Methodologies to Study Frequency-Dependent Single Event Effects Sensitivity in Flash-Based FPGAs

N. Battezzati, S. Gerardin, A. Manuzzato, D. Merodio, A. Paccagnella, C. Poivey, L. Sterpone, and M. Violante

***Abstract*—Flash-based FPGAs are more and more interesting for space applications because of their robustness against Single Event Upsets (SEUs) in configuration memory. However, as Single Event Effects (SEEs) are still a concern both for user memory and the configurable logic, accurate evaluations are needed to identify mitigation techniques for securing their use in space missions. In this paper the SEE sensitivity of circuits implemented in Flash-based FPGAs is evaluated with respect to the working frequency and different routing schemes. We outline different methodologies that can be used in order to characterize SEE sensitivity, using both heavy-ions radiation experiments and analytical approaches. Experimental results detail the contributions of different SEEs as a function of operating frequency and routing on a realistic circuit.**

***Index Terms*—Field-programmable gate array (FPGA), flash-based, frequency, radiation testing, single event effects (SEEs).**

I. Introduction

FIELD-PROGRAMMABLE gate arrays (FPGAs) are getting more and more interesting for space applications because of their relatively high performance and low power dissipation joined by reconfiguration capabilities. However, the space environment is hostile for electronic devices, due to the high amount of radiation, mostly electrons, protons and heavy-ions [1]. Because of this, either hardened-by-technology or hardened-by-design FPGAs, where mitigation techniques are transparently applied within the device must be used in space. Otherwise, if one wants to utilize Commercial-Off-The-Shelf (COTS) FPGAs, user-implemented mitigation techniques are mandatory. COTS are very appealing as they promise a large performance increase combined with potential cost savings, with respect to technology-hardened FPGAs. Commercial FPGAs are thus under investigation for analyzing their sensitivity to radiation effects, and to identify which countermeasures could be applied in order to enable their usage in future space missions. In particular, Flash-based FPGAs are an attractive solution for applications operating in harsh radiation environments because the configuration memory does not upset like in SRAM-based ones. In such devices, the implementation of the desired circuit is accomplished by programming the floating-gate switches that compose the configuration memory [2]. These switches are used to configure the logic function of the FPGA logic cells (tiles) and the routing paths. Experiments have proven the immunity of these devices to Single Event Upsets (SEUs) [3] in the configuration memory. In this scenario, however, Single Event Transient (SET) phenomena become more critical. Radiation can induce voltage glitches in the combinational logic that could propagate to memory elements, and, if latched, the glitches result in either single or multiple memory elements being upset, depending on the fan-out stems of the affected combinational logic. Moreover, SEUs are possible in the user memory. Different kinds of Single Event Effects (SEEs) can thus induce faults in designs implemented in Flash-based FPGAs; from SEUs in user memory (flip flops and embedded SRAMs) to SETs in combinational logic, which can lead to multiple upsets, or in global lines, like clock and reset, that can lead to functional failures of the whole circuit or parts of it. In this scenario, it is important to understand the impact of the several possible effects on a design, defining the corresponding criticality level and identifying the different circuit parameters weight on the overall design sensitivity.

The main contribution of this work is a set of methodologies that we propose to analyze the most significant parameters that could affect the SEE-induced error rate in a realistic circuit (e.g., combinational logic surrounded by register elements implementing an arithmetic function) implemented in Flash-based FPGAs. In particular, we developed hardware/software tools that implement these methodologies. We applied these tools to a realistic case study for assessing the impact of clock frequency and the routing architecture, on SEE sensitivity of a design implemented in Flash-based devices.

The paper is organized as follows: Sections II and III outline background information and previous works on SEE sensitivity evaluations in Flash-based FPGAs. Section IV describes different methodologies we used to analyze SEE sensitivity. Sections V and VI describe the experimental setup and results obtained and, finally, Section VII draws conclusions and directions for future works.

II. Background

Flash-based FPGAs are electronic devices composed by a matrix of configurable logic cells (tiles), which implement basic logic functions, connected by programmable interconnection routes that allow building complex circuits. Programmability is obtained by means of a Flash memory whose floating gate cells control switches within the FPGA. These switches are used to configure the logic function of the FPGA tiles and the routing paths.

Manuscript received July 16, 2009; revised September 16, 2009. Current version published December 09, 2009.

N. Battezzati, L. Sterpone, and M. Violante are with the Dipartimento of Automatica e Informatica, Politecnico di Torino, Torino 10129, Italy (e-mail: luca.sterpone@polito.it).

S. Gerardin, A. Manuzzato, and A. Paccagnella are with the Dipartimento di Ingegneria dell'Informazione, Università di Padova, Padova 35131, Italy.

D. Merodio and C. Poivey are with the European Space Agency (ESA), Noordwijk 2200 AG, The Netherlands.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNS.2009.2034316

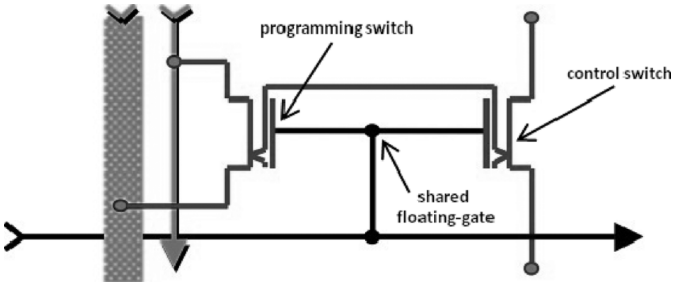

Fig. 1. Flash-based FPGA programming and control switch.

As described in [2], each switch used to connect or disconnect two FPGA nodes is implemented by means of a pass transistor and a smaller device used to control it. The switch programming is executed by writing the floating-gate, shared between the two devices, as in Fig. 1.

The FPGA matrix is composed of a number of logic tiles, which can implement any basic logic function (i.e., combinational or sequential) according to how it is programmed. Indeed, a combinational or sequential function can be defined by programming the switches controlling the paths within the tile itself. Fig. 2 shows a model of the tile, highlighting the different configuration switches. Such an architecture can implement any logic function with up to three inputs and one output, as well as a flip-flop or a latch. Moreover, considering the architecture illustrated in the picture, it is possible to identify three possible effects induced by a single particle striking one of the nodes:

Effect 1, which occurs when a particle hits a sensitive node of a logic gate cell provoking a pulse that propagates through the logic.

Effect 2, which occurs when a pulse happens in the logic configured to implement a latch. In this case, because of the feedback path of the programmed cell, the pulse may turn into an SEU.

Effect 3, which occurs when a particle strikes the floating gate switch provoking, because of the memory cell size, a transient pulse.

SEEs in the Flash-based FPGA switches do not cause permanent damage, as they consist in either SETs that if not sampled disappear after a certain time, or SEUs that although latched can disappear at the next update of the flip-flops (the same holds for latched SETs). Nevertheless, SEEs may affect the behavior of the circuit, and therefore it is mandatory to analyze how the logic, routing and configuration resources contribute to the SEE sensitivity. Moreover, as SETs are transient pulses, it is important to study the SEE sensitivity as a function of the frequency and the routing characteristics that may affect the delay of the circuit.

III. Previous Works

As mentioned in the introduction of this work, experiments have proven the immunity of commercial Flash-based FPGAs to SEUs in the configuration memory [3]. Two kinds of SEEs can thus affect the reliability of designs implemented in these devices: SEUs in user memory and SETs in combinational and sequential logic.

The problem of upsets in flip-flops and embedded SRAMs has been widely addressed in the past. Error Correcting Codes (ECCs) are the conventional solution to cope with soft errors in SRAMs, and Triple Modular Redundancy (TMR) is the most commonly used technique for mitigating soft errors in logic memory (flip-flops) [4]. In the cases where these techniques are too expensive or can not be applied, other methodologies have been studied and successfully applied, such as Built-In Current Sensors (BICS) [5] and Dual Interlocked Storage Cell (DICE) [6], that are mainly used in ASICs or hardened-by-design devices.

On the other hand, SETs are still under investigation, even if some mitigation techniques have been already proposed, such as in [7]. Technology shrinking as well as frequency increasing and voltage lowering, are making such faults more and more effective and frequent. Moreover, SETs depend on many parameters, like operational frequency, full arrival time and location. They can be considered as a random event, which requires an accurate characterization.

Many works investigated the nature of these events, studying the propagation of the transients through the combinational logic and the routing resources in ad-hoc designed circuits [8]–[10]. The origin and the propagation of SETs have been widely investigated on digital systems both by simulation and radiation experiments [11]. Recently, SETs have been investigated in Flash-based FPGAs [12], [13]. Previous works [8], [14], [15] reported SET propagation experiments on custom circuits designed specifically for observing SETs. Also some studies about SET dependency on clock frequency have been presented in [16]. However, to the best of our knowledge, the SET propagation problem has not been studied on realistic designs implemented on Flash-based FPGAs, and therefore the data available today may give a worst-case view of the phenomenon. The circuits studied so far were indeed developed for maximizing the probability of observing SETs by minimizing the effect of logic masking.

Although this kind of circuits allows easily studying the SET effects on the logic and the routing, they are far from being representative of realistic designs, since they have been intrinsically designed to favour and emphasize the SETs phenomena. The SETs observed in such circuits are therefore likely to be much worse than in real circuits, since in realistic circuits the combinational path has a limited number of gates between register elements and therefore narrower SET pulses can be expected [17]. Indeed, the long combinational paths used so far for studying SET propagation have the great advantage to avoid logic masking of the radiation effect; moreover, they can induce broadening or filtering effects [8] that are unrealistic in designs with a very small amount of logic levels between memory elements as happen in realistic circuits.

For these reasons, the set of methodologies we propose in this manuscript aim at performing SET analysis on realistic circuits intended for performing tasks reasonably representative on realistic circuits and to avoid the usage of circuit specifically designed for SET observation.

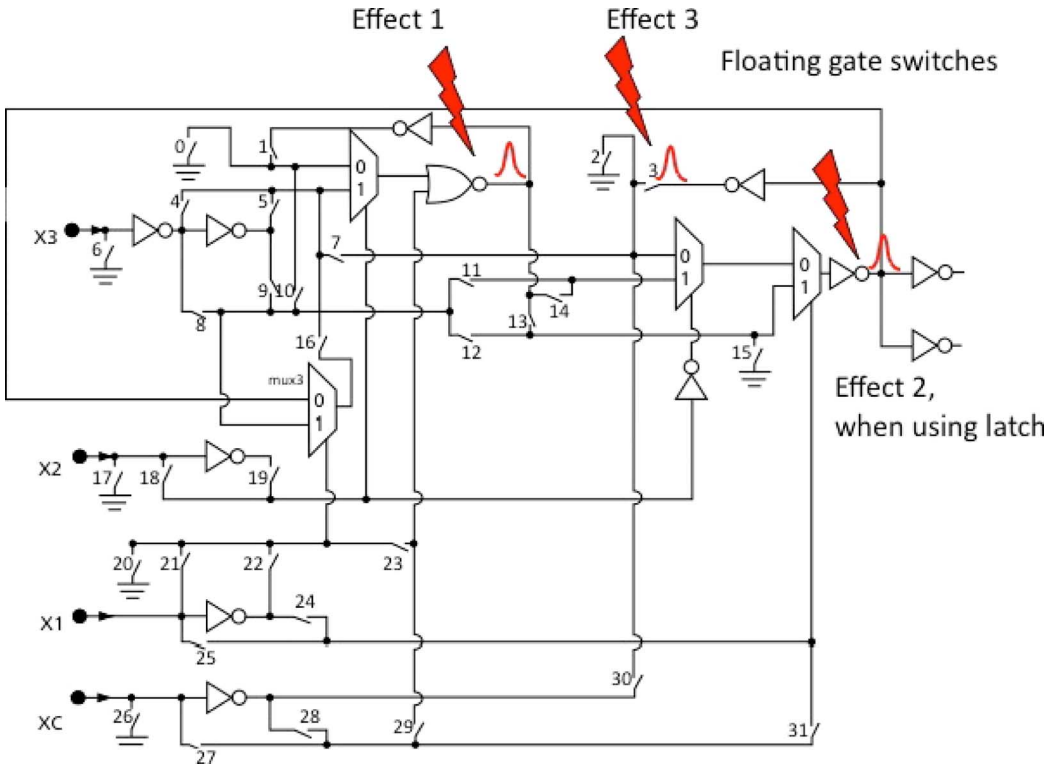

Fig. 2. The Tile logic block scheme and the possible radiation effects.

IV. Proposed Study Methodologies

Three different approaches have been used together in order to analyze data from singular points of view and then combine them to provide consolidated pictures of SEEs in Flash-based FPGAs with respect to different circuit parameters, in particular clock frequency and routing architecture. On the one hand, a flexible radiation testing environment has been set up, to collect data from accelerated experiments that provide realistic results. On the other hand, two software-based techniques have been put together to manage different circuit routing schemes and correlate them with radiation testing data. First, a software tool re-places the circuit resources leaving their functionality unchanged but modifying the routing; secondly, another tool analyzes the FPGA bitstream resulting from the previous transformation, computing the number of sensitive programmable points that changed from the first version of the circuit.

A. Test Setup

The testing environment we developed for evaluating SEEs in Flash-based FPGAs is composed of three main modules: a hardware module hosting the Device Under Test (DUT), another module with a Monitoring Device (MD) and, finally, a Clock Generation Module (CGM), as illustrated in Fig. 3.

The DUT, a Flash-based FPGA, is irradiated while operating at a clock frequency defined by the CGM. In the meanwhile, the MD monitors the DUT outputs (channels) detecting discrepancies in the functioning due to radiation-induced faults. The MD is implemented in an SRAM-based FPGA, but could be implemented in any other technology that provides reconfiguration.

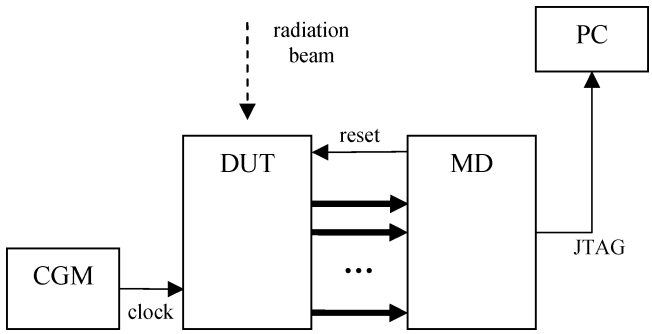

Fig. 3. Test system block diagram.

Finally, a PC is connected to the MD through the JTAG port, allowing us to download experimental results for further analysis and control the test execution.

The design on the MD is a general circuit that can monitor different events happening in the DUT according to how the DUT is designed. The MD is able to count the events happening in the DUT and store them into general-purpose registers that can be read by the software running on the PC. In particular, the MD design is composed of a first stage of N majority voters that vote the DUT output channels in order to protect the counting mechanisms from false faults happening in the transmission stage between the DUT and the MD. The maximum number of channels is defined by the maximum number of data connections available between the DUT and the MD divided by 3, because of output triplication. Every voted channel is then analyzed by a second stage, an FSM that counts faults and resets the DUT transmission stage preparing it for sending a new fault when it

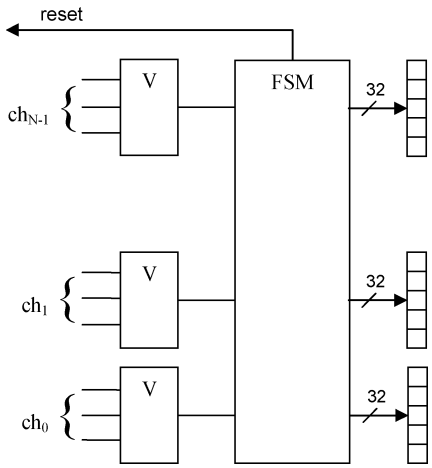

Fig. 4. Monitoring Design architecture.

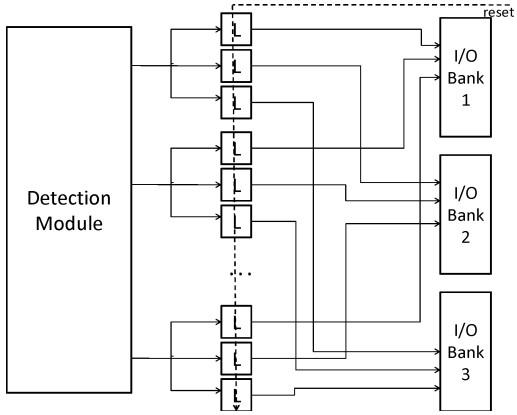

Fig. 5. DUT Design architecture.

```
int resourceRePlace(int d, Set R)
{
  static δ = 0;

  for each (r in R){
    if (r.unplaced == true){
      placeResource (r, d);
      δ = updateDistance();
    }
    δ = resourceRePlace(d, r.outputs());
  }

  return δ;
}
```

Fig. 6. Routing Modification Tool algorithm.

happens. Finally, the FSM stores the counters in 32-bit registers, one per channel, which are read at the end of the test by the PC. The MD design is modular to allow changing both the number of channels and the detection algorithm implemented by the FSM without need for changing the rest of the setup. Such flexibility allows us to observe different kind of SEEs simply reprogramming the MD. The main drawback of such a design is the so-called *blind time*. Indeed, between the moment in which the FSM detects the fault and the transmission stage of the DUT is reset, the MD is blind to new faults. However, because of the MD high operating frequency (at least 100 MHz), and considering that this operation requires few clock cycles, depending on the algorithm implemented by the FSM, the blind time is considerably much lower than the SEE rate of the DUT. The particle flux can be easily adjusted so that it is very unlikely to have an SEE occur within the blind time of the previous detected fault[1]. Fig. 4 shows the basic architecture of the MD as described above.

The circuit implemented in the DUT, is composed of two modules, a SEE detection module, which is the circuit whose behavior under radiation has to be studied, and a transmission module. The former should be as large as possible to expose a large sensitive area to radiation to maximize the probability of observing interesting events during testing, while the transmission part must be as small as possible to minimize the probability of collecting SEE that may be seen as measurement noise. For this reason, this module performs just the minimum to assure a correct data transmission, demanding further elaborations to the MD. Moreover, the transmission module is protected against SEE in order to send correct data. Because of the very narrow duration that certain transient faults can have, the transmission part contains latches that keep the fault active while the MD collects it.

As described above, as soon as the monitor notices a fault arrived on the DUT outputs, it resets the transmission module of the DUT, thus bringing the latches back to their initial state. Because this holding mechanism can be affected by SEUs, each latch is replicated three times, and the output of each replica is mapped on a pin belonging to a different FPGA I/O bank, in order to avoid common mode faults due to a single particle striking the I/O bank itself. Fig. 5 shows the general DUT architecture.

The design can work at different frequencies provided by the CGM. This allows the evaluation of the DUT SEE sensitivity with respect to the *frequency* parameter. By changing the clock frequency f_c, it is possible to perform static ($f_c = 0$ Hz) and dynamic ($f_c > 0$ Hz) tests. Possible faults on the circuit global lines (reset and clock), can be detected in the data post-processing phase because they usually induce burst of errors that are easy to distinguish.

B. Routing Modification/Analysis Tools (RMAT)

The second parameter whose impact on SEE sensitivity we evaluated is *routing*. We developed two software tools for modifying and analyzing the circuit routing architecture. The first one is able to re-place the circuit resources in order to change the routing of the connections, increasing or reducing its length by changing its topology. The second tool analyzes the generated bitstream for estimating the number of possibly sensitive configuration points that are changed between the first version of the circuit and the re-placed one.

The routing modification tool is based on the algorithm represented in Fig. 6.

Two parameters are passed to the $\mathrm{resourceRePlace}()$ function, the desired average distance between connected resources (d) and the set of resources to be re-placed (R). In the first iteration, R contains all the input resources of the circuit. For each resource belonging to R that has not already been placed, the

[1]With an average flux of $9.283\mathrm{E}+03\ \mathrm{particles/s/cm^2}$, we can expect an average number of $1.856\mathrm{E}-04\ \mathrm{particles/cm^2}$ in a blind interval of 20 ns.

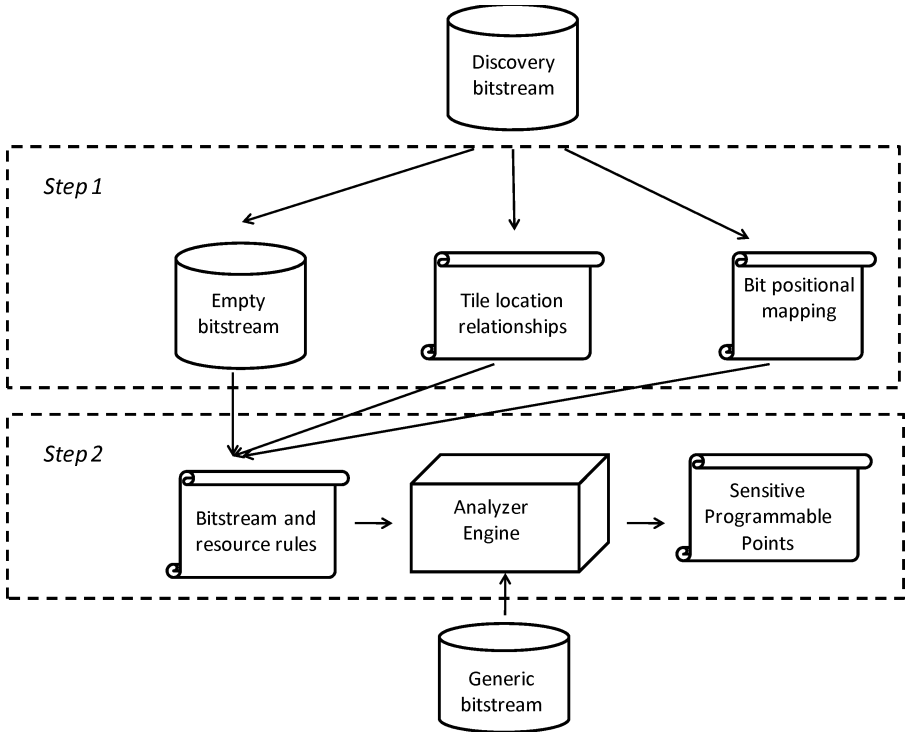

Fig. 7. Routing Analysis Tool flow.

$\mathrm{placeResource}()$ function is called, that inserts r in a spare place within the FPGA matrix. Afterwards, the $\mathrm{resourceRePlace}()$ function is called recursively on the resources connected to r outputs. Every new resource is then placed at a distance d from the resource it is connected to, randomly choosing one of the spare places that satisfy this requirement. If no places are found at the desired distance, a new random place is chosen at distance $\mathrm{d}+1$ or $\mathrm{d}-1$, and so on, until a place for the resource is found. Finally, after each placement operation, the function $\mathrm{updateDistance}()$ is called, to compute the actual average distance δ. When all the resources have been correctly placed, the actual average distance is returned.

The second software tool we developed is a bitstream analyzer able to identify sensitive programmable points of a certain FPGA design, i.e., configurable switches that can induce a fault if struck by a particle. The execution flow is composed by two steps: the *empty bitstream generation* and the *circuit sensitivity analysis*. The first step consists in gathering information about the bitstream structure of a certain device family. It is executed only once per family, and the information it provides are mandatory in order to extract the circuit used resources (both routing and logic ones). The second step, instead, analyzes the design itself and provides the list of sensitive programmable points, and is executed for every different circuit. Fig. 7 shows such flow.

During the first step, the primary phase consists in generating the "discovery" bitstream. A singular design is implemented, in order to discover the bitstream structure and generate the necessary information to proceed to the second step. This design consists in a unique spare gate (such as an inverter or an AND gate) that is simple enough to occupy a single tile of the whole device. Once the discovery bitstream has been generated, the design is moved around the device area in order to produce several copies of the same circuit with the only difference being their locations within the FPGA array. All these replicas are then compared and, on the basis of the regularity of the array architecture and, hypothetically, of the bitstream information, the tile location relationships and a bit positional mapping are produced. The location relationships define where the block of bits containing the information related to a certain tile is located within the bitstream. On the other hand, the bit mapping describes where the bits related to a certain tile are within the block.

Once the first step has been executed for the desired device, the output information can be saved and reused for every design implemented in the same family. The second step has to be executed for every design and analyzes the SEE sensitivity of the implemented circuit on the basis of the output information of the first step. In particular, for every tile of the device, the programmed bits are extracted by the analyzer engine and they are catalogued according to their function using the bitstream and resource rules generated by the first step. By now, the functions are recognized between routing and logic. The first function identifies the bits involved in routing elements, while the second one identifies the bits that program the logic cells.

V. Experimental Setup

On the basis of the radiation test environment described in the previous section, we studied the SEE sensitivity of realistic designs implemented in Flash-based FPGAs and, in particular, the contribution of two different effects: SEUs in the user memory elements and SETs in the logic. During the experiments we tested ProASIC3 Flash-based FPGAs [18], manufactured by Actel with 130-nm CMOS process. We irradiated A3P250 PQ2008 devices, featuring 6,144 logic cells and a 250 k equivalent system gates. So, the DUT is an Actel ProAsic3 250

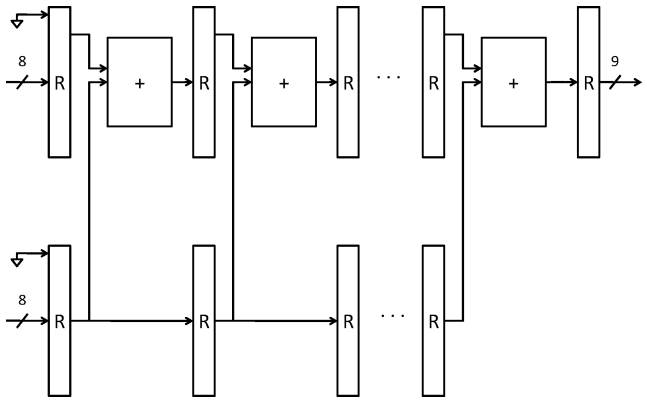

Fig. 8. Representation of the DUT detection module.

device, while the MD has been implemented in a Xilinx Virtex II-Pro 4 FPGA.

In the DUT, we mapped a sequential circuit implementing a multiplication between A and N, where A is the input of the circuit and N is a number defined by the amount of pipeline stages of the circuit. Indeed, every stage implements an addition between A and the result of the previous stage, thus emulating a multiplication of A by i, with i being the number of the current stage. A is a number on 8 bits, and the 9-th bits of the first-stage registers are set to '0' (carry input). In particular, A has been set to "0000001", in order to minimize the probability to have overflow, thus masking faults that are propagating toward the circuit outputs. Such a circuit is more similar to a real design than the circuits used in any previous experiment on Flash-based FPGAs [8], [9] but is still simple enough to allow studying the effects we are investigating by means of exhaustive simulation, in order to confirm and complement the radiation testing results. In this scenario, no feedback paths that usually characterize FSMs structures are present, but a combination of logic and memory elements is implemented, taking into account the maximum depth level of combinational logic, in order not to have unreal paths that can induce effects that are unlikely to happen in real designs. Fig. 8 shows the architecture of the detection module implemented in the DUT. Two replicas of this circuit are mapped in the DUT, and their outputs are XOR-ed in order to be able to catch differences in them. The comparing mechanism is part of the DUT transmission module and is thus triplicated to be insensitive to SEEs. Finally, the implemented multiplier is composed by 35 pipeline stages, and the two replicas occupy more than 95% of the whole FPGA logic cells.

Different kinds of effects can be detected by the designed DUT circuit. First of all, SEUs that occur in user registers; every flip-flop is susceptible to upset because it is not protected by any redundancy mechanism. SEUs can be observed as a single '1' (difference on one output between the two replicas) on the output for one clock cycle. On the other hand, SETs in the combinational logic can induce, if captured, single upsets or also multiple upsets within the same pipeline register, thus being observable as single or multiple '1's on the outputs for one clock cycle. Because SETs are dependent on the clock frequency, we tested the DUT with different frequencies, ranging from 1 kHz up to 100 MHz. Finally, faults affecting global lines, like clock and reset, can induce different behaviors. An SET on the clock line can induce a burst of errors, because many signals are sampled in a wrong manner, or, otherwise, a simple delay could be inserted. A transient fault on the reset line, on the other side, can clear all the pipeline registers, deleting all the faults that are propagating toward the output, thus masking them. In conclusion, being able to recognize burst errors, and taking into account the probability to have faults in the global lines, we can provide an accurate estimation of the number and type of radiation-induced events.

The Monitoring Design (MD) is a circuit that samples the DUT outputs and counts the number of errors appearing on them. As mentioned in the previous section, some precautions have been taken to cope with sampling frequency and I/O banks related issues. In particular we latched the DUT outputs to decouple the DUT functioning frequency and the MD sampling frequency. Because the DUT works at different frequencies during the test, while the MD samples always at 100 MHz, the latches and the last pipeline stage are reset after an error is detected onto the DUT outputs, not to sample it again. The latches have been then tripled in order to avoid SEUs that would have invalidated the results. Finally, as described in [3], I/O banks can be affected to common mode failures induced by a single particle that could lead to a malfunctioning of the whole bank. For this reason the three replicas of every output latch have been mapped on different banks and majority voted in the MD.

VI. Experimental Results and Analysis

In order to evaluate SEEs we performed two kinds of experiments. First of all we tested the DUT design under a heavy ion beam at different frequencies. We estimated the design cross-section as the ratio between the number of reported errors and the beam fluence at 1 kHz, 10 MHz, 50 MHz, 60 MHz, 80 MHz and 100 MHz. We used different parts in order to avoid measures affected by total dose effects, changing them as soon as any problem was detected during the configuration phase. Secondly, at a fixed frequency of 40 MHz, we tested the same circuit but with different placement schemes that stretch out or shorten the routing paths, in order to evaluate the impact of routing elements on the SEE sensitivity.

We performed heavy-ion irradiation both at the SIRAD Facility at the National Laboratory in Legnaro (INFN), Padova, Italy and at the Heavy-ion Irradiation Facility (HIF) located in Louvain-La-Neuve, Belgium. We first irradiated the DUT with an Iodine beam with an LET of 61.8 $\mathrm{MeV} \cdot \mathrm{cm}^2/\mathrm{mg}$. Several runs have been performed for every operating frequency and the results are presented in Fig. 9, which shows the measured design cross-section against the operating frequency. As the reader can observe, up to 50 MHz we did not observe relevant variations on the error rate, thus letting us think that below a certain frequency the cross-section is dominated by SEUs affecting the user memory elements. For higher frequencies we observed a rapid increase of the error rate. In particular, at 100 MHz it is about ten times the error rate at 1 kHz, and the growth is concentrated between 50 and 100 MHz.

In the second experiment, we irradiated the DUT at a fixed frequency of 40 MHz with Xenon ion (LET of 64.8 $\mathrm{MeV} \cdot$

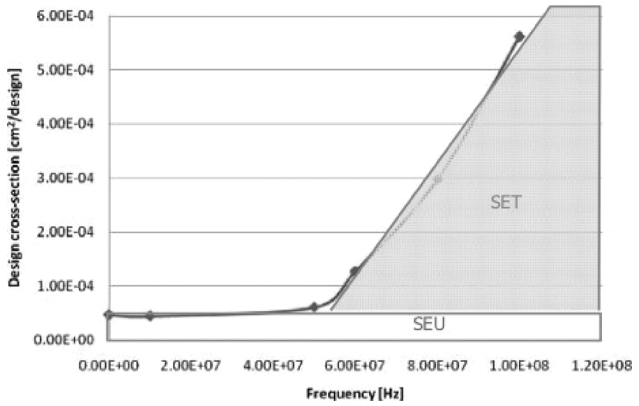

Fig. 9. Measured design SEE cross-section as a function of working frequency.

TABLE I
CHARACTERISTICS OF THE TESTED CIRCUITS.

Circuit	FFs Tiles[#]	Combinational Tiles [#]	Routing resources [#]
Circuit v1	2,484	3,405	126,840
Circuit v2	2,484	3,405	252,446

TABLE II
OBSERVED EVENTS WITH RESPECT TO DIFFERENT PLACEMENT SCHEMES.

Circuit	Observed events [#]	Design cross-section
Circuit v1	710	1.052 E-04
Circuit v2	729	1.217 E-04

$\mathrm{cm}^2/\mathrm{mg}$) but implementing different versions of the same design. In particular, the initial circuit has been placed in several manners in order to change the amount of used routing resources. We then put under the beam two versions of the circuit. As shown in Table I, the amount of combinational and sequential logic is the same, while the routing resources of the second version are twice the ones of the first.

Table II shows the absolute number of measured events and the design cross-section, computed as the ratio between the observed events and the total fluence, for the two versions of the circuit.

According to the obtained results we can conclude that from very low frequencies up to about 50 MHz there is a sort of constant cross-section that we expect due to SEUs in the user memory. This conclusion is bore also out by the results of the second experiment operated at 40 MHz, that show how changing the number of routing resources, thus increasing the SET sensitive area of the circuit, the design cross-section still remains the same. The conclusion goes in accordance with the results presented in [3]. The subsequent growth of the cross section above the threshold frequency can be due to the increasing contribution of SETs. The rising characteristic of the curve above 50 MHz can be explained as the increase of the probability to observe SET pulses at higher frequencies.

VII. CONCLUSIONS AND FUTURE WORKS

We defined a set of methodologies that can be applied in order to evaluate SEE sensitivity of Flash-based FPGAs. First of all, these general and flexible methodologies allow several variations requiring a very low effort, in order to study and analyze the desired kind of effect. Moreover, implementing such methodologies in the way we described in the previous sections of this paper, we could observe different kinds of SEEs, with respect to significant circuit parameters. In particular, interesting results have been collected about SEE sensitivity as a function of the DUT working frequency and the routing architecture. Finally, the DUT design we used allowed analyzing radiation effects in a circuit that is more realistic than the other ones used up to now.

In order to further investigate the observed radiation response we planned several activities. First of all we are going to perform more radiation experiments to increment the resolution of the growing error rate in the frequencies window between 50 MHz and 100 MHz. Such a test should confirm the SEE sensitivity characteristic we observed, enriching the curve in correspondence to other frequencies. Moreover we are going to irradiate different placement solutions at a higher frequency, where SETs should be more relevant, in order to evaluate the impact of differences in the routing paths. Finally, we are going to evaluate SEE propagation probability due to logic masking, in order to correlate current results with the one obtained with simpler circuits, like inverter chains or shift-registers.

REFERENCES

[1] R. Katz, K. LaBel, J. J. Wang, B. Cronquist, R. Koga, S. Penzin, and G. Swift, "Radiation effects on current field programmable technologies," *IEEE Trans. Nucl. Sci.*, vol. 44, no. 6, pt. 1, pp. 1945–1956, Dec. 1997.

[2] T. Speers, J. J. Wang, B. Cronquist, J. McCollum, H. Tseng, R. Katz, and I. Kleyner, "0.25 μm FLASH memory based FPGA for space applications," presented at the Military Aerosp. Program. Logic Devices Conf. (MAPLD), Greenbelt, MD, 1999.

[3] S. Rezgui, J. J. Wang, Y. Sun, B. Cronquist, and J. McCollum, "New reprogrammable and non-volatile radiation tolerant FPGA: RTA3P," in *Proc. IEEE Aerosp. Conf.*, Mar. 1–8, 2008, pp. 1–11.

[4] M. Nicolaidis, "Design for soft error mitigation," *IEEE Trans. Device Mater. Reliab.*, vol. 5, no. 3, pp. 405–418, Sep. 2005.

[5] B. Gilles, M. Nicolaidis, F. G. Wolff, C. A. Papachristou, and S. L. Garverick, "An efficient BICS design for SEUs detection and correction in semiconductor memories," in *Proc. Des. Autom. Test Eur. (DATE) Conf.*, Munich, Germany, Mar. 2004, pp. 592–597.

[6] T. Calin, M. Nicolaidis, and R. Velazco, "Upset hardened memory design for submicron CMOS technology," *IEEE Trans. Nucl.Sci.*, vol. 43, no. 6, pp. 2874–2878, Dec. 1996.

[7] P. Mongkolkachit and B. Bhuva, "Design technique for mitigation of alpha-particle-induced single-event transients in combinational logic," *IEEE Trans. Device Mater. Reliab.*, vol. 3, no. 3, pp. 89–92, Sep. 2003.

[8] S. Rezgui, J. J. Wang, Y. Sun, B. Cronquist, and J. McCollum, "Configuration and routing effects on the SET propagation in flash-based FPGAs," *IEEE Trans. Nucl. Sci.*, vol. 55, no. 6, pt. 1, pp. 3328–3335, Dec. 2008.

[9] N. Battezzati, S. Gerardin, A. Manuzzato, A. Paccagnella, S. Rezgui, L. Sterpone, and M. Violante, "On the evaluation of radiation-induced transient faults in flash-based FPGAs," in *14th IEEE Int. On-Line Test. Symp. (IOLTS)* , Jul. 7–9, 2008, pp. 135–140.

[10] V. F. Cavrois, V. Pouget, D. McMorrow, J. R. Schwank, N. Fel, F. Essely, R. S. Flores, P. Paillet, M. Gaillardin, D. Kobayashi, J. S. Melinger, O. Duhamel, P. E. Dodd, and M. R. Shaneyfelt, "Investigation of the propagation induced pulse broadening (PIPB) effect on single event transient in SOI and bulk inverter chains," *IEEE Trans. Nucl. Sci.*, vol. 55, no. 6, pt. 1, pp. 2842–2853, Dec. 2008.

[11] M. J. Gadlage, R. D. Schrimpf, J. M. Benedetto, P. H. Eaton, D. G. Mavis, M. Sibley, K. Avery, and T. L. Turflinger, "Single event transient pulse widths in digital microcircuits," *IEEE Trans. Nucl. Sci.*, vol. 51, no. 6, pt. 2, pp. 3285–3290, Dec. 2004.

[12] M. P. Baze, J. Wert, J. W. Clement, M. G. Hubert, A. Witulski, O. A. Amusan, L. Massengill, and D. McMorrow, "Propagating SET characterization technique for digital CMOS libraries," *IEEE Trans. Nucl. Sci.*, vol. 53, no. 6, pt. 1, pp. 3472–3478, Dec. 2006.

[13] S. Rezgui, J. J. Wang, Y. Sun, D. D'Silva, B. Cronquist, and J. McCollum, "A few guidelines for the SET characterization of non-volatile FPGAs: Lessons learned," presented at the Military Aerosp. Program. Logic Devices Conf. (MAPLD), Annapolis, MD, 2008.

[14] S. Rezgui, J. J. Wang, Y. Sun, B. Cronquist, and J. McCollum, "SET characterization and mitigation in RTAX-S antifuse FPGAs," in *Proc. IEEE Aerosp. Conf.*, Mar. 7–14, 2009, pp. 1–14.

[15] S. Rezgui, J. J. Wang, E. C. Tung, B. Cronquist, and J. McCollum, "New methodologies for SET characterization and mitigation in flash-based FPGAs," *IEEE Trans. Nucl. Sci.*, vol. 54, no. 6, pt. 1, pp. 2512–2524, Dec. 2007.

[16] M. Berg, J. J. Wang, R. Ladbury, S. Buchner, H. Kim, J. Howard, K. LaBel, A. Phan, T. Irwin, and M. Friendlich, "An analysis of single event upset dependencies on high frequency and architectural implementations within Actel RTAX-S family field programmable gate array," *IEEE Trans. Nucl. Sci.*, vol. 53, no. 6, pp. 3569–3574, Dec. 2006.

[17] B. Narasimham, M. J. Gadlage, B. L. Bhuva, R. D. Schrimpf, L. W. Massengill, W. T. Holman, A. F. Witulski, and K. F. Galloway, "Test circuit for measuring pulse widths of single-event transients causing soft errors," in *Proc. IEEE Int. Conf. Microelectron. Test Structures*, Mar. 2008, pp. 142–146.

[18] Actel Corporation, Mountain View, CA, "ProAsic3 flash family FPGAs datasheet," Oct. 2009.