New SEU Modeling by Architecture Analysis

Jan Pospisil, Jan Schmidt, Petr Fiser
Department of Digital Design
Faculty of Information Technology, Czech Technical University in Prague
Prague, Czech Republic
Email: jan.pospisil@fit.cvut.cz, jan.schmidt@fit.cvut.cz, petr.fiser@fit.cvut.cz

I. Introduction

The majority of recent FPGAs use SRAM memory for both configuration and data storage. These memory cells and other structures can be affected by ionizing radiation (SEE – Single Event Effects) causing change of the stored value (SEU – Single Event Upset). SEU can change the logical function as well as the structure of an implemented design, which has an impact on the design dependability.

There are several design techniques to build a dependable application using unreliable hardware. To validate an application, we must predict its dependability parameters, and furthermore verify that no fault can lead to dangerous behavior. The design also must balance dependability and cost (silicon area, power consumption and performance).

There are two main ways to estimate dependability parameters of a design—the Accelerated Life Test (ALT) [1], which can give very accurate results, but is very expensive and not easily accessible, and a simulation model for which the detailed physical structure of the tested hardware is needed, which is a problem with FPGAs.

In this paper, we propose to obtain a model of the implemented design useful for fault analysis. The model results from duplicating the back-end (physical design) tool of the standard flow, in this case by an open tool. It works on a global FPGA model, which can mimic existing devices and which can be annotated by information necessary for the fault analysis.

II. Previous Work

Several works on SEU simulation or modeling in CMOS ASIC have been published [2], [3], [4]. Also SEU simulation in FPGA is mentioned in several papers, for example in [5], where hardware–software co-simulation was done with SLACC-1V computer acceleration card [6]. This has been further extended by TMR design in [7] and compared with real radiation test in [8]. Another method of design simulation and device testing focused on configuration memory have been published in [9] and [10]. Different approach using an HW emulator based on Atmel FPSLICK has been published in [11] and [12].

The Verilog-to-Routing project (VTR) [13] is an open source academic CAD suite intended to help with both proposing new architectures of programmable circuits and developing EDA tools for such chips. This tool allows a user to create a custom FPGA architecture and perform timing-driven packing, placement and routing on it. There are extensions of VPR with the intention to model real architectures more precisely [14], [15].

A method to classify fault behavior by an arbitrary predicate under any combinational fault model has been published [16]. Using this technique, we will be able to provide a calibrated model for our method and more realistic estimates of dependability and reliability of any design in question.

III. Proposed method

Given a design and a target FPGA device, the purpose of our work is to predict the behavior of the implemented design under a radiation exposure using a model of the implemented design. To construct such a model, a suitable detailed description of the target FPGA architecture is necessary, together with a physical design tool working on that architecture. The constructed model is then calibrated by ALT. The calibration shall cover differences between the real and the modeled architecture as well as differences in the physical design tools. Once calibrated model thus covers all designs on the target architecture.

A. The FPGA model

Three areas of the FPGA architecture must be covered by the architecture model: data storage (RAMs and flip-flops), configurable logic blocks and interconnection. The design model of data storage commonly corresponds to the real implementation, and therefore can easily be derived. In recent devices, configurable logic blocks are composed of LUTs and multiplexers. Their structure is often published. The difficult part is therefore the interconnection.

The interconnection architecture has fairly stabilized in recent FPGAs. It has usually two levels of hierarchy, with local interconnection between adjacent logic blocks and global interconnection over the entire chip. There is also tendency to simplify the interconnection structures—it is simpler to use them in CAD tools and simpler to model. Incidentally, their fault models are also easier to construct and to simulate. The long chains of passive switches known from past devices, disappeared. Also, the selection of inputs for logic blocks is done nearly uniformly.

The modeling process is based on the VTR suite [13]. We chose this because no modern FPGA architecture is known to details and no commercial tools are open enough to allow us to do the intended modifications. VTR takes a high-level architecture model of the FPGA, which is in all cases transformed to a low-level graph model, where each node is a signal source, destination or a configurable connection point. With the knowledge of common solutions, we can estimate possible implementation of each node and its fault models, and annotate the graph with this information.

B. Usage of the model

For each design, the (commercial) tools for the target architecture are run to obtain a design implementation. The model of the implementation for fault analysis is constructed by VPR. The more information we carry from the original implementation, the more accurate the fault analysis model will be.

Given an annotated implementation of a net, alterations of configuration memory content in elements constituting the signal path can be transformed into fault models analyzable by standard methods. Stuck–at faults (constant 0 or 1 at a signal) are easy to analyze. As interconnection multiplexers are commonly implemented by passive switches, stuck–open faults and shorts can occur. These faults were studied in connection with CMOS circuits [17]. SEU in a LUT configuration memory are transformed into changes in logic function performed by that LUT.

Before the resulting annotated netlist of the design can be used for fault analysis, it should be simplified. As many elements of the programmable interconnect are of similar nature, there is a strong chance that their faults will dominate each other, leading to simpler models. The final fault analysis can be done using Monte Carlo fault simulation, Satisfiability–based methods [16] and other methods.

IV. Future Work

Radiation tests needed for calibration of our model are now being prepared in collaboration with other researchers at our university as well as with Nuclear Physics Institute in Řež (NPI). After some data will be acquired from ALT testing, the VPR model will be calibrated and improved.

The problem of radiation tolerant programmable systems is nowadays topical in the ALICE ITS project in CERN [18]. Here data from thousands of pixel detectors need to be processed and an old platform of specialized ASIC chips is planned to be replaced by reconfigurable hardware. Yet such hardware is vulnerable to high-energy particles, which are very frequent near the detector. Therefore, the design must be fault-tolerant and must sustain a very high SEU rate. This problem is being solved in NPI too. Our models and testing designs should be useful there.

V. Conclusions

Recently, FPGA devices are more frequently used in applications demanding dependability and safety. These FPGAs are, nevertheless, manufactured using CMOS technology, often with SRAM memory cells, which are prone to ionizing radiation. In our work, we proposed a method of modeling the behavior of an FPGA in radiation–harsh environment, based on parameters obtained from experiments with the real hardware. The proposed method utilizes academic toolchain VPR. By modifications of this toolchain, and from SEU characteristics gathered from "in vivo" experiments, a modeling and simulation platform for future designs can be constructed, with close-to-reality results.

Acknowledgements

Research described in the paper is supervised by Ing. Jan Schmidt, Ph.D., FIT CTU in Prague and supported by the CTU SGS grant No. SGS13/101/OHK3/1T/18.

References

[1] W. Meeker and L. Escobar, "Pitfalls of accelerated testing," *Reliability, IEEE Transactions on*, vol. 47, no. 2, pp. 114–118, 1998.

[2] B. Bhuva *et al.*, "Switch–level simulation of total dose effects on CMOS VLSI circuits," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 8, no. 9, pp. 933–938, 1989.

[3] N. Kaul *et al.*, "Simulation of SEU transients in CMOS ICs," *Nuclear Science, IEEE Transactions on*, vol. 38, no. 6, pp. 1514–1520, 1991.

[4] M. P. Baze *et al.*, "An SEU analysis approach for error propagation in digital VLSI CMOS ASICs," *Nuclear Science, IEEE Transactions on*, vol. 42, no. 6, pp. 1863–1869, 1995.

[5] E. Johnson *et al.*, "Single–event upset simulation on an FPGA," in *Proceedings of the International Conference on Engineering of Reconfigurable Systems and Algorithms (ERSA)*. CSREA Press, 2002, pp. 68–73.

[6] USC–ISI East., "SLAAC-1V User VHDL," 2013, http://slaac.east.isi.edu/.

[7] N. Rollins *et al.*, "Evaluating TMR techniques in the presence of single event upsets," in *International Conference, Washington DC*, vol. 11, 2003.

[8] E. Johnson *et al.*, "Accelerator validation of an FPGA SEU simulator," *Nuclear Science, IEEE Transactions on*, vol. 50, no. 6, pp. 2147–2157, 2003.

[9] M. Ceschia *et al.*, "Identification and classification of single–event upsets in the configuration memory of SRAM–based FPGAs," *Nuclear Science, IEEE Transactions on*, vol. 50, no. 6, pp. 2088–2094, 2003.

[10] M. Violante *et al.*, "Simulation–based analysis of SEU effects in SRAM–based FPGAs," *Nuclear Science, IEEE Transactions on*, vol. 51, no. 6, pp. 3354–3359, 2004.

[11] P. Kubalik *et al.*, "Fault Injection and Simulation for Fault Tolerant Reconfigurable Duplex System," in *Design and Diagnostics of Electronic Circuits and Systems, 2007. DDECS '07. IEEE*, 2007, pp. 1–4.

[12] J. Kvasnicka *et al.*, "Experimental SEU Impact on Digital Design Implemented in FPGAs," in *Digital System Design Architectures, Methods and Tools, 2008. DSD '08. 11th EUROMICRO Conference on*, 2008, pp. 100–103.

[13] J. Rose *et al.*, "The VTR project: architecture and CAD for FPGAs from verilog to routing," in *Proceedings of the ACM/SIGDA international symposium on Field Programmable Gate Arrays*, ser. FPGA '12. New York, NY, USA: ACM, 2012, pp. 77–86.

[14] M. J. Beauchamp *et al.*, "Embedded floating-point units in FPGAs," in *Proceedings of the 2006 ACM/SIGDA 14th international symposium on Field programmable gate arrays*, ser. FPGA '06. New York, NY, USA: ACM, 2006, pp. 12–20.

[15] C. W. Yu, "A Tool for Exploring Hybrid FPGAs," in *Field Programmable Logic and Applications, 2007. FPL 2007. International Conference on*, 2007, pp. 509–510.

[16] J. Schmidt *et al.*, "The Influence of Implementation Technology on Dependability Parameters," in *Digital System Design (DSD), 2012 15th Euromicro Conference on*, 2012, pp. 368–373.

[17] A. Jayasumana *et al.*, "Design of CMOS circuits for stuck-open fault testability," *Solid-State Circuits, IEEE Journal of*, vol. 26, no. 1, pp. 58–61, 1991.

[18] S. Rossegger, "The ALICE ITS upgrade," in *Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC), 2011 IEEE*, 2011, pp. 513–517.