

Simulated High Speed Fault Injection in Arithmetic Unit using Verilog HDL

M.B Aswini¹, R. Chitra², S. Gopika Devi³, S. Vijay Murugan⁴

U.G. Scholars, Department of ECE, Adhiyamaan College of Engineering Dr. MGR Nagar, Hosur, Tamilnadu, India^{1,2,3}

Assistant Professor, Department of ECE, Adhiyamaan College of Engineering, Hosur, Tamilnadu, India⁴

ABSTRACT: In recent years, there has been a rapid increase in the use of computer-based systems in areas where failures can cost lives and money, such failure occurs in railway traffic control, aircraft flight, telecommunications, and others. This trend has led to a growing interest in the techniques for the validation of the fault tolerance properties of these systems and for the evaluation of their reliability, thus leading to a VLSI technology where fault injection has become a popular technique for experimentally verifying the fault tolerant based designs. In this proposed work a high speed fault injection tool for testing has been developed to build a real time fault injection mechanism with good performance in all the aspects. This paper presents a fault injection Micro Wind Tool (FIMTO) that supports synthesizable fault models for dependability analysis of digital systems. As a case study, design of AU is done and the fault is injected into it.

KEYWORDS: VLSI technology, Fault injection, Arithmetic Unit (AU), Fault injection Micro wind tool.

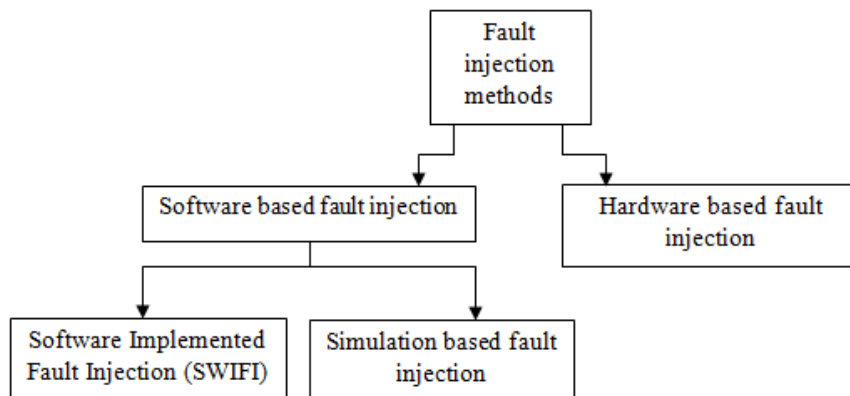
I. INTRODUCTION

In VLSI chip manufacturing technology, a manufacturing defect is defined as a finite chip area with electrically malfunctioning circuitry caused by the errors in the fabrication process which results as the faults. A chip with no manufacturing defect is called a good chip. In IC manufacturing, the faults are unavoidable at times and the number of faults is increasing when we move towards the advanced nano meter level technologies. The following definitions give the basic terminology associated with faults in VLSI testing. A defect in an electronic system is defined as the unintended difference between the implemented hardware and its intended design. An error is defined as the wrong output signal produced by the defective system. An error may be caused due to a defect. A fault is defined as the representation of a defect at the abstracted function level. To test the presence of a fault in a manufactured IC, one can give different input values and see the response on the output pins.

The fault injection is a technique of Fault Tolerant Systems (FTSs) for validation which is being increasingly consolidated and applied in a wide range of fields and applications, and several automatic tools have been designed and implemented. Fault injection is the technique of the Dependability of Fault Tolerant Systems in the accomplishment of controlled experiments where the observation of the system's behaviour in presence of faults which is induced explicitly by the voluntary introduction (injection) to the system.

The fault injection of a system can be implemented within three main techniques, Physical Fault Injection (PFI) this type is accomplished at the physical level, by disturbing the hardware with parameters of the environment such as heavy ions radiation, electromagnetic interference, etc., or modifying the value of the pins of the integrated circuits. Second one Software Implemented Fault injection (SWIFI) this type of technique, also called as Fault Emulation, which consists of reproducing information level errors that would have been produced upon occurring faults in the hardware. This is based on different practical types of injection, such as the modification

of the memory data, or the mutation of the application software or at the lowest service layers for example at operating system level. Final techniques is simulated fault injection in this technique, the system under test is simulated in other computer system. The faults are induced by altering the logical values during the simulation. This work is framed in the simulated fault injection as shown in the Fig 1) a), based on the VHDL hardware description language.



a) Block diagram of different types of fault injection

Fig 1) a) Block diagram of different types of fault injection

II. RELATED WORK

In this [3] presented a method using the Field Programmable Gate Array (FPGA)-based emulation system for fault grading technology. In this the real-time simulation capability would significantly improve the performance of fault grading; this is one of the most time-consuming tasks in the circuit design and test process. In this, the fault simulation for large sequential circuits is a very time-consuming task. There by with the increasing performance of logic emulation technology, a hardware fault emulation system has become only feasible but not very efficient.

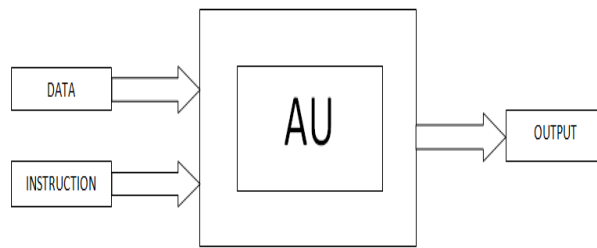
In widespread adoption of VLSI technology on the devices for safety critical application tasks with effective tools for the evaluation and validation of their system reliability. With the continuous increase in the integration level of electronic systems it is making increasingly difficult to guarantee an acceptable degree of reliability, because of the occurrence of transient faults in the electronic systems. To face the above issue, specific mechanisms are required to increase the robustness of electronic devices and systems with respect to possible errors which occurs during their normal functions [4].

The software faults are the major cause of the computational defects in the system. Even though when these systems are tested exhaustively they present some failures due to the activation of residual software faults in the source code. In this, the Software fault injection tools are useful to emulate the presence of software faults and to monitor that the system is allowing one to observe that the system continues to operate as the expected one. This work presents the J-SWFIT tool, will allows us to compare the systems behaviour in the presence and absence of each fault in the system [5].

III. PROPOSED WORK

The good perspective of modelling systems and faults with Hardware Description Language (HDL), has been consolidated as a powerful standard to analyse and design the computer systems. To do that, we performed an injection campaign by means of a fault injection tool deployed for the above purpose. Using one of the most efficient tools for the use of fault injection tool, we have performed an injection campaign on the system and

have calculated the coverage and latencies on the detection and recovery of the produced errors. Block diagram of arithmetic unit as shown in the Fig 2) a).



a) Block diagram of Arithmetic Unit
Fig 2) a) Block diagram of Arithmetic Unit

Fault injection process can be done by applying some extra gates and wires to the original design description and modifying the target Verilog model of the system. One of these extra wires is the Fault Injection System (FIS) which plays the key role in the fault injection experiments.

If a FIS takes the value 1, fault would be activated and if it takes the value 0, the fault would become inactive. Hence the tool called Micro Wind enables the layout of the whole arithmetic unit after programming and simulating. We have chosen this technique fundamentally due to the growing interest of the simulated injection techniques as a complement of the physical fault injection and Fault Emulation (SWIFI) experiments. The greatest advantage of this method over the previous ones is the observability and controllability of all the modelled components of the system.

The simulation can be accomplished in different abstraction levels. Another positive aspect of this technique is that, the possibility of carrying out the validation of the system during the design phase, before we have the final product.

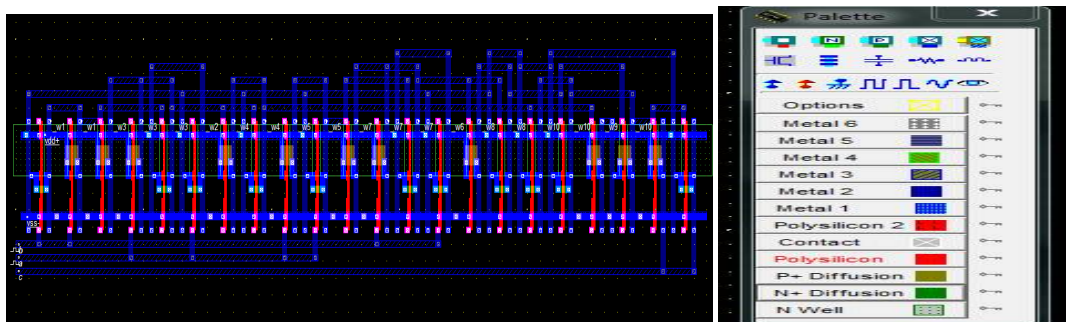
We have seen the growing interest of the simulated injection techniques as a complement of the physical fault and Fault Emulation this proposed design of arithmetic unit also includes the combination of transistor and structural level.

The faults that is to be included in the layout are

- Stuck at -0 fault
- Stuck at -1 fault

Required system under the test is simulated in Fault Injection Micro wind Tool (FIMTO).

IV. EXPERIMENTAL RESULTS



a) AU layout
b) Fig 3) a) Layout of Arithmetic Unit

The rest of the layouts for each faults are being designed and implemented. The above mentioned Fig 3) a) is the obtained layout for Arithmetic Unit and the types of faults and its layout are expected to be obtained.

V. CONCLUSION

This paper described the FPGA-based fault injection tool, called, FIMTO for evaluating the digital systems modelled by Verilog HDL. By exploiting a prototypical version of the described Fault Injection environment, we were able to evaluate its effectiveness with respect to state-of-threat alternative approaches. Unlike co-simulation using ESL methodologies, RTL co-simulation can provide very detailed run-time execution results so that we may evaluate the failure rate or reliability of the target system with precision. Furthermore, unlike other simulated fault injection techniques, such as the saboteur technique or the mutant technique, the proposed Micro Wind simulation.

This proposed work will also be a metric for measuring the robustness of a system in many ways, and as said earlier this method has full controllability and observes ability. Hence, the process can be fully automated, and is applicable with no limitation to any HDL design. One more major advantage that this model would have is that it will support several synthesizable fault models.

REFERENCES

1. Design and implementation of high speed fault injection process for testing FPGA based designs International Journal of Science, Engineering and Technology Research (IJSETR), Volume 3, Issue 10, October 2014
2. Fault Injection into VHDL Models: Experimental Validation of a Fault Tolerant Microcomputer System D. Gil, R. Martínez, J. C.Baraza1, J. V. Busquets, P. J. Gil.
3. K.-T. Cheng, S.-Y. Huang, and W.-J. Dai, "Fault emulation: A new methodology for fault grading," Trans. on the IEEE Computer-Aided Design of Integrated Circuits and Systems, Oct. 2005
4. P. Civera, L. Macchiarulo, M. Rebadengo, M. S. Reorda, and M. A. Violante, "Exploiting FPGA for accelerating fault injection experiments," Proc. Of the International On-Line Testing Workshop, 2001, pp. 9-13.
5. Bruno Pacheco sanches and Regina Moraes, "JSWFIF: A Java Software Fault Injection Tool" 2011
6. FPGA Implementation of FITO Based Fault Simulation and Systems- Elsevier, ISSN 2319-8885 Vol.03, Issue.47 December-2014, Pages:9484-9487
7. Simulated Fault Injection Using Simulator Modification Technique - Jongwhoa Na and Dongwoo Lee, ETRI Journal, Volume 33, Number 1, February 2011
8. Ref.: M. J. Howes and D. V. Morgan, Reliability and Degradation - Semiconductor Devices and Circuits, Wiley, 1981
9. Course slides from "SoC testing and verification" prof Virendra singh, IISc, Bangalore
10. H. R. Zarandi, G. Miremadi, and A. R. Ejlali, "Fault Injection into Verilog Models for Dependability Evaluation of Digital Systems," Proc. of the International Symposium on Parallel and Distributed Computing, Oct. 2003, pp. 281-287.
11. J. Bou, P. Petition, and Y. Crouzet, "MEFISTO-L: A VHDL-based fault injection tool for the experimental assessment of fault tolerant," Proc. of the International Symposium on Fault-Tolerant Computing, Jun. 1998, pp. 168-173.