

# Techniques to improve the reliability of fault-tolerant systems based on self-reconfigurable FPGAs

André Roberto Guerra  
UNIGUAÇU - São Miguel do  
Iguaçu – Paraná - BRASIL  
UNIMEO – Assis Chateaubriand  
– Paraná – BRASIL  
[arguerra@certto.com.br](mailto:arguerra@certto.com.br)

J. M. Martins Ferreira  
FEUP / DEEC – Rua Dr.  
Roberto Frias 4200-465  
Porto - PORTUGAL  
[jmf@fe.up.pt](mailto:jmf@fe.up.pt)

Manuel G. Gericota  
ISEP – Rua Dr. Antonio  
Bernardino de Almeida  
4200-072 Porto –  
PORTUGAL  
[mgg@dee.isep.ipp.pt](mailto:mgg@dee.isep.ipp.pt)

## Abstract

*In this paper it is proposed a new technique to improve the reliability of fault-tolerant systems based on self-reconfigurable FPGAs. The aim is to create a self-tolerant system based on self-reconfiguration. To achieve this objective the work was divided in five main tasks: the analysis of fault inducement mechanisms in FPGAs, its correlation and its matching with existent fault models, or, eventually, if necessary, the proposal of a new model; the design and evaluation of a fault tolerance mechanism for FPGAs; the design and implementation of a methodology able to detect, diagnose and repair the emerging faults; the development and validation of the proposed methodology. This study will be the base for a PhD thesis.*

## 1. Introduction

SRAM-based Field Programmable Gate Arrays (FPGAs) endured a considerable evolution in the last few years, not only in terms of density and performance, but also due to the addition of new features, expanding the areas where these devices can advantageously replace ASICs. New families of FPGAs, considerably less expensive, became a serious alternative, even in the design of critical systems. In this particular case, the design of fault tolerant circuits is required as it is necessary to assure high levels of reliability and availability.

This goal demands online concurrent detection of permanent and transient faults, which should be masked to avoid propagation, while triggering a test procedure to determine their origin, either functional or structural, and to assure the repair of their cause(s), avoiding cumulative effects that may lead to a general system failure.

Therefore, it becomes imperative to study the specific fault inducement mechanisms of these devices and to develop innovative test

methodologies tailored to their unique architecture and to the new applications that are now possible.

Such methodologies have to guarantee both fault tolerance and repair, by detecting faulty resources, and to avoid their use when new incoming functions required by the applications sharing the FPGA are implemented.

Dynamic reconfiguration and the incorporation of self-reconfiguration capabilities in recent FPGAs, allied to the use of hard- or soft- core embedded microcontrollers, enable the development of self-contained fault-tolerant reconfigurable systems [1].

The on-chip controller will be responsible for fault detection and diagnose and for the implementation of repair measures, including all necessary rerouting and floorplanning operations, in a transparent and autonomous way.

## 2. Classes of faults and fault modelling in FPGAs

To be able to create a successful methodology to detect and repair emerging faults in an FPGA and to correlate them with a representative fault model it is necessary to study and characterize their specific fault inducement mechanisms. All defects and events that may affect the operation of the FPGA must be identified. Each of them has to be translated into one or more logic faults to produce a correlation list. This step will enable an evaluation of the accuracy of current fault models when applied to the specific architecture and characteristics of self-reconfigurable SRAM-based FPGAs. If necessary, a new fault model will be proposed to describe behaviours not previously defined by the more common fault models. This correlation between defects/events and their representative faults will setup the basis for the development of a specific fault tolerance methodology. Moreover, it will also supply valuable information to help diagnose their origin and undertake repair measures.

### 3. Evaluation of different fault tolerance methodologies

Different fault tolerance methodologies were proposed in the literature, optimized to different supports and functionalities. As so, a comparative study of the most adequate fault tolerance methodologies to be used with different kinds of functions implemented in FPGAs have to be done to access which of them should be considered. Several factors will be taken into account to help to decide which method is the most suitable to each particular situation, such as the reliability index required by the function or by the application, the adequacy in relation to the functional specification, its speed requirements and the type of resources it uses. While space redundancy can almost be used with any function, the use of time redundancy depends on functional and operational aspects [2]. Another problem to be analysed is that redundant systems are subject to common-mode failures (CMF). In FPGA implementations, CMFs may be avoided by using module synthesis diversity [3]. However, different implementations of a same module imply different propagation delays that may cause metastability in the voters of a TMR and lead to wrong output values. The overhead of each solution, comprising not only the resources occupied at a given moment by a function, but also other aspects such as power consumption and variations on the maximum speed of operation (performance), is another aspect that must be considered. Since performance depends on floorplanning, the introduction of redundancy will have consequences at this level as well.

### 4. Detection-diagnosis-repair

The definition of a fault detection procedure is particularly difficult in fault tolerant implementations, since they are designed precisely to mask such occurrences. However, the presence of a fault will leave part of the implementation out-of-work, and, if subsequent faults occur, the fault tolerance methodology may be unable to guarantee the correct system operation, unless a recovery process is activated to remove the defective resources. Therefore, to avoid an accumulation of faults, faulty modules must be detected and repaired. Fault detection implies the development and implementation of a mechanism to detect faulty modules in spatial and temporal redundancy implementations without disturbing their operation.

Prior to the repair phase it is necessary to perform some kind of diagnosis, as the origin of the fault may be transient, in which case no repair intervention is needed, or permanent. Moreover, if the fault is permanent, its origin may be logical or physical. As in FPGAs device functionality is

defined by the configuration memory, a bit flip in a configuration cell will lead to a permanent functional fault. However, it does not occur due to any defect on the structure of the FPGA and can be logically corrected by a simple partial reconfiguration of the configuration memory. If no configuration change is detected, then a physical defect is most probably the cause of the fault. A physical defect implies the relocation of the affected module and the structural test of the released resources [4]. Procedures will be defined to deal with the different sources of faults and to restore the correct operation of the faulty module.

The validation of the methodology will be done using a prototype circuit implemented on a Virtex device with self- and dynamic reconfiguration [5].

### 5. Conclusions

The increasing use of SRAM-based FPGAs, even in critical systems, makes this study very important as fault tolerance mechanisms play a very important role in all systems demanding high flexibility and high reliability (e.g. aero-space industries, health, financial institutions). Additionally, reconfiguration provides the possibility to maintain the index reliability level by repairing the faulty modules without the necessity of resource substitution, thus providing greater flexibility and autonomy.

### References

- [1] B. J. Blodget, S. P. McMillan, P. Lysaght, "A lightweight approach for embedded reconfiguration of FPGAs", *Proc. DATE Designers' Forum*, pp. 399-400, 2003.
- [2] Lala, P. K., *Self-Checking and Fault-Tolerant Digital Design*, Morgan Kaufman Publishers, San Francisco CA, 2001. ISBN 0-12-434370-8.
- [3] N. R. Saxena, S. Fernandez-Gomez, Wei-Je Huang, S. Mitra, Shu-Yi Yu, E. J. McCluskey, "Dependable Computing and Online Testing in Adaptive and Configurable Systems", *IEEE Design and Test of Computers*, Vol. 17, No. 1, pp. 29-41, January-March 2000.
- [4] Manuel G. Gericota, Gustavo R. Alves, Miguel L. Silva, José M. Ferreira, "On-line Testing of FPGA Logic Blocks Using Active Replication", *Proc. of the Norsk Informatikkonferanse*, Kongsberg, Norway, November 2002, pp. 167-178.
- [5] *Virtex®-II Pro Platform FPGA Handbook*, Xilinx, Inc., 2002.