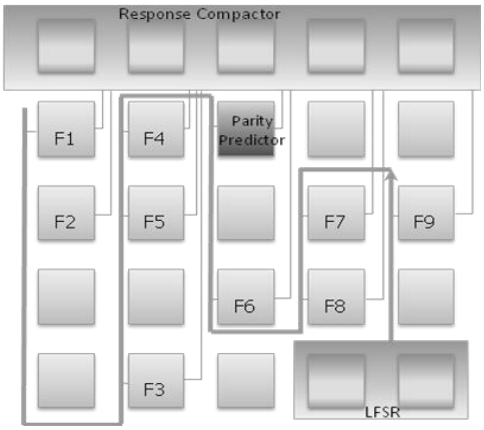

Fig. 3. Modified design for BIST configuration

The compactor circuitry can be represented as a binary *parity matrix* with rows and columns. Each row of the matrix corresponds to a logic block output and each column corresponds to a compactor output. The entry in row and column of the parity matrix is 1 if and only if the compactor output depends on the output of the logic block; otherwise, the entry is 0.

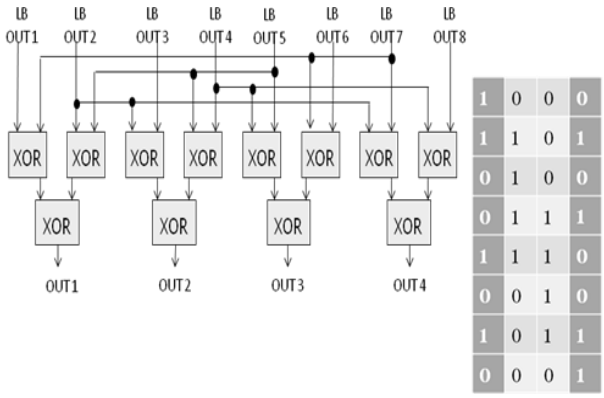

Fig. 4. Parity matrix and corresponding response compactor circuit

Since the user configuration of the logic blocks is known at the test time, the parity bits of the parity matrix can be precomputed and stored in the unused logic blocks. This is implemented by the *parity predictor block.*

The ORA compares the response compactor and parity predictor output to generate unique pass/fail signal and also the location of fault

IV. APPLICATION

16-bit ripple carry adder(RCA) is taken as circuit under test. It requires 16 full adder(FA) .These full adders are considered as logic blocks and tested.

V. SIMULATION RESULTS

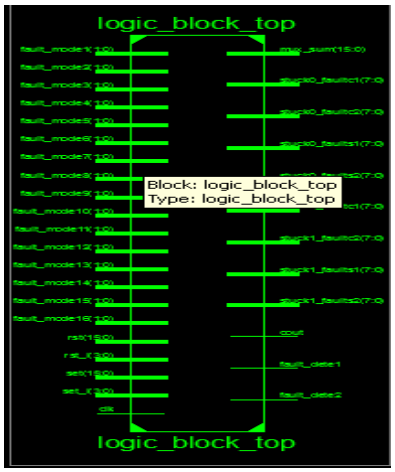

Fig. 5. Schematic view of the project module

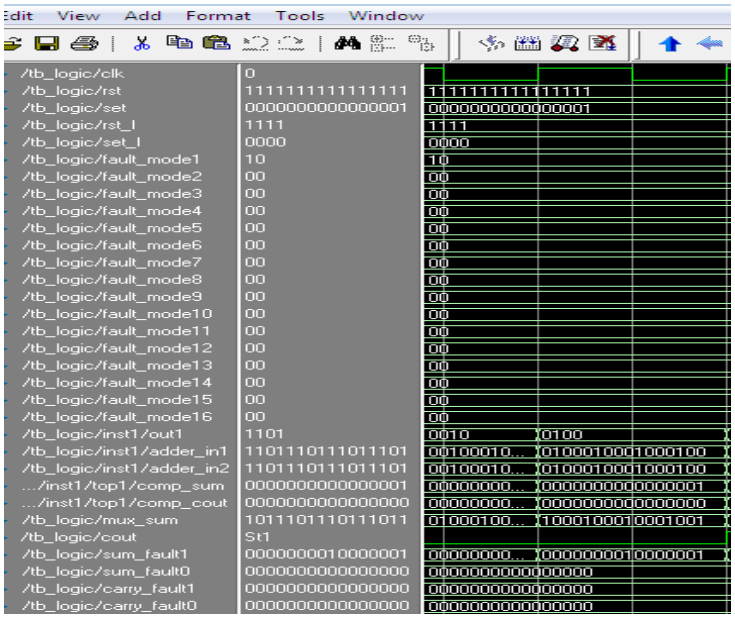

Fig. 6. Results of project module

Fig 6 shows the stuck at 1 fault in first full adder block by looking at first bit of sum output of 16 bit ripple carry adder. Four bit LFSR is used and it is replicated to make it 16 bit input.

The proposed architecture for the Logic block is able to find all the single Struck-At faults as well as multiple faults. Faults are injected randomly on the input and output nodes of the output

ISSN: 2278 – 1323

response analyser. Fault is identified using fault dictionary concept. Total error coverage single faults in the output response analyser has been identified in Logic block 100%. Because of overlapping, total error coverage of multiple fault is 96 %.Therefore high resolution is achieved from the proposed fault diagnosis architecture

TABLE I

ERROR COVERAGE

Faults	Error Coverage
Single SA fault in Logic block	**100%**
Multiple SA fault in the Logic block	**96%**

VI. CONCLUSION

A BIST approach is presented in which multiple faults can be uniquely identified in only one test configuration and later it can be corrected using fault correction methods. The manufacturing yield of FPGA can be increased compared to the traditional scenario in which any defective device was thrown away.

This proposed method can be further improved by automating the design and at run time fault tolerance can be done using partial reconfiguration.

ACKNOWLEDGMENT

The Author thanks the Management and the Principal of The Oxford College of engineering , Bangalore for providing extraordinary lab facilities and encouragement

REFERENCES

[1] M. B. Tahoori, "High Resolution Application Specific Fault Diagnosis of FPGAs," IEEE Trans. Very Large Scale Integr. (VLSI) Circuits, VOL. 19, NO. 10, pp. 1775–1786, October 2011.
[2] M. B. Tahoori, "Application dependent testing of FPGAs," IEEE Trans. Very Large Scale Integr. (VLSI) Circuits, vol. 14, no. 9, pp. 1024–1033, Sep. 2006.
[3] Jamuna. S, "VHDL Implementation of BIST controller" Proc. of Int. Conf. on Advances in Recent Technologies in Communication and Computing 2011.
[4] C. Stroud, "A Designer's Guide to Built-In Self-Test*"*, Kluwer Academic Publishers, Boston MA, 2002.
[5] Dhingra S, Milton D. and Stroud C. E, "BIST for Logic and Memory Resources in FPGAs" *Proc. IEEE North Atlantic Test Workshop*, May 2006.
[6] Edward Stott, Pete Sedcole, Peter Y. K. Cheung "Fault Tolerant Methods For Reliability In FPGAs" in Int.Conference on field prog.logic applications,. Sept 2008, PP415-420.
[7] C.Stroud, S. Konala, C. Ping, and M. Abramovici,"Built-in self-test of logic blocks in FPGAs", in Proc. VLSI Test Symp.,1996,pp.387-392
[8] Ian Kuon, Russell Tessier and Jonathan Rose "FPGA Architecture : Survey and Challanges." Electronic Design Automation, 2008.
[9] M. Abramovici and C. Stroud, "BIST-based detection and diagnosisof multiple faults in FPGAs," in *Proc. Int. Test Conf.*, 2000, pp.785–794.

Kavya V received the Bachelor of Engineering degree in Electronics and Communication Engineering from BMS College of Engineering Bangalore, India, in 2012. She completed MTech from the Department of VLSI Design and Embedded System, at The Oxford College of Engineering, Bangalore,india,in 2014. Her area of interest includes, Digital electronics, Low power VLSI , Communication systems.

ISSN: 2278 – 1323