A study of the Single Event Effects Impact on Functional Mapping within Flash-based FPGAs

F. Abate, L. Sterpone, M. Violante
Politecnico di Torino
DAUIN - Dipartimento di Automatica e Informatica
Torino, Italy

F. Lima Kastensmidt
Universidade Federal do Rio Grande do Sul
Instituto de Informatica
Porto Alegre, Brasil

***Abstract*—Flash-based FPGAs are increasingly demanded in safety critical fields, in particular space and avionic ones, due to their non-volatile configuration memory. Although they are almost immune to permanent loss of the configuration data, they are composed of floating gate based switches that can suffer transient effects if hit by high energetic particles with critical consequences on the implemented logic. This paper presents a new way for the analysis of the impact of Single Event Effects in Flash-based FPGAs. We proposed a new methodology to identify the most critical switches inside the configuration logic block and the most redundant and robust configuration selection for each logic function. The experimental results achieved by fault injection demonstrated the feasibility of the proposed method and show that by using the most robust functional mapping it is possible to enhance the reliability of the entire design with respect to a not robust ones.**

I. INTRODUCTION

Single Event Effects (SEEs) are caused by a single, energetic particle striking the silicon and depositing enough charge to modify the voltage of one or more transistor nodes [1] [2]. The consequence of this transient voltage pulse can be a bit-flip in a memory cell, known as Single Event Upset (SEU); or a Single Event Transient (SET) that can propagate trough the logic and possibly be captured by a memory cell. Device operating both at ground and avionic level and at space environment can be affected by transient errors. In addition, devices operating at space environment can suffer long-term radiation effects, known as Total Ionization Dose(TID), causing transistor behaviour degradation (current, voltage, and other parameters).

Field Programmable Gate Arrays (FPGAs) are programmable devices that are more and more used in many commercial, military and remote applications due to their high density, performance, fast turnaround time, and most important: re-configuration capability. Two programming technologies are widely used to implement the FPGA reprogrammable switches, such as SRAM and Flash cells. SRAM-based FPGAs have been intensely investigated under SEE in last few years [3] [4] [5] [6]. The SRAM-based switches have shown to be very sensitive to SEU in particular in the routing elements. Once a SEU occurs in the SRAM cell that controls the switch, the bit-flip may turn on or off the switch generating a persistent effect that can be corrected by only re-loading the bitstream. Mitigations techniques have been proposed based on Triple Modular Redundancy (XTMR) [5] and scrubbing.

Flash-based FPGAs provide more SEE robust results combining re-configurability with no-volatile memories. However, because flash-based switches are also composed of transistors susceptible to SEE, flash-based switches can present transient errors that may provoke SET in the programmed design. Related works [7] focus on SET propagation through the logic and the effects of these SET pulses and bit-flips in the application. However, no work so far has been discussed the effect of having transients also in the configuration switches and the effect of having the logic tile configuration broken for a short period of time.

In this paper, we analyze the effects of transients in the flash-based switches and how the switch configurations and consequently the functional mapping can influence in the fault tolerance of the final mapped circuit. A methodology is proposed to investigate the most robust functional mapping into the flash-based FPGA logic tiles and to select the most and less SEE sensitive switches.

Starting from a set of assumptions, illustrated in the section II, based on the electrical characteristics of the floating gate technology, the paper proposes a new way of analyzing the SET's influences on Flash-based switches. The method is based on modelling and performing a functional analysis of the basic logic element of commonly used Flash-based FPGAs (the case study considered are FPGAs manufactured by Actel), this is explained in the section III. The experimental analysis performed by using fault injection techniques is illustrated in section IV. Finally, some conclusions and future works are reported in section V.

II. SEE ASSUMPTIONS IN FLASH-BASED FPGAS

The first concern in radiation effect affecting floating gate transistors has been TID. Some experiments leaded on flash-based FPGAs shows the impact of TID effect on the floating gate reliability due to radiation-induced charge leakage [7][10].

However, up to now, the radiation effect on the floating gate transistor is not well understood. A lot of work must still be done to analyze and efficiently evaluate the sensitiveness of this type of FPGA under energetic particles. In this work, we focus on fault injection by developing a case-study model where it is possible to inject faults and analyze the effects.

As an initial case study device, we consider the scheme from a recent released flash-based FPGA fabricated by ACTEL (ProASIC3). Anyway all the considerations are general and can be extended to other flash-based FPGA architectures. The ProASIC3 [9] core consists of logic tiles

978-3-9810801-5-5/DATE09 © 2009 EDAA

called *VersaTile* and routing structures (as shown in Figure 1). The VersaTile can map the logic and sequential gates of a design. Versatiles are connected with each other through routing structures and floating gate (FG) switches. Based on available literature datasheets from manufacturer [9], we set the hypothesis that 32 floating based switches control each tile.

Considering the logic tile scheme illustrated in Figure 1, it is possible to identify four single event effects:

1. The first effect occurs when a particle hits a sensitive node of a logic gate cell provoking a SET that may or may not propagate through the logic.
2. The second effect is defined when a SET occurs in the logic configured to implement a latch. In this case, because of the feedback path the SET turns into a SEU,

3/4. The third and fourth effects provoke the same logical upsets of 1 and 2 but they occur inside the floating gate switch.

Figure 3 shows the complementary floating gate switches model. This CFG switch was presented in [7]. The CFG is defined at on-state when the NMOS switch is turned on. At on-state both NFG and PFG are configured to the high-Vt state, and they are referred as programmed-state. Likewise, when CFG is at the off-state, the switch is turned off and both NFG and PFG are erased to low-Vt state, or erased-state. In this case, there are two main possible effects:

- *Assumption 1: Turn-on or off the NMOS switch*. A particle can hit the drain of the floating gate at off state provoking a SET. This SET can turn-on or off the NMOS switch as described in Figure 3. Consequently, the configuration switches invert their original state for short period of time according to the duration of the SET. Another cause for turning-on and off the NMOS switch can be explained by the fact that a direct ionization in a pass transistor may turn-on or off the channel as discussed in [8].
- *Assumption 2: SET in the logic block:* this effect occurs when transient fault happen in the NMOS drain at off-state, as illustrated in Figure 3.

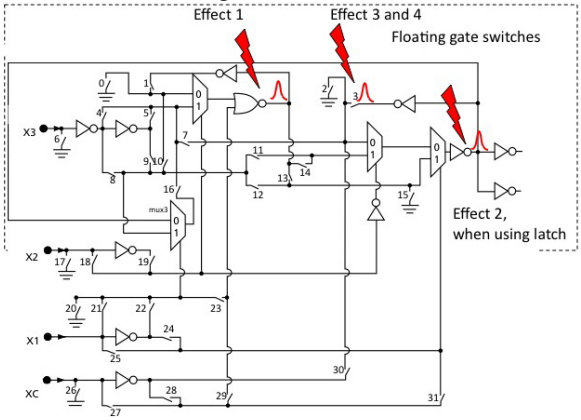

Figure 1. The Versatile Logic Block Scheme and some possible radiation effects.

By analyzing these effects, one can notice that the floating gate configuration switches must be analyzed for fault tolerance. The effect in the Flash-based FPGA switches are not persistent as it is in the case of SRAM-based FPGAs, but it is mandatory to analyze how the configurations can effect the fault tolerance of the final mapped circuit once the switches can present transients.

In fact, not all upset switches may provoke an error because not all of them are used to map each logic function. Please note that this aspect is considered by the proposed analysis methodology since the first goal is to find a best configuration that minimizes the configuration failure points. In other words, to find the most and less SEE sensitive switches in order to be able to propose more robust mappings and schemes for each configurable tile.

III. THE PROPOSED METHODOLOGY TO IDENTIFY THE MOST AND LESS CRITICAL CONFIGURATION POINTS

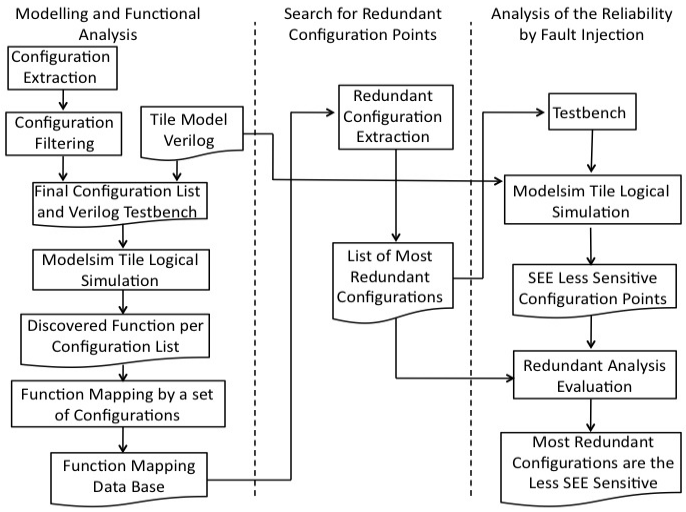

Figure 2. The Proposed Design Flow for the Analysis

The proposed methodology is divided into three phases, Figure 2:

- *Phase I: Modelling and Functional Analysis*: the time model is developed according to the scheme reported in Figure 1. Logical simulation is executed to extract information about the behaviour of the logic block.
- *Phase II: Search for Redundant Configuration Points*: based on the previous results, the best tile configurations is found for logic function considering redundancy properties.
- *Phase III: Analysis of the Reliability by Fault Injection:* the effectiveness of the best configuration is analyzed by fault injection.

A. Modeling and functional analysis

A Tile Model Verilog file has been developed according to the scheme presented in Figure 1. The Verilog HDL language allows to describe the circuit at both gate and transistor level. The 32 points of configuration switches have been considered as MOSFET NMOS pass-transistors controlled by floating gate transistors. A 32 bit input string configures the top-level entity simulating the flash configuration memory behavior. On the base of the 32 points of configuration, the Configuration Extraction operation extracts all the different possible tile configurations. However, considering the elevated number of configuration, the Configuration Filtering discards those

erroneous one from the electrical point of view (as for example, a configuration of the logic in which the on state of two switch would provoke a short circuit in the node where the two enabled paths converge). The reduced configuration set resulting by Configuration Filtering is reported in the Final Configuration List and Verilog Testbench.

Tile Modelsim Logical Simulation discovers the functions corresponding to each configuration of tile. In particular, all the possible input stimuli of the three input LUT (X3, X2, X1) are applied for a specific configuration of the tile. The resulted functions in the form of 8 bit string and the 32-bit configuration string (correspondent to the 32 point of configuration of the logic tile) are written into *Discovered Function per Configuration List*.

Afterwards, the *Function Mapping by a set of Configurations* phase classifies the configurations according to the implemented function. The result of this analysis is a *Function Mapping Data Base* containing all possible configurations corresponding to each logic function that the tile can implement. The obtained result reports 121 different functions.

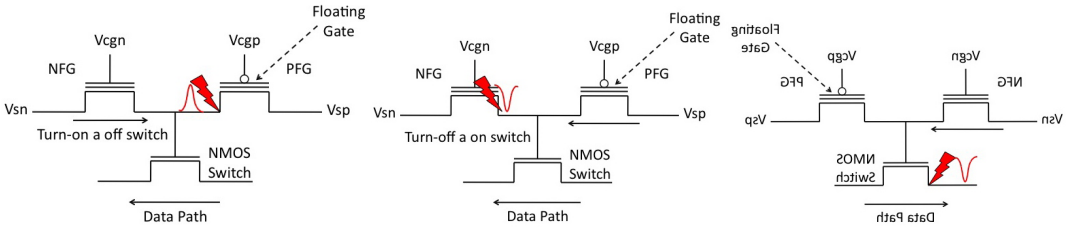

Figure 3. Complementary Floating Gate (CFG) Switch Schematic under radiation effect in the floating gate transistors and NMOS transistors. The NMOS switch can turn on or off depending on the floating gate configurations.

B. *Searching for Redundant Configuration Points*

The high number of configurations collected in the *Function Mapping Data Base* needs further examinations in order to avoid redundant configurations. For each function, the *Redundant Configuration Extraction* phase research the possible forms of redundancy in the configuration bit strings by the use of an automatic tool. In fact, the higher the levels of redundancy of a configuration for a specific function, the better the reliability level when a SET upsets the programmed logic block.

As previously mentioned, for a specific function exists a limited set of configuration, defined as configuration-per-function set. The redundancy property for a bit of a configuration string is the property of switching by a configuration to another of the same configuration-per-function set, by simply upsetting that bit. Those strings containing n bit satisfying the redundant property are defined as n-bit redundant configuration strings. Figure 4 shows an example of 2-bit redundant configuration string. Thus, redundant configuration concept demonstrates that the reliability level changes according to the way the tile is programmed. The configuration string that presents the highest value of bit satisfying the redundant property is also the most robust one.

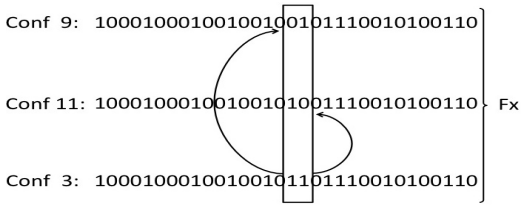

Figure 4. An example of Configuration Redundancy. If one of the two highlighted bits of Conf 3 upsets, Conf 3 switches in Conf 11 or Conf 9 still giving give Fx in output.

As previously mentioned, an automatic program tool for extracting the best configurations set from the reliability point of view has been developed. Inside the same configuration-per-function set, the program creates subsets of configurations according to the different level of redundancy. Subsets are created exploiting the concept of hamming distance between two configurations. In particular, for each configuration the tool finds the subset of strings having hamming distance of 1 with it..At the end of configuration analysis phase, all the most redundant configurations are grouped by function in a *List of Most Redundant Configurations.*

C. *Analyzing the Reliability by Fault Injection*

In this phase faults are injected during *Tile Modelsim Logical Simulation* exploiting the *Tile Model Verilog* description and *Testbench* resulting from the *List of Most Redundant Configurations*. In this phase, the transient upset of a configuration switch is considered as fault model and the output function is compared with the golden one. An additional *Redundant Analysis Evaluation* of the *SEE Less Sensitive Configuration Points* in conjunction with the *List of Most Redundant Configurations* indicates that the *Most Redundant Configuration are the Less SEE Sensitive*.

IV. EXPERIMENTAL RESULTS

The results coming from the Fault Injection phase represent the starting point for several affordable considerations. According to the remarks reported in the previous sections, the bits of a configuration string are classified into three classes:

- *Redundant bit*: the redundancy property is satisfied. Thus, if a bit-flip occurs the configuration string switches to another one giving in output the same correct function.
- *Fault bit*: the bit is sensible and if a SET occurring in the floating gate flash memory upsets the state of the corresponding switch, the output function changes.
- *No-error bit*: the bit is not a redundant bit, but if an upset occurs, the output function persists in the correct state.

It is important to underline that a bit may be associated to different classes according to the implemented function. The graphic illustrated in Figure 5 shows the percentage of the

three classes of configuration bit according to the implemented function.

Additionally, the results reported in Figure 5 emphasizes not only that some configurations for the same function are better than others, but also that some set of functions present a redundancy level higher than others. In detail, the two most robust function result $F_1 = \overline{A+B+C}$ and $F_2 = AC + AB$ both with the 43,75% of redundant bits and the 34,37% of fault one. Conversely, the less tolerant functions are $F_3 = \overline{B} + \overline{AC} + AC$ and $F_4 = \overline{A} + \overline{B}C$ both with 90,625% of fault bits and 9,375% of redundant one. In order to reach a better reliability level it is possible to implement the function F_3 and F_4 as the equivalent F_1 representation reducing of the 56,25% the number of fault bits. Such information may be extremely important during the functional mapping phase of a real circuit in the FPGA. Figure 6 shows how many times each switch is classified as redundant or faulty during the configuration of all the 121 detected functions. The graphic describes the results in case the most redundant configurations are used to program the tile. The results emphasize the weakest switches of the logic tile from the topological point of view.

In order to identify the most critical location inside of the logic tile, we identify the most and less sensitive points of configuration. With reference to Figure 1, the first observation is that the most sensitive configuration points are in the input path or in the not allowed configurations or directly connected to the multiplexer input. They are due to different effects: the frequent errors in 6, 17 are due to the short circuit on the inputs; the couple 18-19 are one of forbidden configuration constrains and they cannot be contemporary in the same state; finally the short circuit caused by the 2 on-state might provoke function changing with high probability. Moreover, 7, 21, 22, 23 and 28 are the less sensitive ones because the logic path they enable does not affect significantly on the implanted function.

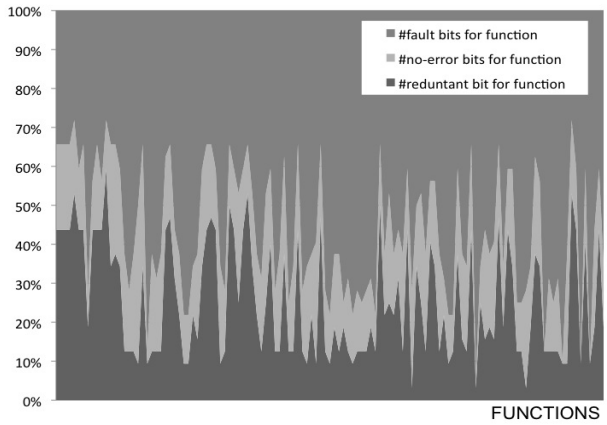

Figure 5. Percentage Graph of Faults, Redundant and No-error bit for each implementable function

V. CONCLUSIONS AND FUTURE WORKS

The work presented in this paper offers an innovative analysis of the SEE effects on the configuration of the Flash-based FPGAs. The major contribution of this work is given by the SET sensitivity analysis of the functional mapping on FPGA configurable logic block implemented on Flash-based FPGAs. The experimental analysis performed on different mapping may lead to an increase of almost 34% of redundant bit and a decrease of about 56% of fault bits on the average. In details, the most sensitive configuration points of the logic tail have been detected and almost the 43,75% provoke a fault on the 66,12% of function, see Figure 6. Some future activities have been delighted thanks to the reached results. First of all, it is possible to develop mitigation techniques focusing on the most critical switches of the circuit or in a most suitable functional mapping. This may reduce area overhead and bring to a better compromise between area cost and reliability. Secondly, a new functional mapper embedding the function's redundant bit information is planned to be developed.

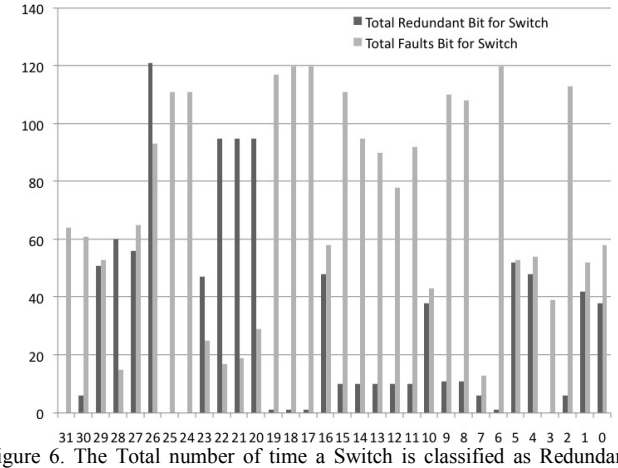

Figure 6. The Total number of time a Switch is classified as Redundant or Fault.

REFERENCES

[1] Ibe, E.; Kameyama, H.; Yahagi, Y.; Yamaguchi, H.; "Single event effects as a reliability issue of IT infrastructure", Information Technology and Applications, Third International Conference on. Volume 1, 4-7 July 2005.

[2] Messenger, G. C.; "Collection of Charge on Junction Nodes from Ion Tracks", *IEEE Transactions on Nuclear Science*, Volume 29, Dec. 1982 Page(s):2024 – 2031.

[3] L. Sterpone and M. Violante, "A new reliability-oriented place and route algorithm for SRAM-based FPGAs", IEEE Transactions on Computers, Vol. 55, Issue 6, June, 2006, pp. 732 – 744.

[4] F. Lima Kanstensmidt, L. Sterpone, L. Carro, M. S. Reorda, "On the optimal design of triple modular redundancy logic for SRAM-based FPGAs", Design, Automation and Test in Europe, 2005, pp. 1290 – 1295, Vol. 2, 2005

[5] C. Carmichael, "Triple Module Redundancy Design Techniques for Virtex FPGAs", Xilinx Application Notes, XAPP197, 2001.

[6] M. Berg, "Fault Tolerance Implementation within SRAM-based FPGA design based upon the increased level of single event upset susceptibility", IOLTS, pp. 89-91, 2006.

[7] J.J. Wang, Sana Rezgui, Yinming Sun, Frank Hawley, Farid Issaq, Salman Kabir, Brian Cronquist, John McCollum, Hui Pan, and Richard Chan, "A Novel Radiation-Tolerant Floating-Gate Configuration Cell for Flash-Based FPGA", NSREC presentation, 2008.

[8] Hutson, J. M.; Ramachandran, V.; Bhuva, B. L.; Zhu, X.; Schrimpf, R. D.; Amusan, O. A.; Massengill, L; "Single Event-Induced Error Propagation Through Nominally-off Transmission Gates", *IEEE Transactions on Nuclear Science*, Volume 53, Part 1, Dec. 2006 Page(s):3558 – 3562.

[9] ProAsic3 Flash Family FPGAs Datasheet.

[10] Wang, J.J.; Samiee, S.; Chen, H.-S.; Huang, C.-K.; Cheung, M.; Borillo, J.; Sun, S.-N.; Cronquist, B.; McCollum, J.; "Total ionizing dose effects on flash-based field programmable gate array", *IEEE Transactions on Nuclear Science*, Volume 51, Part 2, Dec. 2004 Page(s):3759 – 3766.