DESIGN APPROACH FOR FAULT TOLERANCE IN FPGA ARCHITECTURE

Ms. Shweta S. Meshram [1] and Ms. Ujwala A. Belorkar [2]

[1]Department of electronics & telecommunication, Government College of Engineering & Technology, Amravati, Maharashtra.
shwt.meshram@gmail.com
[2] Department of electronics & telecommunication, Hanuman Vyayam Prasarak Mandal's
College of Engineering & Technology, Amravati, Maharashtra.
ujwalabelorkar@rediffmail.com

Abstract

Failures of nano-metric technologies owing to defects and shrinking process tolerances give rise to significant challenges for IC testing. In recent years the application space of reconfigurable devices has grown to include many platforms with a strong need for fault tolerance. While these systems frequently contain hardware redundancy to allow for continued operation in the presence of operational faults, the need to recover faulty hardware and return it to full functionality quickly and efficiently is great. In addition to providing functional density, FPGAs provide a level of fault tolerance generally not found in mask-programmable devices by including the capability to reconfigure around operational faults in the field. Reliability and process variability are serious issues for FPGAs in the future. With advancement in process technology, the feature size is decreasing which leads to higher defect densities, more sophisticated techniques at increased costs are required to avoid defects. If nano-technology fabrication are applied the yield may go down to zero as avoiding defect during fabrication will not be a feasible option Hence, feature architecture have to be defect tolerant. In regular structure like FPGA, redundancy is commonly used for fault tolerance. In this work we present a solution in which configuration bit-stream of FPGA is modified by a hardware controller that is present on the chip itself. The technique uses redundant device for replacing faulty device and increases the yield.

Index Terms - *Fault tolerance, FPGA, hardware controller, redundancy.*

1. INTRODUCTION

The traditional objective during the design of integrated circuit includes timing, area, power and reliability. Designers take this observation into consideration and will redo portion of their design if certain value of this quality measure are not achieved. Most existing systems include measures and assist the designer in meeting the product goals. The situation is different with respect to yield. Yield is still, in most cases, considered an issue which is of concern only to fabrication engineers, and not to chip designers. Most designers are not even aware that some of the design decision which they make, and some of the techniques/tools which they use, have an impact on product yield. In order to make the case for establishing yield as another design objective we must first prove that a chip yield can not only be affected, but consistently improved, by decision made during the decision process.

Like all discrete semiconductor devices, a field programmable gate array can be adversely affected by faults at various stages of component lifetime. While most defects appear immediately following fabrication, occasionally, after extended periods of device use, operational faults can affect in-service programmable components. Common operational faults include open/short metal (4-17% of faults) and transistor stuck-at faults (25-75% of faults) [5] with manifestation rates that vary based on system environmental conditions such as exposure to gamma radiation and extreme temperature. In general, recent trends in FPGA architecture increase the vulnerability of devices to faults.

VLSI technology progress for finer dimension and larger chip area has lead to more complex process and introduction of new and more complex material system [1]. Due to higher defect density and complicated fabrication technique the cost of manufacturing has increased. The increase in defect/fault complexity factor has lead to more devices being effective and hence reduced the yield. Researches for improving yield using various techniques are being carried out since many years [3]. Without using redundancy, the reliability of system is limited by the reliabity of its components. Moreover the system is unprotected from transient errors. Adding fault tolerance to design to improve the dependability of system requires the use of redundancy i.e. One of the ways to achieve higher yield is use of fault tolerance [4]. Incorporating fault tolerance in architecture allows us to use the chip even if a chip is faulty. In this paper we propose a new FPGA like architecture which incorporate fault tolerance in the fabric. Various defects may be produced in a VLSI chip during manufacturing [2]. The existence of defects affects yield and ultimately cost. Field programmable gate arrays (FPGAs) are especially expensive because of the large area penalty taken in favors of reconfigurability. With advancement in process technology, the feature size is decreasing which leads to higher defect densities more sophisticated techniques at increased costs are required to avoid defects [8].

If nano-technology fabrication are applied the yield may go down to zero as avoiding defect during fabrication will not be a feasible option Hence, feature architecture have to be defect tolerant. In regular structure like FPGA, redundancy is commonly used for fault tolerance. In this work we present a solution in which configuration bit-stream of FPGA is modified by a hardware controller that is present on the chip itself. The technique uses redundant device for replacing faulty device and increases the yield. The design is implemented using FPGA Altera Quartus II EP2S158484C3.

2. RELATED WORK

Our research extends previously-reported CAD techniques for overcoming operational FPGA faults. The large majority of previous CAD approaches in this area has focused on recovery from logic cluster (block) faults rather than interconnect faults. In general, these approaches require the functionality of an entire cluster be reimplemented in an unused cluster if a fault is detected.In a fault recovery approach for logic block defects

was described that reserved spare rows and columns of logic blocks to overcome individual block failures [10].

While this approach allowed for recovery with no required on-line device re-route, track width penalties as high as 35% were reported.FPGA arrays were divided into a collection of tiles, each of which could be implemented in one of many pre-compiled layouts [11]. If a logic block fault within a tile occurred, a new tile configuration which left the affected block unused could be substituted.

Recently, an incremental placement approach was described that uses on-line min-max positioning to quickly move faulty logic blocks to unused device blocks [12]. Not only did this technique and most other block movement approaches require incremental re-route following placement, its applicability to coarse-grained cluster-based architectures is limited. Effectively an entire cluster would have to be removed from use even if a fault affected only an isolated LUT of a cluster.

3. PROPOSED TECHNIQUE

In this paper, we are proposing a technique in which a hardware controller on the FPGA takes defect map and configuration file as input & get the modified configuration file as output. We are also using spare device for defect tolerance. If we find any defect in FPGA than the spare device is used and discard the faulty device. The proposed technique is unique because the configuration bit-stream is modified by Hardware not by Software.

The defect map is generated by BIST (built-in-self-test) which is off line Structural type of BIST, present on the chip. Each chip will have different defect map. In some cases fault can also occur after the chip has been manufactured and tested. In our approach the fact that we modify the configuration bit stream in Hardware inside the chip rather than software allow us to handle such faults [2].

4. FAULT AND DEFECT MODEL

Fault is that defect which affects the circuit operation[9]. There are two types of defects present in the wafer, global defect & spot defect. Global defect affect larger area of the chip, on other hand spot defect are completely random in nature and can occur anywhere on the chip and can cause short or open in the wire. In this paper we consider only faults which occur due to spot defect in the FPGA [3].

4.1 HARDWARE CONTROLLER

Figure 1 shows the flow of configuration of a faulty FPGA with the fault tolerant technique.

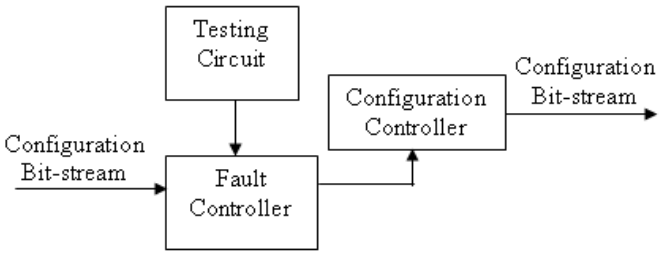
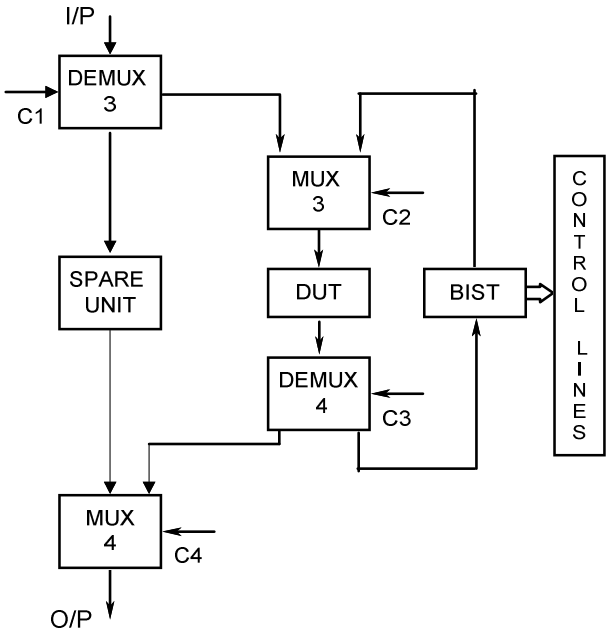


Figure 1 Configuration Flow

Two new hardware blocks are present on the chip, one is the testing circuit which check faulty device in the FPGA and generate fault map. Another is the fault controller that argument configuration bitstream to generate a new bit-stream with the help of fault-map [7].

If a device found faulty, fault controller generate another configuration in which the faulty device in not used and it is shifted to the spare device.



Cl, C2, C3, C4- Control Lines

Fig. 2 Block Diagram of Hardware Controller

Figure 2 show the architectural view of fault tolerance in FPGA architecture, in this we design a device, which is put in Device under Test (DUT) & also kept one spare device, assuming that the spare device is fault free.

4.2 PROPOSED ALGORITHM

Step 1: Send/make control signal so that the DUT of BIST route created and along with disconnect the DUT from practical use and send Busy signal to the external world.

Step 2: Place one defined set of pattern on the DUT to test.

Step 3: Get the output pattern.

Step 4: Compare the output pattern with the expected result.

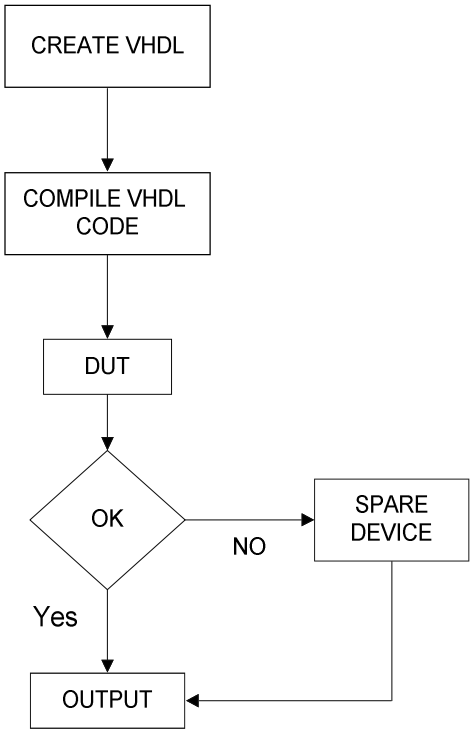


Figure 3 Flowchart

Step 5: If output is OK discarding the remaining step and connect DUT for normal use.

Step 6: If result is faulty, than it state that DUT is faulty.

Step7: Discard DUT and replaced by Spare Unit and get desire output pattern.

5. SIMULATION RESULT

We can design different functional unit present on the chip and simulate the functional unit to get the desire result as follow.

5.1 DECODER

We designed DECODER with input A, B and SEL. If DUT is faulty free then faulty signal became low and give desire output. Following waveform shows the fault free working of decoder with the help of device under test.

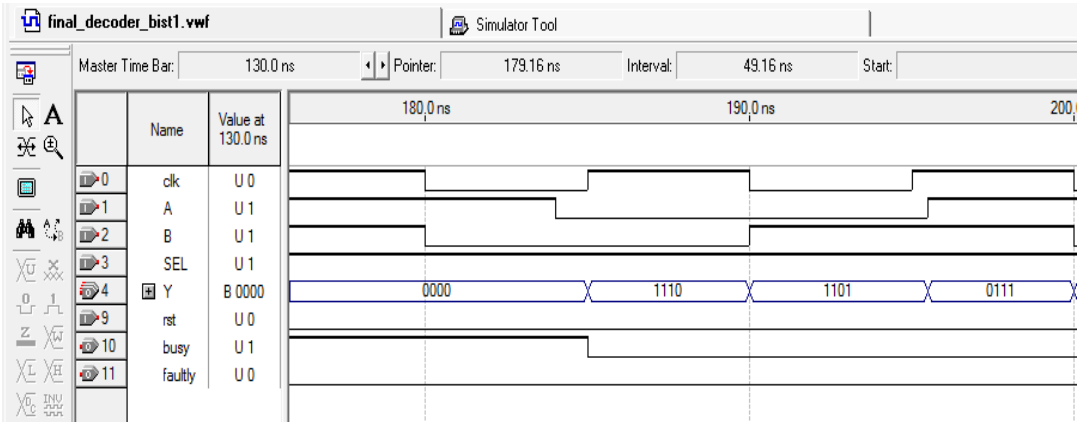


Figure 4 Simulation result of decoder with no Fault

If there is fault in DUT then faulty signal will be high and DUT is replaced by spare unit which is assume to be fault free and give desire output. Following waveform shows the fault free working of decoder with the help of spare device as device under test is faulty.

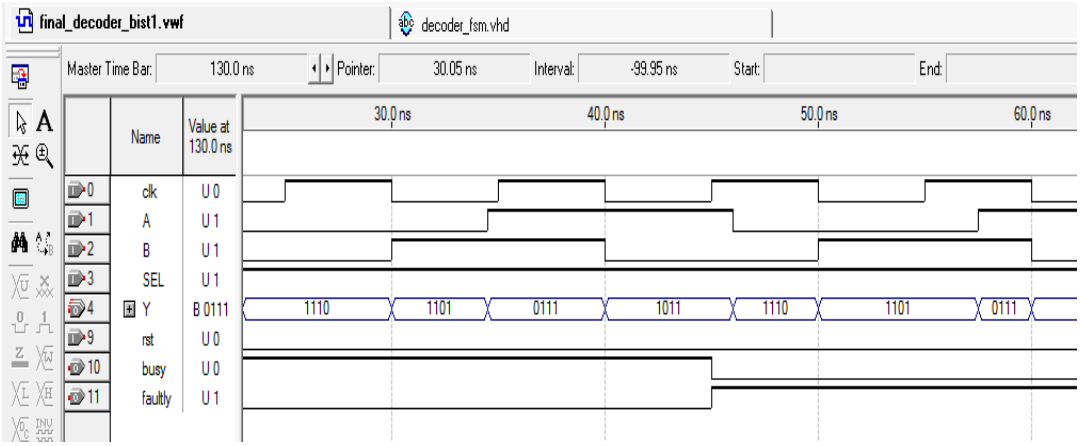


Figure 5 Simulation result with Fault

Table 1: Chip without Hardware Controller

Device Under Test	Logic Element	Power Dissipation	Time delay
Decoder	16	25 mW	0
Mux	14	20 mW	0
Adder	27	29 mW	0

Table 1: Chip with Hardware Controller

Device Under Test	Logic Element	Power Dissipation	Time delay
Decoder	24	30 mW	0.12ns
Mux	24	25 mW	0.21ns
Adder	37	34 mW	0.14ns

5.2 DISCUSSIONS ON THE PROPOSED TECHNIQUE

The cost of incomplete or inadequate testing can be longer time to market, lower client satisfaction, higher design cost, and a potentially shorter product life cycle. The difficulty and cost of adding test headers and connectors can result in incomplete testing of new boards, which can have a tremendously negative impact when a bug or problem is identified.

With advance in process technology, the feature size is decreasing which leads to higher defect densities. We present novel and efficient methods for built-in-self-test (BIST) of FPGAs for detection and diagnosis of permanent faults in current as well as emerging technologies that are expected to have high fault densities. We establish the correctness of the deterministic phases of our BIST technique.

Simulation results show that our BIST technique has very high fault coverage and low fault latency, and supports the theoretical analysis. The proposed method affects three aspect yield, area of chip & delay. By this approach we get the maximum (100%) yield whereas area is concern, as we are using spare device area will increase and due to increase in area the affect time delay of various nets.

6. CONCLUSION

We develop a novel on-line built-in self-test (BIST) technique for testing FPGAs that has a very high effectively even in presence of faults. This approach presents a hardware controller based mechanism with the help of which we can use the device even it is found faulty. The method can be used by FPGA vendor to increase yield & thus decrease the cost. In this propose work we have given detailed designed algorithm for the fault controller for assumed architectures. This work enables us to look in a different paradigm of circuit design where faults in a chip are handled by the circuit inside the chip itself using redundancy.

REFERENCES

[1] *H. Sewell, D. McCafferty, L. Markoya, E. Hendrickx, J. Hermansand, and K. Hermans. 32nm node technology development using interference immersion lithography. Proceedings of the SPIE, 5753:491–501, 2005.*

[2] *N. Campregher, P. Cheung, G. Constantinides, and M. Vasilko. Analysis of yield loss due to random photolithographic defects in the interconnect structure of fpgas. In FPGA, 2005.*

[3] *Koren and Z. Koren. Defect tolerance in VLSI circuits: Techniques and yield analysis. Proceedings of the IEEE, 86(9), Sept 1998.*

[4] *F. Hatori, T. Sakurai, K. Nogami, K. Sawada, M. Takahashi, M. Ichida, M. Uchida, I. Yochii, Y. Kawahara, T. Hibi, Y. Saeki, H. Muroga, A. Tanaka, and K. Kanzanki. Introducing redundancy in field programmable gate arrays. In Custom integrated Circuis Conference, 1993.*

[5] D. P. Siewiorek and R. Swarz. *Reliable Computer Systems:Design and Evaluation*. Digital Press, Burlington,Ma, 1992.

[6] *A. J. Yu and G. G. F. Lemieux. Defect tolerant fpga switch block and connection block with fine grain redundancy for yield enhancement. In IEEE International Conference on Field Programmable Logic and Applications, 2005.*

[7] *S. Mitra, P. Sirvani, and E. McCluskey. Fault location in fpga based reconfigurable systems. In IEEE Intl High Level Design Validation and Test Workshop, 1998.*

[8] *W. J. Huang and E. McCluskey. Column based pre-compiled configuration technique for fpga fault tolerance. In FPGA, 2001*

[9] *N.Z. Haron, M.M. Ibrahim, S.A.M. Junos and Aziz. FPGA Implementation of microcode-based and FSM- based Memory built in Self Test Controllers*

[10] *F. Hanchek and S. Dutt. Methodologoies for Tolerating Cell and Interconnect Faults in FPGAs. IEEE Transactions on Computers, 47(1):15–32, Jan. 1998.*

[11] *N. Shnidman, W. H. Mangione-Smith, and M. Potkonjak. On-Line Fault Detection for Bus-Based Field Programmable Gate Arrays. IEEE Transactions on Very Large Scale Integration, 6(4):656–665, Dec. 1998.*

[12] *J. M. Emmert and D. Bhatia. Reconfiguring FPGA Mapped Designs with Applications to Fault Tolerance and Reconfigurable Computing. In Field Programmable Logic Workshop (FPL'97), Oxford, England, Sept. 1997.*

SHORT BIOGRAPHY

Ms. Shweta S. Meshram was born in Amravati, Maharashtra in 1988. She received the B.E. Degree in Electronics and Telecommunication from Sant Gadge Baba Amravati University, Amravati in 2005 & pursuing M.E. Degree in Electronics System and Communication Engineering. Her interests of research are in Micro Electronic System Design using VLSI Technology & VHDL based FPGA design.

Ms. Ujwala A. Belorkar was born in Amravati, Maharashtra in 1970. She received the M.E. Degree in Digital Electronics from S.G.B. Amravati University, Amravati in 2004 & pursuing Ph.D. Degree in Electronics Engineering. Currently she is working as a Associate Professor & Head in Electronics & Telecommunication Department at H.V.P.M's College of Engineering & Technology, Amravati. Also she is working as a visiting faculty for M.Tech. in Electronics System and Communication Engineering at Govt. College of Engineering Amravati. Her interests are in Micro Electronic System Design using VLSI Technology.