RELIABILITY LIMITS OF TMR IMPLEMENTED IN A SRAM-BASED FPGA: HEAVY ION MEASURES VS. FAULT INJECTION PREDICTIONS

G. Foucard, P. Peronnard, R. Velazco
Laboratoire TIMA
Grenoble, France

***Abstract*—This paper presents experimental results putting in evidence the weaknesses of TMR strategy implemented in SRAM-based FPGAs. Results obtained from radiation ground testing are confronted to fault injection campaigns.**

Keywords: TMR, FPGA, SRAM, fault injections, heavy ions

I. INTRODUCTION

Field Programmable Gate Arrays (FPGAs) are much appreciated by designers because they offer low cost, high performance, fast time to market and great flexibility for the design. Among the available technologies, SRAM-based FPGAs are well suited for space and avionics for their on-site reconfiguration, feature which is not available in Application Specific Integrated Circuits (ASICs). Although ASICs offer better performance and require less energy, they are not the most suitable candidates for small production "markets" such as space and avionics fields because of their initial production cost and lack of reconfiguration.

Despite these attractive characteristics, designers are reluctant to use FPGAs for critical applications as the mission profiles of these systems may include harsh environments, in which it is exposed, for instance, to ionizing radiation [1][2]. Single-event effects (SEE) induced by the interaction of energetic particles with integrated circuits are a well-known threat for space systems directly exposed to cosmic rays and solar flares. But they are also a concern for applications operating in the earth's atmosphere or for devices manufactured using modern process technologies which are in principle sensitive to the interaction of atmospheric neutrons. The most probable consequences of impinging energetic particles on SRAM-based FPGAs are the Single-Event Upsets (SEUs) and Multiple-Bit Upsets (MBUs) occurring either in the embedded design or in configuration memories. Particle induced faults in the design are temporary and they can be recovered by a reset. However faults induced in configuration memories are permanent and the only way the cope with them is to reconfigure the FPGA. Such faults may directly result in a "mutation" of the function implemented in the FPGA [3].

Protection against SEUs in configuration memory must be taken into account by the designer. Design level solutions, such as the well-known Triple Modular Redundancy (TMR) technique, are often adopted in order to build fault-tolerant architectures in SRAM-based FPGAs [4], but they always require finding a compromise between fault-tolerance and resource overhead or performance penalty. However the TMR technique has a weakness: its final voter. Indeed this part of the design is the only one which does not tolerate faults. In this paper are presented preliminary results issued from fault injection campaigns performed on three different versions of a representative application: a crypto-core and two derived fault-tolerant versions, duplex and TMR, implemented in a SRAM-based FPGA (the Xilinx Virtex II). The results of fault injection experiments on the TMR implementation were confronted to the ones issued from "accelerated tests" performed in a cyclotron, during which the tested applications were exposed to the effects of heavy-ions beams.

In section II are provided details of the Device Under Test (DUT). Section III describes the experimental platform used in this work. The crypto-core application implemented in the FPGA is presented in section IV. The test's methodologies are presented in section V. Experimental results issued from particle accelerator experiments and fault injection campaigns are given in section VI. Finally conclusions and perspectives are proposed in section VII.

II. THE DUT

The DUT used throughout these experiments is part of the Xilinx Virtex-II family. The XC2V1000-4FF896C is a 1 million system gates component mounted in a 896 ball Flip-Chip package [5]. The chip is processed in a

978-1-4244-7785-2/10/$26.00 ©2010 IEEE

0.15 µm / 0.12 µm CMOS 8-layer metal technology. Heavy ions produced in particle accelerators have energies much lower than natural ions, thus it is mandatory to thin down the die to 90µm for ground testing in order for the particles to penetrate the 700µm of silicone to reach the active zones. Laser beams may suffer the same problem depending on their parameters.

The specificity of SRAM-based FPGAs is the configuration memory used to configure all the DUT resources. In the case of the DUT chosen here, the memory size is 4,082,592 bits. Each resource has its configuration bit located next to it, therefore these bits are spread all over the core area.

A bit-flip in the configuration memory may have a great impact on the design itself, thus changing the behavior of the application. Moreover faults in the configuration memory will be "permanent" remaining until the next reconfiguration.

As significant examples can be mentioned:

- SEUs occurring in the configuration of a LUT may change its logic function.
- SEUs occurring in an interconnection may create a short cut between two routes or open a route.

Virtex FPGAs offer the possibility to read the configuration memory at any time. This process, called *readback*, offers the possibility to detect SEUs.

III. THESIC+ TESTER

THESIC+ is a generic and flexible test platform developed at TIMA laboratory. It is an upgraded version of the one presented in [6]. Figure 3. depicts the block diagram of the THESIC+ testbed. Its architecture is based on two FPGAs. The COM FPGA contains a LEON2 processor and is in charge of the communication between the user computer and the resources available on the board. It also monitors the DUT current in order to protect it against Single Event Latchups (SELs). Data transfers are performed over the 100Mbits Ethernet network. The Chipset FPGA contains the user design capable of interfacing the DUT with the tester.

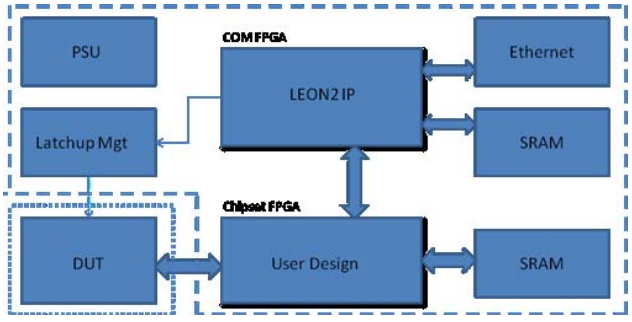

Figure 3. THESIC+ Block diagram

A daughterboard embeds the DUT in order to easily change the component to be tested. Figure 4. depicts the experimental testbed.

Radiation ground testing and fault injection campaigns were automatically performed by means of a controller implemented in the Chipset FPGA which has in charge the following tasks:

- Load FPGA configuration binary file and input vectors for the DUT application.
- Configure the FPGA.
- Provide input vectors to the DUT application.
- Check the DUT application results.
- Readback FPGA configuration memory.
- Send results to the computer.

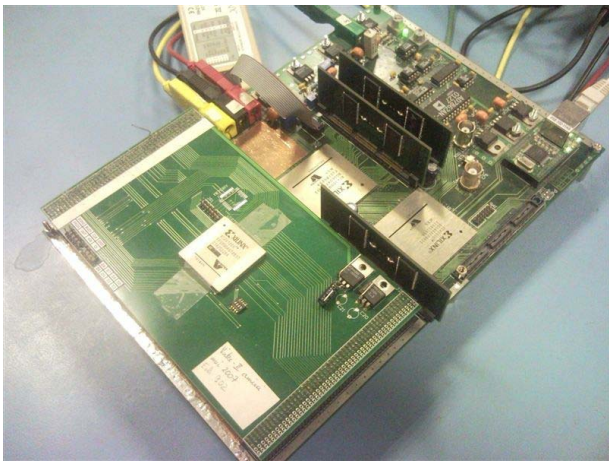

Figure 4. Fault injection experiment setup

IV. THE TESTED APPLICATIONS

The chosen application is based on a triple DES (Data Encryption Standard) algorithm, also named DES3, written in Verilog [7]. It is based on the DES algorithm which encrypts 64-bit data using a 56-bit key in 16 clock cycles. In fact a DES3 encryption is achieve by three consecutive DES encryptions, thus requiring three 56-bit keys and 48 clock cycles.

As the application requires too many I/Os, a controller was added in order to sequentially load the data and the keys.

The first implemented application, named "single", is composed of two identical and chained DES3. The first one encrypts the data and the second one does the reverse process. Consequently the output data should be the same as the input one.

The second application, named "duplex", is composed of two identical "single" applications. Consequently 2 chains of DES3 process the same data at the same time. A comparator tells whether the two outputs are the same

(output value “0”) or if they are different (output value “1”).

The third application, named “TMR”, is composed of three “single” chains doing the same calculation. A majority voter compares the three outputs and provides the status of the result through five significant values:

- “0” when all the results are the identical.
- “1” if chain 1 provides a result different from the two others.
- “2” if chain 2 provides a result different from the two others.
- “3” if chain 3 provides a result different from the two others.
- “4” if the three outputs are different.

The 64-bit output data chosen by the voter is always provided in order to allow an external controller to check its validity. This controller is embedded in the THESIC+ motherboard, thus it is not exposed to the radiation’s beams.

TABLE I. FPGA RESOURCE UTILIZATION

Resources	single		duplex		TMR	
Slice FF	798	7%	1064	10%	1330	12%
LUTs	2176	21%	4397	42%	6831	66%
Slices	1434	28%	2607	50%	3868	75%

The FPGA resource utilization is presented in TABLE I. whereas the architecture of the three tested applications is given in Figure 3.

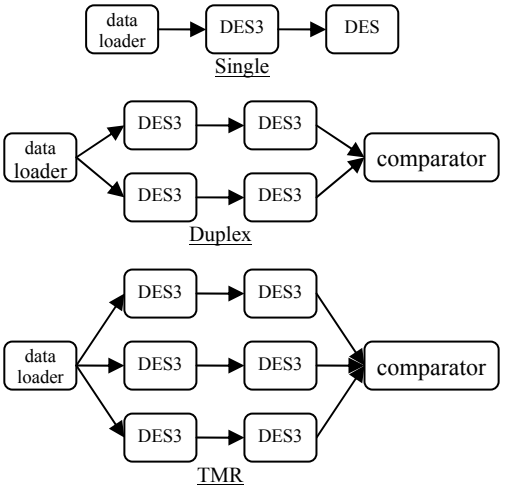

Figure 3. The three tested applications

V. TEST METHODOLOGIES

This section presents the methodologies used for radiation ground testing and for fault injection campaigns.

A. Heavy Ion campaign Figures and Tables

The radiation ground testing campaign was conducted in Louvain la Neuve, Belgium, at the HIF (Heavy Ion Facility) cyclotron [8][9]. Due to the allocated beam time, only the TMR application was exposed to two different particle beams: Carbon and Argon. An important feature of particle’s beam is the energy they deposit in Silicon, magnitude so-called LET (Linear Energy Transfer). A high LET particle would generate a high SEU rate in the configuration memory, thus provoking errors on the application outputs after few runs. The shutter requires some hundred of milliseconds to respond. Consequently exposition time of the DUT must be significantly greater than the shutter response time in order to get an accurate fluency measure. For this reason the selected particle’s LET is located in the bottom and in the elbow of the cross-section curve obtained in reference [11]. The Carbon ion and the Argon ion have respectively a LET of 1.2MeV/mg/cm² and 10.1MeV/mg/cm².

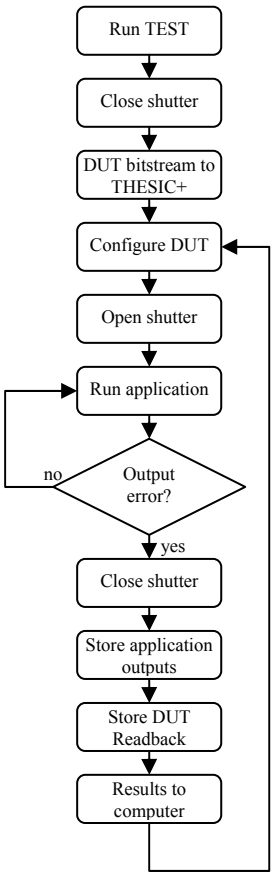

Figure 4. Heavy ion campaign flow diagram

When performing radiation experiments with heavy-ions, the THESIC+ platform and the Virtex-II daughterboard are both placed in a vacuum chamber but only the DUT is exposed to the particles. The so-called "accelerated test" consists in exposing the application while data are encoded continuously (Figure 4.). After each encryption sequence the results and the voter outputs are checked by THESIC+ to confront them to the reference values. Whenever an error is detected by the TMR voter and/or by THESIC+ the beam is automatically stopped by a mechanical shutter. This allows performing the readback followed by the reconfiguration of the FPGA with the guarantee that particles are not perturbing this important step. Afterwards the shutter is removed and the encryption process starts again.

B. Fault injection campaign

Fault injection campaigns were conducted by hardware/software means on the three applications. In these experiments, the content of a bit, randomly chosen among the whole configuration bitstream, is inverted during the FPGA configuration, thus simulating the occurrence of a SEU (Figure 5). A Mersenne random number generator was used [10] for this purpose. The data is then encoded and the outputs are saved in THESIC+ memory. The computer retrieves these data and sends the next fault injection vector to the tester and the procedure starts again.

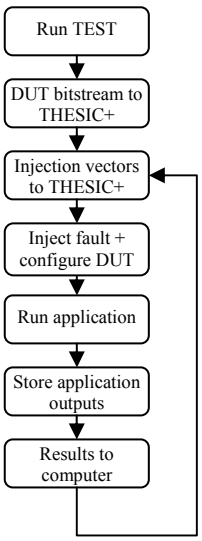

Figure 5. Fault injection flow diagram

TABLE II. NUMBER OF ERRORS OBTAINED ON THE TMR UNDER HEAVY IONS

Particle	1 way error	3 way error	Unexp. TMR error	Critical errors	Nb. of runs	Total fluency
Carbon	51	0	0	0	187,469,275	158,543
Argon	1,278	1	3	34	750,688,226	437,095

TABLE III. MEAN NUMBER OF SECONDS FOR THE OCCURRENCE OF EACH FAULT

Particle	Non-critical error	TMR failure	TMR critical failure	Critical errors
Carbone	7.81	N/A	N/A	N/A
Argon	1.25	1,595	532	46

TABLE IV. MEAN NUMBER OF PARTICLES REQUIRED BETWEEN 2 OCCURRENCES OF EACH FAULT

Particle	Non-critical error	TMR failure	TMR critical failure	Critical errors
Carbone	3,108	N/A	N/A	N/A
Argon	342	437,094	145,698	12,855

TABLE V. PERCENTAGE FOR EACH TYPE OF ERROR ACCORDING TO THE NUMBER OF SEUS IN THE CONFIGURATION MEMORY

Particle	1 way error		3 way error		Unexpected TMR error		Critical errors		Number of RB SEUs
Carbon	51	10.83%	0	0%	0	0%	0	0%	471
Argon	1,278	6.08%	1	0.005%	3	0.014%	34	0.16%	21030

TABLE VI. NUMBER OF ERRORS PREDICTED WITH FAULT INJECTION CAMPAIGNS

Applications	Detected faults		Wrong error detections		Critical errors		% of faults provoking an error	Number of runs
Single	N/A	0%	N/A	0%	1,339	0.50%	0.50%	267,438
Duplex	4,853	2.28%	N/A	0%	141	0.07%	2.35%	212,530
TMR	14,564	3.42%	237	0.06%	319	0.08%	3.55%	426,217

VI. EXPERIMENTAL RESULTS

A. Heavy Ion campaign's results

The radiation test campaign provided results on the behavior of TMR application being exposed to the selected particle's beams: Carbon and Argon. Four types of errors were considered and detected:

- **"1 way errors"** are errors detected by the TMR putting in evidence that one chain outputs a wrong result. The other two chains being right, the voter gives a correct result.
- **"3 way errors"** occur when the three chains output different results. So the voter cannot provide the correct answer but it warns that the encryption results cannot be trusted and so they must be done again.
- **"Unexpected TMR errors"** occur when the TMR provides an error code which does not exist in the application. Thus, the result cannot be considered as relevant and the encryption must be done again.
- **"Critical errors"** occur when the TMR states that the result is correct but while the external THESIC+ controller detects an error. This puts in evidence a critical weakness of the TMR architecture as an undetected error is propagated in the system.

TABLE II. presents the total number of encryptions done, the particle's fluency received by the tested FPGA during the radiation experiment and the number of errors induced for the two ion's beams. TABLE III. shows respectively the mean time between two occurrences of each type of fault. TABLE IV. provides the average number of particles required to generate each fault. In reference [11] it was shown that the Virtex II has a low sensitivity to Carbon ions. Indeed it requires nine times more Carbon ions than Argon ions to generate an error. The time ratio is different as the selected fluxes where different.

TABLE V. presents the percentage for each type of errors according to the number of configuration memory SEUs generated by the particles.

B. Fault injection campaign

TABLE VI. presents the error rates obtained from fault injection campaigns applied on the three previously described applications. The first observation is that the percentage of particles provoking and error increases when applying mitigation schemes. This is due to the increase of the amount of resources required by the applications.

The "single" application provides a raw output, so the only type of error observed are critical errors detected by Thesic+ when the result is wrong. Only 0.50% of injected faults provoked an error on the output.

The "duplex" application is able to detect two different results, representing 2.35% of the injected faults. However it is important to notice that 0.066% of the errors were detected by THESIC+ being not detected by the application.

Finally the "TMR" application was able to detect and propose a correct result for 3.42% of the faults. 0.06% of the injected faults provoked the detection of an error although the result was correct and 0.08% of the injections provide an error which was not seen by the voter.

VII. CONCLUSIONS AND PERSPECTIVES

The work presented in this paper confronted the results issued from radiation testing to those resulting from fault injection campaigns for a cryptocore application and two fault tolerant versions based on duplex and TMR approaches implemented in an SRAM-based FPGA.

The obtained results put in evidence that some faults on the configuration memory may provoke an application "mutation" which results in the inability of the voter to detect the fault. Such faults constitute a critical challenge for applications requiring high reliability such as those devoted to operate in harsh environments (space, avionics, …). For ground level applications, the probability of occurrence of these faults is very low, but, the omnipresence of integrated circuits and systems gives a non negligible probability of occurrence even if the impinging particles, basically thermal neutrons, have very low fluxes.

It is important to note that in this preliminary fault injection study, multiple faults (MBU) that may occur in advanced integrated circuits as the result of the impact of a single particle were not considered. Such a conjecture should be considered but requires a precise knowledge of the layout of the considered memories in future work.

REFERENCES

[1] E. Normand, "*Single-Event Effects in Avionics*", IEEE Trans. Nucl. Sci., Vol. 43, n° 2, pp. 461-474, April 1966.

[2] T. Ma, P. Dressendorfer, "*Ionizing Radiation Effects in MOS Devices and Circuits*", Wiley Eds., New York, 1989.

[3] K. Morgan, M. Caffrey, P. Graham, E. Johnson, B. Pratt, M. Wirthlin, "*SEU-induced persistent error propagation in FPGAs*", IEEE Trans. Nucl. Sci., Vol. 52, n° 6, pp. 2438-45, 2005.

[4] F.L. Kastensmidt, L. Sterpone, L. Carro, M.S. Reorda, "*On the optimal design of triple modular redundancy logic for SRAM-based FPGAs*", Proc. Of Design, Automation and Test in Europe (DATE) 2005, vol. 2, pp. 1290-5, 2005.

[5] Xilinx, "*Virtex-II Platform FPGAs: Complete Data Sheet*", http://www.xilinx.com, March 2005.

[6] F. Faure, P. Peronnard, and R. Velazco, "*Thesic+: A flexible system for see testing*", in Proc. of RADECS, 2002.

[7] Opencores: *http://www.opencores.org/project,des*

[8] G. Berger, G. Ryckewaert, R. Harboe-Sorensen, L. Adams, "*The Heavy Ion Irradiation Facility at CYCLONE - a dedicated SEE beam line*", IEEE NSREC Workshop, 1996.

[9] G. Berger, G. Ryckewaert, R. Harboe-Sorensen, "*CYCLONE – A Multipurpose Heavy Ion, Proton and Neutron SEE Test Site",* RADECS Workshop, pp. 51-55, 1997.

[10] Mersenne twister: http://www.math.sci.hiroshima-u.ac.jp/~m-mat/MT/emt.html

[11] R. Koga, J. George, G. Swift, C. Yui, L. Edmonds, C. Carmichael, T. Langley, P. Murray, K. Lanes, M. Napier, "*Comparison of Xilinx Virtex-II FPGA SEE sensitivities to protons and heavy ions*", IEEE Trans. Nucl. Sci., Vol. 51, n° 5, pp. 2825-33, October 2004.